

Idea1126B-P Linux6.1 User Manual

V2.0



Boardcon Embedded Designer

Overview

The content of this document is intended solely for the Idea1126B-P development board, aiming to help users quickly understand, apply, and test the Idea1126B-P development board.

System Support

Development Board	Debian12	Buildroot
CM1126B-P V2	Y	Y
Idea1126B-P V2		

Revision History

Version	Date	Author	Revision History
V1.0	2025-07-18	Liu Yuan	Initial version
V2.0	2026-03-09	Chen Yu	Add Debian 12 and Buildroot IPC board support.

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

2508 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	5
1.1 Overview	5
1.2 Product Parameters	5
1.3 Hardware Interface Introduction	7
2.Install Drivers and Tool	9
2.1 Install RK Driver Assitant	9
2.2 Install CH9102X Driver	11
2.2.1 How to Connect the Serial Port Tool	11
2.2.2 Install Driver	11
2.3 Install Serial Terminal Tool	12
3.Upgrade Introduction	14
3.1 Upgrade Mode	14
3.1.1 How to Enter Loader Mode	15
3.1.2 How to Enter MaskRom Mode	16
3.2 Burn firmware	17
3.2.1 Burn Update.img Firmware	17
3.2.2 Burn Split Firmware	18
4.Development Environment	20
4.1 Preparing the Development Environment	20
4.2 Installing Libraries and Toolkits	20
5.Compile Source	21
6.Buildroot Test	25
6.1 Serial Terminal	25
6.2 Display	25
6.3 USB	26

6.3.1 USB2.0 Host	26
6.3.2 USB2.0 OTG (Default Device Mode)	27
6.4 Ethernet	29
6.5 SD Card	30
6.6 Relay	30
6.7 WiFi & BLUETOOTH	31
6.8 RF315	35
6.9 Audio	36
6.10 PIR	38
6.11 IR	39
6.12 Photovaristor	40
6.13 LED	40
6.14 GPIO	41
6.15 RS485	42
6.16 CAN	43
6.17 UART5	45
6.18 SPI	46
6.19 Wiegand	47
6.20 Camera	48
6.21 Video Playback	50
7.Buildroot IPC Test	52
7.1 Display	0
7.2 RKIPC	1
7.2.1 Audio (MIC) test	1
7.2.2 RTSP	3
7.2.3 Web Live Streaming	4
7.3 Camera Streaming	5
7.4 Video Decoder (VDEC) Test	8
8.Debian12 Test	11

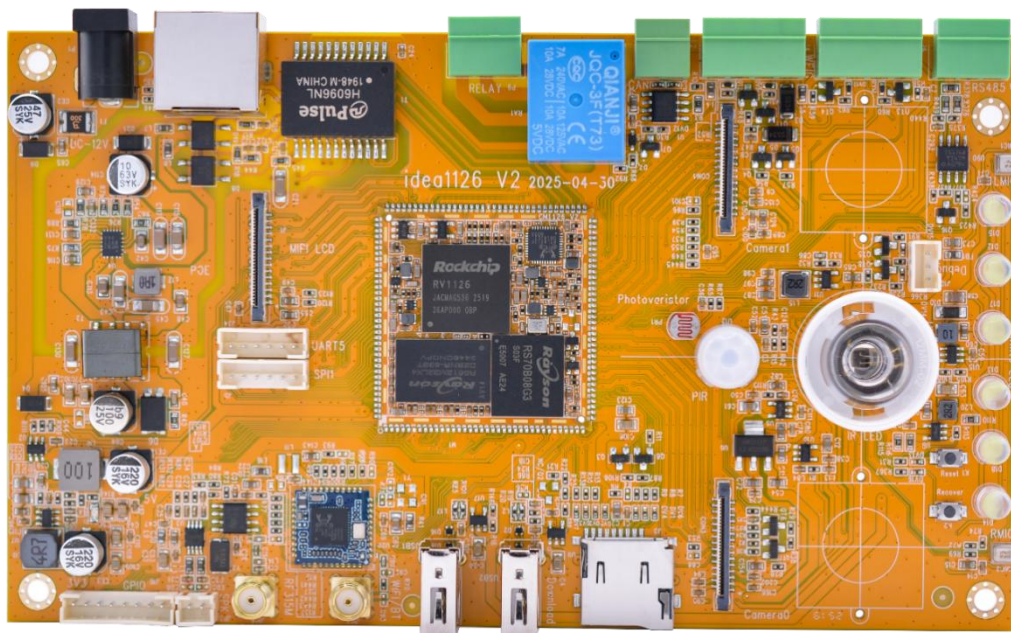
6.1 Serial Terminal	11
8.2 Display	11
8.3 USB	12
6.3.1 USB2.0 Host	12
6.3.2 USB2.0 OTG (Default Device Mode)	14
8.4 Ethernet	16
8.5 SD Card	17
8.6 WiFi & Bluetooth	19
8.6.1 WiFi	20
8.6.2 Bluetooth	23
8.7 Relay	26
8.8 RF315	26
8.9 Audio	28
8.10 PIR	30
8.11 IR	31
8.12 Photovaristor	32
8.13 LED	32
8.14 GPIO	33
8.15 RS485	34
8.16 CAN	35
8.17 UART5	37
8.18 SPI	38
6.19 Wiegand	39
6.20 Camera	40
6.21 Video Playback	42

1.Introduction

1.1 Overview

Idea1126B-P is a high-performance intelligent vision development platform based on the Rockchip RV1126B-P 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, Idea1126B-P supports multiple MIPI camera inputs and 1080P display output. It is widely applicable to smart cameras, AI edge boxes, security surveillance, industrial vision, and more.



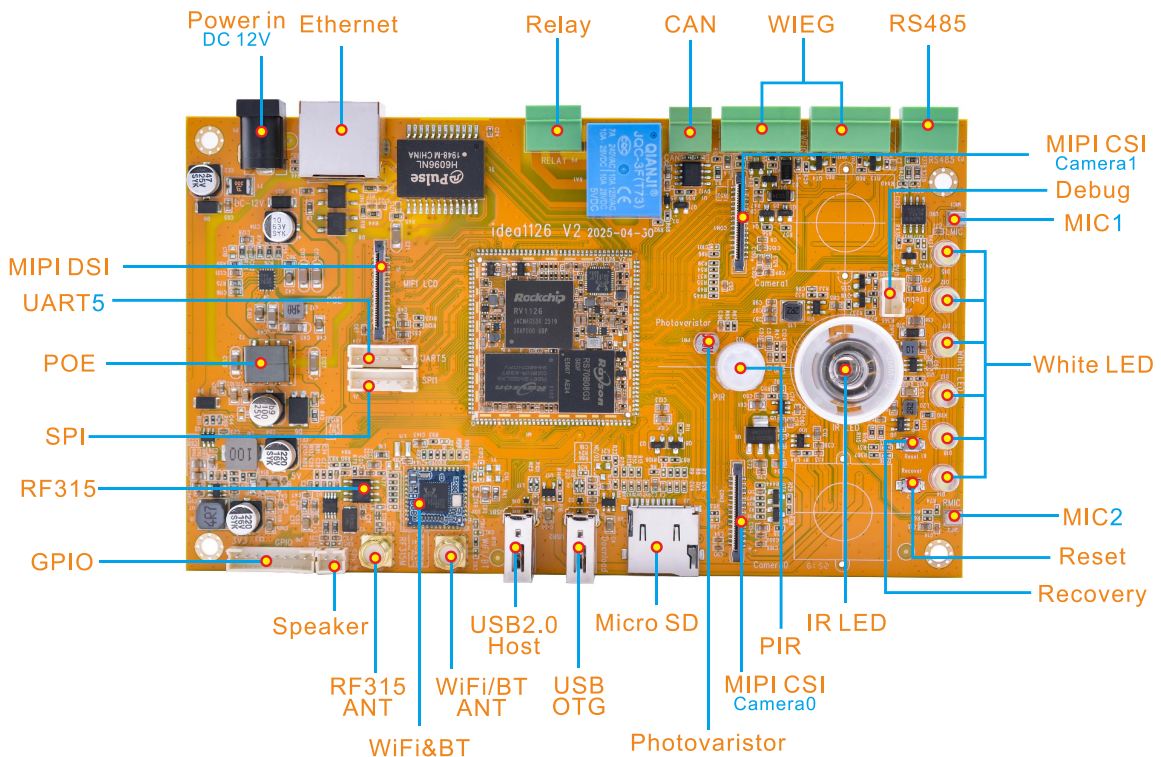
1.2 Product Parameters

Basic Parameters	
SOC	RV1126B-P
CPU	• Quad-core ARM Cortex-A53

		<ul style="list-style-type: none"> • 32KB I-Cache / 32KB D-Cache per core • 512KB shared L2 Cache
NPU		<ul style="list-style-type: none"> • Up to 3 TOPS AI computing power • Supports INT4, INT8, INT16, FP16, BF16, TF32 • Optimized for CNN, Transformer, and other AI models
Video	Decoder	<ul style="list-style-type: none"> • H.265/H.264 up to 3840×2160@30fps • JPEG/MJPEG, VP8, VP9 decoding
	Encoder	<ul style="list-style-type: none"> • H.265/H.264 up to 3840×2160@30fps
AI-ISP		<ul style="list-style-type: none"> • 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		<ul style="list-style-type: none"> • Support 1x MicroSD Card
Display		<ul style="list-style-type: none"> • Support 1x DSI MIPI, up to 1920x1080@60fps
Audio		<ul style="list-style-type: none"> • Support 1x Speaker • Support 2xMIC
USB		<ul style="list-style-type: none"> • Support 2xUSB2.0
Network		<ul style="list-style-type: none"> • Support 1x Gigabit Ethernet • Support 1x WIFI/BT module
Camera		<ul style="list-style-type: none"> • Support 2xCamera
Peripheral communication		<ul style="list-style-type: none"> • Support 1xWiegand • Support 1xSPI • Support 1xRS485 • Support 1xRS232 • Support 1xCAN

Other parameters	Support 1xDebug UART, 1xPIR, 1xIR LED ,6xLED, 1xPhotovaristor, 1xRF315
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	35mm x 34mm
Motherboard dimensions	162mm x 95mm

1.3 Hardware Interface Introduction



Interface parameters	
Power in DC 12V	12V DC power input interface
Ethernet	Gigabit Ethernet RJ45 interface
Relay	Relay output interface for external device control
CAN	CAN communication interface
WIEG	Wiegand interface (input/output supported)
RS485	RS485 communication interface
MIPI CSI Camera1	MIPI CSI camera interface 1
Debug	Debug the serial port
MIC1	Microphone input channel 1
White LED	White LED
MIC2	Microphone input channel 2
Reset	Reset key
Recovery	Recovery key
IR LED	IR LED
PIR	Passive Infrared (PIR) motion sensor
MIPI CSI Camera0	MIPI CSI camera interface 0
Photovaristor	Light sensor
Mirco SD	MicroSD card slot
USB OTG	USB OTG interface (supports device/host switching)
USB2.0 Host	USB 2.0 host interface
WIFI&BT ANT	Wi-Fi/Bluetooth antenna interface
WIFI&BT	WIFI&Bluetooth module
RF315 ANT	RF 315MHz wireless antenna connector
RF315	RF 315MHz wireless module
Speaker	Stereo speaker output
GPIO	General-purpose input/output pins

SPI	SPI interface
POE	Power over Ethernet
UART5	UART5 serial communication interface
MIPI DIS	MIPI display 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

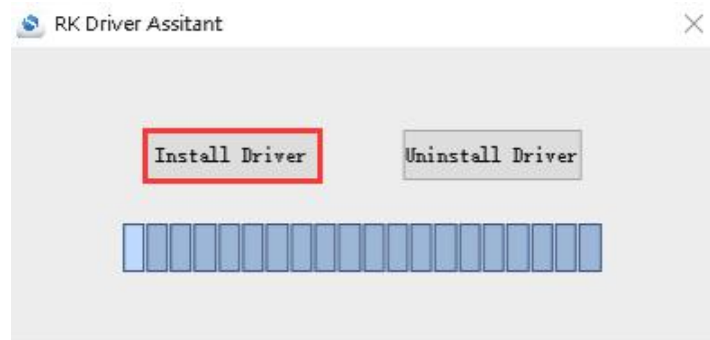
2.1 Install RK Driver Assitant

Step 1: Open [DriverAssitant_v5.1.1/DriverInstall.exe](#).

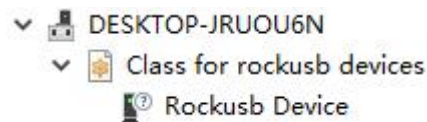
Step 2: To avoid driver conflicts, click “**Uninstall Driver**” to uninstall the driver.



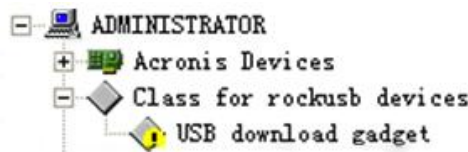
Step 3: Click button “**Install Driver**” to install.



Step 4: After the installation is complete, connect the board and PC with Type_A 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.

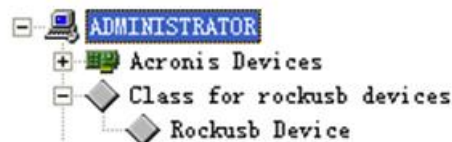


Step 5: If the following device information appears in the **Device Manager** after the operation in Step 4, user need to proceed to the next step.



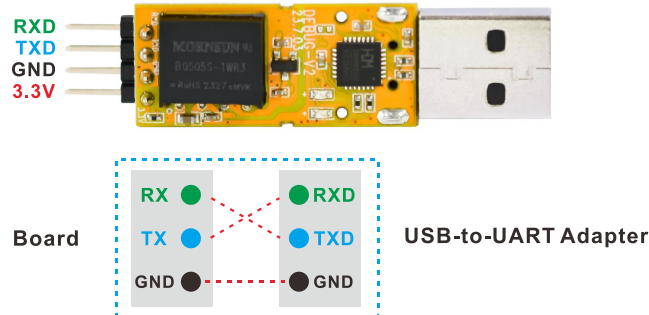
Step 6: The WINDOW will pop up found New Hardware Wizard dialog box, choose to install from the specified location, and then select: *DriverAssitant_y5.1.1/ADBDriver*.

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



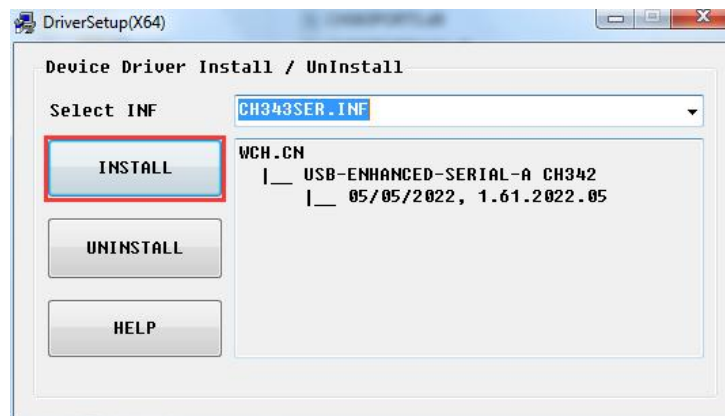
Pin	Connection Description
RXD	Receive, connect to TX pin of the board.
TXD	Transmit, connect to RX pin of the board.
GND	Ground, connect to GND pin of the board.
3V3	No need to connect.

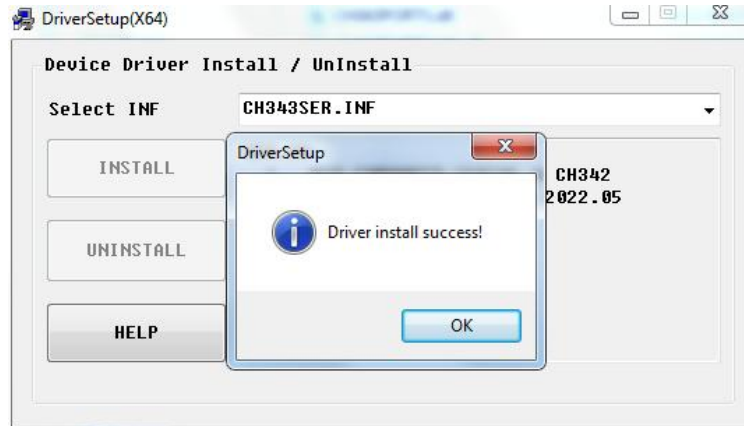
2.2.2 Install Driver

Step 1: Plug the CH9102X Module to the PC

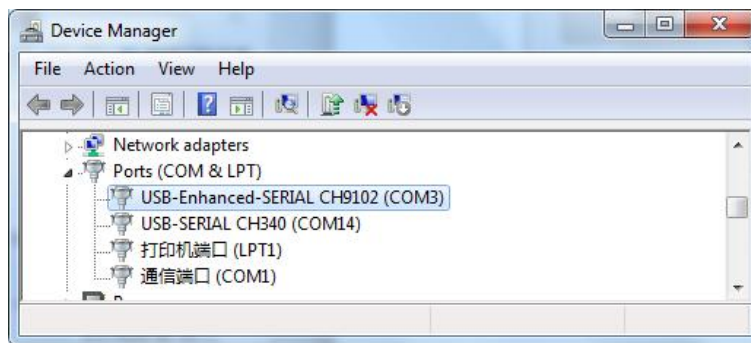
Step 2: Unzip [CH343SER.ZIP](#) on Windows.

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.

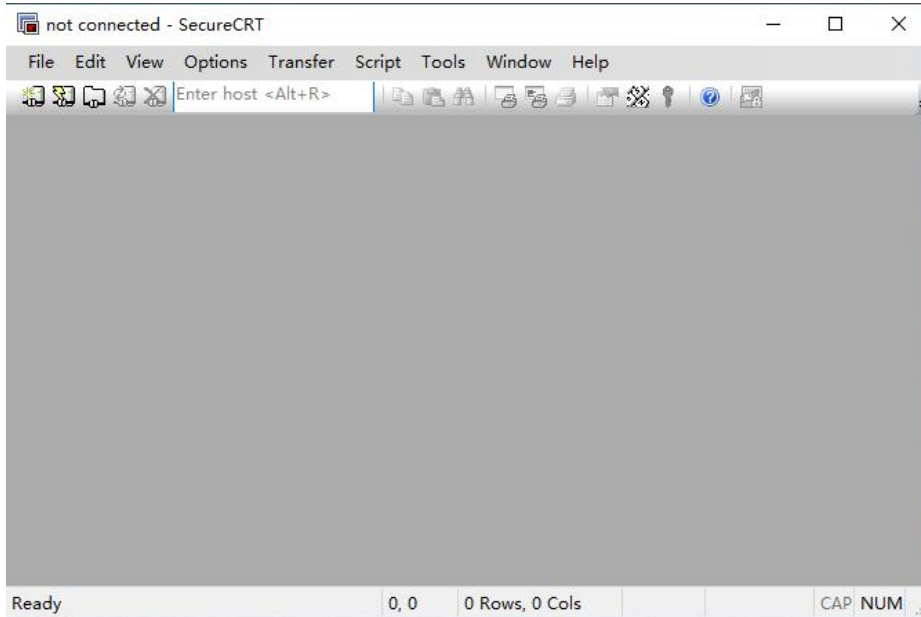


2.3 Install Serial Terminal Tool

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

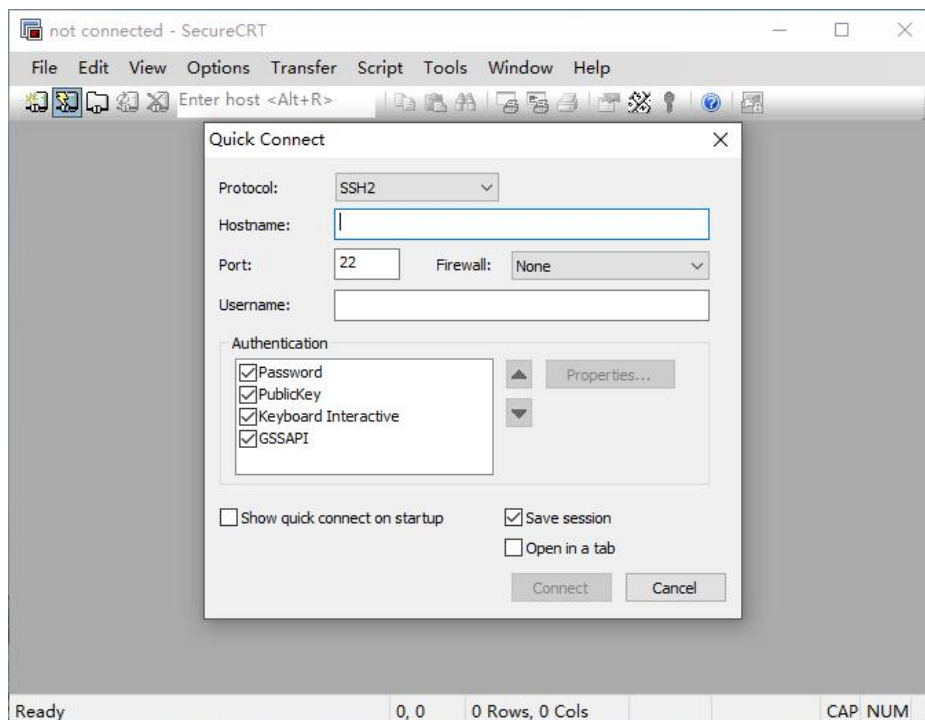
Step 1: Unzip *Platform/SecureCRT.rar* on PC.

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

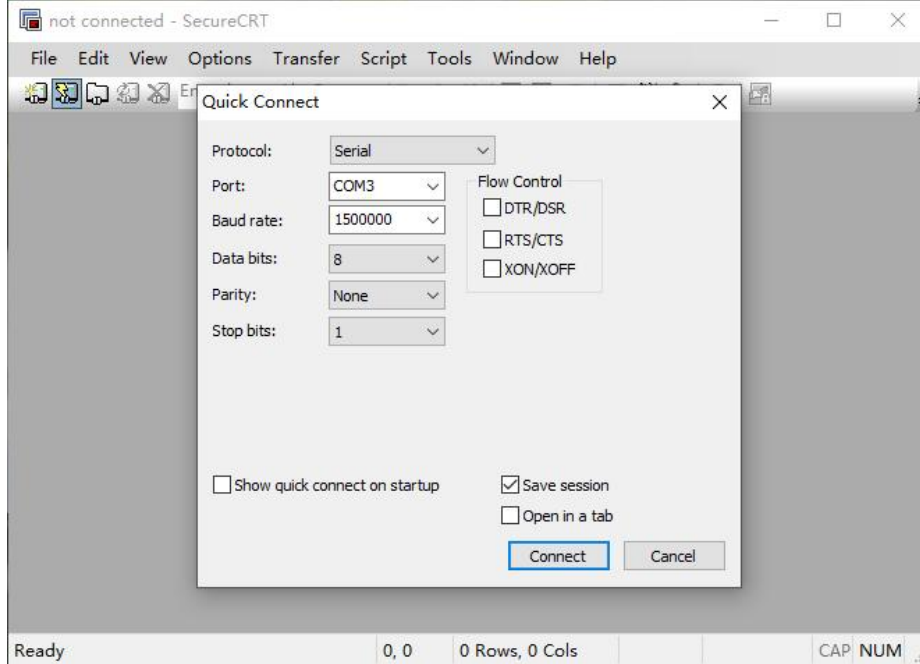


Step 3: Confirm that the CH9102X driver has been installed and the CH9102X module is connecting to the PC.

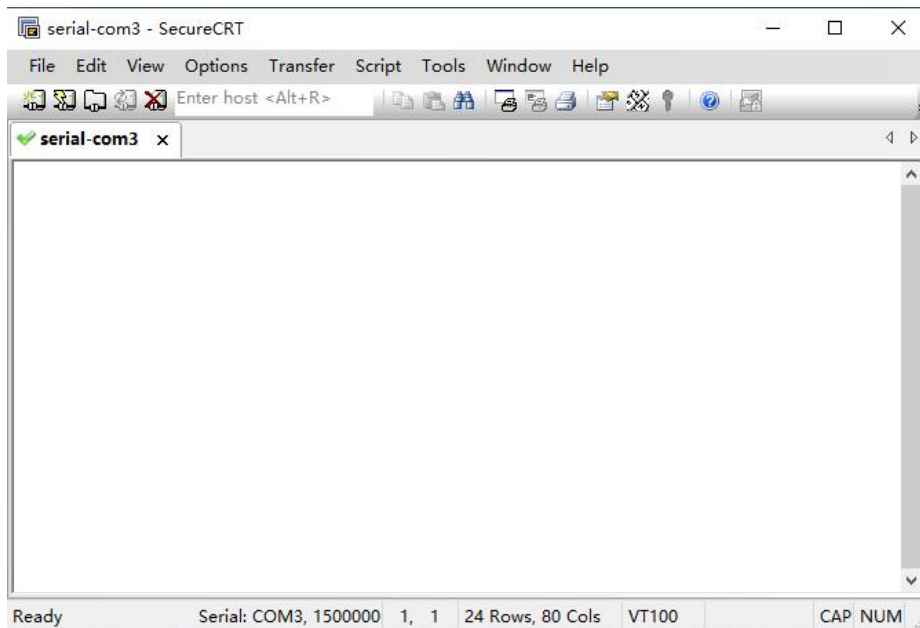
Step 4: Click the “Quick Connect” button to go to the Quick Connect configuration screen.



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

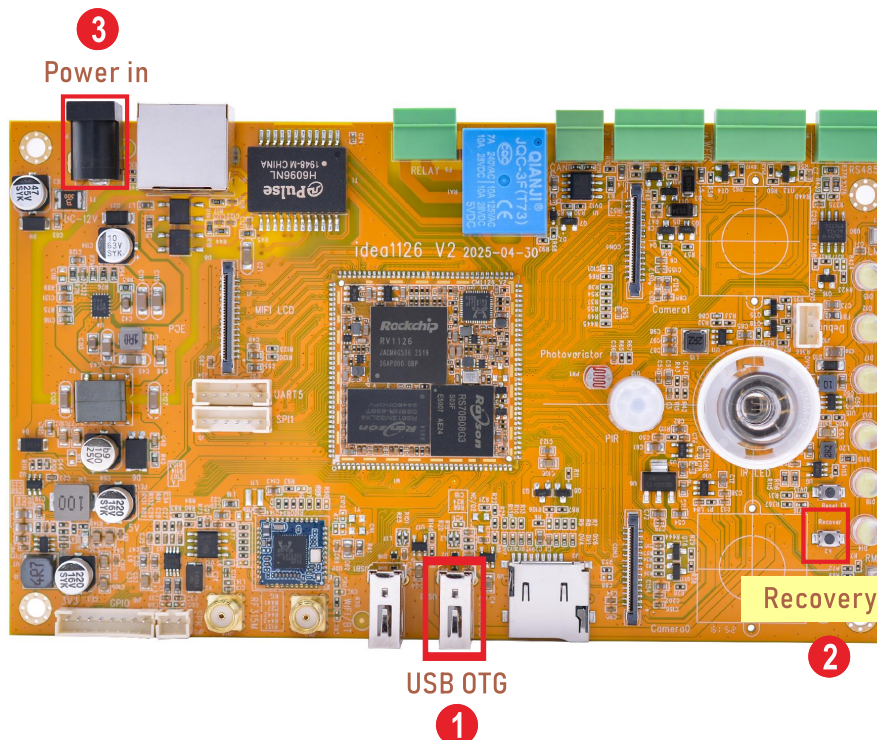
3.1.1.1 Hardware

Step 1: Disconnect the power adapter.

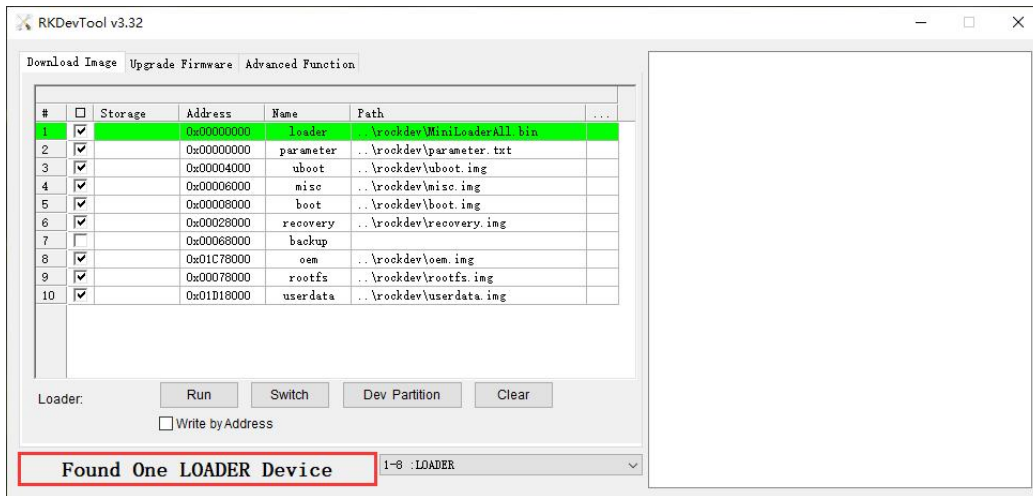
Step 2: Connect one end of the Type_A cable to the host and the other end to the development board.

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 Device”**.



3.1.1.2 Software

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

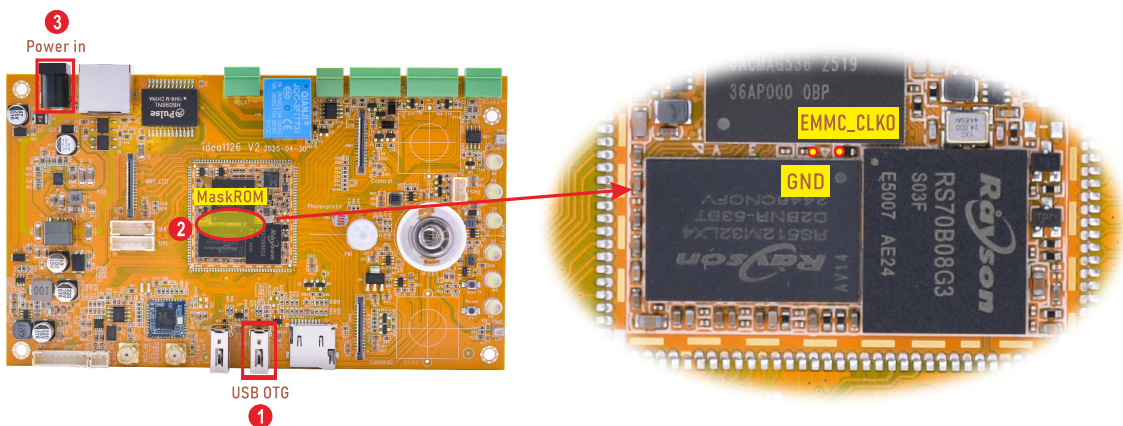
```
# reboot loader
```

3.1.2 How to Enter MaskRom Mode

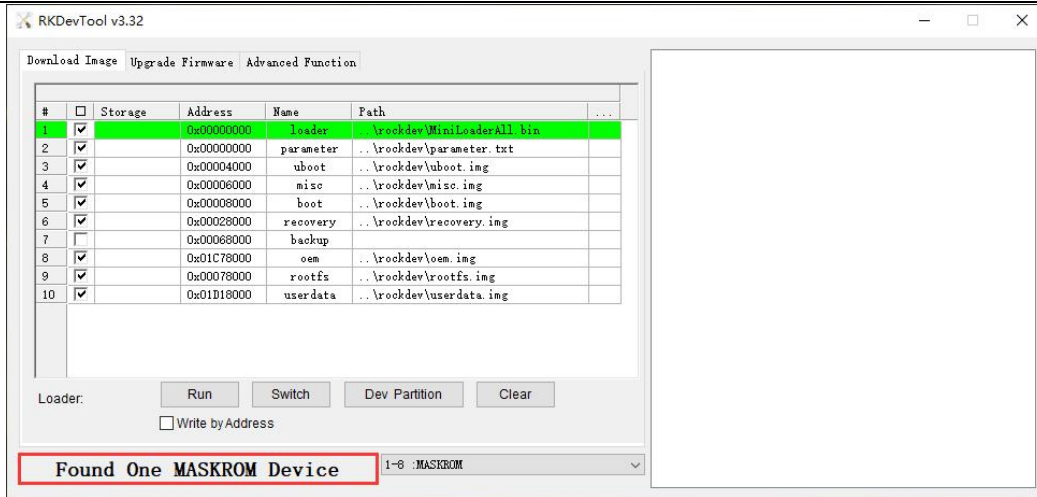
Step 1: Disconnect the power adapter.

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

Step 3: Use tweezers to short the two test points on the CM1126B-P.



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



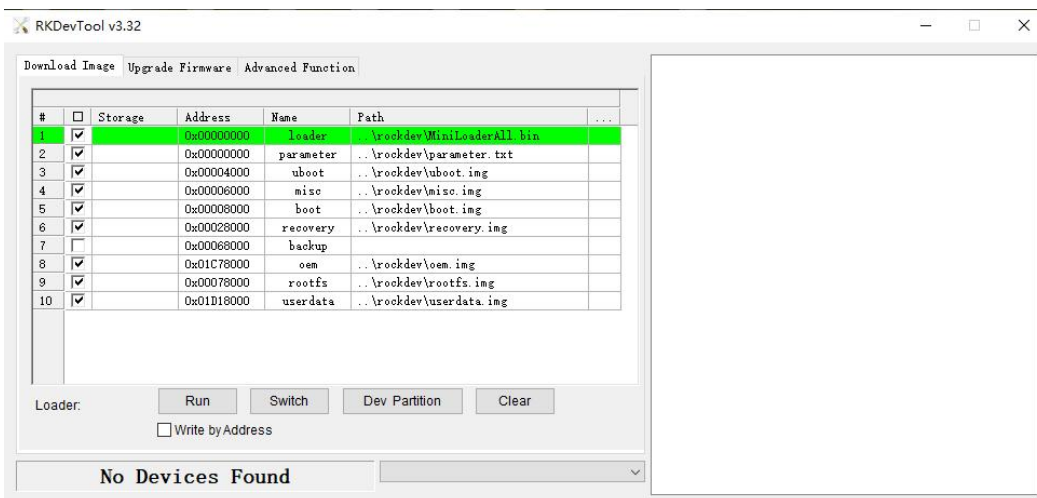
3.2 Burn firmware

Environment: Windows OS (Operating System).

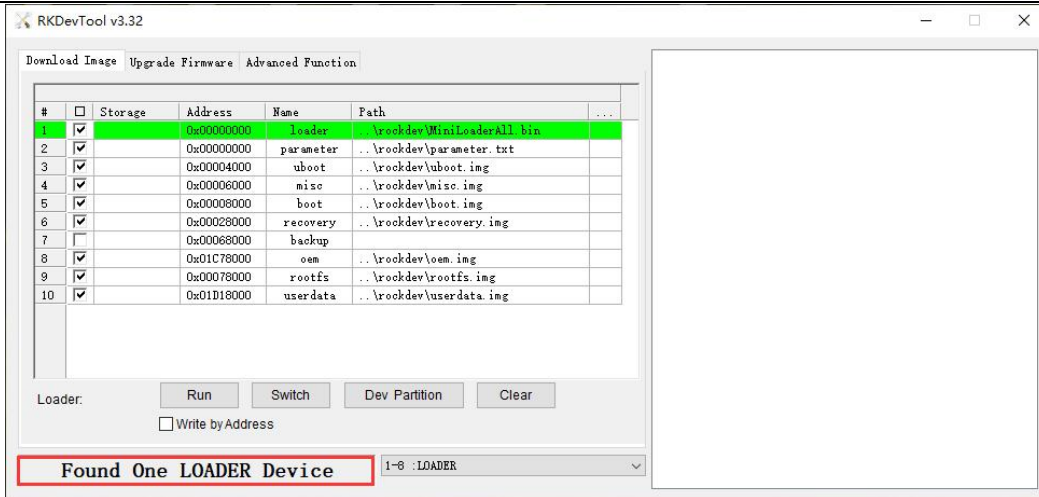
3.2.1 Burn Update.img Firmware

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

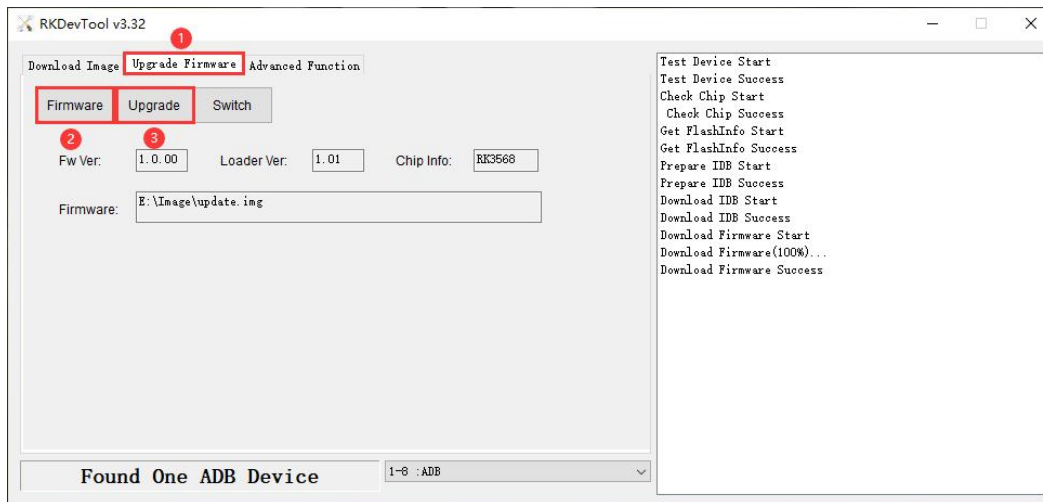
Step 2: Open *RKDevTool\RKDevTool_Release\RKDevTool.exe*.



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.

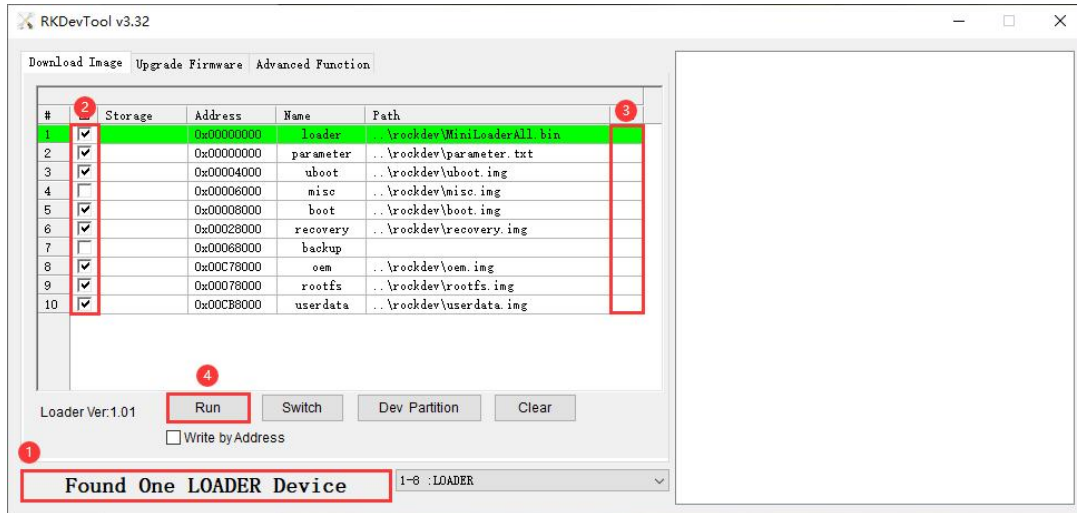
3.2.2 Burn Split Firmware

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

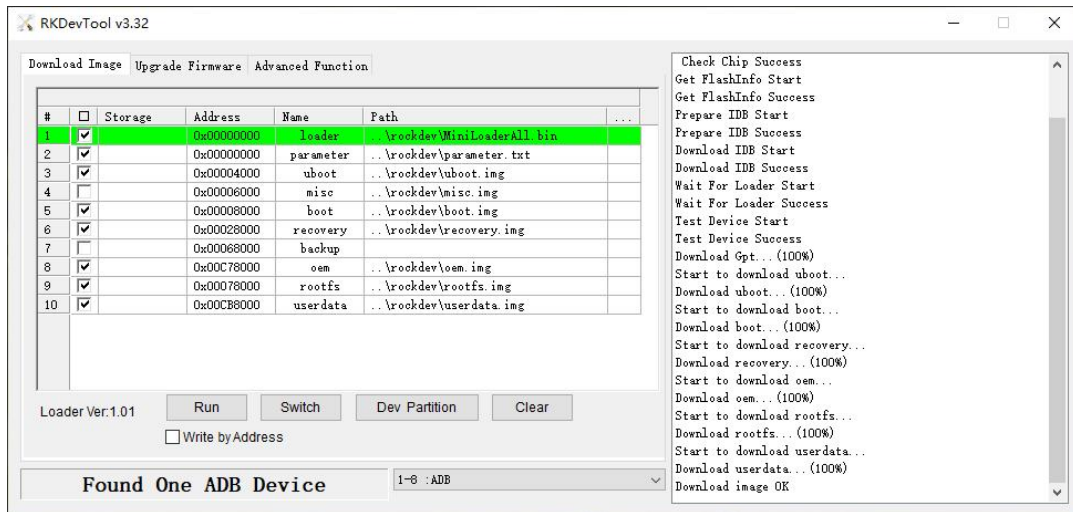
Step 2: Check the partitions to be flashed, multiple partitions can be selected.

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 IDEA1126BP-linux6.1*.tar.gz
$ cd IDEA1126BP-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 IDEA1126BP development board, please input 4or 5 based on your application requirements:

Option 2: General-Purpose Board Configuration (boardcon_idea1126bp_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_idea1126bp_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 4 or 5. These configurations are specifically tuned for the idea1126BP 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

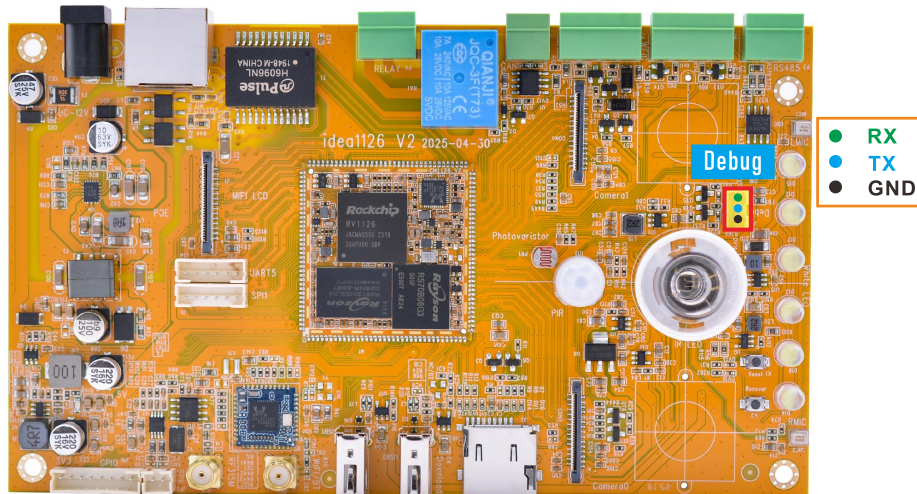
To generate firmware, execute the following command:

```
$ ./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.

```

serial-com8 - SecureCRT
File Edit View Options Transfer Script Tools Window Help
Enter host <Alt+R>
serial-com8 x
[ 9.814378] Bluetooth: hci_uart_register_dev
Realtek Bluetooth :Device setup complete
[ 9.815286] rtk_btcoex: Open BTCOEX
[ 9.815313] rtk_btcoex: rtk_vendor_cmd_to_fw: opcode 0xfc1b
[ 9.818170] rtk_btcoex: BTCOEX hci_rev 0xaa8
[ 9.818200] rtk_btcoex: BTCOEX lmp_subver 0x2df5
HACK: Remove /dev/rfkill to disable external BT power operations.
Successfully init BT for RTL8723DS!
[ 9.845925] Bluetooth: MGMT ver 1.22
Wi-Fi module: RTL8723DS.ko
Installing RTL8723DS.ko ...
[ 9.913077] [WLAN_RFKILL]: rockchip_wifi_get_oob_irq: Enter
Enabling p2p0...
Enabling wlan0...
Successfully init wi-Fi for RTL8723DS!
Done
Successfully initialized wpa_supplicant
rfkill: Cannot open RFKILL control device

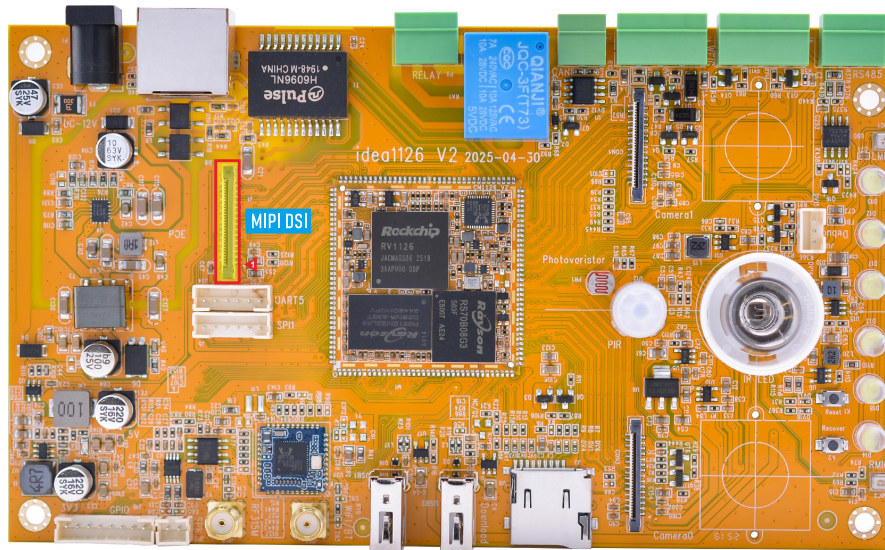
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/# [ 20.318441] platform mtd_vendor_storage: deferred probe pending

root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/#

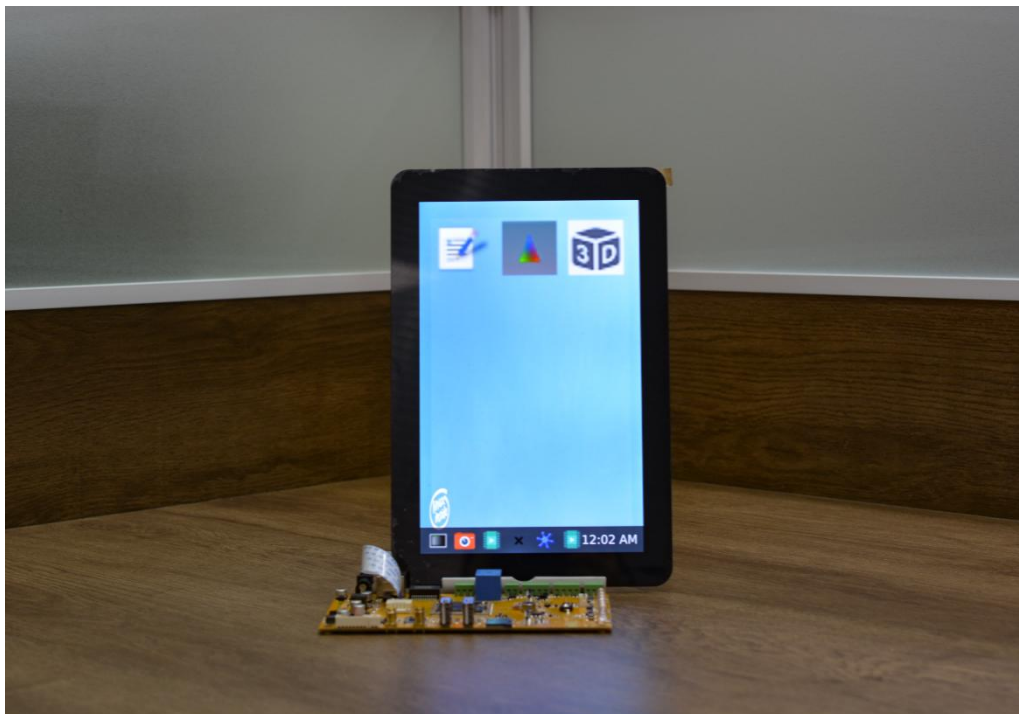
```

6.2 Display

The default MIPI display resolution on Idea1126B-P is 800×1280 @60Hz.



The display effect diagram is as follows:



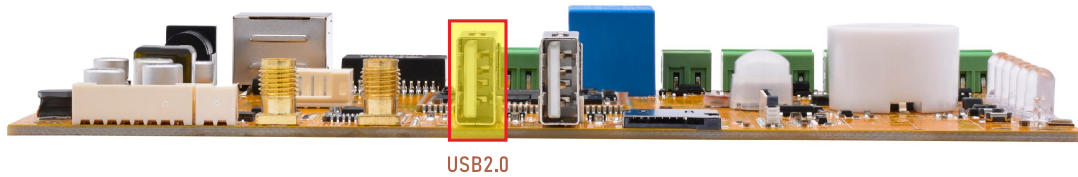
6.3 USB

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
```

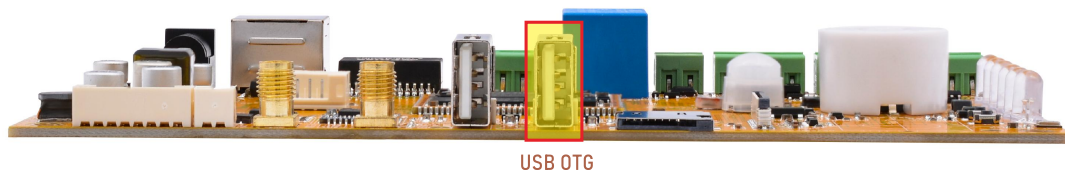
```
root@rv1126b-buildroot:/# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root       6.0G  421M  5.3G   8% /
devtmpfs        959M   0  959M   0% /dev
tmpfs           987M  136K  987M   1% /tmp
tmpfs           987M  500K  987M   1% /run
tmpfs           987M  180K  987M   1% /var/log
tmpfs           987M   0  987M   0% /dev/shm
/dev/mmcblk0p7  123M   12M  109M  10% /oem
/dev/mmcblk0p8  881M  292K  864M   1% /userdata
/dev/sda1       58G   34G  25G  59% /mnt/udisk
```

6.3.2 USB2.0 OTG (Default Device Mode)

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 Type-A cable.

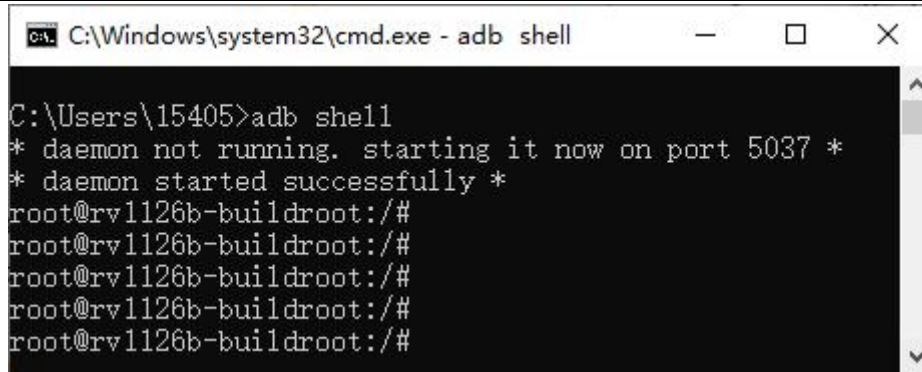


Step 2: Install ADB driver on Windows system.

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
```



```

C:\Windows\system32\cmd.exe - adb shell

C:\Users\15405>adb shell
* daemon not running. starting it now on port 5037 *
* daemon started successfully *
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/#
  
```

Switching OTG modes:

- To switch from Device mode to Host mode:

```
# echo host > /sys/kernel/debug/usb/21500000.usb/mode
```

```

root@rv1126b-buildroot:/# echo host > /sys/kernel/debug/usb/21500000.usb/mode
root@rv1126b-buildroot:/# [ 16.674885] dwc3 21500000.usb: request 00000000af4bc596 was not queued to ep0out
[ 16.675076] android_work: sent uevent USB_STATE=DISCONNECTED
[ 16.781995] phy phy-21400000.usb2-phy.2: illegal mode
[ 16.782031] xhci-hcd xhci-hcd.0.auto: xHCI Host Controller
[ 16.782308] xhci-hcd xhci-hcd.0.auto: new USB bus registered, assigned bus number 1
[ 16.782458] xhci-hcd xhci-hcd.0.auto: hcc params 0x0220fe64 hci version 0x110 quirks 0x0000008022010010
[ 16.782508] xhci-hcd xhci-hcd.0.auto: irq 82, io mem 0x21500000
[ 16.782627] xhci-hcd xhci-hcd.0.auto: xHCI Host Controller
[ 16.782856] xhci-hcd xhci-hcd.0.auto: new USB bus registered, assigned bus number 2
[ 16.782891] xhci-hcd xhci-hcd.0.auto: Host supports USB 3.0 SuperSpeed
[ 16.783066] usb usb1: New USB device found, idVendor=1d6b, idProduct=0002, bcdDevice= 6.01
[ 16.783081] usb usb1: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[ 16.783089] usb usb1: Product: xHCI Host Controller
[ 16.783096] usb usb1: Manufacturer: Linux 6.1.118 xhci-hcd
[ 16.783102] usb usb1: SerialNumber: xhci-hcd.0.auto
[ 16.783599] hub 1-0:1.0: USB hub found
[ 16.783649] hub 1-0:1.0: 1 port detected
[ 16.784045] usb usb2: We don't know the algorithms for LPM for this host, disabling LPM.
[ 16.784156] usb usb2: New USB device found, idVendor=1d6b, idProduct=0003, bcdDevice= 6.01
[ 16.784167] usb usb2: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[ 16.784175] usb usb2: Product: xHCI Host Controller
[ 16.784181] usb usb2: Manufacturer: Linux 6.1.118 xhci-hcd
[ 16.784188] usb usb2: SerialNumber: xhci-hcd.0.auto
[ 16.784605] hub 2-0:1.0: USB hub found
[ 16.784639] hub 2-0:1.0: 1 port detected
[ 19.194206] platform mtd_vendor_storage: deferred probe pending
  
```

- To switch back to Device mode:

```
# echo device > /sys/kernel/debug/usb/21500000.usb/mode
# usbdevice start
```

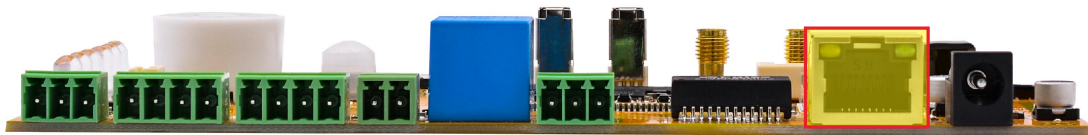
```

root@rv1126b-buildroot:/# echo device > /sys/kernel/debug/usb/21500000.usb/mode
root@rv1126b-buildroot:/# [ 154.564908] xhci-hcd xhci-hcd.0.auto: remove, state 4
[ 154.564960] usb usb2: USB disconnect, device number 1
[ 154.566177] xhci-hcd xhci-hcd.0.auto: USB bus 2 deregistered
[ 154.566578] xhci-hcd xhci-hcd.0.auto: remove, state 4
[ 154.566611] usb usb1: USB disconnect, device number 1
[ 154.568192] xhci-hcd xhci-hcd.0.auto: USB bus 1 deregistered

root@rv1126b-buildroot:/# usbdevice start
[1970-01-01 00:02:42] usbdevice: Starting /usr/bin/usbdevice start, log saved to
/var/log//usbdevice.log
root@rv1126b-buildroot:/# [ 165.563203] dwc3 21500000.usb: device reset
[ 165.651117] android_work: sent uevent USB_STATE=CONNECTED
[ 165.687945] android_work: sent uevent USB_STATE=CONFIGURED
  
```

6.4 Ethernet

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



Ethernet

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

```

root@rv1126b-buildroot:/# [ 1176.092585] rk_gmac-dwmac 21c70000.ethernet eth0: Link is Up - 1Gbps/Full - flow
control rx/tx
[ 1176.092614] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
  
```

Step 2: View network interface information.

```
# ifconfig
```

```

root@rv1126b-buildroot:/# ifconfig
eth0    Link encap:Ethernet  HWaddr E6:4E:29:F1:89:35
inet addr:192.168.0.139  Bcast:192.168.0.255  Mask:255.255.255.0
inet6 addr: fe80::e605:5eb5:7ecc:e39b/64  Scope:Link
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:62 errors:0 dropped:23 overruns:0 frame:0
TX packets:11 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:12052 (11.7 KiB)  TX bytes:1410 (1.3 KiB)
Interrupt:80
  
```

Step 3: Network connection test.

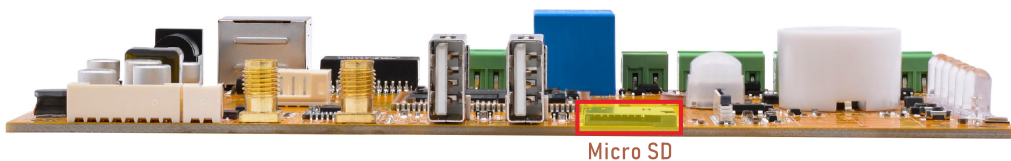
```
# ping -I eth0 www.armdesigner.com
```

```

root@rv1126b-buildroot:/# ping -I eth0 www.armdesigner.com
PING www.armdesigner.com (67.222.54.196) from 192.168.0.139 eth0: 56(84) bytes of data.
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=1 ttl=50 time=187 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=2 ttl=50 time=187 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=3 ttl=50 time=187 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=4 ttl=50 time=187 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=5 ttl=50 time=187 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=6 ttl=50 time=187 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=7 ttl=50 time=184 ms
^C
--- www.armdesigner.com ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6010ms
rtt min/avg/max/mdev = 183.917/186.384/187.376/1.038 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@em3588-buildroot:/# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        14G  701M   13G   6% /
devtmpfs        3.9G   8.0K  3.9G   1% /dev
tmpfs           3.9G  140K  3.9G   1% /tmp
tmpfs           3.9G  568K  3.9G   1% /run
tmpfs           3.9G  252K  3.9G   1% /var/log
tmpfs           3.9G   0  3.9G   0% /dev/shm
/dev/mmcblk0p8  15G  332K   15G   1% /userdata
/dev/mmcblk0p7  123M  12M  110M  10% /oem
/dev/sda1       120G  1.6M  120G   1% /mnt/storage
  
```

6.6 Relay

Default power-on state: The relay is in the energized (closed) state, and the normally open (ON) contact is connected to the common (COM) terminal.



- To release the relay and disconnect the (ON) and (COM) connection, execute the following

command:

```
# echo 1 > /sys/class/leds/raley_power/brightness
```

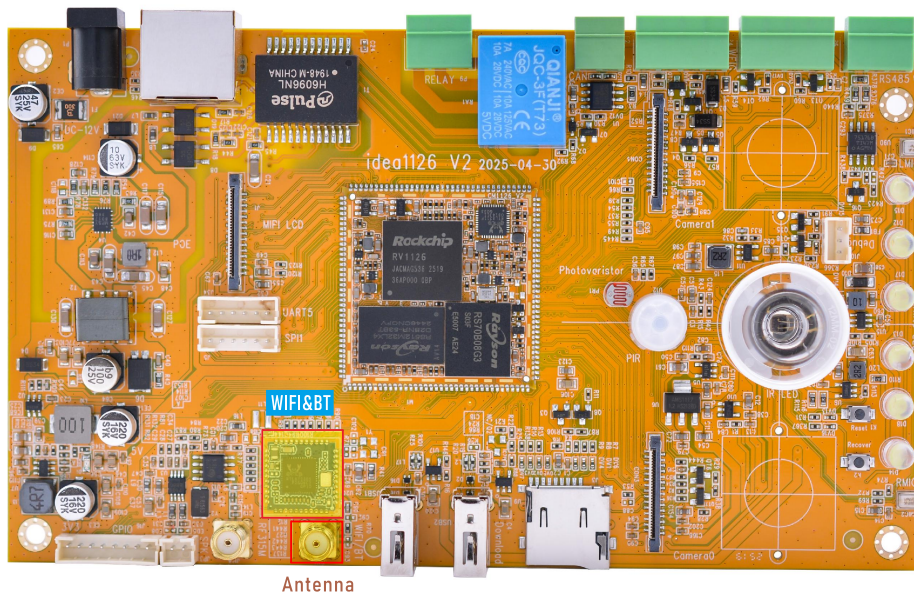
After executing the command: The relay is released, and the **normally closed (NC)** contact is connected to the **common (COM)** terminal.

- To restore the **(ON)** and **(COM)** connection (relay energized), execute the following command:

```
# echo 0 > /sys/class/leds/raley_power/brightness
```

6.7 WiFi & BLUETOOTH

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



Antenna

Step 1: View the device information.

```
# ifconfig
```

```

root@rv1126b-buildroot:/# ifconfig
p2p0    Link encap:Ethernet HWaddr AA:B5:8E:CA:27:4E
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:CA:27:4E
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: B2:22:7A:5A:B6:4A
          ESSID:"DIRECT-4A-HP Laser 136w"
          Protocol:IEEE 802.11gn
          Mode:Master
          Frequency:2.412 GHz (Channel 1)
          Encryption key:on
          Bit Rates:72 Mb/s
          Extra:rsn_ie=30140100000fac040100000fac040100000fac020c00
          IE: IEEE 802.11i/WPA2 Version 1
              Group Cipher : CCMP
              Pairwise Ciphers (1) : CCMP
              Authentication Suites (1) : PSK
          Quality=84/100 Signal level=49/100
          Extra:fm=0003
Cell 02 - Address: B4:F1:8C:6D:D1:24
          ESSID:"Boardcon"
          Protocol:IEEE 802.11bgn
          Mode:Master
          Frequency:2.412 GHz (Channel 1)
          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:
          DD910050F204104A0001101044000102103B00010310470010123456789ABCDEF01234B4F18C6DD124102100194875617765692054
          6563686E6F6C6F677920436F2E2C4C74641023000B576972656C657373204150102400033132331042000531323334351054000800
          060050F2040001101100095443373130322D3130100800020780103C0001011049000600372A000120
          Quality=96/100 Signal level=64/100
  
```

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
root@rv1126b-buildroot:/# [ 175.625243] NOHZ tick-stop error: local softirq work is pending, handler #08!!!
[ 175.625971] 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:CA:27:4E
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:CA:27:4E
inet addr:192.168.0.232 Bcast:192.168.0.255 Mask:255.255.255.0
inet6 addr: fe80::45f9:da11:b0a4:e5c9/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:155 errors:0 dropped:33 overruns:0 frame:0
TX packets:18 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:18958 (18.5 KiB) TX bytes:2584 (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 192.168.0.232 wlan0: 56(84) bytes of data.
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=1 ttl=50 time=186 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=2 ttl=50 time=237 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=3 ttl=50 time=202 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=4 ttl=50 time=1038 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=5 ttl=50 time=201 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=6 ttl=50 time=227 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=7 ttl=50 time=194 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=8 ttl=50 time=205 ms
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=9 ttl=50 time=196 ms
^C
--- www.armdesigner.com ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8030ms
rtt min/avg/max/mdev = 185.953/298.431/1037.928/261.896 ms, pipe 2
```

Step 1: Start the BlueALSA service in A2DP Sink mode.

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

```
root@rv1126b-buildroot:/# bluealsa -p a2dp-sink &
root@rv1126b-buildroot:/# bluealsa -p a2dp-sink &
[1] 2009
root@rv1126b-buildroot:/# bluealsa: [2009] D: storage.c:84: Initializing persistent storage: /var/lib/bluealsa
bluealsa: [2009] D: main.c:604: Starting main dispatching loop
bluealsa: [2009] D: main.c:114: Acquired D-Bus service name: org.bluealsa
bluealsa: [2009] D: bluealsa-dbus.c:392: Registering D-Bus manager: /org/bluealsa
bluealsa: [2009] D: bluez.c:783: Registering battery provider: /org/bluez/hci0/battery
bluealsa: [2009] D: bluez.c:598: Creating media endpoint object: /org/bluez/hci0/A2DP/SBC/sink/1
bluealsa: [2009] D: bluez.c:509: Registering media endpoint: /org/bluez/hci0/A2DP/SBC/sink/1
bluealsa: [2009] D: bluez.c:598: Creating media endpoint object: /org/bluez/hci0/A2DP/SBC/sink/2
bluealsa: [2009] D: bluez.c:509: Registering media endpoint: /org/bluez/hci0/A2DP/SBC/sink/2
```

Step 2: Pair and connect to your phone's Bluetooth using the bluetoothctl command.

```
# bluetoothctl
```

```
[bluetooth]# scan on
```

```
[bluetooth]# scan on
SetDiscoveryFilter success
Failed to start discovery: org.bluez.Error.InProgress
[bluetooth]# pair 2[ 1499.068630] rtk_btcoex: inquiry complete
hci0 type 7 discovering off
[bluetooth]# [ 1504.825822] rtk_btcoex: hci (periodic)inq start
hci0 type 7 discovering on
[NEW] Device 6C:AD:01:6D:F6:BA 6C-AD-01-6D-F6-BA
[CHG] Device 55:83:D0:0A:6E:72 RSSI: 0xffffffffaa (-86)
[NEW] Device 49:FF:C6:1C:99:10 49-FF-C6-1C-99-10
[DEL] Device 6C:00:80:8C:4F:E3 6C-00-80-8C-4F-E3
[NEW] Device 6C:00:80:8C:4F:E3 6C-00-80-8C-4F-E3
[DEL] Device 5E:BE:25:51:1A:5A 5E-BE-25-51-1A-5A
[DEL] Device 57:66:57:2E:82:3F 57-66-57-2E-82-3F
[NEW] Device 2C:A0:42:D1:14:D5 Kang
[NEW] Device 65:A0:3D:CC:85:91 65-A0-3D-CC-85-91
[bluetooth]# pair[ 1515.069221] rtk_btcoex: inquiry complete
hci0 type 7 discovering off

/org/bluez/hci0/A2DP/SBC/sink/2
bluealsa: [2009] D: bluez.c:509: Registering media endpoint: /org/bluez/hci0/A2DP/SBC/sink/2
```

```
[bluetooth]# pair 2C:A0:42:D1:14:D5
```

```
[bluetooth]# pair 2C:A0:42:D1:14:D5
Attempting to pair with 2C:A0:42:D1:14:D5
hci0 device_flags_changed: 2C:A0:42:D1:14:D5 (BR/EDR)
  supp: 0x00000000 curr: 0x00000000
[bluetooth]# [ 1516.950710] rtk_btcoex: hci create connection, start paging
[ 1519.193500] rtk_btcoex: connected, handle 0005, status 0x00
[ 1519.193536] rtk_btcoex: Page success
hci0 2C:A0:42:D1:14:D5 type BR/EDR connected eir_len 5
[CHG] Device 2C:A0:42:D1:14:D5 Connected: yes
hci0 2C:A0:42:D1:14:D5 type BR/EDR connected eir_len 11
[Kang]# [ 1519.262535] rtk_btcoex: io capability request
hci0 2C:A0:42:D1:14:D5 type BR/EDR connected eir_len 5
Request confirmation
[agent] Confirm passkey 144549 (yes/no): yes
```

```
[Kang]# connect 2C:A0:42:D1:14:D5
```

```
[Kang]# connect 2C:A0:42:D1:14:D5
Attempting to connect to 2C:A0:42:D1:14:D5
```

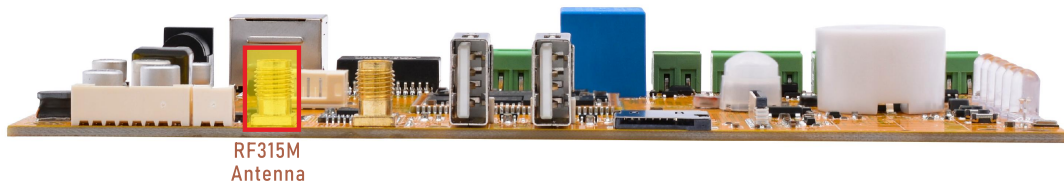
Exit bluetoothctl, Play Bluetooth audio using bluealsa-play

```
# bluealsa-play 2C:A0:42:D1:14:D5 &
```

```
root@rv1126b-buildroot:/# bluealsa-aplay 2C:A0:42:D1:14:D5 &  
[2] 2039  
root@rv1126b-buildroot:/# bluealsa-aplay: [2039] D: aplay.c:846: Creating IO worker 2C:A0:42:D1:14:D5
```

6.8 RF315

Step 1: Connect antenna.



Step 2: Open RF315 debug print:

```
# echo 1 > /sys/module/rf315_decode/parameters/code_print
```

Enable debugging output to monitor received RF signals (via UART, GPIO, or /dev device as applicable).

When pressing a button near the RF receiver, the received code or key value will be printed to the system log or serial console.

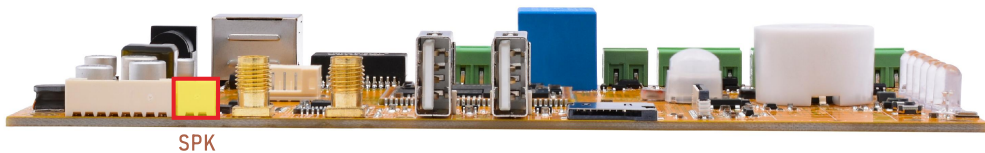


```

toot@rv1126b-buildroot:/# echo 1 > /sys/module/rf315_decode/parameters/code_print
root@rv1126b-buildroot:/# [ 404.385220] RF315 data=0xadf481
[ 404.422044] RF315 data=0xadf481
[ 404.458973] RF315 data=0xadf481
[ 406.833506] RF315 data=0xadf482
[ 406.870369] RF315 data=0xadf482
[ 406.907226] RF315 data=0xadf482
[ 408.541557] RF315 data=0xadf488
[ 408.578451] RF315 data=0xadf488
[ 409.685817] RF315 data=0xadf484
[ 409.722696] RF315 data=0xadf484
[ 409.759572] RF315 data=0xadf484
[ 412.356096] RF315 data=0xadf488
[ 412.392938] RF315 data=0xadf488
    
```

6.9 Audio

Step 1: Connect the speaker.



Step 2: View sound card.

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

```
root@rv1126b-buildroot:/# cat /proc/asound/cards
0 [rockchiprv1126b]: rockchip_rv1126 - rockchip,rv1126b-acodec
  rockchip,rv1126b-acodec
1 [rockchippdmica]: simple-card - rockchip,pdm-mic-array
  rockchip,pdm-mic-array
```

Step 3: speaker play audio.

```
# aplay -Dhw:0,0 /mnt/udisk/video/music/2.wav
```

```
root@rv1126b-buildroot:/# aplay -Dhw:0,0 /mnt/udisk/video/music/2.wav
Playing WAVE '/mnt/udisk/video/music/2.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

Recording

Step 1: View sound card.

```
# arecord -l
```

```
root@rv1126b-buildroot:/# arecord -l
**** List of CAPTURE Hardware Devices ****
card 1: rockchippdmica [rockchip,pdm-mic-array], device 0: 20990000.pdm-dummy_codec dummy_codec-0 [20990000.pdm-dummy_codec dummy_c
odec-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
root@rv1126b-buildroot:/# _
```

Step 2: Use command recording.

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

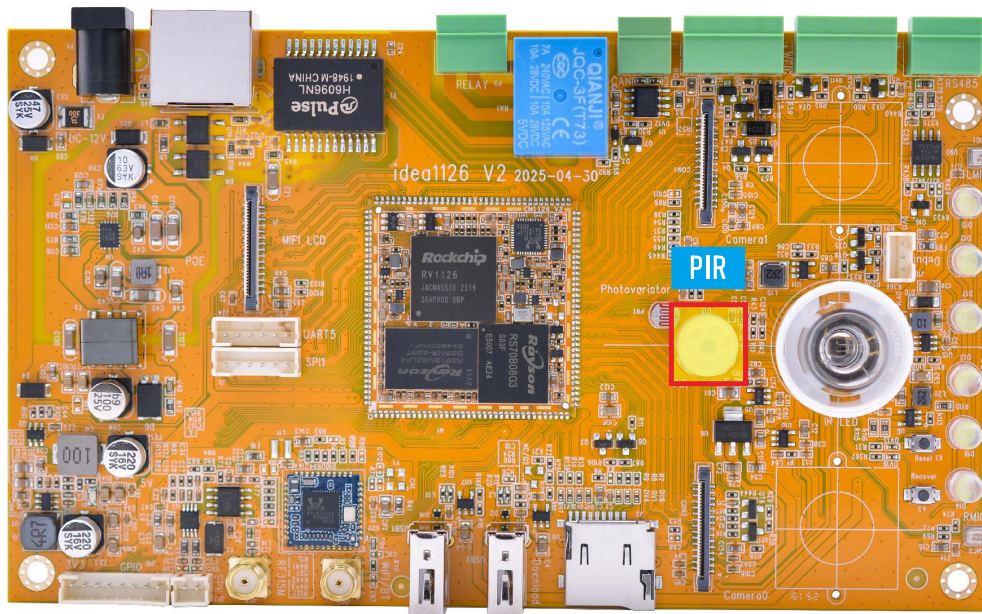
```
root@rv1126b-buildroot:/# arecord -D hw:1,0 -f s16_le -c 2 -r 16000 cap.wav
Recording WAVE 'cap.wav' : Signed 16 bit Little Endian, Rate 16000 Hz, Stereo
```

Step 3: Check whether the audio file has been generated

```
# ls cap.wav
```

```
root@rv1126b-buildroot:/# ls cap.wav
cap.wav
root@rv1126b-buildroot:/#
```

6.10 PIR



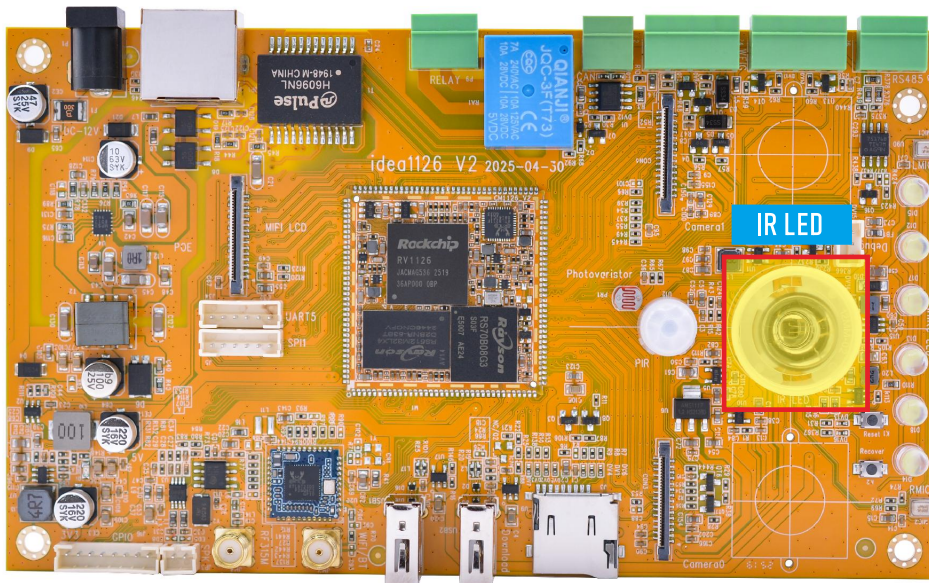
Execute the following command to test:

```
# pir_test.sh
```

When the sensor detects human movement, the output is **1**; when no human is detected, the output is **0**.

```
root@rv1126b-buildroot:/# pir_test.sh
0
1
1
1
0
1
1
1
1
0
1
1
1
1
1
1
0
```

6.11 IR



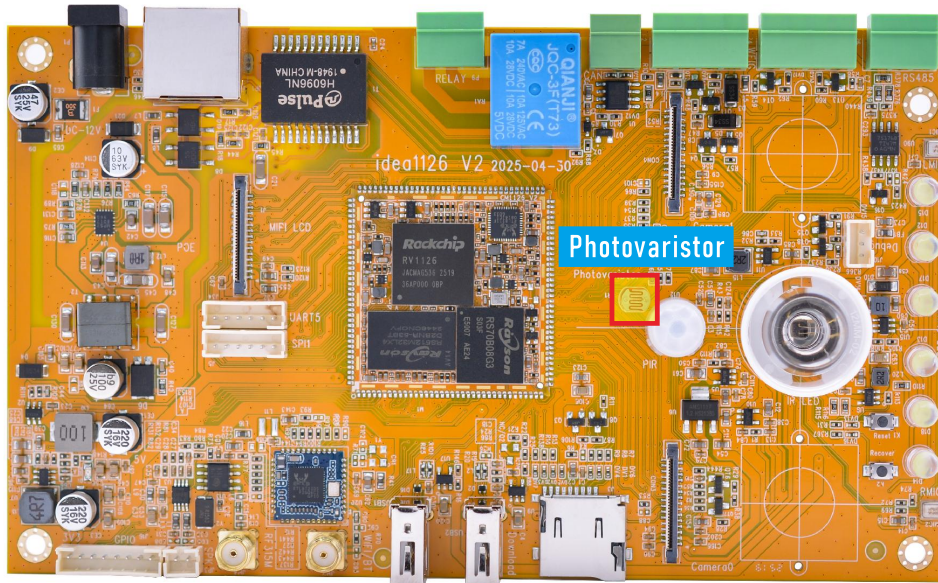
Step 1: Execute the following command to turn on the IR LED.

```
# echo 1 > /sys/class/leds/ir_led/brightness
```

Step 2: Execute the following command to turn off the IR LED.

```
# echo 0 > /sys/class/leds/ir_led/brightness
```

6.12 Photovaristor



Execute the following command to read the raw value of the photoresistor:

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

```
root@rv1126b-buildroot:/# cat /sys/bus/iio/devices/iio\:device0/in_voltage1_raw
1827
root@rv1126b-buildroot:/#
root@rv1126b-buildroot:/# cat /sys/bus/iio/devices/iio\:device0/in_voltage1_raw
6209
root@rv1126b-buildroot:/# cat /sys/bus/iio/devices/iio\:device0/in_voltage1_raw
18340
```

6.13 LED



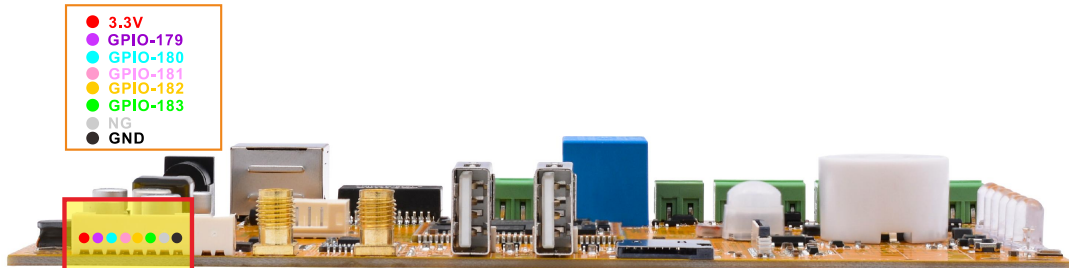
Step 1: Execute the following command to turn on the LED.

```
# echo 1 > /sys/class/leds/user_led/brightness
```

Step 2: Execute the following command to turn off the LED.

```
# echo 0 > /sys/class/leds/user_led/brightness
```

6.14 GPIO



General GPIO Control Command Guide:

```
# Export the GPIO (replace X with the actual GPIO number)
echo X > /sys/class/gpio/export

# Set the direction to output
echo out > /sys/class/gpio/gpioX/direction

# Set output to high level
echo 1 > /sys/class/gpio/gpioX/value

# Read the current GPIO level (should be 1)
cat /sys/class/gpio/gpioX/value

# Set output to low level
echo 0 > /sys/class/gpio/gpioX/value

# Read the GPIO level again (should be 0)
cat /sys/class/gpio/gpioX/value
```

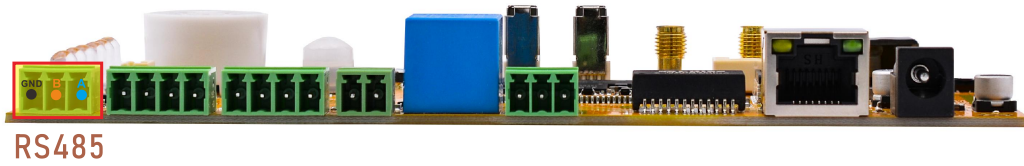
For example, using

GPIO-179:

```
# echo 179 > /sys/class/gpio/export
# echo out > /sys/class/gpio/gpio179/direction
# echo 1 > /sys/class/gpio/gpio179/value
# cat /sys/class/gpio/gpio179/value
# echo 0 > /sys/class/gpio/gpio179/value
# cat /sys/class/gpio/gpio179/value
```

```
root@rv1126b-buildroot:/# echo 179 > /sys/class/gpio/export
root@rv1126b-buildroot:/# echo out > /sys/class/gpio/gpio179/direction
root@rv1126b-buildroot:/# echo 1 > /sys/class/gpio/gpio179/value
root@rv1126b-buildroot:/# cat /sys/class/gpio/gpio179/value
1
root@rv1126b-buildroot:/# echo 0 > /sys/class/gpio/gpio179/value
root@rv1126b-buildroot:/# cat /sys/class/gpio/gpio179/value
0
```

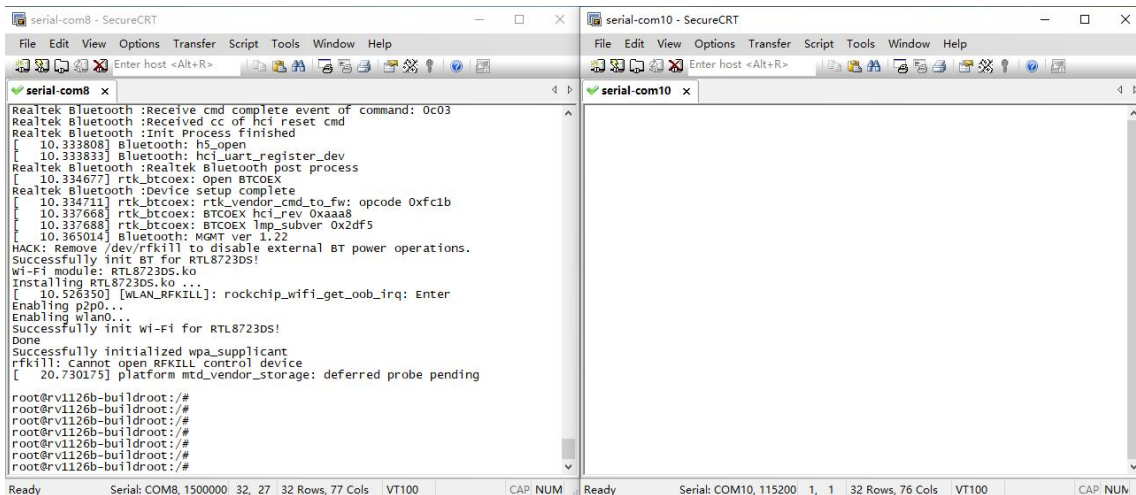
6.15 RS485



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

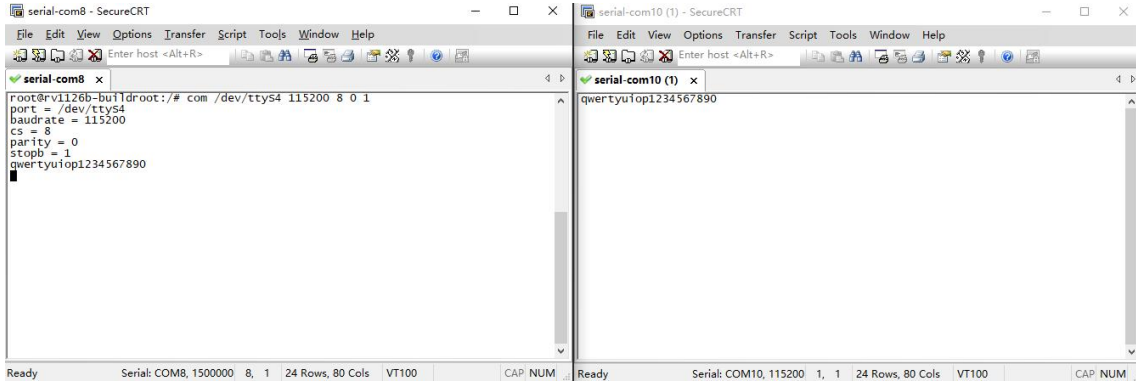


Step 2: Open the corresponding serial terminal, set the baud rate of the board to 1500000, and set the baud rate of the RS485 test tool to 115200.



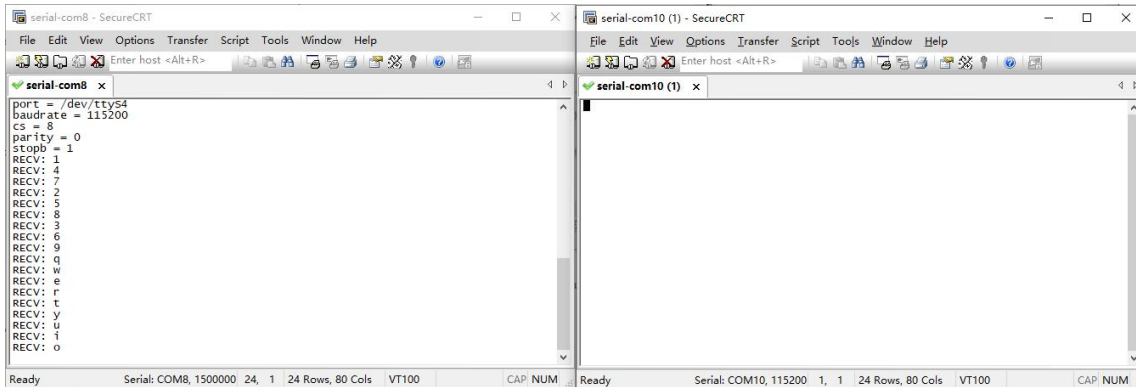
Step 3: Send data for testing

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

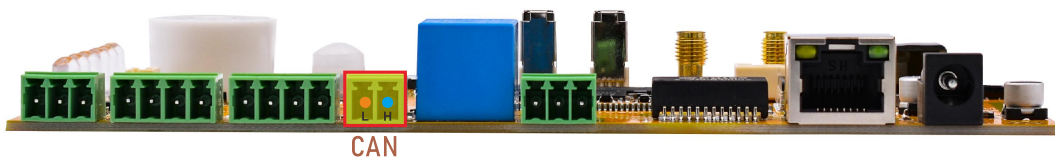


Step 4: Receiving data test.

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



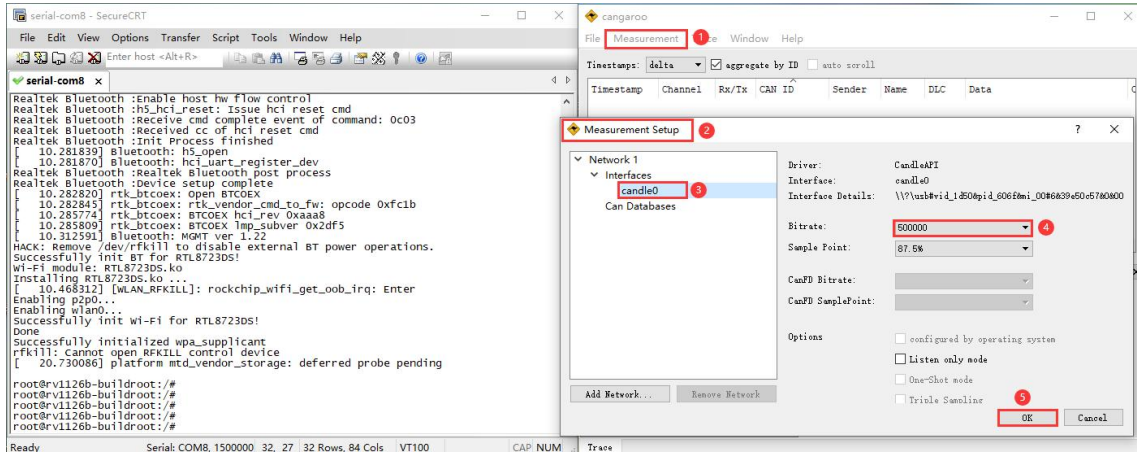
6.16 CAN



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.

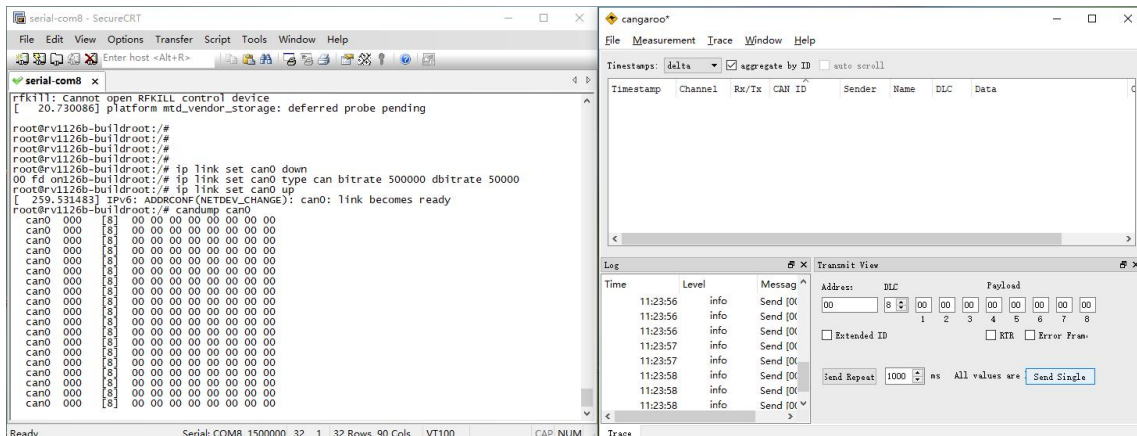


Step 3: Configure and activate the CAN network, setting the bitrate to 500000.

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

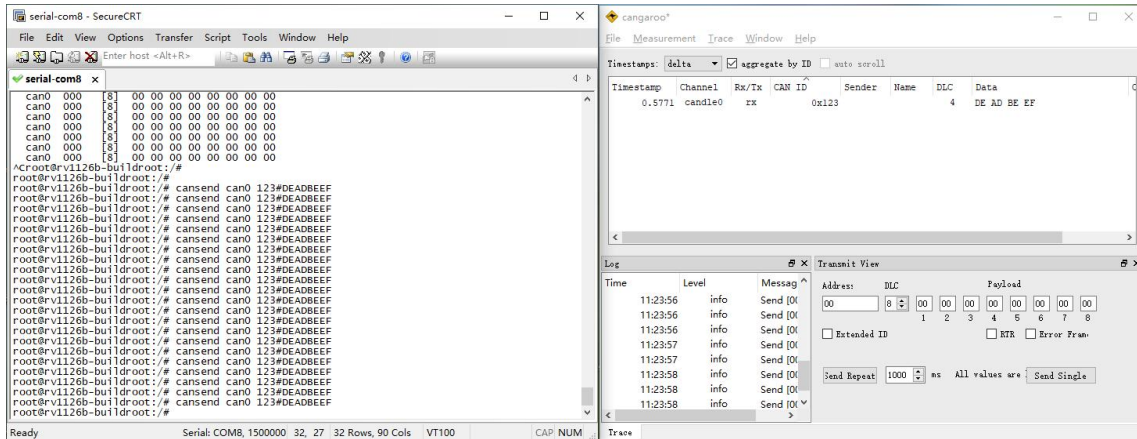
Step 4: Configure CAN as the receiver.

```
# candump can0
```



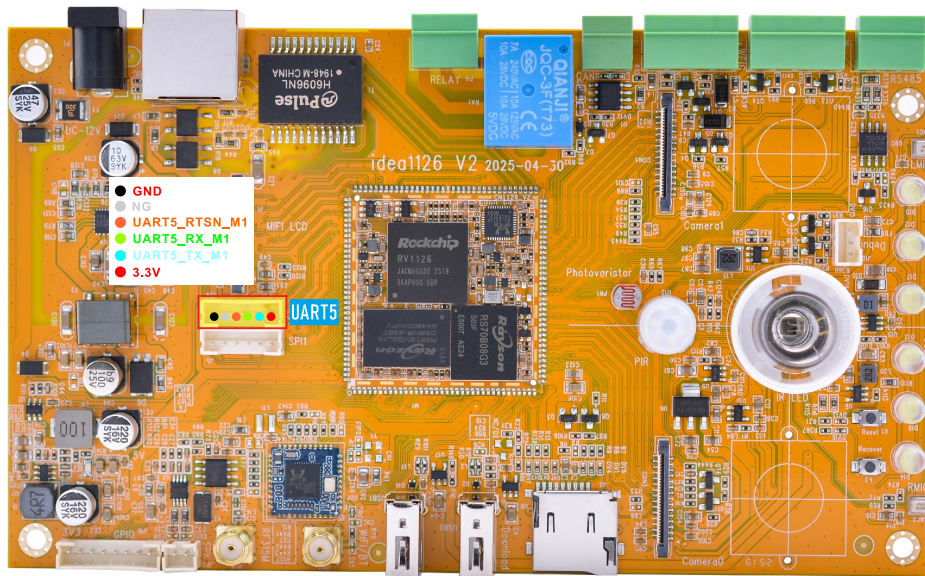
Step 5: Configure CAN as the sender.

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



6.17 UART5

Step 1: Short circuit UART5_TX_M1 and UART5_RX_M1 pins of UART.



Step 2: UART test.

```
# com /dev/ttyS5 115200 8 0 1
```

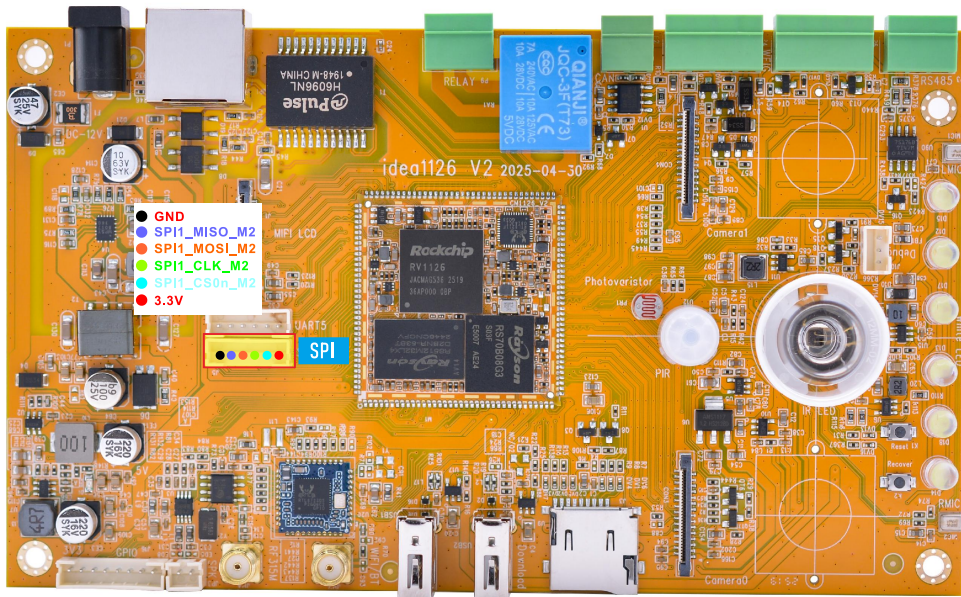
```

root@rv1126b-buildroot:/# com /dev/ttyS5 115200 8 0 1
port = /dev/ttyS5
baudrate = 115200
cs = 8
parity = 0
stopb = 1
KKKKKKKKKK
RECV: KKKKKKKKKK
23232323232
RECV: 23232323232
0000000
RECV: 0000000
898UIUIUIUIU
RECV: 898UIUIUIUIU
J
RECV: J

```

6.18 SPI

Step 1: Short circuit SPI1_MOSI_M2 and SPI1_MISO_M2 pins of SPI.



Step 2: SPI test:

```
# spidev-test -D /dev/spidev1.0
```

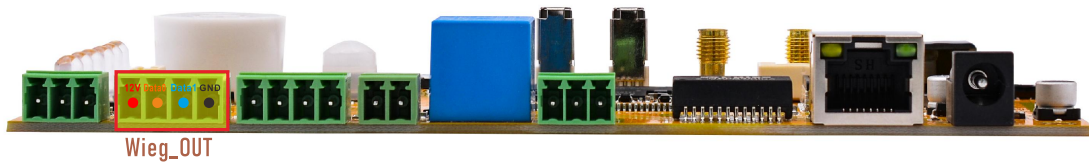
```

root@rv1126b-buildroot:/# spidev-test -D /dev/spidev1.0
spi mode: 0
bits per word: 8
max speed: 10000000 Hz (1000 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

```

6.19 Wiegand



Wiegand output test:

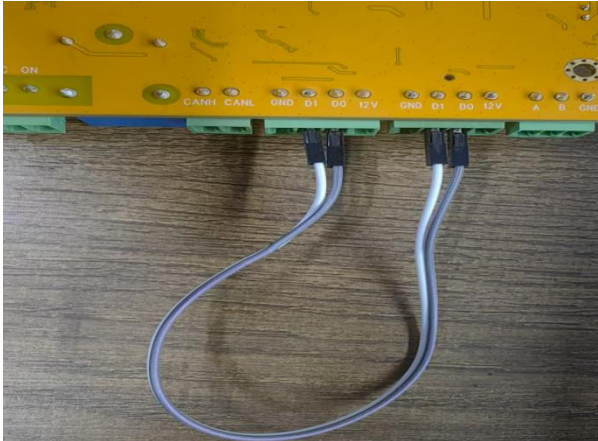
```
# wg_out_test
```

```
root@rv1126b-buildroot:/# wg_out_test
hid is 0xa26f
pid is 0x8b5c
###Setup Wiegand output 26bit or 34bit.###
[1] Wiegand output 26bit
[2] Wiegand output 34bit
[0] quit
-----Select-----
Enter your Selection: 1
[ 8835.976129] wg_output_26bit: HID:6f PID:8b5c
###Setup Wiegand output 26bit or 34bit.###
[1] Wiegand output 26bit
[2] Wiegand output 34bit
[0] quit
-----Select-----
Enter your Selection:
```

Verification: wg_output_26bit: 6f 8b5c

```
(1)Bit Grouping Structure:
1 | 01101111 | 1000101101011100 | 0
^
  Facility   Card Number   ^
(2)Breakdown:
P1 (Leading parity bit): 1
Facility Code: 01101111 = 0x6F
Card Number: 1000101101011100 = 0x8B5C
P2 (Trailing parity bit): 0
```

Wiegand input test:



The Wiegand output D0 and D1 interfaces are respectively connected to the Wiegand input D0 and D1 interfaces.

Open a command window using the adb shell.

```
F:\>adb shell
root@rv1126b-buildroot:/# _
```

Wiegand sending and receiving test.

```

root@rv1126b-buildroot:/# wg_out_test
hid is 0xa26f
pid is 0x8b5c
###Setup Wiegand output 26bit or 34bit.###
[1] Wiegand output 26bit
[2] Wiegand output 34bit
[0] quit
-----Select-----
Enter your Selection: 1
[ 110.938836] wg_output_26bit: HID:6f PID:8b5c
###Setup Wiegand output 26bit or 34bit.###
[1] Wiegand output 26bit
[2] Wiegand output 34bit
[0] quit
-----Select-----
Enter your Selection: 2
[ 113.179818] wg_output_34bit: HID:a26f PID:8b5c
###Setup Wiegand output 26bit or 34bit.###
[1] Wiegand output 26bit
[2] Wiegand output 34bit
[0] quit
-----Select-----
Enter your Selection: _

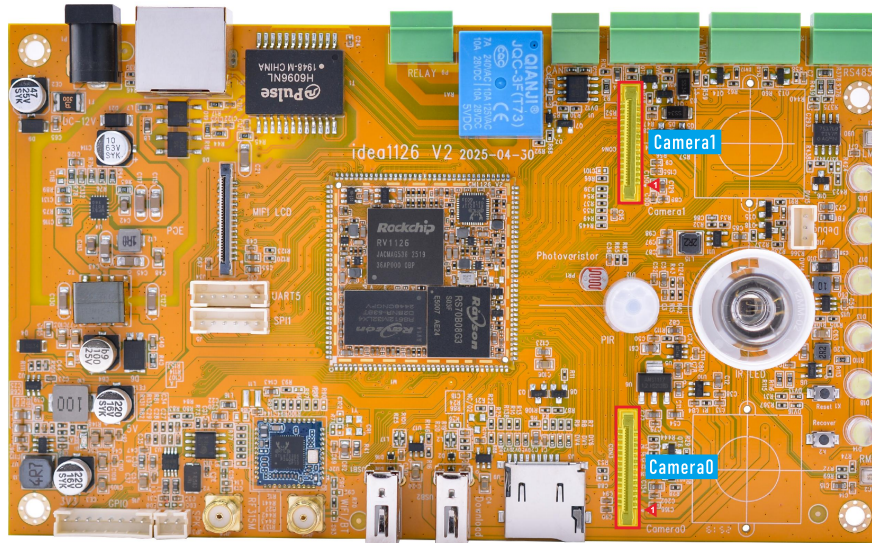
root@rv1126b-buildroot:/# wg_input_test
Listening for Wiegand data on /dev/input/by-path/platform-rockchip-wg-input-event
...
(Press Ctrl+C to exit)

26-bit: HID:6f PID:8b5c Raw=0x2df16b8
34-bit: HID:a26f PID:8b5c Raw=0x344df16b9
_

```

6.20 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)1:4:16:4
/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
[ 84.634247] rkisp_hw 21d00000.isp: set isp clk = 500000000Hz
/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,
```

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
[ 126.715628] 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
[ 126.725242] rkCIF-mipi-lvds2: stream[0] start streaming
[ 126.725583] rockchip-mipi-csi2 mipi2-csi2: stream on, src_sd: 00000006d476f48,
sd_name:rockchip-csi2-dphy3
[ 126.725605] rockchip-mipi-csi2 mipi2-csi2: stream ON
[ 126.725658] rockchip-csi2-dphy3: dphy3, data_rate_mbps 892
[ 126.725686] rockchip-csi2-dphy csi2-dphy3: csi2_dphy_s_stream stream on:1, dphy3, ret 0
[ 126.725699] imx415 1-0036: s_stream: 1. 3864x2192, hdr: 0, bpp: 10
[ 126.835043] rkCIF-mipi-lvds2: Warning: vblank need >= 1000us if isp work in online, cur 859 us
[ 126.868802] rkCIF-mipi-lvds2: Warning: vblank need >= 1000us if isp work in online, cur 859 us
Redistribute latency...
0:00:02.0 / 99:99:99.
  
```

6.21 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.

```

# gst-play-1.0 --videosink="waylandsink fullscreen=true"
/mnt/udisk/video/4KP25/4K25FPS.mp4 --audiosink="alsasink device=hw:0,0"
  
```

Command explanation:

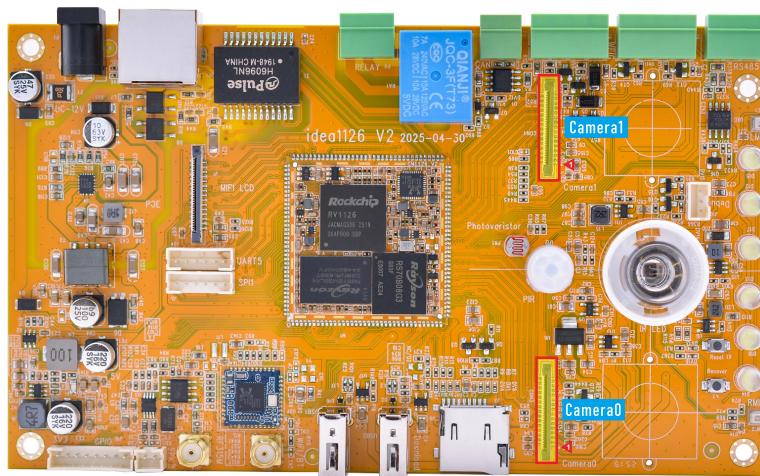
- /mnt/udisk/video/4KP25/4K25FPS.mp4: The media file path to be played.
- --audiosink="alsasink device=hw:0,0": Specifies the audio output device as hw:0,0.

```
root@rv1126b-buildroot:/#
/mnt/udisk/video/4KP25/4K25FPS.mp4 --audiosink="alsasink device=hw:0,0"n=true"
Press 'k' to see a list of keyboard shortcuts.
Now playing /mnt/udisk/video/4KP25/4K25FPS.mp4
Redistribute latency...
Redistribute latency...
mpp[1851]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen 2025-07-01 fix[base]: Fix enc cfg size
mpp[1851]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen 2025-07-01 fix[base]: Fix enc cfg size
mpp[1851]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen 2025-07-01 fix[base]: Fix enc cfg size
mpp[1851]: mpp: unable to create enc vp8 for soc rv1126b unsupported
mpp[1851]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen 2025-07-01 fix[base]: Fix enc cfg size
mpp[1851]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen 2025-07-01 fix[base]: Fix enc cfg size
Redistribute latency...
Redistribute latency...
mpp[1851]: h264d_api: is_avcC=1
mpp[1851]: mpp_buf_slot: mismatch h_stride_by_pixel 4032 - 3840
mpp[1851]: mpp_buf_slot: mismatch h_stride_by_byte 4032 - 3840
mpp[1851]: mpp_buf_slot: mismatch size_total 13063680 - 16588800
mpp[1851]: mpp_buf_slot: mismatch h_stride_by_pixel 4032 - 3840
mpp[1851]: mpp_buf_slot: mismatch h_stride_by_byte 4032 - 3840
mpp[1851]: mpp_buf_slot: mismatch size_total 13063680 - 16588800
Redistribute latency...
0:00:02.2 / 0:02:44.6
```

7. Buildroot IPC Test

Note: This chapter focuses exclusively on features and tests specific to the IPC version. For any peripherals not mentioned in this chapter (such as Serial Terminal, USB, Ethernet, etc.), the testing procedures are identical to the standard Buildroot version and will not be repeated here. Please refer to the Buildroot Testing Guide for those instructions.

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:

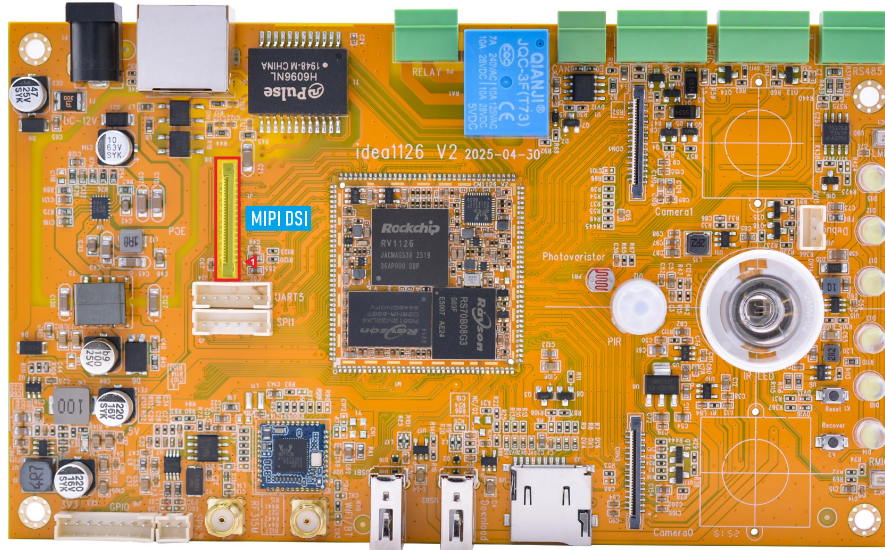
- `killall -9 rkipc` : Forcefully terminate the running IPC process.
- `rkipc &` : Restart the IPC application in the background.

```
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 00000000beebe2f online 0 combo
-1 mst -1
[ 161.944095] mpp_vcodec_chan: mpp_vcodec_chan_destory: destroy chan 0 done

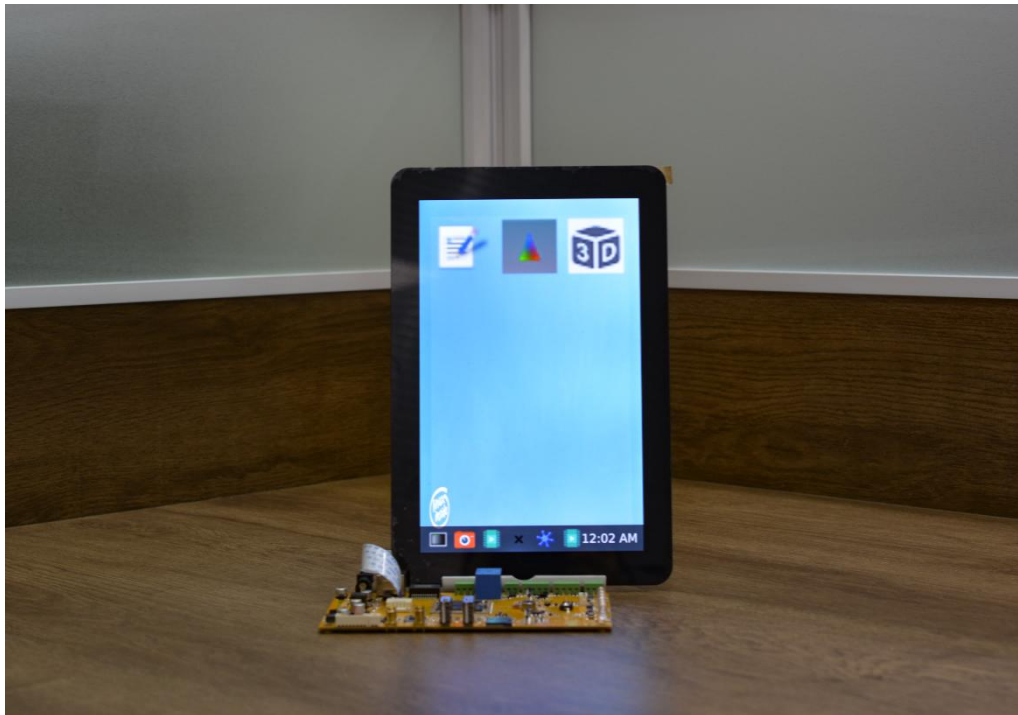
162.254805] rkvpss-vir0: rkvpss_stream_stop id:3 timeout
162.284451] kmpp_venc_chan_put_frm: invalid chan id 0
162.317606] rkCIF-mipi-lvds: sStream[0] start stopping, total mode 0x8, cur 0x8
[ 162.364606] rockchip-mipi-csi2 mipi0-csi2: stream off, src_sd: 000000020ec0176,
sd_name:rockchip-csi2-dphy0
[ 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 Display

The default MIPI display resolution on IDEA1126B-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.

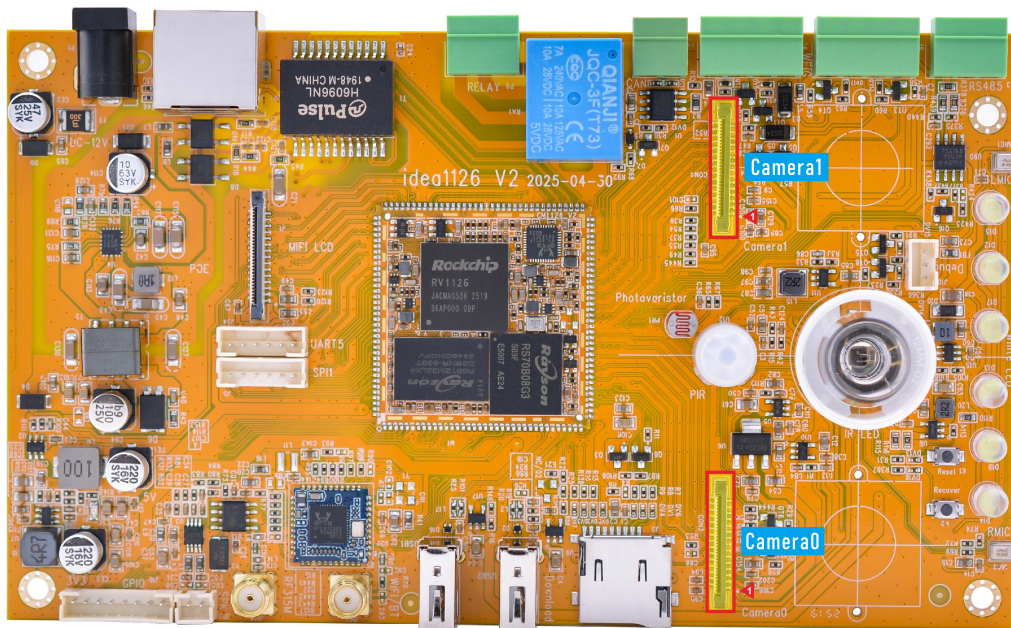
7.2 RKIPC

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.

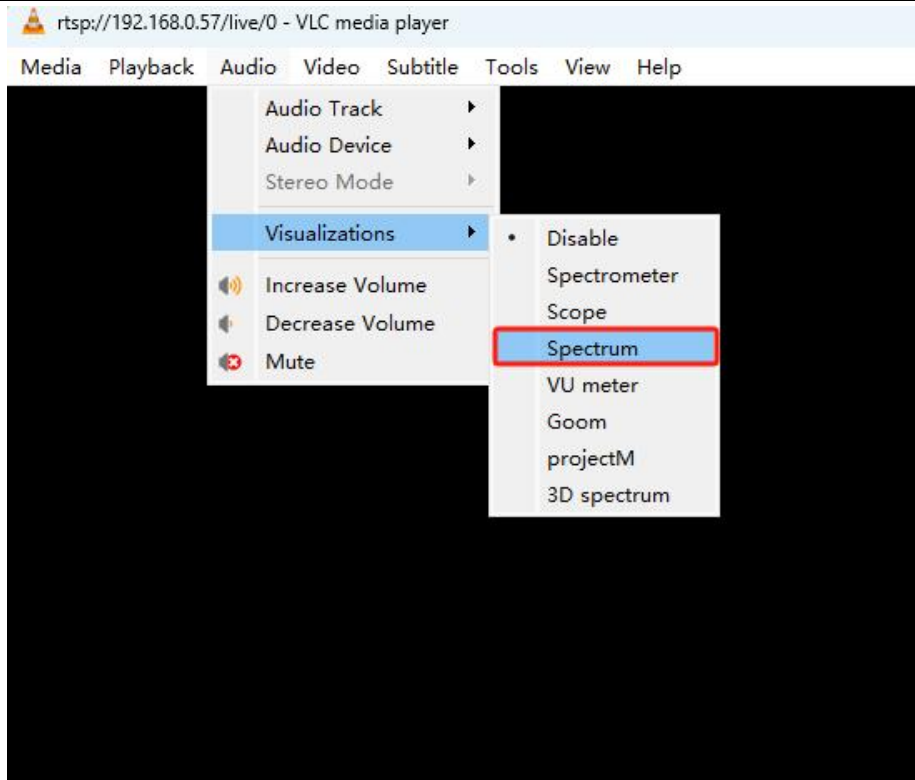
Executing the following instructions to ensure that the RKIPC process is running

```
# ps | grep rkipc
```

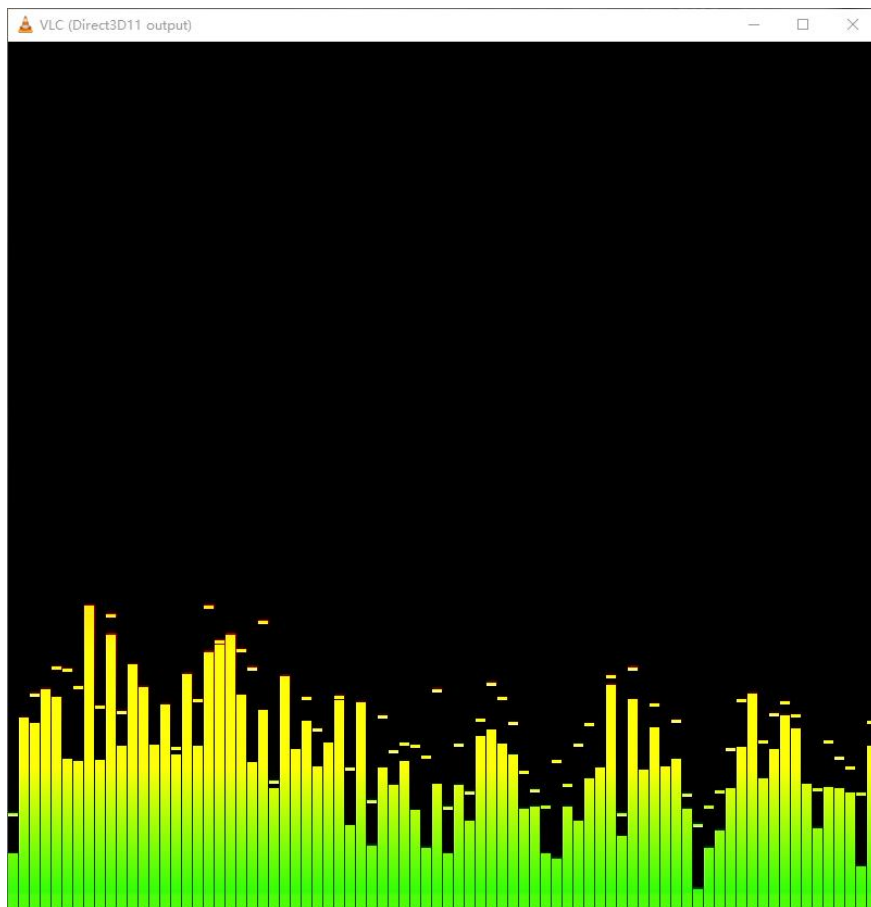
```
# ps | grep rkipc
433 root    1234 S    rkipc -a /oem/usr/share/iqfiles
574 root    1176 S    grep rkipc
```

7.2.1 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:

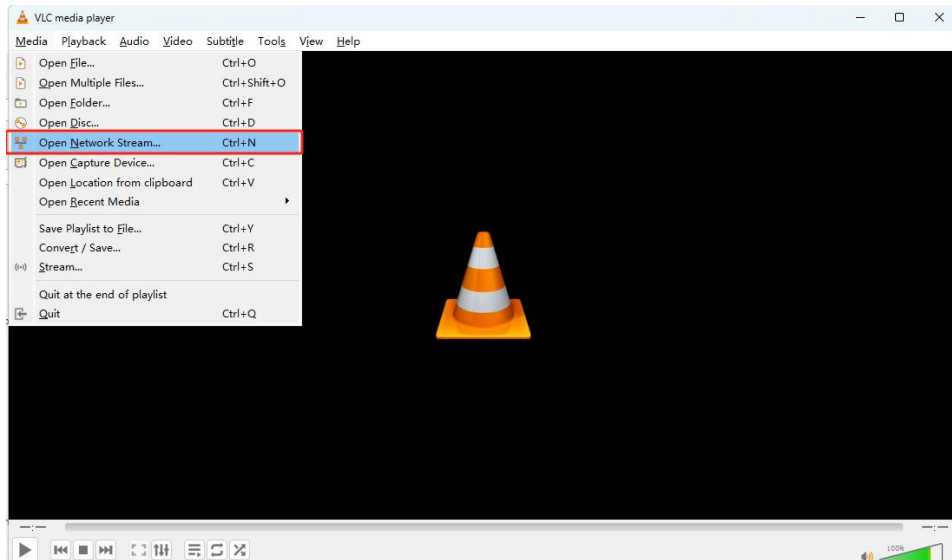


At this point, when speaking to the MIC, the spectrum effect is as follows:



7.2.2 RTSP

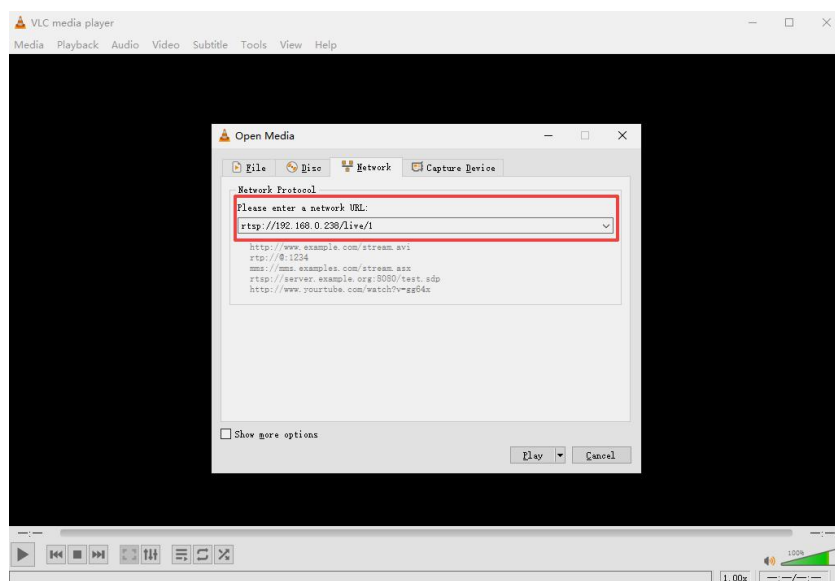
The video footage from the camera can be previewed within the same local area network. After the device is connected to the network, the RTSP software on the PC (such as VLC) can be used to enable network streaming.

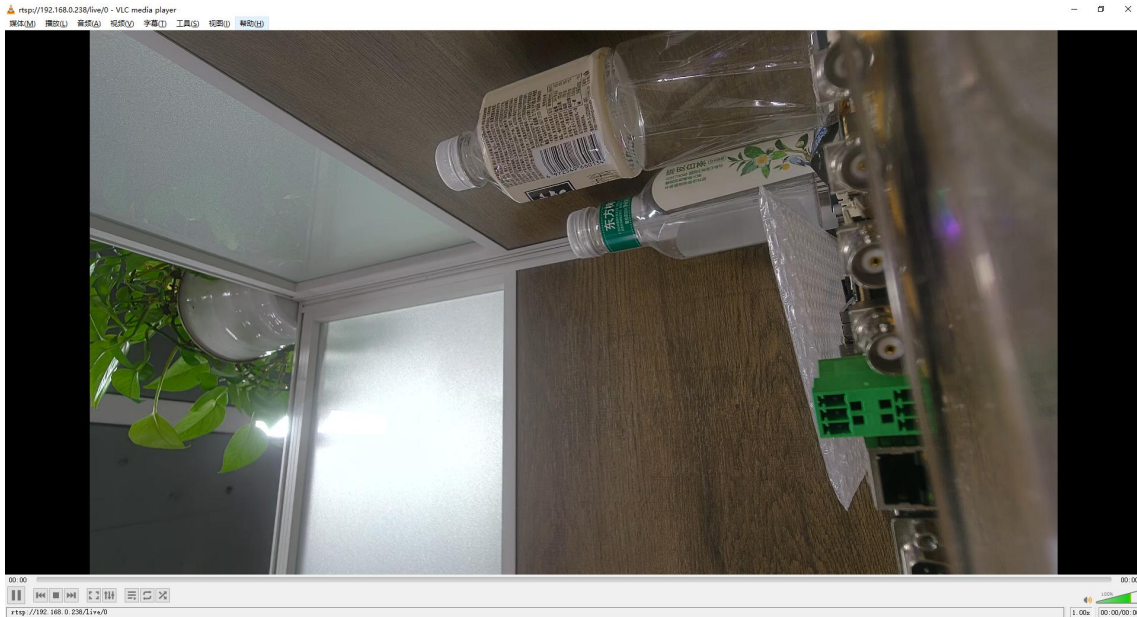


Enter the following address:

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

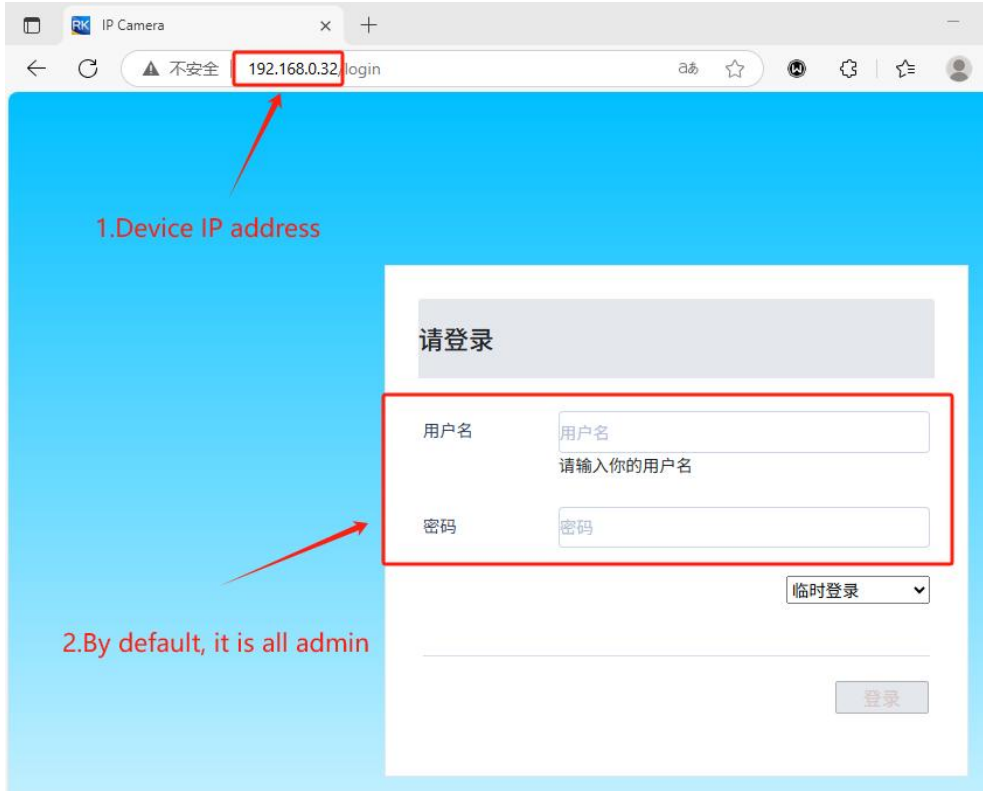
For example, in Section 7.4, using the Ethernet IP address 192.168.0.238, input `rtsp://192.168.0.238/live/0` and click the play button to preview the video from Camera 0. The WiFi streaming method is identical





7.2.3 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.

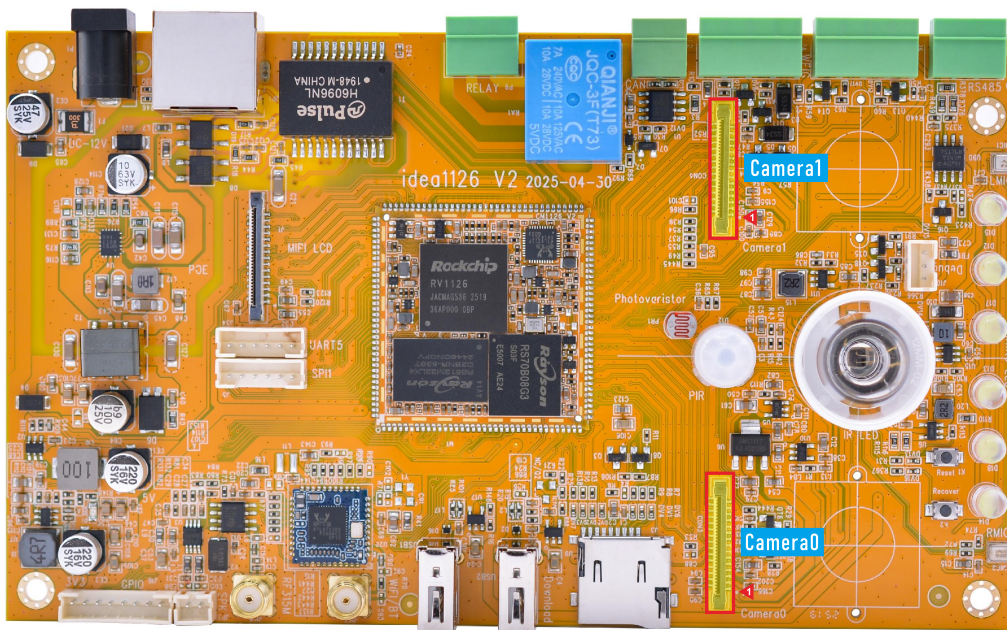


The preview effect is as follows:



7.3 Camera Streaming

Step 1: Power on after connecting the camera

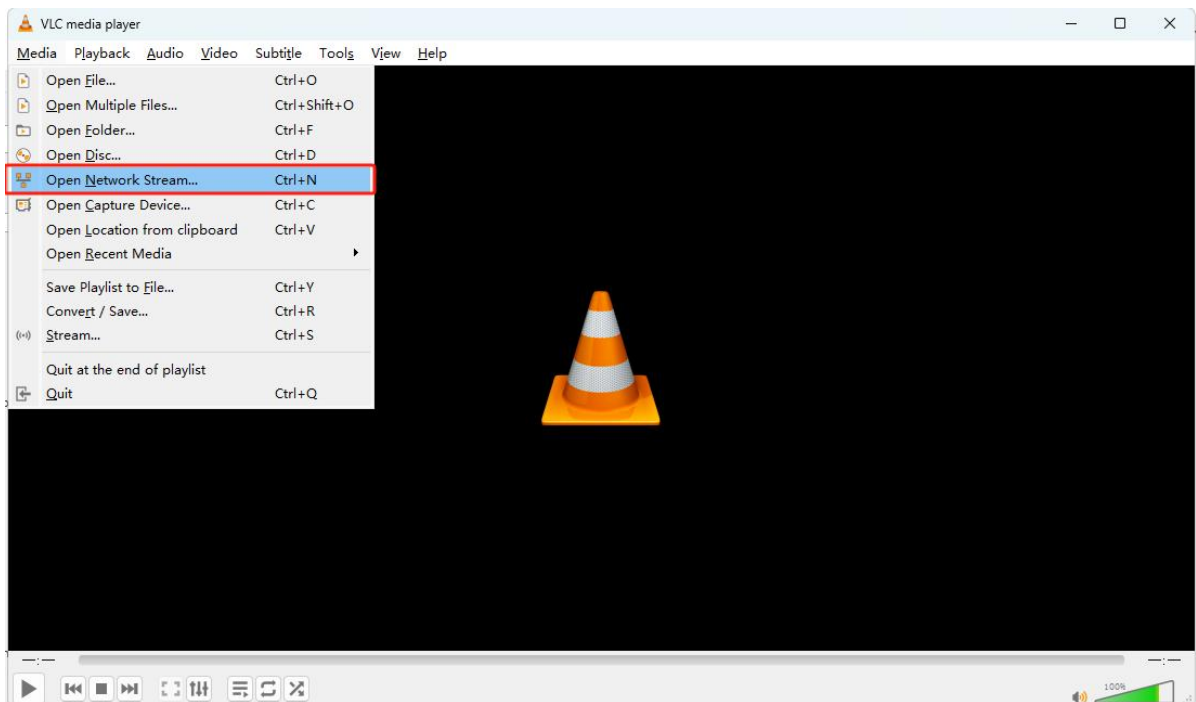


Step 2: Start Streaming Execute the following command to start the dual-camera stream.

```
# 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)

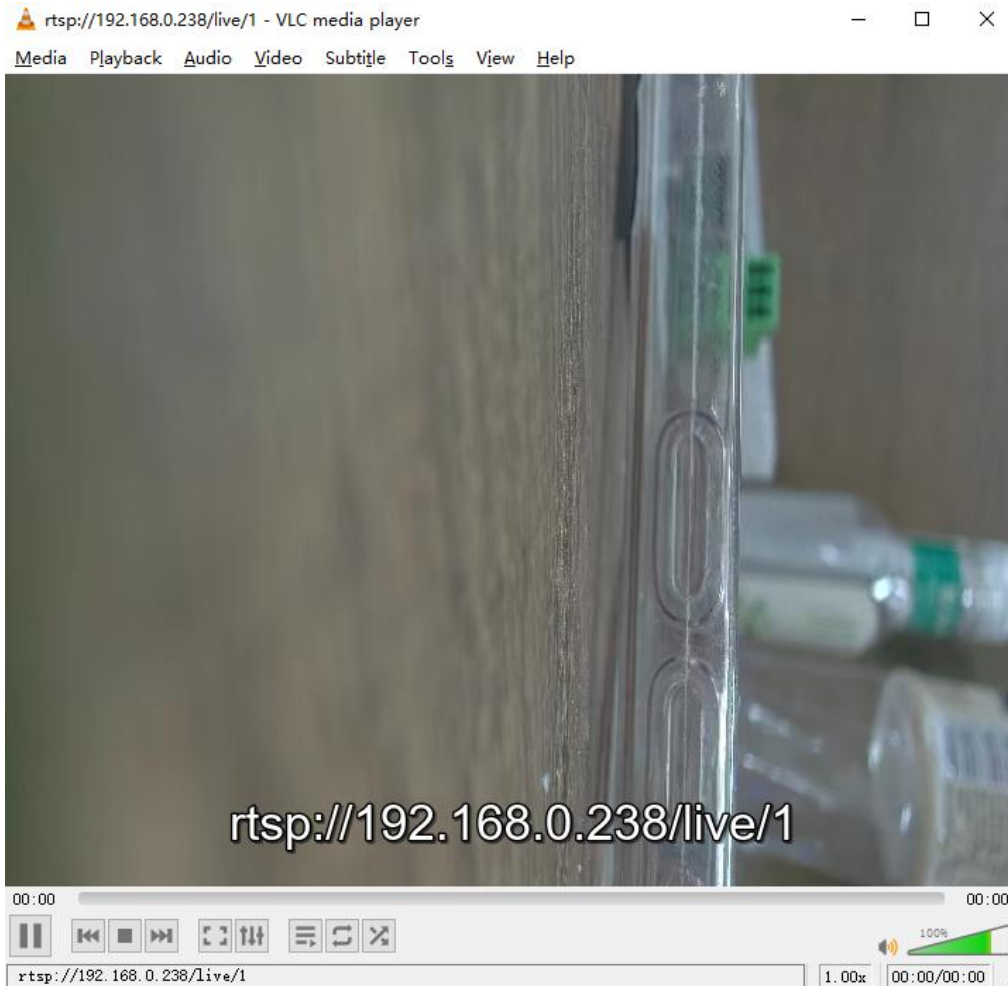
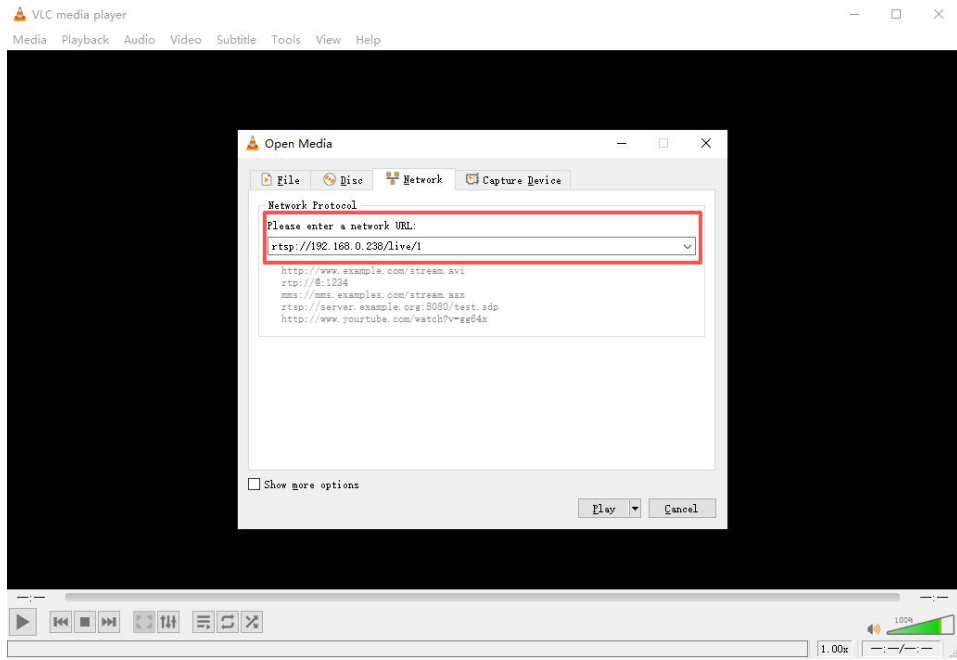
can be used to enable network streaming.



Enter the following address:

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

For example, in Section 6.4, 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.

7.4 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 rerelative 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 rerelative 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 rgm vpss vgs tde avs wbc vo vi ai ao aenc adec
log_level: 0 1 2 3 4 5 6

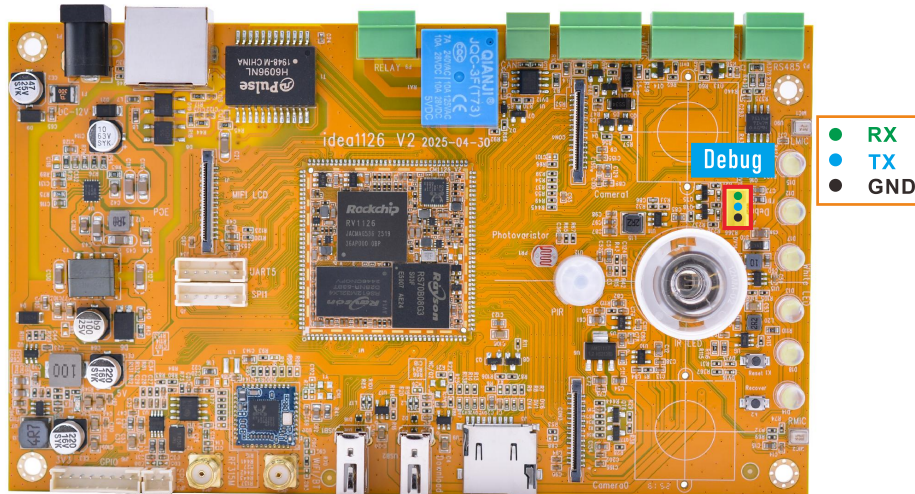
rockit default level 4, can use export rt_log_level=x, x=0,1,2,3,4,5,6 change
(null) 10:26:18-749 {read_log_level :083} text is all=4
(null) 10:26:18-749 {read_log_level :085} module is all, log_level is 4
(null) 10:26:18-749 {dump_version :060}
-----
(null) 10:26:18-749 {dump_version :061} rockit version: 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 relative 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 relative 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

```

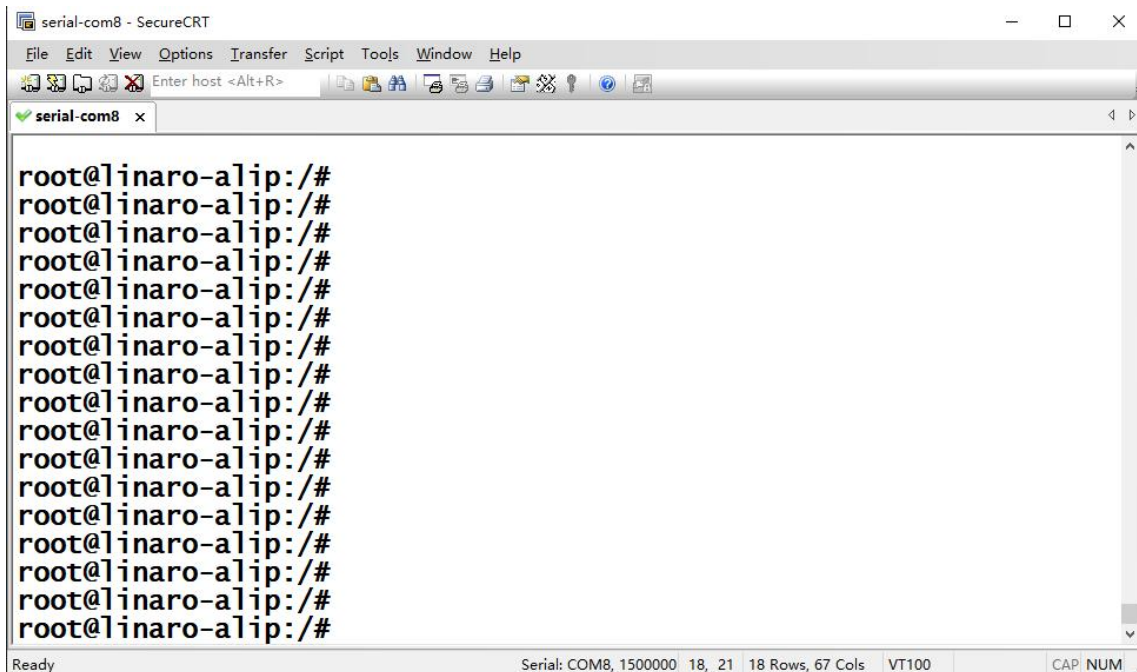
Expected Result: The video should play smoothly on the display screen.

8. Debian12 Test

6.1 Serial Terminal

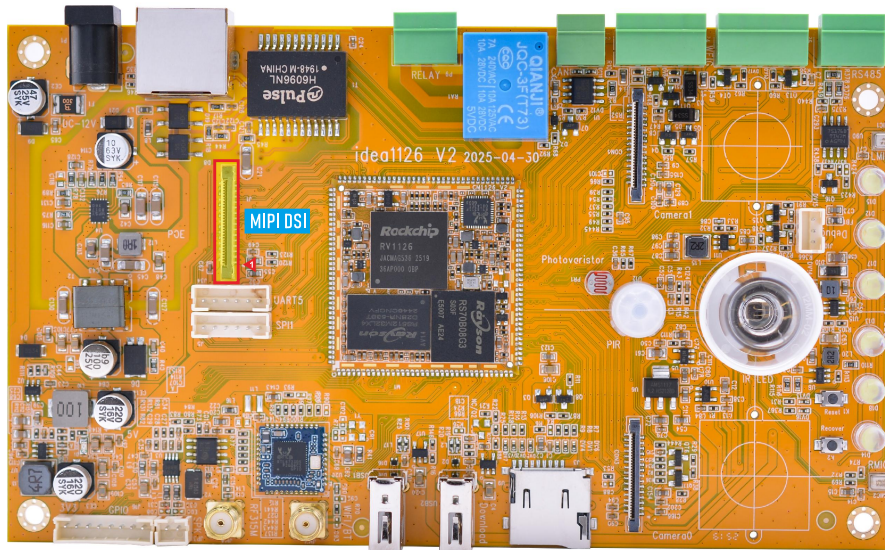


Connect the board and PC with USB Serial cable, then power on, the terminal will output boot information. The default baudrate is 1500000.

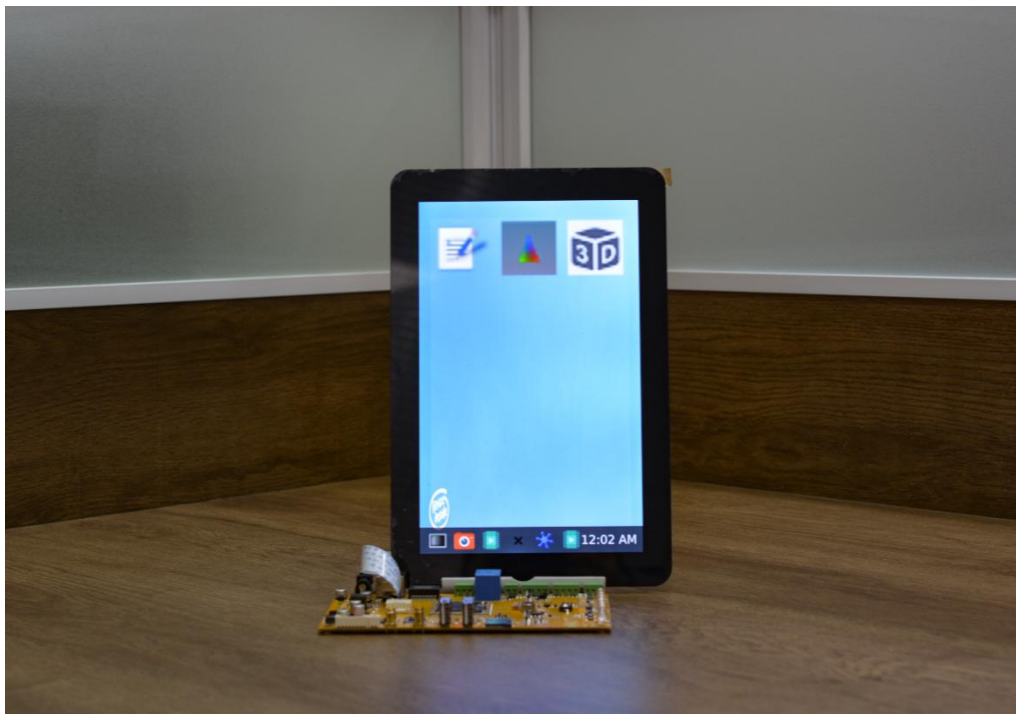


8.2 Display

The default MIPI display resolution on IDEA1126B-P is 800×1280 @60Hz.



The display effect diagram is as follows:



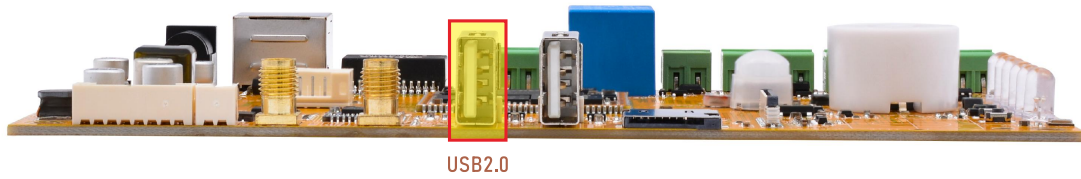
8.3 USB

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

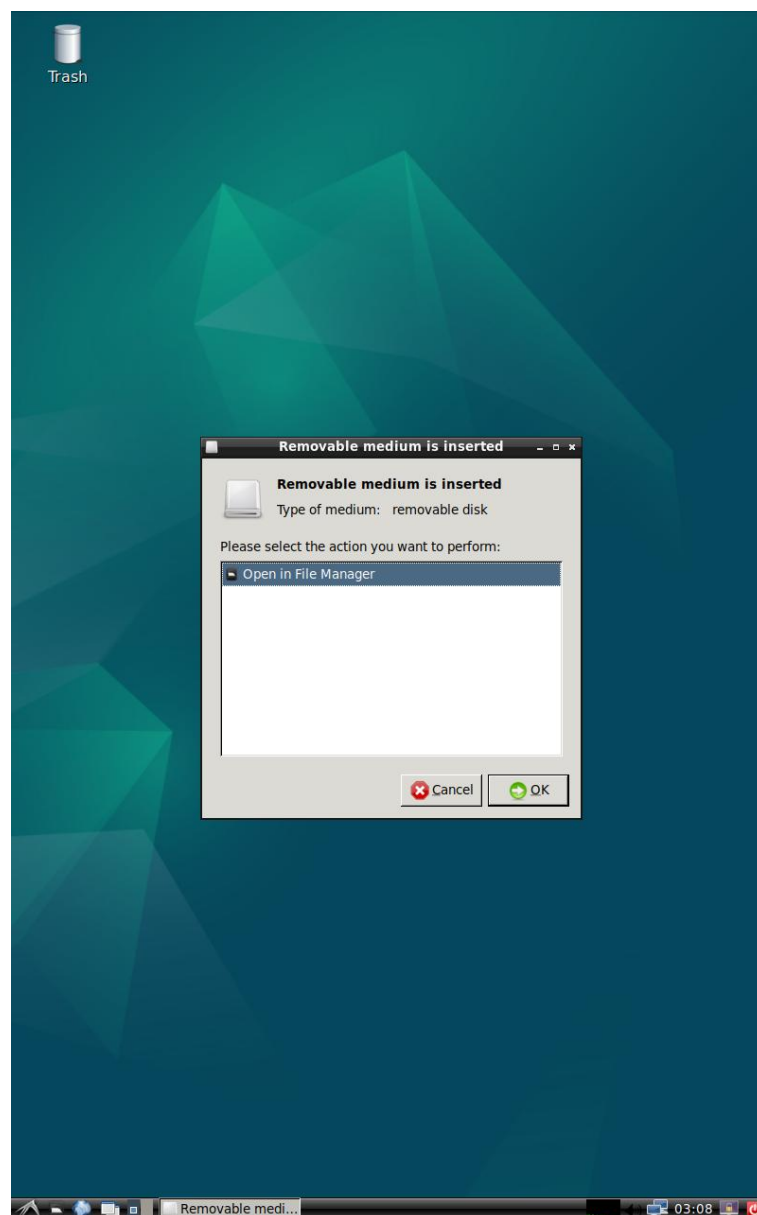
6.3.1 USB2.0 Host

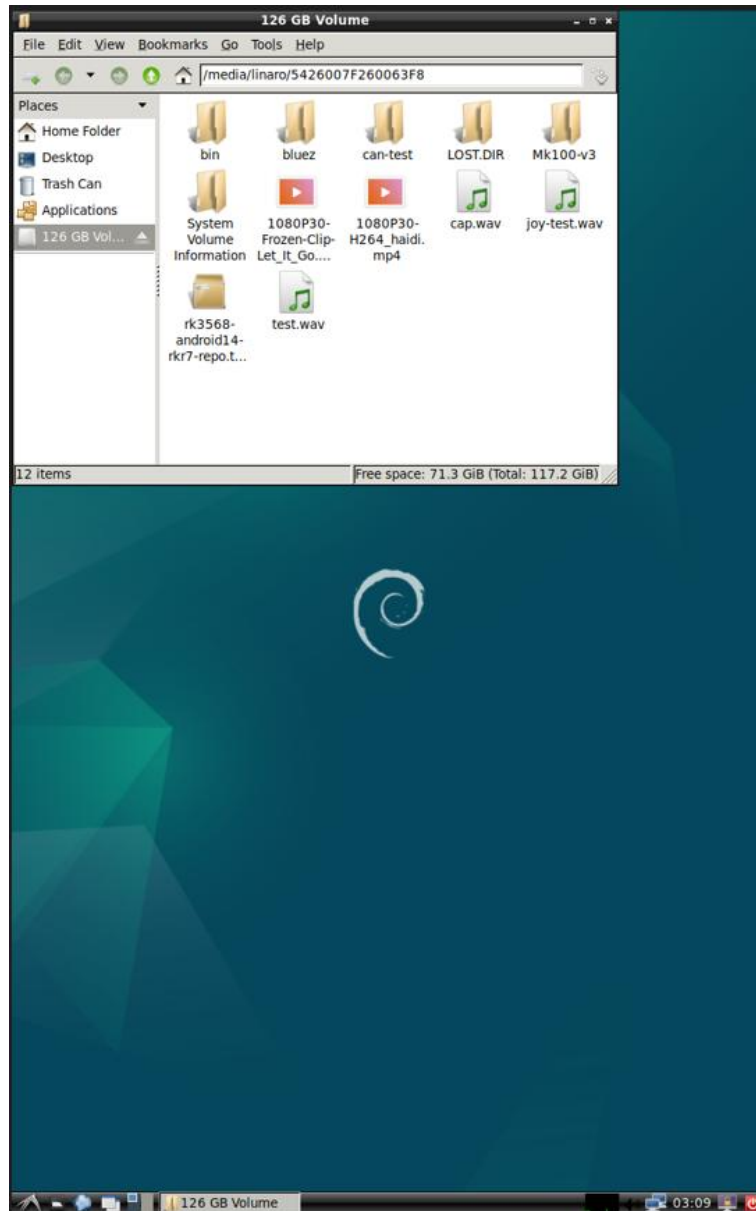
The USB2.0 Host port can be used to connect USB peripherals such as a mouse,

keyboard, flash drive, and other USB devices.



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.



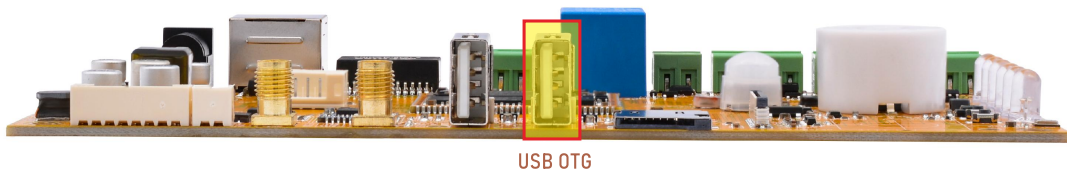


6.3.2 USB2.0 OTG (Default Device Mode)

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 Type-A cable.

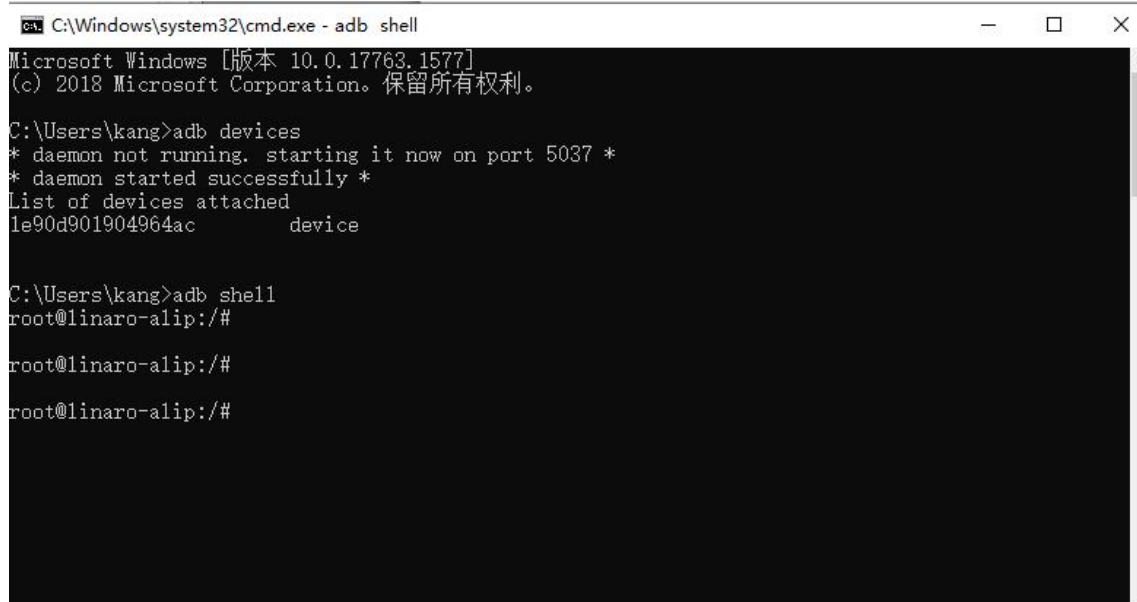


Step 2: Install ADB driver on Windows system.

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
```



```
C:\Windows\system32\cmd.exe - adb shell
Microsoft Windows [版本 10.0.17763.1577]
(c) 2018 Microsoft Corporation。保留所有权利。

C:\Users\kang>adb devices
* daemon not running. starting it now on port 5037 *
* daemon started successfully *
List of devices attached
1e90d901904964ac    device

C:\Users\kang>adb shell
root@linaro-alip:/#

root@linaro-alip:/#

root@linaro-alip:/#
```

Switching OTG modes:

- To switch from Device mode to Host mode:

```
# echo host > /sys/kernel/debug/usb/21500000.usb/mode
```

```

root@linaro-alip:/# echo host > /sys/kernel/debug/usb/21500000.usb/mode
root@linaro-alip:/# [ 163.798437] dwc3 21500000.usb: request 00000000f4e01398 was not queued to ep0out
[ 163.798614] android_work: sent uevent USB_STATE=DISCONNECTED
[ 163.908166] phy phy-21400000.usb2-phy.2: illegal mode
[ 163.908199] xhci-hcd xhci-hcd.0.auto: xHCI Host Controller
[ 163.908439] xhci-hcd xhci-hcd.0.auto: new USB bus registered, assigned bus number 1
[ 163.908595] xhci-hcd xhci-hcd.0.auto: hcc params 0x0220fe64 hci version 0x110 quirks 0x0000008022010010
[ 163.908638] xhci-hcd xhci-hcd.0.auto: irq 82, io mem 0x21500000
[ 163.908758] xhci-hcd xhci-hcd.0.auto: xHCI Host Controller
[ 163.909081] xhci-hcd xhci-hcd.0.auto: new USB bus registered, assigned bus number 2
[ 163.909124] xhci-hcd xhci-hcd.0.auto: Host supports USB 3.0 SuperSpeed
[ 163.909306] usb usb1: New USB device found, idVendor=1d6b, idProduct=0002, bcdDevice= 6.01
[ 163.909322] usb usb1: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[ 163.909332] usb usb1: Product: xHCI Host Controller
[ 163.909339] usb usb1: Manufacturer: Linux 6.1.141 xhci-hcd
[ 163.909346] usb usb1: SerialNumber: xhci-hcd.0.auto
[ 163.909998] hub 1-0:1.0: USB hub found
[ 163.910074] hub 1-0:1.0: 1 port detected
[ 163.910546] usb usb2: We don't know the algorithms for LPM for this host, disabling LPM.
[ 163.910676] usb usb2: New USB device found, idVendor=1d6b, idProduct=0003, bcdDevice= 6.01
[ 163.910688] usb usb2: New USB device strings: Mfr=3, Product=2, SerialNumber=1
[ 163.910697] usb usb2: Product: xHCI Host Controller
[ 163.910704] usb usb2: Manufacturer: Linux 6.1.141 xhci-hcd
[ 163.910711] usb usb2: SerialNumber: xhci-hcd.0.auto
[ 163.911226] hub 2-0:1.0: USB hub found
[ 163.911271] hub 2-0:1.0: 1 port detected
[ 163.943912] read descriptors
[ 163.944001] read strings
[ 165.004092] read descriptors
[ 165.004145] read strings
  
```

- To switch back to Device mode:

```

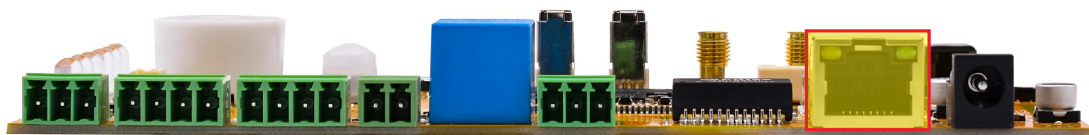
# echo device > /sys/kernel/debug/usb/21500000.usb/mode
# usbdevice start
  
```

```

root@linaro-alip:/# echo device > /sys/kernel/debug/usb/21500000.usb/mode
usbdevice start
[ 78.408625] xhci-hcd xhci-hcd.0.auto: remove, state 4
[ 78.408684] usb usb2: USB disconnect, device number 1
[ 78.409783] xhci-hcd xhci-hcd.0.auto: USB bus 2 deregistered
[ 78.410122] xhci-hcd xhci-hcd.0.auto: remove, state 4
[ 78.410158] usb usb1: USB disconnect, device number 1
[ 78.411505] xhci-hcd xhci-hcd.0.auto: USB bus 1 deregistered
[2025-06-26 14:59:53] usbdevice [1994]: Sending start request to service...
[2025-06-26 14:59:53] usbdevice [1994]: Check /var/log/usb-gadget.log for usb-service log...
root@linaro-alip:/# [ 79.149296] file system registered
[ 79.469402] read descriptors
[ 79.469441] read strings
  
```

8.4 Ethernet

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



Ethernet

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
```

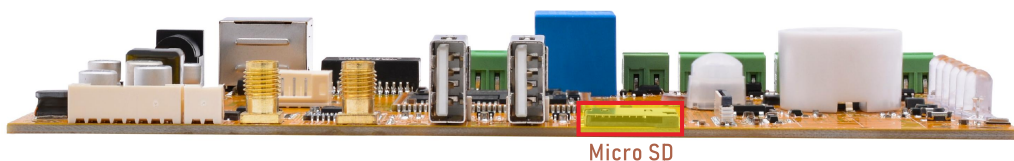
Step 3: Network connection test.

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

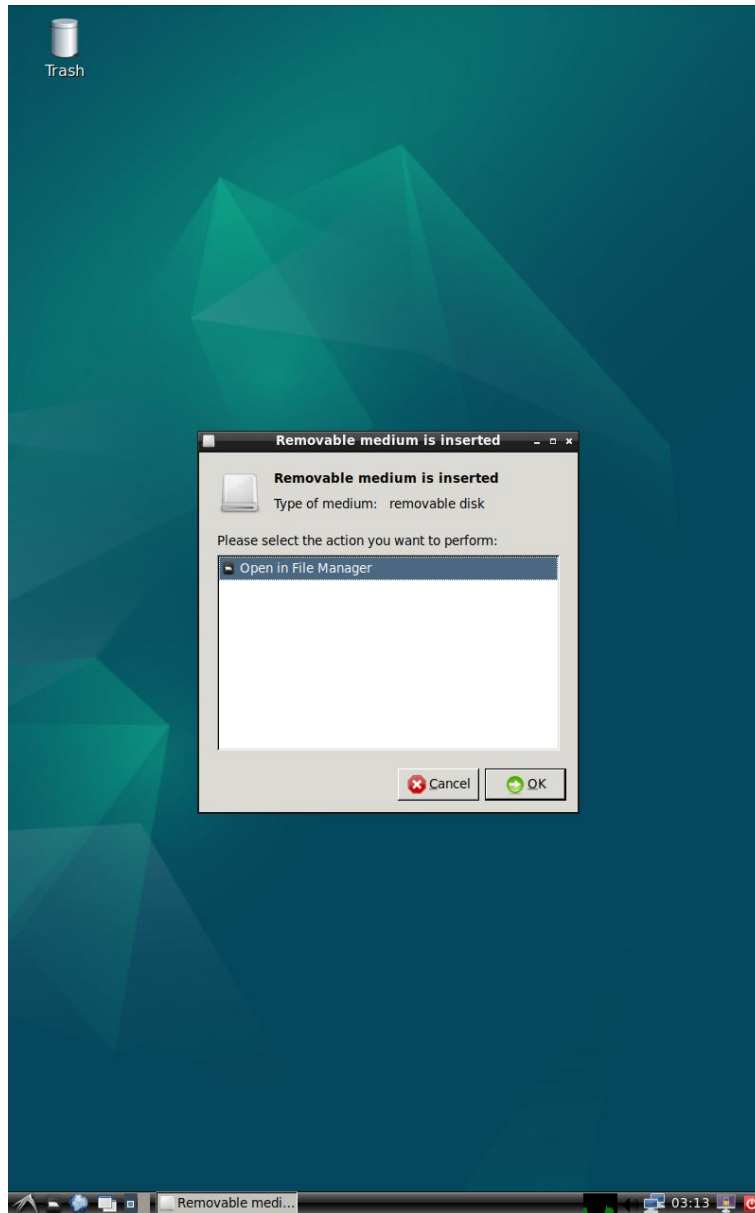
```
root@linaro-alip:/# ping -I eth0 www.armdesigner.com  
PING www.armdesigner.com (67.222.54.196) from 192.168.0.124 eth0: 56(84) bytes of data.  
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=1 ttl=50 time=196 ms  
64 bytes from www.armdesigner.com (67.222.54.196): icmp_seq=2 ttl=50 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
```

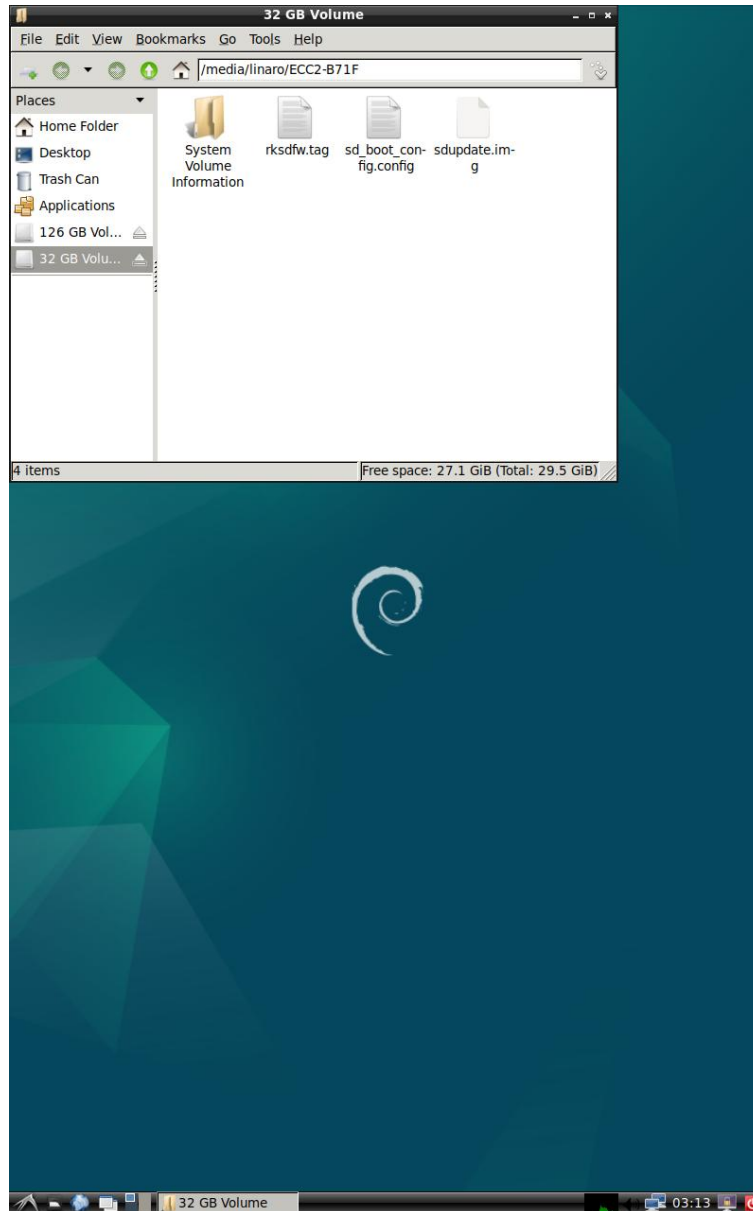
8.5 SD Card

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



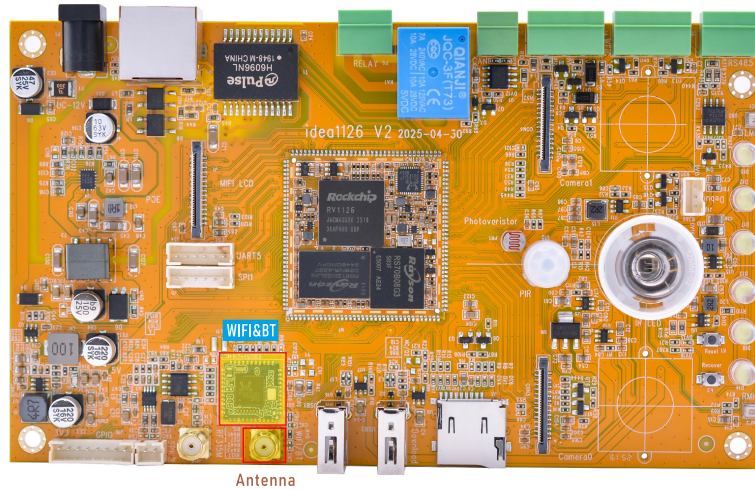
Step 2: After inserting the SD card, if it is recognized successfully, an icon will appear on the desktop. Users need to click the "ok button" in order to access the SD card.





8.6 WiFi & Bluetooth

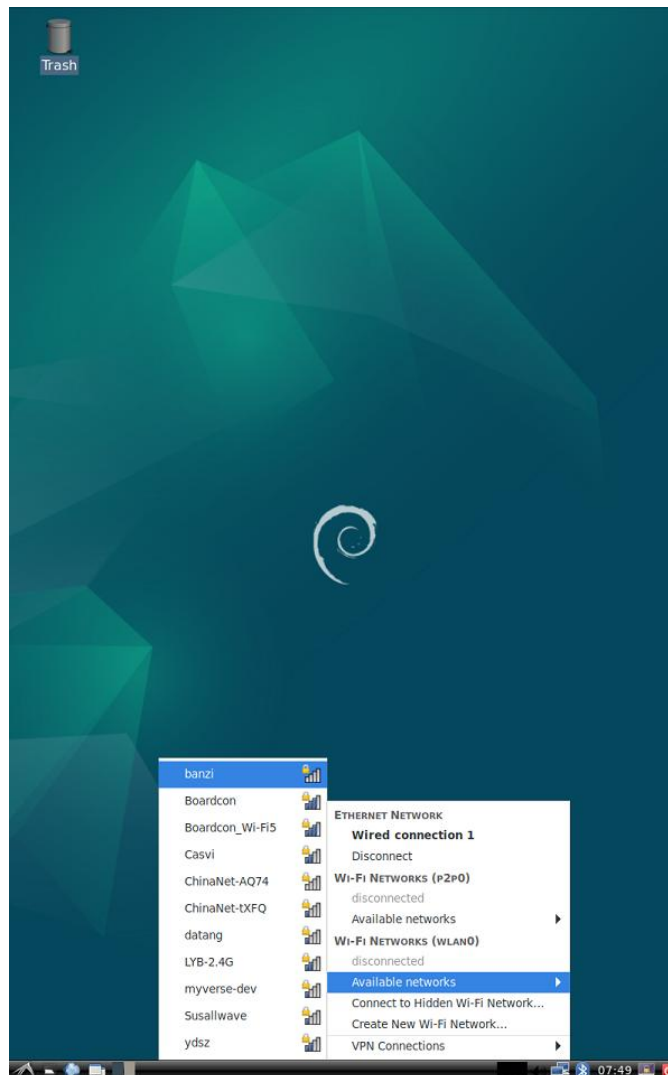
To use Wi-Fi and Bluetooth functions properly, the antenna needs to be connected.



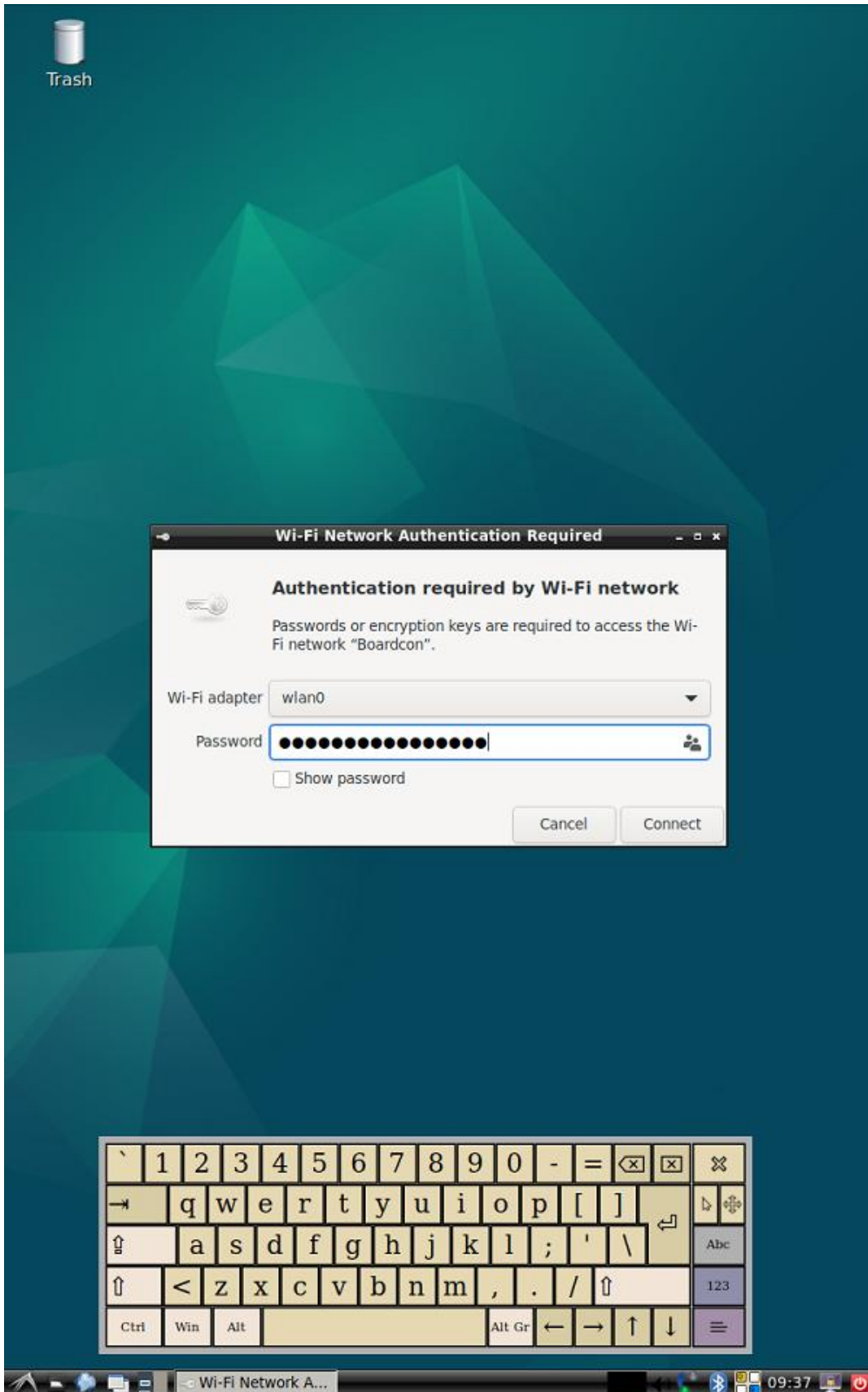
Antenna

8.6.1 WiFi

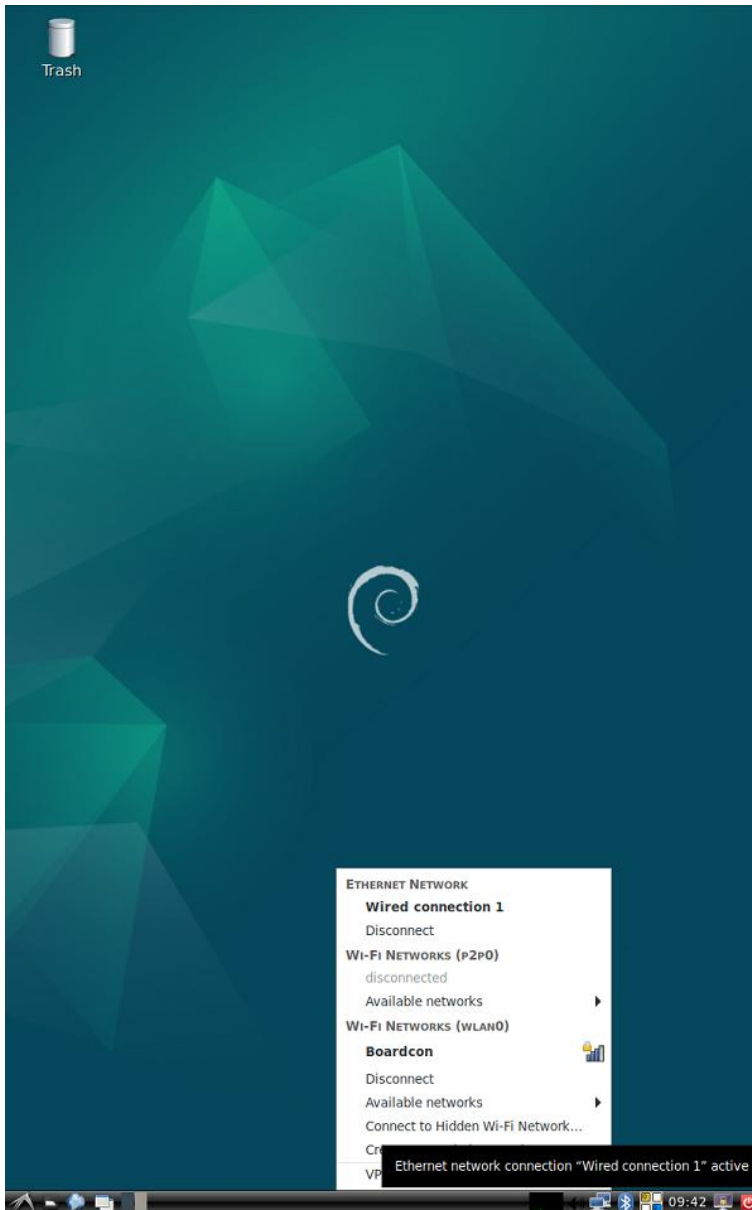
Step 1: Click the network icon in the top right corner of the interface, then select the **"Available Networks"** option to view the list of available hotspots.



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
```

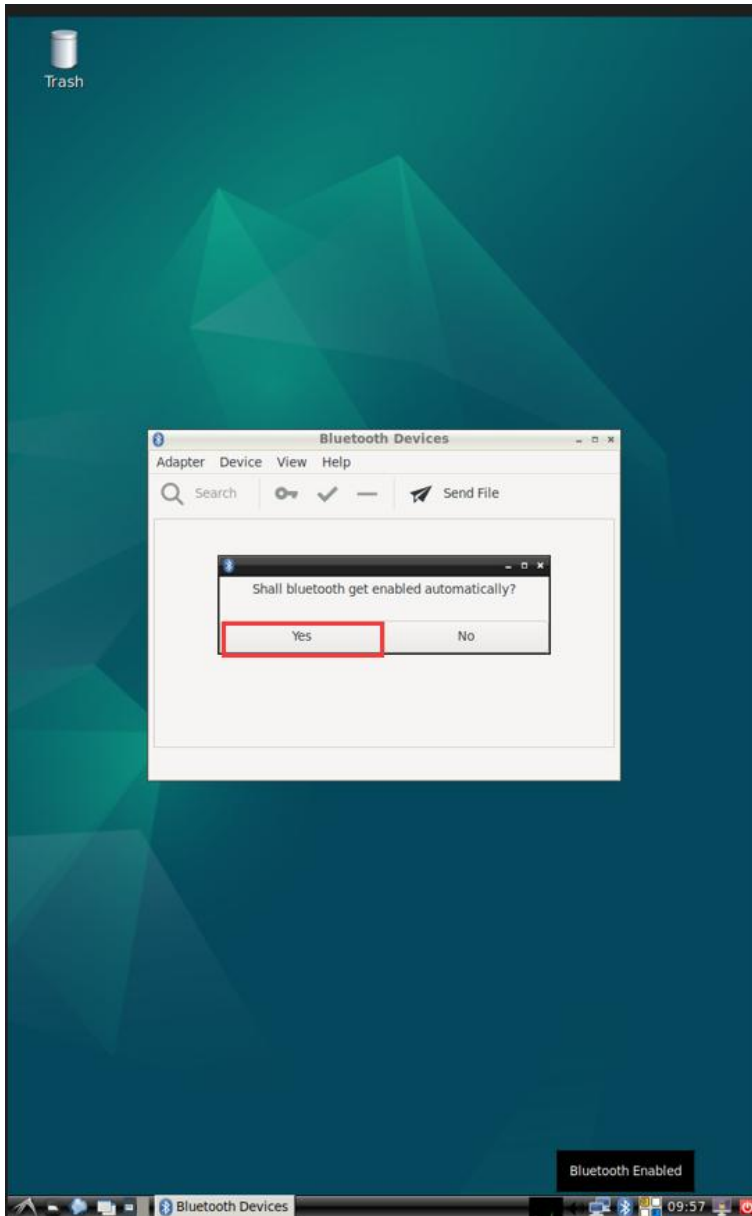
(2) Network connection test.

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

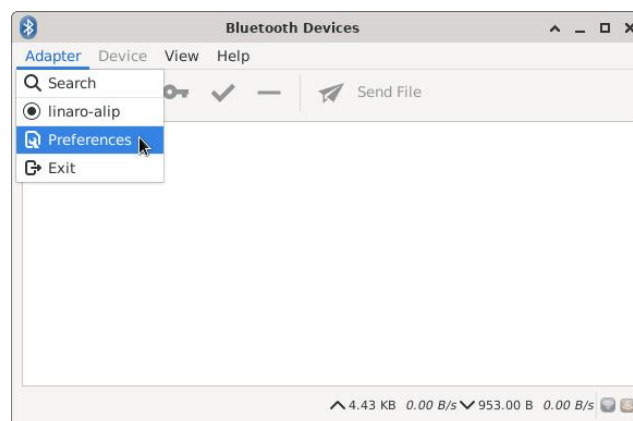
```
root@linaro-alip:/# ping -I wlan0 www.armdesigner.com
PING www.armdesigner.com (67.222.54.196) from 192.168.0.124 wlan0: 56(84) bytes of data.
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): icmp_seq=1 ttl=50 time=227 ms
64 bytes from 67-222-54-196.unifiedlayer.com (67.222.54.196): 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
```

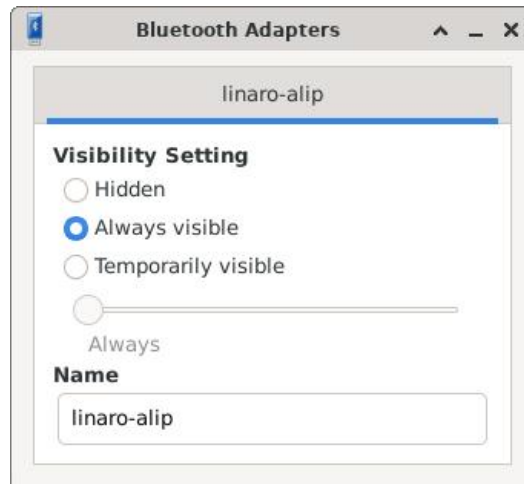
8.6.2 Bluetooth

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

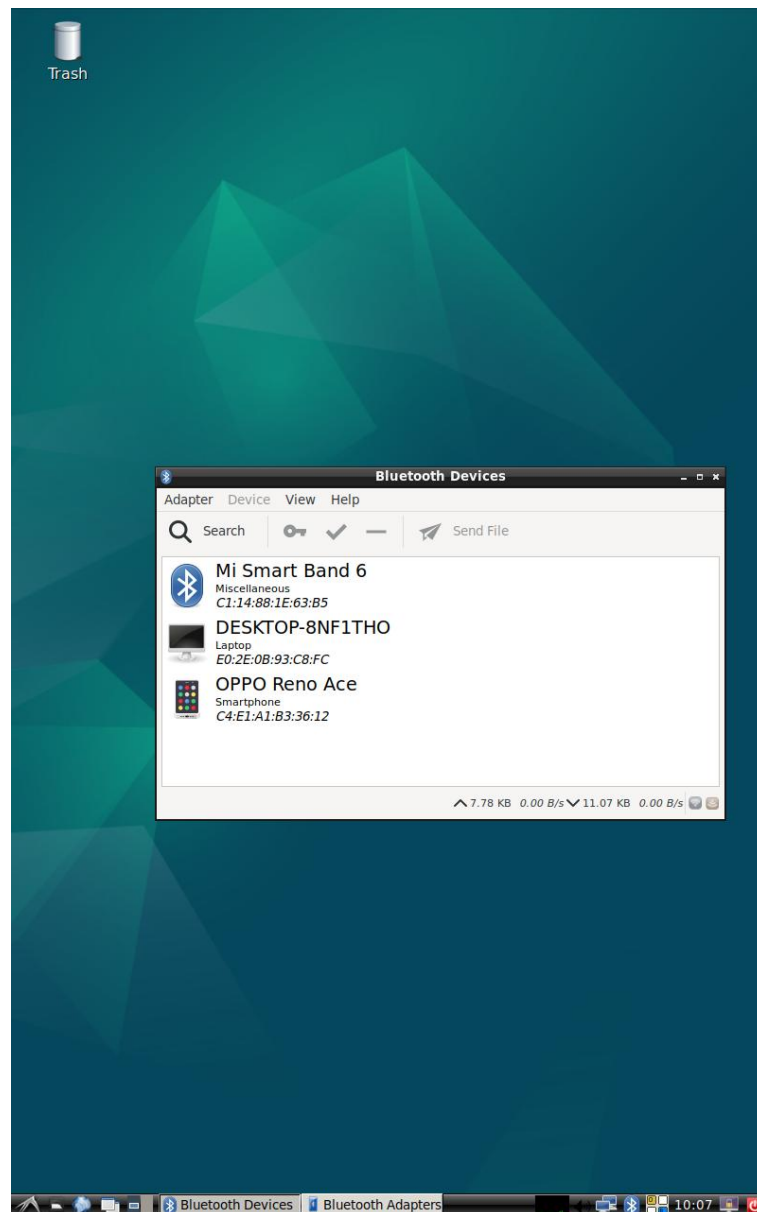


Step 2: The Bluetooth device name is hidden by default. Set it to be visible by clicking **Adapter -> Preferences -> Always visible**.





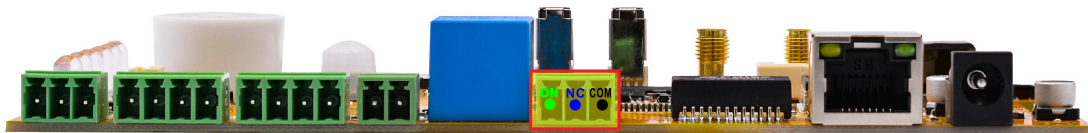
Step 3: Click Search to start searching and select the available device in the list to pair.



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

8.7 Relay

Default power-on state: The relay is in the energized (closed) state, and the normally open (**ON**) contact is connected to the common (**COM**) terminal.



Relay

- To release the relay and disconnect the (**ON**) and (**COM**) connection, execute the following command:

```
# echo 1 > /sys/class/leds/raley_power/brightness
```

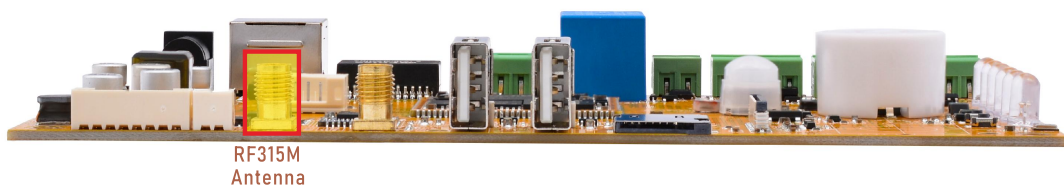
After executing the command: The relay is released, and the **normally closed (NC)** contact is connected to the **common (COM)** terminal.

- To restore the (**ON**) and (**COM**) connection (relay energized), execute the following command:

```
# echo 0 > /sys/class/leds/raley_power/brightness
```

8.8 RF315

Step 1: Connect antenna.



Step 2: Open RF315 debug print:

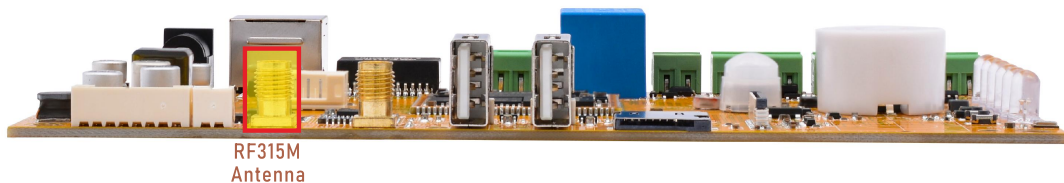
```
# echo 1 > /sys/module/rf315_decode/parameters/code_print
```

Enable debugging output to monitor received RF signals (via UART, GPIO, or /dev device as applicable).

When pressing a button near the RF receiver, the received code or key value will be printed to the system log or serial console.



Step 1: Connect antenna.



Step 2: Open RF315 debug print:

```
# echo 1 > /sys/module/rf315_decode/parameters/code_print
```

Enable debugging output to monitor received RF signals (via UART, GPIO, or /dev device as applicable).

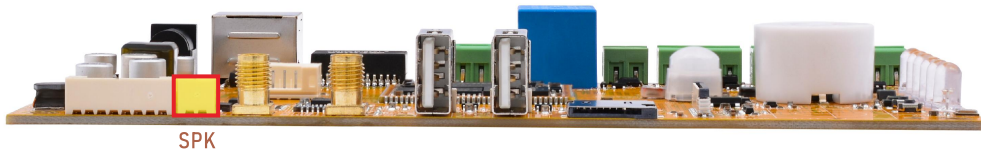
When pressing a button near the RF receiver, the received code or key value will be printed to the system log or serial console.



```
root@linaro-alip:/# echo 1 > /sys/module/rf315_decode/parameters/code_print
root@linaro-alip:/# [ 404.385220] RF315 data=0xadf481
[ 404.422044] RF315 data=0xadf481
[ 404.458973] RF315 data=0xadf481
[ 406.833506] RF315 data=0xadf482
[ 406.870369] RF315 data=0xadf482
[ 406.907226] RF315 data=0xadf482
[ 408.541557] RF315 data=0xadf488
[ 408.578451] RF315 data=0xadf488
[ 409.685817] RF315 data=0xadf484
[ 409.722696] RF315 data=0xadf484
[ 409.759572] RF315 data=0xadf484
[ 412.356096] RF315 data=0xadf488
[ 412.392938] RF315 data=0xadf488
```

8.9 Audio

Step 1: Connect the speaker.



Step 2: View sound card.

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

```
root@linaro-alip:/# cat /proc/asound/cards
0 [rockchiprv1126b]: rockchip_rv1126 - rockchip,rv1126b-acodec
  rockchip,rv1126b-acodec
1 [rockchipdmmica]: simple-card - rockchip,pdm-mic-array
```

Step 3: speaker play audio.

```
# aplay -Dhw:0,0 /mnt/udisk/video/music/2.wav
```

```
root@linaro-alip:/# aplay -Dhw:0,0 /mnt/udisk/video/music/2.wav
Playing WAVE '/mnt/udisk/video/music/2.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

Recording

Step 1: View sound card.

```
# arecord -l
```

```
root@linaro-alip:/# arecord -l
**** List of CAPTURE Hardware Devices ****
card 1: rockchipdmmica [rockchip,pdm-mic-array], device 0: 20990000.pdm-dummy_codec dummy_codec-0 [20990000.pdm-dummy_codec dummy_codec-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
root@linaro-alip:/#
```

Step 2: Use command recording.

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

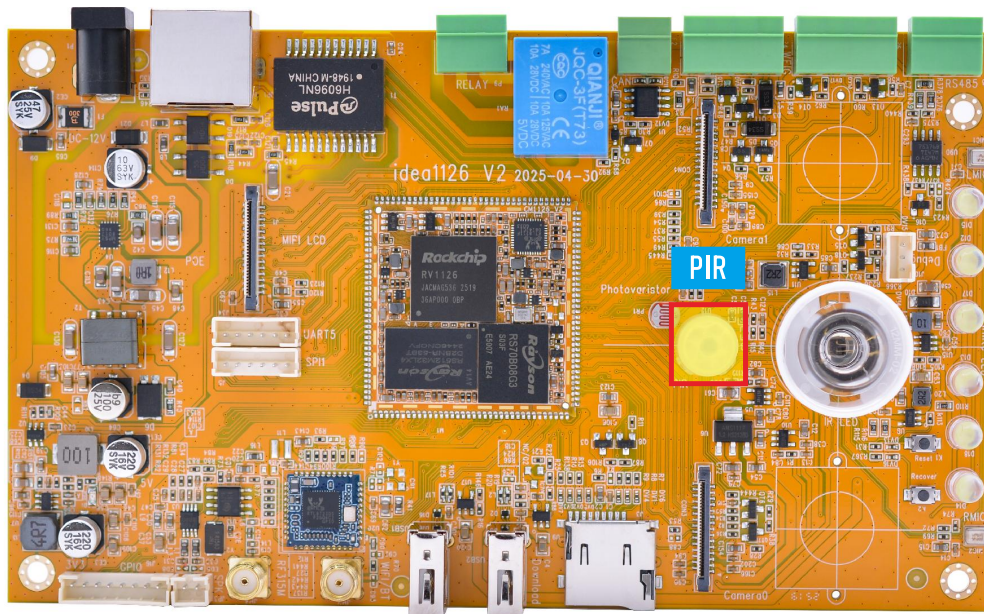
```
root@linaro-alip:/# arecord -D hw:1,0 -f s16_le -c 2 -r 16000 cap.wav
Recording WAVE 'cap.wav' : Signed 16 bit Little Endian, Rate 16000 Hz, Stereo
^CAborted by signal Interrupt...
root@linaro-alip:/#
```

Step 3: Check whether the audio file has been generated

```
# ls cap.wav
```

```
root@linaro-alip:/# ls cap.wav
cap.wav
root@linaro-alip:/#
```

8.10 PIR



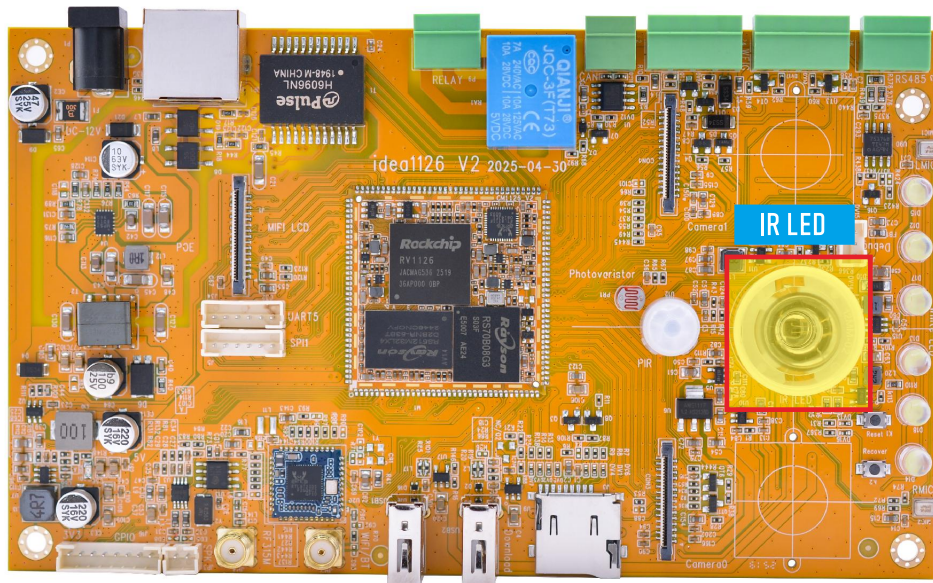
Execute the following command to test:

```
# pir_test.sh
```

When the sensor detects human movement, the output is **1**; when no human is detected, the output is **0**.

```
root@linaro-alip:/# pir_test.sh
0
1
1
1
1
0
1
1
1
1
0
1
1
1
1
1
1
1
1
1
0
```

8.11 IR



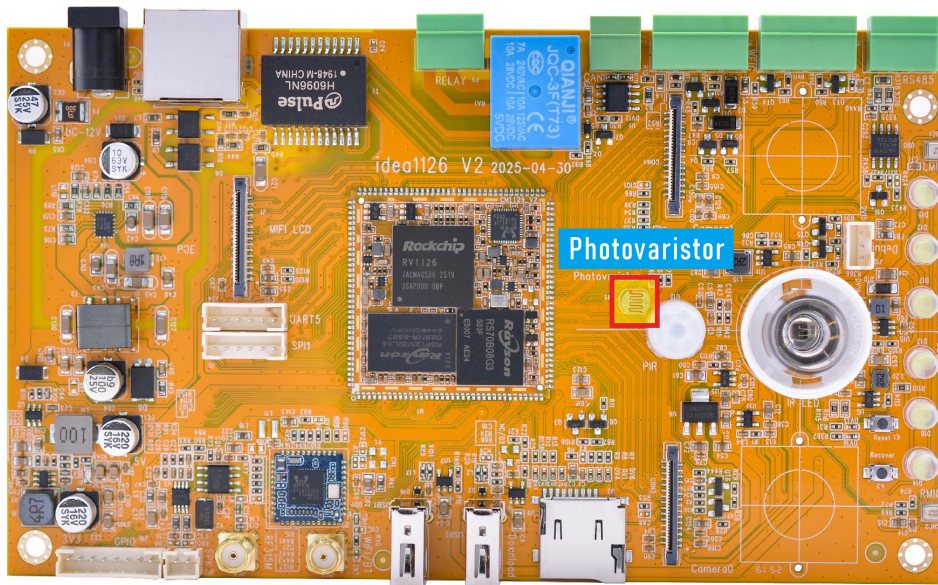
Step 1: Execute the following command to turn on the IR LED.

```
# echo 1 > /sys/class/leds/ir_led/brightness
```

Step 2: Execute the following command to turn off the IR LED.

```
# echo 0 > /sys/class/leds/ir_led/brightness
```

8.12 Photovaristor



Execute the following command to read the raw value of the photoresistor:

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

```
root@linaro-alip:/# cat /sys/bus/iio/devices/iio\:device0/in_voltage1_raw
1827
root@linaro-alip:/#
root@linaro-alip:/# cat /sys/bus/iio/devices/iio\:device0/in_voltage1_raw
6209
root@linaro-alip:/# cat /sys/bus/iio/devices/iio\:device0/in_voltage1_raw
18340
```

8.13 LED



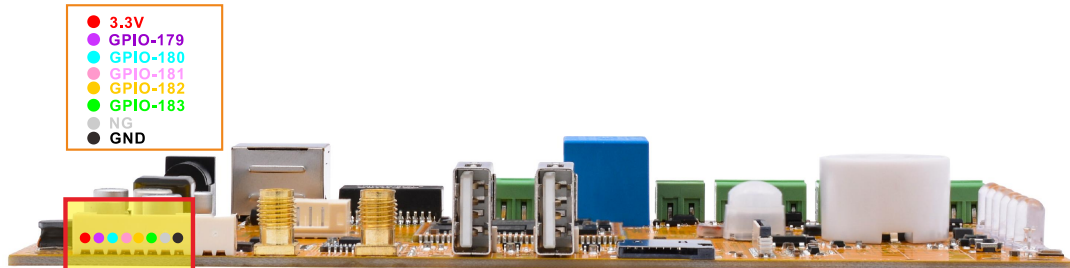
Step 1: Execute the following command to turn on the LED.

```
# echo 1 > /sys/class/leds/user_led/brightness
```

Step 2: Execute the following command to turn off the LED.

```
# echo 0 > /sys/class/leds/user_led/brightness
```

8.14 GPIO



General GPIO Control Command Guide:

```
# Export the GPIO (replace X with the actual GPIO number)
echo X > /sys/class/gpio/export

# Set the direction to output
echo out > /sys/class/gpio/gpioX/direction

# Set output to high level
echo 1 > /sys/class/gpio/gpioX/value

# Read the current GPIO level (should be 1)
cat /sys/class/gpio/gpioX/value

# Set output to low level
echo 0 > /sys/class/gpio/gpioX/value

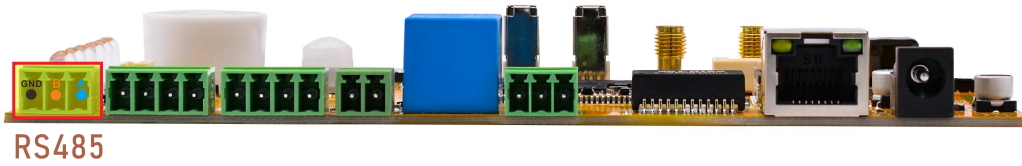
# Read the GPIO level again (should be 0)
cat /sys/class/gpio/gpioX/value
```

For example, using GPIO-179:

```
# echo 179 > /sys/class/gpio/export
# echo out > /sys/class/gpio/gpio179/direction
# echo 1 > /sys/class/gpio/gpio179/value
# cat /sys/class/gpio/gpio179/value
# echo 0 > /sys/class/gpio/gpio179/value
# cat /sys/class/gpio/gpio179/value
```

```
root@linaro-alip:/# echo 179 > /sys/class/gpio/export
root@linaro-alip:/# echo out > /sys/class/gpio/gpio179/direction
root@linaro-alip:/# echo 1 > /sys/class/gpio/gpio179/value
root@linaro-alip:/# cat /sys/class/gpio/gpio179/value
1
root@linaro-alip:/# echo 0 > /sys/class/gpio/gpio179/value
root@linaro-alip:/# cat /sys/class/gpio/gpio179/value
0
```

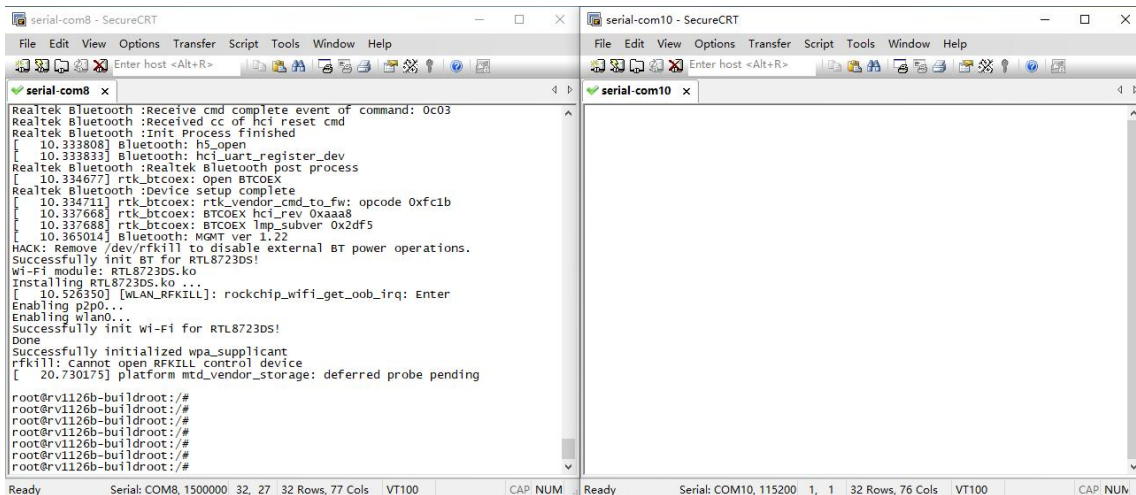
8.15 RS485



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

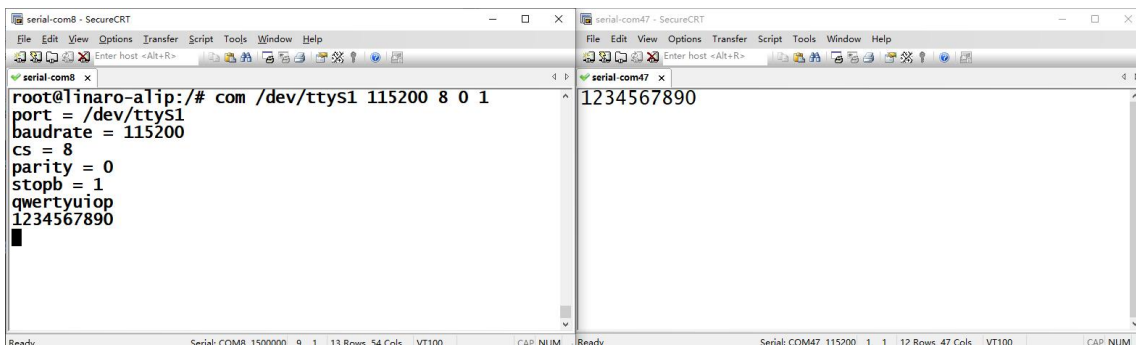


Step 2: Open the corresponding serial terminal, set the baud rate of the board to 1500000, and set the baud rate of the RS485 test tool to 115200.



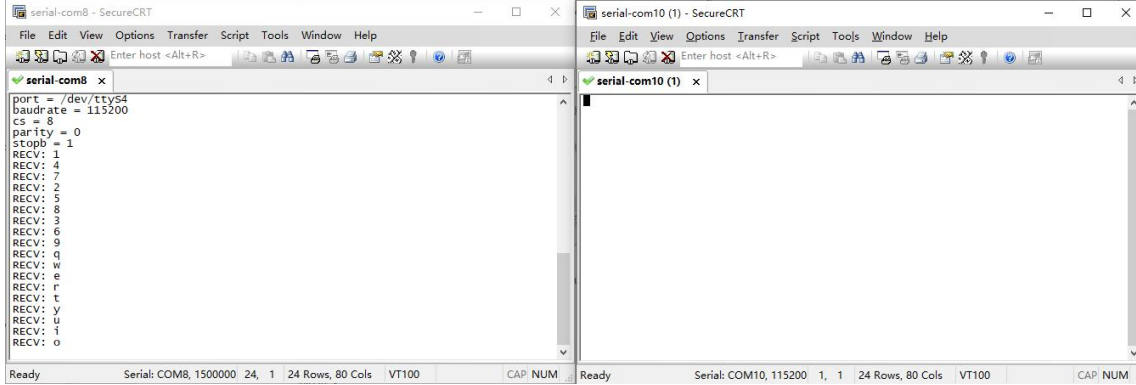
Step 3: Send data for testing

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

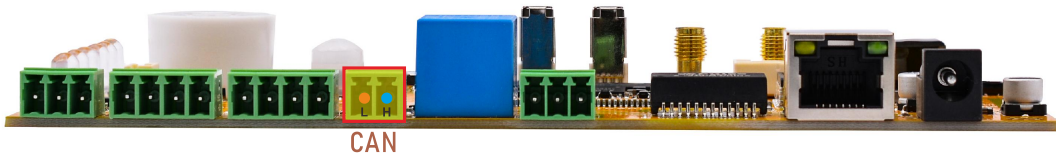


Step 4: Receiving data test.

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



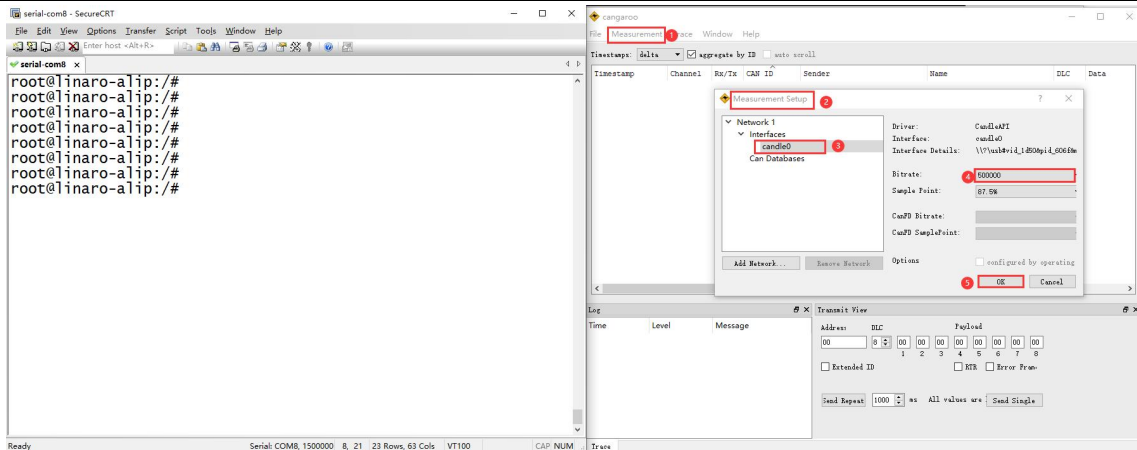
8.16 CAN



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.

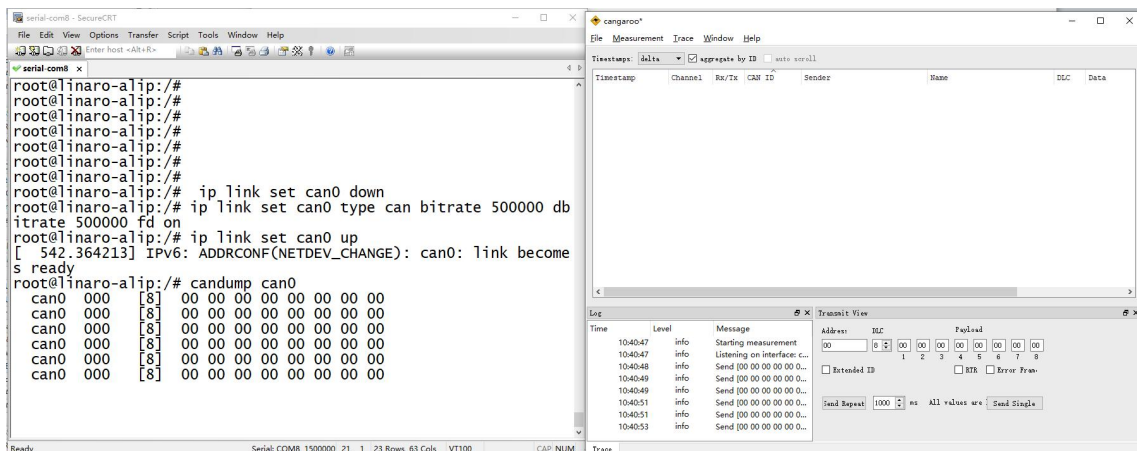


Step 3: Configure and activate the CAN network, setting the bitrate to 500000.

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

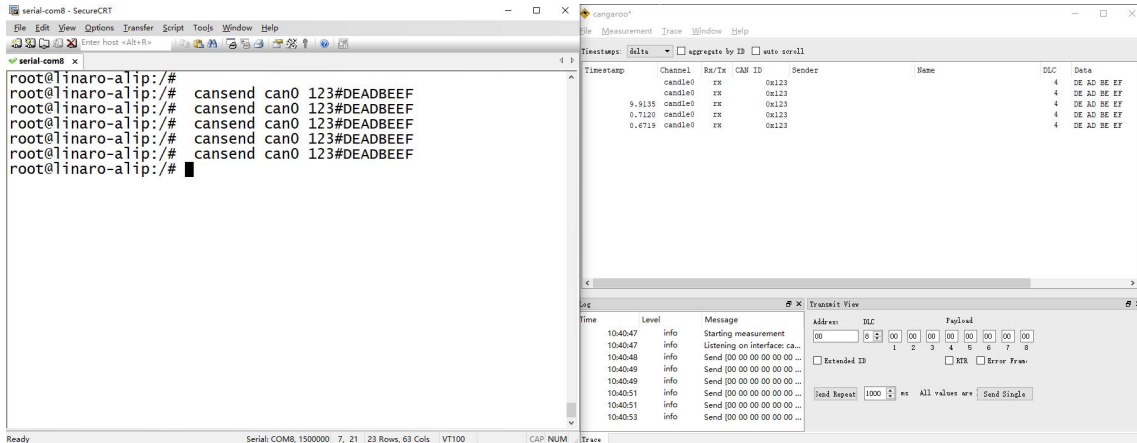
Step 4: Configure CAN as the receiver.

```
# candump can0
```



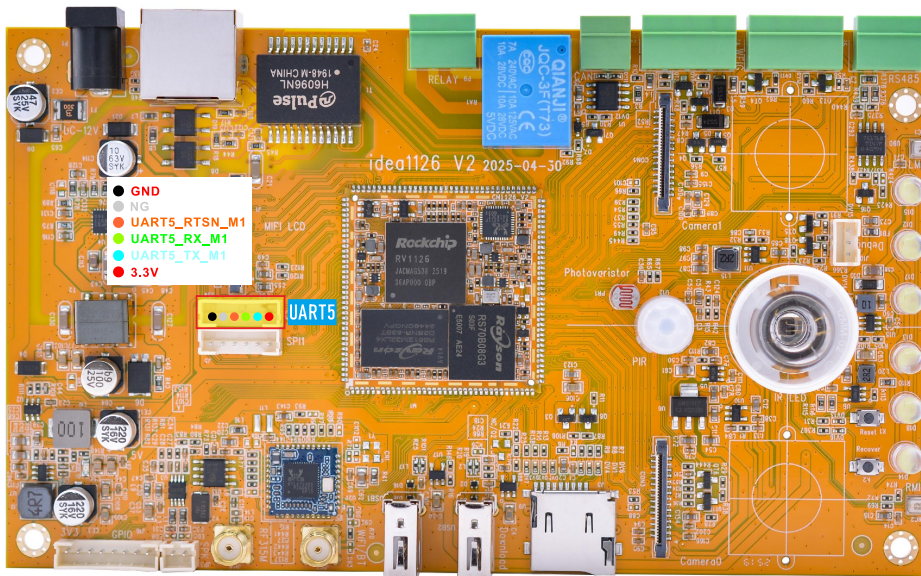
Step 5: Configure CAN as the sender.

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



8.17 UART5

Step 1: Short circuit **UART5_TX_M1** and **UART5_RX_M1** pins of UART.



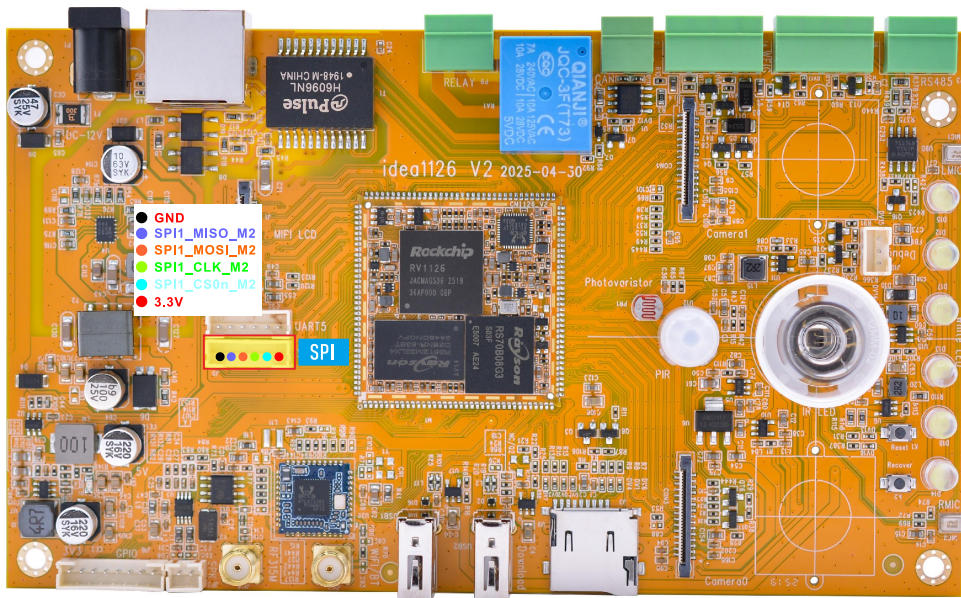
Step 2: UART test.

```
# com /dev/ttyS5 115200 8 0 1
```

```
root@linaro-alip:/# com /dev/ttyS5 115200 8 0 1
port = /dev/ttyS5
baudrate = 115200
cs = 8
parity = 0
stopb = 1
KKKKKKKKKK
RECV: KKKKKKKKKK
23232323232
RECV: 23232323232
0000000
RECV: 0000000
898UIUIUIUIU
RECV: 898UIUIUIUIU
J
RECV: J
```

8.18 SPI

Step 1: Short circuit SPI1_MOSI_M2 and SPI1_MISO_M2 pins of SPI.



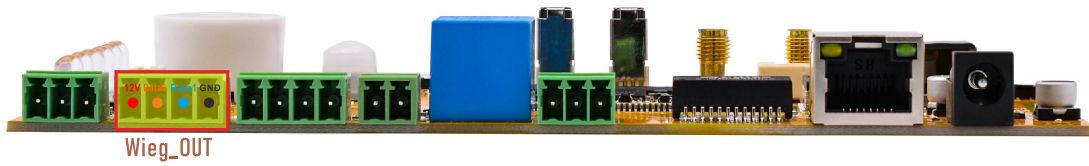
Step 2: SPI test:

```
# spidev-test -D /dev/spidev1.0
```

```
root@linaro-alip:/# spidev-test -D /dev/spidev1.0
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
```

6.19 Wiegand



Wiegand output test:

```
# wg_out_test
```

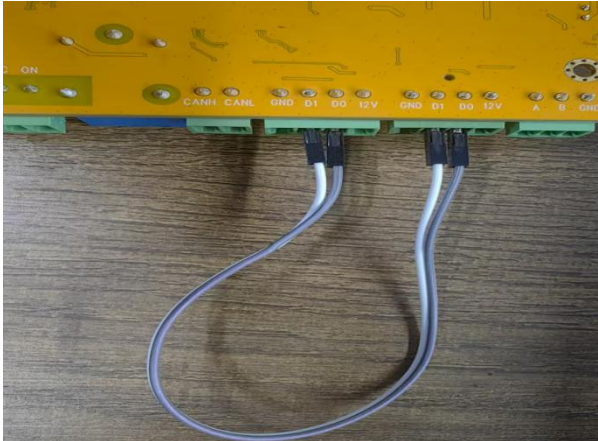
```
root@linaro-alip:/# wg_out_test
hid is 0xa26f
pid is 0x8b5c
###Setup Wiegand output 26bit or 34bit.###
[1] Wiegand output 26bit
[2] Wiegand output 34bit
[0] quit
-----Select-----
Enter your Selection: 1
[ 8835.976129] wg_output_26bit: HID:6f PID:8b5c
###Setup Wiegand output 26bit or 34bit.###
[1] Wiegand output 26bit
[2] Wiegand output 34bit
[0] quit
-----Select-----
Enter your Selection:
```

Verification: wg_output_26bit: 6f 8b5c

```
(1)Bit Grouping Structure:
1 | 01101111 | 1000101101011100 | 0
^          ^          ^
Facility  Card Number  Parity

(2)Breakdown:
P1 (Leading parity bit): 1
Facility Code: 01101111 = 0x6F
Card Number: 1000101101011100 = 0x8B5C
P2 (Trailing parity bit): 0
```

Wiegand input test:



The Wiegand output D0 and D1 interfaces are respectively connected to the Wiegand input D0 and D1 interfaces.

Open a command window using the adb shell.

```
F:\>adb shell
root@rv1126b-buildroot:/# _
```

Wiegand sending and receiving test.

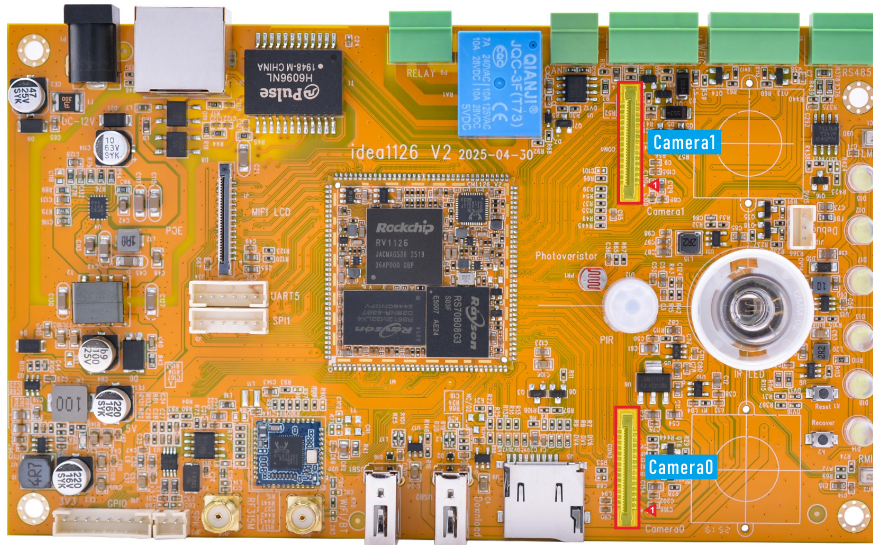
```
root@rv1126b-buildroot:/# wg_out_test
hid is 0xa26f
pid is 0x8b5c
###Setup Wiegand output 26bit or 34bit.###
[1] Wiegand output 26bit
[2] Wiegand output 34bit
[0] quit
-----Select-----
Enter your Selection: 1
[ 110.938836] wg_output_26bit: HID:6f PID:8b5c
###Setup Wiegand output 26bit or 34bit.###
[1] Wiegand output 26bit
[2] Wiegand output 34bit
[0] quit
-----Select-----
Enter your Selection: 2
[ 113.170818] wg_output_34bit: HID:a26f PID:8b5c
###Setup Wiegand output 26bit or 34bit.###
[1] Wiegand output 26bit
[2] Wiegand output 34bit
[0] quit
-----Select-----
Enter your Selection: _

root@rv1126b-buildroot:/# wg_input_test
Listening for Wiegand data on /dev/input/by-path/platform-rockchip-wg-input-event
...
(Press Ctrl+C to exit)

26-bit: HID:6f PID:8b5c Raw=0x2df16b8
34-bit: HID:a26f PID:8b5c Raw=0x344df16b9
_
```

6.20 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
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
/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
[ 84.634247] rkisp_hw 21d00000.isp: set isp clk = 500000000Hz
/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,
```

Step 3: Preview camera1.

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

```

root@linaro-alip:/# ./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
[ 126.715628] 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
[ 126.725242] rkCIF-mipi-lvds2: stream[0] start streaming
[ 126.725583] rockchip-mipi-csi2 mipi2-csi2: stream on, src_sd: 00000006d476f48,
sd_name:rockchip-csi2-dphy3
[ 126.725605] rockchip-mipi-csi2 mipi2-csi2: stream ON
[ 126.725658] rockchip-csi2-dphy3: dphy3, data_rate_mbps 892
[ 126.725686] rockchip-csi2-dphy3: csi2_dphy_s_stream stream on:1, dphy3, ret 0
[ 126.725699] imx415 1-0036: s_stream: 1. 3864x2192, hdr: 0, bpp: 10
[ 126.835043] rkCIF-mipi-lvds2: Warning: vblank need >= 1000us if isp work in online, cur 859 us
[ 126.868802] rkCIF-mipi-lvds2: Warning: vblank need >= 1000us if isp work in online, cur 859 us
Redistribute latency...
0:00:02.0 / 99:99:99.
  
```

6.21 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_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@linaro-alip:/#/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.

```

# gst-play-1.0 --videosink="waylandsink fullscreen=true"
/mt/udisk/video/4KP25/4K25FPS.mp4 --audiosink="alsasink device=hw:0,0"
  
```

Command explanation:

- /mnt/udisk/video/4KP25/4K25FPS.mp4: The media file path to be played.
- --audiosink="alsasink device=hw:0,0": Specifies the audio output device as hw:0,0.

```
root@linaro-alip:/# /mnt/udisk/video/4KP25/4K25FPS.mp4 --audiosink="alsasink device=hw:0,0"n=true"
Press 'k' to see a list of keyboard shortcuts.
Now playing /mnt/udisk/video/4KP25/4K25FPS.mp4
Redistribute latency...
Redistribute latency...
mpp[1851]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen 2025-07-01 fix[base]: Fix enc cfg size
mpp[1851]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen 2025-07-01 fix[base]: Fix enc cfg size
mpp[1851]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen 2025-07-01 fix[base]: Fix enc cfg size
mpp[1851]: mpp: unable to create enc vp8 for soc rv1126b unsupported
mpp[1851]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen 2025-07-01 fix[base]: Fix enc cfg size
mpp[1851]: mpp_info: mpp version: 1ea951af author: xiaoxu.chen 2025-07-01 fix[base]: Fix enc cfg size
Redistribute latency...
Redistribute latency...
mpp[1851]: h264d_api: is_avcC=1
mpp[1851]: mpp_buf_slot: mismatch h_stride_by_pixel 4032 - 3840
mpp[1851]: mpp_buf_slot: mismatch h_stride_by_byte 4032 - 3840
mpp[1851]: mpp_buf_slot: mismatch size_total 13063680 - 16588800
mpp[1851]: mpp_buf_slot: mismatch h_stride_by_pixel 4032 - 3840
mpp[1851]: mpp_buf_slot: mismatch h_stride_by_byte 4032 - 3840
mpp[1851]: mpp_buf_slot: mismatch size_total 13063680 - 16588800
Redistribute latency...
0:00:02.2 / 0:02:44.6
```