

**RN-WN-N01-\***

**Temperature and humidity  
transmitter instruction  
manual  
(Type 485)**

Document version: V1.0



**Table of Contents**

1. product description .....4

2. Equipment installation instructions .....5

3. letter of agreement .....6

4. Configure software use .....10

5. Common problems and solutions .....12

6. contact details .....12

Document history .....

Appendix: Product Dimensional Drawing .....

# 1. product description

## 1.1 product description

The transmitter has a built-in temperature and humidity sensor with a compact size. The circuit uses imported industrial-grade microprocessor chips and imported high-precision temperature sensors to ensure excellent reliability and high precision of the product. The product adopts 485 communication interface standard ModBus-RTU communication protocol, the communication address and baud rate can be set, and the communication distance is up to 2000 meters. The product has anti-reverse connection protection function, the connection will not burn the device.






## 1.2 Features

1. 485 communication interface, standard ModBus-RTU protocol, communication address and baud rate can be set, the communication line can be up to 2000 meters;
2. Temperature accuracy  $\pm 0.5\text{ }^{\circ}\text{C}$ , humidity accuracy  $\pm 4\% \text{ RH}$ , high precision, low drift;
3. Adopt special EMC anti-interference device, can withstand strong electromagnetic interference on site, industrial grade processing chip, wide range of use;
4. 5 ~ 28V wide voltage range power supply, long-distance centralized power supply can still work normally;
5. Anti-connection and reverse protection function of power supply, reverse connection of positive and negative poles will not burn the device.

## 1.3 Main Specifications

DC power supply (default)	5-28V DC	
Maximum power consumption	$\leq 0.05\text{W}$	
Accuracy	humidity	$\pm 4\% \text{RH}$ (5%RH~95%RH, 25 $^{\circ}\text{C}$ )
	temperature	$\pm 0.5\text{ }^{\circ}\text{C}$ (25 $^{\circ}\text{C}$ )
Transmitter circuit operating temperature	-40 $^{\circ}\text{C}$ ~+80 $^{\circ}\text{C}$ , 0%RH~100%RH	
letter of agreement	Modbus-RTU communication protocol	
output signal	485 signal	
Temperature display resolution	0.1 $^{\circ}\text{C}$	
Humidity display resolution	0.1%RH	
Temperature and humidity refresh time	2S	
Long-term stability	temperature	$\leq 0.1\text{ }^{\circ}\text{C}/\text{y}$
	humidity	$\leq 1\% \text{RH}/\text{y}$
Response time	temperature	$\leq 15\text{s}$ (1m/s Wind speed)
	humidity	$\leq 4\text{s}$ (1m/s Wind speed)
parameter settings	Setting via software	

## 1.4 Product number

RN-			Company code
	WN-	Temperature and humidity transmission,	
		N01-	RS485 (Modbus protocol)
			PE 
			PVC 
			MW 
			QT 
			CS 

## 2. Equipment installation instructions

### 2.1 Check before installation

Equipment List:

1. 1 transmitter device
2. Conformity certificate, warranty card, calibration report, etc.
3. USB to 485 (optional)
4. 485 terminal resistance (gift from multiple devices)
5. One buckle

## 2.2 Interface Description

### 2.2.1 Power supply and 485 signal

The wide voltage power input can be 5 ~ 28V. When connecting the 485 signal line, please note that the two lines A \ B cannot be reversed, and the addresses of multiple devices on the bus must not conflict.

### 2.3 wiring

Thread color	Explanation
brown	Positive power supply (5 ~ 28V DC)
black	Negative power supply
green	485-A
blue	485-B

### 2.4 485 Field wiring instructions

When multiple 485 model devices are connected to the same bus, there are certain requirements for field wiring. For details, please consult the after-sales engineer.

## 3. letter of agreement

### 3.1 Communication basic parameters

Coding	8-bit binary
Data bit	8 bit
Parity bit	no
Stop bit	1 person
Error checking	CRC (Redundant Cyclic Code)
Baud rate	2400bit / s, 4800bit / s, 9600 bit / s can be set, the factory default is 4800bit / s

### 3.2 Data frame format definition

Using Modbus-RTU communication protocol, the format is as follows:

Time of initial structure  $\geq$  4 bytes

Address code = 1 byte

Function code = 1 byte

Data area = N bytes

Error check = 16-bit CRC code

End structure time  $\geq$  4 bytes

Address code: It is the address of the transmitter, which is unique in the communication network (factory default 0x01).

Function code: instruction function instruction issued by the host, this transmitter only uses function code 0x03 (read register data).

Data area: The data area is specific communication data, pay attention to the high byte of 16bits data first!

CRC code: two-byte check code.

Host inquiry frame structure:

address code	function code	Register start address	Register length	Check digit low	Check digit high
1byte	1byte	2byte	2byte	1byte	1byte

Slave response frame structure:

address code	function code	Effective bytes	Data area	Second data area	Nth data area	Check code
1byte	1byte	1byte	2byte	2byte	2byte	2byte

### 3.3 Register address

Register address (hex)	PLC or configuration address	content	operating	Support function code
0000 H	40001	Humidity (10 times actual value)	Read only	03、04
0001 H	40002	Temperature (10 times actual value)	Read only	03、04
07D0H	42001	address	Read and write	03、04、06、16
07D1H	42002	Baud rate (0 for 2400, 1 for 4800, and 2 for 9600)	Read and write	03、04、06、16
0050 H	40081	Temperature calibration value (10 times the actual value)	Read and write	03、04、06、16
0051 H	40082	Humidity calibration value (10 times actual value)	Read and write	03、04、06、16

### 3.4 Communication protocol example and explanation

#### 3.4.1 Read the temperature and humidity value of device address 0x01

Inquiry frame (hexadecimal):

address code	function code	starting address	Data length	Check digit low	Check digit high
0x01	0x03	0x00 0x00	0x00 0x02	0xC4	0x0B

Response frame (hexadecimal): (for example, read temperature is -20.5 °C, humidity is 25.8% RH)

address code	function code	Returns the number	Humidity	Temperature	Check digit	Check digit high
--------------	---------------	--------------------	----------	-------------	-------------	------------------

	e	ber of valid byte s	value	value	low	gh
0x01	0x03	0x04	0x01 0x02	0xFF 0x33	0x5B	0xEA

Temperature calculation:

When the temperature is lower than 0 °C, the temperature data is uploaded in the form of complement.

Temperature: FF37 H (Hexadecimal) = -205 => Temperature = -20.5 °C

Humidity calculation:

Humidity: 102H (Hexadecimal) = 258 => Humidity = 25.8% RH

### 3.4.2 Read the temperature calibration value of device address 0x01

Inquiry frame (hexadecimal):

address code	function code	starting address	Data length	Check digit low	Check digit high
0x01	0x03	0x00 0x50	0x00 0x01	0x84	0x1B

Response frame (hexadecimal): (for example, the temperature calibration value read is -1.0 °C)

address code	function code	Returns the number of valid bytes	Temperature calibration value	Check digit low	Check digit high
0x01	0x03	0x02	0xFF 0xF6	0x79	0xF2

Calculation of temperature calibration value:

When the temperature calibration value is less than 0, the temperature calibration value data is uploaded in the form of complement.

Temperature calibration value: FF F6 H (hexadecimal) = -10 => temperature calibration value = -1.0 °C

### 3.4.3 Set the temperature calibration value of device address 0x01

The temperature calibration value is issued at -1.0 degrees. When the temperature calibration value is less than 0, the temperature calibration value data is delivered in the form of a complement.

Temperature calibration value: FF F6 H (hexadecimal) = -10 => temperature calibration value = -1.0 °C

Request frame (hexadecimal):

address code	function code	Register address	Temperature calibration value content	Check digit low	Check digit high
0x01	0x06	0x00 0x50	0xFF 0xF6	0x48	0x6D

Response frame (hexadecimal):



address code	function code	Register address	Temperature calibration value content	Check digit low	Check digit high
0x01	0x06	0x00 0x50	0xFF 0xF6	0x48	0x6D

**Humidity calibration value operation is similar to temperature calibration value operation.**

### 3.4.4 Change the device with address 01 to 02

Request frame (hexadecimal):

address code	function code	Register address	Temperature calibration value content	Check digit low	Check digit high
0x01	0x06	0x07 0xD0	0x00 0x02	0x08	0x86

Response frame (hexadecimal):

address code	function code	Register address	Temperature calibration value content	Check digit low	Check digit high
0x01	0x06	0x07 0xD0	0x00 0x02	0x08	0x86

### 3.4.5 Set the baud rate of device address 0x01 to 4800

Change the baud rate of device 01 to 4800 (00 stands for 2400, 01 stands for 4800, 02 stands for 9600, and 03 stands for unknown baud rate)

Request frame (hexadecimal):

address code	function code	Register address	Baud rate value content	Check digit low	Check digit high
0x01	0x06	0x07 0xD1	0x00 0x01	0x19	0x47

Reply Frame (Hexadecimal)

address code	function code	Register address	Baud rate value content	Check digit low	Check digit high
0x01	0x06	0x07 0xD1	0x00 0x01	0x19	0x47

### 3.4.6 Read device address

Inquiry frame (hexadecimal):

address code	function code	starting address	Data length	Check digit low	Check digit high
--------------	---------------	------------------	-------------	-----------------	------------------

					h
0xFF	0x03	0x07 0xD0	0x00 0x01	0x91	0x159

Response frame (hexadecimal): (for example, the address read is 1)

address code	function code	Returns the number of valid bytes	Current address	Check digit low	Check digit high
0x01	0x03	0x02	0x00 0x01	0x79	0x84

### 3.4.7 Reading device baud rate

Inquiry frame (hexadecimal):

address code	function code	starting address	Data length	Check digit low	Check digit high
0x01	0x03	0x07 0xD1	0x00 0x01	0xD5	0x47

Response frame (hexadecimal): For example, the baud rate read is 4800 (00 stands for 2400, 01 stands for 4800, 02 stands for 9600, and 03 stands for unknown baud rate)

address code	function code	Returns the number of valid bytes	Current address	Check digit low	Check digit high
0x01	0x03	0x02	0x00 0x01	0x79	0x84

## 4. Configure software use

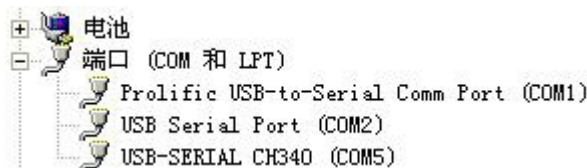
### 4.1 Software selection

Open the data package, select "Debug Software" --- "485 Parameter Configuration

Software"

### 4.2 parameter settings

①, select the correct COM port ("COM" port in "My Computer-Properties-Device Manager-Port"), the following figure lists the driver names of several different 485 converters.



② Only connect one device and power it on, click the test baud rate of the software, the software will test the baud rate and address of the current device, the default baud rate is 4800bit / s, the default address is 0x01.

③ Modify the address and baud rate according to the needs of use, and at the same time can query the current

functional status of the device.

- ④ If the test is unsuccessful, please recheck the device wiring and 485 driver installation.

Parameter	Read	Write	Other
Serial Port Num: COM1	Search Device		
Device Address:	Read	Write	
Device Band Rate:	Read	Write	
Temperature Value:	Read		
Humidity Value:	Read		
Water Leak Status:	Read		
Power Failure Status:	Read		
Light Intensity Value:	Read	Para Set	
CO2 Concentration:	Read		
Switch Output Delay:	Read	Write	
Remote Signal Normal Set:	Read	Write	
Humidity UpperLimit:	Read	Write	
Humidity Lower Limit:	Read	Write	
Temperature Upper Limit:	Read	Write	
Temperature Lower Limit:	Read	Write	
Humidity Hysteresis:	Read	Write	
Temperature Hysteresis:	Read	Write	
Humidity Adjust:	Read	Write	
Temperature Adjust:	Read	Write	
LCD Device Control Mode:	LCD Device Control Mode Set		
Wireless Receiver Para Set:	Wireless Device Para Set		

## **5. Common problems and solutions**

### **The device cannot be connected to a PLC or computer**

possible reason:

- 1) The selected COM port is incorrect.
- 2) The device address is wrong, or there are devices with duplicate addresses (the factory default is all 1).
- 3) Baud rate, check mode, data bit, stop bit error.
- 4) The host's polling interval and waiting time for answering are too short, and both need to be set above 200ms.
- 5) The 485 bus is disconnected, or the A and B lines are reversed.
- 6) If the number of devices is too large or the wiring is too long, power should be supplied nearby, and a 485 booster should be added, and 120 $\Omega$  terminal resistance should be added at the same time.
- 7) The USB to 485 driver is not installed or damaged.
- 8) The equipment is damaged.

