thermokon
Sensortechnik GmbH

Protocol Description

thanos ModBus

&

thanos SR ModBus

1 Index of changes

Revision	Date	Description
A	25.07.2011	First draft
B	12.09.2011	Version LQ added
C	22.09.2011	Additional menu functions added
D	01.02.2012	Update to new firmware (“Operating Unit” 1.6 / “Net Unit” 1.4): - Description for thanos SR x MODBUS added Update to new firmware (“Operating Unit” 1.7.0 / “Net Unit” 1.5.0): - Description for thanos S / SQ added
E	30.03.2012	Update to new firmware (Operating Unit 1.8.0 / Net Unit: 1.6.0): - Description for Soft-/Firmwareupdates added - Register description for Standardscreen, Parameterversion and FanCoil OFF / AUTO added - Sundry corrections
F	10.07.2012	- Update to new configuration software (version 1.3.0.0). - Description for “Restart over Modbus” added.

Table of Contents

2 Introduction

The present document describes the serial interface of the room operating panel

- thanos ModBus
- thanos SR ModBus (functionality like thanos ModBus, but with additional EnOcean ↔ ModBus gateway)

For further information and definitions on the topic Modbus, please see www.modbus.org.

3 Description

3.1 Operating interface

For the thanos type S / SQ the operating interface is divided into one, and for the L / LQ into two zones:

- Menu area for control and display
- Keypad for control (Version L / LQ only)

On thanos S / SQ is also the possibility to configure submenus, over which a similar functionality as the keypad on thanos L / LQ is available. For details, see Chapter 5.4.

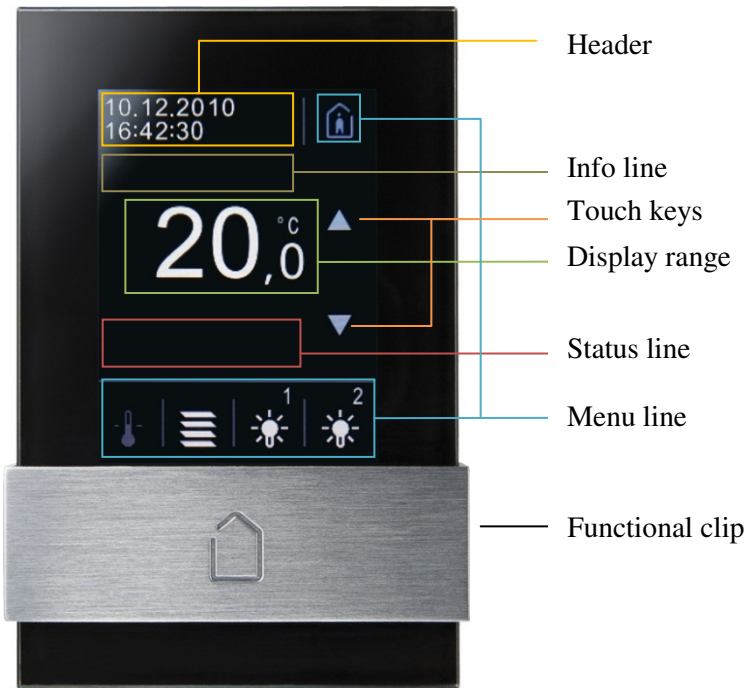


Figure 1 – thanos S

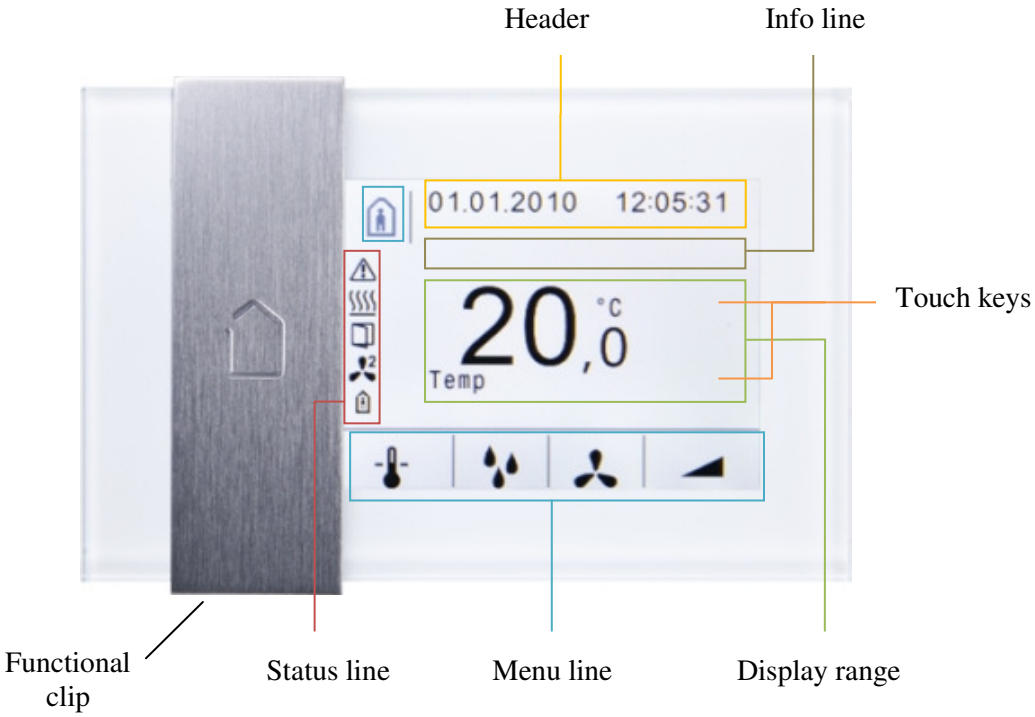


Figure 2 – thanos SQ

Figure 3-3 thanos L Operating interface

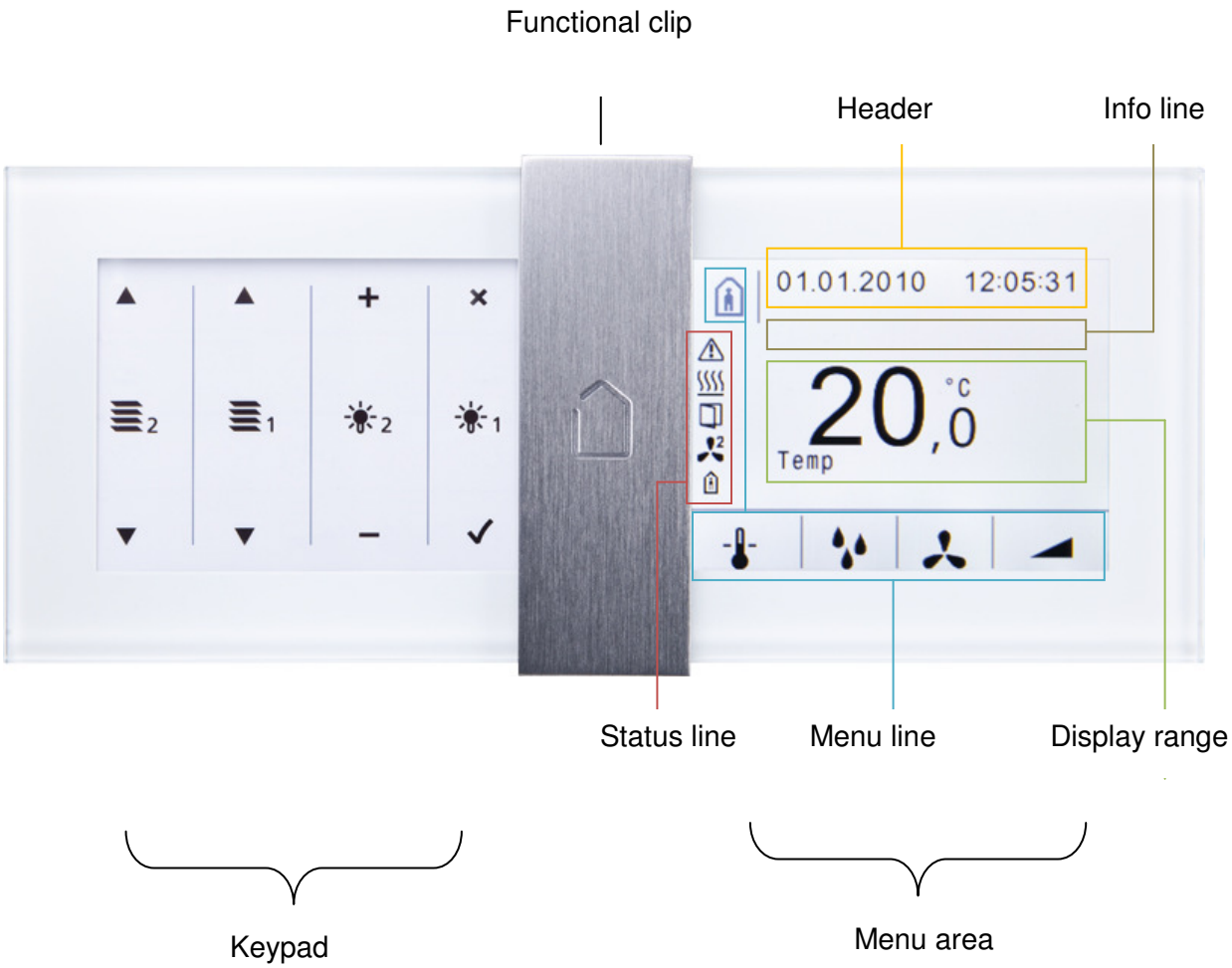


Figure 3-4 - thanos LQ operating interface

3.2 Menu area

Header:
In the header the current date and time can be displayed in different formats.
The thanos has a battery backed Real Time Clock so that the correct time is displayed even after a voltage breakdown.

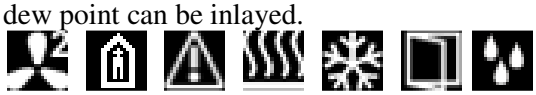
Info line:
In the info line a free selectable message text with a length of up to 14 signs can be displayed.

Display range:
Among others, the following values can be shown in the display range:

- Room temperature, relative humidity (optional)
- 6 effective set points and offset with free selectable unit and description
- 6 external values with free selectable unit and description

Moreover, the values and status of an active menu are displayed.

Status line:
In the status line the symbols for fan stages, room occupancy, failure, heating, cooling, window and dew point can be inlayed.



Menu line:
In the menu line different menu points can be saved. They can be called-off by the user when touching the corresponding symbol.

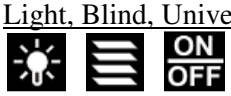
The following menu points can be parameterized:

Set point

Fan coil

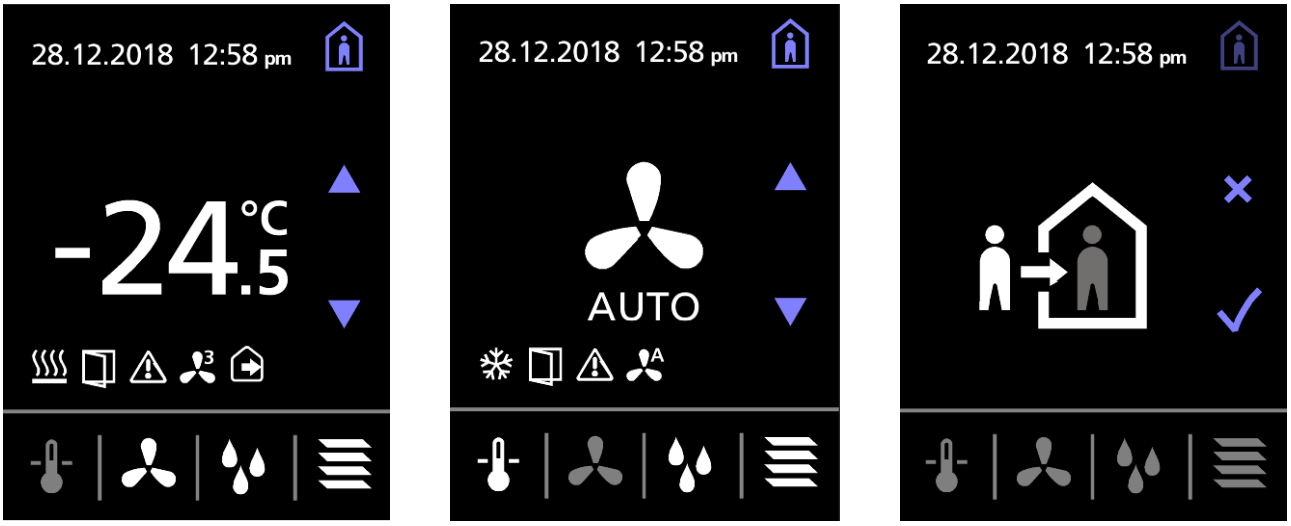
Occupancy mode

Light, Blind, Universal



If a menu point is selected the corresponding symbol is displayed grey-shaded in the menu line and in the display line the value/status is displayed which can be changed in the corresponding menu. By means of the operating key (depending on the function: either ▲ / ▼ or ✓ / ✕) the value/status can be changed afterwards.

Examples:



Menu „Temperature set point“ Menu „Fan coil“ Menu „Occupancy“

Figure 3-5 Menu

3.3 Touch keys

On thanos L / LQ the keypad consists of 8 keys in total. The keys are soft keys so that the functions of the keys can be freely adjusted via the configuration software. If a key is touched, the corresponding function is visually shown in the display.

Example:

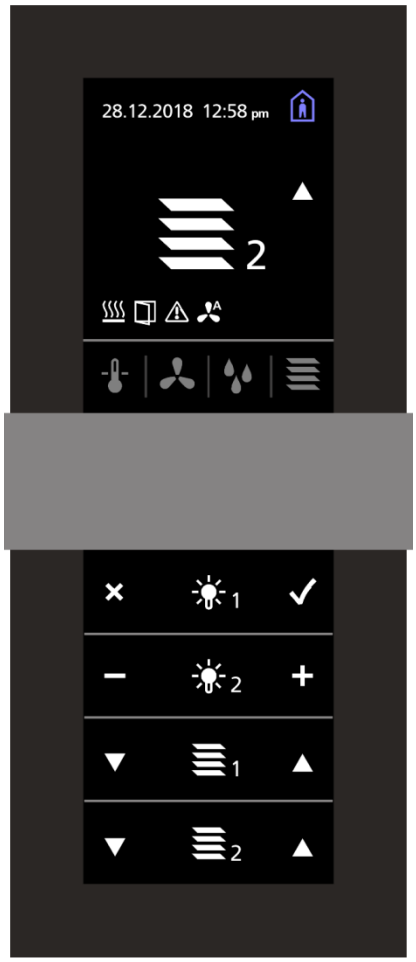


Figure 3-6 Touch keys

In the lower operating interface the key “blind 2 up” was selected. In the display area the corresponding symbol is displayed in big. Next to it the actuated symbol, e.g. ▲ is displayed.
After a freely programmable time the display indication is reset to the original display indication.

4 Hardware Installation

The transceiver can be connected with a twisted-pair-cable (resistance 120 Ohm) to the Bus. It is highly recommended to use shielded cables. The MODBUS-Protocol developed by company Modicon is a disclosed protocol for communication of several intelligent Master-Slave based devices.
For detailled information on installation and mounting please see the product data sheet thanos_Modbus and the data sheet wiring_rs485_network.pdf.

4.1 RS485 Transceiver

The maximum number of bus participants without use of a repeater is preset by the RS485-transceiver. The transceiver used allows 128 devices per bus segment at maximum.

4.2 Protocol

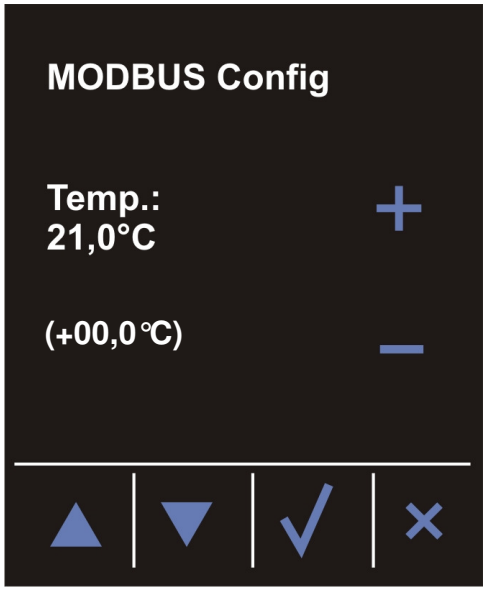
The thanos-MODBUS is a slave-bus participant only allowed to send to the bus on demand of the master. The protocol is identical with the defaults of:

- MODBUS Application Protocol Specification V1.1

- MODBUS via Serial Line Specification & Implementation guide V1.0

4.3 Start-up

For the Modbus specified parameter thanos disposes of an extra menu. The polling is made by simultaneous touching of key 1 and 7 for approx. 5 seconds. Afterwards the following display screen appears:



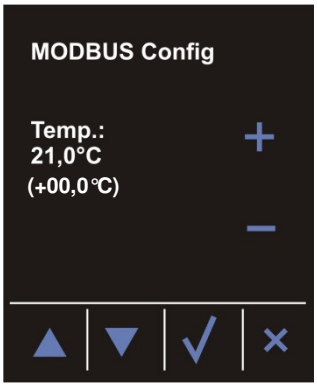
Key functions

Scrolling in the menu

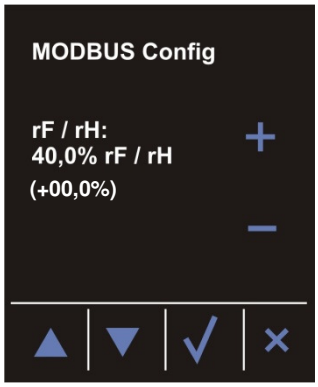
Set values

Accept/Abort the action. Both results in leaving the configuration menus.

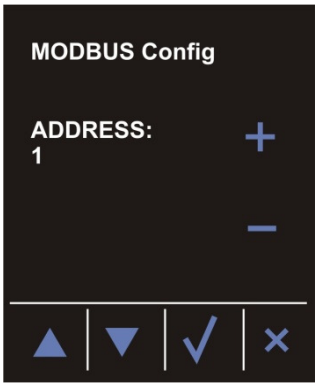
Following settings are available in modification mode: temperature offset, humidity offset, device address, baud rate and parity.



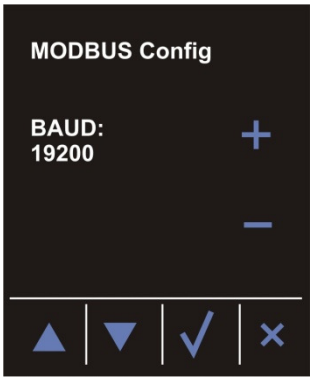
Menu temperature offset



Menu humidity offset



Menu address



Menu baudrate

Menu parity

4.3.1 Temperature offset

Each temperature sensor is factory calibrated. Because of the voltage-depending self-heating of the electronics and the temperature dynamic of the wall, in some cases a subsequent calibration can be necessary. The calibration mode enables a possibility for the user to calibrate the device itself via buttons.

4.3.2 Humidity offset (in case of existing humidity sensor)

Used for calibration of the humidity sensor

4.3.3 Device address

It is possible to adjust addresses from 1-127 – Default: 1.

4.3.4 **Baud rate**

Following baud rate options are supported:

1	9600
2	19200 - Default
3	38400
4	57600
5	115200

4.3.5 **Paritity**

Following parity options are supported:

1	Even - Default
2	Odd
3	None

Mode: RTU (not changeable)

Important notice for operation in the Master/Slave system:

!! The bus address must be adjusted differently for each device

!! Transmission mode, baud rate and parity must be identical

5 Function description

5.1 General

Among others, the menu „General“ includes general information on modules and assemblies as well as a parameter for setting the minimal response time and selection for device activation.

5.1.1 Register

5.1.1.1 Configuration

Configuration register (Holding Register R/W)		
Address	**Name**	**Description**
4x0000	Device coding	Internal Thermokon device code
4x0001	Firmware revision	Current firmware revision
4x0002	Device type	Thanos device type (L or S version 0=S, 1=L)
4x0003	Humidity sensor	Feedback if humidity sensor is available
4x0004	Rotation LC-display	Horizontal or vertical device rotation
4x0005	Device location identification	Assign a location specific code number to the device.
4x000C	Time cleaning function	By touching the functional clip for more than 10s the cleaning mode is activated. While cleaning mode is active the keys are not interpreted.
4x000D	Device Type	Reserved (read only)
4x000E	Modbus Minimum response time [ms]	Minimum response time of the device for adaptations due to master requirements
4x009B	Lock external values	The locking will be enabled when changing the status of room occupation, fan stages and set points as well as menu functions light, shutter/blind and universal. Due to a change of the above-named functions by the user, the corresponding input registers for the parameterized times will be decoupled which means that updates of the current input registers have no influence on them. Updates will only be adopted after expiration of the locking time. The locking provides time for the system to synchronize the current state with the room panel and the superior system.
4x0160	Volume button sound	Regulate volume of button sound between 0 and 100%
4x0171	Standardscreen	Selection if submenu #1 should work as “main menu”. If this is selected, no temperature, set points, etc. will be displayed in the display (only available on types S / SQ).
4x0172	Parameterversion	Version of the Configuration Parameters (read only)
4x0173	FanCoil “OFF / AUTO only”	If this parameter is selected, the user can change the fan stage only between OFF and AUTO.
Configuration bits		
Address	**Name**	**Description**
0x0004	Activate device first	Device is/is not activated by touching the functional clip to get access to the touch keys

Table 5-1 Overview general configuration bits and register general

5.2 Temperature

The temperature range is 0-50°C with a resolution of 0.1K. A possibility to set an offset is given due to possible deviations caused by outer influences. The indication of the temperature in the display can be enabled/disabled, shown with/without decimal place. Furhermore °C and °F are available as units.

5.2.1 Register

5.2.1.1 Configuration

Configuration register (Holding Register R/W)		
Address	**Name**	**Description**
4x0006	Temperature offset	Due to the fact that temperature measuring with flush-mounting sensors is besides the voltage-dependence, self-heating of the electronics also affected by the temperature dynamic of the wall, a recalibration might become necessary in some cases.
Configuration bits		
Address	**Name**	**Description**
0x0000	Display temperature ON/OFF	0= hide 1= show
0x0002	Resolution temperature	0=no decimal place 1= decimal place
0x0005	Unit temperature	0= °F 1= °C

Table 5-2 Overview configuration bits and register temperature

5.2.1.2 Output

Output register (Input Registers R)		
Address	**Name**	**Description**
3x0315	Temperature	Local temperature given by internal sensor value or external register. Includes configured offset in register 6: temperature and offset

Table 5-3 Overview output register temperature

5.3 Humidity

The humidity sensor (if existing) has an accuracy of ±3% in the range of 20-80% rH. The resolution is 0.1%. A possibility to set an offset is given due to possible deviations caused by outer influences. The indication of humidity in the display can be en-/disabled as well as the tenth.

5.3.1 Register

5.3.1.1 Configuration

Configuration register (Holding Register R/W)		
Address	**Name**	**Description**
4x0007	Offset humidity	Compensate deviations due to voltage-dependent self-heating of the electronics and temperature dynamic of the wall
Configuration bits		
Address	**Name**	**Description**
0x0001	Display humidity ON/OFF	0= hide 1= show
0x0003	Resolution humidity	0= no decimal place 1= decimal place

Table 5-4 Overview configuration register humidity

5.3.1.2 Output

Output Register		
Address	**Name**	**Description**
3x010F	Humidity	Local humidity given by internal sensor value or external register 4x0007: Offset humidity

Table 5-5 Overview output register humidity

5.4 Touch keys

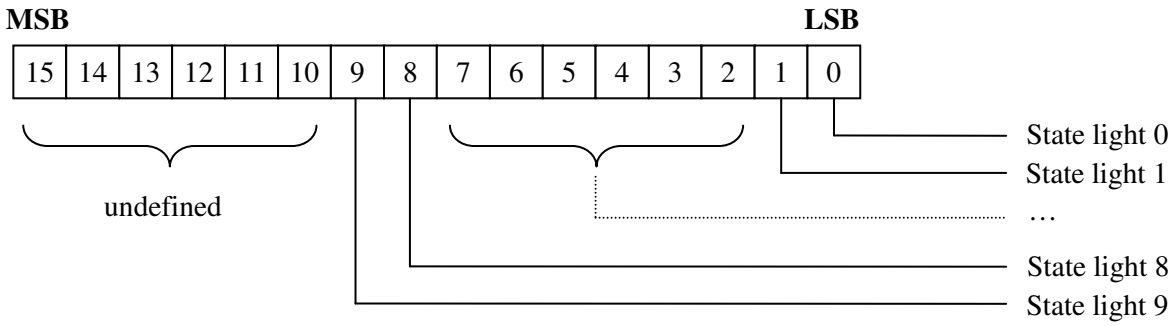
The operating unit of the thanos is divided into 3 areas. In the upper field, the menu area with up to 5 parameterizable buttons is placed, in the second part the direct keypad with 8 keys (L / LQ) or 24 keys (S / SQ – via submenus) is found and in the centre a capacitive function clip is placed.
The keys of the menu area can only be assigned with menu functions while the capacitive clip and the keys of the direct keypad can only be assigned by various toggling and on/off functions. Clip, menu area and keypad can be blocked by a superior BMS. Two output registers are available for indication of the touch keys. The first one is used for the indication of the current states. The second one is a memory function to save the key-actuation since the last read-out. This register is reset after read-out to the current state. There are further registers for the output of special functions (light, dimming, blinds and universal). In those extra registers the states of the functions (status of light, blind…) are indicated.
Furthermore customized indexes from 0-9 (one index for one channel) can be assigned to the functions (e.g. Light ON/OFF) which enables up to 10 functional channels. In every function-register all states are encoded bitwise (Bit 0=Index/channel 0, Bit1= Index/channel 1, Bit 2=Index/channel 2, …).

Example:
You decided to parameterize the keys as follows (see chapter 5.4.1.1):

Key 8= Light OFF Index 1	Key 9=Light ON Index 1
Key 10= Light OFF Index 2	Key 9=Light ON Index 2
Key 12= Light OFF Index 3	Key 9=Light ON Index 3
Key 14= Light OFF Index 4	Key 9=Light ON Index 4

So you created 4 light-channels. Their status can be read and are writable via the output register “status light function”.

Output register „status light function“:



Caution:
It is mandatory to connect each key with a channel/index (see chapter 5.4.1.3) which shall be occupied with a function like light, dimming, shutter/blind or universal.
If two shutters/blinds are necessary they have to be parameterized with index 1 and index 2 to differentiate between them while they were read out. If no index is given, index 0 will be automatically used which results in problems due to the lack of differentiation.

Furthermore the index will be indicated in the display - that provides for a certain identification.

Example: symbol for shutter with index 2.

5.4.1 Button Assignment

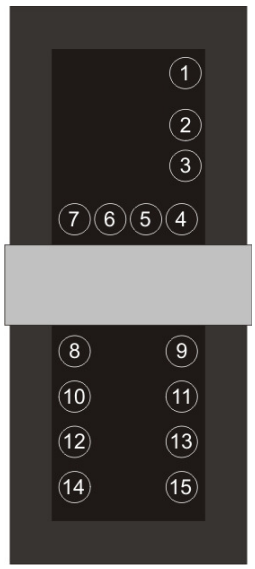


Figure 7
Buttons thanos L

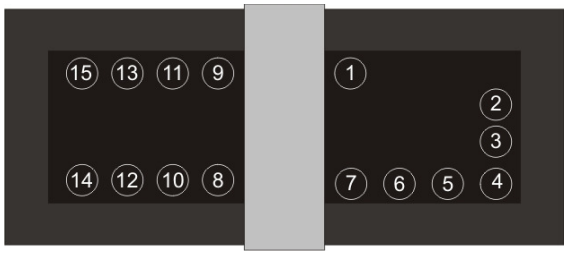


Figure 8
Buttons thanos L / LQ

Instead of the lower direct buttons, which are only available on thanos L / LQ, on thanos S / SQ up to 4 submenus can be configured.
Up to 6 configurable buttons for each submenu can be configured with the functions given below:

- Light on / off
- Light dimm + / -
- Light toggle
- Shutter / Blind up / down
- Universal on / off
- Universal toggle
- Occupancy toggle

To switch to the submenu, one of the buttons 1, 4…7 has to be configured as „sebmenu right“.

The buttons in the submenus are numbered consecutively (8 ... 31).

In addition, the number of submenus must be set via the parameter "number of sub-menus" (range 0 ... 4).

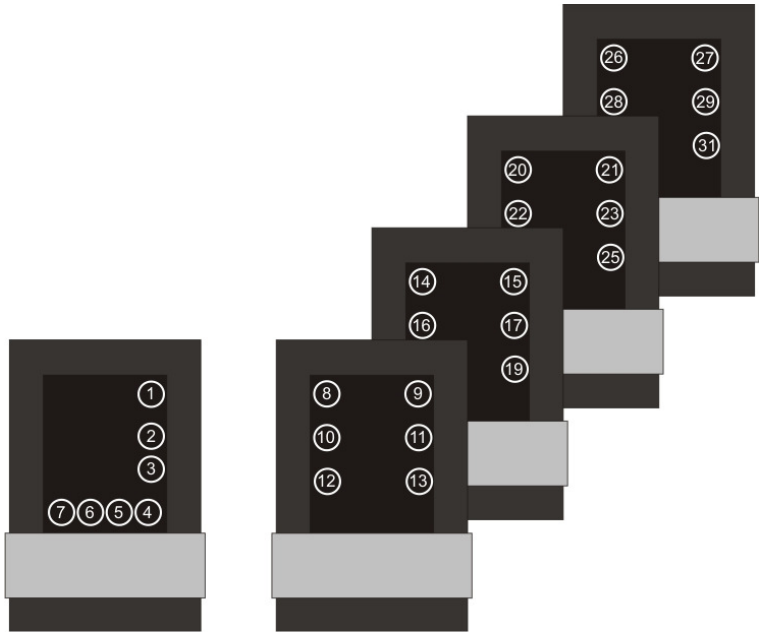


Figure 9
Buttons thanos S

Figure 10
Buttons thanos SQ

5.4.2 Register

5.4.2.1 Configuration

Configuration register (Holding Register R/W)		
Address	**Name**	**Description**
4x008D	Functional clip	Configuration of the functional clip
4x008E	Menu key 1	Configuration of the menu keys
4x008F	Menu key 4	
4x0090	Menu key 5	
4x0091	Menu key 6	
4x0092	Menu key 7	
4x0093	Direct key 8	Configuration of direct-keys
4x0094	Direct key 9	
4x0095	Direct key 10	
4x0096	Direct key 11	
4x0097	Direct key 12	
4x0098	Direct key 13	
4x0099	Direct key 14	
4x009A	Direct key 15	
Caution: Registers continued at **4x161**		
4x161	Direct key 16	
4x162	Direct key 17	
4x163	Direct key 18	
4x164	Direct key 19	
4x165	Direct key 20	
4x166	Direct key 21	
4x167	Direct key 22	
4x168	Direct key 23	
4x169	Direct key 24	
4x16A	Direct key 25	
4x16B	Direct key 26	
4x16C	Direct key 27	
4x16D	Direct key 28	
4x16E	Direct key 29	
4x16F	Direct key 30	
4x170	Direct key 31	

Table 5-6 Overview configuration register touch keys

Keys 2 and 3 cannot be changed because they are used for value modification in the menus.

5.4.2.1 Output

Output register				
Address	**Name**	**Description**		
3x0300	Current key status	Bit0: Functional clip Bit 1: Key 1 Bit 2: Key 2 … Bit 15: Key 15		
3x0301	Status of the key since the last call-off (Memory function)			
3x0302	Status light function		0 (FALSE)	1 (TRUE)
		Bit 0	Licht #0 OFF	Licht #0 ON
		Bit 1	Licht #1 OFF	Licht #1 ON
		…	…	…
		Bit 9	Licht #9 OFF	Licht #9 ON
3x0303	Current status of the „+“-Dimming key		0 (FALSE)	1 (TRUE)
		Bit 0	Dimming key „+“ not pressed channel 0	Dimming key „+“ pressed channel 0
		Bit 1	Dimming key „+“ not pressed channel 1	Dimming key „+“ pressed channel 1
		…	…	…
		Bit 9	Dimming key „+“ not pressed channel 9	Dimming key „+“ pressed channel 9
3x0304	Current status of the „-“-Dimming key		0 (FALSE)	1 (TRUE)
		Bit 0	Dimming key „+“ not pressed channel 0	Dimming key „+“ pressed channel 0
		Bit 1	Dimming key „+“ not pressed channel 1	Dimming key „+“ pressed channel 1
		…	…	…
		Bit 9	Dimming key „+“ not pressed channel 9	Dimming key „+“ pressed channel 9
3x0305	Status shutter /blind function		0 (FALSE)	1 (TRUE)
		Bit 0	Shutter/blind DOWN Channel 0	Shutter/blind UP Channel 0
		Bit 1	Shutter/blind DOWN Channel 2	Shutter/blind UP Channel 1
		…	…	…
		Bit 9	Shutter/blind DOWN Channel 9	Shutter/blind UP Channel 9
3x0306	Current status of the „+“-shutter/blind keys		0 (FALSE)	1 (TRUE)
		Bit 0	Shutter/ blind key „+“ not pressed channel 0	Shutter/ blind key „+“ pressed channel 0
		Bit 1	Shutter/ blind key „+“ not pressed channel 1	Shutter/ blind key „+“ pressed channel 1
		…	…	…
		Bit 9	Shutter/ blind key „+“ not pressed channel 9	Shutter/ blind key „+“ pressed channel 9
3x0307	Current status of the		0 (FALSE)	1 (TRUE)

	„-“-shutter/blind keys	Bit 0	Shutter/ blind key „+“ not pressed channel 0	Shutter/ blind key „+“ pressed channel 0
		Bit 1	Shutter/ blind key „+“ not pressed channel 1	Shutter/ blind key „+“ pressed channel 1
		…	…	…
		Bit 9	Shutter/ blind key „+“ not pressed channel 9	Shutter/ blind key „+“ pressed channel 9
3x0308	Status universal function		0 (FALSE)	1 (TRUE)
		Bit 0	Universal OFF channel 0	Universal ON channel 0
		Bit 1	Universal OFF channel 1	Universal ON channel 1
		…	…	…
		Bit 9	Universal OFF channel 9	Universal ON channel 9
Caution: Registers continued at **3x0347**				
3x0347	Current key status	Bit0: Key 16 Bit 1: Key 17 Bit 2: Key 18 … Bit 15: Key 31		
3x0348	Status of the key since the last call-off (Memory function)			

Table 5-7 Overview output register key

The register *Current key status* represents the key state.
Following registers are special illustrations for the extra functions light, dimming, shutter/blind and universal. The state of the different functions is presented here, but not the state of the key!

5.4.2.2 Input

The input register *Feedback light function*, *Feedback shutter/blind* and *Feedback universal* represent the feedbacks of the actuators when using the toggle function.

Input register				
Address	**Name**	**Description**		
4x040C	Feedback light function		0 (FALSE)	1 (TRUE)
		Bit 0	Light #0 OFF	Light #0 ON
		Bit 1	Light #1 OFF	Light #1 ON
		…	…	…
		Bit 9	Light #9 OFF	Light #9 ON
4x040D	Feedback shutter/blind		0 (FALSE)	1 (TRUE)
		Bit 0	Shutter/blind DOWN channel 0	Shutter/blind UP channel 0
		Bit 1	Shutter/blind DOWN channel 1	Shutter/blind UP channel 1
		…	…	…
		Bit 9	Shutter/blind DOWN channel 9	Shutter/blind UP channel 9
4x040E	Feedback universal		0 (FALSE)	1 (TRUE)
		Bit 0	Universal OFF channel 0	Universal ON channel 0
		Bit 1	Universal OFF channel 1	Universal ON channel 1
		…	…	…
		Bit 9	Universal OFF channel 2	Universal ON channel 9
Input bits				
Address	**Name**	**Description**		
0x0105	Lock keys	Functional clip, menu area and keypad can be locked by BMS.		

Table 5-8 Overview input bits keys

If the Thermokon configuration software is not used, the following table will be helpful for the parameterization of the keys.

Menus			
Index	**Description**		**Value**
1	No special function		0x0000
2	Menu set point 1		0x0001
3	Menu set point 1		0x0002
4	Menu set point 1		0x0003
5	Menu set point 1		0x0004
6	Menu set point 1		0x0005
7	Menu set point 1		0x0006
	Symbols for set points		HAVE TO BE OR-connected with index 2,3,…,7 Example: Showing menu set point 3 with humidity symbol: 0x2003
	A	Symbol temperature	0x1000
	B	Symbol humidity	0x2000
	C	Symbol value	0x3000
8	Menu fan stage		0x0007
9	Menu light		0x0008
10	Menu dimming		0x0009
11	Menu shutters/blinds		0x000A
12	Menu universal		0x000B
	Indications for menu points Light, dimming, shutter/blind and Universal (Value range: 0…9)		HAVE TO BE OR-connected with index 2,3,…,7 Example: Showing menu set point 3 light with index 3: 0x0308
	0	Index 0	0x0000
	1	Index 1	0x0100
	2	Index 2	0x0200
	3	Index 3	0x0300
	4	Index 4	0x0400
	5	Index 5	0x0500
	6	Index 6	0x0600
	7	Index 7	0x0700
	8	Index 8	0x0800
	9	Index 9	0x0900
13	Menu Occupancy		0x000C
Direct keys			
Index	**Description**		**Value**
14	Light on		0x00A0
15	Light off		0x00A1
16	Shutter up		0x00A2
17	Shutter down		0x00A3
18	Universal on		0x00A4
19	Universal off		0x00A5
20	Light toggle		0x00A6
21	Universal toggle		0x00A7
22	Occupancy toggle		0x00A8
23	Light dimm +		0x00A9
24	Light dimm -		0x00AA
	Indication for direct-keys (Value range: 0…9)		HAVE TO BE OR-connected with the configuration value Example: Direct-keys Light ON with

			Index 5: 0x05A0
	0	Index 0	0x0000
	1	Index 1	0x0100
	2	Index 2	0x0200
	3	Index 3	0x0300
	4	Index 4	0x0400
	5	Index 5	0x0500
	6	Index 6	0x0600
	7	Index 7	0x0700
	8	Index 8	0x0800
	9	Index 9	0x0900
23*	Submenu right		0x00C0

Table 5-9 Possibilities of button parameterization

*Submenus only exist in the S-Version!!

5.5 LC-Display

By means of the following configuration properties the indication of the display can be changed.
It is possible to adjust different brightness values for the LCD display and the keypad.
The different values are referring to an active mode, idle mode and stand-by mode.
Any action switches the display in active mode. After a parameterizable time without any actions, the device is reset to stand-by mode first and afterwards to idle mode.
Furthermore, e.g. the interval for toggling the display, submenu display duration, etc. can be changed.

5.5.1 Register

5.5.1.1 Configuration

Configuration register		
Address	**Name**	**Description**
4x000F	Background illumination LCD	It can be chosen between black and white for the background illumination of the LCD
4x0010	Contrast	The contrast of the display is only readable
4x0011	Background illumination LCD Standard-Mode	After an operation the device will be in standard mode (active). Values between 0 (OFF) and 100% are adjustable.
4x0012	Background illumination LCD Idle-Mode	Brightness of LCD in Idle-mode
4x0013	Background illumination LCD Standby-Mode	Brightness of LCD in Standby mode
4x0014	Background illumination key pad Standard	After the device will be in Standard mode (active). Values between 0 (OFF) and 100% are adjustable.
4x0015	Background illumination key pad Idle	Brightness of keypad in idle mode
4x0016	Background illumination key pad Standby	Brightness of keypad in standby mode
4x0017	Time till switch Standard- in Idle mode	Time interval without operation of the device till the LCD switches from Active mode to Idle mode
4x0018	Time till switch Idle- in Standby mode	Time interval without operation of the device till the LCD switches from Idle mode to Standby mode
4x0019	Indication time of a menu point	Time interval without operation of the device till the LCD switches out of a menu to default
4x001A	Indication time of an action	Time interval without operation of the device till the LCD switches out of an operation indication to default
4x001E	Register to configure existing submenus	Up to seven submenus can be chosen (Only available in version S!)
4x001F	Update interval of the LCD in seconds	Time interval of refreshing all values in the default screen
4x0020	Register to configure indication time of each submenu	Time interval without operation of the device resulting in switching back to default screen (Only available in version S!)
Configuration bits		
Address	**Name**	**Description**
0x0008	Indication of menu line 1	Show/hide menu line 1
0x0009	Indication of menu line 1	Show/hide menu line 2
0x000A	Indication of menu line 1	Show/hide menu line 3
0x000B	Indication of menu line 1	Show/hide menu line 4
0x000C	Indication of menu line 1	Show/hide menu line 5

Table 5-10 Overview configuration bits and configuration register LC-Display

5.5.1.2 Input

Bits		
Address	**Name**	**Description**
0x0109	Activate device illumination	Activate LC-Display and key-illumination

Table 5-11 Overview Bits LC-Display

By means of the bit activation of the TFT and key illumination the display can be put in the standard mode by a superior BMS.

5.6 Set points 1-6

Up to 6 set points can be indicated in the display as effective or offset values. The unit can be adjusted individually for each set point. A change of the set point is feasible via the keys or the network. Effective set point and adjusted offset are made available as output values.

5.6.1 Register

5.6.1.1 Configuration

Configuration register (Holding Register R/W)		
Address	**Name**	**Description**
4x0021 4x002C 4x0037 4x0042 4x004D 4x0059	Upper adjustable range of set point	Threshold value for upper range of set point offset adjustment by means of the keys
4x0022 4x002D 4x0038 4x0043 4x004E 4x0059	Lower adjustable range of set point	Threshold value for lower range of set point offset adjustment by means of the keys
4x0023 4x002E 4x0039 4x0044 4x004F 4x005A	Step range	Determines the step range of the manual set point adjustment by means of the keys.
4x0024 4x002F 4x003A 4x0045 4x0050 4x005B	Basic set point after reset	After a restart of the device, the value adjusted here is used as a basic set point. This value remains valid as long as another value is received by the input object base_setpoint_x.
4x0025 4x0030 4x003B 4x0046 4x0051 4x005C	1. ASCII character unit	Set point unit can be displayed with up to 3 characters
4x0026 4x0031 4x003C 4x0047 4x0052 4x005D	2. ASCII character unit	
4x0027 4x0032 4x003D 4x0048 4x0053 4x005E	3. ASCII character unit	
4x0028 4x0033 4x003E 4x0049 4x0054 4x005F	1. ASCII character of set point labeling	4 ASCII characters for labeling the set point
4x0029 4x0034 4x003F 4x004A 4x0055 4x0060	2. ASCII character of set point labeling	
4x002A 4x0035 4x0040 4x004B 4x0056 4x0061	3. ASCII character of set point labeling	

4x002B 4x0036 4x0041 4x004C 4x0057 4x0062	4. ASCII character of set point labeling	
Configuration bits		
Address	**Name**	**Description**
0x001A 0x001E 0x0022 0x0026 0x002A 0x002E	Resolution	Display set point with or without decimal place
0x001B 0x001F 0x0023 0x0027 0x002B 0x002F	Display with adjustment	Selection of set point display upon change of keys. It is possible to display the effective set point or the set point offset.
0x001C 0x0020 0x0024 0x0028 0x002C 0x0030	Display effective value	Select if the effective value shall be displayed cyclically in the main window.
0x001D 0x0021 0x0025 0x0029 0x002D 0x0031	Display offset value	Select if the offset value shall be displayed cyclically in the main window.

Table 5-12 Overview configurations bits and –register set points

5.6.1.2 Output

Output Register (Input Registers R)		
Address	**Name**	**Description**
3x0102 3x0104 3x0106 3x0108 3x010A 3x010C	Set point offset	Current set point offset. Can be changed by the user by means of keys actuation or via the input object Set point offset X.
3x0103 3x0105 3x0107 3x0109 3x010B 3x010D	Set point effective	The effective set point is calculated of the set point offset and the basic set point.

Table 5-13 Overview output register setpoints

5.6.1.3 Input

Input register (Holding Registers R/W)		
Address	**Name**	**Description**
3x0102 3x0104 3x0106 3x0108 3x010A 3x010C	Set point offset	External default of offset by a higher-level system.
3x0103 3x0105 3x0107 3x0109 3x010B	Base set point	External default of a basic set point by a higher-level system. As long as no valid value is received in this object, the value of the configuration property basic set point after reset is valid.

3x010D		
Bits		
Address	**Name**	**Description**
0x0108	Lock set point keys	Locks the keys of the set point adjustment.

Table 5-14 Overview bits und holding register set points

5.7 External values 1-6

Up to 6 external values can be displayed in the LCD. Each unit of the values can also be shown through three ASCII-symbols and a general description in the form of four ASCII-symbols.

5.7.1 Register

5.7.1.1 Configuration

Configuration register		
Address	**Name**	**Description**
4x0063 4x006A 4x0071 4x0078 4x007F 4x0086	1. ASCII character unit	The unit of the external values can be displayed with up to 3 characters
4x0064 4x006B 4x0072 4x0079 4x0080 4x0087	2. ASCII character unit	
4x0065 4x006C 4x0073 4x007A 4x0081 4x0088	3. ASCII character unit	
4x0066 4x006D 4x0074 4x007B 4x0082 4x0089	1. ASCII character of external value labeling	4 ASCII characters for labeling the external value
4x0067 4x006E 4x0075 4x007C 4x0083 4x008A	2. ASCII character of external value labeling	
4x0068 4x006F 4x0076 4x007D 4x0084 4x008B	3. ASCII character of external value labeling	
4x0069 4x0070 4x0077 4x007E 4x0085 4x008C	4. ASCII character of external value labeling	
Configuration bits		
Address	**Name**	**Description**
0x000E 0x0010 0x0012 0x0014	Resolution	Display external value with or without decimal place

0x0016 0x0018		
0x000F 0x0011 0x0013 0x0015 0x0017 0x0019	Display external value	Select if the external value shall be displayed cyclically in the main window.

Table 5-15 Overview configuration bits and register external values

5.7.1.2 Input

Input register		
Address	**Name**	**Description**
4x040F 4x0410 4x0411 4x0412 4x0413 4x0414	External value	External default for external values for indication in the display.

Table 5-16 Overview input register external values

5.8 Messages

Up to 8 messages of 14 byte length can be configured. Input register *4x0409, Show message*, must be written to select the message to be shown.

5.8.1 Register

5.8.1.1 Configuration

Configuration register		
Address	**Name**	**Description**
0x0200 0x0207 0x020E 0x0215 0x021C 0x0223 0x022A 0x0231	Message text	Up to 8 messages of 14 bytes length can be configured

Table 5-17 Overview configuration register messages

5.8.1.2 Eingabe

Input register (Holding Registers R/W)		
Address	**Name**	**Description**
4x0409	Show message	By means of the value 0 no message is displayed. With the values 1-8, the corresponding message 1-8 is inlayed.

Table 5-18 Overview input register messages

5.9 Symbols

In the display the symbols failure, heating, cooling, dew point and window can be indicated.
Illustration of the symbols can be found in chapter 3.2.

Symbol heating/cooling:
It is only possible to display one of both symbols, as the same position is allocated to both symbols.

5.9.1 Register

Bits		
Address	**Name**	**Description**
0x0100	Symbol failure	The symbol „failure“ can be faded in/out by a superior BMS.
0x0101	Symbol heating	The symbol „heating activated“ can be faded in/out by a superior BMS.
0x0102	Symbol cooling	The symbol „cooling activated“ can be faded in/out by a superior BMS.
0x0103	Symbol window	The symbol „window opened“ can be faded in/out by a superior BMS.
0x0104	Symbol dew point	The symbol „dew point“ can be faded in/out by a superior BMS.

Table 5-19 Overview bits symbols

5.10 Time and Date

The current time and date can be set by a superior BMS by means of an input register. An internal real time clock guarantees a sufficient accuracy of the time displayed and the calculation of the date so that only the time must be synchronized in necessary intervals during running operation.

5.10.1 Register

5.10.1.1 Configuration

Configuration register		
Address	**Name**	**Description**
4x001B	Display date	It is possible to choose between different time representations
4x001C	Display time	If time is shown it is possible to display the time with or without seconds.
4x001D	Time mode	The time can be shown in 12h or 24h mode

Table 5-20 Overview configuration register time and date

5.10.1.2 Input

Input register		
Address	**Name**	**Description**
4x0400	Input seconds	Time can be set via these registers. At the same time the internal time can be read out via these registers.
4x0401	Input minutes	
4x0402	Input hours	
4x0403	Input day of month	Date can be set via these registers. At the same time the internal date can be read out via these registers..
4x0404	Input month	
4x0405	Input year	

Table 5-21 Overview input register time and date

5.11 Fan coil

The fan stage can be changed by a higher-level system or locally via a key.
Up to 3 fan stages are feasible. It can be distinguished between manual or automatic mode.

The default of the fan stage can either be made in the manual or the automatic mode. In case the display of the fan stage in automatic mode is not requested, the fan stage display in the automatic mode can be switched off. Only the automatic symbol is displayed in this case and only the automatic byte (oxFFxx) must be preset by the network.

5.11.1 Register

5.11.1.1 Configuration

Configuration register		
Address	**Name**	**Description**
4x0008	Fan coil stages	Setting of the fan stages available at maximum. It is possible to freely configure if an automatic operation shall be available in addition to the manual operation.
4x0009	Fan stage after reset	Default fan coil stage after reset
Configuration coils		
Address	**Name**	**Description**
0x0006	Display fan stage after reset	After the boot up of the device, the fan coil stage is faded in automatically. If the setting is deactivated, the occupancy is faded in as soon as it has changed, regardless whether locally changed or by an update via the network.
0x0007	Display fan stage in auto mode	Selection if a fan stage shall be displayed in the automatic mode. Prerequisite is that the superior BMS provide the latest fan stage.

Table 5-22 Overview configuration bits and -register fan coil

5.11.1.2 Output

Output Register (Input Registers)		
Address	**Name**	**Description**
3x0317	Fan coil stage	Indicates current fan coil stage

Table 5-23 Overview output register fan coil

5.11.1.3 Input

Input register (Holding Registers)		
Address	**Name**	**Description**
4x0406	External fan stage	External default of the fan stage by a higher-level system.
Bits		
Address	**Name**	**Description**
0x0107	Lock fan coil	Locks the local modification of the fan stages.

Table 5-24 Overview input bits and –register fan coil

5.12 Occupancy

The configuration, input and output registers respectively bits belonging to the occupancy mode are listed in the following tables. Room occupancy can be changed by a higher-lever BMS as well as locally via the keys. The current status is determined by the value updated latest as both types have equal rights. An exception is the possibility to lock the external default. See chapter 6.1.10! Local change of room occupancy can be locked by the BMS.

5.12.1 Register

5.12.1.1 Configuration

Configuration register		
Address	**Name**	**Description**
4x000A	Status occupancy after reset	Default setup of the occupancy after a reset
4x000B	Bypass time Occupancy	If a time is set, the presence button on the device is automatically dedicated to bypass time extension.
Configuration register		
Address	**Name**	**Description**
0x0007	Display room occupancy after reset	After boot up of the device, room occupancy is faded in automatically. If the setting is deactivated, occupancy symbol is faded in as soon as it has changed, regardless whether locally changed or by an update via the network.

Table 5-25 Overview configuration bits and register occupancy

5.12.1.2 Output

Output Register		
Address	**Name**	**Description**
3x0318	Current room_occupancy	Outputs the current status of room occupancy.

Table 5-26 Overview output register occupancy

5.12.1.3 Input

Input register (Holding Register)		
Address	**Name**	**Description**
4x0407	Room occupancy	External default of room occupancy by a higher-level system.
Input bits (Coil Register)		
Address	**Name**	**Description**
0x0106	Lock occupancy	Locks the keys for room occupancy.

Table 5-27 Overview input register occupancy

The presence mode can be determined by a superior BMS as well as by a local presence button. The current status is determined by the value recently updated because both variants are equal. The local presence button can be locked by the BMS.

5.13 Digital Inputs

Depending on the type of the device up to 4 digital inputs are available which can be parameterized separetely. Each Input can be occupied with I/O for different functions. Those could be e.g. dew point, window contact, occupancy and controller release. The complete list can be found in table 7.1.

Configuration register		
Address	**Name**	**Description**
4x015C 4x015D 4x015E 4x015F	Function digital input	Configuration of a digital input

Output register		
Address	**Name**	**Description**
0x0343 0x0344 0x0345 0x0346	Status/Value of digital input	Idication of digital input is subject to parameteriztation **As a signal:** 0- Open 1-Closed **As a counter**: 0-65535 (flanks, impulses time/[s])

5.14 PI-controller

5.14.1 General

The thanos has 6 controllers which have a heating and a cooling function available each.
The corresponding configuration, input and output registers respectively bits are listed in the following table. One controller is adjusted for heating and cooling with an own parameter each. Moreover you can find a description of each function.
The control variable of the controller is re-calculated approx. every 10 seconds. Thus, changes, such as e.g. adjustment of set point or triggering of window contact are only considered after expiration of the control time.

5.14.2 Register

5.14.2.1 General controller configuration

Configuration register		
Address	**Name**	**Description**
4x009C 4x00BC 4x00DC 4x00FC 4x011C 4x013C	Base set point "Occupied‘	Controller base set point in occupied state. Equivalent to set point heating
4x009D 4x00BD 4x00DD 4x00FD 4x011D 4x013D	Set point offset "Standby"	Via the BMS the device can be switched into the standby. This parameter determines the difference of the standby set point to the basic set point in dependence on the status of the controller (heating or cooling mode).
4x009E 4x00BE 4x00DE 4x00FE 4x011E 4x013E	Set point offset ‚Unoccupied‘	Via the BMS or operation at the device, the occupancy state can be changed from „unoccupied“to „occupied“. This parameter determines the difference between the „unoccupied“ set point to the basic set point in dependence on the status of the controller (heating mode or cooling mode).
4x00A3 4x00C3 4x00E3 4x0103 4x0123 4x0143	Dead band	Dead band between set point heating and set point cooling Set point heating = base set point – (dead band / 2) Set point cooling = base set point + (deadband / 2)
4x00A8 4x00C8 4x00E8 4x0108 4x0128 4x0148	Controller mode after reset	Controller mode after reset and power-on
4x00A9 4x00C9 4x00E9 4x0109 4x0129 4x0149	Actual value selection	For the actual value of a controller the options for internal temperature sensor, internal humidity sensor or default of an external value via the input register are available.
4x00AA 4x00CA 4x00EA 4x010A 4x012A 4x014A	Set point selection	For the set point of a controller it is possible to use 1 of 6 internal set points or the default of an external value via the input register basic set point.
4x00AB 4x00CB 4x00EB 4x010B	Energy hold off selection	Selection if energy hold off shall only be triggered via the input register *energy lock* or only by the internal status or by both (logical OR circuit link).

4x012B 4x014B		
4x00AC 4x00CC 4x00EC 4x010C 4x012C 4x014C	Occupancy selection	Selection if the presence shall only be triggered via the input register *occupancy* or only by the internal status or by both (logical OR circuit link).
4x00AD 4x00CD 4x00ED 4x010D 4x012D 4x014D	Frost protection set point	The heating controller is released, independently on the adjusted operating mode and controller lock if the room temperature is lower than the frost protection set point. 0x00 deactivates antifreeze protection.
4x00AE 4x00CE 4x00EE 4x010E 4x012E 4x014E	Minimal control variable function	Detailed description can be found below
4x00AF 4x00CF 4x00EF 4x010F 4x012F 4x014F	Heat/Cool symbol access	Different access rights can be assigned to the controller.
4x00B0 4x00D0 4x00F0 4x0110 4x0130 4x0150	Fan coil stages	Number of fan coil stages used by the controller output
4x00B7 4x00D7 4x00F7 4x0117 4x0137 4x0157	PWM cycle time	With cycle time 0 the PWM-controller is deactivated. If the value exceeds 0, the current control variable is converted into a corresponding PWM signal and is output via the output register *PWM signal heating or cooling*.
4x00B8 4x00D8 4x00F8 4x0118 4x0138 4x0158	Assign fan coil controller	Selection if the fan of controller X has access to the „main“ fan.
4x00B9 4x00D9 4x00F9 4x0119 4x0139 4x0159	Display dew point	For every controller it can be configured if the dew point symbole shall be faded in on the display in case of dew point.
4x00BA 4x00DA 4x00FA 4x011A 4x013A 4x015A	Bypass time	Configuration of bypass time
4x00BB 4x00DB 4x00FB 4x011B 4x013B 4x015B	Minimal ON-time for fan coils	Configuration of the minimal ON-time after enabling the fan coil

Table 5-28 Overview configuration register and -bits controller

5.14.2.2 Heating controller configuration

Configuration register (Holding Register R/W)		
Address	**Name**	**Description**
4x009F 4x00BF 4x00DF 4x00FF 4x011F 4x013F	Xp	Proportional range Xp in Kelvin
4x00A0 4x00C0 4x00E0 4x0100 4x0120 4x0140	Tn	Reset time Tn
4x00A4 4x00C4 4x00E4 4x0104 4x0124 4x0144	Ymin	Minimal control variable limit in percent
4x00A5 4x00C5 4x00E5 4x0105 4x0125 4x0145	Ymax	Maximal control variable limit in percent
4x00B1 4x00D1 4x00F1 4x0111 4x0131 4x0151	Threshold stage 1 heating	Threshold values of the control variable used by the fan coil output
4x00B2 4x00D2 4x00F2 4x0112 4x0132 4x0152	Threshold stage 2 heating	
4x00B3 4x00D3 4x00F3 4x0113 4x0133 4x0153	Threshold stage 3 heating	

Table 5-29 Overview configuration register controller

5.14.2.3 Cooling controller configuration

Configuration register (Holding Register R/W)		
Address	**Name**	**Description**
4x00A1 4x00C1 4x00E1 4x0101 4x0121 4x0141	Xp	Proportional range Xp in Kelvin
4x00A2 4x00C2 4x00E2 4x0102 4x0122 4x0142	Tn	Reset time Tn
4x00A6 4x00C6 4x00E6 4x0106	Ymin	Minimal control variable limit in percent

4x0126 4x0146		
4x00A7 4x00C7 4x00E7 4x0107 4x0127 4x0147	Ymax	Maximum control variable limit in percent
4x00B4 4x00D4 4x00F4 4x0114 4x0134 4x0154	Threshold stage 1 cooling	Threshold values of the control variable used by the fan coil output
4x00B5 4x00D5 4x00F5 4x0115 4x0135 4x0155	Threshold stage 2 cooling	
4x00B6 4x00D6 4x00F6 4x0116 4x0136 4x0156	Threshold stage 3cooling	

Table 5-30 Overview configuration register controller

5.14.2.4 Input register

Input register (Holding Registers R/W)		
Address	**Name**	**Description**
4x0421 4x0429 4x0431 4x0439 4x0441 4x0449	Actual value	Actual value of controller, if option “external default” is assigned to configuration parameter *Actual value selection*
4x0422 4x042A 4x0432 4x043A 4x0442 4x044A	Occupancy	Selection between occupied, standby and unoccupied
4x0423 4x042B 4x0433 4x043B 4x0443 4x044B	Energy hold off	Default value by modbus master
4x0424 4x042C 4x0434 4x043C 4x0444 4x044C	Controller mode	Default value by modbus master. Enables the locking of an individual controller or both controllers by a superior BMS.
4x0425 4x042D 4x0435 4x043D 4x0445 4x044D	Base set point	Base set point of controller, if option “external default” is assigned to configuration parameter *Set point selection*
4x0426 4x042E 4x0436 4x043E 4x0446 4x044E	Dew point	Default value by modbus master
4x0427 4x042F 4x0437 4x043F 4x0447 4x044F	Trigger bypass time	Writing a value > 0 leads to the re-triggering of the bypass time. Writing a 0 (null) results in the immediate reset of the bypass time.

4x0428 4x0430 4x0438 4x0440 4x0448 4x0450	Set point offset	Default value by modbus master

Table 5-31 Overview input register controller

5.14.2.5 Output register

Output register (Input Registers R)			
Address	**Name**	**Description**	
3x0319 3x0320 3x0327 3x032E 3x0335 3x033C	Control variable heating	Output register of control variable heating [%]	
3x031A 3x0321 3x0328 3x032F 3x0336 3x033D	Control variable cooling	Output register of control variable cooling [%]	
3x031B 3x0322 3x0329 3x0330 3x0337 3x033E	PWM output heating	Output register PMW signal heating (0 or 1)	
3x031C 3x0323 3x032A 3x0331 3x0338 3x033F	PWM output cooling	Output register PMW signal cooling (0 or 1)	
3x031D 3x0324 3x032B 3x0332 3x0339 3x0340	Controller mode	Outputs actual controller mode	Off Heating Cooling Auto
3x031E 3x0325 3x032C 3x0333 3x033A 3x0341	Fan stage	The fan stage is output in dependence of the thresholds and the current control variable of the controller.	

Table 5-32 Overview output register controller

5.14.3 Controller configuration

A controller is set with own parameters in case of heating or cooling. This enables an optimal adaption of the controller to its environment. It is freely selectable which set point respectively actual value should be used. Therewith a possibility to use internal values for control as well as external values, which are received via bus, is given to control different areas. Examples for a calculation of set points can be found at the end of this chapter.

5.14.4 Occupancy

The set point of the controller is defined by the status of occupancy. Possible states are *Occupied, Unoccupied* and *Standby*. Furthermore the status can be set either via the internal status of occupancy (without standby!!!) or via the superior BMS. Each controller disposes of a bypass mode (Partytime

extension). This is either started via the input register *Bypassmodus re-trigger* (=occupancy status via input register) or via the key (=occupancy status is connected with occupancy status of the controller and the key is configured as bypass-key).

5.14.5 **Controller type**

The controller can be applied as steady, as PWM or as fan coil controller. Therefore corresponding output registers are available.

5.14.6 **Energy hold off / dew point**

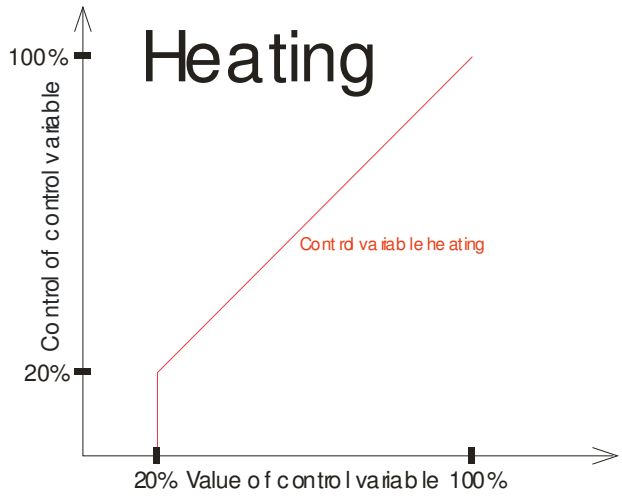
Energy hold off and dew point are directly affecting the control system. If the energy stop is activated the heating controller is automatically deactivated. In case of activated dew point detection the cooling controller is switched off. The energy stop can be triggered by an internal status as well as by the corresponding input register. If the energy stop is activated the anti-frost protection monitoring is active continuously!

5.14.7 Minimal control variable function

By means of the property "Use minimal control variable with control variable = 0"(configuration bit 8 = 0) the minimal control variable is only used, if the control variable is > 0. If bit 8 is = 1, the minimal control variable is also used if the control variable is = 0.

5.14.8 Mode selection variable

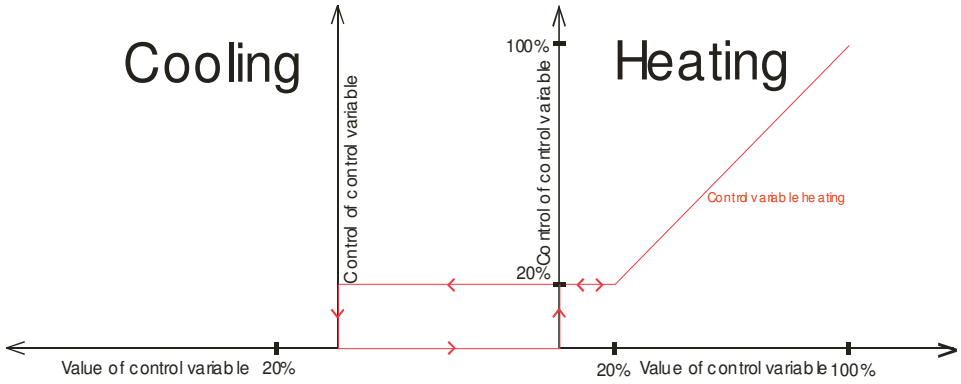
(1) Mode selection control variable = 1
Ymin = 20%



The control variable is only sent to the output if the calculated value of the control variable is bigger than the minimal control variable.

Figure 5-11 Controller sequence according to control variable = 1

(2) mode selection control variable = 0
Ymin = 20%



The minimal control variable at the output remains unchanged until the controller changes the operating mode

Figure 5-12 Controller sequence according to control variable = 0

5.14.9 Calculating Set Points:

(1) OCCUPIED

- *Heating set point* =basic set point – (dead band / 2) + offset
- *Cooling set point* = basic set point + (dead band / 2) + offset
-

(2) UNOCCUPIED

- *Heating set point* = basic set point- (dead band / 2) + offset – offset unoccupied
- *Cooling set point* = basic set point + (dead band / 2) + offset + offset unoccupied

(3) STANDBY

- *Heating set point* = basic set point - (dead band / 2) + offset – offset standby
- *Cooling set point* = basic set point + (dead band / 2) + offset + offset standby

5.15 Restart via Modbus

By the registers 0x0451 and 0x0452 a device restart can be initiated.

- Register 0x0451: 0x73A5
- Register 0x0452: 0x9C3A

If the correct value is written to these registers, the thanos makes a restart.

Input register (Holding register, Read and Write)		
Adresse	**Name**	**Description**
1106	Restart 1	The device will restart, if the registers contain the values given below: „Restart 1“ ⇒ 0x73A5 „Restart 2“ ⇒ 0xC93A
1107	Restart 2	

6 thanos-Modbus Protocol

6.1 Control Commands Supported

The following MODBUS – control commands are supported:

Description	Function code	
Read bits	01 (hex)	1 (dez)
	02 (hex)	2 (dez)
Read register	03 (hex)	3 (dez)
	04 (hex)	4 (dez)
Write individual bit	05 (hex)	5 (dez)
Write individual register	06 (hex)	6 (dez)
Write several bits	0F (hex)	15 (dez)
Write several registers	10 (hex)	16 (dez)

Table 1

6.2 Data Administration

All data in a MODBUS-Slave are assigned to addresses. Data access (read or write) is made by the corresponding control command and the indication of the corresponding data address.

Due to limited memory resources, the maximum number of readable and writable registers and coils in a telegram is limited in dependence on the transmitting mode(ASCII/RTU).

Procedure	RTU
Read register	48
Write register	48
Read coils	56
Write Coils	56

7 Register overview

7.1 Parameter

Configuration register (EEPROM)						
Register	**Modbus Data-address**	**Range**	**Type**	**Description**		**Default**
1 – 369R/W	**0x0000-0x0170**	**Configuration property !! Don't update permanently EEprom!!**				
1	4x0000	0x000A	uint16_t	Device code	Only readable	
2	4x0001	0x0000-0xFFFF	uint16_t	Firmware version Example: 0x1715=Operat.UnitV1.7 / Net Unit V1.5	Only readable	
3	4x0002	0x0000-0x0001	uint16_t	Device version (L or S Version 0=S, 1=L)	Only readable	
4	4x0003	0x0000-0x0001	uint16_t	Humidity sensor existing? 0=No 1=Yes	Only readable	
5	4x0004	0x0000-0x0001	uint16_t	Rotation Display	0-vertical, 1-horizontal	1
6	4x0005	0x0000-0xFFFF	uint16_t	Device location identification	0..65535	0
7	4x0006	0x0000-0xFFFF	int16_t	Temperature offset	e.g. 0x000A = 10 equal to. 1K e.g. 0xFFF6 = -10 equal to. -1K	0
8	4x0007	0x0000-0xFFFF	int16_t	Humidity offset	e.g. 0x000A = 10 equal to 1% e.g. 0xFFF6 = -10 equal to. -1%	0
9	4x0008	0x0000-0x0003 0xFF00-0xFF03	uint16_t	Setting of the maximal available fan stages (1...3 with/without Auto)	0x0000 = none 0x0001 = 1 stage 0x0002= 2 stages 0x0003 = stages 0xFF01 = 1 stage with Automatic 0xFF02= 2 stages with Automatic 0xFF03= 3 stages with Automatic	0
10	4x0009	0x0000-0x0003 0xFF00-0xFF03	uint16_t	Status fan stage after reset (1..3 with/without Auto)	0x0000 = OFF 0x0001 = Stage1 0x0002= Stage 2 0x0003 = Stage 3 0xFF00 = Auto OFF 0xFF01 = Auto Stage 1 0xFF02=Auto Stage 2 0xFF03=Auto Stage 3	0
11	4x000A	0x0000-0x0001	uint16_t	Status room occupancy after reset	0 = unoccupied 1 = occupied 2 = Standby	0
12	4x000B	0x0000-0xFFFF	uint16_t	Bypass time room occupancy	0...65535 = 0...65535 [s]	0
13	4x000C	0x0000-0xFFFF	uint16_t	Time cleaning function	0...65535 = 0...65535 [s]	30
14	4x000D	0x0000-0x0005	uint16_t	Device type	0 - AO2V 1 - DO2R 2 - DO2T 3 - OVR 4 - OVT 5 - 4DI	0
15	4x000E	0x0000-0x0064	uint16_t	Minimum response delay time [ms]	e.g. 0x000A = 10 equal to 10ms	5

16	4x000F	0x0000-0x0001	uint16_t	Background illumination LCD	0=black 1=white	0
17	4x0010	0x0000-0x001F	uint16_t	Contrast	Only readable	8
18	4x0011	0x0000-0x007F	uint16_t	Intensity background illumination LCD Standard-Mode	e.g. 0x007F = 128 equal to 100%	80
19	4x0012	0x0000-0x007F	Int16_t	Intensity background illumination LCD Idle-Mode		40
20	4x0013	0x0000-0x007F	Uint16_t	Intensity background illumination LCD Standby-Mode		10
21	4x0014	0x0000-0x0064	Uint16_t	Intensity background illumination keypad Standard	e.g. 0x0064= 100 equal to 100%	80
22	4x0015	0x0000-0x0064	Uint16_t	Intensity background illumination keypad Idle		40
23	4x0016	0x0000-0x0064	Uint16_t	Intensity background illumination keypad Standby		10
24	4x0017	0x0000-0xFFFF	Uint16_t	Time to switch Standard- in Idle mode	e.g.. 0x000A= 10 equal to 10s	30
25	4x0018	0x0000-0xFFFF	Uint16_t	Time to switch Idle- in Standby mode		60
26	4x0019	0x0000-0xFFFF	Uint16_t	Duration of displaying a menu point		10
27	4x001A	0x0000-0xFFFF	Uint16_t	Duration of displaying an operation		5
28	4x001B	0x0000-0x0002	Uint16_t	Date display	0 = off 1 = Display YYYY.MM.DD 2 = Display DD.MM.YYYY 3 = Display MM.DD.YYYY	2
29	4x001C	0x0000-0x0002	Uint16_t	Time display	0 = off 1 = Display with seconds 2 = Display without seconds	1
30	4x001D	0x0000-0x0001	Uint16_t	Time mode	0 = 24 hours 1 = 12 hours	0
31	4x001E	0x0000-0x0003	Uint16_t	Register to configure available submenus (only S version!!)	0 = without submenu 1 = 1 submenu … 3 = 3 submenus	0
32	4x001F	0x0000-0xFFFF	Uint16_t	Refresh interval of LCD in seconds	e.g. 0x000A= 10 equal to 10s	10
33	4x0020	0x0000-0xFFFF	Uint16_t	Display time of single submenus (only S version!!)	0 = Continous indication >= 1, 2, 3, ... Display in seconds	10
34	4x0021	0x0000-0xFFFF	Uint16_t	Upper range of adjustment Set point offset **Set point 1**	e.g. 0x0032 = 50 equal to 5,0K	30
35	4x0022	0x0000-0xFFFF	Uint16_t	Lower range of adjustment Set point offset **Set point 1**	e.g. 0x0032 = 50 equal to -5,0K	30
36	4x0023	0x0000-0xFFFF	Uint16_t	Step size set point adjustment **Set point 1**	e.g. 0x000A = 10 equal to 1K	5
37	4x0024	0x0000-0xFFFF	Uint16_t	Default set point after reset **Set point 1**	e.g. 0x00C8 = 200 equal to 20,0°C	220
38	4x0025	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **Set point 1**	e.g. 0x0053 equal to ‚S' 0x004F equal to ‚O' 0x004C equal to ‚L' (Indication bottom left)	0x20
39	4x0026	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **Set point 1**		0x20
40	4x0027	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **Set point 1**		0x20
41	4x0028	0x0000-0x00FF	Uint16_t	1. ASCII symbol set point description **Set point 1**		0x53 „S"

42	4x0029	0x0000-0x00FF	Uint16_t	2. ASCII symbol set point description **Set point 1**		0x45 „E“
43	4x002A	0x0000-0x00FF	Uint16_t	3. ASCII symbol set point description **Set point 1**		0x54 „T“
44	4x002B	0x0000-0x00FF	Uint16_t	4. ASCII symbol set point description **Set point 1**		0x31
45	4x002C	0x0000-0xFFFF	Uint16_t	Upper range of adjustment Set point adjustment **Set point 2**	e.g. 0x0032 = 50 equal to 5,0K	30
46	4x002D	0x0000-0xFFFF	Uint16_t	Lower range of adjustment Set point adjustment **Set point 2**	e.g. 0x0032 = 50 equal to. -5,0K	30
47	4x002E	0x0000-0xFFFF	Uint16_t	Step size set point adjustment **Set point 2**	e.g. 0x000A = 10 equal to. 1K	5
48	4x002F	Uint16_t	Uint16_t	Default set point after reset **Set point 2**	e.g. 0x00C8 = 200 equal to. 20,0°C	220
49	4x0030	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **Set point 2**	e.g.: 0x0053 equal to ‚S‘ 0x004F equal to ‚O‘ 0x004C equal to ‚L‘ (Indication bottom left)	0x20
50	4x0031	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **Set point 2**		0x20
51	4x0032	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **Set point 2**		0x20
52	4x0033	0x0000-0x00FF	Uint16_t	1. ASCII symbol set point description) **Set point 2**		0x53 „S“
53	4x0034	0x0000-0x00FF	Uint16_t	2. ASCII symbol set point description **Set point 2**		0x45 „E“
54	4x0035	0x0000-0x00FF	Uint16_t	3. ASCII symbol set point description **Set point 2**		0x54 „T“
55	4x0036	0x0000-0x00FF	Uint16_t	4. ASCII symbol set point description **Set point 2**		0x32
56	4x0037	0x0000-0xFFFF	Uint16_t	Upper range of adjustment Set point offset (Example: 50 = 5,0°C) **Set point 3**	e.g. 0x0032 = 50 equal to 5,0K	30
57	4x0038	0x0000-0xFFFF	Uint16_t	Lower range of adjustment Set point offset (Example: 50 = -5,0°C) **Set point 3**	e.g. 0x0032 = 50 equal to -5,0K	30
58	4x0039	0x0000-0xFFFF	Uint16_t	Step size set point adjustment **Set point 3**	e.g. 0x000A = 10 equal to 1K	5
59	4x003A	Uint16_t	Uint16_t	Default set point 3 after reset **Set point 3**	e.g. 0x00C8 = 200 equal to 20,0°C	220
60	4x003B	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **Set point 3**	e.g. 0x0053 equal to ‚S‘ 0x004F equal to ‚O‘ 0x004C equal to ‚L‘ (Anzeige unten links)	0x20
61	4x003C	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **Set point 3**		0x20
62	4x003D	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **Set point 3**		0x20
63	4x003E	0x0000-0x00FF	Uint16_t	1. ASCII symbol set point description **Set point 3**		0x53 „S“
64	4x003F	0x0000-0x00FF	Uint16_t	2. ASCII symbol set point description **Set point 3**		0x45 „E“
65	4x0040	0x0000-0x00FF	Uint16_t	3. ASCII symbol set point description **Set point 3**		0x54 „T“
66	4x0041	0x0000-0x00FF	Uint16_t	4. ASCII symbol set point description **Set point 3**		0x33
67	4x0042	0x0000-0xFFFF	Uint16_t	Upper range of adjustment Set point offset **Set point 4**	e.g. 0x0032 = 50 equal to 5,0K	30
68	4x0043	0x0000-0xFFFF	Uint16_t	Lower range of adjustment Set point offset **Set point 4**	e.g. 0x0032 = 50 equal to -5,0K	30

69	4x0044	0x0000-0xFFFF	Uint16_t	Step size set point adjustment **Set point 4**	e.g. 0x000A = 10 equal to 1K	5
70	4x0045	Uint16_t	Uint16_t	Default set point 4 after reset **Set point 4**	e.g. 0x00C8 = 200 equal to 20,0°C	220
71	4x0046	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **Set point 4**	e.g.: 0x0053 equal to ‚S' 0x004F equal to ‚O' 0x004C equal to ‚L' (Indication bottom left)	0x20
72	4x0047	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **Set point 4**		0x20
73	4x0048	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **Set point 4**		0x20
74	4x0049	0x0000-0x00FF	Uint16_t	1. ASCII symbol set point description **Set point 4**		0x53 „S"
75	4x004A	0x0000-0x00FF	Uint16_t	2. ASCII symbol set point description **Set point 4**		0x45 „E"
76	4x004B	0x0000-0x00FF	Uint16_t	3. ASCII symbol set point description **Set point 4**		0x54 „T"
77	4x004C	0x0000-0x00FF	Uint16_t	4. ASCII symbol set point description **Set point 4**		0x34
78	4x004D	0x0000-0xFFFF	Uint16_t	Upper range of adjustment Set point offset **Set point 5**	e.g. 0x0032 = 50 equal to 5,0K	30
79	4x004E	0x0000-0xFFFF	Uint16_t	Lower range of adjustment Set point offset **Set point 5**	e.g. 0x0032 = 50 equal to -5,0K	30
80	4x004F	0x0000-0xFFFF	Uint16_t	Step size set point adjustment **Set point 5**	e.g. 0x000A = 10 equal to 1K	5
81	4x0050	Uint16_t	Uint16_t	Default set point 5 after reset **Set point 5**	e.g. 0x00C8 = 200 equal to 20,0°C	220
82	4x0051	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **Set point 5**	e.g: 0x0053 equal to ‚S' 0x004F equal to ‚O' 0x004C equal to ‚L' (indication bottom left)	0x20
83	4x0052	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **Set point 5**		0x20
84	4x0053	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **Set point 5**		0x20
85	4x0054	0x0000-0x00FF	Uint16_t	1. ASCII symbol set point description **Set point 5**		0x53 „S"
86	4x0055	0x0000-0x00FF	Uint16_t	2. ASCII symbol set point description **Set point 5**		0x45 „E"
87	4x0056	0x0000-0x00FF	Uint16_t	3. ASCII symbol set point description **Set point 5**		0x54 „T"
88	4x0057	0x0000-0x00FF	Uint16_t	4. ASCII symbol set point description **Set point 5**		0x35
89	4x0058	0x0000-0xFFFF	Uint16_t	Upper range of adjustment Set point offset **Set point 6**	e.g. 0x0032 = 50 equal to 5,0K	30
90	4x0059	0x0000-0xFFFF	Uint16_t	Lower range of adjustment Set point offset **Set point 6**	e.g. 0x0032 = 50 equal to -5,0K	30
91	4x005A	0x0000-0xFFFF	Uint16_t	Step size set point adjustment **Set point 6**	e.g. 0x000A = 10 equal to 1K	5
92	4x005B	Uint16_t	Uint16_t	Default set point 5 after reset **Set point 6**	e.g. 0x00C8 = 200 equal to 20,0°C	220
93	4x005C	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **Set point 6**	e.g. 0x0053 equal to ‚S' 0x004F equal to ‚O' 0x004C equal to ‚L' (Indication bottom left)	0x20
94	4x005D	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **Set point 6**		0x20
95	4x005E	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **Set point 6**		0x20

96	4x005F	0x0000-0x00FF	Uint16_t	1. ASCII symbol set point description **Set point 6**		0x53 „S“
97	4x0060	0x0000-0x00FF	Uint16_t	2. ASCII symbol set point description **Set point 6**		0x45 „E“
98	4x0061	0x0000-0x00FF	Uint16_t	3. ASCII symbol set point description **Set point 6**		0x54 „T“
99	4x0062	0x0000-0x00FF	Uint16_t	4. ASCII symbol set point description **Set point 6**		0x36
100	4x0063	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **External value 1**	e.g. 0x0045 equal to ‚E‘ 0x0058 equal to ‚X‘ 0x0054 equal to ‚T‘ (Indication bottom left)	0x20
101	4x0064	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **External value 1**		0x20
102	4x0065	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **External value 1**		0x20
103	4x0066	0x0000-0x00FF	Uint16_t	1. ASCII symbol description **External value 1**		0x45 „E“
104	4x0067	0x0000-0x00FF	Uint16_t	2. ASCII symbol description **External value 1**		0x58 „X“
105	4x0068	0x0000-0x00FF	Uint16_t	3. ASCII symbol description **External value 1**		0x54 „T“
106	4x0069	0x0000-0x00FF	Uint16_t	4. ASCII symbol description **External value 1**		0x31
107	4x006A	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **External value 2**	e.g. 0x0045 equal to ‚E‘ 0x0058 equal to ‚X‘ 0x0054 equal to ‚T‘ (Indication bottom left)	0x20
108	4x006B	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **External value 2**		0x20
109	4x006C	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **External value 2**		0x20
110	4x006D	0x0000-0x00FF	Uint16_t	1. ASCII symbol description **External value 2**		0x45 „E“
111	4x006E	0x0000-0x00FF	Uint16_t	2. ASCII symbol description **External value 2**		0x58 „X“
112	4x006F	0x0000-0x00FF	Uint16_t	3. ASCII symbol description **External value 2**		0x54 „T“
113	4x0070	0x0000-0x00FF	Uint16_t	4. ASCII symbol description **External value 2**		0x32
114	4x0071	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **External value 3**	e.g. 0x0045 equal to ‚E‘ 0x0058 equal to ‚X‘ 0x0054 equal to ‚T‘ (Indication bottom left)	0x20
115	4x0072	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **External value 3**		0x20
116	4x0073	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **External value 3**		0x20
117	4x0074	0x0000-0x00FF	Uint16_t	1. ASCII symbol description **External value 3**		0x45 „E“
118	4x0075	0x0000-0x00FF	Uint16_t	2. ASCII symbol description **External value 3**		0x58 „X“
119	4x0076	0x0000-0x00FF	Uint16_t	3. ASCII symbol description **External value 3**		0x54 „T“
120	4x0077	0x0000-0x00FF	Uint16_t	4. ASCII symbol description **External value 3**		0x33
121	4x0078	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **External value 4**	e.g. 0x0045 equal to ‚E‘ 0x0058 equal to ‚X‘ 0x0054 equal to ‚T‘ (Indication bottom left)	0x20
122	4x0079	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **External value 4**		0x20
123	4x007A	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **External value 4**		0x20

124	4x007B	0x0000-0x00FF	Uint16_t	1. ASCII symbol description **External value 4**		0x45 „E“
125	4x007C	0x0000-0x00FF	Uint16_t	2. ASCII symbol description **External value 4**		0x58 „X“
126	4x007D	0x0000-0x00FF	Uint16_t	3. ASCII symbol description **External value 4**		0x54 „T“
127	4x007E	0x0000-0x00FF	Uint16_t	4. ASCII symbol description **External value 4**		0x34
128	4x007F	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **External value 5**	e.g. 0x0045 equal to ‚E‘ 0x0058 equal to ‚X‘ 0x0054 equal to ‚T‘ (Indication bottom left)	0x20
129	4x0080	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **External value 5**		0x20
130	4x0081	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **External value 5**		0x20
131	4x0082	0x0000-0x00FF	Uint16_t	1. ASCII symbol description **External value**		0x45 „E“
132	4x0083	0x0000-0x00FF	Uint16_t	2. ASCII symbol description **External value 5**		0x58 „X“
133	4x0084	0x0000-0x00FF	Uint16_t	3. ASCII symbol description **External value 5**		0x54 „T“
134	4x0085	0x0000-0x00FF	Uint16_t	4. ASCII symbol description **External value 5**		0x35
135	4x0086	0x0000-0x00FF	Uint16_t	1. ASCII symbol unit **External value 6**	e.g. 0x0045 equal to ‚E‘ 0x0058 equal to ‚X‘ 0x0054 equal to ‚T‘ (Indication bottom left)	0x20
136	4x0087	0x0000-0x00FF	Uint16_t	2. ASCII symbol unit **External value 6**		0x20
137	4x0088	0x0000-0x00FF	Uint16_t	3. ASCII symbol unit **External value 6**		0x20
138	4x0089	0x0000-0x00FF	Uint16_t	1. ASCII symbol description **External value 6**		0x45 „E“
139	4x008A	0x0000-0x00FF	Uint16_t	2. ASCII symbol description **External value 6**		0x58 „X“
140	4x008B	0x0000-0x00FF	Uint16_t	3. ASCII symbol description **External value 6**		0x54 „T“
141	4x008C	0x0000-0x00FF	Uint16_t	4. ASCII symbol description **External value 6**		0x36
142	4x008D	0x0000-0xFFFF	Uint16_t	Configuration functional clip	see chapter 5.4	0
143	4x008E	0x0000-0xFFFF	Uint16_t	Configuration menu key 1	see chapter 5.4	0
144	4x008F	0x0000-0xFFFF	Uint16_t	Configuration menu key 4		0
145	4x0090	0x0000-0xFFFF	Uint16_t	Configuration menu key 5		0
146	4x0091	0x0000-0xFFFF	Uint16_t	Configuration menu key 6		0
147	4x0092	0x0000-0xFFFF	Uint16_t	Configuration menu key 7		0
148	4x0093	0x00A0-0x00A8	Uint16_t	Configuration Direct key 8	see chapter 5.4	0
149	4x0094	0x00A0-0x00A8	Uint16_t	Configuration Direct key 9		0
150	4x0095	0x00A0-0x00A8	Uint16_t	Configuration Direct key 10		0
151	4x0096	0x00A0-0x00A8	Uint16_t	Configuration Direct key 11		0

152	4x0097	0x00A0-0x00A8	Uint16_t	Configuration Direct key 12		0
153	4x0098	0x00A0-0x00A8	Uint16_t	Configuration Direct key 13		0
154	4x0099	0x00A0-0x00A8	Uint16_t	Configuration Direct key 14		0
155	4x009A	0x00A0-0x00A8	Uint16_t	Configuration Direct key 15		0
156	4x009B	0x0000-0xFFFF	Uint16_t	Lock external defaults	0...65535 = 0...65535s	5
157	4x009C	0x0000-0xFFFF	Uint16_t	Default set point after reset "Comfort" **Controller 1**	0...65535 = 0...6553,5	220
158	4x009D	0x0000-0xFFFF	Uint16_t	Set point difference "Standby" **Controller 1**	0...65535 = 0...6553,,5	0
159	4x009E	0x0000-0xFFFF	Uint16_t	Set point difference "Unoccupied" **Controller 1**	0...65535 = 0...6553,5	0
160	4x009F	0x0000-0xFFFF	Uint16_t	Controller parameter XP Heating **Controller 1**	0...65535 = 0...6553,5	20
161	4x00A0	0x0000-0xFFFF	Uint16_t	Controller parameter TN Heating **Controller 1**	0...65535 = 0...65535 [s]	1000
162	4x00A1	0x0000-0xFFFF	Uint16_t	Controller parameter XP Cooling **Controller 1**	0...65535 = 0...6553,5	20
163	4x00A2	0x0000-0xFFFF	Uint16_t	Controller parameter TN Cooling **Controller 1**	0...65535 = 0...65535 [s]	1000
164	4x00A3	0x0000-0x03E8	Uint16_t	Dead zone **Controller 1**	0...65535 = 0...6553,5	10
165	4x00A4	0x0000-0x03E8	Uint16_t	Minimum control variable Heating **Controller 1**	0...100,0% = 0...1000dez	0
166	4x00A5	0x0000-0x03E8	Uint16_t	Maximum control variable Heating **Controller 1**	0...100,0% = 0...1000dez	1000
167	4x00A6	0x0000-0x03E8	Uint16_t	Minimumcontrol variable Cooling **Controller 1**	0...100,0% = 0...1000dez	0
168	4x00A7	0x0000-0x0003	Uint16_t	Maximum control variable Cooling **Controller 1**	0...100,0% = 0...1000dez	1000
169	4x00A8	0x0000-0x0002	Uint16_t	Controller mode after reset **Controller 1**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
170	4x00A9	0x0000-0x0005	Uint16_t	Selection actual value **Controller 1**	0 = Input register *Actual value (Addr. 4x0421)* 1 = int. Temperature 2 = int. Humidity	0
171	4x00AA	0x0000-0x0001	Uint16_t	Selection set value **Controller 1**	0 = Input register *Default set point (Addr. 4x0425>* >=1= int. Set point 1	0
172	4x00AB	0x0000-0x0001	Uint16_t	Selection energy hold-off **Controller 1**	0 = Window contact via Input register *Energy hold-off (Addr. 4x0423)* 1 = Window contact via internal state 2 = Register and internal state OR-connected	0
173	4x00AC	0x0000-0xFFFF	Uint16_t	Selection Un-/Occupied **Controller 1**	0 = Data output via Input register *Occupancy (Addr. 4x0422)* 1 = Data output via internal status 2 = Register and internal state OR-connected	0
174	4x00AD	0x0000-0x0001	Uint16_t	Frost protection-"Set point" **Controller 1**	0 - 0...6553,5 = 0...65535 (0 deactivates frost protection)	0
175	4x00AE	0x0000-0x0003	Uint16_t	Minimum control variable **Controller 1**	1 = Control variable > 0 0 = Control variable = 0	0

176	4x00AF	0x0000-0x0003	Uint16_t	Controller has access to symbols "Heating&Cooling " **Controller 1**	0= No access 1= Access to Heating 2= Access to Cooling 3= Access to Heating&Cooling	0
177	4x00B0	0x0000-0x03E8	Uint16_t	Number of fan stages **Controller 1**	0 = none 1 = Stage 1 2 = Stage 2 3 = Stage 3	0
178	4x00B1	0x0000-0x03E8	Uint16_t	Control variable threshold for fan stage 1 Heating **Controller 1**	0...100,0% = 0...1000dez	10
179	4x00B2	0x0000-0x03E8	Uint16_t	Control variable threshold for fan stage 2 Heating **Controller 1**	0...100,0% = 0...1000dez	333
180	4x00B3	0x0000-0x03E8	Uint16_t	Control variable threshold for fan stage 3 Heating **Controller 1**	0...100,0% = 0...1000dez	667
181	4x00B4	0x0000-0x03E8	Uint16_t	Control variable threshold for fan stage 1 Cooling **Controller 1**	0...100,0% = 0...1000dez	10
182	4x00B5	0x0000-0x03E8	Uint16_t	Control variable threshold for fan stage 2 Cooling **Controller 1**	0...100,0% = 0...1000dez	333
183	4x00B6	0x0000-0xFFFF	Uint16_t	Control variable threshold for fan stage 3 Cooling **Controller 1**	0...100,0% = 0...1000dez	667
184	4x00B7	0x0000-0x0001	Uint16_t	PWM cycle time **Controller 1**	0 = No PWM 1...65535 [s]	20
185	4x00B8	0x0000-0x0001	Uint16_t	Selection, if fan coil of controller X interacts with „Main“-Fan controller **Controller 1**	0 = No access 1 = access	0
186	4x00B9	0x0000-0xFFFF	Uint16_t	Display dew point signal of controller **Controller 1**	0 = hide 1 = show	0
187	4x00BA	0x0000-0xFFFF	Uint16_t	Bypass time **Controller 1**	0...6553,5 = 0...65535 [s]	10
188	4x00BB	0x0000-0xFFFF	Uint16_t	Minimal ON time for fan coil **Controller 1**	0...6553,5 = 0...65535 [s]	5
189	4x00BC	0x0000-0xFFFF	Uint16_t	Default set point after reset "Comfort" **Controller 2**	0...6553,5 = 0...65535	220
190	4x00BD	0x0000-0xFFFF	Uint16_t	Set point difference "Standby" **Controller 2**	0...6553,5 = 0...65535	0
191	4x00BE	0x0000-0xFFFF	Uint16_t	Set point difference " Unoccupied " **Controller 2**	0...6553,5 = 0...65535	0
192	4x00BF	0x0000-0xFFFF	Uint16_t	Controller parameter XP Heating **Controller 2**	0...6553,5 = 0...65535	20
193	4x00C0	0x0000-0xFFFF	Uint16_t	Controller parameter TN Heating **Controller 2**	0...6553,5 = 0...65535 [s]	1000
194	4x00C1	0x0000-0xFFFF	Uint16_t	Controller parameter XP Cooling **Controller 2**	0...6553,5 = 0...65535	20
195	4x00C2	0x0000-0xFFFF	Uint16_t	Controller parameter TN Cooling **Controller 2**	0...6553,5 = 0...65535 [s]	1000
196	4x00C3	0x0000-0x03E8	Uint16_t	Dead zone **Controller 2**	0...6553,5 = 0...65535	10
197	4x00C4	0x0000-0x03E8	Uint16_t	Minimum control variable Heating **Controller 2**	0...100,0% = 0...1000dez	0
198	4x00C5	0x0000-0x03E8	Uint16_t	Maximum control variable Heating **Controller 2**	0...100,0% = 0...1000dez	1000
199	4x00C6	0x0000-0x03E8	Uint16_t	Minimum control variable Cooling **Controller 2**	0...100,0% = 0...1000dez	0
200	4x00C7	0x0000-0x0003	Uint16_t	Maximum control variable Cooling **Controller 2**	0...100,0% = 0...1000dez	1000

201	4x00C8	0x0000-0x0002	Uint16_t	Controller mode after Reset **Controller 2**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
202	4x00C9	0x0000-0x0005	Uint16_t	Selection actual value **Controller 2**	0 = Input register *Actual value (Addr. 4x0429)* 1 = int. Temperature 2 = int. Humidity	0
203	4x00CA	0x0000-0x0001	Uint16_t	Selection Set point **Controller 2**	0 =Input register *Default Set point (Addr. 4x042D>* >=1= int. Set point 2	0
204	4x00CB	0x0000-0x0001	Uint16_t	Selection energy hold-off **Controller 2**	0 = Window contact via Input register *Energy hold-off (Addr. 4x042B)* 1 = Window contact via internal state 2 = Register and internal state OR-connected	0
205	4x00CC	0x0000-0xFFFF	Uint16_t	Selection Un-/Occupied **Controller 2**	0 = Data output via Input register *Occupancy (Addr. 4x042A)* 1 = Data output via internal status 2 = Register and internal state OR-connected	0
206	4x00CD	0x0000-0x0001	Uint16_t	Frost protection-"Set point" **Controller 2**	0 - 0...6553,5 = 0...65535 (0 deactivates frost protection)	0
207	4x00CE	0x0000-0x0003	Uint16_t	Minimal control variable **Controller 2**	1 = Control variable > 0 0 = Control variable = 0	0
208	4x00CF	0x0000-0x0003	Uint16_t	Controller has access to symbols "Heating&Cooling " **Controller 2**	0= No access 1= Access to Heating 2= Access to Cooling 3= Access to Heating&Cooling	0
209	4x00D0	0x0000-0x03E8	Uint16_t	Number Fan stages **Controller 2**	0 = none 1 = Stage 1 2 = Stage 2 3 = Stage 3	0
210	4x00D1	0x0000-0x03E8	Uint16_t	Control variable threshold for Fan stage 1 Heating **Controller 2**	0...100,0% = 0...1000dez	10
211	4x00D2	0x0000-0x03E8	Uint16_t	Control variable threshold for Fan stage 2 Heating **Controller 2**	0...100,0% = 0...1000dez	333
212	4x00D3	0x0000-0x03E8	Uint16_t	Control variable threshold for Fan stage 3 Heating **Controller 2**	0...100,0% = 0...1000dez	667
213	4x00D4	0x0000-0x03E8	Uint16_t	Control variable threshold for Fan stage 1 Cooling **Controller 2**	0...100,0% = 0...1000dez	10
214	4x00D5	0x0000-0x03E8	Uint16_t	Control variable threshold for Fan stage 2 Cooling **Controller 2**	0...100,0% = 0...1000dez	333
215	4x00D6	0x0000-0xFFFF	Uint16_t	Control variable threshold for Fan stage 3 Cooling **Controller 2**	0...100,0% = 0...1000dez	667
216	4x00D7	0x0000-0x0001	Uint16_t	PWM Cycle time **Controller 2**	0 = No PWM 1...65535 [s]	20
217	4x00D8	0x0000-0x0001	Uint16_t	Selection, if fan coil of controller X interacts with „Main"-Fan Controller **Controller 2**	0 = No access 1 = Access	0
218	4x00D9	0x0000-0xFFFF	Uint16_t	Display dew point signal of controller **Controller 2**	0 = hide 1 = show	0
219	4x00DA	0x0000-0xFFFF	Uint16_t	Bypass time **Controller 2**	0...6553,5 = 0...65535 [s]	10
220	4x00DB	0x0000-0xFFFF	Uint16_t	Minimal ON time for fan coil **Controller 2**	0...6553,5 = 0...65535 [s]	5
221	4x00DC	0x0000-0xFFFF	Uint16_t	Default Set point after Reset "Comfort" **Controller 3**	0...6553,5 = 0...65535	220
222	4x00DD	0x0000-0xFFFF	Uint16_t	Set point difference "Standby" **Controller 3**	0...6553,5 = 0...65535	0

223	4x00DE	0x0000-0xFFFF	Uint16_t	Set point difference " Unoccupied " **Controller 3**	0...6553,5 = 0...65535	0
224	4x00DF	0x0000-0xFFFF	Uint16_t	Controller parameter XP Heating **Controller**	0...6553,5 = 0...65535	20
225	4x00E0	0x0000-0xFFFF	Uint16_t	Controller parameter TN Heating **Controller 3**	0...6553,5 = 0...65535 [s]	1000
226	4x00E1	0x0000-0xFFFF	Uint16_t	Controller parameter XP Cooling **Controller 3**	0...6553,5 = 0...65535	20
227	4x00E2	0x0000-0xFFFF	Uint16_t	Controller parameter TN Cooling **Controller 3**	0...6553,5 = 0...65535 [s]	1000
228	4x00E3	0x0000-0x03E8	Uint16_t	Dead zone **Controller 3**	0...6553,5 = 0...65535	10
229	4x00E4	0x0000-0x03E8	Uint16_t	Minimum control variable Heating **Controller 3**	0...100,0% = 0...1000dez	0
230	4x00E5	0x0000-0x03E8	Uint16_t	Maximum control variable Heating **Controller 3**	0...100,0% = 0...1000dez	1000
231	4x00E6	0x0000-0x03E8	Uint16_t	Minimum control variable Cooling **Controller 3**	0...100,0% = 0...1000dez	0
232	4x00E7	0x0000-0x0003	Uint16_t	Maximum control variable Cooling **Controller 3**	0...100,0% = 0...1000dez	1000
233	4x00E8	0x0000-0x0002	Uint16_t	Controller mode after Reset **Controller 3**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
234	4x00E9	0x0000-0x0005	Uint16_t	Selection actual value **Controller 3**	0 = Input register *Actual value (Addr. 4x0431)* 1 = int. Temperature 2 = int. Humidity	0
235	4x000EA	0x0000-0x0001	Uint16_t	Selection set point **Controller 3**	0 =Input register *Default Set point (Addr. 4x0435*> >=1= int. Set point 3	0
236	4x00EB	0x0000-0x0001	Uint16_t	Selection energy hold-off **Controller 3**	0 = Window contact via Input register *Energy hold-off (Addr. 4x0433)* 1 = Window contact via internal state 2 = Register and internal status OR-connected	0
237	4x00EC	0x0000-0xFFFF	Uint16_t	Selection Un-/Occupied **Controller 3**	0 = Data output via Input register *Occupancy (Addr. 4x0432)* 1 = Data output via int. state 2 = Register and internal status OR-connected	0
238	4x00ED	0x0000-0x0001	Uint16_t	Frost protection-"Set point" **Controller 3**	0 - 0...6553,5 = 0...65535 (0 deactivates the frost protection)	0
239	4x00EE	0x0000-0x0003	Uint16_t	Minimum control variable **Controller 3**	1 = Control variable > 0 0 = Control variable = 0	0
240	4x00EF	0x0000-0x0003	Uint16_t	Controller has access to symbols "Heating&Cooling " **Controller 3**	0= No access 1= Access to Heating 2= Access to Cooling 3= Access to Heating&Cooling	0
241	4x00F0	0x0000-0x03E8	Uint16_t	Number Fan stage **Controller 3**	0 = none 1 = Stage 1 2 = Stage 2 3 = Stage 3	0
242	4x00F1	0x0000-0x03E8	Uint16_t	Control variable threshold for Fan stage 1 Heating **Controller 3**	0...100,0% = 0...1000dez	10
243	4x00F2	0x0000-0x03E8	Uint16_t	Control variable threshold for Fan stage 2 Heating **Controller 3**	0...100,0% = 0...1000dez	333
244	4x00F3	0x0000-0x03E8	Uint16_t	Control variable threshold for Fan stage 3 Heating **Controller 3**	0...100,0% = 0...1000dez	667

245	4x00F4	0x0000-0x03E8	Uint16_t	Control variable threshold for Fan stage 1 Cooling **Controller 3**	0...100,0% = 0...1000dez	10
246	4x00F5	0x0000-0x03E8	Uint16_t	Control variable threshold for Fan stage 2 Cooling **Controller 3**	0...100,0% = 0...1000dez	333
247	4x00F6	0x0000-0xFFFF	Uint16_t	Control variable threshold for Fan stage 3 Cooling **Controller 3**	0...100,0% = 0...1000dez	667
248	4x00F7	0x0000-0x0001	Uint16_t	PWM Cycle time **Controller 3**	0 = No PWM 1...65535 [s]	20
249	4x00F8	0x0000-0x0001	Uint16_t	Selection, if fan coil of controller X interacts with „Main"-Fan Controller **Controller 3**	0 = No access 1 = Access	0
250	4x00F9	0x0000-0xFFFF	Uint16_t	Display dew point signal of controller **Controller 3**	0 = hide 1 = show	0
251	4x00FA	0x0000-0xFFFF	Uint16_t	Bypass time **Controller 3**	0...6553,5 = 0...65535 [s]	10
252	4x00FB	0x0000-0xFFFF	Uint16_t	Minimum ON time for fan coils **Controller 3**	0...6553,5 = 0...65535 [s]	5
253	4x00FC	0x0000-0xFFFF	Uint16_t	Default s Set point after Reset "Comfort" **Controller 4**	0...6553,5 = 0...65535	220
254	4x00FD	0x0000-0xFFFF	Uint16_t	Set point difference "Standby" **Controller 4**	0...6553,5 = 0...65535	0
255	4x00FE	0x0000-0xFFFF	Uint16_t	Set point difference " Unoccupied " **Controller 4**	0...6553,5 = 0...65535	0
256	4x00FF	0x0000-0xFFFF	Uint16_t	Controller parameter XP Heating **Controller 4**	0...6553,5 = 0...65535	20
257	4x0100	0x0000-0xFFFF	Uint16_t	Controller parameter TN Heating **Controller 4**	0...6553,5 = 0...65535 [s]	1000
258	4x0101	0x0000-0xFFFF	Uint16_t	Controller parameter XP Cooling **Controller 4**	0...6553,5 = 0...65535	20
259	4x0102	0x0000-0xFFFF	Uint16_t	Controller parameter TN Cooling **Controller 4**	0...6553,5 = 0...65535 [s]	1000
260	4x0103	0x0000-0x03E8	Uint16_t	Dead zone **Controller 4**	0...6553,5 = 0...65535	10
261	4x0104	0x0000-0x03E8	Uint16_t	Minimum control variable Heating **Controller 4**	0...100,0% = 0...1000dez	0
262	4x0105	0x0000-0x03E8	Uint16_t	Maximum control variable Heating **Controller 4**	0...100,0% = 0...1000dez	1000
263	4x0106	0x0000-0x03E8	Uint16_t	Minimum control variable Cooling **Controller 4**	0...100,0% = 0...1000dez	0
264	4x0107	0x0000-0x0003	Uint16_t	Maximum control variable Cooling **Controller 4**	0...100,0% = 0...1000dez	1000
265	4x0108	0x0000-0x0002	Uint16_t	Controller mode after Reset **Controller 4**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
266	4x0109	0x0000-0x0005	Uint16_t	Selection Actual value **Controller 4**	0 = Input register *Actual value (Addr. 4x0439)* 1 = int. Temp. 2 = int. Humidity	0
267	4x010A	0x0000-0x0001	Uint16_t	Selection Set point **Controller 4**	0 =Input register *Default Set point (Addr. 4x043D>* >=1= int. Set point 4	0
268	4x010B	0x0000-0x0001	Uint16_t	Selection energy hold-off **Controller 4**	0 = Window contact via Input register *Energy hold-off (Addr. 4x043B)* 1 = Window contact via internal status 2 = Register and internal status OR-connected	0

269	4x010C	0x0000-0xFFFF	Uint16_t	Selection Un-/Occupied **Controller 4**	0 = Data output via Input register *Occupancy (Addr. 4x043A)* 1 = Data output via int. status 2 = Register and internal status OR-connected	0
270	4x010D	0x0000-0x0001	Uint16_t	Frost protection-"Set point" **Controller 4**	0 - 0...6553,5 = 0...65535 (0 deactivates the Frost protection)	0
271	4x010E	0x0000-0x0003	Uint16_t	Minimum control variable **Controller 4**	1 = Control variable > 0 0 = Control variable = 0	0
272	4x010F	0x0000-0x0003	Uint16_t	Controller has access to symbols "Heating&Cooling " **Controller 4**	0= No access 1=Access to Heating 2= Access to Cooling 3= Access to Heating & Cooling	0
273	4x0110	0x0000-0x03E8	Uint16_t	Number of Fan stages **Controller 4**	0 = None 1 = Stage 1 2 = Stage 2 3 = Stage 3	0
274	4x0111	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 1 Heating **Controller 4**	0...100,0% = 0...1000dez	10
275	4x0112	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 2 Heating **Controller 4**	0...100,0% = 0...1000dez	333
276	4x0113	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 3 Heating **Controller 4**	0...100,0% = 0...1000dez	667
277	4x0114	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 1 Cooling **Controller 4**	0...100,0% = 0...1000dez	10
278	4x0115	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 2 Cooling **Controller 4**	0...100,0% = 0...1000dez	333
279	4x0116	0x0000-0xFFFF	Uint16_t	control variable threshold for Fan stage 3 Cooling **Controller 4**	0...100,0% = 0...1000dez	667
280	4x0117	0x0000-0x0001	Uint16_t	PWM Cycle time **Controller 4**	0 = No PWM 1...65535 [s]	20
281	4x0118	0x0000-0x0001	Uint16_t	Selection, if fan coil of controller X interacts with „Main"- Fan Controller **Controller 4**	0 = No access 1 = Access	0
282	4x0119	0x0000-0xFFFF	Uint16_t	Display dew point signal of controller **Controller 4**	0 = hide 1 = show	0
283	4x011A	0x0000-0xFFFF	Uint16_t	Bypass time **Controller 4**	0...6553,5 = 0...65535 [s]	10
284	4x011B	0x0000-0xFFFF	Uint16_t	Minimum ON time for fan coils **Controller 4**	0...6553,5 = 0...65535 [s]	5
285	4x011C	0x0000-0xFFFF	Uint16_t	Default set point after Reset "Comfort" **Controller 5**	0...6553,5 = 0...65535	220
286	4x011D	0x0000-0xFFFF	Uint16_t	Set point difference "Standby" **Controller 5**	0...6553,5 = 0...65535	0
287	4x011E	0x0000-0xFFFF	Uint16_t	Set point difference " Unoccupied " **Controller 5**	0...6553,5 = 0...65535	0
288	4x011F	0x0000-0xFFFF	Uint16_t	Controller parameter XP Heating **Controller 5**	0...6553,5 = 0...65535	20
289	4x0120	0x0000-0xFFFF	Uint16_t	Controller parameter TN Heating **Controller 5**	0...6553,5 = 0...65535 [s]	1000
290	4x0121	0x0000-0xFFFF	Uint16_t	Controller parameter XP Cooling **Controller 5**	0...6553,5 = 0...65535	20
291	4x0122	0x0000-0xFFFF	Uint16_t	Controller parameter TN Cooling **Controller 5**	0...6553,5 = 0...65535 [s]	1000

292	4x0123	0x0000-0x03E8	Uint16_t	Dead zone **Controller 5**	0...6553,5 = 0...65535	10
293	4x0124	0x0000-0x03E8	Uint16_t	Minimum control variable Heating **Controller 5**	0...100,0% = 0...1000dez	0
294	4x0125	0x0000-0x03E8	Uint16_t	Maximum control variable Heating **Controller 5**	0...100,0% = 0...1000dez	1000
295	4x0126	0x0000-0x03E8	Uint16_t	Minimum control variable Cooling **Controller 5**	0...100,0% = 0...1000dez	0
296	4x0127	0x0000-0x0003	Uint16_t	Maximum control variable Cooling **Controller 5**	0...100,0% = 0...1000dez	1000
297	4x0128	0x0000-0x0002	Uint16_t	Controller mode after Reset **Controller 5**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
298	4x0129	0x0000-0x0005	Uint16_t	Selection Acutal value **Controller 5**	0 = Input register *Actual value (Addr. 4x0441)* 1 = int. Temp. 2 = int. Humidity	0
299	4x012A	0x0000-0x0001	Uint16_t	Selection Set point **Controller 5**	0 =Input register *Default Set point (Addr. 4x0445*> >=1= int. Set point 5	0
300	4x012B	0x0000-0x0001	Uint16_t	Selection energy hold-off **Controller 5**	0 = Window contact via Input register *Energy hold-off (Addr. 4x0443)* 1 = Window contact via internal status 2 = Register and internal status OR-connected	0
301	4x012C	0x0000-0xFFFF	Uint16_t	Selection Un-/Occupied **Controller 5**	0 = Data output via Input register *Occupancy (Addr. 4x0442)* 1 = Data output via int. status 2 = Register and internal status OR-connected	0
302	4x012D	0x0000-0x0001	Uint16_t	Frost protection-"Set point" **Controller 5**	0 - 0...6553,5 = 0...65535 (0 deactivates the Frost protection)	0
303	4x012E	0x0000-0x0003	Uint16_t	Minimum control variable **Controller 5**	1 = Control variable > 0 0 = Control variable = 0	0
304	4x012F	0x0000-0x0003	Uint16_t	Controller has access to symbols "Heating&Cooling " **Controller 5**	0 = No access 1=Access to Heating 2=Access to Cooling 3=Access to Heating & Cooling	0
305	4x0130	0x0000-0x03E8	Uint16_t	Number Fan stages **Controller 5**	0 = none 1 = Stage 1 2 = Stage 2 3 = Stage 3	0
306	4x0131	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 1 Heating **Controller 5**	0...100,0% = 0...1000dez	10
307	4x0132	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 2 Heating **Controller 5**	0...100,0% = 0...1000dez	333
308	4x0133	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 3 Heating **Controller 5**	0...100,0% = 0...1000dez	667
309	4x0134	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 1 Cooling **Controller 5**	0...100,0% = 0...1000dez	10
310	4x0135	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 2 Cooling **Controller 5**	0...100,0% = 0...1000dez	333
311	4x0136	0x0000-0xFFFF	Uint16_t	control variable threshold for Fan stage 3 Cooling **Controller 5**	0...100,0% = 0...1000dez	667
312	4x0137	0x0000-0x0001	Uint16_t	PWM Cycle time **Controller 5**	0 = No PWM 1...65535 [s]	20
313	4x0138	0x0000-0x0001	Uint16_t	Selection, if fan coil of controller X interacts with „Main“- Fan Controller **Controller 5**	0 = No Access 1 = Access	0

314	4x0139	0x0000-0xFFFF	Uint16_t	Display dew point signal of controller **Controller 5**	0 = hide 1 = show	0
315	4x013A	0x0000-0xFFFF	Uint16_t	Bypass time **Controller 5**	0...6553,5 = 0...65535 [s]	10
316	4x013B	0x0000-0xFFFF	Uint16_t	Minimum ON time for fan coils **Controller 5**	0...6553,5 = 0...65535 [s]	5
317	4x013C	0x0000-0xFFFF	Uint16_t	Default Set point after Reset "Comfort" **Controller 6**	0...6553,5 = 0...65535	220
318	4x013D	0x0000-0xFFFF	Uint16_t	Set point difference "Standby" **Controller 6**	0...6553,5 = 0...65535	0
319	4x013E	0x0000-0xFFFF	Uint16_t	Set point difference " Unoccupied " **Controller 6**	0...6553,5 = 0...65535	0
320	4x013F	0x0000-0xFFFF	Uint16_t	Controller parameter XP Heating **Controller 6**	0...6553,5 = 0...65535	20
321	4x0140	0x0000-0xFFFF	Uint16_t	Controller parameter TN Heating **Controller 6**	0...6553,5 = 0...65535 [s]	1000
322	4x0141	0x0000-0xFFFF	Uint16_t	Controller parameter XP Cooling **Controller 6**	0...6553,5 = 0...65535	20
323	4x0142	0x0000-0xFFFF	Uint16_t	Controller parameter TN Cooling **Controller 6**	0...6553,5 = 0...65535 [s]	1000
324	4x0143	0x0000-0x03E8	Uint16_t	Dead zone **Controller 6**	0...6553,5 = 0...65535	10
325	4x0144	0x0000-0x03E8	Uint16_t	Minimum control variable Heating **Controller 6**	0...100,0% = 0...1000dez	0
326	4x0145	0x0000-0x03E8	Uint16_t	Maximum control variable Heating **Controller 6**	0...100,0% = 0...1000dez	1000
327	4x0146	0x0000-0x03E8	Uint16_t	Minimum control variable Cooling **Controller 6**	0...100,0% = 0...1000dez	0
328	4x0147	0x0000-0x0003	Uint16_t	Maximum control variable Cooling **Controller 6**	0...100,0% = 0...1000dez	1000
329	4x0148	0x0000-0x0002	Uint16_t	Controller mode after Reset **Controller 6**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
330	4x0149	0x0000-0x0005	Uint16_t	Selection Actual value **Controller 6**	0 = Input register *Actual value (Addr. 04x0449)* 1 = int. Temp. 2 = int. Humidity	0
331	4x014A	0x0000-0x0001	Uint16_t	Selection Set point **Controller 6**	0 =Input register *Default Set point (Addr. 4x044D>* >=1= int. Set point 6	0
332	4x014B	0x0000-0x0001	Uint16_t	Selection energy hold-off **Controller 6**	0 = Window contact via Input register *Energy hold-off (Addr. 4x044B)* 1 = Window contact via internal status 2 = Register and internal status OR-connected	0
333	4x014C	0x0000-0xFFFF	Uint16_t	Selection Un-/Occupied **Controller 6**	0 = Data output via Input register *Occupancy (Addr. 4x044A)* 1 = Data output via int. status 2 = Register and internal status OR-connected	0
334	4x014D	0x0000-0x0001	Uint16_t	Frost protection-"Set point" **Controller 6**	0 - 0...6553,5 = 0...65535 (0 deactivates the Frost protection)	0
335	4x014E	0x0000-0x0003	Uint16_t	Minimum control variable **Controller 6**	1 = Control variable > 0 0 = Control variable = 0	0
336	4x014F	0x0000-0x0003	Uint16_t	Controller has access to symbols "Heating&Cooling " **Controller 6**	0= No Access 1= Access to Heating 2= Access to Cooling 3= Access to Heating & Cooling	0

337	4x0150	0x0000-0x03E8	Uint16_t	Number Fan stages **Controller 6**	0 = none 1 = Stage 1 2 = Stage 2 3 = Stage 3	0
338	4x0151	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 1 Heating **Controller 6**	0...100,0% = 0...1000dez	10
339	4x0152	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 2 Heating **Controller 6**	0...100,0% = 0...1000dez	333
340	4x0153	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 3 Heating **Controller 6**	0...100,0% = 0...1000dez	667
341	4x0154	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 1 Cooling **Controller 6**	0...100,0% = 0...1000dez	10
342	4x0155	0x0000-0x03E8	Uint16_t	control variable threshold for Fan stage 2 Cooling **Controller 6**	0...100,0% = 0...1000dez	333
343	4x0156	0x0000-0xFFFF	Uint16_t	control variable threshold for Fan stage 3 Cooling **Controller 6**	0...100,0% = 0...1000dez	667
344	4x0157	0x0000-0x0001	Uint16_t	PWM Cycle time **Controller 6**	0 = No PWM 1...65535 [s]	20
345	4x0158	0x0000-0x0001	Uint16_t	Selection, if fan coil of controller X interacts with „Main“-Fan Controller **Controller 6**	0 = No Access 1 = Access	0
346	4x0159	0x0000-0xFFFF	Uint16_t	Display dew point signal of controller **Controller 6**	0 = hide 1 = show	0
347	4x015A	0x0000-0xFFFF	Uint16_t	Bypass time **Controller 6**	0...6553,5 = 0...65535 [s]	10
348	4x015B	0x0000-0x111A	Uint16_t	Minimum ON time for fan coils **Controller 6**	0...6553,5 = 0...65535 [s]	5
349	4x015C	0x0000-0x111A	Uint16_t	Function digital Input 1	0...6553,5 = 0...65535 [s] 0x000, make contact 0x001, make contact, dew point controller 0x002, make contact, window contact 0x003, make contact, alarm 0x004, make contact, room occupation 0x005, make contact, show message text 0x006, make contact, enable controller 1 0x007, make contact, enable controller 2 0x008, make contact, enable controller 3 0x009, make contact, enable controller 4 0x00A, make contact, enable controller 5 0x00B, make contact, enable controller 6 **0-Auto/1-Off** 0x00C, make contact, mode controller 1 0x00D, make contact, mode controller 2 0x00E, make contact, mode controller 3 0x00F, make contact, mode controller 4 0x010, make contact, mode controller 5 0x011, make contact, mode controller 6 **0-Heating/1-Cooling** 0x012, make contact, edge counter 0x013, make contact, pulse counter 0x014, make contact, time 0x015, make contact, reset offset set point 1 0x016, make contact, reset offset set point 2	0 make con-tact
350	4x015D	0x0000-0x111A	Uint16_t	Function digital Input 2		

351	4x015E	0x0000-0x111A	Uint16_t	Function digital Input 3 **	0x017, make contact, reset offset set point 3 0x018, make contact, reset offset set point 4 0x019, make contact, reset offset set point 5 0x01A, make contact, reset offset set point 6 0x100, break contact 0x101, break contact, dew point controller 0x102, break contact, Window contact 0x103, break contact, Alarm 0x104, break contact, room occupation 0x105, break contact, Show message_text 0x106, break contact, enable controller 1, 0x107, break contact, enable controller 2, 0x108, break contact, enable controller 3, 0x109, break contact, enable controller 4, 0x10A, break contact, enable controller 5, 0x10B, break contact, enable controller 6, **0-Auto/1-Off**	
352	4x015F	0x0000-0x111A	Uint16_t	Function digital Input 4 **	0x10C, break contact, mode controller 1, 0x10D, break contact, mode controller 2, 0x10E, break contact, mode controller 3, 0x10F, break contact, mode controller 4, 0x110, break contact, mode controller 5, 0x111, break contact, mode controller 6, **0-Heating/1-Cooling** 0x112, break contact, edge counter 0x113, break contact, pulse counter 0x114, break contact, time 0x115, break contact, reset offset set point 1 0x116, break contact, reset offset set point 2 0x117, break contact, reset offset set point 3 0x118, break contact, reset offset set point 4 0x119, break contact, reset offset set point 5 0x11A, break contact, reset offset set point 6	
353	4x0160	0x0000-0x000A	Uint16_t	Volume button sound	0..100%	
354	4x0161	0x00A0-0x00AA	Uint16_t	Direct key 16	see chapter 5.4	
355	4x0162	0x00A0-0x00AA	Uint16_t	Direct key 17		
356	4x0163	0x00A0-0x00AA	Uint16_t	Direct key 18		
357	4x0164	0x00A0-0x00AA	Uint16_t	Direct key 19		
358	4x0165	0x00A0-0x00AA	Uint16_t	Direct key 20		

359	4x0166	0x00A0-0x00AA	Uint16_t	Direct key 21		
360	4x0167	0x00A0-0x00AA	Uint16_t	Direct key 22		
361	4x0168	0x00A0-0x00AA	Uint16_t	Direct key 23		
362	4x0169	0x00A0-0x00AA	Uint16_t	Direct key 24		
363	4x016A	0x00A0-0x00AA	Uint16_t	Direct key 25		
364	4x016B	0x00A0-0x00AA	Uint16_t	Direct key 26		
365	4x016C	0x00A0-0x00AA	Uint16_t	Direct key 27		
366	4x016D	0x00A0-0x00AA	Uint16_t	Direct key 28		
367	4x016E	0x00A0-0x00AA	Uint16_t	Direct key 29		
368	4x016F	0x00A0-0x00AA	Uint16_t	Direct key 30		
369	4x0170	0x00A0-0x00AA	Uint16_t	Direct key 31		
370	4x0171	0-1	Uint16_t	Standardscreen	0: Standardscreen = Submenu 1 1: Standardscreen = Temperature, Set point, Ext. value, … (type S / SQ only)	1
371	4x0172	0x0000-0xFFFF	Uint16_t	Parameterversion (read only)	Parameterversion	
372	4x0173	0-1	Uint16_t	FanCoil „OFF / AUTO only“	0: Normale Lüfterverstellung 1: Lüfterstufenwahl nur AUS oder AUTO	0

Tabelle 7-1 List if configuration registries

Messages (EEPROM)

Register	Data address	Range	description
513 – 577R/W	**0x0200 – 0x0241**	**Configuration parameter, non-volatile !! Don't update permanently EEprom !!**	

BS 1-14 = ASCII character

R 513		R 514		R 515		R 516		R 517		R 518		R 519		R520	
Hi	Lo	Hi	Lo	Hi	Lo	Hi	Lo	Hi	Lo	Hi	Lo	Hi	Lo	Hi	Lo
S	e	t	p	o	i	n	t		l						
0x53	0x65	0x74	0x70	0x6F	0x69	0x6E	0x74	0x20	0x31	0x20	0x20	0x20	0x20	0x20	0x20

Example: Set point 1

- Input of letters and numbers in ASCII format
- Fade-in with input register 0x0209, „Show message"

513 R/W -519	0x0200	0x0000-0xFFFF	Message 1	Register 513		Register 514		…		Register 520	
				High	Low	High	Low			High	Low
				BS 1	BS 2	BS 3	BS 4			BS 13	BS 14
520 R/W -526	0x0207	0x0000-0xFFFF	Message 2	Register 521		Register 522		…		Register 528	
				High	Low	High	Low			High	Low
				BS 1	BS 2	BS 3	BS 4			BS 13	BS 14
527 R/W -533	0x020E	0x0000-0xFFFF	Message 3	Register 529		Register 530		…		Register 536	
				High	Low	High	Low			High	Low
				BS 1	BS 2	BS 3	BS 4			BS 13	BS 14
534 R/W -540	0x0215	0x0000-0xFFFF	Message 4	Register 537		Register 538		…		Register 544	
				High	Low	High	Low			High	Low
				BS 1	BS 2	BS 3	BS 4			BS 13	BS 14
541 R/W -547	0x021C	0x0000-0xFFFF	Message 5	Register 545		Register 546		…		Register 552	
				High	Low	High	Low			High	Low
				BS 1	BS 2	BS 3	BS 4			BS 13	BS 14
548 R/W -554	0x0223	0x0000-0xFFFF	Message 6	Register 553		Register 554		…		Register 560	
				High	Low	High	Low			High	Low
				BS 1	BS 2	BS 3	BS 4			BS 13	BS 14
555 R/W -561	0x022A	0x0000-0xFFFF	Message 7	Register 561		Register 562		…		Register 568	
				High	Low	High	Low			High	Low
				BS 1	BS 2	BS 3	BS 4			BS 13	BS 14
562 R/W -568	0x0231	0x0000-0xFFFF	Message 8	Register 769		Register 770		…		Register 775	
				High	Low	High	Low			High	Low
				BS 1	BS 2	BS 3	BS 4			BS 13	BS 14

Table 7-2 Overview message registers

Configuration bits (EEPROM!)				
Coil	**Data address**	**Type**	**Range**	**Default**
1 – 53 R/W	**0x0000– 0x004F**	**Configuration parameter !! Don't update permanently EEprom !!**		
1	0x0000	Display temperature	0 = off 1 = on	1
2	0x0001	Display humidity	0 = off 1 = on	0
3	0x0002	Resolution temperature	0 = no decimal place 1 = decimal place	1
4	0x0003	Resolution humidity	0 = no decimal place 1 = decimal place	1
5	0x0004	Activate device first	0 = on 1 = off	0
6	0x0005	Unit temperature	0 = °F 1 = °C	1
7	0x0006	Fan stage after reset	0 = off 1 = on	1
8	0x0007	Display room occupancy after reset		1
9	0x0008	Show menu line 1		0
10	0x0009	Show menu line 2		0
11	0x000A	Show menu line 3		0
12	0x000B	Show menu line 4		0
13	0x000C	Show menu line 5		0
14	0x000D	Display stage in auto mode		1
15	0x000E	**External value 1:** Resolution	0 = no decimal place 1 = decimal place	0
16	0x000F	**External value 1:** Show	0 = off 1 = on	0
17	0x0010	**External value 2:** Resolution	0 = no decimal place 1 = decimal place	0
18	0x0011	**External value 2:** Show	0 = off 1 = on	0
19	0x0012	**External value 3:** Resolution	0 = no decimal place 1 = decimal place	0
20	0x0013	**External value 3:** Show	0 = off 1 = on	0
21	0x0014	**External value 4:** Resolution	0 = no decimal place 1 = decimal place	0
22	0x0015	**External value 4:** Show	0 = off 1 = on	0
23	0x0016	**External value 5:** Resolution	0 = no decimal place 1 = decimal place	0
24	0x0017	**External value 5:** Show	0 = off 1 = on	0
25	0x0018	**External value 6:** Resolution	0 = no decimal place 1 = decimal place	0

26	0x0019	**External value 6** Show	0 = off 1 = on	0
27	0x001A	**Set point 1:** Resolution	0 = no decimal place 1 = decimal place	1
28	0x001B	**Set point 1:** Display with adjustment	0 = offset 1 = effective	1
29	0x001C	**Set point 1:** Display effective value	0 = off 1 = on	0
30	0x001D	**Set point 1:** Display offset value	0 = off 1 = on	0
31	0x001E	**Set point 2:** Resolution	0 = no decimal place 1 = decimal place	1
32	0x001F	**Set point 2:** Display with adjustment	0 = offset 1 = effective	1
33	0x0020	**Set point 2:** Display effective value	0 = off 1 = on	0
34	0x0021	**Set point 2:** Display offset value	0 = off 1 = on	0
35	0x0022	**Set point 3:** Resolution	0 = no decimal place 1 = decimal place	1
36	0x0023	**Set point 3:** Display with adjustment	0 = offset 1 = effective	1
37	0x0024	**Set point 3:** Display effective value	0 = off 1 = on	0
38	0x0025	**Set point 3:** Display offset value	0 = off 1 = on	0
39	0x0026	**Set point 4:** Resolution	0 = no decimal place 1 = decimal place	1
40	0x0027	**Set point 4:** Display with adjustment	0 = offset 1 = effective	1
41	0x0028	**Setpoint 4:** Display effective value	0 = off 1 = on	0
42	0x0029	**Set point 4:** Display offset value	0 = off 1 = on	0
43	0x002A	**Set point 5:** Resolution	0 = no decimal place 1 = decimal place	1
44	0x002B	**Setpoint 5:** Display with adjustment	0 = offset 1 = effective	1
45	0x002C	**Set point 5:** Display effective value	0 = off 1 = on	0
46	0x002D	**Set point 5:** Display offset value	0 = off 1 = on	0
47	0x002E	**Setpoint 6:** Resolution	0 = no decimal place 1 = decimal place	1
48	0x002F	**Set point 6:** Display with adjustment	0 = offset 1 = effective	1
49	0x0030	**Set point 6:** Display effective value	0 = off 1 = on	0
50	0x0031	**Set point 6:** Display offset value	0 = off 1 = on	0
51	0x0032	Selection if indices will be shown in the LCD.	0=no 1=yes	1
52	0x0033	Selection if indices will be shown even if their value is 0.	0=no 1=yes	0

Table 7-3 Configuration bits listing

7.2 Data points

Output register (Input Register, Read only!)					
Register	**Modbus Data address**	**Range**	**Type**	**Description**	
769–841R	**0x0300–0x0348**				
769	3x0300	0x0000-0xFFFF	Uint16_t	Current state of key 1...16	Bit0=1->Key1 pressed Bit1=1->Key2 pressed ...
770	3x0301	0x0000-0xFFFF	Uint16_t	State of key since last call-off 1...16	Bit0=1-> Key1 pressed Bit1=1-> Key2 pressed ...
771	3x0302	0x0000-0xFFFF	Uint16_t	State light function	See chapter 5.4
772	3x0303	0x0000-0xFFFF	Uint16_t	Current state of „+“ dimming key	See chapter 5.4
773	3x0304	0x0000-0xFFFF -	Uint16_t	Current state of „-“ dimming key	See chapter 5.4
774	3x0305	0x0000-0xFFFF	Uint16_t	State shutter/blind function	See chapter 5.4
775	3x0306	0x0000-0xFFFF	Uint16_t	Current state of „+“ Shutter/blind key	See chapter 5.4
776	3x0307	0x0000-0xFFFF	Uint16_t	Current state of „-“ Shutter/blind key	See chapter 5.4
777	3x0308	0x0000-0xFFFF	int16_t	State Universal function	See chapter 5.4
778	3x0309	0x0000-0xFFFF	int16_t	Current Offset for set point 1	Bsp.: 50 = 5,0°C
779	3x030A	0x0000-0xFFFF	int16_t	Set point effective 1 (Default set point + set point offset)	Bsp.: 250 = 25,0°C
780	3x030B	0x0000-0xFFFF	int16_t	Current Offset for set point 2	Bsp.: 50 = 5,0°C
781	3x030C	0x0000-0xFFFF	int16_t	Set point effective 2 (Default set point + set point offset)	Bsp.: 250 = 25,0°C
782	3x030D	0x0000-0xFFFF	int16_t	Current Offset for set point 3	Bsp.: 50 = 5,0°C
783	3x030E	0x0000-0xFFFF	int16_t	Set point effective 3 (Default set point + set point offset)	Bsp.: 250 = 25,0°C
784	3x030F	0x0000-0xFFFF	int16_t	Current Offset for set point 4	Bsp.: 50 = 5,0°C
785	3x0310	0x0000-0xFFFF	int16_t	Set point effective 4 (Default set point + set point offset)	Bsp.: 250 = 25,0°C
786	3x0311	0x0000-0xFFFF	int16_t	Current Offset for set point 5	Bsp.: 50 = 5,0°C
787	3x0312	0x0000-0xFFFF	int16_t	Set point effective 5 (Default set point + set point offset)	Bsp.: 250 = 25,0°C
788	3x0313	0x0000-0xFFFF	int16_t	Current Offset for set point 6	Bsp.: 50 = 5,0°C
789	3x0314	0x0000-0xFFFF	int16_t	Set point effective 6 (Default set point + set point offset)	Bsp.: 250 = 25,0°C
790	3x0315	0x0000-0xFFFF	int16_t	Temperature internal sensor or external default value + Offset (Register 0x0006: Temperature Offset).	Bsp.: 210 = 21,0°C
791	3x0316	0x0000-0xFFFF	Uint16_t	Humidity internal sensor or external default value + Offset (Register 0x0007: Humidity Offset)	Bsp.: 500 = 50,0%rF

792	3x0317	0x0000-0xFFFF	Uint16_t	Current fan stage	0x0000 = OFF 0x0001 = Stage1 0x0002=Stage2 0x0003 = Stage3 0xFF00 = Auto OFF 0xFF01 = Auto Stage1 0xFF02=Auto Stage2 0xFF03=Auto Stage3
793	3x0318	0x0000-0x0001	Uint16_t	Current room occupancy	0=unoccupied 1=occupied
794	3x0319	0x0000-0x03E8	Uint16_t	Control variable Heating **Controller 1**	0...100,0% = 0...1000dez
795	3x031A	0x0000-0x03E8	Uint16_t	Control variable Cooling **Controller 1**	0...100,0% = 0...1000dez
796	3x031B	0x0000-0x0001	Uint16_t	Output register PWM-signal Heating **Controller 1**	0 = OFF 1 = ON
797	3x031C	0x0000-0x0001	Uint16_t	Output register PWM-signal Cooling **Controller 1**	0 = OFF 1 = An
798	3x031D	0x0000-0x0004	Uint16_t	Output controller mode **Controller 1**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto Heating 4 = Auto Cooling
799	3x031E	0x0000-0x003	Uint16_t	Output register fan stage **Controller 1**	0 = OFF 1 = Stage 1 2 = Stage 2 3 = Stage 3
800	3x031F	0x0000-0xFFFF	Uint16_t	Effective controller set point **Controller1**	0...6553,5 = 0...65535
801	3x0320	0x0000-0x03E8	Uint16_t	Control variable Heating **Controller 2**	0...100,0% = 0...1000dez
802	3x0321	0x0000-0x03E8	Uint16_t	Control variable Cooling **Controller 2**	0...100,0% = 0...1000dez
803	3x0322	0x0000-0x0001	Uint16_t	Output register PWM-Signal Heating **Controller 2**	0 = OFF 1 = ON
804	3x0323	0x0000-0x0001	Uint16_t	Output register PWM-Signal Cooling **Controller 2**	0 = OFF 1 = ON
805	3x0324	0x0000-0x0004	Uint16_t	Output Controller mode **Controller 2**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto Heating 4 = Auto Cooling
806	3x0325	0x0000-0x003	Uint16_t	Output register Fan Stage **Controller 2**	0 = OFF 1 = Stage 1 2 = Stage 2 3 = Stage 3
807	3x0326	0x0000-0xFFFF	Uint16_t	Effective controller set point **Controller2**	0...6553,5 = 0...65535
808	3x0327	0x0000-0x03E8	Uint16_t	Control variable Heating **Controller 3**	0...100,0% = 0...1000dez
809	3x0328	0x0000-0x03E8	Uint16_t	Control variable Cooling **Controller 3**	0...100,0% = 0...1000dez
810	3x0329	0x0000-0x0001	Uint16_t	Output register PWM-Signal Heating **Controller 3**	0 = OFF 1 = ON
811	3x032A	0x0000-0x0001	Uint16_t	Output register PWM-Signal Cooling **Controller 3**	0 = OFF 1 = ON
812	3x032B	0x0000-0x0004	Uint16_t	Output Controller mode **Controller 3**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto Heating 4 = Auto Cooling
813	3x032C	0x0000-0x003	Uint16_t	Output register Fan Stage **Controller 3**	0 = OFF 1 = Stage 1

					2 = Stage 2 3 = Stage 3
814	3x032D	0x0000- 0xFFFF	Uint16_t	Effective controller set point **Controller3**	0...6553,5 = 0...65535
815	3x032E	0x0000- 0x03E8	Uint16_t	Control variable Heating **Controller 4**	0...100,0% = 0...1000dez
816	3x032F	0x0000- 0x03E8	Uint16_t	Control variable Cooling **Controller 4**	0...100,0% = 0...1000dez
817	3x0330	0x0000- 0x0001	Uint16_t	Output register PWM-Signal Heating **Controller 4**	0 = OFF 1 = ON
818	3x0331	0x0000- 0x0001	Uint16_t	Output register PWM-Signal Cooling **Controller 4**	0 = OFF 1 = ON
819	3x0332	0x0000- 0x0004	Uint16_t	Output Controller mode **Controller 4**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto Heating 4 = Auto Cooling
820	3x0333	0x0000- 0x003	Uint16_t	Output register Fan Stage **Controller 4**	0 = OFF 1 = Stage 1 2 = Stage 2 3 = Stage 3
821	3x0334	0x0000- 0xFFFF	Uint16_t	Effective controller set point **Controller4**	0...6553,5 = 0...65535
822	3x0335	0x0000- 0x03E8	Uint16_t	Control variable Heating **Controller 5**	0...100,0% = 0...1000dez
823	3x0336	0x0000- 0x03E8	Uint16_t	Control variable Cooling **Controller 5**	0...100,0% = 0...1000dez
824	3x0337	0x0000- 0x0001	Uint16_t	Output register PWM-Signal Heating **Controller 5**	0 = OFF 1 = ON
825	3x0338	0x0000- 0x0001	Uint16_t	Output register PWM-Signal Cooling **Controller 5**	0 = OFF 1 = ON
826	3x0339	0x0000- 0x0004	Uint16_t	Output Controller mode **Controller 5**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto Heating 4 = Auto Cooling
827	3x033A	0x0000- 0x003	Uint16_t	Output register Fan Stage **Controller 5**	0 = OFF 1 = Stage 1 2 = Stage 2 3 = Stage 3
828	3x033B	0x0000- 0xFFFF	Uint16_t	Effective controller set point **Controller5**	0...6553,5 = 0...65535
829	3x033C	0x0000- 0x03E8	Uint16_t	Control variable Heating **Controller 6**	0...100,0% = 0...1000dez
830	3x033D	0x0000- 0x03E8	Uint16_t	Control variable Cooling **Controller 6**	0...100,0% = 0...1000dez 0 = OFF 1 = Stage 1 2 = Stage 2 3 = Stage 3
831	3x033E	0x0000- 0x0001	Uint16_t	Output register PWM-Signal Heating **Controller 6**	0 = OFF 1 = ON
832	3x033F	0x0000- 0x0001	Uint16_t	Output register PWM-Signal Cooling **Controller 6**	0 = OFF 1 = ON
833	0x0340	0x0000- 0x0004	Uint16_t	Output Controller mode **Controller 6**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto Heating 4 = Auto Cooling
834	0x0341	0x0000- 0x003	Uint16_t	Output register Fan Stage **Controller 6**	0 = OFF 1 = Stage 1

					2 = Stage 2 3 = Stage 3
835	0x0342	0x0000- 0xFFFF	Uint16_t	Effective controller set point **Controller6**	0...6553,5 = 0...65535
836	0x0343	0x0000- 0x0001	Uint16_t	Digital Input1	Representation depends on parameterization As a signal: 0- Open 1-Closed As a counter: 0-65535 (flanks, Impulses Time/[s])
837	0x0344	0x0000- 0x0001	Uint16_t	Digital Input2	
838	0x0345	0x0000- 0x0001	Uint16_t	Digital Input3	
839	0x0346	0x0000- 0x0001	Uint16_t	Digital Input4	
840	3x0347	0x0000- 0xFFFF	Uint16_t	Current state of key 16...31	Bit0=1->Key16 pressed Bit1=1->Key17 pressed ...
841	3x0348	0x0000- 0xFFFF	Uint16_t	State of key since last call-off 16...31	Bit0=1-> Key16 pressed Bit1=1-> Key17 pressed ...

Table 7-4 List of Output register

Input register (Holding register, Read and Write)						
Register	**Data address**	**Range**	**Type**	**Description**		**Default**
1025–1107 R/W	**4x0400–4x0450**					
1025	4x0400	0x0000-0x003B	Uint16_t	Input seconds	0 – 59 [s]	0
1026	4x0401	0x0000-0x003B	Uint16_t	Input minutes	0 – 59 [min]	0
1027	4x0402	0x0000-0x0017	Uint16_t	Input hours	0 – 23 [h]	0
1028	4x0403	0x0001-0x0006	Uint16_t	Input day of month	1 – 31	1
1029	4x0404	0x0001-0x0006	Uint16_t	Input month	1 – 12	1
1030	4x0405	0x0000-0xFFFF	Uint16_t	Input year		2000
1031	4x0406	0x0000-0x0003 0xFF00-0xFF03	Uint16_t	Default fan stage	0x0000 = OFF 0x0001 = Stage1 0x0002 = Stage2 0x0003 = Stage3 0xFF00 = Auto OFF 0xFF01 = Auto Stage1 0xFF02 = Auto Stage2 0xFF03 = Auto Stage3	0
1032	4x0407	0x0000-0x0001	Uint16_t	Default occupancy	0 - unoccupied 1 – occupied 2 - Standby	0 unoccupied
1033	4x0408	0x0000-0xFFFF	Uint16_t	Bypass time re-trigger	0-65535 [s]	0
1034	4x0409	0x0000-0x0008	Uint16_t	Show message	0 - no message 1..8 - message 1 - 8	0 No message
1035	4x040A	0x0000-0xFFFF	Int16_t	External temperature default	z.B. 170_{dez} = 17.0°C internal sensor: 0x7FFF/32767	0x7FFF
1036	4x040B	0x0000-0x03E8	Int16_t	External humidity default	z.B. 1000_{dez} = 100.0% internal sensor: 0x7FFF/32767	0x7FFF
1037	4x040C	0x0000-0xFFFF	Uint16_t	Feedback status light function	See chapter 5.4	0
1038	4x040D	0x0000-0xFFFF	Uint16_t	Feedback shutter/blind function	See chapter 5.4	0
1039	4x040E	0x0000-0xFFFF	int16_t	Feedback universal function	See chapter 5.4	0
1040	4x040F	0x0000-0xFFFF	Int16_t	External value1	z.B. 0x00E6 = 230 z.B. 0x000A = 10 z.B. 0xFF38 = -200 0...6553,5 = 0...65535	0
1041	4x0410	0x0000-0xFFFF	Int16_t	External value2		0
1042	4x0411	0x0000-0xFFFF	Int16_t	External value3		0
1043	4x0412	0x0000-0xFFFF	Int16_t	External value4		0
1044	4x0413	0x0000-0xFFFF	Int16_t	External value5		0
1045	4x0414	0x0000-0xFFFF	Int16_t	External value6		0
1046	4x0415	0x0000-0xFFFF	Int16_t	Offset set point1		0

1047	4x0416	0x0000-0xFFFF	Int16_t	Standard set point1		0
1048	4x0417	0x0000-0xFFFF	Int16_t	Offset set point2		0
1049	4x0418	0x0000-0xFFFF	Int16_t	Basis set point2		0
1050	4x0419	0x0000-0xFFFF	Int16_t	Offset set point3		0
1051	4x041A	0x0000-0xFFFF	Int16_t	Basis set point3		0
1052	4x041B	0x0000-0xFFFF	Int16_t	Offset set point4		0
1053	4x041C	0x0000-0xFFFF	Int16_t	Basis set point4		0
1054	4x041D	0x0000-0xFFFF	Int16_t	Offset set point5		0
1055	4x041E	0x0000-0xFFFF	Int16_t	Basis set point5		0
1056	4x041F	0x0000-0xFFFF	Int16_t	Offset set point6		0
1057	4x0420	0x0000-0x0002	Int16_t	Basis set point6		0
1058	4x0421	0x0000-0xFFFF	Uint16_t	Actual value **Controller 1**	0...6553,5 = 0...65535	210
1059	4x0422	0x0000-0x0002	Uint16_t	Occupancy **Controller 1**	0 = unoccupied 1 = occupied 2 = Standby	0
1060	4x0423	0x0000-0x0001	Uint16_t	Energy hold-off **Controller 1**	0 = deactivated 1 = activated	0
1061	4x0424	0x0000-0x0003	Uint16_t	Controller mode **Controller 1**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
1062	4x0425	0x0000-0xFFFF	Uint16_t	Base set point **Controller 1**	0...6553,5 = 0...65535	220
1063	4x0426	0x0000-0x0001	Uint16_t	Dew point **Controller 1**	0 = inactive 1 = active	0
1064	4x0427	0x0000-0xFFFF	Uint16_t	Trigger bypass time **Controller 1**	0...6553,5 = 0...65535 [s]	0
1065	4x0428	0x0000-0xFFFF	Int16_t	Default set point offset **Controller 1**	-3276,6 – 3276,7 K	0
1066	4x0429	0x0000-0xFFFF	Uint16_t	Actual value **Controller 2**	0...6553,5 = 0...65535	210
1067	4x042A	0x0000-0x0002	Uint16_t	Occupancy **Controller 2**	0 = unoccupied 1 = occupied 2 = Standby	0
1068	4x042B	0x0000-0x0001	Uint16_t	Energy hold-off **Controller 2**	0 = deactivated 1 = activated	0
1069	4x042C	0x0000-0x0003	Uint16_t	Controller mode **Controller 2**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
1070	4x042D	0x0000-0xFFFF	Uint16_t	Base set point **Controller 2**	0...6553,5 = 0...65535	220
1071	4x042E	0x0000-0x0001	Uint16_t	Dew point **Controller 2**	0 = inactive 1 = active	0
1072	4x042F	0x0000-0xFFFF	Uint16_t	Trigger bypass time **Controller 2**	0...6553,5 = 0...65535 [s]	0

1073	4x0430	0x0000-0xFFFF	Int16_t	Default set point offset **Controller 2**	-3276,6 – 3276,7 K	0
1074	4x0431	0x0000-0xFFFF	Uint16_t	Actual value **Controller 3**	0...6553,5 = 0...65535	210
1075	4x0432	0x0000-0x0002	Uint16_t	Occupancy **Controller 3**	0 = unoccupied 1 = occupied 2 = Standby	0
1076	4x0433	0x0000-0x0001	Uint16_t	Energy hold-off **Controller 3**	0 = deactivated 1 = activated	0
1077	4x0434	0x0000-0x0003	Uint16_t	Controller mode **Controller 3**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
1078	4x0435	0x0000-0xFFFF	Uint16_t	Base set point **Controller 3**	0...6553,5 = 0...65535	220
1079	4x0436	0x0000-0x0001	Uint16_t	Dew point **Controller 3**	0 = inactive 1 = active	0
1080	4x0437	0x0000-0xFFFF	Uint16_t	Trigger bypass time **Controller 3**	0...6553,5 = 0...65535 [s]	0
1081	4x0438	0x0000-0xFFFF	Int16_t	Default set point offset **Controller 3**	-3276,6 – 3276,7 K	0
1082	4x0439	0x0000-0xFFFF	Uint16_t	Actual value **Controller 4**	0...6553,5 = 0...65535	210
1083	4x043A	0x0000-0x0002	Uint16_t	Occupancy **Controller 4**	0 = unoccupied 1 = occupied 2 = Standby	0
1084	4x043B	0x0000-0x0001	Uint16_t	Energy hold-off **Controller 4**	0 = deactivated 1 = activated	0
1085	4x043C	0x0000-0x0003	Uint16_t	Controller mode **Controller 4**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
1086	4x043D	0x0000-0xFFFF	Uint16_t	Standard set point **Controller 4**	0...6553,5 = 0...65535	220
1087	4x043E	0x0000-0x0001	Uint16_t	Dew point **Controller 4**	0 = inactive 1 = active	0
1088	4x043F	0x0000-0xFFFF	Uint16_t	Trigger bypass time **Controller 4**	0...6553,5 = 0...65535 [s]	0
1089	4x0440	0x0000-0xFFFF	Int16_t	Default set point offset **Controller 4**	-3276,6 – 3276,7 K	0
1090	4x0441	0x0000-0xFFFF	Uint16_t	Actual value **Controller 5**	0...6553,5 = 0...65535	210
1091	4x0442	0x0000-0x0002	Uint16_t	Occupancy **Controller 5**	0 = unoccupied 1 = occupied 2 = Standby	0
1092	4x0443	0x0000-0x0001	Uint16_t	Energy hold-off **Controller 5**	0 = deactivated 1 = activated	0
1093	4x0444	0x0000-0x0003	Uint16_t	Controller mode **Controller 5**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
1094	4x0445	0x0000-0xFFFF	Uint16_t	Base set point **Controller 5**	0...6553,5 = 0...65535	220
1095	4x0446	0x0000-0x0001	Uint16_t	Dew point **Controller 5**	0 = inactive 1 = active	0
1096	4x0447	0x0000-0xFFFF	Uint16_t	Trigger bypass time **Controller 5**	0...6553,5 = 0...65535 [s]	0
1097	4x0448	0x0000-0xFFFF	Int16_t	Default set point offset **Controller 5**	-3276,6 – 3276,7 K	0

1098	4x0449	0x0000-0xFFFF	Uint16_t	Actual value **Controller 6**	0...6553,5 = 0...65535	210
1099	4x044A	0x0000-0x0002	Uint16_t	Occupancy **Controller 6**	0 = unoccupied 1 = occupied 2 = Standby	0
1100	4x044B	0x0000-0x0001	Uint16_t	Energy hold-off **Controller 6**	0 = deactivated 1 = activated	0
1101	4x044C	0x0000-0x0003	Uint16_t	Controller mode **Controller 6**	0 = OFF 1 = Heating 2 = Cooling 3 = Auto	3
1102	4x044D	0x0000-0xFFFF	Uint16_t	Base set point **Controller 6**	0...6553,5 = 0...65535	220
1103	4x044E	0x0000-0x0001	Uint16_t	Dew point **Controller 6**	0 = inactive 1 = active	0
1104	4x044F	0x0000-0xFFFF	Uint16_t	Trigger bypass time **Controller 6**	0...6553,5 = 0...65535 [s]	0
1105	4x0450	0x0000-0xFFFF	Int16_t	Default set point offset **Controller 6**	-3276,6 – 3276,7 K	0
1106	4x0451	0x0000-0xFFFF	Uint16_t	Restart 1	The device will restart, if the registers contain the values given below: „Restart 1“ ⇒ 0x73A5 „Restart 2“ ⇒ 0xC93A	0
1107	4x0452	0x0000-0xFFFF	Uint16_t	Restart 2		0

Table 7-5 holding register listing

Input bits (Read/Write)				
Bit	**Data-address**	**Description**	**Range**	**Default**
257 – 266 R/ W	**0x0100– 0x0109**			
257	0x0100	Symbol failure	0 = OFF 1 = show symbol	0 OFF
258	0x0101	Symbol heating	0 = OFF 1 = show symbol	0 OFF
259	0x0102	Symbol cooling	0 = OFF 1 = show symbol	0 OFF
260	0x0103	Symbol window	0 = OFF 1 = show symbol	0 OFF
261	0x0104	Symbol dew point	0 = OFF 1 = show symbol	0 OFF
262	0x0105	Key lock	0 = unlocked 1 = lock	0 unlock
263	0x0106	Lock room occupancy	0 = unlock 1 = lock	0 unlock
264	0x0107	Lock fan stages	0 = unlock 1 = lock	0 unlock
265	0x0108	Lock set points	0 = unlock 1 = lock	0 unlock
266	0x0109	Activation of TFT- and key- illumination	0 = illumination according to status 1 = Illumination standard	0 illumination according to status

Table 7-6 Bits listing

8 EnOcean Gateway

The EnOcean ↔ ModBus gateway is only available on „thanos SR … Modbus …“!

8.1 EnOcean Receiving Registers

EnOcean Receiving Registers							
Register	**Data Address**	**Value Range**	**Data Type**	**Access**	**Description**		**Default**
1281-1543	**4x0500-4x0606**						
1281	4x0500	0x0000-0x0014	Uint16_t	R/W	Learn channel	0 = Learn mode disabled 1…20 = Learn mode enabled for channel 1…20	0
1282	4x0501	0x0000-0xFFFF	Uint16_t	R	Receiving flag – new data at channel 1…16	Bit0–Channel 1 … Bit15–Channel 16 0 = No new data 1 = New data	0
1283	4x0502	0x0000-0x000F	Uint16_t	R	Receiving flag – new data at channel 17…20	Bit0–Channel 17 … Bit3–Channel 20	0
1284	4x0503	0x0000-0x00FF	Uint16_t	R/W	ORG	Data Sensor 1	0
1285	4x0504	0x0000-0x00FF	Uint16_t	R/W	TYPE		0
1286	4x0505	0x0000-0x00FF	Uint16_t	R/W	FUNC		0
1287	4x0506	0x0000-0x00FF	Uint16_t	R	Data-Byte 3		0
1288	4x0507	0x0000-0x00FF	Uint16_t	R	Data-Byte 2		0
1289	4x0508	0x0000-0x00FF	Uint16_t	R	Data-Byte 1		0
1290	4x0509	0x0000-0x00FF	Uint16_t	R	Data-Byte 0		0
1291	4x050A	0x0000-0x00FF	Uint16_t	R/W	ID-Byte 3		0
1292	4x050B	0x0000-0x00FF	Uint16_t	R/W	ID-Byte 2		0
1293	4x050C	0x0000-0x00FF	Uint16_t	R/W	ID-Byte 1		0
1294	4x050D	0x0000-0x00FF	Uint16_t	R/W	ID-Byte 0		0
1295	4x050E	0x0000-0xFFFF	Uint16_t	R	Receiving Time [s]		0
1296	4x050F	0x0000-0x000A	Uint16_t	R/W	Response Channel		0
. . . .							
1531	4x05FA	0x0000-0x00FF	Uint16_t	R/W	ORG	Data Sensor 20	0
1532	4x05FB	0x0000-0x00FF	Uint16_t	R/W	TYPE		0
1533	4x05FC	0x0000-0x00FF	Uint16_t	R/W	FUNC		0
1534	4x05FD	0x0000-0x00FF	Uint16_t	R	Data-Byte 3		0
1535	4x05FE	0x0000-	Uint16_t	R	Data-Byte 2		0

		0x00FF					
1536	4x05FF	0x0000-0x00FF	Uint16_t	R	Data-Byte 1		0
1537	4x0600	0x0000-0x00FF	Uint16_t	R	Data-Byte 0		0
1538	4x0601	0x0000-0x00FF	Uint16_t	R/W	ID-Byte 3		0
1539	4x0602	0x0000-0x00FF	Uint16_t	R/W	ID-Byte 2		0
1540	4x0603	0x0000-0x00FF	Uint16_t	R/W	ID-Byte 1		0
1541	4x0604	0x0000-0x00FF	Uint16_t	R/W	ID-Byte 0		0
1542	4x0605	0x0000-0xFFFF	Uint16_t	R	Receiving Time		0
1543	4x0606	0x0000-0x000A	Uint16_t	R/W	Response Channel		0

Table 8-1 – Register allocation of sensor data

Learn Channel:
Selection of a receiving channel which shall be put into the learning mode.
After a successful teaching-in, the register is automatically reset to 0.

Receiving Flags:
Bit0 → Receive new telegram on channel 1 (0=No, 1=Yes)
Bit1 → Receive new telegram on channel 2 (0=No, 1=Yes)
Bit2 → Receive new telegram on channel 3 (0=No, 1=Yes)
…
The receiving flag register is automatically reset to 0 after having been read out.

ORG:
ORG-Byte of the EnOcean sensor learned-in.

TYPE:
Type information corresponding to the EnOcean EEP standards (www.enocean.com).
The TYPE-Information is automatically sent by the corresponding sensor during the learn-in procedure. If the corresponding sensor does not support an EnOcean EEP standard, the register shows 0x00FF.

FUNC:
FUNC-Information corresponding to the EnOcean EEP standard (www.enocean.com).
The FUNC-Information is sent automatically by the corresponding sensor during the learn-in procedure.
If the corresponding sensor does not support an EnOcean EEP standard, the register shows 0x00FF.

Data-Byte 3…0:
Data byte of the EnOcean sensor learned-in.

ID-Byte 3…0:
ID-Bytes of the EnOcean sensor learned-in.

Receiving-Time:
Information when the last telegram of the learned-in EnOcean sensor was received (s).

Response Channel:
A value in the range 1…10 results in an automatic sending of the transmitting channel (1…10) upon receipt of a telegram of a sensor learned-in.

8.1.1 Learning-in of EnOcean Sensors

Thanos only supervises the data of those wireless sensors for which the identification code is known i.e. which have been saved in the memory. Corresponding to the Table 8-1, 13 registers are assigned to each sensor whereas the registers „ORG“, „TYPE“ and „FUNC“ are including the information on the sensor type and the registers „ID-Byte 3“, „ID-Byte 2“, „ID-Byte 1“ and „ID-Byte 0“ the identification code.

The sensor identification code is either directly written into the register via a MODBUS telegram or is saved automatically of a received “learn RF telegram” in the learning-mode .

8.1.1.1 Learning-in via MODBUS – Write Instruction

By means of the control command „Write register“ (10hex or 06hex) the identification code can be directly written into the corresponding register. The identification code (ORG-Byte and ID-Bytes) clearly identifies every sensor and is noted on the device label of the wireless sensors.

Example: Learn-in Sensor 1 with ID = 01 23 D5 E7 (hex)

Master - Telegram in transmitting mode RTU:

Device	Command	Start address		Number of Registers		Number of Bytes	Data Register 050A		Data Register 050B		Data Register 050C		Data Register 050C		Checksum	
		H Byte	L Byte	H Byte	L Byte		H Byte	L Byte	H Byte	L Byte	H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	10	05	0A	00	04	08	00	01	00	23	00	D5	00	E7	CRC	

Slave – Response Telegram in transmitting mode RTU:

Device	Command	Start address		Number of Registers		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	10	05	0A	00	04	CRC	

If only a RF telegram of the sensor with the ID = 01 23 D5 E7 is received, the sensor values are written into the corresponding data byte and the monitoring time is reset to the value “0”.

8.1.1.2 Learning-in via Learn-Button of wireless sensor

By writing into the register „learn channel“ a receiving channel of thanos can be set into the learn mode. In the learn mode the receiver is waiting for a radio telegram of a sensor for which the learn button was actuated. Then, the identification code received is written directly into the corresponding register.

Example: Switch receiving channel 3 in the learn mode

Master - Telegram in transmitting mode RTU:

Device	Command	Start address		Number of Registers		Number of Bytes	Data Registers 050A		Checksum	
		H Byte	L Byte	H Byte	L Byte		H Byte	L Byte	L CRC	H CRC
02	10	05	00	00	01	02	00	03	CRC	

Slave – Response telegram in transmitting mode RTU:

Device	Command	Start address		Number Register		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	10	05	00	00	01	CRC	

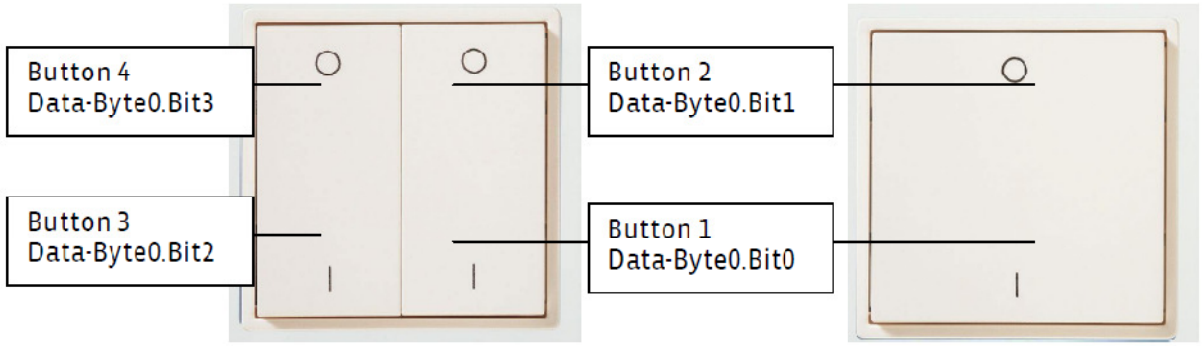
After receipt of a RF learn telegram the register “learn channel” is automatically reset to 0.

8.1.1.3 EnOcean Switch (ORG5)

If an EnOcean switch (1-Byte sensor / ORG5) is learned-in to a receiving channel, the raw data are output in the register „Datenbyte 3“.
Via the register „data byte 0…2“ thanos makes also interpreted data available additionally to the raw data. These registeres are described in the following:

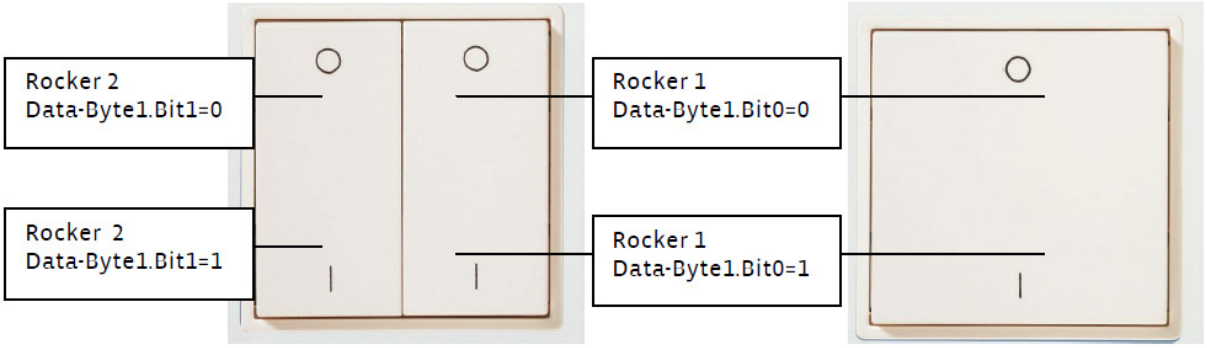
Data-Byte 0

- Current status of keys
- Button function
- All status changes of the key are stored in the device till the next Modbus inquiry and are sent, then.
- After an inquiry of the register, the Data-Byte0 is reset, unless a button is still pressed.
- bit = 1 ==> button pressed, bit = 0 ==> button not pressed



Data-Byte 1

- Current status of rocker
- Switch function
- Button I: Bit0/Bit1 = 1
- Button O: Bit0/Bit1 = 0



Data-Byte 2

- Current status of button
- Button function – status changes of the button are stored in the device till the next Modbus inquiry and are sent, then.
- The button pressed last is stored as RAW value.
- The allocation of the RAW values to the respective button is shown in the data sheet of the keys.

Data-Byte 3

- Current status of button
- The allocation of the RAW values to the respective button is shown in the data sheet of the keys.
- Pressed buttons are not buffered.

Due to the fact, that the Master-Slave-System is too slow with the Modbus, it might come to delays with button actuations.

8.2 EnOcean Transmitting Register

EnOcean Transmitting Register								
Register	**Dataaddress**	**Value Range**	**Type**	**Access**	**Description**			**Default**
1793-1902	**4x0700-4x076D**							
1793	4x0700	0x0000-0x00FF	Uint16_t	R/W	ORG		Data Sender 1	0
1794	4x0701	0x0000-0x00FF	Uint16_t	R/W	Data-Byte 3			0
1795	4x0702	0x0000-0x00FF	Uint16_t	R/W	Data-Byte 2			0
1796	4x0703	0x0000-0x00FF	Uint16_t	R/W	Data-Byte 1			0
1797	4x0704	0x0000-0x00FF	Uint16_t	R/W	Data-Byte 0			0
1798	4x0705	0x0000-0x00FF	Uint16_t	R	ID-Byte 3			0
1799	4x0706	0x0000-0x00FF	Uint16_t	R	ID-Byte 2			0
1800	4x0707	0x0000-0x00FF	Uint16_t	R	ID-Byte 1			0
1801	4x0708	0x0000-0x00FF	Uint16_t	R	ID-Byte 0			0
1802	4x0709	0x0000-0xFFFF	Uint16_t	R/W	Status Byte			0
1803	4x070A	0x0000-0x0001	Uint16_t	R/W	Send	0 = Do not send 1 = Send data		0
. . .								
1892	4x0763	0x0000-0x00FF	Uint16_t	R/W	ORG		Data Sender 10	0
1893	4x0764	0x0000-0x00FF	Uint16_t	R/W	Data-Byte 3			0
1894	4x0765	0x0000-0x00FF	Uint16_t	R/W	Data-Byte 2			0
1895	4x0766	0x0000-0x00FF	Uint16_t	R/W	Data-Byte 1			0
1896	4x0767	0x0000-0x00FF	Uint16_t	R/W	Data-Byte 0			0
1897	4x0768	0x0000-0x00FF	Uint16_t	R	ID-Byte 3			0
1898	4x0769	0x0000-0x00FF	Uint16_t	R	ID-Byte 2			0
1899	4x076A	0x0000-0x00FF	Uint16_t	R	ID-Byte 1			0
1900	4x076B	0x0000-0x00FF	Uint16_t	R	ID-Byte 0			0
1901	4x076C	0x0000-0xFFFF	Uint16_t	R/W	Status Byte			0
1902	4x076D	0x0000-0x0001	Uint16_t	R/W	Send	0 = Do not send 1 = Send data		0

Tabeles 8-2 – Listing of EnOcean transmitting register

ORG:
ORG-Byte of data to be sent.

Data-Byte 3…0:
Data bytes of data to be sent

ID-Byte 3…0:
ID-Bytes of the corresponding transmitting channel (only readable).

Status-Byte:
Status-Byte of data to be sent

Send:
By writing a 1 the transmitting process for the corresponding channel is triggered.
After the transmission the register is automatically reset to 0.

8.2.1 Triggering of a Transmission

By writing a 1 in the register „Send“ a transmitting process can be triggered.
The corresponding values are sent in an EnOcean telegram. Afterwards, the “Send”-Register is automatically set to 0, i.e. it is not necessary to reset the register via another telegram.

Example: Send data via transmission channel no. 1

Master - Telegram transmission mode RTU:

Data to be sent:

Device	Command	Start address		Number of Registers		Number of Bytes	Data Register 0x0700		Data Register 0x0701		Data Register 0x0702		Data Register 0x0703		Data Register 0x0704		Checksum	
		H Byte	L Byte	H Byte	L Byte		H Byte	L Byte	H Byte	L Byte	H Byte	L Byte	H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	10	07	00	00	05	02	00	07	00	AB	00	08	00	13	00	00	CRC	

Slave – Response telegram in transmission mode RTU:

Device	Command	Start address		Number of Registers		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	10	07	00	00	05	CRC	

Triggering of transmission

Master - Telegram in transmission mode RTU:

Device	Command	Start address		Number of Registers		Number of Bytes	Data Register 0x070A		Checksum	
		H Byte	L Byte	H Byte	L Byte		H Byte	L Byte	L CRC	H CRC
02	10	07	0A	00	01	02	00	01	CRC	

Slave – Response telegram in transmission mode RTU:

Device	Command	Start address		Number of Bytes		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	10	07	0A	00	01	CRC	

After receipt of a RF learn telegram the register „Learn Channel“ is automatically reset to 0.

The following RF telegram is transmitted in accordance with the values sent before. The ID of the sender is: 0xFFED8F00

SYNC-BYTE 1		0xA5
SYNC-BYTE 0		0x5A
H-SEQ	LENGTH	0x6B
ORG		0x07
DATA-BYTE 3		0xAB
DATA-BYTE 2		0x08
DATA-BYTE 1		0x13
DATA-BYTE 0		0x00
ID-BYTE 3		0xFF
ID-BYTE 2		0xED
ID-BYTE 1		0x8F
ID-BYTE 0		0x00
STATUS		0x00
CHECKSUM		CS

9 Data Transmission

9.1 Master/Slave Protocol

One master and one or more slaves are connected to the serial bus. The communication between master and slave is exclusively controlled by the master. The slaves are only allowed to send if they have been addressed by the master before. Slaves are only sending back to the master, never to another slave.

9.2 Data Frame

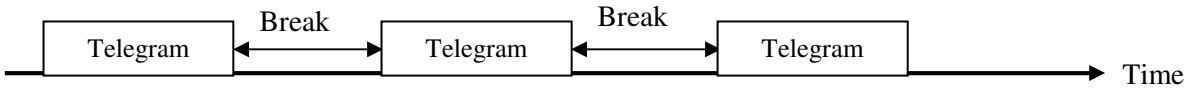
The data are sent to the bus in accordance to severely defined defaults:

Address	Control command	Data	Checksum

In general, a MODBUS telegram starts with the address of the slave, followed by a control command (e.g. read register) and the data. By means of the checksum at the telegram end, the bus participants can recognize transmission errors.

9.3 Transmission Mode RTU

In the transmission mode RTU telegrams are separated by means of transmission breaks:



The period of the transmission breaks for separating telegrams is depending on the adjusted baud rate and amounts to 3,5 * word transmission time (11 bit). With 9600 baud at least 4 ms must pass by and with 57600 at least 1 ms must pass by between two telegrams.

9.3.1 **Telegram Layout**

Address 1 Byte	Control command 1 Byte	Data 0 - 100 byte	Checksum	
			CRC Low	CRC High

9.3.2 **Calculation of CRC-Checksum**

The CRC checksum (Cyclical Redundancy Check) is calculated by the sender out of all bytes transmitted and is attached to the message.

The receiver re-calculates the CRC checksum and compares it with the checksum received. If the values do not correspond, a transmission error is assumed and the data received are rejected.

The least significant byte of the 16 bit checksum is set to the penultimate location and the most significant byte is set at last location.

Calculation of checksum (Programming example in C):

```
crc = 0xFFFF;                                   // CRC-Check, Initialisation
for(i = 0; i < Telegram length-2; i++)
        crc = crc_calc(crc, Telegram data[i]);

crc_low  = crc & 0x00FF;                        // Low-Byte
crc_high = (crc & 0xFF00) >> 8;                 // High-Byte

// Function definition CRC calculation
unsigned int      crc_calc(unsigned int     crc_temp, unsigned int data)
{
        unsigned int      Index_CC=0;           // Loop counter
        unsigned int      LSB=0;                // Help variable

        // Exclusive-Oder des 8Bit-Char with the lower 8Bit of CRC
        crc_temp = ( ( crc_temp ^ data) | 0xFF00) & (crc_temp | 0x00FF) ;

        for(Index_CC = 0; Index_CC<8; Index_CC++)
        {
                LSB = (crc_temp & 0x0001);
                crc_temp          >>= 1;
                if(LSB)
                crc_temp = crc_temp ^ 0xA001;           // calculation polynominal for CRC16
        }

        return(crc_temp);
}
```

10 Examples: Telegrams

10.1 Register

The operating unit has different registers for the configuration, for the display of values and for the input values.

10.1.1 **Configuration of parameter**

The operating unit can be parameterized by the configuration registers and the control commands „Write Register“(10hex or 06hex).

Beispiel: Update upper and lower setpoint 1 range with3K each.

Master - Telegramm in transmission mode RTU:

Device	command	Start address		Number of registers		Number of Bytes	Data Register 1F		Data Register 20		Check Sum	
		H Byte	L Byte	H Byte	L Byte		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	10	00	29	00	02	04	00	1E	00	1E	CRC	

Slave – Response Telegram in Transmission Mode RTU:

Device	command	Start Address		Number of Register		Check Sum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	10	00	29	00	02	CRC	

Set point for temperature will be set when pressing key 1 or 2.

10.1.2 Read-Out Output register

Key and digital input status and values are stored in the output registers. After a reset the basic set points are taken over from the configuration registers for the corresponding set points.

Master - Telegram in Mode RTU		**Slave – Response Telegram in Mode RTU**	
Description	Wert (Hex)	Description	Wert (Hex)
Slave Address	02	Slave Address	02
Command	03	Command	03
Start address High	01	Number of Bytes	14
Start address Low	00	Register value high (0100) Keys 9-16	00
Number of Registers High	00	Register value low (0100) Keys 1-8	08
Number of Registers Low	04	Register value high (0101) Keys 9-16	01
Checksum Low	CRC	Register value low (0101) Keys 1-8	23
Checksum High		Register value high (0102) Offset Setpoint1	00
		Register value low (0102) Offset Setpoint 1	0A
		Register value high (0103) Effektiv Setpoint 1	00
		Register value low (0103) Effektiv Setpoint 1	C8
		Checksum Low	CRC
		Checksum High	

Table 10-1 Read-Out Output Register

10.1.3 Setting of Input Registers

By means of the input registers different values can be overwritten in the operating unit.

Example: Set Time: 14:23:47

Master - Telegram in transmission mode RTU:

device	command	Start address		Number of registers		Number of Bytes	Data Register 513		Data Register 514		Data Register 515		Checksum	
		H Byte	L Byte	H Byte	L Byte		H Byte	L Byte	H Byte	H Byte	L Byte	L Byte	L CRC	H CRC
02	10	02	00	00	03	06	00	47	00	23	00	14	CRC	

Slave – Response Telegram in Transmission Mode RTU:

device	command	Start address		Number of registers		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	10	02	00	00	03	CRC	

Example: Set Date: 23.03.2011

Master - Telegram in transmission mode RTU:

device	command	Start address		Number of registers		Number of Bytes	Data Register 513		Data Register 514		Data Register 515		Checksum	
		H Byte	L Byte	H Byte	L Byte		H Byte	L Byte	H Byte	H Byte	L Byte	L Byte	L CRC	H CRC
02	10	02	03	00	03	06	00	23	00	03	00	11	CRC	

Slave – Response Telegram in Transmission Mode RTU:

device	command	Start address		Number of registers		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	10	02	03	00	03	CRC	

10.2 Coil / Bit Allocation

The operating unit has different configuration bits for adjusting the display value of the display.
By means of the input bits different LEDs, controller etc. can be controlled.

10.2.1 **Writing Configuration Bits**

By means of the control command „Write Bit(s)“ (0Fhex or 05hex) a configuration bit (or more) can be written with the value „1“ or „0“.

Example: Display symbol heating

Master - telegram in Transmission mode RTU:

device	command	Start address		Number of bits		Number of bytes	Data	Checksum	
		H Byte	L Byte	H Byte	L Byte		H Byte	L CRC	H CRC
02	0F	01	01	00	01	01	01	CRC	

Slave-response-telegram in Transmission mode RTU:

device	command	Start address		Number of bits		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	0F	01	01	00	01	CRC	

10.2.2 **Read Bits**

By means of the control command „Read bits“(01_{hex} or 02_{hex}) one or more bits can be read out.

Example: Read out displayed symbols (Data address = 00000_{hex} 00001_{hex})

Master - Telegram in Mode RTU		**Slave – Response Telegram in Mode RTU**	
Description	Wert (Hex)	Description	Wert (Hex)
Device	02	Device	02
Command	01	Command	01
Start address High	00	Number of bytes	01
Start address Low	00	Bit value 0,0,0,0,0,0,bit1,bit0	03
Number of bits High	00	Checksum Low	CRC
Number of bits Low	02	Checksum High	
Checksum Low	CRC		
Checksum High			

Tabelle 10-2 read Bits

11 Update Firmware

To update the thanos firmware, please proceed as follows:

1. Please check if a firmware update of your thanos is feasible at all.
   Therefore, please restart the device.
   During the start procedure the version numbers of the individual software modules are listed in the display.
   A firmware update is only possible if the following is shown in the first line:
   „Bootloader: Version 1.0.0“ (or higher version number)

thanos Start Screen

2. Format an SD memory card (FAT16 or FAT32 file system).

3. Please download the ZIP-archive of the latest firmware from the Thermokon homepage.
   Unpack the ZIP file and copy all the files to the main directory of the SD card.

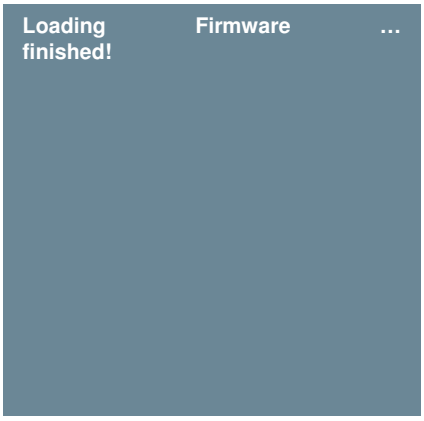
   Download-Link:

   thanos Modbus:
   http://www.thermokon.de/ftp/thanos/thanos_mb_fw_update.zip

4. Remove the thanos operating unit from the wall part and put the SD-card to the operating unit as shown below.

5. Assemble the operating unit to the wall part again. Now, thanos scans automatically for an update on the SD-card and installs the same automatically.

6. After the firmware has been loaded, following message will be displayed:



In order to check if the update procedure is completed successfully, please look at the version number which is indicated in the display during the following start process.

7. Ready [1)] – SD-card can be removed again.

Note:

- The thanos parameter settings are retained even after the firmware update.
- Always use the latest version of the configuration software to ensure error-free operation.
- After the actual firmware a *readme* file is lying in the ZIP archive containing further information for the update. It is very important to read this file carefully before doing the update!

12 Update Configuration Software

To perform an update of thanos configuration software, please proceed as follows:

1. Uninstall the thanos configuration software, which is already located on your PC.

2. Download the ZIP archive of latest configuration software-version.
   Unzip the zip file and run the setup file.
   Please follow the instructions on the screen.

   Download-Link:

   http://www.thermokon.de/ftp/thanos/thanos_mb_eo_csw_update.zip

Note:

- Always use the latest version of the firmware to ensure error-free operation.