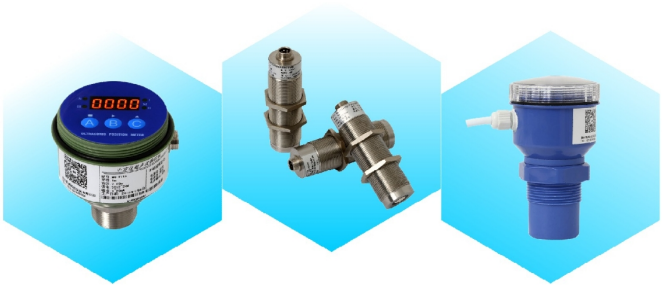
MODBUS-RTU PROTOCOLS



Contents

Ⅰ Definition

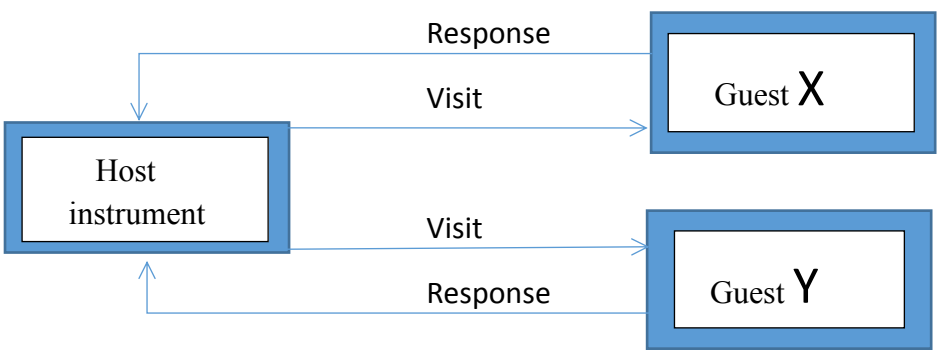
Start-bit	1bit
Data-bits	8bit
Check-bit	none（default） N/O/E
Stop-bit	1bit
Baud rate	9600（default） 600-19200 bps

Ⅱ Communication method, protocol

The communication between the instrument and the external device is RS485，using MODBUS protocol-RTU mode.

Ⅲ Introduction

MODBUS is visit/response protocol, provides function code prescribed service. It used for host/guest communication between different type bus and network connection device. The host sends a command to visit the guest and the guest receives - processes the command then responds to the host.



Ⅳ protocol description

MODBUS has 2 transmission modes， RTU or ASCII. This instrument use RTU.

Address	Function	Data	CRC check

4.1 RTU transmission mode

RTU mode serial bit

Start bit	1	2	3	4	5	6	7	8	check	Stop bit

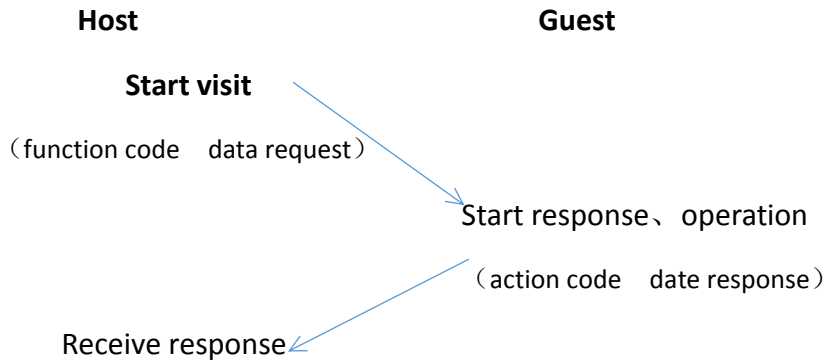
MODBUS-RTU data frames

Address	Function	Data	CRC check
1 byte	1 byte	0-252 byte	2byte

CRC check

CRC check， base on error checking domain of Cyclic redundancy check. This check is performed regardless of whether it has data frame or parity test. CRC contains 16 bits value composed of two 8-bits. CRC attach to date frame, after the calculation, first add the bottom byte, and then attach the high byte. CRC value is calculated by the host, the guest receives the data frame then recalculates it, compares the result with the received CRC value, if it is not equal, and it is an error.

4.2 MODBUS transmission mode



4.3 protocol support

Function code indicates execute functions

Action code type	function code(Decimal)	description
Read data	03	Read menu data
Read data	04	Read data(level or others)
Write data	10	Write data in menu register
Save	12	Date save in EEPROM

Commonly used variables register address description. The following register addresses are all starting addresses.

Instrument	Register (Decimal)	Definition
Level meter	0	Level
	2	temperature
Open channel flow meter （probe signal）	0	Level
	22	Instant flux
	24	Hour flux
	26	Total flux
	28	Accumulative times
Open channel flow meter （current signal）	0	Level
	28	Instant flux
	30	Hour flux
	32	Total flux
	34	Accumulative times

4.4 data storage form in internal register

Variable data is a single-precision floating number (EEE754 standard).

It uses four hexadecimal representations, occupying four internal register addresses, for example 42 F9 80 00.

Storage form:

starting address +3	starting address +2	starting address +1	Register starting address
42	F9	80	00

Ⅴ examples

5.1 host read guest display value data command:

e.g. 01 04 00 00 00 02 71 CB

Address	Function	Register address	Data length	CRC check
01	04	00 00	00 02	71 CB

Read data length is 2 bit, 1bit return 2 bytes.

Guest return command**:**

e.g. 01 04 04 40 A0 00 00 EE 66

Serial address	function	Return data length	Read data	CRC check
01	04	04	40 A0 00 00	EE 66

5.2 host writes data command to guest form

e.g. 01 10 00 00 00 02 04 40 A0 00 00 E6 4D

Serial address	Function	Register address	Register length	Data length	Data	CRC check
01	10	00 00	00 02	04	40 A0 00 00	E6 4D

5.3 host requires guest to save date in EEPROM

01 12 00 A0 00 02 04 00 00 00 A1 99 A5

This is a special command,

When done return to 01 **13 00 00 00 00 00 09 63**

Ⅵ CRC Cyclic Redundancy（CRC16） Check

Cyclic redundancy checks CRC area is 2 bytes, including a 16-bit binary data. CRC value calculated by the sending device, and to calculate the value attached to the message, the receiving device to receive information, the re-calculated CRC value and the calculated values and the reception area of the actual value in the CRC comparison, if they are not the same, Generate an error.

CRC start of all first 16-bit register set as "1", then the adjacent two 8-bit bytes of data into the current register, only the data for each character used to produce 8-bit CRC, the start bit, stop bits and parity bits are not added to the CRC in.

CRC generated during each 8-bit data register value with XOR, the result to the right one (to LSB direction), and "0" fill in the MSB, testing LSB, if LSB is "1" and the pre- different set of fixed values or, if the LSB is "0" is not for XOR.

Repeat the process above office until the shift 8, shift after the completion of the 8th, the next 8-bit data, and the register XOR the current value, in all of the information dealt with, the final value of CRC register values. Generate CRC process:

1, the 16-bit CRC register is loaded into FFFFH.

2, an 8-bit data and low 8-bit CRC register XOR, put the result into the CRC register.

3, CRC register to the right one, MSB zero filling, check the LSB.

4, (if the LSB is 0): Repeat 3, and then shifted to the right one.

(If the LSB is 1): CRC register and XOR Aool H

5, 3 and 4 repeat 8 times until the completion of shift, complete 8-bit byte address.

6, repeat steps 2 to 5, with the next one 8-bit data, until all the bytes processed.

7, CRC final value of CRC register values.

8, the CRC value into the message, the high 8 and low-8 should be separated.

Send a message in the 16-bit CRC value, the first test of the low 8-bit, 8-bit high evacuation.

Ⅶ floating number data format

7.1 IEEE754 standard single-precision floating number analyze

IEEE754 standard single-precision floating number consist of 1 sign bit+8 exponent +23 mantissa, use four hexadecimal digits. e.g. 124.75 use hexadecimal express as 42 F9 80 00. It's calculated as:

24.75 Binary is 1111100.11

Scientific notation express as 1.11110011*2^6

exponent 6＋127＝133,0 for positive, 1 negative. So Binary number for 124.75 is

0 10000101 11110011000000000000000B＝42F98000H

7.2 4 bytes convert to floating numbers base on C language

```
Union                                   //union
{
    Float       testData_float;         //float 4 bytes
    Unsigned    char    testArray[4];   //data
}
TData;
```

note： In the union, floating number and four-byte character groups share a storage space

such as the level value for ultrasonic level meter , read data 42 F9 80

Starting address +3	Starting address +2	Starting address +1	Register starting address
03	02	01	00
42	F9	80	00

00，convert to floating number 124.75。**Data storage form in internal register**:

program：

```
Folat   Tempfloat;
    TData.testArray[3] = 0x42;              //input high byte
    TData.testArray[2] = 0xF9;
```

```
TData.testArray[1] = 0x80;
TData.testArray[0] = 0x00;                    //input lower byte
Tempfloat = testData_float;                   //get floating number
```