

# **ZLAN8308**

## **4G CAT1 DTU**

### **User Manual**

**RS232/485 To 4G**  
**Modbus RTU To 4G Modbus TCP**  
**RS232/485 To MQTT**  
**DLT-645/RTU to Cloud platform JSON**

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

ZL DUI 20200903.1.0



### Version Information

The History of the revision to this document:

			<b>History</b>
Date	Version	Document ID	Revising content
2020-09-03	Rev.1	ZL DUI 20200903.1.0	release

### Copyright information

Information in this document is subject to change without notice. It is against the law to copy the document on any medium except as specifically allowed in the license or nondisclosure agreement. The purchaser may make one copy of the document for backup purposes. No part of this manual may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or through information storage and retrieval systems, for any purpose other than for the purchaser's personal use, without the express written permission of Shanghai ZLAN information Technology Co., Ltd.

# CONTENT

1.	<b>SUMMARY</b> .....
2.	<b>FEATURES</b> .....
3.	<b>TECHNICAL PARAMETERS</b> .....
4.	<b>HARDWARE</b> .....
5.	<b>CONFIGURATION</b> .....
5.1.	Serial Port AT Instruction configuration.....
5.2.	Firmware/Profile mode.....
6.	<b>DEVICE FUNCTION</b> .....
6.1.	Communication Testing.....
6.2.	MODBUS RTU To JSON Testing.....
7.	<b>AT INSTRUCTION</b> .....
7.1.	Login in and Configuration.....
7.2.	Serial port data.....
7.3.	Internet data.....
7.4.	Register Pack and Heartbeat Pack.....
7.5.	Remote management function.....
7.6.	MQTT Data.....
8.	<b>ACCESSARIES</b> .....
9.	<b>AFTER-SERVICE AND TECHNICAL SUPPORT</b> .....

## 1. Summary

ZLAN8308 is new developed item which is high cost-effective CAT1 4G DTU, support 2G GPRS mode。 Realize RS232/485 To 4G, CAT1 4G transmission speed reaches uplink 5Mbps, downlink 10Mbps, much higher than traditional 2G GPRS DTU。

In addition, comparing price between ZLAN8308 and traditional 2G device, added register pack, heartbeat pack, MQTT, Modbus RTU To JSON Cloud server, etc. Compatible with and replace 2G products at the same time to achieve high speed transmission, low delay, support new technology and other characteristics。



图 1 ZLAN8308 外观图

ZLAN8308 inherits the stability of ZLAN 2G product ZLAN8100 and the powerful function of ordinary 4G DTU ZLAN8305. You can configure devices through serial ports, upgrade firmware, configure MQTT/JSON and other advanced Settings. At the same time support through the server remote centralized management of a large number of distributed devices, can be remote configuration, remote view status, remote upgrade program. Cooperating with the public cloud or zlan cloud, the

device management and data viewing and remote control on the Web side can be realized

ZLAN8308 specially designed watchdog circuit can ensure the stable operation of the 4G module for a long time. The product supports the industrial temperature range from -40 degrees to 85 degrees.

ZLAN8308 Applications:

1. Data collection in the field of industrial Internet and industrial automation.
2. Power data acquisition and monitoring.
3. Access control and security.
4. Collection and monitoring of hydrological, meteorological and environmental data.
5. Intelligent transportation, vehicle-mounted data acquisition.

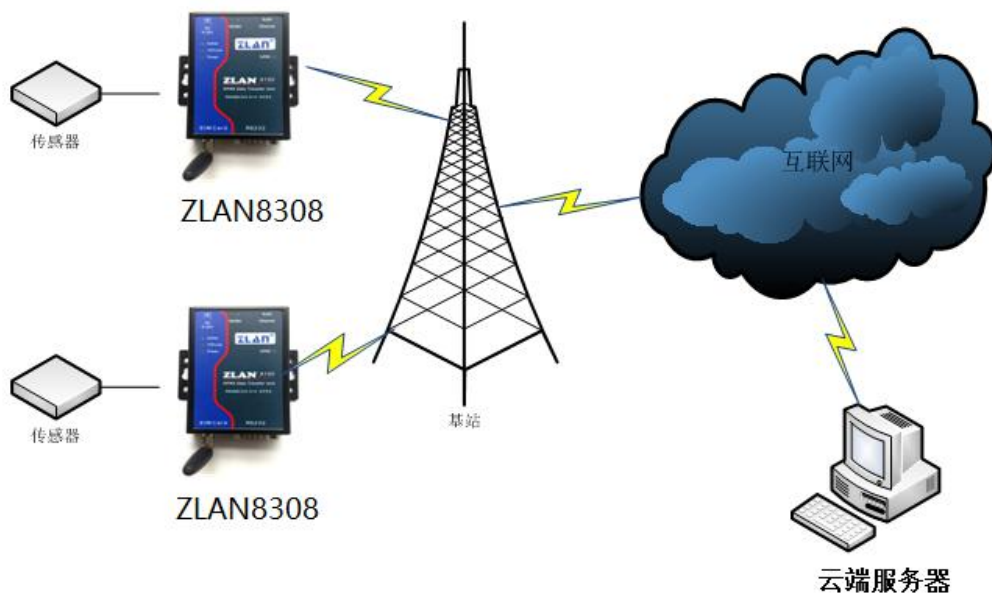


Chart 2. Application Environment

## 1. Function

1. Support 3 modes, TD-LTE/ FDD-LTE/ GSM, including Unicom, 4G, 2G, mobile 4G, 2G and telecom 4G.

2. Support TCP client, UDP mode.
3. Serial port support 300~921600 baud rate, support 5~8 data bits, support no check, odd check, even check, support 1~2 stop bit.
4. Support serial port (RS232/485) To 4G.
5. Support serial port transparent transmission, 8308 supports Modbus RTU to Modbus TCP, MQTT protocol.
6. Support serial port AT command configuration, support ZLVIROM software to view some parameters.
7. Support for serial port configuration of MQTT parameters.
8. Support DTL-645/Modbus RTU automatic collection and conversion to cloud platform JSON format.
9. The 8308 firmware can be updated on the device through the serial port, and the firmware can be updated on the server side through the ZLvircom software.
10. Support server side remote device management, device configuration, device upgrade.

## 2. Technical parameters

Product Specifications		
Name	Specifications	Remark
Support Mode	4G CAT1 support 3 modes: B1/B3/B5/B8@FDD LTE B34/B38/B39/B40/B41@TDD-LTE B3/B8@GSM including Unicom 4G, 2G, Mobile 4G, 2G and telecome 4G Internet.	
Transmission rate	LTE: Max 10Mbps (downlink) /Max 5 Mbps (uplink) GPRS: 85.6Kbps (downlink) /Max85.6Kbps (uplink)	
SIM card	Voltage: 3V, 1.8V; Size: Big card (Small card can be purchased to use the card sleeve)	

Antenna	50Ω/SMA Rubber stick antenna or suction cup antenna is optional	
Interface	RS232/RS485	
Serial port data	Baud rate: 300~921600bps; Digit bit: 5~8 位; Stop bits: 1~2 位; Check bits: none, even, odd.	
Power supply	Q2.1 socket, Can be customized for power terminal input.	
input voltage	DC9V~24V	
Working current	dial/4G When communication 50mA@12V, free 25mA@12V	
Working temperature	-40 degree~85degree	
Storage temperature	-40 degree~120 degree	
Humidity range	0~95% condensation	
Device size	Length×Width×Height=9.4cm×6.5cm×2.5cm	

### 3. Hardware

ZLAN8308 Picture is as Chart 3.



Chart 3 ZLAN8308 Front View

8308 Adopts anti-radiation metal shell, two hanging lugs on both sides, which can be fixed with screws; It can also be equipped with guide rail accessories.

#### Panel lights:

Signal	Green	Blue
Active Signal light	The serial port sends data	The serial port receives data
Link Signal light	8308 Starts but the network is not connected	A constant blue indicates that a TCP connection has been established
4G Signal light	Power supply signal	The blue flashing indicates that 4G is dialing. The dialing starts 15 seconds after the system is powered on. Generally, it can be dialed within 10 seconds.  Blue is always bright, indicating 4G connectivity



Chart 4Interface 1

ZLAN8308 Front interface is shown as chart 4:

1. Input power supply: interface Q2.1 socket, input voltage DC+9V~ +24VDC, power need above 3W。 default adapter is 12V。 Can be customized for power terminal input。
2. RS485 interface: RS485 signal input, Be careful not to connect the power。
3. RJ45 interface: Leave it to later to extend the Ethernet interface. It is currently invalid。

ZLAN8308 Backside interface is as picture Chart 5.:



Chart 5interface 2

4. antenna: 8308 antenna interface using 50  $\Omega$  /SMA (female head), external antenna must be suitable for 4G working band antenna. ZLAN can provide glue stick or

suction cup antenna, the suction cup can be sucked to the metal housing of the chassis (the default suction cup antenna lead is 1.5 meters long)。

5. SIM Card installed: When installing the SIM card, ensure that the device is not powered on. Using a pen tip and a screwdriver, push the SIM card slot out and push the SIM metal face down into the slot。
6. DB9: RS232 signal input。

## 4. Configuration

The device can configure parameters through a serial port and can also be configured remotely by installing configuration software on a remote server after connecting to it。

### 5.1. Serial port AT instruction configuration

Download [ZLVircomconfiguration](http://www.zlmcu.com/download/ZLVirComconfiguration) tool (<http://www.zlmcu.com/download/ZLVirCom.zip>), This software can be configured through the serial port of 8308。

Connect the USB to RS232 line to the serial port of 8308, power on 8308, open ZLvircom (hereinafter referred to as configuration tool), and enter the main interface of the configuration tool as shown in Chart 6.

Click Device Management to select serial port search, as shown in Chart 7, and the serial port parameter selection interface appears, as shown in Chart 8. Select the serial port number, here is COM15, and the baud rate is 115200. 115200 is the factory default setting, and if the user previously set 8308 to other baud rate (such as 9600), it can also be searched。

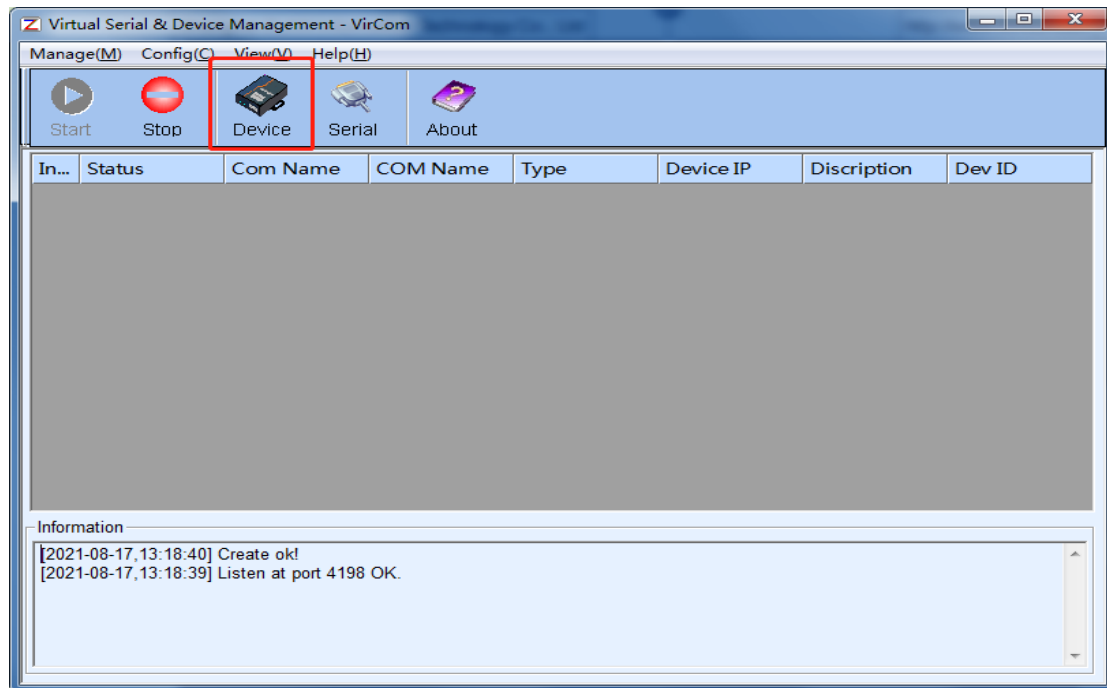


Chart 6 Configure the main tool interface

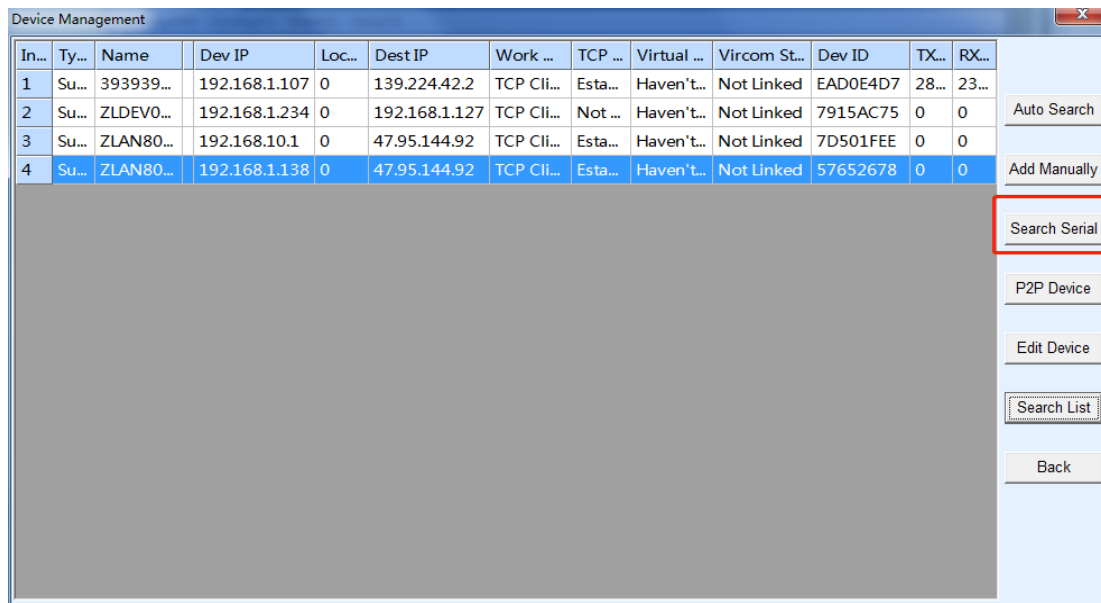


Chart 7 Serial port search interface

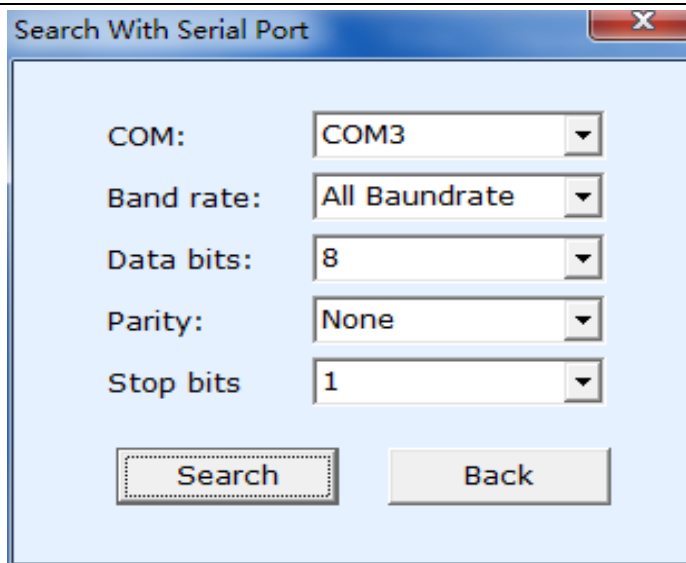


Chart 8 Serial port parameter setting

After power on, wait for 15 seconds, that is, after the dial light starts flashing, select and click the "Search" serial port. At this time, the configuration tool will attempt to communicate with the device. If successful, it will enter the ConfTool interface. See Chart 9 below:

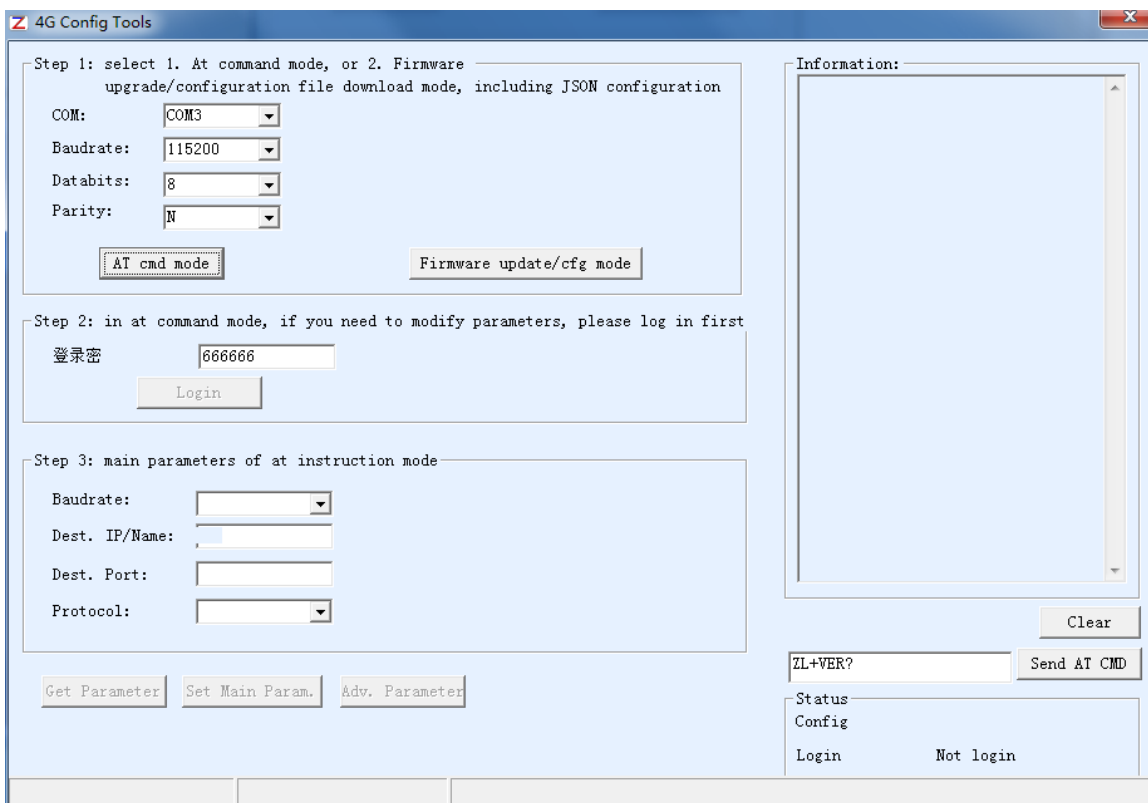


Chart 9 ConfTool Page

Click to enter the AT command mode, and the configuration tool will attempt to communicate with the device. After successful communication, the return information of AT command will be displayed on the right side, and the configuration mode will be shown as having entered the configuration mode, as shown in Chart 10 below:

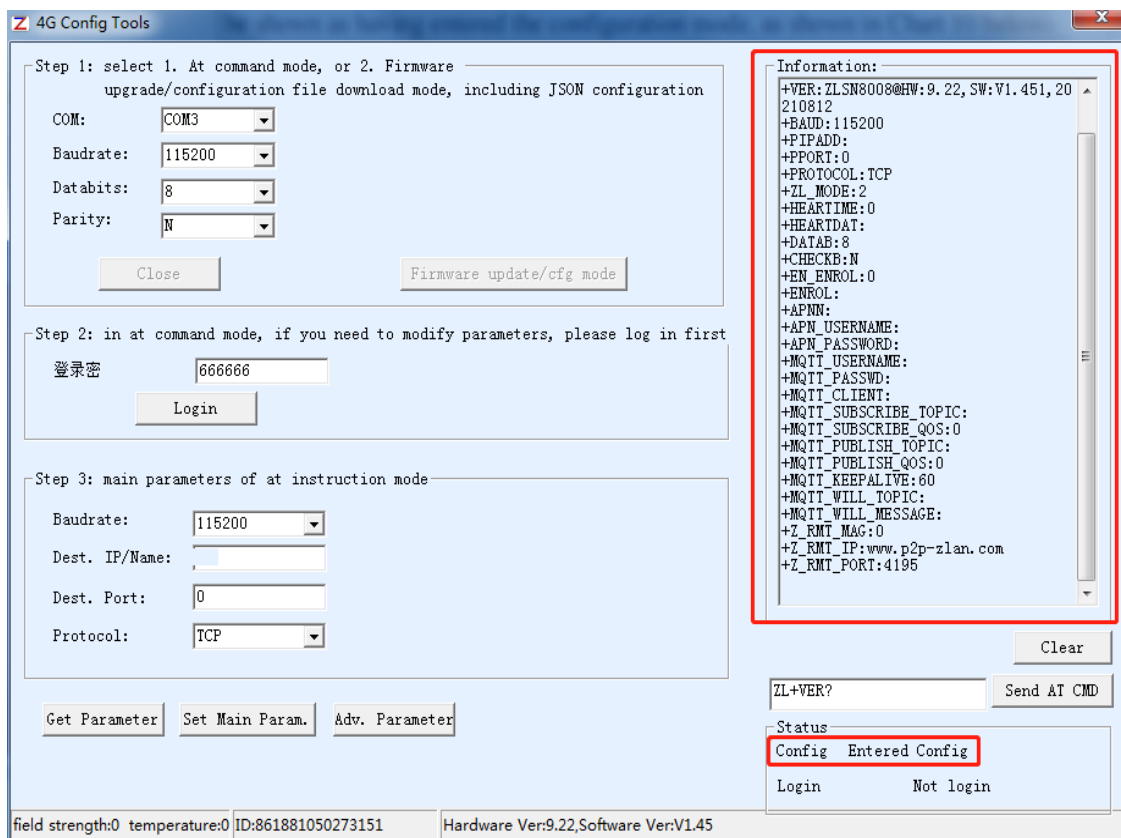


Chart 10 Enter the configuration mode interface

The default login password is 666666. Before clicking "Login", the parameters are read-only and cannot be set or modified. Click the "Login button":

You can see that after logging in, the LOGIN status changes to "Login" and the message "+ Login OK" appears on the right side, as shown in Chart 11.

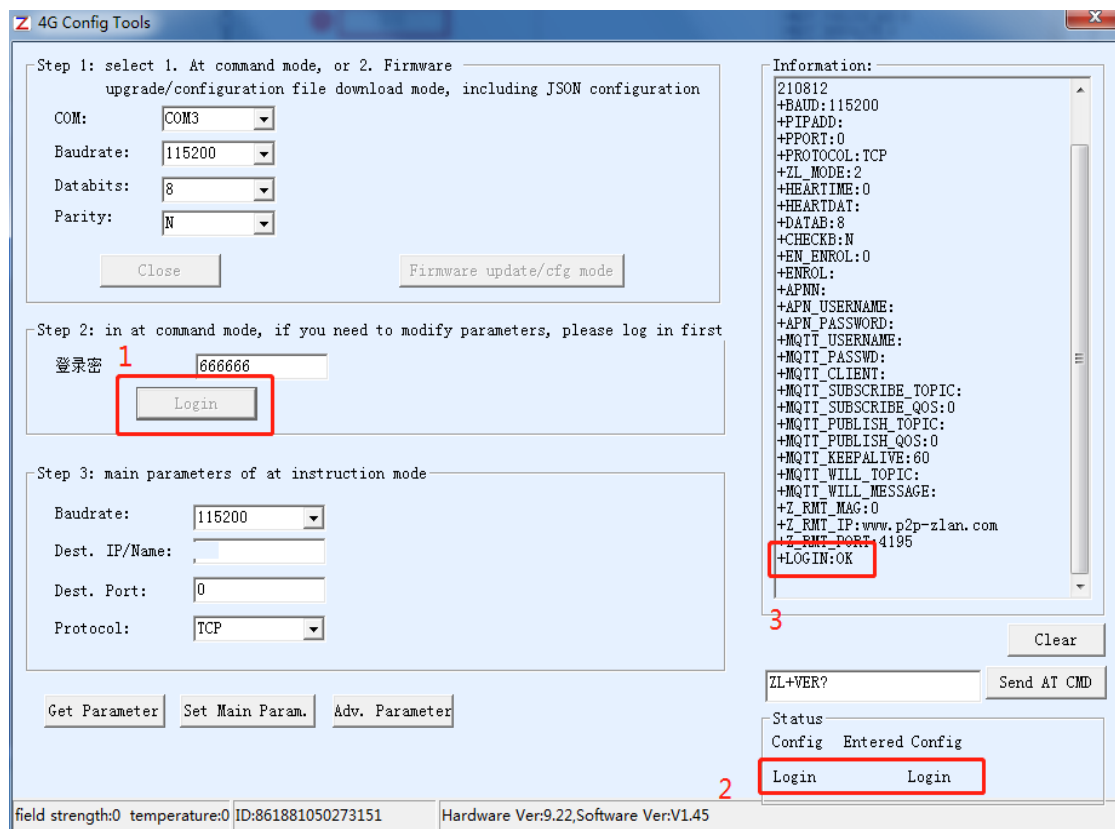


Chart 11 Login page

The main parameters of the AT instruction mode include baud rate, destination IP, destination port, and protocol. Protocol support TCP or UDP protocol. After modifying the corresponding parameters, click "Set Parameters" to set the new parameters to the device. At the same time, the device will return the parameters successfully set, as shown in Chart 12.

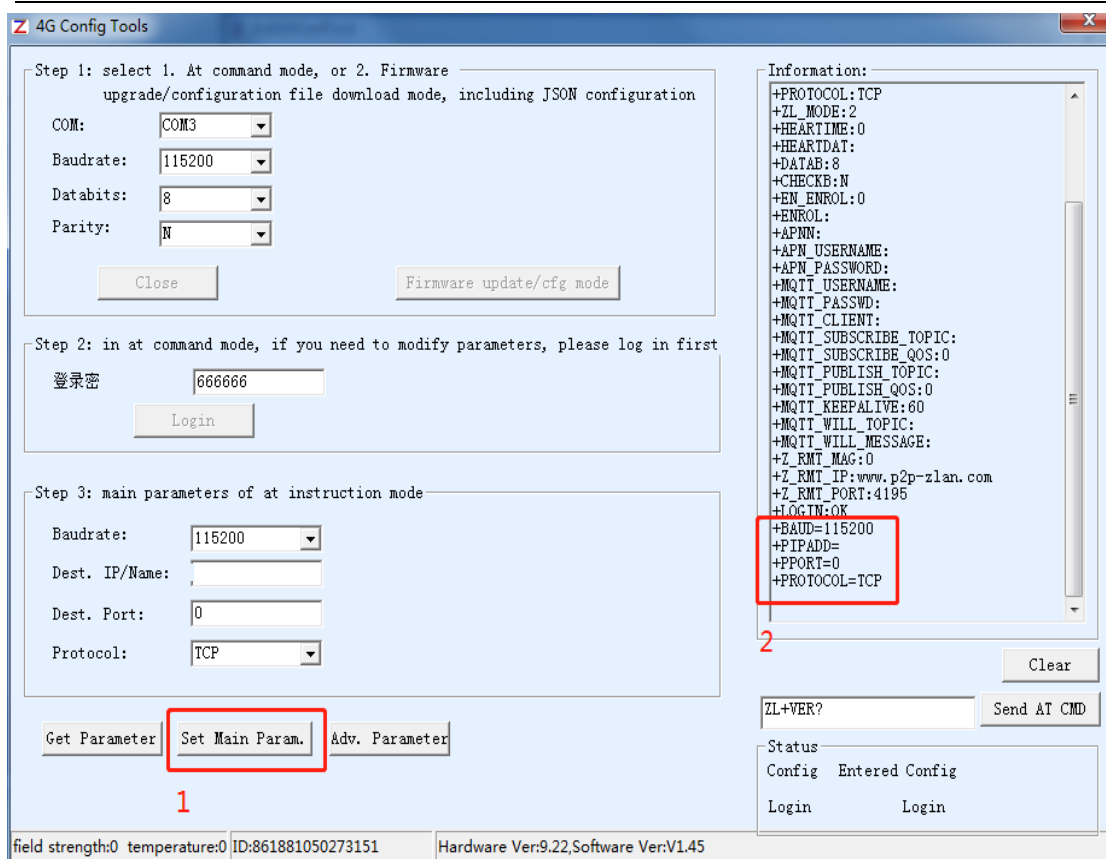


Chart 12 Set the parameters

“The "get parameters" button can get the parameters of the current device. The get parameters are obtained by sending the AT instruction. The data returned by the AT instruction is listed on the right. Refer to the other sections of this article for the AT directive. Since the "Get Parameters" button will be automatically executed once the "Open" is successful, there is generally no need to click the "Get Parameters" button.

Click "Advanced Parameters", and the advanced parameters box is shown in Chart 13. The commonly used parameters are:

1. Heartbeat interval: You can set heartbeat packets with an interval of 15 seconds
2. Heartbeat content: Sets the contents of the heartbeat packet
3. Serial port data bits
4. Serial port check bit
5. Enable the registration package: Enable the registration package
6. Registered package content: The content of the registered package sent after

connecting to the server

7. APN: APN Name of access point.
8. APN User name
9. APN password
10. MQTT data: Used to set the parameters to access the MQTT server
11. Device remote management: it is used for devices with remote management function to access the remote server

After selecting the parameters, click the "Effective Advanced Parameters" button, and observe the information bar on the right to see whether the Settings information returned by the device is consistent with the information filled in, as shown in Chart 14.

Advanced Parameters

Work Parameters

Work Type: MQTT

DNS Server IP:

Heart Beat Interval: Disable

Heart Beat Content:

Serial Data Bits: 8

Serial Parity: N

Login Key:

Enable Register Pkt: Disable

Register Pkt Content:

APN:

APN UserName:

APN Key:

MQTT Parameters

MQTT version: V3.1.1

User Name:

Key:

Client ID:

Subscript topics:

Subscript QOS: 0

Publish Topics:

Publish QOS: 0

Keep Alive Time: 60

Last-will Topics:

Last-will Message:

Remote Device Manage

Enable Remote Device Manage

Server IP/DNS: www.p2p-zlan.com

Server TCP Port: 4195

Set Cancel Get Default

Chart 13 Advanced data

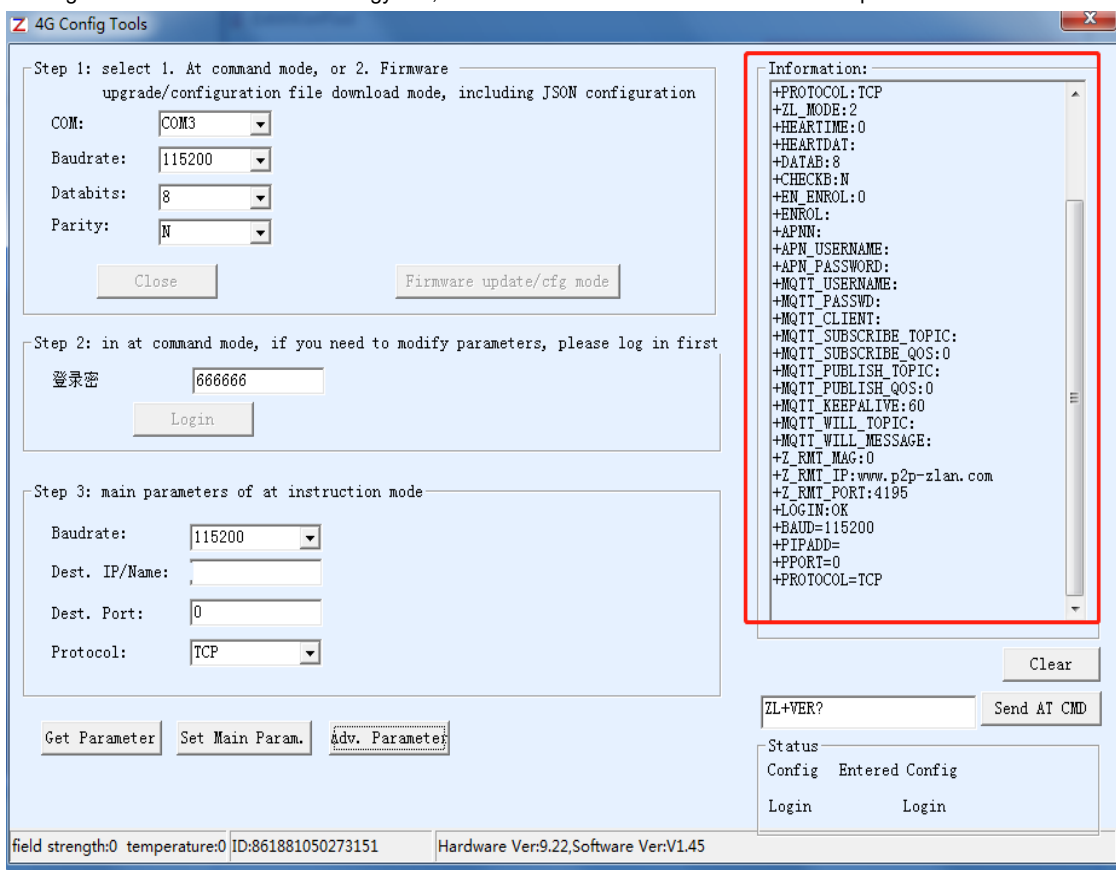


Chart 14 Set advanced parameters to return information

## 5.2. Firmware/profile mode

After entering the Conftool interface, click the firmware/configuration file mode button, as shown in chart. 15, and jump to the firmware/configuration file interface as shown in chart. 16. First, create the root directory of the local configuration page to store the configuration file, click MQTT configuration to input the information of connecting to the MQTT server, and click Save MQTT configuration, as shown in chart. 17. Click the JSON configuration for JSON up-post configuration and save the JSON configuration, as shown in chart 18. Click the "Download" button and the configuration software will download all the files in the directory to the 8308 device. After successful download, the interface will pop up to complete transmission and the device will automatically restart, as shown in chart. 19.

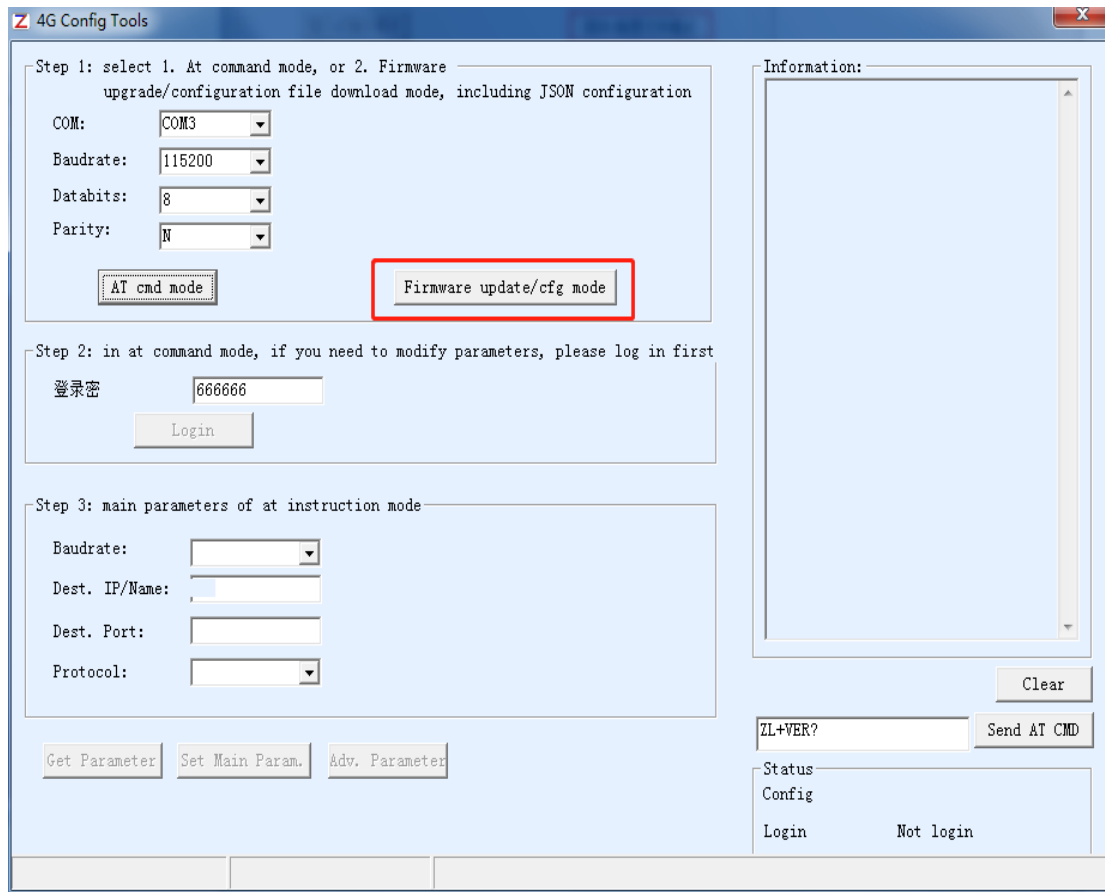


Chart 15 Configuration page

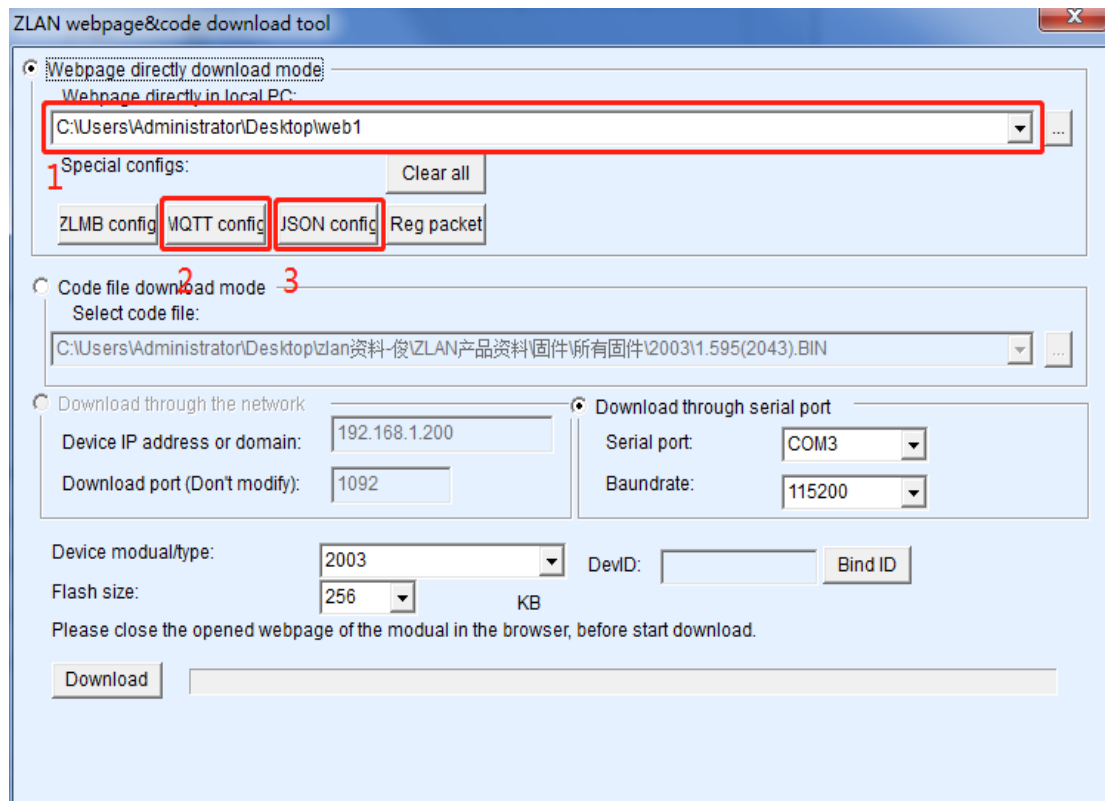
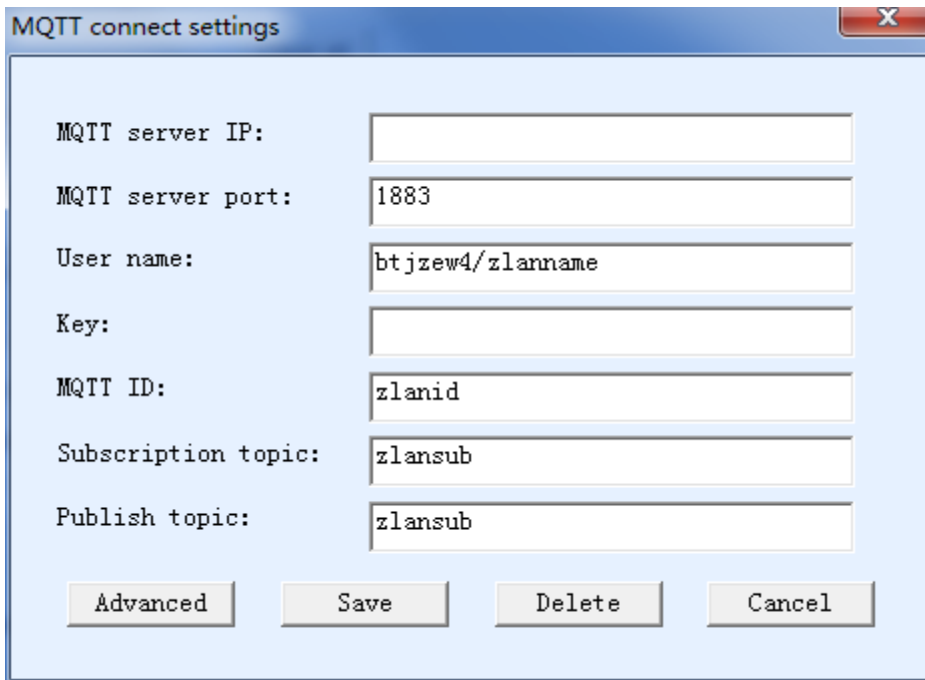


Chart 16Firmware/configuration file page

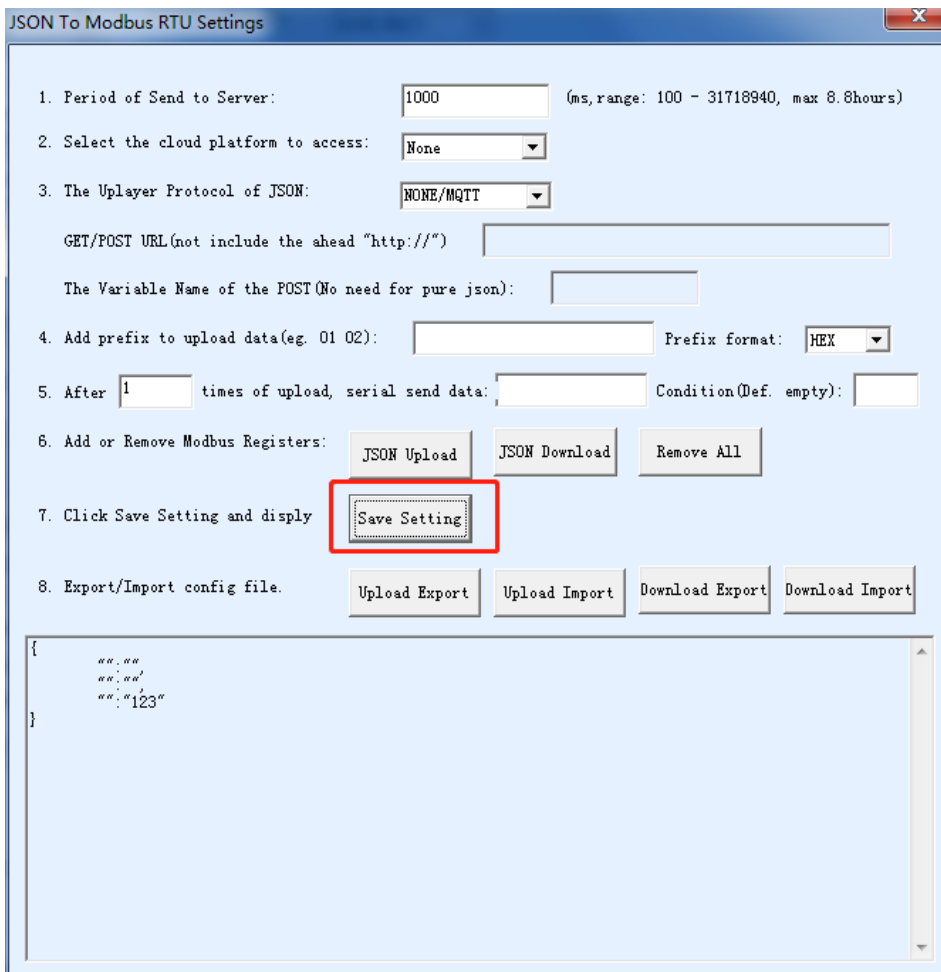


The image shows a dialog box titled "MQTT connect settings" with a close button (X) in the top right corner. It contains several input fields for MQTT configuration:

- MQTT server IP: [Empty text box]
- MQTT server port: 1883
- User name: btjzew4/zlanname
- Key: [Empty text box]
- MQTT ID: zlanid
- Subscription topic: zlanub
- Publish topic: zlanub

At the bottom, there are four buttons: "Advanced", "Save", "Delete", and "Cancel".

Chart 17 MQTT Configuration page



The image shows a dialog box titled "JSON To Modbus RTU Settings" with a close button (X) in the top right corner. It contains several configuration options:

- 1. Period of Send to Server: 1000 (ms, range: 100 - 31718940, max 8.8hours)
- 2. Select the cloud platform to access: None
- 3. The Uplayer Protocol of JSON: NONE/MQTT
- GET/POST URL(not include the ahead "http://"): [Empty text box]
- The Variable Name of the POST(No need for pure json): [Empty text box]
- 4. Add prefix to upload data(eg. 01 02): [Empty text box] Prefix format: HEX
- 5. After 1 times of upload, serial send data: [Empty text box] Condition(Def. empty): [Empty text box]
- 6. Add or Remove Modbus Registers: JSON Upload, JSON Download, Remove All
- 7. Click Save Setting and displ: Save Setting (highlighted with a red box)
- 8. Export/Import config file. Upload Export, Upload Import, Download Export, Download Import

At the bottom, there is a text area containing a JSON snippet:

```
{  
  "": "  
  "": "  
  "": "123"  
}
```

Chart 18 JSON configuration page

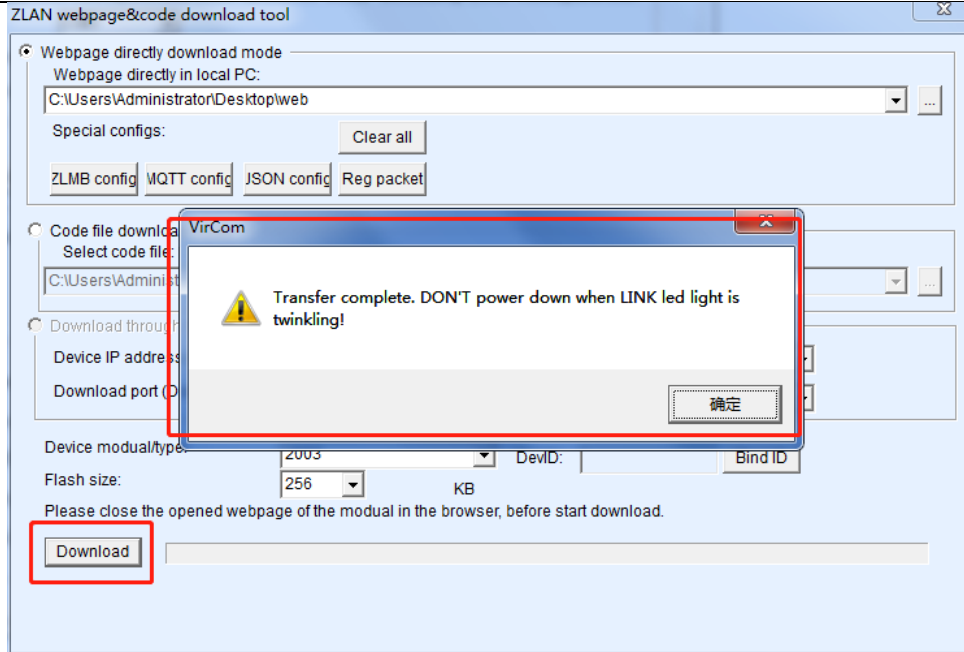


Chart 19 download page

## 5. Device Function

### 6.1. Communication Test

#### 6.1.1 Server transparent transport test

Assuming the following networking structure as shown in the figure below, 8305 is configured to connect to the server\*\*\*.\*\*\*.\*\*\*.\*\*\*'s \*\*\* interface。 Please use the methods in the "Serial port configuration" section. After the configuration is complete, power it back on and it takes 20 to 40 seconds to connect to the server。



Chart 20 Network structure diagram

We run SocketDlgTest, the TCP tool, on the server ([http://www.zlmcu.com/document/tcp\\_debug\\_tools.html](http://www.zlmcu.com/document/tcp_debug_tools.html)).



Chart 21 Server-side tools

Select the local port as 4196 (note that if you are running the ZLVircom tool, you will need to change the port) and click the "Open" button. When The 8308 device is connected to The server, it will display "The No... Is accepted!" The information.

Now connect the serial port of the 8308 device to the USB to 232 serial line, and open the serial debugging tool ([http://www.zlmcu.com/document/com\\_debug\\_tools.html](http://www.zlmcu.com/document/com_debug_tools.html)), Open the correct COM port.

Now the serial port sends data, then the corresponding data will be replied at the server side. Similarly, when the device receives the message replied by the server, the serial port tool will receive the same data here. This demonstrates the network two-way communication from serial port to 4G, as shown in Chart 22 below:

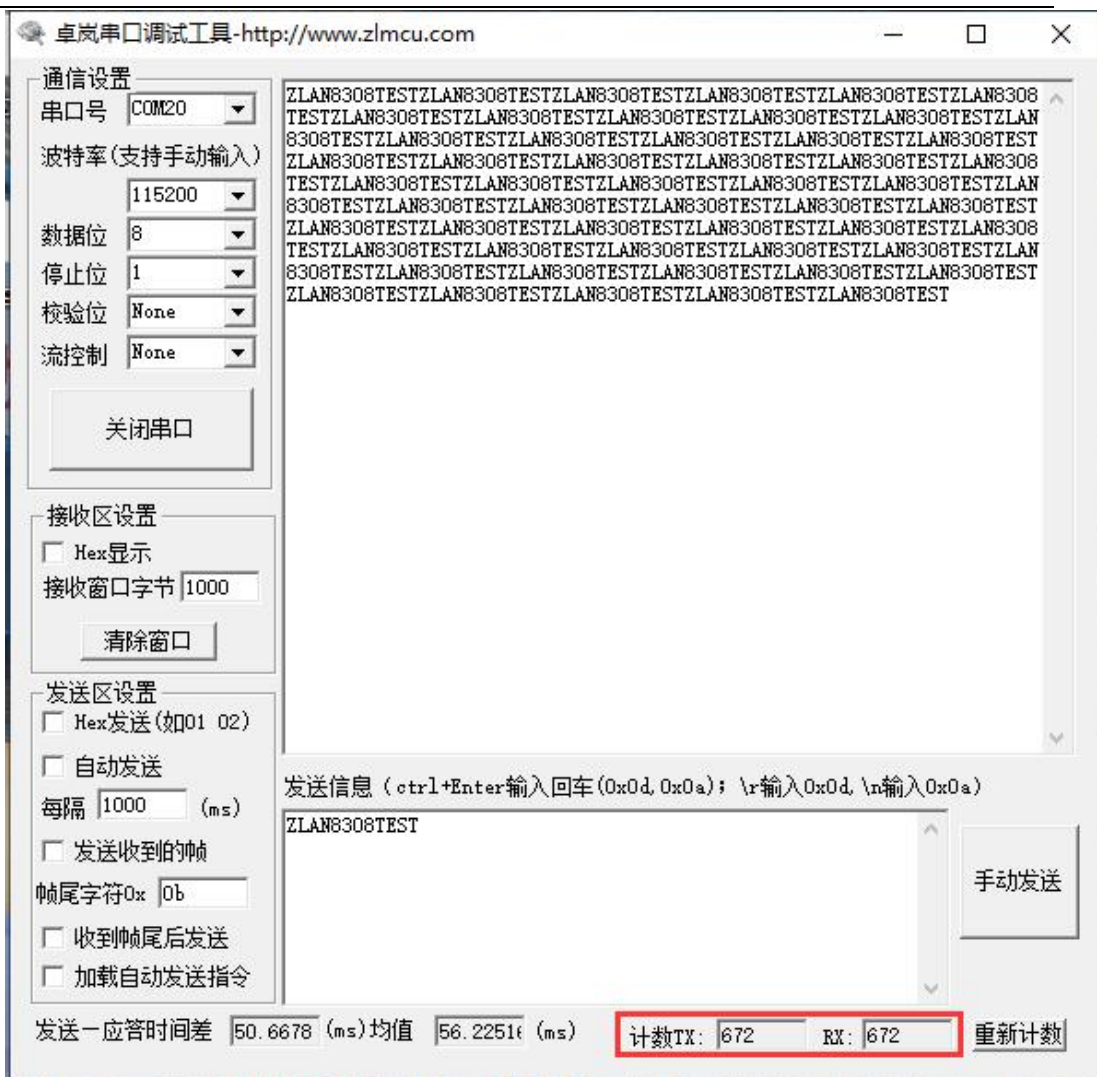


Chart 22 Device side serial port debugging tools

### 6.1.2 Modbus Protocol transformation test

The configuration parameters are basically the same as the no-protocol pass-through test; you only need to change the transformation protocol to the Modbus protocol. The serial port Modbus RTU protocol can be converted to the network Modbus TCP protocol, and the network Modbus TCP protocol can be converted to the serial port Modbus RTU protocol.

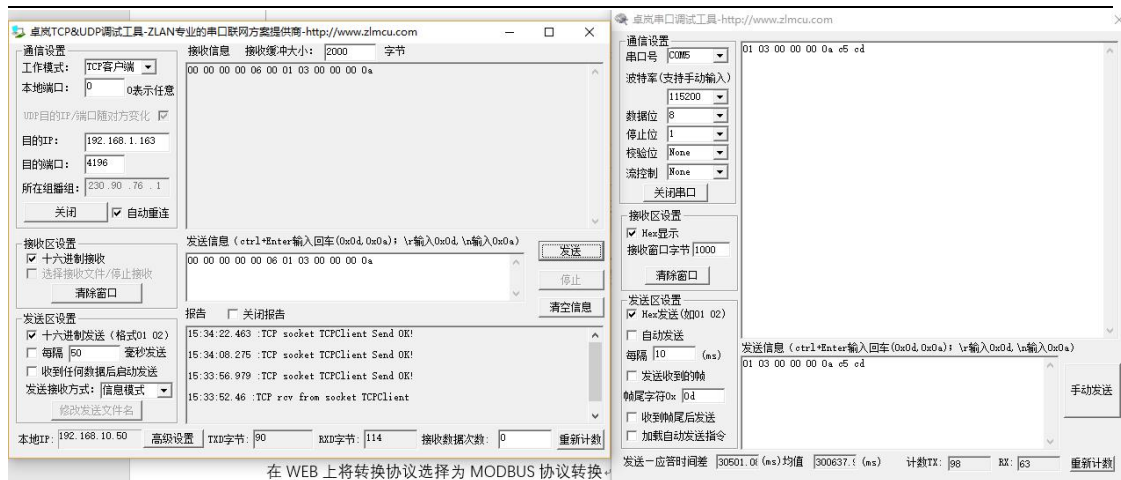


Chart 23 Modbus Protocol transformation test

### 6.1.3 MQTT protocol test

This test is to test the connection to Ali Cloud. Create a new subscription topic called zlan\_test and a publication topic called zlan\_1 on AliCloud, as shown in Chart 24. According to the instructions of the fifth step, first fill in the IP and port configuration of the MQTT server, save the parameters, and fill in the parameters as shown in Chart 25. Then notice the page of advanced parameters to fill in the ID, user name, password of MQTT, including the subject of subscription and release, and the guaranteed time, as shown in Chart 26。

自定义 Topic	操作权限	描述
/a1WSVHIXkDh/\${deviceName}/user/zlan_test	订阅	-
/a1WSVHIXkDh/\${deviceName}/user/zlan_1	发布	-

Chart 24 Ali cloud to add theme

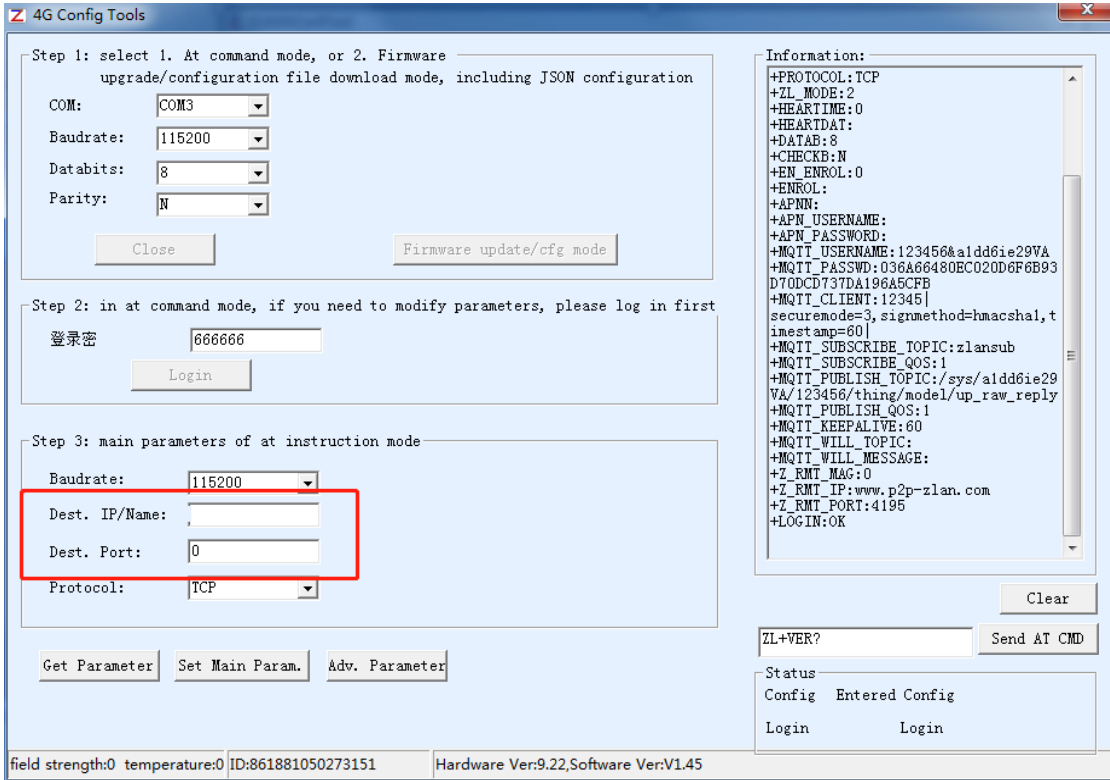


Chart 25 Ali Cloud IP and interface

Advanced Parameters

Work Parameters

Work Type: MQTT

DNS Server IP:

Heart Beat Interval: Disable

Heart Beat Content:

Serial Data Bits: 8

Serial Parity: N

Login Key:

Enable Register Pkt: Disable

Register Pkt Content:

APN:

APN UserName:

APN Key:

MQTT Parameters

MQTT version: V3.1.1

User Name: 123456&a1dd6ie29VA

Key: 036A66480ECC020D6F6B93D70I

Client ID: 12345|securemode=3, signme

Subscript topics: zlanub

Subscript QOS: 1

Publish Topics: /sys/a1dd6ie29VA/123456/1

Publish QOS: 1

Keep Alive Time: 60

Last-will Topics:

Last-will Message:

Remote Device Manage

Enable Remote Device Manage

Server IP/DNS: www.p2p-zlan.com

Server TCP Port: 4195

Set Cancel Get Default

Chart 26 Ali Cloud MQTT Configuration

After setting, open the device management interface of Ali Cloud and enter the log service page to view the information sent on the device, as shown in Chart 27. Data is sent through the serial port of the device, and a message (" Zlan8308test ") is sent through the theme of Zlan\_1 to the MQTT server of Ali Cloud. Ali Cloud receives the data as shown in Chart 28, and Ali Cloud server sends a message (" ALI\_SEND ") through the theme of Zlan \_test to the serial port of the device. As shown in Chart 29, this completes the MQTT transceiver test.

时间	TraceID	消息内容	DeviceName	业务类型(全部)	操作	内容	状态
2021/02/04 17:50:31.317	0a3027d16124032312967569d1e3	-	112121	设备行为	online	[Content]"onlin...	200
2021/02/04 17:50:31.587	0a3027d1612403231597827d1e3	-	112121	订阅	/a1WSVHIXkDh/112121...	[Content]"Subs...	200
2021/02/04 17:50:31.802	0a3027d16124032317997993d1e3	报警	112121	设备到云消息	/a1WSVHIXkDh/112121...	[Content]"Publ...	200
2021/02/04 17:19:05.216	0a3027d161240303452136931e3383	-	112121	设备行为	offline	[Content]"offin...	200
2021/02/04 17:19:04.243	0a3027d161240304406303e5383	报警	112121	设备到云消息	/a1WSVHIXkDh/112121...	[Content]"Publ...	200
2021/02/04 17:19:02.688	0a3027d16124030429655445e5383	报警	112121	设备到云消息	/a1WSVHIXkDh/112121...	[Content]"Publ...	200
2021/02/04 17:19:01.126	0a3027d16124030411254245e5383	报警	112121	设备到云消息	/a1WSVHIXkDh/112121...	[Content]"Publ...	200
2021/02/04 17:18:59.968	0a3027d1612403039653159e5383	报警	112121	设备到云消息	/a1WSVHIXkDh/112121...	[Content]"Publ...	200
2021/02/04 17:18:58.11	0a3027d16124030380102142e5383	报警	112121	设备到云消息	/a1WSVHIXkDh/112121...	[Content]"Publ...	200
2021/02/04 17:18:56.452	0a3027d1612403036451342e5383	报警	112121	设备到云消息	/a1WSVHIXkDh/112121...	[Content]"Publ...	200

Chart 27 Ali Cloud log service

**查看详情** ✕

Topic	/a1WSVHIXkDh/112121/user/zlan_1	
时间	2021/02/04 17:51:52.932	
内容	Text (UTF-8) <span>▼</span>	ZLAN8308TEST <span style="float: right;">复制</span>

关闭

Chart 28 Ali Cloud received serial port data

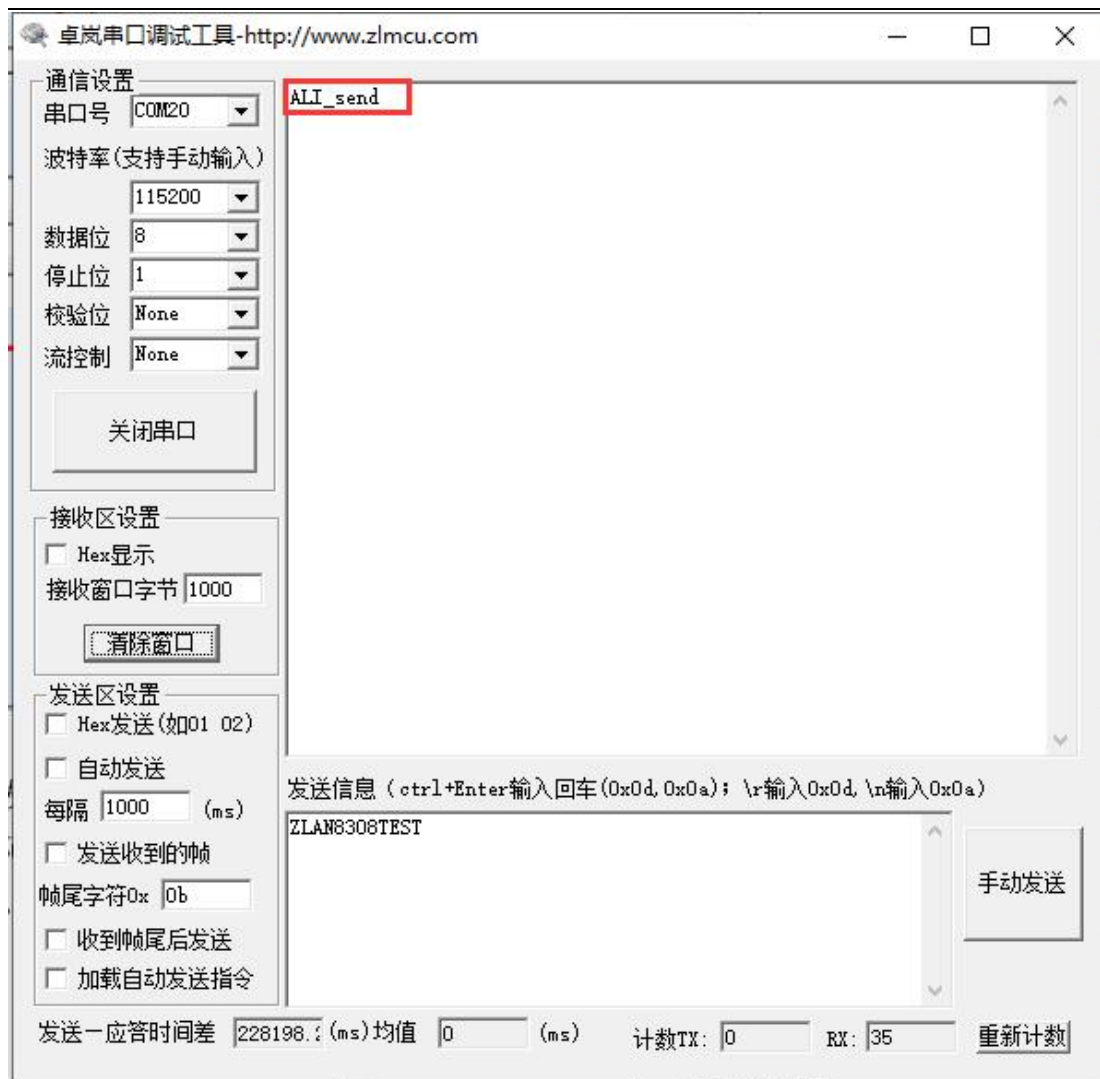


Chart 29 Serial port received Ali cloud data

## 6.2. MODBUS RTU To JSON Test

### 6.2.1 Configure JSON uplink

Through the above part: Modbus protocol transformation test, simple JSON upload template is configured. The configuration process is shown in Chart 30, Chart 31, Chart 32, and Chart 33. Data of some Modbus nodes is collected and converted into JSON format for upload.

The screenshot shows a software window titled "JSON To Modbus RTU Settings" with a close button in the top right corner. The window contains several configuration steps:

- 1. Period of Send to Server: 1000 (ms, range: 100 - 31718940, max 8.8hours)
- 2. Select the cloud platform to access: None
- 3. The Uplayer Protocol of JSON: NONE/MQTT
- GET/POST URL(not include the ahead "http://")
- The Variable Name of the POST(No need for pure json):
- 4. Add prefix to upload data(eg. 01 02): Prefix format: HEX
- 5. After 1 times of upload, serial send data: Condition(Def. empty):
- 6. Add or Remove Modbus Registers: JSON Upload (highlighted with a red rectangle), JSON Download, Remove All
- 7. Click Save Setting and dispaly: Save Setting
- 8. Export/Import config file: Upload Export, Upload Import, Download Export, Download Import

At the bottom of the window is a large empty text area with a vertical scrollbar on the right side.

Chart 30 Configure JSON uplink

Following is the 1. th design of register. It has been added:

JSON node data type:  Object data(Default value, including this node and later ones with { }, need Input JSON keyword)  
 Array data(including data by [ ], without JSON Keyword)

Corresponding JSON Keyword:  Data source: Modbus RTU

Modbus RTU Settings  
- Slave Address:   
- Modbus Function Code:   
- Register Address:

645 Protocol (97 version)  
- 645 Version: 97 Version FE numbers: 0  
- Device ID: 000000000001 (6 bytes)  
- Data type: 9410 (eg. 9410 is the energy sum)

1. Data Length:  Bytes. 4 Bytes order: Big-Endian(Invert) (big-endin 4 bytes: Data ABCD, low address store 2 bytes AB)
2. Decimal Point Places:  digit. After get as integer left shift the decimal point.
3. Enable shift and scale:  Subtract integer:  then divide float:
4. Data Format: Unsigned int Bool value at position bit:
5. Add unit name to rear:
6. Add quotation for data:
7. The Period between two RTU cmd:  (ms) must bigger than 10.
8. Send data to server when data changed:
9. If RS485 device off line set data to 0:  Set data to 1 if online:

Buttons: Enter Embedded, Exit Embedded, Enter Next, Del and Enter, Save and Exit, Cancel and Exit

Chart 31 Configure the acquisition keywords, register address and acquisition interval

Following is the 2. th design of register. It has been added:

JSON node data type:  Object data(Default value, including this node and later ones with { }, need Input JSON keyword)  
 Array data(including data by [ ], without JSON Keyword)

Corresponding JSON Keyword: 2132 Data source: Modbus RTU

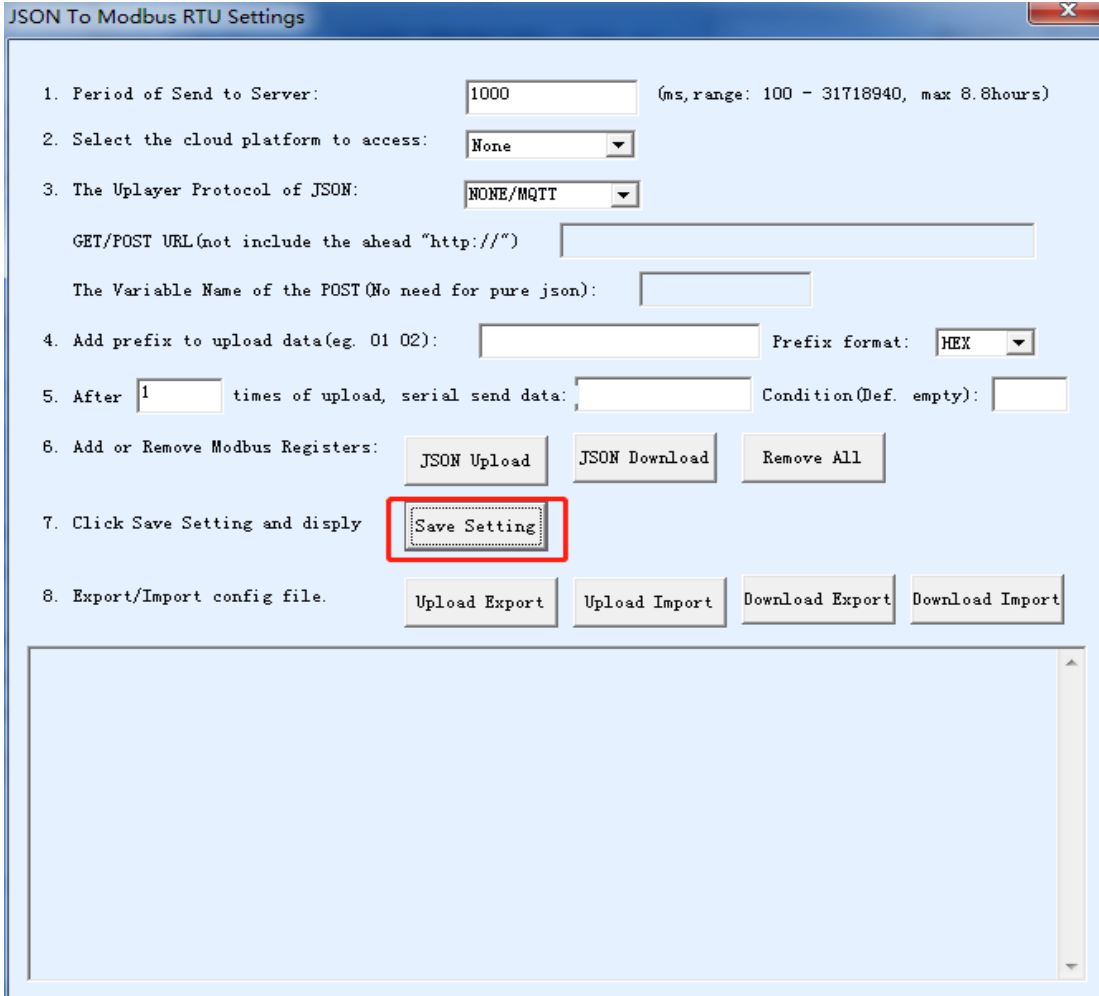
Modbus RTU Settings  
- Slave Address:   
- Modbus Function Code:   
- Register Address:

645 Protocol (97 version)  
- 645 Version: 97 Version FE numbers: 0  
- Device ID: 000000000001 (6 bytes)  
- Data type: 9410 (eg. 9410 is the energy sum)

1. Data Length:  Bytes. 4 Bytes order: Big-Endian(Invert) (big-endin 4 bytes: Data ABCD, low address store 2 bytes AB)
2. Decimal Point Places:  digit. After get as integer left shift the decimal point.
3. Enable shift and scale:  Subtract integer:  then divide float:
4. Data Format: Unsigned int Bool value at position bit:
5. Add unit name to rear:
6. Add quotation for data:
7. The Period between two RTU cmd:  (ms) must bigger than 10.
8. Send data to server when data changed:
9. If RS485 device off line set data to 0:  Set data to 1 if online:

Buttons: Enter Embedded, Exit Embedded, Enter Next, Del and Enter, Save and Exit, Cancel and Exit

Chart 32 Save and exit when the configuration is complete



The screenshot shows a software window titled "JSON To Modbus RTU Settings". It contains several configuration options:

- 1. Period of Send to Server: 1000 (ms, range: 100 - 31718940, max 8.8hours)
- 2. Select the cloud platform to access: None
- 3. The Uplayer Protocol of JSON: NONE/MQTT
- GET/POST URL(not include the ahead "http://")
- The Variable Name of the POST(No need for pure json):
- 4. Add prefix to upload data(eg. 01 02): Prefix format: HEX
- 5. After 1 times of upload, serial send data: Condition(Def. empty):
- 6. Add or Remove Modbus Registers: JSON Upload, JSON Download, Remove All
- 7. Click Save Setting and dispaly: Save Setting (highlighted with a red rectangle)
- 8. Export/Import config file. Upload Export, Upload Import, Download Export, Download Import

Below the settings is a large empty text area for previewing the JSON output.

Chart 33 Save the JSON Settings to view the preview in JSON format

### 6.2.2 Configure Modbus RTU simulation devices

Modbus Slave software is used to simulate the Modbus Slave device, connect the Zlan8308 device to the computer through the serial port line, and open the connection of Modbus Slave. Configuration of Modbus Slave is shown in Chart 34.

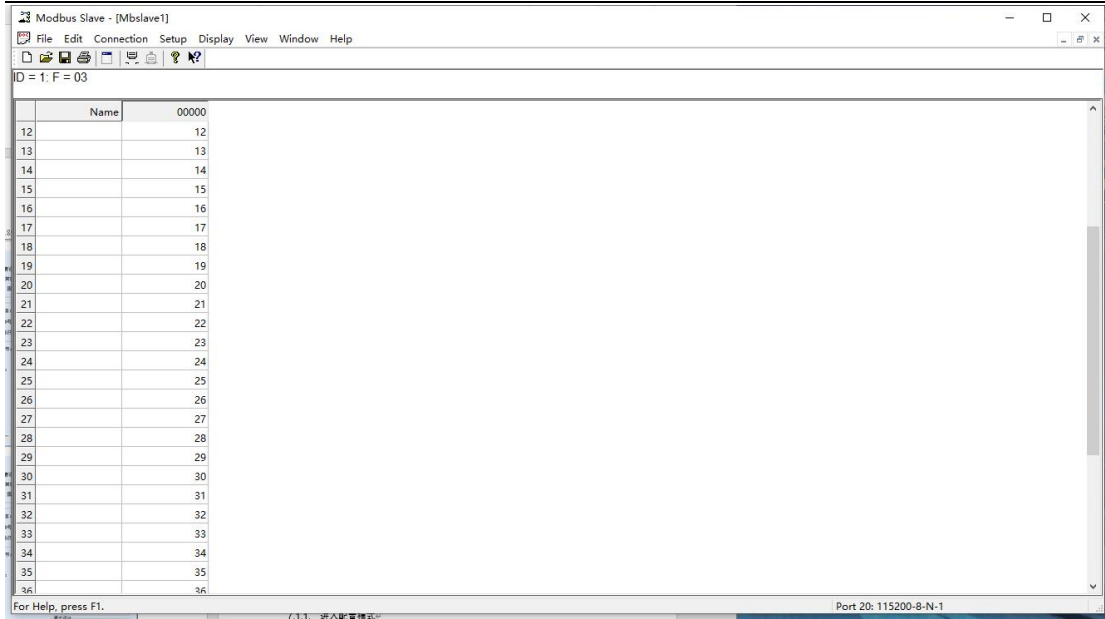


Chart 34 Modbus SlaveFill in analog data

### 6.2.3 Check uplink JSON

Through the AliCloud log service, we can view the JSON data sent above, and we can observe that the collected data is consistent with the data configured by Modbus Slave, which completes the simple test of converting JSON from Modbus.

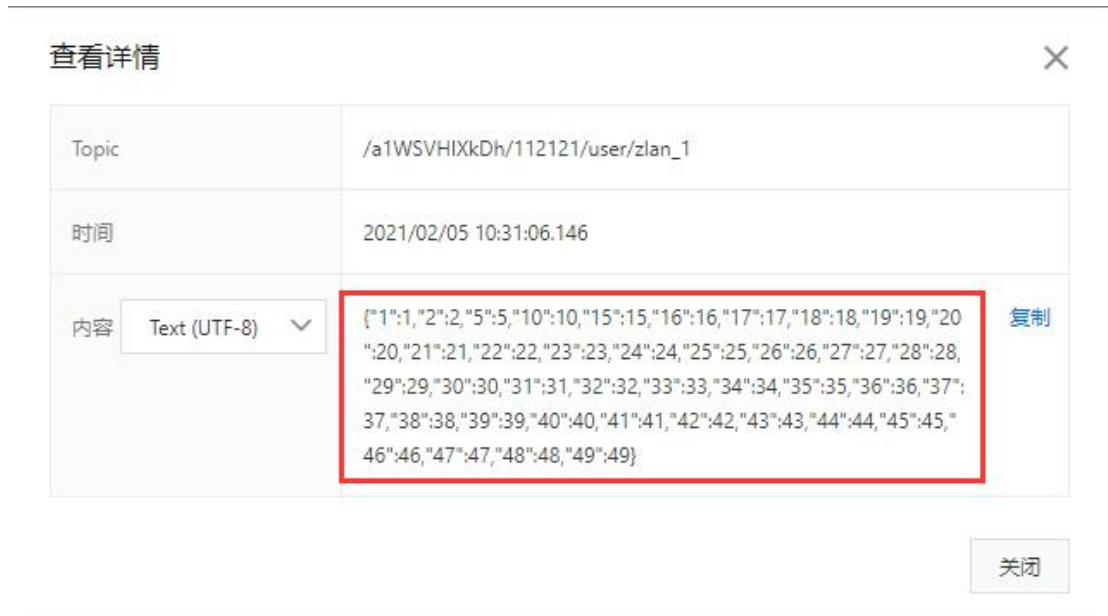


Chart 35 Serial port received Ali cloud data

## 6. AT Instruction

### 7.1. Login and configuration

#### 7.1.1. Enter configuration mode

Instruction: REQUEST CFG MODE

Function: Enter the configuration mode. Send this command during the startup phase of the device. After the device is started up, the device can enter the configuration mode.

Return: CFG MODE\r.

#### 7.1.2. Login

Instruction: ZL+LOGIN=666666\r\n

Function: Log in, modify device parameters need to be in the login state to succeed.

Return: +LOGIN:OK\r\n(success) or +LOGIN:NG\r\n(fail)

#### 7.1.3. Obtain the device state

Instruction: ZL+STAT?\r\n

Function: check signal strength, temperature, voltage of the device

Return: +STAT: Intensity, temperature, voltage\r\n

### 7.2. Serial port Data

#### 7.2.1. Obtain serial port parameters

Instruction: ZL+BAUD?\r\n

Function: Obtain baud rate

Return: +BAUD:n\r\n, n Represents the specific baud rate

#### 7.2.2. Obtain check bits

Instruction: ZL+CHECKB?\r\n

Function: Get check bit

Return: +CHECKB:N/O/E\r\n

N: No check

O:even parity check

E:Odd parity

### 7.2.3. Obtaining data bits

Instruction: ZL+DATAB?\r\n

Function: Obtain digit bits

Return: +DATAB:5/6/7/8\r\n

### 7.2.4. Set serial port parameters

Instruction: ZL+BAUD=n\r\n

Function: Set the baud rate, where n represents the value to set

Return: +BAUD:n\r\n

### 7.2.5. Set Check digit

Instruction: ZL+CHECKB= N/O/E \r\n

Function: Set check bit

Return: +CHECKB:N/O/E\r\n

N: No check

O: even parity check

E: Odd parity

### 7.2.6. Set data bits

Instruction: ZL+DATAB=5/6/7/8\r\n

Function: Set data bits

Return: +DATAB:5/6/7/8\r\n

## 7.3. Network parameters

### 7.3.1. Obtain the destination IP or domain name

Instruction: ZL+PIPADD?\r\n

Function: Get the destination IP or domain name

Return: +PIPADD=ip\r\n

### 7.3.2. Obtain the destination port

Instruction: ZL+PPORT?

Function: Get destination port

Return: +PPORT=n\r\n

### 7.3.3. Obtain the operating mode of the device

Instruction: ZL+PROTOCOL?

Function: Get the device operating mode

Return: +PROTOCOL=TCP/UDP\r\n

### 7.3.4. Get the IP of DNS server

Instruction: ZL+PDNS?

Function: Get the DNS server IP address

Return: +PDNS=ip\r\n

### 7.3.5. Set destination IP or domain name

Instruction: ZL+PIPADD=ip\r\n

Function: Set the destination IP or domain name

Return: +PIPADD=ip\r\n

### 7.3.6. Set destination port

Instruction: ZL+PPORT=n\r\n

Function: Set destination port

Return: +PPORT=n\r\n

### 7.3.7. Setting working mode

Instruction: ZL+PROTOCOL=TCP/UDP \r\n

Function: Setting up working mode

Return: +PROTOCOL=TCP/UDP\r\n

### 7.3.8. Set up DNS server IP

Instruction: ZL+PDNS=ip\r\n

Function: Set the DNS server IP address

Return: +PDNS=ip\r\n

## 7.4. Registration Pack and Heartbeat Pack

### 7.4.1. Check the contents of the registration package

Instruction: ZL+ENROL?\r\n

Check registry package contents (default registry package hexadecimal)

Return: +ENROL:1234567890\r\n

### 7.4.2. Enable the registration package

Check whether to enable the registration package (1 enabled 0 disabled) 返回 :

+EN\_ENROL:1\r\n

### 7.4.3. Set up the contents of the registration package

Instruction: ZL+ENROL=123456\r\n

Set the contents of the registration package (default registration package hexadecimal) to the actual registration package 0X12 0X34 0X56

Return: +ENROL:123456\r\n

### 7.4.4. Enable/Inable Registration Packs

Instruction: ZL+EN\_ENROL=1\r\n

Enable/disable registration package, 1 for enable, 0 for disabled

Return: +EN\_ENROL:1\r\n

## 7.5. Remote management function

### 7.5.1 Check remote management function

Instruction: ZL+Z\_RMT\_MAG?\r\n

Check if remote management function is enabled, 1 is enabled, 0 is disabled

Return: + ZL+Z\_RMT\_MAG:1\r\n

### 7.5.2 Turn on the remote management function

Instruction: ZL+Z\_RMT\_MAG=1\r\n

Enable/Disable remote management function, 1 is enabled, 0 is disabled

Return: + ZL+Z\_RMT\_MAG:1\r\n

### 7.5.3 Check the IP of the remote administration server

Instruction: ZL+Z\_RMT\_IP ?\r\n

Check the remote administration server IP \*\*\*\*\*

Return: + ZL+Z\_RMT\_IP =\*\*\*\*\*\r\n

### 7.5.4 Check the remote administration server port

Instruction: ZL+ Z\_RMT\_PORT ?\r\n

Check the remote administration server port\*\*\*\*

Return: + ZL+ Z\_RMT\_PORT =\*\*\*\*\r\n

Set the IP of the remote management server

Instruction: ZL+Z\_RMT\_IP =\*\*\*\*\*\r\n

Set up the remote administration server IP

Return: + ZL+Z\_RMT\_IP:\*\*\*\*\*\r\n

### 7.5.5 Set up the remote administration server port

Instruction: ZL+ Z\_RMT\_PORT =\*\*\*\*\r\n

Set up the remote administration server port

Return: + ZL+ Z\_RMT\_PORT:\*\*\*\* \r\n

## 7.6. MQTT Specifications

### 7.6.1 Set the MQTT user name

Instruction: ZL+ MQTT\_USERNAME =\*\*\*\*\r\n

Set MQTT user name

Return: + ZL+ MQTT\_USERNAME:\*\*\*\* \r\n

### 7.6.2 Set MQTT password

Instruction: ZL+ MQTT\_PASSWD =\*\*\*\*\r\n

Set MQTT password

Return: + ZL+ MQTT\_PASSWD:\*\*\*\* \r\n

### 7.6.3 Set the MQTT client ID

Instruction: ZL+ MQTT\_CLIENT =\*\*\*\*\r\n

Set the MQTT client ID

Return: + ZL+ MQTT\_CLIENT:\*\*\*\* \r\n

#### 7.6.4 Set up the MQTT publishing topic

Instruction: ZL+ MQTT\_PUBLISH\_TOPIC =\*\*\*\*\r\n

Set up the MQTT publishing topic

Return: + ZL+ MQTT\_PUBLISH\_TOPIC:\*\*\*\* \r\n

#### 7.6.5 Set up MQTT subscription topics

Instruction: ZL+ MQTT\_SUBSCRIBE\_TOPIC =\*\*\*\*\r\n

Set up the MQTT subscription topic

Return: + ZL+ MQTT\_SUBSCRIBE\_TOPIC:\*\*\*\* \r\n

#### 7.6.6 Set the quality of MQTT subscriptions

Instruction: ZL+ MQTT\_SUBSCRIBE\_QOS =1\0\r\n

Sets the MQTT subscription quality 1\0

Return: + ZL+ MQTT\_SUBSCRIBE\_QOS: 1\0\r\n

#### 7.6.7 Set the quality of MQTT release

Instruction: ZL+ MQTT\_PUBLISH\_QOS =1\0\r\n

Sets the MQTT publishing quality 1\0

Return: + ZL+ MQTT\_PUBLISH\_QOS: 1\0\r\n

#### 7.6.8 Set the MQTT guarantee time

Instruction: ZL+MQTT\_KEEPALIVE =\*\*\*\*\r\n

Set the MQTT warranty time

Return: + ZL+ MQTT\_KEEPALIVE:\*\*\*\* \r\n

#### 7.6.9 Set up the MQTT wish subject

Instruction: ZL+ MQTT\_WILL\_TOPIC =\*\*\*\*\r\n

Set up the MQTT wish topic

Return: + ZL+ MQTT\_WILL\_TOPIC:\*\*\*\* \r\n

#### 7.6.10 Set up the MQTT wish information

Instruction: ZL+ MQTT\_WILL\_MESSAGE =\*\*\*\*\r\n

Set up the MQTT wish information

Return: + ZL+ MQTT\_WILL\_MESSAGE:\*\*\*\* \r\n

## 7. Accessories

### 1、 Model No.:

Model No.	Function	Specifications
ZLAN8308	4G to serial port	No RJ45

### 2、 Antenna selection:

The adhesive stick antenna or sucker antenna can be selected. The sucker antenna is 2M by default, and the 3M antenna can be customized。

### 3、 Power input:

The default is plug type Q2.1 socket, which can be customized as terminal type power input。

### 4、 Guide mounting accessories are available。

## 8. After sales service

Shanghai ZLAN Information Technology Co., Ltd.

Address: 12 floor, D building, No. 80 CaoBao road, Xuhui District, Shanghai, China

Phone: 021-64325189

Fax: 021-64325200

Web: <http://www.zlmcu.com>

Email: [support@zlmcu.com](mailto:support@zlmcu.com)