Wireless Modem

User Manual

NB183

Eight Serial Port Server User Manual

Content

Disclaimer

EBYTE reserves all rights to this document and the information contained herein. Products, names, logos and designs described herein may in whole or in part be subject to intellectual property rights. Reproduction, use, modification or disclosure to third parties of this document or any part thereof without the express permission of EBYTE is strictly prohibited.

The information contained herein is provided "as is" and EBYTE assumes no liability for the use of the information. No warranty, either express or implied, is given, including but not limited, with respect to the accuracy, correctness, reliability and fitness for a particular purpose of the information. This document may be revised by EBYTE at any time. For most recent documents, visit www.ebyte.com.

1. Overview

The serial server series products independently developed by Ebyte integrate the TCP/IP protocol stack, which can realize two-way transparent transmission of data from serial port to Ethernet port, Modbus protocol conversion, serial port device cloud and other functions. The industrial design ensures the stability of the equipment in the harsh working environment, and the rich indicator light effects are used to feedback the different working states of the equipment.

It supports the configuration of the host computer. With simple configuration, the networking and cloud access of serial devices can be easily realized.

Widely used in computer room monitoring, environmental monitoring, intelligent transportation, gate control, intelligent express cabinet and other industries.

1.1   Features

● Industrial design is stable and reliable, high-level port protection;

● Abundant LED status indicators to quickly locate the working status of the equipment;

● Support terminal or DC head power supply, 8-28V DC wide voltage input, support reverse connection protection;

● Support terminal interface (RS485) and standard DB9 interface (RS232);

● Baud rate supports 2400～115200bps, supports multiple verification methods;

● Support a variety of registration packets and heartbeat packet sending, such as connecting to send MAC, connecting to send custom, etc.;

● Support stable and reliable host computer and AT command configuration, independent configuration of eight channels does not affect each other;

● Support configurable domain name resolution service, namely DNS;

● Support DHCP to dynamically obtain IP, subnet mask, default gateway, DNS server address;

● Support multiple working modes TCP client, TCP server, UDP client, UDP server, MQTT client, HTTP client;

● Can open 8-way server mode at the same time, support 16 clients to dynamically allocate access, a single server supports 9 client access,

● Support the UDP server mode of memory mode, record the UDP address of the last communication, and use it for the destination address of the next communication;

● Support a variety of Modbus gateways, which can realize the active reporting of RTU devices, support the inter-conversion of Modbus TCP and Modbus RTU protocols, can be configured to automatically collect device data in storage mode, and can also use multi-host mode with one question and one answer;

● Support MQTT gateway function, quickly access Alibaba Cloud and standard MQTT3.1.1 servers (OneNET, Baidu Cloud, Huawei Cloud, etc.);

● Support Modbus data to actively report TCP transparent transmission server, MQTT server and other servers;

● Support HTTP client mode, use HTTP/1.1 protocol, can be configured as GET, POST two request methods

- Can use TCP/IP direct communication or connect communication through "virtual serial port";

- Rich independent LED status indicators, support status feedback such as link, network cable, data transmission and reception, and quickly locate problems;

- Support firmware upgrade or firmware replacement by the host computer through UDP or serial port;

- The isolated version supports 8-channel serial port independent hardware isolation(Supported only on NB183)

2. Quick Start

2.1 Hardware Preparation

One notebook with RJ45 network port;

One NB183/NB183S serial server;

One DC12V 1A power adapter;

a network cable;

One USB to RS485 serial cable;

The specific hardware equipment is prepared as shown in the following figure:

Computer	cable	NB183
DC 12V1A	USB to RS45	several wires

【Note】Only use the RS485 interface for demonstration, and use the corresponding USB converter for other interfaces;

2.2 Software Preparation

Serial debugging assistant (XCOM), network debugging assistant (TCP/IP debugging assistant), Ebyte network configuration tool (configuring the host computer), official website address: www.ebyte.com, product details provide download interface.

Network Assistant	XCOM	亿佰特网络配置工具
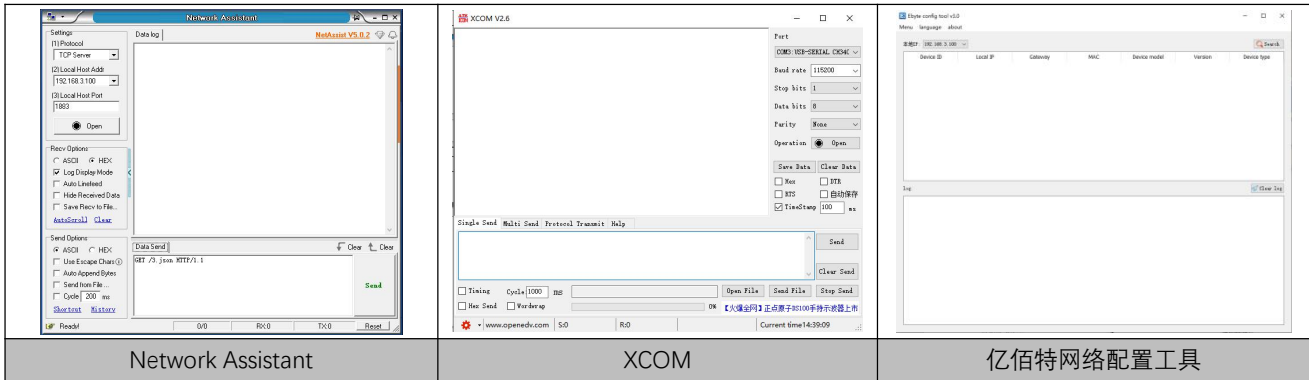
[Note] The host computer displayed in the manual may be different from the superior version provided on the official website. The host computer provided on the official website shall prevail.

2.3 Test steps of device default parameters

Different channels use the same IP and different local ports. For example, the factory-configured channel 1 corresponds to port 8001 and channel 8 corresponds to port 8008 in turn. For details, see the "Correspondence between Channels and Serial Ports" chapter.

2.3.1 Connecting the hardware

              1. Use a network cable to connect the serial port server network port and the computer network port;

2. Use a USB to RS485 serial cable to connect the USB port of the computer and the RS485 interface of the serial server;

3. Use the power adapter (DC 8-28V) to power on the device, and observe whether the indicator light is normal after power-on, refer to "Indicator Indicator";

4. After confirming that there is no problem with the status, proceed to the next configuration;

2.3.2 Device parameter configuration

In order to enable users to quickly have a simple understanding of the serial port server, we use the default parameters of the serial port server to conduct data transparent transmission test. The default parameters of the NB183 serial server device are shown in the table below.

Project	default parameters

IP address	192.168.3.7
subnet mask	255.255.255.0
gateway	192.168.3.1
The working mode of channel 1	TCP_SERVER
The working mode of channel 1	8001
Serial port baud rate	115200
Serial port parameters	NONE/8/1/NONE

Make sure that the IP of the computer and the IP of the serial port server are in the same network segment and cannot conflict. The inspection method is shown in the figure below.
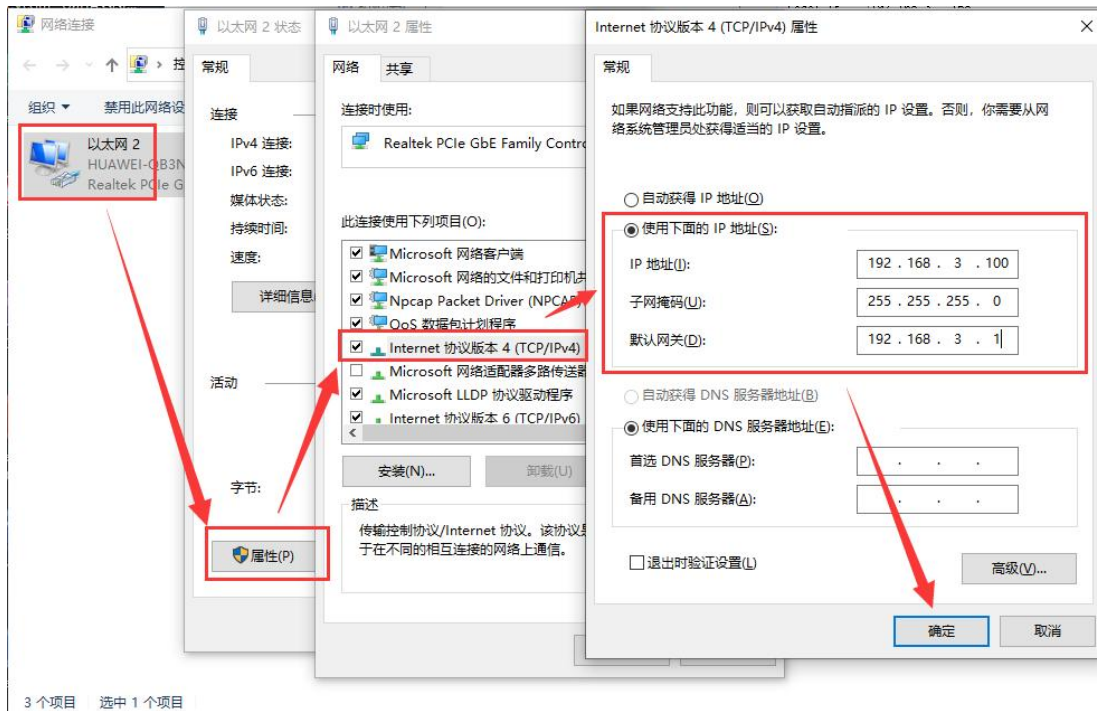
2.3.3 Turn off the computer firewall

If the communication is unsuccessful, the user can try to turn off the firewall of the computer and try again.
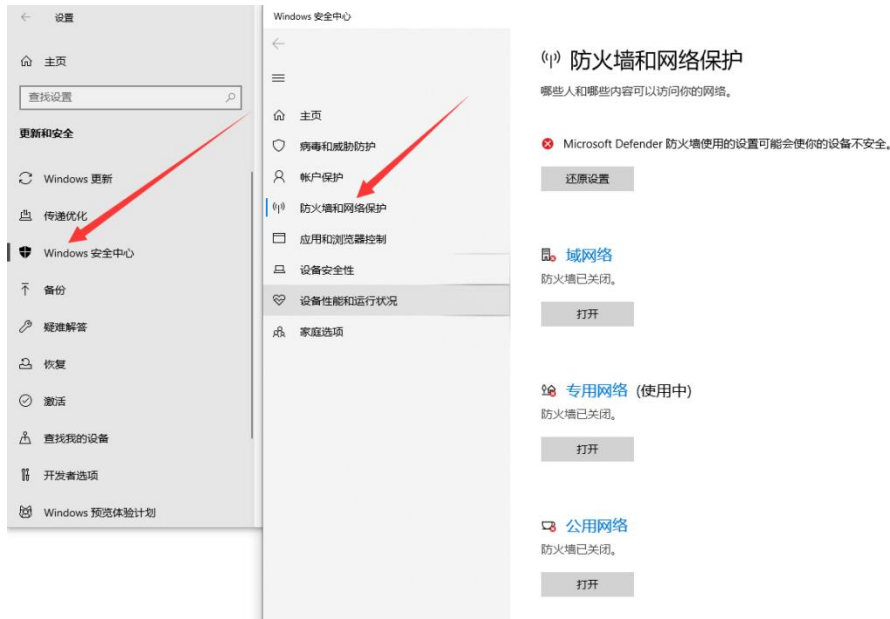
2.3.4 Open "Serial Debug Assistant"

Select the correct serial port number, configure the correct serial port parameters (115200-8N1), and open the serial port, as shown in the following figure:
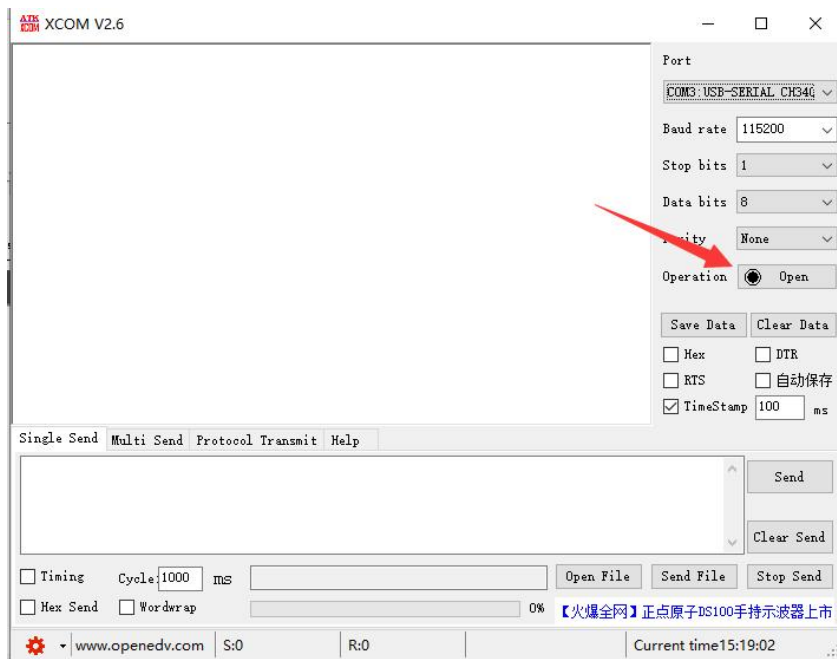
2.3.5 Open Network Assistant

Select "TCP Client" mode, configure the remote IP as "192.168.3.7", and configure the remote port as "8001" as shown in the following figure:

2.3.6 Send and receive data test

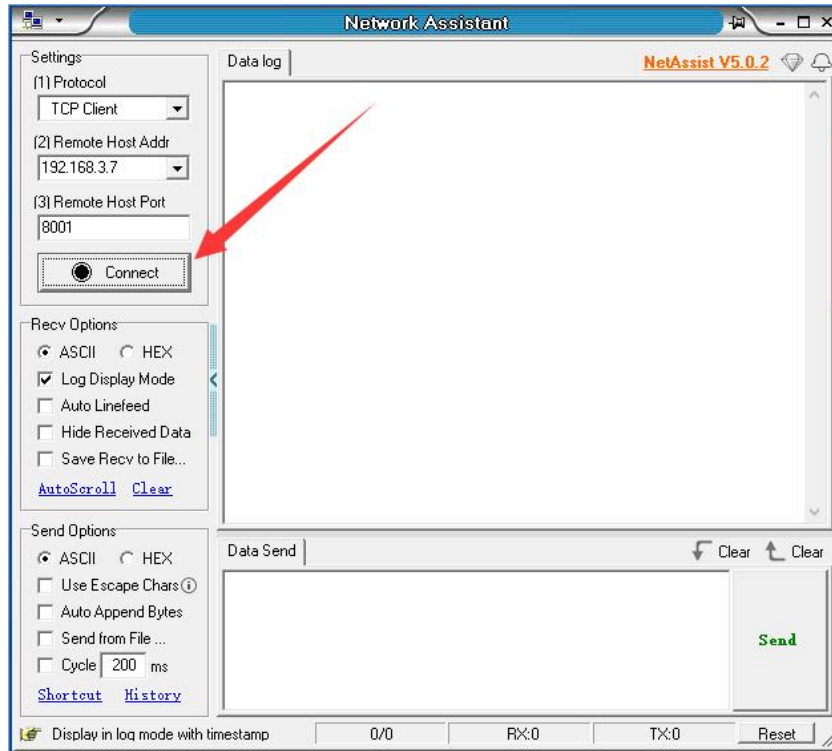Click the [Send] button on the "Network Debugging Assistant" and "Serial Debugging Assistant" respectively to realize the transparent transmission of network and serial data, as shown in the figure below.
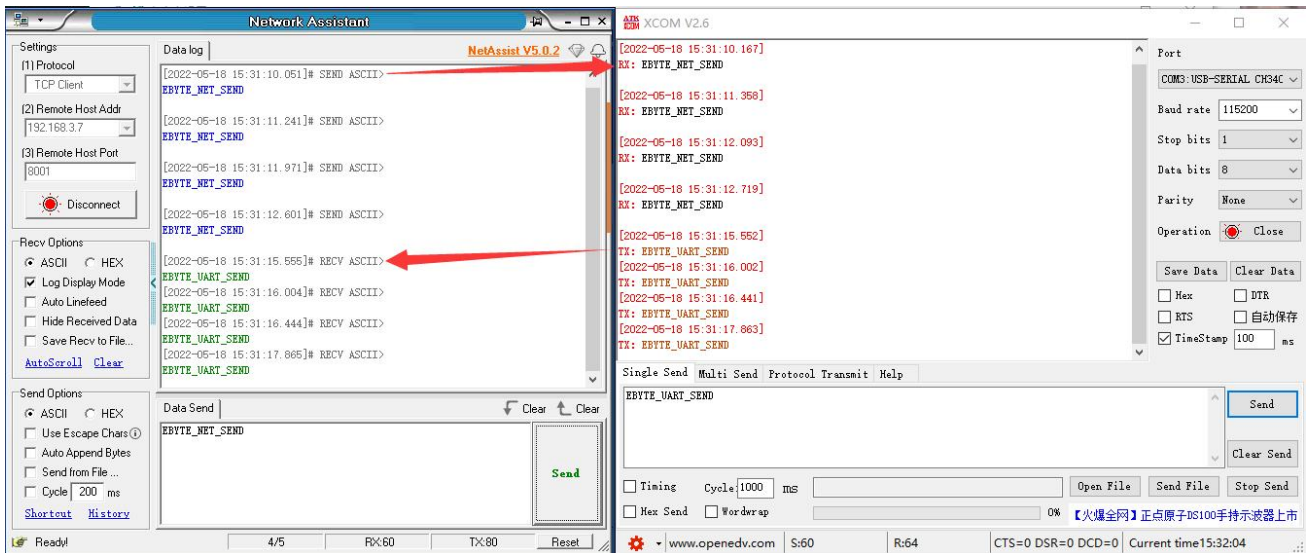
3. Product Overview

3.1 Technical parameters

project	interface	explanation
Power	crimp terminal	DC 8 ~ 28 V, can not input power at the same time;

|	DC female	Crimping terminal: 5.08mm*2 phoenix terminal; DC female head: in-line round hole, outer diameter 5.5mm, inner diameter 2.0mm;
network port	RJ45	10M
serial port	Channel 1 to Channel 4	Interface 1 (RS485, 3.81mm phoenix terminal, support isolation); Interface 2 (RS232, standard RS232 line sequence DB9 female);
	Channel 5 to Channel 8	RS485, 3.81mm Phoenix terminal, support isolation
Operating mode	TCP Server、TCP Client、UDP Server、UDP Client、HTTP Client、MQTT Client （default TCP Server）	
Network protocol	IP、TCP/UDP、IPv4、ICMP、APR、DHCP、DNS、HTTP、MQTT	
How to get IP	DHCP、Static IP (default static IP)	
DNS	support	
Configuration method	Configuration tools, AT commands	
IP address	Customizable (default, 192.168.3.7)	
local port	Customizable (default, channel 1 to channel 8: 8001-8008)	
subnet mask	Customizable (default, 255.255.255.0)	
gateway	Customizable （default，192.168.3.1）	
Serial port cache	512 Byte	
Serial port packaging mechanism	512 Byte	
Serial port baud rate	2400 ～ 115200 bps （default115200）	
data bits	8	
stop bit	1, 2 (default 1)	
Check Digit	None、Odd、Even （defaultNone）	
Flow Control	not support	
Installation method	Positioning hole installation	
Product Size	198mm * 109mm * 26.5mm （length*width*height）	
product weight	518g ± 5g	
Working temperature and humidity	-40 to +85℃, 5% to 95%RH (non-condensing)	
Storage temperature and humidity	-40 ～ +105℃, 5% ～ 95%RH (non-condensing)	

3.2 Interface description
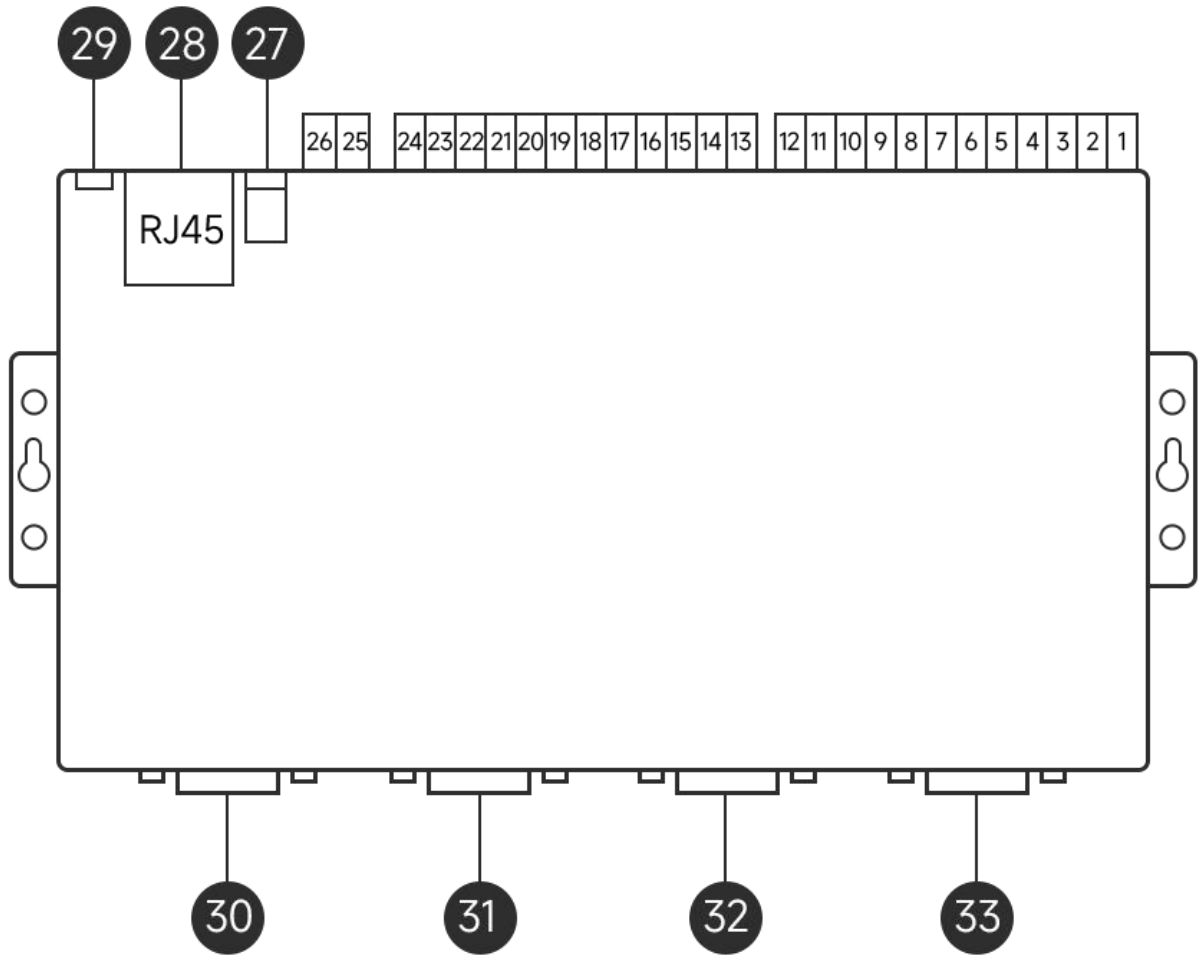
No	Name	Function	Explanation
1	RS485-1-A	channel1-RS485 InterfaceA	3.81mm Phoenix Terminal
2	RS485-1-B	channel1-RS485 InterfaceB	3.81mm Phoenix Terminal
3	RS485-1-G	channel1-RS485 InterfaceG	3.81mm Phoenix Terminal
4	RS485-2-A	channel2-RS485 InterfaceA	3.81mm Phoenix Terminal
5	RS485-2-B	channel2-RS485 InterfaceB	3.81mm Phoenix Terminal
6	RS485-2-G	channel2-RS485 InterfaceG	3.81mm Phoenix Terminal
7	RS485-3-A	channel3-RS485	3.81mm Phoenix Terminal

		InterfaceA	
8	RS485-3-B	channel3-RS485 InterfaceB	3.81mm Phoenix Terminal
9	RS485-3-G	channel3-RS485 InterfaceG	3.81mm Phoenix Terminal
10	RS485-4-A	channel4-RS485 InterfaceA	3.81mm Phoenix Terminal
11	RS485-4-B	channel4-RS485 InterfaceB	3.81mm Phoenix Terminal
12	RS485-4-G	channel4-RS485 InterfaceG	3.81mm Phoenix Terminal
13	RS485-5-A	channel5-RS485 InterfaceA	3.81mm Phoenix Terminal
14	RS485-5-B	channel5-RS485 InterfaceB	3.81mm Phoenix Terminal
15	RS485-5-G	channel5-RS485 InterfaceG	3.81mm Phoenix Terminal
16	RS485-6-A	channel6-RS485 InterfaceA	3.81mm Phoenix Terminal
17	RS485-6-B	channel6-RS485 InterfaceB	3.81mm Phoenix Terminal
18	RS485-6-G	channel6-RS485 InterfaceG	3.81mm Phoenix Terminal
19	RS485-7-A	channel7-RS485 InterfaceA	3.81mm Phoenix Terminal
20	RS485-7-B	channel7-RS485 InterfaceB	3.81mm Phoenix Terminal
21	RS485-7-G	channel7-RS485 InterfaceG	3.81mm Phoenix Terminal
22	RS485-8-A	channel8-RS485 InterfaceA	3.81mm Phoenix Terminal
23	RS485-8-B	channel8-RS485 InterfaceB	3.81mm Phoenix Terminal
24	RS485-8-G	channel8-RS485 InterfaceG	3.81mm Phoenix Terminal
25	-	Negative pole of DC 8-28 V	DC 8-28 V, 2*5.08mm phoenix terminal input; Do not supply power at the same time as the socket;
26	+	Positive pole of DC 8-28 V	DC 8-28 V, 2*5.08mm phoenix terminal input; Do not supply power at the same time as the socket;
27	DC-IN	DC power input	DC 8-28 V; In-line round hole, outer diameter 5.5mm, inner

|			diameter 2.5mm; Do not supply power with the terminals at the same time;
28	Ethernet	Ethernet Interface	Standard RJ45 Ethernet Interface, 10/100M adaptive
29	Reload	Factory reset button	After long press for 5s, NET is always on for 5s, and the device is reset to factory
30	RS232-4	channel4-RS232 Interface	Standard DB9 female, 2-TXD, 3-RXD, 5-GND
31	RS232-3	channel3-RS232 Interface	Standard DB9 female, 2-TXD, 3-RXD, 5-GND
32	RS232-2	channel2-RS232 Interface	Standard DB9 female, 2-TXD, 3-RXD, 5-GND
33	RS232-1	channel1-RS232 Interface	Standard DB9 female, 2-TXD, 3-RXD, 5-GND

3.3 Indicator light description
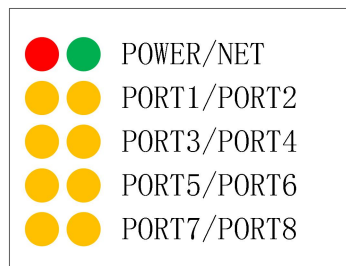
Label	Function	Explanation
POWER	Power Indicator	DC 8-28V power input
NET	Running lights	Unconnected network cable: 100ms on and 900ms off, flashing periodically; Connect the network cable: 1s is a period of flashing;
PORT1	Channel 1 Status Indicator	The link is not connected: the indicator light is off; The link connection is successful: the indicator light is always on; UDP mode: the indicator light is always on; Data sending and receiving indication: flashes when network or serial port sends and receives data;
PORT2	Channel 2 Status Indicator	The link is not connected: the indicator light is off; The link connection is successful: the indicator light is always on; UDP mode: the indicator light is always on; Data sending and receiving indication: flashes when network or serial port sends and receives data;
PORT3	Channel 3 Status Indicator	The link is not connected: the indicator light is off; The link connection is successful: the indicator light is always on;

		UDP mode: the indicator light is always on; Data sending and receiving indication: flashes when network or serial port sends and receives data;
PORT4	Channel 4 Status Indicator	The link is not connected: the indicator light is off; The link connection is successful: the indicator light is always on; UDP mode: the indicator light is always on; Data sending and receiving indication: flashes when network or serial port sends and receives data;
PORT5	Channel 5 Status Indicator	The link is not connected: the indicator light is off; The link connection is successful: the indicator light is always on; UDP mode: the indicator light is always on; Data sending and receiving indication: flashes when network or serial port sends and receives data;
PORT6	Channel 6 Status Indicator	The link is not connected: the indicator light is off; The link connection is successful: the indicator light is always on; UDP mode: the indicator light is always on; Data sending and receiving indication: flashes when network or serial port sends and receives data;
PORT7	Channel 7 Status Indicator	The link is not connected: the indicator light is off; The link connection is successful: the indicator light is always on; UDP mode: the indicator light is always on; Data sending and receiving indication: flashes when network or serial port sends and receives data;
PORT8	Channel 8 Status Indicator	The link is not connected: the indicator light is off; The link connection is successful: the indicator light is always on; UDP mode: the indicator light is always on; Data sending and receiving indication: flashes when network or serial port sends and receives data;

[Note] The status of some special working mode indicator lights:

1. Restore the factory, NET is always on for 5s, and the other indicators remain in the state until the NET goes out and then goes out;

2. Wait for the firmware to be burned, and the rest of the lights except the power indicator will flash with a cycle of 50ms, until the upgrade file starts to be transferred, or the firmware burning is exited;

3. During the firmware burning operation, the other indicators except the power indicator flash in a cycle of 500ms until the upgrade is completed;

3.4 Dimensions
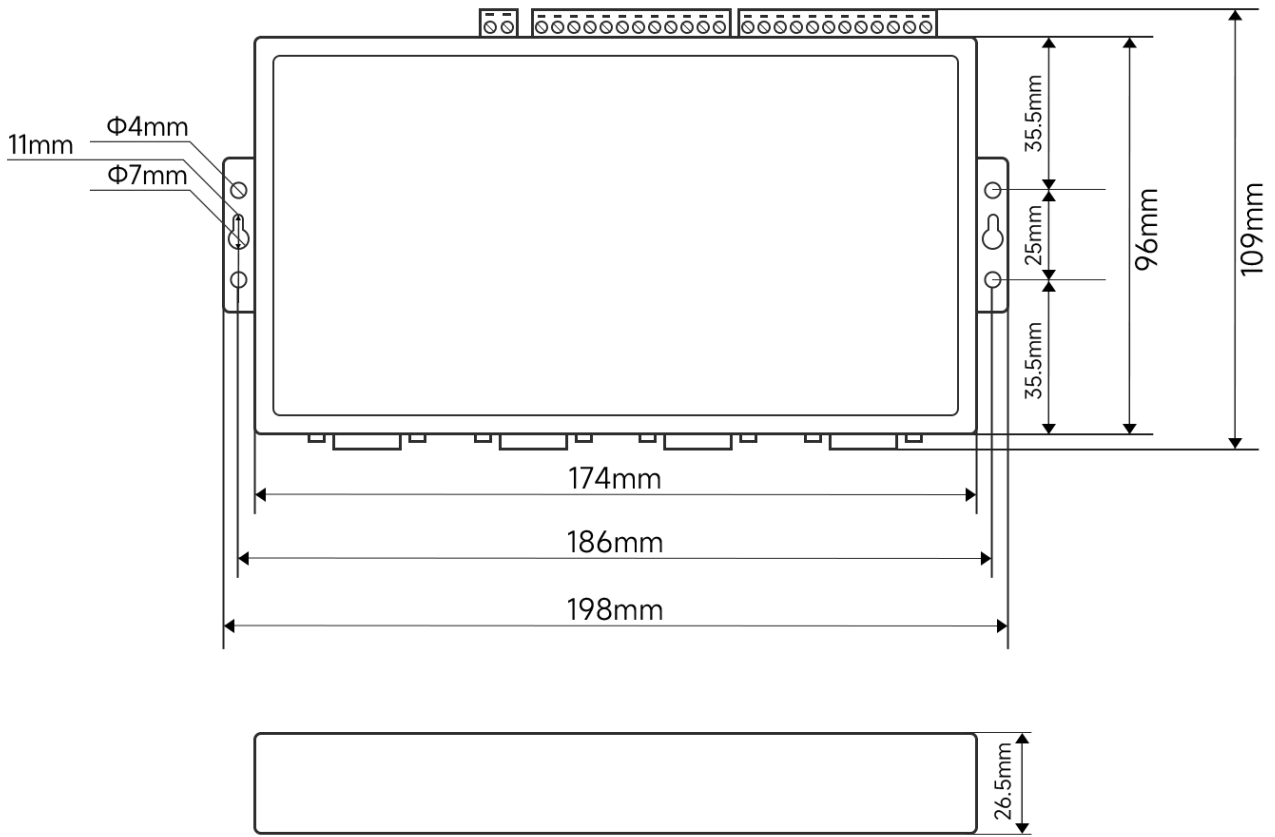
4. Basic Functions

4.1 Correspondence between channel and serial port

Baud rate: 2400, 4800, 9600, 14400, 19200, 38400, 57600, 76800, 115200bps;

Data bits: only 8 bits are supported;

Check digit: support no check (NONE), odd check (ODD), even check (EVEN);

Hardware flow control: not supported

Channel	Serial Port	pilot lamp	Serial protocol	Factory port, mode
Channel1	COM1	PORT1	RS232、RS485	8001、TCPS
Channel2	COM2	PORT2	RS232、RS485	8002、TCPS
Channel3	COM3	PORT3	RS232、RS485	8003、TCPS
Channel4	COM4	PORT4	RS232、RS485	8004、TCPS
Channel5	COM5	PORT5	RS485	8005、TCPS
Channel6	COM6	PORT6	RS485	8006、TCPS
Channel7	COM7	PORT7	RS485	8007、TCPS
Channel8	COM8	PORT8	RS485	8008、TCPS

4.2 Local network parameters

4.2.1 Local IP

STATIC (Static IP): Users can define configuration IP, subnet mask, default gateway, domain name resolution server (DNS server);

DHCP (Dynamic IP Acquisition): The device logs in to the server to automatically obtain the IP address, subnet mask, gateway address, and DNS server address parameters assigned by the server and configure and use

4.2.2 DNS (Domain Name Resolution)

When the user enters the domain name, the DNS server will be automatically queried, the database will be retrieved by the DNS server, and the corresponding IP address will be obtained. In the static IP mode, the user can customize the domain name resolution server, which is used to resolve the private domain name server data. In the dynamic IP mode The device automatically follows the domain name resolution server configured by the routing device. Users only need to modify the DNS server of the routing device without configuring the device.

4.2.3 Network disconnection and reconnection cycle

When the device detects that it is disconnected from the server, it periodically initiates a reconnection request. Therefore, the "disconnection reconnection time" does not affect the connection establishment time under normal conditions. The user can customize the configuration request period, and the default is 5s.

4.2.4 Timeout restart (no data restart)

The device monitors the data sending and receiving status. If the device does not send and receive data for a long time, the device automatically restarts to ensure the stability of long-term work. The default turn-on period of this function is 5

minutes, and the user can customize the turn-on or turn-off timeout restart period or the period of no data restart.

4.3 Hardware reset to factory

Press and hold the Reload pin of the device for 5s until the NET indicator no longer flashes, keep the NET indicator on for 5s, and the device is restored to the factory.

4.4 Device working mode

4.4.1 TCP Server

TCP Server is the TCP server. In TCP Server mode, the device listens to the local port, accepts the client's connection request and establishes a connection for data communication. When the Modbus gateway function is turned off, the device sends the data received by the serial port to all client devices that establish a connection with the device.

The number of clients that can be accessed by the server is dynamically adjusted. First, ensure that each channel of the eight channels can establish a complete communication link. In addition, the device also has 8 dynamic access communication links. For example, if the device turns on the eight-channel server mode, each server Two client devices can be connected, or if one server is enabled on the device, the server can connect to 9 client devices. If the number of client devices exceeds the number of accesses, the device will refuse the connection.

4.4.2 TCP Client

TCP Client is the TCP client. When the device is working, it will actively initiate a connection request to the server and establish a connection to realize the interaction between serial port data and server data.

To use the client, you need to configure the target IP address/domain name and target port accurately.

Eight channels can independently open eight TCP clients.

4.4.3 UDP Server

UDP Server means that the device does not verify the IP address of the data source when communicating with the UDP protocol. After each UDP data packet is received, it saves the source IP address and source port of the data packet, and sets it as the destination IP and port, so The data sent by the device only sends data packets to the source IP address and port where the device received data last time.

This mode is usually used in scenarios where multiple network devices communicate with this device, and the frequency is high, and the TCP Server cannot meet the conditions.

Using UDP Server requires the remote UDP device to send data first, otherwise the data cannot be sent normally.

[Note] In UDP mode, the data sent by the network to the device should be less than 512Bit per packet, otherwise it will cause data loss.

4.4.4 UDP Client

UDP Client is a connectionless transmission protocol that provides transaction-oriented simple and unreliable information transmission services. There is no connection establishment and disconnection, and data can be sent to the other party only by configuring the destination IP and destination port. It is usually used in data transmission scenarios where there is no requirement for the packet loss rate, the data packets are small and the transmission frequency is fast, and the data is to be transmitted to the specified IP.

In UDP Client mode, the device will only communicate with the configured (target IP and target port) remote UDP

devices.

In this mode, the target address is set to 255.255.255.255, and the sent data will be broadcast on the entire network segment, but the transceiver device needs to ensure that the ports are consistent, and the device can also receive broadcast data.

[Note] In UDP mode, the data sent by the network to the device should be less than 512Bit per packet, otherwise it will cause data loss.

4.4.5 HTTP Client

This mode can realize the function of HTTP automatic packet grouping. It provides two methods: GET and POST. Customers can configure URL, Header and other parameters by themselves, and the device will send packets to achieve fast communication between serial data and HTTP server. URL and Header are the most important. It supports 128 bytes of data for a long time, and the eight channels can independently open the HTTP client mode without affecting each other.

The HTTP request data should be less than the packet length (512 bytes), otherwise the device will divide the request data into multiple packets for request, resulting in abnormal request.
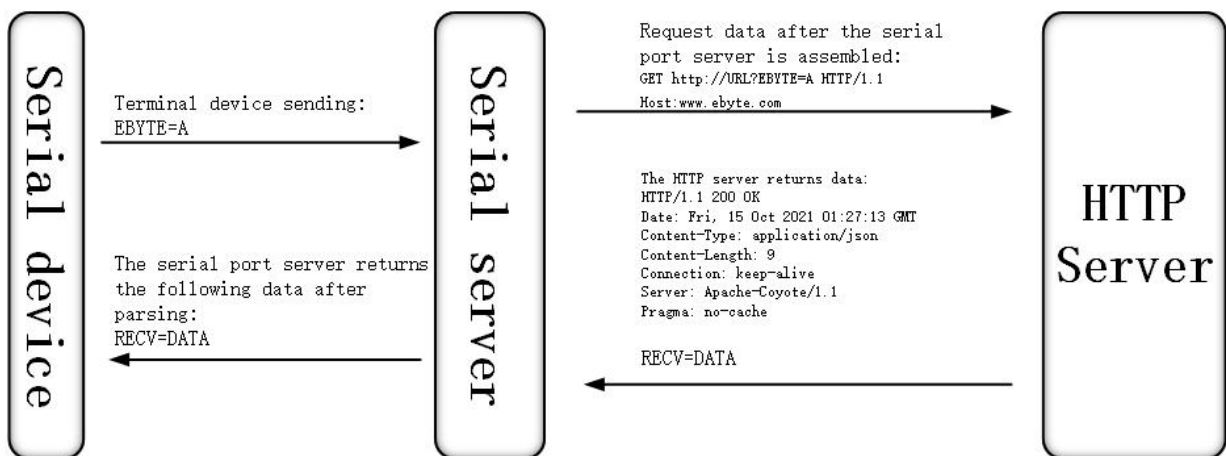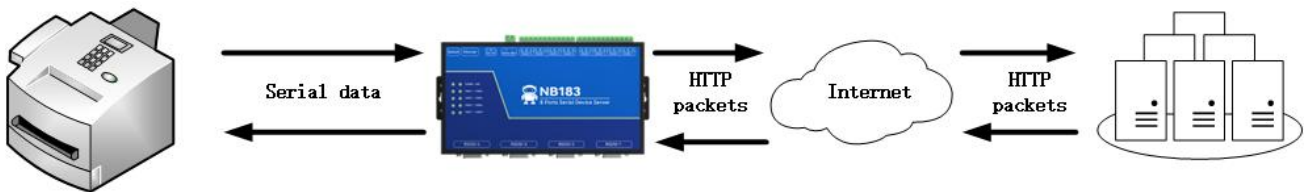
Supports configuring whether to return HTTP protocol headers. The returned data is shown in the following figure:

```
Returns the HTTP header  ➡  HTTP/1.1 200 OK
                             Date: Fri, 15 Oct 2021 01:27:13 GMT
                             Content-Type: application/json
                             Content-Length: 9
                             Connection: keep-alive
                             Server: Apache-Coyote/1.1
                             Pragma: no-cache

                             RECV=DATA


Does not return HTTP headers  ➡  RECV=DATA
```

Configuration instructions, open the host computer, search for the device and enter the device configuration interface, first configure the "network parameters", it is recommended to use the DHCP function to avoid incorrect configuration leading to abnormal device IP (network segment error, IP conflict, etc.), you need to use the HTTP function in the configuration It supports eight channels and configures HTTP client mode at the same time. Here is an example of requesting a "Baidu" webpage through GET (URL: empty, HEADER: Host: www.baidu.com, target domain name: www.baidu.com, target Port: 80, the local port is recommended to use a random port), the specific configuration is shown in the following figure:
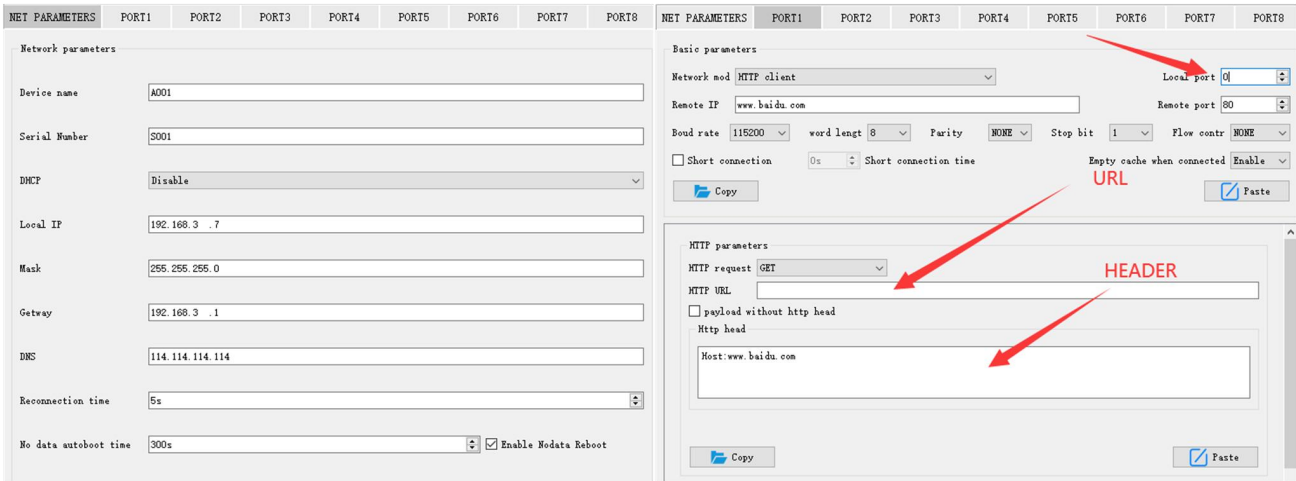
The request data is "/", use the serial port assistant to get the webpage:
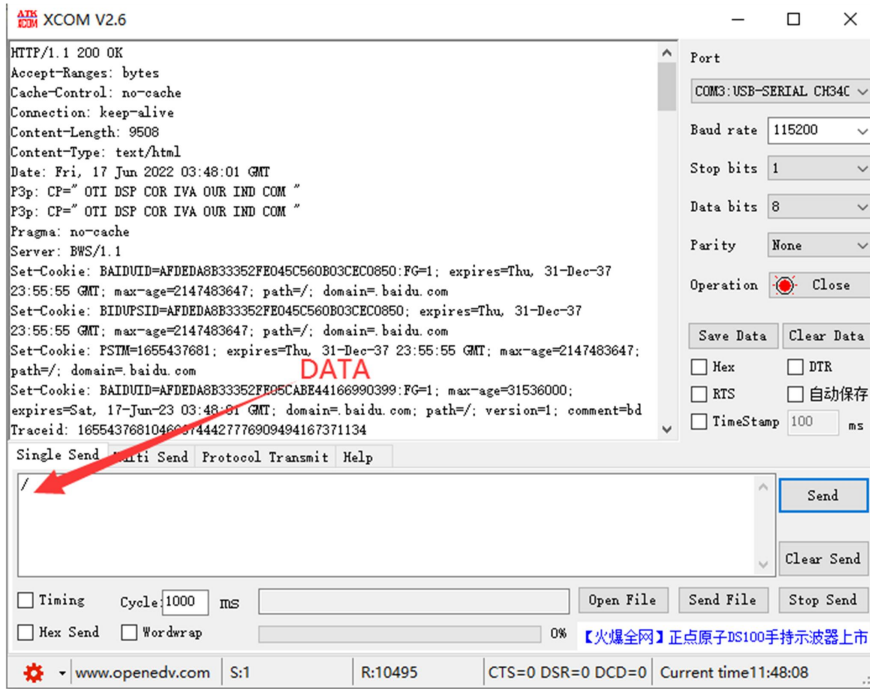
POST description, the header data that is configured as POST request method does not need to configure the data length separately (for example: Content-length: 2729), the device automatically calculates the data length and sends it in groups. Other header data needs to be manually configured, and supports up to 128 characters. Section data configuration.

4.4.6 MQTT Client

Supports quick access to standard MQTT3.1.1 protocol servers (OneNET, Baidu Cloud, Huawei Cloud, user-built and other server types) and Alibaba Cloud servers, supports QoS level configuration (QoS 0, QoS 1), supports super-long text configuration, Convenient and better access to network service operators (server address, three elements, subscription and publishing addresses support up to 128 characters, and Alibaba Cloud product key is 64 characters).
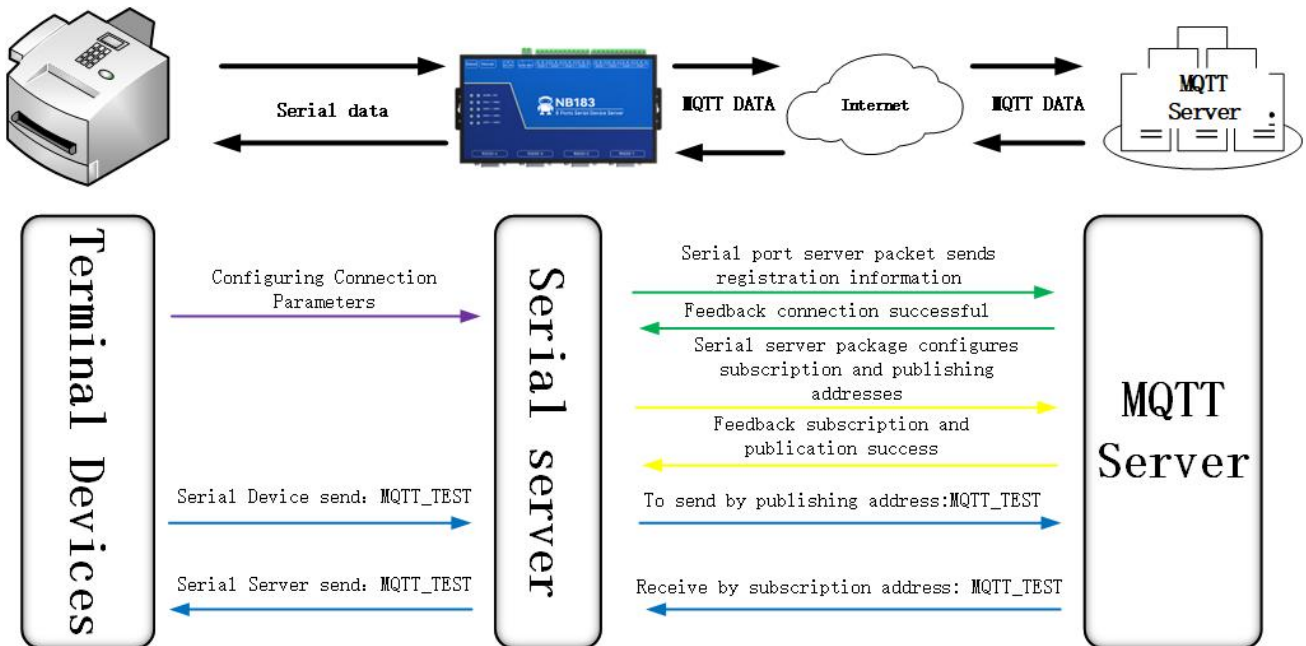
When using the MQTT function, the short link should be closed, otherwise the device will repeatedly connect to the server. It is recommended to use a random port, as shown in the following figure:
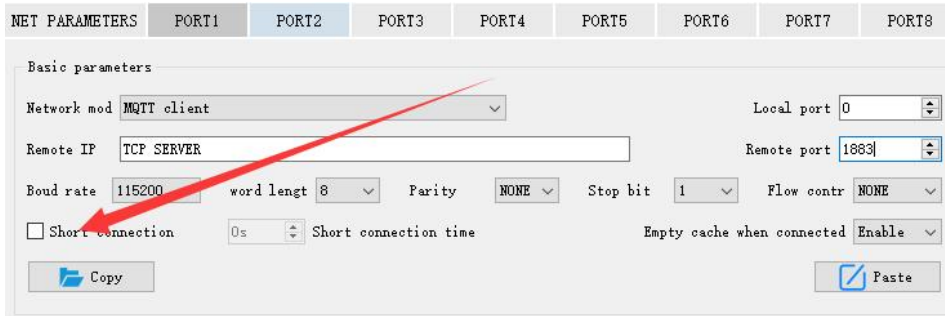
(1) , select the standard MQTT3.1.1, Baidu Cloud, OneNET, Huawei Cloud configuration can refer to the following table to fill in the parameters:

Parameter	Standard MQTT3.3.1	Baidu cloud	OneNET
Equipment name (Client ID)	Client ID	DeviceKey	Equipment ID
username (Device name)	User Name	IoTCoreId/DeviceKey	Product ID
password (Device secret)	Password	DeviceSecret	Equipment name/User Password
PrductKey	Alibaba Cloud parameters, optional		
Post topic	MQTT publish topic address (dynamically generated by OneNET)		
Subscribe to topics	MQTT subscription topic address (dynamically generated by OneNET)		

【Note】

Dynamically generated topic addresses can use the same parameters to achieve the effect of data return, for example: OneNET publishes and subscribes to the same topic address: 123456, which can achieve data return;

Due to the adjustment of the MQTT platform (Baidu Cloud, Huawei Cloud, OneNET), the connection cannot be made after filling in the parameters, and the platform rules shall prevail;

Take the standard MQTT3.1.1 parameter filling as an example, as shown below:
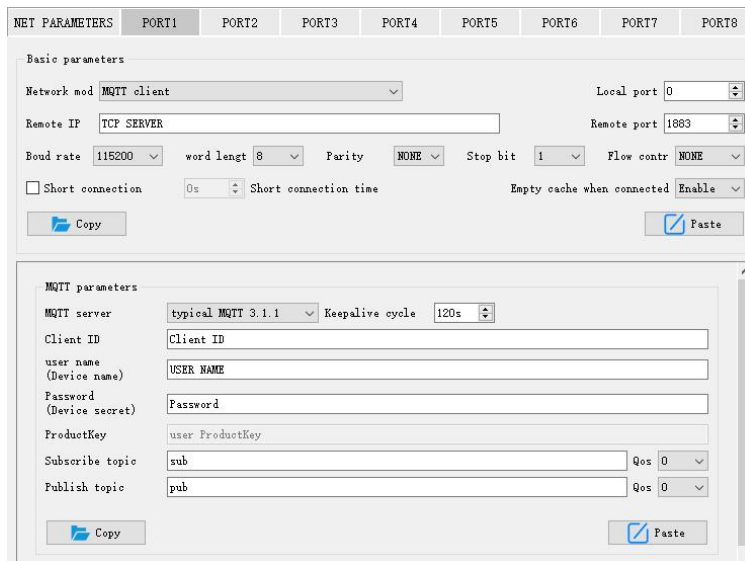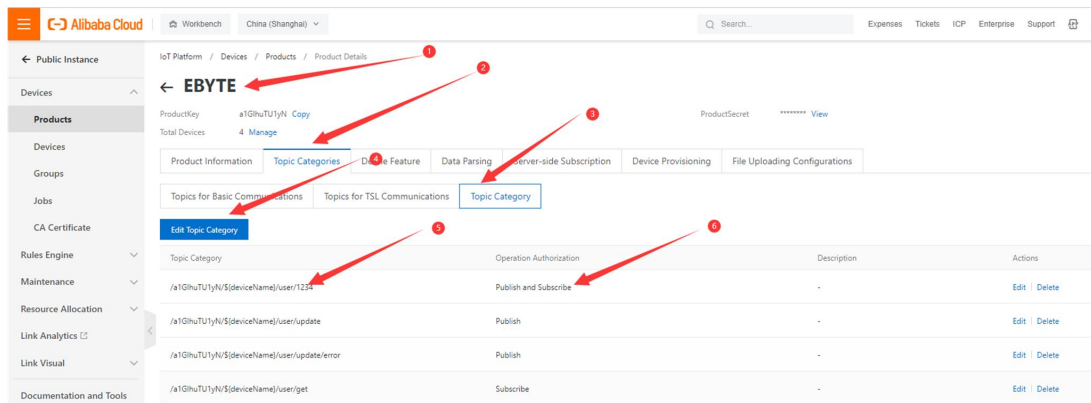
(2) ,Alibaba Cloud

Support the use of Alibaba Cloud's "Three Elements" to directly connect to the server to obtain the "Three Elements" required to connect to Alibaba Cloud, as shown in the figure (only for demonstration cases, users need to use self-built parameters to connect):
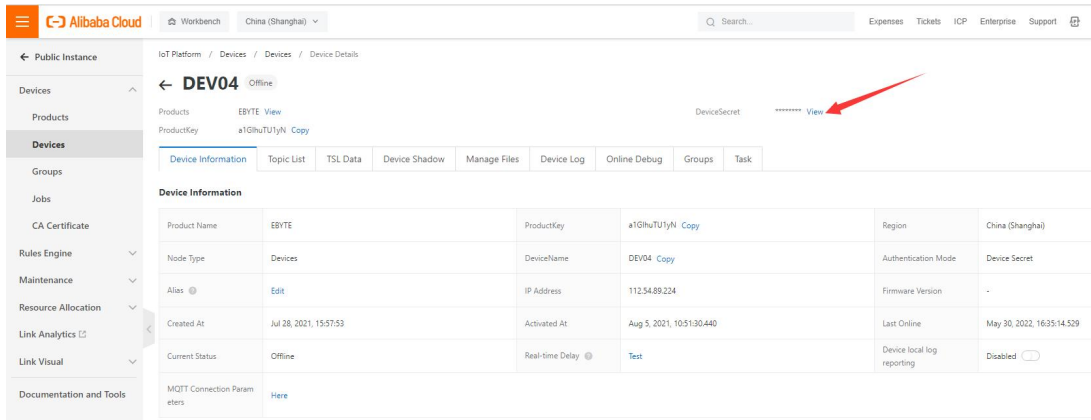
To configure a topic for communication testing:

Configuration theme description:

Select the corresponding product, go to "Custom Topic" under the Topic class list (for details, please refer to the Alibaba Cloud documentation), click "Define Topic Class", set the name to 1234, and grant publish and subscribe permissions (for data return pass).

Configure the device connection parameters, as shown in the following figure (the picture on the left is the host computer, and the picture on the right is the web page configuration):

```
{
    "ProductKey": "a1GlhuTU1yN",
    "DeviceName": "DEV04",
    "DeviceSecret": "xxxxxxxxxxxxxxxxxxxxxxxxxxxxx"
}
```

Alibaba Cloud server address: ProductKey.iot-as-mqtt.cn-shanghai.aliyuncs.com:1883

Topic for subscription and publication: /a1GlhuTU1yN/DEV04/user/1234

Alibaba Cloud MQTT platform communication test:
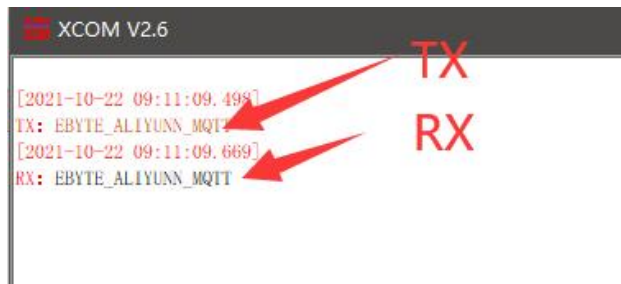
4.5 Channel ports

random port:

The TCP client, UDP client, HTTP client, and MQTT client can configure the local port to 0 (use a random local port), and the server mode cannot use a random port, otherwise the client cannot establish a connection correctly (the device does not correctly port listening).

Using a random port connection can quickly re-establish the connection when the device disconnects from the server unexpectedly, preventing the server from rejecting the connection due to four waves of incompleteness. It is recommended to use a random port in client mode.

When the device configures the TCP client, HTTP client, and MQTT client mode in AT, it will automatically configure a random port, which can be canceled by definition.

static port:

Device fixed port (factory default use: 8001-8008), TCP server mode device listens to the configured port, accepts client connection request and establishes a connection for data communication, TCP client mode device fixed port initiates connection request.

5. Advanced Functions

5.1 Heartbeat package and registration package

5.1.1 Heartbeat Packet

In client mode, users can choose to send heartbeat packets and set the time of heartbeat packets by themselves. The heartbeat packet can be selected in two modes: network heartbeat packet and serial port heartbeat packet. It supports hexadecimal and ASCII transmission. This heartbeat packet is not MQTT heartbeat. It needs to be turned off when using MQTT client mode. MQTT heartbeat can only be set in the "MQTT parameter configuration" column. Configure "Heartbeat Period" below. The content of the MQTT heartbeat packet is restricted by the protocol and is not open for configuration. It is recommended not to configure less than 60s. For example, 120s is recommended in the Alibaba Cloud manual.

Heartbeat packet sending mode:

1. The default is to turn off the heartbeat packet mode.

2. Serial port mode -> The device sends heartbeat content to the serial port bus according to the set heartbeat interval.

3. Network port mode -> The device sends heartbeat content to the network port bus according to the set heartbeat interval.

Customize heartbeat packet content (maximum support 40 bytes (ASCII) data, 20 bytes (HEX) data)

Customize the heartbeat packet sending interval. When it is set to 0, the heartbeat packet function is turned off. If the setting value is greater than zero, the heartbeat packet function is turned on. When it is turned on, the range can be set: (1-65536) seconds.

5.1.2 Registration Package

In the client mode, the user can choose to send the registration package, and set the registration package time by definition.

The registration package supports the following modes:

1. The MAC address (OLMAC) is sent when the network establishes a connection with the device.

2. The data of the custom registration package (OLCSTM) is sent when the network establishes a connection with the device.

3. After the network and the device are connected, each packet of data sent by the device to the network is preceded by a MAC address (EMBMAC).

4. After the connection is established between the network and the device, each packet of data sent by the device to the network is preceded by custom registration packet data (EMBCSTM).

Custom registration package content (maximum support 40 bytes (ASCII) data, 20 bytes (HEX) data)

[Note] It is recommended not to use special characters (such as "," "\" "/", etc.) when configuring the registration package. If you want to use it, it is recommended to use hexadecimal configuration.

5.2 Short connection

In client mode, short network connection is supported (this function is disabled by default). TCP short connection is mainly used to save server resource overhead, and is generally used in multi-point (multi-client)-to-point (server) scenarios.

The TCP short connection function is applied in the TCP Client mode. After the short connection function is turned on, it only requests to connect with the server when sending information. The device will automatically disconnect.

When the short link hold time is set to 0, the short link function is turned off. When the setting range is (2-255) seconds, the short link function is turned on, and the default hold time is 0 seconds (short link is turned off).

5.3 Serial port cache cleaning

When the TCP connection is not established, the data received by the serial port will be placed in the buffer area. The serial port receiving buffer is 512 bytes. After the network connection is successful, you can choose to clear the serial port buffer or send the buffer through the network through configuration.

Enable: The device does not save the data received by the serial port before the connection is established.

Disabled: The network will receive the serial buffered data after the connection is established.

5.4 Modbus Gateway

5.4.1 Protocol Conversion
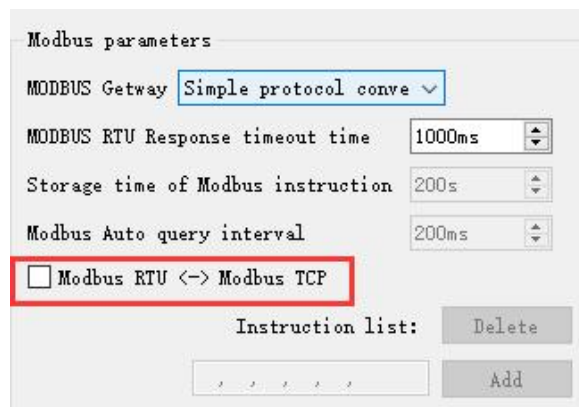
Enabled: Modbus data is checked and non-Modbus data (RTU/TCP) is discarded and not transmitted, and the Modbus RTU protocol and the Modbus TCP protocol are interconverted.

Disabled: Do not perform protocol conversion but verify Modbus data, discard non-Modbus data (RTU/TCP) and not transmit.

5.4.2 Simple Protocol Conversion

Convert Modbus RTU data to Modbus TCP data, or convert Modbus TCP data to Modbus RTU data, to realize the mutual conversion of Ethernet Modbus data and serial port Modbus data.
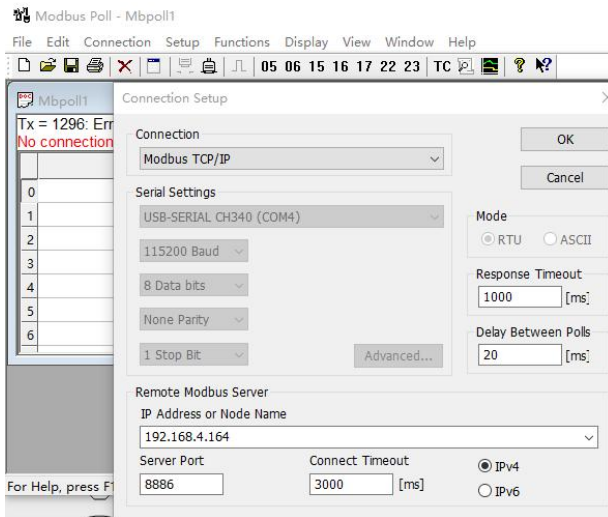
Simple protocol conversion can work in any mode (TCP client, TCP server, UDP client, UDP server, MQTT client). Host Mode".

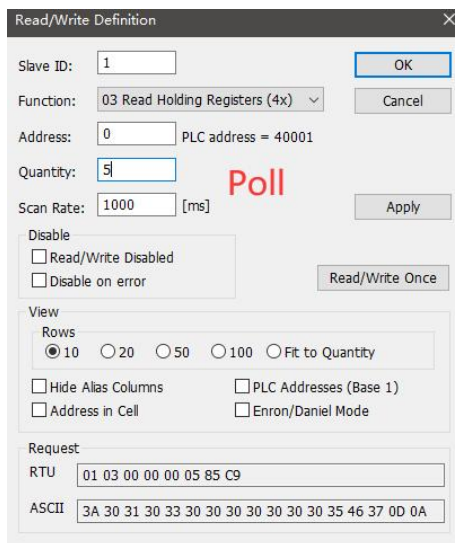Simple protocol conversion configuration:
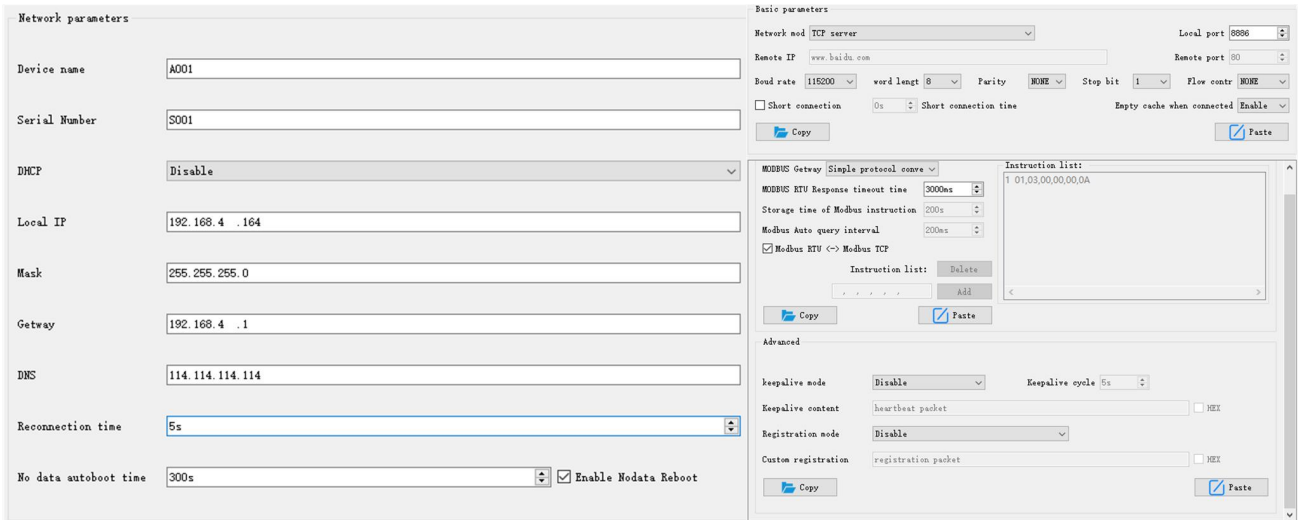
Modbus Poll and Modbus Slave software debugging:
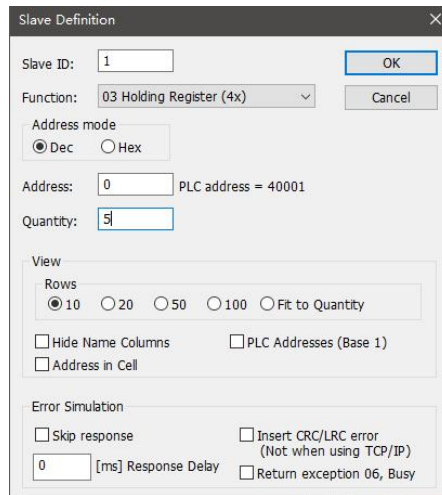
Software connection settings:

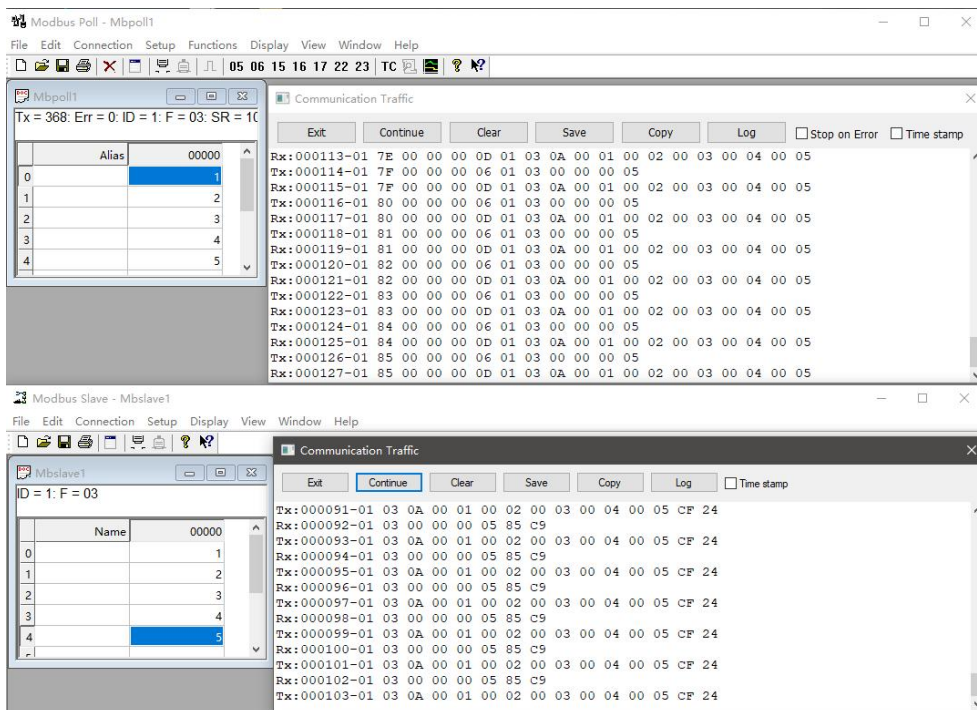Software register read and emulation configuration:

Poll menu select SetupRead/Write Definition



Slave menu select Setup Slave Definition

Communication demo:

5.4.3 Multi-host mode

Relatively simple protocol conversion, only one Modbus master station can exist, while the multi-master mode can access multiple Modbus TCP masters at the same time. When multiple Modbus masters access at the same time, the Modbus gateway will perform bus occupancy scheduling (RS-485 bus Only one request can be processed at a time, while the multi-host mode will be sorted according to the TCP request, and other links will wait), so as to solve the bus conflict problem (currently, in the single-server mode, up to 9 Modbus TCP host connections are supported, and the multi- When connecting hosts at the same time, attention should be paid to the matching of the request interval and the timeout time. Otherwise, the serial port transmission rate is much lower than the Ethernet transmission rate, resulting in packet loss. TCP server mode, and the slave can only be in the serial port, otherwise it will not work properly.

With the increase of the number of hosts, the Modbus timeout time should be increased accordingly. If multiple hosts are required to make continuous high-speed requests, it is recommended to use a "storage gateway", and it is recommended to configure "simple protocol conversion" when no multiple hosts are used.
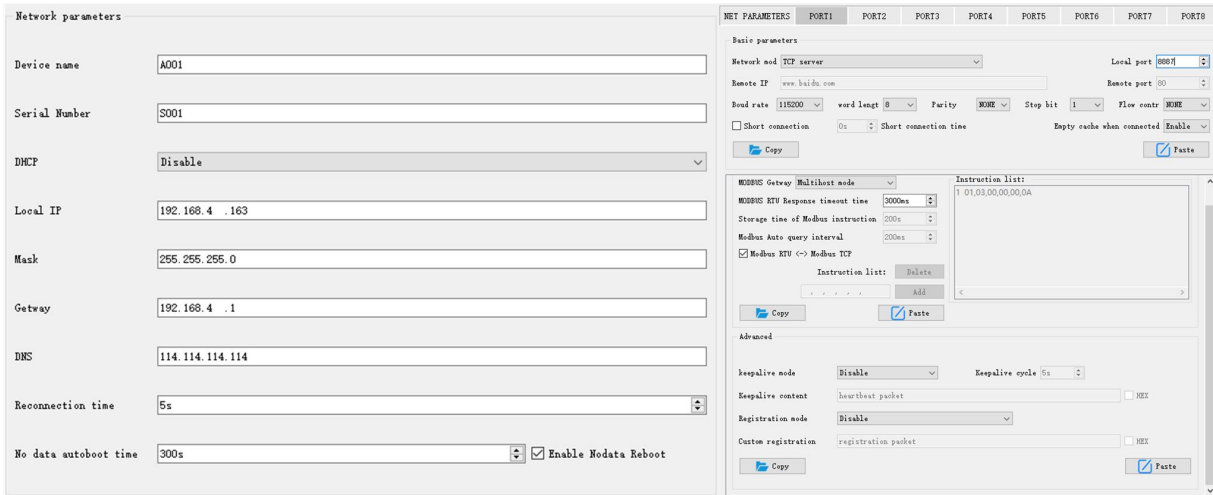
Refer to "Simple Protocol Conversion" for software configuration and register configuration, and open multiple Modbus Poll software at the same time (for example, 3 channels, up to 9 channels can be supported in the single server model).
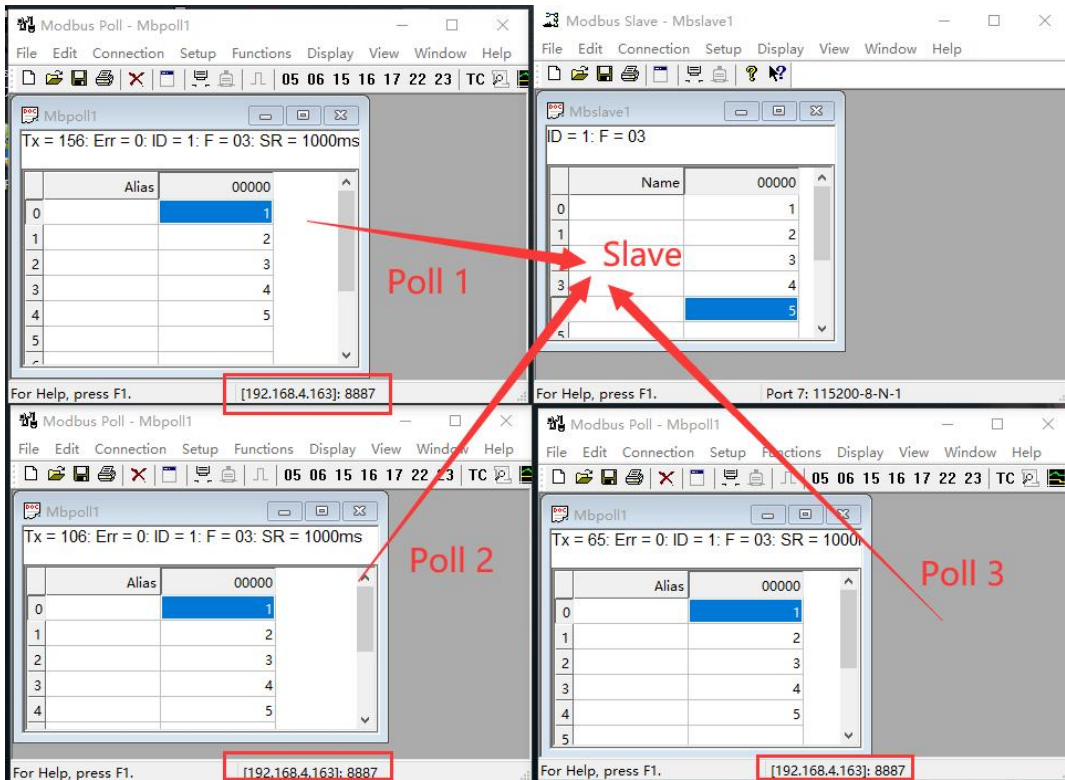
5.4.4 Storage Gateway

The storage gateway not only arbitrates the bus data but also stores the repeated read commands. When different hosts request the same data, the gateway does not need to query the register status of the RTU device multiple times, but directly returns the data cached in the storage area. To a certain extent, the multi-host request processing capability of the gateway is improved, and the time consumed by the entire request process is also shortened. Users can customize the storage area command polling interval and command storage time according to their needs.
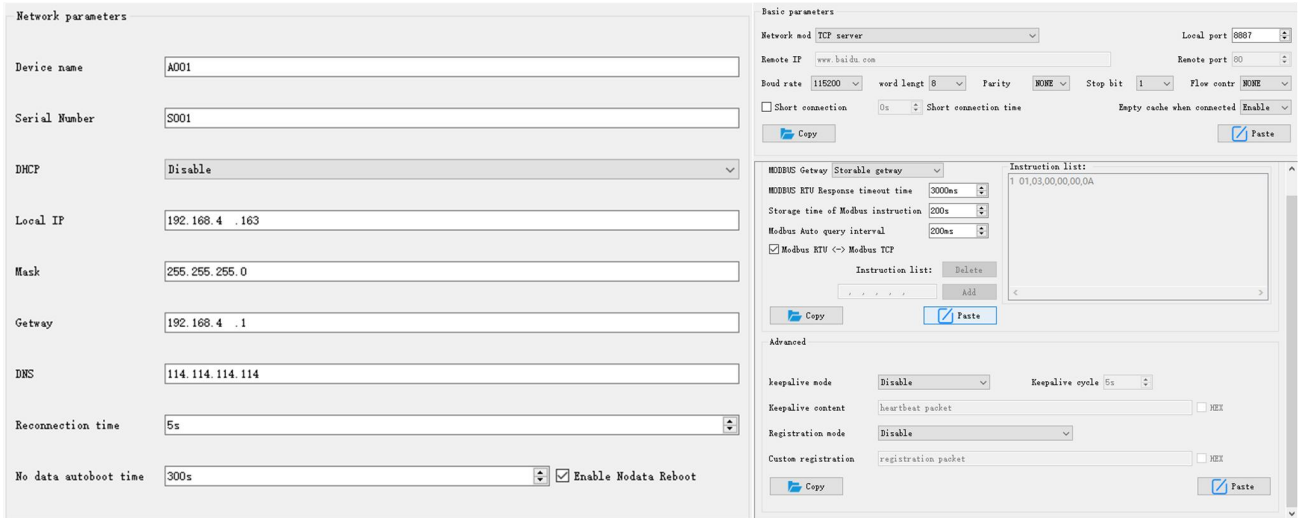
As the optimization of multi-host request performance, the storage gateway can only work in the TCP server mode, which improves the response speed of the network side.

Features:

(1) The gateway has a 2K cache for storing instructions and returning results (reading 10 holding registers as an example, about 67 instructions and returning results can be stored);

(2) The RTU response time-out automatically clears the cache to ensure the real-time and authenticity of the data;

(3) The polling interval can be customized, 0-65535ms (default: 200ms);

(4) The gateway will poll the RTU device according to the storage time of the instruction used for configuration. If the MODBUS host does not query the instruction again during the storage time, the gateway will automatically delete the storage instruction to release the cache;

(5) The first command and control command (05, 06, 0F, 10 function codes) will directly access the RTU device;

(6) Only 01, 02, 03, 04 Modbus function code query result storage is supported;

5.4.5 Configurable Gateway

The gateway automatically polls the RTU device registers according to the pre-configured MODBUS commands (only supports the configuration of the MODBUS read command). The commands in the non-storage table will directly operate the RTU device, and the frequently read commands can be stored in the gateway in advance, which can shorten the Response time (instruction to query configuration), data will not be sent directly to the server and will only be returned after a Modbus host request. Similar to the usage of "Simple Protocol Conversion", if you need to automatically upload data to the server, please select "Auto Upload". Due to the above features, the serial port side of the configurable gateway can only be connected to Modbus slaves.
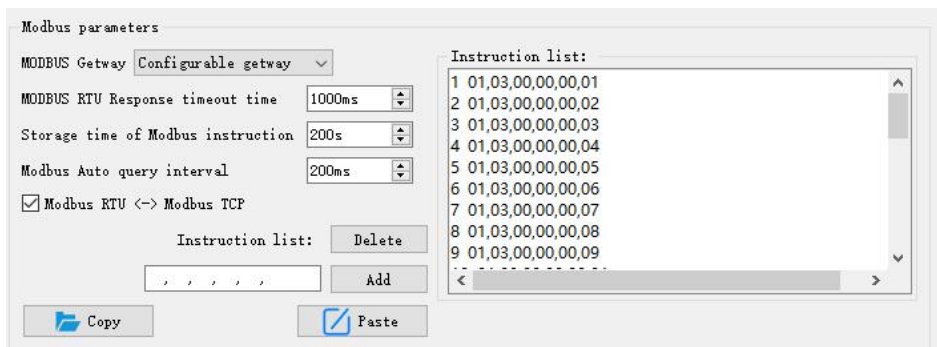
Instruction storage description (increase, instruction error and format error cannot be added):
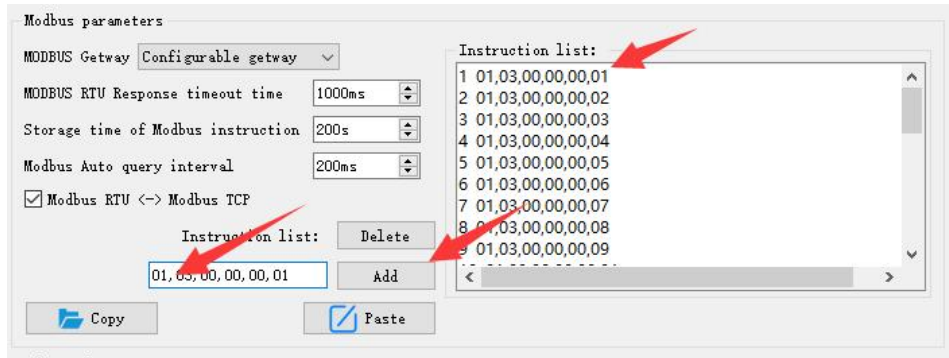
Instruction storage description (deleted):
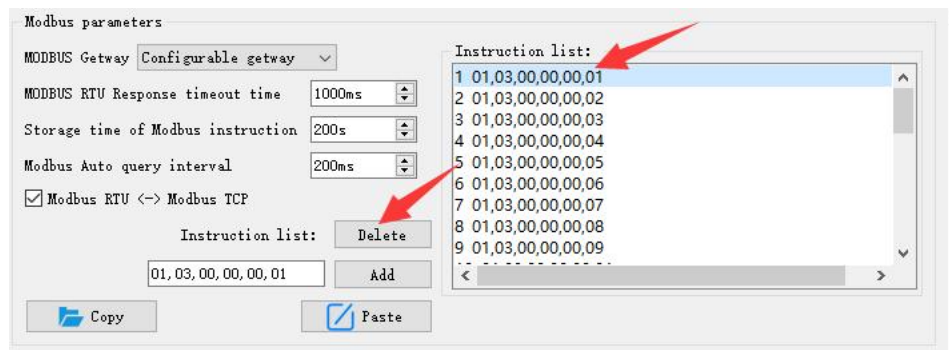

5.4.6 Automatic upload

In client mode (TCP client, UDP client, MQTT client, HTTP client), the gateway will automatically poll the instructions in the stored instruction table and upload it to the server, and the feedback format (Modbus RTU format or Modbus TCP format) can be selected according to the requirements.) and the command polling interval (0-65535ms).

For instruction pre-storage, refer to "Configurable Gateway - Instruction Storage Instructions", and automatically upload the upper computer/web page configuration:

TCP client demo (Modbus RTU format):
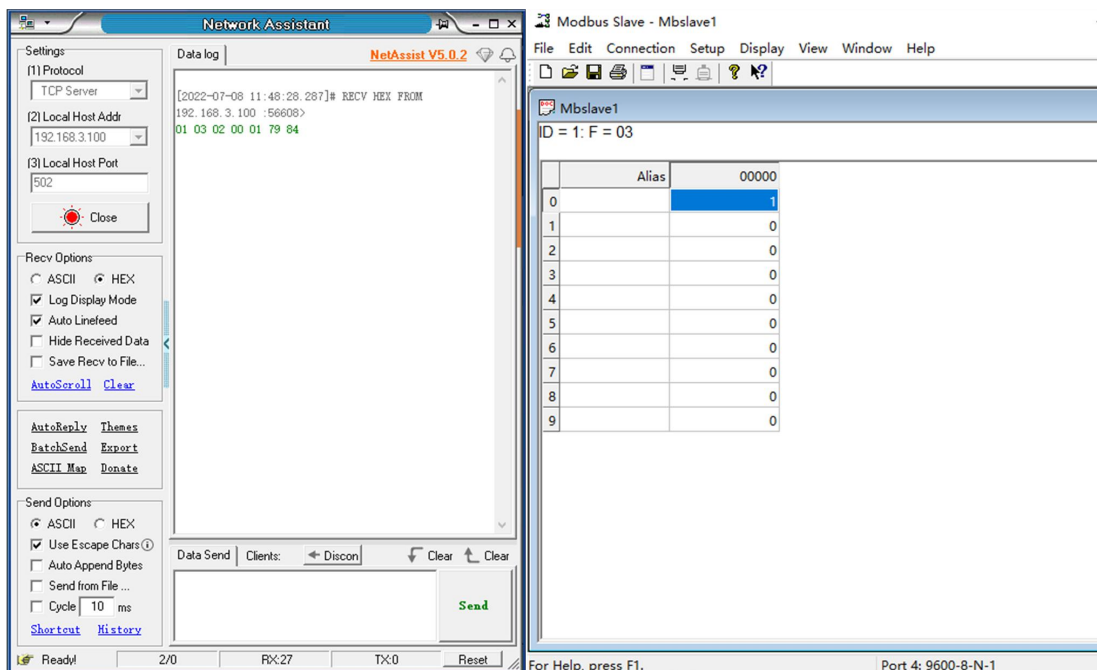
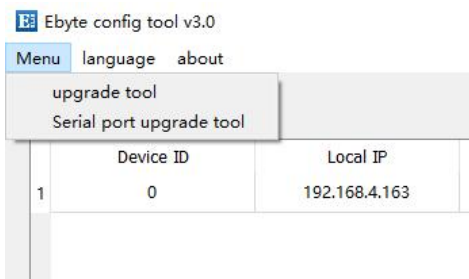TCP client demo (Modbus TCP format):
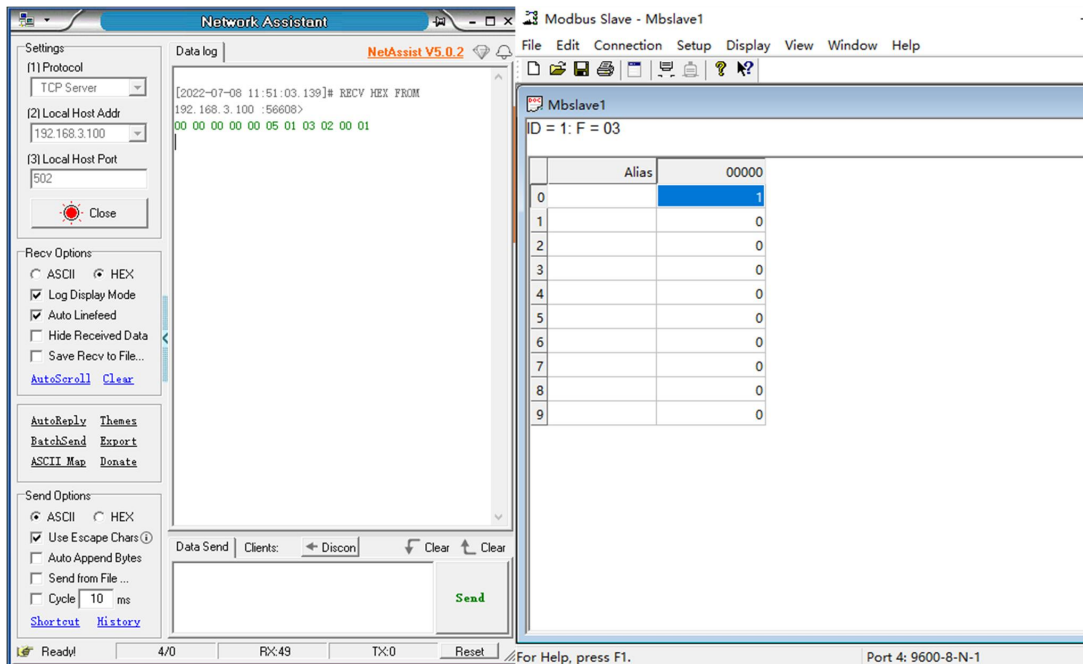

5.5 Firmware Upgrade

Users can upgrade the firmware of the device through the "Ebyte Network Configuration Tool".

5.5.1 UDP upgrade

Step 1: Open "Ebyte Network Configuration Tool" and select "Device Upgrade Assistant" under the "Menu" option;
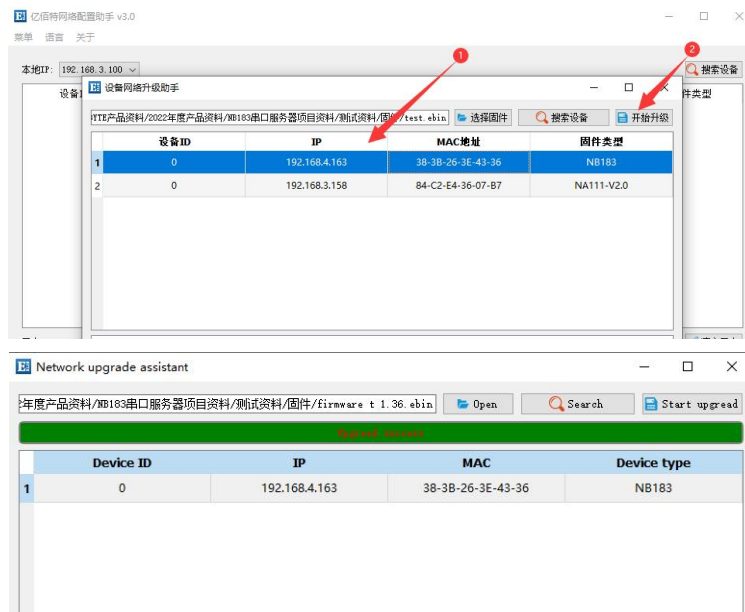

Step 2: Select the firmware to be used, which needs to be obtained from the "Related Downloads" of the corresponding product details on the official website (www.ebyte.com). The demo firmware is not provided on the official website.
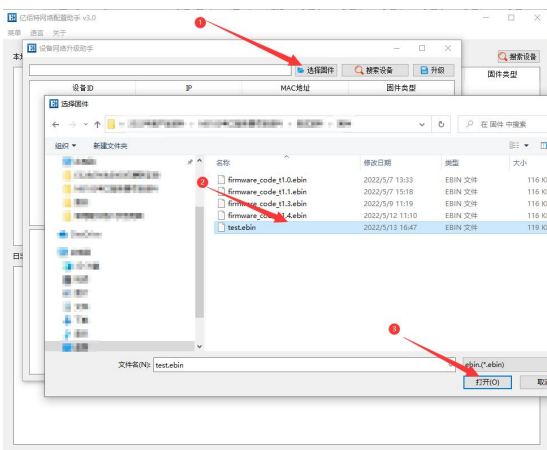
Step 3: Click "Search for devices", the device list displays the currently found devices, click "Stop Search" again.
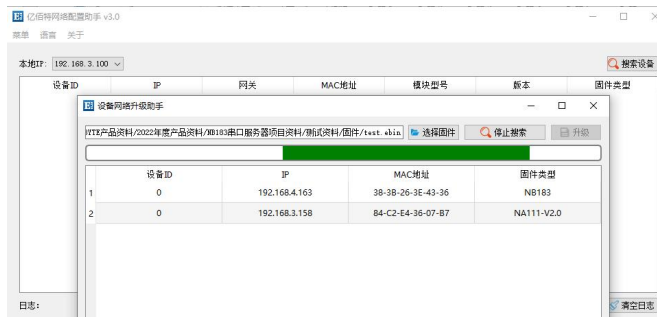
After selecting the device to be upgraded, click Start to upgrade, and wait for the host computer to display that the upgrade is complete.


5.5.2 Serial port upgrade

Only channel 1 (that is, serial port 1, which can use RS232 interface/RS485 interface) supports the use of serial port for upgrade. When the network upgrade fails, or the network environment is complex, it is recommended to use the serial port for upgrade;

Step 1: Make sure the device is powered off and connect serial port 1 correctly;

Step 2: Open "Ebyte Network Configuration Tool", select "Serial Port Upgrade Assistant" under the "Menu" option, select the corresponding serial port and click "Open Serial Port";
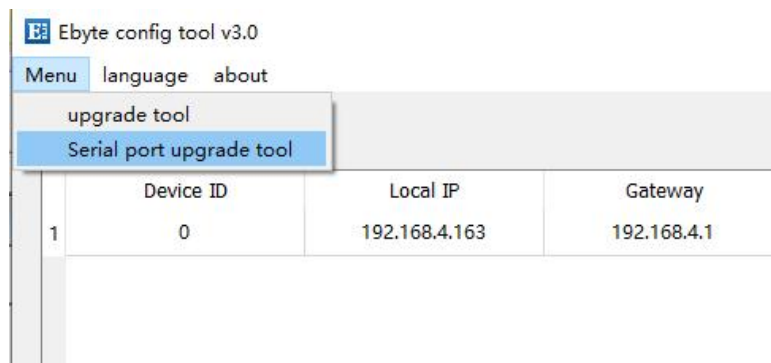

Step 3: Select the firmware to be used, which needs to be obtained from the "Related Downloads" of the corresponding

product details on the official website (www.ebyte.com). The demo firmware is not provided on the official website;
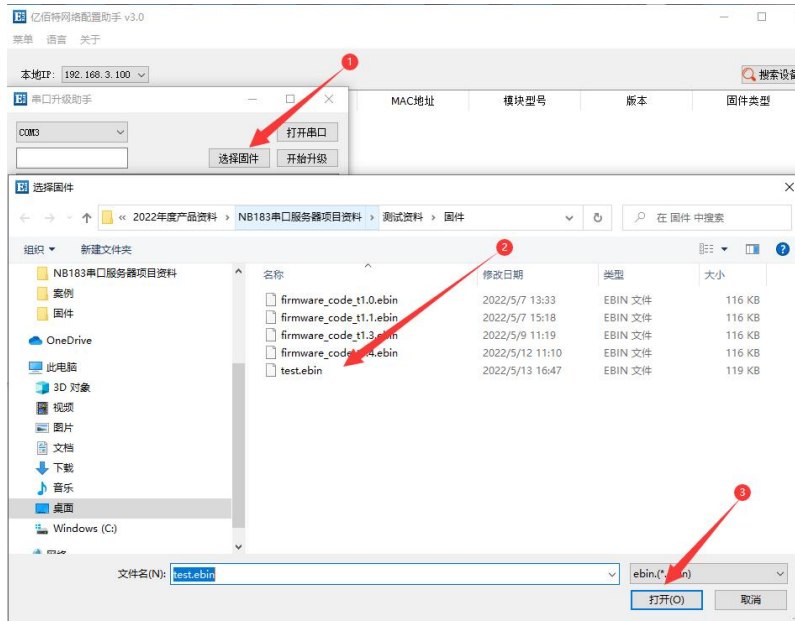
Step 4: Click "Start Upgrade", press and hold Reload of NB183 and then power on the device, wait for the firmware upgrade to complete, click "Cancel" to end the serial port upgrade;
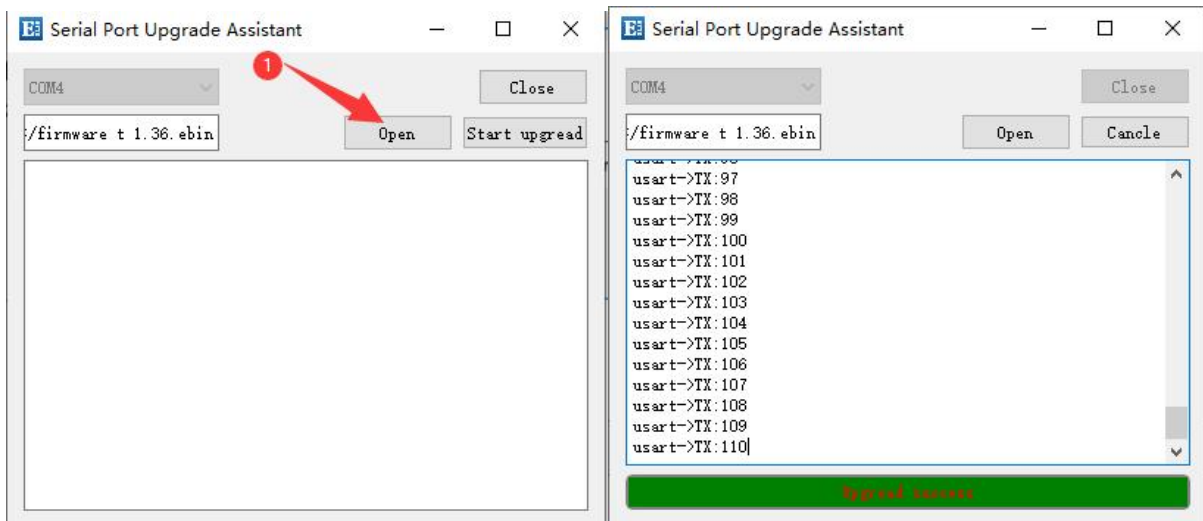
Revision history

Version	Date	Description	Issued by
1.0	2022-07-08	Initial version	LC
1.1	2022-10-20	Merge NB183S	LYL
1.2	2022-10-28	Content changes	LYL

About us

Technical support: support@cdebyte.com

Documents and RF Setting download link:： https://www.cdebyte.com

Thank you for using Ebyte products! Please contact us with any questions or suggestions: info@cdebyte.com

----

Official hotline: 028-61399028   ext. 821

Web:https://www.cdebyte.com

Address: B5 Mould Park, 199# Xiqu Ave, High-tech District, Sichuan, China

Chengdu Ebyte Electronic Technology Co.,Ltd.