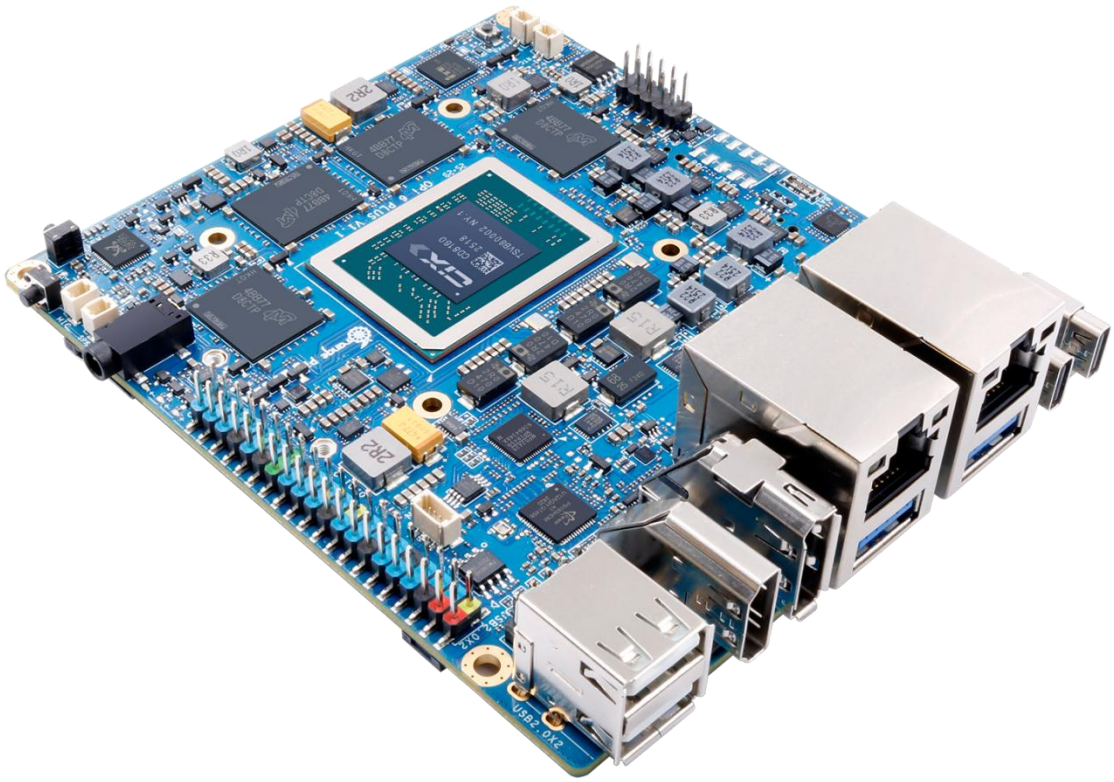
Orange Pi 6 Plus
Android User Manual



Catalogue

1. Orange Pi 6 Plus Basic Features 4
1. 1. What is Orange Pi 6 Plus 4
1. 2. Purpose of Orange Pi 6 Plus 4
1. 3. Orange Pi 6 Plus Hardware Specifications 5
1. 4. Orange Pi 6 Plus top and bottom views 7
1. 5. Orange Pi 6 Plus interface details 9
2. Introduction to the use of development board 11
2. 1. Prepare the necessary accessories 11
2. 2. Download the development board image and related information 18
2. 3. How to burn the Android image to NVMe SSD disk based on Windows PC 19
2. 4. How to burn the Android image to NVMe SSD disk based on Ubuntu PC 25
3. Android 14 system usage instructions 31
3. 1. Supported Android versions 31
3. 2. Android 14 Functional Adaptation 31
3. 3. How to use ADB 32
3. 3. 1. Use a data cable to connect adb debugging 32
3. 3. 2. Using adb debugging with network connection 33
3. 4. ADB system directory mounting method for reading and writing 35
3. 5. How to enter Fastboot on Android 36
3. 6. Wi-Fi connection method 37
3. 7. How to use Wi-Fi hotspot 39
3. 8. How to view the IP address of the Ethernet port 42
3. 9. Bluetooth connection method 44
3. 10. Switching between front and rear cameras in CAM1 and CAM2 on Android 47
3. 11. 40 pin interface UART, I2C, PWM, SPI test 50

3.11.1. 40 Pin UART Test Method 50
3.11.2. 40 Pin I2C Test Method 51
3.11.3. 40-pin PWM test method 53
3.11.4. 40-pin SPI test method 54
4. Compiling method of Android 14 source code 55
4.1. Download the source code of Android 14 55
4.2. Compile the source code of Android 14 55
5. Appendix 58
5.1. User Manual Update History 58
5.2. Image Update History 58

1. Orange Pi 6 Plus Basic Features

1.1. What is Orange Pi 6 Plus

OrangePi6Plus uses Cix's new generation 12-core 64-bit ARMv9 processor, specifically quad-core Cortex®-A720 large-core, quad-core Cortex®-A720 medium-core, and quad-core Cortex®-A520 small-core. The large core has a maximum main frequency of 2.6GHz and an integrated Arm® Immortals™ G720MC10 GPU. The CPU + NPU + GPU have a total computing power of approximately 45 Tops. Optional 16GB, 32GB, and 64GB of memory are available, with display processing capabilities up to 4K.

The Orange Pi 6 Plus offers a wealth of interfaces, including one HDMI output, one DisplayPort output, two USB-C/DP ports, an eDP display port, two M.2M-key PCIe 4.0 x4 ports, an M.2E-key PCIe 4.0 x1 port, two 5G Ethernet ports, USB 2.0 and USB 3.0 ports, headphone and MIC jacks, speaker jacks, an RTC port, and a 40-pin expansion header. It is widely applicable to high-end tablets, edge computing, artificial intelligence, cloud computing, AR/VR, smart security, smart home, and other fields, covering all AIoT industries.

1.2. Purpose of Orange Pi 6 Plus

We can use it to achieve:

- A high-performance Linux desktop computer
- AI development workstation
- Android game consoles
- OpenHarmony computers, etc

Of course, there are many more features. Backed by a robust ecosystem and a wide range of expansion accessories, OrangePi can help users easily move from idea to prototype and then to mass production, making it an ideal creative platform for makers, dreamers, and hobbyists.

1. 3. Orange Pi 6 Plus Hardware Specifications

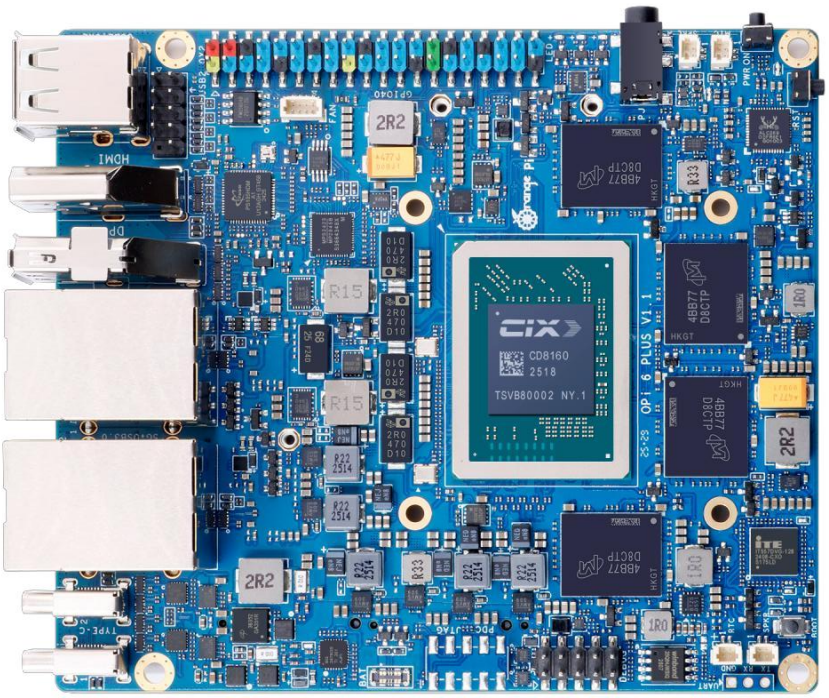
Hardware Features	
processor	• 12-core 64-bit processor • Quad-core big core Cortex®-A720 • Quad-core mid-core Cortex®-A720 • Quad-core small core Cortex®-A520
GPU	Arm® Immortals™ G720 MC10 GPU,support： • Hardware ray tracing • OpenGL® ES3.2 • OpenCL® 3.0 • Vulkan® 1.3
NPU	• Support INT4/INT8/INT16/FP16/BF16 and TF32 accelerate • Computing power up to 28.8TOPs
Video Encoder	• Support AV1/H.265/H.264/VP9/VP8/H.263/MPEG-4/MPEG-2 Format
Video decoder	• Supports H.265/H.264/VP9/VP8 formats
Memory	• 128-bit LPDDR5 type • 16GB、32GB and 64GB Optional capacity • 6000MT/s speed
Storage interface	• 1 x SPI Nor FLASH: 64Mbit • 1 x MicroSD card slot • 2 x NVMe SSD (PCIe4.0x4) M.2 M-Key 2280 slots

USB interface	• 2 x USB3.0 Host • 2 x USB2.0 Host • 9pin reserved pin header：2 x USB2.0 Host • 1 x Type-C USB3.0 Host • 1 x Type-C USB3.0 OTG，support Host 和 Device
Video Output	• 1 x HDMI, up to 4Kp60 • 1 x DisplayPort, up to 4Kp120 • 2 x USB-C DP, up to 4Kp60 • 1 x eDP，Up to 4Kp60
Camera	• 2 x MIPI CSI 4 Lane interface
TP interface	• 6pin 0.5mm Spacing FPC connector
Audio	• 1 x 3.5mm Headphone jack audio input/output • 1 x HDMI Audio output • 1 x DisplayPort Audio Output • 2 x USB-C DP Audio Output • 1 x MIC Audio input interface（2pin 1.0mm Specification） • 2 x Speaker audio output interface（2pin 1.0mm Specification）
Ethernet	• 2 x PCIe 1G/2.5G/5G Ethernet port（RTL8126)
40Pin	• Used to expand UART, I2C, SPI, PWM, GPIO interfaces
PCIE M.2	• Contains PCIe 4.0 x 1/USB2.0 and other interfaces, supports 2230 Wi-Fi6/BT module
powered by	• 2 x Type-C PD 20V 100W Power input

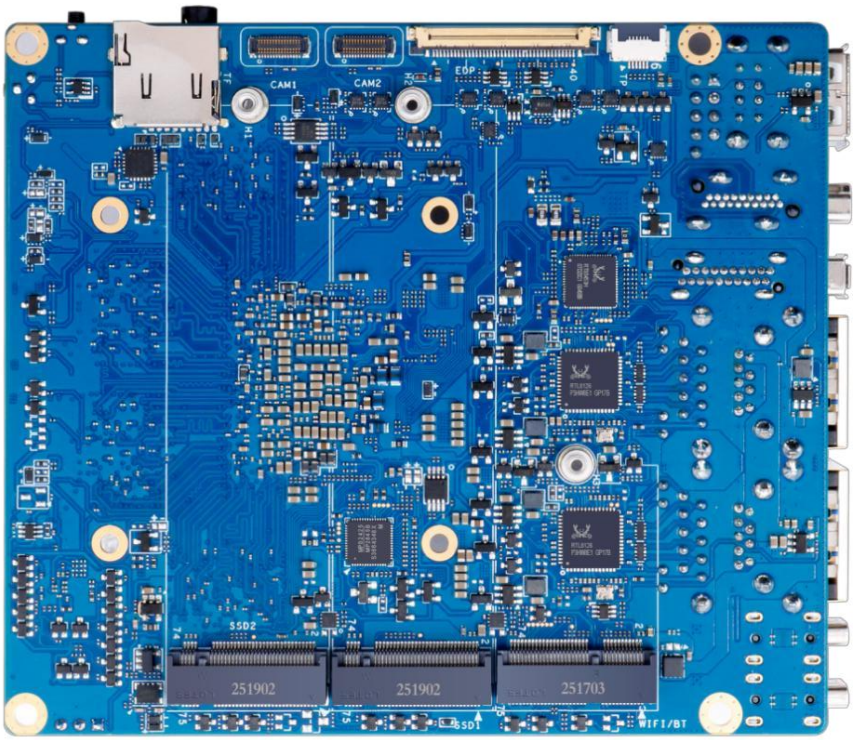
	• Board-to-board battery interface, 12V, three strings
button	• 1 x Fastboot key，1 x Reset button，1 x Power button
LED lights	• 1 x red LED light to indicate battery charging status • 1 x blue LED light to indicate Type-C power supply status • 1 x green LED light to indicate system operation status
Fan connector	• 4-pin 1.0mm pitch specification, used to connect 5V fan, supports PWM control of switch and speed
RTC battery interface	• 2-pin 1.0mm specification, used to power the RTC module
10-pin debug serial port	• 10-pin debug serial port, including： • UART2, BIOS and kernel log • UART4，PM log • UART5，SE log • UART6，Post code log
Supported OS	• Debian、Ubuntu、OpenHarmony、Android 和 Windows etc
Appearance Specifications	
PCB	115mm*100mm
weight	132g

1. 4. Orange Pi 6 Plus top and bottom views

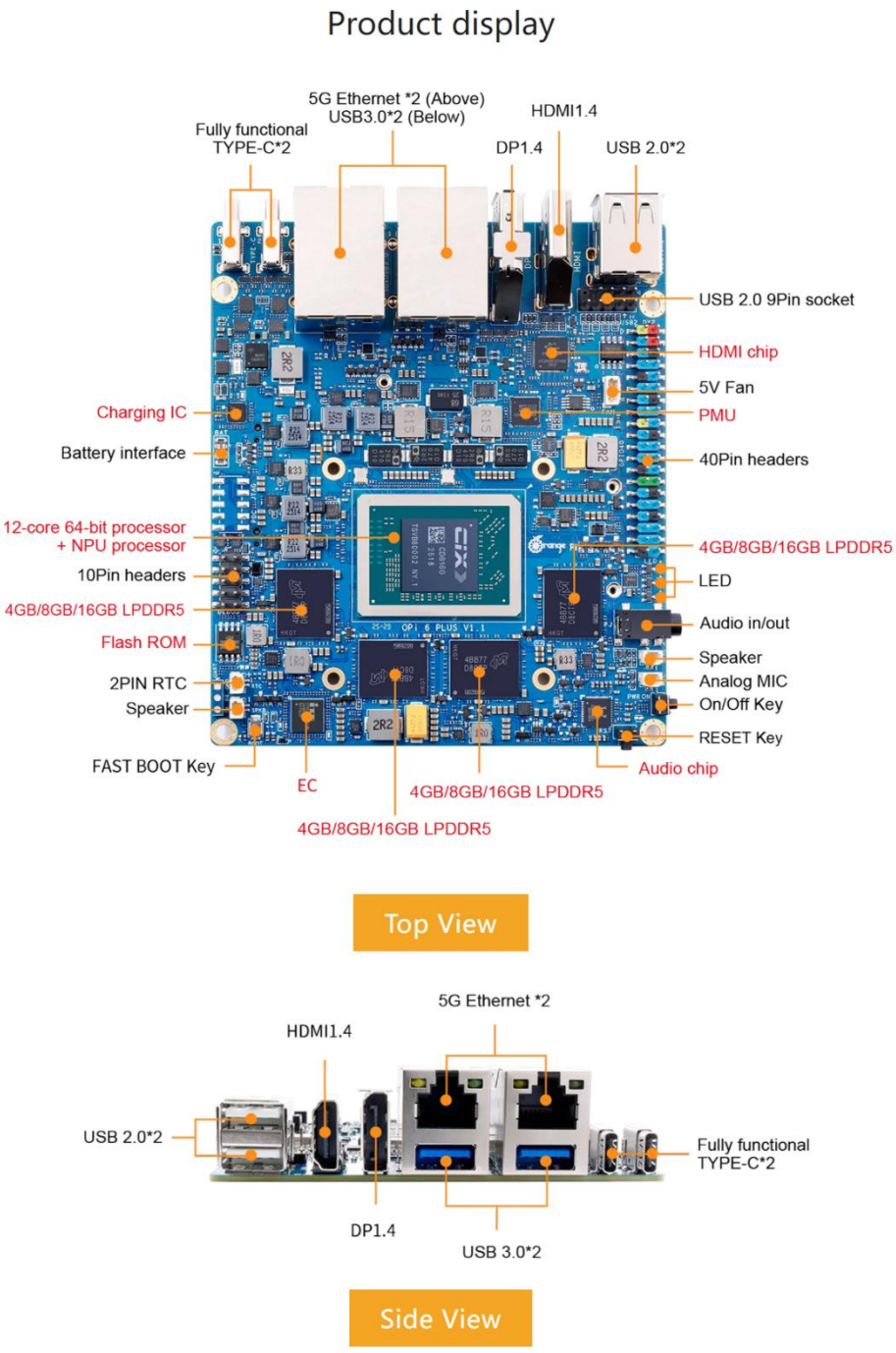
Top-level view：

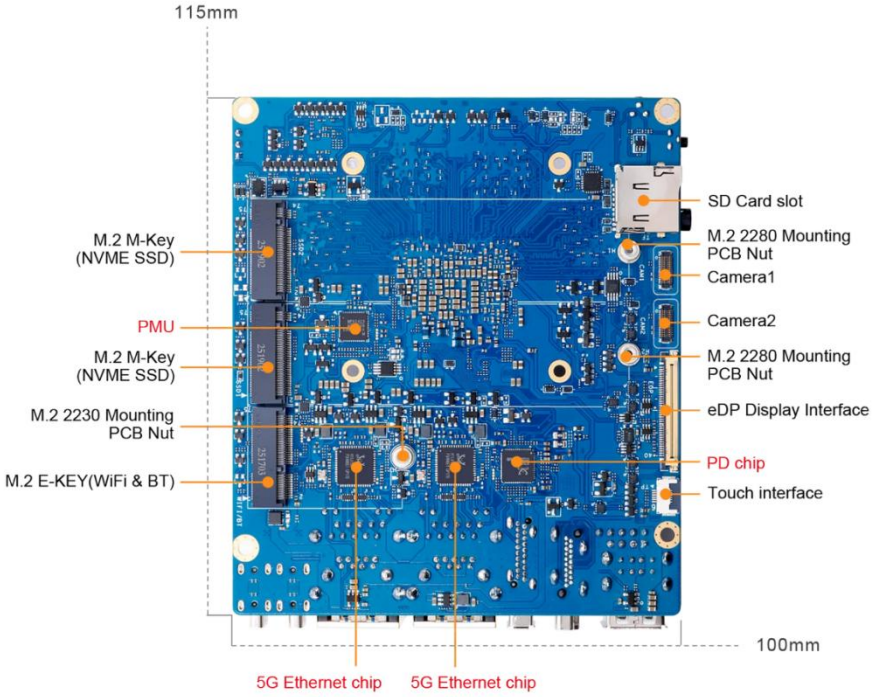


Bottom-level view：

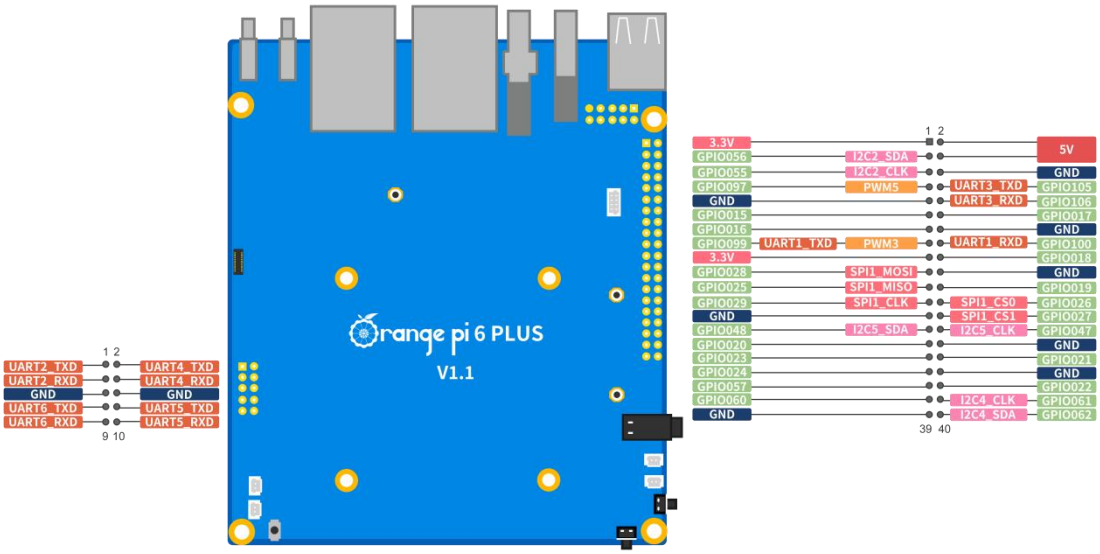


1. 5. Orange Pi 6 Plus interface details




115mm
SD Card slot
M.2 2280 Mounting PCB Nut
Camera1
Camera2
M.2 2280 Mounting PCB Nut
eDP Display Interface
PD chip
Touch interface
M.2 M-Key (NVME SSD)
PMU
M.2 M-Key (NVME SSD)
M.2 2230 Mounting PCB Nut
M.2 E-KEY(WiFi & BT)
100mm
5G Ethernet chip
5G Ethernet chip

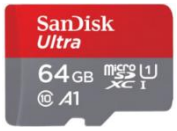
Bottom View


Orange pi 6 PLUS
V1.1
1 2
UART2_TXD
UART2_RXD
GND
UART6_TXD
UART6_RXD
UART4_TXD
UART4_RXD
GND
UART5_TXD
UART5_RXD
9 10
1 2
3.3V
GPIO056
GPIO055
GPIO097
GND
GPIO015
GPIO016
GPIO099
3.3V
GPIO028
GPIO025
GPIO029
GND
GPIO048
GPIO020
GPIO023
GPIO024
GPIO057
GPIO060
GND
I2C2_SDA
I2C2_CLK
PWM5
UART1_TXD
PWM3
SPI1_MOSI
SPI1_MISO
SPI1_CLK
I2C5_SDA
5V
GND
UART3_TXD
UART3_RXD
UART1_RXD
SPI1_CS0
SPI1_CS1
I2C5_CLK
I2C4_CLK
I2C4_SDA
GPIO105
GPIO106
GPIO017
GND
GPIO100
GPIO018
GND
GPIO019
GPIO026
GPIO027
GPIO047
GND
GPIO021
GND
GPIO022
GPIO061
GPIO062
39 40

2. Introduction to the use of development board

2. 1. Prepare the necessary accessories

1) TF card, minimum 8GB capacity, class 10 or higher high-speed SanDisk card



2) TF card reader, used to read and write TF cards

3) Display with HDMI interface

4) HDMI to HDMI cable, used to connect the development board to an HDMI monitor or TV for display

5) DP to HDMI cable, connect the development board to an HDMI monitor or TV via the DP interface for display.

6) Type-C to HDMI cable, connect the development board to an HDMI monitor or TV via the Type-C interface for display.

7) Type-C to USB adapter, used to connect USB storage devices or USB devices such as mouse and keyboard through the Type-C interface.

8) eDP screen, used to display the system interface of the development board.

9) 9-pin USB2.0 adapter cable, used to expand two USB2.0 Host interfaces

10) PCIe Wi-Fi Bluetooth module, such as the RTL8852BE module.

11) A USB mouse and keyboard. Any standard USB mouse and keyboard will do. The mouse and keyboard can be used to control the Orange Pi development board.

12) USB camera

13) 100M or 1000M Ethernet cable to connect the development board to the Internet

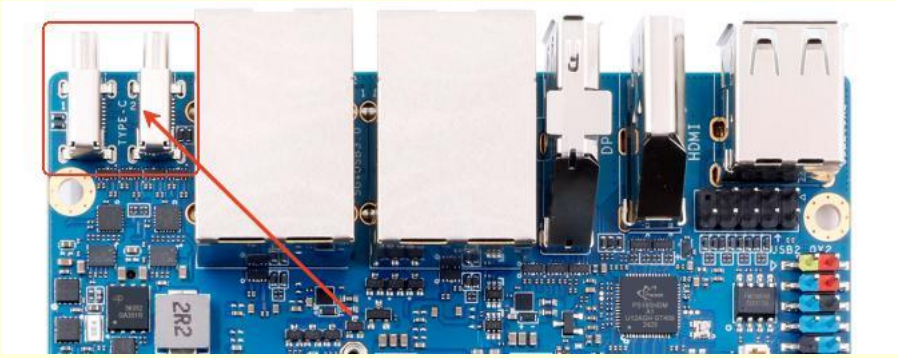
14) Type-c data cable, used for adb debugging and fastboot image burning

15) USB to TTL module and DuPont line. When using the serial port debugging function, you need a USB to TTL module and DuPont line to connect the development board and the computer.

Note that the TTL level used by the development board is 3.3v. In addition to the USB to TTL module shown above, other similar 3.3v USB to TTL modules are generally also acceptable.

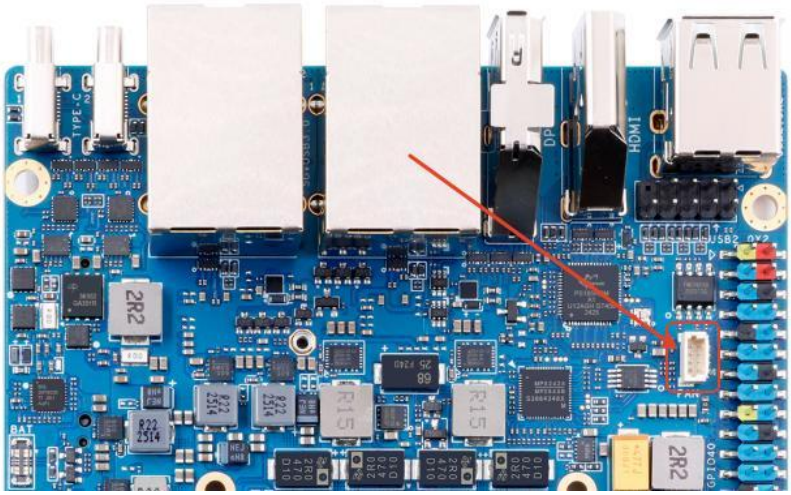
16) Power adapter: Orange Pi 6 Plus is recommended to use a 20V 100W Type-C PD power adapter.

There are two Type-C interfaces on the development board, both of which support powering the development board.



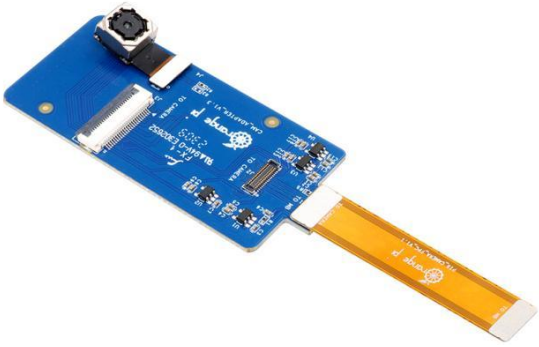
17) 5V cooling fan. As shown in the figure below, the development board has an interface for connecting a PWM cooling fan. The interface specification is **4 pins with a 1.0mm** pitch.

The fan on the development board can adjust the speed and switch through PWM.



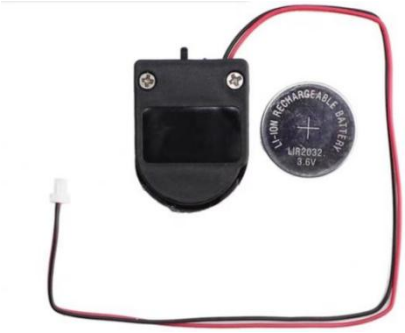
18) 13MP OV13850 camera with MIPI interface.

19) 13MP OV13855 camera with MIPI interface.



20)) M.2 M-KEY 2280 NVMe SSD solid state drive PCIe interface specification is PCIe4.0x4

21))RTC battery, 2-pin connector, 1.0mm pitch



The location of the RTC battery interface on the development board is shown in the figure below.:

2.2. Download the development board image and related information

1) The download link for the Chinese version is

http://www.orangepi.cn/html/hardWare/computerAndMicrocontrollers/service-and-support/Orange-Pi-6-Plus.html

2) The download link for the English version is

http://www.orangepi.org/html/hardWare/computerAndMicrocontrollers/service-and-support/Orange-Pi-6-Plus.html

3) The information mainly includes

a. **Linux source code**: saved on Github

b. **Android image:** saved on Baidu Cloud and Google Drive
c. **Android source code:** saved on Baidu Cloud Drive and Google Cloud Drive
d. **OpenHarmony image:** saved on Baidu Cloud and Google Drive
e. **Ubuntu image:** saved on Baidu Cloud and Google Drive
f. **Debian image:** saved on Baidu Cloud and Google Drive
g. **User manual and schematic diagram:** chip-related data sheets will also be placed here
h. **Official tools:** mainly include the software needed in the use of the development board

2. 3. How to burn the Android image to NVMe SSD disk based on Windows PC

Note that the Android image mentioned here specifically refers to the corresponding developed Android image downloaded from the Orange Pi data download page.

To run the Android system on the development board, you must first flash the BIOS corresponding to the Android version. For specific steps, please refer to the opi-6plus-bios user manual.

1) First, you need to prepare an M-Key 2280 NVMe SSD. The development board's M.2 slot PCIe interface is compatible with PCIe 3.0 and PCIe 4.0.

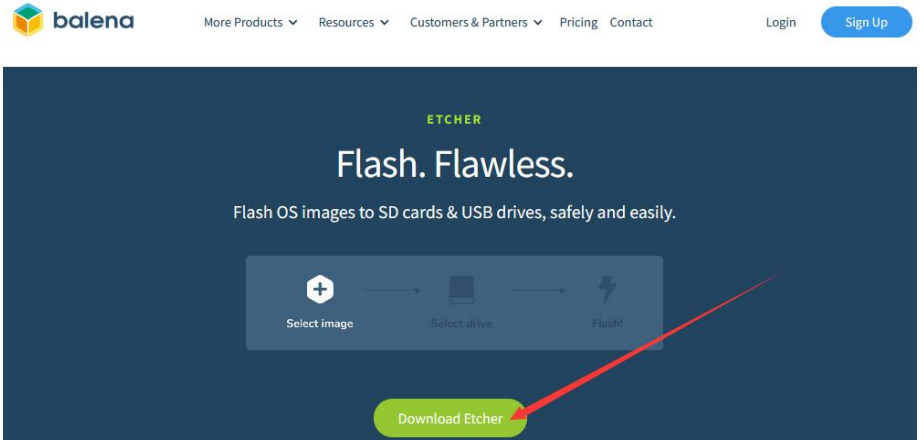
2) Use an NVMe SSD enclosure to connect the SSD to a Windows PC.

3) Download the compressed Android operating system image file you want to burn from Orange Pi's data download page, and then use decompression software to decompress it. Among the decompressed files, the file ending with ".img" is the operating system image file, which is generally over 1GB in size.

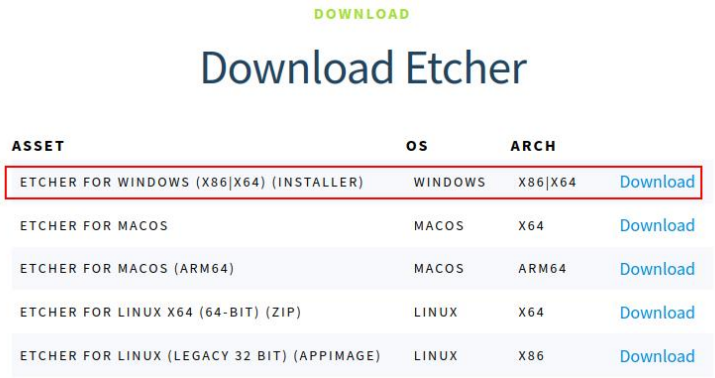
4) Then download the Android image burning software - balenaEtcher, the download address is

https://www.balena.io/etcher/

5) After entering the balenaEtcher download page, click the green download button to jump to the software download selection interface

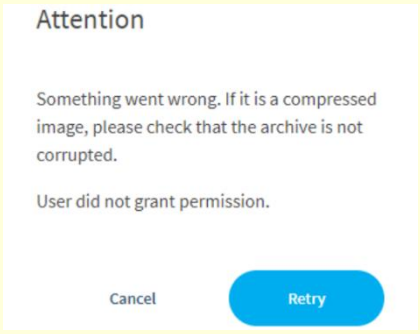


6) Then download the installation package of balenaEtcher Windows version.

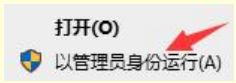


7) Then install balenaEtcher and open it. The balenaEtcher interface after opening is as shown below

If the following error is prompted when opening balenaEtcher:



Select balenaEtcher, right-click it, and then choose Run as administrator.

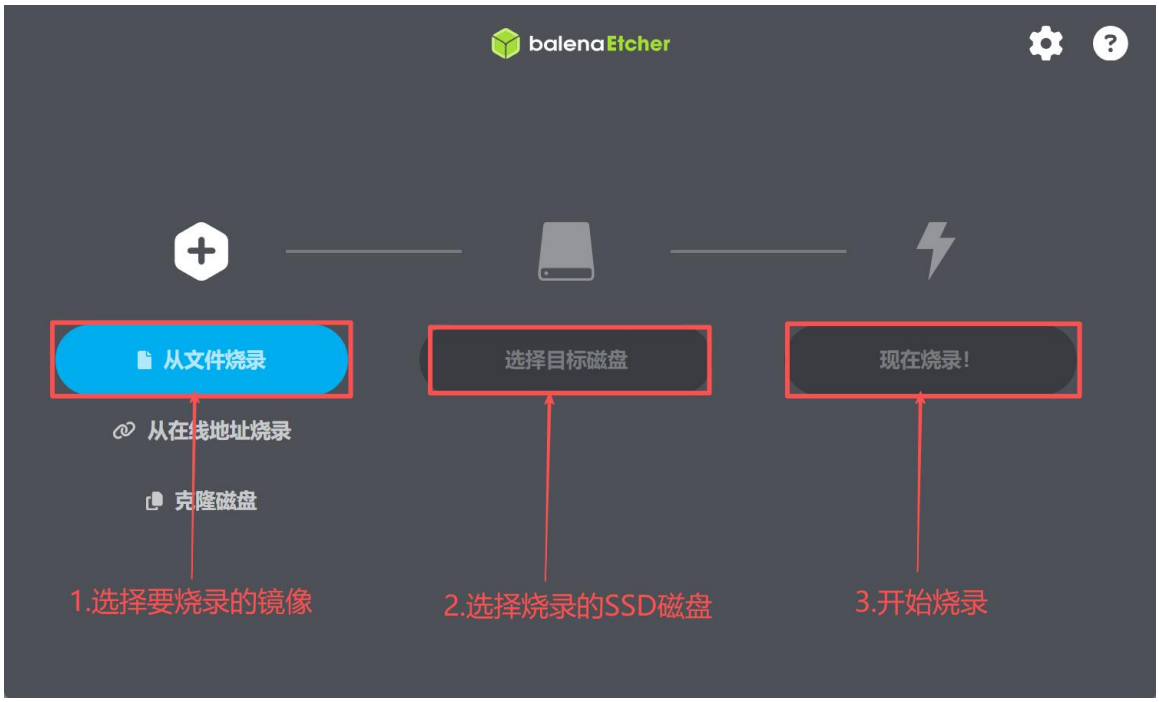


8) The specific steps for burning the Android image using balenaEtcher are as follows

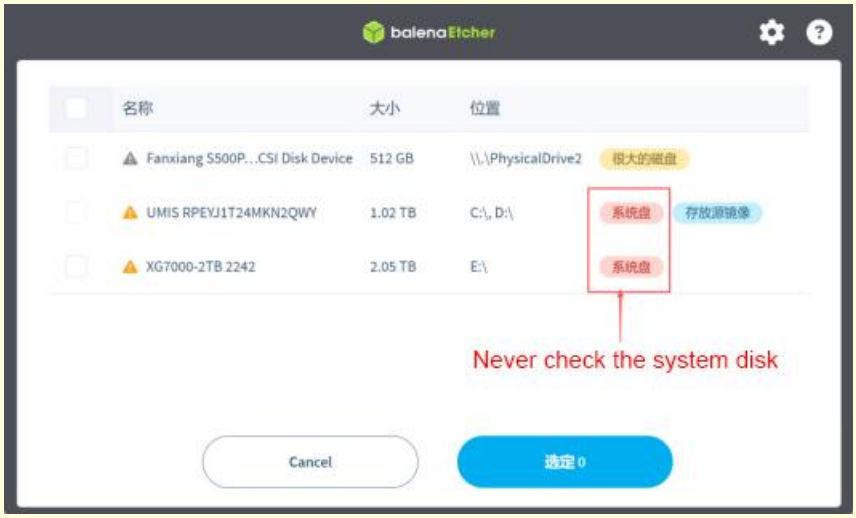
a. First select the path of the Android image file to be burned

b. Then select the drive letter of the SSD, be careful not to choose the wrong one

c. Finally, click Flash to start burning the Android image to the SSD disk.

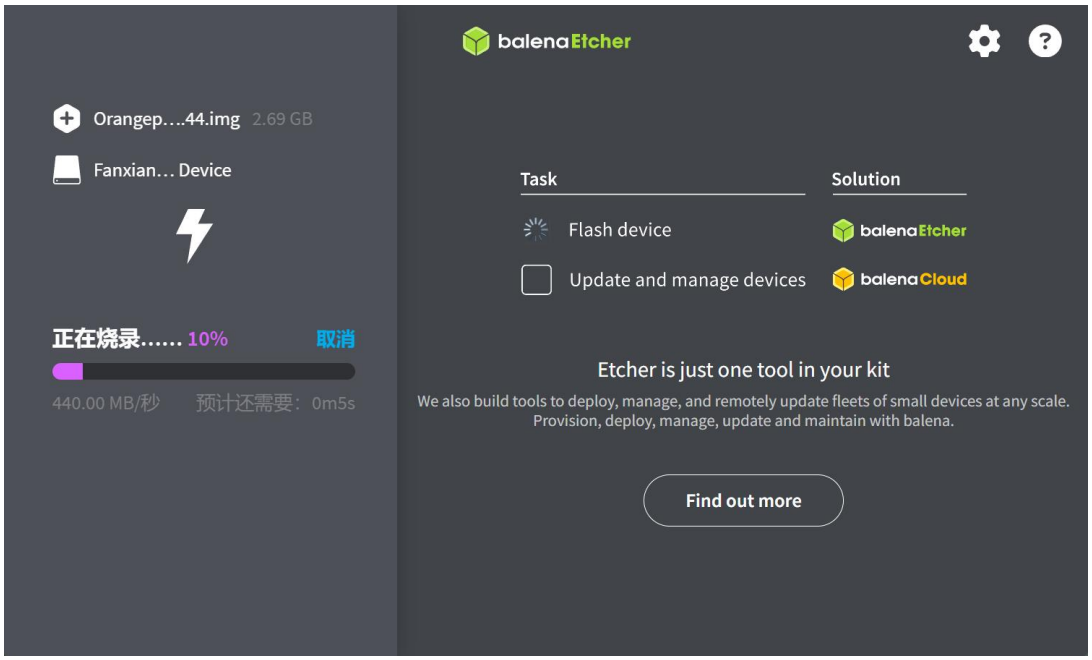


In the second step, when selecting the drive letter, be sure to distinguish between the system disk and the SSD disk to be burned. Do not mistakenly select the system disk



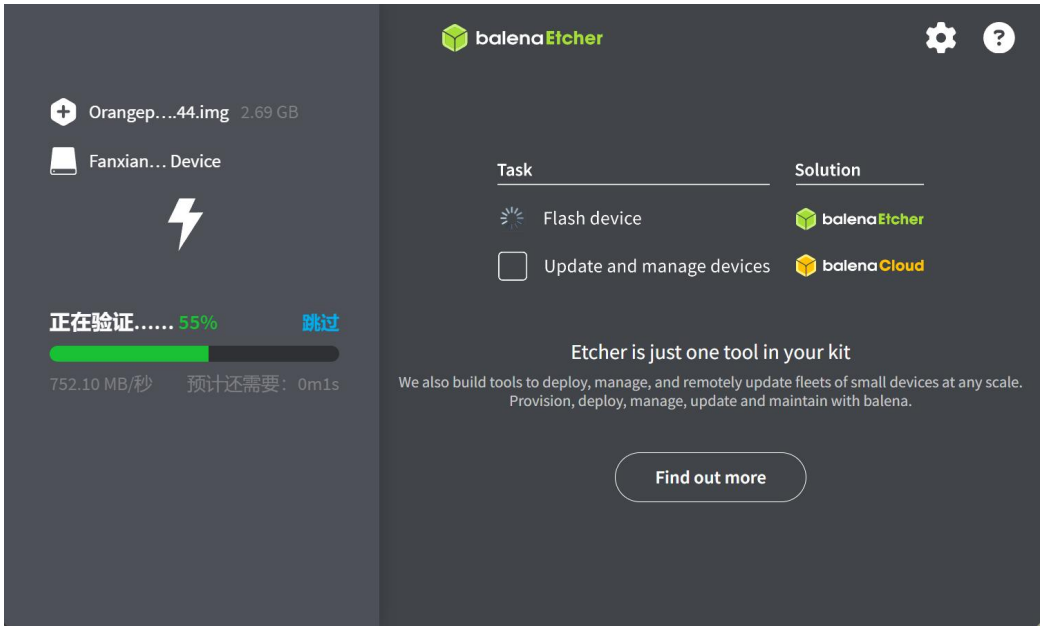
If the disk space is relatively large, the following prompt may appear. Just click "I'm OK".

9) The interface displayed during the process of burning the Linux image in balenaEtcher is shown in the figure below. In addition, the progress bar is purple, indicating that the Linux image is being burned to the SSD disk.

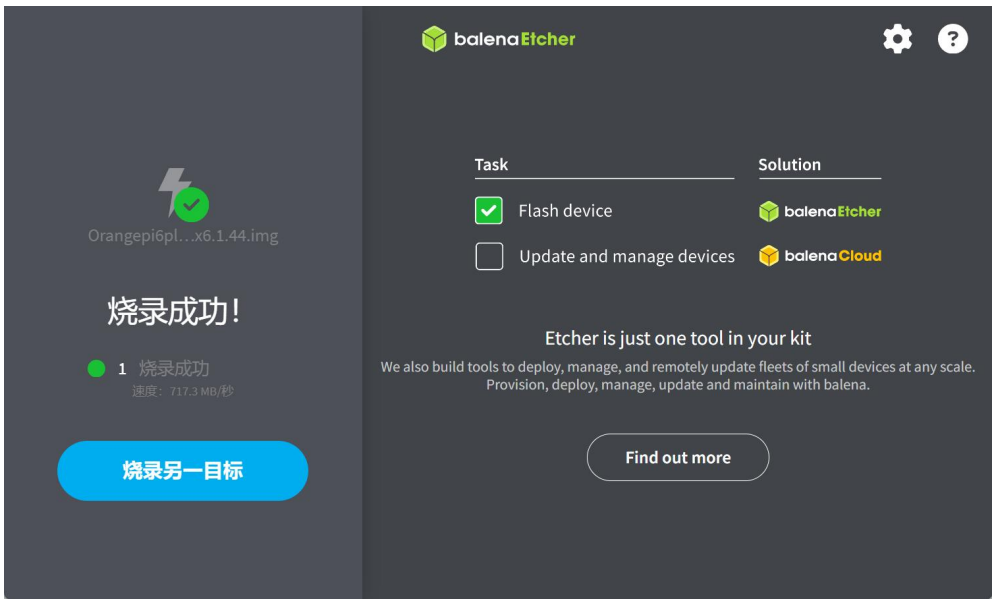


10) After the Linux image is burned, balenaEtcher will verify the image burned to the SSD disk by default to ensure that there are no problems in the burning process. As

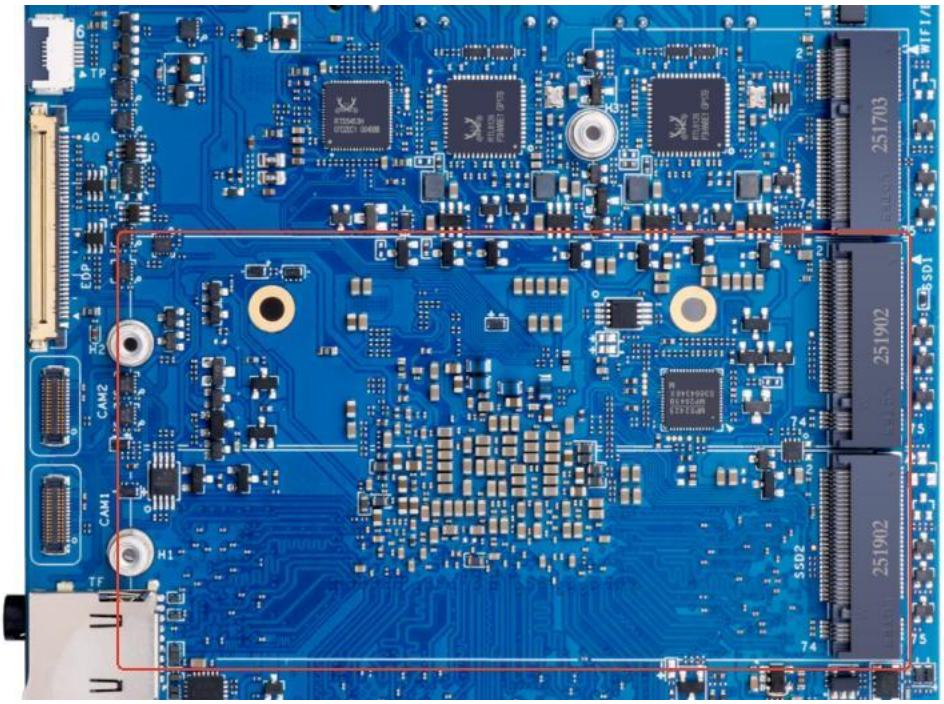
shown in the figure below, the green progress bar indicates that the image has been burned and balenaEtcher is verifying the burned image.



11) After the image is successfully burned, the balenaEtcher interface is as shown below. If a green indicator icon is displayed, it means that the image is burned successfully. You can exit balenaEtcher and then unplug the SSD disk and insert it into the SSD card slot of the development board for use.



12) The Orange Pi 6Plus SSD card slot is shown in the red box below. You can boot the device by inserting the SSD with the Android image burned into any card slot.



2. 4. How to burn the Android image to NVMe SSD disk based on Ubuntu PC

Note that the Linux image mentioned here specifically refers to the Linux distribution image such as Debian or Ubuntu downloaded from the Orange Pi data download page, and Ubuntu PC refers to a personal computer with the Ubuntu system installed.

To run the Android system on the development board, you must first flash the BIOS corresponding to the Android version. For specific steps, please refer to the opi-6plus-bios user manual.

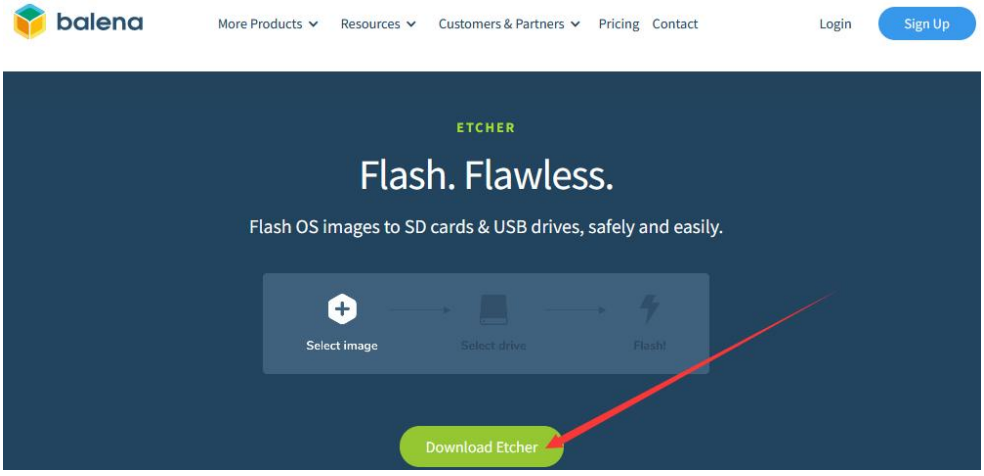
1) First, you need to prepare an M-Key 2280 NVMe SSD. The development board's M.2 slot PCIe interface is compatible with PCIe 3.0 and PCIe 4.0.

2) Use an NVMe SSD enclosure to connect the SSD to the Ubuntu PC.

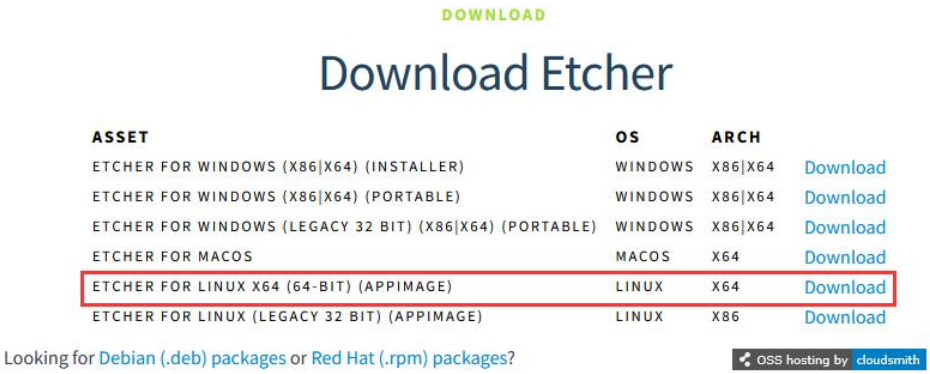
3) Download the balenaEtcher software from the download address:

https://www.balena.io/etcher/

4) After entering the balenaEtcher download page, click the green download button to jump to the software download location



5) Then choose to download the Linux version of the software



6) Download the compressed Linux operating system image file you want to burn from the Orange Pi download page, and then use decompression software to decompress it. Among the decompressed files, the file ending with ".img" is the operating system image file, which is generally over 1GB in size. The decompression command for the compressed file ending with .7z is as follows:

```
test@test:~$ 7z x Orangepi6plus_android_14v1.0.0_linux6.1.44.7z
test@test:~$ ls Orangepi6plus_android_14v1.0.0_linux6.1.44.*
Orangepi6plus_android_14v1.0.0_linux6.1.44.7z
Orangepi6plus_android_14v1.0.0_linux6.1.44.sha      #Checksum File
Orangepi6plus_android_14v1.0.0_linux6.1.44.img      #Image file
```

7) After decompressing the image, you can use the sha256sum -c *.sha command to calculate the checksum. If it is successful, it means that the downloaded image is correct and you can burn it to the SSD with confidence. If it prompts that the checksum does not match, it means that there is a problem with the downloaded image. Please try downloading it again.

```
test@test:~$ sha256sum -c *.sha
Orangepi6plus_android_14v1.0.0_linux6.1.44img: success
```

8) Then double-click balenaEtcher-1.14.3-x64.AppImage in the graphical interface of the Ubuntu PC to open balenaEtcher (no installation required). The interface of balenaEtcher after opening is as shown in the figure below

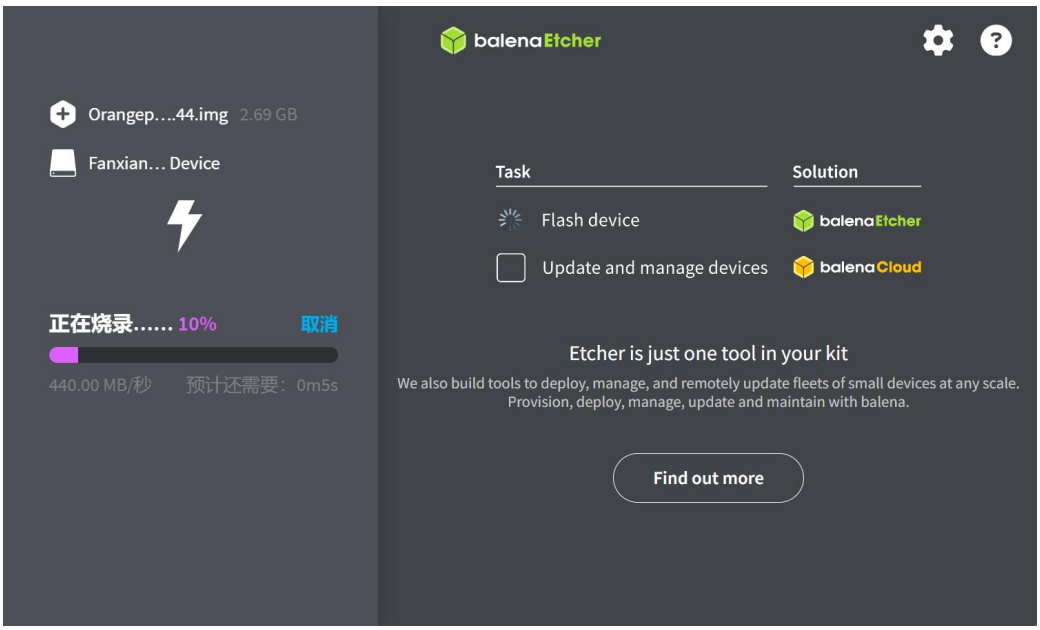
9) The specific steps for burning Android images using balenaEtcher are as follows

d. First select the path of the Android image file to be burned

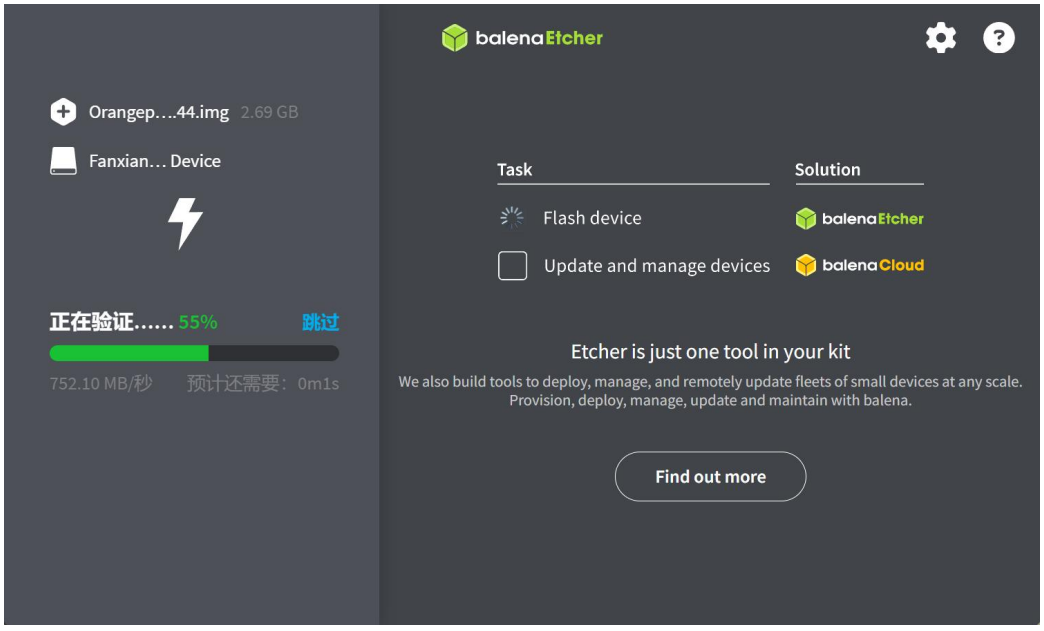
e. Then select the SSD drive letter. Be careful not to make the wrong choice. Do not select the system disk!

f. Finally, click Flash to start burning the Android image to the SSD disk.

10) The interface displayed by balenaEtcher during the burning of Android image is shown in the figure below. In addition, the progress bar is purple, indicating that the Android image is being burned to the SSD disk.

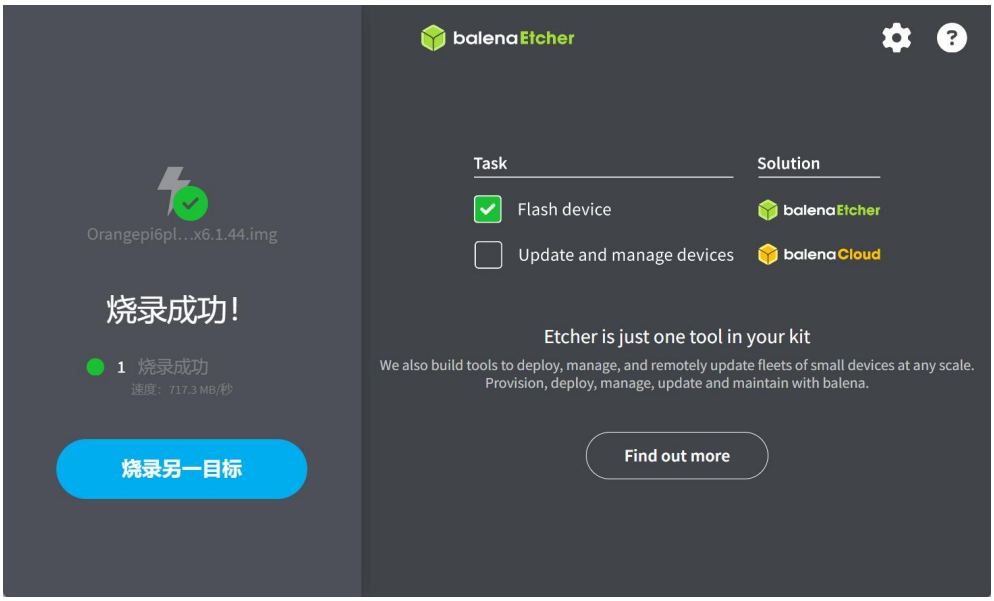


11) After the Linux image is burned, balenaEtcher will verify the image burned to the SSD disk by default to ensure that there are no problems in the burning process. As shown in the figure below, the green progress bar indicates that the image has been burned and balenaEtcher is verifying the burned image.

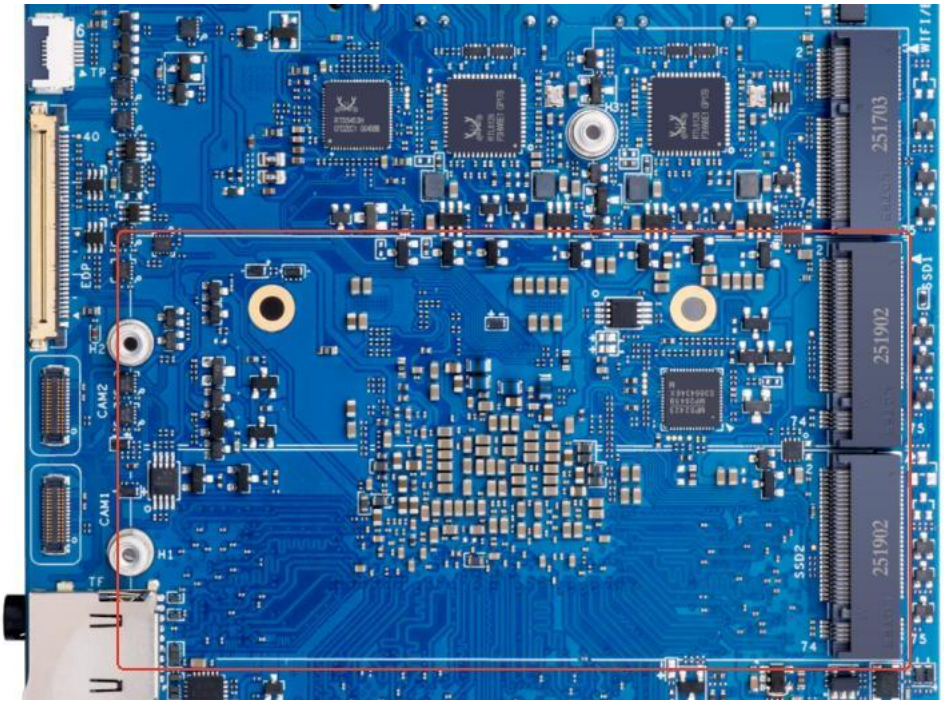


12) After the image is successfully burned, the balenaEtcher interface is as shown below. If a green indicator icon is displayed, it means that the image is burned successfully. You can exit balenaEtcher at this time, then unplug the SSD disk and insert it into the NVMe

SSD card slot of the development board for use.



13) The Orange Pi 6Plus SSD card slot is shown in the red box below. You can boot the device by inserting the SSD with the Android image burned into any card slot.



3. Android 14 system usage instructions

3. 1. Supported Android versions

Android Version	Kernel version
Android 14	**Linux6.1**

3. 2. Android 14 Functional Adaptation

Function	Android14
GPU	OK
NPU	OK
HDMI	OK
DP	OK
TYPE C DP display	OK
USB 2.0*2 host	OK
USB 3.0*2 host	OK
Network port*2	OK
EDP display	OK
Type-C power supply	OK
ADB	OK
NVME*2 boot	OK
USB Camera	OK

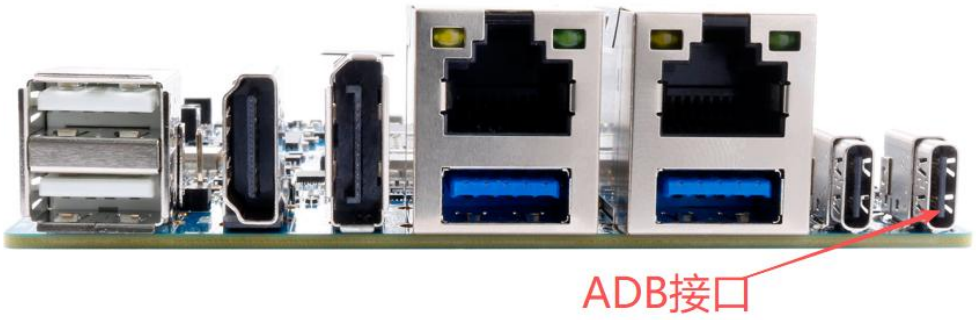
RTC time reading and writing	OK
Headphone jack for playback/recording	OK
Bluetooth audio playback	OK
power button	OK
RESET button	OK
Fastboot button	OK
SD card storage	OK
CAM1	OK
CAM2	OK
40PIN-GPIO	NG
40PIN-UART	OK
40PIN-I2C	OK
40PIN-SPI	OK
40PIN-PWM	OK
Onboard MIC, speaker	OK
Video Codec	OK

3. 3. How to use ADB

3. 3. 1. Use a data cable to connect adb debugging

1) First prepare a good quality TYPE-C data cable

2) Connect one end to the PC and the other end to the Orange Pi 6plus Type-C port



3) Install adb tool on Ubuntu PC

```
test@test:~$ sudo apt-get update
test@test:~$ sudo apt-get install -y adb
```

4) Check that the ADB device is identified

```
test@test:~$ adb devices
List of devices attached
4c00146473c28651dd0     device
```

5) Then you can log in to the Android system through adb shell on the Ubuntu PC

```
test@test:~$ adb shell
Orion_O6:/ $
```

3. 3. 2. Using adb debugging with network connection

Using network adb does not require a USB2.0 male-to-male data cable to connect the computer and the development board. Instead, communication is done over the network. So first make sure the wired or wireless network of the development board is connected, and then obtain the IP address of the development

board, which will be used later.

1) Ensure that the Android system's service.adb.tcp.port is set to port 5555

```
console:/ # getprop | grep "adb.tcp"
[service.adb.tcp.port]: [5555]
```

2) If service.adb.tcp.port is not set, you can use the following command in the serial port to set the network adb port number.

```
console:/ # setprop service.adb.tcp.port 5555
console:/ # stop adbd
console:/ # start adbd
```

3) Install adb tool on Ubuntu PC

```
test@test:~$ sudo apt-get update
test@test:~$ sudo apt-get install -y adb
```

4) Then connect to the network adb on the Ubuntu PC

```
test@test:~$ adb connect 192.168.1.xxx:5555 (Need to be changed to the IP address of the development board)
* daemon not running; starting now at tcp:5037
* daemon started successfully
connected to 192.168.1.xxx:5555

test@test:~$ adb devices
List of devices attached
192.168.1.xxx:5555 device
```

5) Then you can log in to the Android system through adb shell on the Ubuntu PC

```
test@test:~$ adb shell
Orion_O6:/ $
```

3. 4. ADB system directory mounting method for reading and writing

1) After ADB is successfully linked to the development version, open a new terminal and first obtain root permissions

```
test@test:~$ adb root
```

2) Turn off verification and set directories such as /system to be readable and writable

```
test@test:~$ adb remount
Successfully disabled verity
Failed to allocate scratch on /data, fallback to use free space on super
Using overlayfs for /system
Using overlayfs for /system_ext
Using overlayfs for /vendor
Using overlayfs for /product
Verity disabled; overlayfs enabled.
Now reboot your device for settings to take effect
```

3) It will take effect after restarting, restart the machine

```
test@test:~$ adb reboot
```

4) Wait for the machine to restart and ADB to connect again to obtain root permissions

```
test@test:~$ adb root
```

5) Execute again adb remount

```
test@test:~$ adb remount
Verity is already disabled
Remounted /system as RW
Remounted /system_ext as RW
Remounted /vendor as RW
Remounted /product as RW
Remount succeeded
```

6) The file directory in the Android system is now readable and writable, and files can be pushed to the system directory

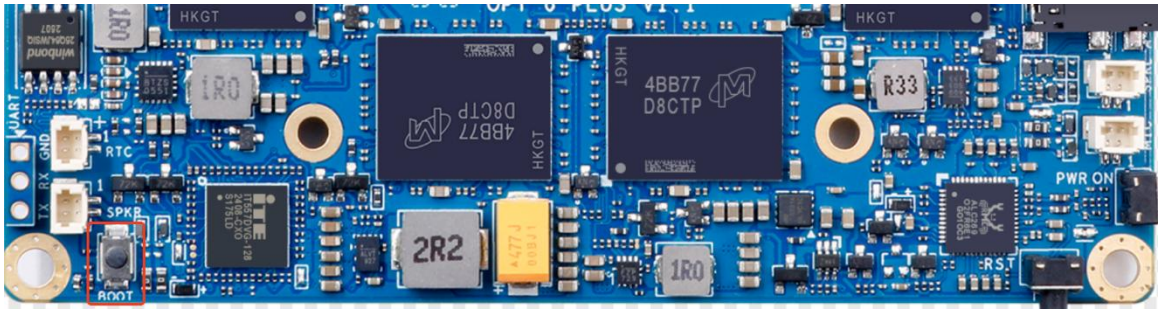
```
test@test:~$ adb push xxx/xxx/xx /system/
```

3. 5. How to enter Fastboot on Android

1) Install the fastboot tool on your Ubuntu PC

```
test@test:~$ sudo apt-get update
test@test:~$ sudo apt install android-tools-fastboot
```

2) Press and hold the Root key and press power on to start the device.



3) After entering fastboot, the serial port will have the following output, then release the Root key

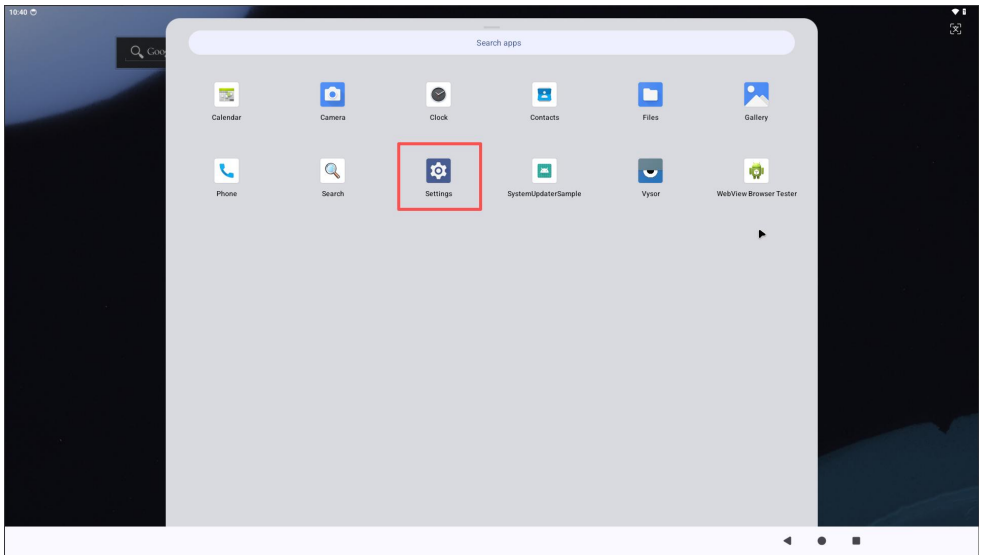
```
Tianocore/EDK2 firmware version BiosV1.0-W20251016-b758d1c5
Press ESCAPE for boot options
PhyDetectDevice: fail to detect ethernet PHY
I2C[5] hardware no response (ISR=0x00000004) during write!
I2C[5] Failed to process I2C message, status No Response
[02.300] [UEFI] E550 BmAfterConsole
Android Fastboot
Fastboot: Initializing...
Fastboot: Initializing done
```

4) Check fastboot devices

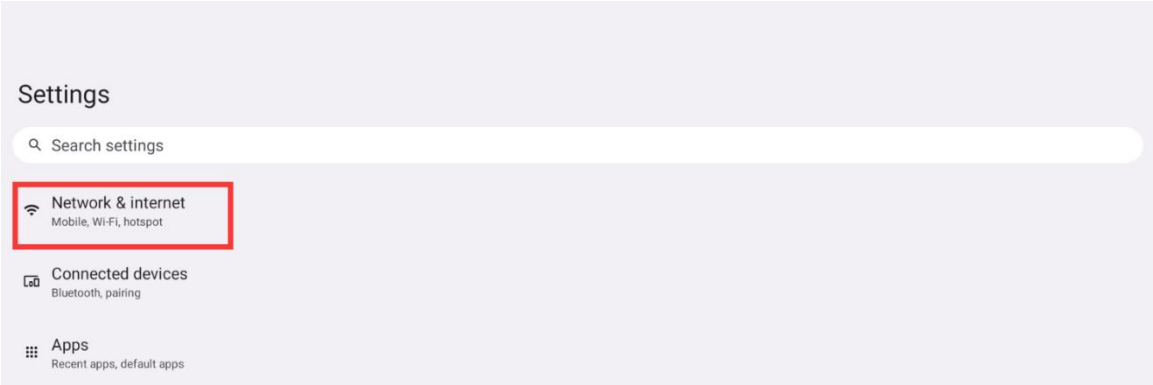
```
test@test:~$ fastboot devices
00500E0C0491CC91          fastboot
```

3. 6. Wi-Fi connection method

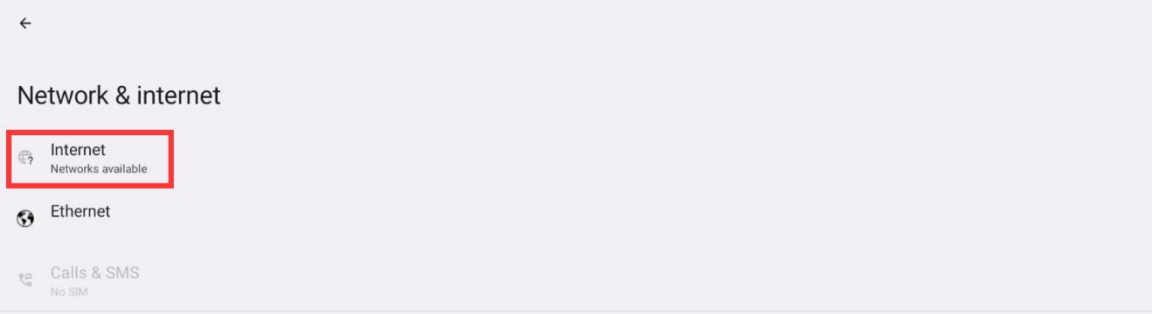
1) First select Settings



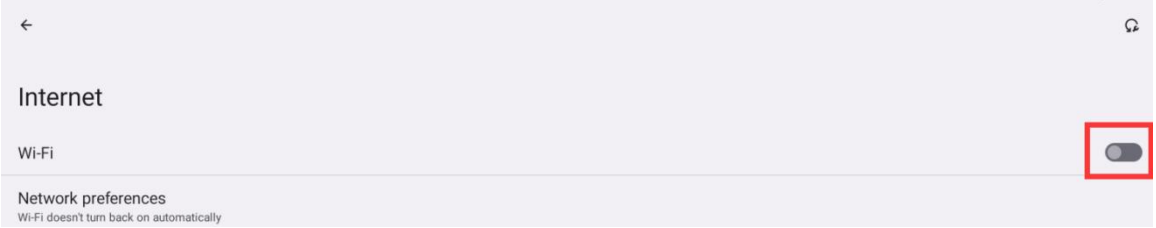
2) Then select Network & Internet



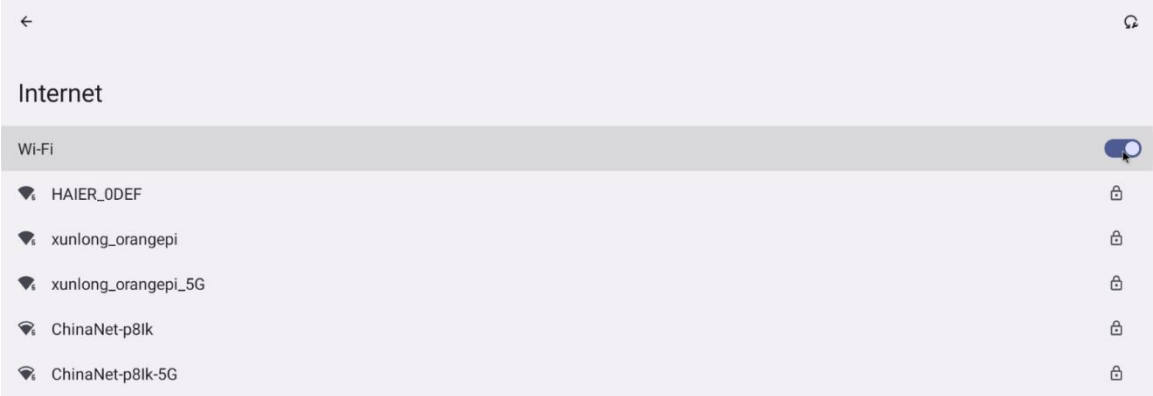
3) Then select Internet



4) Then open WI-FI

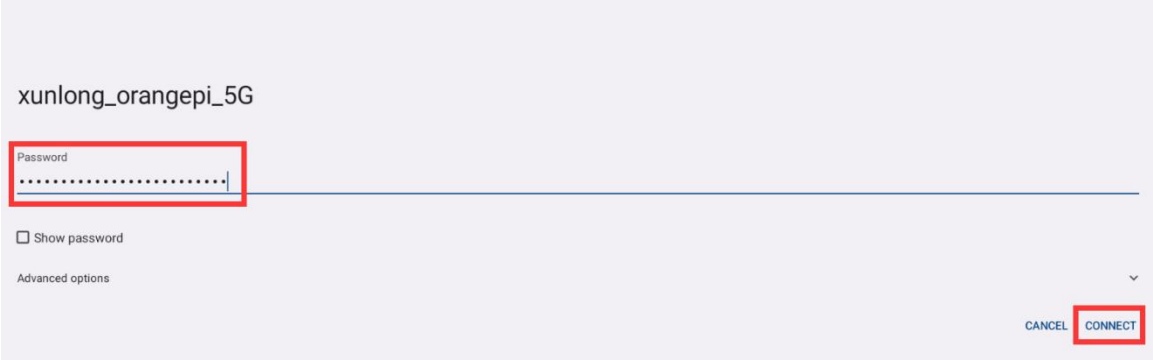


5) After turning on Wi-Fi, you can see the searched signal under Available networks

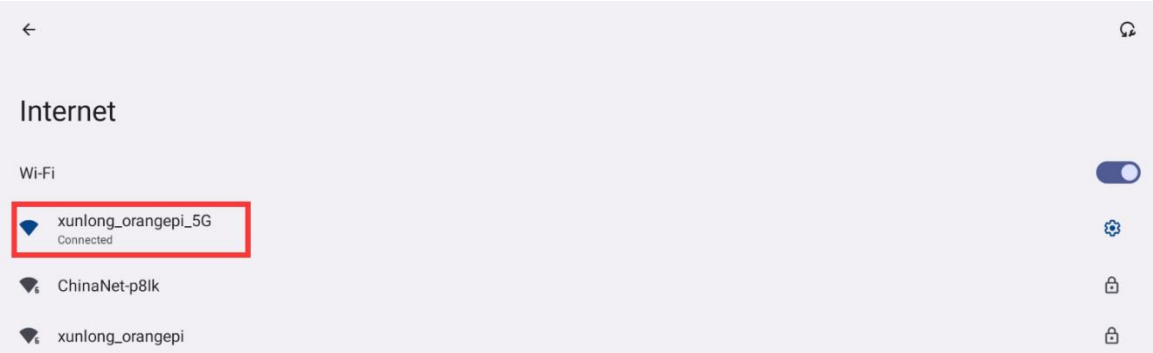


6) After selecting the WI-FI you want to connect to, the password input interface shown in the figure below will pop up

7) Then use the keyboard to enter the password corresponding to WI-FI, and then use the mouse to click the Enter button in the virtual keyboard to start connecting to WI-FI



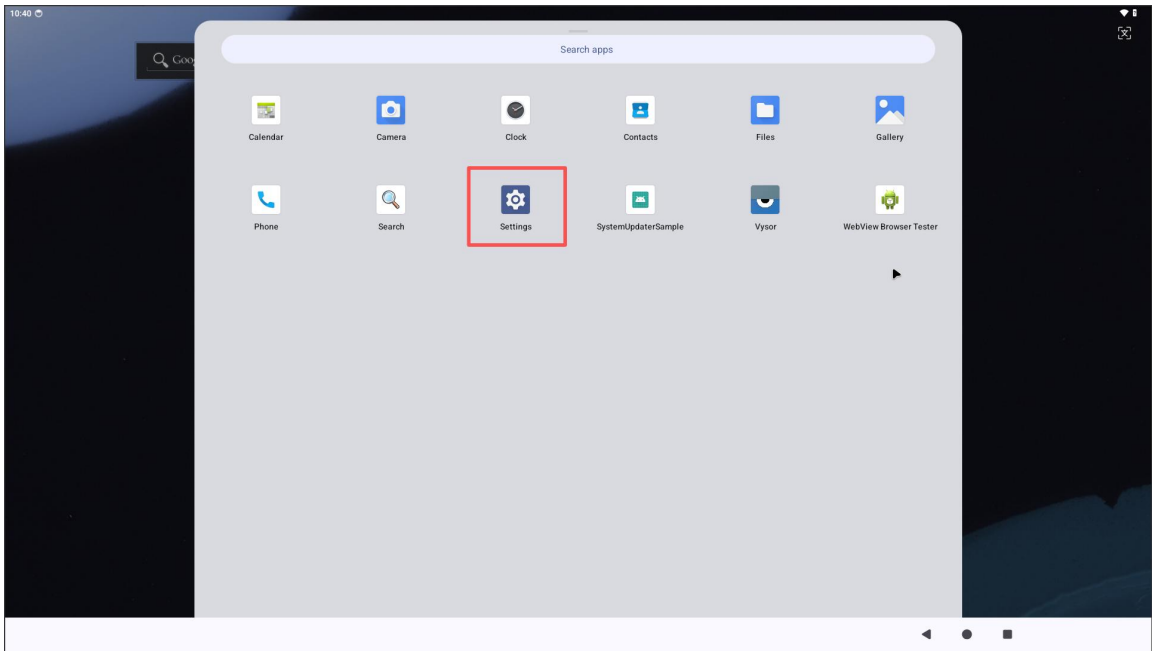
8) The display after successful WI-FI connection is as shown below



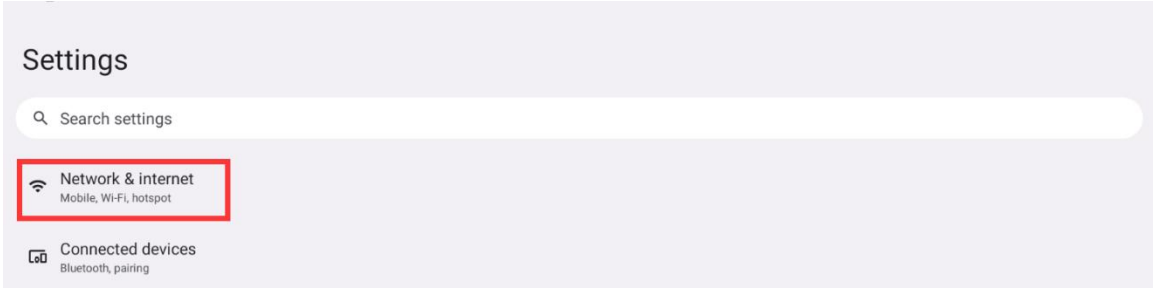
3. 7. How to use Wi-Fi hotspot

1) First, make sure the Ethernet port is connected to the network cable and can access the Internet normally.

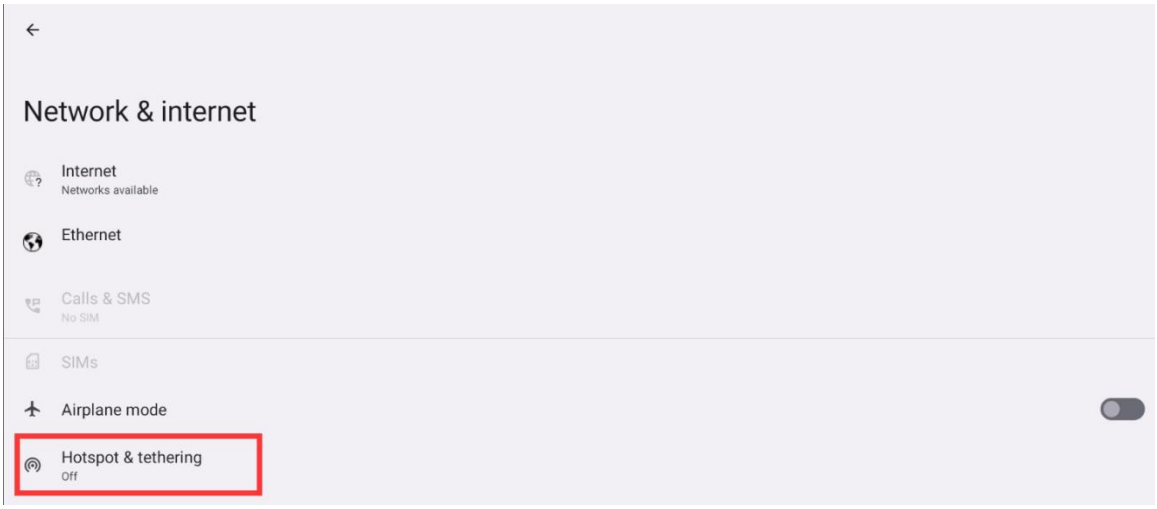
2) Then select **Settings**



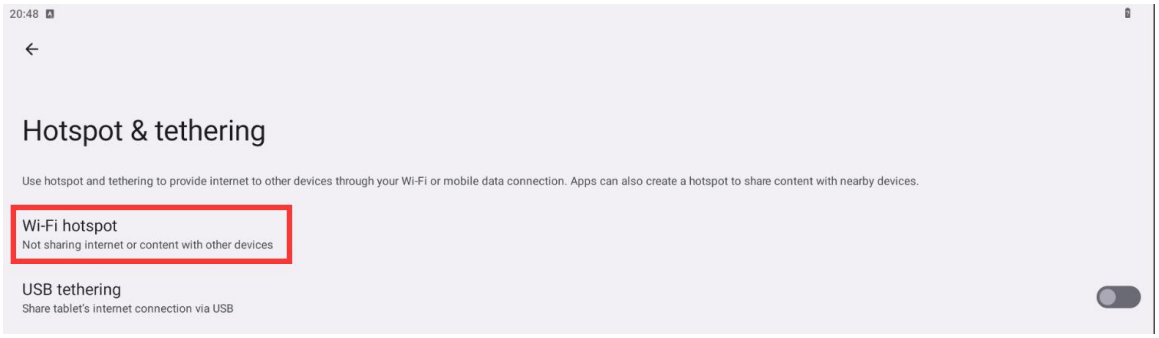
3) Then select **Network & Internet**



4) Then select **Hotspot & tethering**



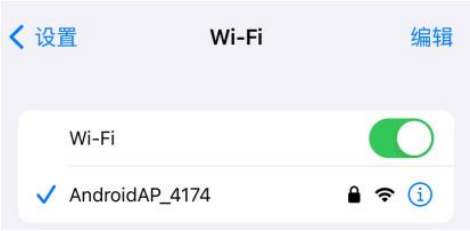
5) Then select **Wi-Fi hotspot**



6) Then turn on **Wi-Fi Hotspot**. You can also see the name and password of the generated hotspot in the picture below. Remember them and use them when connecting to the hotspot. (If you need to change the hotspot name and password, you need to turn off Wi-Fi Hotspot first.)

7) Now you can take out your phone. If everything is normal, you can find the WIFI hotspot with the same name (here it is AndroidAP_4174) shown under Hotspot name in the above picture in the WI-FI list searched by the phone. Then you can click AndroidAP_4174 to connect to the hotspot. The password can be seen under Hotspot password in the above picture.

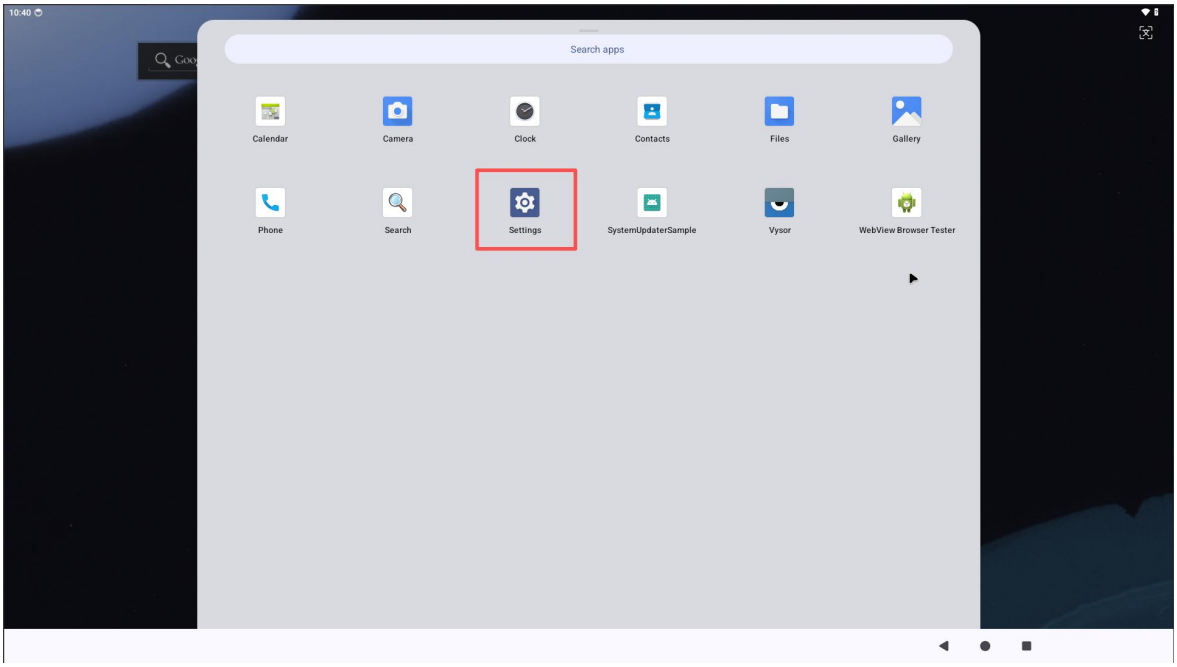
8) After the connection is successful, the following figure will be displayed (the interface may be different on different mobile phones, the specific interface is subject to the display of your mobile phone). Now you can open a web page on your mobile phone to see if you can access the Internet. If you can open the web page normally, it means that the Wi-Fi Hotspot of the development board is working properly.



3. 8. How to view the IP address of the Ethernet port

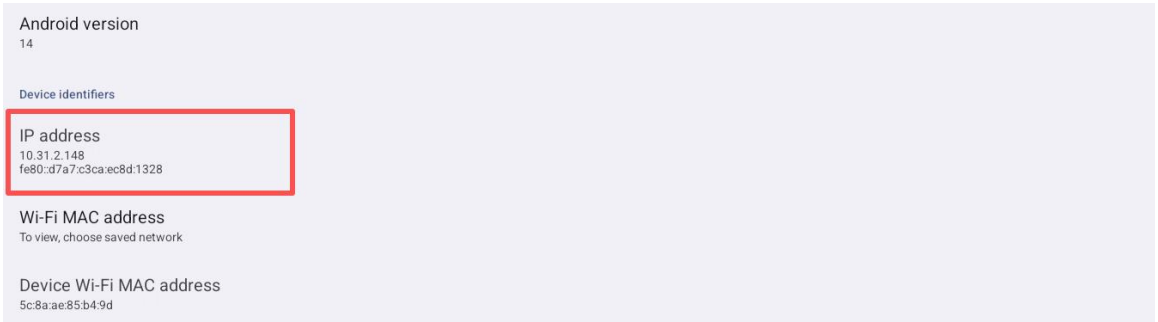
1) First, make sure the Gigabit Ethernet port of the development board is connected to a router or switch.

2) Then open **Settings**



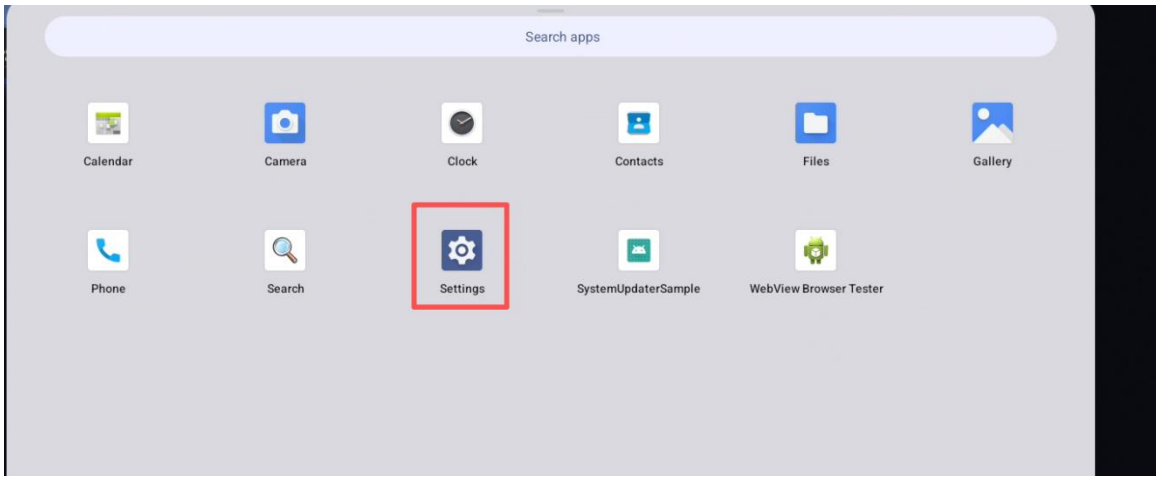
3) Then select **About tablet**

4) Then you can see the IP address information of the development board's wired network port

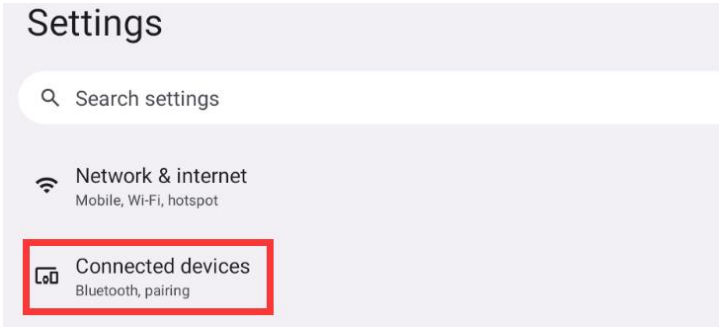


3. 9. Bluetooth connection method

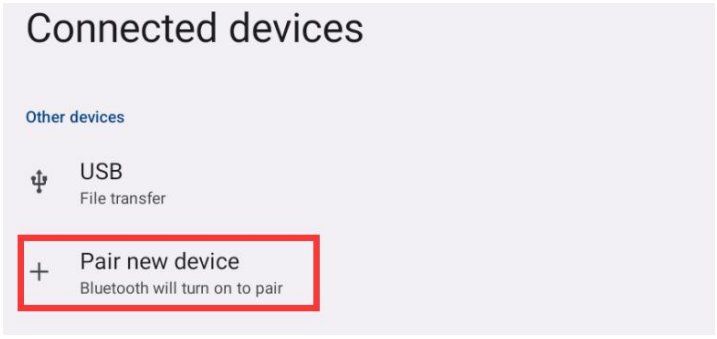
1) First select **Settings**



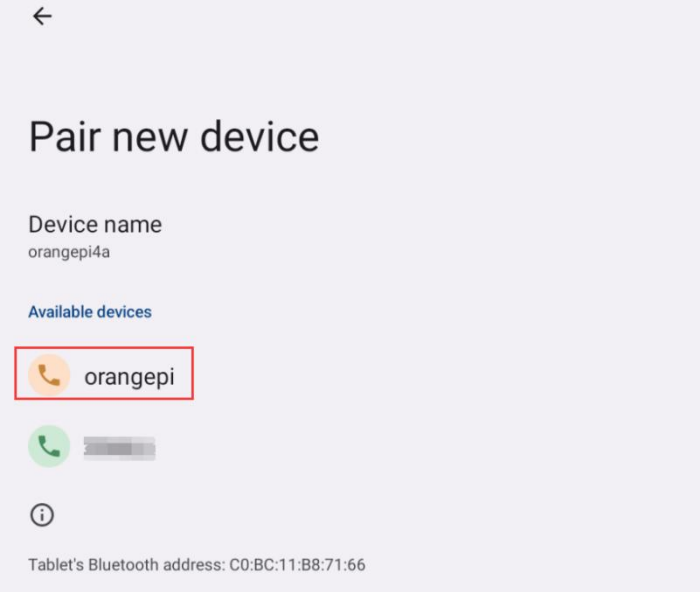
2) Then select **Connected devices**



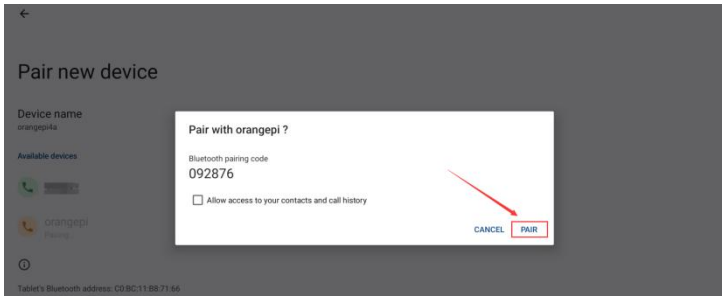
3) Then select **Pair new device** to start scanning for surrounding Bluetooth devices



4) The Bluetooth devices found will be displayed under **Available devices**

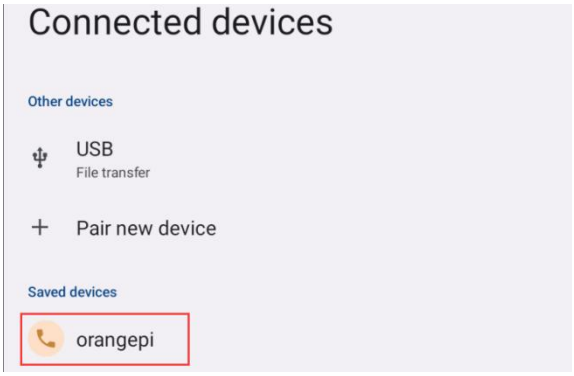


5) Then click on the Bluetooth device you want to connect to start pairing. When the following interface pops up, use the mouse to select the **Pair** option

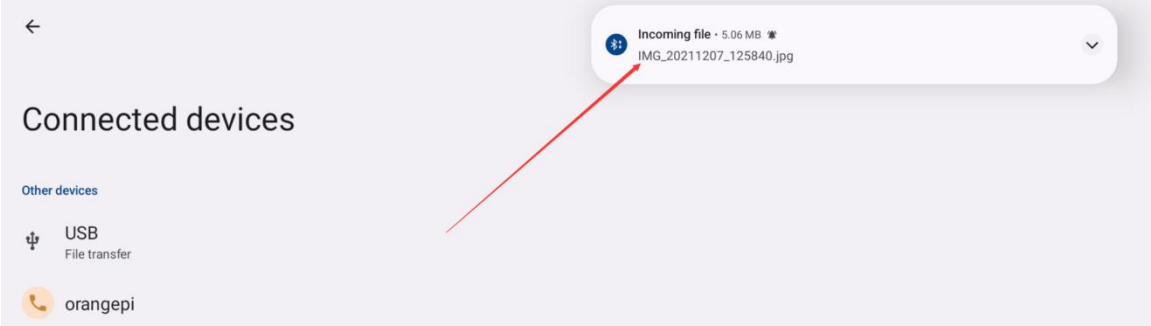


6) The test here is the configuration process of the Bluetooth of the development board and the **Android phone.** At this time, a confirmation interface will pop up on the phone. Click the pairing button on the phone to start the pairing process.

7) After pairing is complete, open **Paired devices** and you can see the paired Bluetooth devices.

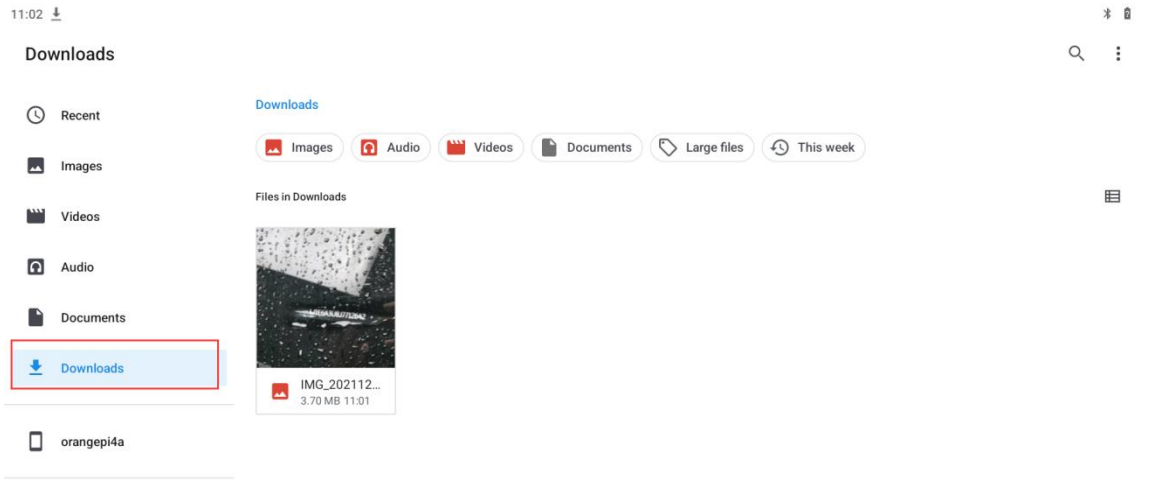


8) You can now use your phone's Bluetooth to send a picture to the development board. After sending, you will see the following prompt in the development board's Android system, then click **Incoming file**



9) Then click **Accept** in the pop-up window to start receiving pictures sent from your phone

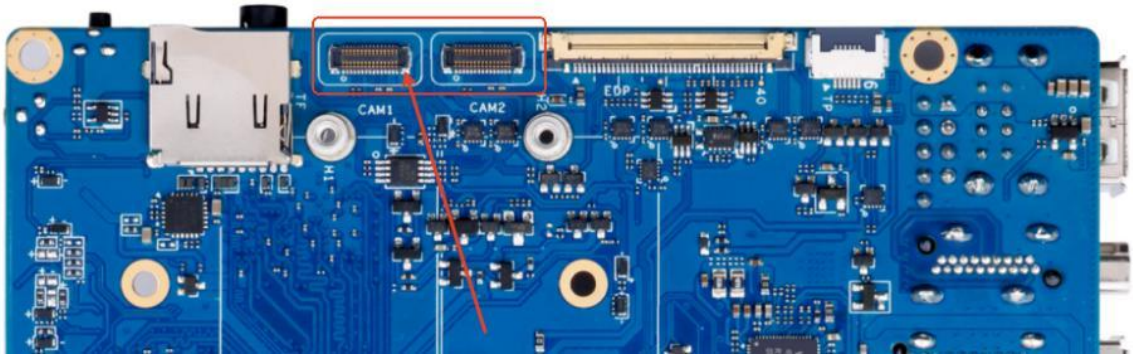
10) The pictures received by the Android system Bluetooth on the development board can be viewed by opening the **Download** directory in the file manager



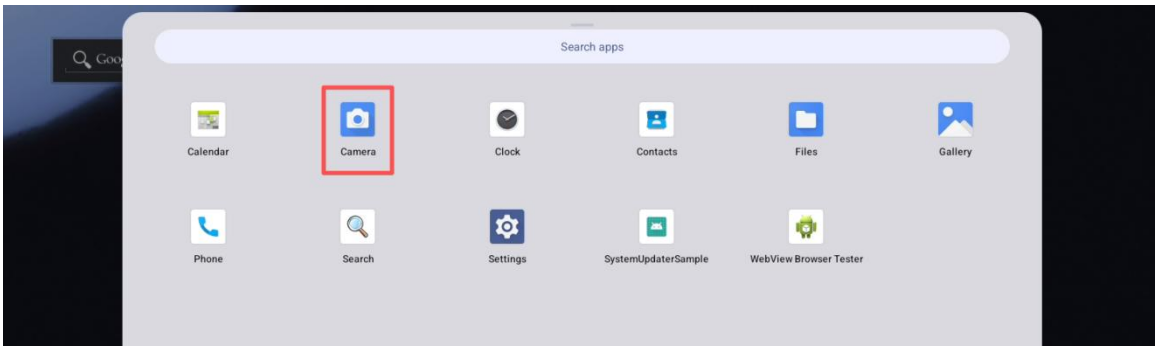
3. 10. Switching between front and rear cameras in CAM1 and CAM2 on Android

1) Currently the development board supports two MIPI cameras, OV13850 and OV13855

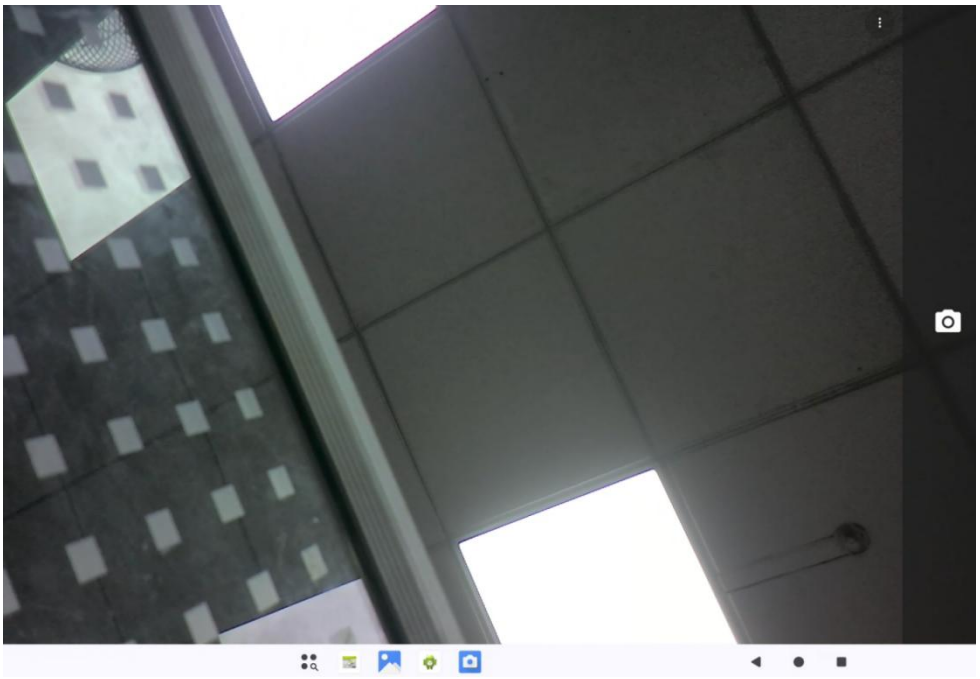
2) There are two camera interfaces on the development board, numbered CAM1 and CAM2, and their locations are as follows:



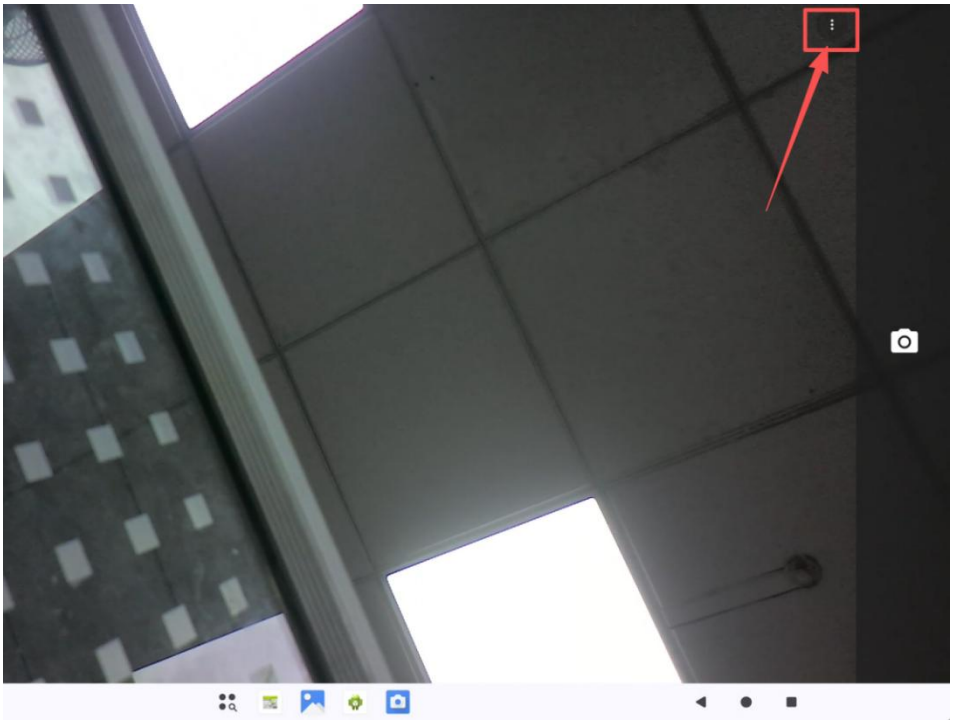
3) Connect the MIPI camera to the development board, turn on the board, and open the camera application



4) The camera preview screen is as follows



5) Click More for more pop-up options



6) Click the switch button to switch between CM1 and CAM2 cameras

3.11. 40 pin interface UART, I2C, PWM, SPI test

3.11.1. 40 Pin UART Test Method

1) The UARTs that can be used in the 40-pin of the development board are UART1 and UART3.

复用功能	复用功能	GPIO	GPIO序号	引脚序号	引脚序号	GPIO序号	GPIO	复用功能
		3.3V		1	2		5V	
	I2C2_SDA(04030000)	GPIO056	65	3	4		5V	
	I2C2_CLK	GPIO055	64	5	6		GND	
	FAN_OUT2(pwm5@04115000)	GPIO097	106	7	8	114	GPIO105	UART3_TXD(040e0000)
		GND		9	10	115	GPIO106	UART3_RXD
		GPIO015	32	11	12	34	GPIO017	
		GPIO016	33	13	14		GND	
UART1_TXD(040c0000)	FAN_OUT0(pwm3@04113000)	GPIO099	108	15	16	109	GPIO100	UART1_RXD
		3.3V		17	18	35	GPIO018	
	SPI1_MOSI	GPIO028	17	19	20		GND	
	SPI1_MISO	GPIO025	14	21	22	36	GPIO019	
	SPI1_CLK	GPIO029	18	23	24	15	GPIO026	SPI1_CS0
		GND		25	26	16	GPIO027	SPI1_CS1
	I2C5_SDA(04060000)	GPIO048	57	27	28	56	GPIO047	I2C5_CLK
		GPIO020	37	29	30		GND	
		GPIO023	40	31	32	38	GPIO021	
		GPIO024	41	33	34		GND	
		GPIO057	66	35	36	39	GPIO022	
		GPIO060	69	37	38	70	GPIO061	I2C4_CLK
		GND		39	40	71	GPIO062	I2C4_SDA(04050000)

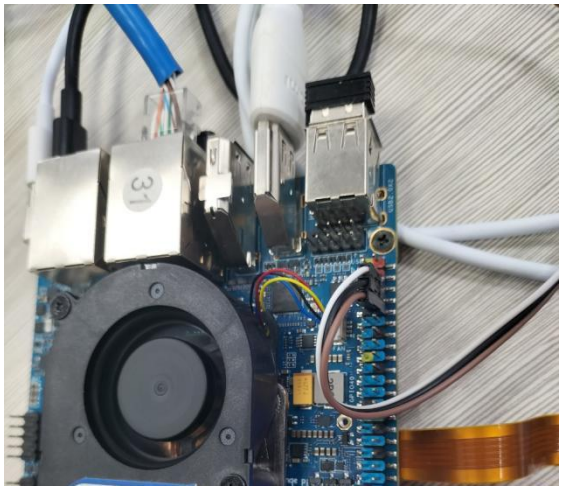
2) The device node corresponding to UART is as follows:

```
Orion_O6:/ # ls /sys/class/tty/ttyAMA* -lh

lrwxrwxrwx 1 root root 0 Apr 25 19:20 /sys/class/tty/ttyAMA1 -> ../../devices/platform/soc@0/40c0000.uart/tty/ttyAMA1

lrwxrwxrwx 1 root root 0 Apr 25 19:20 /sys/class/tty/ttyAMA2 -> ../../devices/platform/soc@0/40d0000.uart/tty/ttyAMA2

lrwxrwxrwx 1 root root 0 Apr 25 19:20 /sys/class/tty/ttyAMA3 -> ../../devices/platform/soc@0/40e0000.uart/tty/ttyAMA3
```

3) Then short-circuit the TX and RX terminals of the UART to test the loopback.

a. Take ttyAMA3 as an example

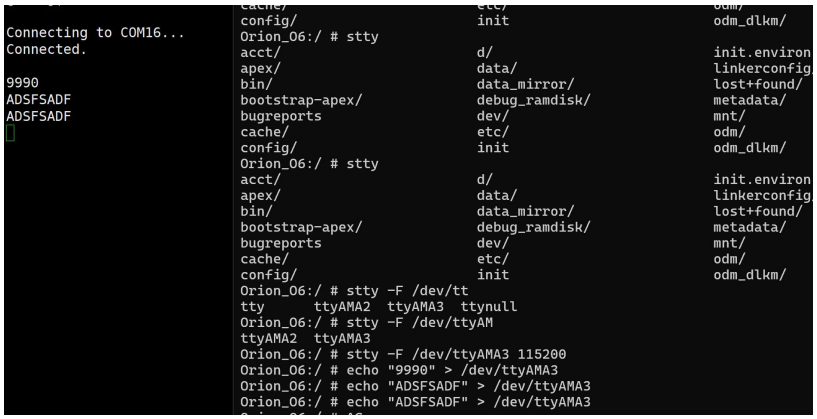
b. Use the serial cable TX, RX to connect to PIN 8, PIN 10, the black ground wire, and the other end of the serial USB port to connect to the PC, as shown below:



c. Use the command in the terminal to set the baud rate to 115200

stty -F /dev/ttyAMA3 115200

d. Pass a value to ttyAMA3 in the terminal, and the PC serial port returns the same content



3. 11. 2. 40 Pin I2C Test Method

1) The I2C interfaces available on the 40-pin development board include I2C2, I2C4, and I2C5.

复用功能	复用功能	GPIO	GPIO序号	引脚序号	引脚序号	GPIO序号	GPIO	复用功能
		3.3V		1	2		5V	
	I2C2_SDA(04030000)	GPIO056	65	3	4		5V	
	I2C2_CLK	GPIO055	64	5	6		GND	
	FAN_OUT2(pwm5@04115000)	GPIO097	106	7	8	114	GPIO105	UART3_TXD(040e0000)
		GND		9	10	115	GPIO106	UART3_RXD
		GPIO015	32	11	12	34	GPIO017	
		GPIO016	33	13	14		GND	
UART1_TXD(040c0000)	FAN_OUT0(pwm3@04113000)	GPIO099	108	15	16	109	GPIO100	UART1_RXD
		3.3V		17	18	35	GPIO018	
	SPI1_MOSI	GPIO028	17	19	20		GND	
	SPI1_MISO	GPIO025	14	21	22	36	GPIO019	
	SPI1_CLK	GPIO029	18	23	24	15	GPIO026	SPI1_CS0
		GND		25	26	16	GPIO027	SPI1_CS1
	I2C5_SDA(04060000)	GPIO048	57	27	28	56	GPIO047	I2C5_CLK
		GPIO020	37	29	30		GND	
		GPIO023	40	31	32	38	GPIO021	
		GPIO024	41	33	34		GND	
		GPIO057	66	35	36	39	GPIO022	
		GPIO060	69	37	38	70	GPIO061	I2C4_CLK
		GND		39	40	71	GPIO062	I2C4_SDA(04050000)

2) The i2c devices viewed by the i2cdetect -l command are as follows:

```
Orion_O6:~# i2cdetect -l
i2c-0   i2c     Cadence I2C at 04010000         I2C adapter
i2c-1   i2c     Cadence I2C at 04020000         I2C adapter
i2c-2   i2c     Cadence I2C at 04030000         I2C adapter
i2c-3   i2c     Cadence I2C at 04040000         I2C adapter
i2c-4   i2c     Cadence I2C at 04050000         I2C adapter
i2c-5   i2c     Cadence I2C at 04060000         I2C adapter
```

3) Connect the sensor to the I2C interface and make sure the address can be scanned normally.

```
root@root:~# i2cdetect -r -y 2
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                         -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- 38 -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
root@root:~#
```

```
root@root:~# i2cdetect -r -y 5
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:                         -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
20: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- 38 -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

3. 11. 3. 40-pin PWM test method

1) The PWMs that can be used in the 40-pin development board are PWM3 corresponding to pin 15 and PWM5 corresponding to pin 7.

复用功能	复用功能	GPIO	GPIO序号	引脚序号	引脚序号	GPIO序号	GPIO	复用功能
		3.3V		1	2		5V	
	I2C2_SDA(04030000)	GPIO056	65	3	4		5V	
	I2C2_CLK	GPIO055	64	5	6		GND	
	FAN_OUT2(pwm5@04115000)	GPIO097	106	7	8	114	GPIO105	UART3_TXD(040e0000)
		GND		9	10	115	GPIO106	UART3_RXD
		GPIO015	32	11	12	34	GPIO017	
		GPIO016	33	13	14		GND	
UART1_TXD(040c0000)	FAN_OUT0(pwm3@04113000)	GPIO099	108	15	16	109	GPIO100	UART1_RXD
		3.3V		17	18	35	GPIO018	
	SPI1_MOSI	GPIO028	17	19	20		GND	
	SPI1_MISO	GPIO025	14	21	22	36	GPIO019	
	SPI1_CLK	GPIO029	18	23	24	15	GPIO026	SPI1_CS0
		GND		25	26	16	GPIO027	SPI1_CS1
	I2C5_SDA(04060000)	GPIO048	57	27	28	56	GPIO047	I2C5_CLK
		GPIO020	37	29	30		GND	
		GPIO023	40	31	32	38	GPIO021	
		GPIO024	41	33	34		GND	
		GPIO057	66	35	36	39	GPIO022	
		GPIO060	69	37	38	70	GPIO061	I2C4_CLK
		GND		39	40	71	GPIO062	I2C4_SDA(04050000)

2) The address corresponding to PWM3 in pin 40 of the development board is 4113000, and the address corresponding to PWM5 is 4115000.

```
root@root:/sys/class/pwm# ls -lh
total 0
lrwxrwxrwx 1 root root 0 Aug  5 14:51 pwmchip0 → ../../devices/platform/soc@0/4110000.pwm0/pwm/pwmchip0
lrwxrwxrwx 1 root root 0 Aug  5 14:51 pwmchip1 → ../../devices/platform/soc@0/4113000.pwm3/pwm/pwmchip1
lrwxrwxrwx 1 root root 0 Aug  5 14:51 pwmchip2 → ../../devices/platform/soc@0/4115000.pwm5/pwm/pwmchip2
lrwxrwxrwx 1 root root 0 Aug  5 14:51 pwmchip3 → ../../devices/platform/soc@0/4070000.i2c/i2c-6/6-0076/4070000.i2c:ec@76:ec-pwm/pwm/pwmchip3
```

3) The command to set the PWM output waveform is as follows：

a. The pwm3 test command is as follows：

```
Orion_O6:/ # cd /sys/class/pwm/pwmchip1
Orion_O6:/sys/class/pwm/pwmchip1# echo 0 > export
Orion_O6:/sys/class/pwm/pwmchip1# echo 100000 > pwm0/period
Orion_O6:/sys/class/pwm/pwmchip1# echo 50000 > pwm0/duty_cycle
```

```
Orion_O6:/sys/class/pwm/pwmchip1# echo 1 > pwm0/enable
```

b. The pwm5 test command is as follows：

```
Orion_O6:~# cd /sys/class/pwm/pwmchip2
Orion_O6:/sys/class/pwm/pwmchip2# echo 0 > export
Orion_O6:/sys/class/pwm/pwmchip2# echo 100000 > pwm0/period
Orion_O6:/sys/class/pwm/pwmchip2# echo 50000 > pwm0/duty_cycle
Orion_O6:/sys/class/pwm/pwmchip2# echo 1 > pwm0/enable
```

3. 11. 4. 40-pin SPI test method

1) Among the 40 pins on the development board, the available SPIs are SPI1.

复用功能	复用功能	GPIO	GPIO序号	引脚序号	引脚序号	GPIO序号	GPIO	复用功能
		3.3V		1	2		5V	
	I2C2_SDA(04030000)	GPIO056	65	3	4		5V	
	I2C2_CLK	GPIO055	64	5	6		GND	
	FAN_OUT2(pwm5@04115000)	GPIO097	106	7	8	114	GPIO105	UART3_TXD(040e0000)
		GND		9	10	115	GPIO106	UART3_RXD
		GPIO015	32	11	12	34	GPIO017	
		GPIO016	33	13	14		GND	
UART1_TXD(040c0000)	FAN_OUT0(pwm3@04113000)	GPIO099	108	15	16	109	GPIO100	UART1_RXD
		3.3V		17	18	35	GPIO018	
	SPI1_MOSI	GPIO028	17	19	20		GND	
	SPI1_MISO	GPIO025	14	21	22	36	GPIO019	
	SPI1_CLK	GPIO029	18	23	24	15	GPIO026	SPI1_CS0
		GND		25	26	16	GPIO027	SPI1_CS1
	I2C5_SDA(04060000)	GPIO048	57	27	28	56	GPIO047	I2C5_CLK
		GPIO020	37	29	30		GND	
		GPIO023	40	31	32	38	GPIO021	
		GPIO024	41	33	34		GND	
		GPIO057	66	35	36	39	GPIO022	
		GPIO060	69	37	38	70	GPIO061	I2C4_CLK
		GND		39	40	71	GPIO062	I2C4_SDA(04050000)

2) After the system starts, you can see the two SPI device nodes below.

```
Orion_O6:~# ls /dev/spidev0.*
/dev/spidev0.0  /dev/spidev0.1
```

3) Then use Dupont wire to short-circuit pin 19 and 21.

4) Use the spidev_test command to test the loopback.

```
Orion_O6:~# spidev_test -v -D /dev/spidev0.0
Orion_O6:~# spidev_test -v -D /dev/spidev0.1
```

4. Compiling method of Android 14 source code

4. 1. Download the source code of Android 14

1) First download the Android 14 source code volume compressed package from Baidu or Google Drive

2) After downloading the compressed package of the Android 14 source code, please check whether the MD5 checksum is correct. If it is not correct, please download the source code again. The method to check the MD5 checksum is as follows:

```
test@test:~$ md5sum -c md5sum
6plus_android14.tar.gz00: OK
6plus_android14.tar.gz01: OK
.....
```

3) Then execute the following command to decompress the Android source code

```
test@test:~$ cat 6plus_android14.tar.gz0* | tar -xvzf -
```

4. 2. Compile the source code of Android 14

Android 13 is compiled on an x86_64 computer with Ubuntu 22.04 installed. The system package dependencies of other versions of Ubuntu may be slightly

different. The image download address of the Ubuntu 22.04 amd64 version is as follows:

https://repo.huaweicloud.com/ubuntu-releases/22.04/ubuntu-22.04.2-desktop-amd64.iso

The recommended hardware configuration for an x86_64 computer for compiling the Android 13 source code is 16GB or more of memory, 200GB or more of hard disk space, and the more CPU cores the better.

1) First install the software package required to compile the Android 14 source code

```
test@test:~$ sudo apt-get update
test@test:~$ sudo apt-get -y install lsb-release autoconf autopoint bc \
        bison build-essential cpiocurl device-tree-compiler \
        dosfstools doxygen fdisk flex gdisk gettext-base git \
        libncurses5 libssl-dev libtinfo5 m4 mtools pkg-config \
        python2 python3 python3-distutils python3-pyelftools \
        rsync snapd unzip uuid-dev wget scons perl libwayland-dev \
        wayland-protocols indent libtool dwarves libarchive-tools \
        xorriso jigdo-file python3-pip vim sudo parted cmake \
        golang libffi-dev u-boot-tools img2simg libxcb-randr0 \
        libxcb-randr0-dev libxcb-present-dev libxau-dev \
        python3-mako libglib2.0-dev-bin binfmt-support qemu \
        qemu-user-static debootstrap multistrap debian-archive-keyring \
        ser2net git-lfs zstd debhelper jq pigz zip
```

2) Start compiling

```
test@test:~/android14# source build/envsetup.sh
test@test:~/android14# lunch sky1_evb-ap2a-userdebug
test@test:~/android14# /build-android.sh -j8
```

3) The compiled product is located in the ./out/target/product/sky1_evb/images/ directory

file name	illustrate
android.sdcard	Android 14 firmware package (can be burned using DD or third-party software)
android_flush_images.bat	Android image windows burning script (fastboot burning method)
android_flush_images.sh	Android image Linux burning script (fastboot burning method)
cix_flash_all_rsa_pr.bin	Android firmware
boot.img	Android Boot Image
init_boot.img	Android init boot image
vendor_boot.img	Android vendor boot image
dtbo-sky1.img	Android dtb image
super.img	Android super mirror
LinuxLoader.efi.cap	Android second loader image
partition-table-default-4096.img	Android 4096 bytes block size GPT image
partition-table-default.img	Android 512 bytes block size GPT image (used by default)

5. Appendix

5. 1. User Manual Update History

Version	date	Update Notes
V1.0	2025-10-17	Added Android flashing and Android 14 usage instructions

5. 2. Image Update History

date	Update Notes
2025-10-17	Orangepi6plus_1.0.0_android14_linux6.1.44..7z