WIZ-IP32 User Manual

V1.0

Update history

Version	Date	Remarks
V1.0	2025/06/10	First release

Copyright Notice

Contact email:supports@wiznet.hk

For more information, please log in:https://www.wizse.com/

Table of Contents

1 Introduction

1.1 Overview

The WIZ-IP32 serial port to Ethernet module supports two data transmission modes: transparent data transmission and AT command. It also supports TCP Server, TCP Client, UDP, and Modbus operating modes. The maximum baud rate of the serial port can reach 1.152Mbps. It is equipped with a matching upper computer configuration software and can also be easily configured via web pages or AT commands.

The WIZ-IP32 serial port to Ethernet module adopts the ARM Cortex-M3 core, making network communication faster, more stable and secure. Users can quickly complete the design of hardware circuits just by following the recommended reference design schematics in the manual, which reduces development difficulty and saves development time.

1.1.1 Functional features

The WIZ-IP32 serial port to Ethernet module features the following main functions and characteristics:

- Supports Modbus RTU/ASCII to Modbus TCP/UDP and Modbus over TCP/UDP conversion
- Supports two data transmission modes: data transparent transmission and AT command
- The baud rate setting range is 16 commonly used baud rate values between 1.2Kbps and 1.152Mbps
- Supports TCP Server, TCP Client, UDP, and Modbus working modes
- Features up to 2048-byte dual serial port receive buffer and 2048-byte network port receive buffer
- Integrates a full hardware TCP/IP protocol stack to ensure fast, secure, and stable data communication
- Flexible dual serial port data packet setting to meet diverse packetization requirements of users
- Supports Keep Alive function to ensure real-time network link availability
- Supports DHCP for automatic IP address acquisition
- Supports DNS function to meet users' needs for module-server communication via domain names
- Supports NetBIOS function for easy module access by name
- Supports connection password verification to enhance communication security
- Supports configuration via dual serial port AT command mode, web page, and upper computer tools
- Supports local firmware upgrade via upper computer and web page

1.1.2 Product features

- 32-bit ARM Cortex-M3 MCU
- Ethernet
  - 10/100M adaptive Ethernet
- Serial port
  - 3.3V TTL×2:TXD, RXD, GND
- Serial communication parameters
  - Baud rate:16 commonly used baud rate values ranging from 1.2Kbps to 1.152Mbps
  - Data bits:7, 8
  - Stop bits:0.5, 1, 1.5, 2
  - Parity: None, Even, Odd
  - Flow control: None
- Input power supply
  - WIZ-IP32:DC 3.3V
- Dimensions (Length×Width×Height)
  - WIZ-IP32:32.50×16.50×13.70 (mm)
- Operating temperature
  - WIZ-IP32: -40℃ ~ +85℃
- Storage temperature
  - WIZ-IP32: -40℃ ~ +95℃, 5 ~ 95% RH

1.1.3 Parameter configuration method

WIZ-IP32 dual serial port to Ethernet modules provide three commonly used parameter configuration methods for users to choose from:

- Serial port AT command configuration, users can integrate WIZ-IP32 into their embedded products, and the main MCU can configure parameters by sending serial port AT commands. Users can also directly configure modules through AT commands using the upper computer serial port tool, as detailed in Chapter 7 AT Commands.
- Web browser configuration, users can configure through a web browser on a computer within the same LAN as the module, as detailed in Chapter 8 Web Page Configuration.

- WIZS2E ConfigTool configuration: Users can configure using the WIZS2E ConfigTool on a computer within the same local area network as the module. For details, please refer to Chapter 6 WIZS2E ConfigTool Software Configuration.

1.2 Product specifications

1.2.1 Electrical characteristics

Unless otherwise specified, the following parameters refer to the values at Temp=25℃.

Voltage and current characteristics

Table 1-1 WIZ-IP32 Electrical characteristics

Symbol	Types	Ratings			
		Min	Typical	Max	Unit
V_{IN}	Module voltage	3.0	3.3	3.6	V
I_{IN}	Module current	92.8	102.5	155.0	mA

Current characteristics

Table 1-2 WIZ-IP32 Current characteristics

Module status	Test value(mA)	Module status	Test value(mA)
Standby	92.8	Communication	155.0

1.2.2 Dimensions

If users need to integrate the WIZ-IP32 serial port to Ethernet module into their mainboards, they can refer to the following mechanical dimensions. Unit: mm.

Dimension drawing of WIZ-IP32

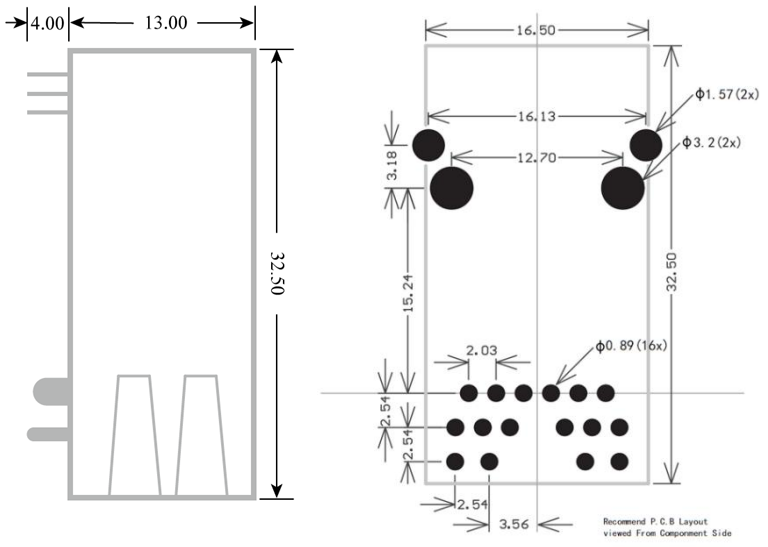


Figure 1-1 Mechanical dimension diagram of WIZ-IP32

1.2.3 Temperature characteristics

Table 1-3 Temperature characteristics

Name	level	Operating temperature	Storage temperature
WIZ-IP32	Industrial grade	-40℃ ～+85℃	-40℃ ～+85℃

2 Hardware description

2.1 Interface description

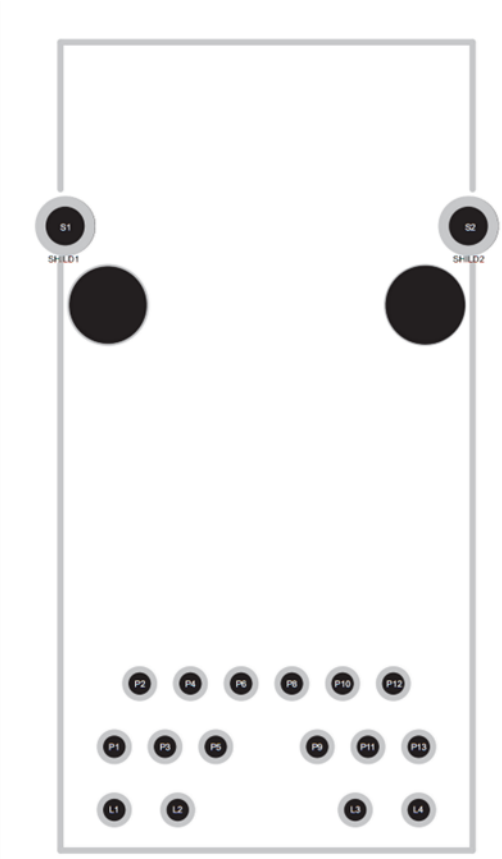


Figure 2-1 WIZ-IP32 pinout diagram

Table 2-1 WIZ-IP32 pin description

Pin Number	Pin Name	Pin Type	Description
P1	ACT	RJ45_LED Control Pin	RJ45 Interface data status LED control
P2	TXD	Serial Port 1	3.3V TTL serial port transmit pin.
P3	RXD	Serial Port 2	3.3V TTL serial port receive pin.
P4	RXD	Serial Port 1	3.3V TTL serial port receive pin.
P5	TXD	Serial Port 2	3.3V TTL serial port transmit pin.
P6	NC	-	-
P8	RESET	Reset Pin	The entire module is reset when a low - level signal is received.
P9	GND	Power Ground	module power ground
P10	NC	-	-
P11	VCC	Positive Power Supply Pin	module positive power supply. Default: DC 3.3V.
P12	NC	-	-
P13	LINK	RJ45_LED Control Pin	RJ45 Interface connection status LED control.
L1	D1-	RJ45_LED	The negative pole of the RJ45 connection status LED is connected to the ACT pin in default.
L2	D1+	RJ45_LED	The positive pole of the RJ45 connection status LED is connected to 3.3V power supply in default.

L3	D2+	RJ45_LED	The positive pole of the RJ45 data status LED is connected to the 3.3V power supply in default.
L4	D2-	RJ45_LED	The negative pole of the RJ45 data status LED is connected to the LINK pin in default.

2.2 Introduction to the Evaluation Board

The WIZ-IP-EVB evaluation board enables users to test and evaluate the serial-to-Ethernet modules of the WIZ-IP32, WIZ-IP75 and WIZ-IP20 models. The interface diagram of the WIZ-IP-EVB is shown in Figure 2-2.

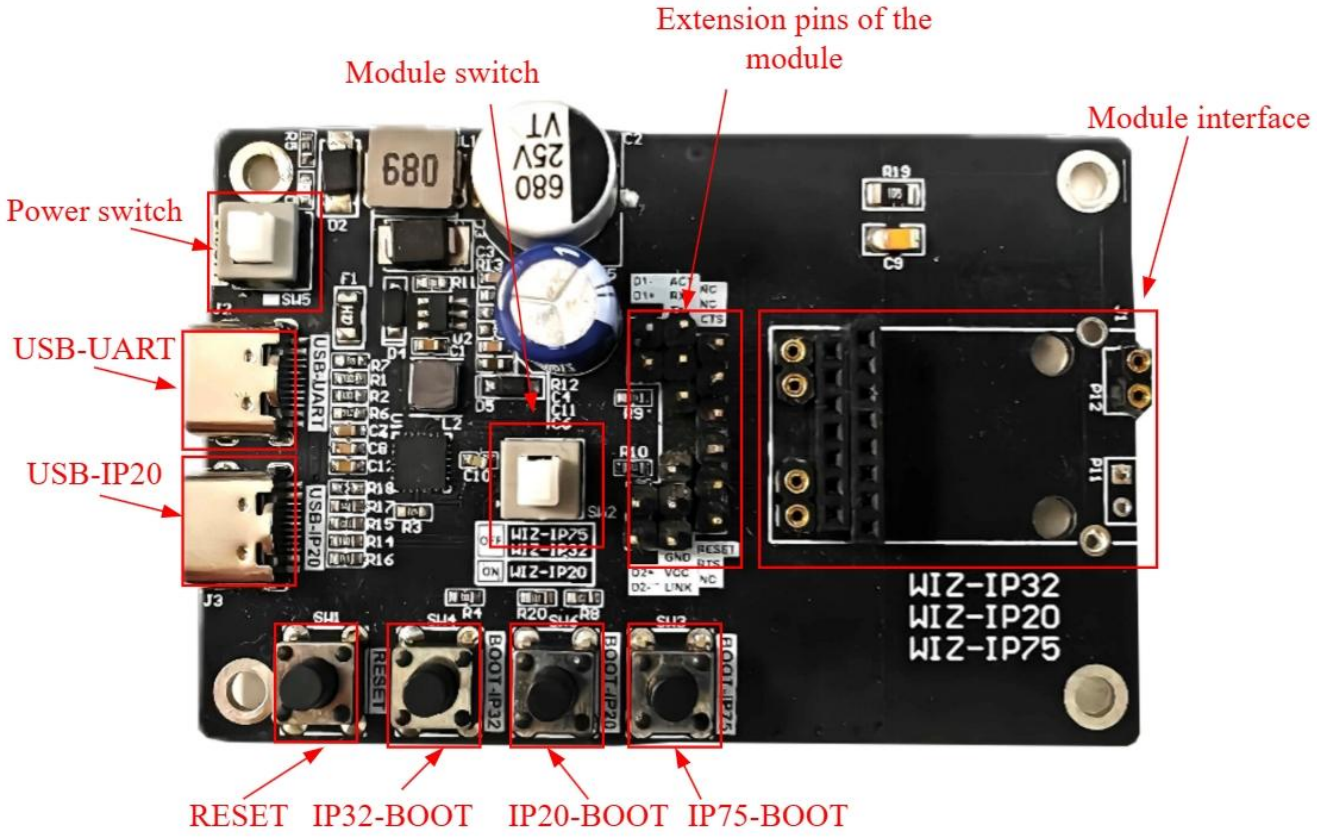


Figure 2-2 WIZ-IP-EVB Evaluation Board Schematic Diagram

- **Power Switch:** Used to control the power on/off of the entire circuit board, determining whether to supply power to the circuit board.
- **USB-UART:** USB to UART interface, providing 5V power supply to the circuit board and modules, enabling data conversion between USB and serial communication interfaces, used for AT command configuration and serial data transmission.
- **USB-IP20:** WIZ-IP20 firmware download serial USB port
- **RESET:** Reset button, when pressed, can make the circuit board or related functional modules restart and return to the initial state.

- **Module Switch:** When the button is pressed, it is configured to use the WIZ-IP20 mode; when released, it is configured to use the WIZ-IP75 or WIZ-IP32 mode.
- **IP32-BOOT:** Controls the startup mode of the hardware WIZ-IP32, used to force the module to enter the boot mode for firmware upgrade, debugging or fault recovery.
- **IP75-BOOT:** Controls the startup mode of the hardware WIZ-IP75, used to force the module to enter the boot mode for firmware upgrade, debugging or fault recovery.
- **IP20-BOOT:** Controls the startup mode of the hardware WIZ-IP20, used to force the module to enter the boot mode for firmware upgrade, debugging or fault recovery.

2.3 WIZ-IP32-EVB schematic diagram reference

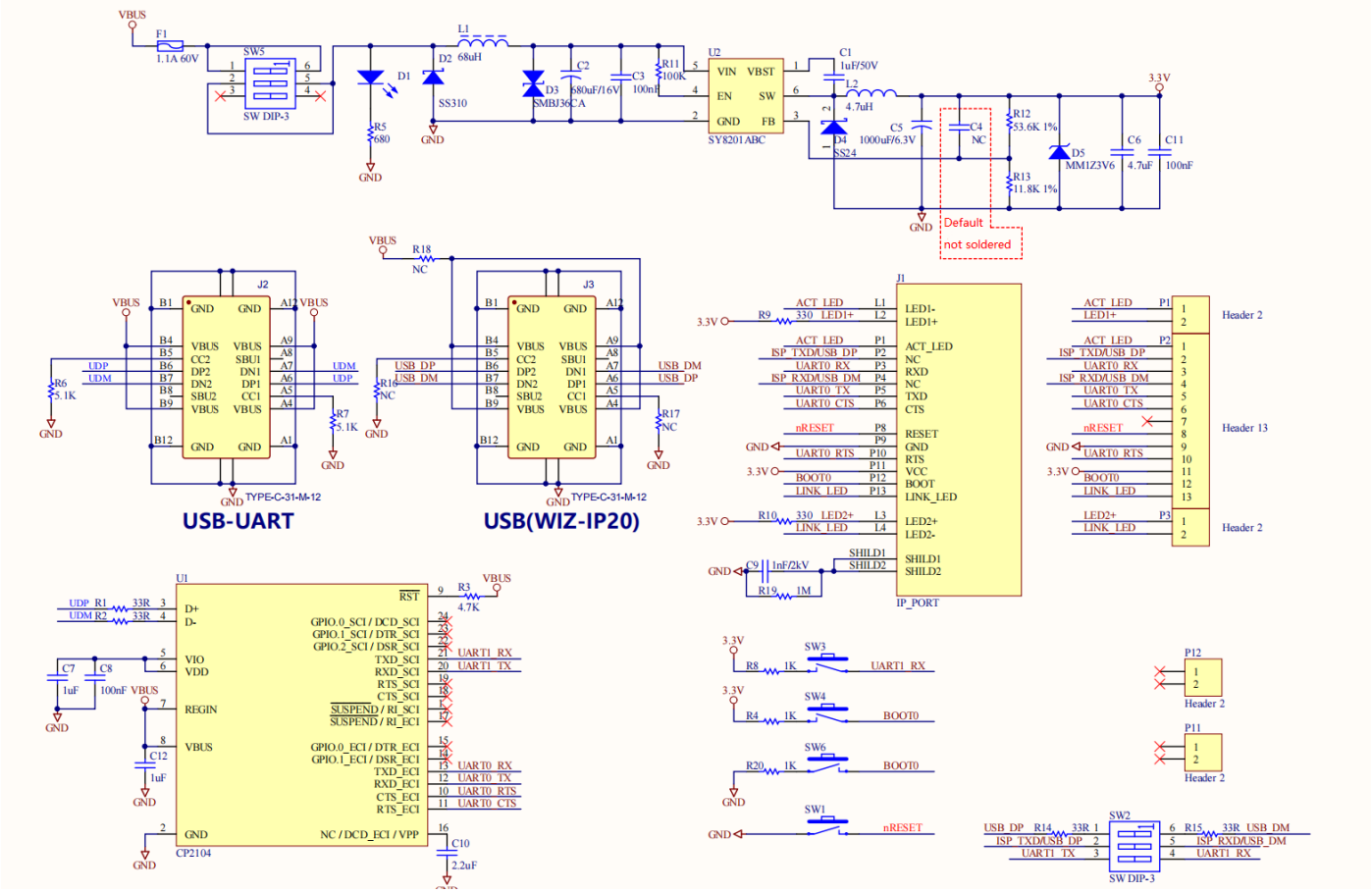


Figure 2-2 WIZ-IP-EVB schematic diagram

2.3 Quick evaluation wiring instructions

By integrating the WIZ-IP32 serial to Ethernet module, users can quickly upgrade the serial port of their modules to an Ethernet interface. Before integration, it is recommended that users test and evaluate the WIZ-IP32 serial-to-Ethernet module using the WIZ-IP32-EVB evaluation board.

As shown in the figure below, to test the data conversion from the serial port to Ethernet, it is necessary

to install the WIZ-IP32 on the WIZ-IP32-EVB first. Then, connect the serial port of the WIZ-IP32-EVB to the computer's USB port using a USB-to-serial adapter, and connect the network port of the WIZ-IP32-EVB to network modules such as routers or switches. At the same time, the user's computer should also be connected to this local area network. In this way, a simple network environment between the computer and the serial port module is set up. Then, as described in the subsequent chapters, the user's serial port module can realize data communication with the computer through Ethernet, and the WIZ-IP32 acts as a bridge for conversion between the serial port protocol and the Ethernet protocol.

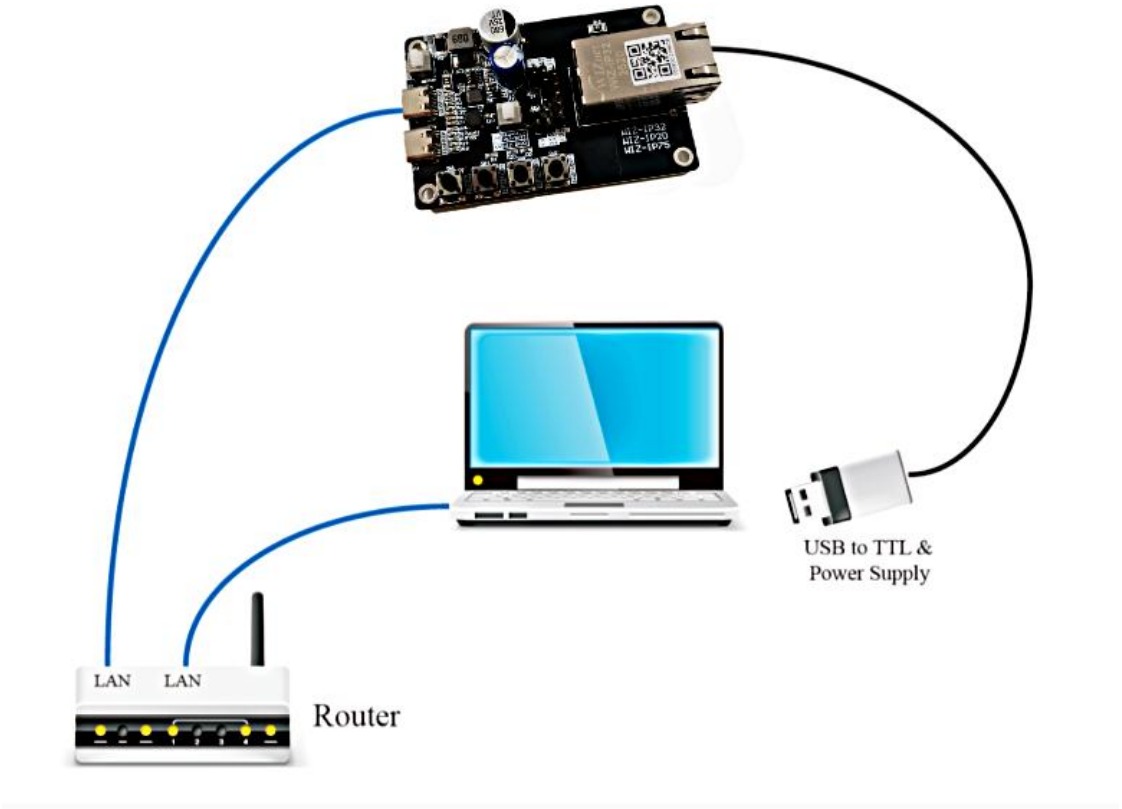


Figure 2-3 WIZ-IP32 wiring diagram

3 Operating modes

WIZ-IP32 serial to Ethernet modules support TCP Server, TCP Client, UDP, and Modbus working modes, which will be briefly explained below.

3.1 TCP server mode

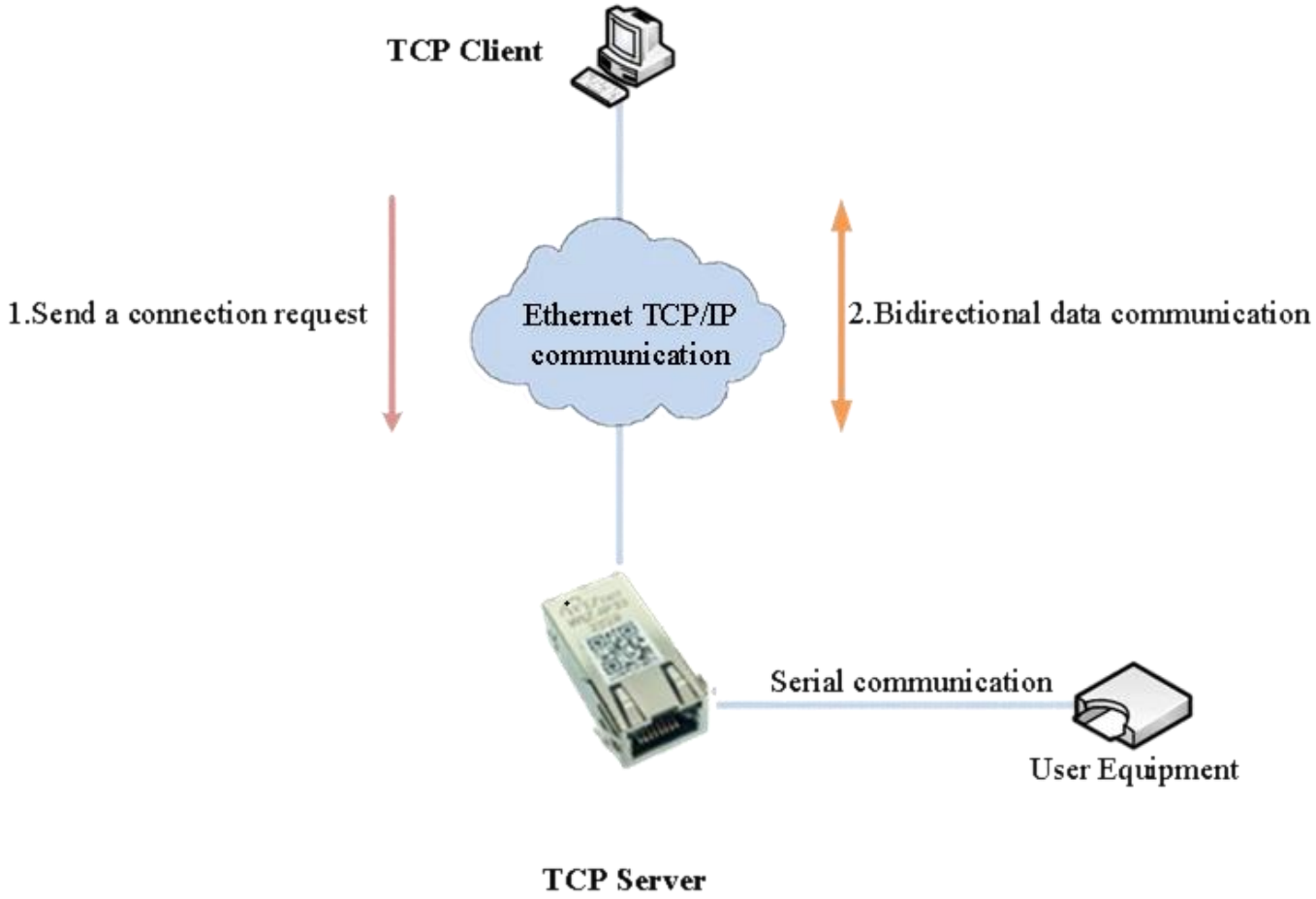


Figure 3-1 TCP Server Mode Schematic

As shown in Figure 3-1, in TCP Server mode, the WIZ-IP32 opens a local port for listening, with the default port number being 5000, waiting for connections from clients. Once a TCP connection is established with a client, bidirectional data communication can be carried out.

3.2 TCP client mode

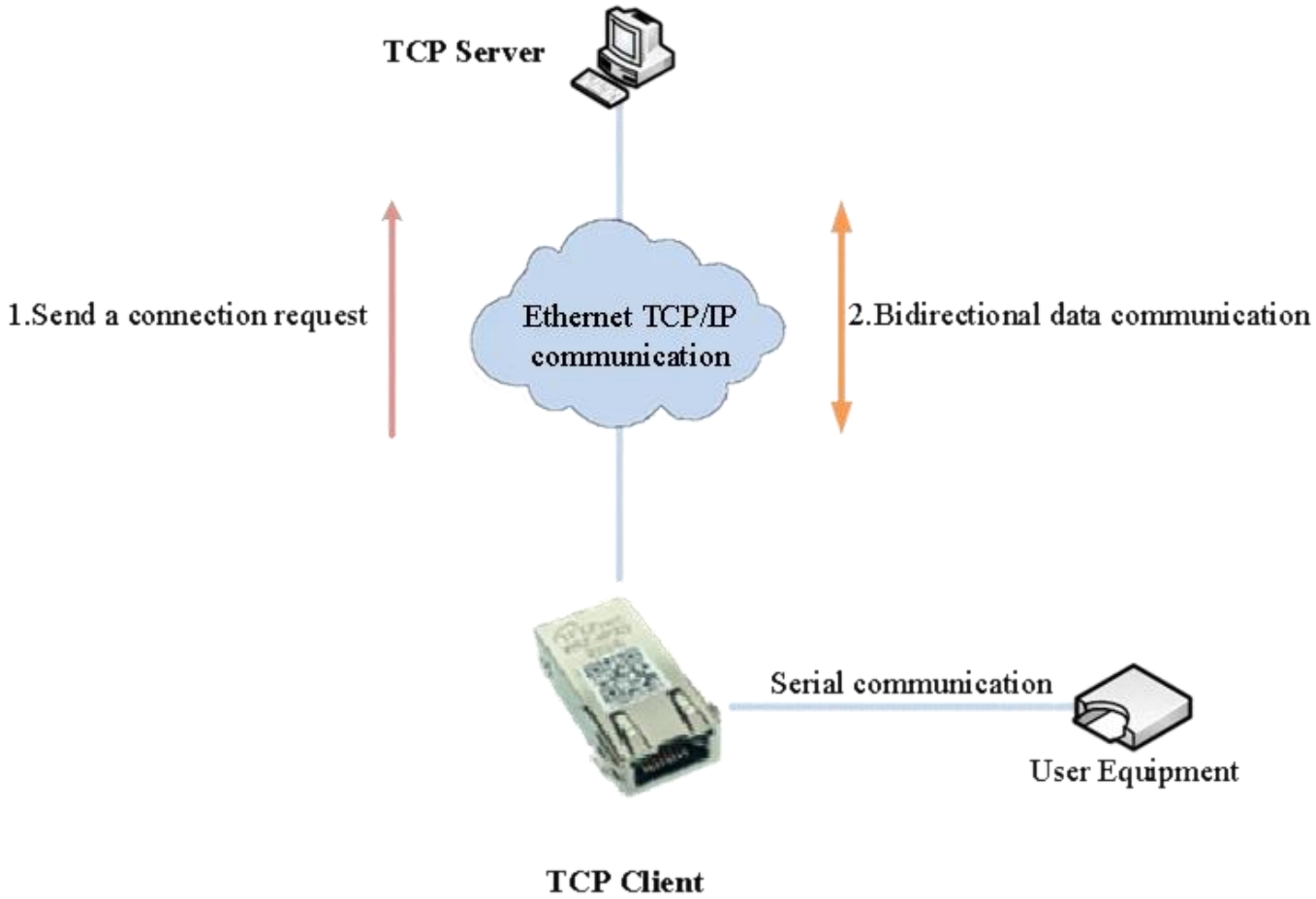


Figure 3-1 Schematic diagram of TCP client mode

As shown in Figure 3-2, in TCP Client mode, WIZ-IP32 will actively initiate connection requests to pre-set TCP servers. After establishing a TCP connection, bidirectional data communication can be carried out.

3.3 UDP mode

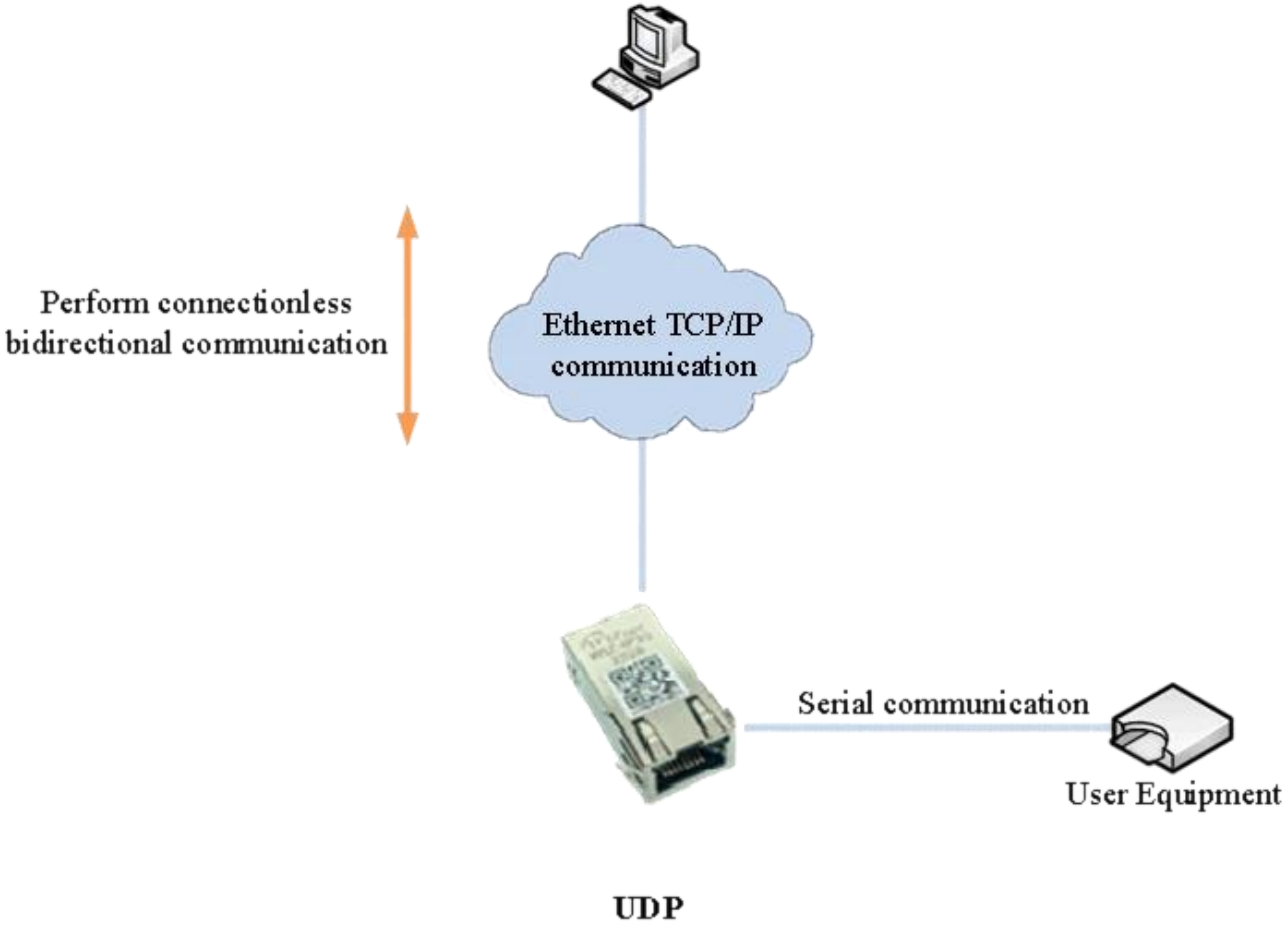


Figure 3-2 UDP mode diagram

As shown in Figure 3-3, the UDP mode does not require connection establishment. UDP communication can be realized by setting the IP addresses and port numbers of the WIZ-IP32 and the remote module.

3.4 Modbus mode

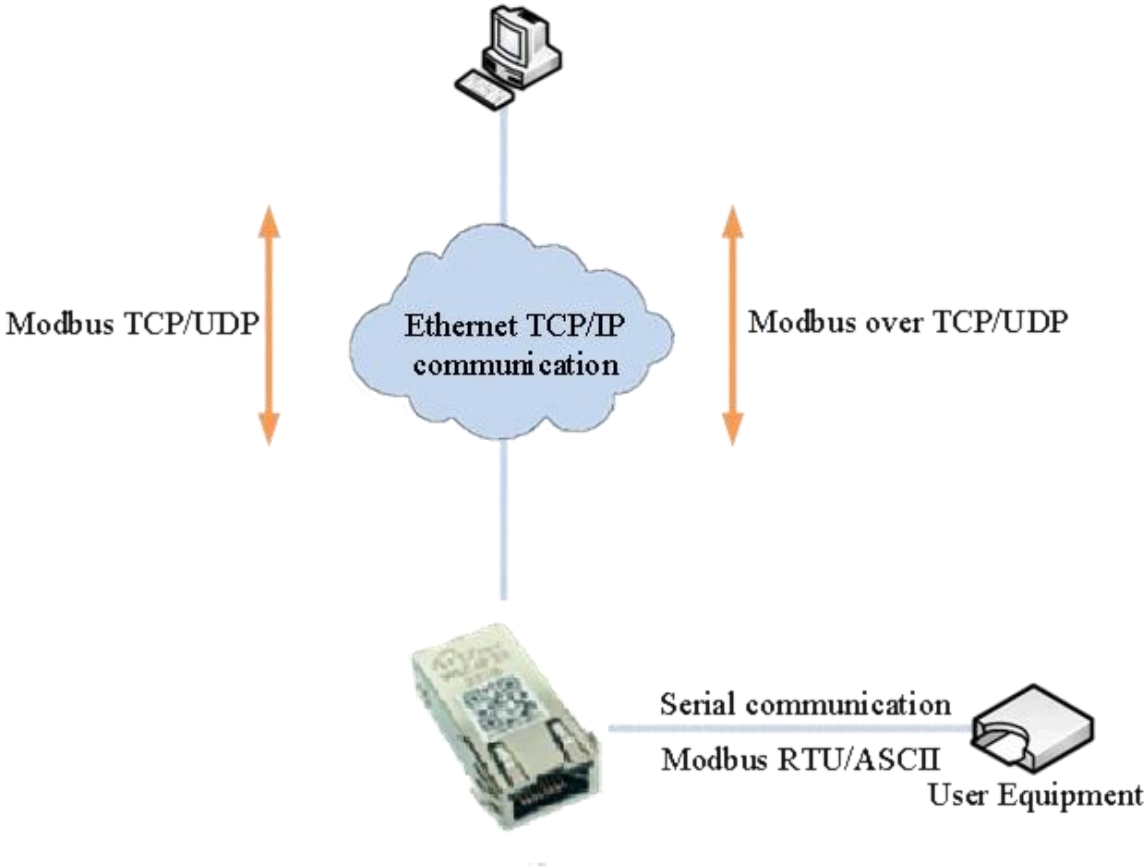


Figure 3-3 Modbus mode diagram

As shown in Figure 3-4, in Modbus mode, the WIZ-IP32 can realize the mutual conversion between Modbus RTU/ASCII protocols and Modbus TCP/UDP protocols. Meanwhile, the WIZ-IP32 also supports Modbus over TCP/UDP transmission.

Note: The Modbus function of the WIZ-IP32 is only available in Data Pass-Through mode. It is not supported when transmitting data via AT commands.

4 Data transfer mode

WIZ-IP32 serial to Ethernet modules support two data transmission modes for users to choose from: data transparent transmission mode and AT command transmission data mode.

4.1 Data pass-through mode

The data transmission mode of WIZ-IP32 has the following characteristics:

1 When using an MCU to configure the WIZ-IP32 for data transmission and reception, in general, before entering data transmission mode, it is necessary to configure parameters in AT command mode and then switch to data transmission mode;

2 After entering data transmission mode, the module will operate according to its configured working mode:

2.1 When the module is in TCP Server mode, it will immediately establish listening until a client establishes a connection with it;

2.2 When the module is in TCP Client mode, it will immediately request a connection from the server until a successful connection is established;

2.3 When the module is in UDP mode, there is no need to establish a connection and wait for user data;

3 After establishing a TCP or UDP connection, there is no need for any AT command operation, and the module can automatically send and receive data at any time, achieving transparent transmission of data;

4 If the TCP connection is disconnected normally, the module will perform the following actions:

4.1 When the module is in TCP Server mode, it will re-establish listening;

4.2 When the module is in TCP Client mode, it will reapply for a connection to the server;

5 When configuring WIZ-IP32 to send and receive data using MCU, if you need to modify the configured parameters, you need to exit the data transmission mode and enter the AT command mode for parameter configuration.

4.2 AT data transfer command mode

The AT command transmission data mode of WIZ-IP32 has the following characteristics:

1 Configuration is done in AT command mode, and data transmission is also performed in AT command mode without switching operating modes;

2 After configuring parameters in AT command mode, relevant AT data transmission commands can be directly called for TCP or UDP communication;

2.1 When configuring the module to TCP Server mode, the module will immediately establish listening until a client establishes a connection with it;

2.2 When configuring the module to TCP Client mode, the module will immediately apply to the server for a connection. If the connection cannot be successfully established within 3 seconds, a new command needs to be sent to apply for a connection;

2.3 When configuring the module in UDP mode, the module does not need to establish a connection and waits for user data;

3 After establishing a TCP connection or UDP, sending or receiving data requires corresponding AT commands for operation, and only one data packet can be sent/received at a time;

4 If a TCP connection is disconnected, the relevant AT data transmission command must be called again to re-establish the connection;

5 If you need to modify the configuration parameters, you can directly call the AT command for configuration.

4.3 How to enter "Data pass-through mode"

Users can enter data transmission mode through three ways:

1 AT command (see Chapter 7 "RESET" and "EXIT" commands for details);

2 Web pages (see "Start Mode" option in Chapter 8 "Basic Settings" tab for details);

3 WIZS2E Config Tool configuration tool (see "Enter... Mode when power on or reset" option in Chapter 6 "Additional Functions" for details).

5 WIZ-IP32's IP address

Before using WIZ-IP32, we need to know its IP address and other parameters. The WIZ-IP32 supports two IP address assignment methods: Static IP and DHCP (Dynamic Host Configuration Protocol). Static IP requires the user to manually configure parameters such as the IP address, subnet mask, and gateway. It should be emphasized that the module IP address cannot be the same as the IP address of other modules in the same local area network, otherwise communication cannot be achieved; DHCP allows the module to automatically obtain an IP address, subnet mask, and gateway from a DHCP server on the network.

5.1 Factory setting of module IP address

The default IP address for WIZ-IP32 serial to Ethernet modules at the factory is 192.168.1.88.

5.2 Obtain module IP information

WIZ-IP32 has two methods to obtain module IP addresses:

1 WIZS2E Config Tool upper computer software, refer to the instructions in Chapter 6.1 for details;

2 AT command query, please refer to Chapter 7 "IP" query command for details

5.3 Method for determining if modules and computers are on the same network segment

Before using a computer to communicate with WIZ-IP32, users need to ensure that their computer is on the same network segment as WIZ-IP32. WIZ-IP32 is set with a default IP address (192.168.1.88) and network mask (255.255.255.0) at the factory. Users can check whether the module is on the same subnet as their computer according to the process shown in the following figure.

If in the same subnet, WIZ-IP32 modules can be directly used for communication. If they are not on the same subnet, it is necessary to set the IP address of the computer.

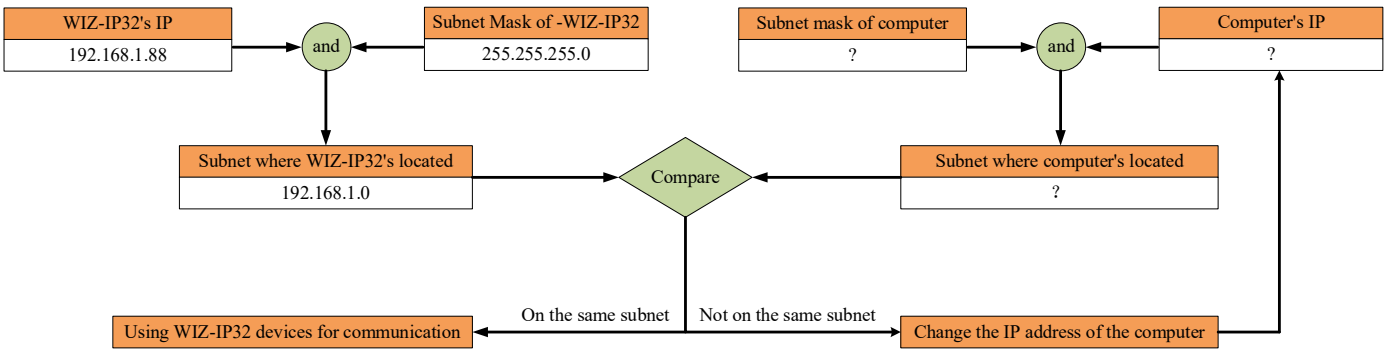


Figure 5-1 WIZ-IP32 and computer IP address same subnet detection

5.4 Computer IP setting method

Taking the Windows 10 operating system as an example for explanation.

Start → Control Panel → Network Sharing Center → Change Adapter Settings → Local Area Connection → Right click Properties → Double click Internet Protocol Version 4 (TCP/IPv4) to obtain the page shown in the following figure. Select 'Use the following IP address' and fill in the IP address, such as 192.168.1.99, subnet mask 255.255.255.0, default gateway 192.168.1.1, DNS section can be left blank, click' OK 'to complete the configuration. You can now communicate with WIZ-IP32 modules.

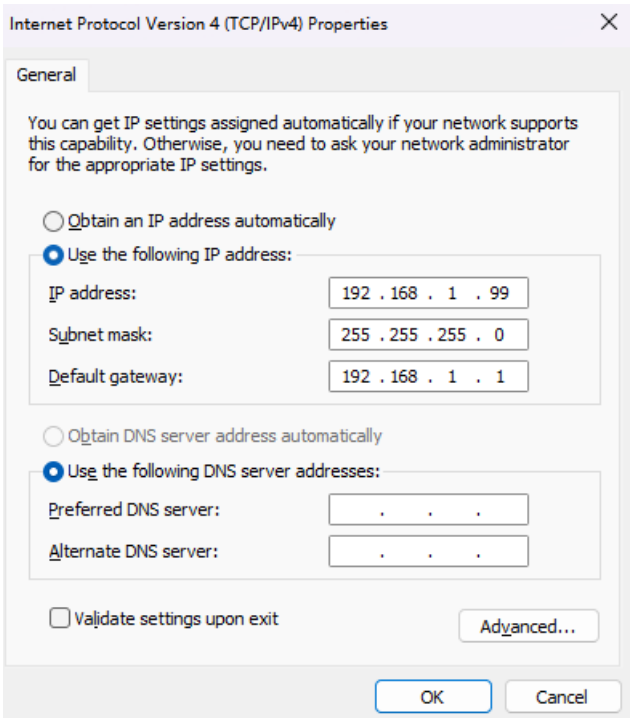


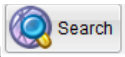
Figure 5-2 Windows 10 IP address setting interface

6 WIZS2E Config Tool software configuration

The WIZS2E Config Tool is a PC-based configuration application that runs on the Windows operating system and is compatible with WIZ-IP32, WIZ-DTU, W7500S2E and W5500S2E series serial port to Ethernet modules. Users can easily search, view, and configure various functions and information of the WIZ-IP32 serial port to Ethernet module through WIZS2E ConfigTool.

Note: It is recommended to turn off the firewall before configuration and ensure that the IP addresses of all modules in the local area network do not conflict.

6.1 Obtain module configuration information

Click the button 

in the toolbar, and the left search window will categorize by module MAC address, listing all searched WIZ-IP32s. Click on the "⊞" in front of the list to expand and view detailed information about the module.

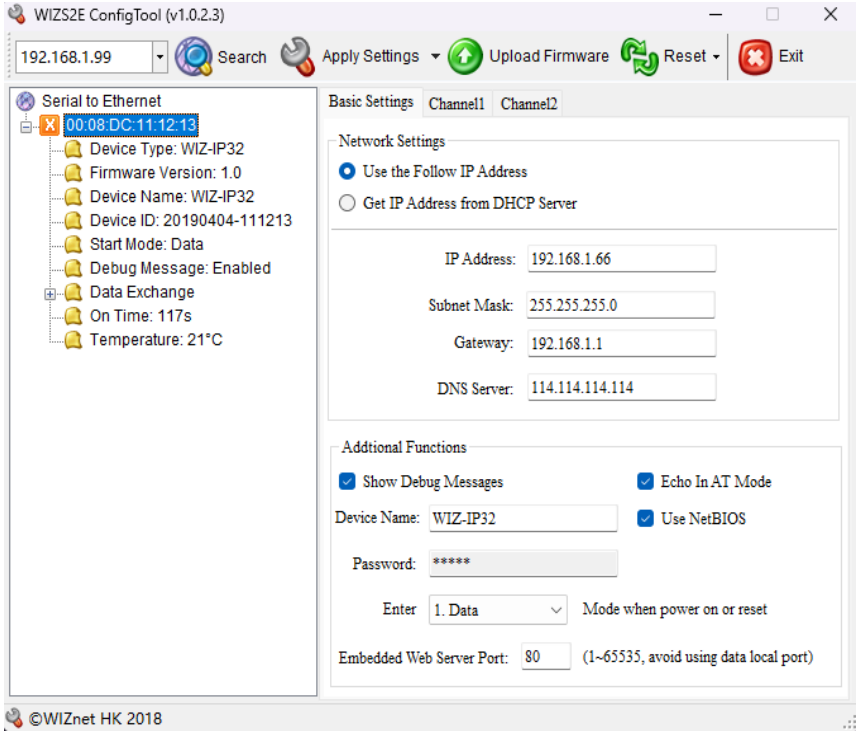


Figure 6-1 Basic configuration interface of WIZS2E Config Tool

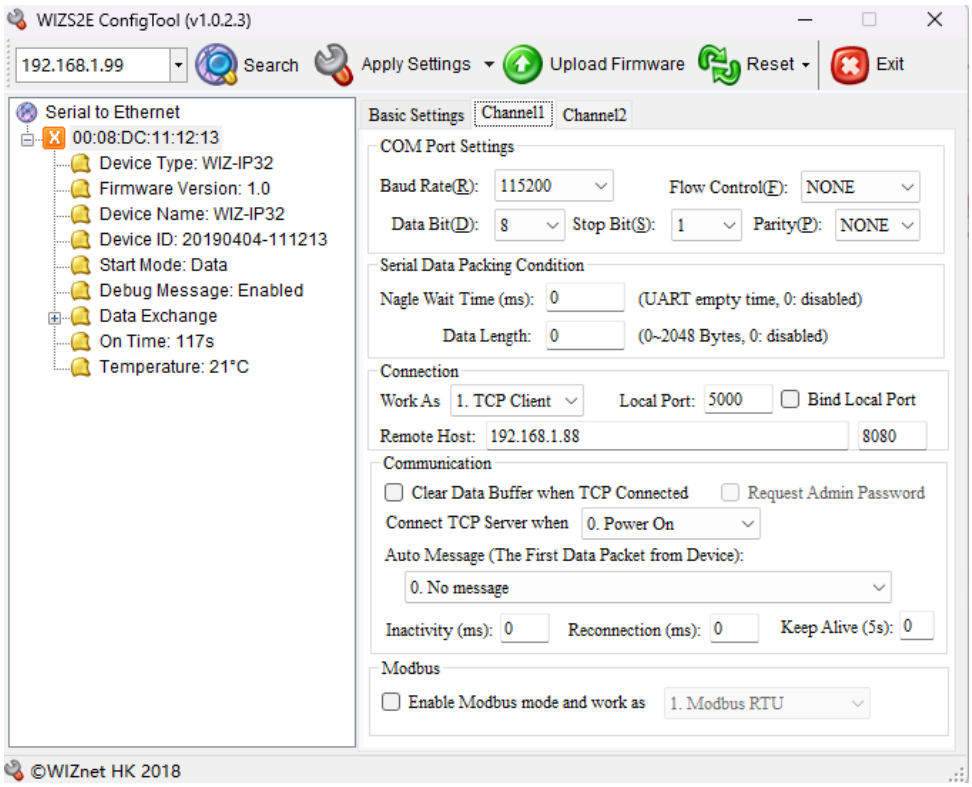


Figure 6-2 Channel1 interface

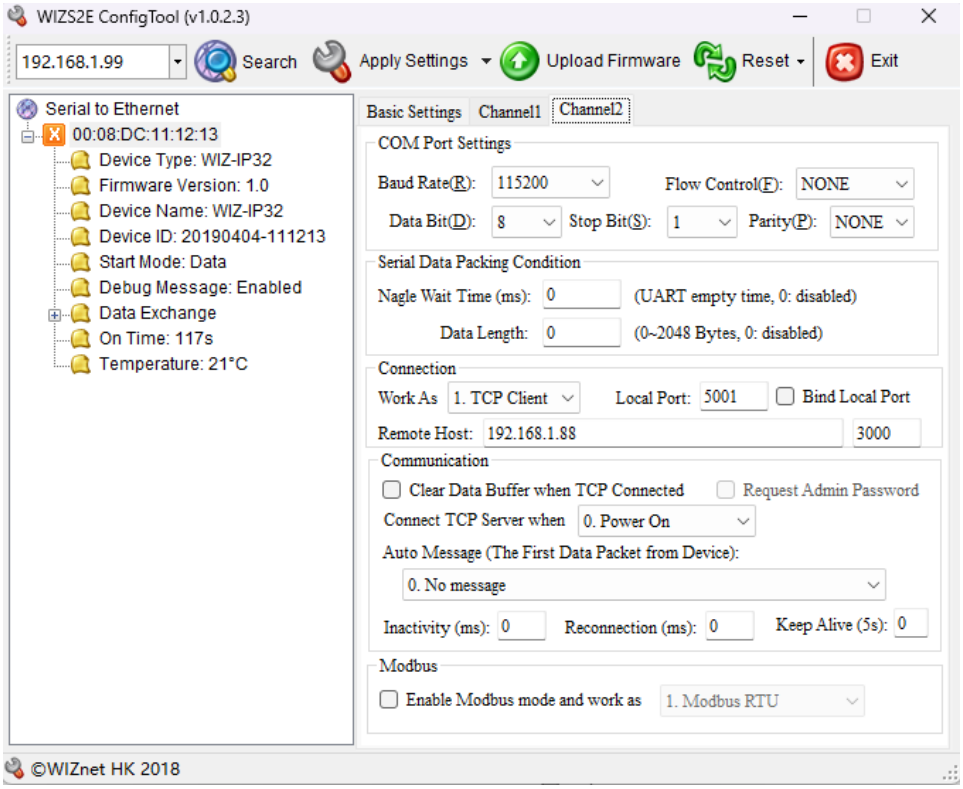
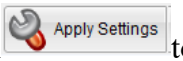


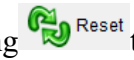
Figure 6-3 Channel2 interface

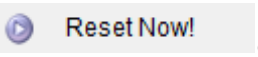
6.2 Modify module configuration information

After searching for the module, users can directly modify the module information in the basic and advanced configuration interfaces, and click the button 

to save the configuration information.

6.3 Reset module

Clicking 

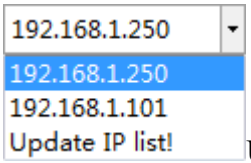
the button will bring up a dropdown menu, and clicking 

on it will immediately restart the module (without saving the configuration).

6.4 Practical functions

6.4.1 Switch network card

To resolve the issue of being unable to determine which network interface (wired or wireless) to use when configuring the module from a laptop, this tool has added a function to switch network cards,



Users can first click "Update IP list!" to update the network card list, and then select the corresponding network card for configuration according to the actual situation.

6.4.2 Right mouse button

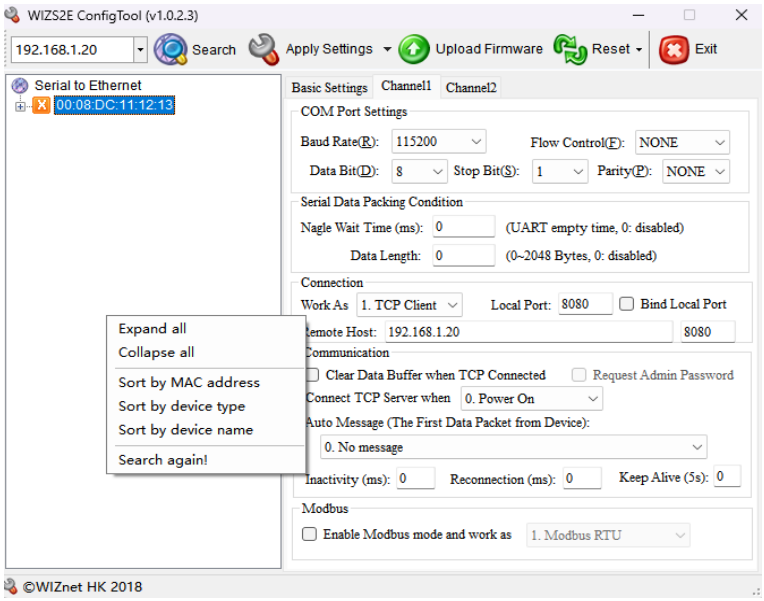


Figure 6-4 Right mouse button function

Right click on the module list on the left, as shown in the figure below, and a list of three functions will appear:

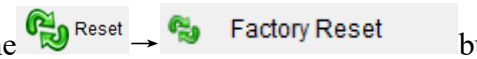
1. Expand/collapse all module details;
2. The searched modules can be sorted by MAC address, module type, or module name;

3. Search again! This function allows users to keep the original module list unchanged when configuring modules in batches, and the newly searched module information will be added after the original module list.

6.5 Factory reset

If users need to restore the WIZ-IP32 serial to Ethernet module to its factory settings, there are three methods: software mode, AT command mode and web page.

6.5.1 Restore software to factory settings

Firstly, click the left mouse button in the module list of WIZS2E Config Tool software to select the module that needs to be restored, and then click the 

button in the toolbar to factory Reset.

6.5.2 Restore factory settings through AT command

Refer to Chapter 7 for details on the "DEFAULT" command.

6.5.3 Web page format

Please refer to Chapter 8.5 for an introduction to the module management interface.

7 Introduction of AT commands

7.1 AT command overview

The AT commands supported by the WIZ-IP32 serial-to-Ethernet modules use a standard interface. They are not case-sensitive and always start with "AT" and end with "\r\n". The format of its commands, return values, and parameters is fixed. Generally speaking, AT commands have the following formats:

1、.Commands without parameter

Format: AT+<command>\r\n

Explanation: No extra parameters or symbol after the command.

Example:

Command: *AT+EXIT\r\n*

Response: *OK\r\n*

2、Read commands

Format: AT＋<command>?\r\n

Description: These commands read the current settings value.

Example:

Command: *AT+ECHO?\r\n*

Response: *[ECHO] Value is: 1\r\nOK\r\n*

3、Command with parameter

Format: AT＋<command>=<parameter>\r\n

Explanation: These commands are for configure certain settings. (except for the DEFAULT command and RESET).

Example:

Command: *AT+ECHO=1\r\n*

Response: *[ECHO] Value is: 1\r\n OK\r\n*

7.2 AT command responds

According to the different AT commands input by the user, WIZ-IP32 will reply with the execution results of the commands, as shown in Table 7-1.

Table 7-1 AT command responds list

Response type	Response	Description
Error message	Command Invalid\r\nERROR\r\n	Invalid command
	<Error Info>\r\n	Invalid parameter or in the wrong mode
Success message	OK\r\n	Command is executed successfully
	[Command] Value is: <value>\r\nOK\r\n	

7.3 Entering AT command Mode.

There are two operating modes for WIZ-IP32 serial to Ethernet modules:AT command mode and data transmission mode. In AT command mode, users can use serial tools or configure various parameters of the module through their MCU, and also support data transmission under AT commands.

Note:If a TCP connection is established before entering AT command mode, the connection will be closed when entering AT command mode.

When WIZ-IP32 is in AT command mode, input the terminal detection command "AT\r\n" to its serial port. If the module correctly receives "AT\r\n", it will reply with "OK\r\n".

When WIZ-IP32 is in data transmission mode, any AT command input to the serial port is invalid. At this point, input "+++" to the serial port, and WIZ-IP32 will switch to AT command mode.

Input rule for "+++":Three "+" must be sent continuously to the serial port at once, and there must be no other characters within 1 second before and after the "+++" in order for WIZ-IP32 to respond correctly and switch to AT command mode.

Note:The default operating mode of WIZ-IP32 at the factory is AT command mode.

7.4 AT command list

WIZ-IP32 serial to Ethernet modules support serial AT command configuration parameters and also support data transmission in AT command mode. Therefore, AT commands can be roughly divided into two categories:AT configuration commands and AT data transmission commands.

7.4.1 AT setting command list

The AT configuration command is used to configure or query various parameters of WIZ-IP32. Specifically divided into control commands, module information configuration commands, serial port information configuration commands, and management commands.

This module provides dual channel operation capability. In subsequent instructions, the [CH] parameter is used to specify the channel number, with a valid range of 1 to 2. Users can select the corresponding channel

for configuration according to their needs. As shown in the table below.(R:searchable; W:Configurable; RW:searchable and configurable)

Table 7-2 Configure command list

Type	Command name	Function	Attribute	Max length	Parameters
Control command	AT	Terminal check	R	-	-
	ECHO	Enable or disable echoing	R/W	1	0:Echo off 1:Echo on (default)
	DEBUGMSGEN	Debug messag	R/W	1	0:Disable 1:Enable (default)
	NAME	Module name	R/W	15	Must be numbers, alphabets or the combination of both
	PASS	Module password	R/W	15	Must be numbers, alphabets or the combination of both (Default:admin)
	DEFAULT	Reset to factory default	W	15	Module password
	RESET	Save and restart module	W	15	Reset if parameter equals to password
	EXIT	Exit AT command	W	-	-
	SAVE	Save settings	W	-	-
Module information configuration command	START_MODE	Start mode	R/W	1	0:AT command mode (default) 1:Data pass-through mode
	C[CH]_OP	Operating mode	R/W	2	0:TCP Server(default) 1:TCP Client 2:UDP 16:Modbus RTU-TCP Server 17:Modbus RTU-TCP Client 18:Modbus RTU-UDP 32:Modbus ASCII-TCP Server 33:Modbus ASCII-TCP Client 34:Modbus ASCII-UDP
	IP_MODE	IP configuration method	R/W	1	0:Static IP mode (default) 1:DHCP mode
	IP	IP address	R/W	15	Default :192.168.1.88
	MASK	Subnet mask	R/W	15	Default :255.255.255.0
	GATEWAY	Gateway	R/W	15	Default :192.168.1.1
	DNS	DNS server address	R/W	15	Default :114.114.114.114
	C[CH]_PORT	Local port number	R/W	5	1 ~ 65535;Default:5000
	C[CH]_BIND	Local port binding	R/W	1	Valid only in TCP Client mode 0:Disable (default) 1:Enable
	DNSEN	DNS for remote host	R/W	1	0:Disable (default) 1:Enable
	C[CH]_CLI_IP1	Remote host IP address	R/W	15	Default:192.168.1.99

	C[CH]_CLI_PP1	Remote host port number	R/W	5	1~65535; Default:5000
	C[CH]_DOMAIN	Remote host name	R/W	32	Default:www.w5500.com
	C[CH]_RECONTIME	Reconnect time	R/W	5	Valid for TCP client The value range is:0~60000 Unit:ms Default:0 (reconnect immediately)
	NETBIOS	NetBIOS	R/W	1	0:Disable(default) 1:Enable
Command for configuring oral information	COM[CH]	Serial port parameters	R/W	10	Default:9,1,0,1,0
	C[CH]_BAUD	Baud rate index	R/W	2	0:1200; 1:2400; 2:4800 3:9600; 4:14400; 5:19200 6:38400; 7:56000; 8:57600 9:115200 (default); 10:128000 11:234000; 12:256000; 13:468000 14:921600; 15:1152000
	C[CH]_DATAB	Data bit index	R/W	1	0:7 digits 1:8-bit (default)
	C[CH]_STOPB	Stop bit	R/W	1	0:0.5 1:1 (default) 2:1.5 3:2
	C[CH]_PARITY	Parity bit	R/W	1	0:Disable (default) 1:Odd 2:Even
	C[CH]_SER_C	Serial flow control	R/W	1	0:No flow control (default)
	C[CH]_BUF_CLS	Clear Buffer if Connected	R/W	1	Valid only in TCP modes 0:Disable (default) 1:Enable
	C[CH]_SER_LEN	packet length	R/W	4	The value range is:0~2048 bytes Default:0 (no packet)
	C[CH]_SER_T	Serial data packing Nagle wait time (ms)	R/W	5	Value range:0~60000 Unit:ms Default:0 (Disable))
	C[CH]_IT	Timeout disconnect time	R/W	5	Valid only in TCP modes Value range:0~60000 Unit:ms Default:0 (disable this function)
	C[CH]_TCPAT	Heartbeat detection time	R/W	3	Valid only in TCP modes Value range:0~255 Unit:5s Default:0 (disable this feature)
	C[CH]_LINK_P	Connection password verification	R/W	1	Valid only in TCP server mode 0:Disable (default) 1:Enable

	C[CH]_LINK_T	Connection establishment conditions	R/W	1	Valid only in TCP modes 0:Connect when power on (default) 1:Connect when receiving data from serial
	C[CH]_LINK_M	Send a message after connecting	R/W	1	Valid only in TCP modes 0:Disable (Default) 1:Send module name 2:Send MAC address 3:Send IP address
Management commands	C[CH]_SEND_NUM	Serial sent byte	R	-	Display range is 0 ~ 4294967295
	C[CH]_RCV_NUM	Serial received byte	R	-	Display range is 0 ~ 4294967295
	C[CH]_NETSEND	Network sent byte	R	-	Display range is 0 ~ 4294967295
	C[CH]_NETRCV	Network received byte	R	-	Display range is 0 ~ 4294967295
	PRE	List preset values	R	-	-
	LIST	List all commands	R	-	-
	RUNTIME	Module uptime	R	-	-
	VER	Firmware version	R	-	-
	MAC	MAC address	R	-	-
	SN	Serial number	R	-	-
	TYPE	Module P/N	R	-	-
	WEB_PORT	Web configuration port number	R/W	5	1 ~ 65535;Default:80

7.4.2 AT data transmission command list

The AT data transmission command enables WIZ-IP32 to achieve data transmission and reception in AT command mode. (R:searchable; W:Configurable; R/W:searchable and configurable)

Table 7-3 AT data transmission command list

Type	Command name	Function	Attribute	Max length	Parameters
Data transmission command	LINK	PHY link status	R	-	0:PHY link not connect 1:PHY link connected
	C[CH]_LISTEN	Listening on TCP	W	-	-
	C[CH]_CONNECT	Initiate TCP connection	W	-	-
	C[CH]_TCP_STATUS	TCP connection status	R	-	0:TCP closed 1:TCP connected
	C[CH]_UDP	Establish UDP	W	-	-
	C[CH]_SEND	Send the length	W	4	Range:0~2048 Default:0 (any length)
	C[CH]_RLEN	Number of bytes of data to be received	R	-	-
	C[CH]_RCV	Receive Data	W	4	Range:0~2048 Default:0 (any length)
	C[CH]_CLEAR	Clear network receiving buffer	W	-	-
	C[CH]_DISCON	Close socket	W	-	-

7.5 Detailed description of AT configuration command

7.5.1 Control command

AT(Terminal check)

Command format		Parameters	Usage
AT		Nil	Read
Response	OK\r\n		
Example	*Command:AT\r\n* *Response:OK\r\n*		

Description: In AT command mode, it can be configured.

ECHO(Enable or disable echoing)

Command format		Parameters	Usage
AT+ECHO?		Nil	Read
AT+ECHO=<parameter>		0:Echo off 1:Echo on (default)	Set
Response	[ECHO] Value is:<value>\r\nOK \r\n		
Example	*Command:AT+ECHO?\r\n* *Response:[ECHO] Value is:1\r\nOK\r\n*		

Command description: The echo function refers to returning the data input from the serial port as it is, and it only exists in the AT command mode. When users configure the module using serial port software, enabling the echo function helps them perform the configuration more conveniently. However, when configuring the module using embedded modules such as microcontrollers, enabling the echo function will cause troubles, and it is recommended to disable it in this case.

DEBUGMSGEN(Debug message)

Command format		Parameters	Usage
AT+ DEBUGMSGEN?		Nil	Read
AT+DEBUGMSGEN=<parameter>		0:Disable 1:Enable (default)	Set
Response	[DEBUGMSGEN] Value is:<value>\r\nOK \r\n		
Example	*Command:AT+DEBUGMSGEN=1\r\n* *Response:[DEBUGMSGEN] Value is:1\r\nOK\r\n*		

Command description: This command is used to configure whether the module enables debug information. The debug information includes the module's basic information and module state change information, which will be output through the serial port during startup for users to check conveniently. If users do not need this information, they can disable the debug information output via this command.

NAME(Module name)

Command format	Parameters	Usage
AT+NAME?	Nil	Read
AT+NAME =<parameter>	User defined module name. It must be numbers, alphabets or the combination of both. Maximum length is 15 bytes. Cannot be null Default: module P/N	Set
Response	[NAME] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+NAME=User1\r\n* *Response:[NAME] Value is:User1\r\nOK\r\n*	

Command description: Users can customize module names through this command.

Note: This command can be used in conjunction with the NetBIOS function. If the NetBIOS function is enabled, users can access the module through the module name in the browser. Please refer to the "AT+NETBIOS" command.

PASS(Module password)

Command format	Parameters	Usage
AT+PASS?	Nil	Read
AT+PASS=<parameter>	User define password. It must be numbers, alphabets or the combination of both. Maximum length is 15 byte. It is case sensitive and cannot be null. Default:admin	Set
Response	[PASS] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+PASS=Admin1\r\n* *Response:[PASS] Value is:Admin1\r\nOK\r\n*	

Command description:The module password is used for restoring factory settings, verifying connection passwords, verifying web login passwords, and saving configurations and restarting the module.

Note:

Factory reset function:

If you need to use the AT command to restore factory settings, you must enter the correct module password. Please refer to the "DEFAULT" command for detailed functions.

Connection password verification function:

To enhance communication security, the WIZ-IP32 module provides a "connection password verification" function. When the module acts as a TCP Server to communicate with a client, if the "connection password verification" function is enabled, the client must enter the module password through the network port after establishing a connection with the module. If the password is correct, communication can start; if the password is incorrect, a prompt to re-enter the correct password will be displayed. For detailed functions, please refer to the "C[CH]_LINK_P" command.

Web login password verification function:

If you need to log in to the module configuration webpage, you must enter the correct module password on the login verification page.

Save configuration and restart module functionality:

To save the current configuration and restart the module, the correct module password must be entered to reset. After using this command, all sockets will be automatically closed. Please refer to the "RESET" command for detailed functions.

DEFAULT(Reset to factory default)

Command format	Parameters	Usage
AT+DEFAULT=<parameter>	module password; Default:admin	Set
Response	OK\r\n	
Example	*Command:AT+DEFAULT=admin\r\n* *Response:OK\r\n*	

Command description: The module password must be completely correct in order to restore factory settings. The module password can be queried and set through the "PASS" command.

RESET(Save and restart the module)

Command format	Parameters	Usage
AT+RESET=<parameter>	module password; Default:admin	Set
Response	OK\r\n	
Example	*Command:AT+RESET=admin\r\n* *Response:OK\r\n*	

Command description:

1 Save the current configuration information;

2 Restart the module to make the configuration information take effect;

3 When users need to directly enter data transmission mode during the next power on, they can configure the parameter value of the "AT+START-MODE=1\r\n" command to be 1, and then configure the "RESET" command;

4 The module password can be queried and set through the "PASS" command.

EXIT(Save and exit command mode)

Command format	Parameters	Usage
AT+EXIT	Nil	Execute
Response	OK\r\n	
Example	*Command:AT+EXIT\r\n* *Response:OK\r\n*	

Command description:

1 Save the current configuration information;

2 Make the configuration information effective;

3 Exit AT command mode and enter data transmission mode.

SAVE(Save settings)

Command format	Parameters	Usage
AT+SAVE	Nil	Execute
Response	OK\r\n	
Example	*Command:AT+SAVE\r\n* *Response:OK\r\n*	

Command description:

1 Save the current configuration information;

2 Make the configuration information effective;

3 After configuring this command, the module remains in AT command mode.

7.5.2 Module information configuration command

START_MODE(Start mode)

Command format	Parameters	Usage
AT+START_MODE?	Nil	Read
AT+START_MODE=<parameter>	0:AT command mode (default) 1:Data pass-through mode	Set
Response	[START_MODE] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+START_MODE=1\r\n* *Response:[START_MODE] Value is:1\r\nOK\r\n*	

Command description:This command configures the running mode for the next startup. When users need to directly enter data transmission mode during the next power on, they can configure the parameter value of the "START_MODE" command to be 1, and then configure the "RESET" command.

C[CH]_OP(Operating mode)

Command format	Parameters	Usage
AT+C[CH]_OP?	Nil	Read
AT+C[CH]_OP=<parameter>	<parameter>: 0:TCP Server(default) 1:TCP Client 2:UDP 16:Modbus RTU-TCP Server 17:Modbus RTU-TCP Client 18:Modbus RTU-UDP 32:Modbus ASCII-TCP Server 33:Modbus ASCII-TCP Client 34:Modbus ASCII-UDP	Set
Response	[C[CH]_OP] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_OP=1\r\n* *Response:[C1_OP] Value is:1\r\nOK\r\n*	

Command description: When transmitting data in AT command mode, if you need to use this command to modify the working mode of the module channel, you need to first call the "DISCON" command to turn off TCP/UDP.

IP_MODE (IP acquisition method)

Command format	Parameters	Usage
AT+IP_MODE?	Nil	Read
AT+IP_MODE=<parameter>	0:Static IP mode (default) 1:DHCP mode	Set
Response	[IP_MODE] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+IP_MODE=1\r\n* *Response:[IP_MODE] Value is:1\r\nOK\r\n*	

Command description: When selecting the static retrieval method, users need to set their own IP address, gateway, subnet mask, DNS server address, etc., or they can choose the default settings; When selecting DHCP mode, WIZ-IP32 will dynamically obtain IP information through DHCP servers in the local area network.

IP(IP address)

Command format	Parameters	Usage
AT+IP?	Nil	Read
AT+IP=<parameter>	Default:192.168.1.88	Set
Response	[IP] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+IP=192.168.1.88\r\n* *Response:[IP] Value is:192.168.1.88\r\nOK\r\n*	

Command description: The IP address is represented in IPv4 standard format, such as 192.168.1.88; Its maximum length is 15 bytes, and the range of values for each decimal number is from 0 to 255. This setting

only takes effect when the IP configuration mode (IP_MODE) is set to 'Static IP Mode'.

MASK(Subnet mask)

Command format	Parameters	Usage
AT+MASK?	Nil	Read
AT+MASK=<parameter>	Default:255.255.255.0	Set
Response	[MASK] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+MASK=255.255.255.0\r\n* *Response:[MASK] Value is:255.255.255.0\r\nOK\r\n*	

Command description: The subnet mask is represented in IPv4 standard format, such as 255.255.255.0, with a maximum length of 15 bytes. This setting only takes effect when the IP configuration mode (IP_MODE) is set to 'Static IP Mode'.

GATEWAY(Gateway)

Command format	Parameters	Usage
AT+GATEWAY?	Nil	Read
AT+ GATEWAY =<parameter>	Default:192.168.1.1	Set
Response	[GATEWAY] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+GATEWAY=192.168.1.1\r\n* *Response:[GATEWAY] Value is:192.168.1.1\r\nOK\r\n*	

Command description: The gateway IP address is represented in IPv4 standard format, such as 192.168.1.1, with a maximum length of 15 bytes. This setting only takes effect when the IP configuration mode (IP_MODE) is in 'Static IP Mode'.

DNS(DNS server address)

Command format	Parameters	Usage
AT+DNS?	Nil	Read
AT+DNS=<parameter>	Default:114.114.114.114	Set
Response	[DNS] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+DNS=114.114.114.114\r\n* *Response:[DNS] Value is:114.114.114.114\r\nOK\r\n*	

Command description: The DNS server address is represented in IPv4 standard format, such as 114.114.114.114, with a maximum length of 15 bytes. This configuration does not accept address inputs ending in ". 0" or ". 255", for example:xxx.xxx.xxx.0 or xxx.xxx.xxx.255.

C[CH]_PORT(Local port number)

Command format	Parameters	Usage
AT+C[CH]_PORT?	Nil	Read
AT+C[CH]_PORT=<parameter>	<parameter>: Range:1~65535, default:5000	Set
Response	[C[CH]_PORT]Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_PORT=5000\r\n* *Response:[C1_PORT] Value is:5000\r\nOK\r\n*	

Command description: This command is used to configure the local port number. The module operates in TCP Server and UDP mode with the corresponding number of channels, and communicates with the user module through this port.

Note: Some protocols in the Ethernet protocol have default port numbers, which should be avoided. The default occupied port numbers are shown in the appendix.

C[CH]_BIND(Local port binding)

Command format	Parameters	Usage
AT+C[CH]_BIND?	Nil	Read
AT+C[CH]_BIND=<parameter>	0:Disable (Default) 1:Enable	Set
Response	[C[CH]_BIND] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_BIND=1\r\n* *Response:[C1_BIND] Value is:1\r\nOK\r\n*	

Command description: When the module operates in TCP Client mode on the corresponding serial channel, after binding the port number, WIZ-IP32 will always use the local port number configured by the "C[CH]_PORT" command.

C[CH]_DNSEN(DNS enable**)**

Command format	Parameters	Usage
AT+C[CH]_DNSEN?	Nil	Read
AT+C[CH]_DNSEN=<parameter>	0:Disable (Default) 1:Enable	Set
Response	[C[CH]_DNSEN] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_DNSEN=1\r\n* *Response:[C1_DNSEN] Value is:1\r\nOK\r\n*	

Command description: This setting takes effect only when the module operates in TCP Client or UDP mode on the corresponding serial channel. After enabling DNS functionality, WIZ-IP32 can access remote hosts using domain names. WIZ-IP32 performs a DNS query upon each power-up.

Notes:

1 If this is enabled and "C[CH]_DOMAIN" is set, "C[CH]_CLI_IP1" command will be invalid. Module will be communicated with the host defined-by "C[CH]_DOMAIN" command.

2 If this is disabled and "C[CH]_CLI_IP1" is set, "C[CH]_DOMAIN" command will be invalid. Module will be communicated with the IP defined-by-"C[CH]CLI_IP1".

3 To use the DNS function, the module's IP information must be configured correctly. It is recommended to configure the module to DHCP mode.

C[CH]_CLI_IP1(Remote host IP address)

Command format	Parameters	Usage
AT+C[CH]_CLI_IP1?	Nil	Read
AT+C[CH]_CLI_IP1=<parameter>	Default:192.168.1.99	Set
Response	[C[CH]_CLI_IP1] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_CLI_IP1=192.168.1.99\r\n* *Response:[C1_CLI_IP1] Value is:192.168.1.99\r\nOK\r\n*	

Command description: This command is only effective in TCP Client or UDP mode, used to configure the remote IP address for the corresponding serial channel of WIZ-IP32. The IP address must comply with IPv4 format, such as 192.168.1.99, as defined in the "IP" command. Where each decimal number ranges from 0 to 255, with a maximum length of 15 bytes.

Note: If the parameter value of the "DNSEN" command is 0, the parameters configured by this command are valid.

C[CH]_CLI_PP1(Remote host port number)

Command format	Parameters	Usage
AT+C[CH]_CLI_PP1?	Nil	Read
AT+C[CH]_CLI_PP1=<parameter>	ange:1~65535, default:5000	Set
Response	[C[CH]_CLI_PP1] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_CLI_PP1=5000\r\n* *Response:[C1_CLI_PP1] Value is:5000\r\nOK\r\n*	

Command description: The module operates in TCP Client and UDP modes with the corresponding number of channels.

Note: Some protocols in the Ethernet protocol have default port numbers, which should be avoided. The default occupied port numbers are shown in the appendix.

C[CH]_DOMAIN((Remote host name)

Command format	Parameters	Usage
AT+C[CH]_DOMAIN?	Nil	Read
AT+C[CH]_DOMAIN=<parameter>	Remote host domain name Default:www.w5500.com Maximum length is 32 bytes	Set
Response	[C[CH]_DOMAIN] Value is:<value> \r\nOK\r\n	
Example	*Command:AT+C1_DOMAIN=www.w5500.com\r\n* *Response:[C1_DOMAIN] Value is:www.w5500.com\r\nOK\r\n*	

This command is used to configure the remote host by domain name, which can avoid the inconvenience caused by the IP address change of the remote host. It is effective when the module operates in TCP Client and UDP modes.

Note: The command takes effect only when the "DNSEN" command's parameter value is set to 1.

C[CH]_RECONTIME(Reconnection interval)

Command format	Parameters	Usage
AT+C[CH]_RECONTIME?	Nil	Read
AT+C[CH]_RECONTIME=<parameter>	Range:0~60000 Unit:ms Default:0 (reconnect immediately)	Set
Response	[C[CH]_RECONTIME] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_RECONTIME=1000\r\n* *Response:[C1_RECONTIME] Value is:1000\r\nOK\r\n*	

Command description: The number of channels corresponding to the module is valid when running in TCP Client and transparent mode. This command sets the time interval for the next connection request between WIZ-IP32 and the remote host after the TCP connection is disconnected.

NETBIOS(NetBIOS)

Command format	Parameters	Usage
AT+NETBIOS?	Nil	Read
AT+NETBIOS=<parameter>	<parameter>: 0:Disable NetBIOS function (default) 1:Enable NetBIOS functionality	Set
Response	[NETBIOS] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+NETBIOS=1\r\n* *Response:[NETBIOS] Value is:1\r\nOK\r\n*	

Command description: The NetBIOS protocol can provide module naming services within a local area network. After enabling the NetBIOS function, users can directly access the built-in web page of the WIZ-IP32 by entering "http://module name" in the browser's address bar. The module name is configured via the "NAME" command.

7.5.3 Serial port information configuration command

COM[CH](Serial parameters)

Command format	Parameters	Usage
AT+COM[CH]?	Nil	Read
AT+COM[CH]=<par1>,<par2>,<par3>,<par4>,<par5>	<par1>:Baud rate parameter, refer to the "C[CH]_BAUD" command <par2>:Data bit parameters, refer to the "C[CH]_DATAB" command <par3>:Parity bit parameter, refer to the "C[CH]_PARITY" command <par4>:Stop bit parameter, refer to the "C[CH]_STOPB" command <par5>:Serial port flow control parameters, refer to the "C[CH]_ESER_C" command	Set
Response	[COM[CH]] Value is:<value1><value2><value3><value4><value5>\r\nOK\r\n explain: <value1>:Baud rate parameter value <value2>:Data bit parameter value <value3>:Parity bit parameter value <value4>:Stop bit parameter value <value5>:Serial port flow control parameter values	
Example	*Command:AT+COM1=9,1,0,1,0\r\n* *Response:[COM1] Value is:9,1,0,1,0\r\nOK\r\n*	

Command description: This command is used to configure or query 5 commonly used parameters of serial channel 1 or 2 at once, or it can be configured or queried separately with the corresponding command.

C[CH]_BAUD(Baud rate)

Command format	Parameters	Usage
AT+C[CH]_BAUD?	Nil	Read
AT+C[CH]_BAUD=<parameter>	<parameter>: 0:1200; 1:2400; 2:4800 3:9600; 4:14400; 5:19200 6:38400; 7:56000; 8:57600 9:115200 (default); 10:128000 11:234000; 12:256000; 13:468000 14:921600; 15:1152000	Set
Response	[C[CH]_BAUD] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_BAUD=9\r\n* *Response:[C1_BAUD] Value is:9\r\nOK\r\n*	

Command description: This command is used to configure or query the baud rate of serial channel 1 or 2.

C[CH]_DATAB(Data bit)

Command format	Parameters	Usage
AT+C[CH]_DATAB?	Nil	Read
AT+C[CH]_DATAB=<parameter>	0:7bit 1:8bit (default)	Set
Response	[C[CH]_DATAB] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_DATAB=1\r\n* *Response:[C1_DATAB] Value is:1\r\nOK\r\n*	

Command description: This command is used to configure or query the data bits of serial channel 1 or 2.

C[CH]_STOPB(Stop bit)

Command format	Parameters	Usage
AT+C[CH]_STOPB?	Nil	Read
AT+C[CH]_STOPB=<parameter>	0:0.5 1:1 (default) 2:1.5 3:2	Set
Response	[C[CH]_STOPB] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_STOPB=1\r\n* *Response:[C1_STOPB] Value is:1\r\nOK\r\n*	

Command description: This command is used to configure or query the stop bit of serial channel 1 or 2.

C[CH]_PARITY(Parity bit)

Command format	Parameters	Usage
AT+C[CH]_PARITY?	Nil	Read
AT+C[CH]_PARITY=<parameter>	<parameter>: 0:Nil (default) 1:Odd 2:Even	Set
Response	[C[CH]_PARITY] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_PARITY=0\r\n* *Response:[C1_PARITY] Value is:0\r\nOK\r\n*	

Command description: This command is used to configure or query the parity of serial channel 1 or 2.

C[CH]_SER_C(Serial port flow control)

Command format	Parameters	Usage
AT+C[CH]_SER_C?	Nil	Read
AT+C[CH]_SER_C =<parameter>	<parameter>: 0:No flow control(default)	Set
Response	[C[CH]_SER_C] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_SER_C=0\r\n* *Response:[C1_SER_C] Value is:0\r\nOK\r\n*	

Command description: This command is used to configure or query the serial channel flow control of

serial channel 1 or 2.

C[CH]_BUF_CLS(Clear buffer if connected)

Command format	Parameters	Usage
AT+C[CH]_BUF_CLS?	Nil	Read
AT+C[CH]_BUF_CLS=<parameter>	<parameter>: 0:Do not clear the serial port buffer after establishing a connection (default) 1:Clear the serial port buffer after establishing a connection	Set
Response	[C[CH]_BUF_CLS] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_BUF_CLS=1\r\n* *Response:[C1_BUF_CLS] Value is:1\r\nOK\r\n*	

Command description: The module is effective when running in TCP mode and data transparent mode with the corresponding number of channels. If the connection suddenly disconnects during data exchange, there may be some data in the serial buffer that has not been sent. This command can be used to determine whether to send this data after the connection is reestablished.

C[CH]_SER_LEN(Serial packaging length)

Command format	Parameters	Usage
AT+C[CH]_SER_LEN?	Nil	Read
AT+C[CH]_SER_LEN=<parameter>	<parameter>: The value range is:0~2048 bytes, default:0 (not specified Package)	Set
Response	[C[CH]_SER_LEN] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_SER_LEN=10\r\n* *Response:[C1_SER_LEN] Value is:10\r\nOK\r\n*	

Command description: This command sets the package length for each data transmission. The module's corresponding channel number is valid when running in data transmission mode.

C[CH]_SER_T(Serial data packing Nagle wait time)

Command format	Parameters	Usage
AT+C[CH]_SER_T?	Nil	Read
AT+C[CH]_SER_T=<parameter>	<parameter>: Value range:0~60000, default:0 (not subcontracted), unit:ms	Set
Response	[C[CH]_SER_T] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_SER_T=1000\r\n* *Response:[C1_SER_T] Value is:1000\r\nOK\r\n*	

Command description: This command sets the serial waiting time. After the waiting time passes, it gathers all the data and transmit in one data package. The module's corresponding serial channel number is valid when running in data transmission mode.

C[CH]_IT(Inactivity timeout)

Command format	Parameters	Usage
AT+C[CH]_IT?	Nil	Read
AT+C[CH]_IT=<parameter>	<parameter>: The time interval for timeout disconnection, with a value range of:0~60000, unit:ms, default:0 (disable this function)	Set
Response	[C[CH]_IT] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_IT=1000\r\n* *Response:[C1_IT] Value is:1000\r\nOK\r\n*	

Command description: The module is effective when running in TCP mode and data transparent transmission mode. When WIZ-IP32 works in TCP mode, whether as a server or a client, it may happen that the other party has disconnected (forced disconnection or network failure), but the module is not informed of the disconnection and continues to maintain this invalid connection. In this case, when either party intends to initiate communication, an error that data cannot be delivered will occur.

Setting this parameter enables the TCP connection to be disconnected when no new data is received through the serial port or Ethernet interface within a continuous timeout period during communication using the TCP protocol. When this value is set to 0, it means the TCP connection will never be disconnected.

C[CH]_TCPAT(TCP keepalive interval)

Command format	Parameters	Usage
AT+C[CH]_TCPAT?	Nil	Read
AT+C[CH]_TCPAT=<parameter>	Value range:0~255, unit:5s Default:0 (disable)	Set
Response	[C[CH]_TCPAT] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_TCPAT=1\r\n* *Response:[C1_TCPAT] Value is:1\r\nOK\r\n*	

Command description: The module is effective when running in TCP mode and data transparent transmission mode. When communicating using the TCP protocol, after each data transmission or receive, the WIZ-IP32 will send a "heartbeat detection packet" to the other party within a fixed time period to test if the connection exists. If no response is received after sending the "heartbeat detection packet", the module will automatically disconnect.

C[CH]_LINK_P(TCP password authentication)

Command format	Parameters	Usage
AT+C[CH]_LINK_P?	Nil	Read
AT+C[CH]_LINK_P=<parameter>	0:Disable (Default) 1:Enable	Set
Response	[C[CH]_LINK_P] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_LINK_P=1\r\n* *Response:[C1_LINK_P] Value is:1\r\nOK\r\n*	

Command description: The module is effective when running in TCP Server mode and data transparent transmission mode. To enhance communication security, the WIZ-IP32 module provides a "connection password verification" function. When the module acts as a TCP Server to communicate with a client, if the "connection password verification" function is enabled, the client must send the module's password as the first data packet after establishing a connection with the module. If the password is correct, communication can start; if the password is incorrect, a prompt will be sent to request re-entering the correct password. The module's password can be queried and set using the "PASS" command.

C[CH]_LINK_T(Connection condition)

Command format	Parameters	Usage
AT+C[CH]_LINK_T?	Nil	Read
AT+C[CH]_LINK_T=<parameter>	0:Connect when power on (default) 1:Connect when receiving data from serial	Set
Response	[C[CH]_LINK_T] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_LINK_T=1\r\n* *Response:[C1_LINK_T] Value is:1\r\nOK\r\n*	

Command description: The module is effective when running in TCP Client mode and data transparent transmission mode. It is used to configure the conditions under which the module establishes a connection with the TCP Server. When configured as "Establish connection after serial port receives data", the module will not initiate a connection to the server. It will only send a connection request to the server immediately after the module's serial port receives the first data packet. After the connection is successful, the module's serial port will send the first data packet to the server and start the normal data transparent transmission mode.

C[CH]_LINK_M(Send Hello Message)

Command format	Parameters	Usage
AT+C[CH]_LINK_M?	Nil	Read
AT+C[CH]_LINK_M=<parameter>	0:Disable (Default) 1:Send module ID 2:Send MAC address 3:Send IP address	Set
Response	[C[CH]_LINK_M] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+C1_LINK_M=1\r\n* *Response:[C1_LINK_M] Value is:1\r\nOK\r\n*	

Command description: The command's corresponding serial channel is valid in TCP Client mode, and is valid in both data transparent mode and AT command transmission data mode. This command is used to configure the information that the module sends immediately after the connection is established.

7.5.4 Management commands

C[CH]_SEND_NUM(Number of bytes sent via serial port)

Command format	Parameters	Usage
AT+C[CH]_SEND_NUM?	Nil	Read
Response	[C[CH]_SEND_NUM] Value is:<value>\r\nOK\r\n Display range:0~4294967295	
Example	*Command:AT+C1_SEND_NUM?\r\n* *Response:[C1_SEND_NUM] Value is:2048\r\nOK\r\n*	

Command description: This command is valid when the corresponding channel of the module is in either data transparent transmission mode or AT command data transmission mode.

C[CH]_RCV_NUM(Number of bytes received via serial port)

Command format	Parameters	Usage
AT+C[CH]_RCV_NUM?	Nil	Read
Response	[C[CH]_RCV_NUM] Value is:<value>\r\nOK\r\n Display range:0~4294967295	
Example	*Command:AT+C1_RCV_NUM?\r\n* *Response:[C1_RCV_NUM] Value is:2048\r\nOK\r\n*	

Command description: This command is valid when the corresponding channel of the module is in either data transparent transmission mode or AT command data transmission mode.

C[CH]_NETSEND(Number of bytes sent via network port)

Command format	Parameters	Usage
AT+C[CH]_NETSEND?	Nil	Read
Response	[C[CH]_NETSEND] Value is:<value>\r\nOK\r\n Display range:0~4294967295	
Example	*Command:AT+C1_NETSEND?\r\n* *Response:[C1_NETSEND] Value is:2048\r\nOK\r\n*	

Command description: This command is valid when the corresponding channel of the module is in either data transparent transmission mode or AT command data transmission mode.

C[CH]_NETRCV(Number of bytes received via network port)

Command format	Parameters	Usage
AT+C[CH]_NETRCV?	Nil	Read
Response	[C[CH]_NETRCV] Value is:<value>\r\nOK\r\n Display range:0~4294967295	
Example	*Command:AT+C1_NETRCV?\r\n* *Response:[C1_NETRCV] Value is:2048\r\nOK\r\n*	

Command description: This command is valid when the corresponding channel of the module is in either

data transparent transmission mode or AT command data transmission mode.

PRE(List preset values)

Command format	Parameters	Usage
AT+PRE?	Nil	Read
Response	DEFAULT: DEFAULT: [NAME]: WIZ-IP32 [PASS]: admin [IP]: 192.168.1.88 [MASK]: 255.255.255.0 [GATEWAY]: 192.168.1.1 [DNS]: 114.114.114.114 [WEB_PORT]: 80 [C1_DOMAIN]: www.w5500.com [C1_PORT]: 5000 [C1_BAUD]: 9 [C1_DATAB]: 1 [C1_PARITY]: 0 [C1_STOPB]: 1 [C1_SER_C]: 0 [C1_SER_T] : 0 [C1_SER_LEN]: 0 [C1_CLI_IP1]: 192.168.1.99 [C1_CLI_PP1]: 5000 [C2_DOMAIN]: www.w5500.com [C2_PORT]: 5001 [C2_BAUD]: 9 [C2_DATAB]: 1 [C2_PARITY]: 0 [C2_STOPB]: 1 [C2_SER_C]: 0 [C2_SER_T] : 0 [C2_SER_LEN]: 0 [C2_CLI_IP1]: 192.168.1.99 [C2_CLI_PP1]: 5000 CURRENT: [NAME] : WIZ-IP32 [PASS]: admin [IP]: 192.168.1.88 [MASK]: 255.255.255.0 [GATEWAY]: 192.168.1.1 [DNS]: 114.114.114.114	

[WEB_PORT]:	80
[C1_DOMAIN]:	www.w5500.com
[C1_PORT]	5000
[C1_BAUD]:	9
[C1_DATAB]:	1
[C1_PARITY]:	0
[C1_STOPB]:	1
[C1_SER_C]:	0
[C1_SER_T] :	0
[C1_SER_LEN]:	0
[C1_CLI_IP1]:	192.168.1.99
[C1_CLI_PP1]:	5000
[C2_DOMAIN]:	www.w5500.com
[C2_PORT]:	5001
[C2_BAUD]:	9
[C2_DATAB]:	1
[C2_PARITY]:	0
[C2_STOPB]:	1
[C2_SER_C]:	0
[C2_SER_T] :	0
[C2_SER_LEN]:	0
[C2_CLI_IP1]:	192.168.1.99
[C2_CLI_PP1]:	5000
OK	

LIST(List all commands)

Command format	Parameters	Usage
AT+LIST?	Nil	Read

Response	[Control Command]		
	AT	AT+ECHO	AT+DEBUGMSGEN
	AT+NAME	AT+PASS	AT+DEFAULT
	AT+RESET	AT+EXIT	AT+SAVE
	[module Settings Command]		
	AT+START_MODE	AT+C1_OP	AT+IP_MODE
	AT+IP	AT+MASK	AT+GATEWAY
	AT+DNS	AT+C1_PORT	AT+C1_BIND
	AT+C1_DNSEN	AT+C1_CLI_IP1	AT+C1_CLI_PP1
	AT+C1_DOMAIN	AT+C1_RECONTIME	AT+NETBIOS
	AT+C2_OP	AT+C2_PORT	AT+C2_BIND
	AT+C2_DNSEN	AT+C2_CLI_IP1	AT+C2_CLI_PP1
	AT+C2_DOMAIN	AT+C2_RECONTIME	
	[Management Command]		
	AT+C1_SEND_NUM	AT+C1_RCV_NUM	AT+C1_NETSEND
	AT+C1_NETRCV	AT+C2_SEND_NUM	AT+C2_RCV_NUM
	AT+C2_NETSEND	AT+C2_NETRCV	AT+PRE
	AT+LIST	AT+RUNTIME	AT+VER
	AT+MAC	AT+SN	AT+TYPE
	AT+WEB_PORT		
	[Data Transfer Command]		
	AT+LINK	AT+C1_LISTEN	AT+C1_CONNECT
	AT+C1_TCP_STATUS	AT+C1_UDP	AT+C1_SEND
	AT+C1_RLEN	AT+C1_RCV	AT+C1_CLEAR
	AT+C1_DISCON	AT+C2_LISTEN	AT+C2_CONNECT
	AT+C2_TCP_STATUS	AT+C2_UDP	AT+C2_SEND
	AT+C2_RLEN	AT+C2_RCV	AT+C2_CLEAR
	AT+C2_DISCON		
	OK		

RUNTIME(Module uptime)

Command format	Parameters	Usage
AT+RUNTIME?	Nil	Read
Response	[RUNTIME] Value is:<value>\r\nOK\r\n Description: The format of <value> is ddd-hh-mm-ss, where "d" stands for days, "h" for hours, "m" for minutes, and "s" for seconds. The display range is 000-00-00-00 ~ 999-23-59-59.	
Example	*Command:AT+RUNTIME?\r\n* *Response:[RUNTIME] Value is:003-15-38-42\r\nOK\r\n*	

VER(Firmware version)

Command format	Parameters	Usage
AT+VER?	Nil	Read
Response	[VER] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+VER?\r\n* *Response:[VER] Value is:V1.0\r\nOK\r\n*	

MAC(MAC address)

Command format	Parameters	Usage
AT+MAC?	Nil	Read
Response	[MAC] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+MAC?\r\n* *Response:[MAC] Value is:00.08.DC.11.12.13\r\nOK\r\n*	

SN(Serial number)

Command format	Parameters	Usage
AT+SN?	Nil	Read
Response	[SN] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+SN?\r\n* *Response:[SN] Value is:20190102-111213\r\nOK\r\n*	

TYPE(Module part numbe)

Command format	Parameters	Usage
AT+TYPE?	Nil	Read
Response	[TYPE] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+TYPE?\r\n* *Response:[TYPE] Value is:WIZ-IP32\r\nOK\r\n*	

WEB_PORT(Web configuration port number)

Command format	Parameters	Usage
AT+WEB_PORT?	Nil	Read
AT+WEB_PORT=<parameter>	Range:1 ~ 65,535, Default:80	Set
Response	[WEB_PORT] Value is:<value>\r\nOK\r\n	
Example	*Command:AT+WEB_PORT=80\r\n* *Response:[WEB_PORT] Value is:80\r\nOK\r\n*	

Command description: This port number is used by users to access WIZ-IP32 for web page configuration using a browser, and it is not recommended to modify it. If configured with any port number other than 80, when accessing WIZ-IP32 through a browser, the address bar should enter "IP:Port", for example:192.168.1.88:8000.

Note: Some protocols in the Ethernet protocol have default port numbers, which should be avoided. The default occupied port numbers are shown in the appendix.

7.6 Detailed description of AT data transmission command

LINK (Detect PHY link status)

Command format	Parameters	Usage
AT+LINK?	Nil	Execute
Response	[LINK] Value is:<value>\r\nOK\r\n 0:no PHY link 1:has PHY link	
Example	*Command:AT+LINK?\r\n* *Response:[LINK] Value is:1\r\nOK\r\n*	

C[CH]_LISTEN(Listening on TCP)

Command format	Parameters	Usage
AT+C[CH]_LISTEN	Nil	Execute
Response Description	OK\r\n Upon successful completion	
Response Description	<Error Info>\r\n Module is not in TCP server mode	
Example	*Command:AT+C1_LISTEN\r\n* *Response:OK\r\n*	

Command description: The module's corresponding serial channel number is valid when running in TCP Server mode, which means that the command can only be configured correctly when the "C[CH]_OP" command parameter is "0", otherwise an error will be reported. It can be closed through the "DISCON" command.

Note: After configuring this command, it is necessary to check whether the TCP connection has been established through the "C[CH]_TCP_STATUS" command.

C[CH]_CONNECT(Initiating TCP connection)

Command format	Parameters	Usage
AT+C[CH]_CONNECT	Nil	Execute
Response Description	OK\r\n Upon successful completions	
Response Description	<Error Info>\r\n Module is not in TCP client mode	
Example	*Command:AT+C1_CONNECT\r\n* *Response:OK\r\n*	

Command description: The number of channels corresponding to the module is valid when running in TCP Client mode. This command can only be configured correctly when the parameter of the "C[CH]_OP" command is "1"; otherwise, an error will be reported. After configuring this command, the module immediately initiates a connection request to the TCP Server. If the TCP connection is not established after 3

seconds, an error will be reported.

Note: After configuring this command, it is necessary to check whether the TCP connection has been established through the "C[CH]_TCP_STATUS" command.

C[CH]_TCP_STATUS(TCP connection status)

Command format	Parameters	Usage
AT+C[CH]_TCP_STATUS?	Nil	Read
Response Description	[C[CH]_TCP_STATUS] Value is:<value>\r\nOK\r\n 0:TCP not connected 1:TCP Connected	
Example	*Command:AT+C1_TCP_STATUS?\r\n* *Response:[C1_TCP_STATUS] Value is:1\r\nOK\r\n*	

Command description: This command is valid when the module is running in both TCP Server and TCP Client modes, and can be used to check whether the TCP connection has been established.

C[CH]_UDP(Establish UDP connection)

Command format	Parameters	Usage
AT+C[CH]_UDP	Nil	Execute
Response Description	OK\r\n Upon successful completions	
Response Description	<Error Info>\r\n Module is not in TCP client mode	
Example	*Command:AT+C1_UDP\r\n* *Response:OK\r\n*	

Command description: The module operates in UDP mode with the corresponding number of channels. This command can only be configured correctly when the parameter of the "C[CH]_OP" command is "2", otherwise an error will be reported.

C[CH]_SEND(Send byte of data)

Command format	Parameters	Usage
AT+C[CH]_SEND=<parameter>	Range:0~2048 (bytes) Default:0 (any length)	Set
Response	[C[CH]_SEND] Value is:<value>\r\nOK\r\n Set the length of data to be sent	
Example1	*Assuming the module is in TCP mode and the TCP connection is normal* *Command:AT+C1_TCP_STATUS?\r\n* *[C1_TCP_STATUS] Value is:1\r\nOK\r\n* *Command:AT+C1_SEND=5\r\n* *Response:[C1_SEND] Value is:5\r\nOK\r\n* *Then send data via serial:12345* *Response:5*	
Example2	*Assume module is in TCP mode and TCP connection is not connected* *Command:AT+C1_TCP_STATUS?\r\n* *[C1_TCP_STATUS] Value is:0\r\nOK\r\n* *Command:AT+C1_SEND=5\r\n* *Response:[C1_SEND] Value is:5\r\nOK\r\n* *Sending data to module serial port:12345* *Response:0*	

Command description:

1. If the module is operating in TCP Server or TCP Client mode, it is necessary to check whether the TCP connection has been established before configuring this command. Please refer to the "TCP_STATUS" command.
2. After successfully configuring this command, the serial port will receive the corresponding return value. Any character sent to the module's serial port next time will be sent as data, after which it will exit data transmission and enter AT command mode.
3. If the specified transmission length is 0, the serial port will package data with a default frame interval of 50ms. Starting from the first byte sent by the serial port, if the serial port remains idle for more than 50ms thereafter, the data will be sent immediately. If the continuous byte length of the serial port reaches 2048, the excess part will be automatically discarded.
4. If the specified transmission length is not 0, that is, the serial port packages data by length, the characters of the specified length sent to the module's serial port will be packaged and sent immediately. If the length exceeds the specified value, the excess part will be automatically discarded; if it is less than the specified length, it will wait indefinitely.
5. After the data is successfully sent from the module's network port, the module's serial port will

immediately output a reply message in the format:"<len>", where <len> is the number of bytes of data successfully sent through the module's network port. Users can use this to determine whether their data has been successfully sent.

C[CH]_RLEN(Number of bytes pending receipt)

Command format	Parameters	Usage
AT+C[CH]_RLEN?	Nil	Read
Response Description	[C[CH]_RLEN] Value is:<value>\r\nOK\r\n Range:0~2048	
Example	*Data to be received:abcdef* *Command:AT+C1_RLEN?\r\n* *Response:[RLEN] Value is:6\r\nOK\r\n*	

Command description: Query the Byte Length of Pending Data in the Network Port Receive Buffer for the corresponding Channel

C[CH]_RCV(Receive data)

Command format	Parameters	Usage
AT+C[CH]_RCV=<parameter>	Range:0 ~ 2048 Default:0 (Whatever length)	Set and execute
Response	[C[CH]_RCV] Value is:<value>\r\nOK\r\n<data>	
Example	*Data to be received:abcdef* *Command:AT+C1_RCV=0\r\n* *Response:[C1_RCV] Value is:6\r\nOK\r\nabcdef*	

Command description:

1. If the module is operating in TCP Server or TCP Client mode, it is necessary to check whether the TCP connection has been established before configuring this command. Please refer to the "TCP_STATUS" command.
2. After successfully configuring this command, the module's serial port will output the data in the network port's receive buffer, then exit data receive and enter AT command mode.
3. The network port's receive buffer size is 2048 bytes. If the total length of data in the network port's receives buffer reaches 2048 bytes, the network port will stop receiving further data.
4. If the length of data in the network port's receive buffer is less than the specified length set by the "RCV" command, no data will be received, meaning the number of received data bytes will be 0.

C[CH]_CLEAR(Clear the network receiving buffer)

Command format	Parameters	Usage
AT+C[CH]_CLEAR	Nil	Execute
Response	OK\r\n	
Example	*Command:AT+C1_CLEAR\r\n* *Response:OK\r\n*	

Command description: If the corresponding channel does not need the data in the network port receiving buffer, this command can be used to clear it.

C[CH]_DISCON(Disconnect)

Command format	Parameters	Usage
AT+C[CH]_DISCON	Nil	Execut
Response	OK\r\n	
Example	*Command:AT+C1_DISCON\r\n* *Response:OK\r\n*	

Command description: Close the TCP or UDP connection for the corresponding number of channels.

7.7 AT command script examples

7.7.1 TCP server mode script example

AT\r\n	*//Terminal detection*
OK\r\n	
AT+DEBUGMSGEN=0\r\n	*//Close serial port debugging information*
[DEBUGMSGEN] Value is:0\r\nOK\r\n	
AT+ECHO=0\r\n	*//Turn off echo display*
[ECHO] Value is:0\r\nOK\r\n	
AT+START_MODE=1\r\n	*//Configure the next boot run mode to be data transmission mode*
[START_MODE] Value is:1\r\nOK\r\n	
AT+C1_OP=0\r\n	*//Configure the working mode to TCP Server*
[C1_OP] Value is:0\r\nOK\r\n	
AT+IP_MODE=0\r\n	*//Configure as a static IP*
[IP_MODE] Value is:0\r\nOK\r\n	
AT+IP=192.168.1.88\r\n	*//Configure local IP address*
[IP] Value is:192.168.1.88\r\nOK\r\n	
AT+C1_PORT=5000\r\n	*// Configure local port number*
[C1_PORT] Value is:5000\r\nOK\r\n	
AT+RESET=admin\r\n	*//Save configuration, the module will enter data transmission mode after restart*
OK\r\n	

7.7.2 TCP client AT command mode script example.

AT\r\n	*//Terminal detection*
OK\r\n	
AT+DEBUGMSGEN=0\r\n	*//Close serial port debugging information*
[DEBUGMSGEN] Value is:0\r\nOK\r\n	
AT+ECHO=0\r\n	*//Turn off echo display*
[ECHO] Value is:0\r\nOK\r\n	
AT+START_MODE=0\r\n	*//Configure the next startup run mode to AT command mode*
[START_MODE] Value is:0\r\nOK\r\n	
AT+C1_OP=1\r\n	*//Configure the working mode to TCP Client*
[C1_OP] Value is:1\r\nOK\r\n	
AT+IP_MODE=1\r\n	*//Configure the module to DHCP mode*
[IP_MODE] Value is:1\r\nOK\r\n	
AT+DNSEN=0\r\n	*//Disable DNS function*
[DNSEN] Value is:0\r\nOK\r\n	
AT+C1_CLI_IP1=192.168.1.99\r\n	*//Configure remote TCP server IP address*
[C1_CLI_IP1] Value is:192.168.1.99\r\nOK\r\n	
AT+C1_CLI_PP1=5000\r\n	*//Configure remote TCP server port number*
[C1_CLI_PP1] Value is:5000\r\nOK\r\n	
AT+SAVE\r\n	*//Save configuration Information;*
OK\r\n	
AT+C1_CONNECT\r\n	*//Initiate a connection request to a remote TCP server*
OK\r\n	
AT+C1_TCP_STATUS?\r\n	*//Query TCP connection status*
[C1_TCP_STATUS] Value is:1\r\nOK\r\n	
AT+C1_SEND=5\r\n	*//Notify the module that 5 bytes of data will be sent*
OK\r\n	
Seial sends data:12345	

Serial receives data:5

TCP server sends data:abcdef

AT+C1_RLEN?\r\n *//Query the length of data to be received*

[C1_RLEN] Value is:6\r\nOK\r\n

AT+C1_RCV=6\r\n *//Receive All*

[C1_RCV] Value is:6\r\n

OK\r\n

abcdef

AT+C1_CLEAR\r\n *//Clear the network port receiving cache*

OK\r\n

AT+C1_DISCON\r\n *// Close TCP connection*

OK\r\n

8 Web configuration

WIZ-IP32 supports web page configuration. Suggest using browsers: Chrome, Firefox, other browsers may have display or working issues. Let's take Chrome browser as an example to illustrate.

Before using web configuration, it is necessary to ensure that WIZ-IP32 can be accessed correctly. That is, if configuring within a local area network, WIZ-IP32 needs to be on the same network segment as the computer. If configuring remotely, WIZ-IP32 needs to be mapped to a public IP. Taking the local area network as an example for configuration, the specific configuration method is introduced as follows.

8.1 Login page

Open the Chrome browser and enter the IP address of the WIZ-IP32 module in the address bar. The factory default is 192.168.1.88, and the login interface shown in the following figure will appear.

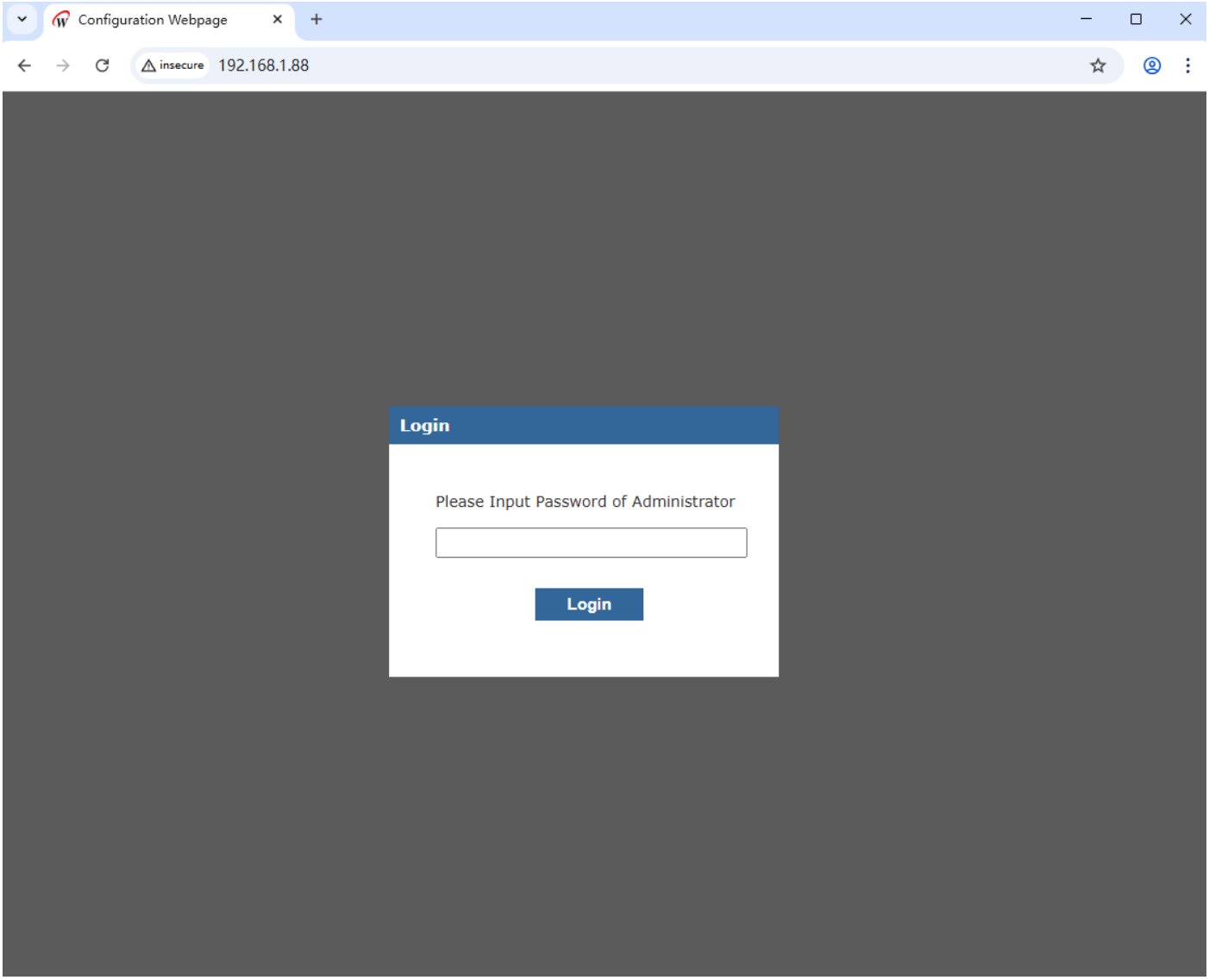


Figure 8-1 Web login interface

The default login password is "**admin**". Click "Login" to log in and enter the WIZ-IP32 homepage. It should be noted that after logging into the WIZ-IP32 webpage, in order to ensure the security of module information, if the user does not take any action within 5 minutes, they need to log in again, as shown in the following figure.

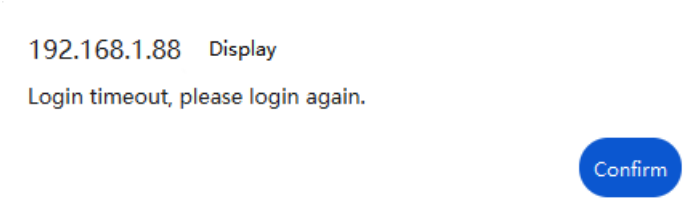


Figure 8-2 Web prompt - login timeout

As shown in the figure below, it is the basic information page after logging in to the WIZ-IP32 configuration page. If you need to log out, click the "Logout" link in the upper right corner. This page provides an overview of the basic configuration information for WIZ-IP32.

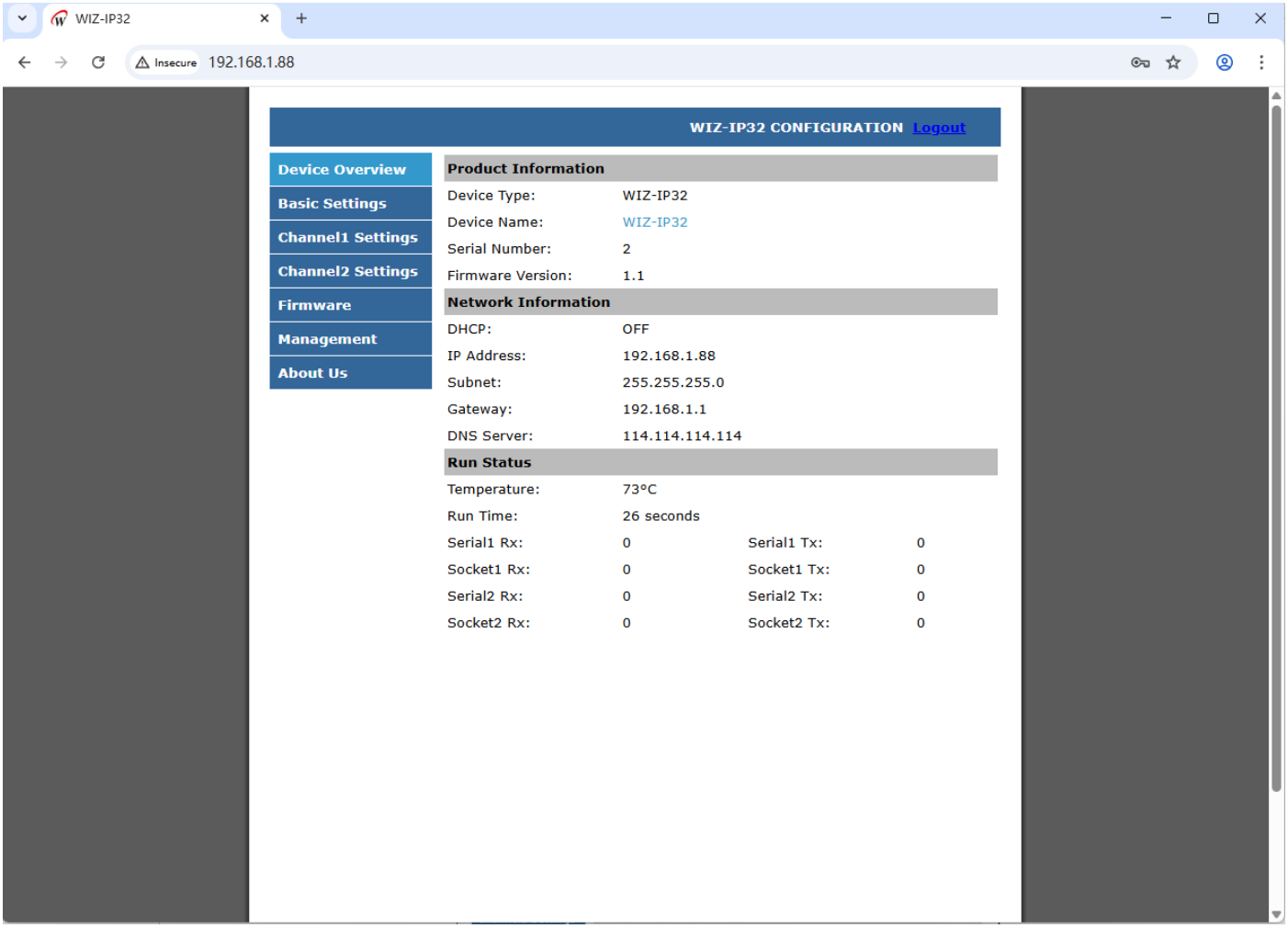


Figure 8-3 Basic information page

8.2 Basic settings

As shown in the following figure, it is the WIZ-IP32 basic parameter configuration page, which is divided into five parts for explanation.

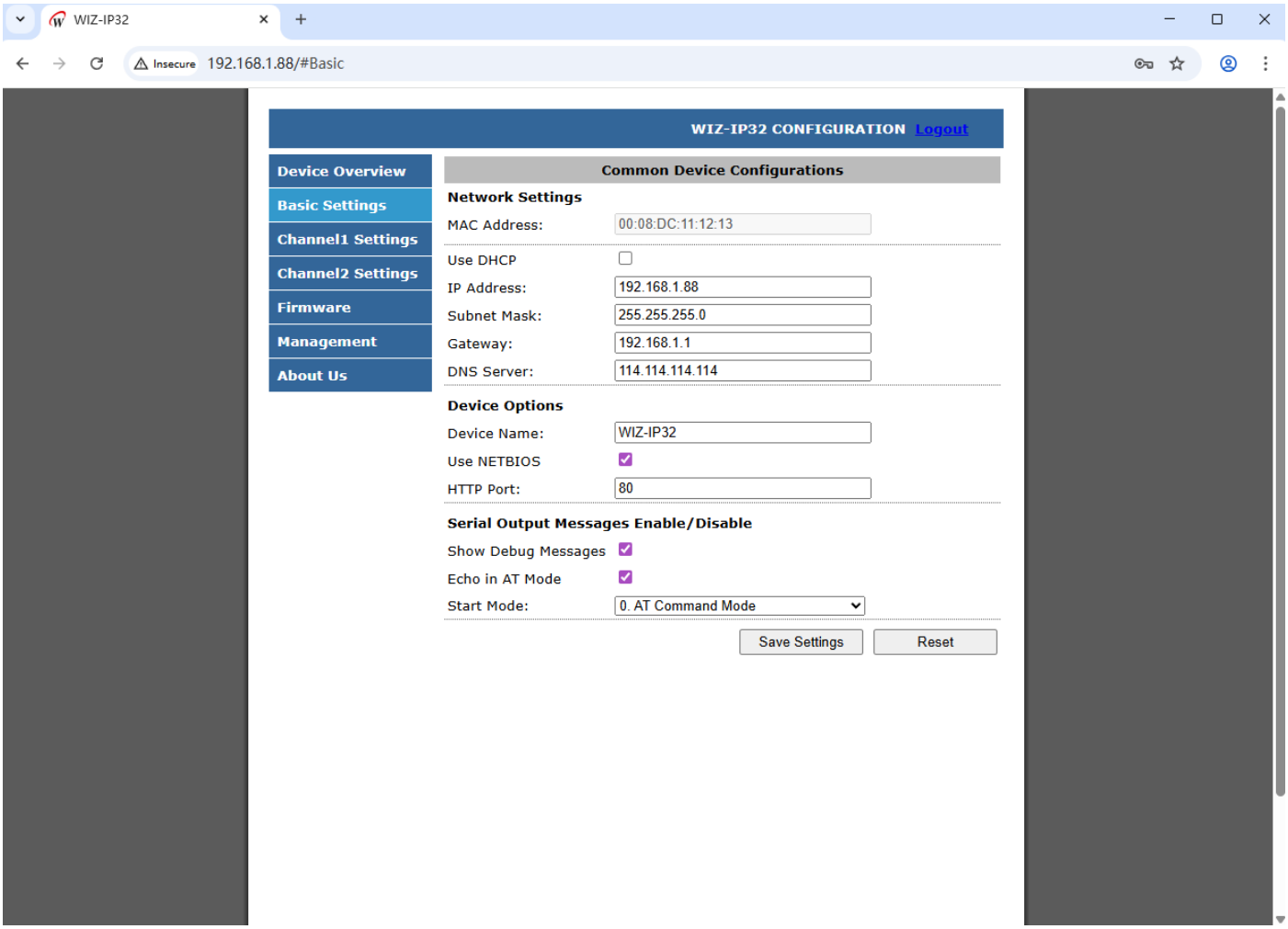


Figure 8-4 Basic configuration page

Network setting

MAC Address: Display the module's MAC address;

Use DHCP: Enable the option to dynamically obtain IP addresses. Checking this option will enable this feature;

IP Address: Display/set module IP address, default is 192.168.1.88;

Subnet Mask: Display/set module subnet mask, default is 255.255.255.0;

Gateway: Display/set module gateway, default is 192.168.1.1;

DNS Server: Display/set module DNS server address, default is 114.114.114.114.

Device options

Device Name: Set/display the module name, which must be a number, letter, or a combination of both,

and cannot be empty. It is case sensitive and can be up to 15 bytes long;

Use NETBIOS: Checking this option will enable the NetBIOS function (it is disabled by default). After enabling the NetBIOS function, users can directly access the built-in web page of WIZ-IP32 by entering "http://module Name" in the browser's address bar.

HTTP Port: The web server port number of WIZ-IP32, with a default of 80 and a value range of 0-65535. If not set to 80, the port number should be entered in the browser address bar, for example:192.168.1.88:8000;

Note: If WIZ-IP32 is working in TCP Server mode, the HTTP Port must not be set to the same local port as the module. Some protocols in the Ethernet protocol have default port numbers, which should be avoided. The default occupied port numbers are shown in the appendix.

Serial output messages enable/disable

Show Debug Message Enable: Display module debugging information. Checking this feature will print module debugging information from the serial port, which is enabled by default;

Echo in AT Mode: Enables the AT command echo function. The echo function means that the WIZ-IP32 device returns the input command to the serial port in its original form, which is then displayed on the serial port software interface. When users configure the module using serial port software, enabling the echo function helps simplify the configuration process. However, when configuring the module with embedded devices such as microcontrollers, enabling the echo function may cause issues—under this circumstance, the echo function must be disabled. This option is checked by default to enable the echo function.

Start Mode: Module startup mode configuration, which can be set to AT command mode and data transmission mode. WIZ-IP32 will run in this mode after the next power on/reset;

Save settings: Click the "Save Settings" button, and the webpage will prompt that the save was successful. You need to further click the "Reset" button to restart the module for all configurations to take effect;

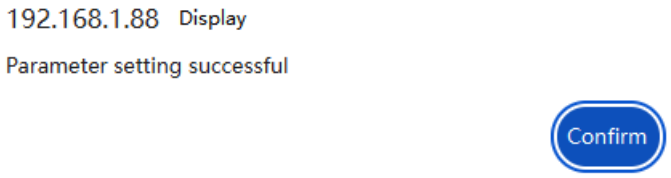


Figure 8-5 Web prompt - save successful

Reset: Reset button, clicking this button can restart the module (without saving configuration function). After clicking this button, the webpage will pop up the following dialog box. Click "OK" to restart the module and the webpage will jump to the login interface.

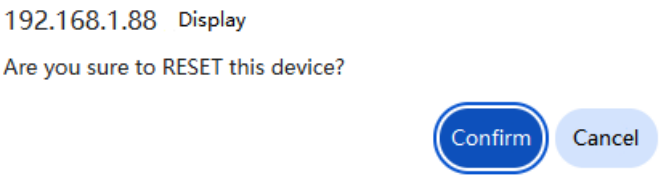


Figure 8-6 Web prompt - reset

8.3 Channel configuration

As shown in the following figure, it is the WIZ-IP32 channel 1 configuration page. The following is a detailed explanation:

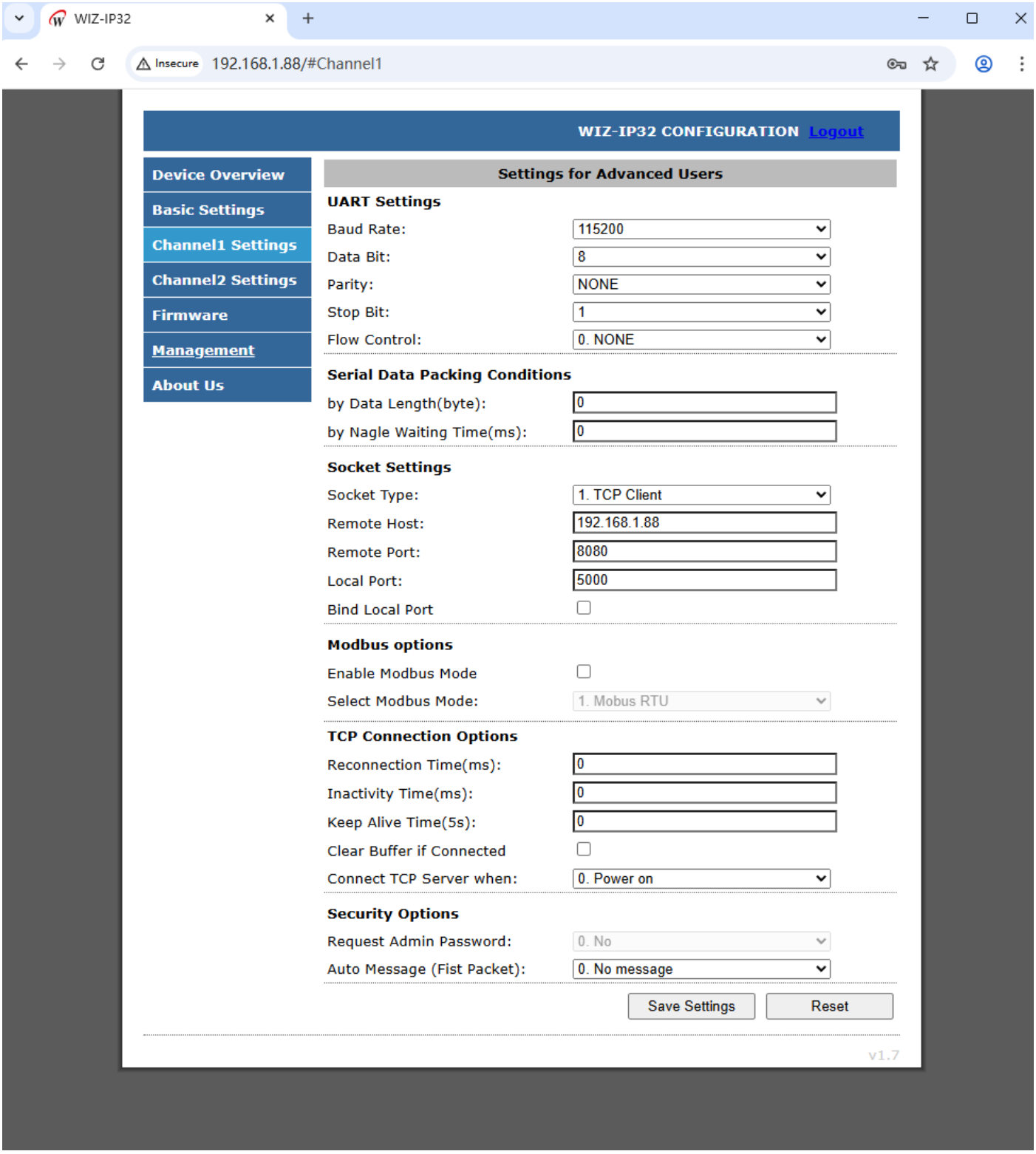


Figure 8-7 Channel 1 configuration page

UART setting

Baud Rate: Baud rate, default is 115200, users can choose 16 commonly used baud rate values between 1.2Kbps and 1.152Mbps;

Date Bit: Data bit, default is 8, can be set to 7 or 8;

Parity: Parity bit, default is NONE, can be set as NONE, ODD, EVEN;

Stop Bit: Stop bit default is 1, can be set to 0.5,1,1.5,2;

Flow Control: Hardware flow control, default is NONE.

Serial data packing conditions

by Data Length(byte): Set/display the byte length of the serial port data packet, package the data input from the module's serial port according to the byte length and convert it into Ethernet data packets for transmission. The default value is 0 (no packet), and the maximum value is 2048 bytes;

by Nagle Waiting Time(ms): Packaging the data input from the module's serial port into Ethernet packets at time intervals, with a default value of 0 (no packet) and a maximum value of 60000 in ms;

Socket Setting

Socket Type: Module working mode configuration, which can be set to TCP Server, TCP Client, and UDP modes;

Remote Host: Display/set the IP address/domain name of the remote host, which is valid when the module is running in TCP Client and UDP mode. The default is 192.168.1.99. If this is set as the domain name of the remote host, the module will automatically perform domain name resolution. The maximum length of the domain name is 32 characters;

Remote Port: Display/set the remote host port number, default is 5000, with a value range of 0~65535;

Note: Some protocols in the Ethernet protocol have default port numbers, which should be avoided. The default occupied port numbers are shown in the appendix.

Local Port: Display/set the local port number, default is 5000, with a value range of 0~65535;

Note: Some protocols in the Ethernet protocol have default port numbers, which should be avoided. The default occupied port numbers are shown in the appendix.

Bind Local Port: Bind local port number, selecting it will enable the bind local port number function, which is effective when the module is running in TCP Client mode.

Modbus option

Enable Modbus Mode: Enable Modbus function, checking it will enable Modbus function;

Select Modbus Mode: Select Modbus mode, which can be set to Modbus RTU or Modbus ASCII.

TCP connection option

Reconnection Time (ms): Set/display the reconnection time, which is valid when the module is running in TCP Client mode. This option sets the waiting time for TCP Client to reconnect to TCP Server after the connection is disconnected. The default value is 0, which means immediate reconnection; The value range is from 0 to 60000, in milliseconds;

Inactivity Time(ms): The time interval for timeout disconnection is valid when the module is running in TCP mode, with a value range of 0~60000, unit: ms, default:0 (disable this function);

Keep Alive Time (5s): Time to check the connection is it existed, valid when the module is running in TCP mode, with a value range of 0-65536 and a unit of 5s. Default:0 (disable this function);

Clear Buffer if Connect: Is the serial port buffer cleared after establishing a connection? It is valid during TCP. If the connection suddenly disconnects during data exchange, some data in the serial port buffer may not have been sent. After the connection is reestablished, can the unfinished data in the serial port buffer be sent? This command can be used to handle this; After selecting, this function will be enabled and turned off by default;

Connect TCP Server when: The connection establishment condition is valid for TCP client, and can be set to "power on" to immediately establish the connection after power on (default) or "Serial data received" to establish the connection after receiving data through the serial port.

Security Options

Request Admin Password: The connection verification password function is valid when the module is running in TCP Server mode. If set to "Check Password", in TCP Server mode, when the client sends a connection request and establishes a connection, the client needs to send the module password to communicate with WIZ-IP32. Otherwise, the connection will be disconnected, and the default is "NO" (this function is turned off);

Auto Message (Fist Packet): After establishing the connection, send the message, which is valid for TCP. You can choose "NONE" to not send the message (default), "Send module Name" to send the module name, "Send MAC Address" to send the module MAC address, or "Send IP Address" to send the module IP address;

8.4 Firmware information

Click on "Firmware" to enter the firmware information page, as shown in the following figure, which has two parts:

Firmware information

Current Firmware Version: The current firmware version number of WIZ-IP32;

Firmware update

lease refer to Chapter 10.2:Remote firmware upgrade via web page for this section

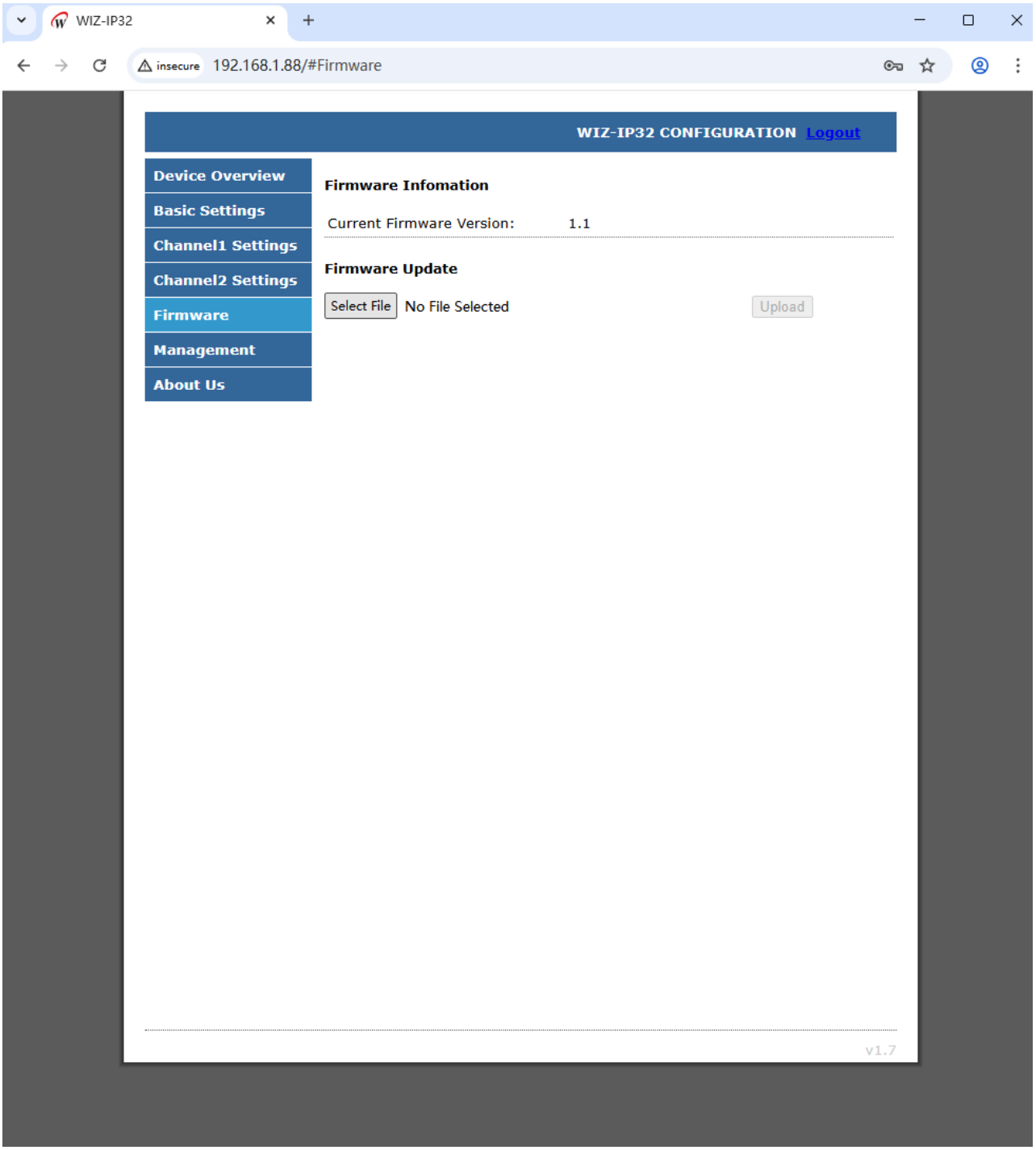


Figure 8-8 Firmware information page

8.5 Module management

Click on "Management" to enter the module management page, as shown in the following figure, which consists of the following two parts:

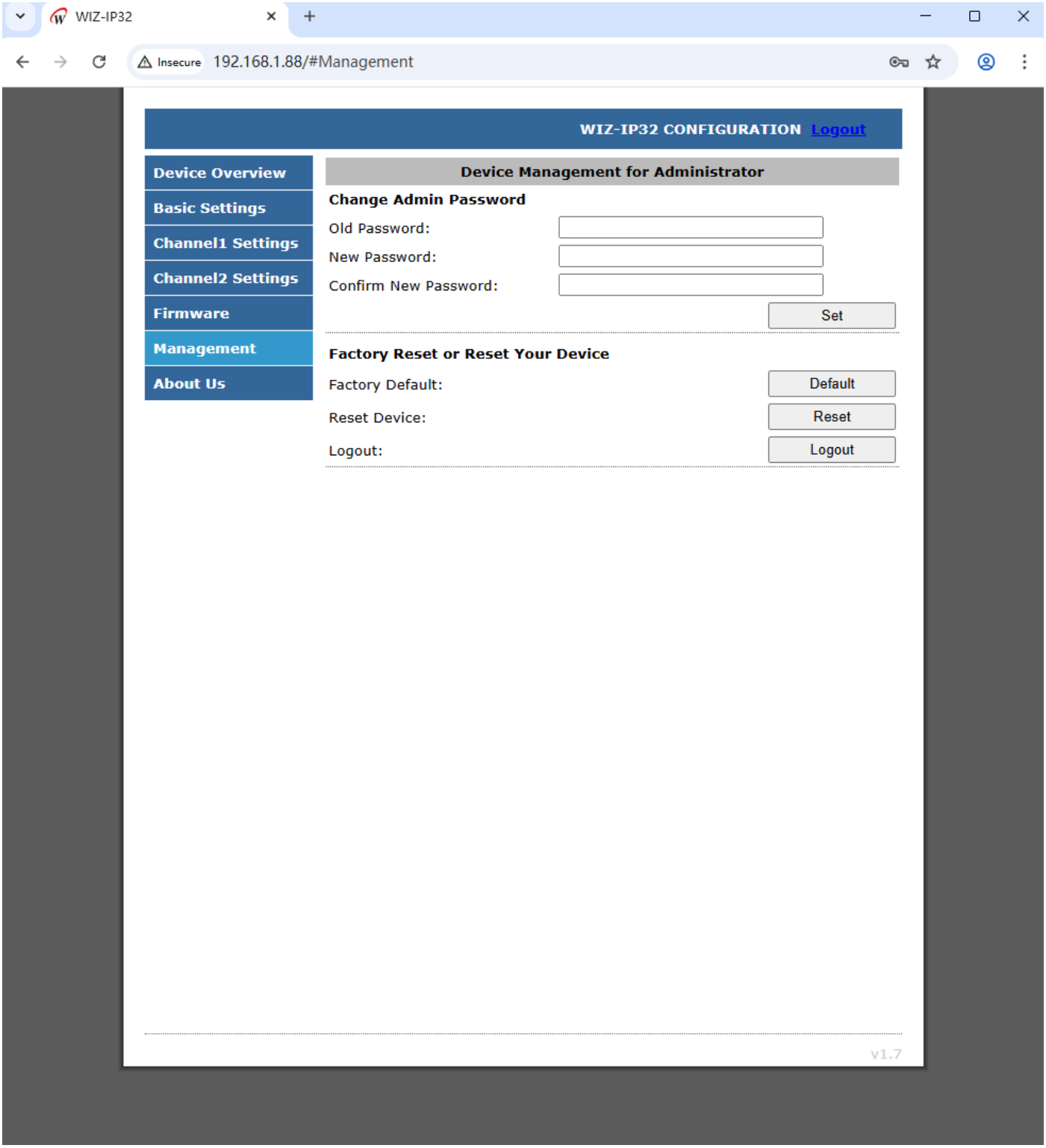


Figure 8-9 Module management page

Change admin password

Old Password: Original module password, default is admin;

New Password: The new module password, with a maximum length of 15 bytes, must be any combination of numbers, letters, or both, and cannot be set to empty;

Confirm Password: Confirm the password for the new module;

Set: Confirm the settings button. When the original module password is entered correctly and the new module password is correct, the page will prompt that the password has been successfully changed, as shown in the following figure. Clicking "OK" will redirect you to the login page.

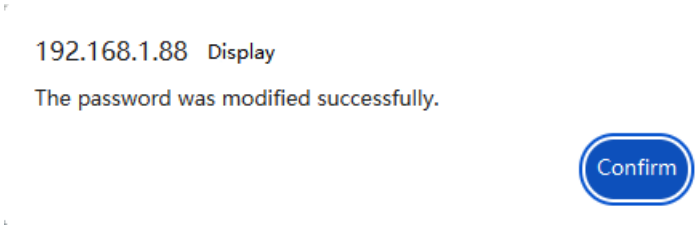


Figure 8-10 Web prompt - module password changed successfully

Factory reset or reset your device

Factory Default: The "Restore Factory Settings" button prompts whether to restore the module to factory settings, as shown in the following figure. Click "OK" to execute the factory settings restoration and redirect to the login page;

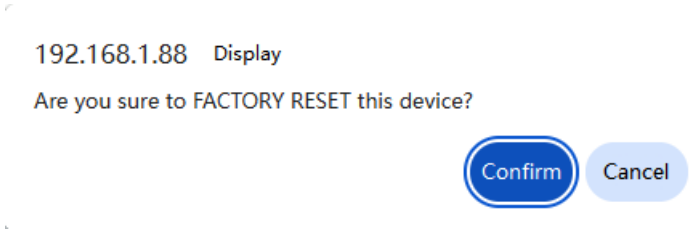


Figure 8-11 Web prompt - restore factory settings

Reset module: Reset module button;

Logout: Exit login button.

9 Restore factory settings

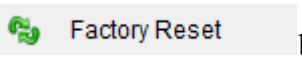
The WIZ-IP32 supports three methods for restoring factory settings: using the configuration software, using AT commands, and using the web page.

9.1 AT command mode

Please refer to Chapter 7 for the "DEFAULT" command.

9.2 Upper computer software mode

Firstly, click the left mouse button in the module list of WIZS2E Config Tool software to select the module that needs to be restored, and then click the 

→ 

button in the toolbar to factory Reset.

9.3 Web page format

Please refer to Chapter 8.5 for an introduction to the module management interface.

10 Firmware upgrade

WIZ-IP32 supports two types of firmware upgrades: Configuration tool firmware upgrade and web remote firmware upgrade, both of which are very convenient. The following will explain each of these two methods separately.

Note: The firmware referred to below must be the bin firmware of WIZ-IP32 provided by WIZnet HK officially.

10.1 Firmware upgrade by WIZS2E ConfigTool

Firstly, modify the WIZ-IP32 that requires firmware upgrade to have the same IP address as the computer in the same network segment. Open the serial port debugging tool for real-time viewing of printed debugging information for the serial port. After searching for WIZ-IP32, click the "Upload Firmware" button and select the firmware, as shown in the following figure.

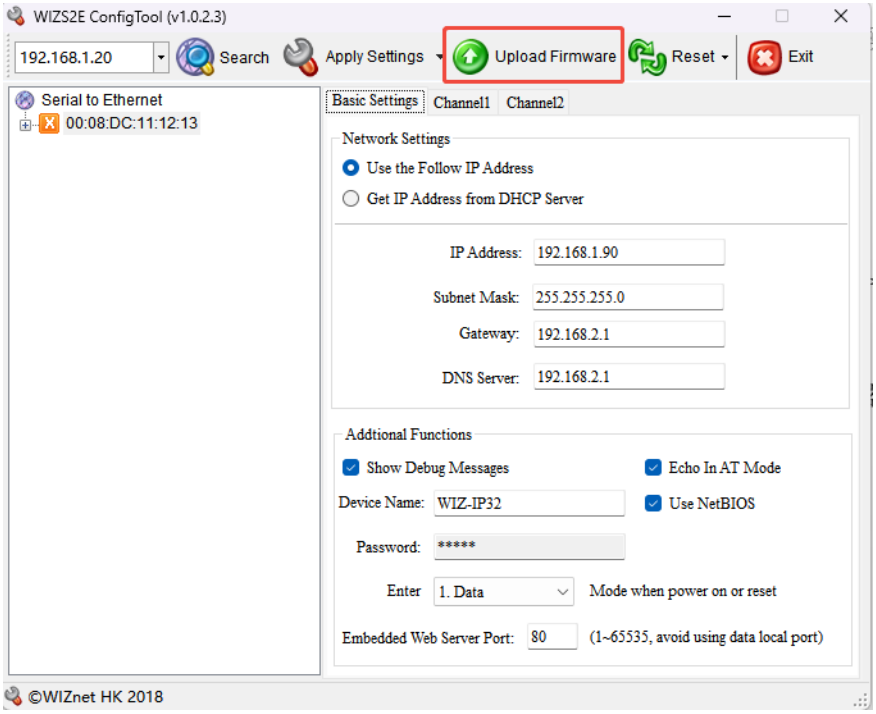


Figure 10-1 Firmware upgrade via WIZS2E ConfigTool

As shown in the figure below, the firmware upgrade is complete.

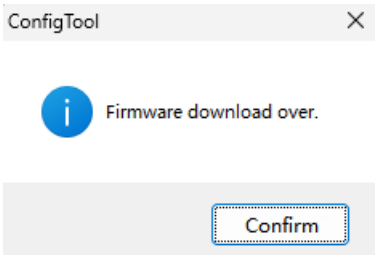


Figure 10-2 Upper computer prompt - firmware upgrade completed

10.2 Firmware upgrade by web page

After logging into the WIZ-IP32 webpage, enter the "Firmware" page as shown in the figure below, click the Choose File button, select the firmware to be updated, and click the Upload button to start updating the firmware.

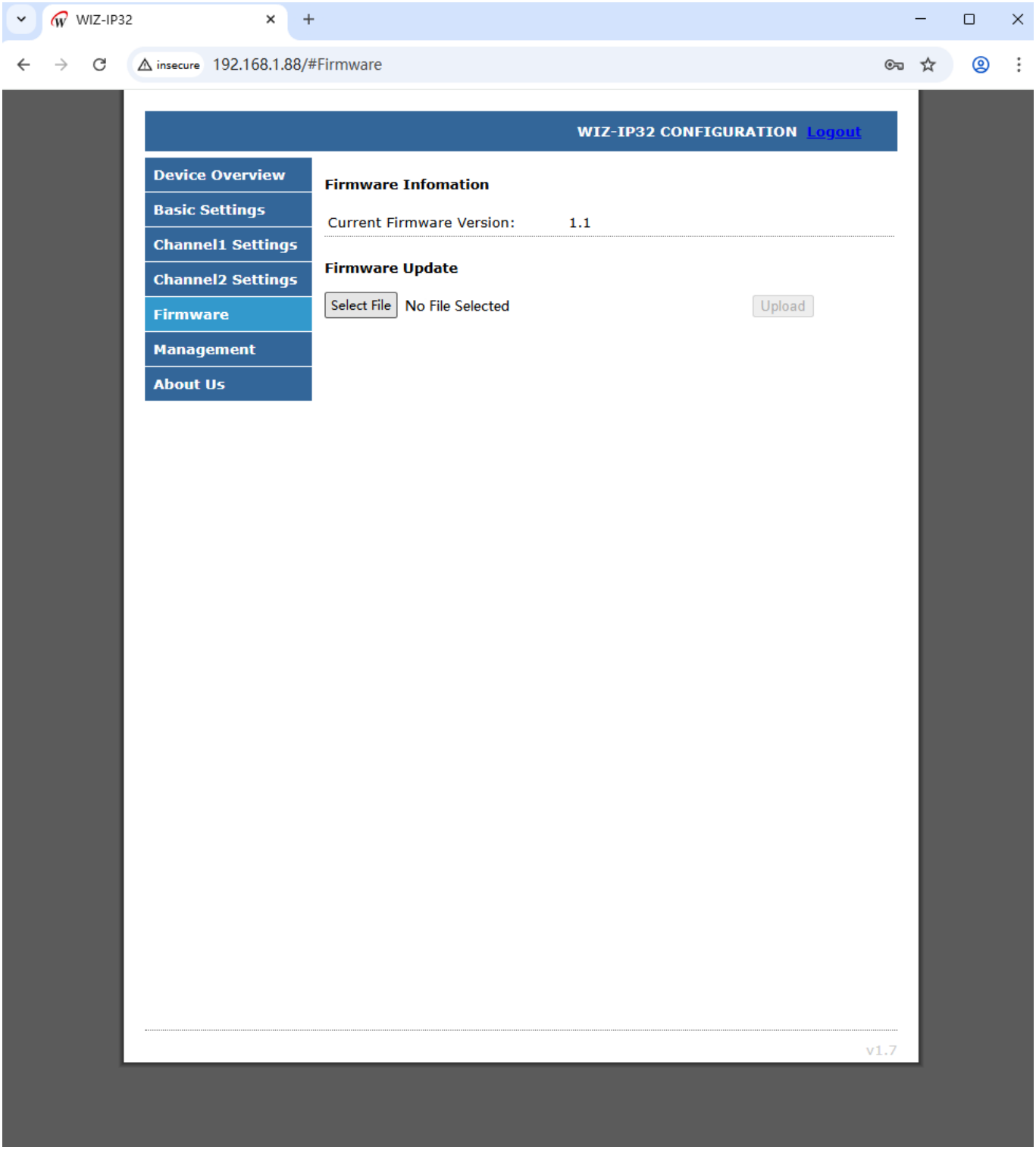


Figure 10-3 Firmware upgrade via Web interface

After the firmware upgrade is completed, WIZ-IP32 will automatically restart to complete the firmware update operation, and the webpage will automatically redirect to the login interface.

11 Appendix

11.1 Default list of occupied ports in Ethernet protocol

Protocol	Port
Reserve	0
TCP port multi-channel server	1
Reserve	2
ECHO	7
Reserve	9
Reserve	11
Reserve	13
Network Status	15
FTP	20
FTP	21
TELNET	23
SMTP	25
Printer	35
Time server	37
Name server	42
Reserve	43
Login Host Protocol	49
DNS	53
DHCP	67
DHCP	68
TFTP	69
Gopher	70
Finger	79
HTTP	80
Remote TELNET	107
SUN	111
NNTP	119
NTP	123
SNMP	161
SNMP	162
IPX	213
Reserve	160~223

Product Repair Procedure

WIZnet HK promises to provide a one-year free warranty service for WIZ-IP32 equipment. Within one year from the date of purchase of WIZ-IP32 equipment, if there are any product quality problems during normal use, users can repair them through the following procedures:

1 Provide proof of purchase.

2 Obtain repair permit from WIZnet HK or distributor.

3 Fill out the product problem report form and provide as much detailed information as possible about the reason for repair and the symptoms of the malfunction, in order to reduce repair time.

4 Pack the equipment, send it to the designated repair address, and attach the product problem report form.

The following are not covered by the warranty. For damages caused to the product that are not covered by the warranty, we will only charge the cost of the components appropriately:

1 Damage to the product caused by human or force majeure factors.

2 Unable to provide proof of product purchase.

3 products with a warranty period exceeding one year.

Product Problem Report Form

Corporate name			
Contacts		Contact Number	
Email		purchasing date	
PRODUCT MODEL		serial number	
notes			

Problem description:(Please provide a detailed description of the problem that occurred and list all the error messages you have seen in detail)

Disclaimers

This document provides information about WIZnet HK products. This document does not grant any intellectual property license, whether express or implied, or by prohibiting speech or other means. WIZnet HK shall not be liable for any other responsibilities except those stated in its product sales terms and conditions. Furthermore, WIZnet HK makes no express or implied warranties with respect to the sale or use of its products, including warranties of fitness for a particular purpose, merchantability, or liability for infringement of any patent, copyright, or other intellectual property rights. WIZnet HK products are not designed for medical, life-saving, or life sustaining purposes. WIZnet HK may make modifications to product specifications and descriptions at any time without prior notice.

WIZ-IP32 modules are high-tech products that may contain certain design defects or undiscovered errors. Once discovered, they will be included in the errata and may result in differences between the product and published specifications. If requested by the customer, the latest errata sheet can be provided.

Before ordering products, please purchase from WIZnet HK or designated distributor through the contact information provided on the "Sales and Service" page of this document and obtain the latest specifications.

For more relevant information, please visit the official website of WIZSE S2E at: https://wizse.com/

Declaration

Application Information

The cases or intentions in this application information are hypothetical and only intended to help users familiarize themselves with the product's features and usage methods. Customers must make modifications and validate based on the characteristics of their product before developing it.

The right to modify documents

WIZnet HK reserves the right to modify this document at any time without prior notice.