

Protocol Description WRF04 RS485 ModBus

Version 1.5, 14.05.2012

Index of Changes

Version	Date	Description
1.0	09.02.2009	First Draft
1.1	04.03..2009	Type 2V added
1.2	18.08.2010	Fixed translation
1.3	06.12.2010	Changes in manual output mode valid since firmware version 1.08, configuration software since version 1.2
1.4	28.11.2011	6-way valve control added valid since firmware version 1.09, configuration software since version 1.3
1.5	14.05.2012	Corrections

Index of Changes	1
1 WRF04-RS485-Modbus	4
1.1 Control	4
1.1.1 Device Types	4
1.1.2 Function Mode of PI-Controller	4
1.1.3 Change-Over Operation	4
1.1.4 Control 6-way valve	4
1.1.5 Energy Stop / Dew Point Detector	4
1.1.6 Override of Outputs	4
1.1.7 Minimal Control Variable	4
1.1.8 Determination of Set Points	5
1.2 Hardware Installation	6
1.3 RS485 Transceiver	6
1.4 Protocol	6
1.5 Configuration Options	6
2 WRF04-RS485-Modbus Protocol	7
2.1 Control Commands Supported	7
2.2 Data Administration	7
2.3 EEPROM – non volatile memory	7
2.4 Register Definition	8
2.4.1 Configuration Register (Holding Register R/W)	8
2.4.2 Output Register (Input Registers R)	11
2.4.3 Input register (Holding Registers R/W)	13
2.5 Bit Allocation / Coil Definition	14
2.5.1 Configuration bits (Coils R/W)	14
2.5.2 Input bits (Coils R/W)	15
3 Data Transmission	16
3.1 Master/Slave Protocol	16
3.2 Data Frame	16
3.3 Transmission Mode RTU	16
3.3.1 Telegram Layout	16
3.3.2 Calculation of CRC-Checksum	17
3.4 Transmission Mode ASCII	18
3.4.1 Telegram Layout	18
3.4.2 Calculation of LRC-Checksum	18
4 Examples: Telegrams	19
4.1 Register	19
4.1.1 Configuration of parameter	19
4.1.2 Read-Out of Output Register	19
4.1.3 Setting of Input Registers	20
4.2 Coil / Bit Allocation	21
4.2.1 Read Bits	21
5 Configuration Software	22

6	Software Installation	22
7	Configuration of WRF04-RS485-Modbus	23
7.1	Software Configuration	23
7.2	Parameter-Frame	24
7.3	Register.....	24
	Picture 7-3: Data.....	24

1 WRF04-RS485-Modbus

The present document describes the serial interface of the room operating unit WRF04-RS485-MODBUS. The MODBUS-Protocol developed by the company Modicon is a disclosed protocol for communication of several intelligent Master-Slave based devices.

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

1.1 Control.

1.1.1 Device Types

The control is integrated in the following devices: AO2V, OVR, OVT, DO2R, DO2T, 6WV.

1.1.2 Function Mode of PI-Controller

The integrated PI-controller controls the temperature (register 0x0102) of set point 1 (Register 0x0104). The control variable resulting is directly output to the outputs. The PI-controller can be set by properties. 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.

1.1.3 Change-Over Operation

The device can be used for a 2-pipe and a 4-pipe wire system. The corresponding selection is made via the configuration tool "Change-Over Operation". If the change-over operation is activated, the corresponding mode (heating or cooling of controller) must be defined. The Change-Over Operation runs via output 1!

1.1.4 Control 6-way valve

Configuring device type 6WV allows to control a 6-way valve. Output 1 is used for heating and cooling:

5..0V = 0..100% actuating variable heating

5..10V = 0..100% actuating variable cooling,

1.1.5 Energy Stop / Dew Point Detector

If a window contact or a dew point detector are connected to the digital inputs and the digital inputs are parameterized the same, both are directly affecting the control.

1.1.6 Override of Outputs

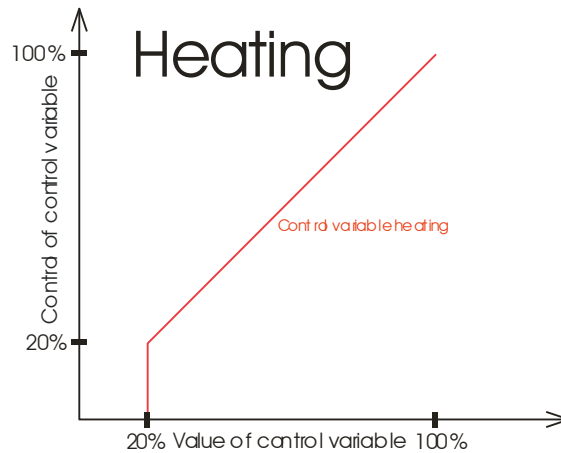
The outputs can be affected and overwritten directly by a master device. Therefore, it is necessary to set the outputs into manual operation (configuration bits 0x000A and 0x000B). In the configuration registers 0x0001E and 0x001D the requested start values for the outputs are parameterized. During operation an overriding of the outputs is possible via the input registers 0x0206 and 0x0207.

1.1.7 Minimal Control Variable

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

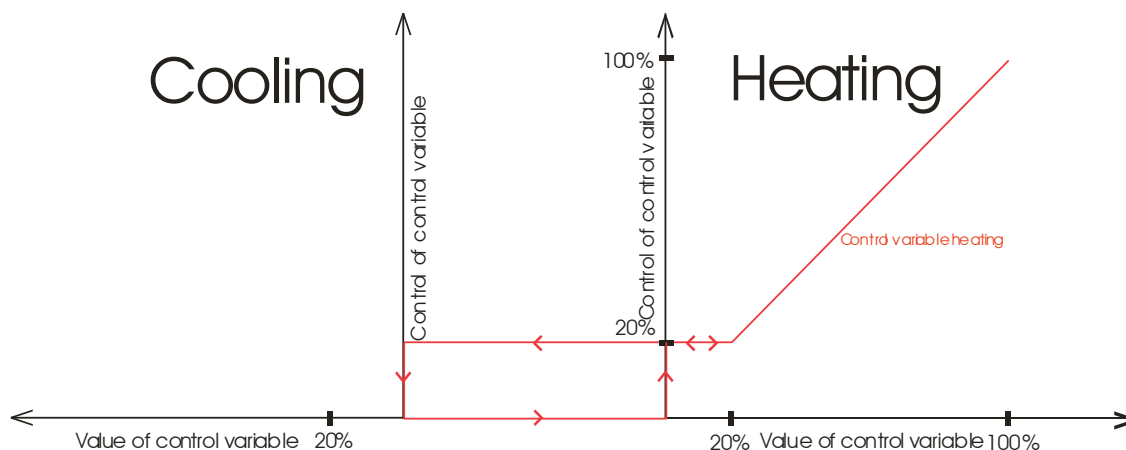
Mode Selection Control Variable (Register 0x0008)

- (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.

- (2) Mode selection Control Variable = 0
Ymin = 20%



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

1.1.8 Determination of Set Points:

(1) OCCUPIED

- *Heating set point* = basic set point + offset(Adr. 0x0202) + poti offset*
- *Cooling set point* = basic set point + dead zone (Adr. 0x000E) + offset(Adr. 0x0202) + Poti-Offset*

(2) UNOCCUPIED

- *Heating set point* = basic set point + offset(Adr. 0x0202) + poti offset* -night lowering
- *Cooling set point* = basic set point + dead zone(Adr. 0x000E) + offset(Adr. 0x0202) + poti offset* + night lowering (Adr. 0x000D)

*if used

1.2 Hardware Installation

The room operating unit can be connected by means of a twisted-pair cable (line resistance 120 Ohm). For detailed information on installation and mounting, please see the product data sheet WRF04-RS485-Modbus and the data sheet wiring_rs485_network.pdf.

1.3 RS485 Transceiver

The maximum number of bus participants without use of a repeater is preset by the RS485-transceiver. The transceiver used enables 32 devices per bus segment at maximum.

1.4 Protocol

The room operating unit WRF04-RS485-Modbus is a slave-bus participant only allowed to send to the bus on demand of the master. The protocol corresponds to the defaults of:

- MODBUS Application Protocol Specification V1.1
- MODBUS via Serial Line Specification & Implementation guide V1.0

1.5 Configuration Options

By means of the DIP switch the device can be adapted to the corresponding bus topology.

5pole DIP switch:

- Bus address of device (1 - 31) via 5 pole DIP switch; DIP switch: 1-5 = 6pole DIP switch:
- Transmitting mode
 - DIP 1 off: RTU
 - DIP 1 on: ASCII
- Baud rate
 - DIP 2 off + DIP 3 off: 9600
 - DIP 2 on + DIP 3 off: 19200
 - DIP 2 off + DIP 3 on: 57600
- Parity
 - DIP 4 off + DIP 5 off: non
 - DIP 4 on + DIP 5 off: even
 - DIP 4 off + DIP 5 on: odd
- Bus terminating resistor 120 Ohm
 - DIP 6 off
 - DIP 6 on
- The number of data bits is fixed and preset to: RTU 8 data bits and ASCII 7 data bits

As the data sheet contains a detailed description of position and meaning of the jumpers, please refer to the file „Produktblatt_wrf04_rs485.pdf“.

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

2 WRF04-RS485-Modbus Protocol

2.1 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

2.2 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.

Procedure	RTU	ASCII
Read register	20	10
Write register	20	10
Read coils	16	8
Write coils	8	8

2.3 EEprom – non volatile memory

Configuration parameters are not allowed to write permanently. Device has maximum write cycles of nonvolatile memory. (dimension: <10000).

2.4 Register Definition

2.4.1 Configuration Register (Holding Register R/W)

Register	Data Address	Value Range	Description	
1 R	0x0000	0x0003	Device coding, not changeable	
2 R	0x0001	0x0012	Firmware version, not changeable	
2 – 60	0x0002 – 0x0025	Configuration of the operating unit, EEPROM- data – !! Don't update permanently EEprom !!		
3 R/W	0x0002	0x0000-0x0010	Device Type	0x0000, Type AO2V (AO1: Heating, AO2: Cooling) (default)
				0x0001, Type DO2R (DO1: Heating, DO2: Cooling)
				0x0002, Type DO2T (DO1: Heating, DO2: Cooling)
				0x0003, Type OVR (DO1: Heating, AO2: Cooling)
				0x0004, Type OVT (DO1: Heating, AO2: Cooling)
				0x0005, Type 6WV (AO1: 5..0V = 0..100% heating, 5..10V = 0..100% cooling)
				0x0010, Type 2V, (AO1: Temperature, AO2:Potentiometer)
4 R/W	0x0003	0x0000-0xFFFF	Device location identification (default = 0x0000)	
5 R/W	0x0004	0x0000-0x00FF	Function button	0x00,without special function (default) 0x20, Room unoccupied 0x21, Room occupied 0x22, Room occupancy toggle 0x23, Bypass mode 0x30, Control Auto/Off toggel
6 R/W	0x0005	0x0000-0xFFFF	Updating interval of display in seconds (default = 0x0A)	
7 R/W	0x0006	0x0000-0x0C80	Min-Response-Delay-Time	signed int, (max 3100 ms) (default = 0x0A = 10 ms)
8 R/W	0x0007	0x0000-0x00FF	Temperature-Offset for calibration of temperature sensor signed int, e.g. 10 _{dec} = +1.0 K, -5 _{dez} = -0.5 K (default = 0x00)	
9 R/W	0x0008	0x0000-0xFFFF	Temperature-Offset for calibration of humidity sensor signed int, e.g. 50 _{dec} = +5.0%, -30 _{dez} = -3.0%	
10 R/W	0x0009	0x0000-0xFFFF	Upper adjustable range set set point (default: 0x00)	Temperature signed int, e.g. 30 _{dec} = + 3.0 K
				Humidity signed int, e.g. 800 _{dec} = 80% rH
11 R/W	0x000A	0x0000-0xFFFF	Lower adjustable range set point (default: 0x00)	Temperature signed int, e.g. -30 _{dec} = - 3.0 K (a figure<0 must be input!)
				Humidity signed int,e.g. 200 _{dez} = 20% rH
12 R/W	0x000B	0x0000-0xFFFF	Set temperature – Basic set point after reset	signed int, e.g. 220 _{dez} = 22.0 °C (default = 0xDC = 22,0 °C)

Register	Data Address	Value Range	Description
13 R/W	0x000C	0x0000-0xFFFF	Set temperature display 0x00 – Basic set point 0x01 – Change over display of heating/cooling set point depending on activated operating mode (default = 0x00)
14 R/W	0x000D	0x0000-0xFFFF	Night setback (unoccupied) (default = 0x32) Heating = set point – night lowering Cooling = set point + night lowering e.g. 40 _{dec} = 4.0 K
15 R/W	0x000E	0x0000-0x0064	Dead band between heating and cooling (default = 0x14) z.B. 20 _{dec} = 2.0
16 R/W	0x000F	0x0000-0x0001	Fan Coil Stage with Auto (default) 0x0000 with Auto 0x0001 without Auto
17 R/W	0x0010	0x0000-0x0064	Proportional range Xp (K) (default = 0x14 = 2.0K) Heating e.g. 40 _{dec} = 4.0 K Xp = 0 deactivates controller
18 R/W	0x0011	0x0000-0x00FF	Reset time Tn (min) (default = 0x64) Heating e.g. 100 _{dec} = 100 min
19 R/W	0x0012	0x0000-0x0064	Maximal control variable limit (default = 0x64) Heating e.g. 100 _{dec} = 100 %
20 R/W	0x0013	0x0000-0x0064	Minimal control variable limit (default = 0x00) Heating e.g. 10 _{dec} = 10 %
21 R/W	0x0014	0x0000-0x00FF	PWM-Cycle time (default = 0x0F) Heating e.g. 15 _{dec} = 15 min
22 R/W	0x0015	0x0000	reserved
23 R/W	0x0016	0x0000-0x0064	Proportional range Xp (K) (default = 0x14 = 2.0K) Cooling e.g. 40 _{dec} = 4.0 K Xp = 0 deaktiviert den Regler
24 R/W	0x0017	0x0000-0x00FF	Reset Time Tn (min) (default = 0x64) Cooling e.g. 100 _{dec} = 100 min
25 R/W	0x0018	0x0000-0x0064	Maximal control variable limit (default = 0x64) Cooling e.g. 100 _{dec} = 100 %
26 R/W	0x0019	0x0000-0x0064	Minimal control variable limit (default = 0x00) Cooling e.g. 10 _{dec} = 10%
27 R/W	0x001A	0x0000-0x00FF	PWM-Cycle time (default = 0x0F) Cooling e.g. 15 _{dec} = 15 min
28 R/W	0x001B	0x0000	reserved
29 R/W	0x001C	0x0000-0x0064	Frost protection (default = 0x32) 0x00 deactivates antifreeze e.g. 50 _{dez} = 5.0 K
30 R/W	0x001D	0x0000-0x03E8	Type: AO2V Analogue manual value after reset 0-10V Output1 Heating* e.g. 1000 _{dec} = 100%
			Type: DO2R, DO2T, OVR, OVT Digital manual value after reset Output1 Heating* 0 - Open >=1 - Closed

Register	Data Address	Value Range	Description	
31 R/W	0x001E	0x0000-0x03E8	Type: AO2V, OVR, OVT Analogue manual value after reset 0-10V Output2 Cooling	e.g. 1000 _{dec} = 100%
			Type: DO2R, DO2T Digital manual value after reset Output2 Cooling*	0 - Open >=1 - Closed
32 R/W	0x001F	0x0000-0x0003 0x0010-0x0013	Selection digital input 1	0x00, Break contact 0x01, Breaker dew point 0x02, Breaker energy hold off 0x03, Breaker occupied 0x04, Breaker controller auto/off 0x05, Breaker controller heating/cooling 0x10, Make contact (default) 0x11, Maker dew point 0x12, Maker energy hold off 0x13, Maker occupied 0x14, Maker controller auto/off 0x15, Maker controller heating/cooling
33 R/W	0x0020	0x0000-0x0003 0x0010-0x0013	Selection digital input 2	
34 R/W	0x0021	0x0000-0x0002	Counter mode digit.input1	0x00, edge 0x01, pulse
35 R/W	0x0022	0x0000-0x0002	Counter mode digit.input 2	0x02, duration
36 R/W	0x0023	0x0000-0x0078	Bypass time occupancy (min)	0x00 – off (default) e.g. 0x0078 = 120 _{dec} = 120 min
37 R/W	0x0024	0x0000-0x0001	Trigger LED	0x00, external trigger 0x01, occupied (ON)/unoccupied(OFF) (default) 0x02, controller active(ON)/inactive(OFF) 0x03, controller cooling active(ON)/inactive(OFF) 0x04, controller heating active(ON)/inactive(OFF)
38 R/W	0x0025	0x0000-0x0003	Controller mode after reset and Off	0 – Controller off 1 – Controller heating 2 – Controller cooling 3 – Controller automatic

2.4.2 Output Register (Input Registers R)

Register	Data Address	Value Range	Description
257 – 273 R	0x0100 – 0x0111	Measuring value (data output)	
257 R	0x0100	0x0000-0x000F	bit0 button 1=pressed, 0=not pressed
258 R	0x0101	0x0000-0x000F	It is buffered if a button was actuated since the last read out of the register. After the read out, all bits are reset to the actual value. bit0 button 1= pressed 0=not pressed
259 R	0x0102	0x0000-0xFFFF	Temperature signed int, e.g. 184 _{dec} = 18.4 °C
260 R	0x0103	0x0000-0xFFFF	Set temperature offset* signed int, Temperature: e.g. -25 _{dez} = -2.5K Humidity: e.g. 510 _{dez} = 51%
261 R	0x0104	0x0000-0xFFFF	Set temperature effective signed int, e.g. 220 _{dec} = 22.0 °C
262 R	0x0105	0x0000-0x03E8	Humidity unsigned int, z.B. 500 _{dez} = 50.0%
263 R	0x0106	0x0000-0x0003 0xFF00-0xFF03	Fan coil stage 0x0000 – off 0x0001 – stage1 0x0002 – stage2 0x0003 – stage3 0xFF00 – auto
264 R	0x0107	0x0000-0x0001	Occupancy mode 0 – unoccupied 1 – occupied
265 R	0x0108	0x0000-0x03FF	Regulating variable heating unsigned int, e.g. 1023 _{dec} =100.0% range 0-1023 corresponds to 0-100%
266 R	0x0109	0x0000-0x03FF	Regulating variable cooling unsigned int, e.g. 1023 _{dec} =100.0% range 0-1023 corresponds to 0-100%
267 R	0x010A	0x0000-0x0001	Local controller lock 0 – not locked 1 – locked
268 R	0x010B	0x0000-0x03E8	Type: AO2V Analogue output value 0-10V Output1 Heating* Unsigned int, e.g. 500 _{dec} = 5V
			Type: DO2R, DO2T, OVR, OVT Digital output state Output1 Heating* 0 – open 1 - closed
269 R	0x010C	0x0000-0x03E8	Type:AO2V, OVR, OVT Analogue output value 0-10V Output2 cooling Unsigned int, z.B. 500 _{dez} = 5V
			Type: DO2R, DO2T Digital output state Output2 cooling 0 – open 1 - closed
270 R	0x010D	0x0000-0x0001	Digital input1 0 - open 1 - closed
271 R	0x010E	0x0000-0x0001	Digital input2 0 - open 1 - closed

272 R	0x010F	0x0000-0xFFFF	Counter value digit. Input1**	0-65535
273 R	0x0110	0x0000-0xFFFF	Counter value digit. Input2**	0-65535
274 R	0x0111	0x0000-0x0003	Controller mode	0 = stop of controller 1 = heating 2 = cooling 3 – automatic heating control 4 – automatic cooling control

- * Value output of local set point adjuster (poti). Depending on assignment of potentiometer!
- ** With any reading out of the counter the value is zeroed! The counter mode is depending on the adjustment of the corresponding configuration property.

2.4.3 Input register (Holding Registers R/W)

Register	Data Address	Value Range	Description
513 - 540	0x0200 - 0x0205		Control (ext. data default)
513 R/W	0x0200	0x0000-0xFFFF	Ext. temperature default signed int, z.B. 170 _{dez} = 17.0°C
514 R/W	0x0201	0x0000-0x03E8	Ext. humidity default unsigned int, z.B. 1000 _{dez} = 100.0%
515 R/W	0x0202	0x0000-0xFFFF	Set point offset signed int, z.B. -25 _{dez} = -2.5K einblenden mit Coilregister 0x0005
516 R/W	0x0203	0x0000-0xFFFF	Base set point signed int, z.B. 220 _{dez} = 22 °C
517 R/W	0x0204	0x0000-0x0001	Occupancy mode 0 – unoccupied (default) 1 – occupied 2 – bypass mode
518 R/W	0x0205	0x0000-0x0003	Controller mode* 0 = controller off 1 = heating (cooling deactivated) 2 = cooling (heating deactivated) 3 = automatic (default)
519 R	0x0206	0x0000-0x03E8	Type: AO2V Analogue manual output value 0-10V Output1 Heating* Unsigned int, e.g. 500 _{dec} = 5V
			Type: DO2R, DO2T, OVR, OVT Digital manual output state Output1 Heating* 0 – open 1 - closed
520 R	0x0207	0x0000-0x03E8	Type: AO2V, OVR, OVT Analogue manual output value 0-10V Output2 cooling Unsigned int, z.B. 500 _{dez} = 5V
			Type: DO2R, DO2T Digital manual output state Output2 cooling 0 – open 1 - closed

*Mode change-over upon use of a 2-wire pipe system, i.e. warm and cold water are sharing the inflow and waste pipe (Change-Over operation, please also see configuration bit Change-Over-operation with data address 0x20A). Active frost protection automatically activates the heating controller.

Data- Address	Description
0xFF00 – 0xFFFF	Range defined by the manufacturer, not allowed to be changed!

2.5 Bit Allocation / Coil Definition

2.5.1 Configuration bits (Coils R/W)

Bit	Data Address	Description
0x0000 – 0x0014		Configuration of Operating unit Bit-Register, EEPROM- Data Configuration of Display – !! Don't update permanently EEprom !!
1 R/W	0x0000	Room temperature 1 = display (default) 0 = do not display
2 R/W	0x0001	Humidity 1 = display (default) 0 = do not display
3 R/W	0x0002	Set point humidity 1 = display (default) 0 = do not display
4 R/W	0x0003	Set point temperature 1 = display (default) 0 = do not display
5 R/W	0x0004	Set point temperature 1 = absolute (default) 0 = offset
6 R/W	0x0005	Change of set point temperature 1 = absolute (default) 0 = offset
7 R/W	0x0006	°C/°F 1 = °C (default) 0 = °F
8 R/W	0x0007	Humidity display 1 = with RH (default) 0 = without RH
9 R/W	0x0008	Use Minimal Control Variable with control variable > 0: = 1 Use Minimal Control Variable with control variable = 0: = 0 (default)
10 R/W	0x0009	Room occupancy after reset 1 = occupied (default) 0 = unoccupied
11 R/W	0x000A	Manual/Automatic operation of Output 1 0 – Auto (default) 1- Manual
12 R/W	0x000B	Manual/Automatic operation of Output 2 0 – Auto (default) 1- Manual
13 R/W	0x000C	Change over mode 0 – 4-wire-system (default) 1 – 2-wire-system*
14 R/W	0x000D	Poti assignment 0 – Temperature (default) 1 – Humidity

* If the Change-Over operation is activated, the corresponding mode of the controller (heating,cooling) must be defined via the Holding Register „Control mode“ with the data address 0x205 or by one of the digital inputs. The Change-Over operation runs via output1.

2.5.2 Input bits (Coils R/W)

Bit	Data Address	Description	
0x0100 – 0x0104	External control Bit-Register		
257 R/W	0x0100	Status signal dew point	0 – Dew point inactive (default) 1 – Dew point active
258 R/W	0x0101	Status signal energy hold off	0 – Energy hold off inactive (def.) 1 – Energy hold off active
259 R/W	0x0102	Release of local presence button ¹	0 – Locked 1 – Released (default)
260 R/W	0x0103	Release of controller ²	0 – Locked 1 – Released (default)
261 R/W	0x0104	Trigger LED ³	0 – OFF (default) 1 – ON

¹ *Release* – Change of presence model vial local presence button possible

Locked– Local presence buttons are locked

² *Controller locked*– Both controllers are inactive, active antifreeze automatically activates the heating controller

Release Controller – Both controllers are released, the controllers are working in the controller mode, which is set in the corresponding configuration property „Controller Mode“.

³ The *LED* can only be controlled, if the configuration register for the LED control is parameterized, accordingly!

3 Data Transmission

3.1 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.

3.2 Data Frame

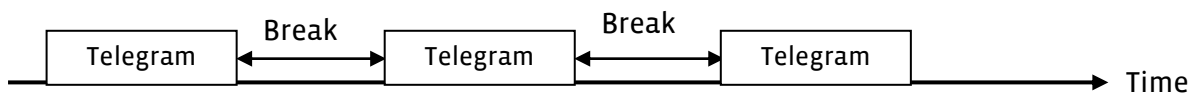
The data are sent to the bus in accordance to severely defined defaults:

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.

3.3 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 $3,5 \times \text{word transmission time (11 bit)}$. With 9600 baud at least 4 ms must pass by and with 57600 at least 1 ms must pass by between two telegrams.

3.3.1 Telegram Layout

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

3.3.2 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-Order 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 polynomial für CRC16
    }

    return(crc_temp);
}
```

3.4 Transmission Mode ASCII

The ASCII transmission mode does not make that high demands on the computer speed of the bus participants. The telegrams are not separated by break times, but by ASCII control characters.

3.4.1 Telegram Layout

The ASCII control character „:“ always identifies the beginning of a telegram. The ASCII control characters „CR“ and „LF“ identify the end of a telegram. The telegram data are output hexa-decimal in the ASCII format:

e.g.: 197dez (1Byte) = C5hex (1 Byte) = C (1 Byte) 5 (1 Byte) ASCII

As one data byte is displayed by 2 ASCII characters, the number of data bytes to be transmitted is doubled compared with the RTU mode.

Start 1 char	Address 2 char	Control command 2 char	Data 0 - 2 x 100 char	Checksum LRC 2 char	End 2 char
:					CR LF

3.4.2 Calculation of LRC-Checksum

The LRC checksum (Longitudinal Redundancy Check) is calculated by the sender out of all bytes transmitted (without „:“, „CR“, „LF“) and pasted in the message of „CR,“ and „LF“. The receiver recalculates the LRC 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 most significant ASCII character of the 8 bit checksum is sent in the telegram before the least significant ASCII character.

Calculation of checksum (programming example in C):

```
lrc = 0;
for(i = 1; i < Telegram length -4; i++)
    lrc = lrc + Telegram data [i];
```

```
lrc = 0xFF - lrc;
lrc = lrc + 1;
```

4 Examples: Telegrams

4.1 Register

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

4.1.1 Configuration of parameter

The operating unit can be parameterized by the configuration registers 3-42 and the control commands „Write Register“(10hex or 06hex).

Example: Digital input 1 as breaker dew point and digital input 2 as maker energy hold off.

Master - Telegram 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	1F	00	02	04	00	01	00	12	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	06	00	02	CRC	

4.1.2 Read-Out of Output Register

Button 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	Value (Hex)	Description	Value (Hex)
Slave Address	02	Slave Adresse	02
Command	03	Command	03
Start address High	01	Number of Bytes	14
Start address Low	00	Register value High (0100)	00
Number of Registers High	00	Register value Low (0100) Button state	01
Number of Registers Low	04	Register value High (0101)	00
Checksum Low	CRC	Register value Low (0101) Button memory	01
Checksum High		Register value High (0102) Temperature	00
		Register value Low (0102) Temperature	DC
		Register valueHigh (0103) Set point offset	FF
		Register Wert Low (0103) Set point offset	E7
		Checksum Low	CRC
		Checksum High	

4.1.3 Setting of Input Registers

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

Example: Activate occupancy bypass mode

Master - Telegram in Transmission mode RTU:

Device	Command	Start address		Data Register 513		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	06	02	04	00	02	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	04	00	02	CRC	

4.2 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.

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 Set point temperature

Master - telegram in Transmission mode RTU:

Slave Address	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	00	04	00	01	01	01	CRC	

Slave – response telegram Transmission mode RTU:

Slave Address	command	Start address		Number of bits		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	0F	00	04	00	01	CRC	

4.2.1 Read Bits

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

Example: Inquiry of operating mode of outputs (Data Address = 0000Ahex 0000Bhex) -> Here: Both outputs are in manual operation

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

5 Configuration Software

By means of a RS485-interface (e.g. RS232-RS485-level converter e.g. ADAM-4520) it is possible to access to the Modbus by the configuration software. The configuration software is not obligatory necessary for the installation of the WRF04-RS485 Modbus. It is possible to use any programme producing Modbus telegrams which is suitable to set registers.

6 Software Installation

For the installation of the configuration software, the setup files „WRF04_Modbus_Config_Setup.exe“ must be started. Please note that you must have administrator rights for the installation. During the installation, please follow the screen instructions.

After a successful operation, the configuration software can be started via the “Starting Menu/Programs/Thermokon“

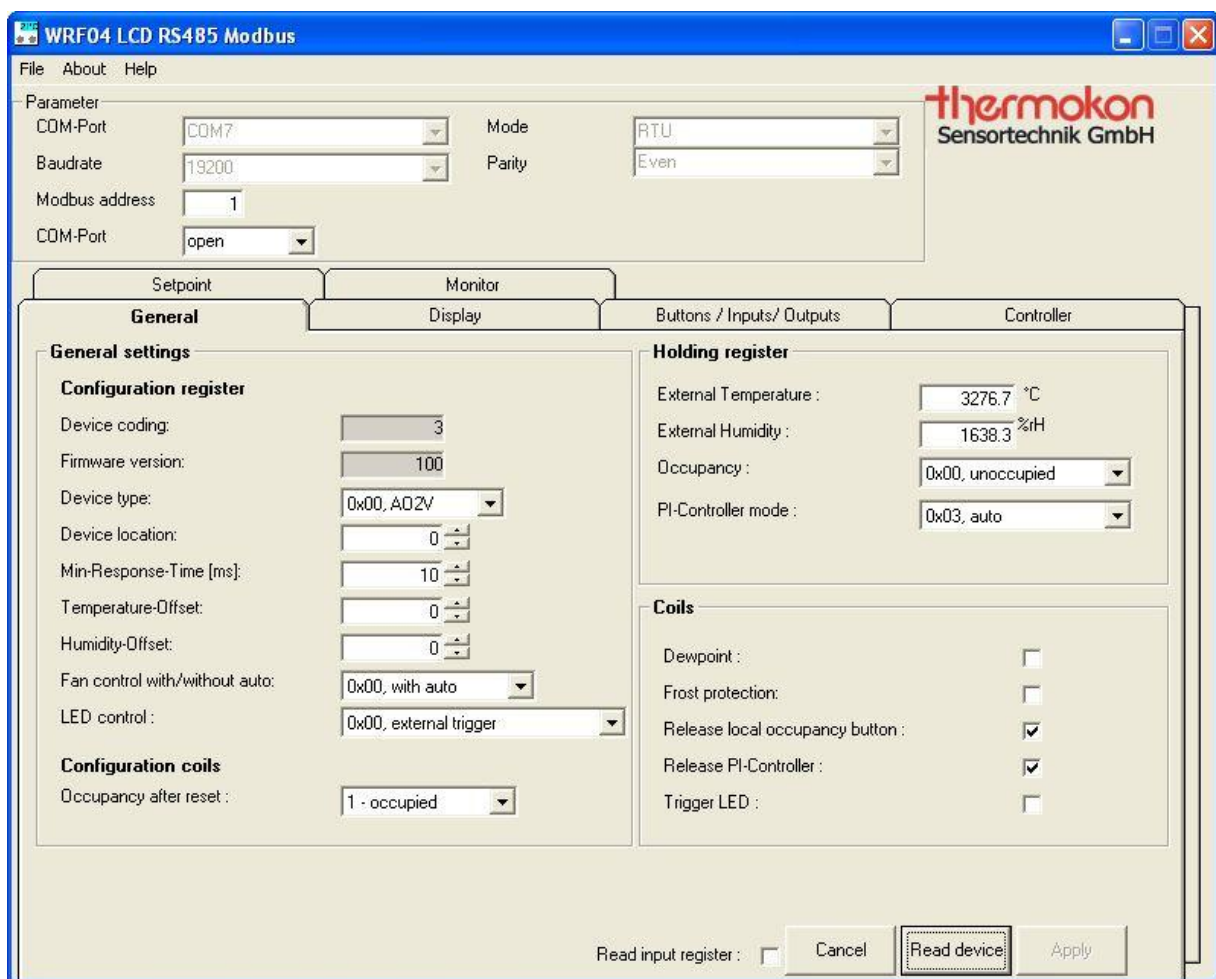
Operating systems supported: Windows9x; WindowsNT; WindowsMe; Windows2000;
 WindowsXP; WindowsServer

7 Configuration of WRF04-RS485-Modbus

7.1 Software Configuration

By means of the configuration software the configuration registers can be clearly adjusted. Output registers of the WRF04 can be read out and input registers can be set. The load of the individual registers is described in chapter 2.4!

Via the menu points "File" and "Saving of Parameter" respectively "Loading of Parameter", the configuration registers can be stored in a text file and can be reloaded into the WRF04-RS485-Modbus.



Picture 7-1: Configuration software

7.2 Parameter-Frame

The Modbus can be accessed via the configuration software by means of a COM-Port. In the "Parameter"-Frame hardware settings can be made. They must be in conformity with the Modbus receiver, in order to produce a connection.

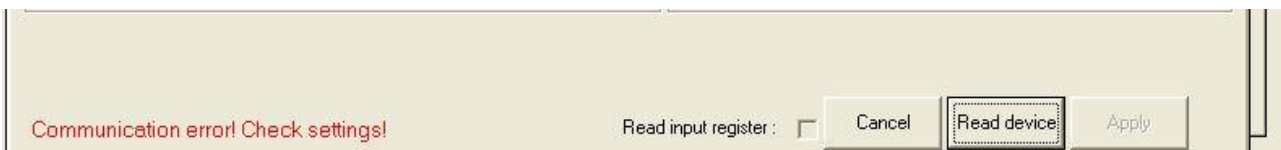
The following options can be selected:

- COM-Port
- Baud rate 9600 , 19200, 57600
- Parity none, even, odd
- Modus for setting of transmission ASCII or RTU
- Modbus address (1-31)

In the field "Modbus address" the address of the WRF06-RS485 Modbus that shall be configured is entered (value between 1 and 31).

Via the selection menu behind "COM-Port" the port can be opened "open" and closed "close". If the COM-Port is used already, an error message is shown.

After having opened the COM-Port, the current register values of the device can be read out via the button "read out device". If no connection to the device can be made, the same is shown by an error message.



Picture 7-2: Communication Problems

7.3 Register

The configuration registers can be set in the different tabs. Furthermore, the output registers can be read and the input registers can be set.

Changes are sent to the WRF06-RS485 Modbus after having pressed the button "take over". By actuating the button "Cancel" the registers of the WRF06-RS485-Modbus are read out again.

By activating the hook "read output register" all output registers are read out cyclically.



Picture 7-3: Data