

Modbus POE ETH Relay

Overview

Hardware Description

Hardware Connection

- Connect the Modbus POE ETH Relay to the LAN via a network cable, and supply power through the power port or the POE.



Note: The above figure is powered by a PoE Ethernet port. If it is an ordinary network cable, it must be powered by an external power adapter, which can be powered by an external power supply in the range of 7~36V.

Electrical and Relay Safety Instructions

- This product must be operated by professional electricians or qualified personnel. During use, ensure electrical safety, leakage protection, and proper insulation.
- Before installing, maintaining, or replacing the relay device, always turn off the power and unplug the device.

- Do not attempt to disassemble the relay device to avoid damage or the risk of electric shock.
- Properly install and place the relay device. Do not use it in humid, overheated, flammable, or explosive environments to prevent accidents caused by improper installation or use.

1. Load Matching

- Ensure the relay's rated voltage and current match the load. Do not exceed the rated capacity.
- For inductive loads (motors, coils, lamps, etc.), the starting current may be much higher than the rated current. Choose a relay with sufficient current margin.

2. Short Circuit and Overcurrent Protection

- Install a **fuse** or **circuit breaker** in the relay circuit to prevent damage due to short circuits or accidental overcurrent.
- Ensure the load circuit has no short circuits during wiring, and select protection components with appropriate current ratings if necessary.

3. Arc and Switching Protection

- Relay switching generates arcs, which can cause contact wear or welding.
- For inductive loads, it is recommended to use **RC snubber circuits** or **varistors** for arc suppression.

4. Installation Environment

- Do not use the relay in humid, high-temperature, flammable, explosive, or dusty environments.
- Install the relay securely to avoid vibrations or shocks that may cause misoperation or damage.

5. Power-Off Operation

- Always cut off power before maintenance, wiring, or replacing the relay to ensure personnel and device safety.
- Latching relays are only powered when changing state. Avoid strong vibrations or strong magnetic fields while the relay is unpowered.

6. Status Confirmation

- After powering on, confirm or reset the relay status as needed to prevent abnormal operation caused by transportation, installation, or external disturbances.
- Avoid power interruption during relay operation to prevent uncertain status or contact damage.

7. Regular Inspection

- Periodically inspect relay contacts, terminals, and insulation to ensure proper operation.
- If abnormal heating, odor, or burn marks are detected, immediately cut off power and replace the relay.

Indicator Light Description

Indicator	Status description
RUN indicator	Ethernet port running indicator, outputs a square wave with a period of 2 seconds when the Ethernet port working normally.
STA indicator	MCU indicator, blinking when the MCU working normally.
TXD indicator	Send indicator, lights up when sending data.
RXD indicator	Receive indicator, lights up when receiving data.
Green indicator on Ethernet port	The green indicator will be on when TCP connection is established, which can be used to determine whether the module has established a communication link with the host software.
Yellow indicator on Ethernet port	Data activity indicator, when data is transmitted through the Ethernet port, the yellow indicator changes its state, which can be used to determine if there is data transmission

Module Parameter Configuration

The module needs to set the module parameters before communication, such as IP address, serial port format, Modbus protocol, etc. There are two modes of setting parameters: Vircom software configuration and web configuration.

Vircom software configuration allows for setting more parameters, but requires software installation. Web configuration does not require installation, but you need to know the IP address first, and the configuration parameters are few. It is recommended to use Virom for configuration.

Note:

1. The configuration can be done in any way, and it is recommended to use Virom software for first test.

2. It is recommended to modify only the IP address for the first configuration, other parameters are not recommended to be modified. The serial port parameters must be the default parameters; modifying the serial port parameters will result in no communication.
3. The module supports both Modbus RTU and Modbus TCP protocols. In the Advanced Settings -> Transfer Protocol, you can choose "None", which means the Modbus RTU protocol. It is not recommended to modify during the first configuration.
4. The selected Modbus TCP protocol must be configured using the Virom software and set to a non-storage Modbus gateway, otherwise the communication will not be normal.

Virom Software Mode Configuration

General Settings

Connect the module to the hardware and connect it to the network. Run the [VirCom](#) software (the computer on which Vircom is installed must be on the same LAN as the module).

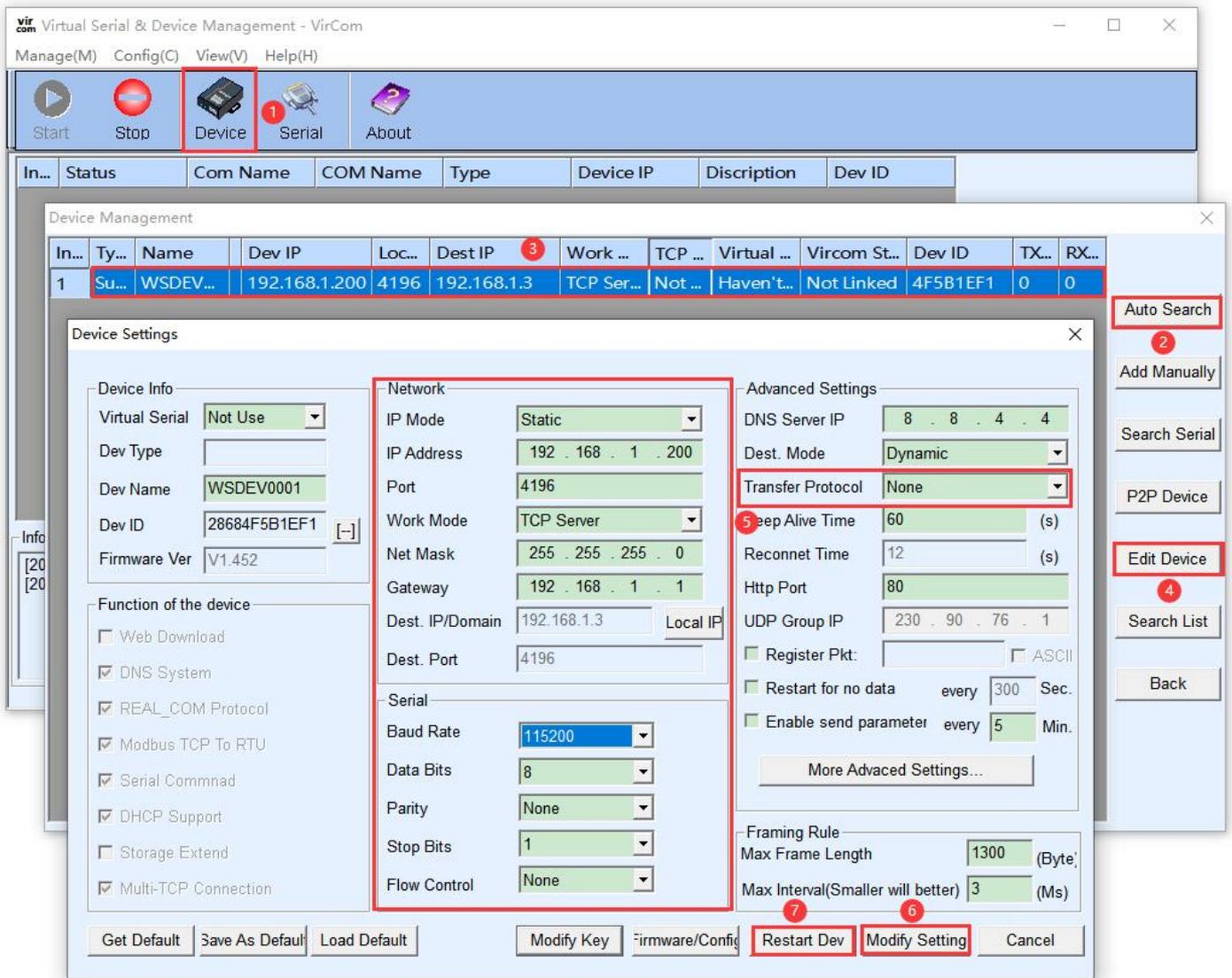
The operation is as follows:

- 1. Click `Device`
- 2. Click `Auto Search`
- 3. Software search recognizes the device connected to the LAN
- 4. Select the device, and then click `Edit Device` or double-click the searched device directly
- 5. Set up the device parameters:
 - Modify the "IP mode" to a static assigned address, set the IP address, note that the static IP address entered must not be used by other devices, and it needs to be on the same LAN as the computer.
 - The working mode is TCP server. The serial port setting is 115200 by default and cannot be modified.
 - The "Transfer Protocol" in "Advanced Settings" defaults to "None", which means using the Modbus RTU protocol; if you select "Modbus TCP protocol", then use the Modbus TCP communication protocol.
 - Click on "More Advanced Settings..." and select the Modbus Gateway Type as a non-storage Modbus gateway.
- 6. Once the settings are complete, click `Modify Setting`

- 7. Click **Restart Dev**, wait for the module to restart, and the new settings will take effect.

Note: It is recommended to modify only the IP address for the first configuration, and do not modify other parameters.

See the figure below for details:



Note: The default Modbus gateway type is storage type, which will automatically send query commands several times, which may cause the controller chip to fail to respond, resulting in no response to the query commands. Therefore, you need to set it as

Multi-host non-storage type.

More Advanced Settings

WIFI Settings

WIFI Work Mode: AP

AP/STA SSID: [Empty]

Encrypt Type: No Encrypt

AP/STA Key: [Empty]

AP Mode Channel: 4

DHCP Server: Enable

RJ45 WIFI Relay: Disable

The Advanced Functions Supported

- IO Port Control
- UDP Group
- Multi-Dest IP
- Proxy Function
- SNMP Function
- P2P Function

RS485 Multi-Host Support Settings

Modbus Gateway Type: Multi-host non-storage type

Enable RS485 Multi-Host

Maximum wait time of RS485 bus for 163 ms(0-8192)

Enable RS485 bus conflict detection

Send data only when RS485 bus is idle for 20 ms

Function Selection

- Modify setting will need key
- Enable receive broadcast
- Enable P2P
- Send MAC when TCP establish
- Detect net using Ping
- No clear serial buffer when link
- Need key when TCP establish

Multi Dest-IP And Port

IP Address or DNS name	Port	Type
0	0	Client Dest.
0	0	Client Dest.
0	0	
0	0	
0	0	
0	0	

OK Cancel

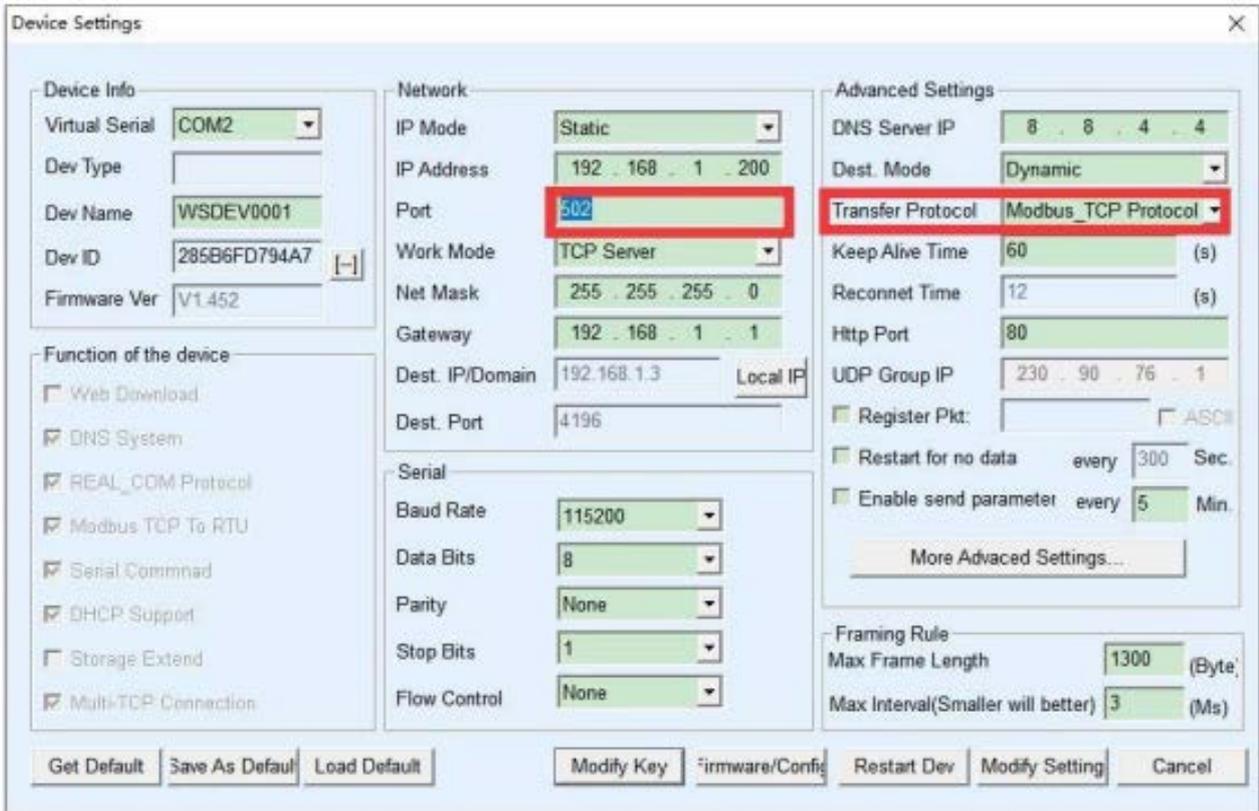
Protocol Setting

Note: It is recommended to use the default Modbus RTU protocol for the first configuration and no modifications are needed.

Although the module transmits data through the network port, it supports two Modbus protocols: Modbus RTU and Modbus TCP. By default, data is transparently transmitted, i.e. using the Modbus RTU protocol.

Modbus TCP Protocol Settings

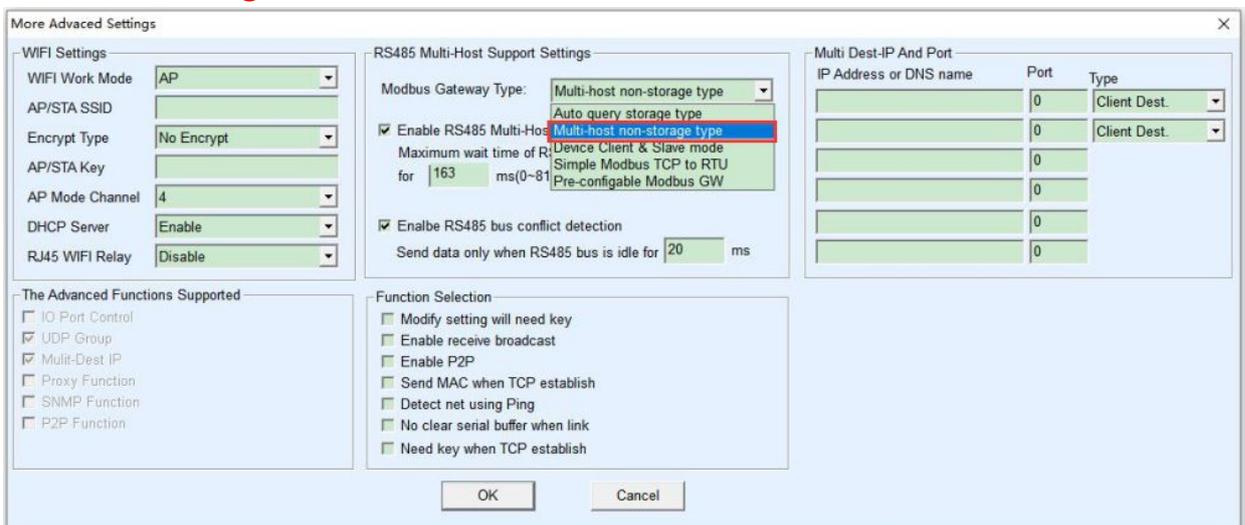
- The "Transfer Protocol" in the "Advanced Settings" can be set to "Modbus TCP protocol". In this case, the Modbus RTU protocol of the main controller will be converted to the Modbus TCP protocol and transmitted through the network port.
- In this case, the device port automatically changes to 502. Users can use the Modbus TCP tool to connect to the IP port 502 of the serial port server.
- Click on "More Advanced Settings..." and select the Modbus Gateway Type as a non-storage Modbus gateway.



Modbus RTU Protocol Settings

- Set "Transfer Protocol" in the "Advanced Settings" to "None", and change to use Modbus RTU protocol.
- Click on "More Advanced Settings..." and select the Modbus Gateway Type as a non-storage Modbus gateway.

Note: The default Modbus gateway type is storage type, which will automatically send query commands several times, which may cause the controller chip to fail to respond, resulting in no response to the query commands. Therefore, you need to set it as Multi-host non-storage



type.

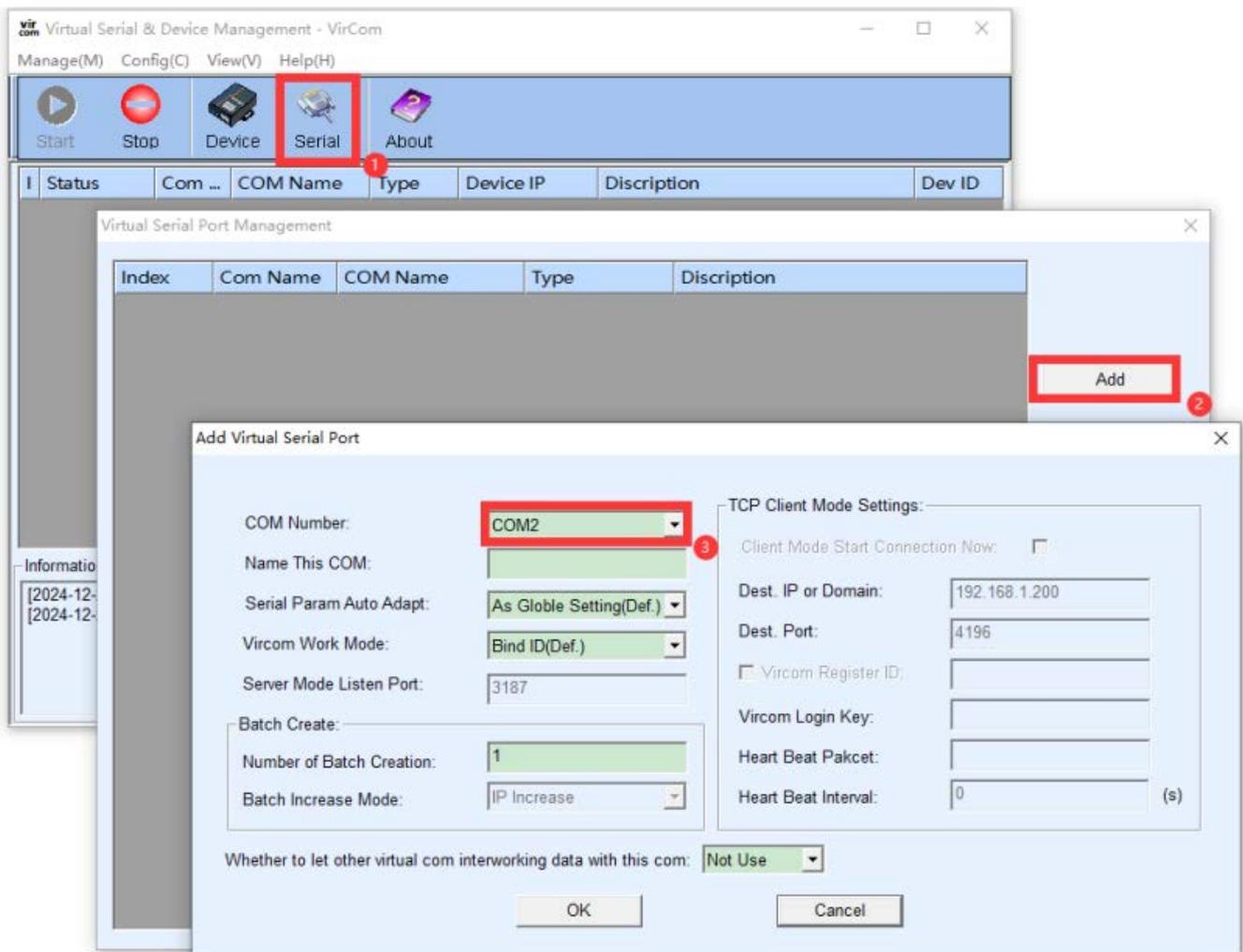
Virtual Serial Port Setting

The module transmits data through a network port (TCP/UDP protocol). In order to enable users to use the PoE port communication even with developed serial port software, a virtual serial port needs to be added. If not needed, this part can be skipped.

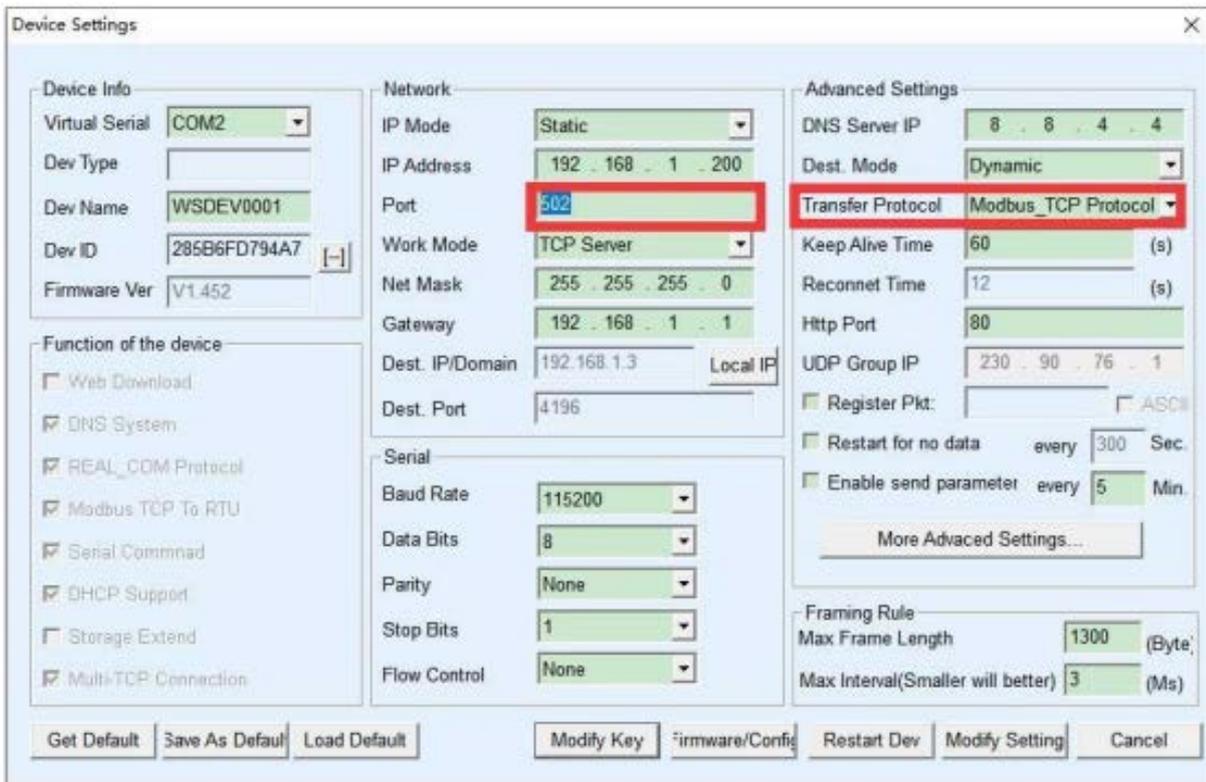
- First, install the virtual serial driver [Virtual serial port driver](#)
- Run Vircom and the user program on the same computer.
- Vircom creates a virtual COM port and connects this COM port to the serial server. When the user program uses the COM communication, it can send data to the user's serial port device through the Vircom serial port server.

The following steps demonstrate this operation:

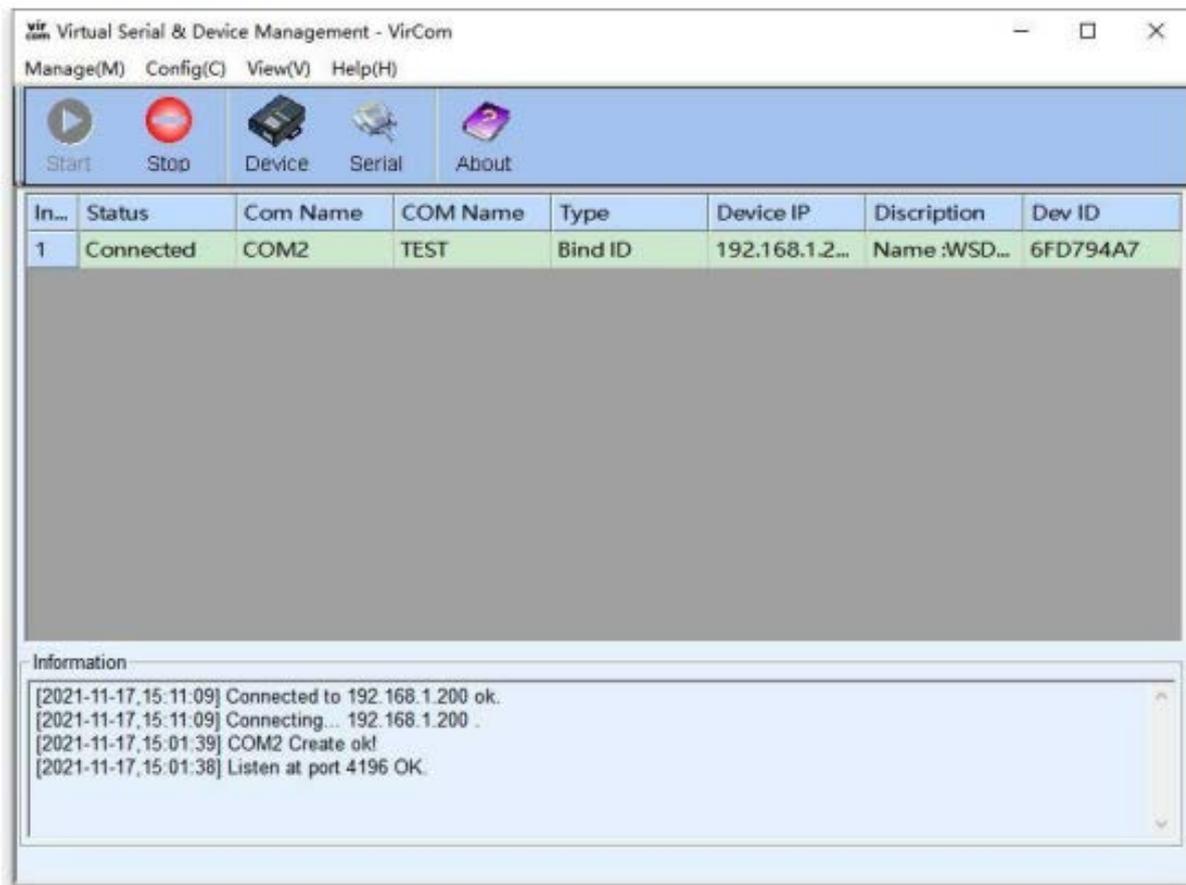
- Click on "Serial Port & Device Management" on the Vircom main interface, then click "Add" and select to add COM2 (Among them, COM2 is the newly emerging COM port on the computer).



- Then enter the device management and double-click the device that needs to be bound to COM2. As shown in the diagram, select COM2 from the Virtual Serial Port list in the top left corner. Then click on "Modify Setting" and then click on "Restart Device".



- Return to the main interface of Vircom. It can be seen that COM2 has been connected to the device whose IP is 192.168.1.200. In this case, the virtual serial port COM2 can be used instead of the network port for communication.

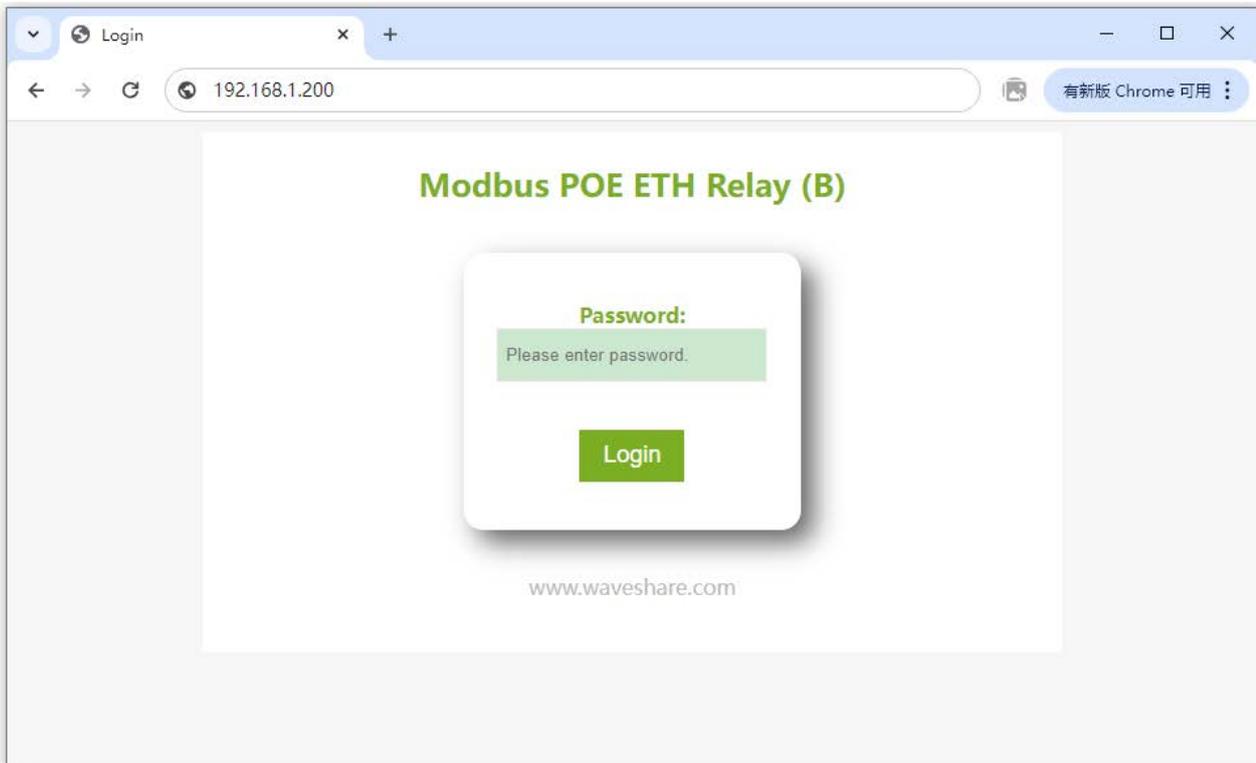


WEB Configuration

Using Vircom, you can search for and configure device parameters in different network segments. For Web configuration, you must first ensure that the computer and the serial server are in the same IP segment, and you need to know the IP address of the serial server in advance.

But Web configuration can be done on any computer without Vircom. (Different products have different web interfaces, which can be switched between Chinese and English)

1. Enter the IP address of the serial server in the browser, such as <http://192.168.1.200> to open the following web page



2. In the Password field, enter your password: The default login password is not set or is set to 123456. If no password is set, you can enter any password and click the Login button to log in. After setting the password to log in, the settings at "Modify Web Login Key" will take effect:

 [Logout](#) Chinese

Device Information

Device Name	WSDEV001	Firmware Version	V1.452	Device MAC	28-86-8C-F4-91-D3
-------------	----------	------------------	--------	------------	-------------------

Network Settings

Device IP	192.168.1.200	Device Port	4196	Device Web Port	80
Work Mode	TCP Server	Subnet Mask	255.255.255.0	Gateway	192.168.1.1
Destination IP/DNS	192.168.1.3	Destination Port	4196	IP mode	Static

Serial Settings

Baudrate	1200	Databits	8	Parity	None
Stopbits	1	Flow control	None		

Advanced Settings

No-Data-Restart	Disable	No Data Restart Time	300 second	5~1270	Reconnect-time	12	1~255 second
-----------------	---------	----------------------	------------	--------	----------------	----	--------------

Multi-Host Settings

Protocol	None	Instruction Time out	0	32~8000ms	Enable Multi-host	No
RS232/485/422 Conflict Time Gap	0					
	5~255ms					

NOTE: 1. Multi-host is always enabled when Protocol is Modbus TCP to RTU. 2. Time out is always 0 when Multi-host is disabled.
3. Time out only can be set as multiply of 32.

Modify Web Login Key

New Key		Input Key Again	
---------	--	-----------------	--

[Submit](#)

3. The serial server parameters can be modified on the web page that appears.

4. After modifying the parameters, click the "Submit" button.

Attention: The system has added webpage settings function by default when it leaves the factory. If the configuration interface page file is overwritten and the webpage cannot be opened, the webpage file needs to be downloaded again.

Please refer to [RS485 TO ETH \(B\) Manual](#)

Example Demonstration

The demo shows how the following two software operate.

SSCOM serial port debugging assistant is more convenient to operate, free of installation, and more convenient for complete display and analysis of instructions, but the disadvantage is that the data is not intuitive.

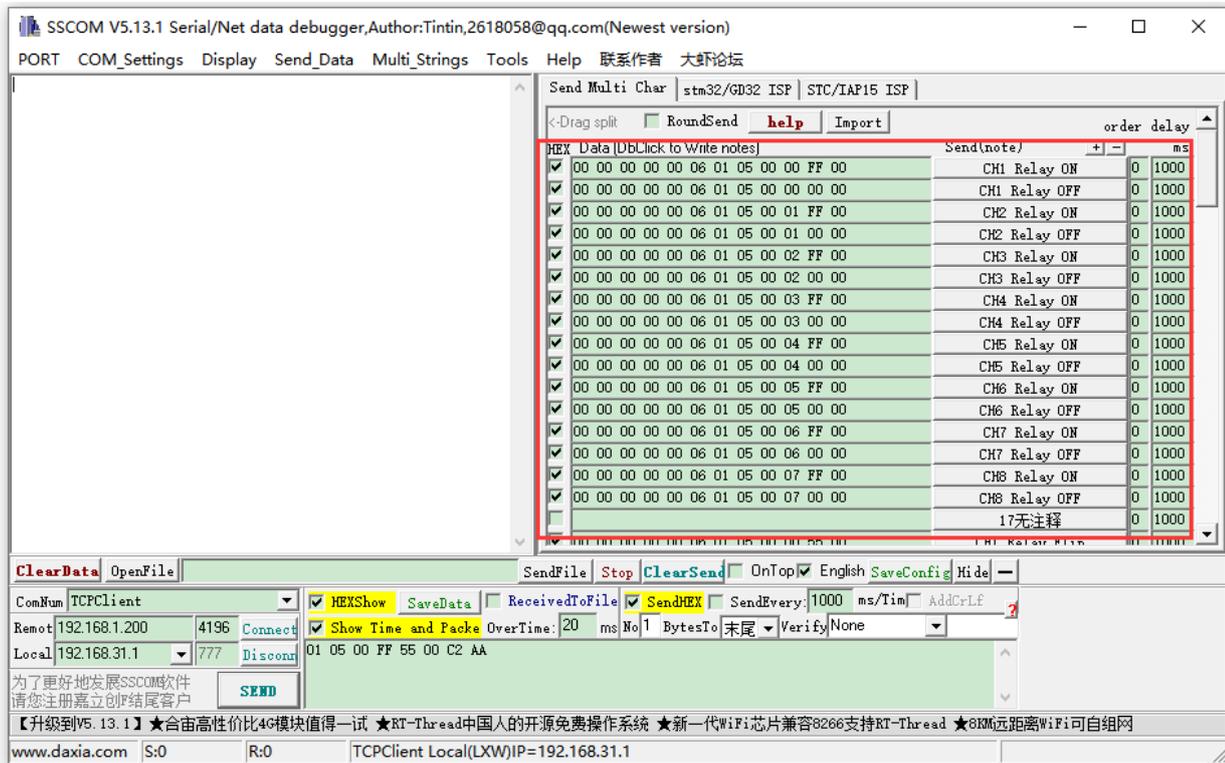
Modbus Poll software is directly operated on the register, and the data display is more convenient to observe, but the disadvantage is that the instruction is not displayed completely, so you need to be familiar with the Modbus register operation.

You can test using any method. It is recommended to use the SSCOM serial port debugging assistant software for the first test.

SSCOM Serial Port Debugging Assistant

Modbus RTU Command: The default configuration is the Modbus RTU command

- 1. Open the serial port debugging assistant window
- 2. Select TCPClient for port number
- 3. Modify the remote IP and port number according to the Vircom settings above
- 4. Click the "Connect" button to connect to the TCP server
- 5. The green light of the network port will light up when the connection is successful
- 6. Click Multi-Char to open the Send Multi-Char window, the default display is the Modbus RTU command, click the corresponding function to send the corresponding command.
- 7. If you use the custom input box below to send the command, you need to set Verify as ModbusCRC16

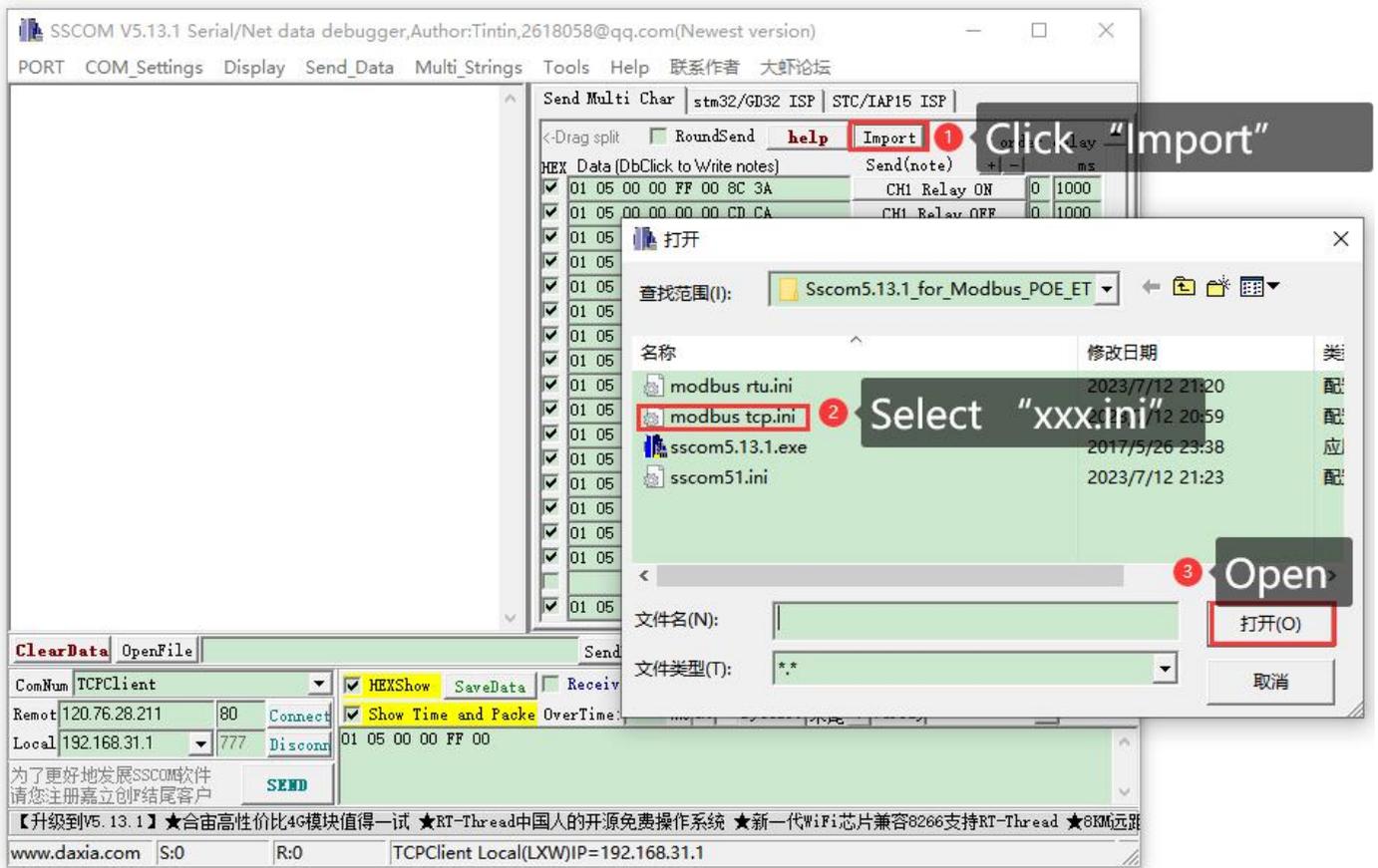


Configure Modbus TCP Directives: If you want to set it as a Modbus TCP Directive, you need to change the commands

- 1. Click on the Import ini button in the Send Multi-Char column
- 2. Select the modbus tcp.ini file to import the Modbus TCP command

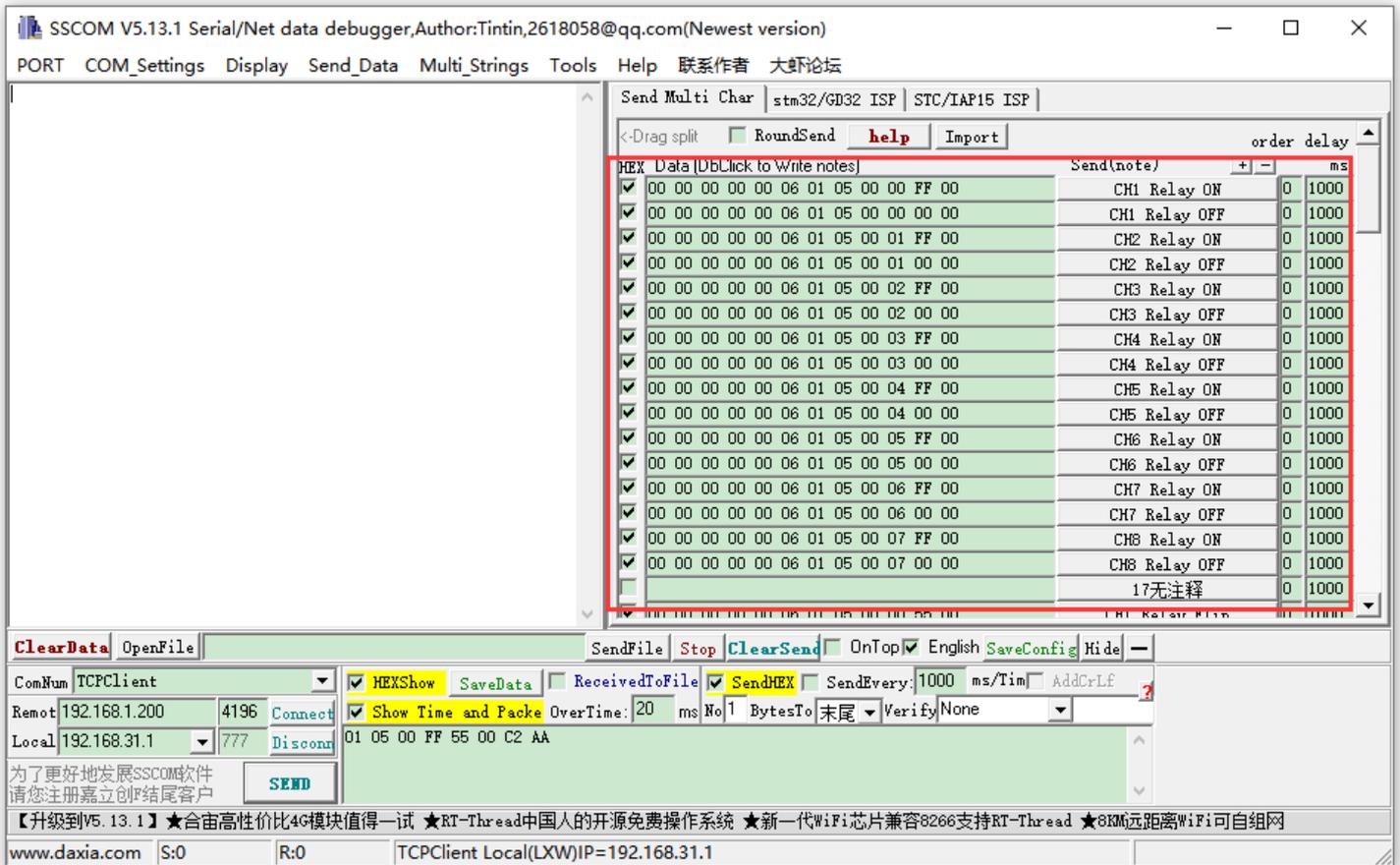
Note: If a popup error message says "A component named HEX0 already exists", then you need to close and reopen the software, which will reload the files and refresh the

buttons.



- 3. After successful import, the following is displayed, click on the function to send the corresponding command.

Note: Modbus tcp does not require CRC checksum, select None for Verify.

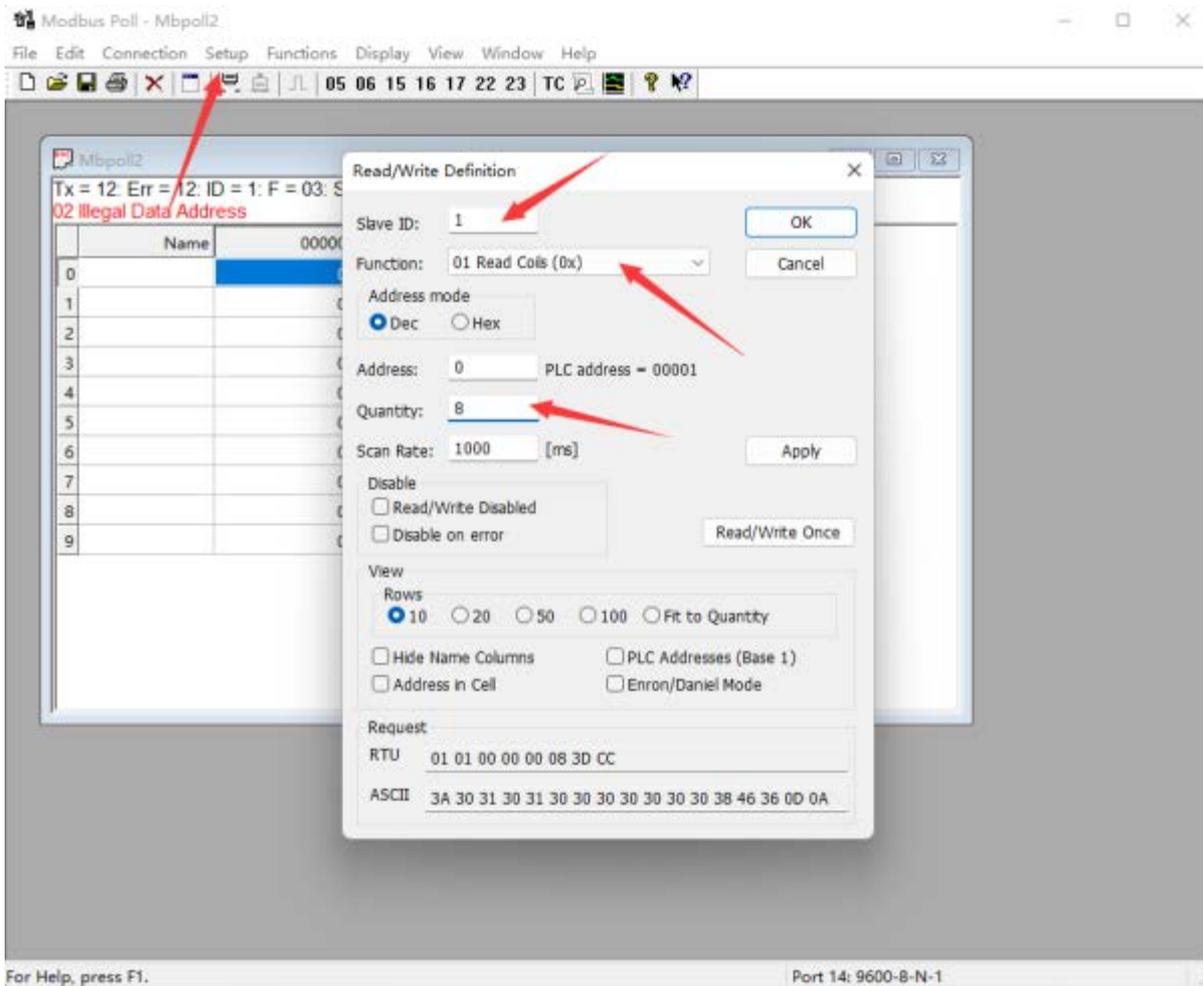


- For detailed Modbus commands, please see the development protocol.

Modbus Poll Software

It is not convenient to use the SSCOM software for observing the data, you can select [Modbus Poll software](#) to read the data. Download and install the Modbus Poll software.

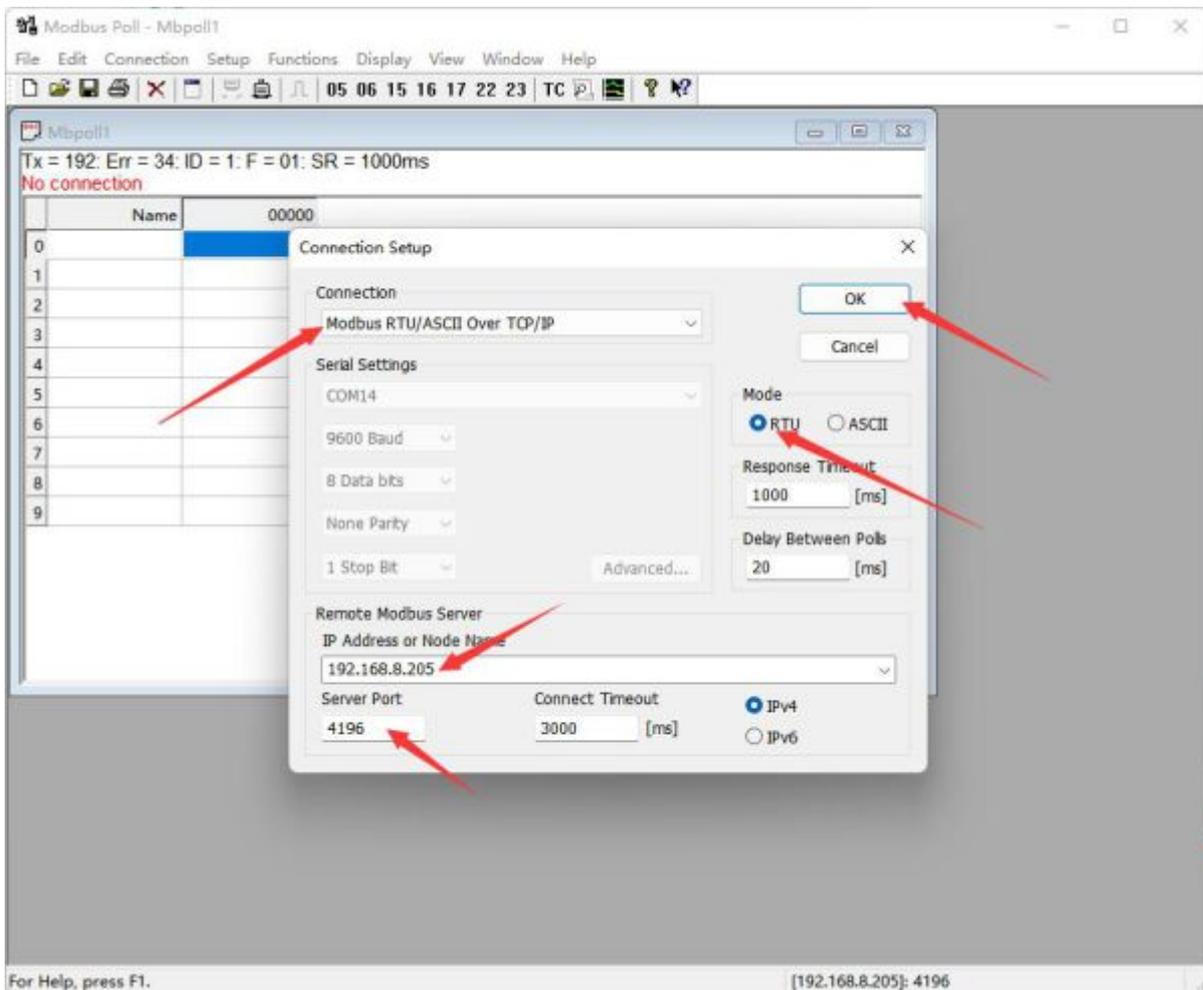
1. Open Modbus Poll software
2. Select Setup->Read/Write Definition, select the actual device address for Slave ID, select 01 Read Coils function code for Function, and change Quantity to 8 channels. Click OK to confirm.



• 3.

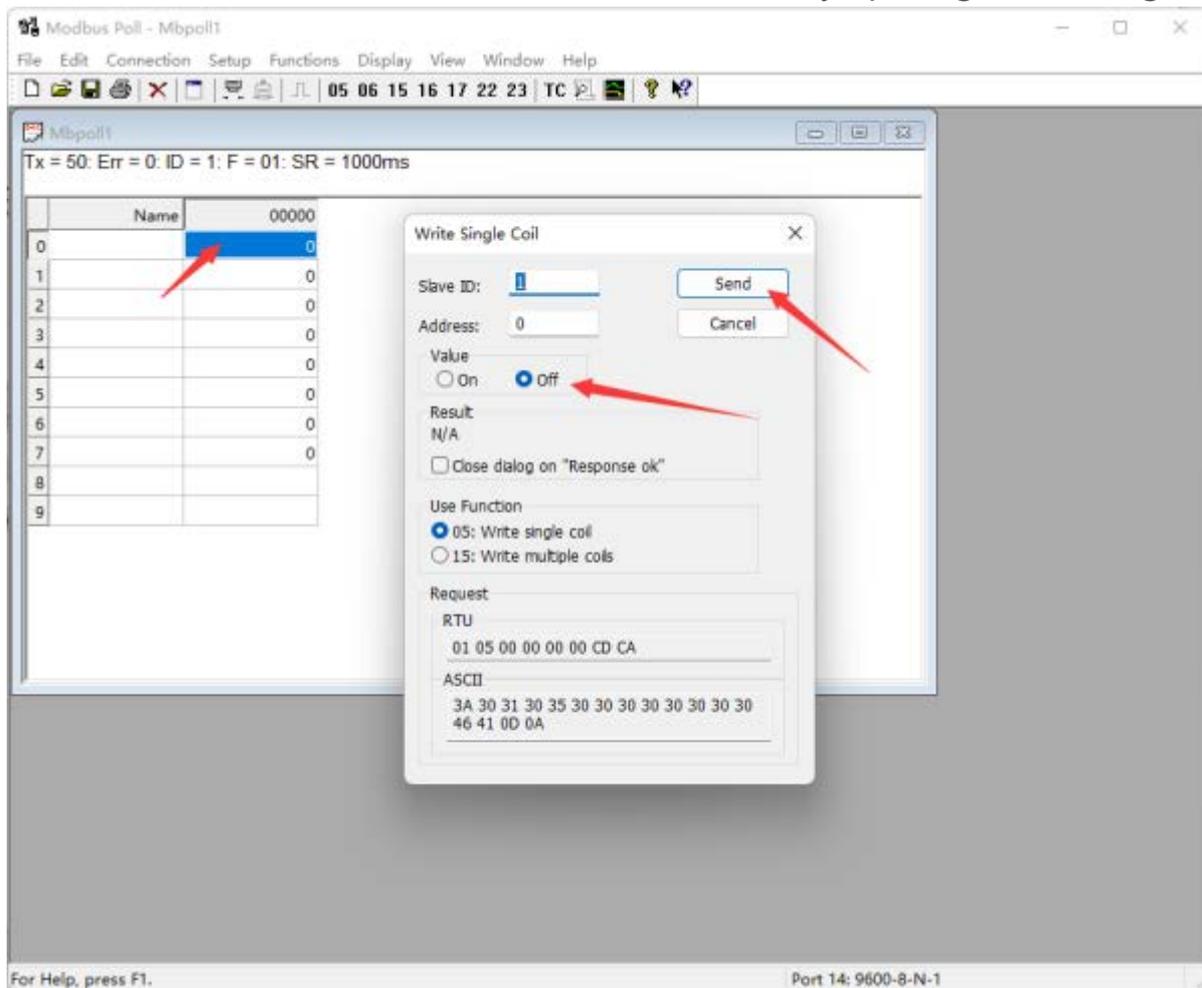
--->If you are using the Modbus RTU protocol, select Connection->Connect Setup, select Modbus RTU/ASCII Over TCP/IP for Connection, select RTU for Mode, and enter the correct IP address and port number. Click OK to connect.

--->If you are using the Modbus TCP protocol, select Connection->Connect Setup, select Modbus TCP/IP for Connection, and enter the correct IP address and port number. Click OK to connect.



4. After the connection is normal, you can check the current relay status. Select the corresponding channel, then double-click the status value to pop up the send page.

Choose On or Off, then Click Send to control the relay opening and closing.



Demo

Raspberry Pi

Connect the Raspberry Pi and the ModBus POE ETH Relay module to the same LAN. Open the Raspberry Pi terminal and run the program by entering the following command:

```
sudo apt-get install unzip

wget https://files.waveshare.com/wiki/Modbus-POE-ETH-Relay/Modbus_POE_ETH_Relay_Code.zip

unzip Modbus_POE_ETH_Relay_Code.zip

cd Modbus_POE_ETH_Relay_Code

#modbus rtu protocol

vi modbus_rtu.py #Change the IP address and port number according to the actual situation

sudo python3 modbus_rtu.py
```

```
#modbus tcp protocol

vi modbus_tcp.py #Change the IP address and port number according to the actual
situation

sudo python3 modbus_tcp.py
```

Note: To run this demo, you need to modify the demo file to change the IP address and port number to the actual IP address and port number of the ModBus POE ETH Relay.

Modbus RTU Development Protocol V2

Function Code Introduction

Function Code	Description	Note
01	Read coil status	Read relay status
03	Read holding register	Read the address and version
05	Write single coil	Write single relay
06	Write single register	Set the baud rate and address
0F	Write multiple coils	Write all relays

Register Address Introduction

Address (HEX)	Address storage content	Register value	Permission	Modbus Function Code
0x0000 0x0007	Channel 1~8 relay address	0xFF00: relay on 0x0000: relay off 0x5500: relay toggle	Read/Write	0x01, 0x05, 0x0F

0x00FF	Control all relays	0xFF00: all relays on 0x0000: all relays off 0x5500: all relays toggle	Write	0x05
0x0100 0x0107	Channel 1~8 relay toggle address	0xFF00: relay toggle 0x0000: relay unchanged	Write	0x05, 0x0F
0x01FF	Control all relays toggle	0xFF00: all relays toggle 0x0000: all relays unchanged	Write	0x05
0x0200 0x0207	Channel 1~8 relay flash on	Interval time: data*100ms Value: 0x0007, Interval time: 7*100MS = 700MS	Write	0x05
0x0400 0x0407	Channel 1~8 relay flash off	Interval time: data*100ms Value: 0x0007, Interval time: 7*100MS = 700MS	Write	0x05
4x4000	Device Address	Directly store Modbus address Device address: 0x0001	Read	0x03
4x8000	Software Version	Converting to decimal and then shifting the decimal point two places to the left will represent the software version 0x0064 = 100 = V1.00	Read	0x03

Operation Command Introduction

Control Single Relay

Send code: 01 05 00 00 FF 00 8C 3A

Field	Description	Note
01	Device Address	Fixed 0x01
05	05 Command	Relay control
00 00	Address	The register address of the relay to be controlled, 0x0000-0x0007

FF 00	Command	0xFF00: relay on; 0x0000: relay off; 0x5500: relay toggle
8C 3A	CRC16	The CRC16 checksum of the first 6 bytes of data

Return code: 01 05 00 00 FF 00 8C 3A

Field	Description	Note
01	Device Address	Fixed 0x01
05	05 Command	Relay control
00 00	Address	The register address of the relay to be controlled, 0x0000-0x0007
FF 00	Command	0xFF00: relay on; 0x0000: relay off; 0x5500: relay toggle
8C 3A	CRC16	The CRC16 checksum of the first 6 bytes of data

For example: [Address 1 device]:

```
Relay 0 on: 01 05 00 00 FF 00 8C 3A
Relay 0 off: 01 05 00 00 00 00 CD CA
Relay 1 on: 01 05 00 01 FF 00 DD FA
Relay 1 off: 01 05 00 01 00 00 9C 0A
Relay 2 on: 01 05 00 02 FF 00 2D FA
Relay 2 off: 01 05 00 02 00 00 6C 0A
Relay 3 on: 01 05 00 03 FF 00 7C 3A
Relay 3 off: 01 05 00 03 00 00 3D CA
Relay 0 toggle: 01 05 00 00 55 00 F2 9A
Relay 1 toggle: 01 05 00 01 55 00 A3 5A
Relay 2 toggle: 01 05 00 02 55 00 53 5A
Relay 3 toggle: 01 05 00 03 55 00 02 9A
```

Control All Relays

Send code: 01 05 00 FF FF 00 BC 0A

Field	Description	Note
01	Device Address	Fixed 0x01
05	05 Command	Relay control
00 FF	Address	Fixed 0x00FF
FF 00	Command	0xFF00: relay on; 0x0000: relay off; 0x5500: relay toggle
BC 0A	CRC16	The CRC16 checksum of the first 6 bytes of data

Return code: 01 05 00 FF FF 00 BC 0A

Field	Description	Note
01	Device Address	Fixed 0x01
05	05 Command	Relay control
00 FF	Address	Fixed 0x00FF
FF 00	Command	0xFF00: relay on; 0x0000: relay off; 0x5500: relay toggle
BC 0A	CRC16	The CRC16 checksum of the first 6 bytes of data

For example: [Address 1 device]:

```
All relays on: 01 05 00 FF FF 00 BC 0A
```

```
All relays off: 01 05 00 FF 00 00 FD FA
```

All relays toggle: 01 05 00 FF 55 00 C2 AA

Read Relay Status

Send code: 01 01 00 00 00 08 3D CC

Field	Description	Note
01	Device Address	Fixed 0x01
01	01 Command	Query relay status
00 00	Relay Start Address	The register address of the relay, 0x0000 - 0x0007
00 08	Relay Number	The number of relays to be read, which must not exceed the maximum number of relays
3D CC	CRC16	The CRC16 checksum of the first 6 bytes of data

Receive code: 01 01 01 00 51 88

Field	Description	Note
01	Device Address	Fixed 0x01
01	01 Command	Query relay status
01	Byte Number	The number of all bytes of the returned status information
00	Query status	Received relay status Bit0: the first relay status; Bit1: the second relay status; And so on, with the idle high bit being zero
51 88	CRC16	The CRC16 checksum of the first 6 bytes of data

For example: [Address 1 device]

```
Send: 01 01 00 00 00 08 3D CC
Receive: 01 01 01 00 51 88 //All relays off
Send: 01 01 00 00 00 08 3D CC
Receive: 01 01 01 01 90 48 //Relay 0 is on, others are off
Send: 01 01 00 00 00 08 3D CC
Receive: 01 01 01 41 91 B8 //Relay 0 and 6 are on, others are off
```

Write Relay Status

Send code: 01 0F 00 00 00 08 01 FF BE D5

Field	Description	Note
01	Device Address	Fixed 0x01
0F	0F Command	Write relay status
00 00	Relay Start Address	The register address of the relay to be controlled, 0x0000 - 0x0007
00 08	Relay Number	The number of relays to be operated, which must not exceed the maximum number of relays
01	Byte Number	The byte number of the status
FF	Relay Status	Bit0: the first relay status; Bit1: the second relay status; And so on, with the idle high bit being zero
BE D5	CRC16	The CRC16 checksum of the first 6 bytes of data

Receive code: 01 0F 00 00 00 08 54 0D

Field	Description	Note
-------	-------------	------

01	Device Address	Fixed 0x01
0F	0F Command	Control all registers
00 00	Relay Start Address	The register address of the relay to be controlled, 0x0000 - 0x0007
00 08	Relay Number	The number of relays to be operated
54 0D	CRC16	The CRC16 checksum of the first 6 bytes of data

For example: [Address 1 device]

```
All relays on: 01 0F 00 00 00 08 01 FF BE D5
All relays off: 01 0F 00 00 00 08 01 00 FE 95
0-1 on; 2-7 off: 01 0F 00 00 00 08 01 03 BE 94
```

Relay Flash ON/OFF Command

Send code: 01 05 02 00 00 07 8D B0

Field	Description	Note
01	Device Address	Fixed 0x01
05	05 Command	Single control command
02	Command	02: flash on, 04: flash off
00	Relay Address	The address of the relay to be controlled, 0x00~0x07
00 07	Interval Time	The interval time: data*100ms Value: 0x0007, Interval time: 7*100MS = 700MS The maximum setting for the flash-on flash-off time is 0x7FFF
8D B0	CRC16	The CRC16 checksum of the first 6 bytes of data

Receive code: 01 05 02 00 00 07 8D B0

Field	Description	Note
01	Device Address	Fixed 0x01
05	05 Command	Single control command
02	Command	02: flash on, 04: flash off
00	Relay Address	The address of the relay to be controlled, 0x00~0x07
00 07	Interval Time	The interval time: data*100ms Value: 0x0007, Interval time: 7*100MS = 700MS
8D B0	CRC16	The CRC16 checksum of the first 6 bytes of data

For example: [Address 1 device]

```
Relay 0 flash on: 01 05 02 00 00 07 8D B0 //700MS
Relay 1 flash on: 01 05 02 01 00 08 9C 74 //800MS
Relay 0 flash off: 01 05 04 00 00 05 0C F9 //500MS
Relay 1 flash off: 01 05 04 01 00 06 1D 38 //600MS
```

Read Software Version Command

Send code: 01 03 80 00 00 01 AD CA

Field	Description	Note
01	Device Address	Fixed 0x01
03	03 Command	Read Holding Register
80 00	Command register	0x8000: read software version
00 01	Byte Number	Fixed 0x0001

AD CA	CRC16	The CRC16 checksum of the first 6 bytes of data
-------	-------	---

Receive code: 01 03 02 00 C8 B9 D2

Field	Description	Note
01	Device Address	Fixed 0x01
03	03 Command	Read Holding Register
02	Byte Number	The number of bytes returned
00 C8	Software Version	Converting to decimal and then shifting the decimal point two places to the left will represent the software version 0x00C8 = 200 = V2.00
B9 D2	CRC16	The CRC16 checksum of the first 6 bytes of data

For example:

Send: 01 03 80 00 00 01 AD CA	
Receive: 01 03 02 00 C8 B9 D2	//0x00C8 = 200 =V2.00

Exception Function Code

When the received command is incorrect or the device is abnormal, an exception response will be returned in the following format:

Receive: 01 85 03 02 91

Field	Description	Note
01	Device Address	0x00 indicates the broadcast address, 0x01-0xFF indicates the device address
85	Exception Function Code	Exception function code = Request function code + 0x80

03	Byte Number	Exception Code
02 91	CRC16	The CRC16 checksum of the first 6 bytes of data

An exception code is a single-byte value that indicates the type of error. Several commonly used exception codes defined by the Modbus protocol:

Exception Code	Name	Description
0x01	Illegal Function	The requested function code is not supported
0x02	Illegal Data Address	The requested data address is incorrect
0x03	Illegal Data Value	The requested data value or operation cannot be executed
0x04	Server Failure	Server equipment failure
0x05	Response	The request has been received and is being processed
0x06	Device Busy	The device is currently busy and cannot perform the requested operation

Modbus TCP Command Introduction

Here is a brief introduction to Modbus TCP and Modbus RTU protocol conversion using the above commands to open the first relay as an example.

- Modbus RTU command: 01 05 00 00 FF 00 8C 3A

Field	Description	Note
01	Device Address	Fixed 0x01
05	05 Command	Relay control

00 00	Address	The register address of the relay to be controlled, 0x00, that is, the first relay
FF 00	Command	0xFF00: Relay on
8C 3A	CRC16	The CRC16 checksum of the first 6 bytes of data

- Modbus TCP command: 00 00 00 00 00 06 01 05 00 00 FF 00

Field	Description	Note
00 00	Message Label	Both be 0x00
00 00	modbus Label	Must both be 0, which means this is Modbus communication
00 06	Byte Length	Indicates the number of all bytes that follow, followed by 6 bytes
01	Device Address	Fixed 0x01
05	05 Command	Relay control
00 00	Address	The register address of the relay to be controlled, 0x00, that is, the first relay
FF 00	Command	0xFF00: Relay on

By comparing the commands above, we can observe that to convert a Modbus RTU command to Modbus TCP protocol, the CRC check is removed, and the command is prefixed with five 0x00 bytes followed by a byte representing the length.

Advanced Applications

- [Relay control through Alibaba Cloud MQTT](#)
- [Relay control through Waveshare Cloud](#)
- [Relay control through HTTP GET/POST](#)

Resources

Demos

- [Demo](#)

Software

- [Vircom configuration software](#)
- [Virtual serial port driver](#)
- [Sscm software](#)
- [Modbus Poll software](#)
- [SecureCRT software](#)

Related Resources

- [Modbus Protocol Specification](#)
- [Modbus Series BootLoader Description](#)