B.IQ push button 3 - 5gang with RTR and display, Flush-mounted (Up) 7566359x, 7566459x, 7566559x

Technical Documentation

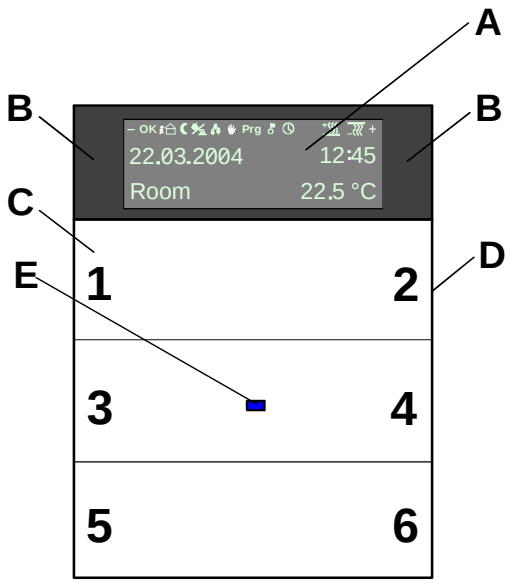
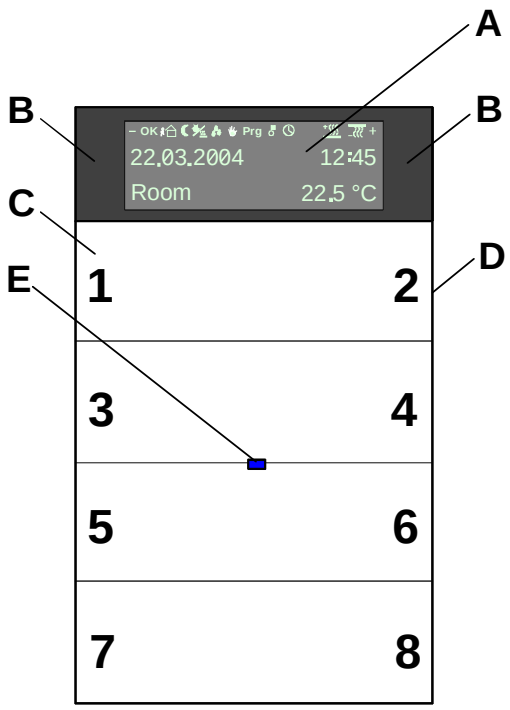
Functional description:

Push button functionality:
Depending on the parameterized software, the B.IQ push button RTR (Room Temperature Regulator) will send telegrams to the instabus EIB if one of its buttons is actuated. These can, for example, be telegrams for switching or touch control, dimming (also single-button dimming), or for Shutter control using various operating concepts. You can also program value transmitter functions such as value transmitter 1 bytes, light scene extensions for recalling externally or internally stored light scenes or value transmitter 2 bytes (for example, temperature or brightness value transmitters). In this connection, you can freely assign the buttons/rockers to the various functions. Distinction can be made between button and rocker functions.
In addition, the B.IQ Push button RTR offers the possibility to disable specific individual rockers or the entire push button. You can also operate the room temperature regulator integrated in the device by actuating the push button.

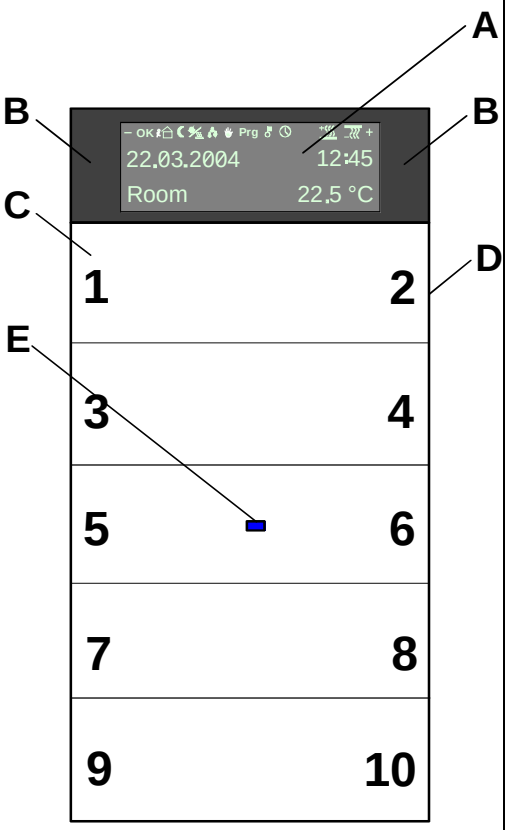
Room temperature regulator functionality:
You can use the B.IQ push button RTR for single-room temperature control. In this connection, the regulator can distinguish between and trigger up to two control circuits which optionally have their own temperature set values. Triggered by control circuit 1, the operating statuses and the operating modes of the two control circuits are switched over together. Thus, for example, you can use separate algorithms to control the radiators on the wall and the floor heating within one room.
Depending on the operating mode, the current temperature set value and on the room temperature, a variable for heating or cooling control can be sent to the instabus EIB for each of the two control circuits.
In a control circuit, the room temperature can be sensed either by the internal (in the push button enclosure) or by an optionally external temperature sensor. If the second control circuit is activated the room temperature of the first circuit will be sensed by the internal sensor, whereas the room temperature of the second circuit will be determined by the external sensor.
If you use only one control circuit you can activate another stage in addition to the heating or cooling basic stage to run an additional heater and/or cooling unit. In this connection, you can set the temperature set value difference between the basic and the additional stage by a parameter. For major deviations between the temperature set value and the actual temperature, you can activate this additional stage to heat up or cool down the room faster. You can assign different control algorithms to the basic and additional stages.

The regulator has five different operating modes (comfort, standby, night, frost/heat protection and regulator disabled) with their separate temperature set values for heating or cooling. For heating and cooling functions, you can select continuous or switching PI or switching 2-point control algorithms.
A room temperature timer allows automatic operating mode control, depending on the time of the day and on the day of the week.

General functions:
Two independent 1 bit or 1 byte timer functions with up to 14 different switching time events permit the time-dependent transmission of control commands to the bus. If desired, you can activate a push button assistance function. In this connection, when you actuate a button its function can be briefly read in the display. The display of a 14-byte text message (e. g. alarm message) received via the bus is also possible. You can switch the display background light.
If the B.IQ Push button RTR is pulled off the bus coupling unit an alarm message (of 1 bit or 1 byte type) can be sent.

Illustration:	**Dimensions:**	**Controls:**
3gang type  	Width: 88.5 mm Height: 118 mm Depth: 15 mm (without PEI)	A: Display B: Display buttons (on the right and left of the display) C: Rockers 1 – 3 (push button function) D: 3 x 2 status-LEDs (white) to indicate the statuses of the buttons or rockers E: 1 device ON LED (blue)
4gang type  	Width: 88.5 mm Height: 147.5 mm Depth: 15 mm (without PEI)	A: Display B: Display buttons (on the right and left of the display) C: Rockers 1 – 4 (push button function) D: 4 x 2 status-LEDs (white) to indicate the statuses of the buttons or rockers E: 1 ON indicator LED (blue)

5gang type



Width: 88.5 mm

Height: 177 mm

Depth: 15 mm (without PEI)

A: Display

B: Display buttons (on the right and left of the display)

C: Rockers 1 – 5 (push button function)

D: 5 x 2 status-LEDs (white) to indicate the statuses of the buttons or rockers

E: 1 device ON LED (blue)

Item numbers:

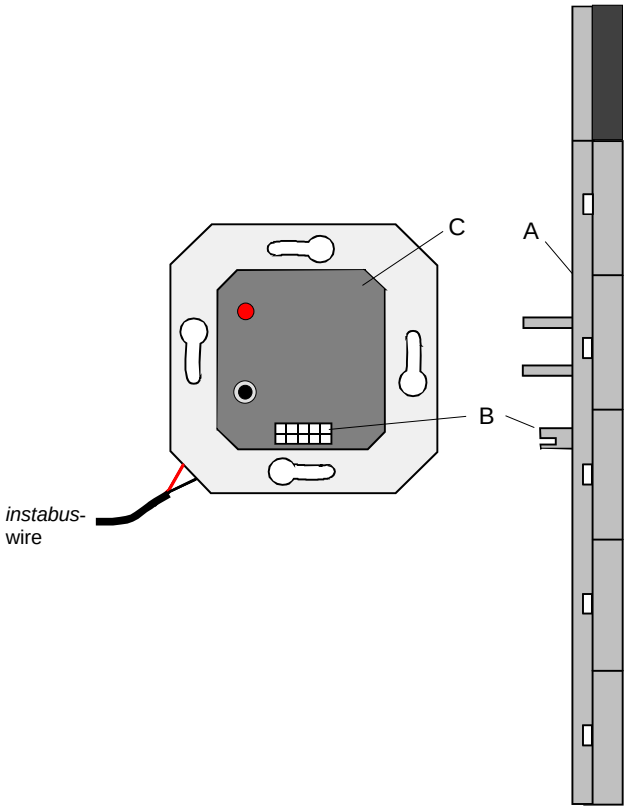
Push button	Berker order no.	
B.IQ push button 3gang with RTR and display	7566 35 9x	
B.IQ push button 4gang with RTR and display	7566 45 9x	
B.IQ push button 5gang with RTR and display	7566 55 9x	

Specifications	
Protective system:	IP 20
Protection class:	III
Mark of conformity:	EIB / KNX
Ambient temperature:	-5 °C to +45 °C
Storage/transport temperature:	-20 °C to +60 °C (storage above +45 °C will shorten life)
Fitting position:	Any (vertical to be preferred/display unit on top)
Minimum distances:	None
Type of fixing:	Plugging onto flush-mounted bus coupling unit (Please note: Remarks on the hardware)
instabus EIB supply	
Voltage:	21 – 32 VDC (through flush-mounted bus coupling unit)
Power consumption:	360 mW typ. (through flush-mounted bus coupling unit/2 bus loads)
Connection:	2 x 5-pole male connector (PEI)
External supply	---
Room temperature regulator (internal temperature sensor):	
Measuring range:	0 °C to + 40 °C ±1 %
Resolution:	0.1 K
Air humidity:	0 % to 95 % (no condensation)
Internal clock:	
Resolution:	1 minute
Time error:	8 minutes per day max. To keep the time error low you should set and thus update the internal clock via the bus every hour.
Response to voltage failure	
Bus voltage only:	All object values will be deleted. Push button function: No response, LEDs will go out. Room temperature regulator: No response, control OFF.
Mains voltage only:	---
Bus and mains voltages:	---
Response to recovery	
Bus voltage only:	Push button function: No response. Room temperature regulator: The regulator will initialize. Depending on the parameterization, various temperature values and the status will be sent and switch-over objects updated.
Mains voltage only:	---
Bus and mains voltages:	---
Input:	---
Output:	---

Wiring diagram: **Terminal assignment:**

The B.IQ push button with RTR and display must be plugged onto a flush-mounted bus coupling unit plus (BCU 1).

For example, 5gang type



A: B.IQ push button 3/4/5gang with RTR and display
B: User interface (PEI)
C: Bus coupling unit plus

Bus coupling unit plus

Berker ordering no.: 7504 00 03

Remarks on the hardware

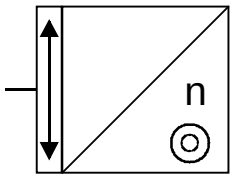
- The B.IQ push button with RTR and display may only be plugged onto the bus coupling unit plus (refer to the bus coupling unit illustration above and to its ordering number). The B.IQ push button with RTR and display will not work when it is plugged onto different flush-mounted bus coupling units.

Software Description

ETS search path:

Push button / B.IQ / B.IQ push button 3gang with RTR and display
Push button / Push button 3gang / B.IQ push button 3gang with RTR and display

ETS symbol:



PEI type	00 Hex	0 Dez	No adapter used.

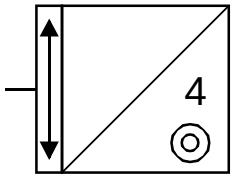
Application:

No.	Brief description:	Name:	Version:
1	B.IQ push button with RTR and display	B.IQ multifunktion RTR + display 161301	0.1

ETS search path:

Push button / B.IQ / B.IQ push button 4gang with RTR and display
Push button / Push button 4gang / B.IQ push button 4gang with RTR and display

ETS symbol:



PEI type	00 Hex	0 Dec	No adapter used.

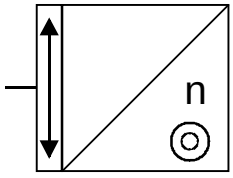
Application:

No.	Brief description:	Name:	Version:
1	B.IQ push button with RTR and display	B.IQ multifunktion RTR + display 161401	0.1

ETS search path:

Push button / B.IQ / B.IQ push button 5gang with RTR and display
Push button / Push button 5gang / B.IQ push button 5gang with RTR and display

ETS symbol:



PEI type	00 Hex	0 Dec	No adapter used.

Application:

No.	Brief description:	Name:	Version:
1	B.IQ push button with RTR and display	B.IQ multifunktion RTR + display 161501	0.1

Application:	1. B.IQ multifunktion RTR + display 161501 B.IQ multifunktion RTR + display 161501 B.IQ multifunktion RTR + display 161501		
Runable from screen form version:	1.2		
Number of addresses (max.):	77	**Dynamic table management**	Yes ☐ No ☒
Number of assignments (max.):	200	**Maximum table length**	77-BCU + 200-application µC
Communication objects:	71		

Push button functions:

The following objects solely apply to "actuation of the rockers = 2 push buttons (2 objects)":

Function: No function (for all buttons [1])

Object	Function	Name	Type	Flag
0-9	Status	Push Push button 1 – push button 10 [1]	1 bit	C, W

Function: Switching/pushing (for all buttons [1])

Object	Function	Name	Type	Flag
0-9	Switching	Push Push button 1 – push button 10 [1]	1 bit	C, W, T

Function: Dimming (for all buttons [1])

Object	Function	Name	Type	Flag
0-9	Switching	Push Push button 1 – push button 10 [1]	1 bit	C, W, T
10-19	Dimming	Push Push button 1 – push button 10 [1]	4 bit	C, T

Function: Shutter (for all buttons [1])

Object	Function	Name	Type	Flag
0-9	Step operation	Push Push button 1 – push button 10 [1]	1 bit	C, T
10-19	Move operation	Push Push button 1 – push button 10 [1]	1 bit	C, T

Function: Value transmitter 1byte (for all buttons [1])

Object	Function	Name	Type	Flag
0-9	Value transmitter	Push Push button 1 – push button 10 [1]	1 byte	C, T

Function: Value transmitter 2byte (for all buttons [1])

Object	Function	Name	Type	Flag
0-9	Value transmitter [temp./brightness]	Push Push button 1 – push button 10 [1]	2-byte	C, T

Function: Light scene extension/recall (for all buttons [1])

Object	Function	Name	Type	Flag
0-9	Scene extension [2]	Push Push button 1 – push button 10 [1]	1 byte	C, T

Function: Operating mode switch-over / timer operation / room temperature timer operation (for all buttons [1])

Object	Function	Name	Type	Flag
No further objects for the push button functionality.				

[1]: The "no function", "switching/pushing", "dimming", "shutter", "value transmitter 1 byte", "value transmitter 2 byte", "light scene extension/recall", "operating mode switch-over", "room temperature time operation" and "timer operation" functions can be selected per push button. The names of the communication objects and the object table (dynamic object structure) will change correspondingly. You can also combine button or rocker functions, respectively.

[2]: The light scene extension object will be hidden when you have set "function as = internal light scene recall".

The following objects solely apply to "actuation of the rockers = rocker (1 object)":

Function: No function (for all rockers [3])

Object	Function	Name	Type	Flag
1/3/5/7/9	Status	Rocker 1 - rocker 5 [3]	1 bit	C, W

Function: Switching (for all rockers [3])

Object	Function	Name	Type	Flag
0/2/4/6/8	Switching	Rocker 1 - rocker 5 [3]	1 bit	C, W, T
1/3/5/7/9	Status	Rocker 1 - rocker 5 [3]	1 bit	C, W

Function: Dimming (for all rockers [3])

Object	Function	Name	Type	Flag
0/2/4/6/8	Switching	Rocker 1 - rocker 5 [3]	1 bit	C, W, T
1/3/5/7/9	Status	Rocker 1 - rocker 5 [3]	1 bit	C, W
10/12/14/16/18	Dimming	Rocker 1 - rocker 5 [3]	4-bit	C, T

Function: Shutter (for all rockers [3])

Object	Function	Name	Type	Flag
0/2/4/6/8	Step operation	Rocker 1 - rocker 5 [3]	1 bit	C, T
1/3/5/7/9	Status	Rocker 1 - rocker 5 [3]	1 bit	C, W
10/12/14/16/18	Move operation	Rocker 1 - rocker 5 [3]	1 bit	C, T

Function: Operating mode switch-over (for all rockers [3])

Object	Function	Name	Type	Flag
No further objects for the push button functionality.				

[3]: The "no function", "switching", "dimming", "shutter" and "operating mode switch-over" functions can be selected per rocker. The names of the communication objects and the object table (dynamic object structure) will change correspondingly. You can also combine button or rocker functions, respectively.

For "rocker or button function = without function" for all rockers / buttons no objects for buttons or rockers are available.

The following objects are provided for the disabling function (push button functionality), the display background light, or for the alarm function, respectively.

Function: Alarm message (data format: switching telegram, 1 bit)

Object	Function	Name	Type	Flag
20	Switching	Alarm signal	1 bit	C, T [4]

Function: Alarm message (data format: value telegram, 1 byte)

Object	Function	Name	Type	Flag
20	Value transmitter	Alarm signal	1 byte	C, T [4]

Function: Alarm message (data format: value telegram, 2 byte)

Object	Function	Name	Type	Flag
20	Value transmitter	Alarm signal	2-byte	C, T [4]

Function: Disabling function

Object	Function	Name	Type	Flag
21	Disabling function	Disabling push button sensor	1 bit	C, W

Function: Switching the display background light

Object	Function	Name	Type	Flag
22	Switching	Switching display	1 bit	C, W

[4]: Only a group address can be assigned to the "alarm signal" object.

"R" flag (read flag): The objects can only be read out when the push button has been plugged on.

Room temperature regulator functions:

Function: Actual temperature

Object	Function	Name	Type	Flag
23	Actual temperature	Measured and matched value	2-byte	C, R, T

Function: Additional temperature sensor

Object	Function	Name	Type	Flag
24	External temperature sensor	Temperature value	2-byte	C, W, T
25	External temperature sensor (for outside temperature)	Temperature value	2-byte	C, W

Function: Basic set value selection

Object	Function	Name	Type	Flag
26	Basic set value	Temperature default value	2-byte	C, W
27	Basic set value 2nd control circuit[5]	Temperature default value	2-byte	C, W

Function: Operating mode switch-over

For "via value (1 byte)" operating mode switch-over:

Object	Function	Name	Type	Flag
28	Operating mode switch-over	KONNEX switch-over	1 byte	C, W, (T)[6]
32	Override object operating mode	KONNEX switch-over	1 byte	C, W

For "via switching (4 x 1 bit)" operating mode switch-over:

Object	Function	Name	Type	Flag
28	Comfort operation	Operating mode switch-over	1 bit	C, W, (T)[6]
29	Standby operation	Operating mode switch-over	1 bit	C, W, (T)[6]
30	Night operation	Operating mode switch-over	1 bit	C, W, (T)[6]
31	Frost-/heat protection	Operating mode switch-over	1 bit	C, W, (T)[6]

Presence object and window status:

Object	Function	Name	Type	Flag
33	Presence object	Presence button / detector	1 bit	C, W, T
34	Window status	Window contact	1 bit	C, W

Function: Operating mode switching

Object	Function	Name	Type	Flag
35	Heating/cooling [7]	Heating/cooling switch-over	1 bit	C, W, (T)

Function: Status information

Object	Function	Name	Type	Flag
36	Controller status	Status indication general	1 byte	C, T
36	Controller status	Status indication single	1 bit	C, T
37	Heating indication	Indication	1 bit	C, T
38	Cooling indication	Indication	1 bit	C, T

[5]: This object will only be active if you have activated the second control circuit, and if both circuits have separate set values.

[6]: As an option, you can set the "T" flags for the operating mode switch-over objects. After you have set the flags, the object values changed in accordance with the reset operating mode will be actively transmitted to the bus.

[7]: This object will only be active for the mixed "heating and cooling" or "basic/additional heating/cooling" mode if one control circuit is used. The "T" flag (transmission flag) will be set when the automatic heating/cooling switch-over is active.

Function: Disabling function (room temperature regulator)

Object	Function	Name	Type	Flag
39	Disabling controller operation	Disabling function	1 bit	C, W
40	Disabling controller	Disabling function	1 bit	C, W
41	Disabling additional stage [8]	Disabling function	1 bit	C, W
41	Disabling 2nd control circuit [8]	Disabling function	1 bit	C, W

Function: Heating variable

No additional stage activated / one control circuit /

For mixed operation: Output of "heating" and "cooling" variables through separate objects:

Object	Function	Name	Type	Flag
42	Heating (1st control circuit)	Continuous variable	1 byte	C, W, T
42	Heating (1st control circuit)	PWM variable	1 bit	C, W, T
42	Heating (1st control circuit)	Switching variable	1 bit	C, W, T

No additional stage activated/one control circuit/

For mixed operation: Output of "heating" and "cooling" variables through one common object:

Object	Function	Name	Type	Flag
42	Heating/cooling (1st control circuit)	Continuous variable	1 byte	C, W, T
42	Heating/cooling (1st control circuit)	PWM variable	1 bit	C, W, T
42	Heating/cooling (1st control circuit)	Switching variable	1 bit	C, W, T

Additional stage activated / one control circuit /

For mixed operation: Output of "heating" and "cooling" variables through separate objects:

Object	Function	Name	Type	Flag
42	Basic heating (1st control circuit)	Continuous variable	1 byte	C, W, T
42	Basic heating (1st control circuit)	PWM variable	1 bit	C, W, T
42	Basic heating (1st control circuit)	Switching variable	1 bit	C, W, T
43	Additional heating (1st control circuit)	Continuous variable	1 byte	C, W, T
43	Additional heating (1st control circuit)	PWM variable	1 bit	C, W, T
43	Additional heating (!ST control circuit)	Switching variable	1 bit	C, W, T

Two control circuits:

Object	Function	Name	Type	Flag
42	Heating (1st control circuit)	Continuous variable	1 byte	C, W, T
42	Heating (1st control circuit)	PWM variable	1 bit	C, W, T
42	Heating (1st control circuit)	Switching variable	1 bit	C, W, T
46	Heating (2nd control circuit)	Continuous variable	1 byte	C, W, T
46	Heating (2nd control circuit)	PWM variable	1 bit	C, W, T
46	Heating (2nd control circuit)	Switching variable	1 bit	C, W, T

[8]: This object will only be visible when you have activated the additional stage or, alternatively, if you use two control circuits.

Additional stage activated/one control circuit/
For mixed operation: Output of "heating" and "cooling" variables through one common object:

Object	Function	Name	Type	Flag
42	Basic heating and cooling (1st control circuit)	Continuous variable	1 byte	C, W, T
42	Basic heating and cooling (1st control circuit)	PWM variable	1 bit	C, W, T
42	Basic heating and cooling (1st control circuit)	Switching variable	1 bit	C, W, T
43	Additional heating and cooling (1st control circuit)	Continuous variable	1 byte	C, W, T
43	Additional heating and cooling (1st control circuit)	PWM variable	1 bit	C, W, T
43	Additional heating and cooling (1st control circuit)	Switching variable	1 bit	C, W, T

Function: Cooling variable.
No additional stage activated / one control circuit /
For mixed operation: Output of "heating" and "cooling" variables through separate objects:

Object	Function	Name	Type	Flag
44	Cooling (1st control circuit)	Continuous variable	1 byte	C, W, T
44	Cooling (1st control circuit)	PWM variable	1 bit	C, W, T
44	Cooling (1st control circuit)	Switching variable	1 bit	C, W, T

Additional stage activated / one control circuit /
For mixed operation: Output of "heating" and "cooling" variables through separate objects:

Object	Function	Name	Type	Flag
44	Basic cooling (1st control circuit)	Continuous variable	1 byte	C, W, T
44	Basic cooling (1st control circuit)	PWM variable	1 bit	C, W, T
44	Basic cooling (1st control circuit)	Switching variable	1 bit	C, W, T
45	Additional cooling (1st control circuit)	Continuous variable	1 byte	C, W, T
45	Additional cooling (1st control circuit)	PWM variable	1 bit	C, W, T
45	Additional cooling (1st control circuit)	Switching variable	1 bit	C, W, T

Two control circuits:

Object	Function	Name	Type	Flag
44	Cooling (1st control circuit)	Continuous variable	1 byte	C, W, T
44	Cooling (1st control circuit)	PWM variable	1 bit	C, W, T
44	Cooling (1st control circuit)	Switching variable	1 bit	C, W, T
48	Cooling (2nd control circuit)	Continuous variable	1 byte	C, W, T
48	Cooling (2nd control circuit)	PWM variable	1 bit	C, W, T
48	Cooling (2nd control circuit)	Switching variable	1 bit	C, W, T

Function: Heating status information variable [9]

Object		Function	Name	Type	Flag
	46	Heating (1st control circuit)	PWM variable	1 byte	C, W, T
	46	Basic heating (1st control circuit)	PWM variable	1 byte	C, W, T
	47	Additional heating (1st control circuit)	PWM variable	1 byte	C, W, T

Function: Cooling status information variable [9]

Object		Function	Name	Type	Flag
	48	Cooling (1st control circuit)	PWM variable	1 byte	C, W, T
	48	Basic cooling (1st control circuit)	PWM variable	1 byte	C, W, T
	49	Additional cooling (1st control circuit)	PWM variable	1 byte	C, W, T

Function: Set value temperature

Object		Function	Name	Type	Flag
	50	Set value temperature	Temperature value	2-byte	C, T, R
	51	Set value temperature 2nd control circuit [10]	Temperature value	2-byte	C, T, R

Function: Time signal

Object		Function	Name	Type	Flag
	52	Time	Time	3-byte	C, W

Function: Date signal

Object		Function	Name	Type	Flag
	53	Date	Date	3-byte	C, W

Timers

Function: Switching for timer 1 or 2 [11]

Object		Function	Name	Type	Flag
	54	Switching	Timer 1	1 bit	C, T
	56	Switching	Timer 2	1 bit	C, T

Function: Value for timer 1 or 2 [11]

Object		Function	Name	Type	Flag
	54	Value	Timer 1	1 byte	C, T
	56	Value	Timer 2	1 byte	C, T

Function: Scene recall for timer 1 or 2 [11]

Object		Function	Name	Type	Flag
	54	Scene extension	Timer 1	1 byte	C, T
	56	Scene extension	Timer 2	1 byte	C, T

Disabling functions for timer 1 or 2:

Object		Function	Name	Type	Flag
	55	Disabling function	Disabling timer 1	1 bit	C, W
	57	Disabling function	Disabling timer 2	1 bit	C, W

[9]: The status information on the PWM variable is only possible for one control circuit.

[10]: This object will only be active if you have activated the second control circuit, and if both circuits have separate set values.

[11]: The "switching", "value" and "scene recall" can be set separately per timer.

Function: Room temperature timer

Object	Function	Name	Type	Flag
58	Disabling function	Disabling room temperature timer	1 bit	C, W

Scene Function

Function: Switching (for all 8 scene objects [12])

Object	Function	Name	Type	Flag
59	Scene output 1	Switching	1 bit	C, W, T
60	Scene output 2	Switching	1 bit	C, W, T
61	Scene output 3	Switching	1 bit	C, W, T
62	Scene output 4	Switching	1 bit	C, W, T
63	Scene output 5	Switching	1 bit	C, W, T
64	Scene output 6	Switching	1 bit	C, W, T
65	Scene output 7	Switching	1 bit	C, W, T
66	Scene output 8	Switching	1 bit	C, W, T

Function: Value (for all 8 scene objects [12])

Object	Function	Name	Type	Flag
59	Scene output 1	Value	1 byte	C, W, T
60	Scene output 2	Value	1 byte	C, W, T
61	Scene output 3	Value	1 byte	C, W, T
62	Scene output 4	Value	1 byte	C, W, T
63	Scene output 5	Value	1 byte	C, W, T
64	Scene output 6	Value	1 byte	C, W, T
65	Scene output 7	Value	1 byte	C, W, T
66	Scene output 8	Value	1 byte	C, W, T

Function: Shutter position (for all 8 scene objects [12])

Object	Function	Name	Type	Flag
59	Scene output 1	Shutter position	1 bit	C, W, T
60	Scene output 2	Shutter position	1 bit	C, W, T
61	Scene output 3	Shutter position	1 bit	C, W, T
62	Scene output 4	Shutter position	1 bit	C, W, T
63	Scene output 5	Shutter position	1 bit	C, W, T
64	Scene output 6	Shutter position	1 bit	C, W, T
65	Scene output 7	Shutter position	1 bit	C, W, T
66	Scene output 8	Shutter position	1 bit	C, W, T

Function: Scene extension

Object	Function	Name	Type	Flag
67	Scene extension	Extension input	1 byte	C, W

General

Function: Text message

Object	Function	Name	Type	Flag
68	Text indication	Display alarm indication	14-byte	C, W

[12]: You can set the "switching", "value" and "Shutter position" functions for each scene object.

Temperature Monitoring

Function: Temperature alarm

Object	Function	Name	Type	Flag
69	Temperature alarm 1	Switching	1 bit	C, T
70	Temperature alarm 2	Switching	1 bit	C, T

Object description
Objects:

0 - 9	Status:	1 bit object for triggering the status-LED of a button or a rocker, resp.
0 - 9	Switching:	1 bit object for sending switching telegrams (ON, OFF).
10 - 19	Dimming:	4-bit object for relative brightness changing between 0 and 100 %.
0 - 9	Step operation:	1 bit object for the step mode (step) of a Shutter.
10 - 19	Move operation:	1 bit object for the move mode (MOVE) of a Shutter.
0 - 9	Scene extension:	1 byte object for recalling or storing (light) scenes (1 - 64).
0 - 9	Value transmitter:	1 byte object for sending dimming value telegrams (0 - 255), for example.
0 - 9	Value transmitter [temp./brightness]:	1 byte object for sending temperature values (0 - 40 °C), brightness values (0 - 1500 lux), or for transmitting 2-byte values (0 - 65535).
20	Switching:	1 bit object for sending an alarm signal.
20	Value transmitter:	1 byte object for sending an alarm message.
20	Value transmitter:	2-byte object for sending an alarm message.
21	Disabling function:	1 bit object for disabling buttons or rockers of the push button.
22	Switching:	1 bit object for switching the display background light.
23	Actual temperature:	2-byte object for the output of the actual temperature. (Possible value range: -99.9 °C to +99.9 °C / internal temperature sensor measuring range: 0 °C to + 40 °C ±1 %.)
24	External temperature sensor:	2-byte object for coupling an external room temperature sensor. (Possible value range: -99.9 °C to +99.9 °C.)
25	External temperature sensor:	2-byte object for coupling an outside temperature sensor. (Possible value range: -99.9 °C to +99.9 °C.)
26	Basic set value:	2-byte object for basic set value external preselection. Depending on the operating mode, the possible value range is restricted by the parameterized frost protection and/or heat protection temperature.
27	Basic set value 2nd control circuit:	2-byte object for basic set value external preselection for the second control circuit using its own set values. Depending on the operating mode, the possible value range is restricted by the parameterized frost protection and/or heat protection temperature.
28	Operating mode switchover:	1 byte object for changing the operating modes to KONNEX.
28	Comfort operation:	1 bit object for changing into the "comfort" mode.
29	Standby operation:	1 bit object for changing into the "standby" mode.
30	Night operation:	1 bit object for changing into the "night" mode.
31	Frost/heat protection:	1 bit object for changing into the "frost/heat protection" mode.
32	Override object operating mode:	1 byte object for higher-order forced control of the operating modes to KONNEX.
33	Presence object:	1 bit object (bidirectional) which sends the state of the presence button to the bus (if the presence object has been enabled the presence button can be parameterized under the push button functionality), or through which a presence detector can, for example, be connected. (Presence = "1", no presence = "0".)
34	Window status:	1 bit object for the connection of window contacts. (Window open = "1", window closed = "0".)

Object description (continued)

Objects:

	35	Heating/cooling:	1 bit object for switching over between the "heating" and "cooling" modes if this is not done automatically by the regulator (object value 1: heating; object value 0: cooling). During automatic switch-over, the active operating mode can be transmitted (depending on the parameter).
	36	Controller status:	1 byte object for general status feedback, or 1 bit object for individual status feedback of parameterized regulator functions.
	37	Heating indication:	1 bit object for indicating whether heating energy is requested (object value = "1": energy request; object value = "0": no energy request).
	38	Cooling indication:	1 bit object for indicating whether cooling energy is requested (object value = "1": energy request; object value = "0": no energy request).
	39	Disabling controller operation:	1 bit object for disabling regulator local operation. (Regulator operation disabled ="1", regulator operation enabled = "0".)
	40	Disabling controller:	1 bit object for deactivating the regulator (activation of dew-point operation). (Regulator deactivated = "1", regulator activated = "0".)
	41	Disabling additional stage:	1 bit object for disabling the additional stage of the regulator. (Additional stage deactivated = "1", additional stage activated = "0".)
	41	Disabling 2nd control circuit:	1 bit object for deactivating the second control circuit. (Control circuit 2 deactivated = "1", control circuit 2 activated = "0".)
	42	Heating (1st control circuit):	1 byte object for the output of a continuous heating mode variable of the first control circuit.
	42	Heating (1st control circuit):	1 bit object for the output of the heating mode switching or PWM variable of the first control circuit.
	42	Basic heating (1st control circuit):	1 byte object for the output of the basic heating mode continuous variable of the first control circuit.
	42	Basic heating (control circuit):	1 bit object for the output of the basic heating mode switching or PWM variable of the first control circuit.
	42	Heating/cooling (1st control circuit):	1 byte object for the output of the alternative heating or cooling mode continuous variable of the first control circuit. (For variable output through common object.)
	42	Heating/cooling (1st control circuit 1):	1 bit object for the output of the alternative heating or cooling mode switching or PWM variable of the first control circuit. (For variable output through common object.)
	42	Basic heating and cooling (1st control circuit):	1 byte object for the output of the alternative basic heating or basic cooling mode continuous variable of the first control circuit. (For variable output through common object.)
	42	Basic heating and cooling (1st control circuit):	1 bit object for the output of the alternative basic heating or basic cooling mode switching or PWM variable of the first control circuit. (For variable output through common object.)
	43	Additional heating (1st control circuit):	1 byte object for the output of a additional heating mode continuous variable of the first control circuit.
	43	Additional heating (1st control circuit):	1 bit object for the output of the additional heating mode switching or PWM variable of the first control circuit.
	43	Additional heating and cooling (1st control circuit):	1 byte object for the output of the alternative additional heating or additional cooling mode continuous variable of the first control circuit. (For variable output through common object.)
	43	Additional heating and cooling (1st control circuit):	1 bit object for the output of the alternative additional heating or additional cooling mode switching or PWM variable of the first control circuit. (For variable output through common object.)

Object description (continued)

Objects:

☐←	44	Cooling (1st control circuit):	1 byte object for the output of the cooling mode continuous variable of the first control circuit.
☐←	44	Cooling (1st control circuit):	1 bit object for the output of the cooling mode switching or PWM variable of the first control circuit.
☐←	44	Basic cooling (1st control circuit):	1 byte object for the output of the basic cooling mode continuous variable of the first control circuit.
☐←	44	Basic cooling (1st control circuit):	1 bit object for the output of the basic cooling mode switching or PWM variable of the first control circuit.
☐←	45	Additional cooling (1st control circuit):	1 byte object for the output of the additional cooling mode continuous variable of the first control circuit.
☐←	45	Additional cooling (1st control circuit):	1 bit object for the output of the additional cooling mode switching or PWM variable of the first control circuit.
☐←	46	Heating (2nd control circuit):	1 byte object for the output of the heating mode continuous variable of the second control circuit.
☐←	46	Heating (2nd control circuit):	1 bit object for the output of the heating mode switching or PWM variable of the second control circuit.
☐←	46	Heating (1st control circuit):	1 byte object for the PWM variable to feed back the continuous variable value for the heating mode. (Only for one control circuit.)
☐←	46	Basic heating (1st control circuit):	1 byte object for the PWM variable to feed back the continuous variable value for the basic heating mode. (Only for one control circuit.)
☐←	47	Additional heating (1st control circuit):	1 byte object for the PWM variable to feed back the continuous variable value for the additional heating mode. (Only for one control circuit.)
☐←	48	Cooling (2nd control circuit):	1 byte object for the output of the cooling mode continuous variable of the second control circuit.
☐←	48	Cooling (2nd control circuit):	1 bit object for the output of the cooling mode switching or PWM variable of the second control circuit.
☐←	48	Cooling (1st control circuit):	1 byte object for the PWM variable to feed back the continuous variable value for the cooling mode. (Only for one control circuit.)
☐←	48	Basic cooling (1st control circuit):	1 byte object for the PWM variable to feed back the continuous variable value for the basic cooling mode. (Only for one control circuit.)
☐←	49	Additional cooling (1st control circuit):	1 byte object for the PWM variable to feed back the continuous variable value for the additional cooling mode. (Only for one control circuit.)
☐→	50	Set value temperature:	2-byte object for the output of the current temperature set value of the first control circuit. Depending on the operating mode, the possible value range is restricted by the parameterized frost protection and/or heat protection temperature.
☐→	51	Set value temperature 2nd control circuit:	2-byte object for the output of the current temperature set value of the second control circuit. Depending on the operating mode, the possible value range is restricted by the parameterized frost protection and/or heat protection temperature.
☐←	52	Time:	3-byte object for receiving the current time via the bus.
☐←	53	Date:	3-byte object for receiving the current time via the bus.

Object description (continued)

Objects:

☐	54	Switching:	1 bit object for transmitting the switching command of the first timer.
☐	54	Value:	1 bit object for transmitting the value command of the first timer.
☐	54	Scene extension:	1 bit object for transmitting the scene command of the first timer.
☐→	55	Disabling function:	1 bit object for disabling the first timer. (The polarity can be parameterized.)
☐	56	Switching:	1 bit object for transmitting the switching command of the second timer.
☐	56	Value:	1 bit object for transmitting the value command of the second timer.
☐	56	Scene extension:	1 bit object for transmitting the scene command of the second timer.
☐→	57	Disabling function:	1 bit object for disabling the second timer. (The polarity can be parameterized.)
☐→	58	Disabling function:	1 bit object for disabling the room temperature regulator timer. (The polarity can be parameterized.)
☐	59 – 66	Scene outputs 1 - 8:	1 bit object for transmitting the up to eight switching commands of a scene.
☐	59 – 66	Scene outputs 1 - 8:	1 byte object for transmitting the up to eight value commands of a scene.
☐	59 – 66	Scene outputs 1 - 8:	1 bit objects for transmitting the up to eight Shutter move commands of a scene.
☐→	67	Scene extension:	1 byte object for externally recalling or saving the internally stored eight light scenes.
☐→	68	Text indication:	14-byte object for receiving a 14-character display text (e. g. alarm message).
☐	69	Switching:	1 bit object for transmitting a temperature monitoring switching telegram (temperature alarm 1/lower temperature value).
☐	70	Switching:	1 bit object for transmitting a temperature monitoring switching telegram (temperature alarm 2/upper temperature value).

Function Scope

Push button functionality

- **General**
  - Free assignment of the switching/pushing, dimming, shutter, light scene extension/recall, value transmitter 1 byte, value transmitter 2 byte and operating mode switch-over functions to the push buttons when the button function has been activated.
  - Free assignment of the switching, dimming, shutter and operating mode switch-over functions to the rockers when the rocker function has been activated.
  - Status indication possible by 6 (3gang type), 8 (4gang type) or 10 (5gang type) white LEDs (status indication for rocker function possible via status objects and status or actuation indication possible for push button function).
  - Even though certain push buttons or rockers have been given "no function", the status-LEDs can be triggered via objects.
  - An disabling object for disabling individual rockers is available (the polarity of the disabling object being selectable).

- **Switching/pushing function**
  - You can select the command to be raised when you press or release a button (ON, OFF, TOGGLE, no function).
  - Single-button operation is possible for the rocker function (only for "command on pressing a rocker = left = TOGGLE, right = TOGGLE").
  - You can parameterize the function of the status-LED for the button function or of the Status indication for the rocker function, respectively.

- **Dimming function**
  - You can adjust the time between dimming and switching and the dimming step width.
  - Telegram repetition and sending stop telegrams are possible.
  - Single-button operation is possible for the rocker function (only for "command on pressing a button = left = TOGGLE, right = TOGGLE").
  - You can parameterize the function of the status-LED for the button function or of the Status indication for the rocker function, respectively.

- **Shutter function**
  - You can select the button function (UP, DOWN, TOGGLE).
  - You can parameterize the operating concept (STEP – MOVE – STEP or MOVE – STEP, respectively).
  - The time between step and move operation is adjustable (for STEP – MOVE – STEP only).
  - The lamella adjustment (time during which a move command can be completed by releasing the button) can be preset.
  - You can parameterize the function of the status-LED for the push button function or of the Status indication for the rocker function, respectively.

- **value transmitter 1 byte / light scene extension/recall function (for button function only)**
  - Value transmitter (1 byte) or light scene recalling with/without storage button functions can be parameterized
  - Move button actuation will enable you to change values for the value transmitter function.
  - You can parameterize the function of the status-LED.
  - When the light scene recall function is active, you can also recall internal scenes.

- **value transmitter function 2 byte (for button function only)**
  - You can parameterize the EIS 5 brightness value transmitter, EIS 5 temperature value transmitter and EIS 10 value transmitter 2 byte functions.
  - Move button actuation will enable you to change values.
  - You can parameterize the function of the status-LED.

Room temperature regulator functionality:

- **General**
  - Five operating modes: Comfort, standby, night, frost/heat protection and controller disabled (disabling e. g. for dew-point operation).
  - Switching-over of the operating modes to KONNEX by a 1 byte object or by individual 1 bit objects.
  - Reading of the room temperature regulator information through an integrated display unit.
  - Local programming mode is possible. It can be activated or deactivated.
  - "No operation": 'Normal operation' and local operation of the regulator by actuation of the display buttons for shifting the basic set value possible.
  - "Limited operation": Switching over into the programming mode is possible. → 'Normal operation' including set value shifting and switching over the operating mode and resetting of the different heating and/or cooling set values is possible.
  - "Complete operation": Full access to the device with local operation. In addition to restricted operation, it grants the user access to the up to three timers (activation/deactivation of the timers and changing of the individual switching programs) and to the "settings" menu (activation/deactivation of the button assistance function, the LCD background light and the button lock as well as of the display of the version data).

- **Heating/cooling system**
  - Operating modes: "Heating", "cooling", "heating and cooling", each of them with or without additional stage.
  - Up to two control circuits with optionally different temperature set values and operating mode common switch-over are possible.
    (For two control circuits, only "heating" or "cooling" with no additional stage can be activated.)
  - PI control (continuous or switching PWM) or 2-point control (switching) can be set as control algorithms.
  - Output of continuous (1 byte) or switching (1 bit) variables.
  - Control parameters for PI regulator (if desired: proportional range, reset time) and 2-point regulator (hysteresis) can be set.

- **Set values**
  - Each operating mode can get assigned its own temperature set values (for heating and/or cooling).
  - The set values for the additional stage are derived from the values of the basic stage by a parameterizable difference between these stages.
  - Temporary or permanent set value shift is possible by local operation on the device (parameterizable set value shift scaling).

- **Functionality**
  - Automatic or object-oriented switching over between "heating" and "cooling".
  - Regulator operation can be optionally disabled through an object.
  - Parameterizable comfort mode prolongation period.
  - Complete (1 byte) or partial (1 bit) status information can be parameterized and transmitted to the bus via an object.
  - Deactivation of the control, the additional stage or of the second control circuit via different objects is possible.

- **Room temperature measuring**
  - Internal and external room temperature sensors can be used.
  - Creating of an internal vs. external measuring value for one control circuit and an enabled external sensor can be parameterized.
  - For two control circuits, the actual temperature value of the second circuit is determined by the external sensor.
  - You can set the sampling period of the external temperature sensor.
  - The actual and the set value temperatures can be sent to the bus (even cyclically) after a parameterizable deviation.
  - The room temperature measuring (actual value) can be separately adjusted for the internal and the external sensor via certain parameters.
  - Frost/heat protection switch-over by the window status (can also be evaluated at some delay) and by automatic frost protection.
  - Temperature alarm with upper or lower temperature limit is possible. Telegrams are released through two separate objects.

- **Variable output**
  - Separate or common output of variables through one or two objects for "heating and cooling".
  - Normal or inverted output of variables can be parameterized.
  - Automatic transmitting and cycle time for the output of variables can be parameterized.

- **Room temperature timer**
  - Time- and day-dependent control of the operating modes with up to 28 different switching times.
  - By local operation, it can be activated or deactivated from programming mode.
  - Besides, the room temperature timer can be disabled via the bus.

Scene functionality:

- Eight independent internal scenes.
- Up to eight objects per scene, i. e. eight different commands can be transmitted.
- Selectable data types are switching (ON/OFF), dimming value (0...255 / 0...100%) or Shutter move commands (UP/DOWN) which can be parameterized per scene and scene object.
- The scenes can be recalled or saved by an extension object.
- Internal scenes can also be recalled without any extension object by push button operation.

Timer functionality:

- Up to two separate timers, each of them having up to 14 different switching times.
- Switching (ON/OFF), value (0…255) or scene recall (1…8) telegrams can be transmitted to the bus as control commands.
- Both timers can be disabled individually via the bus or by local operation.

Push button in general:

- The automatic switching off of the display background light can be parameterized. Alternatively, the display background light can be switched through a separate object or is always ON. The blue device ON LED will always be triggered together with the display background light.
- The display can read an alarm message (text message with a maximum length of 14 characters) received via the bus. Such alarm message must be acknowledged through any button.
- An alarm to be raised after the device has been unplugged from the flush-mounted bus coupling unit can be parameterized (1 bit or 1 byte).
- The push button assistance function can be parameterized. In this connection, when you actuate a push button, its function can be briefly read in the display as a help text.
- The current time and the date as well as the outside temperature can be read by the internal display unit. In this connection, the time and the date can be set through separate objects.

Contents

Contents II

Functional Description

1. General Functions

1.1 Enabling the push button, scene, timer and room temperature regulator functions

In the B.IQ RTR push button, the push button, scene, timer and room temperature regulator functions must be considered as separate items. These individual components can be optionally activated, if necessary. For this purpose, set the "push button function", "scene function", "timer 1/2 function" or "room temperature regulator function" parameters in the "B.IQ RTR push button" parameter branch to "enabled". Only in this case, the parameters and objects assigned to these functions will be activated and can be changed.
Please note that the parameterizable extent of the rockers of the push button will be automatically adapted by the ETS plug-in if you deactivate individual functions.
Therefore, it will, for example, not be possible for you to parameterize switching, dimming, Shutter or value transmitter functions if you have deactivated the push button function. On the other hand, if the room temperature regulator function has been deactivated, you cannot set any room temperature control for the rockers or push buttons.

1.2 Alarm function

When you unplug the B.IQ push button RTR from the flush-mounted bus coupling unit, an ON or OFF telegram or a value telegram can be transmitted via the "alarm signal" object. Alternatively, such telegram triggering can be suppressed by the "alarm function after pulling off the application module = "disabled"" (default) ETS parameter setting.

a) "Reset value" = "No" parameter (default)

When you unplug the push button from the bus coupling unit, an alarm telegram with an alarm value after removal corresponding to the parameterization (switching value "ON" or "OFF" for a switching telegram, or value "0…255" or value (0…655535) for a value telegram) will be sent.

After having been re-plugged, the push button will be ready for operation again after its initialization phase (display reading: "**Initializing**"). In this connection, the value of the alarm object will always be reset to "0" (for the 1 bit switching value and for the 2 byte value) but not actively transmitted to the bus. You need not externally reset the alarm telegram. External write access to the alarm object will overwrite the object value. However, the latter will be overwritten by the parameterized or default values when you unplug or re-plug the push button.

In case of a bus voltage failure, an alarm message transmitted before will be permanently saved. A alarm message saved will be transmitted again upon bus voltage recovery if the push button is not plugged on when the bus voltage is reappearing.

b) "Reset value" = "Yes" parameter

When you unplug the push button from the bus coupling unit, an alarm telegram with an alarm value after removal corresponding to the parameterization (switching value "ON" or "OFF" for a switching telegram, or value "1…255" or value (0…655535) for a value telegram) will be sent.

After having been re-plugged, the push button will be ready for operation again after its initialization phase (display reading: "**Initializing**"). During the initialization phase, the value of the alarm object will be reset to the inverted object value for the 1 bit switching value or to the "0" value for the 1 byte or 2 byte value and actively transmitted to the bus. You need not externally reset the alarm telegram. External write access to the alarm object will overwrite the object value. However, the latter will be overwritten by the parameterized or default values when you unplug or re-plug the push button.

In case of a bus voltage failure, an alarm message transmitted before will be permanently saved. An alarm message saved will be transmitted again upon bus voltage recovery if the push button has not been plugged on. If the push button has been plugged on when the bus voltage is reappearing the alarm will be reset by the sending of the inverted object value for the 1 bit switching value or of the "0" value for the 1 byte or 2 byte value.

Note: The "alarm signal" object can only be linked to a group address. This object can only be read out with the push button plugged on (set the "R" flag).

1.3 Light duration of status LED at operation indication

For all push button functions associated with button operations, you can parameterize the status-LED of a button as operation indication. Only in this case, the LEDs will be lit for the period set by the "Light duration of status LED at operation indication" parameter in the "B.IQ RTR push button" parameter branch if the buttons are actuated. You will have the option to parameterize 1 s, 2 s or 3 s (default).

1.4 Programming mode/local operation

The B.IQ push button RTR has a programming mode directly on the device. By means of this option, you can locally configure various functions or even preset various temperature values or switching times in addition to the parameterization through the ETS plug-in. Switching over into the programming mode or navigating through the menu are facilitated by the two display buttons on the right and left of the display.

The "operation via display buttons" parameter in the "B.IQ RTR push button" parameter branch specifies to what extent local operation from the programming menu will be possible.

- "No operation":

  'Normal operation' and local operation of the regulator by actuation of the display buttons for shifting the basic set value possible.

- "Limited operation":

  Switching over into the programming mode is possible. → 'Normal operation' including set value shifting and switching over the operating mode and resetting of the different heating and/or cooling set values is possible.

- "Complete operation":

  Full access to the device with local operation. In addition to restricted operation, it grants the user access to the up to three timers (activation/deactivation of the timers and changing of the individual switching programs) and to the "settings" menu (activation/deactivation of the button assistance function, the LCD background light and the button lock as well as of the display of the version data).

Note:

A few functions of the programming menu (e. g. set value resetting, setting the timers) will only be possible if access to these parts has been enabled in the ETS plug-in or if these functions are available at all, respectively.
Moreover, the operation of the regulator (display buttons) can be disabled. From the ETS plug-in, you can parameterize whether disabling of the regulator operation shall always take place or shall be object-controlled (refer to "4.6.2 Disabling controller operation", page 92).

You can use two display buttons on the left and right of the display and the rockers or buttons of the push button to operate the programming menu. During the time the programming menu is activated (display reading "**Prg**"), the functions of the display buttons will be indicated in the display directly opposite to the buttons. In the following, the programming mode operating functions are explained:

Functions of the display buttons:

OK (left button)	To call the menu item selected, or to change a setting. When this symbol is hidden you cannot call the menu item selected, or you cannot change the setting. In this connection, it may generally not be possible to call a function, or such function has been disabled by the ETS plug-in.
▲ (right button)	To change between the individual main menu items and between the submenus selected.
Enter (both buttons pressed at the same time)	To call the programming menu (keep the button pressed for at least 3 s) or to accept a value set, such as a set value or a switching time (keep the button pressed for at least 1 s).
- (left button)	To change the (blinking) value selected (e. g. temperature value or switching time) into the negative direction down to the adjusting limit. In 'normal operation' (no programming mode activated), you can use these buttons to shift the set value.
+ (right button)	To change the (blinking) value selected (e. g. temperature value or switching time) into the positive direction up to the adjusting limit. In 'normal operation' (no programming mode activated), you can use these buttons to shift the set value.

Functions of the push button rockers or buttons:

Escape (any push button button)	If you want to exit the programming mode or the menu selected (return to the higher-order menu) press any button of the push button (with the exception of the display buttons). If you do not wish to accept a value you have changed also execute the escape command. In such case, the value you have changed will not be stored in the device. The control will automatically execute an escape command if no further button operation takes place for about 20 s. Note: For settings you have made with the aid of the "**OK**" button, (e. g. activation/deactivation of functions) the last one will be stored in the device, even though you execute an escape command. A temperature set value shift made by means of the display buttons will also be accepted upon an escape command.

1.4.1 Local operation in the normal mode

The device will be in normal operation unless you have activated the programming mode. Irrespective of what you have parameterized (refer to "2.2 Normal operation display data", page 40), the display window will show the current room temperature (default) and, in addition, or alternatively, the outside temperature, the set value temperature or the time and the date (standard display).
If you actuate one of the display buttons the set value temperature of the activated operating mode will appear in the display if you have been enabled to use the display buttons. You can press the right or left button to shift the set value temperature in increments or decrements of 0.1 °C. You can set such set value shift (basic temperature offset) in any operating mode and, optionally, accept it when changing the mode (e. g. comfort → standby) so that this shift will act on all operating modes of the regulator. For more information on the basic set value temperature assignment or shift, please refer to chapter "4. Room Temperature Regulator Functions".

If you have set a Shift of basic set value the " ✋ " hand symbol will appear in the display. After you have set the new temperature value, you can accept the latter by executing the escape command (actuate any button of the push button with the exception of the display buttons) and return to the standard display. If you make no other entries within the next 20 s the value you have set will also be accepted as the new set value, with the display being switched back.

Note: After bus voltage recovery, the regulator will always be in normal operation.

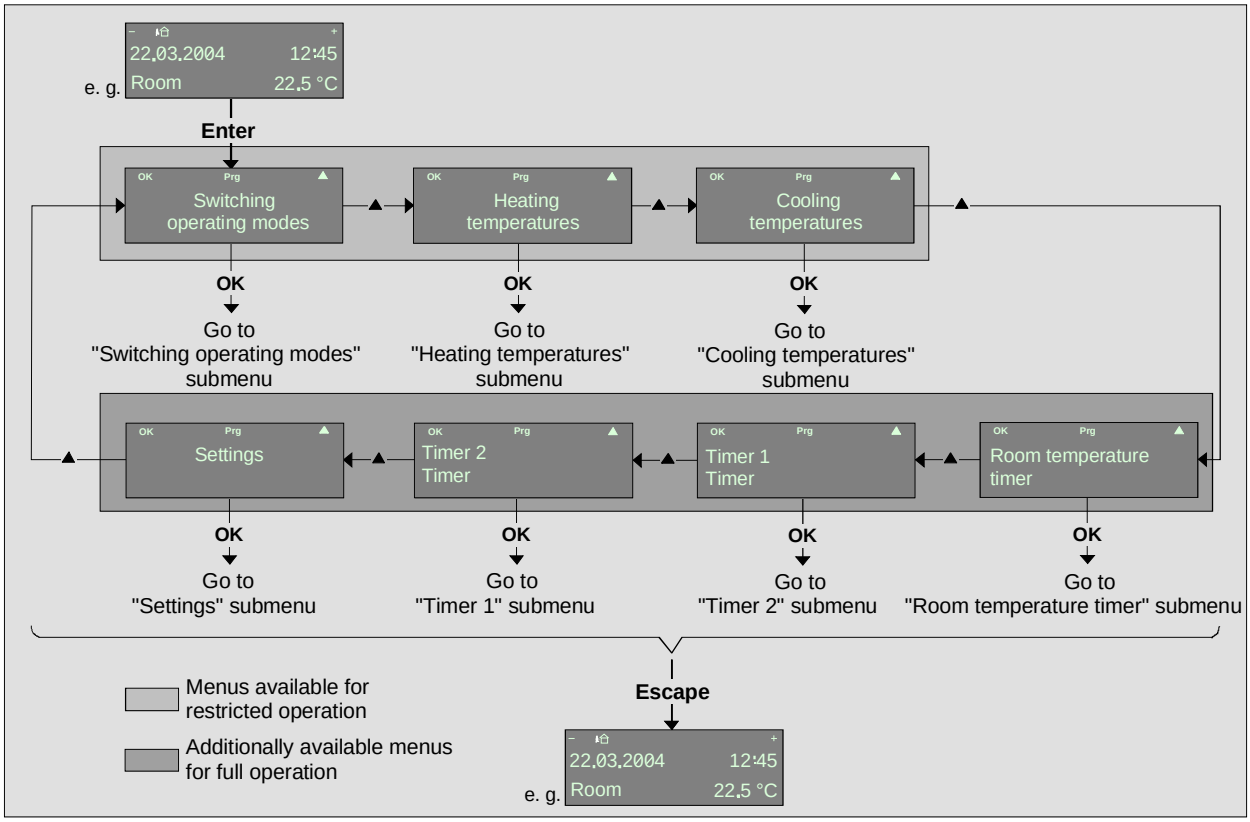
1.4.2 Local operation in the programming mode

In the programming mode, you can activate or deactivate various functions or change settings. So you can switch over the operating mode, change the temperature set values, set the timers or do any other basic settings.

Note:
In general, individual functions or settings may not be accessible, due to the parameterization of the device in the ETS plug-in.

For more information on how to set the operating modes and the room temperature timer or on their functions, respectively, please refer to chapter "4. Room Temperature Regulator Functions". The timers are described in chapter "5. Timers".

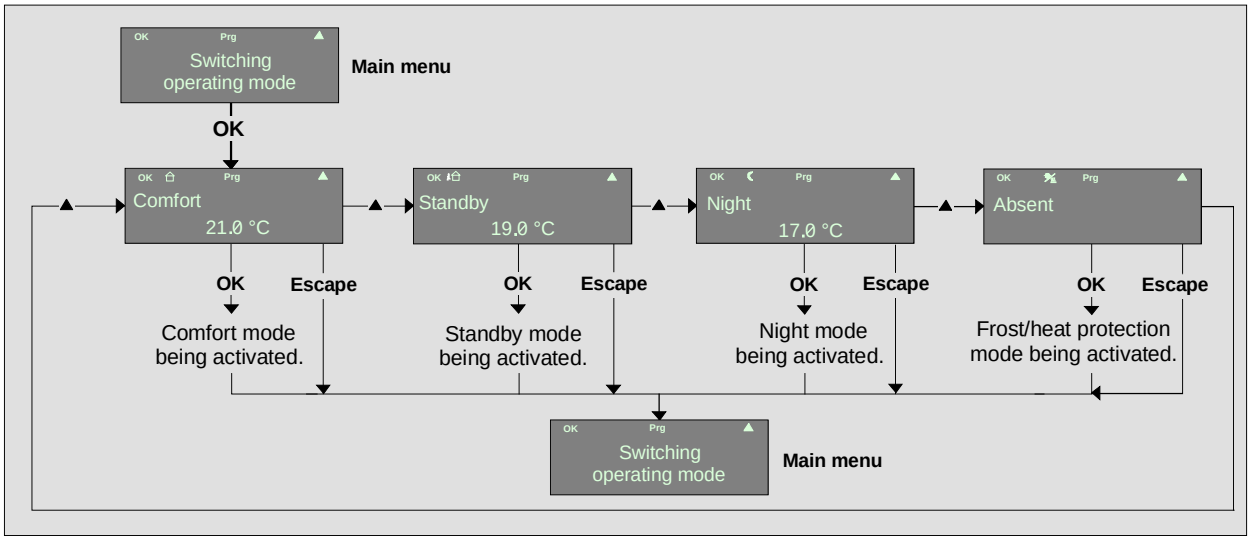
In the following, the individual menu items of the programming mode are explained:



Press both display buttons for about 3 s to call the programming mode. The display will read "**Prg**" and you will be in the main menu. You can use the " ▲ " right display button to change between the individual menu items from there. Press the "**OK**" left display button to call the corresponding submenus after you have selected the main menu.
The "heating temperatures" and "cooling temperatures" menus are visible separately or together, depending on the operating mode parameterized in the ETS plug-in. Also in the "heating and cooling" mixed mode, the cooling temperatures and thus the menu can be separately disablinged.
The submenus for the individual timers also depend on the parameterization and can be suppressed if you have not enabled these functions in the ETS plug-in. Please note that you can preset the designation of the first or second timer in the ETS plug-in. In this connection, the designation entered in the ETS plug-in will also be shown in the first line of the display (20 characters max.).
If you press any button of the push button (except the display buttons), or if you do not operate any button for about 20 s you will return to normal operation.

1.4.2.1 "Operating mode switch-over" submenu

Press the "**OK**" left display button from the main menu to call the "operating mode switch-over" submenu. This will not be possible if a higher-priority mode (e. g. window contact/presence detector) or the KONNEX override object has been activated. In such case, the "**OK**" symbol will be hidden when you select the operating mode switch-over option, with the operating mode remaining unchanged.

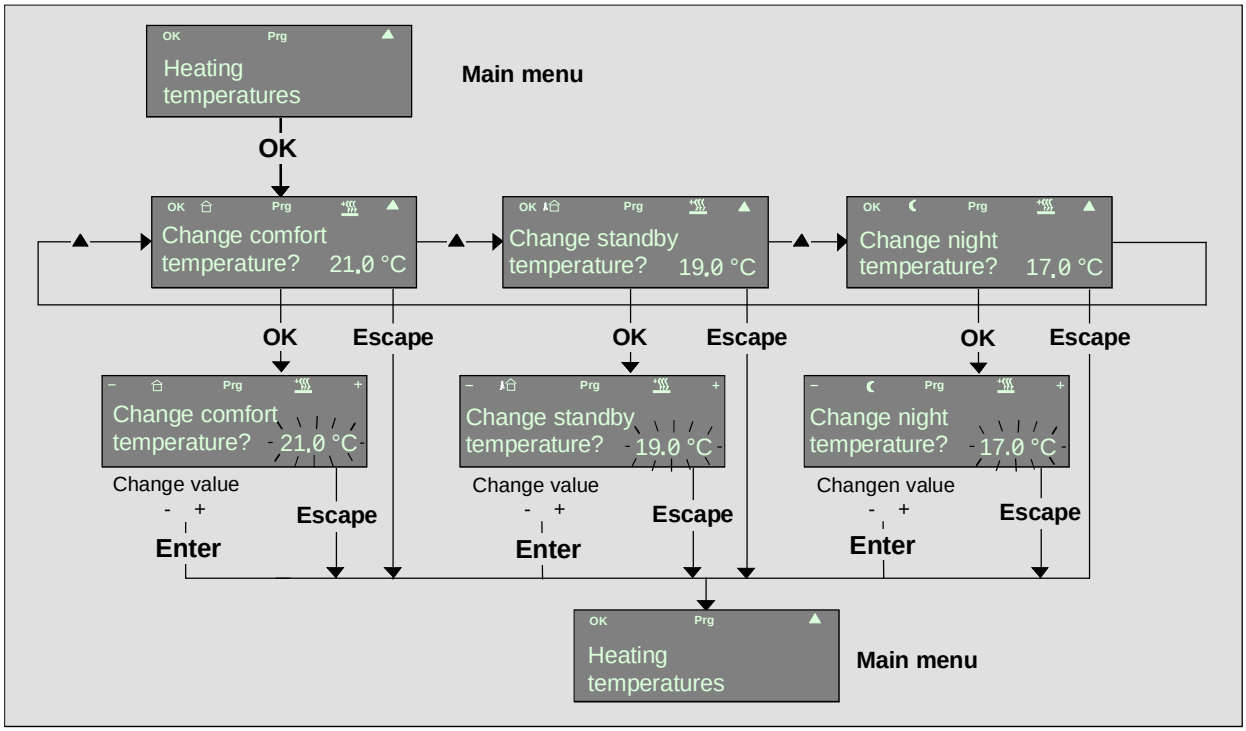


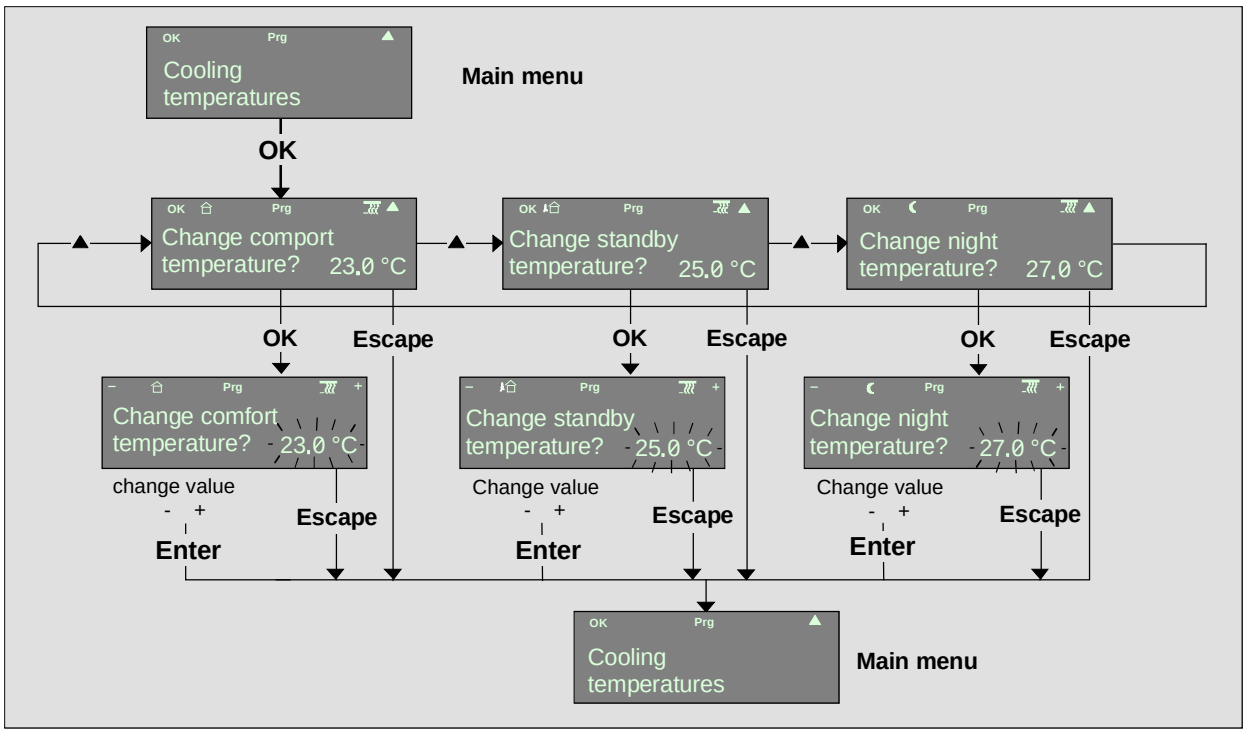
Press the "**OK**" left display button to accept the operating mode you have selected with the " ▲ " right display button. After this, the display will return to the main menu of the programming mode.
The temperature values shown in the display for the different operating modes represent the temperature set values of the selected mode to be temporarily expected. Taking account of the present operating mode of the room temperature regulator and the set value shift possibly made, those set values the regulator will take over as new ones when the operating mode is changed will be displayed here. Please note that you can change these temperature values at any later time during the 'running operation' of the regulator by local operation or by a basic set value change through the object, if enabled.

Note: You can use the "operating mode after reset" parameter in the "room temperature regulator function/ functionality" parameter branch to preselect the operating mode to be activated after bus voltage recovery. After bus voltage recovery, normal operation will always be activated.

1.4.2.2"Heating temperatures" or "cooling temperatures" submenu

Press the "**OK**" left display button from the main menu to call the "heating temperatures" or "cooling temperatures" submenu. Depending on the operating mode set in the ETS plug-in, these menus will be visible and can be available alternatively or together (mixed mode). In the mixed mode, you can disabling the cooling temperature set values during local operation by setting the *"modification of the setpoints 'cooling'"* parameter in the *"room temperature regulator function/set point values"* parameter branch to *"deactivated*.





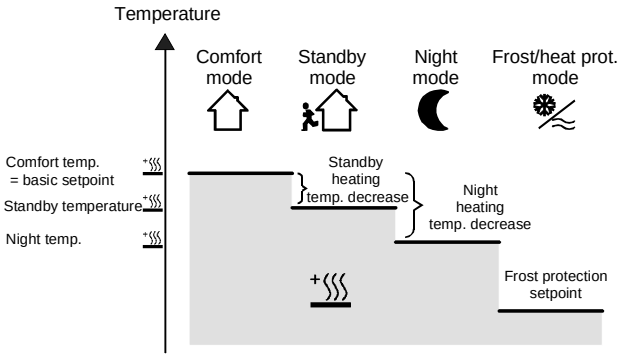
Assigning temperature set values:

You can assign set values for the
- "comfort ⌂",
- "standby ⌂" and
- "night ☾" modes.

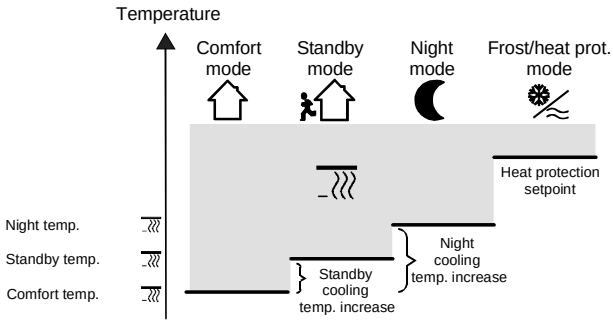
For the assignment of set values, up to six different values will be offered, depending on the operating mode enabled in the ETS plug-in.
Please note that the individual set values have not been enabled in the ETS plug-in for local operation so that you can only view them in the display window without being able to change them (refer to "4.4. Temperature set values", page 80).
Furthermore, if the second control circuit has its own set values, you can only the set the temperature values of the first control circuit in the programming mode.
The following table shows the values you can set:

Activated Operating Mode	Parameterized Operating Mode			
			Heating and Cooling	
	Heating	Cooling	For Heating	For Cooling
Comfort ⌂	e. g. 23.0 °C Comfort set value temperature = basic set value	e. g. 27.0 °C Comfort set value temperature = basic set value	e. g. 23.0 °C Comfort set value temperature = basic set value - ½ dead band for symm. dead band / = basic set value for asymm. dead band	e. g. 27.0 °C Comfort set value temperature = basic set value + ½ dead band for symm. dead band / = basic set value for asymm. dead band
Standby ⌂	e. g. 21.0 °C Standby set value temperature	e. g. 29.0 °C Standby set value temperature	e. g. 21.0 °C Standby set value temperature	e. g. 29.0 °C Standby set value temperature
Night ☾	e. g. 19.0 °C Night set value temperature	e. g. 31.0 °C Night set value temperature	e. g. 19.0 °C Night set value temperature	e. g. 31.0 °C Night set value temperature

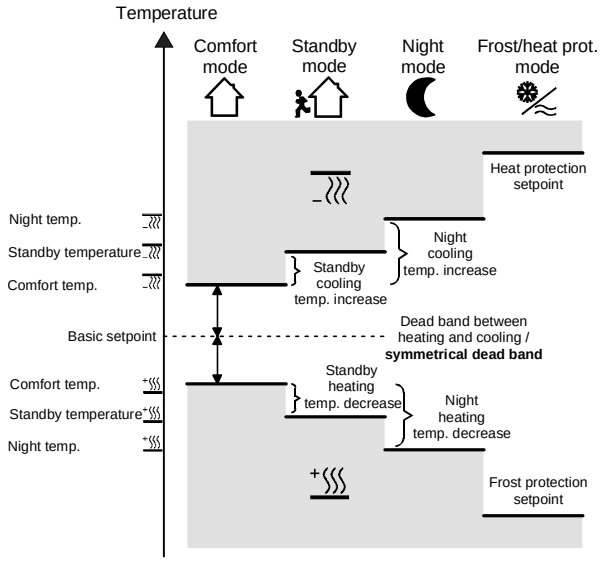
"Heating"

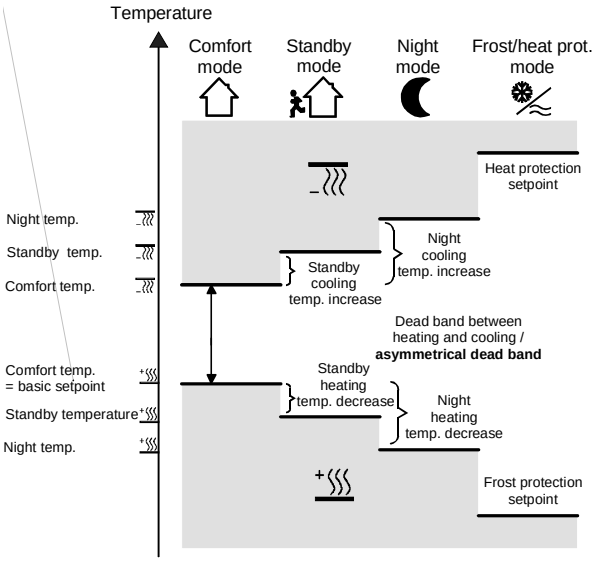


"Cooling"

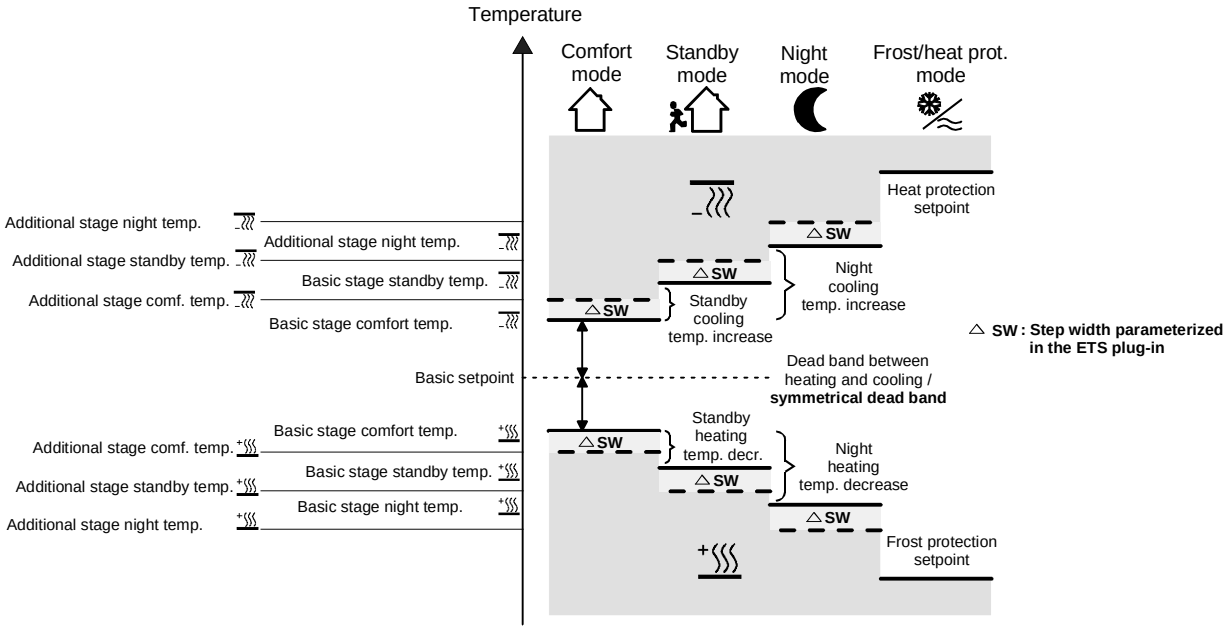


"Heating and Cooling"





Temperature Control with Additional Stage by the "Heating and Cooling" Example with Symmetrical Dead Band…



In the *"heating and cooling"* mode, you can change six different temperature set values if they have been enabled in the ETS. In dependence on the temperature decrease, temperature increase or on the dead band parameterized in the ETS, all temperature set values are derived from the basic set value temperature.
If you change the comfort set value temperature please note in this connection that all the other set value temperature values will also be changed.

The dead band (temperature zone where neither heating nor cooling takes place) is the difference between the comfort temperatures for "heating" and "cooling". Where:

$$T_{\text{comfort cooling setpt.}} - T_{\text{comfort heating setpt.}} = T_{\text{dead band}};\ T_{\text{comfort cooling setpt.}} \geq T_{\text{comfort heating setpt.}}$$

Important notes:

- For a symmetrical dead band, the basic set value is indirectly set by the comfort heating temperature. The basic set value itself will not be read in the display for local operation.

- You can change the comfort cooling set value temperature to alter the dead band. When you alter the dead zone you can expect a shifting of the comfort heating set value temperature and thus of all the other temperature set values if the dead band position is symmetrical. For an asymmetrical dead band position, solely the cooling temperature set values will be changed if you change the comfort cooling set value temperature. By local operation, you can shift the dead band to 0 °C ($T_{\text{comfort cooling set value}} = T_{\text{comfort heating set value}}$). In such case, neither heating nor cooling will take place if the room temperature determined is equal to the comfort set value temperatures.

The "standby" and "night" set value temperatures are derived from the comfort heating or comfort cooling set value temperatures. In this connection, you can assign the temperature increase (for cooling) and the temperature decrease (for heating) of both operating modes in the ETS plug-in.
Through local operation in the programming mode, you can set the "standby" and "night" set value temperatures independently of the values for the temperature increase or decrease originally parameterized in the ETS.
In such case, the "standby" or "night" set value temperatures will always be shifted together with the temperature increase or decrease resulting from local operation if you change the basic set value temperature of the dead band.
You can restore the originally parameterized values after reprogramming by the ETS.

Where:

$$T_{\text{standby heating set value}} \leq T_{\text{comfort heating set value}} \leq T_{\text{comfort cooling set value}} \leq T_{\text{standby cooling set value}}$$

or

$$T_{\text{night heating set value}} \leq T_{\text{comfort heating set value}} \leq T_{\text{comfort cooling set value}} \leq T_{\text{night cooling set value}}$$

For two-stage control, the set values for the additional stage are always derived dynamically from the set values of the basic stage. In this connection, the temperature set values of the additional stage are specified by the difference parameterized in the ETS plug-in. Changing the difference is not possible for local operation.

For changing the temperature basic set value (e. g. by changing the comfort heating set value temperature in the programming mode), distinction must always be made between two different cases:

- Option 1: The basic set value change will be accepted permanently.
- Option 2: The basic set value change will be accepted only temporarily (default).

In this connection, you can use the *"accept modification of the basic temperature setpoint value permanetly"* parameter in the *"room temperature regulator function/set point values"* parameter branch to determine whether you want to permanently (select *"yes"*) or only temporarily (select *"no"*) save the changed basic temperature value.

Option 1:

If you change the basic temperature set value it will be permanently saved in the EEPROM of the push button. In this connection, the newly set value will overwrite the basic temperature originally parameterized by the ETS.

Please take account of that:

- frequent changing of the basic temperature (e. g. several times a day) can adversely affect the life of the device as the non-volatile memory used has been designed for less frequent write access events only.
- you can alternatively assign this temperature via the bus through the "basic set value" object if it has been enabled in the ETS plug-in.

Thus, the basic set value selected on the push button or received through the object will remain stored even though the bus voltage may fail.

Option 2:

The basic set value selected on the push button or received through the object will remain stored in the currently set operating mode only in temporary form. In case of bus voltage failure or after you have changed the operating mode (e. g. from comfort to standby), the basic set value specified by local operation or received through the object will be discarded and replaced by the value originally parameterized in the ETS.

Important:

- As the set value temperatures for the "standby" and "night" modes, or the set values for the "cooling" mode, respectively, are derived from the "heating" basic set value temperature, taking into consideration the decrease, increase or dead band values parameterized in the ETS plug-in or assigned by local operation, these set value temperatures will also linearly shift around the basic set value change made.
  The temperature set values for the standby or night modes, or for the comfort "cooling" mode (dead band), respectively, will always be permanently saved in the EEPROM.

- Please note that you can only change or save any temperature set values by local operation or through the "basic set value" object, if the latter has been enabled in the ETS plug-in (refer to "4.4 Temperature set values", page 80). Any value assigned by local operation will not be taken over into the object.

If you have reset a set value and terminated such entry by an **enter** command the operating mode associated with the set value you have changed will be taken over as active mode.
However, this will only happen if no higher-priority mode (e. g. window contact / presence detector), or no KONNEX override object has been activated.

Example 1:
1 - Actuate the push button to activate the comfort mode " ⌂ ".
2 - Change into the programming mode.
3 - Change the set value for the night mode " ☾ ".
4 - Accept the new set value (**enter**) - change over to normal operation.
5 - The night mode " ☾ " will now be active.

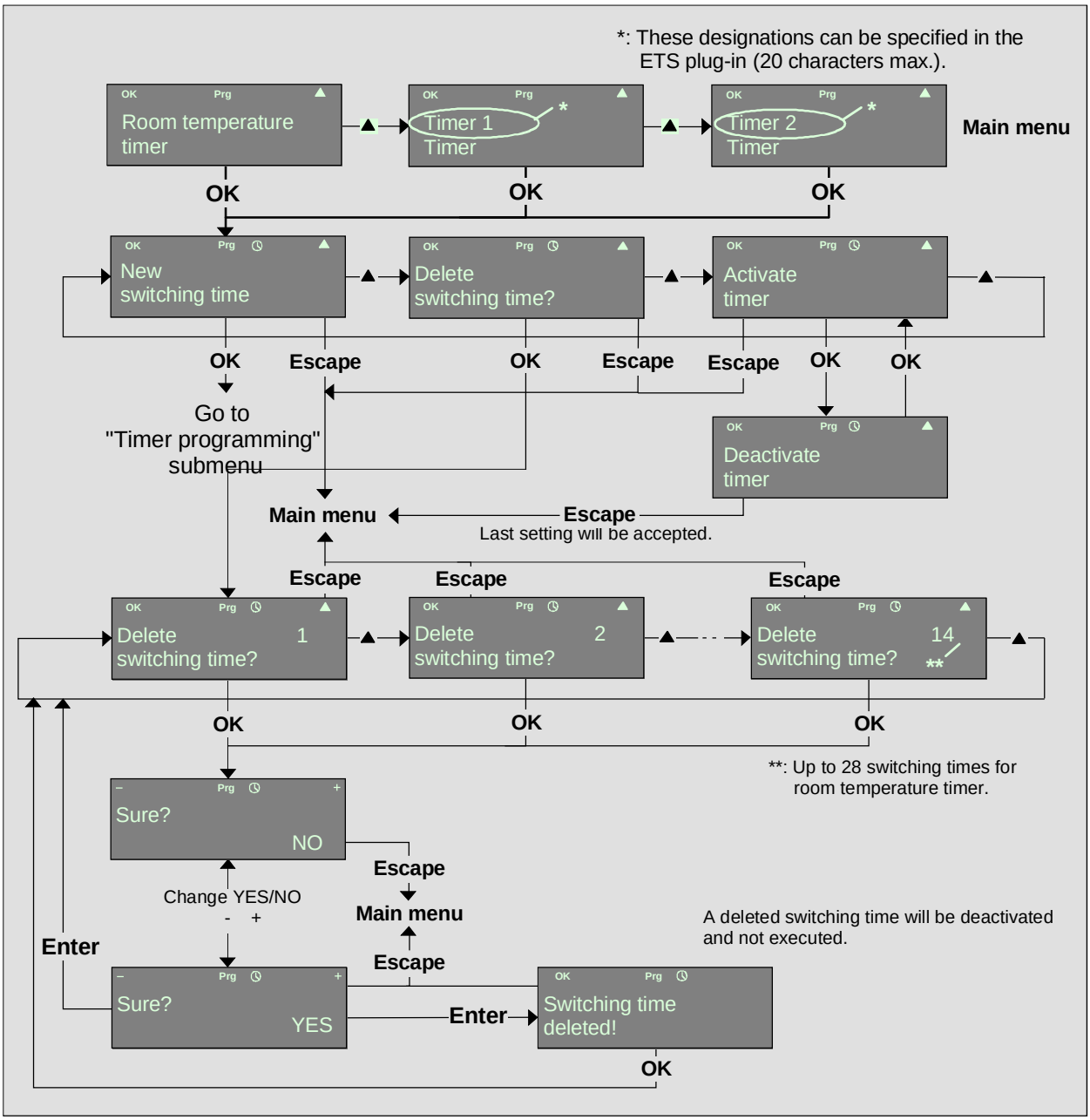
Example 2:
1 - The presence detector is active (comfort mode " ⌂ ").
2 - Change into the programming mode.
3 - Change the set value for the night mode " ☾ ".
4 - Accept the new set value (**enter**) - change over to normal operation.
5 - The comfort mode " ⌂ " will still be active.

You can only still change the operating mode if the associated set value has been enabled in the ETS plug-in for local changing (refer to "4.4 Temperature set values", page 80).
If you want to set any further set values return to the programming mode and proceed again as described above.

Note: You can use the *"operation mode after reset"* parameter in the *"room temperature regulator function/ functionality"* parameter branch to preselect the operating mode to be activated after bus voltage recovery. After bus voltage recovery, normal operation will always be activated.

1.4.2.3"Timer" submenus

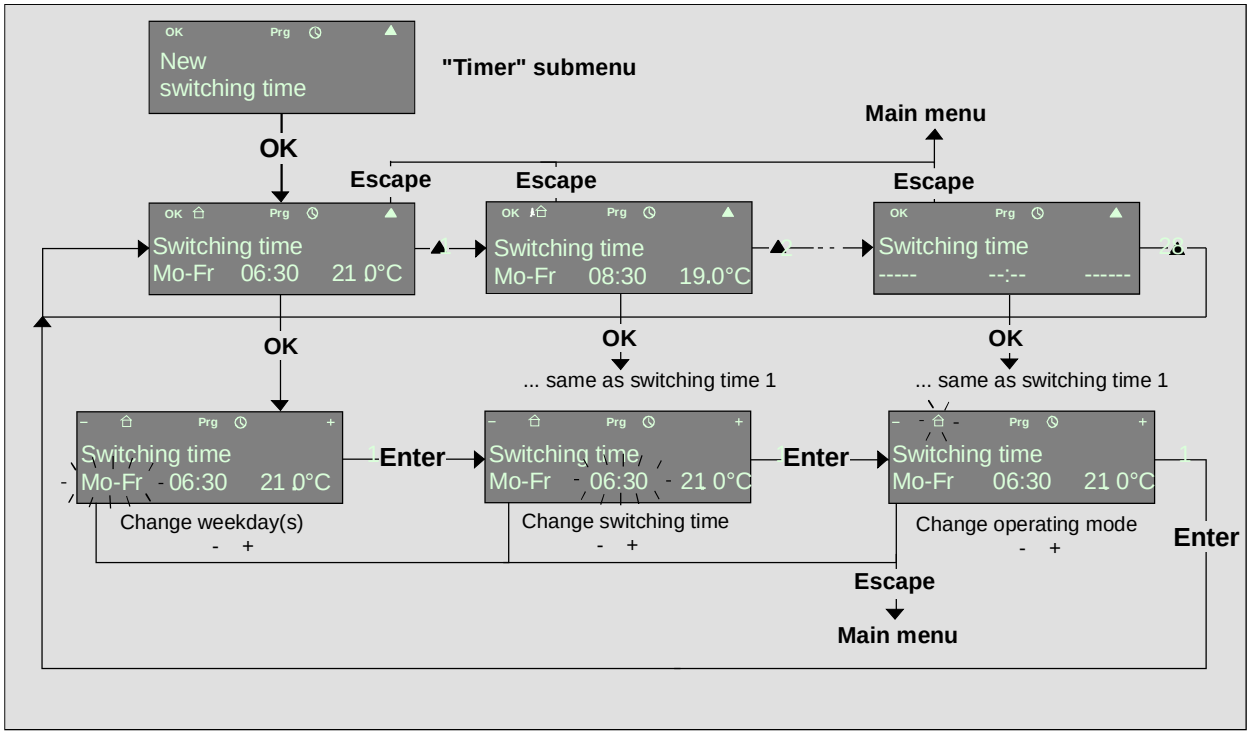
The B.IQ push button RTR knows up to three different timers. These are the "room temperature timer" with its up to 28 different switching times and "timer" 1 or 2, each of the holding up to 14 different switching times. In the programming mode, you can change, create or delete the individual switching times as well as activate or deactivate the timers.
The individual timers must always have been enabled in the ETS plug-in, and full operation must be accessible through the display buttons until local timer operation is possible in the programming mode.
Press the "**OK**" left display button from the main menu to call the "room temperature timer" or "timer" submenu.
Please note that you can give the two timers their own designations in the ETS plug-in (20 characters max.) which will also be shown in the first line of the display. Thus, easy and unambiguous assignment or identification of the timer will be possible. The selection menu shown below is identical for all timers:



The submenus for programming the switching times of the two timers differ from the room temperature timer as follows:

Room temperature timer "timer programming" submenu:

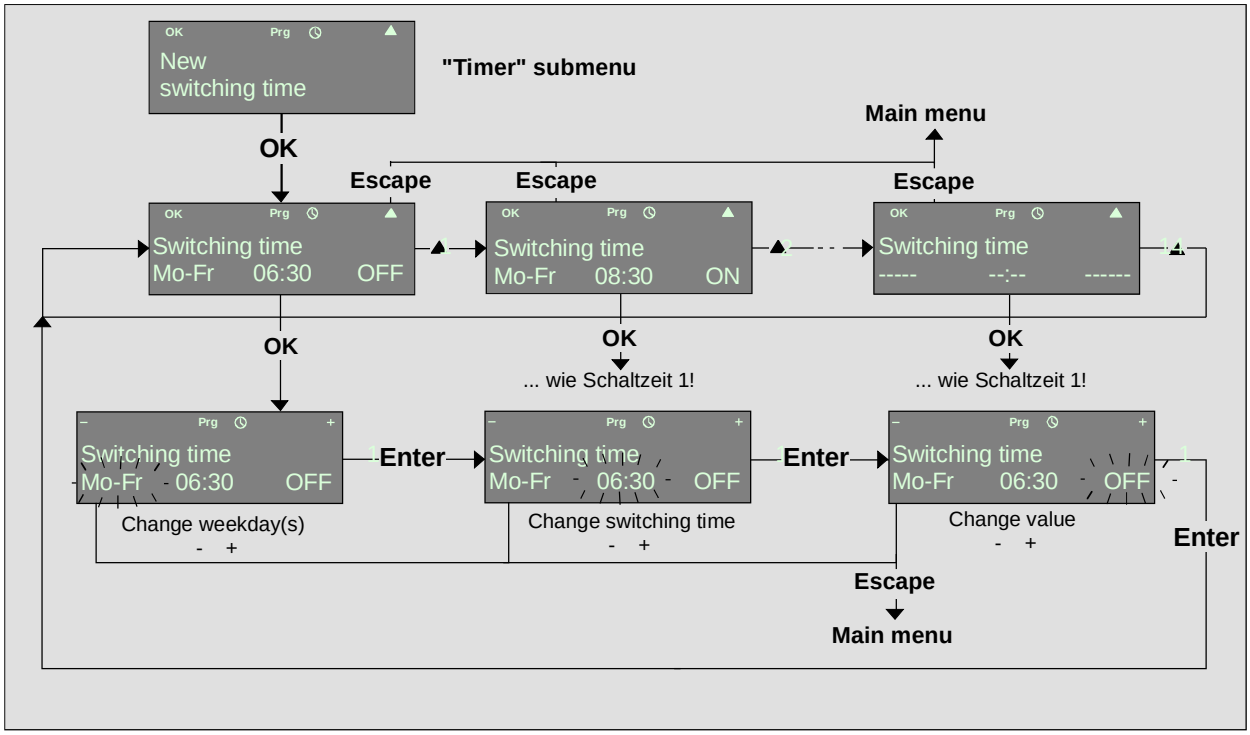
The room temperature timer can distinguish up to 28 different switching times and facilitates the switch-over of the room temperature regulator operating mode exactly to the minute, depending on the time and the day of the week. The switching times will be executed in chronological order.



The temperature values shown in the display for the different switching times represent the temperature set values of each preselected operating mode to be temporarily expected. Taking account of the present operating mode of the room temperature regulator and of the set value shift possibly made, those set values the regulator will take over as new ones as soon as the switching time is executed will be displayed here. Please note that you can change these temperature values at any later time during the 'running operation' of the regulator by local operation or by a basic set value change through the object, if enabled.

Timer 1 to 2 "timer programming" submenu:

Each of these up to two timers (seven-day timers) can distinguish between 14 different switching times and facilitates the transmission of bus commands precisely down to the minute, depending on the time and the day of the week. The switching times will be executed in chronological order.



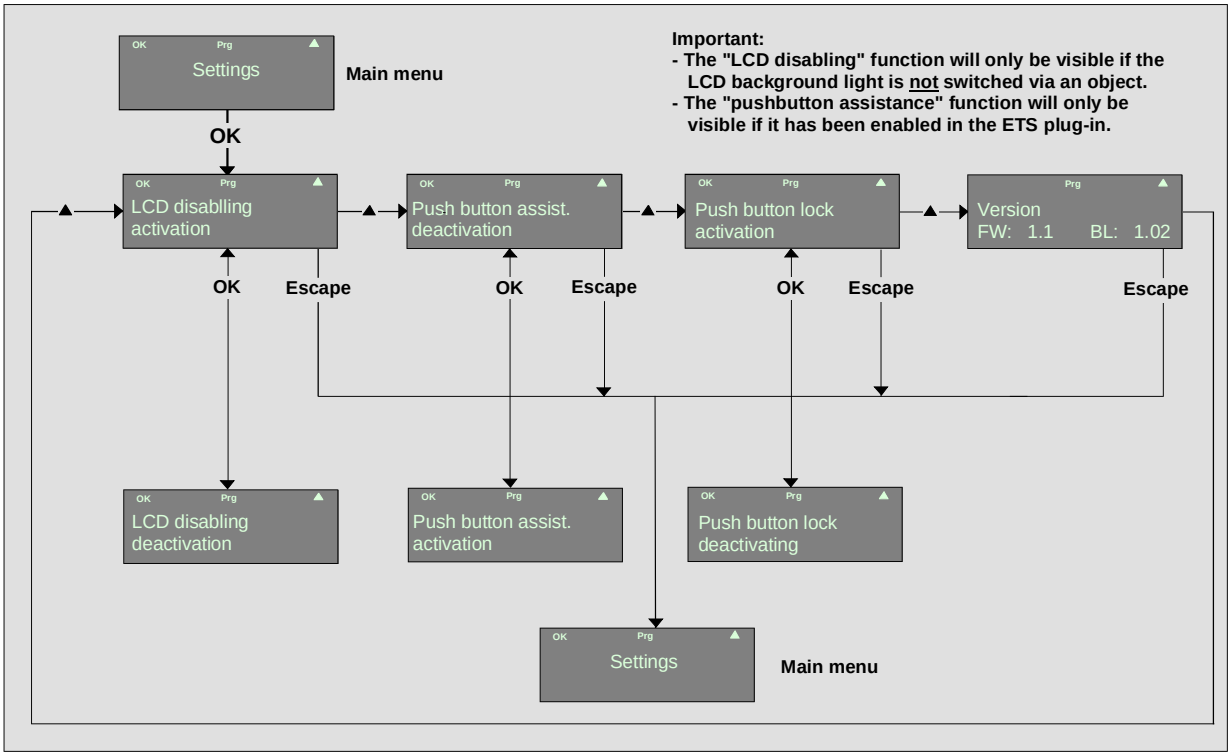
You can send switching commands (ON, OFF), value commands (0...255 / 0...100 %) or scene recall commands (1...8) to the bus and edit them from the programming menu. The data format (switching, value, scene) of the command to be transmitted is parameterized in the ETS plug-in and cannot be changed locally. Depending on the selection made in the ETS plug-in, the value range and the representation of the command are adapted in the programming menu.

Data Format	Value Type	Display Reading
Switching	---	OFF ("0") ON ("1")
Value	0...255 0...100 %	0...255 (Step width: 1) 0...100 % (Step width: 1 %)
Scene recall	---	S01 ... S08

The two timers can have different data formats. However, the up to 14 switching times of a timer have all one common data format.

1.4.2.4 "Settings" submenu

Press the "**OK**" left display button from the main menu to call the "settings" submenu. From this submenu, you can activate the automatic LCD background (LCD illumination) light switch-off or the button lock functions, depending on the parameterization in the ETS. In addition, the current software version data can be displayed. The "settings" submenu will only be visible when the "complete operation" option of the programming menu is active.



LCD background light switch-off:
In the ETS plug-in, you can specify how you want to trigger the LCD background light in general. In this connection, the "permanently ON", "automatic switch-off" and "switching via object" settings are possible. If the light is not switched through the object you can activate automatic switch-off from the programming menu.
For an LCD background light switch-off activated in the programming mode, either the switch-off time preset in the ETS plug-in (for "automatic switch-off" parameterization) or the default time of 10 s (for "permanently ON" parameterization) will be used as illumination period.

Push button assistance function:
A push button assistance function enabled and configured in the ETS plug-in can be deactivated in the programming mode at any time. The button assistance function deactivated from the menu will always be activated after initialization (after a reset). Please note that the button assistance function will be suppressed if the button lock function is active.

Push button lock:
If you have activated the push button lock function (e. g. protection of children) in the programming mode the rockers of the push button will be disabled. In such case, rocker actuation will show no response. This type of disabling is independent of a push button disabling function initiated via the bus. If the button lock function has been activated in the programming mode the " ◌ " symbol will not appear in the display.
The display buttons will still be in operation after an disabling function has been activated from the programming menu.
The disabling function activated from the menu will always be deactivated after initialization (after a reset).

Display of the software version:
The version of the firmware loaded (e. g. **FW: 1.1**) and the software version of the bootloader (e. g. **BL: 1.02**) will be displayed. Together with the ETS plug-in, the B.IQ push button RTR facilitates firmware updating so that future function extensions will be possible without replacing the device.

2. Display

2.1 Basic function

The display unit is located behind the black transparent window between the two display buttons. On this unit, various functions of the integrated room temperature regulator or also of the push button can be displayed.
You can use the *"Illumination"* parameter in the *"display"* parameter branch to preset the function of the display background light. You can permanently switch on the light (*"ON"* setting), allow it to switch off automatically (setting: *"automatic switch-off"*/default) or switch it through a separate object (*"switching via object"* setting).

If you select automatic switch-off the light will switch on if you press any button and will then automatically go out after the time you have set by the *"automatic illumination switch-off"* parameters has elapsed. In this connection, you can preset switch-on times between approx. 1 s and approx. 20 min.

Alternatively, the light can be switched via object 22, *"switching display"*. The polarity of this object is fixed. So at an object value = "1", the light will be on, whereas it will be off at "0". After bus voltage recovery, the object value will always be "0".
In addition, the *"kind of switching"* parameter defines whether the light will remain permanently on at an object value = "1" (*"ON"* setting), or whether it will be automatically disabled after a parameterized time (*"automatic switch off"* setting). In the latter case, the display background light will only be switched back on if another "1" telegram is received via object 22. A "0" telegram will always immediately switch off the light.

The blue device ON LED will always be triggered together with the display background light.

2.2 Display data in normal operation

Temperature values such as the current room inside temperature, the current outside temperature, the current set value temperature, or the time and the date can be shown in the display unit. By the *"display of"* parameter in the *"display"* parameter branch in the ETS plug-in, you can define which of such information will be displayed.
In this connection, you can also have more than one piece of information displayed (e. g. *"date/time/room temperature"* parameter setting). In such case, all three pieces of information will be displayed at the same time.

2.2.1 Temperature display

The display can read the room temperature (actual temperature of the first control circuit) determined by the regulator and, additionally or alternatively, the outside temperature (exterior) received via the bus and/or the current set value temperature of the first control circuit.
The room temperature reading has a resolution of 0.1 °C and covers a range from -99.9 °C to +99.9 °C. The reading will refresh as soon as the determined temperature changes within the resolution interval.
The outside temperature reading has a resolution of 0.1 °C and covers a range from -99.9 °C to +99.9 °C. The reading will refresh as soon as a temperature-value telegram is received via the object 25 *"external temperature sensor"*. If parameterized, the outside temperature will only be read in the display and cannot be used for any further temperature or variable calculation.
The set value temperature reading has a resolution of 0.1 °C. Its possible temperature range depends on the parameterized heating/cooling switch over and is given by the fixed values for the frost and/or heat protection temperature. The reading will refresh as soon as a new set value temperature for the regulator results (e. g. from a change of the operating mode or of the basic set value, etc.). When two control circuits with separate set values are used only the set value of the first control circuit will be displayed.

2.2.2 Date and time

If parameterized, the date and the time can be read in the display. In this connection, the time received via *"time"* object 52 and the date received via *"date"* object 53 for example, transmitted by an EIB clock, will be displayed and the system clock implemented in the device initialized.
Once initialized, this clock will keep running internally, updating the display every minute. The ":" symbol between the hours and minutes will always be blinking at intervals of one second.
The weekday information is obtained from the time signal received and is used for the control of the timers. The weekday is not displayed.

The time signal should be transmitted at least every hour to keep the time error of the internal clock at a minimum. As long as no time or date signal has been received via the objects, the display will read "--:--". The same reading will appear unless the internal clock has been updated via the bus at least once a day (updating check at 4:00 a. m.). However, the internal clock will keep running at the expected time error, with any possibly parameterized switching events of the timers or of the room temperature timer still being executed.
Moreover, the internal time will get lost in case of a bus voltage failure so that the time signal should be transmitted after bus voltage recovery.

2.2.3 Symbols

Depending on the operating state of the device, the following symbols can be shown in the display in addition to the temperature or the time, respectively:

Symbol displayed	Description
-	To change the (blinking) value (e. g. temperature value or switching time) selected in the programming mode by means of the left display button into the negative direction down to the adjusting limit. In 'normal operation' you can use the left display button to shift the set value.
+	To change the (blinking) value (e. g. temperature value or switching time) selected by means of the right display button into the positive direction up to the adjusting limit. In 'normal operation' you can use the right display button to shift the set value.
OK	To call the menu item selected in the programming mode or to change a setting by means of the left display button. When this symbol is hidden you cannot call the menu item selected, or you cannot change the setting. In this connection, it may generally not be possible to call a function, or such function has been disabled by the ETS plug-in.
	"Comfort" mode active or "comfort" mode preselected from the programming menu, or to display the comfort set value temperature.
	"Standby" mode active or "standby" mode preselected from the programming menu, or to display the standby set value temperature.
	"Night" mode active or "night" mode preselected from the programming menu, or to display the night set value temperature.
	"Frost/heat protection" mode active or "frost/heat protection" mode preselected from the programming menu.
	Room temperature regulator disabled (dew-point operation) is active.
	A Shift of basic set value is active. In this case, the set value has been shifted in normal operation by one of the display buttons.
Prg	The programming menu is active.
	A push button disabling function has been activated via the bus. This symbol will also appear even though regulator operation is disabled.
	The room temperature timer, timer 1 or timer 2 are activated.
	The room is being heated or a heating set value temperature is shown in the programming mode.
	The room is being cooled or a cooling set value temperature is shown in the programming mode.
	To change between the individual main menu items and, if necessary, between the selected submenus by means of the right display button.

2.3 Text display (alarm text)

The B.IQ push button RTR can display a text message received via the bus and having a maximum length of 14 characters, according to EIS 15. Such text can, for example, be the alarm message from an alarm central unit. As soon as a character string is received via *"text indication"* object 68, the text contained therein will be shown in the bottom line of the display. Any other display elements will go off. Such alarm message must be acknowledged through any push button of the push button sensor.
The text received will be shown upon each object update. The reception of a text message will not affect the functions of the regulator or of the push button.

2.4 Data displayed during a programming process and during initialization or in case of error

While the device is being programmed by the ETS, the display will show some status information.

When the ETS has built up a connection with the device via the bus, the display will read "**Download parameters**". In this state, the device is being programmed with the project data. During this procedure, all push button and room temperature regulator functions will be deactivated.

You can start a full program download, for example, when updating the firmware or in the event of an error (previously aborted programming process). For this purpose, select item *"with next download: transmit all"* under *"settings - options - hardware"* in the ETS plug-in. In this connection, the firmware matching the device will be automatically loaded and message "**Firmware download loading.....**" shown in the display while programming is taking place. A firmware download can take several minutes.

After one programming process has been successfully completed, or after the push button has been re-plugged onto the bus coupling unit, the display will show for a short time the push button variant and the software version loaded. Reading "**Berker B.IQ 5fach FW: 1.1 BL: 1.02**", for example, indicates a 5gang B.IQ push button RTR with firmware version 1.0 loaded and with a version 1.02 bootloader.

After this, the device will initialize. In this state, the display will read "**Initializing**". Then the device (both the push button and the room temperature regulator) will be ready for operation.

As the bus coupling unit and the B.IQ push button RTR form one unit and cannot be changed arbitrarily after having been programmed, the push button will check after bus voltage recovery or after having been re-plugged onto bus coupling unit whether the parameterizing information in the bus coupling unit agree with those in the memory of the push button.
If the parameterizations do not match as the unit was not previously placed into operation as a whole, or as the push button or the bus coupling unit has been changed, the display of the B.IQ push button RTR will read "**No parameters**" to indicate that it has no valid parameters. In such case, the push button will not respond to any button actuation.

If the display reads "**Error**" the B.IQ push button RTR will be inoperable. In this case, the push button will not show any response and must be replaced.
If the display reads "**Programming-Mode**" the push button has been plugged onto a flush-mounted bus coupling unit which cannot be used. In such case, replace the bus coupling unit by a suitable UP BCU 1 (Berker ordering no.: 750 400 03).

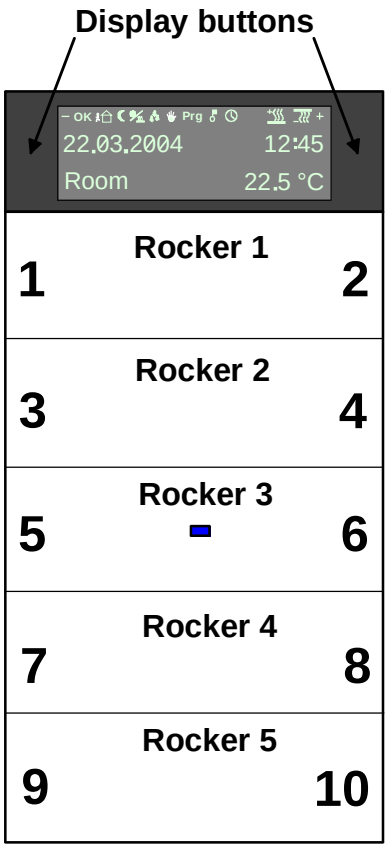
If the display shows "---" while it is reading the software version data and, in addition, if the hourglass symbol appears, there is no valid or runable firmware in the device. Such condition can, for example, be caused if a previously made firmware download was too erroneous or was interrupted for any other reasons. In such case, the B.IQ push button RTR will show no further response.
The remedy recommended for such erroneous situation is to perform a fresh firmware download (refer to "Remarks on the Software - Firmware" at the end of this documentation).

3. Push button Functions

3.1 Rocker arrangements

Depending on the variant configured, the B.IQ push button RTR has up to 5 or 10 rockers, to which certain functions have been firmly assigned, or to which parameterizable functions can be assigned in the ETS plug-in.

For example, 5gang type



Depending on the application (3gang, 4gang or 5gang type) inserted into the ETS project, the required number of rockers or buttons, respectively, will be created automatically in the ETS plug-in.

To make the configuration of the push button functions more clearly arranged a preview window is optionally available in the ETS plug-in. This window can be activated from the *"preview"* menu item in the *"configuration"* menu. If the window is enabled you can mouse-click on one of rockers or buttons to go to the corresponding parameter branch and thus parameterize the selected rocker or button, respectively.

Within the corresponding parameter branch, you can assign a name to a rocker or a button. Such name just serves for better orientation in the plug-in and will neither be displayed in the preview window nor be downloaded into the device.

3.2 Rocker functions

In the B.IQ RTR push button, the push button, room temperature regulator, scene and control functions must be considered as separate items.
For the function as push button, up to 5 rockers, to which various functions can have been assigned, are separately available, depending on the variant configured.

The two display buttons next to the display are always reserved for local operation (basic set point shift / programming mode). The exact functionality of this rocker is discussed in more detail in the operator level switch-over description (refer to "1.4 Programming mode/local operation", page 25). The display buttons have no status-LEDs.

Parameterizable in the ETS plug-in, the rockers can be given the following push button functions: In this connection, distinction is made between rocker and button actuation.

Function	Rocker Actuation	Button Actuation
No function	✓	✓
Switching/touch control	---	✓
Switching	✓	---
Dimming	✓	✓
Shutter	✓	✓
Light scene extension/recall	---	✓
value transmitter 1 byte	---	✓
value transmitter 2 byte	---	✓
Operating mode switch-over *	✓	✓
Room temperature timer operation	---	✓
Timer operation	---	✓

*: The "operating mode switch-over" function is a room temperature regulator feature. The exact functionality of a rocker parameterized to this function is discussed in more detail in the description of the room temperature regulator functions (refer to "4.1.1. Changing the operating modes", page 57).

In general, the push button function can be enabled. For this purpose, set the *"push button function"* parameter in the *"B.IQ RTR push button"* parameter branch in the ETS plug-in to *"enabled"*. If the push button function is *"disabled"* the selection of the push button functions will be matched so that only the functions of the enabled function elements (*"room temperature regulator operation", "room temperature timer operation", "timer operation" and/or "light scene extension recall"*) will be parameterizable in this case.

Through the *"concept of operation"* parameter in the *"push button functions / general"* parameter branch, select the type of rocker actuation. Separately for each rocker, you can parameterize rocker actuation or button actuation. For rocker actuation, the left and right buttons of a rocker form a pair of buttons to which you can assign a joint function. For button actuation, the left and right buttons of a rocker must be considered to be separate of each other so that two functions can be executed.
The same applies to the status-LEDs which form pairs or can be triggered separately, depending on the parameterization. In both cases, you can always parameterize how to trigger the status-LEDs.

You can separately parameterize the push button functions listed in the table to the different rockers or buttons, respectively. This will dynamically change the parameter branch in the ETS plug-in and, consequently, the object table.

You can use the *"function of the rocker"* or *"function of the push button"* parameter in the *"push button function / general / [button description]"* parameter branch to set the function to be executed upon the pressing of a button.

3.2.1 Rocker Actuation

3.2.1.1 "No function"

If the *"function of the rocker"* parameter has been parameterized to *"no function"* the buttons concerned and thus the associated object will be deactivated. Only the status-LEDs can be triggered through the status object. This triggering can be set by the *"show status object via"* in the *"push button function / general / [button description] / rocker X status"* parameter branch.

3.2.1.2 "Switching" function

If the function of the rocker has been parameterized to *"switching"* the *"command on pressing a rocker"* parameter and the *"switching"* object will appear. The *"command on pressing a rocker"* parameter defines the switching commands to be sent to the bus when the left and the right button are being pressed. Executable switching commands can be *"OFF"*, *"ON"* or *"TOGGLE"*. In case of *"TOGGLE"*, the value stored in the switching object will be changed over and sent. The commands have only been preset in combined form for the left and the right button and should be selected correspondingly. In addition, you can suppress the sending of a switching command when a button is being actuated (*"---"* setting).

The status-LEDs of the rocker (left and right) can be triggered through the status object. This triggering can be set by the *"show status object via"* in the *"push button function/general/[button description]/rocker X status"* parameter branch.

3.2.1.3 "Dimming" function

If the function of the rocker has been parameterized to *"dimming"* various parameters for the dimming function and the *"switching"* and *"dimming"* objects will appear. The *"command on pressing a rocker"* parameter defines the switching or dimming commands to be sent to the bus when the left and the right button are being pressed. Executable switching commands can be *"darker (OFF)"*, *"brighter (ON)"* or *"TOGGLE"*.
"Darker (OFF)" will release an OFF telegram upon step button actuation, whereas move actuation will cause a dimming telegram (darker). *"Brighter (ON)"* will release an ON telegram upon step button actuation, whereas move actuation will cause a dimming telegram (brighter). In case of *"TOGGLE"*, the switching status internally stored in the switching object will be changed over when the button is being pressed shortly. If the stored status is ON (OFF) an OFF (ON) telegram will be raised. Pressing the button for a long time a "darker" telegram will be sent after a "brighter" telegram, and vice versa.
The commands have only been preset in combined form for the left and the right button and should be selected correspondingly.

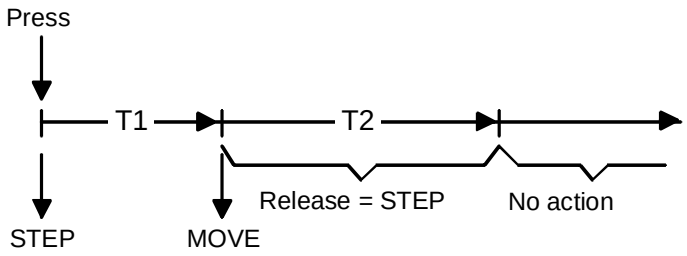
In addition, you can set *"dimming darker / brighter by"* and the *"time between switching and dimming"* options. A *"stop telegram"* at the end of a dimming process (telegram caused when the button is being released) can also be enabled. If you have set *"telegram repetition" = "yes"* dimming telegrams can be sent cyclically while a button is being pressed. In this connection, you can set the *"time between two dimming telegrams"* option. Each time when this period has elapsed, a new dimming telegram with the parameterized step width will be released.

The status-LEDs of the rocker (left and right) can be triggered through the status object. This triggering can be set by the *"show status object via"* in the *"push button function/general/[button description]/rocker X status"* parameter branch.

3.2.1.4 "Shutter" function

If the function of the rocker has been parameterized to *"Shutter"* various parameters for the Shutter function and the *"step operation"* and *"move operation"* objects will appear. The *"operation concept"* parameter will set the step and move telegram sequence to be sent upon button actuation or during a button actuation, respectively.

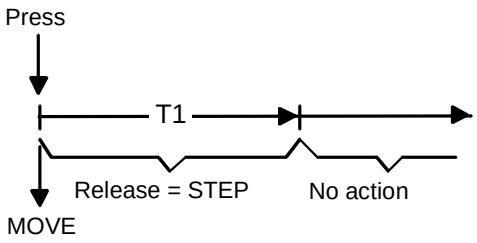
- *"Step - move - step"*:



When a button of the rocker is being pressed a STEP telegram will be sent and start time T1 (*"time between step and move operation"*) started. If you release the button within time T1 no further telegram will be sent. This STEP serves for stopping an ongoing continuous run.

If the push button remains pressed for longer than time T1 a MOVE telegram will be automatically sent after T1 has elapsed, and time T2 (*"lamella adjustment time"*) will be started. If you then release the button within time T2 the push button will send a STEP telegram. This function is used for lamella adjustment. T2 should correspond to a 180° lamella rotation.

- *"Move - step"*:



When a button of the rocker is being pressed a MOVE telegram will be sent, and start time T1 ("lamella adjustment time") will be started. If you then release the button within time T1 the push button will send a STEP telegram. This function is used for lamella adjustment. T1 should correspond to a 180° lamella rotation.

Which polarity the telegrams for move or step operation will have, i. e. which moving direction will be triggered in dependence on the actuated (left or right) button can be set by the *"command on pressing a rocker"* parameter. Executable switching commands can be *"UP"*, *"DOWN"* or *"TOGGLE"*. The commands are only been preset in combined form for the left and the right button and should be selected correspondingly.

The status-LEDs of the rocker (left and right) can be triggered through the status object. This triggering can be set by the *"show status object via"* in the *"push button function/general/[button description]/rocker X status"* parameter branch.

3.2.2 Push button actuation

3.2.2.1 "No function"

If the *"function of the push button"* parameter has been set to *"no function"* the button will be deactivated. Only the status-LEDs can be triggered through the enabled status object. This triggering can be set by the *"function of the status-LED"* in the *"push button function/general/[button description]* parameter branch.

3.2.2.2 "Switching/touch control" function

If the function of the push button has been parameterized to *"switching/pushing"* the *"command on pressing the push button"* and *"command on releasing the push button"* parameters and the *"switching"* object will appear.
The *"command on pressing/releasing the push button"* parameter defines the switching commands to be sent to the bus when the button is being pressed or released, respectively. These two separate parameters even facilitate a single-button function (e. g. pressing = ON, releasing = OFF).
Executable switching commands can be *"OFF"*, *"ON"* or *"TOGGLE"*. In case of *"TOGGLE"*, the value stored in the switching object will be changed over and sent. In addition, the sending of a switching command upon the actuation of the button can be suppressed (*"no function"* setting).

You can use the *"function of the status-LED"* in the *"push button function/general/[button description]* parameter branch to set the function of the status-LED.

3.2.2.3 "Dimming" function

If the function of the button has been parameterized to *"dimming"* (single-button dimming) various parameters for the dimming function and the *"switching"* and *"dimming"* objects will appear. The *"command on pressing the push button"* parameter defines the switching or dimming commands to be sent to the bus when the button is being pressed. Executable switching commands can be *"darker (OFF)"*, *"brighter (ON)"* or *"brighter/darker (TOGGLE)"*.
"Darker (OFF)" will release an OFF telegram upon step button actuation, whereas move actuation will cause a dimming telegram (darker). *"Brighter (ON)"* will release an ON telegram upon step button actuation, whereas move actuation will cause a dimming telegram (brighter). In case of *"brighter/darker (TOGGLE)"*, the switching state internally stored in the switching object will be switched over when the button is being pressed shortly. If the stored status is ON (OFF) an OFF (ON) telegram will be raised. By pressing the button for a long time a "darker" telegram will be sent after a "brighter" telegram, and vice versa.

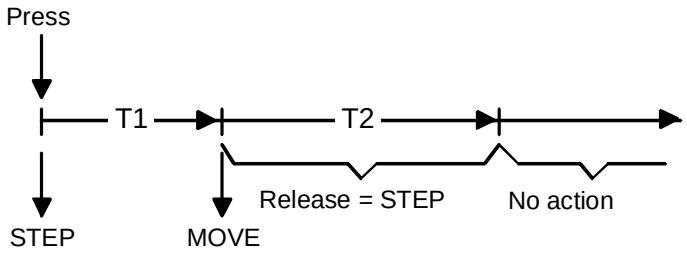
In addition, you can set the *"dimming darker / brighter by"* and the *"time between switching and dimming"* options. A *"stop telegram"* at the end of a dimming process (telegram caused when the button is being released) can also be enabled. If you have set *"telegram repetition"* = *"yes"* dimming telegrams can be sent cyclically while a button is being pressed. In this connection, you can set the *"time between two dimming telegrams"* option. Each time when this period has elapsed, a new dimming telegram with the parameterized step width will be released.

You can use the *"function of the status-LED"* in the *"push button function/general/[button description]* parameter branch to set the function of the status-LED.

3.2.2.4 "Shutter" function

If the function of the button has been parameterized to *"Shutter"* various parameters for the Shutter function and the *"step operation"* and *"move operation"* objects will appear. The *"operation concept (sequence of telegrams)"* parameter will set the step and move telegram sequence to be sent upon button actuation or during a button actuation, respectively.

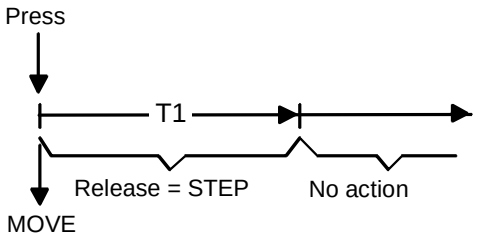
- *"Step - move - step"*:



When the button is being pressed a STEP telegram will be sent and start time T1 (*"time between step and move mode"*) started. If you release the button within time T1 no further telegram will be sent. This STEP serves for stopping an ongoing continuous run.

If the push button remains pressed for longer than time T1 a MOVE telegram will be automatically sent after T1 has elapsed, and time T2 (*"lamella adjustment time"*) will be started. If you then release the button within time T2 the push button will send a STEP telegram. This function is used for lamella adjustment. T2 should correspond to a 180° lamella rotation.

- *"Move - step"*:



When the button is being pressed a MOVE telegram will be sent and start time T1 (*"lamella adjustment time"*) started. If you then release the button within time T1 the push button will send a STEP telegram. This function is used for lamella adjustment. T1 should correspond to a 180° lamella rotation.

Which polarity the telegrams for move or step operation will have, i. e. which moving direction will be triggered upon the actuation of the button is can be set by the *"Shutter button function"* parameter. Executable switching commands can be *"UP"*, *"DOWN"* or *"TOGGLE"*.

You can use the *"function of the status-LED"* in the *"push button function/general/[button description]* parameter branch to set the function of the status-LED.

3.2.2.5 "value transmitter 1 byte" and "value transmitter 2 byte" functions

When a value transmitter 1 byte has been parameterized (e. g. for dimming value transmitter applications) the push button will transmit an 8-bit value to the bus (according to EIS 6) when a button is being actuated. The value to be transmitted can be parameterized in the ETS plug-in and can be within the range from 0 to 255.

If a value transmitter 2 byte has been parameterized 2 byte values can be transmitted to the bus. In this connection, the *"function as"* parameter presets whether this value is a temperature value (according to EIS 5), a brightness value (according to EIS 5) or a dimensionless 2 byte counter value (according to EIS 10).

The range of the parameterizable temperature value is between 0 °C and 40 °C in steps of 1 °C. The brightness value can be within 0 lux and 1500 lux in 50-lux steps. If you parameterize brightness values which do not correspond to these 50-lux steps the plug-in will automatically correct the entered value by rounding up or down. The possible range for the value transmitter 2 byte is between 0 and 65535.

Value changing:

For value transmitter parameterization, you can change the value to be sent by pressing the button for a long time (> 5 s). In this connection, the preset value will be decreased by the parameterized step width and sent. After you have released the button, the value transmitted last will be kept stored. The next move button actuation will change the value setting direction.

You can parameterize the value changing step width for the value transmitter 1 byte or 2 byte. The step width of the temperature value transmitter has been fixed to 1 °C and that of the brightness value transmitter to 50 lux.

The status-LED of the button actuated and that of the opposite button will blink (approx. 3 Hz) when you are changing a value (see next page). Do not press any other button while value changing is active.

You can use the *"function of the status-LED"* in the *"push button function/general/[button description]* parameter branch to set the function of the status-LED.

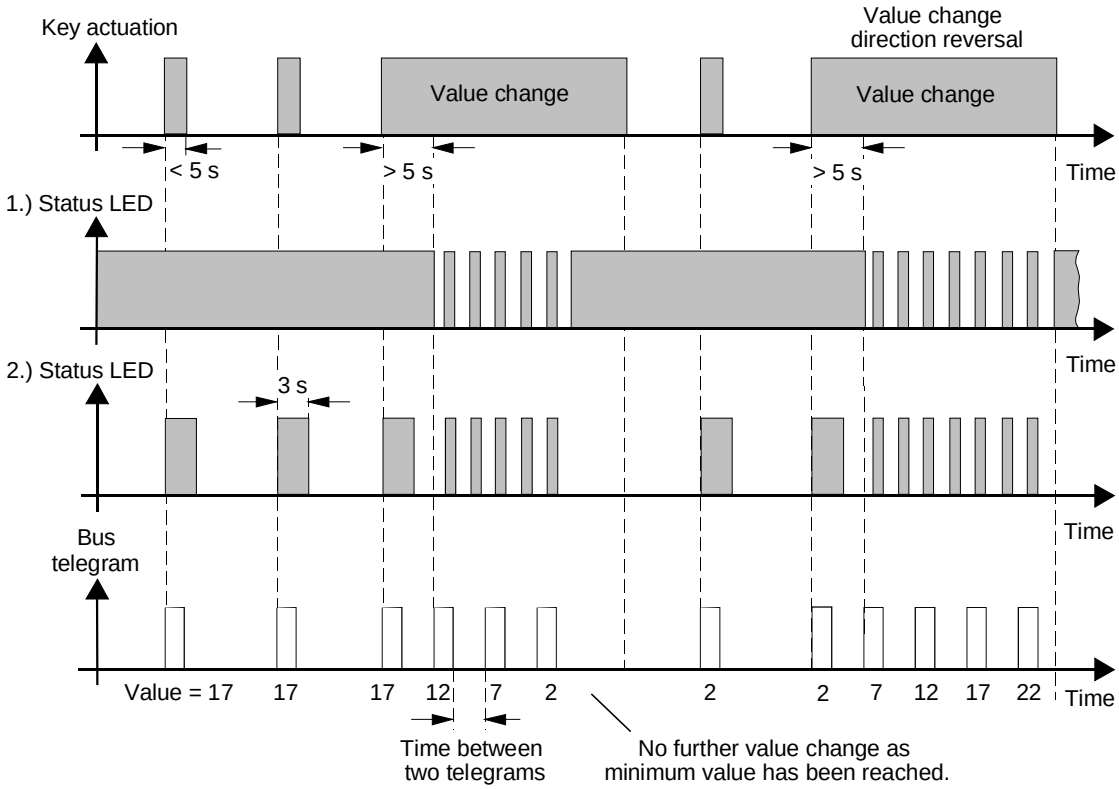
The following example shows how the value of the value transmitter 1 byte can be changed:

1.) *Function of the status-LED* *Always ON*
Value (0...255) *17*
Step width (1…10) *5*

⇨ The status-LED will always be lit. When you are changing the value it will be blinking for the time of such value changing.

2.) *Function of the status-LED* *Operation indication*
Light duration of status LED at operation indication *3 s*
Value (0...255) *17*
Step width (1…10) *5*

⇨ When a button is being actuated the status-LED will be lit for the parameterized time. When you are changing the value it will be blinking for the time of such value changing.



Important:

- The value "0" may possibly never be attained by such changing in dependence on the parameterized step width (refer to the example above). This will ensure that the value originally parameterized by the ETS can be restored when another change is made.

- The values newly set by a change will only be saved in the RAM. After a bus voltage failure or after the application module has been detached and re-plugged, such values will thus be replaced by the preset values originally programmed through the ETS.

3.2.2.6 "Light scene extension/recall" function

For this function, you can distinguish between recalling an 'external' light scene through the light scene extension object or one of the internal scenes of the B.IQ RTR push button. In this connection, the *"function as"* parameter will preset the mode of action.

If you have parameterized the function of the button to *"light scene extension"* the *"light scene extension"* object will be enabled. Upon short push button actuation (< 1 s), you can use this object to recall via the bus light scenes stored together with light scene functions in another bus device. In this connection, the light scene number (1 to 64) parameterized in the ETS plug-in will be transmitted.
If you have set the function to *"internal scene request"* you can recall the scenes stored in the B.IQ push button RTR by a step button operation (< 1 s). For this purpose, you must specify the corresponding scene number (1 to 8) in the ETS plug-in. An extension object will not be necessary for this function. Moreover, you can only recall an internal scene after you have enabled the scene function.
When working as operation indication, the status-LED will be lit for the parameterized time.

You can use the *"memory function"* parameter to specify whether you want to exclusively recall the 'external' light scenes or the 'internal' ones, or if you want to save them, if required, upon a move button operation (> 5 s).

When parameterizing a *"light scene extension"* with the storage function, you can generate a memory telegram, depending on the parameterized light scene number. In this connection, move button operation of > 5 s will cause the sending of such memory telegram.

When parameterizing an *"internal scene request"* with the storage function, you can actuate the button for a long time of > 5 s to save an internal scene in accordance with the parameterized scene number. In this connection, the scene control of the B.IQ push button RTR will request the current values of the scene objects from the actuators via the bus and save them permanently.
Make absolutely sure that the read flags ("R" flags) have been set for the corresponding actuator objects.

The status-LED of the button actuated and that of the opposite button will blink (approx. 3 Hz) during an active saving process. Do not actuate any other button in this state.
Short push button actuation of < 1 s will only recall the parameterized light scene. However, if you actuate the button for longer than 1 s, but shorter than 5 s, you will cause neither a recall nor a saving process.
When working as operation indication, the status-LED will be lit for the parameterized time.

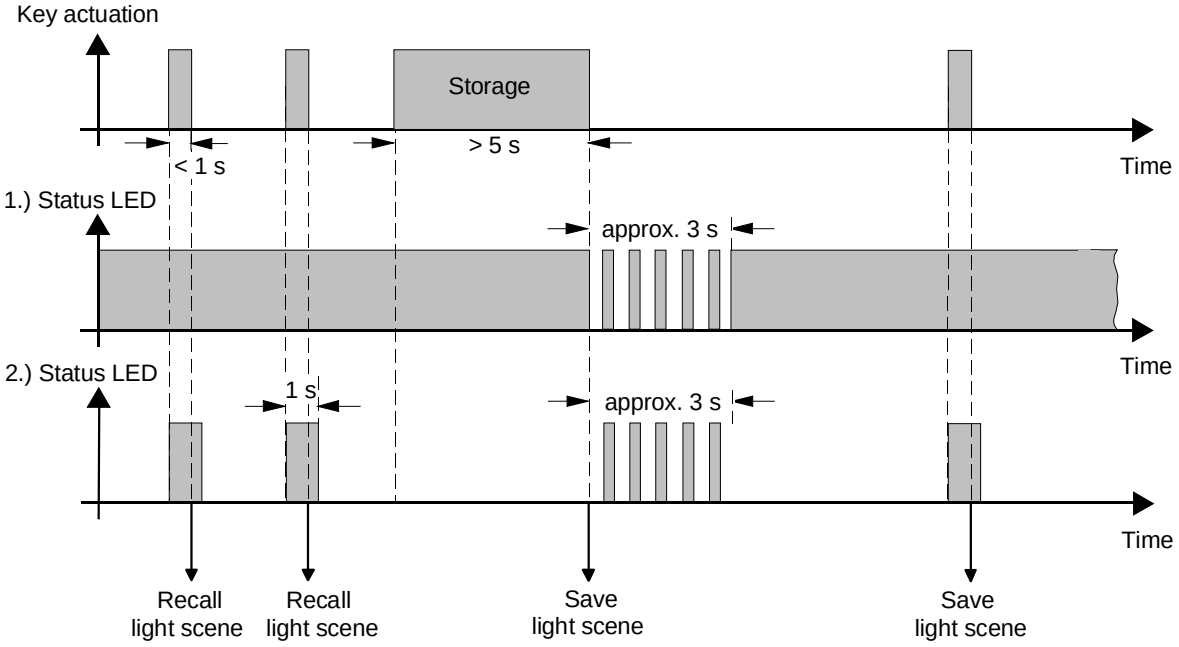
Storage function examples:

1.) *Function of the status-LED* *Always ON*
⇨ The status-LED will always be lit. It will begin to blink for some 3 s if storing takes place.

2.) *Function of the status-LED* *Operation indication*
Light duration of status LED at operation indication *1 s*

⇨ When the button is being actuated the status-LED will be lit for the parameterized time. It will begin to blink for some 3 s if storing takes place.

You can use the *"function of the status-LED"* in the *"push button function/general/[button description]* parameter branch to set the function of the status-LED.



3.2.2.7 "Room temperature timer operation" or "timer operation" function

Once the room temperature timer and/or one of the timers has/have been enabled in the ETS plug-in you can additionally set the following two operating functions for the button functions.

For room temperature timer operation, you can activate or deactivate the room temperature timer in dependence on the parameterized response to a button actuation. Toggling (between the activated and the deactivated state) will also be possible.
For a timer operation, you must specify in the ETS plug-in which of the two timers you want to operate. For this purpose, the *"function"* parameter will define the type of action. Depending on the enabled timer(s), the choice for this parameter is automatically restricted. You can activate or deactivate the timer, depending on what response to a button actuation you have parameterized. Toggling (between the activated and the deactivated state) will also be possible.

You can parameterize the function of the status-LED. In addition to the *"OFF"*, *"ON"* and *"operation indication"* standard settings, you can select the *"display timer active"* and *"display timer inactive"* options. Thus, the status-LEDs can indicate whether a function linked with the associated button is activated or not activated, respectively. Such indication will also be made if the corresponding function has been activated or deactivated in the programming mode.
It should be noted that the operating function of the room temperature timer and of the timer(s) in the programming mode will always be possible, independently of any function parameterized for the buttons.

3.3 Push button disabling function

The push button has an disabling function, by means of which you can disable individual or all rockers, respectively. In addition, you can parameterize that all rockers should behave like an explicitly preset rocker.
You can use the *"disabling function"* parameter in the *"push button functions / disabling"* parameter branch to enable the disabling function. The following settings can be explained as follows:

Settings:

"Single rocker disabled" → - Rockers 1-3 (3gang type), 1-4 (4gang type) or 1-5 (5gang type) can be separately disabled.
- In this connection the display buttons (room temperature regulator operation/programming menu) will always be in operation.

"Push button disabled" → - The entire push button including the display buttons will be disabled. In this case, you can only operate the room temperature regulator via the bus, if enabled for this purpose.

"Push button not disabled" → - No disabling function is enabled (default). The push button will work in its normal mode.

"Function of all rockers like…" → - All rockers of the device will behave in the same was as the one parameterized here. In this case, the button or rocker or button functions assigned to the parameterized rocker will always be executed when you actuate any other rocker. The status-LEDs of the rockers will be triggered in the same way as in 'normal operation'.
- The display buttons are not affected by this disabling function and will show their 'normal' behaviour.

When the disabling function is active the " ". symbol will appear in the display.
In addition, you can locally disable the push buttons of the device from the programming menu, for example as children protection (refer to the "settings" menu). This type of disabling is independent of an disabling function initiated via the bus. If the push button disabling function has been activated in the programming mode the " " symbol will not appear in the display. In addition, the push button assistance function will be suppressed when the button disabling function is active.

Please note that the room temperature regulator operation can be additionally influenced by the regulator disabling function (refer to "4.6 Room temperature regulator disabling functions", page 92). Thus, buttons or rockers which have been assigned to room temperature regulator operation must be disabled by the push button or regulator disabling function. When regulator operation has been disabled the " " symbol will also appear in the display.

You can parameterize the polarity of the disabling object.
If the polarity of the disabling object has been preset to "inverted (disable = 0)" the push button will not be immediately disabled upon bus voltage recovery or after a download. In this case, the disabling function will only be activated when the disabling object is updated (value = 0).

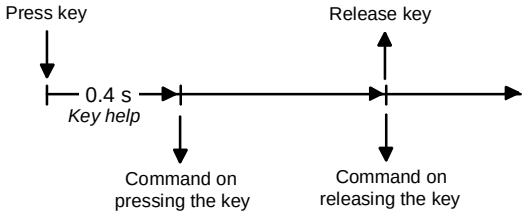
3.4 Push button assistance function

If desired, you can activate a assistance function in the ETS plug-in for the function buttons of the push button sensor (rockers 1 to 5 max.). When this function is enabled the assigned functionality will be shown as a clear-text message in the lower display line for about 3 s upon a short button actuation (< 0.4 s). In the ETS plug-in, you can separately specify the help text for each button which must not exceed 20 characters.
The function of the display buttons is fixed so that any help for them will not be necessary and is thus not implemented.

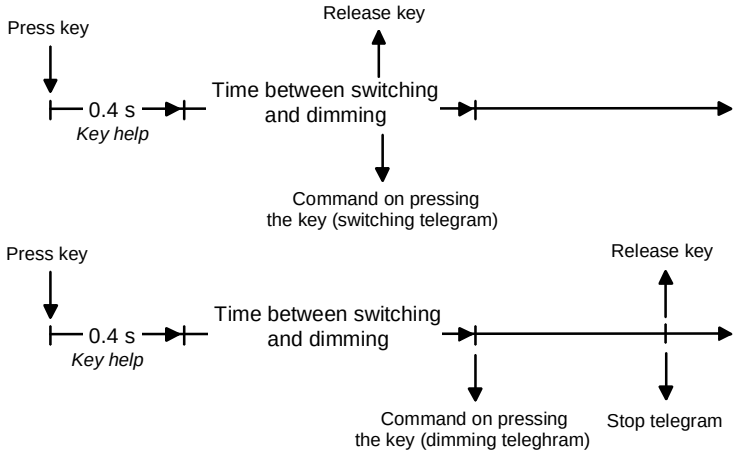
When the push button assistance function has been enabled the assigned function (e. g. switching, dimming, Shutter, etc.) will be executed upon the actuation of the corresponding button for at least 0.4 s.

Examples:

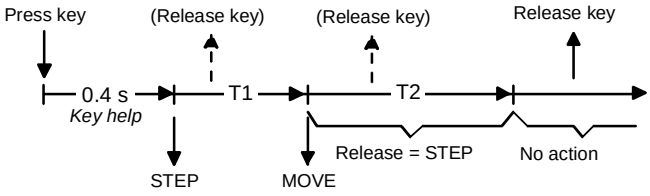
1. "Switching" function:



2. "Dimming" function:



3. "Shutter" function (e. g. "step - move - step" operating concept):



T1 = Time between short-time and long-time operation
T2 = Lamella adjustment time

If a button is actuated for shorter than 0.4 s its assigned function will not be executed. A status-LED parameterized as operation indication will only shine after the 0.4 s button help delay has elapsed.

The button assistance function is a useful means for beginners. For the proficient user, this function will no longer be necessary so that it can subsequently be deactivated after having been enabled in the ETS plug-in or re-activated at any time in the programming mode.
Please note that the button assistance function will be suppressed if a button is disabled.

Functional Description

4. Room Temperature Regulator Functions

4.1 Operating modes

The room temperature regulator has various operating modes. The selection of these modes will, for example, facilitate the activation of different temperature set values, depending on the presence of a person, on the state of the heating or cooling system, on the time of the day, or on the day of the week.

- Comfort mode:

  The comfort mode should be activated if persons are in a room, and the room temperature should, for this reason, be adjusted to an adequately convenient value. The switch-over into this mode can also be controlled by the presence of persons.
  The activated comfort mode will be indicated in the display by the " ⌂ " symbol.

- Standby mode

  If a room is not used during the day because persons are absent, for example, you can activate the standby mode. Thereby, you can adjust the room temperature on a standby value, thus to save heating or cooling energy, respectively.
  The activated standby mode will be indicated in the display by the " ⌂ " symbol.

- Night mode

  During the night hours or during the absence of persons for a longer time, it mostly makes sense to adjust the room temperature to lower values for heating systems (e. g. in bedrooms) or to higher values for cooling systems (e. g. in office rooms). For this purpose, you can activate the night mode.
  The activated night mode will be indicated in the display by the " ☾ " symbol.

- Frost/heat protection mode

  Frost protection will be required if, for example, the room temperature must not fall below critical values while the window is open. Heat protection can be required where the temperature rises too much in an environment which is always warm, mainly due to external influences.
  In such cases, you can activate the frost/heat protection mode and prescribe some temperature set value of its own for either option, depending on whether "heating" or "cooling" has been selected, to prevent freezing or overheating of the room.
  The activated frost/heat protection mode will be indicated in the display by the " ❄ " symbol.

- Comfort mode prolongation (temporary comfort mode)

  You can activate the comfort prolongation option from the night or frost/heat protection mode (not triggered by the *"window status"* object) and use it to adjust the room temperature to a comfort value for some time if, for example, the room is also 'used' during the night hours. This mode can exclusively be activated by a parameterized presence button or, in this case, also by the presence object, respectively. The comfort prolongation option will be automatically deactivated after a definable time has elapsed, or by pressing the presence button once more, or by receiving a presence object value = 0, respectively. You cannot re-trigger this prolongation.
  The activated comfort prolongation option will be indicated in the display by the " ⌂☾ " or " ⌂❄ " symbol.

You can assign an own temperature set value to each *"heating"* or *"cooling"* operating mode (refer to "4.4 Temperature set values", page 80).

Only one operating mode can be activated at a time so that both control circuits will always be in the same mode if two control circuits are used.

4.1.1 Changing the operating modes

You can activate or switch over the operating modes in various ways. Depending on one another in priority, activation or switching over is possible by…

a) local operation on the push button (programming mode), if enabled;

b) local operation on the push button (rockers 1 to 5 max.) and parameterized operating mode switch-over;

c) the objects separately available for each operating mode, or, alternatively, by the KONNEX objects.

About a):
You can activate the programming mode (refer to "1.4 Programming mode/local operation", page 25) to select the "comfort", "standby", "night" or "frost/heat protection (absence)" operating modes from the "operating mode switch-over" menu.
Switch-over to comfort prolongation will not be possible in the programming mode.

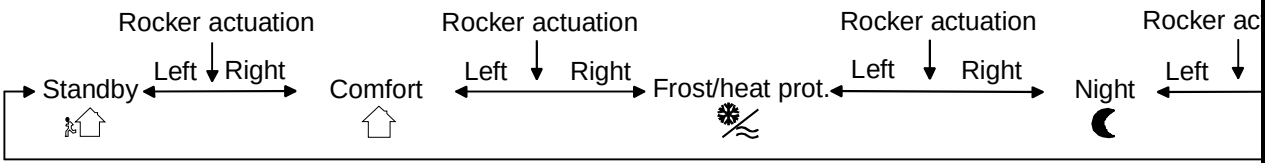
About b):
In addition to operation in the programming mode, you can parameterize the "operating mode switch-over" function to rockers 1 to max. 5 (depending on the variant configured) of the push button (refer to "3.2 Rocker functions", page 45). In this connection, distinction is made between button and rocker functions:

- Button function:
  The function of a button has been set to "operating mode switch-over". In this case, you can set in the ETS plug-in which operating mode you want to activate by pressing this button. For this purpose, the "comfort", "standby", "night" and "frost/heat protection" modes are available.

  To be able to activate the comfort prolongation option you can use the *"presence detection"* and the *"type of presence detection"* parameters in the *"room temperature regulator function / functionality"* parameter branch to additionally enable the presence button in the ETS plug-in. If enabled, the *"presence object"* will appear, and you can select the *"presence button"* setting from the button functions. In this way, you can actuate the presence button to change to the comfort prolongation option or to deactivate the latter prematurely when the night or frost/heat protection mode (not activated by the *"window status"* object) has been activated. Also, you can switch over from the standby to the comfort mode when you actuate the presence button.

  You can parameterize the function of the status-LED. In addition to the *"OFF"*, *"ON"* and *"operation indication"* standard settings, you can select the *"display operating mode active"* and *"display operating mode inactive"* options. Thus, the status-LEDs can indicate whether a function linked with the associated button is activated or not activated, respectively. For this purpose, the corresponding operating mode need not have been activated or deactivated by a button operation.

- Rocker function:
  The function of a rocker has been set to "operating mode switch-over". In such case, you can use the left or right button of the rocker to change the operating mode. The switch-over sequence will always be from the "comfort" to the "standby" and then to the "night" mode, followed by the "frost/heat protection" mode.



  The activation of the comfort prolongation option (presence function) will not be possible for a rocker function.

  Via the status object, you can trigger the status-LEDs of the rocker in the same way as for a push button rocker function, irrespective of the room temperature regulator operation.

If the room temperature regulator operation function is disabled local operation can be disabled by rockers 1 to max. 5, respectively (refer to "4.6.2 Disabling controller operation", page 92).

About c):
Distinction is made whether the operating modes should be switched over via separate 1 bit objects or, alternatively, by the 1 byte KONNEX objects. You can use the *"operating mode switch-over"* parameter in the *"room temperature regulator function"* parameter branch to set the way how to switch over.

• Operating mode switch-over through "switching" (4 x 1 bit):

There is a separate 1 bit switch-over object for each operating mode. Each of these objects allows the current operating mode to be switched over or to be set, depending on the priority.

Taking account of the priority, the following hierarchy will result from the operating mode switch-over by the objects, distinction being made between presence detection by the presence button (Table 1/Fig. 1) and presence detector (Table 2/Fig. 2 on the next page):

Table 1

"Operating mode switch over" objects				Window status Obj. no. 34	Presence button object Obj. no. 33	Activated operating mode
Obj. no. 31	Obj. no. 28	Obj. no. 29	Obj. no. 30			
X	X	X	X	1	X	Frost/heat protection
1	X	X	X	0	0	Frost/heat protection
0	1	X	X	0	0	Comfort
0	0	1	X	0	0	Standby
0	0	0	1	0	0	Night
1	X	X	X	0	1	Comfort prolongation
0	1	X	X	0	1	Comfort
0	0	1	X	0	1	Comfort
0	0	0	1	0	1	Comfort prolongation
0	0	0	0	0	0	Last valid mode set
0	0	0	0	0	1	Comfort/ Comfort prolongation *

X = irrelevant
*: Depending on the last valid mode set.

Fig. 1:

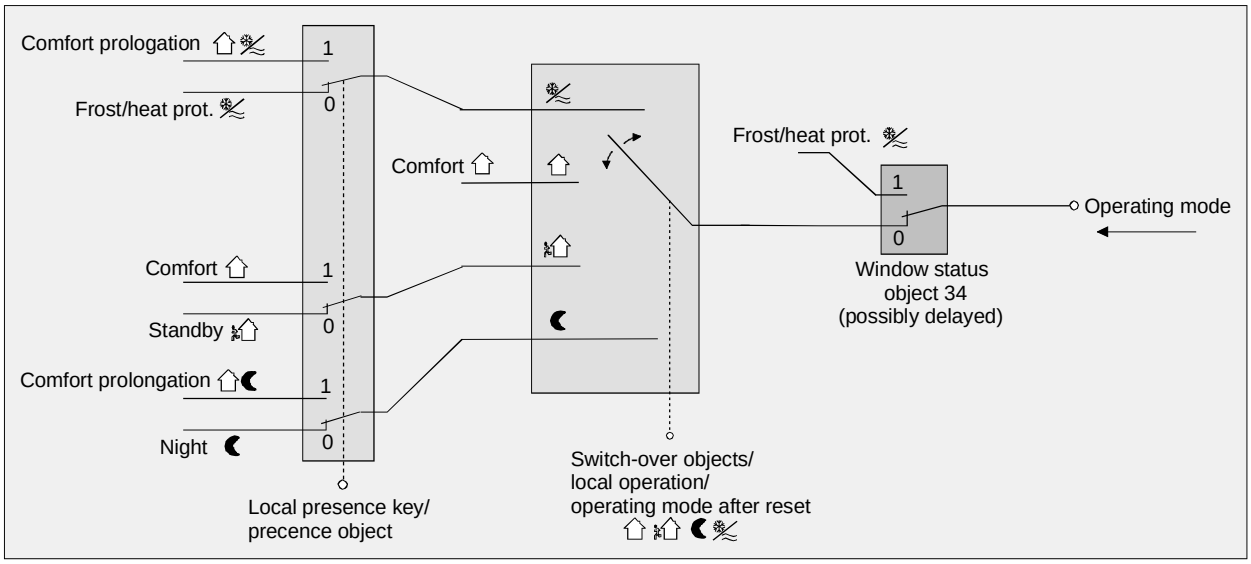
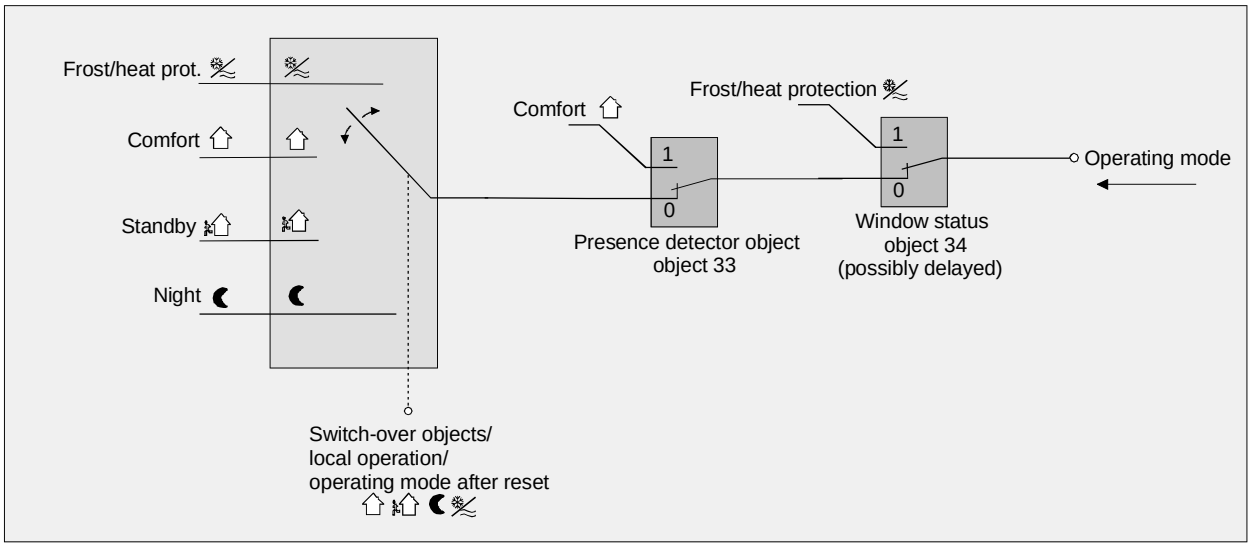


Table 2

"Operating mode switch over" objects				Window status Obj. no. 34	Presence detector object Obj. no. 33	Activated operating mode
Obj. no. 31	Obj. no. 28	Obj. no. 29	Obj. no. 30			
X	X	X	X	1	X	Frost/heat protection
X	X	X	X	0	1	Comfort
1	X	X	X	0	0	Frost/heat protection
0	1	X	X	0	0	Comfort
0	0	1	X	0	0	Standby
0	0	0	1	0	0	Night
0	0	0	0	0	0	Last valid mode set

X = irrelevant

Fig. 2:



Notes on operating mode switch-over through "switching" (4 x 1 bit):

- When the operating modes are switched over the objects (comfort / standby / night / frost/heat protection mode) will always be updated at the same time and can be read out, if necessary (set "R" flag). If the "T" flag has been set for these objects the current values will, in addition, be actively transmitted to the bus when they are changed. After bus voltage recovery or after initialization, respectively, the object which corresponds to the selected operating mode will be updated and its value actively transmitted to the bus if the "T" flag has been set.

- A switch-over via the objects is equal to a local switch-over made on the push button, taking account of the priorities of the operating modes. If no higher-priority mode (e. g. window contact/presence detector) has been activated you can use the button or rocker function to switch over on the device an operating mode preset by an object.

- When a presence button has been parameterized:
  The presence object will be active ("1") for the period of active comfort prolongation.
  The presence object will be automatically deleted ("0") if the comfort prolongation is stopped after the prolongation time has elapsed, or if the operating mode has been changed by a higher-priority operation through the switch-over objects or by local operation.

- If you use further B.IQ push button RTR as extensions to switch over the operating modes such switch-over should be solely effected by push buttons or rockers (push button functionality) which have been parameterized with the "switching" function. Otherwise (for example, for extension parameterization as "operating mode switch-over") the priority evaluation of the incoming telegrams can activate an undesired operating mode at the main unit (B.IQ push button RTR acting as room temperature regulator).

- Only one operating mode can be activated at a time so that both control circuits will always be in the same mode if two control circuits are used.
  The operating mode switch-over of the second control circuit always proceeds in parallel with the first control circuit.

• Operating mode switch-over through "value" (2 x 1 byte):

There is a common 1 byte switch-over object for all operating modes. During the running time, the operating mode can be switched over through this value object immediately after the receipt of only one telegram. In this connection, the value received will set the operating mode.
In addition, a second 1 byte object is available which, by forced control and of higher order, can set an operating mode, irrespective of any other switch-over options. According to the KONNEX specification, both 1 byte objects have been implemented.

Taking account of the priorities, the following hierarchy will result from the operating mode switch-over by the objects, distinction being made between presence detection by the presence button (Table 1/Fig. 1) and by the presence detector (Table 2/Fig. 2 on the next page):

Table 1

"Operating mode switch-over" object ** Obj. no. 28	"Override object mode" object *** Obj. no. 32	Window status Obj. no. 34	Presence button object Obj. no. 33	Activated operating mode
X	01	X	X	Comfort
X	02	X	X	Standby
X	03	X	X	Night
X	04	X	X	Frost/heat protection
X	00	1	X	Frost/heat protection
01	00	0	0	Comfort
02	00	0	0	Standby
03	00	0	0	Night
04	00	0	0	Frost/heat protection
01	00	0	1	Comfort
02	00	0	1	Comfort
03	00	0	1	Comfort prolongation
04	00	0	1	Comfort prolongation
00	00	0	0	Last valid mode set
00	00	0	1	Comfort/ Comfort prolongation *

*: Depending on the last valid mode set. / X = irrelevant
**: Values higher than "04" will not be evaluated. A value of "00" will keep active the last valid operating mode that has been set.
***: Values higher than "04" will not be evaluated. A value of "00" stands for a deactivated override object.

Fig. 1:

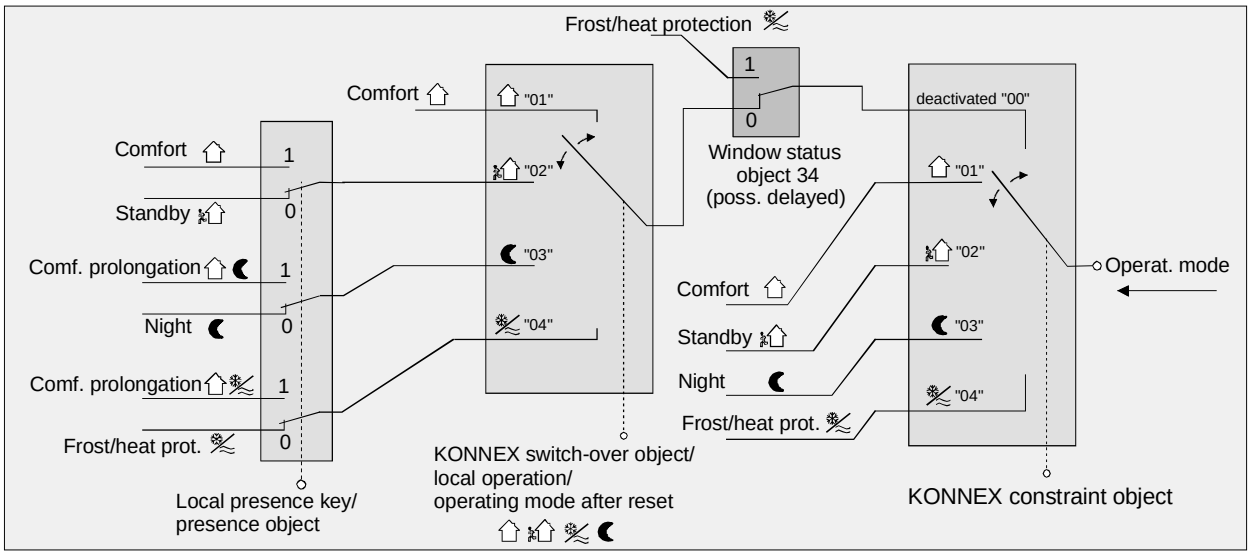
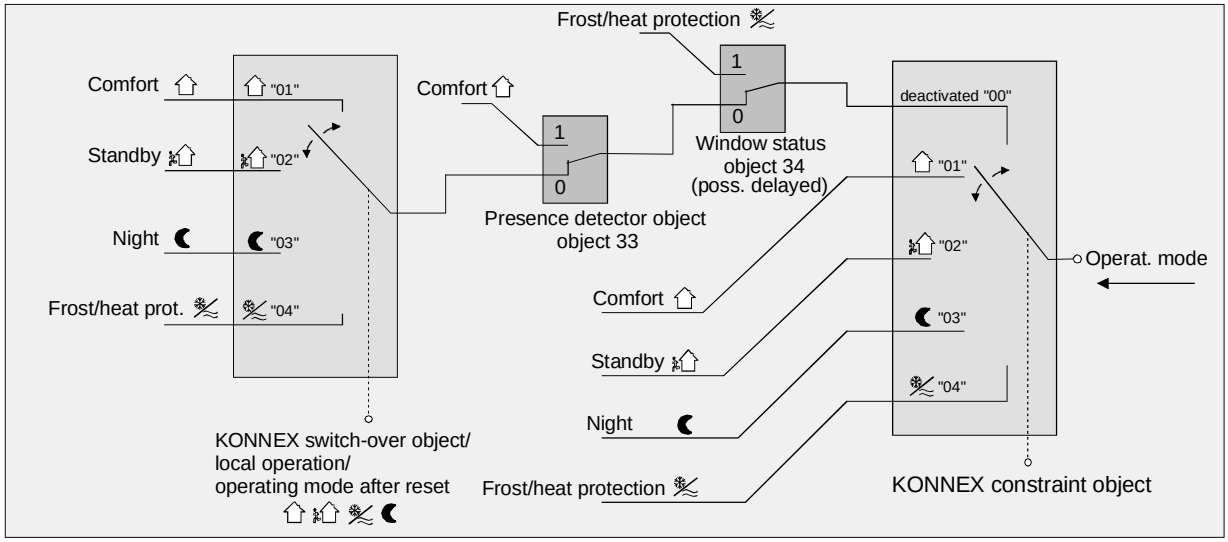


Table 2

"Operating mode switch-over" object ** Obj. no. 28	"Override object mode" object *** Obj. no. 32	Window status Obj. no. 34	Presence detector object Obj. no. 33	Activated operating mode
X	01	X	X	Comfort
X	02	X	X	Standby
X	03	X	X	Night
X	04	X	X	Frost/heat protection
X	00	1	X	Frost/heat protection
X	00	0	1	Comfort
01	00	0	0	Comfort
02	00	0	0	Standby
03	00	0	0	Night
04	00	0	0	Frost/heat protection
00	00	0	0	Last valid mode set

X = irrelevant
**: Values higher than "04" will not be evaluated. A value of "00" will keep active the last valid operating mode that has been set.
***: Values higher than "04" will not be evaluated. A value of "00" stands for a deactivated override object.

Fig. 2:



Notes on operating mode switch-over through "value" (2 x 1 byte):

- When the operating modes are switched over the KONNEX switch-over object will always be updated at the same time and can be read out, if necessary (set the "R" flag). If the "T" flag has been set for this object the current value will, in addition, be actively transmitted to the bus when it is changed. After bus voltage recovery or after initialization, respectively, the object which corresponds to the selected operating mode will be actively transmitted to the bus if the "T" flag has been set.

- A switch-over via the KONNEX switch-over object is equal to a local switch-over made on the push button, taking account of the priorities of the operating modes. If no higher-priority mode (e. g. window contact/ presence detector) and no KONNEX override object has been activated you can use the button or rocker function to switch over by temperature regulator operation on the device an operating mode preset by an object.

  The KONNEX override object will always have the highest priority.

- When a presence button has been parameterized:
  The presence object will be active ("1") for the period of active comfort prolongation.
  The presence object will be automatically deleted ("0") if the comfort prolongation is stopped after the prolongation time has elapsed, or if the operating mode has been changed by a higher-priority operation through the switch-over objects or by local operation, or if an operating mode forced by the KONNEX override object is being deactivated (override object → "00").

- Only one operating mode can be activated at a time so that both control circuits will always be in the same mode if two control circuits are used.
  The operating mode switch-over of the second control circuit always proceeds in parallel with the first control circuit.

4.1.2 Notes on the operating modes

Presence function / comfort prolongation:

By a presence detection, the room temperature regulator can quickly switch over to comfort prolongation upon a push button actuation or go into the comfort mode when a movement is being detected. In this connection, you can use the *"presence detection"* and *"type of presence detection"* parameters in the *"room temperature regulator function / functionality"* parameter branch to set whether presence detection should be movement-controlled by a presence detector or manual through button actuation:

- Presence detection by the presence button:

  If you enable the presence button for presence detection you can select the *"presence button"* setting from the push button functions. In addition, object 33, *"presence object"*, will be enabled.

  In this way, you can actuate the presence button or use a presence object value = "1" to switch over to comfort prolongation when the night or the frost/heat protection mode is active (not activated by the *"window status"* object or by automatic frost protection). The prolongation will be automatically deactivated as soon as the parameterized *"length of comfort prolongation"* time has elapsed. If you press the presence button once more, or if the object receives a value ="0", you can deactivate the comfort prolongation earlier. You cannot re-trigger such prolongation time.
  If you have set the length of comfort prolongation to "0" you cannot activate a comfort prolongation from the night or frost/heat protection mode. In this case, the operating mode will not be changed, although the presence function has been activated.
  If the standby mode is active you can actuate the presence button or use a presence object value = "1" to switch over to the comfort mode. This will also be the case if you have parameterized the length of comfort prolongation to "0". The comfort mode will remain active as long as the presence function remains active, or until another operating mode comes into effect.

  The presence object or the presence function, respectively, will always be deleted whenever a switch-over to a different operating mode takes place, or after a forced mode has been deactivated (associated with KONNEX forced switch-over). The presence object is bidirectional ("W" and "T" flags set by default) so that telegrams with the corresponding object values will be released upon activation (= "1") or deactivation (= "0"), respectively.
  A presence function including its object activated prior to a reset will always be deleted after the reset.

- Presence detection by the presence detector:

  If you enable a presence detector as type of presence detection object 33, *"presence object"*, will only appear. Via this object, you can integrate presence detectors into room temperature control.
  If a movement is detected ("1" telegram) the regulator will switch over into the comfort mode. In this connection, it will not be relevant what has been set by the switch-over objects or by local operation directly on the push button. Only the window contact or the automatic frost protection, or the KONNEX override object are of higher priority.
  After the delay time has elapsed in the presence detector ("0" telegram), the regulator will return to the mode which was active before presence detection, or it will compensate the telegrams of the switch-over objects received during presence detection, respectively.
  During active presence detection, you cannot change the operating mode on the push button.
  A presence function activated prior to a reset will always be deleted after the reset. In this case, the presence detector must transmit a new "1" telegram to activate the presence function.

Window status / automatic frost protection:

The B.IQ push button RTR offers various options to switch over into the frost/heat protection mode. In addition to the switch-over by the corresponding operating mode switch-over object or by an operating mode switch-over on the push button (button function), the frost/heat protection mode can by activated by a window contact or, alternatively, frost protection can be activated by an automatic temperature control option. In this connection, the window contact or the automatic control is given a higher priority among these options (refer to "4.1.1 Changing the operating modes", page 57).
You can use the *"frost/heat protection"* parameter in the *"room temperature regulator function"* parameter branch to set the way how such higher-priority switch-over will take place:

- Frost/heat protection switch-over *"via window status"*:
  Object 34, *"window status",* is enabled. A telegram having the value of = "1" (open window) and sent to this object will activate the frost/heat protection mode. If this is the case, this operating mode cannot be deactivated, neither by local operation nor by the switch-over objects (with the exception of the KONNEX override object).
  Only a telegram with the value of = "0" (closed window) will reset the window status and deactivate the frost/heat protection mode. After this, the operating mode set before the opening of the window or that mode carried by the bus while the window was open will be activated.

  You can optionally parameterize a window status delay. Such delay can make sense if short ventilation of the room by opening the window is not supposed to change the operating mode. You can use the *"window status delay"* parameter to set this delay time between 1 and 255 minutes. The window status will only be changed and thus the frost/heat protection mode activated after this parameterized time has elapsed. A setting of "0" will effect the immediate activation of the frost/heat protection mode when the window is open. The window status will be in effect in the heating and in the cooling mode. The *"window status"* object will be deleted after a reset.

- Frost protection mode switch-over by *"automatic frost protection"*:
  For this setting, automatic switch-over to the frost protection mode can be made at times, depending on the room temperature determined. If there are no window contacts, this setting can prevent unnecessary heating up of the room when windows or external doors are open.
  In connection with this function, a quick temperature drop can be detected by measuring the actual temperature every minute as, for example, is the case when a window is open.
  If the temperature decrease detected reaches a parameterized value the room temperature regulator will automatically switch over to the frost protection mode. You can use the *"automatic frost protection"* parameter to set the maximum temperature drop in K/min for switching over to the frost protection mode.
  After the time preset by the *"frost protection period in automatic mode"* parameter has elapsed, the regulator will return into the mode which was set before frost protection. Re-triggering will not be possible.
  If a switch-over was made by the objects (4 x 1 bit or 1 byte) during frost protection and new operating mode was received this followed-up mode will be set after automatic frost protection.
  The KONNEX override object has a higher priority than the automatic frost protection mode and can interrupt the latter.

  The automatic frost protection mode only acts on heating for temperatures below the set value temperature of the operating mode selected. Thus, no automatic switch-over to frost protection can take place at room temperatures in the dead band or in the active cooling mode if the "heating and cooling" mode is on. Automatic heat protection activation is not intended with this parameterization.

  Compared with the alternative setting of the frost/heat protection detection by the window contact, the automatic frost protection mode will have the same priority when the operating mode is being changed.

  Note:
  Frequent draughts in a room can cause unintentional activation/deactivation of frost protection when the automatic frost protection mode is active, and if the parameterized temperature decrease is not low enough. Switching into the frost/heat protection mode by window contacts should generally be preferred to the automatic option.

Operating mode after reset:

In the ETS plug-in, you can use the *"operating mode after reset"* parameter in the *"room temperature regulator function / functionality"* parameter branch to set which operating mode you want to be activated after bus voltage recovery, after a programming process by the ETS, or after re-plugging the application module to the bus coupling unit. In this connection, the following settings will be possible:

- *"Comfort operation"*:	The comfort mode will be activated after the initializing phase.
- *"Standby operation"*:	The standby mode will be activated after the initializing phase.
- *"Night peration"*:	The night mode will be activated after the initializing phase.
- *"Frost/heat protection operation"*:	The frost/heat protection mode will be activated after the initializing phase.
- *"Restore operation mode before reset"*:	The mode activated before a reset will be restored after the initializing phase of the device.

The objects associated with the activated operating mode will be updated after a reset.

Notes on the *"restore operation mode before reset"* setting:

- Frequent changing of the operating mode (e. g. several times a day) during running operation can adversely affect the life of the device as the read-only memory (EEPROM) used has been designed for less frequent write access events only.
- A presence function including its object activated prior to a reset will be deleted after the reset. The operating mode caused by the presence function, however, will remain active after the reset.
  A comfort prolongation restarted by a reset will be automatically deactivated after its delay time has elapsed.
  The *"window status"* object will be deleted after a reset. Also in this case, the frost/heat protection mode activated before by a window status will remain activated even after a reset.

4.1.3 Controller status

The room temperature regulator can send out its status. For this purpose, an optional collective status signal (1 byte type) or, alternatively, one up to eight single status signals (1 bit type) are available.
The *"status indication of controller"* parameter in the *"room temperature regulator function / variable and status output"* parameter branch will enable the status signal and set the status format.

- *"Status indication of controller" = "controller general"*:

One-byte status object 36 contains the entire status information. Controlled by the control algorithm, the status will be actively transmitted to the bus in cycles every 30 seconds (provided that the "T" flag has been set). If you set the "R" flag you can read out the status.

Setting	Data Description	
Regulator in general 1 byte type	Bit 0: 1: Comfort operation active Bit 1: 1: Standby operation active Bit 2: 1: Night operation active Bit 3: 1: Frost/heat protection active	Bit 4: 1: Controller disabled Bit 5: 1: Heating; 0: cooling Bit 6: 1: Controller inactive (dead band) Bit 7: 1: Frost alarm ($T_{room} \leq + 5$ °C)

- *"Status indication of controller" = "transmit individual state"*:

One-bit status object 36 contains the status information selected by the *"single state"* parameter. Controlled by the control algorithm, the status will be actively transmitted to the bus in cycles every 30 seconds (provided that the "T" flag has been set).If you set the "R" flag you can read out the status.

Parameterization for "Individual Status"	Data Description	
Comfort operation activated	1: Comfort mode/prolongation active	0: No comfort mode
Standby operation activated	1: Standby mode active	0: No standby mode
Night operation activated	1: Night mode active	0: No night mode
Frost/heat protection active	1: Frost/heat protection mode active	0: No frost/heat protection mode
Controller disabled	1: Controller disabled (dew-point operation)	0: Controller not disabled
Heating / Cooling	1: Heating mode	0: Cooling mode
Controller inactivated	1: Controller inactive (dead band)	0: Controller active
Frost alarm	1: Frost alarm ($T_{room} \leq + 5$ °C)	0: No frost alarm ($T_{room} > + 5$ °C)

Explanation of the status signals:

- Comfort operation activated: It will be active if the "comfort ' ⌂ '" mode or some comfort prolongation " ⌂ ☾ " or " ⌂ ≈ ", respectively, has been activated.
- Standby operation activated: It will be active if the "standby ' ⌂ '" mode has been activated.
- Night operation activated: It will be active if the "night " ☾ " mode has been activated.
- Frost/heat protection active: It will be active if the "frost/heat protection ' ≈ '" mode has been activated.
- Controller disabled: It will be active if regulator disabling has been activated (dew-point operation).
- Heating / Cooling: It will be active if the heating mode has been activated, and will be inactive when the cooling mode has been activated. (It will be inactive if the regulator has been disabled.)
- Controller inactivated: It will be active in the *"heating and cooling"* mode, if the room temperature determined is within the dead band. In the individual *"heating"* or *"cooling"* mode, this status information will always be "0". (It will be inactive if the regulator has been disabled.)
- Frost alarm: It will be active if the room temperature determined has reached +5 °C or is below this value. This status signal will have no special influence on the control behaviour.

Upon a reset, status object 36 will be updated after the initializing phase. After this, the status will be updated every 30 seconds in parallel with the variable calculation of the regulator variables.

4.2 Heating/cooling modes and heating/cooling mode switch-over

The room temperature regulator has up to two different modes. These modes specify whether you want the regulator to use its variable to trigger heating systems *("heating"* single mode) or cooling systems *("cooling"* single mode). You can also activate mixed operation, with the regulator being capable of switching over between *"heating"* and *"cooling"* automatically or, alternatively, controlled by an object.
In addition, you can establish two-stage control operation for triggering an additional heating or cooling unit. For two-stage control, separate variables will be calculated as a function of the temperature deviation between the set value and the actual value and transmitted to the bus for the basic and additional stages.
In this connection, the *"heating/cooling mode"* parameter in the *"room temperature regulator functions"* parameter branch sets the operating mode to be executed and, if necessary, enables the additional stage(s).

In the individual *"heating"* or *"cooling"* modes without any additional stage, the regulator will always work with one variable and, alternatively, when the additional stage is enabled, it will use two variables in the parameterized mode. Depending on the room temperature determined and on the specified set value temperatures of the operating modes (refer to "4.4 Temperature set values", page 80), the room temperature regulator will automatically decide whether heating or cooling energy will be required and will calculate the variable for the heating or cooling system (refer to "4.3 Room temperature control and variables", page 70).
For *"heating"* or *"cooling"*, the regulator will always be in the mode set in the ETS plug-in after a reset (bus voltage recovery, re-programming by the ETS, or re-plugging of the application module).

In the *"heating and cooling"* mode, the regulator is capable of triggering heating and cooling systems. In this connection, you can set the switch-over behaviour of the modes.

- *"Switch-over between heating and cooling"* parameter in the *"room temperature regulator functions"* parameter branch set to *"automatically"*:

  In this case, a heating or cooling mode will be automatically activated, depending on the room temperature determined and on the given temperature basic set value, or on the dead band, respectively. If the room temperature is within the preset dead band neither heating nor cooling will take place (both variables = "0"). In this connection, the display will read the heating temperature set value of the activated operating mode when you actuate the display buttons. If the room temperature is higher than the cooling temperature set value cooling will take place. If the room temperature is lower than the cooling temperature set value heating will take place.

  When the heating/cooling mode is changed automatically information can be actively sent to the bus through object 35, *"heating/cooling switch-over",* to indicate whether the regulator is working in the heating mode ("1" telegram) or in the cooling mode ("0" telegram). In this connection, the *"automatic heating/cooling switch-over transmission"* parameter specifies when an operating mode switch-over will be transmitted.

  - *"On changing the heating/cooling"* setting:
    In this case, a telegram will be transmitted solely on changing from heating to cooling (object value = "0") or from cooling to heating (object value = "1"), respectively.

  - *"On changing the output value"* setting:
    By this setting, the current mode will be transmitted whenever the output variable changes. If the variable = "0" the mode which was active last will be transmitted.

  If the room temperature determined is within the dead band the mode activated last will be retained in the object until a switch-over into the other mode takes place, if necessary.

  In addition, the object value can be output in cycles when automatic switch-over is being made. The *"cyclical transmission heating/cooling switch-over"* parameter enables cyclic transmission (factor > "0" setting) and specifies the cycle time.

  Note on automatic mode switch-over:

  Selecting too narrow a dead band may possibly result in permanent changing between heating and cooling. For this reason, you should, if possible, not set the dead band (temperature difference between the set value temperatures for the comfort heating and cooling modes) below the default value.

- *"Switch-over between heating and cooling"* parameter in the *"room temperature regulator functions"* parameter branch set to *"via object"*:

  In this case, the heating/cooling mode will be controlled via object 35, *"heating/cooling switch over"*, independently of the dead band. This type of switch-over can, for example, become necessary if both heating and cooling should be effected through a one-pipe system (heating and cooling system). For this purpose, the temperature of the medium in the one-pipe system must, first of all, be changed by the control system of the installation. Subsequently, you can select the heating/cooling switch-over through the object (often, cold water is used in the one-pipe system for cooling in summer, while hot water is used for heating in winter).
  The *"heating/cooling switch over"* object has the following polarities: "1": heating; "0" cooling. After a reset, the object value will be "0", with *"heating/cooling switch-over after reset"* being activated.

  You can use the *"heating/cooling switch-over after reset"* parameter to set which mode you want to activate after a reset. For the *"heating"* or *"cooling"* settings, the regulator will activate the parameterized heating/cooling switch-over immediately after the initializing phase. If you have parameterized *"heating/cooling switch-over before reset"* the mode which was selected before the reset will be activated.
  If a switch-over is made through the object the mode will first be changed into the one specified to be activated after a reset. Only after the device receives an object update a switch-over into the other mode will take place, if necessary.

  Notes on the *"heating/cooling switch-over before reset"* setting:

  - Frequent changing of the operating mode (e. g. several times a day) during running operation can adversely affect the life of the device as the read-only memory (EEPROM) used has been designed for less frequent write access events only.

Heating and cooling in the mixed mode at the same time (variables > "0") is, in general, not possible.
Only if heating or cooling energy is required in one of the modes and, consequently, the variable is > "0" the " $\overline{+\text{≋}}$ " or " $\overline{-\text{≋}}$ " symbol will appear in the display.

Heating/cooling indication:

Depending on the mode set, you can output information through separate objects whether heating or cooling energy is required at the moment, i. e. whether heating " $\overline{+\text{≋}}$ " or cooling " $\overline{-\text{≋}}$ " takes place.
As long as the heating (cooling) variable is > "0", a "1" telegram will be transmitted through the *"heating"* (*"cooling"*) signal object. Only after the variables have become = "0", the signal telegrams will be reset ("0" telegram being transmitted).

Exception: For two-point control it should be noted that the " $\overline{+\text{≋}}$ " or " $\overline{-\text{≋}}$ " symbol will appear in the display, or that the heating or cooling signal objects will already become active once the temperature falls below the set value of the active operating mode for heating or exceeds that for cooling. In this connection, the parameterized hysteresis will be disregarded (refer to "4.3.1 Control algorithms and calculation of variables", page 70).

Heating and cooling at the same time will not be possible. The signals are exclusively referred to control circuit 1.

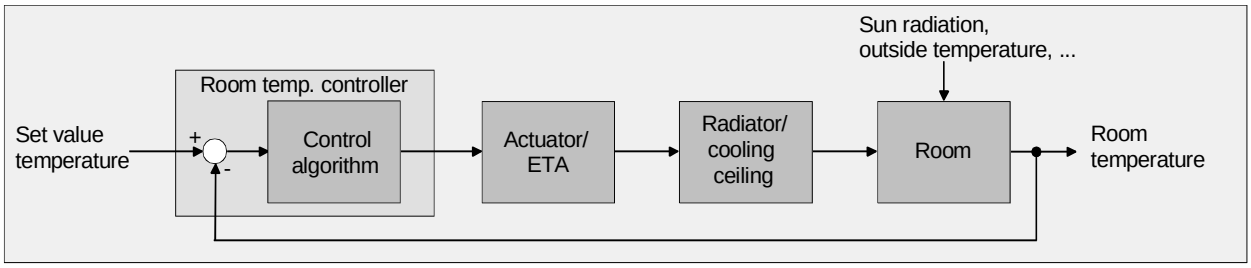
The indication objects can be enabled by the *"heating indication"* or *"cooling indication"* parameters in the *"variable and status output"* parameter branch.

The control algorithm (refer to "4.3 Room temperature control and variables", page 70) controls the signal objects.
Please note that the variable is re-calculated only every 30 s, followed by an updating of the signal objects.

4.3 Room temperature control and variables

4.3.1 Control algorithms, control circuits and calculation of variables

To facilitate convenient temperature control in a living room a specific algorithm which controls the installed heating or cooling systems is required. Taking account of the preset temperature set values and the actual room temperature, the regulator thus determines variables which trigger the heating or the cooling system.
The control system (control circuit) consists of a room temperature regulator, an actuator or switching actuator (when ETA electrothermal drives are used), the actual heating or cooling element (e. g. radiator or cooling ceiling) and of the room. This results in the following controlled system:



The regulator measures the actual temperature (room temperature determined) and compares it with the preselected set value temperature. With the aid of the selected control algorithm, the variable is then calculated from the difference between the actual and the set value temperature. By resetting the variable at regular intervals, the regulator is thus capable of compensating in the control circuit temperature differences between the actual and the desired values caused by external influences (e. g. intensive sun radiation or varying outside temperatures). In addition, the flow temperature of the heating or cooling circuit influences the control system which necessitates adaptations of the variable.

The room temperature regulator in the B.IQ push button RTR facilitates either proportional/integral (PI) control as a continuously working or switching option, or switching 2-point control.

In some practical cases, it can become necessary to use more than one control algorithm. For example, in bigger systems using floor heating, one control circuit which solely triggers the floor heating can be used to keep the latter at a constant temperature. The radiators on the wall, and possibly even in a side area of the room, will be controlled separately by another algorithm.
In such cases, distinction must be made between the different types of control, as floor heating systems, in most cases, require control parameters which are different to those faster-response radiators will need. Moreover, there are cases where the different control systems require different variables with different object widths (1 bit or 1 byte type). This would justify the use of a second regulator.

The B.IQ push button RTR offers the option to activate one or, alternatively, two control circuits. The *"control circuits"* parameter in the *"room temperature regulator function"* parameter branch sets the number of control circuits.

- Using one control circuit:
  If you use only one control circuit you can parameterize the *"heating"*, *"cooling"* or, as an alternative, the mixed *"heating and cooling"* modes. You can also use additional stages in any cases.
  In this connection, you can set different control algorithms for the heating and/or cooling system. Thus, you can use up to four separate algorithms for two-stage heating or cooling operation.

- Using two control circuits:
  If you use two control circuits you can only choose between the *"heating"* or *"cooling"* mode. In this connection, both control circuits will always work in the same operating mode (comfort, standby, etc.). However, you can set different control algorithms for both control circuits. For this type of parameterization, the use of two-stage control is not intended. Both control circuits can alternatively work with joint or with separate set values (refer to "4.4 Temperature set values", page 80).

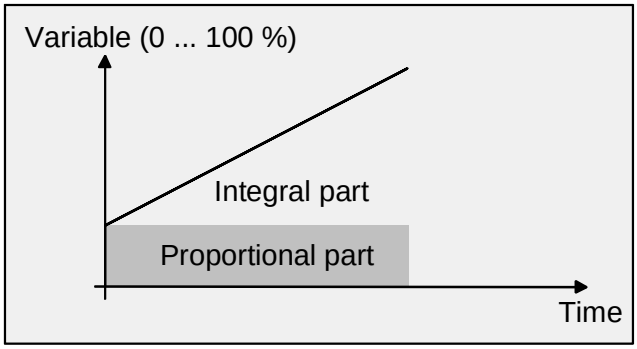
The variables calculated by the control algorithm are output via the *"heating variable"* or *"cooling variable"* communication objects. Depending on the control algorithm selected for the heating and/or cooling mode, the format of the variables objects is, among other things, also specified. So you can create 1 bit or 1 byte variable objects (refer to "4.3.3 Output of variables", page 78).

You can use the *"type of heating control"* or *"type of cooling control"* parameter in the *"room temperature regulator function"* parameter branch to specify the control algorithm, if necessary, also for the additional stages or for both control circuits, respectively.
In this connection, you can select each of the three following algorithms:

1. Continuous PI control:

PI control is an algorithm which consists of a proportional and of an integral part. By the combination of these control properties, you can obtain room temperature control as fast and precise as possible without or only with low deviations.

When you use this algorithm, the room temperature regulator will calculate a new continuous variable in cycles of 30 seconds and send it to the bus via a 1 byte value object if the calculated variable value has changed by a specified percentage. You can use the *"automatic transmission at modification by…"* parameter in the *"room temperature regulator function / variable and status output"* parameter branch to set the change interval in per cent.



An additional heating or cooling stage as PI control works in the same way as the PI control of the basic stage, with the exception that the set value will shift, taking account of the parameterized step width.

Special features of the PI control:

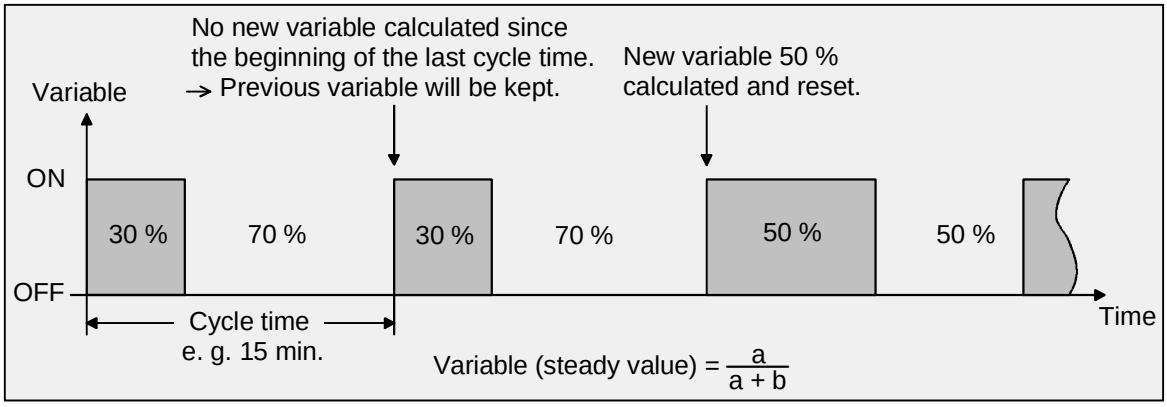
If the room temperature deviation between the actual value and the set value is high enough to have a 100 % variable the room temperature regulator in the B.IQ push button RTR will work with this maximum variable until the room temperature measured has reached its set value. This particular behaviour is known as 'clipping'.
This way, rapid heating up of undercooled rooms or quick cooling in overheated rooms will be achieved. In two-stage heating or cooling systems, this control behaviour also applies to the variables of the additional stages.

2. Switching PI control

For this parameterization, the room temperature will also be kept constant by the PI control algorithm. Taking the mean value for a given time, the same behaviour of the control system will result as you would obtain with a continuous regulator. The difference compared with continuous control is only the way how the variable is output.
The variable calculated by the algorithm in cycles of every 30 seconds is internally converted into a pulse-width-modulated (PWM) variable signal and sent to the bus via a 1 bit switching object after the cycle time has elapsed.
The mean value of the variable signal resulting from this modulation is a measure for the averaged position of the control valve, thus being a reference to the room temperature set, taking account of the cycle time which you can set through the *"cycle time of the switching variable…"* parameter in the *"room temperature regulator function / variable and status output"* parameter branch.
A shift of the mean value, and thus a change in the heating capacity, can be obtained by changing the duty factor of the switch-on and switch-off pulses of the variable signal.

The duty factor will be adapted by the regulator only at the end of a time period, depending on the variable calculated. This applies to any change of the variable, regardless of what the ratio is by which the variable changes (the *"automatic transmission at modification by…"* and *"cycle time for automatic transmission…"* parameters will have no function in this case). Each variable value calculated last during an active time period will be converted. Even after you have changed the set value temperature, for example, by changing the operating mode, the variable will still be adapted after the end of an active cycle time.

The illustration below shows the variable switching signal output in dependence on the internally calculated variable value (first of all, a variable of 30 %, then of 50 %, with the variable output not being inverted).



For a variable of 0 % (permanently OFF) or of 100 % (permanently ON), a variable telegram corresponding to the variable value ("0" or "1") will always be sent after a cycle time has elapsed. 'Clipping' (refer to "continuous PI control") will also be active for this type of control.

Also for switching PI control, the regulator will always use continuous variable values for internal calculation. Such continuous values can additionally be sent to the bus via a separate 1 byte value object, for example, as status information for visualization purposes.
Therefore, when you use switching PI control (RWM), value object 46 will be sent for the heating mode, and value object 48 for the cooling mode. If you use additional stages, value object 47 for additional heating operation and value object 49 for additional cooling operation will be additionally enabled. For the use of two control circuits, the separate 1 byte value object will not be available.
If you want to output the heating and cooling variables through a joint object (refer to "4.3.3 Output of variables", page 78) the continuous value of the activated operating mode will be transmitted via object 46 and, if necessary, via object 47 for the additional stages.
The status value objects will be updated at the same time as the variable is output and will only take place after the parameterized cycle time has elapsed. The *"automatic transmission at modification by…"* and *"cycle time for automatic transmission…"* parameters will have no function in this case.

An additional heating or cooling stage as switching PI control works in the same way as the PI control of the basic stage, with the exception that the set value will shift, taking account of the parameterized step width. All PWM control options will use the same cycle time.

Cycle time:

The pulse-width-modulated variables are mainly used for triggering electrothermal drives (ETA). In this connection, the room temperature regulator sends the switching variable telegrams to a switching actuator preferably equipped with semiconductor switching elements which the drives are connected to.

By setting the cycle time of the PWM signal, you can adapt the control to the drives used. The cycle time sets the switching frequency of the PWM signal and allows adaptation to the adjusting cycle times (the adjusting time it takes the drive to bring the valve from its completely closed to its completely opened position) of the actuators used. In addition to the adjusting cycle time, take account of the dead time (the time in which the actuators do not show any response when being enabled or off). If different actuators with different adjusting cycle times are used take account of the longest of the times. Always note the information given by the manufacturers of the actuators.

In common practice, two different options of how to set the cycle time can be identified:

I. Cycle time > 2 x adjusting cycle time of the actuators used (ETA), e. g. 15 minutes (default).

In this case, the switch-on or switch-off times of the PWM signal are long enough for the actuators to have sufficient time to fully open or fully close within a given time period.

Advantages:
The desired mean value for the variable and thus for the required room temperature will be set relatively precisely, even for several actuators triggered at the same time.

Disadvantages:
It should be noted, that, due to the full valve lift to be continuously 'swept', the life expectancy of the actuators can diminish. For very long cycle times (> 15 minutes) with less sluggishness in the system (e. g. for smaller warm-water radiators), the heat emission into the room, for example, in the vicinity of the radiators, can possibly be non-uniform and be found disturbing.

Important:

- Such setting is recommended for slower, more sluggish heating systems (such as floor heating).
- Even for a bigger number of triggered actuators, maybe of different types, this setting can be recommended to be able to obtain a better mean value of the adjusting travels of the valves.

II. Cycle time > adjusting cycle time of the actuators used (ETA), e. g. 2 minutes

In this case, the switch-on or switch-off times of the PWM signal are too short for the actuators to have enough time to fully open or fully close within a given time period.

Advantages:
This setting ensures continuous water flow through the radiators, for example, thus facilitating uniform heat emission into the room.
If only one actuator is triggered the regulator can continuously adapt the variable to compensate the mean value shift caused by the short cycle time, thus setting the desired room temperature.

Disadvantages:
If more than one actuator is triggered at the same time the desired mean value will become the variable, which will result in a very poor adjustment of the required room temperature, or in adjustment of the latter with major deviations, respectively.

Note:

- Such cycle time setting is recommended for less sluggish heating systems (such as warm-water radiators with a higher flow temperature).

3. Switching 2-point control:

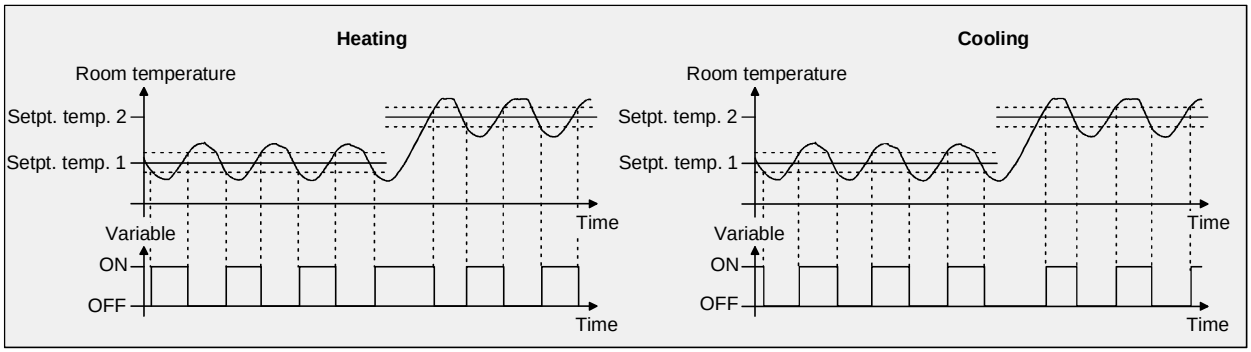
The 2-point control represents a very simple type of temperature control. For this type of control, two hysteresis temperature values are preset. The actuators are triggered by the regulator via switch-on and switch-off variable commands (1 bit type). A continuous variable is not calculated for this type of control. The room temperature is also evaluated by this type of control in cycles every 30 seconds, i. e. the variables will only change at these moments, if required.
The disadvantage of a continuously varying temperature as a result of this option is in contrast with the advantage of this very simple 2-point room temperature control. For this reason, sluggish heating or cooling systems should not be triggered by a 2-point control system, for this can lead to very high overshooting of the temperature, thus resulting in loss of comfort.

When presetting the hysteresis limits, you should distinguish between the following operation modes:

- *"Heating"* or *"cooling"* single modes:

In the heating mode, the regulator will turn on the heating when the room temperature has fallen below a preset limit. The control system will only turn off the heating once a preset temperature limit has been exceeded.
In the cooling mode, the regulator will turn on the cooling system when the room temperature has exceeded a preset limit. The control system will only turn off the cooling system once the temperature has fallen below a preset limit.
In this connection, variable "1" or "0" will be output, depending on the switching status, if the temperature exceeds or falls below the hysteresis limits.
Please note that the " + " or " - " symbol will appear in the display, or that the heating or cooling signal objects will already become active once the temperature is falling below the set value of the active operating mode for heating or is exceeding that for cooling. The hysteresis will be disregarded in this case.
You can parameterize the upper or lower hysteresis limit of the two operating modes in the ETS plug-in.

The following illustration shows a 2-point control example for the *"heating"* or *"cooling"* single modes (heating on the left and cooling on the right; two temperature set values; single-stage heating or cooling; non-inverted variable output).

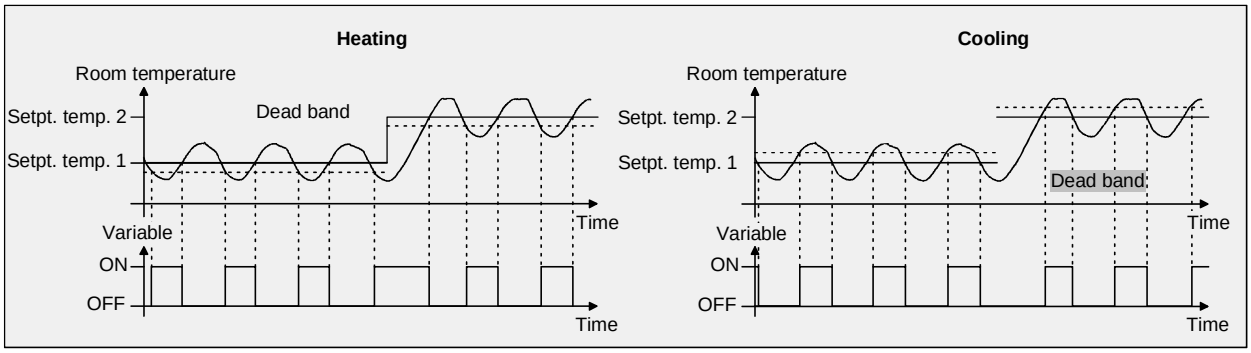


An additional heating or cooling stage as 2-point control works in the same way as the 2-point control of the basic stage, with the exception that the set value and the hysteresis values will shift, taking account of the parameterized step width.

- *"Heating and cooling"* mixed mode:

In the heating mode, the regulator will turn on the heating when the room temperature has fallen below a preset limit. As soon as the room temperature is exceeding the set value of the current operating mode, the control will turn off the heating in the heating mode.
In the cooling mode, the regulator will turn on the cooling system when the room temperature has exceeded a preset limit. As soon as the room temperature is falling below the set value of the current operating mode, the control will turn off the cooling system in the cooling mode.
Thus, there is no longer any upper hysteresis limit for heating or no lower one for cooling, respectively, for these values would be in the dead band. Within the dead band, neither heating nor cooling will take place.
In this connection, variable "1" or "0" will be output, depending on the switching status, if the temperature exceeds or falls below the hysteresis limits or the set values, respectively.
Please note that the " + " or " - " symbol will appear in the display, or that the heating or cooling signal objects will already become active once the temperature is falling below the set value of the active operating mode for heating or is exceeding that for cooling. The hysteresis will be disregarded in this case.
You can parameterize the upper or lower hysteresis limit of the two operating modes in the ETS plug-in.

The following illustration shows a 2-point control example for the *"heating"* and *"cooling"* mixed mode (activated heating on the left and activated cooling on the right; two temperature set values; non-inverted variable output).



An additional heating or cooling stage as 2-point control works in the same way as the 2-point control of the basic stage, with the exception that the set value and the hysteresis values will shift, taking account of the parameterized step width.

4.3.2 Adapting the control algorithms

4.3.2.1 Adapting the PI control

There are various installation systems which can heat or cool a room. By the use of heat transfer media (preferably water or oil) in conjunction with room air convection, it is, for example, possible to uniformly heat or cool the environment. Such systems are, for example, used in heating panels, floor heating systems or cooling ceilings.
As an alternative, or in addition, fan systems can heat or cool rooms. In most cases, these are electric fan heating or cooling systems or refrigeration compressors with fans. Due to the direct heating or cooling of the room air, such heating or cooling systems are rather quick.

To enable the PI control algorithm to efficiently control any conventional heating or cooling systems so that room temperature control works as fast as possible and without any deviations it will be necessary to adjust the control parameters.
For this purpose, some factors which sometimes influence the control behaviour to a considerable extent can be set for a PI control system. For this reason, you can set the room temperature regulator to pre-defined 'experience values' for the most common heating or cooling systems. If the preset values do not give you any satisfactory control result for the corresponding heating or cooling system you have chosen you can optionally use control parameters to optimize adaptation.

You can use the *"type of heating"* or *"type of cooling"* parameters to set pre-defined control parameters for the heating or cooling stage and, if necessary, for the additional stages. These fixed values correspond to field-proven values of a properly planned and installed air conditioning system and will result in an optimum temperature control behaviour. The following can be set for heating or cooling operation, respectively:

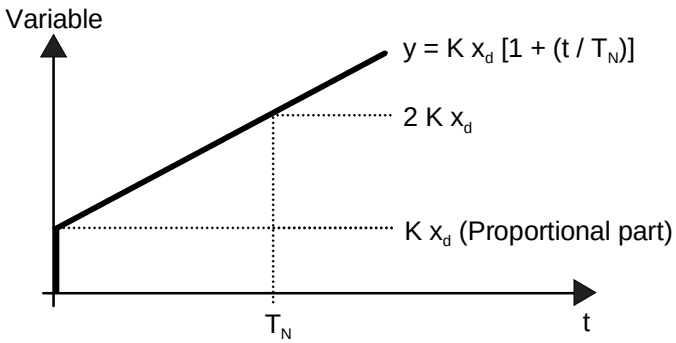
For Heating Control				
Type of Heating	**Preset Values**		**Recommended**	**Recommended**
	Proportional Range	**Reset Time**	**PI Control Type**	**PWM Cycle Time**
• Warm water heating	5 Kelvin	150 minutes	Contin./PWM	15 min **
• Floor heating	5 Kelvin	240 minutes	PWM	15 - 20 min
• Electric heating	4 Kelvin	100 minutes	PWM	10 -15 min
• Blower convector	4 Kelvin	90 minutes	Continuous	---
• SplitUnit *	4 Kelvin	90 minutes	PWM	10 -15 min
For Cooling Control				
Type of Cooling	**Preset Values**		**Recommended**	**Recommended**
	Proportional Range	**Reset Time**	**PI Control Type**	**PWM Cycle Time**
• Cooling ceiling	5 Kelvin	240 minutes	PWM	15 - 20 min
• Blower convector	4 Kelvin	90 minutes	Continuous	---
• SplitUnit *	4 Kelvin	90 minutes	PWM	10 -15 min

*: split, mobile air conditioning unit
**: For smaller, quicker radiators, (e. g. at a higher flow temperature) PWM cycle time 2 - 3 minutes.

If you have set the *"type of heating"* or *"type of cooling"* parameters to *"via control parameter"* you can 'manually' match the control parameters. You can preset the proportional band for heating or cooling (P part) and the reset time for heating or cooling (I part) to influence the control to a considerable extent.

Important:

- Even varying the control parameters by low amounts will lead to a clearly different control behaviour.
- Setting the control parameters of the corresponding heating or cooling system in accordance with the above-mentioned fixed values should be the starting point for adaptation.



x_d : control difference $x_d = x_{set} - x_{act}$
P = 1/K : parameterizable proportional band
K = 1/P : gain

PI control algorithm: variable $y = K x_d [1 + (t / T_N)]$; By deactivating the reset time (setting = "0"):

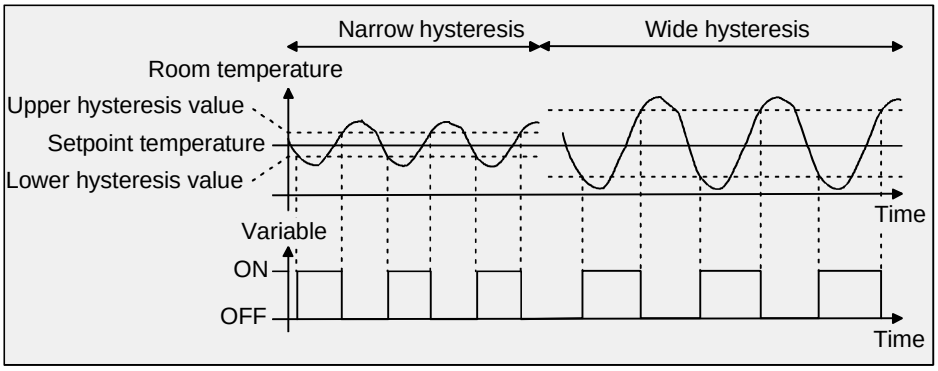
P control algorithm: variable $y = K x_d$

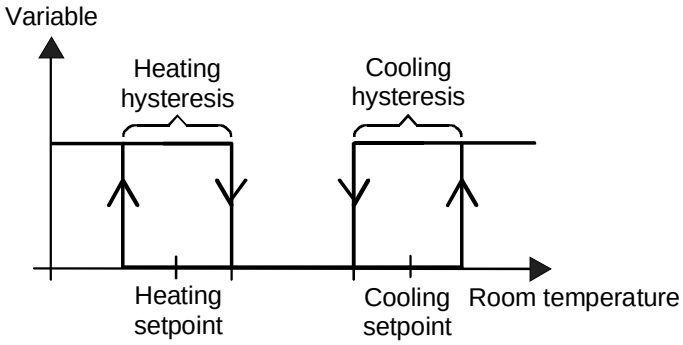
Parameter Setting		**Effect**
P	Small proportional band	High overshooting upon set value changes (possibly even permanent oscillation), quick adjustment on set value.
P	Wide proportional band	No (or low) overshooting, but slowly adjusting.
T_N	Short reset time	Quick compensation of deviations (ambient conditions), risk of permanent oscillation.
T_N	Long reset time	Slow compensation of deviations.

4.3.2.2 Adapting the 2-point control

The 2-point control represents a very simple type of temperature control. For this type of control, two hysteresis temperature values are preset.
You can use parameters to set the upper and lower temperature hysteresis limits. Please take account of that:

- a narrow hysteresis will lead to low temperature fluctuations but to a higher bus load.
- although a wide hysteresis will switch less frequently, it will cause more inconvenient temperature fluctuations.





4.3.3 Output of variables

4.3.3.1 Variables objects

Depending on the control algorithm selected for the heating and/or cooling mode and, if necessary, also for the additional stages, the format of the variables objects is set. Therefore, you can create 1 bit or 1 byte variables objects. The control algorithm calculates the variables at time intervals of 30 seconds and outputs them. For pulse-width-modulated PI control (PWM), the variable is exclusively updated at the end of a time cycle, if required.

Possible object data formats for the variables, separately for both operating modes, for the basic and additional stages, or for both control circuits, are…

- continuous PI control: 1 byte type,
- switching PI control: 1 bit type + additionally 1 byte (e. g. for status indication in connection with visualization),
- switching 2-point control: 1 bit type.

Depending on the operating mode selected, the regulator can trigger heating and/or cooling systems and determine variables and output them via separate objects. In the *"heating and cooling"* mixed mode, distinction is made between two different options:

Option 1: The heating and cooling systems are two separate systems.
In this case, you should set the *"send variable heating and cooling to one common object"* parameter in the *"room temperature regulator functions"* parameter branch to *"no"* (default). Thus, separate objects per variable will be available through which the individual systems can be triggered separately. This setting will enable you to define separate types of control for heating or cooling.

Option 2: The heating and cooling systems are one combined system.
In this case, you can set the *"send variable heating and cooling to one common object"* parameter in the *"room temperature regulator functions"* parameter branch to *"yes"*, if required. Thus, the variables for heating and cooling will be sent to the same object. For two-stage control, another common object will be enabled for the heating and cooling additional stages.
If you use this setting you can only define the same type of control for heating and cooling, as the type of control and the data format must be identical in this case. The control parameters (*"type of heating/ cooling"*) must still be parameterized separately for heating or cooling operation.
A combined variables object can, for example, become necessary if both heating and cooling should be effected through a one-pipe system (combined heating and cooling system). For this purpose, the temperature of the medium in the one-pipe system must, first of all, be changed by the control system of the installation. Subsequently, you can select the heating/cooling switch-over through the object (often, cold water is used in the one-pipe system for cooling in summer, while hot water is used for heating in winter).

Note:

Heating and cooling at the same time (variables > "0") is, in general, not possible.

If required, you can invert the variable prior to transmission. By the *"output of the heating variable"* or *"output of the cooling variable"* parameter, or in case of output through a combined *"variable output"* object, the variable value will be output in inverted form in accordance with the object data format. When two-stage control is used the parameters for inverting the additional stage(s) will also be available.
Where:

for continuous variables:	not inverted:	variable 0 % ... 100 %,	value 0 .. 255,
	inverted:	variable 0 % ... 100 %,	value 255 ... 0,
for switching variables:	non-inverted:	variable OFF / ON,	value 0 / 1,
	inverted:	variable OFF / ON,	value 1 / 0.

4.3.3.2 Automatic transmission

- Continuous PI control:
  For continuous PI control, the room temperature regulator cyclically calculates a new variable every 30 seconds and transmits it to the bus through a 1 byte value object. In this connection, you can use the *"automatic transmission at modification by…"* parameter in the *"room temperature function / variable and status output"* parameter branch to set the variable change interval after which a new variable should be sent to the bus. You can parameterize the change interval to "0" so that no automatic transmission will take place whenever the variable changes.

  In addition to the output of the variable upon a change of the latter, the current variable value can be cyclically sent to the bus. In this connection, further variable telegrams will be released after a parameterized cycle time in addition to the expected times of change, according to the active value.
  This is to ensure that, if the variable undergoes cyclic safety monitoring, telegrams will be received in the actuator or in the triggered switching actuator within the monitoring time. The time interval set by the *"cycle time for automatic transmission…"* parameter should correspond to the monitoring time in the actuator (prefer to parameterize a shorter cycle time in the regulator).
  Setting "0" will deactivate the cyclic transmission of the variable.

  Please note for continuous PI control that no more variable telegrams will be sent upon a change if you have deactivated cyclic transmission and disabled automatic transmission

- Switching PI control (PWM):
  For switching PI control (PWM), the room temperature regulator will also internally calculate a new variable every 30 seconds. However, updating the variable with this type of control will only take place at the end of a time cycle, if required. The *"automatic transmission at modification by…"* and *"cycle time for automatic transmission…"* parameters will have no function in connection with this control algorithm.

- 2-point control:
  For 2-point control, the room temperature and thus the hysteresis values will be evaluated cyclically every 30 seconds so that the variable, if required, will only change at these intervals. As no continuous variables will be calculated for this control algorithm the *"automatic transmission at modification by…"* parameter will show no effect in connection with this control algorithm.

  In addition to the output of the variable upon a change of the latter, the current variable value can be cyclically sent to the bus. In this connection, further variable telegrams will be released after a parameterized cycle time in addition to the expected times of change, according to the active value.
  This is to ensure that, if the variable undergoes cyclic safety monitoring, telegrams will be received in the actuator or in the triggered switching actuator within the monitoring time. The time interval set by the *"cycle time for automatic transmission…"* parameter should correspond to the monitoring time in the actuator (prefer to parameterize a shorter cycle time in the regulator).
  Setting "0" will deactivate the cyclic transmission of the variable.

4.4 Temperature set values

4.4.1 Set value presettings in the ETS

You can preset temperature set values for each operating mode. In the ETS plug-in, you can parameterize the set values for the "comfort", "standby" and "night" modes. If desired, you can adapt the set value temperatures later during running operation by local actuation of the regulator in the programming mode or by control via objects.
For the "frost/heat protection" mode, you can separately parameterize two temperature set values only in the ETS, one for heating (frost protection) and one for cooling operation (heat protection).

When presetting the set value temperatures for the comfort, standby and night modes, always make sure that all set values have a fixed relation with one another, for all values are derived from the basic temperature (basic set value). The *"basic temperature after reset"* parameter in the *"set point values"* parameter branch sets the basic set value which will be loaded as presetting when the device is being programmed by the ETS.
From this value, the temperature set values for the standby and night modes will be derived, taking account of the *"lower/raise the setpoint temperature in the standby mode"* or *"lower/raise the setpoint temperature in the night mode"* parameter, depending on the heating or cooling mode. For the *"heating and cooling"* mode, the dead band will also be taken into account.

In two-stage control operation, all set value temperatures of the additional stage will be derived from the set value temperatures of the basic stage. In this connection, to determine the set value temperatures of the additional stage the *"difference between basic and additional stages"* (step width) firmly parameterized in the ETS plug-in will be deducted from the set values of the basic stage for heating operation or added to them for cooling operation. If the temperature set values of the basic stage are, for example, changed on the push button in the programming mode or by presetting a new basic set value, the set value temperatures of the additional stage will automatically change indirectly at the same time. At a set value difference of "0", both stages will be heating or cooling at the same time, using the same variable.

If you use two control circuits you can preset common set values or, alternatively, separate values for both control circuits. In this connection, the *"own setpoints for 2nd control circuit"* parameter in the *"room temperature regulator function / set point values"* parameter branch will preset the set values:

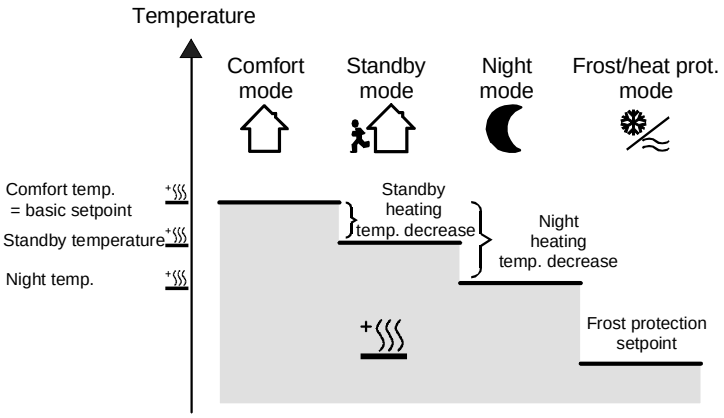
- Setting *"no"* (default):
  Both control circuits have the same set values for the comfort, standby and night modes. The frost and heat protection temperatures are also identical. If enabled, this setting has one common object for presetting the basic set value and one object for the transmission of the set value temperature to the bus.

- Setting *"yes"*:
  Both control circuits have their own separate set values for the comfort, standby and night modes. Only the frost and heat protection temperatures are identical. If enabled, this setting has separate objects per control circuit for presetting the basic set value or for the transmission of the set value temperature to the bus, respectively. You can only change the set value temperature for the first control circuit on the push button in the programming mode.

The operating mode switch-over of the second control circuit always proceeds in parallel with the first control circuit switch-over. If you use two control circuits two-stage control and mixed heating and cooling operation will not be possible.

Depending on the heating/cooling switch-over, please note the relations shown on the next pages when presetting the set values.
If you use two control circuits you can set the heating/cooling switch-over for both control circuits either to *"heating"* or to *"cooling"*. Mixed *"heating and cooling"* operation will not be possible in this connection.

4.4.1.1 *"Heating"* mode set values:



(Recommended set value temperature value presettings)

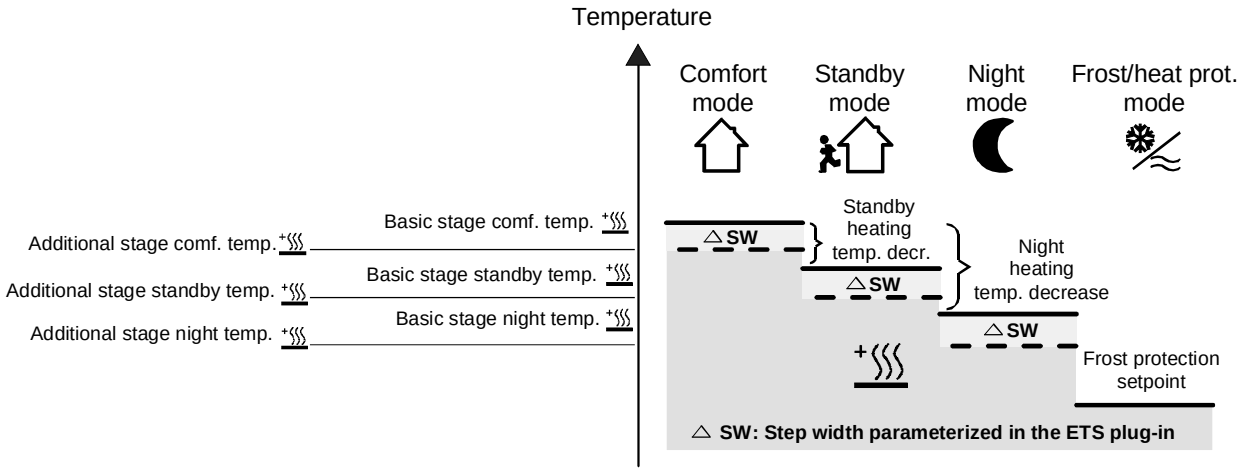
In this mode, the comfort, standby and night set value temperatures are given, and you can preset the frost protection temperature. Where:

$T_{standby\ heating\ set\ value} \leq T_{comfort\ heating\ set\ value}$ or $T_{night\ heating\ set\ value} \leq T_{comfort\ heating\ set\ value}$

The standby and night set value temperatures are derived from the comfort set value temperature (basic set value) in accordance with the parameterized decrease temperatures. On the regulator, you can locally set even other decrease temperatures in the programming mode, if enabled, by changing the night and standby set value temperature values (refer to "1.4 Programming mode/local operation", page 25). Such local operation will only be possible for control circuit 1.

Frost protection is intended to prevent the heating system from freezing. For this reason, you should set the frost protection temperature lower than the night temperature (default: +7 °C).However, you can, in general, select values between +7 °C and +40 °C as frost protection temperature.
For "heating", the possible value range is between +7.0 °C and +99.9 °C and is restricted by the frost protection temperature in the lower range.

For two-stage heating operation, the difference between the basic and additional stages parameterized in the ETS plug-in will also be taken into account.

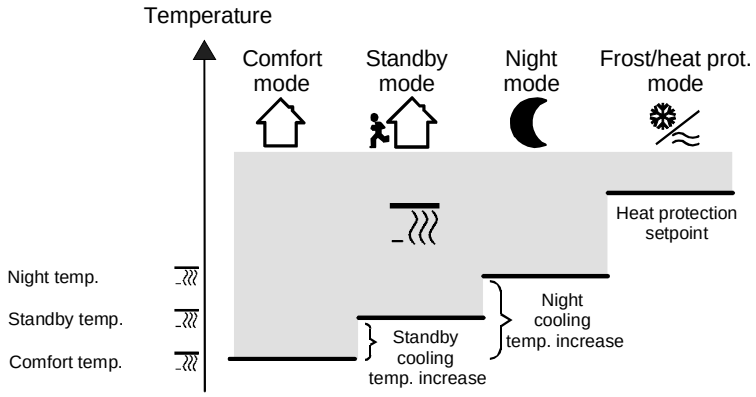


$T_{comfort\ heating\ add.\ st.\ setpt.} \leq T_{comfort\ heating\ basic\ st.\ setpt.}$ / $T_{standby\ heating\ add.\ st.\ setpt.} \leq T_{standby\ heating\ basic\ st.\ setpt.}$
$T_{standby\ heating\ set\ value} \leq T_{night\ heating\ set\ value}$

or

$T_{comfort\ heating\ add.\ st.\ setpt.} \leq T_{comfort\ heating\ basic\ st.\ setpt.}$ / $T_{night\ heating\ add.\ st.\ setpt.} \leq T_{night\ heating\ basic\ st.\ setpt.}$
$T_{night\ heating\ set\ value} \leq T_{comfort\ heating\ set\ value}$

4.4.1.2 *"Cooling"* mode set values:



(Recommended set value temperature value presettings)

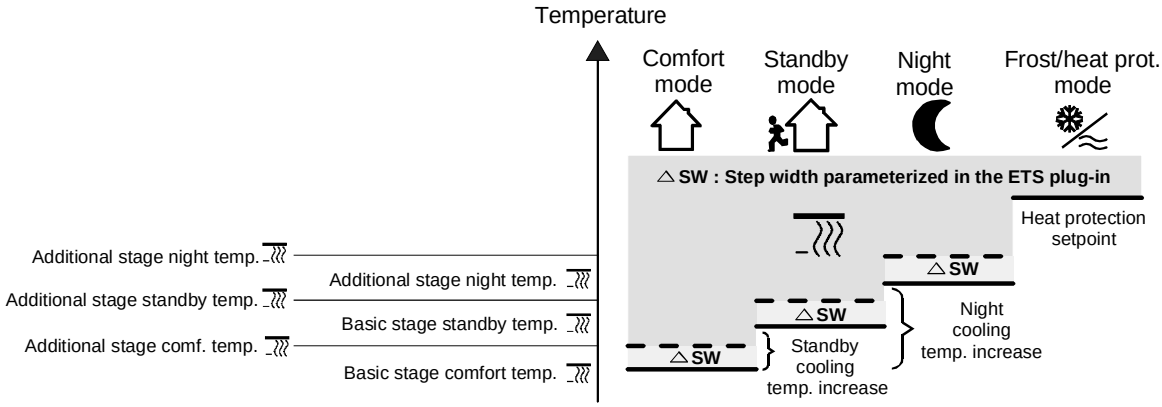
In this mode, the comfort, standby and night set value temperatures are given, and you can preset the heat protection temperature. Where:

$T_{comfort\ cooling\ set\ value} \leq T_{standby\ cooling\ set\ value}$ or $T_{comfort\ cooling\ set\ value} \leq T_{night\ cooling\ set\ value}$

The standby and night set value temperatures are derived from the comfort set value temperature (basic set value) in accordance with the parameterized increase temperatures. On the regulator, you can locally set even other increase temperatures in the programming mode, if enabled, by changing the night and standby set value temperature values (refer to "1.4 Programming mode/local operation", page 25). Such local operation will only be possible for control circuit 1.

Heat protection is intended to prevent a maximum permissible room temperature from being exceeded in order to protect parts of the installation, if necessary. For this reason, you should set the heat protection temperature higher than the night temperature (default: +35 °C). However, you can, in general, select values between +7 °C and +45 °C as heat protection temperature.
For "cooling", the possible value range is between -99.9 °C and +45.0 °C and is restricted by the heat protection temperature in the upper range.

For two-stage cooling operation, the difference between the basic and additional stages parameterized in the ETS plug-in will also be taken into account.

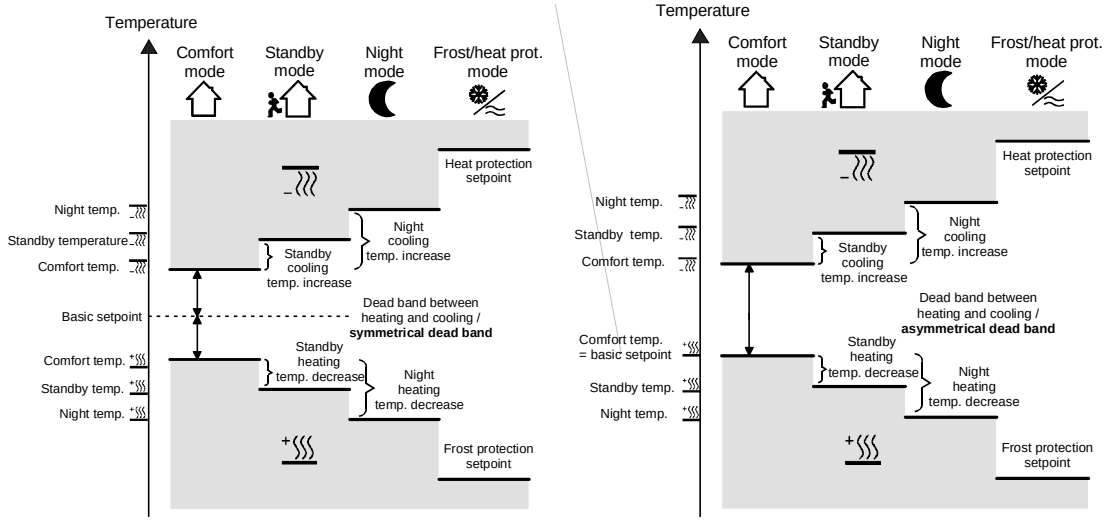


$T_{comfort\ cooling\ basic.\ st.\ setpt.} \leq T_{comfort\ cooling\ add.\ st.\ setpt.}$ / $T_{standby\ cooling\ basic\ .\ st.\ setpt.} \leq T_{standby\ cooling\ add.\ st.\ setpt.}$
$T_{comfort\ cooling\ set\ value} \leq T_{standby\ cooling\ set\ value}$

or

$T_{comfort\ cooling\ basic.\ st.\ setpt.} \leq T_{comfort\ cooling\ add.\ st.\ setpt.}$ / $T_{night\ cooling\ basic\ .\ st.\ setpt.} \leq T_{night\ cooling\ add.\ st.\ setpt.}$
$T_{comfort\ cooling\ set\ value} \leq T_{night\ cooling\ set\ value}$

4.4.1.3 *"Heating and Cooling"* mode set values:



(Recommended set value temperature value presettings)

In this mode, the comfort, standby and night set value temperatures of both operating modes and of the dead band are given. In addition, you can preset the frost and heat protection temperatures. Where:

$$T_{\text{standby heating set value}} \leq T_{\text{comfort heating set value}} \leq T_{\text{comfort cooling set value}} \leq T_{\text{standby cooling set value}}$$

or

$$T_{\text{night heating set value}} \leq T_{\text{comfort heating set value}} \leq T_{\text{comfort cooling set value}} \leq T_{\text{night cooling set value}}$$

The standby and night set value temperatures are derived from the comfort heating or comfort cooling set value temperatures. In this connection, you can assign the temperature increase (for cooling) and the temperature decrease (for heating) of both operating modes in the ETS plug-in. The comfort temperatures themselves are derived from the dead band and the basic set value (refer to "Dead band" on the next page).

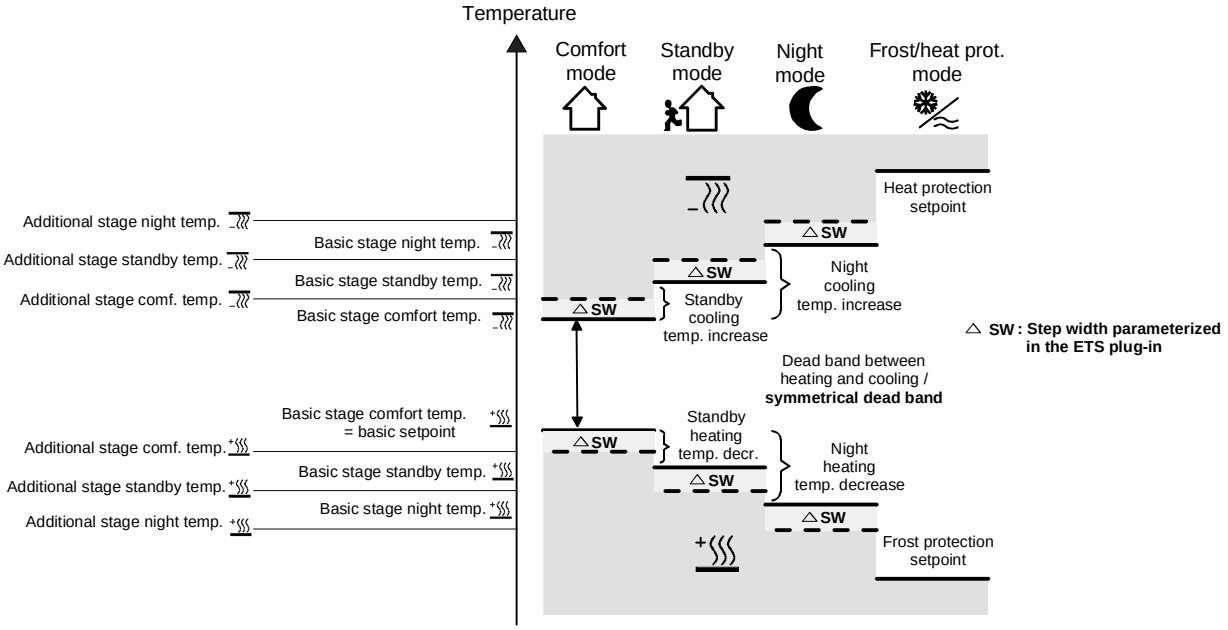
Note:

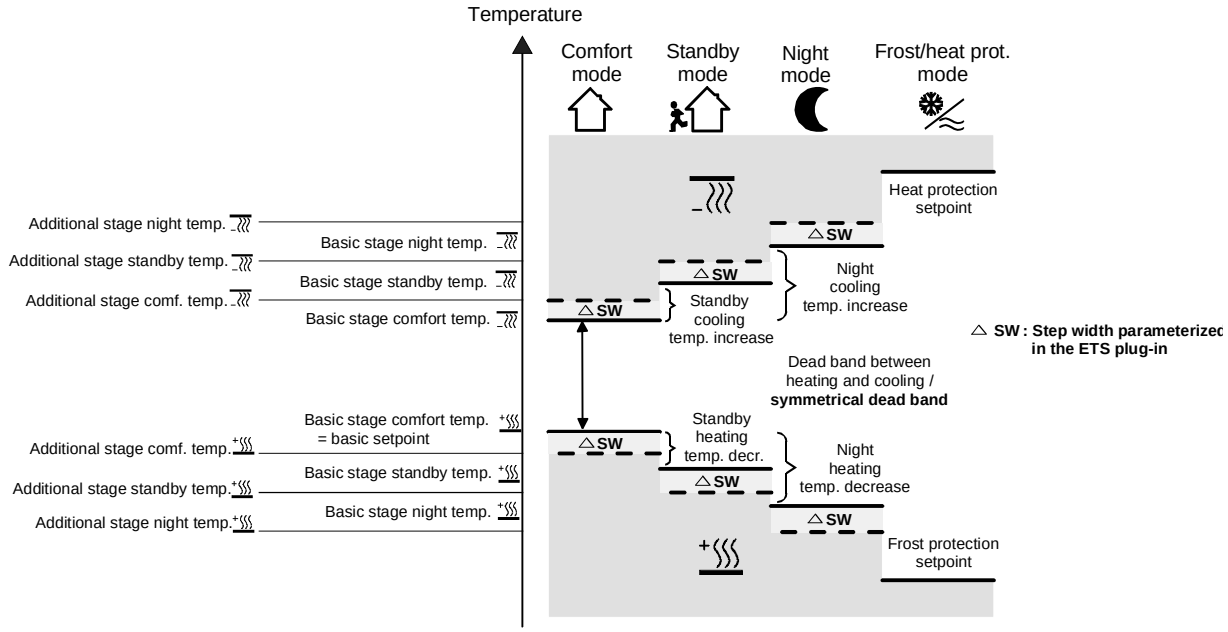
On the regulator, you can locally set even other increase or decrease temperatures in the programming mode, if enabled, by changing the night and standby set value temperature values (refer to "1.4 Programming mode/local operation", page 25). Such local operation will only be possible for control circuit 1.

Frost protection is intended to prevent the heating system from freezing. For this reason, you should set the frost protection temperature lower than the night heating temperature (default: +7 °C).However, you can, in general, select values between +7 °C and +40 °C as frost protection temperature.
Heat protection is intended to prevent a maximum permissible room temperature from being exceeded in order to protect parts of the installation, if necessary. For this reason, you should set the heat protection temperature higher than the night cooling temperature (default: +35 °C). However, you can, in general, select values between +7 °C and +45 °C as heat protection temperature.

For "heating and cooling", the possible value range of a set value temperature is between +7 °C and +45.0 °C and is restricted by the frost protection temperature in the lower range and by the heat protection temperature in the upper range.

For two-stage heating or cooling operation, the difference between the basic and additional stages parameterized in the ETS plug-in will also be taken account of.





$T_{comfort\ heating\ add.\ st.\ setpt.} \leq T_{comfort\ heating\ basic\ st.\ setpt.} \leq T_{comfort\ cooling\ basic.\ st.\ setpt.} \leq T_{comfort\ cooling\ add.\ st.\ setpt.}$

$T_{standby\ heating\ add.\ st.\ setpt.} \leq T_{standby\ heating\ basic\ st.\ setpt.} \leq T_{standby\ cooling\ basic.\ st.\ setpt.} \leq T_{standby\ cooling\ add.\ st.\ setpt.}$

$T_{standby\ heating\ set\ value} \leq T_{comfort\ heating\ set\ value} \leq T_{comfort\ cooling\ set\ value} \leq T_{standby\ cooling\ set\ value}$

or

$T_{comfort\ heating\ add.\ st.\ setpt.} \leq T_{comfort\ heating\ basic\ st.\ setpt.} \leq T_{comfort\ cooling\ basic.\ st.\ setpt.} \leq T_{comfort\ cooling\ add.\ st.\ setpt.}$

$T_{night\ heating\ add.\ st.\ setpt.} \leq T_{night\ heating\ basic\ st.\ setpt.} \leq T_{night\ cooling\ basic.\ st.\ setpt.} \leq T_{night\ cooling\ add.\ st.\ setpt.}$

$T_{night\ heating\ set\ value} \leq T_{comfort\ heating\ set\ value} \leq T_{comfort\ cooling\ set\ value} \leq T_{night\ cooling\ set\ value}$

Dead band:

The comfort heating and cooling set value temperatures are derived from the basic set value, taking account of the dead band preset. The dead band (temperature zone where neither heating nor cooling takes place) is the difference between the comfort set value temperatures.
In the ETS plug-in, you can preset the *"dead band between heating and cooling"*, *"dead band position"* and *"basic temperature after reset"* parameters. In this connection, distinction is made among the following settings:

Dead band position = *"symmetrical"* (default):
The dead band preset in the ETS plug-in is divided into two parts at the basic set value. From the resulting half of the dead band, the comfort set value temperatures are directly derived from the basic set value. Where:

Tbasic set value $- \frac{1}{2}T_{\text{dead band}}$ = Tcomfort heating set value or Tbasic set value + ½Tdead band = Tcomfort cooling set value

➔ $T_{\text{comfort cooling setpt.}} - T_{\text{comfort heating setpt.}} = T_{\text{dead band}}$; $T_{\text{comfort cooling setpt.}} \geq T_{\text{comfort heating setpt.}}$

Important for a symmetrical dead band:

- For a symmetrical dead band, local operation in the programming mode will indirectly set the basic set value through the comfort heating temperature. For this reason, the basic set value will not be shown in the display.
- By changing the comfort cooling set value temperature, you can alter the dead band through local operation, if enabled (dead band shift). When you alter the dead zone you can expect a shifting of the comfort heating set value temperature and thus of all the other temperature set values if the dead band position is symmetrical. You can set the dead band to 0 °C (result: $T_{\text{comfort cooling set value}} = T_{\text{comfort heating set value}}$). In such case, neither heating nor cooling will take place if the room temperature determined is equal to the comfort set value temperatures.

Dead band position = *"asymmetrical"*:
For this setting, the comfort heating set value temperature is equal to the basic set value.
The dead band preset in the ETS plug-in will only become effective from the basic set value towards the comfort cooling temperature. Thus, the comfort mode cooling set value temperature is directly derived from the comfort heating set value. Where:

Tbasic set value = Tcomfort heating set value ➔ Tbasic set value + Tdead band = Tcomfort cooling set value

➔ $T_{\text{comfort cooling setpt.}} - T_{\text{comfort heating setpt.}} = T_{\text{dead band}}$; $T_{\text{comfort cooling setpt.}} \geq T_{\text{comfort heating setpt.}}$

Important for an asymmetrical dead band:

- By changing the comfort cooling set value temperature, you can alter the dead band through local operation, if enabled (dead band shift). For an asymmetrical dead band position, only the cooling temperature set values will be changed if you change the comfort cooling set value temperature. You can set the dead band to 0 °C (result: $T_{\text{comfort cooling set value}} = T_{\text{comfort heating set value}}$). In such case, neither heating nor cooling will take place if the room temperature determined is equal to the comfort set value temperatures.

4.4.2 Changing the set values

4.4.2.1 Changing the basic temperature and the comfort, standby and night mode set value temperatures

When presetting the set value temperatures for the comfort, standby and night modes, always make sure that all set values have a fixed relation with one another, for all values are derived from the basic temperature (basic set value). The *"basic temperature after reset"* parameter in the *"set point values"* parameter branch sets the basic set value which will be loaded as presetting when the device is being programmed by the ETS.

By local operation on the regulator in the programming mode, or controlled by the *"basic set value"* object, it is possible to 'subsequently' change or reset the set value temperatures. After programming, the set value temperatures of the second control circuit can only be adapted in an object-controlled way.

Any change must always be enabled in the ETS plug-in in the *"set point values"* parameter branch. In this connection, it is possible…

- to allow the *"modification of the basic temperature setpoint value"* by directly changing the comfort heating temperature on the device (locally; only for control circuit 1) and/or by setting a new basic set value via the bus (object 26 for control circuit 1/object 27 for control circuit 2).

- to facilitate the *"standby temperature change"* by directly changing the standby heating or cooling temperatures of the first control circuit only on the device (locally).

- to allow the *"night temperature change"* by directly changing the night heating or cooling temperatures of the first control circuit only on the device (locally), and

- to enable the *"dead band shift"* by changing the comfort cooling set value temperature only on the device (locally; control circuit 1 for "heating and cooling").

- to prevent *"modification of the setpoints 'cooling'"* for mixed operation on the device in the programming mode.

If changing is not enabled (setting: *"deactivated"*), 'subsequent' resetting of the value preset by the ETS and local setting of the corresponding temperatures will not be possible.
In case of an unadmitted basic set value resetting via the bus, object 26 or 27 will be hidden.

Changing the basic set value/the comfort heating temperature:

Only when changing the basic set value (by local operation and/or though the object) you can distinguish between two options:

- Option 1: The basic set value change will be accepted permanently.
- Option 2: The basic set value change will be accepted only temporarily (default).

In this connection, you can use the *"accept modification of the basic temperature setpoint value permanetly"* parameter in the *"room temperature regulator function/set point values"* parameter branch to determine whether you want to permanently (select *"yes"*) or only temporarily (select *"no"*) save the basic temperature value change.

Option 1:

If you change the basic temperature set value of the first or second control circuit it will be permanently saved in the EEPROM of the push button. In this connection, the newly set value will overwrite the basic set value temperature originally parameterized by the ETS. This is the only way to keep the basic set value change, even after changing the operating mode or after a reset.

Important:

- Frequent changing of the basic temperature (e. g. several times a day) can adversely affect the life of the device as the non-volatile memory (EEPROM) used has been designed for less frequent write access events only.
- Any value assigned by local operation will not be taken over into object 26.
- After bus voltage recovery, the saved basic set value will still remain active. However, the value of object 26 or 27 will be "0". You can only read out the current basic set value (set the "R" flag) after an external object update.

Option 2:

The basic set value selected on the push button or received through the object will remain stored in the currently set operating mode only in temporary form. In case of bus voltage failure or after you have changed the operating mode (e. g. from comfort to standby), the basic set value specified by local operation or received through the object will be discarded and replaced by the value originally parameterized in the ETS.

Changing the set values for the standby and night modes as well as for the dead band (comfort cooling temperature):

As the set value temperatures for the "standby" and "night" modes, or the set values for the "cooling" mode, respectively, are derived from the basic set value temperature, taking account of the decrease, increase or dead band values parameterized in the ETS plug-in, these set value temperatures will also linearly shift by the basic set value change made.

Only by local operation on the regulator in the programming mode, you can, in addition, set temperature values for the standby and night modes as well as for the dead band other than those parameterized in the ETS for the first control circuit. In such case, the originally parameterized decrease, increase and dead-band temperatures will be replaced by the newly resulting values caused by the locally changed temperature set values. In this connection, the temperature set values for the standby and night modes or for comfort "cooling" (dead band) will always be permanently stored in the EEPROM, no matter what the setting of the "*accept modification of the basic temperature setpoint value permanetly"* parameter is.

4.4.2.2 Shift of basic set value

In addition to the fixed presetting of individual temperature set values by the ETS, by local operation in the programming mode, or by the basic set value object, it is possible to allow the user to shift the basic set value within a selectable range.
Thus, in normal operation (no programming mode active), you can actuate one of the display buttons to show in the display window the set value temperature of the activated operating mode of the first control circuit.

Furthermore, you can press the right/left display button to shift upward/downward the displayed set value temperature in increments/decrements of 0.01 °C.
Please note that such shift of the displayed set value temperature (basic temperature offset) will directly act on the basic set value, thus also shifting all the other temperature set values.
If you use two control circuits with separate set values the set value temperatures of both control circuits will be shifted.

Whether a shift of basic set value will only act on the temporarily activated operating mode or on all the other set value temperatures of the remaining operating modes can be preset by the *"accept modification of shift of basic setpoint value permanetly"* parameter in the *"set point values"* parameter branch.

Setting *"no"* (default):
The Shift of basic set value made will only be active until the operating mode is changed. After a change, the set value shift will be reset to "0".

Setting *"yes"*:
The Shift of basic set value made will, in general, act on all operating modes. The shift will be maintained even after a switch-over.

You can use the *"upward / downward adjustment of basic setpoint temperature"* parameter to define the temperature range selectable for a shift of basic set value. In this connection, you can shift the current set value by a maximum of +/- 10 °C or +/- 10 K, respectively.

Notes on the shift of basic set value:

- As the value for the Shift of basic set value is only saved in a volatile memory (RAM) the shift will get lost after a reset (e. g. bus voltage failure).
- A set value shift will have no effect on the frost or heat protection temperature set values.
- If regulator operation is disabled actuation of the left or right display button to shift the basic set value will show no response.

4.4.3 Transmitting the set value temperature

The set value temperature specified by the active operating mode or, if necessary, readjusted subsequently can be actively transmitted to the bus via object 50 or, if both control circuits are used with separate set values, additionally via object 51, *"setpoint temperature"*.
The *"transmission at setpoint temperature modification by…"* parameter in the *"room temperature regulator functions / set point values"* parameter branch sets the temperature value by which the set value must change until the set value temperature value is automatically transmitted via the object. In this connection, temperature value changes between 0.1 °C and 25.5 °C or 0.1 K and 25.5 K are possible. Setting "0" in this case, will deactivate the automatic transmission of the set value temperature.
In addition, the set value can be transmitted cyclically. The *"cyclical transmission of setpoint temperature"* parameter sets the cycle time (1 to 255 minutes). The value "0" (default) will deactivate the cyclical transmission of setpoint temperature values.
Please note that no more telegrams regarding the set value temperature will be sent if cyclic transmission and automatic transmission upon a change have been deactivated.

You can set the "R" flag on the "set value temperature" object to read out the current set value. After bus voltage recovery, re-programming by the ETS, or after re-plugging the application module, the object value will be updated in accordance with the current set value temperature value and actively transmitted to the bus.

4.5 Room temperature measuring

The room temperature regulator measures the actual temperature and compares it with the preselected set value temperature. With the aid of the selected control algorithm, the variable is then calculated from the difference between the actual and the set value temperature.
In order to always ensure error-free and efficient room temperature control it is of utmost importance that the actual temperature is measured accurately.

The B.IQ push button RTR has an integrated temperature sensor. Alternatively (e. g. if the push button has been installed at an unfavourable place, or under difficult application conditions such as in damp rooms) or additionally (e. g. in large rooms or halls), you can externally connect a second EIB temperature sensor via the bus and use it for actual-value measurements if you only have one control circuit.
If you use both control circuits, the actual temperature value of the second circuit will be determined by the external sensor. In this connection, the actual-temperature measurement of the first control circuit can exclusively be made by the internal sensor.

When choosing the place where you want to install the B.IQ push button RTR or the external sensor, please take account of the following points:

- Integration of the push button into multiple combinations, particularly if flush-mounted dimmers are involved, should be avoided.
- Do not install the sensors in the vicinity of major electric consumers (thermal radiation).
- Do not install the devices in the vicinity of radiators or cooling systems.
- Avoid direct sun radiation onto the temperature sensors.
- The installation of sensors on the inside of an outer wall can adversely affect the temperature measurement.
- Install temperature sensors at least 30 cm away from doors or windows and at least 1.5 m above the floor.

4.5.1 Sensing the temperature and creating measured values

Temperature detection in a control circuit depends upon the parameterization. If you use both control circuits, the actual temperature value of the second circuit will be determined by the external sensor.

One control circuit:
In only one control circuit, the *"temperature detection"* parameter in the *"room temperature regulator function / room temperature measuring"* parameter branch specifies by which sensors you want to determine the actual temperature. In this connection, the following settings will be possible:

- *"Internal sensor"*:
  The temperature sensor integrated in the B.IQ push button RTR is activated. Thus, the actual temperature value is determined only locally in the device.
  This parameterization will initiate control immediately after a reset.

- *"External sensor"*:
  The actual temperature is determined only by the external sensor. The internal sensor is deactivated. In this connection, the external sensor must send the detected temperature value to *"ext. temperature sensor"* 2 byte object 24 (EIS 5) of the B.IQ RTR push button. Alternatively or additionally, the push button can request the current temperature value in cycles (set the "R" flag on the external sensor). For this purpose, set the *"scanning time for external sensor…"* parameter to a value of > "0". You can set the scanning interval from 1 minute to 255 minutes. This parameterization will make the room temperature regulator wait for a temperature value telegram from the external sensor after a reset until control starts and a variable is output, if necessary.

- *"Internal and external sensor"*:
  In this case, both the internal and the external temperature sensor is active. In this connection, the external sensor must send the detected temperature value to *"external temperature sensor"* 2 byte object 24 (EIS 5) of the B.IQ RTR push button. Alternatively or additionally, the push button can request the current temperature value in cycles (set the "R" flag on the external sensor). For this purpose, set the *"scanning time for external sensor…"* parameter to a value of > "0". You can set the scanning interval from 1 minute to 255 minutes. This parameterization will make the room temperature regulator wait for a temperature value telegram from the external sensor after a reset until control starts and a variable is output, if necessary.

  This setting results in creating the actual temperature from the two temperature values measured. In this connection, the *"creating of measuring value internal against external"* parameter specifies the weighting of the temperature values. Thus, you can balance the actual temperature measurement in dependence on the various places of installation of the sensors or on a possibly different heat distribution within the room, respectively. Often, temperature sensors which are under negative external influences (for example, unfavourable place of installation due to sun radiation or in the vicinity of a radiator or of a door/window) are weighted less intensively.

  Example:
  B.IQ push button RTR installed next to the room entrance door (internal sensor). An additional external temperature sensor has been installed on an internal wall in the middle of the room below the ceiling.

  Internal sensor: 21.5 °C (internal sensor measuring range: 0 °C to + 40 °C ±1 %)
  External sensor: 22.3 °C
  Creating measured value: 30 % vs. 70 %

  Result: $T_{internal\ result} = T_{internal} \cdot 0.3 = 6.45$ °C, $T_{external\ result} = T_{external} = 22.3$ °C $\cdot 0.7 = 15.61$ °C →

  $T_{actual\ result} = T_{internal\ result} + T_{external\ result} = \underline{22.06\ °C}$

Two control circuits:
The actual temperature measurement is exclusively made by the internal sensor. The external sensor detects the actual temperature of the second control circuit and must, in this connection, send this temperature value to *"external temperature sensor"* 2 byte object 24 (EIS 5) of the B.IQ RTR push button. Alternatively or additionally, the push button can request the current temperature value in cycles (set the "R" flag on the external sensor). For this purpose, set the *"scanning time for external sensor…"* parameter to a value of > "0". You can set the scanning interval from 1 minute to 255 minutes. This parameterization will make the room temperature regulator wait for a temperature value telegram from the external sensor after a reset until the control process of the second circuit starts and a variable is output, if necessary.

4.5.2 Adjusting the measured values

In some cases, it can become necessary to adjust the temperature measurements between the internal and external sensors. Such adjustment will, for example, be necessary if the temperature measured by the sensors is permanently below or above the room temperature actually prevailing in the vicinity of the sensor. In this connection, the actual room temperature should be determined by a reference measurement made with the aid of a calibrated thermometer.

You can use the *"adjustment internal sensor…"* or *"adjustment external sensor…"* parameter in the *"room temperature regulator function / room temperature measuring"* parameter branch to parameterize the positive (temperature increase, factors: 1 … 127) or the negative (temperature decrease, factors: -128 … -1) temperature adjustment in steps of 0.1 °C. Thus, you will only have to make this adjustment once, and it will be the same for all operating statuses.

Important:

- Increase the measured value if the value measured by the sensor is below the actual room temperature. Decrease the measured value if the value measured by the sensor is above the actual room temperature.
- When the measured value is created by means of the internal and external sensors only for one control circuit the adjusted value will be used for the calculation of the actual value.

4.5.3 Transmitting the actual temperature

The measured actual temperature of the first control circuit can be actively transmitted to the bus via object 23, *"actual temperature"*.
The *"transmission at room temperature modification by…"* parameter in the *"room temperature regulator functions / room temperature measuring"* parameter branch sets the temperature value by which the actual value must change until the actual temperature value is automatically transmitted via object 23. In this connection, temperature value changes between 0.1 °C and 25.5 °C or 0.1 K and 25.5 K are possible. Setting "0" in this case, will deactivate the automatic transmission of the actual temperature.
In addition, the actual value can be transmitted cyclically. The *"cyclical transmission of room temperature"* parameter sets the cycle time (1 to 255 minutes). The value "0" (default) will deactivate the cyclic transmission of the actual temperature value.
You can set the "R" flag on the "actual temperature" object to read out the current actual value.
Please note that no more telegrams regarding the actual temperature will be sent if cyclic transmission and automatic transmission upon a change have been deactivated.

After bus voltage recovery, re-programming by the ETS, or after re-plugging the application module, the object value will be updated in accordance with the current actual temperature value and actively transmitted to the bus.
If no temperature value telegram has been received from the external sensor yet when the external sensor is used in one control circuit only the value created by the internal sensor will be sent. If you only use the external sensor the object will carry a value of "0" after a reset. For this reason, the external sensor should always transmit the current value after a reset.

4.6 Room temperature regulator disabling functions

4.6.1 Disabling the controller

At some operating statuses, it can become necessary to disable room temperature control. For example, you can switch off control in the dew-point mode of a cooling system, or during maintenance work done on the heating or cooling system.

The *"switch off controller (dew-point operation)"* parameter in the *"room temperature regulator functions"* parameter branch will enable object 40, *"disabling controller"*, when you set it to *"via object"*. Moreover, you can use setting *"no"* (default) to permanently deactivate the regulator disabling function.
If a "1" telegram is received through the enabled disabling object the room temperature control of both control circuits will be entirely deactivated. In such case, the variables are = "0", and the " " symbol will appear in the display. However, you can still operate the regulator in this case.

In the two-stage heating or cooling mode, you can separately disable the additional stage. The *"additional stage inhibit object"* parameter in the *"room temperature regulator functions"* parameter branch will enable object 41, *"disabling additional stage"*, when you set it to *"yes"*. Moreover, you can use setting *"no"* (default) to permanently deactivate the additional stage disabling function.
If a "1" telegram is received through the enabled disabling object room temperature control by the additional stage will be deactivated. The variable of the additional stage will be "0", with the basic stage incessantly continuing working.

If you use both control circuits you can separately disable the second control circuit. If a "1" telegram is received via disabling object 41, *"disabling 2nd control circuit",* the room temperature control of the second control circuit will be deactivated, the variable of this circuit being "0". In this case, the first control circuit will incessantly continue working.

Disabling will always be cancelled after a reset.

4.6.2 Disabling controller operation

You can disable room temperature regulator local operation (all buttons associated with the room temperature regulator). Activated disabling of operation will be indicated in the display by the " " symbol. Please note that this symbol is not exclusive and will also light up when a push button disabling function has been activated.

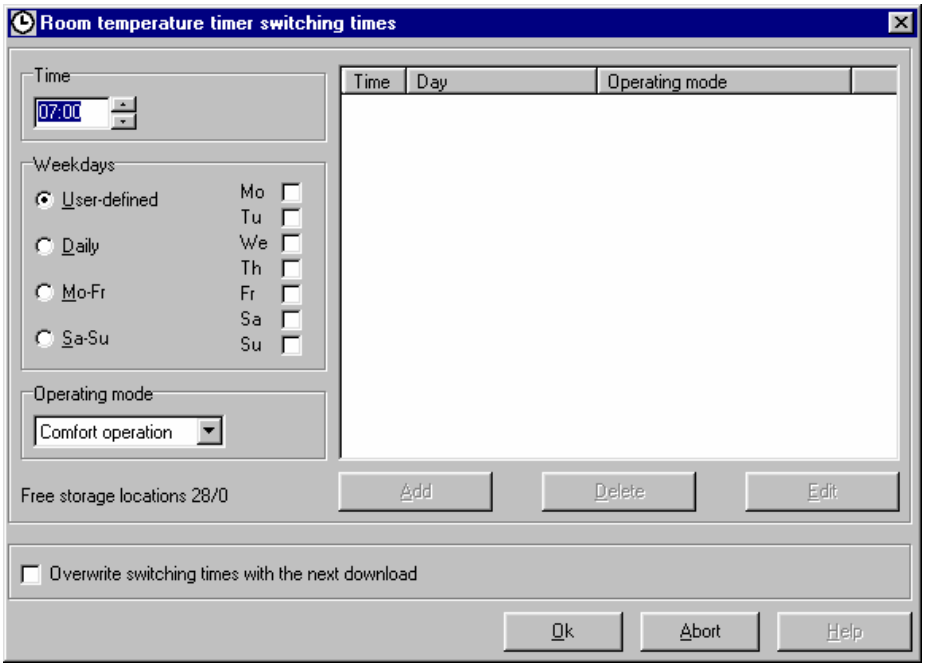
You can use the *"operation of controller inhibitable"* parameter in the *"room temperature regulator functions"* parameter branch to preset whether local operation shall always be impossible (setting: "*always disabled*") or can be disabled by object 39, *"disabling controller operation"* (setting *"via object"*).
If you select *"always disabled"* you cannot parameterize the operating mode switch-over under the button or rocker functions when the push button functionality is involved. In addition, the two display buttons for shifting the basic set value will have no function when you select this setting.
The *"via object"* setting will deactivate local operation when the object receives a "1" telegram. Consequently, local operation will be re-enabled upon the receipt of a "0" telegram. Actuating a button assigned as operating mode switch-over, or actuating the display buttons for shifting the basic set value, will show no response if disabling is active.

Activated disabling of the regulator operation will neither affect the operation of the regulator via the objects nor room temperature control itself, i. e. the control algorithm will work and create variables and status signals.

Regulator operation disabling will always be cancelled after a reset.

4.7 Room temperature timer

The room temperature timer can distinguish up to 28 different switching programs and facilitates the switch-over of the room temperature regulator operating mode, depending on the time and the day of the week. Use the *"room temperature timer = ON"* parameter in the *"room temperature regulator function / room temperature timer"* parameter branch to enable the room temperature timer. Alternatively, you can use the *"room temperature timer = OFF"* setting to permanently disable this function (default).
When the timer function is enabled you can parameterize the switching programs in the ETS plug-in, and they will then be executed in chronological order. The *"room temperature timer"* menu item in the *"timer editor"* menu will call the "room temperature timer switching events" editor:



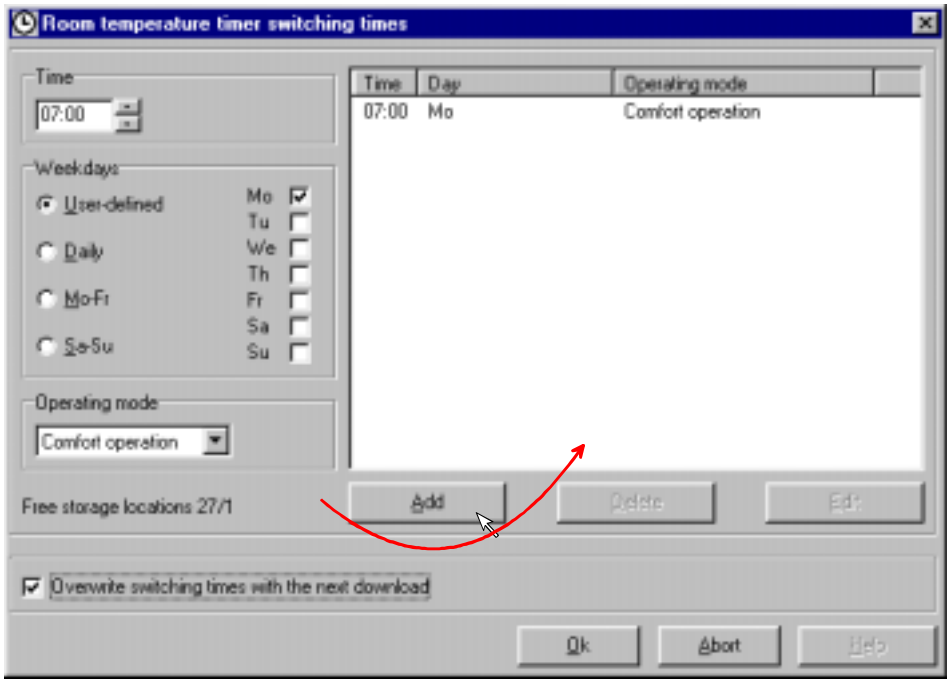
In the left part of the window, you can define the time of the switching program down to the minute. Furthermore, you can define the weekdays on which the switching time event is to take place. You can choose from the "user-defined" (Mo, Tu, …, Su), "daily" (Mo - Su), "Mo – Fr" or "Sa – Su" options.

Each switching program will occupy one memory location in the B.IQ RTR push button. For the "daily", "Mo – Fr" or "Sa – Su" settings, only one memory location will be required. For the "user-defined" option, however, sometimes up to five different switching programs can be created, depending on the parameterized days. The "free storage locations" information parameter in the editor indicates how many memory locations are available (the number in front of the slash) or how many memory locations will have to be occupied by the selected setting (the number behind the slash).

In addition, you must specify the operating mode which you want to activate when a switching program is called. For this purpose, the "comfort", "standby" and "night" operation modes are available.
Please note that an operating mode set by the room temperature timer will have the same priority as a local condition on the push button or caused by the switch-over objects (4 x 1 bit or 1 byte KONNEX switch-over object) and can thus be changed.

The switching times can be configured down to the minute. The switching times are checked by the push button time control also at minute intervals when the timer is active.
The internal clock of the push button should be set by an external time control signal at least every hour to keep the time error as low as possible.
In exceptional cases, there may be major differences between the time followed up in the push button and the time received from the bus so that switching times events will not be executed (skipped switching minute). For this reason, synchronization from the bus should not take place at preset switching times.

After you have defined the switching program (switching time, weekdays, operating mode), you can click on the "add" button to take the program into the program list. This list will appear on the right in the editor window:



Thus, you can preset up to 28 switching programs. You can edit a program created in the right part of the window if you highlight it and click on the "edit" button. Highlighting a program and clicking on the "delete" button will delete the program and remove it from the list.
Click on "OK" to accept the settings into the configuration of the push button.

If you check the "overwrite switching times with the next download" option in the editor window the switching programs of the room temperature timer will be loaded into the device when the entire application is being programmed, or when you are partially programming the parameters.

In addition to presetting the timer programs in the ETS plug-in, you can locally edit on the push button in the programming mode the switching times, the weekdays and the operating modes even after programming. A precondition to do so is that "complete operation" through the display buttons has been enabled (depending on the parameter).
To avoid overwriting of the locally changed data by a subsequent programming process by the ETS you can uncheck the "overwrite switching times with the next download" option.

If it is enabled, you can activate or deactivate the room temperature timer by local operation in the programming mode (refer to "1.4 Programming mode/local operation", page 25) and/or by push button actuation (push button function).
When this function has been activated the " ⏲ " symbol will light up in the display, and the switching programs will be executed in chronological order in accordance with the parameterized switching times. Please note that this " ⏲ " symbol is not exclusive and will also light up when a timer 1 or 2 has been activated.

Info: If the room temperature timer is being activated exactly at the moment of a parameterized switching time the switching program concerned will be executed afterwards.

After switching programs have been fed into the device, the room temperature timer will be activated immediately after the initializing phase, and the programs will be executed. A valid time and a valid weekday received are a prerequisite to this. If no programs have been stored in the device and the function itself has been enabled no switching programs will be executed, although the symbol will be lit after a reset.

An operating mode switch-over by the room temperature timer can additionally and temporarily be suppressed via a separate disabling object. To facilitate the disabling function set the *"lock room temperature timer via object"* parameter in the *"room temperature regulator function / room temperature timer"* parameter branch to *"yes"*. In this case, disabling object 58, *"disabling room temperature timer"*, will be enabled. You can parameterize its polarity.
When the room temperature timer has been disabled through the bus, the " ⏲ " symbol will disappear from the display. The symbol will not go out if other timers have been activated (timer 1 or 2).
During an active disabling function, the operating mode will not be changed by the room temperature timer. If you re-enable the room temperature timer exactly at the moment of a parameterized switching time the switching program concerned will be executed afterwards.
Activations or deactivations of the room temperature timer during the disabling phase will be stored and followed up after the end of disabling.

Important notes on the room temperature timer:

- The room temperature timer will be automatically deactivated when the frost/heat protection mode is being activated. This will avoid unintentional operating mode switch-over and thus unintentional heating or cooling of the room when persons are absent for a longer time, for example.
  You can switch over into the frost/heat protection mode directly on the device (e. g. in the programming mode or as push button operation) or through the switch-over objects (4 x 1 bit or KONNEX). Activation of the frost/heat protection mode by the window status or by automatic frost protection will not deactivate the room temperature timer.
  Also, when an operating mode is preset by the KONNEX override object the room temperature timer will be deactivated until it is re-enabled (value "0").
  An automatically deactivated room temperature timer can be re-activated at any time by push button operation (room temperature timer operation). Automatic re-activation will take place when frost/heat protection is being terminated.

- The internal clock of the push button should be set by an external time control signal at least every hour to keep the time error as low as possible. Unless the internal clock has been updated through the bus at least once per day (update check at 4:00 a. m.) the display of the push button will read "--:--" if the time is indicated in the display (depending on the corresponding parameter). However, the internal clock will keep running with the expected time error, and the switching programs of the room temperature timer will still be executed.

- The weekday information is derived from the time signal. The room temperature timer will only execute the given switching programs after it has received a valid weekday. In normal operation, the weekdays will not be displayed on the push button.

- Receiving a date is not required for the function of the room temperature timer.

4.8 Temperature alarm

The B.IQ push button RTR can monitor two different temperature values, as required. When the upper or lower limits of these parameterized values are exceeded switching telegrams, for example as alarm values, can be sent to the bus.

You can use the *"send temperature alarm via object = yes"* parameter in the *"room temperature regulator function / room temperature measuring"* parameter branch to activate temperature monitoring. Alternatively, setting *"no"* (default) will deactivate temperature monitoring.
When this function is active objects 69 and 70, *"temperature alarm 1"* for the lower temperature value and *"temperature alarm 2"* for the upper temperature value will appear.

When the value falls below the lower temperature limit ($T_{actual} < T_{lower\ limit}$) a "1" telegram will be sent to the bus via the *"temperature alarm 1"* object. As soon as the room temperature reaches the lower limit or exceeds it ($T_{actual} \geq T_{lower\ limit}$) a "0" telegram will be sent to the bus via the *"temperature alarm 1"* object.
When the upper temperature limit ($T_{actual} > T_{upper\ limit}$) is exceeded a "1" telegram will be sent to the bus via the *"temperature alarm 2"* object. As soon as the room temperature reaches the upper limit or falls below it ($T_{actual} \leq T_{upper\ limit}$) a "0" telegram will be sent to the bus via the *"temperature alarm 2"* object.

Every minute, the temperature values are compared with the actual temperature (room temperature) determined. Accordingly, a maximum of one temperature alarm telegram per minute will be transmitted. In addition, the telegrams will only be transmitted when the switching value changes.
Please note that temperature monitoring will only work as soon as a valid actual value is available. So, when an external temperature sensor is used a valid telegram must be received first.
The lower temperature value must be under the upper limit. If you parameterize the values in a way other than described the ETS plug-in will give an error message and prompt you to correct your settings.

Disabling the room temperature regulator (dew-point operation) will have no effect on the temperature alarm.

4.9 Valve protection

To avoid calcification or seizing of the triggered radiator or cooling system control valves you can perform cyclic valve protection. The *"valve protection"* parameter in the *"room temperature regulator function"* parameter branch will activate valve protection if set to *"yes"*.

This protective function will, in general, only be started for variable outputs which are not active, i. e. for outputs which have not requested any heating or cooling energy during the past 24 hours.
For these outputs, the regulator will cyclically set the variable to its maximum value for a period of some 5 minutes once per day, taking account of the following parameterization:

Variable output not inverted: 1 bit variable: "1", 1 byte variable: "255",
Variable output inverted: 1 bit variable: "0", 1 byte variable: "0".

Thus, even such valves which are closed for longer periods will be shortly opened at regular intervals.

Valve protection is controlled by the internal clock and activated at 8:00 a. m. for the output variables concerned. If the internal clock was not yet set after a reset, valve protection will be done for the first time some 32 hours after the reset, at the earliest.

5. Timers

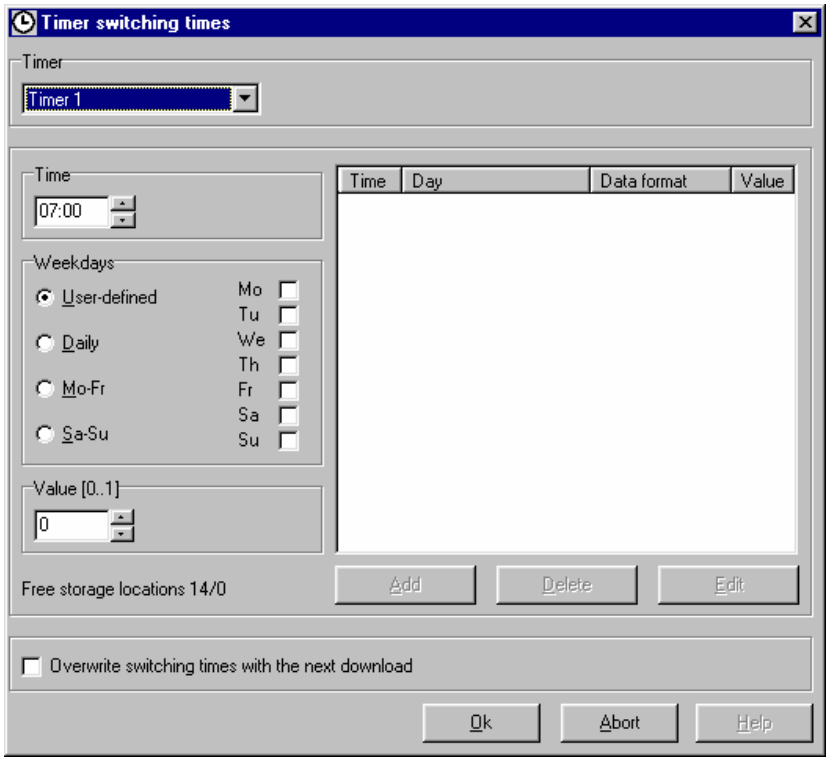
5.1 Function

The B.IQ push button RTR has up to two separate timers. These functions allow time-controlled transmission of switching commands (ON/OFF), value telegrams (0…255) or light scene recall telegrams (1…8) to the bus through separate objects, depending on the data type.

You can use such commands, among other things, for triggering move or position objects of blinds or roller shutters. However, any other bus control options are also possible by such commands.

You can use the *"timer 1"* or *"timer 2"* parameters in the *"B.IQ RTR push button"* parameter branch to separately enable the timers. When the functions are in the deactivated state (default) the parameters and the objects of the timers will be hidden.

As soon as one of the two timers has been enabled, the *"timer"* menu item will be activated in the *"timer editor"* main menu. When you select this menu item the "timer switching times" window will appear. In dependence on the parameterization of the two timers, you can parameterize from this window each of the up to 14 switching times down to the minute, together with the associated switching command (value or scene), depending on the parameterized data type:



In the left part of the window, you can define the time of the switching program down to the minute. Furthermore, you can define the weekdays on which the switching time event is to take place. You can choose from the "user-defined" (Mo, Tu, …, Su), "daily" (Mo - Su), "Mo – Fr" or "Sa – Su" options.

Each switching program will occupy one memory location in the B.IQ RTR push button. For the "daily", "Mo – Fr" or "Sa – Su" settings, only one memory location will be required. For the "user-defined" option, however, sometimes up to five different switching programs can be created, depending on the parameterized days. The "free storage locations" information parameter in the editor indicates how many memory locations are available (the number in front of the slash) or how many memory locations will have to be occupied by the selected setting (the number behind the slash).

After you have set the switching time the weekday and the control command, you can click on the "add" button to create the switching program. You can create the maximum of 14 switching programs per timer from the program list in the right part of the window. If you highlight a program in the list and click on the "edit" button you can edit this program. Highlighting a program and clicking on the "delete" button will delete the program and remove it from the list.
Click on "OK" to accept the settings into the configuration of the push button.

If you check the "overwrite switching times with the next download" option in the editor window the switching programs of the timers will be loaded into the device when the entire application is being programmed or when you are partially programming the parameters.

By local operation in the programming mode, you can edit the switching times of both timers. A precondition to do so is that "complete operation" through the display buttons on the push button has been enabled (depending on the parameter). Thus, you can subsequently change the times or commands programmed through the ETS plug-in. With the next download, you can have the ETS replace the locally changed data by the originally parameterized times. For this purpose, you must have checked the "overwrite switching times with the next download" option in the "timer switching times" window.
If this option is not checked no switching timer data configured in the ETS plug-in will be loaded into the device, neither will any changed switching, value or scene commands. The locally set switching times will thus remain unchanged.

Notes on the timers:

- The switching times can be configured down to the minute. The switching times are checked by the push button time control also at minute intervals when the timer is active.
  The internal clock of the push button should be set by an external time control signal at least every hour to keep the time error as low as possible.
  In exceptional cases, there may be major differences between the time followed up in the push button and the time received from the bus so that switching times events will not be executed (skipped switching minute). For this reason, synchronization from the bus should not take place at preset switching times.
  Unless the internal clock has been updated through the bus at least once per day (update check at 4:00 a. m.) the display of the push button will read "--:--" if the time is indicated in the display (depending on the corresponding parameter). However, the internal clock will keep running with the expected time error, and the switching programs of the control function(s) will still be executed.

- If several switching times have been parameterized to the same time on the same day only the command of that switching time, whose switching time number is of higher order, will be transmitted to the bus.

- The timers will only execute the given switching programs after they have received a valid time.

- Receiving a date is not required for the function of the timers.

5.2 Activating and deactivating the timers

If enabled, you can separately activate or deactivate the timers by local operation in the programming mode (refer to "1.4 Programming mode/local operation", page 25) and/or by push button actuation (push button function). When this function has been activated the " ⏲ " symbol will light up in the display, and the switching programs will be executed in chronological order in accordance with the parameterized switching times. Please note that this " ⏲ " symbol is not exclusive and will also light up when a room temperature timer has been activated.
After switching programs have been fed into the device the timers will be activated immediately after the initializing phase, and the programs will be executed. A valid time and a valid weekday received are a prerequisite to this. If no programs have been stored in the device and the function itself has been enabled no switching programs will be executed, although the symbol will be lit after a reset.

In addition, you can temporarily suppress the transmission of a bus command by the timers via separate disabling objects. To facilitate such disabling function set the *"blocking object"* parameter in the *"timer X"* (X = 1 or 2) parameter branch to *"yes"*. In this case, disabling object 55, *"disabling timer 1"*, or disabling object 57, *"disabling timer 2"* will be enabled. You can parameterize their polarities.
When the timer has been disabled through the bus, the " ⏲ " symbol will disappear from the display. The symbol will not go out if other timers have been activated (timer 1 or 2, or room temperature timer).
During the period of an active disabling function, no commands will be sent to the bus. If you re-enable the room temperature timer exactly at the moment of a parameterized switching time the switching program concerned will not be executed.
Activation or deactivation of the timer during the disabling phase will be stored and effected after the end of disabling.

6. Scene Function

6.1 Scene definition

Similar to a light scene push button, the B.IQ push button RTR has a scene function. Under this function, you can save in the push button up to eight different scenarios. Each scene can trigger up to eight bus outputs (scene objects). You can configure switching, value or Shutter position commands.

You can generally use the *"scene function"* parameter in the *"B.IQ RTR push button"* parameter branch to enable the scene function. When the functions are in the deactivated state (default) the parameters and the objects of the scene functions will be hidden.

Depending on the respective scene recalled, the scene commands are transmitted to the bus via the scene outputs. In the *"scene function – scene X"* (X = 1 to 8) parameter branch, you can separately define the scene command for each output.
In the ETS plug-in, the data type for each scene object can be parameterized in the *"scene function"* parameter branch. Possible types and, thus, available commands are:

Data Type	Scene Command
Switching (1 bit)	ON ("1") OFF ("0")
Value (1 byte)	0...255 alternative * 0...100 %
Shutter / blind position (1 byte)	0...100 % Position "0" = top

*: The *"value type"* parameter in the *"scene function"* parameter branch defines whether dimensionless values (0…255) or percentage values (0…100 %) shall be preset for the *"value"* data type.

Up to eight scene commands per scene can be transmitted to the bus via the output objects. For each scene output, you can parameterize whether a command should be sent at all when a scene is being recalled. The *"transmit output signal = yes"* parameter setting in the *"scene function - scene X"* (X = 1 to 8) parameter branch enables the scene command. Consequently, you can use setting *"no"* to suppress scene commands for the output concerned.

The scene commands are permanently saved in the push button so that they will not get lost after a bus voltage failure.

6.2 Scene recall/scene saving

You can recall a scene by:

- the scene extension object (object 67):
  A scene number received via the scene extension object will recall an internally stored scene. This way of recalling is frequently used by external bus components such as push buttons, display panels or complex scene control systems.

- a local push button function on the push button sensor:
  In addition, you can recall a stored scene by locally actuating a button on the push button. If you have parameterized the push button function to *"light scene extension/recall"*, and if you want the button to act as *"internal scene request"*, you can recall the scenes stored in the B.IQ push button RTR by a short push button actuation (< 1 s). For this purpose, you must specify the corresponding scene number (1 to 8) in the ETS plug-in (refer to "3. Push button Functions", page 44).
  For this function, the extension object will only be required if triggering by external bus components is involved.

Even after they have been programmed by the ETS, you can still change the scenes stored in the B.IQ RTR push button. You can save a scene by:

- the scene extension object (object 67):
  Via the extension object, a storage telegram is received. According to the scene number, the scene control of the B.IQ push button RTR will request the current values of the scene objects from the actuators via the bus and save them permanently.

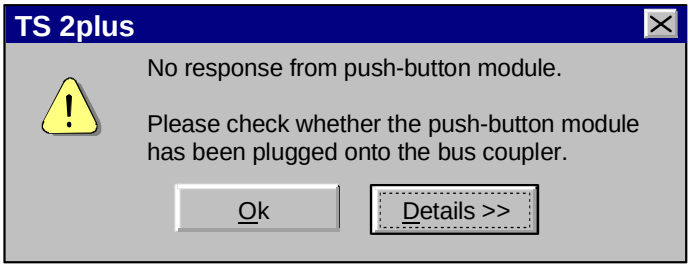
- a local button function on the push button:
  When parameterizing an *"internal scene request"* with the memory function enabled, you can actuate the button for a long time (> 5 s) to save an internal scene in accordance with the parameterized scene number. In this connection, the scene control of the B.IQ push button RTR will request the current values of the scene objects from the actuators via the bus and save them permanently.

In a saving process, the scene commands of the scene concerned and originally configured by the ETS will be replaced by the new values.
If the B.IQ push button RTR does not receive any acknowledgement of a read request no new command will be saved. Non-transmitting scene objects of a scene cannot be changed.
In general, switching commands, value commands or Shutter positions can be saved anew.

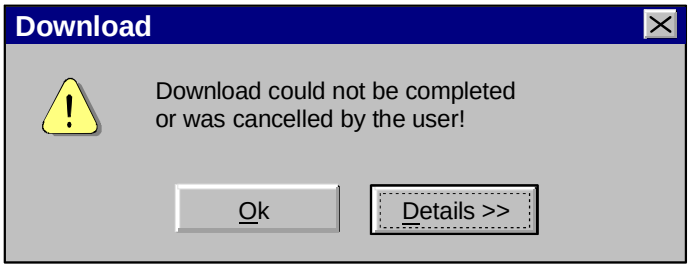
To enable the actuators on the bus to respond to the read request from the B.IQ push button RTR set the read flag ("R" flag) on the actuator objects concerned.
In order to avoid communication problems when recalling or saving scenes do not change the communication flags ("C" flags) of the scene objects on the B.IQ RTR push button.

7. Messages in a Programming Process

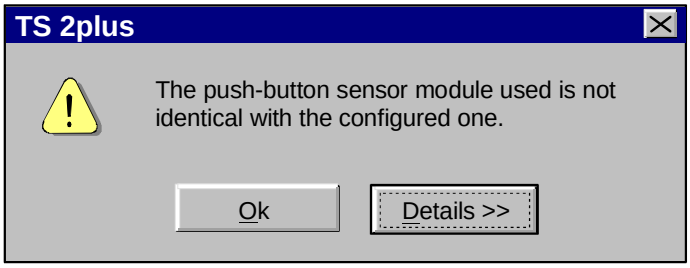
After you have programmed the B.IQ push button RTR with the aid of the ETS plug-in, you can program it in the ETS start-up environment.
During a programming process, the following messages can appear:



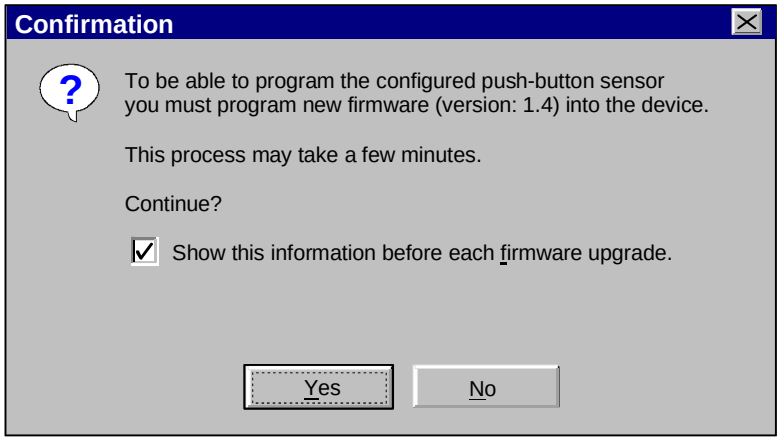
Reason: Attempt to load the application data into the device.
Cause: No B.IQ push button RTR has been plugged onto the bus coupling unit.
Remedy: Plug the B.IQ push button RTR onto the bus coupling unit. When doing so, keep in mind the correct physical address of the bus coupling unit.
Note: You can also program the physical address of the device, even though you have not plugged on the push button. Even for application data partial programming, you must have plugged on the push button.



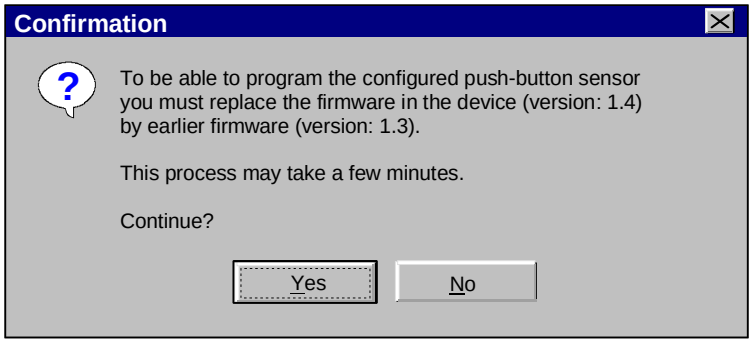
Reason: Attempt to load the application data into the device.
Cause: The programming process was cancelled through the "cancel" button, or there was a communication error.
Remedy: Start a new programming process.
Note: During a programming process, especially when the firmware is being programmed, larger data quantities are sent to the device via the bus. In this connection, the intelligent programming algorithm of the B.IQ push button RTR can recognize communication errors itself and re-transfer the erroneous data. Rarely, errors can occur which cannot even be avoided by repeating the data transfer. In such cases, changing the data interface, the PC or the serial data connection to the data interface can be a remedy.
In the ETS plug-in of the B.IQ RTR push button, you can specify the number of download attempts in the event of a problem from the "settings - options" menu on the "hardware" tab. The default setting of three attempts should only be changed in exceptional cases.
Please note that updating the firmware will only be necessary for special exceptions.



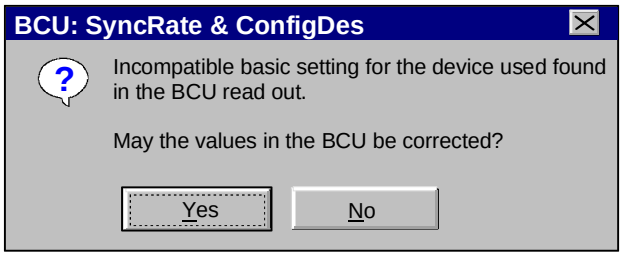
Reason: Attempt to load the application data into the device.
Cause: A B.IQ push button RTR variant other than the configured one has been plugged onto the bus coupling unit (e. g. 5gang-type configured and 2gang-type plugged on).
Remedy: Plug on the variant which conforms with the configuration.



Reason: Attempt to load the application data into the device.
Cause: A B.IQ push button RTR containing earlier firmware (e. g. V 1.3) is being programmed with some later B.IQ push button RTR software version.
Remedy: This message does not represent an error. If you click on "yes" to accept, different firmware which corresponds to the current configuration will be automatically loaded into the device. If you select "no" the earlier device will not be programmed as the parameters and functions preset by the new software are not downwards compatible.
Note: If you uncheck the "show this information before each firmware upgrade" box this message will not appear again, even though you are programming further earlier B.IQ push button RTR versions.
You can re-check this box later in the ETS plug-in of the B.IQ push button RTR from the "settings - options" menu on the "hardware" tab.



Reason: Attempt to load the application data into the device.
Cause: You want to program a B.IQ push button RTR containing new firmware (e. g. V 1.4). In this case, the version in the device is later than that specified by the B.IQ push button RTR software used.
Remedy: This message does not represent an error. If you click on "yes" to accept, the later firmware contained in the device will be replaced by the earlier firmware version specified by the ETS plug-in. If you select "no" the later device will not be programmed as the parameters and functions preset by the earlier software are not upwards compatible. In such case, you should reinstall current B.IQ push button RTR software.
Depending on the changes resulting therefrom, it can possibly become necessary to configure a new device in the ETS.



Reason: Attempt to load the application data into the device.
Cause: The B.IQ push button RTR has been plugged onto a bus coupling unit which does not go with the configuration of the push button. It is probably a bus coupling unit which was originally used for a different purpose or a new device which has not been used yet in the existing B.IQ push button RTR configuration.
Remedy: This message does not represent an error. If you click on "yes" to confirm, the data in the BCU will be overwritten. If you select "no" the later device will not be programmed as the data in the BCU does not go with the configuration of the push button.

Parameters		
Description:	**Values:**	**Comment:**
B.IQ RTR push button		
Push button function	Disabled **Enabled**	This parameter specifies whether the push button function is activated or deactivated.
Room temperature regulator function	Disabled **Enabled**	This parameter specifies whether the room temperature regulator function is activated or deactivated.
Scene function	Disabled **Enabled**	This parameter specifies whether the scene function is activated or deactivated.
Timer 1	**Disabled** Enabled	This parameter specifies whether timer 1 is activated or deactivated.
Timer 2	**Disabled** Enabled	This parameter specifies whether timer 2 is activated or deactivated.
Alarm function after pulling off the application module	**Disabled** Enabled	When the B.IQ push button RTR is pulled off the flush-mounted bus coupling unit an alarm message can be sent to the bus. This parameter specifies whether the alarm function is enabled or disabled.
Alarm object data format	**Switching telegram, 1 bit** Value telegram, 1 byte Value telegram, 2 byte	To specify the data format of the alarm message.
Telegram after removal	OFF telegram **ON telegram**	To specify the value of the switching telegram sent when an alarm message is being raised. Only for data format = "switching telegram, 1 bit".
Value after removal	0 to 255, **255** (for "reset value = no") 1 to 255, **255** (for "reset value = yes")	To specify the value of the value telegram sent when an alarm message is being raised. Only for data format = "value telegram, 1 byte".
Value after removal	0 to 65535, **65535** (for "reset value = no") 1 to 65535, **65535** (for "reset value = yes")	To specify the value of the value telegram sent when an alarm message is being raised. Only for data format = "value telegram, 2 byte".
Reset value	**No** Yes	To specify whether the alarm value should be automatically reset to its inverse value ("0", no alarm) after the application module has been re-plugged.
Light duration of status LED at operation indication	1 s 2 s **3 s**	To define the period for which the status-LED will be lit as operation indication.

Push button assistance function	**Disabled** Enabled	To specify whether the push button assistance function is enabled. When the function has been enabled and a button is actuated (except the display buttons) the display will show a help text which can describe the assigned button function. The button help text can be specified in the plug-in separately for each push button or rocker.
Operation via display buttons		The B.IQ push button RTR has several local operator levels:
	No operation	'Normal operation' and local operation of the regulator by actuation of the display buttons for shifting the basic set value possible.
	Limited operation	Switching over into the programming mode is possible. → 'Normal operation' including set value shifting and switching over the operating mode and resetting of the different heating and/or cooling set values is possible.
	Complete operation	Complete access to the device with local operation. In addition to restricted operation, it grants the user access to the up to three timers (activation/ deactivation of the timers and changing of the individual switching programs) and to the "settings" menu (activation/deactivation of the button assistance function, the LCD background light and the button lock as well as of the version data display).

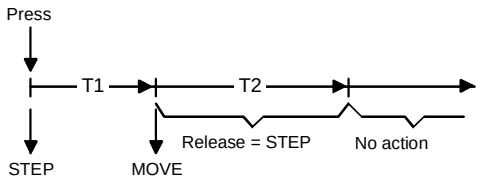
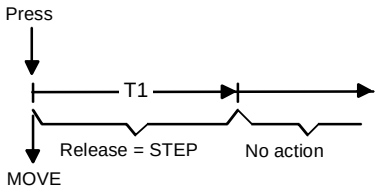
Display		
Illumination		The triggering of the display background light and of the device ON LED (blue) can be defined.
	ON	To permanently switch on the light.
	Automatic switch-off	The light will switch on upon a button actuation and will be automatically disabled after the preset time has elapsed.
	Switching via object	The light is triggered via object 22.
Kind of switching	**ON**	When the light is switched via the object it can be specified whether the light should remain permanently enabled ("ON"), or whether it should automatically go out after the parameterized time has elapsed. Only for "illumination = switching via object".
	Automatic switch-off	
Automatic illumination switch-off (1…1200) * 1 s	0 s to 1200 s, **10 s**	To specify the ON period of the display background light and of the device ON LED (blue). The time can be re-triggered. Only for " illumination = automatic switch-off" or "kind of switching = automatic switch-off".
Display of	Time / room temperature Time / outside temperature Outside temperature **Room temperature** Setpoint temperature Date / time Outside / room temperature Date / time / outside temperature Date / time / room temperature	Various kinds of information can be read by the display. You can select them from here.

Push button function - disable		
Disabling function	**Push button not disabled**	This parameter defines the behaviour of the push button when the disabling function is active. The disabling function is deactivated.
	Function of all rockers like rocker 1…n	When the disabling function is active all rockers of the B.IQ push button RTR will behave in the same way as the parameterized one.
	Single rocker disabled	When the disabling function is active specific individual rockers of the B.IQ push button RTR can be disabled.
	Push button disabled	When the disabling function is active the entire push button will be disabled.
Polarity of blocking object	Inverted (disabling = 0) **Not inverted (disabling = 1)**	To specify the disabling object polarity.
Function like rocker	**Rocker 1** (3-, 4- and 5gang) Rocker 2 (3-, 4- and 5gang) Rocker 3 (3-, 4- and 5gang) Rocker 4 (4- and 5gang) Rocker 5 (5gang)	When the disabling function is active all rockers of the B.IQ push button RTR will behave in the same way as the parameterized one. Only for disabling function = "function of all rockers like rocker 1…n".
Rocker x disabled? X = 1 to 3 (3gang) X = 1 to 4 (4gang) X = 1 to 5 (5gang)	**No** Yes	To specify whether rocker X will be disabled when the disabling function is active, i. e. button actuation (left and right) of this rocker will show no function. Only for disabling function = "single rocker disabled".

Push button function - general		
Rocker X: concept of operation X = 1 to 3 (3gang) X = 1 to 4 (4gang) X = 1 to 5 (5gang)	 **2 push buttons (2 objects)**	For the B.IQ RTR push button, two button functions or one rocker function can be assigned to each of the individual rockers. Two independent push button functions are assigned to rocker X.
	rocker (1 object)	A rocker function is assigned to rocker X.
	Without function	Rocker X has no function, i. e. button actuation (left or right) will have no effect, and the status-LEDs of this rocker cannot be triggered.
Push Push button 1 actuation push button function (3-, 4- and 5gang)		
Functions of button 1	No function **Switching/pushing** Dimming Shutter value transmitter 1 byte value transmitter 2 byte Operating mode switch-over Light scene extension/recall Room temperature timer operation * Timer operation **	To specify the function of button 1. *: Room temperature timer operation can only be parameterized when the room temperature timer has been enabled. **: Timer operation can only be parameterized when at least one control function has been enabled.
Push button 1 function = "no function"		
Text for push button assistance	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short button actuation (< 0.4 s) when the button assistance function has been enabled.
Function of the status-LED		For Push button 1 function = "no function", only the status-LED of the button can be triggered via the corresponding object. Button actuation will show no response.
	Always OFF	The status-LED will always be OFF.
	Always ON	The status-LED will always be ON.
	Status indication (switching object)	The status-LED will indicate the object status.
	Inverted Status indication (switching object)	The status-LED will indicate the inverted object status.

Push button 1 function = "switching/touch control"		
Text for push button assistance	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short button actuation (< 0.4 s) when the button assistance function has been enabled.
Function of the status-LED		To specify the function of the status-LED.
	Always OFF	The status-LED will always be OFF.
	Always ON	The status-LED will always be ON.
	Status indication (switching object)	The status-LED will indicate the object status.
	Inverted Status indication (switching object)	The status-LED will indicate the inverted object status.
	Operation indication	When the push button is actuated the status-LED will light up for the parameterized time.
Command upon pressing the push button	No function ON **TOGGLE** OFF	To specify the command sent when the push button is being pressed.
Command on releasing the push button	**No function** ON OFF TOGGLE	To specify the command sent when the push button is being released.
Push button 1 function = "dimming"		
Text for push button assistance	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short button actuation (< 0.4 s) when the button assistance function has been enabled.
Function of the status-LED		To specify the function of the status-LED.
	Always OFF	The status-LED will always be OFF.
	Always ON	The status-LED will always be ON.
	Status indication (switching object)	The status-LED will indicate the object status.
	Inverted Status indication (switching object)	The status-LED will indicate the inverted object status.
	Operation indication	When the push button is actuated the status-LED will light up for the parameterized time.

Command on pressing the push button, push button function			To specify the response to push button actuation.
	Darker (OFF)		Short push button actuation will release an OFF telegram, whereas move actuation will cause a dimming telegram (darker).
	Brighter (ON)		Short push button actuation will release an ON telegram, whereas move actuation will cause a dimming telegram (brighter).
	Brighter/darker (TOGGLE)		The internally stored switching status will be toggled When the push button is pressed shortly. If the stored status is ON (OFF) an OFF (ON) telegram will be raised. By pressing the button for a long time a "darker" telegram will be sent after a "brighter" telegram, and vice versa.
Dimming brighter by	**100 %** 50 % 25 % 12.5 %	6 % 3 % 1.5 %	To specify the maximum dimming step width of a dimming telegram. With a dimming telegram, you can increase the brightness by a maximum of X %. This parameter depends on the push button function selected.
Dimming darker by	**100 %** 50 % 25 % 12.5 %	6 % 3 % 1.5 %	To specify the maximum dimming step width of a dimming telegram. With a dimming telegram, you can reduce the brightness by a maximum of X %. This parameter depends on the push button function selected.
Stop telegram	**Yes** No		Releasing the push button will send or not send a stop telegram.
Time between switching and dimming (0.1 ... 51) * 1 s	0.1 s to 51 s, **0.4 s** (Step width: 0.1 s)		Time from which on the function of a move actuation (dimming) will be executed.
Telegram repetition	**No** Yes		Cyclic dimming telegram repetition during push button actuation.
Time between two telegrams	**200 ms** 300 ms 400 ms 500 ms	750 ms 1 s 2 s	Time between two telegrams when telegram repetition has been set. Each time this period has elapsed, a new dimming telegram will be sent. Only for telegram repetition = "yes".

Push button 1 function = "Shutter"		
Text for push button assistance	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short button actuation (< 0.4 s) when the button assistance function has been enabled.
Function of the status-LED		To specify the function of the status-LED.
	Always OFF	The status-LED will always be OFF.
	Always ON	The status-LED will always be ON.
	Operation indication	When the push button is actuated the status-LED will light up for the parameterized time.
Operation concept	**Step - move - step** Move - Step	To specify the telegram sequence after pressing of the button. Step - move - step:  When the push button is pressed a STEP telegram will be sent and start time T1 (time between step and move actuation) started. If you release the button within time T1 no further telegram will be sent. This STEP telegram serves for stopping an ongoing continuous run. If the push button remains pressed for longer than time T1 a MOVE telegram will be automatically sent after T1 has elapsed, and time T2 (lamella adjustment time) will be started. If you then release the button within T2 a STEP telegram will be sent. This function is used for lamella adjustment. T2 should correspond to a 180° lamella rotation. Move - step:  When the push button is pressed a MOVE telegram will be sent and start time T1 (lamella adjustment time) started. If you then release the button within T1 a STEP telegram will be sent. This function is used for lamella adjustment. T1 should correspond to a 180° lamella rotation.

Time between step and move operation (0.1 ...127.5) * 1 s	0.1 s to 127.5 s, **0.3 s** (Step width: 0.1 s)	Time from which on the function of a move button actuation will be executed. Only for operation concept = "step - move - step".
Function of shutter push button	**UP**	Short push button actuation will release a STEP telegram (UP), whereas move actuation will cause a MOVE telegram (UP).
	DOWN	Short push button actuation will release a STEP telegram (DOWN), whereas move actuation will cause a MOVE telegram (DOWN).
	TOGGLE	This setting enables the moving direction internally stored and followed up via the bus to be changed upon each move actuation (MOVE). If a step actuation causes a STEP message to be sent, such STEP will always be opposite to the last MOVE message. Several successive STEP telegrams will always have the same direction.
Lamella adjustment time, (0 ... 127.5) * 1 s	0 s to 127.5 s, **0.6 s** (step width: 0.1 s)	Time during which a MOVE telegram sent can be stopped by releasing the button (STEP). This function serves for lamella adjustment of a Shutter.
Push button 1 function = "value transmitter 1 byte"		
Text for push button assistance	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short button actuation (< 0.4 s) when the button assistance function has been enabled.
Function of the status-LED		To specify the function of the status-LED.
	Always OFF	The status-LED will always be OFF.
	Always ON	The status-LED will always be ON.
	Operation indication	When the push button is actuated the status-LED will light up for the parameterized time.
Value (0…255)	0 to 255, **255**	To specify the value to be sent.
Adjustment of values by long pressing of a push button	**Disabled** Enabled	If the push button remains pressed for at least 5 s the current value will be cyclically reduced by the parameterized step width and sent. After the push button is released, the value sent last will be kept stored. This parameter specifies whether value changing will be possible.
Time between two telegrams	0.5 s, **1 s**, 2 s, 3 s	Time between two cyclic telegrams When the push button is pressed for a long time.
Step width (1…10)	1 to 10, **10**	Step width by which the value set will be decreased upon move button actuation.

Push button 1 function = "value transmitter 2 byte"		
Text for push button assistance	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short button actuation (< 0.4 s) when the button assistance function has been enabled.
Function of the status-LED		To specify the function of the status-LED.
	Always OFF	The status-LED will always be OFF.
	Always ON	The status-LED will always be ON.
	Operation indication	When the push button is actuated the status-LED will light up for the parameterized time.
Function as	**Temperature value transmitter, EIS 5** Brightness value transmitter, EIS 5 Value transmitter, EIS 10	To specify the function to be executed.
Temperature value (0...40) * 1 °C	0 to 40 °C in 1 °C steps, **25 °C**	To set the temperature value to be sent. Only for function as = "temperature value transmitter, EIS 5".
Brightness value (0...1500) * 1 lux	0 to 1500 lux in 50 lux steps, **500 lux**	To set the brightness value to be sent. Only for function as = "brightness value transmitter, EIS 5".
Value (0…65535)	0 to 65535, **0**	To set the EIS 10 value to be sent. Only for function as = "value transmitter, EIS 10".
Adjustment of values by long pressing of a push button	**Disabled** Enabled	If the push button remains pressed for at least 5 s the current value will be cyclically reduced by the parameterized step width and sent. After the push button is released, the value sent last will be kept stored. This parameter specifies whether value changing will be possible.
Time between two telegrams	0.5 s, **1 s**, 2 s, 3 s	Time between two cyclic telegrams when the push button is pressed for a long time.
Step width	Temperature value transmitter, EIS 5 **1 °C** Brightness value transmitter, EIS 5 **50 lux** Value transmitter, EIS 10 1 75 2 100 5 200 **10** 500 20 750 50 1000	Step width by which the value set will be decreased upon move push button actuation.

Push button 1 function = "operating mode switch-over"		
Text for push button assistance	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short button actuation (< 0.4 s) when the button assistance function has been enabled.
Function of the status-LED		To specify the function of the status-LED.
	Always OFF	The status-LED will always be OFF.
	Always ON	The status-LED will always be ON.
	Display operating mode active	The status-LED will shine as soon as the operating mode assigned to it has been activated by the actuation of the push button.
	Display operating mode inactive	The status-LED will shine as soon as the operating mode assigned to it has been deactivated by the actuation of the push button.
	Operation indication	When the push button is actuated the status-LED will light up for the parameterized time.
Operating mode on pressing a push button	**Comfort operation** Standby operation Night operation Frost/heat protection operation Presence button *	To specify the room temperature regulator function to be activated by the actuation of the push button. *: The presence button can only be parameterized when the "presence detection by presence button" option has been enabled.
Push button 1 function = "light scene extension/recall"		
Text for push button assistance	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short button actuation (< 0.4 s) when the button assistance function has been enabled.
Function of the status-LED		To specify the function of the status-LED.
	Always OFF	The status-LED will always be OFF.
	Always ON	The status-LED will always be ON.
	Operation indication	When the push button is actuated the status-LED will light up for the parameterized time.
Function as		To define the scene recall functional principle.
	Light scene extension	To recall an 'external' light scene via the light scene extension object by push button actuation.
	Internal scene request *	To recall an 'internal' scene by push button actuation. *: This setting can only be parameterized when the light scene function of the B.IQ push button RTR has been enabled.

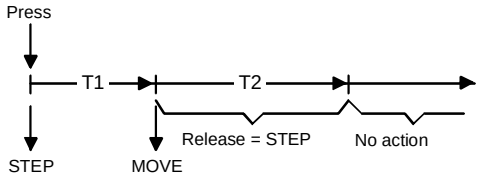
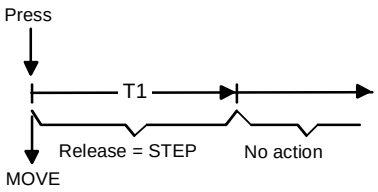
Light scene (1…64)	1 to 64, **1**	To specify the light scene number to be sent via the object. Only for "function as = light scene extension".
Scene (1…8)	1 to 8, **1**	To specify the number of the internal scene to be recalled. Only for "function as = internal scene request".
Memory function	**No** Yes	This parameter will enable the memory function. When this function is enabled, a long push button actuation (> 5 s) will transmit a storage telegram or save the internal scene according to the parameterized number, respectively.
Push button 1 function = "room temperature timer operation"		
Text for push button assistance	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short button actuation (< 0.4 s) when the button assistance function has been enabled.
Function of the status-LED		To specify the function of the status-LED.
	Always OFF	The status-LED will always be OFF.
	Always ON	The status-LED will always be ON.
	Display room temperature timer active	The status-LED will shine as soon as the room temperature timer has been activated by the actuation of the push button.
	Display room temperature timer inactive	The status-LED will shine as soon as the room temperature timer has been deactivated by the actuation of the push button.
	Operation indication	When the push button is actuated the status-LED will light up for the parameterized time.
Reaction on pressing a push button	Activate room temperature timer (ON)	Actuating the push button will activate the room temperature timer. The parameterized switching programs will be executed.
	Deactivate room temperature timer (OFF)	Actuating the push button will deactivate the room temperature timer. The parameterized switching programs will not be executed or will be suppressed, respectively.
	Deactivate / activate room temperature timer (TOGGLE)	Actuating the push button will activate or deactivate the room temperature timer. Switch-over between executing and suppressing the switching programs.

Push button 1 function = "timer operation"		
Text for push button assistance	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short button actuation (< 0.4 s) when the button assistance function has been enabled.
Function of the status-LED		To specify the function of the status-LED.
	Always OFF	The status-LED will always be OFF.
	Always ON	The status-LED will always be ON.
	Display timer active	The status-LED will shine as soon as the timer has been activated by the actuation of the push button.
	Display timer inactive	The status-LED will shine as soon as the timer has been deactivated by the actuation of the push button.
	Operation indication	When the push button is actuated the status-LED will light up for the parameterized time.
Function	Both timers enabled: **Timer 1** Timer 2 Only timer 1 enabled: **Timer 1** Only timer 2 enabled: **Timer 2**	Depending on which timer(s) has (have) been enabled, this parameter specifies which of the timer(s) should be triggered by the push button function.
Reaction on pressing a push button	Activate timer (ON)	Actuating the push button will activate the corresponding timer. The parameterized switching programs will be executed.
	Deactivate timer (OFF)	Actuating the push button will deactivate the corresponding timer. The parameterized switching programs will not be executed or will be suppressed, respectively.
	Deactivate / activate timer (TOGGLE)	Actuating the push button will activate or deactivate the over-over between executing and suppressing the switching programs.
For push button function – general – button 2, refer to Push button 1 (3-, 4- and 5gang).		
For push button function – general – button 3, refer to Push button 1 (3-, 4- and 5gang).		
For push button function – general – button 4, refer to Push button 1 (3-, 4- and 5gang).		
For push button function – general – button 5, refer to Push button 1 (3-, 4- and 5gang).		
For push button function – general – button 6, refer to Push button 1 (3-, 4- and 5gang).		
For push button function – general – button 7, refer to Push button 1 (4- and 5gang).		
For push button function – general – button 8, refer to Push button 1 (4- and 5gang).		
For push button function – general – button 9, refer to Push button 1 (5gang only).		
For push button function – general – button 10, refer to Push button 1 (5gang only).		

Push button function – general – rocker 1 (3-, 4- and 5gang)		
Function of the rocker	No function **Switching** Dimming Shutter Operating mode switch-over	To specify the function of rocker 1.
Rocker 1 function = "no function".		
For rocker 1 function = "no function", only the status-LED of the rocker can be triggered via the corresponding status object. Rocker or push button actuation will show no response. Status-LED parameter only (refer to "rocker 1 status").		
Rocker 1 function = "switching"		
Text for push button assistance [left]	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short actuation of the left button (< 0.4 s) when the push button assistance function has been enabled.
Text for push button assistance [right]	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short actuation of the right button (< 0.4 s) when the push button assistance function has been enabled.
Command on pressing a rocker	Left = ---, right = --- Left = OFF, right = ON **Left = ON, right = OFF** Left = TOGGLE, right = TOGGLE Left = OFF, right = OFF Left = ON, right = ON	To specify the commands to be sent when both push buttons of the rocker are being pressed.
Rocker 1 function = "dimming"		
Text for push button assistance [left]	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short actuation of the left button (< 0.4 s) when the push button assistance function has been enabled.
Text for push button assistance [right]	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short actuation of the right button (< 0.4 s) when the push button assistance function has been enabled.

Command on pressing a rocker			To specify the response to rocker button actuation.
	Left = brighter (ON), Right = darker (OFF)		Short push button actuation (left button) will release an ON telegram, while move actuation (left button) will cause a dimming telegram (brighter). Short push button actuation (right button) will release an OFF telegram, while move actuation (right button) will cause a dimming telegram (darker).
	Left = darker (OFF), Right = brighter (ON)		Short push button actuation (left button) will release an OFF telegram, while move actuation (left button) will cause a dimming telegram (darker). Short push button actuation (right button) will release an ON telegram, while move actuation (right button) will cause a dimming telegram (brighter).
	Left = TOGGLE, right = TOGGLE		The internally stored switching status will be toggled When the push button is pressed shortly. If the stored status is ON (OFF) an OFF (ON) telegram will be raised. By pressing the button for a long time a "darker" telegram will be sent after a "brighter" telegram, and vice versa.
	Left = brighter (ON), Right = brighter (ON)		Short push button actuation (left button) will release an ON telegram, while move actuation (left button) will cause a dimming telegram (brighter). Short push button actuation (right button) will also release an ON telegram, while move actuation (right button) will also cause a dimming telegram (brighter).
	Left = darker (OFF), Right = darker (OFF)		Short push button actuation (left button) will release an OFF telegram, while move actuation (left button) will cause a dimming telegram (darker). Short push button actuation (right button) will also release an OFF telegram, while move actuation (right button) will also cause a dimming telegram (darker).
Dimming brighter by	**100 %** 50 % 25 % 12.5 %	6 % 3 % 1.5 %	To specify the maximum dimming step width of a dimming telegram. With a dimming telegram, you can increase the brightness by a maximum of X %.
Dimming darker by	**100 %** 50 % 25 % 12.5 %	6 % 3 % 1.5 %	To specify the maximum dimming step width of a dimming telegram. With a dimming telegram, you can reduce the brightness by a maximum of X %.
Stop telegram?	**Yes** No		Releasing one of the push buttons (left or right) will send or not send a stop telegram.

Time between switching and dimming (0.1 ... 51) * 1 s	0.1 s to 51 s; **0.4 s** (step width: 0.1 s)	Time from which on the function of a move actuation (dimming) will be executed.
Telegram repetition	**No** Yes	To enable cyclic dimming telegram repetition during push button actuation.
Time between two dimming telegrams	**200 ms** 750 ms 300 ms 1 s 400 ms 2 s 500 ms	Time between two telegrams when telegram repetition has been set. Each time this period has elapsed, a new dimming telegram will be sent. Only for telegram repetition = "yes".
Rocker 1 function = "Shutter"		
Text for push button assistance [left]	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short actuation of the left button (< 0.4 s) when the push button assistance function has been enabled.
Text for push button assistance [right]	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short actuation of the right button (< 0.4 s) when the push button assistance function has been enabled.

Operation concept	**Step - move - step** Move - step	To specify the telegram sequence after push button actuation. Step - move - step:  When the push button is pressed a STEP telegram will be sent and start time T1 (time between step and move actuation) started. If you release the button within time T1 no further telegram will be sent. This STEP telegram serves for stopping an ongoing continuous run. If the push button remains pressed for longer than time T1 a MOVE telegram will be automatically sent after T1 has elapsed, and time T2 (lamella adjustment time) will be started. If you then release the button within T2 a STEP telegram will be sent. This function is used for lamella adjustment. T2 should correspond to a 180° lamella rotation. Move - step:  When the push button is pressed a MOVE telegram will be sent and start time T1 (lamella adjustment time) started. If you then release the button within T1 a STEP telegram will be sent. This function is used for lamella adjustment. T1 should correspond to a 180° lamella rotation.
Time between step and move operation (0.1 ...127.5) * 1 s	0.1 s to 127.5 s; **0.3 s** (step width: 0.1 s)	Time from which on the function of a long push button actuation will be executed. Only for operation concept = "step - move - step".

Command on pressing a rocker	**Left - Shutter UP/ right - Shutter DOWN**	Short push button actuation (left button) will release a STEP telegram (UP), whereas move actuation (left button) will cause a MOVE telegram (UP). Short push button actuation (right button) will release a STEP telegram (DOWN), whereas move actuation (right button) will cause a MOVE telegram (DOWN).
	Left - Shutter DOWN/ right - Shutter UP	Short push button actuation (left button) will release a STEP telegram (DOWN), whereas move actuation (left button) will cause a MOVE telegram (DOWN). Short push button actuation (right button) will release a STEP telegram (UP), whereas move actuation (right button) will cause a MOVE telegram (UP).
	Left - Shutter TOGGLE/ right - Shutter TOGGLE	This setting enables the moving direction internally stored and followed up via the bus to be changed upon each move actuation (MOVE). If a step actuation causes a STEP message to be sent, such STEP will always be opposite to the last MOVE message. Several successive STEP telegrams will always have the same direction.
	Left - Shutter UP/ right - Shutter UP	Short push button actuation (left button) will release a STEP telegram (UP), whereas move actuation (left button) will cause a MOVE telegram (UP). Short push button actuation (right button) will also release a STEP telegram (UP), whereas move actuation (right button) will also cause a MOVE telegram (UP).
	Left - Shutter DOWN/ right - Shutter DOWN	Short push button actuation (left button) will release a STEP telegram (DOWN), whereas move actuation (left button) will raise a MOVE telegram (DOWN). Short push button actuation (right button) will also release a STEP telegram (DOWN), whereas move actuation (right button) will also cause a MOVE telegram (DOWN).
Lamella adjustment time, (0 ... 127.4) * 1 s	0 s to 127.5 s, **0.6 s** (Step width: 0.1 s)	Time during which a MOVE telegram sent can be stopped by releasing the push button (STEP). This function serves for lamella adjustment of a shutter/ blind.

Rocker 1 function = "operating mode switch-over"		
Text for push button assistance [left]	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short actuation of the left button (< 0.4 s) when the push button assistance function has been enabled.
Text for push button assistance [right]	[20-character text] The parameterized push button function is preset.	Here, the push button assistance text can be defined which will be displayed upon a short actuation of the right button (< 0.4 s) when the push button assistance function has been enabled.
Operating mode on pressing a rocker	**Switch-over between the operationg modes comfort, standby, night and frost/heat protection-operation** (No other settings possible.)	To specify the function of rocker 1.
For rocker 2, refer to rocker 1 (3-, 4- and 5gang).		
For rocker 3, refer to rocker 1 (3-, 4- and 5gang).		
For rocker 4, refer to rocker 1 (4- and 5gang).		
For rocker 5, refer to rocker 1 (5gang only).		
Push button function - general - rocker 1 - rocker 1 status (3-, 4- and 5gang)		
Rocker 1 function = "no function", "switching", "dimming", "Shutter" and "operating mode switch-over"		
Show status object via	**Left and right status-LEDs** Left status-LED Right status-LED Inverted left and right status-LEDs Inverted left status-LED Inverted right status-LED Left and right status-LEDs always ON Left and right status-LEDs always OFF	To define the functions of the status-LEDs.
For push button function - general - rocker 2 - rocker 2 status, refer to rocker 1 (3-, 4- and 5gang).		
For push button function - general - rocker 3 - rocker 3 status, refer to rocker 1 (3-, 4- and 5gang).		
For push button function - general - rocker 4 - rocker 4 status, refer to rocker 1 (4- and 5gang).		
For push button function - general - rocker 5 - rocker 5 status, refer to rocker 1 (5gang only).		

Room temperature regulator function		
Operating mode switch-over	**Via value (1 byte)**	Switching over the operating modes via the bus proceeds by a 1 byte object in accordance with the KONNEX specification. In addition, a higher-order override object is available for this setting.
	Via switching (4 x 1 bit)	The operating modes will be switched over via the bus in the 'classical' way through separate 1 bit objects.
Control circuits (HA (high access))	**1 control circuit**	The room temperature regulator will trigger one control circuit only.
	2 control circuits	The room temperature regulator can trigger up to two control circuits.
Heating/cooling mode (HA) Mode of the control ciruits **	**Heating** Cooling Heating and cooling * Basic and additional heating * Basic and additional cooling * Basic/additional heating/cooling *	To set the heating/cooling switch-over. *: The mixed "heating and cooling" mode and two-stage control operation will not be possible when two control circuits are used. **: When two control circuits are configured.
Additional stage inhibit object (HA)		The additional stages can be separately disabled via the bus. The parameter enables the disabling object.
	No	The additional stages cannot be separately disabled.
	Yes	The additional stages can be disabled via the disabling object. Only for two-stage heating or cooling operation.
Send variable heating and cooling to one common object (HA)	**No** Yes	If the parameter is set to "yes" the heating or cooling variable will be sent to a common object. This function can be used if the same heating system in the room is used for cooling in summer and for heating in winter. Only for mixed "heating and cooling" mode using additional stages, if necessary.

Type of heating control (for basic and additional stages, if necessary) (HA)	**Continuous PI control** Switching PI control (PWM) Switching 2-point control (ON/OFF)	To select a control algorithm (PI or 2-point) with data format (1 byte or 1 bit) for the heating system.
Type of heating (for basic and additional stages, if necessary) (HA)	**Warm water heating (5 K / 150 min)** Floor heating (5 K / 240 min) Floor heating (4 K / 100 min) Blower convector (4 K / 90 min) SplitUnit (4 K / 90 min) Via control parameter	To match the PI algorithm to various heating systems with experience values for the proportional range and reset time control parameters. Separate control parameter input. Only for "type of heating control" = "PI".
Heating proportional band (10 ... 127) * 0.1 K (HA)	10...127, **50**	To separately set the "proportional band" control parameter. Only for "type of heating control" = "via control parameter".
Heating reset time (0 ... 255) * 1 min; 0 = inactive (HA)	0...255, **150**	To separately set the "reset time" control parameter. Only for "type of heating control" = "via control parameter".
Heating 2-point controller hysteresis upper limit (5 ... 127) * 0.1 K (HA)	5...127, **5**	To define the heating switch-on and switch-off times. Only for "type of heating control" = "2-point".
Heating 2-point controller hysteresis lower limit (-128 ... -5) * 0.1 K (HA)	-128...-5, **-5**	To define the heating switch-on and switch-off times. Only for "type of heating control" = "2-point".
Type of cooling control (for basic and additional stages, if necessary) (HA)	**Continuous PI control** Switching PI control (PWM) Switching 2-point control (ON/OFF)	To select a control algorithm (PI or 2-point) with data format (1 byte or 1 bit) for the cooling system. Only for "send heating and cooling variables to a common object" = "no". If "send heating and cooling variables to a common object" = "yes" the "type of heating control" parameter settings will be taken over for cooling operation.
Type of cooling (for basic and additional stages, if necessary) (HA)	**Cooling ceiling (5 K / 240 min)** Blower convector (4 K / 90 min) SplitUnit (4 K / 90 min) Via control parameter	To match the PI algorithm to various cooling systems with experience values for the proportional range and reset time control parameters. Separate control parameter input. Only for "type of cooling control" = "PI".

Cooling proportional band (10 ... 127) * 0.1 K (HA)	10...127, **50**	To separately set the "proportional band" control parameter. Only for "type of cooling" = "via control parameter".
Cooling reset time (0 ... 255) * 1 min; 0 = inactive (HA)	0...255, **240**	To separately set the "reset time" control parameter. Only for "type of cooling" = "via control parameter".
Cooling 2-point controller hysteresis upper limit (5 ... 127) * 0.1 K (HA)	5...127, **5**	To define the cooling switch-on and switch-off times. Only for "type of cooling control" = "2-point".
Cooling 2-point controller hysteresis lower limit (-128 ... -5) * 0.1 K (HA)	-128...-5, **-5**	To define the cooling switch-on and switch-off times. Only for "type of cooling control" = "2-point".
Operation of controller inhibitable		You can disable room temperature regulator local operation (all push buttons). Activated disabling of operation will be indicated in the display by the " " symbol.
	No	Disabling function deactivated.
	Always disabled	It will always disable the operation of the regulator.
	Via object	To enable object 39, "disabling controller operation".
Switch off controller (dew point)	**No** Via object	This parameter enables object 40, "disabling controller". There will be no more control action until enabling (variables = 0). Activated disabling of operation (dew-point operation) will be indicated in the display by the " " symbol.
Frost/heat protection		You can specify the way how the regulator will switch over into the frost/heat protection mode.
	Automatic frost protection	Automatic frost protection is activated. This can effect automatic switch-over into the frost protection mode, depending on the room temperature.
	Via window status	The switch-over into the frost/heat protection mode proceeds via the "window status" object.

Automatic frost protection	**OFF** 0.2 K / min 0.3 K / min 0.4 K / min 0.5 K / min 0.6 K / min	To specify the temperature decrease the room temperature must reduce within one minute before the regulator switches over into the frost protection mode. When "OFF" is selected automatic frost protection will be deactivated. Only for "frost/heat protection = automatic frost protection".
Automatic frost protection period (1..255) * 1 min	1 to 255, **20**	To define the time after which the regulator will automatically deactivate frost protection when it is in the automatic frost protection mode. Only for enabled automatic frost protection.
Window status delay (0..255) * 1 min; 0 = inactive	0 to 255, **0**	To define the delay time after which the frost/heat protection mode will be activated by the window status. Only for "frost/heat protection = via window status".
Switch-over between heating and cooling		If heating/cooling switch-over has been parameterized you can change between heating and cooling.
	Automatically	Depending on the operating mode and the room temperature, this switch-over is automatic.
	Via object	The switch-over takes proceeds solely via object 35, *"heating/cooling"*. Only for the "heating and cooling" or "basic/ additional heating/cooling" mixed mode.
Heating/cooling switch-over after reset	**Heating** Cooling Heating/cooling switch-over before reset	To specify the preset heating/cooling switch-over after reset. Only for "switch-over between heating and cooling = via object".
Automatic heating/cooling switch-over transmission	**On changing the heating/cooling** On changing the output value	To specify when a heating/cooling switch-over telegram will be automatically transmitted to the bus via object 35, *"heating/cooling switch-over"*. Only for "switch-over between heating and cooling = automatically".

Cyclical transmission heating/cooling switch-over (0…255) * 1 min; 0 = inactive	0 to 255, **0**	To specify whether the current status of object 35, *"heating/cooling"*, should be cyclically transmitted to the bus when switching over is automatic. You can set the cycle time. Setting "0" will deactivate the cyclic transmission of the object value. Only for "switch-over between heating and cooling = automatically".
Valve protection	**No** Yes	The valve will be cyclically opened (every 24 hrs.). This is a precaution against deposits and prevents choking of the valve.
Room temperature regulator function - set point values		
Own set values for 2nd control circuit	**No** Yes	If you use both control circuits the second control circuit can have separate set values. Setting "yes" will enable set value presetting for the second control circuit. Only for "control circuits = 2 control circuits".
Basic temperature after reset (7.0 ... 40.0) * 1 °C	7.0 °C to 40 °C, **21 °C**	To specify the basic set value after initialization.
Basic temperature in the 2nd control circuit after reset (7.0 ... 40.0) * 1 °C	7.0 °C to 40 °C, **21 °C**	To specify the basic set value of the second control circuit after initialization. Only for "own set values for second control circuit = yes".
Accept modification of shift of basic setpoint value permanetly	**No** Yes	By the Shift of basic set value (trough display buttons), the temperature of the current operating mode of both control circuits can be adapted. You can shift the temperature up or down within a preset value range. Setting "no" will delete the temperature shift when a change to the new operating mode takes place. Setting "yes" will keep the temperature shift when a change into a different operating mode takes place.

Modification of the basic temperature setpoint value	**Deactivated** Permit via display buttons Permit via object Permit via display buttons and bus	To specify whether a basic temperature change of the first control circuit will be possible via the bus or locally on the device.
Modification of the basic temperature setpoint in the 2nd control circuit	**Deactivated** Permit via object	To specify whether a basic temperature change of the second control circuit will be possible via the bus or locally on the device. Only for "own set values for second control circuit = yes".
Accept modification of the basic temperature setpoint value permanetly	**No** Yes	This parameter specifies whether the basic temperature readjusted via the bus or locally on the device shall be stored in the memory permanently (setting *"yes"*) or only temporarily (setting *"no"*). If you select "yes" the changed basic value will be kept, even after an operating mode switch-over and a reset. Only for "basic temperature set value change = permit via display buttons", "permit via object" or "permit via display buttons and bus".
Modification of the setpoints "cooling"	**Deactivated** Permit via display buttons	To facilitate the changing of the cooling set values of the first control circuit on the device in the mixed mode. When the changing option is deactivated the "cooling temperatures" menu will not be accessible in the programming mode. Only for mixed "heating/cooling switch-over" = "heating and cooling" with additional stage, if necessary.
1st control circuit standby temperature change	**Deactivated** Permit via display buttons	To facilitate the changing of the basic standby temperature of the first control circuit on the device.
1st control circuit night temperature change	**Deactivated** Permit via display buttons	To facilitate the changing of the night temperature of the first control circuit on the device.
Frost protection setpoint temperature (7...40) * 1 °C	7 °C to 40 °C, **7 °C**	To specify the set value temperature when frost protection is active. Only for "heating/cooling switch-over" = "heating" or "heating and cooling", with additional stage, if necessary.
Heat protection setpoint temperature (7...45) * 1 °C	7 °C to 45 °C, **35 °C**	To specify the set value temperature when heat protection is active. Only for "heating/cooling switch-over" = "cooling" or "heating and cooling", with additional stage, if necessary.

Dead band position	**Symmetrical** Asymmetrical	The comfort heating and cooling set value temperatures are derived from the basic set value, taking account of the preset dead band. The dead band (temperature zone where neither heating nor cooling takes place) is the difference between the comfort set value temperatures. Symmetrical: The preset dead band is divided into two ranges at the basic set value. From the resulting half of the dead band, the comfort set value temperatures are directly derived from the basic set value. Asymmetrical: For this setting, the comfort heating set value temperature is equal to the basic set value. The preset dead band will solely become effective from the basic set value towards the comfort cooling temperature. Thus, the comfort mode cooling set value temperature is directly derived from the comfort heating set value. Only for the "heating and cooling" or "basic/ additional heating/cooling" mixed mode.
Dead band between heating and cooling (0...127) * 0.1 K	0 to 127, **20**	The comfort heating and cooling set value temperatures are derived from the basic set value, taking account of the preset dead band. The dead band (temperature zone where neither heating nor cooling takes place) is the difference between the comfort set value temperatures. Only for the "heating and cooling" or "basic/ additional heating/cooling" mixed mode.
Dead band shift	**Deactivated** Permit via display buttons	To specify whether the dead band and thus the comfort cooling temperature can be set on the device in the programming mode from the "cooling temperature values" submenu. Only for the "heating and cooling" or "basic/ additional heating/cooling" mixed mode, and for "cooling set values change" = permit via display buttons.

Difference between basic and additional stages (0…127) * 0.1 K	0 to 127, **20**	For two-stage control operation, you must specify at what temperature difference compared to the basic stage you want to include the additional stage into control. Only for two-stage control operation.
Transmission at setpoint temperature modification by (0...255) * 0.1 K; 0 = no autom. transmission	0 to 255, **1**	To specify the set value change value by what the current value will be automatically sent to the bus via the *"setpoint temperature"* object. If you use both control circuits with separate set values both set values can be transmitted.
Cyclical transmission of setpoint temperature (0…255) * 1 min; 0 = inactive	0 to 255, **0**	To specify whether the set value temperature shall be output via the *"setpoint temperature"* object. If you use both control circuits with separate set values both set values can be transmitted.
Upward adjustment of basic setpoint temperature (0...10) * 1 K	0 to 10, **3**	To specify the maximum adjusting value range you can set for shifting the basic set value temperature upward. (Refer to "Basic temperature set value change".)
Downward adjustment of basic setpoint temperature (-10...0) * 1 K	-10 to 0, **-3**	To specify the maximum adjusting value range you can set for shifting the basic set value temperature downward. (Refer to "Basic temperature set value change".)
Lower the setpoint temperature during standby operation (heating) (-128...0) * 0.1 K	-128 to 0, **-20**	To decrease the standby heating set value temperature by this value, compared with the basic set value. Only for "heating/cooling switch-over" = "heating" or "heating and cooling", with additional stages, if necessary.
Lower the setpoint temperature during night operation (heating) (-128...0) * 0.1 K	-128 ... 0, **-40**	To decrease the night heating set value temperature by this value, compared with the basic set value. Only for "heating/cooling switch-over" = "heating" or "heating and cooling", with additional stages, if necessary.

Lower the setpoint temperature during standby operation (heating - control circuit 2) (-128...0) * 0.1 K	-128 to 0, **-20**	To decrease the standby heating set value temperature by this value, compared with the basic set value. Only for "heating/cooling switch-over = heating" and two control circuits with separate set values.
Lower the setpoint temperature during night operation (heating - control circuit 2) (-128...0) * 0.1 K	-128 to 0, **-40**	To decrease the night heating set value temperature by this value, compared with the basic set value. Only for "heating/cooling switch-over = heating" and two control circuits with separate set values.
Raise the setpoint temperature during standby operation (cooling) (0...127) * 0.1 K	0 to 127, **20**	To increase the standby cooling set value temperature by this value, compared with the basic set value. Only for "heating/cooling switch-over" = "cooling" or "heating and cooling", with additional stages, if necessary.
Raise the setpoint temperature during night operation (cooling) (0...127) * 0.1 K	0 to 127, **40**	To increase the night cooling set value temperature by this value, compared with the basic set value. Only for "heating/cooling switch-over" = "cooling" or "heating and cooling", with additional stages, if necessary.
Raise the setpoint temperature during standby operation (cooling - control circuit 2) (0...127) * 0.1 K	0 to 127, **20**	To increase the standby cooling set value temperature by this value, compared with the basic set value. Only for "heating/cooling switch-over = cooling" and two control circuits with separate set values.
Raise the setpoint temperature during night operation (cooling - control circuit 2) (0...127) * 0.1 K	0 to 127, **40**	To increase the night cooling set value temperature by this value, compared with the basic set value. Only for "heating/cooling switch-over = cooling" and two control circuits with separate set values.
Room temperature regulator function - functionality		
Operation mode after reset	Restore operating mode before reset **Comfort operation** Standby operation Night operation Frost/heat protection operation	Parameter to set the operating mode to be activated after the push button initialization phase. Frequent changing of the operating mode (several times a day) when the "operating mode before reset" parameter is being set can adversely affect the life of the device as the non-volatile memory used has been designed for the storage of permanent values only.

Presence detection		When a person is present in the room it makes sense that the regulator switches over into the comfort mode or into the comfort prolongation mode, respectively. This parameter specifies though which 'detectors' detection will be made.
	None	No presence detection.
	Via object	Presence is detected via a separate object or through a presence button on the push button (push button function).
Type of presence detection	Presence button	Presence will be detected by a presence button on the push button (button function) or via the presence object (e. g. external push button). If presence is detected comfort prolongation or the comfort mode will be activated.
	Presence detector	Presence will be detected via an external presence detector. The detector will be connected via the presence object. If presence is detected the comfort mode will be called as long as the presence detector detects some movement. Only for "presence detection = via object".
Length of comfort prolongation (0...255) * 1 min; 0 = OFF	0 to 255, **30**	During presence detection, the regulator can temporarily switch over to comfort prolongation, depending on the active operating mode. The parameter specifies the time after which the comfort prolongation will be automatically terminated. Only for "type of presence detection = presence button".
Room temperature regulator function - room temperature measuring (HA)		
Temperature detection		To specify which sensor will be used for room temperature measuring in the first control circuit.
	Internal sensor	Internal sensor: Built-in sensor of the B.IQ RTR push button.
	External sensor	External sensor: An external sensor connected via the bus, e. g. under difficult measuring conditions (in swimming baths or similar facilities).
	Internal and external sensor	Internal and external sensors: Both sensors are used, e. g. in large rooms. Only for one control circuit.

Creating of measuring value internal against external	10 % vs. 90 % 20 % vs. 80 % 30 % vs. 70 % 40 % vs. 60 % **50 % vs. 50 %** 60 % vs. 40 % 70 % vs. 30 % 80 % vs. 20 % 90 % vs. 10 %	To specify the weighting of the measured temperature value of the internal and external sensor. This will form a resulting overall measured value used for further evaluation of the room temperature. Only for one control circuit and "temperature detection = internal and external sensor".
Adjustment internal sensor (-128...127) * 0.1 K	-128 to 127, **0**	To specify the value by which the measured room temperature value of the internal sensor is adjusted. Only for "temperature detection = internal sensor" or "internal and external sensor", or for two control circuits.
Adjustment external sensor (-128...127) * 0.1 K	-128 to 127, **0**	To specify the value by which the measured room temperature value of the external sensor is adjusted. Only for "temperature detection = external sensor" or "internal and external sensor", or for two control circuits.
Scaning time for external sensor (0…255) * 1 min; 0 = inactive	0 to 255, **0**	To specify the temperature value scanning time of the external sensor. "0" = the sensor will automatically send its temperature value. Only for "temperature detection = external sensor" or "internal and external sensor", or for two control circuits.
Transmission at room temperature modification by (0..255) * 0.1 K; 0 = no automatic transmission	0 to 255, **3**	To specify the room temperature value change extent after which the current values will be automatically sent to the bus via object 23, "actual temperature".
Cyclical transmission of room temperature (0…255) * 1 min; 0 = inactive	0 to 255, **15**	To specify whether or at what intervals the measured room temperature of the first control circuit shall be output via object 23, "actual temperature".

Room temperature regulator function - set value and status output (HA)		
Automatic transmission at modification by (0...100) * 1 %; 0 = inactive	0 to 100, **3**	To specify the variable change extent after which the continuous variables will be automatically transmitted via the variables objects. Only if at least one type of control has been parameterized to "continuous PI control".
Cycle time of the switching variable (1...255) * 1 min	1 to 255, **15**	To specify the cycle time for the pulse-width-modulated (PWM) variable. Only if at least one type of control has been parameterized to "switching PI control (PWM)".
Cycle time for automatic transmission (0…255) * 1 min; 0 = inactive	0 to 255, **10**	Time interval for the cyclic transmission of the variable via the variables objects. Only if at least one type of control has been parameterized to "continuous PI control" or to "switching 2-point control".
Output of the heating variable	Inverted **Normal**	Continuous: variable = 100 % - normal variable Switching: variable = 1 - normal variable Heating normal variable output Only for "heating/cooling mode" = "heating" or "heating and cooling".
Output of the heating variable in the 2nd control circuit	Inverted **Normal**	Continuous: variable = 100 % - normal variable Switching: variable = 1 - normal variable Control circuit 2 heating normal variable output Only for "mode of the control ciruits = heating" with two control circuits.
Output of the heating basic stage variable	Inverted **Normal**	Continuous: variable = 100 % - normal variable Switching: variable = 1 - normal variable Heating basic stage normal variable output Only for "heating/cooling mode = basic and additional heating" or "basic/additional heating/cooling".
Output of the heating additional stage variable	Inverted **Normal**	Continuous: variable = 100 % - normal variable Switching: variable = 1 - normal variable Heating additional stage normal variable output Only for "heating/cooling mode = basic and additional heating" or "basic/additional heating/cooling".

Output of the cooling variable	Inverted **Normal**	Continuous: variable = 100 % - normal variable Switching: variable = 1 - normal variable Cooling normal variable output Only for "heating/cooling mode" = "cooling" or "heating and cooling".
Output of the cooling variable in the 2nd control circuit	Inverted **Normal**	Continuous: variable = 100 % - normal variable Switching: variable = 1 - normal variable Control circuit 2 cooling normal variable output Only for "mode of the control ciruits = cooling" with two control circuits.
Output of the cooling basic stage variable	Inverted **Normal**	Continuous: variable = 100 % - normal variable Switching: variable = 1 - normal variable Cooling basic stage normal variable output Only for "heating/cooling mode = basic and additional cooling" or "basic/additional heating/cooling".
Output of the cooling additional stage variable	Inverted **Normal**	Continuous: variable = 100 % - normal variable Switching: variable = 1 - normal variable Cooling additional stage normal variable output Only for "heating/cooling mode = basic and additional cooling" or "basic/additional heating/cooling".
Heating indication	**No** Yes	To enable the "heating" indicator function and thus object 37, "heating indicator". The signal is exclusively referred to the first control circuit.
Cooling indication	**No** Yes	To enable the "cooling" indicator function and thus object 38, "cooling indicator". The signal is exclusively referred to the first control circuit.
Status indication of controller		The regulator can output its current status.
	No status	No status output.
	Controller general	The regulator status is generally output via the 1 byte object (object 36, *"controller status"*).
	Transmit individual state	The regulator status preset by the *"individual status"* parameter will be output via the 1 bit object (object 36, *"controller status"*).

Single state	**Comfort mode active** Standby mode active Night mode active Frost/heat protection mode active Regulator disabled Heating/cooling mode Regulator inactive Frost alarm	To specify the regulator status to be transmitted. Only for "regulator status" = "transmit individual status".
Room temperature regulator functions - room temperature timer		
Room temperature timer	ON **OFF**	To enable the room temperature timer.
Lock room temperature timer via object		The execution of the room temperature timer switching programs can be suppressed via the bus by the disabling function. To enable the disabling function and object 5, *"disabling room temperature timer"*. The room temperature timer disabling function is not deactivated.
	Yes **No**	
Polarity of blocking object	Inverted (disabling = 0) **Not inverted (disabling = 1)**	To specify the room temperature timer disabling object polarity. Only for "lock room temperature timer via object = yes".

Scene function		
Data type Output 1	**Switching** Value Shutter / blind position	To specify the data type of the scene output.
Value Type	**0...100 %** 0...255	To define the value type for a 1 byte scene object. Depending on this setting, you can either set percentages or dimensionless values for the scene commands. Only for "data type = value".
Data type Outputs 2 to 8	Refer to output 1 data type.	
Scene function - [1] scene 1		
Name	[Text], **scene 1**	Here, you can assign a name to the internal scene. This text will only be used in the ETS plug-in for better orientation and will not be downloaded into the device.
Transmit output signal	Yes **No**	To specify whether a scene command shall be transmitted via the selected scene output when the scene is being recalled.
Value	**ON** OFF	To define the switching value which will be sent to the bus when a scene is being re-called. Only for "transmit output signal = yes" and "data type = switching".
Value (0…100) * 1 %	0 to 100 %, **0 %**	To define the value which will be sent to the bus when a scene is being recalled. Only for "transmit output signal = yes", "data type = value" and "value type = 0...100 %".
Value (0…255)	0 to 255, **0**	To define the value which will be sent to the bus when a scene is being recalled. Only for "transmit output signal = yes", "data type = value" and "value type = 0...255".
Shutter / blind position (0...100) * 1 % (0 => top)	0 to 100 %, **0 %**	To define the Shutter position value which will be sent to the bus when a scene is being recalled. Only for "transmit output signal = yes" and "data type = Shutter / blind position".
Scene function - [X] scene X, for X = 2 to 8, refer to scene 1.		

Timer 1		
Name	[20-character text], **timer 1**	Here, you can assign a name timer 1. This name will also be downloaded into the device and indicated in the programming menu.
Data format	**Switching** Value Scene request	To specify whether switching or value or a scene recall commands shall be sent to the bus when the timer is active. You can define the commands themselves in the timer editor.
Blocking object (e. g. sun sensor)		The execution of the switching times of timer 1 can be suppressed via the bus by the disabling function.
	Yes	To enable the disabling function and object 55, *"disabling timer 1"*.
	No	The timer 1 disabling function is not enabled.
Polarity of blocking object	Inverted (disabling = 0) **Not inverted (disabling = 1)**	To specify the timer 1 disabling object polarity. Only for "blocking object = "yes".
For timer 2, refer to timer 1.		

Remarks on the Software

- **Parameter access**

  To be able to set all parameters of the B.IQ push button RTR you must have set the access option in the ETS plug-in to "high access" (HA). To set the type of access select or deselect the "high access" menu item in the "configuration" menu.

- **Dimming function (push button functionality)**

  For the correct function of the status-LEDs as Status indications the dimming actuator connected must send back its status to the switching object when the button function is active or to the status object when the rocker function is active (set the "T" flag on the actuator).
  For correct functioning in the button function mode (brighter/darker (TOGGLE)), the dimming actuator connected must also send back its status to the switching object.
  For the push button or rocker function, only the switching object will be followed up internally and externally. The dimming object (dimming direction) will be followed up only internally so that the dimming direction will not always be reversed upon another pressing of the button if extensions are used (two or more push buttons to dim one lamp).
  For dual-button operation in connection with the button function, the objects of the associated buttons must be given the same group address.

- **Value transmitter function (push button functionality)**

  When you change values by move button actuation the newly set values will only be saved in the RAM, i. e. these values will be replaced by those presettings as originally programmed through the ETS after a voltage failure or a bus reset.

- **Status indication (push button functionality)**

  The status-LEDs (in the status indication mode) indicate the instantaneous status of the switching object when the push button function is active. If you actuate a push button (e. g. ON) and the push button sensor does not receive a positive acknowledgement (IACK) from an actuator addressed the object status will be updated, with the corresponding status-LED lighting up.

- **ETS plug-in system requirements**

Operating system:	Windows 9x, ME, NT 4.0, 2000, XP
ETS:	ETS 2 v 1.2 a or later, ETS 2 v 1.3 a recommended.
PC:	Pentium I processor (or similar), 166 MHz, 32 MB or better recommended.

- **ETS functions**

  The "device info" or "device memory viewer" ETS functions are not possible for the B.IQ RTR push button.

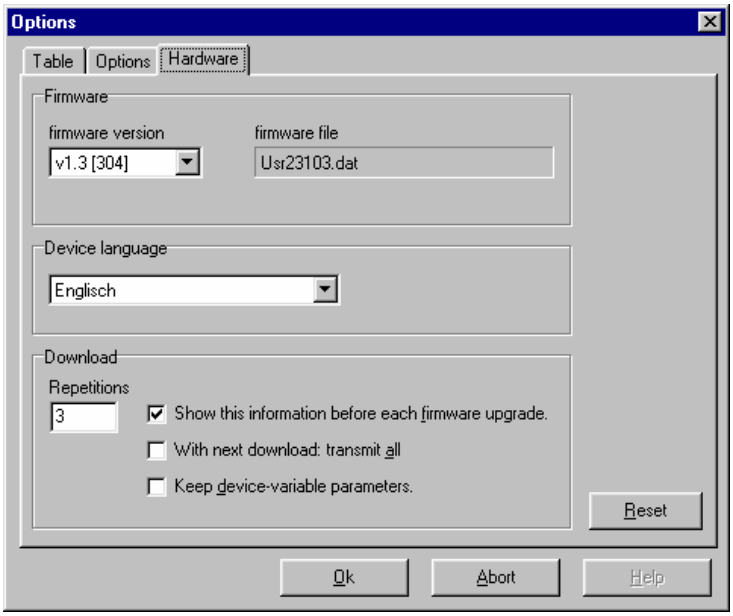
  Even executing the "shrink database" ETS function will lead to project data corruption for the B.IQ push button RTR up to including version v 1.3 and should be avoided under any circumstances. Install ServiceRelease "a" for the ETS2 v 1.3 as a remedy.

- **Firmware**

The B.IQ push button RTR offers the possibility to update the application software in the device. The ETS plug-in facilitates this firmware download by loading the data into the application module via the bus.
In this way, you can even bring earlier B.IQ RTR push buttons to the latest standard without having to replace the device. Only the B.IQ push button RTR software in the ETS must be up-to-date in this connection. Normally, a firmware download will only be necessary if you want to update an earlier device.
For 'normal' application data programming procedures, it will not be necessary to transfer the firmware.
Even for the first start-up, the firmware is already factory-pre-programmed in the push button.

A firmware download takes a few minutes. During a firmware download, the display reads the message **"Firmware download - loading….."**.

If a firmware download should become necessary you must call the "options" menu item from the "settings" menu in the ETS plug-in. This will open the options dialog. On the "hardware" tab, you can preset the following parameters:



- Device language: The language (text read-outs in the display) of the B.IQ push button RTR device can be parametrized

- Firmware version: This selection box lists the firmware versions known to the software. Here, you should always select the most current version (highest number). New firmware versions will, in the future, be provided by a separate software update.

- The firmware download will be started together with the application download. To ensure the loading of the firmware into the device during the next programming process check the "with next download: transmit all" box.

- Prior to a programming process, the software will automatically detect whether the firmware existing in the device corresponds to the version specified by the software. If this is not the case the software will offer a firmware upgrade or downgrade in the form of a dialog. If you uncheck the "show this information before each firmware upgrade" box this message will no longer appear, even though you are programming further B.IQ push button RTR versions containing 'inappropriate firmware.
  Later, you can reactivate this check box in the ETS plug-in from the options dialog on the "hardware" tab.

- If you have checked the "keep device-variable parameters" box the temperature set values (decrease/increase/standby, night, dead band, basic set value) of the first control circuit locally changeable on the device or via the bus will not be replaced by the values parameterized in the ETS plug-in during a download.