Machine Translated by Google

1. Introduction to the Agreement

The communication protocol is MODBUS-RTU protocol. This product only supports function codes 0x03, 0x06, 0x10; the communication interface is TTL serial port;

two. Introduction to communication protocols

Information transmission is asynchronous, Modbus-RTU mode is in 11-bit bytes.

Word format (serial data) 10-bit binary	
Start bit 1	
8 data bits	
No parity bit	
Stop bit 1 bit	

Data frame structure:

The data frame	Address	Function	data area	CRC check
interval is more than 3.5 bytes	code 1 byte	code 1 byte	N bytes	2 bytes

Before sending data, the data bus is required to be stationary, that is, no data transmission time is greater than 3.5 (for example: the baud rate is 9600

5ms) Message sending must start with a pause interval of at least 3.5 bytes, and the entire message frame must be used as a continuous

Continuous data transmission stream, if there is a pause time of more than 3.5 byte times before the frame is completed, the receiving device will refresh

Incomplete message and assumes the next byte is the address field of a new message. Likewise, if a new message arrives less than

It starts after the previous message within 3.5 characters, and the receiving device will consider it as a continuation of the previous message.

1.1 Address code:

The address code is the first byte (8 bits) of each communication information frame, ranging from 1 to 255. This byte indicates the location set by the user

The slave at the address will receive the information sent by the master. Each slave must have a unique address code, and only if it matches the address

Only the slave machine with the address code can respond to the feedback information. When the slave sends back information, the data sent back start with their respective address codes. host

The address code sent by the machine indicates the slave address to be sent, and the address code returned by the slave machine indicates the slave address sent back. Mutually

The corresponding address code indicates where the information comes from.

1.2 Function code:

The function code is the second byte transmitted in each communication information frame. The function codes that can be defined by the ModBus communication protocol are 1 to

127. Sent as a host request, the function code is used to tell the slave what action it should perform. In response to the slave, the slave returns

The function code is the same as the function code sent from the host, and indicates that the slave has responded to the host and has performed relevant operations.

This machine only supports 0x03, 0x06, and 0x10 function codes.

function code	Define	Operation (binary)
0x03	reading register data,	Read data from one or more registers
0x06	writing a single register,	Write a set of binary data to a single register
0x10	writing multiple registers	Write multiple sets of binary data to multiple registers

1.3 Data area

The data area includes what kind of information needs to be sent back or what action needs to be performed by the slave. This information can be data (such as: start

(off input/output, analog input/output, register, etc.), reference address, etc. For example, the host uses function code

03 tells the slave to return the value of the register (including the starting address of the register to be read and the length of the read register), then return

The data returned includes the data length and data content of the register.

0x03 Read function host format

Address	Function code register starting address Number of register addresses n (1~32) 1 byte 2 byte			CRC check code
code 1 byte		2 bytes		2 bytes

0x03 Read function slave return format

Address code function code returns the number of registers			Number of registers	CRC check code

Machine Translated by Google

Factory default baud rate 115200 device address 1

Introduction to protocol registers (the data in a single register address is double-byte data)

name	illustrate	Number of bytes	decimal point	unit	reading and writing	Register (decim
V-SET	Voltage setting	2	2	V	R/W	0
I-SET	Current setting	2	3	A	R/W	1
VOUT	Output voltage display value	2	2	V	R	2
IOUT	Output current display value	2	3	A	R	3
POWER	Output power display value	2	2	W	R	4
UIN	Input voltage display value	2	2	V	R	5
AH-LOW	Output AH Low 16-bit output	2	0	htK	R	6
AH-HIGH	AH High 16-bit output WH	2	0	htK	R	7
WH-LOW	Low 16-bit output WH High	2	0	wxya	R	8
WH-HIGH	16-bit on time - hour start	2	0	wxya	R	9
OUT_H	time - minute on time	2	0	H	R	10
OUT_M	-Seconds internal	2	0	M	R	11
OUT_S	temperature value	2	0	S	R	12
T_IN	external temperature	2	1	F/C	R	13
T_EX	value key lock	2	1	F/C	R	14
LOCK	protection	2	0	-	R/W	15
PROTECT	status constant	2	0	-	R/W	16
CVCC	voltage constant current	2	0	-	R	17
ONOFF	status switch	2	0	-	R/W	18
FC	output	2	0	-	R/W	19
B-LED	temperature symbol	2	0	-	R/W	20
SLEEP	backlight	2	0	M	R/W	twenty
MODEL	brightness level	2	0	-	R	twenty
VERSION	screen time product	2	0	-	R	twenty
SLAVE-ADD	model firmware	2	0	-	R/W	twenty
BAUDRATE_L	version	2	0	-	R/W	25
T-IN-OFFSET	number slave address	2	1	F/C	R/W	26
T-EX-OFFSET	baud rate internal	2	1	F/C	R/W	27
BUZZER	temperature	2	0	-	R/W	28
EXTRACT-M	correction external	2	0	-	R/W	29
DEVICE	temperature correction buzzer switch quick adjustment Output data group device status	2	0	-	R/W	30
MPPT-SW	MPPT switch	2	0	-	R/W	31
MPPT-K	MPPT maximum power point coefficient	2	0	-	R/W	32
BatFul	full power cut-off current	2	0	-	R/W	33
CW-SW	constant power switch	2	0	-	R/W	34
CW	constant power value	2	0	-	R/W	35
MASTER	Host type	2	0	0	R/W	45

Note 6: The read and write values of the switch output function are 0 and 1, 0 is the closed state, and 1 is the open state.

Note 7: The reading and writing range of the backlight brightness level is 0-5, with level 0 being the darkest and level 5 being the brightest.

Note 8: The written value of the quick recall data group function is 0-9, and the corresponding data group data will be automatically recalled after writing.

Note 9: Description of WiFi related registers

name	Detailed	Register address
MASTER	description of host type (0x3B3A: WIFI, others to be determined)	0030H
WIFI-CONFIG WIFI pairing status (0: invalid 1: Touch pairing 2: AP pairing)		0031H
WIFI-STATUS WIFI status (0: Invalid network 1: Connected to router 2: Successfully connected to server 3: Touch configuration Pair 4:AP pairing)		0032H
IPV4-H	The first two bytes of the IP address are 0xC0A8	0033H
IPV4-L	The last two bytes of the IP address are 0x0108	0034H

IPV4-H: 0xC0A8 IPV4-L: 0x0108

IPv4 = 192.168.1.8

1.4 Error check code (CRC check):

The host or slave can use the check code to determine whether the received information is correct. Due to electronic noise or some other interference, Errors sometimes occur during the transmission of information. The error check code (CRC) can check whether the host or slave is transmitting communication data. Whether the information in the sending process is wrong, wrong data can be discarded (whether sending or receiving), which increases the system system safety and efficiency. The CRC (redundant cyclic code) of the MODBUS communication protocol contains 2 bytes, which is a 16-bit binary number. The CRC code is calculated by the sending device (host) and placed at the end of the sent information frame. The device (slave) that receives the information then restarts Newly calculate the CRC of the received information, and compare whether the calculated CRC matches the received one. If the two do not match, then Indicates an error. The CRC check code is sent with the low bit first and the high bit last.

Calculation method of CRC code:

(1) Preset a 16-bit register to hexadecimal FFFF (that is, all 1); call this register a CRC register;

(2) Combine the first 8-bit binary data (the first byte of the communication information frame) with the lower 8 bits of the 16-bit CRC register Bit-wise exclusive OR and put the result in the CRC register;

(3) Shift the contents of the CRC register one bit to the right (toward the lower bit), fill the highest bit with 0, and check the shifted-out bit after the right shift;

(4) If the shifted bit is 0: Repeat step 3 (shift one bit to the right again); if the shifted bit is 1: CRC register and multi-item Formula A001 (1010 0000 0000 0001) performs XOR;

(5) Repeat steps 3 and 4 until the right shift is performed 8 times, so that the entire 8-bit data is processed;

(6) Repeat steps 2 to 5 to process the next byte of the communication information frame;

(7) After calculating all bytes of the communication information frame according to the above steps, the high and low values of the 16-bit CRC register obtained Bytes are exchanged;

(8) The final content of the CRC register is the CRC code.

3. Communication examples

Example 1: The host reads the output voltage and output current display values.

The message format sent by the host:

The master	Number of bytes	Sent information	Remark
sends the slave	1	01	Sent to slave with address 01
address	1	03	Read register
function code register	2	0002H	Register starting address
starting address register address number	2	0002H	2 bytes in total
CRC code	2	65CBH	The CRC code is calculated by the host

Slave response	Number of bytes	Returned information	Remarks:
slave address	1	01H	Send to the slave write register with
function	1	10H	address 01. The
code register starting	2	0000H	starting address of the
address register address number	2	0002H	register is a total of
CRC code	2	41C8H	2 bytes and the CRC code is calculated by the slav