

2-Channel/4-channel Analog Signal Isolated Data Acquisition A/D Converter SYAD02/04A Series

Features:

- Low cost, small size
- Analog signal acquisition, isolated conversion RS-485/232 output
- Analog signal output accuracy > 0.05%
- RS-485/232 output, can be remotely reconfigured for modules accuracy
- Input signal/Output signal isolation: 3000VDC
- Wide power supply range : 8 ~ 50VDC
- High reliability, convenient programming, easy installation and wiring
- User can set module address and baud rate, etc. by programming
- 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
- Sensors signal test
- Analogue signal A/D conversion. Data Acquisition and record of industrial field data
- Development of medical and industrial products

Product Overview:

SYAD02/04A A-D Converters achieve the signal acquisition between sensor and the host to test analog signal or control remote device. The configuration software can be used for a variety of sensors including analog input, analog output, and digital signal input / output (I / O), SYAD02/04A series of products can be used in industrial automation RS-232/485 bus system, 4-20mA / 0-10V 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. Each serial port may be associated with up to 256 PCS SYAD02/04A series modules. The communication methods use 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 computer programming. SY series products are based on single chip of intelligent monitoring and control systems, all the user to set the calibration value, address, baud rate, data format, checksum status and other configuration information is stored in nonvolatile memory EEPROM inside.

SY series products are designed and manufactured according to industry standard. The signal output and communication interface are isolated, it can withstand 3000VDC isolation voltage and with anti-interference ability, high reliability. Operating temperature range: - 45°C ~ +80°C.

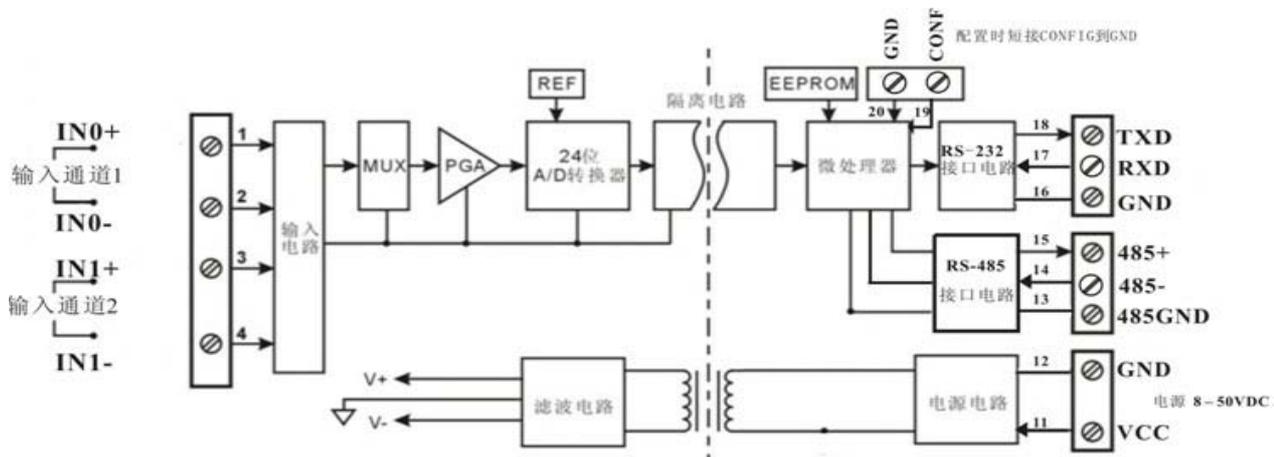


Figure 1 SYAD02A Functional Block Diagram

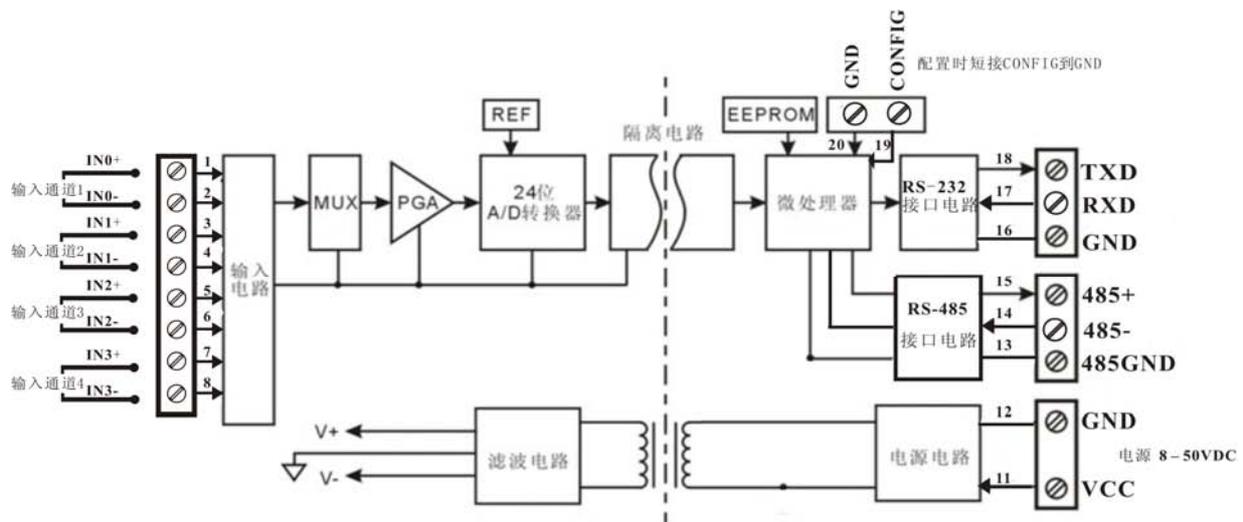


Figure 1 SYAD04A Functional Block Diagram

Functions of SYAD02/04A:

SYAD02/04A signal isolation data acquisition module could be used to test 2 channel voltage/current; it could also be used to test 4-channel current/voltage signal which are common grounded and signal isolated.

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 inputs selection should be the same when selecting two input signals.

2. Communication protocol

Communication Interface: 1 standard RS-485 interface or 1 standard RS-232 interface.

Communication protocol: Support two types of protocols, one is standard Modbus RTU communication protocol and the other is character communication protocol ASCII. 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、38400bps) 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 diode inside the module can actively restrain all kinds of surge impulse and protect the module. The inside Digital filter could also restrain the power frequency jamming from power grid.

Model selection:

SY AD 02(4)A -U(A)□

Channel

02: 2 Channel

04: 4 Channel

input voltage or current signal

U1: 0-5V

U2: 0-10V

U3: 0-75mV

U4: 0-2.5V

U5: 0-±5V

U6: 0-±10V

U7: 0-±100mV

U8: Customized

A1: 0-1mA

A2: 0-10mA

A3: 0-20mA

A4: 4-20mA

A5: 0-±1mA

A6: 0-±10mA

A7: 0-±20mA

A8: Customized

Note: Support both RS232 and 485 interface

Examples of Model Selection:

1. SYAD02A A4

Means 2channel , 4-20ma signal input, output is RS485 or RS232 interface

2. SYAD02A A7

Means 2channel , 0-5V signal input, output is RS485 or RS232 interface

3. SYAD04A U1

Means 4channel , 0-±5V signal input, output is RS485 or RS232 interface

SYAD02/04A General Parameters:

(typical @ +25°C, Vs为24VDC)

Input: Voltage / Current

Accuracy: 0.05%

Input maladjustment: ±0.1 uA/°C

Temperature Drift: ±15 ppm/°C (±30 ppm/°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)

>1MΩ(voltage input)

Band width: -3 dB 10 Hz

Switching rate: 10 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 communication protocol and MODBUS RTU communication protocol

Baud rate (300、600、1200、2400、4800、9600、19200、38400bps) software selectable

address (00H~FFH) software selectable

Response Time: 100 ms max

Operation Power Supply: +8 ~ 50 VDC wide power supply range, internal anti-reverse & over-voltage protection

watt consumption: < 1W

Operation Temperature: - 45 ~ +80°C

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

Storage Temperature: - 55 ~ +85°C

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

Isolation withstand voltage: between input and output: 3KVDC, 1 minute, leakage current 1mA

Output and power are common-ground.

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

Dimensions: 123 mm x 70 mm x 30mm

Weight: approximately 72g



Figure 2 SYAD02/04A External View

Wiring Diagram

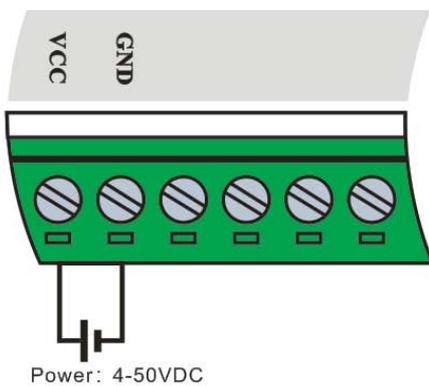


Figure 4 power supply wiring scheme

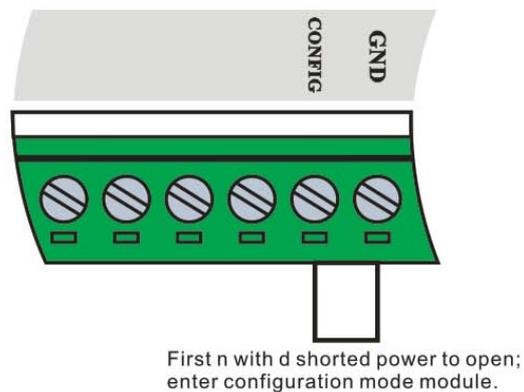


Figure 5 CONFIG wiring scheme

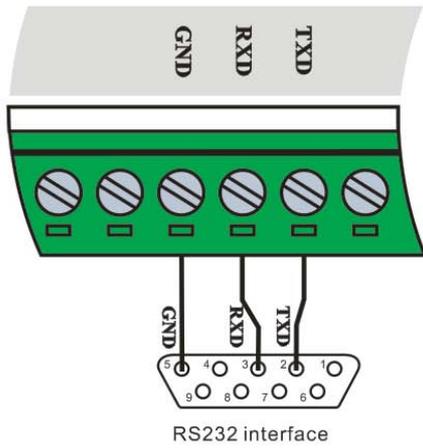


Figure 6 RS-232 wiring scheme

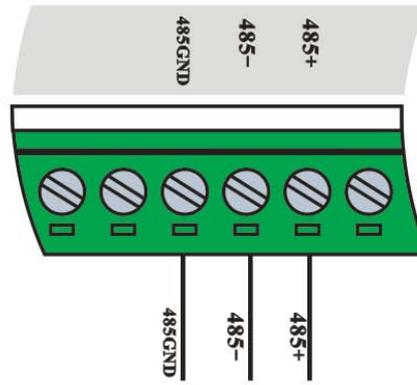


Figure 7 RS-485 wiring scheme

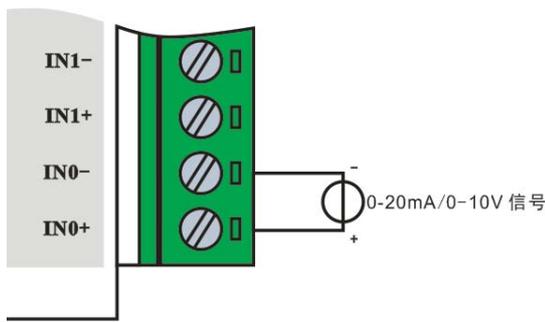


Figure 8 Input wiring scheme

Dimension: Unit:mm

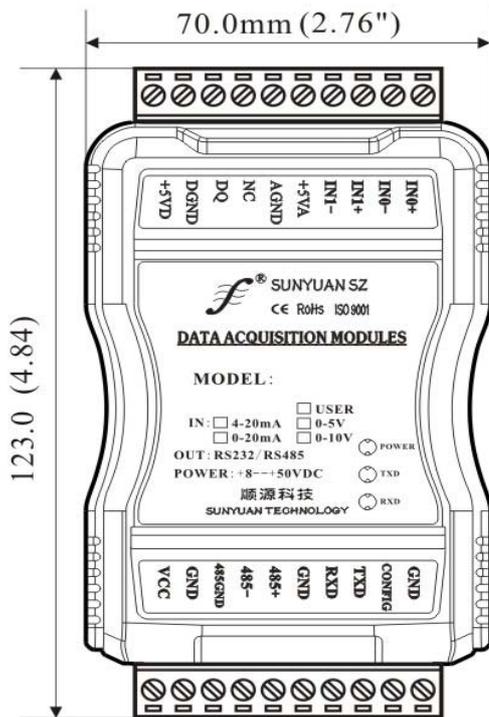


Figure 4 Front View

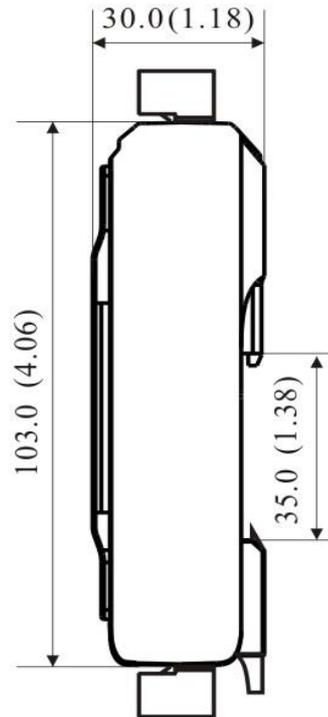


Figure 5 Side View

Pin Description :

Pin	Name	Description	Pin	Name	Description
1	IN0+	Input0 +	11	VCC	Power supply +
2	IN0-	Input0 -	12	GND	Power supply GND
3	IN1+	Input1 +	13	485GND	485GND
4	IN1-	Input1 -	14	485-	RS-485-
5	IN2+	Input2 +	15	485+	RS-485+
6	IN2-	Input2 -	16	GND	232GND
7	IN3+	Input3 +	17	RXD	RS232RXD
8	IN3-	Input3 -	18	TXD	RS232TXD
9	NC	NC	19	CONFIG	Configure initiation
10	NC	NC	20	GND	CONFIG GND

Note: The pin 5,6,7,8 in 2channel product is NC, in 4channel product is signal input terminal.

Initialization of SYAD02/04A Module:

If want to connect RS-485/RS-232 network, it is needed to assign a unique address code (between 00H ~ FFH) for each SYAD02/4A . The address code is hexadecimal digits. However, all the new SY modules use the same factory initial setting as following:

Address code is 01H

baud rate is 9600 bps

Checksum prohibited

Communication protocol is character communication protocol ASCII

As the new module code is the same, their address will be contradictory with other modules, so when you set up the system, you must re-configure each analog output module address. Users can modify SYAD02/04A module's address through configuration commands after connecting SY 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 configuration state firstly; otherwise it can not be modified.

Let the modules be in default status:

SYAD02/04A module has a Pin marked CONFIG. Connect CONFIG Pin to the ground (GND) as Figure 4, and then connect power supply. In this way, the module is in configuration status.

Under default status, the module setting is as following:

Address code: 00H

Baud rate: 9600 bps

Checksum: disable

ASCII Character communication protocol

At this time, users can modify SYAD02/04A 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 configuration 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.

Mode Calibration

The calibration must carry out under the communication protocol of ASCII. The Modbus protocol can not support calibration.

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.
2. For SYAD02/4A, Zero is calibrated when the input is 0; full scan is calibrated when input 120% 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: 6V.
3. Input zero signal to the channel, usually it is 0mA or 0V.
4. 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.
5. Input current or voltage signal of 120% of FSR to the channel which needs calibration.
6. When the signal is stable, send gain calibration \$ AA0N command (N represents the channel code currently being calibrated, 0 or 1) to the analog input module.
7. Calibration is finished.

ASCII Character Communication Protocol

Brief introduction

When the controller communicates under ASCII character protocol mode, every 8-bit of 1 message is transmitted as two ASCII bytes. E.g. Transmit the number 34: it transmits 0x33 (ASCII code of 3) and 0x34 (ASCII code of 4) separately. The advantage is that the interval of sending the bytes has no limitation.

Character protocol command set

The command consists of a series of characters, such as the leading code, address ID, variables, and optional checksum byte and a stop command (cr). SYAD02/04A module does not support the broadcast address, so the host only commands one SYAD02/04A module each 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.	1-C
[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 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 \$022(cr)
 Module response !02000600 (cr)

Enable checksum
 User command \$022B8 (cr)
 Module response !02000640AD (cr)

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

B8=(0x24+0x30+0x32+0x32) AND 0xFF

'!' = 0x21 '0' = 0x30 '2' = 0x32 '4' = 0x34 '6' = 0x36

AD=(0x21+0x30+0x32+0x30+0x30+0x30+0x36+0x34+0x30) AND 0xFF

Common command

1. Read analog input module data
2. Read the analog output value of channel N
3. Module allocation
4. Read allocation status
5. Offset calibration
6. Full scale calibration
7. Read module name
8. Enable or prohibit channel command
9. Read channel status command
10. Set communication protocol 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, 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 analog input module 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) command is valid.

?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, PERCENT OF FSR, complement of hexadecimal or ohms. Refer to Chapter 3 Command Set.

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

More instructions: If there is a address error or communication error,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 (cr)

Description : on module of 23H, input is (data format is engineering unit):

Channel 0: +04.765mA channel 1: +04.756mA

2. Read the data of analog input module in Channel N

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

Syntax : #AA(N)(cr)

Parameter Description : # delimiter character

AA Module address, value range 00H~FFH。

N channel 0 or 1 for 2 channel and 0-3 for 4 channel module

(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)。

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)

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

3. Module allocation

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 : %AANN TTCCFF(cr)

Parameter Description : % delimiter character

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

NN hexadecimal address, NN is from 00 to FF(Hexadecimal)

TT type code. SYAD 02/4A should be set as 00

CC hexadecimal represents baud rate code. **Refer to table 2**

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

Table 2 Baud rate code

FF The 8-bit of hexadecimal is used to represent data format and checksum. Note: From bits2 to bits 5,if it is no need to use, it is need to be set to 0.

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

Table 3 data format, checksum code

Bit7: Reserved bits should be set to 0

Bit6: checksum status: 0 represents prohibit; 1 represents permission

Bit5-bit2: no need to use and 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 , carriage return (0DH)。

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

?AA(cr) invalid command or illegal operation, no jumper allocation before change the 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)

More instructions : If need to reallocate module, user should install jumper to let the module be in configuration status. Now Module address is 00H, that is to say AA=00H, NN = new address. Otherwise it will respond error signal.

Once the command is executed, 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)**

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, SYAD 02/4A series should be set to 00.

06 Indicate baud rate9600bps

00 Indicate data format is engineering unit and the checksum is forbidden.

4. Read the allocation 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: **!AATCCFF(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)

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 **\$302(cr)**
 Module response **!300F0600(cr)**

Description : **!** delimiter character
30 analog input module address is 30H
00 input type code
06 baud rate 9600 bps.
00 data format is engineering unit and the checksum is forbidden.

5. Offset calibration

Description : Adjust the zero offset of channel N

Syntax : **\$AA1N(cr)**

Parameter Description : **\$** delimiter character

AA Module address, value range 00H~FFH
1 Offset calibration
N channel 0~3
(cr) end character , carriage return (0DH)。

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)。

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 command **\$2310(cr)**
 Module response **!23(cr)**

Description : offset calibrating channel 0 of the address 23H

6. Gain calibration command

Description : calibrate the gain of channel N

Syntax : **\$AA0N(cr)**

Parameter Description : **\$**delimiter character

AA Module address, value range 00H~FFH
0 Gain calibration command
N channel 0~3
(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)。

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 command **\$2303(cr)**

Module response **!23(cr)**

Description : gain calibrating channel 3 of address 23H

7. Read module name

Description : Read module name

Syntax : \$AAM(cr)

Parameter Description : \$ delimiter character

AA Module address, value range 00H~FFH

M Read module name

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

Response: **!AA(Module Name)(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。

(Module Name) module name could be SYADA08、SYAD02/04A 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 **!08SYAD02A(cr)**

Description : Module SYAD02A is in address 08H module.

8. Enable or prohibit command channel

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

Syntax : \$AA50V(cr)

Parameter Description : \$ **delimiter character**

AA Module address, value range 00H~FFH。

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

V A hexadecimal number, The corresponding binary expressions of 1~0 represents 1~0 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)。

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 **\$08503(cr)**
 Module response **!08 (cr)**

Description : set channel value to 0x03。

3 is 0011, represents to enable channel 0, channel 1.

9. 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: **!AA0V(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。

V 1 hexadecimal number, bits 1~0 represents 1~0 channel

Bit is 0: prohibit the channel

Bit is 1: enable the channel

(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 **\$186 (cr)**
 Module response **!1803(cr)**

Description : the current channel status is 0x03.

0x03 is 0000 and 0011, means all the channels of module at address 18H have been enabled.

10. Set communication protocol command

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

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

Examples:

1. input range is $\pm 20\text{mA}$, input is 4 mA:

	user 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. output range is $\pm 10\text{V}$, output is 2.5V

	user command	#01(cr)
Engineering Unit	Module response	>+02.500(cr)
% of FSR	Module response	>+025.00(cr)
complement of hexadecimal	Module response	>1FFFFF(cr)

3. conversion relation between hexadecimal complement and analog data

When the module is the voltage/current input type. It is possible to be converted as the following formula: $X/7FFFFFF=Xin/Xf$

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

Xin: represents the analog signal what the channel inputted.

Xf: represents the analog signal of the scale span of product, e.g: 20mA.

4. In a voltage input type, Xin is the channel voltage input Vin, Xf is the full scale voltage input.

5. If the input voltage is negative voltage/current, read the value and take back plus one, then, put it into the above formula to calculate the corresponding analog values.

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

Master device can communicate with one slave device, but also communicate with all slave 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 slave 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 an error occurred during message reception, or the slave device can not perform its command, the slave 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"> hexadecimal, ASCII character 0~9, A~F every ASCII byte consists of one hexadecimal character 	<ul style="list-style-type: none"> 8 bit binary system, hexadecimal 0~9, A~F Every 8 bit consists of one or two hexadecimal character
Data bit	<ul style="list-style-type: none"> 1 start bit 7 data bits, the smallest significant bit sent firstly 1 parity check bit, none if there is no calibration 1 stop bit (with calibration), 2 Bits (without calibration) 	<ul style="list-style-type: none"> 1 start bit 8 data bits, the smallest significant bit sent firstly 1 parity check bit, none if there is no calibration 1 stop bit (with calibration), 2 Bits (without calibration)
Error detection	LRC (Longitudinal rigmarole check)	CRC (circulate rigmarole check)

1. Set Modbus RTU mode

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

- Short circuit CONFIG (the 3rd pin) and GND (the 4th pin)
- Correctly link Power and other communication interface cables
- Switch on power supply, module is in configuration state automatically, communication address is **00**, baud rate is **9600**.
- The module is to finish initialization, wait 1 minute.
- Send command **\$00P1 (cr)(00 is the relevant target module address)**, check response, if it is **!00(cr)**, setup correctly.
- Switch off power supply, cut off the connecting between CONFIG and GND
- 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

Slave device response

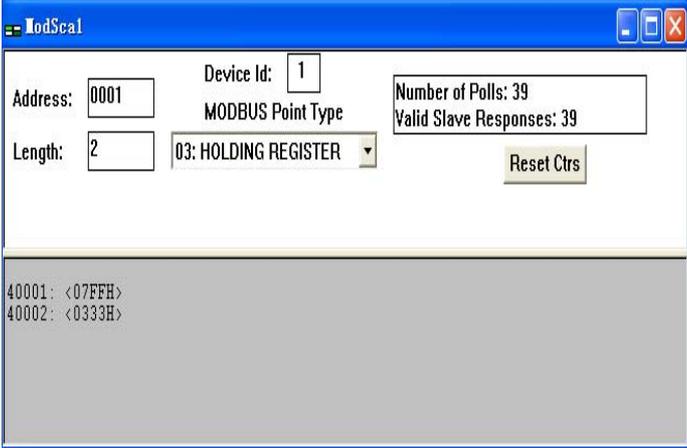
Register data in the response information of the slave device corresponds to the two bytes of each register; the first data byte is high, the second is the low data. Response format is as following:

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 (CRC)
01	03	10	19	99	00	00	02	9E 68

Slave device response information Rx: 01 03 10 19 99 00 00 00 00 00 00 00 00
00 04 00 00 00 00 87 69

- 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 Table is the register information checked by MODSCAN software and its instruction

	<p>Instructions:</p> <p>Address is the start address of the register Device ID is module address Length is the number of registers been read MODBUS Point Type :Modbus function type selection</p> <p>40001: corresponds to register data, analog input value of channel 0 40002: corresponds to register data, analog input value of channel 1</p>
--	--

4. Instructions of register:

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 (only for 4channel product)
40004	IN3	Read only	Measured Value of Channel 3 (only for 4channel product)
40211	Module Name	Read only	Higher bits: 0x40 Lower bits: 0x21
40221	Channel Status	Read/Write	Higher bits: 0x00 Lower bits: Channel Status (0x3 / 0xFF)

Table 5 Modbus RTU register Instructions