Product manual

ABB i-bus® KNX
Room thermostat
Fan Coil
RTF/A 1.1

Systemised building technology

ABB

1 Function properties

The RTR-Fan Coil RTF/A 1.1 is a continuous KNX room thermostat for fan convectors (Fan Coil) in 2 and 4 pipe systems.
It measures the current room temperature (actual value) and sends a continuous correcting variable (0...100%), e.g. to an ABB i-bus® Fan Coil actuator to reach the required room temperature (set value).

The RTF/A 1.1 works both in heating mode and in cooling mode.
The fan stage can also be selected manually via a button.

Three binary inputs (see external interface) can be used to connect switches or buttons (floating) for switching, dimming or controlling venetian blinds.
The blind and dimmer channels can also be operated using a single button if preferred (one-panel operation)
Alternatively, an external temperature sensor can be connected to input 3 (analogue).

To permit easy adjustment of the set values to meet the requirements for home comfort and energy-saving, the RTF/A 1.1 supports four operating modes:

- Comfort
- Standby
- Night mode
- Antifreeze mode

A set value is assigned to each operating mode.

The **comfort mode** is used if there are persons in the room

In **standby mode** the set value is reduced slightly. This operating mode is used if the room is not occupied but is expected to be occupied soon.

In **night mode** the set value is reduced further as the room is not expected to be used for several hours.

In **antifreeze mode**, the room is regulated at a temperature that prevents damage to the radiators due to freezing if the outside temperatures are very low.
This can be desirable for 2 reasons:
- The room will not be occupied for several days.
- A window has been opened and no heating is required temporarily.

The operating modes are generally controlled using a time switch.
For optimum control, however, presence watchdogs or presence buttons and window contacts are recommended.
See also chapter Calculating the set value.

1.1 Operation

For operation and display, the RTF/A 1.1 is equipped with a setting wheel and 5 LEDs for displaying the current fan level.
The fan level can also be set manually using the button provided to the right of the LEDs (override mode).

1.2 The device LEDs

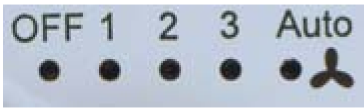


Table 1

LED	Indicator	Description
Auto	Fan is in automatic mode	Fan level is controlled based on the set value, as per parameter settings. See parameter page *Operation*.
0	Fan level 0 = fan is off.	Override mode: Fan level is selected manually by pressing the button.
1	Fan level 1	
2	Fan level 2	
3	Fan level 3	

The setting wheel can be used either for the **setting** set value or the**shift** set value depending on the parameter settings.

1.3 Benefits of the RTF/A 1.1

- Continuous PI room thermostat
- Manual pre-selection of the fan level if necessary
- Changing the operating mode via presence and window objects
- Heating and cooling modes
- Setting wheel for setting or moving set values
- Infinite control system via continuous setting value
- 3 binary inputs for conventional actuators for controlling switches, dimmers and blinds
- Adjustable direction of control of the binary inputs
- Blinds and dimmers can also be controlled using one-panel operation

1.3.1 Special features

The RTF/A 1.1 has 3 external inputs for buttons, switches or an external sensor. This permits the control of switch, dimmer or blind actuators.

2 Technical data

Power supply:	Bus voltage
Permissible operating temperature:	0°C ...+ 50°C
Protective class:	III
Protection:	EN 60529: IP 21
Dimensions:	HxWxD 80x84x28 (mm)

Inputs:

Quantity:	3
Contact voltage:	3.3 V provided internally
Contact current:	1 mA
Maximum cable length:	5 m

3 The application program

3.1 Selection in the product database

Manufacturer	ABB AG
Product range	Heating, air conditioning, ventilation
Product type	Fan coil controller
Program name	FanCoil operation and control / 1.2

The ETS database can be found on our website: www.abb.de/knx

3.2 Parameter pages

Function	Description
Adjustments	Device type and activation of the external interface.
Set values	Set value after download, values for night/frost operation, etc.
Cooling set values	Dead zone and temperature increases due to operating mode
Operation	Function of the setting wheel and the button.
Actual value	Type/function of the sensor, calibration
Control system	System type, heating/cooling parameters, etc.
Operating mode	Operating mode after reset, presence sensor
Inputs E1...E3	Function of the connected contact, switches, dimmers, blinds.

3.3 *Communication objects*

3.3.1 Features of the objects

The RTF/A 1.1 has 12 communication objects.
Some objects can carry out different functions depending on the parameter settings.

No.	Function	Object name	Type	Flags			
				K	L	S	U
0	*Set required temperature*	*Basic set value*	2 byte EIS5	✓	✓	✓	
	shift	*Manual set value shift*	2 byte EIS5	✓	✓	✓	
1	*monitor current set value*	*current set value*	2 byte EIS5	✓	✓		✓
2	*Transmit actual value*	*Actual value*	2 byte EIS5	✓	✓		✓
3	*Preselect the operating mode*	*Operating mode selection*	1 byte KNX	✓	✓	✓	
	1 = night, 0 = standby	*Night < - > standby*	1 bit EIS1	✓	✓	✓	
4	*Input for presence signal*	*Presence*	1 bit EIS1	✓	✓	✓	
	1 = Comfort	*Comfort*	1 bit EIS1	✓	✓	✓	
5	*Input for window status*	*Window position*	1 bit EIS1	✓	✓	✓	
	1 = antifreeze	*Frost/heat protection*	1 bit EIS1	✓	✓	✓	
6	*issue current operating mode*	*current operating mode*	1 byte KNX DTP	✓	✓		✓
7	*Send current control value*	*Heating control value*	1 byte EIS6	✓			✓
	Send current control value	*Heating and cooling control value*	1 byte EIS6	✓			✓
8	*Send control value*	*Cooling control value*	1 byte EIS6	✓			✓
				K	L	S	U

No.	Function	Object name	Type	Flags			
				K	L	S	U
9	*Send switch telegram*	*Switch input 1*	1 bit EIS1	✓	✓	✓	✓
	Send ON/OFF telegram	*Dim E1 On/Off*		✓	✓	✓	✓
	Send ON/OFF telegram	*Dim E1/E2 On/Off*		✓	✓	✓	✓
	Slat	*Blind E1 Step/Stop*		✓	✓		✓
	Slat	*Blind E1/E2 Step/Stop*		✓	✓		✓
10	*Send up/down telegram*	*Blind E1 up/down*	1 bit EIS1	✓	✓		✓
	Send dim telegram	*Dim E1*	4 bit EIS2	✓	✓		✓
11	*Send switch telegram*	*Switch input 2*	1 bit EIS 1	✓	✓	✓	✓
	Send ON/OFF telegram	*Dim E2 On/Off*		✓	✓	✓	✓
	Slat	*Blind E2 Step/Stop*		✓	✓		✓
12	*Blind E2 up/down*	*Send up/down telegram*	1 bit EIS 1	✓	✓		✓
	Blind E1/E2 up/down	*Send up/down telegram*		✓	✓		✓
	Dim E2	*Send dim telegram*	4 bit EIS2	✓	✓		✓
	Dim E1/E2	*Send dim telegram*		✓	✓		✓
13	*Send switch telegram*	*Switch input 3*	1 bit EIS1	✓	✓	✓	✓
	Send ON/OFF telegram	*Dim E3 On/Off*		✓	✓	✓	✓
	Slat	*Blind E3 Step/Stop*		✓	✓		✓
14	*Blind E2 up/down*	*Send up/down telegram*	1 bit EIS1	✓	✓		✓
	Dim E3	*Send dim telegram*	4 bit EIS2	✓	✓		✓
15	*Heating = 0, cooling = 1*	*Switchover betw. heating and cooling*	1 bit EIS1	✓	✓	✓	
16	*send/receive*	*Fan level in override mode*	1 byte EIS 6	✓	✓	✓	✓
17	*0 = auto / 1 = override*	*Override / auto fan*	1 bit EIS1	✓	✓	✓	✓
				K	L	S	U

The communication flags

Flag	Name	Meaning
K	Communication	Object is communication-compatible
L	Read	Object status can be queried (ETS / display, etc.)
S	Write	Object can receive
U	Send	Object can send

Number of communication objects	18
Number of group addresses	34
Number of assignments	35

3.3.2 Description of the objects

- **Object 0 "*Basic set value" / "Manual set value shift*"**

This object can carry out 2 different functions.
This means that, depending on the parameter settings on the setting wheel, either a new set temperature can be specified or the current set temperature can be shifted by a specific value

Parameter: *Function of the setting wheel*	Function of the object
Manual shift for internal controller *blocked, but basic object set value present*	**Set required temperature:** The basic set value is specified for the first time during commissioning via the application and stored in the *basic set value* object. It can then be re-set at any time via object 0 (limited by the minimum or maximum valid set value). In the event of a bus voltage failure, this object is protected; when the bus voltage returns, the last value is restored. The object can be written to any number of times.
Basic set value for internal controller *blocked, but man. shift object present*	Shift set temperature: The object receives a temperature difference in EIS 5 format. This difference can be used to change the required room temperature (current set value) from the basic set value. In comfort mode (heating), the following applies: current set value (obj. 1) = basic set value (setting wheel) + manual set value shift (obj. 0) Values outside the parameterised range (see *Max. set value shift on the setting wheel*) are limited to the maximum or minimum value. Note: The shift always relates to the basic set value and not the current set value.
Manual shift with message object	Object 0 sends the shift set on the setting wheel on a Fan Coil actuator.

- **Object 1 *"current set value"***

This object sends the current set temperature as EIS 5 telegram (2 byte) to the bus.
The set process can be set on the parameter page *Set values*.

- **Object 2 *"actual value"***

This object sends the temperature currently measured by the sensor (if sending is permitted by the parameter settings)

- **Object 3 *"Operating mode preselection" / "Night <-> Standby"***

The function of this object is dependent on the *Objects for setting the operating mode* parameter on the parameter page *Operating mode*.

Objects for setting the operating mode	Function of the object
new: Operating mode, presence, window status	1 byte object for selecting one of 4 operating modes. 1 = comfort, 2 = standby, 3 = night, 4 = Frost protection (heat protection) If a different value is received (0 or >4), the comfort operating value is activated. The details in brackets relate to cooling operation
old: Comfort, Night, Frost	With this setting, this object is a 1Bit object. This means that the Night or Standby operating mode can be activated 0=Standby 1=Night

- **Object 4 *"Presence" / "Comfort"***

The function of this object is dependent on the *Objects for setting the operating mode* parameter on the parameter page *operating mode*.

Objects for setting the operating mode	Function of the object
new: Operating mode, presence, window status	**Presence:** This object can be used to receive the status of a presence watchdog (e.g. button, movement sensor). A 1 to this object activates the Comfort operating mode.
old: Comfort, Night, Frost	**Comfort:** A 1 to this object activates the Comfort operating mode. This operating mode has priority over night and standby modes. The comfort mode is deactivated again by sending a 0 to the object.

- **Object 5 "Window position" / "Frost/heat protection"**

The function of this object is dependent on the *Objects for setting the operating mode* parameter on the parameter page <u>*operating mode*</u>.

Objects for setting the operating mode	Function of the object
<u>*new: Operating mode, presence, window status*</u>	**Window position:** This object can be used to receive the status of a window contact. A 1 on this object activates the frost/heat protection operating mode.
<u>*old: Comfort, Night, Frost*</u>	**Frost/heat protection:** A 1 to this object activates the Frost protection operating mode. During cooling operation, the heat protection operating mode is activated. The frost/heat protection operating mode has the highest priority. The frost/heat protection operation flashes when in force until it is cancelled again by a 0.

- **Object 6 "current operating mode"**

Sends the current operating mode as a 1 byte value (see below: Coding of the operating modes).
The send behaviour can be set on the *Operating mode* parameter page.

Coding of the HKL (HVAC) operating modes:

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

- **Object 7 *"Heating control value", heating and cooling control value***

Sends the current heating control value (0...100%) or heating or cooling with 2-pipe system. See *Fan Coil system used* parameter on the *Control* parameter page.

- **Object 8 *"Cooling control value"***

Sends the cooling control value in EIS 6 format

- **Objects 9, 10, 11, 12, 13, 14 for the inputs *E1, E2* and *E3***

These objects are available when the interface on the *Settings* parameter page is activated.

Its function is dependent on the parameters *Function of E1*, *Function of E2* and *Function of E3* on the relevant parameter pages (inputs E1, E2 and E3).

A comprehensive description is given in the appendix in the chapter: External interface.

- **Object 15 *"Switching between heating and cooling"***

This object is used with 2 pipe heating/cooling systems or if an automatic switchover between heating and cooling is not required.
The cooling operation is switched on with a 1 and heating operation is switched on with a 0.

- **Object 16 *"Fan level in override mode"***

The fan level can be set manually by pressing the button on the unit.
This object then sends a percentage corresponding to the parameterised threshold values.

This function both be blocked via a parameter and also be time-limited or permanently switched.

See parameter page *Operation* and in the appendix: Fan override mode .

- **Object 17 *"Fan override/auto"***

Sends if an override fan level is selected via the button.
This means that a fan coil actuator can be put into override operation.

Depending on the application, the override mode is triggered either by a 0 or a 1.
→ See parameter *Switch fan between auto and override* on parameter page *Operation.*
When returning to automatic mode, the status of the object is inverted again.

3.4 Parameters

The default values are all printed in **bold**.

3.4.1 Settings

Designation	Values	Meaning
Device type	**RTF/A 1.1**	Fixed setting
Function of the external interface	**none** active	Specifies whether or not the external interface is used.

3.4.2 Set values

Designation	Values	Meaning
Basic stt value after downloading the application	*18 °C, 19 °C, 20 °C,* ***21 °C****, 22 °C, 23 °C, 24 °C, 25 °C*	Output set value for temperature control.
Minimum basic set value	*5°C,* ***6°C****, 7°C, 8°C, 9°C, 10°C, 11°C, 12 °C, 13°C, 14°C, 15°C,16°C 17°C, 18°C, 19 °C, 20 °C*	If a basic set value is received on object 0 which is lower than the value set here, it is limited to that value.
Maximum valid basic set value	*20°C, 21°C, 22°C 23°C, 24 °C, 25°C 27 °C, 30 °C,* ***32 °C***	If a basic set value is received on object 0 which is higher than the value set here, it is limited to that value.
Lowering in standby mode (for heating)	*0.5 K, 1 K, 1.5 K* ***2 K****, 2.5 K, 3 K 3.5 K, 4 K*	Example: with a basic set value of 21°C in heating mode and a 2K drop, RTF/A 1.1 regulates at a set value of 21 – 2 = 19°C.
Lowering in night mode (for heating)	*3 K, 4 K,* ***5 K*** *6 K, 7 K, 8 K*	By how much is the temperature to be reduced in night mode?
Set value for frost protection mode (for heating)	*3 °C, 4 °C, 5 °C,* ***6°C****, 7 °C, 8 °C 9 °C, 10 °C*	Temperature specification for frost protection mode in heating mode (The heat protection mode applies in cooling mode).
Set value shift applies	*only in comfort mode* *with comfort and standby mode* ***with comfort, standby and night mode***	In which operating modes is the set value shift to be effective? This setting relates both to the shift via bus telegram and via the setting wheel.

Designation	Values	Meaning
current set value in comfort mode	 ***Send actual value (heating < > cooling)*** *Send mean value betw. heating and cooling*	Feedback of the current set value via the bus: the set value to which the temperature is to be controlled should always be sent (= <u>curreInt stt value</u>). **Example** with basic stt value 21°C and <u>*Dead zone*</u> 2K: With heating, 21°C is sent, and with cooling, the basic set value + dead zone is sent (21°C + 2K = 23°C) In the comfort operating in heating mode and in cooling mode, the same value is sent, namely basic set value + half dead zone, so that room users are not irritated where appropriate. **Example** with basic set value 21°C and deadzone 2K: Mean value = 21°+1K =22°C The regulated value is 21°C or 23°C
cycl. sending of the current set value	 ***not cyclical, only with a change*** *every 2 mins.* *every 3 mins.* *every 5 mins.* *every 10 mins.* *every 15 mins.* *every 20 mins.* *every 30 mins.* *every 45 mins.* *every 60 mins.*	How often is the current valid set value to be sent? only send with a change. send cyclically

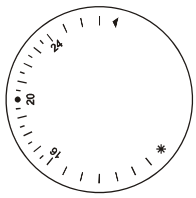
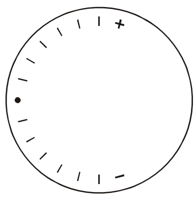
3.4.3 Cooling set values

This page only appears if on the parameter page *Settings* the control function *Heating and cooling* has been selected (*userdefined* control).

Table 2

Designation	Values	Meaning
Dead zone between heating and cooling	*1 K* ***2 K*** *3 K* *4 K* *5 K* *6 K*	Specifies the buffer zone between the set values for heating and in cooling mode. With a switching (2-point) control, the dead zone is increased via the hysteresis. See in glossary: <u>Dead zone</u>
Raising in standby mode (for cooling)	*0.5 K, 1 K, 1.5 K* ***2 K**, 2.5 K, 3 K* *3.5 K, 4 K*	In cooling mode, the temperature is raised in standby
Raising in night mode (for cooling)	*3 K, 4 K,* ***5 K*** *6 K, 7 K, 8 K*	see Raising in standby mode
Set value for heat protection mode (for cooling)	***42 °C (i.e. roughly equates to no heat protection)*** *29 °C, 30 °C, 31 °C, 32 °C, 33 °C, 34 °C, 35 °C*	The heat protection represents the maximum temperature permitted for the controlled area. With cooling it fulfils the same task as the frost protection mode during heating, i.e. save energy and at the same time prevent non-permissible temperatures.

3.4.4 *Operation*

Designation	*Values*	Meaning
Function of the setting wheel	*Basic set value for internal controller* *(please use the following setting wheel)* 	The setting is used to specify the basic setl value. A set value shifit is possible via object 0 The setting wheel with numbers is affixed to the device.
	Manual shift for internal controller *(please use the following setting wheel)* 	The basic set value can be increased or reduced via the setting wheel within the parameterised limits (see table below). The +/- setting wheel is affixed to the device.
	Blocked, but basic object set value present	The setting wheel has no function (protection against unwanted operation) The basic set value can be changed in the application or via *object 0*.
	Blocked, but object man.shift present	The setting wheel has no function (protection against unwanted operation) The basic set value is changed in the application and can be increased or reduced via object 0.
	Manual shift with message object	If the fan coil actuator has its own control unit, the RTF/A 1.1 can send a set value shift to the controller via object 0.

Designation	Values	Meaning
Max. set value shift on the setting wheel	*+/- 1 K,* ***+/- 2 K****, +/- 3 K* *+/- 4 K, +/- 5 K,*	Limits the possible setting range for the *Set value shift* function. Applies to the values received via object 0 (*manual set value shift*).
Actuation of the LEDs	*always off* ***always active*** *Time limit active*	The LEDs are not used In automatic mode the auto LED will be on. In override mode, the fan levels *Off, 1, 2, 3* are displayed In override mode, the fan levels *Off, 1, 2, 3* are displayed for 10 s after the button is pressed.
Switch over fan between auto and override	 ***via override/auto object, override = 1*** *via auto/override object, override = 0*	Direction of control of the override object for adjusting to the fan coil actuator used. See in appendix: <u>Fan override mode</u> Override mode is triggered by a 1. Override mode is triggered by a 0.

Designation	Values	Meaning
Function of the button: Fan level	*blocked* ***press continuously*** *press for 5 mins.* *press for 15 mins.*	The button is deactivated The fan level can be selected by pressing a button. The fan coil actuator is changed by a telegram from obj. 17 for an unlimited period into <u>override mode</u>. As above, but override mode is ended after the selected time elapses.
Threshold value for fan level 1	*Send value 1* *Send value 2* *Send value 3* *Send value 4* *Send value 5* *0 %, 10 %, 20 %* *30 %, 40 %, 50 %* *60 %, 70 %, 80 %* *90 %, 100 %*	From what control value is the first fan level to be switched on? By selecting the value 1 to value 5, fan levels 1 to 5 can be actuated directly via a numerical value. No transformation into % is required.
Threshold value for fan level 2 *(larger than fan level 1 !!)*	*Send value 1* *Send value 2* *Send value 3* *Send value 4* *Send value 5* *0 %, 10 %, 20 %* *30 %, 40 %, 50 %* *60 %, 70 %, 80 %* *90 %, 100 %*	At what control value is the first fan level to be switched over to the second fan level? **Important:** The value for level 2 must always be larger than the value set for level 1
Threshold value for fan level 3 *(larger than fan level 2 !!)*	*Send value 1* *Send value 2* *Send value 3* *Send value 4* *Send value 5* *0 %, 10 %, 20 %* *30 %, 40 %, 50 %* *60 %, 70 %, 80 %* *90 %, 100 %*	At what control value is the second fan level to be switched over to the third fan level? **Important:** The value for level 3 must always be larger than the values set for levels 1 and 2.

3.4.5 Actual value

Designation	Values	Meaning
What actual value to use *or* *function of the external sensor**	***of the internal sensor***	The room temperature is measured in the device.
Comparison value for internal sensor *In 1/10 K (-64 .. 63)*	*manual input –64 ... 63* *Default value =* ***0***	Positive or negative correction of the measured temperature in 1/10K increments. Examples: a) RTF/A 1.1 sends 20.3°C. A calibrated thermometer is used to measure a room temperature of 21.0°C. To raise the temperature of the RTF/A 1.1 to 21 °C, "7" (i.e. 7 x 0.1K) must be entered. b) RTF/A 1.1 sends 21.3°C. The measurement is 20.5°C. To lower the temperature of the RTF/A 1.1 to 20.5 °C, "-8" (i.e. -8 x 0.1K) must be entered.

Designation	Values	Meaning
Send the actual value *or* *send the external actual value**	*not with a change* *with a change of 0.2 K* *with a change of 0.3 K* ***with a change of 0.5 K*** *with a change of 0.7 K* *with a change of 1 K* *with a change of 1.5 K* *with a change of 2 K*	Is the current room temperature to be sent? If it is, from what minimum change is it to be sent again? This setting is used to keep the bus load as low as possible.
Cycl. sending of the actual value *or* *Cycl. sending of the external actual value**	*do not send cyclically* *every 2 mins., every 3 mins.* *every 5 mins., every 10 mins.* *every 15 mins., every 20 mins.* ***every 30 mins****., every 45 mins.* *every 60 mins.*	How often is the actual value to be sent regardless of the temperature changes?

3.4.6 Control system

Designation	Values	Meaning
Fan coil system used	*2-pipe system* ***4-pipe system***	There is only one water circuit which circulates the cooling or heating medium depending on the time of year. The system consists of 2 separate water circuits for heating and cooling.
*Switchover betw. heating and cooling**	***automatic*** *via object*	RTF/A 1.1 automatically changes to cooling mode when the actual temperature is above the set value The cooling mode can only be activated on the bus side via object 15 (1= cooling). When this object is not set (= 0), the cooling mode remains switched off. In the 2-pipe system ⇒, always via object.
Setting the control parameters	***via system type*** *user-defined*	Standard application Professional application: <u>P/PI controller</u> parameterised on own account
System type for heating system	*Radiator heating* ***Fan coil unit***	PI-controller with: Integration time = 90 minutes Strip width = 2.5 K Integration time = 180 minutes Strip width = 4 K
System type for cooling system	*Cooling ceiling* ***Fan coil unit***	PI-controller with: Integration time = 90 minutes Strip width = 2.5 K Integration time = 180 minutes Strip width = 4 K

* Can only be set with 4-pipe systems.
With a 2-pipe system, the switchover is always carried out via object 15 .

** Change since the last send

Designation	Values	Meaning
Send the heating/cooling control value	*with a change of 1 %* *with a change of 2 %* *with a change of 3 %* ***with a change of 5 %*** *with a change of 7 %* *with a change of 10 %* *with a change of 15 %*	After what % change** in the control value is the new value to be sent? Small values increase the control accuracy, but also the bus load.
cycl. sending of the heating/cooling control value	*not cyclical, only with a change* *every 2 mins., every 3 mins.* *every 5 mins., every 10 mins.* *every 15 mins., every 20 mins.* *every 30 mins., every 45 mins.* ***every 60 mins.,***	how often does the current heating control value need to be sent (regardless of changes)?
	User-defined control parameters	
Proportional range of the heating control unit	*1 K, 1.5 K,* ***2 K****, 2.5 K, 3 K* *3.5 K, 4 K, 4.5 K* *5 K, 5.5 K, 6 K* *6.5 K, 7 K, 7.5 K* *8 K, 8.5 K*	Professional setting for modifying the control behaviour for the room. Small values result in large changes in the control values; larger values result in a less dramatic change in the control value. See in appendix: Temperature control
Integration time of the heating controller	*15 mins., 30 mins., 45 mins.* *60 mins., 75 mins., 90 mins.* *105 mins., 120 mins., 135 mins.* ***150 mins****., 165 mins., 180 mins.* *195 mins., 210 mins., 225 mins.*	The integration time determines the response time of the controller. It specifies the increase by which the output control value is increased in addition to the P-component. The I-component remains active for as long as there is a control deviation. The I-component is increased by the P-component. See in appendix: Behaviour of the PI controller

Designation	Values	Meaning
Proportional range of the cooling controller	*1 K, 1.5 K, 2 K, 2.5 K, 3 K 3.5 K,* ***4 K****, 4.5 K 5 K, 5.5 K, 6 K 6.5 K, 7 K, 7.5 K 8 K, 8.5 K*	Professional setting for modifying the controller behaviour for the room. Large values result in less dramatic changes in the control values when the control deviation remains the same, and gives more precise control than lower values.
Integration time of the cooling controller	*15 mins., 30 mins., 45 mins., 60 mins., 75 mins.,* ***90 mins.****, 105 mins.,, 120 mins., 135 mins., 150 mins., 165 mins., 180 mins., 195 mins., 210 mins., 225 mins.*	The integration time determines the response time of the control. It specifies the increase by which the output control value is increased in addition to the P-component. The I-component remains active for as long as there is a control deviation. The I-component is increased by the P-component. See in appendix: Behaviour of the PI controller

3.4.7 Operating mode

Designation	Values	Meaning
Objects for setting the operating mode	*new: Operating mode, presence, window status* ***old: Comfort, night, frost (not recommended)***	The RTF/A 1.1 can change the operating mode depending on the window and presence contacts. Traditional setting without window and presence status.
Operating mode after reset	*Frost protection* *Nighttime temperature reduction* ***Standby*** *Comfort*	Operating mode after commissionign or re-programming
*Type of presence sensor * (on obj. 4)*	 ***Presence watchdog*** *Presence button*	The presence sensor activates the comfort operating mode Comfort operating mode for as long as the presence object is set. 1. If the object *operating mode setting* (object 3) is sent again after the presence object has been set, the new operating mode is activated and the status of the presence object is ignored. 2. If the presence object is set with night/frost operation, it is reset after the parameterised comfort extension elapses (see below). 3. The presence object is not fed back to the bus.

* See also in appendix: Set the presence object with set value shift

Designation	Values	Meaning
Comfort extension via presence button in night mode	*none* *30 min.* *1 hour* *1.5 hours* ***2 hours*** *2.5 hours* *3 hours* *3.5 hours*	- Party button: this means that the RTF/A 1.1 can change back to comfort mode for a limited time from night/frost mode via the presence object. If the device was previously in standby, the time limit does not apply. The comfort mode is then not cancelled until the next manual or bus-controlled operating mode change.
cycl. sending of the current operating mode	***not cyclical, only with a change*** *every 2 mins., every 3 mins.* *every 5 mins., every 10 mins.* *every 15 mins., every 20 mins.* *every 30 mins., every 45 mins.* *every 60 mins.*	how often is the current operating mode to be sent?

3.4.8 Inputs E1, E2, E3

Designation	Values	Meaning
Function of E1, E2 or E3: ***Switch***		
Reaction to the contact closing	*Off* ***On*** *Circ* *none*	Send switch-off command Send switch-on command Reverse last switch command Do not send
Reaction to the contact opening	***Off*** *On* *Circ* *none*	See above
send cyclically	***not cyclical, only with a change*** *every 2 mins., every 3 mins.* *... every 45 mins., every 60 mins.*	At what interval is the status of the switch object to be sent?

Function of E1 (+ E2): Blind up (down)		
Function of E1	*Blind up*	Briefly press the button: Step/stop or slat turn (obj. 9) Long button press: Up telegram (obj.12)
Function of E2	*Blind down*	Briefly press the button: Step/stop or slat turn (obj. 9) Long button press: Down telegram (obj.12)
Function of E1, E2, E3: One-panel blind operation		
Function of E1 *(or E2, E3)*	*One-panel blind operation*	Briefly press the button: Step/stop or slat turn. The sent value is set against the telegram of the last movement command Long button press: Up / down The direction of movement is reversed each time the button is pressed. Down is always started first after a bus failure or reset.

Designation	*Values*	Meaning
Function of E1 (+ E2): Brighter/darker dimming		
Function of E1	*Brighter dimming*	Briefly press the button: On / off (obj. 9) Long button press: Brighter dimming (obj.12)
Function of E2	*Dimming darker*	Briefly press the button: On / off (obj. 9) Long button press: Darker dimming (obj.12)

Function of E1, E2, E3: One-panel dimming operation		
Function of E1 (or E2, E3)	*One-panel dimming operation*	Briefly press the button: On/off. The switch status is reversed each time the button is pressed. Long button press: Brighter / darker. The dimming direction is reversed each time the button is pressed. Dimming brighter is always started after a bus failure or reset. When an extended operation is released, a stop telegram is sent.
Common parameters for blind and dimming functions		
Long button press down	*300 ms* *400 ms* ***500 ms*** *600 ms* *700 ms* *800 ms* *900 ms* *1000 ms*	Limit value to distinguish between short and long button presses (in 1/1000s). Depending on whether a button is pressed for a long time or a short time, 2 different functions can be carried out.
Function of E3: Temperature sensor		
See function of the external sensor on the parameter side actual value		

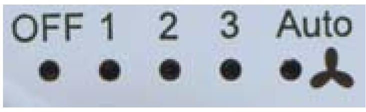
See in appendix: External interface

4 Commissioning

4.1 Actuators for heating and cooling control

There are several options available for actuating the heating and cooling devices. For more information, see the current ABB i-bus® product overview, chapter 9 "Heating and cooling".

4.2 Control value display



E D C B A

The current control value is displayed when the button (top right) is pressed for longer than 2 seconds.

LED	Control value
no LED	0 %
LED A (Auto)	1 - 25%
LED B (level 3)	26 - 50%
LED C (level 2)	51 - 75%
LED D (level 1)	76 - 100%

LED E indicates whether heating (red) or cooling (blue) is currently being carried out.

5 Appendix

5.1 Fan override mode

This function permits the manual pre-selection of the fan level, both via the button on the device and via the bus.
It can be time-activated on the *Operation* parameter page or activated or blocked permanently.

Button operation

Button press	Function	LED
1	Fan off	OFF
2	Fan level 1	1
3	Fan level 2	2
4	Fan level 3	3
5	Auto	Auto

Important: Depending on the actuator used, either a 1 or a 0 is required to trigger override operation.
This behaviour is adjustable, see parameter *Switch fan between auto and override* on parameter page *Operation.*

Send behaviour with override using fan coil actuator (override = 1):
Object 17 sends a 1 to the fan coil actuator, triggering override operation.
Object 16 sends the control value for the fan level selected in accordance with the threshold value set.

Important: The override control value sent should always be slightly higher than the threshold setting in the fan coil actuator.

Example:
For actuating the ABB i-bus fan coil actuator LF/A 1.1

Threshold value for fan level	Set values with RTF/A 1.1	LFA/S adjusts the level automatically
1	Send value 1	1
2	Send value 2	2
3	Send value 3	3

5.2 *Determining the current operating mode*

The current set value can be be adjusted for the relevant requirements by selecting the operating mode.
The operating mode can be set via objects 3..5.
There are two ways of doing this:

5.2.1 New operating modes

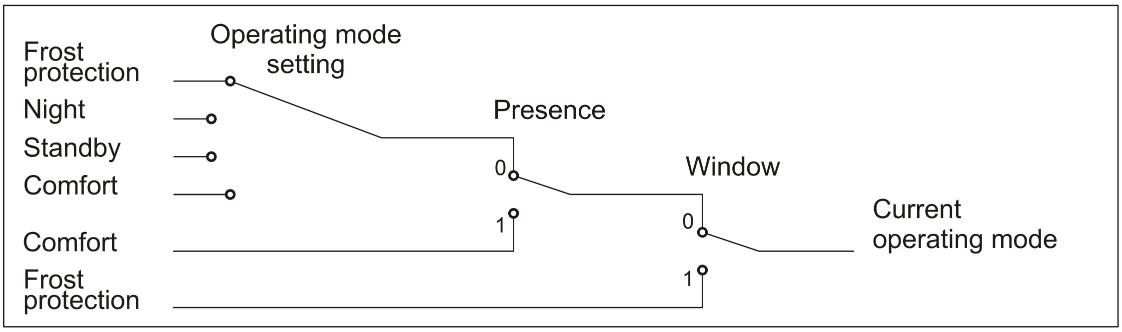
If New... has been selected on the operating type parameter page for the *Setting the operating mode* parameter, the current operating mode can be specified as follows:

Operating mode selection Object 3	Presence Object 4	Window status Object 5	current operating mode (Object 6)
any	any	1	Frost/heat protection
any	1	0	Comfort
Comfort	0	0	Comfort
Standby	0	0	Standby
Night	0	0	Night
Frost/heat protection	0	0	Frost/heat protection

Typical application:

A time switch is used to activate the *Standby* or *Comfort* operating mode in the morning and the *Night* operating in the evening via object 3.
During holiday periods, another channel on the frost/heat protection time switch is also selected via object 3.

Object 4 is connected to a presence watchdog. When presence is deteced, RTF/A 1.1 switches to the Comfort operating mode (see table).
Object 5 is connected to a window contact via the bus (binary input).
When a window is opened, RTF/A 1.1 changes to the frost protection operating mode.



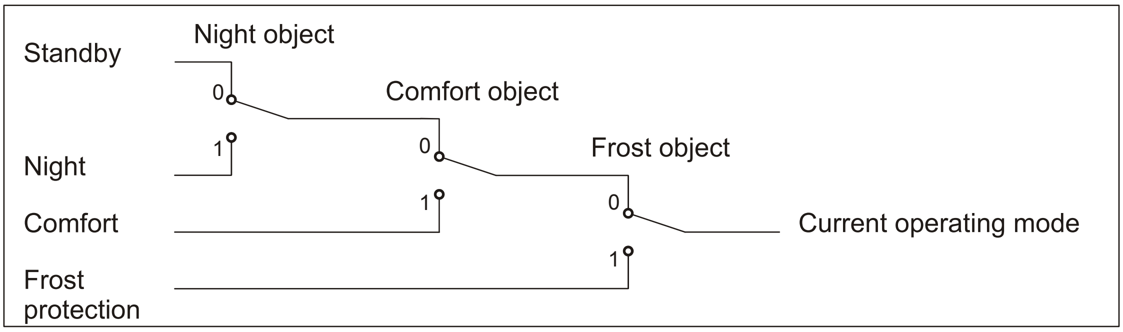
5.2.2 Old operating modes

If Old... has been selected on the operating mode parameter page for the *Setting the operating mode* parameter, the current operating mode can be set as follows:

Night Object 3	Comfort Object 4	Frost/heat protection object 5	current operating mode Object 6
any	any	1	Frost/heat protection
any	1	0	Comfort
Standby	0	0	Standby
Night	0	0	Night

Typical application: Object 3 activates the standby operating mode in the morning and the *Night* operating mode in the evening via a time switch.

During holiday periods, another channel on the frost/heat protection time switch is also selected via object 5.
Object 4 (Comfort) is connected to a presence watchdog. If presence is detected, RTF/A 1.1 switches to the Comfort operating mode (see table).
Object 5 is connected to a window contact: When a window is opened, RTF/A 1.1 changes to the frost protection operating mode.



The old process has two disadvantages in comparison to the new process:
1. To access the night operating mode from the comfort operating mode, 2 telegrams are required (2 channels of a time switch where applicable): Object 4 must be set to "0" and object 3 to "1".
2. If the window is opened and closed again at times when the frost/heat protection time switch is selected, the frost/heat protection operating mode is cancelled.

5.2.3 Calculating the set value

5.2.3.1Set value calculation in heating operation

See also: Basic set value and current set value

current set value for heating

Operating mode	Current set value
Comfort	Basic set value +/- set value shift
Standby	Basic set value +/- set value shift – drop in standby mode
Night	Basic set value +/- set value shift – drop in night mode
Frost/heat protection	parameterised set value for frost protection mode

Example:
Heating in the comfort operating mode.

Parameter page	Parameters	Setting
Set values	*Basic set value after reset*	*21 °C*
	Drop in standby mode (with heating)	*2 K*
Operation	*Max. set value shift on the setting wheel*	*+/- 2 K*

The set value has previously been increased by 1 K using the setting wheel.

Calculation:

Current set value = Basic set value + set value shift
= 21°C + 1K
= 22°C

If the device is changed to standby mode, the current set value is calculated as follows:

Current set value = basic set value + set value shift – drop in standby mode
= 21°C + 1K – 2K
= 20°C

5.2.3.2 Set value calculation in cooling mode

current set value for cooling

Operating mode	Current set value
Comfort	Basic set value + set value shift + dead zone
Standby	Basic set value + set value shift + dead zone + increase in standby mode
Night	Basic set value + set value shift + dead zone + increase in night mode
Frost/heat protection	parameterised set value for heating protection mode

Example:
Cooling in the comfort operating mode.

The room temperature is too high, RTF/A 1.1 has switched to cooling mode

Parameter page	Parameters	Setting
Set values	*Basic set value after reset*	*21 °C*
Cooling set values	*Dead zone between heating and cooling*	*2 K*
	Increase in standby mode (with cooling)	*2 K*
Operation	*Max. set value shift on the setting wheel*	*+/- 2 K*

The set value has previously been reduced by 1 K using the setting wheel.

Calculation:

Current set value = Basic set value + set value shift + dead zone
= 21°C - 1K + 2K
= 22°C

Changing to standby mode means another increase in the set value (energy saving), which produces the following set value.
Set value = basic set value + set value shift + dead zone + increase in standby mode
= 21°C - 1K + 2K + 2K
= 24°C

5.3 Set value shift

The current set value can be changed in two different ways on the RTF/A 1.1.

- incrementally using the setting wheel (see parameter page *Operation*, *Function of the setting wheel*)
- via object 0 *Manual set value shift*

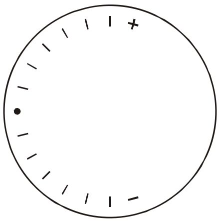
The amount for the set value shift relative to the basic set value is sent by object 1 for any change (e.g. -1.00).

The limits for the shift are specified on the *Operation* parameter page using the *maximum set value shift on the setting wheel* parameter and apply to both types of set value shift.

5.3.1 Set temperature shift via the setting wheel

This option is available if the setting wheel has been enabled for this purpose on the *Operation* parameter page:

For this function, the +/- setting wheel is plugged onto the device (see illustration).



In the centre position of the setting wheel, the set value shift is zero.
If the setting wheel is turned to the left as far as it will go (+), the set value is increased by the maximum parameterised set value shift.

The shift can be set very accurately using the gradations on the setting wheel.
The temperature change per dash is dependent on the maximum parameterised set value shift.

maximum set value shift on the setting wheel	Kelvin / °C per dash
+/- 1 K (i.e. +/-1°C)	1/6
+/- 2 K	1/3
+/- 3 K	1/2

5.3.2 Set temperature shift via object 0

This option is only available if the following settings have been selected on the *Operation* parameter page:

Parameter page	Parameters	Setting
Operation	*Function of the setting wheel*	*Basic set value for internal controller* **or** *blocked, but man. shift object present*

Here the set value is changed directly by sending the required shift to object 0.
To do this, the difference (where applicable as a negative value) is sent to object 0 in EIS5 format.

The shift always relates to the basic set value (as parameterised or specified by the setting wheel) and not to the current set value.

Example Basic set value 21°C:

If the value 2.00 is set to obj. 0, the new set value is calculated as follows:
21°C + 2.00K = 23.00°C.
To then bring the set value to 22°C, the difference is sent again to the parameterised basic set value (here; 21°C), in this case 1.00K (21°C + 1.00K = 22°C)

5.3.3 Set the presence object with set value shift

The RTF/A 1.1 allows you to set the presence object solely by increasing the set value on the setting wheel. This means that the device changes to comfort mode and the room temperature is brought to a comfortable level.
This function can be activated using the *For set value increase on the setting wheel → Set presence object* parameter on the *Operation* parameter page
No reset is carried out due to the set temperature reducing.

Exit comfort mode.

Presence sensor on object 4	Presence object
Presence watchdog	is reset by the presence watchdog. If no watchdog is fitted, a time switch can be used at night to reset the presence object every 2 hours (obj. 4).
Presence button	is reset in night mode after the timer elapses* or via a time switch (see above).

* Parameter *Comfort extension via presence button in night mode* on the *Operating mode* parameter page.

5.4 External interface

The external interface is activated on the *Settings* parameter page.
It consists of the 3 inputs E1, E2 and E3.

E1 and E2 are purely binary inputs, E3 can be used both as a binary input and as an analogue input for an external temperature sensor.
All 3 inputs are connected in the base via the connection terminals

The following functions can be realised:

- Switching (1 switch or button)
- Blind up/down (with 2 buttons on E1 + E2)
- One-panel blind operation (with 1 button)
- Dimming brighter/darker (with 2 buttons on E1 + E2)
- One-panel dimming operation (with 1 button)

If the blind and dimming functions are implemented via 2 buttons, E1 and E2 are automatically connected with one another and act jointly on objects 9, 10 and 12.

5.4.1 Overview: Function of objects 9 .. 14.

Function of E1

Function of E1	Function		
	Object 9	Object 10	Object 12
Switch	sends the switch status of the E1 input	*Not used*	*Not used*
Blind UP Note: E2 is automatically set to blind DOWN.	Sends commands for step/stop in the up direction or slat positioning	*Not used*	Sends up command to blind
One-panel blind operation	Sends commands for step/stop or slat positioning	Sends up/down command to blind	*Not used*
Brighter dimming Note: E2 is automatically set to dim darker	Sends on/off commands to the dimmer	*Not used*	Sends 4 bit dimming commands
One-panel dimming operation	Sends on/off commands to the dimmer	Sends 4 bit dimming commands	*Not used*

Function of E2

Function of E2	Function	
	Object 11	Object 12
Switch	sends the switch status of the E2 input	*Not used*
One-panel blind operation	Sends commands for step/stop or slat positioning	Sends up/down command to blind
One-panel dimming operation	Sends on/off commands to the dimmer	Sends 4 bit dimming commands
Blind down	Fixed setting if E1 is parameterised on *Blind up*. See previous table:*Function of E1*	
Dimming darker	Fixed setting if E1 is parameterised on *Dimming brighter*. See previous table:*Function of E1*	

Function of E3

Function of E3	Function	
	Object 13	Object 14
Switch	sends the switch status of the E2 input	*Not used*
One-panel blind operation	Sends commands for step/stop or slat positioning	Sends up/down command to blind
One-panel dimming operation	Sends on/off commands to the dimmer	Sends 4 bit dimming commands
*Temperature sensor**	*Not used*	*Not used*

*The measured actual value is sent by object 2.

5.4.2 E1...E3 as switching inputs

If an input has been parameterised as a switch input, switches can also be used as buttons. The status of the relevant object (obj. 9...11) is switched according to the parameterisation.

ON / OFF with switch

Parameter page	Parameters	Setting
Input E1(E2, E3)	*Reaction to the contact closing*	*On*
	Reaction to the contact opening	*Off*

ON / OFF with button (see surge relay)

Parameter page	Parameters	Setting
Input E1(E2, E3)	*Reaction to the contact closing*	*Circ*
	Reaction to the contact opening	*none*

See above: Overview: Function of the objects 9 .. 14.

5.4.3 E1...E2 blind up/down

Two buttons are connected to actuate a blind (E1 + E2).
Objects 9 (step/stop) and 10 (up/down) are linked in this case using a KNX blind actuator.

With both inputs, a distinction is made between a short press and a long press of the button. The time to distinguish between long and short button presses is set on the *Input E1* parameter page.
With a short press, the relevant program (ON or OFF) is sent to the slat object (obj. 9), with a long press a telegram is sent to the motion object (obj.12).
Only one or the other object is ever pressed at a time.
If a button is held down, the other is ineffective.

Button press	E1	E2
long (Acts on object 12)	Up telegram (0)	Down telegram (1)
short (Acts on object 9)	Step/stop telegram in up direction (0)*	Step/stop telegram in down direction (1)*

*The decision between step and stop is carried out in the blind actuator itself, depending on the operating position.

See above: Overview: Function of the objects 9 .. 14.

5.4.4 One-panel blind operation

Advantage: One-panel operation requires only one button and therefore only occupies one input.
Functional principle: The motion or step direction is reversed with every button press.

Table 3

Button press	E1, E2, E3
long	Up or down telegram (0)
short	Step/stop telegram in up or down direction (0)*

See above: Overview: Function of the objects 9 .. 14.

5.4.5 E1...E2 brighter/darker dimming

Two buttons are connected to implement a dimmer function.
Objects 9 (dimming on/off) and 12 (dimming??? blind??? up/down) must then be connected to ABB I-BUS® KNX DIM ACTUATORS

If the *Dimming brighter* function is selected on E2, the corresponding function, i.e. *Dimming darker* is automatically set for E2.

With both inputs, a distinction is made between a short press and a long press of the button. The time to distinguish between long and short button presses is set on the *Input E1* parameter page.
With a short press, the relevant telegram (ON or OFF) is sent; with a long press the telegram is sent to the dimming object (obj.12).

Button press	E1	E2
long (Acts on object 12)	- When the button is pressed, a start telegram is sent for brighter dimming - on release, a stop telegram	- When the button is pressed, a 4-bit start telegram is sent for darker dimming - on release, a stop telegram
short (Acts on object 9)	Switch-on telegram	Switch-off telegram

See above: Overview: Function of the objects 9 .. 14.

5.4.6 One-panel dimming operation

Advantage: One-panel operation requires only one button and therefore only occupies one input.
Functional principle: With each press of the button, the dimming direction is reversed or the light is switched on or off.

Button press	E1
long	- When the button is pressed, a start telegram is sent for brighter or darker dimming - on release, a stop telegram
short	Switch-on or switch-off telegram

See above: Overview: Function of objects 9 .. 14.

5.5 Temperature control

5.5.1 Introduction

If the RTF/A 1.1 is not configured as a switching controller, it can be parameterised either as a P or as a PI controller, whereby PI regulation is preferable.

With the proportional controller (P controller), the control value is modified statically to the control deviation.
The proportional/integral controller (PI controller) is much more flexible, i.e. it controls dynamically (faster and more accurately).

To explain the functioning of both temperature controllers, the room being heated is compared to a container in the following example

The fill level of the container represents the room temperature.
The water supply represents the radiator output.
The heat losses of the room are represented by a tap.

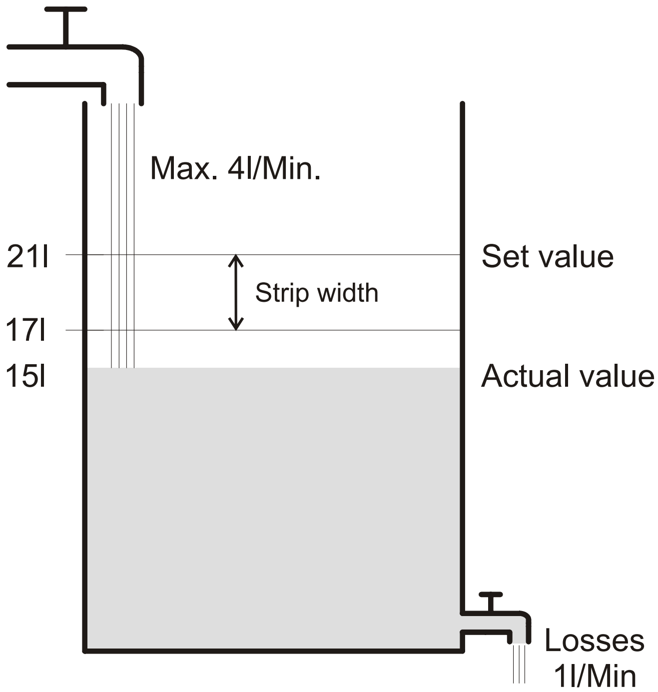
In our example, the maximum supply quantity is assumed to be 4 litres per minute and also represents the maximum heating output of the radiator in our example.
This maximum output is reached at a control value of 100%.
Accordingly, with a control value of 50% only half the water volume, i.e. 2 litres per minute, flows into our container.

The strip width is 4l.
This means that the controller will regulate at 100%, as long as the actual value will be less than or equal to (21l – 4l) = 17l.

Problem definition:

- Required filling quantity:
  21 litres (= set value)
- At what point is the supply to be gradually reduced to avoid an overflow?
  4l under the required fill quantity, i.e. at 21l - 4l = 17l (= strip width)
- Outlet filling quantity
  15l (=actual value)
- The losses are 1l/minute

5.5.2 Behaviour of the P controller



If the filling quantity is 15l, a control deviation is produced of 21l – 15l = 6l
As our actual value is outside the strip width, the controller will actuate the supply with 100%
i.e. at 4l / minute.

The supply quantity (= control value) is calculated using the control deviation (set value – actual value) and the strip width.
Control value = (control deviation / strip width) x 100

The table below clearly explains the behaviour and therefore also the limits of the P controller.

Filling level	Control value	Supply	Losses	Supply filling level
15l	100%	4 l/min	1 l/min	3 l/min
19l	50%	2 l/min		1 l/min
20l	25%	1 l/min		0 l/min

In the last line, it is clear that the filling level cannot increase any more because the supply is allowing exactly the amount of water that can flow out again via losses.
The result is a consistent control deviation of 1l, the set value can never be reached.
If the losses were 1l higher, the remaining control deviation would increase by the same amount and the filling level would never exceed the 19l mark.
In a room, this would mean that the control deviation increases with a falling outside temperature.

P controller as temperature controller

The P controller behaves in exactly the same way with a heating control as in the previous example.
The set temperature (21°C) can never be fully reached.

The remaining control deviation becomes higher the greater the heat losses, i.e. the lower the outside temperatures fall.

5.5.3 Behaviour of the PI controller

Unlike the pure P controller, the PI controller works dynamically.
With this type of controller, the set value does not remain unchanged even with a constant deviation.

In the first moment, the PI controller sends the same control value as the P controller, but this is increased even more the longer the set value is not reached.
This increase is carried out in a time-controlled way via what is known as the integration time.
With this calculation process, the control value is only then no longer changed if the set value and the actual value are the same.
This produces a balance between the supply and run-off in our example.

Note on temperature control:

Good control depends on the balancing of the strip width and integration time with the room which is to be heated.
The strip width affects the increment of the control value change:
Large strip width = finer increments with a change in control value.
The integration time affects the reaction time to temperature changes
Long integration time = slow reaction.
A poor balance can mean that either the control value is exceeded (overshoots), or the controller takes too long to reach the set value.

Normally, the best results are achieved using the *via system type* setting.

Parameter page	Parameters	Setting
Control system	*Setting the control parameters*	*via system type*

6 Glossary

6.1 Continuous and switching control

A switching (2-point) control detects only 2 status conditions; or on off.
A continuous control works with a set value between 0% and 100% and can therefore dose the energy supply precisely. This achieves convenient and precise control.

6.2 Hysteresis

The hysteresis determines the difference between switch-on and switch-off temperature with a controller. It can be both positive and negative.
With a combination of heating and cooling control, it affects the amount of the dead zone.

Without hysteresis, the controller would switch on and off without interruption as long as the temperature is in the range of the set value.

6.2.1 Negative hysteresis:

Heating: The heating is carried out until the set value is reached.
The heating is only switched on again when the temperature has fallen below the threshold *set value hysteresis*.
Cooling: Cooling is carried out until the *set value – hysteresis* threshold is reached.
The system is only switched on again if the temperature has risen above the set value.

Cooling example:
Cooling with set value 25 °C, hysteresis = 1°C and ambient temperature 27 °C.
The cooling is switched on and only switches back off again once a temperature of 24 °C (25 °C – 1 °C) is reached.
Once the temperature rises above 25 °C, the system is switched back on again.

6.2.2 Positive hysteresis

Heating is carried out until the temperature reaches the *set value + hysteresis* threshold.
The heating is only switched on again if the temperature has dropped below the set value.

Heating example:
Heating with set value 20°C, hysteresis = 1°C and ambient temperature 19 °C.
The heating is switched on and only switches back off again once a temperature of 21 °C (20 °C – +1 °C) is reached.

Once the temperature drops below 20 °C, the system is switched back on again.

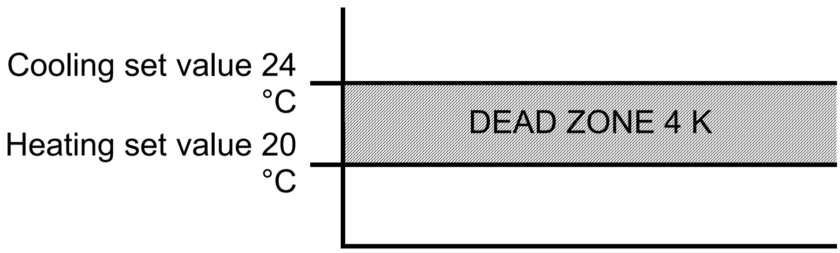
6.3 Dead zone

The dead zone is a buffer area between the heating and cooling modes.
Within this dead zone, neither heating or cooling is carried out.

Without this buffer zone, the system would be permanently changing between heating and cooling. If the set value were to be not reached, the heating would be activated and the set value would barely be reached before the cooling would start, the temperature would drop back below the set value and the heating would switch back on again.

Depending on the type of control, the dead zone may increase by the value of the hysteresis.

6.3.1 Heating and cooling with continuous control



The dead zone (4 K) is not affected.

6.4 Basic set value and current set value

The **basic set value** is used as the standard temperature for the comfort operating mode and as a reference temperature for the temperature drop in the standby and night operating modes.
The parameterised basic set value (see basic set value after downloading the application) is stored in object 0 and can be changed at any time via the bus by sending a new value to object 0 (EIS5).
After a reset (bus return), the last basic set value used is restored.

The **current set value** is the set value actually used for control purposes. It is the result of all drops or increases due to operating mode control functions.

Example:
With a basic set value of 22°C and a drop in night operation of 4K (for night mode) of the current set value: 22°C – 4K = 18°C. During the day (in comfort mode), the current set value is 22°C (if the cooling mode is not active).

The formation of the current set value based on the basic set value can be seen on the block diagram on the next page:

On the left is the basic set value which has been specified via object 0 or set on the setting wheel.
On the right is the current set value, i.e. the value to which the room temperature is effectively controlled.

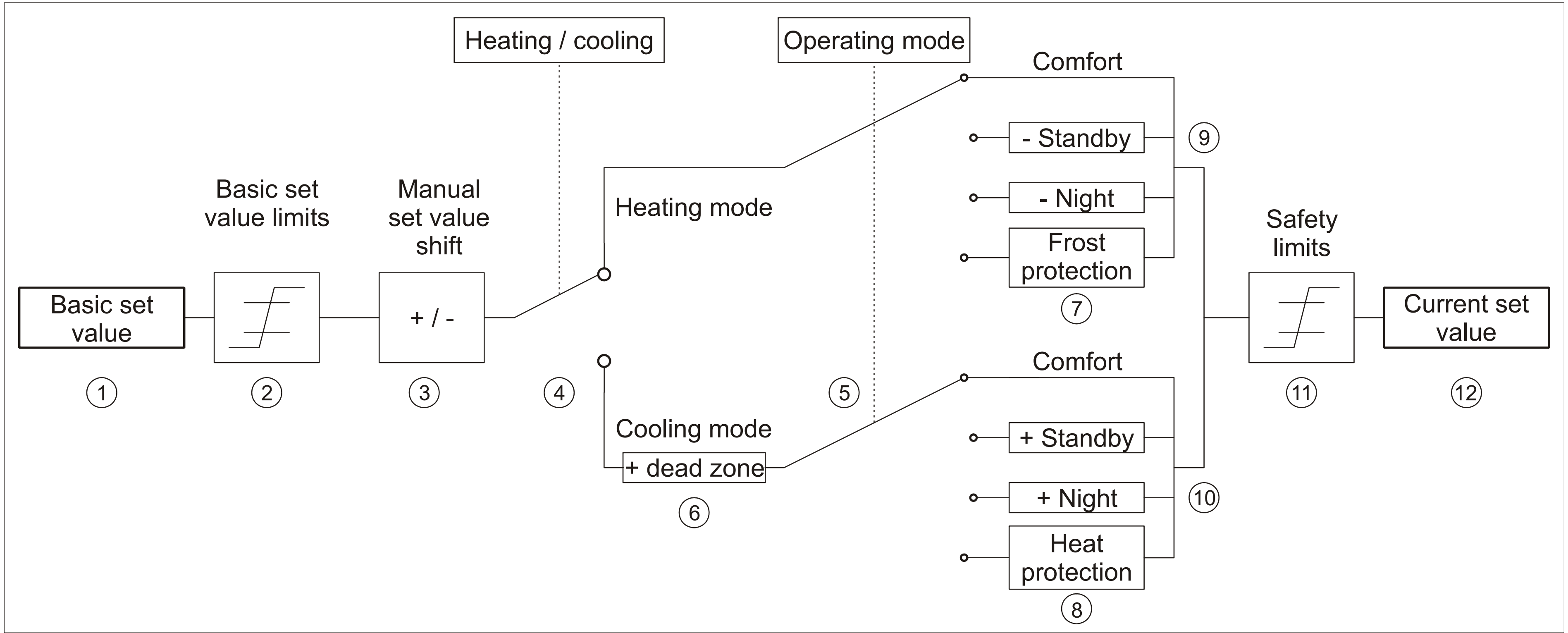
As shown on the block diagram, the current set value depends on the operating mode (5) and on the control function selected (4).

The basic set value limits (2) prevent an incorrect basic set value specification to object 0. These are the following parameters:

- minimum valid basic set value
- maximum valid basic set value
- minimum setting on the setting wheel
- maximum setting on the setting wheel

If the set value is outside the parameterised values for frost and heat protection due to a set value shift, it is limited to these values via the safety limits (11).

6.4.1 Set value calculation



1 Specified basic set value from object 0 or setting wheel
2 Max. and min. valid basic set values / setting on the setting wheel
3 Manual set value shift
4 Switching betw. heating or cooling: Automatic or via object 6
5 Selecting the operating mode
6 The set value is increased in cooling mode by the amount of the dead zone
7 The set value is replaced with the set value for frost protection mode
8 The set value is replaced with the set value for heat protection mode
9 Set value according to drops due to the operating mode
10 Set value according to increases due to the operating mode
11 The limits for frost and heat protection must be observed
12 Current set value after mode-dependent increases, drops and limits

The information provided in this manual are subject to technical modification.

Print no. 2CDC 508 060 D0101