2-channel grating scale magnetic grating scale encoder or 4-channel high-speed DI pulse signal to Modbus TCP network module WJ192

Product features:

- Grating ruler magnetic grating ruler decoding is converted to standard Modbus TCP protocol
- High speed grating ruler magnetic grating ruler 4 times frequency counting, frequency up to 5MHz
- Support two grating rulers to count at the same time, which can identify the positive and reverse directions
- Can be set as 4-way independent DI high-speed counter
- Encoder and DI input support PNP and NPN input
- Filter time can be set when relay and mechanical switch input
- All data can be viewed directly on the webpage without other software
- Encoder count value and DI count support automatic saving in case of power failure
- Isolation between DI input and network communication interface
- Clear and set the count value through the network communication interface
- Wide power supply range: 8~32VDC
- High reliability, convenient programming and easy application
- Standard DIN35 guide rail installation, convenient for centralized wiring
- The user can set the module IP address and other parameters through the web page
- Overall dimension: 106 mm x 37 mm x 59 mm

1 WJ192 Module Appearance

Typical applications:

- Grating ruler magnetic grating ruler length measurement
- Flow meter pulse counting or flow measurement
- Product count of production line
- Position Data Measurement Drawing of CNC Machine Tool
- Encoder signal is transmitted to IPC
- Intelligent factory and industrial Internet of Things
- Directly transfer data to control center instead of PLC

Product Overview:

WJ192 product realizes signal acquisition between sensor and host, which is used to decode the signal of grating scale magnetic grating scale encoder.WJ192 series products can be used in Ethernet bus industrial automation control system, automated machine tools, industrial robots, three coordinate positioning system, displacement measurement, stroke measurement, angle measurement, speed measurement, flow measurement, product counting, etc.

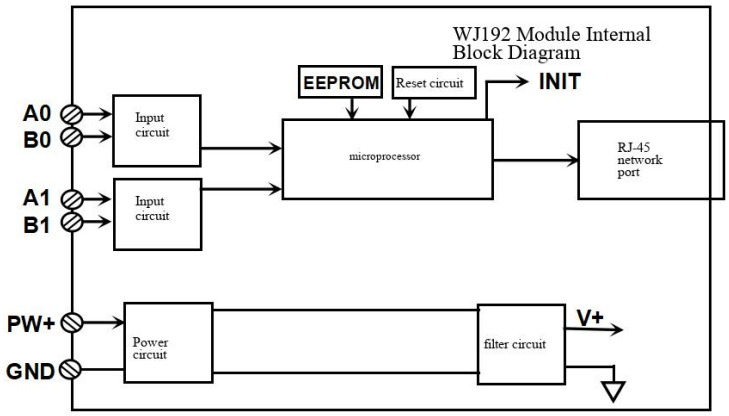


Figure 2 WJ192 Module Internal Block Diagram

The products include signal isolation, pulse signal capture, signal conversion and Ethernet communication.The communication mode adopts ASCII code communication protocol or MODBUS TCP communication protocol. TCP is a protocol based on the transport layer. It is a widely used, connection oriented and reliable protocol.Users can directly set the module IP address, subnet mask, etc. on the web page.

WJ192 series products are intelligent monitoring and control systems based on single chip microcomputer. Configuration information such as module IP address and subnet mask set by users are stored in non-volatile memory EEPROM.

WJ192 series products are designed and manufactured according to industrial standards, with signal input/communication isolation, strong anti-interference capability and high reliability.Operating temperature range -40 ℃~+85 ℃.

Function introduction:

WJ192 remote I/O module can be used to measure two encoder signals, and can also be set as four independent counters or DI status measurement.

1. Signal input

2-way encoder signal input or 4-way DI independent counter can be connected to dry contact and wet contact. Please refer to the wiring diagram for details.

2. Communication protocol

Communication interface: RJ-45 network interface.The Link light (green light) is always on and the Data light (yellow light) flashes irregularly after the network cable is plugged in.

Communication protocol: It supports MQTT protocol and can connect to various MQTT servers such as Alibaba Cloud, Tencent Cloud, Huawei Cloud, China Mobile IOT OneNET, Private Cloud, etc.MODBUS TCP protocol can also be used to realize data exchange of industrial Ethernet.

It also supports TCP/UDP/WebSocket and other communication protocols.

Network cache: 2K Byte (both receiving and sending)

Communication response time: less than 10mS.

3. Anti interference

Checksum can be set as required.There are transient suppression diodes inside the module, which can effectively suppress various surge pulses, protect the module, internal digital filtering, and also can well suppress power frequency interference from the power grid.

Product selection:

WJ192 - RJ45

Communication interface

RJ45: output as RJ-45 network interface

WJ192 general parameters:

(typical @+25 ℃, Vs is 24VDC)

Input type: **encoder AB signal input, 2-channel (A0/B0~A1/B1).**

Low level: input<1V

High level: input 3.5~30V

Encoder frequency range 0-5MHz (all channels input at the same time)

The encoder counting range is -2147483648 °～+2147483647. It uses 4-fold frequency counting. It is automatically saved after power failure. After positive overflow, it automatically increases from the minimum negative value, and after negative overflow, it automatically decreases from the maximum positive value.

DI counter signal input, 4-channel (A0, B0~A1, B1).

Low level: input<1V

High level: input 3.5~30V

The measuring frequency can reach 5MHz.If the low speed measurement is required to be accurate, it can be set to 10KHz low speed measurement mode.

The range of DI counter is 0~4294967295, which is automatically saved when power is off, and automatically increases from 0 after positive overflow.

Input resistance: 30K Ω

Communication: MQTT communication protocol or MODBUS TCP communication protocol or TCP/UDP

Web page: support the online viewing of data on the web page, and support the setting of module parameters on the web page.

Interface: RJ-45 network interface, built-in isolation transformer.

Working power supply:+8~32VDC wide power supply range, internal anti reverse connection and overvoltage protection circuit

Power consumption: less than 3W

Operating temperature: -45~+80 ℃

Operating humidity: 10~90% (no condensation)

Storage temperature: -45~+80 ℃

Storage humidity: 10~95% (no condensation)

Overall dimensions: 106 mm x 37 mm x 59 mm

Pin definition:

Pin	Name	Description	Pin	Name	Description
one	INIT	Enter the AP configuration mode switch	**three**	B1	Encoder 1 signal B input terminal
			four	A1	Encoder 1 signal A input terminal
			five	GND	Power supply negative terminal, signal common terminal
two	RJ-45	RJ-45 network port	**six**	B0	Encoder 0 signal B input terminal
			seven	A0	Encoder 0 signal A input terminal
			eight	GND	Negative terminal of power supply
			nine	PW+	Positive end of power supply

Table 1 Pin definition

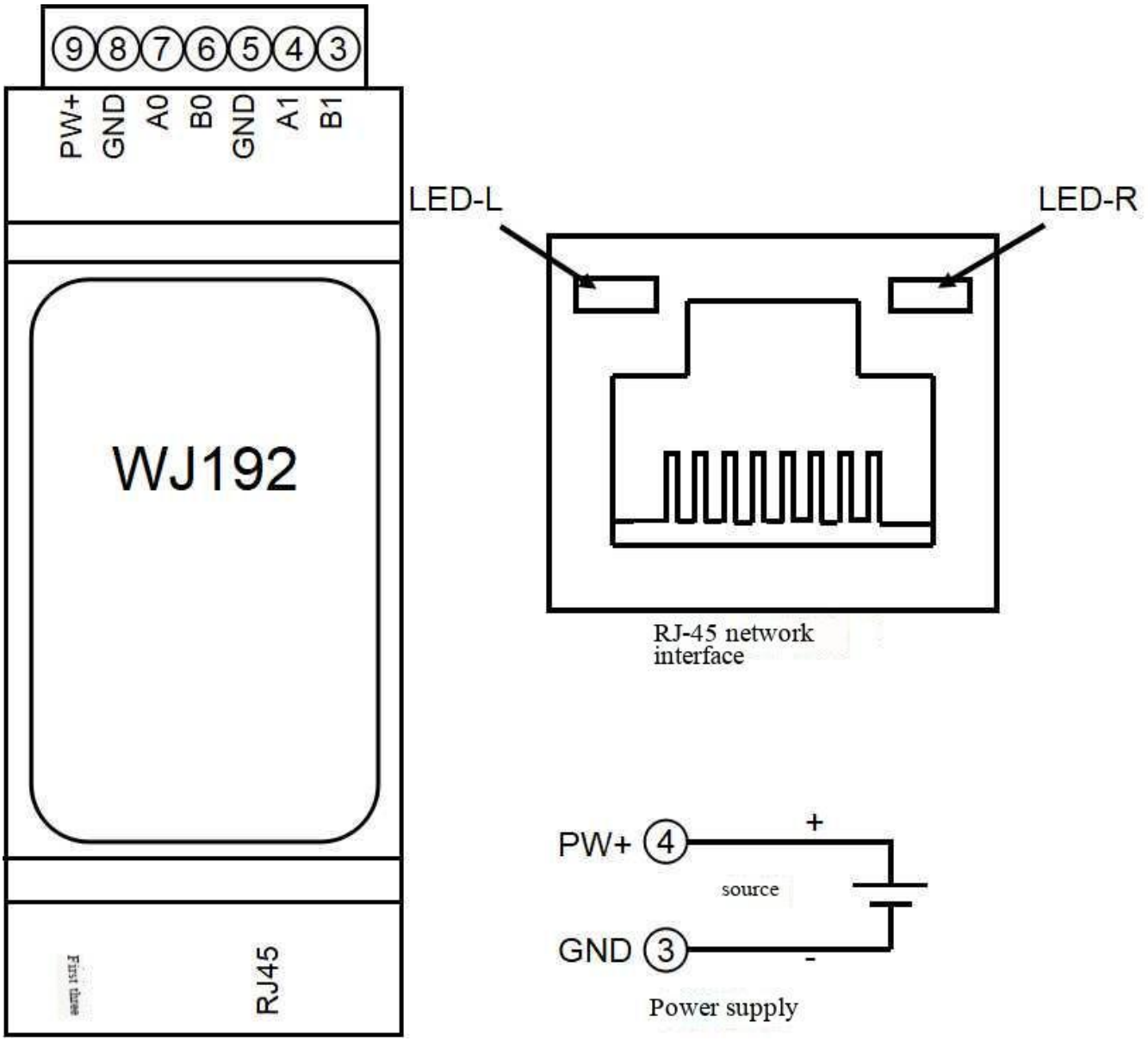
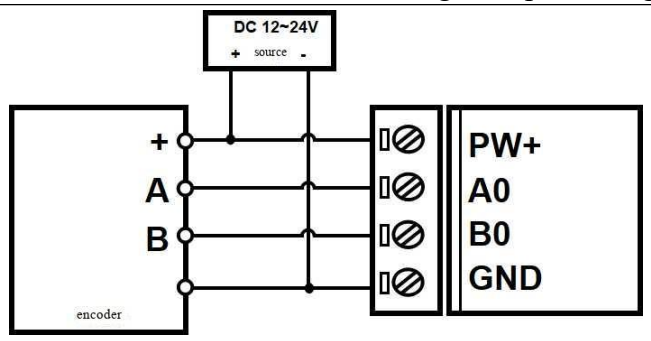
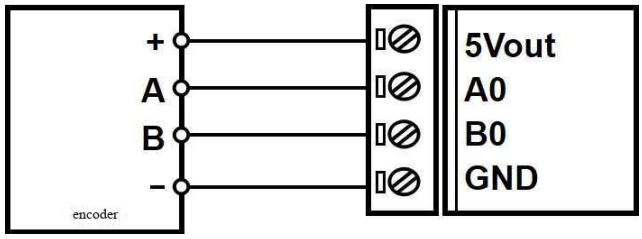


Figure 3 WJ192 Module Wiring Diagram

Encoder signal input wiring diagram (**counting mode 0**)



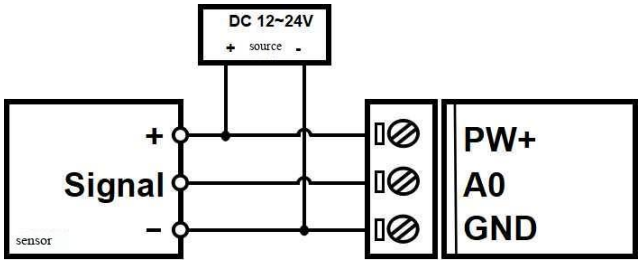
Encoder powered by 12-24V



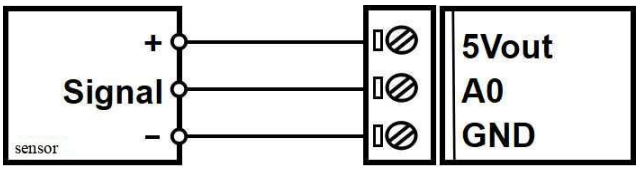
Encoder powered by 5V

Note: The factory default is to turn off the pull-up. If it is an NPN encoder, the internal pull-up resistor needs to be turned on;Others such as NPN encoder with pull-up resistance, PNP encoder, push-pull encoder, etc. can be used directly.Turn on or off the internal pull-up resistor can be set on the configuration page, as shown in the following figure:

DI independent counter input wiring diagram (**counting mode 1**)

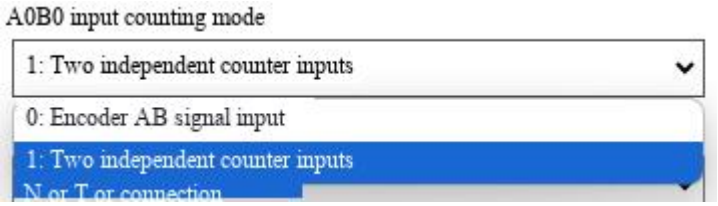


12-24V powered sensor



5V powered sensor

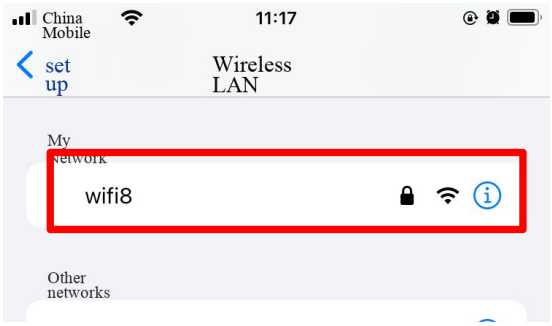
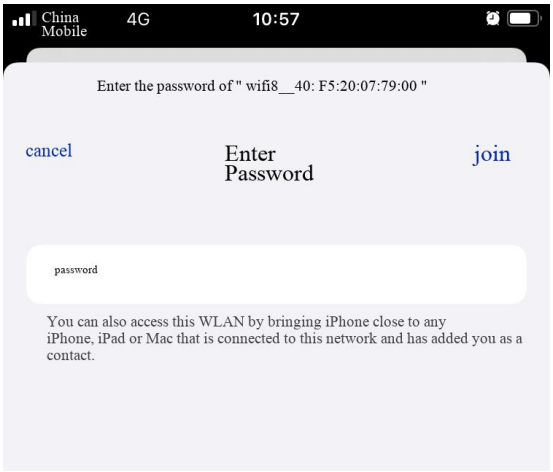
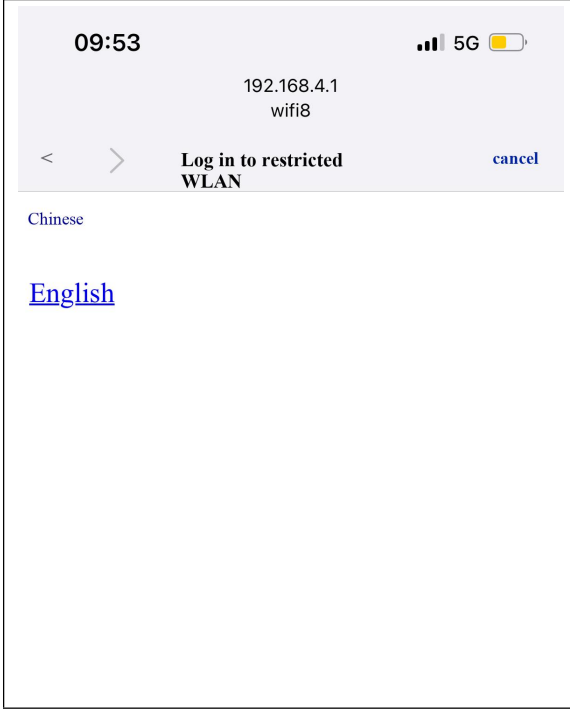
Note 1: The factory default is working mode 0, and the DI count needs to be changed to working mode 1 by issuing a command, as shown in the following figure:



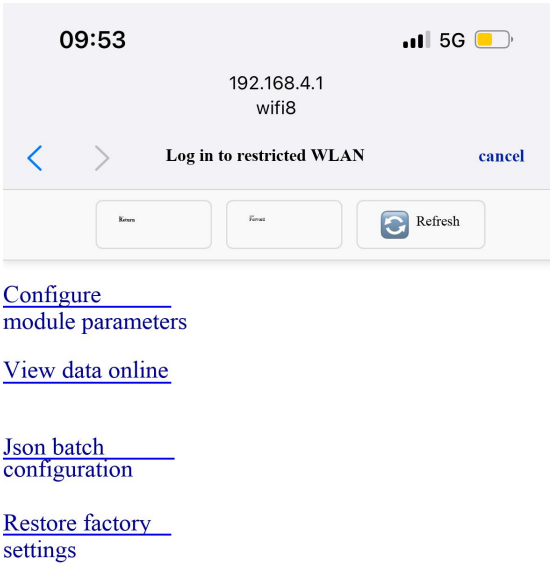
Note 2: The factory default is to turn off pull-up. If it is an NPN sensor, dry contact or switch input, the internal pull-up resistance needs to be turned on;Others, such as NPN sensor with pull-up resistance, PNP sensor, push-pull sensor, TTL level, etc., can be used directly.Turn on or off the internal pull-up resistor can be set on the configuration page, as shown in the following figure:

First, you can configure the WJ192 module through your mobile phone

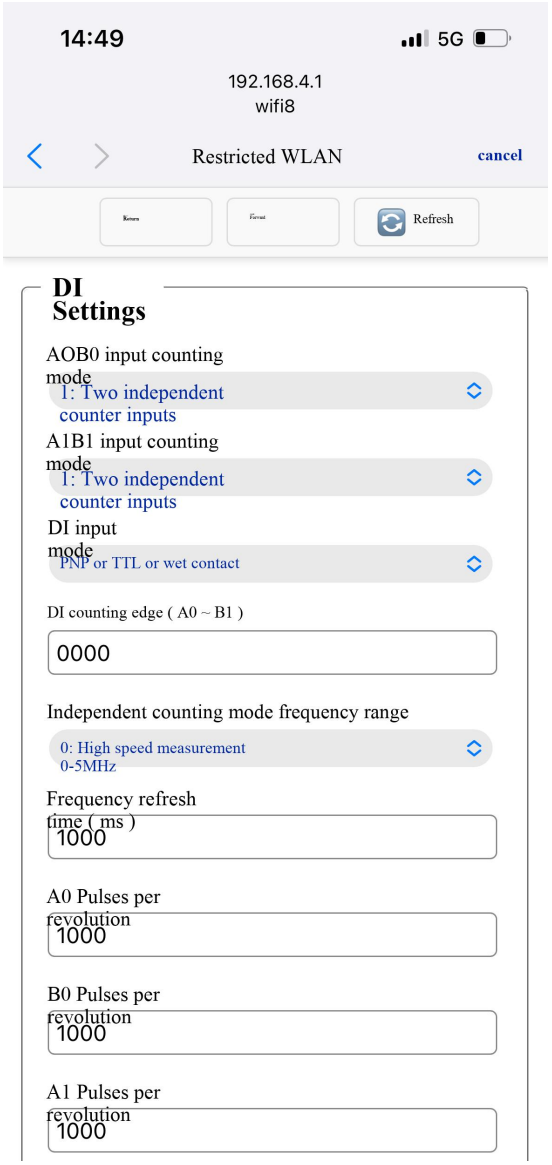
In addition, if the network segment of the computer is 192.168.0.xx, you can also connect the network cable to the module, and enter the default IP address (192.168.0.7) of the browser module to log in to the webpage configuration module of the module.

<table>
<tr>
<td> 
</td>
<td><h2>1. Let the module enter AP mode</h2>
(1) Turn on the power, press and hold the module switch (INIT) for 3 seconds, and then release it.
(2) Open the mobile phone "WLAN" or "Settings → WLAN", find the WiFi named "wifi8" to connect.</td>
</tr>
<tr>
<td> 
</td>
<td>The factory password of this module is 12345678, and then "join".</td>
</tr>
<tr>
<td> 
</td>
<td><h2>2. Enter the module webpage.</h2>
After connecting the WiFi of the module, it will automatically jump to the module's built-in webpage after a few seconds, as shown in the left figure.If the mobile phone cannot automatically jump, you can also open the mobile browser and enter the website 192.168.4.1 to log in.
Click the configuration module parameter link to enter the configuration interface.

Then select Chinese or English.</td>
</tr>
</table>



Select Chinese here to enter the Chinese menu.



3. Configure module DI parameters

Please modify the following parameters according to actual needs:

（1） A0B0~A1B1 input counting mode:

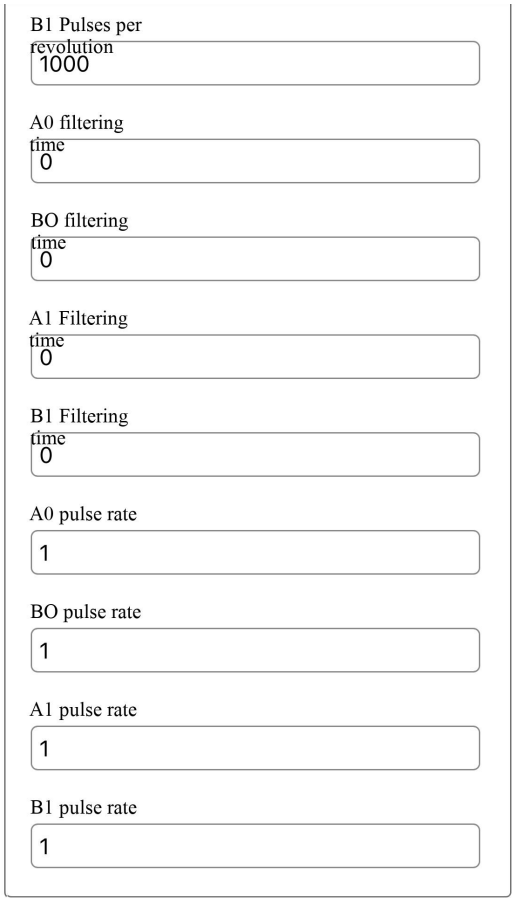
Counting mode 0: encoder AB signal input;

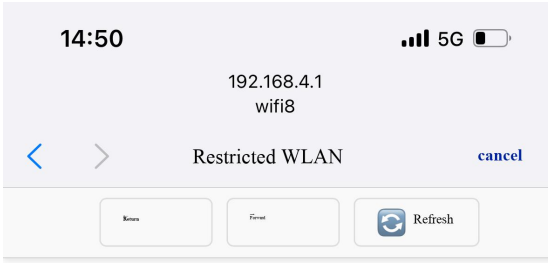
Counting mode 1: two independent counter inputs;

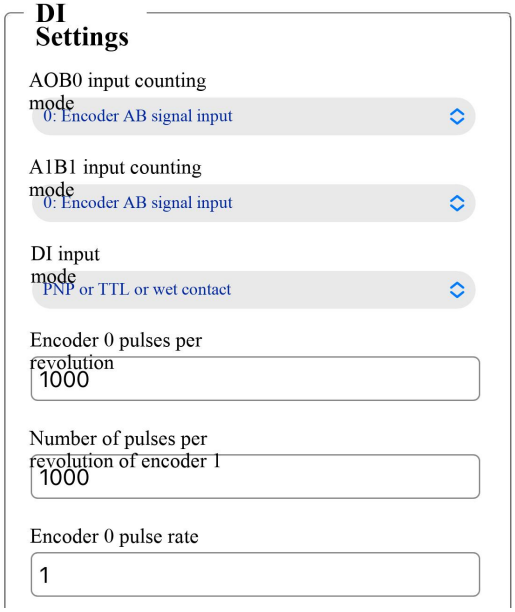
Please fill in according to the actual input sensor, and select encoder AB signal input for grating ruler and magnetic grating ruler.

（2） DI counting edge: different edge trigger counts can be set, 0 indicates the rising edge count, and 1 indicates the falling edge count.Normal use uses the default rising edge count.

（3） A0~B1 pulses per revolution: pulses per revolution of DI. If you need to measure the speed, please set it according to the actual parameters.The module will automatically convert the rpm.

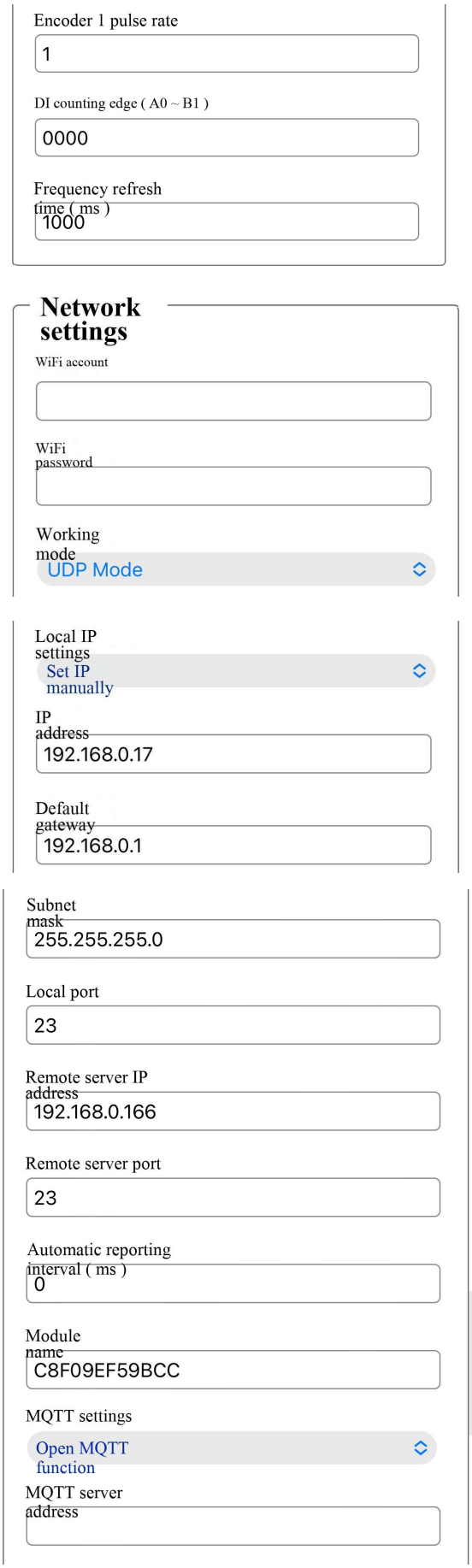






（4） A0~B1 filtering time: the value range is 0 to 65535.
If it is 0, it means no filtering;Other values represent the filtering time, in mS (millisecond).If the DI input point is a mechanical switch or a mechanical relay, it is recommended to set the filtering time as 20mS.

（5） A0~B1 pulse magnification: set the actual value corresponding to each pulse. The default value is 1. The actual engineering value is converted from this value to the actual pulse.For example, if each pulse is 0.005mm, it can be set to 0.005, then the actual engineering value is 0.005 * pulse number.



（6） Encoder 0~1 pulses per revolution: pulses per revolution of the encoder. If you need to measure the speed, please set it according to the actual parameters.The module will automatically convert the rpm.

（7） Encoder 0~1 pulse magnification: set the actual value corresponding to each pulse. The default value is 1. The actual engineering value is converted from this value and the actual number of 4-fold frequency pulses.For example, if each pulse is 0.005mm, it can be set to 0.005, then the actual engineering value is 0.005 * pulse number.

4. Configure module network parameters

Please modify the following parameters according to actual needs:

（8） WiFi account: Connect the WiFi covered here.(Do not fill in WiFi parameters for those connected by network cable, otherwise wifi networking will be preferred)

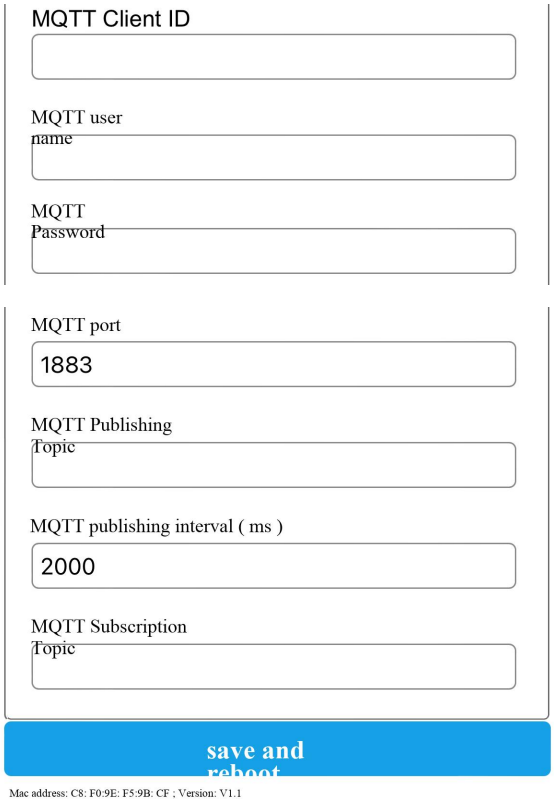
（9） WiFi password: fill in the WiFi password. If it is already connected, do not enter it again.

（10） Working mode: select the working mode and fill in according to the actual application.

Optional TCP Server, TCP Client, UDP, MODBUS TCP, Websocket, etc.

（11） Local IP setting: If only MQTT protocol is used, it can be set to automatically acquire IP. If Modbus TCP or web page is required to access data, it is recommended to manually set it to a fixed IP to facilitate communication through IP address and module.

（12） IP address: set the IP address of the module. It must be the network segment where the current



WiFi is located and not the same as the IP address of other devices in the LAN.For example, if the IP address of the WiFi router is 192.168.0.1, you can set the IP address of the module to 192.168.0.7

（13） Default gateway: the gateway of the module. Fill in the IP address of the current WiFi router.For example, the IP address of the WiFi router is 192.168.0.1. Just fill in the IP address

（14） Subnet mask: the subnet mask of the module. If there is no cross network segment, fill in the default value of 255.255.255.0

（15） Local port: communication port of the module, and 502 port is generally used for MODBUS communication.

（16） Remote server IP address: the remote server IP address, the server that TCP Client and UDP need to connect to.

（17） Remote Server Port: The port of the server.

（18） Auto Report Interval: the time interval for the module to report data regularly. If it is set to 0, data will not be automatically reported.

（19） Automatic reporting of count changes: a piece of data is reported when the count changes. It can only be used in situations where the data changes very slowly, or a large amount of data will be sent.

（20） Module Name: the user defines the name of a module to distinguish different modules.

（21） MQTT setting: If MQTT communication is used, the MQTT function needs to be turned on.

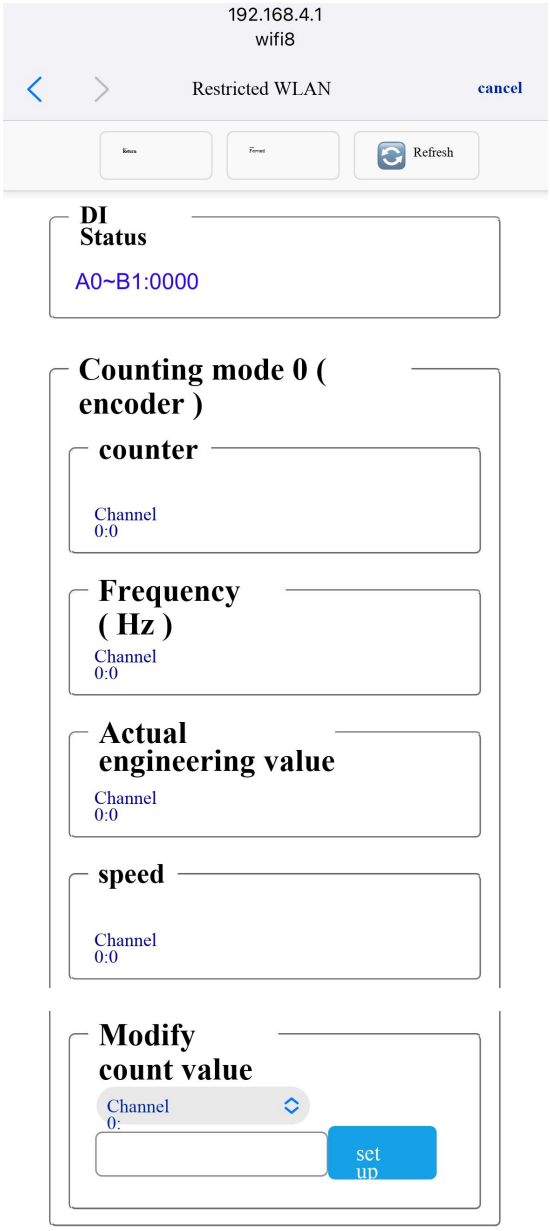
（22） MQTT server address: fill in the URL of the MQTT server,
For example: broker.emqx.io
If the IP address of the local server is 192.168.0.100, you can write 192.168.0.100

（23） MQTT Client ID， Please fill in the user name, password, port, publishing topic, subscription topic and other parameters according to the requirements of the MQTT server.The QoS of MQTT is 0 and cannot be modified.

（24） MQTT publishing interval: the interval between the module automatically publishing data to the MQTT server, in ms. Setting it to 0 means canceling the scheduled publishing function.

5. Save parameters

After parameter setting, click the Save and Restart button, the module will save the parameters and restart automatically, then turn the switch on the side of the module to the normal position, and the module will work according to the set parameters.



6. View data online on the webpage

Click the online view data link on the module home page to enter the data view interface.As shown in the left figure.

The left figure shows that channel 0 is set to mode 0 (encoder), and channel 1 is set to mode 1 (single pulse)

If the IP address of the module is 192.168.0.7, the user can also access the link 192.168.0.7/readData to obtain data in Json format.

DI status indicates the level status of the input.

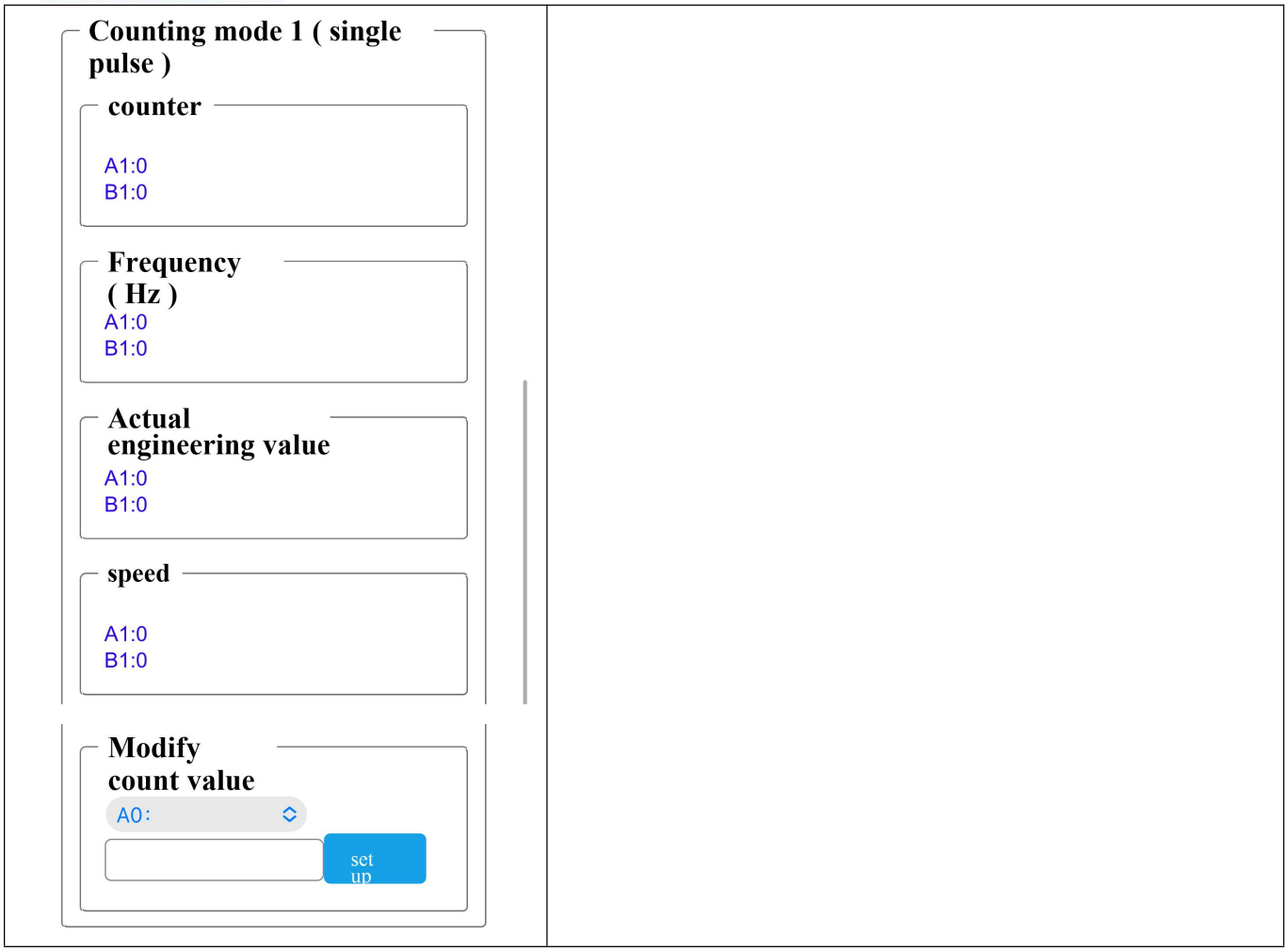
The pulse counter is the cumulative number of measured pulses.

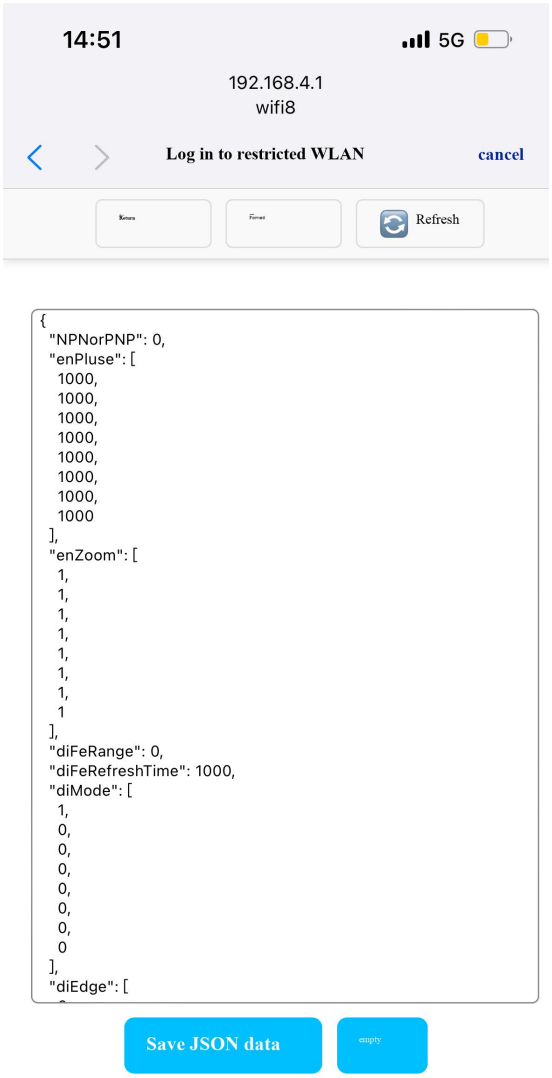
The pulse frequency is the number of pulses per second.

The actual engineering value is obtained by multiplying the value of the pulse counter by the pulse multiplication rate set on the web page.It is used to automatically convert the actual flow, length, output and other data.

The speed is converted from frequency and pulses per revolution.It is used to automatically convert the actual rpm.

To clear the count value, you can write 0 to the table, and then click Set to clear the count value.You can also set other values to modify the count value.


Counting mode 1 (single pulse)
counter
A1:0
B1:0
Frequency (Hz)
A1:0
B1:0
Actual engineering value
A1:0
B1:0
speed
A1:0
B1:0
Modify count value
A0:
set up



8. Set parameters in batch

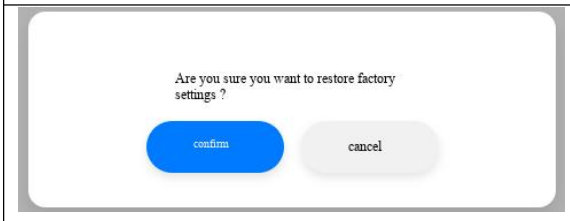
Click the Json batch configuration link on the module homepage to enter the batch setting interface.As shown in the left figure.
The data must be in standard Json format. You can set all parameters or only some parameters.
If there are many products to set, you can save time by batch setting.
Click Save Json data after filling in.

Example 1: Only modify the WiFi account password to send:

```
{
  "WifiSsid": "w",
  "WifiPassword": "12345678",
  "setIP": 1,
  "ipAddress": "192.168.0.5",
  "gateway": "192.168.0.1",
  "netmask": "255.255.255.0",
}
```

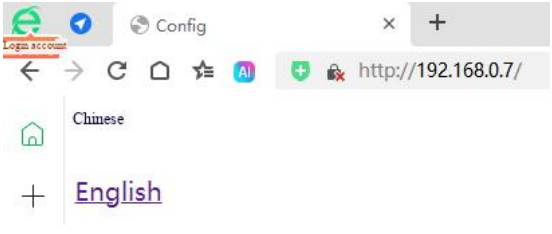
Example 2: Only modifying MQTT parameters can send:

```
{
  "setMQTT": 1,
  "mqttHostUrl": "broker.emqx.io",
  "port": 1883,
  "clientId": "mqtt_test_001",
  "username": "",
  "passwd": "",
  "topic": "mqtt_topic_001",
  "pubTime": 2000,
  "pubonchange": 0
}
```



9. Restore factory settings

Click the "Restore Factory Settings" link on the module home page to pop up a confirmation window, as shown in the left figure.Click "Confirm" to restore factory settings.



10. The module webpage can also be opened on the LAN

If the module has been connected to the local network or wifi, you can enter the module IP in the computer or

	mobile browser, such as 192.168.0.7, to open the module web page (provided that the computer IP or mobile IP is in the same network segment as the module, the login web page should be based on the IP address of the current module), and then you can enter the module internal web page.You can also configure the module or read the module data in the same way as the above table.

Character communication protocol:

MQTT protocol: After the connection is successful, send a command to the MQTT subscription topic of the module, and the reply data is displayed on the MQTT publication topic of the module.

TCP Server, TCP Client, UDP Mode, Web Socket and other working modes: After the connection is successful, you can send commands and receive data.

If automatic reporting is set, the submitted data format is the same as the reply format of (1. Read Data Command).

1、 Read Data Command

Send: # 01 (If scheduled automatic reporting is set, the module will report data regularly instead of sending commands)

Reply: {"devName": "C049EF6A95C4", "time": 1141805, "diMode": [0,1], "diState": [0,0,0], "enCounter": [0,0], "enFrequency": [0,0], "enActualData": [0,0], "enSpeed": [0,0], "diCounter": [0,0,0], "diFrequency": [0,0,0,0], "diActualData": [0,0,0,0 0,0], "diSpeed": [0,0,0,0]}

Format description:

Encoder data is arranged in the order of channel 0~8;The independent DI data is arranged in the order of A0, B0~A7, and B7.

"DevName" module name, which can be modified on the webpage as required

"Time" module internal time, in mS.

"DiMode" module counting mode.**Counting mode 0**: encoder AB signal input;**Counting mode 1**: two independent counter inputs

"DiState" DI state indicates the level state of the input.

The accumulated number of pulses measured by the "enCounter" encoder counter adopts the 4-fold frequency counting method.(**Counting mode 0**)

"EnFrequency" encoder pulse frequency is the number of pulses per second.(**Counting mode 0**)

The actual engineering value of the "enActualData" encoder is obtained by multiplying the value of the encoder pulse counter by the pulse multiplication rate set on the web page.It is used to automatically convert the actual flow, length, output and other data.(**Counting mode 0**)

"EnSpeed" encoder speed is converted from encoder frequency and pulses per revolution.It is used to automatically convert the actual speed or flow per minute.

(**Counting mode 0**)

"DiCounter" The cumulative number of pulses measured by the independent counter.(**Counting mode 1**)

The "diFrequency" pulse frequency is the number of pulses per second.(**Counting mode 1**)

The actual project value of "diActualData" is obtained by multiplying the pulse counter value by the pulse multiplication rate set on the web page.It is used to automatically convert the actual flow, length, output and other data.(**Counting mode 1**)

"DiSpeed" speed is converted from frequency and pulses per revolution.It is used to automatically convert the actual

rpm.**(Counting mode 1)**

You can also read a single group of data, such as the encoder counter:
Send: # 01>enCounter
Reply: {"enCounter": [0,0]}
For example, read the actual engineering value of encoder:
Send: # 01>enFrequency
Reply: {"enFrequency": [0,0]}

Read other parameters and send corresponding parameter characters.

2. Set encoder 0~1 count value command
The encoder 0~1 count value can be 0 or other values, and can be reset or modified.
Send: $01 {"setEn0Count": 0, "setEn1Count": 0}
Only set a single channel: $01 {"setEn0Count": 0}
Set the same value to all channels at the same time: $01 {"setAllENCount": 0}
Reply:!01 (cr) **indicates successful setting;**?01 (cr) indicates command error

3. Set pulse counter A0~B1 count value command
Set the value of pulse counter A0~B1, which can be 0 or other values, and can be cleared or modified.
Send: $01 {"setA0Count": 0, "setB0Count": 0, "setA1Count": 0, "setB1Count": 0} or $01 {"setA0Count": 1000, "setB0Count": 2000, "setA1Count": 3000, "setB1Count": 1}
Only set a single channel: $01 {"setA0Count": 0}
Set the same value to all channels at the same time: $01 {"setAllDICount": 0}
Reply:!01 (cr) indicates successful setting;?01 (cr) indicates command error

4、 Read configuration command
Read the configuration parameters of the module, or view them directly on the web page.
Sent:% 01ReadConfig
Re:
{"NPNorPNP":0,"enPluse":[1000,1000],"enZoom":[1,1],"diFeRange":0,"diFeRefreshTime":1000,"diMode":[0,1],"diEdge":[0,0,0,0],"diPluse":[1000,1000,1000,1000],"diFilter":[0,0,0,0],"diZoom":[1,1,1,1],"saveData":1,"WifiSsid":"","WifiPassword":"","workmode":0,"setIP":1,"ipAddress":"192.168.0.77","gateway":"192.168.0.1","netmask":"255.255.255.0","localPort":23,"remoteServerIp":"192.168.0.100","remotePort":23,"sendTime":0,"devName":"C049EF6A95C4","setMQTT":0,"mqttHostUrl":"","port":1883,"clientId":"","username":"","passwd":"","topic":"","pubTime":2000,"subtopic":""}
5、 Set Configuration Command
The configuration parameters of the module can also be set directly on the web page.All or some parameters can be set, and the module will restart automatically after setting.
Send:
%01WriteConfig{"NPNorPNP":0,"enPluse":[1000,1000],"enZoom":[1,1],"diFeRange":0,"diFeRefreshTime":1000,"diMode":[0,1],"diEdge":[0,0,0,0],"diPluse":[1000,1000,1000,1000],"diFilter":[0,0,0,0],"diZoom":[1,1,1,1],"saveData":1,"WifiSsid":"","WifiPassword":"","workmode":0,"setIP":1,"ipAddress":"192.168.0.77","gateway":"192.168.0.1","netmask":"255.255.255.0","localPort":23,"remoteServerIp":"192.168.0.100","remotePort":23,"sendTime":0,"devName":"C049

EF6A95C4","setMQTT":0,"mqttHostUrl":"","port":1883,"clientId":"","username":"","passwd":"","topic":"","pubTime":2000,"subtopic":""}

You can also set only a single parameter, for example, to modify the IP address:% 01WriteConfig {"ipAddress": "192.168.0.15"}

For example, set power failure to not save count value:% 01WriteConfig {"saveData": 0}

Reply:!01 (cr) **indicates successful setting;**?01 (cr) indicates command error

6、 Restore factory settings command

Restoring factory settings is to restore the configuration module data to the factory default state.

Send: $01 {"factoryReset": 1}

Reply:!01 (cr) indicates successful setting;?01 (cr) indicates command error

Modbus communication protocol:

The register table of Modbus TCP communication protocol is as follows:

Support function code 01

Address 0X (PLC)	**Address (PC, DCS)**	**Data content**	**attribute**	**Data description**
00001	zero	A0 input status	read-only	Level status of channel A0~B1 0 indicates low level input, 1 indicates high level input
00002	one	B0 input status	read-only	
00003	two	A1 input status	read-only	
00004	three	B1 input status	read-only	
00005	four	A0 input status	read-only	Inverted value of level state of channel A0~B1 1 indicates low level input, 0 indicates high level input
00006	five	B0 input status	read-only	
00007	six	A1 input status	read-only	
00008	seven	B1 input status	read-only	

Support function codes 03, 06, 16

Address 4X (PLC)	**Address (PC, DCS)**	**Data content**	**attribute**	**Data description**
40001~40002	0~1	Encoder 0 count	Read/ Write	Encoder AB phase counter **(counting mode 0)**
40003~40004	2-3	Encoder 1 count	Read/	The data is a signed long integer in

			Write	hexadecimal format. Negative numbers use two complements, Positive number (0x00000000~0x7FFFFFFF), Negative number (0xFFFFFFFF~0x80000001), The storage order is CDAB. The counting method is 4-fold frequency counting, and the data is 4 times of the actual pulse number. The counter is cleared to write 0 directly to the corresponding register, You can also write other values as needed.
40005~40006	4-5	Frequency of encoder 0	read-only	Pulse frequency of encoder **(counting mode 0)** The data is a 32-bit floating point number, and the storage order is CDAB. The data is the frequency calculated according to the actual number of pulses per second, not 4 times the frequency.
40007~40008	6~7	Encoder 1 frequency	read-only	
40009~40010	8~9	Encoder 0 actual engineering value	read-only	Actual engineering value of encoder **(counting mode 0)** The data is a 32-bit floating point number, and the storage order is CDAB. It is the value obtained by multiplying the encoder counter value by the pulse multiplication rate set on the web page
40011~40012	10-11	Actual engineering value of encoder 1	read-only	
40012~40014	12-13	Speed of encoder 0	read-only	Encoder speed **(counting mode 0)** The data is a 32-bit signed long integer, and the storage order is CDAB.The speed is converted according to the number of pulses per revolution set in the configuration page.
40015~40016	14-15	Speed of encoder 1	read-only	
Address 4X (PLC)	**Address (PC, DCS)**	**Data content**	**attribute**	**Data description**
40017~40018	16-17	Channel A0 count	Read/ Write	Channel A0~B1 counter **(counting mode 1)** The data is an unsigned long integer, The storage order is CDAB.Hexadecimal format, （ 0x00000000~0xFFFFFFFF ）， The counter is cleared to write 0 directly to the
40019~40020	18-19	Channel B0 count	Read/ Write	
40021~40022	20~21	Channel A1 count	Read/ Write	
40023~40024	22-23	Channel B1 count	Read/	

			Write	corresponding register, or other values can be written as required.
40025~40026	24-25	Frequency of channel A0	read-only	Pulse frequency of channel A0~B1, **(counting mode 1)** The data is a 32-bit floating point number, and the storage order is CDAB.
40027~40028	26-27	Frequency of channel B0	read-only	
40029~40030	28~29	Frequency of channel A1	read-only	
40031~40032	30~31	Frequency of channel B1	read-only	
40033~40034	32~33	Engineering value of channel A0	read-only	Actual engineering value of channel A0~B1 **(counting mode 1)** The data is a 32-bit floating point number, and the storage order is CDAB. The value is the pulse count value multiplied by the pulse magnification set on the web page.It is used for automatic calculation of flow or length.
40035~40036	34~35	Engineering value of channel B0	read-only	
40037~40038	36~37	Engineering value of channel A1	read-only	
40039~40040	38~39	Engineering value of channel B1	read-only	
40041~40042	40~41	Speed of channel A0	read-only	Speed of channel A0~B1 **(counting mode 1)** Long integer (0x00000000~0xFFFFFFFF), The storage order is CDAB, The speed is converted according to the pulse number set in the configuration page.
40043~40044	42~43	Speed of channel B0	read-only	
40045~40046	44~45	Speed of channel A1	read-only	
40047~40048	46~47	Speed of channel B1	read-only	
forty thousand and sixty-eight	sixty-seven	Count clear register	write	Unsigned integer, the default value is 0. Modify this register to clear the encoder counter or channel counter.After modification, the register will automatically return to 0. Write 10: set encoder 0 count value to 0, Write 11: set encoder 1 count value to 0, **Write 18: set all encoder count values to 0,** Write 20: set the channel A0 count value to 0, Write 21: set the channel B0 count value to 0, Write 22: set channel A1 count value to 0,

				Write 23: Set the channel B1 count value to 0, **Write 36: Set all channel count values to 0.** Writing other values is invalid.
forty thousand two hundred and eleven	two hundred and ten	Module name	read-only	High bit: 0x01 Low bit: 0x92

Modbus TCP communication example:

01 (0x01) read coil

In a remote device, use this function code to read the 1 to 2000 continuous status of the coil.The request PDU specifies the starting address, that is, the first coil address and coil number specified.Address the coil from zero.Therefore, addressing coils 1-16 are 0-15.

The coil in the response message is divided into a coil according to each bit in the data field.The indication status is 1=ON and 0=OFF.The first data is used as the LSB (least significant bit) of the byte, and the subsequent coil data is arranged in high order to form a byte of 8 bits.If the number of returned outputs is not a multiple of eight, zero will be used to fill the remaining bits in the last data byte (up to the high end of the byte).The number of bytes field indicates the complete number of bytes of data

Example of function code 01:

request			**response**		
Field Name		**Hexadecimal**	**Field Name**		**Hexadecimal**
MBAP header	Transmission ID	01	MBAP header	Transmission ID	01
		00			00
	Agreement Flag	00		Agreement Flag	00
		00			00
	length	00		length	00
		06			04
	Unit identifier	01		Unit identifier	01
Function code		01	Function code		01
Start address Hi		00	Number of bytes		01
Start address Lo		twenty	output data		00
Output quantity Hi		00			
Output quantity Lo		08			

03 (0x03) Read Holding Register

In a remote device, use this function code to read the contents of the continuous block of the holding register.The request PDU specifies the starting register address and the number of registers.Address registers from zero.Therefore, addressing registers 1-16 are 0-15.In the response message, each register has two bytes, the first byte is the data high bit, and the second byte is the data low bit.

Example of function code 03:

request			response		
Field Name		**Hexadecimal**	**Field Name**		**Hexadecimal**
MBAP header	Transmission ID	01	MBAP header	Transmission ID	01
		00			00
	Agreement Flag	00		Agreement Flag	00
		00			00
	length	00		length	00
		06			05
	Unit identifier	01		Unit identifier	01
Function code		03	Function code		03
Start address Hi		00	Number of bytes		02
Start address Lo		twenty	Register value Hi		00
Register No. Hi		00	Register value Lo		00
Register No. Lo		01			

05 (0x05) Write single coil

On a remote device, use this function code to write a single output as ON or OFF.The request PDU indicates the forced coil address.Address the coil from zero.Therefore, address coil address 1 is 0.The constant of the coil value field indicates the ON/OFF status of the request.The hexadecimal value 0xFF00 requests that the coil be ON. The hexadecimal value 0x0000 requests that the coil be OFF.All other values are illegal and have no effect on the coil.

The correct response is the same as the request.

Example of function code 05:

request			response		
Field Name		**Hexadecimal**	**Field Name**		**Hexadecimal**
MBAP header	Transmission ID	01	MBAP header	Transmission ID	01
		00			00
	Agreement Flag	00		Agreement Flag	00
		00			00
	length	00		length	00
		06			06
	Unit identifier	01		Unit identifier	01
Function code		05	Function code		05
Output address Hi		00	Output address Hi		00
Output address Lo		00	Output address Lo		00
Output value Hi		FF	Output value Hi		FF
Output value Lo		00	Output value Lo		00

06 (0x06) Write a single register

In a remote device, use this function code to write a single holding register.The request PDU indicates the address to be written to the register.Address registers from zero.Therefore, address register address 1 is 0.

The correct response is the same as the request.

Example of function code 06:

request			response		
Field Name		**Hexadecimal**	**Field Name**		**Hexadecimal**
MBAP header	Transmission ID	01	MBAP header	Transmission ID	01
		00			00
	Agreement Flag	00		Agreement Flag	00
		00			00
	length	00		length	00
		06			06
	Unit identifier	01		Unit identifier	01
Function code		06	Function code		06
Register address Hi		00	Register address Hi		00
Register address Lo		00	Register address Lo		00
Register value Hi		00	Register value Hi		00
Register value Lo		FF	Register value Lo		FF

15 (0x0F) Write multiple coils

On a remote device, use this function code to write multiple outputs as ON or OFF.The request PDU indicates the forced coil address.Address the coil from zero.Therefore, address coil address 1 is 0.The constant of the coil value field indicates the ON/OFF status of the request.The data is converted from hexadecimal system to binary system and arranged in bits. If the bit value is 1, the request coil is ON, and if the bit value is 0, the request coil is OFF.

Example of function code 15:

request			response		
Field Name		**Hexadecimal**	**Field Name**		**Hexadecimal**
MBAP header	Transmission ID	01	MBAP header	Transmission ID	01
		00			00
	Agreement Flag	00		Agreement Flag	00
		00			00
	length	00		length	00
		08			06
	Unit identifier	01		Unit identifier	01
Function code		0F	Function code		0F
Start address Hi		00	Start address Hi		00
Start address Lo		00	Start address Lo		00
Number of coils Hi		00	Number of coils Hi		00

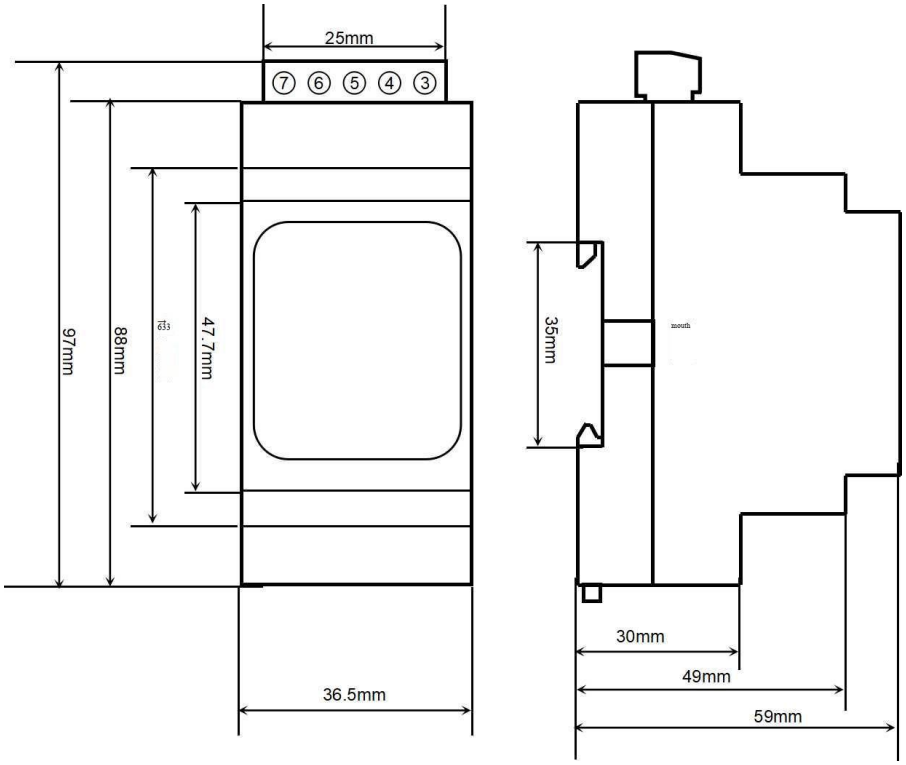
Number of coils Lo	02	Number of coils Lo	02
Number of bytes	01		
Output value	02		

16 (0x10) Write multiple registers

In a remote device, use this function code to write multiple holding registers.The request PDU indicates the address to be written to the register.Address registers from zero.Therefore, address register address 1 is 0.Example of function code 16:

request			**response**		
Field Name		**Hexadecimal**	**Field Name**		**Hexadecimal**
MBAP header	Transmission ID	01	MBAP header	Transmission ID	01
		00			00
	Agreement Flag	00		Agreement Flag	00
		00			00
	length	00		length	00
		0B			06
	Unit identifier	01		Unit identifier	01
Function code		ten	Function code		ten
Start register address Hi		00	Start register address Hi		00
Start register address Lo		00	Start register address Lo		00
Number of registers Hi		00	Number of registers Hi		00
Number of registers Lo		02	Number of registers Lo		02
Number of bytes		04			
Register value Hi		00			
Register value Lo		05			
Register value Hi		00			
Register value Lo		06			

Overall dimension: (unit: mm)



Can be installed on standard DIN35 guide rail

Warranty:

Within two years from the date of sale of this product, if the user complies with the storage, transportation and use requirements, but the product quality is lower than the technical indicators, the product can be returned to the factory for free maintenance.In case of damage due to violation of operating regulations and requirements, the device cost and maintenance cost shall be paid.

Copyright:

Trademark:

Other trademarks and copyrights mentioned in this manual belong to their respective owners.

Version No.: V1.0

Date: March 2026

This document was translated by AI, so there may be inappropriate descriptions. If you have any questions, please contact us via email: sales@wayjun.net