# 1. Introduction

EX-9018/18-M/18BL/18BL-M/19/19-M is a thermocouple input module and 8 input channels are differential type .

Specifications:
Interface: RS-485, 2 wires
Speed (bps): 1200, 2400, 4800, 9600, 19.2K, 38.4K, 15.2K
Analog Input type: Differential input
Analog Channels Numbers: 8
Analog Resolution: 16 bits
Unit Conversion: Thermocouple, mV, V or mA
Thermocouple Type: J, K, T, E, R, S, B, N
Sampling Rate :10 Samples/Second
Bandwidth : 15.7 Hz
Accuracy : ±0.1%
Zero Drift : 0.5μV/°C
Span Drift : 25ppm/°C
CMR@50/60Hz : 150dB
NMR@50/60Hz : 100dB
Input Impedance : 20M Ohms
Voltage Range: ±2.5V, ±1V,±500mV,±100mV,±50mV,±15mV
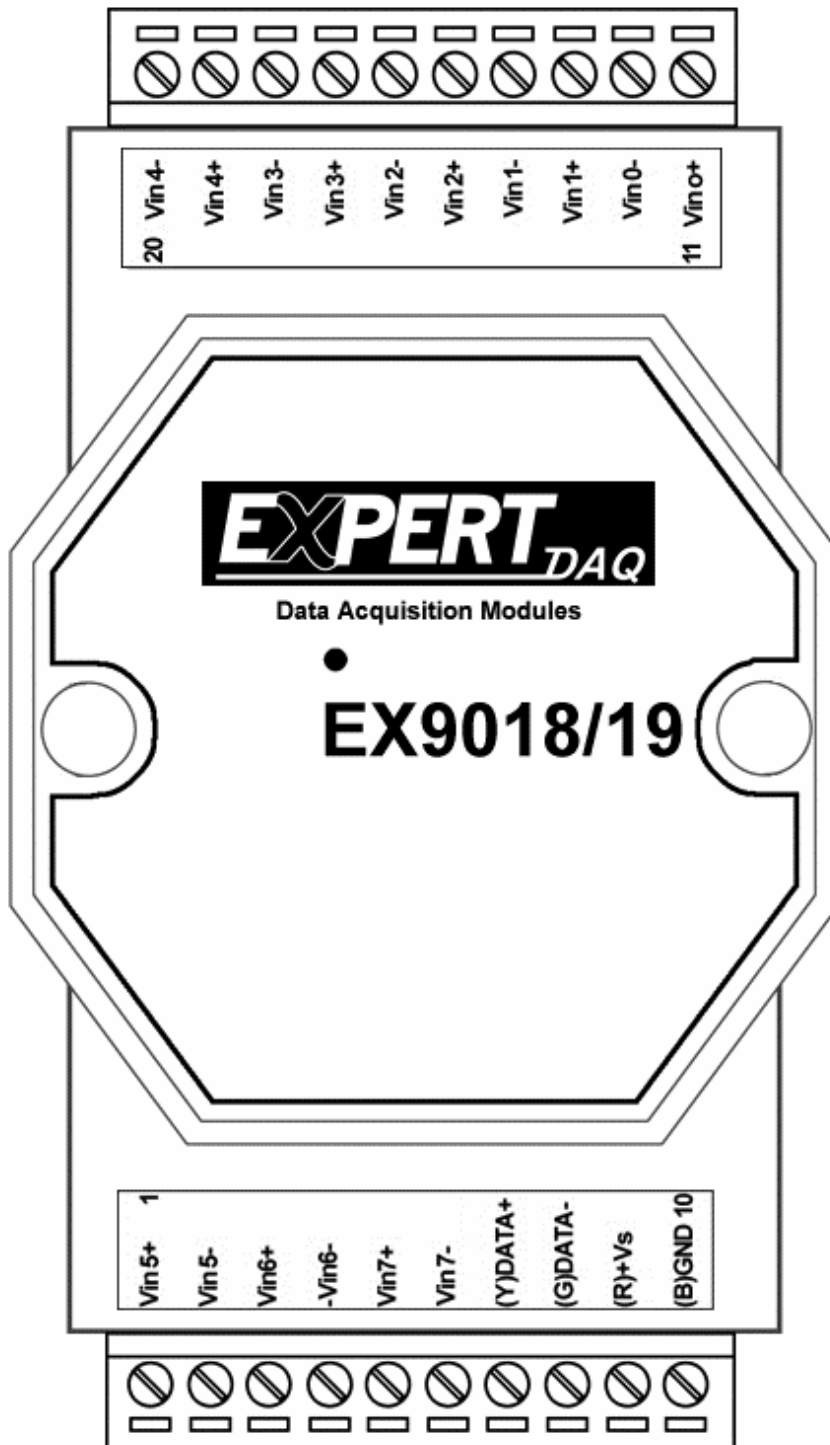Current Measurement: ±20mA (with external 125 ohms resistor)
Power supply: +10V to +30V

Thermal couple break line detection (EX9018BL/BL-M/19/19-M

only)

# Thermocouple Measurement

Two wires composed of dissimilar metal are joined at one end and heated, the open circuit voltage is a function of the junction temperature and the composition of the two metals. All dissimilar metals exhibit this effect. The voltage is called "seebeck voltage". For small changes in temperature the Seebeck Voltage is linearly proportional to temperature.

For measure the Seebeck Voltage directly is not available because we must connect a voltmeter to the thermocouple and the voltmeter leads themselves create a new thermoelectric circuit firstly. Therefore we need to eliminate the junction thermoelectric to measure to correct Seebeck Voltage and this is called "Cold Junction Compensation". For most thermocouples, the Seebeck Voltage is 0V while in 0°**C**. The simple way to cancel the junction voltage is to put the junction into 0°**C** environment and the junction voltage is 0V. In general, this is not a good method for most application. Typical method is to measure the junction temperature by thermistor and measure the junction voltage from the junction temperature that we may get the Seebeck Voltage from measured thermocouple voltage and junction voltage and we may calculate the temperature from the Seebeck Voltage.
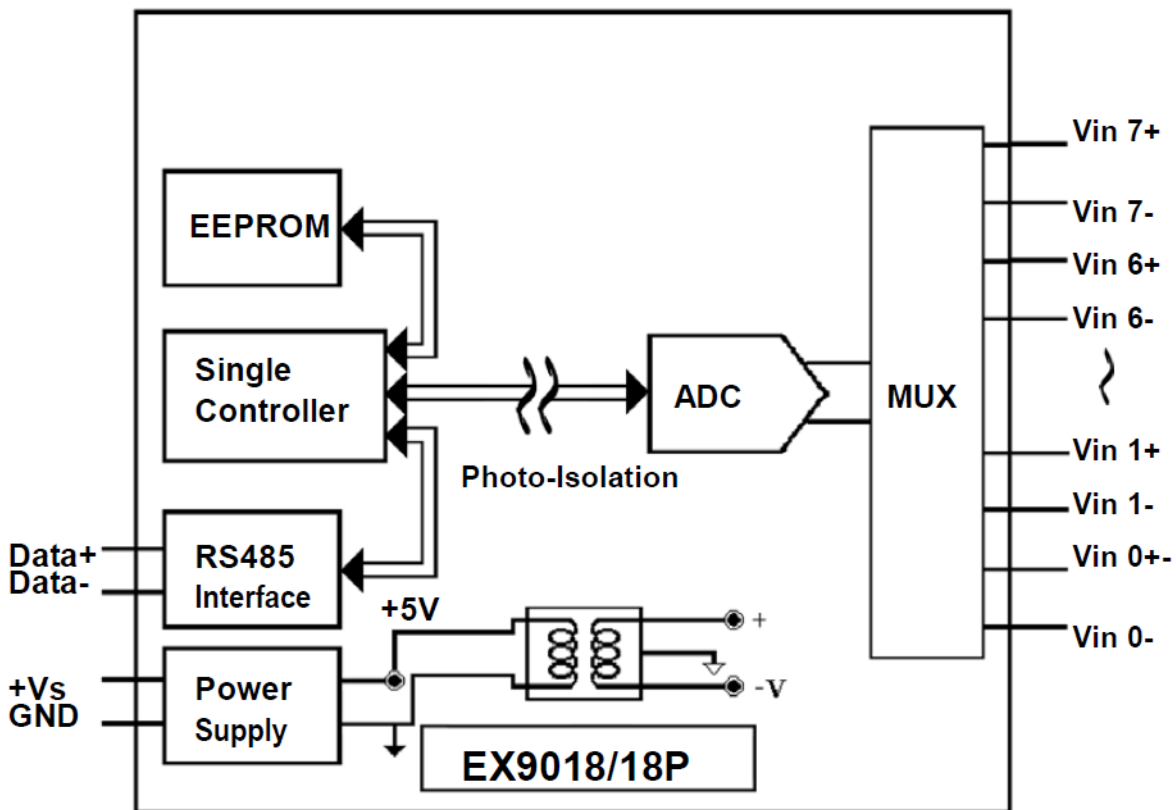
EX-9018/18-M/18BL/18BL-M/19/19-M

# Specifications

| | EX-9018 EX-9018-M | EX-9018BL EX-90018BL-M | EX-9019 EX-9019-M |
|---|---|---|---|
| **Interface** | RS-485, 2 wires | | |
| **Speed(bps)** | 1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 | | |
| **Analog Input type** | Differential input | | |
| **Input Channels** | 8 | | |
| **Resolution** | 16 bits | | |
| **Individual channel configuration** | Not support | | Support |
| **Voltage Input** | -15mV ~ +15mV<br>- 50mV ~ +50mV<br>- 100mV ~ +100mV<br>- 500mV ~ +500mV<br>- 1V ~ +1V<br>- 2.5V ~ +2.5V | | |
| **Current Input** | -20mA ~ +20mA (with 125ohms resistor) | | |
| **Sensor Input** | J, K, T, E, R, S, B, N | | |
| **Sampling Rate** | 10 sample/second | | |
| **Bandwidth** | 15.7Hz | | |
| **Accuracy** | ±0.1% | | |
| **Zero Drift** | $0.5\mu V/^\circ C$ | | |
| **Span Drift** | $25ppm/^\circ C$ | | |
| **CMR@50/60Hz** | 150dB | | |
| **NMR@50/60Hz** | 100dB | | |
| **Input Impedance** | 20M ohms | | |
| **Power supply** | +10V ~ +30V | | |
| **Modbus RTU** | EX9018-M | EX9018BL-M | EX9019-M |
| **Thermal couple break line detect** | Not support | Support | |

**Notes**:
1. Warm-UP for 30 minutes is recommended before starting operation!
2. EX-9018-M: EX-9018 w/ Modbus function
3. EX-9018BL: EX-9018 w/ Break Line function for Thermocouple
4. EX-9018BL-M: EX-9018BL w/ Modbus function
5. EX-9019: EX-9018BL w/ Individual Channel setting
6. EX-9019-M: EX-9019 w/ Modbus function

# 1.2 Wire connection

## 1.2.1 Block Diagrams:



EX-9018//18BL//19

# 1.2.2 Wiring diagram for the EX-9018/18BL/19



## 1.3 Default Settings

Default settings for the EX-9018/18BL/19 modules are as follows:

. Module Address: 01
. Analog Input Type: type 0F
. Baud Rate: 9600 bps
. Checksum disabled
. Engineering unit format
. Filter set at 60Hz rejection

Default settings for the EX-9018-M/18BL-M/19-M modules are as follows:
. Protocol: Modbus RTU
. Module Address: 01
. Analog Input Type: type 0F
. Baud Rate: 9600 bps
. Filter set at 60Hz rejection

# 1.4 INIT* Mode Operation

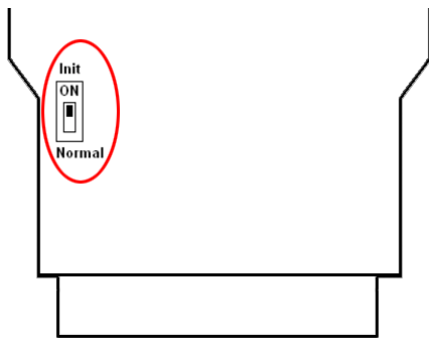Each EX9000 module has a build-in EEPROM to store configuration information such as address, type, baudrate and other information. Sometimes, user may forget the configuration of the module. Therefore, the EX9000 have a special mode named "INIT* mode" to help user to resolve the problem. The "INIT* mode" is setting as Address=00, Budrate=9600bps, no Checksum .

Originally, the INIT* mode is accessed by connecting the INIT* terminal to the GND terminal. New EX9000 modules have the INIT* switch located on the rear side of the module to allow easier access to the INIT* mode. For these modules, INIT* mode is accessed by sliding the INIT* switch to the Init position as shown below.



To enable INIT* mode, please following these steps:
Step1. Power off the module
Step2. Connect the INIT* pin with the GND pin.
(or sliding the INIT* switch to the Init* ON position)
Step3. Power on
Step4. Send command $002 (cr) in 9600bps to read the
Configuration stored in the module's EEPROM.

There are commands that require the module to be in INIT* mode. They are:
1. %AANNTTCCFF when changing the Baud Rate and checksum settings. See Section 2.1 for details.
2. $AAPN, See Section 2.19 for details.

# 1.5 Module Status for DIO, AIO

**Power On Reset** or **Module Watchdog Reset** will let all output goto **Power On Value**. And the module may accept the host's command to change the output value.

**Host Watchdog Timeout** will let all output goto **Safe Value**. The module's status(read by command~AA0) will be 04, and the output command will be ignored.

# 1.6 Dual Watchdog Operation for DIO, AIO

**Dual Watchdog=Module Watchdog + Host Watchdog**

The Module Watchdog is a hardware reset circuit to monitor the module's operating status. While working in harsh or noisy environment, the module may be down by the external signal. The circuit may let the module to work continues and never halt.

The Host Watchdog is a software function to monitor the host's operating status. Its purpose is to prevent the network from communication problem or host halt. When the timeout interval expired, the module will turn all outputs to predefined Safe Value. This can prevent the controlled target from unexpected situation.

The EX9000 module with Dual Watchdog may let the control system more reliable and stable.

# 1.7 Reset Status

The Reset Status is set while the module power on or reset by module watchdog and is cleared while the command read Reset Status ($AA5) applied. This is useful for user to check the module's working status. When the Reset Status is set means the module is reset and the output may be changed to the PowerOn Value. When the Reset Status is clear means the module is not rested and the output is not changed.

# 1.8 Calibration(Warning: Please don't calibrate before you really understand.)

Set the module of data format to which you wish to calibration first.

Protocol: ASCII mode.

Address: 01

Input type: which type you wish to calibration (If want calibrate the thermocouple type, please set the type to **-50~+50mV** because use the same calibrate parameter)

Filter: which rejection you wish to calibration

Perform Zero Calibration:

1. Send the command "$01501" to **CH0 enable, CH1~7 disable.**
2. Apply zero voltage/current to module's **channel 0** (Vin0+ to Vin0-)
3. Send the command "~01E1" to enable calibration.
4. Send the command "$011" to perform zero calibration.

Perform Span Calibration:

1. Send the command "$01502" to **CH1 enable, CH0 & 2~7 disable.**
2. Apply Span voltage/current to module's **channel 1**

| Type code | 01 | 02 | 03 | 04 | 05 | 06 |
|-----------|------|------|--------|------|-------|-------|
| Span | +15mV | +50mV | +100mV | +1V | +2.5V | +20mA |

3. Send the command "~01E1" to enable calibration.
4. Send the command "**~01CALS00550000**" to perform span calibration.
5. Send the command "#011" to check the CH0 input value is correct.

    If the value is over the signal, decrease the value "550000".

    If the value is less the signal, increase the value "550000".

    The value should between **500000~5B0000**, Hexadecimal format.

**Note: While calibrate type of current, need connect external shunt resistor, 125ohms, 0.1% to channel1.**

# 1.9 Configuration Tables

## Baud Rate Setting (CC)

| Code | 03 | 04 | 05 | 06 | 07 | 08 | 09 | 0A |
|---|---|---|---|---|---|---|---|---|
| Baud rate | 1200 | 2400 | 4800 | 9600 | 19200 | 38400 | 57600 | 115200 |

## Sensor Type & V/I Range Setting (TT)

| Code | Range | Format | +F.S. | Zero | -F.S. |
|---|---|---|---|---|---|
| 00 | -15~+15mV | Engineer unit | +15.000 | +00.000 | -15.000 |
| | | % of F.S.R. | +100.00 | +000.00 | -100.00 |
| | | 2's complement | 7FFF | 0000 | 8000 |
| 01 | -50~+50mV | Engineer unit | +50.000 | +00.000 | -50.000 |
| | | % of F.S.R. | +100.00 | +000.00 | -100.00 |
| | | 2's complement | 7FFF | 0000 | 8000 |
| 02 | -100~+100mV | Engineer unit | +100.00 | +000.00 | -100.00 |
| | | % of F.S.R. | +100.00 | +000.00 | -100.00 |
| | | 2's complement | 7FFF | 0000 | 8000 |
| 03 | -500~+500mV | Engineer unit | +500.00 | +000.00 | -500.00 |
| | | % of F.S.R. | +100.00 | +000.00 | -100.00 |
| | | 2's complement | 7FFF | 0000 | 8000 |
| 04 | -1~+1V | Engineer unit | +1.0000 | +0.0000 | -1.0000 |
| | | % of F.S.R. | +100.00 | +000.00 | -100.00 |
| | | 2's complement | 7FFF | 0000 | 8000 |
| 05 | -2.5~+2.5V | Engineer unit | +2.5000 | +0.0000 | -2.5000 |
| | | % of F.S.R. | +100.00 | +000.00 | -100.00 |
| | | 2's complement | 7FFF | 0000 | 8000 |
| 06 | -20~+20mA | Engineer unit | +20.000 | +00.000 | -20.000 |
| | | % of F.S.R. | +100.00 | +000.00 | -100.00 |
| | | 2's complement | 7FFF | 0000 | 8000 |
| 0E | Type J T/C -210~+760°C | Engineer unit | +760.00 | +000.00 | -210.00 |
| | | % of F.S.R. | +100.00 | +000.00 | -027.63 |
| | | 2's complement | 7FFF | 0000 | DCA2 |
| 0F | Type K T/C -270~+1372°C | Engineer unit | +1372.0 | +0000.0 | -0270.0 |
| | | % of F.S.R. | +100.00 | +000.00 | -019.68 |
| | | 2's complement | 7FFF | 0000 | E6D0 |
| 10 | Type T T/C | Engineer unit | +400.00 | +000.00 | -270.00 |

| | -270~+400°C | % of F.S.R. | +100.00 | +000.00 | -067.50 |
|---|---|---|---|---|---|
| | | 2's complement | 7FFF | 0000 | DCA2 |
| 11 | Type **E** T/C -270~+1000°C | Engineer unit | +1000.0 | +000.00 | -0270.0 |
| | | % of F.S.R. | +100.00 | +000.00 | -027.00 |
| | | 2's complement | 7FFF | 0000 | DD71 |
| 12 | Type **R** T/C 0~+1768°C | Engineer unit | +1768.0 | +0000.0 | -0000.0 |
| | | % of F.S.R. | +100.00 | +000.00 | -100.00 |
| | | 2's complement | 7FFF | 0000 | 0000 |
| 13 | Type **S** T/C 0~+1768°C | Engineer unit | +1768.00 | +0000.0 | -0000.0 |
| | | % of F.S.R. | +100.00 | +000.00 | -100.00 |
| | | 2's complement | 7FFF | 0000 | 0000 |
| 14 | Type **B** T/C 0~+1820°C | Engineer unit | +1820.0 | +0000.0 | -0000.0 |
| | | % of F.S.R. | +100.00 | +000.00 | -100.00 |
| | | 2's complement | 7FFF | 0000 | 0000 |
| 15 | Type **N** T/C -270~+1300°C | Engineer unit | +1300.0 | +0000.0 | -0270.0 |
| | | % of F.S.R. | +100.00 | +000.00 | -020.77 |
| | | 2's complement | 7FFF | 0000 | F54D |

## Data Format Setting (FF)

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| FS | CS | reserved | | | | DF | |

| Key | Description |
|---|---|
| DF | Data format 00: Engineering unit 01: % of FSR (full scale range) 10: 2's complement hexadecimal |
| CS | Checksum setting 0: Disabled 1: Enabled |
| FS | Filter setting 0: 60Hz rejection 1: 50Hz rejection |

**Note**: The reserved bits should be zero.

## Burnout (Break line for Thermocouple) status's reading

| Engineering Unit | **+9999.9** |
|---|---|
| % of FSR | **+1315.7** |
| 2's Complement HEX | **7FFF** |

# 2.0 Command set

## 2.1 %AANNTTCCFF

**Description:** Set Module Configuration.
**Syntax: %AANNTTCCFF[CHK](cr)**

%       a delimiter character
AA      address of setting/response module(00 to FF)
NN      new address for setting/response module(00 to FF)
TT      represents the type code. Type code determines the
        input range.
        If TT=FF the type of all channels keep no change.
CC      new baudrate for setting module.
FF      new data format for setting module.

IF the configuration with new baudrate or new checksum setting, before using this command, the rear slide switch must be in the ON(INIT*) position. The new setting is saved in the EEPROM and will be effective after the next power-on reset.

**Response:** Valid Command:      **!AA**
Invalid Command:                  **?AA**

**Example:**

Command:   %0203080602           Receive: !02
Set module    address **02** to **03**.
              Input type code=**08** (-10~+10V) for all channels
              Baudrate=**06** (9600)
              Dataformat=**02** (2's complement hexadecimal)

## 2.2 #AA

**Description:** Read Analog Input

**Syntax: #AA[CHK](cr)**

\#       delimiter character

AA    address of reading/response module(00 to FF)

**Response:** Valid Command:          >(Data)

(Data) analog input value for its format while use #AA command to EX-9018BL/9019, the data is the combination for each channel respectively.

## Example :

Command: #04

Receive:>+051.23+041.53+072.34-023.56+100.00-051.33+066.46+074.22

The module address 04 is EX-9018BL/9019. Read address 04 for getting data of all 8 channels.

# 2.3 #AAN

**Description :** Read Analog Input from channel N

**Syntax : #AAN[CHK](cr)**

\#       delimiter character

AA    address of reading/response module(00 to FF)

N       channel to read, from 0 to 7

**Response:** Valid Command:          >(Data)

Invalid Command:          ?AA


(Data) analog input value for its format


## Example :

Command : #032                              Receive : >+025.13

Read address 03 channel 2, get data successfully.

Command : #029                              Receive : ?02

Read address 02 channel 9, return error channel number.

# 2.4 $AA0

**Description:** Perform Span Calibration

**Syntax:** **$AA0[CHK](cr)**

$     delimiter character

AA   address of setting/response module (00 to FF)

0     command for performing zero calibration

**Response:** Valid Command:       **!AA**

            Invalid Command:     **?AA**

**Example :**

Command : $010                    Receive : !01

Perform address 01 zero calibration on channel 0, return success.

Command : $020                    Receive : ?02

Perform address 02 zero calibration on channel 2 , return not enable calibration before perform calibration command.

**Warning: Pls don't calibrate before you really understand.**

# 2.5 $AA1

**Description:** Perform Zero Calibration

**Syntax: $AA1[CHK](cr)**

$      delimiter character

AA     address of setting/response module (00 to FF)

1      command for performing span calibration

**Response:** Valid Command:      **!AA**

                Invalid Command:      **?AA**

## Example:

Command: $011                 Receive: !01

Perform address 01 span calibration on channel 0, return success.

Command: $021                 Receive: ?02

Perform address 02 span calibration on channel 2, return not enable calibration before perform calibration command.

**Warning: Pls don't calibrate before you really understand.**

# 2.6 $AA2

**Description:** Read configuration.
**Syntax: $AA2[CHK](cr)**

$   delimiter character

AA   address of reading/response module (00 to FF)

2   command for read configuration

**Response:** Valid Command:  **!AATTCCFF**
      Invalid Command: **?AA**

TT  type code of module

CC  baudrate code of module

FF  data format of module

## Example:

Command: $012        Receive: !01400600

Read the configuration of module 01, input range of -2.5~+2.5V, baudrate 9600, no checksum.

**Note: check configuration Tables**

# 2.7 $AA3

**Description:** Reads cold junction temperature.

**Syntax: $AA3[CHK](cr)**

$  delimiter character

AA  address of reading/response module(00 to FF)

4  command to read cold junction temperature

**Response:** Valid Command:   **>(Data)**

      Invalid Command:   **?AA**


(Data) CJC temperature in degrees Celsius, consisting of a sign byte, '+' or '-' and followed by 5 decimal digits with a fixed decimal point in tenth of a degree

**Example:**

Command: $013        Receive: >+0030.2

Read address 01 cold junction temperature, and the module responds with +0030.2

# 2.8 $AA5VV

**Description:** Set Channel Enable

**Syntax: $AA5VV[CHK](cr)**

$      delimiter character

AA      address of setting/response module (00 to FF)

5      command for set channel enable

VV      are two hexadecimal values. The values are interpreted by the module as two binary words (4-bit). The first word represents the status of channel 4~7, and the second word represents the status of channel 0~3. Value 0 means the channel is disabled, value 1 means the channel is enabled.

**Response:** Valid Command:      **!AA**

         Invalid Command:      **?AA**

**Example:**

Command :$0152A            Receive : !01

Set address 01 to enable channel 1,3,5 and disable channel 0,2,4,6,7 return success.

Command : $016            Receive : !012A

Read address 01 channel status, return channel 1,3,5 are enabled and channel 0,2,4,6,7 are disabled.

# 2.9 $AA6

**Description:** Read Channel Status

**Syntax:** **$AA6[CHK](cr)**

$        delimiter character

AA     address of reading/response module (00 to FF)

6        command for read channel status

**Response:** Valid Command:        **!AAVV**

           Invalid Command:        **?AA**

VV     are two hexadecimal values. The values are interpreted by the module as two binary words (4-bit). The first word represents the status of channel 4~7, and the second word represents the status of channel 0~3. Value 0 means the channel is disabled, value 1 means the channel is enabled.

## Example:

Command :$0152A        Receive : !01

Set address 01 to enable channel 1,3,5 and disable channel 0,2,4,6,7 return success.

Command : $016                     Receive : !012A

Reads Read address 01 channel status, return channel 1,3,5 are enabled and channel 0,2,4,6,7 are disabled.

# 2.10 $AA7CiRrr
## (For EX-9019/19M only)

**Description:** Sets the type code of a channel individually.

**Syntax: $AA7CiRrr[CHK](cr)**

$        delimiter character

AA     address of setting/response module(00 to FF)

7        set the channel range code

Ci      i specifies the input channel to be set

Rrr    rr represents the type code of the channel to be set.

**Response:** Valid comma nd:       **!AA**

                   Invalid command:      **?AA**

## Example:

Command: $017C3R08             Receive: !01

Sets the type code for channel 3 of module 01 to be
08 (-10~+10V) and the module returns a
valid response.

Command: $037C1R40             Receive: ?03

Sets the type code for channel 1 of module 03 to be
40. The module returns an invalid response because
the type code is invalid.

# 2.11 $AA8Ci
## (For EX-9019/19M only)

**Description:** Reads the type code information of a channel.

**Syntax: $AA8Ci[CHK](cr)**

$        delimiter character

AA     address of reading/response module(00 to FF)

8        read the type code of a channel

Ci     specifies which channel to be access for the type code

**Response:** Valid command:        **!AACiRrr**

            Invalid command:       **?AA**

i        specifies which channel to be access for the type code

rr      represents the type code of the channel to be read

## Example:

Command: $018C0                  Receive: !01C0R03

Reads the type(input range) of channel 0 of module 01 to be 03 (-10~+10V).

# 2.12 $AA9

**Description:** Read cold junction offset.

**Syntax: $AA9[CHK](cr)**

$         delimiter character

AA      address of reading/response module (00 to FF)

9         command for read cold junction offset

**Response:** Valid Command:      **!AAsnnnn**

                    Invalid Command:    **?AA**

S         sign of cold junction offset

nnnn   cold junction offset in 0.01°C unit

**Example:**

Command: $019                          Receive: !01+0010

Read the cold junction offset is **+0010(Hex)*0.01=+0.16°C**

# 2.13 $AA9snnnn

**Description:** Set cold junction offset value.
**Syntax: $AA9snnnn[CHK](cr)**

$           delimiter character

AA          address of reading/response module (00 to FF)

9           command for set cold junction offset

s           sign of cold junction offset

nnnn        cold junction offset (Hex) in 0.01°C unit (0000~0999)

**Response:** Valid Command:        **!AA**

                Invalid Command:      **?AA**

**Example:**

Command: $019+0010                    Receive: !01

Set the cold junction offset to **+0010(Hex)\*0.01=+0.16°C**

# 2.14 $AAB
## (For EX-9018BL/18BLM/19/19M only)

**Description:** Read channel burnout status.

**Syntax: $AAB[CHK](cr)**

$       delimiter character

AA     address of reading/response module (00 to FF)

B      diagnose the analog inputs

**Response:** Valid command:        **!AANN**

             Invalid command:     **?AA**

NN     (range 00-FF) is a hexadecimal number that equals the 8-bit parameter, representing the status of analog input channels. Bit value 0 means normal status; and bit value 1 means channel open wiring.

## Example:

Command: $01B                 Receive: !0101

Diagnoses the analog inputs of module 01. The module returns a valid response that channel 0 is open wiring and channel 1~7 are all normal

# 2.15 ~AACe

**Description:** Enable/Disable cold junction compensation.

**Syntax: ~AACe[CHK](cr)**

~        delimiter character

AA      address of reading/response module (00 to FF)

C        command for Enable/Disable CJC

e        e=0 disable CJC, e=1 enable CJC

**Response:** Valid Command:      **!AA**

Invalid Command:    **?AA**

**Example:**

Command: ~01C1                          Receive: !01

Enable CJC for all channels.

## 2.16 $AAF

**Description:** Read Firmware Version

**Syntax: $AAF[CHK](cr)**

$       delimiter character

AA     address of reading/response module(00 to FF)

F       command for read firmware version

**Response:** Valid command:       **!AA(Data)**

               Invalid command:       **?AA**

(Data) Firmware version of module

**Example:**

Command : $01F                        Receive : !01M6.92

Read address 01 firmware version, return version M6.92

# 2.17 $AAM

**Description:** Read Module Name

**Syntax:** $AAM[CHK](cr)

$  delimiter character

AA  address of reading/response module(00 to FF)

M  command for read module name

**Response:** Valid command:   **!AA(Data)**

     Invalid command:   **?AA**

(Data) Name of module

**Example:**

Command : $01M      Receive : !019019

Read address 01 module name, return name 9019.

# 2.18 $AAP
# (For EX-9018M/18BLM/19M only)

**Description:** Read protocol information of Module

**Syntax:** $AAP[CHK](cr)

$       delimiter character

AA    address of reading/response module(00 to FF)

P     command for read protocol information of module

**Response:** Valid command:           **!AAS**

                Invalid command:     **?AA**


S The protocol supported by the module

    0: the protocol set in EEPROM is Normal mode

    1: the protocol set in EEPROM is ModbusRTU mode


**Example:**

Command: $01P                Response: !010

Reads the communication protocol of module 01 and
returns a response of 0 meaning the protocol that will be used at
the next power on reset is normal mode.


Command: $01P1             Response: !01

Sets the communication protocol of module 01 to
Modbus RTU and returns a valid response. And the next power
on reset is in ModbusRTU mode.

# 2.19 $AAPN
## (For EX-9018M/18BLM/19M only)

**Description:** Set the protocol information of Module

**Syntax:** $AAPN[CHK](cr)

$      delimiter character

AA    address of reading/response module(00 to FF)

P      command for read protocol information of module

N     The protocol supported by the module

      0: the protocol set in EEPROM is Normal mode

      1: the protocol set in EEPROM is ModbusRTU mode

**Response:** Valid command:      **!AA**

                Invalid command:      **?AA**

**Example:**

Command: $01P1                 Response: !01

Sets the communication protocol of module 01 to

Modbus RTU and returns a valid response. And the next power

on reset is in ModbusRTU mode.

# 2.20 ~AAEV

**Description:** Enable/Disable Calibration

**Syntax: ~AAEV[CHK](CR)**

~      delimiter character

AA    address of setting/response module (00 to FF)

E      command for enable/disable calibration

V      1=Enable/0=Disable calibration

**Response:** Valid Command:         **!AA**

Invalid Command:       **?AA**

**Example:**

Command : $010                    Receive: ?01

Perform address 01 span calibration, return the command is invalid before enable calibration.

Command : ~01E1                   Receive: !01

Set address 01 to enable calibration, return success.

Command: $010                     Receive: !01

Preform address 01 span calibration, return success.

**Warning: Pls don't calibrate before you really understand.**

# 2.21 ~AAO(Data)

**Description:** Set Module Name

**Syntax: ~AAO(Data)[CHK](cr)**

~        delimiter character

AA    address of setting/response module(00 to FF)

O        command for set module name

(Data) new name for module, max 6 characters

**Response:** Valid command:          **!AA**

                       Invalid command:          **?AA**

**Example:**

Command:~01O9019                              Receive :!01

Set address 01 module name 9019, return success.

## 2.22 ~**

**Description:** Host OK.

Host send this command to all modules for send the information "Host OK"

**Syntax: ~**[CHK](cr)**

~        delimiter character

**        command for all modules

**Response:** No response.

**Example:**

Command: ~**        No response

# 2.23 ~AA0

**Description:** Read Module Host Watchdog Status.

**Syntax: ~AA0[CHK](cr)**

~  delimiter character

AA  address of reading/response module(00 to FF)

0  command for read module status

**Response:** Valid command:  **!AASS**

Invalid command:  **?AA**


SS  module status, 00=host watchdog timeout status is clear,04=host watchdog timeout status is set. The status will store into EEPROM and only may reset by the command~AA1.

# 2.24 ~AA1

**Description:** Reset Module Host Watchdog Status.

**Syntax: ~AA1[CHK](cr)**

~        delimiter character

AA      address of setting/response module(00 to FF)

1        command for reset module status

**Response:** Valid command:            **!AA**

Invalid command:            **?AA**

## 2.25 ~AA2

**Description:** Read Host Watchdog Timeout Value
**Syntax: ~AA2[CHK](cr)**
~       delimiter character
AA    address of reading/response module(00 to FF)
2       command for read host watchdog timeout value
**Response:** Valid command :          **!AAEVV**
                    Invalid command:          **?AA**

E      host watchdog enable status, 1=Enable, 0=Disable
VV    timeout value in HEX format, each count is 0.1 second
        01=0.1 second and FF=25.5 seconds

# 2.26 ~AA3EVV

**Description:** Set Host Watchdog Timeout Value

**Syntax: ~AA3EVV[CHK](cr)**

~       delimiter character

AA    address of setting/response module(00 to FF)

3       command for set host watchdog timeout value

E       1=Enable/0=Disable host watchdog

VV    timeout value, from 01 to FF, each for 0.1 second

**Response:** Valid command:       **!AA**

              Invalid command:       **?AA**

**Example:**

Command : ~010                      Receive : !0100

Read address 01 modules status, return host watchdog timeout status is clear.

Command : ~013164               Receive : !01

Set address 01 host watchdog timeout value 10.0 seconds and enable host watchdog, return success.

Command : ~012                      Receive : !01164

Read address 01 host watchdog timeout value, return that host watchdog is enabled, and time interval is 10.0 seconds.

Command : ~**                      No response

Reset the host watchdog timer.

Wait for about 10 seconds and don't send command~**, the LED of module will go to flash. The flash LED indicates the host watchdog timeout status is set.

Command : ~010                      Receive : !0104

Read address 01 module status, return host watchdog timeout status is set.

Command : ~012                              Receive : !01064

Read address 01 host watchdog timeout value, return that host watchdog is disabled, and time interval is 10.0 seconds.

Command : ~011                              Receive : !01

Reset address 01 host watchdog timeout status, return success And the LED of this module stop flash.

Command : ~010                              Receive : !0100

Read address 01 module status, return host watchdog timeout status is clear.

# 2.27 ~AABOE
## (For EX-9018BL/18BLM/19/19M only)

**Description:** Set Burnout(Break Line for Thermocouple) detect enable/disable of module

**Syntax:** ~AABOE[CHK](cr)

$      delimiter character

AA    address of reading/response module(00 to FF)

BO    command for Set Burnout(Break Line for Thermocouple) detect enable/disable

E      1=Enable/0=Disable burnout(Break Line for Thermocouple) detect

**Response:** Valid command:      **!AA**

          Invalid command:      **?AA**

## Example:

Command: ~01BO1            Response: !01

Sets the burnout(Break Line for Thermocouple) detect of module 01 to enable.

# 2.28 ~AAME
## (For EX-9018M/18BLM/19M only)

**Description:** Set the data format of channel's response in ModbusRTU mode

**Syntax:** ~AAME[CHK](cr)

$      delimiter character

AA    address of reading/response module(00 to FF)

M     command for Set the data format of response in ModbusRTU mode

E      1=2's complement / 0=engineer unit

**Response:** Valid command:        **!AA**

               Invalid command:       **?AA**

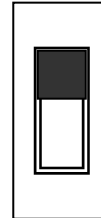## Example:

Command: ~01M1               Response: !01

Set the channel's response data format of module 01 to 2's complement in ModbusRTU mode.
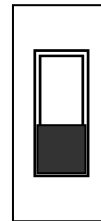
# EX-9018-M/9018BL-M/9019M Modbus Quick Start

**1. The default setting is MODBUS mode after Power On.**

**2. Sliding the INIT\* switch to the Init(ON) position**

**of rear side then Power On will enter INIT\* mode**

**(use ASCII command).**

**3. On ASCII command mode, user can set other setting like Address,**

   **Baudrate, …by use ASCII command or EX-9000 utility (Please check the**

   **EX-9000 user manual).**

**4. After change the setting finish, Sliding the INIT\***

**switch to the Normal(1) position of rear side, the**

**new setting will be effective after the next power-on**

**reset.**

The Modbus protocol was originally developed for Modicon controllers by Modicon Inc. Detailed information can be found at http://www.modicon.com/techpubs/toc7.html. Visit http://www.modbus.orq to find more valuable information.

9000M series modules support the Modbus RTU protocol. The communication Baud Rates range from 1200bps to 115200bps. The parity, data bits and stop bits are fixed as no parity, 8 data bits and 1stop bit. The following Modbus functions are supported.

**This function code is used to read from 1 to 8 continuous analog input channels.**

**Request**

| 00 | Address | 1Byte | 1 to 247 |
|---|---|---|---|
| 01 | Function code | 1Byete | 0x04 |
| 02-03 | Starting channel | 2 Bytes | 0 to 7 for reading analog inputs |
| 04-05 | Number of input Channels(N) | 2Bytes | 1 to 8;(Starting channel+N)<=8 for reading analog inputs |

**Response**

| 00 | Address | 1Byte | 1 to 247 |
|---|---|---|---|
| 01 | Function code | 1Byete | 0x04 |
| 02 | Byte count | 1 Byte | 2 x N |
| 03~ | Data of input channels | 2 x N Bytes | |

**Error Response**

| 00 | Address | 1Byte | 1 to 247 |
|---|---|---|---|
| 01 | Function code | 1Byete | 0x84 |
| | Exception code | 1 Byte | 02:starting channel out of range 03:( starting channel+number of input channels) out of range, incorrect number of bytes received |

# 01(0x01) Read WDT timeout status

## Request

| 00 | Address | 1 Byte | 1-247 |
|---|---|---|---|
| 01 | Function code | 1 Byte | 0x01 |
| 02~03 | Starting channel | 2 Bytes | 0x010D |
| 04~05 | Read WDT timeout status | 2 Bytes | 0x0001 |

## Response

| 00 | Address | 1 Byte | 1-247 |
|---|---|---|---|
| 01 | Function code | 1 Byte | 0x01 |
| 02 | Byte count | 1 Byte | 1 |
| 03 | Read WDT timeout status | 1 Byte | 0x00 The WDT timeout status is clear<br>0x01 The WDT timeout status is enable |

## Error Response

| 00 | Address | 1 Byte | 1-247 |
|---|---|---|---|
| 01 | Function code | 1 Byte | 0x81 |
| 02 | Exception code | 1 Byte | Refer to the Modbus standard for more details. |

# 03(0x03) Read WDT timeout Value

## Request

| 00 | Address | 1 Byte | 1-247 |
|----|---------|--------|-------|
| 01 | Function code | 1 Byte | 0x03 |
| 02~03 | Starting channel | 2 Bytes | 0x01E8 |
| 04~05 | Read WDT timeout value | 2 Bytes | 0x0001 |

## Response

| 00 | Address | 1 Byte | 1-247 |
|----|---------|--------|-------|
| 01 | Function code | 1 Byte | 0x03 |
| 02 | Byte count | 1 Byte | 2 |
| 03~ | Read WDT timeout value | **1 Byte** | 0x0000~0x00FF WDT timeout value, 0~255, in 0.1 second |

## Error Response

| 00 | Address | 1 Byte | 1-247 |
|----|---------|--------|-------|
| 01 | Function code | 1 Byte | 0x83 |
| 02 | Exception code | 1 Byte | Refer to the Modbus standard for more details. |

# 03(0x03) Send Host OK

## Request

| 00 | Address | 1 Byte | 1-247 |
|---|---|---|---|
| 01 | Function code | 1 Byte | 0x03 |
| 02~03 | Starting channel | 2 Bytes | 0x3038 |
| 04~05 | Send Host OK | 2 Bytes | 0x0000 |

## No Response

# 04(0x04) Send Host OK

## Request

| 00 | Address | 1 Byte | 1-247 |
|---|---|---|---|
| 01 | Function code | 1 Byte | 0x04 |
| 02~03 | Starting channel | 2 Bytes | 0x3038 |
| 04~05 | Send Host OK | 2 Bytes | 0x0000 |

## No Response

# 05(0x05) Set WDT timeout /Clear WDT timeout status

## Request

| 00 | Address | 1 Byte | 1-247 |
|---|---|---|---|
| 01 | Function code | 1 Byte | 0x05 |
| 02~03 | WDT timeout | 2 Bytes | 0x0104 Set WDT timeout enable/disable<br>0x010D Clear WDT timeout status |
| 04~05 | WDT timeout | 2 Bytes | 0xFF00 for WDT timeout enable<br>0x0000 for WDT timeout disable<br>0xFF00 for Clear WDT timeout status |

## Response

| 00 | Address | 1 Byte | 1-247 |
|---|---|---|---|
| 01 | Function code | 1 Byte | 0x05 |
| 02~03 | WDT timeout | 2 Bytes | The value is the same as byte 02 and 03 of the Request |
| 04~05 | WDT timeout | 2 Bytes | The value is the same as byte 04 and 05 of the Request |

## Error Response

| 00 | Address | 1 Byte | 1-247 |
|---|---|---|---|
| 01 | Function code | 1 Byte | 0x85 |
| 02 | Exception code | 1 Byte | Refer to the Modbus standard for more details. |

# 06(0x06) Set WDT timeout Value

## Request

| 00 | Address | 1 Byte | 1-247 |
|---|---|---|---|
| 01 | Function code | 1 Byte | 0x06 |
| 02~03 | Starting channel | 2 Bytes | 0x01E8 |
| 04~05 | WDT timeout value | 2 Bytes | 0x0000~0x00FF WDT timeout value, 0~255, in 0.1 second |

## Response

| 00 | Address | 1 Byte | 1-247 |
|---|---|---|---|
| 01 | Function code | 1 Byte | 0x06 |
| 02~03 | WDT timeout value | 2 Bytes | The value is the same as byte 02 and 03 of the Request |
| 04~05 | WDT timeout value | 2 Bytes | The value is the same as byte 04 and 05 of the Request |

## Error Response

| 00 | Address | 1 Byte | 1-247 |
|---|---|---|---|
| 01 | Function code | 1 Byte | 0x86 |
| 02 | Exception code | 1 Byte | Refer to the Modbus standard for more details. |

# 9018-M Modbus mapping:

| Input register address | | | |
|---|---|---|---|
| Analog input Value | 30001~30008 | R | 0x0000~0x7FFF |
| Cold junction temperature | 30129 | R | hex to dec then / 10 |
| Input type Code | 30201~30208 | R | 0x0000~0x0015 |
| Module name | 30483~30484 | R | 0x0090 0x1800 (9018M, "M"->0) |
| Channel enable | 30221 | R | 0x0000~0x00FF (0:off, 1:on) |
| Modbus data format | 30269 | R | 0x0000: engineer format, 0x0001: Hex 2's comp |
| Channel offset value | 30291~30298 | R | 0x8000~0x7FFF Hex 2's comp to dec then / 100 |
| **Holding register address** | | | |
| Analog input Value | 40001~40008 | R | 0x0000~0x7FFF |
| Cold junction temperature | 40129 | R | hex to dec then / 10 |
| Input type Code | 40201~40208 | R | 0x0000~0x0015 (check the table of user manual) |
| Module name | 40483~40484 | R | 0x0090 0x1800 (9018M, "M"->0) |
| Channel enable | 40221 | R/W | 0x0000~0x00FF (0:off, 1:on) |
| Modbus data format | 40269 | R/W | 0x0000: engineer format, 0x0001: Hex 2's comp |
| Channel offset value | 40291~40298 | R/W | 0x8000~0x7FFF Hex 2's comp to dec then / 100 |
| **Sub-function (0x46)** | | | |
| Module name | AA 46 00 | R | 01 46 00 **00 90 18 00** |
| Set module's address | AA 46 04 NN 00 00 00 | W | NN: new address, 01~F7(1~247) new address is effective after module reboot. |

# 9018BL-M Modbus mapping:

| Input register address | | | |
|---|---|---|---|
| Analog input Value | 30001~30008 | R | 0x0000~0x7FFF |
| Cold junction temperature | 30129 | R | hex to dec then / 10 |
| Input type Code | 30201~30208 | R | 0x0000~0x0015 |
| Module name | 30483~30484 | R | 0x0090 0x18B0 (9018BM, "M"->0) |
| Channel enable | 30221 | R | 0x0000~0x00FF (0:off, 1:on) |
| Modbus data format | 30269 | R | 0x0000: engineer format, 0x0001: Hex 2's comp |
| Burnout status | 30281 | R | 0x0000~0x00FF (0:normal, 1:breakline)<br>Note: only effective in thermocouple type |
| Channel offset value | 30291~30298 | R | 0x8000~0x7FFF Hex 2's comp to dec then / 100 |
| **Holding register address** | | | |
| Analog input Value | 40001~40008 | R | 0x0000~0x7FFF |
| Cold junction temperature | 40129 | R | hex to dec then / 10 |
| Input type Code | 40201~40208 | R | 0x0000~0x0015 (check the table of user manual) |
| Module name | 40483~40484 | R | 0x0090 0x18B0 (9018BM, "M"->0) |
| Channel enable | 40221 | R/W | 0x0000~0x00FF (0:off, 1:on) |
| Modbus data format | 40269 | R/W | 0x0000: engineer format, 0x0001: Hex 2's comp |
| Burnout status | 40281 | R | 0x0000~0x00FF (0:normal, 1:breakline)<br>Note: only effective in thermocouple type |
| Channel offset value | 40291~40298 | R/W | 0x8000~0x7FFF Hex 2's comp to dec then / 100 |
| **Sub-function (0x46)** | | | |
| Module name | AA 46 00 | R | 01 46 00 **00 90 18 B0** |
| Set module's address | AA 46 04 NN 00 00 00 | W | NN: new address, 01~F7(1~247)<br>new address is effective after module reboot. |

# 9019-M Modbus mapping:

| Input register address | | | |
|---|---|---|---|
| Analog input Value | 30001~30008 | R | 0x0000~0x7FFF |
| Cold junction temperature | 30129 | R | hex to dec then / 10 |
| Input type Code | 30201~30208 | R | 0x0000~0x0015 |
| Module name | 30483~30484 | R | 0x0090 0x1900 (9019M, "M"->0) |
| Channel enable | 30221 | R | 0x0000~0x00FF (0:off, 1:on) |
| Modbus data format | 30269 | R | 0x0000: engineer format, 0x0001: Hex 2's comp |
| Burnout status | 30281 | R | 0x0000~0x00FF (0:normal, 1:breakline) <br> Note: only effective in thermocouple type |
| Channel offset value | 30291~30298 | R | 0x8000~0x7FFF Hex 2's comp to dec then / 100 |
| **Holding register address** | | | |
| Analog input Value | 40001~40008 | R | 0x0000~0x7FFF |
| Cold junction temperature | 40129 | R | hex to dec then / 10 |
| Input type Code | 40201~40208 | R/W | 0x0000~0x0015 (check the table of user manual) |
| Module name | 40483~40484 | R | 0x0090 0x1900 (9019M, "M"->0) |
| Channel enable | 40221 | R/W | 0x0000~0x00FF (0:off, 1:on) |
| Modbus data format | 40269 | R/W | 0x0000: engineer format, 0x0001: Hex 2's comp |
| Burnout status | 40281 | R | 0x0000~0x00FF (0:normal, 1:breakline) <br> Note: only effective in thermocouple type |
| Channel offset value | 40291~40298 | R/W | 0x8000~0x7FFF Hex 2's comp to dec then / 100 |
| **Sub-function (0x46)** | | | |
| Module name | AA 46 00 | R | 01 46 00 **00 90 19 00** |
| Set module's address | AA 46 04 NN 00 00 00 | W | NN: new address, 01~F7(1~247) <br> new address is effective after module reboot. |

## MODBUS Engineering Data Format Table

| Type Code | Input Type | Min. | Max. | Formula |
|---|---|---|---|---|
| 00 | -15mV ~ +15mV | -15000 | 15000 | Volt=(Modbus data)/1000(mV) |
| 01 | -50mV ~ +50mV | -5000 | 5000 | Volt=(Modbus data)/100(mV) |
| 02 | -100mV ~ +100mV | -10000 | 10000 | Volt=(Modbus data)/100(mV) |
| 03 | -500mV ~ +500mV | -5000 | 5000 | Volt=(Modbus data)/10(mV) |
| 04 | -1V ~ +1V | -10000 | 10000 | Volt=(Modbus data)/10000(V) |
| 05 | -2.5V ~ +2.5V | -25000 | 25000 | Volt=(Modbus data)/10000(V) |
| 06 | -20mA ~ +20mA | -20000 | 20000 | Current=(Modbus data)/1000(mA) |
| 0E | Type J:  -210°C to 760°C | -2100 | 7600 | Temp.=(Modbus data)/10(°C) |
| 0F | Type K:  -270°C to 1372°C | -2700 | 13720 | |
| 10 | Type T:  -270°C to 400°C | -2700 | 4000 | |
| 11 | Type E:  -270°C to 1000°C | -2700 | 10000 | |
| 12 | Type R:  0°C to 1768°C | 0 | 17680 | |
| 13 | Type S:  0°C to 1768°C | 0 | 17680 | |
| 14 | Type B:  0°C to 1820°C | 0 | 18200 | |
| 15 | Type N:  -270°C to 1300°C | -2700 | 13000 | |

Example: Assume type of channel is +/-10V and MODBUS data=0x2030(Hex)=8240(Dec)
The voltage of channel is 8240/1000=8.24V

Example: Assume type of channel is +/-500mV and MODBUS data=0xEF1B(Hex)=-4325(Dec)
The voltage of channel is -4235/10=423.5mV

Example: Assume type of channel is +/-20mA and MODBUS data=0x3B84(Hex)=15236(Dec)
The current of channel is 15236/1000=15.236mA

## MODBUS Hex 2's complement Data Format Table

| Type Code | Input Type | Min. | Max. | Formula |
|---|---|---|---|---|
| 00 | -15mV ~ +15mV | 8000 | 7FFF | Volt=(Modbus data*15)/0x7FFF(mV) |
| 01 | -50mV ~ +50mV | 8000 | 7FFF | Volt=(Modbus data*50)/0x7FFF(mV) |
| 02 | -100mV ~ +100mV | 8000 | 7FFF | Volt=(Modbus data*100)/0x7FFF(mV) |
| 03 | -500mV ~ +500mV | 8000 | 7FFF | Volt=(Modbus data*500)/0x7FFF(mV) |
| 04 | -1V ~ +1V | 8000 | 7FFF | Volt=(Modbus data*1)/0x7FFF(mV) |
| 05 | -2.5V ~ +2.5V | 8000 | 7FFF | Volt=(Modbus data*2.5)/0x7FFF(mV) |
| 06 | -20mA ~ +20mA | 8000 | 7FFF | Current=(Modbus data*20)/0x7FFF(mA) |
| 0E | Type J:  -210°C to 760°C | DCA2 | 7FFF | Temp.=(Modbus data*760)/0x7FFF(°C) |
| 0F | Type K:  -270°C to 1372°C | E6D0 | 7FFF | Temp.=(Modbus data*1372)/0x7FFF(°C) |
| 10 | Type T:  -270°C to 400°C | A99A | 7FFF | Temp.=(Modbus data*400)/0x7FFF(°C) |
| 11 | Type E:  -270°C to 1000°C | DD71 | 7FFF | Temp.=(Modbus data*1000)/0x7FFF(°C) |
| 12 | Type R:  0°C to 1768°C | 0000 | 7FFF | Temp.=(Modbus data*1768)/0x7FFF(°C) |
| 13 | Type S:  0°C to 1768°C | 0000 | 7FFF | Temp.=(Modbus data*1768)/0x7FFF(°C) |
| 14 | Type B:  0°C to 1820°C | 0000 | 7FFF | Temp.=(Modbus data*1820)/0x7FFF(°C) |
| 15 | Type N:  -270°C to 1300°C | E56B | 7FFF | Temp.=(Modbus data*1300)/0x7FFF(°C) |

Example: Assume type of channel is +/-10V and MODBUS data=0x2030(Hex)=8240(Dec)
The voltage of channel is (8240*10)/32767=2.514V

Example: Assume type of channel 1 is +/-500mV and MODBUS data=0xEF1B(Hex)=-4325(Dec)
The voltage of channel is (-4235*500)/32767=-64.622mV

Example: Assume type of channel 1 is +/-20mA and MODBUS data=0x3B84(Hex)=15236(Dec)
The current of channel is (15236*20)/32767=9.299mA