thermokon®

HOME OF SENSOR TECHNOLOGY

Description

WRF06INC RS485 Modbus

Version 1.2, 30.09.2020

1 Revision

Revision	Date	Description
1.2	30.09.2020	Corrections, register 9: temperature offset
1.1	25.09.2019	From version 3.4: Option occupancy added
1.0	11.08.2018	First issue

2 WRF06INC-RS485

This document describes the serial interface of the room control unit WRF06INC-RS485. The Modbus protocol developed by Modicon is an open protocol for the communication of several intelligent devices on a master-slave basis.

Further information and definitions on Modbus are available at www.modbus.org.

2.1 Hardware Installation

The device can be connected by means of a twisted-pair cable (line resistance 120 Ohm).
Detailed information on the installation and mounting can be found in the product data sheet of the corresponding device and the data sheet wiring_rs485_network.pdf.

2.2 RS485 Transceiver

The maximum number of BUS participants without the use of a repeater is defined by the RS485 transceiver.
The transceiver used in the device enables 32 devices per BUS segment.

2.3 Protocol

The device is a slave BUS participant, which is only allowed to send to the bus upon request of a master. The protocol is in accordance with the definitions of:
- MODBUS Application Protocol Specification V1.1 (Link)
- MODBUS over Serial Line Specification & Implementation guide V1.0 (Link)

2.4 Communication

Two dip switches are used to set the device address and various Modbus options.

5-pole Dip Switch ADDRESS: (1 - 31)

5-pole Dip Switch Modbus options.

(a) Baud rate: 9600, 19200, 38400, 57600
(b) Parity: Odd, Even, None
(c) Communication: RTU, ASCII

A description of the dip switches and the possible settings can be found on the next page.

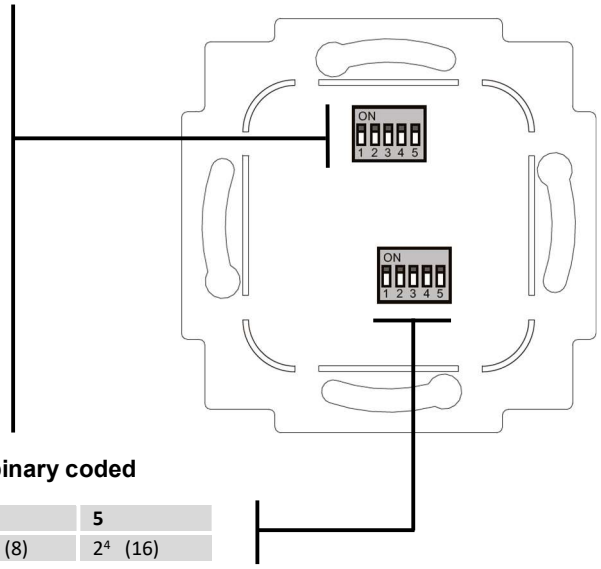
The transmission parameters are preset to 19200Bd, 8E1. Communication works via Modbus RTU protocol.

Modbus Options (Switch block 1, upper block)

1	Mode
off	RTU (default)
on	ASCII

2	3	Baud rate
off	off	9600
on	off	19200 (default)
off	on	38400
on	on	57600

4	5	Parity
off	off	None - 2 stopbits
on	off	even (default)
off	on	odd
on	on	None - 1 stopbit



Modbus Address (Switch block 2, lower block) binary coded

Dip switch	1	2	3	4	5
Priority	2^0 (1)	2^1 (2)	2^2 (4)	2^3 (8)	2^4 (16)

3 Function description

The WRF06INC RS485 is a room control unit with Modbus. It offers the possibility of setpoint adjustment and the indication / display of an operating signal by a LED.
The setpoint can be arbitrarily changed by rotating an encoder in a predetermined range, e.g. -3K ... + 3K, that can be set via Modbus. The current state of the setpoint adjustment is represented by the LEDs.
An operating mode LED can be switched on / off by Modbus presetting and/or pressing the encoder button.

3.1 Encoder

Rotating the encoder is used for setpoint adjustment. The push function can be assigned different functionality. It can e.g. be used without any function, indicating its status only via Modbus (real-time and memory registers) or directly activating the operating mode LED in addition.

3.2 Operating mode LED function

Via Modbus, the operating mode LED can be switched on / off from external (overwrite). If the encoder button is parameterized so that it can control the LED, the last updated value (Modbus or encoder button) decides on the status of the LED.

3.3 Set Point

The setpoint can be adjusted with the encoder in a previously configured range. The configuration parameter “setpoint adjustment range” (protocol address 3) can be used to adjust the range from +/- 0 to +/- 10K. Likewise, the number of pulses / steps per °C can be configured. The factory setting of the number of pulses / steps is 3 per ° C.
The setpoint offset set by the user on the device can be reset or overwritten directly via the input register “Default setpoint offset”. A second input register “Default setpoint offset” (transfer by pressing a key) saves the default value. Only an actuation of the encoder button ensures that the value is accepted. The button must be configured accordingly. Here it can be configured just in case the operating LED is activated when the value is accepted or not.
The local setpoint adjustment can be disabled by default via Modbus.
After a reset, the offset value equals to = 0K.

3.4 Room occupancy

Ig configured the room occupancy can be switched between occupied / unoccupied by pressing the encoder key. The operating mode LED is switched on in occupancy state. When changing from occupied to unoccupied, the setpoint is reset. In the unoccupied state, it is not possible to modify the setpoint.

3.5 LED

The device has 8 LEDs. The setpoint adjustment range is displayed by means of 7 LED’s, the eighth LED is used to display the current status.
If the set offset value is 0, this is indicated by the green LED placed in the middle. If the offset is negative, the blue LEDs are switched on one after the other, and the red one after the other. The step size of the next LED switched on can be configured individually.

Rotating or pressing the encoder button sets the brightness of all LEDs to 100%. After a configurable time has elapsed without activating the encoder, the device goes into sleep mode and the LEDs are dimmed to a set value. This dimming value as well as the time can be set in a configuration register. The operating mode LED can be switched ON or OFF via Modbus or by pressing the encoder button. After a reset (warm or cold start), only the green LED will light up.

3.6 Temperature

The WRF06INC RS485 can be used with an internal or external sensor. The value is displayed via the corresponding input register. As soon as an external sensor is connected, its value is used as the actual temperature. In addition, the status of the sensor can be queried via the register “Alarm message external temperature sensor”. This register can take three values: 0 for normal operation, 100 for sensor short circuit and 200 for interruption.
A configuration register is available to compensate for any measurement deviations (line resistance, etc.).
The measuring range is 0..50 °C.

4 WRF06INC RS485-Modbus Protocol

4.1 Supported Control Commands

The following MODBUS control commands are supported by the device:

Description	Function code	
Read Holding Register	03 (hex)	3 (dez)
	04 (hex)	4 (dez)
Write Single Register	06 (hex)	6 (dez)
Write Multiple Register	10 (hex)	16 (dez)

Table 1 Control Commands

4.2 Data Management

All data in a Modbus slave are assigned addresses. The data (read or write) can be accessed by the corresponding control command and the corresponding data address.

4.3 Modbus Register Description

4.3.1 Configuration Registers (Holding Register R/W)

Modbus Holding Register (R/W)	
Protocol address	**Description**
2	**(#) Min-Response-Delay-Time in ms** *Unsigned Integer* 0x0A = 10dez = 10ms *factory default: 5ms*
3	**(#) set point range** *Signed Integer* i.e. 0x12C = 300dez = +-3K = range -3K … +3K *factory default: +-3K*
4	**(#) number of pulses per °C of the encoder** *Unsigned Integer* i.e. 0x03 = 3dez = 3 pulses *factory default: 3*
5	**(#) increment in K until the next LED is switched on** *Unsigned Integer* i.e. 0x32 = 50dez = 0.5K *factory default: 1K*
6	**(#) LED brightness in sleep mode 0-100%** *Unsigned Integer* i.e. 0x50 = 80dez = 80% *factory default*: 20%
7	**(#) LED brightness in active mode 0-100%** *Unsigned Integer* i.e. 0x50 = 80dez = 80% *factory default: 100%*
8	**(#) duration LED brightness 100% in seconds** *Unsigned Integer* i.e. 0x00FF = 256dez = 256s *factory default: 10s*
9	**(#) Offset in K for temperature sensor (internal/external)** *Signed Integer* i.e. 0x0A = 10dez = 0.1K, 0xFF9C = -100dez = -1K *factory default: 0K*
10	**(#) function button** *Unsigned Integer* 0: no internal function *(factory default)* 1: Toggle LED 2: accept setpoint selection (Register 516) 3: accept setpoint selection (Register 516) with LED control 4: Room occupancy

The registers marked # are stored in the EEPROM. They may only be written during the configuration of the device and not during operation!

4.3.2 Input Registers R

Modbus Holding Register (R/W)	
Protocol address	**Description**
256	**set point Offset in K** *Signed integer* i.e. 0x64 = 100dez = 1K, 0xFF9C = -100dez = -1K
257	**temperature internal/external sensor (0-50°C, display 0-5000)** *Unsigned Integer* i.e. 0x09C4 = 2500dez = 25°C
258	**alarm external temperature sensor** *Unsigned integer* 0: no failure, 100: short circuit, 200: break
259	**button state** *Unsigned Integer* 0: not pressed 1: pressed
260	**button memory function** *Unsigned Integer* 0: was not pressed 1: has been pressed since the last read
261	**State of operation-LED** *Unsigned Integer* 0: OFF/UNOCCUPIED 1: ON/OCCUPIED

Button memory function
Displays, if the encoder has been pressed since the last read. After reading, the value of the register is reset to zero.

Temperature internal/external sensor
If an external sensor is detected, the output automatically switches to the external sensor. By default, the value of the internal sensor is displayed.

4.3.3 Holding Registers R/W

Modbus Holding Register (R/W)	
Protocol address	**Description**
512	**control operating mode LED** *Unsigned Integer* 0: OFF/UNOCCUPIED *(factory default)* 1: ON/OCCUPIED

Modbus Holding Register (R/W)	
513	**control set point offset in K** *Signed integer* i.e. 0x64 = 100dez = 1K, 0xFF9C = -100dez = -1K *factory default: 0K*
514	**lock set point adjustment** *Unsigned Integer* 0: not locked *(factory default)* 1: lock
515	**control set point offset in K (accept by pressing the button)** *Signed integer* i.e. 0x64 = 100dez = 1K, 0xFF9C = -100dez = -1K *factory default: 0K*

Control operating mode LED
Allows direct activation / overwrite of the operating mode LED from external.

Control set point Offset
Allows direct activation / overwrite of setpoint LEDs from external.

Lock setpoint adjustment
A received “1” in the holding register and an associated inhibition of the adjustment is maintained until the master sends a “0” or the device performs a reset.

Control set point offset (accept by pressing the button)
Enables a setpoint input, which is only accepted by pressing the button. The button has to be configured accordingly.

Attention: The input registers must be written with the command Write Single Register (Function code 6)!

5 Data Transmission

5.1 Master/Slave Protocol

One Master and one or several Slaves are connected to the serial bus. The communication between Master and Slave is solely controlled by the Master. Slaves are only allowed to send if they were called by the Master before. Slaves are only sending to a Master, never to another Slave.

5.2 Data Frame

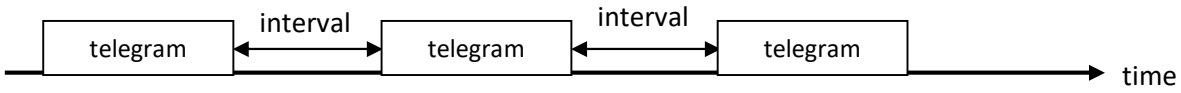
Data are sent to the bus in accordance with strictly defined defaults:

Address	Control Command	Data	Checksum

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

5.3 Transmission Mode RTU

In the transmission mode RTU telegrams are separated by transmission pauses.



The period of the transmission pauses for the separation of telegrams is depending on the set baud rate and amounts to 3.5-times the Word-Transmission Time (11 Bit). With 9600 Baud at least 4 ms and with 57,600 at least 1ms must pass by between two telegrams.

5.3.1 Telegram Structure

Address 1 Byte	Control Command 1 Byte	Data 0 - 100 Byte	Checksum

5.3.2 Calculation of CRC-Checksum

The CRC checksum (Cyclic Redundancy Check) is calculated by the sender by means of all bytes transmitted and is attached to the message. Then, the receiver calculates the CRC checksum again and compares the same with the checksum received. If the values are not matching, a transmission error must be assumed and the data received are rejected. The low-order byte of the 16 bit checksum is sent in the telegram next to the last position and the high-order byte at the last position.

Calculation of Checksum (programming example in C):

```
crc = 0xFFFF;                                   // CRC-Check, Init
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

// Calculate CRC
unsigned int crc_calc(unsigned int crc_temp, unsigned int data)
{
        unsigned int        Index_CC=0;
        unsigned int        LSB=0;
        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);
}
```

5.4 Sample telegrams

5.4.1 Register

The keypad has various registers for configuration, display of values and input values.

5.4.2 Configuration of the parameter

The room control unit can be parameterised with the configuration registers via the control commands "register write" (10hex or 06hex).

Example: changing the Minimum Response Delay Time

Master - Telegram with transmission mode RTU:

Device	Command	Start Address		Number of Register		Number of Bytes	Data Register 06		Checksum	
		H Byte	L Byte	H Byte	L Byte		H Byte	L Byte	L CRC	H CRC
02	10	00	02	00	01	02	00	1F	CRC	

Slave – answer telegram with transmission mode RTU:

Device	Command	Start Address		Number of Register		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	10	00	02	00	01	CRC	

5.4.3 Reading the input register

Temperature, setpoint offset and external sensor status are stored in the output registers.

Master - telegram with RTU		Slave - answer telegram with RTU	
Description	Value (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)	00
Number of Register High	00	Register value Low (0100) set point-Offset	00
Number of Register Low	04	Register value High (0101)	FF
Checksum Low	CRC	Register value Low (0101) temperature internal Sensor	FF
Checksum High		Register value High (0102)	09
		Register value Low (0102) temperature external Sensor	C4
		Register value High (0103)	00
		Register value Low (0103) Status external Sensor	00
		Checksum Low	CRC
		Checksum High	

Offset: 0K, internal sensor not available, external sensor: 25.0 °C, external sensor status: OK

5.4.4 Set holding registers

Example: LED reset to default

Master - telegram with transmission mode RTU:

Device	Command	Start Address		Data		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	06	02	00	00	01	CRC	

Slave – answer telegram with transmission mode RTU:

Device	Command	Start Address		Data		Checksum	
		H Byte	L Byte	H Byte	L Byte	L CRC	H CRC
02	06	02	00	00	01	CRC	