HB416PVI instrument transmission standard with the computer using RS485 communication, details are as follows:

Baud Rate: 1200, 2400, 4800, 9600, 19200

Start bit: 1

Data bits: 8

Parity: None

Stop Bits: 1

One. MODBUS_RTU frame structure

Message at least 3.5 character pause time interval beginning; entire message frame must be transmitted as a continuous data stream, prior to the completion of the frame if there are more than 3.5 character times of pause time, the receiving device flushes the incomplete message and assumes that the next byte is a new message address fields. Similarly, if a new message in less than 3.5 character times before a message and then start receiving device will consider it a continuation of the previous message.

A standard information structure is as follows:

Start	Address field	Functional domains	Data field	CRC check	End
T1-T2-T3-T4	8Bit	8Bit	n-8Bit	16Bit	T1-T2-T3-T4

Address field: the host will be contacted by the slave's address into the address field of the message to strobe from the device, a single machine has an address range from 1 ... 15 (decimal).

Address 0 is used as a broadcast address, so that all can understand from the machine.

Functional domains: efficient coding range is 1 ... 255 (decimal); when the message sent from the host from the machine, the function code will be advertised from the machine needs to do. For example: read / write a register of data content.

Data field: sent from the host machine from the machine data field contains the domain of action to complete the necessary additional information; such as: register address.

The CRC: CRC generation, the low byte first, high byte in the post.

Note: This instrument communication between frames when the response interval greater than or equal 9600bps baud rate is not greater than 5ms

Parameter type And password	Display symbol	Parameter function	Data format	Data length (bytes)	Range of data	Address	Default	Read / Write r / w
First Close	IUTY	Voltage unit	Decimal	2	0: V; 1: KV	0x1000	0	r / w
	PTCP	PT ratio	Decimal	2	0 - 9999	0x1001	1	r / w
Start Code	IITY	Current unit	Decimal	2	0: A; 1: KA	0x1002	0	r / w
	CTCP	CT ratio	Decimal	2	0 - 9999	0x1003	1	r / w
Of 0	PTY	Power unit	Decimal	2	0: W; 1: KW; 2: MW	0x1004	0	r / w
Parameter 0	ID	RS485 address	Decimal	2	0 - 255	0x1005	0	r / w
	BAUD	Communication baud rate	Decimal	2	0: 1200; 1: 2400; 2: 4800 3: 9600; 4: 19200	0x1006	4	r / w
Number 8								

The 9 **Home**								
	NULL	Air				0x1007		
Power Close	P - WS	Alarm only use selected	Decimal	2	**0: NO; 1: H; 2: L; 3: HL**	0x1008	0	r / w
Rate Code	P - H	Power limit alarm value	Decimal	2	0. 0 - 9999	0x1009	9999	r / w
Reported 0		Decimal point and units	Decimal	2	Dot (0 - 3); 0: w; 1: kw; 2: mw	0x100A	0, 0	r / w
Police 0	P - DI	Power limit alarm back to the difference	Decimal	2	0. 0 - 9999	0x100B	0	r / w
3		Decimal point and units	Decimal	2	Dot (0 - 3); 0: w; 1: kw; 2: mw	0x100C	0, 0	r / w
6	PH - O	**Power limit alarm relay selection**	Decimal	2	0, 1, 2	0x100D	0	r / w
	P - L	Power lower limit value alarm	Decimal	2	0. 0 - 9999	0x100E	0	r / w
		Decimal point and units	Decimal	2	Dot (0 - 3); 0: w; 1: kw; 2: mw	0x100F	0, 0	r / w
	P - DI	Power back to the difference between the lower limit alarm	Decimal	2	0. 0 - 9999	0x1010	0	r / w
		Decimal point and units	Decimal	2	Dot (0 - 3); 0: w; 1: kw; 2: mw	0x1011	0, 0	r / w
	PL - O	**Power limit alarm relay selection**	Decimal	2	0, 1, 2	0x1012	0	r / w
	NULL	Air				0x1013		
	NULL	Air				0x1014		
	NULL	Air				0x1015		
	NULL	Air				0x1016		
	NULL	Air				0x1017		
Electricity Close	I - WS	Alarm only use selected	Decimal	2	**0: NO; 1: H; 2: L; 3: HL**	0x1018	0	r / w
	I - H	Current limit alarm value	Decimal	2	0. 0 - 9999	0x1019	9999	r / w
Flow Code		Decimal point and units	Decimal	2	Dot (0 - 3); 0: A; 1: kA;	0x101A	0, 0	r / w
Reported 0	I - DI	Current limit alarm back to the difference	Decimal	2	0. 0 - 9999	0x101B	0	r / w
		Decimal point and units	Decimal	2	Dot (0 - 3); 0: A; 1: kA;	0x101C	0, 0	r / w
Police 0	IH - O	**Current limit alarm relay selection**	Decimal	2	0, 1, 2	0x101D	0	r / w
3	I - L	Current limit alarm value	Decimal	2	0. 0 - 9999	0x101E	0	r / w
		Decimal point and units	Decimal	2	Dot (0 - 3); 0: A; 1: kA;	0x101F	0, 0	r / w
7	I - DI	Current limit alarm back to the difference	Decimal	2	0. 0 - 9999	0x1020	0	r / w
		Decimal point and units	Decimal	2	Dot (0 - 3); 0: A; 1: kA;	0x1021	0, 0	r / w
	IL - O	**Current limit alarm relay selection**	Decimal	2	0, 1, 2	0x1022	0	r / w
	NULL	Air				0x1023		
	NULL	Air				0x1024		
	NULL	Air				0x1025		
	NULL	Air				0x1026		
	NULL	Air				0x1027		
Electricity Close	U - WS	Alarm only use selected	Decimal	2	**0: NO; 1: H; 2: L; 3: HL**	0x1028	0	r / w
	U - H	Voltage limit alarm value	Decimal	2	0. 0 - 9999	0x1029	9999	r / w

Press Code		Decimal point and units	Decimal	2	Dot (0 - 3); 0: V; 1: kV;	0x102A	0, 0	r / w
	U - DI	Voltage limit alarm back to the difference	Decimal	2	0. 0 - 9999	0x102B	0	r / w
Reported 0		Decimal point and units	Decimal	2	Dot (0 - 3); 0: V; 1: kV;	0x102C	0, 0	r / w
	UH - O	**Voltage limit alarm relay selection**	Decimal	2	0, 1, 2	0x102D	0	r / w
Police 0		Voltage limit alarm value	Decimal	2	0. 0 - 9999	0x102E	0	r / w
3	U - L	Decimal point and units	Decimal	2	Dot (0 - 3); 0: V; 1: kV;	0x102F	0, 0	r / w
	U - DI	Voltage limit alarm back to the difference	Decimal	2	0. 0 - 9999	0x1030	0	r / w
8		Decimal point and units	Decimal	2	Dot (0 - 3); 0: V; 1: kV;	0x1031	0, 0	r / w
	UL - O	**Voltage limit alarm relay selection**	Decimal	2	0, 1, 2	0x1032	0	r / w
Power measurement		Voltage	Hex	2	0--65536	0x0000		'R'
		Decimal point and units	Decimal	2	Dot (0 - 3); 0: V; 1: kV;	0x0001		'R'
		Current	Hex	2	0--65536	0x0002		'R'
		Decimal point and units	Decimal	2	Dot (0 - 3); 0: A; 1: Ka;	0x0003		'R'
		Power	Hex	2	0 - 32767 (complement)	0x0004		'R'
		Decimal point and units	Decimal	2	Dot (0 - 3); 0: W; 1: kW; 2: MW	0x0005		'R'
		Switch 1						'W'
		Switch 2						'W'

MODBUS RTU communication protocol

Communication of data types and formats:

Information transmission is asynchronous, and in bytes. In the master and slave communication information is passed between the 10-bit word format:

Word format (serial data)	10-bit binary
Start bit	A
Data bits	8-bit
Parity bit	No
Stop bit	A

- **Communication data (information frame) format**

Data format	Address code	Function code	Data area	CRC check
Data length	1 byte	1 byte	N bytes	16-bit CRC (cycle redundancy code)

1. **Communication and information transfer process:**

When the communication command from the sending device (host) to the receiving device (slave), in line with the corresponding address codes to receive communications from the machine commands, function codes in accordance with relevant requirements and read the information, if the CRC is correct, then the implementation of the corresponding task, and then implementation of the results (data) back to the host. The information returned includes address code, function code, data and after the implementation of CRC. If the CRC error does not return any information.

1.1 address code:

Address code is a communication of information for each frame of the first byte (8 bits), from 0 to 255. This byte indicates the address set by the user from the machine will receive the information is sent by the host. Each slave must have a unique address code, address code and only in line with the response from the machine to send information back. When sending information back from the machine when the data are sent back to their starting address code. Host sends the address code that will be sent to the slave address, the address returned from the machine code that echo the slave address. The corresponding address code indicates that the information came from where.

1.2 Function code:

Communications and information transmitted for each frame is the second byte. ModBus communication protocol defined function code is 1 to 127. Beijing T & H Co., Ltd. HB series instruments / transmitters only to one part of the function code. As a host request, through the function code tells the slave what action should be implemented. Response as a slave, the slave returns the function code sent from the host with the function code to the same, and that has been the response from the host machine and related operations have been carried out.

Table 8.1 MODBUS function code part

Function code	Definitions	Operations (binary)
02	Read the digital input DI	Read all the way or path switch state input number (tele)
01	Read status output OUT	Read all the way or multi-channel digital output status data
03	Read register data	Read data from one or more registers
05	Write switching output OUT	Control all the way to relay "ON / OFF" output, remote control
06	Write single register	Write binary data to a single register
10	Write multiple registers	Multiple sets of binary data is written to multiple registers

1.3 data area:

Data area including the need to send the slave back to what information or perform any action. This information can be data (such as: digital input / output, analog input / output, registers, etc.), the reference address. For example, the host told by the function code 03 returned from a machine register values ??(including the start address register to be read and read register length), the returned data includes the data length and data register contents. Different from the machine, not the same address and data information (communication should be given information table).

HB Series meter / transmitter with Modbus communication protocol, the host (PLC, RTU, PC machine, DCS, etc.) the use of communication commands (function code 03), can be arbitrarily read the data register (the data table in Appendix). HB series meter / transmitter data register storage capacity up to several hundred (eg: current, voltage, power, power factor, etc.), and are 16-bit (2 bytes) of binary data, and high in the former ; a maximum number of registers can be read (both the quantity of electricity) is 50.

HB response command format is the slave address, function code, data area and the CRC. Data areas is two bytes, and the high front (except for electric energy).

4. Stationary time-

Requirements of the data bus to send data before the time that is still no data transmission time is greater than (5ms 9600 baud).

2. Introduction to **MODBUS** function code **(HB416PVI electrical parameters only support the following function code table)**

2.3 Function code "03": read multiple input register

For example: To read the address of the host 01, the starting address 017A of the three register data from the machine.

Data from the machine's address and data registers:

Register address	Register data (16 hex)	Corresponding to the power
0000	00C8	A-phase voltage
0001	00C9	B-phase voltage
0002	00CA	C-phase voltage

Host sends the message format:

Host sends	Bytes	Send a message	Remarks
Slave Address	1	01	01 sent to the address from the machine
Function code	1	03	Read register
Start address	2	0000	Starting at address 0000
Read data length	2	0003	Read three registers (a total of 6 bytes)
CRC code	2	05CB	CRC code calculated by the host

Returned from the machine response message format:

Response from machine	Bytes	The information returned	Remarks
Slave Address	1	01	From slave 01
Function code	1	03	Read register
Data length	1	01	Three registers (a total of 6 bytes)
The data register 1	2	00C8	The contents of the register address is 0000
The data register 2	2	00C9	The contents of the register address 0001
The data register 3	2	00CA	The contents of the register address 0002
CRC code	2	E60C	Calculated by the CRC code from the machine

Read the power situation is negative:

For example: host address 01 to read the power meter measurement

For example: To read the address of the host 01, the starting address 017A of the three register data from the machine.

Data from the machine's address and data registers:

Register address	Register data (16 hex)	Corresponding to the power
0004	99D1	Power value
0005	0003	Decimal point position

Host sends the message format:

Host sends	Bytes	Send a message	Remarks
Slave Address	1	01	01 sent to the address from the machine
Function code	1	03	Read register
Start address	2	0000	Starting at address 0000
Read data length	2	0002	Read two registers (4 bytes)
CRC code	2	85CA	CRC code calculated by the host

Returned from the machine response message format:

Response from machine	Bytes	The information returned	Remarks
Slave Address	1	01	From slave 01
Function code	1	03	Read register
Data length	1	04	Two registers (4 bytes)
The data register 1	2	99D1	The contents of the register address is 0000
The data register 2	2	0003	The contents of the register address 0001
CRC code	2	E60C	Calculated by the CRC code from the machine

2.4 Function code "05": write 1 way switch output ("Remote Control")

Example 1: switch output point OUT1, its current status as "points", the host to control the relay, "co."

Control command is: **"FF00"** to control the relay, "together";

"0000" for the control relay "points"

Host sends the message format:

Host sends	Bytes	Send a message	Remarks
Slave Address	1	01	01 sent to the address from the machine
Function code	1	05	Write switch output state
BIT-bit output	2	0000	The corresponding output relay BIT0 bit (OUT1)
Control commands	2	FF00	Relay outputs to control the "joint" status bit
CRC code	2	8C3A	CRC code calculated by the host

Returned from the machine response message format:

And the host sends the same message.

Example 2: switching output OUT2, its current status as "points", the host to control the relay, "co."

Control command is: **"FF00"** to control the relay, "together";

"0000" for the control relay "points"

Host sends	Bytes	Send a message	Remarks
Slave Address	1	01	01 sent to the address from the machine

Function code	1	05	Write switch output state
BIT-bit output	2	0001	The corresponding output relay BIT1 bit (OUT1)
Control commands	2	FF00	Relay outputs to control the "joint" status bit
CRC code	2	9C0A	CRC code calculated by the host

Returned from the machine response message format:

And the host sends the same message.

2.5 Function code "10": write multiple registers

Host use this function code to save data to multiple tables of data memory in HB to go. Modbus communication protocol refers to the register is 16 bits (ie 2 bytes), and high in the former. This HB is two bytes of memory.

For example: the host should 270F, 03E7 saved to address 1002, 1003 to register from the machine (from the machine address code 01).

Host sends the message format:

Host sends	Bytes	Send a message	Remarks
Slave Address	1	01	01 sent to the address from the machine
Function code	1	10	Write multiple registers
Start address	2	1002	To write the start address register
Save the data length	2	0002	Save the data word length
Save the data bytes	1	04	Save the data length in bytes (4 bytes)
Save data 1	2	270F	The data to be written to address 1002
2 to save data	2	03E7	The data to be written to address 1003
CRC code	2	C5BB	CRC code calculated by the host

Returned from the machine response message format:

Response from machine	Bytes	Send a message	Remarks
Slave Address	1	01	01 sent to the address from the machine
Function code	1	10	Write multiple registers
Start address	2	1002	To write the start address register
Save the data length	2	0002	Save the data word length
CRC code	2	E4C8	Calculated by the CRC code from the machine

3. Error check (CRC check):

Master or slave can check code to determine whether the right to receive information. As the electronic noise or some other interference, the information during transmission error sometimes occurs, the error check code (CRC) can verify that the host or slave data transmission process in the communication of information is wrong, wrong data can give up (regardless of is sending or receiving), which increases the system's security and efficiency.

MODBUS communication protocol of the CRC (cycle redundancy code) consists of 2 bytes, or 16-bit binary number. CRC code from the sending device (host), is placed in the rear send a message frame. Receiving the information device (slave) and then recalculate the information received by the CRC, the CRC is calculated by comparing the receiver to match, if they do not match, then the error.

- **CRC code is calculated as:**

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

2, the first 8-bit binary data (communication of information both the first byte of the frame) and 16-bit CRC register, or the lower 8 bits are different, the results put in the CRC register;

3, the contents of the CRC register one right (towards low) with 0 fill the highest position, and check out right after the bit;

4, if out of the bit is 0: Repeat Step 3 (again shifted to the right one); if out of place as 1: CRC register with the polynomial A001 (1010 0000 0000 0001) XOR;

5 Repeat steps 3 and 4, until the right 8 times, so that all of the 8-bit data were processed;

6 Repeat steps 2 through 5, the communication frame to the next byte of information processing;

7, the communication of information bytes of the frame all the steps above calculation is completed, the resulting 16-bit CRC register is the high and low bytes are exchanged;

8, the last to get the contents of the CRC register is: CRC code.