SIM8262E-M2 5G HAT

Features

- Based on Qualcomm platform, support 5G NSA and SA networking, support multi-mode and multi-band
- Integrated multi-constellation system dual-frequency positioning: GPS, GLONASS, BeiDou, Galileo and QZSS
- Support Windows / Linux / Android and other operating systems
- USB 3.1 port (USB 2.0 compatible) for connecting to PC, Raspberry Pi, or Jetson Nano host board to enable high speed 5G communication
- Onboard M.2 B KEY slot, compatible with 5G modules such as SIM8202X-M2 / SIM8200EA-M2 / SIM8262X-M2 series
- Onboard UART, PWR, and RST control pin, built-in voltage level translator, enabled via DIP switch, for use with hosts like Raspberry Pi or Arduino
- Onboard USB-C connector, enabled via switch, for connecting standalone power supply for the module, allows more loads, stable and flexible power supply
- Onboard power switch, reset button, and LED indicator, easy to switch the control module on and off and check the running status
- Onboard 4-way SMA to IPEX antenna adapting interface, factory default welded SMA terminals, antenna installation is more convenient
- Onboard two SIM card slots, dual-card single-standby, can be switched and enabled by AT command
- Onboard audio interface and audio decoding chip, which can be used for voice manipulation such as making calls
- High-efficiency power supply circuit, up to 3A output current

5G Module Parameters

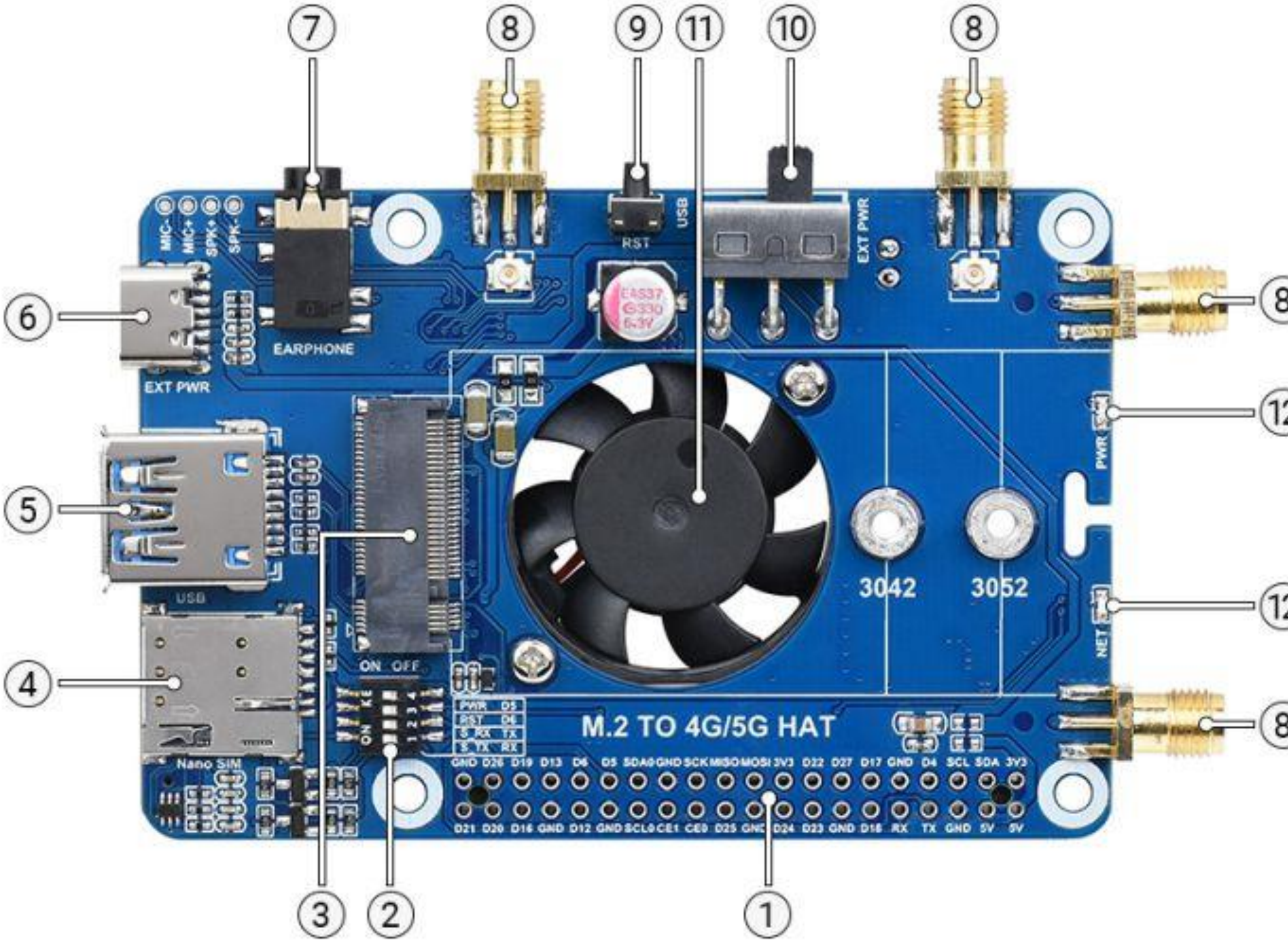
	SIM8200EA-M2	SIM8202G-M2	SIM8262E-M2	SIM8262A-M2
5G Standard	3GPP R15		3GPP R16	

Chip solution	Qualcomm Snapdragon X55		Qualcomm Snapdragon X62	
Working Band				
Sub-6G	n1, n2, n3, n5, n7, n8, n12, n20, n28, n38, n40, n41, n48, n66, n71, n78	n1, n2, n3, n5, n7, n8, n12, n20, n28, n38, n40, n41, n66, n71, n77, n78, n79	n1, n3, n5, n7, n8, n20, n28, n38, n40, n41, n77, n78,n79	n2, n5, n7, n12, n13, n14, n25, n30, n41, n48, n66, n71, n77, n78, n79
LTE-FDD	B1, B2, B3, B4, B5, B7, B8, B12, B13, B14, B17, B18, B19, B20, B25, B26, B28, B29, B30, B32, B66, B71		B1, B3, B5, B7, B8, B18, B19, B20, B26, B28, B32	B2, B4, B5, B7, B12, B13, B14, B25, B26, B29, B30, B66, B71
LTE-TDD	B34, B38, B39, B40, B41, B42, B43, B48		B38, B39, B40, B41, B42, B43	B41, B46, B48
WCDMA	B1, B2, B3, B4, B5, B8		B1, B5, B8	B2, B4, B5
GNSS	GPS, GLONASS, BeiDou, Galileo and QZSS			
Data Transfer				
Sub-6G	2.4 Gbps (DL) / 500 Mbps (UL)			
LTE	1 Gbps (DL) / 200 Mbps (UL)			
HSPA+	42 Mbps (DL) / 5.76 Mbps (UL)			

<table>
<tr><th colspan="3">Software Function</th></tr>
<tr><td>Operating System</td><td colspan="2">Windows/Linux/Android</td></tr>
<tr><td>Communication Protocol</td><td colspan="2">TCP/IP, IPV4, IPV6, Multi-PDP, FTP, FTPS, HTTP, HTTPS, MQTTS, DNS, SSL3.0</td></tr>
<tr><td>Dial-up</td><td colspan="2">RNDIS, NDIS, PPP, MBIM</td></tr>
<tr><td>Text Message (SMS)</td><td colspan="2">Support MT, MO, CB, Text, PDU</td></tr>
<tr><td>Firmware Upgrade</td><td colspan="2">Supports firmware upgrade via USB interface</td></tr>
<tr><th colspan="3">Hardware Description</th></tr>
<tr><td>SIM Card</td><td colspan="2">1.8V/2.95V</td></tr>
<tr><td rowspan="2">Antenna Interface</td><td colspan="2">for 3G/4G/5G/GNSS</td></tr>
<tr><td>6 × IPEX-4 ports</td><td>4 × IPEX-4 ports</td></tr>
<tr><td>Power Supply</td><td colspan="2">3.135~4.4V</td></tr>
<tr><td>Outline Package</td><td colspan="2">M.2</td></tr>
</table>

Dimensions	52.0 × 30.0 × 2.3mm	42.0 × 30.0 × 2.3mm	
Operating Temperature	-30℃ ~ +70℃		
Application Scenarios			
Applicable Area	China, US, Japan, Korea, Europe, Middle East, Americas	All regions except Americas	Americas
	The applicable area is for reference only, and the appropriate module should be selected according to the frequency band covered and supported by the local operator's network.		
Typical Applications	CPE, smart gateway, drone, live video, telemedicine, smart security		

What's On Board

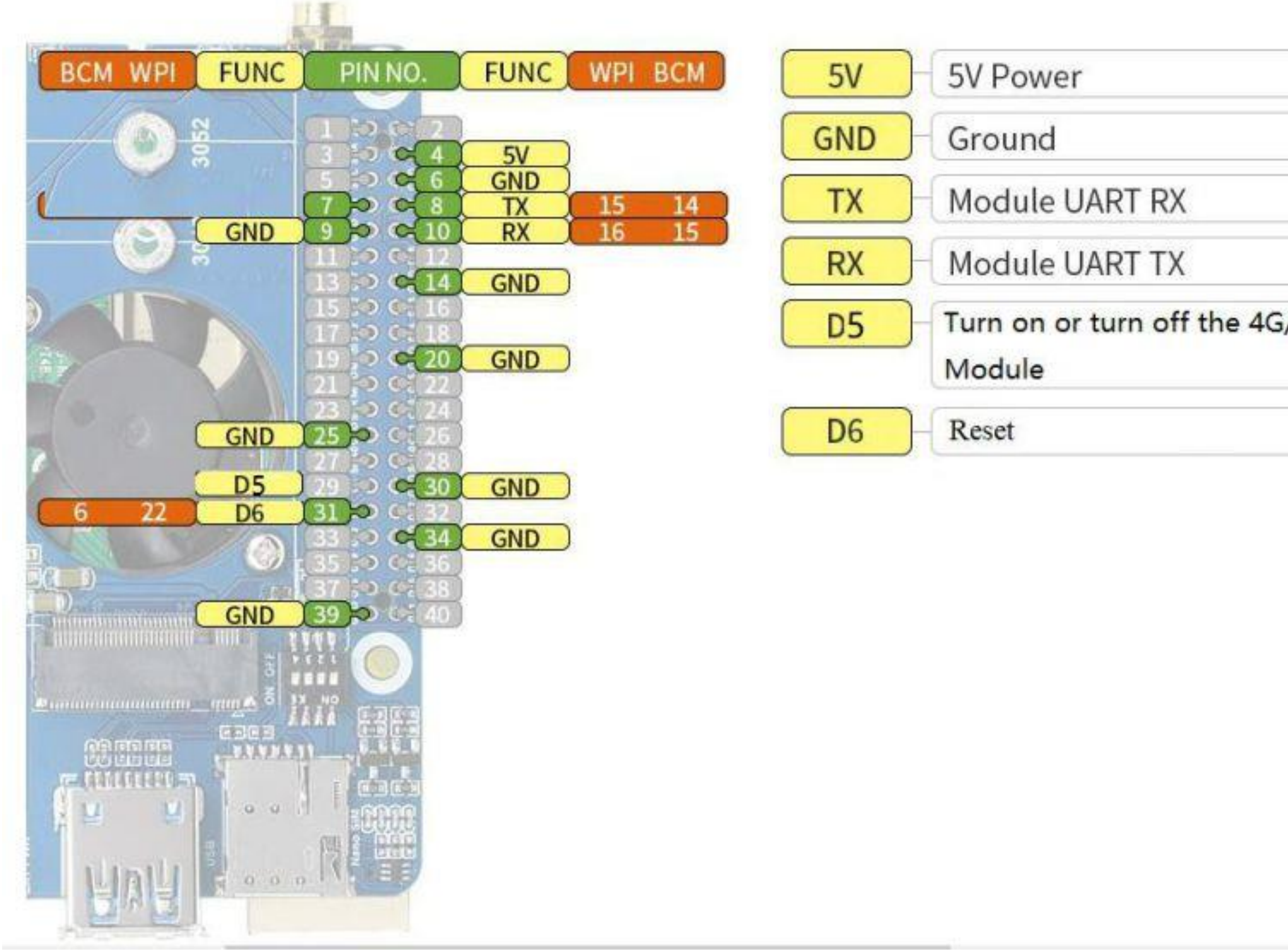


Number	Name	Description
①	Raspberry Pi GPIO header	For connecting to Raspberry Pi
②	Switch	Enable the corresponding pin
③	M.2 connector	Compatible with RM500U-CN / RM500Q-CN / RM500Q-GL /RM50XQ-AE and other series of 5G modules
④	SIM card holder	Onboard two SIM card slots, dual card single standby. The default SIM1 card slot works, SIM2 is on the back, requires module support, and must be switched through AT commands
⑤	USB3.1 interface	Backward compatible with USB 2.0, can be used to connect to PC/Raspberry Pi/Jetson Nano, etc.
⑥	USB Type-C connector	5V 3A input; stable and flexible power supply

⑦	Audio port	SIM82XX series support audio function, RM50XX series do not support this audio function
⑧	Antenna interface	Onboard four-way antenna, strong signal
⑨	Reset switch	One-key reset
⑩	Power Switch	To facilitate the power supply mode of the control module: ——If set to USB, the module will provide power through the "⑤.USB3.1 interface"; ——If set to EXT PWR, the module will provide power through the "⑥.USB Type-C interface" external power supply
⑪	Cooling fan	Cool down the Raspberry Pi and 5G module at the same time
⑫	Indicator light	Check the module running status anytime, anywhere

1. Pinout Definition

After connecting to Raspberry Pi with a 2*20 female header, these pins (TX, RX, D4, and D6) can be connected or not through the DIP switch:



5V	5V Power
GND	Ground
TX	Module UART RX
RX	Module UART TX
D5	Turn on or turn off the 4G/ Module
D6	Reset

4G/5G modules function testing

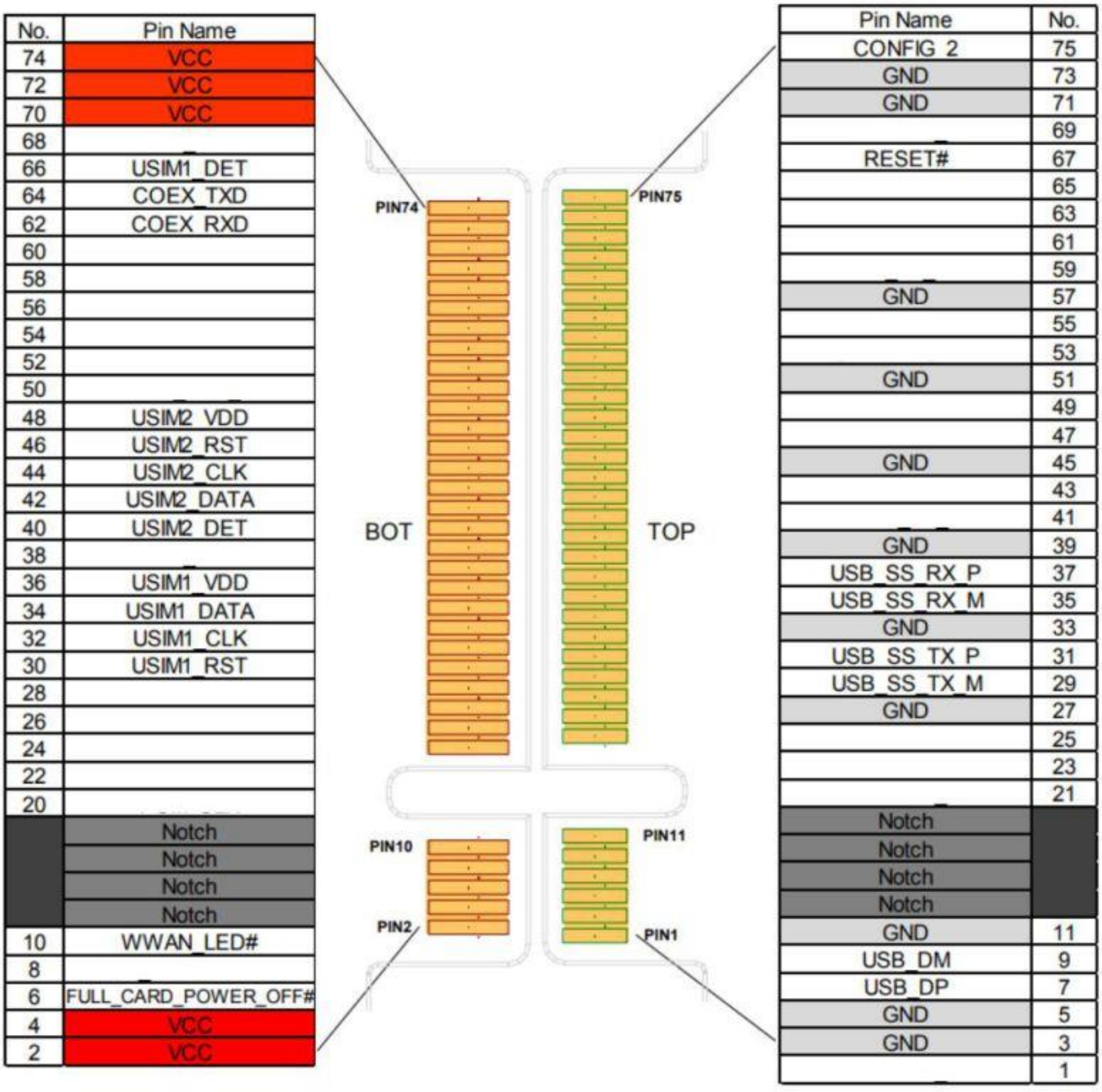
Category	4G/5G Module	Network Communication	GNSS Positioning	Voice calls through Earphone Port	Dual SIMs	UART Interface	External Power Supply?
5G	SIM8202G-M2	5G/4G/3G	Support	Support	Support	Support	Optional, but

							recommended
5G	SIM8262E-M2	5G/4G/3G	Support	Support	Support	Support	Optional, but recommended
5G	SIM8200EA-M2	5G/4G/3G	Support	Support	Support	Support	Optional, but recommended
5G	RM500U-CN	5G/4G/3G	NOT Support	NOT Support	Support	Support	Recommended
5G	RM500Q-GL	5G/4G/3G	Support	NOT Support	Support	NOT Support	Recommended
5G	RM500Q-AE	5G/4G/3G	Support	NOT Support	NOT Support	NOT Support	Recommended
5G	RM502Q-AE	5G/4G/3G	Support	NOT Support	NOT Support	NOT Support	Recommended
LTE-A	EM06-E	LTE-A/4G/3G	NOT Support	NOT Support	NOT Support	NOT Support	Optional

LTE-A	A7906E	LTE-A/4G/3G	NOT Support	NOT Support	NOT Support	NOT Support	Optional
4G	SIM7600G-H-M2	4G/3G/2G	Support	Support	NOT Support	Support	Optional

4G/5G Module Compatibility

If you need to use the M.2 TO 4G/5G HAT for other 4G/5G modules, you can refer to the M.2 connection diagram below, check whether there is any pin conflict, and then connect to test:



No.	Pin Name
74	VCC
72	VCC
70	VCC
68	
66	USIM1_DET
64	COEX_TXD
62	COEX_RXD
60	
58	
56	
54	
52	
50	
48	USIM2_VDD
46	USIM2_RST
44	USIM2_CLK
42	USIM2_DATA
40	USIM2_DET
38	
36	USIM1_VDD
34	USIM1_DATA
32	USIM1_CLK
30	USIM1_RST
28	
26	
24	
22	
20	
	Notch
	Notch
	Notch
	Notch
10	WWAN_LED#
8	
6	FULL_CARD_POWER_OFF#
4	VCC
2	VCC

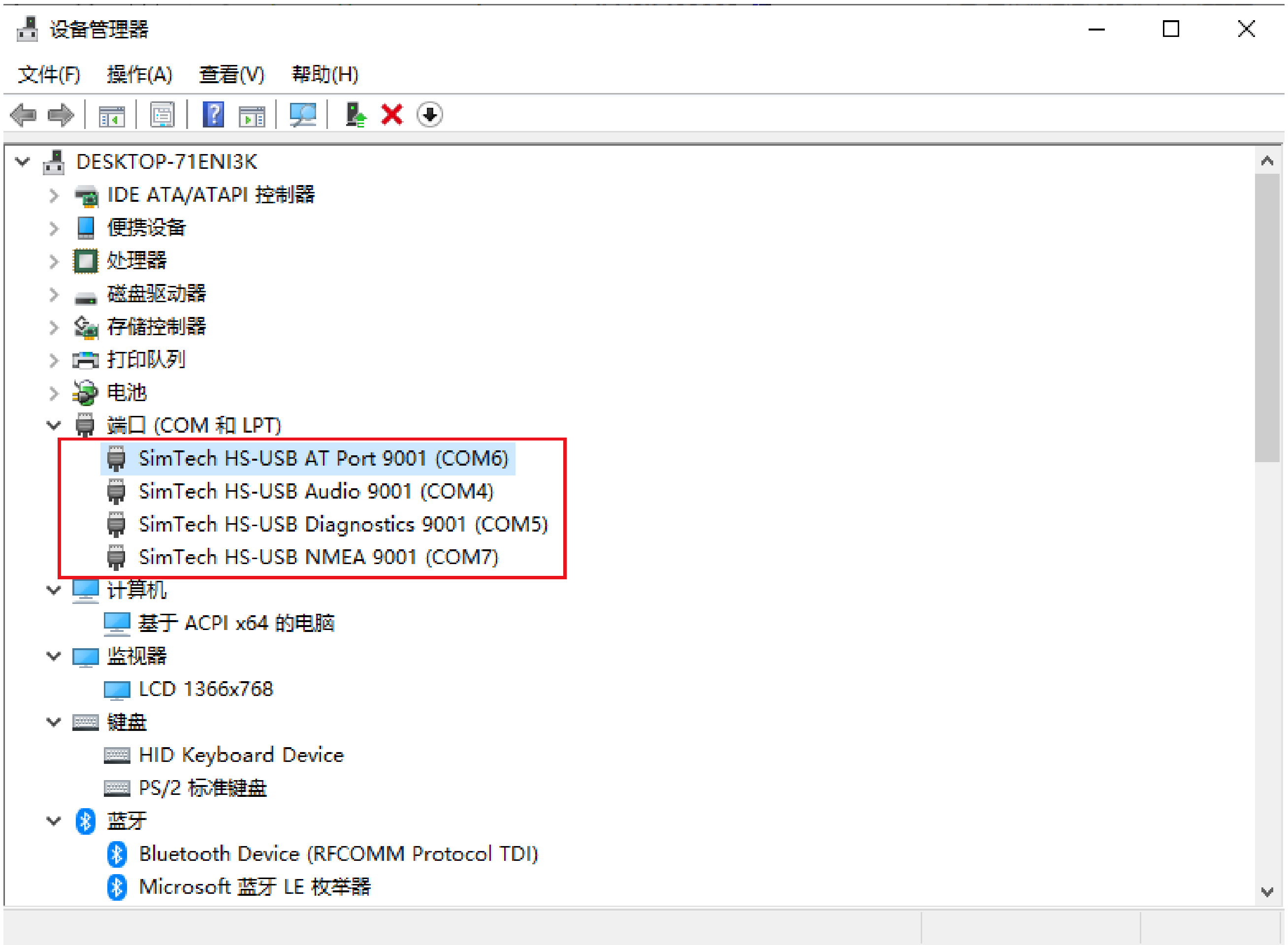
Pin Name	No.
CONFIG_2	75
GND	73
GND	71
	69
RESET#	67
	65
	63
	61
	59
GND	57
	55
	53
GND	51
	49
	47
GND	45
	43
	41
GND	39
USB_SS_RX_P	37
USB_SS_RX_M	35
GND	33
USB_SS_TX_P	31
USB_SS_TX_M	29
GND	27
	25
	23
	21
Notch	
Notch	
Notch	
Notch	
GND	11
USB_DM	9
USB_DP	7
GND	5
GND	3
	1

User Manual

Working With Windows

2. Direct Installation

Download a driver (Resource->Software->SIM8200 Driver) on your computer and then unzip it.
Enter SIM8200_OS_Driver\Windows directory.
For most hosts, you can enter "1_install" directory and then click "setup.exe" to install.

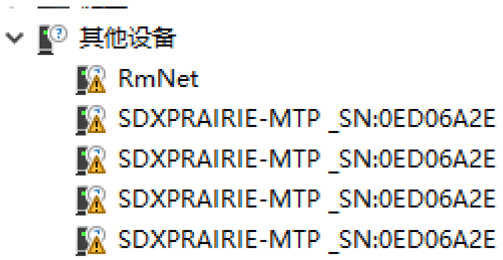


- DESKTOP-71ENI3K
  - IDE ATA/ATAPI 控制器
  - 便携设备
  - 处理器
  - 磁盘驱动器
  - 存储控制器
  - 打印队列
  - 电池
  - 端口 (COM 和 LPT)
    - SimTech HS-USB AT Port 9001 (COM6)
    - SimTech HS-USB Audio 9001 (COM4)
    - SimTech HS-USB Diagnostics 9001 (COM5)
    - SimTech HS-USB NMEA 9001 (COM7)
  - 计算机
    - 基于 ACPI x64 的电脑
  - 监视器
    - LCD 1366x768
  - 键盘
    - HID Keyboard Device
    - PS/2 标准键盘
  - 蓝牙
    - Bluetooth Device (RFCOMM Protocol TDI)
    - Microsoft 蓝牙 LE 枚举器

After connecting, a mobile network icon will appear. You can disconnect from other networks and test the mobile network.

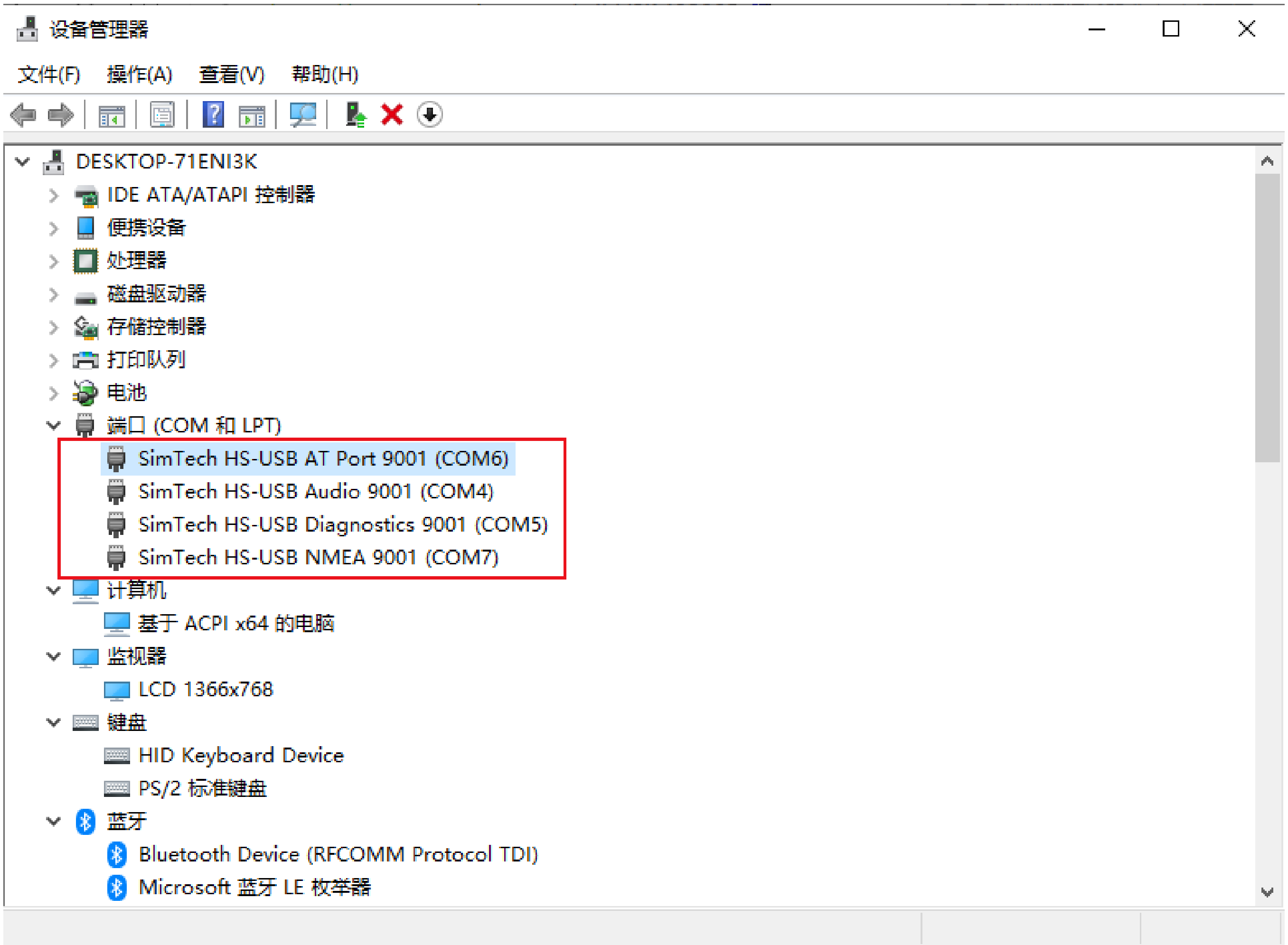
3. Install Driver Manually

For some hosts, the port may not appear even if "1_install" is installed. In this case, you need to use the files in "2_AddManully" to add them manually. The way to use it is to find the unrecognized device in the device manager and right-click to add the driver, as follows:
Power on the G module and turn the switch to ON, the module starts, and the computer will recognize 4 unknown devices (maybe some boards will recognize 5 or 6), only 5 in the picture:



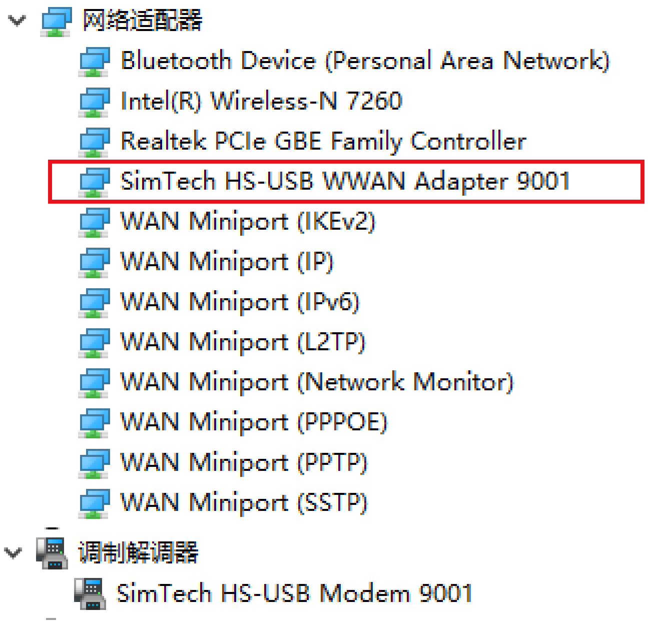
其他设备
RmNet
SDXPRAIRIE-MTP _SN:0ED06A2E
SDXPRAIRIE-MTP _SN:0ED06A2E
SDXPRAIRIE-MTP _SN:0ED06A2E
SDXPRAIRIE-MTP _SN:0ED06A2E

Right-click the device, update the driver manually, choose SIM8200_OS_Driver\Windows, and then choose the driver according to the version of your OS. You need to update for all the four/five/six devices.:



DESKTOP-71ENI3K
IDE ATA/ATAPI 控制器
便携设备
处理器
磁盘驱动器
存储控制器
打印队列
电池
端口 (COM 和 LPT)
SimTech HS-USB AT Port 9001 (COM6)
SimTech HS-USB Audio 9001 (COM4)
SimTech HS-USB Diagnostics 9001 (COM5)
SimTech HS-USB NMEA 9001 (COM7)
计算机
基于 ACPI x64 的电脑
监视器
LCD 1366x768
键盘
HID Keyboard Device
PS/2 标准键盘
蓝牙
Bluetooth Device (RFCOMM Protocol TDI)
Microsoft 蓝牙 LE 枚举器

Four COM ports: AT is used for AT command controlling, and Audio is used for dail up. Diagnostics is used for debugging, and NMEA is used for GPS.



- 网络适配器
  - Bluetooth Device (Personal Area Network)
  - Intel(R) Wireless-N 7260
  - Realtek PCIe GBE Family Controller
  - SimTech HS-USB WWAN Adapter 9001
  - WAN Miniport (IKEv2)
  - WAN Miniport (IP)
  - WAN Miniport (IPv6)
  - WAN Miniport (L2TP)
  - WAN Miniport (Network Monitor)
  - WAN Miniport (PPPOE)
  - WAN Miniport (PPTP)
  - WAN Miniport (SSTP)
- 调制解调器
  - SimTech HS-USB Modem 9001

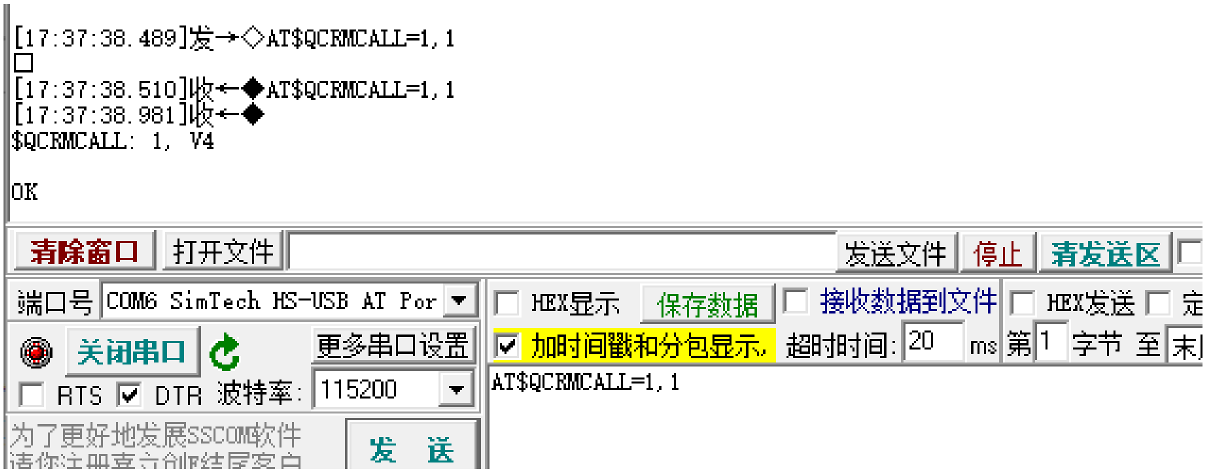
A mobile network will be set up automatically after updating, you can disconnect other networks and test it.

4. Manual NDIS Dial-up Internet

If the above 2 steps have been done, Windows cannot access the Internet, you need to manually start the NDIS dial-up.
Open the sim8200 AT port and send the command:

```
AT$QCRMCALL=1,1+Enter
```

After dialing is successful, as shown in the figure below, the computer can go online normally.



```
[17:37:38.489]发→◇AT$QCRMCALL=1,1
□
[17:37:38.510]收←◆AT$QCRMCALL=1,1
[17:37:38.981]收←◆
$QCRMCALL: 1, V4

OK
```

端口号 COM6 SimTech HS-USB AT Por

☐ RTS ☑ DTR 波特率: 115200

☐ HEX显示 ☐ 接收数据到文件 ☐ HEX发送 ☐ 定

☑ 加时间戳和分包显示, 超时时间: 20 ms 第 1 字节 至 末

AT$QCRMCALL=1,1

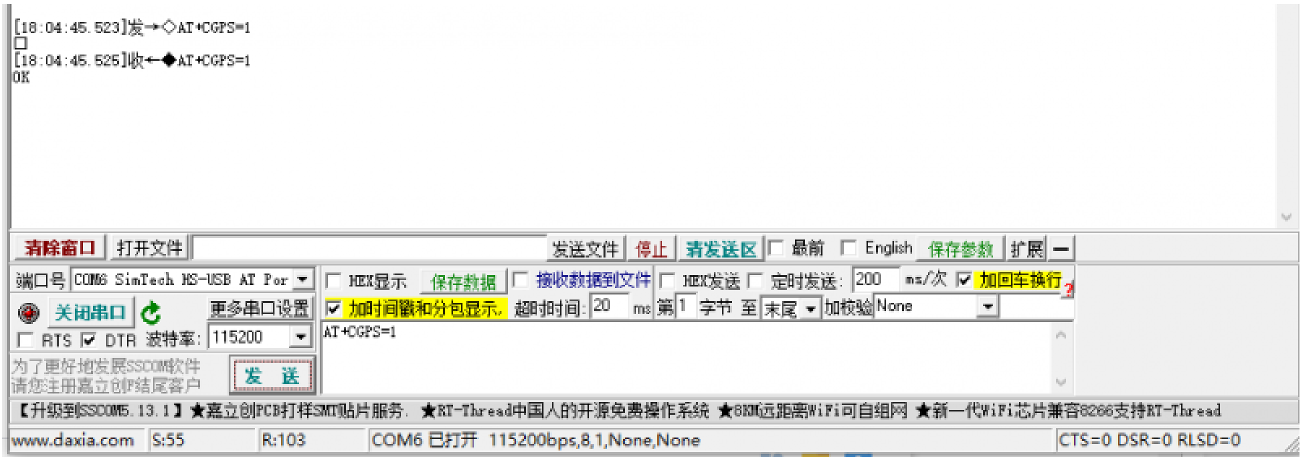
At this time, the NDIS dialing takes effect, and the computer can connect to the network; if it returns No Carrier, it may have dialed up, and you can go online to see it directly.

5. GPS Positioning

SIM8200EA-M2 is ANT5; SIM8202G-M2 is ANT2; SIM8262E-M2 is ANT3; see the module hardware manual for details

Connect the passive GPS antenna to the ANT5 of the module, and place the antenna outdoors facing the sky. Then send the AT command to turn on the GPS:

```
AT+CGPS=1 #Enter
```



```
[18:04:45.523]发→◇AT+CGPS=1
□
[18:04:45.525]收←◆AT+CGPS=1
OK
```

端口号 COM6 SimTech HS-USB AT Por

☐ RTS ☑ DTR 波特率: 115200

☐ HEX显示 ☐ 接收数据到文件 ☐ HEX发送 ☐ 定时发送: 200 ms/次 ☑ 加回车换行

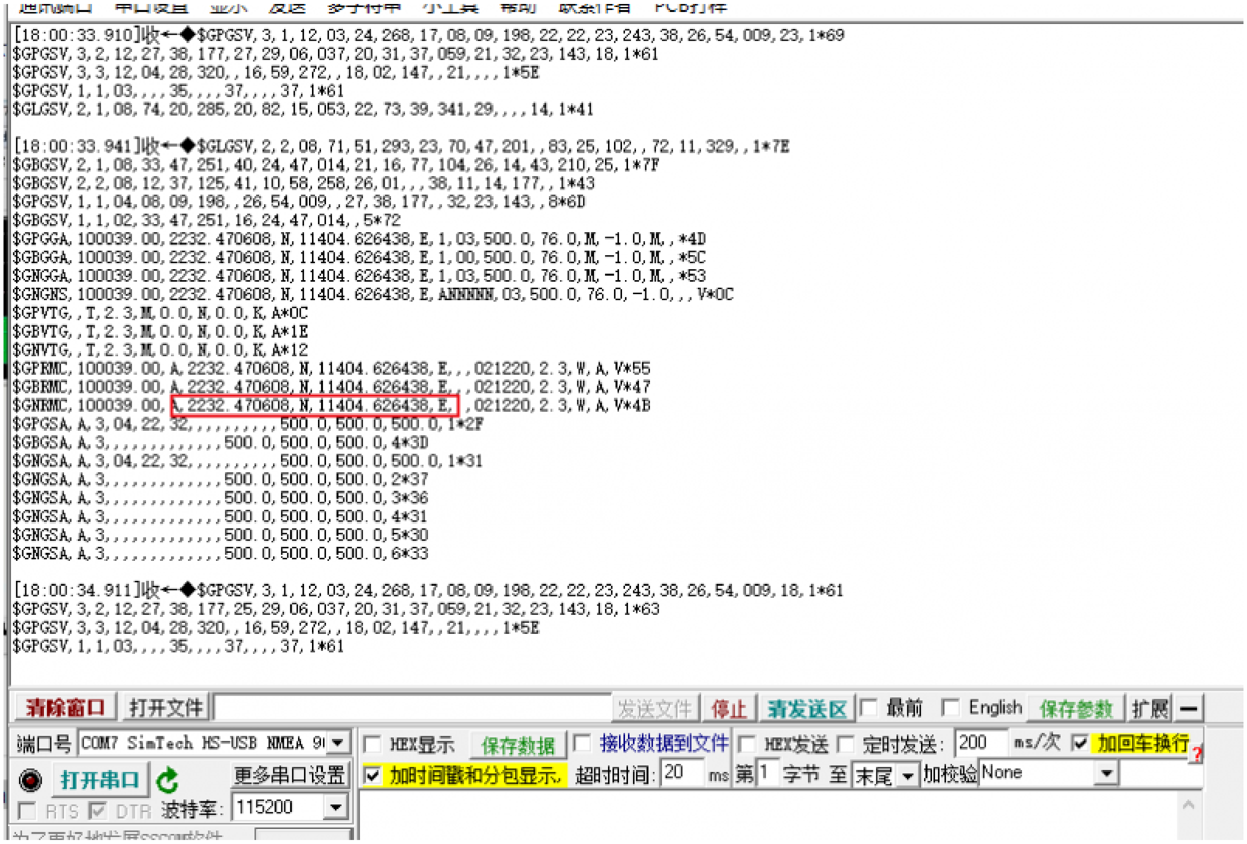
☑ 加时间戳和分包显示, 超时时间: 20 ms 第 1 字节 至 末尾 加校验 None

```
AT+CGPS=1
```

【升级到SSCOM5.13.1】★嘉立创PCB打样SMT贴片服务. ★RT-Thread中国人的开源免费操作系统 ★8KM远距离WiFi可自组网 ★新一代WiFi芯片兼容8266支持RT-Thread

www.daxia.com S:55 R:103 COM6 已打开 115200bps,8,1,None,None CTS=0 DSR=0 RLSD=0

Now open the NEMA port, you can get GPS data:



```
[18:00:33.910]收←◆$GPGSV,3,1,12,03,24,268,17,08,09,198,22,22,23,243,38,26,54,009,23,1*69
$GPGSV,3,2,12,27,38,177,27,29,06,037,20,31,37,059,21,32,23,143,18,1*61
$GPGSV,3,3,12,04,28,320,,16,59,272,,18,02,147,,21,,,,1*5E
$GPGSV,1,1,03,,,,35,,,,37,,,,37,1*61
$GLGSV,2,1,08,74,20,285,20,82,15,053,22,73,39,341,29,,,,14,1*41

[18:00:33.941]收←◆$GLGSV,2,2,08,71,51,293,23,70,47,201,,83,25,102,,72,11,329,,1*7E
$GBGSV,2,1,08,33,47,251,40,24,47,014,21,16,77,104,26,14,43,210,25,1*7F
$GBGSV,2,2,08,12,37,125,41,10,58,258,26,01,,,38,11,14,177,,1*43
$GPGSV,1,1,04,08,09,198,,26,54,009,,27,38,177,,32,23,143,,8*6D
$GBGSV,1,1,02,33,47,251,16,24,47,014,,5*72
$GPGGA,100039.00,2232.470608,N,11404.626438,E,1,03,500.0,76.0,M,-1.0,M,,*4D
$GBGGA,100039.00,2232.470608,N,11404.626438,E,1,00,500.0,76.0,M,-1.0,M,,*5C
$GNGGA,100039.00,2232.470608,N,11404.626438,E,1,03,500.0,76.0,M,-1.0,M,,*53
$GNGNS,100039.00,2232.470608,N,11404.626438,E,ANNNNN,03,500.0,76.0,-1.0,,,V*0C
$GPVTG,,T,2.3,M,0.0,N,0.0,K,A*0C
$GBVTG,,T,2.3,M,0.0,N,0.0,K,A*1E
$GNVTG,,T,2.3,M,0.0,N,0.0,K,A*12
$GPRMC,100039.00,A,2232.470608,N,11404.626438,E,,,021220,2.3,W,A,V*55
$GBRMC,100039.00,A,2232.470608,N,11404.626438,E,,,021220,2.3,W,A,V*47
$GNRMC,100039.00,A,2232.470608,N,11404.626438,E,,,021220,2.3,W,A,V*4B
$GPGSA,A,3,04,22,32,,,,,,,,,,500.0,500.0,500.0,1*2F
$GBGSA,A,3,,,,,,,,,,,,,500.0,500.0,500.0,4*3D
$GNGSA,A,3,04,22,32,,,,,,,,,,500.0,500.0,500.0,1*31
$GNGSA,A,3,,,,,,,,,,,,,500.0,500.0,500.0,2*37
$GNGSA,A,3,,,,,,,,,,,,,500.0,500.0,500.0,3*36
$GNGSA,A,3,,,,,,,,,,,,,500.0,500.0,500.0,4*31
$GNGSA,A,3,,,,,,,,,,,,,500.0,500.0,500.0,5*30
$GNGSA,A,3,,,,,,,,,,,,,500.0,500.0,500.0,6*33

[18:00:34.911]收←◆$GPGSV,3,1,12,03,24,268,17,08,09,198,22,22,23,243,38,26,54,009,18,1*61
$GPGSV,3,2,12,27,38,177,25,29,06,037,20,31,37,059,21,32,23,143,18,1*63
$GPGSV,3,3,12,04,28,320,,16,59,272,,18,02,147,,21,,,,1*5E
$GPGSV,1,1,03,,,,35,,,,37,,,,37,1*61
```

端口号 COM7 SimTech HS-USB NMEA 9

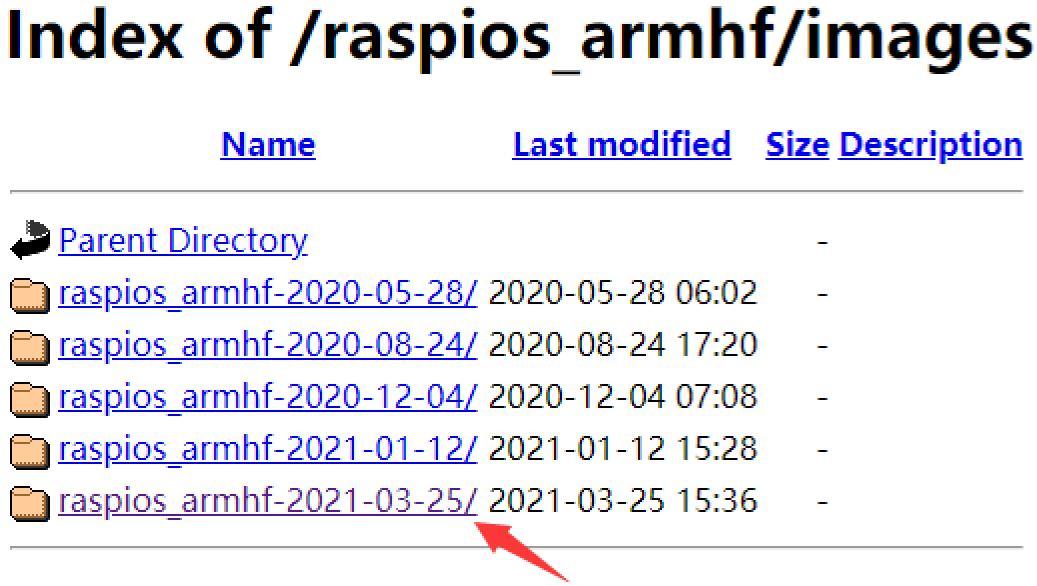
☐ RTS ☑ DTR 波特率: 115200

☐ HEX显示 ☐ 接收数据到文件 ☐ HEX发送 ☐ 定时发送: 200 ms/次 ☑ 加回车换行

☑ 加时间戳和分包显示, 超时时间: 20 ms 第 1 字节 至 末尾 加校验 None

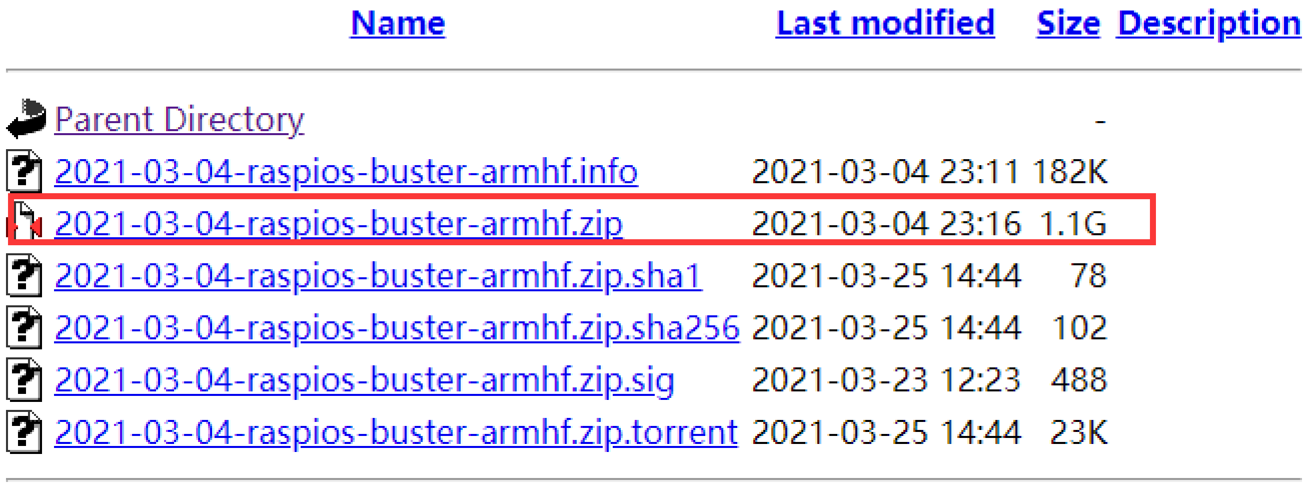
Working With Raspberry Pi

It is recommended that you use the latest system image (the latest system address) of the Raspberry Pi, and the system's Linux kernel version 5.4. If your kernel is different, it is recommended that you update to the same version as this one. This tutorial is based on the 5.4 kernel, which can minimize errors.



Index of /raspios_armhf/images

Name	Last modified	Size	Description
Parent Directory		-	
raspios_armhf-2020-05-28/	2020-05-28 06:02	-	
raspios_armhf-2020-08-24/	2020-08-24 17:20	-	
raspios_armhf-2020-12-04/	2020-12-04 07:08	-	
raspios_armhf-2021-01-12/	2021-01-12 15:28	-	
raspios_armhf-2021-03-25/	2021-03-25 15:36	-	



Name	Last modified	Size	Description
Parent Directory		-	
2021-03-04-raspios-buster-armhf.info	2021-03-04 23:11	182K	
2021-03-04-raspios-buster-armhf.zip	2021-03-04 23:16	1.1G	
2021-03-04-raspios-buster-armhf.zip.sha1	2021-03-25 14:44	78	
2021-03-04-raspios-buster-armhf.zip.sha256	2021-03-25 14:44	102	
2021-03-04-raspios-buster-armhf.zip.sig	2021-03-23 12:23	488	
2021-03-04-raspios-buster-armhf.zip.torrent	2021-03-25 14:44	23K	

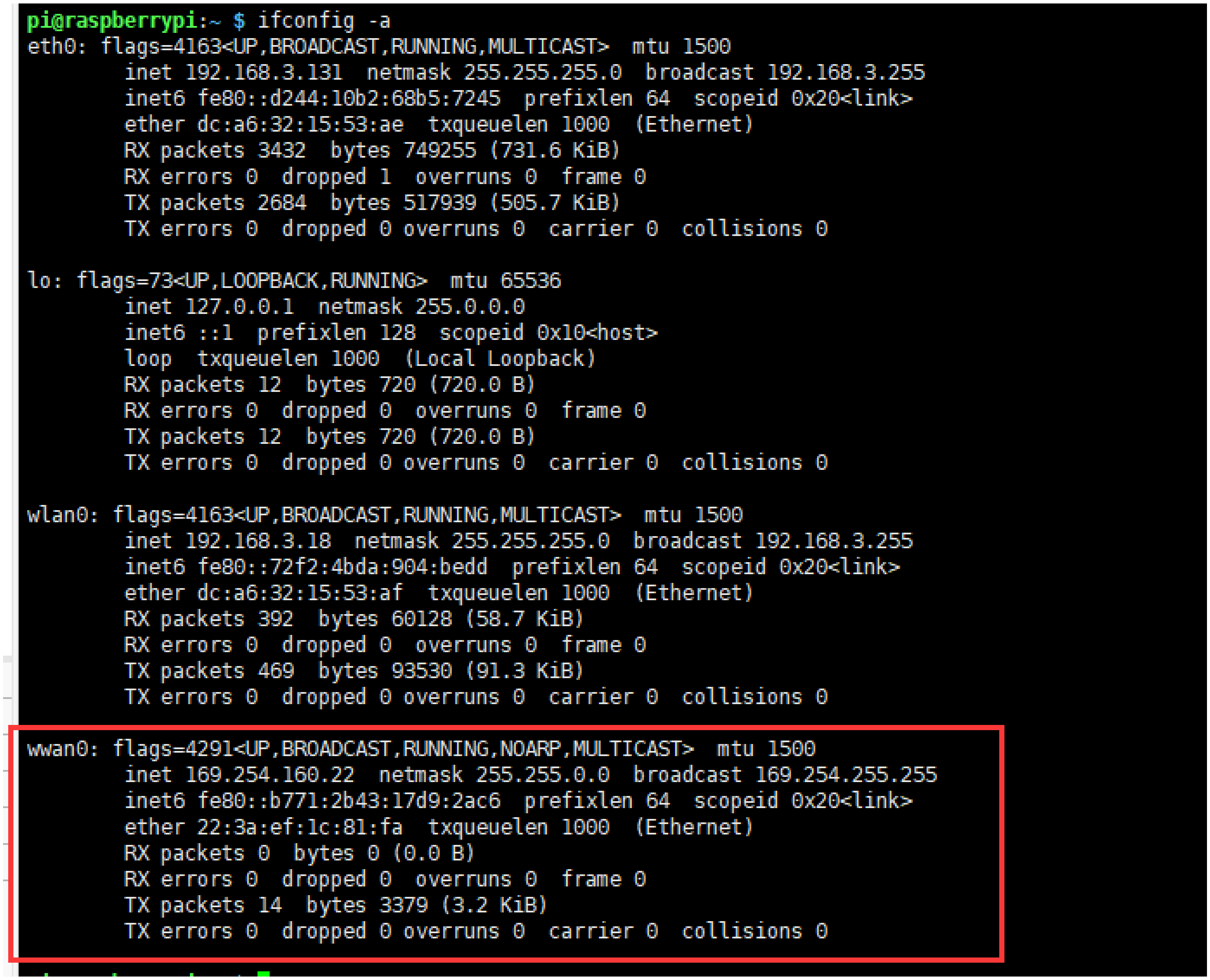
If you are using other Linux systems, please download the driver under SIM8200_OS_Driver\linux, and transplant it according to the document under it; you can also use the built-in driver of the system for SIM820X RNDIS dial-up internet.

6. Configuration At The First Time

Please don't type the wrong letter, it's better to copy and paste.

```
sudo apt-get install p7zip-full

wget https://www.waveshare.com/w/upload/8/89/SIM8200_for_RPI.7z

7z x SIM8200_for_RPI.7z  -r -o./SIM8200_for_RPI

sudo chmod 777 -R SIM8200_for_RPI

cd SIM8200_for_RPI

sudo ./install.sh
```

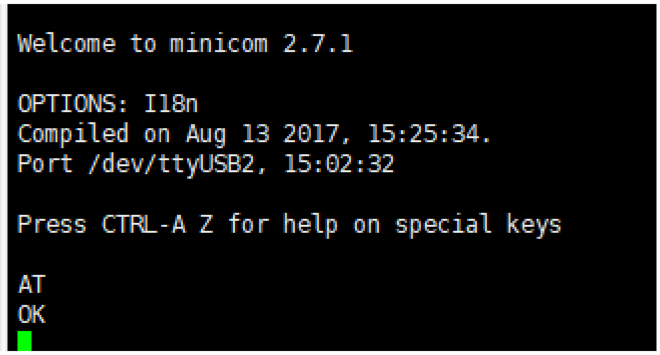
Please do not delete or modify the "option" directory, "qmi_wwan_simcom" directory, "default.script", "install.sh" files, otherwise it will affect the loading of the driver.
If there is an error, please confirm whether the system is "2020-08-20-raspios-buster-armhf", and take a screenshot of the error message so that engineers can help you analyze and solve the problem.
Run "ifconfig -a" to see that "WWAN0" has been generated.



```
pi@raspberrypi:~ $ ifconfig -a
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.3.131  netmask 255.255.255.0  broadcast 192.168.3.255
        inet6 fe80::d244:10b2:68b5:7245  prefixlen 64  scopeid 0x20<link>
        ether dc:a6:32:15:53:ae  txqueuelen 1000  (Ethernet)
        RX packets 3432  bytes 749255 (731.6 KiB)
        RX errors 0  dropped 1  overruns 0  frame 0
        TX packets 2684  bytes 517939 (505.7 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 12  bytes 720 (720.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 12  bytes 720 (720.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.3.18  netmask 255.255.255.0  broadcast 192.168.3.255
        inet6 fe80::72f2:4bda:904:bedd  prefixlen 64  scopeid 0x20<link>
        ether dc:a6:32:15:53:af  txqueuelen 1000  (Ethernet)
        RX packets 392  bytes 60128 (58.7 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 469  bytes 93530 (91.3 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wwan0: flags=4291<UP,BROADCAST,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 169.254.160.22  netmask 255.255.0.0  broadcast 169.254.255.255
        inet6 fe80::b771:2b43:17d9:2ac6  prefixlen 64  scopeid 0x20<link>
        ether 22:3a:ef:1c:81:fa  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 14  bytes 3379 (3.2 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Test AT Command

```
sudo apt-get install minicom

sudo minicom -D /dev/ttyUSB2
```

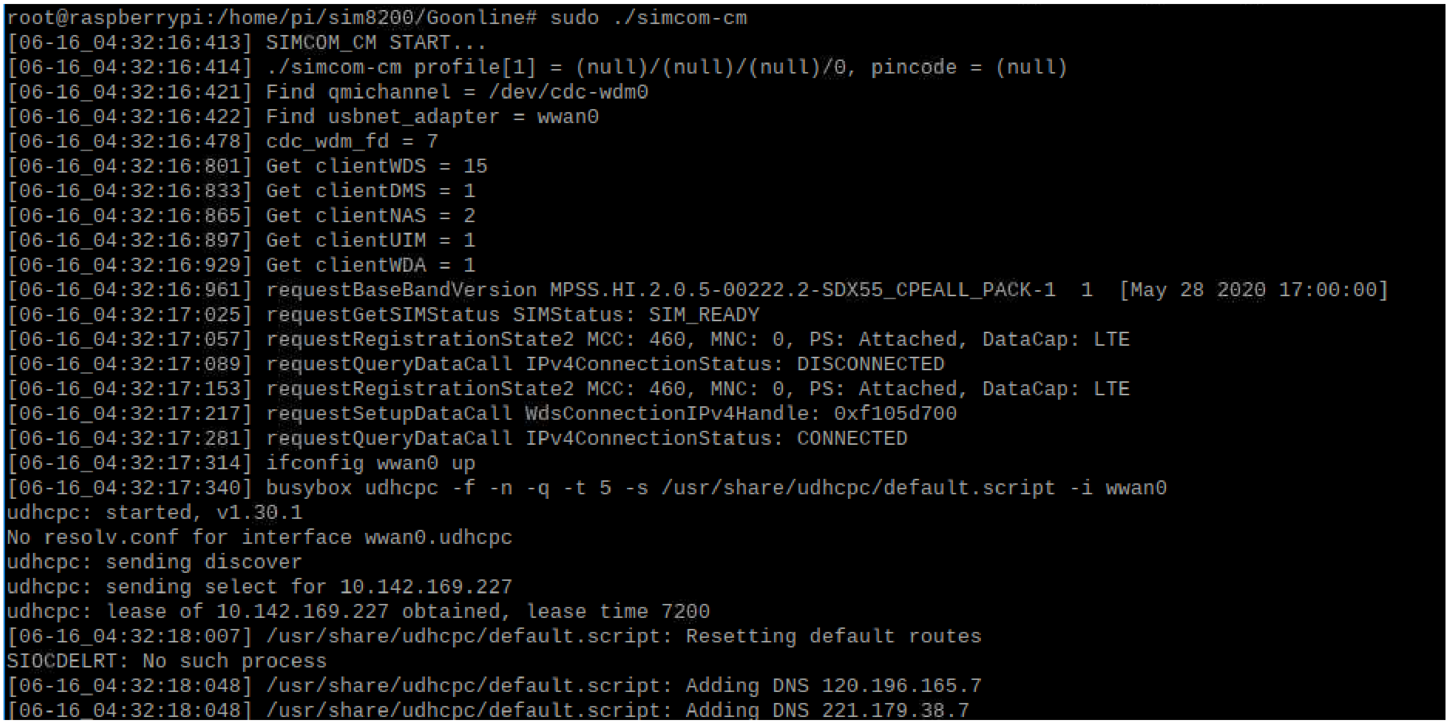


```
Welcome to minicom 2.7.1

OPTIONS: I18n
Compiled on Aug 13 2017, 15:25:34.
Port /dev/ttyUSB2, 15:02:32

Press CTRL-A Z for help on special keys

AT
OK
```

5G Networking

```
cd Goonline

make

sudo ./simcom-cm
```

After running codes, DNS information is shown in the figure below:

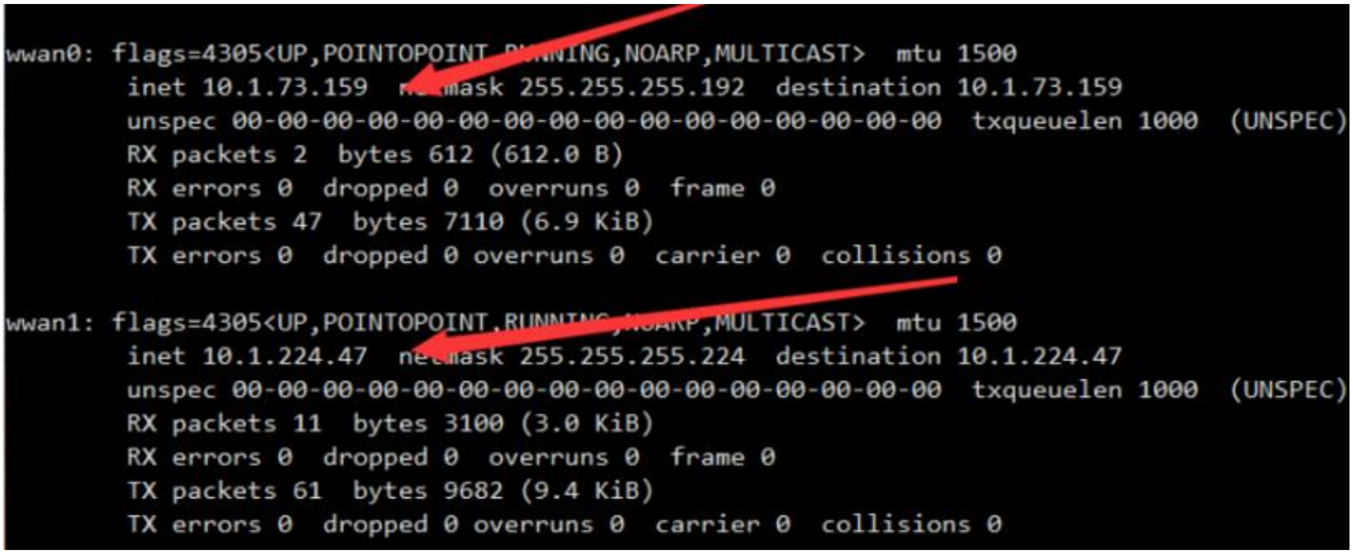


```
root@raspberrypi:/home/pi/sim8200/Goonline# sudo ./simcom-cm
[06-16_04:32:16:413] SIMCOM_CM START...
[06-16_04:32:16:414] ./simcom-cm profile[1] = (null)/(null)/(null)/0, pincode = (null)
[06-16_04:32:16:421] Find qmichannel = /dev/cdc-wdm0
[06-16_04:32:16:422] Find usbnet_adapter = wwan0
[06-16_04:32:16:478] cdc_wdm_fd = 7
[06-16_04:32:16:801] Get clientWDS = 15
[06-16_04:32:16:833] Get clientDMS = 1
[06-16_04:32:16:865] Get clientNAS = 2
[06-16_04:32:16:897] Get clientUIM = 1
[06-16_04:32:16:929] Get clientWDA = 1
[06-16_04:32:16:961] requestBaseBandVersion MPSS.HI.2.0.5-00222.2-SDX55_CPEALL_PACK-1  1  [May 28 2020 17:00:00]
[06-16_04:32:17:025] requestGetSIMStatus SIMStatus: SIM_READY
[06-16_04:32:17:057] requestRegistrationState2 MCC: 460, MNC: 0, PS: Attached, DataCap: LTE
[06-16_04:32:17:089] requestQueryDataCall IPv4ConnectionStatus: DISCONNECTED
[06-16_04:32:17:153] requestRegistrationState2 MCC: 460, MNC: 0, PS: Attached, DataCap: LTE
[06-16_04:32:17:217] requestSetupDataCall WdsConnectionIPv4Handle: 0xf105d700
[06-16_04:32:17:281] requestQueryDataCall IPv4ConnectionStatus: CONNECTED
[06-16_04:32:17:314] ifconfig wwan0 up
[06-16_04:32:17:340] busybox udhcpc -f -n -q -t 5 -s /usr/share/udhcpc/default.script -i wwan0
udhcpc: started, v1.30.1
No resolv.conf for interface wwan0.udhcpc
udhcpc: sending discover
udhcpc: sending select for 10.142.169.227
udhcpc: lease of 10.142.169.227 obtained, lease time 7200
[06-16_04:32:18:007] /usr/share/udhcpc/default.script: Resetting default routes
SIOCDELRT: No such process
[06-16_04:32:18:048] /usr/share/udhcpc/default.script: Adding DNS 120.196.165.7
[06-16_04:32:18:048] /usr/share/udhcpc/default.script: Adding DNS 221.179.38.7
```

After connecting two SIM820X to the Raspberry Pi through USB, two network cards—

wwan0 and wwan1 can be recognized. The two network cards can be dialed at the same time through the following commands: (The network speed cannot be superimposed)

```
sudo ./simcom-cm -i wwan0

sudo ./simcom-cm -i wwan1
```



```
wwan0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 10.1.73.159  netmask 255.255.255.192  destination 10.1.73.159
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 1000  (UNSPEC)
        RX packets 2  bytes 612 (612.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 47  bytes 7110 (6.9 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wwan1: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 10.1.224.47  netmask 255.255.255.224  destination 10.1.224.47
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 1000  (UNSPEC)
        RX packets 11  bytes 3100 (3.0 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 61  bytes 9682 (9.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

- Note: If the IP cannot be obtained or the networking is not successful, use the following commands to obtain the IP and set the DNS networking:

```
sudo dhclient -v wwan0

sudo route add -net 0.0.0.0 wwan0
```

Auto-run

If you want to set the codes auto-run after booting, you can modify rc.local file:

```
sudo nano /etc/rc.local
```

Add the line to file as below:

```
sudo /home/pi/SIM8200-M2_5G_HAT_code/Goonline/simcom-cm &
```

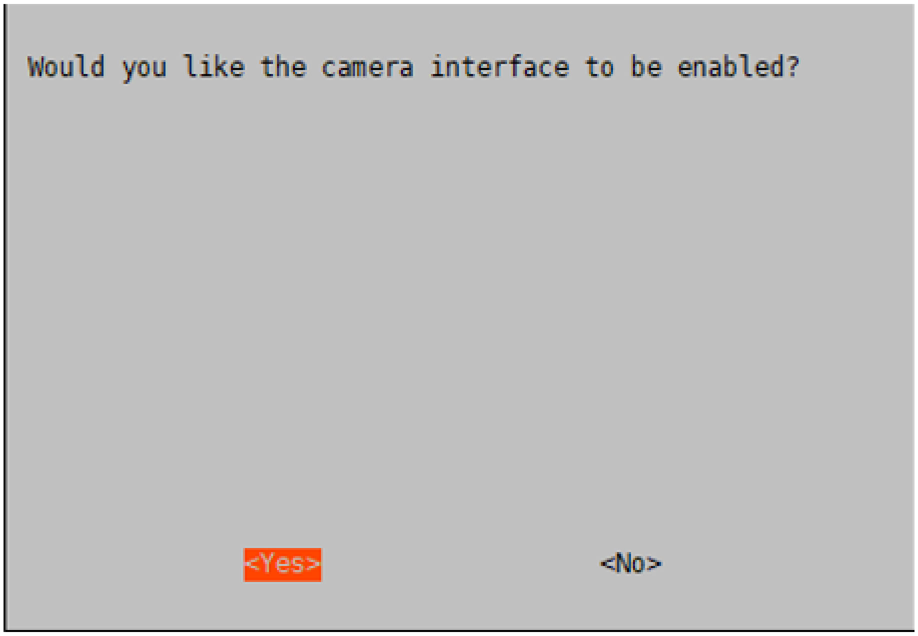
Note that you have to add "&" to the end of the command, make sure that the command can be run in the background, or the Pi may not boot normally.

Live streaming with ffmpeg

If you are using the 2020-08-20-raspios-buster-armhf image, then you don't need to install anything as the system already comes with ffmpeg.
Going straight to the topic, assuming you already have a camera and it's properly connected to the Raspberry Pi, then proceed with the tutorial.

- The camera must be enabled by running the raspi-config command before using it:

```
sudo raspi-config

Select Enable Camera, select YES
```



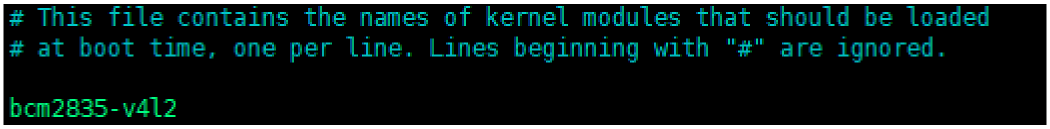
Would you like the camera interface to be enabled?

- If it is a camera using the CSI interface, the system cannot find the device node of /dev/video0. A line for bcm2835-v4l2 needs to be added to the /etc/modules file:

```
sudo nano /etc/modules
```

add:

```
bcm2835-v4l2
```

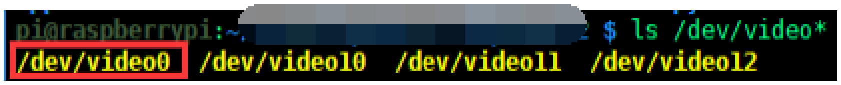


```
# This file contains the names of kernel modules that should be loaded
# at boot time, one per line. Lines beginning with "#" are ignored.

bcm2835-v4l2
```

Then after the system starts, the system will load the module name in this file and restart the system:

```
sudo reboot

ls /dev/video*
```

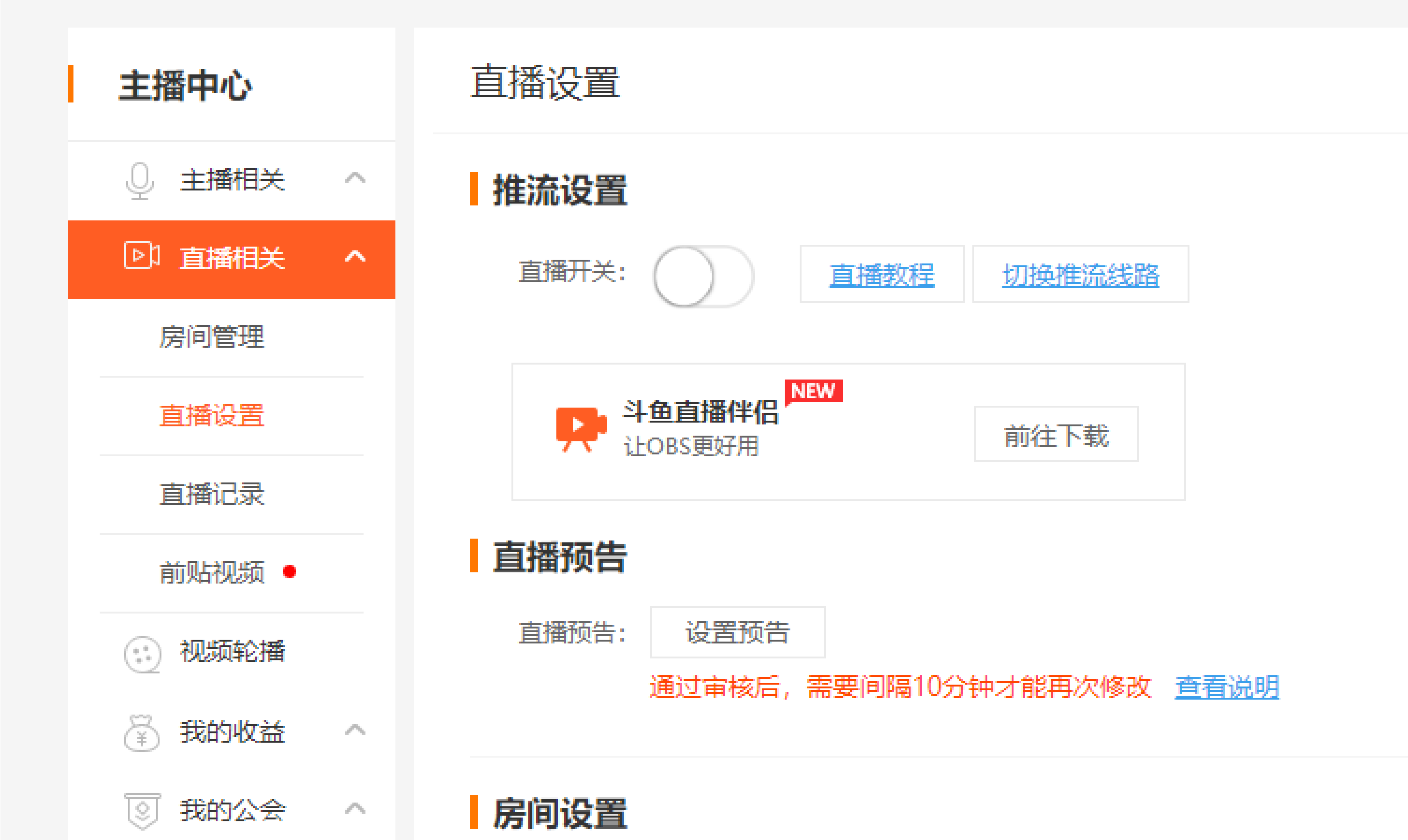
The video0 device node is found below.



```
pi@raspberrypi:~[redacted] $ ls /dev/video*
/dev/video0  /dev/video10  /dev/video11  /dev/video12
```

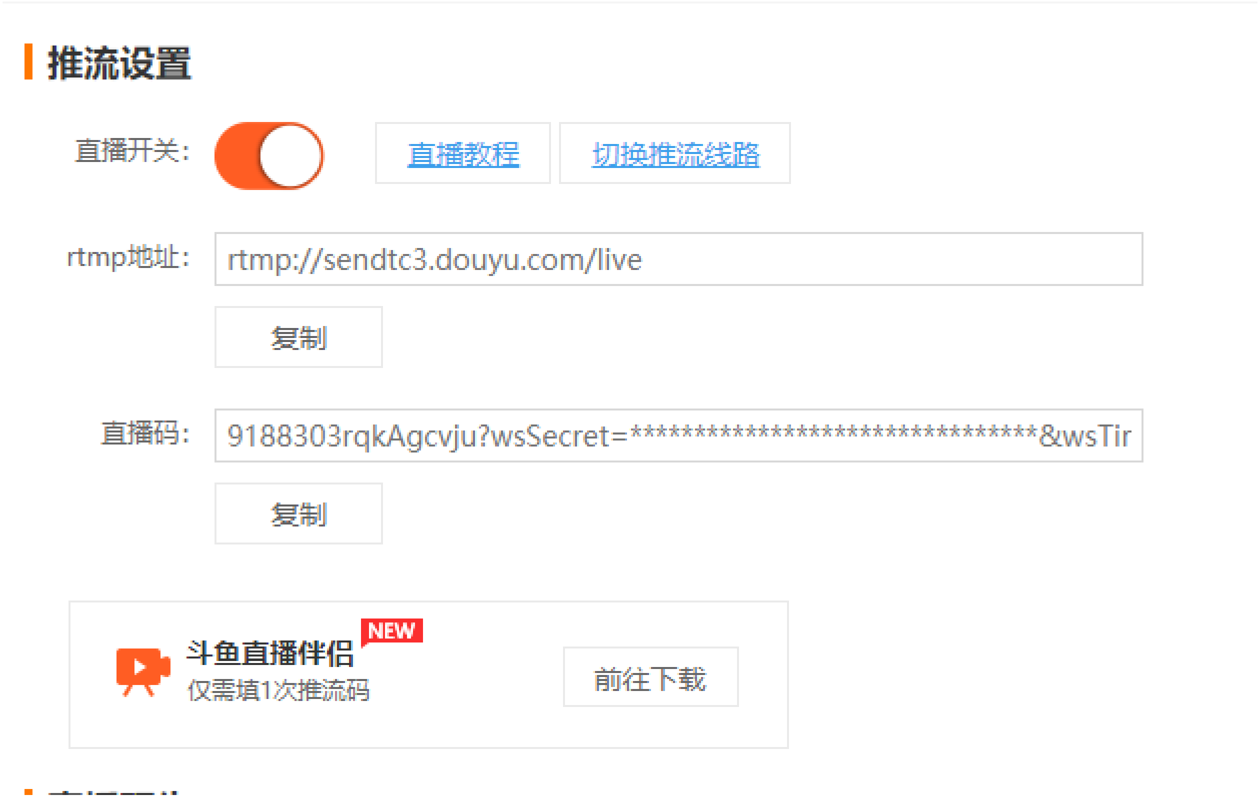
Note: If after performing the first steps, please confirm that the operation and instructions are correct.

Suppose you are using Douyu Live now, register your account and enable the host function, open Douyu Live Host Center, and find the live broadcast settings.



通过审核后，需要间隔10分钟才能再次修改

Open video plug flow setting:



推流设置

直播开关：

rtmp地址：rtmp://sendtc3.douyu.com/live

直播码：9188303rqkAgcvju?wsSecret=********************************&wsTir

斗鱼直播伴侣

仅需填1次推流码

The rtmp address and live code will be obtained, and the Raspberry Pi will execute the command:

```
ffmpeg -f video4linux2 -s 640x480 -r 25 -i /dev/video0 -c:v h264_omx -f flv "rtmp
address/live code"
```

For example: open a terminal first, runs the 5G network:

```
cd Goonline

sudo ./simcom-cm
```

and then open another terminal, run the following command:

```
ffmpeg -f video4linux2 -s 640x480 -r 25 -i /dev/video0 -c:v h264_omx -f flv "rtmp:
//sendtc3.douyu.com/live/9188303rTNGmU7CS?wsSecret=ef762877aae120262eaf23c3f60a28b
f&wsTime=5f59dbf0&wsSeek=off&wm=0&tw=0&roirecognition=0"
```

rtmp://sendtc3.douyu.com/live is the address, and the next section is the live code. Enter the room number at this time, you can observe the live broadcast, the delay is about 1-2S.

How To Use OpenWrt

Introduction

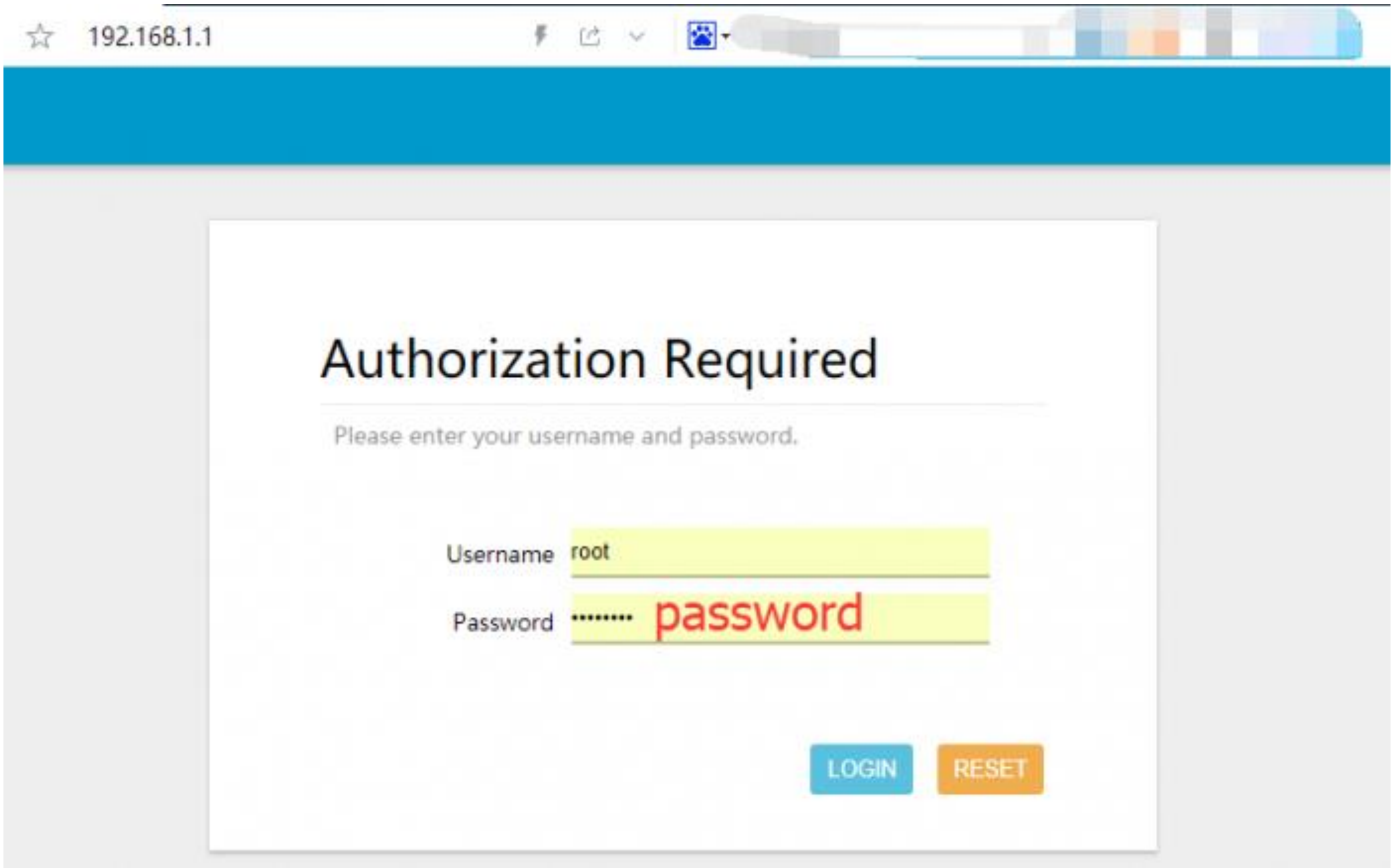
Soft routing is using desktops or servers and other equipment with software. It mainly depends on the settings of the software to achieve the functions of the router. The hard routing is a unique hardware device, including a processor, power supply, and embedded software to provide router functionality.
OpenWrt is a very popular soft routing system. It is a highly modular and highly automated embedded Linux system with powerful network components and scalability. It is often used in industrial control equipment, routers, and other equipment.
In addition to the functions of general home routers, OpenWrt soft routing can also achieve port forwarding, intranet penetration, 4G networking, FTP server and more powerful functions.

• Program the image

Download the RPI OpenWrt system, unzip the system in the Imgs directory, and use the burning tool to burn the system to the SD card.

• Login & Initial Settings

- After the OpenWrt system is turned on, the Raspberry Pi is equivalent to a router. Therefore, you can use a network cable to connect the Raspberry Pi to the computer according to the use of the router (you can also use the mobile phone to search for WIFI, the default name is "OpenWrt"). Enter 192.168.1.1 on the webpage, the default username: root, the default password: password, and enter the Web management interface of OpenWrt.



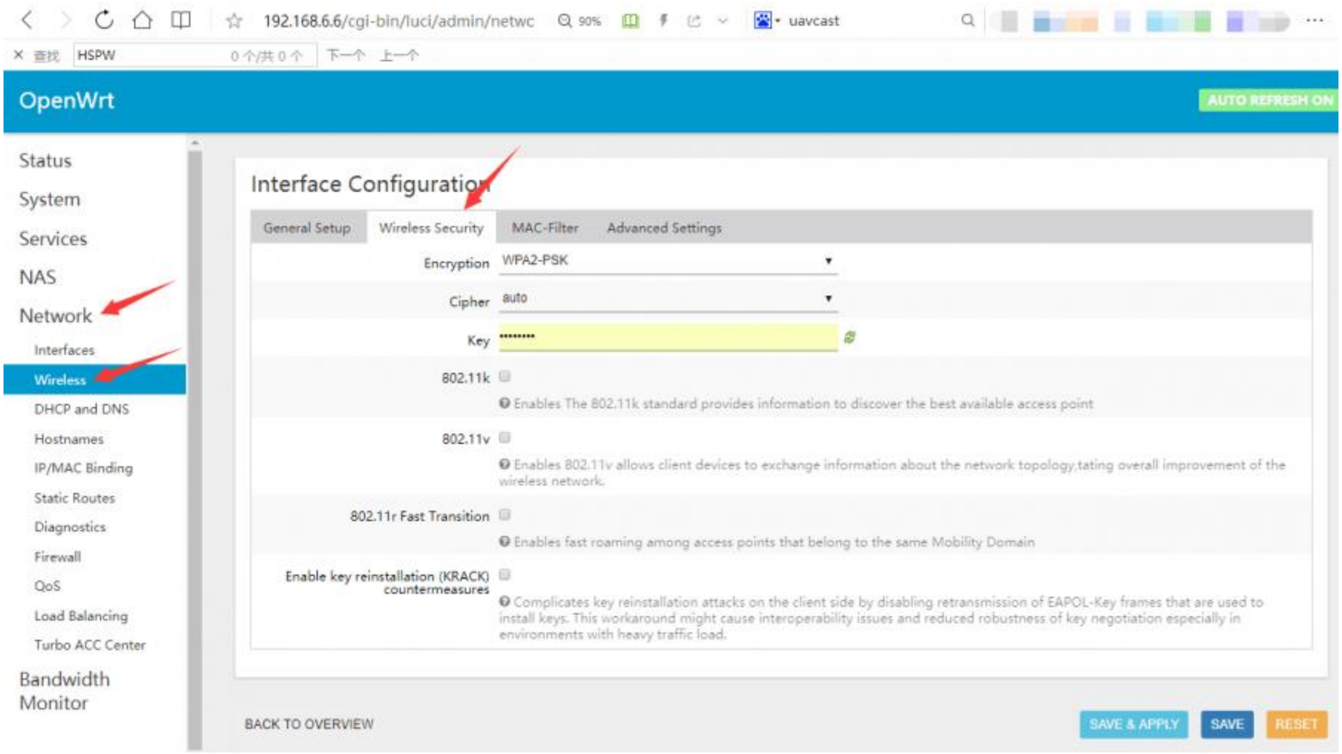
Authorization Required

Please enter your username and password.

Username: root

Password: ******** password

- Set WIFI password: Network -> Wireless -> Modify -> Interface Configuration -> Wireless Security, as shown below:



Interface Configuration

Encryption: WPA2-PSK

Cipher: auto

Key: ********

802.11k □
Enables The 802.11k standard provides information to discover the best available access point

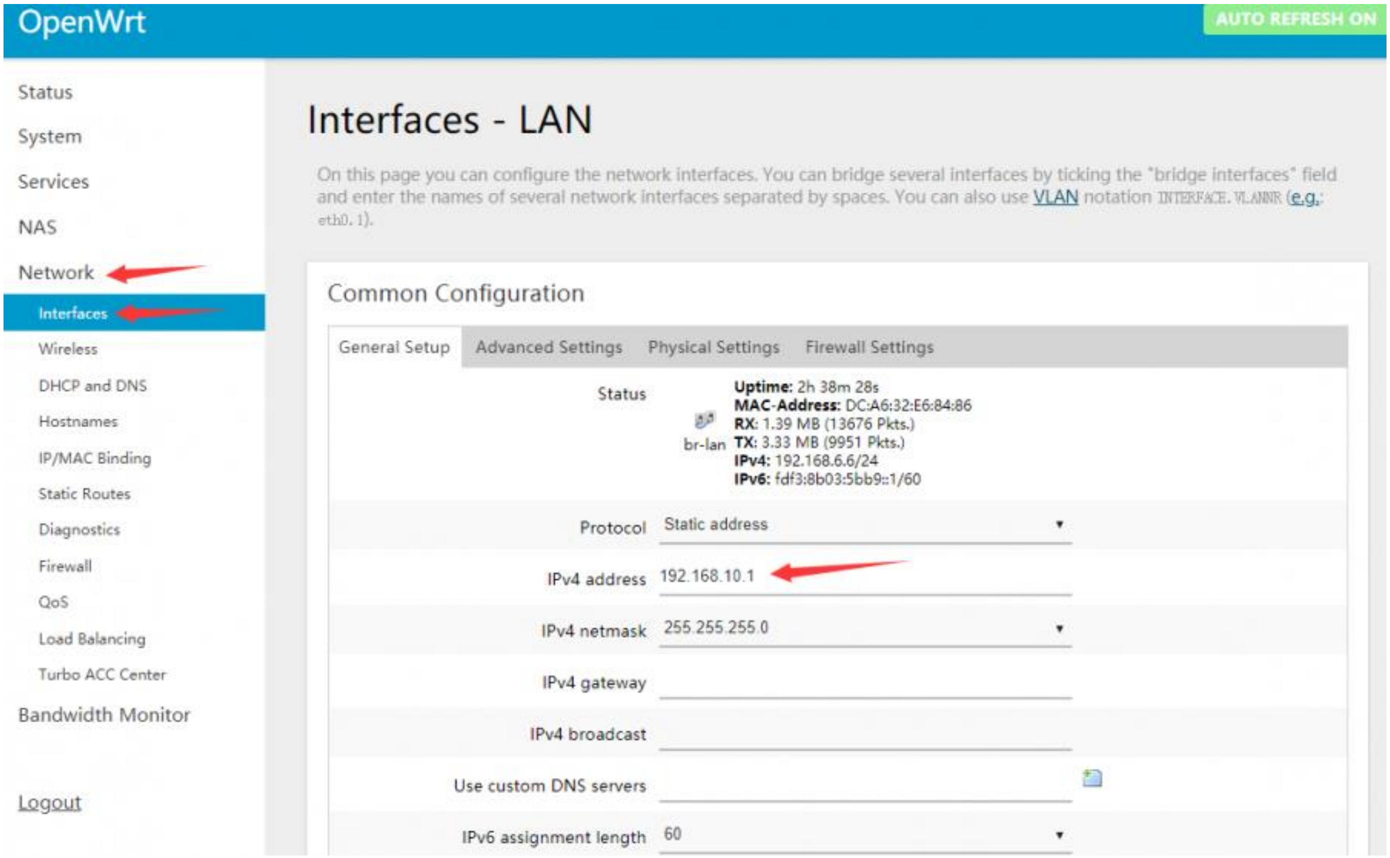
802.11v □
Enables 802.11v allows client devices to exchange information about the network topology,tating overall improvement of the wireless network.

802.11r Fast Transition □
Enables fast roaming among access points that belong to the same Mobility Domain

Enable key reinstallation (KRACK) countermeasures □
Complicates key reinstallation attacks on the client side by disabling retransmission of EAPOL-Key frames that are used to install keys. This workaround might cause interoperability issues and reduced robustness of key negotiation especially in environments with heavy traffic load.

- Create the new interface: Network -> Interface -> Create interface

- Modify the IPv4 address of the lan port to a different IP that is not the same as the lan port IP of other routers in your home. (Many routers default the lan port IP to 192.168.1.1. If you do not modify the IP of the OpenWrt, it will easily lead to conflicts and fail to connect to the Internet.)
- If necessary, it is also recommended to disable the IPv6 allocation length. After the modification is completed, click "Save & Apply", and re-use 192.168.10.1 to access the OpenWrt console.



Interfaces - LAN

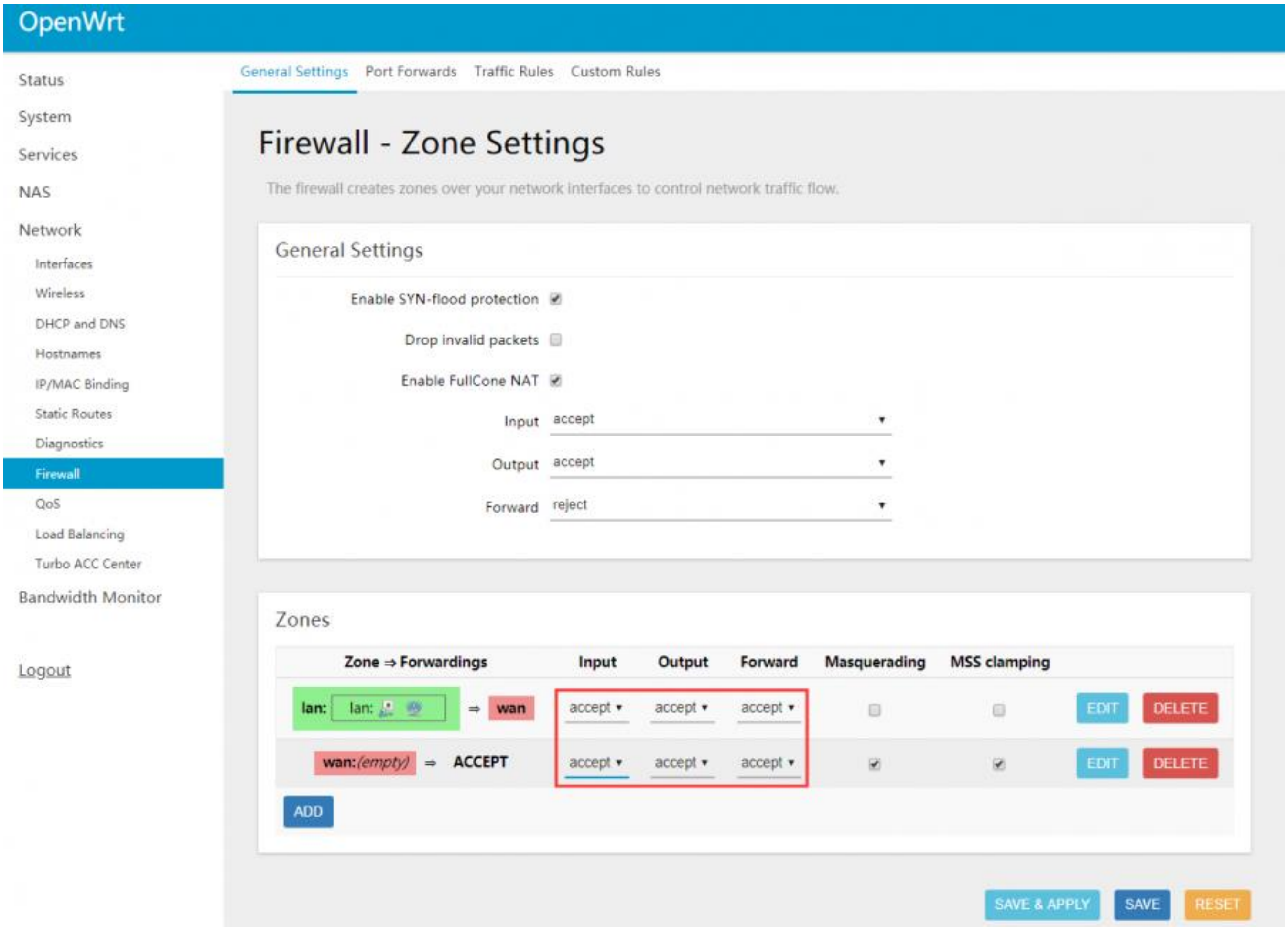
On this page you can configure the network interfaces. You can bridge several interfaces by ticking the "bridge interfaces" field and enter the names of several network interfaces separated by spaces. You can also use VLAN notation INTERFACE.VLANNR (e.g.: eth0.1).

Common Configuration

Status br-lan **Uptime:** 2h 38m 28s
MAC-Address: DC:A6:32:E6:84:86
RX: 1.39 MB (13676 Pkts.)
TX: 3.33 MB (9951 Pkts.)
IPv4: 192.168.6.6/24
IPv6: fdf3:8b03:5bb9::1/60

Protocol: Static address
IPv4 address: 192.168.10.1
IPv4 netmask: 255.255.255.0
IPv4 gateway:
IPv4 broadcast:
Use custom DNS servers:
IPv6 assignment length: 60

- In addition, it is recommended to adjust the Firewall setting to connect the OpenWrt terminal and Web management interface through the local area.
- Network —>Firewall, change all "reject" to "accept", click "Save & Apply" after modification, as shown in the picture below:



Firewall - Zone Settings

The firewall creates zones over your network interfaces to control network traffic flow.

General Settings

Enable SYN-flood protection ☑

Drop invalid packets □

Enable FullCone NAT ☑

Input: accept

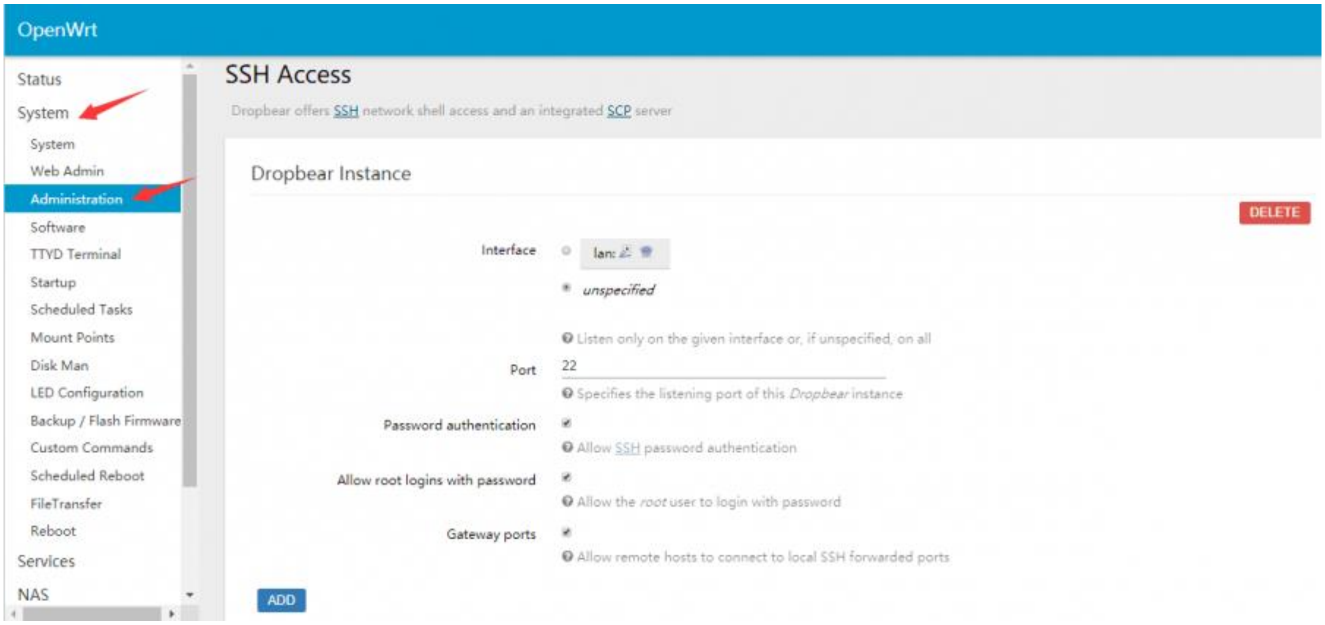
Output: accept

Forward: reject

Zones

Zone ⇒ Forwardings	Input	Output	Forward	Masquerading	MSS clamping		
lan: lan: ⇒ wan	accept	accept	accept	□	□	EDIT	DELETE
wan: *(empty)* ⇒ ACCEPT	accept	accept	accept	☑	☑	EDIT	DELETE

- And then select System -> Administration, modify the allowed interface for SSH access to "unspecified" (that is, any interface can be accessed by ssh), check the Gateway port, and click "Save & Apply" after the modification is completed.



SSH Access

Dropbear offers SSH network shell access and an integrated SCP server

Dropbear Instance

Interface ○ lan:

● *unspecified*

Listen only on the given interface or, if unspecified, on all

Port 22

Specifies the listening port of this *Dropbear* instance

Password authentication ☑

Allow SSH password authentication

Allow root logins with password ☑

Allow the *root* user to login with password

Gateway ports ☑

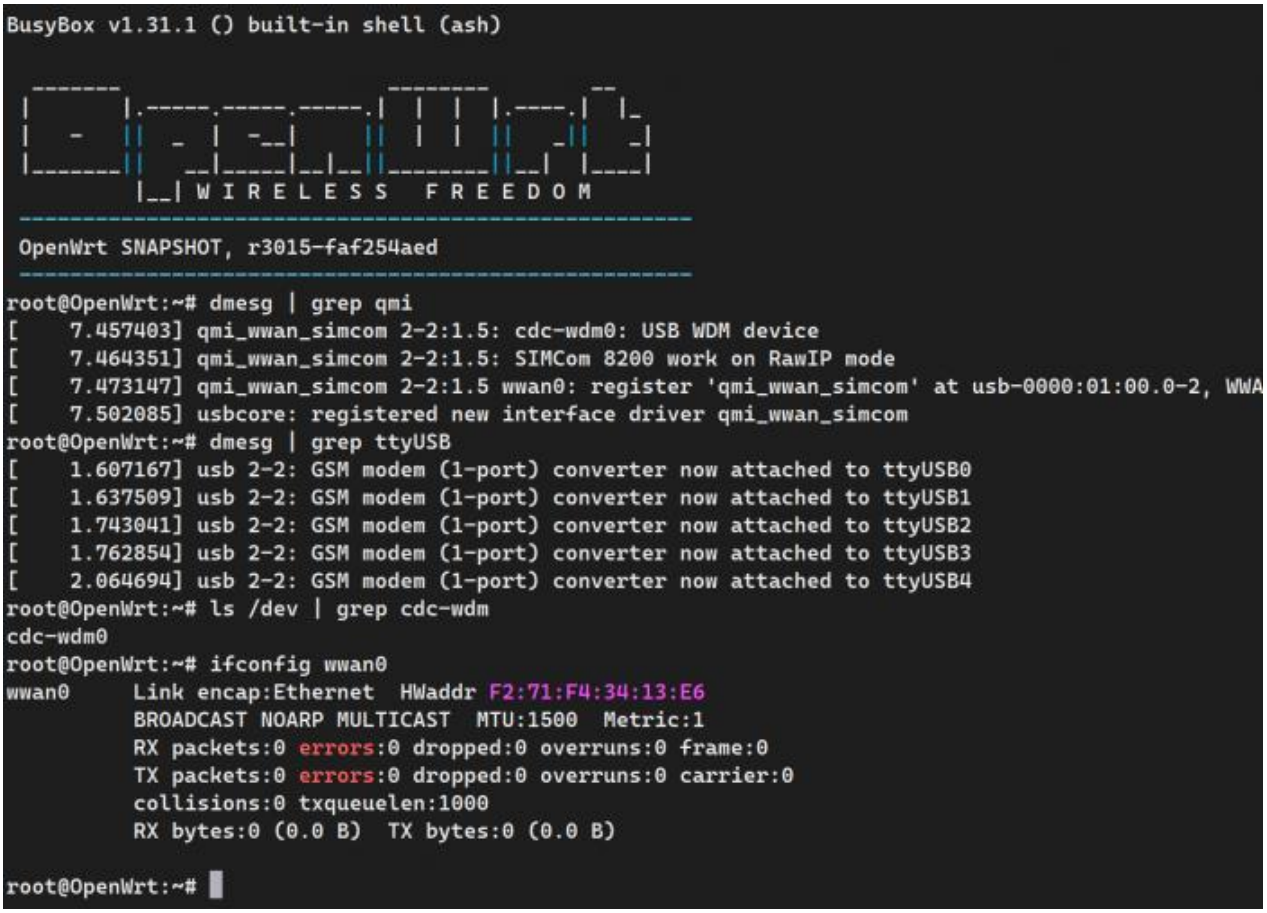
Allow remote hosts to connect to local SSH forwarded ports

- At this point, you can connect to the OpenWrt web management interface or terminal through the IP address of the lan port or wan port.

Check the working status of the driver

Connect to the OpenWrt terminal via SSH, and run the following commands to view the qmi driver, USB device, network port registration, and network port status:

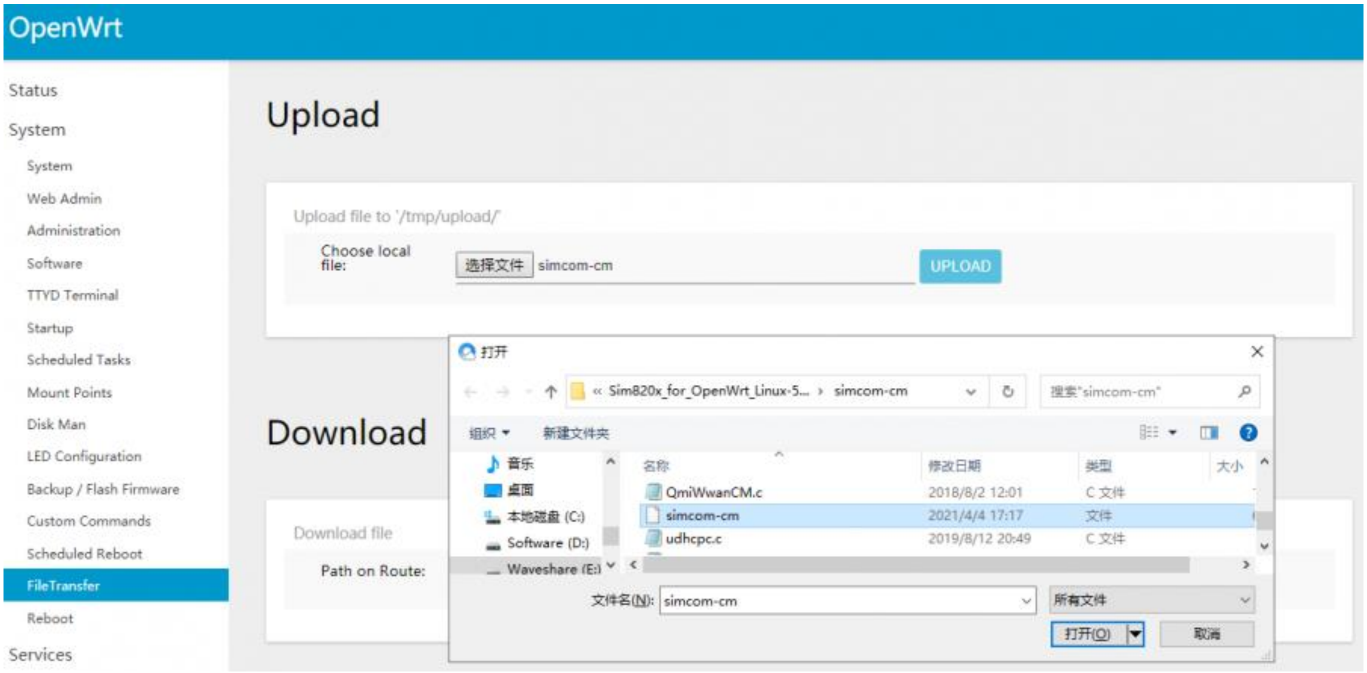
```
dmesg | grep qmi

dmesg | grep ttyUSB

ls /dev | grep cdc-wdm

ifconfig wwan0
```



```
BusyBox v1.31.1 () built-in shell (ash)

  _______                     ________        __
 |       |.-----.-----.-----.|  |  |  |.----.|  |_
 |   -   ||  _  |  -__|     ||  |  |  ||   _||   _|
 |_______||   __|_____|__|__||________||__|  |____|
          |__| W I R E L E S S   F R E E D O M
 -----------------------------------------------------
 OpenWrt SNAPSHOT, r3015-faf254aed
 -----------------------------------------------------
root@OpenWrt:~# dmesg | grep qmi
[    7.457403] qmi_wwan_simcom 2-2:1.5: cdc-wdm0: USB WDM device
[    7.464351] qmi_wwan_simcom 2-2:1.5: SIMCom 8200 work on RawIP mode
[    7.473147] qmi_wwan_simcom 2-2:1.5 wwan0: register 'qmi_wwan_simcom' at usb-0000:01:00.0-2, WWA
[    7.502085] usbcore: registered new interface driver qmi_wwan_simcom
root@OpenWrt:~# dmesg | grep ttyUSB
[    1.607167] usb 2-2: GSM modem (1-port) converter now attached to ttyUSB0
[    1.637509] usb 2-2: GSM modem (1-port) converter now attached to ttyUSB1
[    1.743041] usb 2-2: GSM modem (1-port) converter now attached to ttyUSB2
[    1.762854] usb 2-2: GSM modem (1-port) converter now attached to ttyUSB3
[    2.064694] usb 2-2: GSM modem (1-port) converter now attached to ttyUSB4
root@OpenWrt:~# ls /dev | grep cdc-wdm
cdc-wdm0
root@OpenWrt:~# ifconfig wwan0
wwan0     Link encap:Ethernet  HWaddr F2:71:F4:34:13:E6
          BROADCAST NOARP MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

root@OpenWrt:~#
```

Configure networking

- Select System -> FileTransfer, select and upload simcom-cm in the simcom-cm directory of the folder:

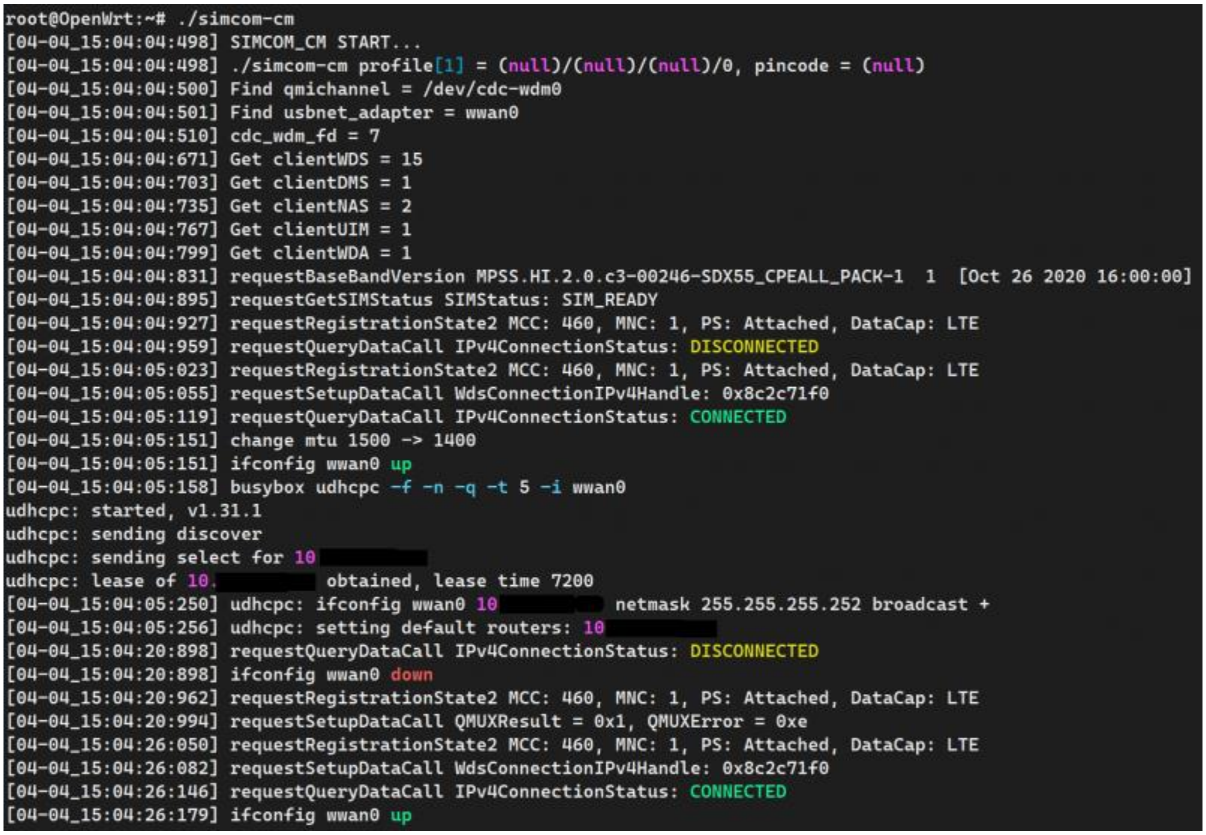


Use the above "File Transfer" to upload, the uploaded program is located in "simcom-cm" under the /tmp/upload/ directory.

Enter the following commands in the terminal:

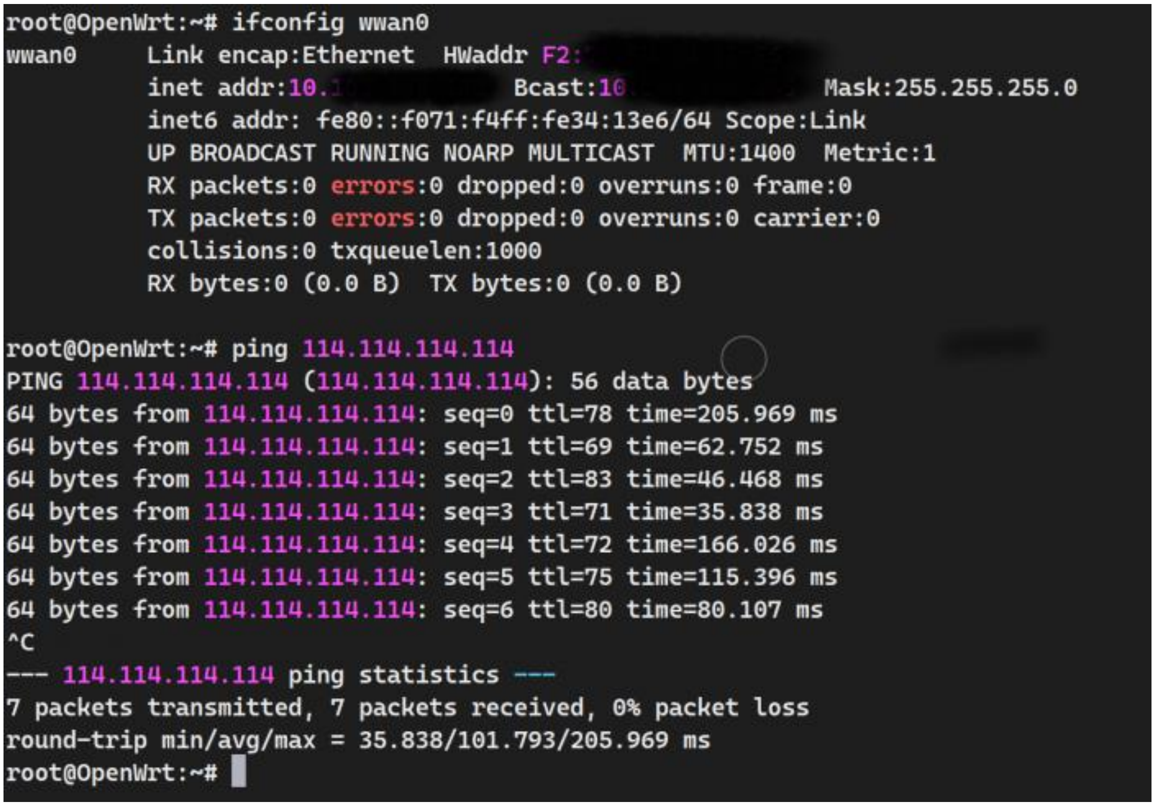
```
cp /tmp/upload/simcom-cm /

chmod a+x simcom-cm

./simcom-cm
```

【Note】: Closing this terminal will cause the networking program to stop, which will cause the network to be disconnected. It is recommended to run in the background.



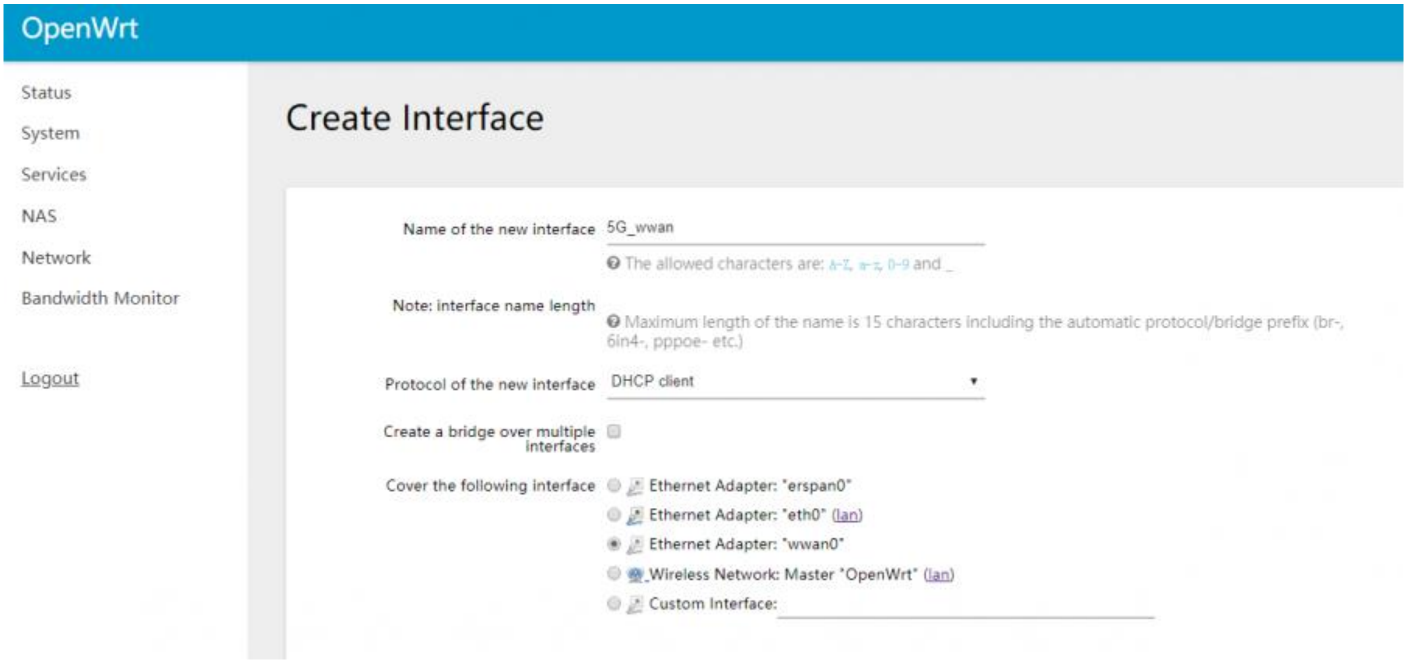
```
root@OpenWrt:~# ./simcom-cm
[04-04_15:04:04:498] SIMCOM_CM START...
[04-04_15:04:04:498] ./simcom-cm profile[1] = (null)/(null)/(null)/0, pincode = (null)
[04-04_15:04:04:500] Find qmichannel = /dev/cdc-wdm0
[04-04_15:04:04:501] Find usbnet_adapter = wwan0
[04-04_15:04:04:510] cdc_wdm_fd = 7
[04-04_15:04:04:671] Get clientWDS = 15
[04-04_15:04:04:703] Get clientDMS = 1
[04-04_15:04:04:735] Get clientNAS = 2
[04-04_15:04:04:767] Get clientUIM = 1
[04-04_15:04:04:799] Get clientWDA = 1
[04-04_15:04:04:831] requestBaseBandVersion MPSS.HI.2.0.c3-00246-SDX55_CPEALL_PACK-1  1  [Oct 26 2020 16:00:00]
[04-04_15:04:04:895] requestGetSIMStatus SIMStatus: SIM_READY
[04-04_15:04:04:927] requestRegistrationState2 MCC: 460, MNC: 1, PS: Attached, DataCap: LTE
[04-04_15:04:04:959] requestQueryDataCall IPv4ConnectionStatus: DISCONNECTED
[04-04_15:04:05:023] requestRegistrationState2 MCC: 460, MNC: 1, PS: Attached, DataCap: LTE
[04-04_15:04:05:055] requestSetupDataCall WdsConnectionIPv4Handle: 0x8c2c71f0
[04-04_15:04:05:119] requestQueryDataCall IPv4ConnectionStatus: CONNECTED
[04-04_15:04:05:151] change mtu 1500 -> 1400
[04-04_15:04:05:151] ifconfig wwan0 up
[04-04_15:04:05:158] busybox udhcpc -f -n -q -t 5 -i wwan0
udhcpc: started, v1.31.1
udhcpc: sending discover
udhcpc: sending select for 10[redacted]
udhcpc: lease of 10.[redacted] obtained, lease time 7200
[04-04_15:04:05:250] udhcpc: ifconfig wwan0 10[redacted] netmask 255.255.255.252 broadcast +
[04-04_15:04:05:256] udhcpc: setting default routers: 10[redacted]
[04-04_15:04:20:898] requestQueryDataCall IPv4ConnectionStatus: DISCONNECTED
[04-04_15:04:20:898] ifconfig wwan0 down
[04-04_15:04:20:962] requestRegistrationState2 MCC: 460, MNC: 1, PS: Attached, DataCap: LTE
[04-04_15:04:20:994] requestSetupDataCall QMUXResult = 0x1, QMUXError = 0xe
[04-04_15:04:26:050] requestRegistrationState2 MCC: 460, MNC: 1, PS: Attached, DataCap: LTE
[04-04_15:04:26:082] requestSetupDataCall WdsConnectionIPv4Handle: 0x8c2c71f0
[04-04_15:04:26:146] requestQueryDataCall IPv4ConnectionStatus: CONNECTED
[04-04_15:04:26:179] ifconfig wwan0 up
```

- At this time, open a new terminal of OpenWrt and enter the command: ifconfig wwan0. You can see that the wwan0 network port has successfully obtained the operator IP and can ping the external network.



```
root@OpenWrt:~# ifconfig wwan0
wwan0     Link encap:Ethernet  HWaddr F2:[redacted]
          inet addr:10.[redacted]  Bcast:10[redacted]  Mask:255.255.255.0
          inet6 addr: fe80::f071:f4ff:fe34:13e6/64 Scope:Link
          UP BROADCAST RUNNING NOARP MULTICAST  MTU:1400  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)

root@OpenWrt:~# ping 114.114.114.114
PING 114.114.114.114 (114.114.114.114): 56 data bytes
64 bytes from 114.114.114.114: seq=0 ttl=78 time=205.969 ms
64 bytes from 114.114.114.114: seq=1 ttl=69 time=62.752 ms
64 bytes from 114.114.114.114: seq=2 ttl=83 time=46.468 ms
64 bytes from 114.114.114.114: seq=3 ttl=71 time=35.838 ms
64 bytes from 114.114.114.114: seq=4 ttl=72 time=166.026 ms
64 bytes from 114.114.114.114: seq=5 ttl=75 time=115.396 ms
64 bytes from 114.114.114.114: seq=6 ttl=80 time=80.107 ms
^C
--- 114.114.114.114 ping statistics ---
7 packets transmitted, 7 packets received, 0% packet loss
round-trip min/avg/max = 35.838/101.793/205.969 ms
root@OpenWrt:~#
```

- Enter the Web management interface of OpenWrt, click Network —>Interface —>Create a new interface



Create Interface

Name of the new interface: 5G_wwan

The allowed characters are: A-Z, a-z, 0-9 and _

Note: interface name length

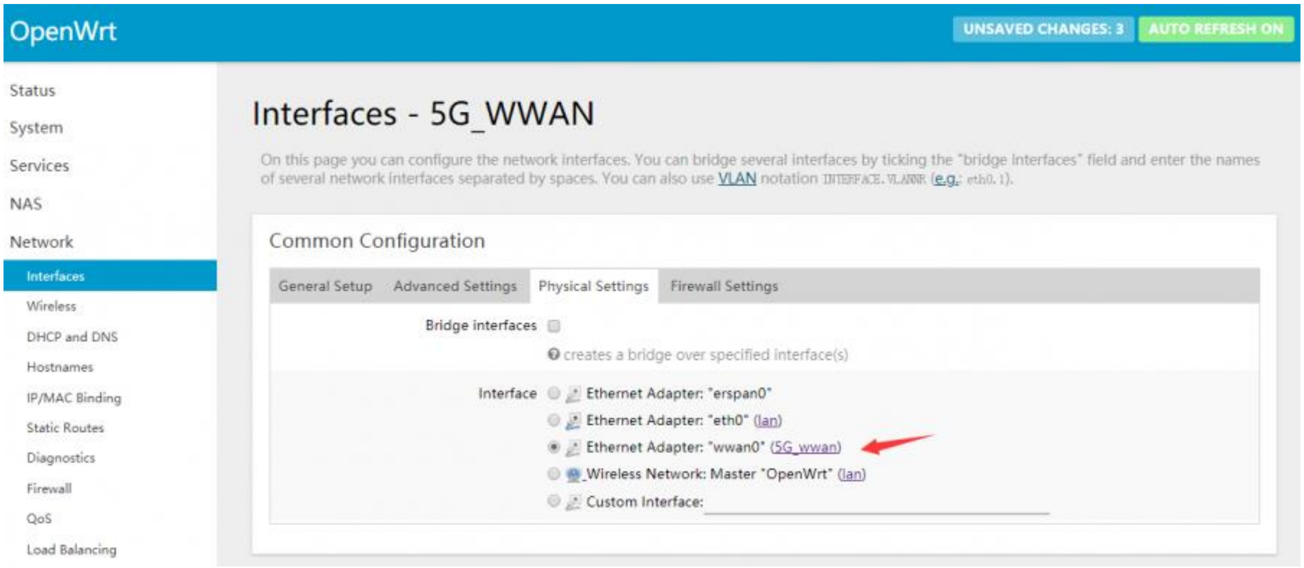
Maximum length of the name is 15 characters including the automatic protocol/bridge prefix (br-, 6in4-, pppoe- etc.)

Protocol of the new interface: DHCP client

Create a bridge over multiple interfaces: □

Cover the following interface:
○ Ethernet Adapter: "erspan0"
○ Ethernet Adapter: "eth0" (lan)
● Ethernet Adapter: "wwan0"
○ Wireless Network: Master "OpenWrt" (lan)
○ Custom Interface:

- Enter the interface as shown in the picture below and confirm that the interface selection in "Physical Settings" is "wwan0".



Interfaces - 5G_WWAN

On this page you can configure the network interfaces. You can bridge several interfaces by ticking the "bridge interfaces" field and enter the names of several network interfaces separated by spaces. You can also use VLAN notation INTERFACE.VLANNR (e.g.: eth0.1).

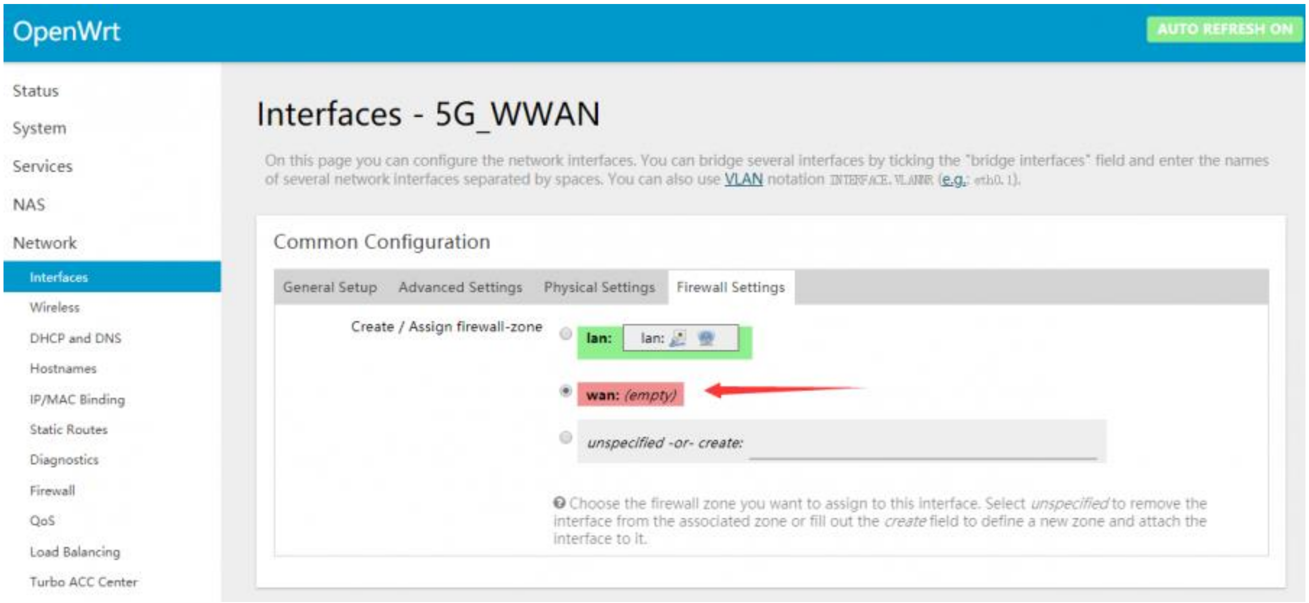
Common Configuration

Bridge interfaces: □

creates a bridge over specified interface(s)

Interface:
○ Ethernet Adapter: "erspan0"
○ Ethernet Adapter: "eth0" (lan)
● Ethernet Adapter: "wwan0" (5G_wwan)
○ Wireless Network: Master "OpenWrt" (lan)
○ Custom Interface:

- Confirm that the interface selection in "Firewall Settings" is "wan"



Interfaces - 5G_WWAN

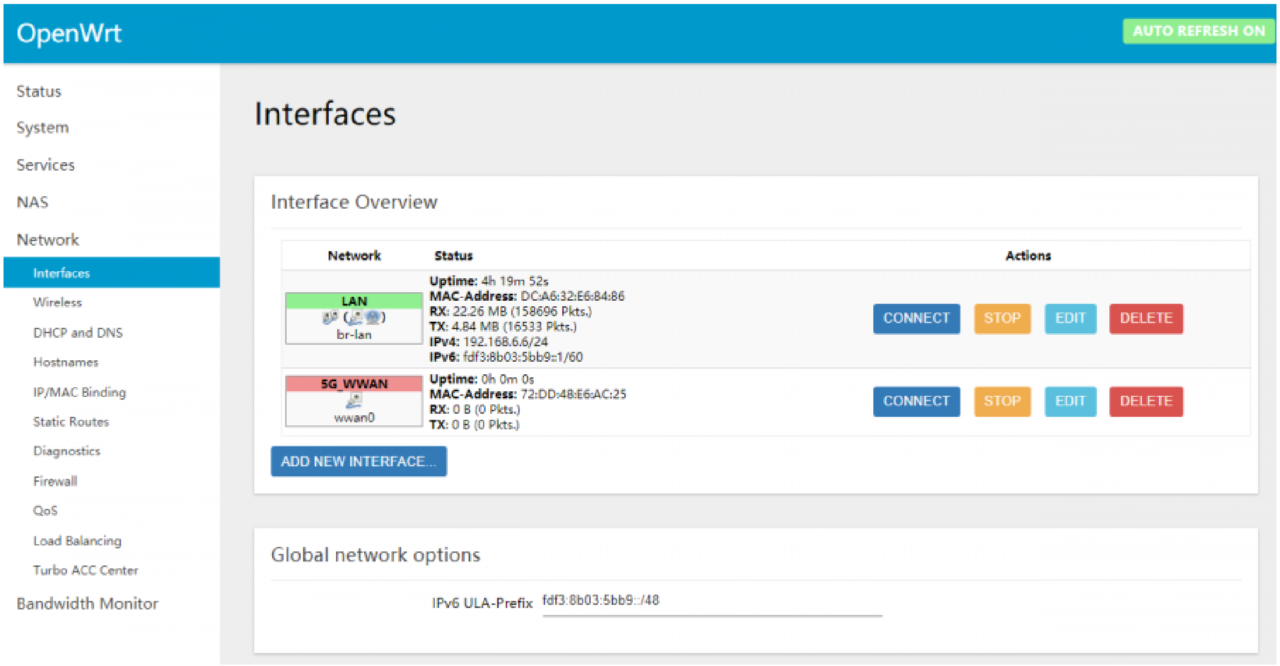
On this page you can configure the network interfaces. You can bridge several interfaces by ticking the "bridge interfaces" field and enter the names of several network interfaces separated by spaces. You can also use VLAN notation INTERFACE.VLANNR (e.g.: eth0.1).

Common Configuration

Create / Assign firewall-zone
○ lan: lan:
● wan: *(empty)*
○ *unspecified -or- create:*

Choose the firewall zone you want to assign to this interface. Select *unspecified* to remove the interface from the associated zone or fill out the *create* field to define a new zone and attach the interface to it.

- Click "Save & Apply" to complete the network port settings, then return to the interface below, network-interface, you can see that the network port has been correctly identified.



Interfaces

Interface Overview

Network	Status	Actions
LAN br-lan	**Uptime:** 4h 19m 52s **MAC-Address:** DC:A6:32:E6:84:86 **RX:** 22.26 MB (158696 Pkts.) **TX:** 4.84 MB (16533 Pkts.) **IPv4:** 192.168.6.6/24 **IPv6:** fdf3:8b03:5bb9::1/60	CONNECT STOP EDIT DELETE
5G_WWAN wwan0	**Uptime:** 0h 0m 0s **MAC-Address:** 72:DD:48:E6:AC:25 **RX:** 0 B (0 Pkts.) **TX:** 0 B (0 Pkts.)	CONNECT STOP EDIT DELETE

Global network options

IPv6 ULA-Prefix fdf3:8b03:5bb9::/48

Then the other devices can be connected to the wireless "OpenWrt" or through the network cable to connect to OpenWrt's own network port for networking.

5G network speed test

In terms of the speed measurement, as the Raspberry Pi comes with a Gigabit Ethernet port, and there are few USB network cards above Gigabit, we use the "SpeedTest For Python" tool to test the speed with the command.
Connect to the terminal of the OpenWrt, enter the command one by one to test:

```
opkg update
opkg install python3
opkg install python3-pip
pip install speedtest_cli
speedtest 或 speedtest_cli
```

Working With Jetson Nano

It is recommended that you use the system image jetson-nano-sd-card-image (updated in October 2020). The Linux kernel version of this system is 4.9.140-tegra. The previous system is 4.4. This tutorial is based on the 4.9 kernels. If there is a difference, please update to the same version as this one, which will minimize the chance of your using it incorrectly.
If you are using other linux systems, please download the driver under SIM8200_OS_Driver\linux and port it according to the documentation under it.

Configuration Required for First Use

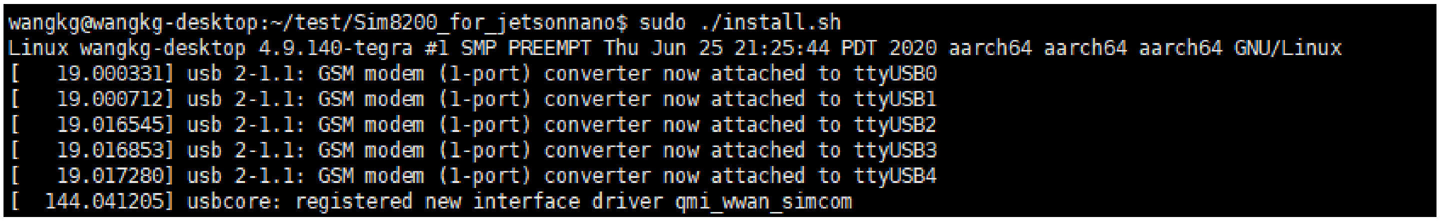
- It's best to copy and paste in case you type the wrong letters:

```
sudo apt-get install p7zip-full
wget https://www.waveshare.com/w/upload/0/07/Sim8200_for_jetsonnano.7z
7z x Sim8200_for_jetsonnano.7z -r -o./Sim8200_for_jetsonnano
sudo chmod 777 -R Sim8200_for_jetsonnano
cd Sim8200_for_jetsonnano
sudo ./install.sh
```

Please do not delete or modify the "options", "qmi_wwan_simcom", "default.script", "install.sh" directory files, otherwise it will affect the loading of the driver.
If there is an error, please confirm whether the system kernel is 4.9.140-tegra, and take a

screenshot of the error message so that engineers can help you analyze and solve the problem.

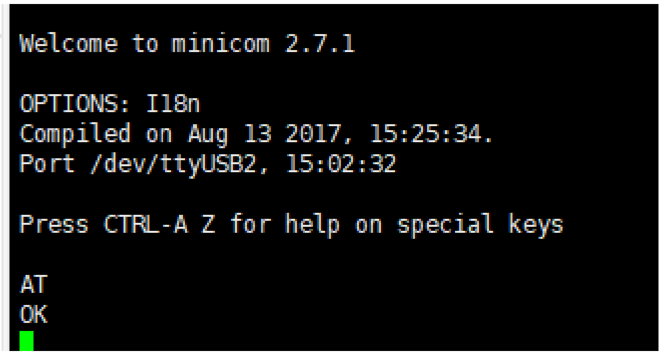
Run "ifconfig -a" to see that WWAN0 has been generated.



```
wangkg@wangkg-desktop:~/test/Sim8200_for_jetsonnano$ sudo ./install.sh
Linux wangkg-desktop 4.9.140-tegra #1 SMP PREEMPT Thu Jun 25 21:25:44 PDT 2020 aarch64 aarch64 aarch64 GNU/Linux
[   19.000331] usb 2-1.1: GSM modem (1-port) converter now attached to ttyUSB0
[   19.000712] usb 2-1.1: GSM modem (1-port) converter now attached to ttyUSB1
[   19.016545] usb 2-1.1: GSM modem (1-port) converter now attached to ttyUSB2
[   19.016853] usb 2-1.1: GSM modem (1-port) converter now attached to ttyUSB3
[   19.017280] usb 2-1.1: GSM modem (1-port) converter now attached to ttyUSB4
[  144.041205] usbcore: registered new interface driver qmi_wwan_simcom
```

AT Test Command

```
sudo apt-get install minicom

sudo minicom -D /dev/ttyUSB2
```

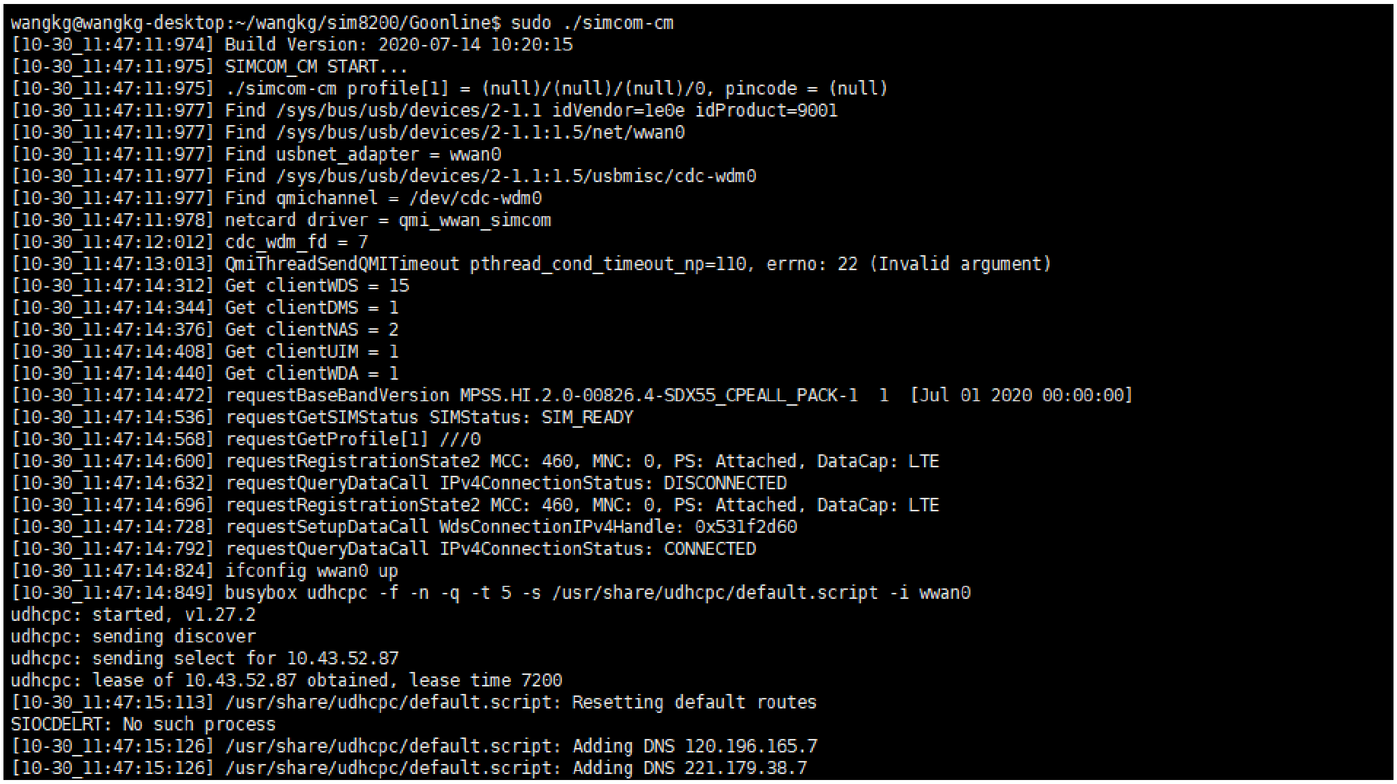


```
Welcome to minicom 2.7.1

OPTIONS: I18n
Compiled on Aug 13 2017, 15:25:34.
Port /dev/ttyUSB2, 15:02:32

Press CTRL-A Z for help on special keys

AT
OK
```

5G Network

```
cd Goonline

make

sudo ./simcom-cm
```

You can successfully see that DNS has been generated at the bottom.



```
wangkg@wangkg-desktop:~/wangkg/sim8200/Goonline$ sudo ./simcom-cm
[10-30_11:47:11:974] Build Version: 2020-07-14 10:20:15
[10-30_11:47:11:975] SIMCOM_CM START...
[10-30_11:47:11:975] ./simcom-cm profile[1] = (null)/(null)/(null)/0, pincode = (null)
[10-30_11:47:11:977] Find /sys/bus/usb/devices/2-1.1 idVendor=1e0e idProduct=9001
[10-30_11:47:11:977] Find /sys/bus/usb/devices/2-1.1:1.5/net/wwan0
[10-30_11:47:11:977] Find usbnet_adapter = wwan0
[10-30_11:47:11:977] Find /sys/bus/usb/devices/2-1.1:1.5/usbmisc/cdc-wdm0
[10-30_11:47:11:977] Find qmichannel = /dev/cdc-wdm0
[10-30_11:47:11:978] netcard driver = qmi_wwan_simcom
[10-30_11:47:12:012] cdc_wdm_fd = 7
[10-30_11:47:13:013] QmiThreadSendQMITimeout pthread_cond_timeout_np=110, errno: 22 (Invalid argument)
[10-30_11:47:14:312] Get clientWDS = 15
[10-30_11:47:14:344] Get clientDMS = 1
[10-30_11:47:14:376] Get clientNAS = 2
[10-30_11:47:14:408] Get clientUIM = 1
[10-30_11:47:14:440] Get clientWDA = 1
[10-30_11:47:14:472] requestBaseBandVersion MPSS.HI.2.0-00826.4-SDX55_CPEALL_PACK-1  1  [Jul 01 2020 00:00:00]
[10-30_11:47:14:536] requestGetSIMStatus SIMStatus: SIM_READY
[10-30_11:47:14:568] requestGetProfile[1] ///0
[10-30_11:47:14:600] requestRegistrationState2 MCC: 460, MNC: 0, PS: Attached, DataCap: LTE
[10-30_11:47:14:632] requestQueryDataCall IPv4ConnectionStatus: DISCONNECTED
[10-30_11:47:14:696] requestRegistrationState2 MCC: 460, MNC: 0, PS: Attached, DataCap: LTE
[10-30_11:47:14:728] requestSetupDataCall WdsConnectionIPv4Handle: 0x531f2d60
[10-30_11:47:14:792] requestQueryDataCall IPv4ConnectionStatus: CONNECTED
[10-30_11:47:14:824] ifconfig wwan0 up
[10-30_11:47:14:849] busybox udhcpc -f -n -q -t 5 -s /usr/share/udhcpc/default.script -i wwan0
udhcpc: started, v1.27.2
udhcpc: sending discover
udhcpc: sending select for 10.43.52.87
udhcpc: lease of 10.43.52.87 obtained, lease time 7200
[10-30_11:47:15:113] /usr/share/udhcpc/default.script: Resetting default routes
SIOCDELRT: No such process
[10-30_11:47:15:126] /usr/share/udhcpc/default.script: Adding DNS 120.196.165.7
[10-30_11:47:15:126] /usr/share/udhcpc/default.script: Adding DNS 221.179.38.7
```

Others

If you need to use the product on other systems, please download the SIM8200_OS_Driver file, and refer to the SIMCOM official documentation to add the driver.

About the Speed

Due to the differences between actual and laboratory conditions, the 5G speed will not be ideal and stable at 100MBPS. There are the following points:

- Base station distance, the closer to the 5G base station, the better the signal and the faster the speed;
- Base station load, the fewer people using it, the faster the speed will be, and the rush hour will be slower;
- Number of base stations: Due to spectrum relationships, the same amount of 4G coverage requires double the number of 5G base stations.
- Operator: You need to confirm whether your 5G card is limited in speed, you can periodically ask the operator to reset your network.

- Indoors is worse than outdoor: building penetration loss, and indoor diffraction loss.
- PS: The current number of base stations still does not have good coverage, and the speed measurement in different locations is not the same.

Resource

Assembly Guide

- Assembly Guide

Program

- RPI Demo
- Jetson Nano Demo
- RPI OpenWrt System

Software

- SIM8200 Driver
- SSCOM Software
- GPS Software
- Xshell software
- Unicode software

Datasheet

- SIM8262E-M2 SPEC
- SIM82XX_SIM83XX Series_AT Command Manual_V1.02.pdf
- SIM83X0-SIM82X0 Series Module Hardware Design Manual
- SIM83X0-SIM83XX Series Module Hardware Design
- SIM82XX_SIM83XX_Series_SSL_Application_Note
- SIM82XX_SIM83XX Series_TCPIP_Application Note_V1.01

- SIM82XX_SIM83XX_Series_GNSS_Application_Note_V1.00.pdf
- SIM82XX_SIM83XX_Series_HTTP(S)_Application_Note_V1.01.pdf
- SIM82XX_SIM83XX_Series_LBS_Application_Note_V1.01.pdf
- SIM82XX_SIM83XX_Series_MQTT(S)_Application_Note_V1.01.pdf
- SIM82XX_SIM83XX_Series_SMS_Application_Note_V1.01.pdf
- SIM82XX_SIM83XX_Series_TTS_Application_Note_V1.01.pdf

Question: Why is my 5G speed not as fast as it says online?

Answer:

First of all, there are slight differences between laboratory data and actual data. In practice, due to insufficient base stations, poor signal coverage, and many 5G coverage holes, the network speed may not reach the highest peak. Under normal circumstances, the 4G upload speed is stable at 20-40Mbps, and the 5G upload speed is stable at 100Mbps (the fewer people, the faster the speed is closer to the base station.)

Question:SIM820X-M2 5G HAT does not recognize the SIM card, what should I do?

Answer:

- The default is to use SIM card 1, it may be switched to SIM card 2, please use the following command to switch to SIM card 1:

```
AT+SMSIMCFG=1,1
```

- Or directly switch to SIM card 2:

```
AT+SMSIMCFG=1,2
```

- If it still doesn't work, consider that the airplane mode is turned on, and use the following command to turn off the airplane mode:

```
AT+CFUN=1
```

Question: What should I do if the SIM820X module NDIS dial fails and the driver cannot be installed?

Answer:

*It is recommended to use RNDIS Dial-up

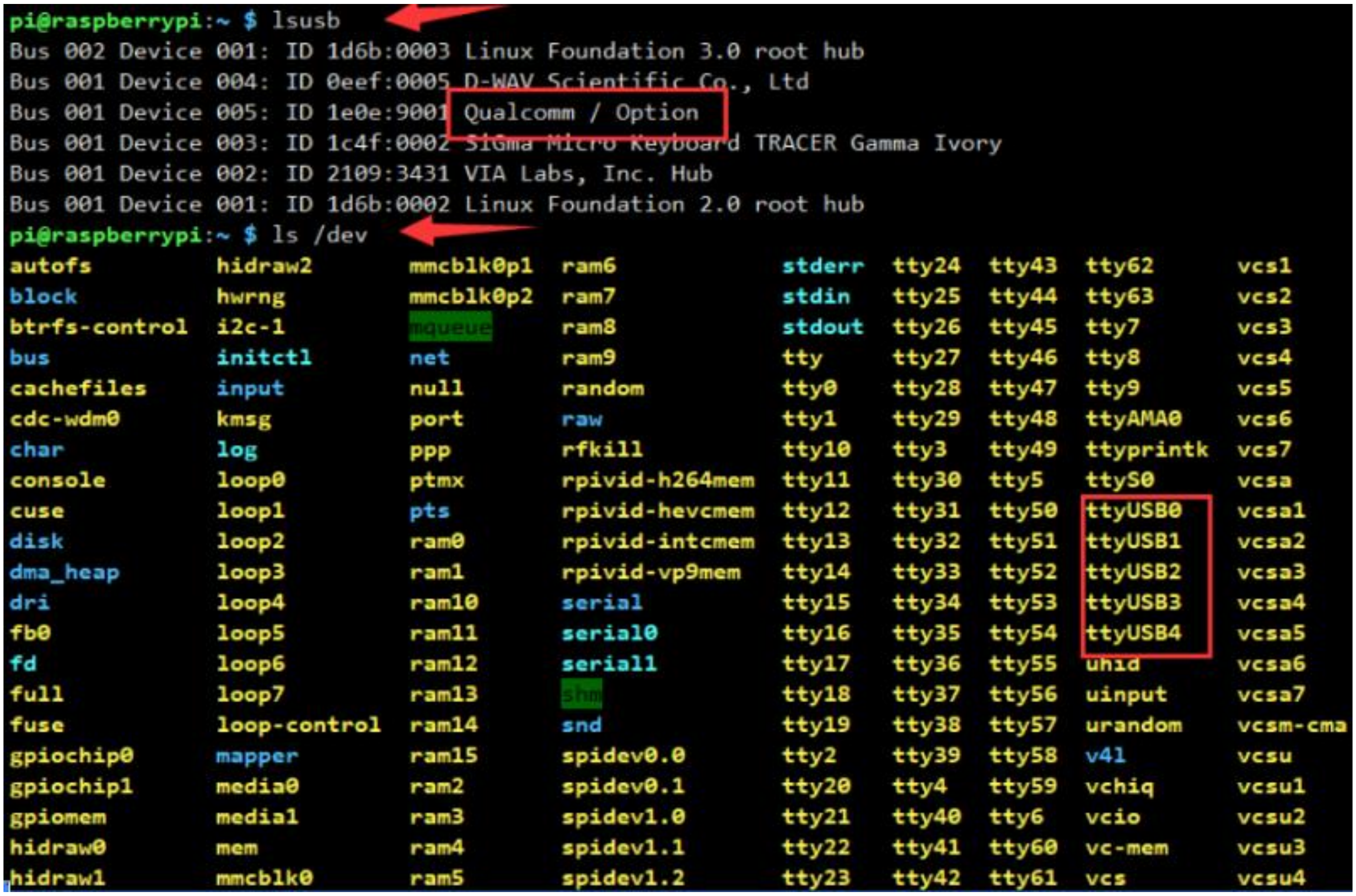
- You can program the newest Raspbian system and reconfigure NDIS dial-up.
- Also, you can use the pre-configured image.

Question: How can I test whether the 4G/5G module is connected to the device successfully?

Answer:

Use the following commands to check whether the SIM820X and Raspberry Pi are well connected. If there is no problem, you can see the Qualcomm device and ttyUSB0-USB4:

```
lsusb
ls /dev
```



```
pi@raspberrypi:~ $ lsusb
Bus 002 Device 001: ID 1d6b:0003 Linux Foundation 3.0 root hub
Bus 001 Device 004: ID 0eef:0005 D-WAV Scientific Co., Ltd
Bus 001 Device 005: ID 1e0e:9001 Qualcomm / Option
Bus 001 Device 003: ID 1c4f:0002 SIGma Micro Keyboard TRACER Gamma Ivory
Bus 001 Device 002: ID 2109:3431 VIA Labs, Inc. Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
pi@raspberrypi:~ $ ls /dev
autofs         hidraw2       mmcblk0p1  ram6            stderr  tty24  tty43  tty62       vcs1
block          hwrng         mmcblk0p2  ram7            stdin   tty25  tty44  tty63       vcs2
btrfs-control  i2c-1         mqueue     ram8            stdout  tty26  tty45  tty7        vcs3
bus            initctl       net        ram9            tty     tty27  tty46  tty8        vcs4
cachefiles     input         null       random          tty0    tty28  tty47  tty9        vcs5
cdc-wdm0       kmsg          port       raw             tty1    tty29  tty48  ttyAMA0     vcs6
char           log           ppp        rfkill          tty10   tty3   tty49  ttyprintk   vcs7
console        loop0         ptmx       rpivid-h264mem  tty11   tty30  tty5   ttyS0       vcsa
cuse           loop1         pts        rpivid-hevcmem  tty12   tty31  tty50  ttyUSB0     vcsa1
disk           loop2         ram0       rpivid-intcmem  tty13   tty32  tty51  ttyUSB1     vcsa2
dma_heap       loop3         ram1       rpivid-vp9mem   tty14   tty33  tty52  ttyUSB2     vcsa3
dri            loop4         ram10      serial          tty15   tty34  tty53  ttyUSB3     vcsa4
fb0            loop5         ram11      serial0         tty16   tty35  tty54  ttyUSB4     vcsa5
fd             loop6         ram12      serial1         tty17   tty36  tty55  uhid        vcsa6
full           loop7         ram13      shm             tty18   tty37  tty56  uinput      vcsa7
fuse           loop-control  ram14      snd             tty19   tty38  tty57  urandom     vcsm-cma
gpiochip0      mapper        ram15      spidev0.0       tty2    tty39  tty58  v4l         vcsu
gpiochip1      media0        ram2       spidev0.1       tty20   tty4   tty59  vchiq       vcsu1
gpiomem        media1        ram3       spidev1.0       tty21   tty40  tty6   vcio        vcsu2
hidraw0        mem           ram4       spidev1.1       tty22   tty41  tty60  vc-mem      vcsu3
hidraw1        mmcblk0       ram5       spidev1.2       tty23   tty42  tty61  vcs         vcsu4
```

Question: Why does the NET light not flash after powering on for a period of time and cannot access the Internet?

Answer:

In this case, the network connection may not be successful. Follow the steps below to troubleshoot:

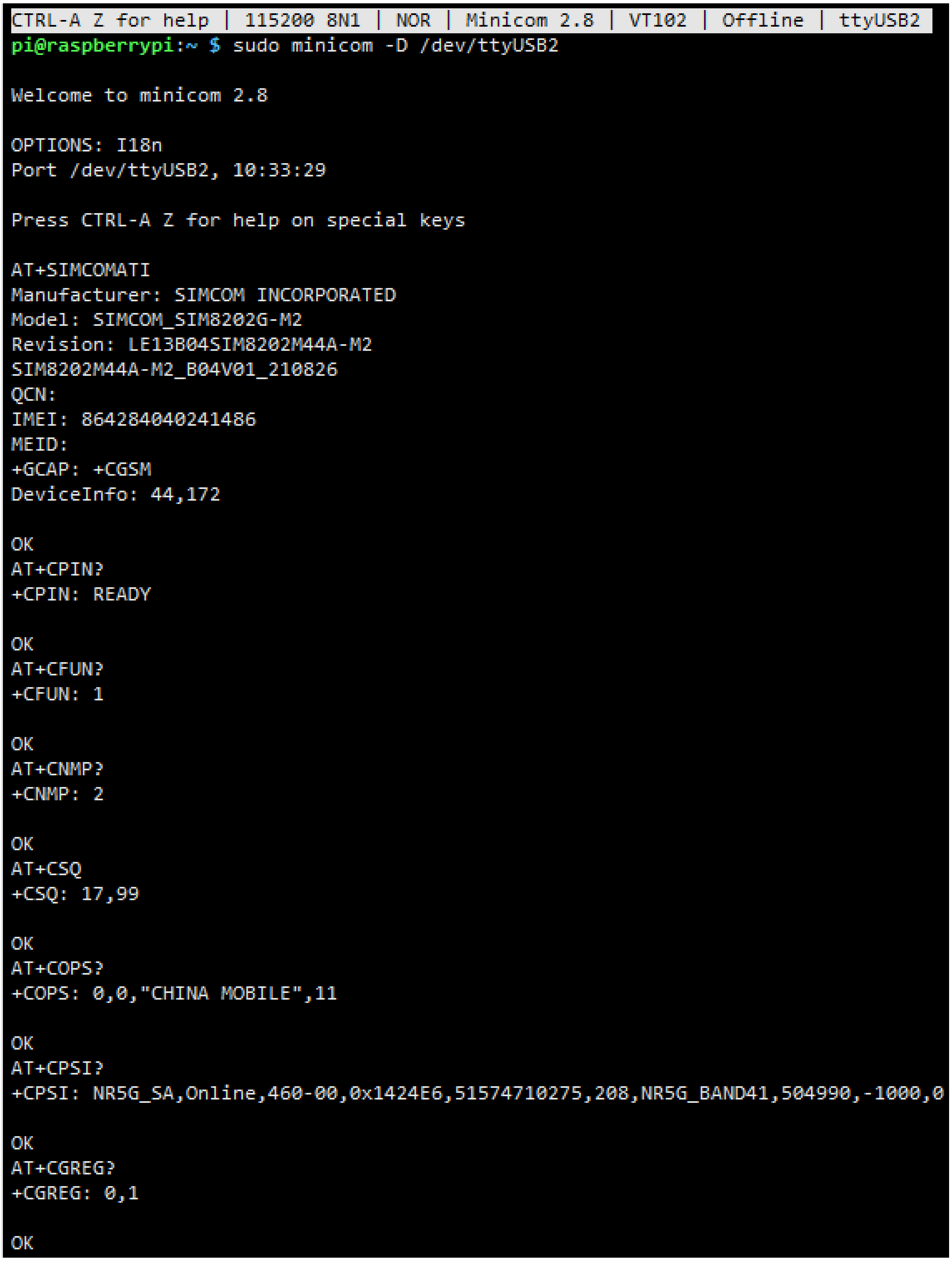
1. First confirm the hardware problem:

- Check whether the connected SIM card can communicate and surf the Internet normally on mobile phones and other devices:
- Check whether the antenna is connected properly;
- Check that the NET light is flashing, and it is always on, indicating that it is not registered to the network

2. After confirming that the hardware is OK, you can use the following command (AT LOG) to further confirm the network situation:

```
*Check the firmware version information: AT+SIMCOMATI
```

```
*Check if the sim card is in good contact: AT+CPIN?

*Turn the radio on (turn off airplane mode): AT+CFUN=1

*Check that the network mode is set to auto-seek: AT+CNMP=2

*Check the signal quality of the current environment: AT+CSQ

*Check if the registration to the network is successful: AT+CGREG?

*Check internet connection: AT+CPSI?

*Check if the APN corresponds to the operator of the SIM card: AT+CGDCONT?

*Check operator access: AT+COPS?
```



```
CTRL-A Z for help | 115200 8N1 | NOR | Minicom 2.8 | VT102 | Offline | ttyUSB2
pi@raspberrypi:~ $ sudo minicom -D /dev/ttyUSB2

Welcome to minicom 2.8

OPTIONS: I18n
Port /dev/ttyUSB2, 10:33:29

Press CTRL-A Z for help on special keys

AT+SIMCOMATI
Manufacturer: SIMCOM INCORPORATED
Model: SIMCOM_SIM8202G-M2
Revision: LE13B04SIM8202M44A-M2
SIM8202M44A-M2_B04V01_210826
QCN:
IMEI: 864284040241486
MEID:
+GCAP: +CGSM
DeviceInfo: 44,172

OK
AT+CPIN?
+CPIN: READY

OK
AT+CFUN?
+CFUN: 1

OK
AT+CNMP?
+CNMP: 2

OK
AT+CSQ
+CSQ: 17,99

OK
AT+COPS?
+COPS: 0,0,"CHINA MOBILE",11

OK
AT+CPSI?
+CPSI: NR5G_SA,Online,460-00,0x1424E6,51574710275,208,NR5G_BAND41,504990,-1000,0

OK
AT+CGREG?
+CGREG: 0,1

OK
```

Question:SIM820X has no network after fixed 5G, what should I do if the 5G network speed is very slow?

Answer:

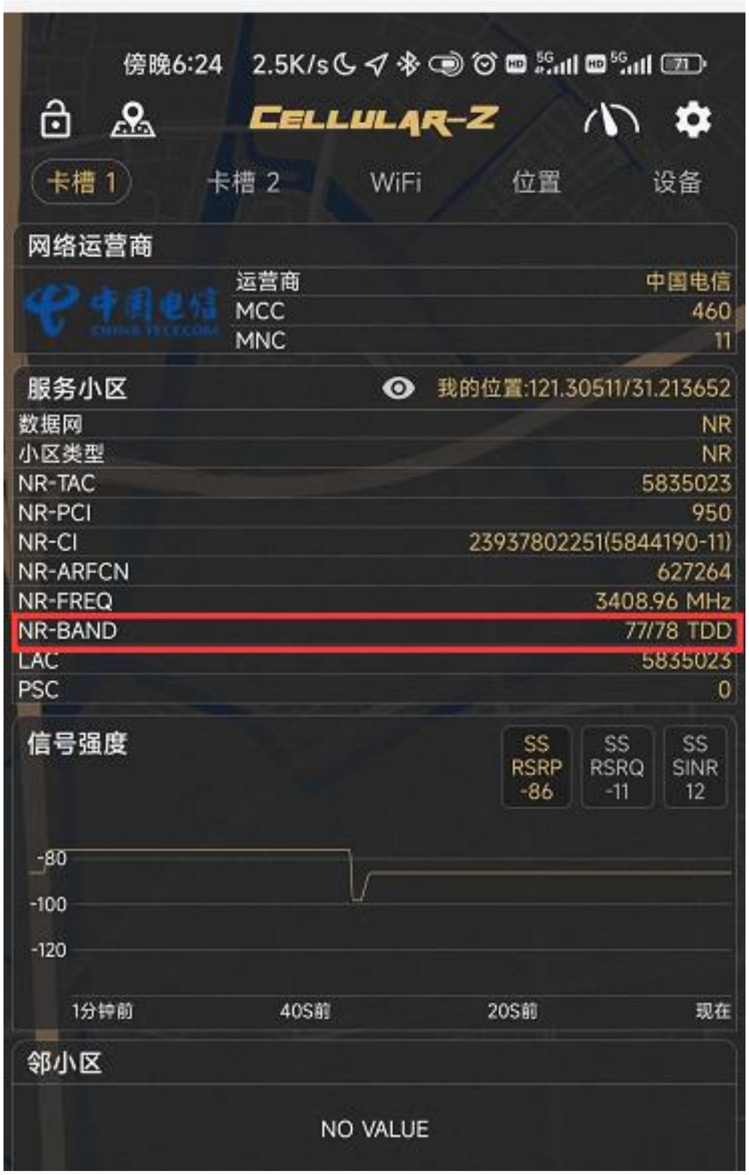
Confirm that the SIM card connected to the SIM820X can use 5G in mobile phones and other devices through methods such as speed measurement:

The fact that the mobile phone can obtain 5G means that there is a 5G signal in the local area, but the mobile phone supports much more 5G frequency bands than the module, and the 5G module may not necessarily be able to register to the 5G network, or ask the local operator for the 5G frequency, and the 5G module supports Under the benchmarking; or go to the local city center (where there are 5G NR cells) to test!

- After confirming that the SIM card is normal and the local 5G network is covered, follow the steps below to fix 5G (to increase the network speed).

```
AT+CNBP? //(Can copy the frequency band to Notepad) Backup frequency band

AT+CNBP = 0x100600000FC00000, 0x000000000000000046004600001e2000908dd, 0x000000000
00000000000000000000070420081a00808d7, 0x0000000000000000000000 //

AT+CNMP=71 // Fixed 5G
```

- The 5G operator's SIM card frequency band should cover the SIM820X frequency band; sometimes the 4G network is queried, and the speed has reached 5G. You can directly measure the speed to confirm that the 5G network speed is available.
- To confirm whether the test environment has 5G network coverage, you can install Cellular-Z on a 5G mobile phone (all major app stores can download it) to view the detailed network information:



网络运营商

运营商	中国电信
MCC	460
MNC	11

服务小区 我的位置:121.30511/31.213652

数据网	NR
小区类型	NR
NR-TAC	5835023
NR-PCI	950
NR-CI	23937802251(5844190-11)
NR-ARFCN	627264
NR-FREQ	3408.96 MHz
NR-BAND	77/78 TDD
LAC	5835023
PSC	0

信号强度

SS RSRP	SS RSRQ	SS SINR
-86	-11	12

邻小区

NO VALUE

- Some IoT cards (pure traffic cards) can obtain a 5G network on mobile phones, but cannot obtain a 5G network on SIM820X; please replace ordinary mobile phone cards that can make calls and send text messages to test.

Question: How to fix a specific frequency band?

Answer:

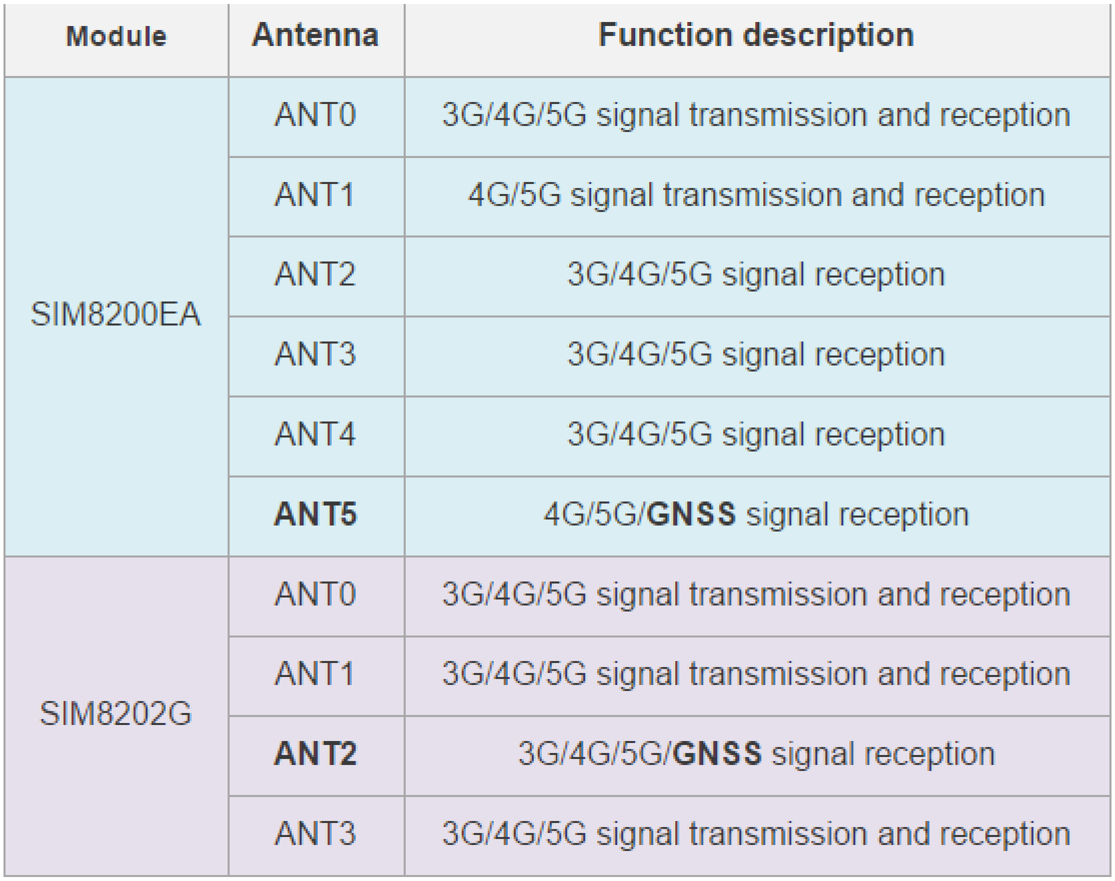
Specific frequency bands can be fixed by the following commands, here fixed at 41 and 78:

```
AT+CSYSSEL="nr5g_band",41:78
```

Question: The SIM820X has so many antenna ports, do they have to be connected? What is the function?

Answer:

The antenna must be connected, and the corresponding functions are as follows:



Module	Antenna	Function description
SIM8200EA	ANT0	3G/4G/5G signal transmission and reception
	ANT1	4G/5G signal transmission and reception
	ANT2	3G/4G/5G signal reception
	ANT3	3G/4G/5G signal reception
	ANT4	3G/4G/5G signal reception
	ANT5	4G/5G/**GNSS** signal reception
SIM8202G	ANT0	3G/4G/5G signal transmission and reception
	ANT1	3G/4G/5G signal transmission and reception
	ANT2	3G/4G/5G/**GNSS** signal reception
	ANT3	3G/4G/5G signal transmission and reception

Question: What should I do if I can't receive the GPS signal and get the location information?

Answer:

Plug the GPS antenna into the GNSS antenna socket (SIM8200EA is ANT5; SIM8202G is ANT2), and place the receiver in the open air (note that it cannot be tested in rainy weather). It takes about 1 minute to receive the positioning signal after power-on.

Question: Is SPI or USB for data transmission?

Answer:

You can use USB to transfer data, and runs "sudo ./install" to load the USB driver. We can use the USB connector of Raspberry Pi 3B/4B to communicate, also you can take the same steps to operate in windows for USB loading. Without a USB driver, it will be identified as an unknown device.

Question: Does the Raspberry Pi control occupy pins?

Answer:

Not occupied by default.

Question: Why does the Raspberry Pi fail to load when running sudo ./install?

Answer:

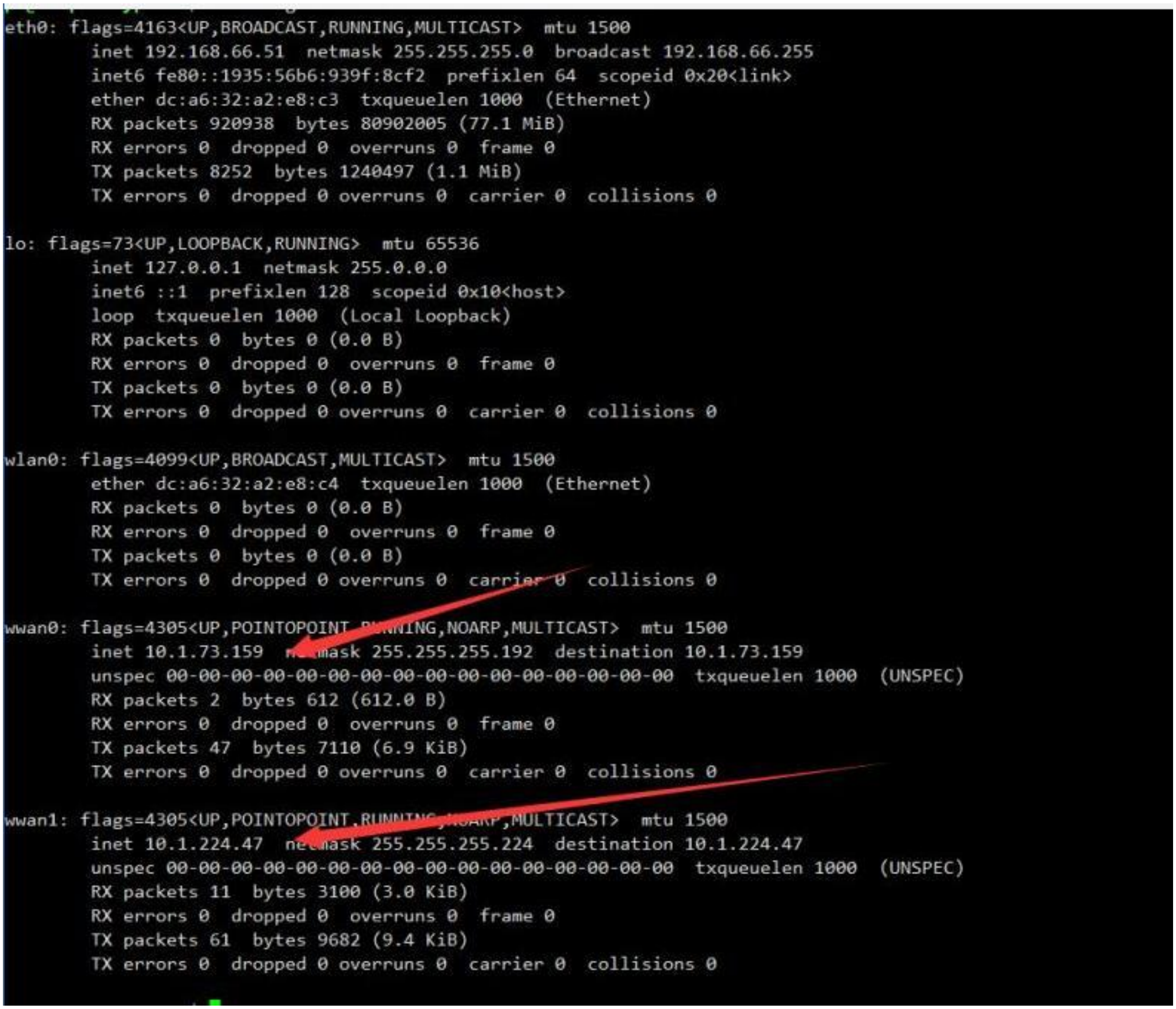
Make sure that your system kernel is above 5.4. Do not use sudo update to upgrade the Raspberry Pi to the latest version, otherwise, the kernel version will be upgraded to a version higher than the current firmware and it will not be recognized.

Question: Can I use multiple 5G modules at the same time?

Answer:

Yes, connect to different USB ports of different hosts, take Raspberry Pi as an example, and the instructions for dialing up the Internet at the same time are as follows:

```
sudo ./simcom-cm -i wwan0
sudo ./simcom-cm -i wwan1
```



```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.66.51  netmask 255.255.255.0  broadcast 192.168.66.255
        inet6 fe80::1935:56b6:939f:8cf2  prefixlen 64  scopeid 0x20<link>
        ether dc:a6:32:a2:e8:c3  txqueuelen 1000  (Ethernet)
        RX packets 920938  bytes 80902005 (77.1 MiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 8252  bytes 1240497 (1.1 MiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (Local Loopback)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wlan0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether dc:a6:32:a2:e8:c4  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wwan0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 10.1.73.159  netmask 255.255.255.192  destination 10.1.73.159
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 1000  (UNSPEC)
        RX packets 2  bytes 612 (612.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 47  bytes 7110 (6.9 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

wwan1: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST>  mtu 1500
        inet 10.1.224.47  netmask 255.255.255.224  destination 10.1.224.47
        unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00  txqueuelen 1000  (UNSPEC)
        RX packets 11  bytes 3100 (3.0 KiB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 61  bytes 9682 (9.4 KiB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

Question: Is it possible to save text messages to a place other than the SIM card? The following commands can be used to save the SMS to the module memory:

Answer:

```
AT+CPMS="MT"
```

Question: Some SIM card operators require the VOLTE function to make calls. How to enable or disable the VOLTE function?

Answer:

The VOLTE function can be turned on with the following command:

```
at+voltesetting=1

 at+cnv=/nv/item_files/modem/mmode/ue_usage_setting,1,01,1
```

The VOLTE function can be turned off with the following command:

```
at+voltesetting=0

at+cnv=/nv/item_files/modem/mmode/ue_usage_setting,0,01,1
```