

# ISO AD A08

## 8-channel Isolated 4-20mA Signal Acquisition Module

### I Product Introduction

#### Features:

- 8-ch 4-20mA signal acquisition, signal between channel to channel is isolated
- RS-485/232 output can adjust the accuracy of module by programme
- Testing accuracy > 0.05%
- Three-port isolation among Signal Input/output/power supply: 3000VDC
- Wide power supply: 8~50VDC
- High reliability, convenient programming, easy to install
- Users can set module address, baud rate, etc. by programming
- Support Modbus RTU communication protocol

#### Applications:

- signal testing/monitoring/controlling
- Utility system such as intelligent building control and security monitoring project
- RS-232/485 industrial automation control system
- Industrial field signal isolation and long-line transmission
- Equipment operating monitoring
- Sensor signal testing
- Acquisition and record of industrial field data
- Development of medical and industrial products
- 4-20mA signal acquisition

#### Product introduction:

ISO AD series converter achieve the signal isolated acquisition between sensor and the host and can be used for a variety of sensors .ISO AD series of products can be used in industrial automation RS-232/485 bus system,4-20mA signal measurement, monitoring and control, small-signal measurement, and industrial-site signal isolation and long-term transmission and so on.

Products include power supply isolation, signal isolation, linearization, A/D conversion and RS-485 serial communication circuit module. Each serial port can connect maximum 256 PCS of ISOAD series modules, means of communication using character communication protocol or MODBUS RTU communication protocol. Its instruction set compatible with the ADAM modules, the baud rate is set by the user, with other manufacturers control module hanging in the same RS-485 bus, to facilitate host programming.

ISO AD series of products are based on single chip of intelligent monitoring and control systems, all the user sets such as the calibration value, address, baud rate, data format, checksum status and other configuration information is stored in nonvolatile memory EEPROM inside.

ISODA series of products according to industry standard design, manufacture, signal output / communication interface isolation, can withstand 3000VDC isolation voltage, anti-interference ability, high reliability. Operating temperature range: - 45°C ~ +80°C .

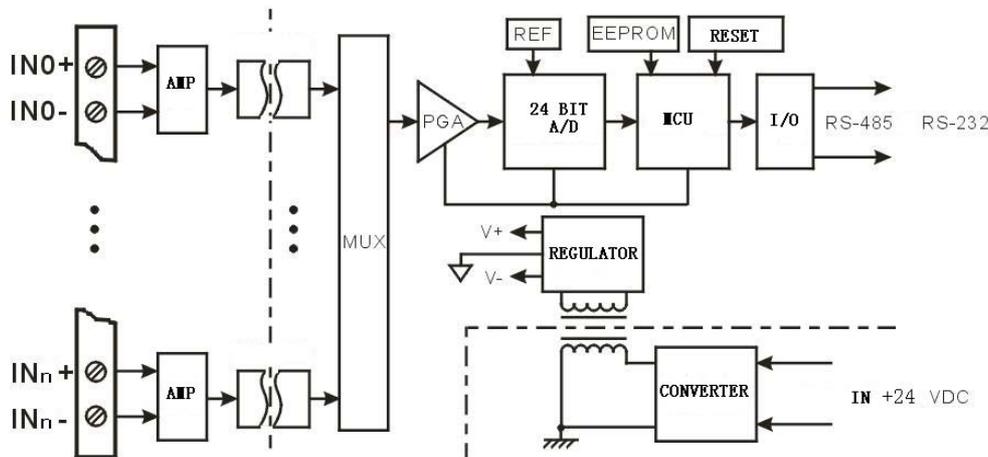


Figure 1 ISOAD A08 functional scheme

**2、 Function of ISOAD A08:**

ISOAD A08 is a 8 channel signal isolation acquisition module which is used for measuring 4-20mA/0-20mA current signal.

2.1 analog signal input

The acquisition precision of the product is up to the 24 bits. There are 8 channels for 4-20mA signal input. Every channel has been isolated and all the signals of Sunyuan products have been calibrated before delivery. Users could also program and calibrate the signals according to their needs by themselves easily.

2.2 Communication protocol

Communication interface: 1 Road Standard RS-485 communication interface, or one road standard RS-232 communication interface, which could be selected when ordering.

Communication Protocol: support standard Modbus RTU protocol and characters protocol. The communication protocol could be used by programming. Network communication is achieved between it and a variety of brands of PLC, RTU or computer monitoring system.

Data format: 10 bits. 1 start bit, 8 data bits, 1 stop bit.

Address: module address (0x00 ~ 0xFF) and baud rate (300 bps, 600 bps, 1200 bps, 2400 bps, 4800 bps, 9600 bps, 19200 bps, 38400bps) can be set; the communication networks, connected by twisted-pair shielded cable, could reach up to 1200 meters. Communication interface is of high anti-jamming design, ± 15KV ESD protection, and the communication response time is less than 100mS.

2.3 Anti-interference

The checksum could be set customized. Inside the module there is transient suppression diodes which can inhibit a variety of surge pulse and protect module and internal digital filter from grid frequency interference.

**3、 Model selection:**

ISOAD A□ - □

Enter the channel code \_\_\_\_\_ communication interface

code range: 01 ~ 08, number of channel

**01:** 1 channel 4-20mA/0-20mA signal input

**485:** output is RS-485 interface

**02:** 2 channel 4-20mA/0-20mA signal input

**232:** output is RS-232 interface

**03:** 3 channel 4-20mA/0-20mA signal input

- 
- 
- 

**08:** 8 channel 4-20mA/0-20mA signal input

Note 1: If the selected product has less than 8 channels, it is the same AD A08 appearance, wiring scheme, and communication protocol with ISO. The only difference is that the selected one has less channels to measure data.

Mode selection eg.1: Model: **ISOAD A01-485** Indicates 1 channel, 4-20mA/0-20mA signal input, input is RS-485 interface

Mode selection eg.2: Model: **ISOAD A06-232** Indicates 6 channel, 4-20mA/0-20mA signal input, input is RS-232interface

Mode selection eg.3: Model: **ISOAD A08-485** Indicates 8 channel, 4-20mA/0-20mA signal input, input is RS-485interface

**4、ISOAD A08 general parameter:**

(typical @ +25°C, Vs is 24VDC (It is recommended to use power supply under high stability and precision environment.)

Input type: input current 4-20mA/0-20mA

Precision: @ +25°C (table 1)

Input detuning: ±0.1 uA/°C

Temperature drift: ±15 ppm/°C (±30 ppm/°C, max)

Input resistance: < 250 Ω (4-20mA/0-20mA input current)

Bandwidth: -3 dB 10 Hz

Conversion rate: 10 Sps

Common mode restraint (CMR): 120 dB (1kΩ Source Imbalance @ 50/60 Hz)

Norm mode restraint (NMR): 60 dB (1kΩ Source Imbalance @ 50/60 Hz)

Input terminal protection: over-voltage protection and over-current protection

Communication: support standard Modbus RTU protocol and character communication protocol, RS-485 or RS232 interface.

Baud rate (300 bps、600 bps、1200 bps、2400 bps、4800 bps、9600 bps、19200 bps、38400bps) software selection is available

Location (0x00~0xFF) software selection is available

Communication response time: 100 ms max

Power supply: +8 ~ 50 VDC wide range power supply, internal anti-reverse and overvoltage protection circuit

Power consumption: less than 1W

Operating temperature: - 45 ~ +80°C

Humidity: 10 ~ 90% (no condensation)

Storage temperature: - 45 ~ +80 °C

Storage humidity: 10 ~ 95% (no condensation)

Isolation voltage: input channel 1 / input channel 2 / ... / input channel 8 / output / auxiliary power supply between every two group: 3KVDC, 1 minute, leakage current 1mA.

Shock voltage: 3KVAC, 1.2/50us (peak)

Dimensions: 145 mm x 82 mm x 42mm

Weight: about 256g

**ISO-AD-A08 Accuracy and Non linearity limit parameter: @+25°C 24VDC**

Range Description	Accuracy (Typical)	Accuracy (Maximum)	Non linearity limit (Maximum)	Noise (Peak-to-Peak)
Current/voltage	±0.02 % FS	±0.05 % FS	±0.05 % FS	±0.01 % FS

**Table 1** ISOAD-A08 measuring accuracy and Non linearity limit

Note 2: The input signal isolation use passive isolating technology, the conversion accuracy of input signal will be reduced when near zero input, so for the 0-20mA input, the accuracy in Table 1 is accuracy of 0.2mA ~ 20mA input (1%~100%), and accuracy of 0.5% FS is in the input 0mA ~ 0.2mA( 0% to 1%),.

## II. Pin description and wiring diagram

### 1、 Pin description :

pin	name	description
1	PW+	Power +
2	PW-	Power -
3	CONFIG	Config
4	GND	Output Ground
5	TXD	RS-232 send port
6	RXD	RS-232 receive port
7	DATA+	RS-485 +
8	DATA-	RS-485 -
9	IN6+	Channel 6 +
10	IN6-	Channel 6 -
11	IN7+	Channel 7 +
12	IN7-	Channel 7 -
13	IN0+	Channel 0 +
14	IN0-	Channel 0 -
15	IN1+	Channel 1 +
16	IN1-	Channel 1 -
17	IN2+	Channel 2 +
18	IN2-	Channel 2 -
19	IN3+	Channel 3 +
20	IN3-	Channel 3 -
21	IN4+	Channel 4 +
22	IN4-	Channel 4 -
23	IN5+	Channel 5 +
24	IN5-	Channel 5 -



Table 2 Pin description

Figure 2 ISOAD A08 module scheme

Note 3: If the selected product channel number (N) is less than 8, then the available channel is from channel 0 to channel (N-1). There is none electrical connections in wiring terminals of Channel N to channel 7 and none inside the product, so they are invalid channels

### 1、 Wiring diagram :

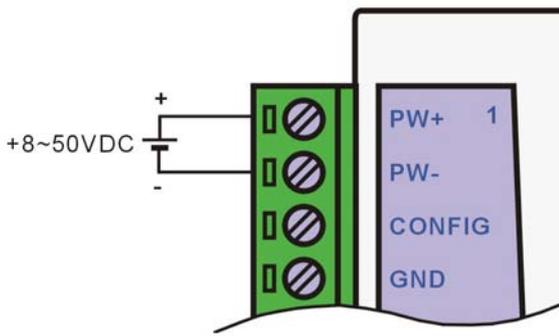


Figure 3 ISOAD-A08 power supply wiring

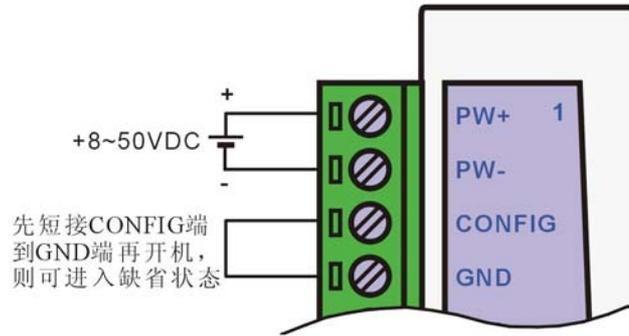


Figure 4 ISOAD-A08 default status wiring

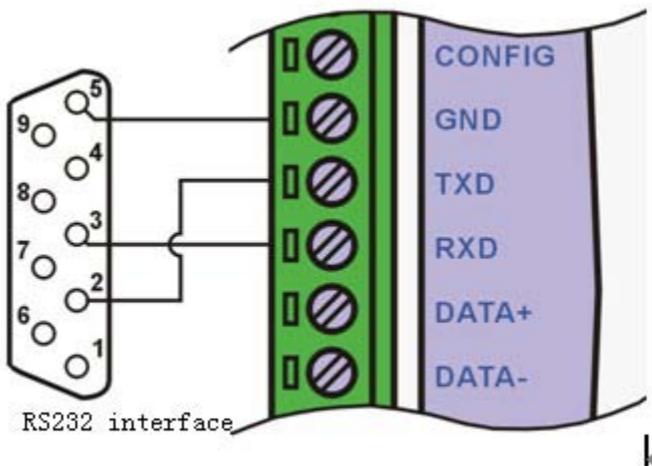


Figure 7 ISOAD-A08 RS-232 interface wiring

### III. Module Initialization and Allocation

#### 1、Initialization of ISOAD A08 module:

All ISOAD A08 module, if using RS-485 network, must be allocated a unique address code, the range of which is between 00 to FF. However, all new ISOAD A08 module use a initial settings made by factory as follows:

Address code is 01

9600 bps baud rate

Checksum is disabled

Character communication protocol

As the new module code is the same, their address will be contradictory with other modules, when you set up the system, you must re-configure each analog input module address. Users can modify ISOAD A08 module's address through configuration commands after connecting ISOAD A08 module power cable and RS485 communication lines. Baud rate, parity, status, and communication protocols can also be adjusted according to user's requirements. Before the adjusting process, one must set the module to the default state firstly; otherwise it can not be modified.

**2、 Module allocation under default state:**

All ISOAD A08 modules have a special pin labeled CONFIG. Connecting CONFIG pin to circuit ground (GND pin) (as Figure 4), and then switch on the power, the module will enter into the default state. Under this state, the module is configured as follows:

- Address code is 00
- 9600 bps baud rate
- Checksum is disable
- Character communication protocol

At this time, users can modify ISOAD A08 module, baud rate, checksum state and other parameters by configuring the command and select the means of communication by setting the module's communication protocol command. If the configuration of a module is uncertain, the module could be in default state by installing the jumper and then reconfigure the module.

If user needs the module to be set to MODBUS RTU communication protocol, see introduction of MODBUS communication protocol section for information.

**IV. ISOAD A08 Command Set Under Character Communication Protocol**

Command is constituted by a series of characters, such as the leading code, address ID, variables, optional checksum byte and a character to show the terminal of command (cr). The host machine, with a wildcard address “\*\*” synchronization command, only command a ISOAD A08 module once a time.

Command format: **(Leading Code)(Addr)(Command)[data][checksum](cr)**

**(Leading code)** It is the first letter of a command. Every command requires a leading code, such as %, \$, #, @, ... etc. **1- C**

**(Addr)** Address code of a module. Its range is from 00 to FF (hexadecimal) if there is no allocation. **2- C**

**(Command)** Display command code or variable value. **Extent of variable value**

**[data]** Some output command requires. **Extent of variable value**

**[checksum]** Checksum in the bracket displays optional parameters. It is required only when checksum is used. **2- C**

**(cr)** a control code character which is used as the end code. It is 0x0D. **1- C**

[Checksum] is needed when enabling checksum. It takes up 2 characters. Commands and responses must have checksum and feature. Checksum is used to check all input commands to help user finding the command error from the host machine to the module, or response error from module to the host. Checksum character follows command or response character and in front of the carriage return. Calculation method: Get the sum of ASCII code value of all prior issued bytes. This sum and hexadecimal digits 0xFF is two characters (the hexadecimal number).

E.g. prohibit checksum

- User command **\$002(cr)**
- Module response **!00020600 (cr)**
- Enable checksum (checksum)
- User command **\$022B8 (cr)**
- Module response **!02000640 AD (cr)**
- ‘\$’ = 0x24 ‘0’ = 0x30 ‘2’ = 0x32
- B6=(0x24+0x30+0x30+0x32) AND 0xFF

'1' = 0x21    '0' = 0x30    '2' = 0x32    '6' = 0x36  
 A9=(0x21+0x30+0x30+0x30+0x32+0x30+0x36+0x30+0x30) AND 0xFF

**Response of commands :**

Response message depends on a variety of commands. Response also consists of several characters, including leading code, variables and end tags. There are two leading codes of response signal: '!' or '>' indicates a valid command and '?' means invalid. By checking the response information, user can monitor whether the command is valid or not.

Note: 1, in some cases, many commands use the same command syntax. User needs to ensure the address he uses is correct in a command. If one use the wrong address and this address represents another module, then the command will take effect in another module, resulting in an error.

2, the command must be entered in capital letters.

**1、 Read the data of analog input module**

Instruction: Read the analog input data of all channels at the current allocated data format.

Parameter instruction: #        leading code of command

AA    module address, range 00~FF (hexadecimal)

(cr)    end character, upper computer Return key (0DH) .

Response grammar: >(data)(cr)        valid command

?AA(cr)        invalid command or illegal operation

Parameter instruction: >        response leading code

(data) On behalf of all the channel data been read. Data format is engineering units, FSR percentage, and hexadecimal complement number. Details see section 3 of the command set.

(cr)    end character, upper computer Return key (0DH) .

More instructions: If there is a syntax error or communication error or address does not exist, the module does not respond.

If a channel has been shut down, then the data read out as space character.

E.g. user command        #23(cr)

Module

response

>+04.765+04.756+04.632+04.000+05.001+06.000+08.800+16.000(cr)

Instruction: the input current of every channel on address 23H (data format is engineering unit):

Channel 0: +04.765mA    Channel 1: +04.756mA    Channel 2: +04.632mA    Channel

3: +04.000mA

Channel 4: +05.001mA    Channel 5: +06.000mA    Channel 6: +08.800mA    Channel 7 :  
 +16.000mA

**2、 Read the data of analog input module in Channel N**

Instruction: Read the analog input data of channel N at the current allocated data format.

Command grammar: #AAN(cr)

Parameter instruction: #        leading code of command

AA    module address, range 00~FF (hexadecimal)

N        channel code 0 ~ 7

(cr)    end character, upper computer Return key (0DH)

Response grammar: >(data)(cr)        valid command

?AA(cr)        invalid command or illegal operation

Parameter instruction: >        response leading code

(data) data read in channel N. Data format is engineering units, FSR percentage, and hexadecimal complement number. Details see section 3 of the command set.

(cr)    end character, upper computer Return key (0DH)

More instructions: If there is a syntax error or communication error or address does not exist, the module

does not respond.

E.g. user command #230(cr)  
 Module response >+04.632 (cr)

Instruction : The input current of channel 0 on address 23H module is +04.632mA (data format is engineering unit)

**3、 analog input module allocation**

Instruction: Set address, input range, baud rate, data format, checksum status for an analog input module. Configuration information is stored in nonvolatile memory EEPROM.

Command grammar: %AANNTCCFF(cr)

Parameter instruction: % leading code

- AA module address, dereference range 00~FF(hexadecimal).
- NN on behalf of hexadecimal address, NN is from 00 to FF.
- TT hexadecimal represents type code. ISOAD A08product should be set as 00.
- CC hexadecimal represents baud rate code.

baud rate code	baud rate
01	300bps
02	600bps
03	1200bps
04	2400bps
05	4800bps
06	9600bps
07	19200bps
08	38400bps

Table 3 baud rate code

FF The 8-bit of hexadecimal is used to represent data format and checksum. Note: From bits2 to bits 5 is unnecessary to be set to 0.

Bit7	Bit 6	Bit 5	Bit 4	Bit 3	Bit2	Bit 1	Bit 0
------	-------	-------	-------	-------	------	-------	-------

Table 4 data format, checksum code

- Bit7: Reserved bits should be set to 0**
- Bit6: checksum status: 0 represents prohibit; 1 represents permission**
- Bit5-bit2:** should be set to 0
- Bit1-bit0:** data format bit
  - 00: Engineering Units
  - 01: % of FSR
  - 10: complement of hexadecimal (Twos complement)
  - 11: ohms(only the thermal resistance could be set)

(cr) end character, upper computer Return key (0DH)

Response grammar: !AA(cr) command is valid.

?AA(cr) invalid command or illegal operation, no jumper allocation before baud rate checksum status.

Parameter instruction: ! Response leading code indicates valid command.

? Response leading code indicates invalid command.

**AA** indicates input module address  
**(cr)** end character, upper computer Return key (0DH)。

More instructions: 1. If need to reallocate module, user should install jumper to let the module be in default status. The wiring method is as Figure 4. In this way, module address is 00。

2. The new address will be set as response information and be send back to the host machine by the module after reallocation.

If there is a syntax error or communication error or address does not exist, the module does not respond.

e.g. User command **%0011000600(cr)**  
module response **!11(cr)**

Instruction: **%** Leading code of the command  
**00** Indicate the analog module original address which user wants to allocate is 00H.  
**11** Indicate the hexadecimal address of the new address is 11H.  
**00** type code, ISOAD A08 series should be set to 00.  
**06** Indicate baud rate9600bps。  
**00** Indicate data format is engineering unit and the checksum is forbidden.

#### 4. command to read the allocation status

Instruction: read the allocation of the specified module

Command grammar: **\$AA2(cr)**

Parameters instruction: **\$** leading code of the command

**AA** module address, dereference range00~FF(hexadecimal).  
**2** read the allocation status  
**(cr)** end character, upper computer Return key (0DH)。

Response grammar: **!AATTCCFF(cr)** valid command

**?AA(cr)** invalid command or illegal operation

Parameters instruction: **!** Response leading code

**AA** input module address。  
**TT** Type code  
**CC** baud rate code, table 3 for reference  
**FF** table 4 for reference  
**(cr)** end character, upper computer Return key (0DH)。

More instructions: If there is a syntax error or communication error or address does not exist, the module does not respond.

E.g.: user's command **\$302(cr)**  
module response **!30000600(cr)**

Instructions: **!** Response leading code  
**30** analog input module address is 30H 。  
**00** input type code  
**06** baud rate9600 bps。  
**00** data format is engineering unit and the checksum is forbidden.

#### 5. Offset calibration command

Instructions: Adjust the offset of channel N

Command grammar: **\$AA1N(cr)**

Parameters instruction: **\$** Leading code of command

**AA** module address, dereference range00~FF(hexadecimal).  
**1** Misregistration adjusting command

N channel code 0~7

(cr) end character, upper computer Return key (0DH)。

Response grammar: !AA(cr) valid command

?AA(cr) invalid command or illegal operation

Parameters instruction: ! Response leading code which means valid command

? Response leading code which means invalid command

AA input module address

(cr) end character, upper computer Return key (0DH)。

More instructions:

The product has been calibrated before delivery, and the user can directly use it without calibration. When calibrating an analog input, offset calibration should be done firstly, and then calibrate the gain. During the calibration, analog input module needs to be connected with the appropriate input signal on the channel which is to be calibrated. Different input range requires different input voltage or current. See the calibration module section.

If there is a syntax error or communication error or address does not exist, the module does not respond.

E.g.: user's command **\$2310(cr)**

module response **!23(cr)**

Instructions: offset calibrating channel 0 of address 23H

## 6、Gain calibration command

Instructions: calibrate the gain of channel N

Command grammar: \$AA0N(cr)

Parameters instruction: \$ Leading code of command

AA module address, dereference range 00~FF(hexadecimal)。

0 Gain calibration command

N channel No. 0~7

(cr) end character, upper computer Return key (0DH)。

Response grammar: !AA(cr) valid command

?AA(cr) invalid command or illegal operation

Parameters instruction: ! Response leading code means valid command

? Response leading code means invalid command

AA Input module address

(cr) end character, upper computer Return key (0DH)。

More instructions: The product has been calibrated before delivery, and the user can directly use it without calibration.

When calibrating an analog input, offset calibration should be done firstly, and then calibrate the gain. During the calibration, analog input module needs to be connected with the appropriate input signal on the channel which as to be calibrated. Different input range requires different input voltage or current. See the calibration module section.

If there is a syntax error or communication error or address does not exist, the module does not respond.

E.g.: user's command **\$2303(cr)**

module response **!23(cr)**

Instructions: gain calibrating channel 0 of address 23H

## 7、Read module name

Instructions: read the name of the analog input module

Command grammar: **\$AAM(cr)**

Parameters instruction: \$ Leading code of command

**AA** module address, dereference range 00~FF(hexadecimal)。

**M** read the name of module

**(cr)** end character, upper computer Return key (0DH)。

Response grammar: **!AA(ModuleName)(cr)** valid command

**?AA(cr)** invalid command or illegal operation

Parameters instruction: **!** Response leading code means valid command

**?** Response leading code means invalid command

**AA** input module address

**(ModuleName)** Module name could be ISOADA08、ISO 4011 or ISO4014 etc. represent the number of module you used.

**(cr)** end character, upper computer Return key (0DH)。

More instructions: If there is a syntax error or communication error or address does not exist, the module does not respond.

E.g.: user's command **\$08M(cr)**

module response **!08ISOAD A08 (cr)**

Instructions: Module ISOAD A08 is in address 08H module

## 8、 Enable or prohibit command

Instructions: Enable or prohibit the data acquisition channel of the module

Command grammar: **\$AA5VV(cr)**

Parameters instruction: \$ Leading code of command

**AA** module address, dereference range 00~FF(hexadecimal)。

**5** The command means enable or prohibit the data acquisition channel of the module

**VV** two bids (hexadecimal), 3~0 at the first bid represents 7~2 channel.

3~0 at the second bid 3~0 channel

Bit is 0: prohibit the channel

Bit is 1: enable the channel

**(cr)** end character, upper computer Return key (0DH)。

Response grammar: **!AA(cr)** valid command

**?AA(cr)** invalid command or illegal operation

Parameters instruction: **!** Response leading code means valid command

**?** Response leading code means invalid command

**AA** input module address。

**(cr)** end character, upper computer Return key (0DH)。

More instructions: If there is a syntax error or communication error or address does not exist, the module does not respond.

E.g.: user's command **\$08537(cr)**

module response **!08 (cr)**

Instructions: set the channel to be 0x37。

3 is 0011, represents enable channel 5 and channel 4, and prohibit channel 7 and channel 6.

7 is 0111, represents enable channel 2, channel 1 and channel 0, and prohibit channel 3.

## 9、 Read the channel status

Instructions: read the channel status of analog input module

Command grammar: **\$AA6(cr)**

Parameters instruction: \$      Leading code of command  
AA    module address, dereference range 00~FF(hexadecimal).  
6      read the channel status  
(cr)   end character, upper computer Return key (0DH)。

Response grammar: !AAV(cr)    valid command  
?AA(cr)    invalid command or illegal operation

Parameters instruction: !      Response leading code means valid command  
?      Response leading code means invalid command。  
AA    input module address。  
VV    two bids (hexadecimal), bits 3~0 of the first number represent 7~4 channel  
              Bids 3~0 of the second number represent 3~0 channel  
              Bit is 0: prohibit the channel  
              Bit is 1: enable the channel  
(cr)   end character, upper computer Return key (0DH)。

More instructions: If there is a syntax error or communication error or address does not exist, the module does not respond.

E.g.:    user's command    **\$186 (cr)**  
              module response    **!18FF(cr)**

Instructions: the current channel status is 0xFF.  
0xFF is 1111 and 1111, means all the channels of module at address 18H have been enabled.

## 10、Set communication protocol command

Instructions: Set the communication protocol of module as character communication protocol or Modbus RTU protocol.

Command grammar: \$AAPV(cr)

Parameters instruction: \$      Leading code of command  
AA    module address, dereference range 00~FF(hexadecimal).  
P      set communication protocol  
V      the code of the protocol could be 0 or 1.  
0:    character communication protocol  
1:    Modbus RTU protocol  
(cr)   end character, upper computer Return key (0DH)。

Response grammar: !AA(cr)    valid command  
?AA(cr)    invalid command or illegal operation

Parameters instruction: !      Response leading code represents valid command  
?      Response leading code means invalid command。  
AA    input module address。  
(cr)   end character, upper computer Return key (0DH)。

More instructions: If there is a syntax error or communication error or address does not exist, the module does not respond.  
Setting communication protocol command is only valid under default status.

E.g.1:    user's command    **\$00P1(cr)**  
              module response    **!00 (cr)**

Instructions: set communication protocol as Modbus RTU protocol.

E.g.2:    user's command    **\$00P0(cr)**  
              module response    **!00 (cr)**

Instructions: Set the communication protocol of module as character communication protocol

V. The input range and data format

There are four data format which are been used by analog input module: 00: (Engineering Units)  
 01: (% of FSR)  
 10: hexadecimal (complement)  
 11: ohms (only thermal resistance product could set ohms.)

Input range	Data format	Scale span	Zero	resolution ratio
4~20mA	Engineering Units	+20.000	+04.000	0.1uA
	% of FSR	+100.00	+020.00	0.01%
	complement of hexadecimal	7FFFFFFF	199999	1LSB
0~20mA	Engineering Units	+20.000	+00.000	1uA
	% of FSR	+100.00	+000.00	0.01%
	complement of hexadecimal	7FFFFFFF	000000	1LSB

Table5 input range and data format

E.g.:

1、input range is 4~20mA, input is 4 mA:

user's command #01 (cr)  
 engineering unit module response >+04.000(cr)  
 % of FSR module response >+020.00(cr)  
 Complement of hexadecimal module response >199999(cr)

2、conversion relation between hexadecimal data and analog

Using 24 bids AD converting clips improves the precision of the product greatly. The analog is converted to be hexadecimal data by AD clips as the following formula shows.

$$\frac{X}{7FFFFFFF} = \frac{I_{in} R_0}{I_m R_0}$$

Instructions: X: represents the corresponding hexadecimal data of the analog.

Iin: represents the analog signal the channel input.

Im: represents the analog of the scale span of product, 20mA.

R0: represents sampling resistor.

Data read by the RTU is the high-volume four hexadecimal data of conversion. In the calculation of the value of the corresponding data in engineering units, it is necessary to fill the last two hexadecimal data 00 (set as six bids hexadecimal data), otherwise the results might not be accurate. If needs to view all the six bits hexadecimal data, the user could select the complement hexadecimal form in the output data

format.

For example, if the user checked the data is 1999 on 4001 register by Modbus software, the calculation process is as follows:

1. after the complement, the 6 bits hexadecimal data of 1999 is 199900.
2. convert 199900 and 7 FFFFF into decimal data.
3. calculate the  $I_{in}$  value of analog input

## **VI. Calibration of Module**

The product has been calibrated before delivery, and the user can directly use it without calibration.

The user could recalibrate the module via the calibration function of the product. During the calibration, analog input module needs to be connected with the appropriate input signal on the channel which as to be calibrated. Different input range requires different input signal.

In order to improve calibration accuracy, it is recommended to use the following equipment:

- 1, a stable output, low noise DC voltage / current signal source
- 2, one five and a half bits or a higher-precision voltage / current measuring instruments to monitor the accuracy of the input signal

### **The calibration process**

1. Select the input channel to be calibrated and connect to the corresponding input signal according to the input range of the module  
When the input is 4mA, ISOAD A08 module is zero calibrated; when the input is 24mA, full-scale is calibrated.
2. Input signal current 4mA to the channel which needs calibration.
3. When the signal is stable, sent offset calibration \$ AA1N command (N represents the channel code currently being calibrated) to the analog input module.
4. Input signal current 24mA to the channel which needs calibration.
5. When the signal is stable, sent gain calibration \$ AA0N command (N represents the channel code currently being calibrated) to the analog input module.
6. Set offset and gain calibration for each channel in turn.
7. Calibration is completed.

## **VII. Brief introduction of Modbus communication protocol**

Modbus protocol defines the information structure which could be identified and used by controller. When communicating on the Modbus network, the protocol allows each controller to know its own device address, and identify the data addressing to it, then decide the effective type and take out data contained in the information. The controller can also organize the response information and use the Modbus protocol to send out this information.

The controller takes use of master-subordinate technique. That is to say, only one device (the master) can initiate transmission (the query), and other devices (subordinate ones) respond according to data queried by the master device. A typical master device: host and programmable instrument. Typical subordinate device: programmable controllers.

Master device can communicate with one subordinate device, but also communicate with all subordinate devices by broadcast. For the former situation, the device will respond one message; for the latter, the device will not respond. Modbus protocol establishes a master query format: device (or broadcast) address, function code, all the data to be sent, and an error detection field.

Response message of subordinate device is also constituted by the Modbus protocol, including the domain to be act, any data to be returned, and an error detection field. If a error occurred during message reception, or the subordinate device can not perform its command, the subordinate device will create an error message and sent it out as a response

Modbus has two means of transmission: ASCII and RTU

Communication in ASCII mode, each 8 bit byte in a message is sent as two ASCII character. It uses the LRC error detection, the advantage of which is that the interval of sending the character is only 1s without error; while in RTU mode, every 8 bit byte contains two 4 bit of hexadecimal characters. It uses the CRC error detection, and the advantage is at the same baud rate, it can transmit more data than ASCII. About ASCII mode and RTU data format, more information can be found in the relevant Modbus communication protocol:

	ASCII mode	RTU mode
Code system	<ul style="list-style-type: none"> <li>▪ hexadecimal, ASCII character 0~9, A~F</li> <li>▪ every ASCII byte consists of one hexadecimal character</li> </ul>	<ul style="list-style-type: none"> <li>▪ 8 bit binary system, hexadecimal 0~9, A~F</li> <li>▪ Every 8 bit consists of one or two hexadecimal character</li> </ul>
Data bit	<ul style="list-style-type: none"> <li>▪ 1 start bit</li> <li>▪ 7 data bits, the smallest significance bit significance bit sent firstly</li> <li>▪ 1 parity check bit, none if there is no calibration</li> <li>▪ 1 stop bit (with calibration) , 2 Bits (without calibration)</li> </ul>	<ul style="list-style-type: none"> <li>▪ 1 start bit</li> <li>▪ 8 data bits, the smallest significance bit significance bit sent firstly</li> <li>▪ 1 parity check bit, none if there is no calibration</li> <li>▪ 1 stop bit (with calibration) , 2 Bits (without calibration)</li> </ul>
Error detection	LRC (Longitudinal rigmarole check)	<ul style="list-style-type: none"> <li>▪ CRC (circulate rigmarole check)</li> </ul>

**1、 set Modbus RTU mode**

The device use ASCII command set as default protocol. To use MODBUS RTU, follow these steps to set:

1. Short circuit CONFIG (the 3<sup>rd</sup> pin) and GND (the 4<sup>th</sup> pin)
2. Correctly link Power and other interface cables
3. Switch on power supply, module is in default state automatically, communication address is **00**, baud rate is **9600**.
4. The module is to finish initialization, about 5 minutes
5. Send command **\$00P1 (cr)**, check response, if it is **!00(cr)**, set correctly.
6. Switch off power supply, cut off the connecting between CONFIG and GND
7. Module has been set to MODBUS RTU protocol

**2、 Data format under MODBUS RTU**

Host query message specifies the start address of the register to be read and the quantity of registers. The start address of the register addressing is 0000. Checking format is as following:

Slave Address	Function	Starting Address Hi	Starting Address Lo	No. of Points Hi	No. of Points Lo	Error Check(CRC)
01	03	00	00	00	08	44 0C

Host send query information Tx:    01 03 00 00 00 08 44 0C  
 01 device address  
 03 function code  
 00 00 the start address of register addressing  
 00 08 quantity of register  
 44 0C CRC check code

Subordinate device response

Register data in the response information of subordinate device corresponds to the two bytes of each register; the first data byte is high, the second is the low data. Response format is as following:

Slave Address	Function	Byte Count	Data Hi(Register 40001)	Data Lo(Register 40001)	Data Hi(Register 40002)	Data Lo(Register 40002)	.....	Data Lo(Register 40008)	Error Check(9E)
01	03	10	19	99	00	00	.....	02	9E

Subordinate device response information Rx: 01 03 10 19 99 00 00 00 00 00 00 00

- 01 device address
- 03 function code
- 10 the quantity of bytes
- 19 99 the first register data
- 00 00 the second register data
- 
- 
- 
- 87 69 CRC check code

The below Figure is the register information checked by MODSCAN software and its instruction

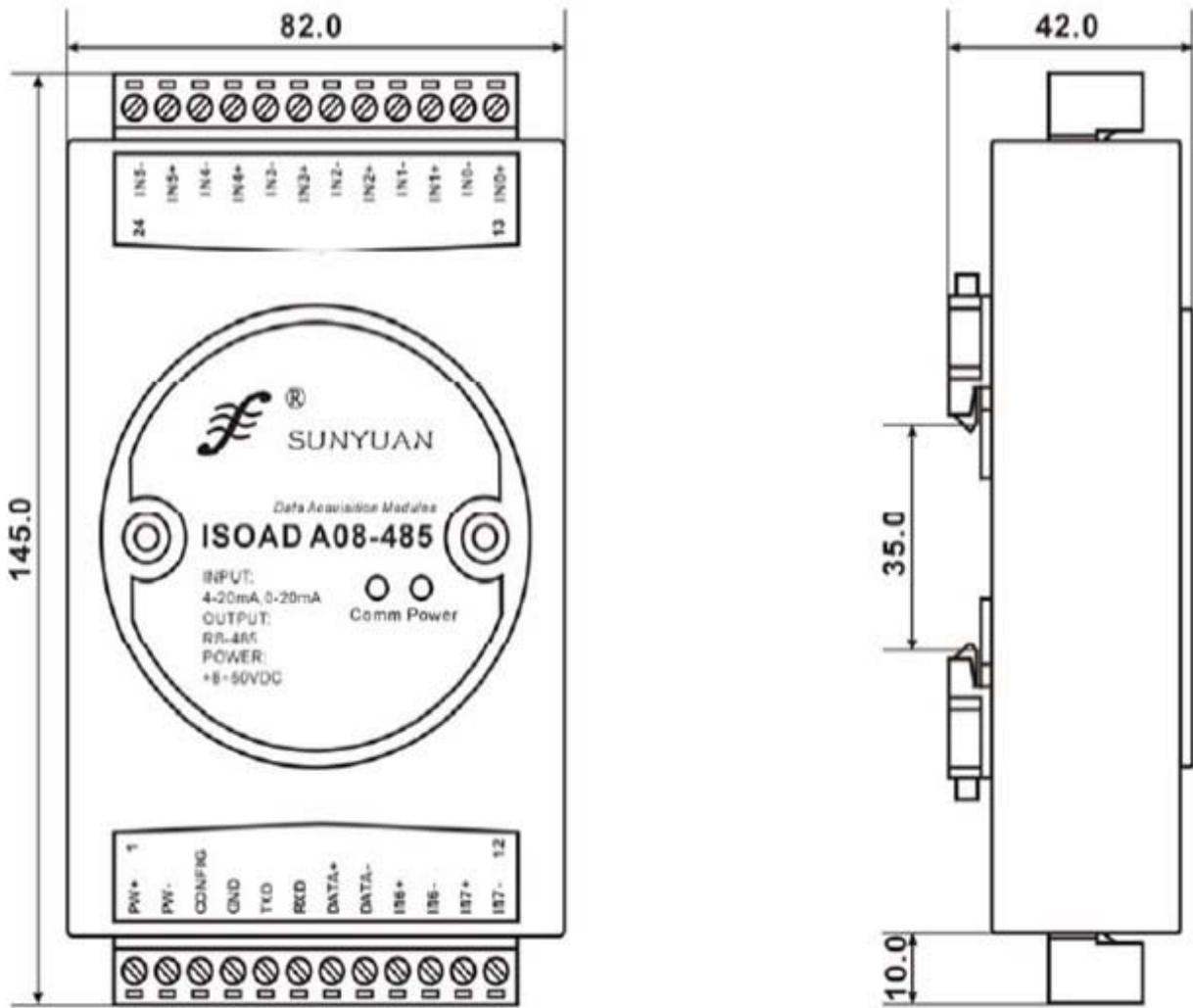
<p>The screenshot shows the ModScan software interface. It includes fields for Address (0001), Device Id (1), Length (2), and MODBUS Point Type (03: HOLDING REGISTER). It also displays the Number of Polls (39) and Valid Slave Responses (39). At the bottom, it shows register data: 40001: &lt;07FFH&gt; and 40002: &lt;0333H&gt;.</p>	<p>Instructions:</p> <ul style="list-style-type: none"> <li>Address is the start address of the register</li> <li>Device ID is module address</li> <li>Length is the number of registers been read</li> <li>MODBUS Point Type :Modbus function type selection</li> <li>40001: corresponds to register data, analog input value of channel 0</li> <li>40002: corresponds to register data, analog input value of channel 1</li> </ul>
---	--

4、Instructions of register:

Address	Data	R/W	Description
4X			
40001	IN0	Read only	Measured Value of Channel 0
40002	IN1	Read only	Measured Value of Channel 1
40003	IN2	Read only	Measured Value of Channel 2
40004	IN3	Read only	Measured Value of Channel 3
40005	IN4	Read only	Measured Value of Channel 4
40006	IN5	Read only	Measured Value of Channel 5
40007	IN6	Read only	Measured Value of Channel 6
40008	IN7	Read only	Measured Value of Channel 7
40211	Module Name	Read only	Higher bits: 0x01 Lower bits: 0x08
40221	Channel Status	Read/Write	Higher bits: 0x00 Lower bits: Channel Status (0xFF)

VIII. Dimension

(unit: mm)



AD-A08 outline dimensional drawing