ZLAN5212DI

Optocoupler isolation type

2 serial port server/Modbus gateway

User Manual

RS485 to TCP/IP

Modbus RTU to Modbus TCP

Copyright© Shanghai ZLAN Information Technology Co., Ltd.

All right reserved

Document DI: ZL DUI 20231219.1.0

CopyRight©2008 Shanghai ZLAN Information Technology Co., Ltd. All right

Version Information

The History of the revision to this document:

History

Date	Version	Document ID	Revising content
2023-12-19	Rev.1	ZL DUI 20231219.1.0	Release

Contents

1. Summary

ZLAN5212DI is a serial port server/IoT gateway with RS485 optocoupler isolation specially designed for industrial environments. It can effectively isolate the impact of interference on the RS485 bus on the operation of the equipment and ensure the stability of the equipment. Suitable for use in environments with interference on the RS485 bus. 9~24V wide voltage, terminal type power access, with shell ground wire.

5212DI has 2 RS485 interfaces and an Ethernet interface. Comes with mounting ears and rail buckles, supporting 3 installation methods. Small size and easy to install. The 2-way serial port of 5212DI can use the same IP address, and different serial ports are distinguished by ports; each serial port can also use different IP addresses. This design improves flexibility and manageability. It can not only save IP resources through shared addresses, but also enable independent network communication through different IP address.

Figure 1   ZLAN5212DI

ZLAN5212DI supports 2 stop bits, supports data bits less than 8 bits (such as 7-bit data and 1 stop bit), and supports baud rates up to 921.6Kbps.

ZLAN5212DI can convert 2 channels of TCP/IP protocol to RS485. The 2 channels work at the same time and can be configured to different baud rates. Data collection of 485 devices can be realized through the network. The network side can use a virtual

serial port to connect to serial port software or direct TCP/IP communication software.

ZLAN5212DI supports transparent transmission and Modbus RTU to Modbus TCP function, has storage Modbus gateway characteristics, and also supports MQTT protocol. Supports MODBUS RTU, 645 protocol conversion to JSON and scheduled upload function through MQTT/HTTP POST.

ZLAN5212DI can be applied to：

● Power electronics, smart meters and energy consumption monitoring;

● As an IoT gateway and a communication bridge between devices and the cloud;

● Remote monitoring and program downloading of various automated PLCs;

● Various configuration software and device communication interfaces;

● Equipment networking in the field of access control and security;

● Network information collection of medical equipment;

Typical application connections are shown in Figure 2. The original serial device RS485 is connected to the RS485 port of ZLAN5212DI, and 5212DI is connected to the computer through a network cable. The software on the computer establishes a connection with the 5212DI through TCP/IP, virtual serial port, or cloud server. After that, any data sent by the serial port device will be transparently transmitted to the computer software, and the data sent by the software to ZLAN5212DI through the network will also be transparently transmitted to the serial port device.

Figure 2   Connection diagram

2. Features

2.1 Hardware characteristics

ZLAN5212DI has the following features:

1. 1. Three options are available: guide rail installation, backplane mounting ear installation, and desktop mounting ear fixed installation. It is especially suitable for industrial guide rail installation. The narrow width can save installation space and make installation and disassembly more convenient.

2. Industrial grade power supply method: terminal block wiring method, convenient for use in industrial situations. Can be powered by a wide voltage of 9~24V. With shell ground wire, it can effectively guide interference and surge.

3. RS485 serial port optocoupler isolation, isolation voltage 1500V. It is suitable for strong interference environments where ordinary serial servers cannot work properly. Effectively isolate the ground circuit between the serial port device and the serial port server.

4. Rich panel indicators facilitate debugging: In terms of connection, there are not only NET lights that indicate whether the network cable is connected, but also LINK lights that indicate the establishment of TCP connections; in terms of data indicators, there are "serial port to network port" and "network port". to serial port" independent indicator light. Convenient for on-site debugging.

2.2 Software function

1 The 2 serial ports can be configured with different baud rates and work independently without interfering with each other.

2 Supports IP "clone" technology: 2 serial ports can be distinguished by port or IP. Port differentiation: 2 IPs can be merged into one IP with different ports, suitable for networks lacking IP; IP differentiation: one IP can be transformed into 2 IPs with the same port, suitable for applications with fixed port numbers (for example, Modbus TCP port 502 needs to be fixed).

3 Supports TCP server, TCP client, UDP mode, UDP multicast. When acting as a TCP client, it also supports TCP server functions. As a TCP server, it supports up to 30 TCP connections, and as a TCP client, it supports 7 destination IPs.

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

5 Supports the function of sending MAC address on device connection to facilitate cloud management of devices.

6 Provides a secondary development package DLL development library for searching and configuring equipment on the computer side.

7 Supports Web browser configuration, supports DHCP to dynamically obtain IP, and DNS protocol to connect domain name server address.

8   Supports remote search of devices, configuration of device parameters, and device program upgrades in the cloud.

2.3 Advanced software features

The advanced soft functions of ZLAN5407M support including:

1   Supports Modbus gateway function and supports Modbus RTU to Modbus TCP. Can support storage type Modbus, can automatically collect device data and store it; also supports non-storage mode Modbus gateway; supports ZLMB configurable table Modbus gateway function.

2   Support multi-host function: In the one-question-one-answer query mode, the network port port is supported to allow multiple computers to access the same serial port device at the same time. It is also possible to implement multi-host applications in which one serial port is converted to multiple serial ports.

3   Support MQTT gateway function.

4   Supports Modbus RTU, Modbus TCP and 645 instrument to JSON protocols, and supports HTTP POST and HTTP GET formats to upload data.

5   Supports NTP protocol to obtain network time, which is used for serial port output and the latter is used for protocol content uploading.

6   Supports custom heartbeat package and registration package functions: it can facilitate communication and device identification with the cloud.

7   Supports the function of requiring password verification for TCP connection establishment to ensure connection security.

8   Supports HTTP data submission and distribution functions. The cloud can directly use the HTTP GET command to interact with the device's serial port data.

3. Technical Parameter

Table 1

Figure	
Interface：	485：Terminal

Power supply：	Terminal
Size：	L x W x H =88mm*62mm*33mm
Communicate Interface	
Ethernet:	10M/100M，2 KV surge protection
Serial：	RS485×2：485A、485B
Serial Parameters	

Baud rate:	300~921600bps	Parity：		None, Odd, Even, Mark, Space
Data size:	5～9	Flow control：		XON/XOFF，NONE

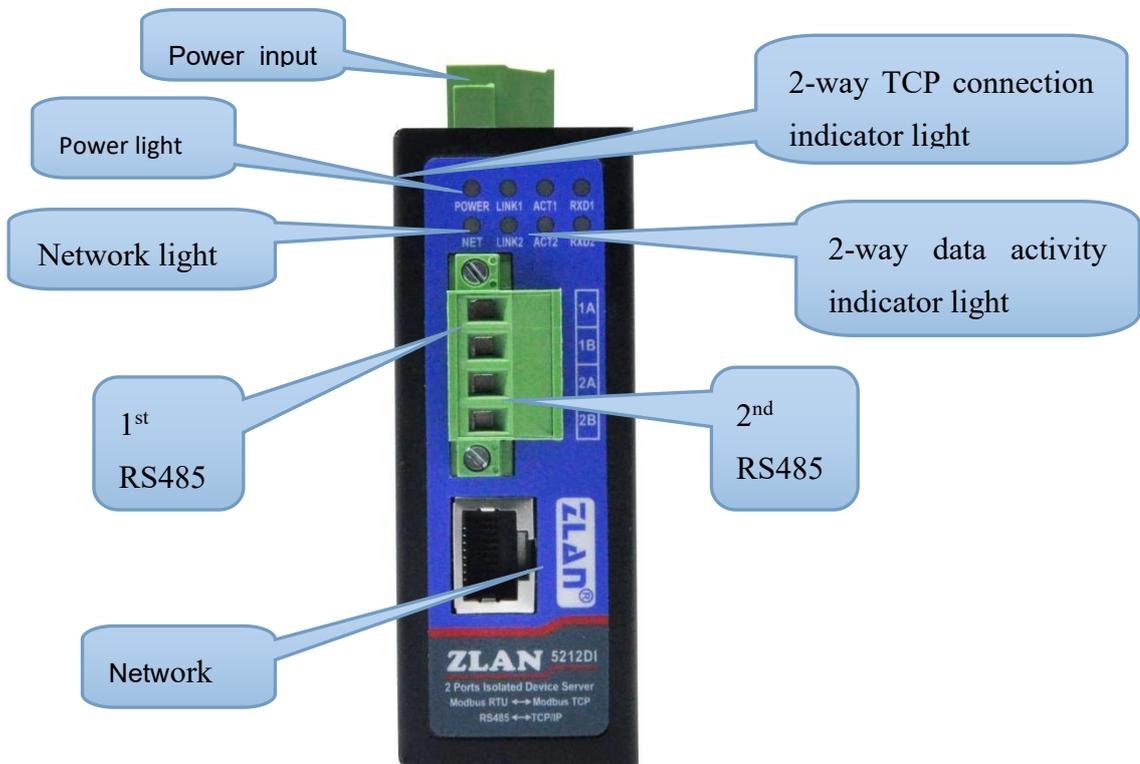| **Software**	
Protocol:	ETHERNET、IP、TCP、UDP、HTTP、ARP、ICMP、DHCP、DNS
Setting method：	ZLVirCOM、WEB（Web pages can be customized）、Device management function library、Serial port AT command configuration、Device cloud management（ZLAN Cloud）
Net communication method:	Modbus TCP、MQTT、JSON、RealCom、HTTP
Modbus gateway：	Supports multi-host mode, storage mode, and pre-configured table（ZLMB）
Communication method：	TCP/IP Direct communication, virtual serial port method
Work mode	
TCP server, TCP client（the TCP server also coexists）, UDP, UDP multicast	
Power	
Power：	9～24V DC，100mA@12V
Environment	
Running temperature:	-40~85℃
Storage temp:	-45~165℃
Humidity：	5~95%RH

4.Instructions

4.1 Hardware

Figure 3

The main view of the ZLAN5212DI serial server is shown in the figure above. The outer shell is made of black radiation-resistant SECC metal shell.

1   **Power input:** The wiring terminal is a 5.08mm terminal, V+ is connected to 9~24V, V- is connected to GND, and there is also the shell ground.

2   **RS485**：5212DI has two 485 serial ports, just use RS485 to connect 1B and 1A or 2B and 2A, where 1B/2B represents the 485 negative line, and 1A or 2A represents the 485 positive line; 485 can carry 32 units. The longest communication distance is 1200 meters. Generally, it is necessary to use a terminal resistor when the RS485

11

line exceeds 300 meters. The 485 terminal resistor is 120 ohms.

3   **Network port:** connect the network cable and support automatic crossover.

4   **Indicator lights:** divided into Power, NET, Link, ACT, and RXD lights, which respectively represent power supply, network cable connection light, connection indication, data indication from the network port to the serial port, and data indication from the serial port to the network port.

Table 2

Power light	Power indicator light
NET light	The network cable is connected and the indicator light shows orange.
Link1/2 light	When the TCP connection is established (or in UDP mode), the Link is green. It can be used to determine whether the serial port server has established a communication link with the host computer software.
ACT1/2 light	The indicator light is green when the network port sends data to the serial port and the serial port sends data to the network port.
RXD1/2 light	When the serial port sends data to the network port, the indicator light is blue.

Use indicator lights to debug communication methods:

1) If the NET light is not orange, the network cable is not connected properly. Please check the network cable.

2) If the Link light is not green (only TCP working mode is considered), the host computer software has not established a connection with the serial port server. Please consider whether the IP address is configured in the same network segment.

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

4) If the RXD light is not blue, the serial port data is not returned.

5. **Installation method:** The equipment shell is equipped with 35mm standard rail buckles and mounting lugs. If there are guide rails, the equipment can be directly installed on the guide rails.

Figure 4 Guide rail buckle

The installation of mounting brackets on the back panel of the device and the fixed installation of desktop mounting brackets are as shown in the following figure:

Figure 5 Mounting ear installation

4.2 Hardware Connection

Generally speaking, the serial port server only needs to connect the power supply, serial port, and network cable.

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

The serial port needs to be connected according to the user's serial port device. If you need to connect the first 485 port, just connect the positive terminal of 485 to 1A and the negative terminal of 485 to 1B.

The network port is connected to an ordinary network cable, which can be directly connected to the 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 you do not need the virtual serial port function, you can download the installation-free version. Download address: http://www.zlmcu.com/download.htm

Table 3 ZLVircom Version

Software	Description
ZLVircom Device Management Tool (free-installation version)	The non-installation version does not include the virtual serial port function.
ZLVircom Device Management Tool (installation version)	The installation version contains ZLVircom_x64.msi and ZLVircom_x86.msi. For 64-bit operating systems, install x64, and for 32-bit operating systems, install the x86 version.

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

4.4 Parameter configuration

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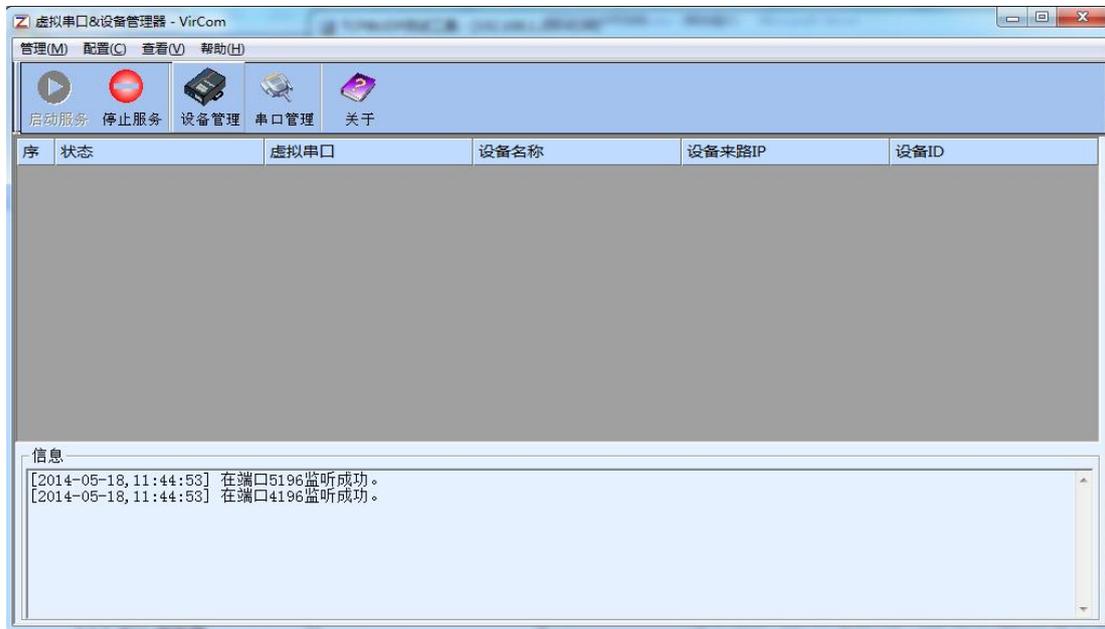
ZLVircom are under the same switch.

Figure 6 ZLVircom Main Interface

You can see the list of devices that have found 2 PORTs. Therefore, it is necessary to sort by name so that different PORTs of the same module are in continuous areas. The way to sort is to click "Device Name" in the title bar. When leaving the factory, the PORT of each device has been named ECF78C-01~ECF78C-02. The ECF78C on the left is the ID of the device, which is the last 6 digits of the ID of the first PORT. -01~-02 indicates which PORT it is. Since the device name can also be modified, if the name has been modified, you can see that the PORT number of the device is 1~2 through the "PORT" list in the list.

Figure 7 Device List

After sorting, the 2 PORTs of the module have been arranged continuously. At this time, start from the first device ECF78C-01 and drag the mouse downward until all 2 PORT devices are selected. Then click "Batch Edit". "Batch editing" will cause 2 PORTs to be modified at one time with the correct IP and port.
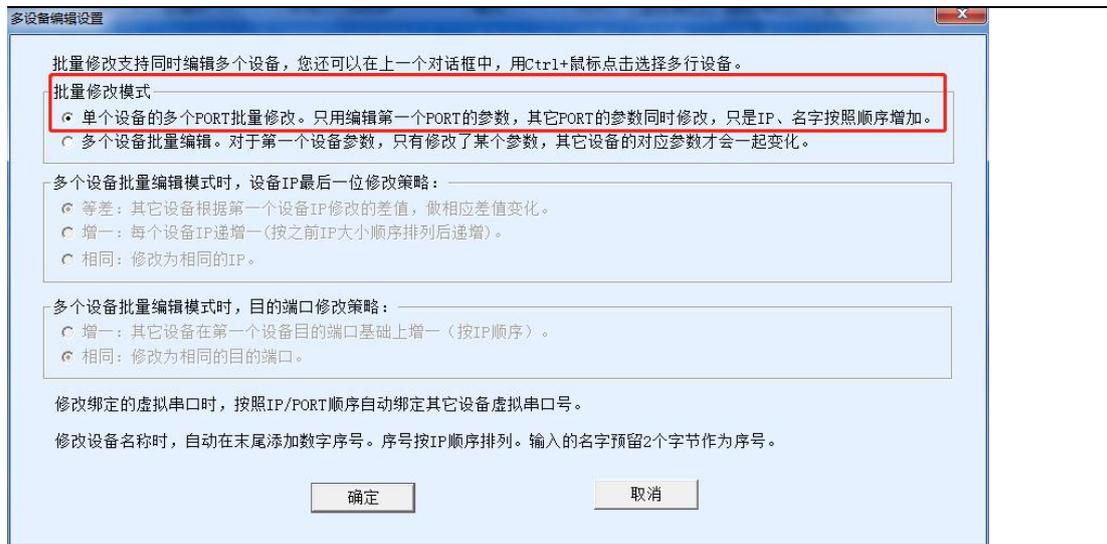
Figure 8 Batch setting options

Since all PORTs 1~2 are selected, the software recognizes that the two PORTs of this module are modified at once, as shown in the figure above. When selecting the batch modification method for 2 PORTs of a single module, you do not need to modify the content of the parameter dialog box. You can modify it directly by clicking "Modify Settings". At this time, although the PORT1 device parameters are not modified, the 2nd PORT will be modified. Modify to the device parameters of PORT1.

Figure 9 Parameter dialog box

In the parameter dialog box, the user can choose to modify parameters such as the baud

rate. If it is a TELNET-to-serial port, please select the TELNET protocol. Among them, ECF78C-01 is automatically filled in by the software based on the PORT ID and does not need to be filled in by the user. IP address 192.168.1.221 is the first IP address, and port 5001 is the port number of the first PORT. Then click the "Edit Settings" button. The software can batch modify 2 PORTs at one time.

Different PORTs of the same module can also be set to different IPs, but if the port is not fixed and the IP must be modified, it is not recommended to use multiple IPs. Here's how to change it to one PORT and multiple IPs.
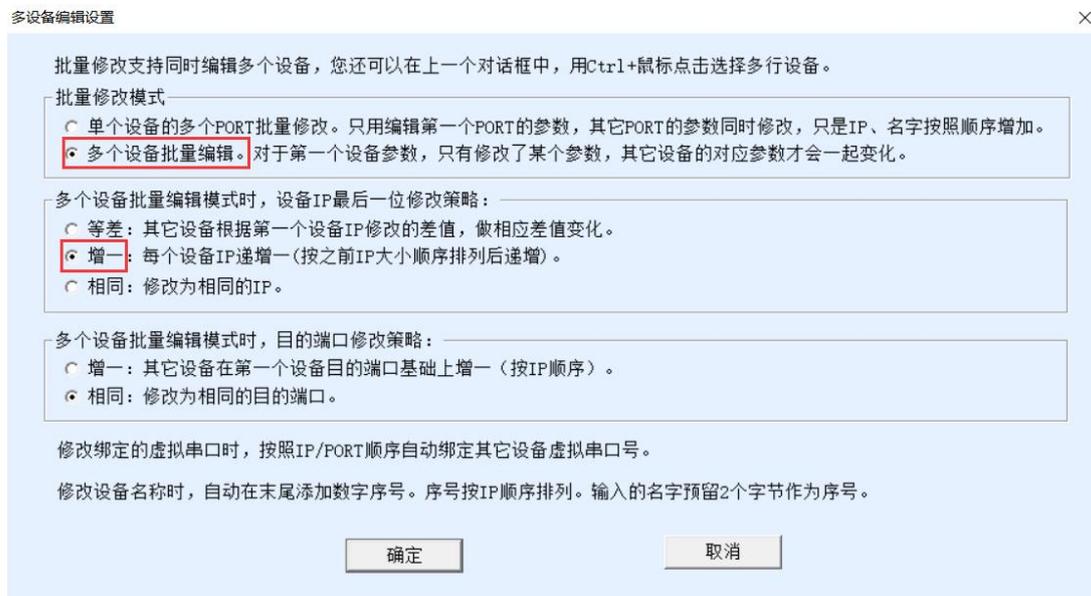
First select the PORT number that needs to be modified, then click "Batch Edit"

Figure 10 Multi-IP modification step 1

In the batch modification configuration, cancel the "single device multiple PORTs" mode and change it to "multiple devices batch editing". This mode will not intelligently identify IPs and ports.

Figure 11 Multi-IP modification step 2

Select "increase one" in the IP mode, that is, according to the first IP set, subsequent IPs will automatically increase by 1.

Figure 12 Multi-IP modification step 3

In the modify parameters dialog box, be sure to modify the IP address. If you do not modify it, no operation will be performed. Because this mode will only modify the changed parameters, and unchanged parameters will not be affected.

Figure 13 Multi-IP modification results

Judging from the modification results, there are now two IPs, from 192.168.1.221 to 192.168.1.222.

4.5 Detailed parameter meanings

Because the functions and parameters of the two serial ports are independent and similar, the subsequent parameters only introduce the parameters of one serial port. The detailed meaning is as follows:

Table 4 Parameter meaning

Parameter	Value Range	Instruction

Name		
Virtual Serial	Non-in use, created virtual serial port	You can bind the current device to a created virtual serial port. Please add a COM port in "Serial Port Management" on the main interface first.
Dev Type		Show only the model of the core module
Dev Name	Any	You can give the device an easy-to-read name, up to 9 bytes
Dev ID		The factory's sole ID, cannot be modified
Firmware Version		The firmware version of core module
Supporting Function		Please refer to the Table 4 "support functions"
IP Mode	Static, DHCP	The user can choose Static or DHCP (Dynamic acquisition of IP)
IP Address		The IP Address of networking products
Port	0~65535	The listening port when the serial port server is in TCP Server or UDP mode. When acting as a client, it is best to specify the port as port 0, which will help improve the connection speed. When port 0 is used, the system will randomly assign a local port. The difference between this time and the non-zero port is: (1) When the local port is 0, a new TCP connection is re-established with the PC when the module restarts. The old TCP connection may not be closed, and there may be multiple false connections with the device. connect. Generally, the host computer hopes to close the old connection when the module is restarted; specifying a non-zero port will close the old connection. (2) When the local port is 0, TCP reestablishes the connection faster. When the serial port server is in TCP client mode, it also acts as a TCP server to monitor incoming

		connections on the port. At this time, the local port number used by the TCP client to connect to the server is "port +1000".
Work Mode	TCP Server(TCP Server Mode),TCP Client(TCP Client Mode),UDP Mode, UDP Multicast	When set as a TCP server, the serial port server waits for computer connection; when set as a TCP client, the serial port server actively initiates a connection to the network server specified by the destination IP.
Net Mask	Eg：255.255.255.0	Must be same as net mask of local LAN.
Gateway	Eg：192.168.1.1	Must be the same as the local LAN gateway.
Dest. IP/Domain		In the TCP Client or UDP mode, the data will be sent to the destination IP or the computer of domain name instruction.
Dest. Port		In the TCP Client or UDP mode, the data is sent to the destination port of the destination IP.
Baud Rate	300、600、1200、2400、4800、7200、9600、14400、19200、28800、38400、57600、76800、115200 、 230400 、460800、921.6K	Serial baud rate
Data Bits	5、6、7、8、9	
Parity	None, Even, Odd, Mark, Space	
Stop Bits	1、2	
Flow Control	None (no flow control), CTS/RTS、DTR/DCR、XON/XOFF	RS232 port valid
DNS Server IP		When the destination IP is described by a domain name, the DNS server IP needs to be

		filled in. When the IP mode is DHCP, there is no need to specify a DNS server, it will be automatically obtained from the DHCP server.
Dest. Mode	Static, Dynamic	In TCP client mode: After using static destination mode, the device will automatically restart after failing to connect to the server for 5 consecutive times.
Transfer Protocol	NONE 、 Modbus TCP<->RTU 、 Real_COM、TELNET	NONE means that the data forwarding from the serial port to the network is transparent; Modbus TCP<->RTU will convert the Modbus TCP protocol directly into the RTU protocol to facilitate cooperation with the Modbus TCP protocol; RealCOM is designed to be compatible with the old version of the REAL_COM protocol. A protocol for virtual serial ports, but when using virtual serial ports, you do not necessarily need to select the RealCom protocol. The TELNET protocol supports the network to log in to our equipment through TELNET to communicate with the serial port.
Keep Active Time	0～255	Heartbeat interval. (1) When 1 to 255 is selected, if the device is in TCP client working mode, it will automatically send TCP heartbeats every "keep-alive timer". This guarantees TCP validity of the link. When set to 0, there will be no TCP heartbeat. (2) When set to 0~254, when the conversion protocol is selected as the REAL_COM protocol, the device will send a data with a length of 1 and a content of 0 every keep-alive timer to implement the heartbeat mechanism in the Realcom protocol. When set to 255, there will be no realcom heartbeat. (3) When set to 0~254, if the device works as a TCP client, the device will send device parameters to the destination computer every keep-alive timer.

		When set to 255, there will be no parameter sending function, allowing remote device management.
Reconnect Time	0～255	When in TCP client mode, when the connection is not successful, the TCP connection is reinitiated to the computer every "disconnection and reconnection time". It can be 0~254 seconds. If set to 255, it means never to reconnect. Note that the first TCP connection (such as powering on the hardware, restarting the device through the zlvircom software, and no data light) will generally be made immediately. Only after the first connection fails will the "disconnection and reconnection time" be waited for and then try again, so "Disconnection and reconnection time" will not affect the normal connection establishment time of the network and server.
Http Port	1～65535	The default is 80
UDP Group IP		UDP multicast
Enable registration package		When the TCP connection is established, the registration packet is sent to the computer. After enabling the registration package, the realcom protocol must be selected. Supports TCP server and TCP client methods.
Range of Frame Length	1~1400	One of the serial port framing rules. After receiving the data of this length, the serial server serial port sends the received data to the network as a frame.
packet interval	0～255	Serial port framing rule two. When the data received by the serial port of the serial port server pauses and the pause time is greater than this time, the received data will be sent to the network

		as a frame.

The functions supported by the device are explained as follows:

Table 5 Device Supported Function

Name	Instruction
Domain name system	The destination IP can be a domain name (such as the www server address at the beginning).
REAL_COM protocol	A non-transparent serial port server protocol, suitable for virtual serial port binding of multiple serial port servers through the Internet. Because the protocol contains the device MAC address, it helps the host computer identify the device. Under normal circumstances, it can not be used.
Modbus TCP to RTU	Modbus TCP to RTU can be realized. It also supports multi-host function
Serial port modification parameters	Supports serial port AT commands to configure and read device parameters.
Obtain IP automatically	Support DHCP client protocol
Multiple TCP connections	When acting as a TCP server, it supports more than 1 TCP connection.
UDP multicast	UDP multicast
Multi-purpose IP	When acting as a TCP client, it supports simultaneous connections to 7 destination IPs.
P2P function	Supports access to devices in any network through P2P traversal technology. Models with an N suffix support this function.
TELNET function	Supports connecting to ZLAN serial port server through Telnet protocol and monitoring device serial port.

4.6 Modify parameters without restarting

Since PORT1 and 2 belong to the same module, generally modifying the parameters of one of the PORT will restart the entire module. However, the following modifications will only restart this PORT, and other PORT in this module will not be affected.

1. Just click the "Restart Device" button without modifying any parameters.

2. The conversion protocol switches between "none" and "TELNET protocol".

2. Modify only one or more of the following parameters:

   a) Local port, destination port

   b) Baud rate, data, check bit, flow control, stop bit

   c) Device name, data packet interval, data packet length

4.7 TCP communication test

Because the functions and parameters of the two serial ports are independent and similar, the subsequent communication introduction will only introduce the configuration of one serial port.

After configuring the device parameters, you can use serial port tools and TCP debugging tools to test TCP connection
communication.

Figure 14 TCP communication diagram

Assume that the COM port of the PC (USB to RS485 cable) is connected to the serial port of the serial server, then open the ZLComDebug (http://www.zlmcu.com/download/Comdebug.rar) serial port debugging assistant and open the corresponding COM port diagram 15; Open the TCP&UDP debugging assistant SocketTest (http://www.zlmcu.com/download/SocketTest.rar) and use it as a TCP client. Fill in the

destination IP as the IP of the serial server (currently 192.168.1.221) and the destination port. to 5001 and click the "Open" button in Figure 16. Enter "socket send" in SocketTest and click Send. The data will be transferred to the RS485 interface through the network port of the serial port server, and then sent to ZLComDebug, and then displayed in ZLComDebug; conversely, enter "Comdebug send" in ZLComDebug and click The send can also be sent to socket test and displayed.

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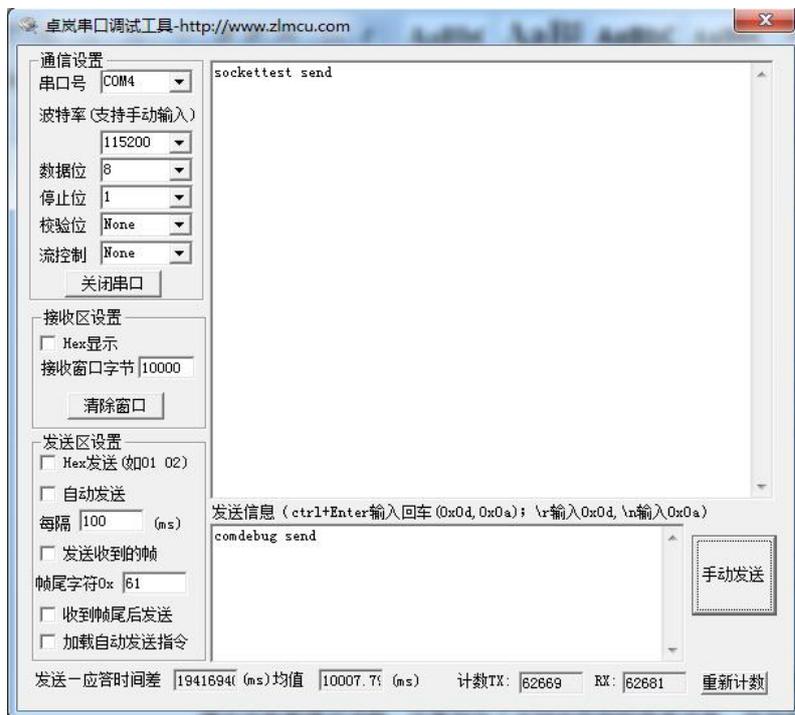

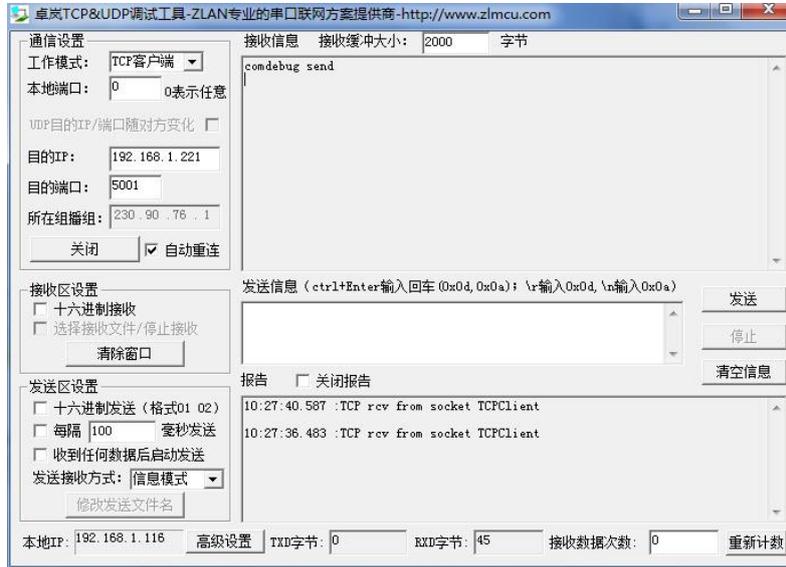Figure 15 comdebug transceiver interface

Figure 16 sockettest transceiver interface

4.8 Virtual serial port test

SocketTest in Figure 14 communicates directly with the serial port server through TCP. In order to allow the user's already developed serial port software to communicate with the serial port server, a virtual serial port needs to be added between the user program and the serial port server. As shown in Figure 17, ZLVircom and the user program run on a computer, and ZLVircom virtualizes a COM port so that the COM port corresponds to the serial port server. When the user program opens COM communication, it can be sent to the user's serial port device through the ZLVircom serial port server. This step is demonstrated below:

Figure 17 The function of virtual serial port

26

Click "Serial Port Management" on the main interface of ZLVircom, then click "Add" and choose to add COM5, where COM5 is a COM port that does not exist on the computer.
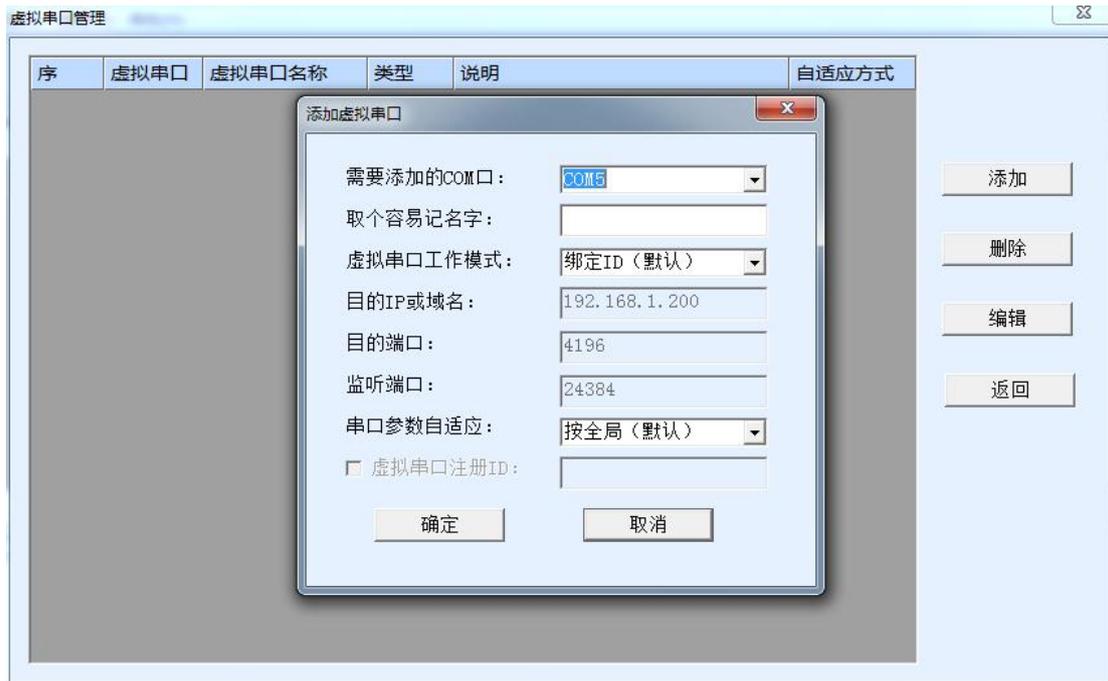

Figure 18 Add virtual serial port

Then enter the device management and double-click the device that needs to be bound to COM5. Select COM5 in the "Virtual Serial Port" list in the upper left corner. Then click "Edit Settings." And return to the main interface of ZLVircom. You can see that COM5 is connected to the device with IP address 192.168.1.211. At this time, COM5 can be used instead of SocketTest for communication.
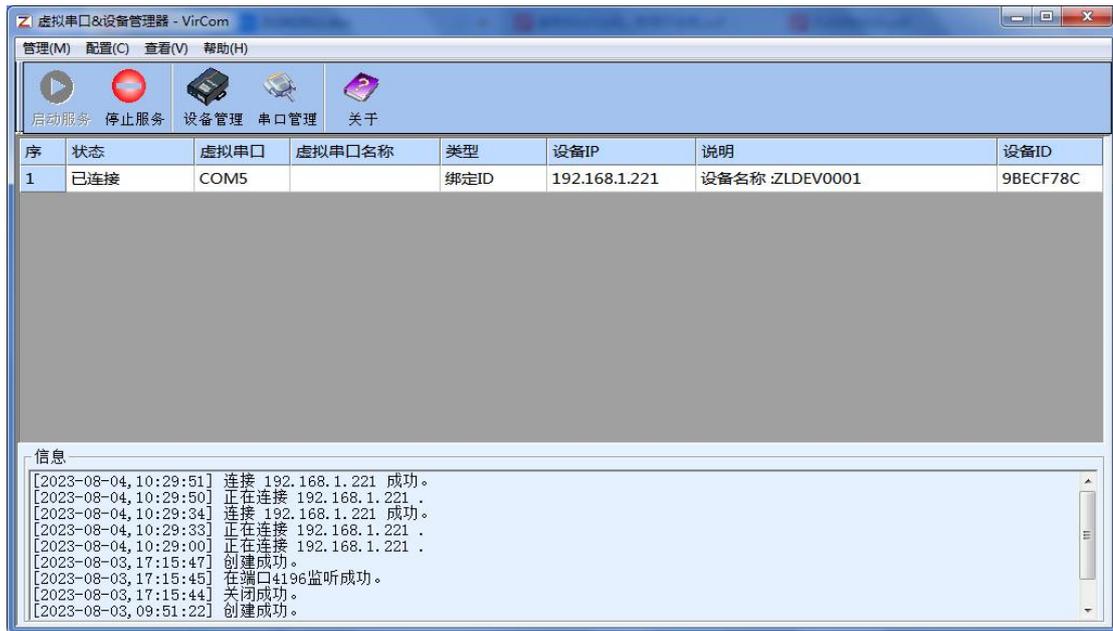

Figure 19 The virtual serial port has been connected

Open ZLComdebug to simulate the user's serial port program, open COM5 (the virtual serial port above), and open another ZLComdebug to simulate a serial port device, open COM4 (hardware serial port). At this time, the COM5 sending data link is as follows: COM5ZLVircom serial server network port serial server serial port COM4. On the contrary, data can also be transmitted from COM4 to COM5: COM4 serial port server serial port serial port server network port ZLVircomCOM5. As shown in Figure 20, both parties send and receive data.

If COM4 is replaced with a user serial port device, COM5 can communicate with the user device.
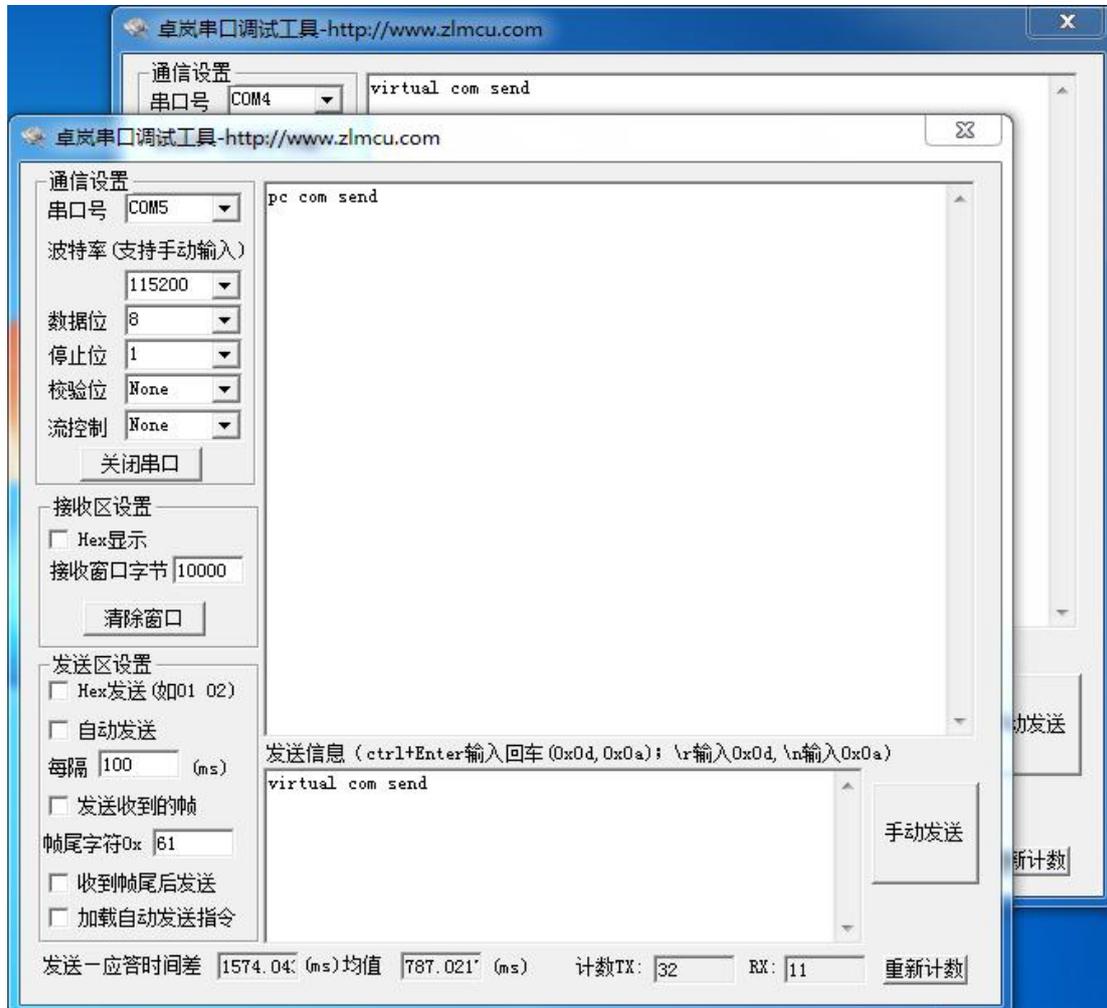
Figure 20 Communication through virtual serial port

4.9 Modbus TCP Test

By default, serial port and network port data are transmitted transparently. If you need to convert Modbus TCP to RTU, you need to select the conversion protocol as "Modbus TCPRTU" in the device settings dialog box, as shown in Figure 21. At this time, the device port automatically changes to 502. At this time, the user's Modbus TCP tool is connected to the 502 port of the serial server's IP, and the Modbus TCP instructions sent will be converted into RTU instructions and output from the serial port. For example, if the serial server network port receives the Modbus TCP command 00 00 00 00 00 06 01 03 00 00 00 0a, the serial port will

output the command 01 03 00 00 00 0a c5 cd. Note: The serial port may send multiple 01 03 00 00 00 0a c5 cd instructions. This is because the default Modbus uses storage mode and will automatically poll and query instructions. I will explain how to switch to non-storage mode later.

Figure 21 Enable Modbus TCP function

If the user's Modbus TCP software is used as a slave, you need to select the conversion protocol, then change the working mode to client, change the destination IP to the IP of the computer where the Modbus TCP software is located, and the destination port is 502, as shown in the figure 22 shown.

Figure 22 Modbus TCP as client.

4.10 Web configuration

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

1. Enter the IP address of the PORT1 port of the serial port server in the browser, such as http://192.168.1.221, and open the following web page.
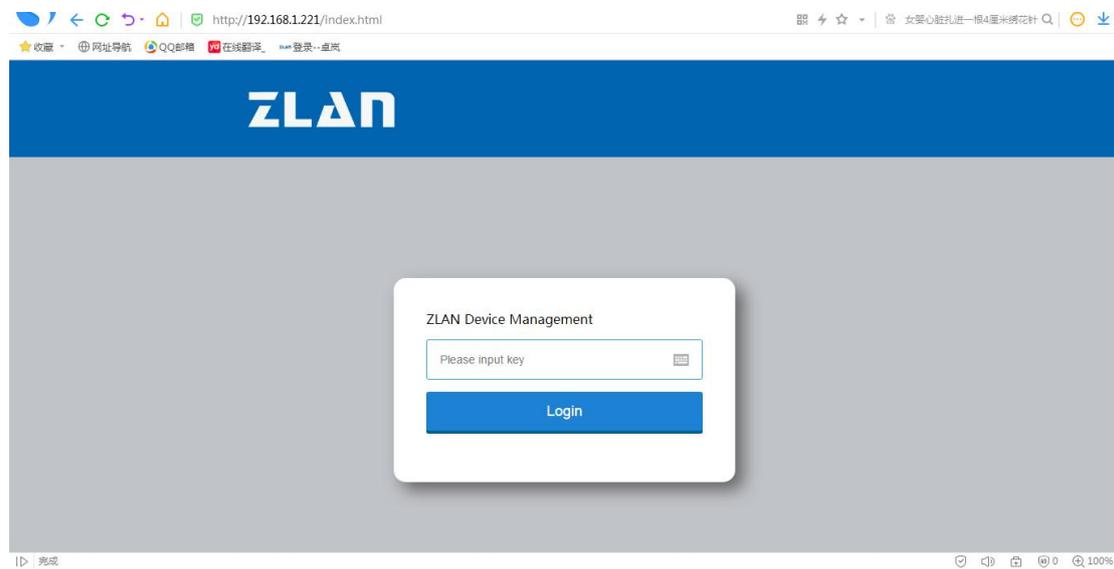

Figure 23 web login page

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

Figure 24 Web configuration interface

3. Select the PORT1-2 you want to modify in the web page that appears. Modify the corresponding PORT port parameters and click Submit. For related parameters, please refer to Table 4 Parameter Meaning.

4. Notes: 1. The IP address of the web page is the IP of the first serial port (PORT1). The IP of the second PORT port cannot enter the web page. 2. Submitting modifications will only submit the parameters of the current PORT port. For example, after modifying the parameters on the PORT2 port page, clicking Submit will only modify the parameters of PORT2.

5. Working model and conversion agreement

Different serial port server working modes and conversion protocols can be selected in different applications, so that they can be used more stably and reliably, as detailed below.

The use of serial port servers is basically divided into two types: with virtual serial port and non-virtual serial port, as shown in Figure 14 TCP communication diagram and Figure 17 the role of virtual serial port respectively. The user software that needs to be connected with

the virtual serial port mode is a serial port interface (COM port), that is, the user software and the user equipment are both serial ports; in the non-virtual serial port mode, the user software communicates directly with TCP/IP, but the user equipment is still a serial port.

In the non-virtual serial port mode, the "conversion protocol part" is divided into 4 modes: transparent transmission, Modbus TCP to RTU, Realcom protocol and TELNET. If the user software is a fixed-protocol Modbus TCP protocol and the host computer is Modbus RTU, you need to select the Modbus TCP to RTU mode; the Realcom protocol is currently only used as a TCP client on a multi-serial server to connect to a server, and the server uses a virtual serial port. The TELNET protocol is suitable for monitoring the serial port of the device when connecting to the ZLAN module through the Telnet protocol.

The usage is summarized as follows:

Table 6 Network configuration modes

No.	Virtual Serial Port Usage	Device Work Mode	Transfer Protocol	Instructions
1	Use	TCP Server	None	Suitable for the user to open the COM port to collect data actively.
2	Use	TCP Server	None	Suitable for the occasion when the device sends data actively. If the TCP server is selected, the problem that the device cannot be reconnected after being disconnected may occur.
3	No Use	TCP Server	Modbus TCP to RTU	Applicable to user software is Modbus TCP and user equipment is Modbus RTU. And Modbus TCP is used as the master station.
4	No Use	TCP Client	Modbus TCP to RTU	Applicable to user software is Modbus TCP and user equipment is Modbus RTU. And Modbus RTU is used as the master station.
5	Use	TCP Client	Realcom Protocol	When a multi-serial server serves as a TCP client and uses a virtual serial port, it is best to use the Realcom protocol.

6	No Use	TCP Server	Telnet Protocol	It is suitable for monitoring the serial port of the device when connecting to the ZLAN serial port server through the Telnet protocol.
7	No Use	TCP Client	None	It is suitable for connecting a large number of devices to a cloud. And generally the cloud is a public IP server on the Internet.
8	No Use	TCP Server	None	It is suitable for devices and computers on the same local network to monitor locally without the need for cross-Internet communication.

5.1 Virtual serial port mode

If the user software uses the COM port for communication, it must use the virtual serial port mode. Including some PLC software, configuration software, instrument software, etc.

Then check whether the monitoring computer and device are both on the local network:

a) If the computer is a public IP server rented on the Internet, then the device must use TCP client mode to allow the device to connect to the server. At this time, you can choose 2 and 5 in Table 6. If it is a multi-serial port server, you must choose 5.

b) If they are all on 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 actively sends data, you must use the device as a TCP client in method 2, otherwise you can choose method 1.

5.2 Direct TCP/IP communication mode

If there is no need for Modbus TCP protocol conversion or virtual serial port, the user software may directly communicate with the network port of the serial port server through TCP/IP, and the serial port server will convert the TCP/IP data into serial port data and send it to the serial port device.

Generally, users with this type of usage develop their own host computer network communication software, which integrates the analysis of the device's serial communication protocol. This method is more flexible and efficient than virtual serial port. Corresponds to 7 and 8 in Table 6.

In the "TCP Communication Test" section, we briefly describe how the serial port server communicates when it serves as a TCP server. Here we will describe how the TCP client, UDP mode, and multiple TCP connections communicate with computer software. Among the computer software, SocketTest (software that imitates user TCP/IP communication) is used as an example.

ZLAN serial port to network port module complies with the standard TCP/IP protocol, so any network terminal that complies with this protocol can communicate with the serial port server. ZLAN Technology provides a network debugging tool (SocketDlgTest program) to simulate the network terminal and Serial server communication.

In order for two network terminals (here, the network debugging tool and the serial port 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 adopted, one party must be the server and the other party must be the client. Then the client can access the server. Both are clients or services. Communication cannot be achieved.

When the serial port server serves as the client, there must be three corresponding relationships, as shown in Figure 25. (1) Working mode correspondence: The working mode of the serial port server is the server mode of the client corresponding to the network tool. (2) IP address correspondence: The destination IP of the serial port server must be the IP address of the computer where the network tool is located. (3) Port correspondence: 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 data can be sent and received after the connection is established.

Figure 25 Serial server as client

5.2.2 Client connects to multiple servers

When the ZLAN serial port server serves as a TCP client, it can connect to 7 destination IP addresses at the same time, and the data sent by the serial port will be sent to 7 destination IP addresses at the same time. If there are not that many servers, the remaining destination IPs will be left vacant. How to use it:

Figure 26 The first destination IP and port

Figure 27 The remaining 2 to 7 IPs and ports

The first IP is set in the device setting interface as shown in Figure 26, where the first IP can be a domain name. For the remaining 2 to 7 destination IPs, click the "More Advanced Options" button in the device settings interface to open more advanced options for settings.

After all 7 destination IPs are set, the connection can be automatically made. If the connection cannot be made, it will wait for the "disconnection and reconnection" time and then reconnect repeatedly.

5.2.3 TCP Server mode

When the serial port server serves as the server, there are also three corresponding relationships, as shown in Figure 28, which will not be explained one by one here. After setting up this way, 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.
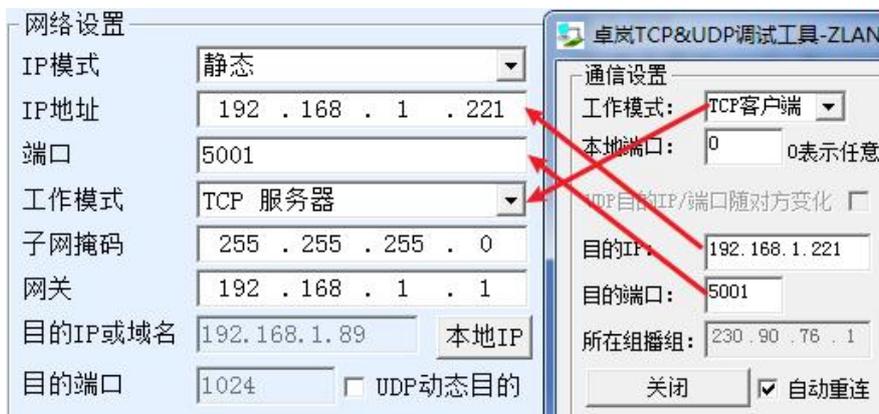

Figure 28 Serial server as server

When the serial server serves as the server, it can accept 30 TCP connections at the same time. Data received by the serial port will be forwarded to all established TCP connections. If you need to send data only to the TCP that has recently received network data packets, you need to enable the multi-host function. Please refer to 7.4. Multi-host function.

5.2.4   Act as both client and server

ZLAN serial port server supports accepting TCP connections when the device is in TCP client mode, that is, it also has TCP server function.

Figure 29   Act as both client and server

By default, when using ZLVircom for configuration, if the working mode is changed to "TCP client" mode, the port (that is, the local port) will automatically become 0 (0 means randomly selecting an idle port). 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 the 5001 port of 192.168.1.221 for communication, and the device will also act as a client. The end is connected to port 1024 of 192.168.1.189. It should be noted that since the local port 5001 is occupied by the server, when acting as a client, the local port uses "port + 1000", that is, the software on 192.168.1.189 will see that the incoming port of the device is 5001 + 1000 =6001.

5.2.5 UDP Model

In UDP mode, the parameter configuration is shown in Figure 30. The left side is the configuration of the serial port server in ZLVircom, and the right side is the setting of the network debugging tool SocketDlgTest. First of all, both must be in UDP working mode. In addition, as indicated by the red arrow, the destination IP and destination port of the network tool must point to the local IP and local port of the serial port server. Indicated by the blue arrow, the destination IP of the serial port server must be the IP address of the computer where the network tool is located, and the destination port of the serial port server must be the local port of the network debugging tool. Only after these network parameters are configured can bidirectional UDP data communication be guaranteed.

Figure 30 UDP mode parameter configuration

5.3 TELNET protocol

When using some Telnet tools to log in to the serial port server, double characters appear. For example, when logging in using Windows Telnet. At this time, you need to select the conversion protocol as the TELNENT protocol. After logging in using Telnet, there are no more double characters.
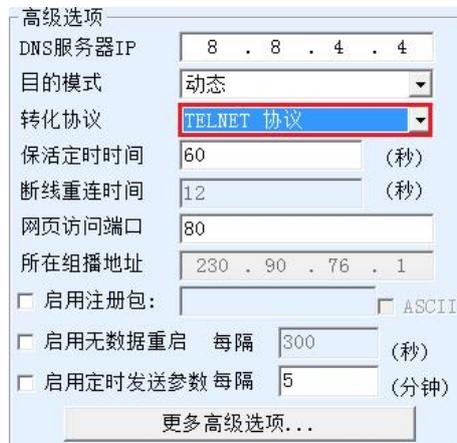

Figure 31 TELNET protocol

When using secureCRT and the TAB key cannot be associated and the problem of double characters occurs, it means that secureCRT is not in "one character at a time" mode. At this time, you only need to change the device port to 23, because the "Send SGA (only port 23)" function in secureCRT is automatically checked. So when our device is on port 23, it will automatically enter "one character at a time" mode

Figure 32 Port is 23

For the method of modifying secureCRT and detailed introduction, please refer to "Telnet Mode Monitoring Serial Port Precautions".

6. Equipment debugging

6.1 Network physical connection

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

After the connection is established, the first step is to check whether the NET light is always on. Otherwise, please check whether the network cable is connected properly.

6.2 Network TCP connection

When the device uses the method of dynamically obtaining IP, it cannot directly connect to the computer network port. Because there is no DHCP server available (usually the DHCP server is the router in the LAN). So please specify the IP when connecting directly. At the same time, the computer also needs to be assigned a fixed IP.
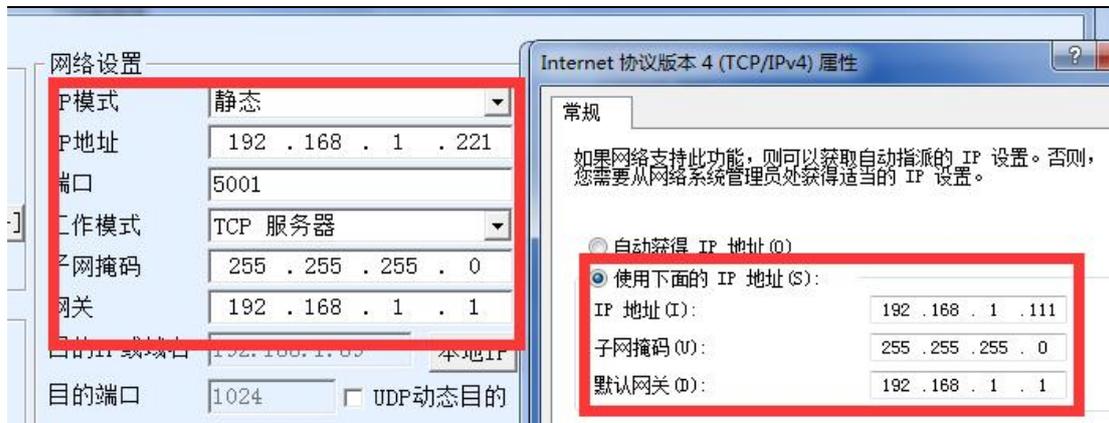
Figure 33 Configured in the same network segment

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

Since ZLVircom supports cross-network segment search and configuration, if the device can be searched but cannot communicate, the IP address is not configured properly. In this case, ZLVircom can be used to configure the device in the same network segment.

After configuration, you can see that the Link light turns green when a TCP connection is established. The green Link light can also be seen through ZLVircom. For example, in the device management list, if the TCP connection is listed as "Established", it means the Link light is green, as shown in Figure 34. This can facilitate remote diagnosis.

Figure 34 Connection status and data sending and receiving status

6.3 Data sending and receiving

When the Link light turns green, data can be sent and received between the software and the serial port server. At this time, if the software sends a data, the Active light will turn green, and the duration will generally be at least 1 second. Data will also be 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 bits, stop bits, check bits) are configured.

The serial port device will generally reply to the correct issued command. Once there is a

reply (the serial port sends data to the network port), the RXD light will turn blue. Otherwise, please check the serial port parameters or whether there is a problem with the serial port cable connection.

In order to facilitate remote debugging, ZLVircom also supports remote viewing of data transmission and reception, as shown in Figure 34. The TXD is the amount of data sent by the serial port of the serial port server. When refreshing the device list, seeing changes in this value indicates that data has been sent. The Active light will also be green; if you see the value of RXD changing, it means that the serial port device has returned data, and the RXD light will be blue.

6.4 ZLVircom Remote monitoring data

When using a virtual serial port, ZLVircom supports real-time capture of data sent and received by the virtual serial port. It is convenient for users to debug the system. The usage method is as follows:

Assume that virtual serial port communication has been established according to the 4.8 virtual serial port test method. Now you need to monitor the data passing through the virtual serial port. Open the ZLVircom menu/Configuration/Software Configuration/Open the vircom configuration dialog box.
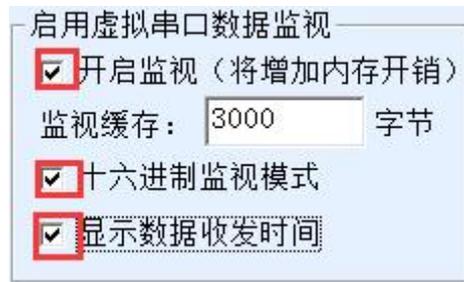

Figure 35 Enable ZLVirocm monitoring

Tick the three options of turning on monitoring, hexadecimal monitoring mode, and displaying 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 that needs 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 opened dialog box, you can see the instructions issued by the host computer and the instructions returned by the device, as shown in Figure

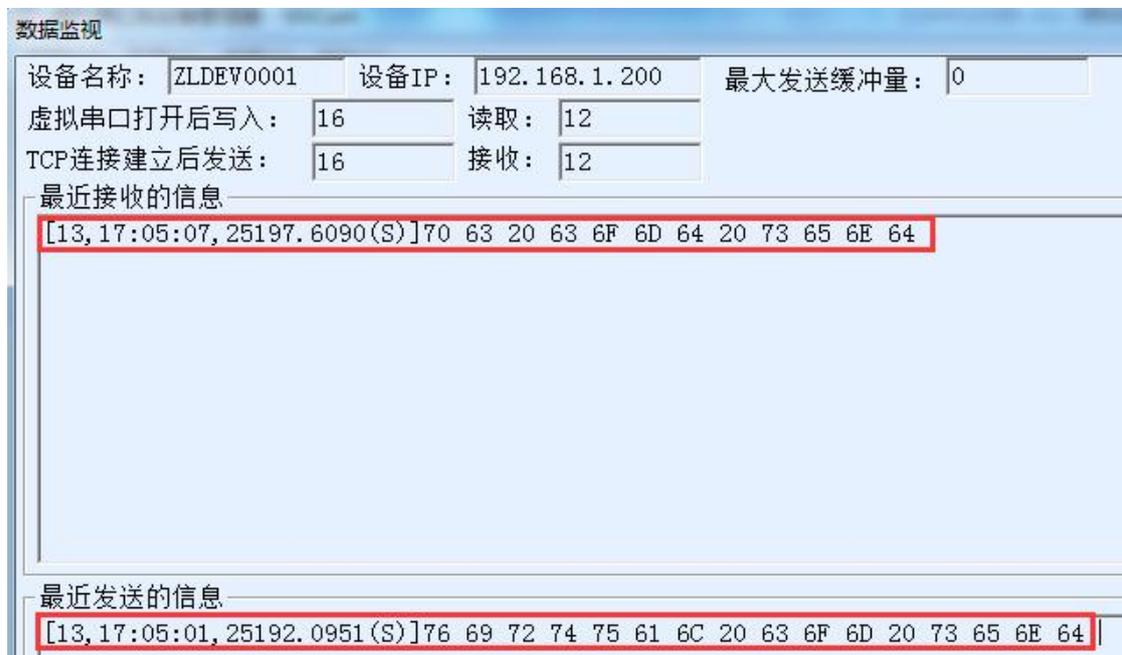This function can facilitate on-site communication debugging.

Figure 37 Monitoring sending and receiving data

7. Modbus Advanced features

The serial port server with Modbus gateway function itself does not have station address and register. It is a communication bridge. It will generate Modbus based on the Salve ID, function code, register number and register quantity in the Modbus TCP command sent by the user software to the Modbus gateway. RTU is specified and output from the serial port. Think of it as a protocol "translator".

7.1 Enable Modbus gateway

First of all, the serial port server should support Modbus gateway, that is, the "Modbus TCP to RTU" function in Table 5 of the functions supported by the device in the device settings dialog box should be checked.

By default, the serial port server is in ordinary transparent transmission mode. If you need to convert to Modbus gateway mode, please select the "Modbus TCPRTU" option in "Conversion Protocol". After that, the device automatically changes the "Port" parameter to 502 (the port of the Modbus server). The Modbus gateway is now enabled.

If the serial port RTU device is used as a slave station, the host computer Modbus TCP software is connected to the 502 port of the Modbus gateway. At this time, the Modbus gateway needs to work in TCP server mode; if the serial port RTU device is used as the master station, the Modbus gateway works on the TCP client, and the destination IP Fill in the IP of the computer where the Modbus TCP software is located. The destination port is generally 502.

7.2 Storage Modbus gateway

ZLAN5212DI can save the contents of the read register inside the gateway, so that the speed of Modbus TCP query can be greatly improved,

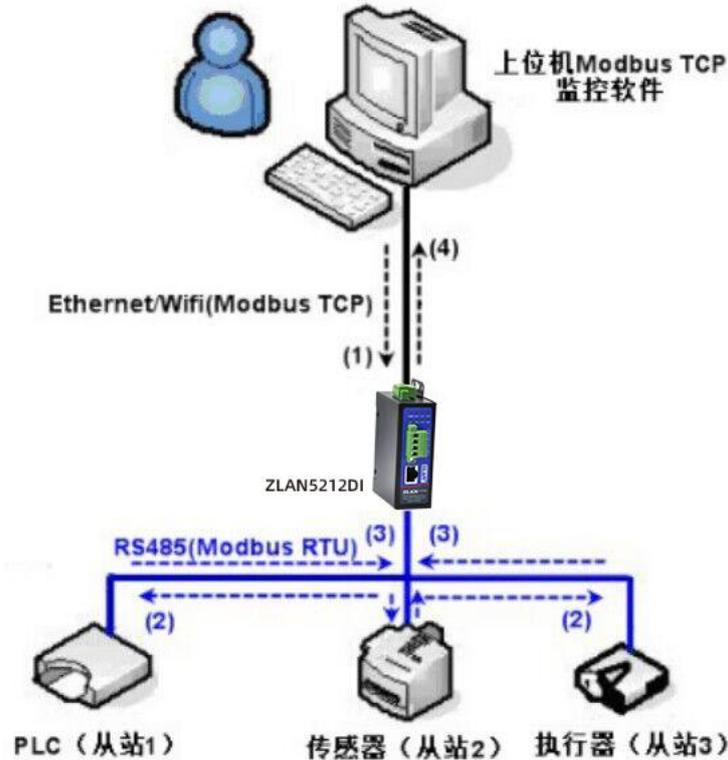and the performance is even better when supporting multi-host access.

Figure 38 Working mode of storage Modbus gateway

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, then the device responds to the Modbus RTU command to the Modbus gateway, and then the Modbus gateway is converted into Modbus TCP again and sent to the monitoring host computer.

We know that Modbus TCP is network communication, the transmission speed is very fast, and it can generally respond within 3ms, while Modbus RTU is RS485, which generally only has a speed of 9600bps. It generally takes at least 30ms to send and return an instruction. The query response time of ordinary non-storage Modbus gateways is relatively long. In addition, if there are many host computers querying data at the same time, the serial port

will be congested. If the network is compared to a highway and the serial port is compared to a single-plank bridge, then the original method is to pass the traffic volume of the highway on the single-plank bridge.

The register-saving Modbus gateway solves the above problems. It can temporarily save the register data obtained by the query inside the Modbus gateway, so that when the Modbus TCP query comes, the Modbus gateway can return the command immediately, truly taking advantage of the fast characteristics of Modbus TCP. On the other hand, ZLAN5212DI can actively send instructions from the serial port to automatically update the content of the currently saved register data and save a copy of the latest register value.

In addition, the module is also a fully automatic configuration-free Modbus gateway. Users do not need to configure the required register addresses, function codes, slave addresses, etc. ZLAN5212DI will automatically identify and dynamically add these registers based on the Modbus TCP instructions sent from the network port.

ZLAN5212DI can show good response speed when monitoring multiple computers. No matter what the baud rate of the serial port is, it can generally respond to the host within 3ms. And it shows good real-time update speed of serial port data.

The register-saving Modbus gateway is a true Modbus TCP to Modbus RTU conversion. It truly takes advantage of Modbus TCP's fast speed and the ability to query multiple hosts at the same time.

Note that when the serial port server serves as a TCP client, it does not have storage type function and will automatically switch to non-storage type. The following lists the characteristics of storage Modbus:

1. The first Modbus TCP query command is non-storage type. Because you must wait for the RTU device to return data slowly before you can reply to the register content of the network port.

2. If a specific command is no longer queried by the network-side host

computer within 5 seconds, the command will be automatically deleted and will no longer be sent from the serial port to the RTU device.

3. Currently, 10K Modbus cache can be stored. For ordinary single-register queries, approximately 500 instructions can be stored at the same time.

4. When multiple instructions are being queried at the same time, they are sent in order. The first instruction is sent, the first instruction response waits for 485 anti-collision time (refer to the multi-host part), the second instruction is sent... After the last command is responded to, it will return to the first command.

7.3 Disable storage features

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

The method to disable the storage type is to 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.

Note that when using the Web method to configure the conversion protocol, the default is a non-storage Modbus gateway.
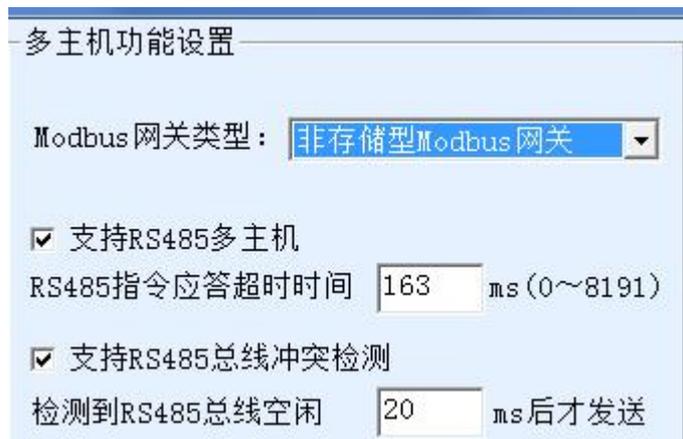

Figure 39 Settings of the new version of zlvircom

7.4 Multi-host function

The "RS458 multi-host support" and "RS485 bus conflict detection function" shown in Figure 39 are ZLAN's multi-host functions. They are generally enabled and disabled at the same time. After enabling it, the device that converts the protocol to Modbus TCP will have the storage Modbus gateway function, otherwise it will be a non-storage Modbus gateway; if the conversion protocol is None, the user-defined RS485 protocol can generally have a serial port device that multiple hosts can access at the same time. function, which cannot be realized in a pure RS485 network, because multiple master stations sending at the same time will cause conflicts on the RS485 bus. The multiple hosts of ZLAN serial port server can "coordinate" the RS485 bus to achieve the purpose of multi-host access.
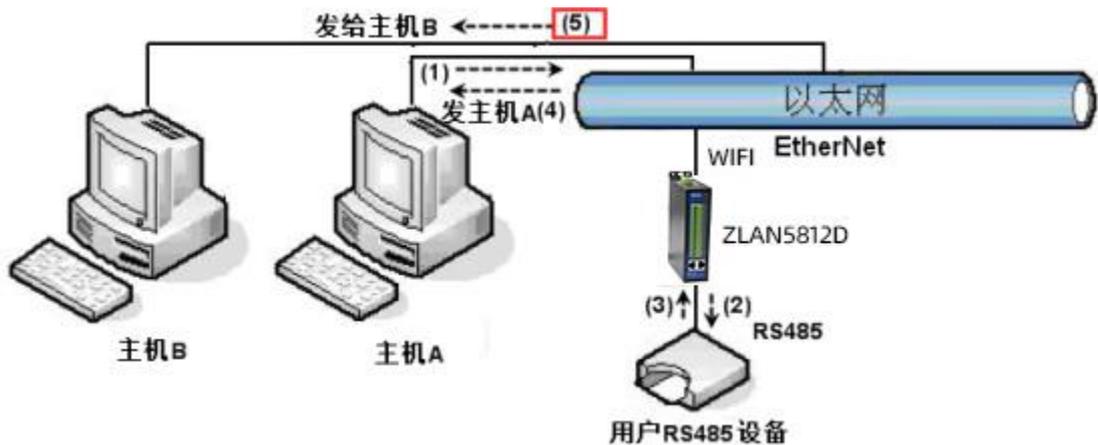

Figure 40 Multi-host function demonstration

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 the (1) command, the RS485 device receives the (2) command, and the RS485 device returns (3) command, but the network port of the serial port server will send (4) to host A and (5) to host B at the same time. Since host B did not send a query, but it also received the response command (5), host B may cause a

communication exception error. In multi-host mode, there will only be command (4) but not command (5), because the serial port server will automatically remember the host that needs to be returned, and will only return the command to the nearest communicating host. Host A's query will only reply to A, Host B queries and replies to Host B.

Another function is that in normal mode, if host A and host B send data at the same time, the instructions will be merged on the RS485 bus, making it impossible to recognize them normally. In multi-host mode, the serial port server can schedule the commands of A and B on the bus. Sequence, thus effectively solving the conflict problem of simultaneous access by multiple machines.

When the conversion protocol is "None", the multi-host function is not enabled by default. When you need to enable multi-host, please click "More Advanced Options" in the device configuration dialog box, and then check "RS485 Multi-host Support".

7.5 Multiple host parameters

The meanings of "RS458 multi-host support" and "RS485 bus conflict detection function" are introduced below.
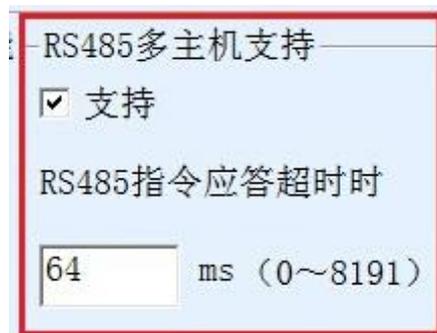

Figure 41 RS485 multi-host support

The RS485 command response timeout time is: the maximum time interval between the serial port server serial port starting to send this command and receiving the response. The time filled in should be greater than the actual maximum time interval. Because if it is determined to have timed out, the next instruction will be sent.

Figure 42 RS485 anti-collision idle time

RS485 bus conflict time: Indicates how many milliseconds the serial port server waits after receiving the reply to the first command before sending the second command. This parameter actually defines the speed of command polling. This value is recommended to be above 20ms. The parameter "Maximum waiting time 3 seconds" generally does not need to be modified.

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 options to remove them), and the above two times will also be automatically configured according to the baud rate . However, if the user's Modus command is relatively long or the conversion protocol is "None", these two parameters need to be configured manually.

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

1. Figure 42 shows the "RS485 bus anti-collision time", which can generally be set to twice the "packet interval" in the lower right corner of the parameter configuration interface, but the minimum cannot be less than 20.

2. Figure 41 shows the "RS485 command response timeout", which is generally determined based on the command length of the incoming and

outgoing response. If the sent command is N bytes and the response is M bytes, then the recommended value is: "Data Packet Interval" ×(N+M+5)+100.

7.6 Non-storage multi-host

Non-storage Modbus must be used in some places. This is because when an event occurs, the PLC reads the register data, but the data read is the previous data collected by the storage type, which is logically incorrect. , so it must also support non-storage Modbus collection. But on the other hand, it also needs to support multiple hosts at the same time. You only need to select the modbus gateway type as a non-storage modbus gateway.
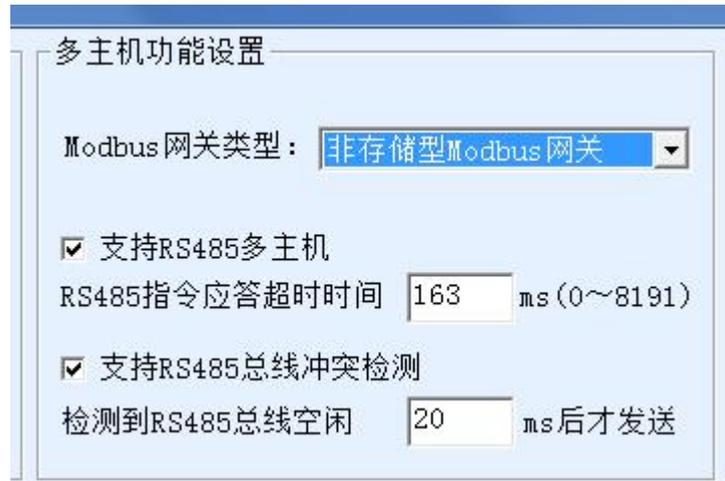

Figure 43 Multi-host non-storage setup

7.7 Modbus under multi-purpose IP

As shown in Figure 40, if the serial port device (RTU device) serves as the master station, and the network port device (Modbus TCP device) serves as the slave station, and there are multiple network port slave devices at the same time. At this time, you can let the serial port server serve as a client to connect to these multiple network port devices at the same time according to the method described in the client's connection to multiple servers.

The function that needs to be implemented at this time is: when the serial

port RTU sends a command, it can be sent to multiple network port devices. The network port device uses the Slave ID field to identify whether to send it to itself. Only the network port device corresponding to the Slave ID responds. The network port response is sent to the serial port server and then converted into RTU commands and output from the serial port to the RTU device.

At this time, it should be noted that the two check boxes shown in Figure 42, "RS485 bus anti-collision time" and "RS485 command response timeout time" shown in Figure 41, need to be removed. Otherwise, the above forwarding function cannot be realized.

Another application method is: although the serial port server serves as a client to connect to multiple network port devices, the RTU device is not the master station. The network port device still sends first, and the RTU device responds (as a slave station). At this time, the "RS485 bus anti-collision time" and "RS485 command response timeout time" still need to be checked, so that the function of multiple hosts accessing an RTU device at the same time can be realized.

For the new version of ZLVircom, you can directly select "Device serves as slave station for the client" in the Modbus gateway type to complete the above settings.
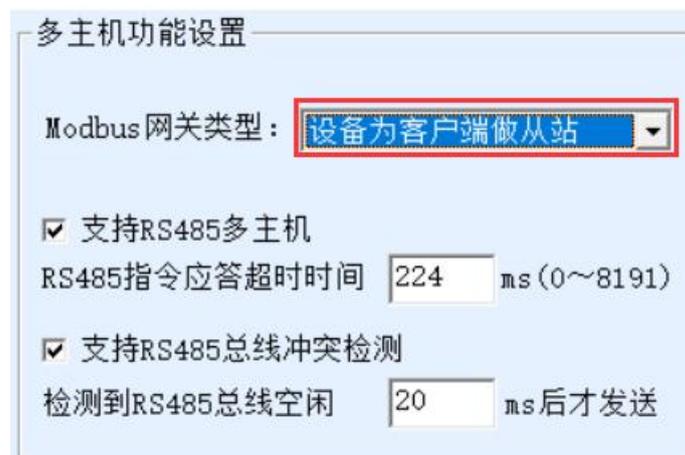

Figure 44 Settings of the new version of zlvircom

8. Registration package and heartbeat package

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

8.1 Registration package

The definition of the registration package is that when the computer software establishes a TCP connection with the serial port server module (hereinafter referred to as the module), 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 monitoring the Internet of Things, because cloud software generally runs on public servers on the Internet, and modules are scattered at various collection and monitoring points. How to make the cloud software recognize the module is very important and is necessary to achieve Internet of Things communication.

Shanghai ZLAN's serial port server provides the following multiple registration methods.

8.1.1 Send MAC address on connection

Send MAC address on connection: When the module connects 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 does not require writing a registration package for each device, so it is simple and effective. The method of use is: click "More Advanced Options" in the device settings dialog box, find "Send MAC address when TCP is established" in the upper middle part, tick the front, then return to the settings interface, click "Modify Settings".

Figure 45 Send MAC address on connection

8.1.2 Realcom protocol

The Realcom protocol is a mature protocol that contains registration packages and heartbeat packages. Users can use this protocol to implement the registration package and heartbeat package functions. The method to enable the Realcom protocol is: in the "Device Settings" dialog box, select "Conversion Protocol" as "REAL_COM Protocol". Note that the Enable Registration Package part needs to be blank and unchecked.

Figure 46 Enable realcom protocol

After enabling the Realcom protocol, communication will not be transmitted transparently. It has the following characteristics:

1. When the device establishes a TCP connection with 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]~MAC[0]

are the MAC addresses of the device.

2. When the device sends data to the network, it will automatically add the 3-byte header prefix of FA 01 01.

3. Every keep-alive timer, the device sends a 00 1-byte heartbeat packet to the software.

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

8.1.3 Custom registration package

The custom registration package method allows users to fill in an arbitrary registration package format. The method is: in the device settings interface, configure as follows:

Figure 47 Setting up the registration package

The difference from the REAL_COM protocol is that the registration package is enabled here, and the registration package information such as 31 32 33 34 is filled in. Note that this is hexadecimal, that is, the actual data sent is the string 1234. If you need string display, click the "ASCII" option next to it.

When the device is connected to the cloud software, it can automatically send a hexadecimal registration package of 31 32 33 34. This registration package method is more flexible and allows the device to adapt to the existing cloud registration package format; however, there is no wildcard such as MAC

in the registration package, and different registration packages need to be configured separately for each device, which is more cumbersome. The above two methods of sending MAC address and REALCOM are configured the same for each device, but the registration packages are naturally different due to different MACs.

The longest registration packet length is 33 bytes. This method supports registration packets and heartbeat packets in UDP mode.

8.1.4 Configuration file

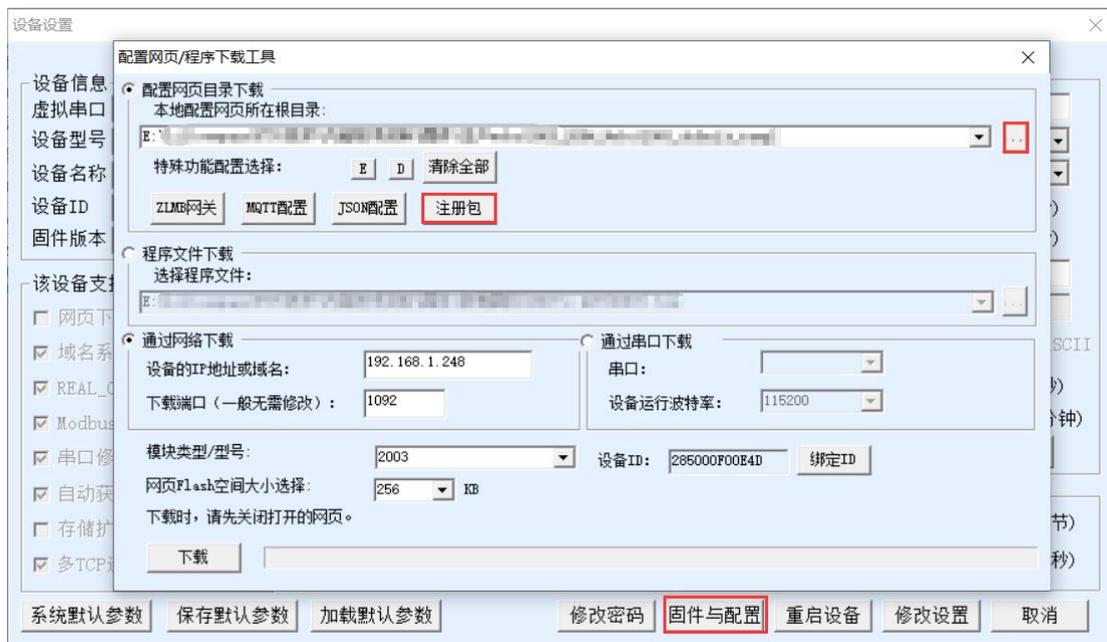For longer registration packages, configuration files can be used.

Figure 48 Download registration package

Click the "Firmware and Configuration" button in the parameter setting dialog box, open the "Configuration Web Page/Program Download Tool", select the "Configuration Web Page Directory Download" method in the

pop-up dialog box, and select an empty directory as the directory where the configuration file is stored. Then click the "Register Package" button.
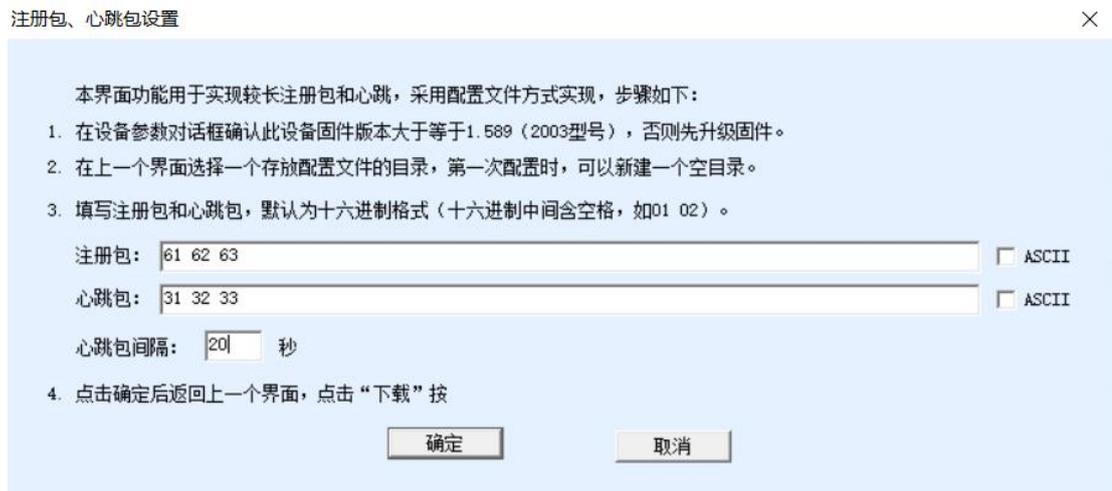

Figure 49 Configuring the registration package

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

8.2 Heartbeat bag

Heartbeat packets are mainly used to detect whether the communication link is disconnected. The method of implementation is that the device sends a heartbeat packet data to the server software every once in a while. This data will be discarded after being received by the server and will not be regarded as valid data for communication.

The heartbeat packet mainly has two functions: first, it can let the host computer software know that the device is active; 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 way to restore network

communication. a means.

Figure 50 Keep-alive timing time

As shown in Figure 51, the sending time of the heartbeat packet is set through the "keep-alive timing time".

8.2.1 Implied heartbeat

Even if no heartbeat packet is set, the implicit heartbeat function will be enabled when the ZLAN device is on the TCP client. Therefore, the implicit heartbeat function means that the device sends data, but the server cannot actually receive the heartbeat data. Therefore, it cannot play the first function of the heartbeat packet, which is the function of the server detecting whether the device is active; but because the device actually sends data, it can play the second function of the heartbeat packet, that is, the device detects the TCP connection. Whether it is a normal function, once a disconnection is detected, the TCP connection can still be automatically re-established.

8.2.2 REALCOM protocol

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

8.2.3 Customized heartbeat package

First fill in the registration package according to the method of customizing the registration package in 8.1.3. Then add a heartbeat packet as follows: Click the "More Advanced Options" button in the device settings, write the hexadecimal heartbeat packet in the second line of the multi-purpose IP and port, and change the option on the right to "Parameter Packet" Purpose".
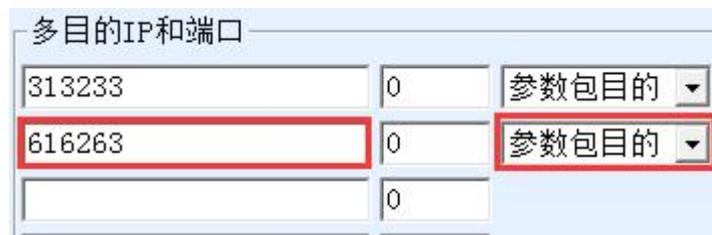
Figure 51 Customized registration package

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

8.2.4 Configuration file

Refer to the configuration file usage of the registration package.

9.Network port modification parameters

Network port parameter modification is to realize the function of searching for devices and modifying device parameters like zlvircom software, that is, managing devices and modifying parameters through the network port of the serial port server. Suitable for users who integrate search and configuration functions into their software.

Network port parameters are modified through the "UDP Management Port Protocol", such as:

1. Computer software sends UDP broadcast packets with destination port 1092 in the network. When the device receives the data packet, it will return its own information to the computer software to achieve the purpose of searching for the device.

2. The computer software sends a UDP parameter modification command to the 1092 port of the device to modify the device parameters.

For a detailed introduction to network port modification parameters, please refer to the "ZLAN Networking Product UDP Management Port Protocol" document. It can also be implemented directly using the device management function library of the 10 device management function library.

10. Device management function library

This function 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, and Delphi.

Detailed API interface introduction documents and VC call Demo cases are provided. It can realize device search, parameter modification, P2P function call, etc.

The development library can be obtained from the ZLAN official website: look for "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

For information on using the MQTT gateway function, please refer to "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 to JSON

For information on Modbus RTU conversion to JSON and related JSON usage, please refer to

"JSON Data Collection Gateway"

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

"Usage of ZLAN Cloud and Collection Equipment"

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

"Method for uploading 645 instruments in JSON format"

http://www.zlmcu.com/document/645_Instrument_JSON.html；

"ZLAN MQTT and JSON to Modbus gateway usage"

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

13. Serial port modification parameters

Users can read and set parameters by sending instructions to the serial port of the serial port server. It is suitable for users who choose chip or module level products to control and configure through the serial port. 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 the serial port command.

ZLAN serial port commands have the following characteristics:

1. The serial port command uses a 10-byte data preamble, so there is no need to pull low and high another configuration pin to distinguish communication data or commands, and there is no need to switch between

command mode and communication mode, making it more flexible and convenient to use.

2. The command set includes various command formats such as saving parameters, not saving parameters, restarting the device, etc.

3. A variety of applications can be implemented, such as reading the MAC address of the serial server, such as modifying the working mode of the serial server. When switching from TCP server to TCP client mode, you can actively connect to the server; when switching from TCP client to TCP server You can disconnect from the server.

4. For detailed operation methods of modifying serial port parameters, please refer to: "Serial Port Modifying Parameters and Hardware TCPIP Protocol Stack"

14. Remote device management

The so-called remote device management refers to the ability to maintain and manage the device through ZLVircom software, including restarting the device, modifying parameters, and upgrading firmware. This function is suitable for users who manage devices through ZLVircom.

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

1. Automatic search: The device and the computer are under the same switch. No matter whether they are in the same network segment or not, the ZLVircom search method on the computer is: ZLVircom sends a broadcast query. After receiving the query, all devices will reply with their own parameters. ZLVircom tools. This method searches all devices at once.
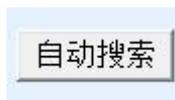

自动搜索

Figure 52 Automatic search

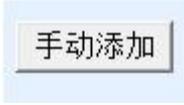2. Manual addition: divided into two situations:

___

手动添加

Figure 53 Add manually

a) Large routers split the network: In some large networks, broadcast packets are split by routers, so the broadcast packets cannot reach the device, but the device IP can be pinged. At this time, it generally needs to be added manually to solve the problem. The method of manual addition is to click "Manual Add" in the "Device Management" dialog box to add the first and last IPs to query the devices one by one.

b) The public network server queries the internal network device: the serial port server is on the internal network and acts as a TCP server mode, and zlvircom is on the server with public network IP. At this time, you need to do a 1092 UDP port mapping on the router of the network where the device is located, map it to the IP of the device, and then zlvircom manually adds the device. The IP is the public IP of the device.

3. TCP client: When the device acts as a TCP client, it will initiate a TCP connection to port 4196 of the destination IP (116.15.2.3). After the connection is established, it will automatically send its own parameter system to the UDP port (note that it is not a TCP port) of the destination port (here is 4196) at each keep-alive timer, allowing zlvircom to be able to operate on this computer (116.15.2.3). Device found. If the destination port is not 4196, you need to modify the default parameter receiving port of zlvircom by modifying the menu/configuration/software configuration/default listening port. Then start zlvircom. If a TCP port conflict pops up, ignore it and continue execution.

Figure 54 Client

4. Send parameters regularly: Even if the serial port server is in TCP server mode, you can check the "send parameters regularly" function to send parameters to the destination port of the destination IP (here 116.15.2.3) every 5 minutes. zlvircom that receives parameters on this port on this server can manage these devices.

Figure 55 Send parameters regularly

In order to easily identify the device, if remote management is required, please give the device an easy-to-remember name.

5. Device operation management: After enabling the cloud management function on the device side, you can see the device on the designated background where ZLAN device cloud management is installed. Device configuration, firmware upgrade, configuration download, etc. can be performed.

15. Firmware upgrade method

Different models can upgrade their own programs, but they cannot upgrade each other's programs. Whether the device is found in the device list through automatic search, manual addition, or P2P search, you can use this method to upgrade the firmware.

1 Obtain the ZLAN5212DI firmware file from ZLAN, such as 1.445(2012).BIN.

2 In the ZLVircom tool, first search for 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 this dialog box again. Click the "Firmware & Configuration" button in the lower right corner of the dialog box.

Figure 57 Firmware and configuration buttons

3. As shown in Figure 58, select the "Program File Download" single option. In Program Files, select the firmware file. The IP address part of the serial port server has been automatically filled in, no need to write it again, and the module type/model has been automatically selected. Then click Download.
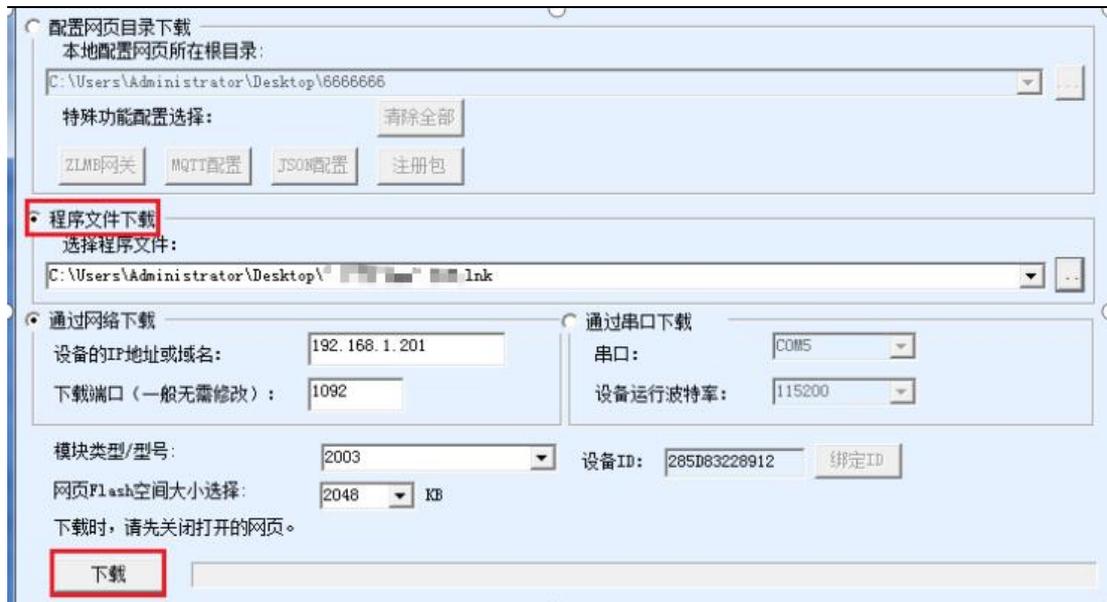
Figure 58 ZLAN5212DI firmware upgrade method

4.At this time, the download progress bar starts to move, and the download time is about 30 seconds. You will see the device's ACT light flashing during the downloading process, and at the end of the download, you will see the LINK light flashing several times. Then the program pops up a prompt box saying "Don't power off the LINK light flashing device after the transfer is completed". Note: This is just the completion of the transfer. The writing process to flash takes about 3 seconds. At this time, the LINK light will flash. Please do not cut off the power during this period

5. After the download is completed, the general program will automatically restart, and there is generally no need to cut off the power. If you see the running indicator light flashing, if it does not restart automatically, please stop the LINK light flashing for more than 30 seconds and power on again.

6. Web configuration interface update: After the firmware upgrade, the configuration web page inside the module also needs to be updated, otherwise it can no longer be configured through the Web, but communication will not be affected. There is no need to download web pages without web configuration. The method to download the Web is: as shown in Figure 59,

change the "Program File" download mode to "Web Page Directory Download".
And select the root directory where the local web page is located as the
directory where the web page files that need to be downloaded are located
(this directory can be obtained from ZLAN), click Download, and download all
the files in the local web page directory to the file system inside the device.

Figure 59ZLAN5212DI web upgrade method

7.Note：

7.1 If it prompts that the download failed, it will not damage the device.
Please restart the download. In addition, when the LINK light flashes at the
end of the download, please do not cut off the power, otherwise the device
will be damaged.

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

Figure 60 Check the firmware version after the upgrade is completed

16. Ordering information

Table 8 Ordering Information

Item	Instruction
ZLAN5212DI	Rail type 485 isolation type two serial port server

17. After-service

Shanghai ZLAN Information Technology Co., Ltd.

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

Phone: 021-64325189

Fax: 021-64325200

Web: http://www.zlmcu.com

Email: support@zlmcu.com