YOXO1007

User manual

Single chip solution for Internet of Things

The following modifications have been made to the document:

ZL DUI 202005281.4.0

The following modifications have been made to the document:

Version information

The following changes have been made to the document:

Modification record

Date	Document number	Modify content.
2020-05-28	ZL DUI 202005281.1.0	Release version
2021-09-30	ZL DUI 202005281.2.0	Modify the reference circuit diagram
2021-10-21	ZL DUI 202005281.3.0	Add power circuit filter capacitance
2022-12-30	ZL DUI 202005281.4.0	Add IO level conversion circuit

Ownership information

CATALOGUE

1. Overview

1.1. Introduction

YOXO1007 is a chip launched by Shanghai ZLAN for Internet of Things applications. YOXO is the chip brand of Shanghai ZLAN. YOXO1007 can realize serial port to Ethernet, UART to TCP/UDP, MQTT gateway, Modbus gateway, JSON to Modbus autonomous acquisition, docking cloud platform and other functions in a single chip, and its network communication mode is Ethernet. Serial port rates range from 110 to 921.6kbps. It can also carry out simple encryption, edge computing and other functions. It has the characteristics of high integration, high cost performance, simple development, strong stability and low power consumption.

YOXO1007 adopts QFN48 package, small size, its Ethernet interface has integrated PHY function, the peripheral circuit only needs power supply, crystal vibration and resistance and capacitance, the hardware design is relatively simple. The working current is about 25mA, 3.3V single power supply. The basic function of YOXO1007 series is to realize the protocol conversion from UART to TCP/IP, that is, the data received by UART is transferred to the cloud/computer, and the data sent by the cloud/computer is forwarded to the UART. To achieve these functions, there is no need for user development programs, and the internal program of YOXO1007 is already solidified and can be used by users. It can be regarded as a single-chip level serial port to network port chip, Internet of Things gateway chip.

Figure 1 YOXO1007 physical picture

It is already integrated with various functions required for the Internet of Things (IOT), including:

1) Search and configure parameters of YOXO1007 through network port;
2) Download the customized Web page through the network port and set parameters such as IP address and baud rate on the Web page.
3) The working mode of YOXO1007 can be configured to automatically connect 7 cloud servers;
4) YOXO1007 can be configured with registration and heartbeat packet functions;
5) Support MQTT gateway function.
6) Support configurable JSON to Modbus/DLT-645 function, support to interconnect with public cloud and ZLAN cloud.
7) Can be configured as http mode, can be directly connected with the cloud asp/php GET/POST command;
8) Can send instructions through the active serial port, collect instrument data and upload it in a customized format;
9) With Modbus gateway function, Modbus TCP to Modbus RTU;
10) Provide device configuration tool ZLVircom, which can support virtual serial port and provide dynamic link library for device management on Windows platform.
11) Support SNMP protocol, can convert various serial port protocols into SNMP protocol (need to select YOXO1007-snmp model).
12) YOXO1007N supports P2P Intranet traversal communication to achieve networking through ID anytime and anywhere (need to select YOXO1007N model).

1.2. Block diagram

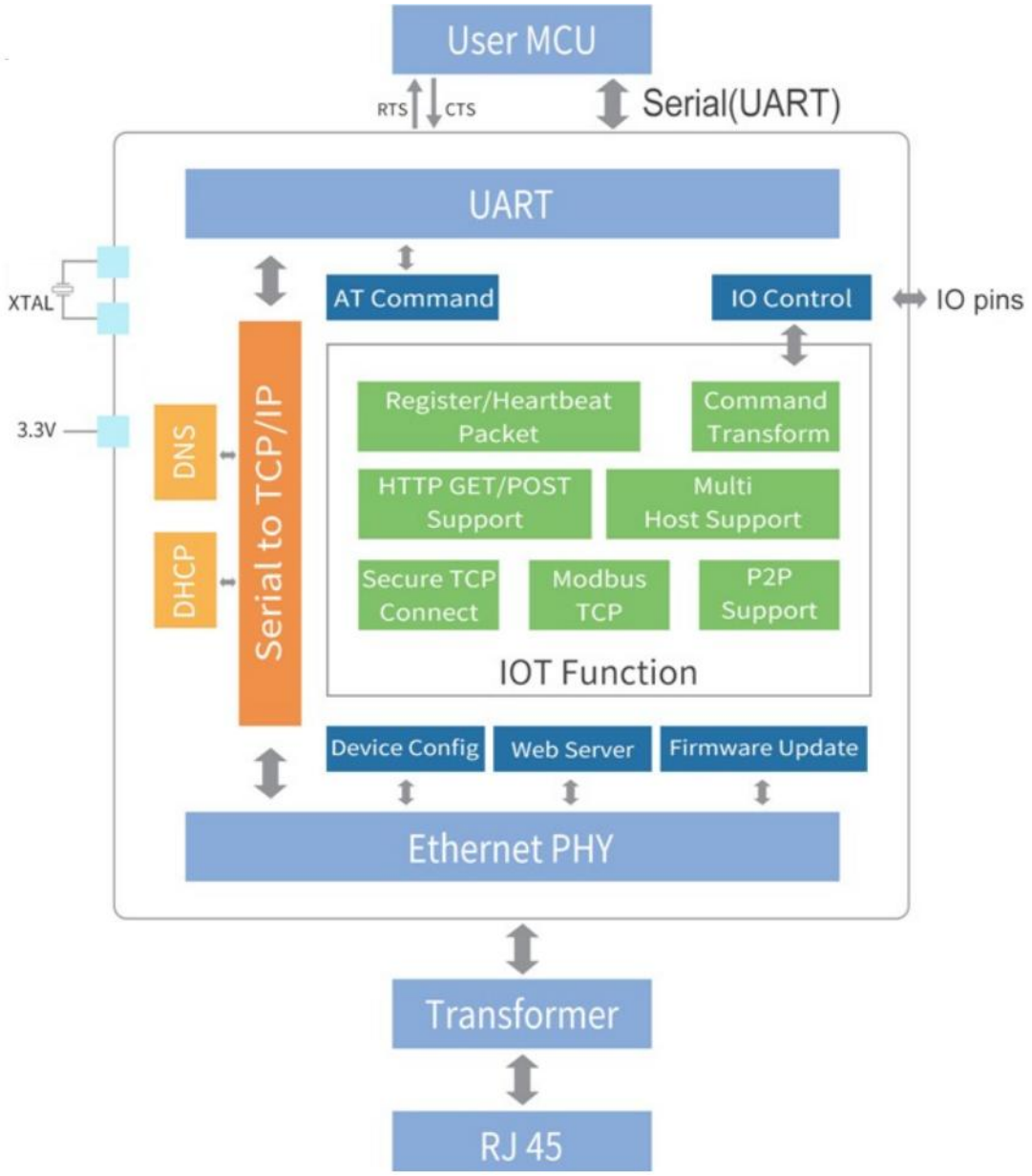


Figure 2 YOXO1007 block diagram

YOXO1007 series internal firmware program is already solidified, no secondary development is required, but the program can be upgraded through the network port. The external interfaces of the chip are mainly UART, Ethernet and IO interfaces. YOXO1007 internal function modules include: network configuration of the chip parameters function, serial port configuration through the similar AT command function, through the embedded Web configuration function, IO control input and output pin function.

1.3. peculiarity

1.1.1. Hardware Features

- High single-chip integration: MAC and PHY interfaces with internal Ethernet integration, no need to expand RAM, FLASH, PHY, and only need external resistance capacitors.
- Supports Auto Negotiation Full-duplex&half duplex.
- 3.3V working voltage, 3.3V signal tolerance.
- The baud rate ranges from 110 to 921.6kbps, and data bits range from 5 to 9. The parity bits can be none, odd, even, mark, or space. The hardware flow control and software flow control are supported.
- QFN48 encapsulation.
- Rich indicator pins: TCP connection establishment indicator, network cable connection indicator, data communication indicator.
- Supports the transmit control pin RS485_EN connected to RS485.
- -40° C~85° C industrial-grade chip.

1.1.2. Basic software features

- Internally solidified full-featured serial port to TCP/IP software without secondary development.
- Supports TCP server, TCP client, UDP mode, and UDP multicast. The TCP client also supports the TCP server function. The TCP server supports 30 TCP connections and the TCP client supports 7 destination IP addresses.
- Support chip connection to send MAC address function, convenient cloud management device.
- Provide computer side search, configuration chip secondary development kit DLL development library.
- Supports Web browser configuration, dynamic IP address acquisition by DHCP, and DNS server addresses.
- Support cloud remote chip search, chip parameter configuration, chip firmware program upgrade.
- Remotely check the TCP connection status of the chip and the status of serial port data sending and receiving through the software. The virtual serial port supports data monitoring.

- Support to send AT-like instructions through serial port to read and write chip parameters and control.

1.1.3. Advanced software features

- 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. Supports configurable Modbus gateway ZLMB.
- Multiple hosts: In the question-and-answer query mode, multiple computers can access the same serial port at the same time.
- Support custom heartbeat packet and registration packet functions: can facilitate communication and device identification with the cloud.
- Supports password authentication to establish TCP connections, ensuring connection security.
- Supports http data submission and delivery. The cloud can directly use http GET commands to interact with serial port data of the device.
- Support to control the level of 8 IO ports through arbitrary network instructions.
- Supports MQTT gateway function.
- Supports the JSON-to-MODBUS RTU/DLT-645 protocol and interconnects with public and Zoran clouds.
- In addition, YOXO1007N supports P2P network NAT traversal. YOXO1007N-snmp supports the SNMP protocol. YOXO1007-W supports sending control commands via a serial port via a built-in web page.

1.4. Advantages

The YOXO1007 series has the following four advantages: zero software development, integrated rich IOT functions, no need for an external MCU for some applications, and seamless connection to the original serial port program.

1.1.4. Zero software development

Figure 3 YOXO1007 Application drawing

The internal program of YOXO1007 series chip does not need to be developed by the user, and the data can be transmitted to the computer program on the network by directly sending and receiving data from the UART of the MCU of the user. The user uses the following steps:

(1) ZLAN provides a complete peripheral circuit diagram of YOXO1007 chip, and users solder YOXO1007 to the circuit board according to this circuit diagram design.

(2) After the circuit board of YOXO1007 is powered on, the RJ45 network port of the circuit board is connected to the network. Any computer in the network searches the device through ZLVircom or the program developed by the user himself, and uses ZLVircom to check the IP and port of YOXO1007. If necessary, the IP and baud rate can be configured. Connect the Socket (TCP/IP) software on your computer to the device IP and port.

(3) The relevant indicator indicates whether YOXO1007 and the computer establish TCP/IP.

(4) After the connection is established, the data sent by the Socket (TCP/IP) software to YOXO1007 will be sent to the user's MCU through UART, and the user's MCU data can also be sent to the computer Socket (TCP/IP) software through UART.

(5) If necessary, the user MCU can also read the MAC of YOXO1007 through AT-like instructions after power-on, modify the IP and other operations.

The above steps do not require complex software development work, hardware development can provide a complete circuit diagram. Only need to use ZLVircom one-time configuration of YOXO1007 IP, baud rate and other parameters can be used after saving, truly achieve zero software development.

It is generally possible to complete the development of an Ethernet-based iot product in 1 week. This zero software development feature provides a quick solution for users to quickly market products, and avoids the problems of long software development time and unstable software development.

1.1.5. Integrate rich IOT capabilities

YOXO1007 has integrated many IOT functional modules, reducing the workload of users to develop related iot functions. Here are some typical functions:

(1) Registration packet and heartbeat packet function: When the Internet of Things device is connected to the cloud, its ID needs to be sent to the cloud software to facilitate the cloud to identify the device number. YOXO1007/1043 has a variety of registration and heartbeat packet modes.

(2) YOXO1007 can be configured as http mode, and can be directly connected with the GET/POST command of asp/php in the cloud;

(3) YOXO1007 can be configured as MQTT function, can be configured with JSON to Modbus RTU function, can independently collect device information, actively send public cloud or Zhuo Lanyun.

(4) With the function of Modbus gateway, Modbus TCP to Modbus RTU;

(5) YOXO1007N supports P2P Intranet traversal communication to realize networking through ID anytime and anywhere: PC can monitor any Intranet device without cloud environment.

1.1.6. Some applications do not require an external MCU

Since YOXO1007 chip does not need MCU to configure through serial port and SPI, the internal curing program can already realize UART to TCP/IP.

Here are a few examples:

(1) Docking wireless serial port module: YOXO1007 can be used as a gateway for some wireless module data acquisition and conversion into TCP/IP. One end of this gateway is a wireless module, such as Bluetooth, Zigbee, Rola and other UART interface modules, and the other end is Ethernet. At this time, only the UART of YOXO1007 and the UART of the wireless module need to be directly connected, and there is no need to add an MCU.

Figure 4 Connecting to a wireless module

(2) Instrument acquisition: YOXO1007 supports automatic acquisition, and the data sent in the customized format is not functional, requiring an MCU to be added on the circuit board for data acquisition and protocol conversion.

Figure 5 Instrument acquisition

(3) Serial port to network port gateway: YOXO1007 itself is a powerful serial port server single chip, if you need to achieve the function of serial port to network port gateway, no additional MCU, no software development. In addition, YOXO1007 has RS485EN pin, which can support various types of 485 chips.

1.1.7. Seamless connection to the original serial port program

Figure 6 No modification of user MCU serial port program

The original internal MCU of many RS232/RS485 devices has a UART interface, and the MCU internal program has been developed, based on the following reasons may not want to re-develop the MCU internal program:

(1) The MCU serial port program has been verified by many years of application and is stable and reliable. If redeveloped into an interface such as SPI, there will be stability considerations.

(2) Due to short R&D progress and insufficient R&D resources, if the original

MCU program can be used directly, it will speed up the progress and reduce R&D investment.

(3) The original MCU program code has been lost or is not being maintained by a technician.

YOXO1007 provides a good solution for the network upgrade of such RS485/232 equipment, as long as the original part of the 232/485 chip is connected to YOXO1007. There is no need for software development, only need to design the circuit to complete the product upgrade.

In addition, ZLVircom equipped with YOXO1007 has the function of virtual serial port, which can also meet the requirements if the user does not want to modify the serial port software of the upper computer.

1.5. Product selection

Model number	name	Feature
YOXO1007	Serial server, Modbus gateway, MQTT gateway single chip	Serial port to Ethernet function, Modbus RTU to Modbus TCP; Multi-host; Registered packet Heartbeat packet; TCP setup requires a password. http GET/POST request; MQTT functions, etc
YOXO1007N	P2P single chip	Add P2P function based on YOXO1007.
YOXO1007-SNMP	SNMP single chip	On the basis of YOXO1007, the SNMP to Modbus RTU function is added.
YOXO1007-W	Web control chip	The function of output control instructions through web pages.

1.6. Target application

YOXO1007 can be used in many networked smart devices:

- Internet of Things gateway: Internet of Things wireless data collection and then converted to TCP/IP upload; Such as MQTT gateway, JSON gateway, interfacing cloud platform and so on.

- Access control: all kinds of RFID and access control equipment are converted into TCP/IP for communication;
- Security: all kinds of perimeter alarm products, such as electric fences, infrared probes, fire and other products of remote alarm;
- Instrument acquisition: collection and uploading of instrument data of various electric meters and energy consumption monitoring meters;
- Modbus Gateway: YOXO1007 itself is a Modus gateway single chip, which can realize the upgrade of Modbus RTU device to Modbus TCP device;
- Industrial automation: All kinds of industrial equipment RS485/232 signals into TCP/IP signals for collection and control.
- Smart Home device: Smart home gateway.
- Network IO controller: YOXO1007 itself has 8 IO input and output pins, and a single chip can realize the design of network 8-way DI/DO products.
- Industrial automation: Industry 4.0 and automated production.
- Building intelligence and energy monitoring: including the collection and transmission of wind and solar power generation.
- Medical monitoring equipment: information of medical equipment.

2. Pin definition

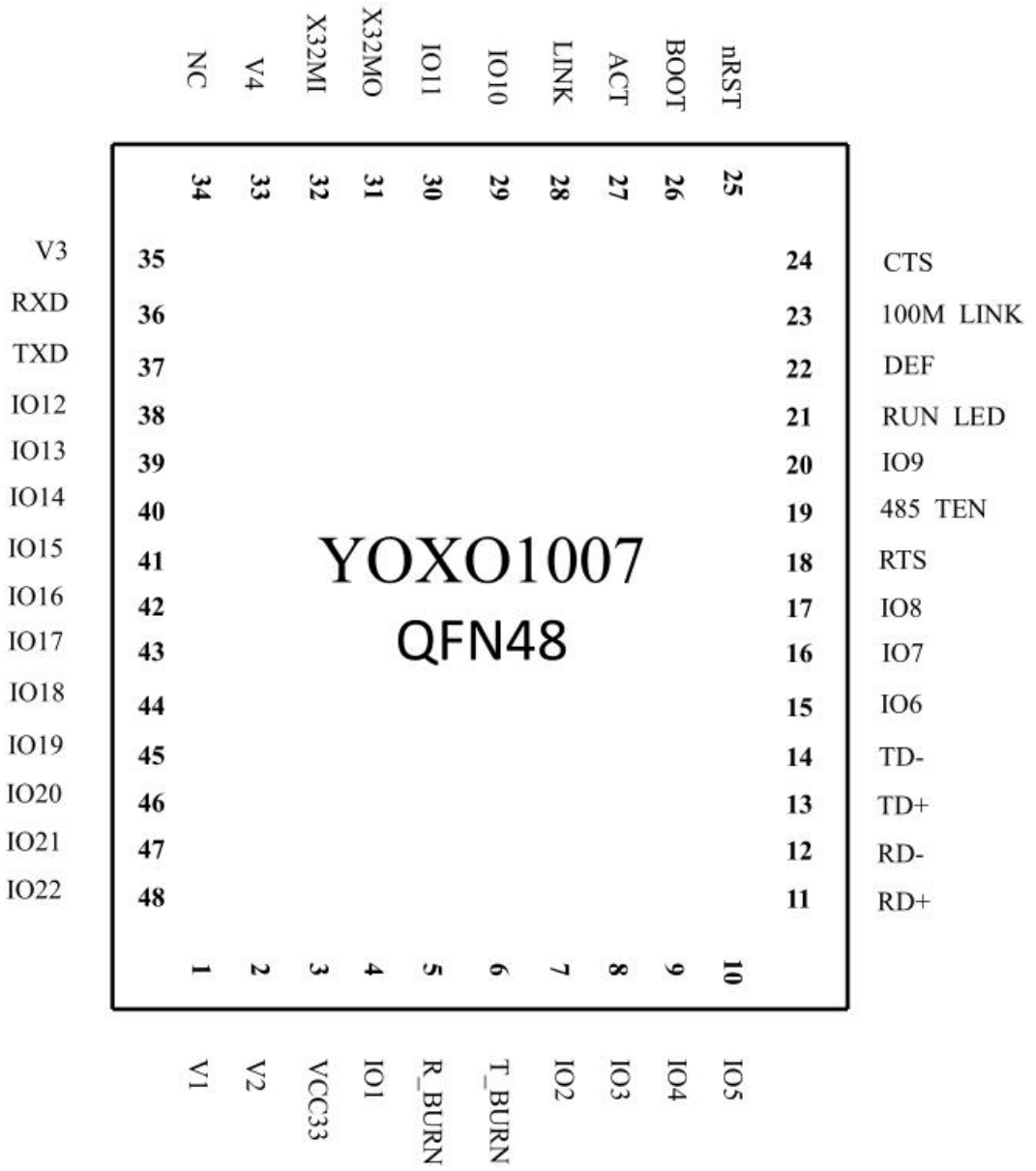


Figure 7 YOXO1007 pin diagram

Table 1 Pin description

Pin name	type	Pin number	Instructions
Power supply, reset, crystal oscillator			
GND	POWER	0	Not marked in the figure, this 0 pin is the bottom of the chip square ground, cooling pad. Must be linked to GND!
VCC33	POWER	3	3.3V power input. Max. 3.9V. The 3.3V Input power supply needs to pass through YOXO8110 and then connect to VCC33 to play the role of stable power supply. In addition, the VCC33 can optionally connect to GND via a 100 ohm resistor. Refer to the following circuit diagram for details.
V1		1	Connect to V3 (pin35) via a magnetic bead (e.g. BLM18AG121SN1D). It is grounded by a 2.2u capacitor.
V2		2	Directly connect to V1 (pin1).
V3		35	Connected to V1 (pin1) via magnetic beads. Ground through a 0.1u capacitor.
V4		33	Ground through the 2.2u capacitor.
nRST	RST	25	Reset pin, active low, recommended max811reus or similar reset chip. Noise can be filtered by 0.1u capacitance. Minimum low time 100ns.
X32MO、X32MI		31,32	32M crystal oscillator.
NC		34	Suspended, no connection required.
Ethernet pin			
RD+,RD-	IN	11,12	Ethernet reception
TD+,TD-	OUT	13,14	Ethernet transmission
Common serial port pins			
RXD,TXD	IN/OUT	36,37	Serial input/output pins, can be directly connected to the MCU serial port, 3.3V TTL level. Notice RXD connects to the TXD of the user MCU, and TXD connects to the RXD of the user MCU.
CTS	IN	24	Hardware flow control input, when the chip is configured as CTS/RTS, DSR/DTR serial port mode, only CTS=0, the chip serial port will output data.

RTS	OUT	18	Hardware flow control output. After the flow control is set to CTS/RTS and DTR/DSR, usually RTS=0 and RTS=1 means that YOXO1007 cannot receive data, and the user MCU should stop sending data to YOXO1007. The reasons why YOXO1007 cannot receive data include: chip initialization, TCP connection is not established, and YOXO1007 serial port receive buffer is full.
Function pin			
RUN_LED	OUT	21	When the chip works normally, it will output a square wave with a period of 2 seconds. You can connect the LED after 4.7K resistance and then ground it, light the LED, indicating that the chip has worked.
100M_LINK	OUT	23	Indicates that the network cable is properly connected. If the value is 0, the RJ45 cable of the module is properly connected.
LINK	OUT	28	TCP connection indicator. If the value is 0, a TCP connection has been established between the module and the network server (or the module is in UDP mode) and the network cable connection is normal. In this case, the module can send and receive data normally. If the network cable is removed, LINK changes to 1. That is, the 100M_LINK indicator must be on when LINK is on.
ACT	OUT	27	Data activity light. If the value is 0, data is being sent or received over the serial port. However, when there is data, the pin will jump between 0 and 1, ACT=1, and can not be used as a sign of no data communication.
DEF	IN	22	Parameter reset. If the value is set to 0 and stays for more than 1 second, the module restarts, and uses the default IP address 192.168.1.254, static IP address mode, gateway address of 255.255.255.0, and gateway address of 192.168.1.1. This facilitates resetting the user to the default IP if they forget the IP. The user's own reset parameters can also be set through the param.txt configuration file.
485_TEN	OUT	19	485 Indicates the sending control end. The value is 0 in normal times and 1 when the module sends data to the serial port. The TXD_EN pin of the MAX485 chip can be connected directly.
IO1~IO22	OUT/IN		Subsequent chip I/O function expansion.
Other pins			
R_BURN,T_BURN	IN/OUT	5,6	The original BOOT firmware is used for upgrading, and generally no connection is required.

BOOT	IN	26	After short-circuiting with the ground, the device is in the original BOOT upgrade mode and can be suspended in the air under normal circumstances.

Look out：All pins are 3.3V. If you need to connect a 5V level, it is recommended to connect the PIN through a 3.3V to 5V level conversion chip (such as TXS0104E).

3. Hardware design guidance

The design example ZLSN3007S is used here to explain the design of the schematic diagram.

3.1.1. ZLSN3007S physical drawing

Figure 8 ZLSN3007S real picture

ZLSN3007S is TTL level serial port to network port products, it adopts RJ45 integrated network transformer scheme.

3.1.2. Detailed explanation of ZLSN3007S schematic diagram

The schematic diagram of ZLSN3007S is shown in Figure 9 and Figure 10.

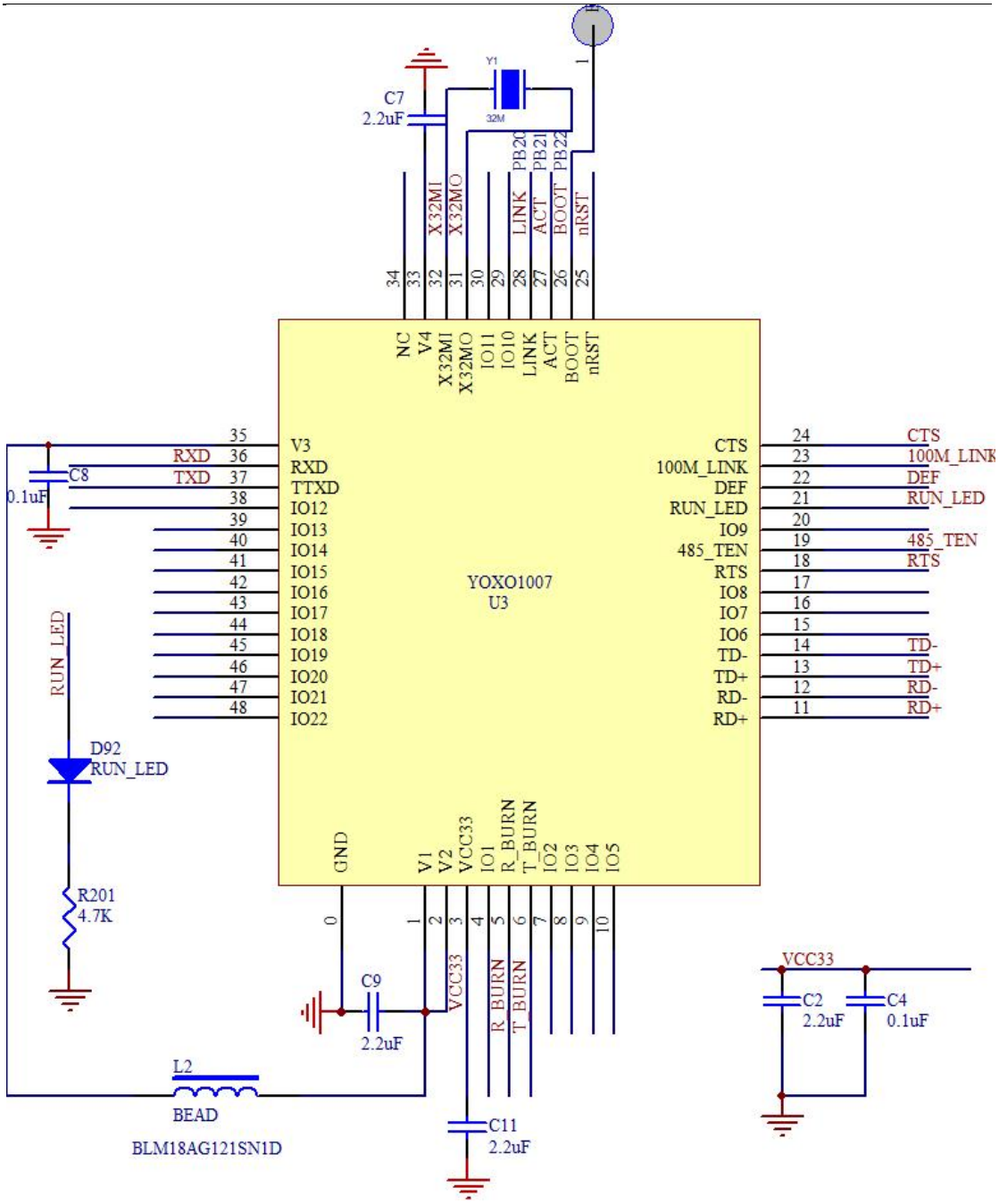


Figure 9 YOXO1007 Core part of the schematic diagram

The main chip section is shown in Figure 9

1 The 3007S leads to the following pins: CTS, LINK, 485_TEN, RTS, nRST, TXD, RXD, DEF, which are directly led from the 1003 chip.

2 The power input is 3.3V. For details, see the power schematic diagram below.

3 RUN_LED Drives an LED to indicate that YOXO1007 is working properly.

4 nRST: can be reset externally, low level, if no reset can be suspended.

5 The connection modes of V1, V2, V3, and V4 are described in the pin definition.

6 BOOT, R_BURN, and T_BURN generally do not require connection and are used for firmware upgrade from the original manufacturer.

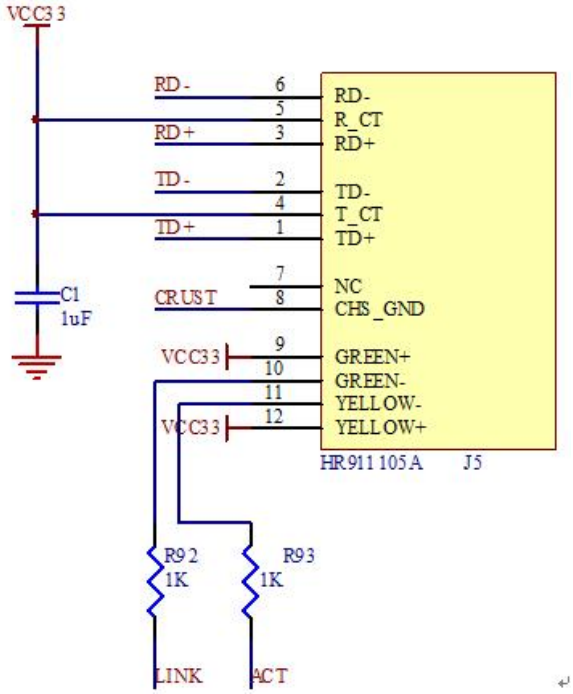


Figure 10 YOXO1007 schematic network interface part (RJ45 integrated network transformer)

Figure 10 shows the design of the network port of YOXO1007

1 Where LINK and ACT are connected to the lights in the RJ45 to indicate TCP connection and data activity.

2 If an independent network transformer is used, the following network port circuit diagram is used

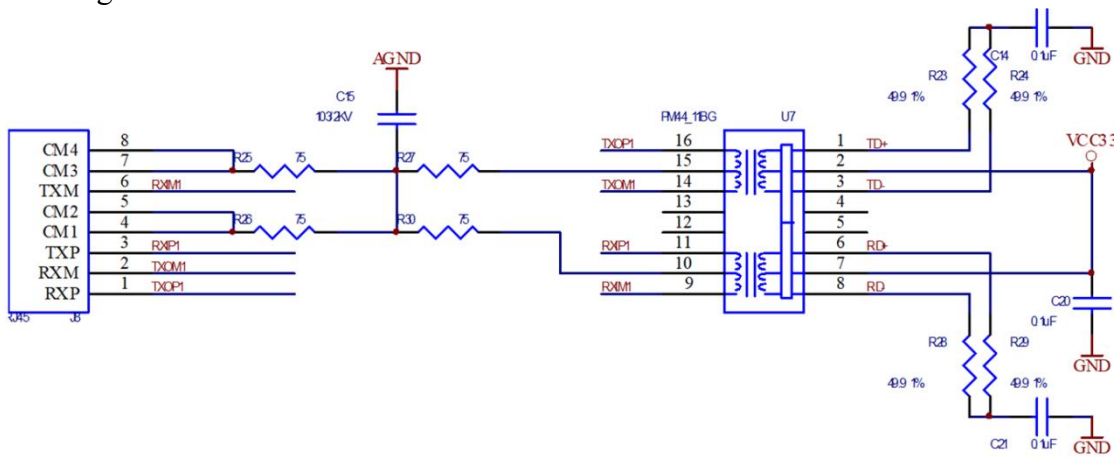


Figure 11 YOXO1007 independent network transformer connection method

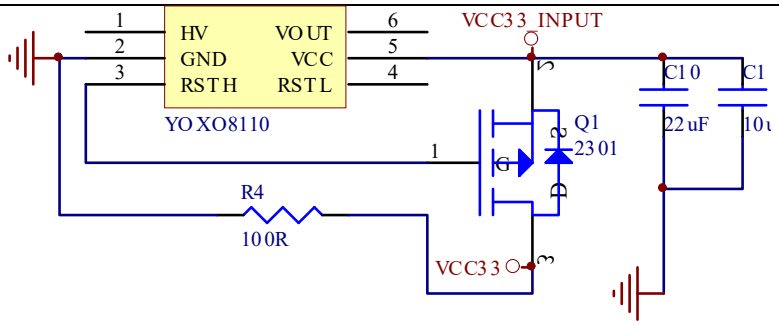


Figure 12 Power supply of YOXO1007 schematic diagram

The power supply part of YOXO1007 is shown in Figure 12:

1. The 3.3V power supply VCC33 INPUT with external input needs to be converted into VCC33 for chip use through the above circuit.
2. The YOXO8110 chip can be provided together with YOXO1007.
3. The recommended model of triode Q1 is: CJ2301 S1
4. R4 is a 100 ohm resistance. When the pin of YOXO1007 is connected with the communication line and control line of the external device, the R4 must be welded to play a stability role.
5. C10 and C11 are ordinary capacitors of 22uf and 10uf respectively. Ensure that VCC_INPUT has a filter capacitor greater than 30uf.

3.1.3. 5V level conversion

If the user's single chip microcomputer RXD, TXD is 5V level (not 3.3V level), or LINK, ACT, CTS, RTS and other pins are connected to 5V level, please use TTL level conversion chip. Otherwise, the chip works abnormally or is damaged.

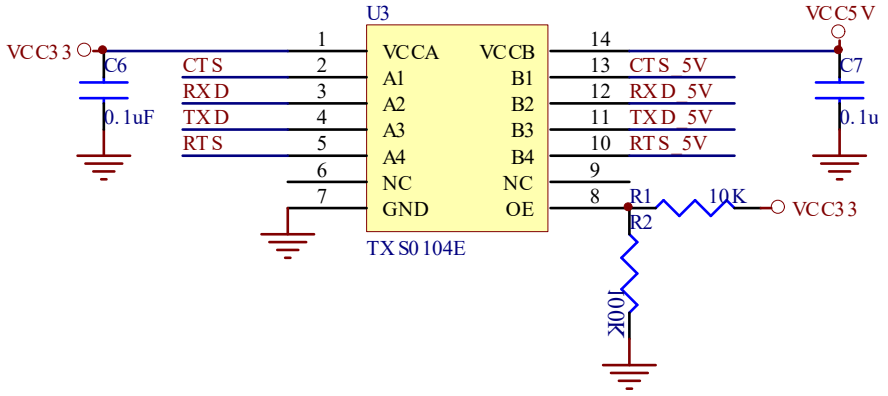


Figure 13 Level conversion of IO interface in YOXO1007 schematic diagram

3.1.4. PCB wiring

PCB wiring is not too high requirements, please note the following points:

1. The crystal oscillator should be as close to the pin as possible. Do not place network lines RX-, RX+, TX-, TX+ near the crystal oscillator.
2. RX- and RX+ are a pair of difference lines, and TX- and TX+ are a pair of difference lines. A pair of difference lines should be routed together to avoid routes that cannot be separated.

4. Function Description

4.1 Software Installation

ZLVircom chips can be used for the IP configuration parameters, such as, and create a virtual serial port, download address http://www.zlmcu.com/download/ZLVirCom.zip. If no virtual serial port function, you can download free installation version of http://www.zlmcu.com/download/ZLVirComs.zip. Download address:

Table 2 ZLVircom version

Software name	Instructions
ZLVircomDevice Management Tool (non-installed version)	The non-installed version does not include the virtual serial port function.
ZLVircom-Device Management Tool (Installation version)	Installation version with virtual serial port driver.

4.2 Parameter Settings

After ZLVircom is installed, the circuit board of the chip is powered on, and the circuit board RJ45 is connected to the network where the computer is located. The ZLvircom software is run as shown in Figure 14, and then click "Equipment management" as shown in Figure 15. 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.

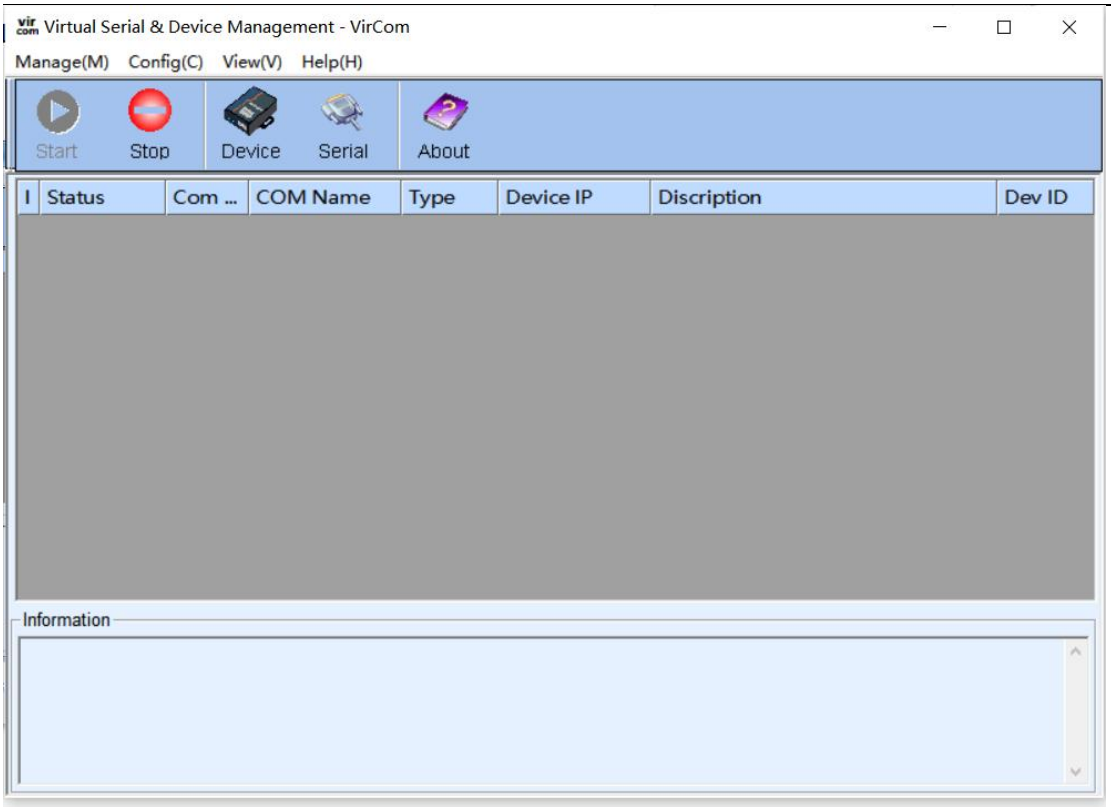


Figure 14 ZLVircom main interface

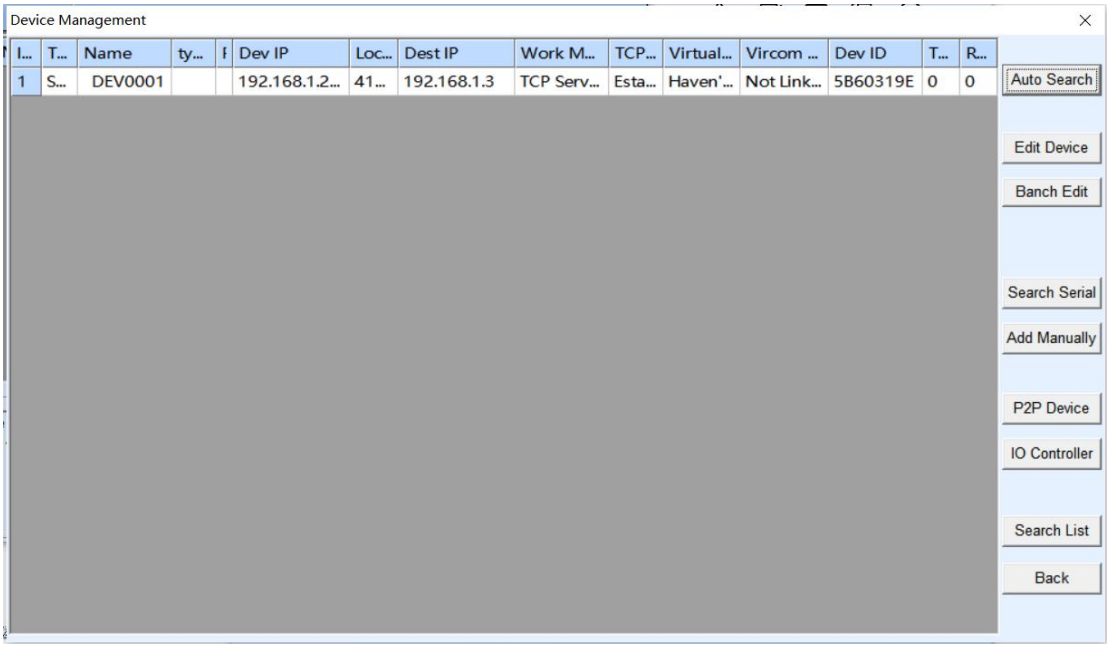


Figure 15 Device list

View all online devices in the device list. Click "Edit Device" to configure parameters.

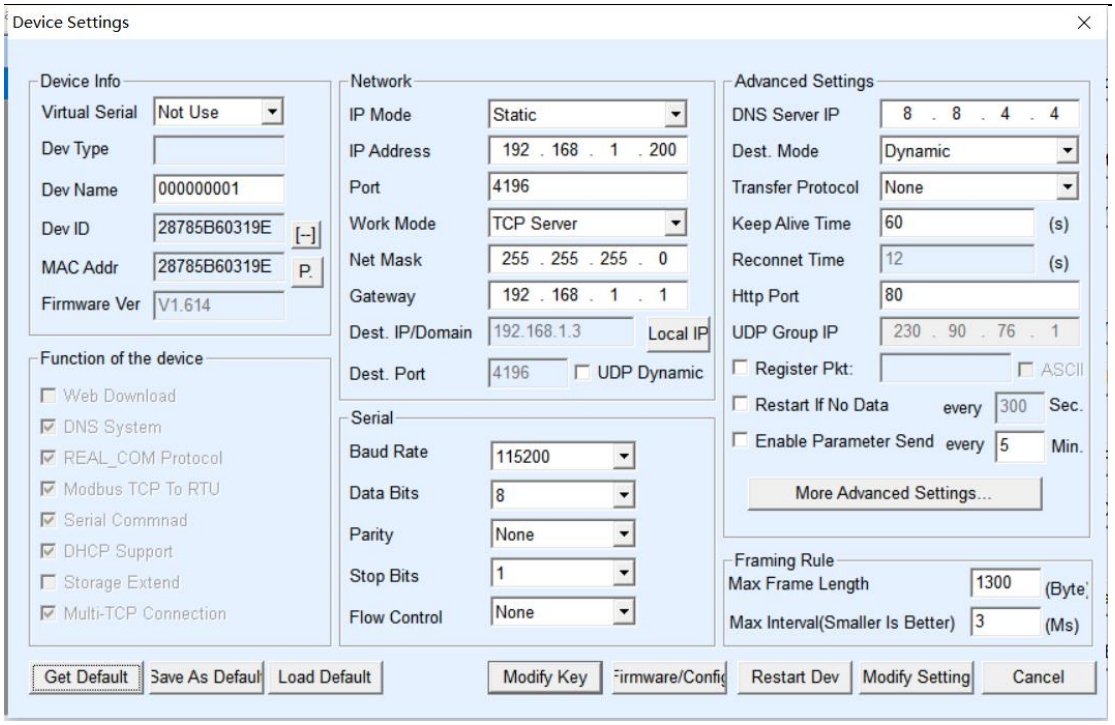


Figure 16 Device parameters

In this interface, the user can set the parameters of the device, and then click "Modify Settings", then the parameters are set to the flash of the device, power failure is not lost. At the same time, the device automatically restarts.

The main parameters are: baud rate, data bit, check bit in serial port Settings; IP address, subnet mask, gateway in network Settings; Sometimes according to the computer software, you also need to configure the working mode of the serial port server.

The meanings of other parameters are as follows:

Table 3 Parameter meanings

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	Static , dynamic	TCP client mode: In static destination mode, the

mode		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	0～255	In TCP client mode, when the connection fails, the

reconnection time		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 5 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.
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.

4.3 TCP Communication Test

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

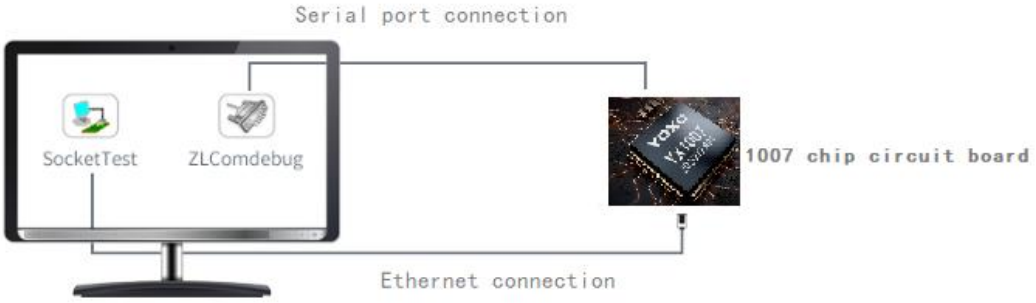


Figure 17 Schematic diagram of TCP communication

Assuming the PC COM port (USB turn RS232/485/ttl line) 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 18; 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 192.168.1.200), The destination port is 4196, then click the "Open" button as shown in Figure 19. On the Ttest of SocketTest, input "socket send" and click send, the data is transferred to RS232 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 forwarding function of serial port to network port and network port to serial port data of serial port server.

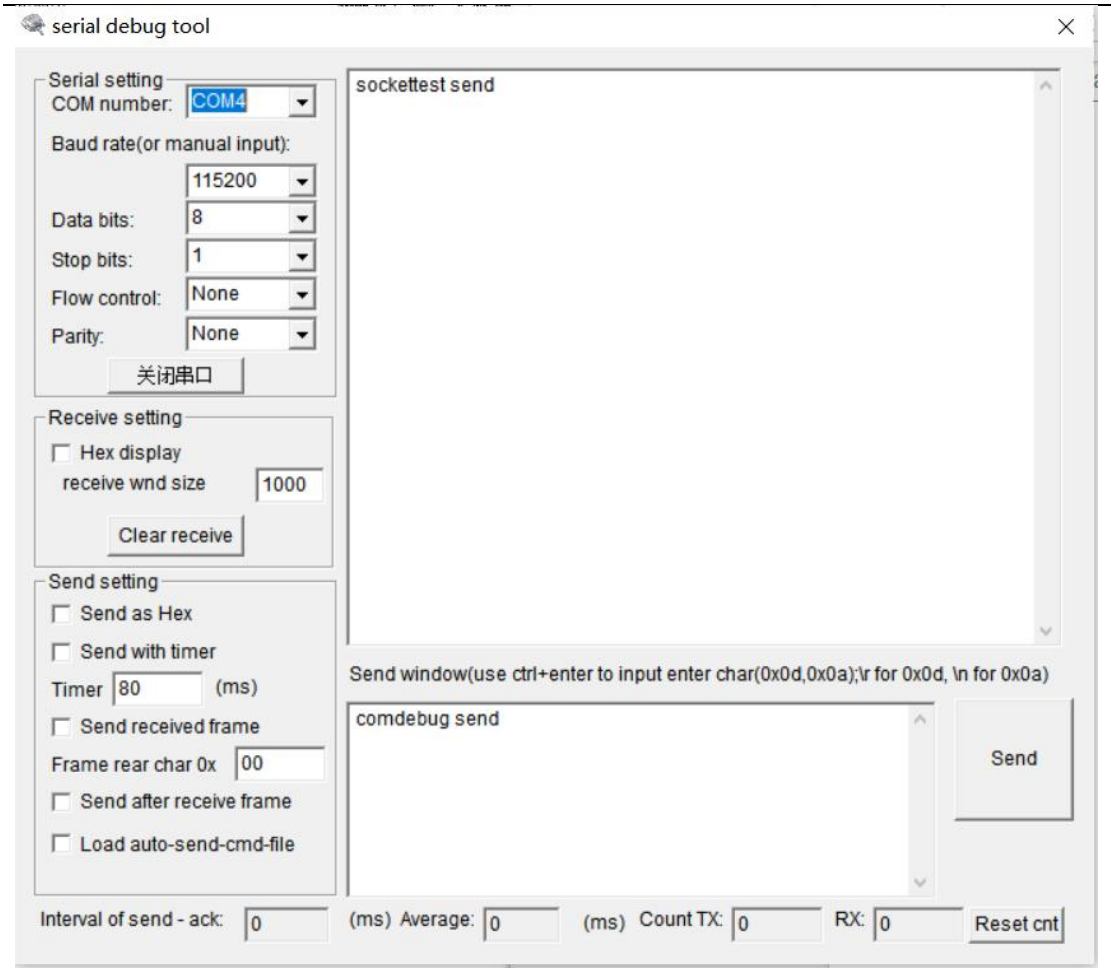


Figure 18 comdebug sending and receiving interface

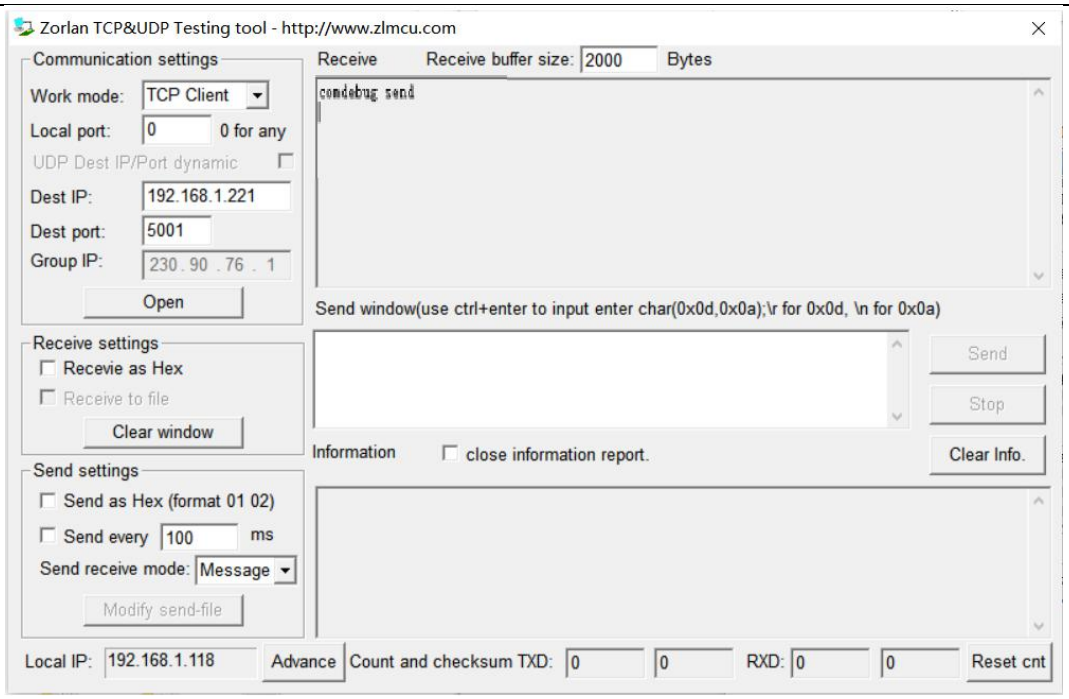


Figure 19 sockettest receiving and sending interface

4.4 Virtual Serial Port Testing

The SocketTest shown in Figure 17 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 20, ZLVircom and the user program run on a computer. ZLVircom virtualizes a COM port and makes this COM port correspond to this 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:

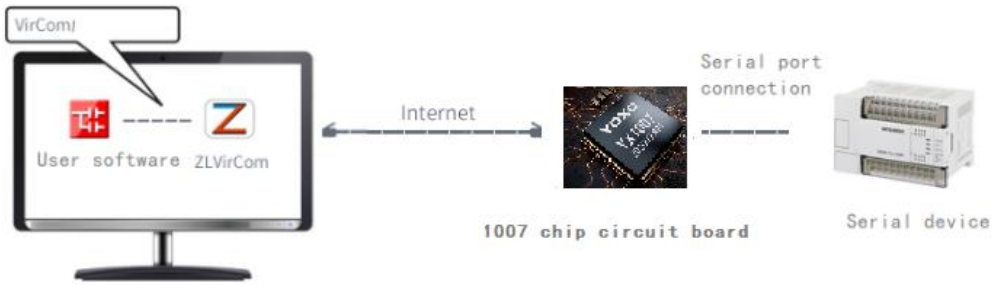


Figure 20 Functions of the virtual serial port

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.

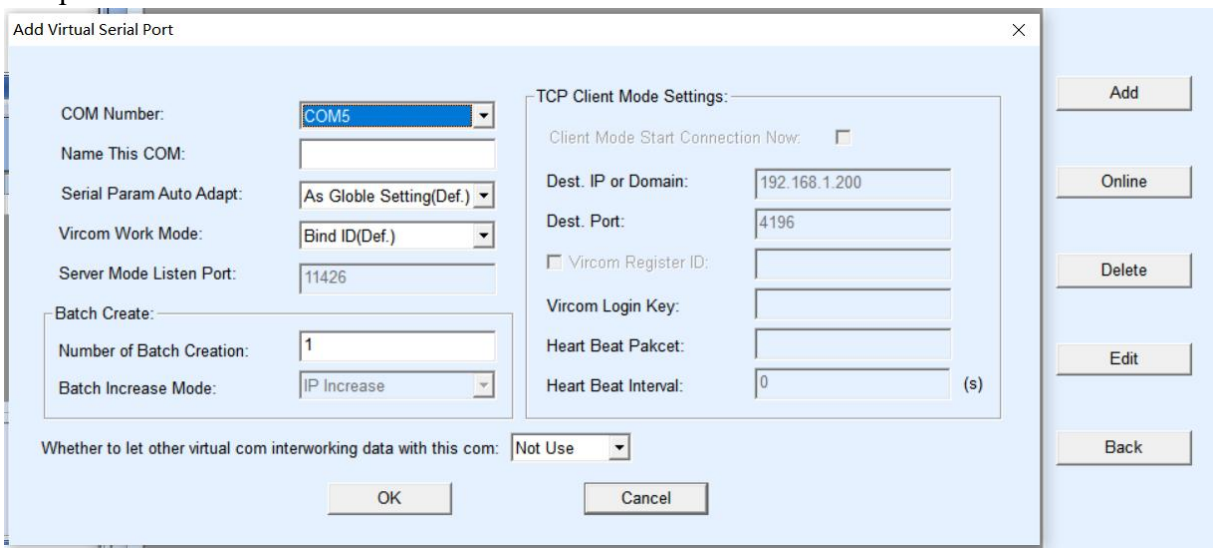


Figure 20 Add virtual serial port

Then go to Device Management and double-click the device that you want to bind to COM5. As shown in Figure 16, 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 192.168.1.200. COM5 can be used instead of SocketTest to communicate at this time.

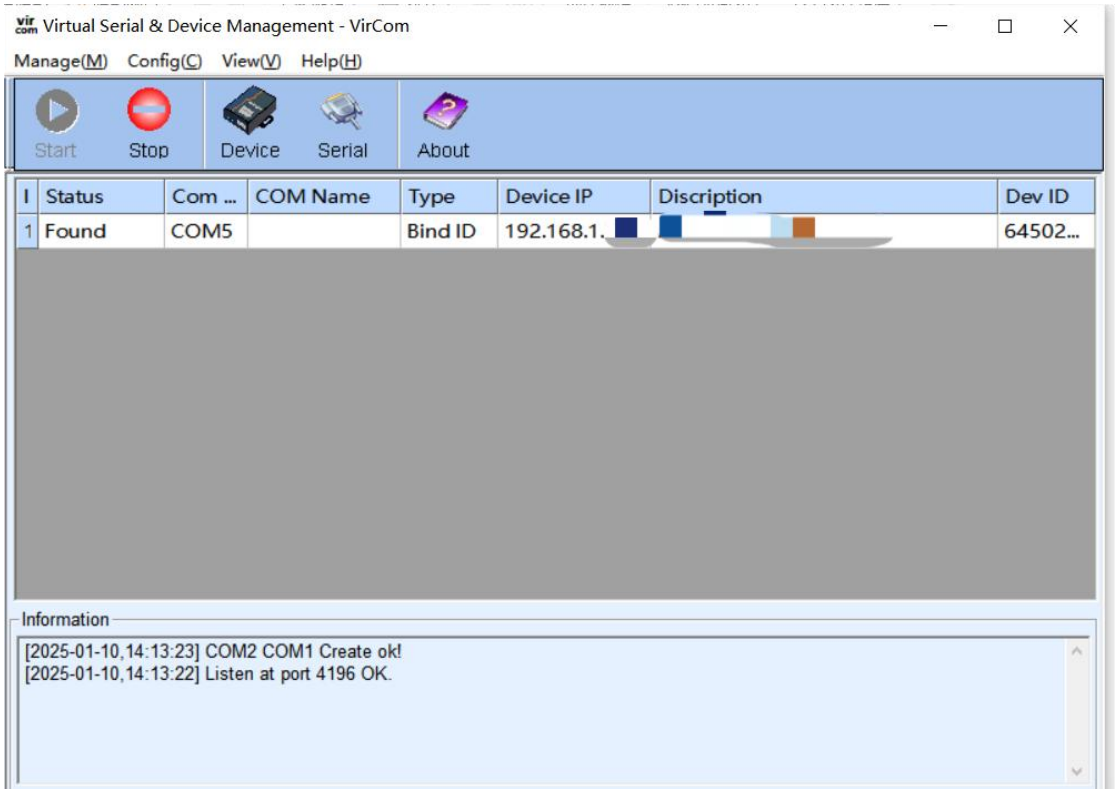


Figure 22 virtual serial port is connected

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 23 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.

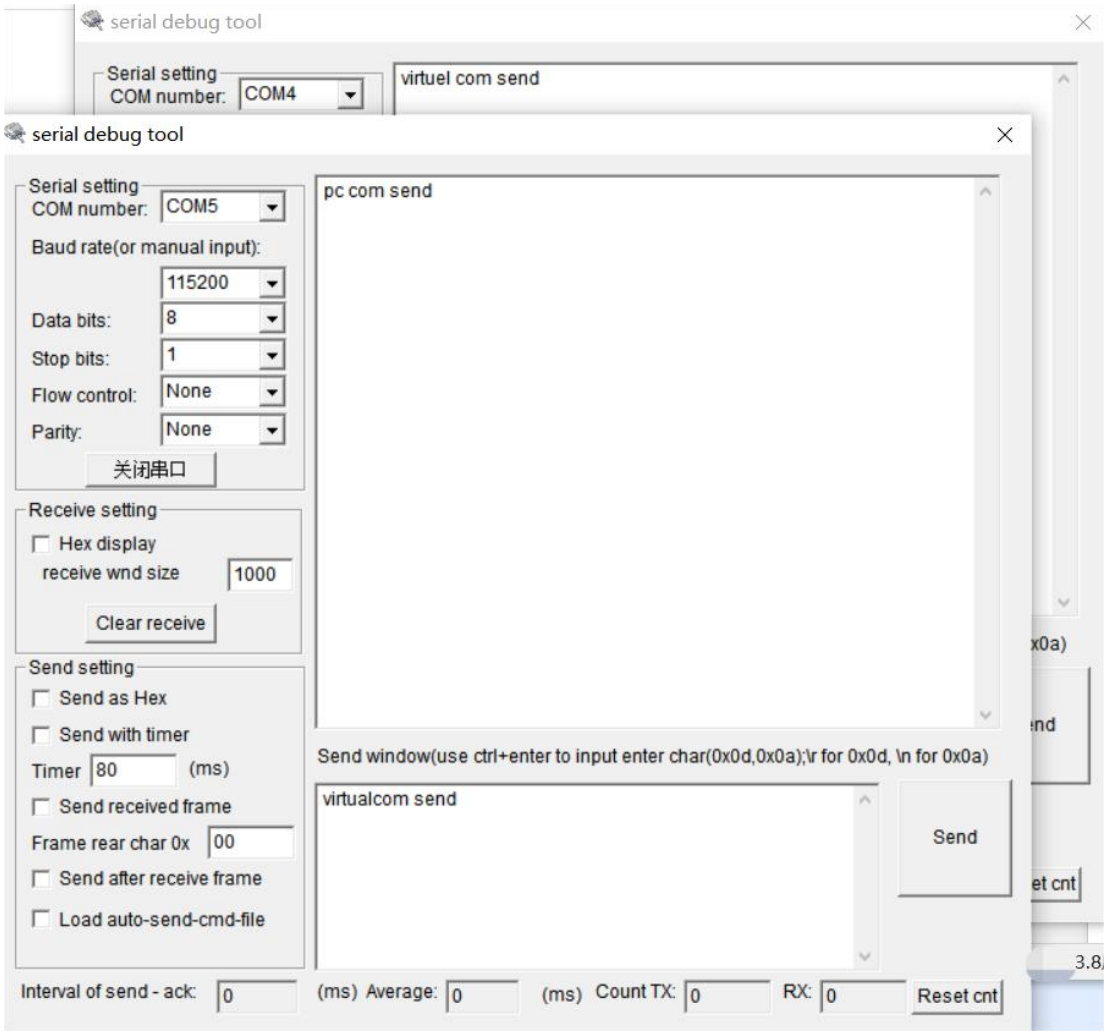


Figure 23 Communication through the virtual serial port

4.5 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 "Modbus TCPRTU" as the conversion protocol in the device setup dialog box, as shown in Figure 24. 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.

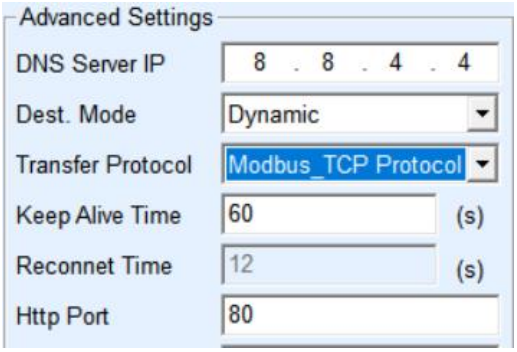


Figure 24 Enable Modbus TCP function

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 25.

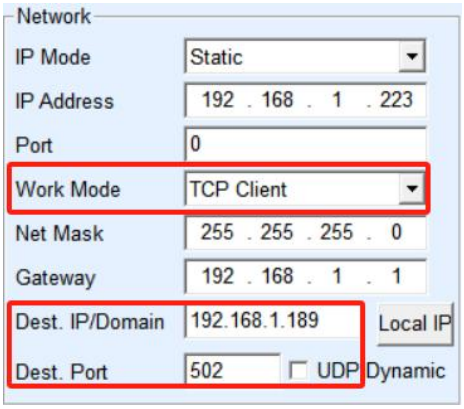


Figure 25 Modbus TCP as client side

4.6 Configuration Using the Web UI

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://192.168.1.200.

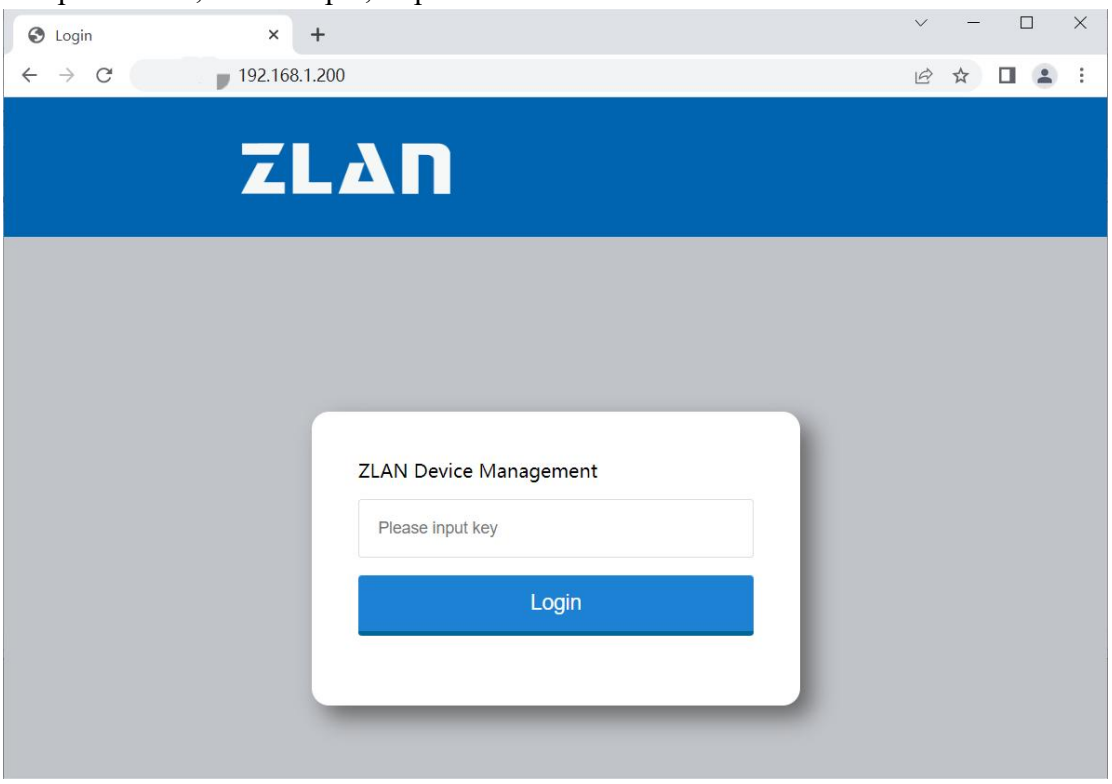


Figure 26 WEB

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

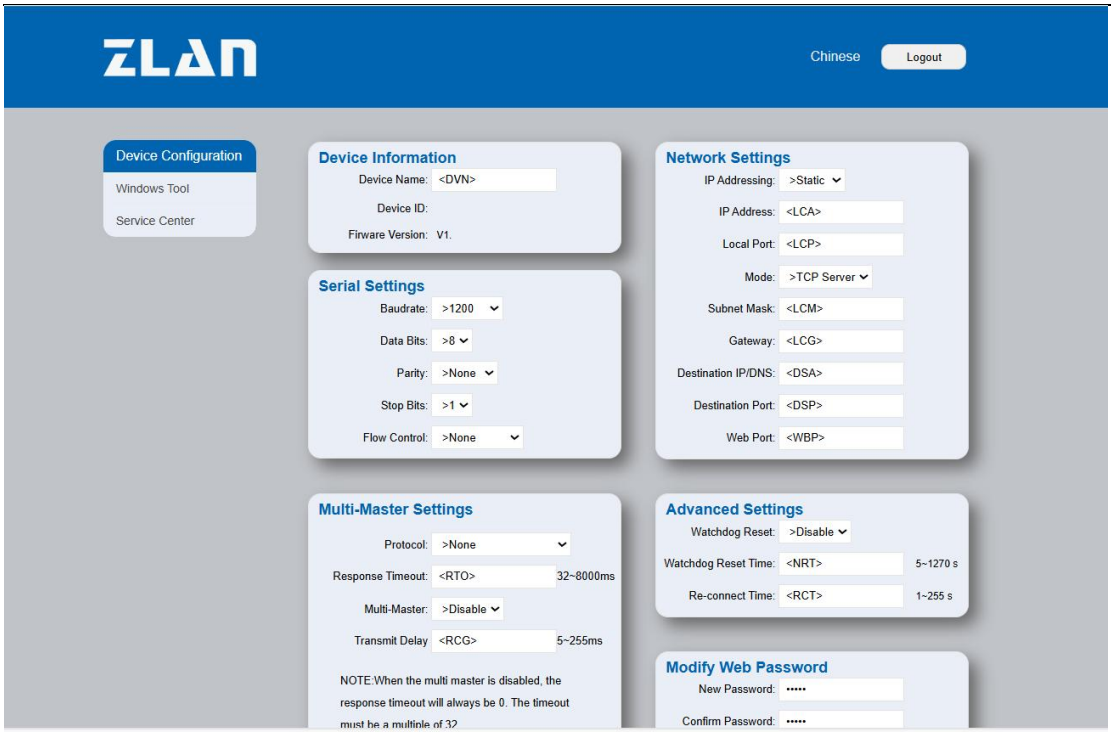


Figure 27 Web configuration page

3. You can modify serial port server parameters in the displayed web page. For details about related parameters, see Table 4.
4. Click the "Submit Modification" button after modifying the parameters.

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: with virtual serial port and non-virtual serial port, as shown in the TCP communication diagram in Figure 17 and the function of virtual serial port in Figure 20. 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 part" is divided into transparent transmission, Modbus TCP to RTU and Realcom protocol three modes. If the user software uses the Modbus TCP protocol and the lower computer uses Modbus RTU, select Modbus TCP to RTU. The Realcom protocol is used only when the multi-serial server serves as the TCP client to connect to a server and the virtual

serial port is used on the server.

Usage is summarized as follows:

Table 5 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	None	It is suitable for connecting a

		side		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 5. 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.

a)

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 ⑥ and ⑦ in Table 5.

The section "4.3TCP Communication Testing" mainly briefly 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 server complies with the standard TCP/IP protocol, so any network terminal that complies with the protocol can communicate with the serial server, ZLAN technology provides a network debugging tool (SocketDlgTest program) to simulate the network terminal to communicate with the serial 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 acts as the client, there must be three mappings, as shown in Figure 28. (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.

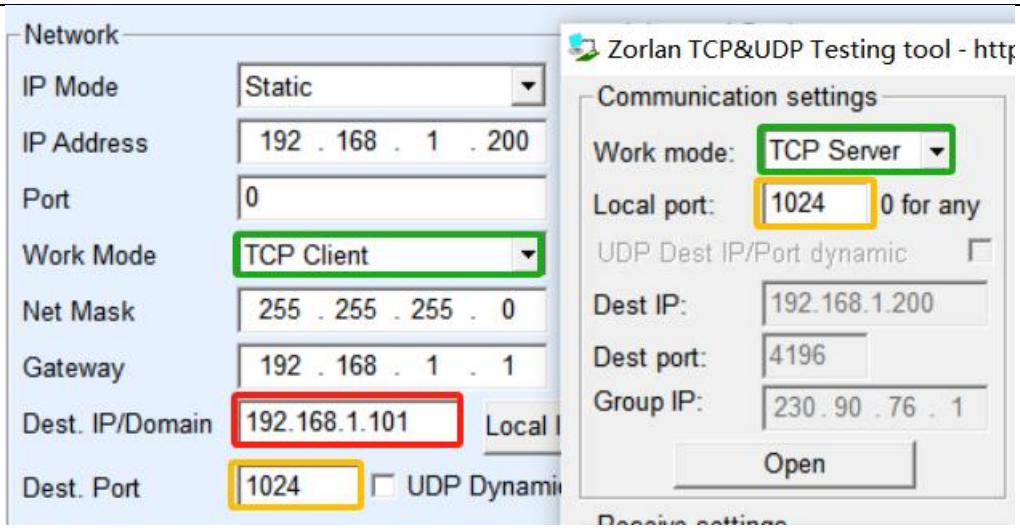


Figure 28 Serial server as client

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：

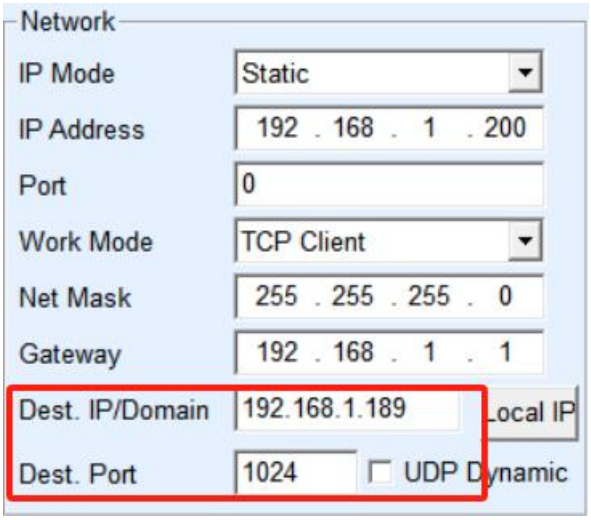


Figure 29 First destination IP address and port number

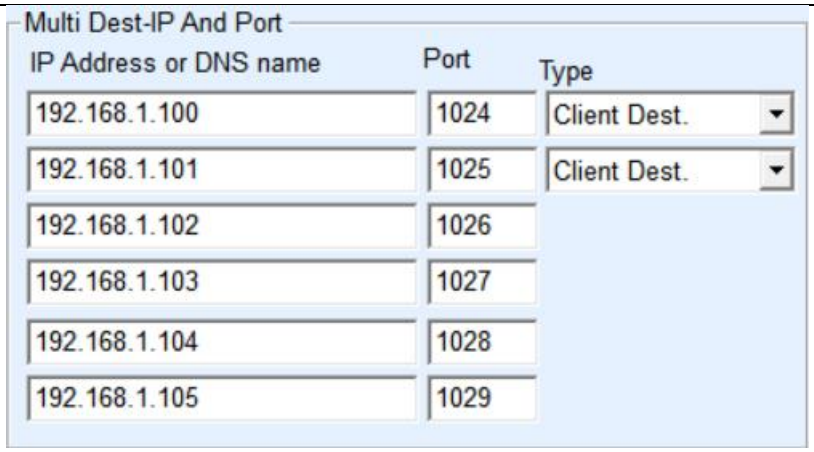


Figure 30 left 2～7 IP and terminals

The first IP address is set in the device setup screen shown in Figure 29, 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

When the serial server acts as the server, there are also three mappings, as shown in Figure 31, 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.

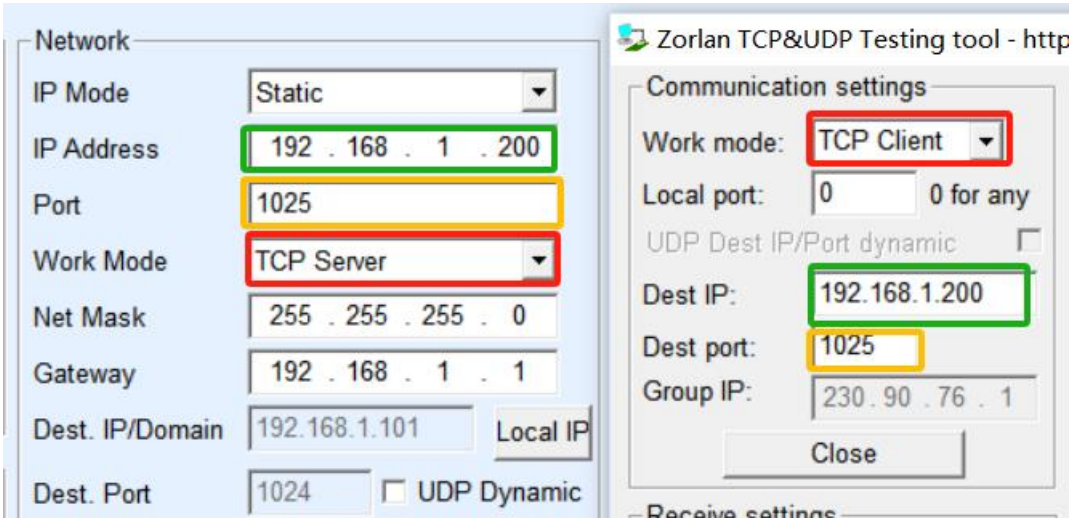


Figure 31 Serial server as server

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 recently received network packets, you need to enable the multi-host function. For details, see 7.4 Multi-Host Function.

5.2.4. Be 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.

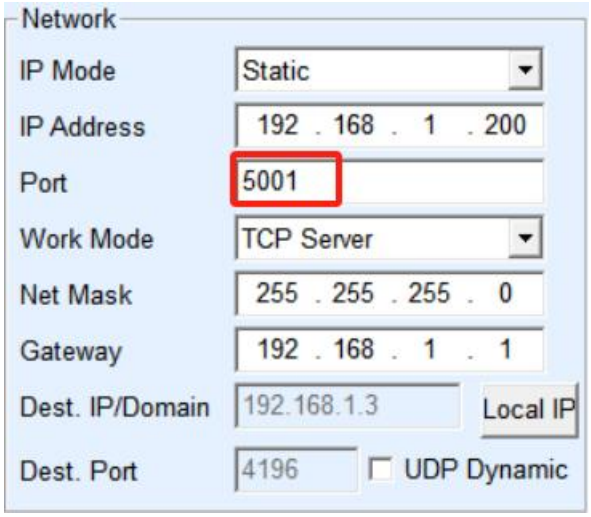


Figure 32 acts as both client and server

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 32, the computer software can now connect to the 1024 port of 192.168.1.200 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 1024 is occupied by the server, the local port used by the client is "port +1", that is, the software on 192.168.1.189 sees that the incoming port of the device is 1024+1=1025.

5.2.5. UDP Mode

In UDP mode, parameter configuration is shown in Figure 33. 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.

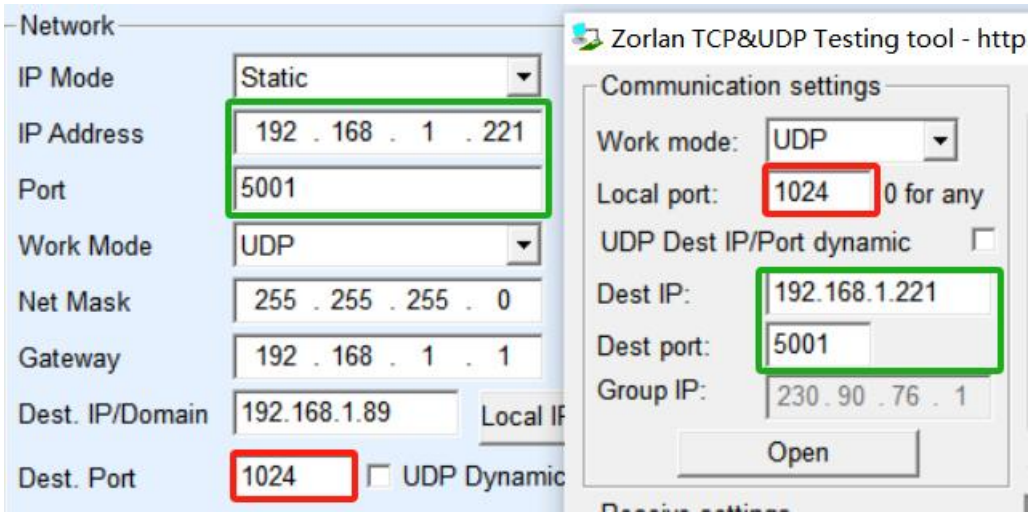


Figure 33 UDP mode parameter configuration

5.3. Device pairs

If the host computer is not a Socket program (SocketDlgTest) or a virtual serial port, but two devices are connected through network ports, the configuration method is similar. First, the user needs to connect two devices and computers to the same LAN. ZLVircom runs on this computer, the purpose of connecting the computer is only for configuration, and the computer does not need to be connected after configuration.

Click on ZLVircom's device management to find these two devices, as shown in Figure 35. Then click "Device Edit" to configure the device. Device pairs can be classified into TCP pairs and UDP pairs. In TCP interconnection mode, the parameters of the two devices are shown in Figure 34. The parameters shown by the arrows must correspond to each other, just as they correspond to the PC connection. After the TCP connection is successful, you can return to the Device Management dialog box to check the connection status. As shown in Figure 35, if the status of both devices is Connected, the TCP link between the two devices has been established.

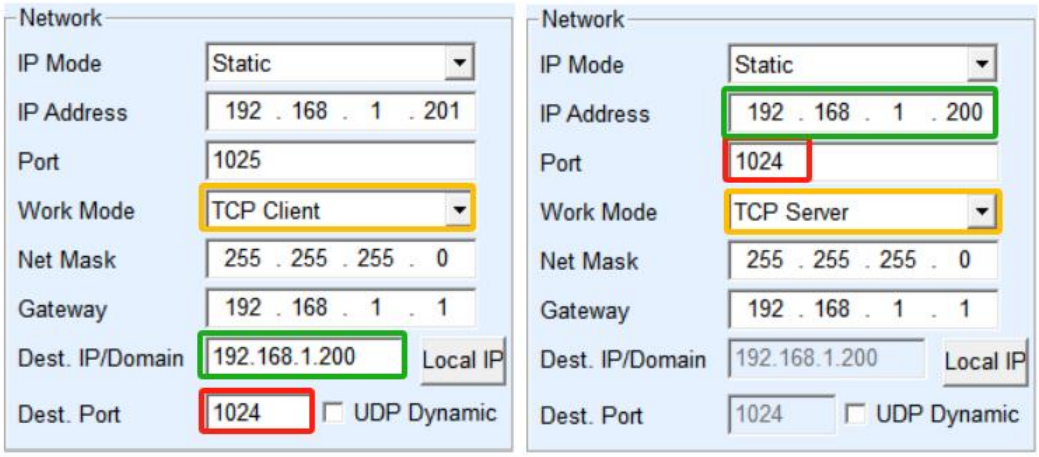


Figure 34 Configuration of TCP device pairing parameters

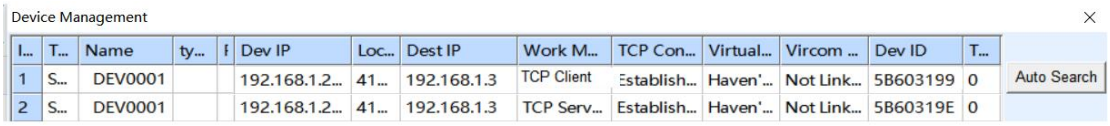


Figure 35 Successful connection check of TCP devices

For UDP couplings, the configuration parameters are shown in Figure 36. The parameters corresponding to the arrows must be one-to-one. UDP pair connection Data is automatically sent to the specified device if the parameters are correctly configured without checking the connection status.

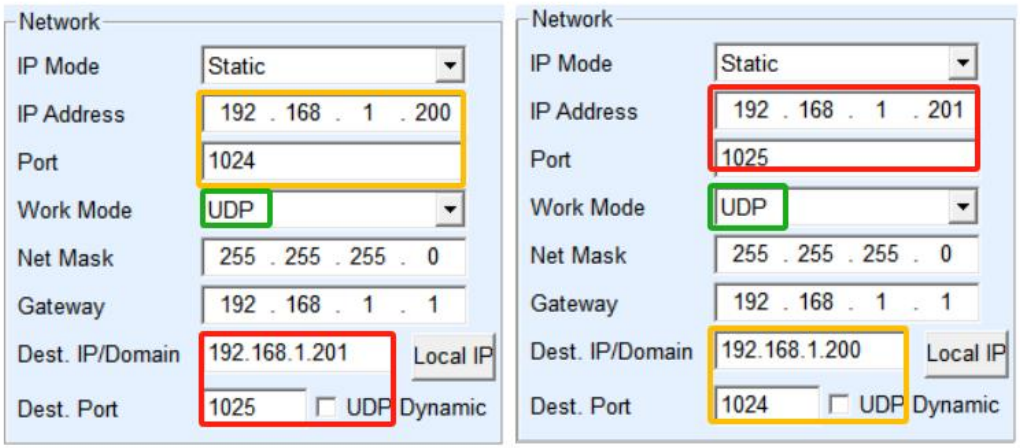


Figure 36 Configuration of UDP device pairing parameters

Finally, you need to remind that if the device is in pair, in addition to the network port parameters according to the above Settings, you must also set the correct serial port parameters. The baud rate of the serial port server must be consistent with that of the user's device. After this configuration, user devices can send data to each other through the serial ports of the two serial servers.

6. Device commissioning

6.1. Physical network connection

A serial port server can be connected to a 10M/100M switch or a computer network port using a crossover cable or a direct network cable.

After the connection is established, check whether the Link indicator is green. Otherwise, check whether the network cable is properly connected.

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.

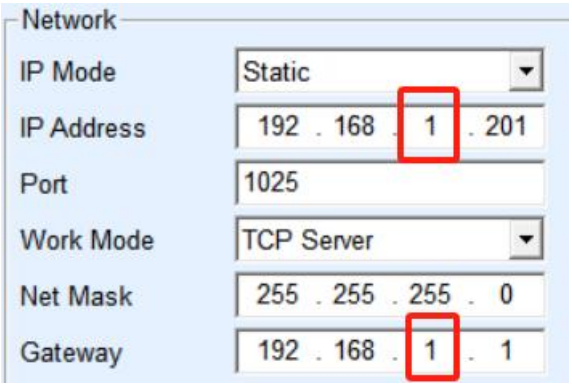


Figure 37 Configuration in the same network segment

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 37.

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, perform 4.3TCP Communication Test or 4.4 Virtual Serial Port Test. You can see that the Link indicator turns blue when a TCP connection is established. The blue Link light can also be seen through ZLVircom. For example, in the device management list, if the TCP connection is listed as "established", the Link light is blue, as shown in Figure 38, which can facilitate

remote diagnosis.

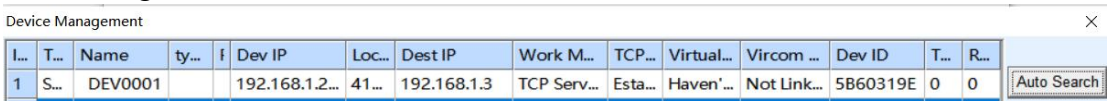


Figure 38 Connection status and data sending and receiving status

6.3. Data sending and receiving

When the Link indicator turns blue, data can be sent and received between the software and the serial port server. If the software sends data, the Active light turns green 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 responds to correct command delivery. Once a response is received (the serial port sends data to the network port), the Active color turns blue. Otherwise, check whether the serial port parameters or the serial cable are connected properly.

In order to facilitate remote debugging, ZLVircom also supports remote viewing of data sent and received. As shown in Figure 38, TXD is the amount of data sent by serial port server. When refreshing the device list, if the value changes, it indicates that data has been sent, and the Active light will turn green. If you see the value of RXD changing, it indicates that the serial device has returned data, and Active is blue.

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.

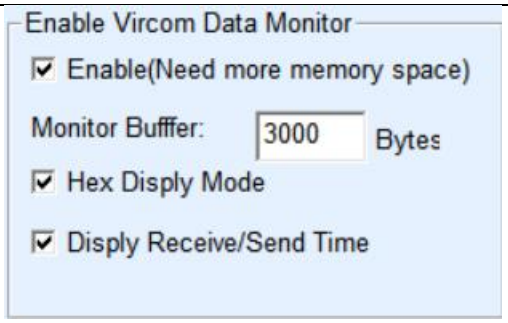


Figure 39 ZLVirocm monitoring enabled

Check the three options of enable monitoring, hexadecimal monitoring mode, and display data sending and receiving time, as shown in Figure 39. Then click OK. Assuming that data has been sent and received before, now select a virtual serial port that you want to monitor in the main interface, and then select Menu/View/Monitor, as shown in Figure 40.

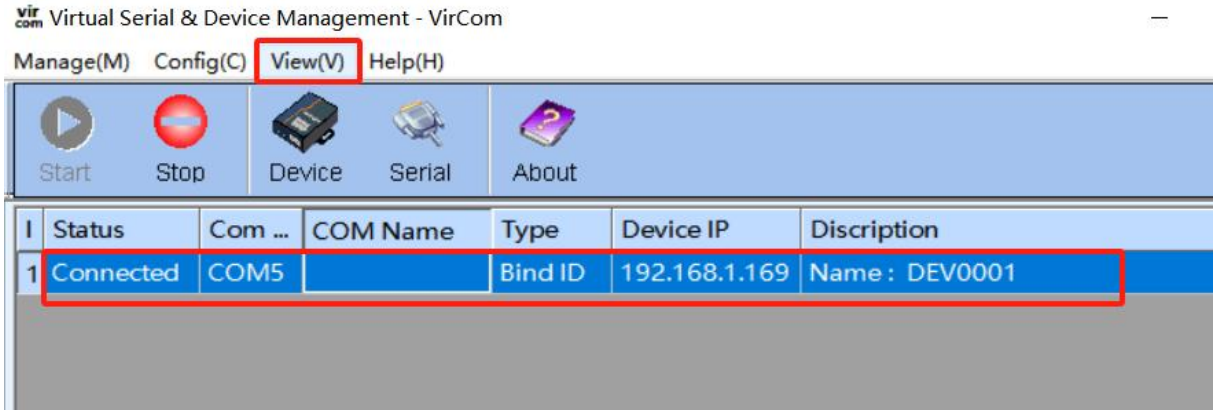


Figure 40 Open ZLVirocm monitoring

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 41. This function can facilitate the field communication debugging.

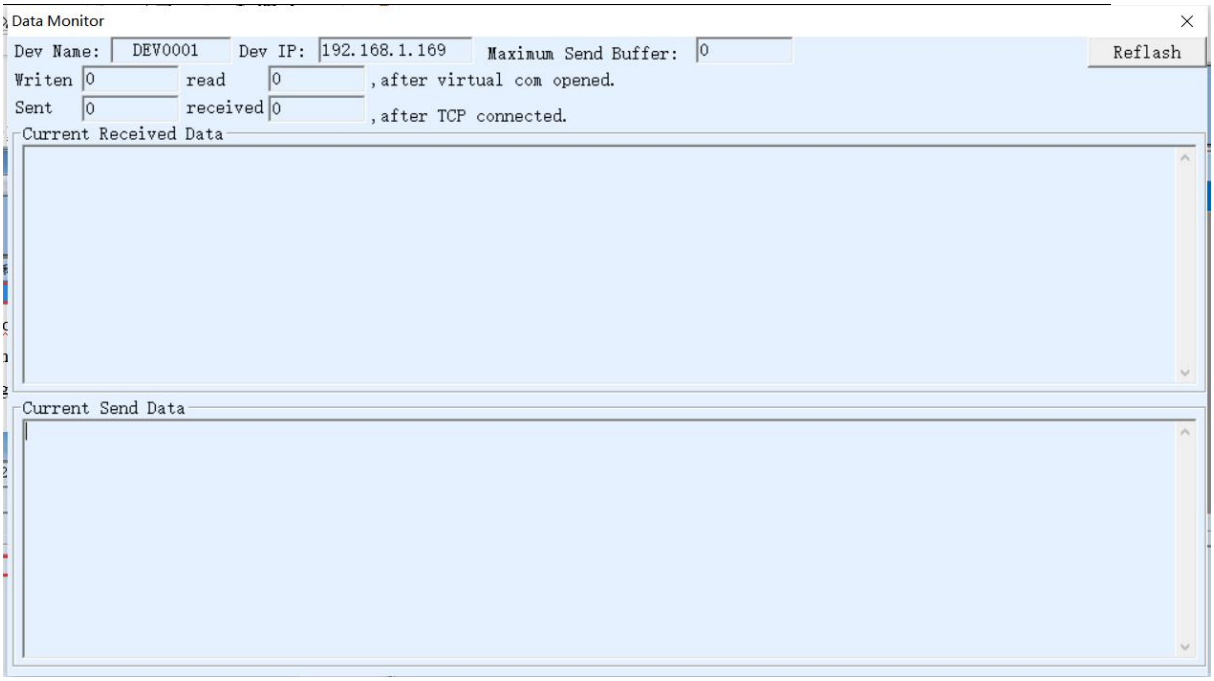


Figure 41 Monitoring and receiving data

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

The YOXO1007 can save the contents of the read register inside the gateway, so the speed of Modbus TCP query can be greatly improved, and the performance is better when supporting multi-host access.

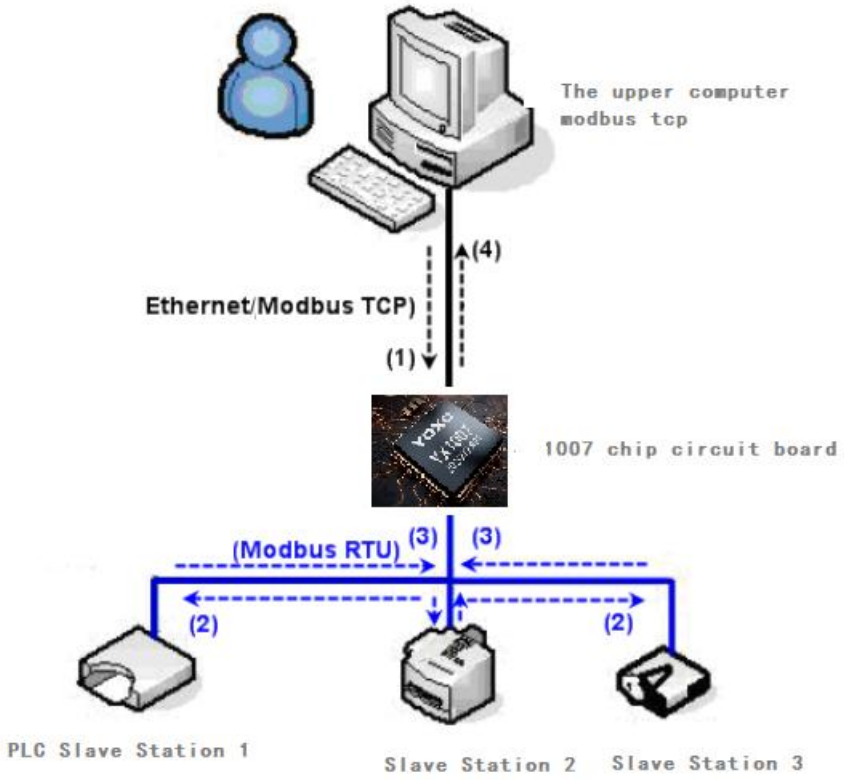


Figure 42 Storage Modbus gateway working mode

As shown in Figure 42, 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, YOXO1007 can actively send instructions from the serial port to automatically update the content of the data currently saved in the register and save a copy of the latest register value.

In addition, YOXO1007 is a fully automatic configuration free Modbus gateway chip, the user does not need to configure the required register address, function code, slave station address and so on. These registers are automatically identified and dynamically added according to the Modbus TCP command sent by the network port.

When monitored by multiple computers, TOXO1007 showed good reaction speed. No matter what the baud rate of the serial port, the Toxo1007 was generally able to send 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. Currently, 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, remove the one that is supported and the one that is enabled as shown in Figure 43 and click OK. 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.

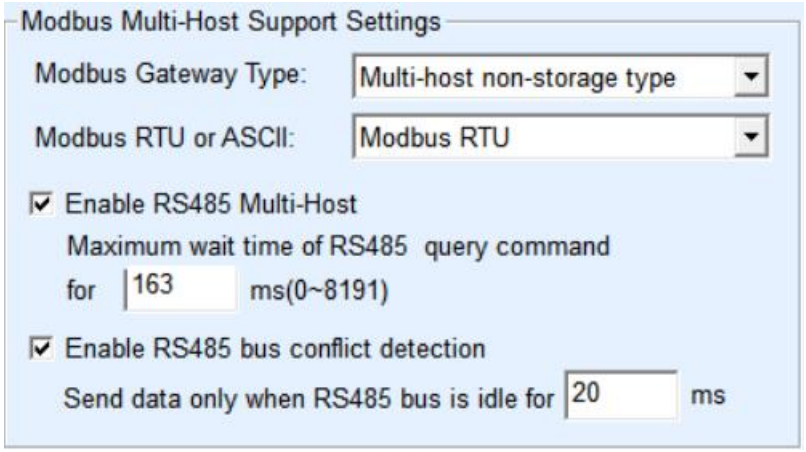


Figure 43 Disabling storage-type functionality

7.4. Multi-host function

As shown in Figure 43, "RS458 multi-host support" and "RS485 bus collision detection function" are the multi-host functions of ZLAN. 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 ZLAN serial port server can "coordinate" the RS485 bus to achieve the purpose of multi-host access.

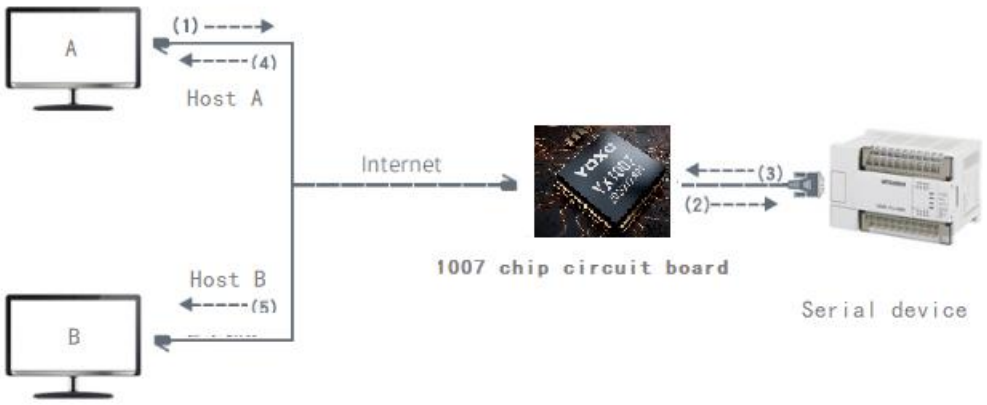


Figure 44 Multi-host function demonstration

As shown in Figure 44, in ordinary 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.

Figure 45 RS232 multi-host support

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.

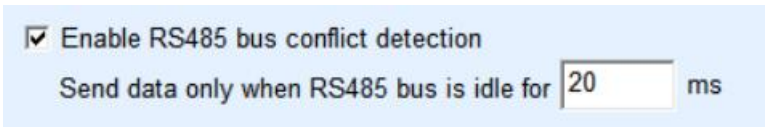


Figure 46 Free time of RS232 conflict prevention

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 46 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 45 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 model 7 must also support non-storage Modbus collection. But on the other hand, it also needs to support multiple hosts at the same time, in order to cooperate with this way, you can choose in the gateway type.

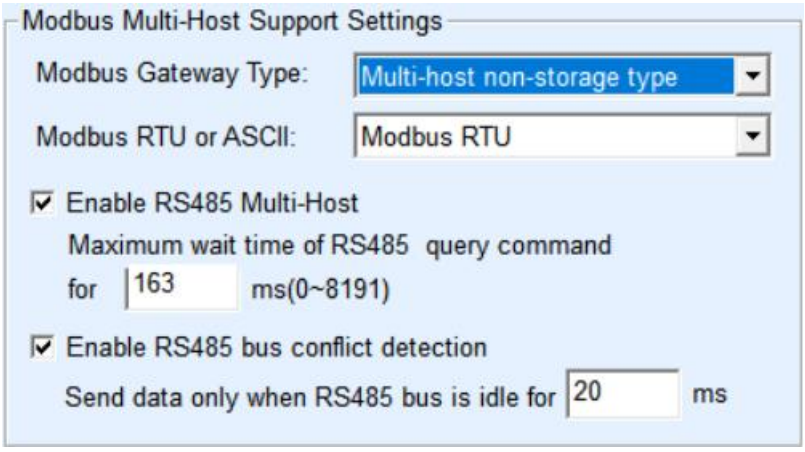


Figure 47 Multi-host non-storage setup

7.7. Modbus for Multi-destination IP Addresses

As shown in Figure 44, 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 serial port server as a client to connect to multiple network ports. For details, see 5.2.2 Connecting a Client to Multiple Servers.

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 46 and "RS485 command response timeout time"

as shown in Figure 45 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.

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 the MAC address on the connection: This method is supported not only for model 4 (such as 5143), but also for normal models. The method is to send its mac address to the cloud when the module is connected 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".

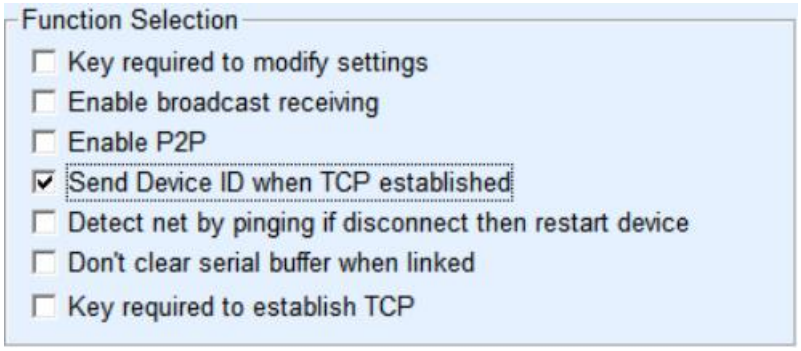


Figure 48 Send MAC address via connection

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.

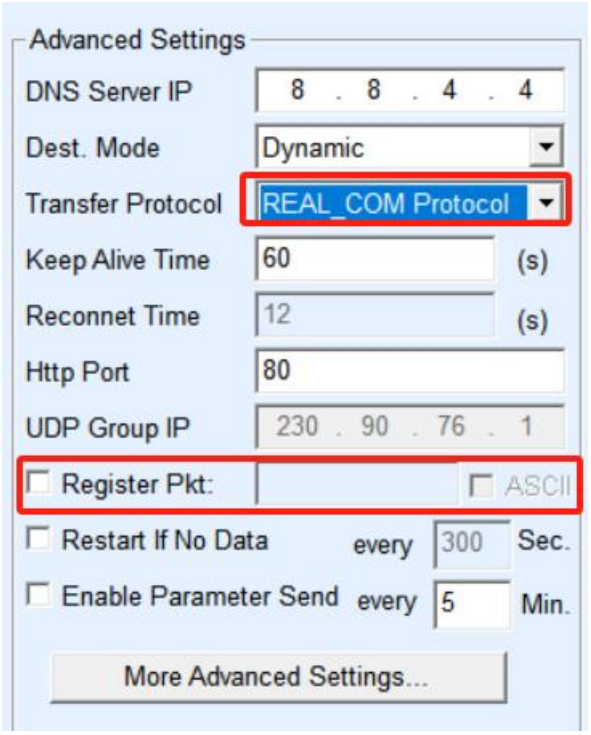


Figure 49 Enable realcom protocol

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.

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

2. 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 screen, configure as follows:

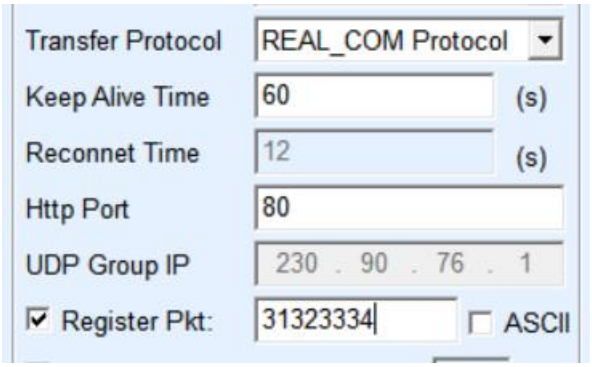


Figure 50 Set up register packet

The difference with the REAL_COM protocol is that the registration package is enabled and the registration package information such as 31, 32, 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, 32, 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

The heartbeat packet and registration are downloaded through the configuration file, and the length is longer. In the parameter configuration, click the "Firmware/Configuration" button, and in the pop-up "Configuration page/program download" click the "Registration package" button. After the configuration is complete, download it to the device.

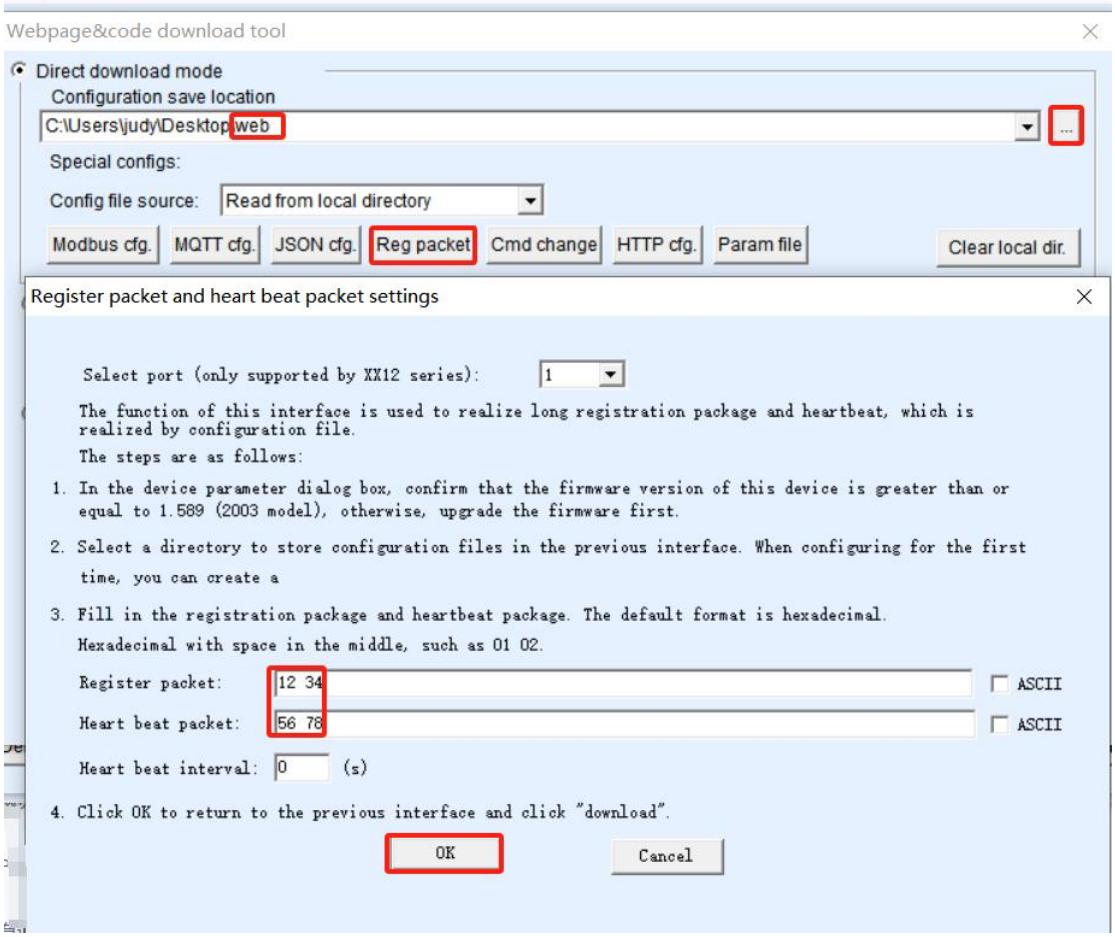


Figure 51 Registering heartbeat packets in configuration file mode

8.2. Heart beat

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.

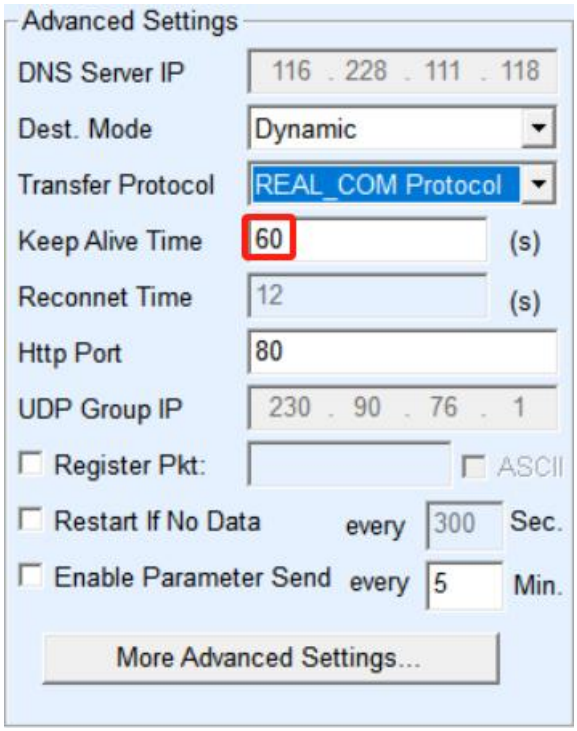


Figure 52 Keepalive timing time

As shown in Figure 52, 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 protocol

As described in 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. Add heartbeat packets as follows: Click "More Advanced Options" on the device, enter the hexadecimal heartbeat packet in the second line of the multi-destination IP address and port, and change the option on the right to "Parameter Packet Destination".

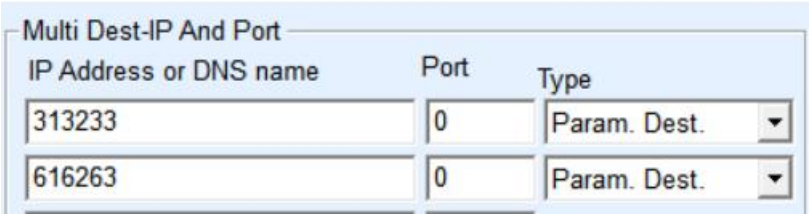


Figure 53 Customize the registration package

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

9. httpd client communication function

This function is used to send the data from the serial server directly to the server program based on the web architecture, which can simplify the software development workload in the cloud.

When the Internet of Things acquisition terminal and the web server (httpd program) interact, if the data can be submitted to the web server in accordance with the specification format of the http GET and POST instructions, then the web server can use the existing php/asp language to process and store the data. This saves the user the effort of redeveloping the web application interface.

In order to support this feature, you need to download a httpd.txt configuration file in the ZLAN serial port server. Downloads can be made using zlvircom's firmware upgrade feature.

Features of ZLAN httpd client communication include:

1. Send on the device: Support GET/POST mode to directly convert serial port data into http format, which can be directly recognized by the server.

2.Web server delivery: The Web server can also send the required data to the serial port server through the GET/POST command, and the valid data content can be output from the serial port of the serial port server. When the serial server receives the data, it can also give a specific reply to the Web server, indicating that the data is received.

3. The input and output data can be converted between hexadecimal and string, so that the Web server can send data by character, and the serial port can output hexadecimal data to control the serial port device.

For more information, refer to the ZLAN httpd Client Communication Mode documentation.

10. P2P function introduction

YOXO1007N chip has P2P function. The P2P function allows users to connect to the device anytime, anywhere through ID (non-IP mode), and without forwarding through the server, directly realizing the P2P link between the device and the computer. A public IP address and port mapping are not required on the device. It has the characteristics of easy to use and high communication efficiency.

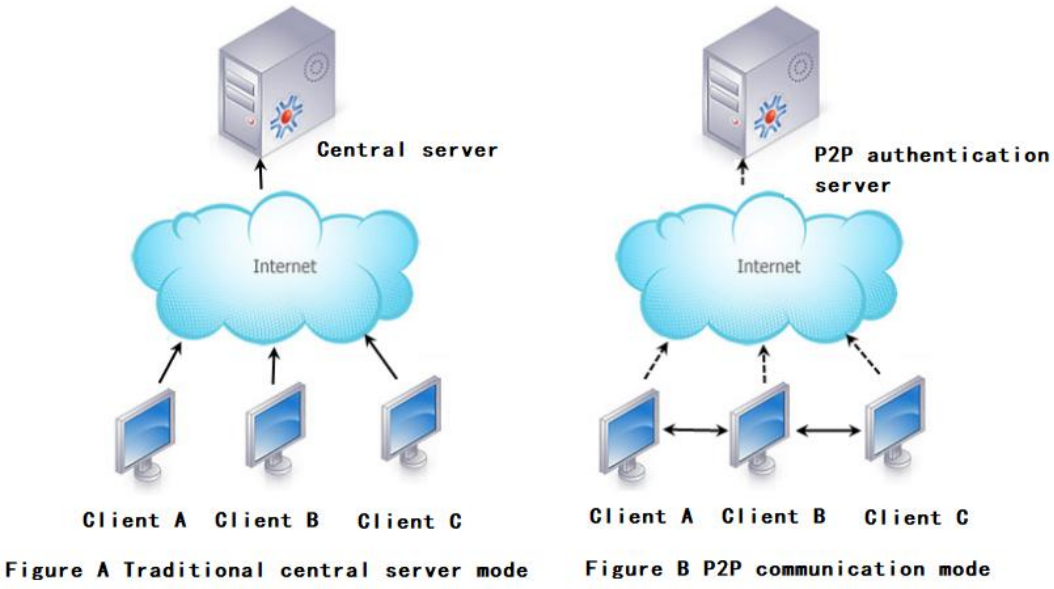


Figure 54 Comparison between traditional mode and P2P mode

Although P2P mode also has a central server, but the central server only plays the role of verification and matchmaking, and does not participate in the forwarding of data communication.

The YOXO1007N chip provided by ZLAN must be used with the specified P2P authentication server, there are two ways:

1. The YOXO1007N uses the default P2P server of ZLAN after delivery, and users do not need to configure it. Use is the client software also uses ZLAN default P2P server connection equipment.

2. Users use their own P2P server, purchase P2P server software from Shanghai ZLAN, YOXO1007N will point to the user's own P2P server.

In addition to the YOXO1007N chip, ZLAN can provide:

1. Using P2P protocol development library on PC, it is convenient for users to integrate P2P functions into their own PC software.

2. ZLVircom, a virtual serial port software supporting P2P device management and based on P2P, is provided.

The YOXO1007N chip can provide users with a simple device networking solution, allowing users to connect devices around the world through ID without building a complex cloud platform. For details, refer to the ZLAN P2P Introduction document.

11. Modify network port parameters

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

Network port modification parameters are implemented through the UDP Management Port Protocol, for example:

1. The computer software sends UDP broadcast packets whose destination port is 1092 on the network. When the device receives the packet, it will return its information to the computer software to search for the device.

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

For details about network port modification parameters, see the ZLAN Network Products UDP Management Port Protocol. It can also be implemented directly using the device management library of the 12 Device management library.

12. Device management library

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

The "UDP management port protocol" has been integrated in the device management function library ZLDevManage. This is a DLL windows platform development library, can be called by VC, VB, Delphi and other development tools.

Provide detailed API interface documentation and VC call Demo cases. It can realize device search, parameter modification, P2P function call and so on.

Development libraries can be obtained from the ZhuoLan's official website: http://zlmcu.com/download.htm looking for "equipment management function library" page. For details, please refer to "ZLAN WinP2p and Device Management Development Library"

13. Modify serial port parameters

You can read and set parameters by sending commands 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 serial ports. You can set the following parameters: IP address, baud rate, device name, and working mode. After the new parameters are set, you can restart the serial port server by using the serial port command.

ZLAN serial command has the following characteristics:

1. Serial port instructions use 10 bytes of data lead code, so there is no need to pull down the configuration pin to distinguish between communication data and commands, and there is no need to switch between command mode and communication mode, which is more flexible and convenient to use.

2. The command set contains commands such as save parameters, do not save

parameters, and restart the device.

3. It can realize a variety of applications, such as reading the MAC address of the serial port server, such as changing the working mode of the serial port server. When switching from the TCP server to the TCP client mode, it can actively connect to the server; When switching from a TCP client to a TCP server, you can disconnect from the server.

For details about how to modify serial port parameters, see the Serial Port Modification Parameters and Hardware TCPIP Protocol Stack.

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.)

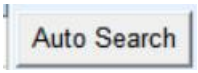


Figure 55 automatic search

2. Manually add: There are two situations:

Figure 56 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.

c)

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.

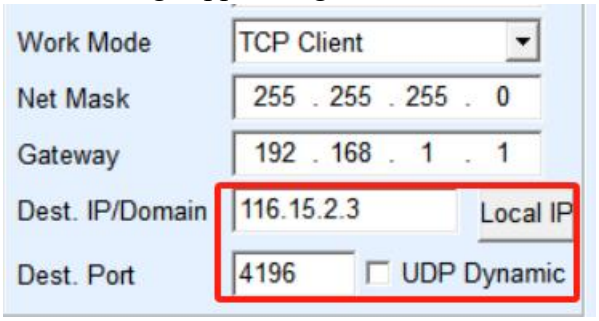


Figure 57 Client

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.

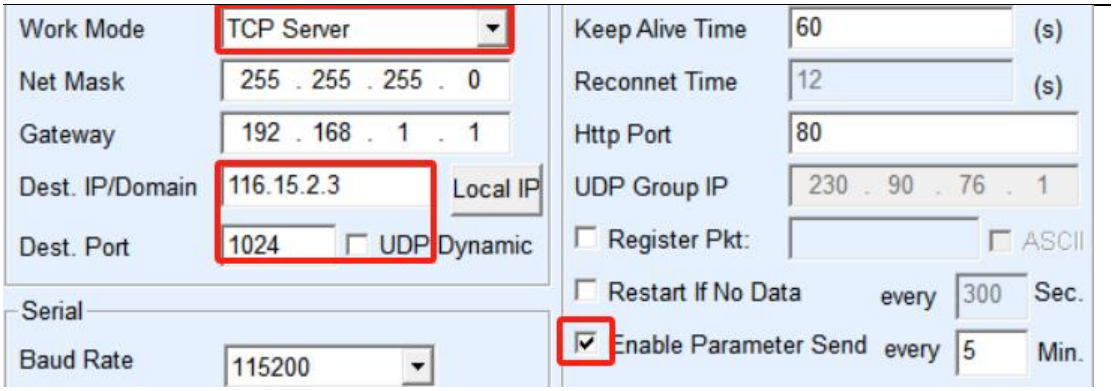


Figure 58 Timing transmission parameters

To facilitate the identification of devices, if remote management is required, please name the device something easy to remember.

15. Firmware upgrade method

Supports firmware upgrade over the network. 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 1 Get the firmware file for YOXO1007 from ZLAN, such as 1.440 (1007).bin.

2 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 59 Upgrade button

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

Figure 60 Upgrade button

3 Select the "Program File Download" option, as shown in Figure 62. 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.

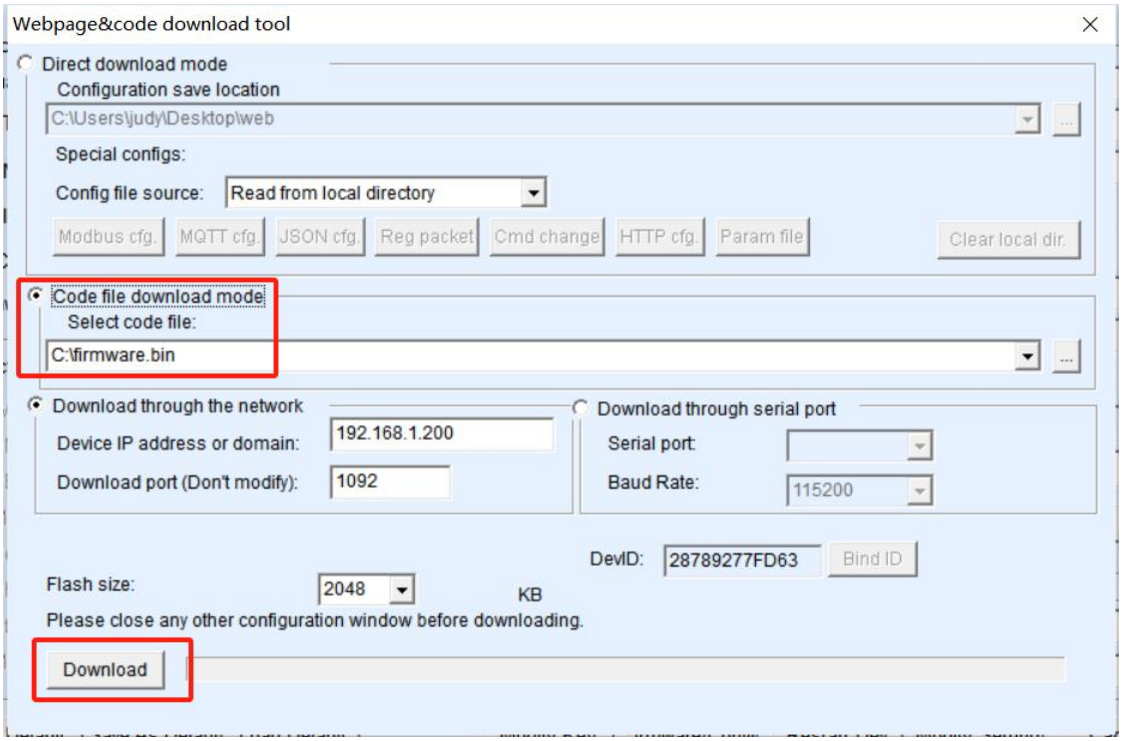


Figure 61 Firmware upgrade method

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 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 Web 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. The method of downloading Web is: change the download mode of

"program file" to "Web directory download", as shown in Figure 62. 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 ZLAN), and click Download to download all files in the local web page directory to the internal file system of the device.

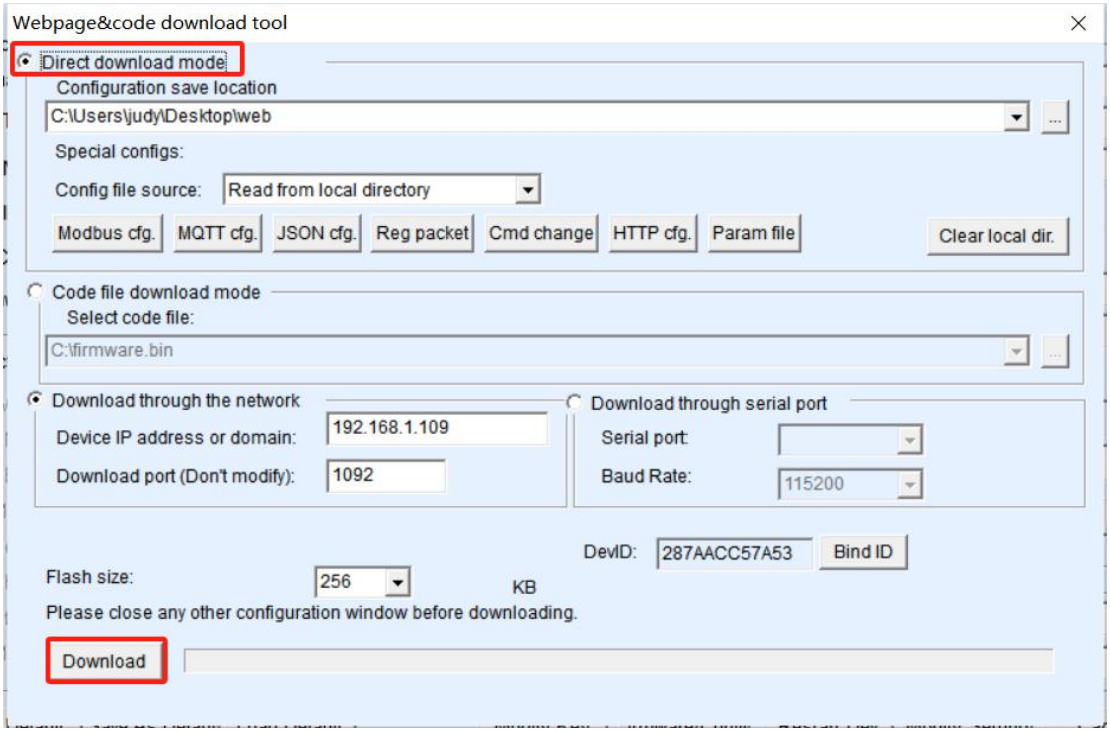


Figure 62web upgrade method

7 Note:

7.1 f 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.

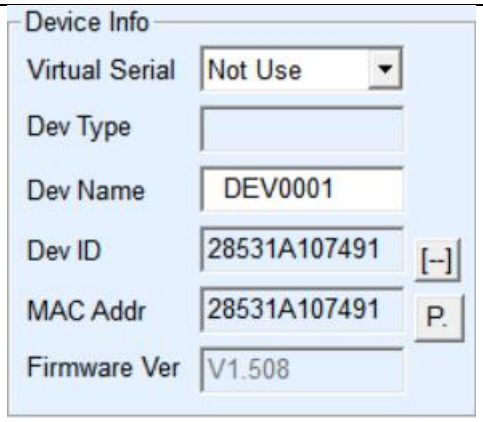


Figure 63 Checking the firmware version after the upgrade

16. Appendix 1: Detailed configuration parameters

Some of the parameter Settings are relatively complex, and are listed here in detail.

16.1. Keep the timing period alive

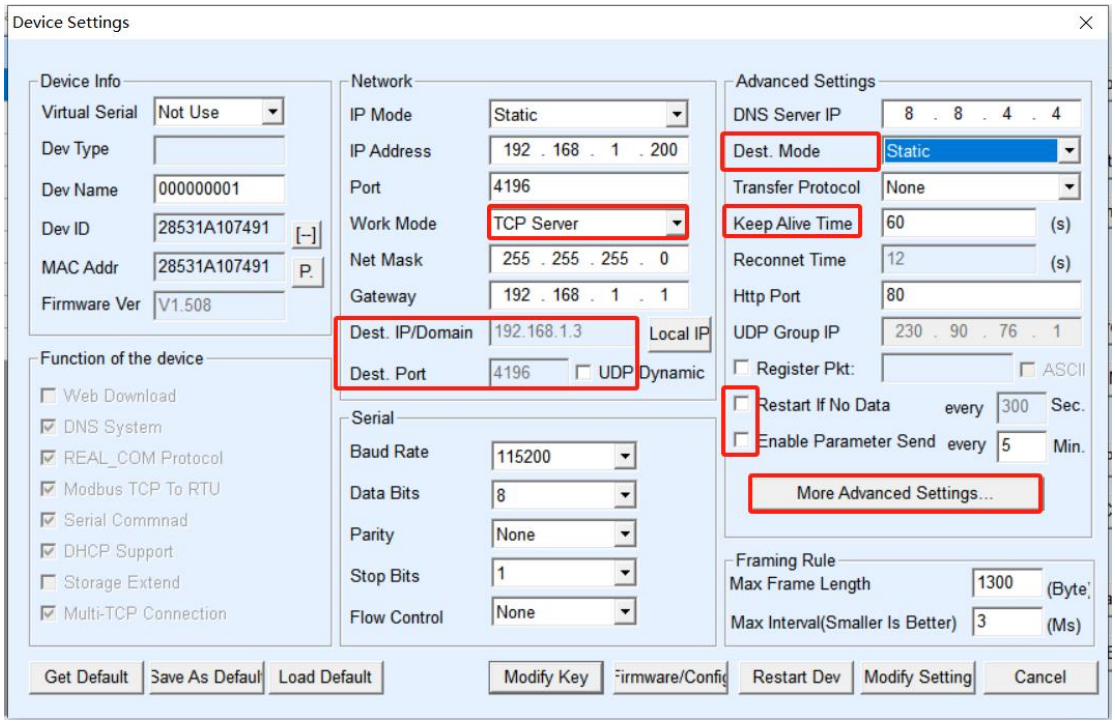


Figure 64 Settings related to keepalive timing

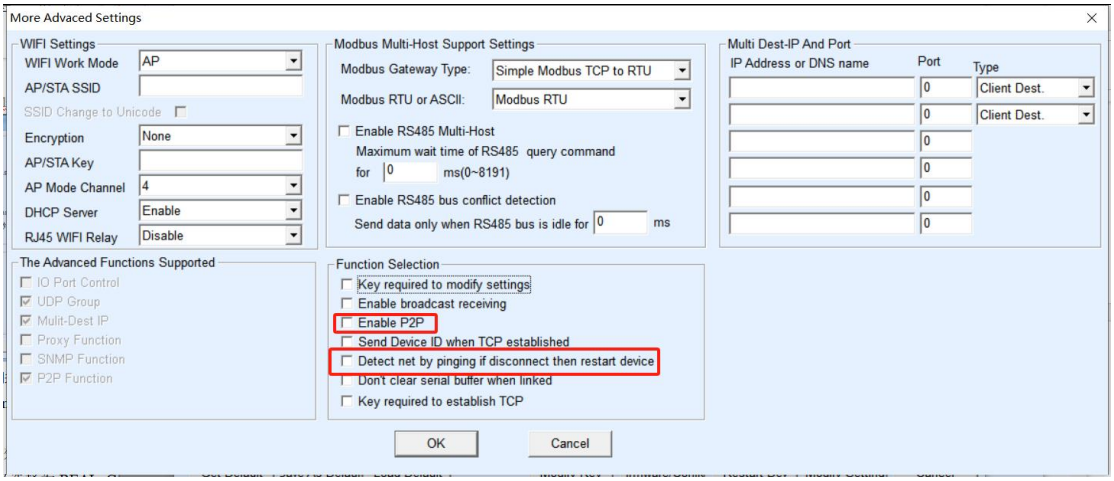


Figure 65 Advanced Settings related to the keepalive timing

1. TCP Heartbeat Interval: When the working mode is TCP Client and the Keepalive interval is set to 1 to 255, if the device is in TCP Client working mode, it automatically sends TCP heartbeat packets at Keepalive Interval. This ensures the TCP validity of the link. If the value is set to 0, there is no TCP heartbeat.
2. REAL_COM Heartbeat interval: If the conversion protocol is REAL_COM and the keepalive interval is set to 0 to 254, the device will send data with length of 1 and content of 0 at every keepalive interval to implement the Realcom heartbeat mechanism. If the value is set to 255, there is no realcom heartbeat.
3. Client send parameter interval: When the device is a TCP Client and the Keepalive interval is set to 0 to 254, the device sends device parameters to the destination computer at keepalive interval. If the value is set to 255, no parameter sending function is available. The parameters received by the server enable remote device management.
4. Restart Time without data: If restart without data is selected, the device will be restarted every 5 times the keepalive interval (60 x 5 seconds by default). If no data is available, the device will be restarted. There is also a time input box after the data-free restart, but this box is not independent, it is 5 times the time of the alive timer.
5. ping network disconnection check time: If the ping packet disconnection check is enabled, the server is pinged every Keepalive Interval (60 seconds by default). If no ping response is received for three consecutive times, the device is restarted. The domain name or IP address of the ping server is set in the destination IP address or domain name. The destination port can be any.

16.2. Disconnection time

1. Disconnection and reconnection time: By default, the disconnection and reconnection time indicates the interval for reconnecting to the server when the first connection fails in TCP Client mode. However, the device is connected immediately after the first power-on without waiting for the disconnection time. If the value is set to 255, no reconnection is performed.
2. 4G device reset: When the device is used as a 4G device (the destination mode is static, that is, working in 4G device mode), if the connection fails for 10 consecutive times, the device automatically restarts. That is, if the server cannot be connected, the disconnection time determines the reset and restart time of the 4G device. By default, the device restarts after 12 seconds x 10=120 seconds.

16.3. Set the sending parameter time

1. Periodic sending parameter interval: If the device is in TCP Server mode and needs to send parameters to the destination server, select Enable Periodic Sending Parameter Parameters. Then, the device sends parameters to the server at a periodic interval (unit: minute).

16.4. P2P heartbeat time

When serving as a 4G device (the destination mode is static, that is, working in 4G device mode), in P2P mode, the device sends a P2P heartbeat to a fixed server every 30 seconds. If 30 heartbeats (about 15 minutes) are not answered, the device restarts automatically.

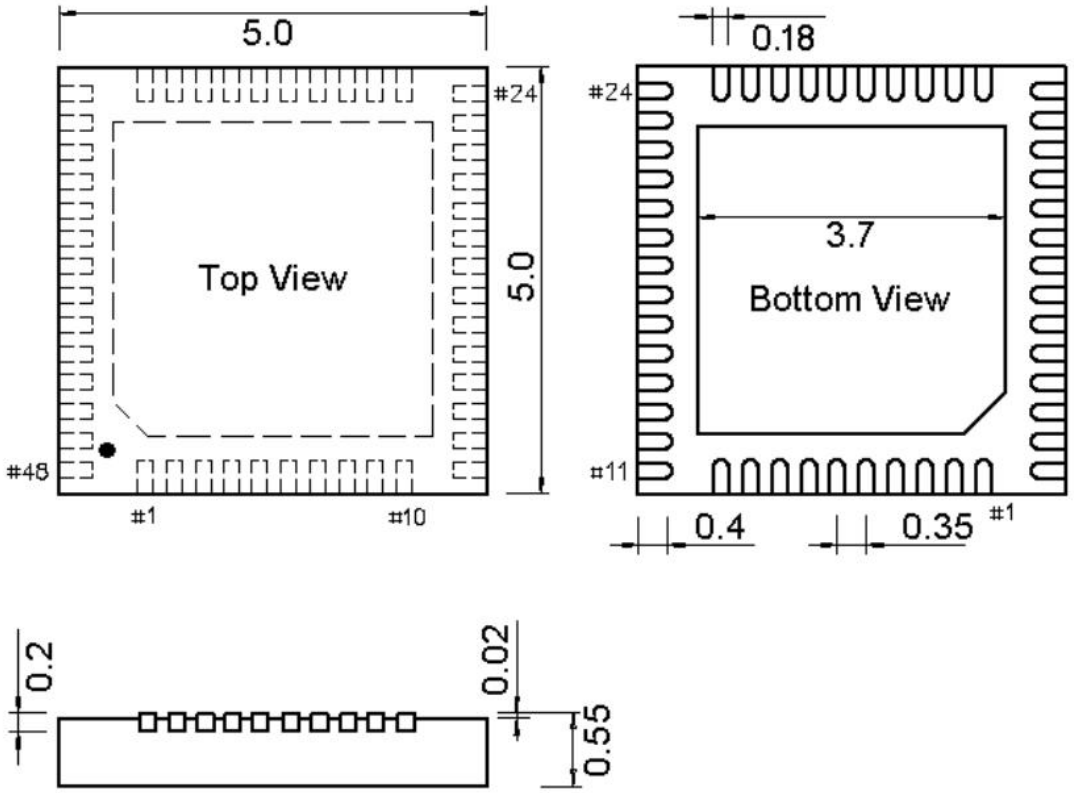
17. Appendix 2: Electrical characteristics

name	Typical value	Instructions
Working current	25mA	
VCC3.3V	3.3V	The limit value is not higher than 3.9V
IO port withstand voltage	3.3V	The limit value is not higher than VCC3.3V+0.4V
Ambient temperature at storage time	/	-40°C到120°C

Ambient temperature at work	/	-40°C到85°C

18. Appendix 3: Package dimensions

QFN48 package as shown below, unit mm. The size error is not more than 0.2mm.



19. 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