ZLAN5112
High performance iot gateway /Modbus Serial server

RS232/485/422 turn TCP/IP
Modbus RTU TO Modbus TCP
Modbus RTU/645 turn JSON

Copyright © 2008 Shanghai Zlan Information Technology Co., Ltd. All rights reserved

ZL DUI 20231219.1.0

The following modifications have been made to the document:

Change log

Date	Version number	Document number	Modify content.
2023-12-19	Rev.1	ZLDUI 20231219.1.0	Release version

Ownership Information

Without the consent of the copyright owner, no part of this document may be reproduced in paper or electronic form.

This document is intended solely for assisting readers in using the product, and Shanghai ZLan Company shall not be liable for any loss or error arising from the use of the information contained herein. The products and texts described in this document are under continuous development and improvement. Shanghai ZLan Information Technology Co., Ltd. reserves the right to modify this document without prior notice to users.

CATALOGUE

1. OVERVIEW 5
2. FUNCTIONAL FEATURES 7
   2.1 Hardware Features 7
   2.2 Software Functions 8
   2.3 Advanced Software Functions 9
3. TECHNICAL PARAMETER 9
4. HARDWARE 11
   4.1 Hardware Description 11
   4.2 Hardware Connections 13
   4.3 Software Installation 14
   4.4 Parameter Configuration 14
   4.5 Detailed parameter meaning 19
   4.6 Modifying Parameters Without Restarting the system 25
   4.7 TCP Communication Test 25
   4.8 Virtual serial port test 28
   4.9 .Modbus TCP testing 31
   4.10 Web configuration method 33
5. WORKING MODE AND CONVERSION PROTOCOL 34
   5.1. Virtual Serial Port Mode 36
   5.2. Direct TCP/IP communication mode 37
   5.3. TELNET Agreement 42
6. DEVICE COMMISSIONING 43
   6.1. Physical network connection 43
   6.2. Network TCP connection 43
   6.3. Data sending and receiving 44
   6.4. ZLVircom Remote monitoring data 45
7. MODBUS ADVANCED FUNCTION 46
   7.1. Enable the Modbus gateway 47
   7.2. Storage Modbus Gateway 47
   7.3. Disable the storage-type function 50

7.4. Multi-host function 50
7.5. Multiple host parameters 52
7.6. Non-storage multi-host 53
7.7. Modbus for Multi-destination IP Addresses 53
8. REGISTER THE PACKET AND HEARTBEAT PACKET 55
8.1. Register package 55
8.2. Heartbeat packets 59
9. MODIFY NETWORK PORT PARAMETERS 61
10. DEVICE MANAGEMENT LIBRARY 62
11. MQTT GATEWAY 62
12. MODBUS RTU TURN JSON 62
13. MODIFY SERIAL PORT PARAMETERS 63
14. REMOTE DEVICE MANAGEMENT 64
15. FIRMWARE UPGRADE METHOD 66
16. ORDERING INFORMATION 69
17. AFTER-SALES SERVICE AND TECHNICAL SUPPORT 69

1. Overview

ZLAN5112 is a high-performance iot gateway specially designed for industrial environments, with serial port server /Modbus gateway function, large memory support 1000 collection points bit-to-JSON, support RS485/422/232 interface communication, 9 ~ 24V wide voltage, terminal power access, with shell ground cable.

The 5112 has one 485/422 interface, one RS232 interface, and one Ethernet interface. Equipped with mounting ears can be fitted with guide buckle, small size, easy to install. The RS485/422 port and 232 port of the 5112 can use the same IP address. Different serial ports are distinguished by port. You can also use different IP addresses for the two routes. This design improves flexibility and manageability, both by saving IP resources through shared addresses and by enabling independent network communication through different IP addresses.

Figure 1 ZLAN5112

The ZLAN5112 supports 2-bit stop bits, data bits smaller than 8 bits (such as 7-bit data with 1 stop bit), and baud rates up to 921.6Kbps.

ZLAN5112 can realize the conversion of TCP/IP protocol to RS232/485/422,

RS485/422 and RS232 work at the same time and can be configured for different baud rates. It can realize the data acquisition of 485 devices through the network, and the network side can use the virtual serial port to connect the serial port software or the direct TCP/IP communication software.

ZLAN5112 supports transparent transmission and Modbus RTU to Modbus TCP function, has a storage Modbus gateway feature, and also supports MQTT protocol. Support MODBUS RTU/645 protocol to JSON over MQTT/HTTP periodically send the function.

ZLAN5112 can be used in:

- Power electronics, smart meters and energy monitoring;
- as the Internet of Things gateway as the communication bridge between devices and the cloud;
- all kinds of automation PLC remote monitoring and program download;
- all kinds of configuration software and equipment communication interfaces;
- Access control security field equipment networking;
- Network information collection of medical equipment;

A typical application connection is shown in Figure 2. The original serial device RS485/232 is connected to the RS485/232 port of the ZLAN5112, and the 5112 is connected to the computer through a network cable. The software on the computer connects to 5112 through TCP/IP, virtual serial port, or cloud server. Thereafter, any data sent by the serial device is transparently transmitted to the computer's software, and the data sent by the software to the ZLAN5112 over the network is transparently transmitted to the serial device.

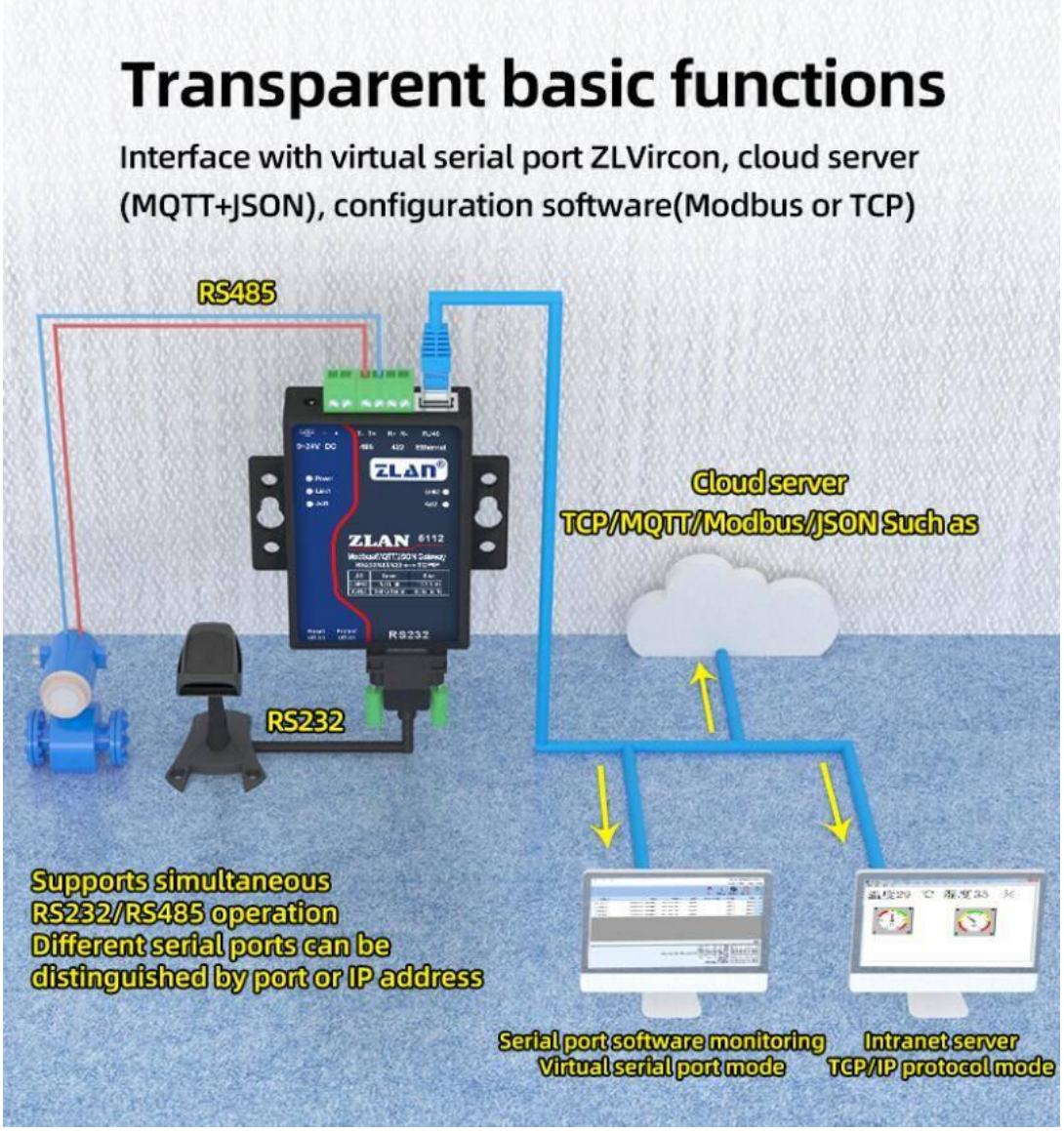

Figure 2 Connection diagram

2. Functional features

2.1 Hardware Features

ZLAN5112 has the following features:

1. There are two options: guide rail installation and desktop mounting ear fixed installation. Especially suitable for industrial rail installation, disassembly is more convenient.
2. Industrial power supply mode: terminal wiring mode, easy to use in industrial

occasions. Can be 9 ~ 24V wide voltage supply. With housing ground wire, effective diversion interference and surge.

3. Rich panel indicators for easy debugging: In connection, there are not only NET lights indicating whether the network cable is well connected, but also LINK lights indicating the establishment of TCP connections; For data indicators, there are independent indicators for serial port to network port and network port to serial port. Easy to debug on site.

2.2 Software Functions

1 RS232 and RS485/422 serial ports can be configured with different baud rates and work independently without interfering with each other.

2 Support IP "split" technology: Two serial ports can be distinguished by port or IP. Port differentiation: Two IP addresses can be combined into one IP, and the ports are different, which is suitable for networks lacking IP. IP address differentiation: One IP address becomes two IP addresses with the same port number. It is suitable for applications with fixed port numbers (for example, Modbus TCP requires fixed port 502).

3 Support TCP server, TCP client, UDP mode, UDP multicast. The TCP client also supports the TCP server function. It supports up to 100 TCP connections as a TCP server and 7 destination IP addresses as a TCP client.

4 Support virtual serial port, equipped with Windows virtual serial port & device management tool ZLVircom.

5 Supports the MAC address sending function when the device is connected, facilitating device management by the cloud.

6 Provide computer side search, configure the device secondary development kit DLL development library.

7 Supports Web browser configuration, dynamic IP address acquisition by DHCP, and DNS.

8 Support cloud remote device search, device parameter configuration, and device program upgrade.

2.3 Advanced Software Functions

Advanced software features supported by the 5112 are:

1 1 Supports the Modbus gateway function and transfers from Modbus RTU to Modbus TCP. It can support storage Modbus, which can automatically collect device data and store it. Non-storage Modbus gateways are also supported. Support ZLMB configurable table Modbus gateway function.

2 Multiple hosts: In the question-and-answer query mode, multiple computers can access the same serial port at the same time. It can also realize multi-host application from one serial port to multiple serial ports.

3 Support MQTT gateway function, support MQTTS encryption.

4 Support Modbus RTU, Modbus TCP, and 645 protocol to JSON format, support HTTP POST, HTTP GET format upload data.

5 Support NTP protocol to obtain network time, used for serial port output for protocol content.

6 Support custom heartbeat packet and registration packet functions: can facilitate communication and device identification with the cloud.

7 Password authentication is required for TCP connection establishment to ensure connection security.

8 Supports http data submission and delivery. The cloud can directly use http GET commands to interact with serial port data of the device.

3. Technical parameter

Table 1 Technical parameters

Outline	
Interface：	RS485/422：terminal； RS232：Standard 232 male head
Power supply：	5.5mm inside negative outside positive standard power socket; 9-24V terminal power supply
Size：	L x W x H =94mm*65mm*25mm（Housing size, excluding interface, ±1mm)
Communication	

Ethernet：	10M/100M, 2 KV surge protection		
Serial port：	RS485/422×1；RS232X1		
Serial port parameter			
Baud rate：	300~921600bps	Check bit：	None, Odd , even Mark, Space
Digit bits：	5～9 bits	Flow control：	RTS/CTS，DTR/DCR, NONE
Software			
Protocol：	ETHERNET、IP、TCP、UDP、HTTP、ARP、ICMP、DHCP、DNS		
Configuration：	ZLVirCOM tool, WEB browser (can customize the web page), device management function library, serial port AT command configuration, device cloud management (ZLan server)		
Conversion protocol:	Modbus TCP、MQTT、JSON、RealCom、HTTP		
Modbus gateway：	Support multi-host mode, storage mode, Pre-configuration table (ZLMB) mode		
communication method：	Socket、virtual serial port、Device management library		
Working mode			
TCP server, TCP client, UDP, Real Com Driver, Modbus TCP, Telnet			
Power supply：			
Power：	9～24V DC，100mA@12V		
Environment requests			
Operation temperature：	-40~85℃		
Storage temperature：	-45~120℃		
Humidity range：	5~95%humidity accordingly		

4. Hardware

4.1 Hardware Description

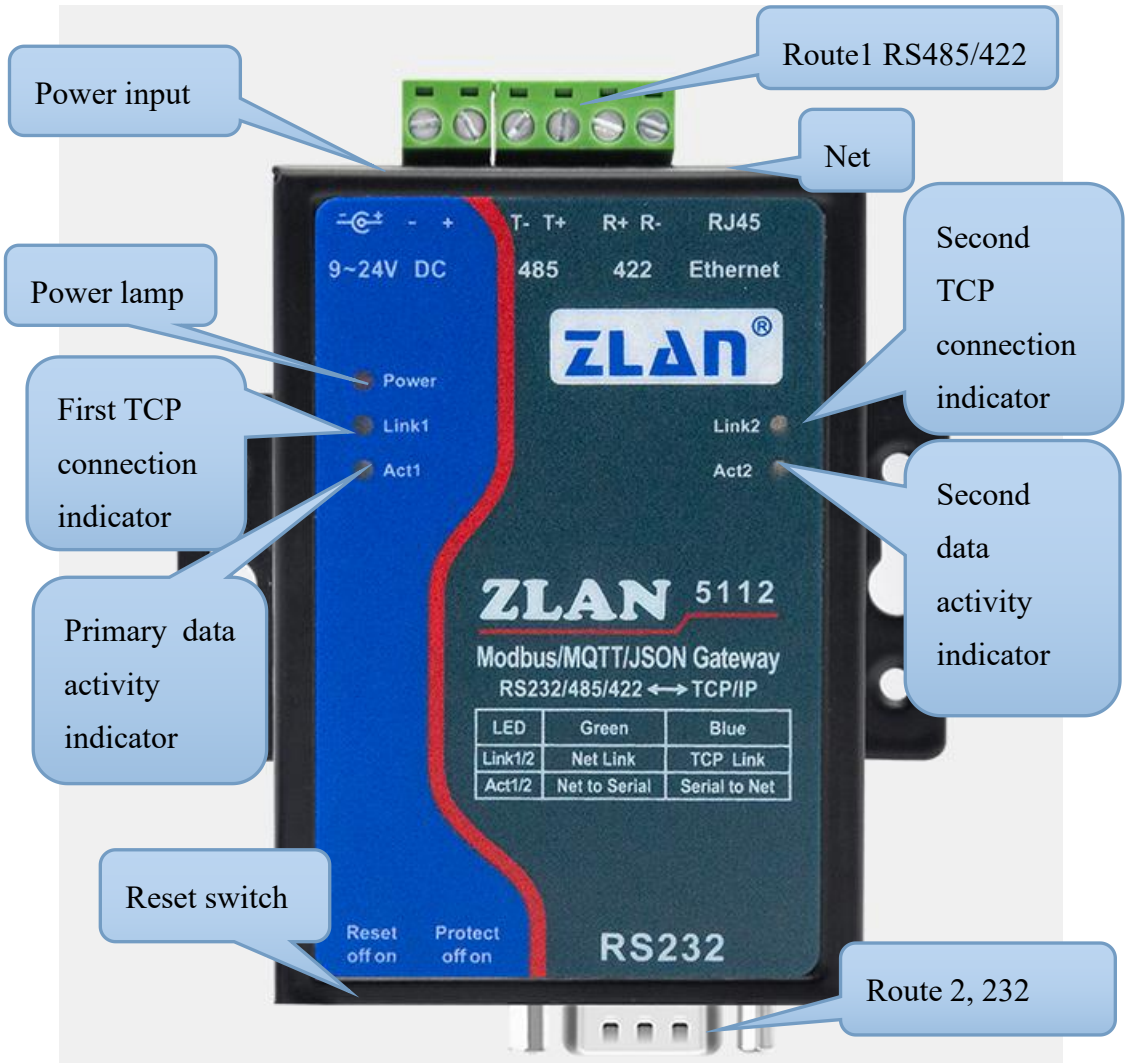

Figure 3 Main view

The main view of the ZLAN5112 serial port server is shown in the figure above. The shell uses a black radiation-resistant SECC metal shell.

1 Power input: The wiring terminals supply power to the 5.5mm round hole power adapter or 9-24V terminals.

2 Serial port: 5112 has 1 485/422 serial port,485 can be loaded with 32 sets. The maximum communication distance is 1200 meters. Generally, when the RS485 line exceeds 300 meters, it is necessary to use the terminal resistance, and the 485 terminal resistance is 120 ohms. 1 RS232 interface, standard 232 male head.

3 Network port: Connects network cables and supports automatic crossover.

4 Indicator: divided into Power, Link1/2, ACT1/2, light, respectively indicate power, connection indicator, data indicator.

Table 2 Indicator meanings

Power lamp	Power indicator light
Link1/2 lamp	When the TCP connection is established (or in UDP mode), Link is green. It can be used to determine whether the serial port server establishes a communication link with the host software.
ACT1/2 lamp	When the network port sends data to the serial port, the serial port sends data to the network port.

Use the indicator to debug communication methods:

1) If the Link indicator is not green (only considering the TCP working mode), the host software is not connected to the serial port server. Please consider whether the IP address is configured on the same network segment.

2) If the ACT light is not green, there is no data communication.

5 Installation method: Install the desktop mounting ears

Figure 4 Hanging ears

If there is a guide rail, you can place another order to choose the guide rail buckle, and directly install the equipment into the guide rail, as shown in the following figure:

Figure 5 Installation diagram

4.2 Hardware Connections

Generally, the serial port server only needs to be connected to the power supply, serial port, and network cable.

The on-site 2-wire power supply can be used, and the positive and negative terminals of the power supply can be directly connected.

The serial port must be connected according to the user serial port device. Connect the positive of 485 to T+ and the negative of 485 to T-. RS232 is male, standard 232 male line sequence, docking customer 232 equipment.

Network port connection A network cable can be directly connected to a

computer or connected to the network through a switch.

4.3 Software Installation

ZLVircom can be used to configure parameters such as device IP and create virtual serial ports. If the virtual serial port function is not required, you can download the no-installation version. Download address：http://www.zlmcu.com/download.htm

Figure 1 ZLVircom version

Software name	Specifications
ZLVircom Device Management Tool (non-installed version)	The non-installed version does not include the virtual serial port function.
ZLVircom-Device Management Tool (Installation version)	The installed version contains ZLVircom_x64.msi and ZLVircom_x86.msi. 64-bit operating systems install x64, 16-bit operating systems install x86 versions.

Follow the default prompts during installation. After installation, zlvircom will be started every time the computer starts, which is used to create a virtual serial port.

4.4 Parameter Configuration

After ZLVircom is installed and equipment hardware connection is completed, ZLvircom software is run as shown in the figure, and then click "Equipment management" as shown in the figure. ZLVircom can be used to search and configure equipment parameters in different network segments, which is very convenient, as long as the equipment and the computer running ZLVircom are in the same switch.

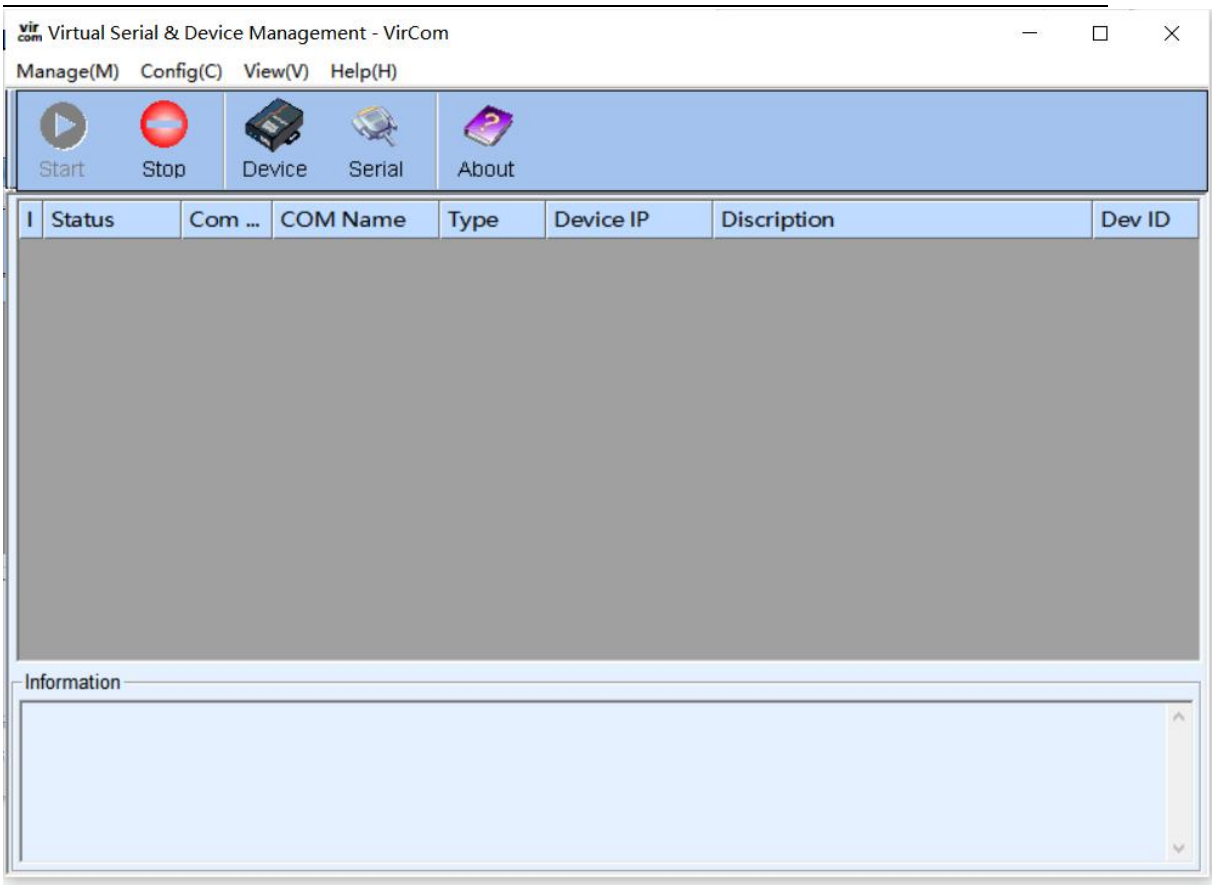

Figure 6 ZLVircom main interface

The list of devices with two ports is displayed. So you need to sort by name so that the different ports of the same module are in a contiguous area. The method of sorting is to click on the title bar "Device name". Before delivery, the ports of each device are named ECF78C-01 to ECF78C-02. The ECF78C on the left is the device ID, that is, the last six digits of the first PORT ID. -01 to -02 indicates the PORT. Note The device name can be changed. If the device name has been changed, you can view the PORT ID of the device in the PORT list.

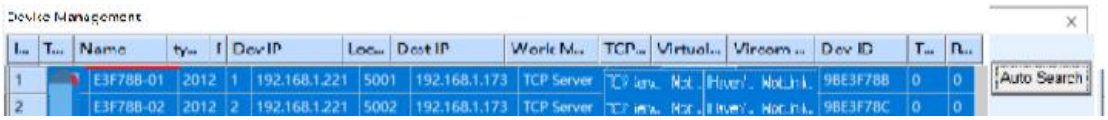

Figure 7 List of devices

After sorting, the two ports of the module are arranged consecutively. At this time, start the selection of the first device ECF78C-01 and drag the mouse down until the devices with all two ports are selected. Then click "Batch Edit". Batch Edit will cause

both ports to be modified at once, with the correct IP and PORT.

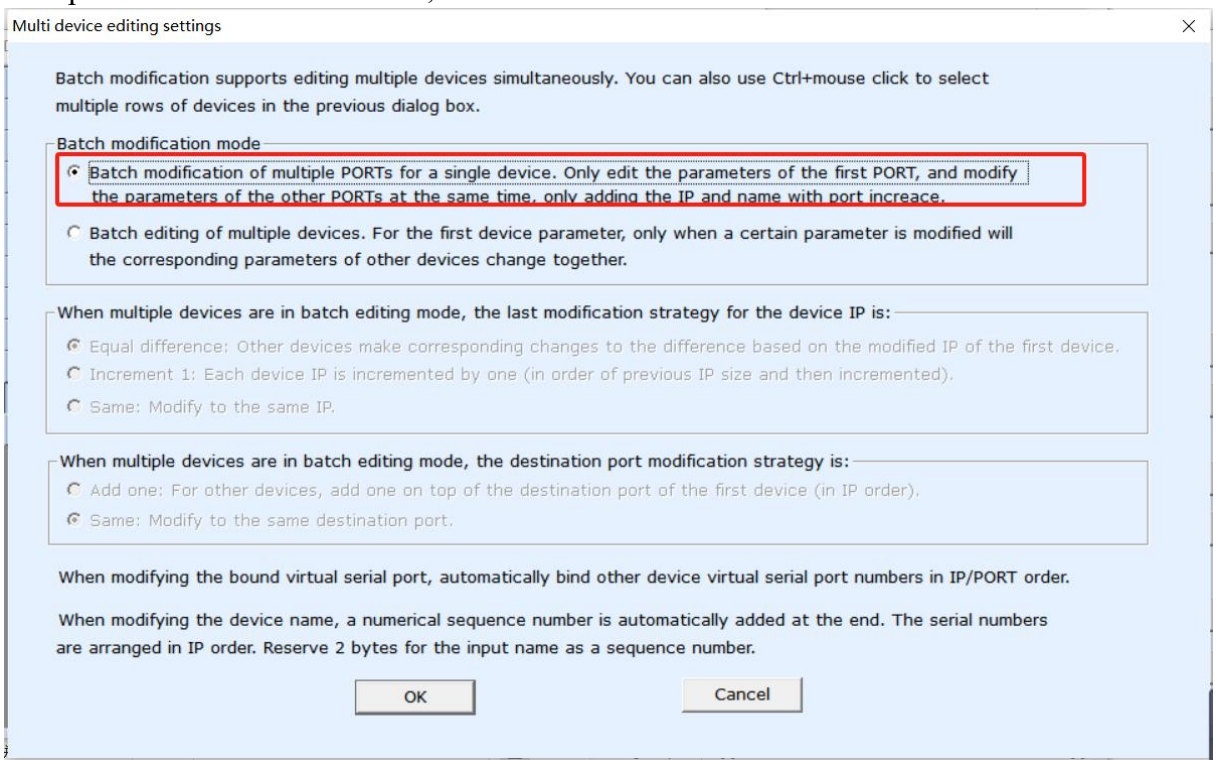

Figure 8 Batch setup options

Since all ports of the 1~2 channels are selected, the software recognizes the one-time modification of the 2 ports of this module, as shown in the figure above. When you select the batch modification method for two ports of a single module, you do not need to modify the content of the parameter dialog box, you can directly click "Modify Settings" to modify it. In this case, although the PORT1 device parameter is not modified, the second PORT will be modified into the device parameter of PORT1.

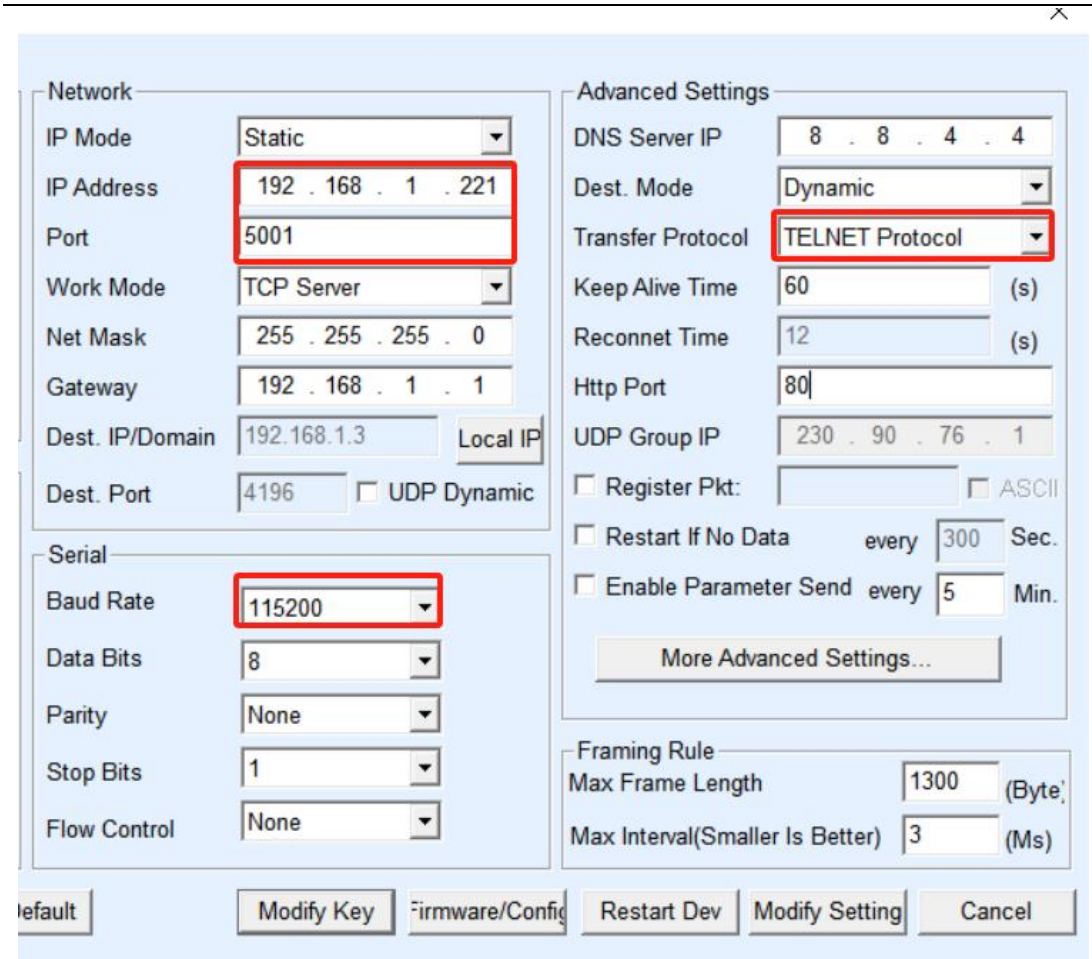

Figure 9 parameter diagram

In the parameter dialog box, you can modify the baud rate and other parameters. If you want to switch from TELNET to serial port, select TELNET. ECF78C-01 is automatically filled in by the software according to the PORT ID. IP address 192.168.1.221 is the first IP address, and PORT 5001 is the number of the first port. Then click the "Modify Settings" button. The software changes two ports in batches at a time.

Different ports of the same module can also be set to different IP addresses. However, if you do not need to change the IP addresses of fixed ports, you are advised not to use multiple IP addresses. Here's how to change to one PORT, multiple ips.

First, select the PORT number you want to modify, and then click "Batch Edit".

In batch configuration modification, you can change the Multiple Ports for a single

device mode to Batch Edit for multiple devices mode, which does not intelligently identify IP addresses and ports.

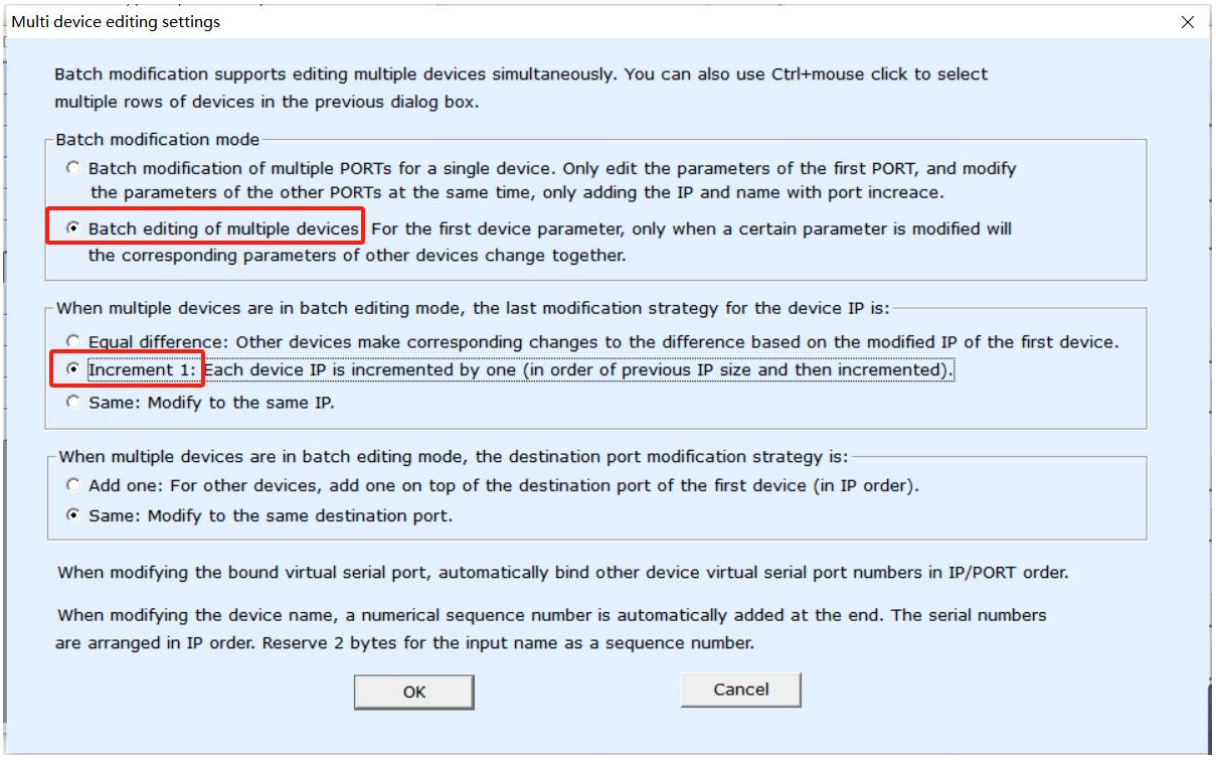

Figure 11 Modifying multiple IP addresses Step 2

In IP mode, select Add One. That is, the IP addresses that follow the first IP address are automatically added by one.

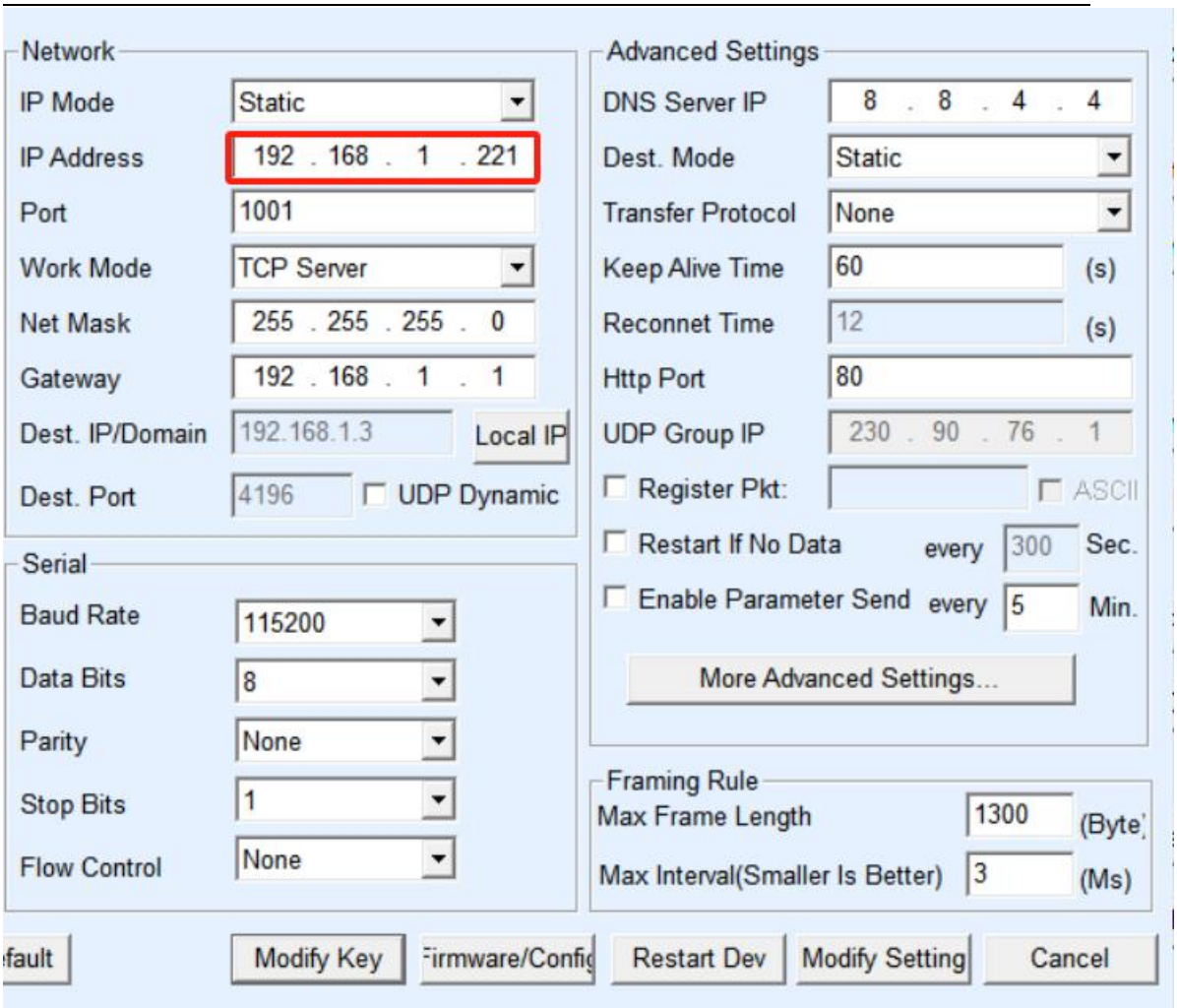

Figure 12 Modifying multiple IP addresses Step 3

In the Modify Parameters dialog box, be sure to change the IP address. If you do not change the IP address, no operation will be performed. Because this mode will only change the change of the parameter down, no change of the parameter will not affect.

ECF78C-01	2012	1	192.168.1.221	5001	192.168.1.3	TCP Server
ECF78C-02	2012	2	192.168.1.222	5002	192.168.1.3	TCP Server

Figure 13 Result of modifying multiple IP addresses

According to the modification result, the number of IP addresses is now 2, from 192.168.1.221 to 192.168.1.222.

4.5 Detailed parameter meaning

Because the functions and parameters of the two serial ports are independent and

similar, the subsequent parameters only describe the parameters of one serial port. The meanings are as follows:

Figure 2 parameter contents

Parameter name	value range	Contents
virtual serial port	none, created virtual serial port	You can bind the current device to an existing virtual serial port. Add a COM port in Serial Port Management on the home screen.
Device model		Only the model of the core module is displayed
Device name	random	You can give the device an easy-to-read name, up to 9 bytes, support Chinese names.
Device ID		factory unique ID, cannot be modified.
Firmware version		Firmware version of the core module
Functions supported by the device		See Table 3 for features supported by the device
IP mode	static、DHCP	Users can choose between static or DHCP (dynamic IP acquisition)
IP address		IP address of the serial port server
Interface	0～65535	Listening port of the serial port Server in TCP Server or UDP mode. If you use port 0 as the client, you are advised to set port 0 to improve the connection speed. If port 0 is used, the system randomly assigns a local port. The difference between this and non-zero port is: (1) When the local port is 0, a new TCP connection is established with the PC when the module restarts, and the old TCP connection may not be closed, and the device may have multiple fake connections. Generally, the host computer wants to close the old connection

		when the module restarts; Specifying a non-zero port closes the old connection. (2) If the local port is 0, the TCP connection takes a shorter time to re-establish. When the serial port server is in TCP client mode, it also acts as the TCP server to listen for incoming connections on the port. In this case, the local port number used by the TCP client to connect to the server is Port +1000.
Working mode	TCP server mode, TCP client mode, UDP mode, UDP multicast mode	When set to TCP server, the serial server waits for the computer to connect. If TCP client is configured, the serial port server initiates a connection to the network server specified by the destination IP address.
Subnet mask	For eg.：255.255.255.0	The subnet mask must be the same as that of the local LAN.
Gateway	For eg.：192.168.1.1	It must be the same as the local LAN gateway
Destination IP address or domain name		In TCP client or UDP mode, data is sent to the computer indicated by the destination IP or domain name.
Destination port		In TCP client or UDP mode, data is sent to the destination port of the destination IP address.
Baud rate	300、600、1200、2400、4800、7200、9600、14400、19200、28800、38400、57600、76800、115200、230400、460800、921.6K	Serial port baud rate
Digit bits	5、6、7、8、9	
Check bits	None, Even, Odd, tag,	

	space	
Stop bits	1、2	
Flow control	No flow control, hard flow control CTS/RTS, hard flow control DTR/DCR, soft flow control XON/XOFF	Only available for RS232 serial port
DNS server		If the destination IP address is described by a domain name, enter the IP address of the DNS server. If the IP address mode is DHCP, you do not need to specify the DNS server. The DNS server automatically obtains the IP address from the DHCP server.
Destination mode	Static , dynamic	TCP client mode: In static destination mode, the device automatically restarts after five consecutive failed attempts to connect to the server.
Transfer protocol	NONE 、 Modbus TCP<->RTU 、 Real_COM、TELNET	NONE indicates that data is transmitted transparently from the serial port to the network. Modbus TCP<->RTU will convert Modbus TCP protocol directly into RTU protocol, which is convenient to cooperate with Modbus TCP protocol; RealCOM is designed to be compatible with the older version of the REAL_COM protocol. It is a virtual serial port protocol. However, it is not necessary to select the RealCom protocol when using the virtual serial port. The TELNET protocol allows the network to log in to our device through TELNET to communicate with the serial port
Keepalive timing time	0～255	Heartbeat interval. (1) If the value ranges from 1 to 255 and the device is in TCP client working mode,

		the device automatically sends TCP heartbeat packets at Keepalive intervals. This ensures the TCP validity of the link. If the value is set to 0, there is no TCP heartbeat. (2) If the value is set to 0 to 254, and the conversion protocol is REAL_COM, the device will send data with length 1 and content 0 at keepalive intervals to implement the heartbeat mechanism in the Realcom protocol. If the value is set to 255, there is no realcom heartbeat. (3) When the value is set to 0 to 254, if the device works on the TCP client, the device will send device parameters to the destination computer at keepalive intervals. If the value is set to 255, no parameter is sent, enabling remote device management.
Disconnected reconnection time	0～255	In TCP client mode, when the connection fails, the TCP connection is re-initiated to the computer at disconnection Reconnection time intervals. The value ranges from 0 to 254 seconds. If the value is set to 255, the reconnection is never performed. Note that the first TCP connection (such as hardware power-on, device restart through zlvircom software, and no data light) is generally carried out immediately, and only after the first connection fails will it wait for the "disconnection reconnection time" to try again, so the "disconnection reconnection time" will not affect the normal connection establishment time between the network and the server.
Web access port	1～65535	Default is 80

Multicast address		Under UDP multicast
Enable registration package		When a TCP connection is established, the registration packet is sent to the computer. The realcom protocol must be selected after the registration package is enabled. TCP server and TCP client modes are supported。
Digit packet length	1~1400	One of the serial port framing rules. Serial port server After receiving data of this length, the serial port sends the received data to the network as one frame.
Packet interval	0～255	Serial frame rule 2. When the data received by the serial port server stops for a period longer than the specified period, the received data is sent to the network as a frame。

The functions supported by the device are described as followings：

Figure 3 Device supported functions

Name	Specifications
Domain name system	The destination IP address can be a domain name (such as the first www server address).
REAL_COM protocol	A non-transparent serial port server protocol, suitable for multiple serial port servers to bind virtual serial ports over the Internet. Because the protocol contains the MAC address of the device, it helps the host computer to identify the device. Generally, it can not be used.
Modbus TCP to RTU	Can realize Modbus TCP to RTU. It also supports the multi-host function.
Serial port modification parameter	Support serial port class AT instruction to configure and read device parameters.
Automatic acquisition IP	Support DHCP client terminal protocol

Multi TCP connection	The TCP server supports more than one TCP connection.
UDP broadcast	UDP broadcast
Multi purpose IP	As a TCP client, seven destination IP addresses can be connected at the same time.
P2P function	Supports the function of accessing devices in any network through P2P traversing technology. Models ending in N support this function.
TELNET function	Connect to the Zlan serial port server through Telnet to monitor the serial port of the device.

4.6 Modifying Parameters Without Restarting the system

Because PORT1 and port2 belong to the same module, modifying the parameters of one PORT will restart the entire module. However, in the following cases, only the local PORT will be restarted, and other ports of the module will not be affected.

1. Just click the "Restart Device" button without modifying any parameters.
2. Switch between None and TELNET Protocol.
3. Modify only one or more of the following parameters:
   a) Local port and destination port
   b) Baud rate, data as, check bit, flow control, stop bit
   c) Device name, packet interval, packet length

4.7 TCP Communication Test

Because the serial port functions and parameters of RS485/422 and RS232 are independent and similar, the subsequent communication introduction only describes the configuration of one of the serial ports.

After device parameters are configured, you can use the serial port tool or TCP debugging tool to test the TCP connection.

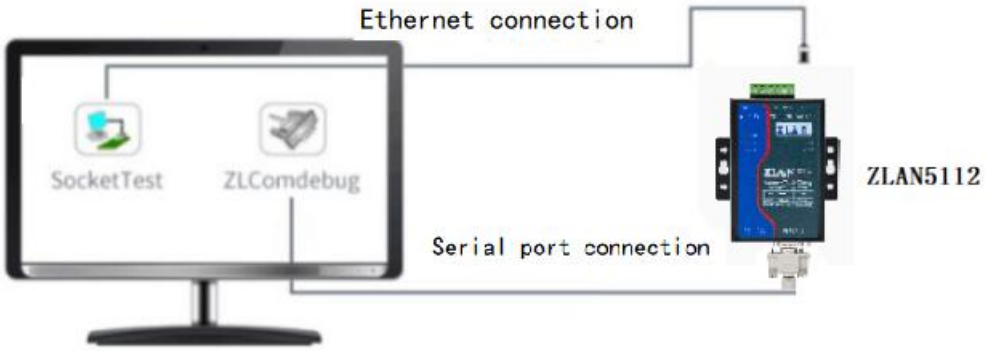

Figure 14 TCP communication diagram

Assuming the PC COM port (USB turn RS485/232 lines) and serial port serial port server connection, then open the ZLComDebug (http://www.zlmcu.com/download/Comdebug.rar) serial debugging assistant, and open the corresponding COM figure 15; Open TCP&UDP debugging assistant SocketTest (http://www.zlmcu.com/download/SocketTest.rar), and as a way of TCP client, fill in the destination IP for a serial port server IP (currently 192.168.1.221), The destination port is 5001, then click the "Open" button as shown in Figure 16. On the Ttest of SocketTest, input "socket send" and click send, the data is transferred to the RS485 interface through the network port of the serial server, and then sent to ZLComDebug, which is then displayed in ZLComDebug. On the other hand, input "Comdebug send" in ZLComDebug and click "send" to send to socket test.

This demonstration demonstrates the transparent data forwarding function of serial port to network port and network port to serial port of serial port server.

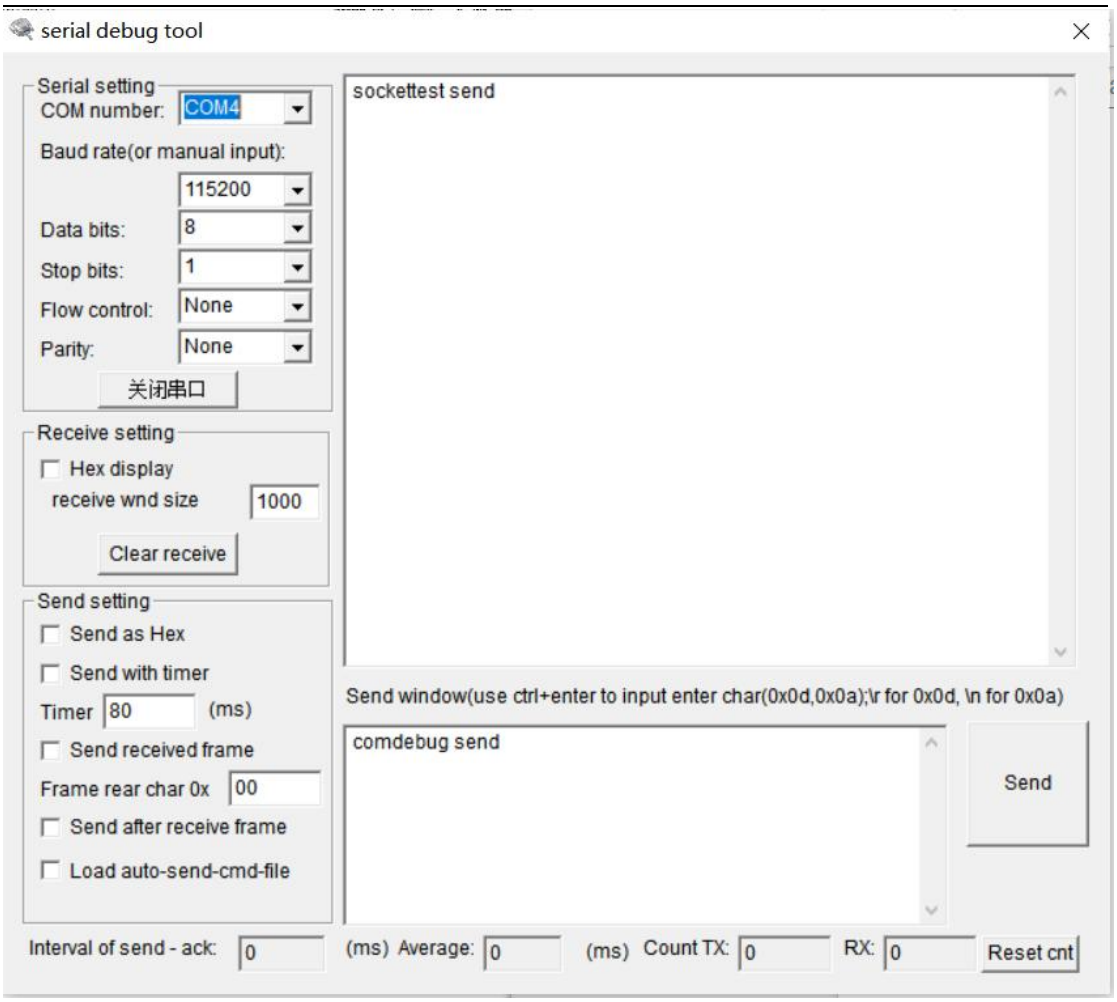

Figure 15 comdebug send and receive page

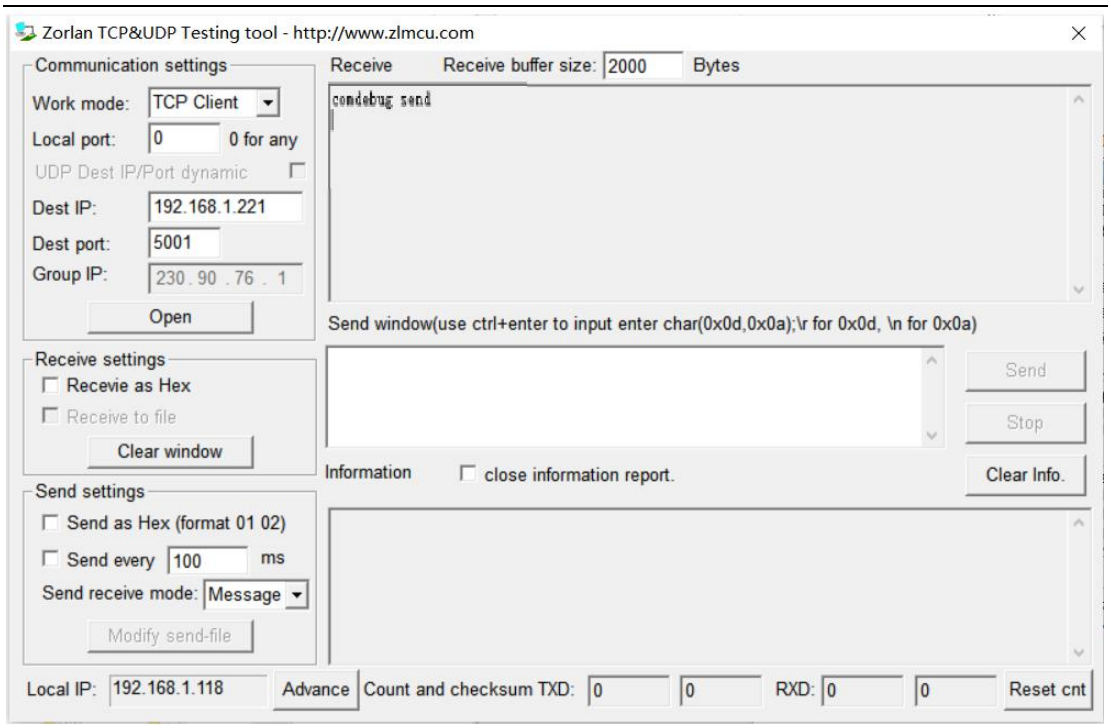

Figure 16 sockettest send and receive page

4.8 Virtual serial port test

The SocketTest shown in Figure 14 communicates directly with the serial server through TCP. In order to enable users to communicate with the serial server even with the developed serial software, a virtual serial port needs to be added between the user program and the serial server. As shown in Figure 17, ZLVircom and the user program run on a computer. ZLVircom virtualizes a COM port and makes this COM port correspond to the serial port server. When the user program opens COM communication, it can be sent to the user serial port device through the ZLVircom serial port server. Here's how to do it:

Figure 17 Functions for virtual serial port

Click "Serial port management" in the main interface of ZLVircom, then click "Add", select to add COM5, where COM5 is the COM port that does not exist in the computer.

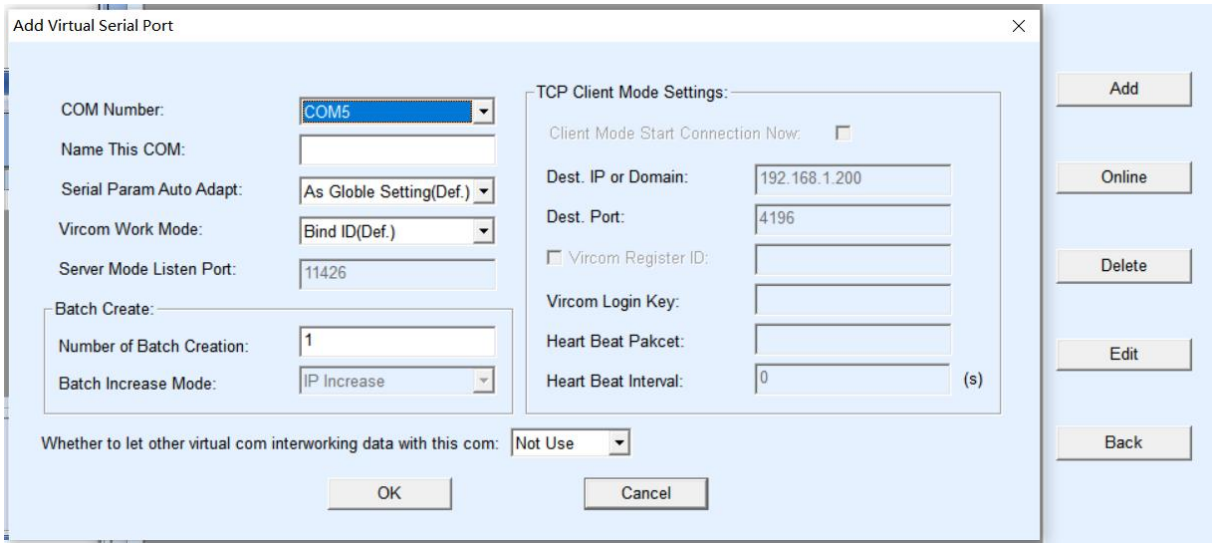

Figure 18 Add virtual serial port

Then go to Device Management and double-click the device that you want to bind to COM5. Select COM5 from the Virtual Serial Port list in the upper left corner. Then click "Modify Settings". And return to ZLVircom's main interface. You can see that COM5 has connected to the device whose IP address is 192.168.1.211. COM5 can be used instead of SocketTest to communicate at this time.

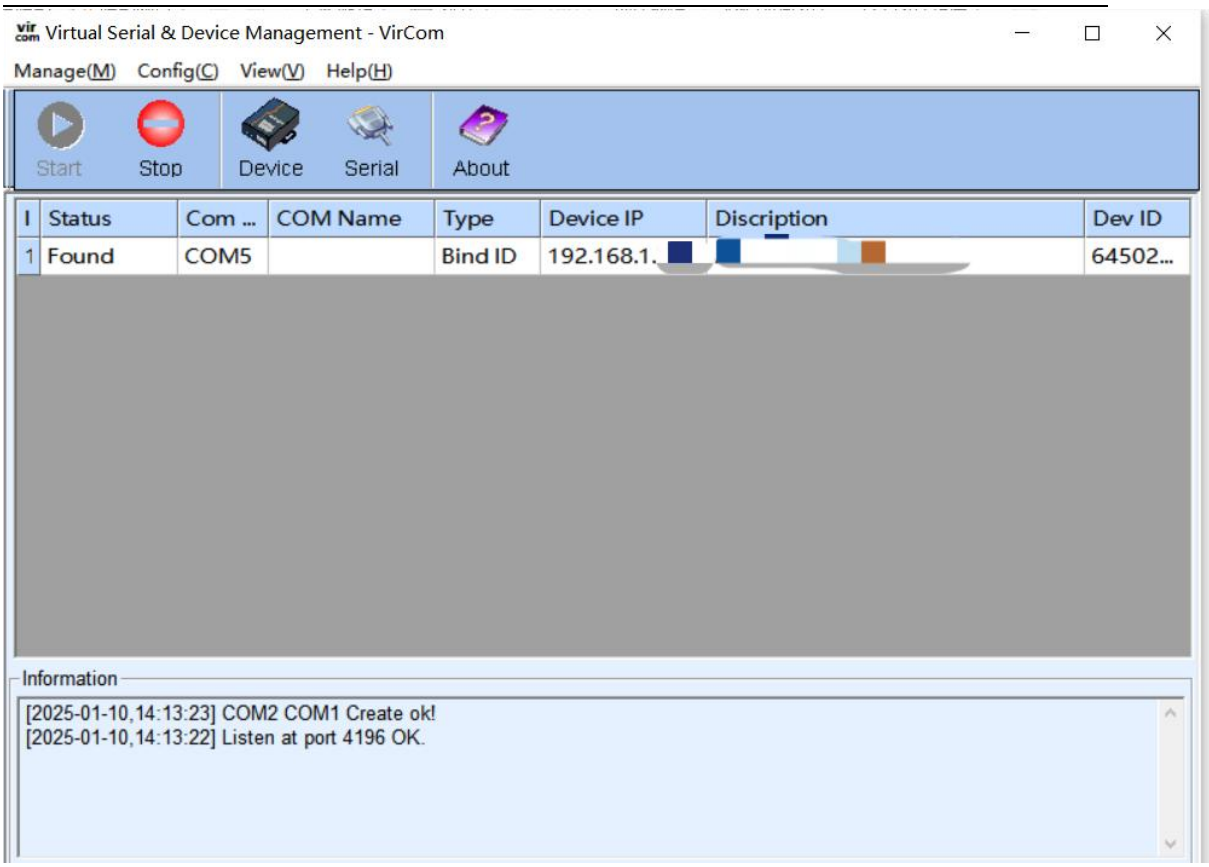

Figure 19 virtual serial port is connected

Open ZLComdebug to simulate the user's serial port program, open COM5(the above virtual serial port), and open a ZLComdebug to simulate a serial device, open COM4(hardware serial port). The data link sent by COM5 is as follows: COM5ZLVircom serial port Server network port Serial port Server serial port COM4. Conversely, COM4 to COM5 can also transmit data: COM4 serial port server serial port Serial port server network port ZLVircomCOM5. Figure 22 shows the two parties sending and receiving data.

If the COM4 is replaced by the user serial port device, the COM5 can realize the communication with the user device.

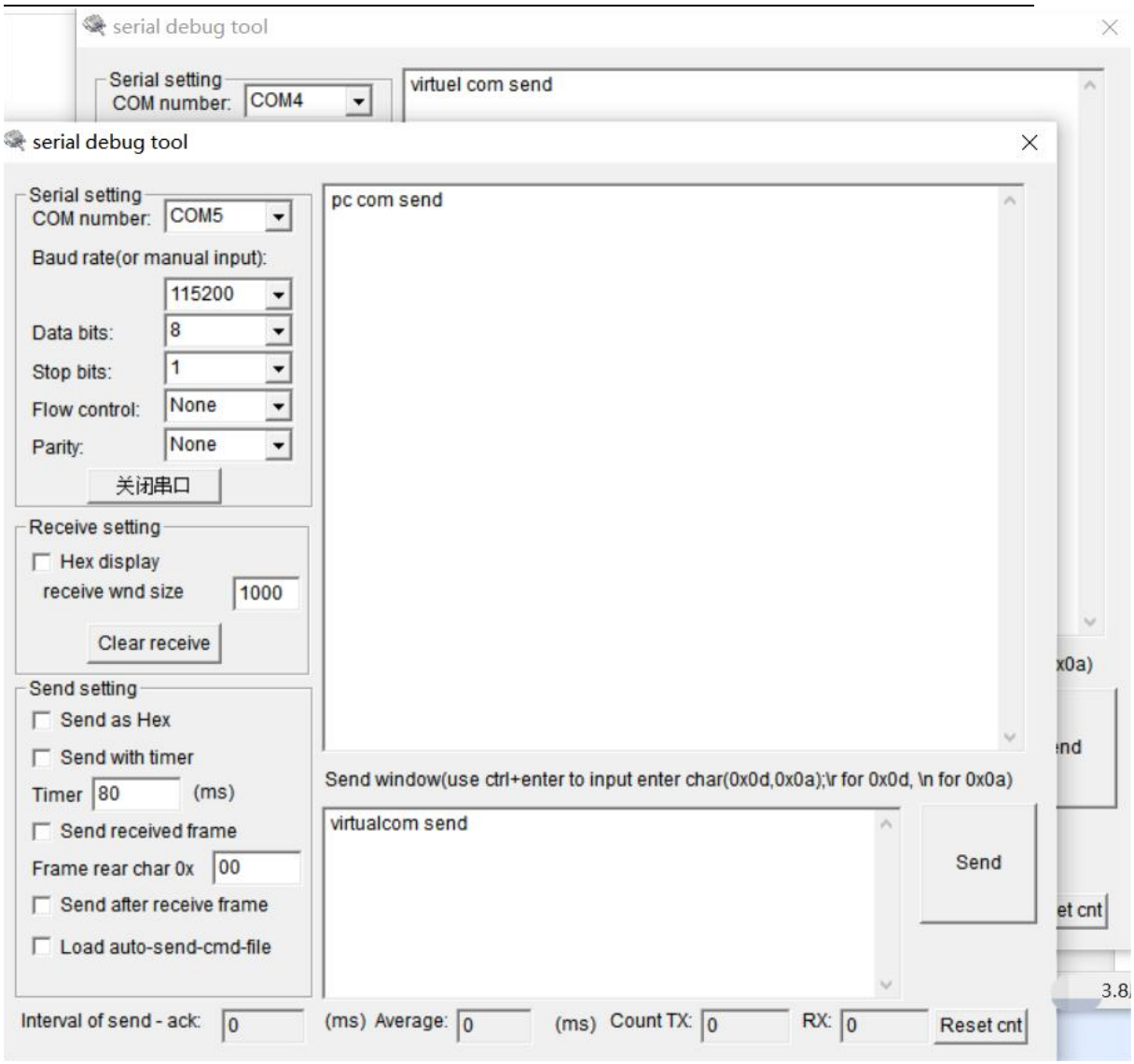

Figure 20 Communication through the virtual serial port

4.9 .Modbus TCP testing

By default, serial port and network port data are transmitted transparently. If you need to convert Modbus TCP to RTU, you need to select "Modbus TCPRTU" as the conversion protocol in the device setup dialog box, as shown in Figure 23. In this case, the device port automatically changes to 502, and the Modbus TCP tool connects to port 502 of the serial port server IP address. The Modbus TCP command is converted to RTU command and output from the serial port. For example, if the port of the serial port server receives the Modbus TCP command 00 00 00 00 00 06

01 03 00 00 00 00 00 0a, the serial port outputs the command 01 03 00 00 00 00 00 a c5 cd. Note: The serial port may send more than one 01 03 00 00 00 00 0a c5 cd command, this is because the default Modbus is stored mode, will automatically rotate the query command. I'll explain how to switch to non-storage mode later.

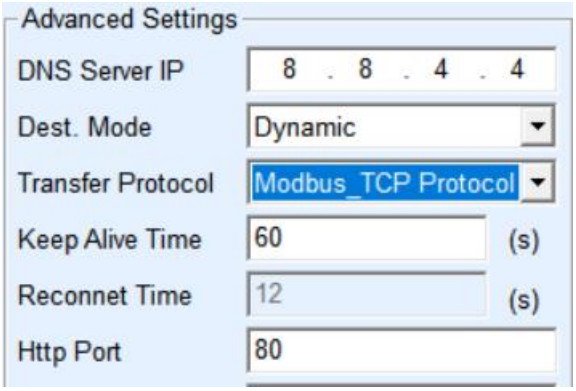

Figure 21 Enable Modbus TCP function

如 If the Modbus TCP software of the user is used as the Slave station, it is necessary to change the working mode to the client on the basis of selecting the conversion protocol, change the destination IP address to the IP address of the computer where the Modbus TCP software is located, and the destination port is 502, as shown in Figure 22.

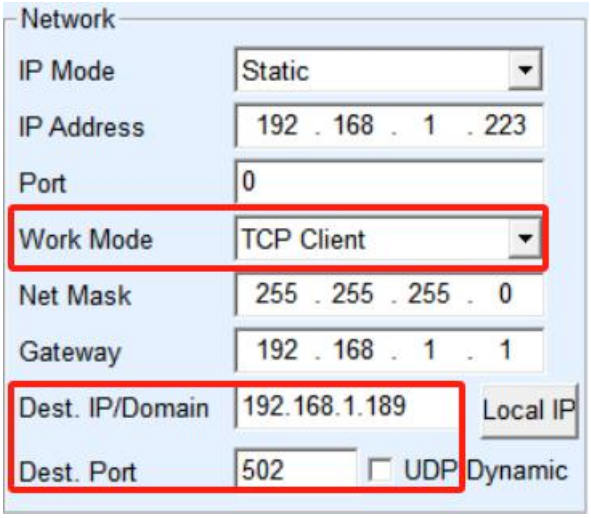

Figure 24 Modbus TCP as client side

4.10 Web configuration method

ZLVircom can search and configure device parameters in different network segments. Web configuration requires first to ensure that the computer and the serial server are in the same IP segment, and the IP address of the serial server needs to be known in advance. But Web configuration can be done on any computer without ZLVircom.

1. In the address box of your browser, enter the IP address of PORT1 of the serial port server, for example, http://192.168.1.221.

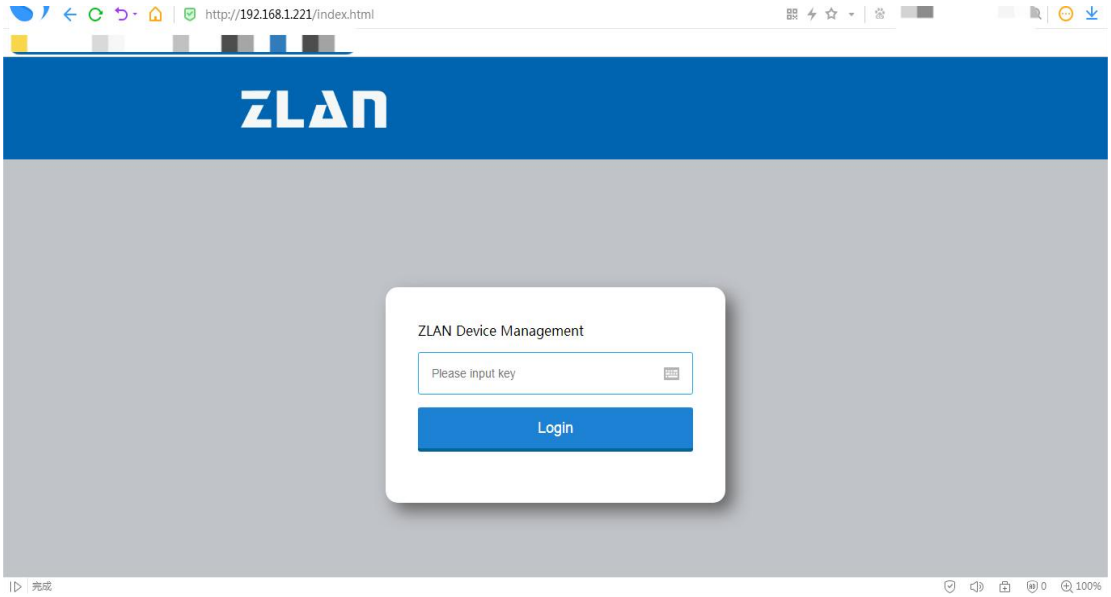

Figure 23 web login in page

2. Enter a Password in Password: there is no password by default. Click the login button to sign in.

Figure 24 Web configuration page

3. On the page that appears, select PORT1-2 that you want to modify. Modify the corresponding PORT parameters and click Submit. For details about related parameters, see Table 4.
4. Precautions: 1. The IP address of the web page is the IP of the first serial PORT (PORT1), and the IP of the second port port cannot enter the web page. Submitting changes Only the parameters of the current PORT are submitted. For example, after parameters are modified on the PORT2 page, click Submit to modify only the parameters of PORT2.

5. Working mode and conversion protocol

In different applications can choose different serial server working mode, conversion protocol, so as to be more stable and reliable use, the following details.

The use of serial port server is basically divided into two types: virtual serial port and non-virtual serial port, as shown in FIG. 14 TCP communication diagram and FIG. 17 function of virtual serial port. The user software that needs to be connected with the virtual serial port is a serial port (COM port), that is, the user software and the user device are serial ports. Non-virtual serial port mode The user software is directly TCP/IP communication but the user device is still serial port.

In the non-virtual serial port mode, the conversion protocol section is divided into transparent transmission, Modbus TCP to RTU, Realcom, and TELNET. If the user software uses the Modbus TCP protocol and the lower computer uses Modbus RTU, select Modbus TCP to RTU. Realcom is used only when a multi-serial port server serves as the TCP client to connect to a server and the virtual serial port is used on the server. TELNET is used to monitor the serial port of the device when the server is connected to the Zolan module through Telnet.

Usage is summarized as follows:

Table 6 Network configuration modes

No.	Virtual serial port usage	device working mode	Transfer protocol	Specifications
1	Use	TCP server	none	It is suitable for the occasion when the user software opens COM port to collect data actively.
2	Use	TCP client	none	If the TCP server is selected, the device may fail to reconnect after disconnection.
3	None	TCP server	Modbus TCP to RTU	The user software is Modbus TCP and the user device is Modbus RTU. And the Modbus TCP master station.
4	None	TCP client side	Modbus TCP to RTU	The user software is Modbus TCP and the user device is Modbus RTU. And Modbus RTU is the master station.
5	Use	TCP client side	Realcom protocol	When the multi-serial port server serves as the TCP client and the virtual serial port is used, the Realcom protocol is

				recommended.
6	None	TCP server	Telnet protocol	This mode applies to monitoring device serial ports when connecting to the Zlan serial port server over Telnet.
7	None	TCP client side	None	It is suitable for connecting a cloud with a large number of devices. In general, the cloud is a server with a public IP address on the Internet.
8	None	TCP server	None	Applicable to devices and computers on the same local network, local monitoring, no need to cross the Internet communication.

5.1. Virtual Serial Port Mode

If the user software uses the COM port to communicate, it must use the virtual serial port mode. Including some PLC software, configuration software, instrument software and so on.

Then see if the monitoring computers and devices are on the local network：

a) If the computer is a server with a public IP address leased on the Internet, the device must use TCP client mode to connect the device to the server. In this case, you can select 2 and 5 in Table 6. If multiple serial servers are used, you must select 5.

b) All in the local network (can ping each other), it depends on whether the host computer actively queries or the device actively sends data. If the device acts as the TCP client to send packets, mode 2 must be used. Otherwise, mode 1 can be used.

5.2. Direct TCP/IP communication mode

If the Modbus TCP protocol is not required and the virtual serial port is not required, the user software may directly communicate with the network port of the serial port server for TCP/IP communication, and the serial port server converts TCP/IP data to serial port data and sends it to the serial port device.

Generally, the user of this kind of usage develops the host computer network communication software by himself, which integrates the analysis of the serial communication protocol of the device. This method is more flexible and efficient than virtual serial port. Correspond to 7 and 8 in Table 4.

The section "3.4.6 TCP Communication Test" mainly describes how to communicate when the serial server acts as a TCP server. Here we will talk about how TCP clients, UDP mode, and multi-TCP connections communicate with computer software. The Ttest computer software is based on SocketTest (software that mimics user TCP/IP communication) as an example.

Zlan serial port to network port module complies with the standard TCP/IP protocol, so any network terminal that complies with the protocol can communicate with the serial port server, Zlan technology provides a network debugging tool (SocketDlgTest program) to simulate the network terminal to communicate with the serial port server.

For two network terminals (in this case, the network debugging tool and the serial server) to communicate, their parameter configurations must be paired.

5.2.1. TCP client mode

There are two working modes in TCP mode: TCP server and TCP client. No matter which mode is used, one side must be the server and the other side is the client, and then the client can access the server.

When the serial server is the client, there must be three mappings, as shown in Figure 25. (1) Working mode: The working mode of the serial port server is the server mode of the network tool corresponding to the client. (2)IP address: The destination IP address of the serial port server must be the IP address of the computer where the network tool is located. (3) Port: The destination port of the serial port server must be

the local port of the network tool. After this setting, the serial port server can automatically connect to the network tool and send and receive data after the connection is established.

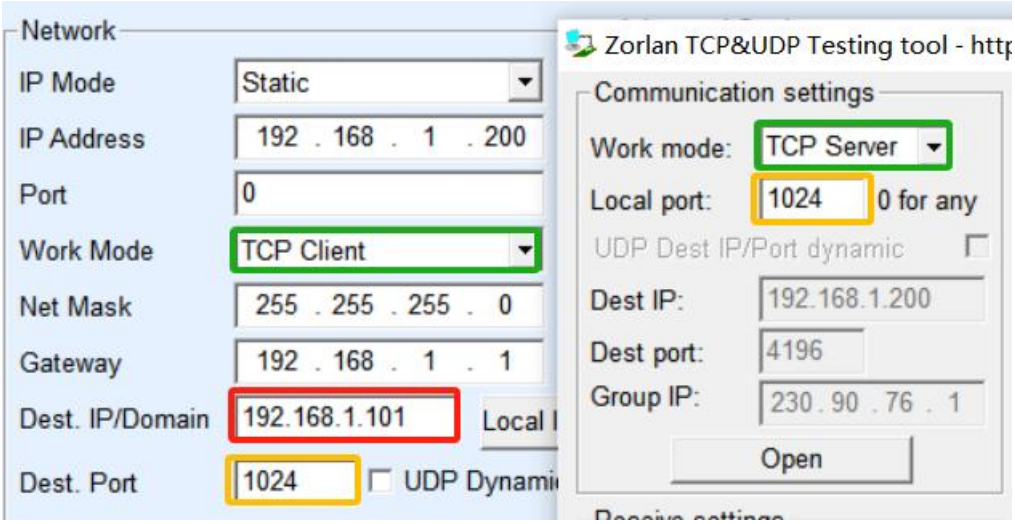

Figure 25 The serial server serves as the client

5.2.2. The client connects to multiple servers

When the ZLAN serial port server serves as the TCP client, seven destination IP addresses can be connected at the same time. Data sent from the serial port is sent to all seven destination IP addresses at the same time. If there are not that many servers, the rest of the destination IP is vacant. Its use is as follows：

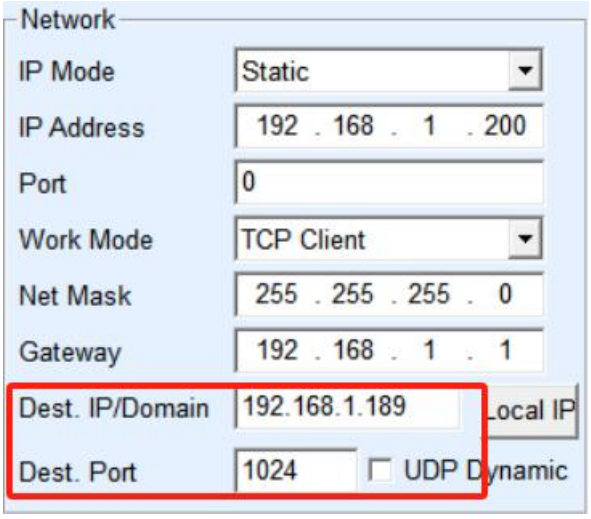

Figure 26 1st destination IP and terminal

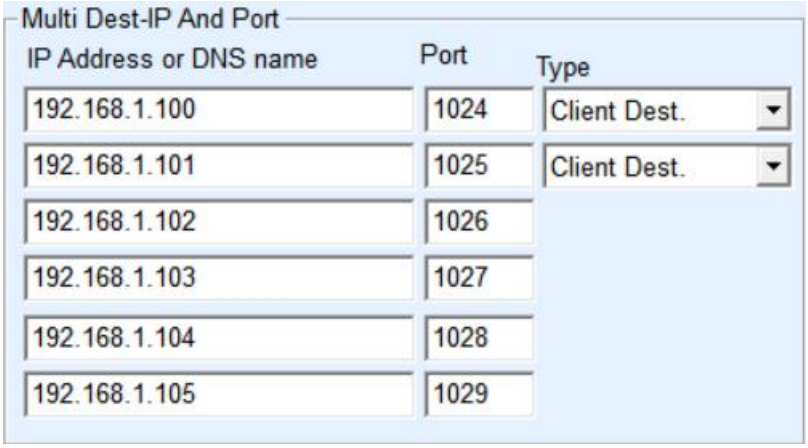

Figure 27 left 2～7 IP and terminals

The first IP address is set in the device setup screen shown in Figure 27, where the first IP address can be a domain name. Remaining 2 to 7 destination IP addresses On the device Settings screen, click More Advanced Options to open more advanced options.

All seven destination IP addresses can be automatically connected after being set. If the destination IP addresses fail to be connected, they are reconnected repeatedly after the Disconnection time.

5.2.3. TCP server mode

当 When the serial server acts as the server, there are also three mappings, as shown in Figure 28, which are not explained here. After this setting, click the open button of the network tool to establish a TCP connection with the serial port server. After the connection is established, you can send and receive data.

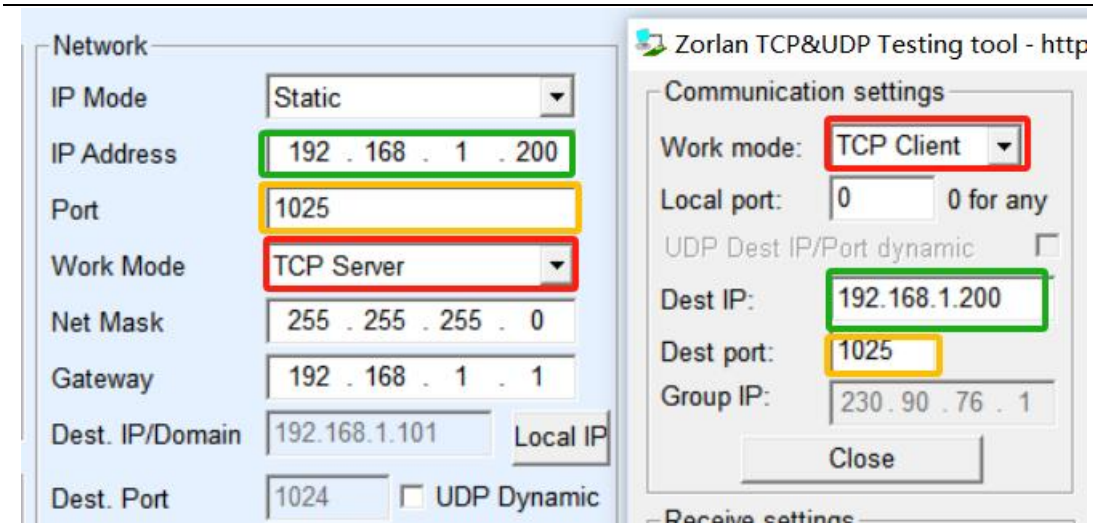

Figure 28The serial server serves as the server

When the serial port server serves as the server, it can accept 30 TCP connections at the same time. The data received by the serial port is forwarded to all established TCP connections. If you want to send data only to the TCP that has recently received network packets, you need to enable the multi-host function. For details, see 4.7.4 Multi-Host Function.

5.2.4. Act as both client and server

Zlan serial port server supports the device in the TCP client mode can also accept TCP connections, that is, also has the TCP server function.

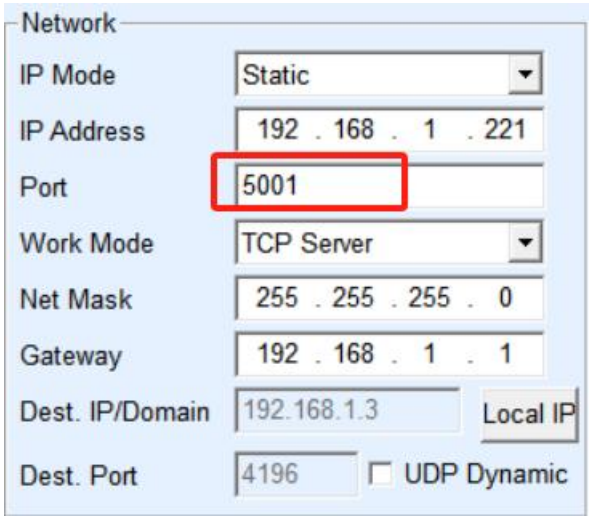

Figure 29 As both client and server

By default, when ZLVircom is configured, if the working mode is changed to

"TCP client" mode, the port (that is, the local port) will automatically become 0 (0 means that an idle port is randomly selected). In order to support the TCP server mode, the computer software must know the local port of the device, so a value needs to be specified here. As shown in Figure 29, the computer software can now connect to port 5001 of 192.168.1.221 for communication. In addition, the device acts as a client and connects to port 1024 of 192.168.1.189. Note that because the local port 5001 is occupied by the server, the local port used by the client is port +1000, that is, the software on 192.168.1.189 sees that the incoming port of the device is 5001+1000=6001.

5.2.5. UDP Mode

In UDP mode, parameter configuration is shown in Figure 30. The left side is the configuration of serial server in ZLVircom, and the right side is the setting of network debugging tool SocketDlgTest. First, both must be in UDP working mode. In addition, the destination IP address and port of the network tool must point to the local IP address and port of the serial port server. The blue arrow indicates that the destination IP address of the serial port server must be the IP address of the computer where the network tool is installed, and the destination port of the serial port server must be the local port of the network debugging tool. These network parameters must be configured to ensure two-way UDP data communication.

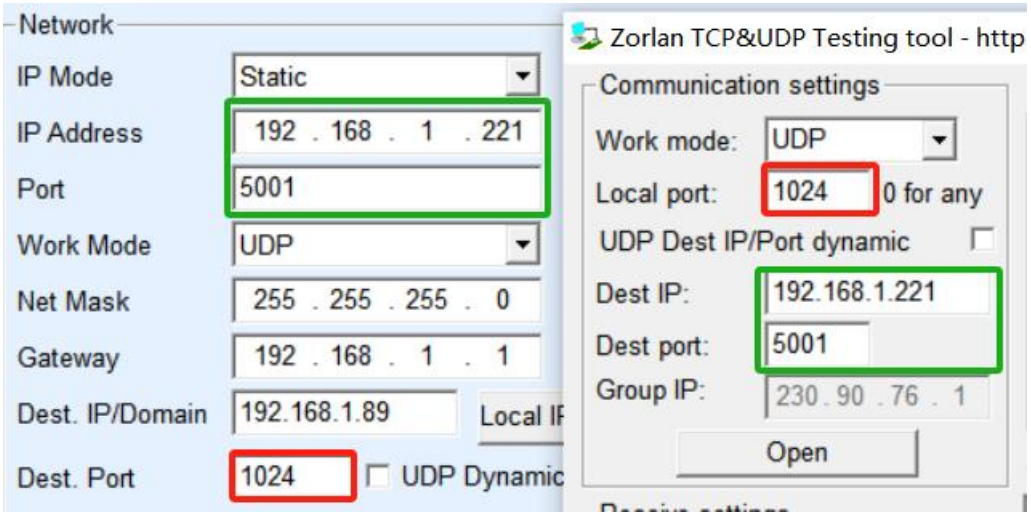

Figure 30 UDP mode parameter configuration

5.3. TELNET Agreement

When you log in to the serial port server using some Telnet tools, double-characters are displayed. For example, when you log in to the system using Telnet in windows. In this case, the conversion protocol needs to be selected as the TELNENT protocol. After you log in through Telnet, there are no more double characters.

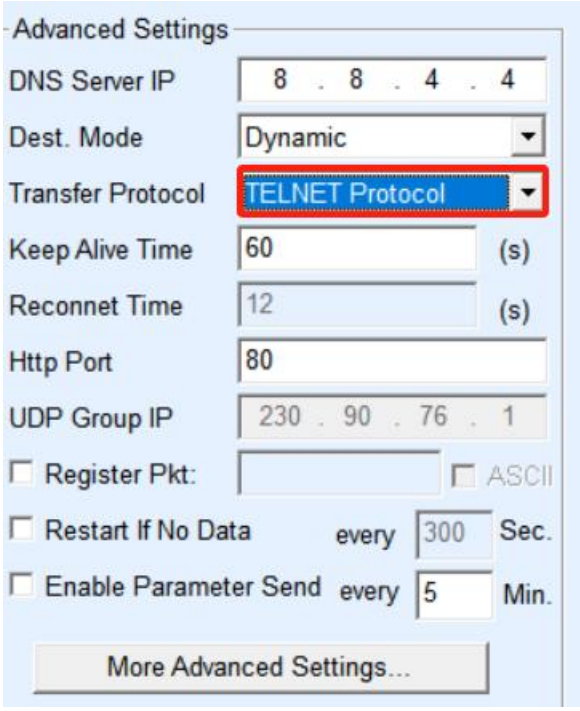

Figure 31 TELNET protocol

If the TAB key fails to be associated and two characters are displayed when secureCRT is used, it indicates that secureCRT is not in one character at a time mode. In this case, you only need to change the device port to 23, because the Send SGA (Only 23 ports) function is automatically selected in secureCRT. Therefore, when our device is on port 23, it will automatically enter the mode of "one character at a time".

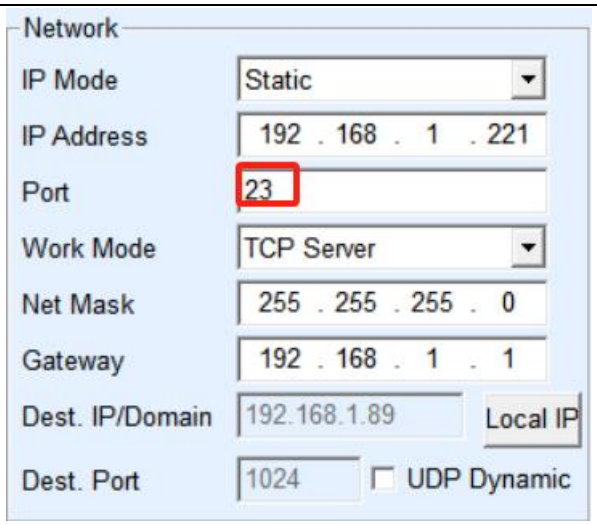

Figure 32 Terminal is 23

For details about how to modify secureCRT, kindly review file Precautions for Monitoring the Serial Port in Telnet Mode.

6. Device commissioning

6.1. Physical network connection

The module can be connected to the 10M/100M switch or directly connected to the computer network port using the crossover network cable or the direct network cable.

6.2. Network TCP connection

If the device dynamically obtains IP addresses, do not directly connect to the network port of the computer. Because there is no DHCP server to use (generally DHCP servers are routers in the LAN). Therefore, specify the IP address for direct connection. The computer also needs to specify a fixed IP address.

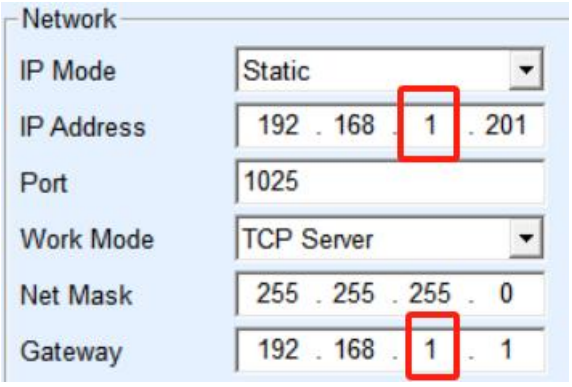

Figure 33 configurate into the same network segment

Whether connected directly or through a switch, when configured with static IP, the device and the computer need to be on the same network segment (unless communicating across gateways), as shown in Figure 33.

Since ZLVircom supports cross-network segment search and configuration, the IP addresses that can be searched but cannot be communicated are generally not well configured. In this case, ZLVircom can be used to configure devices on the same network segment.

After the configuration is complete, you can use the steps in 3.4.6 TCP Communication Test or 3.4.7 Virtual Serial Port Test to see that the network port indicator is steady green when a TCP connection is established. The green light can also be seen through ZLVircom. For example, in the device management list, if the TCP connection is listed as "established", the green light is on in the table, as shown in Figure 34, which can facilitate remote diagnosis.

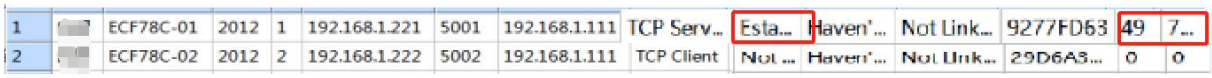

Figure 34 Connection status and data sending and receiving status

6.3. Data sending and receiving

When the network port is green on, data can be sent and received between the software and the serial port server. If the software sends data, the yellow indicator of the network port blinks for at least one second. Data is also output from the serial port of the serial port server, but whether the output data is correct depends on whether the correct serial port parameters (baud rate, data bit, stop bit, check bit) are configured.

The serial port device usually replies to the correct command delivery. Once there is a reply (the serial port sends data to the network port), the yellow light blinks. Otherwise, check the serial port parameters or whether the serial port cable is improperly connected.

In order to facilitate remote debugging, ZLVircom also supports remote viewing of data sent and received. As shown in Figure 34, TXD is the amount of data sent by serial port server. When refreshing the device list, if the value changes, it indicates that there is data sent, and the yellow light will blink. If you see the value of RXD changing, it indicates that the serial device has returned data. The yellow light will flash.

6.4. ZLVircom Remote monitoring data

In the case of using virtual serial port, ZLVircom supports real-time capture of data sent and received by virtual serial port. Convenient for users to debug the system, the use method is as follows:

Assume that the communication of the virtual serial port has been established according to the method of 3.4.7 Virtual Serial Port test. Now you need to monitor the data through the virtual serial port. Open ZLVircom's menu/Configuration/Software configuration/Open vircom configuration dialog box.

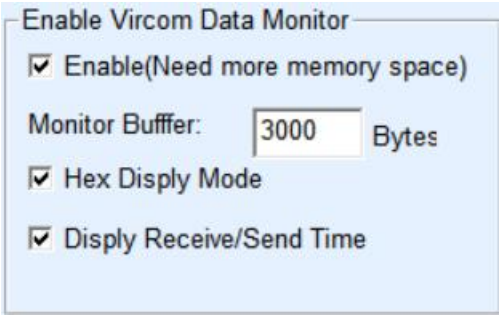

Figure 36 Enable ZLVirocm monitoring

Check the three options of enable monitoring, hexadecimal monitoring mode, and display data sending and receiving time, as shown in Figure 35. Then click OK. Assuming that data has been sent and received before, now select a virtual serial port to be monitored in the main interface, and then select Menu/View/Monitor, as shown in Figure 36.

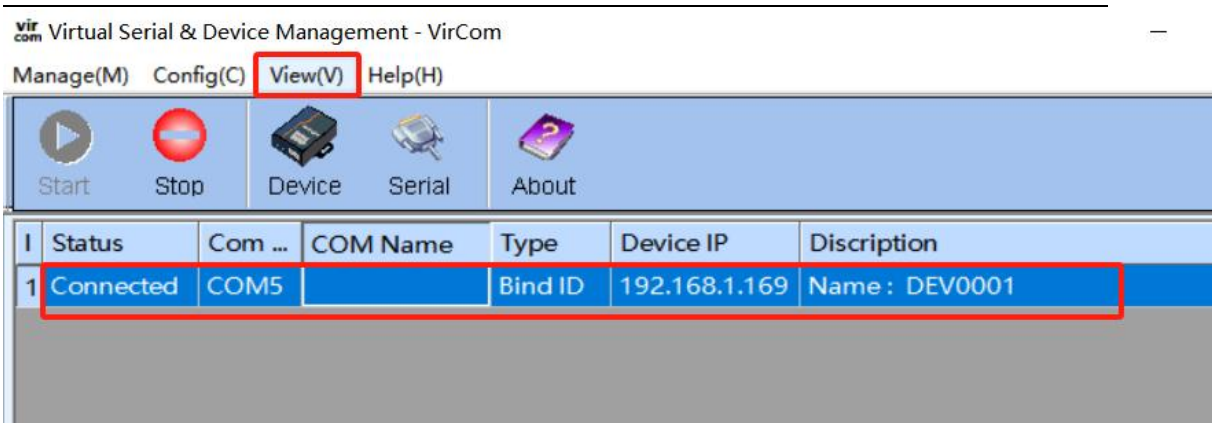

Figure 36 Open ZLVirocm monitoring

From the open dialog box, you can see the instructions sent by the host computer and the instructions returned by the device, as shown in Figure 39. This function can facilitate the field communication debugging.

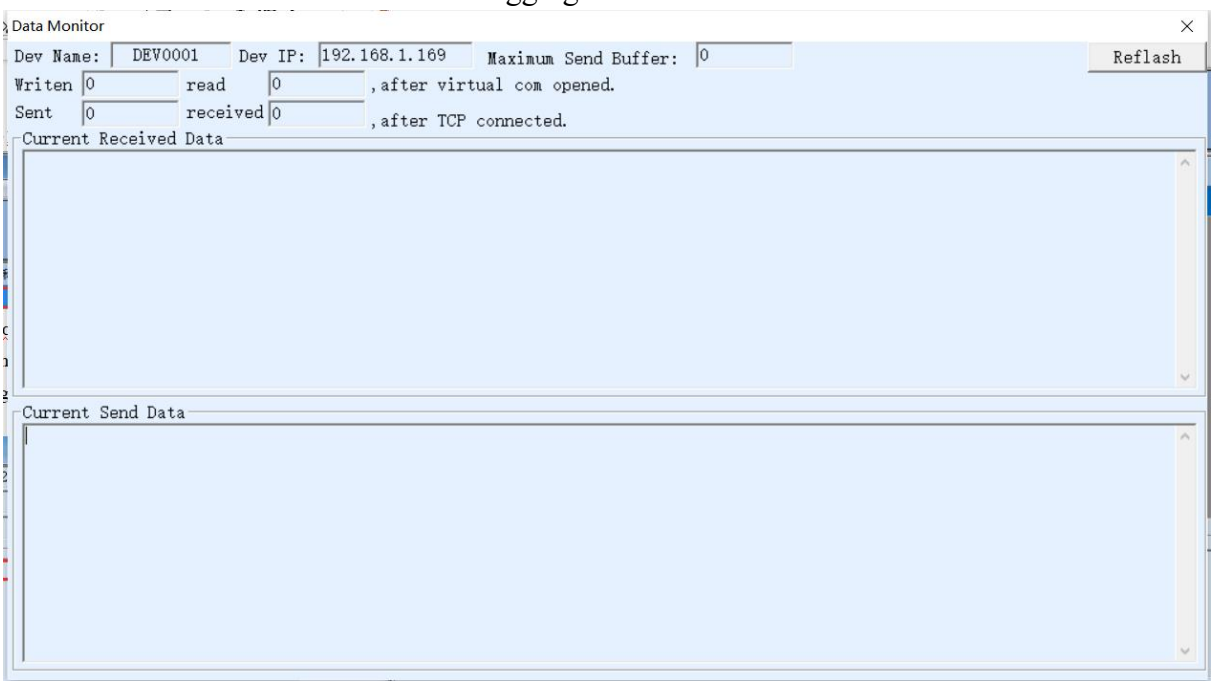

Figure 37 Monitor send and receive data

7. Modbus Advanced function

Serial port server with Modbus gateway function itself does not have station address and register, it is a communication bridge, it will generate Modbus RTU specification according to the user software to Modbus gateway in the Modbus TCP

instruction, function code, register number, register number, and output from the serial port. Think of it as a protocol "translator."

7.1. Enable the Modbus gateway

First of all, the serial port server should support Modbus gateway, that is, the function of "Modbus TCP to RTU" in the function supported by the device in Table 3 of the device setup dialog box should be checked.

By default, the serial port server is in common transparent mode. If you want to switch to Modbus gateway mode, select Modbus TCPRTU in Conversion Protocol. The device automatically changes Port to 502 (port of the Modbus server). The Modbus gateway is enabled.

When the serial port RTU device serves as the slave station, the Modbus TCP software on the upper computer connects to port 502 of the Modbus gateway, and the Modbus gateway needs to work in TCP server mode. If serial port RTU is used as the primary station, the Modbus gateway works on the TCP client, and the destination IP address is the IP address of the computer on which the Modbus TCP software is installed. The destination port is usually 502.

7.2. Storage Modbus Gateway

ZLAN5112 can save the contents of the read registers inside the gateway, so the speed of Modbus TCP queries can be greatly improved, and the performance is better when supporting multi-host access.

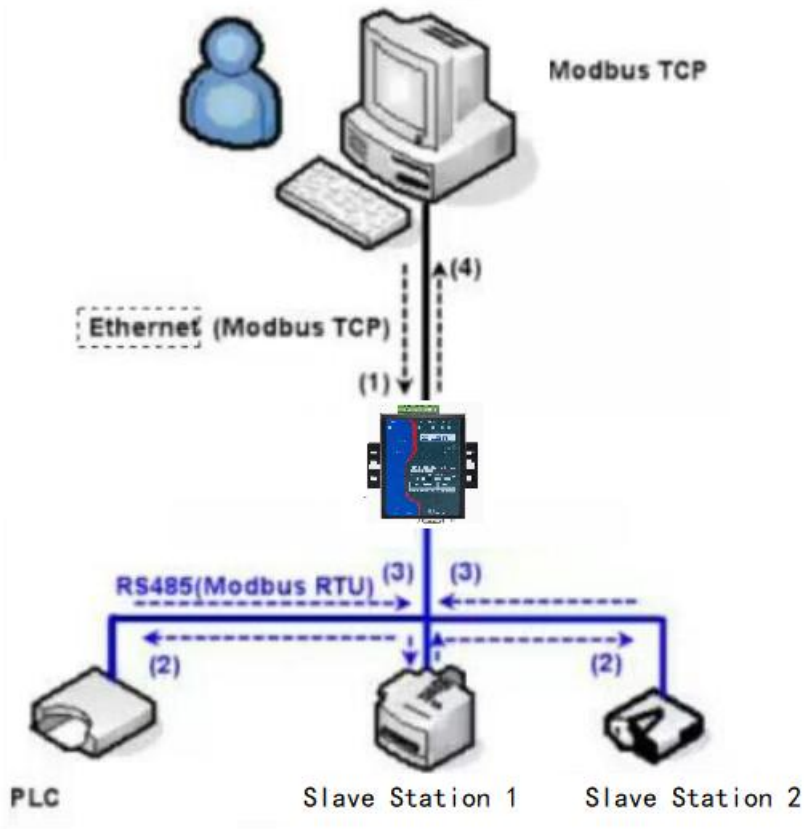

Figure 38 storage Modbus gateway working mode

As shown in Figure 38, the normal Modbus TCP data flow direction is (1) (2) (3) (4). That is, the Modbus TCP command is first converted into the corresponding command of Modbus RTU, and then the device responds to the Modbus RTU command to the Modbus gateway, and then the Modbus gateway is converted into Modbus TCP and sent to the monitoring host computer.

We know that Modbus TCP is a network communication, the transmission speed is very fast, generally within 3ms can answer, while Modbus RTU is RS485, generally only 9600bps speed, generally send and return an instruction at least 30ms. In this way, the query response time of the common non-storage Modbus gateway is relatively long. In addition, if there are a lot of upper computers to query data at the same time, then the serial port will be congested. If the network is like a highway and the serial port is like a single-log bridge, then the original way is to pass the traffic of the highway on the single-log bridge.

Register-saving Modbus gateway solves the above problems. It can temporarily save the register data obtained by the query in the Modbus gateway, so that when the Modbus TCP query comes, the Modbus gateway can immediately return the command, and really bring the fast characteristics of Modbus TCP into play. On the other hand, ZLAN5112 can actively send instructions from the serial port to automatically update the data content of the currently saved register and save a copy of the latest register value.

In addition, the module is also a fully automatic configuration free Modbus gateway, the user does not need to configure the required register address, function code, slave station address, etc. The ZLAN5112 automatically identifies and dynamically adds these registers according to Modbus TCP instructions from the network port.

When monitored by multiple computers, ZLAN5112 can show good response speed, no matter what baud rate the serial port is, it can generally give the host response data within 3ms. And it shows a good speed of real-time update of serial data.

The register-saving Modbus gateway is the real Modbus TCP to Modbus RTU, which really plays the advantages of fast Modbus TCP and multi-host query at the same time.

Notice When the serial port server serves as the TCP client, it does not have the storage function and automatically switches to the non-storage mode.

The features of storage Modbus are listed below:

1. The first Modbus TCP query instruction is non-stored. Because you must wait for the RTU device to return the data slowly before you can return the register contents to the network port.
2. If a specific command is no longer queried by the host computer on the network within 5 seconds, the command is automatically deleted and no longer sent from the serial port to the RTU device.
3. ZCurrently, 10K Modbus cache can be stored, and about 500 instructions can be stored at the same time for ordinary single-register queries.
4. When multiple instructions are queried at the same time, they are sent in order of

priority. The first instruction is sent and the first instruction is answered. 485 anti-conflict time (refer to the multi-host part) The second instruction is sent. . Do not return to the first command until the last command has been answered.

7.3. Disable the storage-type function

Although the memory Modbus has a fast response speed, some users do not want the RTU device to not want to receive a large number of query instructions, which affects the internal processing speed of the instrument. At this point, you can disable the storage function.

To disable storage type, click the "More Advanced Options" button in the "Parameter Configuration" dialog box and select Simple Modbus TCP to RTU. Then go back to device Settings and click Modify Settings.

Notice When you configure the conversion protocol in Web mode, it is a non-storage Modbus gateway by default.

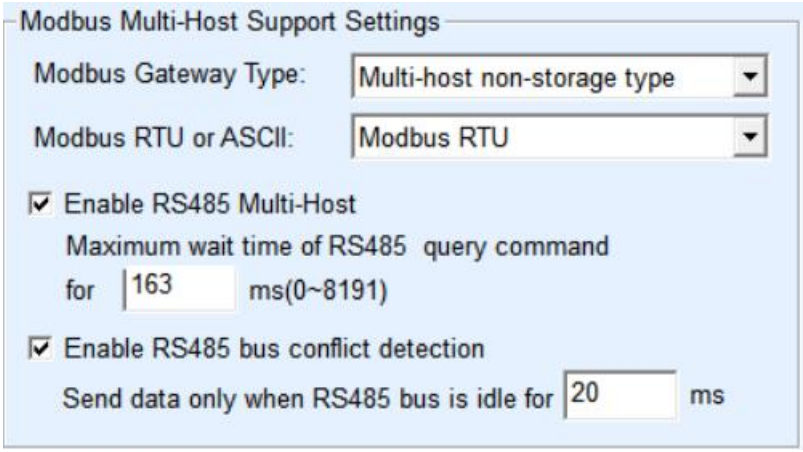

Figure 39 new version zlvircom setting

7.4. Multi-host function

As shown in Figure 39, "RS458 multi-host support" and "RS485 bus collision detection function" are the multi-host functions of Zolan. They are generally enabled and disabled simultaneously. After this function is enabled, the device whose protocol is converted to Modbus TCP has the function of a storage Modbus gateway;

otherwise, it is a non-storage Modbus gateway. If the conversion protocol is none, the user-defined RS485 protocol can generally also have the function of a serial port device accessed by multiple hosts at the same time, which can not be achieved in a pure RS485 network, because multiple master stations send conflicts on the RS485 bus at the same time. The multi-host of Zolan serial port server can "coordinate" the RS485 bus to achieve the purpose of multi-host access.

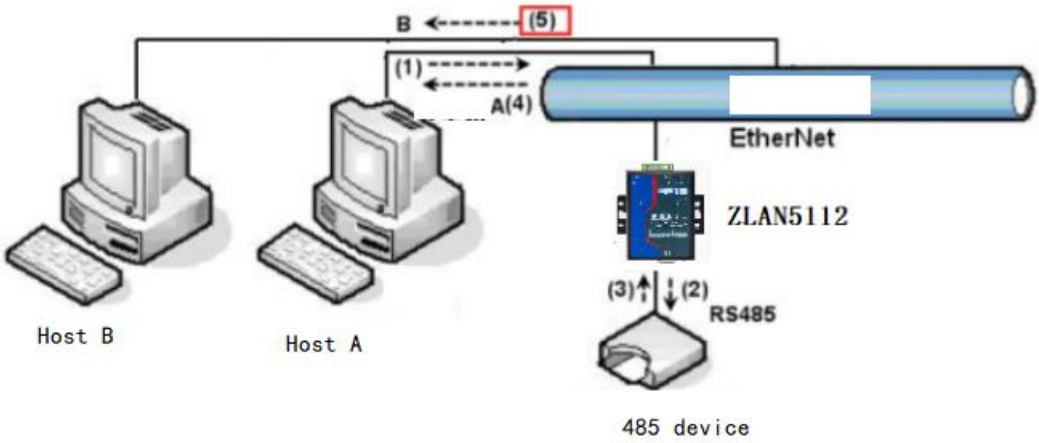

Figure 40 Multi-host function demonstration

As shown in Figure 40, in normal mode, when two hosts: host A and host B are connected to the serial port server at the same time, host A sends (1) instruction, the RS485 device receives (2) instruction, and the RS485 device returns (3) instruction, but the port end of the serial port server sends (4) to host A and (5) to host B at the same time. Because host B did not send the query, but it also received the reply instruction (5), host B may generate a communication exception error. In multi-host mode, there will be only instructions (4) and no instructions (5), because the serial server will automatically remember the host that needs to be returned, and only the command will be returned to the most recent communication host, host A queries will only reply to A, host B queries will reply to host B.

Another effect is that in the ordinary mode, host A and host B send data at the same time will produce a combination of instructions on the RS485 bus, so that it can not be recognized normally; In the multi-host mode, the serial port server can schedule the sequence of A and B in using the bus, so as to effectively solve the conflict problem of multi-machine simultaneous access.

If the conversion protocol is set to None, the multi-host function is disabled by default. To enable multiple hosts, click More Advanced Options in the Device configuration dialog box and select RS485 Multi-Host Support.

7.5. Multiple host parameters

The meanings of RS458 Multi-host support and RS485 bus conflict detection are described as follows.

图 41 RS485 多主机支持

The response timeout period of an RS485 command is as follows: Indicates the maximum interval between sending the command and receiving the response from the serial port on the server. The value must be greater than the actual maximum interval. Because if it is determined to be a timeout, the next instruction will be sent.

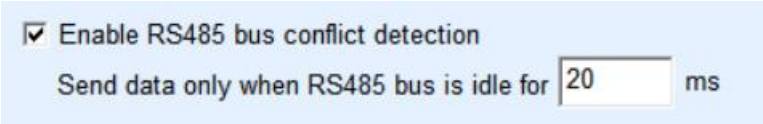

Figure 42 RS485 Conflict prevention idle time

RS485 bus collision time: indicates how many milliseconds the serial port server waits to send the second command after receiving the reply of the first command. This parameter actually defines the speed of instruction rotations. The recommended value is above 20ms. Generally, you do not need to change the value of Maximum Wait time 3 seconds.

When the user uses ZLVircom to select the conversion protocol as "after Modbus TCP to RTU", ZLVricom will automatically check the above two enable boxes (unless the user manually enters the advanced option to remove), and the above two times will be automatically configured according to the baud rate. However, if the Modus command is long or the protocol is converted to None, you need to manually set the two parameters.

The following describes the recommended values for setting the above parameters:

1. Figure 42 shows the RS485 bus collision prevention time. Generally, you can set it to twice the value of Packet Interval in the lower right corner of the parameter configuration page, but the value cannot be less than 20.
2. Figure 41 shows the "RS485 command response timeout time", which is generally determined by the length of the command to and from the reply. If the send command is N bytes and the reply is M bytes, the recommended value is "packet interval" x (N+M+5) +100.

7.6. Non-storage multi-host

Some places must use non-storage Modbus, this is because when a certain event occurs to read the register data, but the read data is the data before the storage collection, which is logically incorrect, so it must also support non-storage Modbus collection. But on the other hand, it is also necessary to support multiple hosts at the same time, and you only need to select a non-storage modbus gateway type.

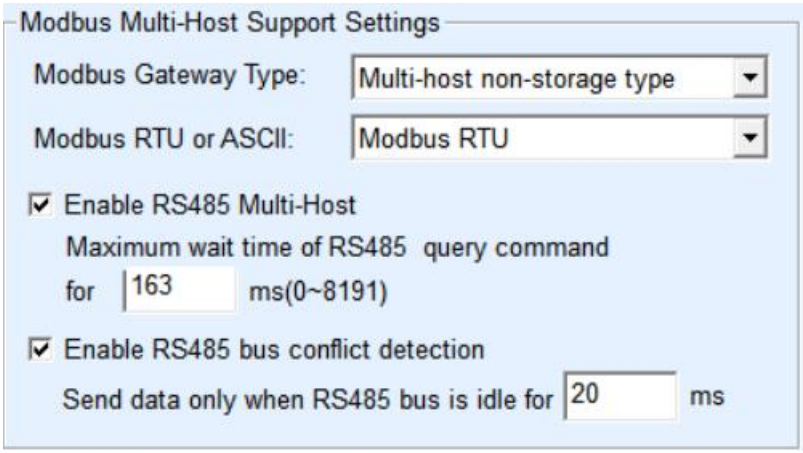

Figure 43 Multi-host non-storage setup

7.7. Modbus for Multi-destination IP Addresses

As shown in Figure 40, if the serial port device (RTU device) acts as the primary station and the network port device (Modbus TCP device) acts as the secondary station, there are multiple network port secondary devices at the same time. In this case, you can use the method described in the section about Connecting a Client to multiple servers to enable the serial port server to connect to multiple network ports simultaneously as a client.

In this case, the following functions need to be implemented: When the serial port RTU sends commands to multiple network port devices, the network port device identifies whether to send commands to itself through the Slave ID field, and only the network port device corresponding to the Slave ID responds. After the network port response is sent to the serial port server, it is converted into an RTU command and sent to the RTU device through the serial port.

At this time, it should be noted that the two check marks "RS485 bus collision prevention time" as shown in Figure 42 and "RS485 command response timeout time" as shown in Figure 41 need to be removed. Otherwise, the preceding forwarding function cannot be implemented.

Another application method is: Although the serial port server is connected to multiple network port devices as a Client, the RTU device is not the master station, but the network port device sends first, and the RTU device replies (as the slave station). In this case, the two checkboxes of RS485 bus Conflict Prevention time and RS485 command response timeout time need to be selected, so that multiple hosts can access an RTU device at the same time.

For the new version of ZLVircom, the above Settings can be completed by directly selecting "the device does the slave station for the client" in the Modbus gateway type.

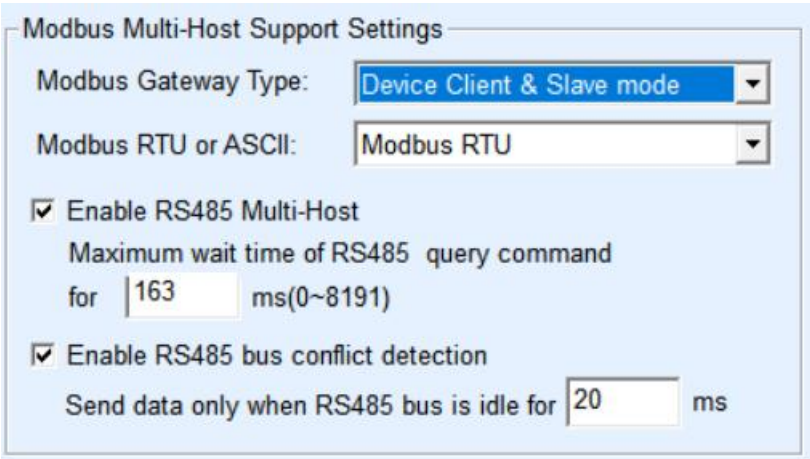

Figure 44 new version of zlvircom setting

8. Register the packet and heartbeat packet

Registration packets and heartbeat packets are a feature suitable for communication between devices and cloud software.

8.1. Register package

The definition of the registration package is that when the computer software and the serial port server module (hereinafter referred to as the module) establish a TCP connection, the module will first send a string of codes to the software, so that the software can know which module is communicating with itself. This string of codes is the registration package.

The registration package is very suitable for the monitoring of the Internet of things, because the cloud software generally runs on the public network server on the Internet, and the modules are scattered in various collection and monitoring points. How to make the cloud software identify the module is very important, and it is necessary to achieve Internet of Things communication.

Shanghai Zlan serial server provides the following registration methods.

8.1.1. Send MAC addresses on the connection

Send MAC address on connection: When the module is connected to the cloud, it sends its mac address to the cloud. Since the MAC address is unique, the device can be uniquely identified. This method is simple and effective because it does not require registration package writing for each device. How to use: click "More Advanced Options" in the device Settings dialog box, find "Send MAC address when TCP is established" in the upper middle, tick in the front, and then return to the Settings interface, click "Modify Settings".

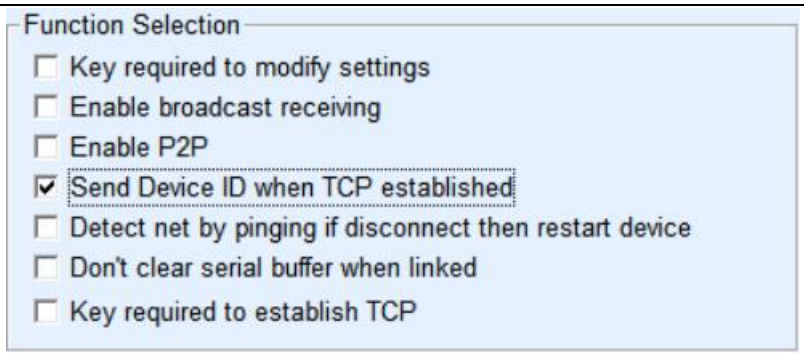

Figure 45 Send MAC address via connection

8.1.2. Realcom agreement

Realcom protocol is a mature protocol containing registration packets and heartbeat packets. Users can use this protocol to realize registration packets and heartbeat packets. To enable the Realcom protocol, perform the following operations: In the Device Settings dialog box, set Conversion Protocol to REAL_COM protocol. Note that the part of enabling the registration package must be left blank.

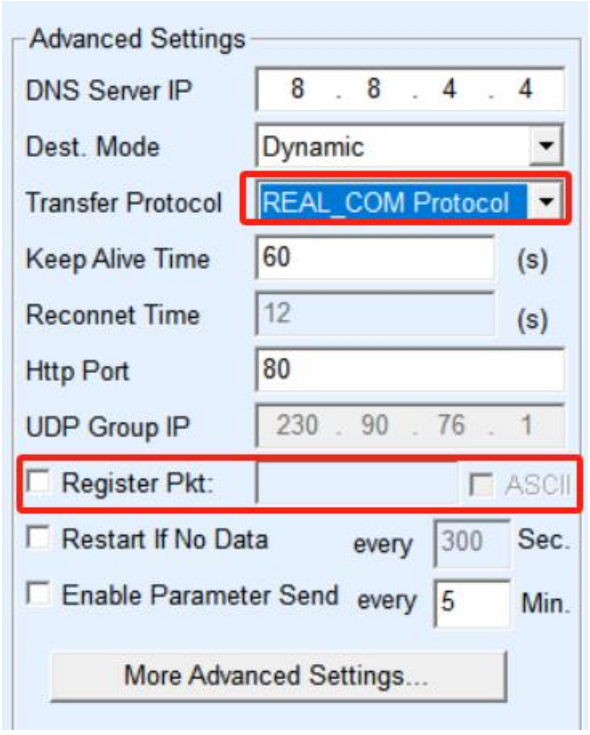

Figure 46Enable realcom protocol

After the Realcom protocol is enabled, it will not be transparent transmission communication, which has the following characteristics:

1. After a TCP connection is established between the device and the cloud, the device automatically sends a hexadecimal registration packet FA 07 13 02 FA 02 MAC[5] MAC[4] MAC[3] MAC[2] MAC[1] MAC[0] FA FF. MAC[5] to MAC[0] indicates the MAC address of the device.

2. When the device sends data to the network, it automatically adds the three-byte prefix FA 01 01.

3. At a keepalive interval, the device sends a 1-byte heartbeat packet of 00 to the software.

The REAL_COM protocol can be used as the registration package of the device because the registration package contains the MAC address. However, due to its fixed format, only cloud software can design the REALCOM protocol to be compatible with this approach.

8.1.3. Customize the registration package

The custom registration package mode allows the user to fill in any registration package format. The method is as follows: On the device Settings page, perform the following configurations：

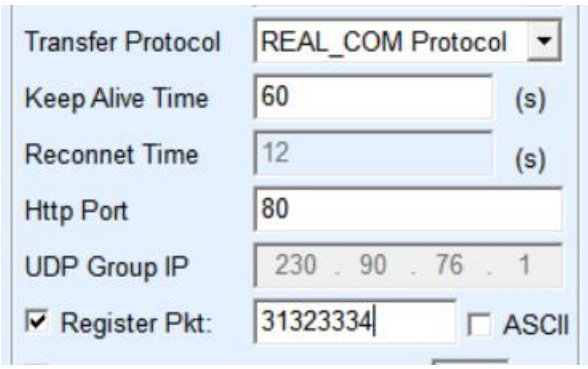

Figure 47 Set up register packet

The difference with the REAL_COM protocol is that the registration package is enabled and the registration package information such as 31, 16, 33, 34 is filled in. Note that this is in hexadecimal, which means that the actual data sent is the string 1234. If you want a string display, click the "ASCII" option next to it.

When the device is connected to the cloud software, it can automatically send the hexadecimal registration package of 31, 16, 33, and 34. This registration package method is more flexible, allowing the device to adapt to the existing cloud registration

package format; However, there is no wildcard such as MAC in the registration package, so you need to configure a different registration package for each device, which is complicated. The above two methods of sending MAC addresses and REALCOM are the same for each device, but the registration package is naturally different due to different MACs.

The maximum registered package length is 33 bytes. This mode supports registered packets and heartbeat packets in UDP mode.

8.1.4. Configuration files

Configuration files can be used for longer registration packages.

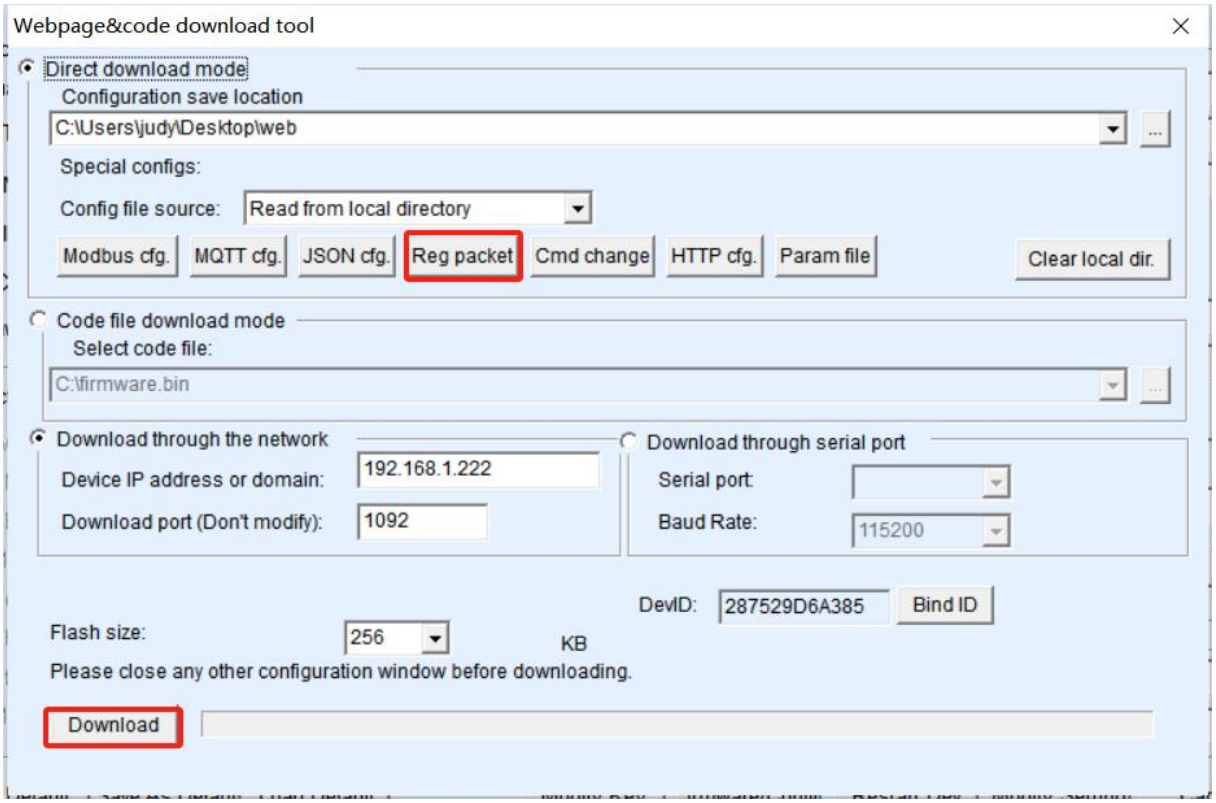

Figure 48 download register packet

Click the "Firmware and Configuration" button in the parameter setting dialog box to open the "Configure Web page/program download tool". In the dialog box that appears, select "Configure Web Page directory download" and select an empty directory as the directory for storing the configuration file. Then click the "Register

Package" button.

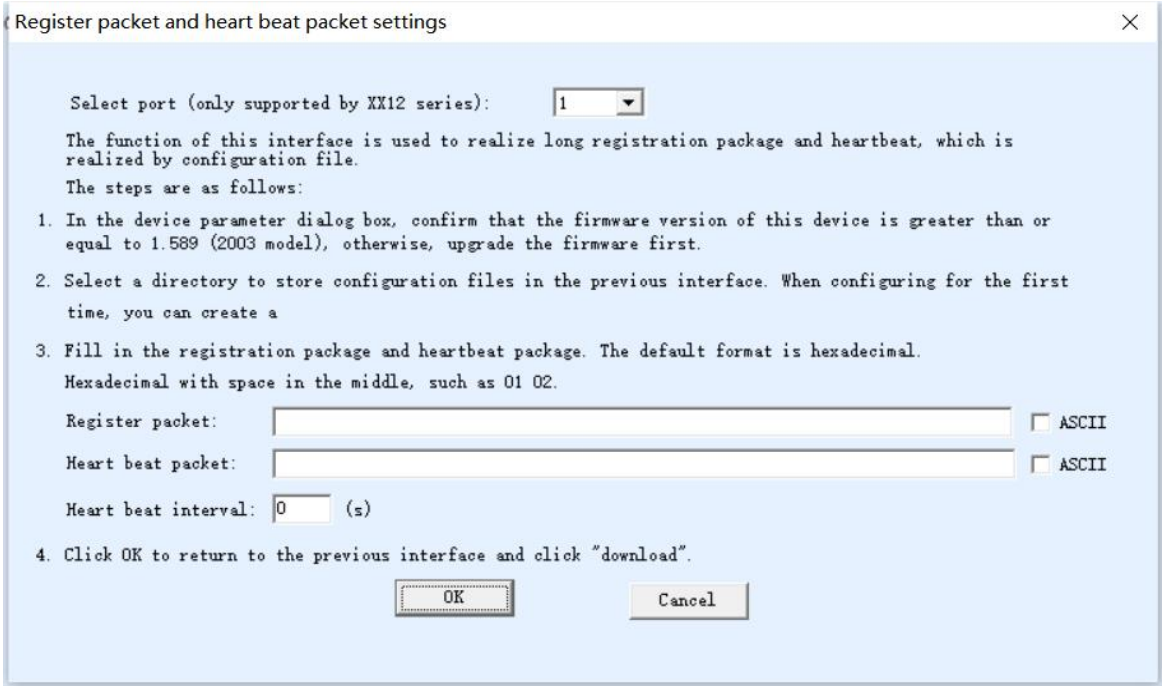

Figure 49 set up register packet

Here, set the registration packet, heartbeat packet, heartbeat packet interval and click OK. After returning to the previous screen, click the "Download" button to download the configuration file to the device.

8.2. Heartbeat packets

The heartbeat packet is used to detect whether the communication link is down. The method of implementation is that every once in a while the device sends a heartbeat packet data to the server software, and the data will be discarded after the server receives it, and will not be regarded as valid communication data.

The heartbeat packet has two main functions: first, it can let the upper computer software know that the device is in the active state; Secondly, if the device fails to send the heartbeat, the device on the TCP client will automatically re-establish the TCP connection, so it is a means to restore network communication.

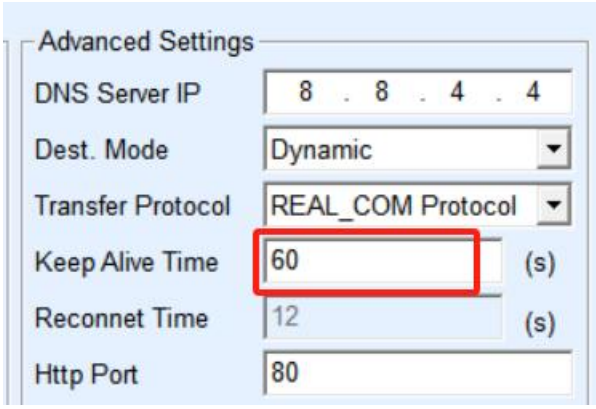

图 50 保活定时时间

As shown in Figure 51, the sending time of the heartbeat packet is set by the Keepalive Timing time.

8.2.1. Hidden heartbeat

Even if no heartbeat packets are set, the implicit heartbeat function is enabled on the Zlan device when the TCP client is deployed. Therefore, the implied heartbeat function means that the device sends data but the server does not receive the heartbeat data. Therefore, it can not play the first function of the heartbeat packet, that is, the function of the server to detect the activity of the device; However, because the device actually sends data, it can play the second function of the heartbeat packet, that is, the function of the device to detect whether the TCP connection is normal. Once the connection is detected, the TCP connection can be automatically re-established.

8.2.2. REALCOM agreement

As described in 8.8.1.8.1.2Realcom protocol, the REALCOM protocol can send a 1-byte data of 00 at every keepalive interval. This data is the heartbeat packet of the realcom protocol.

8.2.3. Customize the heartbeat packet

Fill in the registration package by following instructions in 8.1.3 Customizing the Registration Package. Then add heartbeat packets as follows: Click the "More Advanced Options" button in the device Settings, write the hexadecimal heartbeat

packet in the second line of the multi-destination IP and port, and change the option on the right to "Parameter packet Destination".

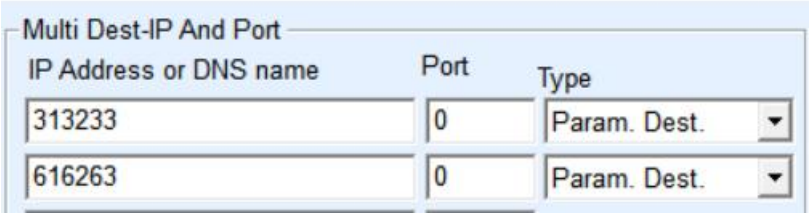

Figure 51Customize the registration package

The sum of the registered packet and heartbeat packet must be less than 33 bytes. The first line is actually the registration package.

8.2.4. Configuration files

For details, see how to use the configuration file of the registration package.

9. Modify network port parameters

Modifying parameters through the network port is to achieve the function of searching for devices and modifying device parameters, just like the zlvircom software does. This is done by managing devices and changing parameters through the network port of the serial server. It is suitable for users who want to integrate the search and configuration functions into their own software.

Modifying parameters through the network port is implemented via the "UDP Management Port Protocol", for example:：

1. The computer software sends a UDP broadcast packet with the destination port set to 1092 over the network. When a device receives this packet, it sends back its information to the computer software, achieving the purpose of searching for devices.
2. The computer software sends a UDP command to modify parameters to the device's port 1092, achieving the purpose of modifying the device's parameters.

A detailed introduction to modifying network port parameters can be found in the document "ZLan Networking Product UDP Management Port Protocol". It can also be achieved by using the device management function library of the 10 device

management function library directly.

10. Device management library

This feature is suitable for users who need to integrate device management functions into their own software.

The "UDP Management Port Protocol" has been integrated into the device management function library ZLDevManage. This is a DLL development library for the Windows platform, which can be called by various development tools such as VC, VB, Delphi, etc.

Detailed API interface documentation and VC calling Demo cases are provided. It can achieve device search, parameter modification, P2P function calls, etc.

The development library can be obtained from the ZLan official website: find "Device Management Function Library" on the http://zlmcu.com/download.htm page. For details, please refer to "ZLan WinP2p and Device Management Development Library".

11. MQTT Gateway

Please refer to the use of MQTT gateway function

《How to use MQTT gateway》

http://www.zlmcu.com/document/Usage_of_MQTT_Gateway.html。

《Zlan MQTT and JSON to Modbus Gateway Usage》

http://www.zlmcu.com/document/MQTT&JSON_to_Modbus.html；

12. Modbus RTU Turn JSON

For Modbus RTU to JSON and related JSON usage, see

JSON Data Acquisition Gateway

http://www.zlmcu.com/document/jsondata.html；

ZLan Server and the Use of Collection Equipment

http://www.zlmcu.com/document/zlancloud.html；
645 instrument in JSON format on the method
http://www.zlmcu.com/document/645_Instrument_JSON.html；
Zlan MQTT and JSON to Modbus Gateway
Usagehttp://www.zlmcu.com/document/MQTT&JSON_to_Modbus.html；
Supports more than 1000 bit to JSON.

13. Modify serial port parameters

Users can read and set parameters by sending commands to the serial port server's serial port. This is suitable for users who need to control and configure products at the chip or module level through serial port. The parameters that can be set include: IP address, baud rate, device name, working mode, etc. After the new parameters are set, the serial port server can be restarted through serial port commands.

The serial commands of Zlan have the following features:

1. Serial port commands use a 10-byte data preamble, so there is no need to distinguish between communication data and commands by pulling down or pulling up additional configuration pins, nor is there a need to switch between command mode and communication mode, making usage more flexible and convenient.
2. The command set includes various command formats such as saving parameters, not saving parameters, and restarting the device.
3. It can implement a variety of applications, such as reading the MAC address of the serial port server, or modifying the operating mode of the serial port server. For example, when switching from TCP server to TCP client mode, it can actively connect to the server; when switching from TCP client to TCP server mode, it can disconnect from the server connection.

Please refer to "Serial Port Parameter Modification and Hardware TCP/IP Stack Protocol" for detailed operation methods on modifying serial port parameters.

14. Remote device management

Remote device management refers to the ability to maintain and manage devices through ZLVircom software, including restarting devices, modifying parameters, and updating firmware. This feature is suitable for users who manage devices through ZLVircom.

For ZLVircom software, as long as the device can be found in the device list, remote management can be performed. Remote management of devices can be divided into the following situations:

1. Automatic search: Under the same switch, regardless of whether they are on the same network segment, the way ZLVircom searches for devices on a computer is: ZLVircom sends a broadcast query (all devices receive the query and reply with their parameters to the ZLVircom tool. This method searches for all devices at once.)

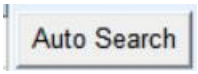

Figure 52 automatic search

2. Manually add: There are two situations:

Figure 53 manually added

   a) Large routers divide the network: In some large networks, broadcast packets are divided by routers, so broadcast packets cannot reach the device end, but pinging the device IP is always through. At this point, manual addition is generally needed to solve the problem. The method of manual addition is to click "Manual Addition" in the "Device Management" dialog box to add the first and last IP for individual queries of the device.
   b) Public network server queries internal network devices: Serial servers are in the internal network and operate in TCP server mode, while zlvircom is on a server with a public IP. At this point, a UDP port mapping of 1092 needs to be set up on the router of the network where the device is located, mapping to the IP of the device, and then zlvircom manually adds this device, with the IP being the public IP of the device end.

3. TCP Client: When the device acts as a TCP client, it will initiate a TCP connection to the destination IP (116.15.2.3) on port 4196. Once the connection is established, it will automatically send its parameter system to the destination's UDP port (not the TCP port) every keep-alive time interval, allowing zlvircom to detect the device on this computer (116.15.2.3). If the destination port is not 4196, you will need to modify zlvircom's default parameter receiving port by changing the menu/config/software configuration/default listening port. After starting zlvircom, if a TCP port conflict message appears, ignore it and continue with the execution.

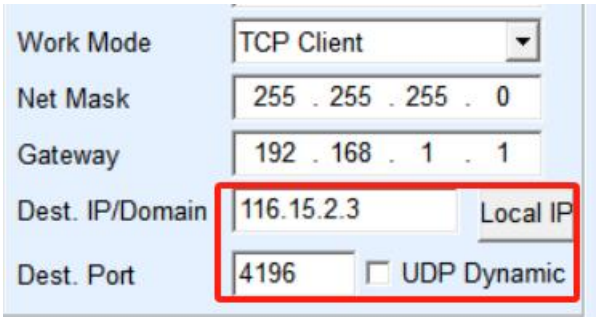

Figure 54 client

4. Scheduled transmission parameters: Even when in TCP server mode, the serial server can select the "Scheduled Transmission Parameters" feature to send parameters every 5 minutes to the destination IP (here, 116.15.2.3) on the destination port. The zlvircom on this server's port can receive parameters and manage these devices.

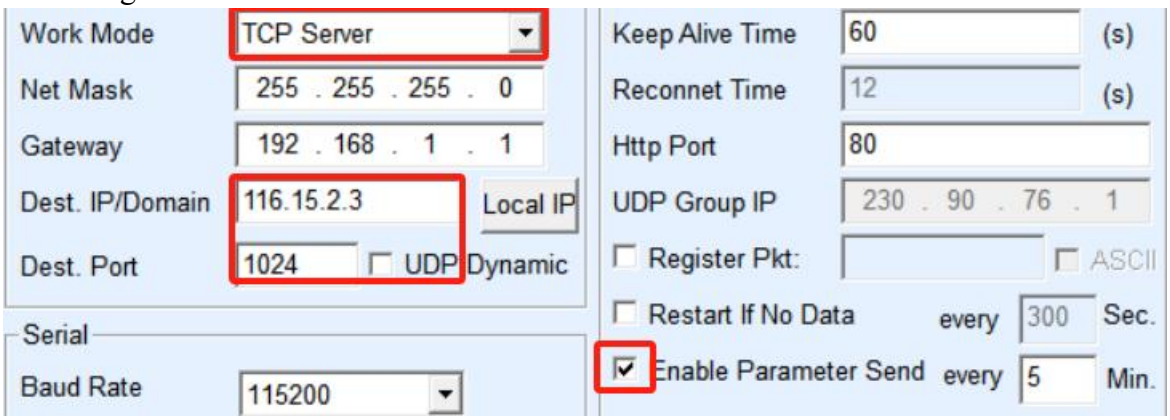

Figure 55 Timing transmission parameters

To facilitate the identification of devices, if remote management is required,

please name the device something easy to remember.

5. Device Operation Management: After enabling the cloud management feature on the device side, you can see the device on the designated platform equipped with Zlan Device Cloud Management. You can perform device configuration, firmware upgrade, configuration download, and more.

15. Firmware upgrade method

Different models can upgrade their own programs, but not each other's. You can use this method to upgrade the firmware of devices found in the device list by automatic search, manual add, or P2P search.

1 Get the firmware file for ZLAN5112 from Zlan, such as 1.445(2012).bin.

2 In the ZLVircom tool, first search the device that needs to be upgraded, and then enter the device parameter editing dialog box. First click "Restart Device" once.

Figure 56 Restart button

After the device restarts, use the same method to search for the device and enter the dialog box again. Click the "Firmware and Configuration" button in the lower right corner of the dialog box.

Figure 57 Firmware and configuration buttons

3 Select the "Program File Download" option, as shown in Figure 58. In the program file, select the firmware file. The IP address of the serial port server is automatically filled in, and the module type/model is automatically selected. Then click Download.

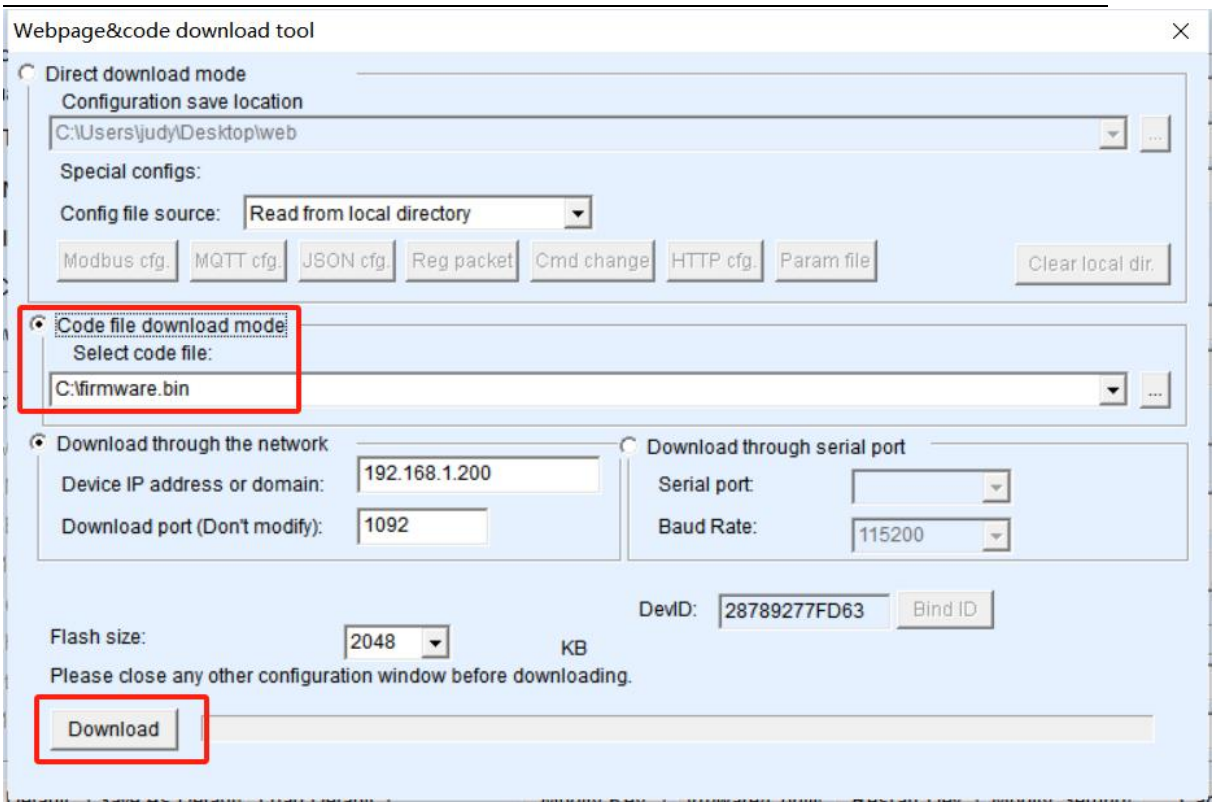

Figure 58 Firmware upgrade method of ZLAN5112

4 At this time, the download progress bar starts to move, and the download time is about 30 seconds. During the download process, you will see the ACT light of the device blink, and at the end of the download, you will see the LINK light blink several times. Then the program pops up "transmission completed LINK light flashing device do not power off" prompt box. Note: This is only the completion of transmission, write to the flash process takes about 3 seconds, at this time the LINK light will blink, during this period please do not power off.

5 5 After downloading the general program will automatically restart, generally do not need to power off. If the running indicator is blinking, stop the LINK indicator blinking for more than 30 seconds and power it on again.

6 6Web Configuration interface update: After the firmware upgrade, the configuration page inside the module also needs to be updated. Otherwise, the configuration cannot be configured through the Web, but the communication will not be affected. It is also possible not to download web pages without web configuration. To download the Web, change the "program file" download mode

to "Web directory download", as shown in Figure 59. In addition, select the root directory of the local web page as the directory of the web file to be downloaded (this directory can be obtained from Zolan), and click Download to download all files in the local web page directory to the internal file system of the device.

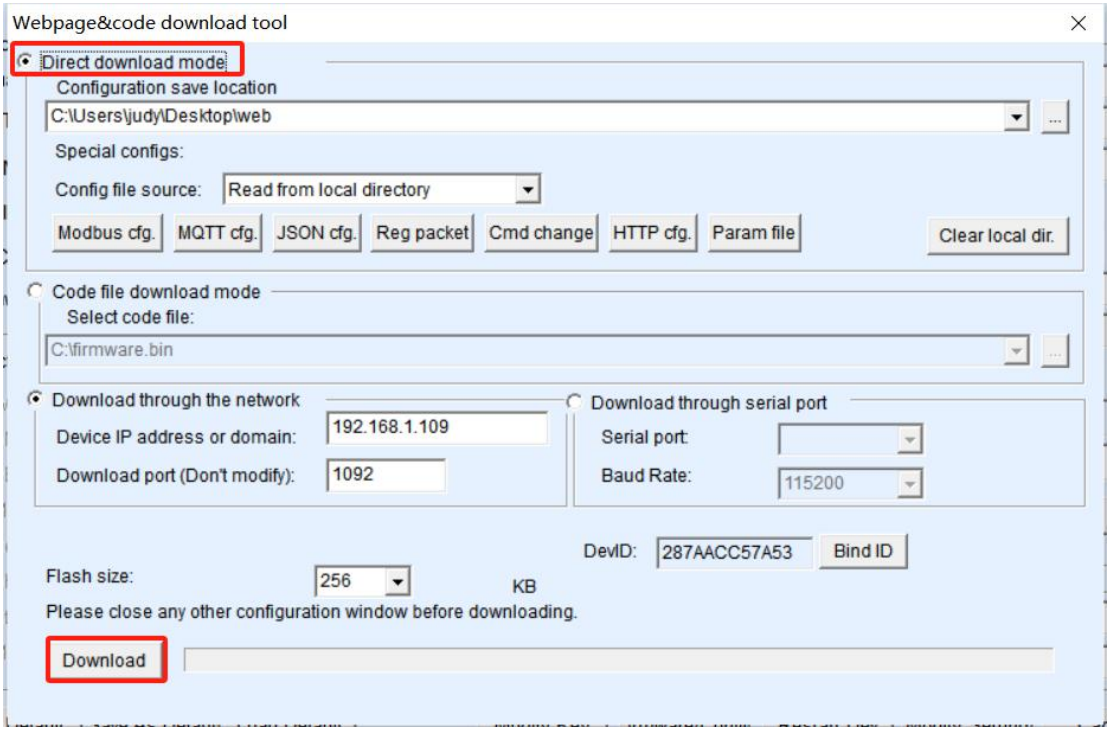

Figure 59ZLAN5112 web upgrade method

7 Look out：

7.1 If the message "Download failure" is displayed, the device will not be damaged. You can download the device again. In addition, when the LINK light blinks at the end of the download, do not power off, otherwise the device will be damaged.

7.2 Check the firmware version through ZLVircom to know whether the new firmware has been downloaded successfully.

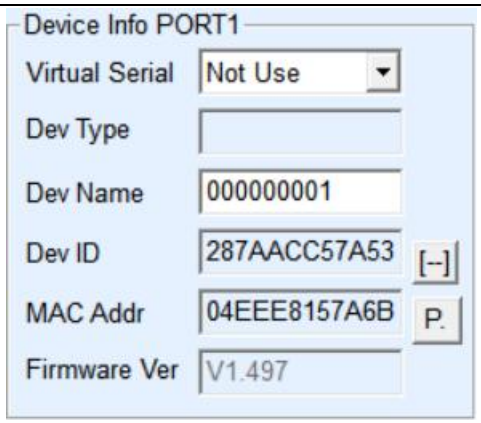

Figure 60 Checking the firmware version after the upgrade

16. Ordering information

Table 8 Ordering information

Model number	introduce
ZLAN5112	High performance iot gateway

17. After-sales service and technical support

Shanghai Zlan Information Technology Co., Ltd.

Address: Room 2001, No. 28 Yuanwen Road, Minhang District, Shanghai

Phone: 021-64165189

Fax: 021-64165200

URL：http://www.zlmcu.com

Email：support@zlmcu.com