EM1126B-P Linux6.1 User Manual

V2.0

Boardcon Embedded Designer

Overview

The content of this document is intended solely for the EM1126B-P development board, aiming to help users quickly understand, apply, and test the EM1126B-P development board.

System Support

Development Board	Debian12	Buildroot
Mini1126 V5 EM1126B-P V4	Y	Y

Revision History

Version	Date	Author	Revision History
V1.0	2025-08-12	Liu Yuan	Initial version
V2.0	2026-1-21	Chen Yu	Add IPC board support to Buildroot

Disclaimer

The information in this manual is for reference only. While Boardcon strives to ensure its accuracy, no guarantees are made regarding its completeness or correctness. All content is subject to change without prior notice. Boardcon reserves the right to revise the content of this manual without prior notification.

Boardcon embedded design limited

2007-11 Haofang Tianji Plaza, 11008 Beihuan Avenue, Nanshan District, Shenzhen, Guangdong, China. 518051

URL: www.armdesigner.com | www.boardcon.com

Email: market@armdesigner.com

Technical Support Inquiries: support@armdesigner.com

Tel: +86-755-26481393 | +86-755-27571591

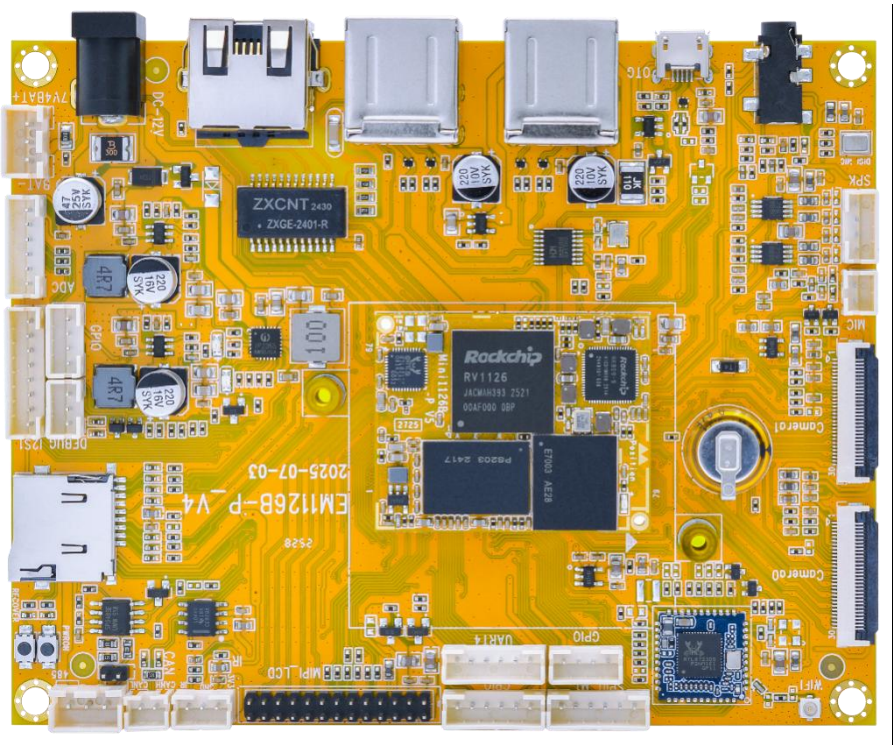
Content

1.Introduction

1.1 Overview

EM1126B-P is a high-performance intelligent vision development platform based on the Rockchip RV1126B AI SoC. It integrates a quad-core Cortex-A53 processor and a 3TOPS NPU, specifically designed for edge AI computing, video analytics, and image recognition applications.

With a built-in multi-channel AI-ISP image processing engine, H.264/H.265 codec, and a rich set of multimedia and peripheral interfaces, EM1126B-P supports MIPI camera inputs and 1080P display output. It is widely applicable to smart cameras, AI edge boxes, security surveillance, industrial vision, and more.



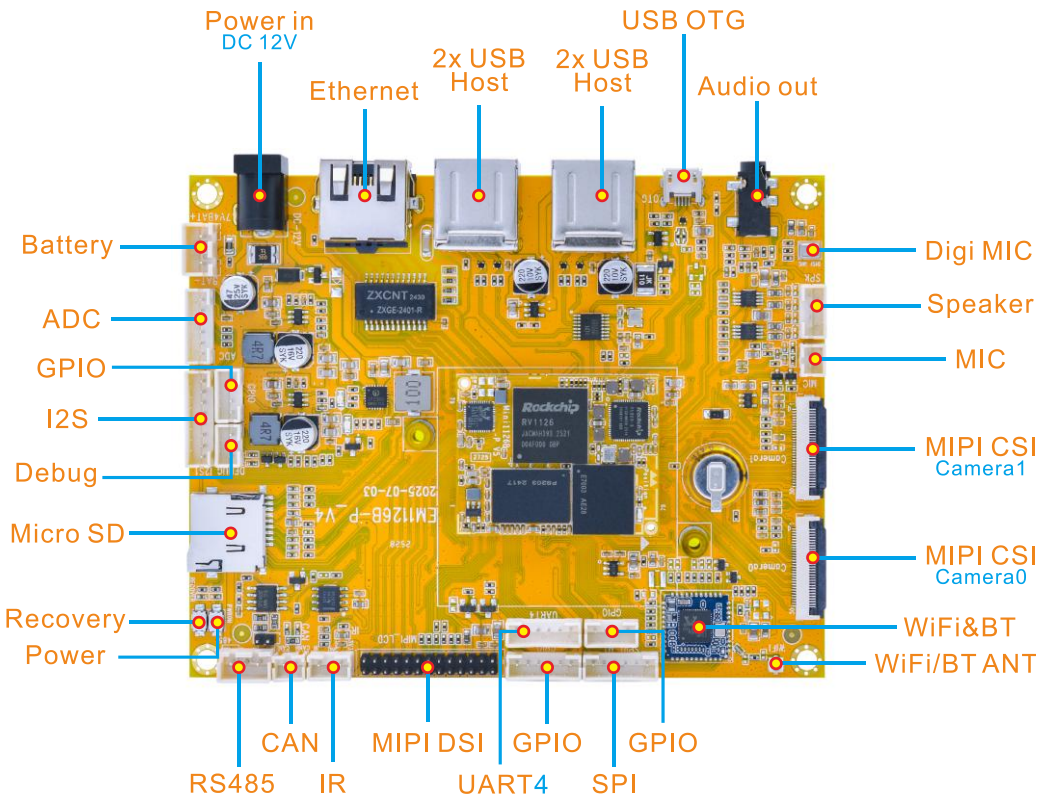
1.2 Product Parameters

Basic Parameters	
SOC	RV1126B-P

CPU		• Quad-core ARM Cortex-A53 • 32KB I-Cache / 32KB D-Cache per core • 512KB shared L2 Cache
NPU		• Up to 3 TOPS AI computing power • Supports INT4, INT8, INT16, FP16, BF16, TF32 • Optimized for CNN, Transformer, and other AI models
Video	Decoder	• H.265/H.264 up to 3840×2160@30fps • JPEG/MJPEG, VP8, VP9 decoding
	Encoder	• H.265/H.264 up to 3840×2160@30fps
AI-ISP		• Built-in 12MP AI Image Signal Processor • Supports multi-channel HDR, denoise, geometric correction
RAM		2GB LPDDR4
ROM		8GB eMMC
Support system		Buildroot
Hardware Parameters		
Extended Storage		• Support 1x MicroSD Card
Display		• Support 1x DSI MIPI
Audio		• Support 1x Speaker • Support 1x Headset • Support 1x Differential MIC • Support 1x Digital MIC
USB		• Support 4x USB2.0
Network		• Support 1x Gigabit Ethernet

	• Support 1x WIFI/BT module
Camera	• Support 2x Camera
Peripheral communication	• Support 1x SPI • Support 1x RS485 • Support 1x RS232 • Support 1x CAN
Other parameters	Support 1xDebug UART, 1xOTG, 1xIR, 1xRTC, 1x Lion Battery connector
Electrical Parameters	
Power supply input voltage	12V/3A
RTC input voltage	3V/0.6uA
Operating temperature	0~70°
Storage temperature	-40~85°
Structural Parameters	
Core board dimensions	38mm x 30mm
Motherboard dimensions	120mm x 95mm

1.3 Hardware Interface Introduction



Interface parameters	
Power in DC 12V	12V DC power input interface
Ethernet	Gigabit Ethernet RJ45 interface
2x USB Host	Dual-layer USB2.0 HOST interface
USB OTG	USB OTG interface
Audio out	Headset out
Digi MIC	Digital MEMS MIC
Speaker	Speaker output
MIC	Differential MIC
MIPI CSI Camera1	MIPI CSI camera interface 1
MIPI CSI Camera0	MIPI CSI camera interface 0
WIFI&BT	WIFI&Bluetooth module

WIFI&BT ANT	Wi-Fi/Bluetooth antenna interface
GPIO	GPIO extension interface
SPI	SPI interface
UART4	UART4 serial communication interface
MIPI DIS	MIPI display interface
IR	IR interface
CAN	CAN communication interface
RS485	RS485 communication interface
Power	Power key
Recovery	Recovery key
Mirco SD	MicroSD card slot
Debug	Debug the serial port
ADC	ADC interface
Battery	Lion Battery interface

2.Install Drivers and Tool

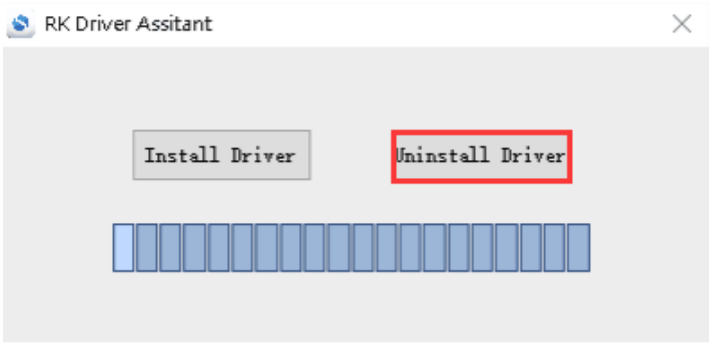
To download firmware and debug in the terminal, the following drivers and software need to be installed (for Windows computers):

Number	Driver name	Driver	Use
1	RK Driver Assitant	DriverInstall.exe	OTG USB driver installation assitant
2	CH9102x	SETUP.EXE	Serial port debugging driver
3	Serial Terminal Tool	SecureCRT.exe	Debugging tool

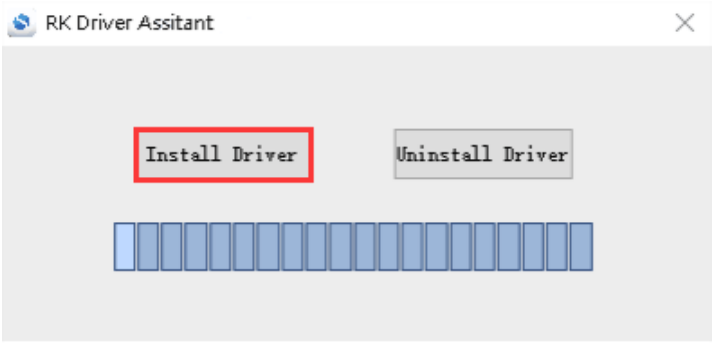
2.1 Install RK Driver Assitant

Step 1: Open *DriverAssitant_v5.1.1/DriverInstall.exe*.

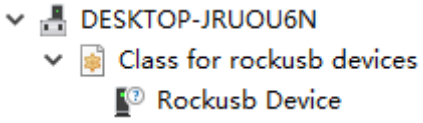
Step 2: To avoid driver conflicts, click **“Uninstall Driver”** to uninstall the driver.



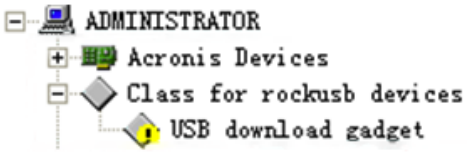
Step 3: Click button **"Install Driver"** to install.



Step 4: After the installation is complete, connect the board and PC with Micro USB cable and press the **Recovery** key and hold then power the board, the following information is displayed in the Computer **Device Manager**, indicating that the USB driver was successfully installed.



Step 5: If the following device information appears in the **Device Manager** after the operation in Step 4, user need to proceed to the next step.



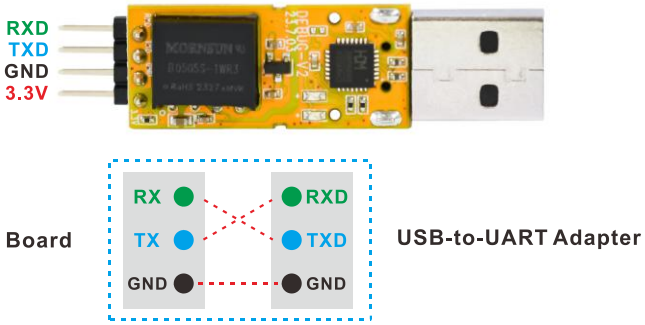
Step 6: The WINDOW will pop up found New Hardware Wizard dialog box, choose to install from the specified location, and then select:

DriverAssitant_v5.1.1/ADBDriver.

Step 7: After the installation is completed, the following device information can be seen in the Computer **Device Manager**.

2.2 Install CH9102X Driver

2.2.1 How to Connect the Serial Port Tool



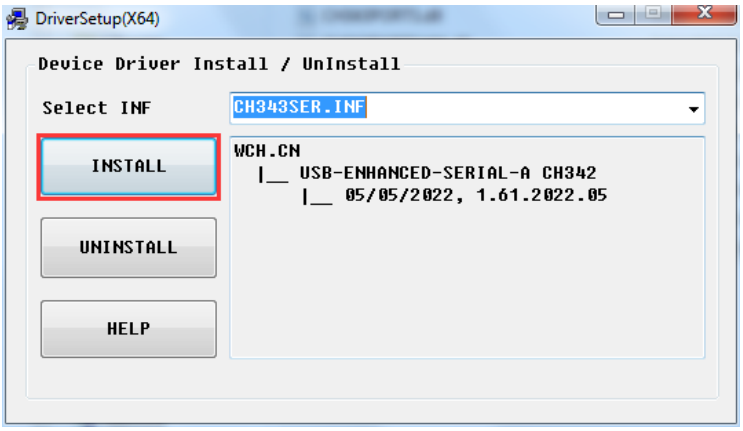
Pin	Connection Description
RXD	Receive, connect to TX pin of the board.
TXD	Transmit, connect to RX pin of the board.
GND	Ground, connect to GND pin of the board.
3V3	No need to connect.

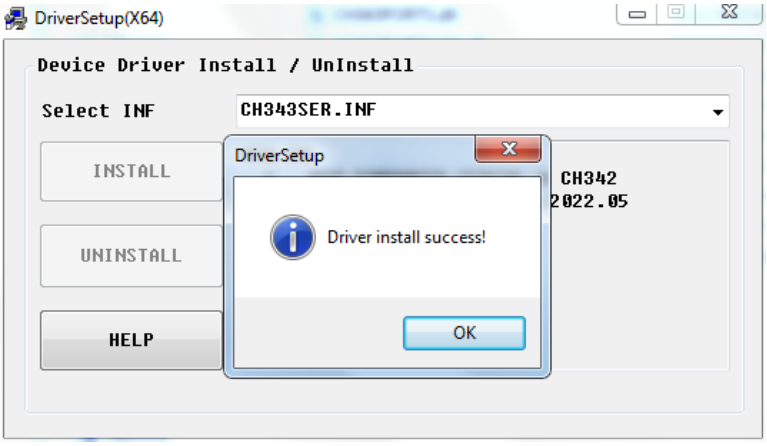
2.2.2 Install Driver

Step 1: Plug the CH9102X Module to the PC

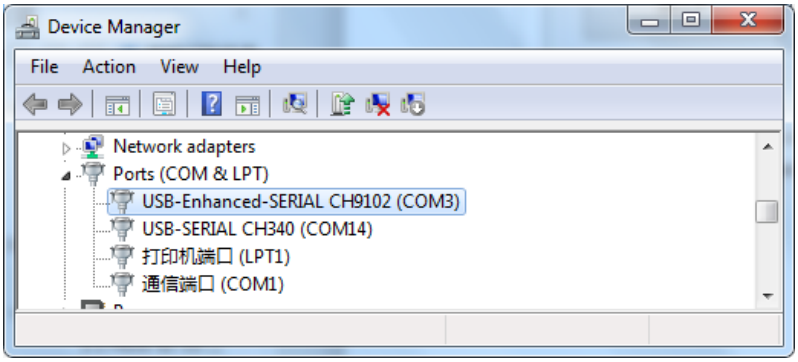
Step 2: Unzip *CH343SER.ZIP* on Windows.

Step 3: Select and install the corresponding *SETUP.EXE* according to the computer properties.





Step 4: After the installation is completed, the device will be listed under **Device Manager** ports with unique serial port assigned.

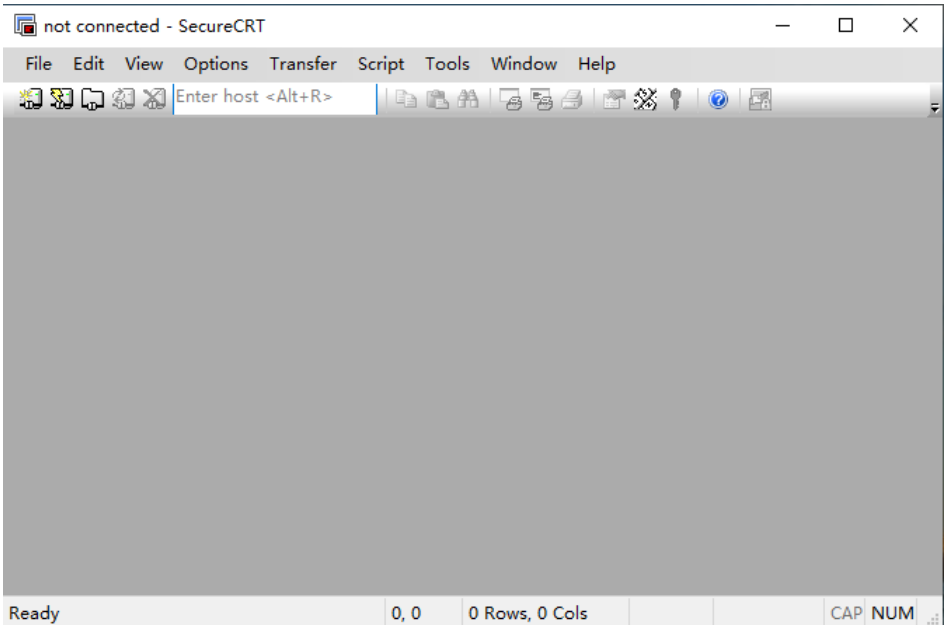


2.3 Install Serial Terminal Tool

The serial terminal SecureCRT is used for debugging in Windows. It can be used directly after decompression.

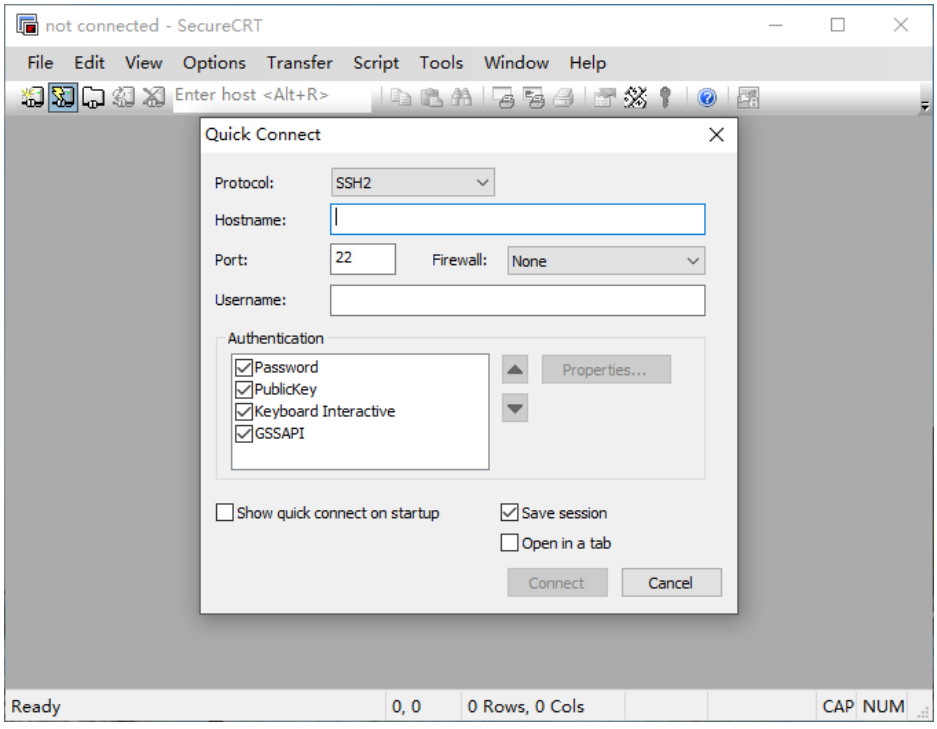
Step 1: Unzip *Platform/SecureCRT.rar* on PC.

Step 2: Click *SecureCRT/SecureCRT.exe* open the SecureCRT.

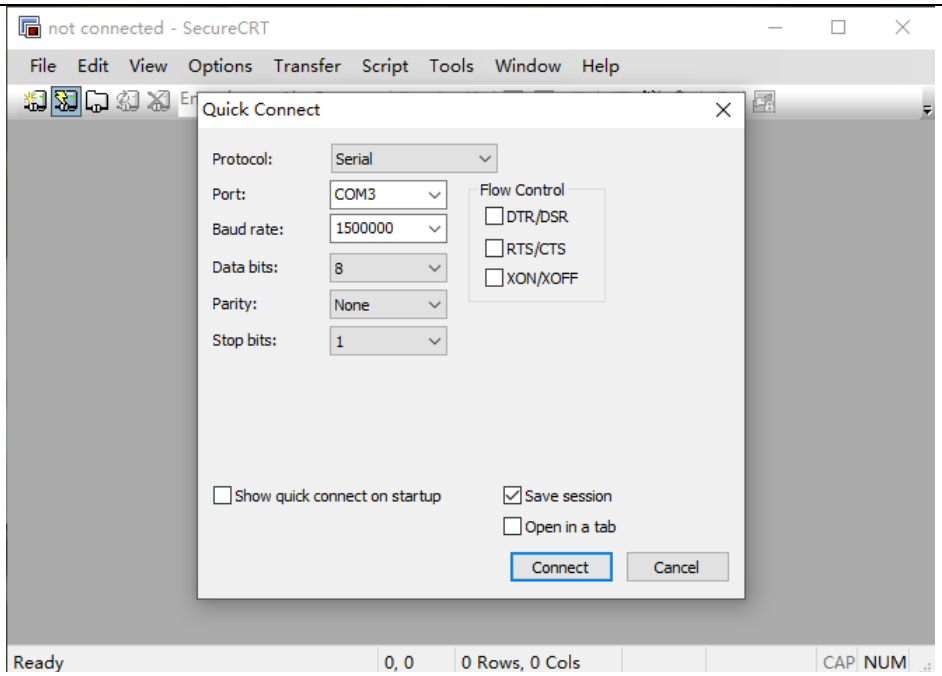


Step 3: Confirm that the CH9102X driver has been installed and the CH9102X module is connecting to the PC.

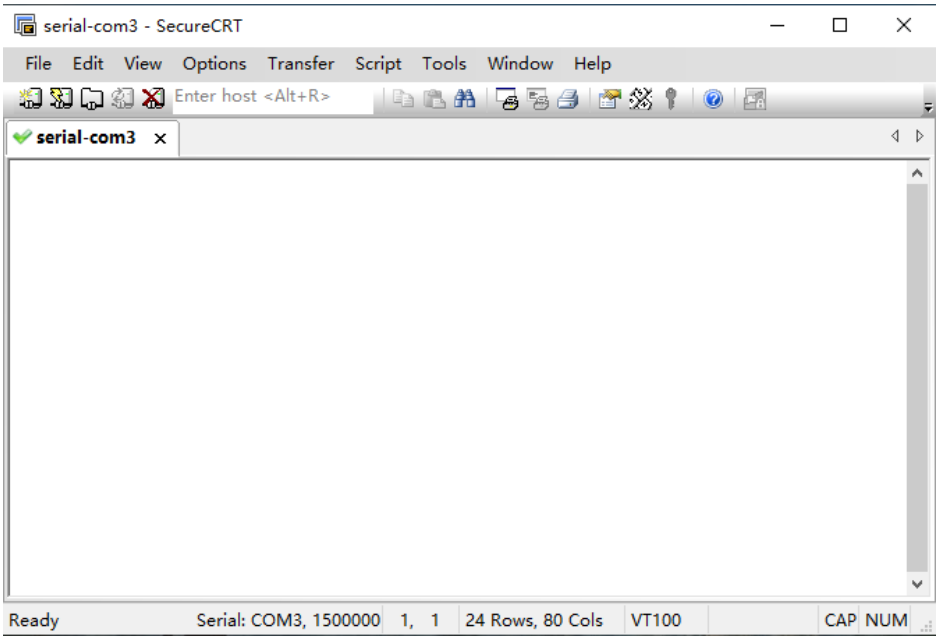
Step 4: Click the **"Quick Connect"** button to go to the Quick Connect configuration screen.



Step 5: Configure as shown in the following figure.



Step 6: After clicking **"Connect"** button, the terminal serial interface will be successfully accessed.



3.Upgrade Introduction

3.1 Upgrade Mode

The firmware can be upgraded via USB cable in two modes:

1. **Loader Mode:**

   The standard mode used for firmware upgrades.

2. **MaskRom Mode:**

   A last-resort mode used when the device is bricked. Entering MaskRom mode requires hardware manipulation, which involves certain risks. It should only be attempted if Loader mode is unavailable.

• Prerequisite

Before upgrading the firmware via USB cable, ensure that the necessary drivers are installed. For installation instructions, refer to the section Install RK Driver Assistant.

3.1.1 How to Enter Loader Mode

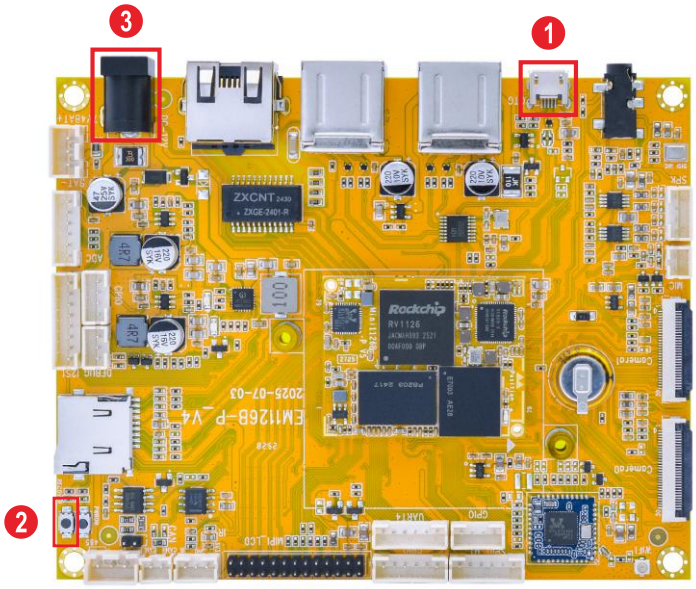
3.1.1.1 Hardware

Step 1: Disconnect the power adapter.

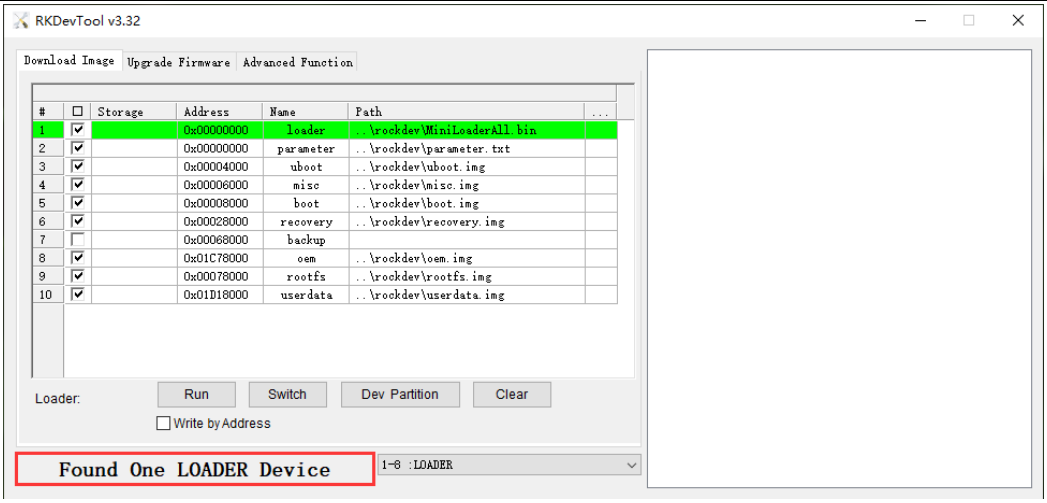
Step 2: Connect one end of the Micro USB cable to the host and the other end to the development board.

Step 3: Press and hold the **Recovery** button on the board

Step 4: Connect the power supply.



Step 5: After a few seconds, release the **Recovery** button when the flashing tool shows **"Found one LOADER Devic"**.



3.1.1.2 Software

After connecting the Micro USB cable, execute the following command in the serial debug terminal or adb shell.

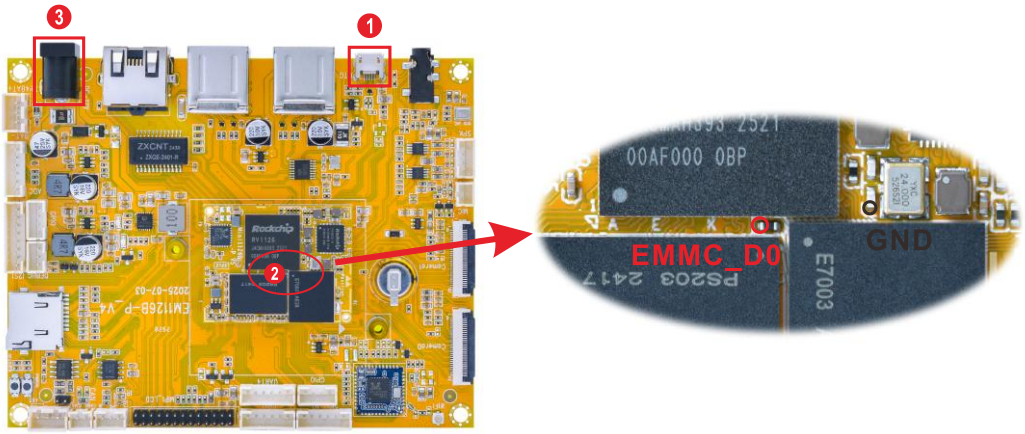
```
# reboot loader
```

3.1.2 How to Enter MaskRom Mode

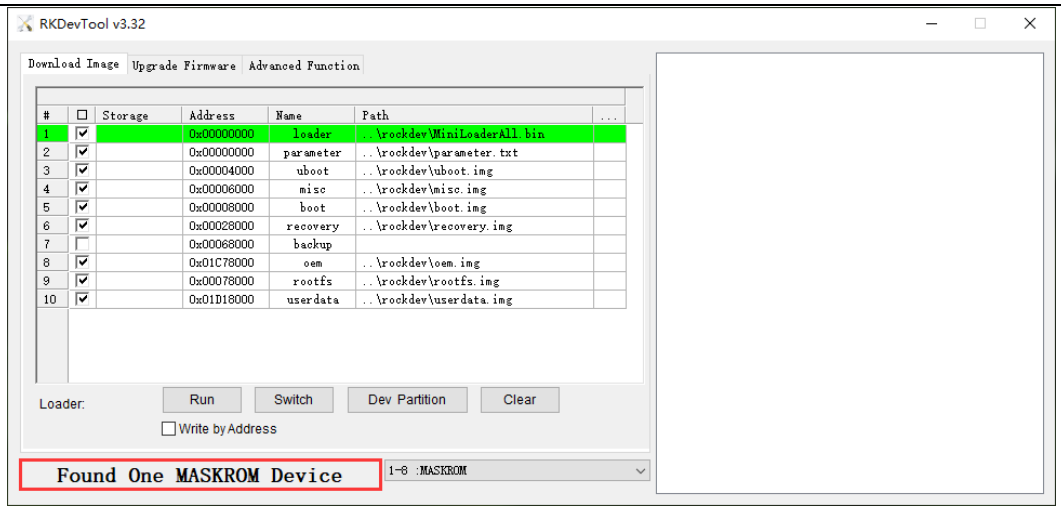
Step 1: Disconnect the power adapter.

Step 2: Connect one end of the Micro USB cable to the host and the other end to the development board.

Step 3: Use tweezers to short the two test points on the Mini1126B-P.



Step 4: After connecting the power cable, the device will enter MaskRom mode.



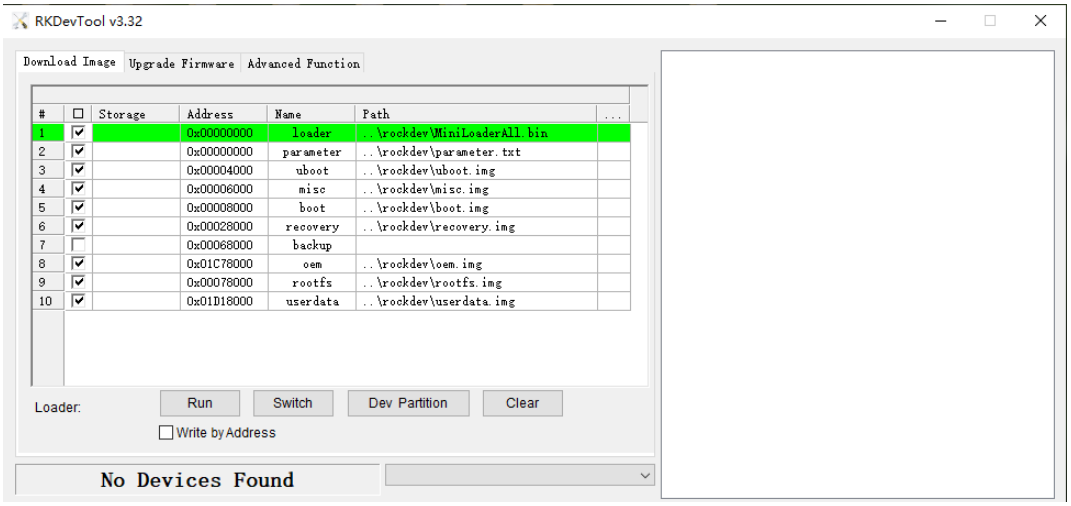
3.2 Burn firmware

Environment: Windows OS (Operating System).

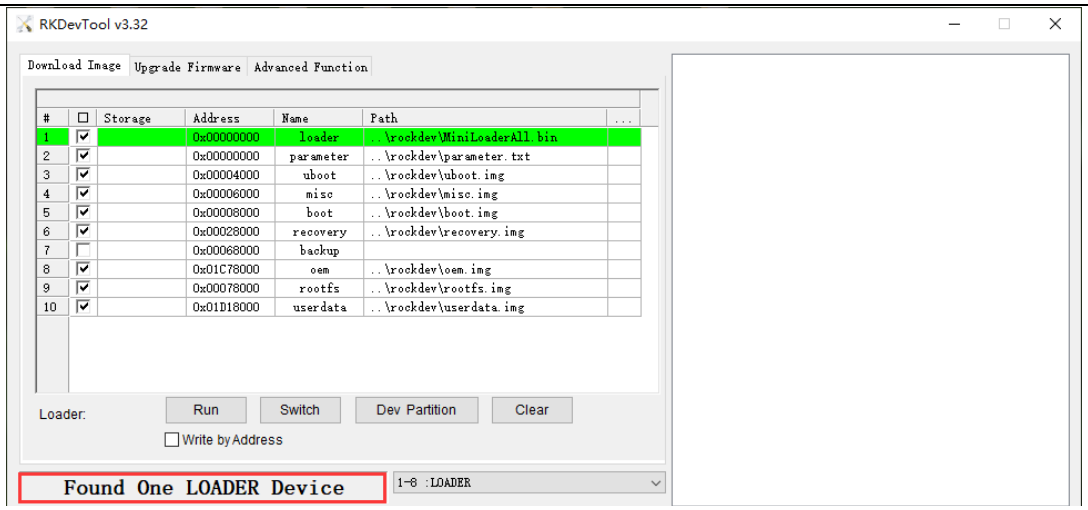
3.2.1 Burn Update.img Firmware

Step 1: Unzip *RKDevTool.rar* on Windows.

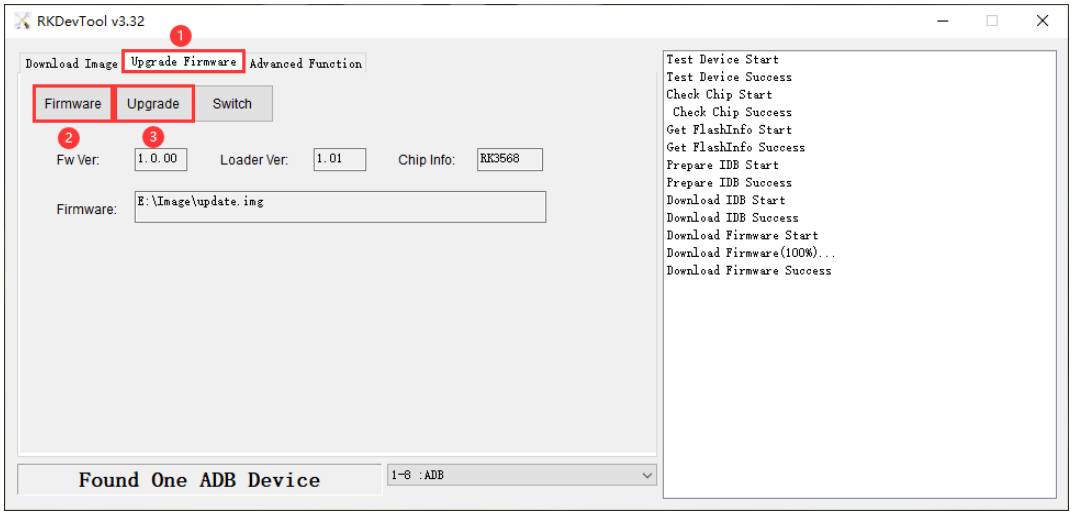
Step 2: Open *RKDevTool\RKDevTool_Release\RKDevTool.exe.*



Step 3: Switch to **loader mode**. (*How to Enter Loader Mode*)



Step 4: Click **Upgrade Firmware** -> **Firmware**, select **update.img**, then click **Upgrade** to flash.



After the flashing is complete, the board will automatically reboot.

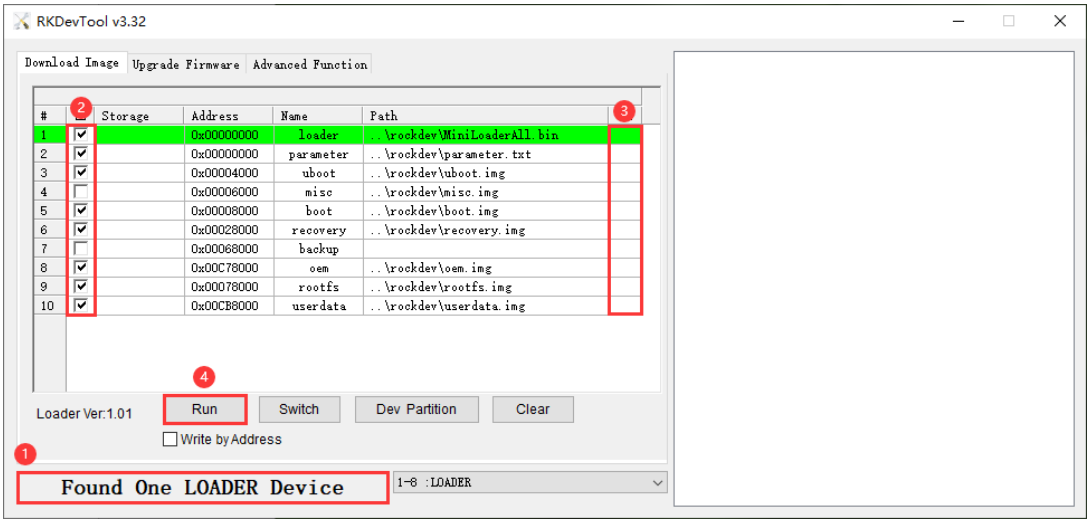
3.2.2 Burn Split Firmware

Step 1: Switch to **Loader mode**.

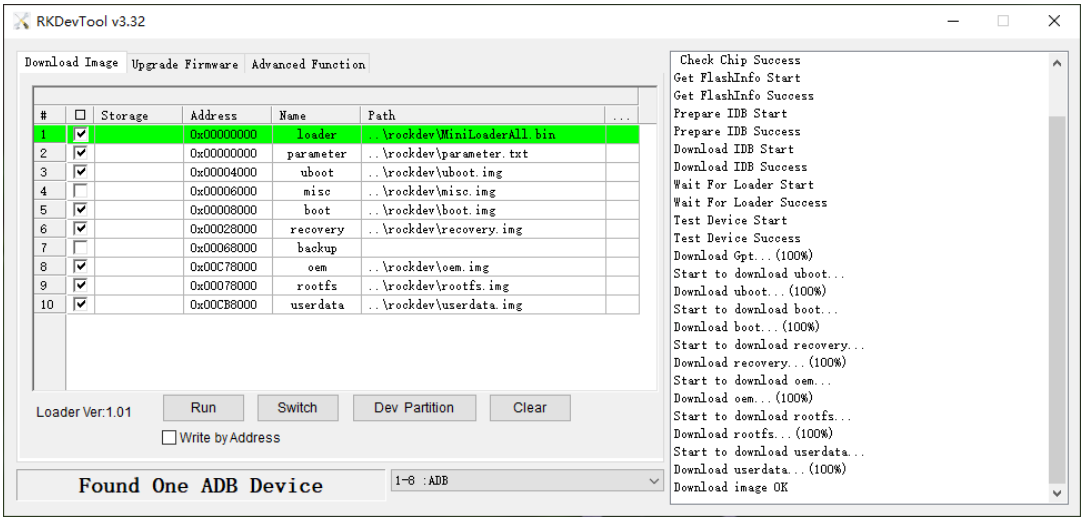
Step 2: Check the partitions to be flashed, multiple partitions can be selected.

Step 3: Ensure the image file path is correct. If necessary, click the blank cell next to the path to reselect it.

Step 4: Click the **Run** button to flash the image.



After the flashing is complete, the board will automatically reboot.



4.Development Environment

4.1 Preparing the Development Environment

It is recommended to use Ubuntu 22.04 or higher version for compilation. If you encounter an error during compilation, user can check the error message and install the corresponding software packages accordingly. Other Linux versions may need to adjust the software package accordingly. In addition to the system requirements, there are other hardware and software requirements.

Hardware requirements	Software requirements
64-bit system, hard disk space should be greater than 200G. If you do multiple builds, you will need more hard drive space.	Ubuntu 22.04

4.2 Installing Libraries and Toolkits

The contents of this directory only provide the software package installation commands that are needed to build the compiled SDK environment. Please install other tools such as samba and ssh yourself.

PC OS	Network	Permission
Ubuntu 22.04	online	root

To install the required tools, execute the following commands:

```
$ sudo apt-get install git ssh make gcc libssl-dev liblz4-tool libmpc-dev
$ sudo apt-get install expect g++ patchelf chrpath gawk texinfo chrpath diffstat
$ sudo apt-get install binfmt-support live-build bison flex fakeroot libgmp-dev
$ sudo apt-get install cmake gcc-multilib g++-multilib unzip device-tree-compiler
$ sudo apt-get install ncurses-dev libgucharmap-2-90-dev bzip2 expat gpgv2
$ sudo apt-get install cpp-aarch64-linux-gnu g++-aarch64-linux-gnu
$ sudo apt install python2 python-is-python3
```

5.Compile Source

Step 1: Unzip the Source

To extract the source files, execute the following commands:

```
$ tar xvf EM1126BP-linux6.1*.tar.gz
$ cd EM1126BP-linux6.1
```

Step 2: Configure the Compiled Board

To configure the board, execute:

```
$ ./build.sh lunch
```

Select Board Configuration After executing ./build.sh lunch, the system will list available defconfig files.

```
$ ./build.sh lunch
############### Rockchip Linux SDK ###############
Manifest: rv1126b_linux6.1_release_v1.2.0_20251220.xml
GIT commit: "4236b4b boardcon: fix IPC defconfig to use em1126bp_ipc buildroot config"
Log colors: message notice warning error fatal
Log saved at /home/chenyu/RV1126bp/RV1126B_LINUX6.1_V1.2.0/output/sessions/2026-01-21_16-41-16
Pick a defconfig:
1. rockchip_defconfig
2. boardcon_em1126bp_defconfig
3. boardcon_em1126bp_ipc_defconfig
4. boardcon_idea1126bp_defconfig
5. boardcon_idea1126bp_ipc_defconfig
6. rockchip_rv1126b_dv_64_evb1_v10_defconfig
7. rockchip_rv1126b_dv_64_evb1_v12_defconfig
8. rockchip_rv1126b_evb1_v10_defconfig
9. rockchip_rv1126b_evb4_v10_defconfig
10. rockchip_rv1126b_fastboot_defconfig
11. rockchip_rv1126b_ipc_32_evb1_v10_defconfig
12. rockchip_rv1126b_ipc_64_evb1_v10_defconfig
13. rockchip_rv1126b_robot_defconfig
14. rockchip_rv1126b_tiny_32_evb2_v10_defconfig
15. rockchip_rv1126bp_evb1_v10_defconfig
16. rockchip_rv1126bp_fastboot_defconfig
17. rockchip_rv1126bp_ipc_32_evb1_v10_defconfig
18. rockchip_rv1126bp_ipc_64_evb1_v10_defconfig
19. rockchip_rv1126bp_robot_defconfig
Which would you like? [1]:
```

Choose the Correct Configuration For the EM1126BP development board, please input 2 or 3 based on your application requirements:

Option 2: General-Purpose Board Configuration (boardcon_em1126bp_defconfig)

① Framework: Based on GStreamer (Open-source, high compatibility).

② Default Behavior: Automatically launches the Weston desktop environment.

③ Use Case: Best for general multimedia applications requiring plugin extensibility.

Option 3: IPC (IP Camera) Board Configuration (boardcon_em1126bp_ipc_defconfig)

① Framework: Utilizes Rockit MPI (Rockchip proprietary). Provides modular hardware acceleration (VI, VO, VDEC, VENC, VPSS).

② Default Behavior: Automatically launches the RKIPC daemon (RTSP/RTMP streaming & Web UI).

③ Use Case: Best for dedicated IP Camera products requiring optimized hardware acceleration.

Warning: Please ensure you select either option 2 or 3. These configurations are specifically tuned for the EM1126BP hardware. Selecting other options (or generic Rockchip configs) may cause kernel mismatches and prevent the system from booting properly.

Step 3: Compile U-Boot

To compile uboot, execute the following command:

```
$ ./build.sh uboot
```

Step 4: Compile Kernel

To compile kernel, execute the following command:

```
$ ./build.sh kernel
```

Step 5: Compile Recovery

To compile recovery, execute the following command:

```
$ ./build.sh recovery
```

Step 6: Compile rootfs

(1) Compile Buildroot

To compile buildroot, execute the following command:

```
$ ./build.sh buildroot
```

(2)Compile Debian12 (Permission: root)

To compile debian12, execute the following command:

```
$ sudo ./build.sh debian
```

After compilation, a linaro-rootfs.img is generated in the *debian/* directory.

Step 7: Generate and Check Firmwares

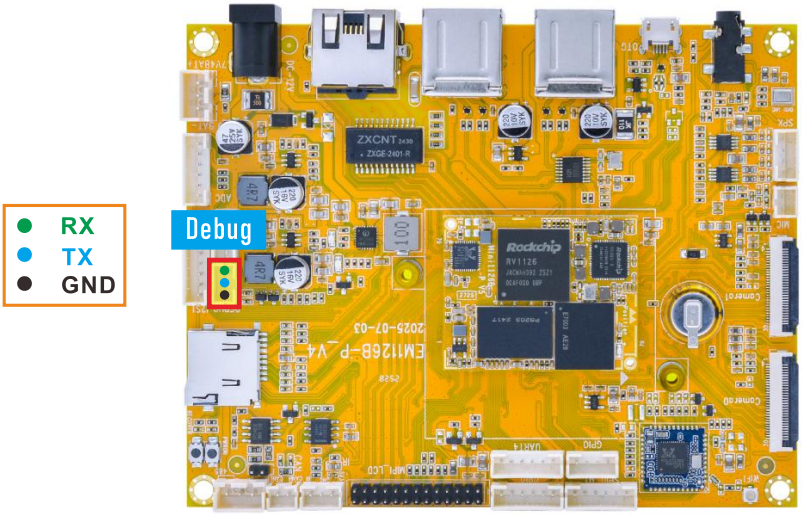
To generate firmware, execute the following command:

```
$ ./build.sh firmware
$ ./build.sh updateimg
```

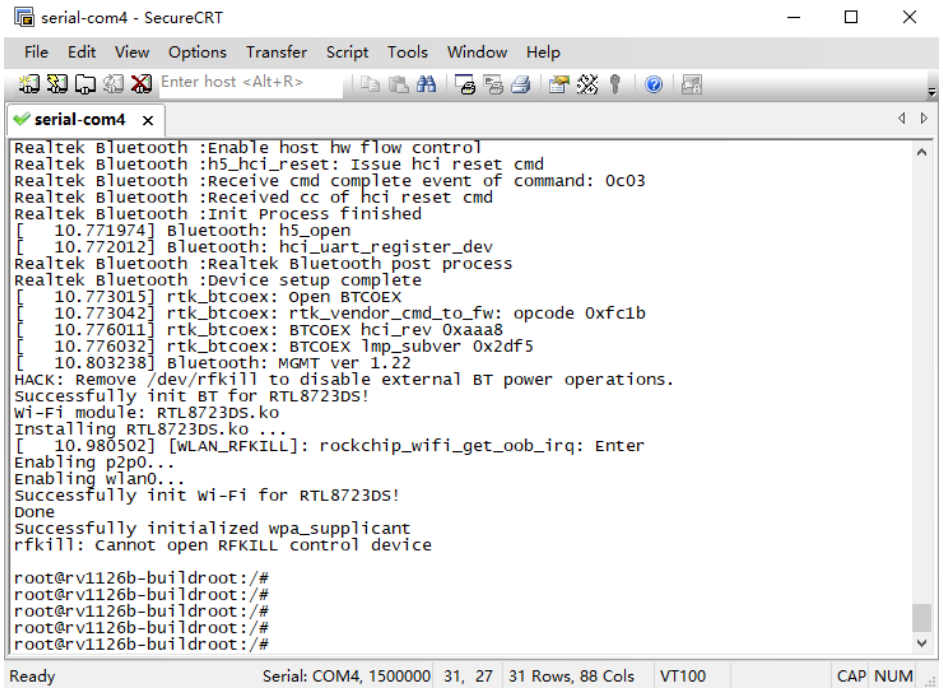
Images and **update.img** are generated in *rockdev/* directory.

6.Buildroot Test

6.1 Serial Terminal

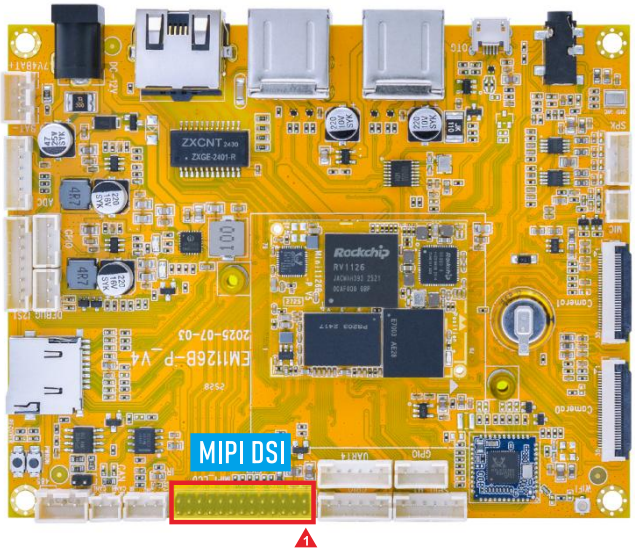


Connect the board and PC with USB Serial cable, then power on, the terminal will output boot information. The default baudrate is 1500000.



6.2 Display

The default MIPI display resolution on EM1126B-P is 800×1280 @60Hz.



The display effect diagram is as follows:

6.3 USB

This system provides one USB2.0 OTG interface and one USB2.0 Host interface. Both support connection to common USB peripherals and development tools.

6.3.1 USB2.0 Host

The USB2.0 Host port can be used to connect USB peripherals such as a mouse, keyboard, flash drive, and other USB devices.

• When a USB flash drive is inserted, the system will automatically mount it.

• Check the mounted directory using the following commands:

```
# df -h
```

```
root@rv1126b-buildroot:/# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       6.0G  421M  5.3G   8% /
devtmpfs        960M     0  960M   0% /dev
tmpfs           987M  136K  987M   1% /tmp
tmpfs           987M  544K  987M   1% /run
tmpfs           987M  164K  987M   1% /var/log
tmpfs           987M     0  987M   0% /dev/shm
/dev/mmcblk0p8  881M  292K  864M   1% /userdata
/dev/mmcblk0p7  123M   12M  109M  10% /oem
/dev/sda1        58G   39G   20G  67% /mnt/udisk
/dev/sdb2       448M  327M  121M  74% /media/udisk1
/dev/sdb1        28G   24G  4.7G  84% /media/udisk2
```

6.3.2 USB OTG(ADB)

By default, the OTG port operates in **Device mode**, allowing ADB connections for debugging.

To enable ADB on a Windows host:

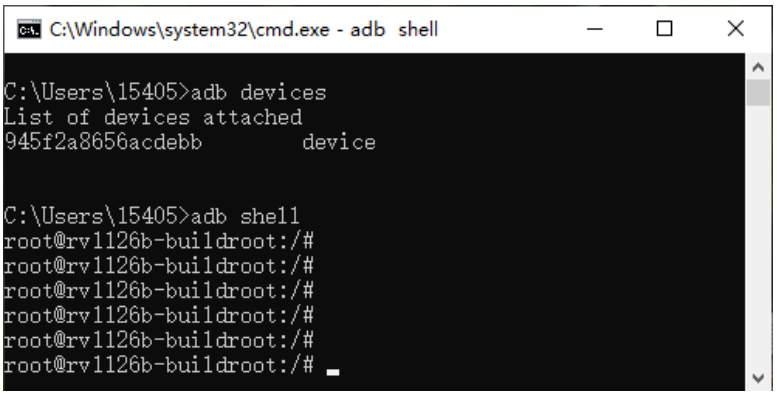
Step 1: Connect the board and PC host with Micro USB cable.

Step 2: Install ADB driver on Windows system.

Step 3: Press **Windows + R**, type cmd, and press **Enter** to open the command prompt

Step 4: Run the following command to check ADB connection.

```
# adb shell
```



6.4 Ethernet

Step 1: Connect the network cable to the Ethernet port.

According to the log, it can be seen that the Gigabit Ethernet recognition is successful.

```
root@rv1126b-buildroot:/# [  397.838650] rk_gmac-dwmac 21c70000.ethernet eth0: Link is Up - 1Gbps/Full - flow
control rx/tx
[  397.838728] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
```

Step 2: View network interface information.

```
# ifconfig
```

```
root@rv1126b-buildroot:/# ifconfig
eth0      Link encap:Ethernet  HWaddr FA:B2:1C:0E:F2:D4
          inet addr:192.168.0.124  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::8f56:d77d:6c3a:38a2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:85 errors:0 dropped:16 overruns:0 frame:0
          TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7960 (7.7 KiB)  TX bytes:1410 (1.3 KiB)
          Interrupt:74
```

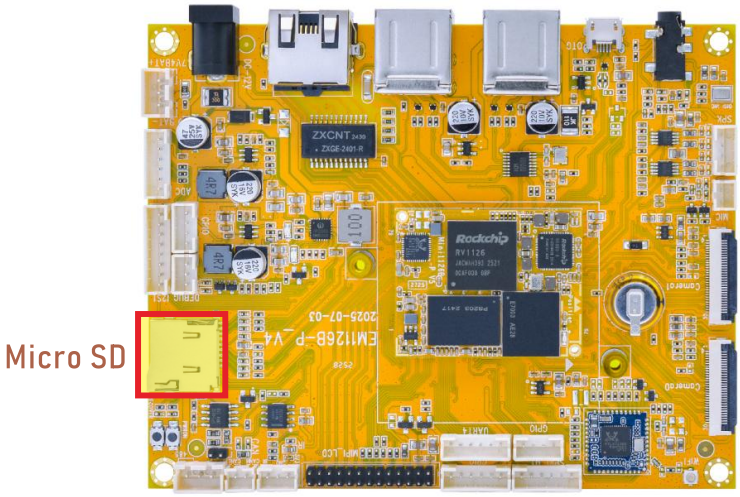
Step 3: Network connection test.

```
# ping -I eth0 www.armdesigner.com
```

```
root@rv1126b-buildroot:/# ping -I eth0 www.armdesigner.com
PING www.armdesigner.com (67.222.54.196) from 192.168.0.124 eth0: 56(84) bytes of data.
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=1 ttl=50 time=196 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=2 ttl=50 time=191 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=3 ttl=50 time=194 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=4 ttl=50 time=193 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=5 ttl=50 time=191 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=6 ttl=50 time=194 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=7 ttl=50 time=191 ms
^C
--- www.armdesigner.com ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6008ms
rtt min/avg/max/mdev = 190.700/192.683/195.603/1.853 ms
```

6.5 SD Card

Step 1: Insert the micro SD card into the card slot.



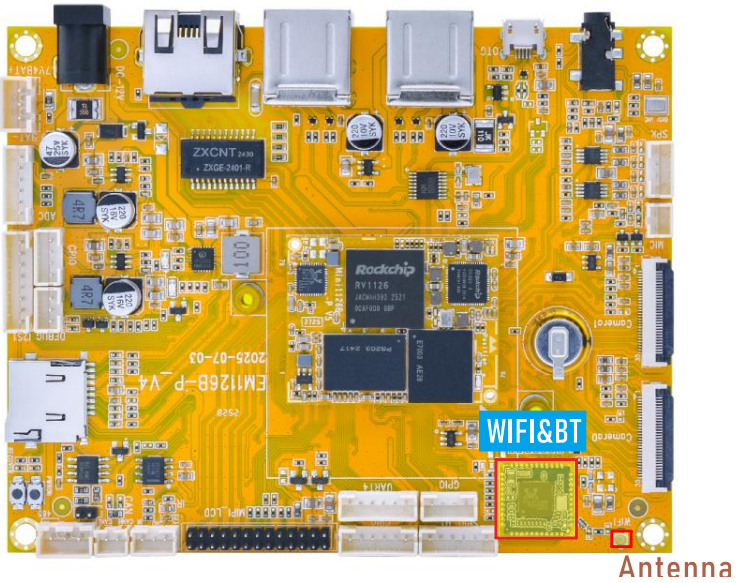
Step 2: The system will automatically mount it, view the device mount path.

```
# df -h
```

```
root@rv1126b-buildroot:/# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       6.0G  421M  5.3G   8% /
devtmpfs        960M     0  960M   0% /dev
tmpfs           987M  136K  987M   1% /tmp
tmpfs           987M  556K  987M   1% /run
tmpfs           987M  172K  987M   1% /var/log
tmpfs           987M     0  987M   0% /dev/shm
/dev/mmcblk0p8  881M  292K  864M   1% /userdata
/dev/mmcblk0p7  123M   12M  109M  10% /oem
/dev/sda1        58G   39G   20G  67% /mnt/udisk
/dev/sdb2       448M  327M  121M  74% /media/udisk1
/dev/sdb1        28G   24G  4.7G  84% /media/udisk2
/dev/mmcblk1p1  120G   85M  119G   1% /mnt/sdcard
```

6.6 WiFi & Bluetooth

To use Wi-Fi and Bluetooth functions properly, the antenna needs to be connected.



6.6.1 WiFi

Step 1: View the device information.

```
# ifconfig
```

```
root@rv1126b-buildroot:/# ifconfig
p2p0      Link encap:Ethernet  HWaddr AA:B5:8E:B9:3B:9A
          UP BROADCAST 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)

wlan0     Link encap:Ethernet  HWaddr A8:B5:8E:B9:3B:9A
          UP BROADCAST 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)
```

Step 2: Scan for available WiFi hotspots.

```
# iwlist wlan0 scan
```

```
root@rv1126b-buildroot:/# iwlist wlan0 scan
wlan0     Scan completed :
          Cell 01 - Address: B4:F1:8C:6D:D1:25
                    ESSID:""
                    Protocol:IEEE 802.11bgn
                    Mode:Master
                    Frequency:2.437 GHz (Channel 6)
                    Encryption key:off
                    Bit Rates:300 Mb/s
                    IE: Unknown: DD230050F204104A000110104400010210080002078010 3C000101104900060037 2A000120
                    Quality=80/100  Signal level=44/100
                    Extra:fm=0002
          Cell 02 - Address: B4:F1:8C:FD:D1:29
                    ESSID:"Boardcon_Wi-Fi5"
                    Protocol:IEEE 802.11bgn
                    Mode:Master
                    Frequency:2.437 GHz (Channel 6)
                    Encryption key:on
                    Bit Rates:300 Mb/s
                    Extra:wpa_ie=dd1a0050f20101000050f20202000050f2040050f20201000050f202
                    IE: WPA Version 1
                        Group Cipher : TKIP
                        Pairwise Ciphers (2) : CCMP TKIP
                        Authentication Suites (1) : PSK
                    Extra:
                    IE: IEEE 802.11i/WPA2 Version 1
                        Group Cipher : TKIP
                        Pairwise Ciphers (2) : CCMP TKIP
                        Authentication Suites (1) : PSK
                    IE: Unknown:
DD910050F204104A000110104400010210 3B000103104700101234567 89ABCDEF01234B4F18C6DD124102100194875617765692054
6563686E6F6C6F677920436F2E2C4C7464102300 0B576972656C6573732041501024000331323310420005313233435105400080 0
060050F204000110110009544337313032 2D313010080002078010 3C000101104900060037 2A000120
                    Quality=95/100  Signal level=43/100
                    Extra:fm=0003
          Cell 03 - Address: B4:F1:8C:6D:D1:29
                    ESSID:""
                    Protocol:IEEE 802.11bgn
```

Step 3: Connect to the hotspot.

```
#  wifi-connect.sh SSID PSK
```

```
root@rv1126b-buildroot:/# wifi-connect.sh Boardcon Boardcon43435656
connect to WiFi ssid: Boardcon, Passwd: Boardcon43435656
Successfully initialized wpa_supplicant
rfkill: Cannot open RFKILL control device
 [10486.879381] IPv6: ADDRCONF(NETDEV_CHANGE): wlan0: link becomes ready
```

Step 4: View the network interface status.

```
# ifconfig
```

```
root@rv1126b-buildroot:/# ifconfig
p2p0      Link encap:Ethernet  HWaddr AA:B5:8E:B9:3B:9A
          UP BROADCAST 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)

wlan0     Link encap:Ethernet  HWaddr A8:B5:8E:B9:3B:9A
          inet addr:192.168.0.46  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::3ffc:4a8:f34d:9023/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:148 errors:0 dropped:22 overruns:0 frame:0
          TX packets:18 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:18923 (18.4 KiB)  TX bytes:2592 (2.5 KiB)
```

Step 5: Test the WiFi network.

```
# ping -I wlan0 www.armdesigner.com
```

```
root@rv1126b-buildroot:/# ping -I wlan0 www.armdesigner.com
PING www.armdesigner.com (67.222.54.196) from 192.168.0.46 wlan0: 56(84) bytes of data.
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=1 ttl=50 time=420 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=2 ttl=50 time=211 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=3 ttl=50 time=409 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=4 ttl=50 time=221 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=5 ttl=50 time=248 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=7 ttl=50 time=250 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=8 ttl=50 time=206 ms
^C
--- www.armdesigner.com ping statistics ---
8 packets transmitted, 7 received, 12.5% packet loss, time 7029ms
rtt min/avg/max/mdev = 206.037/280.618/420.143/86.123 ms
```

6.6.2 Bluetooth

On Buildroot, Bluetooth can be configured to function as a Bluetooth speaker.

Step 1: Select audio output (2=Speaker, 3=Headset).

Headset example:

```
# amixer -c 0 cset numid=37 3
```

```
root@rv1126b-buildroot:/# amixer -c 0 cset numid=37 3
numid=37,iface=MIXER,name='Playback Path'
  ; type=ENUMERATED,access=rw------,values=1,items=11
  ; Item #0 'OFF'
  ; Item #1 'RCV'
  ; Item #2 'SPK'
  ; Item #3 'HP'
  ; Item #4 'HP_NO_MIC'
  ; Item #5 'BT'
  ; Item #6 'SPK_HP'
  ; Item #7 'RING_SPK'
  ; Item #8 'RING_HP'
  ; Item #9 'RING_HP_NO_MIC'
  ; Item #10 'RING_SPK_HP'
  : values=3
```

Step 2: Start the BlueALSA service in **A2DP Sink** mode.

```
# bluealsa -p a2dp-sink &
```

```
root@rv1126b-buildroot:/# bluealsa -p a2dp-sink &
[1] 1892
root@rv1126b-buildroot:/# bluealsa: [1892] D: storage.c:84: Initializing persistent storage: /var/lib/bluealsa
bluealsa: [1892] D: main.c:604: Starting main dispatching loop
bluealsa: [1892] D: main.c:114: Acquired D-Bus service name: org.bluealsa
bluealsa: [1892] D: bluealsa-dbus.c:392: Registering D-Bus manager: /org/bluealsa
[ 2407.009711] Bluetooth: hu 000000001493c12c retransmitting 1 pkts
[ 2407.010857] rtk_btcoex: BTCOEX hci_rev 0xaaa8
[ 2407.010883] rtk_btcoex: BTCOEX lmp_subver 0x2df5
bluealsa: [1892] D: bluez.c:783: Registering battery provider: /org/bluez/hci0/battery
bluealsa: [1892] D: bluez.c:598: Creating media endpoint object: /org/bluez/hci0/A2DP/SBC/sink/1
bluealsa: [1892] D: bluez.c:509: Registering media endpoint: /org/bluez/hci0/A2DP/SBC/sink/1
bluealsa: [1892] D: bluez.c:598: Creating media endpoint object: /org/bluez/hci0/A2DP/SBC/sink/2
bluealsa: [1892] D: bluez.c:509: Registering media endpoint: /org/bluez/hci0/A2DP/SBC/sink/2
```

Step 3: Scan for the Bluetooth device address.

```
# hcitool scan
```

```
root@rv1126b-buildroot:/# hcitool scan
Scanning ...
[11524.399998] rtk_btcoex: hci (periodic)inq start
[11524.649082] Bluetooth: hu 00000000fc1e59f2 retransmitting 1 pkts
[11534.893160] rtk_btcoex: inquiry complete
        28:D0:43:39:FE:0E       HUANG
        A8:35:12:9A:EB:4D       liuy
        2C:A0:42:D1:14:D5       Kang
```

Step 4: Start audio playback via BlueALSA (route A2DP to ALSA).

```
# bluealsa-aplay A8:35:12:9A:EB:4D &
```

```
root@rv1126b-buildroot:/# bluealsa-aplay A8:35:12:9A:EB:4D &
[2] 1890
root@rv1126b-buildroot:/# bluealsa-aplay: [1890] D: aplay.c:1194: Starting main loop
```

Step 5: Set the Bluetooth adapter to be discoverable.

```
# hciconfig hci0 piscan
```

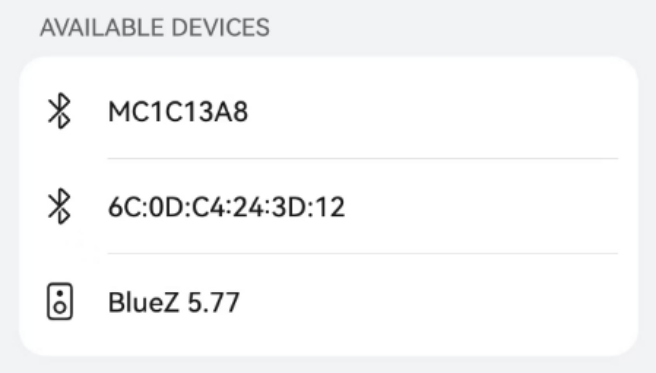
```
root@rv1126b-buildroot:/# hciconfig hci0 piscan
```

Step 6: Control and configure the Bluetooth device.

```
# bluetoothctl
```

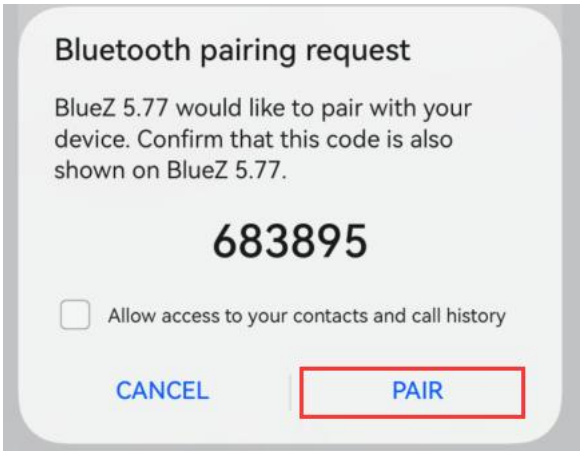
```
root@rv1126b-buildroot:/# bluetoothctl
hci0 new_settings: powered connectable discoverable bondable ssp br/edr le secure-conn
Agent registered
[CHG] Controller A8:B5:8E:B9:3B:9B Pairable: yes
[bluetooth]#
```

Step 7: On the phone, locate the device name of the speaker: **BlueZ 5.77**, and click to connect.



Step 8: Permissions must be confirmed on both the phone and the speaker.

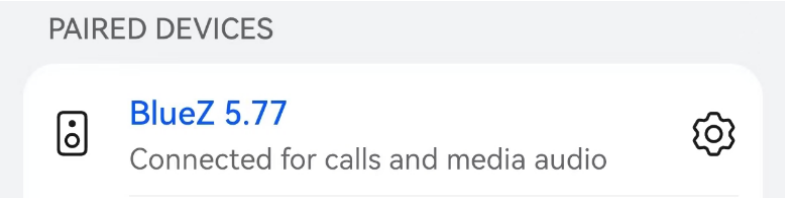
Phone:



Bluetooth speaker:

```
root@rv1126b-buildroot:/# bluetoothctl
hci0 new_settings: powered connectable discoverable bondable ssp br/edr le secure-conn
Agent registered
[CHG] Controller A8:B5:8E:B9:3B:9B Pairable: yes
[bluetooth]# [  594.070562] rtk_btcoex: hci accept conn req
[  594.287289] rtk_btcoex: connected, handle 0004, status 0x00
[  594.287332] rtk_btcoex: Page success
hci0 A8:35:12:9A:EB:4D type BR/EDR connected eir_len 5
hci0 A8:35:12:9A:EB:4D type BR/EDR connected eir_len 5
[bluetooth]# bluealsa: [1883] D: bluez.c:1199: Signal: org.freedesktop.DBus.ObjectManager.InterfacesAdded()
[NEW] Device A8:35:12:9A:EB:4D A8-35-12-9A-EB-4D
[A8-35-12-9A-EB-4D]# [  594.378463] rtk_btcoex: io capability request
hci0 A8:35:12:9A:EB:4D type BR/EDR connected eir_len 11
[CHG] Device A8:35:12:9A:EB:4D Name: liuy
[CHG] Device A8:35:12:9A:EB:4D Alias: liuy
hci0 A8:35:12:9A:EB:4D type BR/EDR connected eir_len 5
hci0 A8:35:12:9A:EB:4D type BR/EDR connected eir_len 5
hci0 A8:35:12:9A:EB:4D type BR/EDR connected eir_len 5
hci0 A8:35:12:9A:EB:4D type BR/EDR connected eir_len 5
Request confirmation
[agent] Confirm passkey 630905 (yes/no): yes
[A8-35-12-9A-EB-4D]# [  598.188357] rtk_btcoex: link key notify
hci0 new_link_key A8:35:12:9A:EB:4D type 0x05 pin_len 0 store_hint 1
hci0 device_flags_changed: A8:35:12:9A:EB:4D (BR/EDR)
     supp: 0x00000000  curr: 0x00000000
[CHG] Device A8:35:12:9A:EB:4D Bonded: yes
[A8-35-12-9A-EB-4D]# [  598.248646] rtk_btcoex: l2cap op 2, len 16, out 0
[  598.248742] rtk_btcoex: RX l2cap conn req, hndl 0x0004, PSM 0x0001, scid 0x005f
hci0 A8:35:12:9A:EB:4D type BR/EDR connected eir_len 5
[  598.248762] rtk_btcoex: PSM(0x0001) do not need parse
```

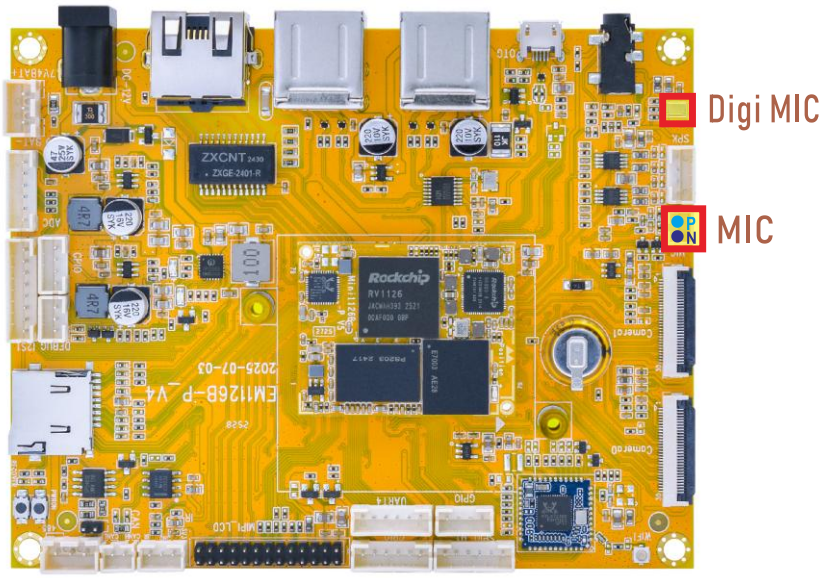
Step 9: The connection is successful.



6.7 Audio

6.7.1 Audio Input

Step 1: Connect the MIC.



Step 2: View sound card.

```
# cat /proc/asound/cards
```

```
root@rv1126b-buildroot:/# cat /proc/asound/cards
 0 [rockchiprk809co]: simple-card - rockchip,rk809-codec
                      rockchip,rk809-codec
 1 [rockchippdmmica]: simple-card - rockchip,pdm-mic-array
                      rockchip,pdm-mic-array
```

Step 3: Record audio.

• Using the Analog MIC.

```
# amixer -c 0 cset numid=38 1
# arecord -Dhw:0,0 -f cd -t wav test.wav
```

```
root@rv1126b-buildroot:/# amixer -c 0 cset numid=38 1
numid=38,iface=MIXER,name='Capture MIC Path'
  ; type=ENUMERATED,access=rw------,values=1,items=4
  ; Item #0 'MIC OFF'
  ; Item #1 'Main Mic'
  ; Item #2 'Hands Free Mic'
  ; Item #3 'BT Sco Mic'
  : values=1
root@rv1126b-buildroot:/# arecord -Dhw:0,0 -f cd -t wav test.wav
Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

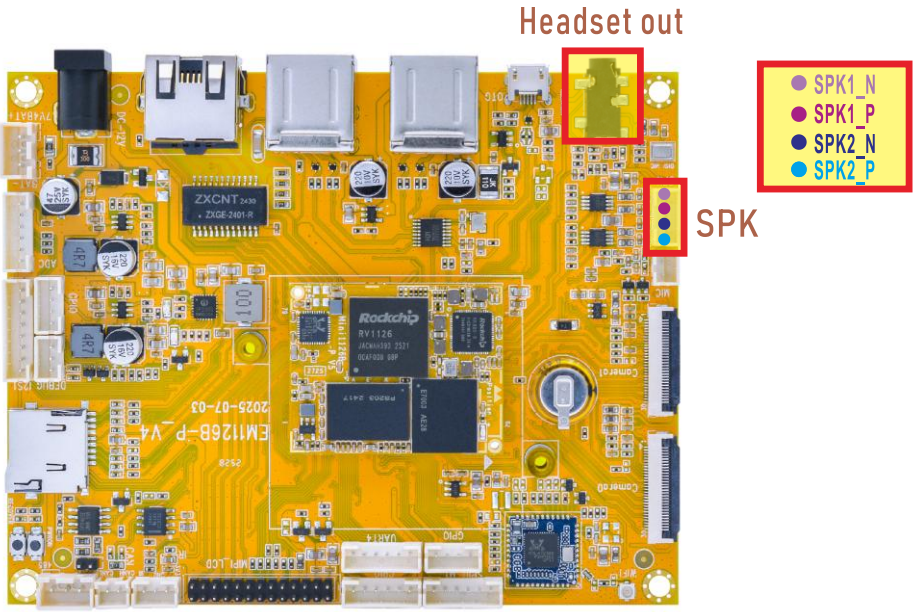
• Using the Digi MIC (PDM).

```
# arecord -D hw:1,0 -f s16_le -c 2 -r 16000 cap.wav
```

```
root@rv1126b-buildroot:/# arecord -D hw:1,0 -f s16_le -c 2 -r 16000 cap.wav
Recording WAVE 'cap.wav' : Signed 16 bit Little Endian, Rate 16000 Hz, Stereo
```

6.7.1 Audio Output

Step 1: Plug in the headset and connect the speaker.



Step 3: Play audio.

• Through the headset.

```
# amixer -c 0 cset numid=37 3
```

```
root@rv1126b-buildroot:/# amixer -c 0 cset numid=37 3
numid=37,iface=MIXER,name='Playback Path'
  ; type=ENUMERATED,access=rw------,values=1,items=11
  ; Item #0 'OFF'
  ; Item #1 'RCV'
  ; Item #2 'SPK'
  ; Item #3 'HP'
  ; Item #4 'HP_NO_MIC'
  ; Item #5 'BT'
  ; Item #6 'SPK_HP'
  ; Item #7 'RING_SPK'
  ; Item #8 'RING_HP'
  ; Item #9 'RING_HP_NO_MIC'
  ; Item #10 'RING_SPK_HP'
  : values=3
```

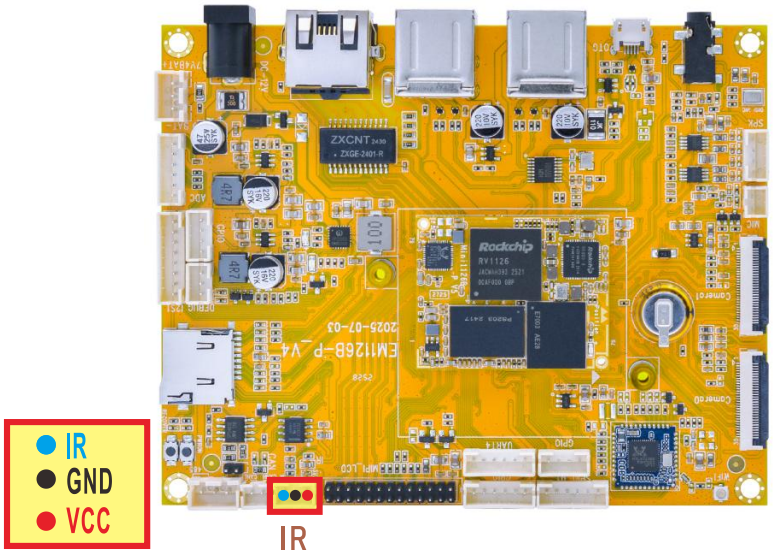
• Through the speaker.

```
# amixer -c 0 cset numid=37 2
```

```
root@rv1126b-buildroot:/# amixer -c 0 cset numid=37 2
numid=37,iface=MIXER,name='Playback Path'
  ; type=ENUMERATED,access=rw------,values=1,items=11
  ; Item #0 'OFF'
  ; Item #1 'RCV'
  ; Item #2 'SPK'
  ; Item #3 'HP'
  ; Item #4 'HP_NO_MIC'
  ; Item #5 'BT'
  ; Item #6 'SPK_HP'
  ; Item #7 'RING_SPK'
  ; Item #8 'RING_HP'
  ; Item #9 'RING_HP_NO_MIC'
  ; Item #10 'RING_SPK_HP'
  : values=2
```

6.8 IR

Step 1: Connect the IR receiver.



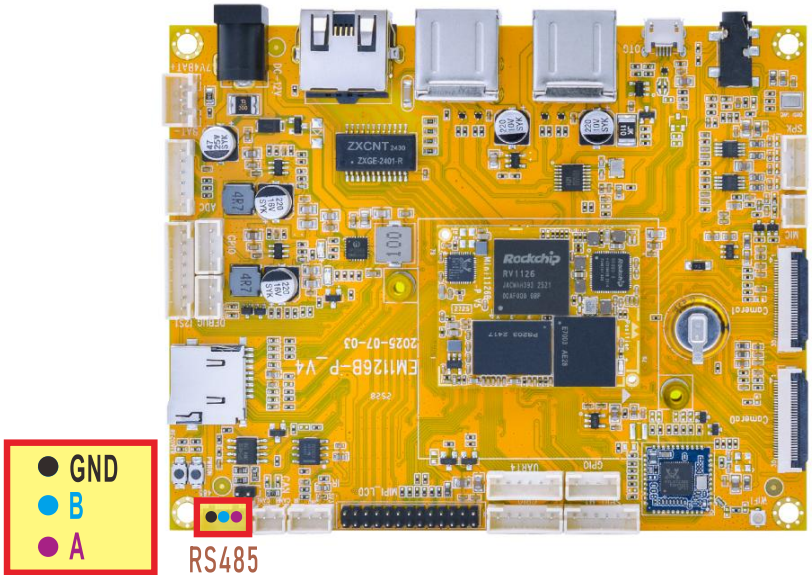
Step 2: Open IR debugging print.

```
# echo 1 > /sys/module/rockchip_pwm_remotectl/parameters/code_print
```

Step 3: When pressing a button on the remote control towards the IR receiver, the key value will be printed to the log.

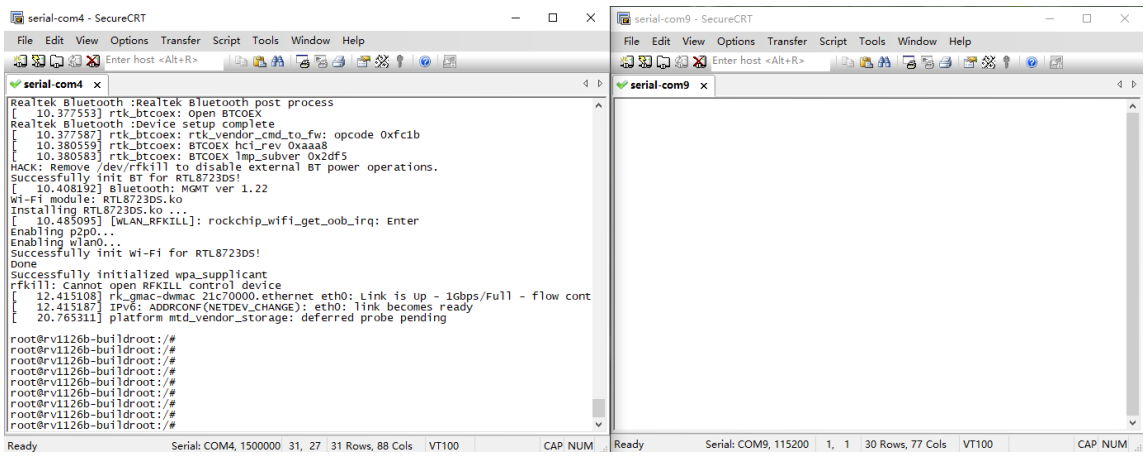
```
/code_printb-buildroot:/# echo 1 > /sys/module/rockchip_pwm_remotectl/parameters/
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/# [  293.297790] USERCODE=0x1818
[  293.325015] RMC_GETDATA=9a
[  294.217858] USERCODE=0x1818
[  294.244995] RMC_GETDATA=99
[  294.837300] USERCODE=0x1818
[  294.864505] RMC_GETDATA=98
[  295.241483] USERCODE=0x1818
[  295.268684] RMC_GETDATA=9b
[  296.021481] USERCODE=0x1818
[  296.048663] RMC_GETDATA=e6
[  296.597978] USERCODE=0x1818
[  296.625116] RMC_GETDATA=e7
[  296.953745] USERCODE=0x1818
[  296.980882] RMC_GETDATA=e4
[  297.242006] USERCODE=0x1818
[  297.269143] RMC_GETDATA=e5
```

6.9 RS485



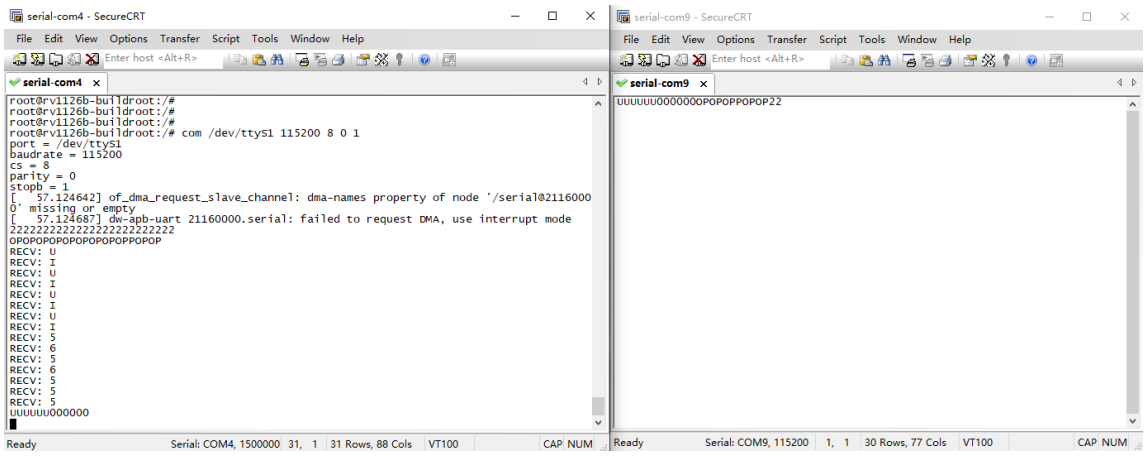
Step 1: As shown in the diagram, connect the RS485 test tool to the development board.

Step 2: Open the corresponding serial terminal, set the baud rate of the board to 1500000, and set the baud rate of the RS485 test tool to 115200.



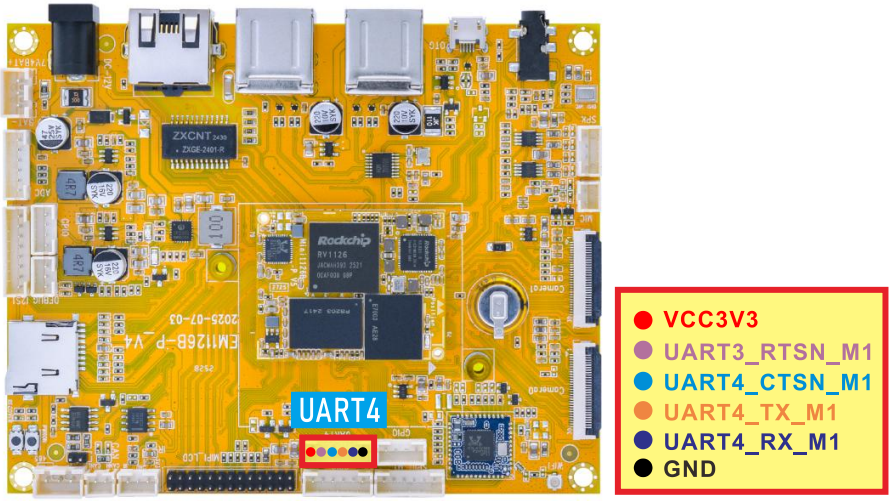
Step 3: Execute the following command on the board to test the RS485 transmission and reception functionality.

```
# com /dev/ttyS1 115200 8 0 1
```



6.10 UART

Step 1: Short circuit **UART4_RX_M1** and **UART4_TX_M1** pins of UART.

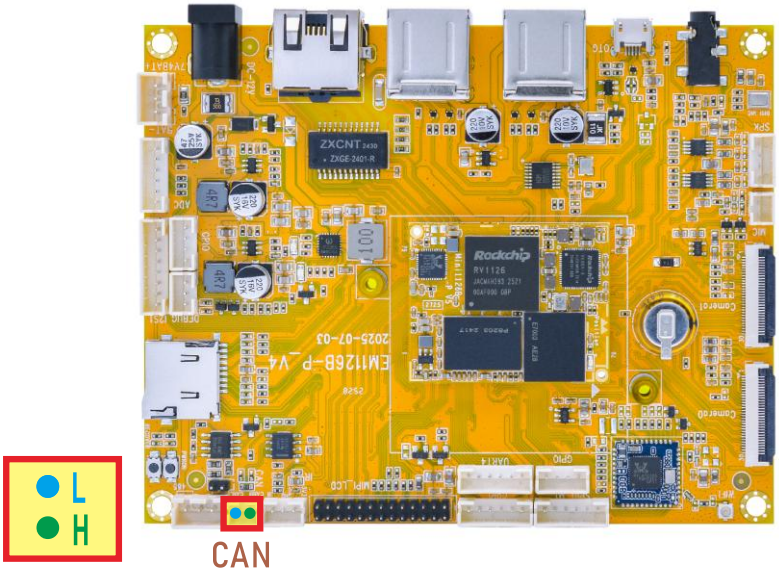


Step 2: UART1 test.

```
# com /dev/ttyS4 115200 8 0 1
```

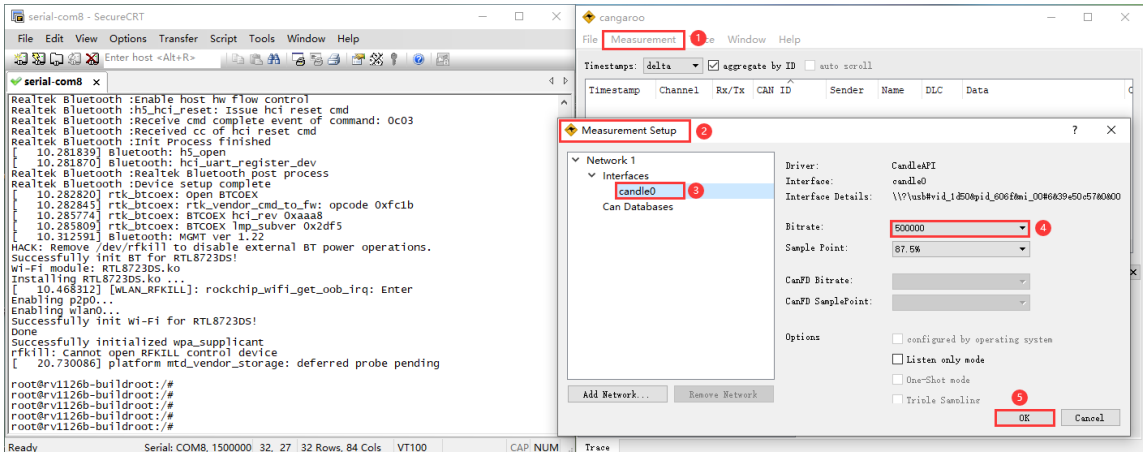
```
root@rv1126b-buildroot:/# com /dev/ttyS4 115200 8 0 1
port = /dev/ttyS4
baudrate = 115200
cs = 8
parity = 0
stopb = 1
JKJKJKJKJKJK
RECV: JKJKJKJKJKJK
89898989PP
RECV: 89898989PP
2255UU
RECV: 2255UU
YTYT
RECV: YTYT
00000
RECV: 00000
```

6.11 CAN



Step 1: Connect the CAN test tool to the board as shown in the diagram below.

Step 2: Open the CAN test software and set the baud rate to 500000.

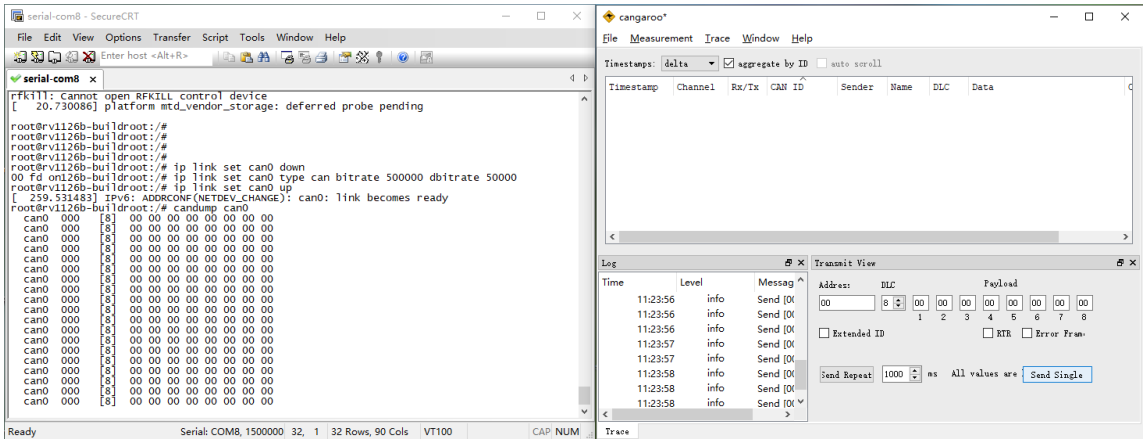


Step 3: Configure and activate the CAN network, setting the bitrate to 500000.

```
# ip link set can0 down
# ip link set can0 type can bitrate 500000 dbitrate 500000 fd on
# ip link set can0 up
```

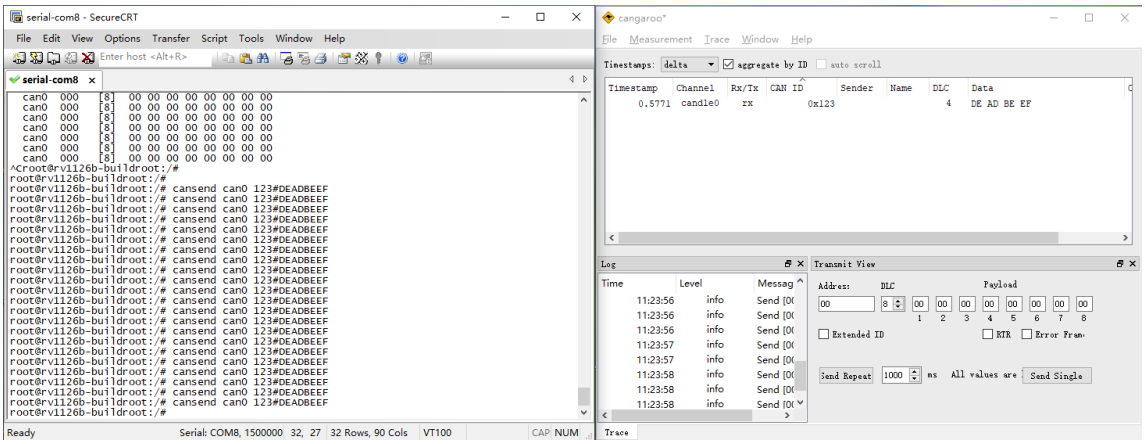
Step 4: Configure CAN as the receiver.

```
# candump can0
```



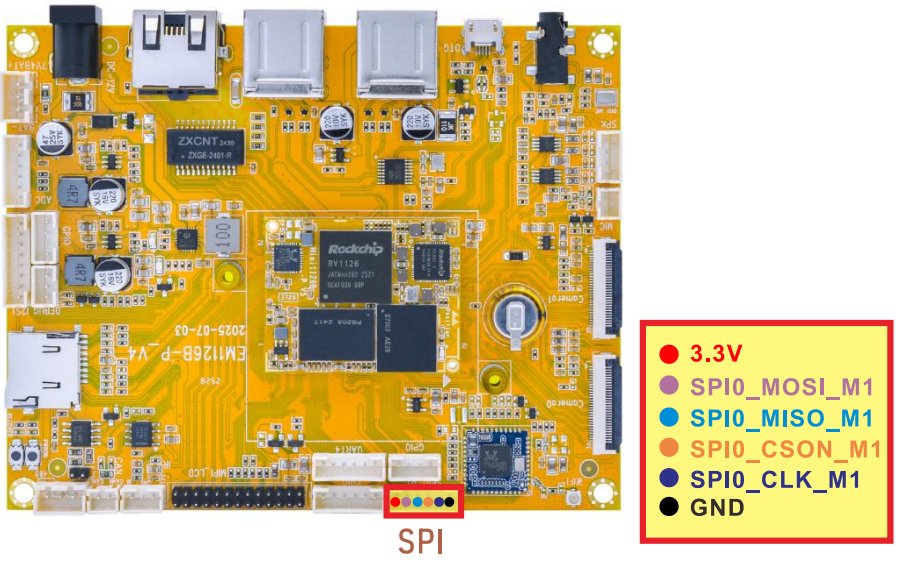
Step 5: Configure CAN as the sender.

```
# cansend can0 123#DEADBEEF
```



6.12 SPI

Step 1: Short circuit **SPI0_MOSI_M1** and **SPI0_MISO_M1** pins of SPI.

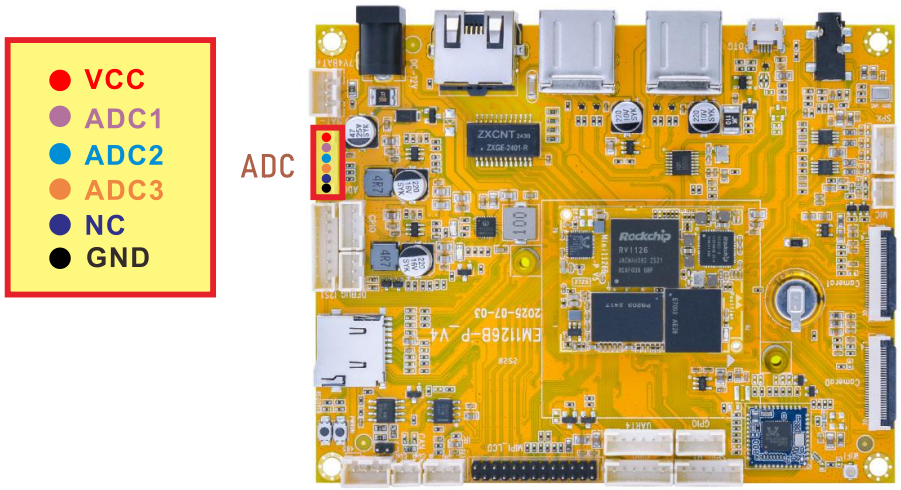


Step 2: SPI test:

```
# spidev-test
```

```
root@rv1126b-buildroot:/# spidev-test
spi mode: 0
bits per word: 8
max speed: 10000000 Hz (10000 KHz)

EE FF FF FF FF FF
40 00 00 00 00 95
FF FF FF FF FF FF
FF FF FF FF FF FF
FF FF FF FF FF FF
DE AD BE EF BA AD
F0 0D
```

6.13 ADC



Connect the ADC pin (ADC1/ADC2/ADC3) to high and low levels respectively. After shorting the pin to the desired level, run the following command.

Example for **ADC1**:

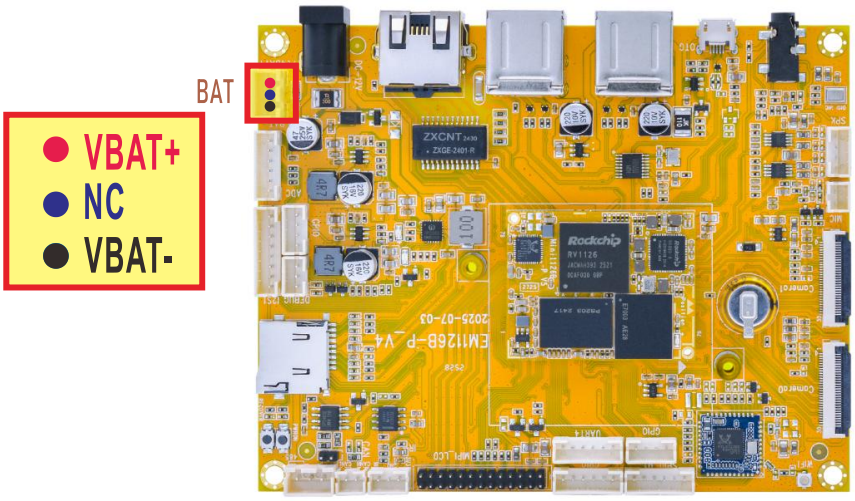
```
# cat /sys/bus/iio/devices/iio:\device0/in_voltage1_raw
```

```
root@rv1126b-buildroot:/# cat /sys/bus/iio/devices/iio:\device0/in_voltage1_raw
42
root@rv1126b-buildroot:/# cat /sys/bus/iio/devices/iio:\device0/in_voltage1_raw
8191
```

Repeat with in_voltage2_raw or in_voltage3_raw for **ADC2/ADC3**.

6.14 Battery supply

Step 1: Connect the 7.4V Li-ion battery.

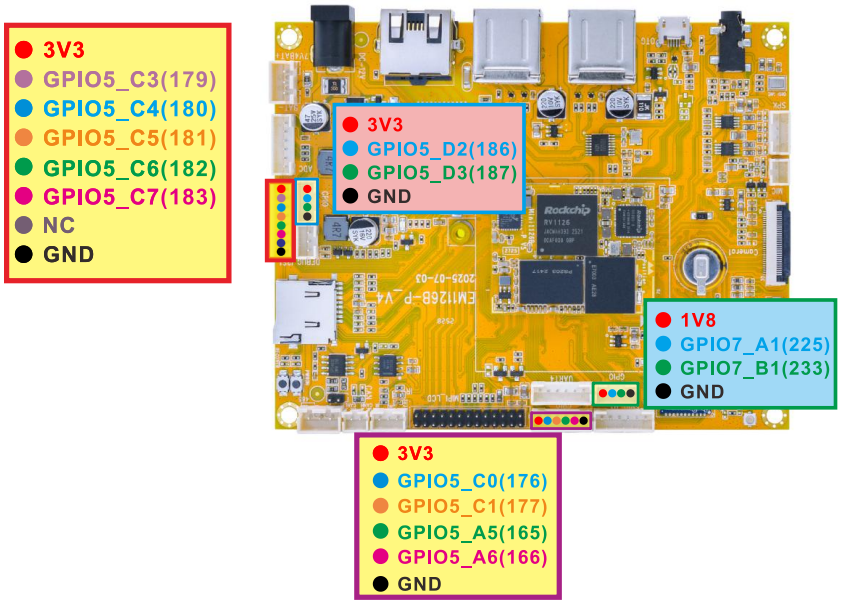


Step 2: Run the following command to check the current voltage:

```
# cat /sys/class/power_supply/battery/voltage_now
```

```
root@rv1126b-buildroot:/# cat /sys/class/power_supply/battery/voltage_now
7856000
root@rv1126b-buildroot:/# cat /sys/class/power_supply/battery/voltage_now
7832000
root@rv1126b-buildroot:/# cat /sys/class/power_supply/battery/voltage_now
7824000
```

6.15 GPIO



General GPIO Control Command Guide：

```
# Export the GPIO and set it as output
gpio_ctrl.sh x export

# Set output to high level
gpio_ctrl.sh x set 1

# Read the current GPIO level
gpio_ctrl.sh x get

# Set output to low level (should be 1)
gpio_ctrl.sh x set 0

# Read the GPIO level again (should be 0)
gpio_ctrl.sh x get

# Unexport GPIO
gpio_ctrl.sh x unexport
```

For example, using

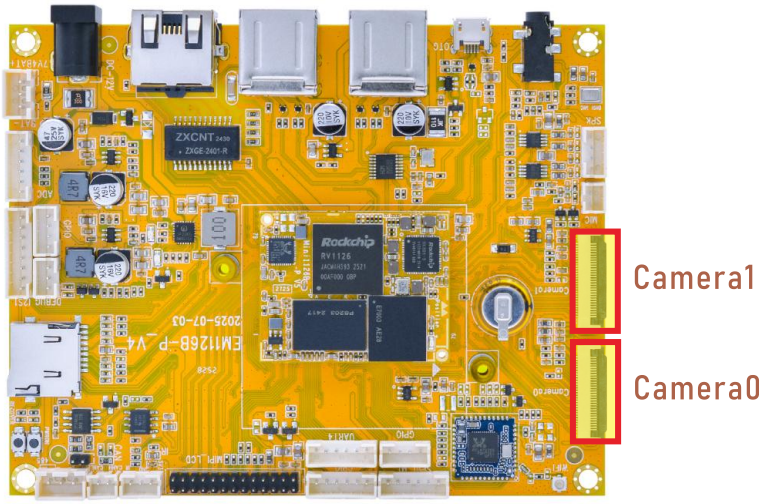
GPIO-186:

```
# gpio_ctrl.sh 186 export
# gpio_ctrl.sh 186 set 1
# gpio_ctrl.sh 186 get
# gpio_ctrl.sh 186 set 0
# gpio_ctrl.sh 186 get
# gpio_ctrl.sh 186 unexport
```

```
root@rv1126b-buildroot:/# gpio_ctrl.sh 186 export
GPIO186 exported as out
root@rv1126b-buildroot:/# gpio_ctrl.sh 186 set 1
GPIO186 set to 1
root@rv1126b-buildroot:/# gpio_ctrl.sh 186 get
GPIO186 value: 1
root@rv1126b-buildroot:/# gpio_ctrl.sh 186 set 0
GPIO186 set to 0
root@rv1126b-buildroot:/# gpio_ctrl.sh 186 get
GPIO186 value: 0
root@rv1126b-buildroot:/# gpio_ctrl.sh 186 unexport
GPIO186 unexported
```

6.16 Camera

Step 1: Power on after connecting the camera.



Step 2: Preview camera0.

```
# ./rockchip-test/camera/boardcon-camera0-test.sh
```

```
root@rv1126b-buildroot:/# ./rockchip-test/camera/boardcon-camera0-test.sh
Setting pipeline to PAUSED ...
Using mplane plugin for capture
Pipeline is live and does not need PREROLL ...
Pipeline is PREROLLED ...
Setting pipeline to PLAYING ...
/GstPipeline:pipeline0/GstV4l2Src:v4l2src0: crop-bounds = < (int)0, (int)0, (int)3840, (int)2160 >
New clock: GstSystemClock
/GstPipeline:pipeline0/GstV4l2Src:v4l2src0.GstPad:src:    caps    =    video/x-raw,    format=(string)NV12,
width=(int)640,      height=(int)480,      framerate=(fraction)30/1,      pixel-aspect-ratio=(fraction)1/1,
interlace-mode=(string)progressive, colorimetry=(string)bt601
[ 1930.612903] rkisp_hw 21d00000.isp: set isp clk = 500000000Hz
/GstPipeline:pipeline0/GstCapsFilter:capsfilter0.GstPad:src:  caps  =  video/x-raw,  format=(string)NV12,
width=(int)640,      height=(int)480,      framerate=(fraction)30/1,      pixel-aspect-ratio=(fraction)1/1,
interlace-mode=(string)progressive, colorimetry=(string)bt601
/GstPipeline:pipeline0/GstWaylandSink:waylandsink0.GstPad:sink:  caps  =  video/x-raw,  format=(string)NV12,
width=(int)640,      height=(int)480,      framerate=(fraction)30/1,      pixel-aspect-ratio=(fraction)1/1,
interlace-mode=(string)progressive, colorimetry=(string)bt601
/GstPipeline:pipeline0/GstCapsFilter:capsfilter0.GstPad:sink:  caps  =  video/x-raw,  format=(string)NV12,
width=(int)640,      height=(int)480,      framerate=(fraction)30/1,      pixel-aspect-ratio=(fraction)1/1,
interlace-mode=(string)progressive, colorimetry=(string)bt601
[ 1930.628943] rkisp rkisp-vir0: first params buf queue
[ 1930.629259] rkcif-mipi-lvds: stream[0] start streaming
[    1930.629653]    rockchip-mipi-csi2    mipi0-csi2:    stream    on,    src_sd:    00000000eead2424,
sd_name:rockchip-csi2-dphy0
[ 1930.629676] rockchip-mipi-csi2 mipi0-csi2: stream ON
[ 1930.629727] rockchip-csi2-dphy0: dphy0, data_rate_mbps 892
[ 1930.629760] rockchip-csi2-dphy csi2-dphy0: csi2_dphy_s_stream stream on:1, dphy0, ret 0
[ 1930.629772] imx415 1-0036: s_stream: 1. 3864x2192, hdr: 0, bpp: 10
[ 1930.742265] rkcif-mipi-lvds: Warning: vblank need >= 1000us if isp work in online, cur 859 us
[ 1930.776050] rkcif-mipi-lvds: Warning: vblank need >= 1000us if isp work in online, cur 859 us
Redistribute latency...
0:00:01.8 / 99:99:99.
```

Step 3: Preview camera1.

```
# ./rockchip-test/camera/boardcon-camera1-test.sh
```

```
root@rv1126b-buildroot:/# ./rockchip-test/camera/boardcon-camera1-test.sh
Setting pipeline to PAUSED ...
Using mplane plugin for capture
Pipeline is live and does not need PREROLL ...
Pipeline is PREROLLED ...
Setting pipeline to PLAYING ...
/GstPipeline:pipeline0/GstV4l2Src:v4l2src0: crop-bounds = < (int)0, (int)0, (int)3840, (int)2160 >
New clock: GstSystemClock
/GstPipeline:pipeline0/GstV4l2Src:v4l2src0.GstPad:src:    caps    =    video/x-raw,    format=(string)NV12,
width=(int)640,      height=(int)480,      framerate=(fraction)30/1,      pixel-aspect-ratio=(fraction)1/1,
interlace-mode=(string)progressive, colorimetry=(string)1:4:16:4
[ 1964.566104] rkisp_hw 21d00000.isp: set isp clk = 500000000Hz
/GstPipeline:pipeline0/GstCapsFilter:capsfilter0.GstPad:src:  caps  =  video/x-raw,  format=(string)NV12,
width=(int)640,      height=(int)480,      framerate=(fraction)30/1,      pixel-aspect-ratio=(fraction)1/1,
interlace-mode=(string)progressive, colorimetry=(string)1:4:16:4
/GstPipeline:pipeline0/GstWaylandSink:waylandsink0.GstPad:sink: caps = video/x-raw, format=(string)NV12,
width=(int)640,      height=(int)480,      framerate=(fraction)30/1,      pixel-aspect-ratio=(fraction)1/1,
interlace-mode=(string)progressive, colorimetry=(string)1:4:16:4
/GstPipeline:pipeline0/GstCapsFilter:capsfilter0.GstPad:sink:  caps  =  video/x-raw,  format=(string)NV12,
width=(int)640,      height=(int)480,      framerate=(fraction)30/1,      pixel-aspect-ratio=(fraction)1/1,
interlace-mode=(string)progressive, colorimetry=(string)1:4:16:4
[ 1964.575935] rkcif-mipi-lvds2: stream[0] start streaming
[     1964.576375]     rockchip-mipi-csi2     mipi2-csi2:     stream     on,     src_sd:     00000000f185780c,
sd_name:rockchip-csi2-dphy3
[ 1964.576401] rockchip-mipi-csi2 mipi2-csi2: stream ON
[ 1964.576454] rockchip-csi2-dphy3: dphy3, data_rate_mbps 892
[ 1964.576486] rockchip-csi2-dphy csi2-dphy3: csi2_dphy_s_stream stream on:1, dphy3, ret 0
[ 1964.576497] imx415 3-0036: s_stream: 1. 3864x2192, hdr: 0, bpp: 10
[ 1964.685408] rkcif-mipi-lvds2: Warning: vblank need >= 1000us if isp work in online, cur 859 us
[ 1964.720318] rkcif-mipi-lvds2: Warning: vblank need >= 1000us if isp work in online, cur 859 us
Redistribute latency...
0:00:02.0 / 99:99:99.
```

6.17 Video Playback

(1) The directory for the built-in video testing scripts in the system: */rockchip-test/video*

```
root@rv1126b-buildroot:/# ls /rockchip-test/video/
test_gst_multivideo.sh      test_gst_video_fps.sh     video_test.sh
test_gst_video.sh           test_gst_video_maxfps.sh
test_gst_video_benchmark.sh  video_stresstest.sh
```

Simply execute the script.

```
root@rv1126b-buildroot:/# /rockchip-test/video/test_gst_video.sh
Setting pipeline to PAUSED ...
Pipeline is PREROLLING ...
Redistribute latency...
mpp[1734]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[1734]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[1734]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[1734]: mpp: unable to create enc vp8 for soc rv1126b unsupported
mpp[1734]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[1734]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
Redistribute latency...
mpp[1734]: h264d_api: is_avcC=1
mpp[1734]: mpp_buf_slot: mismatch h_stride_by_pixel 1472 - 1280
mpp[1734]: mpp_buf_slot: mismatch h_stride_by_byte 1472 - 1280
mpp[1734]: mpp_buf_slot: mismatch size_total 1589760 - 1843200
mpp[1734]: mpp_buf_slot: mismatch h_stride_by_pixel 1472 - 1280
mpp[1734]: mpp_buf_slot: mismatch h_stride_by_byte 1472 - 1280
mpp[1734]: mpp_buf_slot: mismatch size_total 1589760 - 1843200
Pipeline is PREROLLED ...
Prerolled, waiting for async message to finish...
Setting pipeline to PLAYING ...
Redistribute latency...
New clock: GstSystemClock
0:00:24.3 / 0:00:29.5 (82.2 %)
```

(2) Use the gst-play-1.0 command to play the video.

```
# amixer -c 0 cset numid=37 2
# gst-play-1.0 --videosink="waylandsink fullscreen=true"
/mnt/udisk/video/1080P30/1080P30-H264_haidi.mp4
```

Command explanation:

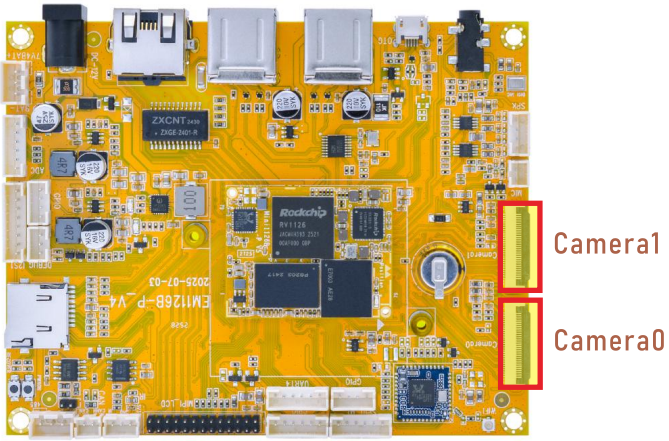
• amixer -c 0 cset numid=37 2: Audio through the speaker.

• /mnt/udisk/video/4KP25/4K25FPS.mp4: The media file path to be played.

```
root@rv1126b-buildroot:/#
  /mnt/udisk/video/1080P30/1080P30-H264_haidi.mp4k="waylandsink fullscreen=true"
Press 'k' to see a list of keyboard shortcuts.
Now playing /mnt/udisk/video/1080P30/1080P30-H264_haidi.mp4
Redistribute latency...
Redistribute latency...
mpp[2811]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[2811]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[2811]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[2811]: mpp: unable to create enc vp8 for soc rv1126b unsupported
mpp[2811]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[2811]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
Redistribute latency...
Redistribute latency...
mpp[2811]: h264d_api: is_avcC=1
mpp[2811]: mpp_buf_slot: mismatch h_stride_by_pixel 1984 - 1920
mpp[2811]: mpp_buf_slot: mismatch h_stride_by_byte 1984 - 1920
mpp[2811]: mpp_buf_slot: mismatch size_total 3237888 - 4177920
mpp[2811]: mpp_buf_slot: mismatch h_stride_by_pixel 1984 - 1920
mpp[2811]: mpp_buf_slot: mismatch h_stride_by_byte 1984 - 1920
mpp[2811]: mpp_buf_slot: mismatch size_total 3237888 - 4177920
Redistribute latency...
0:00:13.3 / 0:05:07.2
```

7.Buildroot IPC Test

Testing Guide: Please connect both Camera 0 and Camera 1 prior to powering on the device to ensure the IPC application launches correctly. By default, the IPC application starts automatically upon boot. You must execute a command to stop the IPC preview before testing other functions.




```
# killall -9 rkipc
# rkipc &
```

Command explanation:

• killall -9 rkipc : Forcefully terminate the running IPC process.

• rkipc & : Restart the IPC application in the background.

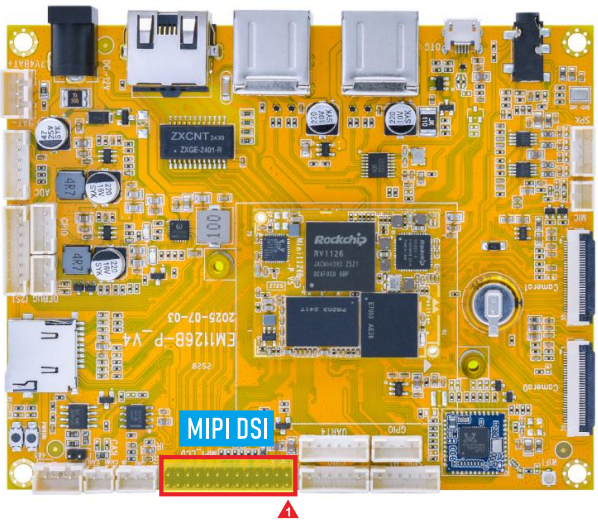
```
root@rv1126b-buildroot:/# killall -9 rkipc
root@rv1126b-buildroot:/# [   161.943509] mpp_vcodec_chan: mpp_vcodec_chan_destory: destroy chan 3 hnd
000000009a140a3a online 0 combo -1 mst -1
[  161.943711] mpp_vcodec_chan: mpp_vcodec_chan_destory: destroy chan 3 done
[  161.943724] mpp_vcodec_chan: mpp_vcodec_chan_destory: destroy chan 1 hnd 00000000b1bcaced online 0 combo
-1 mst -1
[  161.943872] mpp_vcodec_chan: mpp_vcodec_chan_destory: destroy chan 1 done
[  161.943885] mpp_vcodec_chan: mpp_vcodec_chan_destory: destroy chan 0 hnd 000000000beebe2f online 0 combo
-1 mst -1
[  161.944095] mpp_vcodec_chan: mpp_vcodec_chan_destory: destroy chan 0 done
[
[  162.254805] rkvpss-vir0: rkvpss_stream_stop id:3 timeout
[  162.284451] kmpp_venc_chan_put_frm: invalid chan id 0
[  162.317606] rkcif-mipi-lvds: stream[0] start stopping, total mode 0x8, cur 0x8
[       162.364606]    rockchip-mipi-csi2    mipi0-csi2:    stream    off,    src_sd:    0000000020ec0176,
sd_name:rockchip-csi2-dphy0
[  162.364642] rockchip-mipi-csi2 mipi0-csi2: stream OFF
[  162.364673] rockchip-csi2-dphy csi2-dphy0: csi2_dphy_s_stream_stop stream stop, dphy0
[  162.364694] rockchip-csi2-dphy csi2-dphy0: csi2_dphy_s_stream stream on:0, dphy0, ret 0
[  162.364729] imx415 1-0036: s_stream: 0. 3864x2192, hdr: 0, bpp: 10
[  162.365338] rkcif-mipi-lvds: stream[0] stopping finished, dma_en 0x0
[  163.374811] rkisp-vir0: waiting on params stream off event timeout
[  163.694807] rkvpss-vir0: rkvpss_stream_stop id:0 timeout
[  163.706286] ivs_release 20
[  163.706857] vi_release 27
[  163.706929] rockit vpss stream off
[  163.707031] rockit vpss stream off
[  163.707105] rockit vpss stream off
[  163.707440] venc_release 19
[  163.707819] vrgn_release 30
[  163.733310] vsys_release, 6
[  163.802879] dw9714 1-000c: dw9714_set_power(1046) on(0)
```

7.1 Serial Terminal

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.1 of the Buildroot Testing Guide.

7.2 Display

The default MIPI display resolution on EM1126B-P is 800×1280 @60Hz.



The display effect diagram is as follows:

Note: The MIPI screen displays the live preview from Camera 0. As the Buildroot IPC version does not include a graphical desktop environment, the screen will go black if the IPC application is stopped.

7.3 USB

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.3 of the Buildroot Testing Guide.

7.4 Ethernet

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.4 of the Buildroot Testing Guide.

7.5 SD Card

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.5 of the Buildroot Testing Guide.

7.6 WIFI & Bluetooth

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.6 of the Buildroot Testing Guide.

7.7 Audio

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.7 of the Buildroot Testing Guide.

7.8 IR

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.8 of the Buildroot Testing Guide.

7.9 RS485

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.9 of the Buildroot Testing Guide.

7.10 UART

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.10 of the Buildroot Testing Guide.

7.11 CAN

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.11 of the Buildroot Testing Guide.

7.12 SPI

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.12 of the Buildroot Testing Guide.

7.13 ADC

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.13 of the Buildroot Testing Guide.

7.14 Battery supply

Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.14 of the Buildroot Testing Guide.

7.15 GPIO

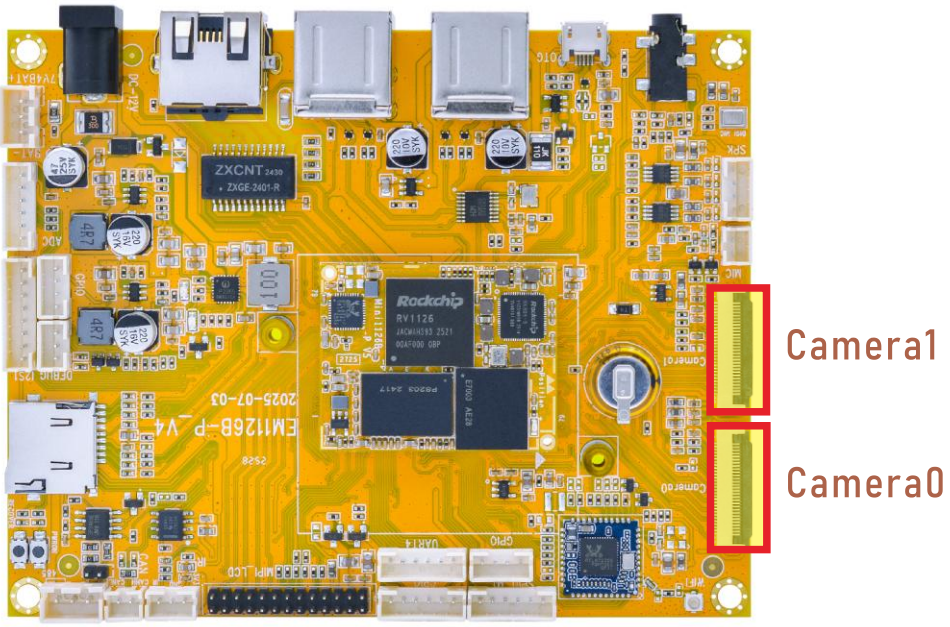
Note: The test procedure in this section is identical to the standard Buildroot version. Please refer to Section 6.15 of the Buildroot Testing Guide.

7.16 RKIPC

To run RKIPC successfully, please be sure to prepare the following environment. Otherwise, RKIPC cannot start up normally.

Step 1: Power off.

Step 2: Connect the two IMX415 camera modules to the camera0 and camera1 interfaces respectively



Step 3: Power on.

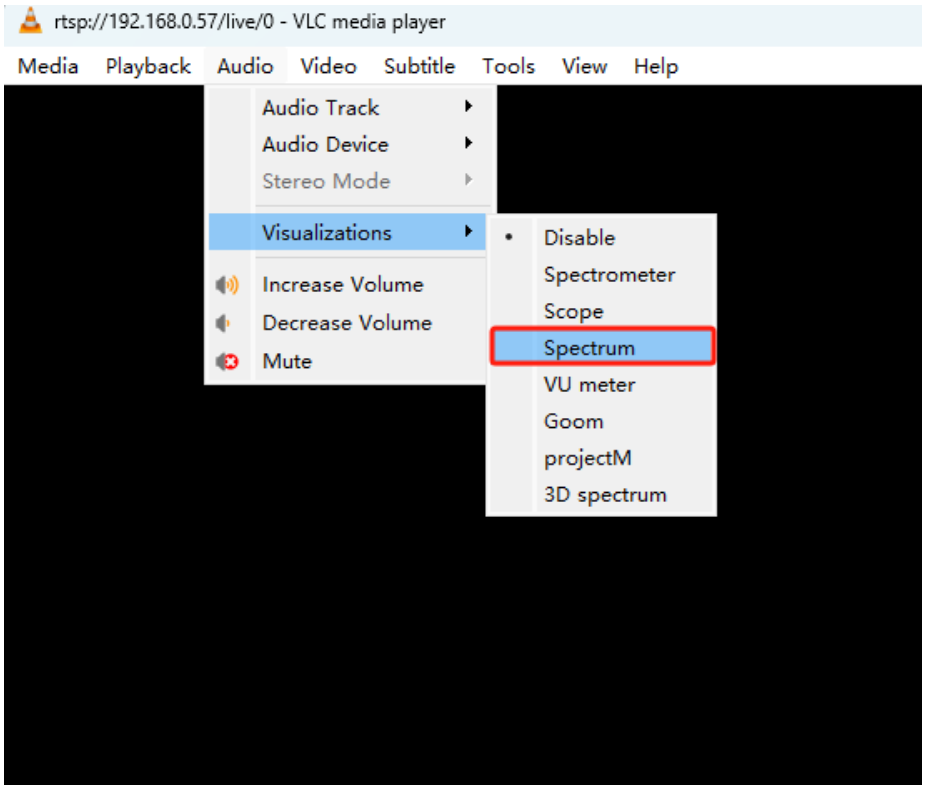
Executing the following instructions to ensure that the RKIPC process is running

```
# ps | grep rkipc
```

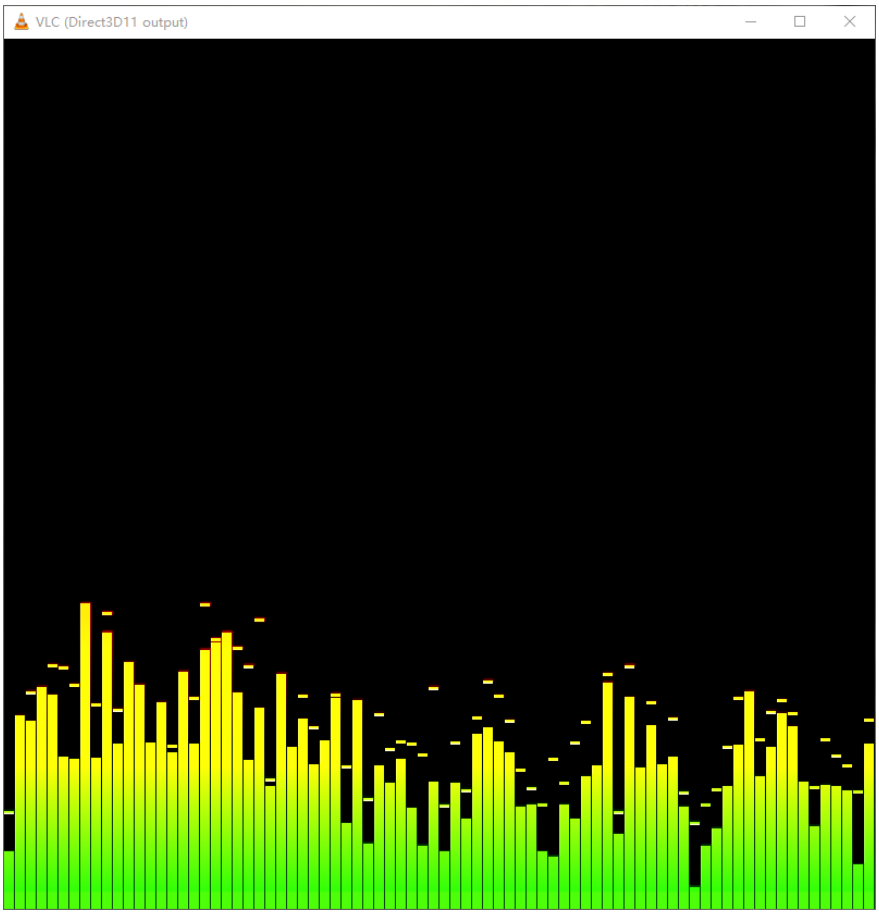
```
# ps | grep rkipc
  433 root      1234 S    rkipc -a /oem/usr/share/iqfiles
  574 root      1176 S    grep rkipc
```

7.16.1 Audio (MIC) test

The RTSP bitstream preview can be accessed through the PC end, and the audio can be obtained simultaneously. The audio can be played directly from the speaker or detected using the visual audio provided by the software. The method for viewing the audio spectrum is as follows:

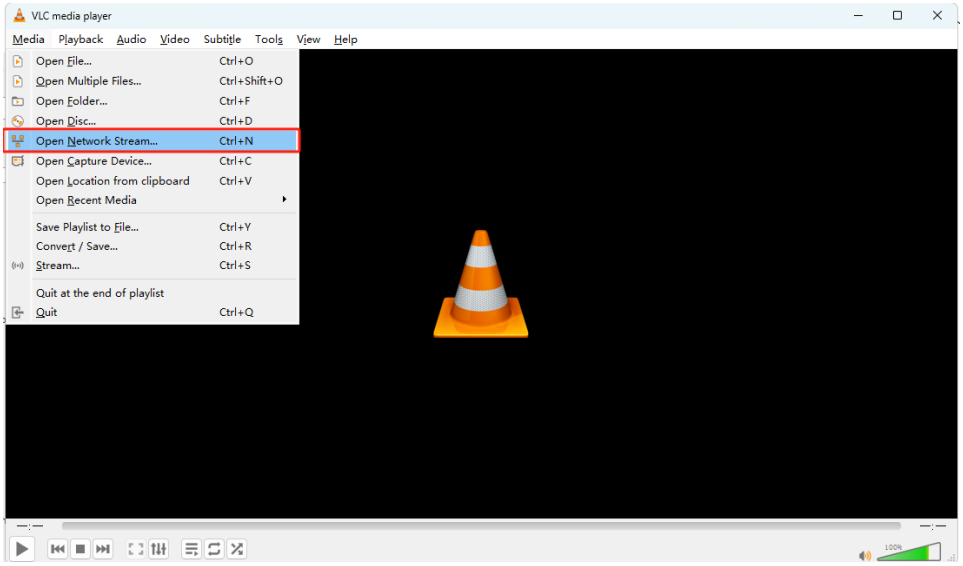


At this point, when speaking to the MIC,the spectrum effect is as follows:



7.16.2 RTSP

The video footage from the camera can be previewed within the same local area network. After the device is connected to the network, the RTSP software on the PC (such as VLC) can be used to enable network streaming.

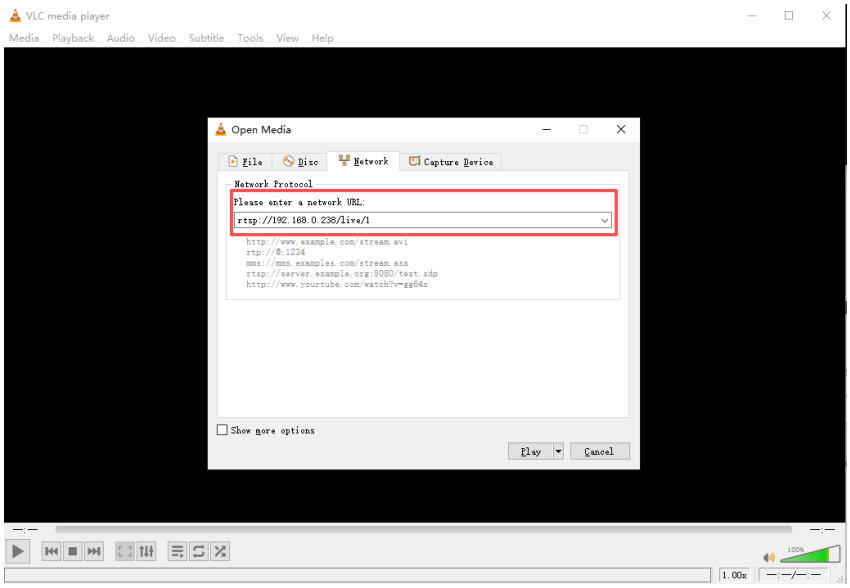


Enter the following address:

```
# rtsp:// (IP address of the device) /live/0
```

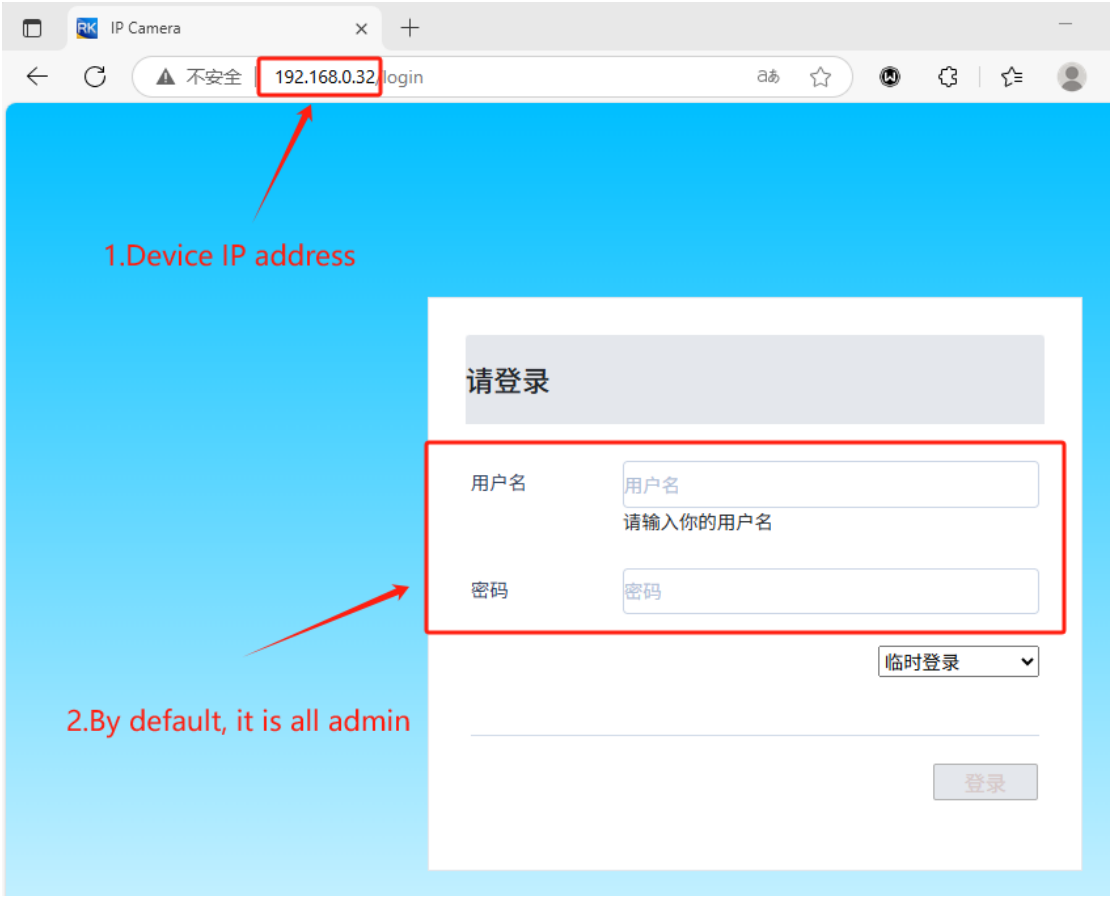
For example, in Section 7.4, using the Ethernet IP address 192.168.0.238, input rtsp://192.168.0.238/live/0 and click the play button to preview the video from Camera 0.

The WiFi streaming method is identical



7.16.3 Web Live Streaming

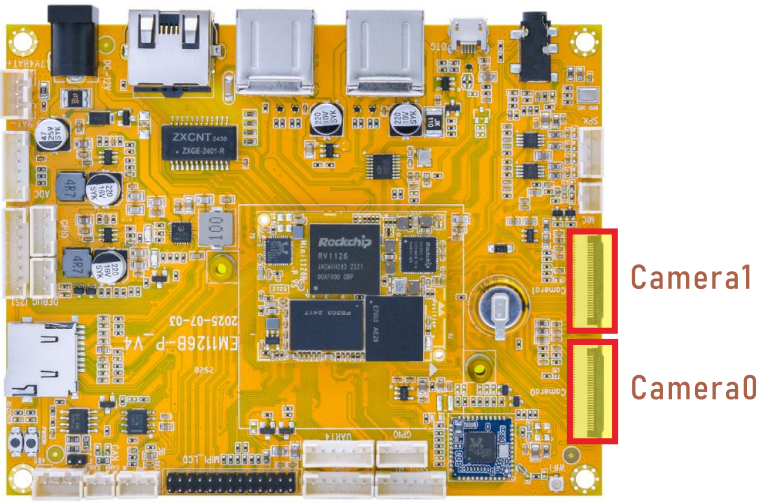
The Web end supports main stream, sub-stream, and H264/H265 live streaming. Access the device's IP address through a browser, enter the username and password (both default to admin), and enter the live preview system.



The preview effect is as follows:

7.17 Camera Streaming

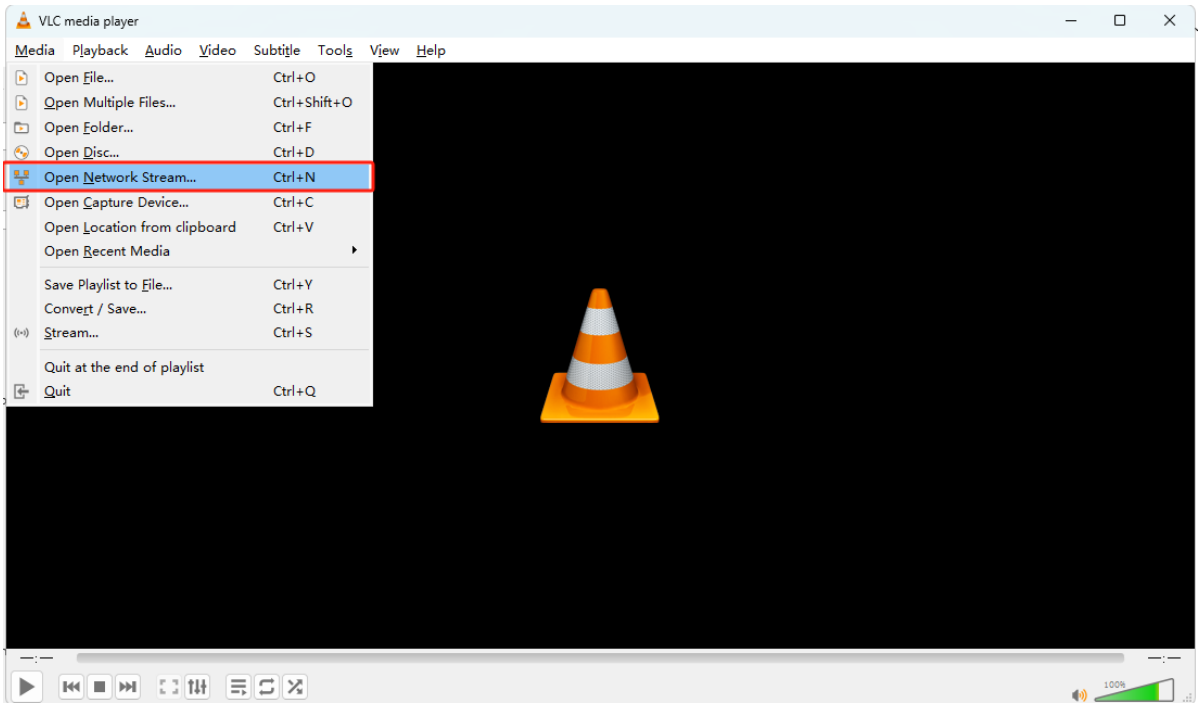
Step 1: Power on after connecting the camera



Step 2: Start Streaming Execute the following command to start the dual-camera stream.

```
# sample_demo_dual_camera -s 0 -W 1920 -H 1080 -w 720 -h 576 -f 30 -r 0 -s 1 -W 1920
-H 1080 -w 720 -h 576 -f 30 -r 0 -n 1 -b 1
```

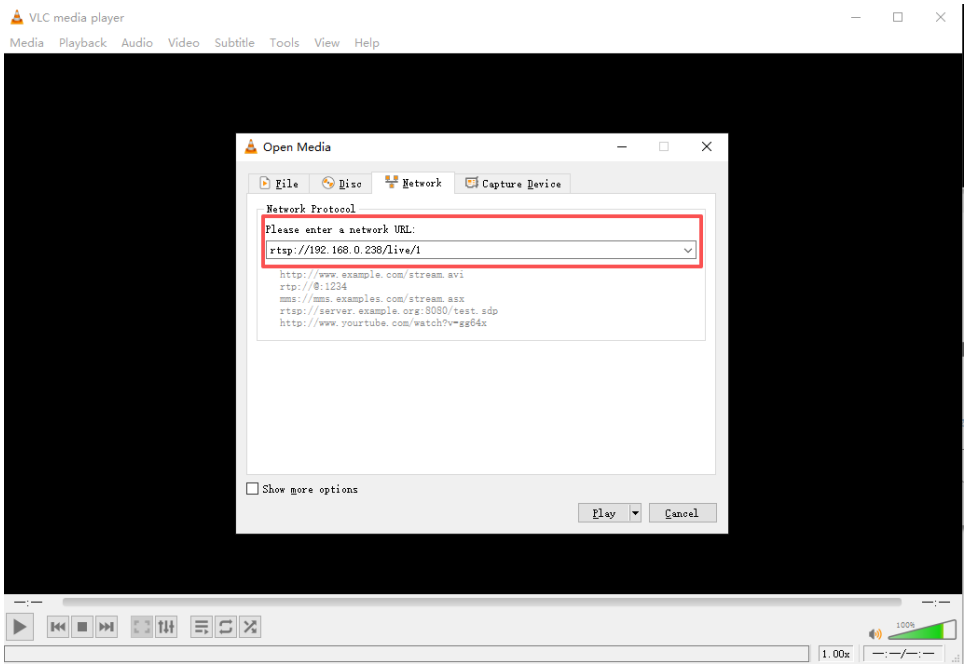
After the device is connected to the network, the RTSP software on the PC (such as VLC) can be used to enable network streaming.

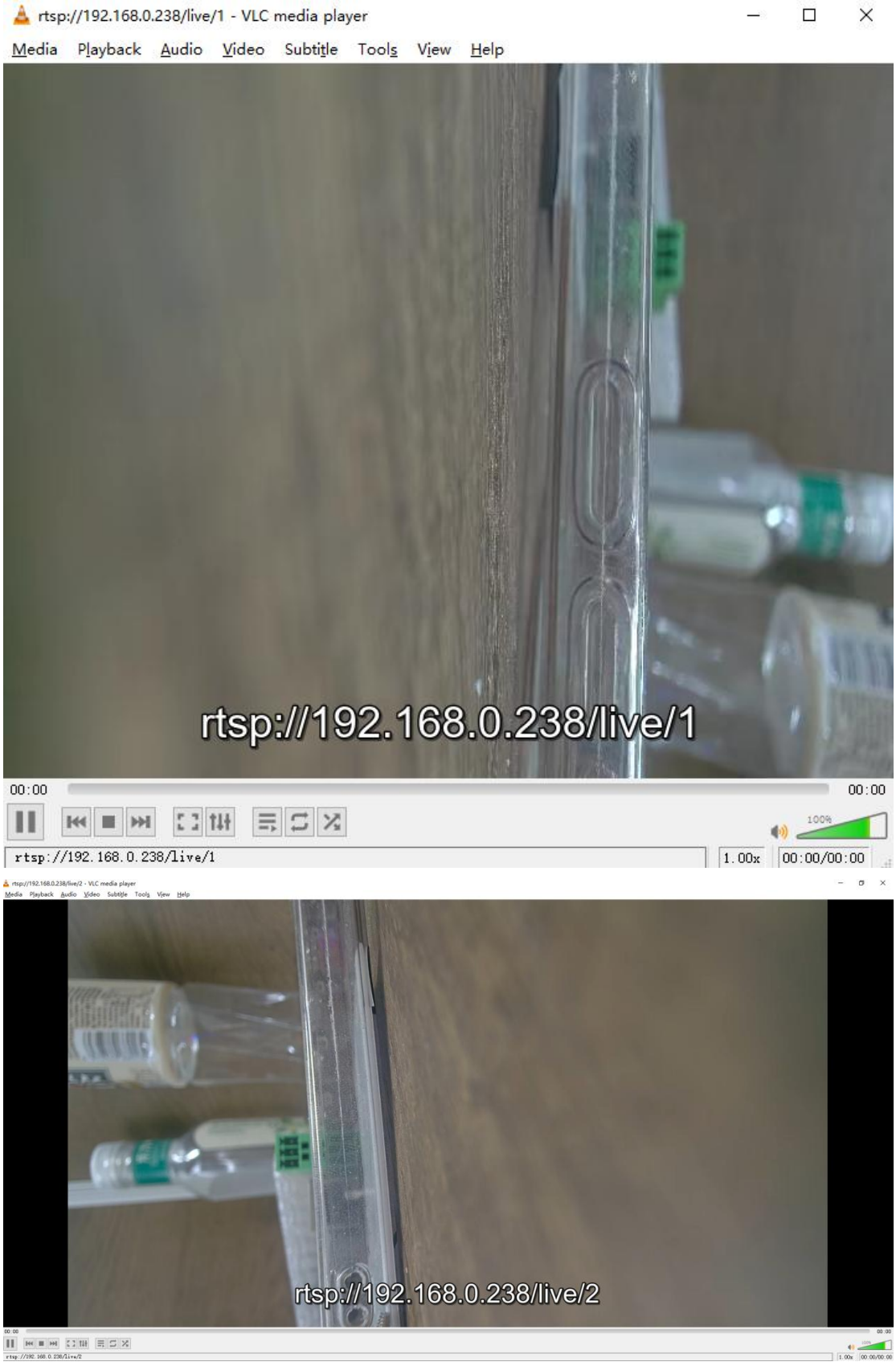


Enter the following address:

```
# rtsp:// (IP address of the device) /live/1
# rtsp:// (IP address of the device) /live/2
```

For example, in Section 7.4, using the Ethernet IP address 192.168.0.238: Input rtsp://192.168.0.238/live/1 to preview Camera 0, and rtsp://192.168.0.238/live/2 to preview Camera 1. Click the play button to start the preview. The WiFi streaming method is identical.





As shown in the figure above, the dual-channel video is displayed normally in VLC, confirming that the IPC RTSP streaming function is working correctly.

7.18 Video Decoder (VDEC) Test

This section verifies the hardware video decoding capabilities for H.264 and H.265 formats using the Rockchip MPI interface.

Note: The test files used below are raw video streams (.h264/.h265) without audio tracks. Therefore, the video will play without sound, which is the expected behavior.

1: H.264 Decoding Test Execute the following command to test H.264 decoding

```
# simple_vdec_bind_vo -i /usr/share/video/testh264.h264
```

```
root@rv1126b-buildroot:/# simple_vdec_bind_vo -i /usr/share/video/testh264.h264
Input: /usr/share/video/testh264.h264, Bind: 1
rockit_load start
[  474.241060] vsys_release, 6
v4l2_tx probe successv4l2_rx_probe success
rockit_load end
rockit log path (null), log_size = 0
log_file = (nil)
(null)          10:23:30-030 {log_level_init    :209}

 please use echo name=level > /tmp/rt_log_level set log level
       name: all cmpi mb sys vdec venc rgn vpss vgs tde avs wbc vo vi ai ao aenc adec
       log_level: 0 1 2 3 4 5 6

rockit default level 4, can use export rt_log_level=x, x=0,1,2,3,4,5,6 change
(null)          10:23:30-030 {read_log_level    :083} text is all=4
(null)          10:23:30-030 {read_log_level    :085} module is all, log_level is 4
(null)                                          10:23:30-031    {dump_version                       :060}
-------------------------------------------------------
(null)          10:23:30-031 {dump_version      :061} rockit version: git-5057bd373 Thu Dec 11 09:34:24 2025
+0800
(null)          10:23:30-031 {dump_version      :062} rockit building: built-Chu 2025-12-15 10:15:00
(null)                                          10:23:30-031    {dump_version                       :063}
-------------------------------------------------------
(null)          10:23:30-031 {monitor_log_level :141} #Start monitor_log_level thread, arg:(nil)
[  476.397178] vsys dev open 6
vsys dev open 4
[  476.406974] rkvpss-offline: CLK_CORE_VPSS set to 396000000 Hz (requested 400000000 Hz)
load library(librga.so) in releative path
Video resolution: 800x1280, Codec: H.264
mpp[3035]: mpp_info: mpp version: 53ff3d39 author: Yandong Lin   2025-12-23 fix[hal_vepu]: poll max set to
1 on split out lowdelay mode
mpp[3035]: hal_264d_com: control info: fmt 0, w 800, h 1280
mpp[3035]: mpp_dec: mpp_dec_proc_cfg found MPP_DEC_SET_FRAME_INFO fmt 0
mpp[3035]: mpp_buf_slot: mismatch h_stride_by_pixel 960 - 800
mpp[3035]: mpp_buf_slot: mismatch h_stride_by_byte 960 - 800
mpp[3035]: mpp_buf_slot: mismatch size_total 1843200 - 2048000
mpp[3035]: mpp_buf_slot: set frame info: w  800 h 1280 hor  800 ver 1280
mpp[3035]: mpp_dec: setting default w  800 h 1280 h_str  800 v_str 1280
Video codec: H.264/AVC
Video: 800x1280, timebase: 1/1200000
[SEND] Frame 100, PTS=0 us
get a hdl = 0x55ae200870
[  476.632294] phys = 4c3f1000
get a hdl = 0x55ae2009d0
[  476.636354] phys = 56400000
get a hdl = 0x55ae2011e0
load library(librga.so) in releative path
rga_api version 1.10.5_[9]
udhcpc: broadcasting discover
[SEND] Frame 200, PTS=0 us
udhcpc: broadcasting discover
```

2. H.265 Decoding Test Execute the following command to test H.265 decoding:

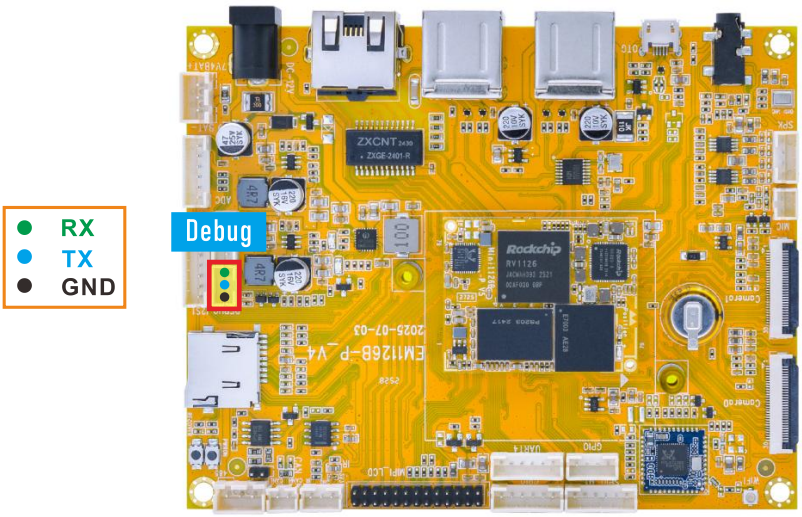
```
# simple_vdec_bind_vo -i /usr/share/video/testh265.h265
```

```
root@rv1126b-buildroot:/# simple_vdec_bind_vo -i /usr/share/video/testh265.h265
Input: /usr/share/video/testh265.h265, Bind: 1
rockit_load start
[  578.786312] vsys_release, 6
v4l2_tx probe successv4l2_rx_probe success
rockit_load end
rockit log path (null), log_size = 0
log_file = (nil)
(null)          10:26:18-749 {log_level_init    :209}

 please use echo name=level > /tmp/rt_log_level set log level
       name: all cmpi mb sys vdec venc rgn vpss vgs tde avs wbc vo vi ai ao aenc adec
       log_level: 0 1 2 3 4 5 6

rockit default level 4, can use export rt_log_level=x, x=0,1,2,3,4,5,6 change
(null)          10:26:18-749 {read_log_level    :083} text is all=4
(null)          10:26:18-749 {read_log_level    :085} module is all, log_level is 4
(null)                                          10:26:18-749    {dump_version                       :060}
-------------------------------------------------------
(null)          10:26:18-749 {dump_version      :061} rockit version: git-5057bd373 Thu Dec 11 09:34:24 2025
+0800
(null)          10:26:18-749 {dump_version      :062} rockit building: built-Chu 2025-12-15 10:15:00
(null)                                          10:26:18-749    {dump_version                       :063}
-------------------------------------------------------
[  645.116044] vsys dev open 6
(null)          10:26:18-749 {monitor_log_level :141} #Start monitor_log_level thread, arg:(nil)
[  645.125395] rkvpss-offline: CLK_CORE_VPSS set to 396000000 Hz (requested 400000000 Hz)
vsys dev open 4
load library(librga.so) in releative path
Video resolution: 800x1280, Codec: H.265
mpp[3489]: mpp_info: mpp version: 53ff3d39 author: Yandong Lin   2025-12-23 fix[hal_vepu]: poll max set to
1 on split out lowdelay mode
mpp[3489]: mpp_dec: mpp_dec_proc_cfg found MPP_DEC_SET_FRAME_INFO fmt 0
mpp[3489]: mpp_buf_slot: mismatch h_stride_by_pixel 960 - 832
mpp[3489]: mpp_buf_slot: mismatch h_stride_by_byte 960 - 832
mpp[3489]: mpp_buf_slot: mismatch size_total 2211840 - 1916928
mpp[3489]: mpp_buf_slot: set frame info: w  800 h 1280 hor  832 ver 1280
mpp[3489]: mpp_dec: setting default w  800 h 1280 h_str  832 v_str 1280
Video codec: H.265/HEVC
Video: 800x1280, timebase: 1/1200000
[SEND] Frame 100, PTS=0 us
get a hdl = 0x559764aa80
[  645.391701] phys = 565a8000
get a hdl = 0x55976051b0
[  645.394481] phys = 4c4fd000
get a hdl = 0x559756ccf0
load library(librga.so) in releative path
rga_api version 1.10.5_[9]
[SEND] Frame 200, PTS=0 us
udhcpc: broadcasting discover
[SEND] Frame 300, PTS=0 us
udhcpc: broadcasting discover
[SEND] Frame 400, PTS=0 us
udhcpc: broadcasting discover
```

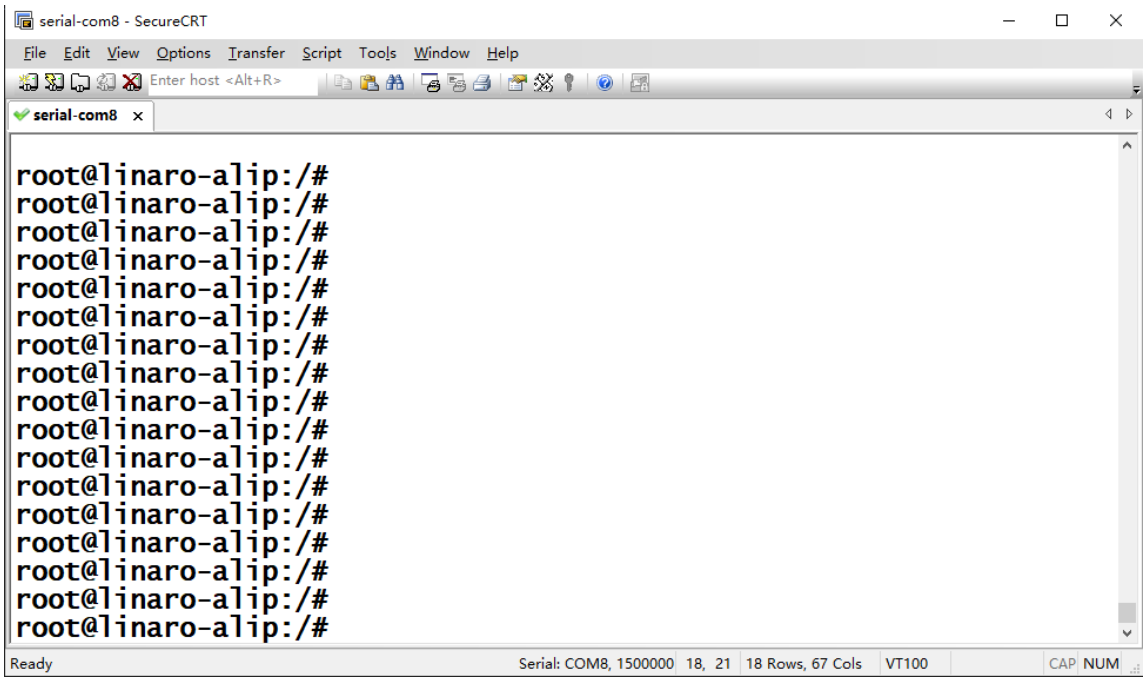
Expected Result: The video should play smoothly on the display screen.

8.Debian12 Test

8.1 Serial Terminal

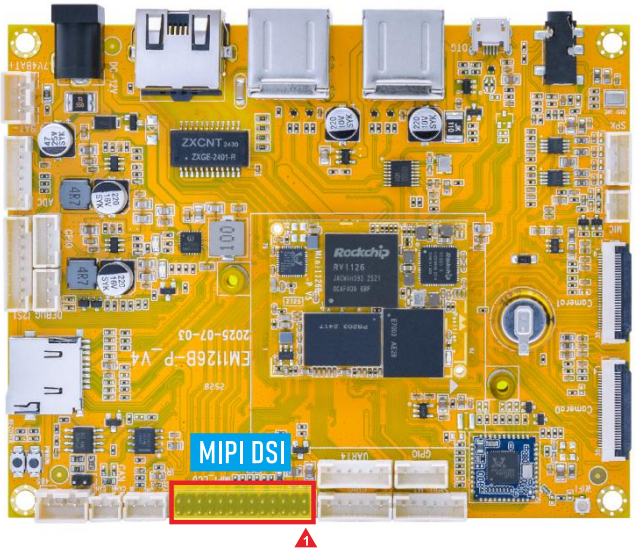


Connect the board and PC with USB Serial cable, then power on, the terminal will output boot information. The default baudrate is 1500000.



8.2 Display

The default MIPI display resolution on EM1126B-P is 800×1280 @60Hz.



The display effect diagram is as follows:

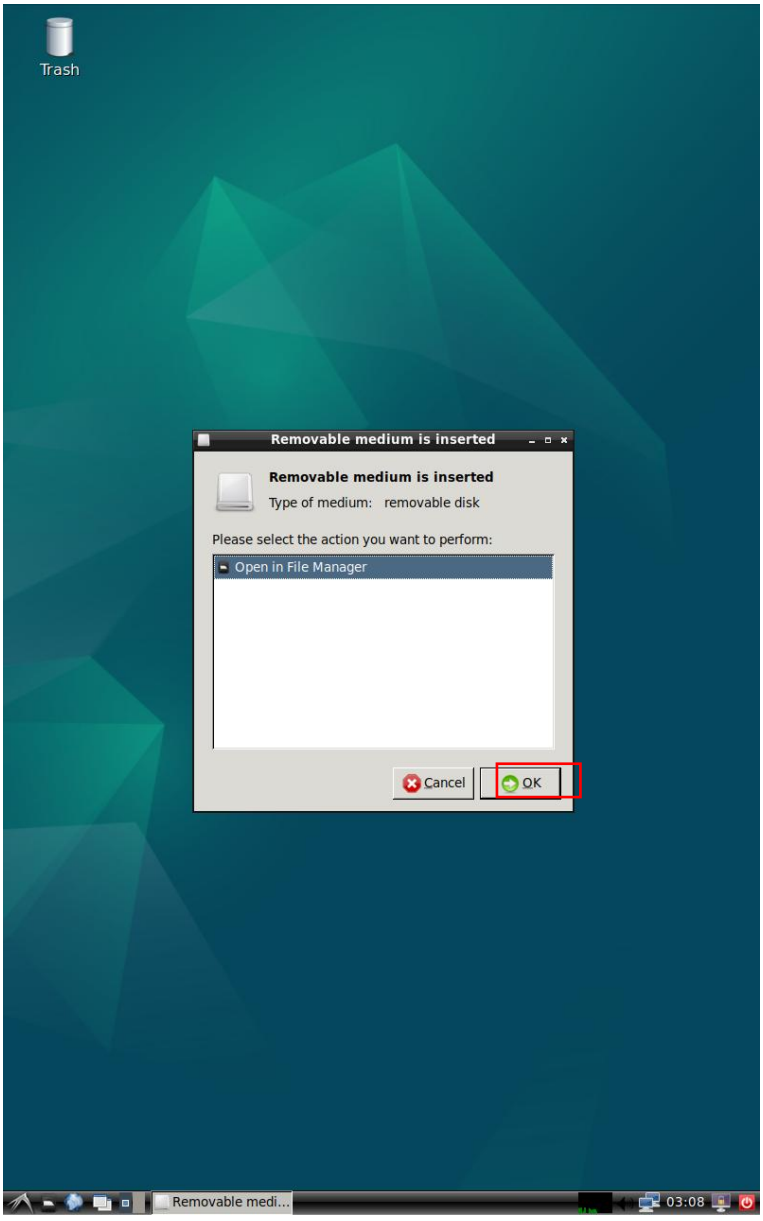
8.3 USB

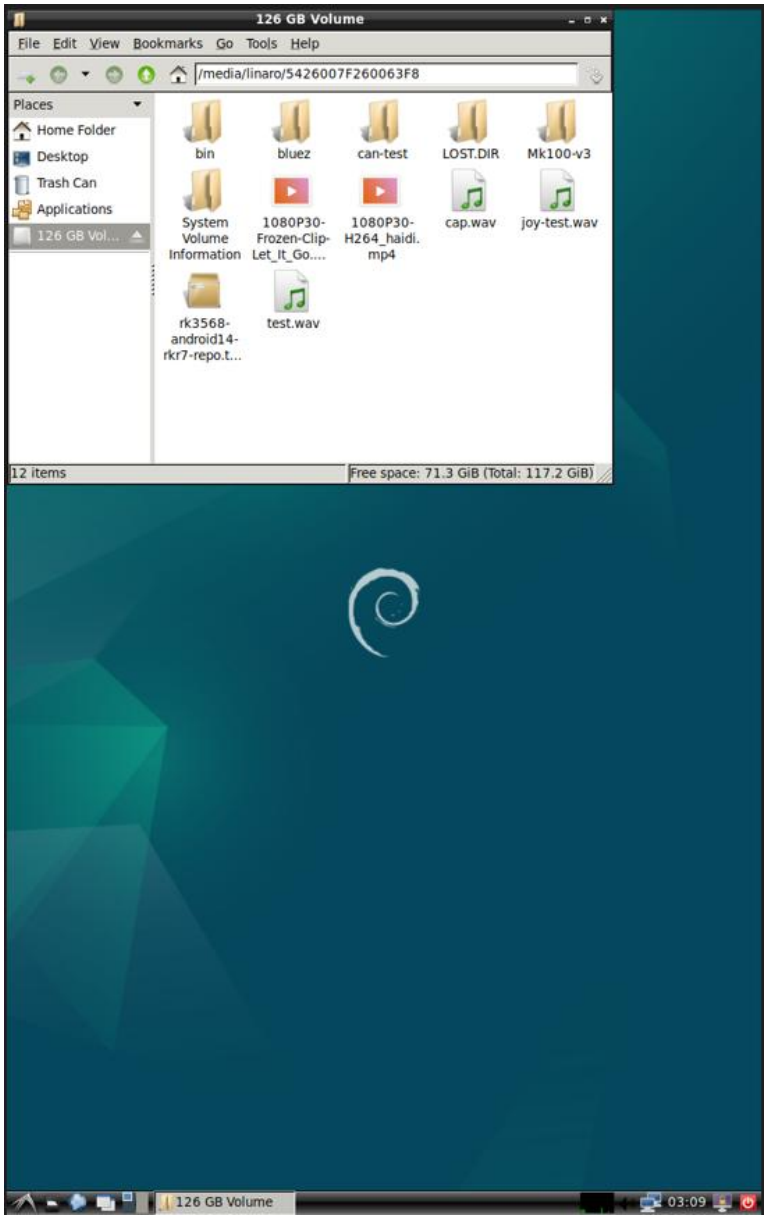
This system provides one USB2.0 OTG interface and one USB2.0 Host interface. Both support connection to common USB peripherals and development tools.

8.3.1 USB2.0 Host

The USB2.0 Host port can be used to connect USB peripherals such as a mouse, keyboard, flash drive, and other USB devices.

After connecting the USB flash drive, if the device is successfully recognized, an icon will appear on the desktop. Users need to click the icon in order to access the files on the device.





8.3.2 USB OTG(ADB)

By default, the OTG port operates in **Device mode**, allowing ADB connections for debugging.

To enable ADB on a Windows host:

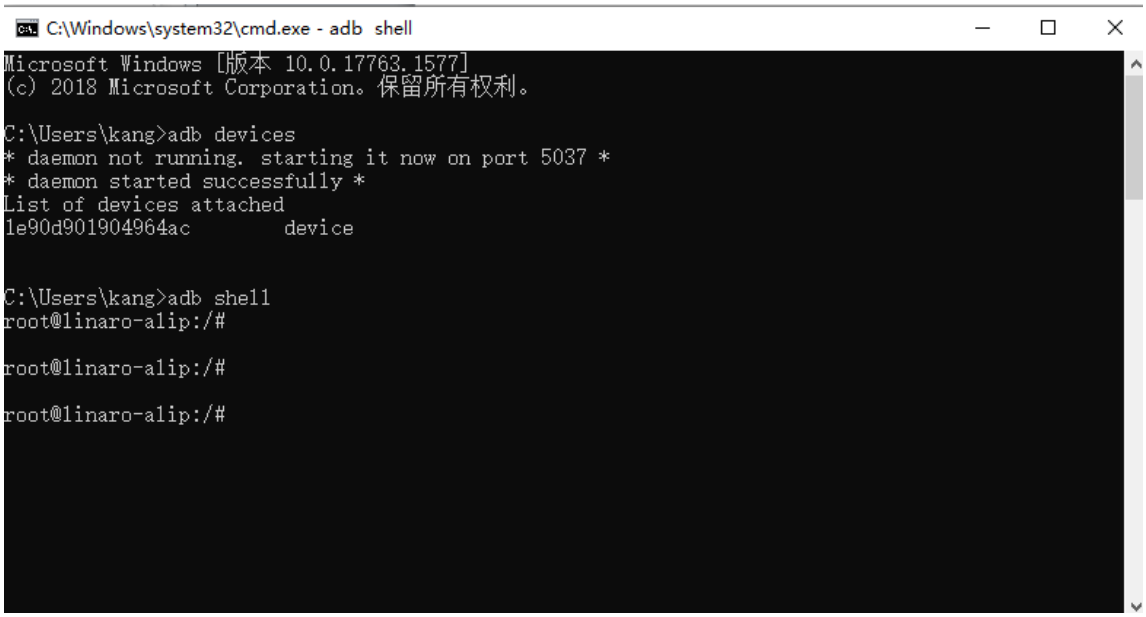
Step 1: Connect the board and PC host with Micro USB cable.

Step 2: Install ADB driver on Windows system.

Step 3: Press **Windows + R**, type cmd, and press **Enter** to open the command prompt

Step 4: Run the following command to check ADB connection.

```
# adb shell
```



8.4 Ethernet

Step 1: Connect the network cable to the Ethernet port.

According to the log, it can be seen that the Gigabit Ethernet recognition is successful.

```
root@linaro-alip:/# [  397.838650] rk_gmac-dwmac 21c70000.ethernet eth0: Link is Up - 1Gbps/Full - flow control
rx/tx
[  397.838728] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
```

Step 2: View network interface information.

```
# ifconfig
```

```
root@linaro-alip:/# ifconfig
eth0      Link encap:Ethernet  HWaddr FA:B2:1C:0E:F2:D4
          inet addr:192.168.0.124  Bcast:192.168.0.255  Mask:255.255.255.0
          inet6 addr: fe80::8f56:d77d:6c3a:38a2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:85 errors:0 dropped:16 overruns:0 frame:0
          TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7960 (7.7 KiB)  TX bytes:1410 (1.3 KiB)
          Interrupt:74
```

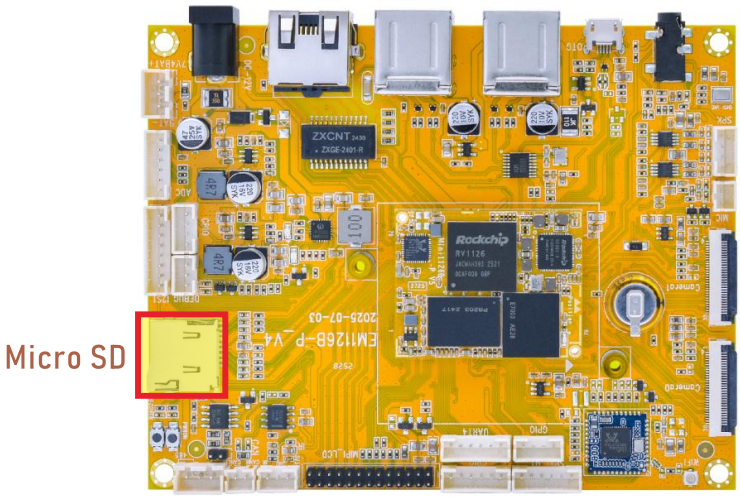
Step 3: Network connection test.

```
# ping -I eth0 www.armdesigner.com
```

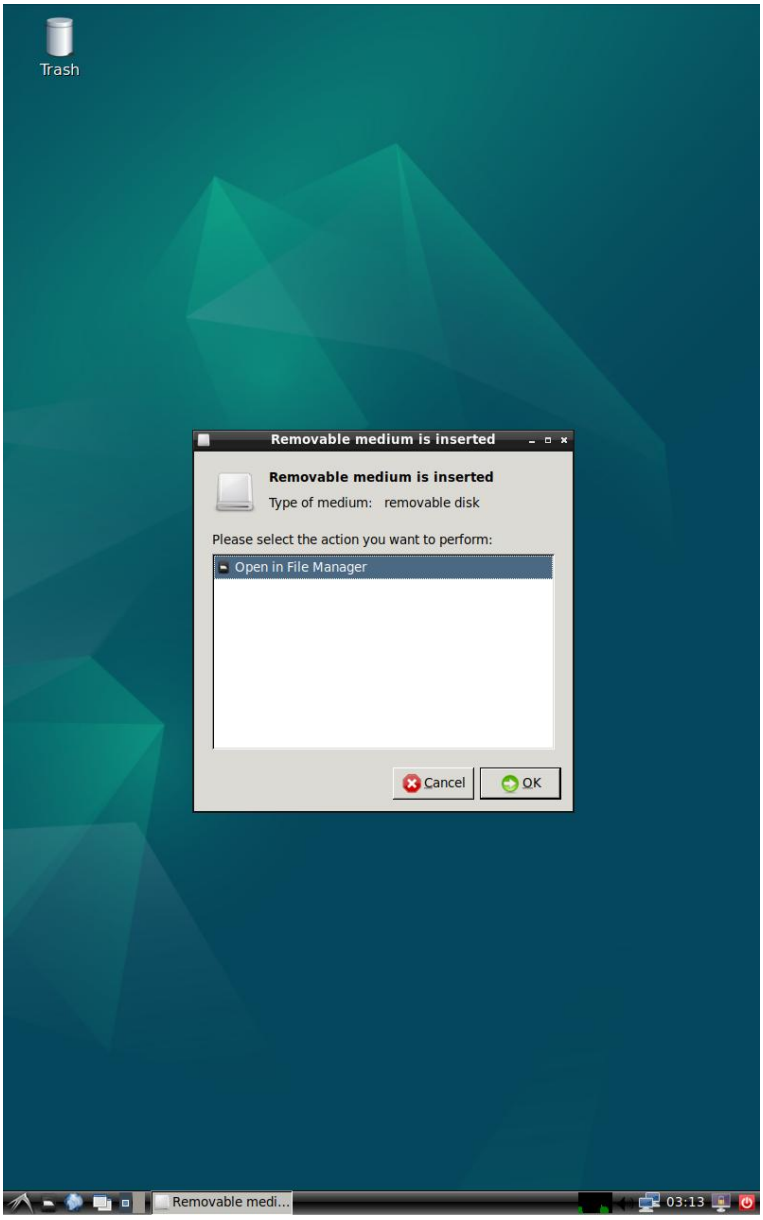
```
root@linaro-alip:/# ping -I eth0 www.armdesigner.com
PING www.armdesigner.com (67.222.54.196) from 192.168.0.124 eth0: 56(84) bytes of data.
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=1 ttl=50 time=196 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=2 ttl=50 time=191 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=3 ttl=50 time=194 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=4 ttl=50 time=193 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=5 ttl=50 time=191 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=6 ttl=50 time=194 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=7 ttl=50 time=191 ms
^C
--- www.armdesigner.com ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6008ms
rtt min/avg/max/mdev = 190.700/192.683/195.603/1.853 ms
```

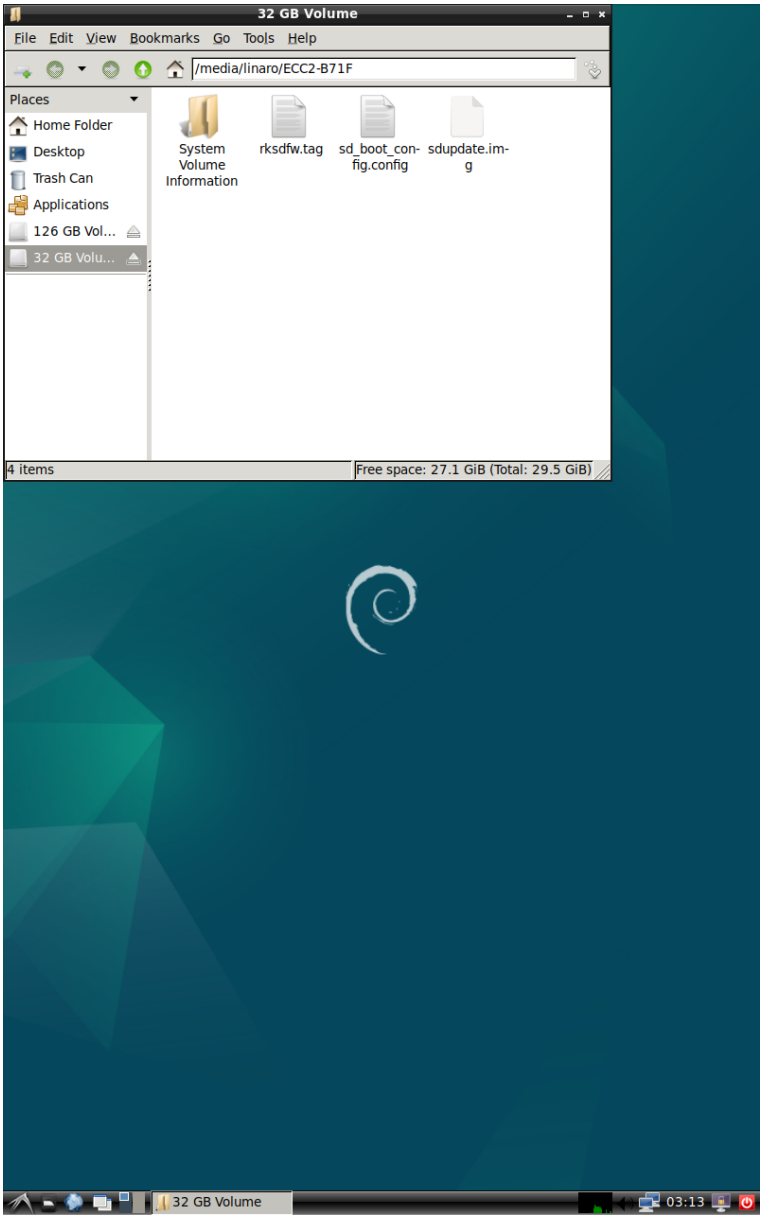
8.5 SD Card

Step 1: Insert the micro SD card into the card slot.



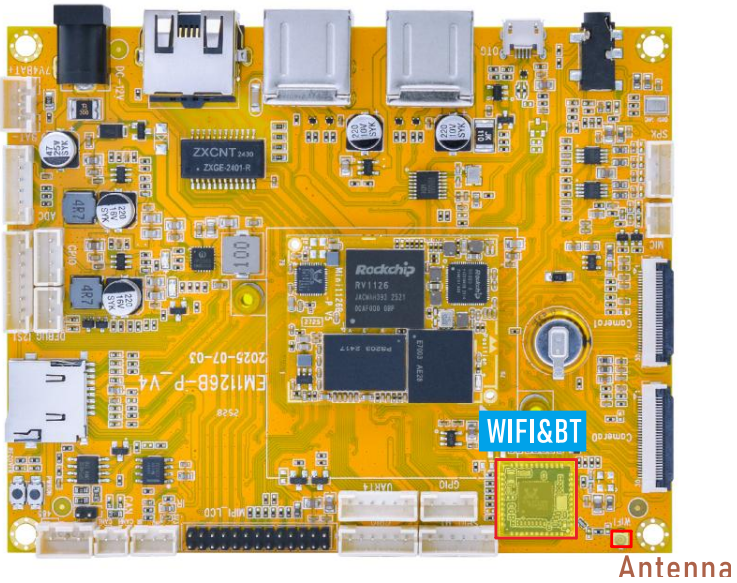
Step 2: After inserting the SD card, if it is recognized successfully, an icon will appear on the desktop. Users need to click the "ok button" in order to access the SD card.


Trash
Removable medium is inserted
Removable medium is inserted
Type of medium: removable disk
Please select the action you want to perform:
Open in File Manager
Cancel
OK
Removable medi...
03:13



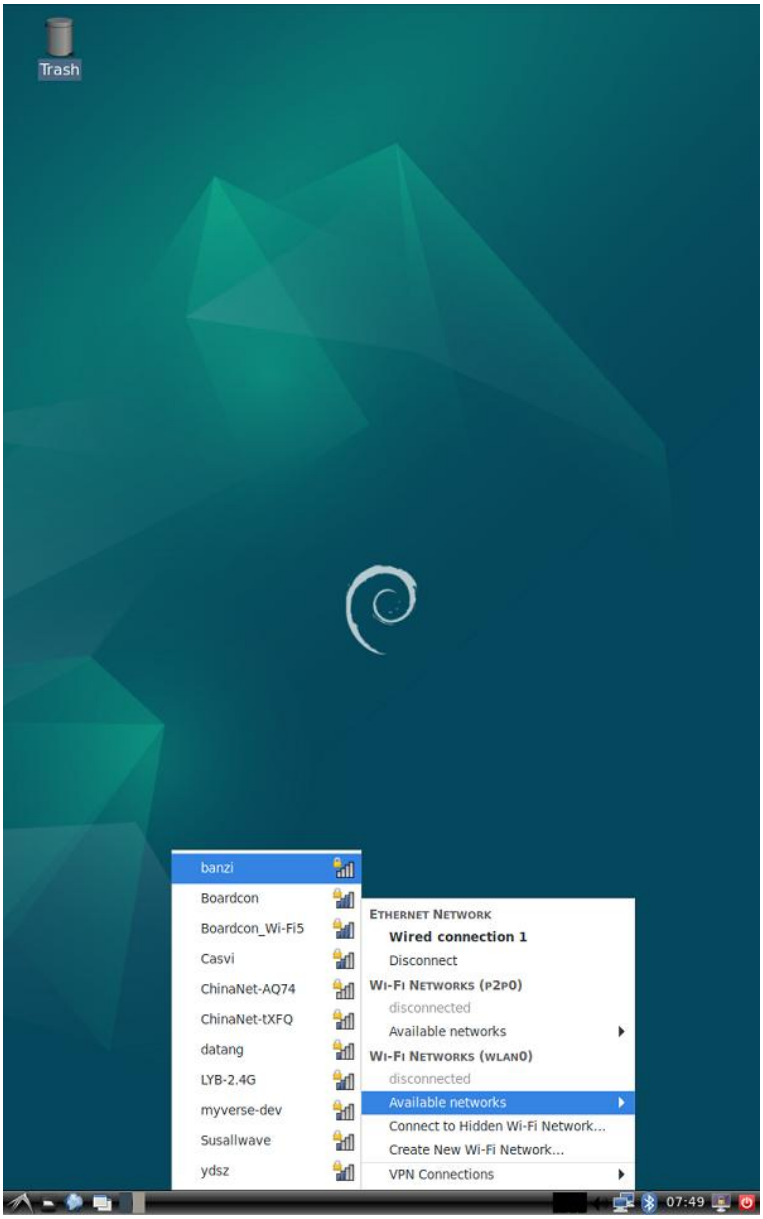
8.6 WiFi & Bluetooth

To use Wi-Fi and Bluetooth functions properly, the antenna needs to be connected.

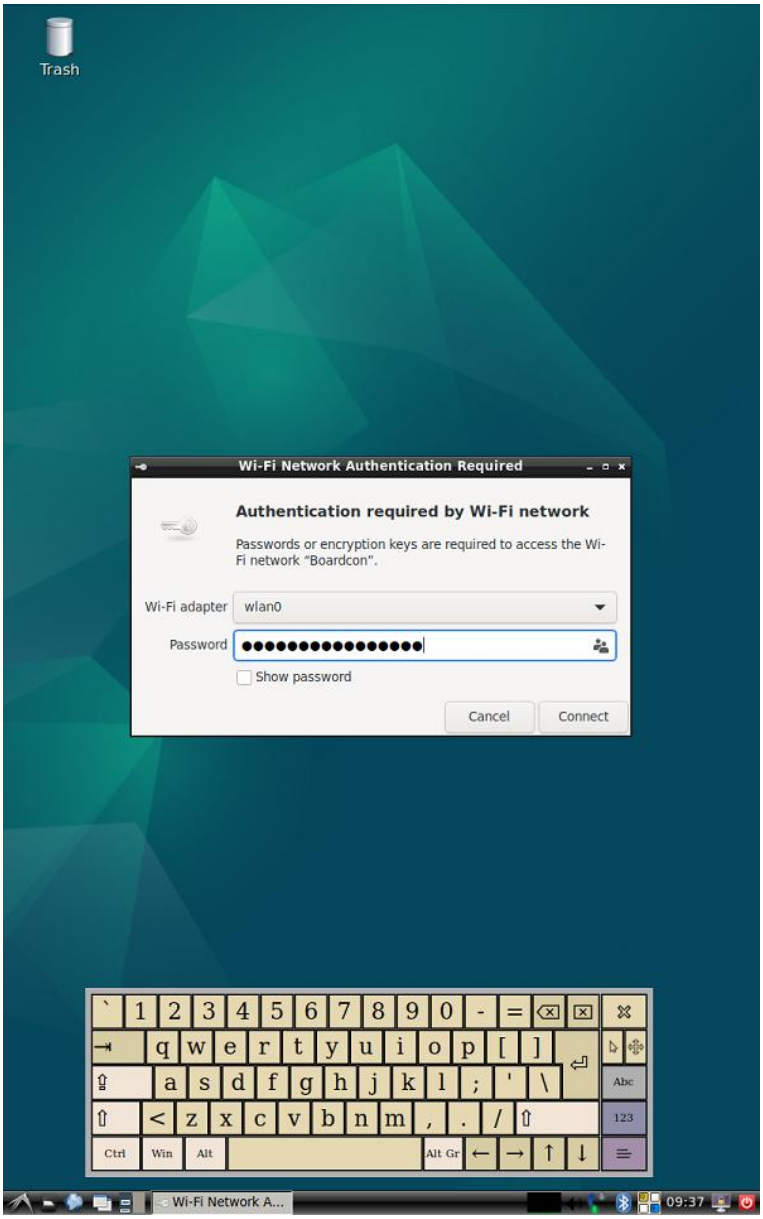


8.6.1 WiFi

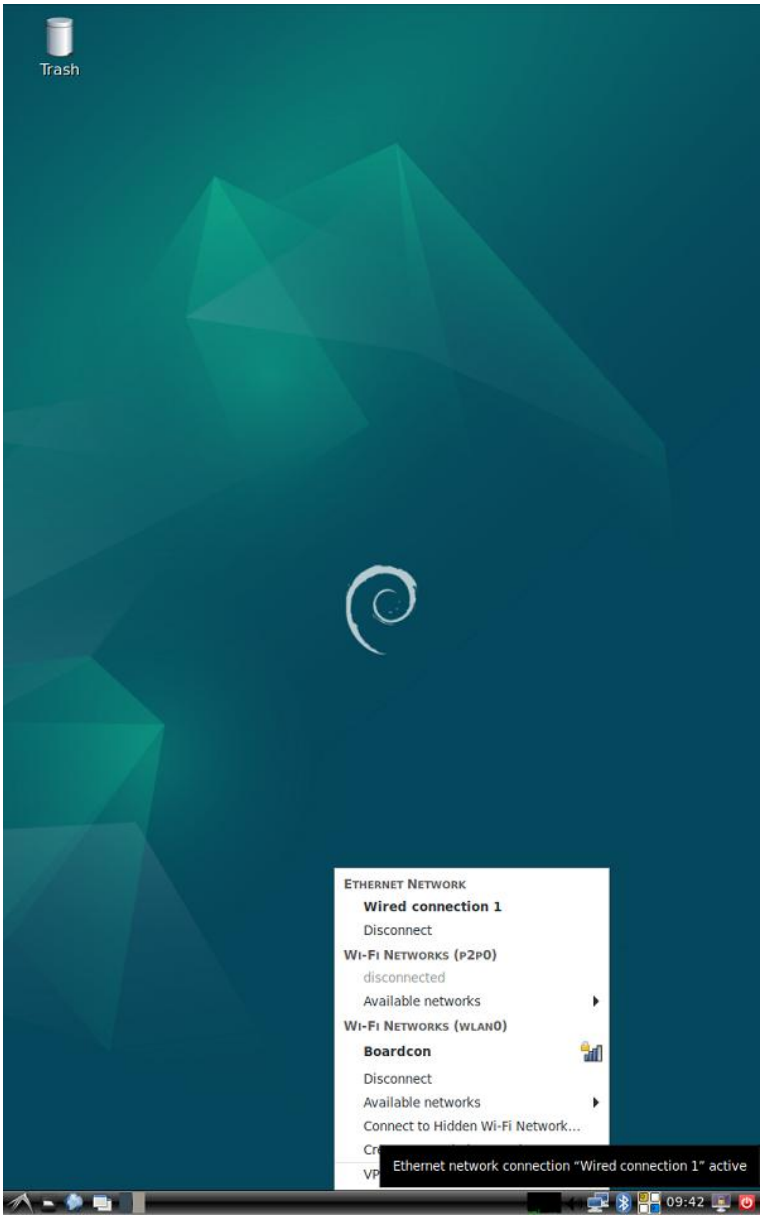
Step 1: Click the network icon in the top right corner of the interface, then select the "**Available Networks**" option to view the list of available hotspots.



Step 2: Select the SSID from the list of available networks and enter the password.



Step 3: After the WiFi successfully connects to the hotspot, the system will display the corresponding connection status icon in the top right corner.



Step 4: Users can test network connectivity using the desktop's built-in browser or verify it through the following command method.

(1) View network interface information.

```
# ifconfig
```

```
p2p0: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
       ether aa:b5:8e:b9:3b:7e  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

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
       inet 192.168.0.124  netmask 255.255.255.0  broadcast 192.168.0.255
       inet6 fe80::f2aa:dd95:fcf:85ab  prefixlen 64  scopeid 0x20<link>
       ether a8:b5:8e:b9:3b:7e  txqueuelen 1000  (Ethernet)
       RX packets 1673  bytes 206217 (201.3 KiB)
       RX errors 0  dropped 286  overruns 0  frame 0
       TX packets 37  bytes 4629 (4.5 KiB)
       TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

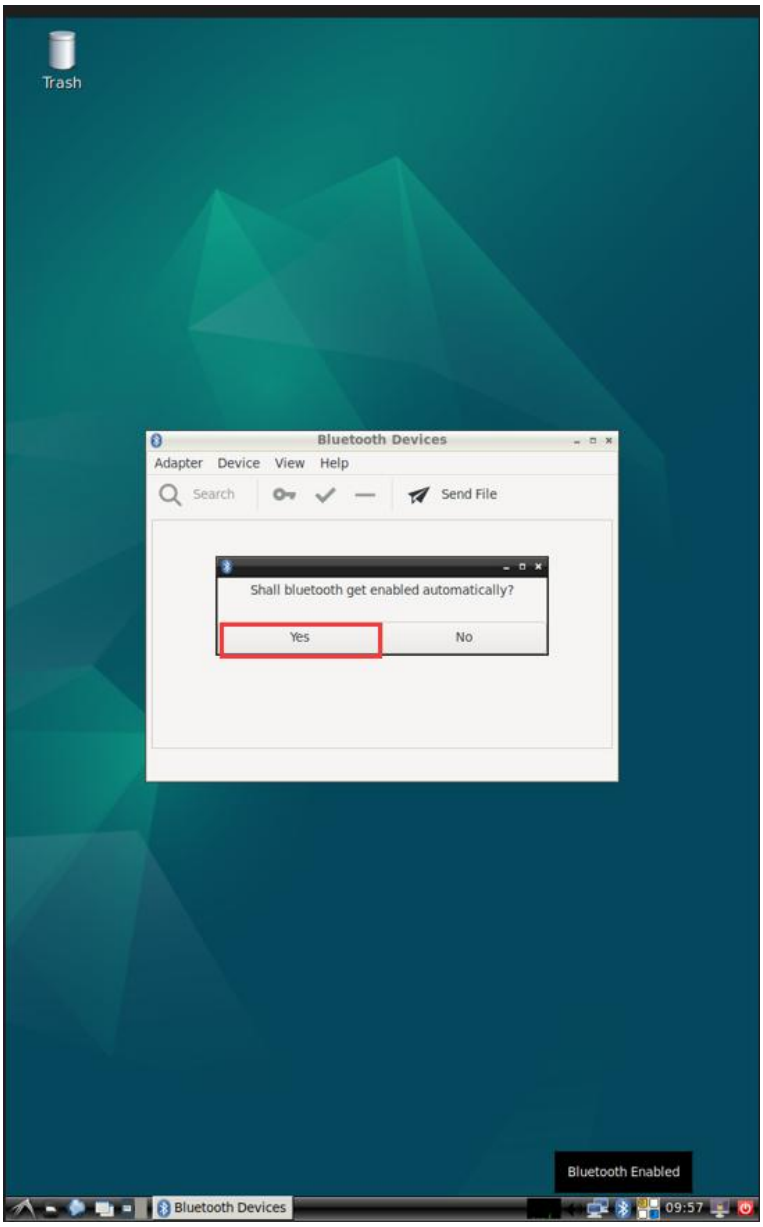
(2) Network connection test.

```
# ping -I wlan0 www.armdesigner.com
```

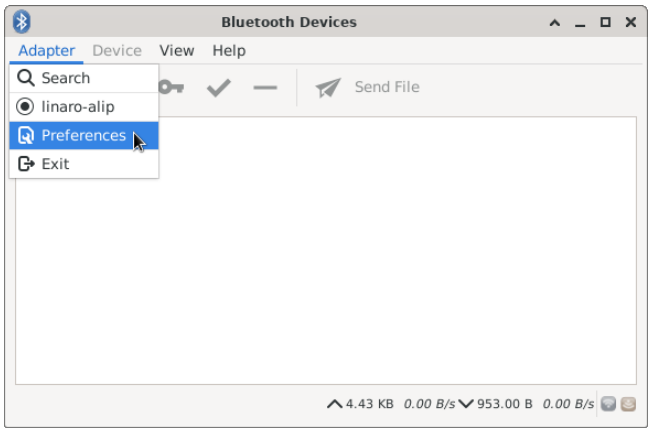
```
root@linaro-alip:/# ping -I wlan0 www.armdesigner.com
PING www.armdesigner.com (67.222.54.196) from 192.168.0.124 wlan0: 56(84) bytes of data.
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=1 ttl=50 time=227 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=2 ttl=50 time=216 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=3 ttl=50 time=243 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=4 ttl=50 time=529 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=5 ttl=50 time=279 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=6 ttl=50 time=219 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=7 ttl=50 time=251 ms
```

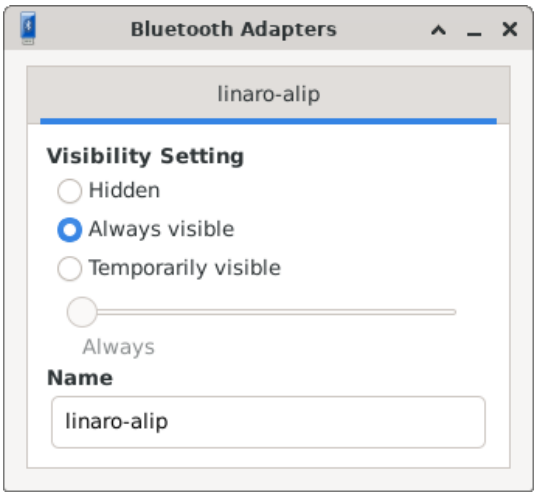
8.6.2 Bluetooth

Step 1: Click the Bluetooth icon in the top right corner of the desktop.

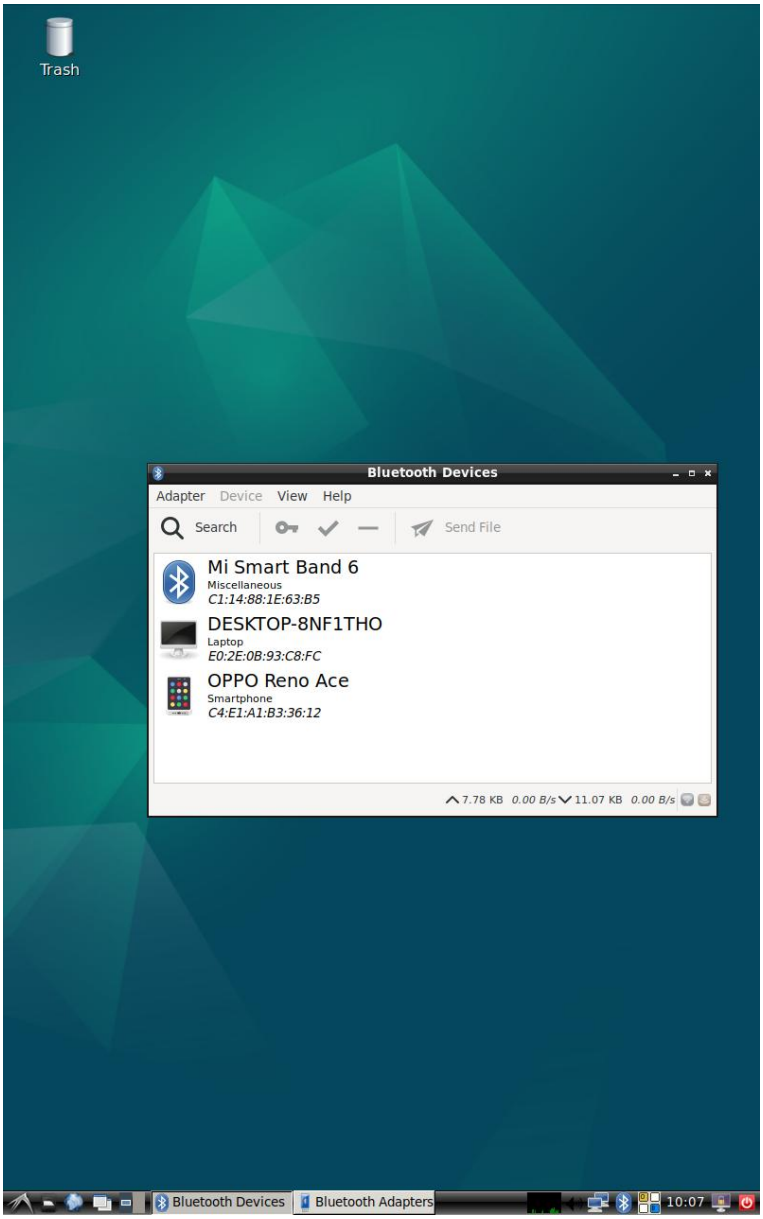


Step 2: The Bluetooth device name is hidden by default. Set it to be visible by clicking **Adapter** -> **Preferences** -> **Always visible**.





Step 3: Click Search to start searching and select the available device in the list to pair.

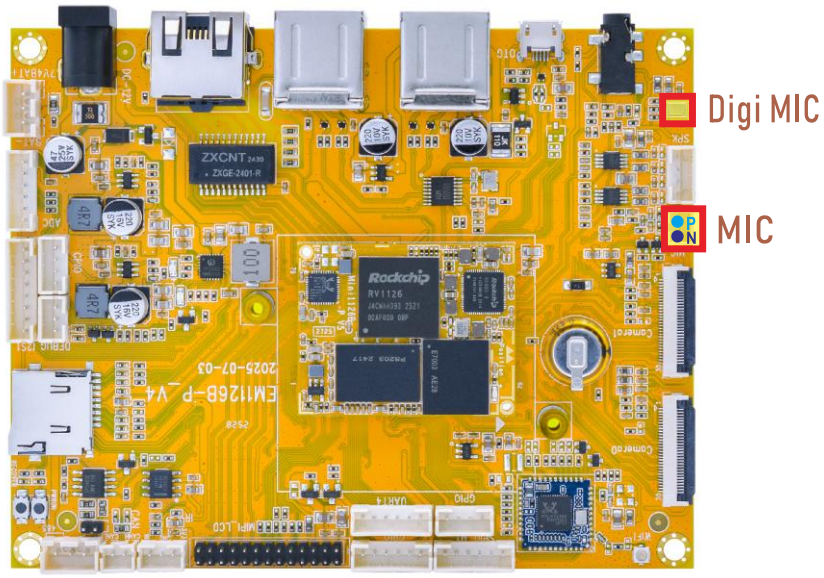


After successful configuration, Bluetooth devices can communicate with each other directly.

8.7 Audio

8.7.1 Audio Input

Step 1: Connect the MIC.



Step 2: View sound card.

```
# cat /proc/asound/cards
```

```
root@linaro-alip:/# cat /proc/asound/cards
 0 [rockchiprk809co]: simple-card - rockchip,rk809-codec
                      rockchip,rk809-codec
 1 [rockchippdmmica]: simple-card - rockchip,pdm-mic-array
                      rockchip,pdm-mic-array
```

Step 3: Record audio.

• Using the Analog MIC.

```
# amixer -c 0 cset numid=38 1
# arecord -Dhw:0,0 -f cd -t wav test.wav
```

```
root@linaro-alip:/# amixer -c 0 cset numid=38 1
numid=38,iface=MIXER,name='Capture MIC Path'
  ; type=ENUMERATED,access=rw------,values=1,items=4
  ; Item #0 'MIC OFF'
  ; Item #1 'Main Mic'
  ; Item #2 'Hands Free Mic'
  ; Item #3 'BT Sco Mic'
  : values=1
root@rv1126b-buildroot:/# arecord -Dhw:0,0 -f cd -t wav test.wav
Recording WAVE 'test.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

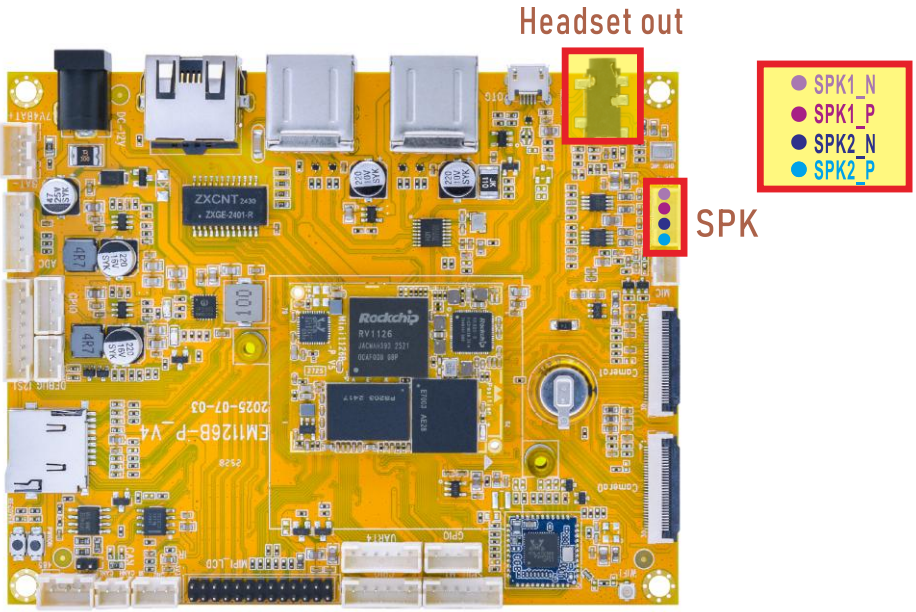
• Using the Digi MIC (PDM).

```
# arecord -D hw:1,0 -f s16_le -c 2 -r 16000 cap.wav
```

```
root@linaro-alip:/# arecord -D hw:1,0 -f s16_le -c 2 -r 16000 cap.wav
Recording WAVE 'cap.wav' : Signed 16 bit Little Endian, Rate 16000 Hz, Stereo
```

8.7.1 Audio Output

Step 1: Plug in the headset and connect the speaker.



Step 3: Play audio.

• Through the headset.

```
# amixer -c 0 cset numid=37 3
```

```
root@linaro-alip:/# amixer -c 0 cset numid=37 3
numid=37,iface=MIXER,name='Playback Path'
  ; type=ENUMERATED,access=rw------,values=1,items=11
  ; Item #0 'OFF'
  ; Item #1 'RCV'
  ; Item #2 'SPK'
  ; Item #3 'HP'
  ; Item #4 'HP_NO_MIC'
  ; Item #5 'BT'
  ; Item #6 'SPK_HP'
  ; Item #7 'RING_SPK'
  ; Item #8 'RING_HP'
  ; Item #9 'RING_HP_NO_MIC'
  ; Item #10 'RING_SPK_HP'
  : values=3
```

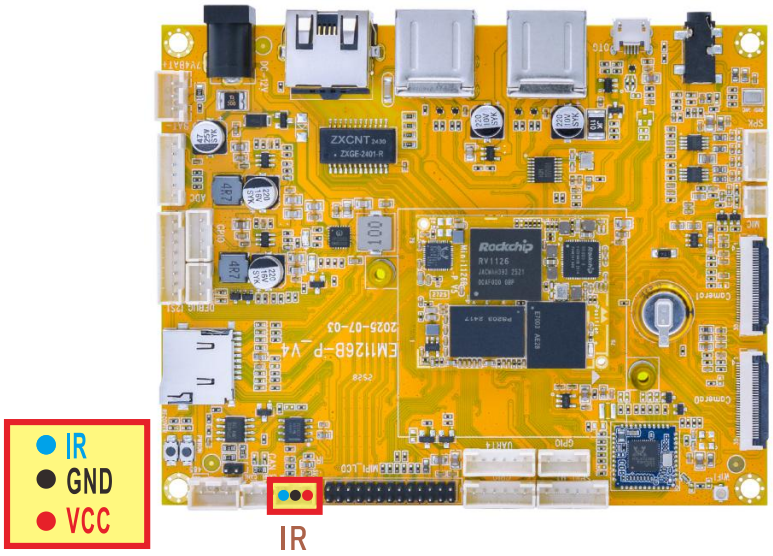
• Through the speaker.

```
# amixer -c 0 cset numid=37 2
```

```
root@linaro-alip:/# amixer -c 0 cset numid=37 2
numid=37,iface=MIXER,name='Playback Path'
  ; type=ENUMERATED,access=rw------,values=1,items=11
  ; Item #0 'OFF'
  ; Item #1 'RCV'
  ; Item #2 'SPK'
  ; Item #3 'HP'
  ; Item #4 'HP_NO_MIC'
  ; Item #5 'BT'
  ; Item #6 'SPK_HP'
  ; Item #7 'RING_SPK'
  ; Item #8 'RING_HP'
  ; Item #9 'RING_HP_NO_MIC'
  ; Item #10 'RING_SPK_HP'
  : values=2
```

8.8 IR

Step 1: Connect the IR receiver.



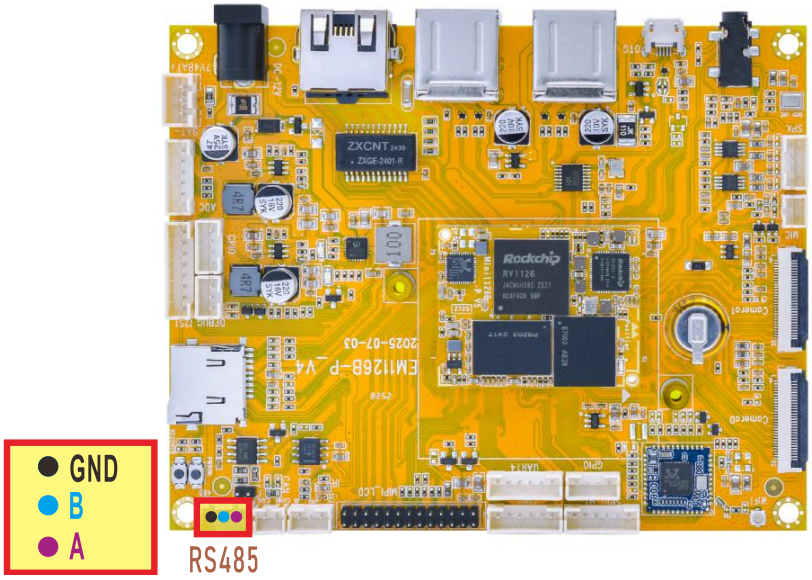
Step 2: Open IR debugging print.

```
# echo 1 > /sys/module/rockchip_pwm_remotectl/parameters/code_print
```

Step 3: When pressing a button on the remote control towards the IR receiver, the key value will be printed to the log.

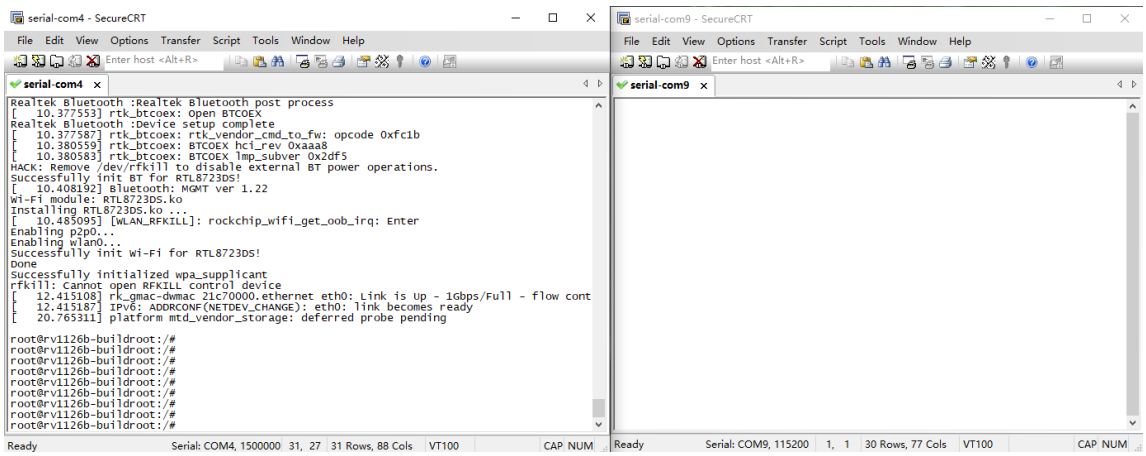
```
/code_print -alip:/# echo 1 > /sys/module/rockchip_pwm_remotectl/parameters/
root@linaro-alip:/#
root@linaro-alip:/# [  293.297790] USERCODE=0x1818
[  293.325015] RMC_GETDATA=9a
[  294.217858] USERCODE=0x1818
[  294.244995] RMC_GETDATA=99
[  294.837300] USERCODE=0x1818
[  294.864505] RMC_GETDATA=98
[  295.241483] USERCODE=0x1818
[  295.268684] RMC_GETDATA=9b
[  296.021481] USERCODE=0x1818
[  296.048663] RMC_GETDATA=e6
[  296.597978] USERCODE=0x1818
[  296.625116] RMC_GETDATA=e7
[  296.953745] USERCODE=0x1818
[  296.980882] RMC_GETDATA=e4
[  297.242006] USERCODE=0x1818
[  297.269143] RMC_GETDATA=e5
```

8.9 RS485



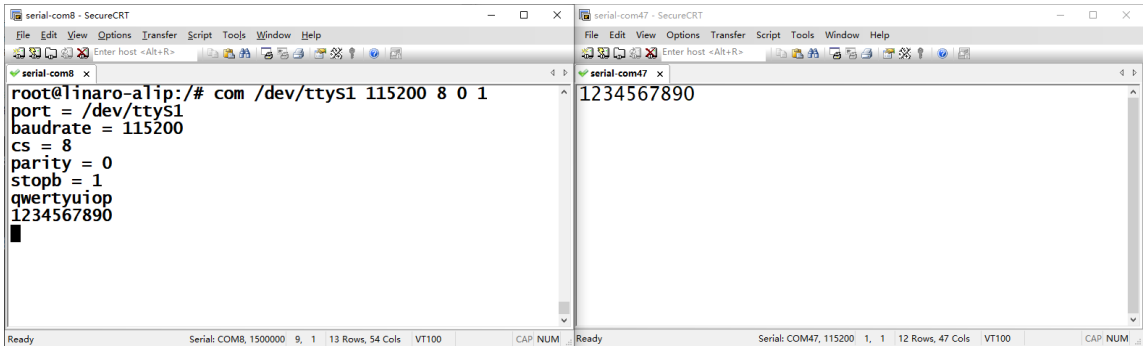
Step 1: As shown in the diagram, connect the RS485 test tool to the development board.

Step 2: Open the corresponding serial terminal, set the baud rate of the board to 1500000, and set the baud rate of the RS485 test tool to 115200.



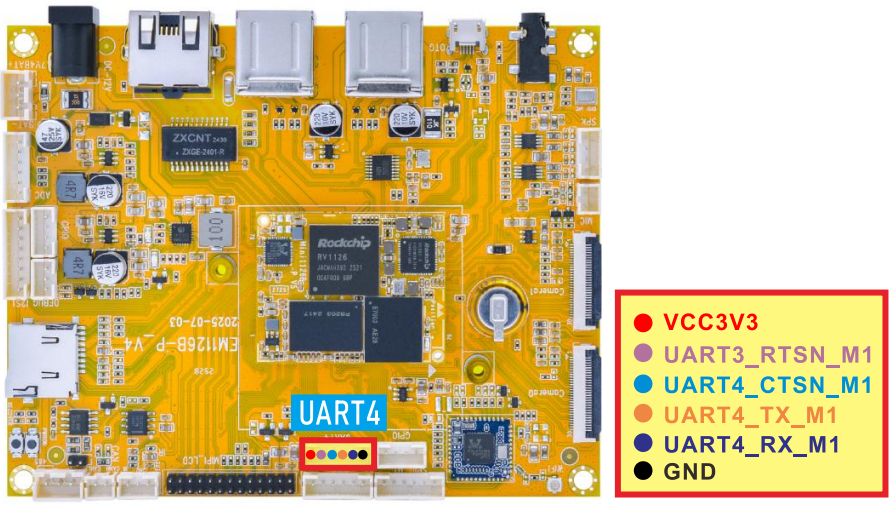
Step 3: Execute the following command on the board to test the RS485 transmission and reception functionality.

```
# com /dev/ttyS1 115200 8 0 1
```



8.10 UART

Step 1: Short circuit **UART4_RX_M1** and **UART4_TX_M1** pins of UART.

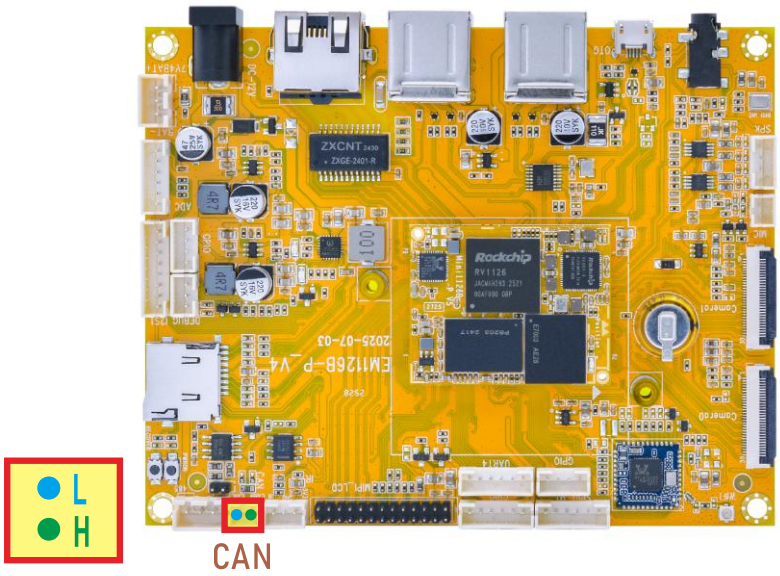


Step 2: UART1 test.

```
# com /dev/ttyS4 115200 8 0 1
```

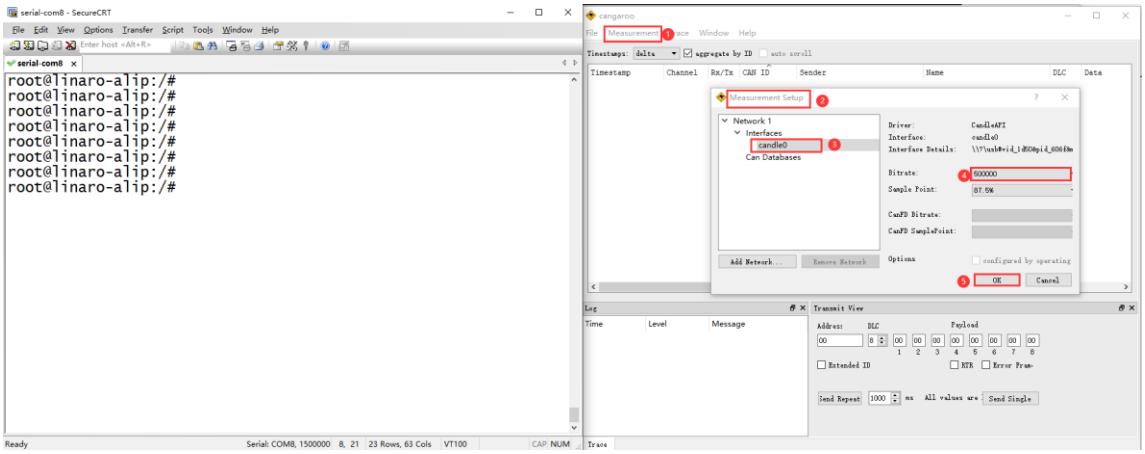
```
root@linaro-alip:/# com /dev/ttyS4 115200 8 0 1
port = /dev/ttyS4
baudrate = 115200
cs = 8
parity = 0
stopb = 1
JKJKJKJKJKJK
RECV: JKJKJKJKJKJK
89898989PP
RECV: 89898989PP
2255UU
RECV: 2255UU
YTYT
RECV: YTYT
00000
RECV: 00000
```

8.11 CAN



Step 1: Connect the CAN test tool to the board as shown in the diagram below.

Step 2: Open the CAN test software and set the baud rate to 500000.

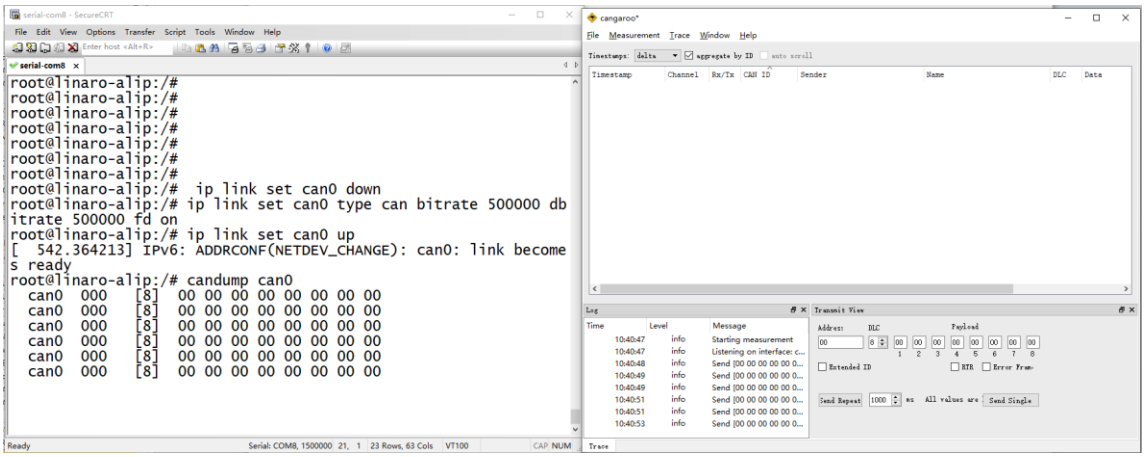


Step 3: Configure and activate the CAN network, setting the bitrate to 500000.

```
# ip link set can0 down
# ip link set can0 type can bitrate 500000 dbitrate 500000 fd on
# ip link set can0 up
```

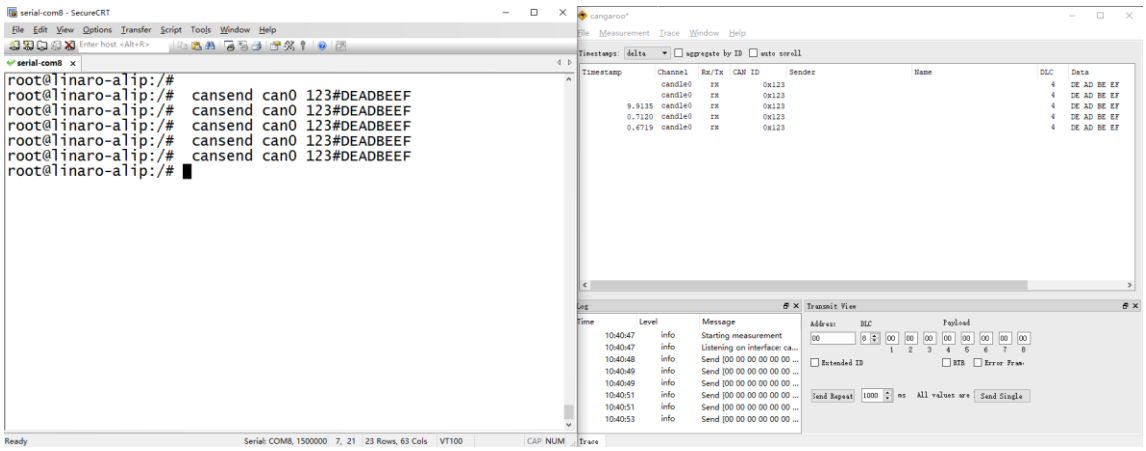
Step 4: Configure CAN as the receiver.

```
# candump can0
```



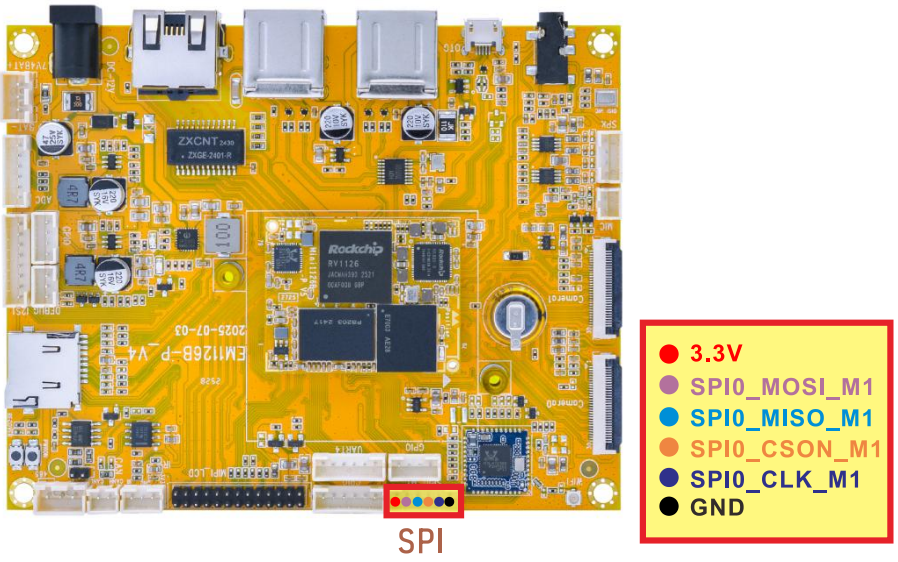
Step 5: Configure CAN as the sender.

```
# cansend can0 123#DEADBEEF
```



8.12 SPI

Step 1: Short circuit **SPI0_MOSI_M1** and **SPI0_MISO_M1** pins of SPI.

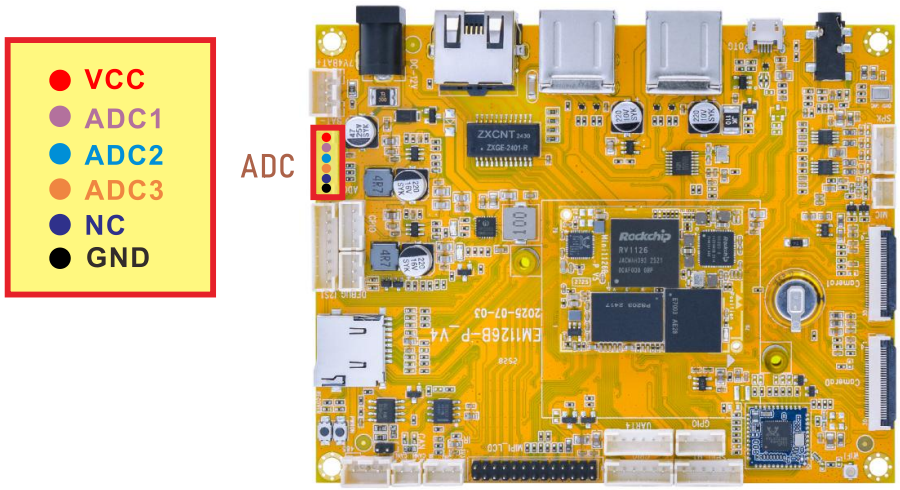


Step 2: SPI test:

```
# spidev-test
```

```
root@linaro-alip:/# spidev-test
spi mode: 0
bits per word: 8
max speed: 10000000 Hz (10000 KHz)

EE FF FF FF FF FF
40 00 00 00 00 95
FF FF FF FF FF FF
FF FF FF FF FF FF
FF FF FF FF FF FF
DE AD BE EF BA AD
F0 0D
```

8.13 ADC



Connect the ADC pin (ADC1/ADC2/ADC3) to high and low levels respectively. After shorting the pin to the desired level, run the following command.

Example for **ADC1**:

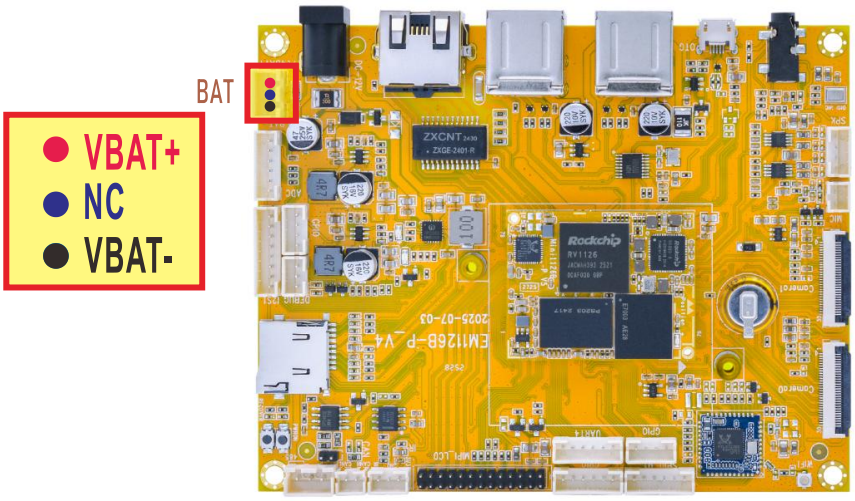
```
# cat /sys/bus/iio/devices/iio:\device0/in_voltage1_raw
```

```
root@linaro-alip:/# cat /sys/bus/iio/devices/iio:\device0/in_voltage1_raw
42
root@linaro-alip:/# cat /sys/bus/iio/devices/iio:\device0/in_voltage1_raw
8191
```

Repeat with in_voltage2_raw or in_voltage3_raw for **ADC2/ADC3**.

8.14 Battery supply

Step 1: Connect the 7.4V Li-ion battery.

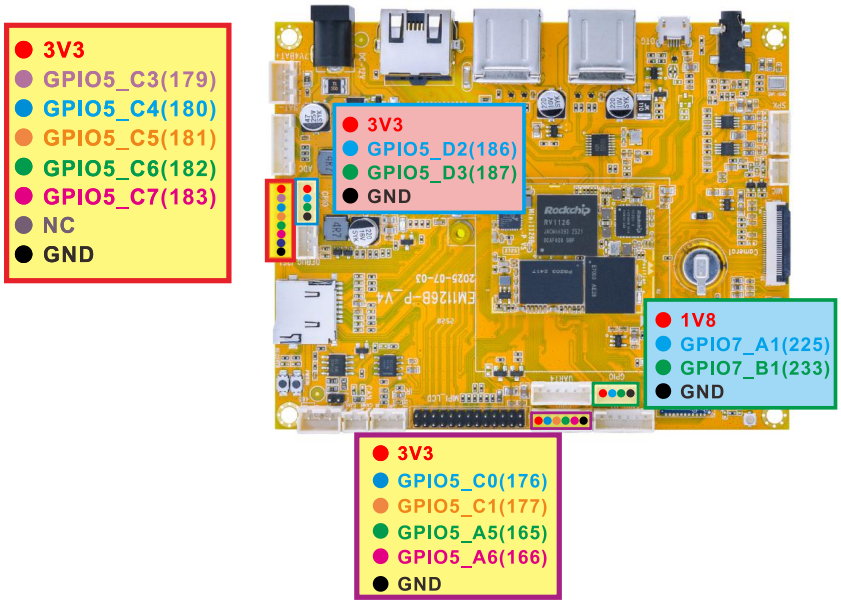


Step 2: Run the following command to check the current voltage:

```
# cat /sys/class/power_supply/battery/voltage_now
```

```
root@linaro-alip:/# cat /sys/class/power_supply/battery/voltage_now
7856000
root@linaro-alip:/# cat /sys/class/power_supply/battery/voltage_now
7832000
root@linaro-alip:/# cat /sys/class/power_supply/battery/voltage_now
7824000
```

8.15 GPIO



General GPIO Control Command Guide：

```
# Export the GPIO and set it as output
gpio_ctrl.sh x export

# Set output to high level
gpio_ctrl.sh x set 1

# Read the current GPIO level
gpio_ctrl.sh x get

# Set output to low level (should be 1)
gpio_ctrl.sh x set 0

# Read the GPIO level again (should be 0)
gpio_ctrl.sh x get

# Unexport GPIO
gpio_ctrl.sh x unexport
```

For example, using

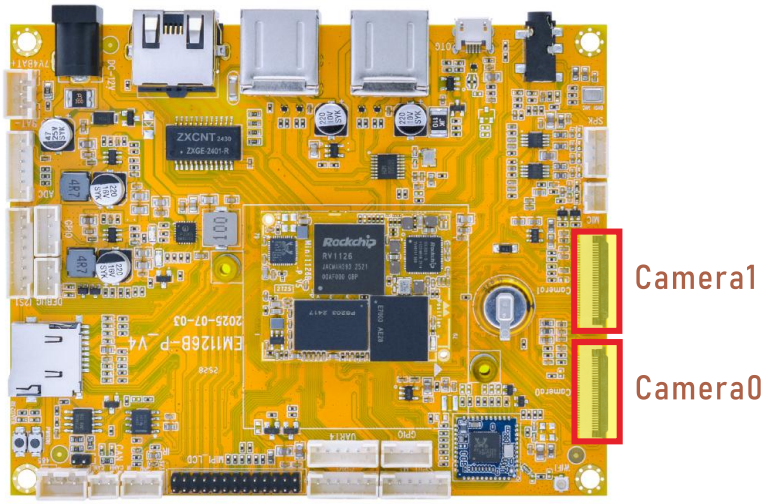
GPIO-186:

```
# gpio_ctrl.sh 186 export
# gpio_ctrl.sh 186 set 1
# gpio_ctrl.sh 186 get
# gpio_ctrl.sh 186 set 0
# gpio_ctrl.sh 186 get
# gpio_ctrl.sh 186 unexport
```

```
root@linaro-alip:/# gpio_ctrl.sh 186 export
GPIO186 exported as out
root@linaro-alip:/# gpio_ctrl.sh 186 set 1
GPIO186 set to 1
root@linaro-alip:/# gpio_ctrl.sh 186 get
GPIO186 value: 1
root@linaro-alip:/# gpio_ctrl.sh 186 set 0
GPIO186 set to 0
root@linaro-alip:/# gpio_ctrl.sh 186 get
GPIO186 value: 0
root@linaro-alip:/# gpio_ctrl.sh 186 unexport
GPIO186 unexported
```

8.16 Camera

Step 1: Power on after connecting the camera.



Step 2: Preview camera0.

```
# ./rockchip-test/camera/boardcon-camera0-test.sh
```

```
root@linaro-alip:/# ./rockchip-test/camera/boardcon-camera0-test.sh
[  469.662329] dw9714 1-000c: cmd 0x80685600 not supported
[  469.662978] dw9714 3-000c: cmd 0x80685600 not supported
Setting pipeline to PAUSED ...
Using mplane plugin for capture
Pipeline is live and does not need PREROLL ...
/GstPipeline:pipeline0/GstKMSSink:kmssink0: display-width = 800
/GstPipeline:pipeline0/GstKMSSink:kmssink0: display-height = 1280
Pipeline is PREROLLED ...
Setting pipeline to PLAYING ...
/GstPipeline:pipeline0/GstV4l2Src:v4l2src0: crop-bounds = < (int)0, (int)0, (int)3840, (int)2160 >
New clock: GstSystemClock
/GstPipeline:pipeline0/GstV4l2Src:v4l2src0.GstPad:src:    caps    =    video/x-raw,    format=(string)NV12,
width=(int)1920,    height=(int)1080,    framerate=(fraction)30/1,    interlace-mode=(string)progressive,
colorimetry=(string)1:3:5:1
/GstPipeline:pipeline0/GstCapsFilter:capsfilter0.GstPad:src:  caps  =  video/x-raw,  format=(string)NV12,
width=(int)1920,    height=(int)1080,    framerate=(fraction)30/1,    interlace-mode=(string)progressive,
colorimetry=(string)1:3:5:1
/GstPipeline:pipeline0/GstKMSSink:kmssink0.GstPad:sink:    caps    =    video/x-raw,    format=(string)NV12,
width=(int)1920,    height=(int)1080,    framerate=(fraction)30/1,    interlace-mode=(string)progressive,
colorimetry=(string)1:3:5:1
[  469.711515] rkisp_hw 21d00000.isp: set isp clk = 500000000Hz
/GstPipeline:pipeline0/GstCapsFilter:capsfilter0.GstPad:sink:  caps  =  video/x-raw,  format=(string)NV12,
width=(int)1920,    height=(int)1080,    framerate=(fraction)30/1,    interlace-mode=(string)progressive,
colorimetry=(string)1:3:5:1
[  469.721459] rkcif-mipi-lvds: stream[0] start streaming
[       469.721845]    rockchip-mipi-csi2    mipi0-csi2:    stream    on,    src_sd:    00000000791e80fd,
sd_name:rockchip-csi2-dphy0
[  469.721875] rockchip-mipi-csi2 mipi0-csi2: stream ON
[  469.721947] rockchip-csi2-dphy0: dphy0, data_rate_mbps 892
[  469.721985] rockchip-csi2-dphy csi2-dphy0: csi2_dphy_s_stream stream on:1, dphy0, ret 0
[  469.722002] imx415 1-0036: s_stream: 1. 3864x2192, hdr: 0, bpp: 10
[  469.831647] rkcif-mipi-lvds: Warning: vblank need >= 1000us if isp work in online, cur 859 us
[  469.866749] rkcif-mipi-lvds: Warning: vblank need >= 1000us if isp work in online, cur 859 us
Redistribute latency...
0:00:05.7 / 99:99:99.
```

Step 3: Preview camera1.

```
# ./rockchip-test/camera/boardcon-camera1-test.sh
```

```
root@linaro-alip:/# ./rockchip-test/camera/boardcon-camera1-test.sh
[  515.198932] dw9714 1-000c: cmd 0x80685600 not supported
[  515.199450] dw9714 3-000c: cmd 0x80685600 not supported
Setting pipeline to PAUSED ...
Using mplane plugin for capture
Pipeline is live and does not need PREROLL ...
/GstPipeline:pipeline0/GstKMSSink:kmssink0: display-width = 800
/GstPipeline:pipeline0/GstKMSSink:kmssink0: display-height = 1280
Pipeline is PREROLLED ...
Setting pipeline to PLAYING ...
/GstPipeline:pipeline0/GstV4l2Src:v4l2src0: crop-bounds = < (int)0, (int)0, (int)3840, (int)2160 >
New clock: GstSystemClock
/GstPipeline:pipeline0/GstV4l2Src:v4l2src0.GstPad:src:    caps    =    video/x-raw,    format=(string)NV12,
width=(int)1920,    height=(int)1080,    framerate=(fraction)30/1,    interlace-mode=(string)progressive,
colorimetry=(string)bt709
/GstPipeline:pipeline0/GstCapsFilter:capsfilter0.GstPad:src:  caps  =  video/x-raw,  format=(string)NV12,
width=(int)1920,    height=(int)1080,    framerate=(fraction)30/1,    interlace-mode=(string)progressive,
colorimetry=(string)bt709
/GstPipeline:pipeline0/GstKMSSink:kmssink0.GstPad:sink:    caps    =    video/x-raw,    format=(string)NV12,
width=(int)1920,    height=(int)1080,    framerate=(fraction)30/1,    interlace-mode=(string)progressive,
colorimetry=(string)bt709
[  515.246746] rkisp_hw 21d00000.isp: set isp clk = 500000000Hz
/GstPipeline:pipeline0/GstCapsFilter:capsfilter0.GstPad:sink:  caps  =  video/x-raw,  format=(string)NV12,
width=(int)1920,    height=(int)1080,    framerate=(fraction)30/1,    interlace-mode=(string)progressive,
colorimetry=(string)bt709
[  515.263727] rkisp rkisp-vir1: first params buf queue
[  515.263997] rkcif-mipi-lvds2: stream[0] start streaming
[       515.264591]    rockchip-mipi-csi2    mipi2-csi2:    stream    on,    src_sd:    00000000044cba21,
sd_name:rockchip-csi2-dphy3
[  515.264632] rockchip-mipi-csi2 mipi2-csi2: stream ON
[  515.264698] rockchip-csi2-dphy3: dphy3, data_rate_mbps 892
[  515.264736] rockchip-csi2-dphy csi2-dphy3: csi2_dphy_s_stream stream on:1, dphy3, ret 0
[  515.264750] imx415 3-0036: s_stream: 1. 3864x2192, hdr: 0, bpp: 10
[  515.374070] rkcif-mipi-lvds2: Warning: vblank need >= 1000us if isp work in online, cur 859 us
[  515.409200] rkcif-mipi-lvds2: Warning: vblank need >= 1000us if isp work in online, cur 859 us
Redistribute latency...
0:00:05.9 / 99:99:99.
```

8.17 Video Playback

(1) The directory for the built-in video testing scripts in the system: */rockchip-test/video*

```
root@linaro-alip:/#  ls /rockchip-test/video/
test_dec-gst.sh     test_enc-gst.sh         test_gst_video_maxfps.sh
test_dec-mpv.sh     test_gst_multivideo.sh  video_stresstest.sh
test_dec-parole.sh  test_gst_video.sh       video_test.sh
```

Simply execute the script.

```
root@linaro-alip:/# /rockchip-test/video/test_gst_video.sh
Setting pipeline to PAUSED ...
Pipeline is PREROLLING ...
Redistribute latency...
mpp[2644]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[2644]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[2644]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[2644]: mpp: unable to create enc vp8 for soc rv1126b unsupported
mpp[2644]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[2644]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
Redistribute latency...
mpp[2644]: h264d_api: is_avcC=1
mpp[2644]: mpp_buf_slot: mismatch h_stride_by_pixel 1472 - 1280
mpp[2644]: mpp_buf_slot: mismatch h_stride_by_byte 1472 - 1280
mpp[2644]: mpp_buf_slot: mismatch size_total 1589760 - 1843200
mpp[2644]: mpp_buf_slot: mismatch h_stride_by_pixel 1472 - 1280
mpp[2644]: mpp_buf_slot: mismatch h_stride_by_byte 1472 - 1280
mpp[2644]: mpp_buf_slot: mismatch size_total 1589760 - 1843200
Pipeline is PREROLLED ...
Prerolled, waiting for async message to finish...
Setting pipeline to PLAYING ...
Redistribute latency...
New clock: GstSystemClock
0:00:02.3 / 0:00:29.5 (8.0 %)
```

(2) Use the gst-play-1.0 command to play the video.

```
# amixer -c 0 cset numid=37 2
# gst-play-1.0 --videosink=kmssink
/media/linaro/5426007F260063F8/1080P30-H264_haidi.mp4 --audiosink="alsasink
```

Command explanation:

• amixer -c 0 cset numid=37 2: Audio through the speaker.

•/media/linaro/5426007F260063F8/1080P30-H264_haidi.mp4: The media file path to be played.

```
root@linaro-alip:/# gst-play-1.0 --videosink=kmssink /media/linaro/5426007F260063F8/1080P30-H264_haidi.mp4
--audiosink="alsasink device=hw:0,0"
Press 'k' to see a list of keyboard shortcuts.
Now playing /media/linaro/5426007F260063F8/1080P30-H264_haidi.mp4
Redistribute latency...
mpp[3600]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[3600]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[3600]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[3600]: mpp: unable to create enc vp8 for soc rv1126b unsupported
mpp[3600]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
mpp[3600]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen   2025-07-01 fix[base]: Fix enc cfg size
Redistribute latency...
mpp[3600]: h264d_api: is_avcC=1
mpp[3600]: mpp_buf_slot: mismatch h_stride_by_pixel 1984 - 1920
mpp[3600]: mpp_buf_slot: mismatch h_stride_by_byte 1984 - 1920
mpp[3600]: mpp_buf_slot: mismatch size_total 3237888 - 4177920
mpp[3600]: mpp_buf_slot: mismatch h_stride_by_pixel 1984 - 1920
mpp[3600]: mpp_buf_slot: mismatch h_stride_by_byte 1984 - 1920
mpp[3600]: mpp_buf_slot: mismatch size_total 3237888 - 4177920
Redistribute latency...
Redistribute latency...
Redistribute latency...
0:01:26.2 / 0:05:07.2
```