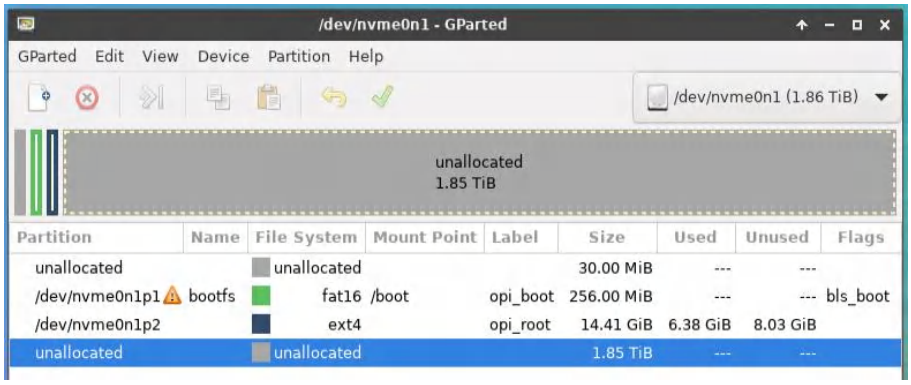
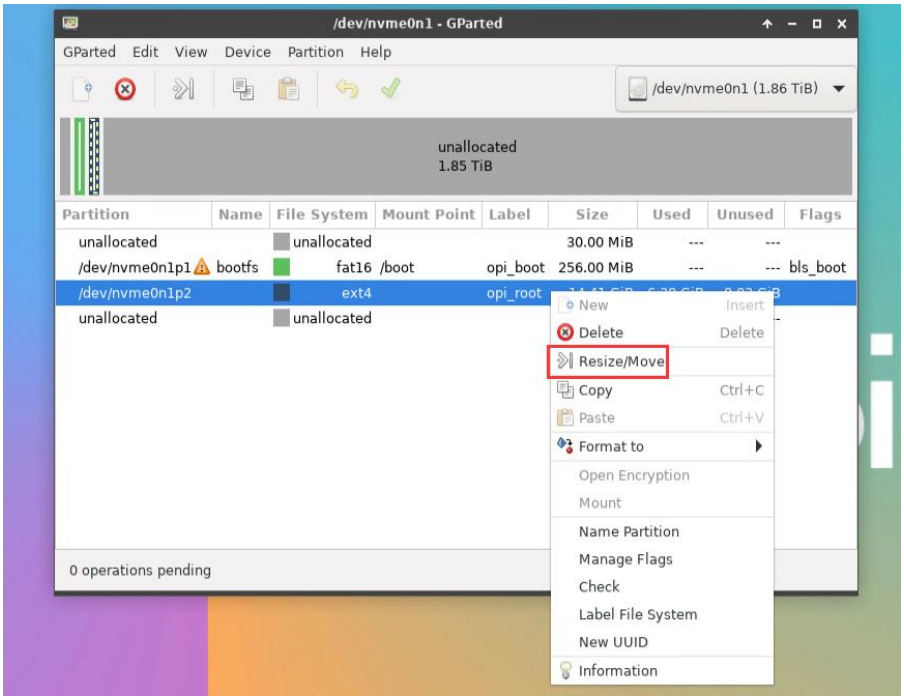
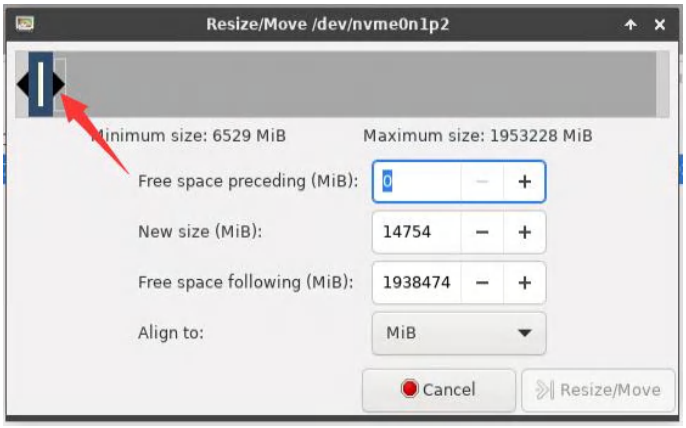


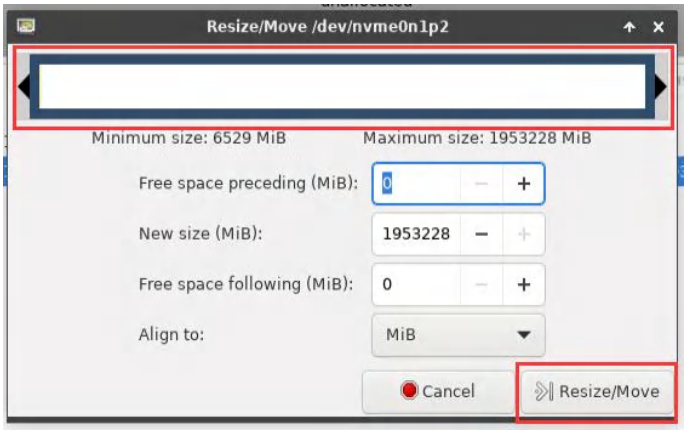

f) Then select the **/dev/nvme0n1p2** partition, click the right button again, and then select **Resize/Move**



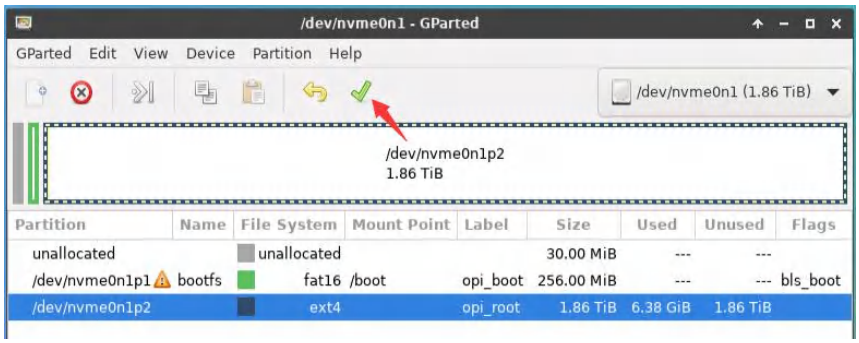

g) Then drag the capacity to the maximum at the position shown in the figure below



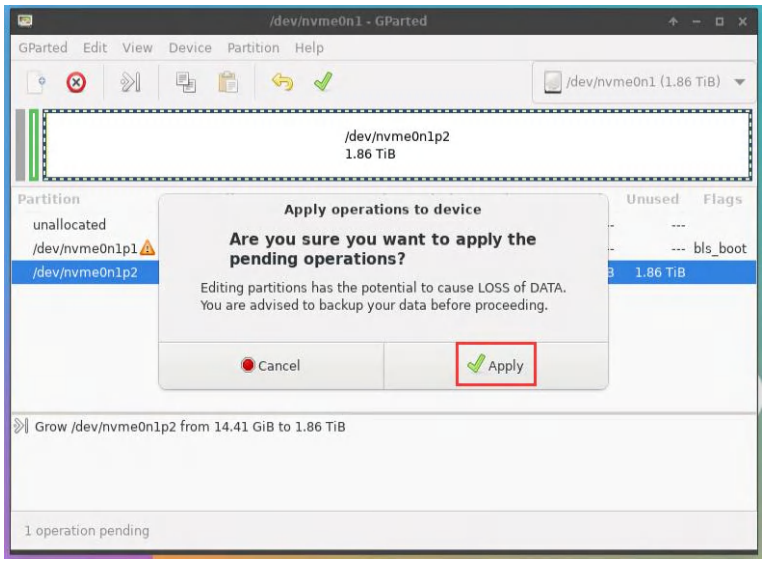
h) Then click **Resize/Move**



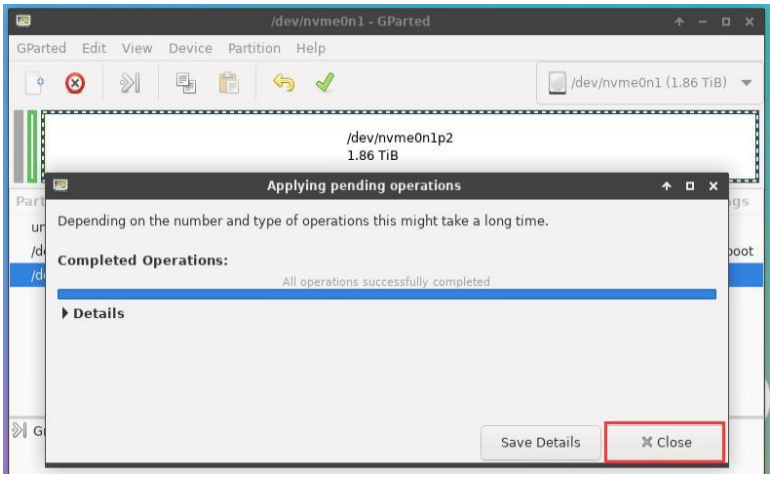

i) Then click the green √ in the position below




j) Then click **Apply**




k) Then click **Close** to close

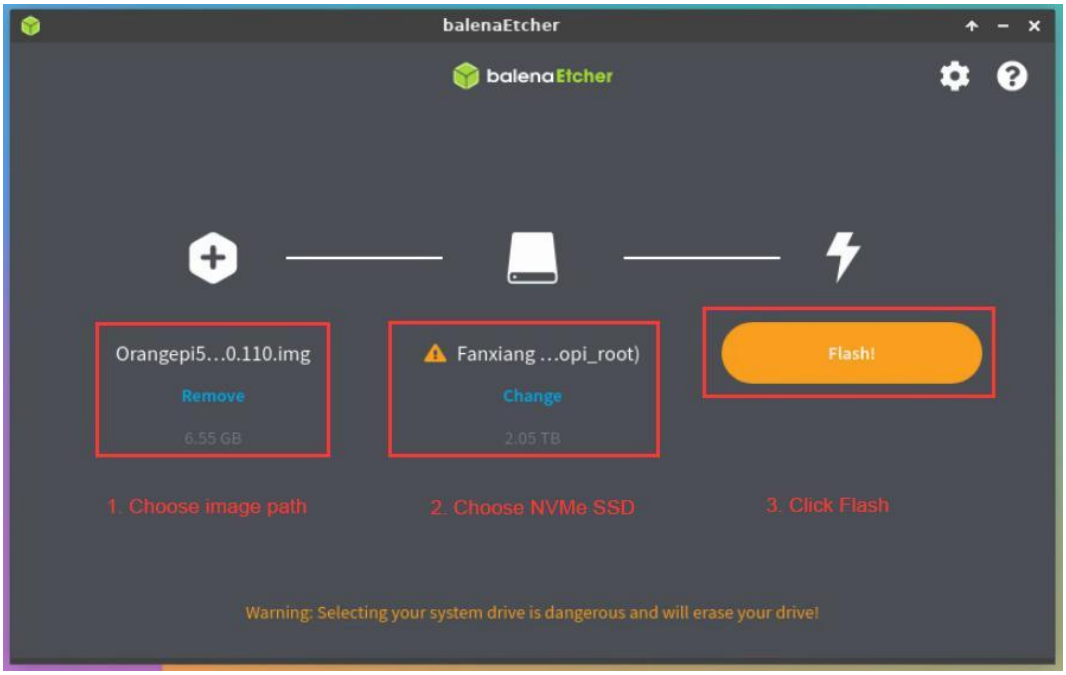



m. At this point, you can use the **sudo poweroff** command to shut down. Then please pull out the TF card, and then short press the power button to turn on, then the Linux system in SPIFlash+NVMe SSD will be started.

10) Step **9)** is to clone the system in the TF card to the NMVe SSD. We can also directly burn the Linux image file to the NVMe SSD. Here are the steps:

a. Upload the Linux image file to the Linux system of the development board

b. Then use balenaEtcher to burn




c. **After using this method to burn the image, there is no need to manually expand the capacity, and it will automatically expand the capacity at the first startup.**

# 2. 7. How to burn Android image to TF card

## 2. 7. 1. Method of burning Android image to TF card through USB2.0 burning port

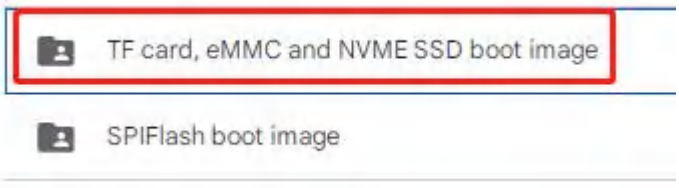
1) First prepare a TF card with 8GB or larger capacity. The transmission speed of the TF card must be class10 or above. It is recommended to use a TF card of SanDisk and other brands

2) You also need to prepare a good quality USB2.0 male-to-male data cable

3) Then download Rockchip driver **DriverAssitant_v5.12.zip** and burning tool **RKDevTool_Release_v3.15.zip** from **Orange Pi's data download page**

4) Then download the Android image from **Orange Pi's download page**.

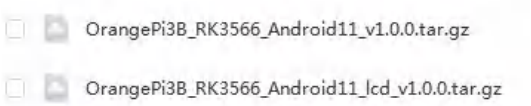
a. After opening the download link of the Android image, you can see the following two types of Android images, please select the image in the **TF card and eMMC startup image** folder to download




b. After entering the **TF card and eMMC startup image** folder, you can see the following two images, the difference between them is:

a) The first image is dedicated to HDMI display and supports 4K display. If you don't use LCD screen, please download the image without lcd

b) If you want to use lcd screen, please choose image with lcd



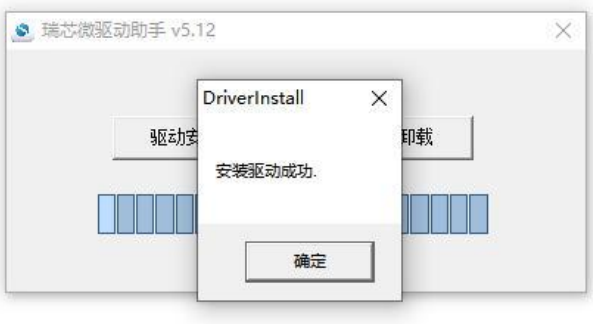

5) Then use the decompression software to decompress the compressed package of the downloaded Android image. Among the decompressed files, the file ending with ".img" is the Android image file, and the size is more than 1GB

6) Then use the decompression software to decompress **DriverAssitant_v5.12.zip**, and then find the **DriverInstall.exe** executable file in the decompressed folder and open it


7) After opening **DriverInstall.exe**, the steps to install the Rockchip driver are as follows

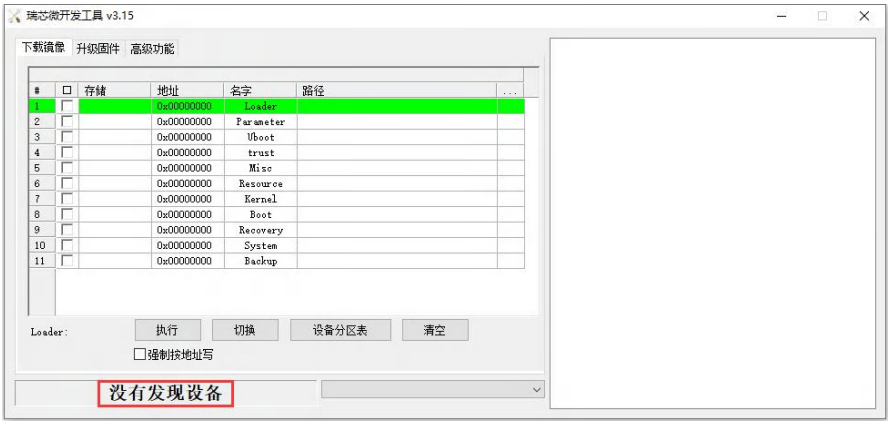
a. Click the "**Driver Installation**" button


b. After waiting for a period of time, a pop-up window will prompt "**The driver is installed successfully**", and then click the "**OK**" button.



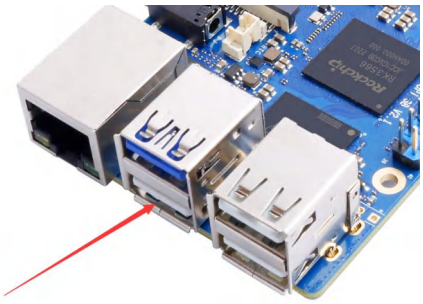

8) Then decompress **RKDevTool_Release_v3.15.zip**, this software does not need to be installed, just find **RKDevTool** in the decompressed folder and open it


9) After opening the **RKDevTool** burning tool, because the computer has not connected to the development board through the USB2.0 male-to-male data cable at this time, the lower left corner will prompt "**No device found**"



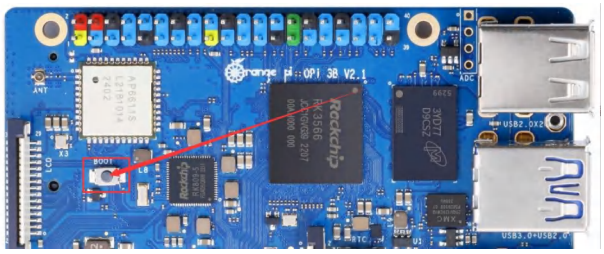

10) Then start burning the Android image to the TF card

a. First, connect the development board to the Windows computer through the USB2.0 male-to-male data cable. The position of the USB2.0 programming interface of the development board is shown in the figure below

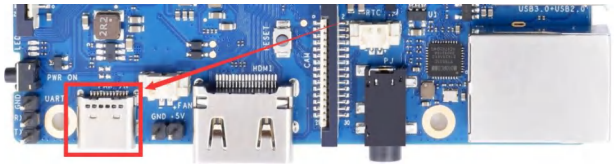


b. Then make sure that the development board is not inserted into the TF card and not connected to the power supply

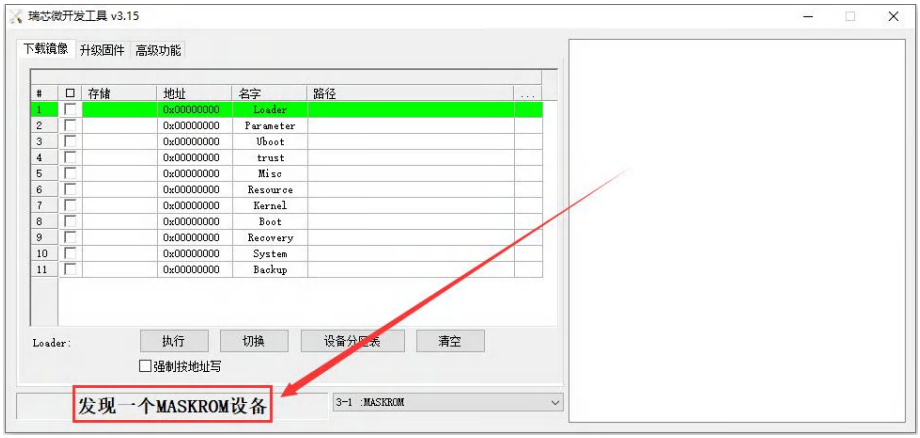
c. Then press and hold the MaskROM button on the development board, the position of the MaskROM button on the development board is shown in the figure below:



d. Then connect the power supply of the Type-C interface to the development board, and power on

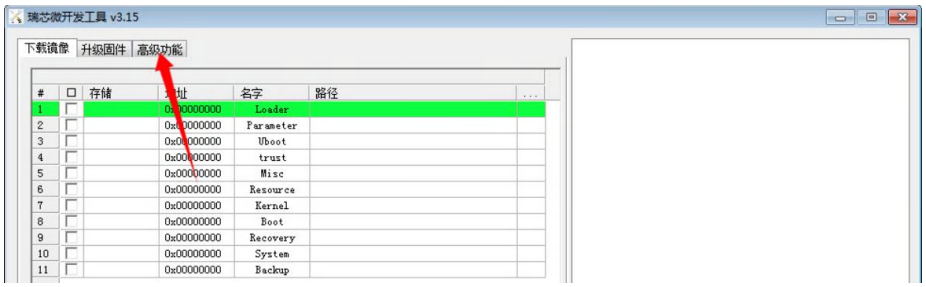


e. If the previous steps are successful, the development board will enter the **MASKROM** mode at this time, and the interface of the burning tool will prompt "**found a MASKROM device**"

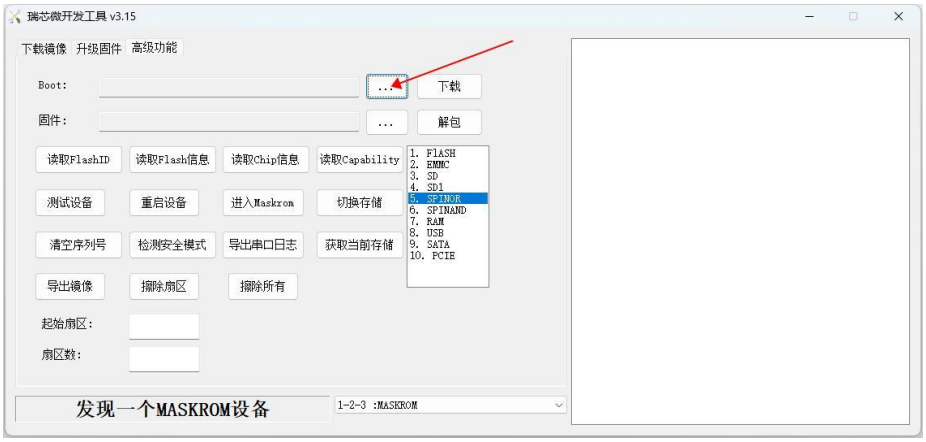


f. Then insert the TF card into the development board

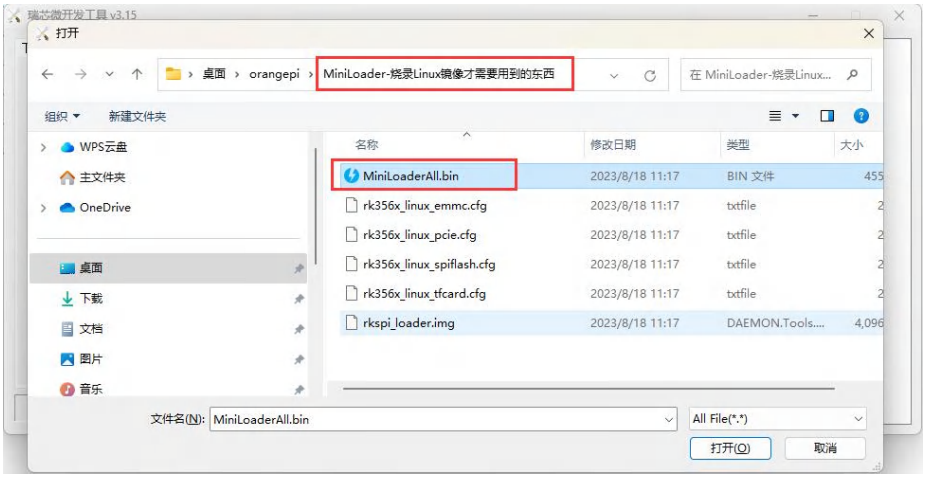
g. Then please select **advanced features**



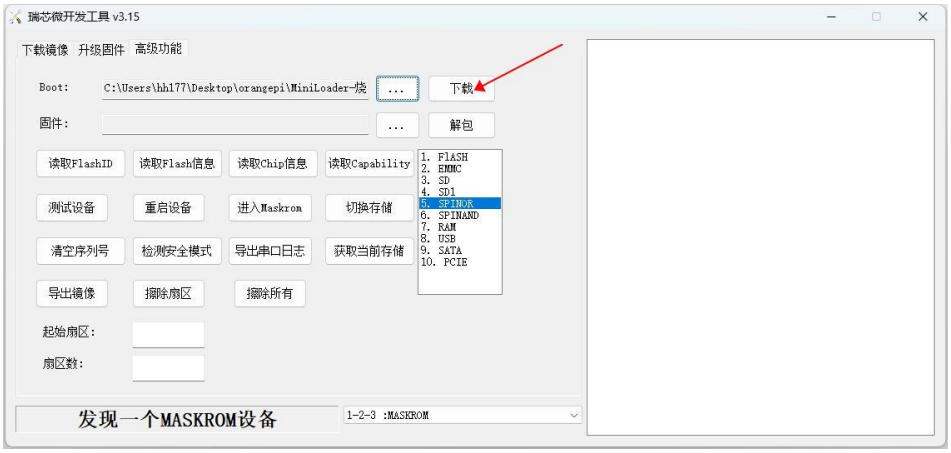

h. Then click the position shown in the figure below



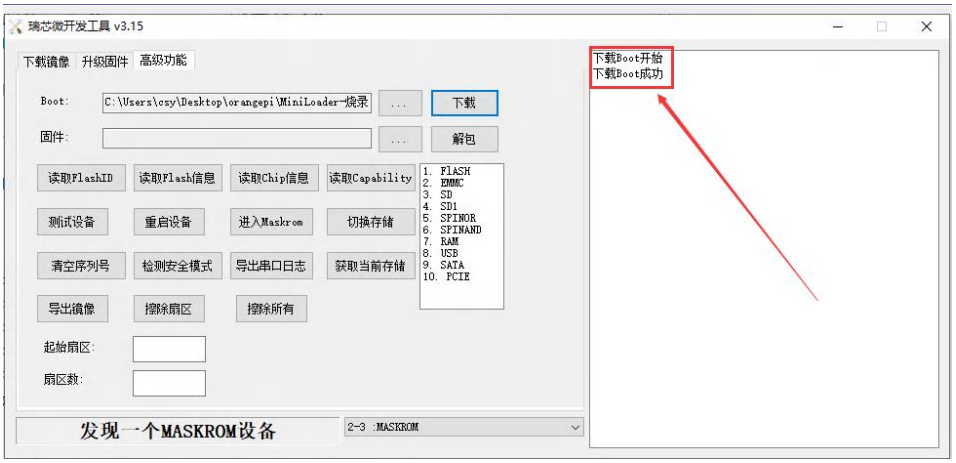

i. Then select **MiniLoaderAll.bin** in the **MiniLoader** folder downloaded earlier, and click to open




j. Then click **download**

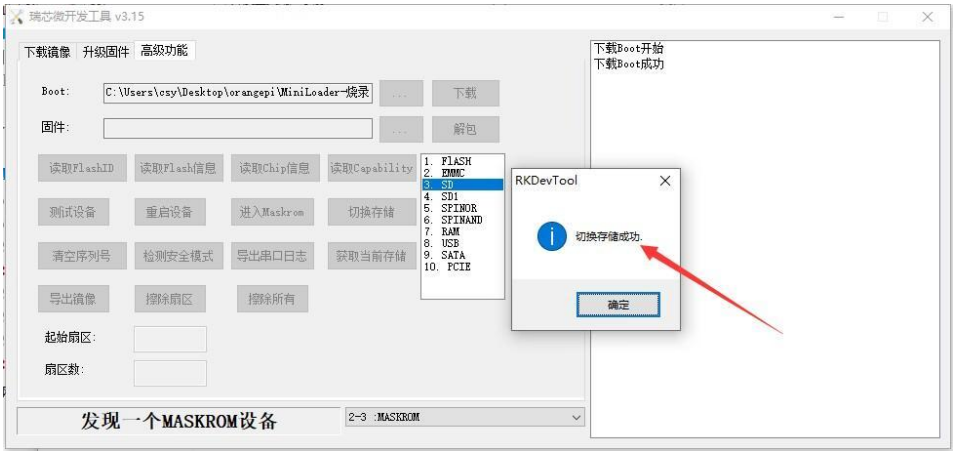



k. The display after downloading **MiniLoaderAll.bin** is shown in the figure below




l. Then select the storage device as **SD**, and then click **Switch Storage**

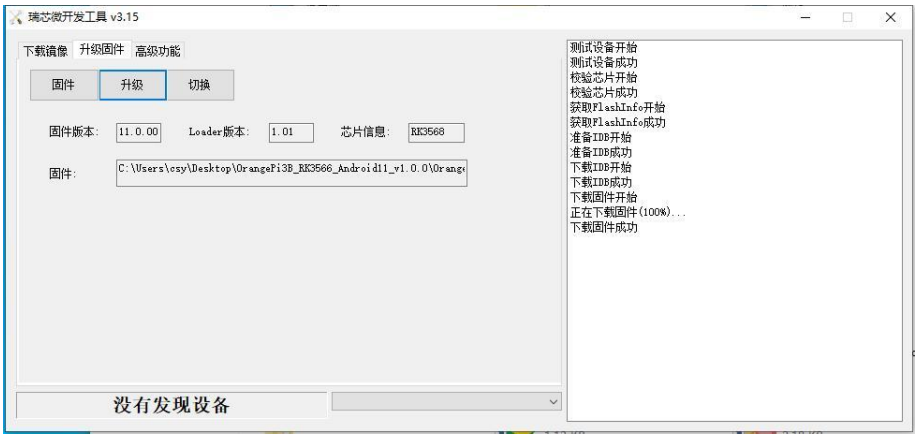

m. The display of successful switching is shown in the figure below




n. Then click the "**Upgrade Firmware**" column of the burning tool


o. Then click the "**Firmware**" button to select the path of the Android image that needs to be burned


p. Finally, click the "**Upgrade**" button to start burning, and the log during the burning process is shown in the figure below. After burning is completed, the Android system will start automatically.




## 2.7.2. How to use SDDiskTool to burn Android image to TF card

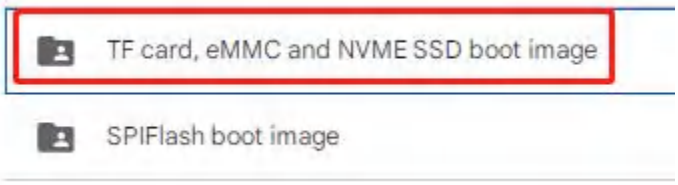
1) First prepare a TF card with 8GB or larger capacity. The transmission speed of the TF card must be class10 or above. It is recommended to use a TF card of SanDisk and other brands

2) Then use the card reader to insert the TF card into the computer

3) Then download the SDDiskTool programming tool from the **Orange Pi data download page**, **please make sure that the version of the SDDiskTool tool is the latest v1.72.**

4) Then download **the Android11 image from the Orange Pi download page**

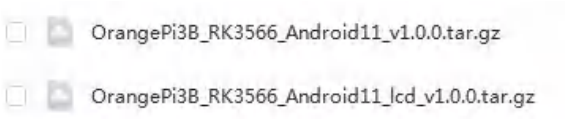
a. After opening the download link of the Android image, you can see the following two types of Android images, please select the image in the **TF card and eMMC startup image** folder to download




b. After entering the **TF card and eMMC boot image** folder, you can see the following two images, the difference between them is:

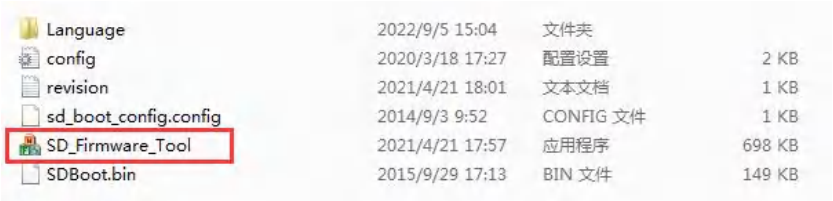
a) The image without lcd is specially used for HDMI display and supports 4K display. If you do not use the LCD screen, please download the image without lcd

b) If you want to use LCD screen, please choose image with lcd




5) Then use decompression software to decompress the compressed package of the downloaded Android image. Among the decompressed files, the file ending with ".img" is the Android image file, and the size is more than 1GB

6) Then use decompression software to decompress **SDDiskTool_v1.72.zip**, this software does not need to be installed, just find **SD_Firmware_Tool.exe** in the decompressed folder and open it




7) After opening **SDDiskTool**, if the TF card is recognized normally, the inserted disk device will be displayed in the "**Select Removable Disk Device**" column. P**lease make sure that the displayed disk device is consistent with the drive letter of the TF card you want to burn**, if there is no display, you can try to unplug the TF card


8) After confirming the drive letter, you can format the TF card first, click the **restore disk button** in SDDiskTool, or use the **SD Card Formatter** mentioned above to format the TF card


9) Then start to write the Android image to the TF card

a. First check "SD Boot" in "Select Function Mode"

b. Then select the path of the Android image in the "Select to upgrade firmware" column

c. Finally click the "Start Create" button to start burning the Android image to the TF card


10) After burning, you can exit the SDDiskTool software, and then you can pull out the TF card from the computer and insert it into the development board to start


## 2. 8. How to burn Android image to eMMC

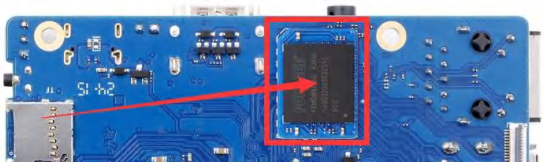
**Note, after burning the image into eMMC, if the test finds that it cannot be started, please clear the SPIFlash and try again. For the method of clearing SPIFlash, please refer to the method of using RKDevTool to clear SPIFlash.**

### 2. 8. 1. Method of burning Android image into eMMC through USB2.0 burning port

**Note that all the following operations are performed on a Windows computer.**

1) The development board reserves an eMMC expansion interface. Before programming the system to eMMC, you first need to purchase an eMMC module that matches the eMMC interface of the development board. Then install the eMMC module to the development board.

The eMMC module and the method of plugging into the development board are as follows:

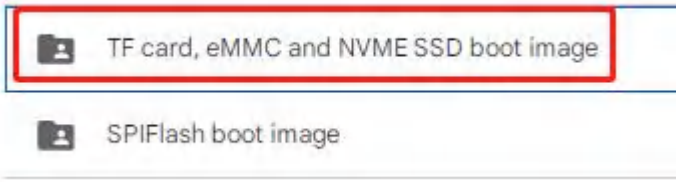


2) You also need to prepare a good quality USB2.0 male-to-male data cable

3) Then download Rockchip driver **DriverAssitant_v5.12.zip** and burning tool **RKDevTool_Release_v3.15.zip** from **Orange Pi's data download page**

4) Then download the Android image from **Orange Pi's download page.**

a. After opening the download link of the Android image, you can see the following two types of Android images, please select the image in the **TF card and eMMC startup image folder** to download




b. After entering the **TF card and eMMC startup image folder**, you can see the following two images, the difference between them is

a) The first image is dedicated to HDMI display and supports 4K display. If you don't use LCD screen, please download the image without lcd

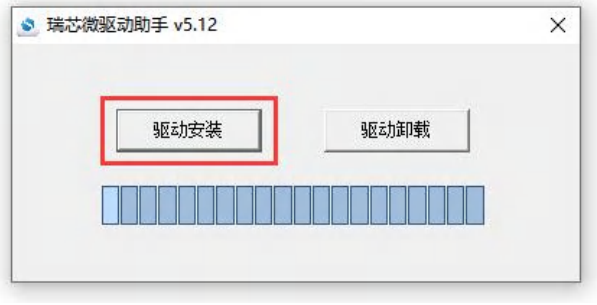
b) If you want to use lcd screen, please choose image with lcd


5) Then use decompression software to decompress the compressed package of the downloaded Android image. Among the decompressed files, the file ending with ".img" is the Android image file, and the size is more than 1GB

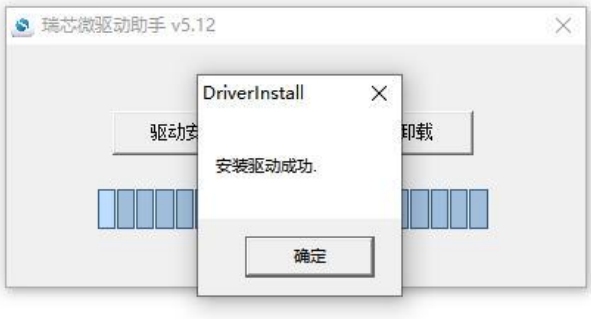
6) Then use decompression software to decompress **DriverAssitant_v5.12.zip**, and then find the **DriverInstall.exe** executable file in the decompressed folder and open it


7) After opening **DriverInstall.exe**, the steps to install the Rockchip driver are as follows

a. Click the "**Driver Installation**" button

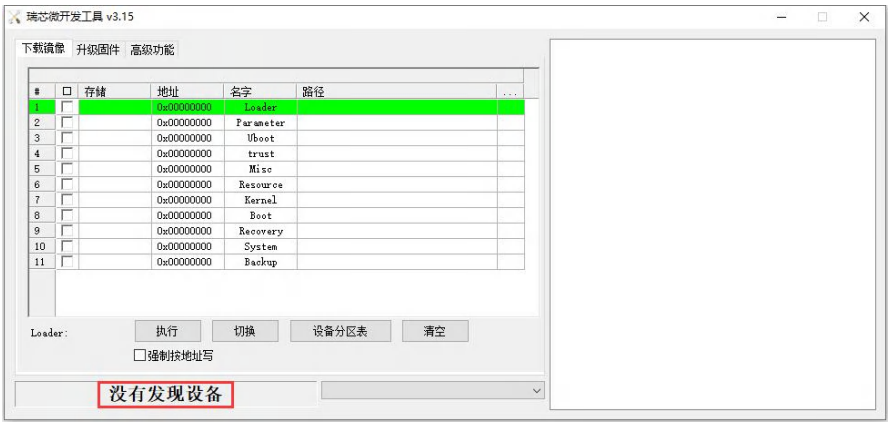



b. After waiting for a period of time, a pop-up window will prompt "**driver installed successfully**", and then click the "**OK**" button.



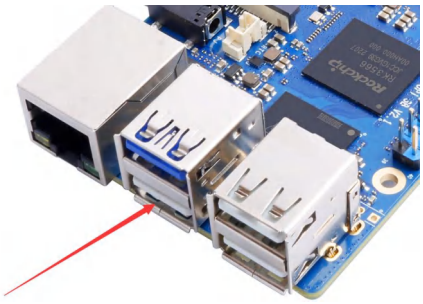

8) Then decompress **RKDevTool_Release_v3.15.zip**, this software does not need to be installed, just find **RKDevTool** in the decompressed folder and open it


9) After opening the **RKDevTool** burning tool, because the computer is not connected to the development board through the USB2.0 male-to-male data cable at this time, the lower left corner will prompt "**No device found**"



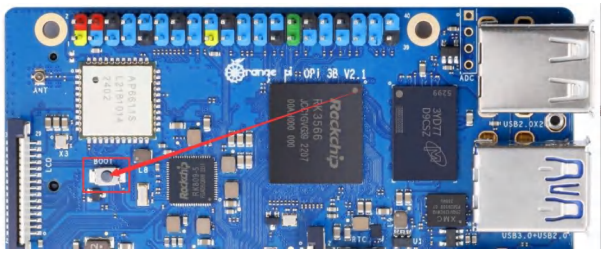

10) Then start burning the Android image into eMMC

a. First, connect the development board to the Windows computer through the USB2.0 male-to-male data cable. The position of the USB2.0 programming interface of the development board is shown in the figure below

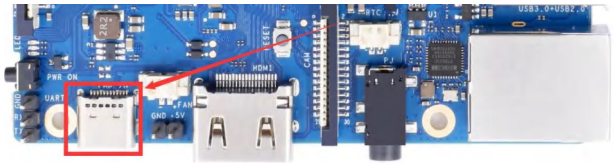


b. Then make sure that the development board is not inserted into the TF card and not connected to the power supply

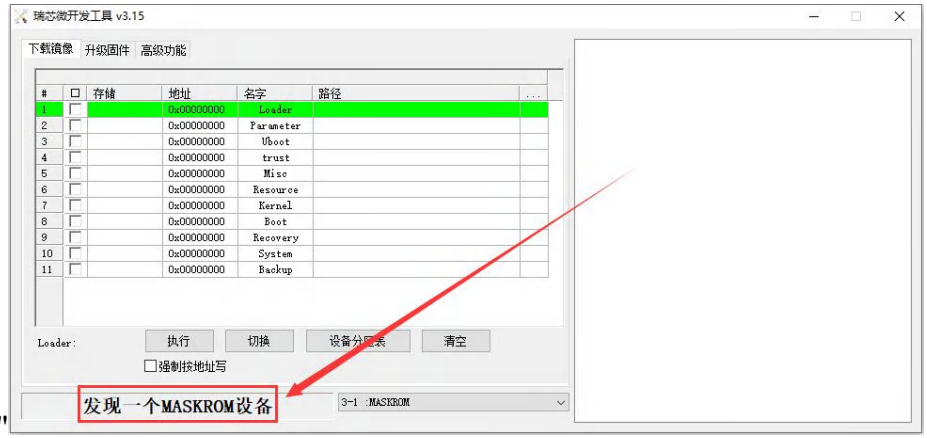
c. Then press and hold the MaskROM button on the development board, the position of the MaskROM button on the development board is shown in the figure below:



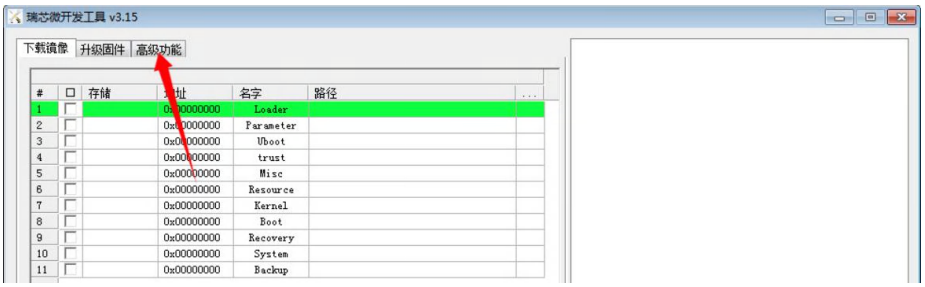
d. Then connect the power supply of the Type-C interface to the development board



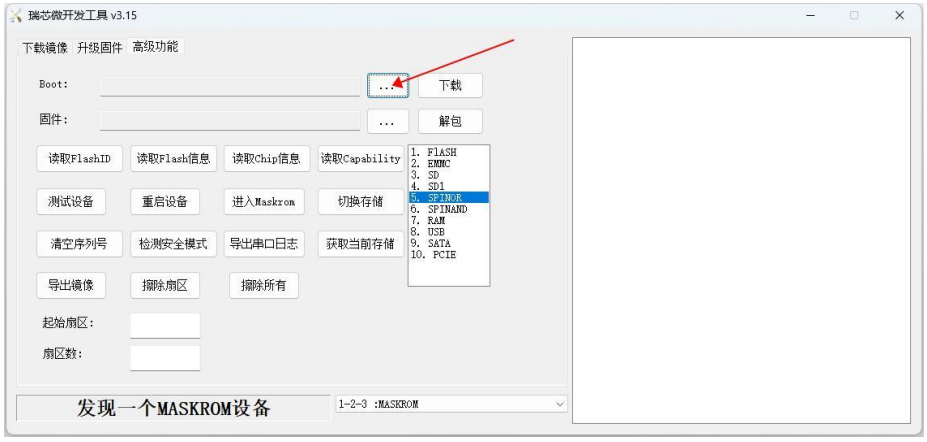
e. If the previous steps are successful, the development board will enter the **MASKROM** mode at this time, and the interface of the burning tool will prompt "**found a MASKROM device**"



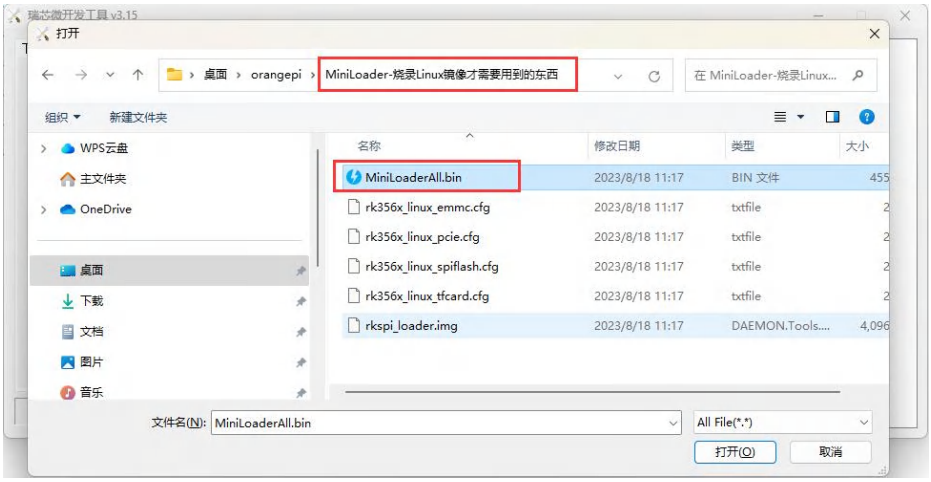

f. Then please select **Advanced Features**



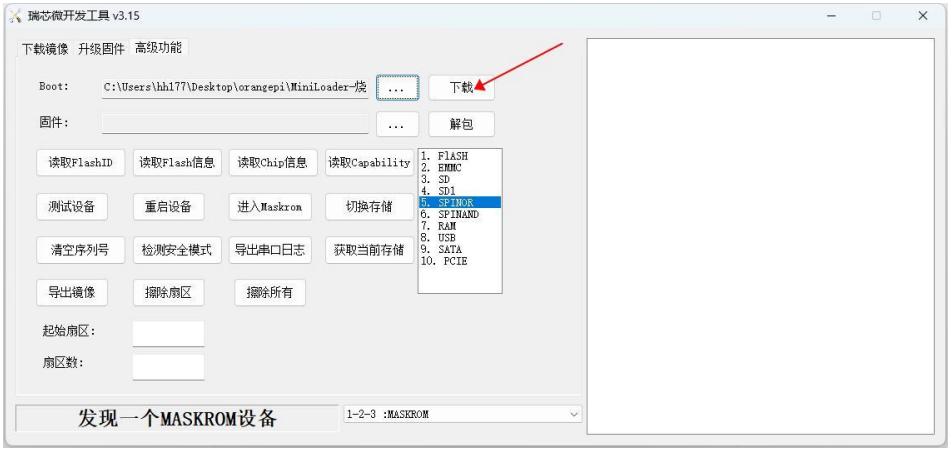

g. Then click the position shown in the figure below



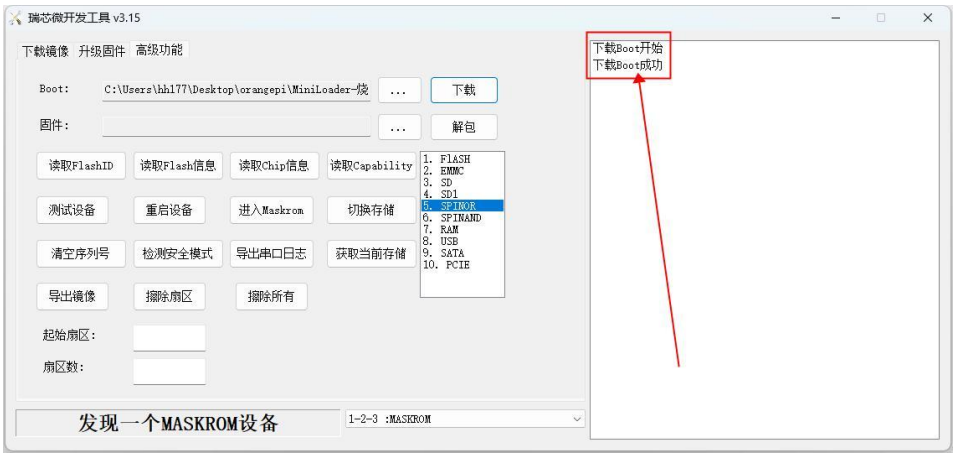

h. Select **MiniLoaderAll.bin** in the MiniLoader folder downloaded earlier, and click to open.



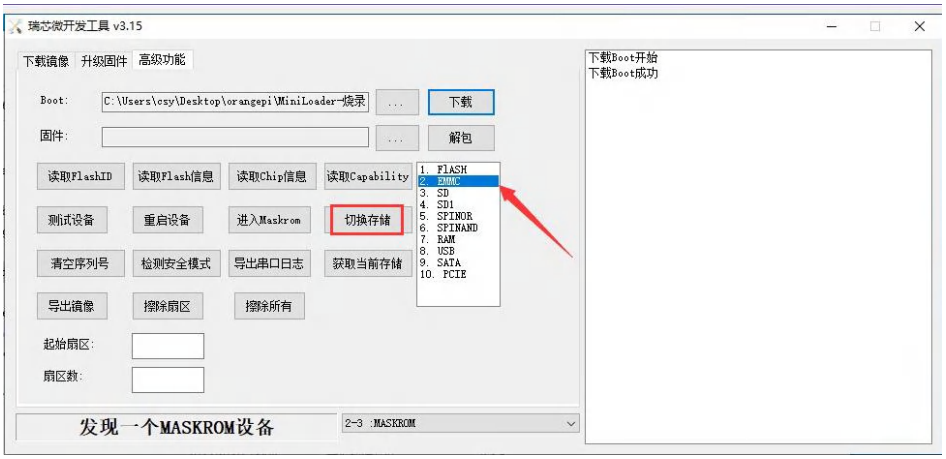

i. Then click **Download**



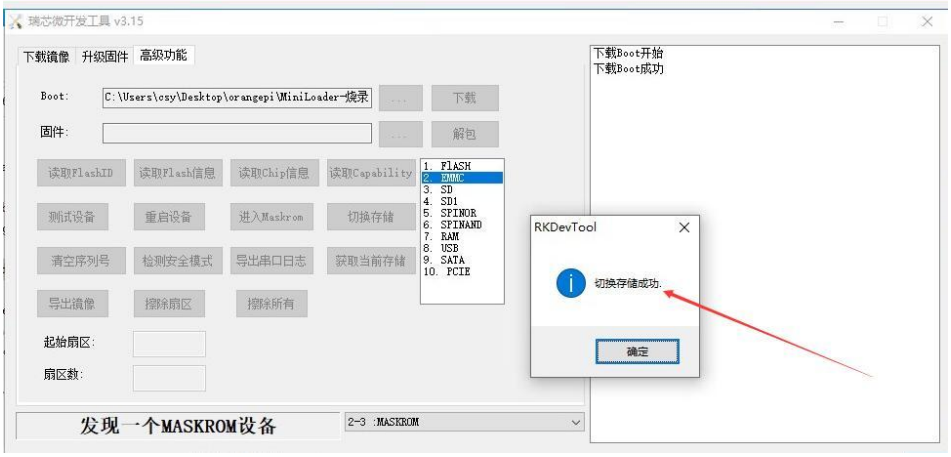

j. The display after downloading **MiniLoaderAll.bin** is shown in the figure below




k. Then select the storage device as **EMMC**, and then click Switch **Storage**



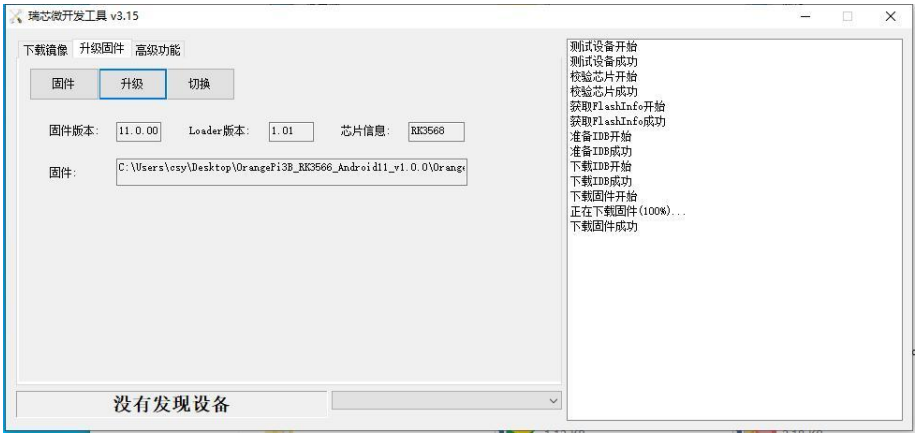

l. The display of successful switching is shown in the figure below




m. Then click the "**Upgrade Firmware**" column of the burning tool


n. Then click the "**Firmware**" button to select the path of the Android image that needs to be burned

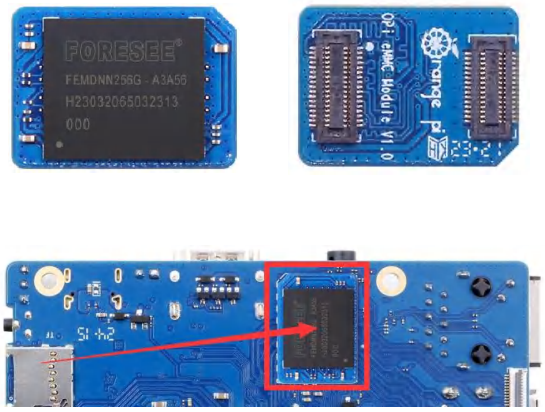

o. Finally, click the "**Upgrade**" button to start burning, and the log during the burning process is shown in the figure below. After burning is completed, the Android system will start automatically.




## 2.8.2. How to burn Android11 image into eMMC via TF card

**Note that all the following operations are performed on a Windows computer.**

1) The development board reserves an eMMC expansion interface. Before programming the system to eMMC, you first need to purchase an eMMC module that matches the eMMC interface of the development board. Then install the eMMC module to the development board. The eMMC module and the method of plugging into the development board are as follows:



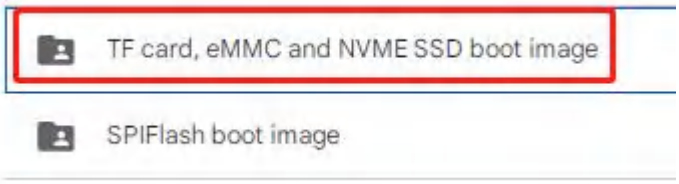

2) You also need to prepare a TF card with 8GB or larger capacity. The transmission speed of the TF card must be class10 or above. It is recommended to use a TF card of SanDisk and other brands

3) Then use the card reader to insert the TF card into the computer

4) Then download the SDDiskTool programming tool from the **Orange Pi data download page**, **please make sure that the version of the SDDiskTool tool is the latest v1.72**

5) Then download the Android image from **Orange Pi's download page**

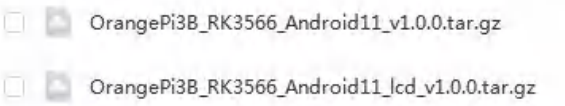
a. After opening the download link of the Android image, you can see the following two types of Android images, please select the image in **the TF card and eMMC startup image folder** to download




b. After entering **the TF card and eMMC boot image folder**, you can see the following tow images, the difference between them is:

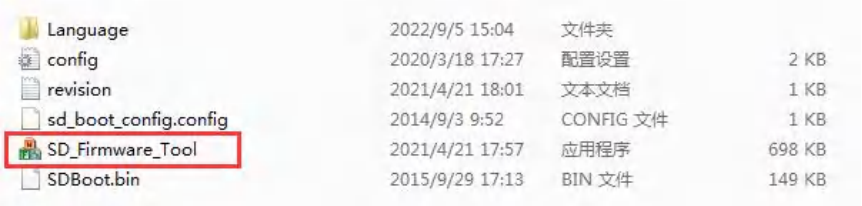
a) The first image is dedicated to HDMI display and supports 4K display. If you don't use LCD screen, please download the image without lcd

b) If you want to use lcd screen, please choose image with lcd

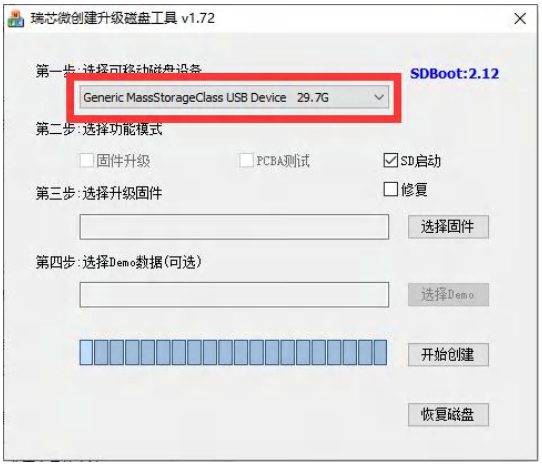



6) Then use the decompression software to decompress the compressed package of the downloaded Android image. Among the decompressed files, the file ending with ".img" is the Android image file, and the size is more than 1GB

7) Then use decompression software to decompress **SDDiskTool_v1.72.zip**, this software does not need to be installed, just find **SD_Firmware_Tool.exe** in the decompressed folder and open it




8) After opening **SDDiskTool**, if the TF card is recognized normally, the inserted disk device will be displayed in the "**Select Removable Disk Device**" column. **Please make sure that the displayed disk device is consistent with the drive letter of the TF card you want to burn**, if there is no display, you can try to unplug the TF card.




9) After confirming the drive letter, you can format the TF card first, click the **restore**

**disk button** in SDDiskTool, or use the **SD Card Formatter** mentioned above to format the TF card

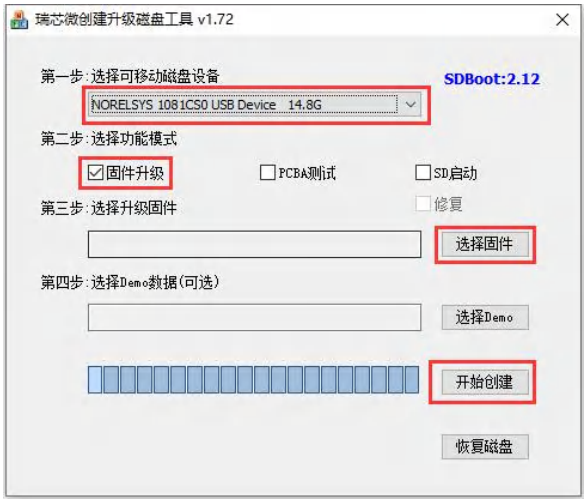

10) Then start to write the Android image into the TF card

a. First confirm that the displayed drive letter is the drive letter corresponding to the TF card under "**Select Removable Disk Device**"

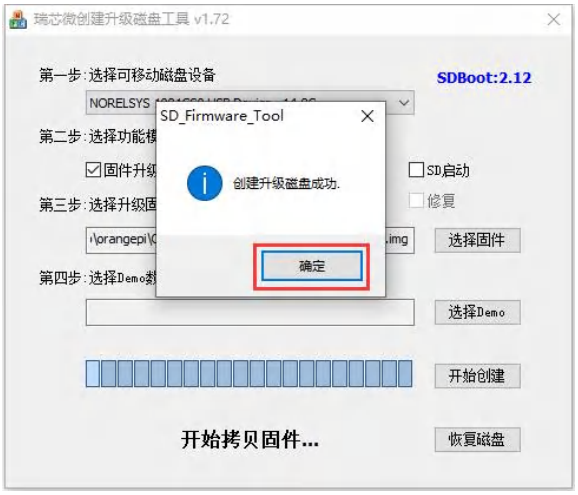
b. Then select "**Firmware Upgrade**" in "**Select Function Mode**"

c. Then select the path of the Android firmware in the "**Select Upgrade Firmware**" column

d. Finally click the "**Start Create**" button to start burning

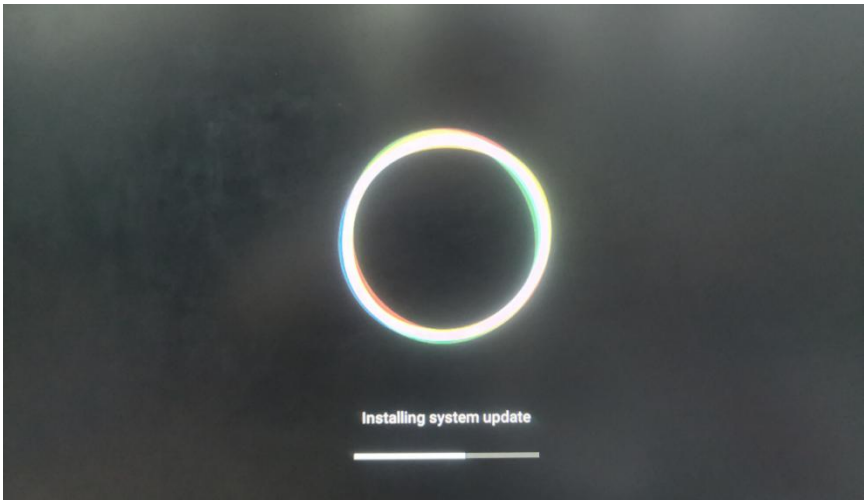



11) After the burning is completed, the display is as shown in the figure below, and then you can exit SDDiskTool

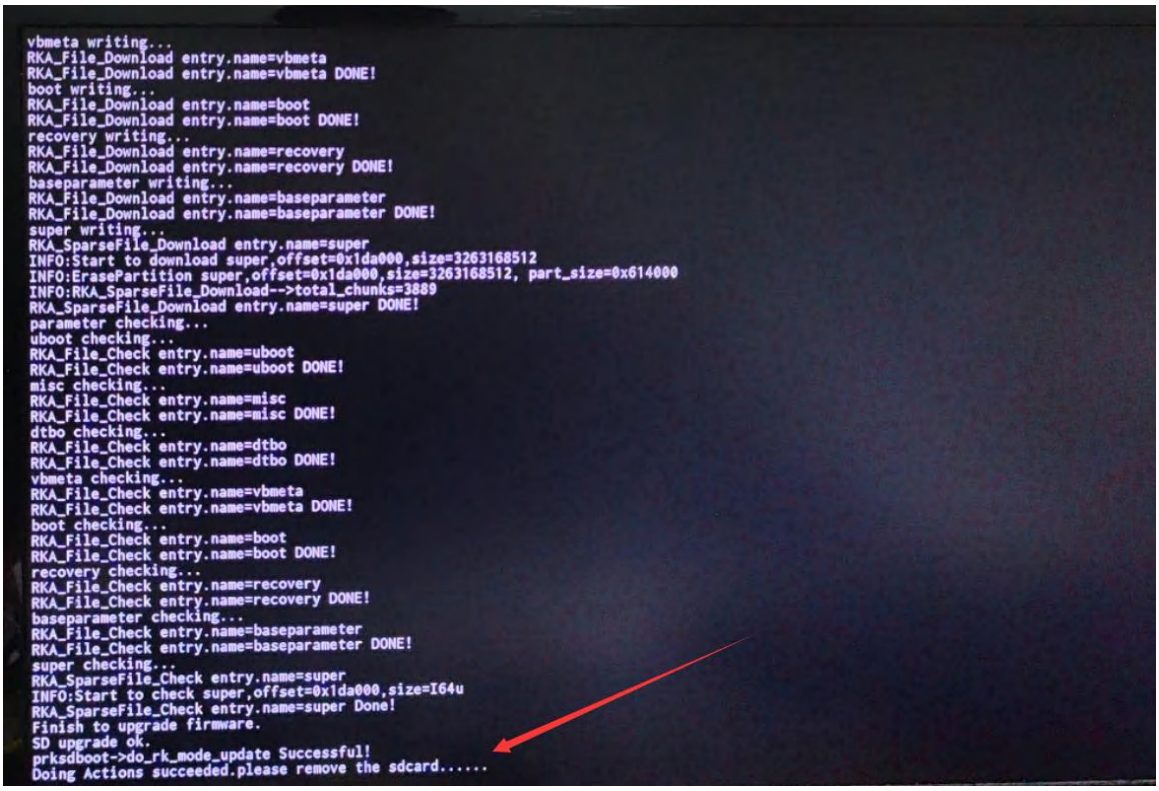



12) Then pull out the TF card from the computer and insert it into the development board. After the development board is powered on, it will automatically start burning the Android image in the TF card to the eMMC of the development board.

13) If the development board is connected to an HDMI display, you can also see the progress bar of burning the Android image to eMMC from the HDMI display




14) When the HDMI monitor displays the following information, it means that the burning of the Android image into the eMMC has been completed. At this time, the TF card can be pulled out, and then the Android system in the eMMC will start.

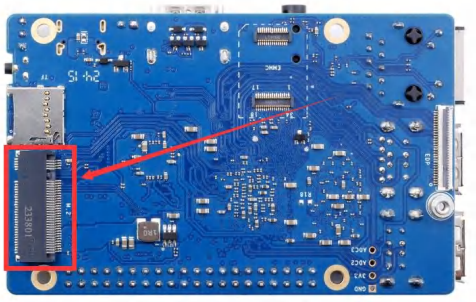



## 2. 9. How to burn Android image to SPIFlash+NVMe SSD

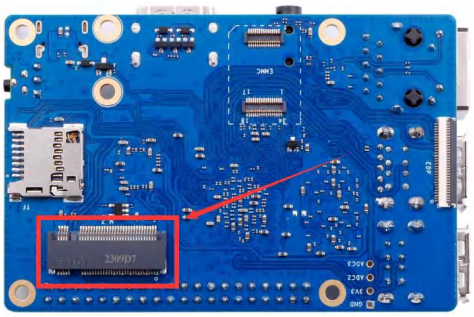
**Note that all the following operations are performed on a Windows computer.**

1) First, you need to prepare an NVMe SSD solid-state drive

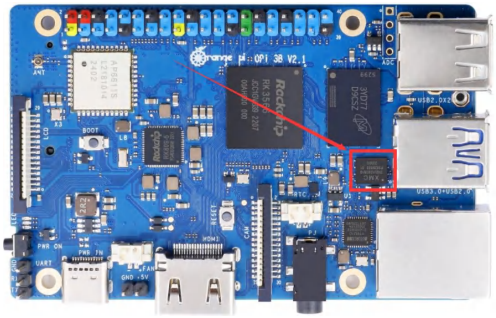
2) Then insert the NVMe SSD into the M.2 PCIe interface of the development board and secure it. The Orange Pi 3B has two hardware versions, and the M.2 PCIe interface position for version 2.1 is shown in the following figure:



The position of the M.2 PCIe interface in v1.1.1 version is shown in the following figure:



3) The position of SPI Flash on the development board is shown in the following figure, and no other settings are required before starting the burning process

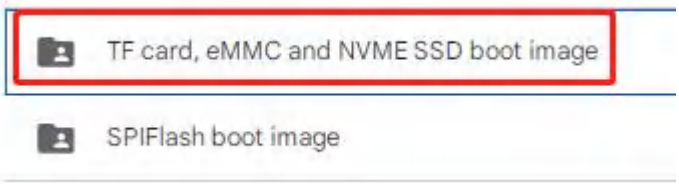


4) We also need to prepare a high-quality USB 2.0 male to male data cable

5) Then download Rockchip driver **DriverAssitant_v5.12.zip** and burning tool **RKDevTool_Release_v3.15.zip** from **Orange Pi's data download page**

6) Then download the image of Android11

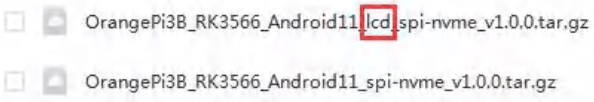
a. After opening the download link of the Android image, you can see the following two types of Android images, please select the image in the S**PIFlash-NVME SSD boot image folder** to download




b. After entering the **SPIFlash-NVME SSD boot image folder**, you can see the following two images. Their differences are:

a) The image without lcd is specially used for HDMI display and supports 4K display. If you do not use the LCD screen, please download the image without lcd

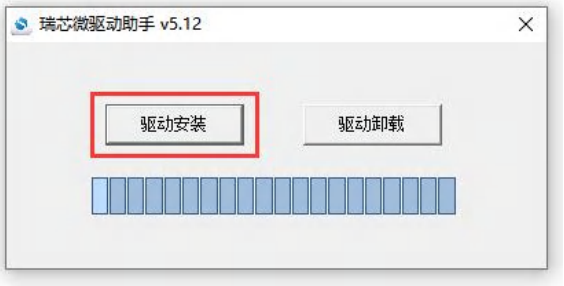
b) If you want to use LCD screen, please choose image with lcd



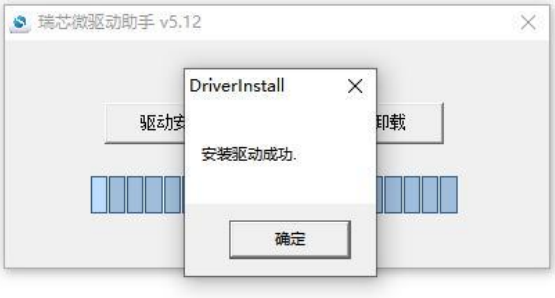

7) Then use the decompression software to decompress **DriverAssitant_v5.12.zip**, and then find the **DriverInstall.exe** executable file in the decompressed folder and open it


8) After opening **DriverInstall.exe**, the steps to install the Rockchip driver are as follows

a. Click the "**Driver Installation**" button

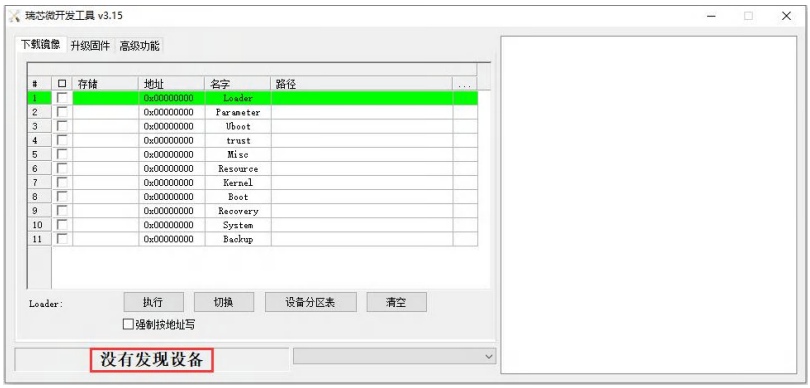



b. After waiting for a period of time, a pop-up window will prompt "**driver installed successfully**", and then click the "**OK**" button.



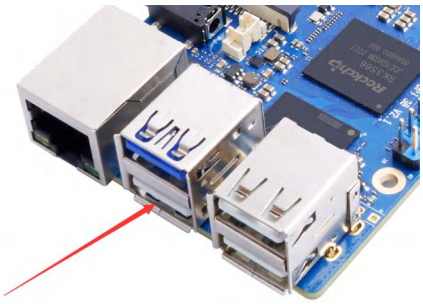

9) Then decompress **RKDevTool_Release_v3.15.zip**, this software does not need to be installed, just find **RKDevTool** in the decompressed folder and open it


10) After opening the **RKDevTool** burning tool, because the computer is not connected to the development board through the USB2.0 male-to-male data cable at this time, the lower left corner will prompt "**No device found**"



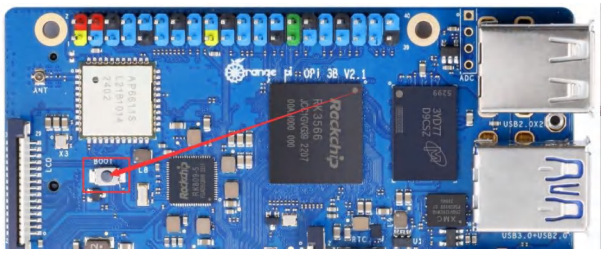

11) Then start burning the Android image to SPIFlash+NVMe SSD

a. First, connect the development board to the Windows computer through the USB2.0 male-to-male data cable. The position of the USB2.0 programming port of the development board is shown in the figure below

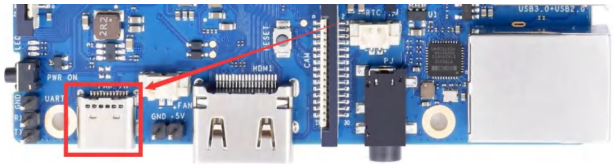


b. Make sure that the development board is not inserted into the TF card and not connected to the power supply

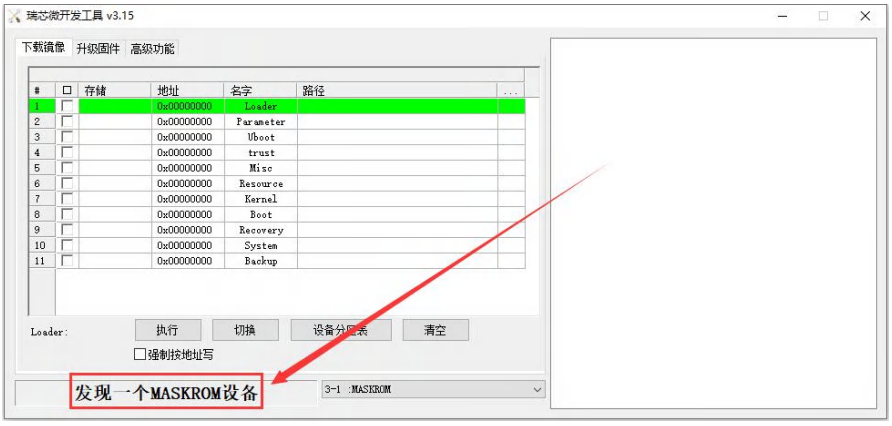
c. Then press and hold the MaskROM button on the development board, the position of the MaskROM button on the development board is shown in the figure below:



d. Then connect the power supply of the Type-C interface to the development board, and power on, and then release the MaskROM button



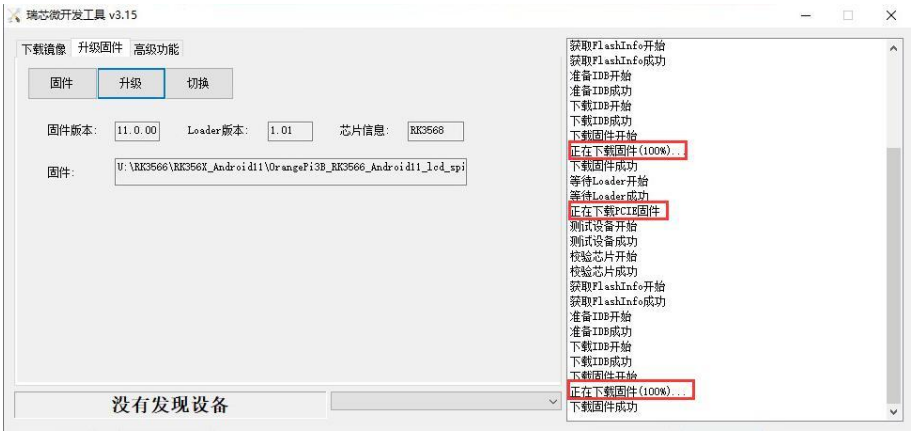
e. If the previous steps are successful, the development board will enter the **MASKROM** mode at this time, and the interface of the burning tool will prompt "**found a MASKROM device**"



f. Then click the "**Upgrade Firmware**" column of the burning tool


g. Then click the "**Firmware**" button to select the Android image to be burned


h. Finally, click the "Upgrade" button to start burning. The burning process is shown in the figure below. You can see that the firmware will be burned to SPIFlash first, and then burned to PCIE. After burning is completed, the Android system will start automatically.



## 2. 10. How to burn Orange Pi OS (OH) image to TF card

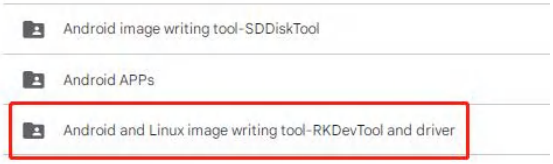
**Note that all operations below are performed on a Windows computer.**

1) First prepare a TF card with 8GB or larger capacity. The transmission speed of the TF card must be class10 or above. It is recommended to use TF cards from SanDisk and other brands.

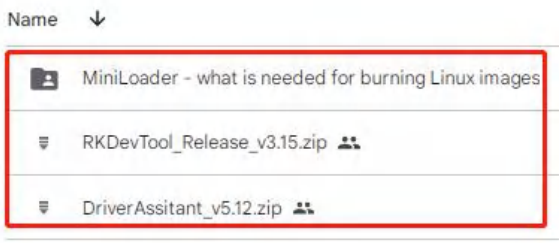
2) You also need to prepare a good quality USB2.0 male-to-male data cable

3) Then download Rockchip microdriver **DriverAssitant_v5.12.zip** and burning tool **RKDevTool_Release_v3.15.zip** from the **Orange Pi data download page**

a. On the **Orange Pi data download page**, first select the **official tool**, and then enter the folder below




b. Then download all the files below




4) Then download the image of OPi OS (OH) from the **Orange Pi data download page**

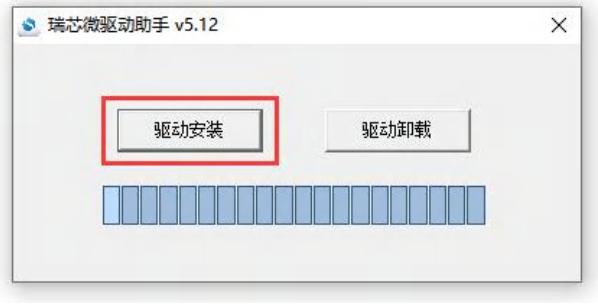
5) Then use decompression software to decompress the compressed package of the

downloaded OPi OS (OH) image

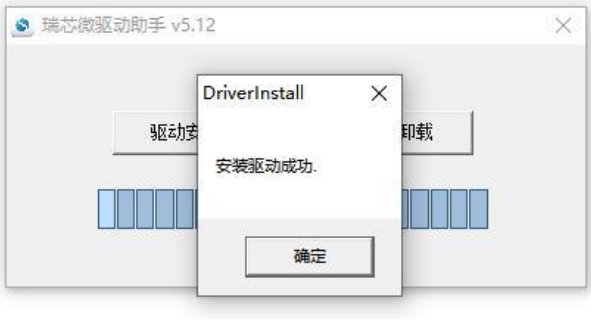
6) Then use decompression software to decompress **DriverAssitant_v5.12.zip**, then find the **DriverInstall.exe** executable file in the decompressed folder and open it.


7) Open **DriverInstall.exe** and install the Rockchip microdriver as follows:

a. Click the "**Driver Installation**" button

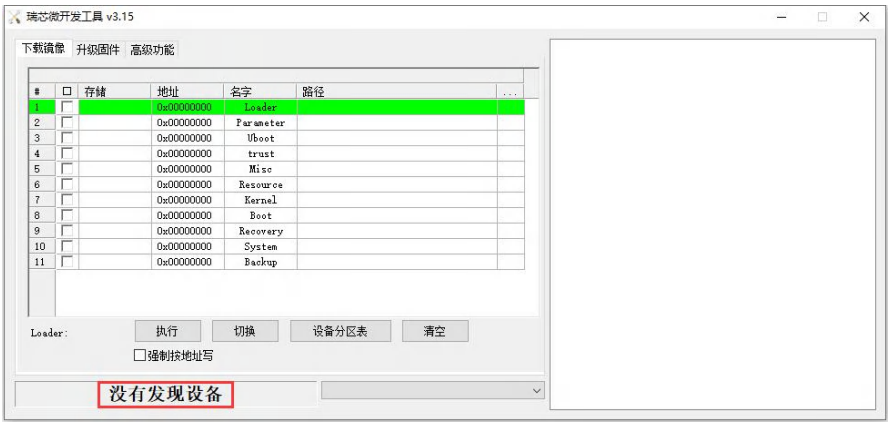



b. After waiting for a period of time, a window will pop up prompting "**Driver installation successful**", then click the "**OK**" button.



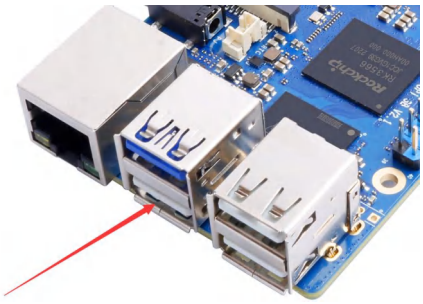

8) Then unzip **RKDevTool_Release_v3.15.zip**. This software does not need to be installed. Just find **RKDevTool** in the unzipped folder and open it.


9) After opening the **RKDevTool** burning tool, because the computer has not yet connected to the development board through the USB2.0 male-to-male data cable, a message "**No device found**" will appear in the lower left corner.



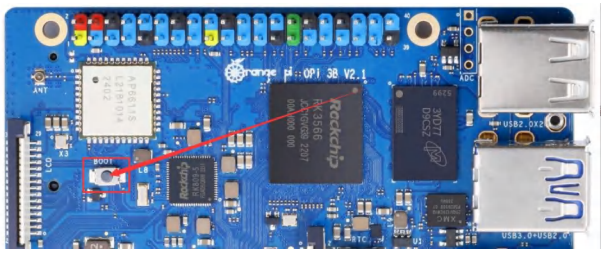

10) Then start burning the OPi OS (OH) image to the TF card

a. First, connect the development board to the Windows computer through a USB2.0 male-to-male data cable. The location of the development board's USB2.0 burning interface is as shown in the figure below.

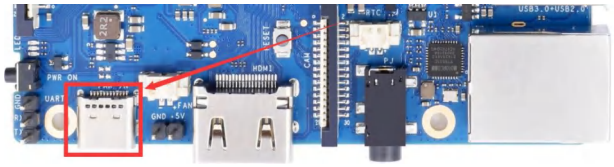


b. **Then make sure that the TF card is not inserted into the development board and the power supply is not connected.**

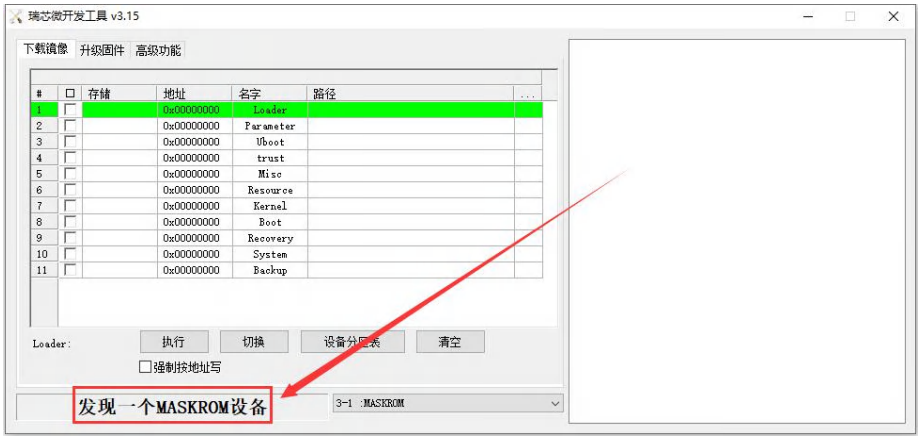
c. Then press and hold the MaskROM button on the development board. The position of the MaskROM button on the development board is as shown in the figure below:



d. Then connect the development board to the power supply of the Type-C interface and power on

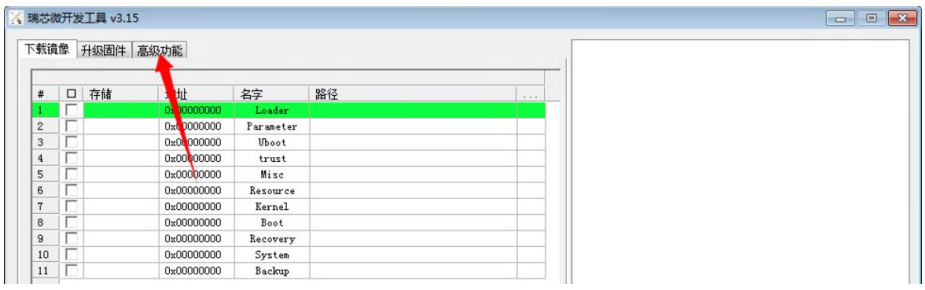


e. If the previous steps go well, the development board will enter **MASKROM** mode at this time, and the interface of the burning tool will prompt "**A MASKROM device was found**"

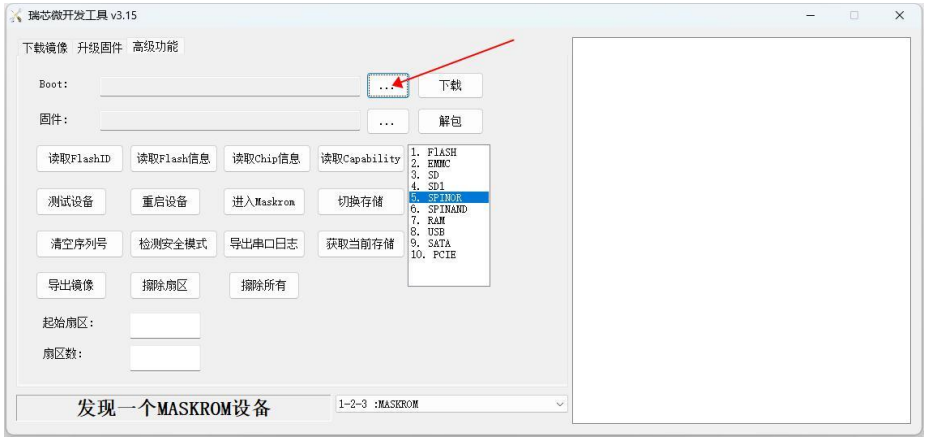



f. Then insert the TF card into the development board

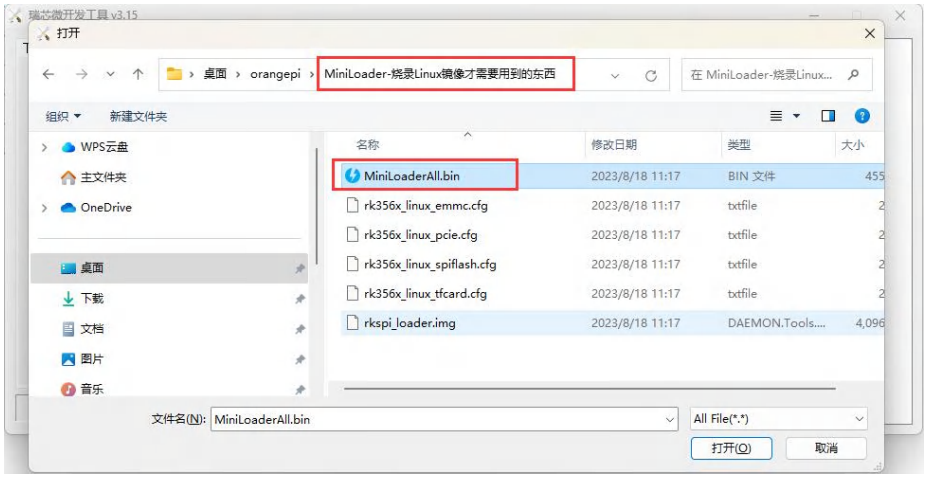
g. Then please select **advanced functions**



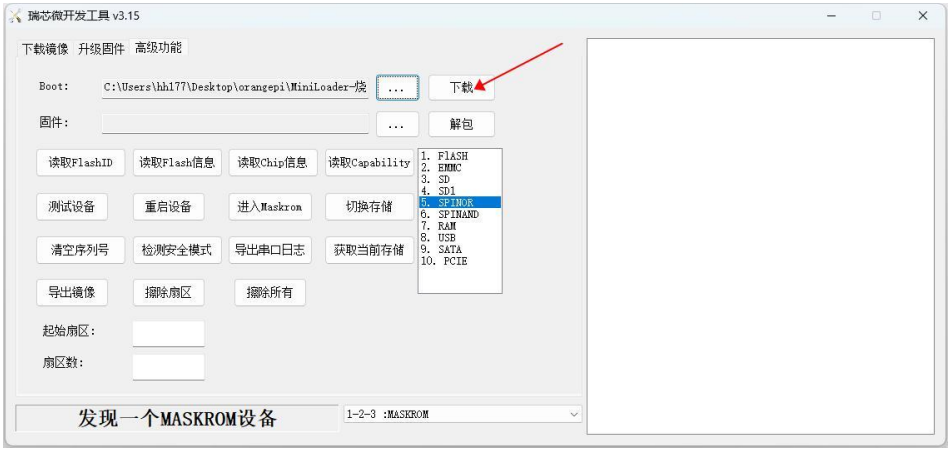

h. Then click the location shown in the picture below



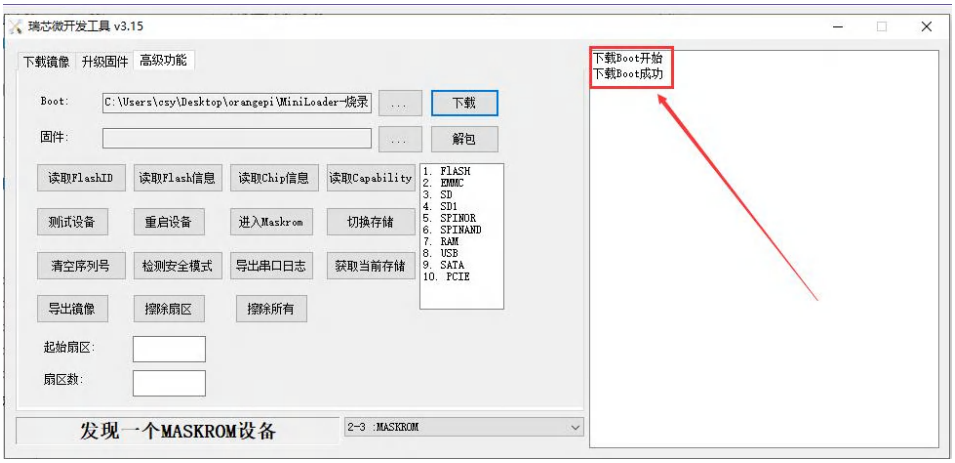

i. Then select **MiniLoaderAll.bin** in the **MiniLoader** folder downloaded earlier, and then click Open




j. Then click **download**

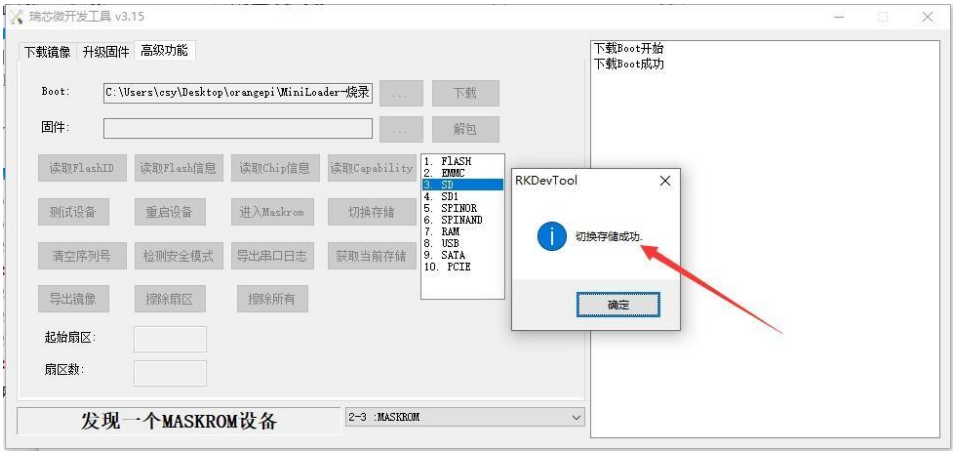



k. After downloading **MiniLoaderAll.bin**, the display is as shown below




l. Then select the storage device as **SD**, and then click to **switch storage**


m. The successful switching is displayed as shown below




n. Then click the "**Upgrade Firmware**" column of the burning tool


o. Then click the "**Firmware**" button to select the path of the OPi OS (OH) image that needs to be burned.


p. Finally, click the "**Upgrade**" button to start burning. The log during the burning process is as shown below. After the burning is completed, the OPi OS (OH) system will automatically start.

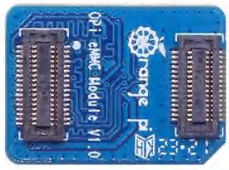
## 2. 11. Method to burn Orange Pi OS (OH) image into eMMC

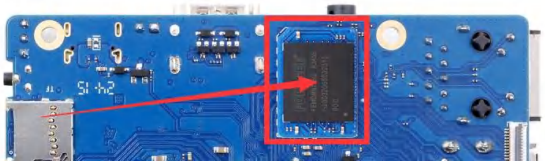
**Note that after burning the image into eMMC, if you test and find that it cannot be started, please clear SPIFlash and try to start again. For the method of clearing SPIFlash, please refer to the section "How to clear SPIFlash using RKDevTool".**

**Note that all operations below are performed on a Windows computer.**

1) The development board has reserved eMMC expansion interface. Before burning the system to eMMC, you first need to purchase an eMMC module that matches the eMMC interface of the development board. Then install the eMMC module to the development board.

The eMMC module and the method of inserting the development board are as follows:

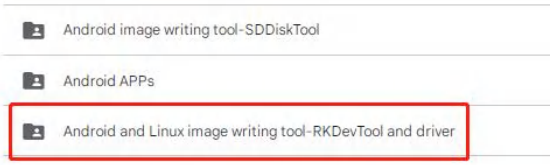



2) You also need to prepare a good quality USB2.0 male-to-male data cable

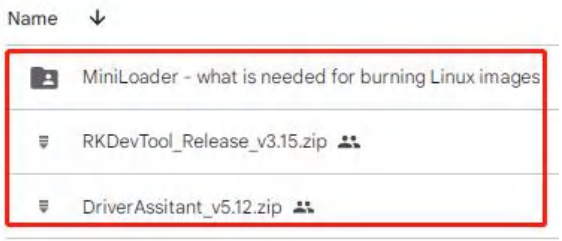
3) Then download Rockchip microdriver **DriverAssitant_v5.12.zip** and burning tool **RKDevTool_Release_v3.15.zip** from the **Orange Pi data download page**

a. On the **Orange Pi data download page**, first select the **official tool**, and then

enter the folder below




b. Then download all the files below



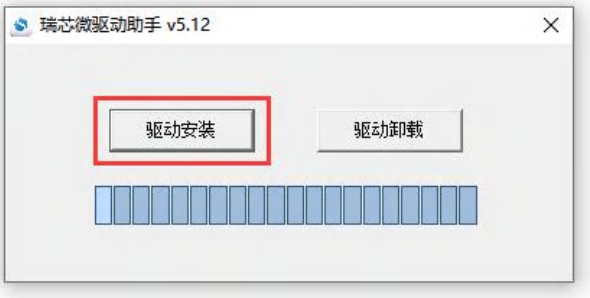

4) Then download the image of OPi OS (OH) from the **Orange Pi data download page**

5) Then use decompression software to decompress the compressed package of the downloaded OPi OS (OH) image

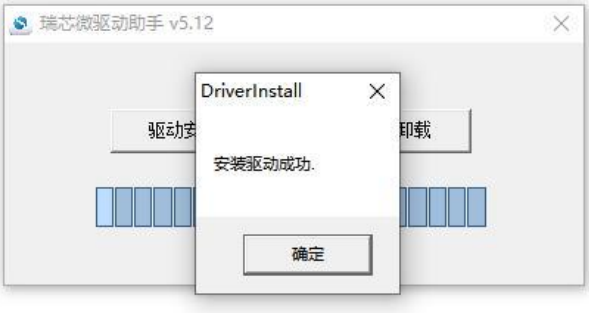
6) Then use decompression software to decompress **DriverAssitant_v5.12.zip**, then find the **DriverInstall.exe** executable file in the decompressed folder and open it.


7) Open **DriverInstall.exe** and install the Rockchip microdriver as follows: 开**DriverInstall.exe**

a. Click the "**Driver Installation**" button

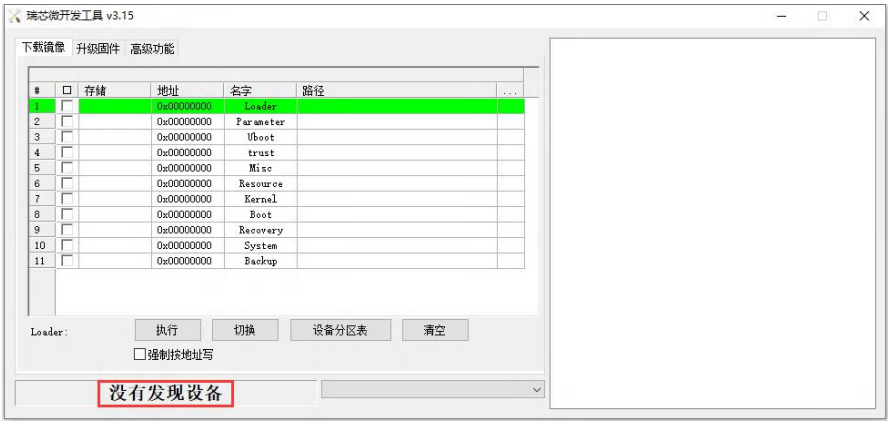



b. After waiting for a period of time, a window will pop up prompting "**Driver installation successful**", then click the "**OK**" button.



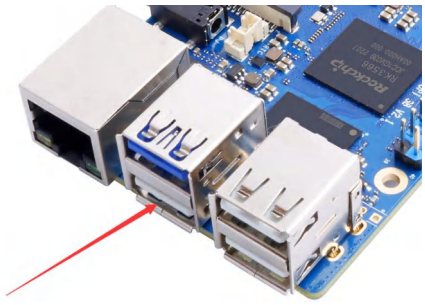

8) Then unzip **RKDevTool_Release_v3.15.zip**. This software does not need to be installed. Just find **RKDevTool** in the unzipped folder and open it.


9) After opening the **RKDevTool** burning tool, because the computer has not yet connected to the development board through the USB2.0 male-to-male data cable, a message "**No device found**" will appear in the lower left corner.



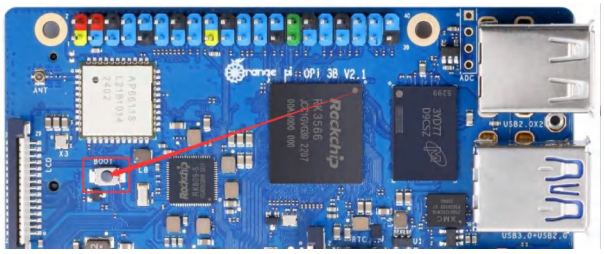

10) Then start burning the OPi OS (OH) image into eMMC

a. First, connect the development board to the Windows computer through a USB2.0 male-to-male data cable. The location of the development board's USB2.0 burning interface is as shown in the figure below.

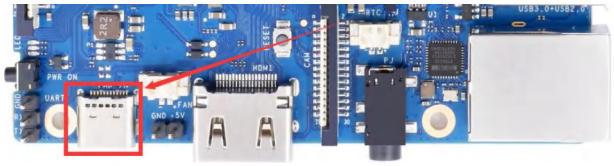


b. Then make sure that the TF card is not inserted into the development board and the power supply is not connected.

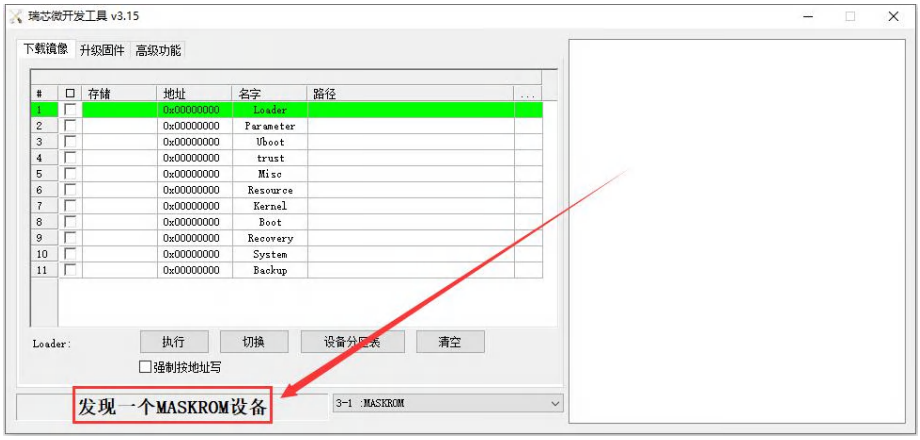
c. Then press and hold the MaskROM button on the development board. The position of the MaskROM button on the development board is as shown in the figure below:



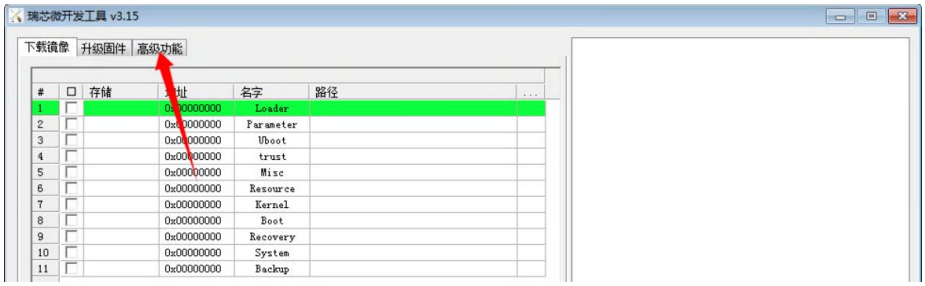
d. Then connect the development board to the power supply of the Type-C interface and power on



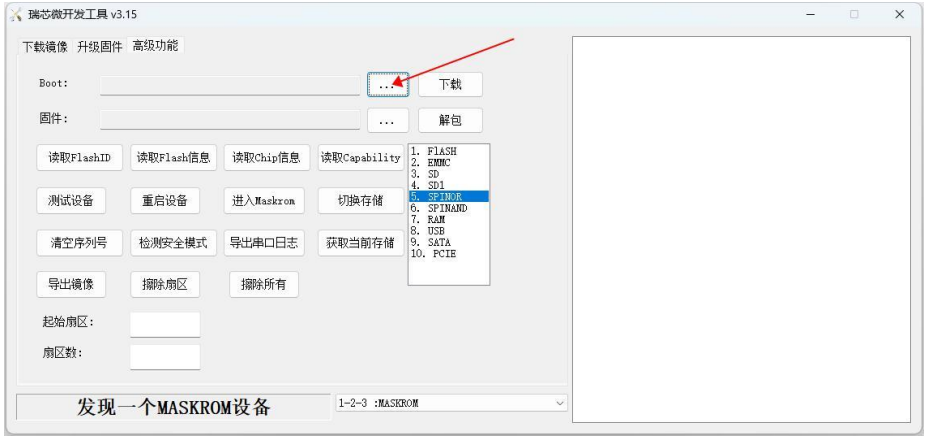
e. If the previous steps go well, the development board will enter **MASKROM** mode at this time, and the interface of the burning tool will prompt "**A MASKROM device was found**"



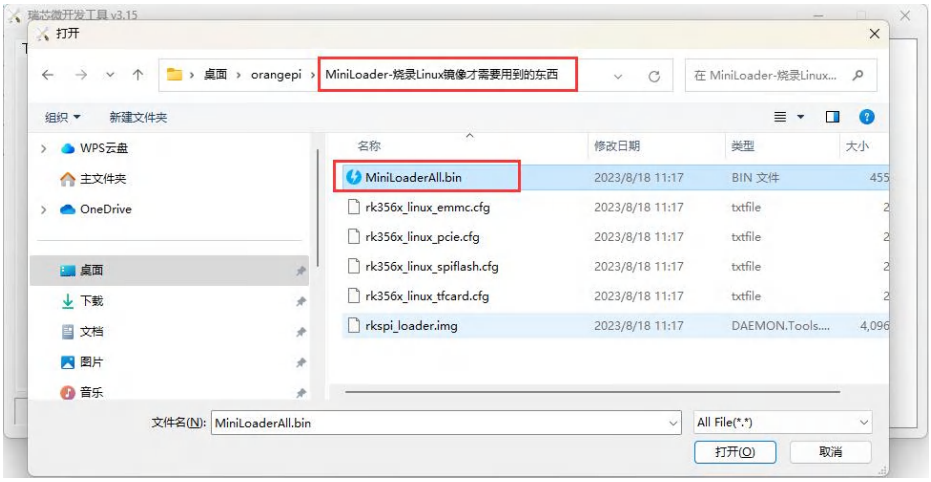

f. Then please select **advanced functions**



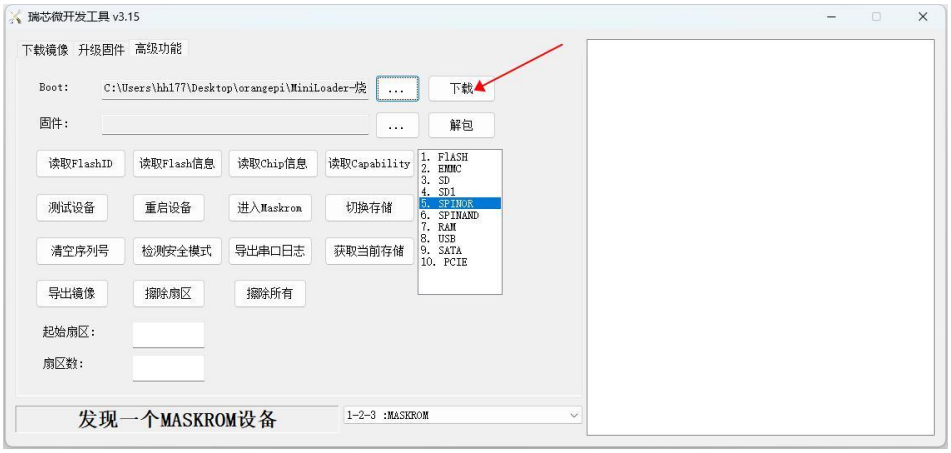

g. Then click the location shown in the picture below



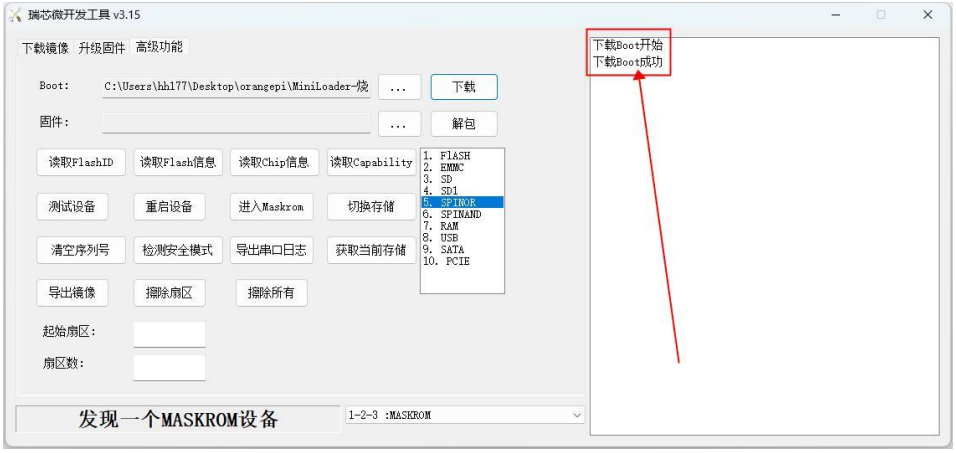

h. Then select **MiniLoaderAll.bin** in the **MiniLoader** folder downloaded earlier, and then click to open



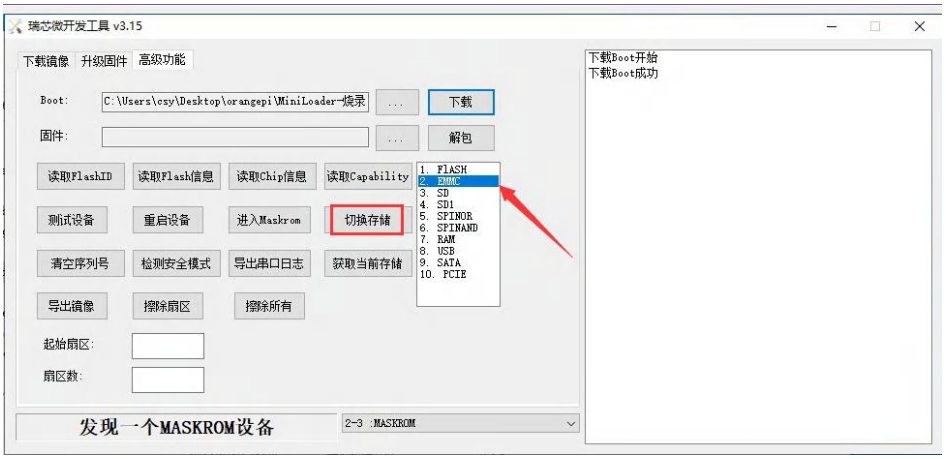

i. Then click **Download**



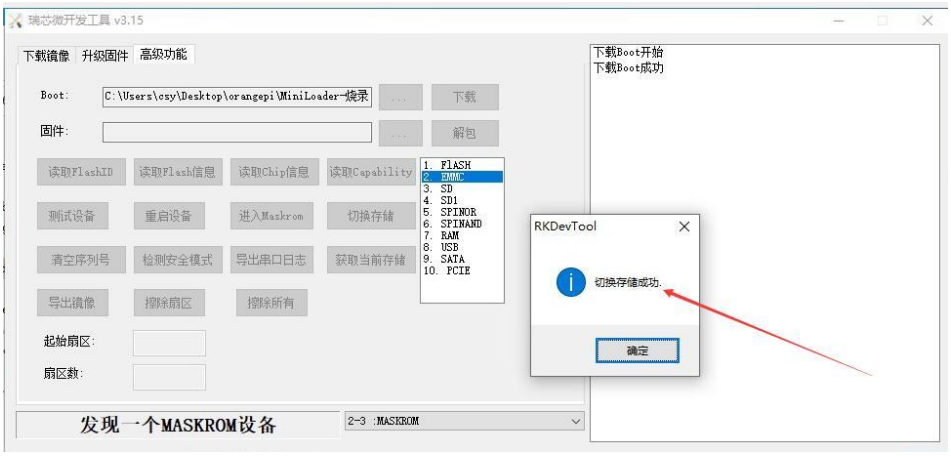

j. After downloading **MiniLoaderAll.bin**, the display is as shown below




k. Then select the storage device as **EMMC**, and then click to **switch storage**




l. The successful switching is displayed as shown below.



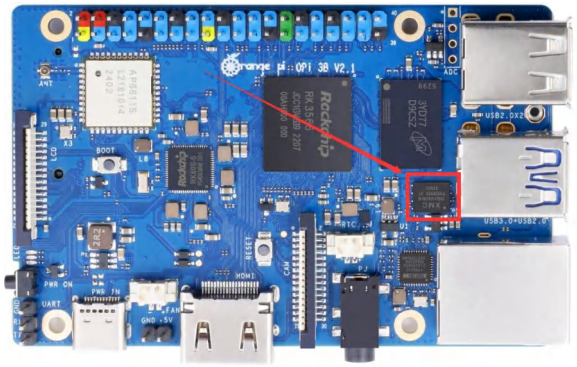

m. Then click the "**Upgrade Firmware**" column of the burning tool


n. Then click the "**Firmware**" button to select the path of the OPi OS (OH) image that needs to be burned.


o. Finally, click the "**Upgrade**" button to start burning. The log during the burning process is as shown below. After the burning is completed, the OPi OS (OH) system will automatically start.


## 2. 12. Using RKDevTool to clear SPIFlash

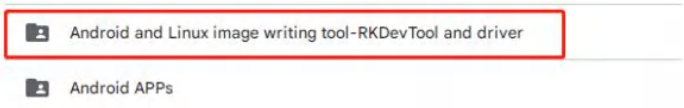
1) The position of SPI Flash on the development board is shown in the figure below



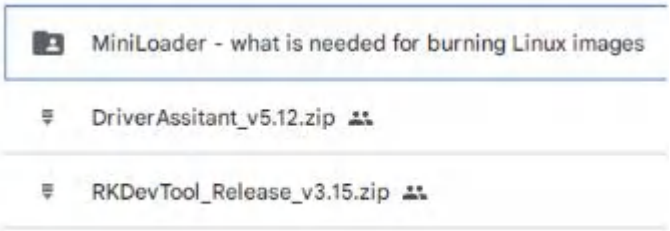
2) First, you need to prepare a good quality USB2.0 male-to-male data cable

3) Then download the Rockchip driver **DriverAssitant_v5.12.zip** and MiniLoader and the burning tool **RKDevTool_Release_v3.15.zip** from the Orange Pi data download page

a. On the download page of Orange Pi, first select the official tool, and then enter the following folder




b. Then download all the files below

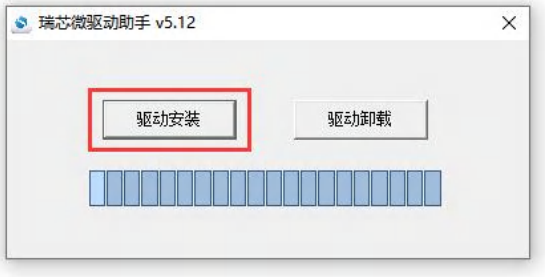



**Note that the "MiniLoader-things needed to burn the Linux image" folder is hereinafter referred to as the MiniLoader folder.**

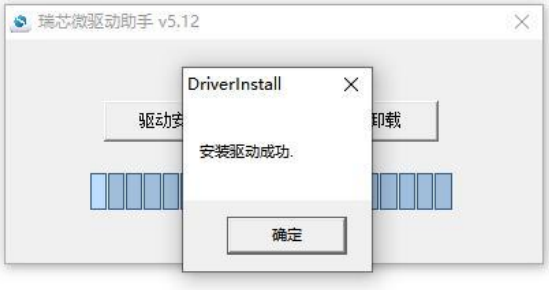
4) Then use decompression software to decompress **DriverAssitant_v5.12.zip**, and then find the **DriverInstall.exe** executable file in the decompressed folder and open it

5) After opening DriverInstall.exe, the steps to install the Rockchip driver are as follows

a. Click the "**Driver Installation**" button

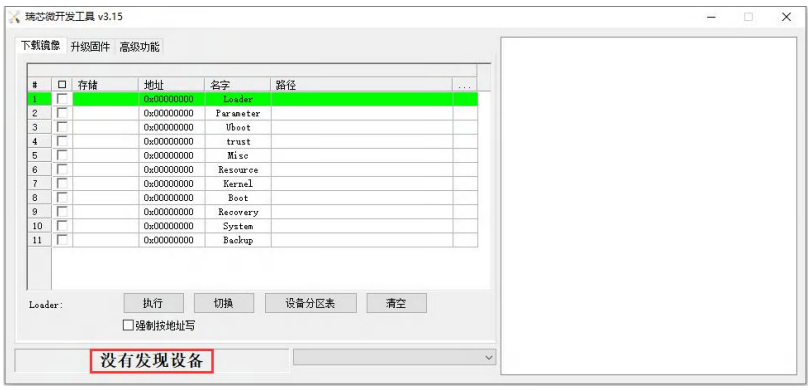



b. After waiting for a period of time, a pop-up window will prompt "**driver installed successfully**", and then click the "**OK**" button.



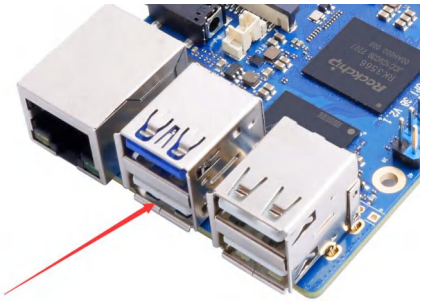

6) Then decompress **RKDevTool_Release_v3.15.zip**, this software does not need to be installed, just find **RKDevTool** in the decompressed folder and open it


7) After opening the **RKDevTool** burning tool, because the computer has not connected to the development board through the USB2.0 male-to-male data cable at this time, the lower left corner will prompt "**No device found**"



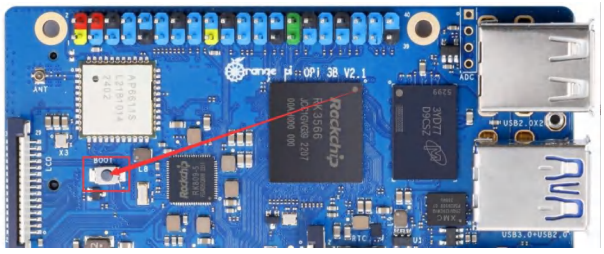

8) Then you can start to clear the content in SPI FLASH

a. First, connect the development board to the Windows computer through the USB2.0 male-to-male data cable. The position of the USB2.0 programming port of the development board is shown in the figure below

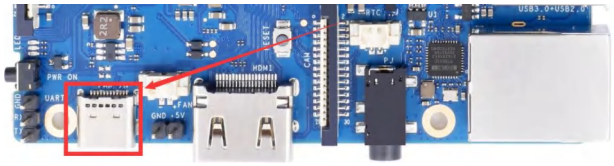


b. Make sure that the development board is not inserted into the TF card and not connected to the power supply

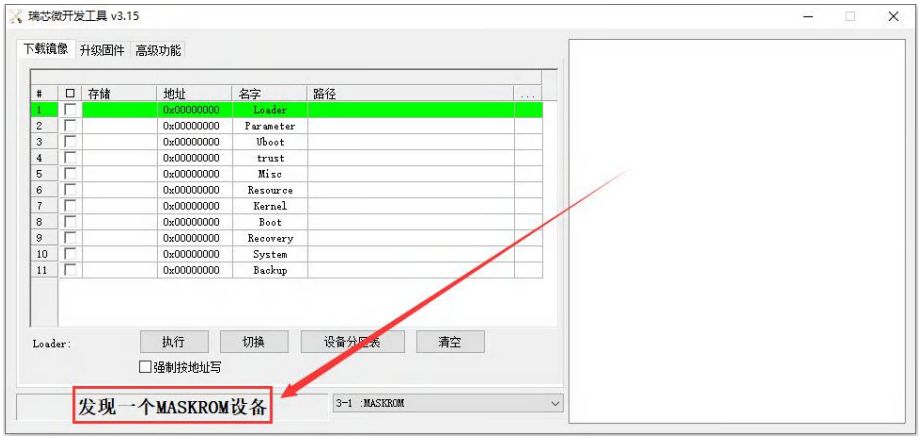
c. Then press and hold the MaskROM button on the development board, the position of the MaskROM button on the development board is shown in the figure below:



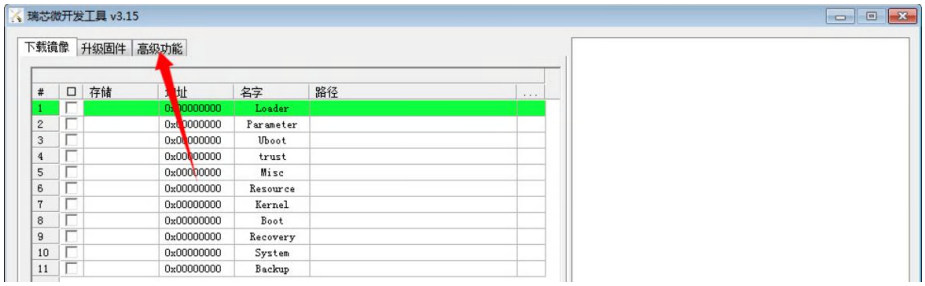
d. Then connect the power supply of the Type-C interface to the development board, and power on, and then release the MaskROM button



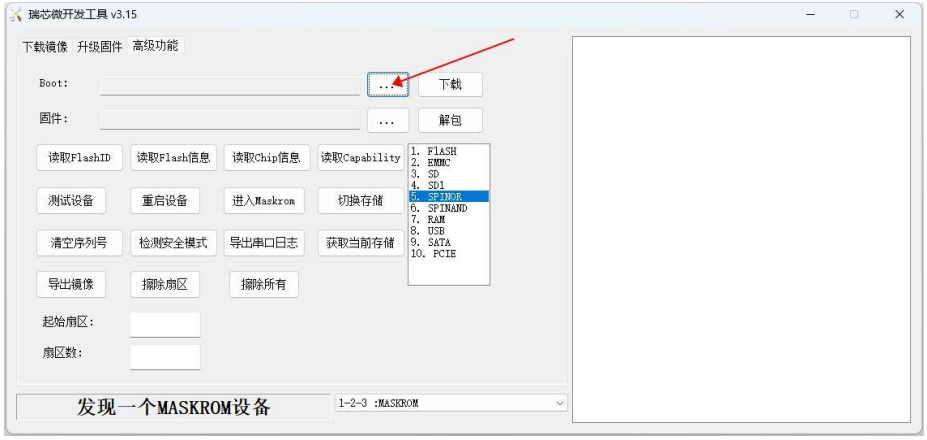
e. If the previous steps are successful, the development board will enter the **MASKROM** mode at this time, and the interface of the burning tool will prompt "**found a MASKROM device**"



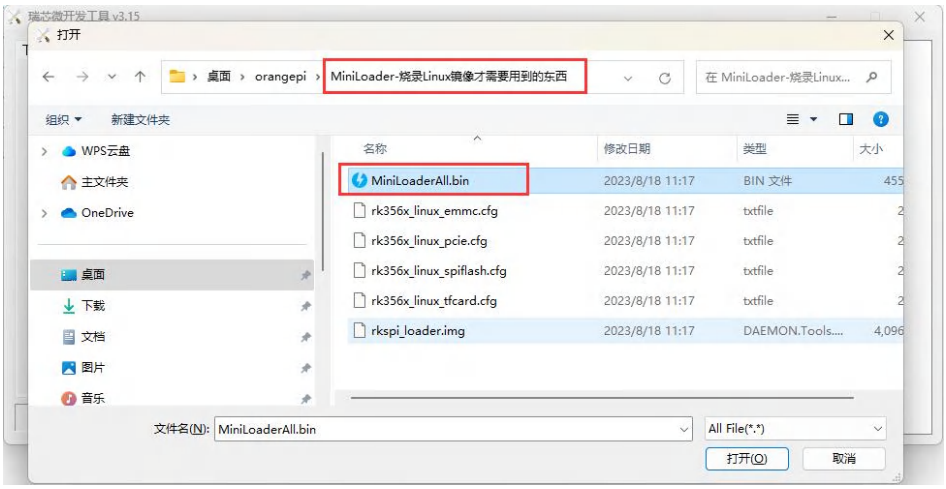

f. Then please select **Advanced Features**



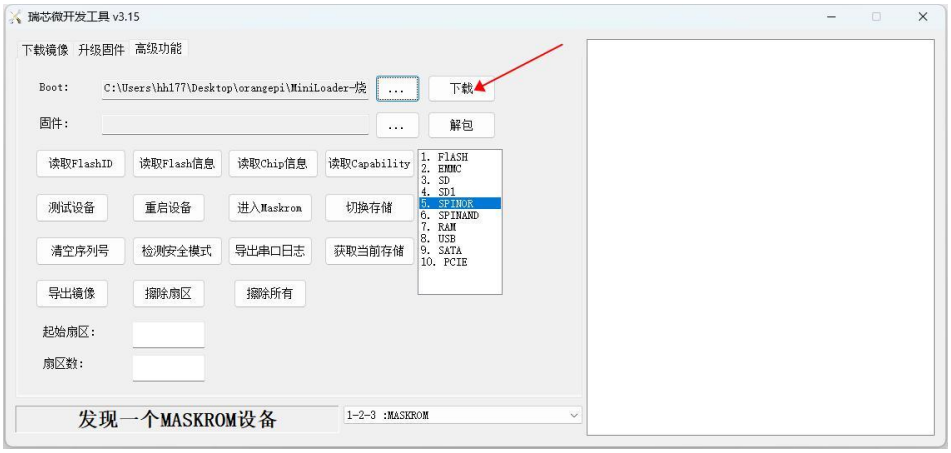

g. Then click the position shown in the figure below



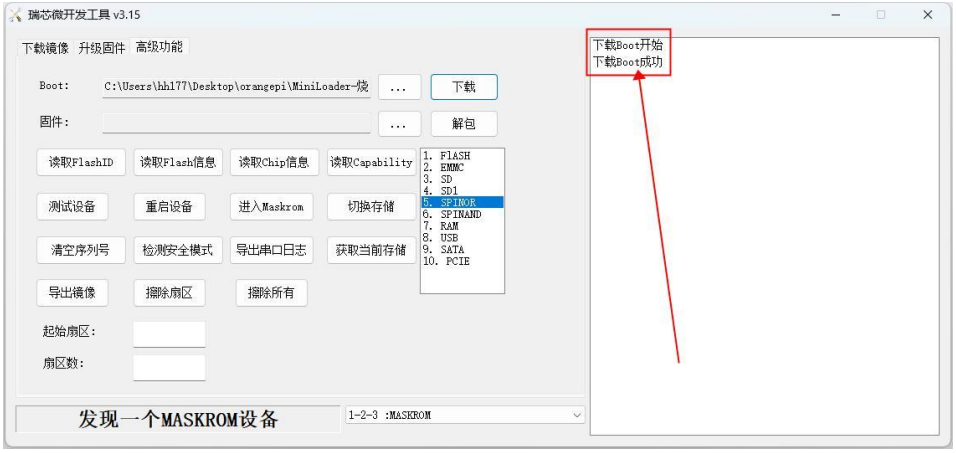

h. Select **MiniLoaderAll.bin** in the **MiniLoader** folder you downloaded earlier, and click Open



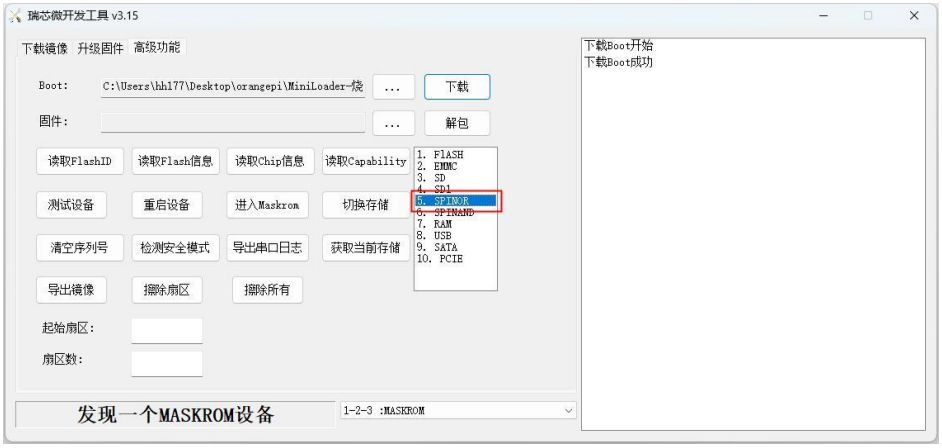

i. Then click **Download**



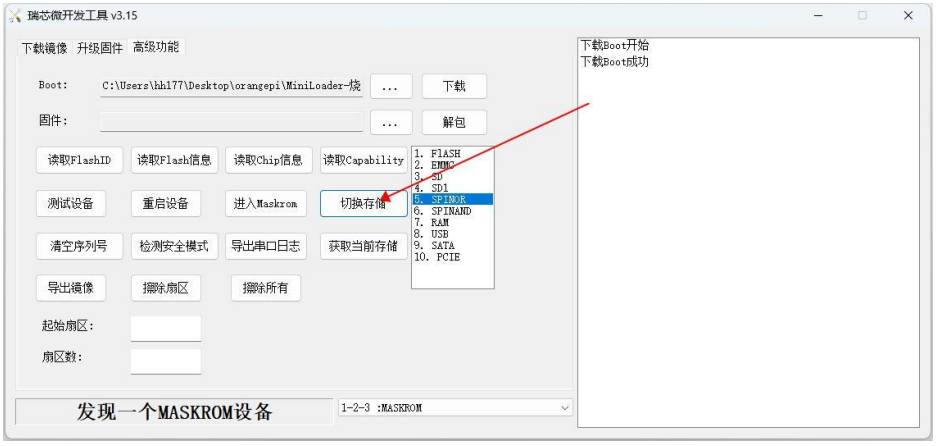

j. The display after downloading **MiniLoaderAll.bin** is shown in the figure below



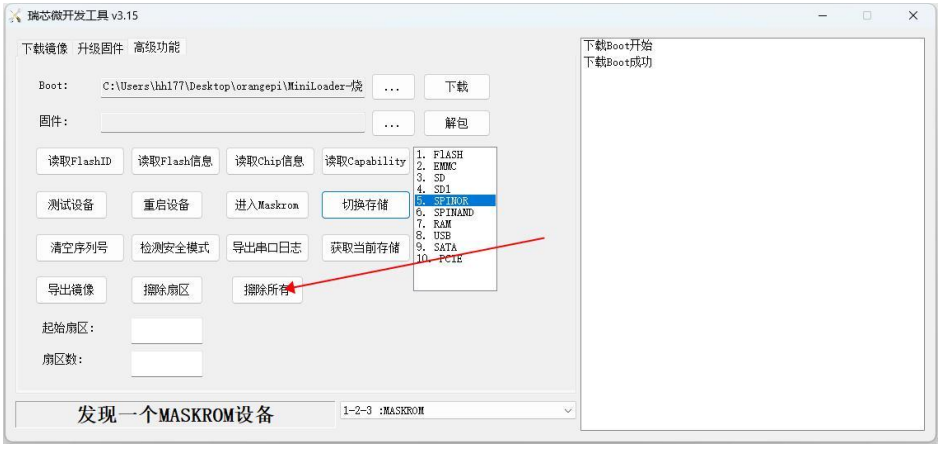

k. Then select the storage device as **SPINOR**



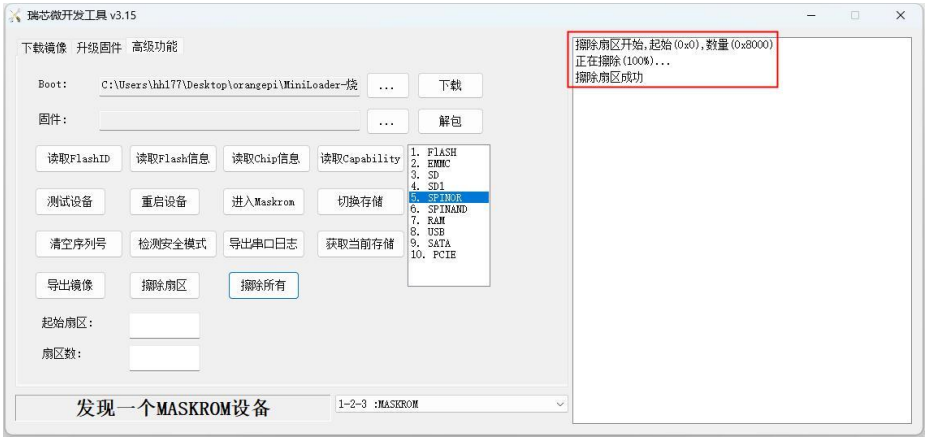

l. Then click Switch **Storage**




m. Then click **Erase All** to start erasing SPIFlash




n. The display log after erasing SPIFlash is shown in the figure below




## 2. 13. Start the Orange Pi development board

1) Insert the TF card with the burned image into the TF card slot of the Orange Pi development board. If the image of SPIFlash+NVMe SSD has been burnt, then there is no need to insert a TF card, just make sure that the NVMe SSD is inserted into the development board normally.

2) The development board has an HDMI interface, and the development board can be connected to a TV or HDMI display through an HDMI-to-HDMI cable. If you have purchased an LCD screen, you can also use the LCD screen to display the system interface of the development board.

3) Connect a USB mouse and keyboard to control the Orange Pi development board.

4) The development board has an Ethernet port, which can be plugged into a network cable for Internet access.

5) Connect a high-quality power adapter with a 5V/3A USB Type-C interface.

**Remember not to plug in a power adapter with a voltage output greater than 5V, as this will burn out the development board.**

**Many unstable phenomena during the power-on and start-up process of the system are basically caused by power supply problems, so a reliable power adapter is very important. If you find that there is a phenomenon of continuous restart during the startup process, please replace the power supply or the Type-C data cable and try again.**

**The Type-C power port does not support PD negotiation.**

**In addition, please do not connect the USB interface of the computer to power the development board.**

6) Then turn on the switch of the power adapter. If everything is normal, you can see the startup screen of the system on the HDMI monitor or LCD screen.

7) If you want to view the output information of the system through the debugging serial port, please use the serial cable to connect the development board to the computer. For the connection method of the serial port, please refer to **the section on how to use the debugging serial port.**

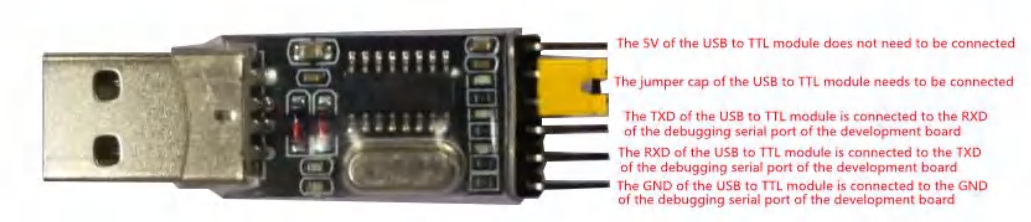
## 2. 14. How to use the debugging serial port

### 2. 14. 1. Connection instruction of debugging serial port

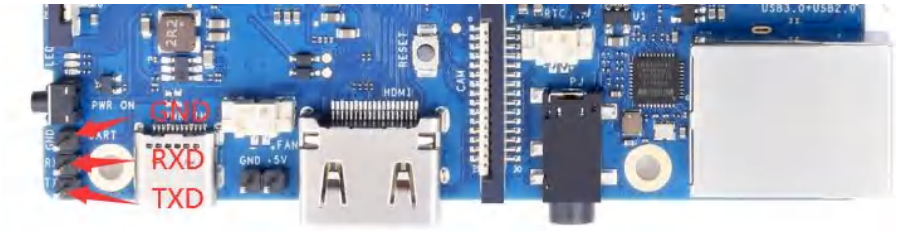
1) First, you need to prepare a 3.3V USB-to-TTL module, and then insert the USB interface end of the USB-to-TTL module into the USB interface of the computer.

**For better compatibility, it is recommended to use CH340 USB to TTL module, please do not use CP2102, PL2303 USB to TTL module.**

**Before purchasing a USB to TTL module, please confirm that the module supports a baud rate of 1500000.**




2) The corresponding relationship between GND, RXD and TXD pins of the debugging serial port of the development board is shown in the figure below



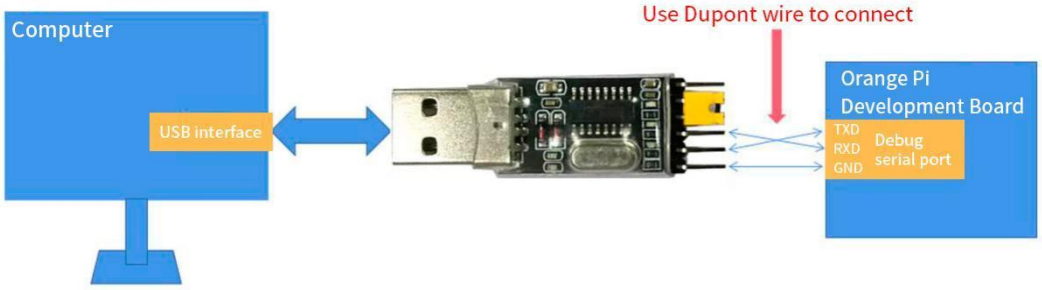
3) The GND, TXD and RXD pins of the USB to TTL module need to be connected to the debugging serial port of the development board through a DuPont line

a. Connect the GND of the USB to TTL module to the GND of the development board

b. The **RX of the USB to TTL module is connected to the TX** of the development board

c. The **TX of the USB to TTL module is connected to the TX** of the development board

4) The schematic diagram of connecting the USB to TTL module to the computer and the Orange Pi development board is as follows




Schematic diagram of connecting the USB to TTL module to the computer and the Orange Pi development board

**The TX and RX of the serial port need to be cross-connected. If you don't want to carefully distinguish the order of TX and RX, you can connect the TX and RX of the serial port casually first. If there is no output in the test, then exchange the order of TX and RX, so there is always a the order is right**

## 2. 14. 2. How to use the debugging serial port on the Ubuntu platform

**There are many serial port debugging software that can be used under Linux, such as putty, minicom, etc. The following demonstrates how to use putty.**

1) First, insert the USB-to-TTL module into the USB interface of the Ubuntu computer. If the connection and recognition of the USB-to-TTL module is normal, you can see the corresponding device node name under **/dev** on the Ubuntu PC. Remember this node name, and then set the serial port software will be used

```
test@test:~$ ls /dev/ttyUSB*
/dev/ttyUSB0
```

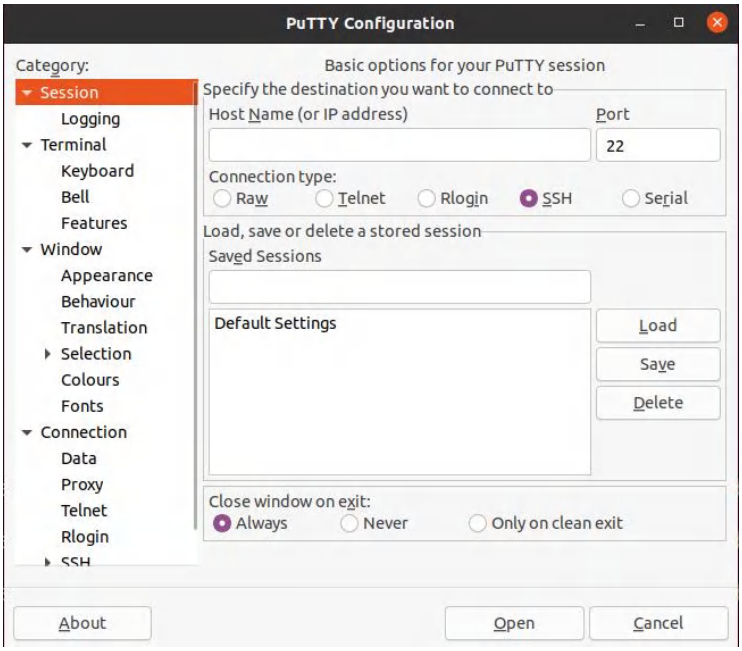
2) Then use the following command to install putty on Ubuntu PC

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

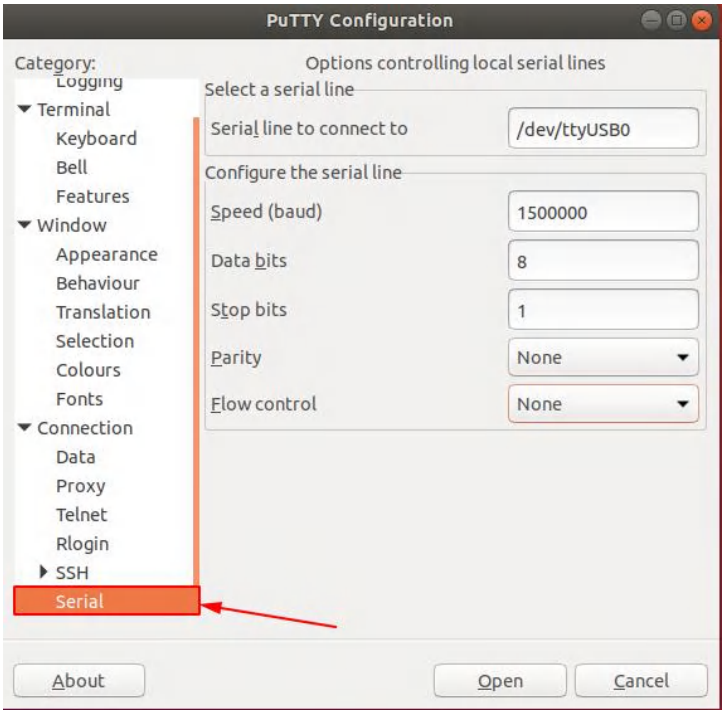
3) Then run putty, **remember to add sudo permission**

```
test@test:~$ sudo putty
```

4) After executing the putty command, the following interface will pop up




5) First select the setting interface of the serial port

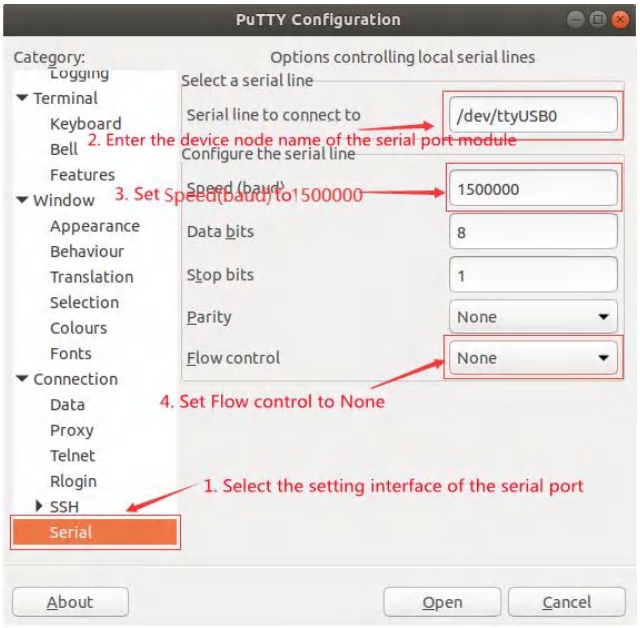



6) Then set the parameters of the serial port

a. Set Serial line to connect toas /dev/ttyUSB0 （(Modified to the corresponding node name, generally /dev/ttyUSB0）

b. Set Speed(baud) as **1500000**（Serial port baud rate）

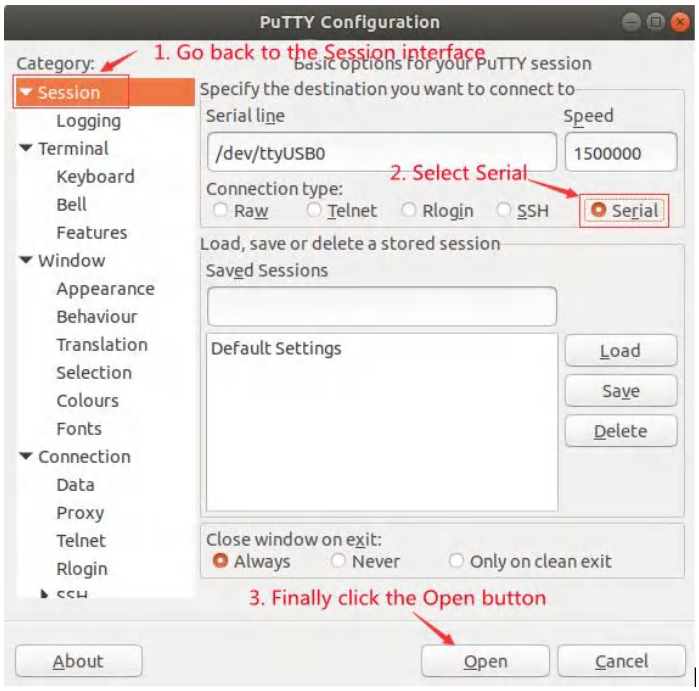
c. Set Flow control as None



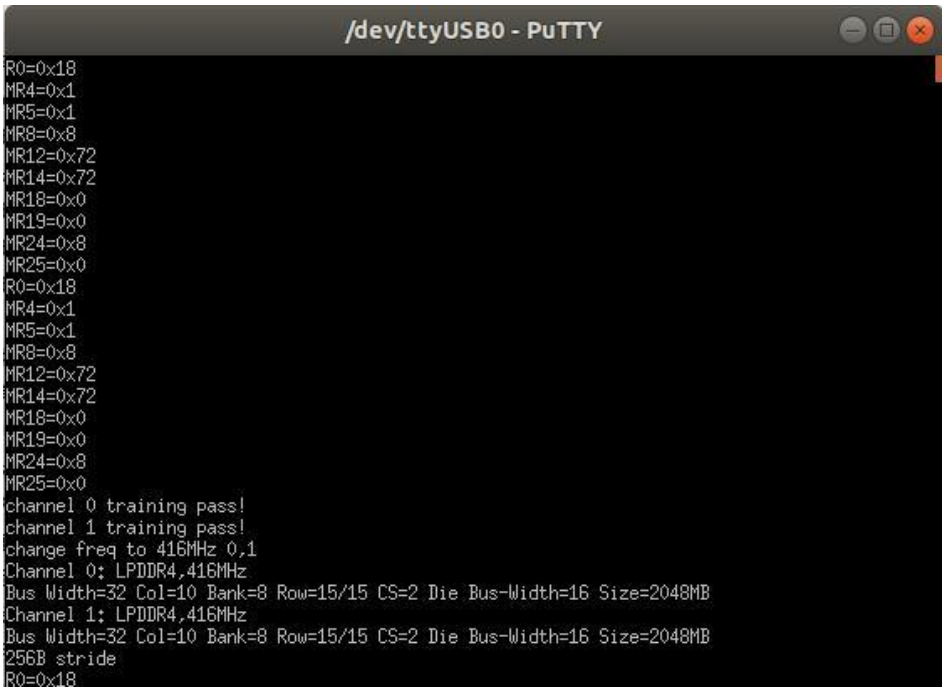
7) After setting the setting interface of the serial port, return to the Session interface

a. First select the Connection type as Serial

b. Then click the Open button to connect to the serial port




8) After starting the development board, you can see the Log information output by the system from the opened serial port terminal



## 2. 14. 3. How to use the debugging serial port on Windows platform

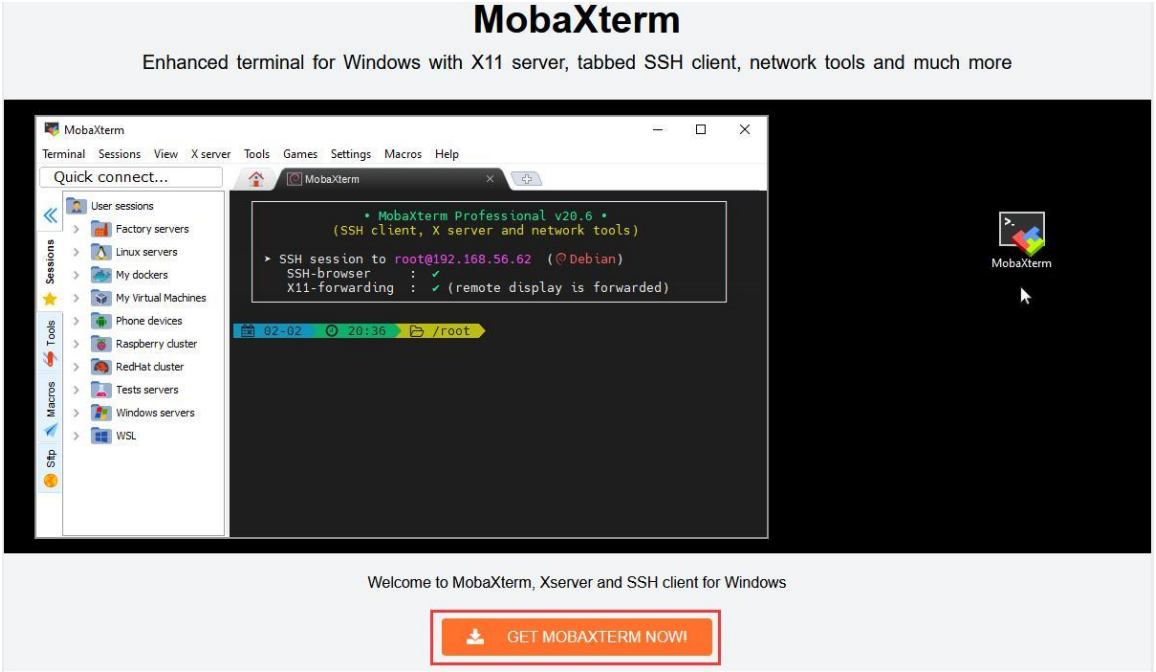
**There are many serial port debugging software that can be used under Windows, such as SecureCRT, MobaXterm, etc. The following demonstrates how to use MobaXterm. This software has a free version and can be used without buying a serial number.**

1) Download MobaXterm

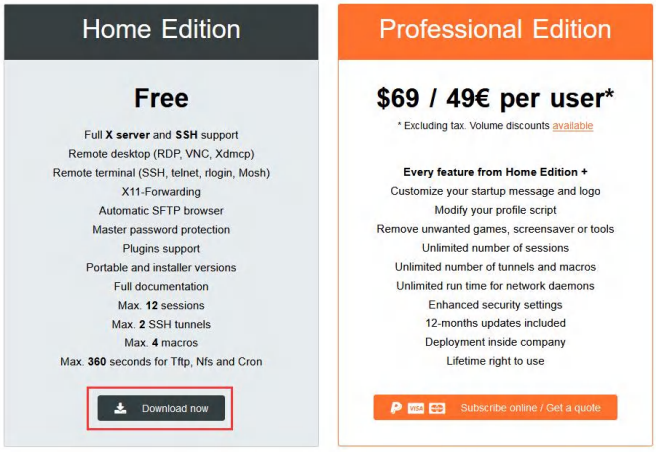
a. Download MobaXterm website as follows

**https://mobaxterm.mobatek.net**

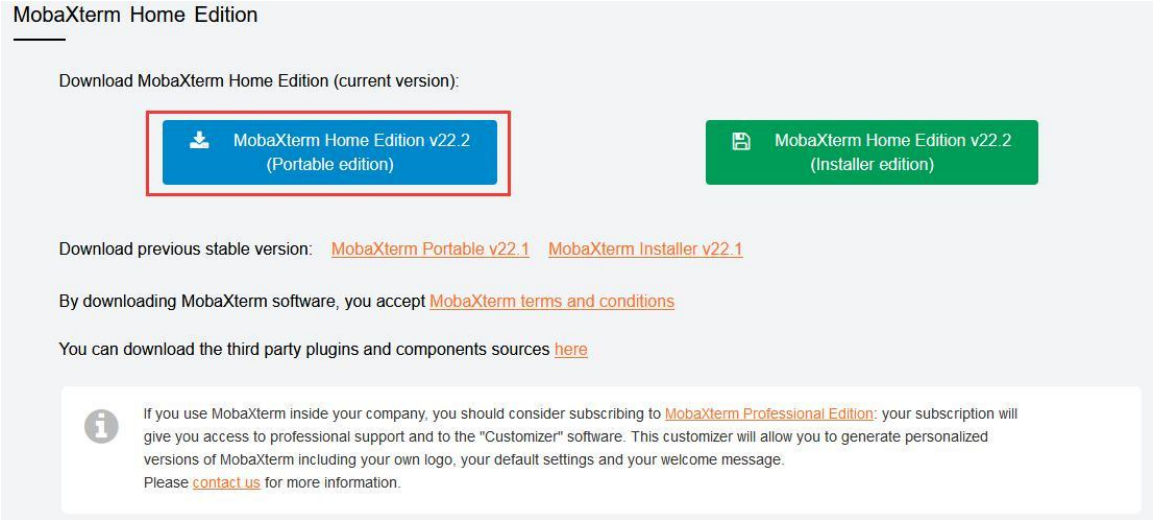
b. After entering the MobaXterm download page, click **GET XOBATERM NOW!**



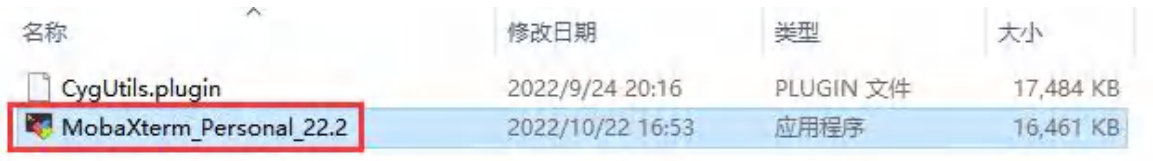

c. Then choose to download the Home version



d. Then select the Portable version. After downloading, you don't need to install it, just open it and use it

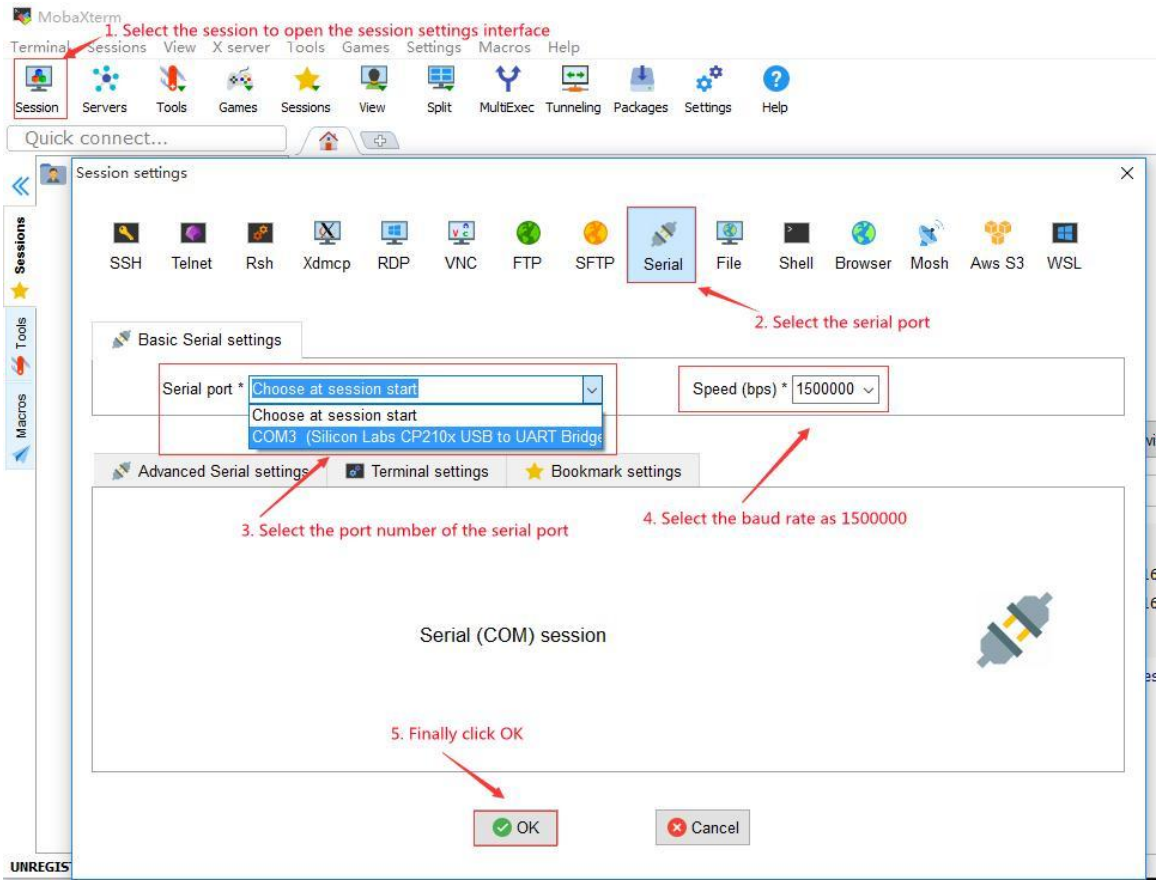



2) After downloading, use decompression software to decompress the downloaded compressed package, you can get the executable software of MobaXterm, and then double-click to open

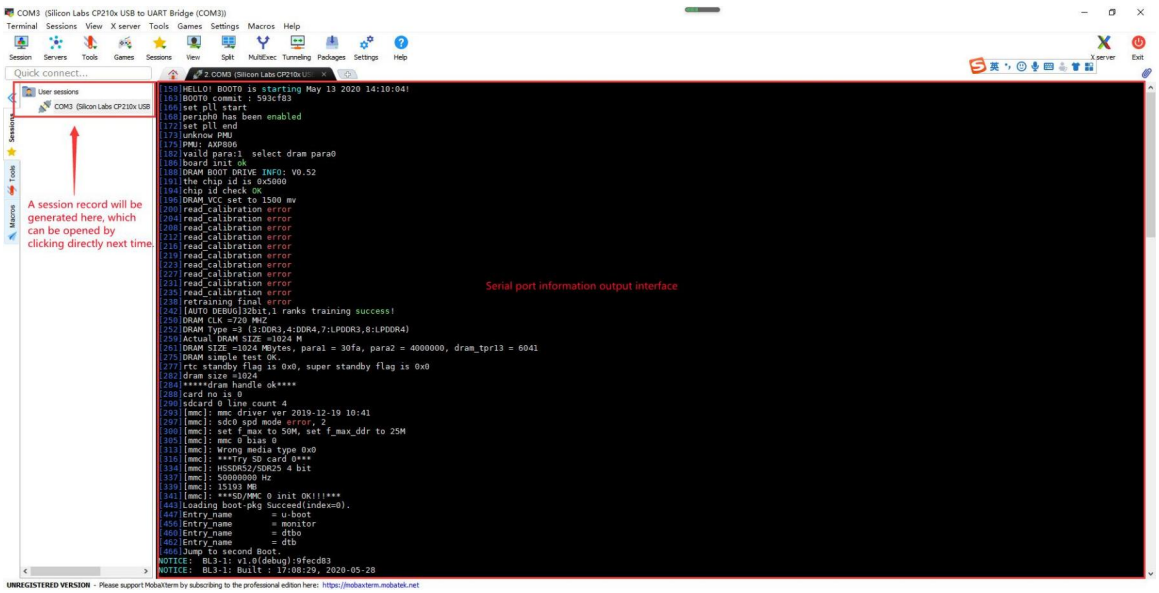



3) After opening the software, the steps to set up the serial port connection are as follows

a. Open the session settings interface
b. Select the serial port type
c. Select the port number of the serial port (select the corresponding port number according to the actual situation)
d. Select the baud rate of the serial port as **1500000**
e. Finally click the "**OK**" button to complete the setting




4) After clicking the "**OK**" button, you will enter the following interface. At this time, start the development board and you can see the output information of the serial port



## 2. 15. Instructions for using the 5v pin in the 40pin interface of the development board to supply power

**The power supply method we recommend for the development board is to use the 5V/3A Type C interface power cord to plug into the Type-C power interface of the development board for power supply. If you need to use the 5V pin in the 40pin interface to power the development board, please make sure that the power cord and power adapter used can meet the power supply requirements of the development board. If the use is unstable, please switch back to the Type-C power supply.**

1) First, you need to prepare a power cord as shown in the figure below


**Please purchase the power cord shown in the picture above by yourself**

2) Use the 5V pin in the 40pin interface to supply power to the development board. The connection method of the power line is as follows

a. The USB A port of the power cord shown in the above picture needs to be plugged into the 5V/3A power adapter connector (**please do not plug into the computer's USB port for power supply**)

b. The red DuPont line needs to be plugged into the 5V pin of the development board 40pin

c. The black Dupont wire needs to be inserted into the GND pin of the 40pin interface

d. The position of the 5V pin and GND pin of the 40pin interface on the development board is shown in the figure below, **remember not to reverse the connection**

# 3. Instructions for use of Ubuntu/Debian Server and Xfce desktop system

**The content of this chapter is written based on the images of the Linux server version and the xfce desktop version.**

## 3.1. Supported Linux image types and kernel versions

| Linux image type | kernel version | server version | desktop version |
|---|---|---|---|
| Debian 11 - Bullseye | Linux5.10 | support | support |
| Ubuntu 20.04 - Focal | Linux5.10 | support | support |
| Ubuntu 22.04 - Jammy | Linux5.10 | support | support |
| Debian 11 - Bullseye | Linux6.6 | support | support |
| Debian12 - Bookworm | Linux6.6 | support | support |
| Ubuntu 22.04 - Jammy | Linux6.6 | support | support |

## 3.2. Linux System adaptation

### 3.2.1. Linux5.10 system adaptation situation

| Function | Debian11 | Ubuntu20.04 | Ubuntu22.04 |
|---|---|---|---|
| USB2.0x3 | OK | OK | OK |
| USB3.0x1 | OK | OK | OK |
| M.2 NVMe SSD Start | OK | OK | OK |
| WIFI | OK | OK | OK |
| Bluetooth | OK | OK | OK |
| GPIO（40pin） | OK | OK | OK |
| UART（40pin） | OK | OK | OK |
| SPI（40pin） | OK | OK | OK |
| I2C（40pin） | OK | OK | OK |
| PWM（40pin） | OK | OK | OK |
| PWM fan interface | OK | OK | OK |

| | | | |
|---|---|---|---|
| 3pin Debug serial port | OK | OK | OK |
| EMMC | OK | OK | OK |
| TF card start | OK | OK | OK |
| HDMI Video | OK | OK | OK |
| HDMI Audio | OK | OK | OK |
| OV5647 Camera | The kernel driver is OK, 3A is not adjusted | | |
| LCD | OK | OK | OK |
| Edp Display | OK | OK | OK |
| Gigabit Ethernet port | OK | OK | OK |
| Network port status light | OK | OK | OK |
| headphone playback | OK | OK | OK |
| headphone recording | OK | OK | OK |
| LED Light | OK | OK | OK |
| RTC | OK | OK | OK |
| GPU | OK | OK | OK |
| NPU | OK | OK | OK |
| VPU | OK | OK | OK |
| watchdog test | OK | OK | OK |
| Chromium Hard solution video | OK | OK | OK |

### 3. 2. 2. Linux6.6 system adaptation situation

| Function | Debian11 | Debian12 | Ubuntu22.04 |
|---|---|---|---|
| USB2.0x3 | OK | OK | OK |
| USB3.0x1 | OK | OK | OK |
| M.2 NVMe SSD Start | OK | OK | OK |
| WIFI | OK | OK | OK |
| Bluetooth | OK | OK | OK |
| GPIO（40pin） | OK | OK | OK |
| UART（40pin） | OK | OK | OK |
| SPI（40pin） | OK | OK | OK |
| I2C（40pin） | OK | OK | OK |
| PWM（40pin） | OK | OK | OK |
| PWM fan interface | OK | OK | OK |
| 3pin Debug serial port | OK | OK | OK |

| | | | |
|---|---|---|---|
| **EMMC** | **OK** | **OK** | **OK** |
| **TF card start** | **OK** | **OK** | **OK** |
| **HDMI Video** | **OK** | **OK** | **OK** |
| **HDMI Audio** | **OK** | **OK** | **OK** |
| **OV5647 Camera** | **NO** | **NO** | **NO** |
| **LCD** | **NO** | **NO** | **NO** |
| **eDP Display** | **NO** | **NO** | **NO** |
| **Gigabit Ethernet port** | **OK** | **OK** | **OK** |
| **Network port status light** | **OK** | **OK** | **OK** |
| **headphone playback** | **NO** | **NO** | **NO** |
| **headphone recording** | **NO** | **NO** | **NO** |
| **LED Light** | **OK** | **OK** | **OK** |
| **RTC** | **OK** | **OK** | **OK** |
| **GPU** | **OK** | **OK** | **OK** |
| **NPU** | **NO** | **NO** | **NO** |
| **VPU** | **NO** | **NO** | **NO** |
| **watchdog test** | **NO** | **NO** | **NO** |

## 3. 3. The format of Linux commands in this manual

1) In this manual, all commands that need to be entered in the Linux system will be marked with the following box

```
```

As shown below, the content in the yellow box indicates the content that needs special attention, except for the commands in it.

```
```

2) Description of the prompt type in front of the command

a. The prompt in front of the command refers to the content of the red part in the box below, which is not part of the Linux command, so when entering the command in the Linux system, please do not enter the content of the red font part

```
orangepi@orangepi:~$ sudo apt update
root@orangepi:~# vim /boot/boot.cmd
test@test:~$ ssh root@192.168.1.xxx
root@test:~# ls
```

b. **root@orangepi:~$** The prompt indicates that this command is entered in **the Linux system of the development board**. The $ at the end of the prompt indicates that the current user of the system is a normal user. When executing a privileged command, you need to add **sudo**

c. **root@orangepi:~#** The prompt indicates that this command is entered in the Linux system of the development board, and the # at the end of the prompt indicates that the current user of the system is the root user, who can execute any desired command

d. **test@test:~$** The prompt indicates that this command is entered in the Ubuntu PC or Ubuntu virtual machine, not in the Linux system of the development board. The $ at the end of the prompt indicates that the current user of the system is an ordinary user. When executing privileged commands, sudo needs to be added **sudo**

e. **root@test:~#** The prompt indicates that this command is entered in the Ubuntu PC or Ubuntu virtual machine, not in the Linux system of the development board. The # at the end of the prompt indicates that the current user of the system is the root user and can execute any command you want

3) What are the commands that need to be entered?

a. As shown below, **the black bold part** is the command that needs to be input, and the content below the command is the output content (some commands have output, some may not), and this part of the content does not need to be input

```
root@orangepi:~# cat /boot/orangepiEnv.txt
verbosity=7
bootlogo=false
console=serial
```

b. As shown below, some commands cannot be written in one line and will be placed on the next line. As long as the black and bold parts are all commands that need to be input. When these commands are entered into one line, the last "\" of each line needs to be removed, this is not part of the command. In addition, there are spaces in different parts of the command, please don’t miss it

```
orangepi@orangepi:~$ echo \
"deb [arch=$(dpkg --print-architecture) \
signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] \
https://download.docker.com/Linux/debian \
$(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
```

## 3. 4. Linux system login instructions

### 3. 4. 1. Linux system default login account and password

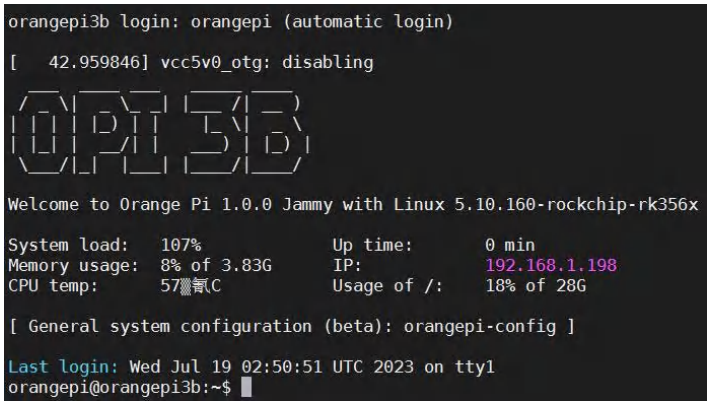
| Account | Passport |
|---|---|
| root | orangepi |
| orangepi | orangepi |

**Note that when entering the password, the specific content of the entered password will not be displayed on the screen, please do not think that there is any fault, just press Enter after inputting.**

**When the wrong password is prompted, or there is a problem with the ssh connection, please note that as long as you are using the Linux image provided by Orange Pi, please do not suspect that the above password is wrong, but look for other reasons.**

### 3. 4. 2. How to set automatic terminal login in Linux system

1) The Linux system automatically logs in to the terminal by default, and the default login user name is **orangepi**



2) Use the following command to set the root user to automatically log in to the terminal

```
orangepi@orangepi:~$ sudo auto_login_cli.sh root
```

3) Use the following command to disable automatic login terminal

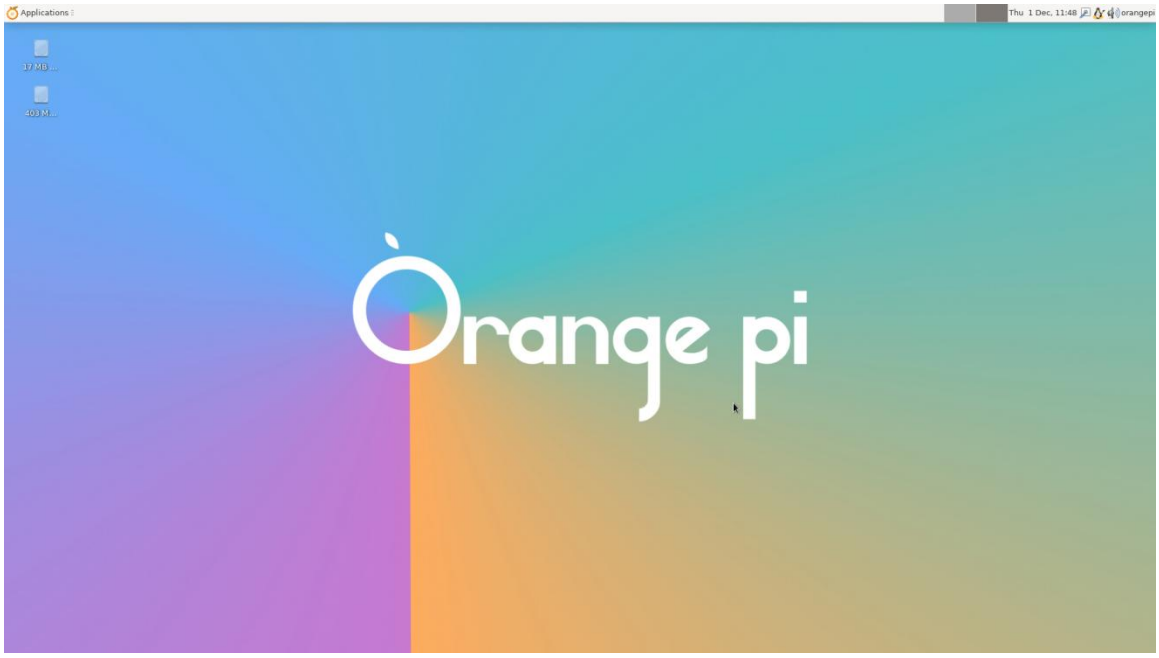
```
orangepi@orangepi:~$ sudo auto_login_cli.sh -d
```

4) Use the following command to set the orangepi user to automatically log in to the terminal again

```
orangepi@orangepi:~$ sudo auto_login_cli.sh orangepi
```

### 3. 4. 3. Instructions for automatic login of Linux desktop version system

1) After the desktop system starts, it will automatically log in to the desktop without entering a password




2) Run the following command to prohibit the desktop system from automatically logging into the desktop

```
orangepi@orangepi:~$ sudo disable_desktop_autologin.sh
```

3) Then restart the system and a login dialog box will appear, at which point a **password** is required to enter the system


### 3. 4. 4. The setting method of root user automatic login in Linux desktop version system

1) Execute the following command to set the desktop system to automatically log in as the root user

```
orangepi@orangepi:~$ sudo desktop_login.sh root
```

2) Then restart the system, and the root user will automatically log in to the desktop


**Note that if you log in to the desktop system as the root user, you cannot use pulseaudio in the upper right corner to manage audio devices.**

**Also note that this is not a bug, since pulseaudio is not allowed to run as root.**

3) Execute the following command to set the desktop system to log in automatically with the orangepi user again

```
orangepi@orangepi:~$ sudo desktop_login.sh orangepi
```

### 3.4.5. The method of disabling the desktop in the Linux desktop version system

1) First enter the following command on the command line，**Please remember to add sudo permission**

```
orangepi@orangepi:~$ sudo systemctl disable lightdm.service
```

2) Then restart the Linux system and you will find that the desktop will not be displayed

```
orangepi@orangepi:~$ sudo reboot
```

3) The steps to reopen the desktop are as follows:

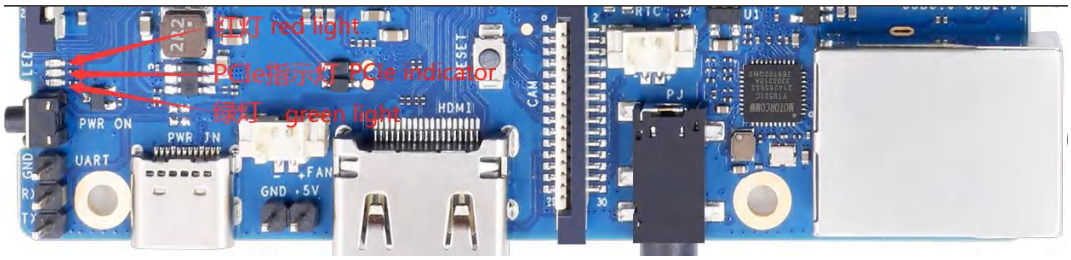
a. First enter the following command on the command line，**Please remember to add sudo permission**

```
orangepi@orangepi:~$ sudo systemctl start lightdm.service
```

b. After the command is executed, the desktop will be displayed

## 3.5. Onboard LED Light Test Instructions

1) There are three LED lights on the development board, one green light, one red light, and one PCIe light. The location is shown in the figure below:




**2) As long as the development board is powered on, the red LED light will always be on, which is controlled by the hardware and cannot be turned off by the software**

3) The green LED light will keep blinking after the kernel is started, which is controlled by software.

4) The PCIe indicator will flash when there is data transmission on the PCIe interface.

5) The method of setting the green light on and off and flashing is as follows

**Note that the following operations should be performed under the root user.**

a. First enter the setting directory of the green light

```
root@orangepi:~# cd /sys/class/leds/status_led
```

b. The command to set the green light to stop flashing is as follows

```
root@orangepi:/sys/class/leds/status_led# echo none > trigger
```

c. The command to set the green light to be on is as follows

```
root@orangepi:/sys/class/leds/status_led# echo default-on > trigger
```

d. The command to set the green light to flash is as follows

```
root@orangepi:/sys/class/leds/status_led# echo default-on > trigger
```

## 3. 6. Network connection test

### 3. 6. 1. Ethernet port test

1) First, insert one end of the network cable into the Ethernet interface of the development board, and connect the other end of the network cable to the router, and ensure that the network is unblocked

2) After the system starts, it will automatically assign an IP address to the Ethernet card through DHCP, **No other configuration is required**

3) The command to view the IP address in the Linux system of the development board is as follows:

**Note that in the following command, Debian12 of linux5.10 needs to modify eth0 to end1, and Debian12 of linux6.6 needs to modify eth0 to end0.**

```
orangepi@orangepi:~$ ip addr show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP group default qlen 1000
    link/ether 4a:fe:2b:3d:17:1c brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.150/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
       valid_lft 43150sec preferred_lft 43150sec
    inet6 fe80::9a04:3703:faed:23be/64 scope link noprefixroute
```

```
valid_lft forever preferred_lft forever
```

**When using ifconfig to view the IP address, if the following information is displayed, it is because sudo is not added. The correct command is：sudo ifconfig**

orangepi@orangepi:~$ **ifconfig**
Command 'ifconfig' is available in the following places
 * /sbin/ifconfig
 * /usr/sbin/ifconfig
The command could not be located because '/sbin:/usr/sbin' is not included in the PATH environment variable.
This is most likely caused by the lack of administrative privileges associated with your user account.
ifconfig: command not found

**There are three ways to check the IP address after the development board starts:**

**1. Connect the HDMI display, then log in to the system and use the ip addr show eth0 command to view the IP address**

**2. Enter the ip addr show eth0 command in the debugging serial terminal to view the IP address**

**3. If there is no debugging serial port and no HDMI display, you can also check the IP address of the development board's network port through the router's management interface. However, in this method, some people often cannot see the IP address of the development board normally. If you can't see it, the debug method looks like this:**

**A）First check whether the Linux system has started normally. If the green light of the development board is blinking, it is generally started normally. If only the red light is on, it means that the system has not started normally;**

**B）Check whether the network cable is plugged in tightly, or try another network cable;**

**C）Try another router (I have encountered many problems with the router, such as the router cannot assign the IP address normally, or the IP address has been assigned normally but cannot be seen in the router);**

**D）If there is no router to replace, you can only connect to an HDMI display or use**

**the debugging serial port to view the IP address**

**In addition, it should be noted that the development board DHCP automatically assigns an IP address without any settings.**

4) The command to test the network connectivity is as follows, the **ping** command can be interrupted through the shortcut key of **Ctrl+C** (Here is an uppercase I, not a lowercase L)

```
orangepi@orangepi:~$ ping www.baidu.com -I eth0
PING www.a.shifen.com (14.215.177.38) from 192.168.1.12 eth0: 56(84) bytes of data.
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=1 ttl=56 time=6.74 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=2 ttl=56 time=6.80 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=3 ttl=56 time=6.26 ms
64 bytes from 14.215.177.38 (14.215.177.38): icmp_seq=4 ttl=56 time=7.27 ms
^C
--- www.a.shifen.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3002ms
rtt min/avg/max/mdev = 6.260/6.770/7.275/0.373 ms
```

### 3.6.2. WIFI connection test

**Please do not connect to WIFI by modifying the /etc/network/interfaces configuration file. There will be problems connecting to the WIFI network in this way.**

#### 3.6.2.1. The server image connects to WIFI through commands

**When the development board is not connected to Ethernet, not connected to HDMI display, but only connected to the serial port, it is recommended to use the commands demonstrated in this section to connect to the WIFI network. Because nmtui can only display characters in some serial port software (such as minicom), and cannot display the graphical interface normally. Of course, if the development board is connected to an Ethernet or HDMI display, you can also use the commands demonstrated in this section to connect to the WIFI network.**

1) First log in to the Linux system, there are the following three ways

a. If the development board is connected with a network cable, you can **remotely**

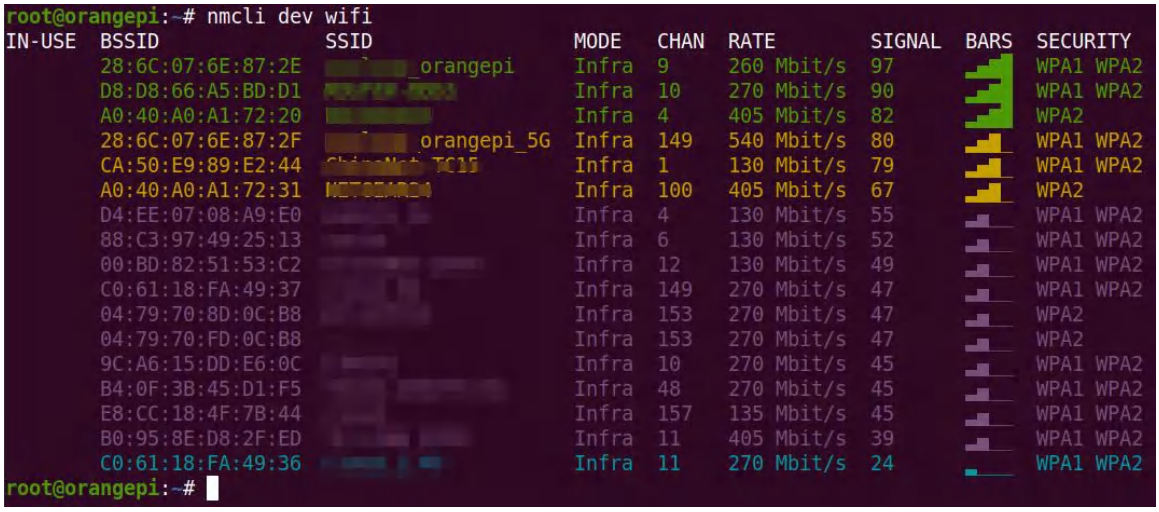
**log in to the Linux system through ssh**

b. If the development board is connected to the debugging serial port, you can use the serial port terminal to log in to the Linux system

c. If the development board is connected to the HDMI display, you can log in to the Linux system through the terminal displayed on the HDMI

2) First use the nmcli dev wifi command to scan the surrounding WIFI hotspots

```
orangepi@orangepi:~$ nmcli dev wifi
```




3) Then use the nmcli command to connect to the scanned WIFI hotspot, where:

a. **wifi_name** needs to be replaced with the name of the WIFI hotspot you want to connect to

b. **wifi_passwd** needs to be replaced with the password of the WIFI hotspot you want to connect to

```
orangepi@orangepi:~$ nmcli dev wifi connect wifi_name password wifi_passwd
Device 'wlan0' successfully activated with 'cf937f88-ca1e-4411-bb50-61f402eef293'.
```

4) You can view the IP address of wifi through the **ip addr show wlan0** command

```
orangepi@orangepi:~$ ip addr show wlan0
11: wlan0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 23:8c:d6:ae:76:bb brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.11/24 brd 192.168.1.255 scope global dynamic noprefixroute wlan0
       valid_lft 259192sec preferred_lft 259192sec
    inet6 240e:3b7:3240:c3a0:c401:a445:5002:ccdd/64 scope global dynamic noprefixroute
       valid_lft 259192sec preferred_lft 172792sec
```

```
    inet6 fe80::42f1:6019:a80e:4c31/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

5) Use the **ping** command to test the connectivity of the wifi network, and the **ping** command can be interrupted through the shortcut key **Ctrl+C**

```
orangepi@orangepi:~$ ping www.orangepi.org -I wlan0
PING www.orangepi.org (182.92.236.130) from 192.168.1.49 wlan0: 56(84) bytes of data.
64 bytes from 182.92.236.130 (182.92.236.130): icmp_seq=1 ttl=52 time=43.5 ms
64 bytes from 182.92.236.130 (182.92.236.130): icmp_seq=2 ttl=52 time=41.3 ms
64 bytes from 182.92.236.130 (182.92.236.130): icmp_seq=3 ttl=52 time=44.9 ms
64 bytes from 182.92.236.130 (182.92.236.130): icmp_seq=4 ttl=52 time=45.6 ms
64 bytes from 182.92.236.130 (182.92.236.130): icmp_seq=5 ttl=52 time=48.8 ms
^C
--- www.orangepi.org ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 41.321/44.864/48.834/2.484 ms
```

### 3.6.2.2. The server image connects to WIFI in a graphical way

1) First log in to the Linux system, there are the following three ways

a. If the development board is connected with a network cable, you can remotely log in to **the Linux system through ssh**

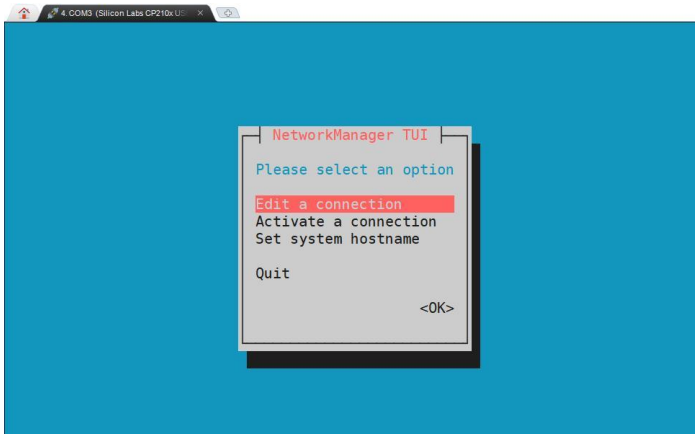
b. If the development board is connected to the debugging serial port, you can use the serial port terminal to log in to the Linux system (please use MobaXterm for the serial port software, and the minicom cannot display the graphical interface)

c. If the development board is connected to the HDMI display, you can log in to the Linux system through the HDMI display terminal

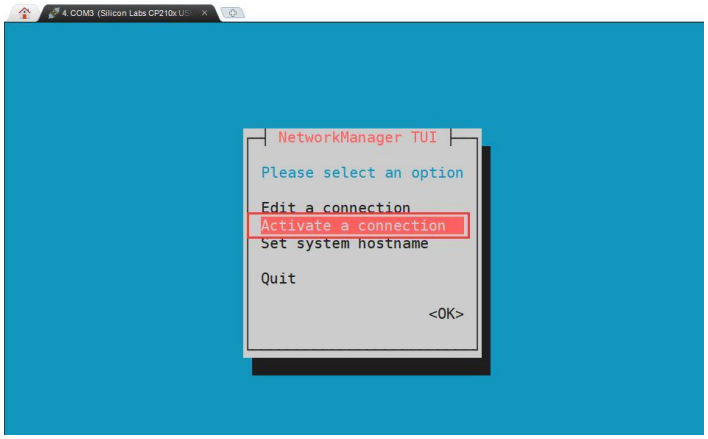
2) Then enter the nmtui command in the command line to open the wifi connection interface

```
orangepi@orangepi:~$ nmtui
```

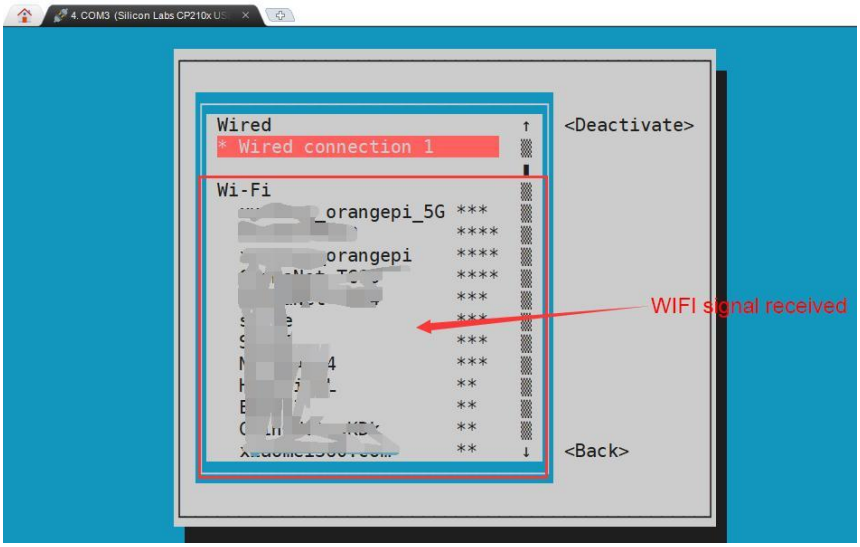
3) Enter the nmtui command to open the interface as shown below



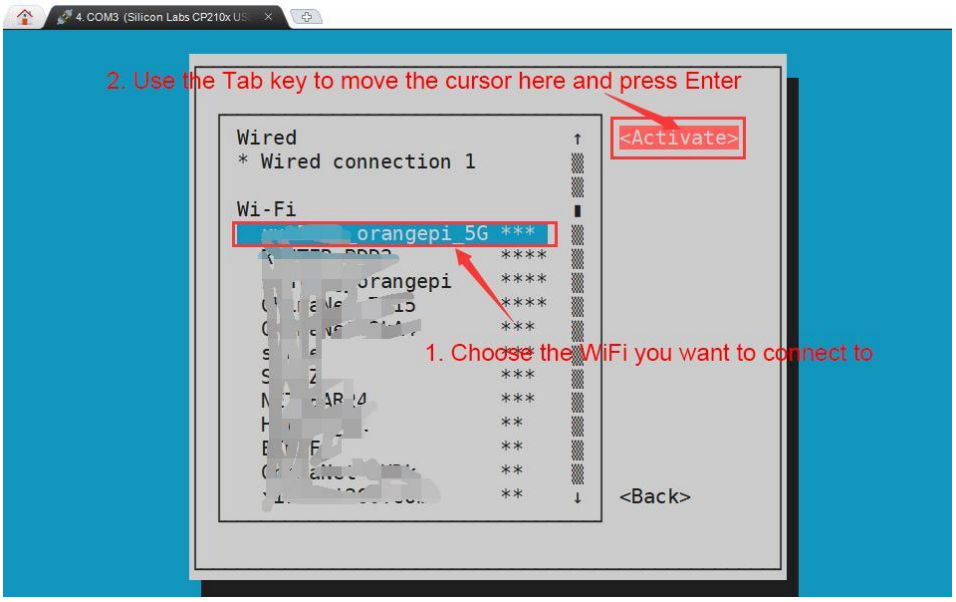

4) Select **Activate a connect** and press Enter



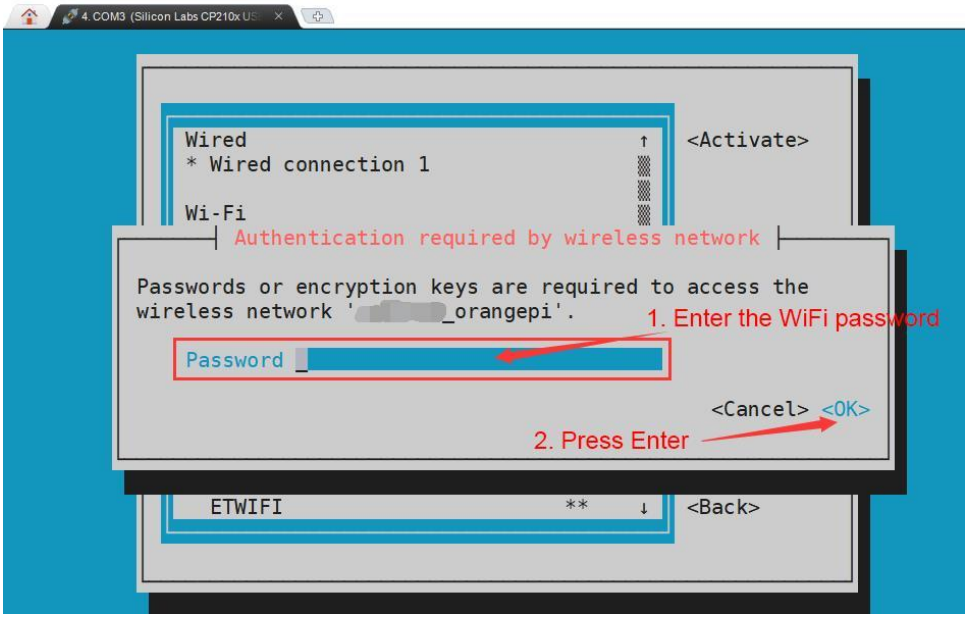

5) Then you can see all the searched WIFI hotspots



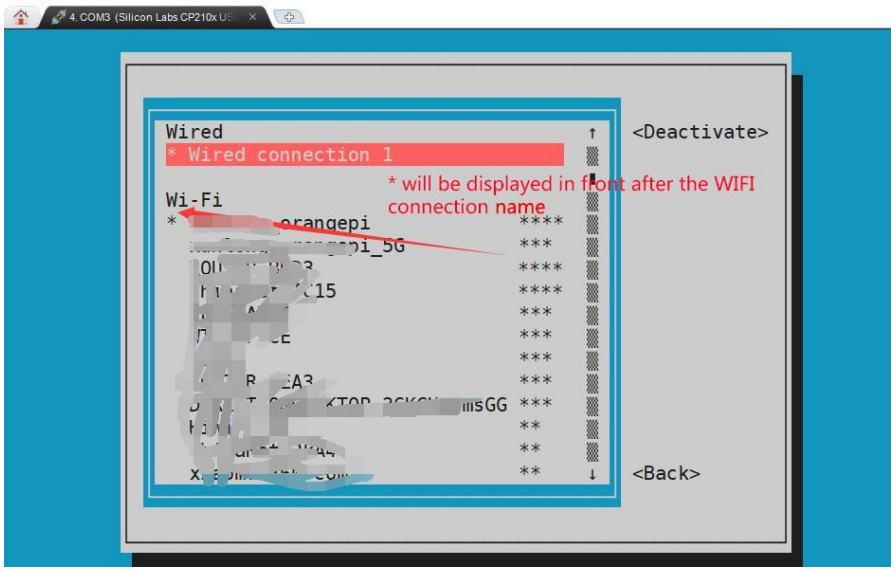
6) Select the WIFI hotspot you want to connect to, then use the Tab key to position the cursor on **Activate** and press Enter




7) Then a dialog box for entering a password will pop up, enter the corresponding password in **Password** and press Enter to start connecting to WIFI




8) After the WIFI connection is successful, a "*" will be displayed in front of the connected WIFI name




9) You can view the IP address of wifi through the **ip addr show wlan0** command

```
orangepi@orangepi:~$ ip addr show wlan0
11: wlan0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 24:8c:d3:aa:76:bb brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.11/24 brd 192.168.1.255 scope global dynamic noprefixroute wlan0
       valid_lft 259069sec preferred_lft 259069sec
    inet6 240e:3b7:3240:c4a0:c401:a445:5002:ccdd/64 scope global dynamic noprefixroute
       valid_lft 259071sec preferred_lft 172671sec
    inet6 fe80::42f1:6019:a80e:4c31/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

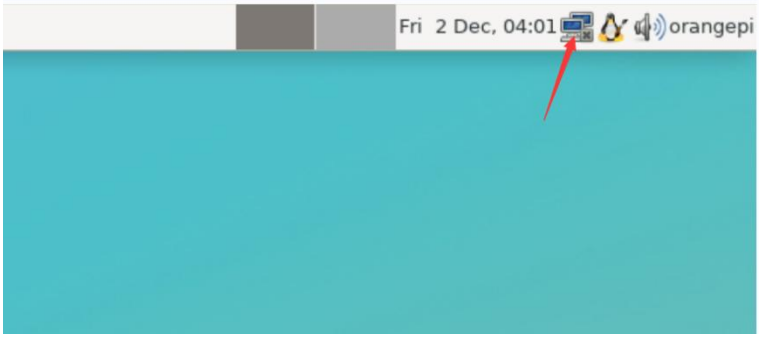
10) Use the **ping** command to test the connectivity of the wifi network, and the **ping** command can be interrupted through the shortcut key **Ctrl+C**

```
orangepi@orangepi:~$ ping www.orangepi.org -I wlan0
PING www.orangepi.org (182.92.236.130) from 192.168.1.49 wlan0: 56(84) bytes of data.
64 bytes from 182.92.236.130 (182.92.236.130): icmp_seq=1 ttl=52 time=43.5 ms
64 bytes from 182.92.236.130 (182.92.236.130): icmp_seq=2 ttl=52 time=41.3 ms
64 bytes from 182.92.236.130 (182.92.236.130): icmp_seq=3 ttl=52 time=44.9 ms
64 bytes from 182.92.236.130 (182.92.236.130): icmp_seq=4 ttl=52 time=45.6 ms
64 bytes from 182.92.236.130 (182.92.236.130): icmp_seq=5 ttl=52 time=48.8 ms
```

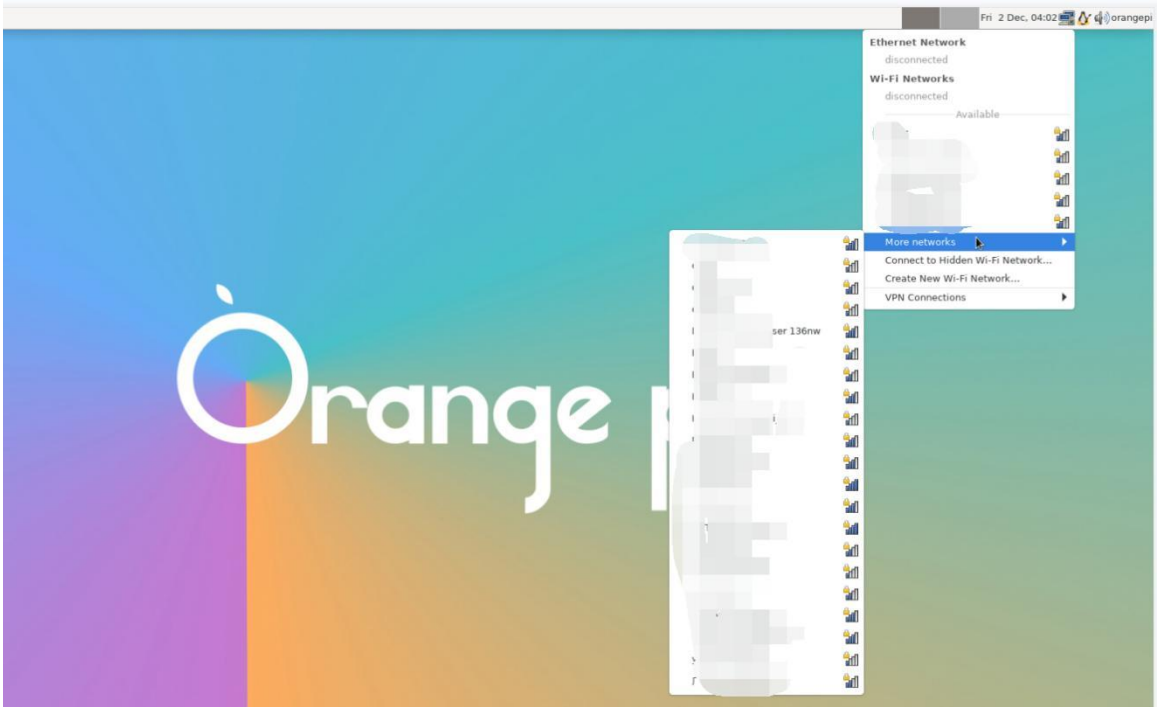
```
^C
--- www.orangepi.org ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4006ms
rtt min/avg/max/mdev = 41.321/44.864/48.834/2.484 ms
```

### 3.6.2.3. Test method of desktop image

1) Click the network configuration icon in the upper right corner of the desktop (please do not connect the network cable when testing WIFI)

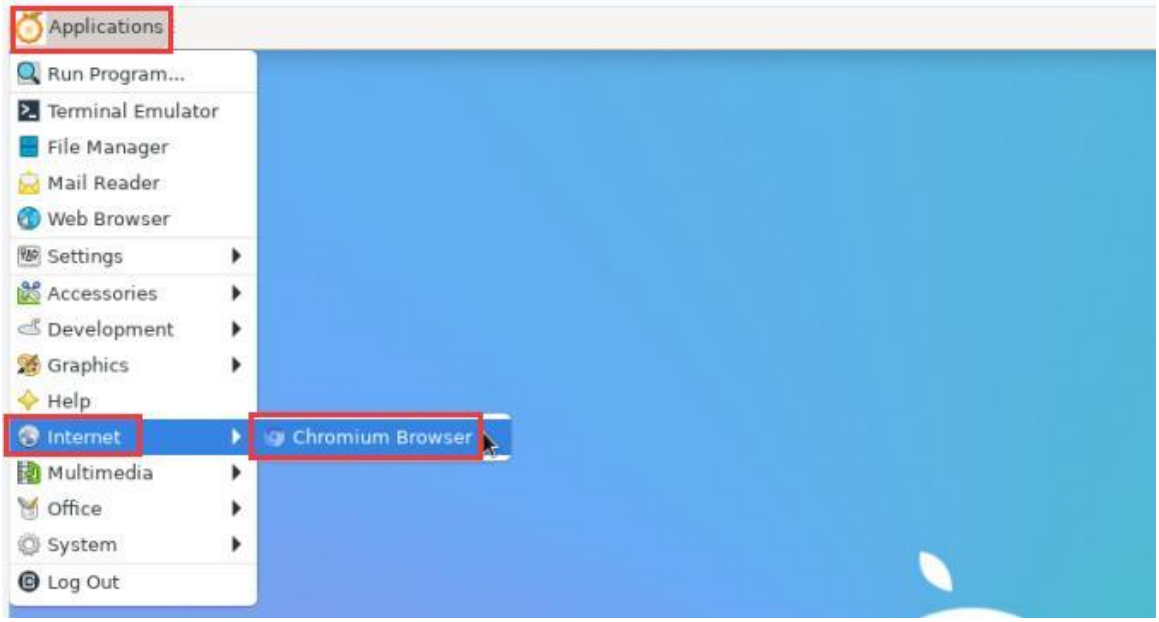



2) Click **More networks** in the pop-up drop-down box to see all scanned WIFI hotspots, and then select the WIFI hotspot you want to connect to.

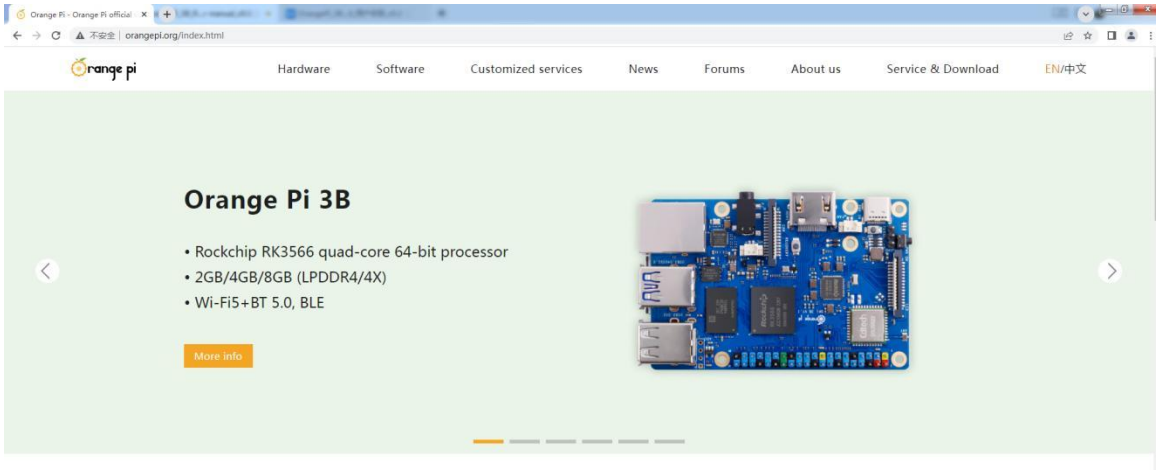


3) Then enter the password of the WIFI hotspot, and then click **Connect** to start connecting to WIFI


4) After connecting to WIFI, you can open the browser to check whether you can access the Internet. The entrance of the browser is shown in the figure below




5) If you can open other web pages after opening the browser, it means that the WIFI connection is normal




### 3. 6. 3. How to set a static IP address

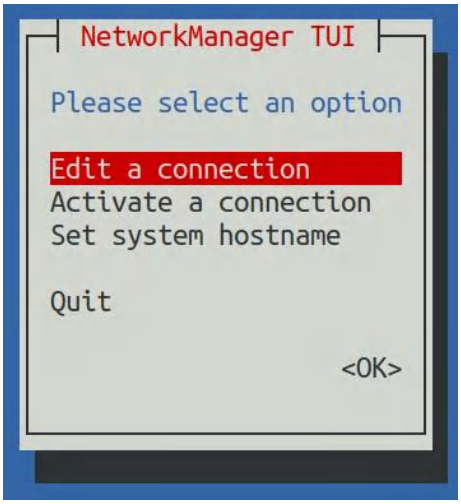
**Please do not set a static IP address by modifying the /etc/network/interfaces configuration file.**

#### 3. 6. 3. 1. Use the nmtui command to set a static IP address

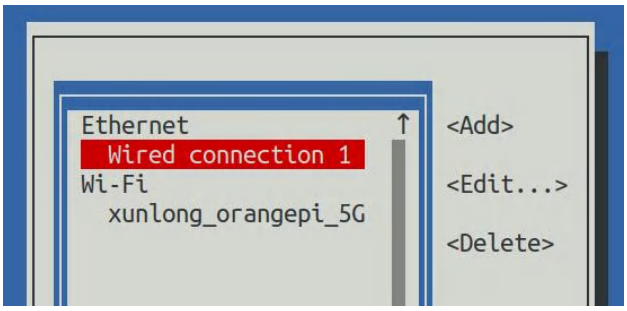
1) First run the **nmtui** command

```
orangepi@orangepi:~$ nmtui
```

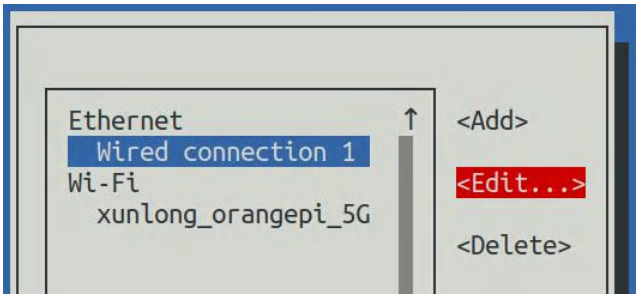
2) Then select **Edit a connection** and press Enter



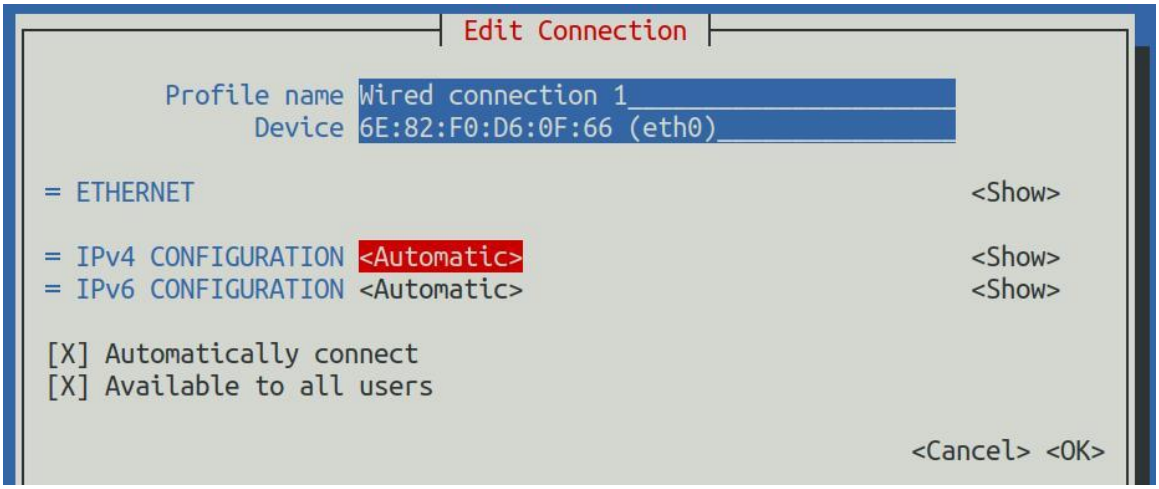

3) Then select the network interface that needs to set a static IP address, for example, to set the static IP address of the **Ethernet** interface, select **Wired connection 1**.



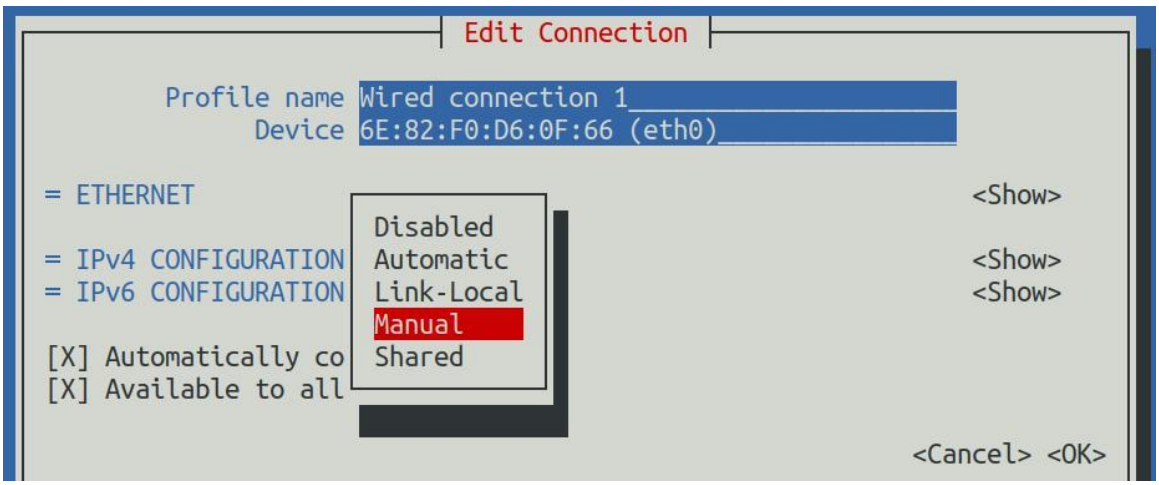

4) Then select **Edit** with the **Tab** key and press the Enter key



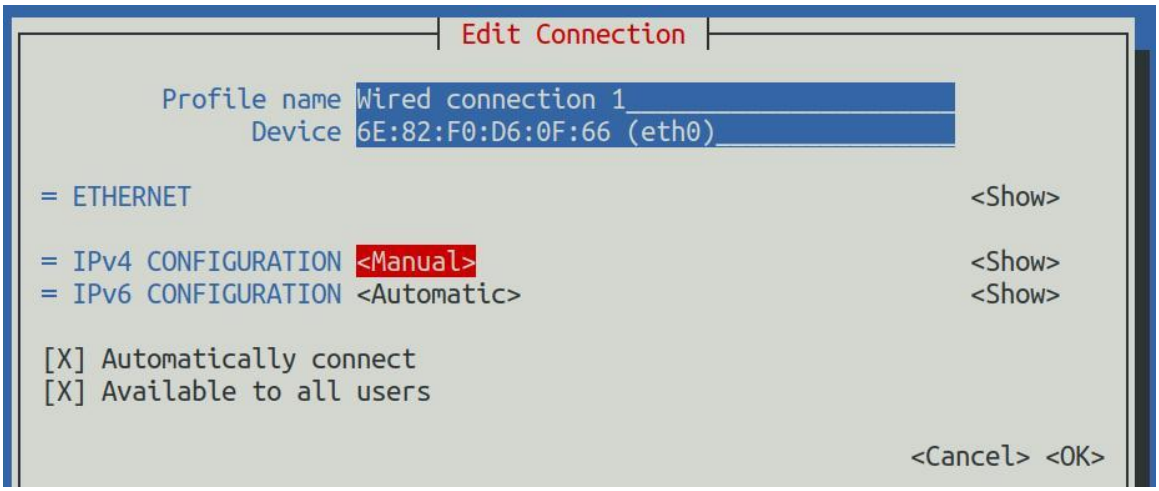

5) Then use the Tab key to move the cursor to the **<Automatic>** position shown in the figure below to configure IPv4



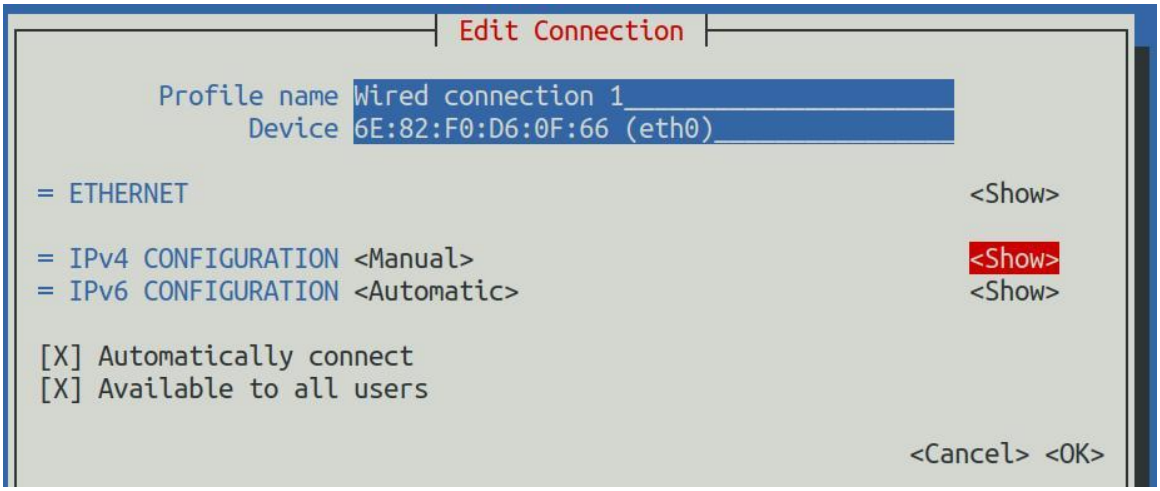

6) Then press Enter, select **Manual** with the up and down arrow keys, and press Enter to confirm



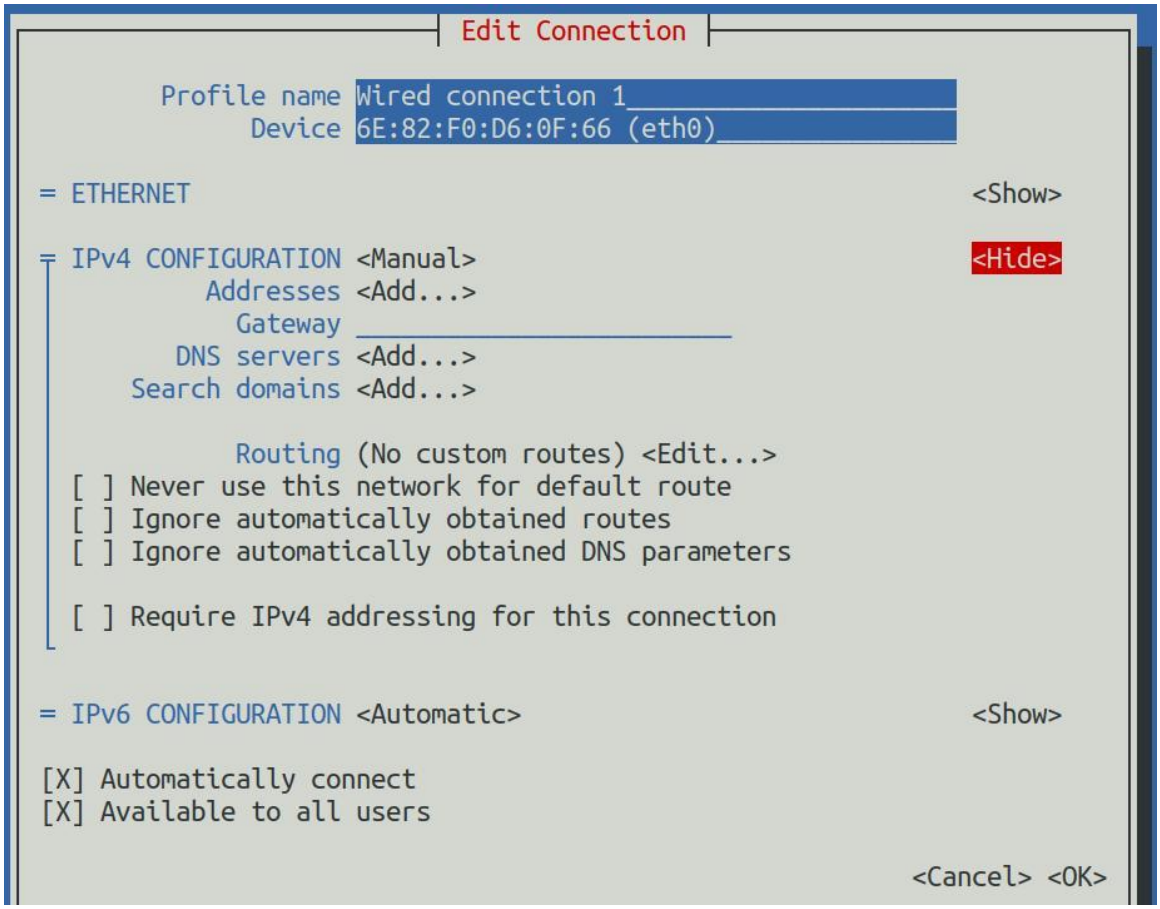

7) The display after selection is shown in the figure below



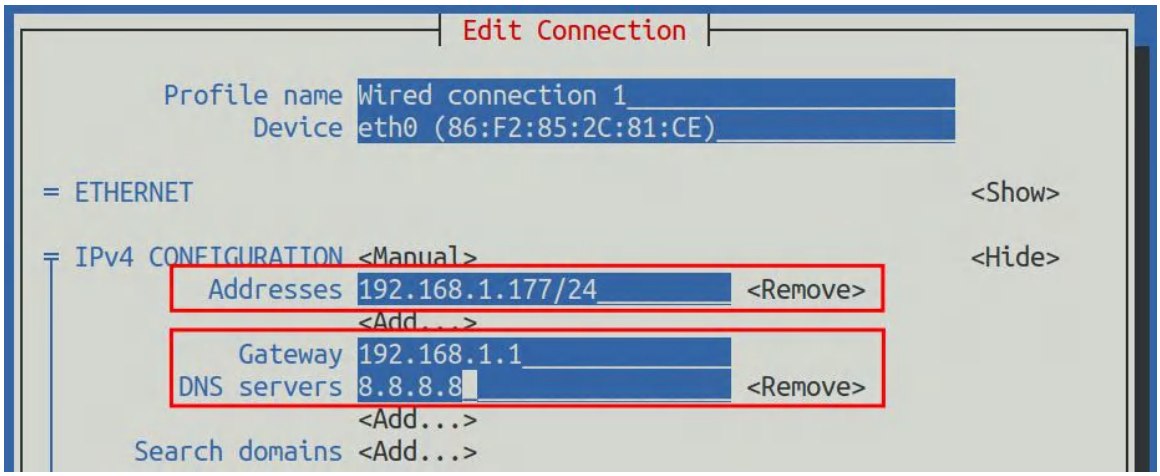

8) Then move the cursor to the **<Show>**



9) Then press Enter, the following setting interface will pop up after entering

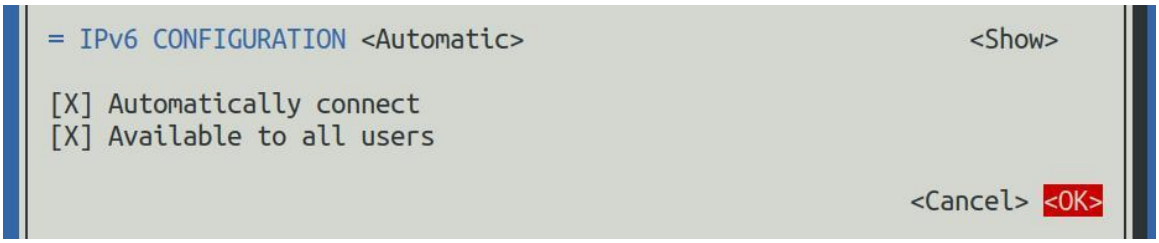



10) Then you can set the IP address (Addresses), gateway (Gateway) and DNS server address in the position shown in the figure below (there are many other setting options in it, please explore by yourself)，**Please set it according to your specific needs, the value set in the figure below is just an example**

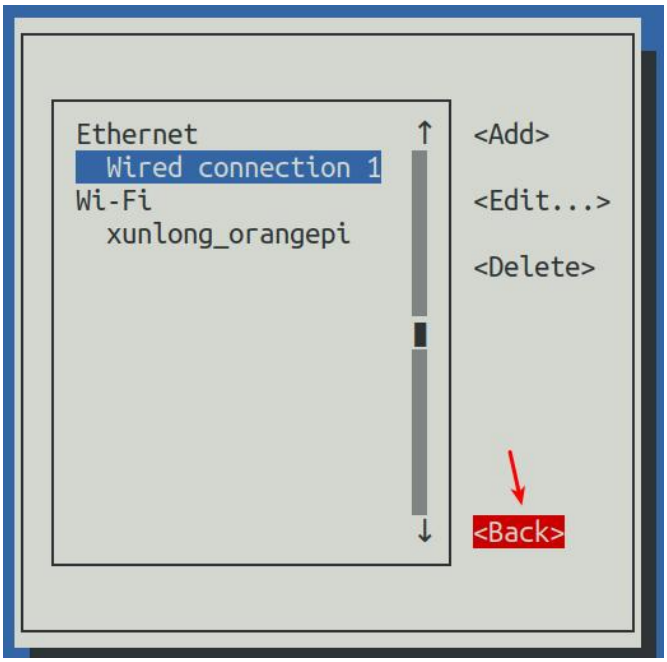



11) After setting, move the cursor to**<OK>** in the lower right corner, and press Enter to

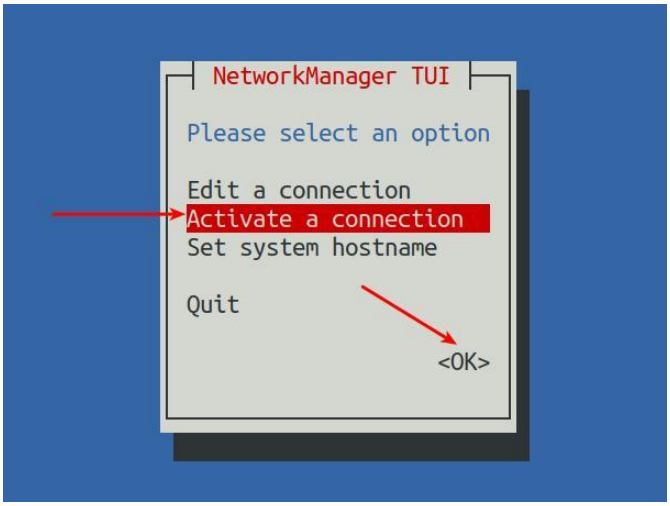
confirm



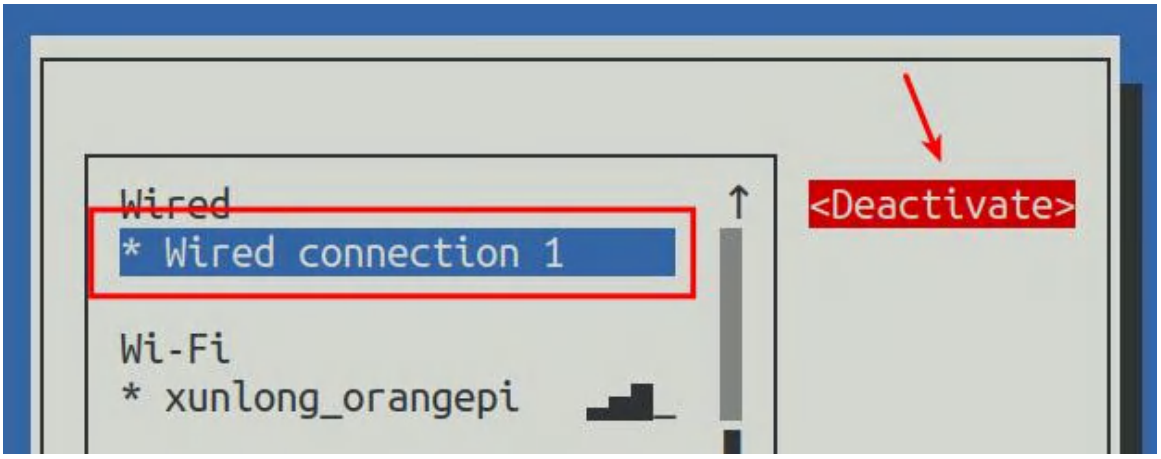

12) Then click**<Back>** to return to the previous selection interface



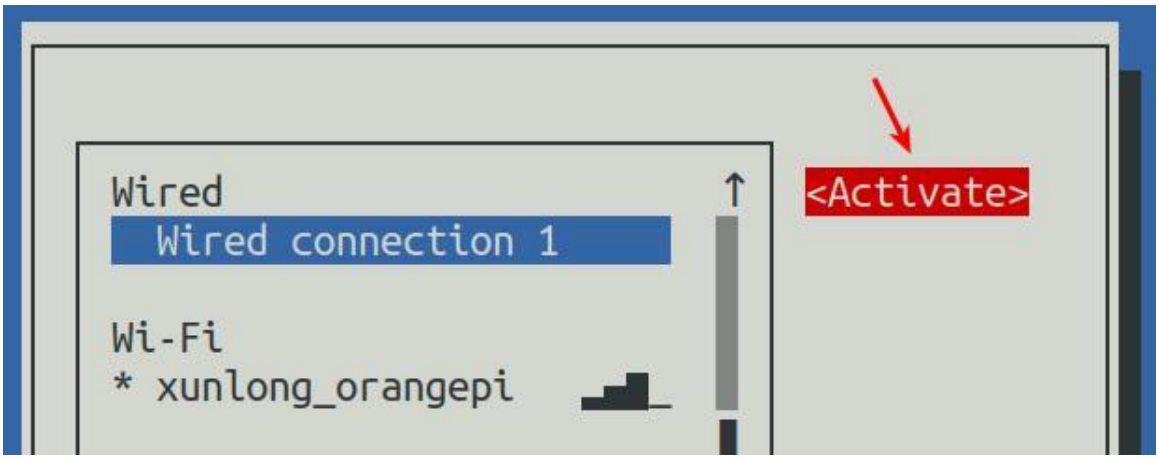

13) Then select **Activate a connection**, then move the cursor to**<OK>**, and finally click Enter



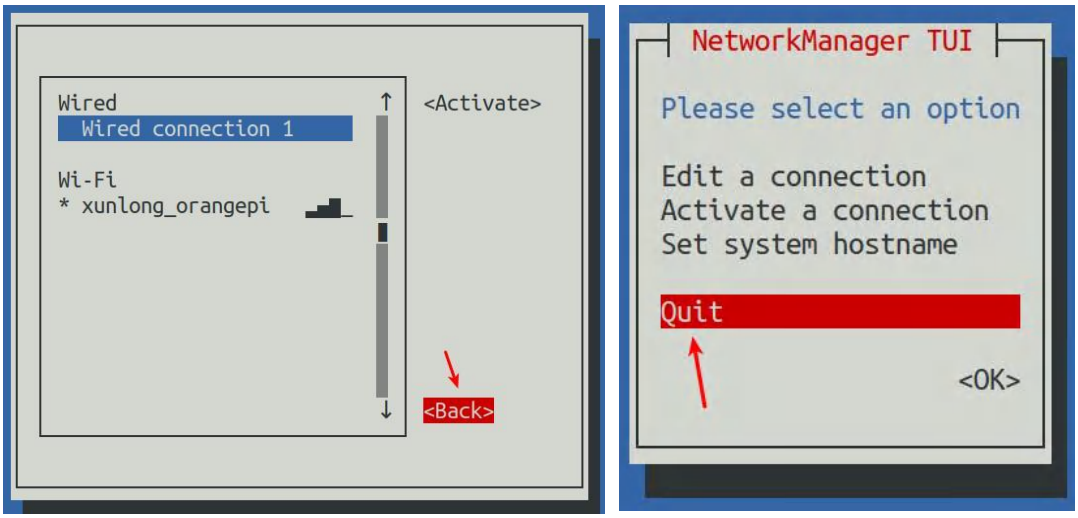
14) Then select the network interface that needs to be set, such as **Wired connection 1**, then move the cursor to**<Deactivate>**, and press Enter to disable **Wired connection 1**




15) Then please do not move the cursor, and then press the Enter key to re-enable **Wired connection 1**, so that the static IP address set earlier will take effect




16) Then you can exit nmtui through the**<Back>** and **Quit** buttons




17) Then through **ip addr show eth0**, you can see that the IP address of the network port has changed to the static IP address set earlier

```
orangepi@orangepi:~$ ip addr show eth0
3: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 5e:ac:14:a5:92:b3 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.177/24 brd 192.168.1.255 scope global noprefixroute eth0
       valid_lft forever preferred_lft forever
    inet6 241e:3b8:3240:c3a0:e269:8305:dc08:135e/64 scope global dynamic noprefixroute
       valid_lft 259149sec preferred_lft 172749sec
    inet6 fe80::957d:bbbe:4928:3604/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

18) Then you can test the connectivity of the network to check whether the IP address is configured OK, and the **ping** command can be interrupted through the shortcut key **Ctrl+C**

```
orangepi@orangepi:~$ ping 192.168.1.47 -I eth0
PING 192.168.1.47 (192.168.1.47) from 192.168.1.188 eth0: 56(84) bytes of data.
64 bytes from 192.168.1.47: icmp_seq=1 ttl=64 time=0.233 ms
64 bytes from 192.168.1.47: icmp_seq=2 ttl=64 time=0.263 ms
64 bytes from 192.168.1.47: icmp_seq=3 ttl=64 time=0.273 ms
64 bytes from 192.168.1.47: icmp_seq=4 ttl=64 time=0.269 ms
64 bytes from 192.168.1.47: icmp_seq=5 ttl=64 time=0.275 ms
```

```
^C
--- 192.168.1.47 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4042ms
rtt min/avg/max/mdev = 0.233/0.262/0.275/0.015 ms
```

### 3.6.3.2. Use the nmcli command to set a static IP address

1) If you want to set the static IP address of the network port, please insert the network cable into the development board first. **If you need to set the static IP address of WIFI, please connect the WIFI first, and then start to set the static IP address**

2) Then use the **nmcli con show** command to view the name of the network device, as shown below

a. **orangepi** is the name of the WIFI network interface (the names are not necessarily the same)

b. **Wired connection 1** is the name of the Ethernet interface

```
orangepi@orangepi:~$ nmcli con show
NAME                UUID                                  TYPE      DEVICE
orangepi            cfc4f922-ae48-46f1-84e1-2f19e9ec5e2a  wifi      wlan0
Wired connection 1  9db058b7-7701-37b8-9411-efc2ae8bfa30  ethernet  eth0
```

3) Then enter the following command, where

a. **"Wired connection 1"** means to set the static IP address of the Ethernet port. If you need to set the static IP address of the WIFI, please change it to the corresponding name of the WIFI network interface (you can get it through the **nmcli con show** command)

b. After **ipv4.addresse** is the static IP address to be set, which can be modified to the value you want to set

c. **ipv4.gateway** indicates the address of the gateway

```
orangepi@orangepi:~$ nmcli con mod "Wired connection 1"   \
ipv4.addresses "192.168.1.110"   \
ipv4.gateway "192.168.1.1"   \
ipv4.dns "8.8.8.8"   \
ipv4.method "manual"
```

4) Then restart the Linux system

```
orangepi@orangepi:~$ sudo reboot
```

5) Then re-enter the Linux system and use the **ip addr show eth0** command to see that the IP address has been set to the desired value

```
orangepi@orangepi:~$ ip addr show eth0
3: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether 5e:ae:14:a5:91:b3 brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.110/32 brd 192.168.1.110 scope global noprefixroute eth0
       valid_lft forever preferred_lft forever
    inet6 240e:3b7:3240:c3a0:97de:1d01:b290:fe3a/64 scope global dynamic noprefixroute
       valid_lft 259183sec preferred_lft 172783sec
    inet6 fe80::3312:861a:a589:d3c/64 scope link noprefixroute
       valid_lft forever preferred_lft forever
```

### 3. 6. 4. Method to create WIFI hotspot through create_ap

**create_ap is a script that helps quickly create WIFI hotspots on Linux, and supports bridge and NAT modes. It can automatically combine hostapd, dnsmasq and iptables to complete the setting of WIFI hotspots, avoiding users from making complicated configurations. The github address is as follows:**

**https://github.com/oblique/create_ap**

**If you are using the latest image, the create_ap script has been pre-installed. You can create a WIFI hotspot through the create_ap command. The basic command format of create_ap is as follows:**

**create_ap [options] <wifi-interface> [<interface-with-internet>]**
**[<access-point-name> [<passphrase>]]**

*** options：You can use this parameter to specify the encryption method, frequency band of WIFI hotspot, bandwidth mode, network sharing method, etc. You can get the options through create_ap -h.**
*** wifi-interface：The name of the wireless network card**
*** interface-with-internet：The name of the network card that can connect to the**

**Internet, usually eth0**
*** access-point-name：Hotspot name**
*** passphrase：hotspot password**

### 3.6.4.1. create_ap method to create WIFI hotspot in NAT mode

1）Enter the following command to create a WIFI hotspot with the name **orangepi** and password **orangepi** in NAT mode

**Note that in the following command, Debian12 needs to modify eth0 to end1.**

```
orangepi@orangepi:~$ sudo create_ap --no-virt -m nat wlan0 eth0 orangepi orangepi
```

2）If the following information is output, it means that the WIFI hotspot is successfully created.

```
orangepi@orangepi:~$ sudo create_ap --no-virt -m nat wlan0 eth0 orangepi orangepi
Config dir: /tmp/create_ap.wlan0.conf.Ji9Coeqo
PID: 5526
Network Manager found, set wlan0 as unmanaged device... DONE
Sharing Internet using method: nat
hostapd command-line interface: hostapd_cli -p
/tmp/create_ap.wlan0.conf.Ji9Coeqo/hostapd_ctrl
wlan0: interface state UNINITIALIZED->ENABLED
wlan0: AP-ENABLED
```

3）At this time, take out your mobile phone and find the WIFI hotspot named **orangepi** created by the development board in the searched WIFI list. Then you can click **orangepi** to connect to the hotspot. The password is **orangepi** set above

4）The display after successful connection is as shown below


5）In NAT mode, the wireless device connected to the development board's hotspot requests an IP address from the development board's DHCP service, so there will be two different network segments. For example, the development board's IP here is 192.168.1.X

```
orangepi@orangepi:~$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.150  netmask 255.255.255.0  broadcast 192.168.1.255
        inet6 fe80::938f:8776:5783:afa2  prefixlen 64  scopeid 0x20<link>
        ether 4a:a0:c8:25:42:82  txqueuelen 1000  (Ethernet)
        RX packets 25370  bytes 2709590 (2.7 MB)
        RX errors 0  dropped 50  overruns 0  frame 0
        TX packets 3798  bytes 1519493 (1.5 MB)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
        device interrupt 83
```

The DHCP service of the development board will assign the IP address of **192.168.12.0/24** to the device connected to the hotspot by default. At this time, click on the connected WIFI hotspot **orangepi**, and then you can see that the IP address of the mobile phone is **192.168.12.X**


6）If you want to specify a different network segment for the connected device, you can specify it through the -g parameter. For example, use the -g parameter to specify the network segment of the access point AP as 192.168.2.1.

**Note that in the following command, Debian12 needs to modify eth0 to end1.**

```
orangepi@orangepi:~$ sudo create_ap --no-virt -m nat wlan0 eth0 orangepi orangepi -g 192.168.2.1
```

At this time, after connecting to the hotspot through the mobile phone, click on the connected WIFI hotspot **orangepi**, and then you can see that the IP address of the mobile phone is **192.168.2.X**


7）Without specifying the **--freq-band** parameter, the hotspot created by default is in the 2.4G frequency band. If you want to create a hotspot in the 5G frequency band, you can specify it through the **--freq-band 5** parameter. The specific command is as follows

**Note that in the following command, Debian12 needs to modify eth0 to end1.**

```
orangepi@orangepi:~$ sudo create_ap --no-virt -m nat wlan0 eth0 orangepi orangepi --freq-band 5
```

8）If you need to hide the SSID, you can specify the **--hidden** parameter. The specific command is as follows

**Note that in the following command, Debian12 needs to modify eth0 to end1.**

```
orangepi@orangepi:~$ sudo create_ap --no-virt -m nat wlan0 eth0 orangepi orangepi --hidden
```

At this time, the mobile phone cannot search for WIFI hotspots. You need to manually specify the WIFI hotspot name and enter the password to connect to the WIFI hotspot.


### 3.6.4.2. create_ap method to create WIFI hotspot in bridge mode

1）Enter the following command to create a WIFI hotspot with the name **orangepi** and password **orangepi** in bridge mode

**Note that in the following command, Debian12 needs to modify eth0 to end1.**

```
orangepi@orangepi:~$ sudo create_ap --no-virt -m bridge wlan0 eth0 orangepi orangepi
```

2）If the following information is output, it means that the WIFI hotspot is successfully created.

```
orangepi@orangepi:~$ sudo create_ap --no-virt -m bridge wlan0 eth0 orangepi orangepi
[sudo] password for orangepi:
Config dir: /tmp/create_ap.wlan0.conf.hXrfLdof
```

```
PID: 8372
Network Manager found, set wlan0 as unmanaged device... DONE
Sharing Internet using method: bridge
Create a bridge interface... br0 created.
hostapd command-line interface: hostapd_cli -p
/tmp/create_ap.wlan0.conf.hXrfLdof/hostapd_ctrl
wlan0: interface state UNINITIALIZED->ENABLED
```

3）At this time, take out your mobile phone and find the WIFI hotspot named **orangepi** created by the development board in the searched WIFI list. Then you can click **orangepi** to connect to the hotspot. The password is **orangepi** set above.


4）The display after successful connection is as shown below


5）In bridge mode, the wireless device connected to the hotspot of the development board also requests an IP address from the DHCP service of the main router (the router to which the development board is connected). For example, the IP of the development board here is **192.168.1.X**

```
orangepi@orangepi:~$ ifconfig eth0
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 192.168.1.150  netmask 255.255.255.0  broadcast 192.168.1.255
```

```
inet6 fe80::938f:8776:5783:afa2  prefixlen 64  scopeid 0x20<link>
ether 4a:a0:c8:25:42:82  txqueuelen 1000  (Ethernet)
RX packets 25370  bytes 2709590 (2.7 MB)
RX errors 0  dropped 50  overruns 0  frame 0
TX packets 3798  bytes 1519493 (1.5 MB)
TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
device interrupt 83
```

The IP of the device connected to the WIFI hotspot is also assigned by the main router, so the mobile phone connected to the WIFI hotspot and the development board are in the same network segment. At this time, click on the connected WIFI hotspot **orangepi**, and then you can see the IP address of the mobile phone. Also **192.168.1.X**.


6）Without specifying the **--freq-band** parameter, the hotspot created by default is in the 2.4G frequency band. If you want to create a hotspot in the 5G frequency band, you can specify it through the **--freq-band 5** parameter. The specific command is as follows

**Note that in the following command, Debian12 needs to modify eth0 to end1.**

```
orangepi@orangepi:~$ sudo create_ap --no-virt -m bridge wlan0 eth0 orangepi orangepi --freq-band 5
```

7）If you need to hide the SSID, you can specify the **--hidden** parameter. The specific command is as follows

**Note that in the following command, Debian12 needs to modify eth0 to end1.**

```
orangepi@orangepi:~$ sudo create_ap --no-virt -m bridge wlan0 eth0 orangepi orangepi --hidden
```

At this time, the mobile phone cannot search for WIFI hotspots. You need to manually specify the WIFI hotspot name and enter the password to connect to the WIFI hotspot.


## 3. 7. SSH remote login development board;

**Linux systems enable ssh remote login by default and allow the root user to log in to the system. Before logging in with ssh, you first need to ensure that the Ethernet or wifi network is connected, and then use the ip addr command or check the router to obtain the IP address of the development board.**

### 3. 7. 1. SSH remote login development board under Ubuntu

1) Obtain the IP address of the development board

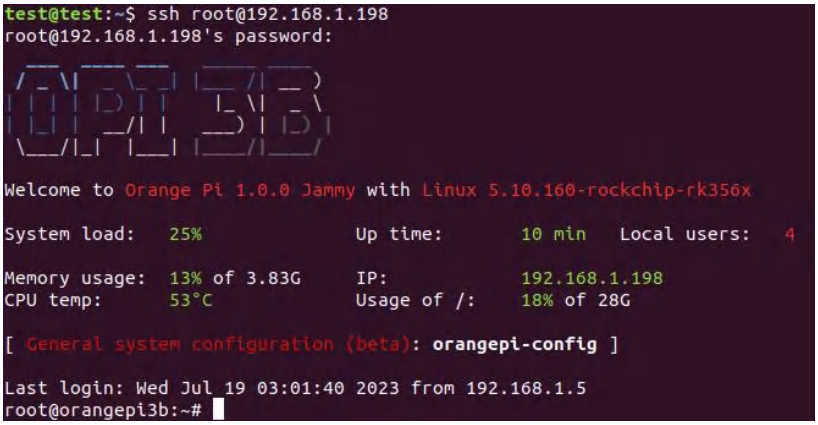
2) Then you can remotely log in to the Linux system through the ssh command

```
test@test:~$ ssh root@192.168.1.xxx          (Need to be replaced with the IP address of the development board)
root@192.168.1.xx's password:          (Enter the password here, the default password is orangepi)
```

**Note that when entering the password, the specific content of the entered password will not be displayed on the screen, please do not think that there is any fault, just press Enter after inputting.**

**If you are prompted to refuse the connection, as long as you are using the image provided by Orange Pi, please do not suspect that the password orangepi is wrong, but look for other reasons.**

3) After successfully logging in to the system, the display is as shown in the figure below




**If ssh cannot log in to the Linux system normally, please first check whether the IP address of the development board can be pinged. If the ping is ok, you can log in to the Linux system through the serial port or HDMI display and then enter the following command on the development board and try again. Is it possible to connect:**

root@orangepi:~# **reset_ssh.sh**

**If it still doesn't work, try to reset the system.**

### 3. 7. 2. SSH remote login development board under Windows

1) First obtain the IP address of the development board

2) Under Windows, you can use MobaXterm to remotely log in to the development board, first create a new ssh session

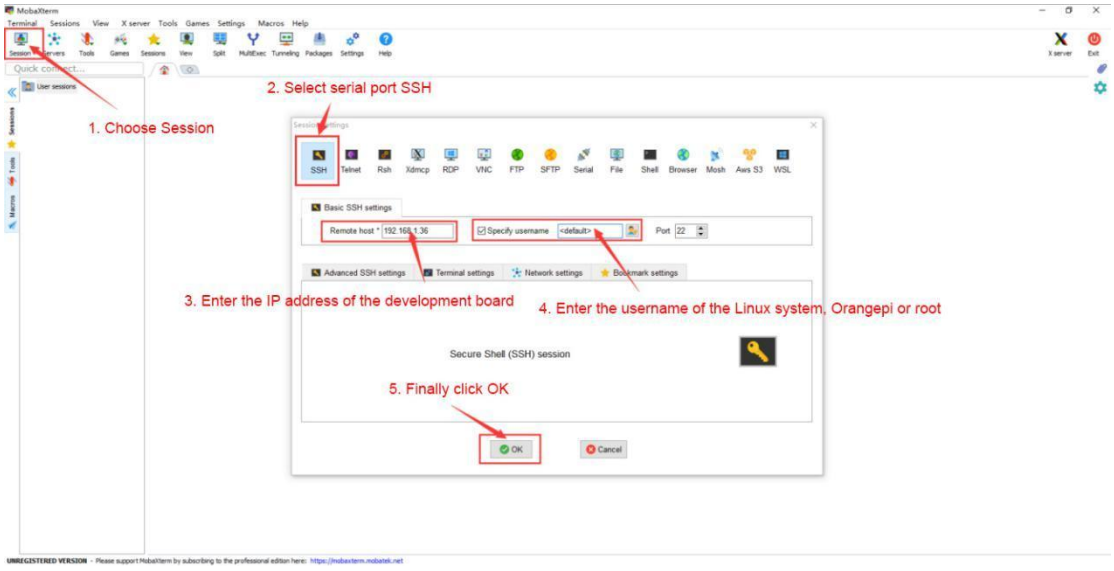
a. Open **Session**

b. Then select **SSH** in **Session Setting**

c. Then enter the IP address of the development board in the **Remote host**

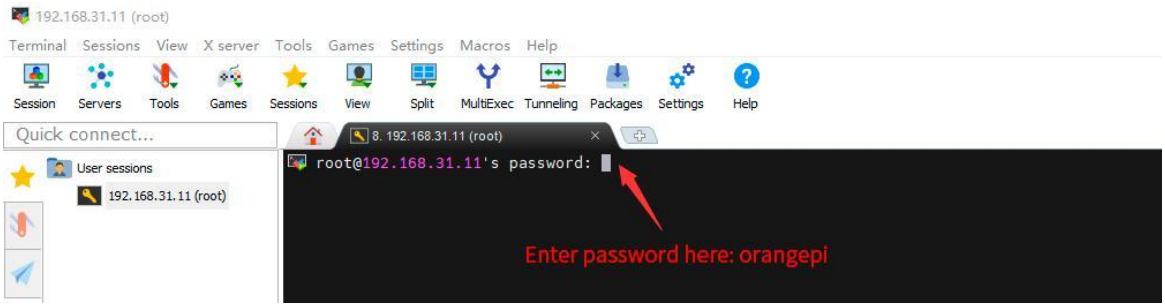
d. Then enter the user name **root** or **orangepi** of the Linux system in **Specify username**

e. Finally click **OK**

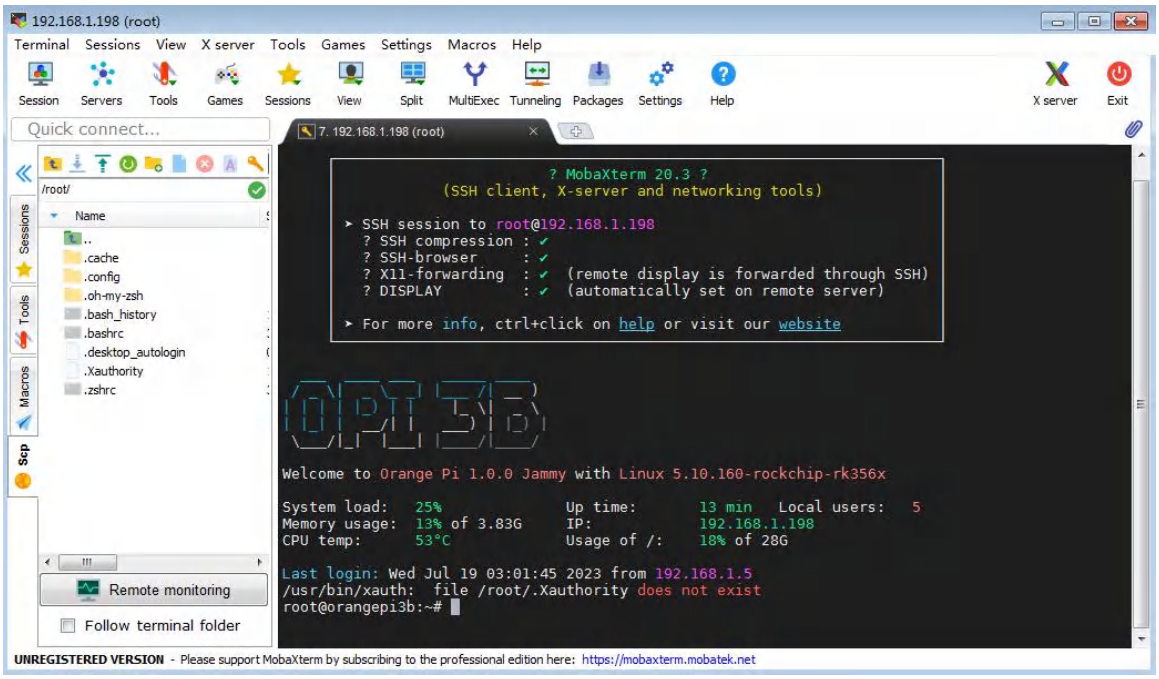



3) Then you will be prompted to enter a password. The default passwords for root and orangepi users are orangepi

**Note that when entering the password, the specific content of the entered password will not be displayed on the screen, please do not think that there is any fault, just press Enter after inputting.**




4) After successfully logging in to the system, the display is as shown in the figure below




# 3. 8. The method of uploading files to the Linux system of the development board

## 3. 8. 1. How to upload files to the development board Linux system in Ubuntu PC

### 3. 8. 1. 1. How to upload files using the scp command

1) Use the scp command to upload files from the Ubuntu PC to the Linux system of the development board. The specific commands are as follows

a. **file_path**: need to be replaced with the path of the file to be uploaded

b. **orangepi**: It is the user name of the Linux system of the development board, and it can also be replaced with other ones, such as root

c. **192.168.xx.xx**: It is the IP address of the development board, please modify it according to the actual situation

d. **/home/orangepi**: The path in the Linux system of the development board, which can also be modified to other paths

```
test@test:~$ scp    file_path    orangepi@192.168.xx.xx:/home/orangepi/
```

2) If you want to upload a folder, you need to add the -r parameter

```
test@test:~$ scp -r dir_path orangepi@192.168.xx.xx:/home/orangepi/
```

3) There are more usages of scp, please use the following command to view the man manual

```
test@test:~$ man scp
```

### 3.8.1.2. How to upload files using filezilla

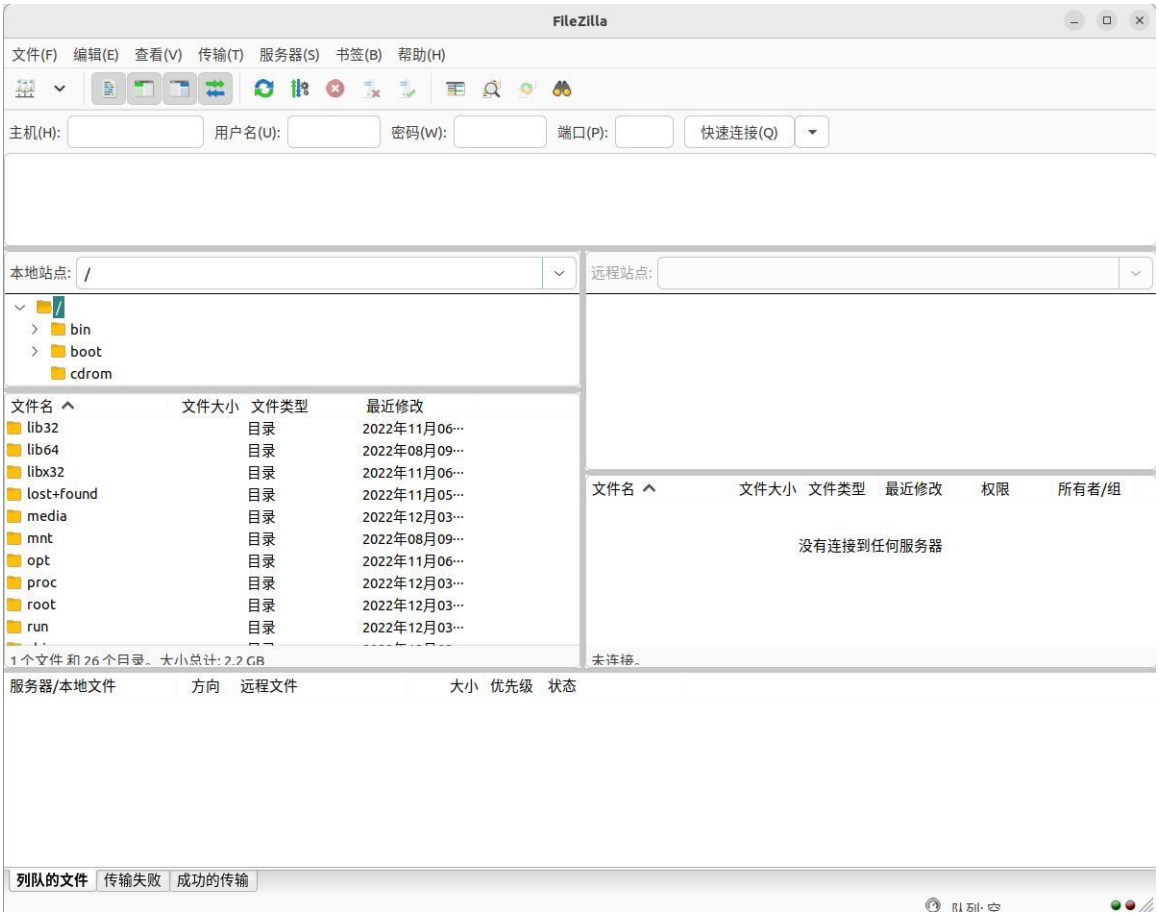
1) First install filezilla in Ubuntu PC

```
test@test:~$ sudo apt install -y filezilla
```

2) Then use the following command to open filezilla

```
test@test:~$ filezilla
```

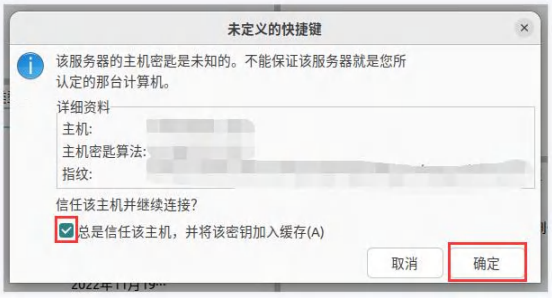
3) The interface after filezilla is opened is as follows, and the display under the remote site on the right is empty



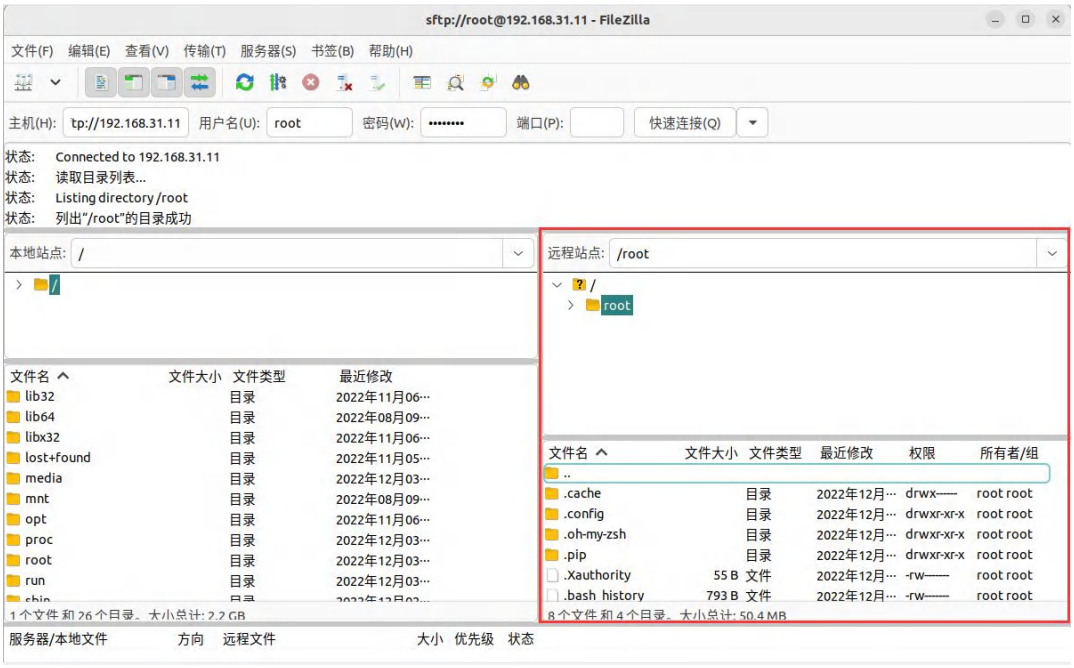
4) The method of connecting the development board is shown in the figure below


5) Then choose to **save the password**, and then click **OK**

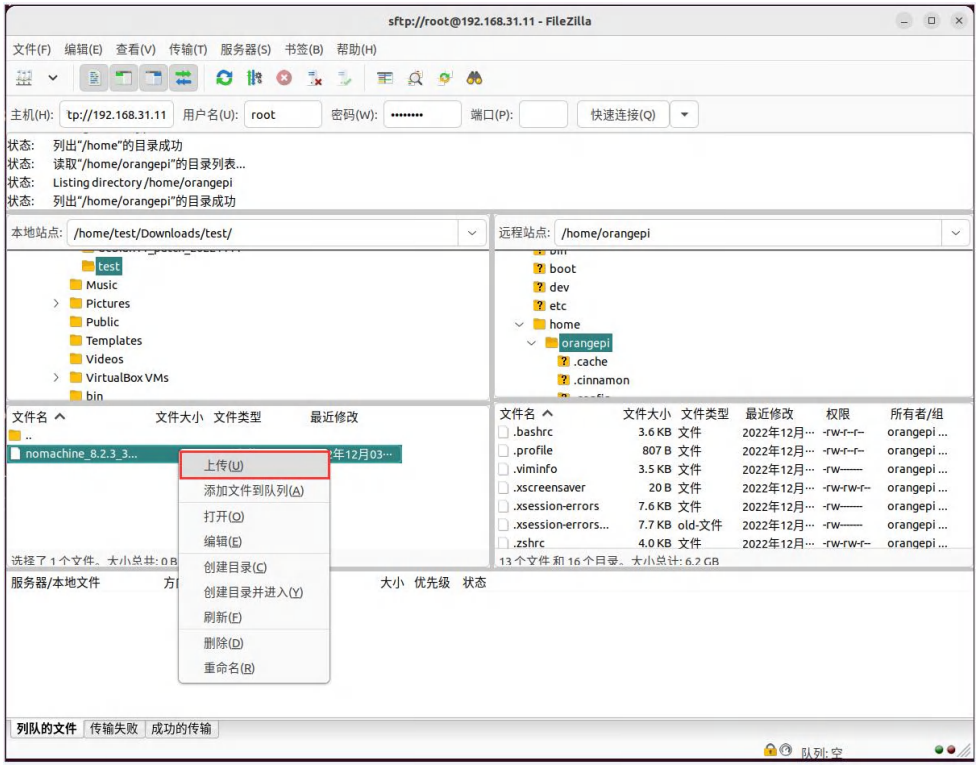

6) Then choose to always **trust this host**, and then click **OK**




7) After the connection is successful, you can see the directory structure of the development board Linux file system on the right side of the filezilla software




8) Then select the path to be uploaded to the development board on the right side of the filezilla software, and then select the file to be uploaded on the Ubuntu PC on the left side of the filezilla software, then click the right mouse button, and then click the upload option to start uploading the file to the development board.




9) After the upload is complete, you can go to the corresponding path in the Linux system

of the development board to view the uploaded file

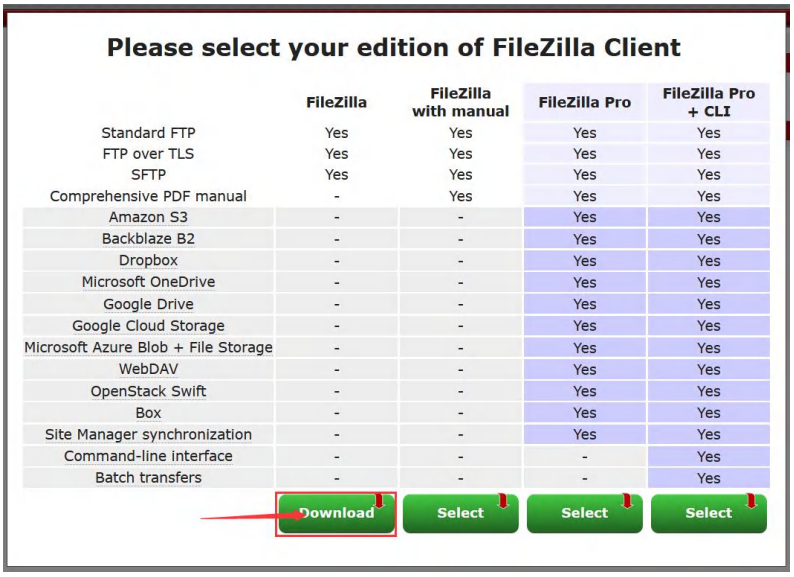
10) The method of uploading a folder is the same as that of uploading a file, so I won't go into details here

### 3. 8. 2. The method of uploading files from Windows PC to the Linux system of the development board

#### 3. 8. 2. 1. How to upload files using filezilla

1) First download the installation file of the Windows version of the filezilla software, the download link is as follows

**https://filezilla-project.org/download.php?type=client**

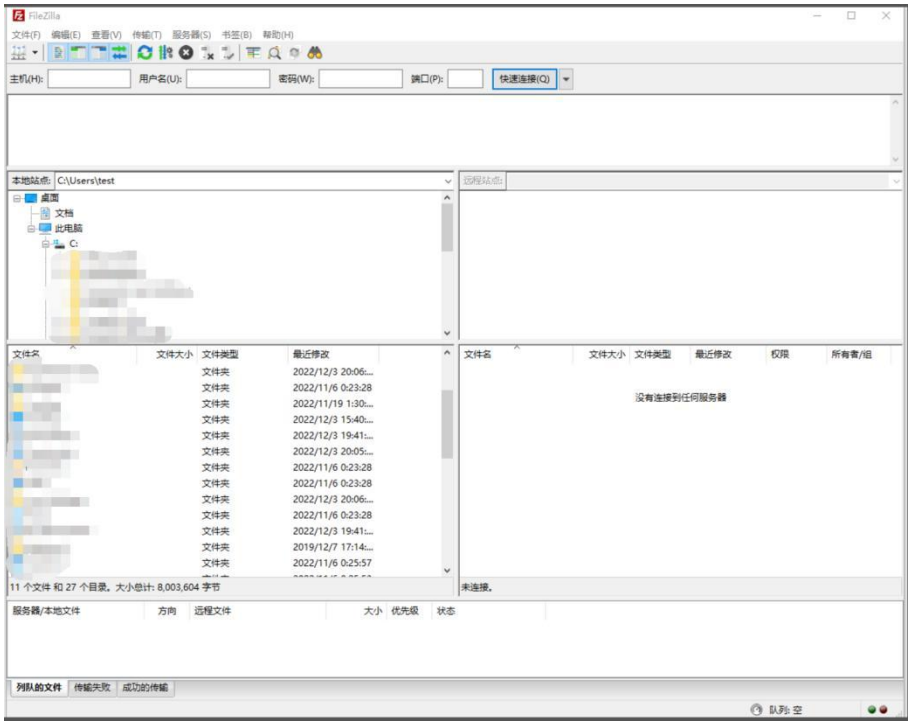




2) The downloaded installation package is as follows, and then double-click to install directly

**FileZilla_Server_1.5.1_win64-setup.exe**

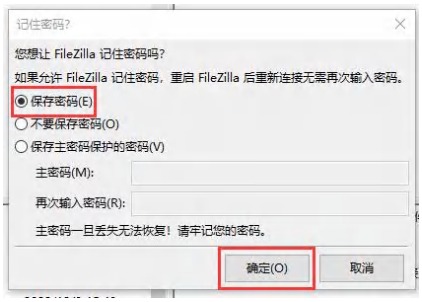
During the installation process, please select **Decline** on the following installation interface, and then select **Next>**


3) The interface after filezilla is opened is as follows, and the display under the remote site on the right is empty

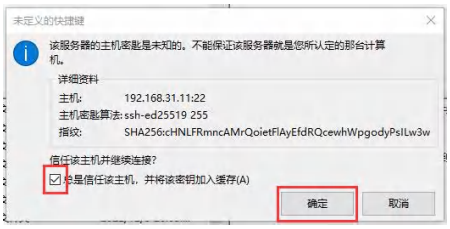


4) The method of connecting the development board is shown in the figure below:

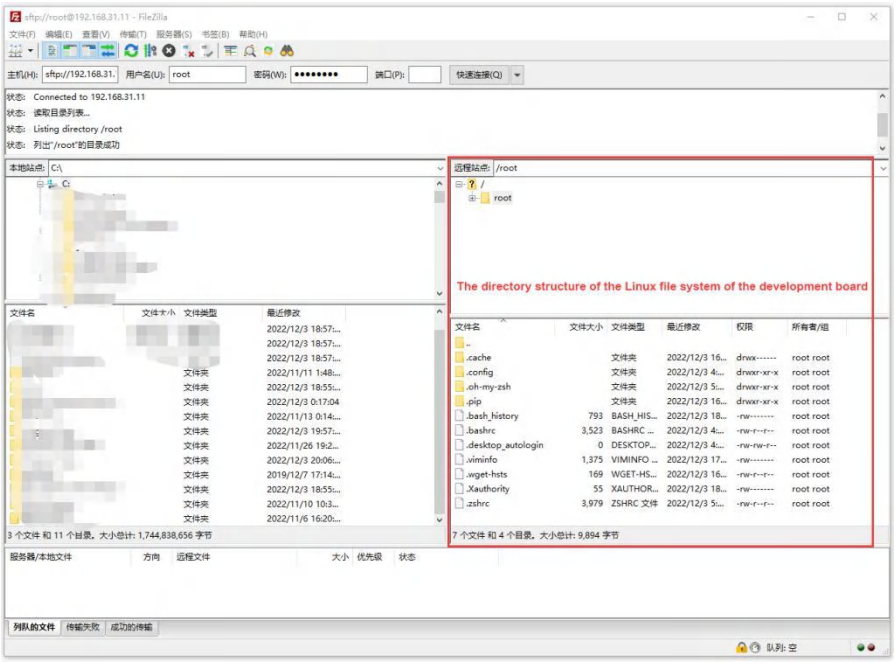

5) Then choose to **save the password**, and then click **OK**



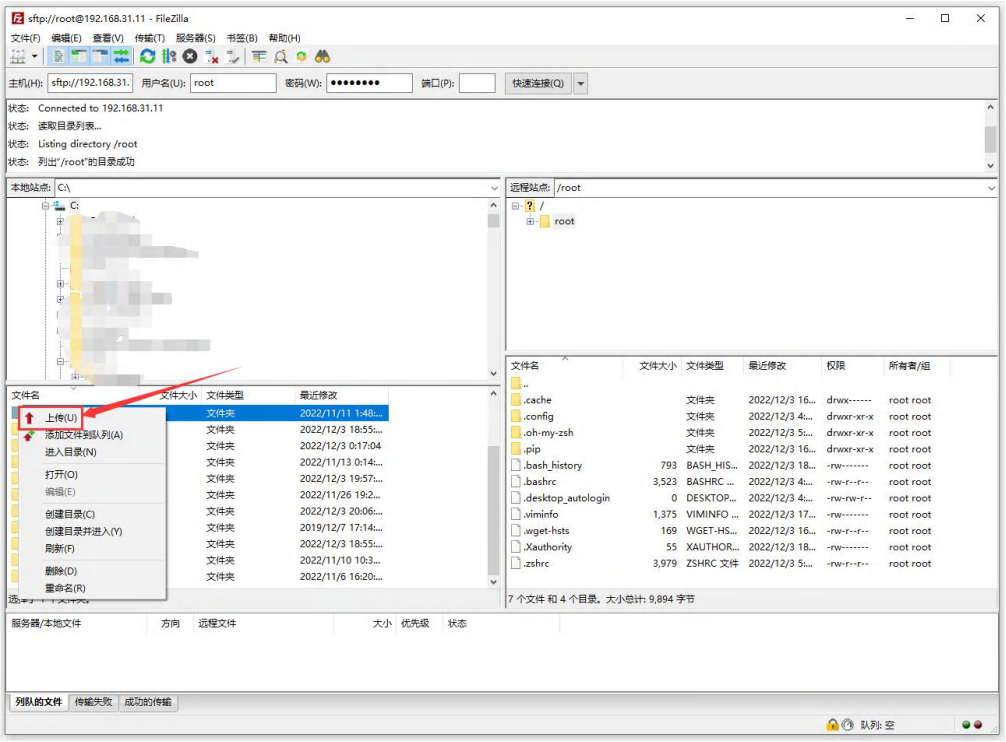

6) Then choose to always **trust this host**, and then click **OK**




7) After the connection is successful, you can see the directory structure of the development board Linux file system on the right side of the filezilla software




8) Then select the path to be uploaded to the development board on the right side of the filezilla software, and then select the file to be uploaded on the Ubuntu PC on the left side of the filezilla software, then click the right mouse button, and then click the upload option to start uploading the file to the development board.



9) After the upload is complete, you can go to the corresponding path in the Linux system of the development board to view the uploaded file

10) The method of uploading a folder is the same as that of uploading a file

## 3.9. HDMI test

### 3.9.1. HDMI display test

1) Use HDMI to HDMI cable to connect Orange Pi development board and HDMI monitor

2) After starting the Linux system, if the HDMI monitor has image output, it means that the HDMI interface is in normal use

**Note that although many notebook computers have an HDMI interface, the HDMI interface of the notebook generally only has the output function, and does not have the function of HDMI in, that is to say, the HDMI output of other devices cannot be displayed on the notebook screen.**

**When you want to connect the HDMI of the development board to the HDMI port of the laptop, please make sure that your laptop supports the HDMI in function.**

**When the HDMI is not displayed, please check whether the HDMI cable is plugged in tightly. After confirming that there is no problem with the connection, you can change a different screen and try to see if it is displayed.**

### 3.9.2. HDMI to VGA display test

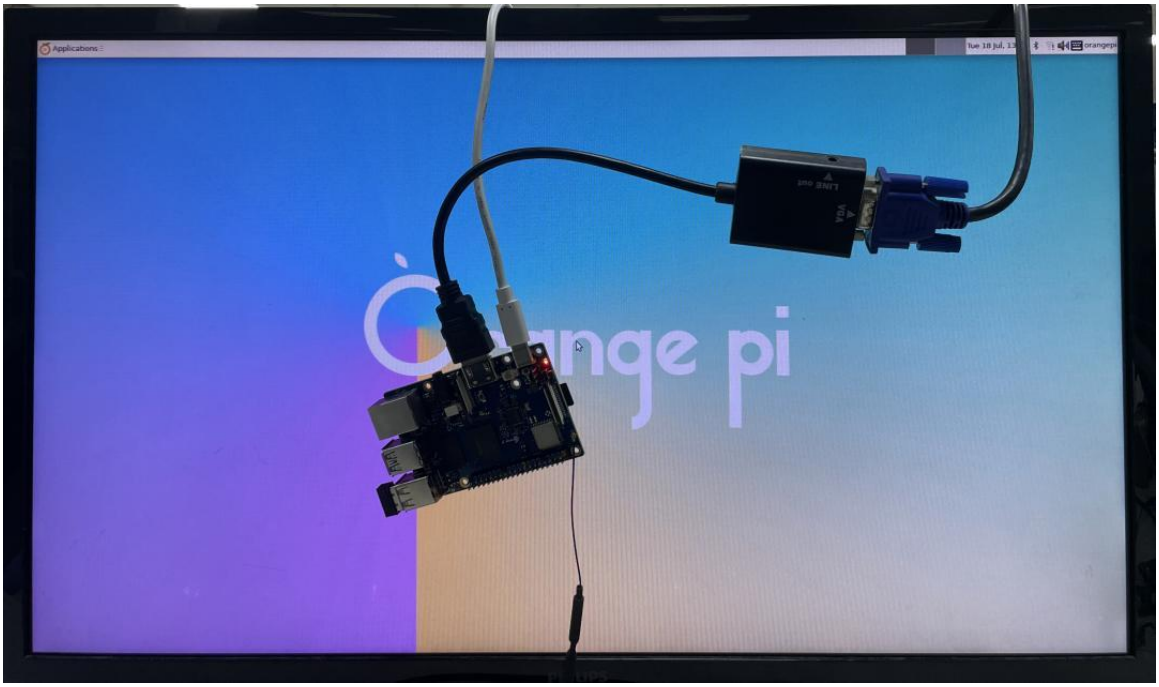
1) First, you need to prepare the following accessories

a. HDMI to VGA converter

b. A VGA cable

c. A monitor or TV that supports VGA interface

2) HDMI to VGA display test as shown below

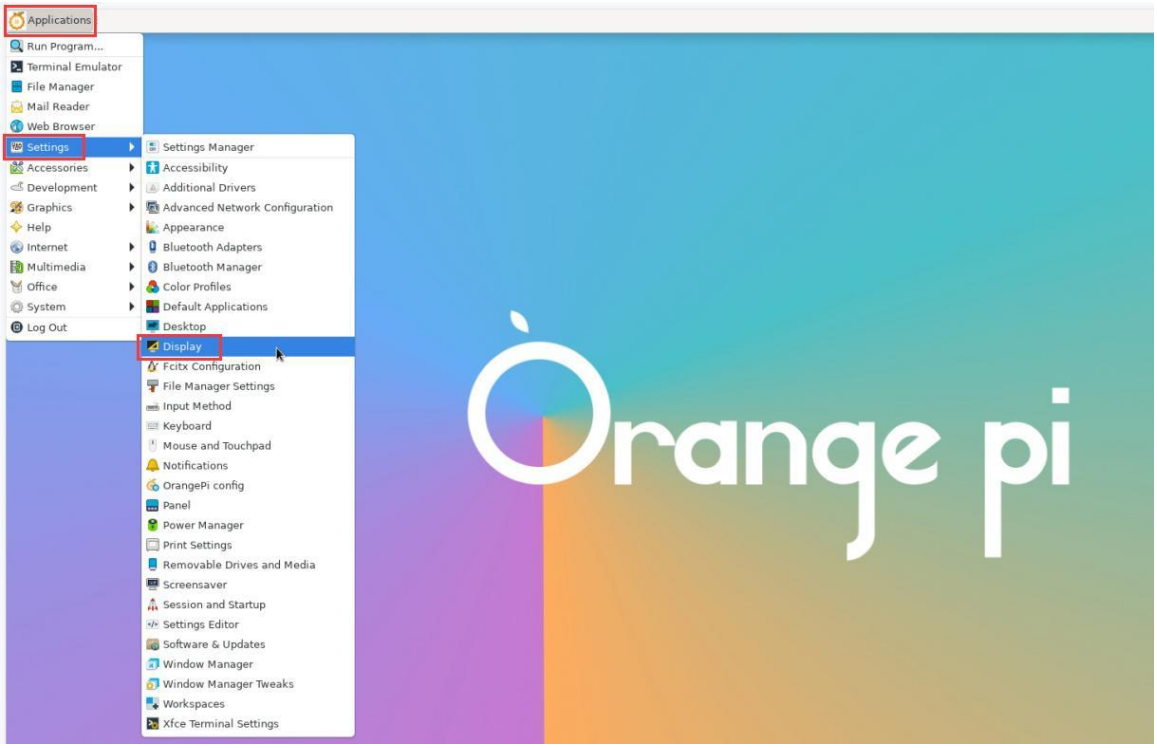


**When using HDMI to VGA display, the development board and the Linux system of the development board do not need to make any settings, only the HDMI interface of the development board can display normally. So if there is a problem with the test, please check whether there is a problem with the HDMI to VGA**

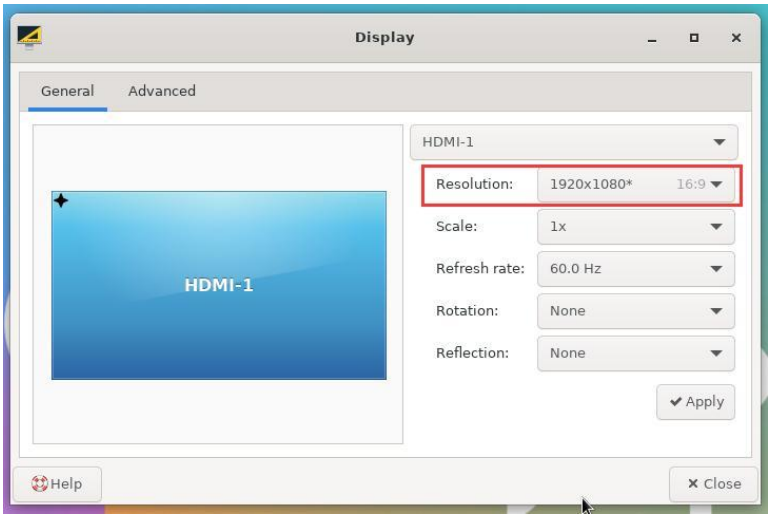
**converter, VGA cable and monitor.**

### 3. 9. 3. HDMI resolution setting method

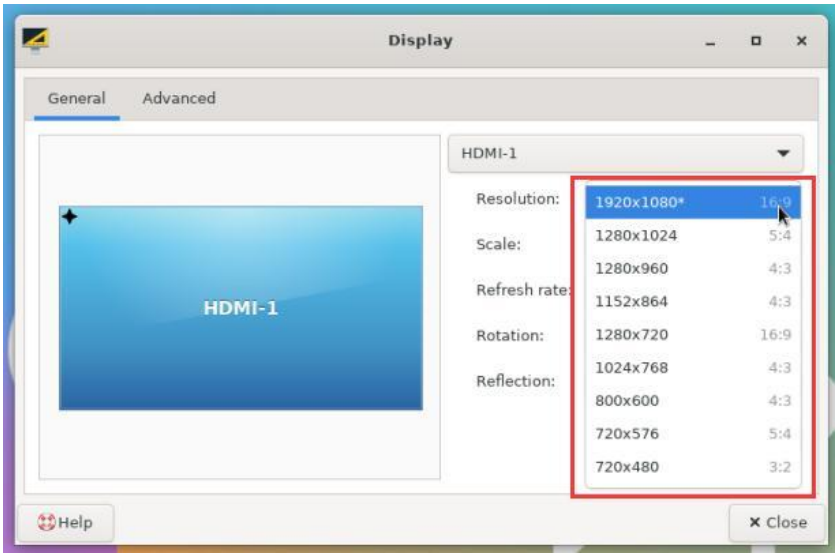
1) First open **Display** in **Settings**



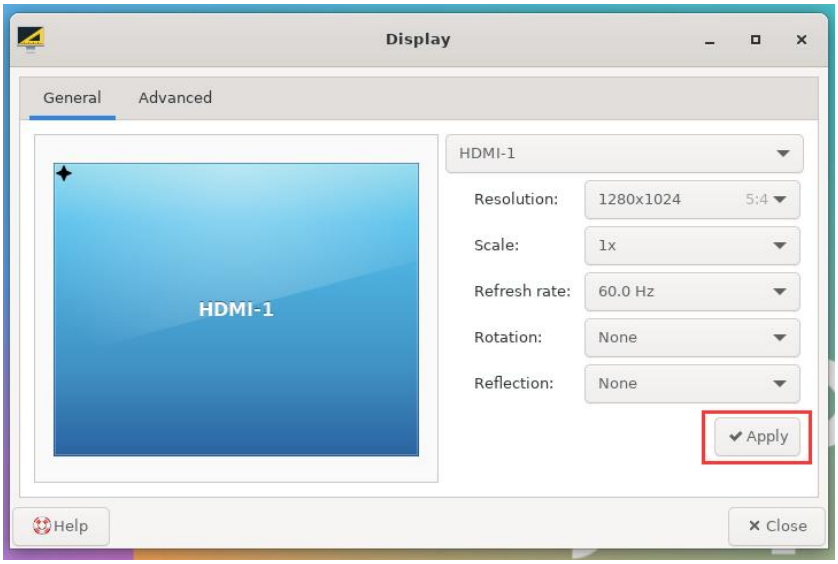

2) Then you can see the current resolution of the system



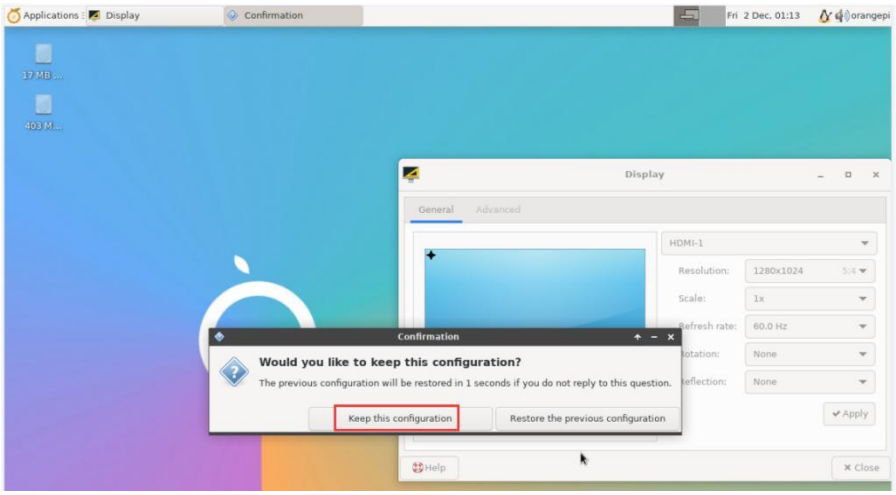

3) Click the drop-down box of Resolution to see all resolutions currently supported by the monitor




4) Then select the resolution you want to set, and click Apply



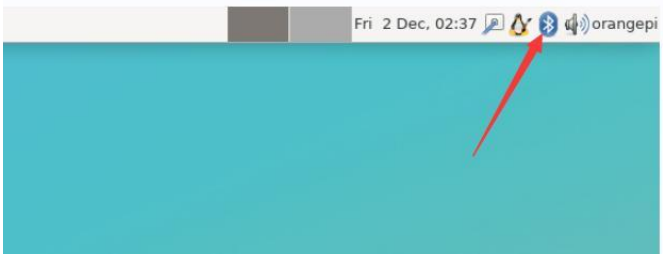

5) After the new resolution is set, select **Keep the configuration**



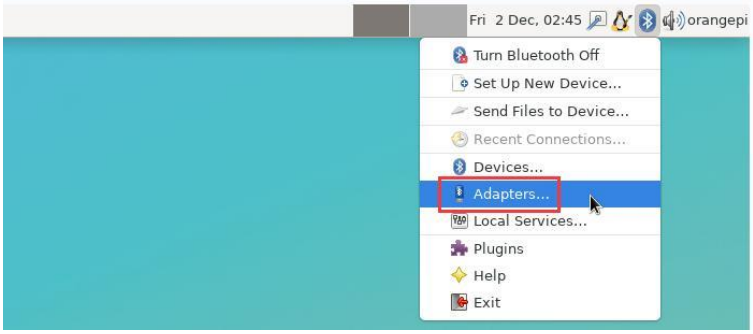

## 3. 10. How to use Bluetooth

### 3. 10. 1. Test method of desktop image

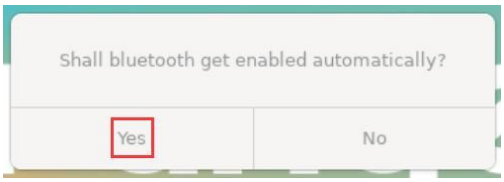
1) Click the Bluetooth icon in the upper right corner of the desktop.



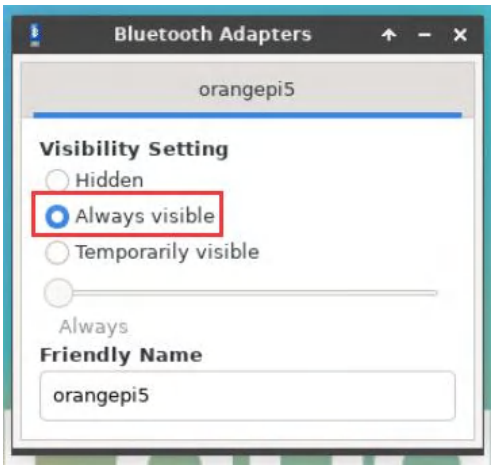

2) Then select the adapter



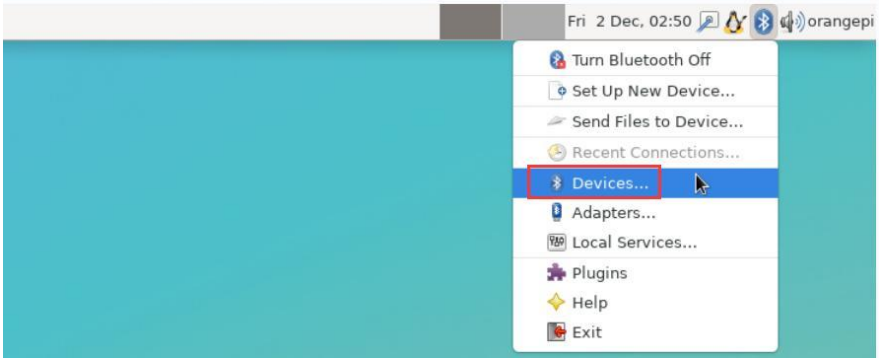

3) If there is a prompt on the following interface, please select **Yes**



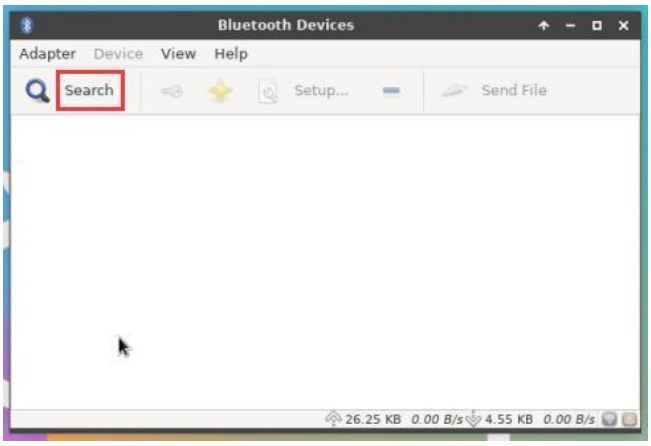

4) Then set **Visibility Setting** as **Always visible** in the Bluetooth adapter settings interface, and then close it



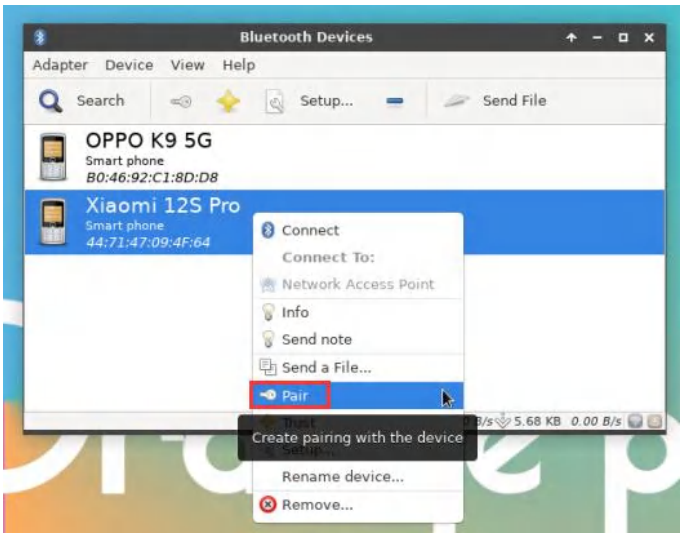

5) Then open the configuration interface of the Bluetooth device



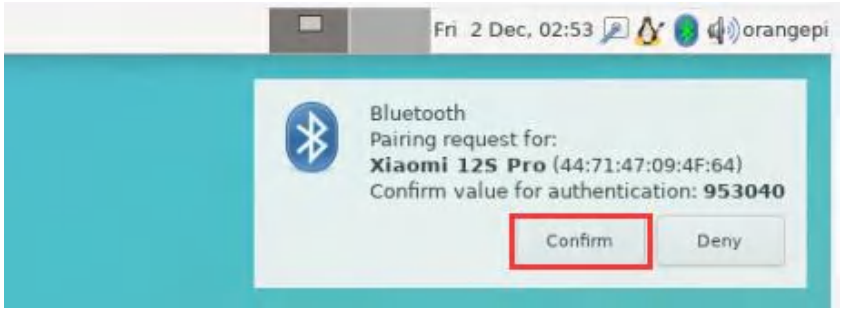

6) Click **Search** to start scanning the surrounding Bluetooth device



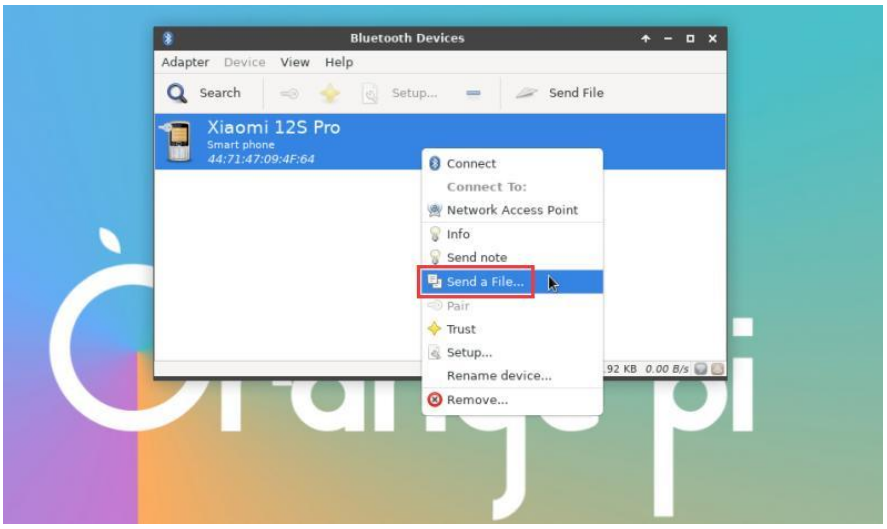

7) Then select the Bluetooth device you want to connect to, and then click the right button of the mouse to pop up the operation interface for this Bluetooth device, select **Pair** to start pairing, and the demonstration here is to pair with an Android phone



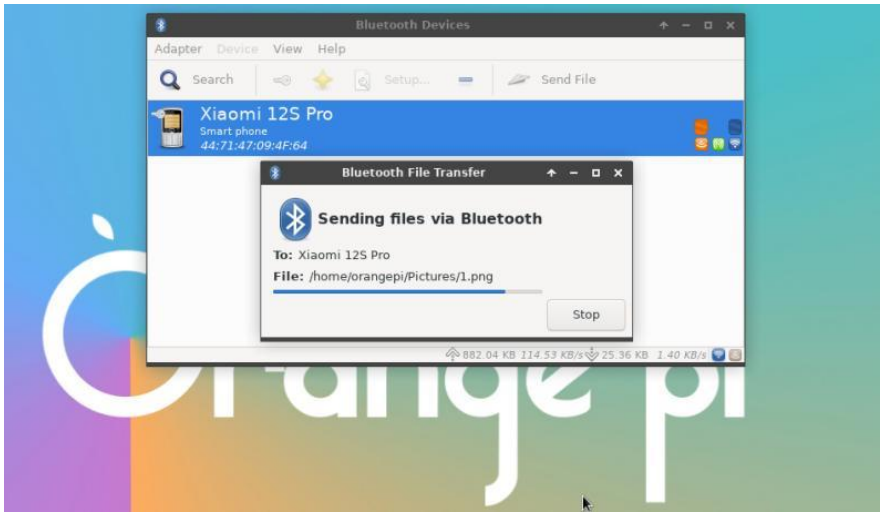

8) When pairing, the pairing confirmation box will pop up in the upper right corner of the desktop. Select **Confirm** to confirm. At this time, the mobile phone also needs to be confirmed



9) After pairing with the mobile phone, you can choose the paired Bluetooth device, then right-click and select **Send a File** to start sending a picture to the phone




10) The interface of the sending picture is shown below




## 3. 11. USB interface test

**The USB interface can be connected to a USB hub to expand the number of USB interfaces.**

### 3. 11. 1. Connect the USB mouse or keyboard to test

1) Insert the USB interface keyboard into the USB interface of the Orange Pi development board

2) Connect Orange PI development board to HDMI display

3) If the mouse or keyboard can operate normally, it means that the USB interface is working normally (the mouse can only be used in the desktop version of the system)

### 3.11.2. Connect the USB storage device test

1) First insert the U disk or USB mobile hard disk into the USB interface of the Orange Pi development board

2) Execute the following command, if you can see the output of sdX, it means that the U disk is recognized successfully

```
orangepi@orangepi:~$ cat /proc/partitions | grep "sd*"
major minor  #blocks  name
   8        0   30044160 sda
   8        1   30043119 sda1
```

3) Use the mount command to mount the U disk into **/mnt**, and then you can view the file in the U disk

```
orangepi@orangepi:~$ sudo mount /dev/sda1 /mnt/
orangepi@orangepi:~$ ls /mnt/
test.txt
```

4) After mounting, you can view the capacity usage and mount point of the U disk through the **df -h** command

```
orangepi@orangepi:~$ df -h | grep "sd"
/dev/sda1       29G  208K   29G   1% /mnt
```

### 3.11.3. USB wireless network card test

The currently **tested** USB wireless network cards are shown below. Please test it by yourself for other models of USB wireless network cards. If you cannot use it, you need to transplant the corresponding USB wireless network card driver

| No. | Model | |
|---|---|---|
| 1 | RTL8723BU<br>Support 2.4G WIFI+BT4.0 |  |

| | | |
|---|---|---|
| 2 | RTL8811<br>Support 2.4G +5G WIFI |  |
| 3 | RTL8821CU<br>Support 2.4G +5G WIFI<br>Support BT 4.2 |  |

### 3.11.3.1. RTL8723BU test

1) First insert the RTL8723BU wireless network card module into the USB interface of the development board

2) Then the Linux system will automatically load the RTL8723BU bluetooth and WIFI-related kernel modules, through the lsmod command, you can see that the following kernel modules have been automatically loaded

```
orangepi@orangepi:~$ lsmod
Module                  Size  Used by
rfcomm                 57344  16
rtl8xxxu              106496  0
rtk_btusb              61440  0
```

3) Through the dmesg command, you can see the loading information of the RTL8723BU module

```
orangepi@orangepi:~$ dmesg
......
[    83.438901] usb 2-1: new high-speed USB device number 2 using ehci-platform
[    83.588375] usb 2-1: New USB device found, idVendor=0bda, idProduct=b720, bcdDevice= 2.00
[    83.588403] usb 2-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[    83.588422] usb 2-1: Product: 802.11n WLAN Adapter
[    83.588443] usb 2-1: Manufacturer: Realtek
[    83.588460] usb 2-1: SerialNumber: 00e04c000001
[    83.601974] Bluetooth: hci0: RTL: examining hci_ver=06 hci_rev=000b lmp_ver=06
```

```
lmp_subver=8723
[    83.603894] Bluetooth: hci0: RTL: rom_version status=0 version=1
[    83.603920] Bluetooth: hci0: RTL: loading rtl_bt/rtl8723b_fw.bin
[    83.610108] Bluetooth: hci0: RTL: loading rtl_bt/rtl8723b_config.bin
[    83.611274] Bluetooth: hci0: RTL: cfg_sz 68, total sz 22564
[    83.658494] rtk_btusb: Realtek Bluetooth USB driver ver 3.1.6d45ddf.20220519-142432
[    83.658651] usbcore: registered new interface driver rtk_btusb
[    83.667124] usb 2-1: This Realtek USB WiFi dongle (0x0bda:0xb720) is untested!
[    83.667137] usb 2-1: Please report results to Jes.Sorensen@gmail.com
[    83.890140] usb 2-1: Vendor: Realtek
[    83.890153] usb 2-1: Product: 802.11n WLAN Adapter
[    83.890159] usb 2-1: rtl8723bu_parse_efuse: dumping efuse (0x200 bytes):
......
[    83.890412] usb 2-1: RTL8723BU rev E (SMIC) 1T1R, TX queues 3, WiFi=1, BT=1, GPS=0, HI PA=0
[    83.890417] usb 2-1: RTL8723BU MAC: 00:13:ef:f4:58:ae
[    83.890421] usb 2-1: rtl8xxxu: Loading firmware rtlwifi/rtl8723bu_nic.bin
[    83.895289] usb 2-1: Firmware revision 35.0 (signature 0x5301)
[    84.050893] Bluetooth: hci0: RTL: fw version 0x0e2f9f73
[    84.266905] Bluetooth: RFCOMM TTY layer initialized
[    84.266949] Bluetooth: RFCOMM socket layer initialized
[    84.266999] Bluetooth: RFCOMM ver 1.11
[    84.884270] usbcore: registered new interface driver rtl8xxxu
[    84.912046] rtl8xxxu 2-1:1.2 wlx0013eff458ae: renamed from wlan0
```

4) Then you can see the RTL8723BU WIFI device node through the **sudo ifconfig** command. Please refer to **the WIFI connection test** chapter for WIFI connection and testing methods

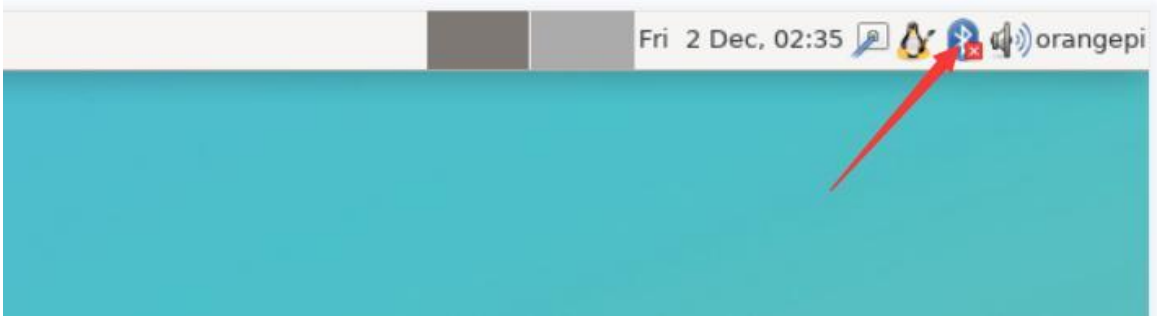
```
orangepi@orangepi:~$ sudo ifconfig wlx0013eff458ae
wlx0013eff458ae: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 00:13:ef:f4:58:ae  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
```

```
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

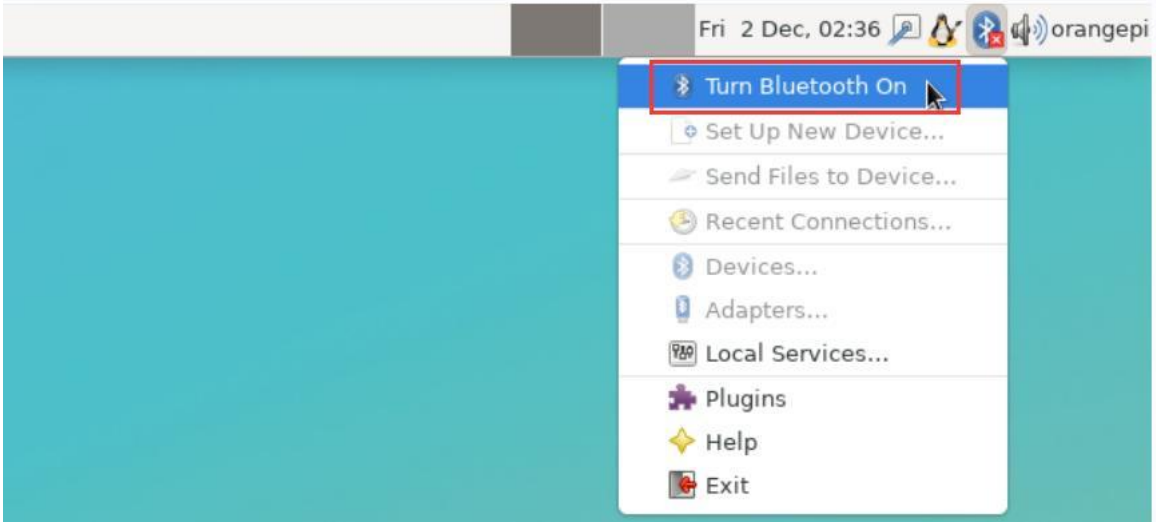
5) Then you can see the USB Bluetooth device through the **hciconfig** command

```
orangepi@orangepi:~$ sudo apt update && sudo apt install bluez
orangepi@orangepi:~$ hciconfig
hci0:   Type: Primary  Bus: USB
        BD Address: 00:13:EF:F4:58:AE  ACL MTU: 820:8  SCO MTU: 255:16
        DOWN
        RX bytes:1252 acl:0 sco:0 events:125 errors:0
        TX bytes:23307 acl:0 sco:0 commands:125 errors:0
```

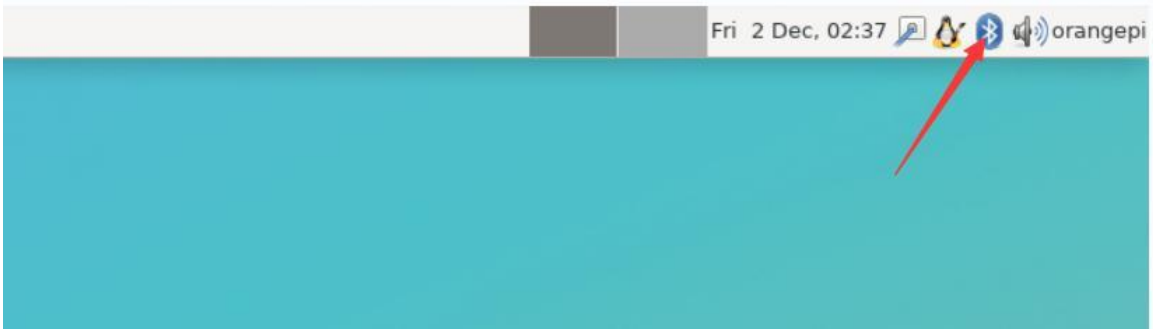
6) You can also see the Bluetooth icon on the desktop. At this time, Bluetooth has not been opened, so a red **x** will be displayed




7) Click **Turn Bluetooth On** to turn on Bluetooth




8) The display after turning on Bluetooth is as follows




9) Please refer to the **Bluetooth use method** for Bluetooth test method, so I won't go into details here.

### 3.11.3.2. RTL8811 test

1) First insert the RTL8811 wireless network card module into the USB interface of the development board.

2) Then the Linux system will automatically load the kernel modules related to RTL8811 WIFI, and you can see that the following kernel modules have been automatically loaded through the lsmod command

```
orangepi@orangepi:~$ lsmod
Module                  Size  Used by
8821cu               1839104  0
```

3) Through the dmesg command, you can see the loading information of the RTL8811 module

```
orangepi@orangepi:~$ dmesg
[   118.618194] usb 2-1: new high-speed USB device number 2 using ehci-platform
[   118.767152] usb 2-1: New USB device found, idVendor=0bda, idProduct=c811, bcdDevice= 2.00
[   118.767181] usb 2-1: New USB device strings: Mfr=1, Product=2, SerialNumber=3
[   118.767199] usb 2-1: Product: 802.11ac NIC
[   118.767219] usb 2-1: Manufacturer: Realtek
[   118.767235] usb 2-1: SerialNumber: 123456
[   119.500530] usbcore: registered new interface driver rtl8821cu
[   119.525498] rtl8821cu 2-1:1.0 wlx1cbfced9d260: renamed from wlan0
```

4) Then you can see the WiFi device node through the **sudo ifconfig** command. Please refer to the **WiFi connection test chapter** for WIFI connection and testing methods. I won't go into details here

```
orangepi@orangepi:~$ sudo ifconfig wlx1cbfced9d260
wlx1cbfced9d260: flags=4099<UP,BROADCAST,MULTICAST>  mtu 1500
        ether 1c:bf:ce:d9:d2:60  txqueuelen 1000  (Ethernet)
        RX packets 0  bytes 0 (0.0 B)
        RX errors 0  dropped 0  overruns 0  frame 0
        TX packets 0  bytes 0 (0.0 B)
        TX errors 0  dropped 0 overruns 0  carrier 0  collisions 0
```

### 3. 11. 4. USB Camera Test

1) First, you need to prepare a USB camera that supports UVC protocol as shown in the figure below or similar, and then insert the USB camera into the USB port of the Orange Pi development board

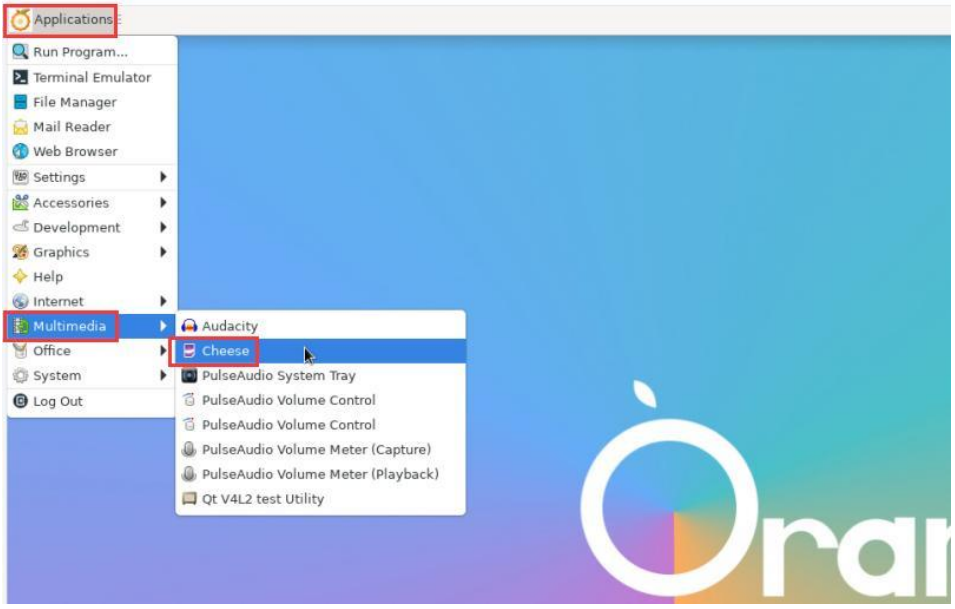

2) You can see that the USB camera's device node information is /dev/video0 through the v4l2-ctl command

```
orangepi@orangepi:~$ v4l2-ctl --list-devices
Q8 HD Webcam: Q8 HD Webcam (usb-fc880000.usb-1):
        /dev/video0
        /dev/video1
        /dev/media0
```

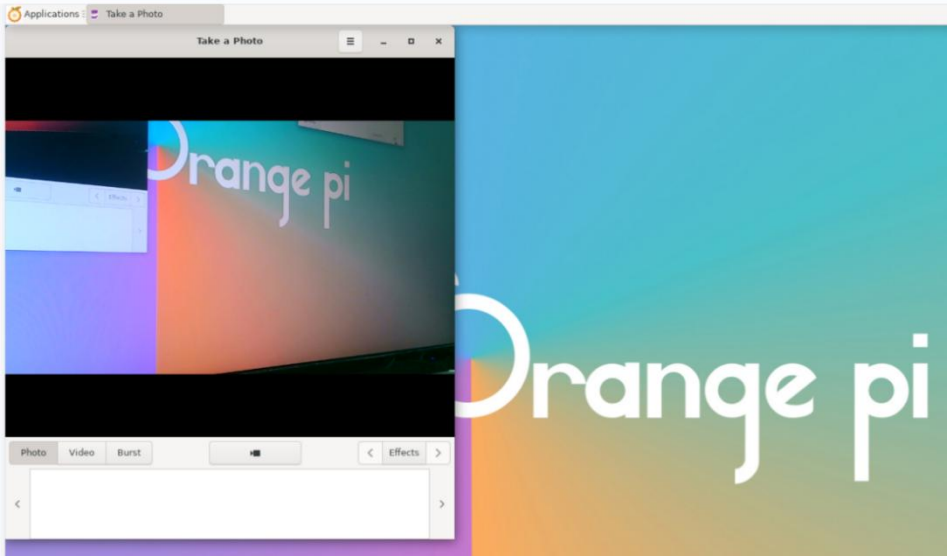
**Note that the l in v4l2 is a lowercase letter l, not the number 1.**

**In addition, the serial number of the video is not necessarily video0, please refer to what you actually see.**

3) In the desktop system, Cheese can be used to directly open the USB camera. The method of opening Cheese is shown in the figure below:




The interface after Cheese turns on the USB camera is shown in the figure below:




4) How to test the USB camera using fswebcam

a. Install fswebcam

```
orangepi@orangepi:~$ sudo apt update
orangepi@orangepi:~$ sudo apt-get install -y fswebcam
```

b. After installing fswebcam, you can use the following command to take pictures

a) -d The option is used to specify the device node of the USB camera
b) --no-banner Used to remove the watermark of photos
c) -r The option is used to specify the resolution of the photo
d) -S The option is used to set the number of previous frames to skip
e) ./image.jpg The name and path for setting the generated photos

```
orangepi@orangepi:~$ sudo fswebcam -d /dev/video0   \
--no-banner -r 1280x720 -S 5 ./image.jpg
```

c. In the server version of the Linux system, you can use the scp command to transfer the taken pictures to the Ubuntu PC for mirror viewing after taking pictures

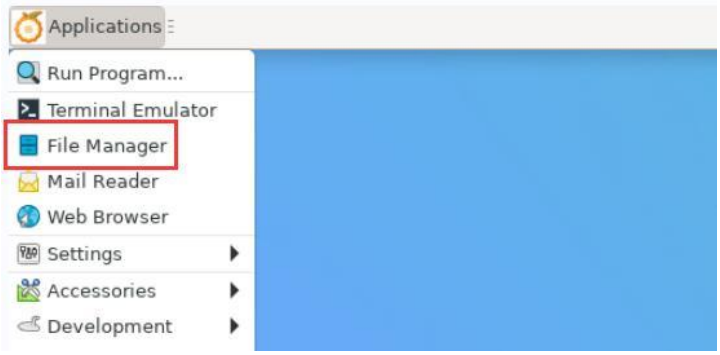
```
orangepi@orangepi:~$ scp image.jpg test@192.168.1.55:/home/test （Modify the IP address and path according to the actual situation）
```

d. In the desktop version of the Linux system, you can directly view the captured pictures through the HDMI display

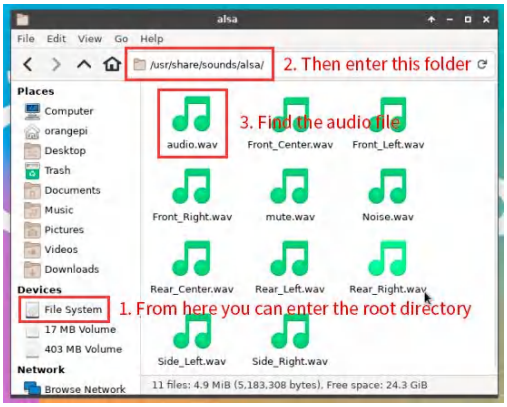
## 3. 12. Audio Test

### 3. 12. 1. Test audio methods in the desktop system

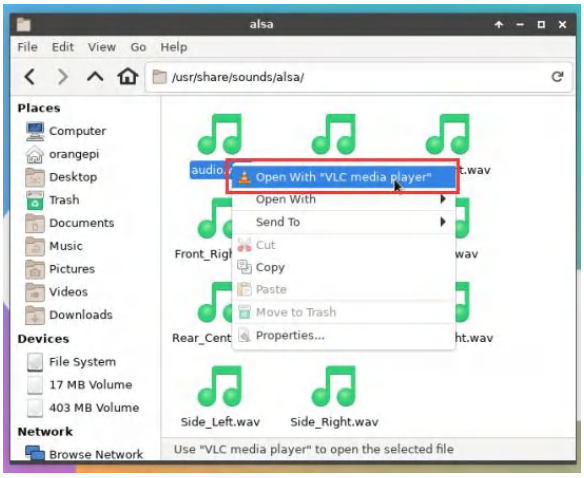
1) First open the file manager




2) Then find the following file (if there is no audio file in the system, you can upload a audio file to the system by yourself)

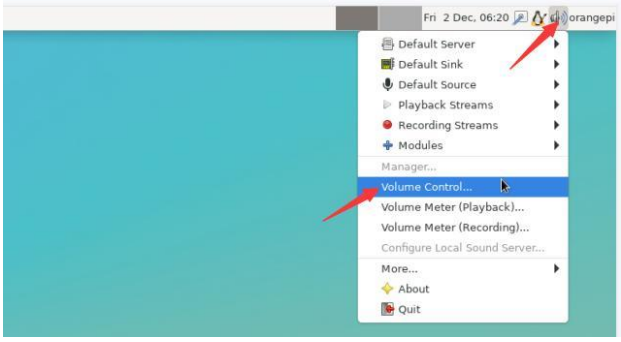



3) Then select the audio.wav file, right click and select open with vlc to start playing

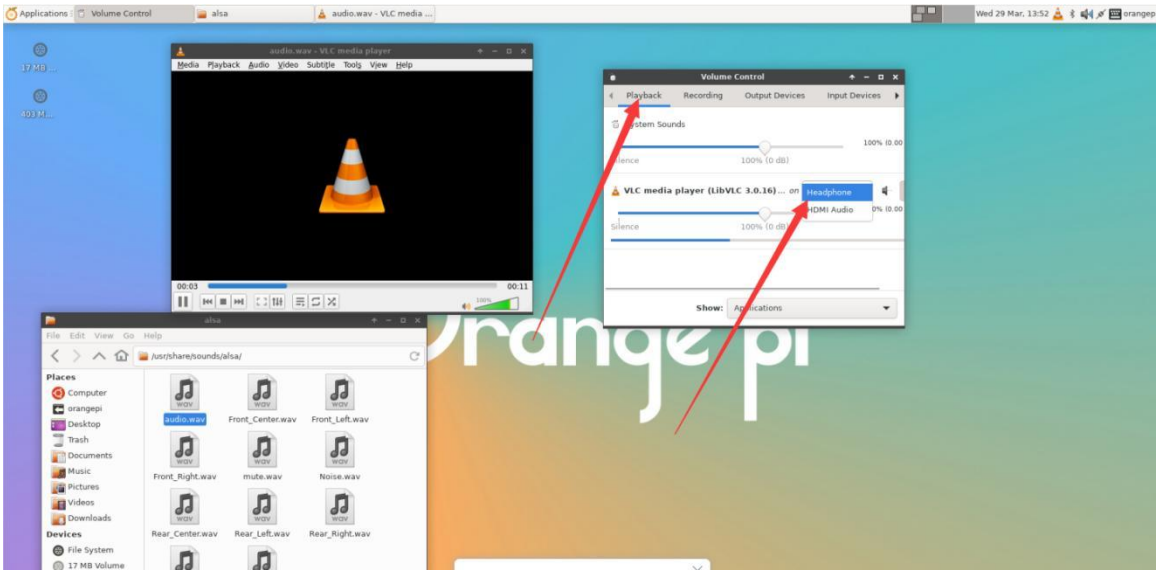



4) Methods to switch different audio equipment such as HDMI playback and headset playback

a. First open the volume control interface




b. When playing audio, the audio equipment options that play software can be used will be displayed in **Playback**. As shown in the figure below, which audio equipment you need to play here can be set.

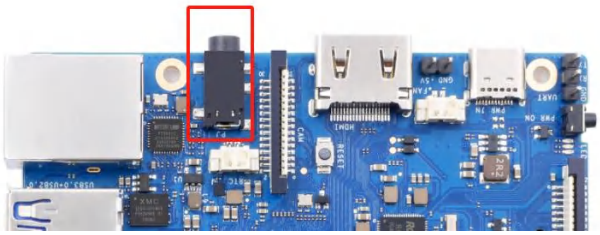


## 3. 12. 2. How to play audio with commands

### 3. 12. 2. 1. Headphone interface play audio test

**Note that linux6.6 is not supported yet.**

1) First insert the headset into the headphone jack of the development board



2) Then you can use the **aplay -l** command to check the sound card devices supported by the Linux system

   a. The output of a.linux5.10 system is as follows. Card 0 is the sound card device of rk809, which is the sound card device of the headset.

```
orangepi@orangepi:~$ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: rockchiprk809 [rockchip-rk809], device 0: dailink-multicodecs rk817-hifi-0 [dailink-multicodecs rk817-hifi-0]
  Subdevices: 0/1
```

```
  Subdevice #0: subdevice #0
card 1: rockchiphdmi [rockchip,hdmi], device 0: fe400000.i2s-i2s-hifi i2s-hifi-0 [fe400000.i2s-i2s-hifi i2s-hifi-0]
  Subdevices: 0/1
  Subdevice #0: subdevice #0
Subdevice #0: subdevice #0
```

3) Then use the **aplay** command to play the audio file that comes with the system. If the headset can hear the sound, it means that the hardware can be used normally.

a. Playback command for linux5.10 system:

```
orangepi@orangepi:~$ aplay -D hw:0,0 /usr/share/sounds/alsa/audio.wav
Playing WAVE 'audio.wav' : Signed 16 bit Little Endian, Rate 44100 Hz, Stereo
```

### 3.12.2.2. HDMI Audio Play Test

1) First use the HDMI to HDMI cable to connect the Orange PI development board to the TV (other HDMI displays need to ensure that the audio can be played)

2) Then check the HDMI sound card serial number. From the output below, you can know that the HDMI sound card is **card 1**.

a. The output of the linux5.10 system is as follows, card 1 is the HDMI sound card device

```
orangepi@orangepi:~$ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: rockchiprk809 [rockchip-rk809], device 0: dailink-multicodecs rk817-hifi-0 [dailink-multicodecs rk817-hifi-0]
  Subdevices: 0/1
  Subdevice #0: subdevice #0
card 1: rockchiphdmi [rockchip,hdmi], device 0: fe400000.i2s-i2s-hifi i2s-hifi-0 [fe400000.i2s-i2s-hifi i2s-hifi-0]
  Subdevices: 0/1
  Subdevice #0: subdevice #0
  Subdevice #0: subdevice #0
```

b. The output of the b.linux6.6 system is as follows. Card 0 is the HDMI sound card device

```
orangepi@orangepi:~$ aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: HDMI [HDMI], device 0: fe400000.i2s-i2s-hifi i2s-hifi-0 [fe400000.i2s-i2s-hifi i2s-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 2: RK809 [Analog RK809], device 0: fe410000.i2s-rk817-hifi rk817-hifi-0 [fe410000.i2s-rk817-hifi rk817-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
```

3) Then use the **aplay** command to play the audio file that comes with the system. If the sound can be heard on the HDMI display or TV, it means that the hardware can be used normally.

a. Playback command for linux5.10 system:

```
orangepi@orangepi:~$ aplay -D hw:1,0 /usr/share/sounds/alsa/audio.wav
```

b. Playback command for linux6.6 system

```
orangepi@orangepi:~$ aplay -D hw:0,0 /usr/share/sounds/alsa/audio.wav
```

### 3. 12. 3. Use the command to test the recording method

**Note that linux6.6 is not supported yet.**

1) The Orange Pi 3B development board does not have an onboard MIC, and audio can only be recorded through headphones with a MIC function. After inserting the headset with MIC function into the development board, run the following command to record an audio period through the headset.

a. Commands for linux5.10 system:

```
orangepi@orangepi:~$ amixer -c 0 cset name='Capture MIC Path' 'Main Mic'
orangepi@orangepi:~$ arecord -D hw:0,0 -d 5 -f cd -t wav /tmp/test.wav
```

## 3. 13. Temperature Sensor

1) The command to view the system temperature sensor is:

```
orangepi@orangepi:~$ sensors
soc_thermal-virtual-0
Adapter: Virtual device
```

```
temp1:          +41.9°C  (crit = +115.0°C)

gpu_thermal-virtual-0
Adapter: Virtual device
temp1:          +43.8°C
```

2) The command to view the current temperature of the nvme ssd solid state drive is:

```
orangepi@orangepi:~$ sudo smartctl -a /dev/nvme0 | grep "Temperature:"
Temperature:                        40 Celsius
```

## 3. 14. 40 Pin interface pin explanation

2) Orange Pi 3B Development board 40 Pin interface pins, please refer to the figure below


3) The function of the Orange Pi 3B development board 40 PIN interface pins is shown in the table below.

a. Below is a complete pins of 40pin

| 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 | 复用功能 |
|---|---|---|---|---|---|---|---|---|---|
| | | 3.3V | | 1 | 2 | | 5V | | |
| | I2C2_SDA_M1 | GPIO4_B4 | 140 | 3 | 4 | | 5V | | |
| | I2C2_SCL_M1 | GPIO4_B5 | 141 | 5 | 6 | | GND | | |
| | PWM15_IR_M1(fe700030) | GPIO4_C3 | 147 | 7 | 8 | 25 | GPIO0_D1 | UART2_TX_M0 | |
| | | GND | | 9 | 10 | 24 | GPIO0_D0 | UART2_RX_M0 | |
| | | GPIO3_C6 | 118 | 11 | 12 | 119 | GPIO3_C7 | | |
| | | GPIO4_A0 | 128 | 13 | 14 | | GND | | |
| | UART7_TX_M2 | GPIO4_A2 | 130 | 15 | 16 | 131 | GPIO4_A3 | UART7_RX_M2 | |
| | | 3.3V | | 17 | 18 | 129 | GPIO4_A1 | | |
| I2C4_SDA_M0 | SPI3_MOSI_M0 | GPIO4_B2 | 138 | 19 | 20 | | GND | | |
| | SPI3_MISO_M0 | GPIO4_B0 | 136 | 21 | 22 | 132 | GPIO4_A4 | UART9_TX_M2 | |
| I2C4_SCL_M0 | SPI3_CLK_M0 | GPIO4_B3 | 139 | 23 | 24 | 134 | GPIO4_A6 | SPI3_CS0_M0 | |
| | | GND | | 25 | 26 | 126 | GPIO3_D6 | | |
| UART3_RX_M0 | I2C3_SDA_M0 | GPIO1_A0 | 32 | 27 | 28 | 33 | GPIO1_A1 | I2C3_SCL_M0 | UART3_TX_M0 |
| | UART9_RX_M2 | GPIO4_A5 | 133 | 29 | 30 | | GND | | |
| | | GPIO3_D4 | 124 | 31 | 32 | 144 | GPIO4_C0 | PWM11_IR_M1(fe6f0030) | |
| | | GPIO3_D7 | 127 | 33 | 34 | | GND | | |
| | | GPIO3_D0 | 120 | 35 | 36 | 125 | GPIO3_D5 | | |
| | | GPIO3_D3 | 123 | 37 | 38 | 122 | GPIO3_D2 | | |
| | | GND | | 39 | 40 | 121 | GPIO3_D1 | | |

b. The following form is a picture on the left half of the full table above, which can

be seen clearly

| 复用功能 | 复用功能 | GPIO | GPIO序号 | 引脚序号 |
|---|---|---|---|---|
| | | 3.3V | | 1 |
| | I2C2_SDA_M1 | GPIO4_B4 | 140 | 3 |
| | I2C2_SCL_M1 | GPIO4_B5 | 141 | 5 |
| | PWM15_IR_M1(fe700030) | GPIO4_C3 | 147 | 7 |
| | | GND | | 9 |
| | | GPIO3_C6 | 118 | 11 |
| | | GPIO4_A0 | 128 | 13 |
| | UART7_TX_M2 | GPIO4_A2 | 130 | 15 |
| | | 3.3V | | 17 |
| I2C4_SDA_M0 | SPI3_MOSI_M0 | GPIO4_B2 | 138 | 19 |
| | SPI3_MISO_M0 | GPIO4_B0 | 136 | 21 |
| I2C4_SCL_M0 | SPI3_CLK_M0 | GPIO4_B3 | 139 | 23 |
| | | GND | | 25 |
| UART3_RX_M0 | I2C3_SDA_M0 | GPIO1_A0 | 32 | 27 |
| | UART9_RX_M2 | GPIO4_A5 | 133 | 29 |
| | | GPIO3_D4 | 124 | 31 |
| | | GPIO3_D7 | 127 | 33 |
| | | GPIO3_D0 | 120 | 35 |
| | | GPIO3_D3 | 123 | 37 |
| | | GND | | 39 |

c. The following form is a picture on the right half of the top table above, which can be seen clearly

| 引脚序号 | GPIO序号 | GPIO | 复用功能 | 复用功能 |
|---|---|---|---|---|
| 2 | | 5V | | |
| 4 | | 5V | | |
| 6 | | GND | | |
| 8 | 25 | GPIO0_D1 | UART2_TX_M0 | |
| 10 | 24 | GPIO0_D0 | UART2_RX_M0 | |
| 12 | 119 | GPIO3_C7 | | |
| 14 | | GND | | |
| 16 | 131 | GPIO4_A3 | UART7_RX_M2 | |
| 18 | 129 | GPIO4_A1 | | |
| 20 | | GND | | |
| 22 | 132 | GPIO4_A4 | UART9_TX_M2 | |
| 24 | 134 | GPIO4_A6 | SPI3_CS0_M0 | |
| 26 | 126 | GPIO3_D6 | | |
| 28 | 33 | GPIO1_A1 | I2C3_SCL_M0 | UART3_TX_M0 |
| 30 | | GND | | |
| 32 | 144 | GPIO4_C0 | PWM11_IR_M1(fe6f0030) | |
| 34 | | GND | | |
| 36 | 125 | GPIO3_D5 | | |
| 38 | 122 | GPIO3_D2 | | |
| 40 | 121 | GPIO3_D1 | | |

4) There are a total of **28** GPIO ports in the 40pin interface. The voltage of all GPIO ports is **3.3v**

## 3.15. How to install wiringOP

**Note that wiringOP has been pre-installed in the Linux image released by Orange Pi. Unless the code of wiringOP is updated, there is no need to re-download,**

**compile and install, and use it directly.**

**The storage path of the compiled wiringOP deb package in orangepi-build is: orangepi-build/external/cache/debs/arm64/wiringpi_x.xx.deb**

**After entering the system, you can run the gpio readall command. If you can see the following output, it means that wiringOP has been pre-installed and can be used normally.**

```
root@orangepi3b:~# gpio readall
 +------+-----+----------+--------+---+   PI3B   +---+--------+----------+-----+------+
 | GPIO | wPi |   Name   |  Mode  | V | Physical | V |  Mode  | Name     | wPi | GPIO |
 +------+-----+----------+--------+---+----++----+---+--------+----------+-----+------+
 |      |     |     3.3V |        |   |  1 || 2  |   |        | 5V       |     |      |
 |  140 |   0 |    SDA.2 |     IN | 1 |  3 || 4  |   |        | 5V       |     |      |
 |  141 |   1 |    SCL.2 |     IN | 1 |  5 || 6  |   |        | GND      |     |      |
 |  147 |   2 |    PWM15 |     IN | 0 |  7 || 8  | 1 | ALT1   | RXD.2    | 3   | 25   |
 |      |     |      GND |        |   |  9 || 10 | 1 | ALT1   | TXD.2    | 4   | 24   |
 |  118 |   5 | GPIO3_C6 |     IN | 0 | 11 || 12 | 0 | IN     | GPIO3_C7 | 6   | 119  |
 |  128 |   7 | GPIO4_A0 |     IN | 0 | 13 || 14 |   |        | GND      |     |      |
 |  130 |   8 |    TXD.7 |     IN | 0 | 15 || 16 | 0 | IN     | RXD.7    | 9   | 131  |
 |      |     |     3.3V |        |   | 17 || 18 | 0 | IN     | GPIO4_A1 | 10  | 129  |
 |  138 |  11 | SPI3_TXD |     IN | 0 | 19 || 20 |   |        | GND      |     |      |
 |  136 |  12 | SPI3_RXD |     IN | 0 | 21 || 22 | 0 | IN     | TXD.9    | 13  | 132  |
 |  139 |  14 | SPI3_CLK |     IN | 0 | 23 || 24 | 0 | IN     | SPI3_CS1 | 15  | 134  |
 |      |     |      GND |        |   | 25 || 26 | 0 | IN     | GPIO3_D6 | 16  | 126  |
 |   32 |  17 |    SDA.3 |     IN | 1 | 27 || 28 | 1 | IN     | SCL.3    | 18  | 33   |
 |  133 |  19 |    RXD.9 |     IN | 0 | 29 || 30 |   |        | GND      |     |      |
 |  124 |  20 | GPIO3_D4 |     IN | 0 | 31 || 32 | 0 | IN     | PWM11    | 21  | 144  |
 |  127 |  22 | GPIO3_D7 |     IN | 0 | 33 || 34 |   |        | GND      |     |      |
 |  120 |  23 | GPIO3_D0 |     IN | 0 | 35 || 36 | 0 | IN     | GPIO3_D5 | 24  | 125  |
 |  123 |  25 | GPIO3_D3 |     IN | 0 | 37 || 38 | 0 | IN     | GPIO3_D2 | 26  | 122  |
 |      |     |      GND |        |   | 39 || 40 | 0 | IN     | GPIO3_D1 | 27  | 121  |
 +------+-----+----------+--------+---+----++----+---+--------+----------+-----+------+
 | GPIO | wPi |   Name   |  Mode  | V | Physical | V |  Mode  | Name     | wPi | GPIO |
 +------+-----+----------+--------+---+   PI3B   +---+--------+----------+-----+------+
```

1) Download the code of wiringOP

```
orangepi@orangepi:~$ sudo apt update
orangepi@orangepi:~$ sudo apt install -y git
orangepi@orangepi:~$ git clone https://github.com/orangepi-xunlong/wiringOP.git -b next
```

**Note that Orange Pi 3B needs to download the code of the wiringOP next branch, please don't miss the parameter of -b next。**

**If there is a problem with the download code from GitHub, you can use the wiringOP source code that comes with the Linux image directly, and the storage**

**location is:/usr/src/wiringOP**

2) Compile and install wiringOP

orangepi@orangepi:~$ **cd wiringOP**
orangepi@orangepi:~/wiringOP$ **sudo ./build clean**
orangepi@orangepi:~/wiringOP$ **sudo ./build**

3) Test the output of the gpio readall command as follows

```
root@orangepi3b:~# gpio readall
 +------+-----+----------+--------+---+   PI3B   +---+--------+----------+-----+------+
 | GPIO | wPi |   Name   |  Mode  | V | Physical | V |  Mode  | Name     | wPi | GPIO |
 +------+-----+----------+--------+---+----++----+---+--------+----------+-----+------+
 |      |     |     3.3V |        |   |  1 || 2  |   |        | 5V       |     |      |
 |  140 |   0 |    SDA.2 |     IN | 1 |  3 || 4  |   |        | 5V       |     |      |
 |  141 |   1 |    SCL.2 |     IN | 1 |  5 || 6  |   |        | GND      |     |      |
 |  147 |   2 |    PWM15 |     IN | 0 |  7 || 8  | 1 | ALT1   | RXD.2    | 3   | 25   |
 |      |     |      GND |        |   |  9 || 10 | 1 | ALT1   | TXD.2    | 4   | 24   |
 |  118 |   5 | GPIO3_C6 |     IN | 0 | 11 || 12 | 0 | IN     | GPIO3_C7 | 6   | 119  |
 |  128 |   7 | GPIO4_A0 |     IN | 0 | 13 || 14 |   |        | GND      |     |      |
 |  130 |   8 |    TXD.7 |     IN | 0 | 15 || 16 | 0 | IN     | RXD.7    | 9   | 131  |
 |      |     |     3.3V |        |   | 17 || 18 | 0 | IN     | GPIO4_A1 | 10  | 129  |
 |  138 |  11 | SPI3_TXD |     IN | 0 | 19 || 20 |   |        | GND      |     |      |
 |  136 |  12 | SPI3_RXD |     IN | 0 | 21 || 22 | 0 | IN     | TXD.9    | 13  | 132  |
 |  139 |  14 | SPI3_CLK |     IN | 0 | 23 || 24 | 0 | IN     | SPI3_CS1 | 15  | 134  |
 |      |     |      GND |        |   | 25 || 26 | 0 | IN     | GPIO3_D6 | 16  | 126  |
 |   32 |  17 |    SDA.3 |     IN | 1 | 27 || 28 | 1 | IN     | SCL.3    | 18  | 33   |
 |  133 |  19 |    RXD.9 |     IN | 0 | 29 || 30 |   |        | GND      |     |      |
 |  124 |  20 | GPIO3_D4 |     IN | 0 | 31 || 32 | 0 | IN     | PWM11    | 21  | 144  |
 |  127 |  22 | GPIO3_D7 |     IN | 0 | 33 || 34 |   |        | GND      |     |      |
 |  120 |  23 | GPIO3_D0 |     IN | 0 | 35 || 36 | 0 | IN     | GPIO3_D5 | 24  | 125  |
 |  123 |  25 | GPIO3_D3 |     IN | 0 | 37 || 38 | 0 | IN     | GPIO3_D2 | 26  | 122  |
 |      |     |      GND |        |   | 39 || 40 | 0 | IN     | GPIO3_D1 | 27  | 121  |
 +------+-----+----------+--------+---+----++----+---+--------+----------+-----+------+
 | GPIO | wPi |   Name   |  Mode  | V | Physical | V |  Mode  | Name     | wPi | GPIO |
 +------+-----+----------+--------+---+   PI3B   +---+--------+----------+-----+------+
```

## 3. 16. 40Pin interface GPIO, I2C, UART, SPI, and PWM test

**Note that if you need to set overlays to open multiple configurations at the same time, please use a space to write in one line like the following space.**

orangepi@orangepi:~$ **sudo vim /boot/orangepiEnv.txt**

**overlays=spi3-m0-cs0-spidev i2c2-m1 i2c3-m0 uart7-m2 uart9-m2 pwm11-m1**

### 3. 16. 1. 40pin GPIO port test

**The Linux system released by Orange Pi has a pre-installed blink_all_gpio program, which will set all 28 GPIO ports in the 40pin to switch between high and low levels continuously.**