R6 system manual

System Modbus TCP Slave Description

Shenzhen Huacheng Industrial Control Co.

Catalog

1 Preface

This description introduces the Modbus TCP protocol of this system. This protocol is supported simultaneously with the [remote communication protocol] of this system.

1.1 Modification records

Category: A- Add M- Modify D- Delete

serial number	version number	category	modify the content	date	Revised by	audit
3	A2	A	1. Modify the document format	November 9, 2020	Cheng Guoxing	
2	A1	A	1. Modify the IO read and write instructions, 0 to 4 are general IO, 5 to 7 are M values, and 8 is EUIO 2. Add general output point for bitwise operation, M value function and EU output point, function code 0x5	November 4, 2020	Cheng Guoxing	
1	A0	A	1. First Edition Guidelines	November 4, 2020	Cheng Guoxing	

2 Brief description

1. Using the modbusTCP protocol, the host acts as a modbus slave;
2. The data in the example is in hexadecimal;
3. Data Format:

   (1) In bytes, the high byte is in front and the low byte is behind.
   (2) 16bit data: occupy one register, the high 8 bits are in front and the low 8 bits are in the back during transmission;
   (3) 32bit data: 2 registers are occupied, the high 16bit data is located in the low address, and the low 16bit data is located in the high address middle;

address	value
Addr0	bit31~bit16
Addr1	bit15~bit0

(4) For 64bit data: occupy 4 registers, the highest 16bit data is located at the low address, and the lowest 16bit data is located at the highest address;

address	value
Addr0	bit63~bit48
Addr1	bit47~bit32
Addr2	bit31~bit16
Addr3	bit15~bit0

4. Request APU example

	illustrate	size	example
MBAP header	transaction identifier Hi	1	0x15
	transaction identifier Lo	1	0x01
	protocol identifier	2	0x0000
	length	2	0x0006
	unit identifier	1	0xFF
MODBUS request	function code	1	0x03
	initial address	2	0x0005
	number of registers	2	0x0001

3 Address definition and operation requirements

3.1 Read register operation (0x03)

3.1.1 Read version number length

Request version number length:
(1) Address: 0x0000
(2) Number of registers: 1
(3) Example: `00 00 00 00 00 06 01 03 00 00 00 01`

	illustrate	size	example
MBAP header	transaction identifier Hi	1	00
	transaction identifier Lo	1	00
	protocol identifier	2	00 00
	length	2	00 06
	unit identifier	1	01
MODBUS request	function code	1	03
	initial address	2	00 00
	Number of registers	2	00 01

Response version number length:
Example: `00 00 00 00 00 05 01 03 02 00 29`
Explanation: The length of the version number in bytes is 0x29

	illustrate	size	Example
MBAP header	transaction identifier Hi	1	00
	transaction identifier Lo	1	00
	protocol identifier	2	00 00
	length	2	00 05
	unit identifier	1	01
MODBUS response	function code	1	03
	data bytes	1	02
	data	2	00 29

3.1.2 Read the version number

It is necessary to read the data length of the version number first, and then use this length to read the version number;
since the bit width of the modbus holding register is 16, when the length of the version number is singular, the value of the last register is valid for the upper 8 bits and the lower

8 bits. Fill with 0. The starting address is fixed at 0x01, and the number of registers read is calculated by this method: (version number bytes + 1)/2

Request to read version number:

(1) Start address: 0x00 01
(2) Number of registers: (version number bytes + 1)/2
Example: `00 00 00 00 00 06 01 03 00 01 00 15`

	illustrate	size	example
MBAP header	transaction identifier Hi	1	00
	transaction identifier Lo	1	00
	protocol identifier	2	00 00
	length	2	00 06
	unit identifier	1	01
MODBUS request	function code	1	03
	initial address	2	00 01
	Number of registers	2	00 15

Response version number:

Example:

```
0x00 0x00 0x00 0x00 0x00 0x2c 0x01 0x03 0x2a
0x41 0x4d 0x38 0x2d 0x51 0x43 0x2d 0x52 0x58
0x45 0x2d 0x37 0x2e 0x38 0x2e 0x30 0x32 0x2d
0x62 0x61 0x74 0x65 0x37 0x5f 0x46 0x41 0x4b
0x45 0x5f 0x54 0x47 0x46 0x5f 0x45 0x4e 0x43
0x4f 0x44 0x45 0x52 0x00
```

Explanation: After converting each byte of the data area into characters, you can get the version: *"AM8-QC-RXE-7.8.02-bate7_FAKE_TGF_ENCODER"*

	illustrate	size	example
MBAP header	transaction identifier Hi	1	00
	transaction identifier Lo	1	00
	protocol identifier	2	00 00
	length	2	00 2c
	unit identifier	1	01
MODBUS response	function code	1	03

	illustrate	size	example
	data bytes	1	2a
	data	2a	0x41 0x4d 0x38 0x2d 0x51 0x43 0x2d 0x52 0x58 0x45 0x2d 0x37 0x2e 0x38 0x2e 0x30 0x32 0x2d 0x62 0x61 0x74 0x65 0x37 0x5f 0x46 0x41 0x4b 0x45 0x5f 0x54 0x47 0x46 0x5f 0x45 0x4e 0x43 0x4f 0x44 0x45 0x52 0x00

3.1.3 Read Counter List

First read the number of counters, and then request the corresponding counter ID according to the number of counters. The number of counters occupies one register, and each counter ID occupies 2 registers; If the number of counters is more than the number of requests, the number of requests will be intercepted to respond. If the actual number is less than the number of requests, the rest will be filled with 0xFF.

Number of request counters:

Read the current number of counters, occupying only one register, so the starting address and number of registers are fixed; Example: `00 00 00 00 00 06 01 03 00 82 00 01`

	illustrate	size	example
MBAP header	transaction identifier Hi	1	00
	transaction identifier Lo	1	00
	protocol identifier	2	00 00
	length	2	00 06
	unit identifier	1	01
MODBUS request	function code	1	03
	initial address	2	00 82
	Number of registers	2	00 01

Number of response counters:

Example: `00 00 00 00 00 0x05 0x01 0x03 0x02 0x00 0x02` Explanation: The number of counters read is 2;

	illustrate	size	example
MBAP header	transaction identifier Hi	1	00
	transaction identifier Lo	1	00
	protocol identifier	2	00 00
	length	2	00 05
	unit identifier	1	01

	illustrate	size	example
MODBUS response	function code	1	03
	data bytes	1	02
	data	2	0x00 0x02 (counter number)

Request a list of current counters:

(1) Address range: 0x0083 ~ 0x0882, the effective address is determined according to the number of existing counters (0x83 + number of registers × 2), the starting address must be the starting address of the target ID, such as 0x0083 is the start of the ID of the 0th counter Address, 0x0085 is the ID start address of the first counter;
(2) Number of registers to read: Since one counter ID occupies 2 registers, and the lower 16 bits are stored in a small address, the number of registers requested needs to be a multiple of 2, such as 2, 4, 6;
(3) Example 1: `00 00 00 00 00 06 01 03 00 83 00 04`
Explanation: Request to read 2 consecutive counter IDs starting from the 0th counter;

	illustrate	size	example
MBAP header	transaction identifier Hi	1	00
	transaction identifier Lo	1	00
	protocol identifier	2	00 00
	length	2	00 06
	unit identifier	1	01
MODBUS request	function code	1	03
	start address 2	00 83	
	Number of registers	2	00 04

(4) Example 2: `00 00 00 00 00 06 01 03 00 85 00 04`
Explanation: Request to read two consecutive counter IDs starting from the first counter;

Response current counter list:

Example: `00 00 00 00 00 0b 01 03 08 00 00 00 00 00 00 00 01`
Explanation: 2 counter IDs are read, 0 and 1;

	illustrate	size	example
MBAP header	transaction identifier Hi	1	00
	transaction identifier Lo	1	00
	protocol identifier	2	00 00
	length	2	00 0b