USR-TCP232-S2 Software Manual

File version:V1.0.1

Jinan USR IOT Technology Limited　　　　　　　**www.usriot.com**

1. Overview

1.1. Brief Introduction

TCP232-S2 is a new and tiny size serial to Ethernet module which realizes data bidirectional transparent transmission between TTL Port and RJ45 Port, it can also used in RS232/ RS485 by level shift circuit.

S2 is equipped with Cortex-M0 core. It has characters of low power, fast speed, high efficiency, strong compatibility, it is easy to use.

2. Function

2.1. Configuration

User should set parameter as follows:

❖ Work mode:
  ◆ TCP Client、TCP Server、UDP Client、UDP Server、HTTPD Client
❖ Default TCP/UDP connection parameter
  ◆ Connection type (Server, Client, HTTPD Client）
  ◆ Target Port
  ◆ Target IP address
  ◆ Local Port
❖ Serial Port parameter
  ◆ Baud rate
  ◆ Data bit
  ◆ Check bit
  ◆ Stop bit
  ◆ RS485 function
❖ IP address and module password
  ◆ The way of getting IP address.
  ◆ Module name and password

After setting parameter then reset the module .

2.2. Work mode

USR-TCP232-S2 work mode : TCP Client, TCP Server, UDP Client, UDP Server, HTTPD Client.
Work mode can be changed by webpage or software.

2.2.1. Connection diagram:
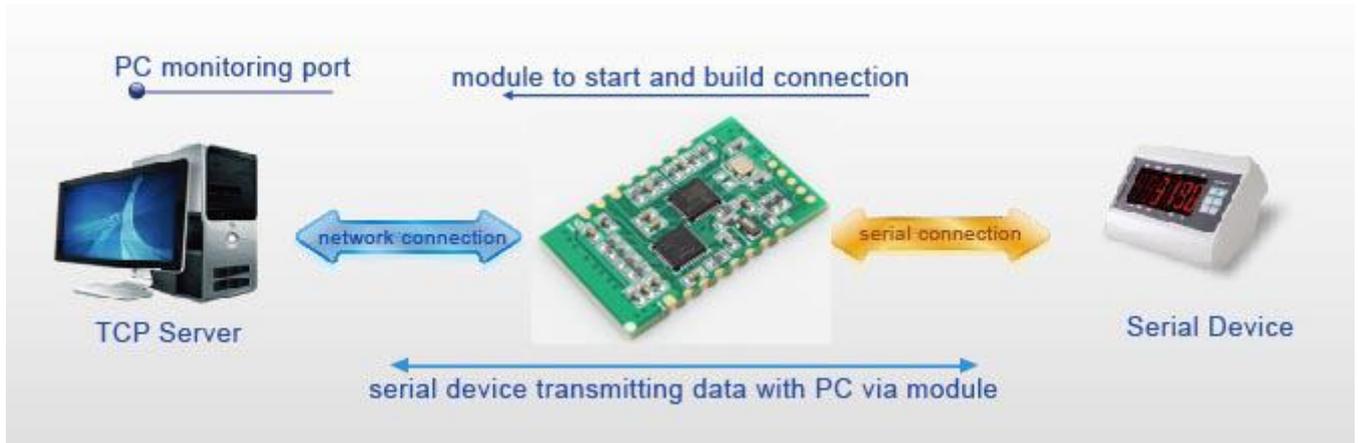
Diagram 1

Note: For security , module only receive data from set target IP and Port., and send data to set target IP And Port .

2.2.2. TCP Client mode

It has to be connected before transferring data.

1) In TCP Client Mode,TCP232-S2 connects TCP Server actively, establish a connection to transmit date

2) In TCP Client Mode, It has function of identifying disconnected link. When connected, it will send keepalive package every 15s. If unconnected ,it can be detected timely and enforceTCP232-S2 to disconnect the former link to establish a new one.

3) When TCP232-S2 try to connect remote server ,if the local port number is not "0" ,it will establish a connection with the same source port every time.

4) It has synchronizing function of baud rate, user should install USR VCOM Software.

5) When local port number is "0" , it means local port is random.
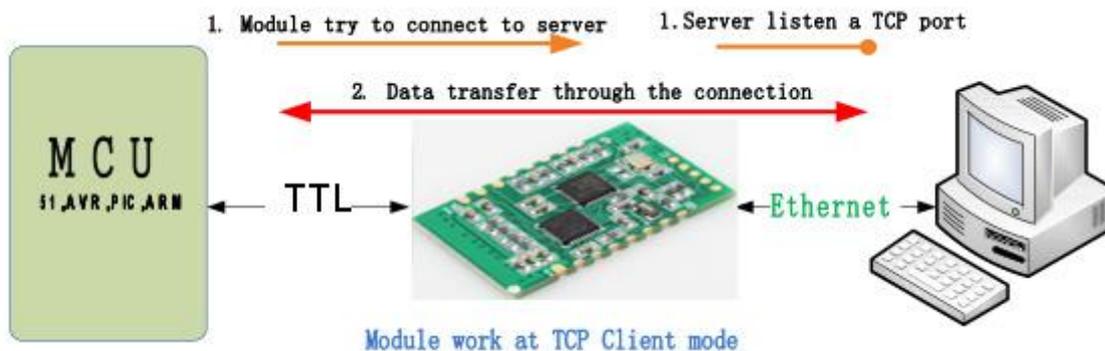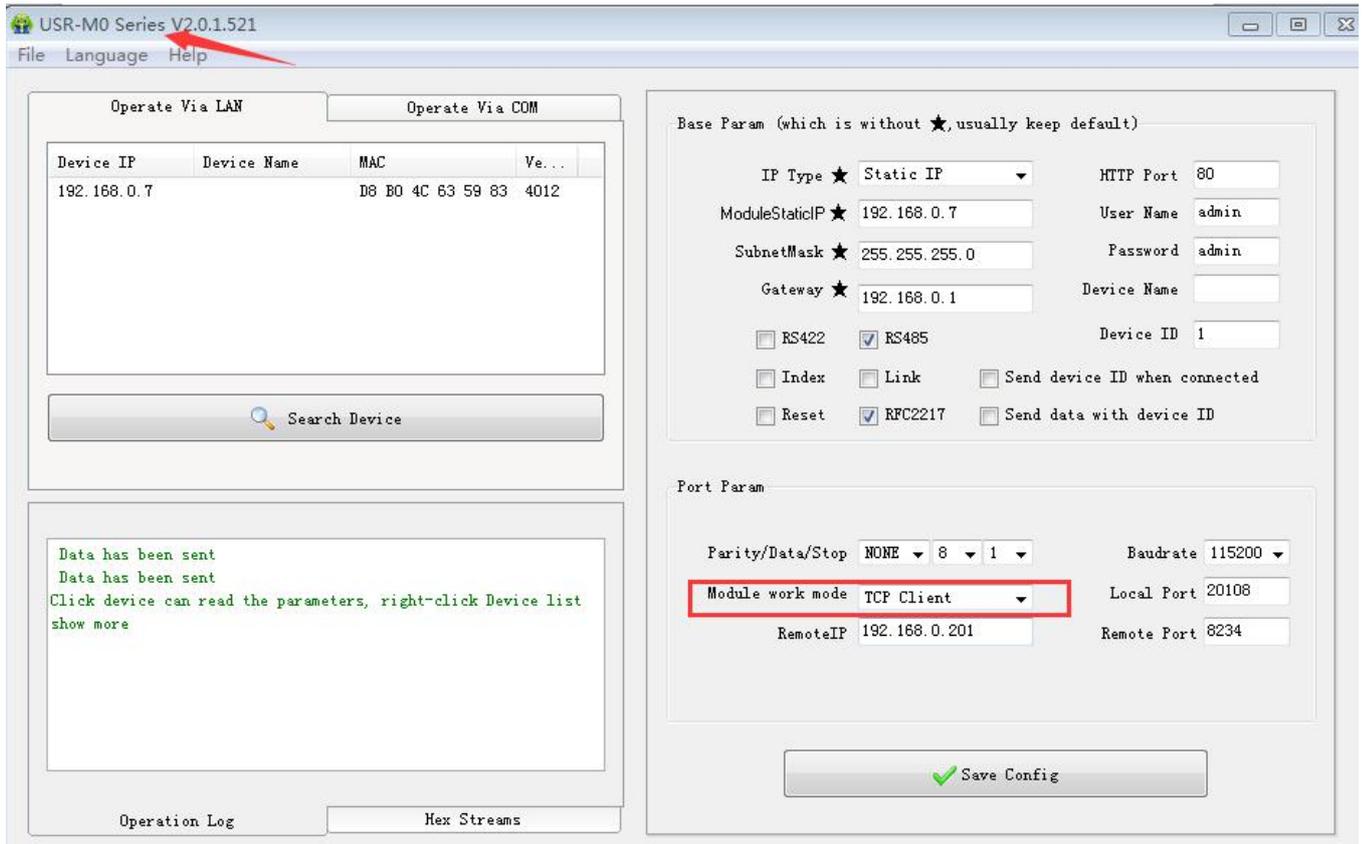
Diagram 2

2.2.3. TCP Server mode

1) It has to be connected before transferring data.

2) In TCP Server Mode, S2 monitors local port, it will response and establish a connection when there is a request. Up to 4 links at the same time. Once received data, S2 serial port will send data to all the devices which connect to TCP232-S2.

3) It has synchronizing function of baud rate, user should install USR VCOM Software.

4) In TCP Server Mode, The maximum number can be configured by user. TCP Client number is from 1 to 16, default value 4. When the Client link is more than 16, the new link will replace the former link from Link 1 If the Client link is more than 4, send and receive data at same time , the data flow should be within 2.5 KB/s

USR-TCP232-S2 work at TCP server mode

1. module listen TCP port

1. PC connect to the port

2. data transfer from MCU to PC

New pc connected

Data transfer go and back

It can accept 4 links at the same time
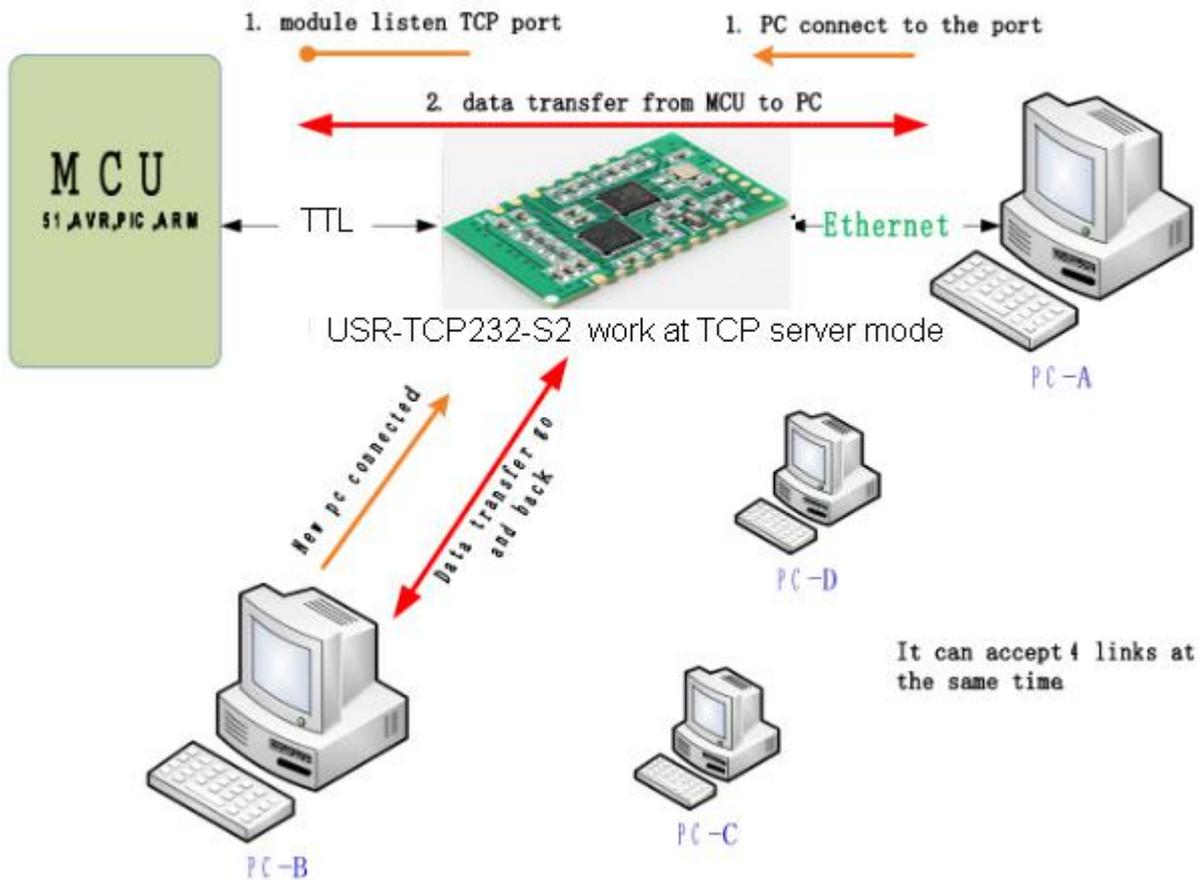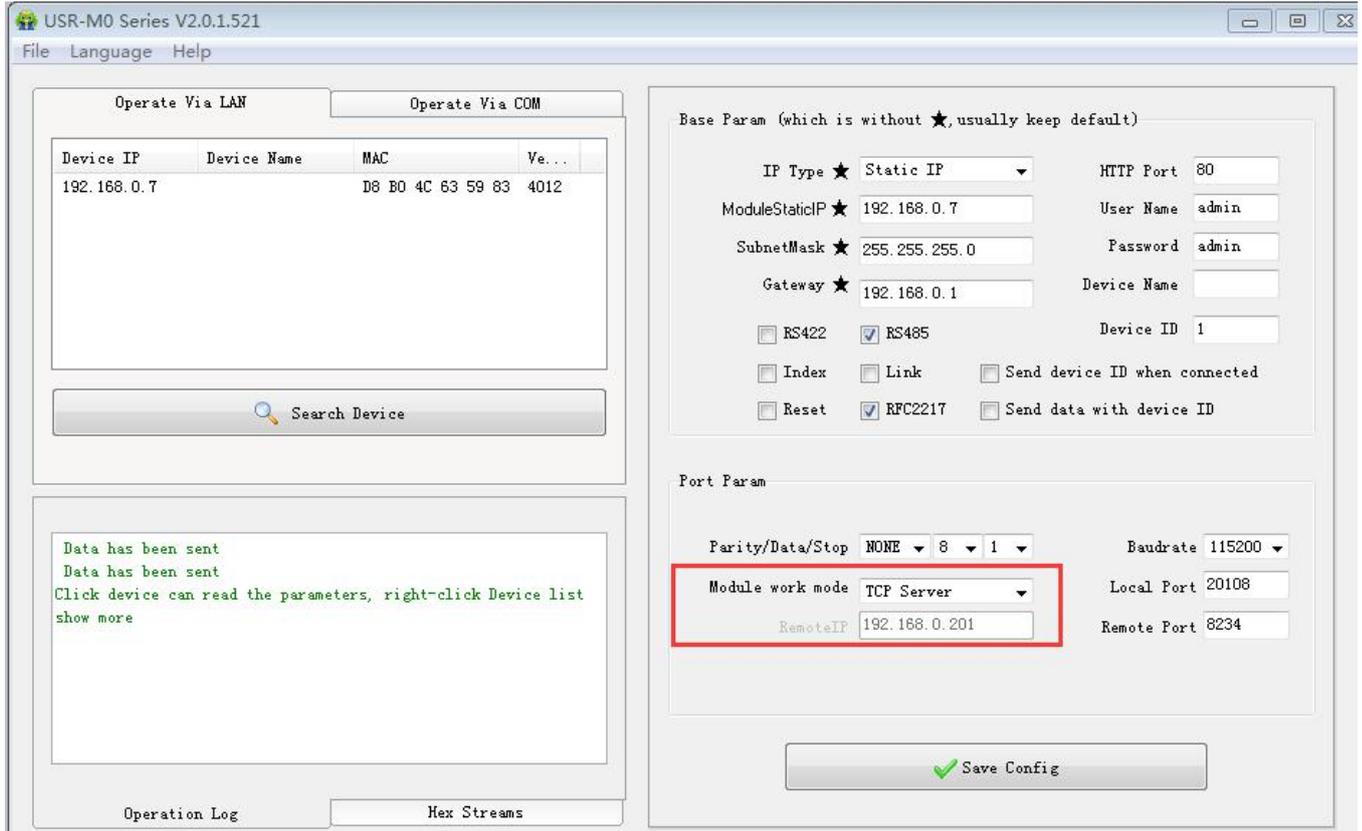
Diagram 3

2.2.4. UDP Client mode

1) The mode belongs to UDP Protocol.

2) In UDP Client Mode, TCP232-S2 won't establish the connection actively. It can only communicate with the target port whose IP has been set. When serial port receive data, it send data to target IP and port. If data doesn't come from this channel, it will not be accepted by TCP232-S2.

3) In UDP Client Mode, if target IP is set as 255.255.255.255, it can realize function of entire network broadcast, also can receive broadcast data. If broadcast in network segment ,eg.192.168.0.255, it can only send data ,can't receive data.

4) Under UDP Client, maximum data length sent from MCU to TCP232-S2 is 1460.
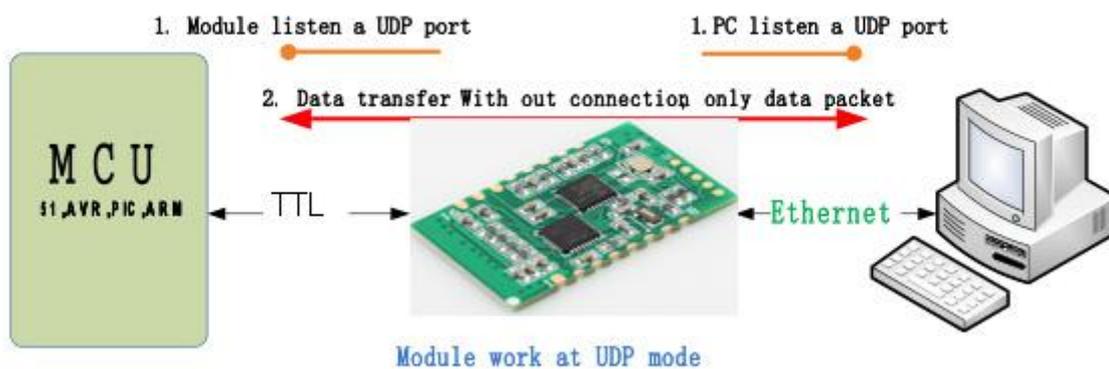
Diagram 4

2.2.5. UDP Server mode

1) UDP Server is based on normal UDP, it doesn't validate the source of IP address. Once received UDP data, it convert target IP to data source IP, similar to TCP Server.

2) In UDP Server Mode, TCP232-S2 records an IP, Once it receives data, it will send data to record IP. TCP232-S2 also works as a server, can receive data from Ethernet and convert target IP to data source IP.

Diagram 5

2.2.6. HTTPD Client

This function is used for developer.

1) Module S2 send data to HTTP Server or receive from HTTP Server, complex HTTP protocol will be done by S2, it is convenient for user to programming.

2) S2 received data from HTTP Server will send to serial port without process.

3) According to demand, user can define HTTP content., user can add,delete,modify every HTTP Content (if HTTP request is post, S2 will add Connection and Content-Length automatically)

2.2.7. TCP and UDP mechanism

Table 1   TCP and UDP mechanism

	TCP	UDP
Advantages	Stable; Not easy to lose data package; Reliable connection mechanism;	Transmission interval is accurate; No connection mechanism; Easy and flexible;
Disadvantage	Easy to block up Information;	Under bad network condition, it is

	Because of check and resend mechanism, interval isn't accurate	high risky to losing data package

2.3. Configuration Protocol

Configuration flow: to configure protocol cross segment ,all communication protocol is set by UDP Broadcast, Little-endian .In UDP communication,target port number should be 1500. Local port number is random. All communication protocol is UDP broadcast.

2.3.1. Network setting protocol command

Table-2- Query command

Function	Head of data package	Length (command-1 bite)	Command	MAC address (6 bite)	User name and (12 bite)	Parameter	Check bit (sum)
Search	FF	01	01	-	-	-	02
Restart	FF	xx	02	[MAC]	[usrname] [password]	-	xx
Read configuration	FF	xx	03	[MAC]	[usrname] [password]	-	xx
Basic setting	FF	xx	05	[MAC]	[usrname] [password]	Basic parameter	xx
Port 0 setting	FF	xx	06	[MAC]	[usrname] [password]	Port parameter	xx
Factory Default	FF	xx	0b	[MAC]	[usrname] [password]	-	xx
Cloud setting	FF	xx	0c	[MAC]	[usrname] [password]	Port parameter	xx
Heartbeat package Setting	FF	xx	21	[MAC]	[usrname] [password]	Port parameter	xx
Extension setting	-	-	-	-	-	-	-

To confirm command accuracy ,we set algorithm of sending data and check method: sum check . From length byte (include length) to check bit (not include check) ,the sum is check value, only keep lower byte.

Search command : Fixed command FF 01 01 02, sum check 02=01+01

Restart command : send (22 byte) FF 13 02 00 71 77 7C 42 2F 61 64 6D 69 6E 00 61 64 6D 69 6E 00 FC,

sum check: FC=13+02......6E+00,

From 4th to 9th is TCP232-T2 MAC address. The last 12 byte before check bit is user name and password, both is 6 byte, if not, write 0 (the last byte of user name and password should be 0 , the

following name and password is the same rule)

Read Configuration command :send (22 byte) FF 13 03 00 71 77 7C 42 2F 61 64 6D 69 6E 00 61 64 6D 69 6E 00 FD,

    FD= 13 + 03 + 00 + ... + 6E + 00

    From 4th to 9th is TCP232-T2 MAC address. The last 12 byte before check bit is user name and password, both is 6 byte, if not, write 0.

Basic setting command: it includes 67 basic parameter.

    E.g : send FF 56 05 00 71 77 7c 42 2F 61 64 6d 69 6e 00 61 64 6d 69 6e 00 95 63 03 00 00 00 50 00 00 07 00 a8 c0 c9 00 a8 c0 00 FF FF FF 55 6c 6c 2d 66 32 00 00 00 00 00 00 00 00 00 00 00 61 64 6d 69 6e 00 61 64 6d 69 6e 00 00 01 00 00 00 00 00 00 00 00 00 00 00 00 00 00 04, check byte uses sum check method 04 = 56 + 05 + 00+ ... + 01;

    From 4th to 9th is TCP232-T2 MAC address,10th to 22th is user name and password .The following byte is basic setting parameter,the last byte is sum check byte.

Table 3 - basic setting command

Name	Byte	Example	Instruction
ucSequenceNum	1	00	Reserve package head
ucCRC	1	00	Reserve package head
ucVersion	1	00	Reserve package head

ucFlags	1	80	8th 0:DHCP；
			       1:Static IP
			7 th 0:Turn off factory default function
			       1:Turn on factory default function

usLocationURLPort	2	00 00	Not enabled, reserved protocol
usHTTPServerPort	2	50 00	HTTP service port
ucUserFlag	1	00	Not enabled, reserved protocol
ulStaticIP	4	07 00 A8 C0	Static IP
ulGatewayIP	4	C9 00 A8 C0	Gateway
ulSubnetMask	4	00 FF FF FF	Subnet mask
ucModName	14	55 53 52 2D 4B 32 00 00 00 00 00 00 00 00	Module name
Protocol reserved	2	00 00	Must be 0
username	6	61 64 6D 69 6E 00	User name
password	6	61 64 6D 69 6E 00	Password
ucNetSendTime	1	00	Not enabled, reserved protocol
uiId	2	01 00	Device ID

ucIdType	1	A4	bit0(1): send ID when connecting ;
			bit1(2): send ID when sending data;
			bit2(4): RS485;
			bit4(16): Reset;
			bit5(32):Link-state;
			bit6(64): index function;
			bit7(128):Similar RFC2217 function,

mac_addrs	6	00 00 00 00 00 00	Device Mac address（only support query）
DNSGatewayIP	4	01 00 A8 C0	DNS address（only support query）
ucReserved	4	00 00 00 00	Unused

Port configuration: The parameter which is used to configure Lan port and Serial port is 63 byte .

　　E.g.: send (63 byte) FF 52 06 00 71 77 7C 42 2F 61 64 6D 69 6E 00 61 64 6D 69 6E 00 00 C2 01 00 08 01 01 01 00 00 00 00 8C 4E 2A 20 31 39 32 2E 31 36 38 2E 30 2E 32 30 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 c9 00 A8C0 00 01 00 00 00 00 00 00 00 05 00 00 00 00 F0,

　　Check byte method : F0=52 + 06 + ... + 00 ;

　　From 4th to 9th is TCP232-T2 MAC address,10th to 22th is user name and password.the following is port parameter +1 byte and check bit .

Table 4 - Port setting command

Name	Byte	Example	Instruction
ulBaudRate	4	00 C2 01 00	Serial port baud rate
ucDataSize	1	08	Serial port data bite (0X05/0x06/0x07/0x08)
ucParity	1	01	Serial port check bite 1:no, 2:odd, 3:even, 4:mark, 5:space
ucStopBits	1	01	Serial port stop bite (0x01/0x02)
ucFlowControl	1	00	Not enabled, reserved protocol
ulTelnetTimeout	4	00 00 00 00	Not enabled, reserved protocol
usTelnetLocalPort	2	8C 4E	Local port
usTelnetRemotePort	2	2a 20	Remote port
uiTelnetURL	30	31 39 32 2E 31 36 38 2E 30 2E 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	IP address or domain name send by ASCII E.g.: 192.168.0.1
ulTelnetIPAddr	4	00 00 00 00	Not enabled,
ucFlags	1	00	Cloud function: For 5th , 0: turn off cloud　　　1: turn on cloud
ucWorkMode	1	01	Work mode : 0: UDP, 1:TCP Client, 2: UDP Server, 3: TCP Server 4: HTTPD Client
HTPucFlags	1	00	0：HTTPD GET;　1:HTTPD POST
tc_number	1	04	Client number that TCP Server can connect
uiPackLen	2	00 00	Not enabled, reserved protocol
ucPackTime	1	0A	Serial port packing time
ucTimeCount	1	91	Write the read data as it is .
ucReserved	5	00 00 00 00 00	Write 00 00 00 00 00. For extension use

Factory default : send FF 13 0B 00 71 77 7C 42 2F 61 64 6D 69 6E 00 61 64 6D 69 6E 00 07;

　　Sum check: 07 = 13 + 0B+ ... + 6E + 00

From 4th to 9th is TCP232-T2 MAC address,10th to 21th is user name and password , both is 6 byte ,if not , write 0.

Cloud function command : send FF 13 0c 00 71 77 7c 42 2F 61 64 6d 69 6e 00 61 64 6d 69 6e 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 08;

Sum check：07 = 13 + 0c + ... + 6E + 00+...+00;

From 4th to 9th is TCP232-T2 MAC address,10th to 21th is user name and password , both is 6 byte ,if not , write 0. The following is Cloud ID and password, the last one is check bit.

Heartbeat package function: send FF 13 21 00 71 77 7c 42 2F 61 64 6d 69 6e 00 61 64 6d 69 6e 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 1d;

Sum check: 1d = 13 + 21 + ... + 6E + 00+...+00;

From 4th to 9th is TCP232-T2 MAC address,10th to 21th is user name and password, both is 6 byte ,if not , write 0. The following is heartbeat and registration Sending direction,heartbeat time, heartbeat package length,registration package length,heartbeat package,registration package,heartbeat package and registration package are both 40 byte, if not,write 0, the last one is check bit.

Table 5 - Heartbeat&registration package command

Name	Byte	Example	Instruction
H_R_ucFlags	1	00	1th is 1 : heartbeat package to Lan port. 2th is 1: heartbeat package to Serial port. 3th is 1: send registration package. 4th is 1: data carry registration package. 5th is 1: httpd remove package head. 6th is 1: turn on serial port setting parameter function. 7th is 1: input heartbeat package 16 binary system. 8th is : input registration package 16 binary system.
heart_times	1	3c	heartbeat time
heart_len	1	00	heartbeat package length
register_len	1	00	registration package length
heartbeat	40	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	heartbeat package content

register_s	40	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	registration package content

2.3.2. Network echo command.

Search command return result : (36 byte) FF 24 01 00 00 c0 a8 00 07 00 71 77 7c 42 2F 01 0c 00 00 55 53 52 2d 4b 32 00 00 00 00 00 00 00 00 00 00 00 F2

Initial value is 0X00, subtract each byte in sequence.

F2= 00 - FF - 24 - 01 - 00 - 4B - ... - 31 - 00 – 00.

Table 6 - Search instruction reply command

Name	Byte	Example	Instruction
TAG_STATUS	0	FF	Fixed number
Packet_length	1	24	Fixed number
CMD_DISCOVER_TARGET	2	01	Fixed number
Board_type	3	00	Fixed number
Board_ID	4	00	Fixed number
Client_IP_address	5~8	C0 A8 00 07	Device IP（big-endian）
MAC_address	9~14	AC CF 23 20 FE 3D	Device MAC（big-endian）
Firemware_version	15~18	01 00 00	Firmware version
Application_title	19~34	55 53 52 2D 4B 32 00 00 00 00 00 00 00 00 00 00	Device name
checksum	35	F0	Initial value is 0x00, subtract TAG_STATUS byte in sequence u until the last byte. Result is checksum

Return result of restart command: (4 byte)

 FF 01 02 4B ,if user name and password is correct 4B = 'K'

 FF 01 02 50 ,user name and password is wrong   50 ＝'P'

Return result of read command: return result is all parameters of TCP232-T2 ,264 byte

Not include check and protocol ,return parameter directly. Refer to Table 2-4. Return result

In 3 package. 130 byte, 50 byte, 84 byte.

e.g.:   95 63 03 00 00 00 50 00 00 07 00 A8 C0 01 00 A8 C0 00 FF FF FF 55 53 52 2D 54 43 50 32 33 32 2D 33 31 30 00 00 61 64 6D 69 6E 00 61 64 6D 69 6E 00 00 01 00 A4 AC CF 23 20 FE 10 00 00 00 00 00 00 00 00 80 25 00 00 08 01 01 01 00 00 00 00 8C 4E 2A 20 31 39 32 2E 31 36 38 2E 31 2E 31 33 33 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 85 01 A8 C0 00 01 00 00 00 00 00 00 00 00 00 00 00;

00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

00 30 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00;

If password is wrong ,it returns FF 01 03 50.

Return result of basic parameter setting :

If correct ,it return   FF 01 05 4B.   If password is correct 4B = 'K'
                          FF 01 05 50   If password is wrong 50 ＝'P'

Return result of serial port parameter setting :

If correct, it return   FF 01 06 4B   If password is correct 4B = 'K'
                           FF 01 06 50   If password is wrong 50 ＝'P'

Return result of cloud parameter setting :

If correct, it return   FF 01 06 4B   If password is correct 4B = 'K'
                           FF 01 06 50   If password is wrong 50 ＝'P'

Return result of heartbeat registration parameter setting :

If correct, it return   FF 01 06 4B   If password is correct 4B = 'K'
                           FF 01 06 50   If password is wrong 50 ＝'P'

Other return result:

If check sum is wrong , it return "E" + right check sum.

Right operation: FF 01 CMD 'K '

If user name and password is wrong , it return FF 01 CMD 'P'

Other is wrong ,it returns FF 01 CMD'E'

2.3.3. Monitor function

If user want to monitor message, please do as follows with USR-M0 software.
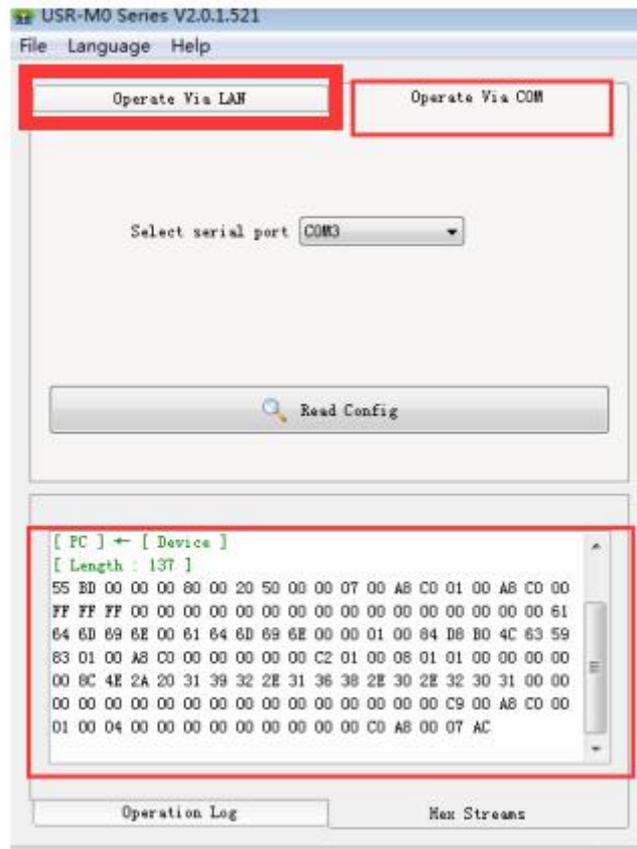
Diagram 7

2.3.4. Serial port setting protocol

Firstly ,user connect Reload(CFG) to the ground, user can refer to setting command or use software ,sending data baud is 9600, 8 data bit, 1 stop bit, （9600,n,8,1）.

No matter what the baud rate ever is , baud rate switches to 9600 idiomatically, send 'U' to port, ,it means In the status of configuration, after receiving complete data package and check is right ,it returns "K" , if check Is wrong, it returns "E" and calculated check bit, this bit is important when sending data by manual test. For Other error,it just returns 'E.

Restore factory setting Reload(CFG) and port setting Reload(CFG) is the same key. If user want to restore factory setting,   firstly short circuit Reload(CFG) , and port can't send data until 5 seconds later. <Note> after setting ,please pull up Reload(CFG) or don't connect it ,then module can work again.

Data package head for port setting protocol has 3 types:

① 55BE write port0 configuration.
② 55 BD read T2 configuration.
③ 55 BE write basic configuration.
④ 55 BB write extended function parameter configuration.
⑤ 55 B1 5A restart command;

Basic configuration : It includes some serial port parameter configuration. Total 67 byte, check bit xx= 00 + 00 +00+80 ... + 00. Send 55 BE+basic parameter+check bit, If configuration is right ,it returns BE 4B, otherwise ,it returns BE 45.

Table 7 - Port setting command

Name	Byte	Example	Instruction
ucSequenceNum	1	00	Reserve package head
ucCRC	1	00	Reserve package head
ucVersion	1	00	Reserve package head

ucFlags	1	80	8th 0:DHCP；
			        1:Static IP
			7 th 0:Turn off factory default function
			        1:Turn on factory default function

usLocationURLPort	2	00 00	Not enabled, reserved protocol
usHTTPServerPort	2	50 00	HTTP service port
ucUserFlag	1	00	Not enabled, reserved protocol
ulStaticIP	4	07 00 A8 C0	Static IP
ulGatewayIP	4	C9 00 A8 C0	Gateway
ulSubnetMask	4	00 FF FF FF	Subnet mask
ucModName	14	55 53 52 2D 4B 32 00 00 00 00 00 00 00 00	Module make
Protocol reserved	2	00 00	Must be 0
username	6	61 64 6D 69 6E 00	User name
password	6	61 64 6D 69 6E 00	Password
ucNetSendTime	1	00	Not enabled, reserved protocol
uiId	2	01 00	Device ID

ucIdType	1	A4	bit0(1): send ID when connecting ;
			bit1(2): send ID when sending data;
			bit2(4): RS485;
			bit4(16): Reset;
			bit5(32):Link-state;
			bit6(64): index function;
			bit7(128):Similar RFC2217 function,

mac_addrs	6	00 00 00 00 00 00	Device Mac address（only support query）
DNSGatewayIP	4	01 00 A8 C0	DNS address（only support query）
ucReserved	4	00 00 00 00	Unused

Port setting command: it include 65 byte, sum check method . Send 55 BF+basic parameter + check bit. If correct ,it returns BE 4B. Otherwise, it returns BE 45.

Table 8 - Port setting command

Name	Byte	Example	Instruction
ulBaudRate	4	00 C2 01 00	Serial port baud rate
ucDataSize	1	08	Serial port data bite (0X05/0x06/0x07/0x08)
ucParity	1	01	Serial port check bite 1:no, 2:odd, 3:even, 4:mark, 5:space

ucStopBits	1	01	Serial port stop bite (0x01/0x02)
ucFlowControl	1	00	Not enabled, reserved protocol
ulTelnetTimeout	4	00 00 00 00	Not enabled, reserved protocol
usTelnetLocalPort	2	8C 4E	Local port
usTelnetRemotePort	2	2a 20	Remote port
uiTelnetURL	30	31 39 32 2E 31 36 38 2E 30 2E 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	IP address or domain name send by ASCII E.g.: 192.168.0.1
ulTelnetIPAddr	4	00 00 00 00	Invalid byte , arbitrary configuration
ucFlags	1	00	Cloud function: For 5th , 0: turn off cloud 1: turn on cloud
ucWorkMode	1	01	Work mode : 0: UDP, 1:TCP Client, 2: UDP Server, 3: TCP Server 4: HTTPD Client
HTPucFlags	1	00	0:HTTPD GET; 1:HTTPD POST
tc_number	1	04	Client number that TCP Server can connect
uiPackLen	2	00 00	Not enabled, reserved protocol
ucPackTime	1	0A	Serial port packing time
ucTimeCount	1	91	Write the read data as it is .
ucReserved	5	00 00 00 00 00	Write 00 00 00 00 00. For extension use

Read configuration command : TCP232-T2 send 55 BD, return value is 137 byte, there is no check bit. TCP232-T2 return parameter is as follows.

Table 9 - Read configuration command

Name	Byte	Example	Instruction
ucSequenceNum	2	55 bd	Reserve package head
ucCRC	1	00	Reserve package head
ucVersion	1	00	Reserve package head
ucFlags	1	80	8th 0:DHCP; 1:Static IP 7th 0:Turn off factory default function 1:Turn on factory default function
usLocationURLPort	2	00 00	Not enabled, reserved protocol
usHTTPServerPort	2	50 00	HTTP service port
ucUserFlag	1	00	Not enabled, reserved protocol
ulStaticIP	4	07 00 A8 C0	Static IP
ulGatewayIP	4	C9 00 A8 C0	Gateway
ulSubnetMask	4	00 FF FF FF	Subnet mask
ucModName	14	55 53 52 2D 4B 32 00 00 00 00 00 00 00 00	Module name

Protocol reserved	2	00 00	Must be 0
username	6	61 64 6D 69 6E 00	User name
password	6	61 64 6D 69 6E 00	Password
ucNetSendTime	1	00	Not enabled, reserved protocol
uiId	2	01 00	Device ID
ucIdType	1	A4	bit0(1): send ID when connecting ; bit1(2): send ID when sending data; bit2(4): RS485; bit4(16): Reset; bit5(32):Link-state; bit6(64): index function; bit7(128):Similar RFC2217 function,
mac_addrs	6	00 00 00 00 00 00	Device Mac address（only support query）
DNS Gateway IP	4	01 00 A8 C0	DNS address（only support query）
ucReserved	4	arbitrary value	Unused
ulBaudRate	4	00 C2 01 00	Serial port baud rate
ucDataSize	1	08	Serial port data bite (0X05/0x06/0x07/0x08)
ucParity	1	01	Serial port check bite 1:no, 2:odd, 3:even, 4:mark, 5:space
ucStopBits	1	01	Serial port stop bite (0x01/0x02)
ucFlowControl	1	00	Not enabled, reserved protocol
ulTelnetTimeout	4	00 00 00 00	Not enabled, reserved protocol
usTelnetLocalPort	2	8C 4E	Local port
usTelnetRemotePort	2	2a 20	Remote port
uiTelnetURL	30	31 39 32 2E 31 36 38 2E 30 2E 31 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	IP address or domain name send by ASCII E.g.: 192.168.0.1
ulTelnetIPAddr	4	00 00 00 00	Not enabled,
ucFlags	1	00	Cloud function: For 5th , 0: turn off cloud 1: turn on cloud
ucWorkMode	1	01	Work mode : 0: UDP, 1:TCP Client, 2: UDP Server, 3: TCP Server 4: HTTPD Client
HTPucFlags	1	00	0：HTTPD GET; 1:HTTPD POST
tc_number	1	04	Client number that TCP Server can connect
uiPackLen	2	00 00	Not enabled, reserved protocol
ucPackTime	1	0A	Serial port packing time
ucTimeCount	1	91	Write the read data as it is .
ucReserved	5	00 00 00 00 00	Write 00 00 00 00 00. For extension use
Current IP	4	07 00 A8 C0	Current module IP address

Version	1	01	Version number

Serial port configuration extended function parameter: extended function 112 character, send 55 BB + special function character + check bit. If correct , it returns BB 4B. Otherwise It returns BB 45.

Table 10 - Serial port setting extended function parameter command

Name	Byte	Example	Instruction
Device ID	20	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	Module Cloud ID
CLOUD_password	8	00 00 00 00 00 00 00 00	Module Cloud password
H_R_ucFlags	1	00	1th is 1 : heartbeat package to Lan port. 2th is 1: heartbeat package to Serial port. 3th is 1: send registration package. 4th is 1: data carry registration package. 5th is 1: httpd remove package head. 6th is 1: turn on serial port setting parameter function. 7th is 1: input heartbeat package 16 binary system. 8th is : input registration package 16 binary system.
heart_times	1	3c	heartbeat time
heart_len	1	00	heartbeat package length
register_len	1	00	registration package length
heartbeat	40	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	heartbeat package content
register_s	40	00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00	registration package content

2.3.5. Similar T24 series setting protocol

Network command : Fixed format 40 byte data package, please refer to the following format and data , TCP232-T2 will return 35 bytes data package. 30 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 37 38 39 30 31 32 33 34 35 36 37 38 39. When setting parameter ,it send 40 bytes, returns 35 bytes.

Table 11 -Network command

Name	Byte	Example	Instruction
MAC	6	00 CE 83 25 4D 60	Module MAC that need to be set
Old password	6	31 31 30 34 31 35	Password for network module，110415 is initial password.
Target IP	4	C9 00 A8 C0	Target IP
Target Port	2	2A 20	Target Port
Module IP	4	07 00 A8 C0	Module IP
Module Port	2	8C 4E	Module Port
Gateway	4	C9 00 A8 C0	Gateway IP
Work mode	1	01	0:UDP Client, 1:TCP Client 2:UDP Server, 3:TCP Server
Baud rate	3	00 C2 01	Port baud rate
Port parameter bit	1	03	Data bit, stop bit, check bit
Independent ID	3	00 00 00	ID-H,ID-L,ID-type, if don't need, write 0 （ID type has addition meaning ,refer to appendixⅠ）
Subnet mask	4	00 FF FF FF	Subnet mask,Little Endian

Table 12 - Network return parameter

Name	Byte	Example	Instruction
MAC	6	00 CE 83 25 4D 60	Module MAC that need to be set
Version	1	42	Version
Target IP	4	C9 00 A8 C0	Target IP
Target Port	2	2A 20	Target Port
Module IP	4	07 00 A8 C0	Module IP
Module Port	2	8C 4E	Module Port
Gateway	4	C9 00 A8 C0	Gateway IP
Work mode	1	01	0:UDP Client, 1:TCP Client 2:UDP Server, 3:TCP Server
Baud rate	3	00 C2 01	Port baud rate
Port parameter bit	1	03	Data bit, stop bit, check bit refer to appendix 1)
Independent ID	3	00 00 00	ID-H,ID-L,ID-type, if don't need, write 0 （ID type has addition meaning ,refer to appendix 2）
Subnet mask	4	00 FF FF FF	Subnet mask,Little Endian

Port parameter set : Port parameter set of similar T24 set protocol.

Read parameter command: send 55BC to TCP232, return message and set

Parameter command as follows:

Table 13 -Read parameter return command

Name	Byte	Example	Instruction
Head of data package	2	55 BC	Head of data package
Target IP	4	C9 00 A8 C0	Target IP
Target Port	2	2A 20	Target Port
Module IP	4	07 00 A8 C0	Module IP
Module Port	2	8C 4E	Module Port
Gateway	4	C9 00 A8 C0	Gateway IP
Work mode	1	01	0:UDP Client, 1:TCP Client 2:UDP Server, 3:TCP Server
Baud rate	3	00 C2 01	Port baud rate
Port parameter bit	1	03	Data bit, stop bit, check bit
Independent ID	3	00 00 00	ID-H,ID-L,ID-type, if don't need, write 0（ID type has addition meaning ,refer to appendixⅠ）
Subnet mask	4	00 FF FF FF	Subnet mask,Little Endian
Firmware version	1	58	Firmware version lowest byte
Sum Check	1	B9	Sum check, from target IP to sum check (result keep low byte)

Table 14 - Set parameter command

Name	Byte	Example	Instruction
Head of data package	2	55 BA	Head of data package
Target IP	4	C9 00 A8 C0	Target IP
Target Port	2	2A 20	Target Port
Module IP	4	07 00 A8 C0	Module IP
Module Port	2	8C 4E	Module Port
Gateway	4	C9 00 A8 C0	Gateway IP
Work mode	1	01	0:UDP Client, 1:TCP Client 2:UDP Server, 3:TCP Server
Baud rate	3	00 C2 01	Port baud rate
Port parameter bit	1	03	Data bit, stop bit, check bit

Independent ID	3	00 00 00	ID-H,ID-L,ID-type, if don't need, write 0（ID type has addition meaning ,refer to appendix Ⅰ）
Subnet mask	4	00 FF FF FF	Subnet mask,Little Endian
Sum Check	1	61	Sum check, from target IP to sum check (result keep low byte)

2.4. UART Framing mechanism

2.4.1. packing method

TCP232-T2 accepts time packing method.

1) packing time standard: more than 4 byte packing time.
2) Calculation method :

√ TCP232-T2 Port parameter :data bit 8 byte, stop bit 1 byte, start bit 1byte.

√ Length of one data : 8+1+1=10

√ Packing time of four byte:

$$T = \frac{1}{\text{Baud rate}} * 10 * 4$$

√ Port packing length : 400 byte.

√ When baud rate is 115200, default packing time 0.4ms.

2.4.2. Flow calculation

When TCP232-T2 work in Cloud mode ,it receives network data, then it sends to
Port. Because port speed is limited, sometimes data will overflow.
E.g network data will send m byte data every n seconds. To check whether data can
Overflow : (support notwork is in good station ,network data transmission time is negligible) If data won't
overflow ,   m byte data can be finished in n seconds.

M byte data transmission time :

$$T = \frac{1}{\text{Baud rate}} * 10 * m$$

If   n> 2T ,data won't overflow, TCP232-T2 work well.

2.5. Special Function

2.5.1. RS485 Function

S2 reserved 485 pin, this function can be set by software, won't effect RS232 communication.

2.5.2. Transparent transmission cloud

This function is used to get communication between module and MCU, transmit remote data transparently .   http://cloud.usr.cn/en

Cloud user name and password can be set through software.

2.5.3. Link Function

Link pin can be used as indication pin for TCP connection status.

When connected, it output low level; When unconnected, high level.

When S2 is under TCP mode, Link pin will pull down, otherwise, it stays in high level.

When S2 is under DUP mode, Link pin will always pull down. By default ,it is not checked .

2.5.4. Factory Reset

1) Hardware: At first, check "Reload" in webpage or software, then Reload can only used to restore factory settings. After setting then module will reset, pull " Reload " down to 0 V level for 5 seconds then pull CFG(Reload) up to 3.3 V or don't connect it, factory reset is finished.

2) Software: finish it by set-up software.

3) AT Command: After entering into AT Command, then send AT+ clear.

2.5.5. Reset Function

When S2 works as TCP Client, S2 connects to TCP Server actively. When start Reset function, S2 try to connect to TCP Server for 30 times. If failed, S2 will restart automatically. By default, it won't be chose.

2.5.6. ID Function

When S2 works as TCP Client, it send module ID when establishing connection or carry ID when in communication, S2 ID is decimal .1-65535 (ID function and transparent transmission can't work at same time)

Chose"Send ID after connection" ,module send 4 byte ID (2 byte positive coding +2 byte negative coding)

2.5.7. Index Function

When S2 works as TCP Server, it can establish 16 links simultaneously at most. The max number can be set from 1 to 16. Default is 4. Take 16 link as example, Server send data to 16 Client simultaneously or Server can't distinguish the data source, Index can realize the choice of data source of sending or receiving.

Index function can be set by software or web-page.

2.5.8. Similar RFC2217 Function

This function is used to change USR-TCP232-S2 serial port parameter through network.

E.g: change baud rate from 115200bps to 9600bps. It can be set by software or webpage.

By default ,it is in open state.

When using VCOM software, this function also works, the baud rate of software in PC will match Autonomic with the baud rate of USR-TCP232-S2, don't need to focus on serial port baud rate.

After restart, it is default parameter .

2.6. New Function

2.6.1. AT Command

According to AT Command protocol , TCP232-S2 enter into AT Command mode and set parameter
More details ,please refer to 《USR-TCP232-S2 software manual》
When pull low CFG(Reload)Pin firstly, S2 will set parameter by port , AT Command is invalid.
When enter into AT Command Mode, then pull down CFG(Reload), S2 will set parameter by port firstly,
AT Command is invalid. User pull up or don't connect CFG(Reload), it enter into AT Command Mode again.

2.6.2. Display IP and Data

On the web page of TCP232-S2, it can display the IP of device and sent/received data byte, and the total
data byte of TCP232-S2.
In TCP Mode, it can display the TOP 5 device IP and sent/received data byte. In UDP Mode, it only
display sent/received data byte,don't display connection IP.

2.6.3. Set Client Number in TCP Server Mode

In TCP Server Mode, The maximum number can be configured by user. TCP Client number is from 1
to 16, default value 4. When the Client link is more than 16, the new link will replace the former link from Link 1
When the Client link is more than 4, send and receive data at same time , the data flow should be within
2.5 KB/s.

2.6.4. Defined MAC Address

Mac address can be modified.   Factory Mac address is exclusive.

2.6.5. Defined DNS Server IP

To resolve server domain name, user should send data by gateway or router, then gateway or router
Distributes IP address, it can show IP in the webpage. User can set specific domain name resolution IP,
to specific gateway or router to resolve domain name.

2.6.6. Defined Registration Package

The content of registration package can be defined, 40 bytes at most.
It includes sending registration package when connecting and carrying registration package when sending
data. It can be used singly or together.
It is set by webpage, support decimal input and hexadecimal input. By default ,this function isn't open..

2.6.7. Defined Heartbeat Package

The content of heartbeat package can be defined, 40 bytes at most. Time set from 1s to 255s
It can ensure connection is reliable, put an end to connect feign death; It can send to LAN Port or Serial
Port singly or at same time.

2.6.8. HTTPD Client

TCP232-S2 has HTTPD Client , it support GET and POST. In HTTPD Client, package head and end can be modified . In GET Mode, package head data is replaced by "$'. In POST Mode, put port data in the end, "$" means data length in package head, TCP232-S2 will assign a value, user don't need to modify. The Parameter can be set in webpage.

2.7. Firmware Upgrade
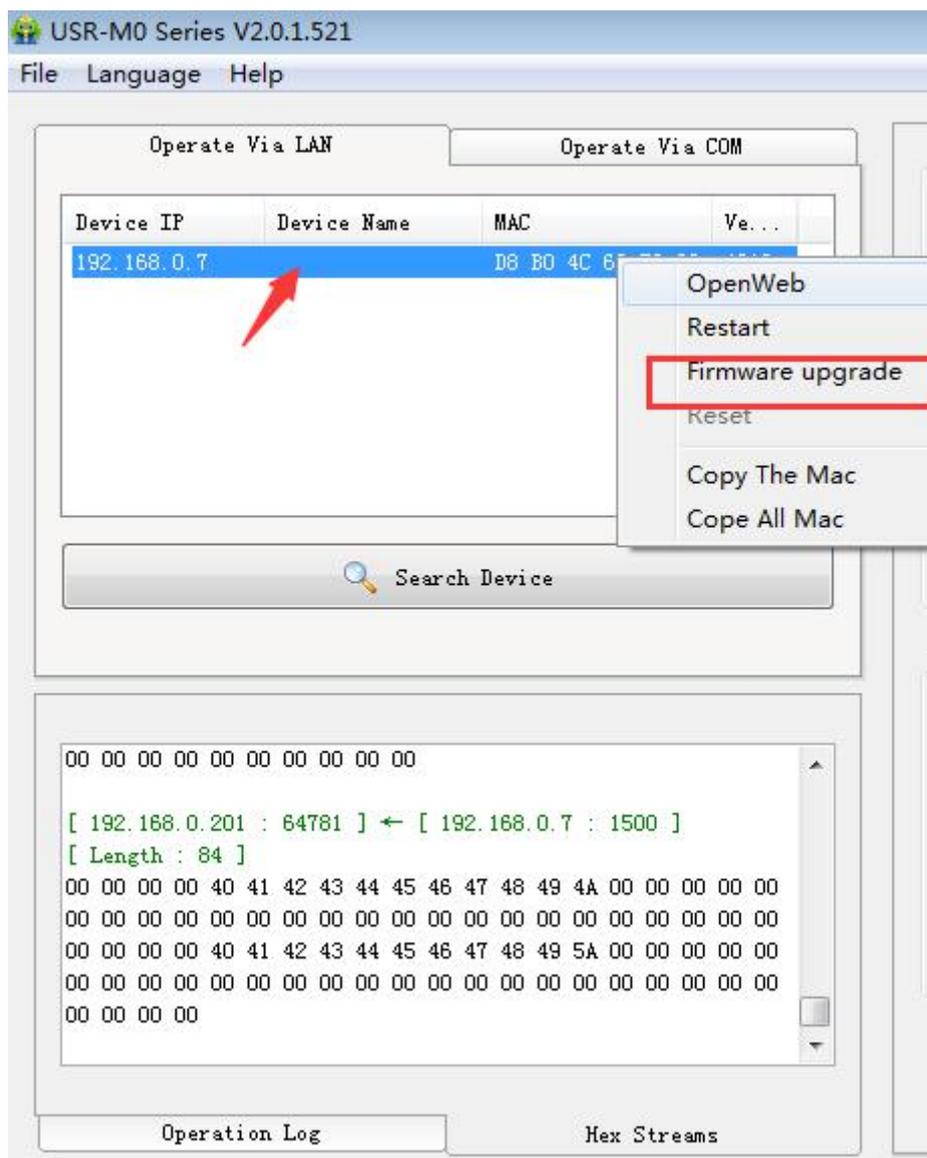
**Diagram 17**   Firmware upgrade

3. Webpage

By default, user name and password : admin .   IP :192.168.0.201

At first, user should connect PC to TCP232-S2 Lan port ,then set parameter by webpage or software. TCP232-S2 connects Auto-MDI/MDIX RJ45 port with 10/100Mbps. It support Router connection and direct connection.

1)TCP232-S2 connect PC by Lan cable, user should set parameter for PC. When TCP232-S2 connects to power supply and PC, module will connect PC directly. (Module IP and PC IP should in the same Network segment)

2)TCP232-S2 connects PC by router, one PC can connect several modules or one module connects Several PC (When modules works as TCP Server , at most it can connect 16 clients)

3.1. Log in

PC connect S2 by RJ45 port, when light blinked all the time, open software ,check module IP
Open a browser, type 192.168.07 , Name and password：admin
User can also log in by software.
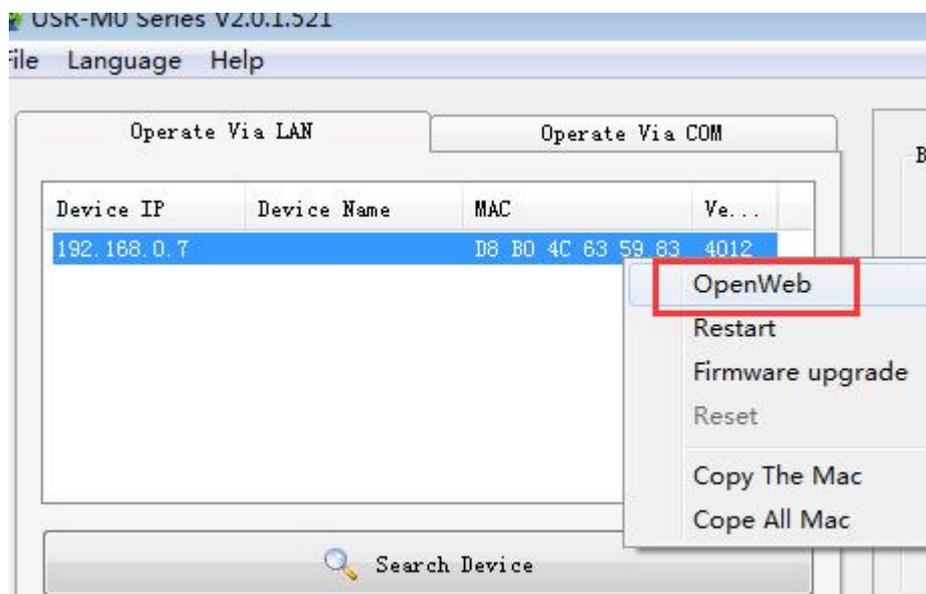
3.2. State Configuration

Module name , Current IP address , Mac address, Remote IP/TX/RX , TX Count/ RX Count.
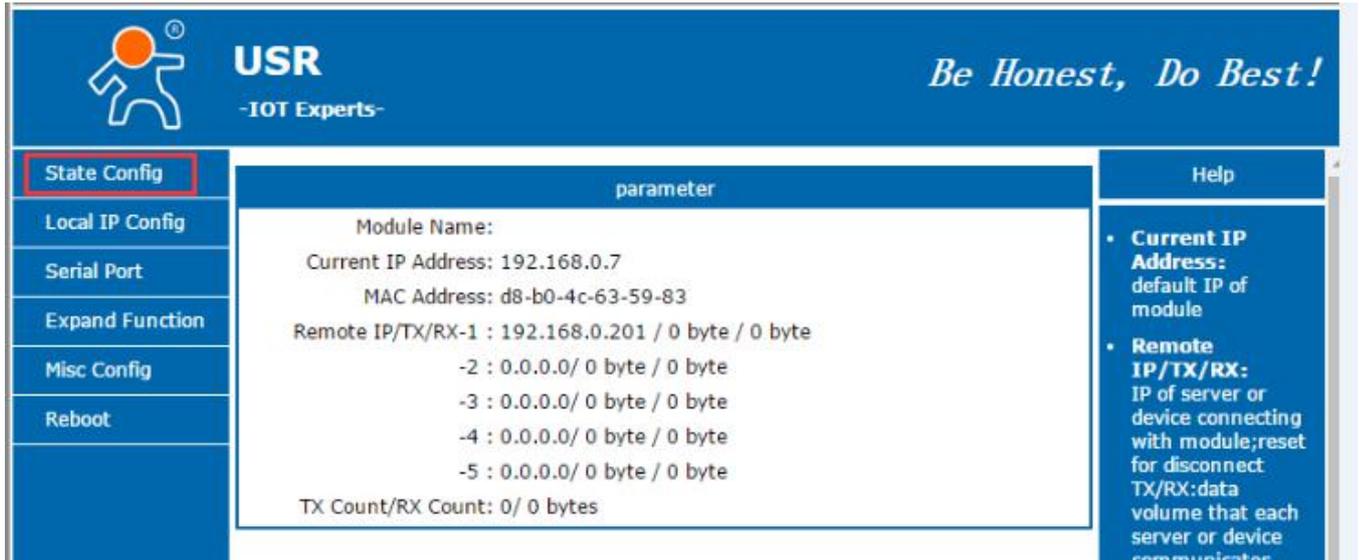
Diagram 18   State Configuration

3.3. Local IP

1)IP type : Static IP means fixed IP,   DHCP means acquire IP automatically.

2)Static IP : when user choose Static IP, don't conflict IP address in the LAN.

3)Subnet mask: used to distinguish network segment, default value is 255.255.255.0

4)Gateway: Router IP address, it should be set correctly when used for domain name resolution.

5)DNS Gateway : Server IP of domain name resolution , by default it is module gateway.
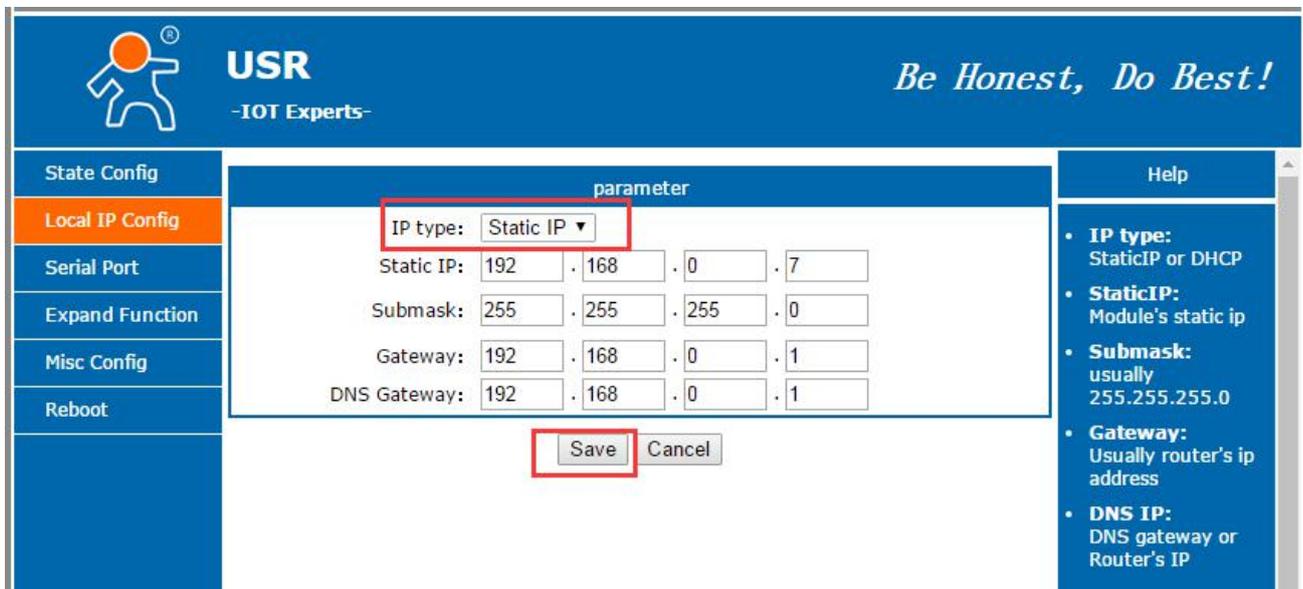

Diagram 19   Local IP
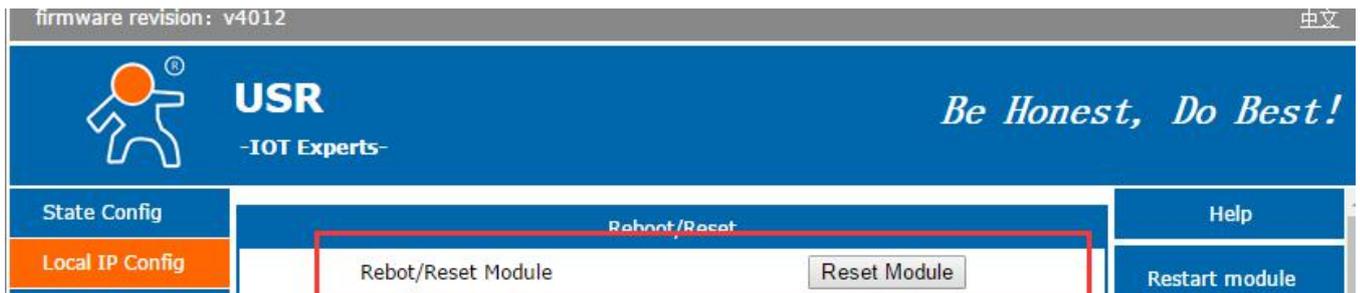
Diagram 20   Local IP

3.4. Serial Port Parameter

1) Baud rate: 600bps～460.8Kbps
2) Stop bit: 1,2.
3) Data bit : 5,6,7,8.
4) Check bit: NONE,ODD,EVEN,MARK,SPACE.
5) Local Port :By default ,it is local port number, it can set 0 if connect outer net.
6) Remote: Remote server port
7) Work mode :TCP Server, TCP Client,UDP Client,UDP Server,HTTPD Client.
8) Remote Server address: it can be IP address or server domain name.
9) HTTPD:HTTPD GET or HTTPD POST.
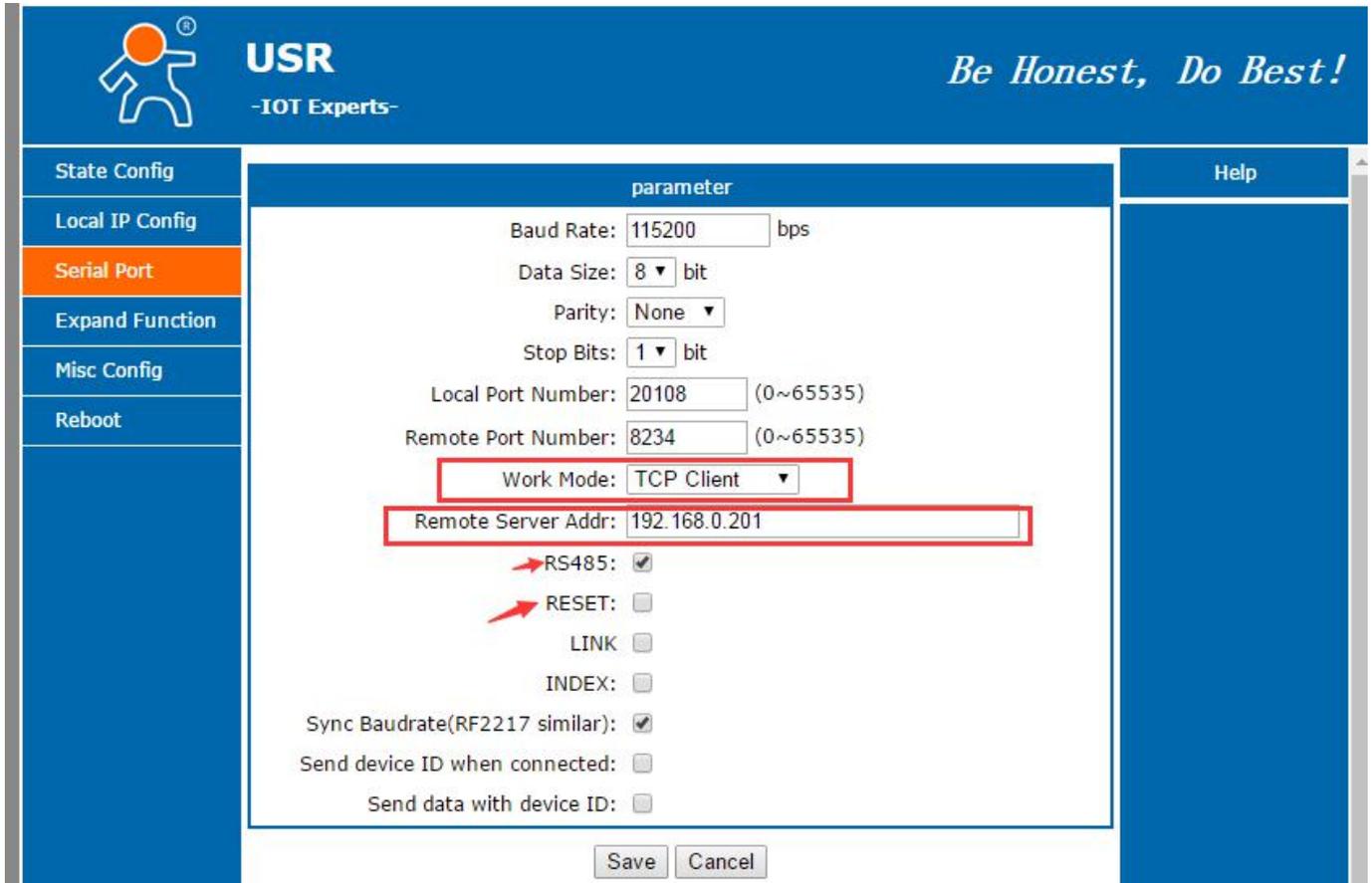
**Diagram 21   Serial Port parameter**

3.5. Expand Function
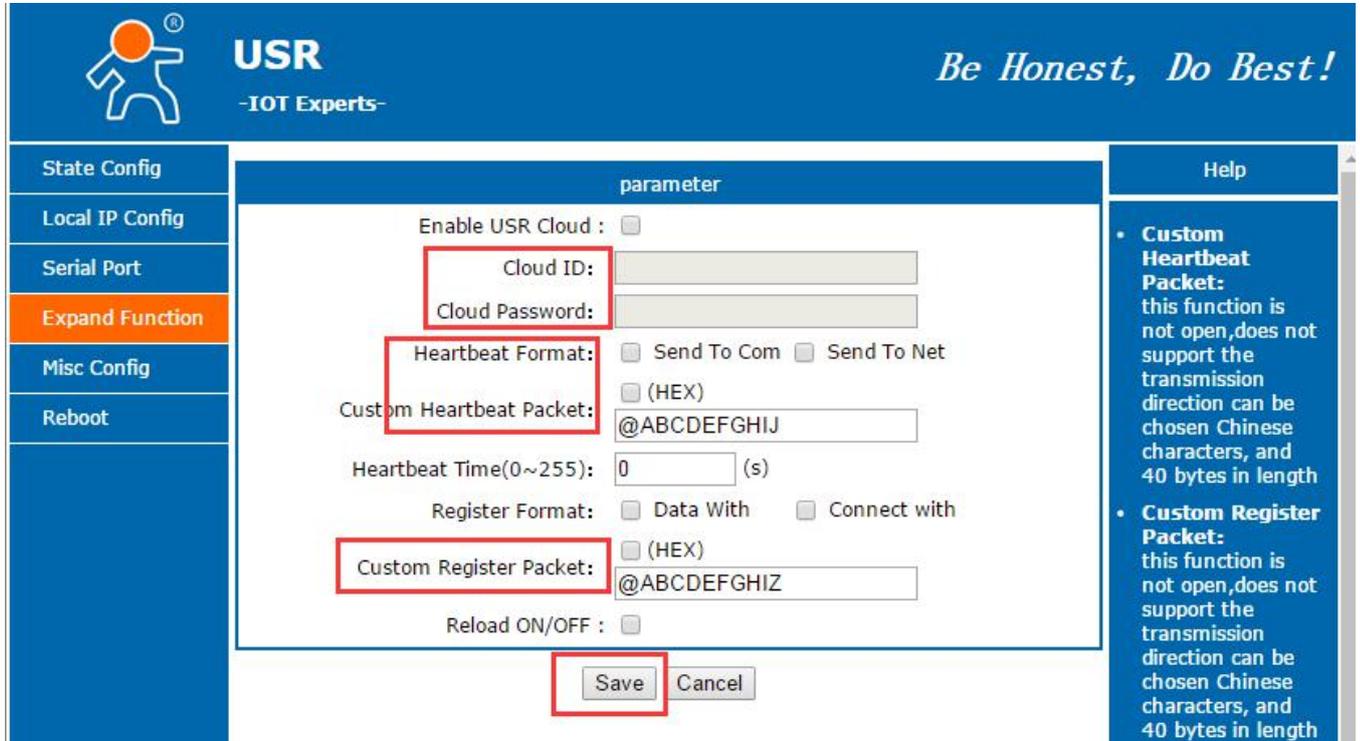
Diagram 22   Expand Function

3.6. Misc Configuration

After setting , click "save" ,then restart the module .
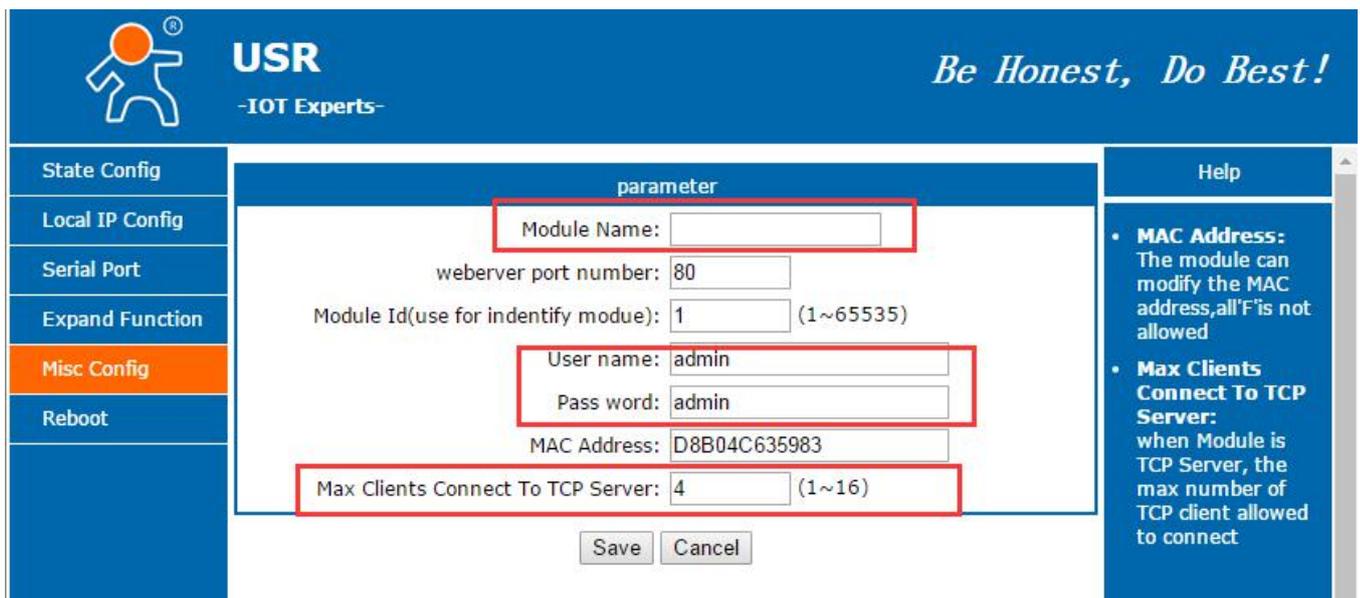
Diagram 23   Misc Setting

3.7. Reboot

Only have reset function.
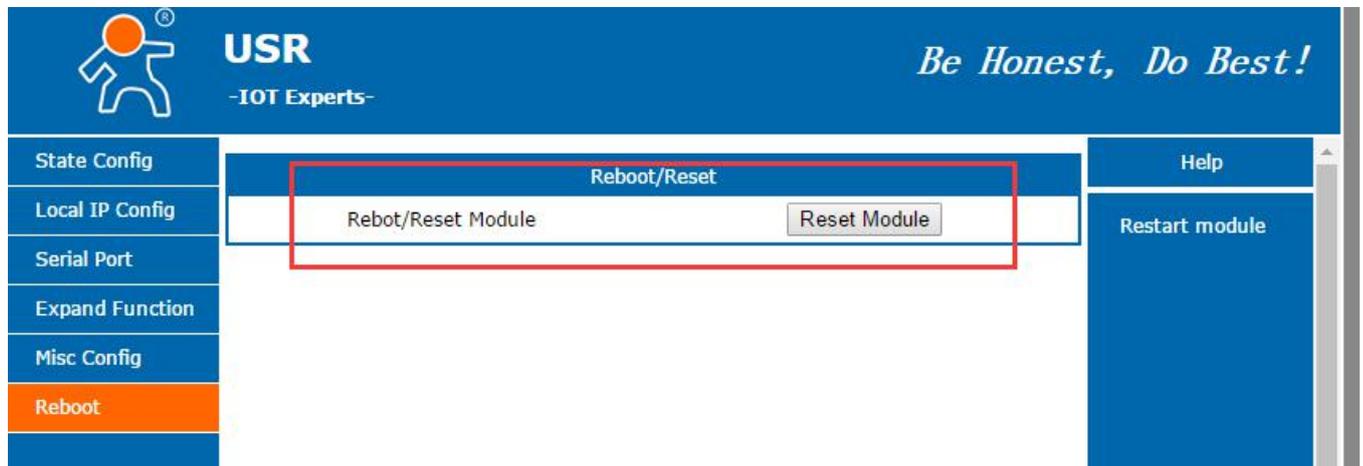
Diagram 24   Reboot Setting

4. AT Command

After entering into AT Command , Cloud function is forbidden until quits the mode.

When pull down CFG(Reload), T2 will do read setting through port to parameter, AT command won't work. If user has pull down CFG(Reload) to AT mode, serial configuration is preferred , AT command won't work, when pull up CFG(Reload) or don't connect it, it enter into AT mode again.

4.1. AT Command Model

1) Send +++ to T2 module from serial port, T2 send "a "

2) Send "a" in 3s after receiving "a".

3) T2 return to + OK , enter into AT Demand Model.

4)Swift Command Mode into Cloud Mode by AT+ENTM . In command mode, input "AT+ENTM ' Enter key ,then into Cloud Mode .

4.2. AT Command

NO	command	instruction	Restart effect
Basic command			
1	E	Module AT command echo setting	N
2	Z	Reset module	N
3	ENTM	Quit AT Command, into transparent mode	N
4	MAC	Query MAC address	N

5	VER	Query module firmware version	N
6	CLEAR	Restore factory setting	N
Network parameter			
7	WANN	Set/Query module network parameter	Y
8	DHCPEN	DHCP enable function setting	Y
Port parameter			
9	UART	Set/Query module port parameter	Y
10	SOCKA1	Set/Query network protocol parameter	Y

4.3. AT Command Set

< > Necessary part

[   ] Optional part

Command : AT+<CMD><CR> or AT+<CMD>=<data><CR>

AT+ :prefix

<CMD>: content

"=" : parameter setting

<data>: parameter content

<CR>: end mark, "enter" press   ASCII or 0X0D or 0X0A;

Response reply:

<CR><LF>+OK<CR><LF>

<CR>:ASCII 0x0d;

<LF>:ASCII 0x0a;

4.3.1. AT+E

Function: Query / Set   AT Command echo setting

Format: Query : A+E<CR>

      <CR><LF>+OK=<on/off><CR><LF>

   Set: A+E=<on/off><CR>

     <CR><LF>+OK=<on/off><CR><LF>

Parameter: 1.on : turn on echo function, echo command in AT command.

     2.off : turn off echo function.

4.3.2. AT +ENTM

Function: enter into transparent transmission

Format:

   Set

   AT+ENTM<CR>

   <CR><LF>+OK<CR><LF>

Parameter : Null   (when execute this commend, module switch into Cloud mode)

4.3.3. AT+Z

Function: restart module

Format :

    Set

    AT+Z<CR>

    <CR><LF>+OK<CR><LF>

Parameter : Null   (when execute this commend, module restarts ,quit AT Command)

4.3.4. AT+VER

Function: Set/Query module firmware version

Format: Query : AT+VER<CR>

                <CR><LF>+OK=<ver><CR><LF>

Parameter: ver: query module

4.3.5. AT+MAC

Function: query module MAC

Format:

    Query

    AT+MAC<CR>

    <CR><LF>+OK=<MAC><CR><LF>

Parameter : mac module MAC Address

4.3.6. AT+CLEAR

Function: factory default

Format:

    Set

    AT+CLEAR<CR>

    <CR><LF>+OK<CR><LF>

4.3.7. AT+UART

Function: set/query UART parameter.

Format:

Query：AT+UART<CR>

<CR><LF>+OK=<baudrate,data_bits,stop_bit,parity,flowctrl，uartnum><CR><LF>

Set：AT+UART=<baudrate,data_bits,stop_bit,parity,flowctrl，uartnum ><CR><LF>

  <CR><LF>+OK<CR><LF>

Parameter: baud rate: 9600,19200,38400,57600,115200,230400,380400,460800(module can set baud rate randomly)

Data bits :5,6,7,8

Stop bits: 1,1.5,2（refer to module）

Parity: check bits

None(no check bits)

Even(even check)

ODD(odd check)

Mask (1 check)

Space (0 check)

Flow control : hardware flow control (CTS RTS) ,T2 has no hardware flow control, can only set NFC

NFC: no hardware flow control .

FC: has hardware flow control .

485: support 485 , when it turns on , RS_EN pin is the same as RTS pin.

UART number : the port number need to set (T2 only has 1 port, so write 0 here)

4.3.8. AT SOCKA1

Function: set/query UART parameter.

Format:

Query：AT+UART<CR>

     <CR><LF>+OK=<baudrate,data_bits,stop_bit,parity,flowctrl，uartnum><CR><LF>

Set： AT+UART=<baudrate,data_bits,stop_bit,parity,flowctrl，uartnum ><CR><LF>

     <CR><LF>+OK<CR><LF>

Parameter: protocol: protocol type ,includes:

               TCPS --TCP Server

               TCPC-- TCP Client

               UDPS-- UDP Server

               UDPC-- UDP Client

IP: when the module is set "client" , server IP address.

Port :protocol port ,decimal number ,less than 65535.

4.3.9. AT+DHCPEN

Function: turn on /off DHCP

Format:

     Query: AT+DHCPEN<CR>

         <CR><LF>+OK=<Type><CR><LF>

     Set:AT+DHCPGW=<Type><CR>

         <CR><LF>+OK<CR><LF>

Parameter : Type: on means turn on DHCP , off means turn off DHCP.

4.3.10. AT+WANN

Function: set/query WAN IP

Format:

  Query:

  AT+WANN<CR>

  <CR><LF>+OK=<mode,address,mask,gateway><CR><LF>

Set:

AT+WANN=<mode,address,mask,gateway><CR>

<CR><LF>+OK<CR><LF>

Parameter:

mode:Network IP mode(T2 can only set static mode)

Static: static IP

DHCP: dynamic IP (address,mask,gateway can be omitted)

Address : IP address

Mask: subnet mask

Gate way: gateway address

5. Appendix

AppendixⅠ:Meaning of Port parameter bit

Bit number	Instruction	Value	Description
1:0	Data bit choice	00	5 bit data bit
		01	6 bit data bit
		10	7 bit data bit
		11	8 bit data bit
2	Stop bit	00	1 bit stop bit
		01	2 bit stop bit
3	Check bit enable	00	Unable check bit
		01	Enable check bit
5:4	Check bit type	00	ODD check
		01	EVEN check
		10	Set 1
		11	Zero Clearing
8:6	Undefined	000	Write 0

AppendixⅡ:ID type of Independent ID

This byte is the last byte of 3 bytes, additional meaning is as follows:

bit0(1) send ID when connecting;

bit1(2)send ID when send data;

bit2(4) RS485;

bit3(8) NC;

bit4(16) Reset;

bit5(32) Link-state;

bit6(64) tcp server index;

bit7(128) baud rate synchronization, similar RFC2217.

When each bit is 1, it means related function is invalid. Otherwise, it is valid, default is

RS485 and similar RFC2217 set 1 .(turn on function, value is 0x84)

Appendix Ⅲ:Upper computer Socket programming example

Server Socket code:

```c
1.  #include <stdio.h>
2.  #include <string.h>
3.  #include <sys/socket.h>
4.  #include <netinet/in.h>
5.  #include <stdlib.h>
6.  #include <syslog.h>
7.  #include <errno.h>
8.  #define MAX_LISTEN_NUM 5
9.  #define SEND_BUF_SIZE 100
10. #define RECV_BUF_SIZE 100
11. #define LISTEN_PORT 1010
12. int main()
13. {
14.   int listen_sock = 0;
15.   int app_sock = 0;
16.   struct sockaddr_in hostaddr;
17.   struct sockaddr_in clientaddr;
18.   int socklen = sizeof(clientaddr);
19.   char sendbuf[SEND_BUF_SIZE] = {0};
20.   char recvbuf[RECV_BUF_SIZE] = {0};
21.   int sendlen = 0;
22.   int recvlen = 0;
23.   int retlen = 0;
24.   int leftlen = 0;
25.   char *ptr = NULL;
26.   memset((void *)&hostaddr, 0, sizeof(hostaddr));
27.   memset((void *)&clientaddr, 0, sizeof(clientaddr));
28.   hostaddr.sin_family = AF_INET;
29.   hostaddr.sin_port = htons(LISTEN_PORT);
30.   hostaddr.sin_addr.s_addr = htonl(INADDR_ANY);
31.   listen_sock = socket(AF_INET, SOCK_STREAM, 0);
32.   if(listen_sock < 0)
33.   {
34.       syslog(LOG_ERR, "%s:%d, create socket failed", __FILE__, __LINE__);
35.       exit(1);
36.   }
37.   if(bind(listen_sock, (struct sockaddr *)&hostaddr, sizeof(hostaddr)) < 0)
38.   {
```

```
39.        syslog(LOG_ERR, "%s:%d, bind socket failed", __FILE__, __LINE__);
40.        exit(1);
41.    }
42.    if(listen(listen_sock, MAX_LISTEN_NUM) < 0)
43.    {
44.        syslog(LOG_ERR, "%s:%d, listen failed", __FILE__, __LINE__);
45.        exit(1);
46.    }
47.    while(1)
48.    {
49.        app_sock = accept(listen_sock, (struct sockaddr *)&clientaddr, &socklen);
50.        if(app_sock < 0)
51.        {
52.            syslog(LOG_ERR, "%s:%d, accept failed", __FILE__, __LINE__);
53.            exit(1);
54.        }
55.        sprintf(sendbuf, "welcome %s:%d here!/n", inet_ntoa(clientaddr.sin_addr.s_addr), clientaddr.sin_port);
56.        //send data
57.        sendlen = strlen(sendbuf) +1;
58.        retlen = 0;
59.        leftlen = sendlen;
60.        ptr = sendbuf;
61.        //while(leftlen)
62.        {
63.            retlen = send(app_sock, ptr, sendlen, 0);
64.        if(retlen < 0)
65.        {
66.            if(errno == EINTR)
67.                retlen = 0;
68.        else
69.            exit(1);
70.        }
71.        leftlen -= retlen;
72.        ptr += retlen;
73.        }
74.        //receive data
75.        recvlen = 0;
76.        retlen = 0;
77.        ptr = recvbuf;
78.        leftlen = RECV_BUF_SIZE -1;
79.        //do
80.        {
81.            retlen = recv(app_sock, ptr, leftlen, 0) ;
82.        if(retlen < 0)
83.        {
```

```
84.            if(errno == EINTR)
85.                retlen = 0;
86.          else
87.                exit(1);
88.        }
89.       recvlen += retlen;
90.       leftlen -= retlen;
91.       ptr += retlen;
92.      }
93.      //while(recvlen && leftlen);
94.      printf("receive data is : %s", recvbuf);
95.     close(app_sock);
96.    }
97.   close(listen_sock);
98.
99.   return 0;
100.    }
```

Client Socket code:

```
1.   #include <stdio.h>
2.   #include <string.h>
3.   #include <sys/socket.h>
4.   #include <netinet/in.h>
5.   #include <syslog.h>
6.   #include <errno.h>
7.   #include <stdlib.h>
8.   #define MAX_LISTEN_NUM 5
9.   #define SEND_BUF_SIZE 100
10. #define RECV_BUF_SIZE 100
11. #define SERVER_PORT 1010
12. int main()
13. {   int sock_fd = 0;
14.      char recvbuf[RECV_BUF_SIZE] = {0};
15.      char sendbuf[SEND_BUF_SIZE] = {0};
16.      int recvlen = 0;
17.      int retlen = 0;
18.      int sendlen = 0;
19.      int leftlen = 0;
20.      char *ptr = NULL;
21.      struct sockaddr_in ser_addr;
22.      memset(&ser_addr, 0, sizeof(ser_addr));
23.      ser_addr.sin_family = AF_INET;
24.      inet_aton("127.0.0.1", (struct in_addr *)&ser_addr.sin_addr);
25.      ser_addr.sin_port = htons(SERVER_PORT);
```

```
26.      sock_fd = socket(AF_INET, SOCK_STREAM, 0);
27.      if(sock_fd < 0)
28.      {
29.          syslog(LOG_ERR, "%s:%d, create socket failed", __FILE__, __LINE__);
30.          exit(1);
31.      }
32.      if(connect(sock_fd, (struct sockaddr *)&ser_addr, sizeof(ser_addr)) < 0)
33.      {
34.          syslog(LOG_ERR, "%s:%d, connect socket failed", __FILE__, __LINE__);
35.          exit(1);
36.      }
37.       //receive data
38.       recvlen = 0;
39.       retlen = 0;
40.       ptr = recvbuf;
41.       leftlen = RECV_BUF_SIZE -1;
42.       //do
43.       {
44.           retlen = recv(sock_fd, ptr, leftlen, 0) ;
45.        if(retlen < 0)
46.        {
47.            if(errno == EINTR)
48.              retlen = 0;
49.          else
50.              exit(1);
51.        }
52.        recvlen += retlen;
53.        leftlen -= retlen;
54.       ptr += retlen;
55.       }
56.      //while(recvlen && leftlen);
57.      printf("receive data is : %s", recvbuf);
58.      sprintf(sendbuf, "hello server/n");
59.      //send data
60.      sendlen = strlen(sendbuf) +1;
61.      retlen = 0;
62.      leftlen = sendlen;
63.      ptr = sendbuf;
64.     // while(leftlen)
65.      {
66.          retlen = send(sock_fd, ptr, sendlen, 0);
67.       if(retlen < 0)
68.       {
69.           if(errno == EINTR)
70.              retlen = 0;
```

```
71.        else
72.            exit(1);
73.      }
74.      leftlen -= retlen;
75.      ptr += retlen;
76.    }
77.    close(sock_fd);
78. }
```

6. Contact information

Company: Jinan USR IOT Technology Limited.
Address: Floor 11,Building1,No.1166 Xinluo Street, Gaoxin Distric, Jinan, Shandong, 250101 China.
Tel: 86-531-55507297   86-531-88826739-803
Web: http://www.usriot.com/
Support:http://h.usriot.com/
Email: sales@usr.cn

7. Disclaimer

The document provides information about USR-TCP232-S2 module, it doesn't grant any license to the intellectual property rights. Except the responsibility declared in the product sale clause, USR does not assure any other responsibilities. In addition, USR does not make any warranties for the sale and use of this product, including the suitability of products for a particular purpose, merchant ability or fitness for any patent, copyright or other intellectual property infringement.USR may make changes to specifications and products description without notice.

8. Undated History

V 1.0 2016-07-09   First Version