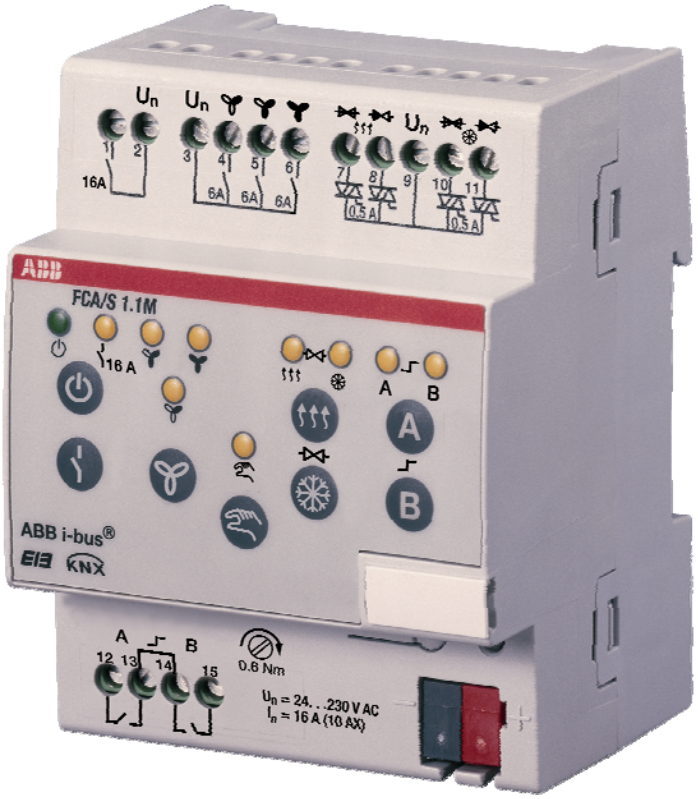
Product Manual

ABB i-bus® KNX
Fan Coil Actuator
FCA/S 1.1M

Intelligent Installation Systems



This manual describes the function of the Fan Coil Actuator FCA/S 1.1M
Subject to changes and errors excepted.

Exclusion of liability:
Despite checking that the contents of this document match the hardware and software, deviations cannot be completely excluded. We therefore cannot accept any liability for this.
Any necessary corrections will be inserted in new versions of the manual.
Please inform us of any suggested improvements.

Contents

Contents

Contents

1 General

Ventilator convectors, also referred to as blower convectors or Fan Coil units, are used for distributed heating and cooling applications. There are installed in a room and powered via a central heating and cooling system. The room temperature can be quickly adjusted to suit individual preferences using this system.

The Fan Coil Actuator FCA/S 1.1M has two outputs for control of motor power operated or thermal heating and cooling valves. Multi-level fans with up to three fan speeds can be switched via floating contacts. Furthermore, two binary inputs, e.g. for monitoring of a window contact and the dew point are available.
The additional contact is used, for example, to control an electric heater.

This manual provides you with detailed technical information relating to the Fan Coil Actuator, its installation and programming. The application of the device is described using examples.

This manual is divided into the following sections:

Chapter 1	General
Chapter 2	Device technology
Chapter 3	Commissioning
Chapter 4	Planning and application
Chapter A	Appendix

1.1 Product and functional overview

The Fan Coil Actuator FCA/S 1.1M controls a single-phase fan with up to three fan speeds via a step or changeover control. This ensures that no two fan speeds can be switched on simultaneously. An additional programmable switch-over delay is provided for this purpose. Three-phase drives are not supported. The additional output can be used for control of an electrical load. The device can be operated manually.

The FCA/S 1.1M controls motor-power operated heating and cooling valves as well as multi-level fans via the ABB i-bus®.

Two binary inputs are available, for example, as signalling contacts for window contact and dew point monitoring. The scanning voltage for the binary inputs is provided by the device.

The Actuator is a modular installation device with a module width of 4 space units in Pro*M* Design for installation in the distribution board. The connection to the ABB i-bus® is established using the front side bus connection terminal. The Fan Coil Actuator does not require an auxiliary voltage supply.
The assignment of the physical addresses as well as the parameterisation is carried out with Engineering Tool Software ETS (from Version ETS2 V1.3a) with a *.VD2 file. If ETS3 is used a *.VD3 type file or higher must be imported.

Note
The illustrations of the parameter windows in this manual correspond to the ETS3 parameter windows. The user program is optimised for ETS3. In the ETS2 it is possible that the parameter page is automatically split if all parameters are used.

1.1.1 Product overview

	FCA/S 1.1M
Inputs	
Binary via contact scanning	2

	FCA/S 1.1M
Outputs	
Switching contact 16 A (10AX)	1
Switching contact 6 A	3
Electronic 0.5 A	4

1.1.2 Functional overview

	FCA/S 1.1M
Inputs	**2**
Window contact	1
Condensed water (dew point) sensor	1

	FCA/S 1.1M
Outputs 16 A (10 AX) switch	**1**
Electrical auxiliary heater	1
Outputs 6 A switches	**3**
3 speed fan	3
Outputs 0.5 A switches	**4**
Valve heating	2
Valve cooling	2

2 Device technology

2CDC 071 112 F0008

The Fan Coil Actuator FCA/S 1.1M is a modular installation device (MDRC) in pro*M* design. It is intended for installation in the distribution board on 35 mm mounting rails. The assignment of the physical address as well as the parameter settings is carried out with ETS2 from version V1.3a or higher.

The device is powered via the ABB i-bus® and does not require and additional auxiliary voltage supply. The FCA/S 1.1M is operational after connection of the bus voltage.

2.1 Technical data

Supply	Bus voltage	21…32 V DC
	Current consumption, bus	< 12 mA
	Leakage loss, bus	Maximum 250 mW
	Leakage loss, device	Maximum 2.85 W*
*The maximum power consumption of the device results from the following specifications:	KNX bus connection Relay 16 A Relay 6 A Relay 0.5 A	0.25 W 1.0 W 0.6 W 1.0 W
Connections	KNX	Via bus connection terminals
	Inputs/Outputs	Via screw terminals
Operating and display elements	Programming button/LED	For assignment of the physical address
	Button /LED	For toggling between manual operation/operation via ABB i-bus® and displays
	Button On/Off /LED	Programmable function
	Fan speed button	For switching through the individual fan speeds
	LED fan speed 1 LED fan speed 2 LED fan speed 3	For display of fan speed 1 For display of fan speed 2 For display of fan speed 3
	Valve heating button /LED	For control and display
	Valve cooling button /LED	For control and display
	Button switch contact /LED	For switching and display
	Button binary input /LED	For switching and display
	Button binary input /LED	For switching and display
Enclosure	IP 20	to DIN EN 60 529
Safety class	II	to DIN EN 61 140
Isolation category	Overvoltage category	III to DIN EN 60 664-1
	Pollution degree	2 to DIN EN 60 664-1
KNX safety extra low voltage	SELV 24 V DC	

Temperature range	Operation	-5 °C…+45 °C
	Transport	-25 °C…+70 °C
	Storage	-25 °C…+55 °C
	Storage at temperatures exceeding +45 °C reduces the service life!	
Ambient conditions	Maximum air humidity	93 %, no condensation allowed
Design	Modular installation device (MDRC)	Modular installation device, Pro*M*
	Dimensions	90 x 72 x 64.5 mm (H x W x D)
	Mounting width in space units	4 modules at 18 mm
	Mounting depth	64.5 mm
Installation	On 35 mm mounting rail	to DIN EN 60 715
Mounting position	As required	
Weight	0.1 kg	
Housing/colour	Plastic housing, grey	
Approvals	KNX to EN 50 090-1, -2	Certification
CE mark	In accordance with the EMC guideline and low voltage guideline	

2.1.1 Electronic outputs

Rated values	Number of	4, non-isolated, short-circuit proofed
	Un rated voltage	24…230 V AC (50/60 Hz)
	I_n rated current (per output pair)	0.5 A
	Continuous current	0.5 A resistive load at T_u up to 20 °C
		0.3 A resistive load at T_u up to 60 °C
	Inrush current	Maximum 1.6 A, 10 s at T_u up to 60 °C
		T_u = ambient temperature

2.1.2 Binary inputs

Rated values	Number of	2
	U_n scanning voltage	32 V, pulsed
	I_n scanning current	0.1 mA
	Scanning current I_n at switch on	Maximum 355 mA
	Permissible cable length	≤ 100 m one-way, at cross-section 1.5 mm²

2.1.3 Fan rated current 6 A

Nominal values:	Number of	3 contacts
	U_n rated voltage	250/440 V AC (50/60 Hz)
	I_n rated current (per output)	6 A
Switching currents	AC3* operation (cos φ = 0.45) DIN EN 60 947-4-1	6 A/230 V
	AC1* operation (cos φ = 0.8) DIN EN 60 947-4-1	6 A/230 V
	Fluorescent lighting load to DIN EN 60 669-1	6 A/250 V (35 µF)[2)]
	Minimum switching performance	20 mA/5 V 10 mA/12 V 7 mA/24 V
	DC current switching capacity (ohmic load)	6 A/24 V=
Service life	Mechanical endurance	> 10^7
	Electronic endurance to DIN IEC 60 947-4-1	
	AC1* (240 V/cos φ = 0.8)	> 10^5
	AC3* (240 V/cos φ = 0.45)	> 1.5 x 10^4
	AC5a* (240 V/cos φ = 0.45)	> 1.5 x 10^4
Switching times[1)]	Maximum relay position change per output and minute if all relays are switched simultaneously. The position changes should be distributed equally within the minute.	60
	Maximum relay position change per output and minute if only one relay is switched.	240

[1] The specifications apply only after the bus voltage has been applied to the device for at least 10 seconds. Typical delay of the relay is approx. 20 ms.

[2)] The maximum inrush-current peak may not be exceeded

***What do the terms AC1, AC3 and AC5a mean?**

In Intelligent Installation Systems different switching capacity and performance specifications which are dependent on the special application have become established in industrial and residential systems. These performance specifications are rooted in the respective national and international standards. The tests are defined so that typical applications, e.g. motor loads (industrial) or fluorescent lamps (residential) are simulated.

The specifications AC1 and AC3 are switching performance specifications which have become established in the industrial field.

Typical application:

AC1 – Non-inductive or slightly inductive loads, resistive furnaces
(relates to switching of ohmic/resistive loads)

AC3 – Squirrel-cage motors: Stating, switching off motors during running
(relates to (inductive) motor load)

AC5a – Switching of electric discharge lamps

These switching performances are defined in the standard DIN EN 60947-4-1 *Contactors and motor-starters - Electromechanical contactors and motor-starters*. The standard describes starter and/or contactors which previously preferably used in industrial applications.

2.1.4 Rated current output 16 A

Nominal values:	Number of	1
	U_n rated voltage	250/440 V AC (50/60 Hz)
	I_n rated current	16 A
Switching currents	AC3* operation (cos φ = 0.45) DIN EN 60 947-4-1	8 A/230 V
	AC1* operation (cos φ = 0.8) DIN EN 60 947-4-1	16 A/230 V
	Fluorescent lighting load AX to DIN EN 60 669-1	16 A/250 V (70 µF)[2)]
	Minimum switching performance	100 mA/12 V
		100 mA/24 V
	DC current switching capacity (resistive load)	16 A/24 V=
Service life	Mechanical endurance	> 3 x 10^6
	Electronic endurance to DIN IEC 60 947-4-1	
	AC1* (240 V/cos φ = 0.8)	> 10^5
Switching times[1)]	Maximum relay position change per output and minute if all relays are switched simultaneously. The position changes should be distributed equally within the minute.	60
	Maximum relay position changes per output and minute if only one relay is switched.	120

[1] The specifications apply only after the bus voltage has been applied to the device for at least 10 seconds. Typical delay of the relay is approx. 20 ms.

[2)] The maximum inrush-current peak may not be exceeded

***What do the terms AC1, AC3 and AC5a mean?**

In Intelligent Installation Systems different switching capacity and performance specifications which are dependent on the special application have become established in industrial and residential systems. These performance specifications are rooted in the respective national and international standards. The tests are defined so that typical applications, e.g. motor loads (industrial) or fluorescent lamps (residential) are simulated.

The specifications AC1 and AC3 are switching performance specifications which have become established in the industrial field.

Typical application:

AC1 – Non-inductive or slightly inductive loads, resistive furnaces (relates to switching of ohmic/resistive loads)

AC3 – Squirrel-cage motors: Stating, switching off motors during running (relates to (inductive) motor load)

AC5a – Switching of electric discharge lamps

These switching performances are defined in the standard DIN EN 60947-4-1 *Contactors and motor-starters - Electromechanical contactors and motor-starters*. The standard describes starter and/or contactors which previously preferably used in industrial applications.

2.1.5 Lamp load output

Lamps	Incandescent lamp load	2300 W
Fluorescent lamp T5 / T8	Uncorrected Parallel compensated DUO circuit	2300 W 1500 W 1500 W
Low-volt halogen lamps	Inductive transformer Electronic transformer	1200 W 1500 W
	Halogen lamp 230 V	2300 W
Dulux lamp	Uncorrected Parallel compensated	1100 W 1100 W
Mercury-vapour lamp	Uncorrected Parallel compensated	2000 W 2000 W
Switching performance (switching contact)	Maximum peak inrush-current I_P (150 µs)	400 A
	Maximum peak inrush-current I_P (250 µs)	320 A
	Maximum peak inrush-current I_P (600 µs)	200 A
Number of electronic ballast's (T5/T8, single element)[1)]	18 W (ABB EVG 1 x 58 CF)	23
	24 W (ABB EVG-T5 1 x 24 CY)	23
	36 W (ABB EVG 1 x 36 CF)	14
	58 W (ABB EVG 1 x 58 CF)	11
	80 W (Helvar EL 1 x 80 SC)	10

[1)] For multiple element lamps or other types the number of electronic ballast's must be determined using the peak inrush current of the electronic ballast's.

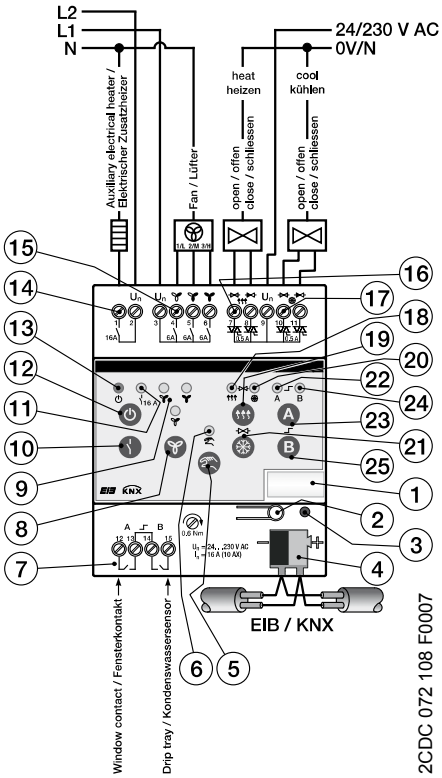
Application program	Max. number of communication objects	Max. number of group addresses	Max. number of associations
Fan Coil Actuator/1	70	85	85

Note

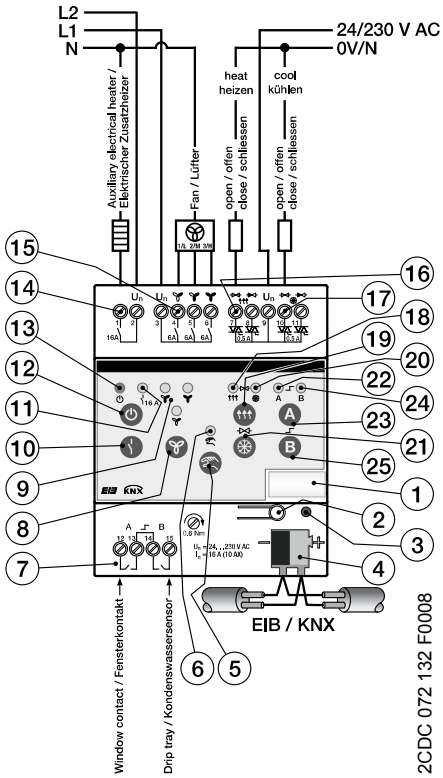
The programming requires EIB Software Tool ETS2 V1.3a or higher.

If ETS3 is used a *.VD3 or higher type file must be imported. The user program can be found in the ETS2 / ETS3 at ABB/Heating, Cooling, Blower/Fan Coil Actuator 1-fold.

2.2 Circuit diagrams



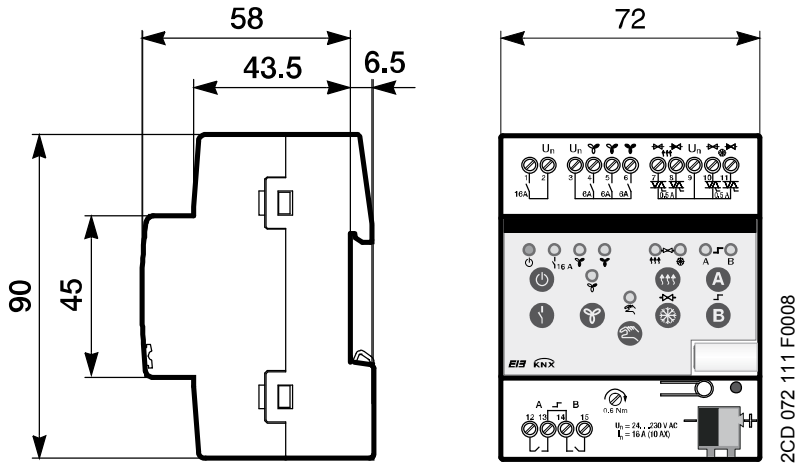
FCA/S 1.1M
with electromotor valve drives



FCA/S 1.1M
With electro-thermal valve drives

1 Label carrier
2 Programming button
3 Programming LED
4 Bus terminal connection
5 Button
6 LED
7 Binary inputs (A, B)
8 Button fan speed
9 LED fan speed 1-3
10 Button switch contact
11 LED switch contact
12 Button On/Off
13 LED On/Off
14 Output switch contact
15 Fan
16 Valve heating
17 Valve cooling
18 LED valve heating
19 Button valve heating
20 LED valve cooling
21 Button valve cooling
22 LED binary input A
23 Button binary input A
24 LED binary input B
25 Button binary input B

2.3 Dimension drawing



2.4 Assembly and installation

The FCA/S 1.1M is a modular installation device for fast installation in the distribution board on 35 mm mounting rails to DIN EN 60 715.

The connection to the bus is implemented using the supplied bus connection terminal.

The device is ready for operation after connection to the bus voltage.

Accessibility of the devices for the purpose operation, testing, visual inspection, maintenance and repair must be provided (conform to DIN VDE 0100-520).

Commissioning requirements

In order to commission the device, a PC with ETS (from ETS2 V1.3a or higher) as well as an interface to the ABB i-bus®, e.g. via a KNX interface, is required.

The device is ready for operation after connection to the bus voltage. No additional auxiliary voltage is required.

The installation and commissioning may only be carried out by electrical specialists. The appropriate norms, guidelines, regulations and specifications should be observed when planning and setting up electrical installations.

- Protect the device from damp, dirt and damage during transport, storage and operation.
- Only operate the device within the specified technical data!
- The device should only be operated in an enclosed housing (distribution board)!

Supplied state

The device is supplied with the physical address 15.15.255. The application program is pre-installed. It is therefore only necessary to load group addresses and parameters during commissioning. However the complete application program can be reloaded if required. A longer downtime may result if the application program is changed or after a discharge.

Download behaviour

Depending on the PC which is used, the progress bar for the download may take up to one and a half minutes before it appears due to the complexity of the device.

Assignment of the physical address

The assignment and programming of the physical address is carried out in the ETS.

Cleaning

If devices become dirty, they can be cleaned using a dry cloth. Should a dry cloth not remove the dirt, the devices can be cleaned using a slightly damp cloth and soap solution. Corrosive agents or solutions should never be used.

Maintenance

The device is maintenance-free. No repairs should be carried out by unauthorised personnel if damage occurs, e.g. during transport and/or storage. The warranty expires if the device is opened.

Foil keypad

The manual control keys may not be operated with pointed or sharp-edged objects, e.g. screwdrivers or pens. This may damage the keypad.

3 Commissioning

3.1 Overview

The application program ***Fan Coil-Actuator/1*** is available for the Fan Coil Actuator. Programming requires ETS2 version V1.3a or higher.

The following functions are available:

Additional output	For control of auxiliary electrical heating, e.g. in the Winter ⇔ Summer transition phase.
Fans	A 3 speed fan is controlled alternately with a two-way connection or with speed switching.
Valve Heating/Cooling	One valve for heating and one valve for cooling are controlled. The control of the valves can be implemented as PWM (constant) control or as 3-point control (opening and closing). The valve outputs are short circuit protected.
Binary input	Two binary inputs are available. These are used for example, to monitor the window contact and condensation (dew point).

The ABB i-bus® device FCA/S 1.1M with 6 A outputs is available for Fan Coil applications. This device can be operated manually e.g., for commissioning purposes.

This eliminates the danger of destruction of the fan motors by improper switching. The Fan Coil Actuator features relays in each output which are mechanically independent of the other outputs. Switching noises cannot be avoided due to the mechanical nature of the design.

The installation location of the Fan Coil Actuator can either be centrally in the distribution board or distributed in a Fan Coil unit. Normally the Fan Coil Actuator is used in conjunction with a room temperature controller for an individual room temperature control system. The room temperature controller sends a control variable which is used to control the fan speeds via the Fan Coil Actuator.

Fan Coil controls

- Fan with three fan speeds
- With changeover or step control
- 2-pipe system heating and cooling
- 2-pipe system heating or cooling
- 3 pipe system
- 4 pipe system

For further information see: Planning and applications, examples

Configuration design types

A Fan Coil unit can be configured as a compact device or a modular installation device:

- *Compact devices:* These are supplied with enclosures and are available as self-contained units for wall or ceiling mounting.
- *Modular installation devices:* These have no enclosures and are mounted in the wall, in the ceiling or in the floor. The air is blown into the room through a grill.

Air supply

Fan Coil units are available as recirculation or a mixed air devices.

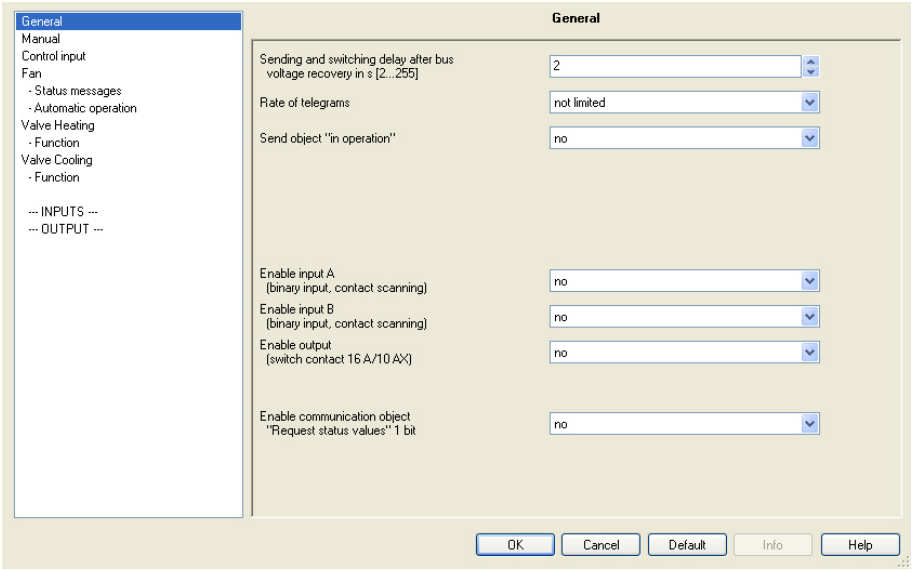
- *Recirculation devices*: The room air is directed past heat exchangers by the fans.
- *Mixed air devices*: The room air is mixed with fresh air. The mixing ratio between recalculated and fresh air can usually be adjusted.

3.2 Parameters

Note
The standard settings for the options are underlined, e.g. yes/no.

3.2.1 Parameter window *General*

Higher level parameters can be set in the *General* parameter window.



Sending and switching delay after bus voltage recovery in s [2…255]

Options: 2…255

Only telegrams are received during the send and switching delay. The telegrams are not processed however and the outputs remain unchanged. No telegrams are sent on the bus.

After the sending and switching delay, telegrams are sent and the state of the outputs is set to correspond to the parameterisation or the communication object values.

If communication objects are read during the sending and switching delay (e.g. by a visualisation system), these read requests are stored and a response is sent after the send and switching delay has been completed.

An initialisation time of about two seconds is included in the delay time. The initialisation time is the time that the processor requires to be functional.

How does the device behave with bus voltage recovery?

After bus voltage recovery the device always waits for the send delay time to elapse before sending telegrams on the bus.

Note
The set switch delay does not act on the electronic outputs (Valve Heating/Cooling)!

Rate of telegrams

Options: not limited
1/2/3/5/10/20 telegrams/second
0.05/0.1/3/5/10/20 telegrams/second

A telegram limitation is implemented to control the bus load created by the device.

Send object “in operation”

Options: no
yes

The *in operation* communication object indicates the correct function of the device on the bus. This cyclic telegram can be monitored by an external device.

Note
After bus voltage recovery the communication object is sent after the set sending and switching delay.

- *yes:* Additional parameters appear:

  Telegram repeated in s [1…65,535]

  Options: 1…60…65,535

  Here a time interval is set which the communication object *In operation* uses to cyclically send a telegram.

  Send value cyclically

  Options: 0
  1

  The setting defines the value that the communication object sends on the bus.

Enable input A
(binary input, contact scanning)

Enable input B
(binary input, contact scanning)

Options: no
yes

- *yes:* The input is activated. A further parameter window is provided on the left half of the window.

Note
The inputs are equipped as binary inputs with contact scanning. The scanning voltage is provided by the device.

Enable output
(switch contact 16 A/10 AX)

Options: no
yes

- *yes:* The output is activated. A further parameter window is provided on the left half of the window.

Enable communication object
"Request status values" 1 bit

Options: no
yes

- *yes:* A 1 bit communication object, *Request status values* is enabled.

Via this communication object all status messages can be requested provided that they have been parameterised with the option *after a change or request*.

With option *yes* the following parameters are visible:

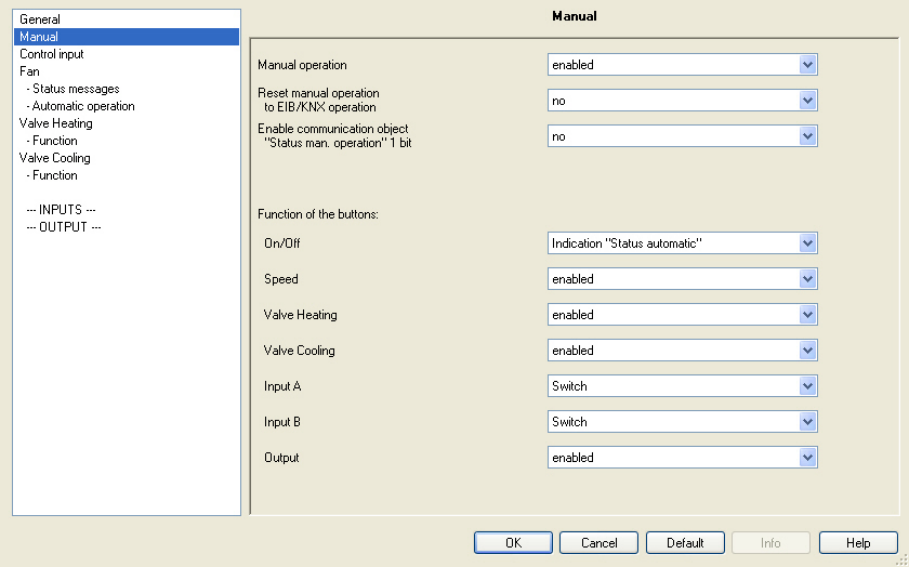
recall with object value

Options: 0
1
0 or 1

- *0:* The status messages are sent with the value 0.
- *1:* The status messages are sent with the value 1.
- *0 or 1:* The status messages are sent with the value 0 or 1.

3.2.2 Parameter window *Manual*

In the parameter window *Manual* all the settings for manual operation can be made.



Function of manual operation

After connection to the bus the device is in *EIB/KNX mode*. The LED is off. All *LEDs* indicate the actual input state. The respective *Buttons* are non-functional. It is possible to switch between *manual operation* and *EIB/KNX operation* by pressing the button.

If manual operation is activated, the current fan speed remains set and can only be operated manually. Here any limitations, forced operations and programmed dwell times are not considered.
If manual operation is deactivated, the fan sets to a speed to which it would also be set without manual operation, e.g. via the object values. The setting occurs with the parameterised dwell times!

Switch on of manual operation:
Press button until the yellow LED lights continuously.

Switch off of manual operation:
Press button until LED no longer lights.

Note
If button is released again before two seconds have elapsed, the LED reverts to its old state and there is no reaction. If manual operation is disabled via the application program, there is no reaction and the device remains in the EIB/KNX operation state. If it has been disabled, LED is switched on or over after it has flashed for three seconds.

Manual operation

Options: enable/disable via communication object
<u>enabled</u>
disabled

This parameter defines if the switch over between the operating states *manual operation* and *EIB/KNX operation* is enabled or disabled via the button on the device.

- *via object enable/disable:* The communication object *enable/block manual operation – Manual operation* appears.

Telegram value 0 = enable button
1 = disable button

Note
The manual operation overwrites the input states.

Reset manual operation to EIB/KNX operation

Options: <u>no</u>
after 1/3/10/30 minute(s)

This parameter determines how long the Fan Coil Actuator remains in the *Manual operation* mode after pressing the button.

- *no:* The Fan Coil Actuator remains in *Manual operation* until the button is pressed again.
- *after X minutes:* The Fan Coil Actuator remains in *Manual operation* after the last button push until either button is pushed again or the programmed time has timed out.

Enable communication object "Status man. operation" 1 bit

Options: <u>no</u>
yes

- *yes:* An additional parameter appears:

  Send object value

  Options: no, update only
  <u>after a change</u>
  after request
  after a change or request

  - *no, update only*: The status is updated but not sent (it can be read via the communication object).
  - *after a change:* The status is sent after a change.
  - *after request:* The status is sent after a request.
  - *after a change or request:* The status is sent after a change or a request.

Function of the buttons and LEDs:

Note
The respective LED indicates the current input states. The foil keypad can be operated when manual operation has been activated. If group addresses have been assigned, telegrams will be sent on the bus! Any signal changes from the installed system will not be considered. With switchover to the operating state *EIB/KNX operation* the respective LED again indicates their current input states. The communication objects are updated and telegrams are sent.

On/Off

Options: Indication "Status automatic“
LED/button with objects

Note
The button has no further functions with both options.

- *Indication “Status automatic”:* The LED indicates the current operating state of the Fan Coil Actuator:

  LED on = fan automatic activated

  LED off = fan automatic not activated

- *LED/button with objects:* The communication objects *LED On/Off* (No.: 3) and *Button On/Off* (No.: 2) appear.
  With these communication objects it is possible to freely select the function.

Speed

Options: enabled
disabled

With this parameter the button can also be enabled or disabled.

- *Enabled:* The button is enabled.
- *Disabled:* The button is disabled.

Note
By pressing the Speed button the individual fan speeds can be switched through. This is according to the following sequence: 0 => 1 => 2 => 3 => 0 => 1 => 2 => 3 =>… The speeds are indicated in the sequence: Fan speed 1 with LED , Fan speed 2 with LED , Fan speed 3 with LED . If none of the speed LEDs are on the fan is off.

Valve Heating

Options: enabled
disabled

With this parameter the button can be enabled or disabled.

- *Enabled:* The button is enabled.
- *Disabled:* The button is disabled.

Note
By pressing the Valve Heating button the connected valve is controlled. The indicator LED heating indicates the same value as the 1 bit status of the valve control: 0 = LED heating off/Valve position equal to zero 1 = LED heating on/Valve position equal to zero With a state change the new state is immediately indicated.

Function with manual operation

A target position which may not have yet been achieved is approached. The first button push switches the target position to 0 % if it was unequal to 0 % beforehand and 100 % if it was unequal to 100 % beforehand. Thereafter each push of a button toggles the target position between 0 % and 100 %. During deactivation of manual operation the target position is approached which would have been set without manual operation.

With manual operation a set characteristic curve correction is not taken into consideration.

Button function at overload

The description concerns the button heating and the LED heating.

The device is in the EIB/KNX operating state and the output of the heating valve has a malfunction, e.g. due to an overload.

The valve heating LED flashes with a frequency of five hertz to indicate the malfunction. At the same time the communication object *Overload Valve Heating* sends the value 1. With the respective valve heating button the fault is acknowledged and the communication object *Overload Valve Heating* is reset to the value 0. In order to acknowledge the fault with the valve heating button the button must be pressed for longer than four seconds.

If the fault continues to persist after acknowledgement, the valve heating button indicates it by flashing (five hertz). The communication object *Overload Valve Heating* is set again to the value 1.

Note
If the device is in *Manual operation* the acknowledgement function *Overload* is not available.

Valve Cooling

The operation of the cooling valve does not differ from the operation of the heating valve.

For further information see: description Valve Heating

Input A

Options: Block
Switch
Button

With this parameter the button can be disabled, or programmed as a switch or push button.

- *Block:* The button is disabled.
- *Switch:* With every actuation the states of the input and the LED are changed.
- *Push buttons:* Press button => input closed, LED on
  Release button => input opened, LED off

Note
By pressing button A binary input A is simulated. The display indicates the current input state. The parameterised features are executed.

Input B

The operation of input A does not differ from the operation of input B.

Output

Options: enabled
disabled

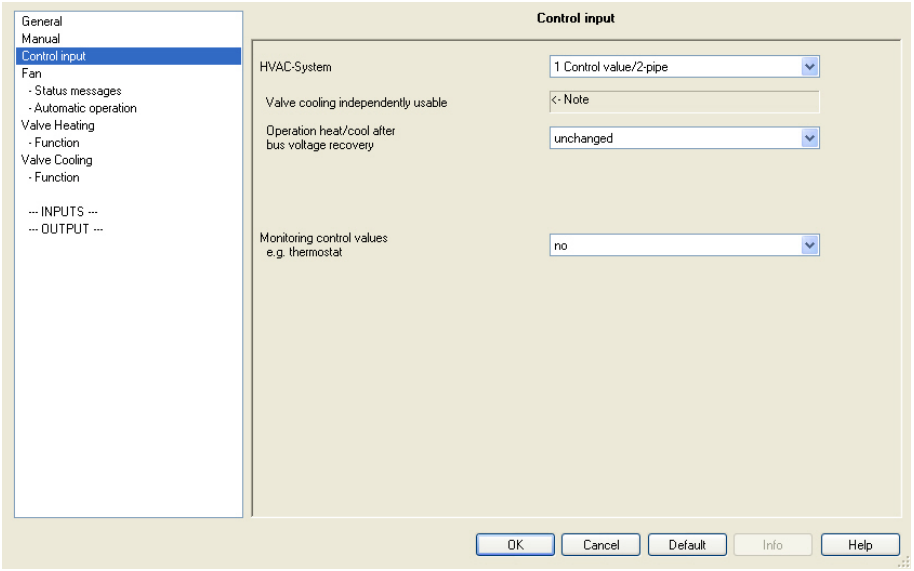
With this parameter the button can be enabled or disabled.

- *Enabled:* The button is enabled.
- *Disabled:* The button is disabled.

Note
By pressing the button the relay is toggled. The display indicates the contact state. LED on = contact closed LED off = contact opened

3.2.3 Parameter window *Control input*

In this parameter window all settings for the *Control input* are undertaken.



HVAC-System

Options: 1 Control value/2-pipe
1 Control value/4-pipe, with switching object.
2 Control values/2-pipe
2 Control values/2-pipe, with switching object
2 Control values/4-pipe

This parameter determines how the control of the control value on the Fan Coil Actuator is sent via the bus. Thus the FCA/S 1.1M is optimally matched.

The individual functions are described in the following chapters.

Important
If a valve is deactivated due to a conversion of the HVAC system, the valve will be fully closed. A correction curve which may be set will be ignored! A deactivated valve can only be moved by manual operation! The manual operation always fully opens and closes the valve. A correction curve which may be set will be ignored!

Monitoring control values e.g. thermostat

Options: no
yes

- *yes:* The communication object *Fault control value* is monitored and the monitoring of the input/control value(s) can be undertaken. Hereby for example, a thermostat can be monitored.

Note
During a fault when the control signal from the thermostat is no longer received, the Fan Coil Actuator autonomously performs a PWM calculation (pulse width modulation). For this purpose the Fan Coil Actuator uses the programmable PWM cycle time

With option *yes* in parameter *Monitoring control values, e.g. Thermostat,* further parameters become visible:

Monitoring time in s [30…65,535]

Options: 30…120…65,535

With this parameter the time is set with which the telegrams on the input/setting values of the FCA/S are monitored:
Communication objects *Control value heating*, *Control value cooling* or *Control value heating/cooling*.

If a setting variable is not received within the parameterised time, a communication malfunction or a defective thermostat is the cause.

The reaction of the FCA/S to a setting value not received can be defined in the following parameters.

Send object value (Object "Control value fault" 1 bit)

Options: no, update only
after a change
after request
after a change or request

- *no, update only*: The status is updated but not sent.
- *after a change:* The status is sent after a change.
- *after request:* The status is sent after a request.
- *after a change or request:* The status is sent after a change or a request.

Control value after control fault in % [0…100]

Options: 0…30…100

This control value in percent can be set with a control value fault should the control fail.

3.2.3.1 HVAC-System – *1 Control value/2-pipe*

If option *1 Control value/2-pipe* is selected, additional parameters appear:

Valve cooling independently usable

This parameter serves as a note or remark.

Valve cooling

The cooling valve can be used additionally and independently via the communication object *Control value, Cooling (extra!)*.
The valve cooling is not monitored in the process.

Valve heating

Via communication object *Control value heating/cooling* the valve heating and the fan are controlled.

Operation heat/cool after bus voltage recovery

Options: unchanged
Heating
Cooling

Using this parameter the reaction after bus voltage recovery is set.

- *Unchanged:* After bus voltage recovery the state which existed before bus voltage failure is set.
- *Heating:* After bus voltage recovery the *Heating* state is set.
- *Cooling:* After bus voltage recovery the *Cooling* state is set.

3.2.3.2 HVAC-System – *1 Control value/4-pipe, with switching object*

If option *1 Control value/4-pipe with switching object* is selected, additional parameters appear:

Toggle Heating/Cooling via object

This parameter serves as a note or remark.

> **Valve Heating/Cooling**
>
> Via the communication object *Control value heating/cooling* the valve cooling/heating and the fan are controlled.
>
> Toggle between heating and cooling is implemented via the communication object *Toggle, Heating/Cooling*.
>
> The corresponding inactive/non-actuated valve is thus automatically closed when toggled.

Operation heat/cool after bus voltage recovery

Options: unchanged
Heating
Cooling

Using this parameter the reaction after bus voltage recovery is set.

- *Unchanged:* After bus voltage recovery the state which existed before bus voltage failure is set.
- *Heating:* After bus voltage recovery the *Heating* state is set.
- *Cooling:* After bus voltage recovery the *Cooling* state is set.

Object value for heating the object "Toggle heating/cooling"

Options: 1
0

With this parameter you set the object value used to toggle between heating and cooling.

- *1:* As soon as a telegram with the value 1 is received, heating is activated and cooling deactivated.
- *0:* As soon as a telegram with the value 0 is received, heating is activated and cooling deactivated.

3.2.3.3 HVAC-System – *2 Control values/2-pipe*

If option *2 Control values/2-pipe* is selected, additional parameters appear:

Toggle Heating/Cooling via automatically controlled value

This parameter serves as a note or remark.

Valve Heating/Valve Cooling

Via the communication object *Control value, heating* the heating valve is controlled.

Via the communication object *Control value, cooling* the cooling valve is controlled.

Toggling between heating and cooling is implemented by updating the control values. The heating/cooling status is then set accordingly.

If a control with a value > 0 is received, the fan and the cooling valve are controlled. The heating valve remains closed.

If a control value with a value = 0 is received, this is ignored if the other control value > 0.

Operation heat/cool after bus voltage recovery

Options: unchanged
Heating
Cooling

Using this parameter the reaction after bus voltage recovery is set.

- *Unchanged:* After bus voltage recovery the state which existed before bus voltage failure is set.
- *Heating:* After bus voltage recovery the *Heating* state is set.
- *Cooling:* After bus voltage recovery the *Cooling* state is set.

3.2.3.4 HVAC-System – *2 Control values/2-pipe, with switching object*

If option *2 Control values/2-pipe, with switching object* is selected additional parameters appear:

Toggle Heating/Cooling via object

This parameter serves as a note or remark.

Valve Heating/Valve Cooling

Via the communication object *Control value, heating* the heating valve is controlled.

Via the communication object *Control value, cooling* the cooling valve is controlled.

Toggle between heating and cooling is implemented via the communication object *Toggle, Heating/Cooling*. The heating/cooling status is set.

The fan and the valve cooling are controlled. The heating valve remains closed

Operation heat/cool after bus voltage recovery

Options: unchanged
Heating
Cooling

Using this parameter the reaction after bus voltage recovery is set.

- *Unchanged:* After bus voltage recovery the state which existed before bus voltage failure is set.
- *Heating:* After bus voltage recovery the *Heating* state is set.
- *Cooling:* After bus voltage recovery the *Cooling* state is set.

Object value for heating the object "Toggle heating/cooling"

Options: 1
0

With this parameter you set the object value used to toggle between heating and cooling.

- *1:* As soon as a telegram with the value 1 is received, heating is activated and cooling deactivated.
- *0:* As soon as a telegram with the value 0 is received, heating is activated and cooling deactivated.

3.2.3.5 HVAC-System – *2 Control values/4-pipe*

If option *2 Control values/4-pipe* is selected additional parameters appear:

Toggle Heating/Cooling via automatically control value

This parameter serves as a note or remark.

Valve Heating/Valve Cooling

Via the communication object *Control value, heating* the heating valve is controlled.

Via the communication object *Control value, cooling* the cooling valve is controlled.

Toggling between heating and cooling is implemented by updating the control values. The heating/cooling status is then set accordingly.

If a control with a value > 0 is received, the fan and the corresponding valve are controlled.

The other valve is closed.

If a control value with a value = 0 is received, this is ignored if the other control value > 0.

Operation heat/cool after bus voltage recovery

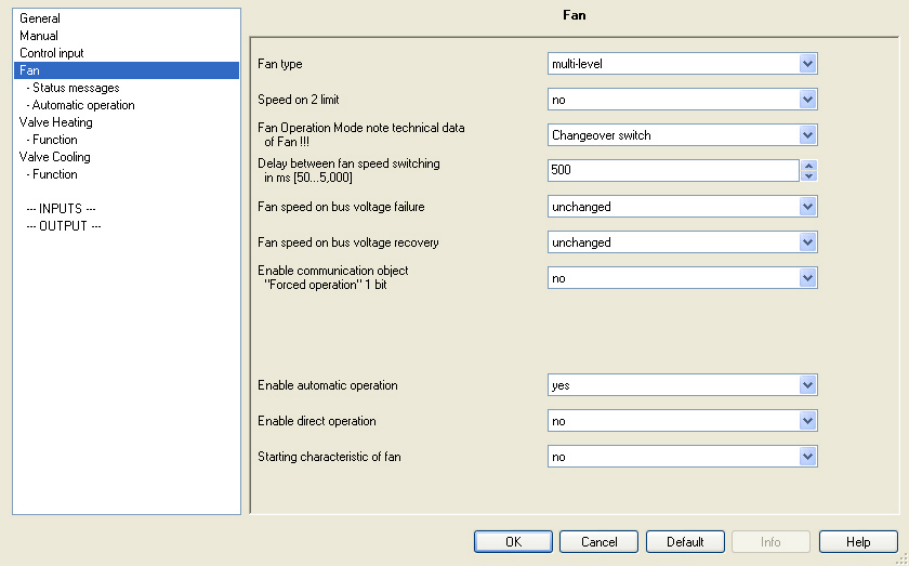
Options: unchanged
Heating
Cooling

Using this parameter the reaction after bus voltage recovery is set.

- *Unchanged:* After bus voltage recovery the state which existed before bus voltage failure is set.
- *Heating:* After bus voltage recovery the *Heating* state is set.
- *Cooling:* After bus voltage recovery the *Cooling* state is set.

3.2.4 Parameter window *Fan – multi-level*

In this parameter window all settings for the *Multi-level fan* are undertaken.



Fan type

Option: multi-level
one-level

This parameter defines the fan type which is to be controlled. The fan type to be controlled is set with this parameter.

- *multi-level:* A fan with two or three speeds is to be controlled.
- *one-level:* A fan with one speed is to be controlled.

Speed on 2 limit

Option: no
yes

The fan speeds can be limited to 2 here. The following settings are the same as those for a three speed fan, but are only limited to two speeds.

- *no:* A three speed fan is controlled.
- *yes:* A two speed fan is controlled via fan speeds 1 and 2.
  Fan speed 3 is non-functional.

Fan Operation Mode note technical data of Fan!!!

Option: Changeover switch
Step switch

The control of the fan is set with this parameter. The mode of fan control should be taken from the technical data of the fan. Usually the fans are actuated with a changeover switch.

How does a two-way changeover circuit function?

Only one output is switched on with the parameterisation as a changeover switch, i.e. the second fan speed is set so that only the second input of the fan is switched on. The FCA/S switches on the second relay of the fan group.

The delay time between the speed switch over and a minimum dwell time in a valve speed are programmable. The minimum dwell time in a fan speed is only active in automatic mode.

How does speed switching function?

On a step switch all the previous outputs are switched on, i.e. the second fan speed is set so that the first and second input of the fan is switched on. The FCA/S switches on the first and second relay of the fan group. With step switch control, no erratic and sudden switch on of the fan is possible. The small fan speeds are activated consecutively (outputs switched on) until the required fan speed is achieved.

The parameterised delay time between two fan speeds has the effect that the current fan speed must be switched on for at least this time before the next valve speed is switched on. The parameterised minimum dwell time in a fan speed has the same effect as a changeover switch, i.e. it is only active in automatic mode and is added to the switchover delay.

Delay between fan speed switching in ms [50…5,000]

Option: 50…<u>500</u>…5,000

A switchover delay can be programmed with this parameter. As this time is a fan specific factor, it is always considered in automatic mode as well as with manual switching or during the start-up phase.

Note
The minimum switching time of the relay must be considered, see technical data.

What is a switchover delay?

Some ventilation equipment requires a switchover delay between speed changes (contact change). This delay corresponds to the delay time, in which the current fan speed is switched off and the next fan speed is not yet switched on. The necessary delay is a fan specific factor and can be taken from the technical data of the fan.

What effect does the switchover delay have with a changeover switch?

On a fan with a changeover switch, this delay time defines the time delay in which the current fan speed (contact) has been switched off but the next fan speed has not yet been switched on. The delay time is entered in ms.

Fan speed on bus voltage failure

Option: unchanged
off

Hereby, the behaviour of the fan with a bus voltage failure is defined.

Fan speed on bus voltage recovery

Options: unchanged
off
1
2
3

Hereby, the behaviour of the fan with a bus voltage recovery is defined.

- *Unchanged:* The outputs and the fan speeds of the fan remain unchanged.
- *Off:* The fan is switched off.
- *1, 2 or 3:* The fan switches to fan speed 1, 2 or 3.

Caution

The FCA/S 1.1M is supplied ex-works with a default setting (factory default). This ensures the fan setting is switched off when the bus voltage is applied to the relay for the first time. Thus, damage to the device due to unintentional switch on during transport, e.g. due to vibration, is avoided.

It is advisable to apply a bus voltage before connecting the fan in order to achieve a defined switch state of the fan. This eliminates the possibility of the destruction of the fan due to an incorrect contact setting.

Enable communication object "Forced operation" 1 bit

Options: no
yes

- *yes:* A 1 bit *Forced operation* communication object is enabled. At the same time further parameters are made visible:

  Forced operation on object value

  Options: 1
  0

  This parameter defines how to react to a telegram.

  - *1:* Forced operation is activated by a telegram with value 1.
  - *0:* Forced operation is activated by a telegram with value 0.

Note
During forced operation the limitations set in *Automatic operation* are ignored. They become active again after forced operation is ended.

  Limitation with forced operation

  Options: 3, 2, 1, off
  unchanged
  OFF
  1
  1, off
  2
  2, 1
  2, 1, off
  3
  3, 2
  3, 2, 1

  This parameter sets which fan speed is set with active forced operation or which may not be exceeded or undershot.

Enable automatic operation

Options: no
yes

- *yes: Automatic operation* is enabled. Then an additional *Automatic operation* parameter window appears.

Enable direct operation

Options: no
yes

- *yes: Direct operation* is enabled. Then an additional *Direct operation* parameter window appears.

Starting characteristic of fan

Options: no
yes

This parameter enables the fan to start from the OFF state with a defined fan speed. This fan speed is immediately applied.

In order to guarantee a safe start of the fan motor, it can be useful to start the fan motor first with a higher fan speed. Thus a higher torque for the start up phase of the fan is achieved.

Note
A step switch normally means however that the previous fan speeds are usually switched on consecutively. With the changeover switch the fan speed is immediately switched on.

The delay between the switch-over of two fan speeds (contact change) is considered.

The dwell times in a fan speed which are considered in automatic mode, are inactive and will only be considered after the start up phase.

The start-up behaviour is a technical characteristic of the fan. For this reason this behaviour has a higher priority than an active limitation or forced operation.

With the option *yes* in the parameter *Starting characteristic of fan* the additional two parameters become visible:

Switch on over fan speed

Options: 1/2/3

Here you set which fan speed the fan uses to start from the OFF state.

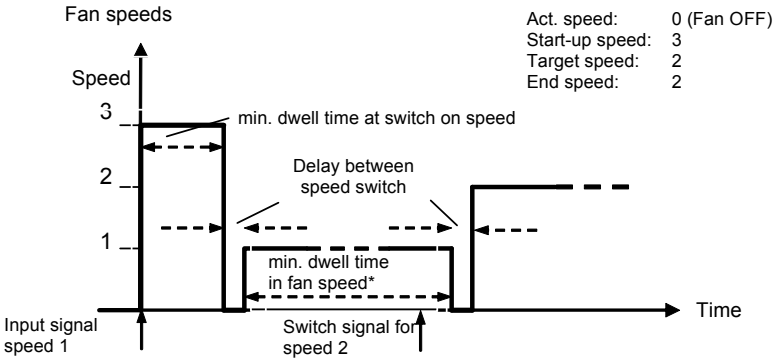
Minimum dwell period in switch on fan speed in s [1…65,535]

Options: 1…5…65,535

This parameter defines the minimum dwell time in one of the switch on speeds.

Example: Starting behaviour of a three speed fan

The illustration shows the response in automatic operation with the option *Switch on over fan speed 3*, if the fan receives the command from the OFF state to set *Speed 1*.

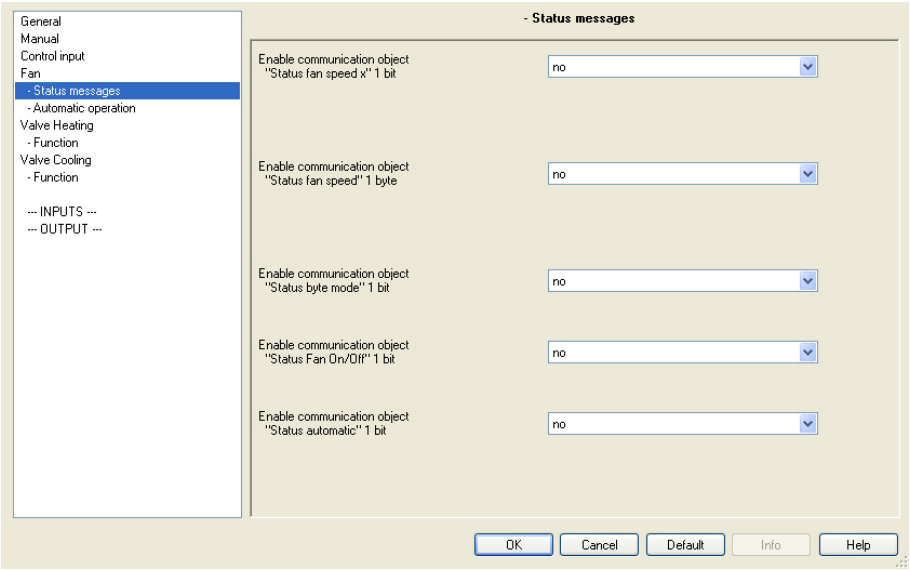


* The parameter *Minimum dwell period in fan speed in s [0…65,535]* in the parameter window *Automatic operation* is only active and programmable, if the option *yes* has been selected in the *Enable automatic operation* parameter. In the parameter window *Fan* you can find the parameter *Enable automatic operation*.

Important
In manual mode the forced operation, the four limitations, e.g. frost/heat protection and the switchover delays continue to apply and should be considered. The parameterised dwell time for automatic mode is ignored during manual operation.

3.2.4.1 Parameter window *Status messages*

In this parameter window the *Status messages* are defined.



Enable communication object "Status fan speed x" 1 bit

Options: no
yes

- *yes:* Three 1 bit communication objects, *Status fan speed x*, x = 1 to 3 are enabled.

The setting of a fan speed is displayed via these communication objects. You can parameterise the status to indicate a current fan speed or a required fan speed.

What is the current fan speed?

The *current fan speed* is the speed at which the fan is actually operating.

What is the required fan speed?

The *required fan speed* is the fan speed which has to be achieved, e.g. when the transition and dwell times are completed.

With option *yes* the following parameters are visible:

Meaning

Options: current fan speed
required fan speed

This parameter defines whether the status of the *current fan speed* or the *required fan speed* is displayed.

Note
The limitations are included in this observation, i.e. if a limitation allows only fan speed 2, the fan is operating in fan speed 2 and for example a telegram to switch up is received, the *required fan speed* remains at 2 as fan speed 3 cannot be achieved due to the limitation.

Send object value

Options: no, update only
after a change
after request
after a change or request:

- *no, update only*: The status is updated but not sent.
- *after a change:* The status is sent after a change.
- *after request:* The status is sent after a request.
- *after a change or request:* The status is sent after a change or a request.

Enable communication object "Status fan speed" 1 byte

Options: no
yes

- *yes:* The communication object *Status fan speed* is enabled.

This status byte defines the figure value of the fan speed.

This display can be differentiated with the selection of *current fan speed* from the *required fan speed*. Initially the switchover times, dwell times and the start-up phase must be completed before the *required fan speed* is achieved.

What is the current fan speed?

The *current fan speed* is the speed at which the fan is actually operating.

What is the required fan speed?

The *required fan speed* is the fan speed which has to be achieved, e.g. when the transition and dwell times are completed.

With option *yes* the following parameters are visible in the parameter:

Meaning

Options: current fan speed
required fan speed

This parameter defines whether the status of the *current fan speed* or the *required fan speed* is displayed.

Note
The limitations are included in this observation, i.e. if a limitation allows only fan speed 2, the fan is operating in fan speed 2 and for example a telegram to switch up is received, the *required fan speed* remains at 2 as fan speed 3 cannot be achieved due to the limitation.

Send object value

Options: no, update only
after a change
after request
after a change or request:

- *no, update only*: The status is updated but not sent.
- *after a change:* The status is sent after a change.
- *after request:* The status is sent after a request.
- *after a change or request:* The status is sent after a change or a request.

Enable communication object "Status byte mode" 1 byte

Options: no
yes

- *yes:* The communication object *Status byte mode* is enabled.

From this status byte the states heating, cooling, automatic, forced operation and the four limitations are indicated directly via a 1 bit coding.

For further information see: Status byte code table

With option *yes* a further parameter is visible:

Send object value

Options: no, update only
after a change
after request
after a change or request

- *no, update only*: The status is updated but not sent.
- *after a change:* The status is sent after a change.
- *after request:* The status is sent after a request.
- *after a change or request:* The status is sent after a change or a request.

Enable communication object "Status Fan On/Off" 1 bit'

Options: no
yes

The communication object *Fan status* can be enabled with this parameter.

Some fans initially require an ON command before they are set to a fan speed from the OFF state. This ON command has effects a main switch which has to be switched on.
This demand can be implemented with any switch output which is controlled via the *Fan status* communication object. The corresponding switch object of the switch actuator should be connected with the *Fan status* communication object.

With option *yes* a further parameter is visible:

Send object value

Options: no, update only
after a change
after request
after a change or request

- *no, update only*: The status is updated but not sent.
- *after a change:* The status is sent after a change.
- *after request:* The status is sent after a request.
- *after a change or request:* The status is sent after a change or a request.

The following parameter only becomes visible if the option *yes* has been selected in the *Enable automatic operatio*n parameter in the *Fan* parameter window.

Enable communication object "Status automatic" 1 bit

Options: no
yes

The communication object *Status Automatic* is enabled with this parameter.

Telegram value 1 = fan coil actuator is in automatic operation
0 = automatic operation switched off

- *yes:* An additional parameter becomes visible:

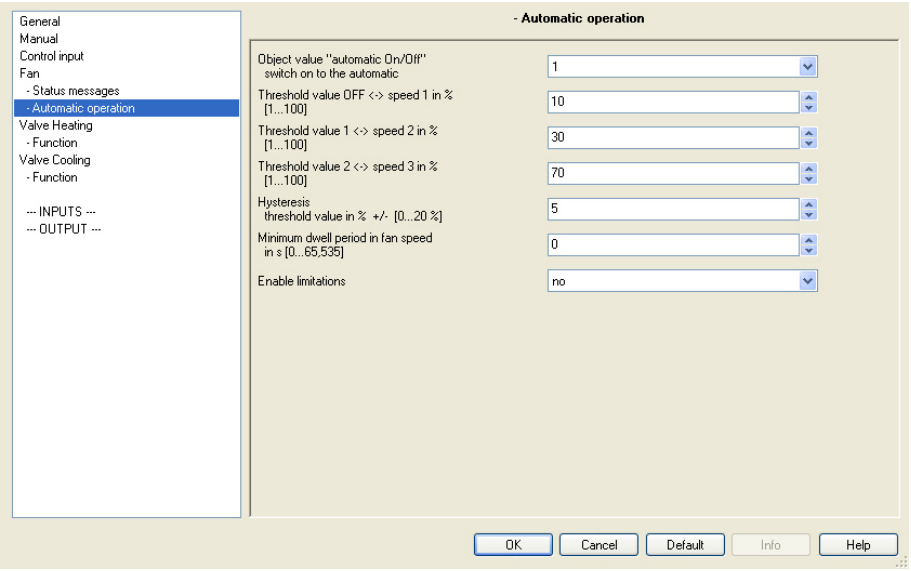
  Send object value

  Options: no, update only
  after a change
  after request
  after a change or request

  - *no, update only*: The status is updated but not sent.
  - *after a change:* The status is sent after a change.
  - *after request:* The status is sent after a request.
  - *after a change or request:* The status is sent after a change or a request.

3.2.4.2 Parameter window *Automatic operation*

This parameter window is visible if in parameter window *Fan* the option *yes* has been selected in the *Enable automatic operation* parameter.



In this parameter window the threshold values for switchover of the fan speed are defined. Furthermore, the limitations can also be enabled.

The corresponding valve control objects receive the value 1 if a fan speed is set. If a fan speed is not set the communication object will receive the value 0.

Important
The Fan Coil Actuator evaluates the threshold values in ascending order, i.e. first of all the threshold value for *OFF <-> Fan speed 1* is checked followed by *Fan speed 1 <-> Fan speed 2* etc. The correct method of function is only assured if the threshold value for *OFF <-> Fan speed 1* is less than the threshold value *Fan speed 1 <-> Fan speed 2* and this is less than *Fan speed 2 <-> Fan speed 3, etc.*

Object value “automatic On/Off” switch on to the automatic

Options: 1
0

This parameter defines how to react to a telegram.

- *1:* Automatic is activated by a telegram with value 1.
- *0:* Automatic is activated by a telegram with value 0.

Threshold value OFF <-> speed 1 in %
[1…100]

Options: 1…10…100

Here the threshold value is set at which switch on of fan speed 1 occurs. If the value in the control value object is greater than or equal to the parameterised threshold value, fan speed 1 is switched on. If the value is less, it is switched off.

Threshold value 1 <-> speed 2 in %
[1…100]

Options: 1…30…100

Here the threshold value is set at which switch over to fan speed 2 occurs. If the value in the control value object is greater than the parameterised threshold value, switch over to fan speed 2 occurs.

Threshold value 2 <-> speed 3 in %
[1…100]

Options: 1…70…100

Here the threshold value is set at which switch over to fan speed 3 occurs. If the value in the object *Control Value, Heating* or *Control Value, Cooling* is greater than the parameterised value, switch over to fan speed 3 occurs.

Hysteresis
threshold value in % +/- [0…20 %]

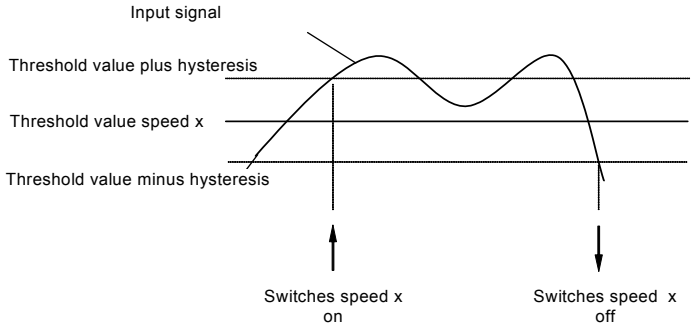
Options: 0…5…20

Here a hysteresis is set from which switchover to the next fan speed occurs. The hysteresis applies for all three threshold values.

The setting 0 causes immediate switching without hysteresis.

The entered percentage value is directly added to or subtracted from the percentage value of the *Fan speed x threshold value*. The result equals the new upper or lower threshold.

Example: three speed fan, hysteresis with fan control



Using hysteresis a continuous switching between the fan speeds around the threshold value with deviating input signals can be avoided.

Important

How does the fan react if the switch thresholds overlap by the use of hysteresis?

1) The hysteresis defines from which point the set speed transition occurs.
2) If the speed transition occurs, the new speed is determined using the control value and the set switch thresholds. The hysteresis is not considered.
3) A control variable with the value 0 always results in speed 0.

An example:

Parameterised: Threshold value OFF <-> speed 1 = 10 %
Threshold value 1 <-> speed 2 = 20 %
Threshold value 2 <-> speed 3 = 30 %
Hysteresis 15 %

Behaviour when ascending from speed 0:
– Speed 0 transitions at 25 % (≥ 10 % + hysteresis).
– The new speed is 2 (25 % is between 20 and 30 %).
– Accordingly speed 1 is omitted.

Behaviour when ascending from speed 3:
– Speed 3 transitions at 14 % (< 30 % + hysteresis).
– The new speed is 1 (15 % is between 10 and 20 %).
– Accordingly speed 2 is omitted.

Minimum dwell period in fan speed in s [0…65,535]

Options: 0…30…65,535

This parameter defines the dwell time for a fan in the fan speed until is switches to the next higher or lower fan speed. The input is made in seconds.

A setting of 0 indicates non-delayed switching. The minimum switching time of the relay should be taken from the Technical data.

The dwell time in a fan speed is only considered in automatic mode.

Important
The time is set to the value 0 with manual switching.

Enable limitations

Option: no
yes

- *yes:* Further parameters become visible.

At the same time 4 communication objects for limitation of the fan speed are enabled:

- *Limitation 1,* e.g. for standby operation
- *Limitation 2,* e.g. for night shutdown
- *Limitation 3,* e.g. for comfort operation
- *Limitation 4*, e.g. for frost/heat protection

Speed ranges (limitations) are defined for the fan with the speed limitation function which may not be exceeded or undershot.

Four limitations are available. This can be used for example for the control of various operating modes, e.g. frost/heat protection, night shut down and standby. In normal cases the thermostat takes these operating modes into account in its control variable for the actuator.

Important
The parameterised starting behaviour which is a technical characteristic of the fan has a higher priority than a limitation or forced operation, i.e. if a limitation is activated in fan speed 2 and a start-up behaviour is parameterised via fan speed 3, the following behaviour will result: The fan is in the OFF state and receives a control signal for fan speed 1. Initially the fan operates at fan speed 3 (start-up speed) and then proceeds to fan speed 2 which is defined by the limitation. The actual required fan speed 1 will not be achieved due to the limitation.

The sequence of the displayed parameters corresponds with their priorities, i.e. the parameters with the highest priority has limitation 1 followed by limitation 2, 3 and 4.

Note
The fault operation, e.g. as with a malfunction of the thermostat has a lower priority than the fan limitation, i.e. by a limitation of the fan speed during a thermostat malfunction only the upper or the lower limit of the fan limitation can be set at best.

When automatic mode is exited, e.g. by a manual action, the limitations 1 to 4 remain.

The following points apply for limitations:

- The fan speed and valve position can be parameterised independently.
- The limitation need not necessarily apply to one fan speed only. It can also encompass another range of the fan speeds, i.e. only certain fan speeds can be set if the limitation is active. Hereby a limited control is also possible.
- The limitation is activated if a telegram with the value 1 is received on the limitation object. The limitation is deactivated if a telegram with the value 0 is received on the limitation object. A manual action ends automatic mode.
- If a limitation is activated the Fan Coil Actuator switches to the parameterised fan speed regardless of the control value. If during the activation of the limitation another fan speed or a fan speed outside the range of the “limitation range” is set, the required fan speed or the limit fan speed of the range is set.
- After switch off of the limitations, the fan speed and the communication objects for valve control are recalculated and executed, i.e. during limitation the actuator operates normally in the background, the outputs are not changed and implementation only occurs after the end of limitation.

There are the same parameters for each of the individual four limitations used to limit the fan speeds. The priority is according to the listed sequence. The highest priority is assigned to limitation 1, e.g. Frost/Heat protection; the lowest priority is assigned to limitation 4, e.g. standby operation.

Speed with limitation 1
Speed with limitation 2
Speed with limitation 3
Speed with limitation 4

Options: 3, 2, 1, off
unchanged
OFF
1
1, off
2
2, 1
2, 1, off
3
3, 2
3, 2, 1

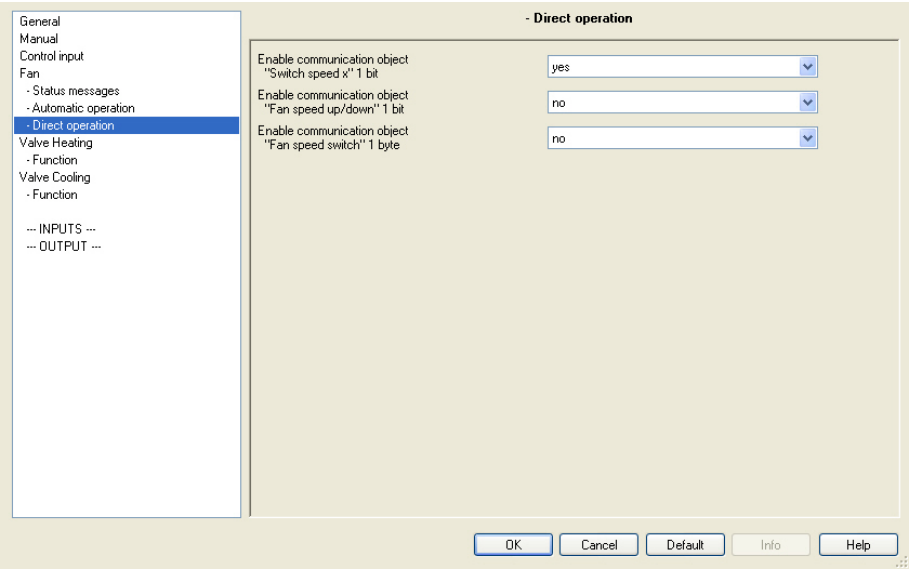
With this parameter you set which fan speed is set with active limitation or not exceeded or undershot.

- *3, 2, 1, off:* Everything is possible.
- *Unchanged:* The state is retained.
- *OFF:* Off.
- *1:* limited to speed 1.*
- *1, off* limited to speed 1 and off.
- *2:* limited to speed 2.*
- *2, 1:* limited to speed 2 and 1.
- *2, 1, off:* limited to speed 2, 1 and off.
- *3:* limited to speed 3.*
- *3, 2*: limited to speed 3 and 2.
- *3, 2, 1:* limited to speed 3, 2 and 1.

* The control value is ignored.

3.2.4.3 Parameter window *Direct operation*

This parameter window is visible if in parameter window *Fan* the option *yes* has been selected in the *Direct operation* parameter.



Through forced operation for example, recirculation, valve off and fan on can be implemented.

In manual mode the programmed dwell time in a fan speed is ignored in order to detect an immediate reaction in manual operation. The transition time between two fan speeds remains active in order to protect the fan.

Enable communication objects "Switch speed x" 1 bit

Options: no
yes

- *yes:* Three 1 bit communication objects *Speed 1*, *Speed 2* and *Speed 3* are enabled.

The Fan Coil Actuator receives a setting command via these communication objects. The FCA/S calculates the fan control and switches the corresponding outputs based on these limitations.

Telegram value 1 = Fan speed x is switched on
0 = Fan speed x is switched off

If several ON commands are received consecutively in a short period of time at various *Fan speed x* objects, the value last received by the fan control is the decisive value. This also applies for the OFF command. If the switched off fan speed again receives an OFF command it is carried out, i.e. another speed switched on at this time will be switched off and the command received last – in this case an OFF command – will be implemented.

Important
The forced operations, the four limitations and the switch over delays are still valid and are considered. The parameterised dwell time for automatic mode is ignored during manual operation.

Enable communication object
”Fan speed up/down" 1 bit

Options: no
yes

- *yes:* A 1 bit *Fan speed up/down* communication object is enabled.

Telegram value 1 = a fan speed is switched up
0 = a fan speed is switched down

If the maximum fan speed is achieved and a further telegram with the value 1 is received the fans speed will remain as it is.

Important
The forced operations, the four limitations and the switch over delays are still valid and are considered. The parameterised dwell time for automatic mode is ignored during manual operation.

With multiple manual up or down switching operations the target speed will be increased or reduced by a speed step. This is possible until the maximum or minimum possible speed is achieved. Further up or down commands are ignored and not executed. Each new switching command initiates a new calculation of the target speed. This means that a target speed changes by a switching command until this is achieved.

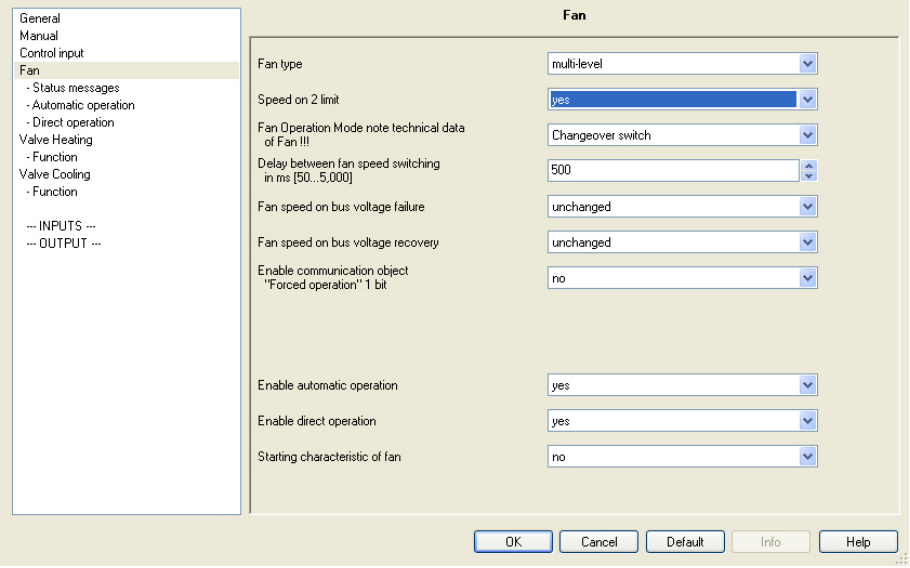
Enable communication object
"Fan speed switch" 1 byte

Options: no
yes

- *yes:* A 1 byte *Switch speed* communication object is enabled.

3.2.5 Parameter window *Fan – Two-level*

In this parameter window all settings for the *Two-level fan* are undertaken.



If a fan with two fan speeds is to be controlled via the FCA/S 1.1M, the following parameters must be set:

In the *Fan* parameter window the option *multi-level* must be selected for the *Fan type* parameter.

The parameter *Speed on 2 limit* must be selected with *yes*.

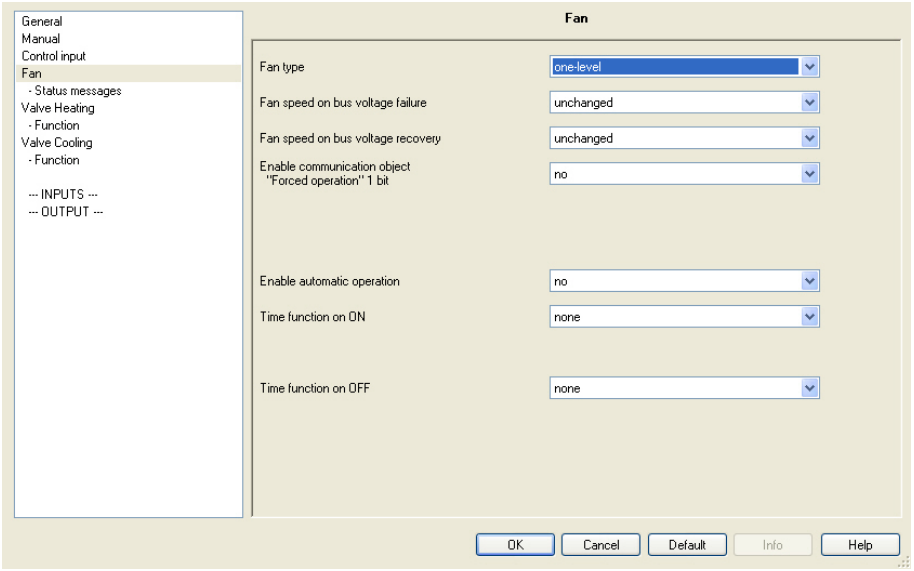
Now a two speed fan is controlled via fan speeds 1 and 2.

Fan speed 3 with all its parameters and options is now non-functional.

Note
Further parameters and their setting possibilities are described in parameter window **multi-level fan**.

3.2.6 Parameter window *One-level fan*

In this parameter window all settings for the *one-level fan* are undertaken.



Fan type

Option: multi-level
one-level

The fan type to be controlled is set with this parameter.

If a fan with two or three speeds is to be controlled, the option multi-level must be selected.

If a fan with one speed is to be controlled, the option one-level must be selected.

Fan speed on bus voltage failure

Option: unchanged
off
on

The behaviour of the fan with a bus voltage failure is defined here.

Fan speed on bus voltage recovery

Options: unchanged
off
on

The behaviour of the fan with a bus voltage recovery is defined here.

- *Unchanged:* The outputs and the fan speeds (speeds) of the fan remain unchanged.
- *Off:* The fan is switched off.
- *On:* The fan is switched on.

Caution

The FCA/S 1.1M is supplied ex-works with a default setting (factory default). This ensures the fan setting is switched off when the bus voltage is applied to the relay for the first time. Thus, damage to the device due to unintentional switch on during transport, e.g. due to vibration, is avoided.

It is advisable to apply a bus voltage before connecting the fan in order to achieve a defined switch state of the fan. This eliminates the possibility of the destruction of the fan due to an incorrect contact setting.

Enable communication object "Forced operation" 1 bit

Options: no
yes

- *yes:* A 1 bit *Forced operation* communication object is enabled. At the same time further parameters are made visible:

  Forced operation on object value

  Options: 1
  0

  - *1:* Forced operation is activated by a telegram with value 1.
  - *0:* Forced operation is activated by a telegram with value 0.

  Behaviour with forced operation

  Options: unchanged
  off
  on

  This parameter defines how the fan should respond with forced operation.

Enable automatic operation

Options: no
yes

- *yes:* Automatic mode is enabled; an additional *Automatic operation* parameter window appears.

Time function on ON

Options: none
switching delay
minimum time

The time function at fan ON is defined with this.

- *none:* No time function is executed.
- *switching delay:* The fan is switched on using this delay.
- *minimum time:* The fan remains ON for at least this time.

With option *switching delay* the following parameters are visible:

Time in s [1…65,535 x 0.1]

Options: 1…20…65,535

The fan is switched on using this delay.

With option *minimum time* the following parameters are visible:

Time in s [1…65,535]

Options: 1…20…65,535

The fan remains ON for at least this time.

Time function on OFF

Options: none
switching delay
minimum time

The time function at fan OFF is defined with this.

- *none:* No time function is executed.
- *switching delay:* The fan is switched off using this delay.
- *minimum time:* The fan remains OFF for at least this time.

With option *switching delay* the following parameters are visible:

Time in s [1…65,535 x 0.1]

Options: 1…20…65,535

The fan is switched off using this delay.

With option *minimum time* the following parameters are visible:

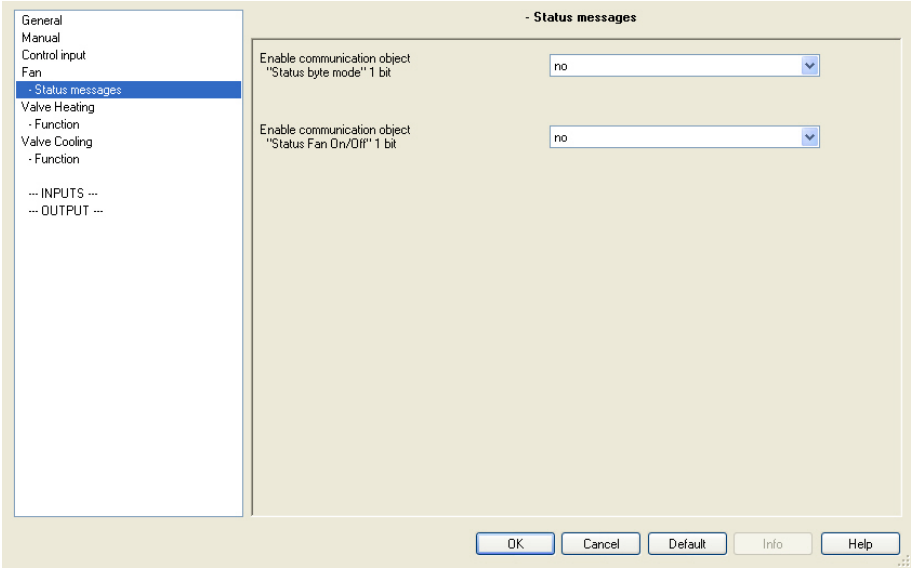
Time in s [1…65,535]

Options: 1…20…65,535

The fan remains OFF for at least this time.

3.2.6.1 Parameter window *Status messages*

In this parameter window the *Status messages* are defined.



Enable communication object "Status byte mode" 1 byte

Options: no
yes

- *yes:* The communication object *Status byte mode* is enabled.

From this status byte the states heating, cooling, automatic, forced operation and the four limitations are indicated directly via a 1 bit coding.

For further information see: Status byte code table

With option *yes* a further parameter is visible:

Send object value

Options: no, update only
after a change
after request
after a change or request

- *no, update only*: The status is updated but not sent.
- *after a change:* The status is sent after a change.
- *after request:* The status is sent after a request.
- *after a change or request:* The status is sent after a change or a request.

Enable communication object "Status Fan On/Off" 1 bit'

Options: no
yes

The communication object *Fan status* can be enabled with this parameter.

Some fans initially require an ON command before they are set to a fan speed from the OFF state. This ON command has effects a main switch which has to be switched on.
This demand can be implemented with any switch output which is controlled via the *Fan status* communication object. The corresponding switch object of the switch actuator should be connected with the *Fan status* communication object.

With option *yes* a further parameter is visible:

Send object value

Options: no, update only
after a change
after request
after a change or request

- *no, update only*: The status is updated but not sent.
- *after a change:* The status is sent after a change.
- *after request:* The status is sent after a request.
- *after a change or request:* The status is sent after a change or a request.

The following parameter only becomes visible if the option *yes* has been selected in the *Automatic operatio*n parameter in the *Fan* parameter window.

Enable communication object "Status automatic" 1 bit

Options: no
yes

The communication object *Status Automatic* is enabled with this parameter.

Telegram value 1 = automatic operation active
0 = automatic operation inactive

- *yes:* The following parameter also becomes visible:

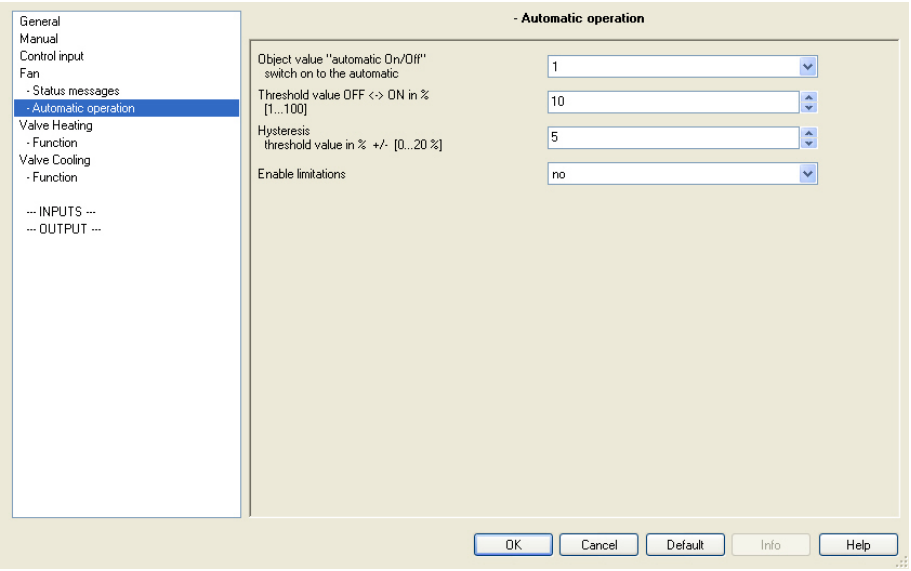
  Send object value

  Options: no, update only
  after a change
  after request
  after a change or request

  - *no, update only*: The status is updated but not sent.
  - *after a change:* The status is sent after a change.
  - *after request:* The status is sent after a request.
  - *after a change or request:* The status is sent after a change or a request.

3.2.6.2 Parameter window *Automatic operation*

This parameter window is visible if in parameter window *Fan* the option *yes* has been selected in the *Enable automatic operation* parameter.



In this parameter window the threshold values for switchover of the fan speed are defined. Furthermore, the limitations can also be enabled.

The corresponding valve control objects receive the value 1 if a fan speed is set. If a fan speed is not set the communication object will receive the value 0.

Object “Automatic On/Off”:
switch on to the automatic

Options: 1
0

This parameter defines how to react to a telegram.

- *1:* Automatic is activated by a telegram with value 1.
- *0:* Automatic is activated by a telegram with value 0.

Threshold value OFF <-> ON in %
[1…100]

Options: 1…10…100

Here the threshold value at which switch on occurs is defined. If the value in the control value object is greater than or equal to the parameterised threshold value, it is switched on. If the value is less, then it is switched off.

Hysteresis
threshold value in % +/- [0…20 %]

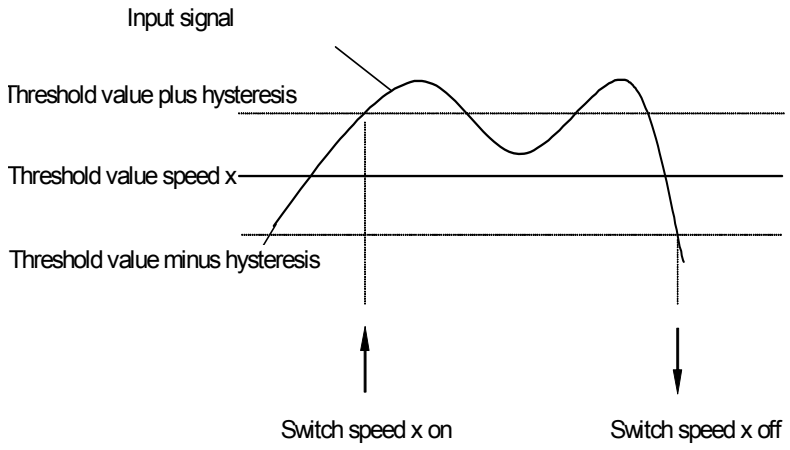
Options: 0…5…20

Here a hysteresis is set from which switchover to the next fan speed occurs. The hysteresis applies for all three threshold values.

The setting 0 causes immediate switching without hysteresis.

The entered percentage value is directly added to or subtracted from the percentage value of the *Fan speed x threshold value*. The result equals the new upper or lower threshold.

Example, a three speed fan, hysteresis with fan control



By using hysteresis, continuous switching between the fan speeds around the threshold value with deviating input signals can be avoided.

Enable limitations

Option: no
yes

- *yes:* Further parameters become visible.

At the same time 4 communication objects for limitation of the fan speed are enabled:

- *Limitation 1,* e.g. for standby operation
- *Limitation 2,* e.g. for night shutdown
- *Limitation 3,* e.g. for comfort operation
- *Limitation 4*, e.g. for frost/heat protection

Speed ranges (limitations) are defined for the fan with the speed limitation function which may not be exceeded or undershot.

Four limitations are available. This can be used for example for the control of various operating modes such as frost/heat protection, night shut down and standby. In normal cases the thermostat takes these operating modes into account in its control variable for the actuator.

Important
The parameterised starting behaviour which is a technical characteristic of the fan has a higher priority than a limitation or forced operation, i.e. if a limitation is activated in fan speed 2 and a start-up behaviour is parameterised via fan speed 3, the following behaviour will result: The fan is in the OFF state and receives a control signal for fan speed 1. Initially the fan operates at fan speed 3 (start-up speed) and then proceeds to fan speed 2 which is defined by the limitation. The actual required fan speed 1 will not be achieved due to the limitation.

The sequence of the displayed parameters corresponds with their priorities, i.e. the parameters with the highest priority has limitation 1 followed by limitation 2, 3 and 4.

Note
The fault operation, e.g. as with a malfunction of the thermostat has a lower priority than the fan limitation, i.e. by a limitation of the fan speed during a thermostat malfunction only the upper or the lower limit of the fan limitation can be set at best.

When automatic mode is exited, e.g. by a manual action, the limitations 1 to 4 remain.

The following points apply for limitations:

- The fan speed and valve position can be parameterised independently.
- The limitation need not necessarily apply to one fan speed only. It can also encompass another range of the fan speeds, i.e. only certain fan speeds can be set if the limitation is active. Hereby a limited control is also possible.
- The limitation is activated if a telegram with the value 1 is received on the limitation object. The limitation is deactivated if a telegram with the value 0 is received on the limitation object. A manual action ends automatic mode.
- If a limitation is activated, the Fan Coil Actuator switches to the parameterised fan speed regardless of the control value. If during the activation of the limitation another fan speed or a fan speed outside the range of the “limitation range” is set, the required fan speed or the limit fan speed of the range is set.
- After switch off of the limitations, the fan speed and the communication objects for valve control are recalculated and executed, i.e. during limitation the actuator operates normally in the background, the outputs are not changed and implementation only occurs after the end of limitation.

There are the same parameters for each of the individual four limitations used to limit the fan speeds. The priority is according to the listed sequence. The highest priority is assigned to limitation 1, e.g. Frost/Heat protection; the lowest priority is assigned to limitation 4, e.g. standby operation.

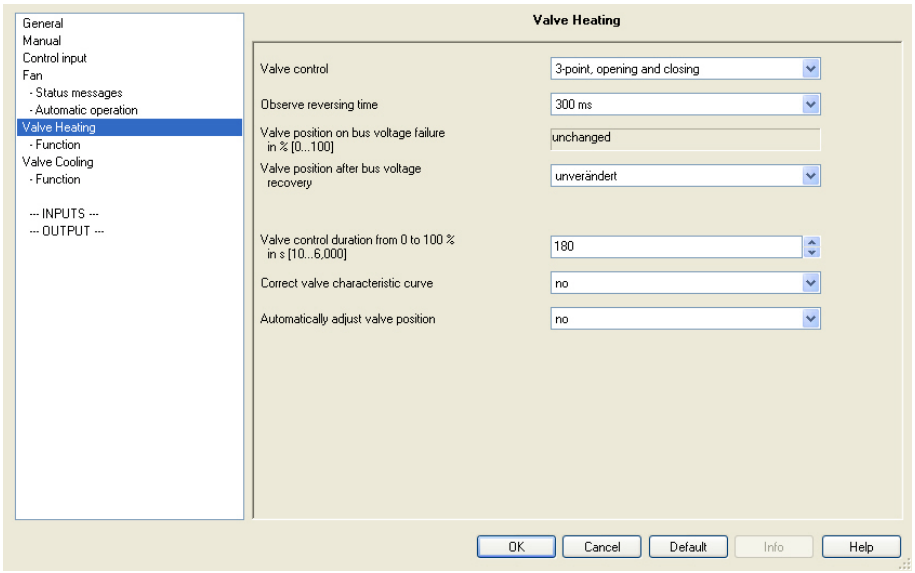
Speed with limitation 1
Speed with limitation 2
Speed with limitation 3
Speed with limitation 4

Options: inactive
unchanged
Off
On

With this parameter you set which fan speed is set with active limitation or not exceeded or undershot.

3.2.7 Parameter window *Valve Heating – 3-point, opening and closing*

If option *3-point, opening and closing* is selected with the parameter *Valve control*, the following parameters are visible in the *Valve Heating* parameter window.



Valve control

Options: 3-point, opening and closing
Continuous, PWM

With this parameter the properties of the connected valve are set (PWM = pulse width modulation).

Observe reversing time

Options: no
100/300/500/700/1,000 ms

A reversing time pause is set via this parameter.

The time should be taken from the technical data of the valve.

Valve position on bus voltage failure in % [0…100]

Note: unchanged

The valve remains unchanged at its position with a bus voltage failure.

Valve position after bus voltage recovery

Option: unchanged
select

Using this parameter the position of the valves after bus voltage recovery can be set.

- *select:* An additional parameter appears:

Valve position in % [0…100]

Option: 0…100

Using this parameter the position of the valves after bus voltage recovery can be set in percent.

Valve control duration from 0 to 100% in s [10…6,000]

Option: 10…180…6,000

With this parameter a time is set in seconds which the connected valve requires to move from position 0 % (valve closed) to position 100 % (valve fully open).

The time should be taken from the technical data of the valve.

Correct valve characteristic curve

Option: no
yes

Of the option *yes* is set in the parameter, the parameter window *Curve* in which the valve is set appears.

Automatically adjust valve position

Option: no
yes

- *yes:* Furthermore, the parameters *Number of valve controls up to adjustment [1…65,535]* appears.
- *no:* If this is set in the parameter nothing happens.

Note
A manual resolution of the adjustment is not possible!

Adjust with control value 0 %

Any action with control value 0 % is executed as an adjustment, i.e.:

- The valve is fully closed regardless of the curve.
- The closing position is exceeded by 5 % of the total time, max. one minute.
- This function cannot be interrupted!
- Thereafter the current valve position is moved to and the adjustment counter is set to zero.

The following applies with automatic adjustment

- The adjustment counter is incremented by 1 every time the valve stops.
- If the parameterised limit of the adjustment counter is exceeded in the closing direction, the adjustment starts.
- If higher priorities are activated at the time of automatic adjustment, the adjustment will be performed subsequently.
- The adjustment is interrupted by higher priority events.
- The valve is fully closed regardless of the curve.

- The closing position is exceeded by 5 % of the total time, max. one minute. This function cannot be interrupted! Thereafter the current valve position is moved to and the adjustment counter is set to zero

Note
A valve adjustment has occurred if a control of the drive has actually been undertaken. If priorities and curves prevent this, the adjustment counter will not change.

Referencing

A referencing or homing run can be understood as a complete closing of the valve.

Referencing is undertaken after:

- Every reset of the bus.
- A change of version.
- Every reset of an un-parameterised device.
- A download with modified adjustment time.

The following should be considered:

- Referencing cannot be interrupted.
- The closing position is exceeded by 5 % of the total time, max. one minute.
- After the referencing run, the current valve position is moved to and the adjustment counter is set to zero.

For further information see: Priority sequence

Number of valve controls up to adjustment [1...65,535]

Option: 1…100…65,535

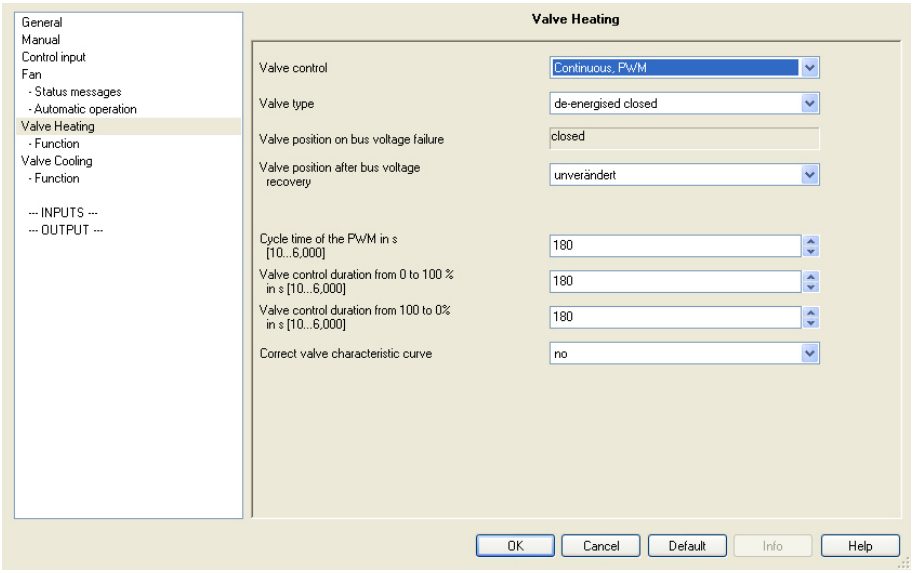
With this parameter the number of operations (valve controls) after which automatic adjustment is undertaken can be set.

Note
All actions greater than zero (motor does not move) are counted. The number should be taken from the technical data of the valve manufacturer.

3.2.7.1 Parameter window *Valve Heating – Continuous PWM*

If the option *Continuous, PWM* has been selected with parameter *Valve control*, the following parameters in the parameter window *Valve Heating* are visible.

PWM = pulse width modulation



Valve type

Options: de-energised closed
de-energised opened

Using this parameter the valve type for the connected valve is set.

How does a de-energised closed valve behave?

If no current flows in the control circuit the valve is closed. The valve is opened as soon as current flows in the control circuit.

How does a de-energised opened valve behave?

If no current flows in the control circuit the valve is opened. The valve is closed as soon as current flows in the control circuit.

- *de-energised closed:* The following parameter becomes visible:

Valve position on bus voltage failure

Option: closed

This option is intended as a marker. The valve remains closed at bus voltage failure.

- *de-energized opened:* The following parameter becomes visible:

  Valve position on bus voltage failure

  Option: opened

  This option is intended as a marker. The valve remains opened at bus voltage failure.

Valve position after bus voltage recovery

Option: unchanged
select

Using this parameter the position of the valves after bus voltage recovery can be set.

- *select:* An additional parameter appears:

  Valve position in % [0…100]

  Option: 0…100

  Using this parameter the position of the valves after bus voltage recovery can be set in percent.

Cycle time of the PWM in s [10…6,000]

Option: 10…180…6,000

This is used to set the cycle time of the PWM control.

Important
The minimum pulse duration is defined as 0.5 seconds. Thus, with very short cycle times (< 1 min.) there are very short switch on times (with small percentage values) or switch off times (with higher percentage values).

Valve control duration from 0 to 100% in s [10…6,000]

Option: 10…180…6,000

With this parameter a time is set in seconds which the connected valve requires to move from position 0 % (valve closed) to position 100 % (valve fully open).

Note
The time should be taken from the technical data of the valve and corresponds with the total runtime.

Valve control duration from 100 to 0 % in s [10…6,000]

Option: 10…180…6,000

With this parameter a time is set in seconds which the connected valve requires to move from position 100 % (valve open) to position 0 % (valve fully closed).

Note
The time should be taken from the technical data of the valve and corresponds with the total runtime.

Fast heat up/cool down

In addition to the adjustable time an additional time is determined in dependence on the change in control value. Thus, faster heat up or cool down of a room is achieved.
For determination of the additional time the difference between the current and the new control value is determined.
The additional time is dependent on how large the control value change should be from the current control value to the new control value.

Example
If the change in control value ascends, i.e. the current control value is at 10 % and the new control value is at 20 %, fast heat up is activated. If the change in control value descends, i.e. the current control value is at 60 % and the new control value is at 40 %, fast cool down is activated.

For further information see: Fast heat up/cool down

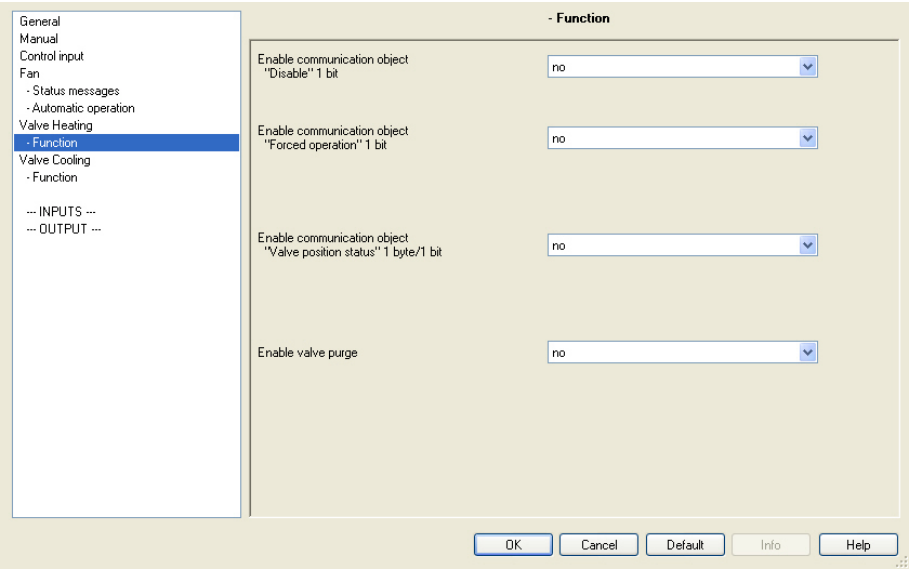
Correct valve characteristic curve

Option: no
yes

If the option *yes* is set in the parameter, the parameter window *Curve* in which the valve is set appears.

3.2.7.2 Parameter window *Function*

Various communication objects can be enabled in the parameter window *Function*.



Enable communication object "Disable" 1 bit

Options: no
yes

- *yes:* The 1 bit communication object *Block* is enabled and can then be used to block.

With option *yes* the following parameters are visible:

Disable on object value

Options: 1
0

Here you set the object value used to block the valve.

Enable communication object "Forced operation" 1 bit

Options: no
yes

- *yes:* The 1 bit communication object *Forced operation* is enabled and can then be used for forced operation.

With option *yes* the following parameters are visible:

Forced operation on object value

Options: 1
0

Here you set the object value used to forcibly operate the valve.

Valve position on forced operation in % [0…100]

Options: 0…<u>30</u>…100

Here the forced operation of the valve position in percent is set.

Enable communication object "Valve position status"

Options: <u>no</u>
1 bit
1 byte

Note
The status valve position is sent immediately after the control value is received.

- *1 bit* The following parameters are visible:

  Send object value

  Options: no, update only
  <u>after a change</u>
  after request
  after a change or request

  - *no, update only*: The status is updated but not sent.
  - *after a change:* The status is sent after a change.
  - *after request:* The status is sent after a request.
  - *after a change or request:* The status is sent after a change or a request.

  Object value with valve position >0

  Options: <u>1</u>
  0

- *1 byte:* The following parameter is visible:

  Send object value

  Options: no, update only
  <u>after a change</u>
  after request
  after a change or request

  - *no, update only*: The status is updated but not sent.
  - *after a change:* The status is sent after a change.
  - *after request:* The status is sent after a request.
  - *after a change or request:* The status is sent after a change or a request.

Enable valve purge

Options: no
yes

- *yes:* The 1 bit *Enable valve purge* communication object is enabled.

Note
If the valve purge is interrupted by a higher priority it will restart after the completion of the priority task, unless, i.e. the control value was 100 % or it was active for the duration of the purge time due to the higher priority. The valve position for purging is always the control value 100 %. A correspondingly matched curve is considered.

With option *yes* the following parameters are visible:

Enable communication object "Status valve purge" 1 bit

Options: no
yes

- *yes:* The 1 bit *Status valve purge* communication object is enabled.

The status of the valve purge and the following additional parameters are visible via this communication object.

Send object value

Options: no, update only
after a change
after request
after a change or request

- *no, update only*: The status is updated but not sent.
- *after a change:* The status is sent after a change.
- *after request:* The status is sent after a request.
- *after a change or request:* The status is sent after a change or a request.

Note
The status is sent immediately as soon as a new control value is received.

Duration of valve purge in min.
[1…255]

Options: 1…10…255

The time duration for the valve purge is set with this parameter. In this time the valve is fully opened. When the time has elapsed, the state before the purge is re-established.

Note
The opening time of the valve must be considered when entering the purge time.

Automatic valve purge

Options: no
yes

- *yes:* The following parameters are visible:

  Purge cycle in weeks
  [1…12]

  Options: 1…6…12

  The counter for automatic purging starts to run when the parameter is downloaded. The time is reset each time it is downloaded.

  The time is reset as soon as purging is completed. This can occur either through automatic purging or via the communication object *Trigger valve purge*.

Note
Purging can also be triggered via the bus with the communication object *Trigger valve purge*. After bus voltage recovery and download the purge cycle continues, the bus failure time – the time for which the bus actually failed – is not considered. The purging cycle will restart if *Purge cycle in weeks [1…12]* is changed after the download.

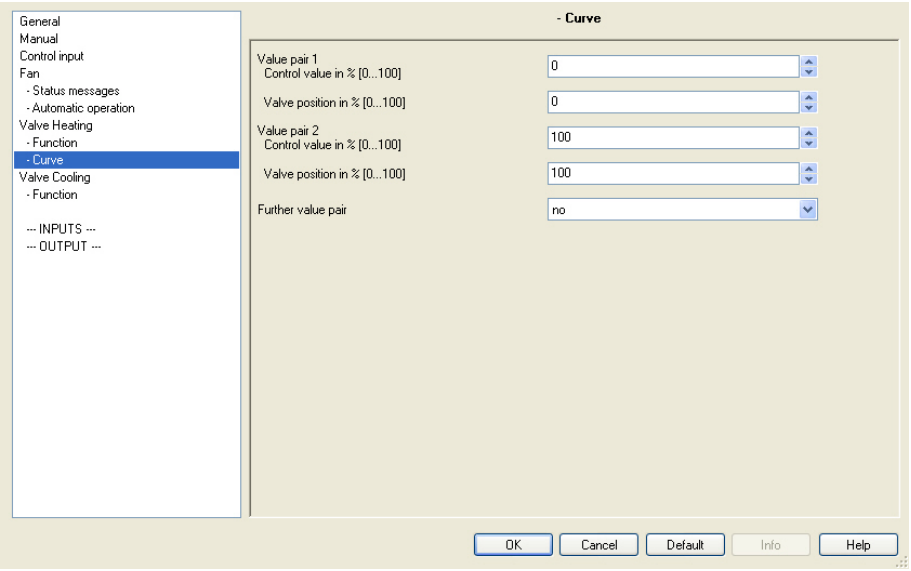
  Reset purge cycle
  from control value in % [1...99]

  Options:1…99

  Hereby the purge cycle from the set control value is reset.

3.2.7.3 Parameter window *Curve*

The parameter window *Curve* is visible if in parameter window *Valve Heating* the parameter *Correct valve characteristic curve* has been selected with *yes*.



The following must be considered with the curve entries:

- The value pairs can be entered in any sequence. They are sorted in ascending order of the control value in the device and intermediate values are interpolated.
- If value pairs have the same control value, the value pair with the largest value position applies. All other value pairs are ignored.
- The value pair with the smallest valve position applies for the correction of the smaller control values.
- If no value pair has been entered for the control value 0 %, the valve position of the first value pair applies for all control values from 0 to the first value pair.
- If no value pair has been entered for the control value 100 %, the valve position from the last value pair up to 100 % applies for the last value pair.

Caution

A parameterisation with the same control value leads to an undefined state and should be strictly avoided.
Otherwise it can lead to destruction of the HVAC system.

Value pair 1
Control value in % [0...100]
Options: 0…100

Valve position in % [0…100]
Options: 0…100

Value pair 2
Control value in % [0...100]
Options: 0…100

Valve position in % [0…100]
Options: 0…100

Value pair 1 forms the lower limit and value pair 2 forms the upper limit of the curve.

The possibility of activating other value pairs allows different curve characteristics to be realised.

For further information see: Valve curve

A total of four value pairs can be set.

Further value pair
Options: no
yes

- *yes:* A further value pair can be set.

Value pair 3
Control value in % [0...100]
Options: 0…50…100

Valve position in % [0…100]
Options: 0…50…100

Further value pair
Options: no
yes

- *yes:* A further value pair can be set.

Value pair 4
Control value in % [0...100]
Options: 0…50…100

Valve position in % [0…100]
Options: 0…50…100

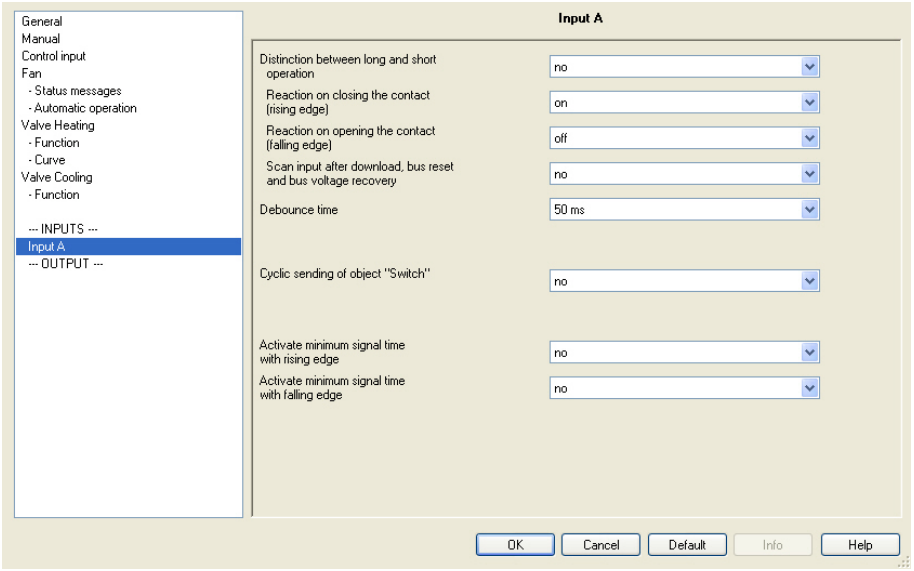
3.2.8 Parameter window *Valve Cooling*

Valve cooling does not differ from valve heating.

The descriptions of the parameter setting options and adjustable communication objects for the valve cooling are described under *parameter window Valve Heating*.

3.2.9 Parameter window *Input A*

In this parameter window all settings for the *Input A* are undertaken.



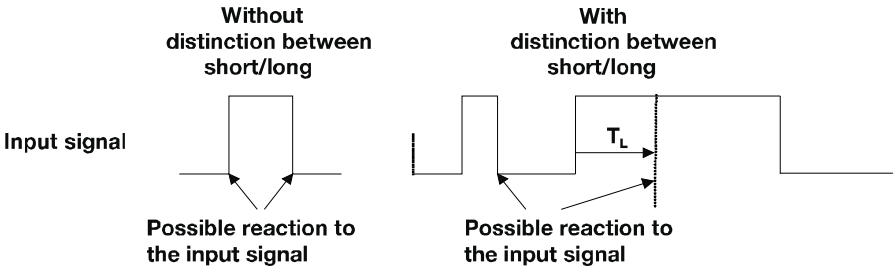
Difference between long and short operation

Options: no
yes

Using this parameter you decide if the input differentiates between short and long operation.

- *yes:* After opening/closing of the contact it must first of all be ascertained if a short or long operation has occurred here. Only thereafter will a possible reaction be triggered.

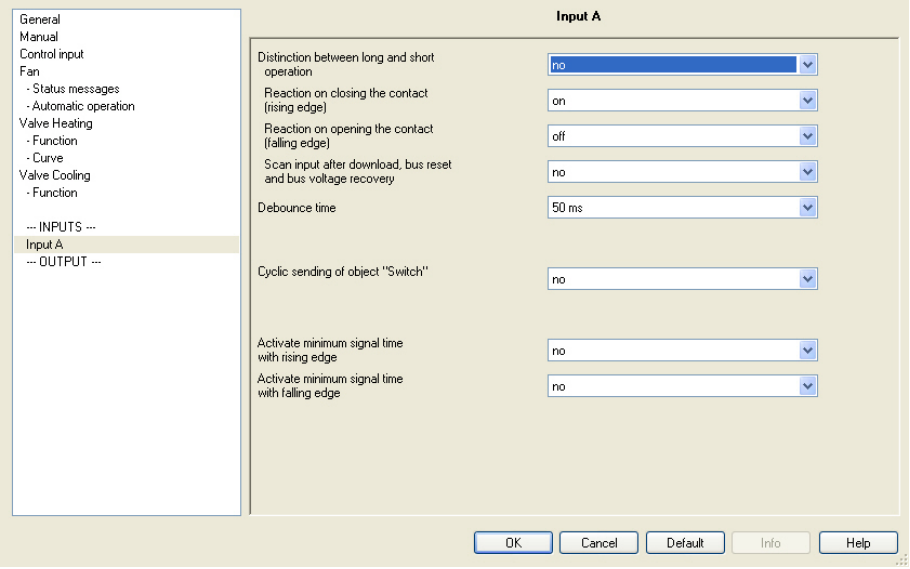
The following drawing shows the function in detail:



T_L is the time duration from where a long operation is detected.

3.2.9.1 Parameter *Difference between long and short operation – no*

If the option *no* is selected with the parameter *Difference between long and short operation*, the following parameters in the parameter window *Input A* are visible.



Reaction on closing the contact (rising edge)

Options: <u>on</u>
off
TOGGLE
No reaction
Terminate cyclic sending

Reaction on opening the contact (falling edge)

Options: on
<u>off</u>
TOGGLE
No reaction
Terminate cyclic sending

For each edge a definition is made to determine if the object value *on*, *off* or *TOGGLE* is switched or if there should be *no reaction*.

Scan input after download, bus reset and bus voltage recovery

Options: no
yes

- *yes:* The object value is scanned after a download, bus reset and bus voltage recovery.
- *no:* The object value is not scanned after a download, bus reset and bus voltage recovery.

With option *yes* the following additional parameters are visible in the parameter:

Inactive wait state after bus voltage recovery in s [0…30,000]

Options: 0…30,000

Here the waiting time after a bus voltage recovery is set. After the waiting time has elapsed the state on the input terminals is scanned. The input reacts as if the state on the input terminals has just been set/not set.

Note
The inactive waiting time does not add to the actual, adjustable send delay time. This can be set separately.

Debounce time

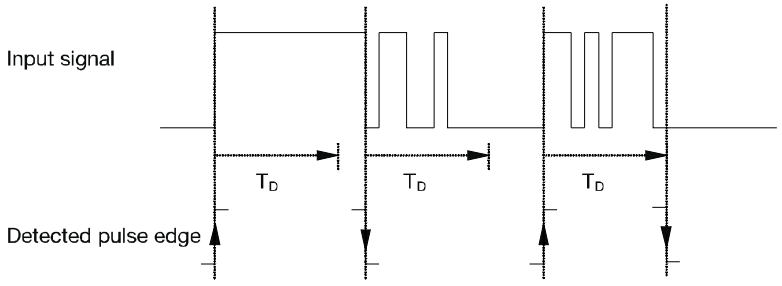
Options: 20/30/50/70/100/150 ms

Debouncing prevents unwanted multiple operations of the input, e.g. due to bouncing of the contact.

What is the debounce time?

If an edge is detected at an input, the input will react immediately to this edge (e.g. by sending a telegram). At the same time the duration of the debounce time T_D starts. The signal on the input is not evaluated within the debounce time duration.

Example: Debounce time of the input signal for a detected edge:



After detection of an edge on the input, further edges are ignored for the debounce time T_D.

Cyclic sending of object “Switch”

Options: no
yes

- *yes:* The following parameters are visible:

  with object value

  Options: 1
  0
  0 or 1

  - *1:* The object value is sent cyclically with 1.
  - *0:* The object value is sent cyclically with 0.
  - *0 or 1:* The communication object is sent cyclically.

    What is cyclic sending?

    Cyclic sending enables the communication object *Switch* to send automatically at a fixed interval.

    If cyclic sending is only carried out for a specific object value (ON or OFF), this condition refers to the value of the communication object. It is therefore possible in principle to start cyclic sending by sending a value to the communication object *Switch*. As this behaviour is generally unwanted, the flags *Write* and *Update* of the communication object are deleted in the preliminary setting so that they cannot be changed via the bus. If this functionality is required irrespectively, these flags should be set accordingly.

    When the *Switch* communication object and after bus recovery changes (after the send delay time has elapsed), the object value is sent immediately on the bus and the transmission cycle time restarts.

  Telegram repeated in s [1…65,535]

  Options: 1…60…65,535

  The send cycle time describes the time used between two cyclically sent telegrams.

Activate minimum signal time with rising edge

Options: no
yes

- *yes:* The following parameter is visible:

  in value x 0.1 s [0...65,535]

  Options: 0…65,535

Activate minimum signal time with falling edge

Options: no
yes

Yes: The following parameter is visible:

in value x 0.1 s [0...65,535]

Options: 0…65,535

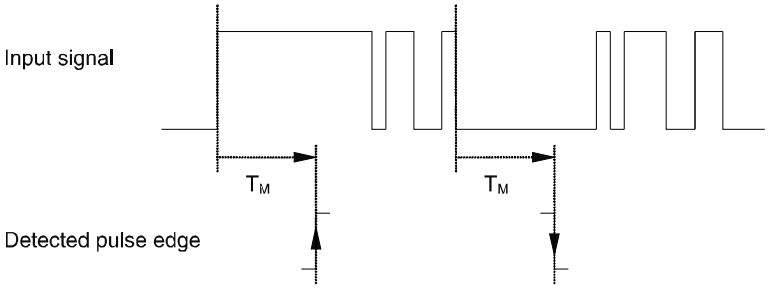
What is the minimum signal duration?

In contrast to the debounce time, a telegram is only sent after the minimum signal duration has elapsed.

The individual functions:

If an edge is detected on the input, the minimum signal duration will commence. No telegram is sent on the bus at this time. The signal on the input is observed within the minimum signal duration. If a further edge appears at the input during the minimum signal duration, it will be interpreted as a new operation and the minimum signal duration restarts.
If the input signal duration has not changed during the minimum signal duration, an edge is detected and a telegram is sent on the bus.

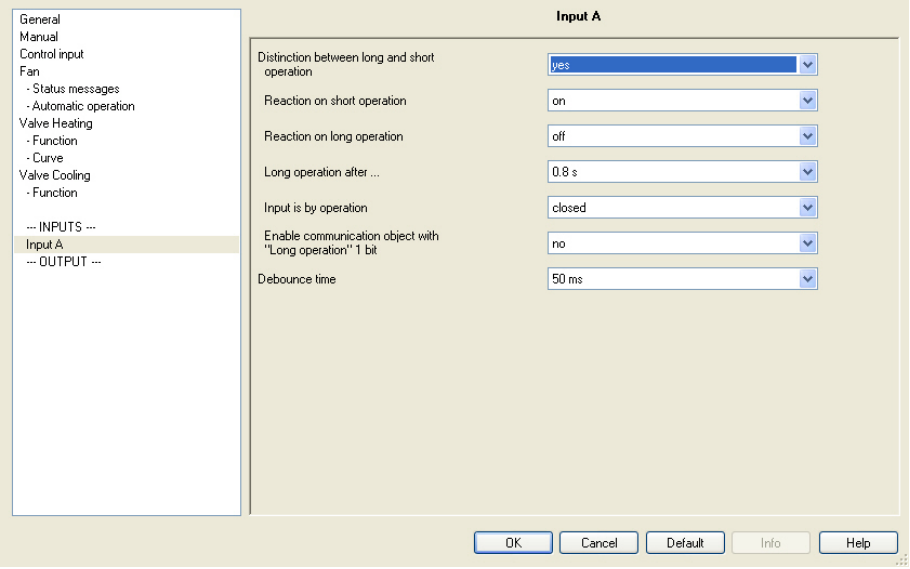
Example: minimum signal duration of the input signal for a detected edge:



As only two edges remain stable for the minimum signal duration T_M, only these are detected as valid.

3.2.9.2 Parameter *Difference between long and short operation – yes*

If the option *yes* is selected with the parameter *Difference between long and short operation*, the following parameters in the parameter window *Input A* are visible.



Reaction on short operation

Options: <u>on</u>
off
TOGGLE
no reaction

Reaction on long operation

Options: on
<u>off</u>
TOGGLE
no reaction

For each edge a definition is made to determine if the object value *on*, *off* or *TOGGLE* is switched or if there should be *no reaction*.

Long operation after…

Options: 0.3/0.4/0.5/0.6/<u>0.8 s</u>
1/1.2/1.5 s
2/3/4/5/6/7/8/9/10 s

Here the time period T_L after which an actuation is considered a “long” operation is defined.

Connected contact type

Options: <u>closed</u>
opened

- *closed:* The input is closed with actuation.
- *opened:* The input is opened with actuation.

Enable communication object with "Long operation" 1 bit

Options: no
yes

Debounce time

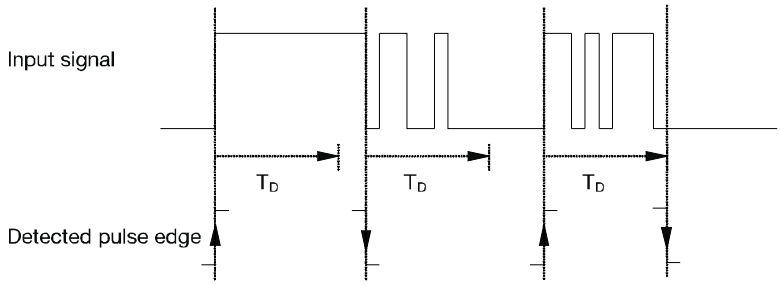
Options: 20/30/50/70/100/150 ms

Debouncing prevents unwanted multiple operations of the input, e.g. due to bouncing of the contact.

What is the debounce time?

If an edge is detected at an input, the input will react immediately to this edge (e.g. by sending a telegram). At the same time the duration of the debounce time T_D starts. The signal on the input is not evaluated within the debounce time duration.

Example: debounce time of the input signal for a detected edge:



After detection of an edge on the input, further edges are ignored for the debounce time T_D.

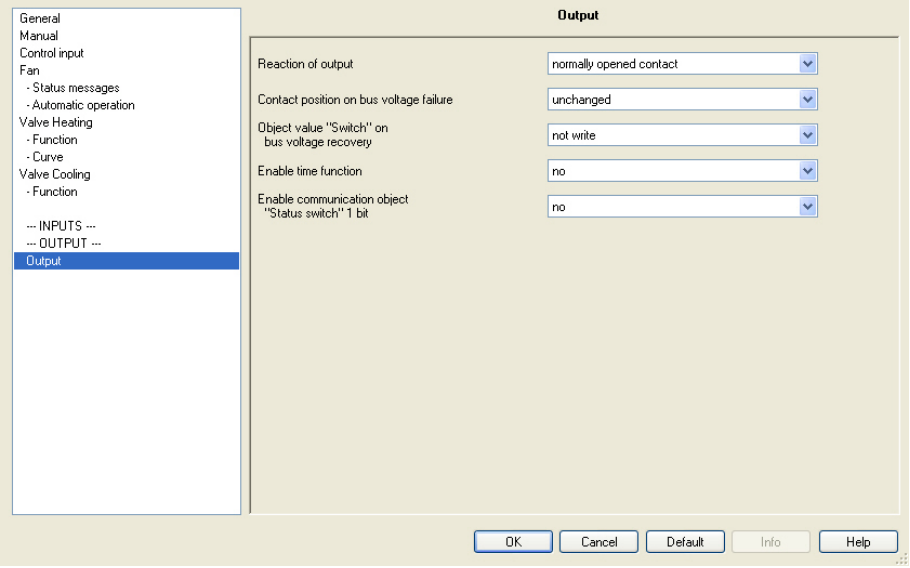
3.2.10 Parameter window *Input B*

Input B does not differentiate from input A.

The descriptions of the parameter setting options and adjustable communication objects for the *Input B* are described under *Input A*.

3.2.11 Parameter window *Output*

In this parameter window all settings for the *Output* are undertaken.



Reaction of output

Options: Normally closed contact
<u>normally opened contact</u>

It can be set in this parameter whether the output operates as a *Normally closed contact* or *Normally open contact*.

- *normally opened contact:* An ON command (1) closes the contact and an OFF command (0) opens the contact.
- *normally closed contact:* An ON command (1) opens the contact and an OFF command (0) closes the contact.

Contact position at bus voltage failure

Options: opened
closed
<u>unchanged</u>

The output can adopt a defined state on bus voltage failure via this parameter.

- *opened:* The contact is opened with bus voltage failure.
- *closed:* The contact is closed with bus voltage failure.
- *Unchanged:* No change of the contact position.

Note
The behaviour with bus voltage failure, recovery and download is to be monitored.

Object value "Switch" on bus voltage recovery

Options: not write
write with 0
write with 1

With this parameter the output can be influenced by the value of the *Switch* communication object on bus voltage recovery.

The *Switch* communication object can be written with either a 0 or 1 when the bus voltage recovers.
The contact position is redefined and set in dependence on the set device parameterisation.

- *not write:* The value 0 is written in communication object *Switch* and remains so until the communication object is changed via the bus. The contact position is only re-evaluated at this time.

Note
The behaviour with bus voltage failure, recovery and download is to be monitored. The Fan Coil Actuator draws the energy for switching the contact from the bus. After bus voltage is applied, sufficient energy is only available after about ten seconds in order to switch all contacts simultaneously. Depending on the set transmission and switching delay after recovery of bus voltage set in the parameter window *General*, the individual outputs will only assume the desired contact position after this time. If a shorter time is set, the FCA/S will only switch the first contact when sufficient energy is stored in the actuator, in order to ensure that enough energy is available to immediately bring all outputs safely to the required position with a renewed bus voltage failure

Enable time function

Options: no
yes

- *yes:* An additional parameter window appears in which the staircase lighting time is programmed.
  At the same time the communication object *block staircase lighting* is enabled.

Enable communication object "Status switch" 1 bit

Options: no
yes

- *yes:* Further parameters are visible:

  Send object value (Object "Status switch")

  Options: no, update only
  after a change
  after request
  after a change or request

  - *no, update only*: The status is updated but not sent.
  - *after a change:* The status is sent after a change.
  - *after request:* The status is sent after a request.
  - *after a change or request:* The status is sent after a change or a request.

  Object value of contact position (Object "Status Switch")

  Options: 1 = closed, 0 = open
  0 = closed, 1 = open

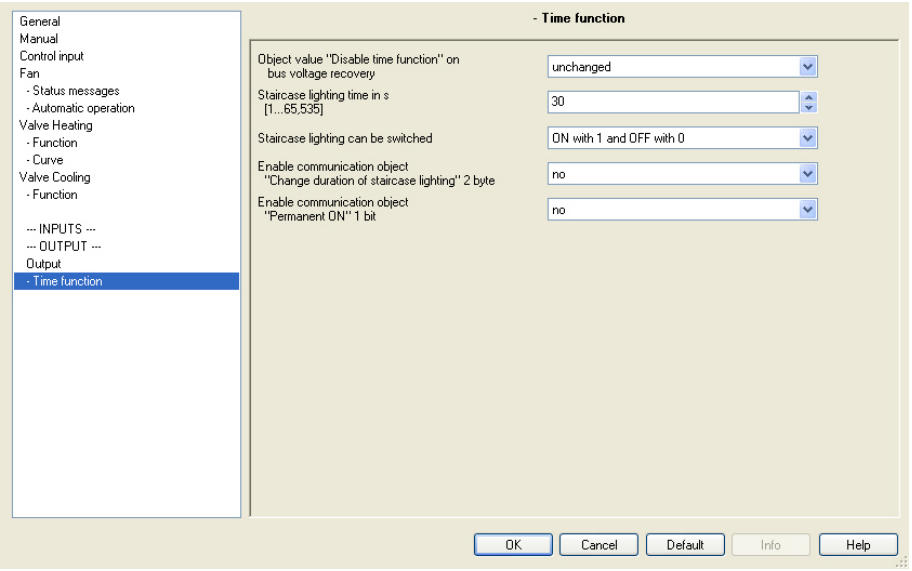
  With this parameter the communication object value of the switch status (*Status Switch*) is defined.

  - *1 = closed, 0 = open:* A closed contact is represented by communication object value 1 and an open contact is represented by the value 0.
  - *0 = closed, 1 = open:* A closed contact is represented by communication object value 0 and an open contact is represented by the value 1.

Note
The contact position, and thus the switch status can result from a series of priorities and links.

3.2.11.1 Parameter window *Time function*

This parameter window is only visible if in parameter window *Output* the option *yes* has been selected in the *Enable time function* parameter.



Object value "Disable time function" on bus voltage recovery

Options: unchanged
1, i.e. Disable time function
0, i.e. Enable time function

- *Unchanged:* The time function remains as it is.
- *1, i.e. Disable time function:* The time function light is blocked.

Note
They can only be enabled via the communication object *Disable staircase light*.

- *0, i.e. Enable time function:* The time function is enabled and active after a bus voltage failure.

Note
If the staircase light is disabled when a time function is operational, the light will stay ON until it is switched OFF manually.

Staircase lighting time in s
[0…65,535]

Options: 0…30…65,535

The staircase light defines how long the contact is closed, how long the light remains on after an ON command. The input is made in seconds.

Staircase lighting can be switched

Options: ON with 1 and OFF with 0
ON with 1 no action with 0
ON with 0 or 1, switch OFF not possible

With this parameter you can set which telegram value is used to switch on and prematurely switch off the staircase lighting.

- *ON with 0 or 1, switch OFF not possible:* The staircase lighting function is switched on independently of the value of the incoming telegram. Premature switch off is not possible.

Enable communication object
"Change duration of staircase lighting" 2 byte

Options: no
yes

- *yes:* A 2 byte *Change duration of staircase lighting* communication object is enabled. The staircase lighting time can be changed via the bus with this. The value defines the staircase lighting time in seconds. The staircase lightning time which has already commenced is completed. A change of the staircase lighting time is used the next time it is accessed.
- *no:* No modification of the staircase lighting time is possible via the bus.

Note
With bus voltage failure the changed staircase lighting time is saved. Only after a renewed download of the application program is the staircase lighting time overwritten.

How does the staircase lighting behave with bus voltage failure?

The behaviour at bus voltage failure is determined by the parameter *Reaction on bus voltage failure* in the parameter window *General*.

How does the staircase lighting behave with bus voltage recovery?

The behaviour at bus voltage recovery is defined by two conditions.

A By the communication object *Block staircase light*.
If the staircase light is blocked after bus voltage recovery, the staircase light can only be switched ON or OFF via the communication object *Switch*.

B By the parameterisation of the communication object *Switch*.
Whether the light is switched ON or OFF with bus voltage recovery depends on the programming of the *Switch* communication object.

Enable communication object "Permanent ON" 1 bit

Options: no
yes

- *yes:* The communication object *Permanent ON* is enabled.

If the communication object *Permanent ON* is assigned with the value 1, the output is switched on irrespective of the value of the communication object *Switch* and remains switched on until the communication object *Permanent ON* has the value 0. After ending the *Permanent ON* state the staircase will react as defined in the following parameters.

Example
This communication object can be used for example to allow the caretaker or maintenance and cleaning personnel to initiate a permanent ON.

The following parameter is visible if in parameter *Enable communication object "Permanent ON", 1 bit* the option *yes* has been selected.

Restart of staircase time after end of permanent ON

Options: no
yes

- *yes:* The lighting remains on and the staircase lighting time restarts.
- *no:* The lighting switches off if *Permanent ON* is ended.

The function of continuously ON is controlled via the *Permanent ON* communication object value. If the communication object receives a telegram with the value 1, the output is switched on irrespective of the value of the communication object *Switch* and remains switched on until the communication object *Permanent ON* has the value 0.

3.2.12 Commissioning without bus voltage

How is the device switched on and put into operation?

The device can be made operational by applying an auxiliary voltage from the power supply (NTI).

After applying the voltage the LED will indicate its current state.

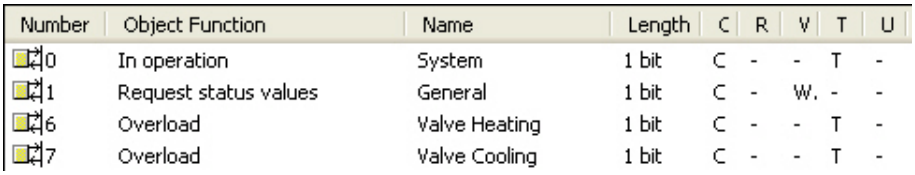
The manual switchover button (hand symbol) must be pressed to light up the respective LED. Thereafter, the device can be operated via the foil keypad. It is thus possible to try out all functions of the Fan Coil Actuator via the buttons, before the complete installation is put into operation, e.g. you can test if the fan speed switches up and down to suit the fan speed.

If the manual switchover button is pressed until the corresponding LED switches off, the device will switch off again.

The LED indicates its current state.

3.3 Communication objects

3.3.1 General



Number	Object Function	Name	Length	C	R	V	T	U
0	In operation	System	1 bit	C	-	-	T	-
1	Request status values	General	1 bit	C	-	W.	-	-
6	Overload	Valve Heating	1 bit	C	-	-	T	-
7	Overload	Valve Cooling	1 bit	C	-	-	T	-

No.	Function	Object name	Data type	Flags
0	**In operation**	**System**	**EIS 1, 1 bit DPT 1.002**	**C, R, T**

In order to regularly monitor the presence of the Fan Coil Actuator on the ABB i-bus®, a monitoring telegram can be sent cyclically on the bus.
The communication object is enabled if in the parameter *Send object "In operation"* in the parameter window *General* has been selected with *yes*.

As long as the communication object is activated it sends an In operation telegram.

Telegram value: 1= system operational

1	**Request status values**	**General**	**EIS 1, 1 bit DPT 1.017**	**C, R, T**

If a telegram with the value x (x = 0/1/0 or 1) is received in the communication object, all status objects are sent on the bus, as long as these have not been programmed with the option *after a change* or *request*.

The following function results for the option x = 1:

Telegram value: 1 = all status messages are sent, provided they are programmed with the option *after a change or request*.
0 = nothing happens.

2…5		**Manual operation**		

See description *Manual operation*.

6	**Overload**	**Valve Heating**	**EIS 1, 1 bit DPT 1.005**	**C, R, T**

The communication object sends a 1 with a fault, e.g. through a thermal overload on the output of the heating valve.
The communication object is always visible.

Telegram value: 1 = there is a fault on the output *Valve Heating*.
0 = fault acknowledgement.

7	**Overload**	**Valve Cooling**	**EIS 1, 1 bit DPT 1.005**	**C, R, T**

The communication object sends a 1 with a fault, e.g. through a thermal overload on the output of the cooling valve.
The communication object is always visible.

Telegram value: 1 = there is a fault on the output *Valve Cooling*.
0 = fault acknowledgement.

8...9				

Not assigned.

3.3.2 Manual

Number	Object Function	Name	Length	C	R	W	T	U
2	Enable/block manual operation	Manual operation	1 bit	C	-	W	-	-
3	LED On/Off	Manual operation	1 bit	C	-	W	-	-
4	Button On/Off	Manual operation	1 bit	C	R	-	T	-
5	Status of manual operation	Manual operation	1 bit	C	R	-	T	-

No.	Function	Object name	Data type	Flags
2	**Enable/block manual operation**	**Manual operation**	**EIS 1, 1 bit DPT 1.003**	**C, W**

The *Manual operation* of the Fan Coil Actuator is enabled or blocked via this communication object.

Note

If this communication object is assigned to a group address, the manual operation is blocked after each download, bus reset or bus voltage recovery. If the communication object is not assigned, manual operation is enabled.

If the value 0 is in this communication object, then the Fan Coil Actuator is switched over to *Manual operation* using the button on the device.

During this setting the states of the inputs and outputs are transferred to the input terminals. If the value 1 is in this communication object, then the Fan Coil Actuator is switched over to bus operation.

Telegram value: 0 = enable button
1 = block button

3	**LED On/Off**	**Manual operation**	**EIS 1, 1 bit DPT 1.001**	**C, W**

Using this communication object the LED On/Of is controlled on the foil keypad.

Telegram value: 0 = LED off
1 = LED on

4	**Button On/Off**	**Manual operation**	**EIS 1, 1 bit DPT 1.001**	**C, R, T**

Only by pressing the button will a telegram with the object value be sent.

Telegram value: 0 = button off
1 = button on

5	**Status of manual operation**	**Manual operation**	**EIS 1, 1 bit DPT 1.003**	**C, R, T**

On this communication object the Fan Coil Actuator sends the information regarding whether it is in *Manual operation* or *Operation via EIB/KNX*.
The status is sent after a change.

Telegram value: 0 = operation via EIB/KNX
1 = manual operation

3.3.3 Control input

3.3.3.1 Communication objects *HVAC System – 1 control value/2-pipe*

Number	Object Function	Name	Length	C	R	W	T	U
30	Control value, Heating/Cooling	Control input	1 Byte	C	-	W.	-	-
31	Control value, Cooling(extra!)	Control input	1 Byte	C	-	W.	-	-

No.	Function	Object name	Data type	Flags
30	**Control value, Heating/Cooling**	**Control input**	**EIS 6, 1 byte DPT 5.001**	**C, W**

The communication object is enabled if in parameter window *Control Input* the parameter *HVAC System* has been selected with the option *1 Control values/2-pipe*.

Using this communication object the control value heating or cooling is predefined as a 1 byte % value.

Telegram value: 0 % = off, no heating or cooling
100 % = on, largest control value, maximum heating or cooling

No.	Function	Object name	Data type	Flags
31	**Control value, Cooling (extra!)**	**Control input**	**EIS 6, 1 byte DPT 5.001**	**C, W**

Note

Independent of communication object 30, the cooling valve can be additionally controlled without monitoring via the communication object 31.

The communication object is enabled if in parameter window *Control Input* the parameter *HVAC System* has been selected with the option *1 Control values/2-pipe*.

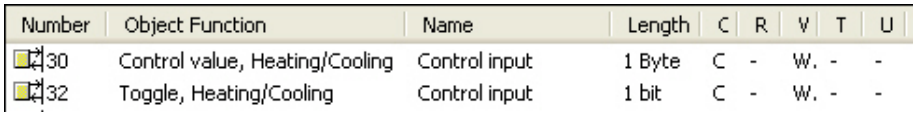
Using this communication object the control value cooling is predefined as a 1 byte % value.

Telegram value: 0 % = off, no cooling
100 % = on, largest control value, maximum cooling

No.	Function	Object name	Data type	Flags
32				

Not assigned.

3.3.3.2 Communication objects HVAC System – *1 Control values/4-pipe, with switching object*



Number	Object Function	Name	Length	C	R	V	T	U
30	Control value, Heating/Cooling	Control input	1 Byte	C	-	W.	-	-
32	Toggle, Heating/Cooling	Control input	1 bit	C	-	W.	-	-

No.	Function	Object name	Data type	Flags
30	**Control value, Heating/Cooling**	**Control input**	**EIS 6, 1 byte DPT 5.001**	**C, W**

The communication object is enabled if in parameter window *Control Input* the parameter *HVAC System* has been selected with the option *1 Control values/4-pipe, with switching object*.

Using this communication object the control value heating or cooling is predefined as a 1 byte % value.

Telegram value: 0% = off, no heating or cooling
100% = on, largest control value, maximum heating or cooling

No.	Function	Object name	Data type	Flags
31				

Not assigned.

No.	Function	Object name	Data type	Flags
32	**Toggle, Heating/Cooling**	**Control input**	**EIS 6, 1 bit DPT 1.100**	**C, W**

The communication object is enabled if in parameter window *Control Input* the parameter *HVAC System* has been selected with the option *1 Control values/4-pipe, with switching object*.

If the value 1 is set in the parameter:

Telegram value: 0 = cooling deactivated
1 = heating activated

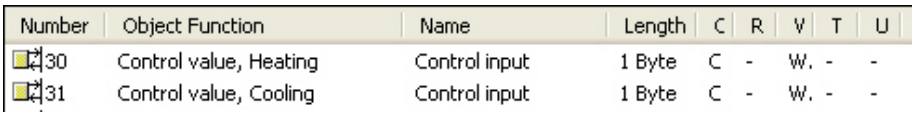
If the value 0 is set in the parameter:

Telegram value: 0 = heating activated
1 = cooling deactivated

Note

If communication object 32 *Toggle, Heating / Cooling – Control input* receives a value, the monitoring time is started.

3.3.3.3 Communication objects HVAC System – *2 control values/2-pipe*



Number	Object Function	Name	Length	C	R	V	T	U
30	Control value, Heating	Control input	1 Byte	C	-	W.	-	-
31	Control value, Cooling	Control input	1 Byte	C	-	W.	-	-

No.	Function	Object name	Data type	Flags
30	**Control value, Heating**	**Control input**	**EIS 6, 1 byte DPT 5.001**	**C, W**

The communication object is enabled if in parameter window *Control Input* the parameter *HVAC System* has been selected with the option *2 Control values/2-pipe*.

Using this communication object the control value heating is predefined as a 1 byte % value.

Telegram value: 0% = off, no heating
100% = on, largest control value, maximum heating

No.	Function	Object name	Data type	Flags
31	**Control value, Cooling**	**Control input**	**EIS 6, 1 byte DPT 5.001**	**C, W**

The communication object is enabled if in parameter window *Control Input* the parameter *HVAC System* has been selected with the option *2 Control values/2-pipe*.

Using this communication object the control value cooling is predefined as a 1 byte % value.

Telegram value: 0% = off, no cooling
100% = on, largest control value, maximum cooling

No.	Function	Object name	Data type	Flags
32				

Not assigned.

3.3.3.4 Communication objects HVAC System – *2 Control values/2-pipe, with switching object*

Number	Object Function	Name	Length	C	R	V	T	U
30	Control value, Heating	Control input	1 Byte	C	-	W	-	-
31	Control value, Cooling	Control input	1 Byte	C	-	W	-	-
32	Toggle, Heating/Cooling	Control input	1 bit	C	-	W	-	-

No.	Function	Object name	Data type	Flags
30	**Control value, Heating**	**Control input**	**EIS 6, 1 byte DPT 5.001**	**C, W**

The communication object is enabled if in parameter window *Control Input* the parameter *HVAC System* has been selected with the option *2 Control values/2-pipe, with switching object*.

Using this communication object the control value heating is predefined as a 1 byte % value.

Telegram value: 0% = off, no heating
100% = on, largest control value, maximum heating

No.	Function	Object name	Data type	Flags
31	**Control value, Cooling**	**Control input**	**EIS 6, 1 byte DPT 5.001**	**C, W**

The communication object is enabled if in parameter window *Control Input* the parameter *HVAC System* has been selected with the option *2 Control values/2-pipe, with switching object*.

Using this communication object the control value cooling is predefined as a 1 byte % value.

Telegram value: 0% = off, no cooling
100% = on, largest control value, maximum cooling

No.	Function	Object name	Data type	Flags
32	**Toggle, Heating / Cooling**	**Control input**	**EIS 6, 1 bit DPT 1.1000**	**C, W**

The communication object is enabled if in parameter window *Control Input* the parameter *HVAC System* has been selected with the option *2 Control values/2-pipe, with switching object*.

If the value 1 is set in the parameter:

Telegram value: 0 = cooling deactivated
1 = heating activated

If the value 0 is set in the parameter:

Telegram value: 0 = heating activated
1 = cooling deactivated

Note

If communication object 32 *Toggle, Heating / Cooling – Control input* receives a value, the monitoring time is started.

3.3.3.5 Communication objects HVAC System – *2 control values/4-pipe*

Number	Object Function	Name	Length	C	R	V	T	U
30	Control value, Heating	Control input	1 Byte	C	-	W.	-	-
31	Control value, Cooling	Control input	1 Byte	C	-	W.	-	-

No.	Function	Object name	Data type	Flags
30	**Control value, Heating**	**Control input**	**EIS 6, 1 byte DPT 5.001**	**C, W**

The communication object is enabled if in parameter window *Control Input* the parameter *HVAC System* has been selected with the option *2 Control values/2-pipe*.

Using this communication object the control value heating is predefined as a 1 byte % value.

Telegram value: 0 % = off, no heating
100 % = on, largest control value, maximum heating

No.	Function	Object name	Data type	Flags
31	**Control value, Cooling**	**Control input**	**EIS 6, 1 byte DPT 5.001**	**C, W**

The communication object is enabled if in parameter window *Control Input* the parameter *HVAC System* has been selected with the option *2 Control values/2-pipe*.

Using this communication object the control value cooling is predefined as a 1 byte % value.

Telegram value: 0 % = off, no cooling
100 % = on, largest control value, maximum cooling

No.	Function	Object name	Data type	Flags
32				

Not assigned.

3.3.3.6 Communication object *Fault control Value*

Number	Object Function	Name	Length	C	R	V	T	U
33	Fault control value	Control input	1 bit	C	R	-	T	-

No.	Function	Object name	Data type	Flags
33	**Fault control value**	**Control input**	**EIS 1, 1 bit DPT 1.005**	**C, R, T**

The communication object is enabled if in parameter window *Control Input* the parameter *Monitoring control value e.g. thermostat* is selected with the option *yes*.

This communication object indicates a malfunction of the control value, e.g. of a thermostat.

The Fan Coil control reports a fault and assumes the safety position with the communication object *Fault control value*. This safety position affects the fan speed and the valves.

Telegram value: 0 = no fault
1 = fault

Note

If the object value *Control Value, Heating*, *Control Value, Cooling* or *Control Value, Heating/Cooling* remains off for a parameterised time, a fault of the thermostat is assumed. If communication object 32 *Toggle, Heating / Cooling – Control input* receives a value, the monitoring time is started.

3.3.4 Multi-level fan

Number	Object Function	Name	Length	C	R	W	T	U
10	Fan speed switch	Fan	1 Byte	C	-	W.	-	-
11	Switch speed 1	Fan	1 bit	C	-	W.	-	-
12	Switch speed 2	Fan	1 bit	C	-	W.	-	-
13	Switch speed 3	Fan	1 bit	C	-	W.	-	-
14	Fan speed up/down	Fan	1 bit	C	-	W.	-	-
15	Status fan On/Off	Fan	1 bit	C	-	-	T	-
16	Status fan speed	Fan	1 Byte	C	R	-	T	-
17	Status fan speed 1	Fan	1 bit	C	R	-	T	-
18	Status fan speed 2	Fan	1 bit	C	R	-	T	-
19	Status fan speed 3	Fan	1 bit	C	R	-	T	-
21	Limitation 1	Fan	1 bit	C	-	W.	-	-
22	Limitation 2	Fan	1 bit	C	-	W.	-	-
23	Limitation 3	Fan	1 bit	C	-	W.	-	-
24	Limitation 4	Fan	1 bit	C	-	W.	-	-
25	Forced operation	Fan	1 bit	C	-	W.	-	-
26	Automatic On/Off	Fan	1 bit	C	-	W.	-	-
28	Status automatic	Fan	1 bit	C	R	-	T	-
29	Status byte mode	Fan	1 Byte	C	R	-	T	-

No.	Function	Object name	Data type	Flags
10	**Fan speed switch**	**Fan**	**EIS 6, 1 byte DPT 5.010**	**C, W**

The communication object is enabled if in parameter window *Fan* the parameter *Enable direct operation* and *Enable communication object "Fan speed switch" 1 byte* are selected with option *yes*.

With this communication object the fan can be switched on via a 1 byte communication object of a fan speed. If another fan speed is switched on at this point it will be switched off. A new fan speed is switched on taking the transition times, dwell times and start-up phase into consideration.

Limitations through forced operation or one of the four limitations 1…4 are retained. Automatic operation is disabled. A renewed activation of automatic mode occurs via the communication objects *Automatic On/Off.*

The following telegram values result:

1 byte value	Hexadecimal	Binary value bit 76543210	Fan speed
0	00	00000000	0 (OFF)
1	01	00000001	Fan speed 1
2	02	00000010	Fan speed 2
3	03	00000011	Fan speed 3
>3	>03	>00000011	Values greater than 3 are ignored

No.	Function	Object name	Data type	Flags
11	**Switch speed 1**	**Fan**	**EIS 1, 1 bit DPT 1.001**	**C, W**
The communication object is enabled if in parameter window *Fan* the parameter *Enable direct operation* and *Enable communication object "Switch speed x" 1 bit* are selected with option *yes*. Via the 1 bit communication object the Fan Coil Actuator can receive a control value for fan speed 1. Limitations through forced operation or one of the four limitations 1…4 are retained. Automatic operation is disabled. A renewed activation occurs via the communication objects *Automatic On/Off.* If several ON commands 1 are received by the various speed objects, the value last received for the fan control is decisive. This also applies for the OFF command 0. If the actuator for the switched off fan speed again receives an OFF command, it is carried out, i.e. another speed switched on at this time will be switched off even though the respective fan speed object does not act directly on the fan speed. The last command – in this case the OFF command of another fan speed – is always executed. Telegram value: 0 = fan off 1 = fan on in speed 1				
12	**Switch speed 2**			
See communication object 11				
13	**Switch speed 3**			
See communication object 11				

No.	Function	Object name	Data type	Flags
14	**Fan speed up/down**	**Fan**	**EIS 1, 1 bit DPT 1.007**	**C, W**

The communication object is enabled if in parameter window *Fan* the parameter *Enable direct operation* and *Enable communication object "Fan speed up/down" 1 bit* are selected with option *yes*.

With this communication object the fan can be switched one fan speed further up or down via a 1 bit telegram. Switching (up/down) is determined by the telegram value.

With multiple manual up or down switching operations the target speed will be increased or reduced by a speed step. This is possible until the maximum or minimum possible speed is achieved. The parameterised limitations are considered here. Further up or down commands are ignored and not executed. Each new switching command initiates a new calculation of the target speed.

Telegram value: 0 = switch fan speed down
1 = switch fan speed up

15	**Status Fan On/Off**	**Fan**	**EIS 1, 1 bit DPT 1.001**	**C, T**

The communication object is enabled if in parameter window *Status messages* the parameter *Enable communication object "Status Fan On/Off" 1 bit* has been selected with the option *yes*.

The communication object receives the communication object value 1 (ON), if at least one fan speed is not equal to zero (OFF). The value of the communication object is sent if not equal to zero. This communication object thus defines the status of the fan, whether it is switched on or off and the target speed is also indicated.

Telegram value: 0 = off
1 = on

Note

Some fans require an ON command before you set a fan speed. With the communication object *Status Fan On/Off,* the fan can for example, be switched on centrally with a switch actuator via the main switch.

No.	Function	Object name	Data type	Flags
16	**Status fan speed**	**Fan**	**noEIS, 1 byte DPT 5.010**	**C, R, T**

The communication object is enabled if in parameter window *Status messages* the parameter *Enable direct operation "Status fan speed" 1 byte* is selected with option *yes*.

You can parameterise whether only the communication object value is updated or if they are only sent on the bus after a change or on request. It is possible to parameterise if the actual or required speed are displayed with the status object.

With this communication object it is possible for example to display the fan speed on the display as a direct figure value.

The following telegram values apply for the 1 byte object:

Figure value	Hexadecimal	Binary value bit 76543210	Fan speed
0	00	00000000	0 (OFF)
1	01	00000001	Fan speed 1
2	02	00000010	Fan speed 2
3	03	00000011	Fan speed 3

17	**Status fan speed 1**	**Fan**	**EIS 1, 1 bit DPT 1.001**	**C, R, T**

The communication object is enabled if in parameter window *Status messages* the parameter *Enable direct operation "Status fan speed x" 1 bit* is selected with option *yes*.

It is possible to parameterise if a communication object value is only updated and not sent, sent on request or only sent when changed.

Furthermore, you can parameterise if the status should indicate a current fan speed or a required fan speed. With this communication object is possible to display the fan speed in a visualisation or to indicate it on a display.

Telegram value: 0 = fan speed off
1 = fan speed on

18	**Status fan speed 2**			

See communication object 17

19	**Status fan speed 3**			

See communication object 17

20				

Not assigned.

No.	Function	Object name	Data type	Flags
21	**Limitation 1**	**Fan**	**EIS 1, 1 bit DPT 1.003**	**C, W**
The communication object is enabled if in parameter window *Automatic operation* the parameter *Enable limitations* has been selected with the option *yes*. The limitation 1 is active if a telegram with the value 1 is received on the communication object *Limitation 1*. The *Limitation 1* is deactivated if a telegram with the value 0 is received on the communication object *Limitation 1.* When *Limitation 1* is activated, the fan can only assume the set fan speed or fan speed range in the parameter window *Fan B Limitation*. The valve position is independently programmable from the fan limitation. Telegram value: 0 = limitation x inactive 1 = limitation x active				
22	**Limitation 2**			
See communication object 21				
23	**Limitation 3**			
See communication object 21				
24	**Limitation 4**			
See communication object 21				

No.	Function	Object name	Data type	Flags
25	**Forced operation**	**Fan**	**EIS 1, 1 bit DPT 1.003**	**C, W**

The communication object is enabled if in parameter window *Direct operation* the parameter *Enable communication object "Forced operation" 1 bit* has been selected with the option *yes*.

If a forced operation is activated, the Fan Coil Actuator switches independently from the control value and its parameterised Limitation 1…4 to forced operation.

The fan speed and valve position(s) during forced operation can be parameterised individually from one another.

Telegram value: 0 = no forced operation
1 = forced operation

No.	Function	Object name	Data type	Flags
26	**Automatic On/Off**	**Fan**	**EIS 1, 1 bit DPT 1.003**	**C, W**

The communication object is enabled if in parameter window *Fan* the parameter window *automatic operation* has been enabled.

If automatic mode is enabled, it will be activated on this communication object with the value 1 after a download, bus reset or via a telegram.
Automatic mode is switched off, if a signal is received on a “manual communication object”. During forced operation the automatic mode remains active; however, it is only operated within the allowed limits.

If the value 1 is set in the parameter:

Telegram value: 0 = automatic operation off
1 = automatic operation on

If the value 0 is set in the parameter:

Telegram value: 0 = automatic operation on
1 = automatic operation off

No.	Function	Object name	Data type	Flags
27				

Not assigned.

No.	Function	Object name	Data type	Flags
28	**Status automatic**	**Fan**	**EIS 1, 1 bit DPT 1.003**	**C, R, W**

The communication object is enabled if in parameter window *Status messages* the parameter *Enable direct operation "Status automatic" 1 bit* is selected with option *yes*.

It is possible to parameterise if a communication object value is only updated and not sent, sent on request or only sent when changed.

The communication object indicates the status of the automatic mode.

Telegram value: 0 = inactive
1 = activated

No.	Function	Object name	Data type	Flags
29	**Status byte mode**	**Fan**	**noEIS, 1 byte noDPT**	**C, R, T**

The communication object is enabled if in parameter window *Status messages* the parameter *Enable direct operation "Status byte mode" 1 byte* is selected with option *yes*.

The operating state of the fan can be displayed or sent on the bus via this communication object. It is possible to parameterise if a communication object value is only updated and not sent, sent on request or only sent when changed.

Bit sequence: 76543210

Bit 7: Forced operation
Telegram value 0: inactive
1: active

Bit 6: Limitation 4
Telegram value 0: inactive
1: active

Bit 5: Limitation 3
Telegram value 0: inactive
1: active

Bit 4: Limitation 2
Telegram value 0: inactive
1: active

Bit 3: Limitation 1
Telegram value 0: inactive
1: active

Bit 2: Thermostat fault
Telegram value 0: inactive
1: active

Bit 1: Automatic
Telegram value 0: inactive
1: active

Bit 0: Heating/Cooling
Telegram value 0: Cooling
1: Heating

Note

Bit 0: If toggling between heating and cooling is undertaken automatically using control variables, the status is only switched in bit 0 if a value > 0 is received on the control value.

For further information see: ***Status byte code table***

3.3.5 One-level fan

Number	Object Function	Name	Length	C	R	W	T	U
11	Switch	Fan	1 bit	C	-	W	-	-
15	Status fan On/Off	Fan	1 bit	C	-	-	T	-
21	Limitation 1	Fan	1 bit	C	-	W	-	-
22	Limitation 2	Fan	1 bit	C	-	W	-	-
23	Limitation 3	Fan	1 bit	C	-	W	-	-
24	Limitation 4	Fan	1 bit	C	-	W	-	-
25	Forced operation	Fan	1 bit	C	-	W	-	-
26	Automatic On/Off	Fan	1 bit	C	-	W	-	-
28	Status automatic	Fan	1 bit	C	R	-	T	-
29	Status byte mode	Fan	1 Byte	C	R	-	T	-

No.	Function	Object name	Data type	Flags
10				
Not assigned.				
11	**Switch**	**Fan**	**EIS 1, 1 bit DPT 1.001**	**C, W**
The communication object is enabled if in parameter window *Fan* the parameter *Fan type* has been selected with the option *one-level*. With this 1 bit communication object the fan can be switched on or off. Limitations through forced operation or one of the four limitations 1…4 are retained. Automatic operation is disabled. A renewed activation occurs via the communication objects *Automatic On/Off.* If several ON commands 1 are received, the value last received for the fan control is decisive. This also applies for the OFF command 0. If the actuator for the switched off fan speed again receives an OFF command, it is carried out, i.e. another speed switched on at this time will be switched off even though the respective fan speed object does not act directly on the fan speed. The last command – in this case the OFF command of another fan speed – is always executed. Telegram value: 0 = fan off 1 = fan on				
12… 14				
Not assigned.				

No.	Function	Object name	Data type	Flags
15	**Status fan On/Off**	**Fan**	**EIS 1, 1 bit DPT 1.001**	**C, T**

The communication object is enabled if in parameter window *Status messages* the parameter *Enable direct operation "Status fan On/Off" 1 bit* is selected with option *yes*.

The communication object receives the communication object value 1 (ON), if the fan speed is not equal to zero (OFF). The value of the communication object is updated and sent when the fan speed is changed.

This communication object thus defines the status of the fan and whether it is switched on or switched off. It can also be used for control of a main switch for the fan.

Telegram value: 0 = off
1 = on

Note

Some fans require an ON command before you set a fan speed. With the communication object *Status Fan On/Off,* the fan can for example, be switched on centrally with a switch actuator via the main switch.

16… 20				

Not assigned.

21	**Limitation 1**	**Fan**	**EIS 1, 1 bit DPT 1.003**	**C, W**

The communication object is enabled if in parameter window *Automatic operation* the parameter *Enable limitations* has been selected with the option *yes*.

The limitation 1 is active if a telegram with the value 1 is received on the communication object *Limitation 1.* The *Limitation 1* is deactivated if a telegram with the value 0 is received on the communication object *Limitation 1*

When *Limitation 1* is activated, the fan can only assume the set fan speed or speed range in the parameter window *Fan limitation*. The valve position is independently programmable from the fan limitation.

Telegram value: 0 = limitation x inactive
1 = limitation x active

22	**Limitation 2**			

See communication object 21

23	**Limitation 3**			

See communication object 21

24	**Limitation 4**			

See communication object 21

No.	Function	Object name	Data type	Flags
25	**Forced operation**	**Fan**	**EIS 1, 1 bit DPT 1.003**	**C, W**

The communication object is enabled if in parameter window *Fan* the parameter *Enable communication object "Forced operation" 1 bit* has been selected with the option *yes*.

If a forced operation is activated, the Fan Coil Actuator switches independently from the control value and its parameterised Limitation 1…4 to forced operation.

The fan speed and valve position(s) during forced operation can be parameterised individually from one another.

Telegram value: 0 = no forced operation
1 = forced operation

No.	Function	Object name	Data type	Flags
26	**Automatic On/Off**	**Fan**	**EIS 1, 1 bit DPT 1.003**	**C, W**

The communication object is enabled if in parameter window *Fan* the parameter window automatic operation has been enabled.

If automatic mode is enabled, it will be activated on this communication object with the value 1 after a download, bus reset or via a telegram.
Automatic mode is switched off, if a signal is received on a “manual communication object”. During one of the four limitations or forced operation the automatic mode remains active, but however, it is only operated in the allowed limits.

If the value 1 is set in the parameter:

Telegram value: 0 = automatic operation off
1 = automatic operation on

If the value 0 is set in the parameter:

Telegram value: 0 = automatic operation on
1 = automatic operation off

No.	Function	Object name	Data type	Flags
27				

Not assigned.

No.	Function	Object name	Data type	Flags
28	**Status automatic**	**Fan**	**EIS 1, 1 bit DPT 1.003**	**C, R, W**

The communication object is enabled if in parameter window *Status messages* the parameter *Enable communication object "Status automatic" 1 bit* is selected with option *yes*.

It is possible to parameterise if a communication object value is only updated and not sent, sent on request or only sent when changed.

The communication object indicates the status of the automatic mode.

Telegram value: 0 = inactive
1 = activated

No.	Function	Object name	Data type	Flags
29	**Status byte mode**	**Fans**	**noEIS, 1 byte noDPT**	**C, R, T**

The communication object is enabled if in parameter window *Status messages* the parameter *Enable communication object "Status byte mode" 1 byte* is selected with option *yes*.

The operating state of the fan can be displayed or sent on the bus via this communication object. It is possible to parameterise if a communication object value is only updated and not sent, sent on request or only sent when changed.

Bit sequence: 76543210

Bit 7: Forced operation
Telegram value 0: inactive
1: active

Bit 6: Limitation 4
Telegram value 0: inactive
1: active

Bit 5: Limitation 3
Telegram value 0: inactive
1: active

Bit 4: Limitation 2
Telegram value 0: inactive
1: active

Bit 3: Limitation 1
Telegram value 0: inactive
1: active

Bit 2: Thermostat fault
Telegram value 0: inactive
1: active

Bit 1: Automatic
Telegram value 0: inactive
1: active

Bit 0: Heating/Cooling
Telegram value 0: Cooling
1: Heating

Note

Bit 0: If toggling between heating and cooling is undertaken automatically using control variables, the status is only switched in bit 0 if a value > 0 is received on the control value.

For further information see: Status byte code table

3.3.6 Valve heating

Number	Object Function	Name	Length	C	R	W	T	U
35	Block	Valve Heating	1 bit	C	-	W	-	-
36	Forced operation	Valve Heating	1 bit	C	-	W	-	-
37	Trigger valve purge	Valve Heating	1 bit	C	-	W	-	-
38	Status valve purge	Valve Heating	1 bit	C	R	-	T	-
39	Status valve position	Valve Heating	1 bit	C	R	-	T	-

No.	Function	Object name	Data type	Flags
35	**Block**	**Valve Heating**	**EIS 1, 1 bit DPT 1.003**	**C, W**

The valve is blocked via this communication object.
If the block is enabled, the highest priority is retained and the current control value is retained, i.e., the valve remains stationary. Movement to a target position which may not have yet been achieved will be performed to completion. If the block is removed the target position which has been set without the block is moved to.

Telegram value: 0 = valve not blocked
1 = valve blocked

No.	Function	Object name	Data type	Flags
36	**Forced operation**	**Valve Heating**	**EIS 1, 1 bit DPT 1.003**	**C, W**

This communication object sets the output in a defined state and blocks it. If the value 1 is received, forced operation is activated and the output triggers the programmed valve position. If the value 0 is received forced operation ends. The contact position is retained until the FCA/S receives a new setting signal.

Telegram value: 0 = end forced operation
1 = start forced operation

No.	Function	Object name	Data type	Flags
37	**Trigger valve purge**	**Valve Heating**	**EIS 1, 1 bit DPT 1.017**	**C, W**

The valve purge is triggered using this communication object.

Telegram value: 0 = end valve purge, valve will be closed
1 = start valve purge, valve will be opened

Note for value 0
A purge currently underway is interrupted. A purge not undertaken due to a higher priority will no longer be undertaken. The purge cycle with automatic purge will be restarted.

38	Status valve purge	Valve Heating	EIS 1, 1 bit DPT 1.003	C, R, T

The status of the valve purge is visible via this communication object.

Telegram value: 0 = valve purge not active
1 = valve purge active

Note

The status is displayed as soon as a purge has been activated. The status remains active even when the purge has been interrupted, e.g. by a priority.

39	Status valve position	Valve Heating	EIS 1, 1 bit DPT 1.001	C, R, T

This communication object is visible if in parameter window *Valve Heating* in the *Function* window, the parameter option *1 bit* has been selected for *Enable communication object "Valve position status"*.

The status of the valve position is visible via this communication object. Hereby, the target position where the valve should move to is always transferred. The display LED Heating ( ) indicates the same value as the status.

Telegram value: 0 = Valve position equal to zero/LED heating off
1 = Valve position not equal to zero/LED heating on

39	Status valve position	Valve heating	EIS 5, 1 byte DPT 5.001	C, R, T

This communication object is visible if in parameter window *Valve Heating* in the *Function* window, the parameter option *1 byte* has been selected for *Enable communication object "Valve position status"*.

The status of the valve position is visible via this communication object. Hereby, the target position where the valve should move to is always transferred. The display LED Heating ( ) indicates the same value as the status.

Telegram value: 0…255 = valve position is displayed directly as a figure value

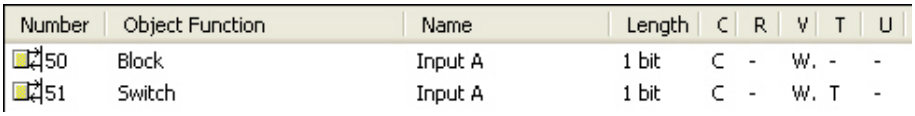
At 0 = LED heating off
At > 0 LED heating on

3.3.7 Valve cooling

Valve cooling does not differ from valve heating.

The descriptions of the parameter setting options and adjustable communication objects for the valve cooling are described under *parameter window Valve Heating* or communication objects *Valve Heating*.

3.3.8 Input A



Number	Object Function	Name	Length	C	R	W	T	U
50	Block	Input A	1 bit	C	-	W.	-	-
51	Switch	Input A	1 bit	C	-	W.	T	-

No.	Function	Object name	Data type	Flags
50	**Block**	**Input A**	**EIS 1, 1 bit DPT 1.003**	**C, W**

Using the communication object *Block* the input circuitry can be blocked or enabled. With the enable of a blocked input no telegram is sent on the bus. With activated communication object *Block* the inputs and *Manual operation* are blocked.

Note

When the input is blocked there is fundamentally no reaction, but:

– Waiting for a long button operation or a minimum signal duration is suspended.
– A signal change on the terminals or with manual operation is ignored.
– Communication objects continue to be updated and sent if necessary.

When enabling an input a change of the signal states (compared to before the block) leads to immediate processing, e.g.:

– The minimum actuation or detection of a long/short button push starts.
– Communication objects are sent if necessary.

Telegram value: 0 = enable input
1 = block input

51	**Switch**	**Input A**	**EIS 1, 1 bit DPT 1.001**	**C, W, T**

In accordance with the parameter setting, this communication object can be switched by actuation of the on, off or TOGGLE input. With toggle the previous value, e.g. 1, is toggled directly to the value 0. The communication object can be sent cyclically, e.g. for life sign monitoring of the sensor. It is important to note that the communication object can be written to externally. Thus, cyclic sending is interrupted or may not be possible.

Telegram value: 0 = off
1 = on

52	**Long switch operation**	**Input A**	**EIS 1, 1 bit DPT 1.001**	**C, T**

This communication object is only visible if the parameter *Distinction between long and short operation* and *Enable communication object with "Long operation" 1 bit* has been selected with yes.

This additional communication object is assigned to the long actuation. Communication object *Input A – Switch* no longer reacts to a long operation.

Telegram value: 0 = no
1 = yes

3.3.9 Input B

Input B does not differentiate from input A.

The descriptions of the parameter setting options and adjustable communication objects for the *Input B* are described under *Input A* and communication objects *Input A*.

3.3.10 Output

Number	Object Function	Name	Length	C	R	W	T	U
45	Switch	Output	1 bit	C	-	W	-	-
46	Permanent ON	Output	1 bit	C	-	W	-	-
47	Disable time function	Output	1 bit	C	-	W	-	-
48	Change duration of staircase l	Output	2 Byte	C	R	W	-	-
49	Status switch	Output	1 bit	C	R	-	T	-

No.	Function	Object name	Data type	Flags
45	**Switch**	**Output**	**EIS 1, 1 bit DPT 1.001**	**C, W**

This communication object is used for switching of the output On/Off.
The device receives a switch command via the switch object.

Normally opened contact:
Telegram value 1 = switch ON
0 = switch OFF

Normally closed contact:
Telegram value 1 = switch OFF
0 = switch ON

No.	Function	Object name	Data type	Flags
46	**Permanent ON**	**Output**	**EIS 1, 1 bit DPT 1.003**	**C, W**

With this communication object the output can be forcibly switched on.

If the communication object is assigned with the value 1, the output is switched on irrespective of the value of the object *Switch* and remains switched on until the communication object *Permanent ON* has the value 0. After ending the permanent ON state, the state of the communication object *Switch* is used.

Permanent ON only switched ON and “masks” the other functions. This means that the other functions (e.g. staircase) continue to run in the background but do not initiate a switching action. After the end of permanent ON the switching state which would result without the permanent ON function becomes active. The behaviour for the staircase lighting function after permanent ON is programmed in parameter window *Output time function*

This communication object can be used for example to allow the service or maintenance and cleaning personnel to initiate a permanent ON. The device receives a switch command via the switch object.

Permanent On becomes inactive after a download or bus voltage recovery.

Telegram value 1 = activates permanent ON mode
0 = deactivates permanent ON mode

No.	Function	Object name	Data type	Flags
47	**Disable time function**	**Output**	**EIS 1, 1 bit DPT 1.003**	**C, W**

This communication object is visible if in parameter window *Output General* the option *yes* has been selected in the *Enable staircase lighting time* parameter. The staircase lighting time can be enabled or disabled via this communication object.

After bus voltage recovery, the communication object value with the parameter *Value object "Disable Time Function" on bus voltage recovery* can be determined in the parameter window *Output staircase lighting time*.

With a disabled time function 1 the output can only be switched on or off, the staircase lighting time function will not be triggered.

Telegram value 1 = staircase light disabled
0 = staircase light enabled

The contact position at the time of disabling and enabling is retained and will only be changed with the next switch command to the communication object *Switch*.

No.	Function	Object name	Data type	Flags
48	**Change duration of staircase lighting**	**Output**	**EIS 10, 2 byte DPT 7.005**	**C, R, W**

This communication object is visible if in the parameter window *Time function* the parameter *Enable communication object "Change duration of staircase lighting" 2 byte* has been selected with *yes.*

The staircase lighting time can changed via the bus with this communication object. The time is defined in seconds.

The staircase lightning time which has already commenced is completed. A change of the staircase lighting time is used the next time it is accessed.

With bus voltage failure the changed staircase lighting time is retained. Only after a complete download of the application program, a version change, when the device has been discharged or with an ETS reset, is the staircase lighting time duration overwritten with the value set in the parameters.

No.	Function	Object name	Data type	Flags
49	**Status switch**	**Output**	**EIS 1, 1 bit DPT 1.001**	**C, R, T**

In the parameter window *Output* you can parameterise whether the communication object value *no, update only*, after a change or *after request* is sent on the bus. The communication object value directly indicates the current contact position of the switching relay.

The status value can be inverted.

Telegram value 1 = relay ON or OFF depending on the parameterisation
0 = relay OFF or ON depending on the parameterisation

4 Planning and application

In this section you will find a description of different types of fans, blowers and fan coil controllers. Here also are some tips and application examples are described for practical use of the device.

4.1 Heating, air-conditioning, ventilation control with Fan Coil units

The Fan Coil Actuator FCA/S 1.1M controls single-phase fans, blowers or fan coil units. Three speed single phase fans with step or changeover control are possible.

Special fan properties such as switchover pauses, dwell times and a start-up phase can be parameterised. Up to two input variables for heating and cooling signals, e.g. for a thermostat are available. As output variables, the Fan Coil Actuators generate up to two valve communication objects, which they can use to control the valves in a heating or cooling circuit.

The separate fan and valve parameterisation in the FCA/S provides a maximum in flexibility and very many combination possibilities for various applications in the heating, ventilation and air-conditioning (HVAC) field.

4.1.1 Terms

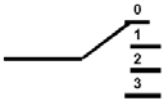
Fan Coil unit is a term used for a valve convector or blower convection unit.

The Fan Coil unit is connected to a central heating and cooling water supply and generates the desired temperature for the room. A room can be heated, cooled and ventilated using a Fan Coil unit.

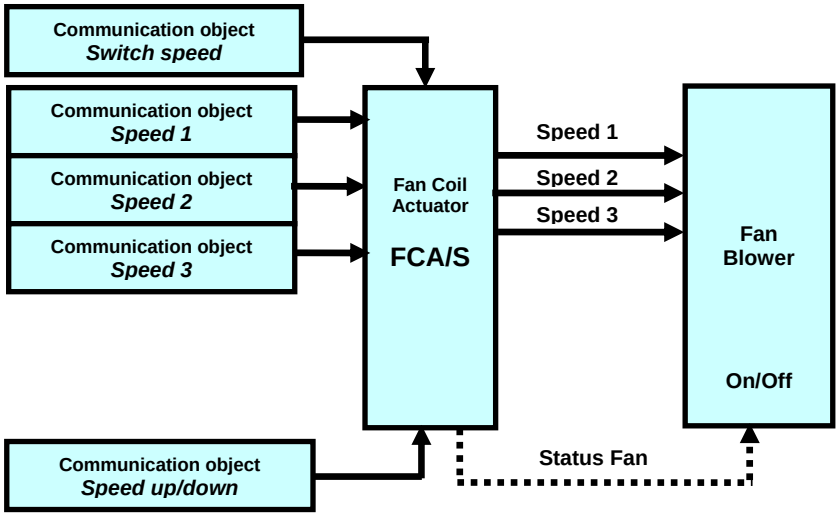
4.1.2 Fan operation

In fan operation a single phase fan, blower or convector can be controlled. In combination with a valve control 2-, 3- or 4-pipe system can be implemented.
The fans are controlled via a 3 speed controller. For this purpose 3 windings are tapped off of the fan motor. The speed results in dependence on the tap-off. It must be ensured that two contacts are not switched on simultaneously. For control purposes at least one 3 speed changeover switch with zero position is usually used. This switch is simulated with a group of outputs in the Fan Coil Actuator.

Three speed changeover switch



The control of the FCA/S is implemented in accordance with the following schematic principle:



Principle schematic of fan control via ABB i-bus®

With three *Fan speed x switch* (x = 1, 2, or 3) communication objects that are independent of each other, the fan speeds are controlled via the outputs of the Fan Coil Actuator.

Alternatively, the fan control can be implemented via a 1 byte communication object *Switch speed* or via the communication object *Speed up/down*.

Some ventilation controls require an additional central switch on mechanism (main switch) in addition to the speed switch. This can be implemented with a further output of the Fan Coil Actuator. The output must be linked to the *communication object Status Fan On/Off*. Hereby, the main switch is switched on if at least one fan speed is set. If the fan is OFF (*Status Fan On/Off* = 0), the main switch is also switched off.

4.1.2.1 Fan with changeover switch

Control of a fan is usually implemented with a changeover switch.

The following control table results for a three-speed fan, which simulates the FCA/S with a group of switch outputs:

	Output B 1	Output B 2	Output B 3
OFF	0	0	0
Fan speed 1	1	0	0
Fan speed 2	0	1	0
Fan speed 3	0	0	1

4.1.2.2 Fan with step switch

In some cases the fan is controlled via a step switch. The following control table results for a three-speed fan, which simulates the FCA/S with its outputs:

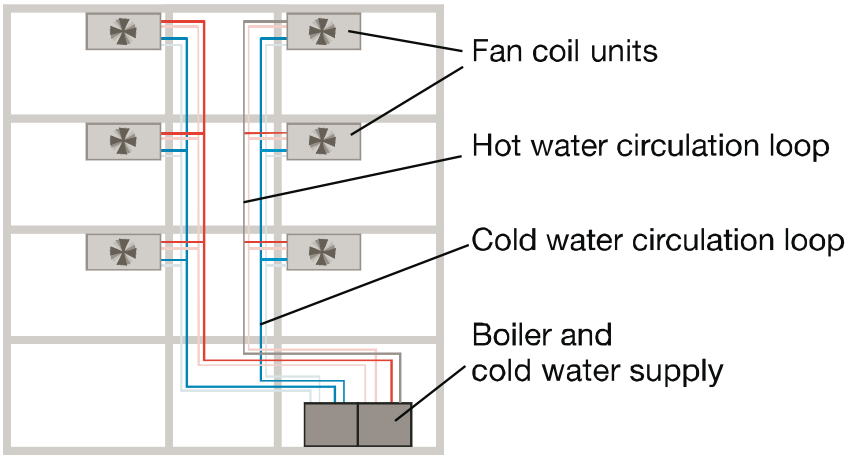
	Output B 1	Output 2	Output 3
OFF	0	0	0
Fan speed 1	1	0	0
Fan speed 2	1	1	0
Fan speed 3	1	1	1

The step switch cannot be switched on rapidly. If for example, fan speed 3 is to be switched on from the OFF state, fan speeds 1 and 2 must be controlled with the associated dwell times first.

4.1.3 Configuration of a HVAC system with Fan Coil units

A HVAC system with Fan Coil units (HVAC = heating, ventilation, air-conditioning) consists of a central heating and cooling water system. The Fan Coil units are installed in rooms and directly connected to the heating and cooling circuit, see illustration device technology.

4.1.4 Design of a Fan Coil unit

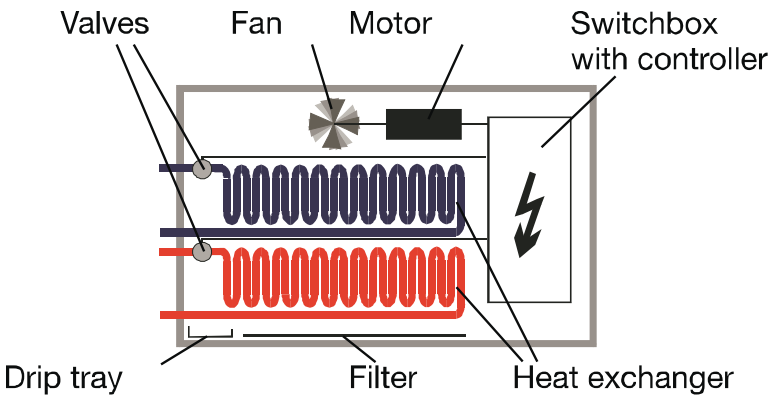


The Fan Coil unit consists of a fan or blower-convector and one or two heat exchangers, which emit heating or cooling power to the room.

If only one heat exchanger and one heating or cooling circuit are available, you have a 2 pipe system.

If two heat exchangers with two separate heating and cooling circuits are in use, you have a 4 pipe system. The Fan Coil Actuator directly controls the fan and provides one or two communication objects for control of the valves.

The heat exchanger and the fan are the most important components of a Fan Coil unit. Heating or cooling water flows in the heat exchanger depending on the desired room temperature. The flow of water through the heat exchanger is controlled via the valves.

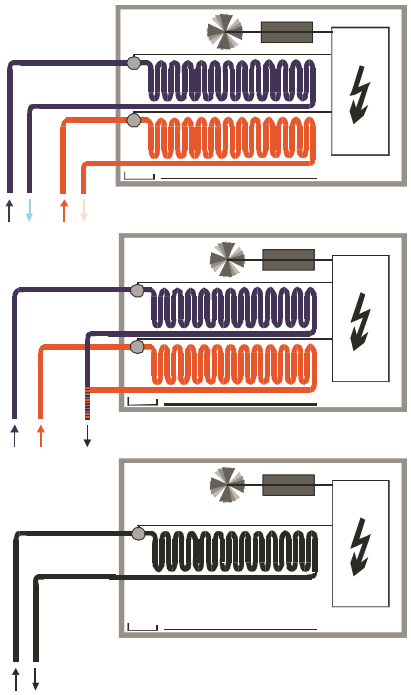


The fan blows air past the heat exchanger and into the room through a filter. The air is heated or cooled in the heat exchangers and thus generates the desired room temperature. The fan is driven by a motor. The motor and the valves are controlled by a Fan Coil Actuator.

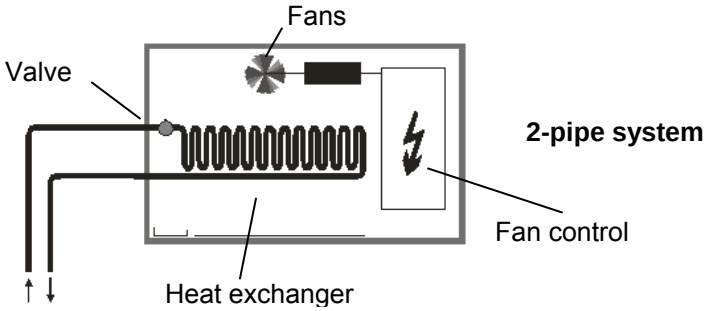
The water condensation which results during cooling collects in a condensation water trough.

4.1.5 Pipe systems

A Fan Coil unit can be configured as a 2-, 3- or 4-pipe system.



4.1.5.1 2-pipe system, configuration

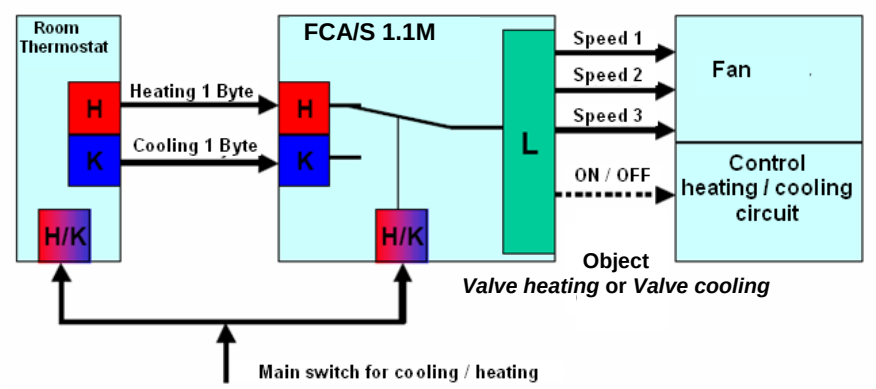


The 2-pipe system consists of just a single water circuit which is heated or cooled alternately to suit the season. In a 2-pipe Fan Coil unit there is only one heat exchanger with a valve.

Note
In some HVAC systems cooling is undertaken exclusively with a 2-pipe Fan Coil unit. The heating function is undertaken by a conventional heater or an electrical heater.

4.1.5.2 2-pipe system, heating and cooling

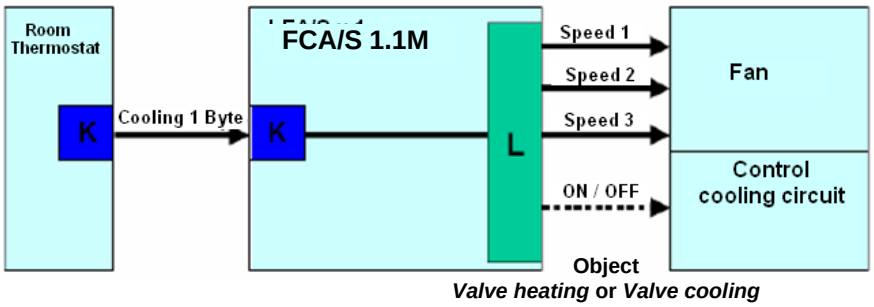
In this system only one heat exchanger is available for heating and cooling. Depending on the weather, warm or cold water is supplied centrally to the pipe system (2 pipes). The Fan Coil Actuator or the thermostat is informed if warm or cold water is currently flowing through the system. Depending on this setting both control values act on just a single valve. The thermostat decides which control value (heating/cooling) is actively sent. The FCA/S controls the fan speed and only one valve.



2-pipe system heating and cooling (3 speed fan)

4.1.5.3 2-pipe system, heating or cooling

In this system one heat exchanger is available for heating or cooling. The control value for heating or cooling is provided by a thermostat. Only warm or only cold water is supplied centrally to the pipe system (2 pipes). Depending on this setting one control value acts on one valve. The thermostat sends the control value (cooling) and the FCA/S controls the fan speed and the valve.



Principle schematic of 2-pipe heating or cooling system (3 speed fan)

Note
Both 2-pipe systems can be established using a 3 speed fan or blower. Depending on the control value (1 byte or 1 bit) which is sent from a thermostat, the Fan Coil Actuator determines the corresponding fan speeds via programmable threshold values.

For a continuous control value (1 byte; 0...100 %) the threshold values for the fan speeds can be defined for example as follows:

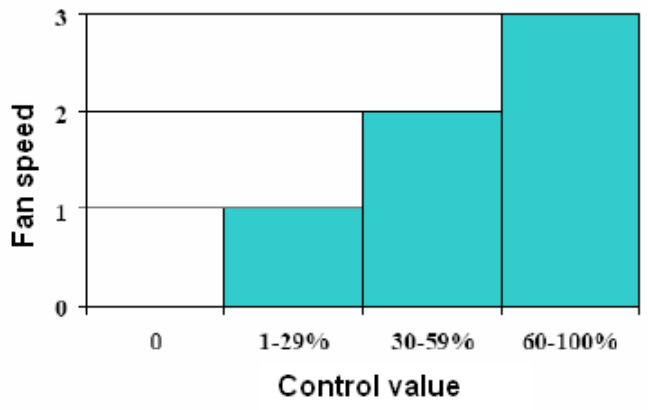
Example

Three-speed fan:

Fan speed 1: 1… 29 %
Fan speed 2: 30… 59%
Fan speed 3: 60…100%

Switch thresholds in the FCA/S:

Off -> fan speed 1 = 1%
Fan speed 1 -> 2 = 30%
Fan speed 2 -> 3 = 60%

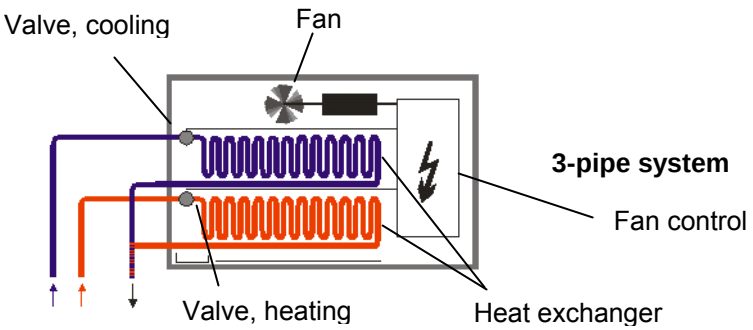


Function diagram for 2-pipe fan coil system (3-speed fan)

4.1.5.4 3-pipe system, configuration

The 3-pipe system has a similar design to the 4-pipe system. There is a separate inlet for heating and cooling water as well as two separate heat exchangers with one valve each. In contrast to a 4 pipe system, the 3-pipe system has a common return for both heating and cooling water.

The Fan Coil Actuator directly controls the fan and provides two communication objects for control of the valves.

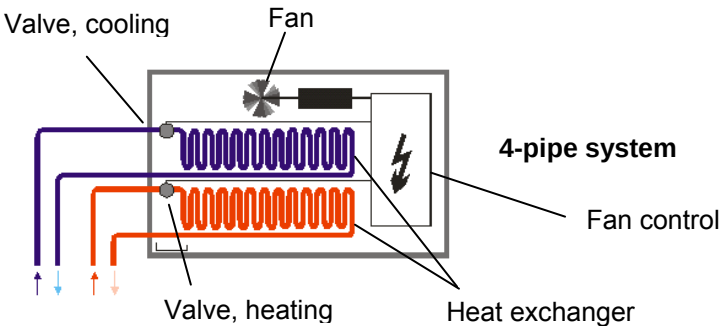


4.1.5.5 4-pipe system, configuration

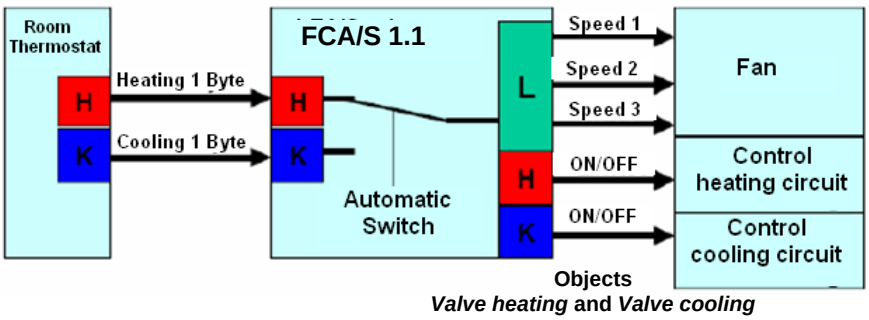
In a 4-pipe system two separate heat exchangers (for heating and cooling) are available. Warm and cold water is provided centrally to two separate pipe systems (of 2 pipes each).

The thermostat onsite decides if heating or cooling is applied.
The thermostat sends either a separate heating or cooling signal, or a common heating/cooling signal. If only one signal is sent for heating and cooling, an additional signal decides if the system is in heating or cooling mode and the valve which has to be controlled.

The Fan Coil Actuator directly controls the fan and provides two communication objects for control of the valves.



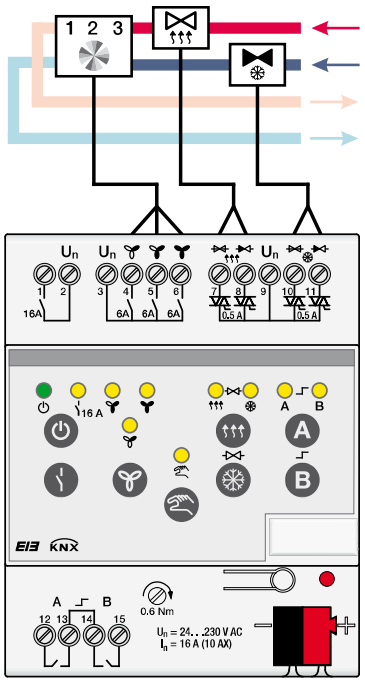
4-pipe system, configuration



4-pipe system heating and cooling (3 speed fan)

4.2 System configuration with a Fan Coil Actuator

In this function the Fan Coil Actuator is used for control of the heating and cooling valve as well as for switching the fan outputs. Temperature detection is undertaken by a thermostat.



Even the offset of the set point value as well as changeover of the operating modes is implemented by the thermostat. The sensors can be connected directly to the Fan Coil Actuator in order to consider the monitoring of the condensed water and the window contact.

To correctly implement this function the thermostat must send the actual temperature as well as the corresponding operating mode to the Fan Coil Actuator via the bus.

4.2.1 Automatic operation

With automatic fan control a fan drive is connected directly to the Fan Coil Actuator and switched via three floating contacts. A single speed, two speed or three speed fan can be connected.

The fan speed is set automatically in dependence on the control value. For example, the following control value ranges can be programmed for the corresponding fan speeds:

Control value	Fan speed
0… 9 %	0 (fan off)
10… 39 %	1
40… 69 %	2
70…100 %	3

The Fan Coil Actuator FCA/S 1.1M is purely an actuator which does not have a controller for a room temperature controller (thermostat). Control of the room temperature is implemented using a room temperature controller which generally detects the room temperature. The FCA/S 1.1M primarily controls a fan and valves. In addition to a manual control via the communication objects *Fan speed x*, *Fan speed switch* or *Fan speed up/down*, the Fan Coil Actuator can also operates in automatic mode together with a thermostat. Communication objects *Control value, Heating*, *Control value, Cooling,* or when operating with just a single input variable, the object *Control Value, Heating/Cooling* are available.

The automatic mode is enabled in the parameter window *Fan* with the parameter *Enable automatic operation*. Depending on the HVAC system, this is set in the parameter window *Control input* and the control value objects are enabled.

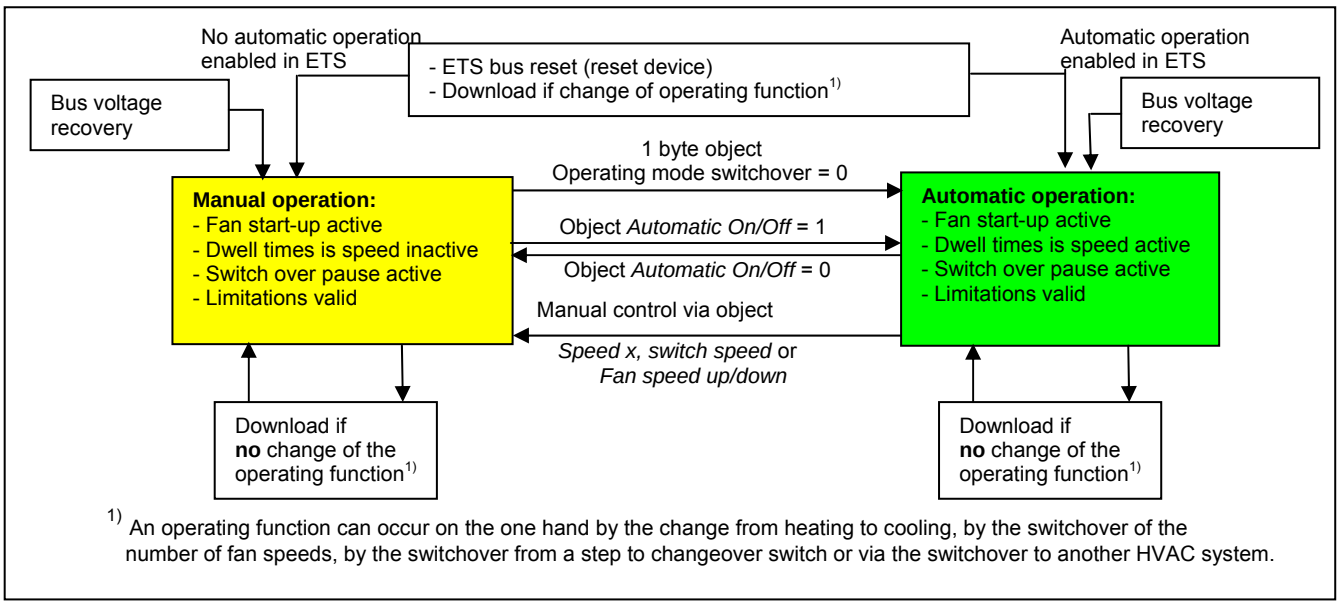
An automatic operation parameterised in the ETS only becomes active after the first download. With a subsequent download the automatic operating state (active, inactive) is retained as it was before the download. There is however an exception when system properties such as HVAC systems, fan control (changeover, step control) or the fan speed count has been changed (1/2/3). In these cases the automatic mode is activated if the automatic mode has been enabled in the ETS.

Automatic mode is switched off either by a manual setting command via the communication objects *Speed x*, *Fan speed switch* or *Fan speed up/down,* or if a telegram with the value 0 is received via the communication object *Automatic On/Off*.

The automatic operation can be reactivated by the communication object *Automatic On/Off* or the 1 byte communication object *Change Limitation*.

An activation of one of the four limitations or the forced operation does not end automatic operation. By using a range limit (several fan speeds are permissible), a limited automatic control with several fan speeds is possible.

The following functional diagram shows the relationship between automatic and manual operation of the Fan Coil Actuator.



1) An operating function can occur on the one hand by the change from heating to cooling, by the switchover of the number of fan speeds, by the switchover from a step to changeover switch or via the switchover to another HVAC system.

4.2.2 Direct operation

With direct fan control via the ABB i-bus®, a fan drive is connected directly to the Fan Coil Actuator and switched via three floating contacts. A single speed, two speed or three speed fan can be connected.

The Fan Coil Actuator sets the fan speed in accordance with the value received via the ABB i-bus®. The value is received as a 1 byte value.
The conversion of the received 1 byte value to the fan speed occurs as with the automatic fan control via the parameterised threshold values.

1 byte value	Fan speed
0… 9 %	0 (fan off)
10… 39 %	1
40… 69 %	2
70…100 %	3

4.2.3 Switchover between automatic and direct operation

In the Fan Coil Actuator you can switch between automatic operation and direct operation. The changeover to manual fan control is implemented via a 1 bit value. The fan speed is switched in accordance with the received 1 byte value.

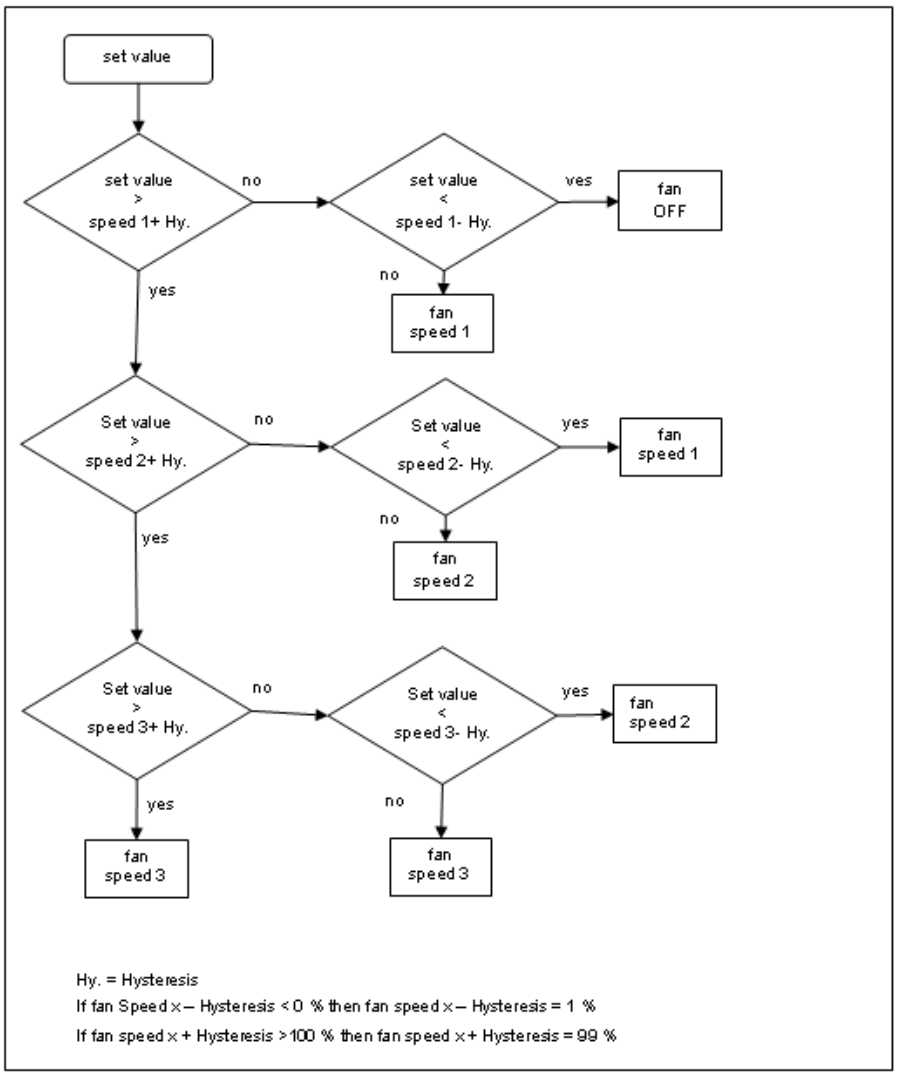
The fan control is changed back to automatic operation if a 1 is received in the respective communication object.

The current status of automatic operation is fed back via a 1 bit value.

4.2.4 Logic of the step switch

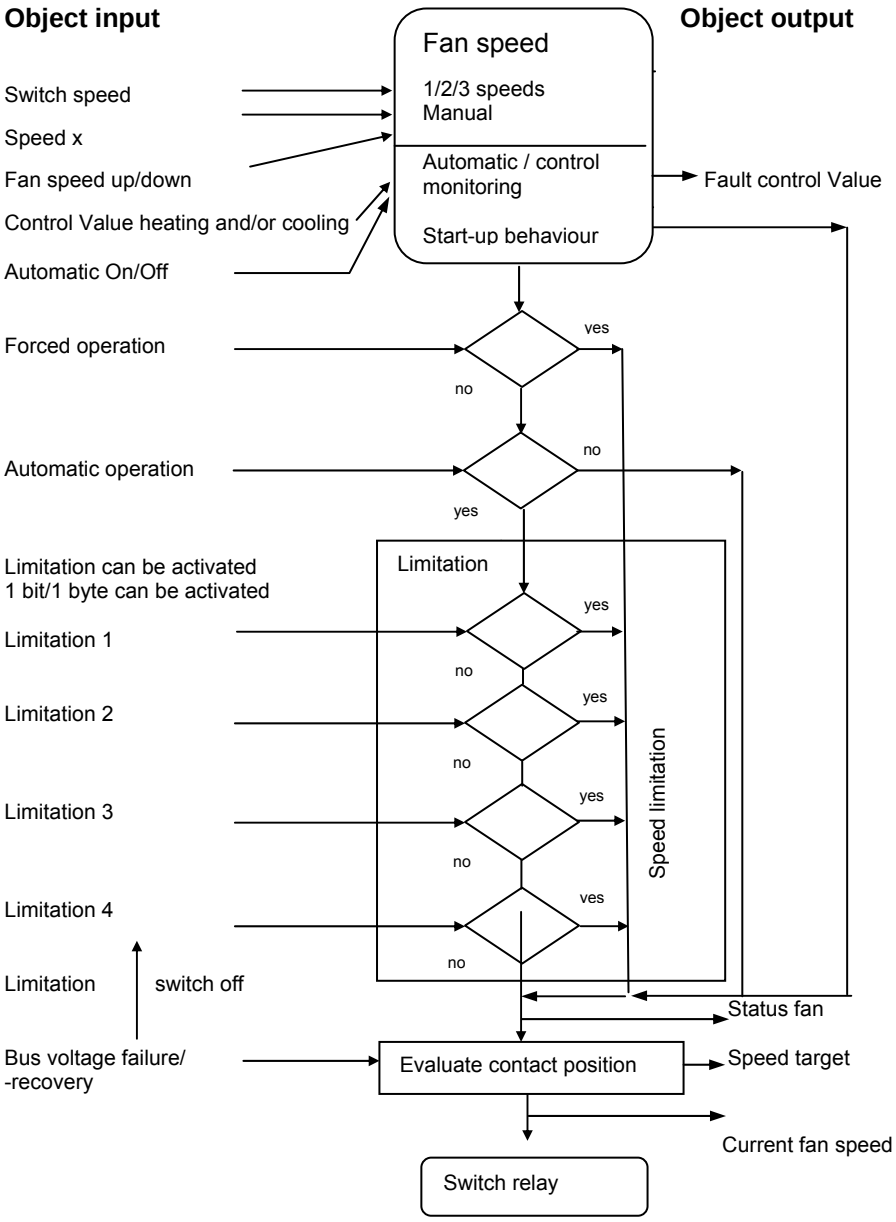
The following illustration indicates the logic of a switchover speed for a Fan Coil Actuator in dependence on the control values and the parameterised threshold values and hysteresis.

The diagram relates to a three speed fan without parameterised fan limitations. The fan limitations are only relevant after the fan speed has been determined and do not change the flow chart.



4.2.5 Fan operation functional diagram

The following illustration indicates the sequence in which the functions of the fan control are processed. Communication objects, which lead to the same box have the same priority and are processed in the sequence in which the telegrams are received.



4.3 Valve drives, valves and controller

4.3.1 Electromotor valve drives

Electromotor valve drives open and close valves via a small electric motor. Electromotor valve drives are offered as proportional or as 2 or 3 way valve drives.

Proportional valve drives are controlled via an analogue signal, e.g. 0-10 V. They can not be controlled with the Fan Coil Actuator. 2 or 3 point valve drives are controlled via switching of the supply voltage.

2 point valve drives are controlled via the commands OPEN and CLOSE. The valve can be completely open or completely closed.
2 point valves are controlled via a 2 point control or pulse width modulation (PWM). 2-point valve drives which require 2 point control can not be controlled with the Fan Coil Actuator.

The Fan Coil Actuator supports the control of electric motor 3 point valve drives. These are connected via three connection cables to the Fan Coil Actuator: Neutral conductor, switched phase to OPEN,
switched phase for CLOSE. Using 3 point control valve drives, the valve can be opened by any desired percentage and the position can be retained over an extended period. If the valve does not move, no voltage is applied to the motor.

The valve is opened wide enough to allow the exact quantity of hot or cold water to flow that is required to bring the heat exchanger to the required temperature. Thus the valve is controlled via the valve opening (0…100 %). The control usually used in most cases is continuous control.

Alternatively, an electric motor valve drive is controlled via pulse width modulation. Thereafter, the valve is opened fully each time for a few minutes and then it is closed again.

4.3.2 Electro-thermal valve drives

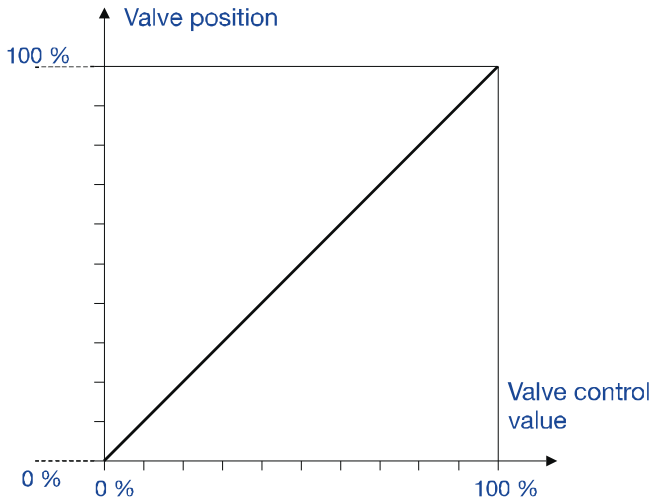
Electro-thermal drives are adjusted due to heat expansion of a material caused by a flow of electric current. Electro-thermal valve drives are controlled by pulse width modulation. The Fan Coil Actuator supports the control of electro-thermal valve drives via pulse width modulation.

Electro-thermal valve drives are offered in the *de-energised closed* and *de-energised opened* variants. Depending on the variant, the valve is opened when voltage is applied and closed when no voltage is applied, or vice versa.

Electro-thermal valve drives connected via two connection cables to the Fan Coil Actuator.

4.3.3 Valve curve

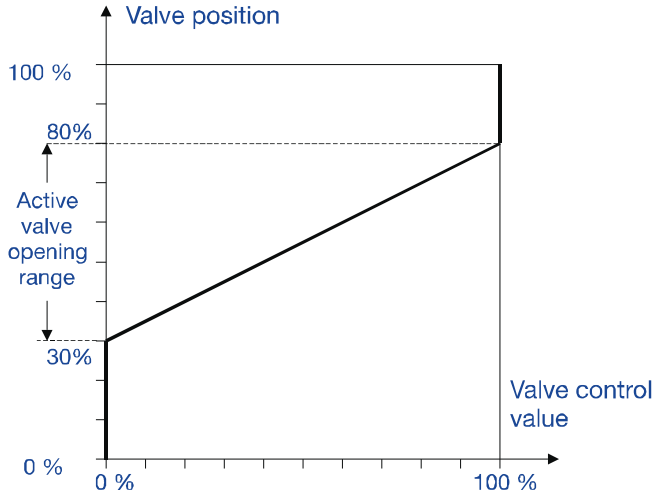
The Fan Coil Actuator controls valves with linear valve curves.
The valve control is matched linearly to the control value.
The valve is closed with a control value of 0 %, i.e. also 0 %.
The valve is fully open with a control value of 100 %, i.e. also 100 %.
The same ratio also applies for all intermediate values.



Linear valve curve

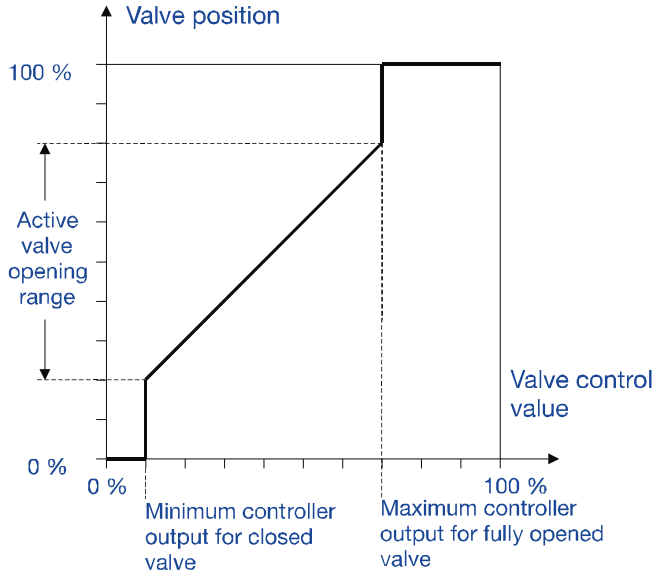
These valve curves can be matched for different valve types. Many valves e.g., have practically no flow when barely opened and achieve maximum flow at 60-80 %. Furthermore, many valves emit an annoying whistling sound at low flows.

These effects can be taken into consideration by limitation of the active valve opening range. The positioning frequency of the valve drive may also be reduced by this limitation.



Limitation of the active valve opening range

A further adaptation of the valve curve is implemented via the limitation of the valve control value. The valve output does not react in the upper and lower range due to this limitation. Thus for example, a valve movement with a minimal heating or cooling requirement can be avoided.



Limitation of the valve control value

A further adaptation of the curve can be undertaken in the parameter window *Curve* which is separately adjustable for the heating and the cooling valve. The valve control value can be adapted to the control value using the adjustable parameters there. The positioning frequency of the valve drive may also be reduced by this function.

A reduction of the positioning frequency reduces the current requirement for positioning and increases the service life of the valve. However, a reduced positioning frequency will also impair the accuracy of the temperature control.

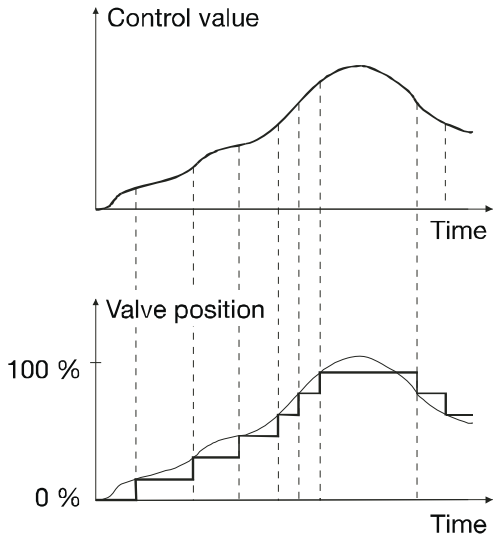
4.3.4 Control types

The following control types are commonly used for the control of valves in heating, air-conditioning and ventilation applications.

- Continuous control
- Pulse width modulation (PWM)
- Pulse width modulation – calculation

4.3.4.1 Continuous control

With continuous control, a control value is calculated based on the target temperature and the actual temperature and is used for optimum control of the temperature. The valve is brought to a position which complies with the calculated control value. With this method the valve can be fully opened, fully closed and even positioned in every intermediate position.



With continuous control the most accurate control of the temperature can be achieved without any resulting overshoots. At the same time the positioning frequency of the valve drive can be kept low.
Continuous control can be implemented with the Fan Coil Actuator for electro-motor 3 point or ABB i-bus® valve drives.
This is implemented via a 1 byte control.

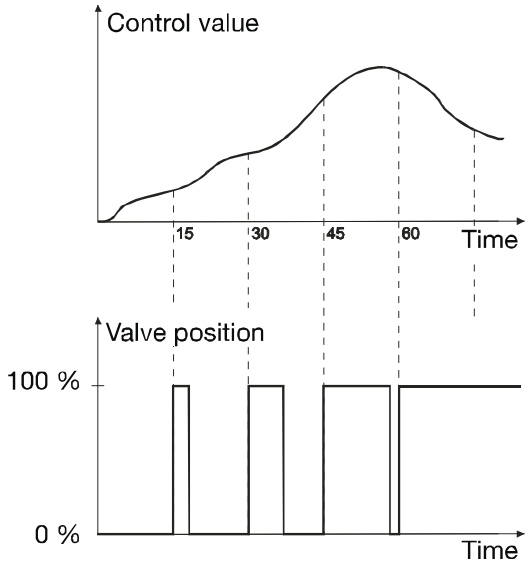
What is a 1 byte control?

For 1 byte control, a value of 0…255 (corresponds to 0 %…100 %) is preset by the room thermostat. At 0 % for example, the valve is closed and at 100 % it is fully opened.

4.3.4.2 Pulse width modulation (PWM)

With pulse width modulation the valve is operated as with 2 point control exclusively in the positions *fully opened* and *fully closed*. In contrast to a 2 point control the position is not controlled via limit values, but rather by calculated control values similar to continuous control.

The control value is fixed for a timed cycle and recalculated in the duration for valve opening. The control value 20 % at a cycle time of 15 minutes, e.g., will be recalculated for a valve opening time of three minutes. The control value 50 % results in a valve opening time of 7.5 minutes.

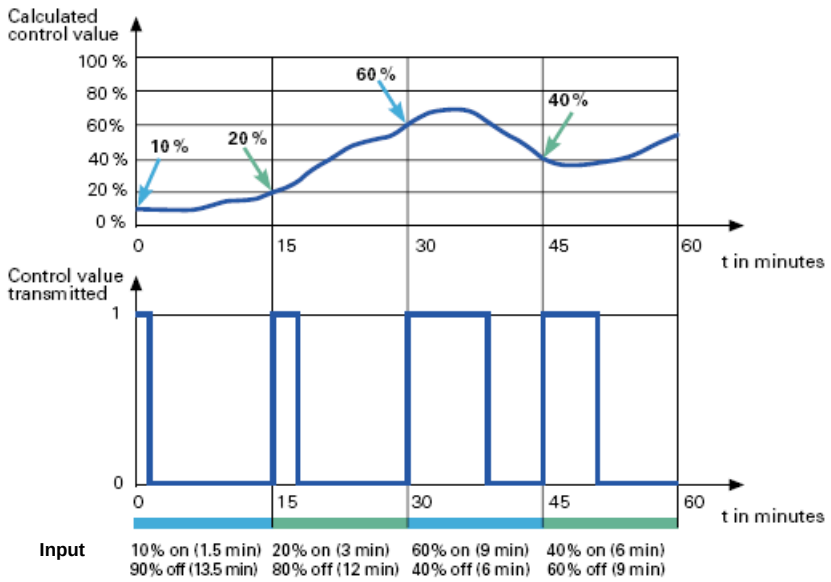


With pulse width modulation a relatively accurate control of the temperature can be achieved without any resulting overshoots. Simple, attractively-priced control valves can be used. The positioning frequency of the control valve is relatively high.

Pulse width modulation can be used with the Fan Coil Actuator in conjunction with electromotor or electro-thermal valve drives.

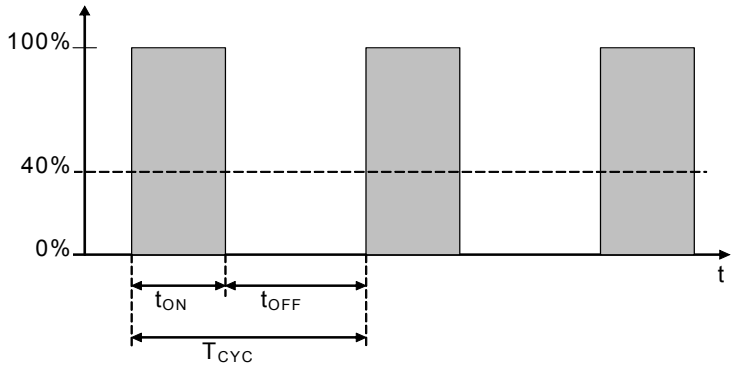
An example of this is when the FCA/S receives a 1 byte control value (continuous control) as an input signal, and this value together with the parameterised cycle time from a PWM calculation is converted into a signal for a 2 point control (on-off-value).

With PWM control, the received control value (0...100 %) calculated in the control algorithm is converted to a pulse width modulation. The conversion is based on a constant cycle time. If the FCA/S e.g., receives a control value of 20 %, then for a cycle time of 15 minutes the valve will be opened for three minutes (20% of 15 minutes) and closed for 12 minutes (80 % of 15 minutes).



4.3.4.3 Pulse width modulation – calculation

With pulse width modulation the control is implemented by a variable mark-space ratio.



During the time t_{ON} the valve is opened and during the time t_{OFF} it is closed. Due to t_{ON} = 0.4 x t_{CYC} the valve is set to about 40 % on. t_{CYC} is the so-called PWM cycle time for continuous control.

4.4 Behaviour with, …

4.4.1 Bus voltage recovery (BW)

General

- At bus voltage recovery (BW) the object values can be parameterised, if not they are set to the value 0. For exceptions refer to the table, e.g. automatic operation.
- Timers are out of operation and should be restarted.
- Status objects are sent as long as the option *after a change* has been set.
- The contact position is not known with 100 % certainty after bus voltage recovery. It is assumed that the contact position has not changed during the bus failure (no manual operation possibilities). Only after a new switch event is the contact position known to the Fan Coil Actuator.
- The send delay is only active at bus voltage recovery!

Switch contact output

- The object value *Staircase lighting time* remains unchanged as before bus voltage failure.
- The object value *Disable Time Function* is independent of the selected option.
- The object value *Permanent ON* remains unchanged as before bus voltage failure.
- The switch contact output switches as follows:
  - After the set object value *Switch* with bus voltage recovery.
  - Is the parameter *Object value "Switch" at bus voltage recovery* is not parameterised, the behaviour at bus voltage failure is decisive.
  - If none of the two above options is selected, the last position is retained as with bus voltage failure.

Note
If a staircase lighting time was active at bus voltage failure, it will restart.

Inputs

- The inactive waiting time is only active at bus voltage recovery.

Valves

- The purging cycle restarts if it was active before the failure.
- The priorities blocking, forced operation, purging and adjustment are re-established and executed as priorities.

  The priorities are defined as follows:

  1. Referencing
  2. Manual operation
  3. Communication object *Block*
  4. Communication object *Forced operation*
  5. Valve Purge
  6. Adjustment
  7. Control variables

Note
1 corresponds to the highest priority.

- The value parameterised for bus voltage recovery is only carried out if no higher priority (with the exception of manual operation/reference run) was active before the failure. If during bus voltage recovery and an active priority a new *Control value* is received, it will replace the *Control value* which was defined in the parameterisation.

4.4.2 Reset via Bus

What is an ETS Reset?

Generally and ETS Reset is defined as a reset of the device via the ETS. The ETS Reset is initiated in the ETS3 under the menu point *Commissioning* with the function *Reset device*. This stops the user program and it is restarted.

Switch contact output

- The object value *Staircase lighting time* receives its parameterised value.
- The object value *Disable Time function* is 0, i.e., the time function is not blocked.
- The object value *Permanent ON* is 0, i.e., permanent on is not active.
- The switch contact output goes to the safely opened state.

Note
For all resets after delivery including the first download, the response will comply with that of a reset via the bus. A send and switch delay is not executed. All states are reset.

4.4.3 Download (DL)

General

After a change of the fan control (speed control or changeover control) of the fan type a full reset of the actuator is required in order to avoid incorrect function. Those full reset has the same effect as reset of the device in the ETS. In this case the objects are normally written with the value 0. The timers stop and are set to 0. Status objects are set to 0 (with the exception of automatic, if it is active) and contacts are opened.

With the normal download, where no re-parameterisation of the fan type and fan control has occurred, an action has the effect that in the ideal case no unwanted reactions are initiated and thus normal operation is not influenced. Object values remain unchanged. Timer will not operate and must only be restarted. Status values are updated and sent. The contact position remains unchanged and only changes with the next switch command.

Note
After a download with a change, the application complies in behaviour to a reset of the device in the ETS. If the application of the same version is reloaded after discharge, the behaviour is the same as with a download

Switch contact output

The object value *Staircase lighting time* remains unchanged.

The object value *Disable Time function* remains unchanged.

Exception: The object value is set to 0 if there is no assignment to the communication object.

Note
Otherwise the block for the time function is removed, if the object *Disable Time function* is not available. The switch contact output will otherwise use the new parameters.

The object value *Permanent ON* remains unchanged.

The switch contact output remains unchanged.

4.4.4 Reaction on bus voltage failure

After the contact positions have set with bus voltage recovery, the Fan Coil Actuator remains functional until the bus voltage recovers.

Only the energy for a non-delayed switching action is available when the bus voltage fails for each output. Reversing times, dwell times and start-up behaviour cannot be considered. For this reason, it is only possible for the fan at bus voltage recovery to retain the fan speed (unchanged) or to switch off.

The special behaviour is described in the following table.

4.4.5 Behaviour with bus voltage failure, recovery and download

Behaviour of the fan speed on a download, ETS bus reset voltage failure and recovery

Behaviour with:	**Bus voltage recovery (BW)**	**Bus voltage failure (BF)**	**Download,** if no change of the operating function [1] occurs.	**ETS bus reset and download** (with change of operating function[1]) **complete reset**
Fans				
Fan speed	Can be parameterised	Can be parameterised	Unchanged or moves from a previously selected required speed, if this has not been achieved by switchover pauses and dwell times.	OFF, contacts open
Forced operation	Inactive	No function Fan speed as parameterised with BF	OFF, inactive	OFF, inactive
Limitation x x = 1…4	Inactive	No function Fan speed as parameterised with BF	OFF, inactive	OFF, inactive
Automatic operation	Automatic mode is activated, if automatic mode is possible.	No function	Is retained if already available. Remains inactive if already inactive.	Automatic mode is activated if automatic mode is possible, otherwise not active.
Communication object “Status automatic”	Is updated and sent in dependence on the parameterisation	No function	Is updated and sent in dependence on the parameterisation	Is updated and sent in dependence on the parameterisation (always, when changed, not)
Communication object Status Fan On/Off	Will be updated and sent	No function	Unchanged, implemented when the next telegram is received	Is updated (OFF, object value 0) and sent
Communication object Valve control	Values are recalculated and sent after the parameterised send delay	No function	Unchanged and sent	Cooling or cooling/heating, object value 0
Status byte	Values are updated and sent in dependence on the parameterisation	No function	Values are updated and sent in dependence on the parameterisation	Values are updated and sent in dependence on the parameterisation (always, when changed, not).

[1] An operating function can occur by the change from fan speed 1, 2 or 3 or to the switchover to a speed and changeover circuit of the fan control.

Behaviour of the output on a download, ETS bus reset voltage failure and recovery

Behaviour with:	**Bus voltage recovery (BW)**	**Bus voltage failure (BF)**	**Download,** if no change of the operating function [1] occurs.	**ETS bus reset and download** (with change of operating function[1]) **complete reset**
Output				
Communication object *Switch*	Can be parameterised	Communication object no longer available.	Unchanged. Evaluation only after a new event has been received.	Contacts go to a safe state. Renewed evaluation only after a new event has been received.
Time function inhibit communication object “Time function disable”	Can be parameterised	Communication object no longer available. Timer stops. Contact position parameterised with BF	Unchanged.	Contacts go to a safe state. Renewed evaluation only after a new event has been received.
Staircase lighting	In the parameter window you can be set if the time function is disabled or not disabled after bus voltage recovery. Timer stops. Light stays on, if staircase lighting time has run with BF. Otherwise unchanged. Change only after a new event has been received. The changed staircase lighting time set via the bus is lost and is replaced by the ETS programmed time.	No function Contact position with bus voltage failure can be parameterised	Unchanged. Change only after an event has been received. e.g. the staircase lighting remains on until it is started again or switched off	Running staircase lighting time stops. Switch contact is opened. Staircase lighting timer is set to 0. Staircase lighting time is set to the value parameterised in the ETS. The staircase lighting time sent via the bus is overwritten and is lost. If a time function is parameterised this will remain active. The communication object Disable time function is reset to the value 0 (time function activated).
Permanent ON	Permanent ON becomes inactive. Contact position is determined via object value “Switch”.	No function Contact position with bus voltage failure parameterised	Is inactive after a down load	Inactive

[1] An operating function can occur by the change from fan speed 1, 2 or 3 or to the switchover to a speed and changeover circuit of the fan control.

Behaviour of the valves on a download, ETS bus reset voltage failure and recovery

Behaviour with:	**Bus voltage recovery (BW)**	**Bus voltage failure (BF)**	**Download,** if no change of the operating function [1] occurs.	**ETS bus reset and download** (with change of operating function[1]) **complete reset**
Valves			Object values are available	
Valve operation (contact position)	Can be parameterised	Can be parameterised	Calculation (PWM) / evaluation will be continued with the existing object values (input values) Object value is retained	Calculation / evaluation for valve control is set. Valve is closed (reference run = run time + 5 %)
Functions	Unchanged	Unchanged, however without function. Contact position is programmable.	Will be accepted, if changed	Will be accepted, if changed
Monitoring (communication object "Thermostat fault")	Monitoring time will be restarted. Object value is 0	No monitoring	Monitoring time will be restarted. Object value unchanged	Monitoring time will be restarted. Thermostat fault is reset
Behaviour forced operation	Inactive, must be reactivated	Inactive	Inactive	Becomes inactive
Valve Purge	Monitoring time restarts	Time is lost. No purging.	Monitoring time restarts	Monitoring time restarts

[1] An operating function can occur by the change from fan speed 1, 2 or 3 or to the switchover to a speed and changeover circuit of the fan control.

4.5 Priorities with, …

4.5.1 Valve Heating/Cooling

The priorities are defined as follows:

1. Referencing
2. Manual operation
3. Communication object "Block"
4. Communication object "Forced operation"
5. Valve Purge
6. Adjustment
7. Control variables

Note
1 corresponds to the highest priority.

4.6 Fast heat up/cool down

4.6.1 Heat up

If the new valve position is greater than the current position during heat up, the contact will close immediately.

The closing time is calculated from:

- T_{up} = Valve adjustment duration from 0 to 100%
- V_{cur} = Current valve position [0...255]
- V_{new} = New valve position [0...255]
- T_{new} = Switch on time of the PWM at the new valve position
- T_{cyc} = PWM cycle time
- T_{+1} = Is added on the way to V_{new} at every position passed through

Calculation of the closing time

$$T_{new} = \frac{T_{cyc}}{255 * V_{new}}$$

$$T_{+1} = \frac{T_{up}}{255} * \frac{V_{cur}}{255}$$

Calculation of the closing time at switchover

$$T = T_{new} + \left(T_{+1}\left[byV_{cur}\right]\right) + \left(T_{+1}\left[byV_{cur} + 1\right]\right) + \ldots + \left(T_{+1}\left[byV_{new}\right]\right)$$

This means:

- For a movement from 0…99 % the contact remains closed for about $T_{up} + T_{cyc}$.
- For a change in the lower % range it results in significantly shorter closing times than for changes in the upper % range.
- Thereafter the contact is opened in accordance with the new PWM cycle and the PWM cycle is started.

4.6.2 Cooling down

If the new valve position is less than the current position during cooling down, the contact will open immediately.

The opening time is calculated from:

- T_{down} = Valve adjustment duration from 100 to 0 %
- V_{cur} = Current valve position [0...255]
- V_{new} = New valve position [0...255]
- T_{new} = Switch off time of the PWM at the new valve position
- T_{cyc} = PWM cycle time
- T_{+1} = Is added on the way to V_{new} at every position passed through

Calculation of the opening time

$$T_{new} = \frac{T_{cyc}}{255 * (255 - V_{new})}$$

$$T_{+1} = \frac{T_{down}}{255} * \frac{255 - V_{cur}}{255}$$

Calculation of the opening time at switchover

$$T = T_{new} + (T_{+1}[byV_{cur}]) + (T_{+1}[byV_{cur} + 1]) + ... + (T_{+1}[byV_{new}])$$

This means:

- For a movement from 99…0 % the contact remains opened for about T_{down} + T_{cyc}.
- For a change in the lower % range it results in significantly shorter opening times than for changes in the upper % range.
- Thereafter the contact is opened in accordance with the new PWM cycle and the PWM cycle is started.

A Appendix

A.1 Scope of delivery

The Fan Coil Actuator is supplied together with the following components. Please check the items received using the following list.

- 1 pcs. FCA/S 1.1M, Fan Coil Actuator, MDRC
- 1 pcs. installation and operating instructions
- 1 pcs. bus connection terminal (red/black)

A.2 Status byte forced/operation

Bit No.		7	6	5	4	3	2	1	0
8 bit value	Hexadecimal	Forced operation	Limitation 4	Limitation 3	Limitation 2	Limitation 1	Thermostat fault	Automatic	Heating/Cooling
0	00								
1	01								■
2	02							■	
3	03							■	■
4	04						■		
5	05						■		■
6	06						■	■	
7	07						■	■	■
8	08					■			
9	09					■			■
10	0A					■		■	
11	0B					■		■	■
12	0C					■	■		
13	0D					■	■		■
14	0E					■	■	■	
15	0F					■	■	■	■
16	10				■				
17	11				■				■
18	12				■			■	
19	13				■			■	■
20	14				■		■		
21	15				■		■		■
22	16				■		■	■	
23	17				■		■	■	■
24	18				■	■			
25	19				■	■			■
26	1A				■	■		■	
27	1B				■	■		■	■
28	1C				■	■	■		
29	1D				■	■	■		■
30	1E				■	■	■	■	
31	1F				■	■	■	■	■
32	20			■					
33	21			■					■
34	22			■				■	
35	23			■				■	■
36	24			■			■		
37	25			■			■		■
38	26			■			■	■	
39	27			■			■	■	■
40	28			■		■			
41	29			■		■			■
42	2A			■		■		■	
43	2B			■		■		■	■
44	2C			■		■	■		
45	2D			■		■	■		■
46	2E			■		■	■	■	
47	2F			■		■	■	■	■
48	30			■	■				
49	31			■	■				■
50	32			■	■			■	
51	33			■	■			■	■
52	34			■	■		■		
53	35			■	■		■		■
54	36			■	■		■	■	
55	37			■	■		■	■	■
56	38			■	■	■			
57	39			■	■	■			■
58	3A			■	■	■		■	
59	3B			■	■	■		■	■
60	3C			■	■	■	■		
61	3D			■	■	■	■		■
62	3E			■	■	■	■	■	
63	3F			■	■	■	■	■	■
64	40		■						
65	41		■						■
66	42		■					■	
67	43		■					■	■
68	44		■				■		
69	45		■				■		■
70	46		■				■	■	
71	47		■				■	■	■
72	48		■			■			
73	49		■			■			■
74	4A		■			■		■	
75	4B		■			■		■	■
76	4C		■			■	■		
77	4D		■			■	■		■
78	4E		■			■	■	■	
79	4F		■			■	■	■	■
80	50		■		■				
81	51		■		■				■
82	52		■		■			■	
83	53		■		■			■	■
84	54		■		■		■		
85	55				■		■		■

Bit No.		7	6	5	4	3	2	1	0
8 bit value	Hexadecimal	Forced operation	Limitation 4	Limitation 3	Limitation 2	Limitation 1	Thermostat fault	Automatic	Heating/Cooling
86	56		■		■		■	■	
87	57		■		■		■	■	■
88	58		■		■	■			
89	59		■		■	■			■
90	5A		■		■	■		■	
91	5B		■		■	■		■	■
92	5C		■		■	■	■		
93	5D		■		■	■	■		■
94	5E		■		■	■	■	■	
95	5F		■		■	■	■	■	■
96	60		■	■					
97	61		■	■					■
98	62		■	■				■	
99	63		■	■				■	■
100	64		■	■			■		
101	65		■	■			■		■
102	66		■	■			■	■	
103	67		■	■			■	■	■
104	68		■	■		■			
105	69		■	■		■			■
106	6A		■	■		■		■	
107	6B		■	■		■		■	■
108	6C		■	■		■	■		
109	6D		■	■		■	■		■
110	6E		■	■		■	■	■	
111	6F		■	■		■	■	■	■
112	70		■	■	■				
113	71		■	■	■				■
114	72		■	■	■			■	
115	73		■	■	■			■	■
116	74		■	■	■		■		
117	75		■	■	■		■		■
118	76		■	■	■		■	■	
119	77		■	■	■		■	■	■
120	78		■	■	■	■			
121	79		■	■	■	■			■
122	7A		■	■	■	■		■	
123	7B		■	■	■	■		■	■
124	7C		■	■	■	■	■		
125	7D		■	■	■	■	■		■
126	7E		■	■	■	■	■	■	
127	7F		■	■	■	■	■	■	■
128	80	■							
129	81	■							■
130	82	■						■	
131	83	■						■	■
132	84	■					■		
133	85	■					■		■
134	86	■					■	■	
135	87	■					■	■	■
136	88	■				■			
137	89	■				■			■
138	8A	■				■		■	
139	8B	■				■		■	■
140	8C	■				■	■		
141	8D	■				■	■		■
142	8E	■				■	■	■	
143	8F	■				■	■	■	■
144	90	■			■				
145	91	■			■				■
146	92	■			■			■	
147	93	■			■			■	■
148	94	■			■		■		
149	95	■			■		■		■
150	96	■			■		■	■	
151	97	■			■		■	■	■
152	98	■			■	■			
153	99	■			■	■			■
154	9A	■			■	■		■	
155	9B	■			■	■		■	■
156	9C	■			■	■	■		
157	9D	■			■	■	■		■
158	9E	■			■	■	■	■	
159	9F	■			■	■	■	■	■
160	A0	■		■					
161	A1	■		■					■
162	A2	■		■				■	
163	A3	■		■				■	■
164	A4	■		■			■		
165	A5	■		■			■		■
166	A6	■		■			■	■	
167	A7	■		■			■	■	■
168	A8	■		■		■			
169	A9	■		■		■			■
170	AA	■		■		■		■	
171	AB	■		■		■		■	■

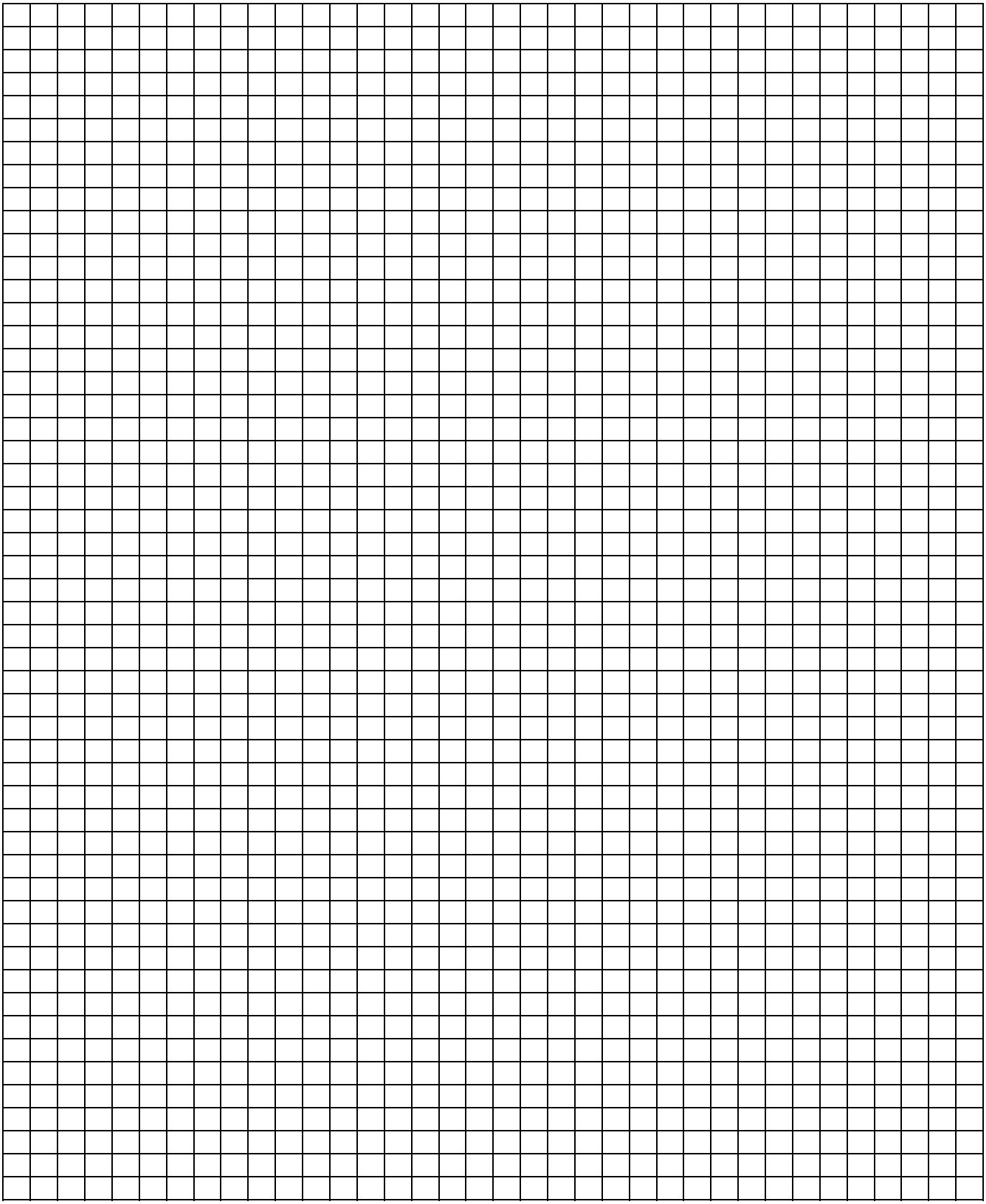
Bit No.		7	6	5	4	3	2	1	0
8 bit value	Hexadecimal	Forced operation	Limitation 4	Limitation 3	Limitation 2	Limitation 1	Thermostat fault	Automatic	Heating/Cooling
172	AC	■		■		■	■		
173	AD	■		■		■	■		■
174	AE	■		■		■	■	■	
175	AF	■		■		■	■	■	■
176	B0	■		■	■				
177	B1	■		■	■				■
178	B2	■		■	■			■	
179	B3	■		■	■			■	■
180	B4	■		■	■		■		
181	B5	■		■	■		■		■
182	B6	■		■	■		■	■	
183	B7	■		■	■		■	■	■
184	B8	■		■	■	■			
185	B9	■		■	■	■			■
186	BA	■		■	■	■		■	
187	BB	■		■	■	■		■	■
188	BC	■		■	■	■	■		
189	BD	■		■	■	■	■		■
190	BE	■		■	■	■	■	■	
191	BF	■		■	■	■	■	■	■
192	C0	■	■						
193	C1	■	■						■
194	C2	■	■					■	
195	C3	■	■					■	■
196	C4	■	■				■		
197	C5	■	■				■		■
198	C6	■	■				■	■	
199	C7	■	■				■	■	■
200	C8	■	■			■			
201	C9	■	■			■			■
202	CA	■	■			■		■	
203	CB	■	■			■		■	■
204	CC	■	■			■	■		
205	CD	■	■			■	■		■
206	CE	■	■			■	■	■	
207	CF	■	■			■	■	■	■
208	D0	■	■		■				
209	D1	■	■		■				■
210	D2	■	■		■			■	
211	D3	■	■		■			■	■
212	D4	■	■		■		■		
213	D5	■	■		■		■		■
214	D6	■	■		■		■	■	
215	D7	■	■		■		■	■	■
216	D8	■	■		■	■			
217	D9	■	■		■	■			■
218	DA	■	■		■	■		■	
219	DB	■	■		■	■		■	■
220	DC	■	■		■	■	■		
221	DD	■	■		■	■	■		■
222	DE	■	■		■	■	■	■	
223	DF	■	■		■	■	■	■	■
224	E0	■	■	■					
225	E1	■	■	■					■
226	E2	■	■	■				■	
227	E3	■	■	■				■	■
228	E4	■	■	■			■		
229	E5	■	■	■			■		■
230	E6	■	■	■			■	■	
231	E7	■	■	■			■	■	■
232	E8	■	■	■		■			
233	E9	■	■	■		■			■
234	EA	■	■	■		■		■	
235	EB	■	■	■		■		■	■
236	EC	■	■	■		■	■		
237	ED	■	■	■		■	■		■
238	EE	■	■	■		■	■	■	
239	EF	■	■	■		■	■	■	■
240	F0	■	■	■	■				
241	F1	■	■	■	■				■
242	F2	■	■	■	■			■	
243	F3	■	■	■	■			■	■
244	F4	■	■	■	■		■		
245	F5	■	■	■	■		■		■
246	F6	■	■	■	■		■	■	
247	F7	■	■	■	■		■	■	■
248	F8	■	■	■	■	■			
249	F9	■	■	■	■	■			■
250	FA	■	■	■	■	■		■	
251	FB	■	■	■	■	■		■	■
252	FC	■	■	■	■	■	■		
253	FD	■	■	■	■	■	■		■
254	FE	■	■	■	■	■	■	■	
255	FF	■	■	■	■	■	■	■	■

■ = applicable

A.3 Ordering information

Short description	Designation	Order No.	bbn 40 16779 EAN	Price group	Weight 1 pc. [kg]	Packaging [pc.]
FCA/S 1.1M	Fan Coil Actuator, MDRC	2CDG 110 084 R0011	**665 08 7**	26	0.1	1

A.4 Notes



Pub. No. 2CDC 508 062 D0201

Your KNX-Partner

www.abb.com/knx