

SMA-3568 Debian11 Software Usage Guide

V1.0

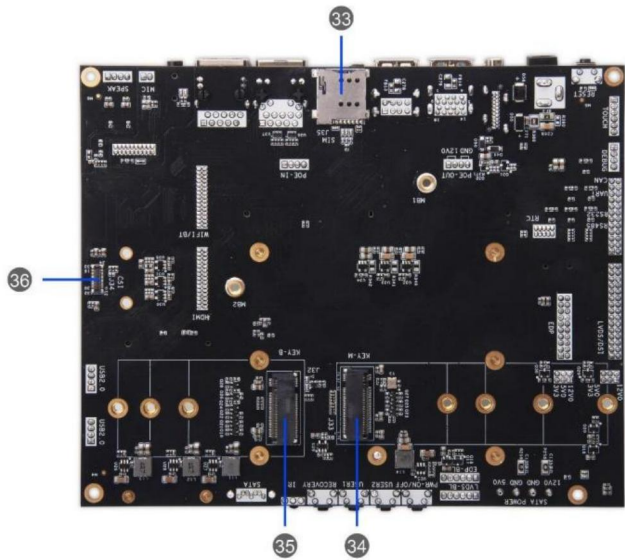
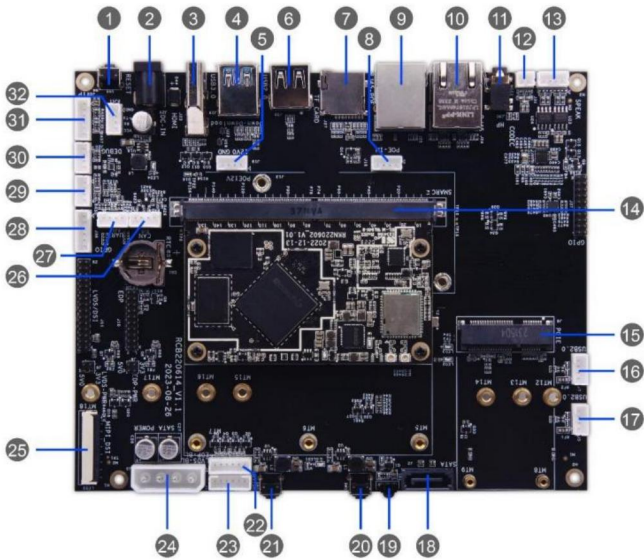
Catalog

1. Product interface definition:	4
2. SMA-3568 Debian11 OS	4
3. Function Description	5
3.1 How to access the OS	5
3.2 Network Function	8
3.2.1 Wired Network	8
3.2.2 WIFI	9
3.2.3 Bluetooth	10
3.2.4 LTE	11
3.3 CAN	12
3.4 GPIO	15
3.5 UART	16
3.6 I2C	18
3.7 SPI	19
3.8 Audio interface	20
3.8.1 Loudspeaker	20
3.8.2 MIC	21
3.8.3 Headphones	22
3.9 CSI Camera	23
3.10 Screen	24
3.10.1 LVDS	24
3.10.2 MIPI	25
3.10.3 EDP	26
3.11 External Storage Devices	27
3.11.1 U disk	27
3.11.2 TF card	28
3.11.3 SSD	28
3.11.4 SATA	29
3.12 IR Remote Control	30
3.13 RTC	31
3.14 OTG Port	32

REVISION HISTORY

Time	Version	Purpose	Author	Comment
10/4/2025	1.0	Create a document	CM	

1. Product interface definition:

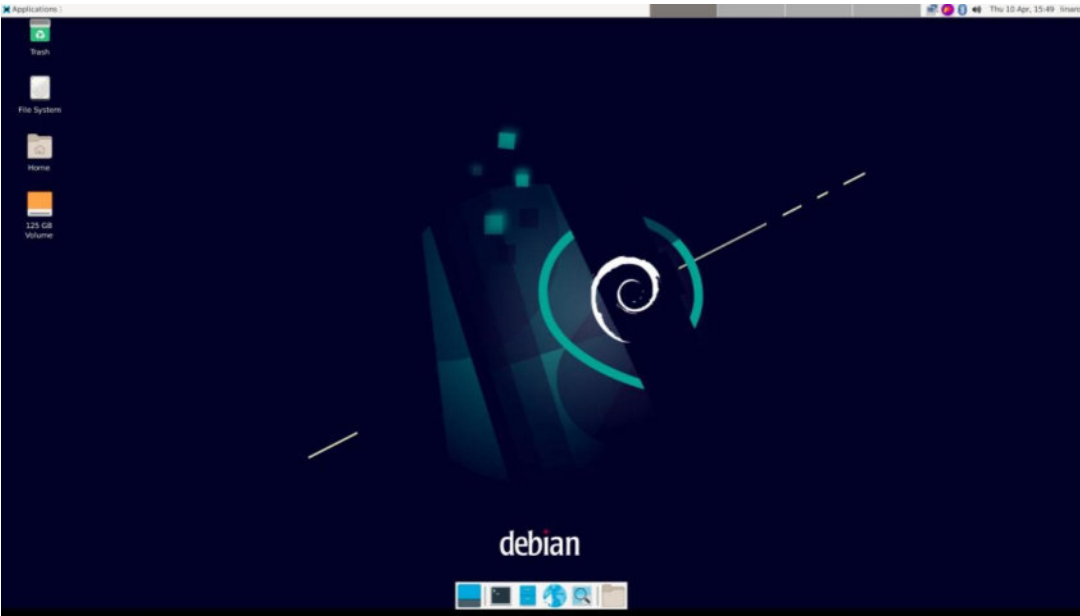


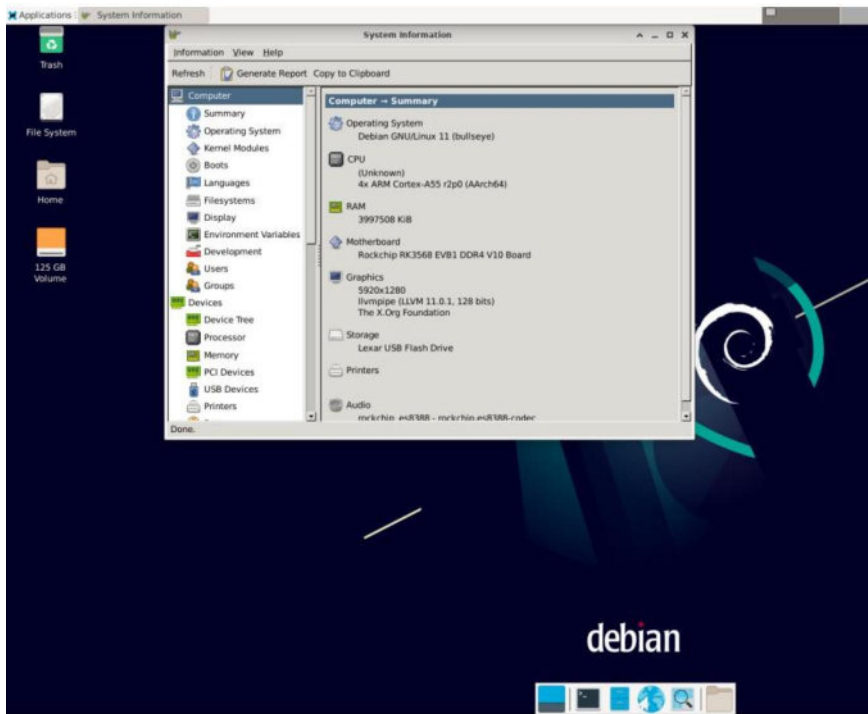
Label	Function	Note
1	Reset Key	
2	DC IN	*1
3	HDMI OUT	*1(Up to 4K P60)
4	USB3.0 Type-A	*1
5	DC IN	1 * 4P Header, pitch 2.0mm (for POE IN or DC IN)
6	USB2.0 Type-A	*1
7	TF Card Slot	*1
8	POE OUT	1 * 4P Header, pitch 2.0mm
9	RJ45_1	1000M Ethernet Port
10	RJ45_2	1000M Ethernet Port
11	3.5mm Slot	For Headphone
12	MIC IN	1 * 4P Header, pitch 2.0mm(for Analog Mic)
13	SPEAKER	1 * 4P Header, pitch 2.0mm(Up to 2*5W)
14	SODIMM SOCKET	MXM 314P 0.5mm H7.8 SMD
15	Mini PCIe	*1 (for 4G LTE Module)
16	USB2.0	1 * 4P Header, pitch 2.0mm
17	USB2.0	1 * 4P Header, pitch 2.0mm
18	SATA Data Slot	*1
19	IR IN	*1

20	UPDATE KEY	*1
21	POWER KEY	*1
22	eDP BACKLIGHT	1 * 6P Header, pitch 2.0mm
23	LVDS BACKLIGHT	1 * 6P Header, pitch 2.0mm
24	SATA POWER	*1
25	MIPI DSI Connector	1 * 40P FPC, pitch 0.5mm
26	CAN	1 * 4P Header, pitch 2.0mm
27	UART	1 * 4P Header, pitch 2.0mm
28	GPIO	1 * 6P Header, pitch 2.0mm
29	UART	1 * 4P Header, pitch 2.0mm
30	UART for DEBUG	1 * 4P Header, pitch 2.0mm
31	I2C for TOUCH	1 * 6P Header, pitch 2.0mm
32	UART	1 * 4P Header, pitch 2.0mm
33	SIM CARD SLOT	*1
34	M.2(KEY-M)	*1
35	M.2(KEY-B)	*1
36	MIPI CSI Connector	1*40P, FPC pitch 0.5mm (MIPI CSI)

2.SMA-3568 Debian11 OS

SMA-3568 Supported OS : Debian GNU/Linux 11 \n \l



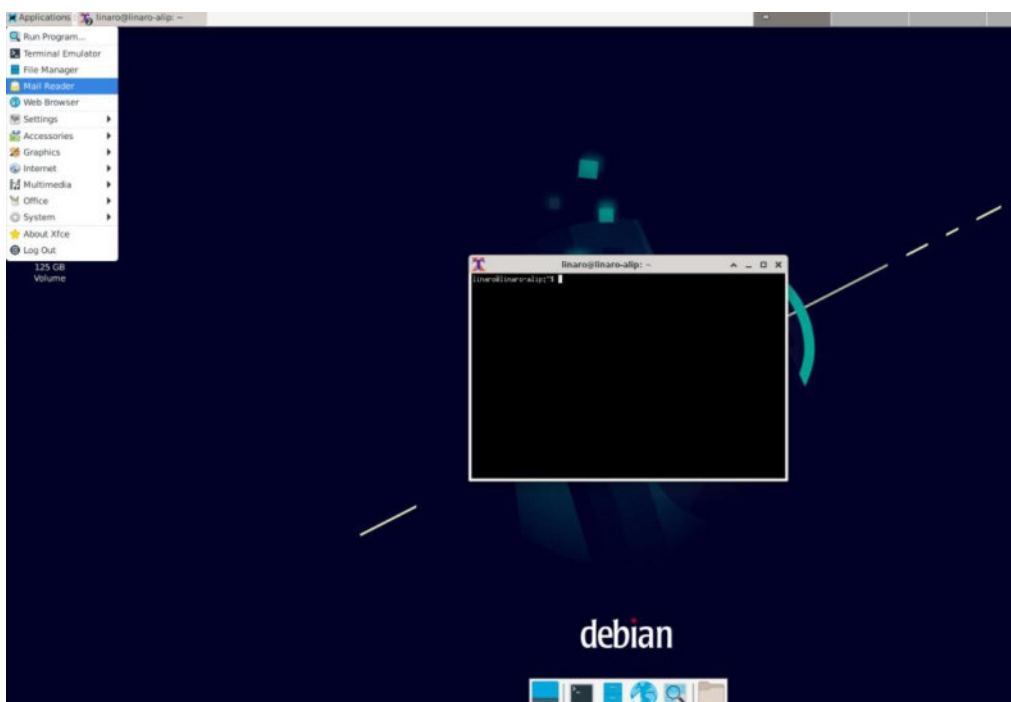


3. Function Description

3.1 How to access the OS

3.1.1 HDMI Display

SMA-3568 support HDMI OUT display, default resolution 1920x1080; It can be connected to the monitor and wait for the system start up, connecting the keyboard, open a terminal via the main menu -> “Application ”-> “Terminal Emulator”.



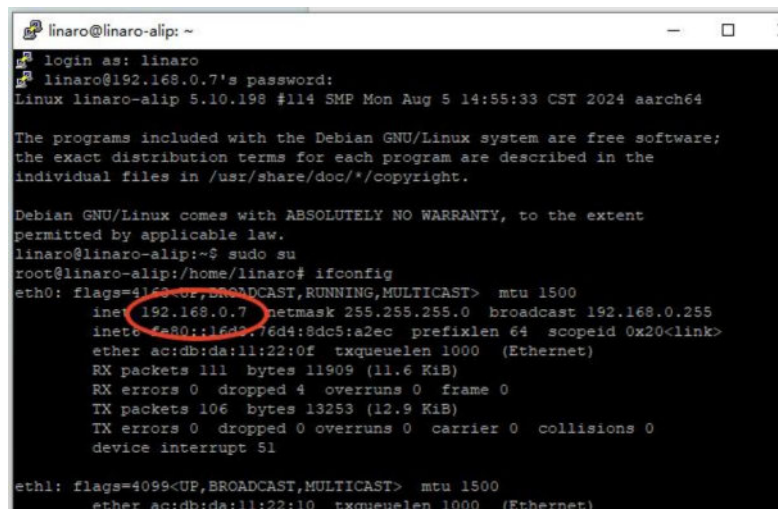
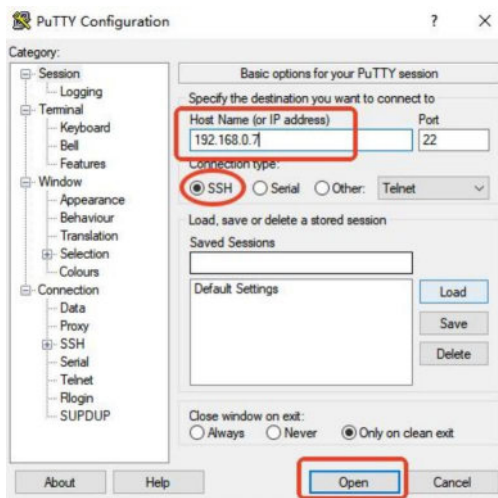
3.1.2 Remote Connection

Connect the PC and SMA-3568 in a local area network, get the IP address and connect via the serial tool (putty.exe) using SSH.

Protocol : SSH2

User name : linaro

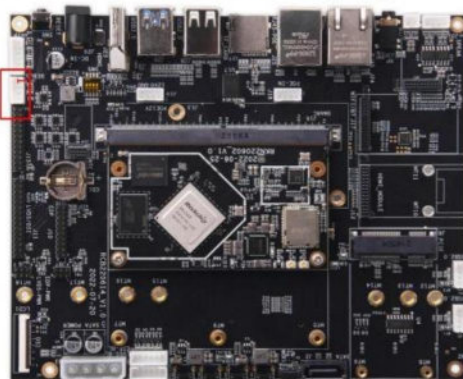
Password : linaro



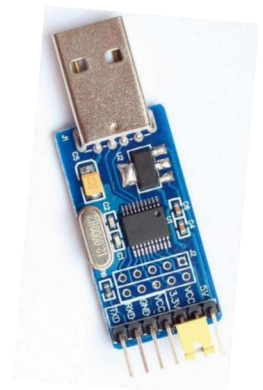
3.1.3 Serial Access

(1) For serial port access through USB to UART, connect the USB port to the computer, and connect the other end to the serial port conversion board.

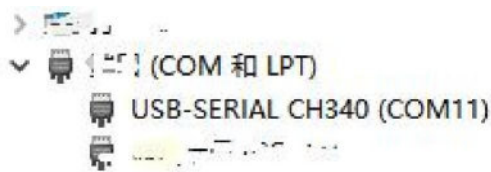
DEBUG UART Connector (J20)



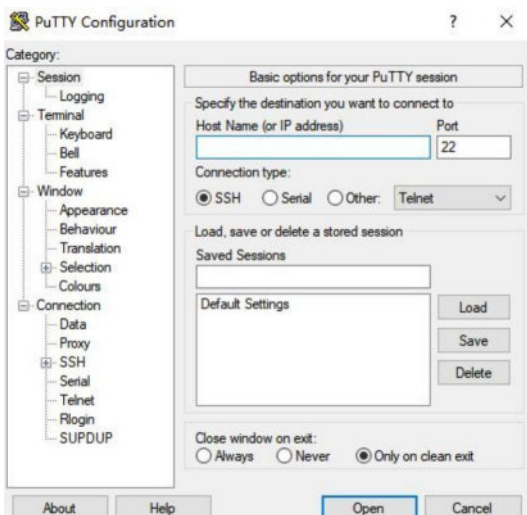
Pin NO	Definition
1	UART2_RX_MO_DEBUG_3V3
2	GND
3	UART2_TX_MO_DEBUG_3V3
4	NC



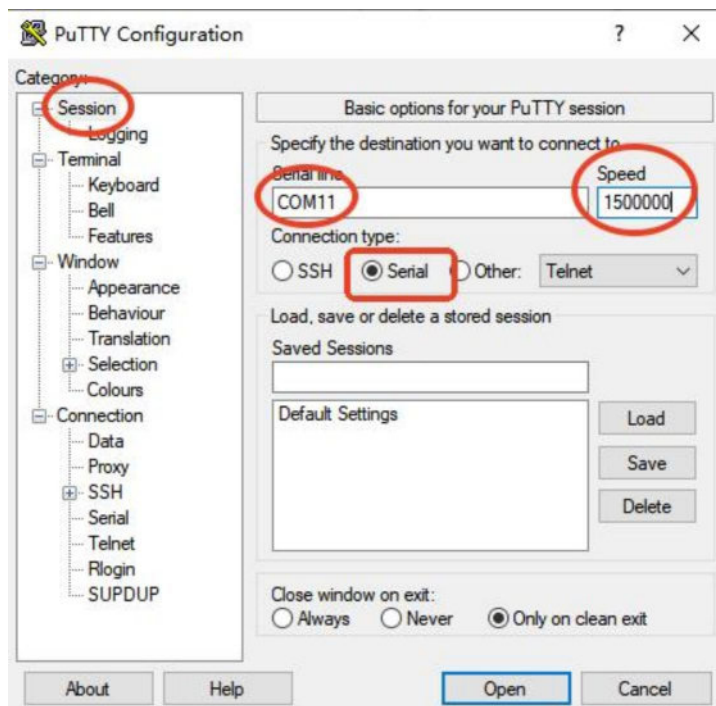
(2) IN the Device Manager of your computer, view the serial port number.



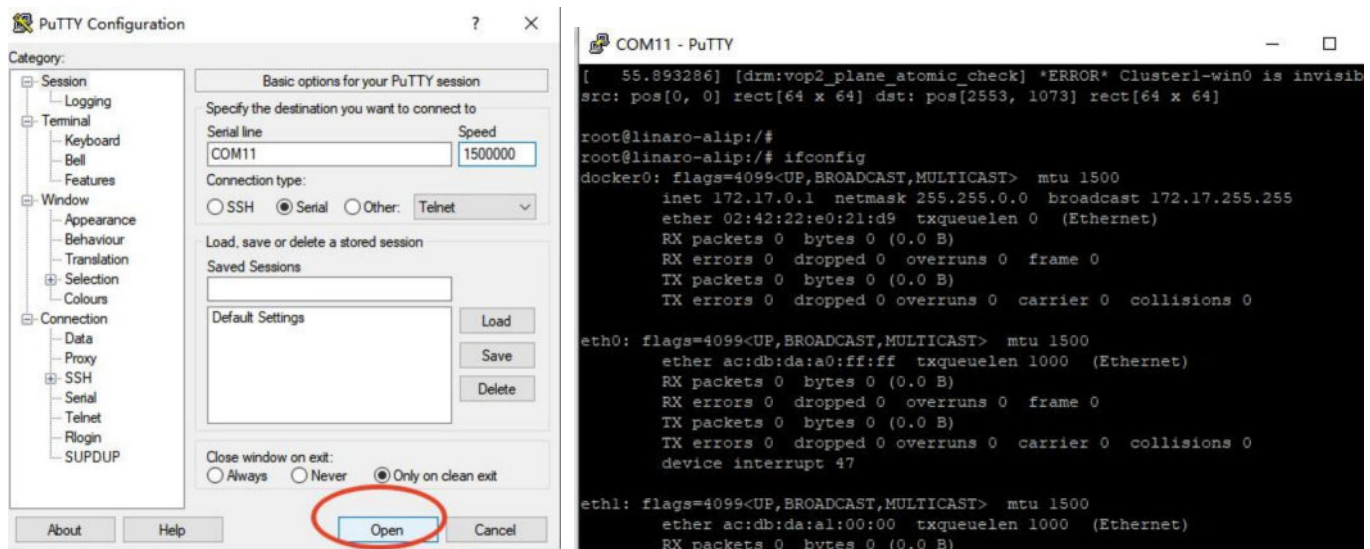
(3) Double-click putty.exe to open the serial port tool.



(4) Click "Session", select the Serial port "serial", after the computer is connected to the DEBUG port, select the new serial port name COMx, and set the corresponding serial port baud rate (1500000).



(5) Click "Open" to start the tool.

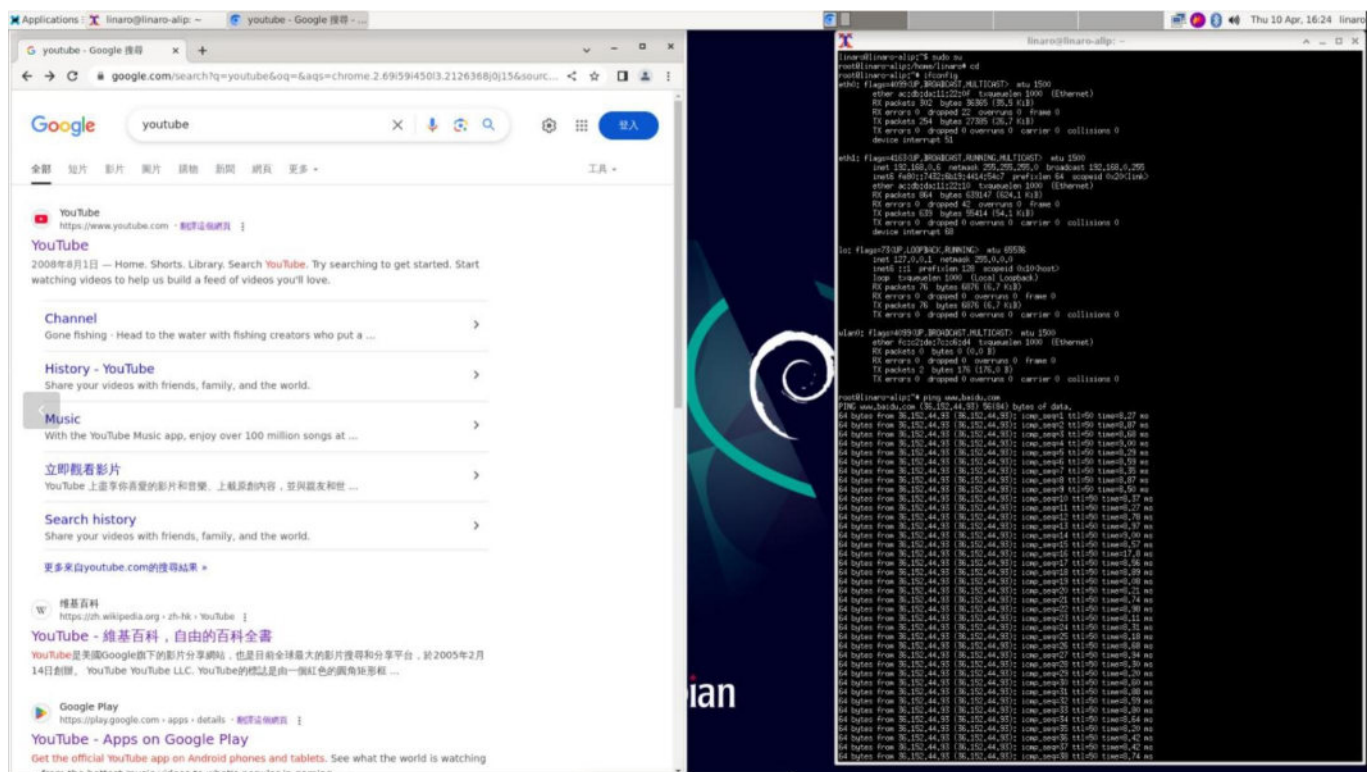


3.2 Network Function

3.2.1 Wired Network

Wired connection: Execute the command to view the obtained IP address, which can be verified through a browser or ping Baidu to verify that the network is functioning properly.

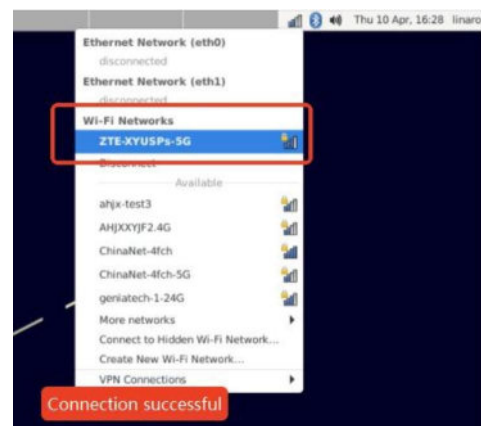
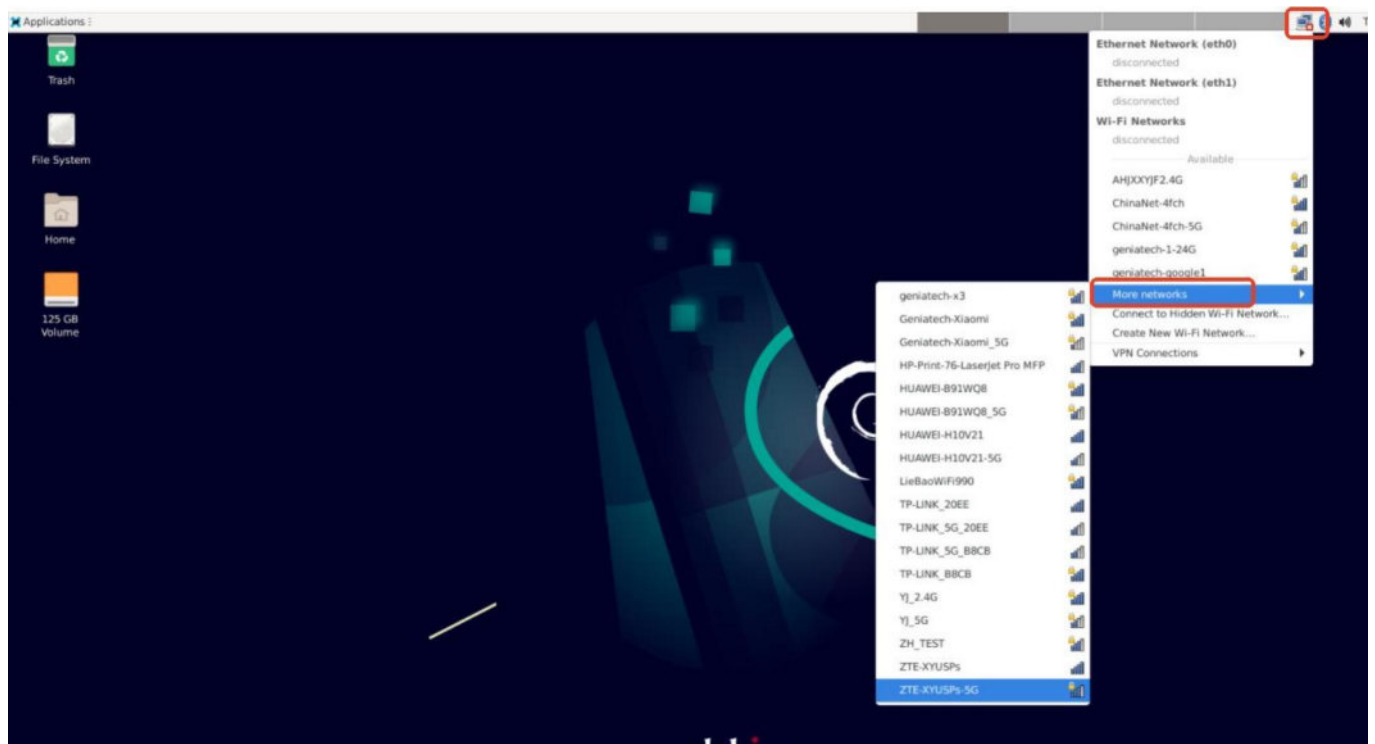
Check the IP address: **ifconfig**

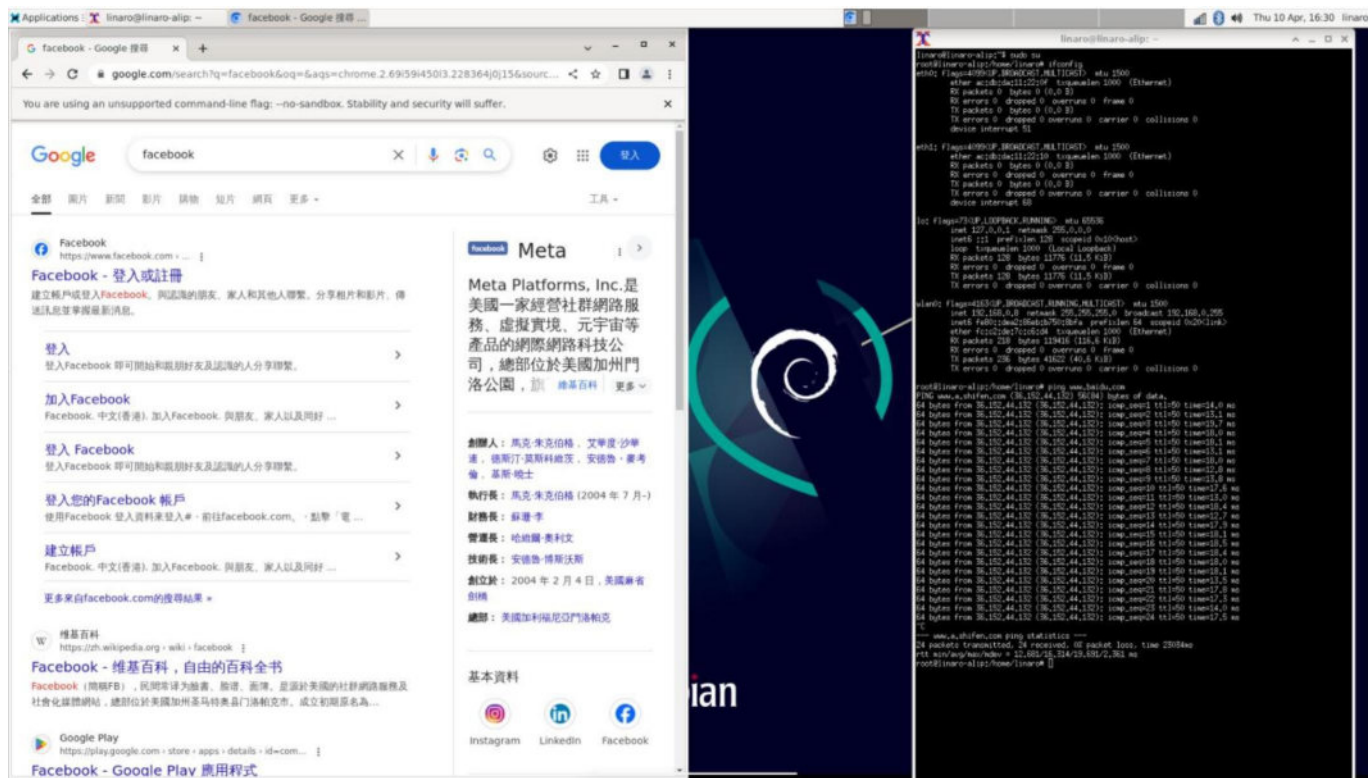


3.2.2 WIFI

WiFi interface connection: click “Network status” in the upper right corner -> click “More networks” -> select search WiFi -> Enter password/direct connection; Execute instructions, check the obtained IP address, ping through Baidu or through the browser, verify the normal network.

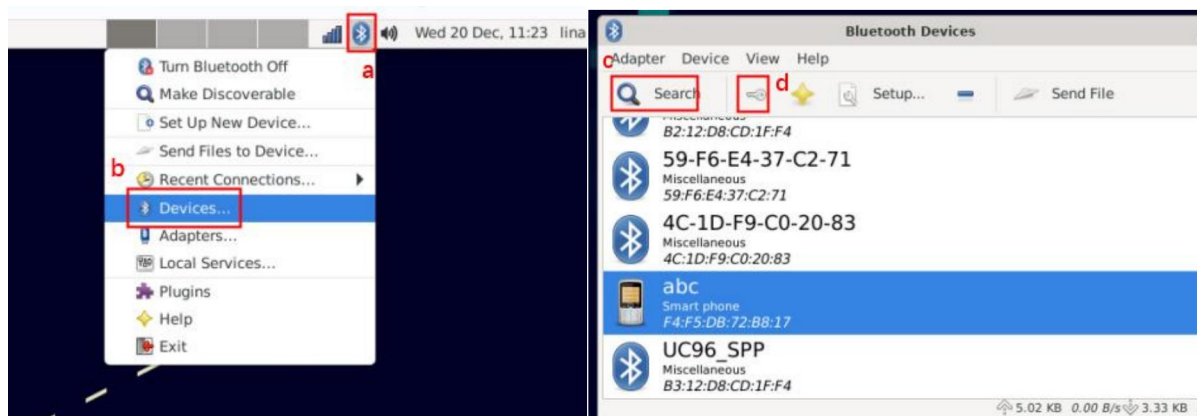
Check the IP address: [ifconfig](#)

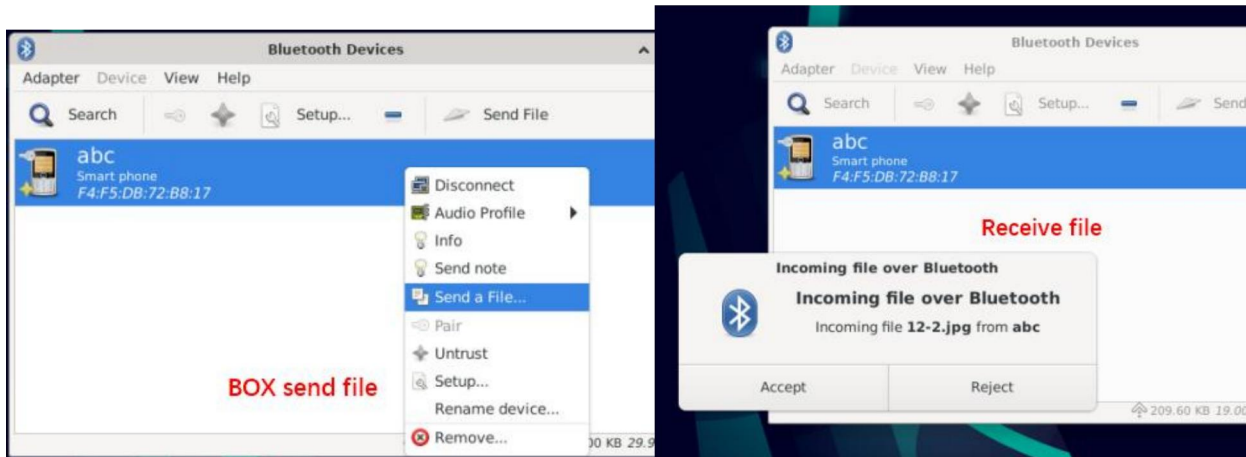




3.2.3 Bluetooth

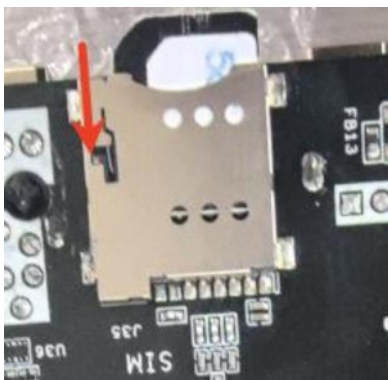
Bluetooth interface connection: a. Click on the “Bluetooth” icon in the upper right corner
 -> b. Select and click “Devices”, Enter the “Bluetooth Devices” interface -> c.
 Click “Search”, Search for Bluetooth devices in the environment -> d. Select and click on
 the “Key” icon to pair -> e. Click on the paired board Bluetooth on the mobile phone to
 connect -> f. Right click on the connected Bluetooth device, select and click on “Send a
 File” to send the file.





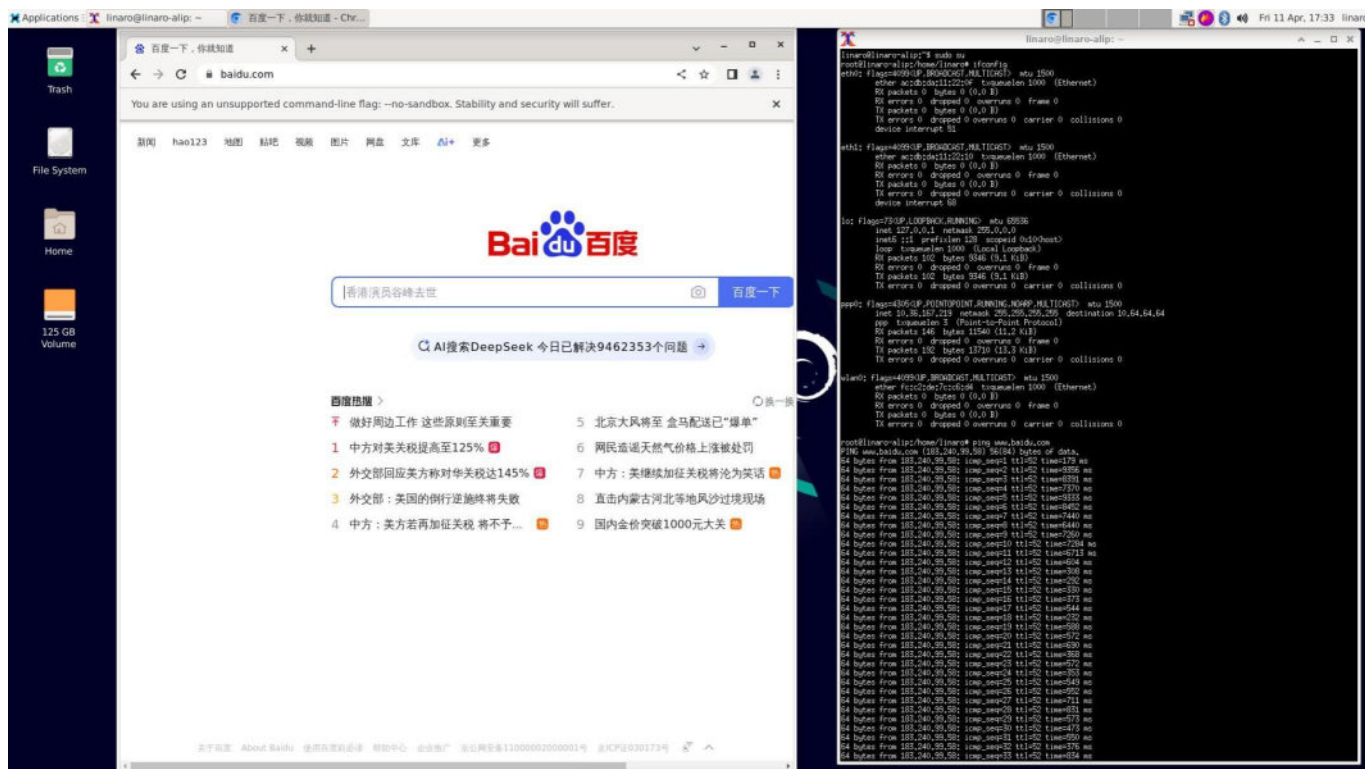
3.2.4 LTE

1. Preparation work: When the board is powered off, connect the LTE 4G module (such as EC25-EUX) and connect the 4G antenna; Insert SIM card (notch facing outward)



2. Power on the board and wait for the system to start up; Wait for about 30 seconds to find the ppp0 node and IP address. Verify network connectivity through interface browser or ping Baidu

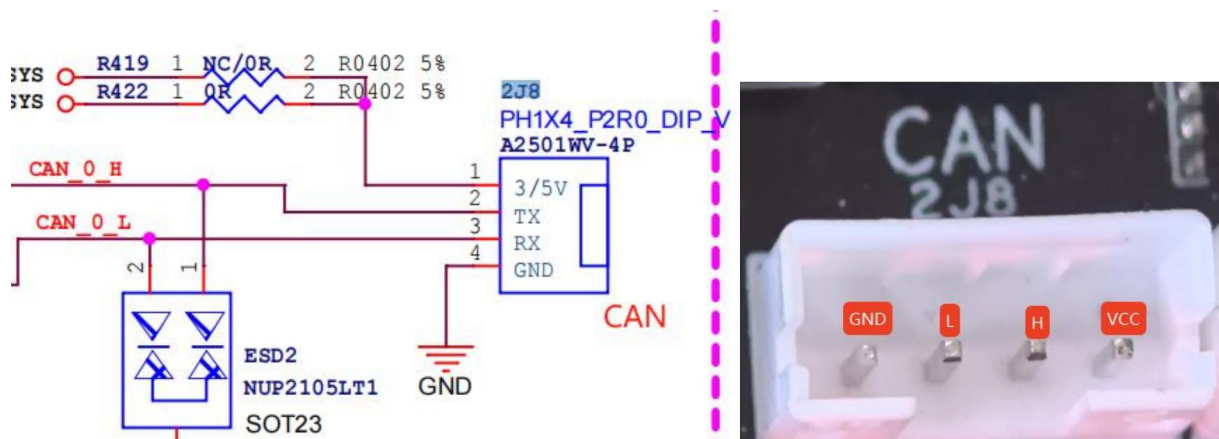
Check the IP address: **ifconfig**



3.3 CAN

3.3.1 Hardware connection

1. The pin definition for bit number 2J8 is shown below:

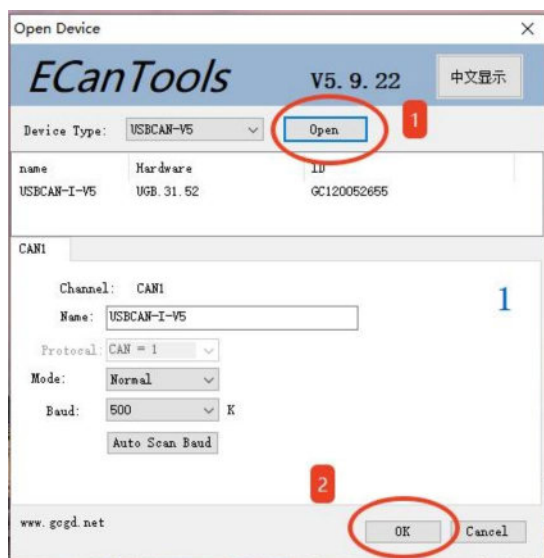


2. Connect the PC to the CAN interface using the CAN test box. Connect to the corresponding CAN test box with reference to the CAN pin schematic; As shown in the picture below:



3.3.2 Terminal operation

1. Open Ecantools on PC , Click to open the device and confirm



2. Serial port check confirms that CAN devices have been identified

ifconfig -a

```
root@linaro-alip:~# ifconfig -a
can0: flags=128<NOARP> mtu 16
    unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 10 (UNSPEC)
    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
    device interrupt 80

eth0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether ac:db:da:11:22:0f 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
    device interrupt 51

eth1: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether ac:db:da:11:22:10 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
    device interrupt 68

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
```

3. Input the following command through the serial port:

ip link set can0 type can bitrate 500000 triple-sampling on
ifconfig can0 up

```
root@linaro-alip:~# ip link set can0 type can bitrate 500000 triple-sampling on
root@linaro-alip:~#
root@linaro-alip:~# ifconfig can0 up
[ 763.898829] [dhd] CFG80211-ERROR: wl_cfg80211_netdev_notifier_call : wdev null. Do nothing
[ 763.899862] [dhd] CFG80211-ERROR: wl_cfg80211_netdev_notifier_call : wdev null. Do nothing
[ 763.899941] IPv6: ADDRCONF(NETDEV_CHANGE): can0: link becomes ready
root@linaro-alip:~# [ 763.899960] [dhd] CFG80211-ERROR: wl_cfg80211_netdev_notifier_call : wdev null. Do nothing

root@linaro-alip:~# ifconfig
can0: flags=193<UP, RUNNING, NOARP> mtu 16
    unspec 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00 txqueuelen 10 (UNSPEC)
    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
    device interrupt 80

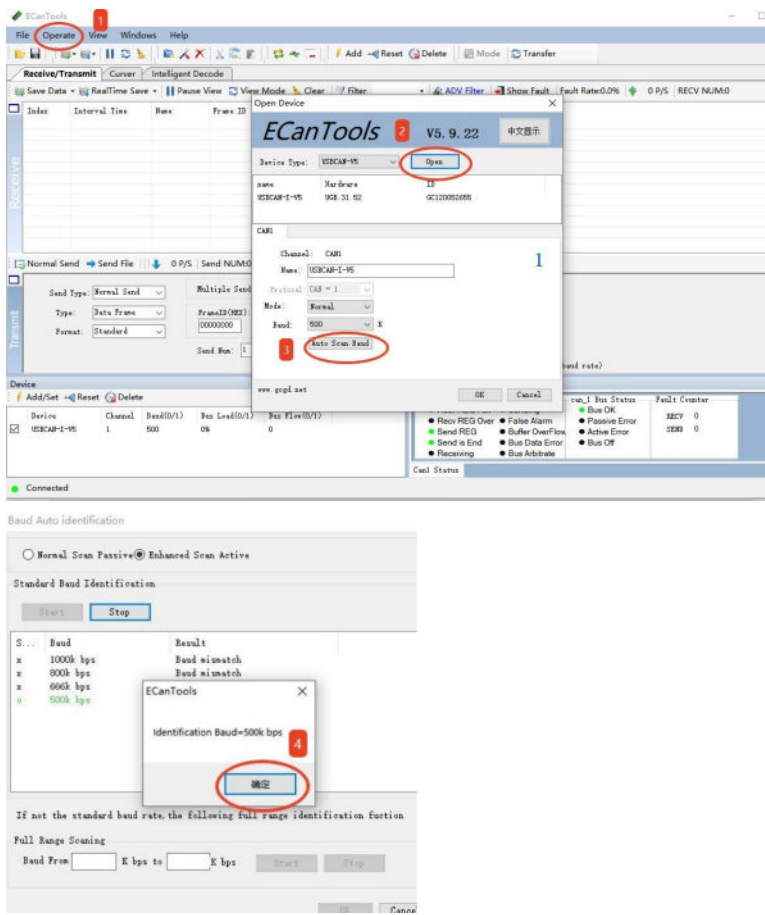
eth0: flags=4099<UP, BROADCAST, MULTICAST> mtu 1500
    ether ac:db:da:11:22:0f txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 51

eth1: flags=4099<UP, BROADCAST, MULTICAST> mtu 1500
    ether ac:db:da:11:22:10 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
    device interrupt 51
```

candump can0 //Receive instructions

cansend can0 123#0102030405060670780980 //Send instructions

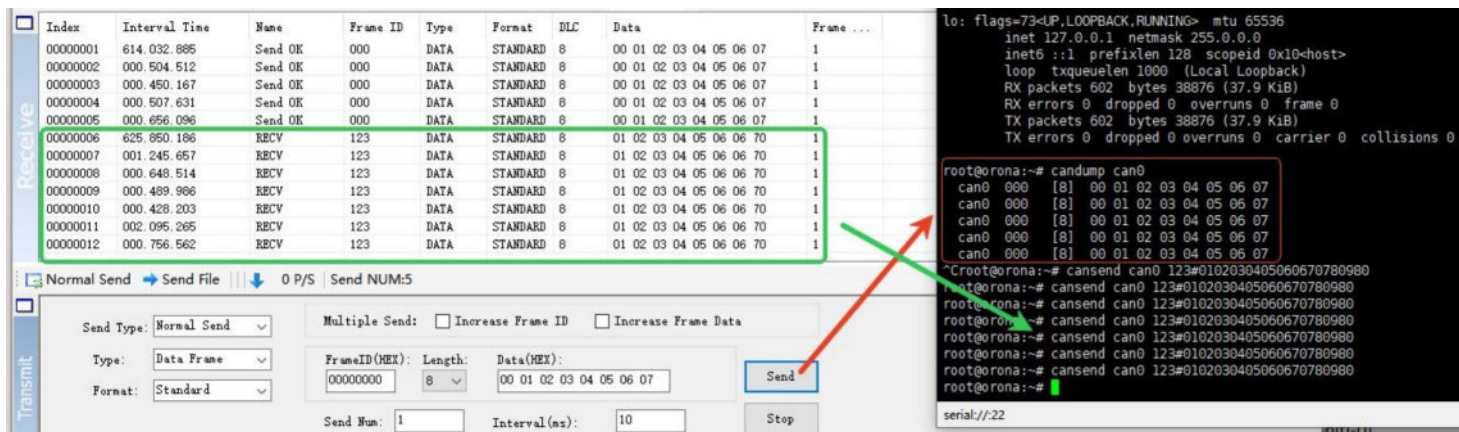
4. Ecantools automatically identifies the baud rate as follows:



5. After Ecantools is set up, execute the following commands in the serial port, and CAN sends and receives data as shown in the figure below:

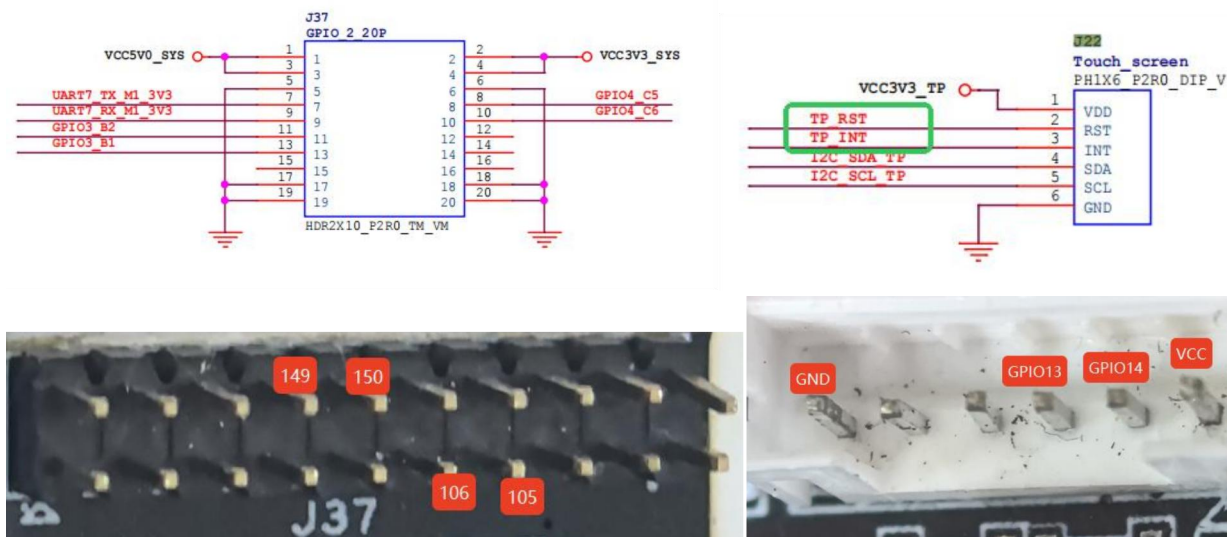
Received: **candump can0**

Send: **cansend can0 123#0102030405060670780980**



3.4 GPIO

1. The GPIO pin schematic definition for bit number J22/J37 is shown below:



2. Manually execute instructions to raise and lower the voltage of the GPIO pins;

Multimeter test voltage

echo 106 > /sys/class/gpio/export

echo out > /sys/class/gpio/gpio106/direction

echo 1 > /sys/class/gpio/gpio106/value //Raise the voltage to 3.4V

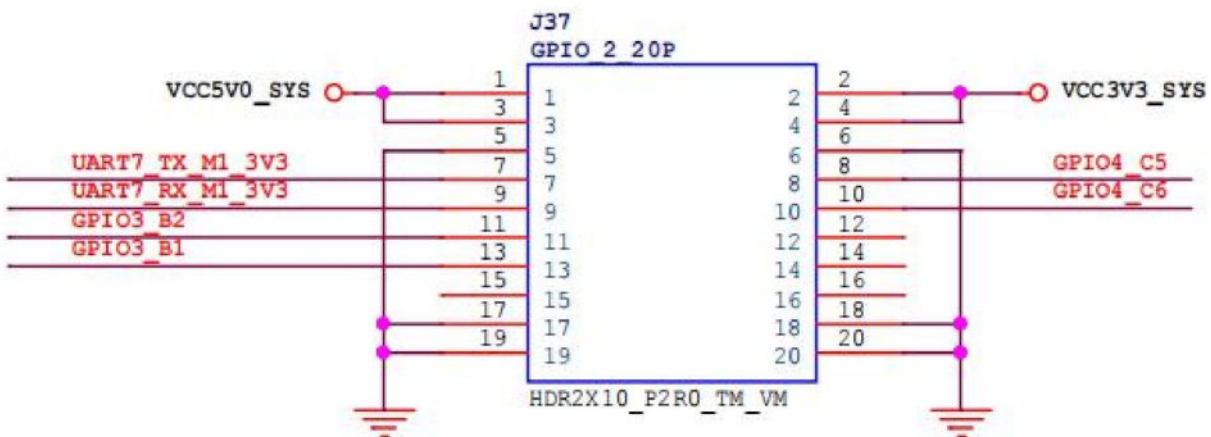
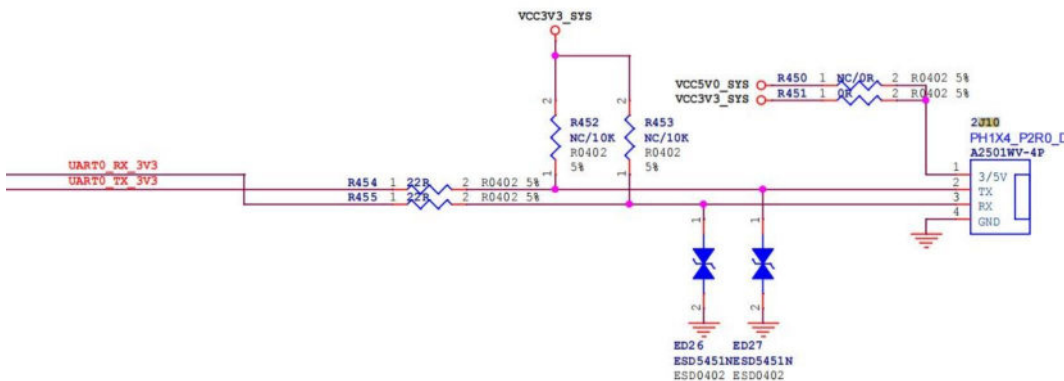
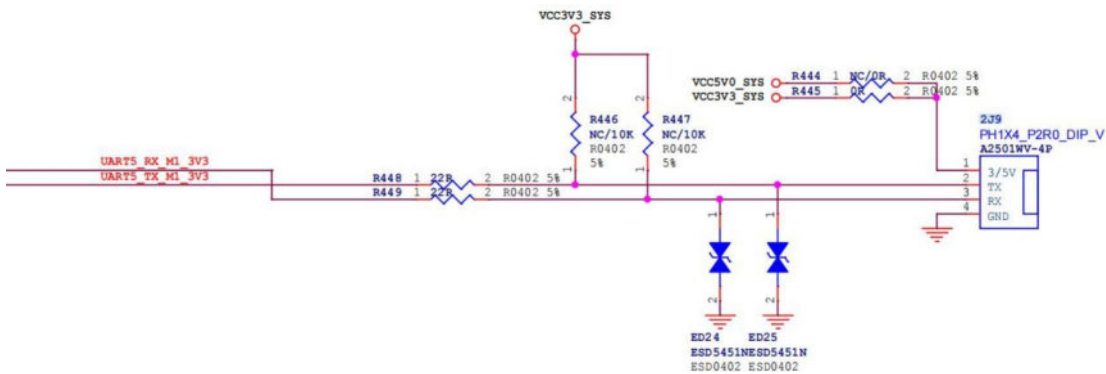
```
echo 0 >/sys/class/gpio/gpio106/value //Lower the voltage to 0V
```

Note: The verification method of GPIO105/149/150/13/14 is the same as above.

(GPIO105/149/150 can measure high voltage about 3.4V, GPIO13/14 can measure high voltage about 1.8V)

3.5 UART

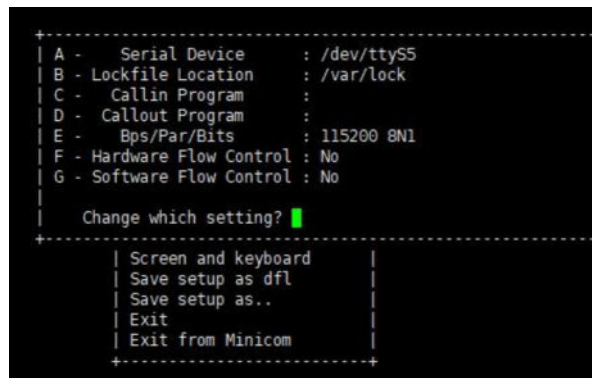
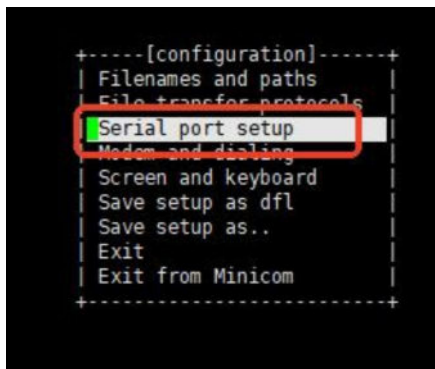
1. Pin schematic definition for the bit number 2J9/2J10/J37 UART.



2. Enter instructions in the Debug port, set the port number (ttyS5) and baud rate, and save them. Test that the two debugging windows can send and receive data from each other.

minicom -s //Set port number and baud rate

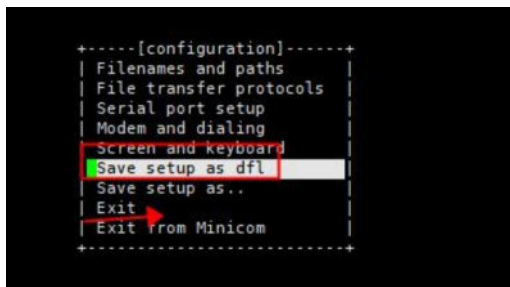
Minicom usage method: Use the up, down, left, and right keys to enter different options. In the serial port setup, you can set the port number and baud rate by changing the front letters in the options (A: change port number, E: change baud rate, etc.)



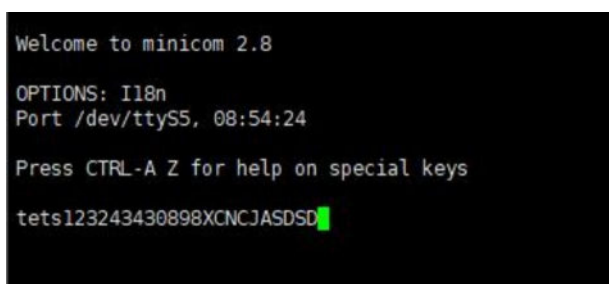
After setting up

Save setup as dfl

Exit



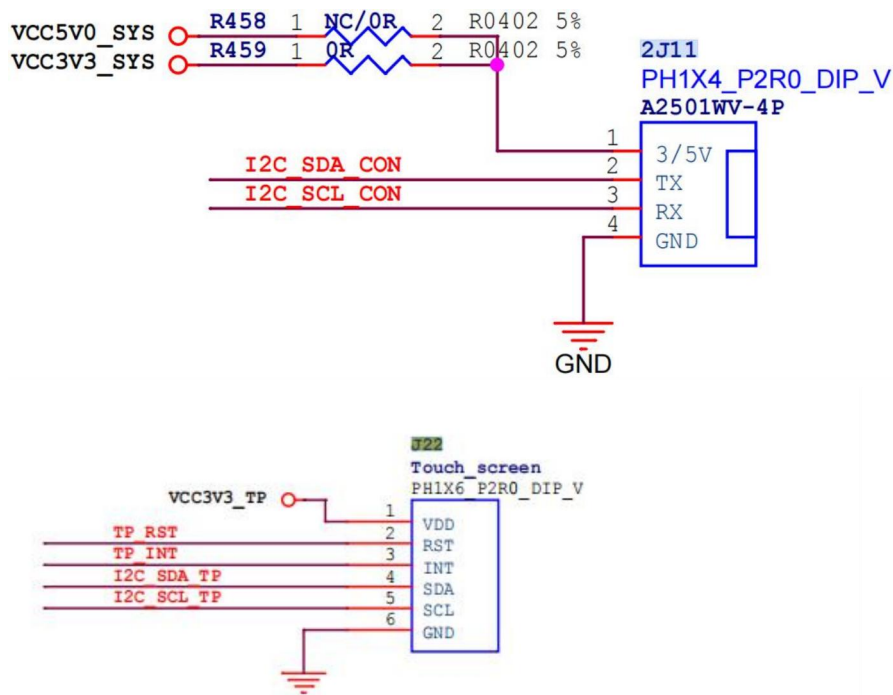
3. The TX/RX of bit number 2J9 can be automatically collected under the minicom interface.



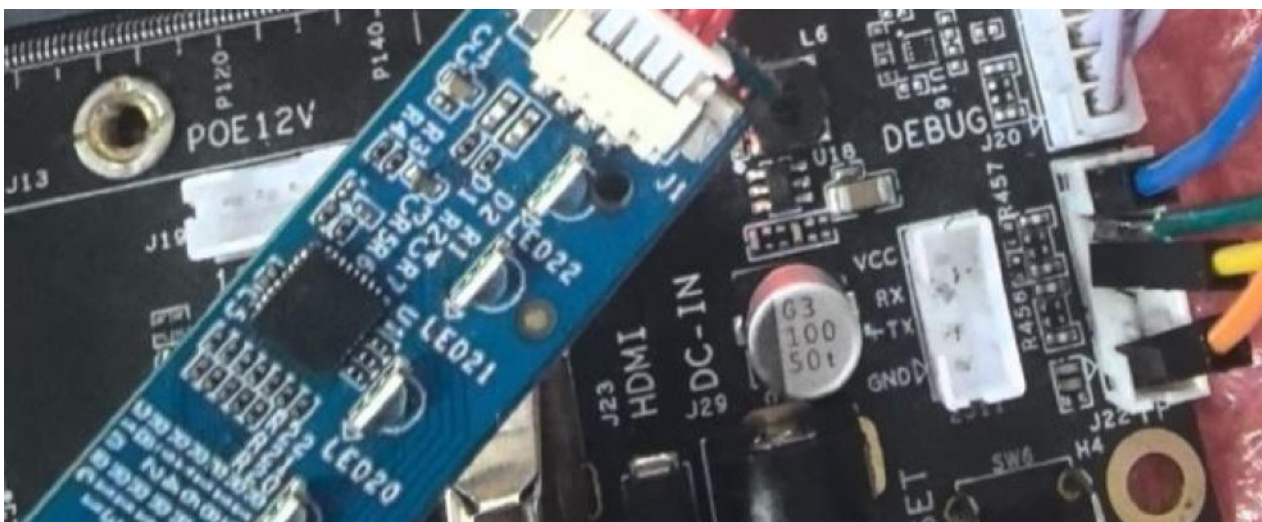
Note: Bit number 2J10 (ttyS0)/bit number J37 (ttyS7) Test method is the same as above.

3.6 I2C

1. The pin definition for bit number 2J11/J22 is shown below:



Note: The CLK/SDA/GND/VCC corresponding pins of the 1.8V I2C device are respectively connected to the board, as shown in the figure below:



2. By entering a command in the Debug port, you can identify a successful I2C mount on the I2C bus.

Not connected to the I2C device, execute the command: **i2cdetect -y 1**

```
root@linaro-alip:/# i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

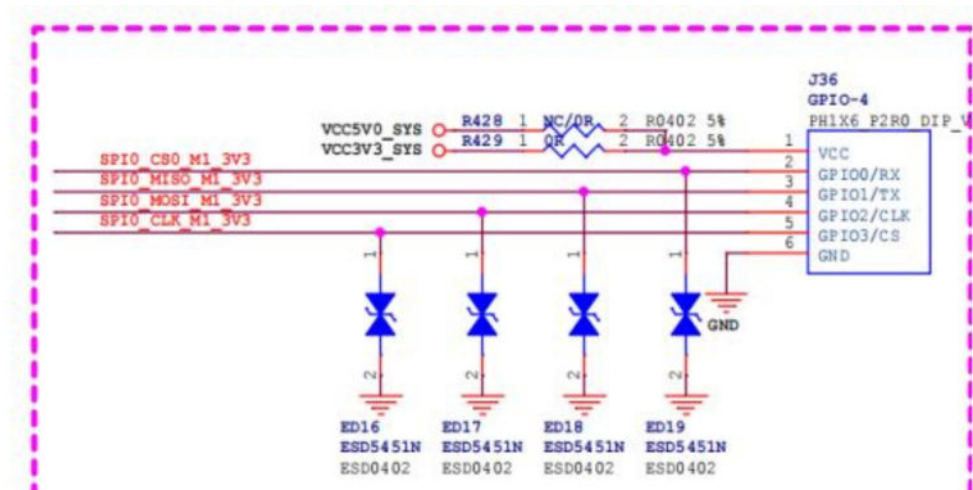
After connecting to the I2C device, execute the command: **i2cdetect -y 1**

```
root@linaro-alip:/# i2cdetect -y 1
   0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
10:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
20:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
30:  --  --  --  34  --  --  --  --  --  --  --  --  --  --  --  --
40:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
50:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
60:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
70:  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --  --
```

Note: The I2C nodes on bit number 2J11/J22 are the same (i2c1), so the view I2C mount instruction is the same.

3.7 SPI

After short-circuiting mosi and miso, the SPI can perform a loopback test, and execute the following commands through the Debug interface:



Path to enter spidev_test system: **cd /usr/bin**

Create files with custom content: **vi spi.txt**

Generate the out.txt file: **./spidev_test -l -i spi.txt -o out.txt**

See that the file is the same as the spi.txt created earlier: **cat out.txt**

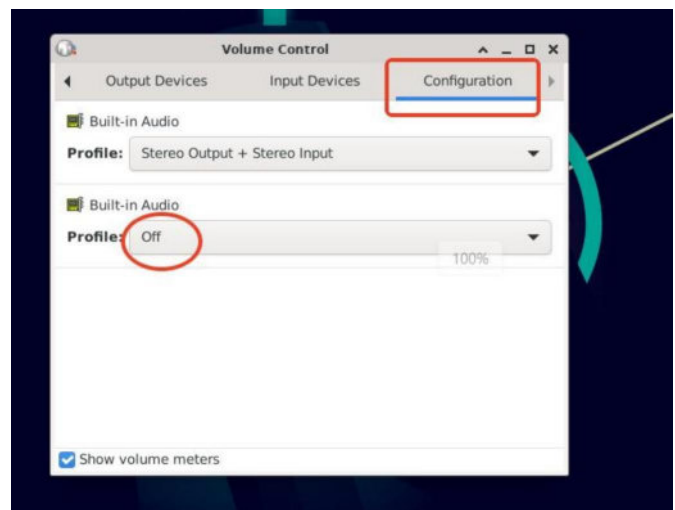
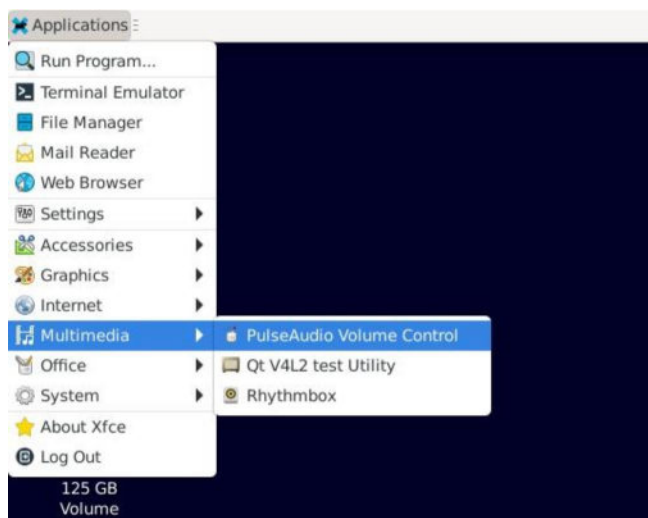
```
root@linaro-alip:/usr/bin# ./spidev_test -l -i spi.txt -o out.txt
spi mode: 0x20
bits per word: 8
max speed: 500000 Hz (500 KHz)
root@linaro-alip:/usr/bin# cat out.txt
test1234
0987geniatech
ASDF
root@linaro-alip:/usr/bin# cat spi.txt
test1234
0987geniatech
ASDF
root@linaro-alip:/usr/bin#
```

3.8 Audio interface

3.8.1 Loudspeaker

Prerequisite: Connect a 4-pin speaker.

Method 1: Interface setting: click "Applications" -> Select "Multimedia" -> Select "PulseAudio Volume Control" -> Click "Configuration" -> Select "Off" in the second line; The speaker can output sound normally.



Method 2: Command to set the speaker to play music, and execute the following command on the debug port

View the audio node:**aplay -l**

```
root@linaro-alip:~# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: rockchiphdmi [rockchip,hdmi], device 0: fe400000.i2s-i2s-hifi i2s-hifi-0 [fe400000.i2s-i2s-hifi
i2s-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: rockchipes8388c [rockchip,es8388-codec], device 0: fe410000.i2s-ES8388 HiFi ES8388.3-0011-0 [fe4
10000.i2s-ES8388 HiFi ES8388.3-0011-0]
  Subdevices: 0/1
  Subdevice #0: subdevice #0
root@linaro-alip:~#
```

Play the audio file:**aplay -Dhw:1,0 101.wav**(Only WAV audio files are supported).

```
root@linaro-alip:~# aplay -Dhw:1,0 101.wav
Playing WAVE '101.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
[ 82.505425] Entering es8388_set_dai_sysclk,freq=12288000
Warning: rate is not accurate (requested = 44100Hz, got = 48000Hz)
[ 82.506274] #83xx#check with es8388_pcm_hw_params() 928 rate[48000Hz with 12288000Hz MCLK] cnt[0][4]
please, try the plug plugin
[ 82.508139] #83xx#check with es8388_pcm_hw_params() 978 rate[48000Hz with 12288000Hz MCLK] cnt[0][4]
[ 82.513800] #83xx-01#check es8388_DEF_VOL with es8388_mute 1013 cnt[16] [30 : 30]

```

3.8.2 MIC

Prerequisite: Connect a 2-pin MIC.

View the recording node: **arecord -l**

```
root@linaro-alip:~# arecord -l
**** List of CAPTURE Hardware Devices ****
card 1: rockchipes8388c [rockchip,es8388-codec], device 0: fe410000.i2s-ES8388 HiFi ES8388.3-0011-0 [fe4
10000.i2s-ES8388 HiFi ES8388.3-0011-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
root@linaro-alip:~#
```

Record wav audio files:**arecord -D hw:1,0 --period-size=1024 --buffer-size=4096 -r**

44100 -c 2 -f s16_le q.wav

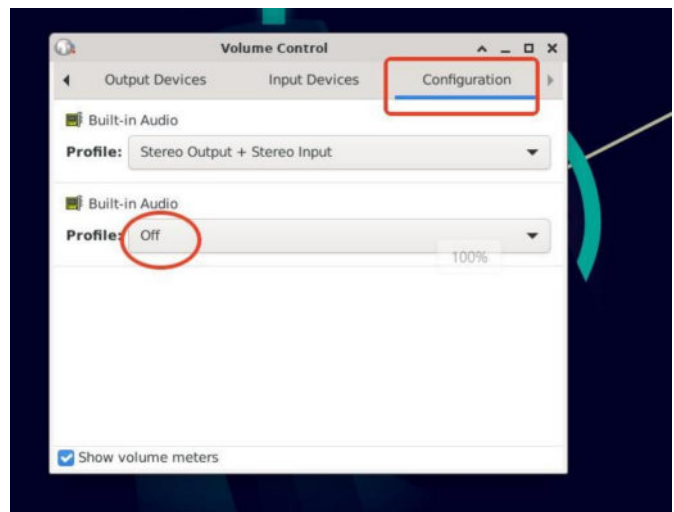
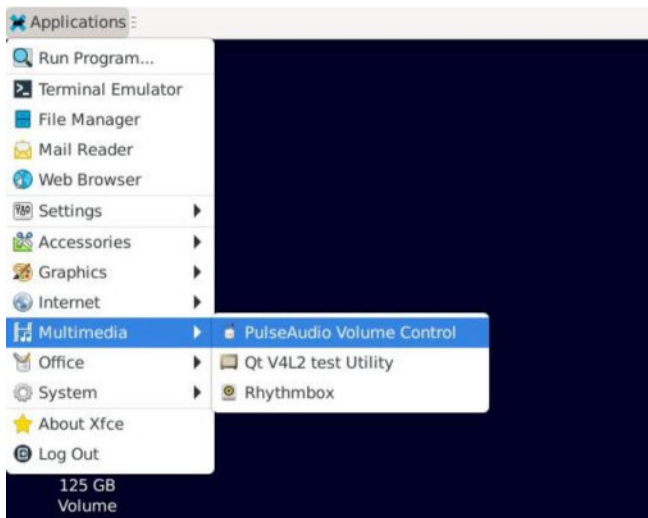
```
root@linaro-alip:~# ls
root@linaro-alip:~# arecord -D hw:1,0 --period-size=1024 --buffer-size=4096 -r 44100 -c 2 -f s16_le q.wav
Recording WAVE 'q.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
[ 819.565475] Entering es8388_set_dai_sysclk,freq=12288000
Warning: rate is not accurate (requested = 44100Hz, got = 48000Hz)
[ 819.566368] #83xx#check with es8388_pcm_hw_params() 928 rate[48000Hz with 12288000Hz MCLK] cnt[0][4]
please, try the plug plugin
[ 819.568236] #83xx#check with es8388_pcm_hw_params() 978 rate[48000Hz with 12288000Hz MCLK] cnt[0][4]
[ 819.574030] #83xx-01#check es8388_DEF_VOL with es8388_mute 1013 cnt[20] [30 : 30]
^CAborted by signal Interrupt...
[ 826.005926] Entering es8388_set_dai_sysclk,freq=0
root@linaro-alip:~#
root@linaro-alip:~# ls -l
total 1188
-rw-r--r-- 1 root root 1212460 Jul  9 10:42 q.wav
root@linaro-alip:~#
```

3.8.3 Headphones

Prerequisites: The headphone jack on the board is connected to the headphone.

3.8.3.1 Play function

Method 1:Interface setting: click "Applications" -> Select "Multimedia" -> Select "PulseAudio Volume Control" -> Click "Configuration" -> Select "Off" in the second line;
The speaker can output sound normally.



Method 2:Command to set the speaker to play music, and execute the following command on the debug port

View the audio node:**aplay -l**

```
root@linaro-alip:~# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: rockchipdmi [rockchip,hdm], device 0: fe400000.i2s-i2s-hifi i2s-hifi-0 [fe400000.i2s-i2s-hifi
i2s-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: rockchipes8388c [rockchip,es8388-codec], device 0: fe410000.i2s-ES8388 HiFi ES8388.3-0011-0 [fe4
10000.i2s-ES8388 HiFi ES8388.3-0011-0]
  Subdevices: 0/1
  Subdevice #0: subdevice #0
root@linaro-alip:~#
```

Play the audio file:**aplay -Dhw:1,0 101.wav**(Only WAV audio files are supported).

```
root@linaro-alip:~# aplay -Dhw:1,0 101.wav
Playing WAVE '101.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
[ 82.505425] Entering es8388_set_dai_sysclk,freq=12288000
Warning: rate is not accurate (requested = 44100Hz, got = 48000Hz)
[ 82.506274] #83xx#check with es8388_pcm_hw_params() 928 rate[48000Hz with 12288000Hz MCLK] cnt[0][4]
please, try the plug plugin
[ 82.508139] #83xx#check with es8388_pcm_hw_params() 978 rate[48000Hz with 12288000Hz MCLK] cnt[0][4]
[ 82.513800] #83xx-01#check es8388_DEF_VOL with es8388_mute 1013 cnt[16] [30 : 30]
```

3.8.3.2 Recording function(American standard)

View the recording node: **arecord -l**

```
root@linaro-alip:~# arecord -l
**** List of CAPTURE Hardware Devices ****
card 1: rockchip-es8388c [rockchip,es8388-codec], device 0: fe410000.i2s-ES8388 HiFi ES8388.3-0011-0 [fe410000.i2s-ES8388 HiFi ES8388.3-0011-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
root@linaro-alip:~#
```

Record wav audio files:**arecord -D hw:1,0 --period-size=1024 --buffer-size=4096 -r 44100 -c 2 -f s16_le 1.wav**

```
root@linaro-alip:~# ls
q.wav
root@linaro-alip:~# arecord -D hw:1,0 --period-size=1024 --buffer-size=4096 -r 44100 -c 2 -f s16_le 1.wav
Recording WAVE '1.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
[ 1450.060246] Entering es8388_set_dai_sysclk,freq=12288000
Warning: rate is not accurate (requested = 44100Hz, got = 48000Hz)
[ 1450.060967] #83xx#check with es8388_pcm_hw_params() 928 rate[48000Hz with 12288000Hz MCLK] cnt[0][4]
please, try the plug plugin
[ 1450.062635] #83xx#check with es8388_pcm_hw_params() 978 rate[48000Hz with 12288000Hz MCLK] cnt[0][4]
[ 1450.068672] #83xx-01#check es8388_DEF_VOL with es8388_mute 1013 cnt[27] [30 : 30]
^CAborted by signal Interrupt...
[ 1454.753148] Entering es8388_set_dai_sysclk,freq=0
root@linaro-alip:~#
root@linaro-alip:~# ls -l
total 2048
-rw-r--r-- 1 root root 876588 Jul  9 10:52 1.wav
-rw-r--r-- 1 root root 1212460 Jul  9 10:42 q.wav
root@linaro-alip:~#
```

3.9 CSI Camera

After connecting the CSI camera, run the following command:



Preview screen:**gst-launch-1.0 v4l2src io-mode=2 device=/dev/video0 !**

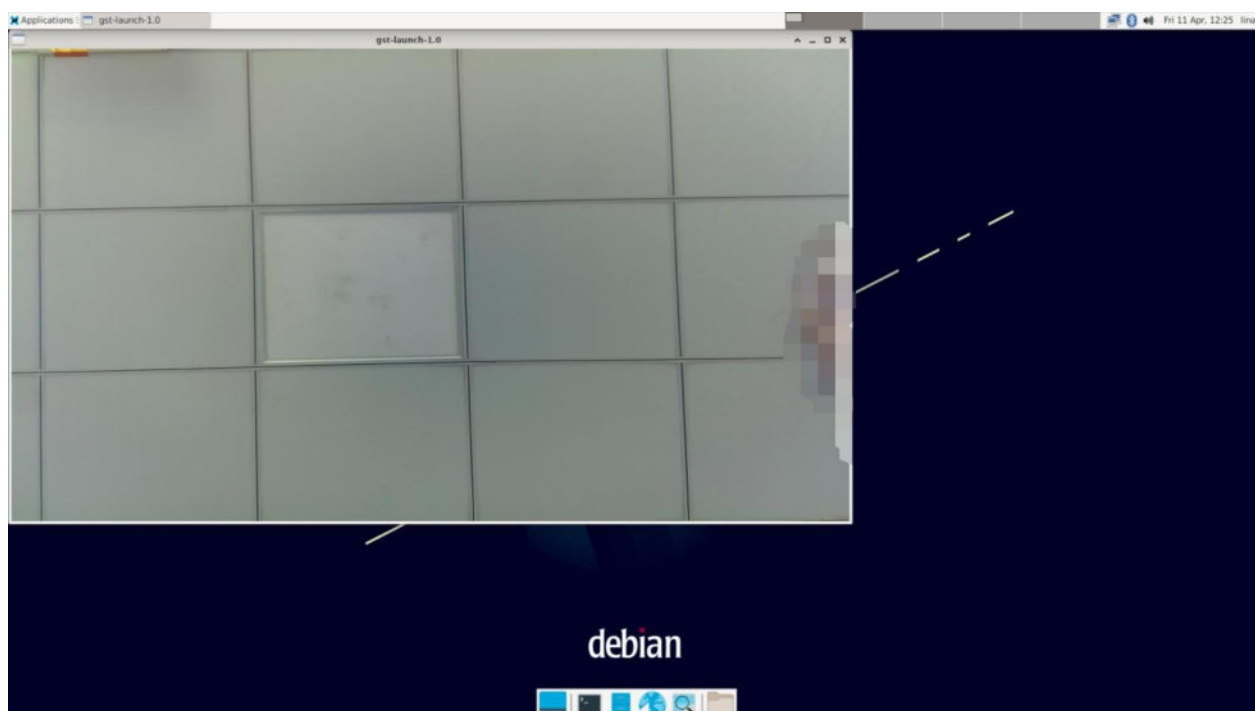
video/x-raw,format=NV12,width=1280,height=720,framerate=30/1 ! xvimagesink

```

root@linaro-alip:~# gst-launch-1.0 v4l2src io-mode=2 device=/dev/video0 ! video/x-raw,format=NV12,width=1280,height=720,framerate=30/1 ! xvimagesink
Setting pipeline to PAUSED ...
Using mplane plugin for capture
Pipeline is live and does not need PREROLL ...
Pipeline is PREROLLED ...
Setting pipeline to PLAYING ...
New clock: GstSystemClock
[ 78.108749] [drm] only support 1 gamma
[ 78.125104] [drm] only support 1 gamma
[ 78.155049] rkisp_hw fdff0000.rkisp: set isp clk = 297000000Hz
[ 78.156859] rockchip-csi2-dphy0: dphy0, data_rate_mbps 600
[ 78.157025] rockchip-csi2-dphy csi2-dphy0: csi2_dphy_s_stream stream on:1, dphy0, ret 0
0:00:01.7 / 99:99:99.

```

激活
转到

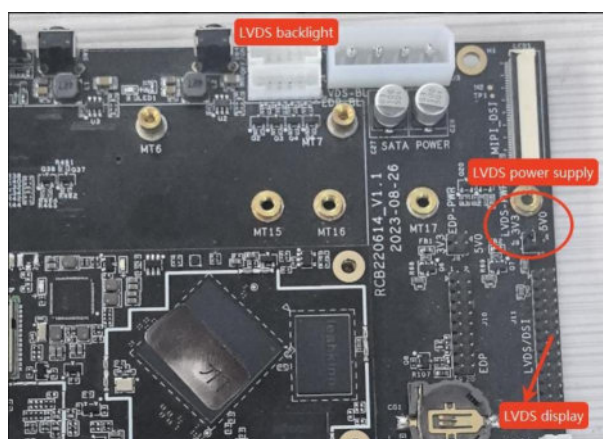


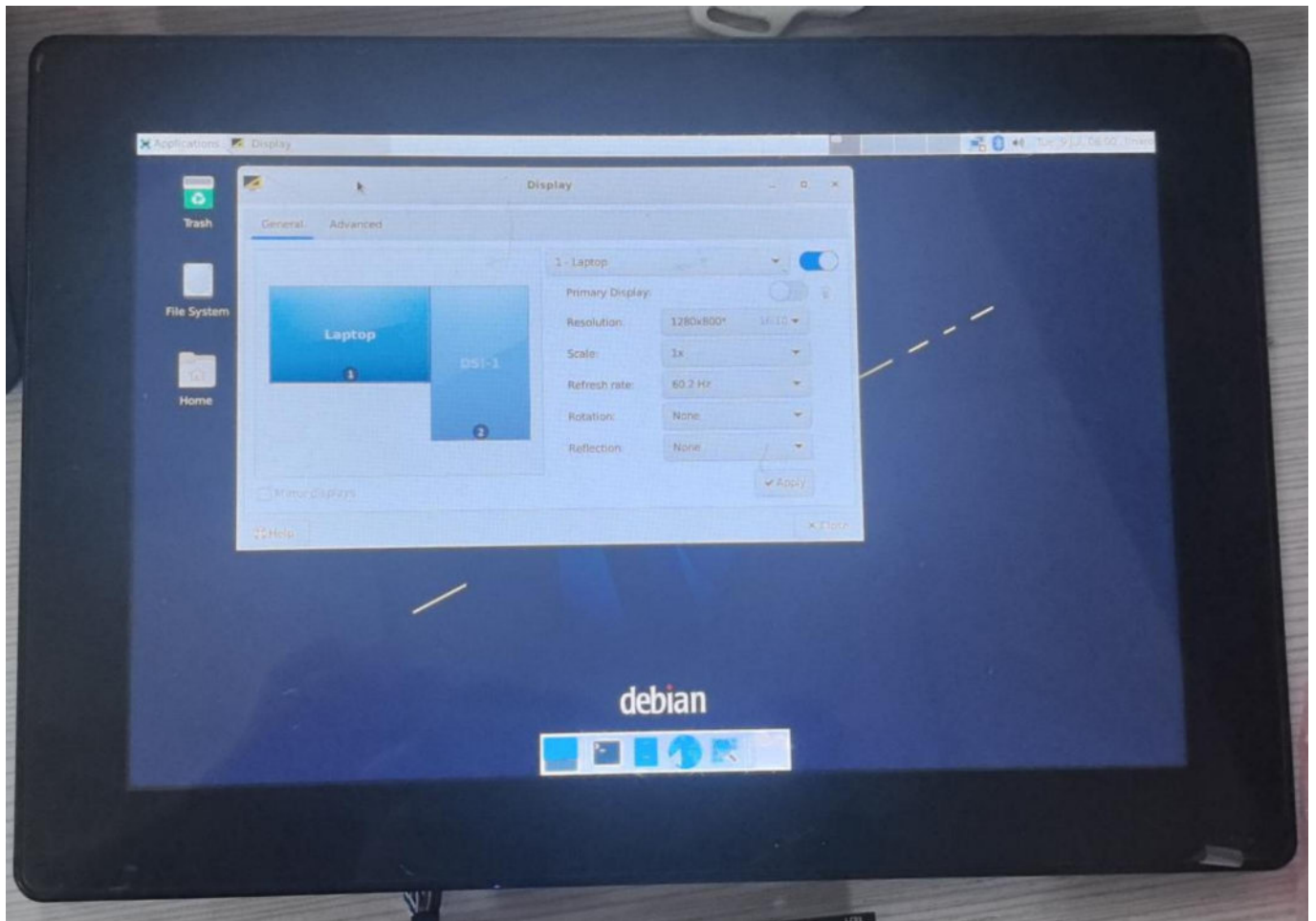
3.10 Screen

3.10.1 LVDS

Connect the LVDS screen (screen cable slot number: J11, backlight cable slot number: J1), without connecting to HDMI; The interface output is displayed in LVDS, and the mouse click function is normal.

Note: LVDS screen light, bit number: J9 need to jump to 3.3V by jumping cap.





3.10.2 MIPI

Connect MIPI screen, not HDMI; MIPI displays the interface output, and the mouse click function is normal.

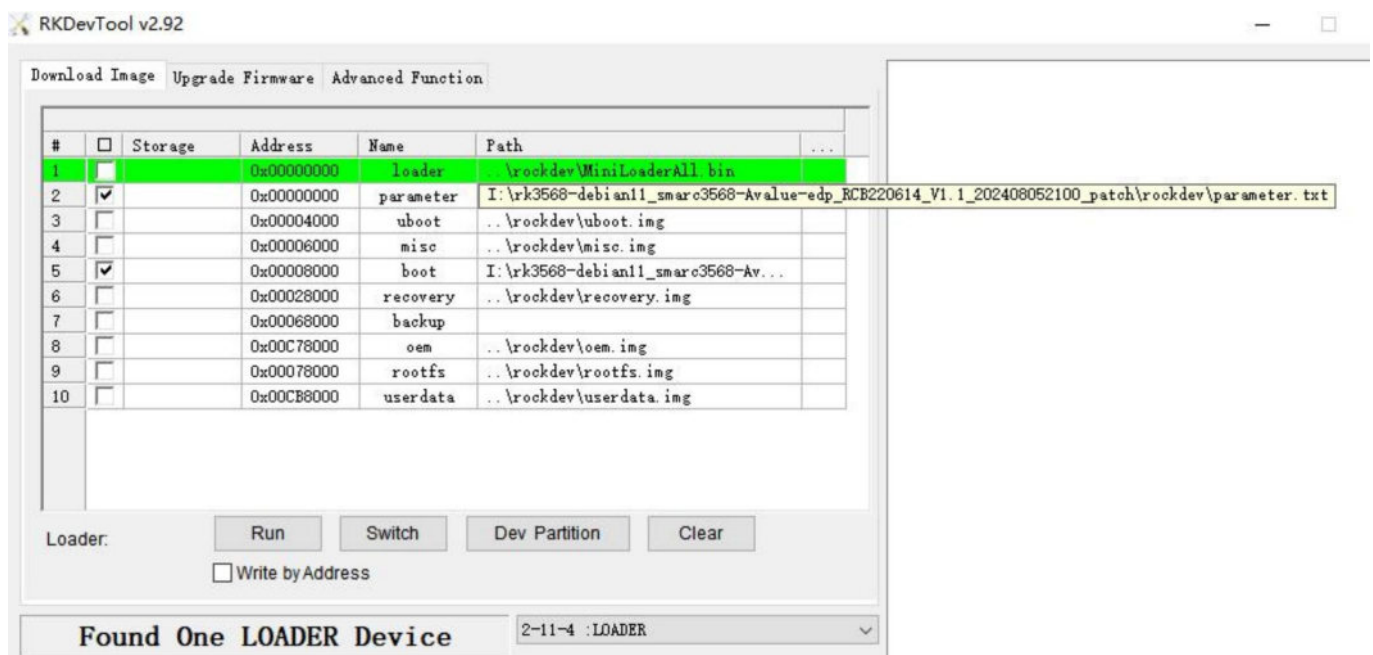


3.10.3 EDP

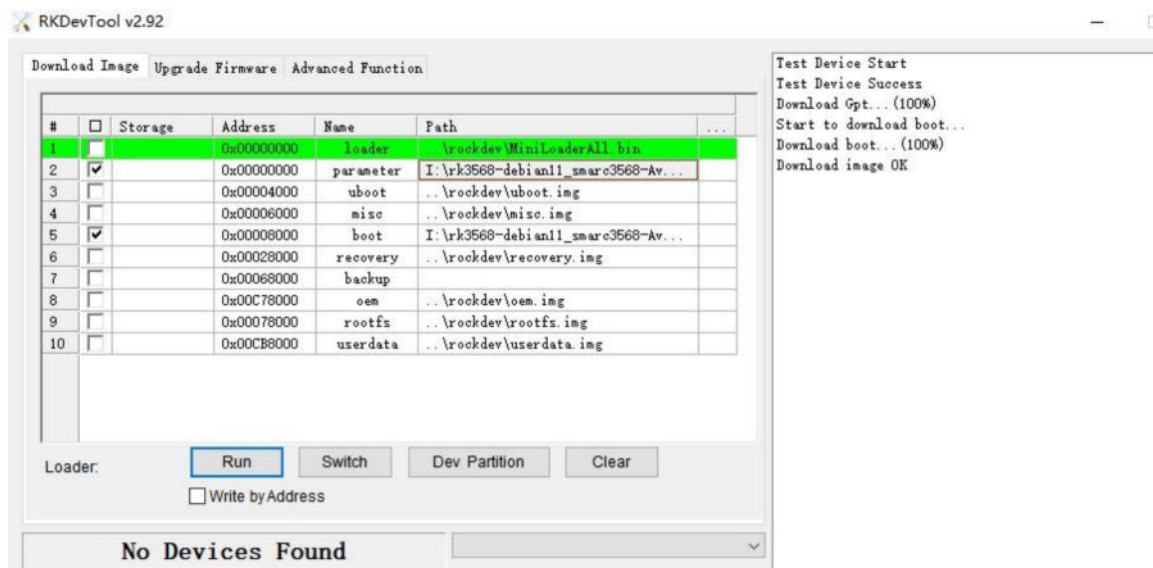
Note: LVDS is not compatible with EDP systems; The LVDS screen output is displayed by default, and the EDP screen output shows that the boot.img file needs to be re-burned.

Operation steps:

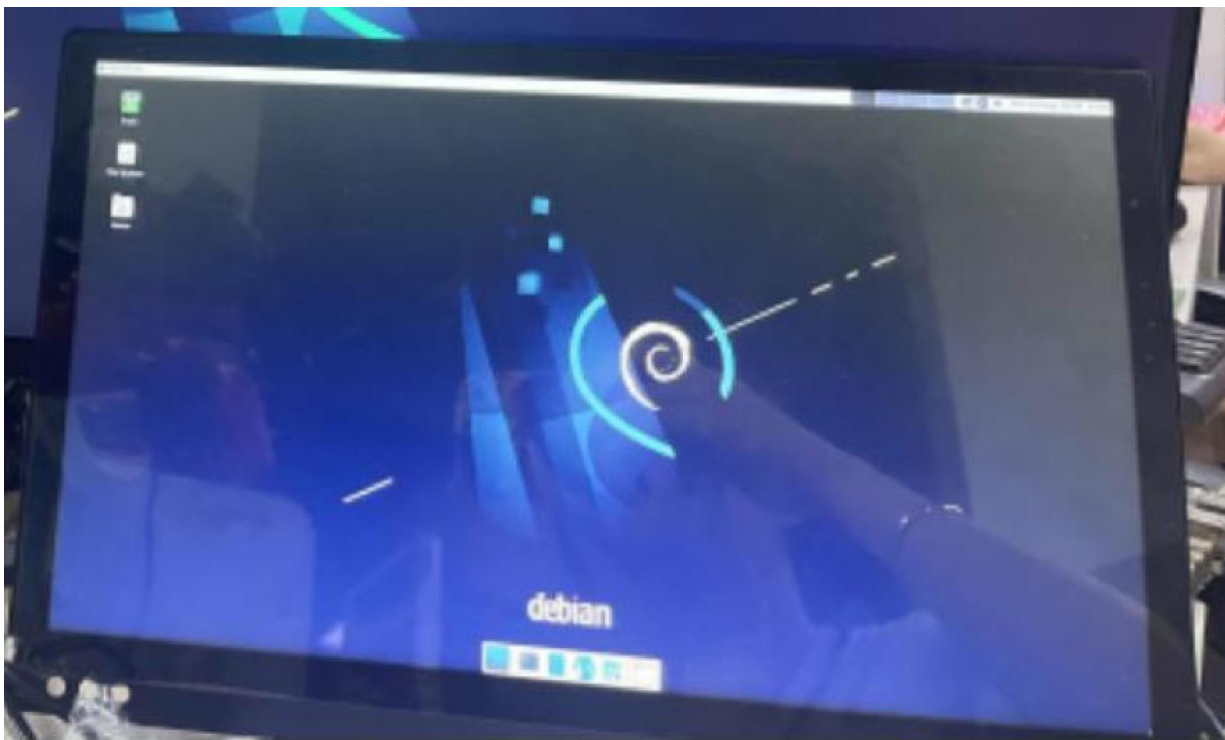
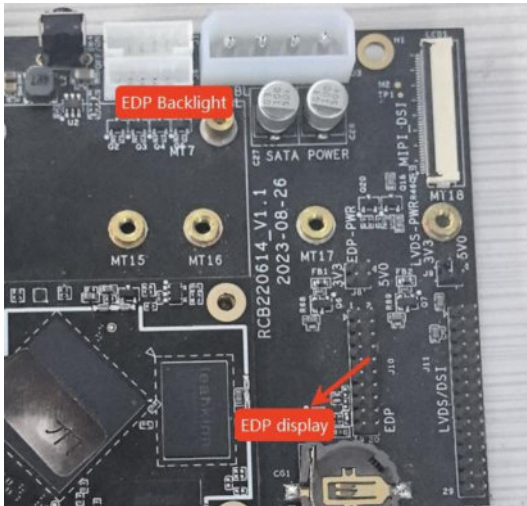
- (1) Contact our service personnel to display patch files corresponding to EDP.
- (2) Enter the burning mode, open the RK burning tool on the PC, and load boot.img and parameter.txt files.



- (3) Click "Run" to complete the burning after the progress bar is loaded.



(4) After restarting the system, verify the EDP display function: Connect EDP screen (screen slot number: J10, backlight slot number: J4), do not connect HDMI; EDP display interface output, and mouse click function is normal.

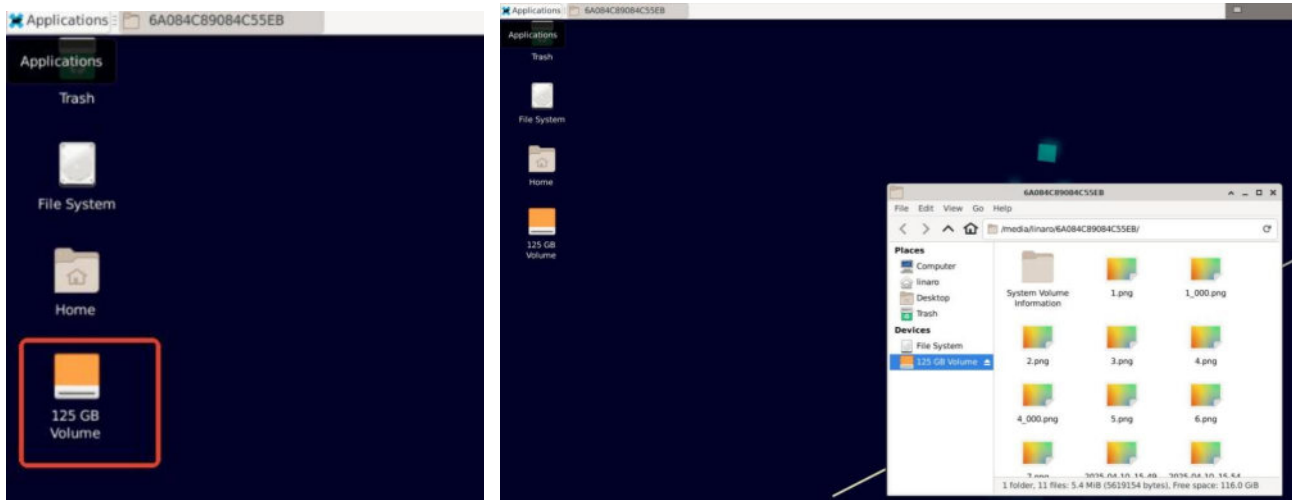


3.11 External Storage Devices

3.11.1 U disk

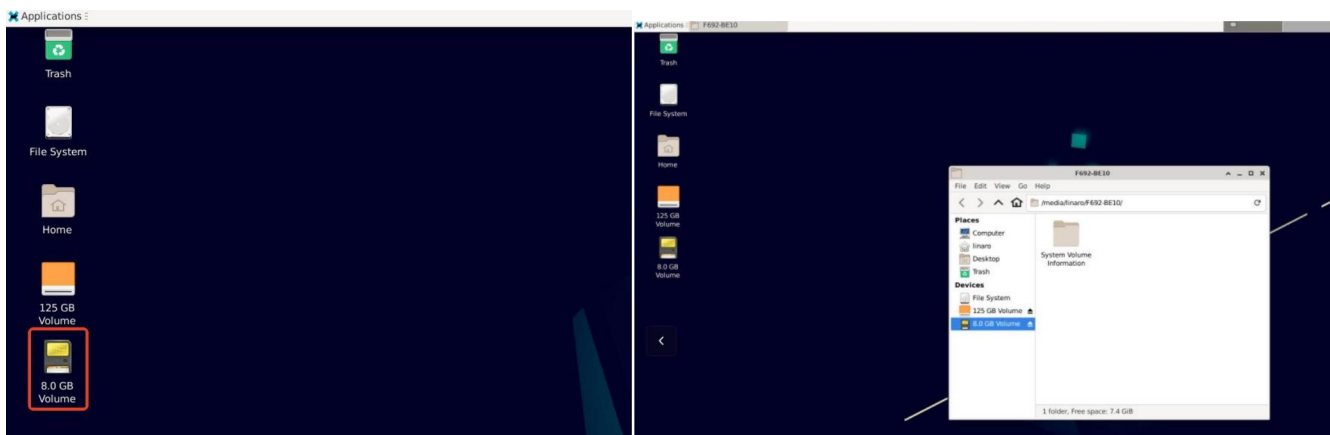
First, insert the U disk and then turn on the power on the board. After the system starts, HDMI OUT will display a USB icon, indicating that the USB drive has been recognized. Double click the icon to enter the corresponding file path of the USB drive, as shown in

the following figure:



3.11.2 TF card

First, insert the TF card and then turn on the power on the board. After the system starts, HDMI OUT will display a TF card icon, indicating that the TF card has been recognized. Double click the icon to enter the file path corresponding to the TF card, as shown in the following figure:



3.11.3 SSD

After the board is connected to the SSD, execute the following commands. You can view the SSD mounting path.

View the SSD mounting path:**df -h**

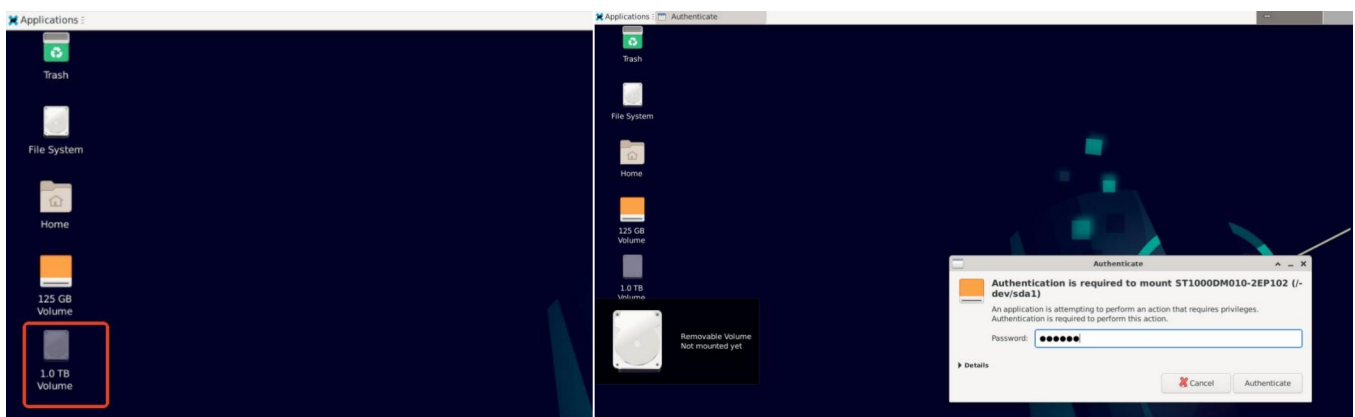
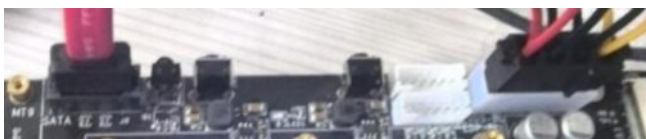
```

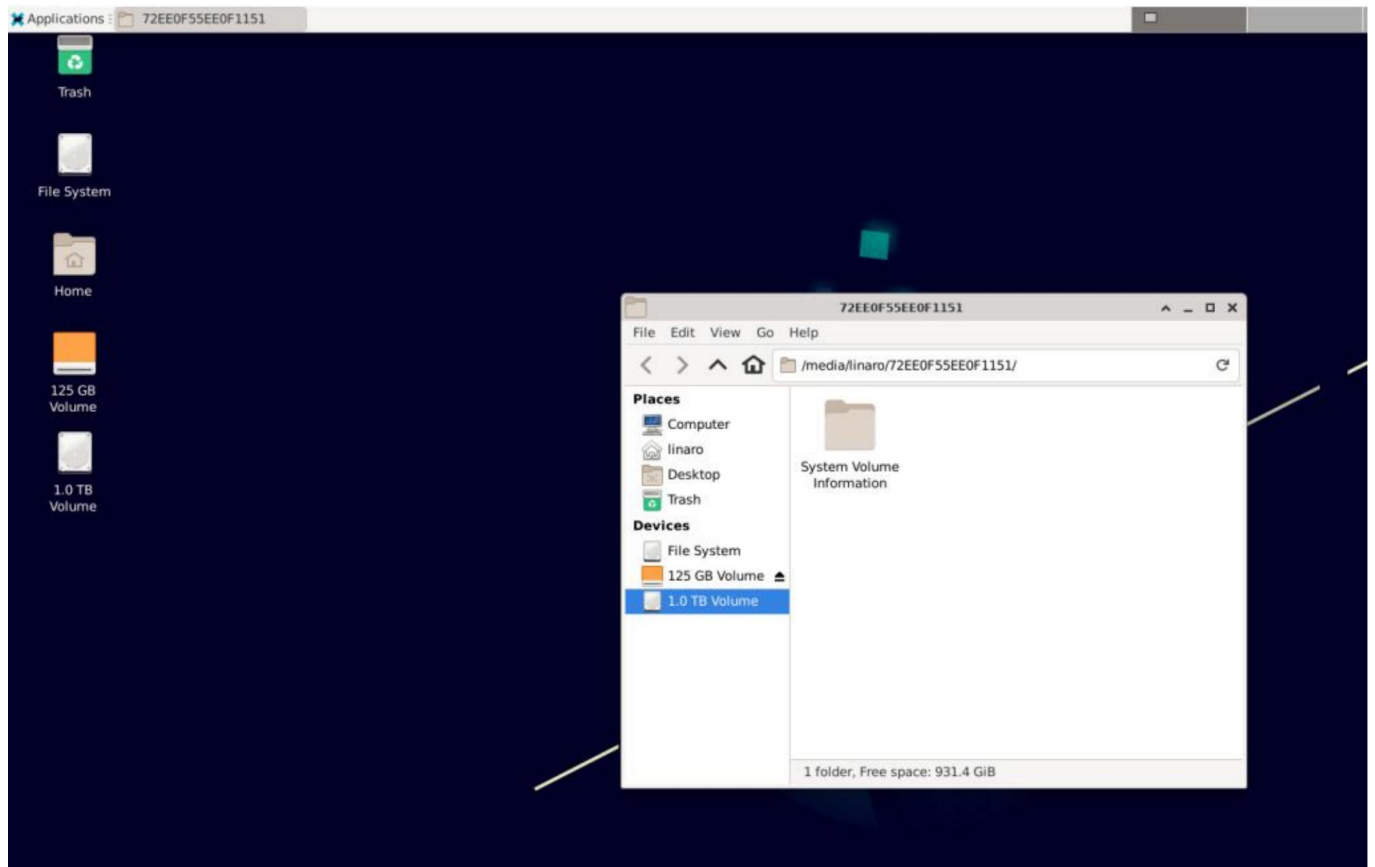
root@linaro-alip:/# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        5.9G  3.5G  2.1G  63% /
devtmpfs         1.9G   8.0K  1.9G   1% /dev
tmpfs            2.0G    0  2.0G   0% /dev/shm
tmpfs            781M   1.1M  780M   1% /run
tmpfs            5.0M   4.0K  5.0M   1% /run/lock
/dev/mmcblk0p7   123M   12M  108M  10% /oem
/dev/mmcblk0p8   23G   288K   23G   1% /userdata
/dev/nvme0n1     239G   72M  239G   1% /mnt/ssd
tmpfs            391M   40K  391M   1% /run/user/1000
root@linaro-alip:/# cd /mnt/ssd
root@linaro-alip:/mnt/ssd# ls -l
total 0
root@linaro-alip:/mnt/ssd# █

```

3.11.4 SATA

After connecting SATA as shown in the following figure, HDMI OUT will display a SATA icon, indicating that SATA has been recognized. Double click the icon and enter the password: linaro to enter the corresponding file path for SATA, as shown in the following figure:





3.12 IR Remote Control

Execute command: `evtest`, select: 0; Short press various buttons in the infrared remote control, and the serial port has corresponding key values for printing and output; As shown in the following figure:

```
^Croot@linaro-alip:/# evtest
No device specified, trying to scan all of /dev/input/event*
Available devices:
/dev/input/event0:      fdd70030.pwm
/dev/input/event1:      rk805 pwrkey
/dev/input/event2:      hdmi_cec_key
/dev/input/event3:      adc-keys
/dev/input/event4:      SEMICO USB Keyboard
/dev/input/event5:      SEMICO USB Keyboard Consumer Control
/dev/input/event6:      SEMICO USB Keyboard System Control
/dev/input/event7:      Logitech USB Optical Mouse
Select the device event number [0-7]: 0
Input driver version is 1.0.1
Input device ID: bus 0x19 vendor 0x524b product 0x6 version 0x100
Input device name: "fdd70030.pwm"
Supported events:
Event type 0 (EV_SYN)
Event type 1 (EV_KEY)
Event code 2 (KEY_1)
Event code 3 (KEY_2)
Event code 4 (KEY_3)
```

```

Event code 580 (KEY_APPSELECT)
Properties:
Testing ... (interrupt to exit)

Event: time 1704150443.244841, type 1 (EV_KEY), code 103 (KEY_UP), value 1
Event: time 1704150443.244841, ----- SYN_REPORT -----
Event: time 1704150443.539054, type 1 (EV_KEY), code 103 (KEY_UP), value 0
Event: time 1704150443.539054, ----- SYN_REPORT -----
Event: time 1704150444.214568, type 1 (EV_KEY), code 108 (KEY_DOWN), value 1
Event: time 1704150444.214568, ----- SYN_REPORT -----
Event: time 1704150444.509156, type 1 (EV_KEY), code 108 (KEY_DOWN), value 0
Event: time 1704150444.509156, ----- SYN_REPORT -----
Event: time 1704150444.790387, type 1 (EV_KEY), code 106 (KEY_RIGHT), value 1
Event: time 1704150444.790387, ----- SYN_REPORT -----
Event: time 1704150445.082507, type 1 (EV_KEY), code 106 (KEY_RIGHT), value 0
Event: time 1704150445.082507, ----- SYN_REPORT -----

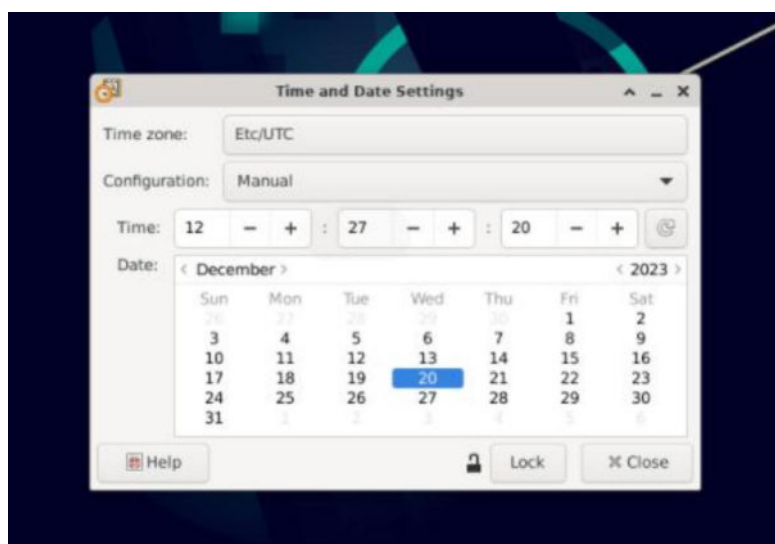
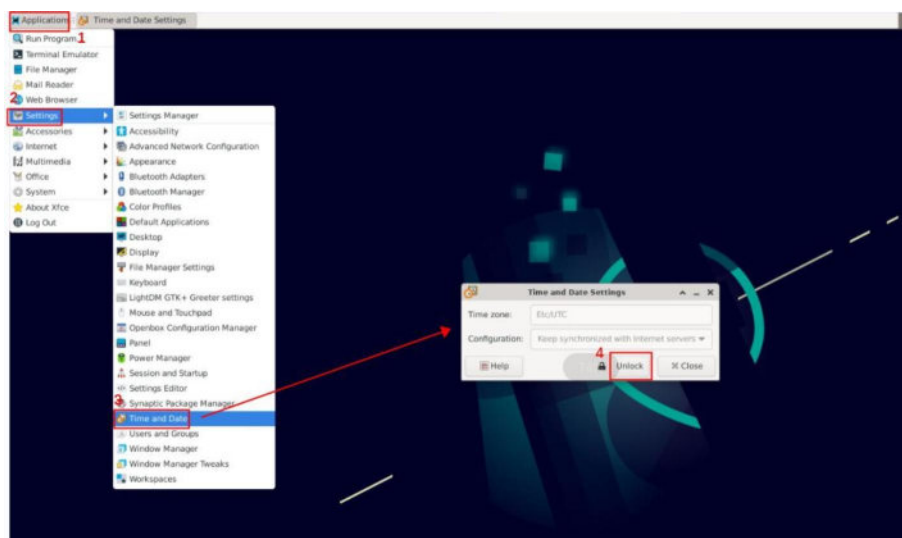
```

print info

3.13 RTC

Prerequisite: Insert RTC battery.

Method 1: Interface settings: Click “Application” ->select “Setting” ->select “Time and Date” ->click “Unlock” (password: linaro) ->switch to “Manual” (to manually set RTC time)



Method 2: Set the command time and execute the following command in the debug port

date -s "2024-01-01 22:43:00" //Set system time

hwclock -w //Write the time into the hardware clock

date //Read system time

hwclock -r //Read hardware clock time

```
root@linaro-alip:/# date -s "2024-01-01 22:43:00"
Mon Jan  1 22:43:00 UTC 2024
root@linaro-alip:/# hwclock -w
root@linaro-alip:/# hwclock -r
2024-01-01 22:43:05.559901+00:00
root@linaro-alip:/# date
Mon Jan  1 22:43:06 UTC 2024
```

3.14 OTG Port

The OTG port can switch between ADB and HOST functions through commands, and the current system defaults to the HOST function as the OTG port (capable of identifying USB devices); The method is as follows:

cat etc/rc.local //View the rc.local file

```
root@linaro-alip:/# cat /etc/rc.local
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.
# Generate the SSH keys if non-existent
if [ ! -f /etc/ssh/ssh_host_rsa_key ]
then
    # else ssh service start in dpkg-reconfigure will fail
    systemctl stop ssh.socket||true
    dpkg-reconfigure openssh-server
fi
echo host > /sys/devices/platform/fe8a0000.usb2-phy/otg_mode
if [ ! -f /usr/local/bin/firstboot.log ]
then
    update-alternatives --set iptables /usr/sbin/iptables-legacy
    iptables -t nat -I POSTROUTING -o eth0 -j MASQUERADE
    sudo chown -R linaro:linaro /home/linaro/.config
    touch /usr/local/bin/firstboot.log
fi
hwclock --hctosys
exit 0
root@linaro-alip:/#
```

ADB function cannot be used,
host function is working properly

vi etc/rc.local //Edit the rc.local file

Usage of Vi Editor: Press "i" to enter editing mode, start inputting characters, press "Esc" to return to command mode, enter ":" wq" to save the modified file and exit, or ":" q!" to exit without saving the file

```
root@linaro-alip:/# cat /etc/rc.local
#!/bin/sh -e
#
# rc.local
#
# This script is executed at the end of each multiuser runlevel.
# Make sure that the script will "exit 0" on success or any other
# value on error.
#
# In order to enable or disable this script just change the execution
# bits.
#
# By default this script does nothing.
# Generate the SSH keys if non-existent
if [ ! -f /etc/ssh/ssh_host_rsa_key ]
then
    # else ssh service start in dpkg-reconfigure will fail
    systemctl stop ssh.socket||true
    dpkg-reconfigure openssh-server
fi
#echo host > /sys/devices/platform/fe8a0000.usb2-phy/otg_mode
if [ ! -f /usr/local/bin/firstboot.log ]
then
    update-alternatives --set iptables /usr/sbin/iptables-legacy
    iptables -t nat -I POSTROUTING -o eth0 -j MASQUERADE
    sudo chown -R linaro:linaro /home/linaro/.config
    touch /usr/local/bin/firstboot.log
fi

hwclock --hctosys
exit 0
root@linaro-alip:/#
```

Comment out this code. The host function becomes unavailable but ADB functions normally

```
C:\Users\Administrator>adb devices
* daemon not running; starting now at tcp:5037
* daemon started successfully
List of devices attached
e4010d1444fda9fc      device

C:\Users\Administrator>adb shell
root@linaro-alip:/# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.6.100 netmask 255.255.255.0 broadcast 192.168.6.255
    inet6 fe80::6d78:660f:3793:635f prefixlen 64 scopeid 0x20<link>
    ether 7e:fl:46:6c:89:6c txqueuelen 1000 (Ethernet)
    RX packets 17369 bytes 25543007 (24.3 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 8628 bytes 579537 (565.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
    device interrupt 47

eth1: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    ether 00:00:00:00:00:01 txqueuelen 1000 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
```

ADB function is working properly

FCC Statement

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna.
- Increase the separation between the equipment and receiver.
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected.
- Consult the dealer or an experienced radio/TV technician for help.

Caution: Any changes or modifications to this device not explicitly approved by manufacturer could void your authority to operate this equipment.

This device complies with part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) This device may not cause harmful interference, and (2) this device must accept any interference received, including interference that may cause undesired operation.

RF Exposure Information

This equipment complies with FCC radiation exposure limits set forth for an uncontrolled environment. This equipment should be installed and operated with minimum distance 20cm between the radiator and your body.