

LIYAN PROGRAMMABLE LOGIC CONTROLLER

Ex Series

Jn Series

Modbus USER'S MANUAL

◆1 Introduction

1.1 The communication architecture of Modbus is as follows

| | | | |
|------------|---------|------|-----------|
| Station No | Command | Data | check sum |
|------------|---------|------|-----------|

If the station number is set to 0, it means that the master station broadcasts to all the slave stations

If the station number is set to 1~247, it means that the master station sends communication data to the designated slave station

1.2 The command codes supported by Ex series and Jn series are as follows

01H : **Bits Read**

03H : **Word Read**

05H : **Bit write**

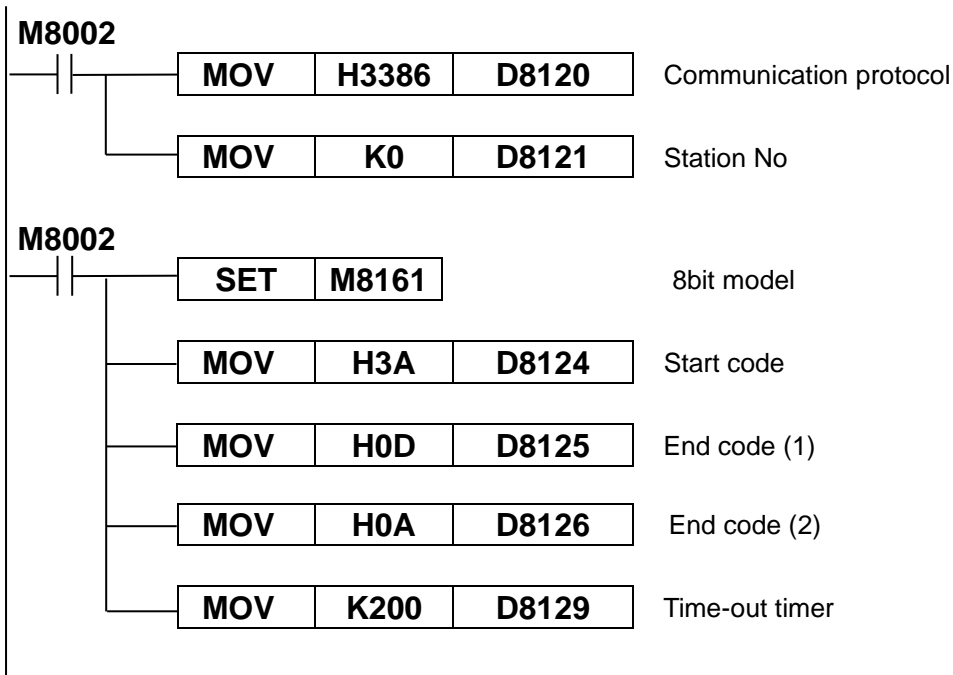
06H : **Word write**

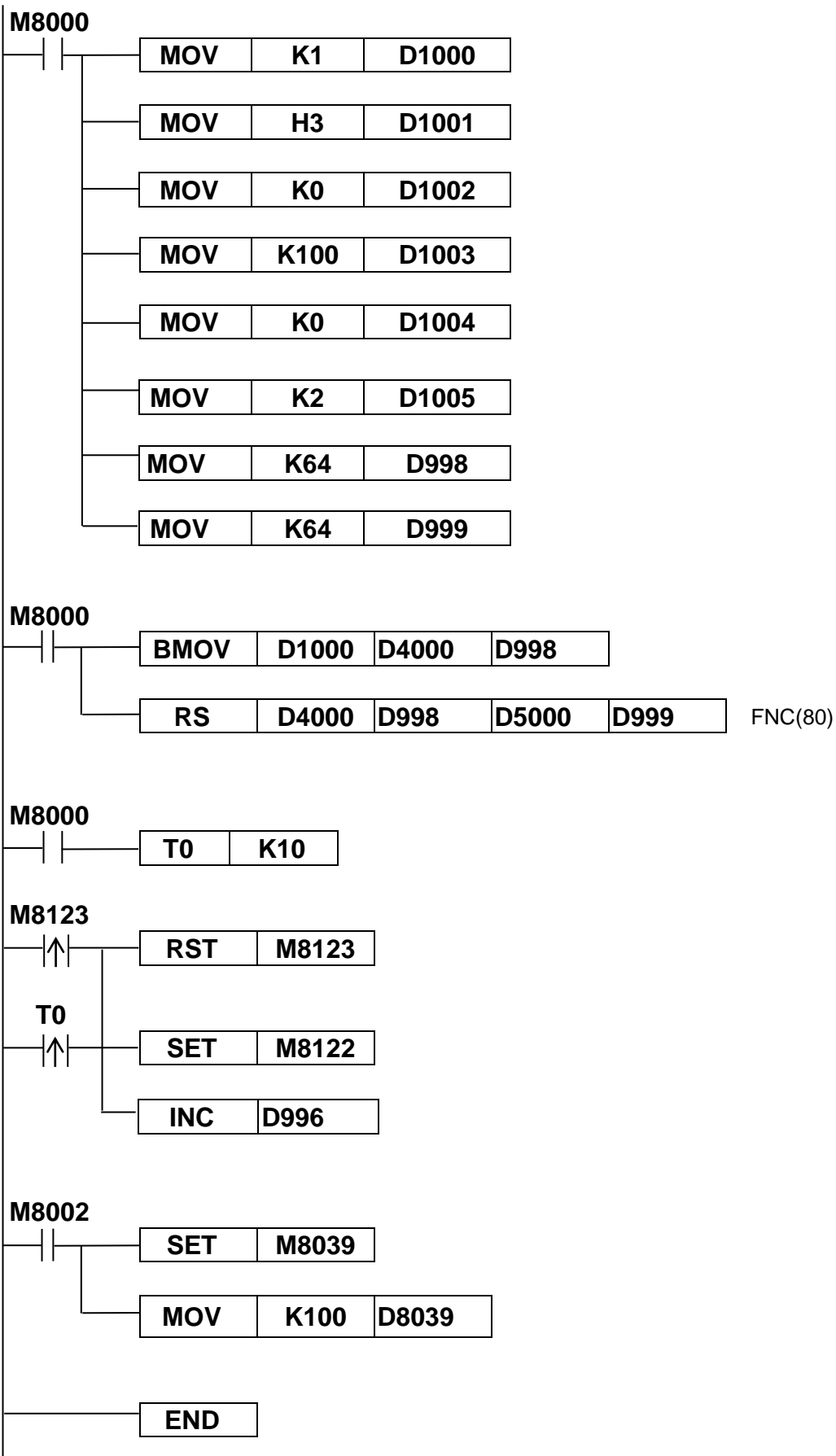
0FH : **Multiple bit write**

10H : **Multiple word write**

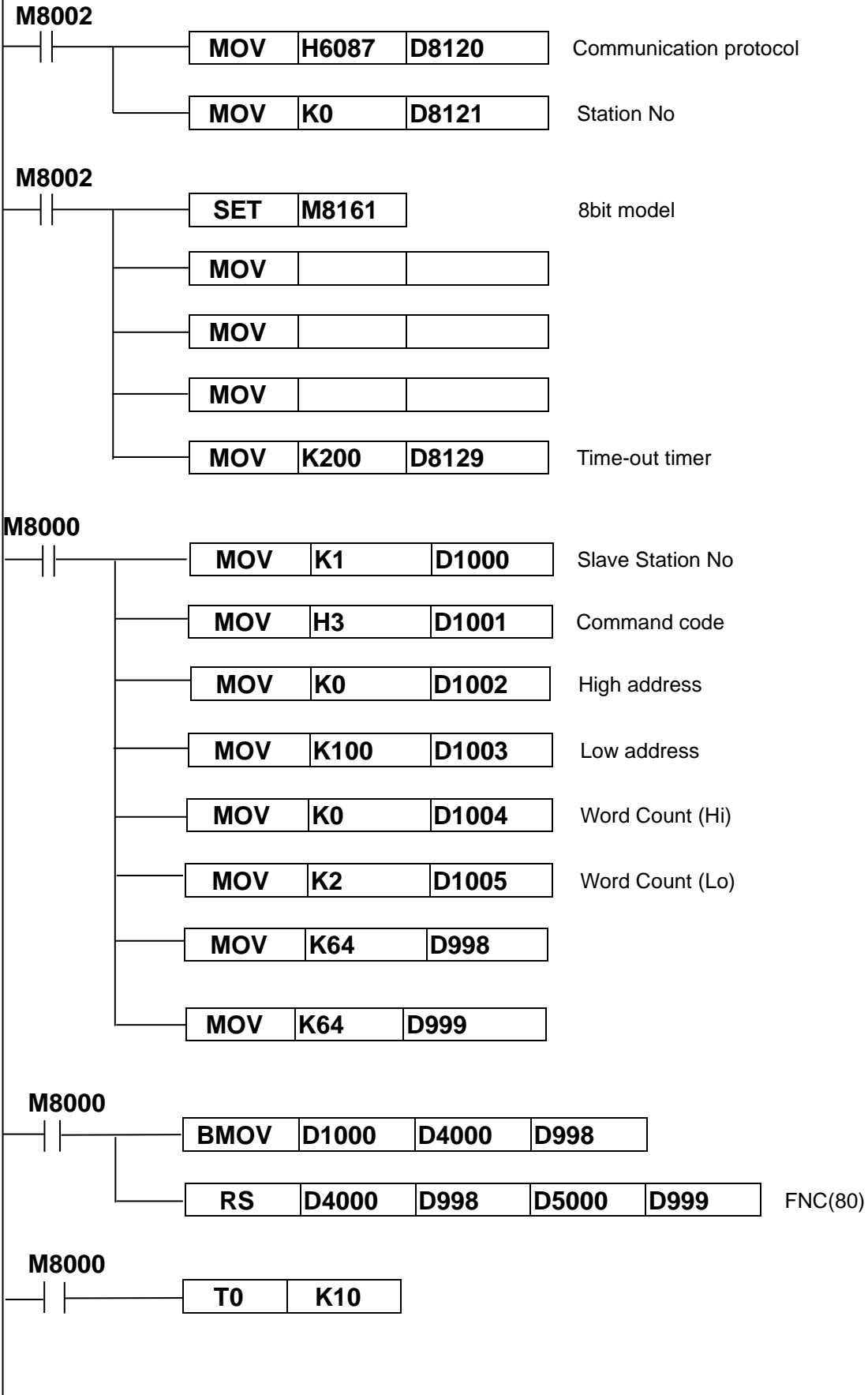
1.2 MODBUS is not the original communication protocol of ExSeries, JnSeries To use the Modbus communication protocol, you need to write the FNC(80)RS

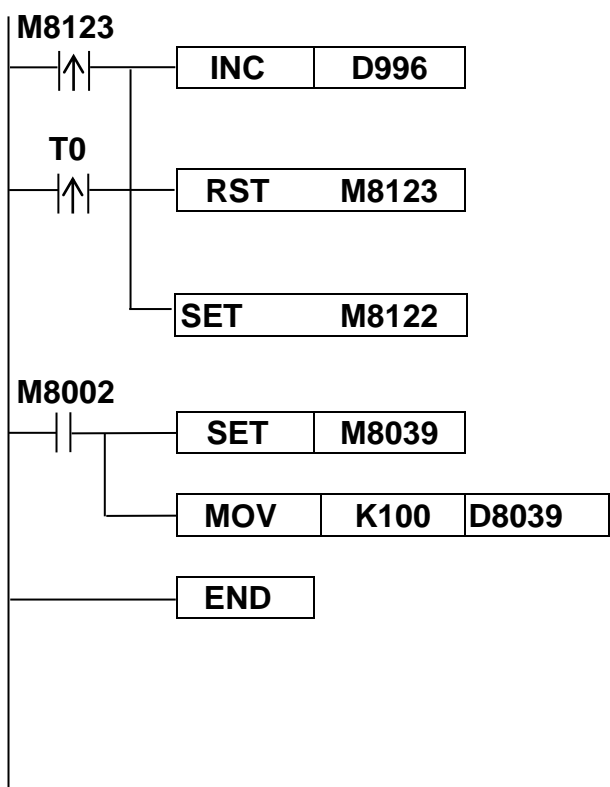
◆1.2.1 Master program by Ascii Mode



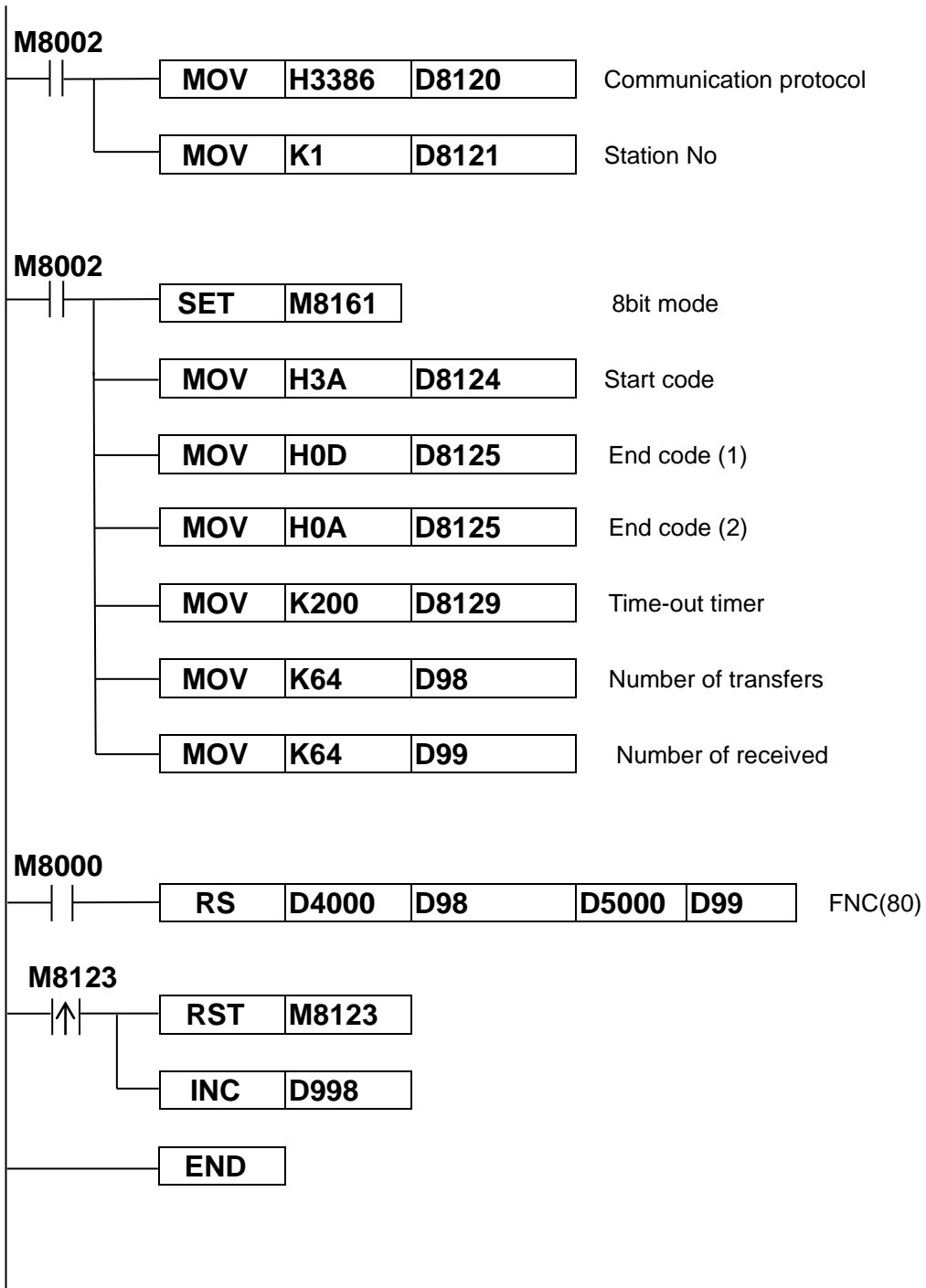


◆ 1.2.2 Master program by RTU Mode

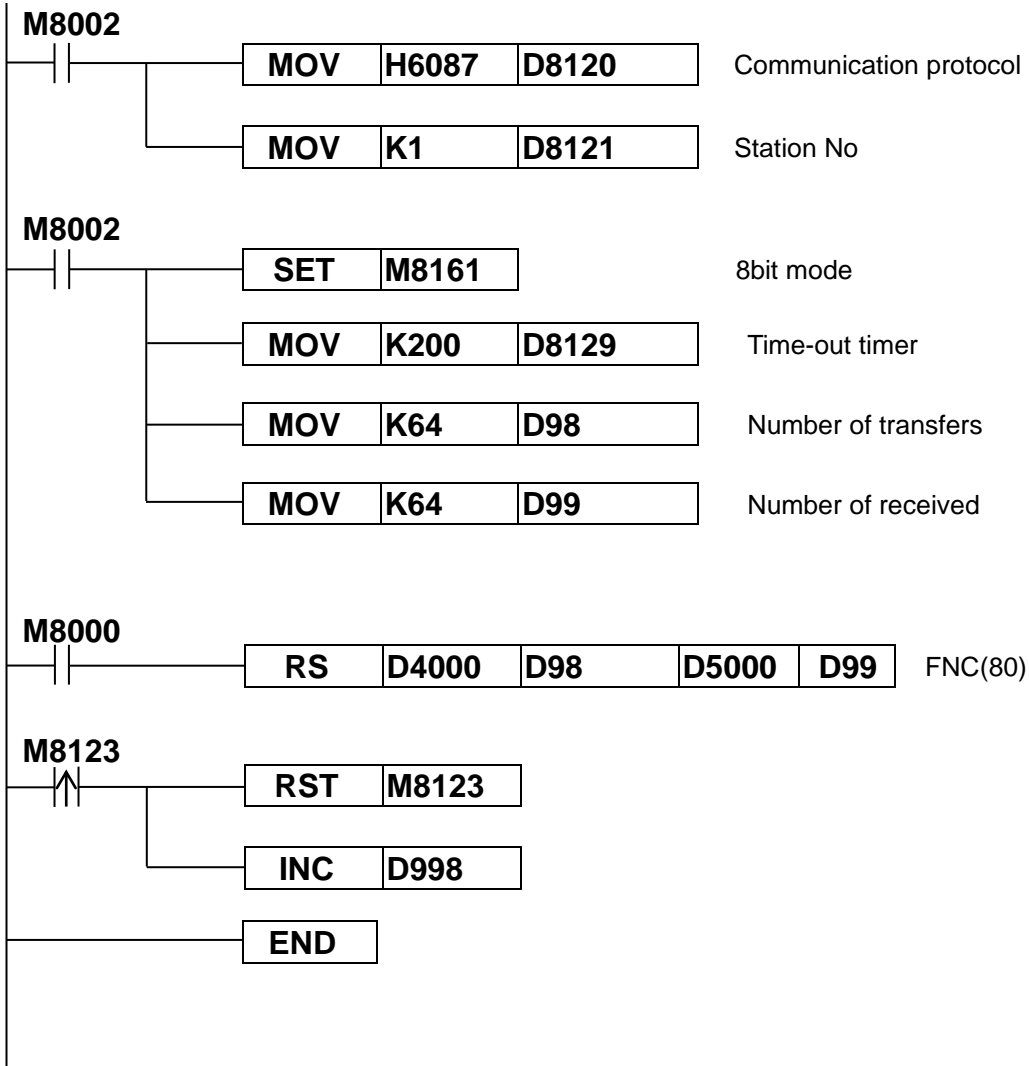




◆ 1.2.3 Slave Station by Ascii Mode



◆ 1.2.4 Slave Station by RTU Mode



2.1 Ex Series and Jn Series Modbus List of component addresses

Bit:

| MODBUS component addresses | | Ex component |
|----------------------------|---------------|--------------|
| only read | Read/Write | |
| 0x0000-0x1DFF | 0x0000-0x0BFF | M0-M3071 |
| 0x1E00-0x1FFF | 0x1E00-0x1EFF | M8000-M8255 |
| 0x2000-0x2FFF | 0x2000-0x23E7 | S0-S999 |
| 0x3000-0x31FF | 0x3000-0x30FF | TS0-TS255 |
| 0x3200-0x32FF | 0x3200-0x32FF | CS0-CS255 |
| 0x3300-0x33FF | 0x3300-0x337F | Y0-Y177 |
| 0x3400-0x347F | - | X0-X177 |

Word:

| MODBUS component addresses | | Ex component |
|----------------------------|---------------|---------------|
| only read | Read/Write | |
| 0x0000-0x1F3F | 0x0000-0x1F3F | D0-D7999 |
| 0x1F40-0x213F | 0x1F40-0x203F | D8000-D8255 |
| 0x2140-0xA13F | 0x2140-0xA13F | Reserved |
| 0xA140-0xA33F | 0xA140-0xA23F | TN0-TN255 |
| 0xA340-0xA407 | 0xA340-0xA407 | CN0-CN199 |
| 0xA408-0xA477 | 0xA408-0xA477 | CN200-CN255*1 |
| 0xA478-0xA537 | 0xA478-0xA537 | M0-M3071 |
| 0xA658-0xA667 | 0xA658-0xA667 | M8000-M8255 |
| 0xA678-0xA6B7 | 0xA678-0xA6B7 | S0-S999 |
| 0xA778-0xA787 | 0xA778-0xA787 | TS0-TS255 |
| 0xA798-0xA797 | 0xA798-0xA7A7 | CS0-CS255 |
| 0xA7A8-0xA7AF | 0xA7A8-0xA7AF | Y0-Y177 |
| 0xA7B8-0xA7BF | - | X0-X177 |

C200~255 for 32bit °

◆2.2 Command code

| Code | Name | Description |
|------|--------------------------|-----------------|
| 01 | Read Coil Status | X, Y, M,S, T, C |
| 03 | Read Holding Registers | T, C, D |
| 05 | Force Single Coil | Y, M,S, T, C |
| 06 | Preset Single Register | T, C, D |
| 15 | Force Multiple Coils | Y, M,S, T, C |
| 16 | Preset Multiple Register | T, C, D |

2.3 Modbus Ascii Mode Command code

◆2.3.1 Command code 01H: Bits Read

< Command message > ex: Read X00~X37 from station number 01

| | | | | | | | | | | | | | | | | |
|-----|------|-----|---------------|-----------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|------|------|
| 3Ah | 01h | 01h | 34h | 00h | 00h | 20h | AAh | 0Dh | 0Ah | | | | | | | |
| ‘:’ | ‘0’ | ‘1’ | ‘0’ | ‘1’ | ‘3’ | ‘4’ | ‘0’ | ‘0’ | ‘0’ | ‘0’ | ‘2’ | ‘0’ | ‘A’ | ‘A’ | ‘CR’ | ‘LF’ |
| STX | StNo | Cmd | Start Address | Bit Count | LRC | END | | | | | | | | | | |

CHK (check sum):

01H+01H+34H+00H+00H+20H=56H

56H Second complement :FFH-56H+1=AAH

< Respond > ex: X00 status is ON The other all OFF

| | | | | | | | | | | | | | | | | | | |
|-----|------|-----|------------|--------------|--------------|--------------|--------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|----------|----------|
| STX | StNo | Cmd | Byte Count | X07~X00 Data | X17~X10 Data | X27~X20 Data | X37~X30 Data | LRC | END | | | | | | | | | |
| ‘:’ | ‘0’ | ‘1’ | ‘0’ | ‘1’ | ‘0’ | ‘4’ | ‘0’ | ‘1’ | ‘0’ | ‘0’ | ‘0’ | ‘0’ | ‘0’ | ‘0’ | ‘F’ | ‘8’ | CR (0Dh) | LF (0Ah) |
| 3Ah | 01h | 01h | 04h | 01h | 00h | 00h | 00h | F8h | 0Dh | 0Ah | | | | | | | | |

◆CHK(check sum)

Ascii use LRC(Longitudinal Redundancy Check) ◦

◆2.3.2 Command code 02H: (Skip)

◆2.3.3 Command code 03H : Word Read

< Command message > ex: Read two words from D0 of station number 01

| | | | | | | | | | | | | | | | | |
|-----|------|-----|-----|-----|---------------|-----|-----|-----|------------|-----|-----|-----|-----|-----|-----|-----|
| 3Ah | 01h | | 03h | | 00h | | 00h | | 00h | | 02h | | FAh | | 0Dh | 0Ah |
| ‘.’ | ‘0’ | ‘1’ | ‘0’ | ‘3’ | ‘0’ | ‘0’ | ‘0’ | ‘0’ | ‘0’ | ‘0’ | ‘0’ | ‘2’ | ‘F’ | ‘A’ | CR | LF |
| STX | StNo | | Cmd | | Start Address | | | | Word Count | | LRC | | END | | | |

CHK (check sum):

$$01H+03H+00H+00H+00H+02H=06H$$

$$06H \text{ Second complement} : FFH-06H+1=FAH$$

< Respond > ex: Value of D0 is 02 , Value of D1 is 08

| | | | | | | | | | | | | | | | | | | | |
|-----|------|-----|-----|-----|------------|-----|-----|-----|----------|-----|----------|-----|-----|-----|-----|-----|----------|----------|-----|
| STX | StNo | | Cmd | | Byte Count | | | | D00 Data | | D01 Data | | LRC | | END | | | | |
| ‘.’ | ‘0’ | ‘1’ | ‘0’ | ‘3’ | ‘0’ | ‘4’ | ‘0’ | ‘0’ | ‘0’ | ‘2’ | ‘0’ | ‘0’ | ‘0’ | ‘8’ | ‘E’ | ‘E’ | CR (0Dh) | LF (0Ah) | |
| 3Ah | 01h | | 03h | | 04h | | | | 00h | | 02h | | 00h | | 08h | | EEh | 0Dh | 0Ah |

◆2.3.4 Command code 04H: (Skip)

◆2.3.5 Command code 05H : Bit write

< Command message > ex: Write station number 01 Y4 to ON

| | | | | | | | | | | | | | | | | |
|-----|-----|-----|-----|-----|---------|-----|-----|-----|------|-----|-----|-----|-----|-----|-----|-----|
| 3Ah | 01h | | 05h | | 33h | | 04h | | FFh | | 00h | | C4h | | 0Dh | 0Ah |
| ‘.’ | ‘0’ | ‘1’ | ‘0’ | ‘5’ | ‘3’ | ‘3’ | ‘0’ | ‘4’ | ‘F’ | ‘F’ | ‘0’ | ‘0’ | ‘C’ | ‘4’ | CR | LF |
| STX | ADR | | CMD | | Address | | | | Data | | LRC | | END | | | |

Note: ON when the data is FF00, OFF when the data is 0000

< Respond > ex: Return the same value

| | | | | | | | | | | | | | | | | | |
|-----|------|-----|-----|-----|---------|-----|-----|-----|------|-----|-----|-----|-----|-----|------|------|-----|
| STX | StNo | | CMD | | Address | | | | Data | | LRC | | END | | | | |
| ‘.’ | ‘0’ | ‘1’ | ‘0’ | ‘5’ | ‘3’ | ‘3’ | ‘0’ | ‘4’ | ‘F’ | ‘F’ | ‘0’ | ‘0’ | ‘C’ | ‘4’ | ‘CR’ | ‘LF’ | |
| 3Ah | 01h | | 05h | | 33h | | | | 04h | | FFh | | 00h | | C4h | 0Dh | 0Ah |

◆2.3.6 Command code 06H: Word write

< Command message > ex: Write data 0084H to D04 of station number 01

| | | | | | | | | | | | | | | | | |
|-----|------|-----|-----|-----|---------------|-----|-----|-----|--------------|-----|-----|-----|-----|-----|-----|-----|
| 3Ah | 01h | | 06h | | 00h | | 04h | | 00h | | 84h | | 71h | | 0Dh | 0Ah |
| ‘.’ | ‘0’ | ‘1’ | ‘0’ | ‘6’ | ‘0’ | ‘0’ | ‘0’ | ‘4’ | ‘0’ | ‘0’ | ‘8’ | ‘4’ | ‘7’ | ‘1’ | CR | LF |
| STX | StNo | | Cmd | | Start Address | | | | Data Content | | LRC | | END | | | |

< Respond > ex: Return the same value

| | | | | | | | | | | | | | | | | | |
|-----|------|-----|-----|-----|---------------|-----|-----|-----|--------------|-----|-----|-----|-----|-----|------|------|-----|
| STX | StNo | | Cmd | | Start Address | | | | Data Content | | LRC | | END | | | | |
| ‘.’ | ‘0’ | ‘1’ | ‘0’ | ‘6’ | ‘0’ | ‘0’ | ‘0’ | ‘4’ | ‘0’ | ‘0’ | ‘8’ | ‘4’ | ‘7’ | ‘1’ | ‘CR’ | ‘LF’ | |
| 3Ah | 01h | | 06h | | 00h | | | | 04h | | 00h | | 84h | | 71h | 0Dh | 0Ah |

◆2.3.7 Command code 07H: (Skip)

◆2.3.8 Command code 08H: (Skip)

◆2.3.9 Command code 0FH : Multiple bit write

< Command message > ex: Force Y07~Y00=01100101, Y13~Y10=0111

| | | | | | | | | | | | | |
|-----|------------|------------|---------------|------------|------------|--------------|--------------|------------|------------|------------|-----|-----|
| 3Ah | 01h | 0Fh | 33h | 00h | 00h | 0Ch | 02h | 65h | 07h | 43h | 0Dh | 0Ah |
| ‘:’ | ‘0’ ‘1’ | ‘0’ ‘F’ | ‘3’ ‘3’ | ‘0’ ‘0’ | ‘0’ ‘0’ | ‘0’ ‘C’ | ‘0’ ‘2’ | ‘6’ ‘5’ | ‘0’ ‘7’ | ‘4’ ‘3’ | CR | LF |
| STX | StNo | Cmd | Start Address | Quantity | Byte CNT | Data Y07~Y00 | Data Y17~Y10 | LRC | END | | | |

< Respond >

| | | | | | | | | | |
|-----|------------|------------|---------------|------------|--------------------------|----------------------------|-----|-----|-----|
| STX | StNo | Cmd | Start Address | Quantity | LRC | END | | | |
| ‘:’ | ‘0’ ‘1’ | ‘0’ ‘F’ | ‘3’ ‘3’ | ‘0’ ‘0’ | ‘0’ ‘0’ ‘0’ ‘C’ | ‘B’ ‘1’ ‘CR’ ‘LF’ | | | |
| 3Ah | 01h | 0Fh | 00h | 00h | 00h | 0Ch | B1h | 0Dh | 0Ah |

◆2.3.10 Command code 10H : Multiple word write

< Command message > ex: Write data 8765H and 4321H into D05 and D04 of station number 01

| | | | | | | |
|-----|------------|------------|---------------|------------|------------|------------|
| 3Ah | 01h | 10h | 00h | 04h | 00h | 02h |
| ‘:’ | ‘0’ ‘1’ | ‘1’ ‘0’ | ‘0’ ‘0’ | ‘0’ ‘4’ | ‘0’ ‘0’ | ‘0’ ‘2’ |
| STX | StNo | Cmd | Start Address | Quantity | | |

| | | | | | | | |
|------------|--------------------|--------------------|------------|------------|------------|-----|-----|
| 04h | 43h | 21h | 87h | 65h | 95h | 0Dh | 0Ah |
| ‘0’ ‘4’ | ‘4’ ‘3’ | ‘2’ ‘1’ | ‘8’ ‘7’ | ‘6’ ‘5’ | ‘9’ ‘5’ | CR | LF |
| Byte Count | The content of D04 | The content of D05 | LRC | END | | | |

< Respond >

| | | | | | | | | | |
|-----|------------|------------|---------------|------------|--------------------------|----------------------------|-----|-----|-----|
| STX | StNo | Cmd | Start Address | Quantity | LRC | END | | | |
| ‘:’ | ‘0’ ‘1’ | ‘1’ ‘0’ | ‘0’ ‘0’ | ‘0’ ‘4’ | ‘0’ ‘0’ ‘0’ ‘2’ | ‘E’ ‘9’ ‘CR’ ‘LF’ | | | |
| 3Ah | 01h | 10h | 00h | 04h | 00h | 02h | E9h | 0Dh | 0Ah |

◆2.3.10 Exception response

In the exception response, the most significant bit of the original command code is set to 1. The following is the response of communication command 06H and exception code 02H

< Command message >

| | | | | | | | | | | | | | |
|-----|------|-----|---------------|---------------------|-----|-----|-----|-----|-----|-----|-----|----|----|
| 3Ah | 01h | 06h | C0h | 00h | 00h | 00h | 39h | 0Dh | 0Ah | | | | |
| ‘.’ | ‘0’ | ‘1’ | ‘0’ | ‘6’ | ‘C’ | ‘0’ | ‘0’ | ‘0’ | ‘0’ | ‘3’ | ‘9’ | CR | LF |
| STX | StNo | Cmd | Start Address | The content of data | | | | LRC | END | | | | |

< Respond >

| | | | | | | | | | | |
|-----|------|-----|----------------|-----|-----|-----|-----|-----|------|------|
| STX | StNo | Cmd | Exception code | LRC | END | | | | | |
| ‘.’ | ‘0’ | ‘1’ | ‘8’ | ‘6’ | ‘0’ | ‘2’ | ‘7’ | ‘7’ | ‘CR’ | ‘LF’ |
| 3Ah | 01h | 86h | 02h | 77h | 0Dh | 0Ah | | | | |

◆Exception code

| | |
|----|--|
| 01 | Illegal command code |
| 02 | Illegal data address |
| 03 | Illegal data value |
| 09 | Check sum Error |
| 13 | Receive buffer overflow |
| 14 | Defective receive character (Ascii only) |
| 15 | Parity error; overrun error; framing error |

2.4 Modbus RTU Mode

◆2.4.1 Command code 01H: Bits Read

< Command message > ex: Read X00~X37 from station number 01

| | 01h | 01h | 34h | 00h | 00h | 20h | 33h | E2h | |
|----------------------------|------|-----|------------------|-----|--------------|-----|------------|-------------|----------------------------|
| STOP 10ms or more | StNo | Cmd | Start Address | | Bit Count | | CRC LOW | CRC High | STOP 10ms or more |

< Respond > ex: X00 status is ON The others all OFF

| STOP 10ms or more | StNo | Cmd | Byte Count | Data X07~X00 | Data X17~X10 | Data X27~X20 | Data X37~X30 | CRC Low | CRC High | STOP 10ms or more |
|----------------------------|------|-----|---------------|-----------------|-----------------|-----------------|-----------------|------------|-------------|----------------------------|
| | 01h | 01h | 04h | 01h | 00h | 00h | 00h | FAh | 2Dh | |

Note:

RTU mode uses CRC (Cyclical Redundancy Check) error detection value, the calculation method is as follows:

- (1) Load the 16bit register (CRC register) with the content of FFFFH.
- (2) Perform Exclusive OR operation on the first byte of the command message and the low byte of 16bit, and store the result back to the CRC register.
- (3) Shift the content of the CRC temporary storage to the right by 1 bit, and fill in the highest bit with 0.
- (4) Check the value of the lowest bit of the CRC register, if it is 0, repeat (3), if it is not 0, perform an Exclusive OR operation between the CRC register and A001H, and store the result back to the CRC register.
- (5) Repeat (3) and (4) until the content of the CRC register is shifted to the right by 8 bits, at which time the byte has been processed.
- (6) Repeat (2) to (5) for the next command message byte until all the bytes are processed, and the content of the CRC register is the error value.
- (7) When transmitting the CRC value, the low bit is transmitted first, and then the high bit value.

◆2.4.2 Command code 02H: (Skip)

◆2.4.3 Command code 03H : Word Read

< Command message > ex: Read two words from D04 of station number 01

| | | | | | | | | | |
|----------------------------|------|-----|------------------|-----|---------------|-----|------------|-------------|----------------------------|
| | 01h | 03h | 00h | 04h | 00h | 02h | 85h | CAh | |
| STOP 10ms or more | StNo | Cmd | Start Address | | Word Count | | CRC Low | CRC High | STOP 10ms or more |

< Respond > ex: D4's data is 123 D5's data is 789

| | | | | | | | | | | |
|----------------------------|------|-----|---------------|-------------|-----|-------------|-----|------------|-------------|----------------------------|
| STOP 10ms or more | StNo | Cmd | Byte Count | D04 Data | | D05 Data | | CRC Low | CRC High | STOP 10ms or more |
| | 01h | 03h | 04h | 01h | 23h | 07h | 89h | C9h | 93h | |

◆2.4.4 Command code 04H: (Skip)

◆2.4.5 Command code 05H : Bit write

< Command message > ex: Write station number 01 Y3 to ON

| | | | | | | | | | |
|----------------------------|------|-----|---------|-----|------|-----|------------|-------------|----------------------------|
| | 01h | 05h | 33h | 03h | FFh | 00h | 73h | 7Eh | |
| STOP 10ms or more | StNo | Cmd | Address | | Data | | CRC Low | CRC High | STOP 10ms or more |

Note: ON when the data is FF00, OFF when the data is 0000

< Respond >

| | | | | | | | | | |
|----------------------------|-----|-----|---------|-----|------|-----|------------|-------------|----------------------------|
| STOP 10ms or more | ADR | CMD | Address | | Data | | CRC Low | CRC High | STOP 10ms or more |
| | 01h | 05h | 33h | 03h | FFh | 00h | 73h | 7Eh | |

◆2.4.6 Command code 06H: Word write

< Command message > ex: Write data 0084H to D04 of station number 01

| | | | | | | | | | |
|----------------------------|------|-----|---------|-----|------|-----|------------|-------------|----------------------------|
| | 01h | 06h | 00h | 04h | 00h | 84h | C8h | 68h | |
| STOP 10ms or more | StNo | Cmd | Address | | Data | | CRC Low | CRC High | STOP 10ms or more |

< Respond >

| | | | | | | | | | |
|----------------------------|------|-----|---------|-----|------|-----|------------|-------------|----------------------------|
| STOP 10ms or more | StNo | Cmd | Address | | Data | | CRC Low | CRC High | STOP 10ms or more |
| | 01h | 03h | 00h | 04h | 00h | 84h | C8h | 68h | |

◆2.4.7 Command code 07H: (Skip)

◆2.4.8 Command code 08H: (Skip)

◆2.4.9 Command code 0FH : Multiple bit write

< Command message > ex: Force Y07~Y00=01100101, Y13~Y10=0111

| | | | | | | | | | | | | |
|---------------------------|------|-----|------------------|-----|----------|-----|---------------|--------------------|--------------------|------------|-------------|----------------------------|
| | 01h | 0Fh | 33h | 00h | 00h | 0Ch | 02h | 65h | 07h | 8Ch | 21h | |
| STOP 10m or more | StNo | Cmd | Start Address | | Quantity | | Byte Count | Data of Y07~Y00 | Data of Y17~Y10 | CRC Low | CRC High | STOP 10ms or more |

< Respond >

| | | | | | | | | | |
|----------------------------|------|-----|------------------|-----|----------|-----|------------|-------------|----------------------------|
| STOP 10ms or more | StNo | Cmd | Start Address | | Quantity | | CRC Low | CRC High | STOP 10ms or more |
| | 01h | 0Fh | 33h | 00h | 00h | 0Ch | 5Ah | 8Ah | |

◆2.4.10 Command code 10H : Multiple word write

< Command message > ex: Write data 8765H and 4321H into D05 and D04 of station number 01

| | | | | | | | | | | | | | | |
|----------------------------|------|-----|------------------|-----|----------|-----|---------------|----------------------|-----|----------------------|-----|------------|-------------|----------------------------|
| | 01h | 10h | 00h | 04h | 00h | 02h | 04h | 43h | 21h | 87h | 65h | 14h | 09h | |
| STOP 10ms or more | StNo | Cmd | Start Address | | Quantity | | Byte Count | Content of D04 | | Content of D05 | | CHC Low | CRC High | STOP 10ms or more |

< Respond >

| | | | | | | | | | |
|----------------------------|------|-----|------------------|-----|----------|-----|------------|-------------|----------------------------|
| STOP 10ms or more | StNo | Cmd | Start Address | | Quantity | | CRC Low | CRC High | STOP 10ms or more |
| | 01h | 10h | 00h | 04h | 00h | 02h | 00h | 09h | |

◆2.4.10 Exception response

In the exception response, the most significant bit of the original command code is set to 1. The following is the response of communication command 06H and exception code 02H

< Command message >

| | 01h | 06h | C0h | 00h | 00h | 02h | 34h | 0Bh | |
|----------------------------|------|-----|------------------|-----|---------------|-----|------------|-------------|----------------------------|
| STOP 10ms or more | StNo | Cmd | Start Address | | Word Count | | CRC Low | CRC High | STOP 10ms or more |

< Respond >

| | | | | | | |
|----------------------------|------|-----|---------------|------------|-------------|----------------------------|
| STOP 10ms or more | StNo | Cmd | Error Code | CRC Low | CRC High | STOP 10ms or more |
| | 01h | 86h | 02h | C3h | A1h | |

◆ Exception code

| | |
|----|--|
| 01 | Illegal command code |
| 02 | Illegal data address |
| 03 | Illegal data value |
| 09 | Check sum Error |
| 13 | Receive buffer overflow |
| 14 | Defective receive character (Ascii only) |
| 15 | Parity error; overrun error; framing error |

3. Master station specifications

This chapter describes the functions supported by the Ex series MODBUS ADP master station.

◆3.1 MODBUS Master command list

| Instruction code | Instruction Secondary code | Command name | Detail |
|------------------|----------------------------|--|--|
| 0×01 | | Bit read | Bit signal readout (R/W) |
| 0×02 | | (Skip) | Bit signal readout (R/O) |
| 0×03 | | Register readout | 16-bit register readout (R/W) |
| 0×04 | | (Skip) | 16-bit register readout R/O) |
| 0×05 | | Single bit write | Bit signal write |
| 0×06 | | Write to a single register | Single 16-bit register write |
| 0×07 | | (Skip) Status readout | 1 byte data read |
| 0×08 | 0×00 | (Skip) Return of the question | Loop back test |
| | 0×01 | (Skip) Restart of communication settings | Communication restart/remote communication reset |
| | 0×02 | (Skip) Return of the diagnostic register | Specified 16-bit register readout |
| | 0×03 | (Skip) ASCII end code changes | Change the end code of the ASCII modal string |
| | 0×04 | (Skip) Forced to only listen mode | Switch the slave station to the answering mode only |
| | 0×0A | (Skip) Clear counter and diagnostic register | Clear the contents of all counters and diagnostic registers |
| | 0×0B | (Skip) Count of bus information | Read the number of detected messages |
| | 0×0C | (Skip) Count of abnormal bus communication | Reading the number of communication abnormalities |
| | 0×0D | (Skip) Count of bus abnormalities | Error status readout |
| | 0×0E | (Skip) Number of slave answers | Read the number of requests received |
| | 0×0F | (Skip) The number of times that the slave does not answer | The number of times that the slave does not answer is read out |
| | 0×10 | (Skip) The number of times the slave answered NAK | The number of times the slave answers NAK is read out |
| | 0×11 | (Skip) The number of times the slave station answered BUSY | The number of times the slave station answers BUSY is read out |
| 0×12 | (Skip) Text exceeds count | Count of text exceeded readout | |
| 0×0B | | (Skip) Count of communication events | Count readout of communication events |
| 0×0C | | (Skip) Record of communication events | Communication event record readout |
| 0×0F | | Multiple bit writing | Multiple bit writing (R/W) |

| | | |
|------|--|--|
| 0×10 | Write to multiple registers | Multiple 16-bit register write (R/W) |
| 0×11 | (Skip) Slave status readout | Slave status readout |
| 0×16 | (Skip) Register logic overwrite write | Write AND logic/OR logic to the slave register |
| 0×17 | (Skip) Reading and writing of multiple registers | Multiple 16-bit register read and write (R/W) |

4. Slave Station specifications

| Instruction code | Secondary code | Command name | Detail |
|------------------|----------------|---|--|
| | 0×01 | Bit read | Bit signal readout (R/W) |
| | 0×02 | (Skip) | Bit signal readout (RO) |
| | 0×03 | Register readout | 16-bit register readout (R/W) |
| | 0×04 | (Skip) | 16-bit register readout (RO) |
| | 0×05 | Bit write | Bit signal write |
| | 0×06 | Write to a single register | Single 16-bit register write |
| | 0×07 | (Skip) Status readout | 1 byte data read |
| 0×08 | 0×00 | (Skip) Return of the question | Loop back test |
| | 0×01 | (Skip) Restart of communication settings | Communication restart/remote communication reset -Clear all counters -Reply to "Only answer mode" -Communication event reset |
| | 0×02 | (Skip) Return of the diagnostic register | Specified 16-bit register readout CH1: M8060-M8067 CH2: M8060-M8062, M8438, M8064-M8067 Note: The upper 8 bits (b15~b8) are not used |
| | 0×03 | (Skip) ASCII End code change | Change the end code of the ASCII modal string |
| | 0×04 | (Skip) Forced to only answer mode | Switch the slave station to the answering mode only Note: When the slave station enters the "answer mode only", all communication control actions of the station will stop, and the slave station will not respond to the master station. |
| | 0×0A | (Skip) Clear counter and diagnostic register. | Clear the contents of all counters and diagnostic registers |
| | 0×0B | (Skip) Count of bus information | Read the number of detected messages |
| | 0×0C | (Skip) Count of abnormal bus communication | Reading the number of communication abnormalities |
| | 0×0D | (Skip) Count of bus abnormalities | Error status readout |
| | 0×0E | (Skip) Number of slave answers | Read the number of requests received |
| | 0×0F | (Skip) The number of times that the slave does not answer | Reading the number of times that the sub-station does not return |
| | 0×10 | (Skip) The number of times the slave answered NAK | The number of times the sub-station answers NAK is read out |

| | | | |
|--|------|--|--|
| | 0×11 | (Skip) The number of times the slave station answered BUSY | The number of times the slave station answers BUSY is read out |
| | 0×12 | (Skip) Count of text exceeded | The number of times the text exceeds the reading |
| | 0×0B | (Skip) Count of communication events | Reading of communication events |
| | 0×0C | (Skip) Communication event record | Communication event record readout |
| | 0×0F | Multiple bit writing | Multiple bit writing (R/W) |
| | 0×10 | Write to multiple registers | Write multiple 16-bit registers (R/W) |
| | 0×11 | (Skip) Slave status readout | Slave status readout |
| | 0×16 | (Skip) Register logic overwrite write | Write AND logic/OR logic to the slave register |
| | 0×17 | (Skip) Reading and writing of multiple registers | Multiple 16-bit register read and write (R/W) |

A

ASCII List

| HEX | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|-----|-----|-----|----|---|---|---|---|-----|
| 0 | | DLE | SP | 0 | @ | P | ` | p |
| 1 | SOH | DC1 | ! | 1 | A | Q | a | q |
| 2 | STX | DC2 | " | 2 | B | R | b | r |
| 3 | ETX | DC3 | # | 3 | C | S | c | s |
| 4 | EOT | DC4 | \$ | 4 | D | T | d | t |
| 5 | ENQ | NAK | % | 5 | E | U | e | u |
| 6 | ACK | SYN | & | 6 | F | V | f | v |
| 7 | BEL | ETB | ' | 7 | G | W | g | w |
| 8 | BS | CAN | (| 8 | H | X | h | x |
| 9 | HT | EM |) | 9 | I | Y | i | y |
| A | LF | SUB | * | : | J | Z | j | z |
| B | VT | ESC | + | ; | K | [| k | { |
| C | FF | FS | , | < | L | \ | l | |
| D | CR | GS | - | = | M |] | m | } |
| E | SO | RS | . | > | N | ^ | n | ~ |
| F | SI | US | / | ? | O | _ | o | DEL |

B

The CRC check code generation process is as follows:

- 1) Specify a temporary storage number whose content is FFFF (16-bit content is 1), which is called CRC temporary storage.
- 2) Perform an XOR operation on the first 8 bits (b7~b0) of the CRC register and the first 8 bits (station number) of the "error checking calculation range" in the string, and the result obtained is stored in the CRC register.
- 3) Shift the CRC register by one bit, put the leftmost bit (b15) into 0, and then check the carry flag (Carry Flag, the original content of b0).
- 4) When the content of the "carry flag" = 0, repeat the action of 3). When the content of the "carry flag"=1, the content of the CRC register and the polynomial value 0xA001 (1010 0000 0000 0001) execute the XOR operation.
- 5) Repeat the actions of 3) and 4) until 8 displacements. This article is a complete paragraph.
- 6) Repeat the actions of 2)~5) with the new value obtained in the paragraph and the next 8 bits of content (function code) in the string until all the data in the "error checking calculation range" in the string has been executed.
- 7) The content of the last CRC register is the "CRC value"
- 8) Please note that the upper 8 bits and the lower 8 bits of the "CRC value" should be interchanged before this is the "CRC check code". Finally, put the "CRC check code" in the string.
- 9) Take station number "02H" and "function code 07H" as examples, the calculation process of CRC check code

| station number | function code | CRC check code | |
|--------------------|--------------------|--------------------|--------------------|
| (02 _H) | (07 _H) | (41 _H) | (12 _H) |

| calculation process | CRC register value | | | | Carry Flag |
|---|--------------------|-----------------|------|-----------------|-------------|
| Specify a temporary storage 02H (station number) XOR whose content is "FFFF" | 1111 | 1111 | 1111 | 1111 | |
| | | | 0000 | 0010 | |
| | 1111 | 1111 | 1111 | 1101 | |
| Shift one bit to the right (1st degree) polynomial value XOR | 0111 | 1111 | 1111 | 1110 | 1 |
| | 1010 | 0000 | 0000 | 0001 | |
| | 1101 | 1111 | 1111 | 1111 | |
| Shift one bit to the right (2nd degree) polynomial value XOR | 0110 | 1111 | 1111 | 1111 | 1 |
| | 1010 | 0000 | 0000 | 0001 | |
| | 1100 | 1111 | 1111 | 1110 | |
| Shift one bit to the right (3rd order) Shift one bit to the right (4th order) polynomial value XOR | 0110 | 0111 | 1111 | 1111 | 0 1 |
| | 0011 | 0011 | 1111 | 1111 | |
| | 1010 | 0000 | 0000 | 0001 | |
| | 1001 | 0011 | 1111 | 1110 | |
| Shift one bit to the right (5th time) Shift one bit to the right (6th order) polynomial value XOR | 0100 | 1001 | 1111 | 1111 | 0 1 |
| | 0010 | 0100 | 1111 | 1111 | |
| | 1010 | 0000 | 0000 | 0001 | |
| | 1000 | 0100 | 1111 | 1110 | |
| Shift one bit to the right (7th time) Shift the polynomial value XOR by one bit to the right (the 8th degree) | 0100 | 0010 | 0111 | 1111 | 0 1 |
| | 0010 | 0001 | 0011 | 1111 | |
| | 1010 | 0000 | 0000 | 0001 | |
| | 1000 | 0001 | 0011 | 1110 | |
| 07H(function code) XOR | | | 0000 | 0001 | |
| | 1000 | 0001 | 0011 | 1001 | |
| Shift one bit to the right (1st degree) polynomial value XOR | 0100 | 0000 | 1001 | 1100 | 1 |
| | 1010 | 0000 | 0000 | 0001 | |
| | 1110 | 0000 | 1001 | 1101 | |
| Shift one bit to the right (2nd degree) polynomial value XOR | 0111 | 0000 | 0100 | 1110 | 1 |
| | 1010 | 0000 | 0000 | 0001 | |
| | 1101 | 0000 | 0100 | 1111 | |
| Shift one bit to the right (3rd degree) polynomial value XOR | 0110 | 1000 | 0010 | 0111 | 1 |
| | 1010 | 0000 | 0000 | 0001 | |
| | 1100 | 1000 | 0010 | 0110 | |
| Shift one bit to the right (4th order) Shift one bit to the right (5th order) polynomial value XOR | 0110 | 0100 | 0001 | 0011 | 0 1 |
| | 0011 | 0010 | 0000 | 1001 | |
| | 1010 | 0000 | 0000 | 0001 | |
| | 1001 | 0010 | 0000 | 1000 | |
| Shift one bit to the right (6th time) Shift one bit to the right (7th time) Shift one bit to the right (8th time) | 0100 | 1001 | 0000 | 0100 | 0 0 0 |
| | 0010 | 0100 | 1000 | 0010 | |
| | 0001 | 0010 | 0100 | 0001 | |
| CRC value | | 12 _H | | 41 _H | |

LIYAN PROGRAMMABLE LOGIC CONTROLLER

ExModbus-E20130705V100

LIYAN ELECTRIC INDUSTRIAL LTD.

TEL : 886 - 4 – 25613700

FAX : 886 - 4 – 25613408

Website : <http://www.liyanplc.com>

E – mail : twliyan@ms16.hinet.net

The company reserves the right to change the specifications of the model