

## Analog signal isolated acquisition A/D Converter

### ISOAD 16

**Features:**

- Low cost, small size
- Analog signal acquisition, isolated conversion RS-485/232 output
- measuring accuracy > 0.05%, nonlinearity <0.05%
- RS-485/232 output, can be remotely reconfigured for modules accuracy
- Input signal/Output signal 3000VDC
- Operation Power Supply: +24VDC±10%
- High reliability, convenient programming, easy installation and wiring
- User can set module address and baud rate, etc.
- Support Modbus RTU communication protocol

**Applications:**

- Signal testing/ monitoring /control
- Intelligent building control, security and engineering applications
- RS-232/485 industrial automation control system
- Industrial fields signal isolation and long-terms transmission
- Debugging and controlling of equipment operation
- Debugging and controlling of equipment operation
- Acquisition and record of industrial field data
- Development of medical and industrial products
- Current/voltage 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 / 0-5V 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 ASCII 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.

ISOAD 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.

ISOAD 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.

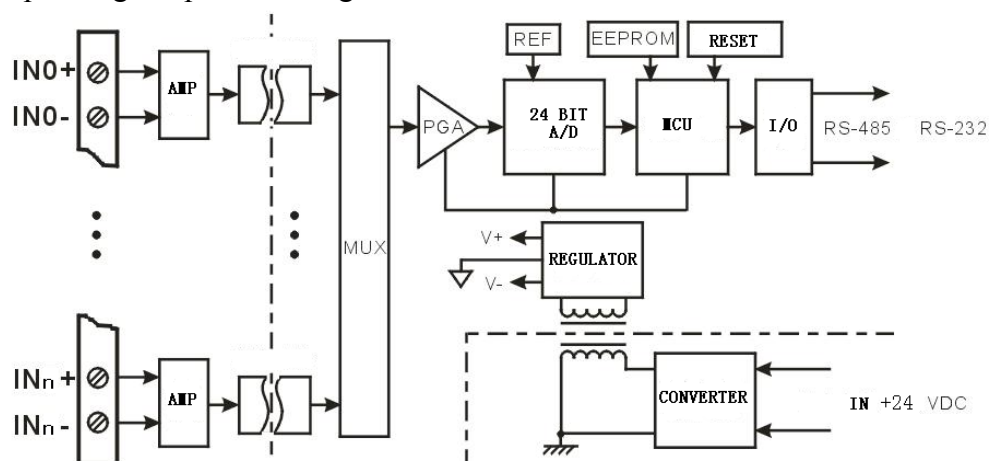


Figure 1 ISOAD A16 functional block diagram

**Function of ISOAD 16:**

ISOAD 16 signal isolation acquisition module could be used to collect 16-channel isolated/un-isolated current or voltage signal.

**1、 Analog signal input**

24-bit output accuracy. The product has been calibrated before delivery. It is also convenient to be programmed and calibrated by the user during the usage.

The specific current or voltage output range need to be corresponding to the selection of product. The two outputs selection should be the same when outputting two signals.

**2、 protocol**

Communication Interface: 1 standard RS-485 interface or 1 standard RS-232 interface. Please note before ordering.

Communication protocol: support two types of protocols, one is standard Modbus RTU communication protocol and the other is character communication protocol. User could decide which protocol to use by programming. The product could have network communication with supervisory computer control system or various types of brands such as PLC, RTU.

Data format: 10 bit. One start bit, 8 data bit, one stop bit.

Communication address (00H~FFH) and baud rate (300、600、1200、2400、4800、9600、19200、38400、57600、115200bps) could be set; The network communication distance is up to 1200 meters, connecting by double ground shield cable.

High anti-jamming of the communication interface,  $\pm 15\text{KV}$  ESD protection, and the response time is less than 100mS.

**3、 Anti-jamming**

Set checksum according to requirements. TVS inside the module can actively restrain all kinds of surge impulse and protect the module. Digital filter inside could also restrain the power frequency jamming from power grid.

**Model selection**

<b>ISOAD16</b>		U(A)□	-	□	
	Vin/In				communication interface
<b>U1:</b> 0-5V		<b>A1:</b> 0-1mA			<b>485:</b> Output is RS-485 interface
<b>U2:</b> 0-10V		<b>A2:</b> 0-10mA			<b>232:</b> Output is RS-232 interface
<b>U3:</b> 0-75mV		<b>A3:</b> 0-20mA			
<b>U4:</b> 0-2.5V		<b>A4:</b> 4-20mA			
<b>U5:</b> 0- $\pm 5\text{V}$		<b>A5:</b> 0- $\pm 1\text{mA}$			
<b>U6:</b> 0- $\pm 10\text{V}$		<b>A6:</b> 0- $\pm 10\text{mA}$			
<b>U7:</b> 0- $\pm 100\text{mV}$		<b>A7:</b> 0- $\pm 20\text{mA}$			
<b>U8:</b> Customized		<b>A8:</b> Customized			

Note: User could choose RS-485 or RS-232 output by the interface wiring jumper at the back of the product.

- 1: **ISOAD16 A4-485** Means 4-20ma signal output, output is RS485 interface  
 2: **ISOAD16 A7-232** Means 0-5V signal output, output is RS232 interface

**ISOAD 16 general parameters:**

 (typical @ +25°C,  $V_s$ 为24VDC)

Input: Voltage / Current

Accuracy: @ +25°C as table 1

Nonlinearity : @ +25°C as table 1

 Input imbalance:  $\pm 0.1 \mu\text{A}/^\circ\text{C}$ 

 Temperature Drift:  $\pm 15 \text{ ppm}/^\circ\text{C}$  ( $\pm 30 \text{ ppm}/^\circ\text{C}$ , max)

Input load: 50Ω (4-20mA/0-20mA/0-±20mA current input)

100Ω (0-10mA/0-±10mA current input)

1KΩ (0-1mA/0-±1mA current input)

&gt;1MΩ (voltage input)

Band width: -3 dB 10 Hz

Switching rate: 2.5-30 KHz Sps

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

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

Input terminal protection: over-voltage/over current protection

Communications Protocol: RS-485 or RS-232 ASCII character protocol and standard MODBUS RTU communication protocol

Baud rate: 300、600、1200、2400、4800、9600、19200、38400bps、57600、115200bps , selectable

Address: 0~255 , selectable

Response Time: 100 ms max

Operation Power Supply: +24VDC±10%, internal anti-reverse &amp; over-voltage protection

watt consumption: &lt;8W

Operation Temperature: - 45 ~ +80°C

Operation Humidity: 10 ~ 90% (non-condensing)

Storage Temperature: - 45 ~ +80°C

Storage Humidity: 10 ~ 95% (non-condensing)

 Isolation withstand voltage: 3KVDC between input and output, 1 minute, stray current 1mA  
 Output and power are common-ground.

Endure Impact Voltage: 3KVAC, 1.2/50us(peak value)

Appearance &amp; dimension: 158\*87.5\*59mm

**ISOAD16 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	$\pm 0.02 \%$ FS	$\pm 0.05 \%$ FS	$\pm 0.05 \%$ FS	$\pm 0.01 \%$ FS

**Table 1** ISOAD16 measuring accuracy and Non linearity limit

**Product picture:**

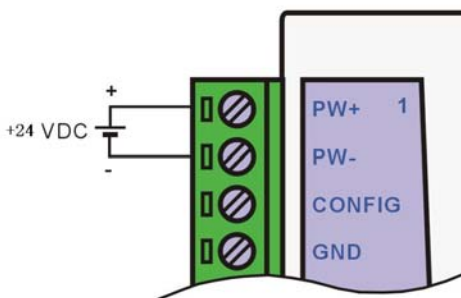

**Pin description:**

Pin	Name	Description	Pin	Name	Description
1	PW+	Power supply +	21	IN4+	Input channel 4+
2	PW-	Power supply -	22	IN4-	Input channel 4-
3	CONFIG	Config initiation	23	IN5+	Input channel 5+
4	GND	ground	24	IN5-	Input channel 5-
5	TXD	RS-232 transmitting	25	IN8+	Input channel 8+
6	RXD	RS-232 receiving	26	IN8-	Input channel 8-
7	DATA+	RS-485 signal +	27	IN9+	Input channel 9+
8	DATA-	RS-485 signal -	28	IN9-	Input channel 9-
9	IN6+	Input channel 6 +	29	IN10+	Input channel 10+
10	IN6-	Input channel 6-	30	IN10-	Input channel 10-
11	IN7+	Input channel 7+	31	IN11+	Input channel 11+
12	IN7-	Input channel 7-	32	IN11-	Input channel 11-
13	IN0+	Input channel 0+	33	IN12+	Input channel 12+
14	IN0-	Input channel 0-	34	IN12-	Input channel 12-
15	IN1+	Input channel 1+	35	IN13+	Input channel 13+
16	IN1-	Input channel 1-	36	IN13-	Input channel 13-
17	IN2+	Input channel 2+	37	IN14+	Input channel 14+
18	IN2-	Input channel 2-	38	IN14-	Input channel 14-
19	IN3+	Input channel 3+	39	IN15+	Input channel 15+
20	IN3-	Input channel 3-	40	IN15-	Input channel 15-

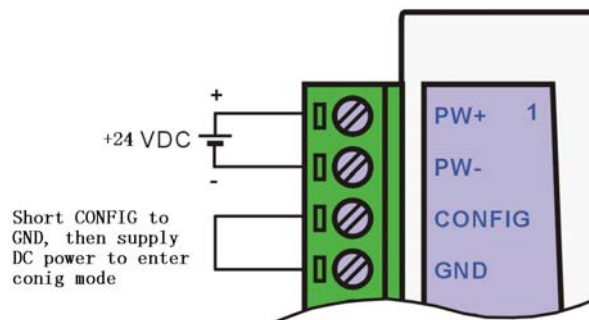
**Table 2** pin description

**Note: If N (channel number) of the selected product is less than 16, the usable channel connector is channel 0 ~ channel (N-1); there is no electric connection from channel N to 16 connector to the product interior, and they are invalid channels.**

**Wiring scheme:**



**Figure 2** ISOAD16 power supply wiring



**Figure 3** ISOAD16 default status wiring

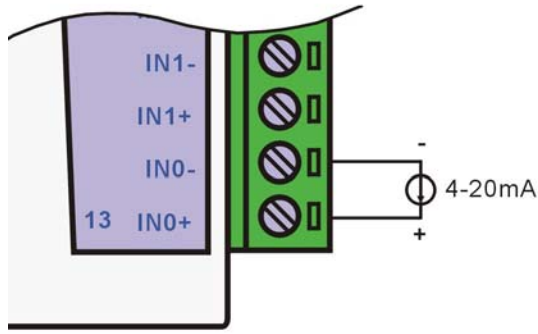


Figure 4 ISOAD16 signal input wiring

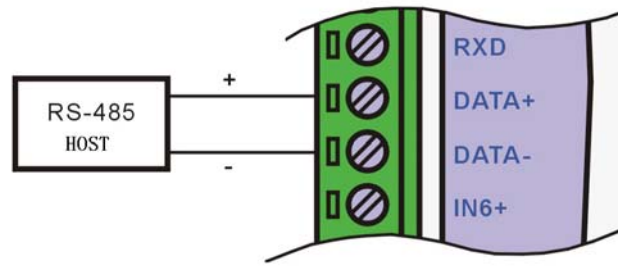


Figure 5 ISOAD16 RS-485 interface wiring

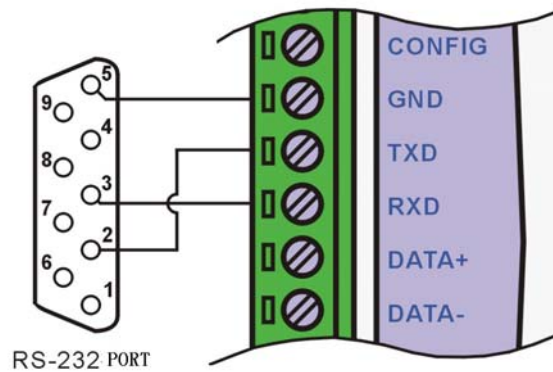


Figure 6 ISOAD16 RS-232 interface wiring

**Initialization of ISOAD16 module:**

All ISOAD16 modules must be assigned a unique address code (between 00H ~ FFH) when joining the network. However, all the new ISOAD16 modules use the same factory default setting as following:  
 Address code of 01H  
 baud rate is 9600 bps  
 Checksum prohibited

As the new module have the same default setting, their address will conflict with each other, so when you set up the system, you must re-configure each analog output module address. Users can modify ISOAD16 module's address through configuration commands after connecting ISOAD16 module power cable and RS485 communication lines. Baud rate, parity, status, and communication protocols can also be re-configure according to user's requirements. Before the adjusting process, one must set the module to the CONFIG mode.

To set the modules to CONFIG mode:

ISOAD16 module has a Pin marked CONFIG. Connect CONFIG Pin to the ground (GND) as Figure 3, and then connect power supply.

Under CONFIG mode, the module setting is as following:

- Address code: 00H
- Baud rate: 9600 bps
- Checksum: disable

At this time, users can modify ISOAD16 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 CONFIG mode 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.

### ISOAD16 ASCII character command set:

The command consists of a series of ASCII characters, such as the leading code, address ID, variables, and optional checksum byte and a end code (cr).

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

(Leading code)	First letter of the command. such as %, \$, #, @, etc.	1 char
(Addr)	Address of the module. Range from 00 to FF (hexadecimal).	2 char
(Command)	command key word.	1 char
[Data]	command parameter	variable
[Checksum]	Checksum of the Command string, when checksum is used.	2 char
(cr)	End code, carriage return, is 0x0D.	1 char

[Checksum] is needed when enabling checksum. It takes up 2 characters. Commands and responses must have checksum feature. Checksum is used to check all input commands to help user finding the command error from the host 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     **\$022(cr)**  
 Module response **!02000600 (cr)**

Enable checksum

User command     **\$002B6 (cr)**  
 Module response **!00020600 A9 (cr)**

'\$' = 0x24    '0' = 0x30    '2' = 0x32

B6=(0x24+0x30+0x30+0x32) AND 0xFF

'!' = 0x21    '0' = 0x30    '2' = 0x32    '6' = 0x36

A9=(0x21+0x30+0x30+0x30+0x32+0x30+0x36+0x30+0x30) AND 0xFF

### Common commands:

- 1、 Read analog input module data
- 2、 set the analog output value of channel N
- 3、 module allocation
- 4、 Read allocation status
- 5、 Offset calibration
- 6、 Full scale calibration
- 7、 Linearity standardization
- 8、 Enable or prohibit channel command
- 9、 read channel status command
- 10、 Set communication mode

### 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、 Ensure the address and command are correct. Some general error derives from wrong**

**address or command.**

**2、 the command must be entered in capital letters.**

**3、 If there is a address error or communication error, the module does not respond.**

### 1、 Read all channel analog input data

Description : Based on the current data format, the command is to read analog input value of all channels from the module.

Syntax : **#AA(cr)**

Parameter Description : # delimiter character

**AA** Module address, value range 00H~FFH(hexadecimal)

**(cr)** end character , carriage return (0DH)

Response: **>(data)(cr)** Response

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

Parameter Description : > **delimiter character**

**(data)** Represent the data of all channels which been read by the command. The data format is in Engineering Units, percentage of FSR, complement of hexadecimal or ohms. Refer to Chapter 3 Command Set.

**(cr)** end character , carriage return (0DH)

Note: If a channel has been shut down, then the data read out as '0' character.

E.g. user command **#23(cr)**

Module response **>+04.765+04.756+04.632+04.000+05.001+06.000+.....+16.000(cr)**

Description : on module of 23H, input is (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 15: +16.000mA

### 2、 Read the analog input data in Channel N

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

Syntax : **#AANN(cr)**

Parameter Description : # delimiter character

**AA** Module address, value range 00H~FFH

**NN** channel number, range: 00~15(decimal), invalid when more than 15.

**(cr)** end character , carriage return (0DH)。

Response: **>(data)(cr)** command is valid.

**?AA(cr)** invalid command or illegal operation or channel is closed.

Parameter Description : > delimiter character

**(data)** data read in channel N. The data format is in Engineering Units, PERCENT OF FSR, complement of hexadecimal. Or ohms. Refer to Chapter 3 Command Set.

**(cr)** end character , carriage return (0DH)。

Note: If a channel has been shut down, then the data read out as '0' character.

E.g. user command **#230(cr)**

Module response **>+04.632 (cr)**

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

### 3、 module configuration

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

Syntax: **%AANNTTCCFF(cr)**

Parameter Description : % delimiter character

**AA** Module address, value range 00H~FFH

**NN** hexadecimal address, NN is from 00 to FF

**TT** hexadecimal represents type code. ISOAD product should be set as 00

**CC** hexadecimal represents baud rate code.

baud rate code	baud rate
01	300 baud
02	600 baud
03	1200 baud
04	2400 baud
05	4800 baud
06	9600 baud
07	19200 baud
08	38400 baud
09	57600 baud
0A	115200 baud

**Table 3 Baud rate**

**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:** Reserved bits should be set to 0

**Bit1-bit0:** data format bit

00: Engineering Units

01: % of FSR

10: complement of hexadecimal (Twos complement)

**(cr)** end character , carriage return (0DH)

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

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

Parameter Description : **!** Delimiter code indicates valid command.

**?** delimiter code indicates invalid command.

**AA** indicates input Module address

**(cr)** end character , carriage return (0DH)

Note: User should install jumper to set the module in CONFIG mode and the default address AA=00.

E.g. user command **%0011000600(cr)**

Module response **!11(cr)**

Description : % delimiter character

**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, ISO AD16 series should be set to 00。
- 06** baud rate is 9600 baud。
- 00** Indicate data format is engineering unit and the checksum is forbidden.

#### 4、Set communication protocol command

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

Syntax : **\$AAPV(cr)**

Parameter Description : **\$ delimiter character**

- AA** Module address, value range 00H~FFH
- P** set communication protocol
- V** the code of the protocol could be 0 or 1.
  - 0:** ASCII character communication protocol
  - 1:** Modbus RTU protocol
- (cr)** end character , carriage return (0DH)

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

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

Parameter Description : **! delimiter character** :command is valid.

**? delimiter character:** command is invalid.

**AA** input Module address。

**(cr)** end character , carriage return (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 command **\$00P1(cr)**  
 Module response **!00 (cr)**

Description : set communication protocol as Modbus RTU protocol

e.g.2: user command **\$00P0(cr)**  
 Module response **!00 (cr)**

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

#### 5、read the configuration status

Description : read the allocation of the specified module

Syntax : **\$AA2(cr)**

Parameter Description : **\$ delimiter character**

- AA** Module address, value range 00H~FFH
- 2** read the allocation status
- (cr)** end character , carriage return (0DH)

Response: **!AATTCCFF(cr)** command is valid.

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

Parameter Description : **! delimiter character**

- 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)

Note:

E.g. user command **\$302(cr)**

Module response     **!300F0600(cr)**  
 Description : !     delimiter character  
               **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

## 6、Offset calibration

Description : Adjust the offset of channel N

Syntax : **\$AA0NN(cr)**

Parameter Description : \$     delimiter character  
                           **AA**   Module address, value range 00H~FFH  
                           **0**     Offset calibration  
                           **NN**   channel 0~15  
                           **(cr)**   end character , carriage return (0DH)。

Response: **!AA(cr)**     command is valid.  
            **?AA(cr)**     invalid command or illegal operation

Response: **!AA(cr)**     command is valid.  
            **?AA(cr)**     invalid command or illegal operation

Parameter Description : !     delimiter code which means valid command  
                           **?**     delimiter code means invalid command  
                           **AA**   input Module address  
                           **(cr)**   end character , carriage return (0DH)

Note: 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.

E.g. user command     **\$23000(cr)**  
               Module response     **!23(cr)**

Description : offset calibrating channel 0 of address 23H

## 7、Gain calibration command

Description : calibrate the gain of channel N

Syntax : **\$AA1NN(cr)**

Parameter Description : \$     delimiter character  
                           **AA**   Module address, value range 00~FF  
                           **0**     Gain calibration command  
                           **NN**   channel 0~15  
                           **(cr)**   end character , carriage return (0DH)

Response: **!AA(cr)**     command is valid.  
            **?AA(cr)**     invalid command or illegal operation

Parameter Description : !     delimiter code means valid command  
                           **?**     delimiter code means invalid command  
                           **AA**   Input Module address  
                           **(cr)**   end character , carriage return (0DH)。

Note: 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.

E.g. user command     **\$23100(cr)**  
           Module response     **!23(cr)**  
 Description : gain calibrating channel 0 of address 23H

## 8、Linearity standardization

Description: set Linearity standardization to an analog input channel

Syntax : **@AANNBB(cr)**

Parameter Description: @     **delimiter character**

**AA**   Module address, value range 00~15(decimal system)

**NN**   channel number

**BB**   input current of standardization status

**(cr)**   end character , carriage return (0DH)。

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

**!AANN@Pn ok(cr)**   command is valid.

Parameter Description : !     delimiter character :command is valid.

?     **delimiter character** :command is invalid.

**AA**   input Module address

**NN**   channel number

**@Pn ok Pn=P0-P8** , means the nine standard poin. Ok means the point is set as standard,

END means the standardization is finished.

**(cr)**   end character , carriage return (0DH)。

Standardization period and its response:

1. standard point is 0, adjust the input current into 0mA, input command is **@AANN00(cr)** , module response is **\$AANN@P0 ok**
2. standard point is 1, adjust the input current into 1mA, input command is **@AANN01(cr)** , module response is **\$AANN@P1 ok**
3. standard point is 2, adjust the input current into 2mA, input command is **@AANN02(cr)** , module response is **\$AANN@P2 ok**
4. standard point is 3, adjust the input current into 3mA, input command is **@AANN03(cr)** , module response is **\$AANN@P3 ok**
5. ....
6. standard point is 7, adjust the input current into 15mA, input command is **@AANN15(cr)** , module response is **\$AANN@P7 ok**
7. standard point is 8, adjust the input current into 20mA, input command is **@AANN20(cr)** , module response is **\$AANN@P8 ok END**

## 9、Enable or prohibit command channel

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

Syntax: **\$AA5VVVV(cr)**

Parameter Description: \$     **delimiter character**

**AA**   Module address, value range 00~FF(hexadecimal)

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

**VVVV** 16 bit (hexadecimal)

3~0 at the first 4 bit represents 15~12channel.

3~0 at the 2<sup>nd</sup> 4 bit represents 11~8 channel.

3~0 at the 3<sup>rd</sup> 4 bit represents 7~4 channel.

3~0 at the 4<sup>th</sup> 4 bit represents 3~0 channel.

Bit is 0: prohibit the channel

Bit is 1: enable the channel

**(cr)** end character , carriage return (0DH)。

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

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

Parameter Description : **!** delimiter character :command is valid.

**?** delimiter character: command is invalid.

**AA** input Module address。

**(cr)** end character , carriage return (0DH)

E.g. user command **\$0853748(cr)**

Module response **!08 (cr)**

Description : set channel value to 0x3748。

3 is 0011, represents enable channel 13 and 12, prohibit channel 15 and 14。

7 is 0111, represents enable channel 10、9 and 8, prohibit channel 11。

4 is 0100, represents enable channel 6, prohibit channel 7、5 and 4。

8 is 1000, represents enable channel 3, prohibit channel 2、1 and 0。

## 10、Read the channel status

Description : Read the channel status

Syntax : **\$AA6(cr)**

Parameter Description : **\$** delimiter character

**AA** Module address, value range 00H~FFH。

**6** Read the channel status

**(cr)** end character , carriage return (0DH)

Response: **!AAVVVV(cr)** command is valid.

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

Parameter Description : **!** delimiter character :command is valid.

**?** delimiter character: command is invalid.

**AA** input Module address

**VVVV** 16bit (hexadecimal)

**(cr)** end character , carriage return (0DH)。

E.g. user command **\$186 (cr)**

Module response **!18FFFF (cr)**

Description : the current channel status is 0xFF。

0xFFFF is 1111 1111 1111 1111, means all the channels have been enabled.

## 11、Read module name

Description : Read module name

Syntax : **\$AAM(cr)**

Parameter Description : **\$** delimiter character

**AA** Module address, value range 00~FF

**M** Read module name

**(cr)** end character , carriage return (0DH)

Response: **!AA(ModuleName)(cr)** command is valid.

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

Parameter Description : **!** delimiter character :command is valid.

**?** **delimiter character** :command is invalid.

**AA** input Module address.

**(ModuleName)** module name could be ISOAD16、ISO 4021 or ISO4014 etc. Means the module type you used.

**(cr)** end character , carriage return (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 **\$08M(cr)**

Module response **!08ISO AD16 (cr)**

Description : Module ISO AD16 is in address 08H module.

**Input range and data format**

There are 3 output data format which are been supported under ASCII character command mode

- 00: (Engineering Units)
- 01: (% of FSR)
- 10: complement hexadecimal

Input range	Data format	Span +	Zero	Span -	Res.
<b>A1: 0-1mA</b> <b>A5: 0-±1mA</b>	Engineering Unit	+1.0000	±0.0000	-1.0000	0.1uA
	% of FSR	+100.00	±000.00	-100.00	0.01%
	complement of hex	7FFFFFFF	000000	800000	1LSB
<b>A2: 0-10mA</b> <b>A6: 0-±10mA</b>	Engineering Unit	+10.000	±00.000	-10.000	1uA
	% of FSR	+100.00	±000.00	-100.00	0.01%
	complement of hex	7FFFFFFF	000000	800000	1LSB
<b>A3: 0-20mA</b> <b>A4: 4-20mA</b> <b>A7: 0-±20mA</b>	Engineering Unit	+20.000	±00.000	-20.000	1uA
	% of FSR	+100.00	±000.00	-100.00	0.01%
	complement of hex	7FFFFFFF	000000	800000	1LSB
<b>U1: 0-5V</b> <b>U5: 0-±5V</b>	Engineering Unit	+5.0000	±0.0000	-5.0000	100uV
	% of FSR	+100.00	±000.00	-100.00	0.01%
	complement of hex	7FFFFFFF	000000	800000	1LSB
<b>U2: 0-10V</b> <b>U6: 0-±10V</b>	Engineering Unit	+10.000	±00.000	-10.000	1mV
	% of FSR	+100.00	±000.00	-100.00	0.01%
	complement of hex	7FFFFFFF	000000	800000	1LSB
<b>U3: 0-75mV</b>	Engineering Unit	+75.000	±00.000	-75.000	1uV
	% of FSR	+100.00	±000.00	-100.00	0.01%
	complement of hex	7FFFFFFF	000000	800000	1LSB
<b>U4: 0-2.5V</b>	Engineering Unit	+2.5000	±0.0000	-2.5000	100uV
	% of FSR	+100.00	±000.00	-100.00	0.01%
	complement of hex	7FFFFFFF	000000	800000	1LSB
<b>U7: 0-100mV</b>	Engineering Unit	+100.00	±000.00	-100.00	10uV
	% of FSR	+100.00	±000.00	-100.00	0.01%
	complement of hex	7FFFFFFF	000000	800000	1LSB
<b>A8 : customized</b> <b>U8 : customized</b>	Engineering Unit	+100.00	±000.00	-100.00	0.01%
	% of FSR	+100.00	±000.00	-100.00	0.01%
	complement of hexadecimal	7FFFFFFF	000000	800000	1LSB

Table 5 input range and data format

E.g.

1、input range is A4: 4~20mA, input is 4 mA:

```

user command  #010(cr)
Engineering Unit response is >+04.000(cr)
% of FSR      response is >+020.00(cr)
    
```

complement of hexadecimal response is **>199999(cr)**

2、input range is U1: 0~5V, input is 3V:

	User command	<b>#010(cr)</b>
Engineering Unit	response is	<b>&gt;+3.0000(cr)</b>
% of FSR	response is	<b>&gt;+060.00(cr)</b>
complement of hexadecimal	response is	<b>&gt;4CCCCC(cr)</b>

### Modbus RTU protocol:

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 **%0001001600 (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
8. There are only 1 output data format which are been supported under MODBUS RTU ASCII mode, the 2 bytes complement hexadecimal.

### Instructions of register of RTU mode:

Address 4X	Data	R/W	Description
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
400xx	...	Read only	Measured Value of Channel XX
40015	IN14	Read only	Measured Value of Channel 14
40016	IN15	Read only	Measured Value of Channel 15
40211	Module Name	Read only	Higher bits: 0xAD Lower bits: 0x16
40221	Channel Status	Read/Write	Higher bits: Channel Status (0xFF) Lower bits: Channel Status (0xFF)

Table 6 Modbus Rtu register Instructions

**calibration and linearity standardization:**

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.

And it could be linearity standardized to improve full span measuring accuracy. Standard point is 9, from P0 to P8.

In order to improve calibration and standardization 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 0 to be calibrated and connect to the corresponding input signal according to the input range of the module

Zero is calibrated when the input is 0; full scan is calibrated when input 100% of FSR. For example, when the input is 4-20mA, zero calibration: 0mA input; full scan calibration: 24mA. When the input is 0-5V, zero calibration: 0V input; full scan calibration: 5V.

2. Input zero signal to the channel, usually it is 0mA or 0V.
3. When the signal is stable, send offset calibration \$ AA1N command (N represents the channel code currently being calibrated, 0 or 1) to the analog input module.
4. Input current or voltage signal of 100% of FSR to the channel which needs calibration.
5. When the signal is stable, send gain calibration \$ AA0 command to the analog input module.
6. Calibration is finished.

Standardization process and response is as following:

1. standard point is 0, adjust the input current into 0mA, input command is @AANN00(cr) , module response is \$AANN@P0 ok
2. standard point is 1, adjust the input current into 1mA, input command is @AANN01(cr) , module response is \$AANN@P1 ok
3. standard point is 2, adjust the input current into 2mA, input command is @AANN02(cr) , module response is \$AANN@P2 ok
4. standard point is 3, adjust the input current into 3mA, input command is @AANN03(cr) , module response is \$AANN@P3 ok
5. ....
6. standard point is 7, adjust the input current into 15mA, input command is @AANN15(cr) , module response is \$AANN@P7 ok
7. standard point is 8, adjust the input current into 20mA, input command is @AANN20(cr) , module response is \$AANN@P8 ok END

**Dimension:** (unit: mm)

158\*87.5\*59



