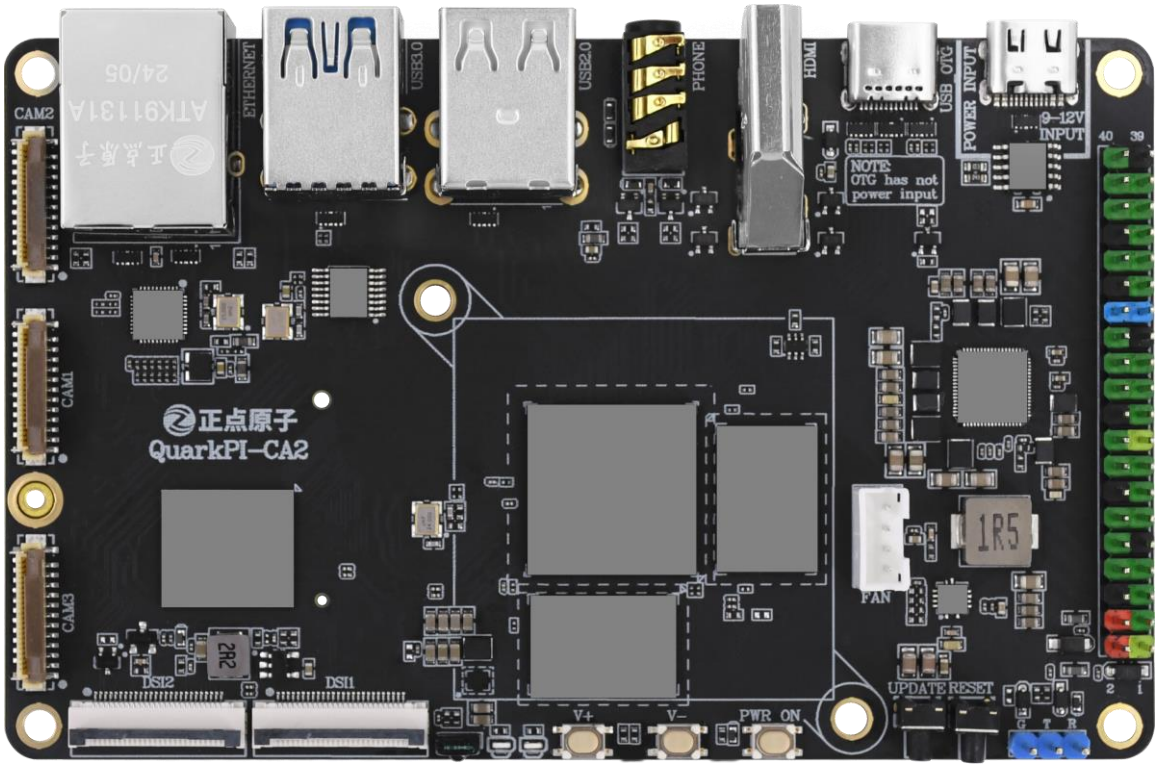
QuarkPi-CA2 Buildroot

Quick test manual

V1.0



ALIENTEK

1. Shopping:

TMALL: https://zhengdianyuanzi.tmall.com

TAOBAO: https://openedv.taobao.com

2. Download

Address: http://www.openedv.com/docs/index.html

3. FAE

Website : www.alientek.com

Forum : http://www.openedv.com/forum.php

Videos : www.yuanzige.com

Fax : +86 - 20 - 36773971

Phone : +86 - 20 - 38271790

Disclaimer

The product specifications and instructions mentioned in this document are for reference only and subject to update without prior notice; Unless otherwise agreed, this document is intended as a product guide only, and none of the representations made herein constitutes a warranty of any kind. The copyright of this document belongs to Guangzhou Xingyi Electronic Technology Co., LTD. Without the written permission of the company, any unit or individual shall not be used for profit-making purposes in any way of dissemination.

In order to get the latest version of product information, please regularly visit the download center or contact the customer service of Taobao ALIENTEK flagship store. Thank you for your tolerance and support.

Revision History:

Version	Version Update Notes	Responsible person	Proofreading	Date
V1.0	release officially	ALIENTEK	ALIENTEK	2025.03.01

Catalogue

Chapter 1. Preparations Before Using the QuarkPi-CA2 Card Computer....1
1.1 Precautions Before Powering On....2
1.2 Serial Port Software Installation....2
1.3 Connecting the Development Board to the PC....2
1.4 SSH login password....4
Chapter 2. Burning the Buildroot Linux System....5
Chapter 3. Function Testing of QuarkPi-CA2 Card Computer....6
3.1 LED Testing....7
3.2 Key Test....7
3.3 MIPI screen test....9
3.4 Touch Testing....10
3.4.1 Multi-touch Testing....10
3.4.2 MIPI backlight test....11
3.5 HDMI Test....11
3.6 DP Interface Test....14
3.7 Audio Testing....14
3.7.1 Playing Audio Test....15
3.7.2 Recording Test....15
3.7.3 Volume Control....16
3.8 Camera Test....17
3.9 USB Test....19
3.9.1 USB HOST test....20
3.9.2 USB TypeC Slave Device Test....21
3.9.3 USB TypeC Host Test....21
3.10 Network Testing....21
3.11 TF (SD) Card & eMMC Test....24
3.11.1 SD Card Test....24
3.11.2 EMMC Test....26
3.12 Main Chip Temperature....27
3.13 CPU Frequency....27
3.14 Headphone Insertion/Extraction Test....28
3.15 View NPU Information....29
3.16 PCIE WIFI & BT Interface Test....29
3.16.1 WIFI Function Test....30
3.16.2 Bluetooth Function Test....33
Chapter 4. Common Function Settings of the Buildroot System....37
4.1 Mount Point Information....38
4.2 Set Static IP....38
4.3 Creating Bootloader Program....40

Chapter 1. Preparations Before Using the QuarkPi-CA2 Card Computer

Here, we will introduce the matters that need to be noted when using the card computer. Please read carefully. Because if these matters are not paid attention to, it may lead to the failure of your experiments or damage to the development board, etc. Therefore, please be sure to pay attention.

1.1 Precautions Before Powering On

First, check if the power supply is plugged in. If it is, then check if it is providing normal power. The card computer uses the Type-C interface for power supply and supports the PD protocol. You can use a charging head that supports the PD protocol to power the card computer. Under normal circumstances, when the Type-C port of the card computer is connected to the power supply, it will automatically power on, and the two LED lights on the card computer will light up, indicating that it has been powered on.

Do not place the card computer on the cluttered workbench. It needs to be placed with a protective film under the base plate. Otherwise, it is prone to come into contact with metal foreign objects, causing short circuits between certain contacts on the card computer. When it has been used for a long time, also pay attention to observing whether any foreign objects have fallen on the card computer to prevent short circuits. Pay attention to waterproofing, moisture-proofing, dust-proofing, etc.

1.2 Serial Port Software Installation

Here, we will briefly introduce the installation of the CH340 USB serial port driver and the MobaXterm terminal. We will not provide a detailed installation tutorial as it is relatively basic.

Install the MobaXterm terminal software (or install terminal software such as Xshell, SecureCRT) in this article, using MobaXterm as an example. Network drive path: Development board network drive A - Basic Materials -> 3_softwares -> MobaXterm_Installer_v12.3.zip. Double-click to open this compressed file, wait for the decompression, and then directly double-click MobaXterm_installer_12.3.msi to install. The MobaXterm installation program will guide the installation.

Install the CH340 USB serial port driver (For the PC (computer) to communicate with the development board's serial port, we need to install this driver). On the development board's CD-ROM A - Basic Data -> 3_softwares -> CH341SER.EXE (If you have already installed it, you don't need to install it again). Find the SETUP.EXE file in the folder under the "SETUP" folder. Double-click to run it, and then click "Install" in the pop-up window. Wait for the pre-installation or the "Installation Successful" window to appear. Note: The default serial port baud rate of the development board is 1500000 (1.5M). Simply enter 1500000 directly in MobaXterm.

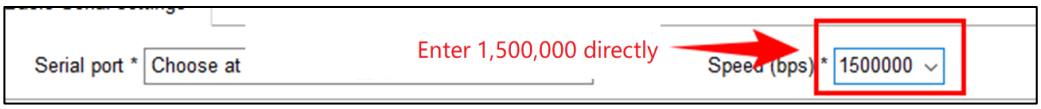


Figure 1.2-1 MoboXterm Serial Port Baud Rate Settings

1.3 Connecting the Development Board to the PC

During the development process, we exchange information with the development board through the serial port. Of course, it can also be done via the network. The network needs to know the IP address of the development board first. Therefore, the serial port is very important at this time.

The QuarkPi-CA2 card computer only provides the debugging serial port pins. You need to prepare a USB to TTL serial port module yourself and connect it according to the wiring method of GND-GND, TX-RX, RX-TX.

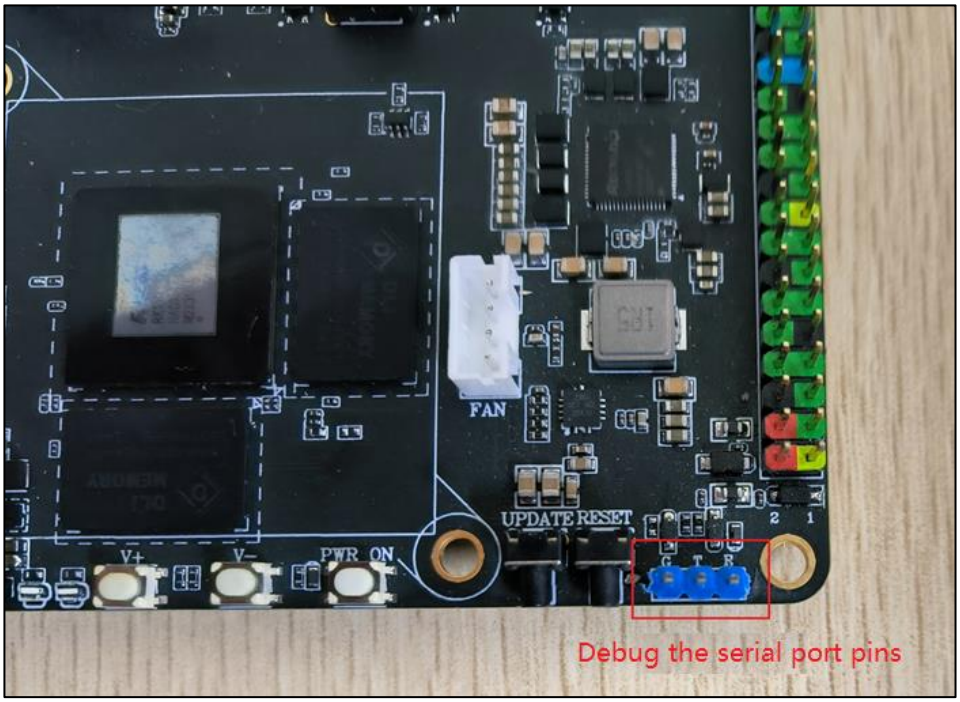


Figure 1.3-1 Card computer debugging serial port pins

The information output via the serial port is as follows (partial).

```
U-Boot 2017.09-g6ccaae8d94-230909 #alientek (Apr 23 2024 - 10:05:10 +0800)

Model: Rockchip RK3588 Evaluation Board
MPIDR: 0x81000000
PreSerial: 2, raw, 0xfeb50000
DRAM:  16 GiB
Sysmem: init
Relocation Offset: eda1c000
Relocation fdt: eb9fa680 - eb9fecc8
CR: M/C/I
Using default environment

optee api revision: 2.0
mmc@fe2c0000: 1, mmc@fe2e0000: 0
Bootdev(atags): mmc 0
MMC0: HS400 Enhanced Strobe, 200Mhz
PartType: EFI
TEEC: Waring: Could not find security partition
DM: v2
boot mode: None
RESC: 'boot', blk@0x0001945e
resource: sha256+
FIT: no signed, no conf required
   - dev=saradc, channel=7, dtb_adc=2748, read=9, found=0
   - dev=saradc, channel=7, dtb_adc=1410, read=9, found=0
   - dev=saradc, channel=7, dtb_adc=0, read=9, found=1
DTB: rk3588-atk-mipi-5p5_720x1280#_saradc_ch7=0.dtb
HASH(c): OK
I2c0 speed: 100000Hz
vsel-gpios- not found!
en-gpios- not found!
vdd_cpu_big0_s0 800000 uV
vsel-gpios- not found!
en-gpios- not found!
vdd_cpu_big1_s0 800000 uV
I2c1 speed: 100000Hz
vsel-gpios- not found!
en-gpios- not found!
vdd_npu_s0 800000 uV
spi2: RK806: 2
ON=0x40, OFF=0x00
vdd_gpu_s0 750000 uV
vdd_cpu_lit_s0 750000 uV
vdd_log_s0 750000 uV
vdd_vdenc_s0 init 750000 uV
vdd_ddr_s0 850000 uV
Device 'gpio@fd8a0000': seq 0 is in use by 'gpio@fd8a0000'
get vp0 plane mask:0x5, primary id:2, cursor_plane:-1, from dts
get vp1 plane mask:0xa, primary id:3, cursor_plane:-1, from dts
get vp2 plane mask:0x140, primary id:8, cursor_plane:-1, from dts
get vp3 plane mask:0x280, primary id:9, cursor_plane:-1, from dts
Could not find baseparameter partition
Model: ATK-DLRK3588 Board
MPIDR: 0x81000000
Rockchip UBOOT DRM driver version: v1.0.1
vp0 have layer nr:2[0 2 ], primary plane: 2
```

Figure 1.3-2 Card computer startup information

1.4 SSH login password

The SSH root user login password for the ALIENTEK QuarkPi-CA2 is root. The default login user when the computer boots is also root, and the password is also root. The default serial port for booting does not require entering a password. The root user automatically logs in. Alright, now you can proceed to the third chapter for hardware testing. The second chapter is about burning the Buildroot Linux system. If you have modified the Buildroot Linux system or damaged the system and want to restore it to the factory state, then you can follow the operations in the second chapter.

Chapter 2. Burning the Buildroot Linux System

The default system burned onto the QuarkPi-CA2 at factory is the Debian Linux system. Before conducting the Buildroot Linux test, it is necessary to burn the Buildroot Linux system onto the card computer. We have prepared a detailed document for the burning process. Please proceed to view 18, User Manual -> 03, Supplementary Documents -> XX "[ALIENTEK] QuarkPi-CA2 Factory Image Burning Guide" (XX represents an undetermined number). By burning the Buildroot Linux system, you can complete the process.

Chapter 3. Function Testing of QuarkPi-CA2 Card Computer

This chapter will introduce how to test the QuarkPi-CA2 card computer. It explains the usage and testing methods of the development board. It is very helpful for development. Note that many tests require self-provided testing tools. Please complete the testing steps according to your personal situation. The hardware has already been tested when it was manufactured, so users do not need to test again.

3.1 LED Testing

The QuarkPi-CA2 card computer of ALIENTEK is equipped with a user LED. The relationship between them and the corresponding hardware pins on the baseboard is as shown in the table below. In the factory system, we use the LED as a heartbeat indicator (can be programmed), and it is generally used to indicate whether the system is operating normally.

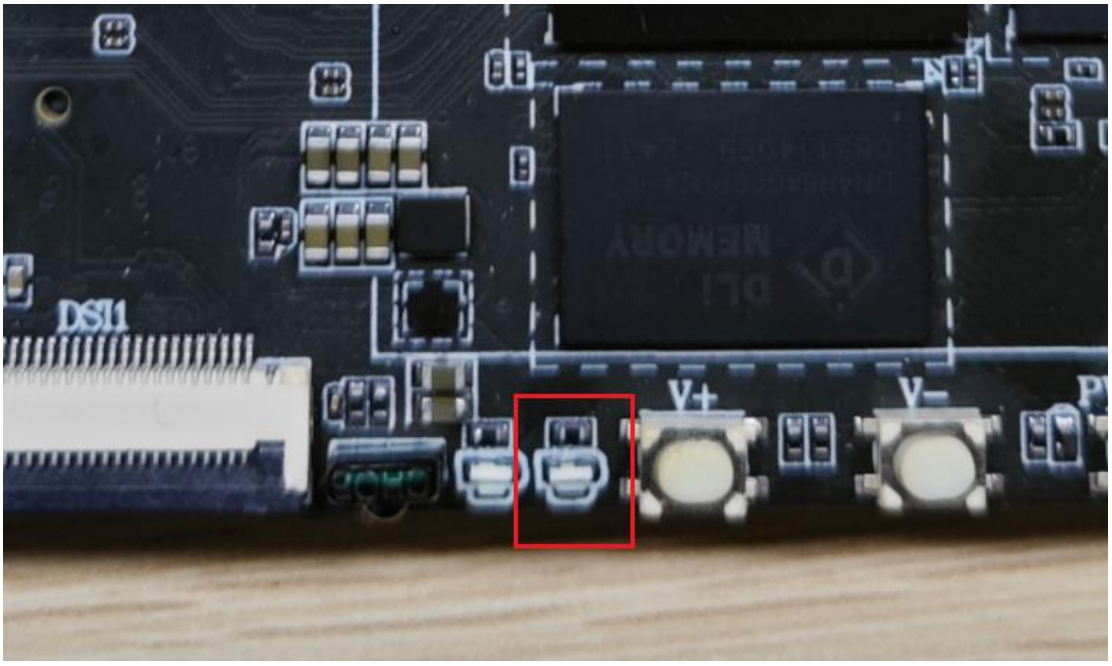


Figure 3.1-1 LED

Pin name	WORKING_LEDEN_H
QuarkPi-CA2	GPIO1_D7

Table 3.1.1 LED Pins

In the factory system, we can control the LED status by using the following instructions. After inputting, check the status of the LED on the development board.

cat /sys/class/leds/work/trigger # View the current trigger mode of LED0 and the supported trigger modes. The default mode is [heartbeat].

echo none > /sys/class/leds/work/trigger # Change the trigger mode of LED0

echo 1 > /sys/class/leds/work/brightness # Turn on LED0. At this moment, it is in the off state. The value should be a positive integer greater than 0. Usually, it is written as 1.

echo 0 > /sys/class/leds/work/brightness # Turn off LED0. At this point, it is in a constant state and its value is 0.

3.2 Key Test

The QuarkPi-CA2 card computer of ALIENTEK Corporation is equipped with 5 keys. Among them, 2 keys are connected to the first channel of the ADC. When these keys are pressed, the ADC can collect different voltage variations on the first channel.

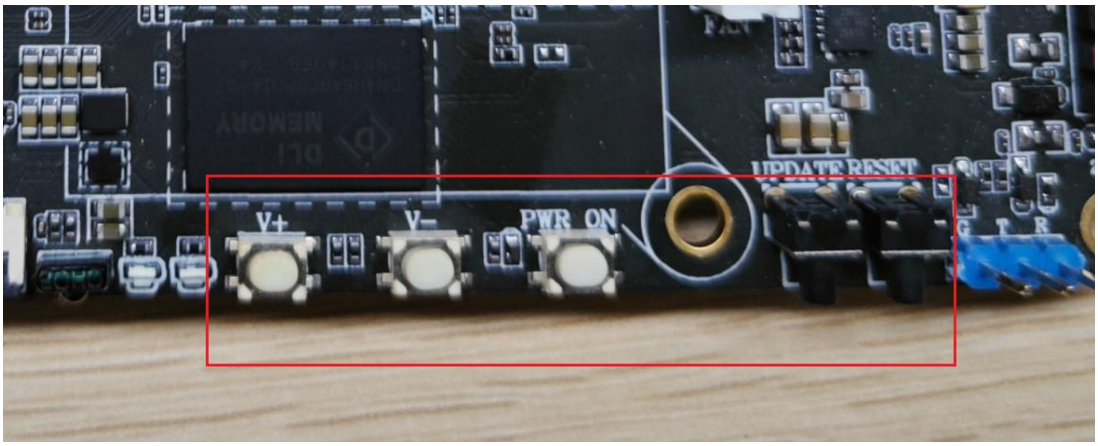


Figure 3.2-1 Card computer keys

	KEY4	KEY5
Name of the button	V+	V-
Value of ADC0	40	945

Run the following command in the serial terminal of the development board for testing, then input the number "9", as the key event is event9. To stop the test, press Ctrl + c. Please press the above four keys on the development board. Do not press any other keys, such as the reset key REST, etc.

```
evtest
```

The running result is as follows. As you can see, by pressing the corresponding key, the key type will be printed and the pressing time value will be 1, while the releasing time value will be 0.

```
root@quarkpi-ca2:/# evtest
No device specified, trying to scan all of /dev/input/event*
Available devices:
/dev/input/event0:      fd8b0030.pwm
/dev/input/event1:      rk805 pwrkey
/dev/input/event2:      rockchip-dp0 rockchip-dp0
/dev/input/event3:      headset-keys
/dev/input/event4:      rockchip-es8388 Headset
/dev/input/event5:      bt-powerkey
/dev/input/event6:      adc-keys
/dev/input/event7:      rockchip-hdmi0 rockchip-hdmi0
Select the device event number [0-7]: 6
Input driver version is 1.0.1
Input device ID: bus 0x19 vendor 0x1 product 0x1 version 0x100
Input device name: "adc-keys"
Supported events:
  Event type 0 (EV_SYN)
  Event type 1 (EV_KEY)
    Event code 114 (KEY_VOLUMEDOWN)
    Event code 115 (KEY_VOLUMEUP)
    Event code 139 (KEY_MENU)
    Event code 158 (KEY_BACK)
Properties:
Testing ... (interrupt to exit)
Event: time 908.240488, type 1 (EV_KEY), code 115 (KEY_VOLUMEUP), value 1
Event: time 908.240488, -------------- SYN_REPORT ------------
Event: time 908.343690, type 1 (EV_KEY), code 115 (KEY_VOLUMEUP), value 0
Event: time 908.343690, -------------- SYN_REPORT ------------
Event: time 909.483753, type 1 (EV_KEY), code 114 (KEY_VOLUMEDOWN), value 1
Event: time 909.483753, -------------- SYN_REPORT ------------
Event: time 909.693944, type 1 (EV_KEY), code 114 (KEY_VOLUMEDOWN), value 0
Event: time 909.693944, -------------- SYN_REPORT ------------
```

Figure 3.2-2 key prints key value

3.3 MIPI screen test

In the Buildroot system image provided by the data disk, since the screen connector of the ALIENTEK is connected to the 4th channel of the ADC of the main chip with an "ID resistor", the factory-installed system will recognize and load different screen resolutions. The MIPI screens used by ALIENTEK come in three types, as shown in the table below:

Screen size	Touch chip	ADC value
5.5-inch screen (720x1280)	gt911	40
5.5-inch screen (1080x1920)	gt911	1410
10.1-inch screen (800x1280)	gt928	2720

The default Linux images only enable the DSI1 interface. If you have purchased a MIPI screen, please connect it to the DSI1 interface. DSI2 is default to be off. Because Weston cannot use two DSI as the main screens.

The 5.5-inch 1080P MIPI screen offers a clearer display effect in reality.

3.4 Touch Testing

After the development board is powered on, we use the evtest command to check the touch events corresponding to the touch screen. The same method as in Section 3.2 can also be used to test whether the screen functions properly for touch input. Inputting the number 2, the corresponding driver for the screen is goodix-ts. Of course, clicking the Qt interface preset at the factory is also acceptable for normal touch.

3.4.1 Multi-touch Testing

The MIPI screen supports a maximum of 5 points of touch. Although Qt supports multi-touch, the controls we use are all single-point. Multi-touch requires the use of a multi-touch program. Therefore, at the factory interface, you cannot experience multi-touch. So we still use the evtest command for testing. At that time, you can use the tslib test (the default file system is not configured). Similarly, the same method as in Section 3.2 can also be used to test whether the screen functions properly. Inputting the number 2, the corresponding driver for the screen is goodix-ts.

```
evtest
```

```
root@quarkpi-ca2:/# evtest
No device specified, trying to scan all of /dev/input/event*
Available devices:
/dev/input/event0:      fd8b0030.pwm
/dev/input/event1:      rk805 pwrkey
/dev/input/event2:      dsi0_ts_gt9xx
/dev/input/event3:      rockchip-dp0 rockchip-dp0
/dev/input/event4:      headset-keys
/dev/input/event5:      rockchip-es8388 Headset
/dev/input/event6:      bt-powerkey
/dev/input/event7:      adc-keys
/dev/input/event8:      rockchip-hdmi0 rockchip-hdmi0
Select the device event number [0-8]: 2
Input driver version is 1.0.1
Input device ID: bus 0x18 vendor 0xdead product 0xbeef version 0x28bb
Input device name: "dsi0_ts_gt9xx"
Supported events:
  Event type 0 (EV_SYN)
  Event type 1 (EV_KEY)
    Event code 330 (BTN_TOUCH)
  Event type 3 (EV_ABS)
    Event code 48 (ABS_MT_TOUCH_MAJOR)
      Value      0
      Min        0
      Max      255
    Event code 50 (ABS_MT_WIDTH_MAJOR)
      Value      0
      Min        0
      Max      255
    Event code 53 (ABS_MT_POSITION_X)
      Value      0
      Min        0
      Max     1080
    Event code 54 (ABS_MT_POSITION_Y)
      Value      0
      Min        0
      Max     1920
    Event code 57 (ABS_MT_TRACKING_ID)
      Value      0
      Min        0
      Max      255
Properties:
  Property type 1 (INPUT_PROP_DIRECT)
Testing ... (interrupt to exit)
Event: time 2356.996143, type 3 (EV_ABS), code 53 (ABS_MT_POSITION_X), value 812
Event: time 2356.996143, type 3 (EV_ABS), code 54 (ABS_MT_POSITION_Y), value 424
Event: time 2356.996143, type 3 (EV_ABS), code 48 (ABS_MT_TOUCH_MAJOR), value 21
```

Figure 3.4-1 Touch test

3.4.2 MIPI backlight test

The backlight of the MIPI screen supports 255 levels of PWM adjustment. The brightness levels range from 0 to 255, with the default being 255. The larger the value, the brighter the screen.

```
cat /sys/class/backlight/backlight/max_brightness
# Check the maximum brightness level of the LCD.
cat /sys/class/backlight/backlight/brightness    # Check the current brightness level
echo 200 > /sys/class/backlight/backlight/brightness
# Modify the brightness level of the current observation screen and adjust the brightness changes.
cat /sys/class/backlight/backlight/brightness # Check the current brightness level again
```

3.5 HDMI Test

Note: Due to the wide variety of HDMI monitors available on the market, some may display slowly. This is mainly because the monitor needs initialization. However, Weston starts up faster, causing

Weston to be unable to find an output for display, resulting in a black screen. The following steps need to be followed to solve the problem. If certain special monitors cannot recognize the corresponding resolution with the development board, then you can force the corresponding resolution by using the Qt display settings of the Weston desktop based on the Buildroot system of ALIENTEK. For more information, please refer to the official RK documentation on HDMI-related settings.

```
/etc/init.d/S49weston restart        # Restart Weston
/etc/init.d/S50systemui restart      # Restart the UI
```

The default factory-installed Linux system supports HDMI. Please connect the display using an HDMI cable for testing the display. Note: The Qt interface is adapted to a resolution screen of 9:16, which means resolutions such as 720x1280, 1080x1920, and 2160x3840 are all supported. However, most displays have a 16:9 resolution, which is 1280x720, 1920x1080, and 3840x2160 (with a maximum resolution not exceeding 8K at 30fps). The default MIPI screen is the main screen. When you connect a MIPI, HDMI, or other screens, they are secondary screens. The secondary screens will display the content of the main screen, so you will see a centered UI interface. This is because MIPI is a vertical screen and HDMI is a horizontal screen, and their proportions do not match, so the mapped content will be centered.

For Weston-related configurations, please refer to Rockchip_Developer_Guide_Buildroot_Weston_CN.pdf and [ALIENTEK] Weston Desktop Qt Display Settings based on the Buildroot system.

Connect the HDMI cable to the HDMI interface shown in the figure.

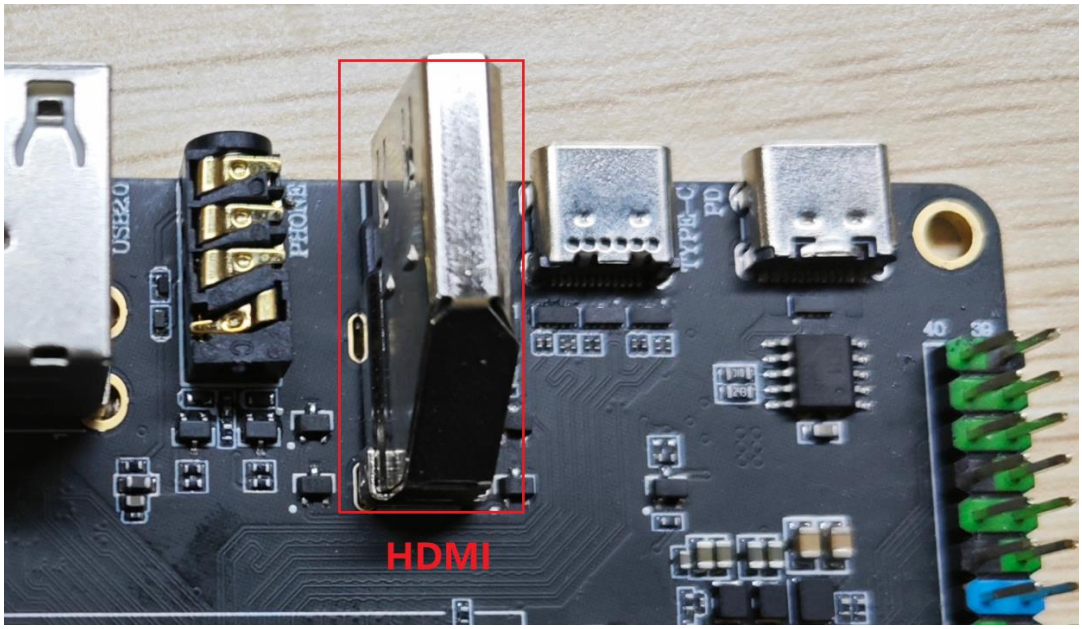


Figure 3.5-1 HDMI interface

The display result of the HDMI screen is as follows.

Figure 3.5-2 HDMI display

Note: When you only have the HDMI interface turned on, then HDMI becomes the main screen. The UI on the HDMI will cover the entire HDMI screen, causing display distortion. If you want to experience the UI functions we developed, you need to rotate the HDMI screen. Note that when HDMI is not the main screen and you rotate the HDMI, the RGA will report an error. This is a problem with the RGA driver.

```
echo "output:HDMI-A-1:rotate90" > /tmp/.weston_drm.conf    # Rotate HDMI by 90 degrees
/etc/init.d/S50systemui restart        # The UI will only adapt to the portrait mode after a restart. It is recommended that you physically rotate your screen by 90 degrees to fully experience the UI.
```

Figure 3.5-3 Rotate 90° for display

Please note: When the hot-pluggable device is connected to the main screen, it cannot be unplugged. Otherwise, Weston will fail to locate the screen output, and Weston will crash. You need to restart Weston and then restart the UI. It is also recommended that you turn on the device after connecting the HDMI cable, otherwise Weston will also fail to find a place to display.

```
/etc/init.d/S49weston restart        # Restart Weston
/etc/init.d/S50systemui restart      # Restart the UI
```

3.6 DP Interface Test

Prepare the Type-C to DP cable. Please refer to the description of the HDMI interface in the previous section. Note that the TYPE-C interface is also used for ADB and also for programming. It is recommended to test with a regular monitor. Connect the TYPE-C interface and then power on the card computer.

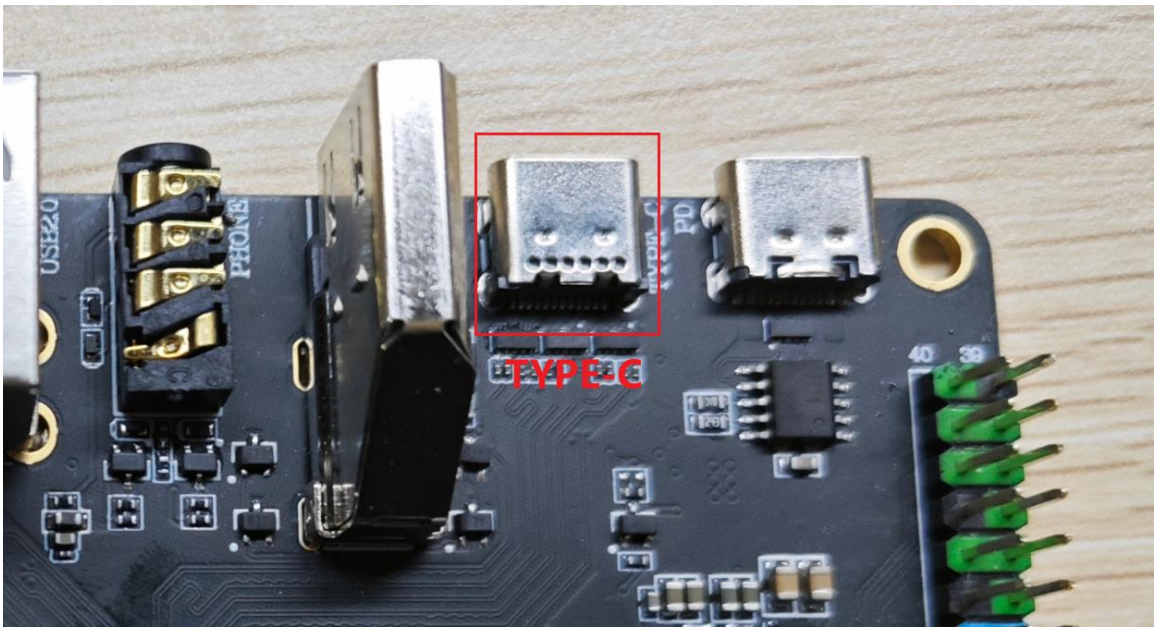


Figure 3.6-1 Full-function Type-C, supporting DP video output

3.7 Audio Testing

The QuarkPi-CA2 has a CODEC ES8388 audio chip. ES8388 is a high-performance, low-power, and low-cost stereo audio codec. It consists of 2-ch ADC, 2-ch DAC, microphone amplifier, headphone amplifier, digital audio effects, and analog mixing and gain functions. This chip uses high-order multi-bit delta-sigma modulation technology to convert data between digital and analog, making the device less sensitive to clock jitter and with lower out-of-band noise.

The card computer has an onboard 3.5mm headphone jack, but no onboard microphone or speaker. Before conducting audio tests, a headphone needs to be inserted. If conducting a recording test, an earphone with a microphone is also required.

Figure 3.7-13.5mm headphone jack

3.7.1 Playing Audio Test

The audio files can be played using system commands aplay or gst-play-1.0. Execute the following command to play the audio file. Users can also copy the audio files to the file system and use the commands aplay/gst-play-1.0 + audio file to play them.

```
aplay /usr/share/sounds/test.wav
# Use aplay to play audio files. It supports wav format but does not support mp3.
gst-play-1.0 /opt/ui/src/apps/resource/audio/*mp3
# Use gst-play-1.0 to play. It supports wav and other formats such as mp3.
```

3.7.2 Recording Test

```
aplay -l    # Here is -L, in lowercase!
```

Check the sound card number, and you will find that es8388 is the first sound card. All the places that use the number 1 are related to this.

```
root@quarkpi-ca2:/# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: rockchipdp0 [rockchip-dp0], device 0: rockchip-dp0 spdif-hifi-0 [rockchip-dp0 spdif-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: rockchipes8388 [rockchip-es8388], device 0: dailink-multicodecs ES8323.1-0011-0 [dailink-multicodecs ES8323.1-0011-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 2: rockchiphdmi0 [rockchip-hdmi0], device 0: rockchip-hdmi0 i2s-hifi-0 [rockchip-hdmi0 i2s-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
root@quarkpi-ca2:/#
```

Figure 3.7-2 The sound card list of the Buildroot system

```
amixer -c 1 cset name='PCM Volume' '192' '192'
# Set the volume of es8388 to 0 - 192. The maximum is 192. The default setting is already at the maximum, so this instruction is not necessary to execute.
arecord -D hw:1,0 -r 44100 -f S16_LE -d 10 -c 2 record.wav          # Start recording
```

Other settings do not need to be configured. The default setting is the maximum. If you need to set something, please use alsamixer and then press F6 to select the corresponding sound card, and then set the corresponding item.

Explanation of arcord command parameters:

- -c 1: Use sound card 0.
- -f S16_LE: Sample in S16_LE format.
- -r 44100: Sampling rate 44.1K
- -d 10: Recording duration 10s
- -c 2 specifies dual-channel
- record.wav: Audio file generated by the recording

To play the audio file recorded above, note: The size of the generated file is related to the format set by the user and the length of the recording. Also, due to the high volume setting during recording, the microphone did not have noise reduction, so there may be noise. Please lower the capture volume (sensitivity) and test again.

```
aplay record.wav    # play recording
```

3.7.3 Volume Control

Use input commands to adjust the volume. The volume range is from 0 to 192. The factory system is set to the maximum volume by default. Please note that in our Qt UI App, the volume setting is for the software volume within the application.

```
amixer -c 1 cset name='PCM Volume' '192' '192'
```

```
root@ATK-DLRK356X:/# amixer cset name='Master Playback Volume' '63','63'
numid=4,iface=MIXER,name='Master Playback Volume'
  ; type=INTEGER,access=rw---RW-,values=2,min=0,max=100,step=0
  : values=63,63
  | dBscale-min=-30.00dB,step=0.30dB,mute=0
root@ATK-DLRK356X:/#
```

Figure 3.7-3 Set the volume

Or you can directly use alsamixer to adjust the volume through the graphical interface. The left and right arrow keys can be used to select the item to be adjusted, and the up and down arrow keys can be used to adjust the volume. If you need to control the es83388 sound card, please press F6, select es83388, and then set the relevant items.

```
alsamixer
```

3.8 Camera Test

The QuarkPi-CA2 card computer supports MIPI CSI cameras. The camera provided by XPT6678 for this card computer currently only supports IMX415 (8 million pixels). If other cameras are supported in future updates, please consult the technical support on Taobao.

The Buildroot Linux system is pre-configured with CAM1, CAM2, and CAM3 interfaces. Please insert the camera module into any one of the three middle CAM interfaces.

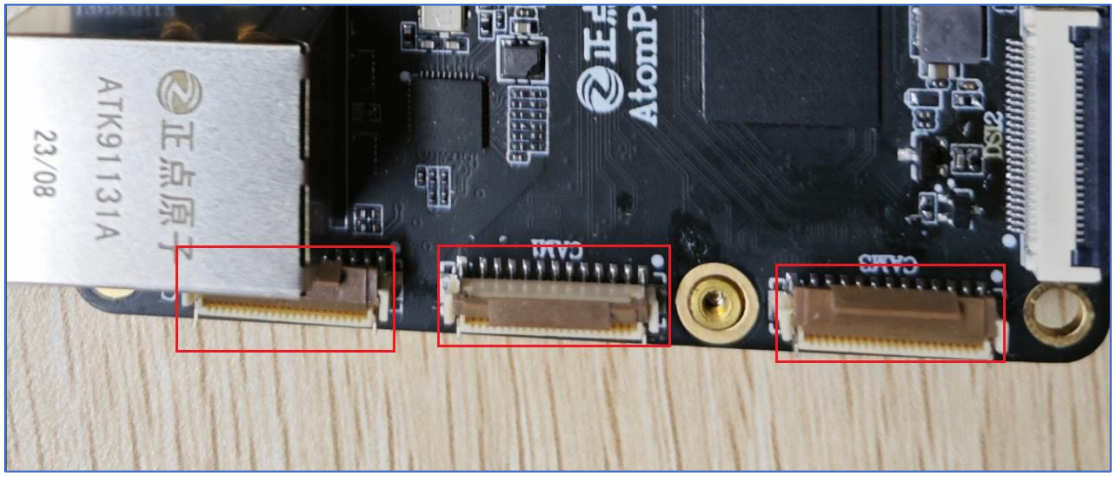


Figure 3.8-1 Camera interface

After installation, it is as shown in the following picture.

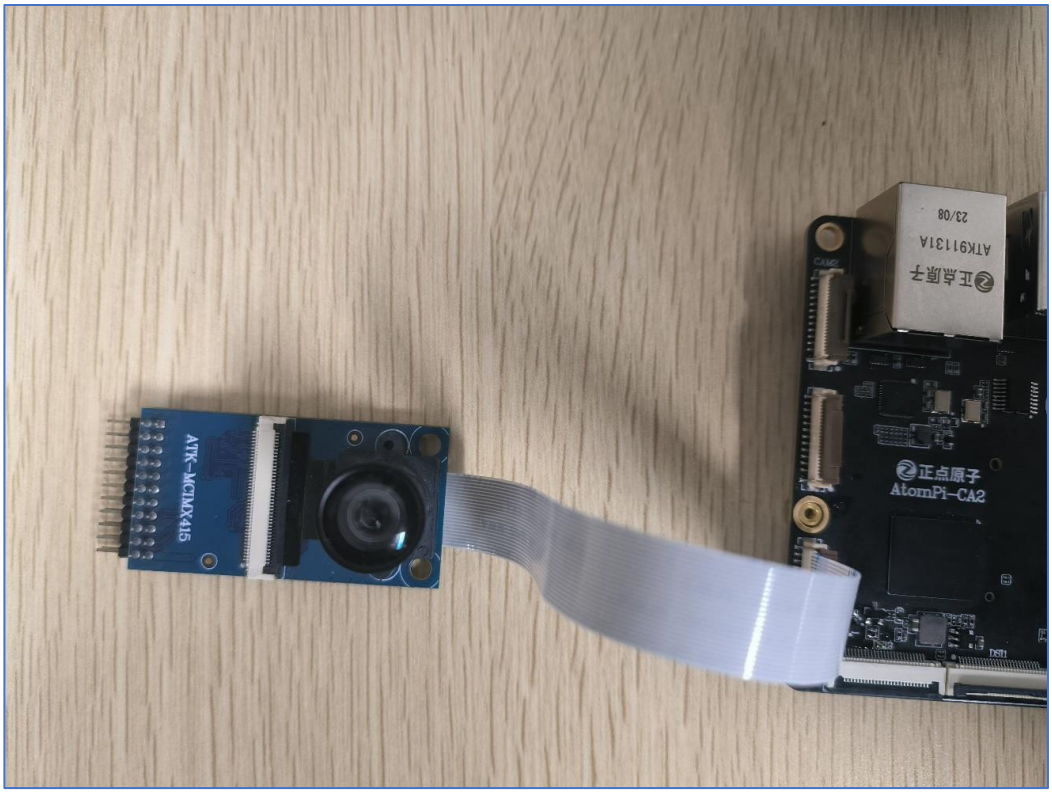


Figure 3.8-2 Connect the camera

By default, it supports simultaneous use of 3 IMX415 cameras. The correspondence between the silk-screen markings on the development board and the /dev/videoX nodes is shown in the table below:

Development board silk-screening	/dev/videoX
CAM1	/dev/video42
CAM2	/dev/video51
CAM3	/dev/video33

Figure 3.8.3 Correspondence between screen printing and video nodes

If you have a MIPI screen or have connected a monitor, then you can click on the Qt camera app developed by ALIENTEK. As shown in the figure below, the author used a MIPI screen to capture an image on the camera. The result is as follows.

Figure 3.8-3 Camera image

Please adjust according to your personal screen resolution. For example, if your resolution is 1080x1920, simply modify the "width" and "height" below accordingly.

```
gst-launch-1.0 v4l2src device=/dev/video22 ! video/x-raw,format=NV12,width=3840,height=2160,\framerate=30/1 ! autovideosink
```

If three IMX415 cameras have been connected simultaneously, the following command can be executed for testing (do not copy directly):

```
gst-launch-1.0 \
v4l2src device=/dev/video33 ! video/x-raw,format=NV12,width=640,height=480,framerate=30/1 ! \ videoconvert ! autovideosink sync=false \
v4l2src device=/dev/video42 ! video/x-raw,format=NV12,width=640,height=480,framerate=30/1 ! \ videoconvert ! autovideosink sync=false \
v4l2src device=/dev/video51 ! video/x-raw,format=NV12,width=640,height=480,framerate=30/1 ! \ videoconvert ! autovideosink sync=false
```

The correct format of the command is as shown in the following figure.

```
root@quarkpi-ca2:/# gst-launch-1.0 \
> v4l2src device=/dev/video33 ! video/x-raw,format=NV12,width=640,height=480,framerate=30/1 ! videoconvert ! autovideosink sync=false \
> v4l2src device=/dev/video42 ! video/x-raw,format=NV12,width=640,height=480,framerate=30/1 ! videoconvert ! autovideosink sync=false \
> v4l2src device=/dev/video51 ! video/x-raw,format=NV12,width=640,height=480,framerate=30/1 ! videoconvert ! autovideosink sync=false
```

3.9 USB Test

USB includes three USB 2.0 interfaces, one USB 3.0 interface and one full-function USB Type-C interface.

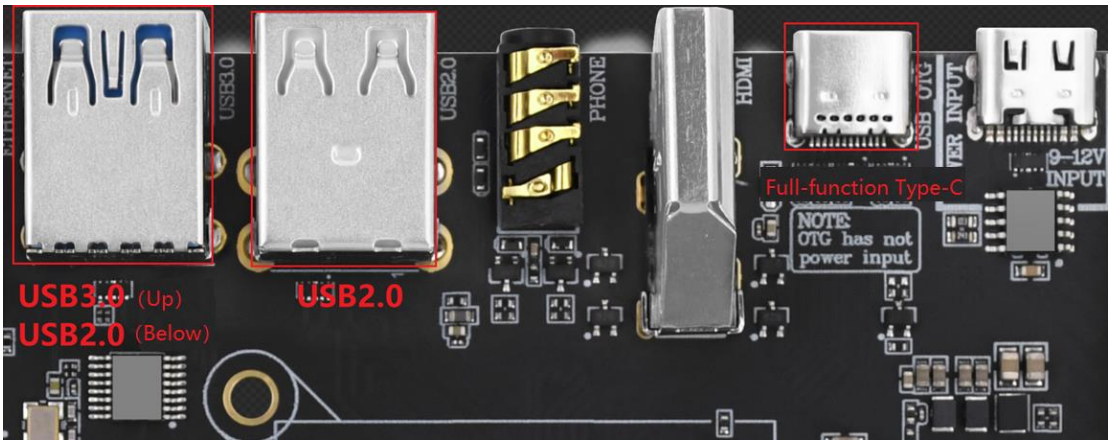


Figure 3.9-1 USB interface

3.9.1 USB HOST test

Insert a FAT32 formatted USB drive directly into the card computer QuarkPi-CA2 using a card reader (any USB interface will do).

```
root@quarkpi-ca2:/# [   38.581022] usb 1-1.3: new high-speed USB device number 3 using ehci-platform
[   38.679795] usb 1-1.3: New USB device found, idVendor=14cd, idProduct=1212, bcdDevice= 1.00
[   38.679839] usb 1-1.3: New USB device strings: Mfr=1, Product=3, SerialNumber=2
[   38.679850] usb 1-1.3: Product: Mass Storage Device
[   38.679860] usb 1-1.3: Manufacturer: Generic
[   38.679869] usb 1-1.3: SerialNumber: 121220160204
[   38.680703] usb-storage 1-1.3:1.0: USB Mass Storage device detected
[   38.682128] scsi host0: usb-storage 1-1.3:1.0
[   39.709588] scsi 0:0:0:0: Direct-Access     Mass     Storage Device   1.00 PQ: 0 ANSI: 0 CCS
[   39.904442] sd 0:0:0:0: [sda] 61952000 512-byte logical blocks: (31.7 GB/29.5 GiB)   Capacity size
[   39.905586] sd 0:0:0:0: [sda] Write Protect is off
[   39.906572] sd 0:0:0:0: [sda] No Caching mode page found
[   39.906598] sd 0:0:0:0: [sda] Assuming drive cache: write through
[   39.912309]  sda: sda1   "sda" is the device node name, and "sda1" represents the first partition.
[   39.916800] sd 0:0:0:0: [sda] Attached SCSI removable disk
[   40.163658] FAT-fs (sda1): utf8 is not a recommended IO charset for FAT filesystems, filesystem will be case sensitive!

root@quarkpi-ca2:/#
root@quarkpi-ca2:/# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        14G  1.3G   12G  10% /
devtmpfs        1.9G  8.0K  1.9G   1% /dev
tmpfs           2.0G  224K  2.0G   1% /tmp
tmpfs           2.0G  624K  2.0G   1% /run
tmpfs           2.0G  164K  2.0G   1% /var/log
tmpfs           2.0G     0  2.0G   0% /dev/shm
/dev/mmcblk0p7  124M   12M  109M  10% /oem
/dev/mmcblk0p8   15G  524K   15G   1% /userdata
/dev/sda1        30G  128K   30G   1% /mnt/udisk   The sda1 partition has been automatically mounted to /mnt
root@quarkpi-ca2:/#
```

We can directly access the /mnt/udisk directory to perform read and write file operations. Note: The test of read/write speed is related to the maximum read/write speed of your personal card reader or U disk/TF card.

Read speed test:

```
hdparm -t /dev/sda1
```

Speed writing test:

```
time dd if=/dev/zero of=/mnt/udisk/test bs=1024k count=100 conv=fdatasync
rm /mnt/udisk/test
```

Here, it represents the writing of a 100 MiB file, with a writing speed of 10.8 MB/s. In fact, the larger the file written, the closer the average writing speed is to the actual value.

```
root@quarkpi-ca2:/# time dd if=/dev/zero of=/mnt/udisk/test bs=1024k count=100 conv=fdatasync
100+0 records in
100+0 records out
104857600 bytes (105 MB, 100 MiB) copied, 10.4239 s, 10.1 MB/s

real    0m10.432s
user    0m0.000s
sys     0m0.383s
root@quarkpi-ca2:/#
```

3.9.2 USB TypeC Slave Device Test

The full-function interface USB TypeC interface (USB TypeC0) on the QuarkPi-CA2 card computer can use the ADB function and acts as a slave device. Please refer to the ADB usage manual written by SparkFun.

3.9.3 USB TypeC Host Test

Connect a USB to TypeC adapter (if the user doesn't have this adapter, this test can be skipped) to the USB drive and plug it into the USB TypeC0/USB TypeC1 interface. The test can be conducted as described in Section 3.10.1.

The USB to TypeC adapter is as follows. Please purchase this adapter as needed by yourself.

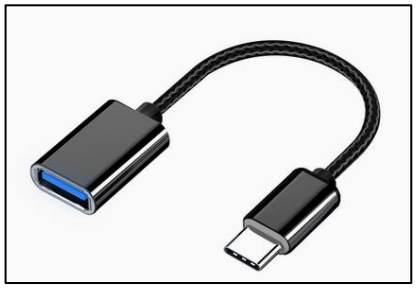


Figure 3.9-2 TYPE-C to USB adapter cable

3.10 Network Testing

The QuarkPi-CA2 is equipped with a gigabit network chip, which is capable of adapting to 10/100/1000M. The following commands can be used to test the network in the file system.

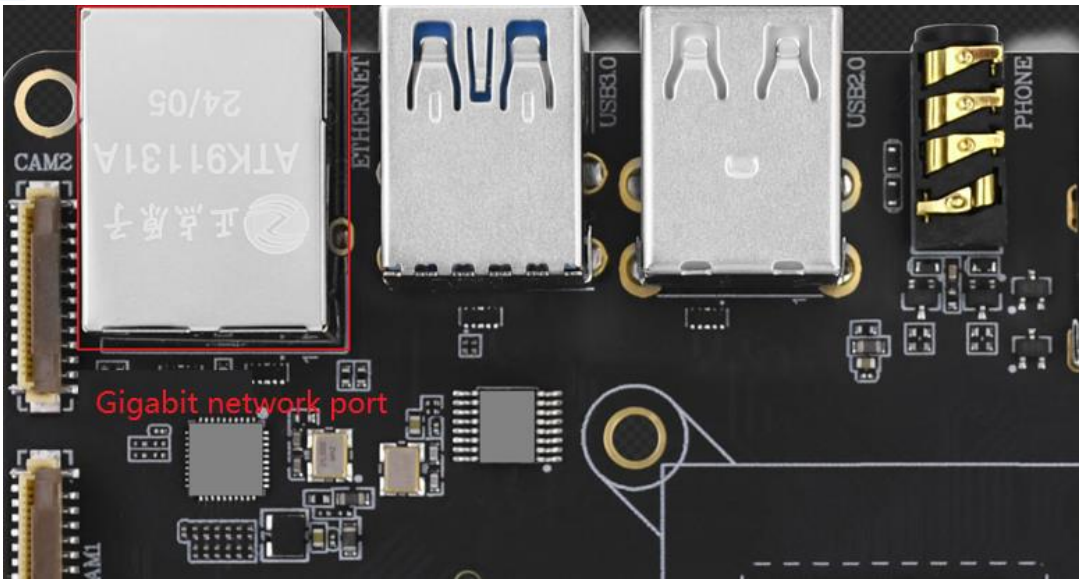


Figure 3.10-1 Built-in Gigabit Ethernet port

Plug the network cable into one of the two ports and ensure that the network can access the internet. Check the IP address obtained from the network.

```
ifconfig
```

```
root@quarkpi-ca2:/# ifconfig
eth0      Link encap:Ethernet  HWaddr 0E:8A:11:91:AB:3A
          inet addr:192.168.6.177  Bcast:192.168.6.255  Mask:255.255.255.0
          inet6 addr: fe80::c8a:11ff:fe91:ab3a/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:540 errors:0 dropped:0 overruns:0 frame:0
          TX packets:54 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:38050 (37.1 KiB)  TX bytes:5521 (5.3 KiB)
          Interrupt:99

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:350 errors:0 dropped:0 overruns:0 frame:0
          TX packets:350 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:22246 (21.7 KiB)  TX bytes:22246 (21.7 KiB)

root@quarkpi-ca2:/#
```

Figure 3.10-2 Automatically obtain IP address

Check if the network can access the internet. If there is a data response, it indicates that the network function is normal. Press Ctrl+c to terminate the instruction.

```
ping www.baidu.com
```

```
root@quarkpi-ca2:/# ping www.baidu.com
PING www.a.shifen.com (183.2.172.42) 56(84) bytes of data.
64 bytes from 183.2.172.42 (183.2.172.42): icmp_seq=1 ttl=52 time=4.16 ms
64 bytes from 183.2.172.42 (183.2.172.42): icmp_seq=2 ttl=52 time=4.17 ms
64 bytes from 183.2.172.42 (183.2.172.42): icmp_seq=3 ttl=52 time=4.08 ms
64 bytes from 183.2.172.42 (183.2.172.42): icmp_seq=4 ttl=52 time=4.07 ms
64 bytes from 183.2.172.42 (183.2.172.42): icmp_seq=5 ttl=52 time=4.03 ms
64 bytes from 183.2.172.42 (183.2.172.42): icmp_seq=6 ttl=52 time=4.14 ms
64 bytes from 183.2.172.42 (183.2.172.42): icmp_seq=7 ttl=52 time=4.39 ms
64 bytes from 183.2.172.42 (183.2.172.42): icmp_seq=8 ttl=52 time=4.48 ms
64 bytes from 183.2.172.42 (183.2.172.42): icmp_seq=9 ttl=52 time=4.29 ms
^C
--- www.a.shifen.com ping statistics ---
9 packets transmitted, 9 received, 0% packet loss, time 8012ms
rtt min/avg/max/mdev = 4.026/4.200/4.477/0.143 ms
root@quarkpi-ca2:/#
```

At the same time, the other network port was also tested in the same way. Plug the network cable into the other port and repeat the above operation.

Test whether it is a gigabit network. Pay attention to using a gigabit network cable, a gigabit router (most home routers are of the hundred-megabit type) or a gigabit switch. The PC (computer) network card must be a gigabit network card. Otherwise, the speed measured or identified may be hundred-megabit! As shown in the figure below, seeing "1Gbps/Full" indicates that it is a gigabit network.

```
[  382.777952] rk_gmac-dwmac fe1c0000.ethernet eth0: Link is Up - 1Gbps/Full - flow control rx/tx
[  382.778086] IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready

root@quarkpi-ca2:/#
```

The iperf3 command can be used to test the connection speed of a gigabit network.

Set Ubuntu as the server. Ubuntu and the development board must be in the same local area network. If Ubuntu does not have the iperf3 command, you need to enter "sudo apt-get install iperf3" to install it.

View Ubuntu IP

```
ifconfig
```

```
lixinkai@lixinkai-virtual-machine:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.6.242  netmask 255.255.255.0  broadcast 192.168.6.255
        inet6 fe80::ca29:f1ee:fb9c:bfda  prefixlen 64  scopeid 0x20<link>
        ether 00:50:56:28:53:b9  txqueuelen 1000  (以太网)
        RX packets 489645  bytes 36512330 (36.5 MB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 2119  bytes 275301 (275.3 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING>  mtu 65536
        inet 127.0.0.1  netmask 255.0.0.0
        inet6 ::1  prefixlen 128  scopeid 0x10<host>
        loop  txqueuelen 1000  (本地环回)
        RX packets 238  bytes 25722 (25.7 KB)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 238  bytes 25722 (25.7 KB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0

lixinkai@lixinkai-virtual-machine:~$
```

Enter "iperf3 -s" to set Ubuntu as the server.

```
iperf3  -s
```

```
lixinkai@lixinkai-virtual-machine:~$ iperf3 -s
-----------------------------------------------------------
Server listening on 5201
-----------------------------------------------------------
```

QuarkPi-CA2 is the client. By using the iperf3 command, simply input the IP address of the server (Ubuntu) and it will work.

```
iperf3 -c 192.168.6.45 -i 1
```

Command parsing:

- -s: Indicates that the current device acts as a server.
- -c: Indicates that the current device acts as a client.
- 192.168.6.242: The IP address of the server.
- -i: Trigger cycle.

```
root@quarkpi-ca2:/# iperf3 -c 192.168.6.242 -i 1
Connecting to host 192.168.6.242, port 5201
[  5] local 192.168.6.177 port 39410 connected to 192.168.6.242 port 5201
[ ID] Interval           Transfer     Bitrate         Retr  Cwnd
[  5]   0.00-1.00   sec   113 MBytes   949 Mbits/sec    0    602 KBytes
[  5]   1.00-2.00   sec   112 MBytes   942 Mbits/sec    0    602 KBytes
[  5]   2.00-3.00   sec   112 MBytes   941 Mbits/sec    0    602 KBytes
[  5]   3.00-4.00   sec   112 MBytes   944 Mbits/sec    0    602 KBytes
[  5]   4.00-5.00   sec   112 MBytes   941 Mbits/sec    0    632 KBytes
[  5]   5.00-6.00   sec   112 MBytes   944 Mbits/sec    0    697 KBytes
[  5]   6.00-7.00   sec   107 MBytes   900 Mbits/sec    0    745 KBytes
[  5]   7.00-8.00   sec   111 MBytes   934 Mbits/sec    0    786 KBytes
[  5]   8.00-9.00   sec   112 MBytes   942 Mbits/sec    0    786 KBytes
[  5]   9.00-10.00  sec   112 MBytes   943 Mbits/sec    0    926 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bitrate         Retr
[  5]   0.00-10.00  sec  1.09 GBytes   938 Mbits/sec    0             sender
[  5]   0.00-10.05  sec  1.09 GBytes   932 Mbits/sec                  receiver

iperf Done.
root@quarkpi-ca2:/#
```

The Bitrate received and transmitted during the test was approximately 900 Mbits/sec, which is the speed of a gigabit network (please do not run other CPU-consuming applications before the test to avoid affecting the test speed). If it is not so, please check whether a gigabit network cable is used, whether the network card of the PC is a gigabit card, whether the router is gigabit, etc. No device with a hundred-megabit speed should be used; otherwise, the result will be a hundred-megabit network.

3.11 TF (SD) Card & eMMC Test

Instruction Prompt:

The time command is often used to measure the running time of a command. dd is used for copying. It reads from the if (input file) file and writes to the of (output file) specified file. bs is the size of each write block, and count is the number of read/write blocks. "if=/dev/zero" does not generate IO, that is, it can continuously output data, so it can be used to test pure write speed.

3.11.1 SD Card Test

After the development board is powered on, insert the TF card into the card slot on the bottom plate of the development board. The following information will be printed. The development board will automatically mount the TF card. (Note: A FAT32 formatted TF card is required.)

Use the command "df -h" to check the partition mounted by the SD card. The result is as follows:

```
root@quarkpi-ca2:/# [ 1187.695129] dwmmc_rockchip fe2c0000.mmc: could not set regulator OCR (-22)
[ 1187.695169] dwmmc_rockchip fe2c0000.mmc: failed to enable vmmc regulator
[ 1187.919388] mmc_host mmc1: Bus speed (slot 0) = 148500000Hz (slot req 150000000Hz, actual 148500000HZ div = 0)
[ 1187.926970] dwmmc_rockchip fe2c0000.mmc: Successfully tuned phase to 225
[ 1187.927008] mmc1: new ultra high speed SDR104 SDHC card at address 59b4
[ 1187.928164] mmcblk1: mmc1:59b4 LX32G 29.5 GiB
[ 1187.929717]  mmcblk1: p1
[ 1188.020536] FAT-fs (mmcblk1p1): utf8 is not a recommended IO charset for FAT filesystems, filesystem will be case sensitive!
[ 1188.023482] FAT-fs (mmcblk1p1): Volume was not properly unmounted. Some data may be corrupt. Please run fsck.

root@quarkpi-ca2:/#
root@quarkpi-ca2:/# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        14G  1.3G   12G  10% /
devtmpfs        1.9G  8.0K  1.9G   1% /dev
tmpfs           2.0G  204K  2.0G   1% /tmp
tmpfs           2.0G  616K  2.0G   1% /run
tmpfs           2.0G  168K  2.0G   1% /var/log
tmpfs           2.0G     0  2.0G   0% /dev/shm
/dev/mmcblk0p7  124M   12M  109M  10% /oem
/dev/mmcblk0p8   15G  556K   15G   1% /userdata
/dev/mmcblk1p1   30G  101M   30G   1% /mnt/sdcard
root@quarkpi-ca2:/#
```

Read speed test is often used to test the reading speed of hard drives. The commands for this are hdparm and dd. In this demonstration, we only show the reading speed of the TF card using hdparm. The data sampling time is too short and the data volume is small. The speed provided here is for reference only.

```
hdparm -t /dev/mmcblk1p1
```

```
root@quarkpi-ca2:/# hdparm -t /dev/mmcblk1p1

/dev/mmcblk1p1:
Timing buffered disk reads:  201 MB in 3.00 seconds = 68538 kB/s
root@quarkpi-ca2:/#
```

It can be seen that the reading speed is 68538 kB/s.

Write speed test

```
time dd if=/dev/zero of=/mnt/sdcard/test bs=1024k count=50 conv=fdatasync
```

```
root@quarkpi-ca2:/# time dd if=/dev/zero of=/mnt/sdcard/test bs=1024k count=50 conv=fdatasync
50+0 records in
50+0 records out
52428800 bytes (52 MB, 50 MiB) copied, 1.67207 s, 31.4 MB/s

real    0m1.718s
user    0m0.001s
sys     0m0.128s
root@quarkpi-ca2:/#
```

Use the "dd" command for reading. Due to the Linux system mechanism, generally there is no need to deliberately release the already used cache. These cache contents can increase the reading and writing speed of the file.

Execute the following command to clear the cache.

```
echo 3 > /proc/sys/vm/drop_caches                    # clear cache
time dd if=/mnt/sdcard/test of=/dev/null bs=1024k    # Read speed test
rm /mnt/sdcard/test                                  # Delete this file after reading it.
```

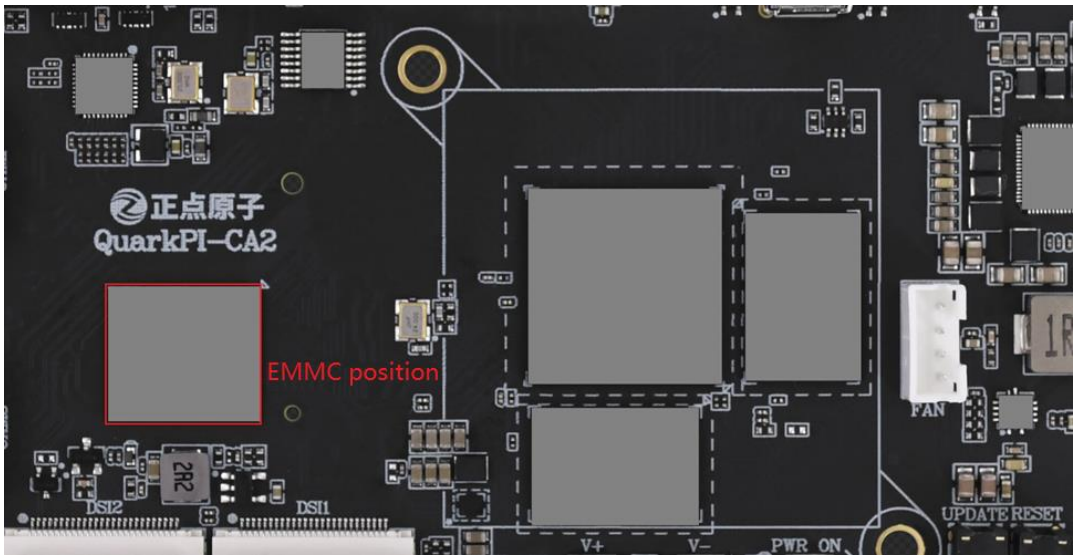
```
root@quarkpi-ca2:/# echo 3 > /proc/sys/vm/drop_caches
[ 1305.485735] sh (844): drop_caches: 3
root@quarkpi-ca2:/# time dd if=/mnt/sdcard/test of=/dev/null bs=1024k
50+0 records in
50+0 records out
52428800 bytes (52 MB, 50 MiB) copied, 0.798219 s, 65.7 MB/s

real    0m0.812s
user    0m0.003s
sys     0m0.136s
root@quarkpi-ca2:/# rm /mnt/sdcard3/test
```

It can be seen that the reading speed is 66.1MB/s. This speed is for reference only. The larger the file being read, the more accurate the speed will be.

3.11.2 EMMC Test

As a storage device, eMMC stores the Linux system. In the future, all the applications developed by users will be stored in this device (if you do not connect an external storage device).



By using the df -h command, we can determine the partitions of the eMMC. The running result is as follows. In fact, the eMMC storage is divided into many partitions, and the one mounted under the root directory "/ " is also an eMMC partition. Here, we take the partition directory mmcblk0p8 mounted as /userdata as an example.

```
df
```

```
root@quarkpi-ca2:/# df
Filesystem      1K-blocks    Used Available Use% Mounted on
/dev/root        14424952 1265208  12544440  10% /
devtmpfs          1991064       8   1991056   1% /dev
tmpfs             1998584     204   1998380   1% /tmp
tmpfs             1998584     608   1997976   1% /run
tmpfs             1998584     168   1998416   1% /var/log
tmpfs             1998584       0   1998584   0% /dev/shm
/dev/mmcblk0p7     126376   11952    110888  10% /oem
/dev/mmcblk0p8   15236084     556  15218924   1% /userdata
root@quarkpi-ca2:/#
```

Read speed test

```
hdparm -t /dev/mmcblk0p8
```

```
root@quarkpi-ca2:/# hdparm -t /dev/mmcblk0p8

/dev/mmcblk0p8:
Timing buffered disk reads:  914 MB in 3.00 seconds = 311727 kB/s
root@quarkpi-ca2:/#
```

It can be seen that the read speed is 311727 KB/s.

Write speed test

```
time dd if=/dev/zero of=/userdata/test bs=1024k count=50 conv=fdatasync
```

```
root@quarkpi-ca2:/# time dd if=/dev/zero of=/userdata/test bs=1024k count=50 conv=fdatasync
50+0 records in
50+0 records out
52428800 bytes (52 MB, 50 MiB) copied, 0.323412 s, 162 MB/s

real    0m0.333s
user    0m0.000s
sys     0m0.089s
root@quarkpi-ca2:/#
```

It can be seen that the writing speed is 162MB/s.

Use the dd command for reading. Due to the mechanism of the Linux system, generally there is no need to specifically release the used cache. These cache contents can increase the reading and writing speed of the file.

Execute the following command to clear the cache.

```
echo 3 > /proc/sys/vm/drop_caches                # clear cache
time dd if=/userdata/test of=/dev/null bs=1024k  # Read speed test
rm /userdata/test                                # Delete this file after reading it.
```

```
root@quarkpi-ca2:/# echo 3 > /proc/sys/vm/drop_caches
[ 1468.153892] sh (844): drop_caches: 3
root@quarkpi-ca2:/# time dd if=/userdata/test of=/dev/null bs=1024k
50+0 records in
50+0 records out
52428800 bytes (52 MB, 50 MiB) copied, 0.177448 s, 295 MB/s

real    0m0.187s
user    0m0.005s
sys     0m0.058s
root@quarkpi-ca2:/# rm /userdata/test
root@quarkpi-ca2:/#
```

It can be seen that the reading speed is 295MB/s. The file being read is relatively small, and this speed is for reference only.

3.12 Main Chip Temperature

Use the cat command to check the temperature of the main chip. The user layer provides an interface, and the displayed value unit is millikelvin, as shown in the following figure.

```
cat /sys/class/thermal/thermal_zone0/temp    # Check the temperature of the main chip
```

```
root@quarkpi-ca2:/# cat /sys/class/thermal/thermal_zone0/temp
36076
root@quarkpi-ca2:/#
```

36076 represents 36.076 degrees.

3.13 CPU Frequency

The command is as follows:

View the current running CPU frequency.

```
cat /sys/devices/system/cpu/cpu*/cpufreq/cpuinfo_cur_freq
```

Check the last set CPU frequency on the software.

```
cat /sys/devices/system/cpu/cpu*/cpufreq/scaling_cur_freq
```

Check the supported CPU frequencies

```
cat /sys/devices/system/cpu/cpufreq/policy0/scaling_available_frequencies
```

Enable the CPU performance mode, run at the highest frequency, for this adjustment, the small core. Note: For testing 8K video playback, it is recommended to enable the performance mode! And all cores are recommended to be enabled; otherwise, there might be lag. I have tested this and found that enabling it makes the performance much smoother.

```
echo performance > /sys/devices/system/cpu/cpufreq/policy0/scaling_governor
echo performance > /sys/devices/system/cpu/cpufreq/policy4/scaling_governor
# Two major cores
echo performance > /sys/devices/system/cpu/cpufreq/policy6/scaling_governor
# Two major cores
```

Switch the governor to userspace

```
echo userspace > /sys/devices/system/cpu/cpufreq/policy0/scaling_governor
echo userspace > /sys/devices/system/cpu/cpufreq/policy4/scaling_governor
# Two major cores
echo userspace > /sys/devices/system/cpu/cpufreq/policy6/scaling_governor
# Two major cores
```

To set the CPU frequency, you need to first switch to the userspace. Please note that this time, it is only for demonstrating the adjustment of the main frequency of the secondary core.

```
echo 1800000 > /sys/devices/system/cpu/cpufreq/policy0/scaling_setspeed
echo 1800000 > /sys/devices/system/cpu/cpufreq/policy4/scaling_setspeed
# Two major cores
echo 1800000 > /sys/devices/system/cpu/cpufreq/policy6/scaling_setspeed
# Two major cores
```

Check the current running CPU frequency again

```
cat /sys/devices/system/cpu/cpu*/cpufreq/cpuinfo_cur_freq
```

3.14 Headphone Insertion/Extraction Test

Use a three-section headphone plug and align it with the headphone hole on the development board for testing. The insertion and extraction of the headphones is essentially an "event", and we can use the evtest program to conduct the test.

Execute the evtest command and view the corresponding events as shown in the following figure. When inserting or extracting the headphones, the program will print the corresponding information.

```
evtest
```

```
root@quarkpi-ca2:/# evtest
No device specified, trying to scan all of /dev/input/event*
Available devices:
/dev/input/event0:      fd8b0030.pwm
/dev/input/event1:      rk805 pwrkey
/dev/input/event2:      rockchip-dp0 rockchip-dp0
/dev/input/event3:      headset-keys
/dev/input/event4:      rockchip-es8388 Headset
/dev/input/event5:      bt-powerkey
/dev/input/event6:      adc-keys
/dev/input/event7:      rockchip-hdmi0 rockchip-hdmi0
Select the device event number [0-7]: 3
Input driver version is 1.0.1
Input device ID: bus 0x19 vendor 0x1 product 0x1 version 0x100
Input device name: "headset-keys"
Supported events:
  Event type 0 (EV_SYN)
  Event type 1 (EV_KEY)
    Event code 164 (KEY_PLAYPAUSE)
Properties:
Testing ... (interrupt to exit)
Event: time 1734593734.568792, type 1 (EV_KEY), code 164 (KEY_PLAYPAUSE), value 1
Event: time 1734593734.568792, -------------- SYN_REPORT ------------
Event: time 1734593735.085423, type 1 (EV_KEY), code 164 (KEY_PLAYPAUSE), value 0
Event: time 1734593735.085423, -------------- SYN_REPORT ------------
Event: time 1734593738.901875, type 1 (EV_KEY), code 164 (KEY_PLAYPAUSE), value 1
Event: time 1734593738.901875, -------------- SYN_REPORT ------------
Event: time 1734593739.521867, type 1 (EV_KEY), code 164 (KEY_PLAYPAUSE), value 0
Event: time 1734593739.521867, -------------- SYN_REPORT ------------
```

Printed information for plugging and unplugging the headphones

3.15 View NPU Information

Check the NPU usage rate. Note: The usage rate will change only when the routine that uses the NPU is executed.

```
cat /sys/kernel/debug/rknpu/load
```

```
root@quarkpi-ca2:/# cat /sys/kernel/debug/rknpu/load
NPU load:  Core0:  0%, Core1:  0%, Core2:  0%,
root@quarkpi-ca2:/#
```

3.16 PCIE WIFI & BT Interface Test

The Buildroot system of QuarkPi-CA2 is by default compatible with the RTL8852BE PCIE WIFI module sold by the ALIENTEK flagship store. The module is shown in the following picture.

Connect the RTL8852BE module to the M.2 interface on the back of QuarkPi-CA2 and attach the Xiao Huali antenna as shown in the picture below.

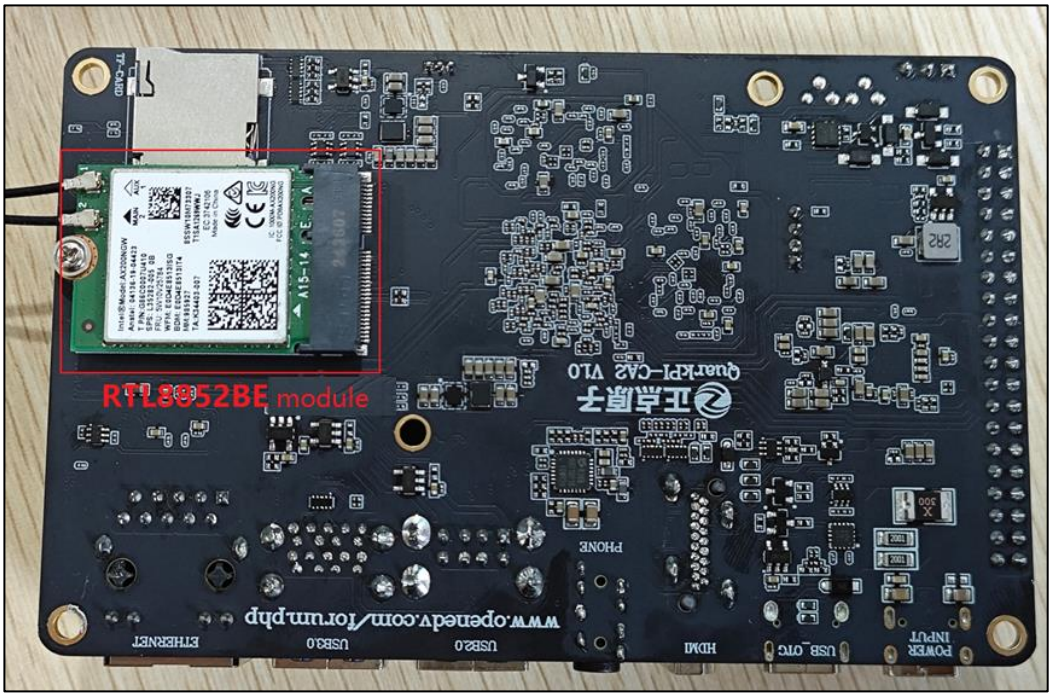


Figure 3.16-1 Installation effect diagram of RTL8852BE

3.16.1 WIFI Function Test

The WIFI driver of RTL8852BE is located in the /lib/modules/ directory of the Buildroot system, including 8852be.ko and rtkm.ko.

```
root@quarkpi-ca2:/#
root@quarkpi-ca2:/# ls /lib/modules/
5.10.160   8852be.ko  bcmdhd.ko       hci_uart.ko   rtkm.ko       RTL8189FU.ko  RTL8821CS.ko
8733bu.ko  8852bs.ko  bcmdhd_pcie.ko  rtk_btusb.ko  RTL8189FS.ko  RTL8723DS.ko  RTL8822CS.ko
root@quarkpi-ca2:/#
root@quarkpi-ca2:/#
```

Figure 3.16-2 RTL8852BE driver

Before conducting the WIFI function test, it is necessary to load the rtkm.ko and 8852be.ko in sequence, as shown in the following figure:

```
root@quarkpi-ca2:/#
root@quarkpi-ca2:/# insmod /lib/modules/rtkm.ko
[  258.356195] rtkm: rtkm_prealloc_init
[  258.356240] rtkm: rtkm_init_phy
[  258.356244] rtkm: rtkm_init_phy: memory expansion:1
[  258.356278] rtkm: ====================== RTKM ======================
[  258.356286] rtkm:  order        use       peak     alloc+       size
[  258.356293] rtkm: ---------------------------------------------------
[  258.356300] rtkm:      1          0          0          4      32768
[  258.356307] rtkm:      2          0          0          2      32768
[  258.356313] rtkm:      3          0          0          2      65536

  258.356320] rtkm:      4          0          0          4     262144
[  258.356326] rtkm:    sum          0          0         12     393216
[  258.356333] rtkm: rtkm_prealloc_init: done ret=0
root@quarkpi-ca2:/# insmod /lib/modules/8852be.ko
[  266.962689] rtl8852be 0004:41:00.0: enabling device (0000 -> 0003)
[  267.268415] eric-tx CALL alloc_txring !!!!
[  267.268847] eric-tx CALL alloc_txring !!!!

  267.275537] rtl8852be 0004:41:00.0 wlP4p65s0: renamed from wlan1
[  267.276347] udevd[1797]: Error changing net interface name wlan0 to wlP4p65s0: File exists
[  267.276387] udevd[1797]: could not rename interface '3' from 'wlan0' to 'wlP4p65s0': File exists
root@quarkpi-ca2:/#
root@quarkpi-ca2:/#
```

Figure 3.16-3 Load the driver

Next, execute "ifconfig -a" and it is found that the nodes "wlP4p65s0" and "wlan0" have appeared.

```
root@quarkpi-ca2:/# ifconfig -a
eth0      Link encap:Ethernet  HWaddr 0E:90:CB:B3:A0:D2
          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)
          Interrupt:97

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:290 errors:0 dropped:0 overruns:0 frame:0
          TX packets:290 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:17868 (17.4 KiB)  TX bytes:17868 (17.4 KiB)

wlP4p65s0 Link encap:Ethernet  HWaddr CA:FE:0F:EE:BE:17
          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 C8:FE:0F:EE:BE:17
          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)

root@quarkpi-ca2:/#
```

Figure 3.16-4 PCIE WIFI node

Next, open /etc/wpa_supplicant.conf and make the configuration.

```
vim /etc/wpa_supplicant.conf
```

The modification result of the /etc/wpa_supplication.conf configuration file is as follows

```
ctrl_interface=/var/run/wpa_supplicant
ap_scan=1
update_config=1

network={
        ssid="ALIENTEK-linux"        WIFI name
        psk="111111111111"           WIFI password
        key_mgmt=WPA-PSK
}
~
~
```

Then execute the following command to connect to WIFI:

```
rfkill unblock wlan
wpa_supplicant -B -i wlP4p65s0 -c /etc/wpa_supplicant.conf
udhcpc -i wlP4p65s0
```

```
root@quarkpi-ca2:/# rfkill unblock wlan
root@quarkpi-ca2:/# wpa_supplicant -B -i wlP4p65s0 -c /etc/wpa_supplicant.conf
Successfully initialized wpa_supplicant
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
nl80211: kernel reports: Match already configured
root@quarkpi-ca2:/# [  890.973821] IPv6: ADDRCONF(NETDEV_CHANGE): wlP4p65s0: link becomes ready

root@quarkpi-ca2:/# udhcpc -i wlP4p65s0
udhcpc: started, v1.36.1
udhcpc: broadcasting discover
udhcpc: broadcasting select for 192.168.3.89, server 192.168.3.1
udhcpc: lease of 192.168.3.89 obtained from 192.168.3.1, lease time 86400
deleting routers
adding dns 192.168.3.1
root@quarkpi-ca2:/#
```

At this point, the WIFI connection has been successfully established, as shown below:

```
root@quarkpi-ca2:/# ifconfig
eth0      Link encap:Ethernet  HWaddr 0E:90:CB:B3:A0:D2
          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)
          Interrupt:97

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:786 errors:0 dropped:0 overruns:0 frame:0
          TX packets:786 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:47628 (46.5 KiB)  TX bytes:47628 (46.5 KiB)

wlP4p65s0 Link encap:Ethernet  HWaddr CA:FE:0F:EE:BE:17
          inet addr:192.168.3.89  Bcast:192.168.3.255  Mask:255.255.255.0
          inet6 addr: fe80::c8fe:fff:feee:be17/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:90 errors:0 dropped:0 overruns:0 frame:0
          TX packets:103 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:11247 (10.9 KiB)  TX bytes:7100 (6.9 KiB)

wlan0     Link encap:Ethernet  HWaddr C8:FE:0F:EE:BE:17
          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)

root@quarkpi-ca2:/#
```

Run the ping command to test whether the internet connection is available.

```
root@quarkpi-ca2:/# ping www.baidu.com
PING www.wshifen.com (103.235.47.188) 56(84) bytes of data.
64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=1 ttl=45 time=216 ms
64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=2 ttl=45 time=218 ms
64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=3 ttl=45 time=217 ms
64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=4 ttl=45 time=218 ms
64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=5 ttl=45 time=216 ms
64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=6 ttl=45 time=217 ms
64 bytes from 103.235.47.188 (103.235.47.188): icmp_seq=7 ttl=45 time=215 ms
^C64 bytes from 103.235.47.188: icmp_seq=8 ttl=45 time=217 ms

--- www.wshifen.com ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 57993ms
rtt min/avg/max/mdev = 215.490/216.637/218.365/0.970 ms
root@quarkpi-ca2:/#
```

3.16.2 Bluetooth Function Test

Before conducting the Bluetooth test, the corresponding Bluetooth driver needs to be loaded. The specific path is /lib/modules/rtk_btusb.ko. Normally, the Bluetooth driver will be automatically loaded during system startup (it is necessary to insert the RTL8852BE PCIE WIFI module first), as shown in the following figure:

```
root@quarkpi-ca2:/# lsmod
Module                  Size  Used by
rtk_btusb              69632  0
root@quarkpi-ca2:/#
```

If the loading is not completed, execute the following command to start the loading:

```
insmod /lib/modules/rtk_btusb.ko
```

Execute the "hciconfig -a" command to view the HCI devices:

```
hciconfig -a
```

```
root@quarkpi-ca2:/# hciconfig -a
hci0:   Type: Primary  Bus: USB
        BD Address: C8:FE:0F:EE:BE:18  ACL MTU: 1021:6  SCO MTU: 255:12
        UP RUNNING
        RX bytes:792 acl:0 sco:0 events:61 errors:0
        TX bytes:3252 acl:0 sco:0 commands:61 errors:0
        Features: 0xff 0xff 0xff 0xfa 0xdb 0xbf 0x7b 0x87
        Packet type: DM1 DM3 DM5 DH1 DH3 DH5 HV1 HV2 HV3
        Link policy: RSWITCH HOLD SNIFF PARK
        Link mode: PERIPHERAL ACCEPT
        Name: 'BlueZ 5.72'
        Class: 0x6c0000
        Service Classes: Rendering, Capturing, Audio, Telephony
        Device Class: Miscellaneous,
        HCI Version: 5.2 (0xb)  Revision: 0xd9b7
        LMP Version: 5.2 (0xb)  Subversion: 0x44d8
        Manufacturer: Realtek Semiconductor Corporation (93)

root@quarkpi-ca2:/#
```

The built-in Bluetooth of the PCIe WIFI module

Then, use the bluetoothctl tool to configure and connect the Bluetooth. Execute the bluetoothctl command to enter the interactive mode.

```
bluetoothctl
```

Execute the following instructions in sequence

```
power on            # Turn on Bluetooth power
```

agent on	# Enable agent
discoverable on	# Enable discoverability
scan on	# Enable device scanning

After scanning your own Bluetooth device, input "scan off" to stop the scanning.

```
scan off
```

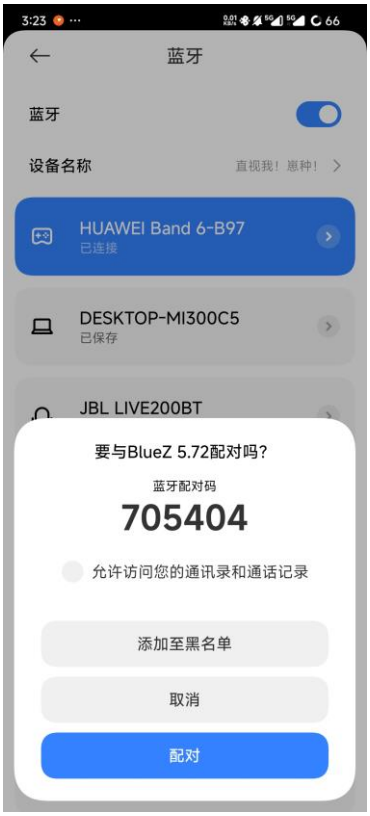
```
root@quarkpi-ca2:/#
root@quarkpi-ca2:/# bluetoothctl
Waiting to connect to bluetoothd...[bluetooth]# hci0 new_settings: powered bondable ssp br/edr le secure-conn
[bluetooth]# Agent registered
[bluetooth]# [CHG] Controller C8:FE:0F:EE:BE:18 Pairable: yes
[bluetooth]# power on
[bluetooth]# Changing power on succeeded
[bluetooth]# discoverable-timeout 0
[bluetooth]# Changing discoverable-timeout 0 succeeded
[bluetooth]# [CHG] Controller C8:FE:0F:EE:BE:18 DiscoverableTimeout: 0x00000000 (0)
[bluetooth]# scan on
[bluetooth]# SetDiscoveryFilter success
[bluetooth]# [  106.581832] rtk_btcoex: hci (periodic)inq start
hci0 type 7 discovering on
[bluetooth]# Discovery started
[bluetooth]# [CHG] Controller C8:FE:0F:EE:BE:18 Discovering: yes
[bluetooth]# [NEW] Device B4:60:77:8E:7B:DA B4-60-77-8E-7B-DA
[bluetooth]# [NEW] Device A6:D7:3C:42:4A:D3 L3250 Series
[bluetooth]# [NEW] Device 4B:C1:D1:94:5B:BC 4B-C1-D1-94-5B-BC
[bluetooth]# [NEW] Device 56:FC:5B:89:14:85 56-FC-5B-89-14-85
[bluetooth]# [NEW] Device B0:F0:0C:EE:C5:0C iKF-King S⁺ APP
[bluetooth]# [NEW] Device F1:22:33:BC:AE:6C F1-22-33-BC-AE-6C
[bluetooth]# [NEW] Device 72:D8:94:FE:3F:E4 72-D8-94-FE-3F-E4
[bluetooth]# [NEW] Device 30:53:C1:84:6A:36 LE_ATH-CKS30TW
[bluetooth]# [NEW] Device 5B:DA:58:85:5B:FE 5B-DA-58-85-5B-FE
[bluetooth]# [NEW] Device F1:22:33:2D:8E:6B F1-22-33-2D-8E-6B
[bluetooth]# [NEW] Device F1:22:33:BC:37:8F F1-22-33-BC-37-8F
[bluetooth]# [NEW] Device F1:22:33:B9:6E:25 AIMA-B96E25
[bluetooth]# [NEW] Device 77:D3:04:62:96:35 77-D3-04-62-96-35
[bluetooth]# [NEW] Device 64:97:6D:67:78:CD 64-97-6D-67-78-CD
[bluetooth]# [NEW] Device 67:6E:7F:D9:2B:4A 67-6E-7F-D9-2B-4A
[bluetooth]# [NEW] Device 5B:27:D1:E5:4B:19 5B-27-D1-E5-4B-19
[bluetooth]# [NEW] Device 5F:31:3F:31:78:9D lock_5F313F31789D
[bluetooth]# [NEW] Device 66:67:AC:EB:E9:5D 66-67-AC-EB-E9-5D
[bluetooth]# [NEW] Device 50:21:7D:46:5D:EA 50-21-7D-46-5D-EA
[bluetooth]# [NEW] Device 00:13:EF:A0:04:F1 DX3 Pro
[bluetooth]# [CHG] Device 67:6E:7F:D9:2B:4A RSSI: 0xffffffbe (-66)
[bluetooth]# [CHG] Device 67:6E:7F:D9:2B:4A ManufacturerData.Key: 0x004c (76)
[bluetooth]# [CHG] Device 67:6E:7F:D9:2B:4A ManufacturerData.Value:
[bluetooth]#   10 07 2c 1f 36 90 bb cc 68                        ..,.6...h
[bluetooth]# [NEW] Device 5E:8D:96:6C:A2:14 5E-8D-96-6C-A2-14
[bluetooth]# [NEW] Device 5E:C2:39:D5:FD:03 5E-C2-39-D5-FD-03
[bluetooth]# [NEW] Device 48:73:CB:85:CA:85 48-73-CB-85-CA-85
[bluetooth]# [NEW] Device 6C:9D:51:BA:6D:6F 6C-9D-51-BA-6D-6F
[bluetooth]# [NEW] Device 04:10:6B:D5:41:0F 直视我！崽种！
[bluetooth]# [NEW] Device FC:E8:00:F8:72:2C EDIFIER BLE
```

fter scanning the Bluetooth MAC address of the Android phone, you can proceed to perform the following configuration steps.

```
pair 04:10:6B:D5:41:0F
```

```
[bluetooth]# [CHG] Controller C8:FE:0F:EE:BE:18 Discovering: no
[bluetooth]# pair 04:10:6B:D5:41:0F
Attempting to pair with 04:10:6B:D5:41:0F
[bluetooth]# hci0 device_flags_changed: 04:10:6B:D5:41:0F (BR/EDR)
[bluetooth]#       supp: 0x00000001  curr: 0x00000000
[bluetooth]# [  272.521612] rtk_btcoex: hci create connection, start paging
[  272.886613] rtk_btcoex: connected, handle 0001, status 0x00
[  272.886644] rtk_btcoex: Page success
hci0 04:10:6B:D5:41:0F type BR/EDR connected eir_len 28
[CHG] Device 04:10:6B:D5:41:0F Connected: yes
[直视我！崽种！]# [  273.161610] rtk_btcoex: io capability request
Request confirmation
```

Input "yes"



Enter the following instruction: Trust MAC

```
trust    04:10:6B:D5:41:0F
```

```
[bluetooth]# trust  04:10:6B:D5:41:0F
[bluetooth]# [CHG] Device 04:10:6B:D5:41:0F Trusted: yes
[bluetooth]# Changing 04:10:6B:D5:41:0F trust succeeded
[bluetooth]#
```

Execute the following command to connect the mobile phone's Bluetooth. Then, use the phone to play music and transfer it via Bluetooth to the card computer for playback. You can plug in the 3.5mm jack of the card computer to hear the music sound.

```
[bluetooth]# connect 04:10:6B:D5:41:0F
Attempting to connect to 04:10:6B:D5:41:0F
[bluetooth]# [ 1446.428629] rtk_btcoex: hci create connection, start paging
[ 1447.507612] rtk_btcoex: connected, handle 0002, status 0x00
[ 1447.507642] rtk_btcoex: Page success
hci0 04:10:6B:D5:41:0F type BR/EDR connected eir_len 28
[CHG] Device 04:10:6B:D5:41:0F Connected: yes
[直视我! 崽种! ]# [ 1447.651645] rtk_btcoex: list_allocate_add: Add a2dp signal channel
[NEW] Endpoint /org/bluez/hci0/dev_04_10_6B_D5_41_0F/sep1
[直视我! 崽种! ]# [NEW] Endpoint /org/bluez/hci0/dev_04_10_6B_D5_41_0F/sep2
[直视我! 崽种! ]# [NEW] Endpoint /org/bluez/hci0/dev_04_10_6B_D5_41_0F/sep3
[直视我! 崽种! ]# [NEW] Endpoint /org/bluez/hci0/dev_04_10_6B_D5_41_0F/sep4
[直视我! 崽种! ]# [NEW] Endpoint /org/bluez/hci0/dev_04_10_6B_D5_41_0F/sep5
[直视我! 崽种! ]# [NEW] Transport /org/bluez/hci0/dev_04_10_6B_D5_41_0F/sep1/fd0
[直视我! 崽种! ]# [ 1447.740653] rtk_btcoex: list_allocate_add: Add a2dp media channel
Connection successful
[直视我! 崽种! ]# [ 1447.774792] input: 直视我! 崽种! (AVRCP) as /devices/virtual/input/input8
[NEW] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 [default]
[直视我! 崽种! ]# [08:24:05.784] event8  - 直视我! 崽种! (AVRCP): is tagged by udev as: Keyboard
[08:24:05.784] event8  - 直视我! 崽种! (AVRCP): device is a keyboard
[08:24:05.785] libinput: configuring device "直视我! 崽种! (AVRCP)".
[CHG] Device 04:10:6B:D5:41:0F ServicesResolved: yes
[直视我! 崽种! ]# [CHG] Transport /org/bluez/hci0/dev_04_10_6B_D5_41_0F/sep1/fd0 Volume: 0x003b (59)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Repeat: off
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Shuffle: off
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Status: paused
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Title: Not Provided
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.TrackNumber: 0x00000001 (1)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.NumberOfTracks: 0x00000001 (1)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Duration: 0x00000000 (0)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Position: 0x00000000 (0)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Position: 0x00000000 (0)
[直视我! 崽种! ]# [CHG] Transport /org/bluez/hci0/dev_04_10_6B_D5_41_0F/sep1/fd0 Volume: 0x007f (127)
[直视我! 崽种! ]# W: [pulseaudio] module-loopback.c: Configured latency of 200.00 ms is smaller than minimum latency, usin
W: [pulseaudio] module-loopback.c: Cannot set requested sink latency of 35.20 ms, adjusting to 39.91 ms
W: [pulseaudio] module-loopback.c: Cannot set requested source latency of 66.67 ms, adjusting to 250.00 ms
[CHG] Transport /org/bluez/hci0/dev_04_10_6B_D5_41_0F/sep1/fd0 Volume: 0x0055 (85)
[直视我! 崽种! ]#
```

After clicking to play music on the mobile phone, you can see the interactive interface of bluetootctl displaying the relevant information of the song.

```
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Title: 蒙娜丽莎的眼泪
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.TrackNumber: 0x00000001 (1)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.NumberOfTracks: 0x00000001 (1)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Duration: 0x00041eb0 (270000)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Album: 蒙娜丽莎的眼泪
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Artist: 林志炫
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Position: 0x00000000 (0)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Position: 0x0000001c (28)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Position: 0x00000539 (1337)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Title: 林志炫 - 蒙娜丽莎的眼泪
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.TrackNumber: 0x00000001 (1)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.NumberOfTracks: 0x00000001 (1)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Duration: 0x00041eb0 (270000)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Album: 蒙娜丽莎的眼泪
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Artist: 林志炫-蒙娜丽莎的眼泪
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Position: 0x00000579 (1401)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Position: 0x00000ce8 (3304)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Title: 词: 郑华娟
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.TrackNumber: 0x00000001 (1)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.NumberOfTracks: 0x00000001 (1)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Duration: 0x00041eb0 (270000)
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Album: 蒙娜丽莎的眼泪
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Track.Artist: 林志炫-蒙娜丽莎的眼泪
[直视我! 崽种! ]# [CHG] Player /org/bluez/hci0/dev_04_10_6B_D5_41_0F/player0 Position: 0x00000d23 (3363)
[直视我! 崽种! ]#
```

Chapter 4. Common Function Settings of the Buildroot System

This section summarizes the functions that ALIENTEK users frequently require for the system.

4.1 Mount Point Information

After the system starts up, the mount points are as follows

```
root@quarkpi-ca2:/# df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        14G  1.3G   12G  10% /
devtmpfs        7.8G  8.0K  7.8G   1% /dev
tmpfs           7.8G  204K  7.8G   1% /tmp
tmpfs           7.8G  624K  7.8G   1% /run
tmpfs           7.8G  172K  7.8G   1% /var/log
tmpfs           7.8G     0  7.8G   0% /dev/shm
/dev/mmcblk0p7  124M   12M  109M  10% /oem
/dev/mmcblk0p8  101G  488K  101G   1% /userdata
root@quarkpi-ca2:/#
```

Figure 4.1-1 Mount points

The important mount points are as follows:

/: Root directory, which is the common root directory we often encounter, containing common folders/files such as /dev/ and /usr/lib.

/oem: Manufacturer folder, where the test files/programs provided by the manufacturer are stored.

/userdata: User data folder. It is recommended that users save their data to this directory because this folder has more space than the root directory. In the future when users create products, it is also recommended to save the data generated by the App to this folder to prevent the root partition from being filled up and affecting startup.

4.2 Set Static IP

Setting a static IP is the most necessary requirement for users. Here we provide a method. Since the ALIENTEK Buildroot system is hosted by connmand, it is only applicable to the ALIENTEK Buildroot system! Since connmand automatically acquires the IP, when we set a static IP, it will be re-acquired by connmand, so this approach is not feasible.

The default situation of the Buildroot Linux system is as follows: After plugging in the network cable, it will automatically obtain the IP:

```
root@quarkpi-ca2:/# ifconfig
eth0      Link encap:Ethernet  HWaddr 0E:90:CB:B3:A0:D2
          inet addr:192.168.6.13  Bcast:192.168.6.255  Mask:255.255.255.0
          inet6 addr: fe80::c90:cbff:feb3:a0d2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:284 errors:0 dropped:0 overruns:0 frame:0
          TX packets:38 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:19079 (18.6 KiB)  TX bytes:3697 (3.6 KiB)
          Interrupt:97

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:54 errors:0 dropped:0 overruns:0 frame:0
          TX packets:54 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4464 (4.3 KiB)  TX bytes:4464 (4.3 KiB)

root@quarkpi-ca2:/# Failed to init Wi-Fi for RTL8852BE!
```

Next, we need to configure connmand to prevent it from automatically obtaining the IP. Only then will the static IP we have set ourselves take effect.

Execute the following commands one by one

```
mkdir /etc/connman
vi /etc/connman/main.conf
```

In the main.conf file, add the following content to blacklist eth0. This way, eth0 will not be managed.

```
[General]
NetworkInterfaceBlacklist = eth0
```

After saving, execute the "reboot" command to restart the development board. You will find that the "eth0" network card is no longer listed. This indicates that the blacklist has taken effect.

```
root@quarkpi-ca2:/# ifconfig
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:62 errors:0 dropped:0 overruns:0 frame:0
          TX packets:62 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4228 (4.1 KiB)  TX bytes:4228 (4.1 KiB)

root@quarkpi-ca2:/#
```

Now, set the static IP and edit the /etc/network/interfaces file

```
vi /etc/network/interfaces
```

Please add the following content: Set the static IP to 192.168.9.11. The specific settings should be adjusted according to your actual network segment.

```
auto eth0
iface eth0 inet static
    address 192.168.9.11
    netmask 255.255.255.255
    network 192.168.9.0
    gateway 192.168.9.1
```

Save the modifications! Execute the following command to restart the development board.

```
sync
reboot
```

Restart the development board again! Check if the static IP is effective. You can see from the following picture that eth0 has been restarted and the IP has changed to 192.168.9.11.

```
ifconfig
```

```
root@quarkpi-ca2:/# ifconfig
eth0      Link encap:Ethernet  HWaddr 0E:90:CB:B3:A0:D2
          inet addr:192.168.9.11  Bcast:0.0.0.0  Mask:255.255.255.255
          inet6 addr: fe80::c90:cbff:feb3:a0d2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:880 errors:0 dropped:0 overruns:0 frame:0
          TX packets:9 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:56527 (55.2 KiB)  TX bytes:726 (726.0 B)
          Interrupt:97

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:66 errors:0 dropped:0 overruns:0 frame:0
          TX packets:66 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:4428 (4.3 KiB)  TX bytes:4428 (4.3 KiB)

root@quarkpi-ca2:/#
```

4.3 Creating Bootloader Program

Creating a boot loader program in the Buildroot system of ALIENTEK is quite simple. All you need to do is add your own startup program in the /etc/init.d/ directory. The author has provided a template as follows.

For example, create a file named S99example in the /etc/init.d/. Remember to follow the format of SxxXXX, where xx should preferably be in the range of 50 to 99. The larger the number, the later the startup. It cannot exceed 100. XXX can be freely named, but it should not contain numbers or special characters.

```
vi /etc/init.d/S99example              # Edit the file
chmod 777 /etc/init.d/S99example       # Grant execution permission
sync                                   # Save
```

Write the following content on S99example

```
PATH=”/usr/local/sbin:/usr/local/bin:/sbin:/bin:/usr/sbin:/usr/bin”

start_example()
{
    # "Here is where you write your auto-start program. The '&' symbol is used to run it in the background. "
    # If your program is a blocking program, it must be run in the background; otherwise,
    # your system will not start properly.
    # xx represents the path of your program! Write the absolute path.
    /xx/example &
}
```

```
stop_example()
{
        # Terminate the program
        Killall example
}

case "$1" in
        start)
                echo -n "starting example…"
                start_example
                echo "done."
                ;;
        stop)
                echo -n "stoping example…"
                stop_example
                echo "done"
                ;;
        restart|reload)
                echo -n "stoping example…"
                stop_example && sleep 3
                echo "done."

                Echo -n "starting example…"
                start_example
                echo "done."
                ;;
        *)
                echo "Usage: $0 {start|stop|restart}
                exit 1
esac

exit 0
```