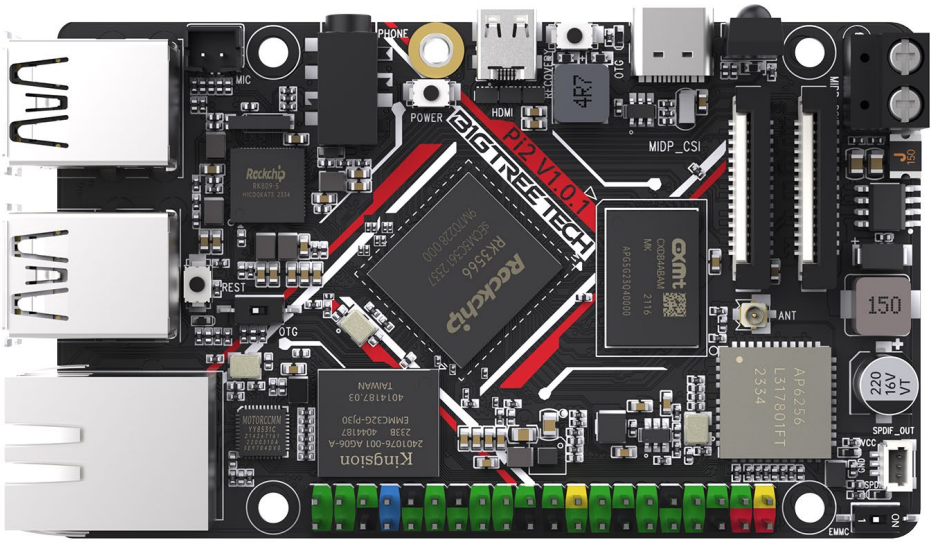
BIGTREETECH

Pi2

User Manual



Revision Log

Version	Date	Revision
v1.00	April 24th, 2024	Initial Version

Table of Contents

1. Product Profile

BIGTREETECH Pi2 uses the higher performance quad-core A55-RK3566 chipset, has rich interface functions, built-in eMMC5.1, onboard support for 2.4G/5G dual-mode, WiFi transmission speeds up to 433.3Mbps, also supports Bluetooth BT5.2 version. It has the same mounting hole positions as the Raspberry Pi, making installation and use convenient and quick.

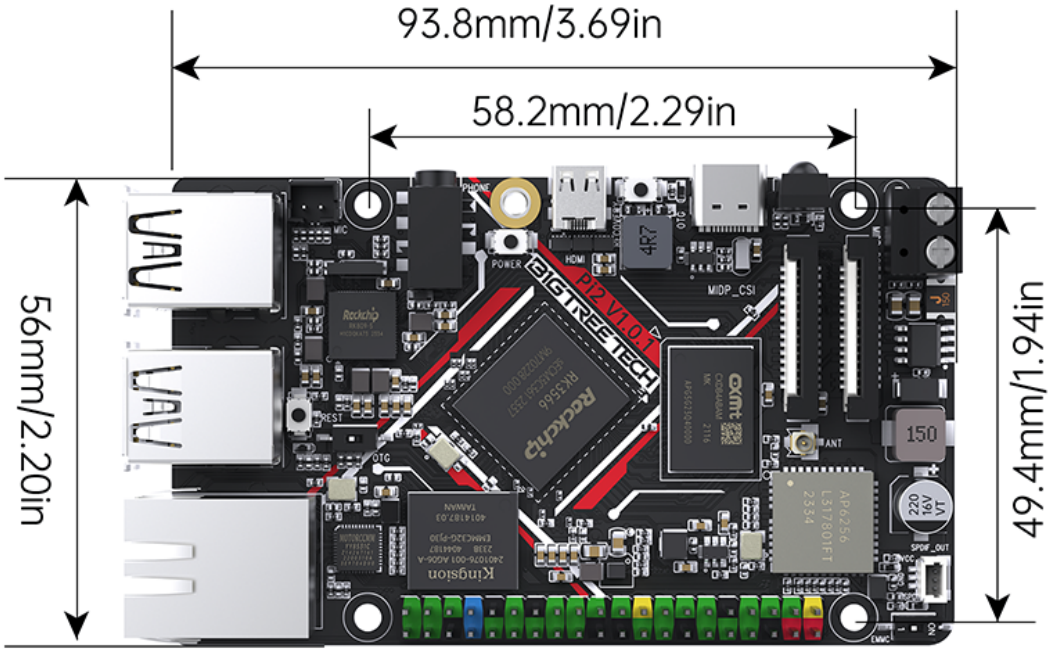
1.1. Feature Highlights

- CPU: Rockchip RK3566, quad-core Cortex-A55 @1.8GHz
- GPU: Mali-G52 1-Core-2EE
- NPU: 0.8 TOPS NPU
- RAM: 2GB LPDDR4 (customizable: 1GB/2GB/4GB/8GB)
- Onboard eMMC 32GB (customizable: 8GB/32GB/64GB/128GB...)
- MIPI DSI display support (320P-1080P 60Hz)
- SPI Flash: Customizable W25Q256JWEIQ
- Dual-lane MIPI CSI-2 Camera Interface (320P-1080P 60Hz)
- 3 x USB 2.0 ports, 1 x USB 3.0 port
- PCIe 2.1 1x1 Lane (Supports M.2 2242 4PIN+5PIN)
- MicroSD card slot (SDIO2.0)
- Networking: Gigabit Ethernet, 433Mbps WiFi, BT 5.2
- Audio: 3.5mm jack supports Mic input
- Capacitive Mic input
- 40-pin GPIO header
- HDMI 2.0 OUT (480P-4K 60Hz)
- Onboard infrared receiver (38kHz)
- Same mounting hole positions as Raspberry Pi
- 24V DC power input

1.2. Specifications

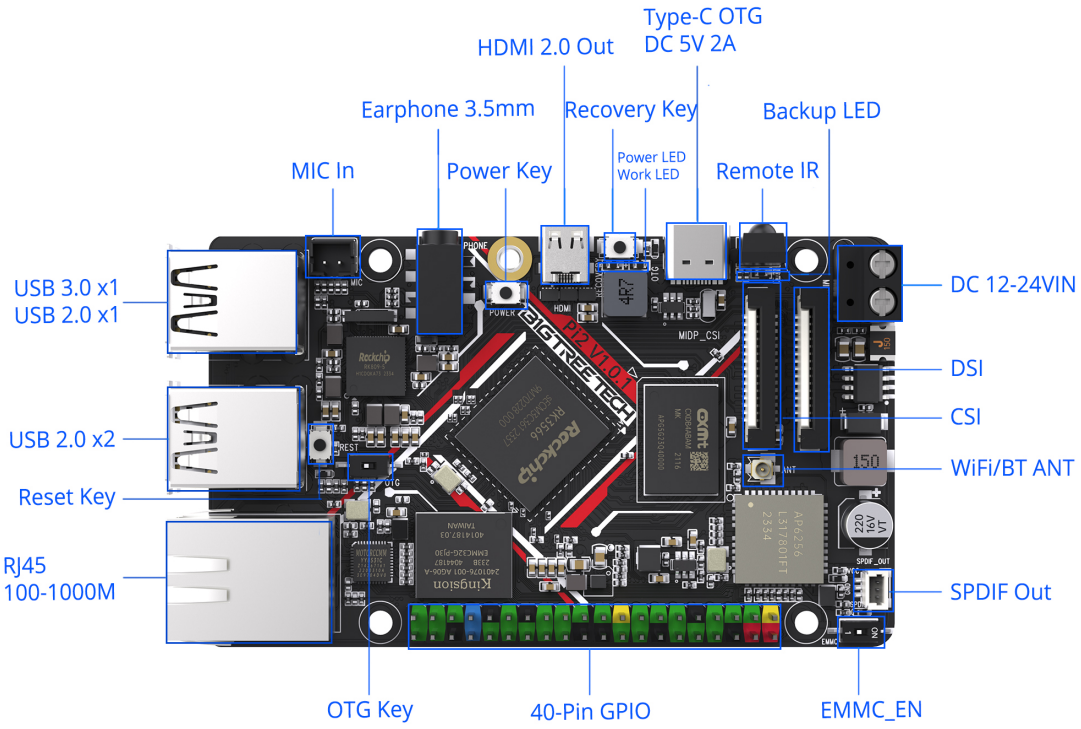
Dimensions: 93.8mm x 56mm
Installation Hole Spacing: 58.2mm x 49.4mm
USB-C Input: DC 5V±5%/2A
Output Voltage: 3.3V±2%/100mA
WiFi: 2.4G/5G, 802.11 ac/a/b/g/n/ wireless standards
Bluetooth: 5.2

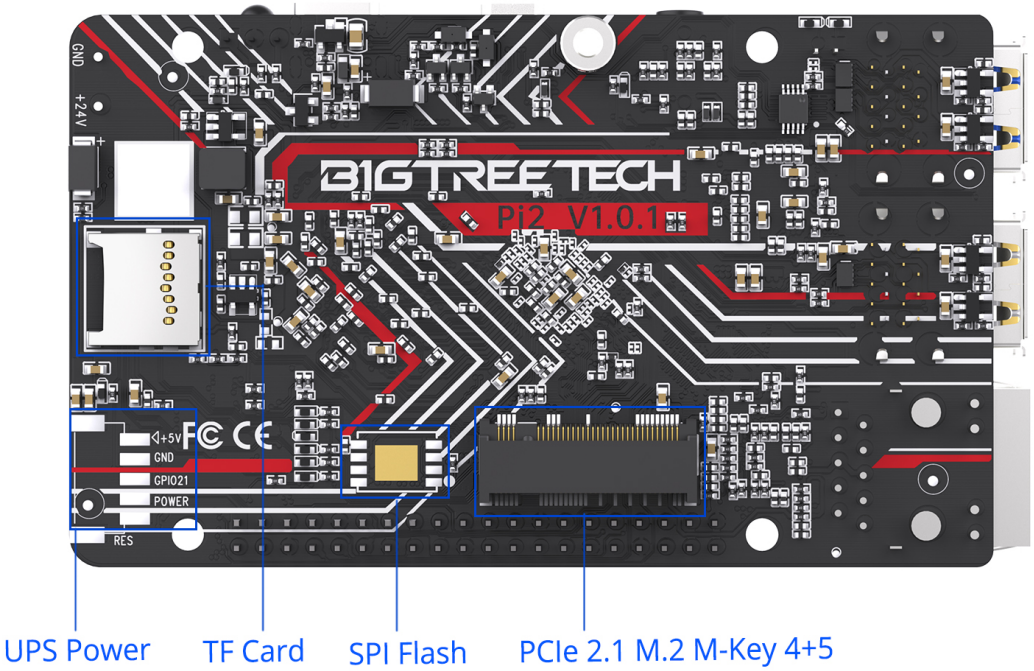
1.3. Dimensions



2. Peripheral Interface

2.1. Interface Diagram





3. Interface Introduction

3.1. Power Supply

Input:

- UBS-C: DC 5V 2A
- Terminal Block: DC 12-24 V

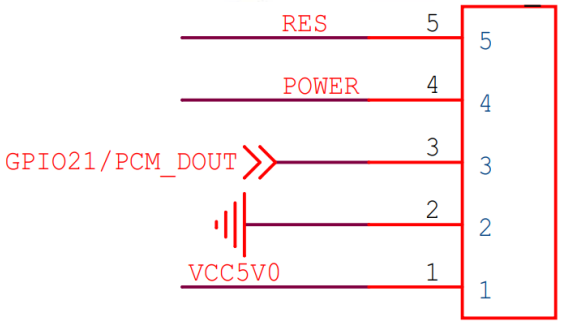
3.2. 40-pin GPIO

40Pin-GPIO

BTT Pi	CB1-eMMC	CB1	PI2/CB2	CM4		CM4	PI2/CB2	CB1	CB1-eMMC	BTT Pi
3.3V	3.3V	3.3V	3.3V	3.3V		5V	5V	5V	5V	5V
PC3	NC	NC	GPIO4_B2	GPIO 2 (I2C1_SDA)		5V	5V	5V	5V	5V
PC0	NC	NC	GPIO4_B3	GPIO 3 (I2C1_SCL)		GND	GND	GND	GND	GND
PC7	PI14	PC7	GPIO3_A1	GPIO 4 (GPCLK0)		GPIO 14 (UART_TX)	GIPIO0_D1	TX	TX	TX
GND	GND	GND	GND	GND		GPIO 15 (UART_RX)	GIPIO0_D0	RX	RX	RX
PC14	PI15	PC14	GPIO0_C7	GPIO 17		GPIO 18 (PCM_CLK)	GIPIO0_B0	PC13	PI7	PC13
PC12	PI6	PC12	GPIO1_A0	GPIO 27		GND	GND	GND	GND	GND
PC10	PI4	PC10	GPIO1_A1	GPIO 22		GPIO 23	GPIO4_C6	PC11	PI5	PC11
3.3V	3.3V	3.3V	3.3V	3.3V		GPIO 24	GPIO4_A3	PC9	PI3	PC9
PH7	PH7	PH7	GPIO3_C1	GPIO 10 (SPI0_MOSI)		GND	GND	GND	GND	GND
PH8	PH8	PH8	GPIO3_C2	GPIO 9 (SPI0_MISO)		GPIO 25	GPIO0_C4	NC	NC	PG13
PH6	PH6	PH6	GPIO3_C3	GPIO 11 (SPI0_SCLK)		GPIO 8 (SPI0_CE0)	GPIO4_A2	NC	NC	PG12
GND	GND	GND	GND	GND		GPIO 7 (SPI0_CE1)	GPIO0_A6	PG8	PI11	PI9
PC2	NC	NC	GPIO0_B4	GPIO 0 (EEPROM_SDA)		GPIO 1 (EEPROM_SCL)	GIPIO0_B3	PG7	PI10	PI10
PC4	NC	NC	GPIO3_D6	GPIO 5		GND	GND	GND	GND	GND
PI5	PI9	PG6	GPIO3_D7	GPIO 6		GPIO 12 (PWM0)	GPIO0_C1	PG9	PI12	PI6
PI14	NC	NC	GPIO0_C0	GPIO 13 (PWM1)		GND	GND	GND	GND	GND
PC6	PI1	PC6	GPIO4_C5	GPIO 19 (PCM_FS)		GPIO 16	GPIO0_A0	NC	NC	PG11
PC15	PI13	PC15	GPIO0_C3	GPIO 26		GPIO 20 (PCM_DIN)	GPIO4_C3	PH10	PH10	PH4
GND	GND	GND	GND	GND		GPIO 21 (PCM_DOUT)	GPIO4_C2	PC8	PI2	PC8

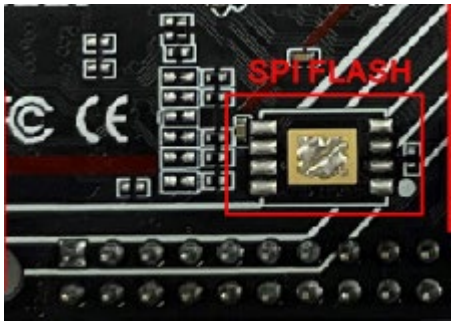
The method for calculating GPIO pins is as follows:
GPIO4_B2 = (GPIO4 - GPIO0) * 32 + ('B' - 'A') * 8 + 2 = 4 * 32 + 1 * 8 + 2 = gpio138
GPIO3_D7 = (GPIO3 - GPIO0) * 32 + ('D' - 'A') * 8 + 7 = 3 * 32 + 3 * 8 + 7 = gpio127

3.3. UPS POWER



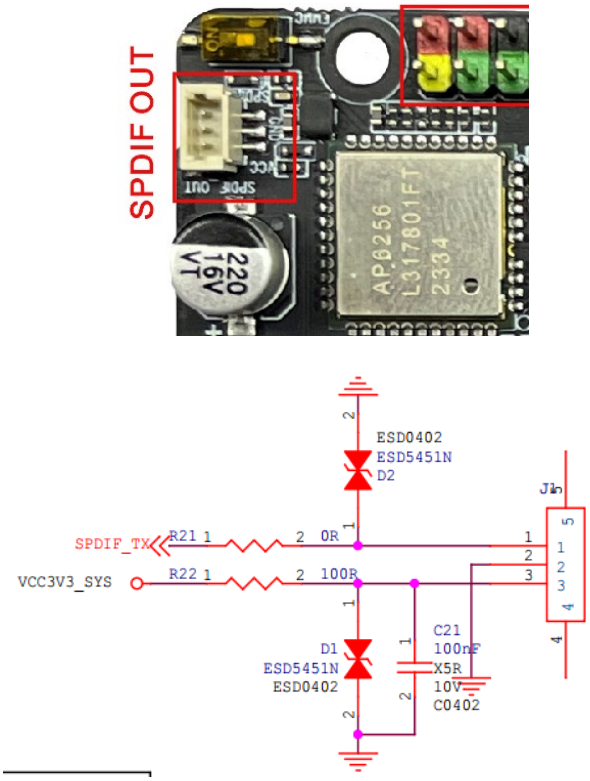
Specifications: HX5P-2.54MM horizontal type, compatible with our company's SKSM emergency power supply board.

3.4. SPI FLASH



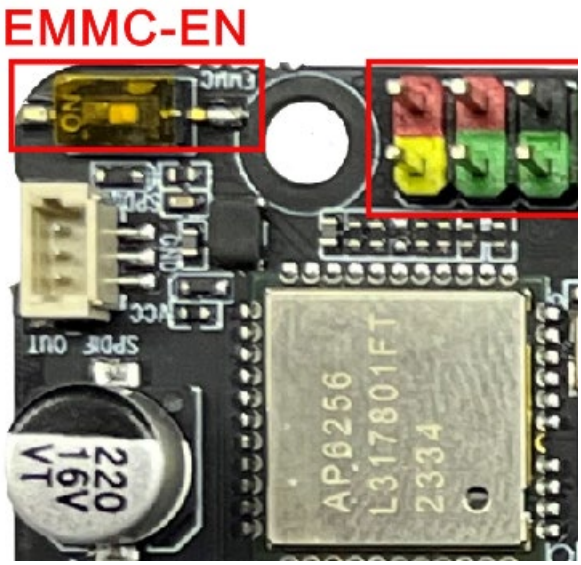
Specification model: W25Q256JWEIQ (NOT included. If needed, please contact us for customization.)

3.5. SPDIF OUT



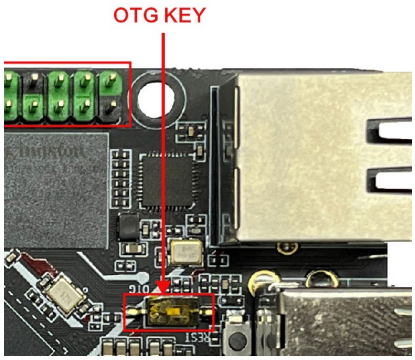
3.6. eMMC-EN

Default position is OFF, which means the eMMC can be used normally. If you do not wish to boot from eMMC, switch the EMMC-EN position to ON. This will short the eMMC signal lines to GND, disabling eMMC boot.

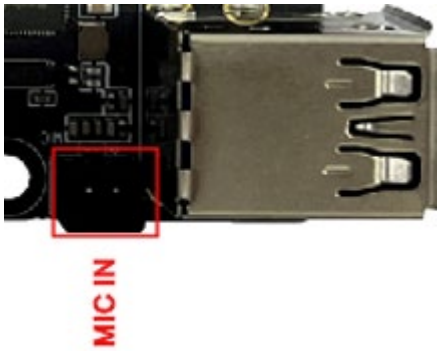


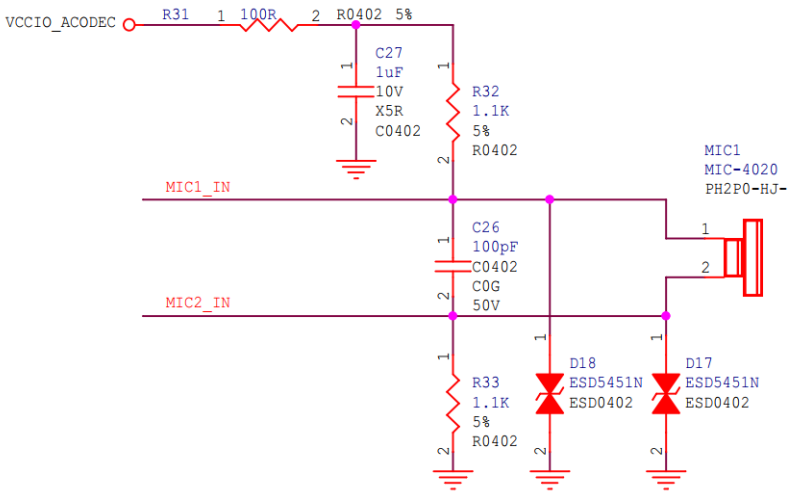
3.7. OTG

To enable OTG mode, switch the OTG KEY to the ON position. (Note: The black USB 2.0 port will not function properly in this mode.)

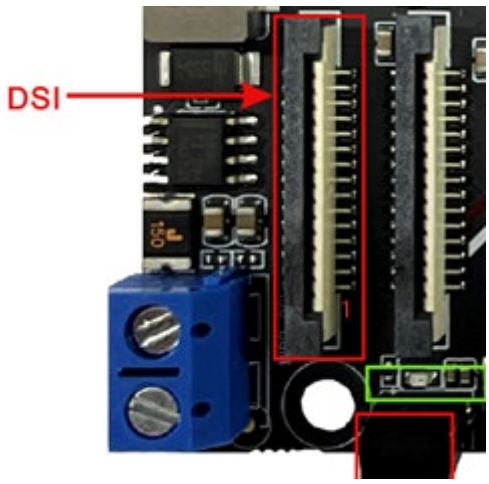


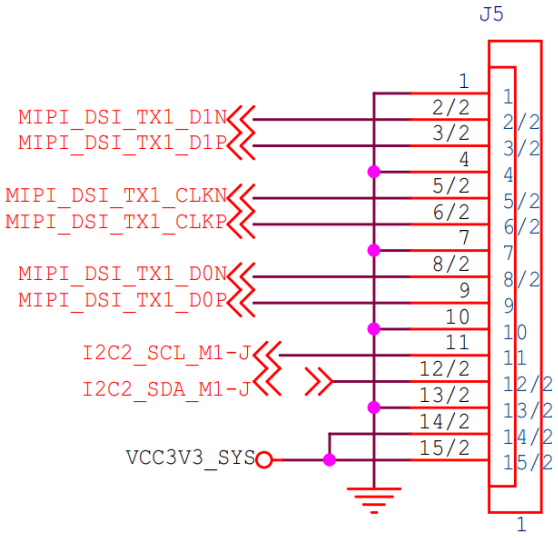
3.8. MIC IN



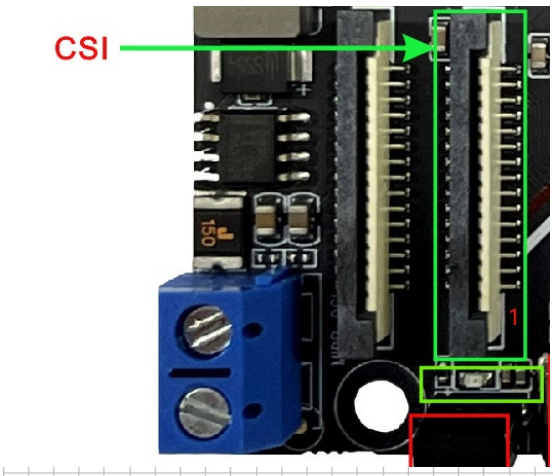


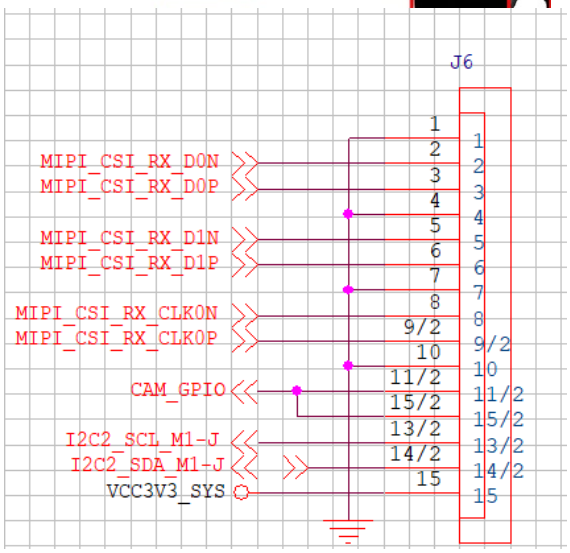
3.9. DSI


DSI


J5
MIPI_DSI_TX1_D1N
MIPI_DSI_TX1_D1P
MIPI_DSI_TX1_CLKN
MIPI_DSI_TX1_CLKP
MIPI_DSI_TX1_D0N
MIPI_DSI_TX1_D0P
I2C2_SCL_M1-J
I2C2_SDA_M1-J
VCC3V3_SYS
1
2/2
3/2
4
5/2
6/2
7
8/2
9
10
11
12/2
13/2
14/2
15/2

3.10. CSI


CSI
1


J6
MIPI_CSI_RX_D0N
MIPI_CSI_RX_D0P
MIPI_CSI_RX_D1N
MIPI_CSI_RX_D1P
MIPI_CSI_RX_CLK0N
MIPI_CSI_RX_CLK0P
CAM_GPIO
I2C2_SCL_M1-J
I2C2_SDA_M1-J
VCC3V3_SYS
1
2
3
4
5
6
7
8
9/2
10
11/2
15/2
13/2
14/2
15

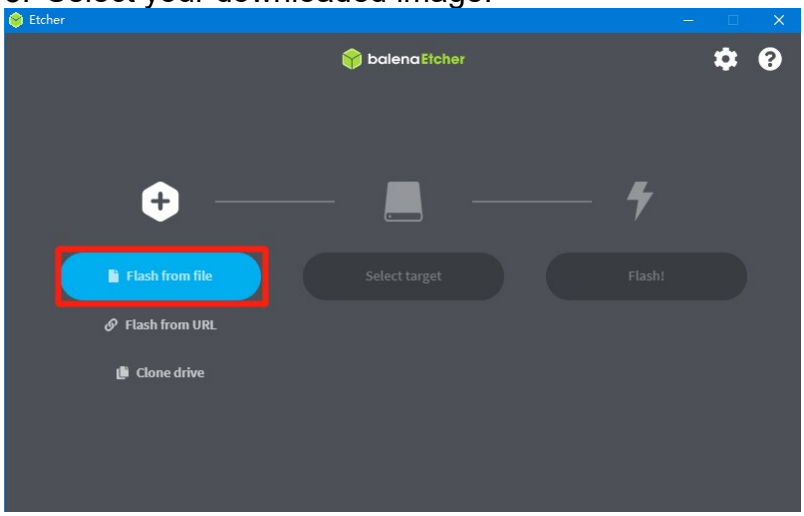
4. Flashing the System

4.1. Download the System Image

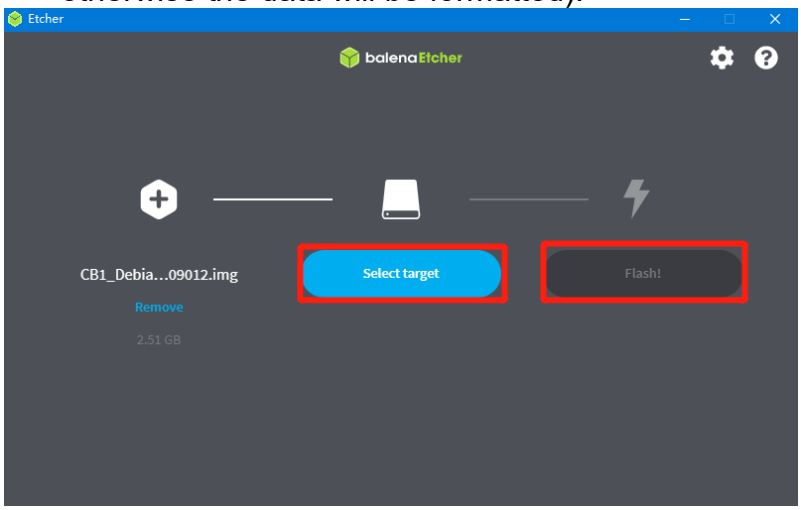
Only use the image we provide: https://github.com/bigtreetech/CB2/releases

4.2. Write System to MicroSD Card

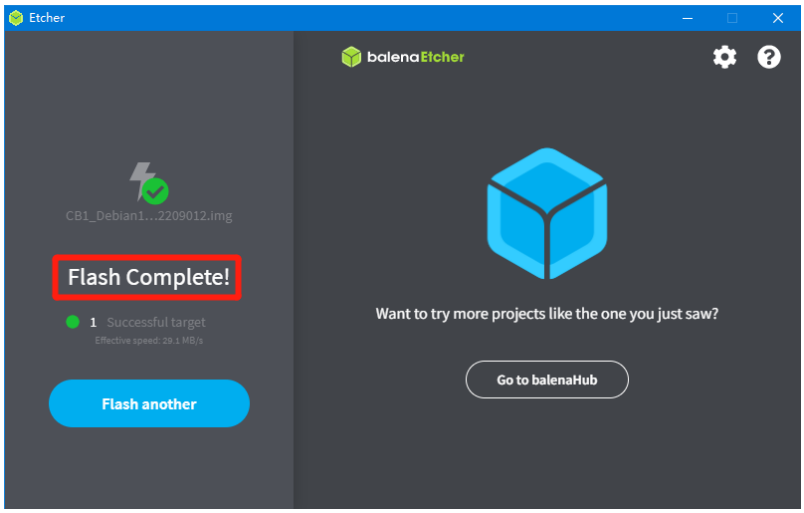
1. Download the balenaEtcher software from [https://www.balena.io/etcher/], install, and run it.
2. Insert the MicroSD card via a reader.
3. Select your downloaded image.



4. Select the MicroSD card and click "WRITE" (WRITE the image will format the MicroSD card. Be careful not to select the wrong storage device, otherwise the data will be formatted).



5. Wait for the process to complete.

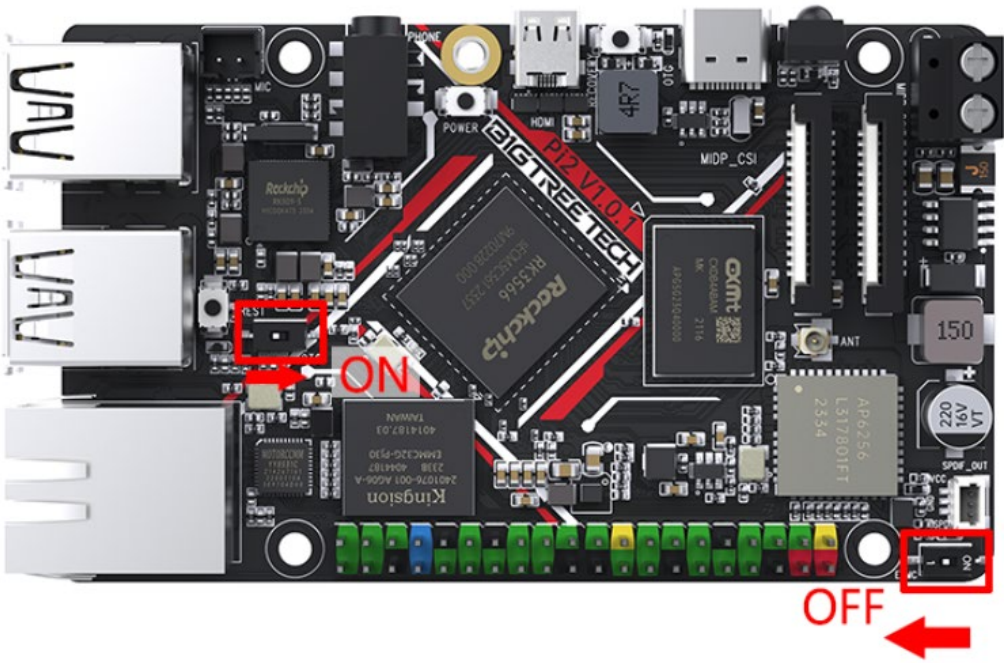


4.3. Writing System onto eMMC

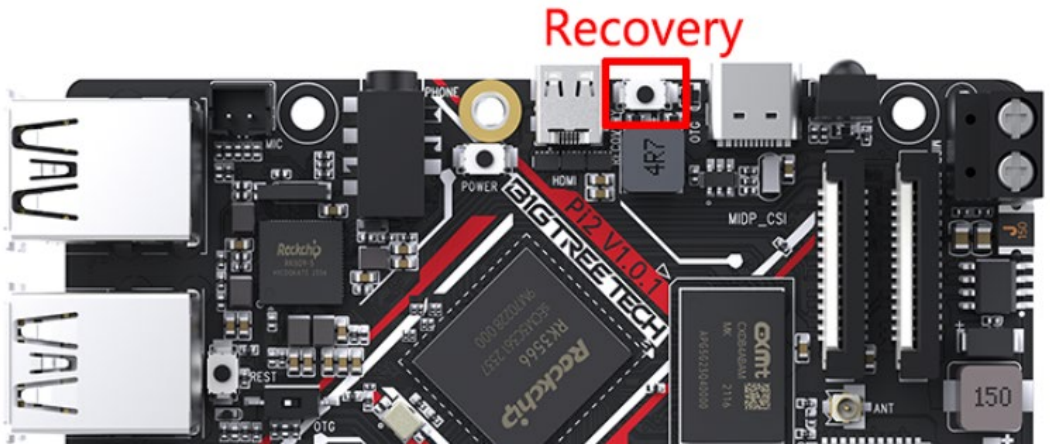
4.3.1. Using RKDevTool to Write the eMMC (Windows)

Download and unzip RKDevTool from BigTreeTech's GitHub repository (https://github.com/bigtreetech/CB2) to your computer. Make sure not to insert a MicroSD card.

1. As shown in the diagram below, toggle the USB OTG switch to the ON position. At this position, the USB OTG port is connected to a Type-C port. Set the eMMC switch to the OFF position, allowing the RK3566 to access the eMMC normally.

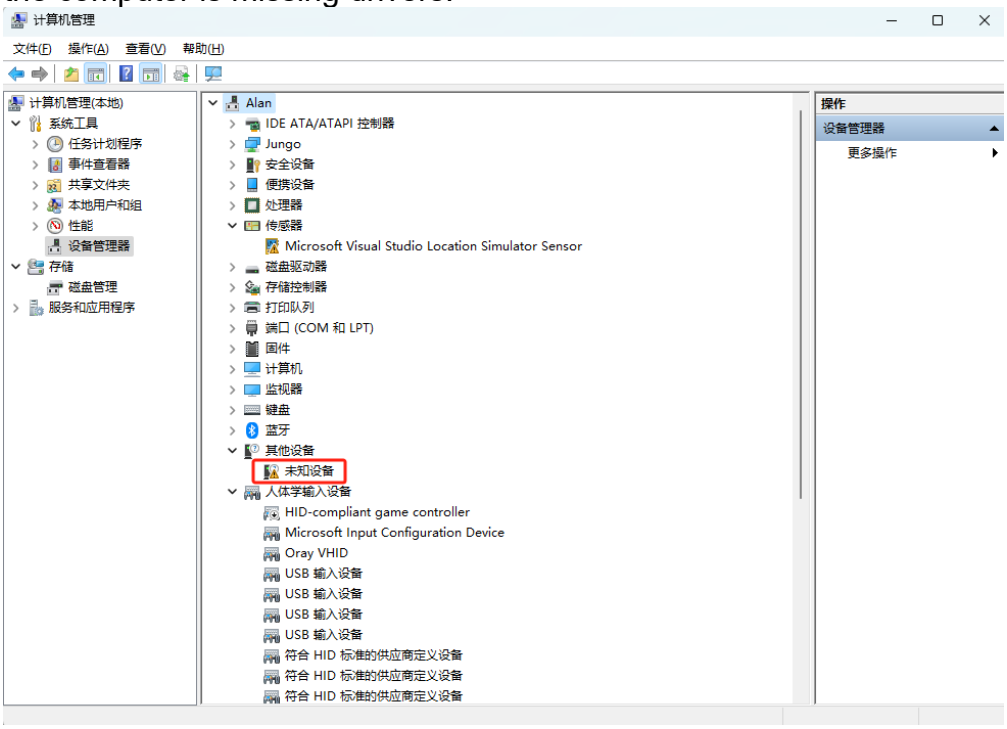


2. Hold down the “Recovery” button, then connect the BIGTREETECH Pi 2 to the computer using a Type-C cable. Power on and after 3 seconds, release the button.

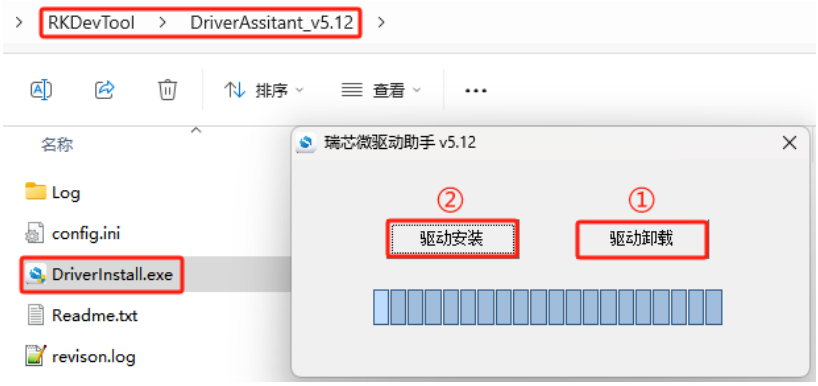


3. Install the driver:

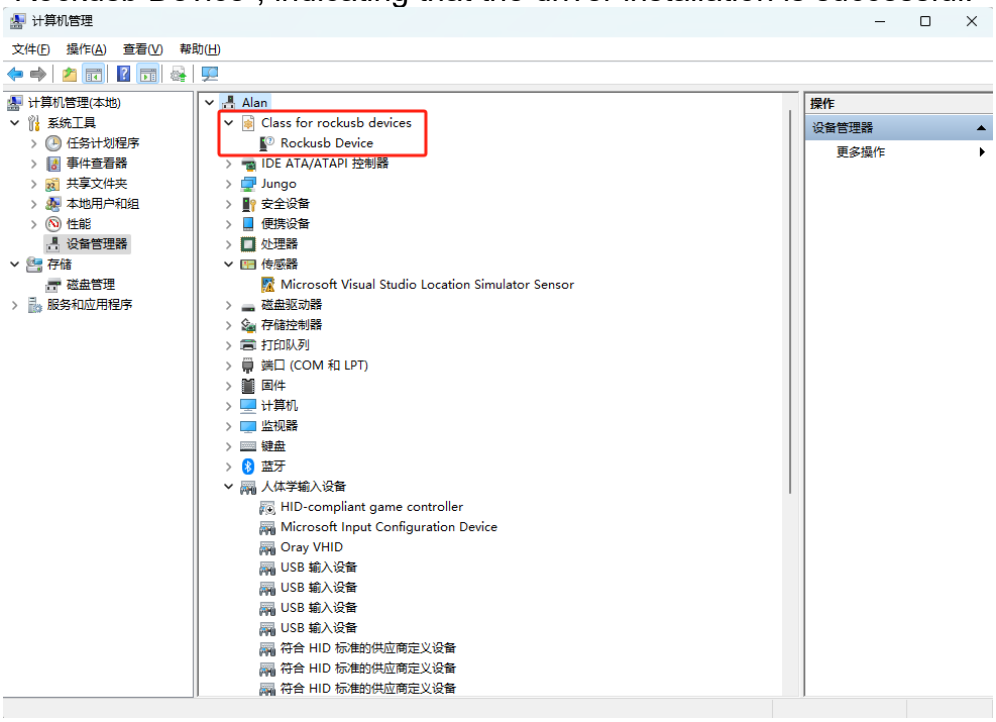
(1) In “Device Manager”, if you see “Unknown Device”, it indicates that the computer is missing drivers.



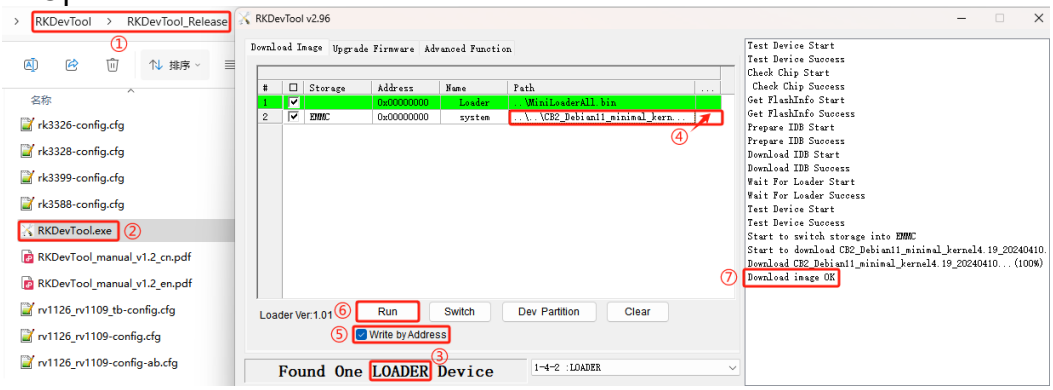
(2) Open the DriverAssistant tool in the downloaded RKDevTool folder, click “① Uninstall Driver”, then click “② Install Driver” to ensure that the latest version of the driver is installed.



(3) After the installation is complete, hold down the “Recovery” button, replug the Type-C cable. "Device Manager" should now recognize a “Rockusb Device”, indicating that the driver installation is successful.



4. Open the “RKDevTool” software:



Note: The parameters in the software are set by default as shown in the image. Normally, you only need to set the “④ actual path of the .img system”. If the parameters in your software do not match those in the image, manually adjust them to match.

#	□	Storage	Address	Name	Path
1	☑		0x00000000	Loader	..\MiniLoaderAll.bin
2	☑	EMMC	0x00000000	System	actual path of the .img system

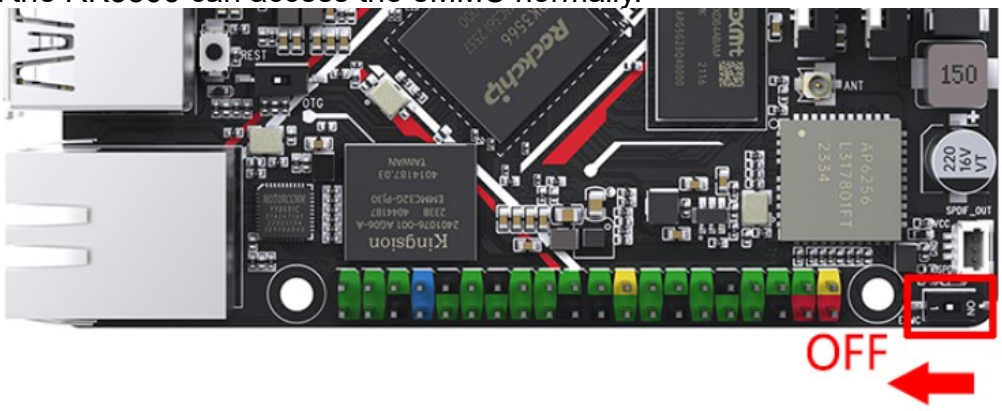
①Find the path where the downloaded RKDevTool is located.
②Open the RKDevTool tool.
③The software will recognize a “LOADER” or “MASKROOM” device.

④Select the system to be written (the OS image must be unzipped as a .img file beforehand; RKDevTool does not support directly writing compressed .xz files).
⑤Check “Write by Address”.
⑥Click “Run” to start writting the system.
⑦“Download image OK” indicates that the system has been successfully burned.

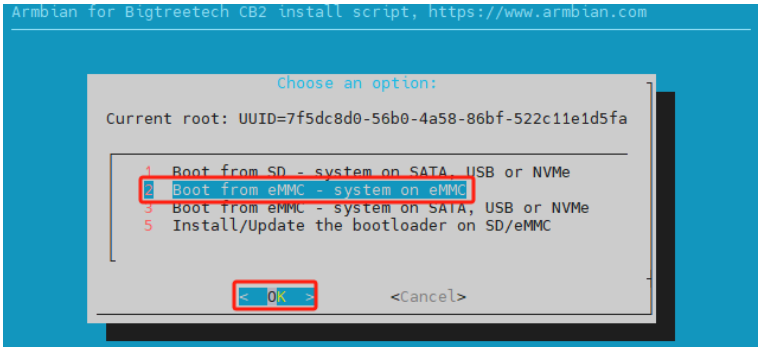
5. After writing is complete, toggle the USB OTG switch to the OFF position to boot normally. Note: Files on the eMMC cannot be accessed by the computer like those on a MicroSD card, so you cannot modify the system.cfg configuration file to set up the WiFi network. Instead, use an Ethernet cable or USB-to-UART connection to configure the terminal.

4.3.2. Writing System onto eMMC Using a MicroSD Card

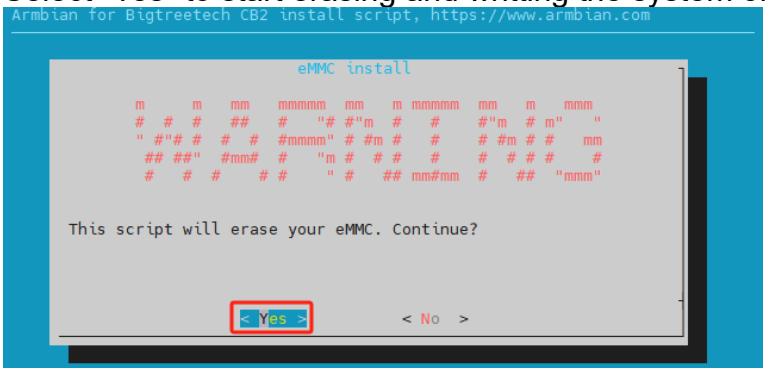
1. As shown in the figure, toggle the eMMC switch to the OFF position so that the RK3566 can access the eMMC normally.



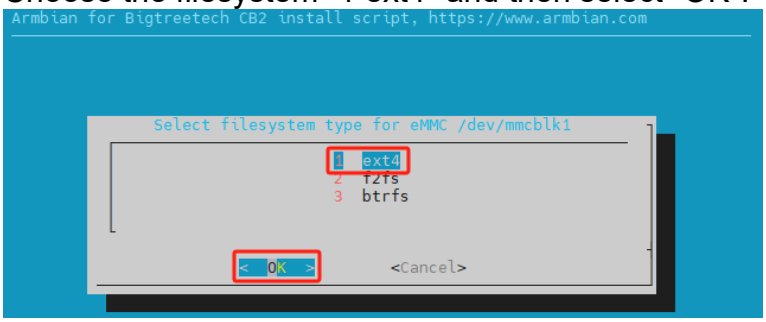
2. First, write the system onto a MicroSD card, then insert the MicroSD card into the board's card slot, and wait for the system to boot.
3. Connect to the system's terminal via Ethernet, WiFi, or USB to UART. Log in with the following credentials:
   login: biqu
   password: biqu
   (1) Execute the command sudo nand-sata-install. In the interface that pops up, select "2 Boot From eMMC - system on eMMC" and then select "OK".



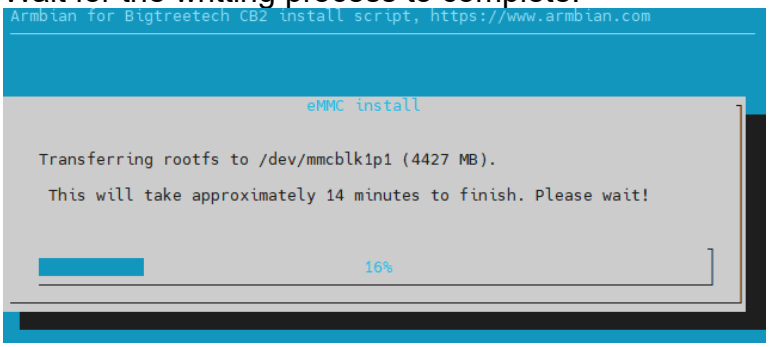
(2) Select "Yes" to start erasing and writting the system onto the eMMC.



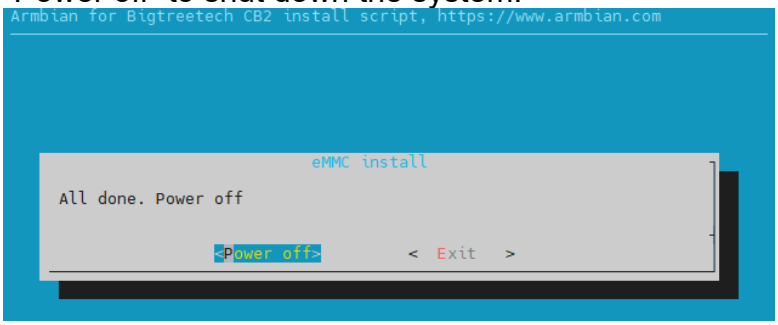
(3) Choose the filesystem "1 ext4" and then select "OK".



(4) Wait for the writting process to complete.



(5) Upon completion, you will be prompted whether to power off. Select “Power off” to shut down the system.



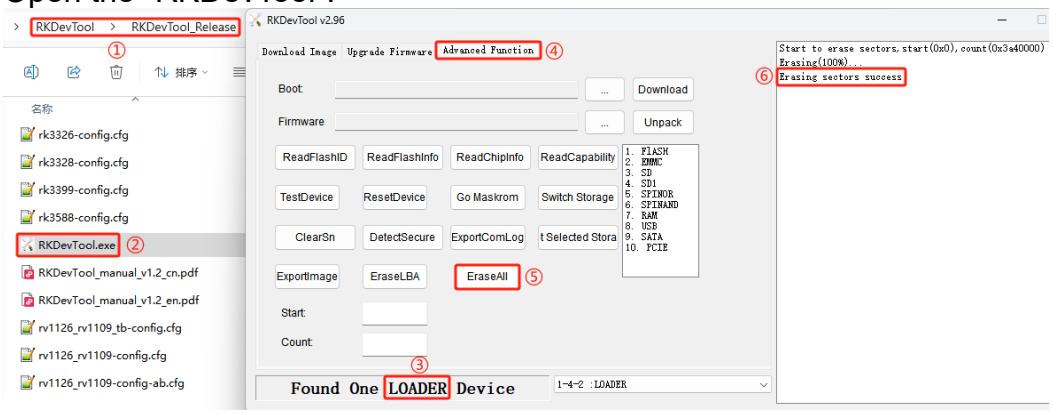
(6) After the system has powered down, disconnect the power supply, remove the MicroSD card, and then reconnect power. The system should now boot from the eMMC.

4.4. Erasing eMMC

When not using the eMMC and using a MicroSD card as the system card instead, it's best to erase the data on the eMMC to prevent the motherboard from mistakenly booting from it.

4.4.1. Using RKDevTool to Erase eMMC (Windows)

1. Refer to the steps in "4.3.1 Using RKDevTool to Write the eMMC (Windows)" to connect the motherboard to the computer.
2. Open the "RKDevTool".



①Find the path where the downloaded RKDevTool is located.
②Open the RKDevTool.
③The software will recognize a "LOADER" device. If it recognizes "MASKROOM," it indicates there is no data in the eMMC, hence no erase operation is necessary.
④Click "Advanced Function."
⑤Click "EraseAll" to begin erasing data from the eMMC.
⑥"Erasing sectors success" indicates the erasure is complete.

4.4.2. Erasing eMMC After Booting from MicroSD Card

1. Refer to the steps in "4.3.2 4.3.2. Writing System onto eMMC Using a MicroSD Card" and log into the system terminal.
2. Run the command sudo mkfs /dev/mmcblk1 and then enter "y" to confirm.

```
biqu@BTT-CB2:~$ sudo mkfs /dev/mmcblk1
mke2fs 1.46.2 (28-Feb-2021)
/dev/mmcblk1 contains a ext2 file system
        created on Wed Apr 24 06:30:21 2024
Proceed anyway? (y,N) y
Discarding device blocks: done
Creating filesystem with 7634944 4k blocks and 1908736 inodes
Filesystem UUID: 51dbd34e-8aef-4f29-9f98-e535341ed141
Superblock backups stored on blocks:
        32768, 98304, 163840, 229376, 294912, 819200, 884736, 1605632, 2654208,
        4096000

Allocating group tables: done
Writing inode tables: done
Writing superblocks and filesystem accounting information: done

biqu@BTT-CB2:~$
```

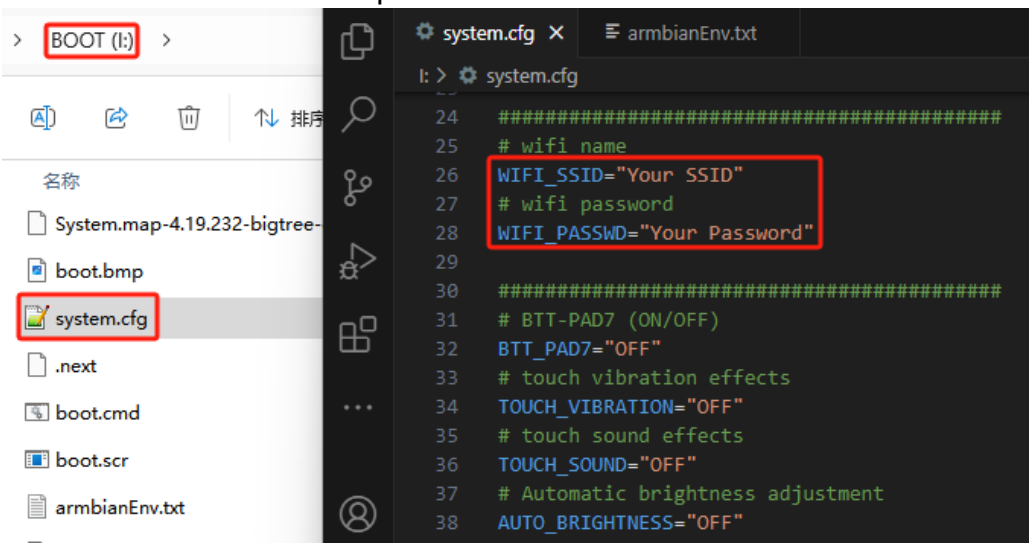
5. System Configuration

5.1. Using Ethernet

Ethernet is plug-and-play and requires no additional setup.

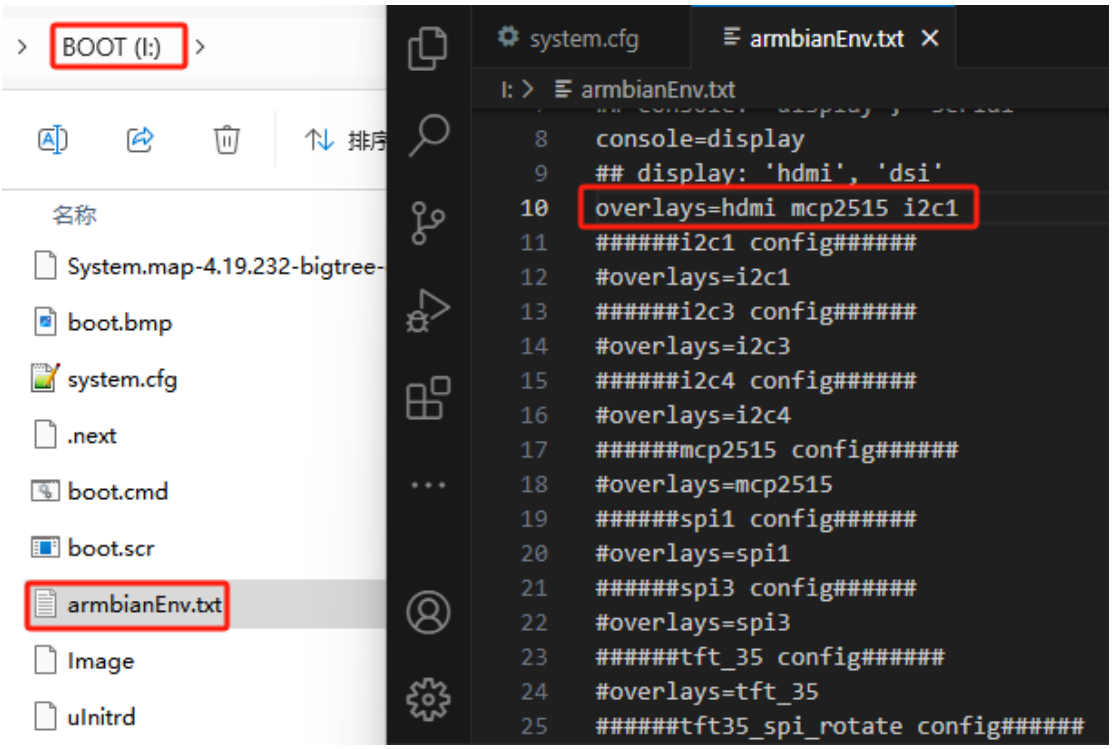
5.2. Setting Up WiFi

After the system image has been written, the MicroSD card will have a FAT32 partition recognized by the computer. In this partition, there is a "system.cfg" file. Open it and replace "Your SSID" with your actual WiFi name and "Your Password" with the actual password.



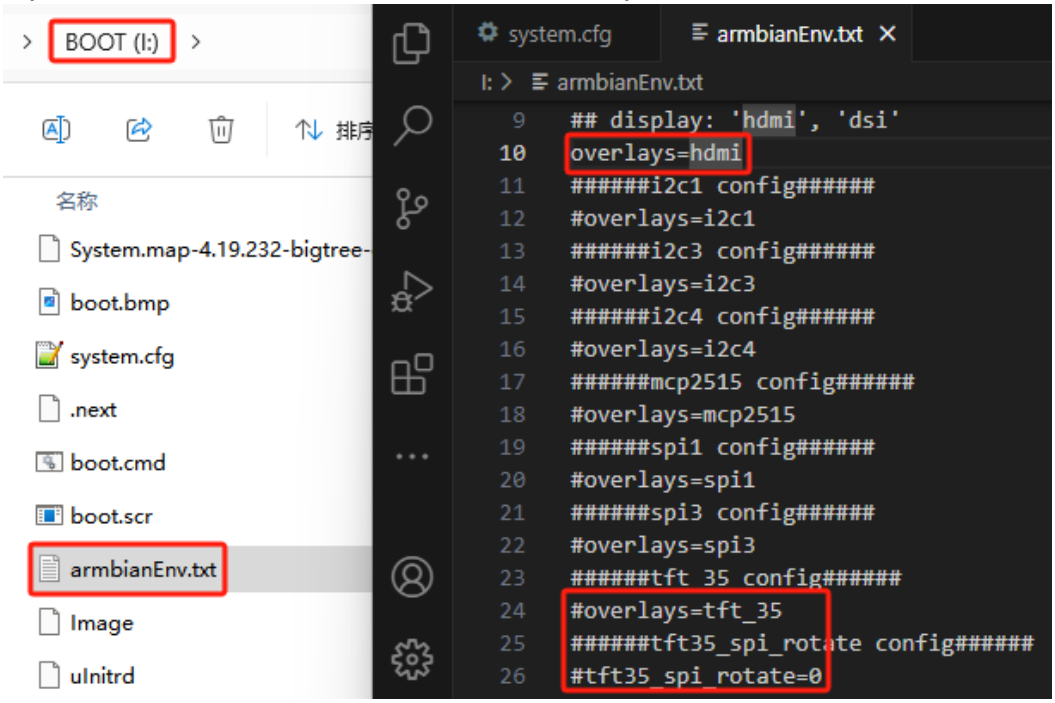
5.3. Configuring Overlays

Open the "armbianEnv.txt" file in the BOOT partition and set the values for overlays. The configuration file supports only one line of overlays at a time; if multiple overlays are enabled, only the last line will take effect. If you need multiple overlays, place the contents of multiple configurations on the same line separated by a space. For example, if you need to use a DSI screen, MCP2515 SPI to CAN module, and I2C1 simultaneously:
overlays=dsi mcp2515 i2c1



5.4. Configuring the Display

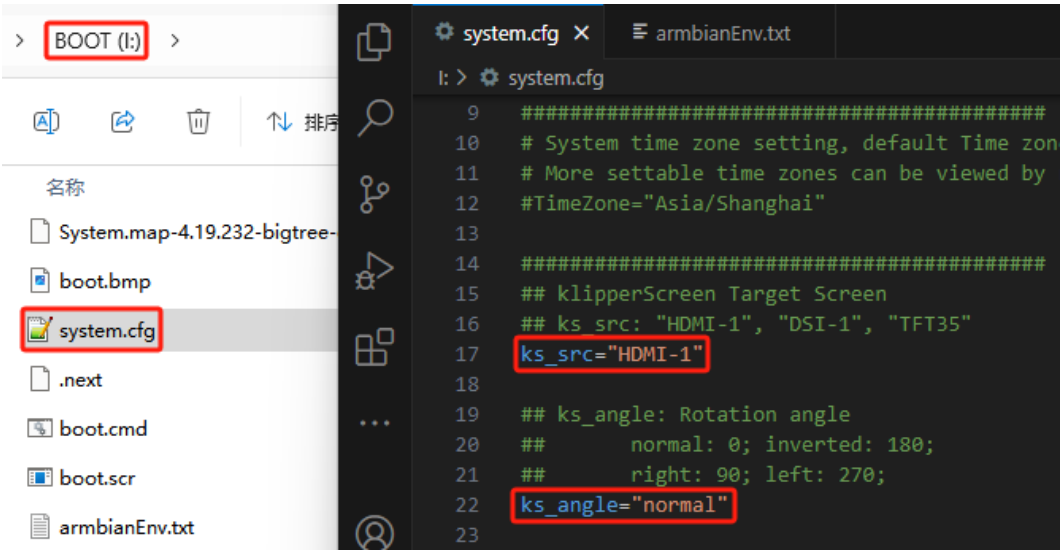
1. Open the "armbianEnv.txt" file in the BOOT partition.



2. The default overlay is set to "hdmi," meaning the system uses an HDMI screen by default. This can be changed to match the actual screen being used, such as:
   - "hdmi": HDMI screen
   - "dsi": DSI screen
   - "tft_35": SPI Screen

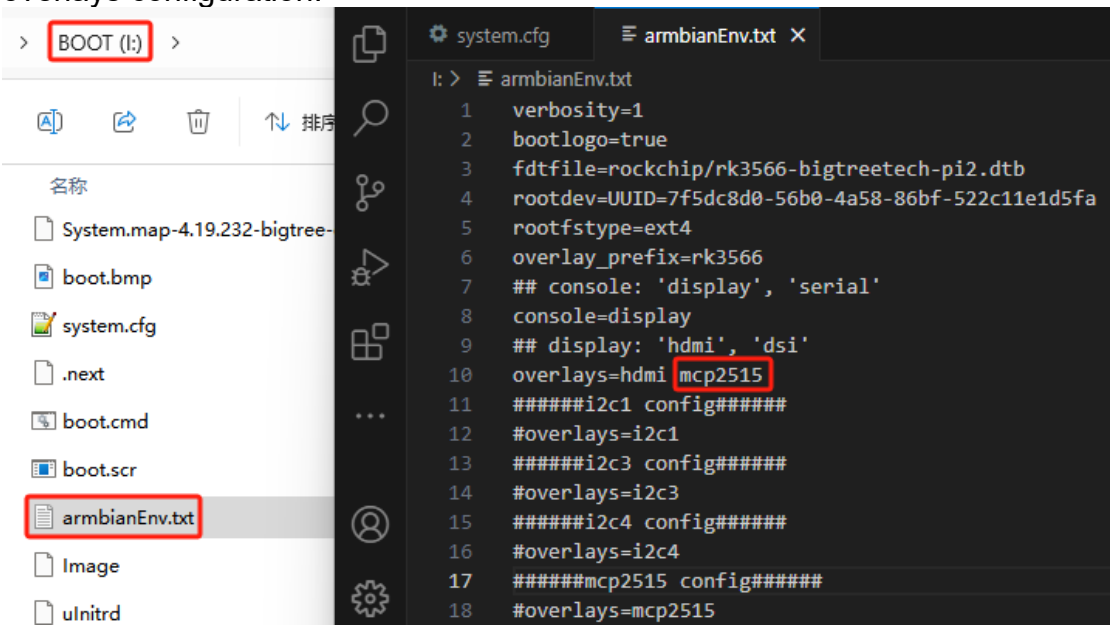
For "tft_35", there is also a "tft35_spi_rotate" parameter for system-level screen rotation, with default "0" meaning no rotation, other options include "90", "180", "270".
Note: Only one screen type can be used at a time.

3. To configure KlipperScreen, open the `system.cfg` file in the BOOT partition. Set the screen type with the parameter `ks_src`, and the rotation angle with `ks_angle`.



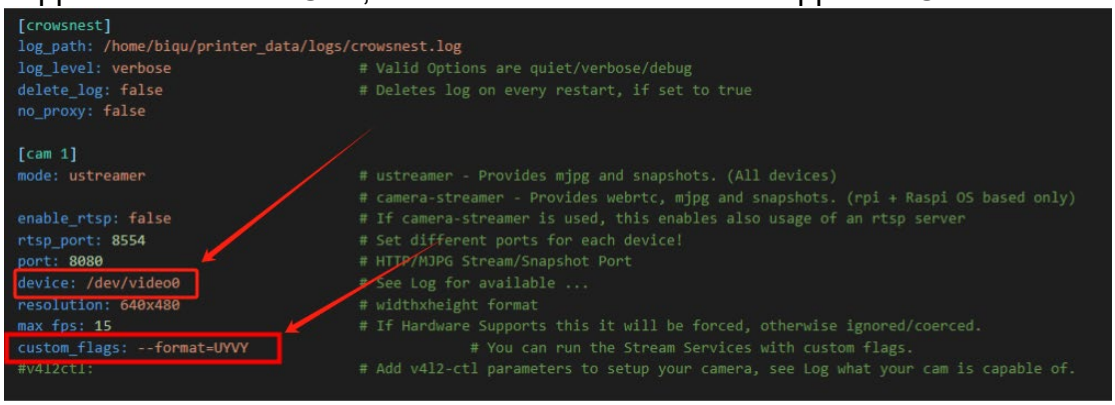
5.5. Using SPI to CAN

Open the "armbianEnv.txt" file in the BOOT partition and add "mcp2515" to the overlays configuration.



5.6. Using CSI Camera and Crowsnest Configuration

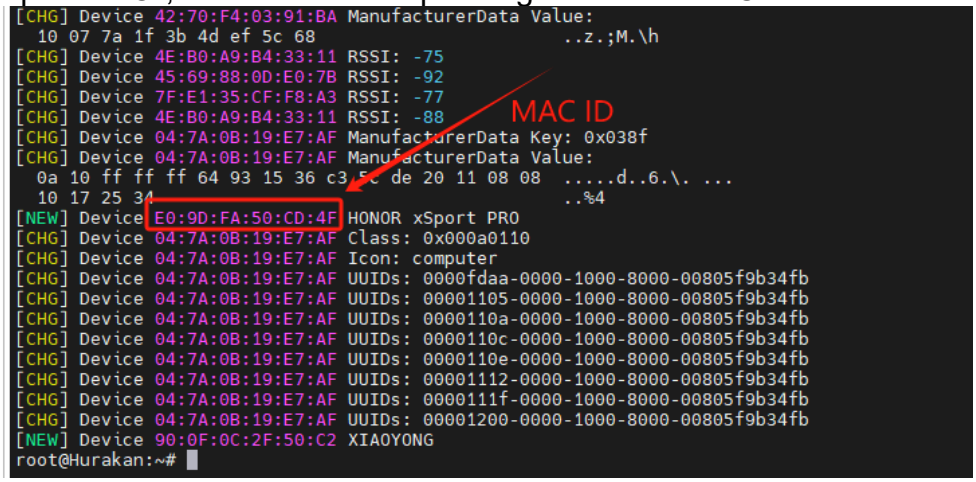
For both RPi v1.3 ov5647 and RPi v2 imx219 cameras, no specific configuration in "armbianEnv.txt" is required as they are plug-and-play. "crowsnest.conf" file configuration is as follows:
device: /dev/video0 # The CSI camera node is fixed as video0
custom_flags: --format=UYVY # The current system's CSI camera does not support the default YUYV, so it needs to be set to the supported UYVY format.



5.7. Using Bluetooth

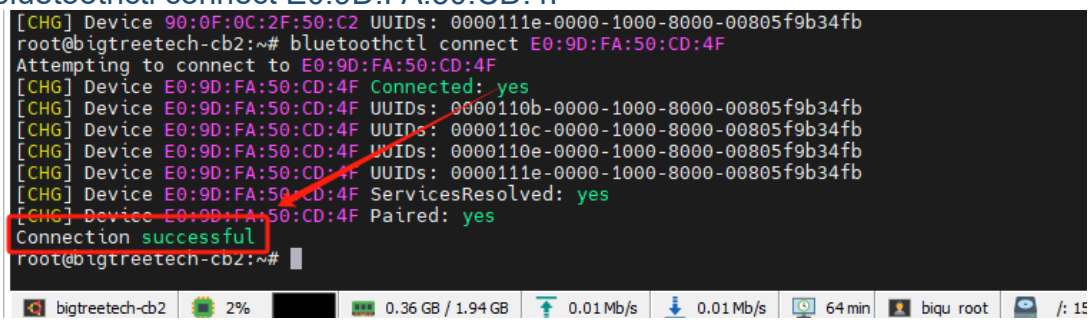
1. To scan for Bluetooth devices, enter the following command, and a list of Bluetooth devices will appear as shown below:
bluetoothctl --timeout 15 scan on

2. Find your Bluetooth device, for example, if your device name is "HONOR xSport PRO", locate the corresponding Bluetooth MAC ID as shown below.

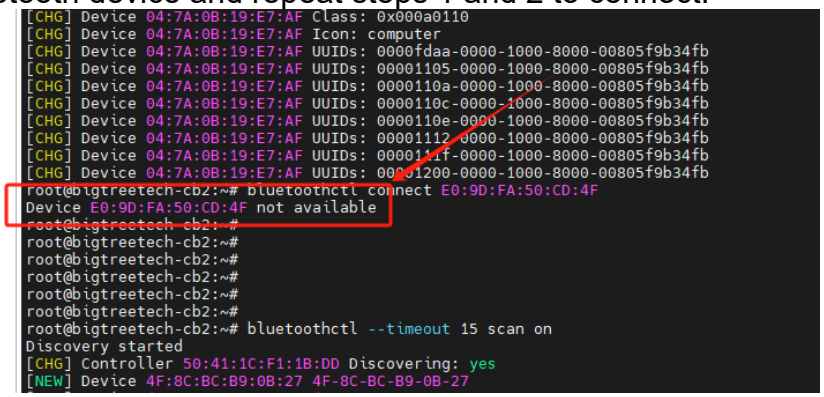


3. To connect to a Bluetooth device, enter the following command, connection success is shown as below

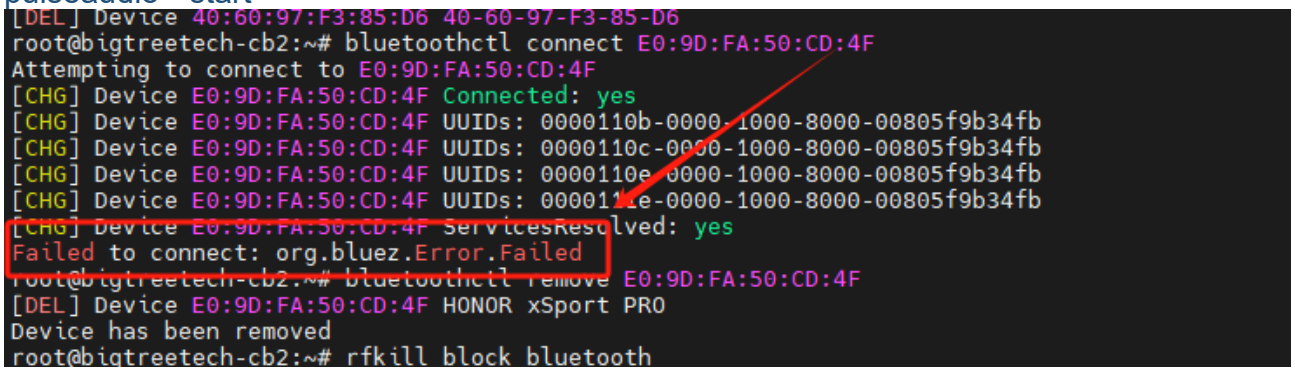
bluetoothctl connect E0:9D:FA:50:CD:4F



(1) If there's an issue while connecting, as shown below, please restart the Bluetooth device and repeat steps 1 and 2 to connect.



(2) If there's an issue while connecting, as shown below, please enter the following commands and then repeat steps 1 and 2:
bluetoothctl remove E0:9D:FA:50:CD:4F (Your Bluetooth device's corresponding MAC ID)
rfkill block bluetooth
sleep 3s
rfkill unblock bluetooth
pulseaudio -k
pulseaudio --start

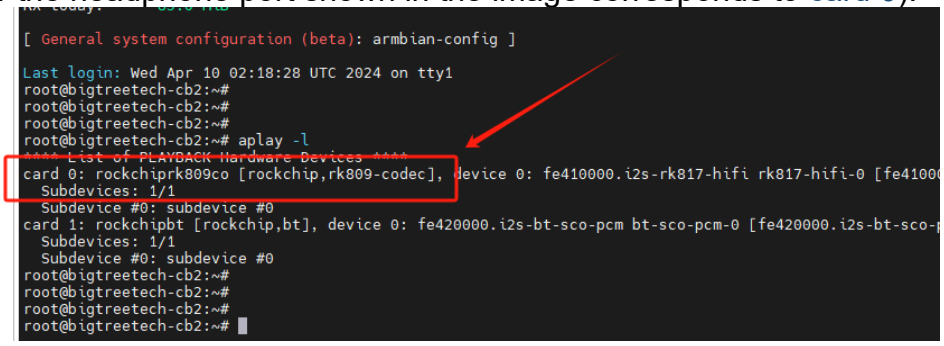


4. If you exit voice playback during the use of Bluetooth and cannot reuse it, manually delete the corresponding playback process. Use the ps command to view the process number, then use kill -9 process_number to delete the corresponding playback process.

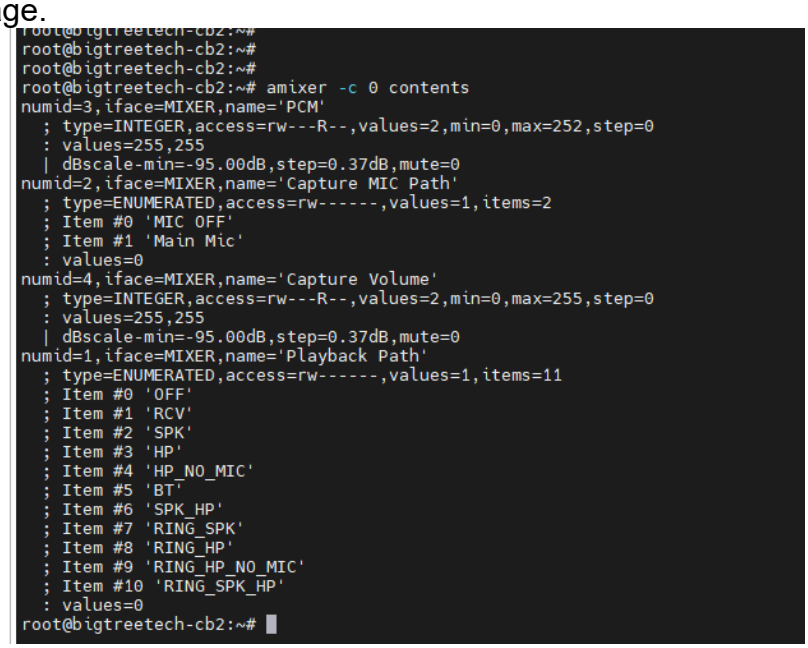
```
biqu@bigtreetech-cb2:~$ ps
    PID TTY          TIME CMD
   2094 pts/0    00:00:00 bash
   2270 pts/0    00:00:00 aplay
   2347 pts/0    00:00:00 ps
biqu@bigtreetech-cb2:~$ kill -9 2270
```

5.8. Setting up 3.5mm Headphones Port

1. Enter the command: aplay -l
Check for the corresponding sound card as shown in the image (the sound card for the headphone port shown in the image corresponds to card 0).

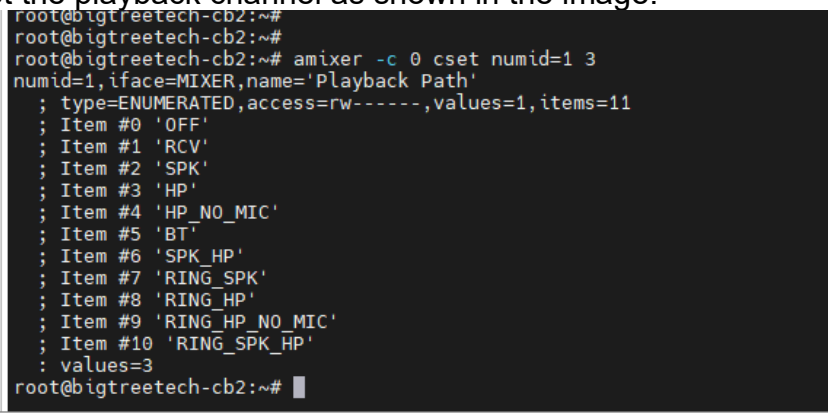


2. Enter the command:
amixer -c 0 contents (0 represents the card 0 found in the previous aplay -l command)
Check the settings for playback and recording channels as shown in the image.



3. Enter the command:
amixer -c 0 cset numid=1 3

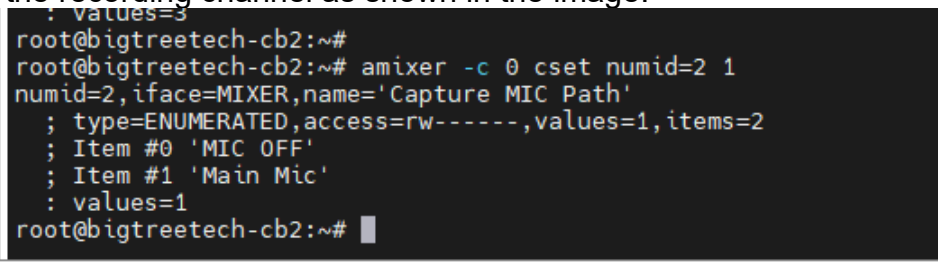
Set the playback channel as shown in the image.



4. Enter the command:

amixer -c 0 cset numid=2 1

Set the recording channel as shown in the image.



5. Enter the following command to play audio, with the audio file directory xxx and the audio file name xxxxx.wav:

aplay -D plughw:0,0 /xxx/xxxxx.wav

6. Enter the following command to record (where 10 represents recording for 10 seconds), storing the recording in directory xxx, file name xxxx.wav:

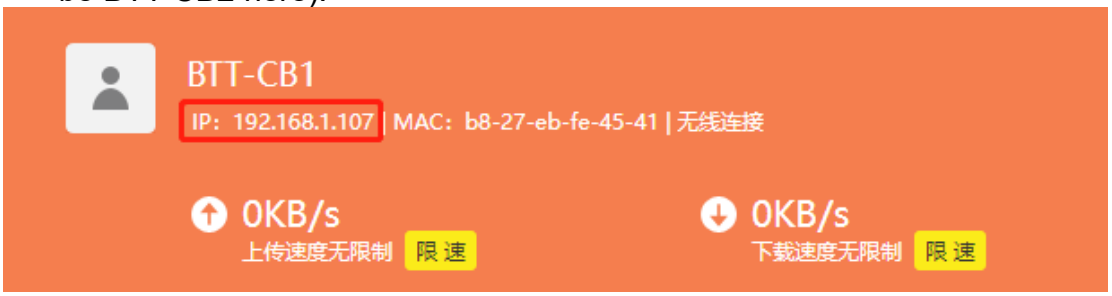
sudo arecord -Dhw:0,0 -d 10 -f cd -r 44100 -c 2 -t wav /xxx/xxxx.wav

7. Enter the following command to play the recording:

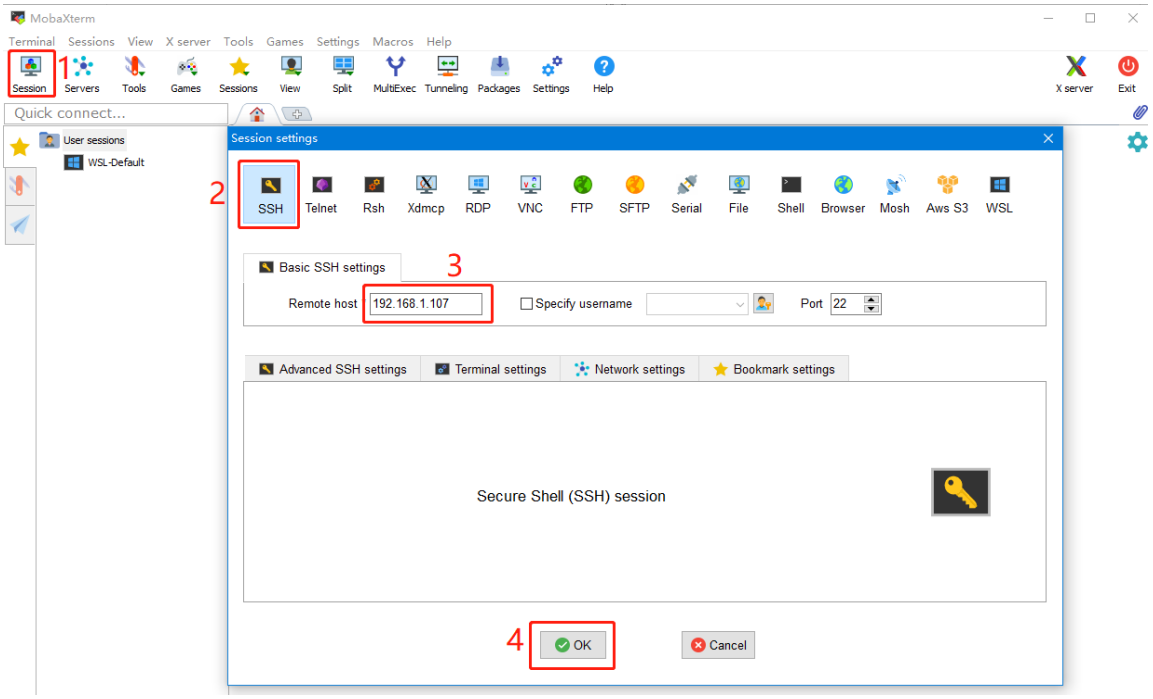
aplay -D plughw:0,0 /xxx/xxxx.wav

6. SSH Connect to Device

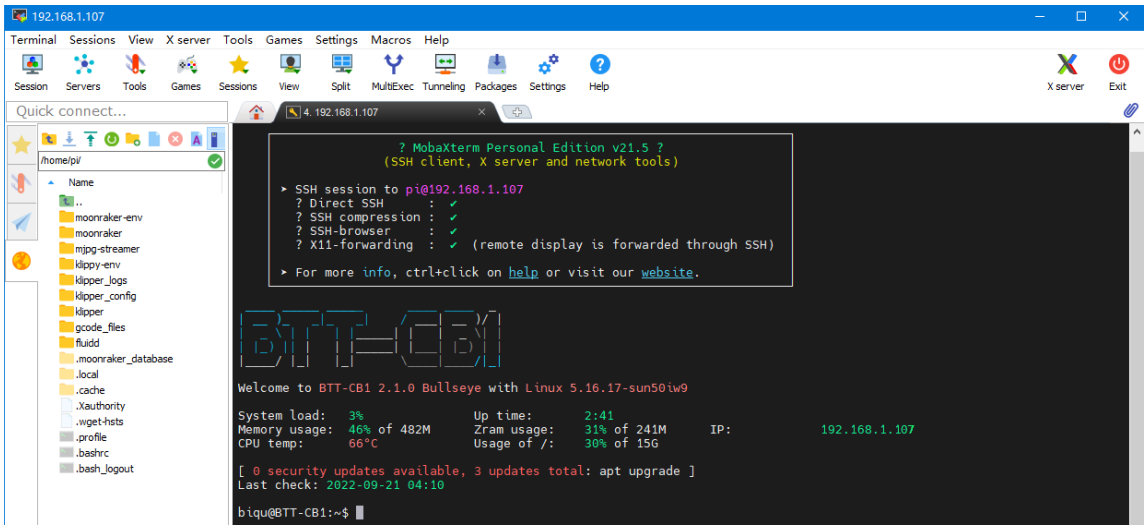
1. Install the SSH software Mobaxterm: https://mobaxterm.mobatek.net/download-home-edition.html
2. After powering on, wait for the system to boot, which typically takes about 1 to 2 minutes.
3. Once the device is connected to WiFi or an Ethernet cable is plugged in, it will automatically be assigned an IP address.
4. Access the router management interface to find the device's IP (it should be BTT-CB2 here).



5. Open Mobaxterm and click "Session", and click "SSH", inset the device IP into Remote host and click "OK" (Note: your computer and the device needs to be in the same network).



6. Login as: biqu password: biqu


192.168.1.107
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help X server Exit
Quick connect...
4. 192.168.1.107
/home/pi/
Name
moonraker-env
moonraker
mjpg-streamer
klippy-env
klipper_logs
klipper_config
klipper
gcode_files
fluidd
.moonraker_database
.local
.cache
.Xauthority
.wget-hsts
.profile
.bashrc
.bash_logout
? MobaXterm Personal Edition v21.5 ?
(SSH client, X server and network tools)
SSH session to pi@192.168.1.107
? Direct SSH : ✓
? SSH compression : ✓
? SSH-browser : ✓
? X11-forwarding : ✓ (remote display is forwarded through SSH)
For more info, ctrl+click on help or visit our website.
Welcome to BTT-CB1 2.1.0 Bullseye with Linux 5.16.17-sun50iw9
System load: 3% Up time: 2:41
Memory usage: 46% of 482M Zram usage: 31% of 241M IP: 192.168.1.107
CPU temp: 66°C Usage of /: 30% of 15G
[0 security updates available, 3 updates total: apt upgrade]
Last check: 2022-09-21 04:10
biqu@BTT-CB1:~$

7. Precautions

1. About 10 seconds after powering on, the system enters the kernel phase. At this time, the blue light stays on, and the green light flashes continuously, indicating that the system is running normally.

2. Root administrator:
   Login: root
   Password: root

   BIQU user:
   Login: biqu
   Password: biqu

3. The PCIe M.2 interface does not support hot-plugging; the solid-state drive must be connected in advance for the device to be recognized.

4. When booting from eMMC, do not insert a MicroSD card. When booting from a MicroSD card, it is necessary to erase the data in the eMMC.

If you need further resources for this product, you can find them at [GitHub](https://github.com/bigtreetech/). If you cannot find what you need, you may contact our after-sales support(service005@biqu3d.com).

If you encounter any other problems during use or have suggestions or feedback, please contact us. Thank you for choosing BIGTREETECH products.