ZLAN1003/ZLAN1043/ZLAN1043N User Manual

Single Chip Serial Port to TCP/IP Solution

Document DI: ZL DUI 201303271.1.0

Version Information

The History of the revision to this document:

History

Date	Version	Document ID	Revising content
2013-03-27	Rev.1	ZL DUI 201303271.1.0	Release
2013-07-16	Rev.2		Increase the power consumption control method
2018-03-05	Rev.3		

Copyright information

Content

1. Summary

1.1 Introduction

ZLAN1003/ZLAN1043/ZLAN1043N is a series of chips for IoT launched by Shanghai ZLAN. It has a 10M/100M Ethernet interface (internally integrated PHY function), a common rate UART, a high-speed UART, and some configurable functional pins. The basic function of ZLAN1003 series is to realize the protocol transformation of UART to TCP/IP, as transfer the data received by UART to the cloud/computer, and the data sent by the cloud/computer is forwarded to UART. The implementation of these functions no need user development program, as the internal procedures of ZLAN1003 are already solidified, user only do configuration can use it. It can be regarded as a single-chip level serial port to Ethernet chip, IoT chip.

Figure 1 ZLAN1003

It has already integrated the various functions required for IOT, including:

1) Search and configure the parameters of ZLAN1003 via the Ethernet port;

2) Download the customized Web via the Ethernet port, and configure IP and baud rate and other parameters through the Web;

3) The working mode of ZLAN1003 can be configured to automatically connect 7 cloud servers;

4) ZLAN1003 can be configured with registered package and heartbeat package function;

5) It can be configured as HTTP mode and can be directly connected with the asp/PHP GET/POST command in the cloud;

6) It can be configured as "Transcoding" function, which can automatically send instructions by serial port, acquire the instrument data and then upload according to the customized format;

7) With Modbus gateway function and Modbus TCP to Modbus RTU;

8) support SNMP protocol, which can convert various serial port protocol into SNMP protocol;

9) ZLAN1043N supports cross-intranet communication in P2P way, and realizes access to Internet via ID anytime and anywhere.

1.2 Block Diagram

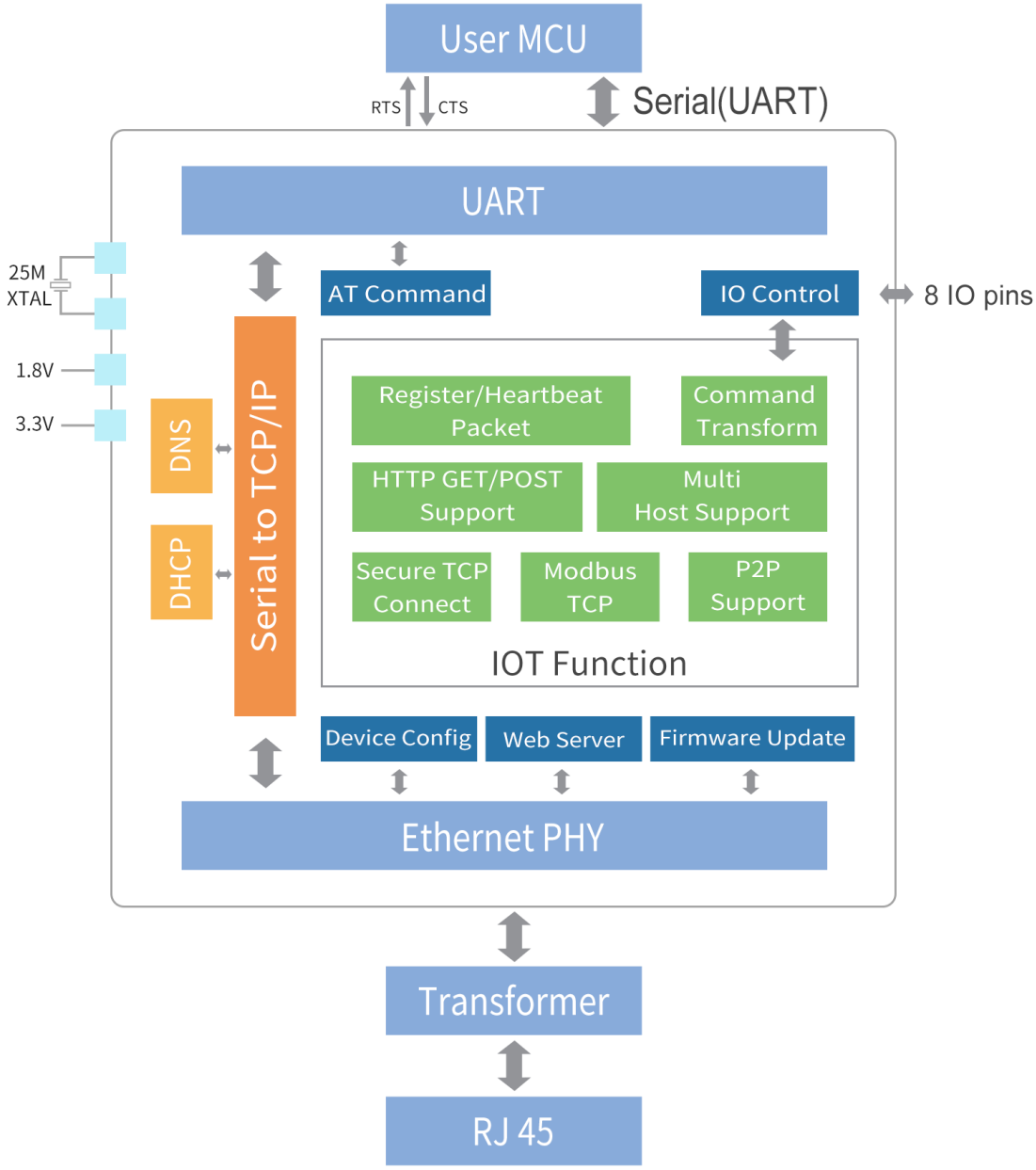


Figure 2 ZLAN1003 Block Diagram

The internal firmware program of ZLAN1003 series is already solidified, no need for secondary development, but can be upgraded through the Ethernet port. The external interface of the chip is mainly UART, Ethernet and IO interface. ZLAN1003 internal function modules include: function of chip parameters of network configuration, function of serial port configuration through AT-like command, function of configuration through embedded Web, function of IO controlling input and output pin, etc.

1.3 Features

1.3.1 Hardware Features

- Single chip of high integration: internal integrated integrate MAC and PHY interface of 10M/100M fast Ethernet, no need to extend RAM, FLASH and PHY, only need to externally connect the resistance capacitance.
- Support Auto Negotiation Full-duplex&half duplex. Support for automatic cross line detection (MDI/MDIX)
- 3.3V working voltage, 5V I/O signal tolerance. Can be 3.3V/1.8V dual power supply or 3.3V single power supply.
- Baud rate of UART1 support 1200~115200bps in high speed mode, support 1200~38400bps in low speed mode. The data bits of UART1 support 5~9 bits, the parity bit can be None, Odd, Even, Mark, Space five types, support hardware flow control and software flow control. UART2 support 1200~921.6Kbps.
- 80 feet LQFP lead free encapsulation.
- Abundant signal light: indicator light for TCP connection establishing, wires connecting, data communication.
- Support 485 sending permit control line RS485_EN pin used on RS485 bus.
- Industrial chip for temperature -40℃ to 85℃.

1.3.2 The Software Features

- Internal solidify serial port to TCP/IP software of full function, no need secondary development.
- Support TCP server, TCP client, UDP mode, UDP multicast. Support TCP Server function when as TCP Client. Support 30 TCP connection when as TCP Server, support 7 destination IP when as TCP Client.
- Support chip connect to send MAC address function, convenient for managing devices in cloud.
- Provide secondary SDK DLL development library of searching and configuring chip in PC.

- Support Web browser configuration, support DHCP dynamic access IP, connecting domain name server address via DNS protocol.
- Support remote searching chip, configuring chip parameters, upgrading chip firmware program in cloud.
- Supports remote checking the TCP connection status, serial port data send-receive status of chip via software. Virtual serial port supports data monitoring.
- Support sending AT-like command to read and control chip parameter through serial port.

1.3.3 ZLAN1043 Software Features

The following functions are not supported by ZLAN1003, only ZLAN1043/1043N support:

- Support Modbus gateway function, support Modbus RTU to Modbus TCP. Can support storage Modbus, can automatically collect device data and store it. Also support non-storage Modbus gateway.
- Support multi-host function: in the query-answer query mode, support multiple computers to simultaneously access the same serial port device at Ethernet port side.
- Support for custom heartbeat packets and register functions: easy to communicate with the cloud and device identification.
- Support requiring password authentication function in TCP establishing connection to ensure connection security.
- Support the "Transcoding" function, can realize the translation work of the protocol to the specific device, let the different device connect the unified software platform.
- Support data submission and release in HTTP method, the cloud can directly use HTTP GET command to interact with serial data of the device.

- Support controlling level of 8 IO ports through arbitrary network instructions.
- In addition, ZLAN1043N supports network NAT crossing function in P2P.

1.4 Advantages

The ZLAN1003 series has four advantages as follows: zero software development, integrated abundant IOT function, some applications without external MCU, seamless connection with the original serial port procedures.

1.4.1 Zero software development

Figure 3 ZLAN1003 Application Diagram

ZLAN1003 series chip internal program does not need the user secondary development, the UART that joint the user MCU directly sends and receives the data so can transmit data to the computer program on the network. The user use steps are as follows:

1) ZLAN provides the complete circuit diagram of ZLAN1003 chip. The user is designed according to this circuit diagram and welding ZLAN1003 to the circuit board.

2) After ZLAN1003 circuit boards power on, the RJ45 port of circuit board of access to network, any computer in the network search device through ZLVircom or user's own development program with network, also can use ZLVircom to check the IP and port of ZLAN1003, can configure the IP and baud rate if necessary. Connect the Socket (TCP/IP) software to the device IP and port.

3) The relevant indicator light indicates whether ZLAN1003 and the computer are

establishing TCP/IP.

4) After the connection is established, the data sent to ZLAN1003 by the Socket (TCP/IP) software will be sent to the user MCU by UART, and the data of the user MCU can also be sent to the computer Socket (TCP/IP) software through UART.

5) If necessary, the user MCU can read the MAC of ZLAN1003 and modify the IP by the AT-like instruction.

There is not a bit of software development in the above steps, and even hardware development offers a complete circuit diagram. Only need to use ZLVircom configure the parameters like IP and baud rate in the first time and stored you can directly use, really achieve the zero software development.

It is generally possible to complete the development of a IoT product based on Ethernet. This zero software development feature provides a quick solution for the quick launch of user products, and also avoids problems such as long software development time and unstable software development.

1.4.2 Integrated rich IOT functionality

ZLAN1003 has already integrated many IOT functional modules, reducing the workload of user developing related IoT functions. Here are a few typical features:

1) Registration packet and heartbeat package function: when the IoT device joint with the cloud it need to send its ID to the cloud software to facilitate the device number. ZLAN1003/1043 has multiple registration packet and heartbeat packet style.

2) ZLAN1043 can be configured as HTTP mode, which can be directly connected with the asp/ PHP GET/POST command in the cloud.

3) ZLAN1043 can be configured as a "Transcoding" function, which can automatically send instructions by serial port, and then upload the instrument

data in a customized format. It is convenient to convert different devices into a unified cloud server format without secondary development.

4) With Modbus gateway function and Modbus TCP to Modbus RTU;

5) ZLAN1043N supports cross-intranet communication in P2P way, and realizes access to Internet via ID anytime and anywhere; it can realize PC monitoring of any intranet device in the environment without cloud.

1.4.3 Some applications require no external MCU.

Because ZLAN1003 chip no need to configure MCU through serial port and SPI, the internal solidified program already can realize UART to TCP/IP, and the ZLAN1043 has powerful "Transcoding" programming function, so users can ignore MCU, this can reduce circuit complexity and save cost.

We list the following situations:

1) Connect the wireless serial port module: ZLAN1003 can be used as the gateway of some wireless module data acquisition to TCP/IP. At one end of the gateway are wireless modules, such as bluetooth, Zigbee, Rola and other UART interface modules, and Ethernet at the other end. At this time, only the UART of ZLAN1003 and the UART of the wireless module can be directly connected, no additional MCU is needed.

Figure 4 Joint Wireless Modules

2) Instrument acquisition: ZLAN1043 supports the powerful "Transcoding" function, which has automatic collection, and can be programmed with the data in the custom format. At this time as long as the transcoding configuration file is

written to ZLAN1043 inside, then ZLAN1043 will send specific instructions, and sent to the cloud according to the fixed format, need to add a MCU for data acquisition and protocol conversion on printed circuit board.

Figure 5 Instrument Collection

3) Serial port to Ethernet port gateway: ZLAN1003 itself is a powerful single chip of serial device server, if you need to implement serial port to Ethernet port gateway function, need to add external MCU, also without software development. In addition, ZLAN1003 has RS485EN pin, which can support all kinds of 485 chips.

1.4.4 Seamless connection with the original serial port procedures.

Figure 6 No Modification for User MCU Serial Port Program

Many RS232/RS485 devices have UART interfaces inside the original MCU, and the internal procedures of MCU have been developed. For the following reasons, users may not wish to redevelop the internal procedures of MCU:

1) The MCU serial port program has been verified for many years and is stable and reliable. If redevelop it to interface such as SPI, there will be stability considerations.

2) Due to short R&D schedule and insufficient R&D resources, directly using the original MUC program will accelerate the progress and reduce the R&D investment.

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

ZLAN1003 provides a good solution for the network upgrade of such RS485/232 devices, so long as joint the part that joint with 232/485 chip before to ZLAN1003. There is no need for software development, just design the circuit to complete the product upgrade.

ZLVircom is equipped with the virtual serial port function. If the user does not want to modify serial port software of the upper computer, it can also meet the requirements.

1.5 Product Selection

Model	Name	Function
ZLAN1003	Serial device server single chip	
ZLAN1043	Modbus gateway single chip	Additional base on ZLAN1003: Modbus RTU to Modbus TCP; Multi-host; Registered heartbeat package; TCP requires passwords; "Transcoding" function; HTTP GET/POST requests; 8 IO level control
ZLAN1043N	P2P single chip	Add P2P function under ZLAN1043
ZLAN1043-SNMP	SNMP single chip	Add SNMP to Modbus RTU function under ZLAN1043
ZLAN1003-W	Web control chip	Output control command function through webpage

1.6 Target Application

ZLAN1003 can be used for many network intelligent devices:

- **IoT gateway:** after IoT wirelessly collecting data, transform data into TCP/IP to

upload;

- **Access control:** all kinds of RFID and access control equipment are converted to TCP/IP for communication;
- **Security:** remote alarm of all kinds of perimeter alarm products, such as electric fence, infrared probe, fire protection and other;
- **Instrument collection:** instrument data collection and upload of electricity meter, energy consumption monitor meter;
- **Modbus gateway:** ZLAN1043 itself is a Modus gateway single chip, which can be used to upgrade Modbus RTU equipment to Modbus TCP equipment;
- **Industrial automation:** RS485/232 signals of various industrial devices are converted into TCP/IP signals for collection and control.
- **Smart home appliances:** smart home gateway.
- **Network IO controller:** ZLAN1043 itself has 8 IO input and output pins, and single chip can realize the design of network type 8 DI/DO products.
- **Industrial automation:** industry 4.0 and automation production.
- **Building intelligence and energy monitoring:** the collection and transmission of power generation such as wind power and solar energy.
- **Medical monitoring equipment:** informationization of medical equipment.

2. Pin Definition

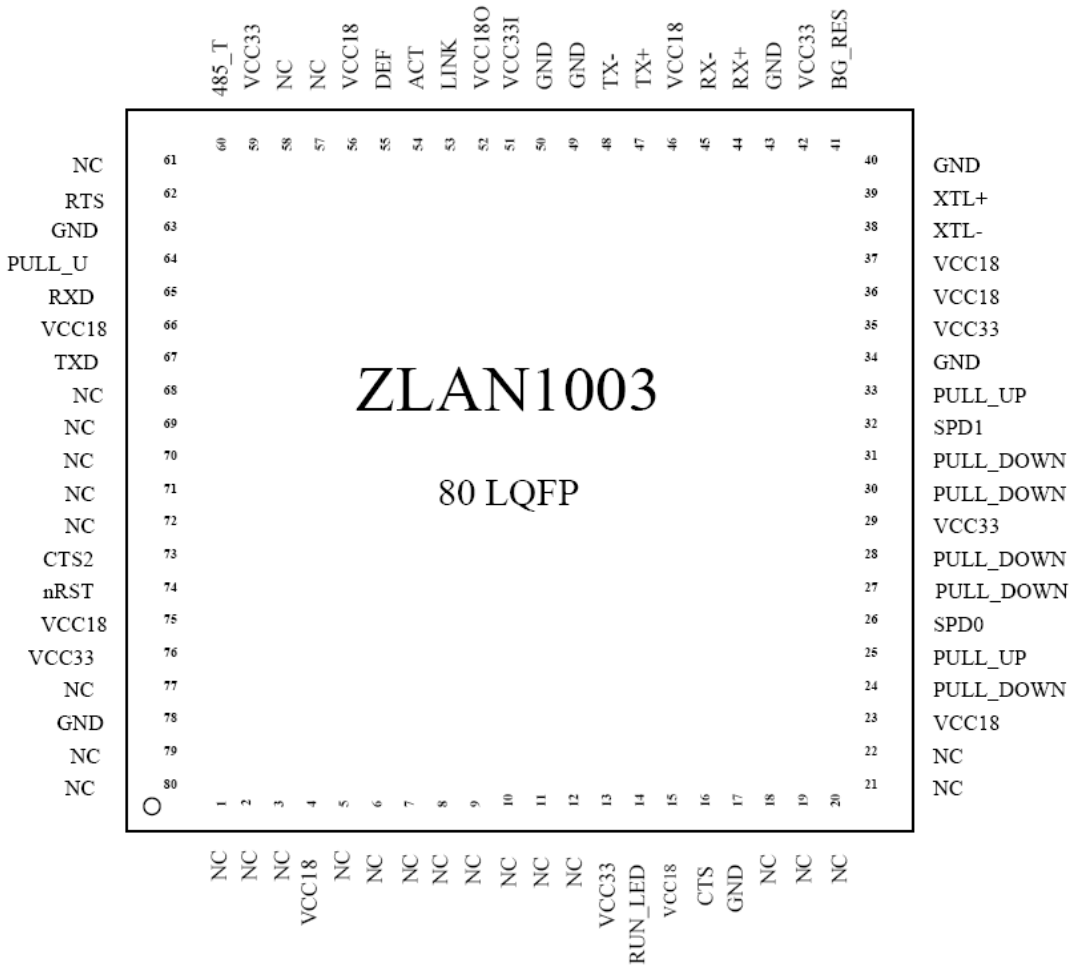


Figure 7 ZLAN Pin

Table 1 Pin Description

Pin Name	Type	Pin No.	Description
Power, Reset, Oscillator			
VCC33	POWER	13,29,35,42,59,76	To connect 3.3V power, main power supply
VCC18	POWER	4,15,23,36,37,46,56,66,75	1.8V input. When in single-power mode, connect VCC18O pin because ZLAN1003 has a built-in 3.3 to 1.8 voltage regulator. VCC18O is 1.8v output, and VCC18 can be supplied by VCC18O; When dual power mode, connect the external 1.8v power supply.
VCC33I	POWER	51	Regulator power input in internal chip. To connect 3.3V power, is the input of regulator output. When no use the internal regulator, use the external 1.8V supply, connect the pin to external VCC18.

VCC18O	OUT	52	1.8V output of interior regulator. No use the internal regulator, use the external 1.8V supply, connect the pin to external VCC18.
GND	GND	17,34,40,43,49,50,63,78	Ground
nRST	IN	74	When nRST is in low electrical level, reset chip. The low level keeping time must over 5ms. Recommended to use the special reset chip, and do not use the resistance capacitance reset circuit.
XTL-, XTL+	IN	38,39	25M oscillator, noted that it's need to connect 1M Ohms resistance.
Ethernet Pin			
RX+,RX-	IN	44,45	Ethernet receiving
TX+,TX-	IN	47,48	Ethernet sending
Ordinary Serial port Pin			
RXD,TXD	IN/OUT	65,67	3.3V TTL level, serial port input/output pin, can directly connect with MCU port. Note that RXD connect user MCU TXD, TXD connect user MCU RXD.
CTS, CTS2	IN	16	Hardware flow control input, when chip configured as CTS/RTS, DSR/DTR mode, only CTS=0 the chip port will output data. If need hardware flow control support please connect CTS and RTS together, or else no need connect these two pins.
RTS	OUT	62	Hardware flow control output. Flow control set as CTS/RTS, DTR/DSR, normally RTS=0, when RTS=1 means ZLAN1003 cannot receive data, user MCU should stop to send data to ZLAN1003. The reason ZLAN1003 cannot receive data including: chip in initialization, TCP connection no built, the receiving buffer of ZLAN1003 port is full.
High-speed Serial port Pin			
CONFIG	IN	33	High speed serial port enable pin, if pull-up through 10K, do not use high-speed serial port; Otherwise, pull-down through 10K to open UART2 high-speed serial port.

RXD2, TXD2	IN/OUT	9, 10	Support serial port of high-speed up to 921.6Kbps, if using high-speed serial port, CONFIG needs to pull down, and need to connect an I2C chip (such as AT24C02).
SCL, SDA	/	71, 72	When high speed serial port is required, the SCL and SDA of the I2C chip (such as AT24C02) are connected with these two pins.
Input & Output			
RUN_LED	OUT	14	Run the indicator light, when MCU works normally, it will output a square wave of 2s period. Can light LED via 10K resistor to indicate the chip in work.
100M	OUT	61	Can call 100M_LINK, means the network line is connected. At 0, indicate that the RJ45 network line of module has been connected.
LINK	OUT	53	TCP connection indicator. When 0, means module has built TCP connection with network server or in UDP Mode, and the cable connection normally, thus the module can send and receive data. If cut cable, the LINK will be 1.
ACT	OUT	54	Data activity indicator. When 0, means ZLAN1003 port has data transceiver. But the pin will be in change between 0 and 1 when there has data, ACT=1, cannot be sign of non-data communication.
DEF	IN	55	Parameter RESET. When be 0 and time keep over 1s, the module will restart with default IP in static mode, Gateway 255.255.255.0, 192.168.1.1. This is avail for user forgot IP, can reset it to default.
485_TEN	OUT	60	485 sending control end, normal 0, when sending data to serial port will be 1. Can directly connect TXD_EN pin of MAX485 chip.
IO_EX	OUT	58	For IO extension.
Other			
RSET_BG	IN	41	Connect AGND through 1% accuracy 12.1K resistor.
PULL_DOWN	IN	24,27,28,30,31	Pull-down to GND via 10K. Special note: cannot combine these pin together to connect resistance for grounding, it must connect the resistance separately.
PULL_UP	IN	25,33,64	Pull-up to VCC33 via 10K

SPD0 、SPD1	IN	26,32	Chip running speed configuration. When pull-up to VCC33 via 10K, configure to high-speed status. When pull-down to GND via 10K, configure to low-speed status. Current in low-speed is 140mA, high-speed is 210mA. The low-speed status support highest baud rate 57600bups, high-speed 115200bps.
NC	/	Other	No specified pin please hang in the air.

The following pins can be converted into IO input/output by the "Transcoding" configuration file of ZLAN1043 chip, and the function used as indicator light and key input will be disabled. The following pins can be used as input and output.

The output is to control the level of any of the pins through instruction defined by any value; As input, can regularly query or initiate the state change of the IO port.

Table 2 Definable IO Pins

IO Table			
Name	IO No	Pin No	
DEF	PIN[0]	55	
485_TEN	PIN[2]	60	
100M	PIN[3]	61	
RTS	PIN[4]	62	
LINK	PIN[5]	53	
ACT	PIN[6]	54	
RUN_LED	PIN[7]	14	
IO_EX	PIN[8]	58	

Note: PIN[1] is reserved, directly go to PIN[2].

3. Hardware Design Direction

Here are two design examples to illustrate the design of the schematic diagram. The design includes the following points:

1) RJ45 integrate the network transformer and no have network transformer: the former is more integrated, but with higher cost. We introduce ZLSN3003S and SNMP card

respectively.

2) 5V voltage input and wide voltage input: the power circuit is different. We introduce ZLSN3003S and SNMP card respectively.

3.1 RJ45 with Network Transformer

3.1.1 ZLSN3003S Figure

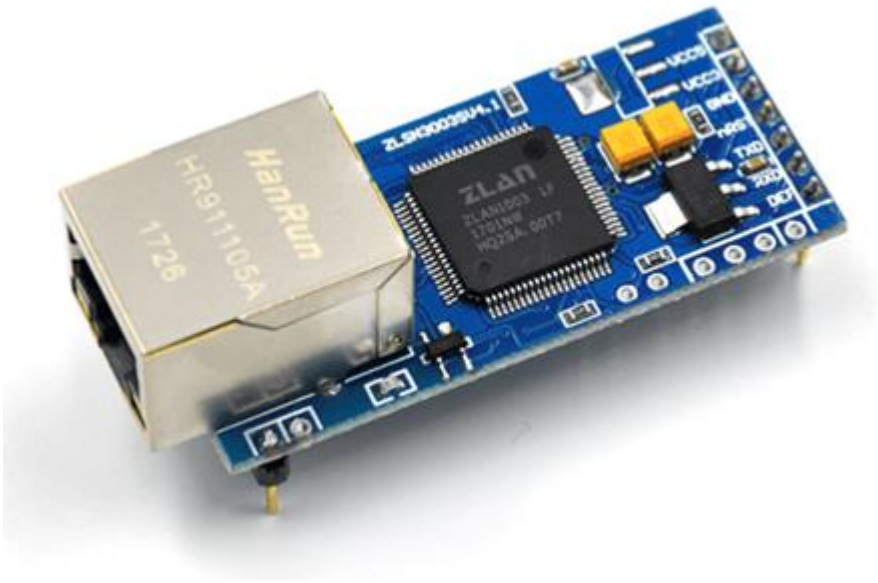


Figure 8 ZLSN3003S

ZLSN3003S is TTL level serial port to Ethernet port product. It adopts the scheme of RJ45 integrated network transformer. The power input is 5V or 3.3v.

3.1.2 ZLSN3003S Schematic Diagram

The schematic diagram of ZLAN3003S is shown in figure 9 and figure 10.

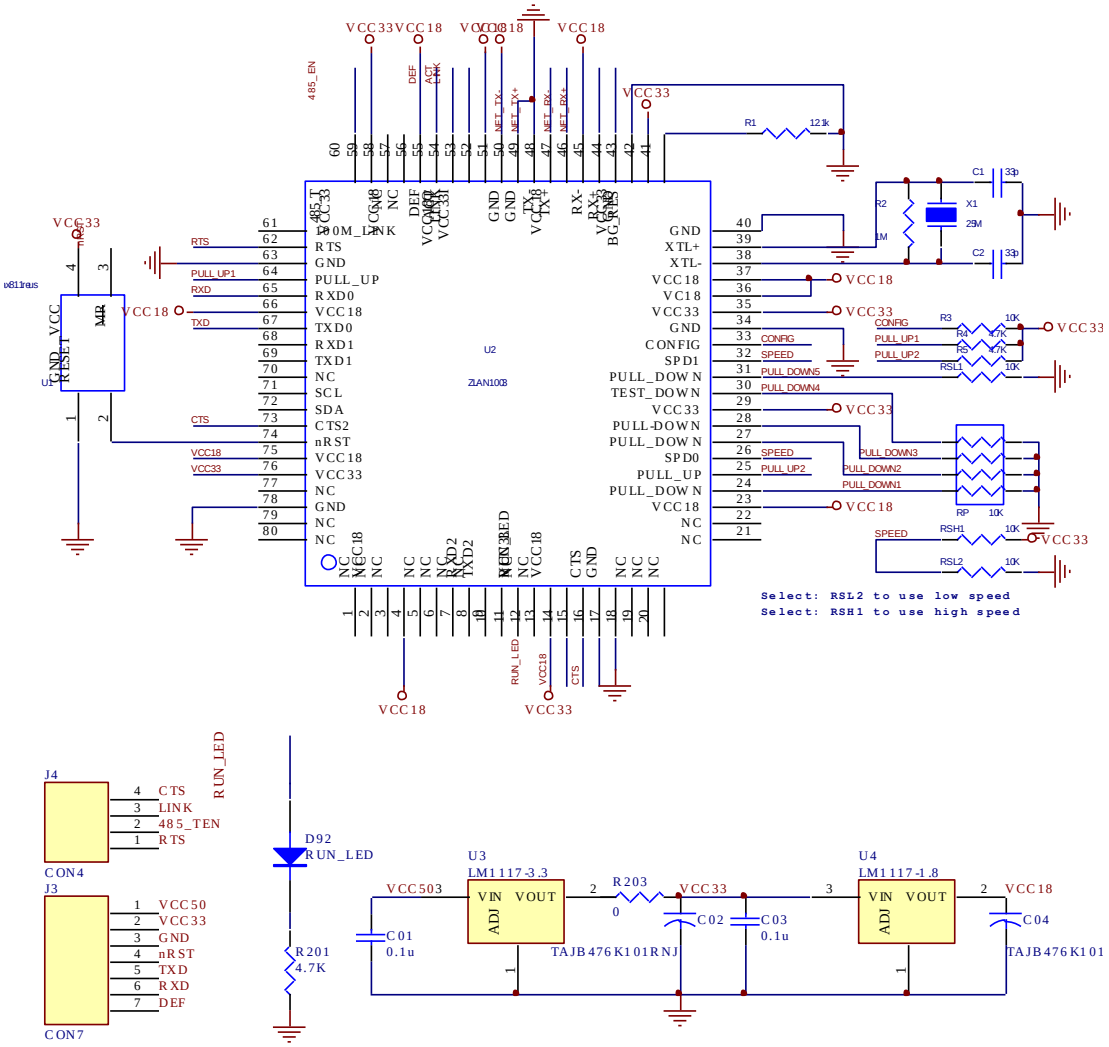


Figure 9 Core part of ZLAN3003S Schematic Diagram

1) Pins extract for 3003S: CTS, LINK, 485_TEN, RTS, nRST, TXD, RXD and DEF. All these pins are directly extracted from the 1003 chip.

2) The power input can be 5V or 3.3V. From the power supply circuit, 5V input was reduced to 3.3V through 1117-3.3 chip, and 3.3v to 1.8v through 1117-1.8. It should note that if the voltage is 9V, 12V, 24V and so on, please use the power plan of SNMP card instead of the LDO chip of 1117. Because of the dual power supply in this design, 1.8V is generated externally, so the VCC33I and VCC18O of the chip both connect with 1.8V power supply.

3) Since the high speed UART is not used here, pull up the CONFIG pin, and let RXD2 and TXD2 hang on.

4) It is important to note that RSH1 and RSL2 can only be welded one when

selecting the speed mode of ZLAN1003. Welding RSH1 is high speed, welding RSL2 is low speed.

5) RUN_LED drives an LED indicating that ZLAN1003 works normally.

6) CTS and CTS2 are connected together and can be used as hardware flow control.

7) The crystal oscillator can use SMT or the straight S49 type, the capacitance is 33p and requires cross-connecting a resistance of 1M ohms.

8) R1 is the 12.1K resistance of 1% precision.

9) The reset chip uses max811reus, where MR pins are for manually reset.

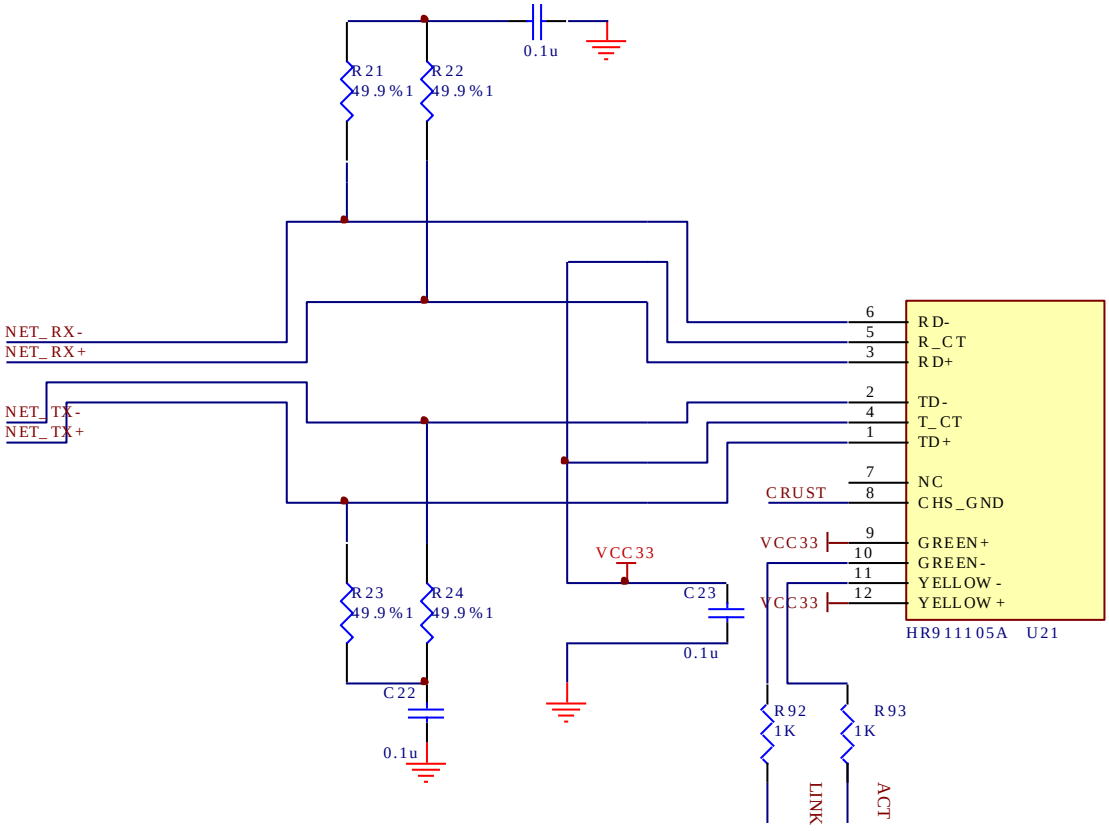


Figure 10 ZLAN1003 Schematic Diagram Ethernet interface Part (RJ45 integrated network transformer)

The design of the Ethernet port part of ZLAN1003 is shown as in figure 10, the LINK and ACT are connected to the lamp on RJ45, to indicate TCP connection and data activity. There are four 49.9 resistors with a precision of %1.

3.2 RJ45 without Network Transformer

3.2.1 SNMP Card Figure

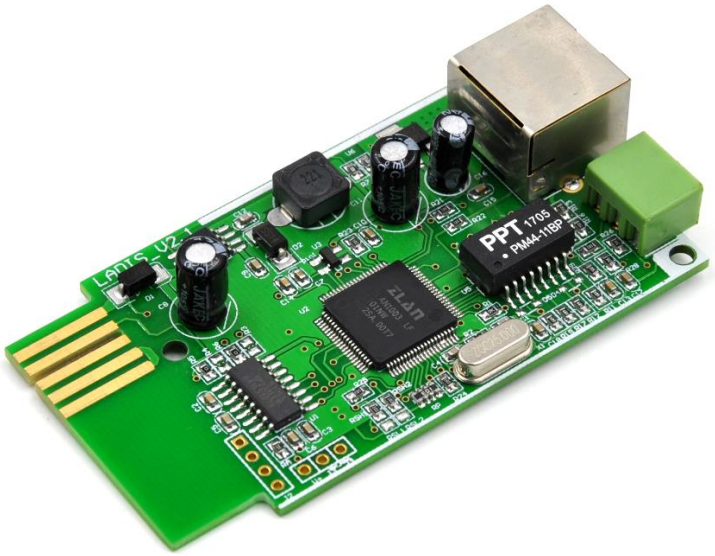


Figure 11 SNMP Card

The SNMP card is the product of powered by connecting with RS232 and 12V power supply of the cast through golden finger, reading the data on RS232 and converting data into SNMP protocol to upload the network. The board card is a two-layer circuit board for RS232 to Ethernet port designed with ZLAN1003, adopts wide voltage input design and adopts the structure of network transformer and RJ45 apart.

3.2.2 SNMP Card Schematic Diagram

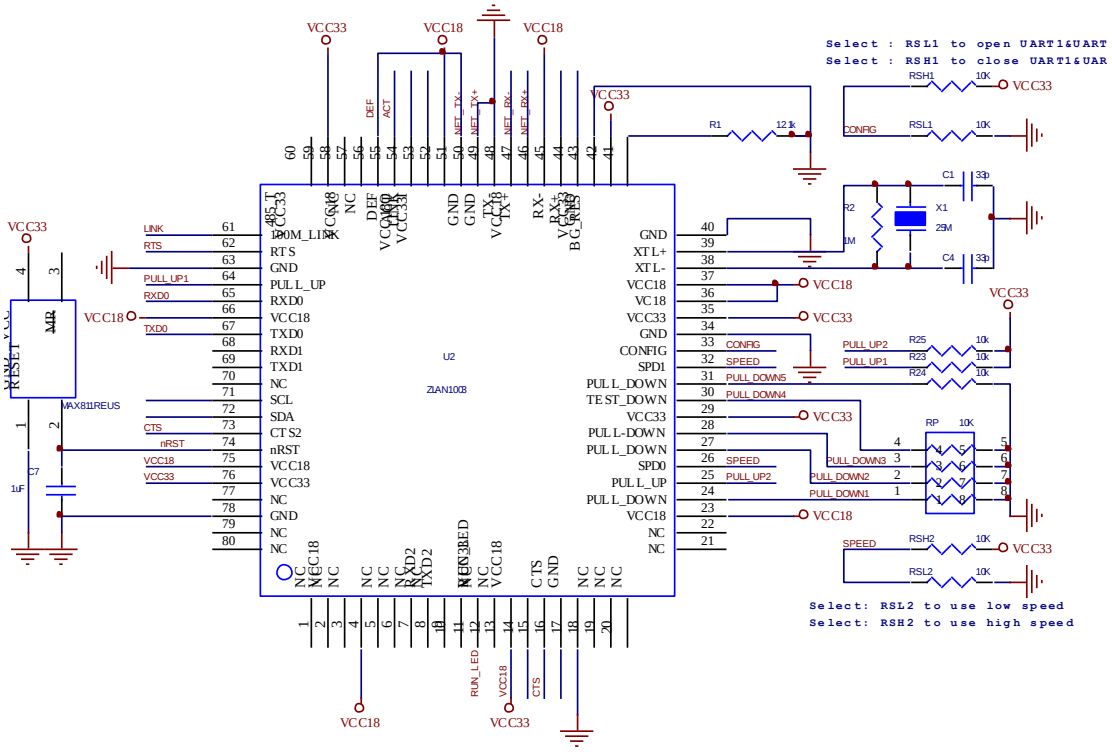


Figure 12 Main Chip Part of SNMP Card ZLAN1003 Schematic Diagram

Here we only introduce the difference with ZLSN3003S, the same part can see the above introduction. Here pin30 is TEST_DOWN, actually it is the same as PULL_DOWN with just pulling down the resistance.

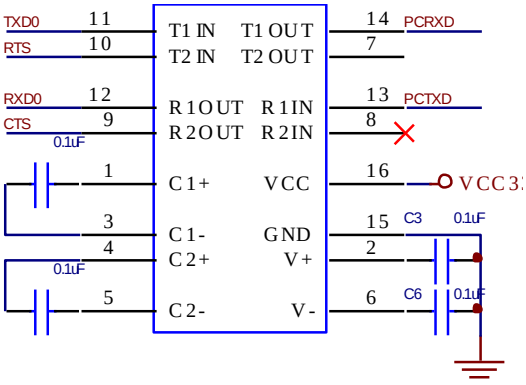


Figure 13 SNMP Card UART to RS232 Circuit

Here mainly use MAX232 chip to convert RSD0 and TXD0 serial port of ZLAN1003 to RS232 level.

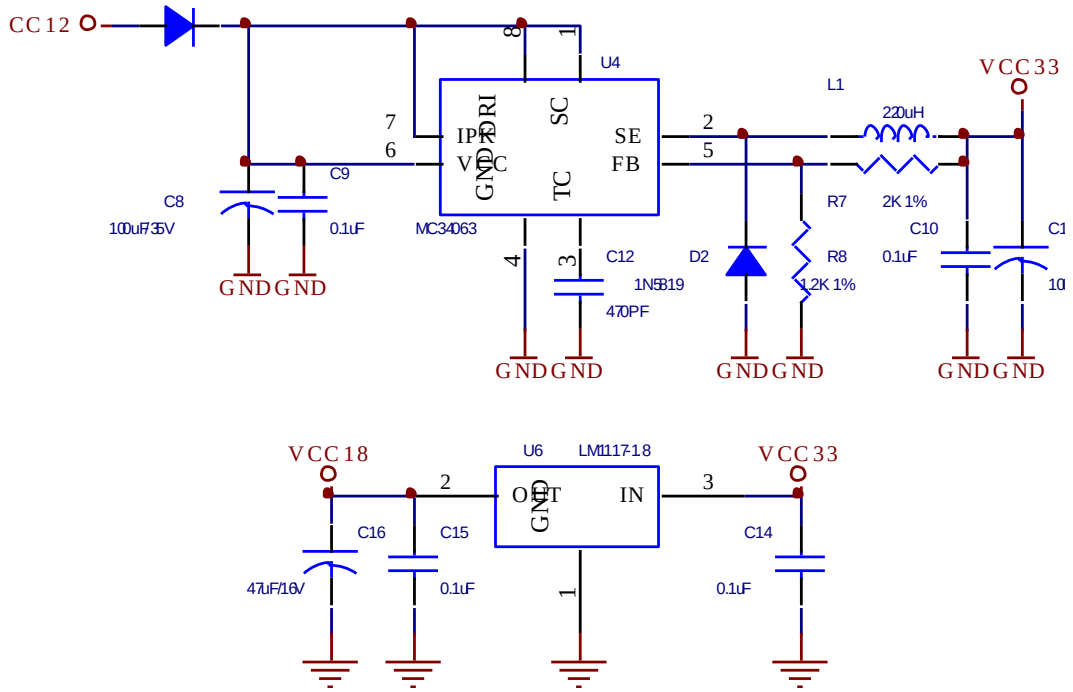


Figure 15 Part Circuit of SNMP Card Power Supply

When using ZLAN1003, if it is more than 5V voltage input, it is recommended to use the DC-DC circuit instead of the LDO chip with low energy conversion efficiency. Here take the example of MC34063 to introduce. The input voltage can be 9V~24V. MC34063 converts the input voltage to 3.3V and then convert to 1.8V through lm1117-1.8 to supply power to the chip.

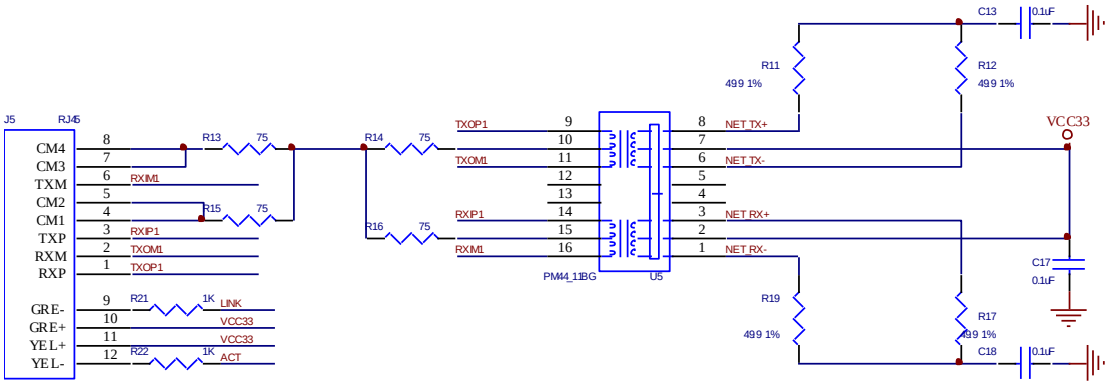


Figure 16 SNMP Card Network Part

The network part is composed of a network transformer PM44_11BG and the RJ45 with the Ethernet port light, and four resistance of 1% precision.

3.3 High speed and low speed selection

ZLAN1003 has two working speed: high speed and low speed.

1) High speed: pull both SPD0 and SPD1 to high level. In figure 9, the RSH resistance is welded and the RSL resistance is not welded. At this time ZLAN1003 highest baud rate reached 115200bps. The total required current of VCC18V is 60mA.

2) Low speed: both SPD0 and SPD1 are pulled to low level. In figure 9, the RSL resistance is welded and the RSH resistance is not welded. At this time ZLAN1003 highest baud rate reaches 57600bps. The total required current of VCC18V is 100mA.

3.4 Use the internal 1.8V stabilized voltage supply

ZLAN1003 VCC18 power supply has two power supply modes: internal voltage regulator generating and external power supply.

1) Internal voltage regulator generating: ZLAN1003 has a regulating circuit inside, in VCC33VI pins enter VCC3.3 V, it produces a 1.8v power output in VCCI8O. The VCC18O connecting with other VCC18 pins supply power to the chip. In figure 4, the R18INNER resistance and not welding the R18OUTER, it uses internal regulator generating type. In this way ZLAN1003 will generate additional heat due to stable pressure. It is not recommended unless it is particularly necessary to reduce costs and reduce volume.

2) External power supply: by default use this method. Users can convert VCC3.3V to 1.8V through an lm1117-1.8 regulating chip, and then supply to ZLAN1003. At this time, VCC33VI and VCC180 both connect the 1.8V power supply from external LM1117-1.8.

3.5 Reduce power consumption design

According to the above analysis, the following methods can be used to reduce the heat of ZLAN1003:

1) External power supply: using the external lm1117-1.8 regulating chip to provide 1.8V power supply, and the internal regulator is disabled. Because the heat dissipation of

lm1117-1.8 is better than ZLAN1003, the overall heat dissipation effect is better than that of the internal voltage regulator. But this approach will increase the cost of designing a 1.8V regulator. This method is recommended.

2) Slow operation: lower the speed of the chip by SPD0 and SPD1, and reduce the total required current of VCC18. The effect of this method is not as effective as the first method, because ZLAN1003 still use the regulating circuit of heating, but the heat will decrease obviously. This approach eliminates the need for additional design costs. However, it is important to note that if the user use the baud rate of 115200bps, the method cannot be used because the low-speed operation does not support 115200bps.

3.6 PCB wiring

PCB wiring don't have high requirement. Please note the following points:

1) The crystal should be as close to the pin as possible. Do not let the network line RX-, RX+, TX-, TX+ pass through the near of the crystal oscillator.

2) RX-, RX+ is a pair of difference lines, TX-, TX+ is a pair of difference lines, and a pair of difference lines should be taken together to avoid the separated route.

3) When do PCB design for ZLAN1003 chip, if the condition allows it can add 5 large guide holes at the bottom, this can increase the heat dissipation. Refer to PCB of SNMP card.

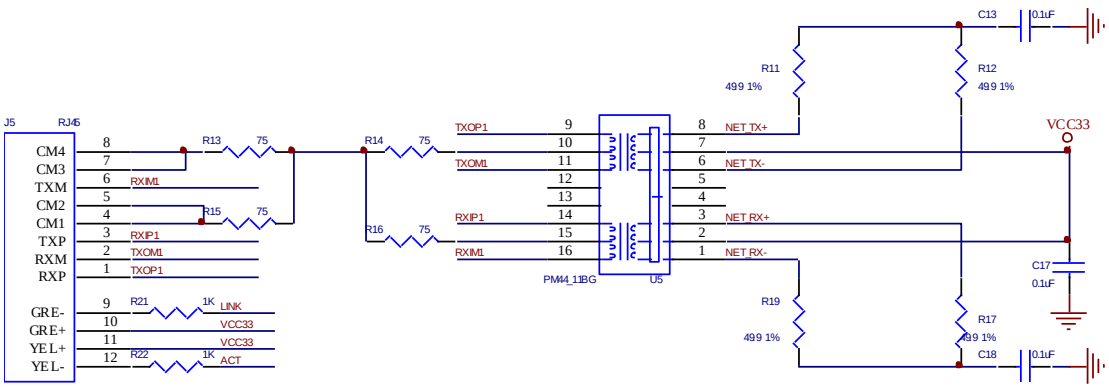


Figure 17 ZLAN1003 Heat Dissipation Design

4. Function Instruction

4.1 Install Software

ZLVircom can be used to configure the parameters such as chip IP and other, as well create virtual serial port.

Install just follow the default prompt. After the installation, ZLVircom will be started every time when the computer start to create virtual serial ports.

4.2 Parameter Configuration

After ZLVircom was installed, and the circuit board of the chip power on, as well connect the RJ45 of the circuit board to the network of the computer. Run the ZLVircom software as figure 18, and click “Device Manage” as shown in figure 19. Using ZLVircom can search and configure the device parameter in different network segments, it’s very convenient that only need the device in the same switch of the computer running ZLVircom software.

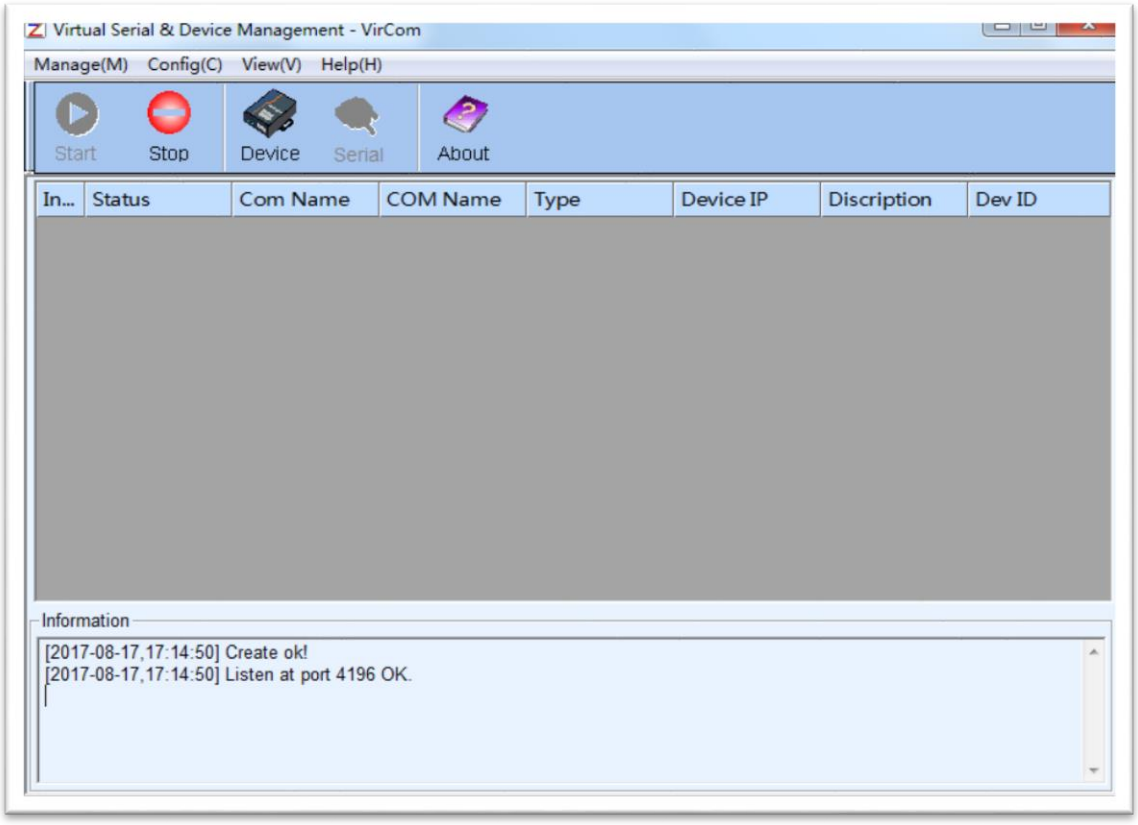


Figure 18 ZLVircom Main Interface

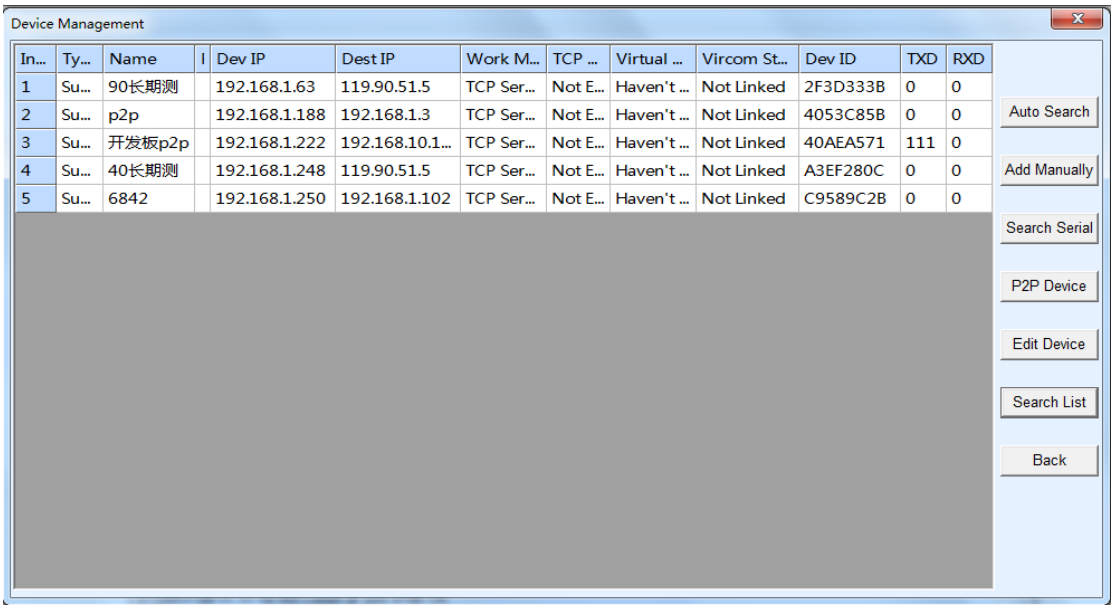


Figure 19 Device List

From the device list, you can see all of the current online devices. Click “Edit Device” to configure the parameters.

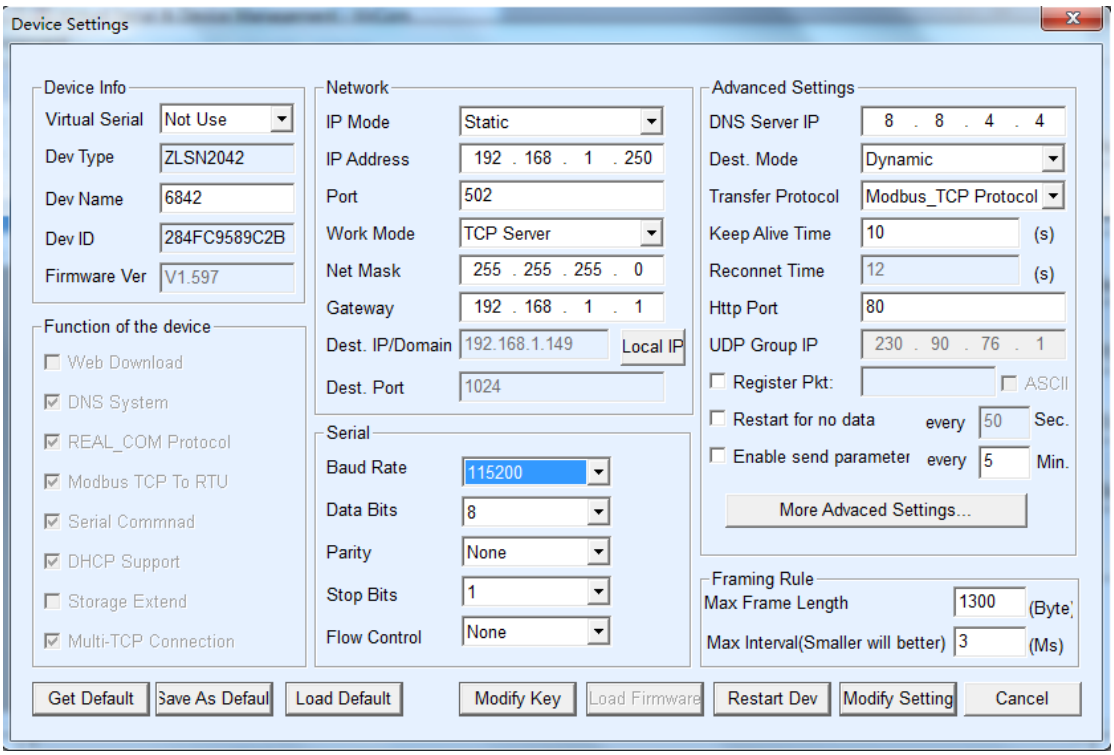


Figure 20 Device Edit Interface

In this interface, the user can set the parameters of the device, then click "Modify Setting", and the parameters are set to the flash of the device, with power-off no lost. The device will restart automatically.

The parameters setting in here are: baud rate, data bit, parity bit of serial port setting; IP

address, subnet mask and gateway of network setting; sometimes depending on computer software, and there need to configure the work mode of the serial device server.

Details meaning of other parameters are as below:

Table 4 Parameter Meaning

Parameter Name	**Value Range**	**Meaning**
Virtual Serial	Non-in use, established virtual serial	You can bind the current device to a created virtual serial port.
Dev Type		Show only the model of the core module
Dev Name	Any	You can give the device a readable name, with a maximum of 9 bytes, and support the Chinese name.
Dev ID		The factory's sole ID, cannot be modified.
Firmware Ver		The firmware version of core module
Function of the Device		Please refer to the part of “5.2 Model and Function”
IP Mode	Static, DHCP	The user can choose Static or DHCP (Dynamic acquisition of IP)
IP Address		The IP Address of serial device server
Port	0～65535	The monitoring port of Serial device server when in the TCP Server or UDP mode. As a client, it is best to specify that the port is port 0, which is good for increasing the connection speed, and the system will randomly assign a local port when using the 0 port. At this time the difference from specifying the non-zero port are: (1) local port is 0, module sets up a new TCP connection with PC when restarting, old TCP connection may not be closed, so that the old TCP connection of the host has been unable to close, specify the non-zero port does not have the problem. Generally host wants to close the old connection when the module is restarted. (2) the local port is 0, the time of TCP rebuilding connection is faster.
Work Mode	TCP Server(TCP Server Mode),TCP Client(TCP Client Mode),UDP Mode, UDP Multicast	When set to TCP Server, the network Server needs to actively connect the serial device server; When set to TCP Client, the serial device server initiates the connection to the network server specified by the destination IP.
Net Mask	Eg: 255.255.255.0	Must be same as net mask of local LAN.
Gateway	Eg: 192.168.1.1	Must be the same as the local LAN gateway. If it is not crossing outer network (such as the cable connecting

		computer), it is best to set the gateway as the IP address of the connected computer.
Dest. IP/Domain		In the TCP Client or UDP mode, the data will be sent to the destination IP or the computer of domain name instruction.
Dest. Port		In the TCP Client or UDP mode, the data is sent to the destination port of the destination IP.
Baud Rate	1200,2400,4800,7200,9600,14400,19200,28800,38400,57600,76800,115200,230400,460800	Serial baud rate
Data Bits	5, 6, 7, 8, 9	
Parity	None, Even, Odd, Mark, Space	
Stop Bits	1,2	
Flow Control	None (no flow control), CTS/RTS, DTR/DCR, XON/XOFF	RS232 port valid
DNS Server IP		When the destination computer is described by a domain name, DNS server is required to resolve the domain name, which specifies the IP of this DNS server. When the IP mode is DHCP, the parameter is not specified and will be automatically acquired.
Dest. Mode	Static, Dynamic	UDP working mode: if the destination computer is described by a domain name, it's best to choose the static mode; If there are multiple computers in the LAN communicating with serial device server through UDP, it is best to choose dynamic mode. TCP server mode: this parameter must be dynamic. TCP client mode: when IP mode is dynamic, the destination IP is reconnected after the device is restarted, so that the correct IP address can be obtained again. Otherwise, it will do direct connection without automatically restarting the device.
Transfer Protocol	NONE, Modbus TCP<->RTU, Real_COM	NONE indicates that the data forwarding from the serial port to the network is transparent; Modbus TCP<->RTU will convert Modbus TCP protocol directly into RTU protocol to facilitate coordination with Modbus TCP protocol; RealCOM is designed to be compatible with the old version of REAL_COM.
Keep Active Time	0～255	(1) Choose 1~255, if the device is in the TCP client working mode, the TCP heartbeat will be sent automatically for every "keep alive time". This can guarantee the TCP availability of links. When set to 0,

		there will be no TCP heartbeat. (2) Set to 0~254, when transformation protocol choose REAL_COM protocol, the device will send a length of 0 to 1 content data for every " keep alive time " to implement the heartbeat mechanism of Realcom. When set to 255, there will be no Realcom heartbeat. (3) Set to 0~254, if the device is working on the TCP client, the device will send the parameters to the destination computer every " keep alive time ". When set to 255, no have the parameter sending function. This mechanism is not normally used, users are not required to pay attention.
Reconnect Time	0～255	Once the serial device server in a TCP client mode disconnect with the server (as long as in the non-connection status), it will initiates a TCP connection to the Server every while, can be 0~254 seconds, if set 255, never for reconnection. Note first TCP connection would immediately (such as hardware on electricity, through zlvircom software restart equipment, no data), only after the first connection failure will try again after waiting for the "break time", so "break time" will not affect the network and server connection setup time under normal circumstances.
Http Port	1～65535	
UDP Group IP		UDP multicast
Enable register package		When the TCP connection is established, the register package is sent to the computer. After you enable the register, you must select the realcom protocol. Support for TCP server and TCP client mode.
Max Frame Length	1～1400	One of the rules of serial. The connected product serial port sends the received data to the network as a frame after receiving the length data.
Max Interval (Smaller will better)	0～255	One of the rules of serial. When there is a pause in the data received by the connected product, and the pause time is greater than that time, the received data is sent to the network as a frame.

The functions supported by the device are described as below:

Table 5 Functions Supported by Device

Name	Instruction
Web page download	Support to control the serial port output command via the web page, only the products with suffix W have this function.
DNS	The destination IP can be domain name (such as the server address starting with www)
REAL_COM protocol	A non-transparent transmission protocol of serial device server, suitable for multi-port serial device server to bind virtual serial port through Internet. Since the protocol contains MAC address it helps the upper computer to identify the device. Normally can no use.
Modbus TCP to RTU	Only the products with the third Number 4 of the model support the function. Can realize Modbus TCP to RTU. Also support multi-host function.
Serial port modify parameter	Support serial port AT-like commands to configure and read device parameters.
Automatic get IP	Support DHCP client protocol.
Storage extension EX function	Subsequent extension
Multi-TCP connection	Support more than one TCP connection when as a TCP server.
IO port control	The model of third number 4 support arbitrary custom commands to control 8 IO outputs.
UDP multicast	UDP multicast
Multiple destination IP	Support for simultaneous connection of 7 destination IP when as a TCP client.
Proxy server	Support for proxy server capabilities (require specific models).
SNMP function	Support SNMP to Modbus RTU protocol. This feature is supported only by the model with suffix -snmp.
P2P function	Support the ability to access the devices in any network through P2P (peer-to-peer) technology. This feature is supported by model of suffix N.

4.3 TCP Communication Test

After configuring the device parameters, can use serial port tool and TCP debugging tool to do communication test for TCP connections.

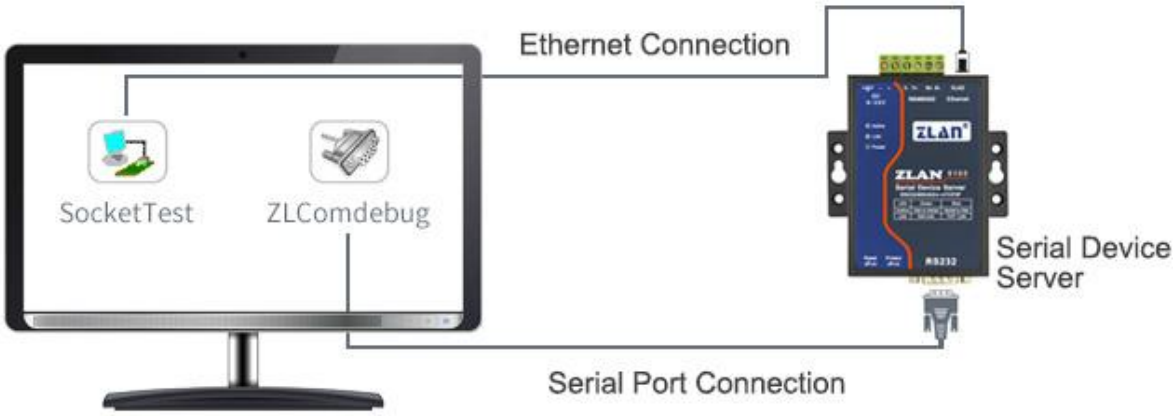


Figure 21 TCP Communication Diagram

Now suppose the PC COM port (USB to RS232 line) connect with serial port of the serial device server, then open the serial debugging assistant ZLComDebug, and open the corresponding COM as shown in figure 22. Open TCP&UDP debugging assistant SocketTest, and as TCP client, fill in the serial device server IP with destination IP (currently 192.168.1.200), destination port is 4196, and then click "open" button as figure 23. In SocketTest, enter "socket send" and click send, then the data is transferred to the RS232 interface via the Ethernet port of the serial device server, next sent to ZLComDebug, and will be displayed in ZLComDebug. Also, enter "Comdebug send" in ZLComDebug, and click send can also be sent to the socket test and displayed.

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

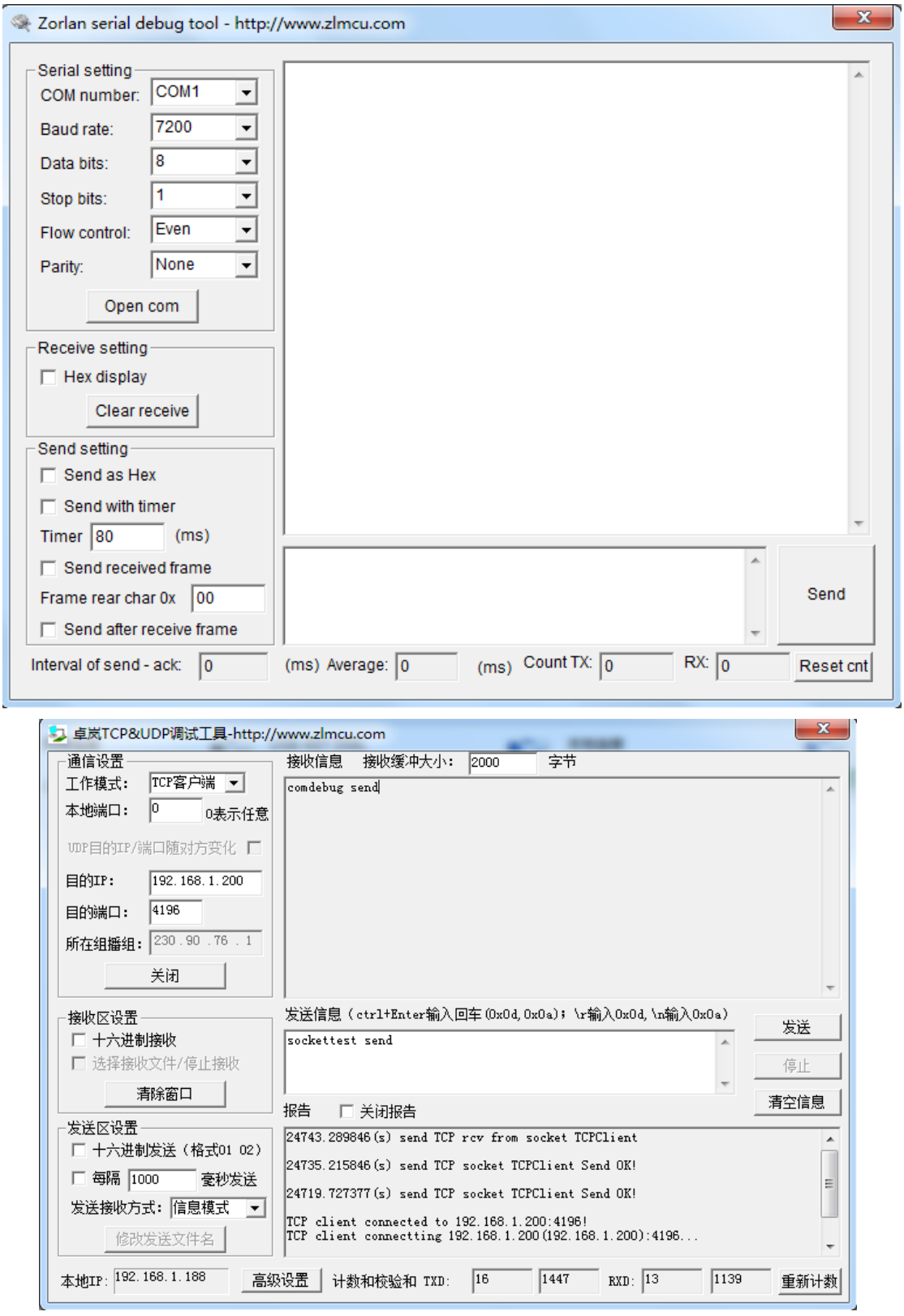


Figure 22 ComDebug Send-receive Interface

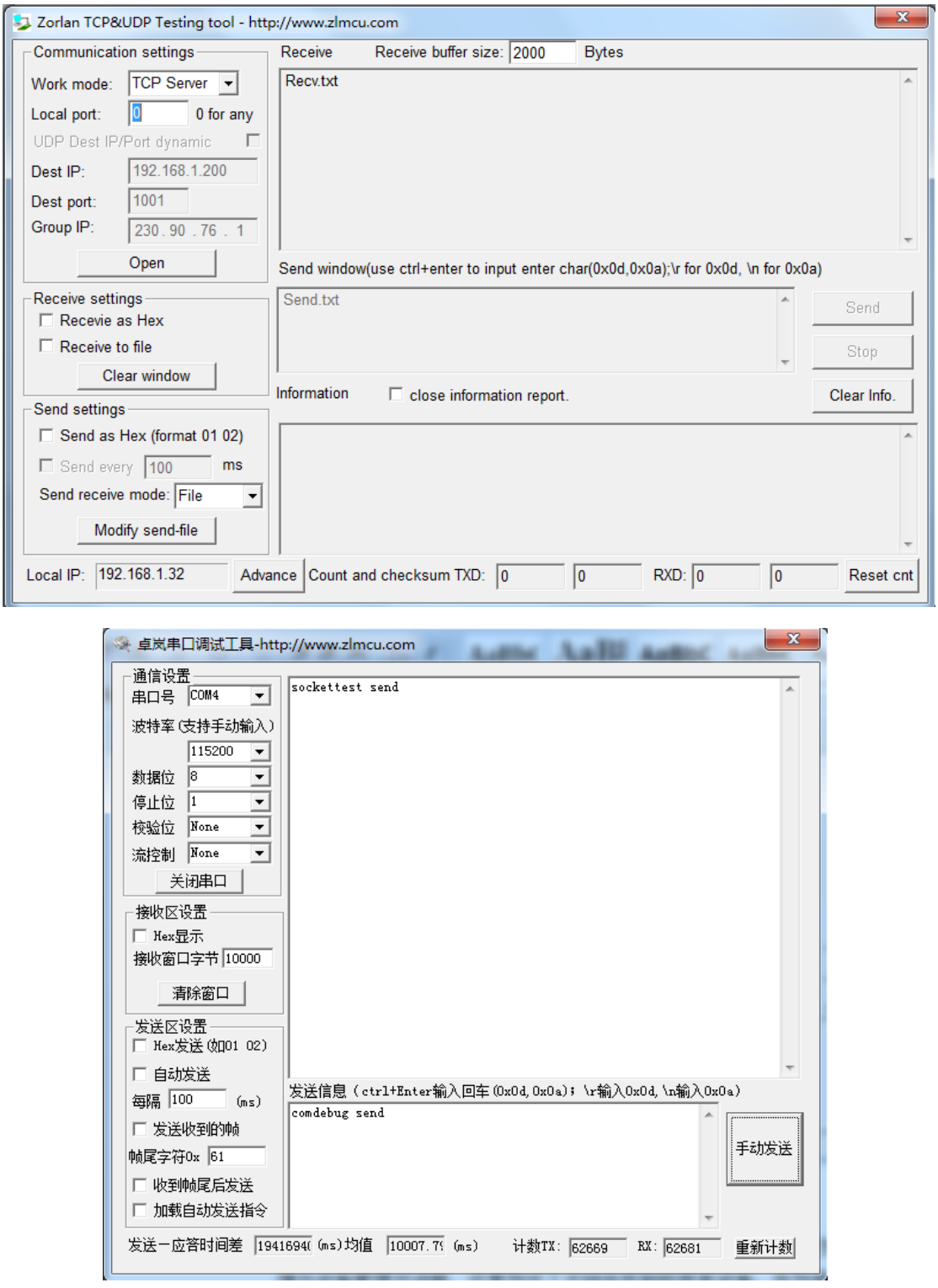


Figure 23 SocketTest Send-receive Interface

4.4 Virtual Serial Port Test

In FIG. 22 SocketTest is through TCP&UDP to communicate with device, in order to let the user's developed serial port software can be used but no need to be modified for TCP communications, need to add a virtual serial port between the user program and serial

device server. As shown in figure 24, ZLVircom and the user program run on a computer, ZLVircom virtually create a COM port, and the COM port corresponds to this serial device server. When the user program opens the COM to communicate it can through ZLVircom → serial device server → send to user serial device. Here shows the operating steps as below:

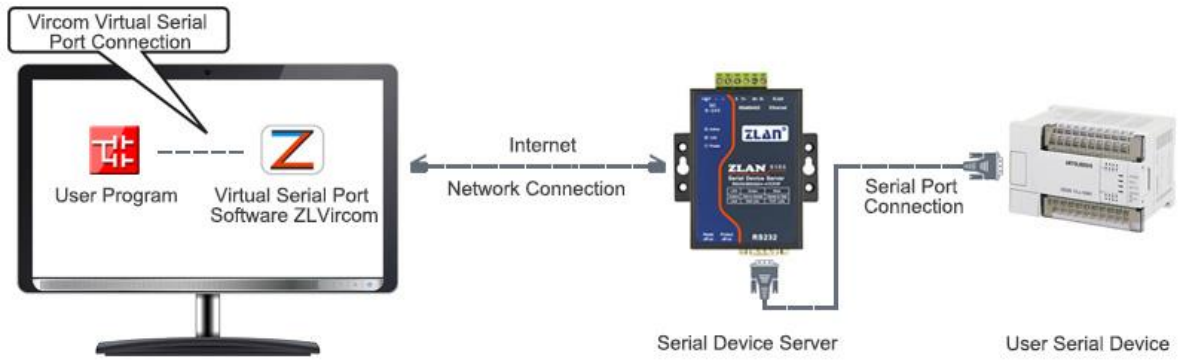


Figure 24 Virtual Serial Port Use

Click on the "Serial Manage" of the main interface of ZLVircom, then click "Add", and select COM5, where COM5 is the COM port that didn't exist on the computer.

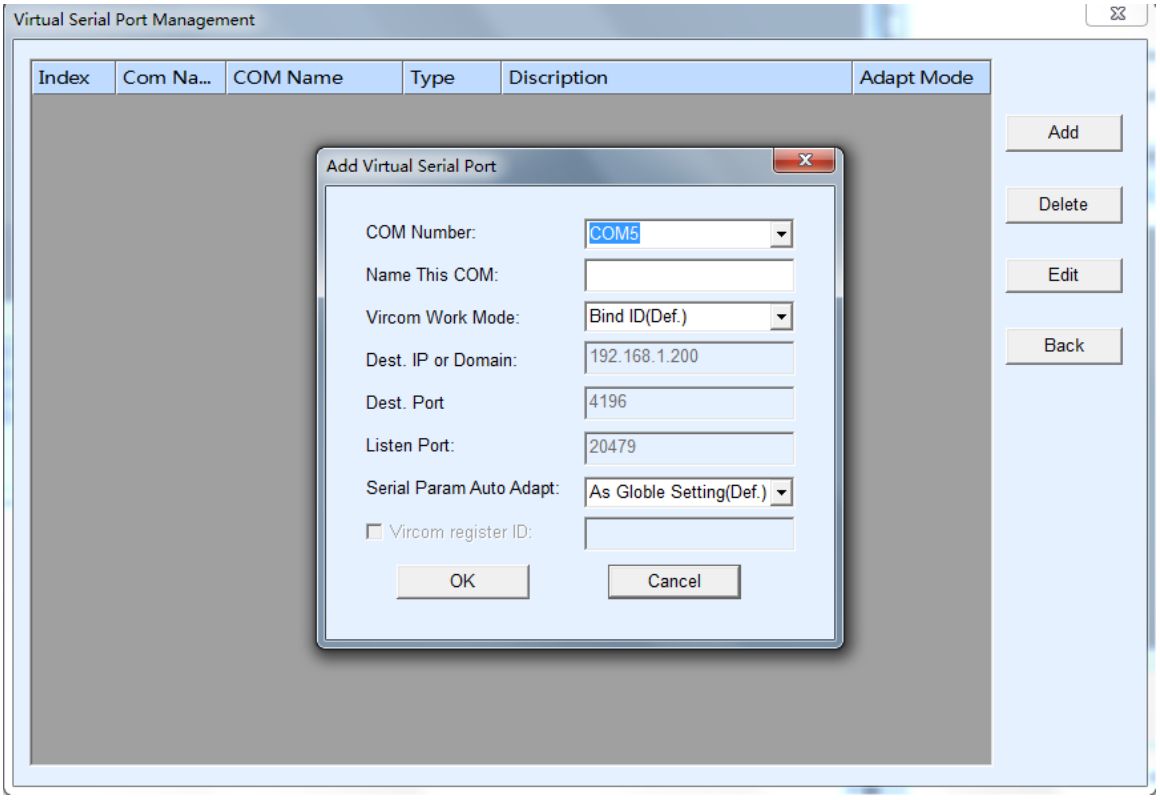


Figure 25 Add Virtual Serial Port

Then enter the "device manage", and double-click the device that you need to bind to the

COM5. As shown in FIG. 20, select COM5 from the "virtual serial port" list in the upper left corner. Then click "modify Settings". And return to the main interface of ZLVircom. You can see that the COM5 has been connected to a device with IP 192.168.1.200. You can use COM5 instead of SocketTest to communicate.

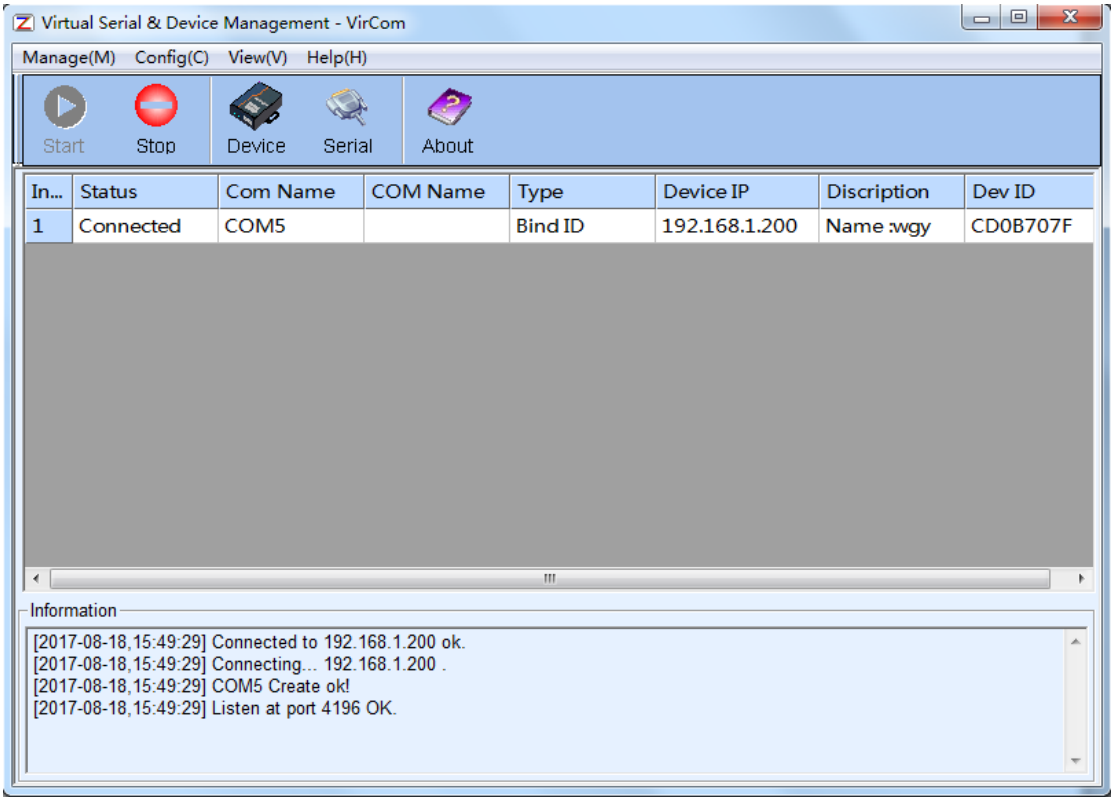


Figure 26 Virtual Serial Port has been connected

Open ZLComdebug to simulate the user's serial port program, open COM5 (the virtual serial port above), and open another ZLComdebug to simulate a serial port device and open COM4 (hardware serial port). The COM5 sending data link is as follows: COM5 → ZLVircom → serial port of serial device server → COM4. Conversely, COM4 to COM5 can also transmit data: COM4 → Ethernet port of serial device server → ZLVircom → COM5. As shown in figure 27, both parties send and receive data.

If COM4 is changed to a user serial device, COM5 can communicate with the user's device.

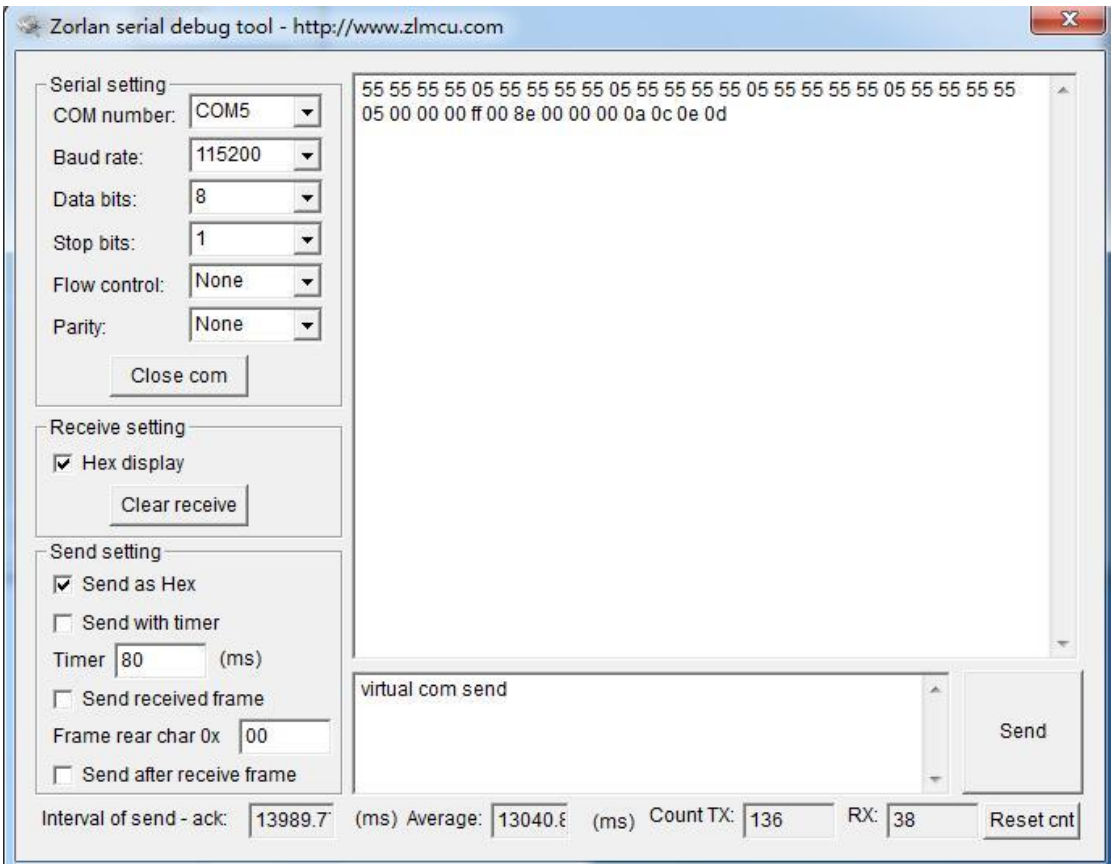


Figure 27 Communication via Virtual Serial Port

4.5 Modbus TCP Test

By default, serial and Ethernet port data are transparently transmitted. If you want to realize Modbus TCP to Modbus RUT, you should choose the converting protocol to “Modbus TCP ←→ RTU” in the device manage dialog box as shown in figure 28. At this point, the device port is automatically changed to 502, the user's Modbus TCP tool connects to the port 502 port of the serial device server, and the sending Modbus TCP command is converted to the RTU instruction output from the serial port. For example, the Ethernet port of the serial device server receives Modbus TCP command 00 00 00 00 00 06 01 03 00 00 0a, then the serial port outputs the command 01 03 00 00 00 0a c5 cd. Note: the serial port may send a number of commands 01 03 00 00 00 0a c5 cd, as the default Modbus adopts storage mode, and will automatically roll polling the query instruction. It will show you how to switch to a non-storage mode later.

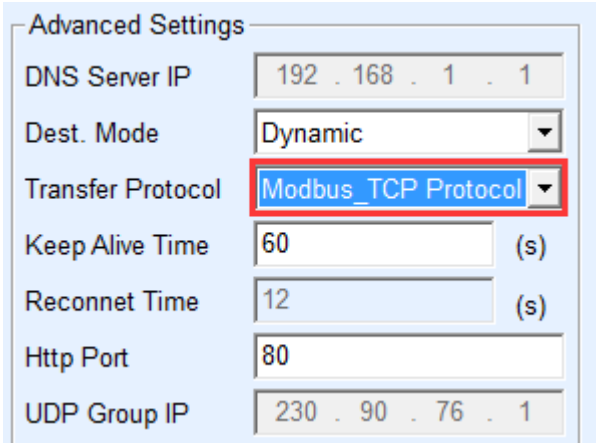


Figure 28 Enable Modbus TCP Function

If the user Modbus TCP software is as Slave, it needs to change work mode to client on the basisi of choosing Transfer Protocol, the destination IP is changed to IP of the computer where the Modbus TCP software in, and destination port is 502, as shown in figure 29.

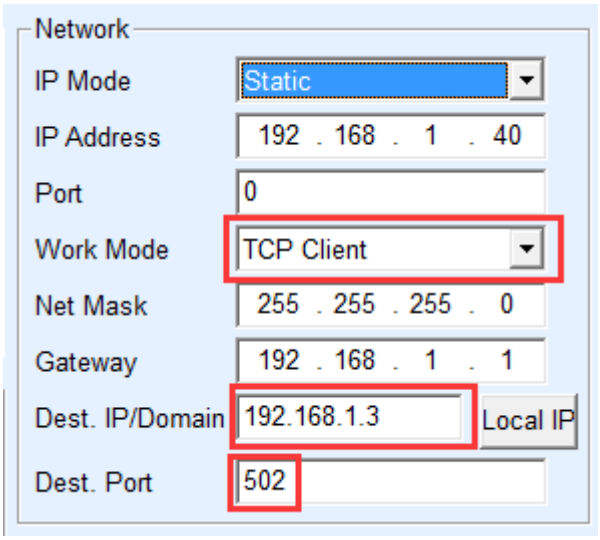


Figure 29 Modbus TCP as Client

4.6 Web Configuration

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

1) Enter the IP address of the serial device server in the browser, such as

http://192.168.1.200, and open the following page.

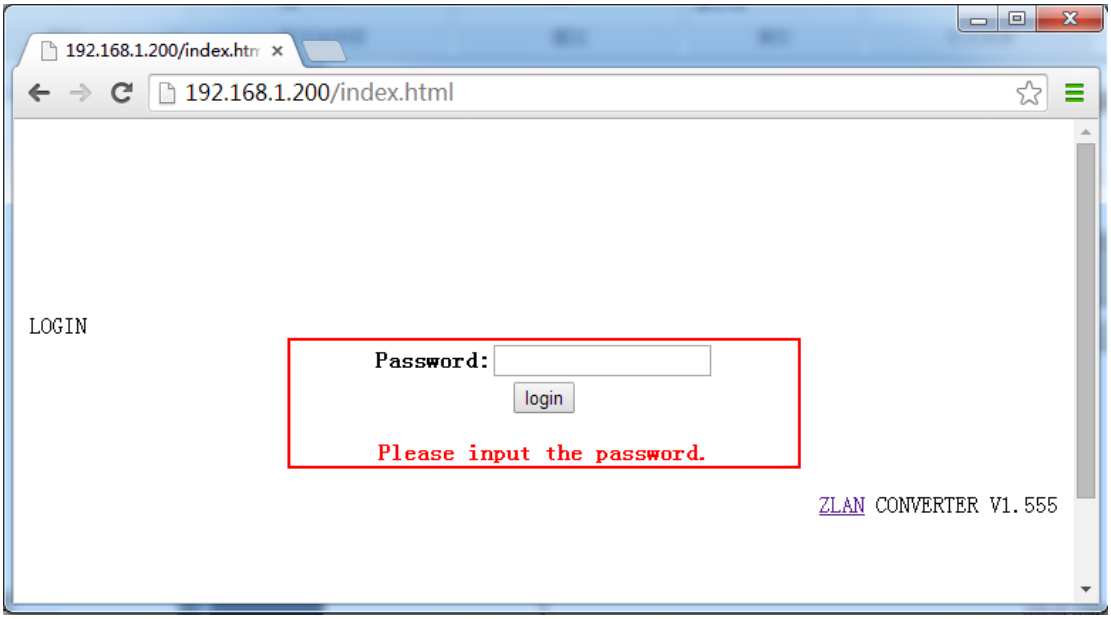


Figure 30

2) Enter Password in “Password”: default is 123456. Click the “login” button to log in.

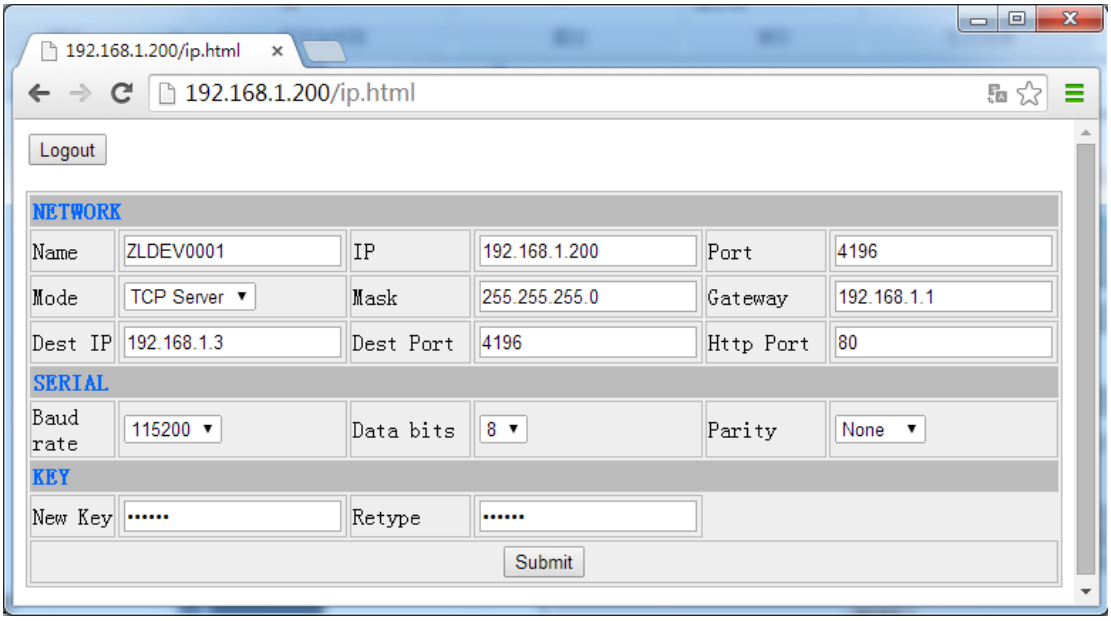


Figure 31

3) In the appearance of the Web page, you can modify the parameters of the serial device server, relative parameters can refer to parameter meaning in Table 4.

4) Click "submit" button after modifying parameters.

5. Work Mode and Transfer Protocol

Different application can choose different work mode and transfer protocol of serial device server to be more stable and reliable.

The use of serial device server is basically divided into two kinds: virtual serial port and non-virtual serial port, respectively, as shown in figure 21 TCP communication diagram and figure 24 virtual serial port use. The virtual serial port mode needs the connected user software is COM port, as user software and user device are serial ports; in Non-virtual serial port mode user software is direct TCP/IP communication but user device is still serial port.

In the non-virtual serial port mode, the “Transfer Protocol” have 3 mode of transparent transmission, Modbus TCP to RTU and Realcom protocol. If the user software is fixed protocol Modbus TCP protocol and the lower computer is Modbus RTU, choose the Modbus TCP to RTU mode. Realcom protocol currently only use in when multi-port serial device server as TCP Client to connect one Server and the Server use virtual serial port.

The usage is summarized as follows:

Table 6 Network Configuration Mode

No	Virtual Serial Port Use	Device Work Mode	Transfer Protocol	Instruction
1	Use	TCP Server	None	Suitable for user software to open COM port to actively collect data.
2	Use	TCP Client	None	Suitable for device actively send data, if choose TCP Server it may have the problem that device cannot reconnect after broken.
3	No use	TCP Server	Modbus TCP to RTU	Suitable for that the user software is Modbus TCP and user device is Modbus RTU, as well the Modbus TCP is main station.
4	No use	TCP Client	Modbus TCP to RTU	Suitable for user software is Modbus TCP and user device is

				Modbus RTU, as well the Modbus RTU is main station.
5	Use	TCP Client	Realcom protocol	Multi-port serial device server as TCP Client, and use virtual serial port, better use Realcom protocol.
6	No use	TCP Client	None	Suitable for a large number of devices to connect a cloud, and generally the cloud is the server with a public IP in Internet.
7	No use	TCP Server	None	Suitable for device and computer in the same local network, monitor locally without cross-Internet communication.

5.1 Virtual Serial Port Mode

If the user software uses COM port to communicate, the virtual serial port mode must be used. As some PLC software, configuration software, instrument software and so on.

Check to see if both the computer and the device are on the local network:

1) If the computer is a server that rents a public network IP on the Internet, then the device must use the TCP client to connect the server. At this point, you can select the ② and the ⑤ in table 6, and if it is multiple serial port servers, you must choose the ⑤.

2) Both are in the local network (which can ping each other), depending on whether the upper machine active query or the device initiatively sends the data. If it is the device initiatively sending data, it is necessary to use the ② type that the device as TCP Client, otherwise you can choose the ① type.

5.2 Direct TCP/IP Communication Mode

If you don't need the Modbus TCP protocol conversion and virtual serial port, here the user software can directly do TCP/IP communication with the Ethernet port of serial device server, the serial device server convert the TCP/IP data to serial port data.

In general, users of this kind of usage develop their own upper computer network

communication software and integrating the analysis of the serial communication protocol of the device. This method is more flexible and efficient than virtual serial port. Corresponding to ⑥ and ⑦ in Table 6.

In the section “4.3 TCP Communication Test” there mainly describe how to do communication when serial device server as TCP Server. Here we will describe how the TCP Client, UDP mode and multi-TCP connection communicate with computer software. And the computer software takes SocketTest (simulate user TCP/IP communication software) for example.

ZLAN serial device server follow the standard TCP/IP protocol, any network terminal comply with the protocol can communicate with the serial device server. ZLAN provides network debugging tool (SocketDlgTest programs) to simulate the network terminal to communicate with the serial device server.

To realize 2 network terminal (here as the network debugging tool and serial device server) can do communication, the parameter configuration must be matched.

5.2.1 TCP Client Mode

Work mode in the TCP mode has two type: TCP server and TCP client, no matter adopt what kind of mode, must one is the Server, the other is the Client, then Client can access the Server, both for the Client or the Server is unable to realize communication.

When serial device server is used as Client, there must be three corresponding relationships, as shown in figure 32. 1) work mode corresponding: The Work Mode of serial device server as Client Mode corresponding to the Server Mode of network tools, 2) IP address corresponding: the destination IP of serial device server must be the IP address of the computer which network tools in, 3) port corresponding: the destination port of serial device server must be the local port of network tools. The networking product will automatically connect the network tools after setting, and the data can be sent and received after the connection is established.

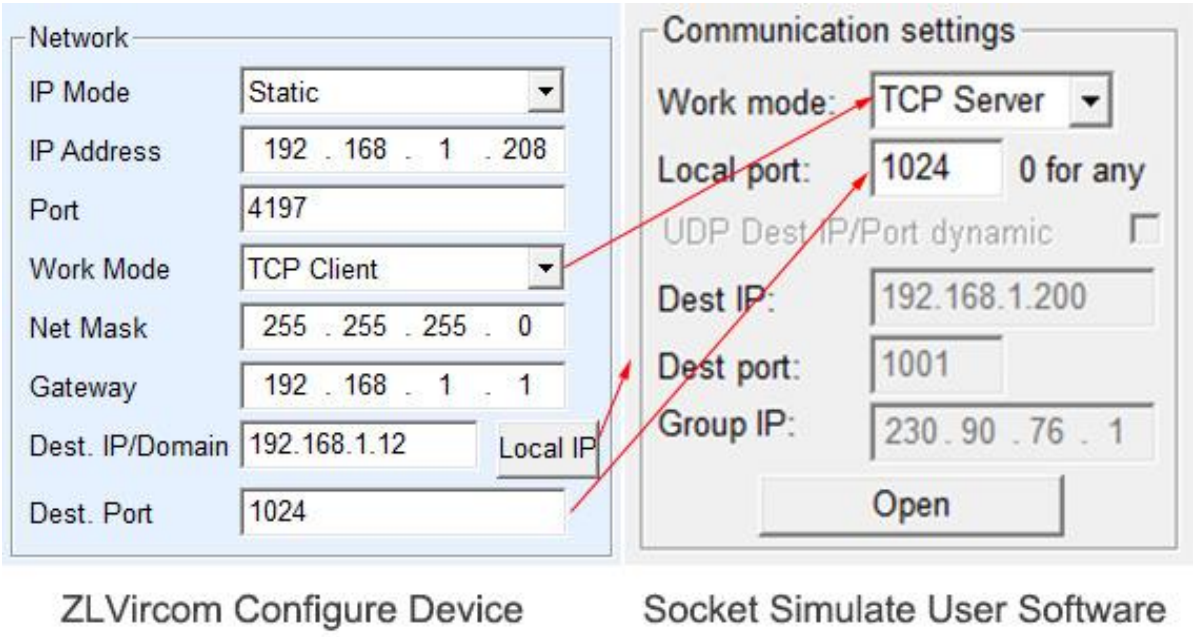


Figure 32 Serial Device Server as Client

5.2.2 Client connect to Multiple Servers

ZLAN serial device server can connect 7 destination IP address at the same time when as TCP Client, the data sent by serial port will be sent to 7 destination IP. If there don't have so many servers, just leave the other destination IP to opening. The usage as below:

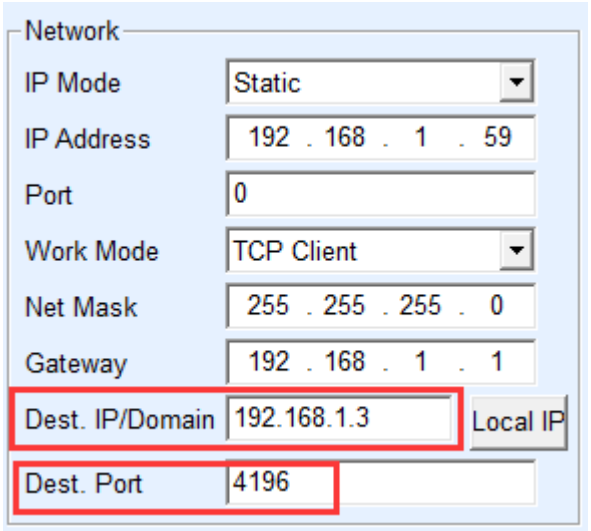


Figure 33 The First Destination IP and Port

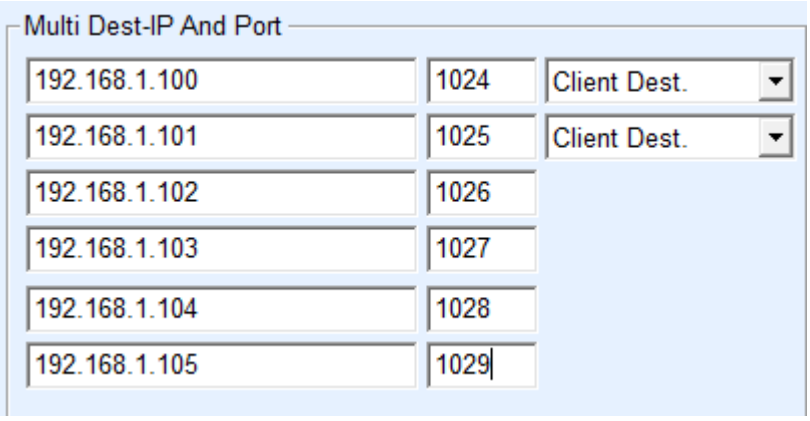


Figure 34 The Rest 2~7 IP and Ports

The first IP is set on the device manage interface as shown in figure 33, where the first IP can be a domain name. The setting of rest 2 to 7 destination IP clicks the "more advanced setting" button in the device manage interface, open more advanced options to set.

All the 7 destination IP finish setting it can be connected automatically, if the connection is not connected, it will repeat reconnecting after waiting for the "broken line reconnect" time.

5.2.3 TCP Server Mode

There are also three corresponding relationships when serial device server as Server, as shown in figure 35. After this setting, click on the open button of the network tool to establish a TCP connection with the networking product, and the data can be sent and received after the connection is established.

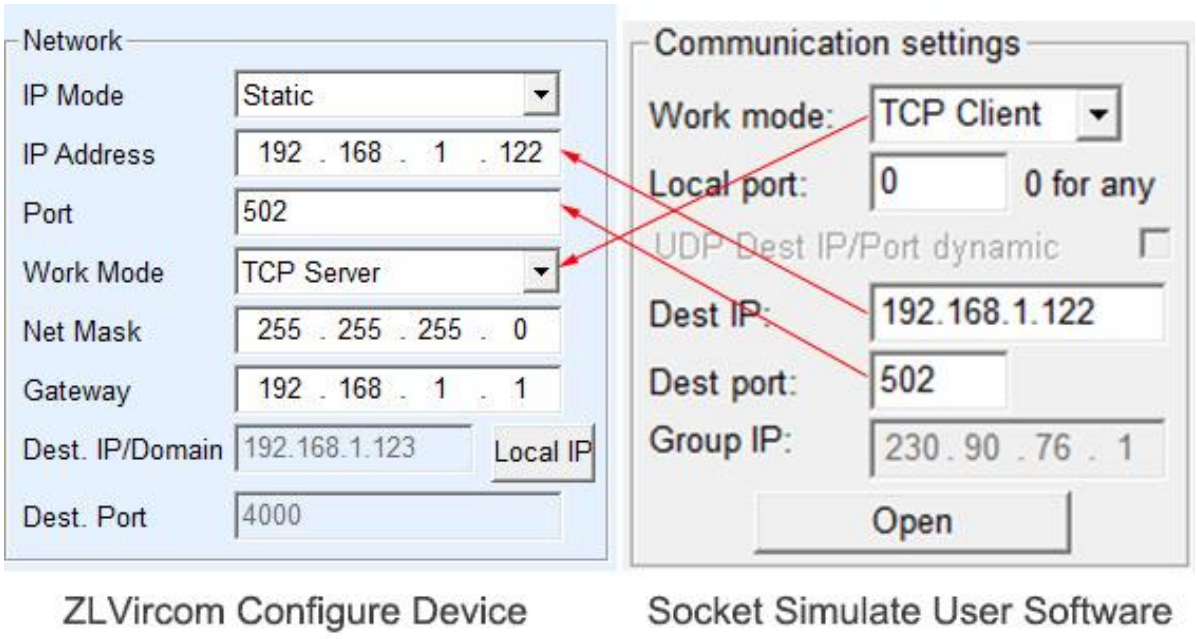


Figure 35 Serial Device Server as Server

The serial device server can accept 30 TCP connections at the same time when as Server. The data the serial port received will be transferred to all the established TCP connections. If you want to realize the data only send to the TCP that received network data package recently, you need to enable the multi-host function, please refer to 7.4 Multi-host Function.

5.2.4 Be Client as well Server

The ZLAN serial device server support the device can accept TCP connections when in TCP Client mode, as also have the TCP Server function.

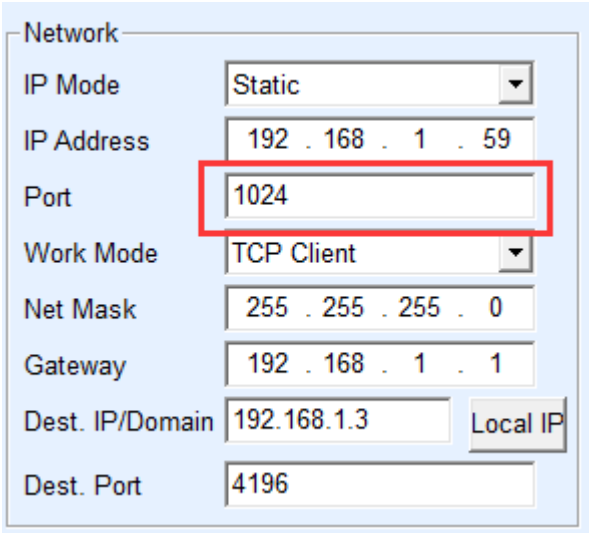


Figure 36 Both as Client and Server

By default use ZLVircom to configure, if change the work mode to “TCP Client” mode, the port (local port) will be automatically changed to 0 (0 means randomly choose a free port). In order to support the TCP Server mode, the computer software must know the local port of the device, here need to specify a value as shown in figure 36. The computer software can connect the 1024 port of 192.168.1.59 to communication, and meanwhile the device will be as Client to connect the 1024 port of 192.168.1.3. Should be noted that because the local port 1024 is occupied by Server, when as Client the local port use “local port +1”, as the origin port of the device seen in the software of 192.168.1.3 is 1024+1=1025.

5.2.5 UDP mode

In UDP mode, the parameter configuration is shown in figure 37, left is the configuration of serial device server in vircom, and right is the setting of SocketDlgTest for network debugging tools. First the two must be both UDP work modes. In addition, the red arrows indicate that the destination IP and port of network tool must point to those of serial device server. The blue arrows indicate that the destination IP of serial device server must be the IP address of computer which the network tool in, and the destination port of serial device server must be the local port of network debugging tool. These network parameters are configured to ensure two-way UDP data communication.

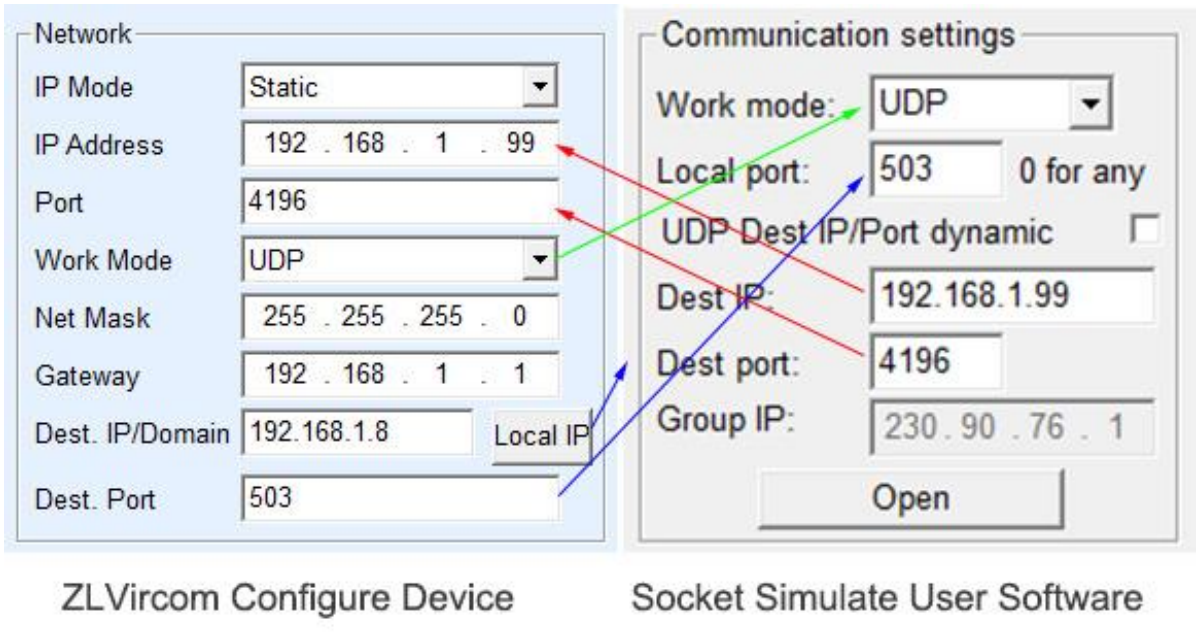


Figure 37 UDP Mode Parameter Configuration

5.2.6 Pair-Connection Mode

If the host is not a Socket program (SocketDlgTest) or ZLVircom, but the two devices are connected via the Ethernet port, the configuration method is similar. First, users need to connect two devices and the computer to the same LAN. This computer runs ZLVircom (or ZLDevManage), it is just to configure, after configuration there no need to connect.

Click on ZLVircom's Device Manage to find these two devices, as shown in figure 39. Then click "device edit" to configure the device. Device pair-connection can be divided into TCP pair-connection and UDP pair-connection. If it is a TCP pair-connection, the parameters of the two devices are shown in figure 38. The parameters shown by the arrow must correspond as the corresponding mode of connection to the PC machine. After the success of the TCP connection, can return to the "Device Manage" dialog to see the connection status, as shown in figure 39, if the state of the two devices are "connected" say TCP link has been established between the two devices.

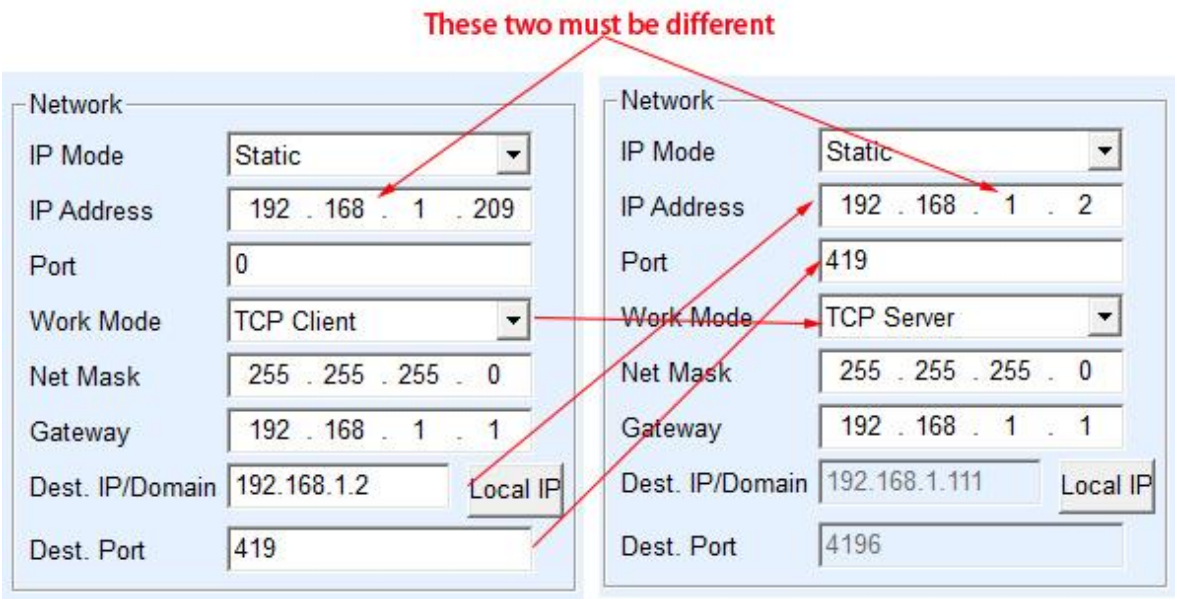


Figure 38 TCP Device Pair-connection Configuration

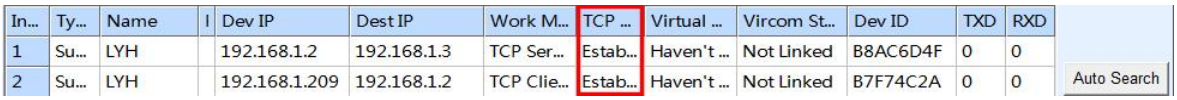


Figure 39 TCP Devices Pair-connection Success Check

If the pair-connection in UDP mode, the configuration parameters are shown in figure 40, and the corresponding parameters of the arrows must be one-to-one. In UDP pair-connection the data will automatically be sent to the specified device as long as the parameters are configured correctly without checking the connection status.

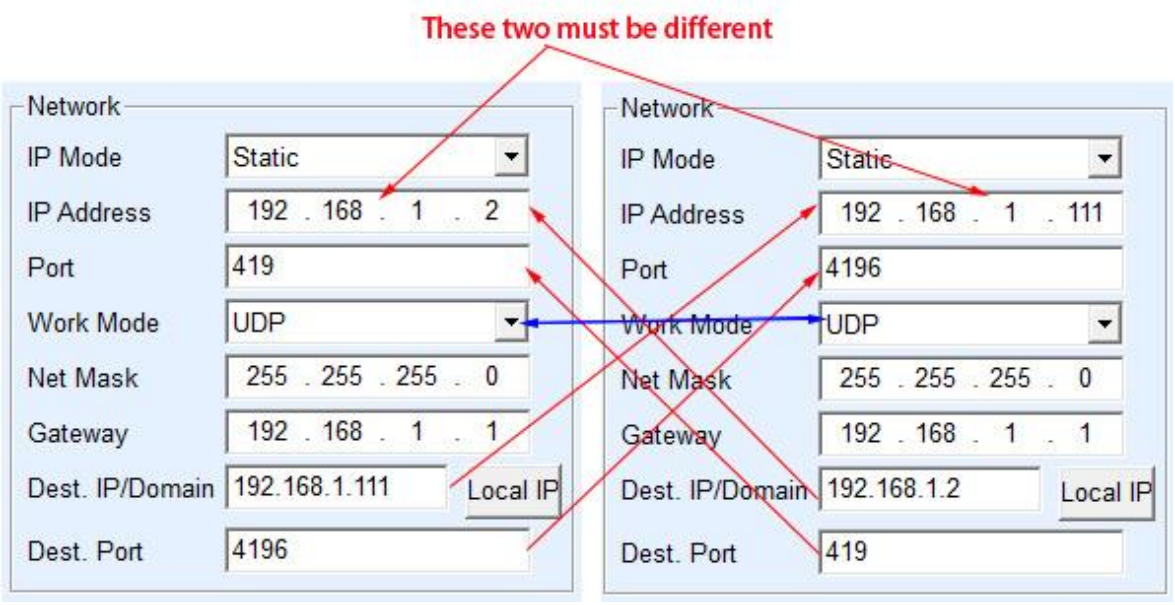


Figure 40 UDP Device Pair-connection Configuration

Finally, it is necessary to remind that if the device is pair-connected, except the Ethernet parameter configuration set as above, the serial port parameters also need to be correctly set. It is mainly because the baud rate of the serial device server and the baud rate of the user's device should be accordance. After this setting, user devices can send data to each other through the serial port of two serial device servers.

6. Device Debugging

6.1 Network Physical Connection

The serial device server can use cross wire or straight wire to connect 10M/100M switch or directly connect to computer Ethernet port.

The first step after establishing connection is to check whether the Link light is green, otherwise please check network wire whether connected.

6.2 Network TCP Connection

The device cannot directly connect to computer Ethernet port when it's in dynamic get IP mode. Because there is no DHCP for use (normally the DHCP server is the router in LAN), the direct connection please specify the IP, also the computer need to specify fixed IP.

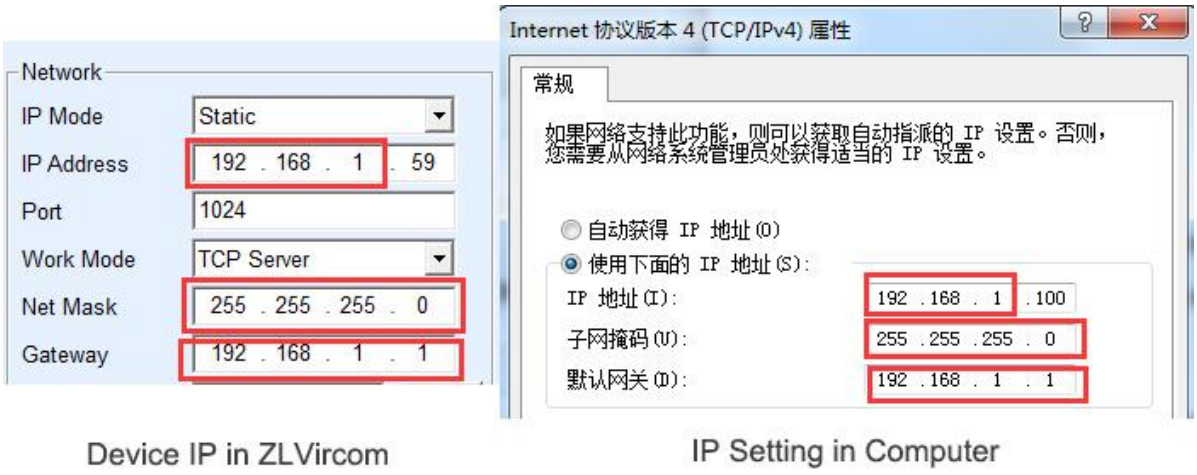


Figure 41 Set in One Network segment

Whether direct connection or through switch, when set to static IP, the device and computer need to be in same network segment (except the cross-gateway communication), as show in figure 41.

As the ZLVircom support cross-network segments to search and configure, if can search but cannot communicate mostly probably the IP address wasn't configured well, at the situation you can use ZLVircom to configure the device in same network segment.

After configured using the steps of 4.3 TCP Communication Test or 4.4 Virtual Serial Port Test you can see the Link light become blue when establishing TCP connections. The Link light being blue also can be seen in ZLVircom, as in the device manage list if the TCP Connection is "Established" that means the Link light is blue, it convenient for remote diagnosis as in figure 42.

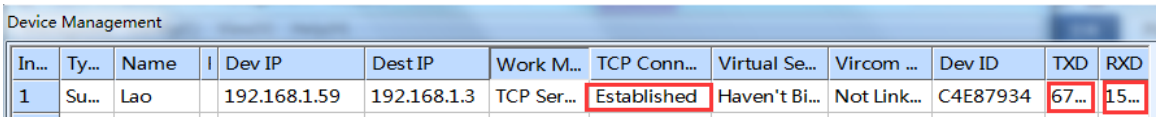


Figure 42 Connection Status and Data Send-receive Status

6.3 Data Sending and Receiving

When the Link light becomes blue, the software and the serial device server can do data receive and send. Here if the software send a data the Active light will become green, last for at least 1 second. The data also will output from the serial port of the serial device server, and whether the output data is still correct you need to check whether the serial port parameters (baud rate, data bit, stop bit, parity bit) are configure correct.

For the correct sending instructions the serial device server usually will reply, once there have the reply (serial port send data to Ethernet port), the Active will become to blue, or else please check the serial port parameter or the serial port line connection whether have problem.

In order to facilitate remote debugging, ZLVircom also supports remote watch the send and receive data, as shown in figure 42, the TXD is the amount of the data output by serial port of the serial device server, when refresh the device list, the value changing means that there have sending data, the Active light also be green; if you see the RXD value is changing it means the serial device return data, the Active light is blue.

6.4 ZLVircom Remote Monitoring Data

In the case of virtual serial port, ZLVircom supports real-time fetching the send-receiving data of virtual serial port. Convenient for user debugging system, usage as below:

Assume that here already establish the communication of virtual serial port according to the method of 4.4 Virtual Serial Port Test, now there need to monitor the data through virtual serial port. Open the ZLVircom Menu/Config/Software Setting to open the vircom configuration dialog.

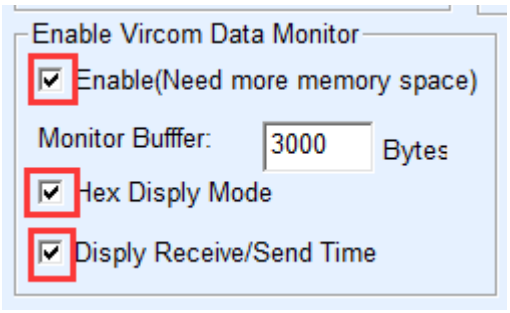


Figure 43 Enable ZLVircom Monitoring

Select the three options Enable, Hex Display Mode, Display Receive/Send Time as figure 43. And click ok. Assume that the data receive/send has been done, now choose the virtual serial port that need to monitor in the main interface, and choose Menu/View/Monitor as show in figure 44.

Check before starting the monitor, hexadecimal monitoring mode, and display data sending and receiving time 3 options, as shown in figure 43. Then click ok. Assuming that the data has been sent and received before, now select a virtual serial port that needs to be monitored in the main interface, and then select the menu/view/monitor, as shown in figure 44.

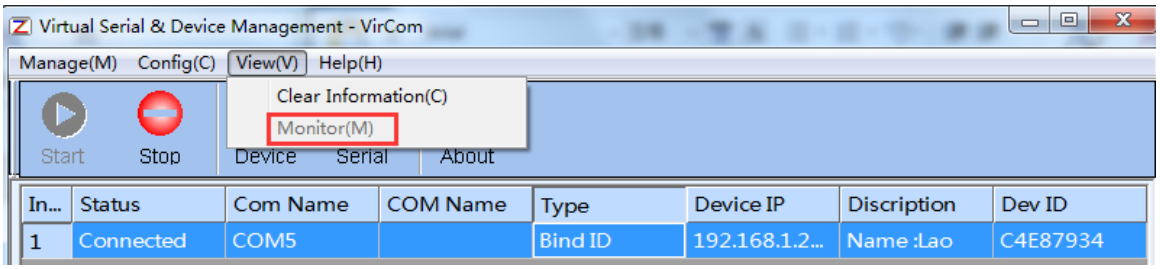


Figure 44 Open ZLVirocm Monitoring

From the open dialog box, you can see the instructions sent by the upper machine and the instructions returned by the device, as shown in figure 45. This function can facilitate the field communication debugging.

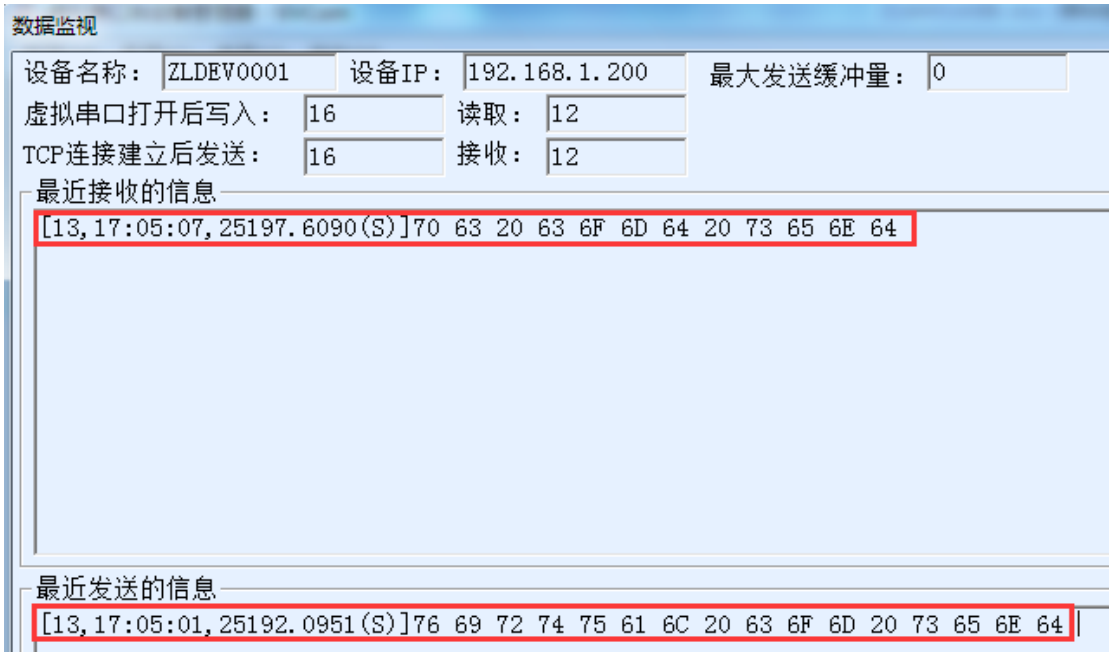


Figure 45 Monitor the Send/Receive Data

7. Modbus Advanced Function

The serial device server with Modbus Gateway function itself does not have the station address and registers, it's a bridge of communication, and will produce Modbus RTU specify appointment according to the Slave ID, function code, register number in the Modbus TCP instructions which user software send to Modbus Gateway, and output from serial port. You can see it as a "Protocol Translator".

7.1 Enable Modbus Gateway

First the serial device server should support Modbus Gateway as the "Modbus TCP to RTU" function in the device supporting function of Table 5 device manage dialog box should be selected.

By default, the serial device server is in the normal pass-through mode. If it needs to be converted to Modbus gateway mode, please select "Modbus TCP<->RTU" in the "Transfer Protocol". Since then, the device automatically changes the "port" parameter to 502 (the port of Modbus server).

The serial port RTU device is as slave, the host Modbus TCP software connects the 502 port of Modbus Gateway, and the Modbus Gateway need to work in TCP Server Mode; if

the serial port RTU device is as master, the Modbus Gateway work in TCP Client, and the destination IP fill the IP of the computer that Modbus TCP software in, the destination port usually is 502.

7.2 Storage Modbus Gateway

The new generation of ZLAN5143 (end with 3 all storage) is Modbus register storage type gateway, compared with the ordinary ZLAN5142 (end with 2 or 0 all non-storage), ZLAN5143 can store the content of reading register inside the gateway, so the Modbus TCP can greatly improve the query speed, support for performance more superior when multiple host access.

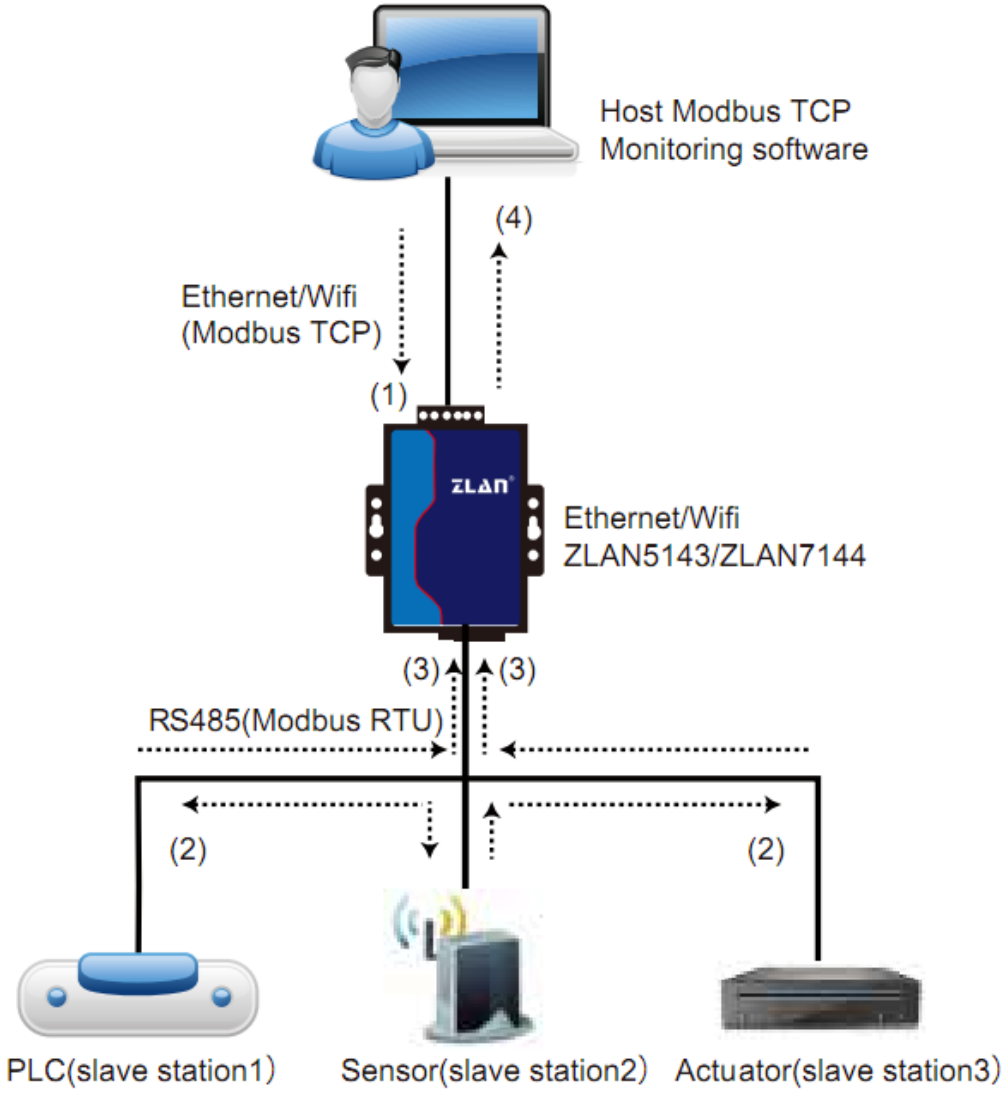


Figure 46 Storage Modbus Gateway Work Mode

As shown in the figure 46, the general Modbus TCP data flow direction is (1) → (2) → (3) → (4). First, the Modbus TCP command is transformed into the corresponding instruction of Modbus RTU, and then the device responds Modbus RTU instruction to Modbus gateway, and then Modbus gateway is converted again to Modbus TCP to be sent to the monitoring upper computer.

We know that Modbus TCP is a network communication, and the transmission speed is very fast. Generally, it can be answered in 3ms, while Modbus RTU is RS485, which generally only has a speed of 9600bps, and generally sends and returns an instruction at least 30ms. This common non-storage Modbus gateway has a longer query response time. In addition, if there are a lot of upper computer at the same time to query the data, then the serial port will be congested. If the network is compared to a highway, the serial port is a foot-bridge, then the original way is to carry the highway traffic on the foot-bridge.

The storage Modbus gateway solves these problems. It can query for the register data temporarily stored in the Modbus gateway inside, so the Modbus TCP queries, Modbus gateway can immediately return instructions, really the features of the Modbus TCP quick play out. On the other hand, the saved Modbus gateway can automatically update the contents of the current saved register data by sending instructions automatically from the serial port, and keep a new register value.

The storage Modbus gateway ZLAN5143 is a fully automatic Modbus gateway with no configuration required. The user does not need to configure the required register address, function code, slave station address and so on. ZLAN5143 will automatically identify and dynamically add these registers.

ZLAN5143 can show a good reaction speed in multiple computer monitoring, regardless of the baud rate of serial port, it is generally able to give the upper response data within 3ms. It also shows the speed of real-time update of serial data.

Register storage Modbus gateway is the real Modbus TCP to Modbus RTU, which really gives full play to the advantages of fast speed, multi-host simultaneous query of Modbus TCP.

Note that when the serial device server is used as the TCP client, it does not have the storage function and will automatically switch to the non-storage type.

The features of the storage Modbus are listed below:

1) The first Modbus TCP query command is non-storage. It is necessary to wait for the RTU device to return the data at a slow speed before replying the register content to the Ethernet port.

2) If a specific instruction no longer has the query form upper computer on the network end in 5 seconds, it will automatically delete this instruction and no longer send it from the serial port to the RTU device.

3) At present, the Modbus cache of 10K can be stored. For ordinary single register query, about 500 instructions are stored at the same time.

4) When there are multiple instructions queried at the same time, send in accordance with the order, send first instruction → first instruction reply → wait for 485 anti-collision time (refer to Multi-host part) → send the next instruction... . Go back to the first instruction after the last instruction finished reply.

7.3 Disable Storage Function

Although the storage Modbus has a fast response speed, some users do not wish that RTU devices to receive a large number of query instructions of affecting the internal processing speed of the instrument. You can turn off the storage function at this point.

The way to disable storage is to click on the "More Advanced Setting" button in the “Device Setting” dialog box, removing the Support and the Enable as shown in figure 47 and click ok. Go back to the device setting and click Modify Setting.

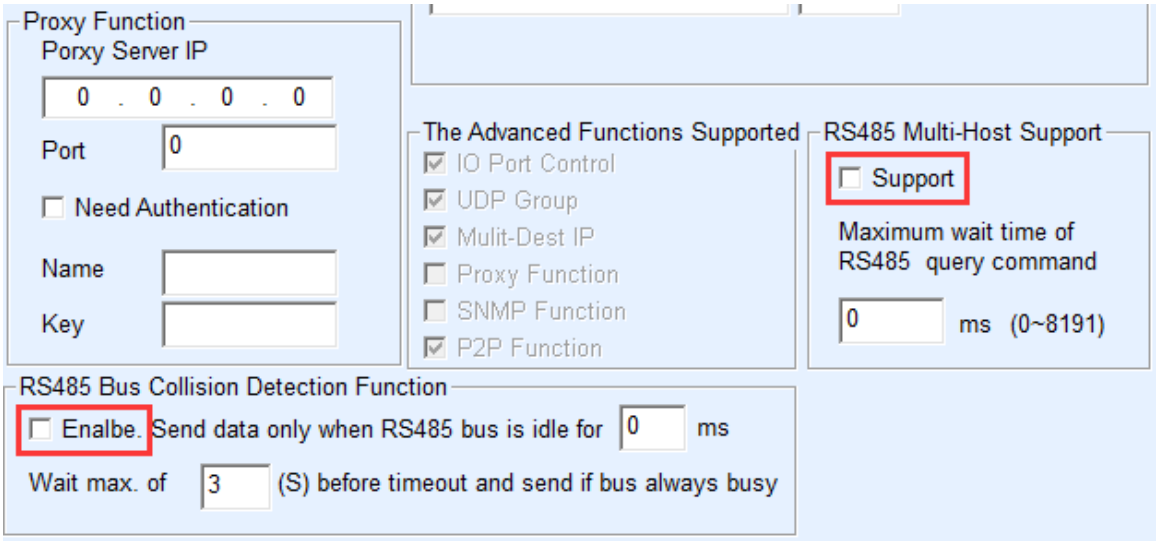


Figure 47 Disable Storage Function

7.4 Multi-host Function

As shown in figure 47, "RS458 Multi-host Support" and "RS485 Bus Collision Detection Function" are ZLAN multi-host functions. They are generally both enabled and disabled at the same time. After enabling, the device with Modbus TCP transfer protocol has the function of storage Modbus gateway, otherwise it is a non-storage Modbus gateway; If the transfer protocol is None, usually can make the RS485 protocol of user customized have the function of multi-host access the serial device at the same time, it cannot be realized in the pure RS485 network, because multiple host simultaneous sending will cause conflict on the RS485 bus. The multi-host of the ZLAN serial device server can "coordinate" RS485 bus to achieve the purpose of multi-host access.

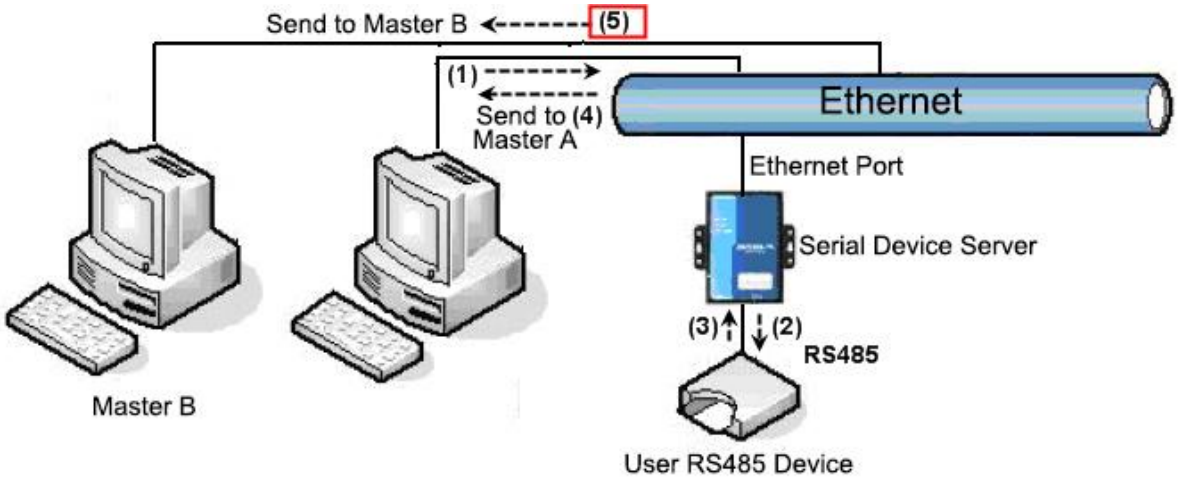


Figure 48 Multi-host Function Demonstration

As shown in figure 48, in common mode, when the two hosts: host A and host B connect the serial device server at the same time, host A send (1) instruction, RS485 device received (2) instruction, RS485 device return (3) instruction, but at the Ethernet port of the serial device server it will send (4) to host A and (5) to host B at the same time. Because host B didn't sent query but receives the reply order (5), it may cause communication error. In multi-host mode, there only have the instruction (4) without (5), as the serial device server will automatically remember the host required return, only return the instruction to the recent communication host, as host A query only return to A, host B query only return to host B.

The other effect is that in normal mode the data sent by host A and host B at the same time will do instruction combination at the RS485 bus, thus unable to identify properly; The serial device server can dispatch A and B to use the sequence of the bus, so as to effectively solve the conflict problem of multi-host simultaneous access.

7.5 Multi-host Parameter

"RS485 Multi-host Support" and "RS485 Bus Collision Detection Function" are introduced as follows.

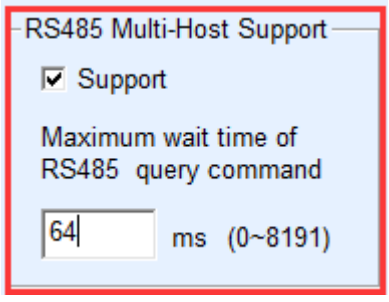


Figure 49 RS485 Multi-host Support

The maximum wait time of RS485 query command is: the maximum time interval of the serial device server serial port sending the command to receiving the reply. The filling time should be greater than the actual maximum time interval. Because if identified to timed out, it will send the next instruction.

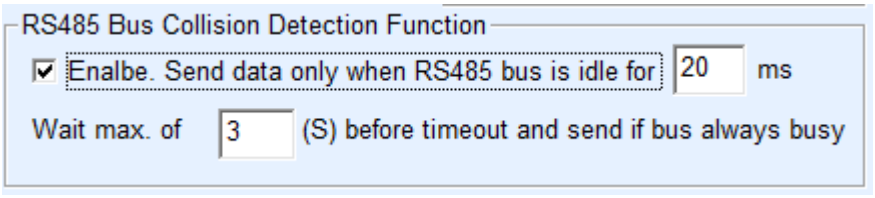


Figure 50 RS485 Anti-collision Idle Time

RS485 bus collision time: represents the waiting milliseconds the serial device server send the second instruction after receiving the reply of the first instruction. This parameter actually defines the speed of instruction rotation. This value is recommended above 20ms. The "Wait max. of 3 (s)" is not required to be modified in general.

When user use ZLVircom to select the transfer protocol to “Modbus TCP to RTU”, ZLVircom will automatic select the above two options (unless the user manually enter the advanced setting to remove), as well the above two time will be configured well automatically according to baud rate. But if the user Modbus command is longer or if the conversion protocol is "None", then the two parameters need to be manually configured.

The recommended values for configuring the above parameters are described below:

1) Figure 50 shows the "RS485 Bus Anti-collision Time", which can be set to double the "Packet Interval" in the lower right corner of the parameter setting interface, but the minimum cannot be less than 20.

2) As shown in figure 49 "Maximum wait time of RS485 query command", generally determined according to the instruction length of responses back and forth, if send instructions for N bytes, response to M byte, it is recommended that the set value is: "packet interval" x (N + M + 5) + 100.

8. Register Package and Heartbeat Packet

Register package and heartbeat packet are a feature suitable for communication between devices and cloud software.

8.1 Register Packet

Register package is defined as, when the computer software and the serial device server

module (hereafter referred to as "module") establish a TCP connection, the module will first send a bunch of code to the software, thus the software can know which module is in communication. This string of code is the register package.

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

Serial device server of Shanghai ZLAN provides the following types of register package.

8.1.1 Connect to send MAC address

Connect to the MAC address: this is not only for the 4 models (such as 5143), but also for regular models. The method is to send your MAC address to the cloud when the module is connected to the cloud. Because the MAC address is unique, it can only identify the device. This approach is simple and no need to write the register package of each device with simple and effective. Using the method is: in the device Settings dialog box, click on the "more advanced setting", found “Send Mac when TCP establish” in the upper and select, then back to the setting interface, click "Modify Settings".

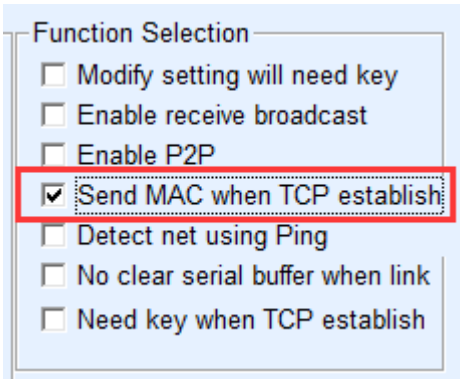


Figure 51 Connect to send MAC address

8.1.2 Realcom Agreement

The Realcom protocol is a mature protocol that contains register package and heartbeat packet that users can use to implement the function of register and heartbeat package. The way to enable Realcom protocol is to select the "Transfer

Protocol" to "REAL_COM Protocol" in the "Device Settings" dialog box, and note that the register packet part needs to be blank and not selected.

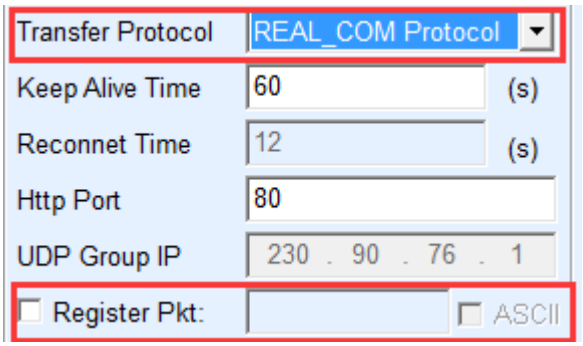


Figure 52 Enable REAL_COM Protocol

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

1) When a TCP connection is established between the device and the cloud, the device automatically sends a hexadecimal register package FA 07 13 02 FA 02 MAC[5] MAC[4] MAC[3] MAC[2] MAC[1] MAC[0] FA FF. MAC[5]~MAC[0] is the MAC address of the device.

2) When the device sends data to the network, it automatically adds three bytes of the head prefix to FA 01 01.

3) The device sends a one-byte heartbeat packet of 00 to the software at every keeping alive time.

The REAL_COM protocol can be used as the registered package for the device because the register package contains a MAC address. But because of its fixed format, cloud-based software can only be used to design a REALCOM protocol to be compatible with this approach.

8.1.3 Custom Register Package

The custom register package mode is that the user can fill in an arbitrary registration package format. The method is: in the device setting interface, the configuration is as follows:

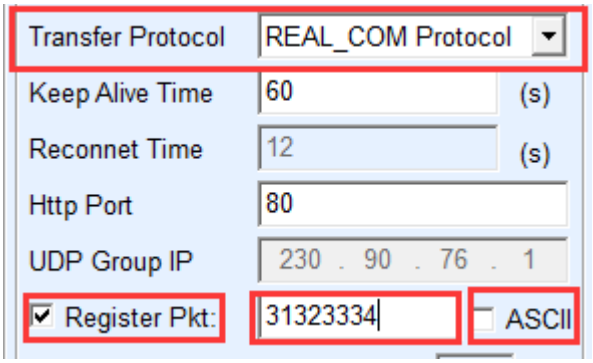


Figure 53 Configure Register Packet

The difference with the REAL_COM protocol is that the register package is enabled, and fill in the register package information such as 31 32 33 34. Notice that this is the hexadecimal, which is the actual number of data that you're sending is a string of 1234. If you need a string display, click the next "ASCII" option.

When the device is connected to the cloud software, a hexadecimal register package of 31 32 33 34 can be automatically sent. This type of register package is flexible and allows the device to adapt to the existing cloud register package format; However, there is no such wildcard as a MAC in the register package, and different registration packages need to be configured separately for each device. The above two methods of sending MAC address and REALCOM have same configuration for each device, but because the MAC is different the register packages different.

The longest registered package has a length of 33 bytes. This mode supports the register package and heartbeat package of UDP mode.

8.1.4 Transcode Configuration File

ZLAN 5143 series support "transcoding" function, the function can write a a transcoding configuration form for the serial device server, so as to realize fully customizable user registration package, and you can use MAC address wildcard, can solve the problem of writing custom register package for each device, and there is no limit to the register package length.

The specific use method can consult ZLAN about "transcoding" function configuration, or refer to 10 Transcoding Function.

8.2 Heartbeat Packet

The heartbeat packet is mainly used to detect whether the communication link is disconnected. The approach is to send a heartbeat packet data to the server software every once in a while, which is discarded by the server after receiving and will not be used as an effective communication data.

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

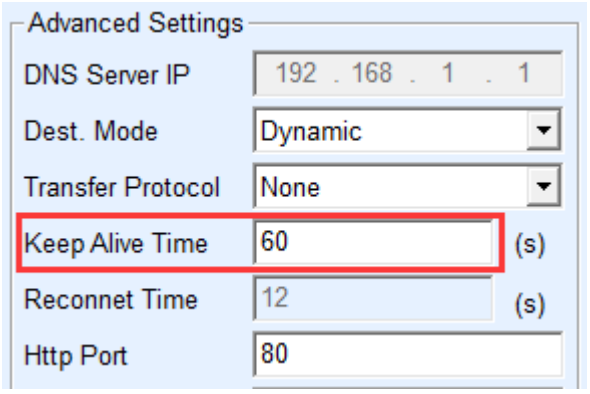


Figure 54 Keep Alive Time

As shown in figure 54, the sending time of heartbeat packet is set by "Keep Alive Time".

8.2.1 Implied Heartbeat

Even if you don't set up any heartbeat packets, the ZLAN device will enable the hidden heartbeat function when it is in the TCP client. So the implied heartbeat function is the device sending data, but the server actually does not receive the heartbeat data. So it can't achieve the first function of the heartbeat packet, that is, the function of whether the server detects the activity or not; But because actually the device has sent data, it can achieve the second function of heartbeat packets, that is the function of the device detecting whether the TCP connection is normal, once detecting the disconnection it still can automatically establish a TCP connection.

8.2.2 REALCOM Agreement

As stated in the Realcom protocol, the Realcom protocol can send a one-byte heartbeat packet of 00 to the software at every keeping alive time, which is the heartbeat package of the Realcom protocol.

8.2.3 Custom Heartbeat Packet

First, fill in the register package according to 8.1.3 Custom Register Package. Then increase the heartbeat packet according to the following method: click on "More Advanced Setting" button on the device Settings, write the hexadecimal heartbeat packets on the second line of the destination IP and port, and change the right option to "Param. Dest.".

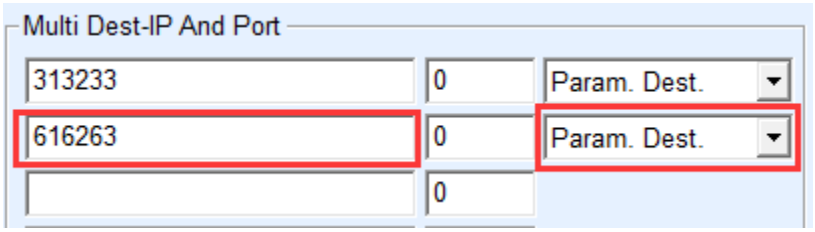


Figure 55 Custom Heartbeat Package

Note that the total number of register packages and heartbeat packages is less than 33 bytes. The first line is actually the register package.

9. Httpd Client Communication Function

This feature is used to directly send the data up-sent by serial device server to a web based server program, which can simplify the software development effort on the cloud.

When IoT acquisition terminal and the web server (httpd program) do interaction, if it can submit the data according to the standard format of http GET and Post command to web server, the web server can use the existing PHP/asp language for data processing and storage. This saves the effort to redevelop web application interface for user.

To support this feature, you need to download a httpd.txt configuration file in the ZLAN serial device server. Download can be implemented by using firmware upgrade function of zlvircom.

The features of ZLAN httpd client communication include:

1) Device up-sending: support converting the serial port data into http format by means of GET/POST, and can be directly identified by the server.

2) Web server down-sending: the Web server can also send the required data to the serial device server through the GET/POST command, and the valid data contents can be output from the serial port of the serial device server. When the serial server receives the data, it can also give a specific response to the Web server, indicating that the data is received.

3) Support the input and output data in any conversion between hexadecimal and strings, convenient for Web server down-sending data by characters, and the serial port output in hexadecimal data to control the serial port device.

The detailed information can be referred to the document of the "ZLAN HTTPD Client Communication Mode".

10. Transcoding Function

The transcoding function of ZLAN serial device server can transform the different device protocol into a unified protocol.

Table 7 Transcoding Example

Ethernet Port Instruction	Serial Port Instruction
01 02 03 04	a1 a2 a3 a4
11 12 13 14	b1 b2 b3 b4
21 22 23 24	c1 c2 c3 c4

As shown in table 7, when the Ethernet port received 01 02 03 04, the serial port will output the instruction of a1 a2 a3 a4, and the Ethernet port actually sent 11 12 13 14 when it received b1 b2 b3 b4. This enables the transformation of different protocols. This is just a simple example of the "Transcoding" function, in practice can transfer more complex protocols.

10.1 Enable Transcoding

The configuration file for the command transformation is written to the httpd.txt file, and downloaded it to ZLAN serial device server. The step is to create an httpd.txt document in the web directory. If the user does not have a web directory, you can create a new directory and create a separate httpd.txt file.

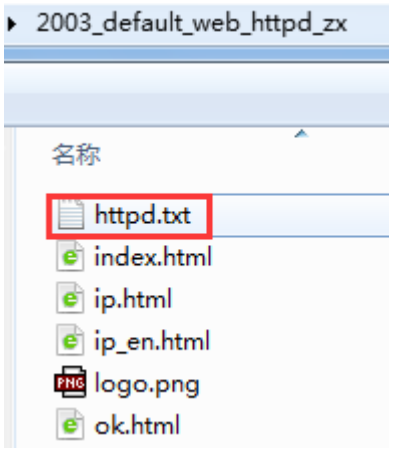


Figure 56 Web Directory

Now click the "Load Firmware" button in "Device Setting" to pop up the web and firmware download page:

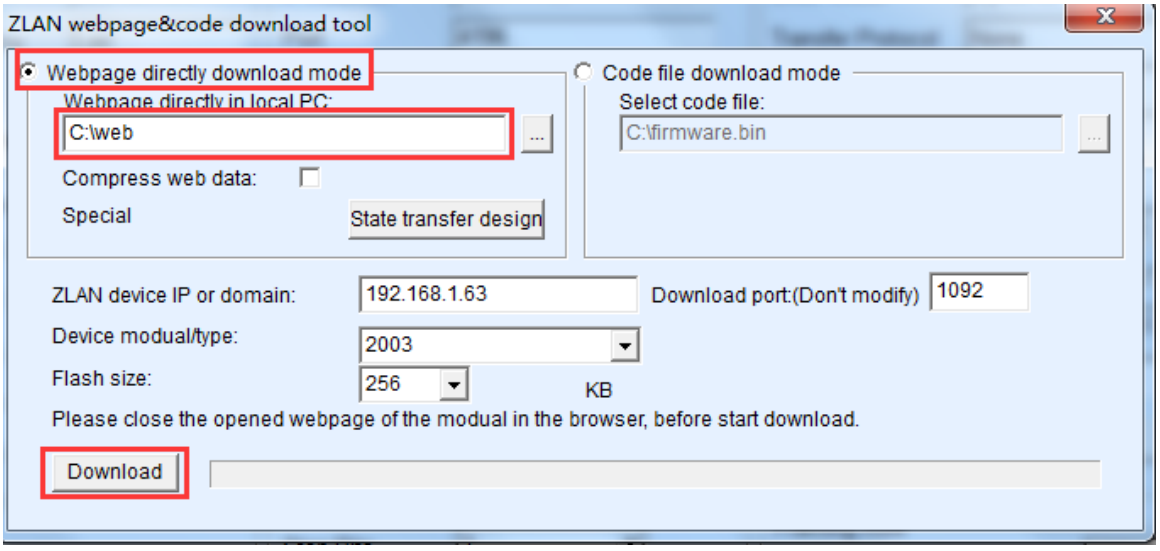


Figure 57 http.txt Download

As shown in above figure, select the “Webpage directly download mode”, and choose the web directory download you just created, click the "download" button to download the httpd.txt to the device. The above IP, model, space size, and port are automatically filled

out without configuration. Note: any time you update httpd.txt, you need to re-power on the device.

10.2 Case of Transcoding Implementation

Because the transcoding function is more complex and longer, here don't introduce the detailed usage, if need it, ZLAN engineer can write the configuration file httpd.txt, which can also provide the configuration file httpd.txt with existing case. The typical cases are as follows:

1) Fixed command translation and conversion.

2) Complex command transformation: the conversion of commands with wildcards, which can shift and assemble commands, such as non-standard RS485 instructions to standard Modbus RTU instructions.

3) Multi-TCP connection identification: when there are multiple TCP connections, the Ethernet port sending data to serial port may add 4 byte prefixes of the IP +port. The serial port sending to the Ethernet port, you can also add the IP address + port to send the data to the specified TCP connection.

4) Increase the MAC address to identify the device before sending TCP or UDP.

5) IP filtering: the device only receives data from the Ethernet port that specifies the IP address.

6) IO pin control: controls the level of some output pins by arbitrary specified instructions.

7) Automatically check the contents of the instrument of the lower position, and send the instructions to the cloud software in the fixed format after assembly.

8) Realize heartbeat package and the register packet function for any length with wildcard characters.

9) Combine multiple serial port instructions and send them to the cloud.

10) The super long serial port instruction is split into several smaller instructions to be

read by the Ethernet port device.

10.3 Notes

1) When httpd.txt is not stored, the device will be used as a normal serial device server. The transcoding function is disabled.

2) You need to delete httpd.txt to make it easy to change httpd.txt to 1.txt and then download again. The re-download of device firmware will also delete the httpd.txt file, and the download of the firmware progress bar only moves 1 grid will also delete the httpd.txt. The method of download firmware can refer to the following sections.

3) You must restart the device after downloading the httpd.txt file.

4) When the httpd.txt is edited with error and cause the parser to fail, it may cause the device to unusual start, here put the def switch (reset switch) of the device to the On position, and recharged. At this point, you will not bring httpd.txt to start. Please download the new httpd.txt or delete the httpd.txt and then set def back to Off position.

11. P2P function

The ZLAN1043N chip has P2P function. The P2P function enables users to connect devices via ID (not IP mode) whenever and wherever they are, without forwarding through the server, directly realizing P2P links between devices and computers. There is no need for the device to have a public network IP and do port mapping. It has the characteristics of convenient use and high communication efficiency.

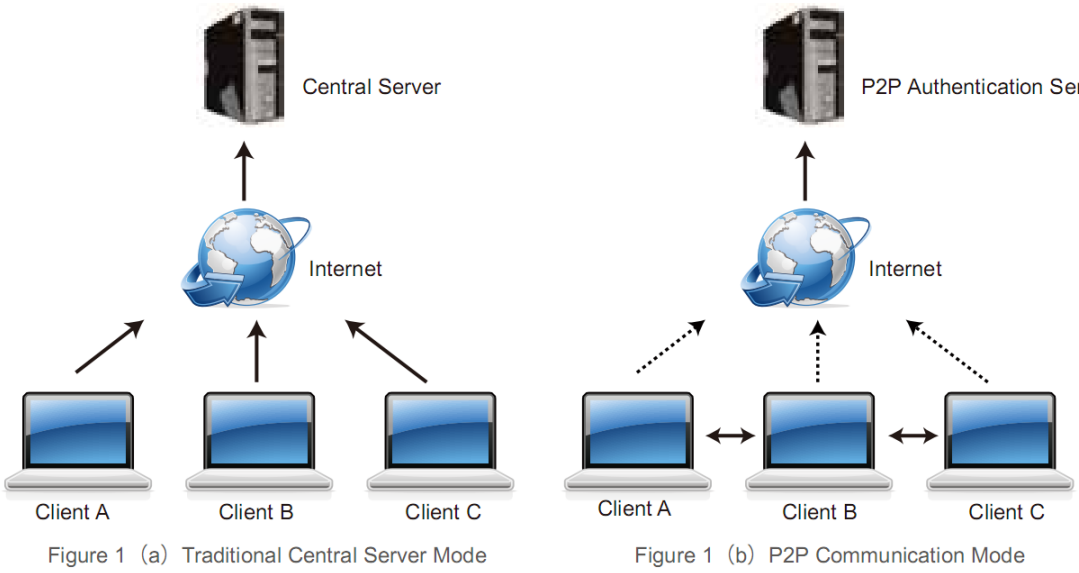


Figure 58 Traditional Mode and P2P Mode

Although the P2P approach also has a central server, the central server is only for verification and connection, and does not participate in the forwarding of data communication.

The ZLAN1043N chip provided by ZLAN must be used in conjunction with the specified P2P authentication server. There are two ways:

1) The ZLAN1043N use ZLAN default P2P server after leaving factory, and users do not need to configure it. The client software also uses the default P2P server to connect device.

2) Users use their own P2P server and purchase P2P server software from Shanghai ZLAN. The ZLAN1043N points to the user's own P2P server after leaving factory.

In addition to providing ZLAN1043N chip, ZLAN can provide:

1) Use the development library of P2P protocol in PC to facilitate users to integrate the P2P functions into their upper computer software.

2) Provide ZLVircom, which supports P2P device manage and P2P based virtual serial port.

The ZLAN1043N chip can provide users with a simple device networking solution that allows

users to connect devices around the world without need to build a complex cloud platform. The specific use method refers to the document of <ZLAN P2P Introduction>.

12. Modify Parameters of Ethernet Port

The modification parameters of the Ethernet port is to realize the functions of searching device and modifying device parameter for ZLVircom software, as through the Ethernet port of the serial device server to manage device and modify parameter. Suitable for integrating search and configuration functions into user software.

The modification parameters of the Ethernet port are realized through the "UDP manage port protocol", such as:

1) The computer software sends UDP broadcast packet with destination port of 1092 on the network. When the device receives the packet, it will return its information to the computer software to achieve the purpose of searching device.

2) The computer software sends UDP modification parameter commands to the 1092 port of the device to achieve the purpose of modifying the device parameters.

The detailed introduction of the modification parameters of the Ethernet port can be referred to the document <UDP Manage Port Agreement of ZLAN Networking Products>. It can also be implemented directly using the device management function library of the 13 Device Manage Function Library.

13. Device Manage Function Library

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

The "UDP manage port protocol" has been integrated into the device management function library ZLDevManage. This is a development library of DLL Windows platform, which can be called by various development tools such as VC, VB and Delphi.

Provide detailed introduction file of API interface and Demo cases of calling VC. It can realize

the device search, parameter modification, P2P function call, etc.

Details can refer to <ZLAN WinP2p and Device Manage Development Library>.

14. Serial Port Modify Parameters

The user can read parameters and set parameters by sending instructions to the serial port of serial device server. It's suitable for users that choose chips or module level products to control and configure through serial port. The parameters that can be set include: IP address, baud rate, device name, work mode, etc. After the new parameters are set, the serial device server can be restarted through serial port instruction.

The ZLAN serial port instruction has the following characteristics:

1) Serial port instructions adopts data lead code of 10 bytes, no need to use additional configuration of polling up-down pins to distinguish whether communication data or command, also no need to switch command mode and communication mode, more flexible and convenient for using.

2) The command set contains a variety of command formats such as saving parameters, not saving parameters, and restarting the device.

3) Can achieve a variety of applications, such as read the MAC address of the serial device server, such as change the work mode of the serial device server from the TCP server to the TCP client mode, can connect to the server actively; You can disconnect from the server when switching from TCP client to TCP server.

The detailed operation method of serial port modifying parameters can refer to: <Serial Port Modify Parameters and Hardware TCPIP Protocol Stack>.

15. Remote Device Manage

The remote device manage is that can use ZLVircom software to maintain and manage for device, including restarting device, modifying parameters and upgrading firmware. This function is suitable for users who manage the device through ZLVircom.

For ZLVircom software, the remote manage can be performed as long as the device can be searched in the device list. The remote manage for device can be divided into the following situations:

1) Automatic search: the device and computer are under the same switch, at this time whether or not in the same network segment, way of ZLVircom searching devices on the computer is: ZLVircom send broadcast queries → all devices reply the ZLVircom tool with own parameters after receiving the query. This method searches all devices at once.

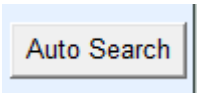


Figure 59 Auto Search

2) Manually add: it is divided into two situations:

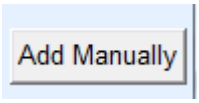


Figure 60 Manually Add

a) Large router segmenting network: in some large networks, broadcast packets are segmented by routers, so that broadcast packets cannot reach the device end, but ping the device IP shows all connected. At this point, you need to manually add it. The manual addition method is to click "Add Manually" in the "Device Manage" dialog box to add head and tail IP then you can query the device one by one.

b) The public network server queries the intranet device: the serial device server is in the intranet and as the TCP server mode, zlvircom is on the server of the public network IP. Here need to make a UDP port mapping of 1092 on the network router the device in, mapping to the IP the device in, then zlvircom manually add this device, IP is the public IP in device side.

3) TCP client: when the device is a TCP client, TCP connection is initiated to port 4196 of destination IP (116.15.2.3). After the connection is established it will automatically send its own parameter system to the UDP port (note that not the TCP port) of the destination

port (here is 4196) each keep alive time, so that they can make zlvircom can search to the device on this computer (116.15.2.3). If the destination port is not 4196 it needs to modify the receiving port of default parameters of ZLVircom, the method is to modify the Menu/Config/Software Setting/Default Listen Port, later restart the ZLVircom, continue to execute if pop-up TCP port conflicts.

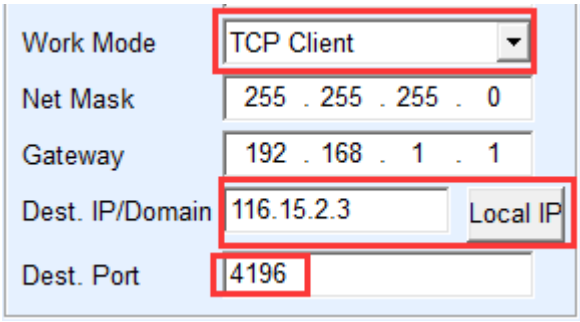


Figure 61 Client

4) Regularly send parameters: even the serial device server is in TCP Server mode, you can select “Enable send parameter” function, and send parameters to the destination port of the destination IP (here 116.15.2.3) every 5 minutes. The ZLVircom receiving parameters in this port of this server can manage the devices.

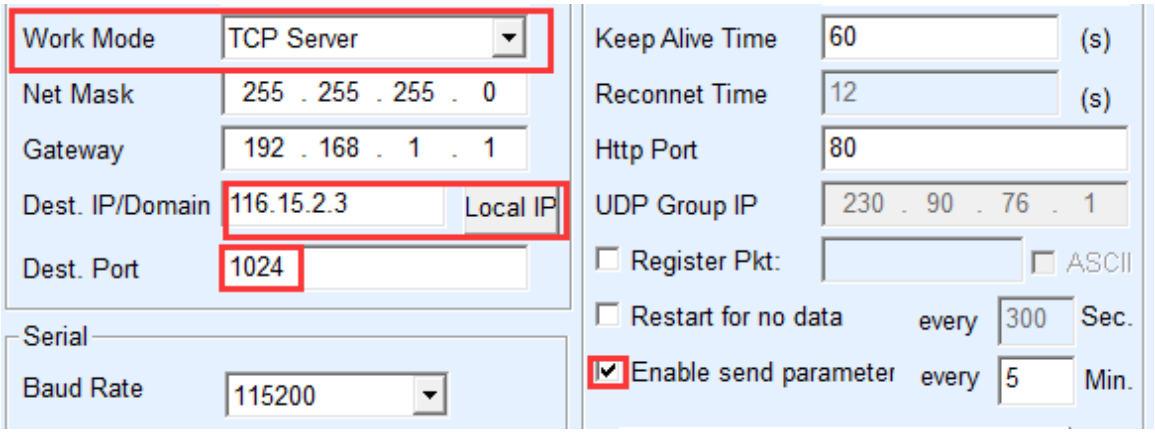


Figure 62 Send Parameter Regularly

To facilitate identification of the device, if remote administration is required, please make the an easy remember name for the device.

16. Firmware Upgrade Method

ZLAN5143BI can upgrade their own programs, but cannot upgrade each other. Whether

device is found in the device list by way of automatic search, manual addition or P2P search, can all use this method to upgrade the firmware.

1) The firmware files of ZLSN2003 are obtained from Z LAN, such as 1.539(2003).BIN.

2) In the ZLVircom tool, first search device needed upgrade, and enter the device parameter edit dialog box. Click "Restart Device" first.

   After the device is restarted, use the same way to search the device, and enter the dialog box again. Click “Load Firmware” button on the down right corner of the dialog box.

Figure 63 Restart Button

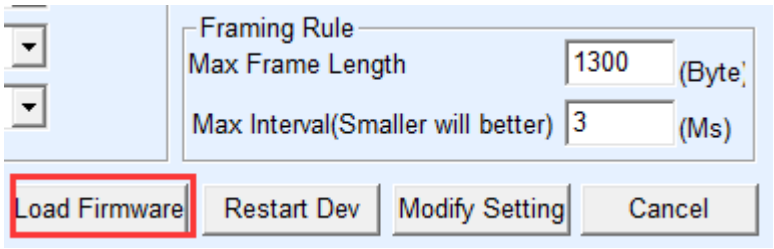


Figure 64 Upgrade Button

3) Select “Code file download mode” option as shown in figure 66. Select the firmware file in the program files. The IP address portion of the serial device server has been automatically filled out, no need to write again, the module type/model has been selected automatically. Then click download.

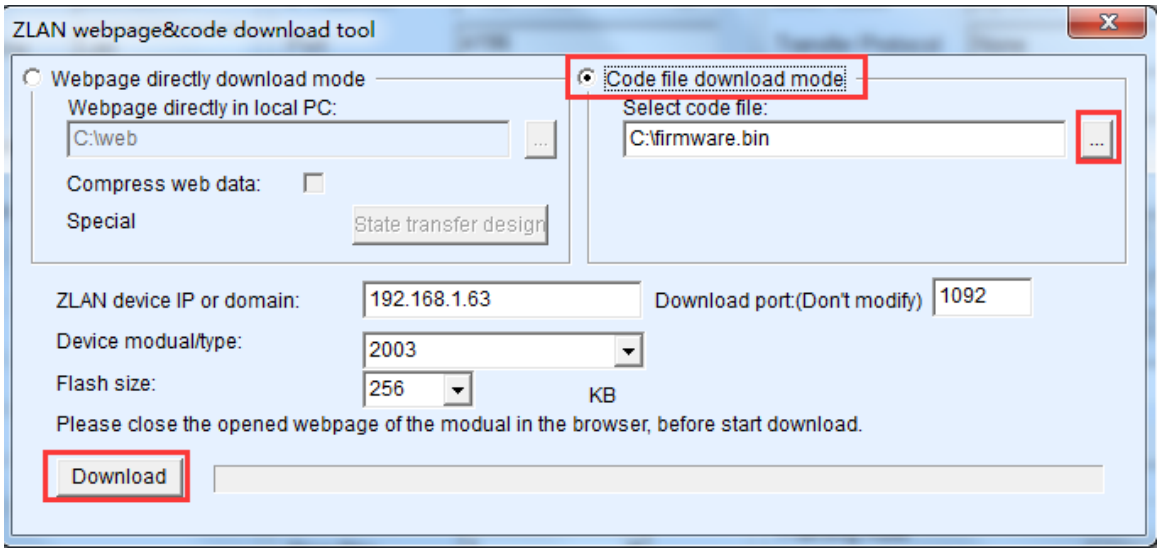


Figure 65 ZLSN2003 Firmware Upgrade Method

4) Here the download progress bar starts to move, the download time is about 30 seconds. During the download, you'll see the ACT light of the device flashes, and at the end of the download, the LINK light flashes a few times. Then the program will pop up the prompt box of "Finish transfer don't power off when the LINK lamp flashing". Note: this is only the transfer finishing, it takes about 3 seconds to write the flash process, and the LINK light will blink. Please do not power off during this period.

5) After the download, generally the program will be restarted automatically, and no need to power off. See the running indicator light flashing, if there is no automatic restart, please repower after the LINK light stop flashing for more than 30 seconds.

6) Web configuration interface update: after firmware upgrade, the internal configuration page of the module also needs to be updated, otherwise it cannot be configured through the Web, but it does not affect the communication. If no need the web configuration you can ignore the webpage download. The way to download the Web is: change the download mode of "code file" to "Webpage directly download mode" as shown in figure 66. And select the root directory that local web page in for the directory (the directory can be gained from ZLAN) of which the webpage needed to download in, click on the download, all the files in local web page directory will be downloaded to the file system inside the device.

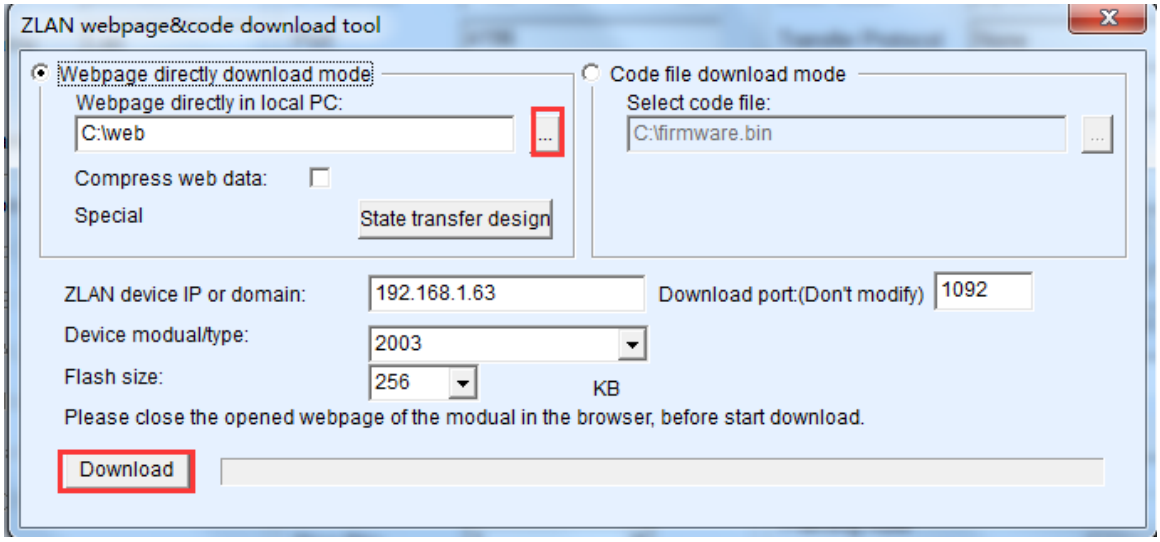


Figure 66 ZLSN2003 Web Upgrade Method

7) Note:

a) If the download fails, the device will not be damaged. Please start downloading again. In addition, when the LINK lamp flickers at the end of the download, please do not power off, otherwise the device will be damaged.

b) Check the firmware version number through ZLVircom to see if the new firmware has been downloaded successfully.

Figure 67 Check Firmware Version after Upgrading

17. Electronic Features

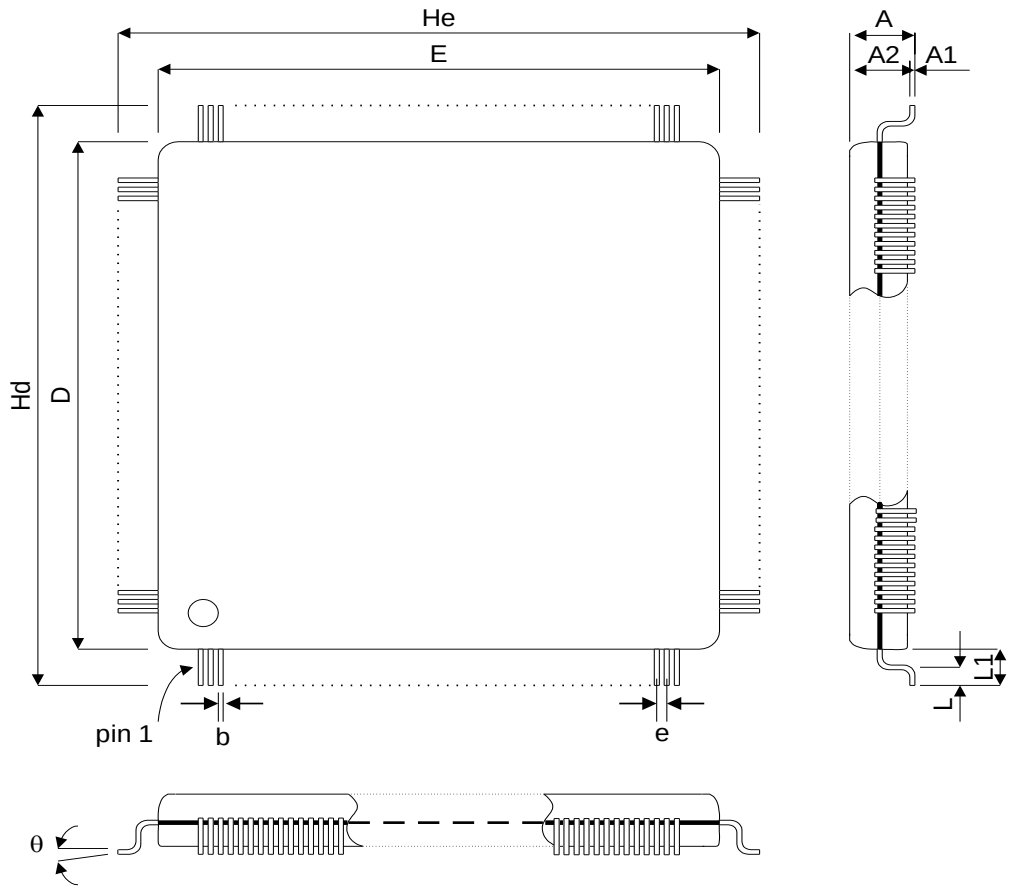
DC Features

Name	Max Range	Suggest operating range	Typical Value	Instruction

VCC1.8V	-0.3～2.16V	0～1.98V	1.8V	
VCC3.3V	-0.3 ~ 3.8V	0 ~ 3.6V	3.3V	
IO voltage	-0.3 ~ 5.8V	0 ~ 5.25V	3.3V	
PIN max current input			20mA	
IO high-level drive current			4mA	
IO low-level drive current			4mA	
Storage temp.	- 40 to 150℃			
3.3V max current			70mA	
1.8V max current			160mA/100mA	SPEED pin configure to high level (High-speed mode) as 160mA; SPEED pin configure to low level (Low-speed mode) as 100mA.
Operating temp.	- 40 to 105℃			
Heat resistance to shell			7.0°C/W	
Heat resistance to environment			45.0°C/W	

18. Encapsulation Size

80-pin LQFP encapsulation



Symbol	Millimeter		
	Min	Typ	Max
A1	0.05	-	0.15
A2	1.35	1.40	1.45
A	-	-	1.60
b	0.17	0.22	0.27
D	12.00 BSC [1]		
E	12.00 BSC		
e	0.50 BSC		
Hd	14.00 BSC		
He	14.00 BSC		
L	0.45	0.60	0.75
L1	1.00 REF		
θ	0°	3.5°	7°