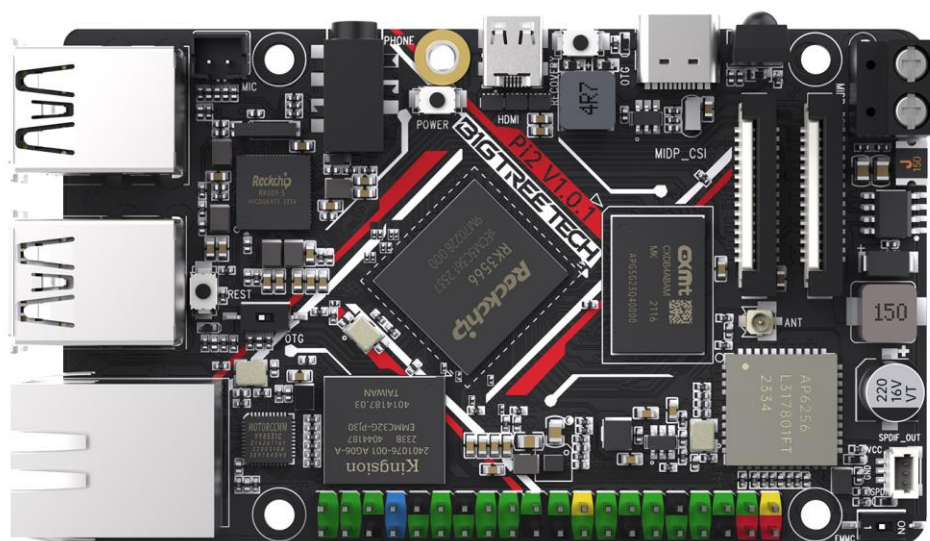


BIGTREE TECH

Pi2

用户手册



目录

目录	2
修订历史	4
一、产品简介	5
1.1 产品特点	5
1.2 产品参数	5
1.3 尺寸图	6
二、外设接口	7
2.1 接线图	7
三、接口介绍	8
3.1 供电方式	8
3.2 40-pin GPIO	8
3.3 UPS POWER	9
3.4 SPI FLASH	9
3.5 SPDIF OUT	10
3.6 eMMC-EN	10
3.7 OTG	11
3.8 MIC IN	11
3.9 DSI	12
3.10 CSI	13
四、烧录系统	14
4.1 下载系统镜像	14
4.2 烧录系统到 MicroSD 卡	14
4.3 烧录系统到 eMMC	15
4.3.1 使用 RKDevTool 烧录系统到 eMMC (Windows)	15
4.3.2 使用 MicroSD 卡烧录系统到 eMMC	18
4.4 擦除 eMMC	19
4.4.1 使用 UMS 擦除 eMMC (Windows)	19
4.4.2 使用 RKDevTool 擦除 eMMC (Windows)	20

4.4.3 从 MicroSD 卡启动系统后擦除 eMMC.....	20
五、系统配置	21
5.1 使用网线	21
5.2 设置 WiFi.....	21
5.3 配置 overlays.....	21
5.4 配置显示屏	22
5.5 SPI 转 CAN 的使用.....	23
5.6 CSI 相机使用及 crowsnest 配置.....	23
5.7 蓝牙的使用.....	24
5.8 3.5mm 圆口耳机设置.....	26
六、SSH 连接设备.....	28
七、注意事项	29

修订历史

版本	日期	修改说明
v1.00	2024/04/24	初稿
v1.01	2024/05/28	增加 minimal 版本系统的账户信息
v1.02	2024/06/05	增加 40 pin GPIO 表格及新的计算方式
v1.03	2025/03/26	增加 40-Pin GPIO 复用功能描述 纠正 UPS Power 部分的说明
v1.04	2025/04/14	V3.0.1 版本系统，u-boot loader 模式变更为 ums 模式
v1.05	2025/04/21	PI2/CB2 的 40-Pin GPIO, Pin32/Pin33 顺序不一样

一、产品简介

BIGTREETECH Pi 2 采用性能更优的四核 A55-RK3566 芯片，接口功能丰富多彩，内置 eMMC5.1，板载支持 2.4G/5G 双模，WIFI 传输速度高达 433.3Mbps，还支持蓝牙 BT5.2 版本，与树莓派相同的安装孔位置，安装使用起来方便快捷。

1.1 产品特点

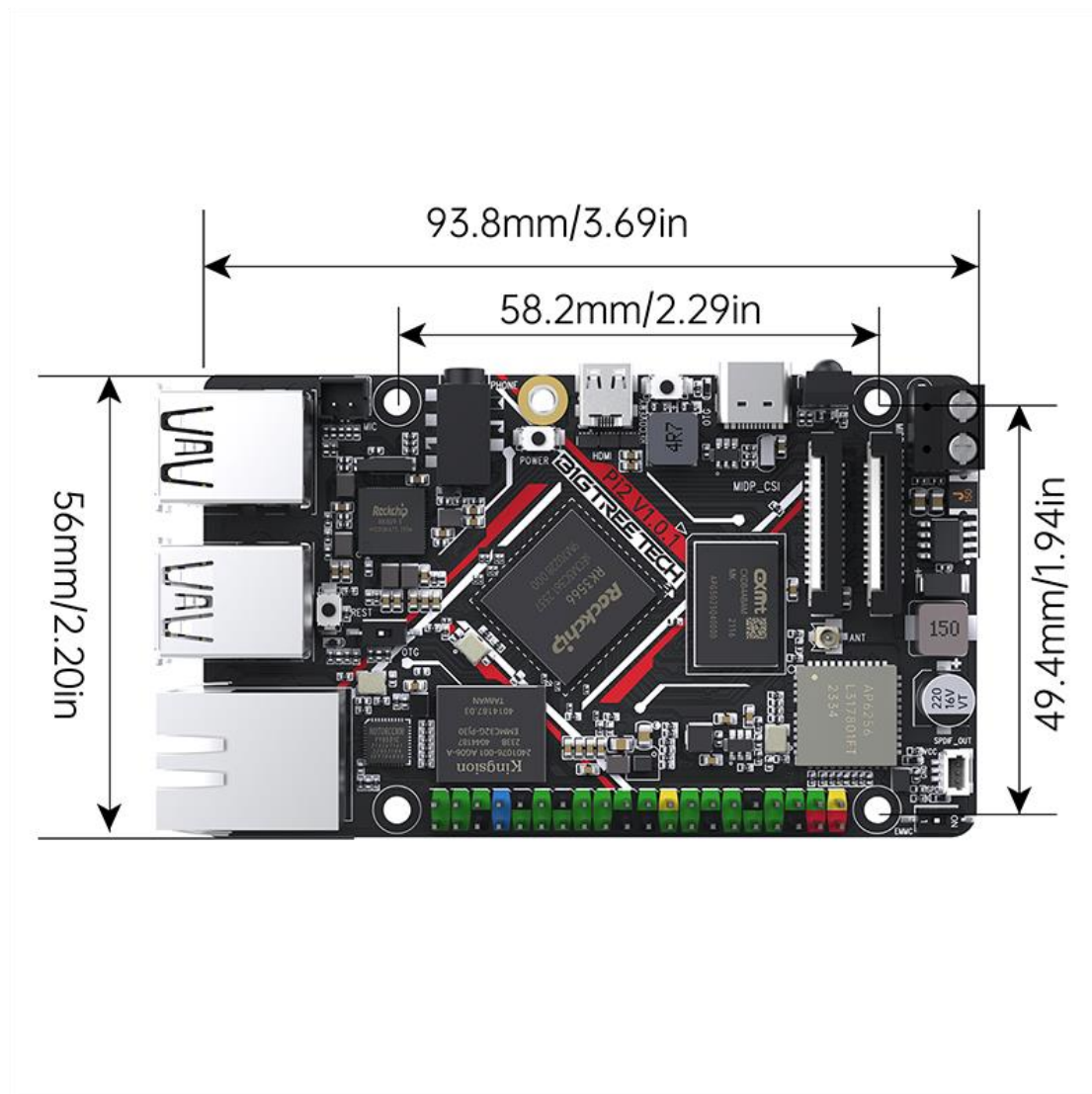
- CPU: 瑞芯微 RK3566, 四核 Cortex-A55 @1.8GHz
- GPU: Mali-G52 1-Core-2EE
- NPU: 0.8 TOPS NPU
- RAM: 2GB LPDDR4 (其它可以定制 1GB/2GB/4GB/8GB)
- 板载 eMMC 32G (其它可以定制 8GB/32GB/64GB/128GB...)
- MIPI DSI 显示支持 (320P-1080P 60HZ)
- SPI FLASH: 可以定制 W25Q256JWEIQ
- 摄像头 2-Lane MIPI CSI2 (320P-1080P 60HZ)
- 3 路 USB2.0 端口 1 路 USB3.0
- PCIe PCIe 2.1 1x1 Lane (支持型号: M.2 2242 4PIN+5PIN)
- 支持 TF 卡 (SDIO2.0)
- 千兆以太网/WiFi (433.3Mbps) /BT5.2
- Audio 接口, 3.5mm 兼容 MIC 输入
- 电容麦输入接口
- 40Pin GPIO
- HDMI2.0 OUT(480P-4k 60hz)
- 板载红外接收头(38khz)
- 与树莓派相同的安装孔位置
- 24VDC 输入接口

1.2 产品参数

1. 外观尺寸: 93.8mm*56mm
2. 安装尺寸: 58.2mm x 49.4mm

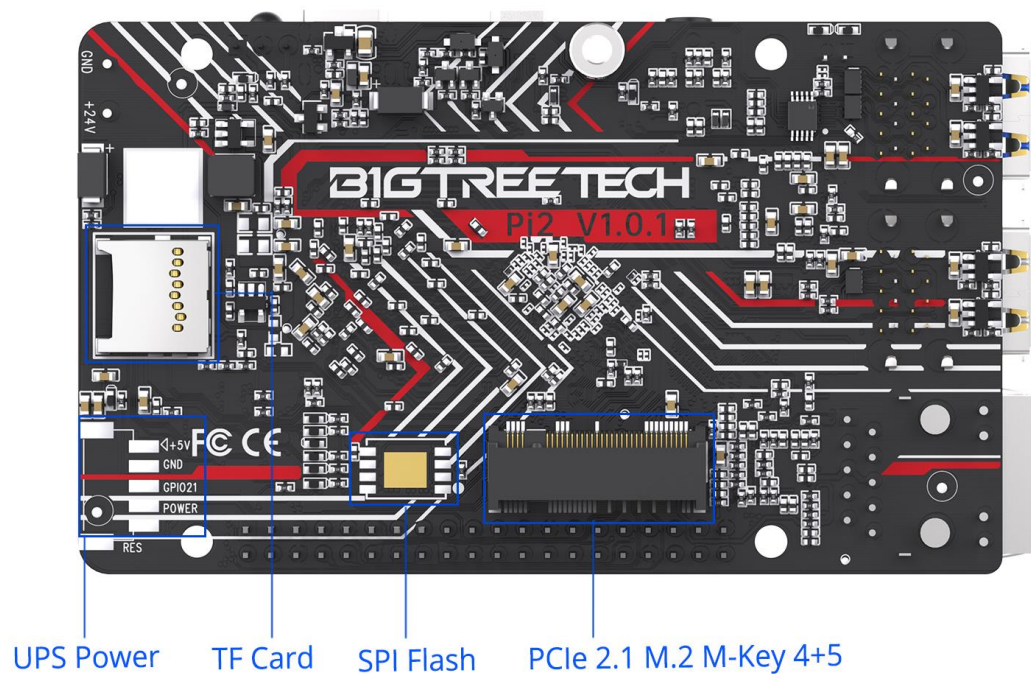
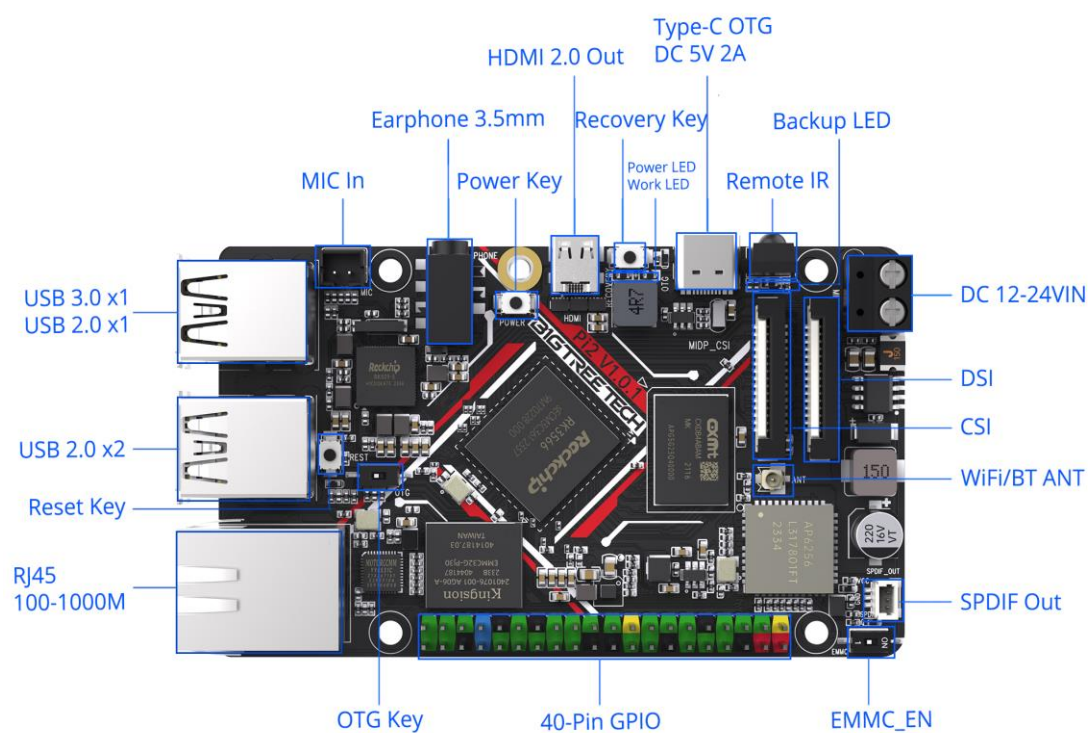
3. Type-C 输入电压: DC 5V \pm 5%/2A
4. 输出电压: 3.3V \pm 2%/100mA
5. WiFi: 2.4G/5G, 802.11 ac/a/b/g/n/ 无线标准
6. BT:5.2

1.3 尺寸图



二、外设接口

2.1 接线图



三、接口介绍

3.1 供电方式

输入：

- USB-C：USB 5V/2A
- 接线端子：DC 12-24V



3.2 40-pin GPIO

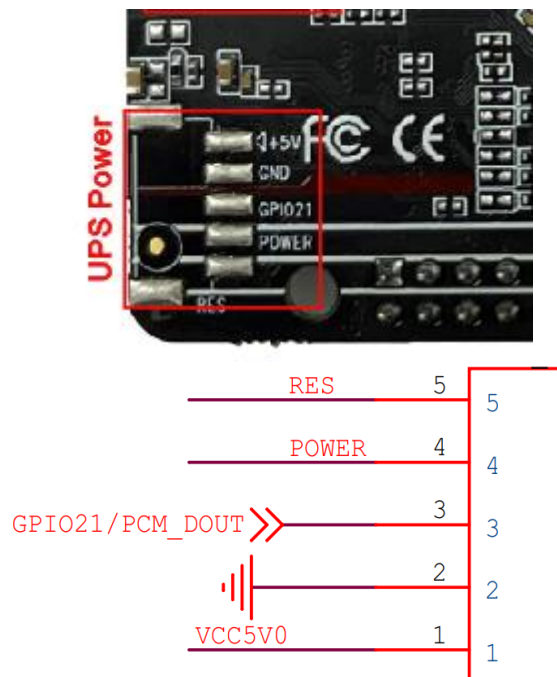
PI2/CB2 40-Pin GPIO											
function5	function4	function3	function2	function1	Pin	function1	function2	function3	function4	function5	
				3.3V	1	5V					
			I2C4_SDA_M0	GPIO4_B2(gpiochip4/gpio10)	3	5V					
			I2C4_SCL_M0	GPIO4_B3(gpiochip4/gpio11)	5	GND					
				GPIO3_A1(gpiochip3/gpio1)	7	GPIO0_D1(gpiochip0/gpio25)	UART2_TX_M0				
				SPI1_CS1	8	GPIO0_D0(gpiochip0/gpio24)	UART2_RX_M0				
				GND	9	GPIO0_B0(gpiochip0/gpio8)					
UART0_CTSn	PWM0_M1			GPIO0_C7(gpiochip0/gpio23)	11	SPI1_CS2					
				GPIO1_A0(gpiochip1/gpio0)	13	GND					
				GPIO1_A1(gpiochip1/gpio1)	15	GPIO4_C5(gpiochip4/gpio22)	PWM13_M1			UART9_RX_M1	
				3.3V	17	SPI1_CS0				UART7_RX_M2	
				GPIO3_C1(gpiochip3/gpio17)	19	GPIO4_A3(gpiochip4/gpio3)					
UART5_TX_M1	SPI1_MISO_M1			GPIO3_C2(gpiochip3/gpio18)	21	GND					
UART5_RX_M1	SPI1_CLK_M1			GPIO3_C3(gpiochip3/gpio19)	23	GPIO0_C4(gpiochip0/gpio20)	PWM5		UART0_RTSn		
				GND	25	GPIO4_A2(gpiochip4/gpio2)				UART7_TX_M2	
					27	GPIO0_A6(gpiochip0/gpio6)					
				I2C1_SDA	29	GPIO0_B3(gpiochip0/gpio11)	I2C1_SCL				
				GPIO3_D6(gpiochip3/gpio30)	31	GND					
				GPIO3_D7(gpiochip3/gpio31)	33	PI2 GPIO0_C0(gpiochip0/gpio16)	PWM1_M0		UART0_RX		
				GPIO0_C1(gpiochip0/gpio17)	35	CB2 GPIO0_C1(gpiochip0/gpio17)	PWM2_M0		UART0_TX		
UART0_TX	PWM2_M0			GPIO0_C0(gpiochip0/gpio16)	37	GND					
UART0_RX	PWM1_M0			GPIO4_C5(gpiochip4/gpio21)	39	GPIO0_A0(gpiochip0/gpio0)					
UART9_TX_M1	SPI3_MISO_M1	PWM12_M1		GPIO0_C3(gpiochip0/gpio19)	40	SPI3_CS0					
				GND		GPIO4_C3(gpiochip4/gpio19)	PWM15_IR_M1	SPI3_MOSI_M1			
						GPIO4_C2(gpiochip4/gpio18)	PWM14_M1	SPI3_CLK_M1			

GPIO 引脚的计算方式如下：

$$\text{GPIO4_B2} = (\text{'B'} - \text{'A'}) * 8 + 2 = 1 * 8 + 2 = \text{gpiochip4/gpio10}$$

$$\text{GPIO3_D7} = (\text{'D'} - \text{'A'}) * 8 + 7 = 3 * 8 + 7 = \text{gpiochip3/gpio31}$$

3.3 UPS POWER



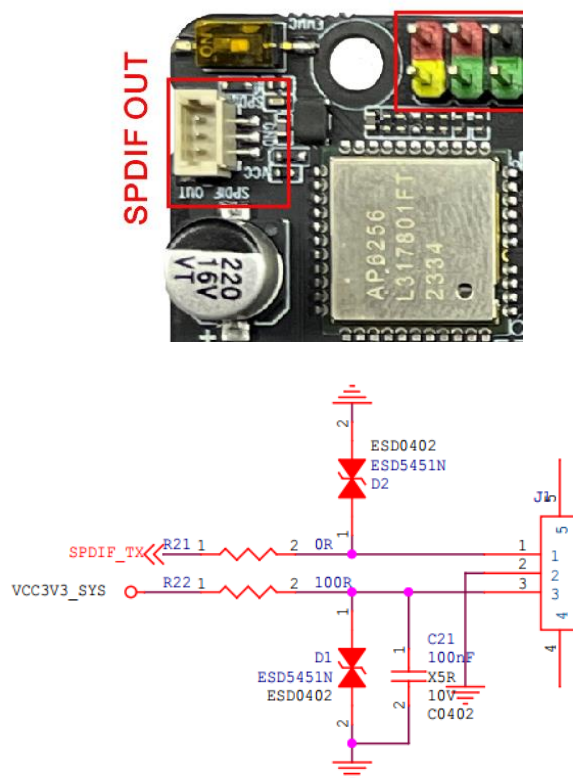
规格为 PH-2.0MM-5Pin 卧式，可配我司的 SKSM 模块使用（需要自行压接线缆和端子）

3.4 SPI FLASH



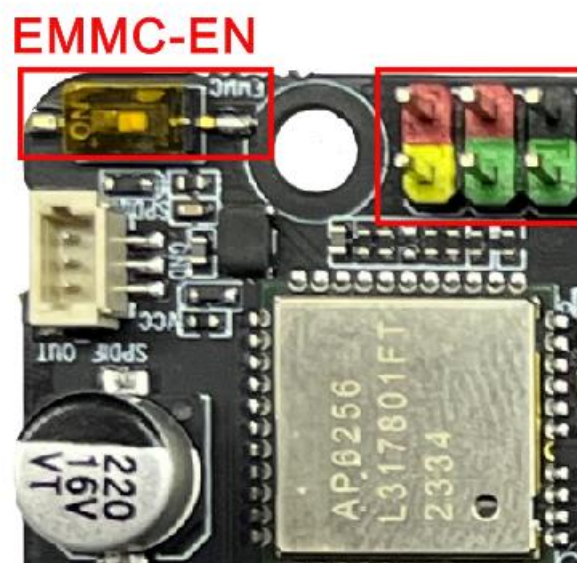
规格型号：W25Q256JWEIQ（出厂默认不带此芯片，如有需要请联系定制）

3.5 SPDIF OUT



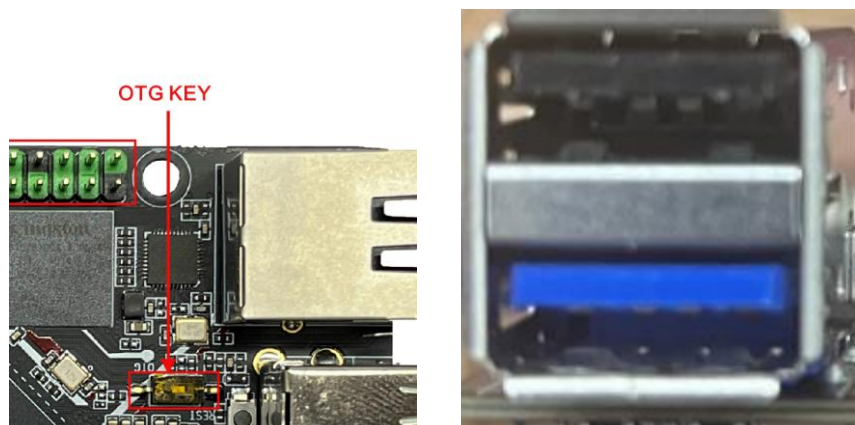
3.6 eMMC-EN

默认 OFF 挡，代表可以正常使用 eMMC，如不使用 eMMC 启动，把 EMMC-EN 档位拨到 ON 档，会将 eMMC 的信号线短路到 GND，禁用 eMMC 启动

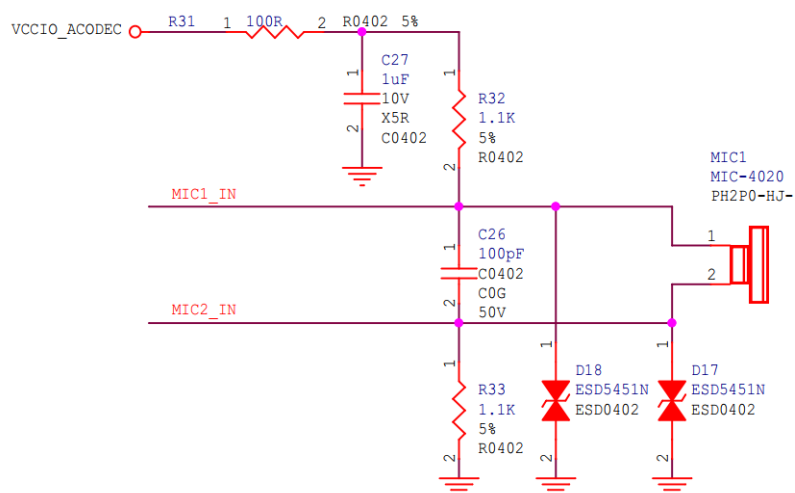
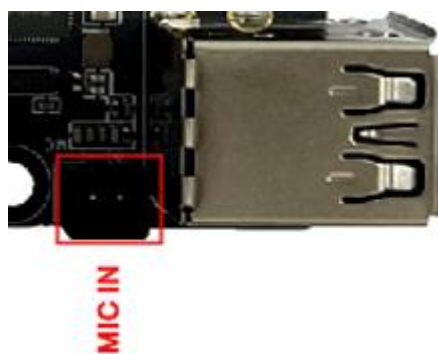


3.7 OTG

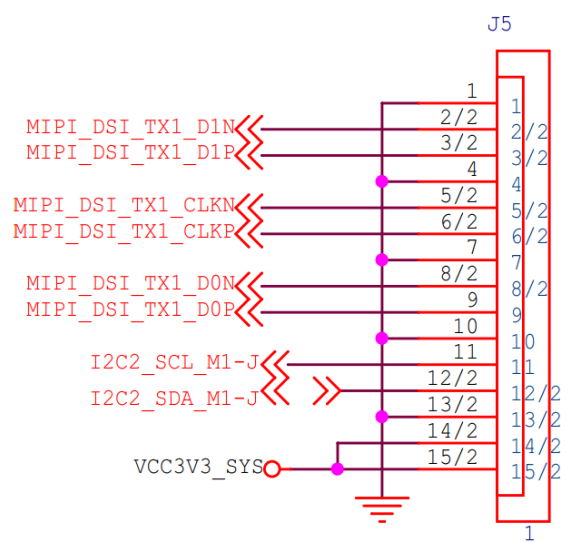
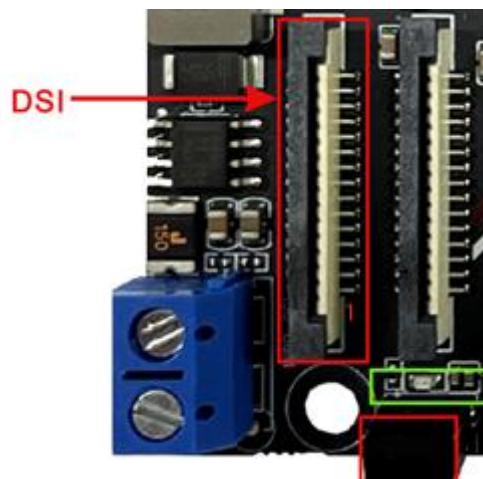
OTG 模式，请把 OTG KEY 拨到 ON 档，（注意：黑色 USB2.0 将不能正常工作）



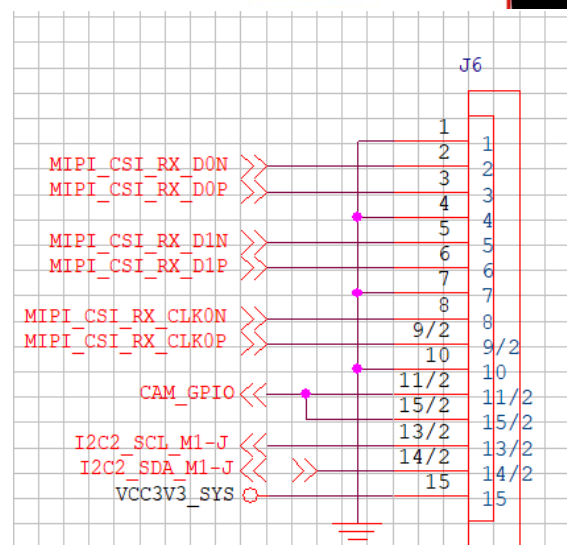
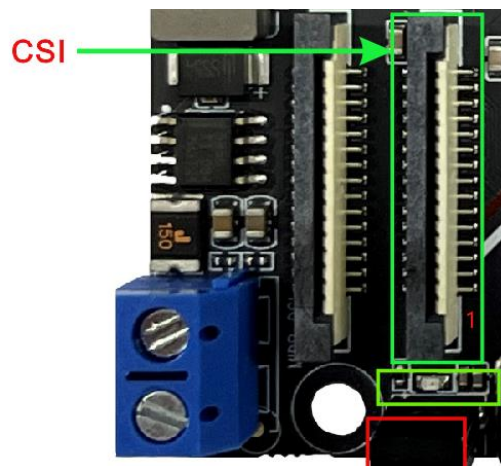
3.8 MIC IN



3.9 DSI



3.10 CSI



四、烧录系统

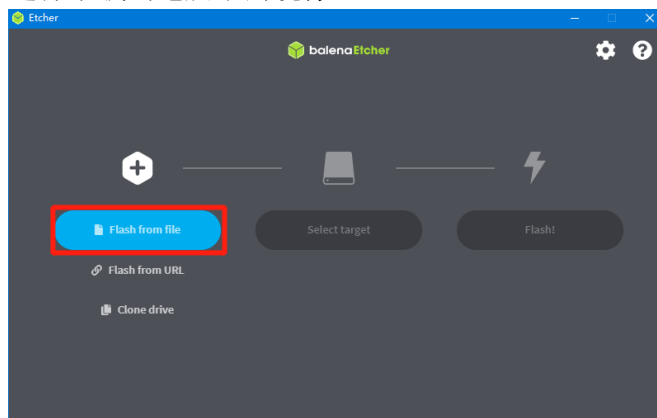
4.1 下载系统镜像

只能下载安装我们提供的系统镜像：

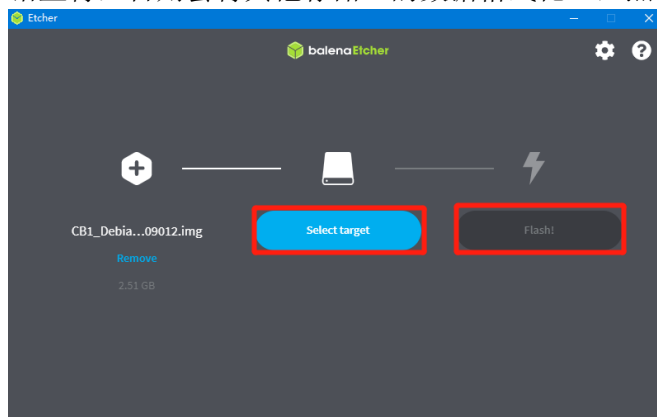
<https://github.com/bigtreotech/CB2/releases>

4.2 烧录系统到 MicroSD 卡

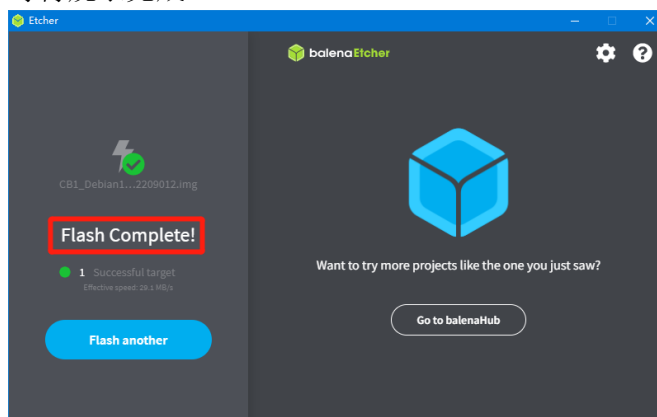
1. 下载 balenaEtcher (<https://www.balena.io/etcher/>) 软件，安装并运行
2. 将 Micro SD 卡通过读卡器插入到电脑
3. 选择下载到电脑中的镜像



4. 选择待烧录的 Micro SD 卡（烧录镜像会将 Micro SD 卡格式化，千万注意不要选错盘符，否则会将其他存储上的数据格式化），点击“烧录”



5. 等待烧录完成

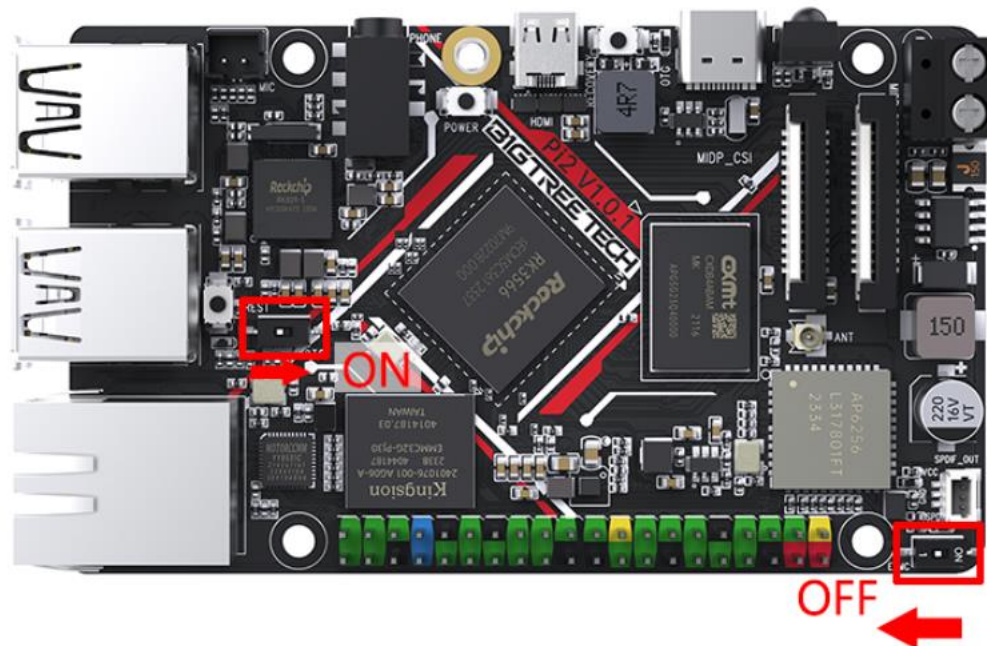


4.3 烧录系统到 eMMC

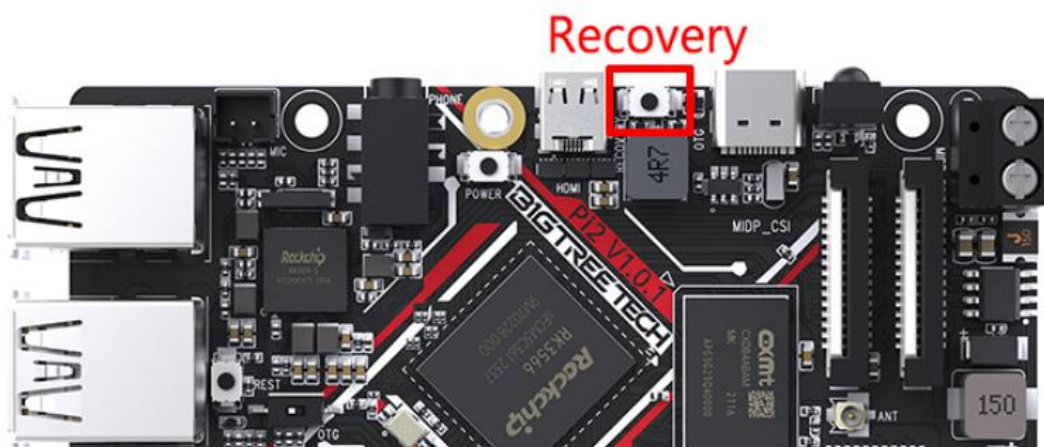
4.3.1 使用 RKDevTool 烧录系统到 eMMC (Windows)

下载 RKDevTool (<https://github.com/bigtreetech/CB2>) 到电脑上并解压。并且注意不要插 MicroSD 卡。

1. 如下图所示，将 USB OTG 的拨码开关拨到 ON 挡位，此时 USB OTG 端口接到了 Type-C 上，将 eMMC 的拨码开发拨到 OFF 挡位，此时 RK3566 可以正常访问 eMMC

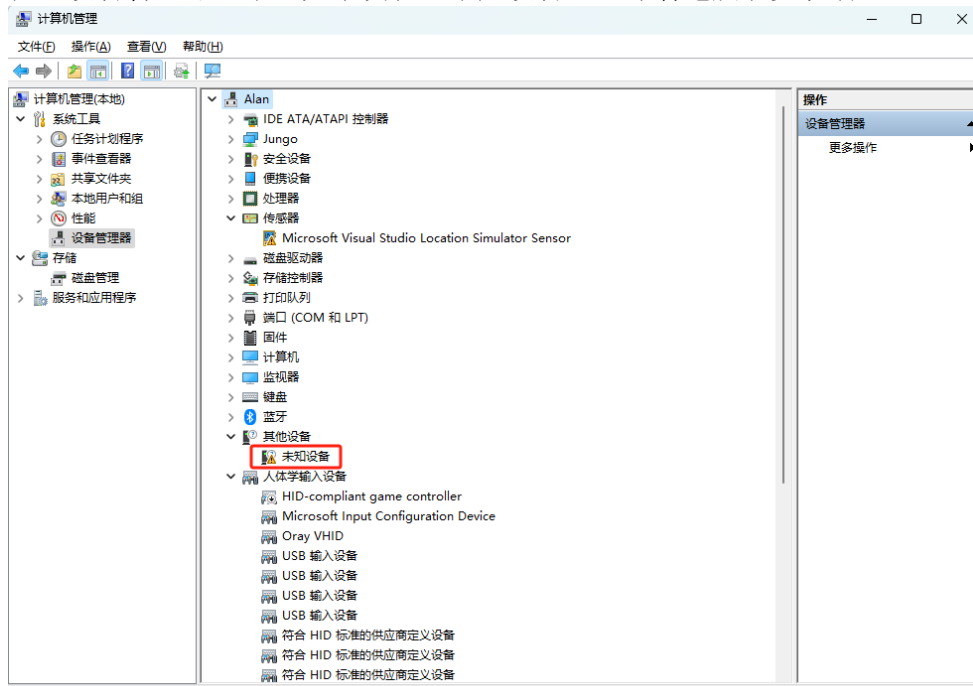


2. 按住“Recovery”键，然后用 Type-C 线将 BIGTREETECH Pi 2 插到电脑上，通电 3s 后即可松开按键。

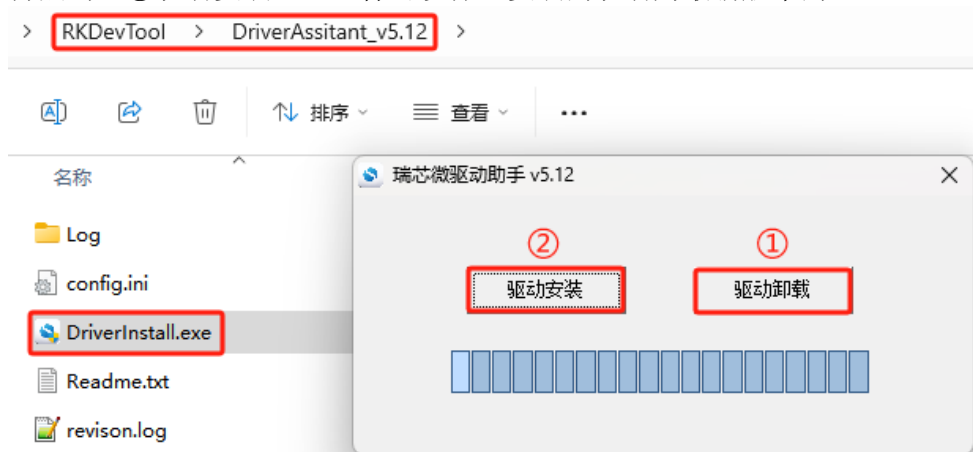


3. 安装驱动

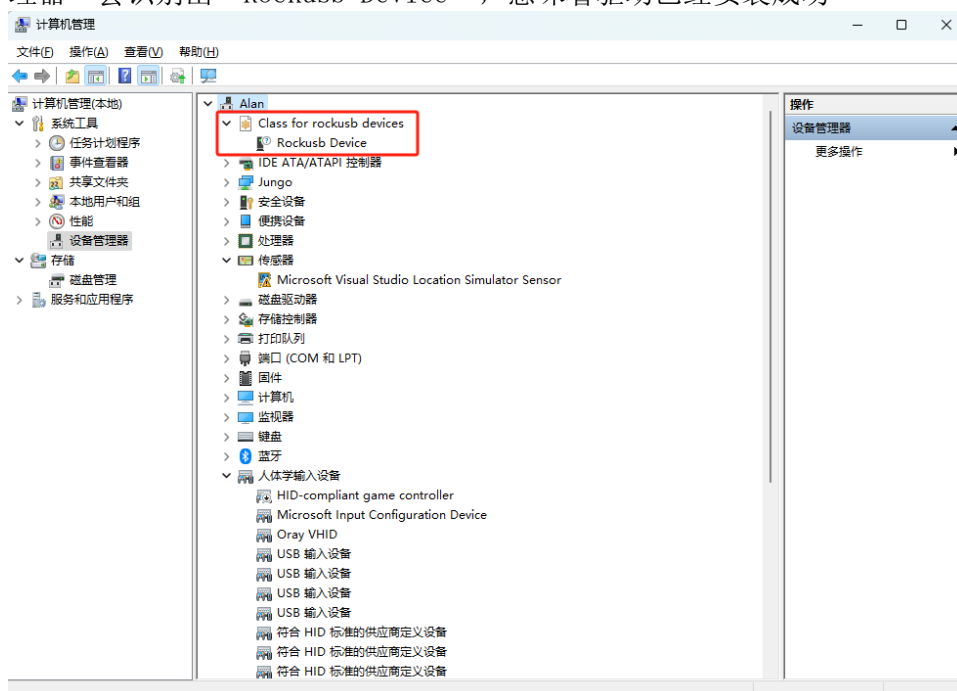
(1) 在“设备管理器”中，如果发现“未知设备”意味着电脑缺少驱动



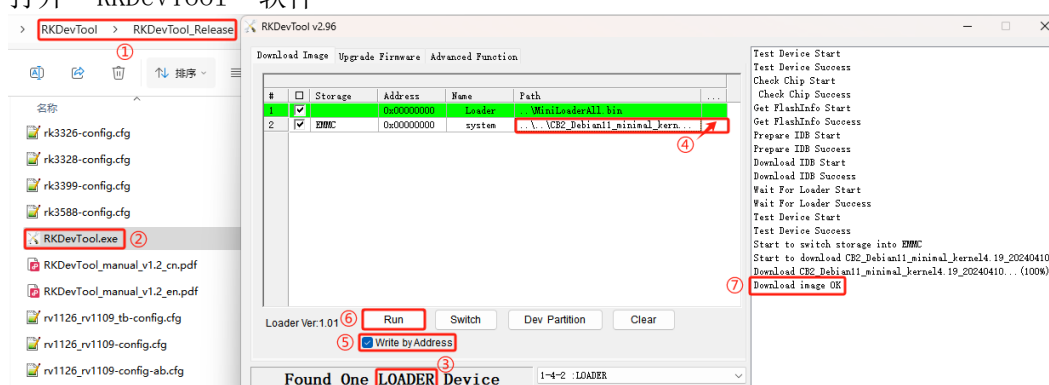
(2) 打开下载的 RKDevTool 中的 DriverAssitant 工具，先点击“①驱动卸载”，再点击“②驱动安装”，这样可以保证安装的驱动为最新版本的。



- (3) 等待安装完成后，按住“Recovery”键，重新拔插一下 Type-C 线，“设备管理器”会识别出“Rockusb Device”，意味着驱动已经安装成功



4. 打开“RKDevTool”软件



注意：软件中的参数默认如图所示，正常情况下仅需要设置④“.img 系统实际的路径”即可。如果您软件中的参数与图中不一致，请手动修改为一致。

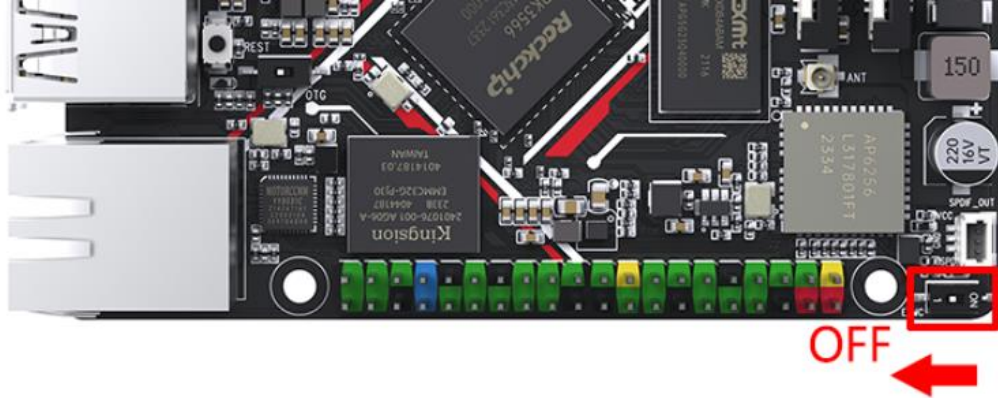
#	<input type="checkbox"/>	Storage	Address	Name	Path
1	<input checked="" type="checkbox"/>		0x00000000	Loader	..\MiniLoaderAll.bin
2	<input checked="" type="checkbox"/>	EMMC	0x00000000	System	.img 系统实际的路径

- ① 找到下载的工具所在的路径
 - ② 打开 RKDevTool 工具
 - ③ 软件会识别出一个“LOADER”或者“MASKROOM”的设备
 - ④ 选择要烧录的系统（系统镜像需要提前解压为.img 文件，此工具不支持直接烧录压缩后的.xz 文件）
 - ⑤ 勾选“Write by Address”
 - ⑥ 点击“Run”，开始烧录系统
 - ⑦ “Download image OK”意味着系统已经烧录成功
5. 烧录完成后，请将 USB OTG 的拨码开关拨到 OFF 挡位，此时即可正常开机使用了。
注意：eMMC 内的文件无法像 MicroSD 卡那样直接被电脑访问，所以无法通过修改

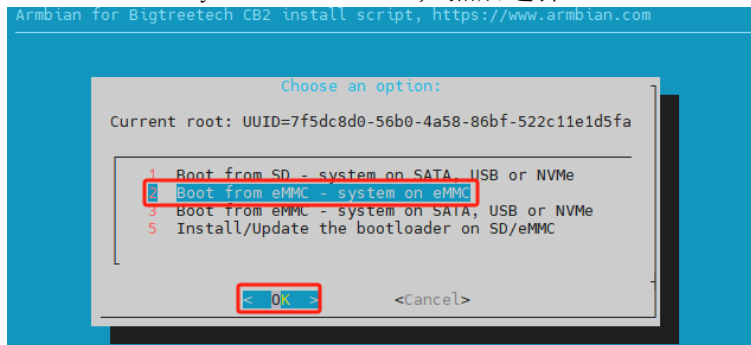
system.cfg 配置文件的方式配置 WiFi 网络，只能用网线或者 USB 转 UART 连接终端，然后通过终端配置。

4.3.2 使用 MicroSD 卡烧录系统到 eMMC

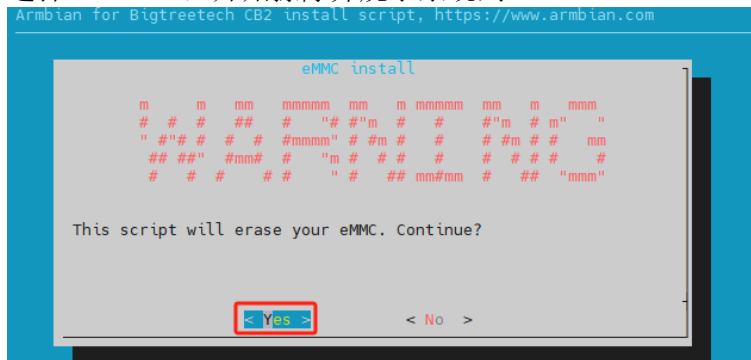
1. 如下图所示，将 eMMC 的拨码开发拨到 OFF 挡位，此时 RK3566 可以正常访问 eMMC



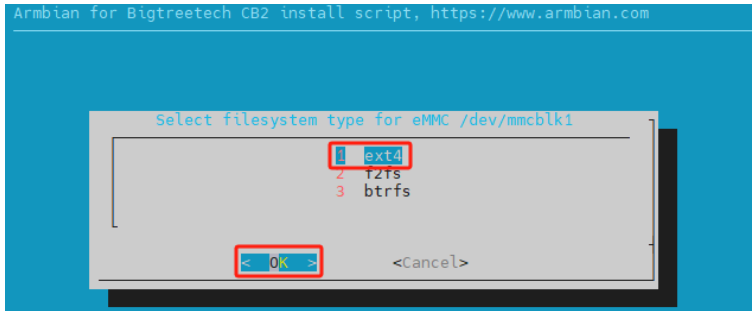
2. 先将系统烧录到 MicroSD 卡中，然后将 MicroSD 卡插到主板的卡槽，然后等待系统启动。
 3. 通过网线，WiFi 或者 USB 转 UART 连接到系统的终端，登录系统
login: biqu
password: biqu
- (1) 运行 `sudo nand-sata-install` 命令，在弹出的界面中，选择 “2 Boot From eMMC - system on eMMC”，然后选择 “OK”



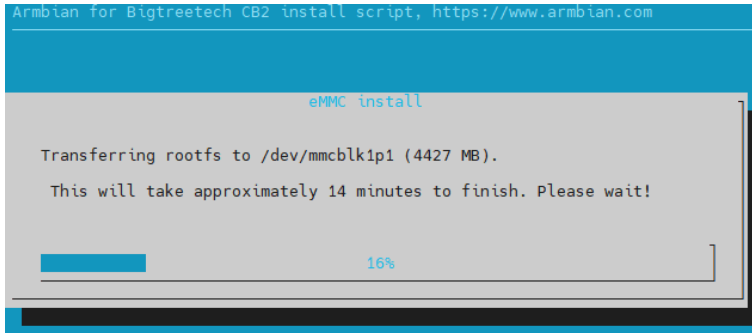
- (2) 选择 “Yes”，开始擦除并烧录系统到 eMMC



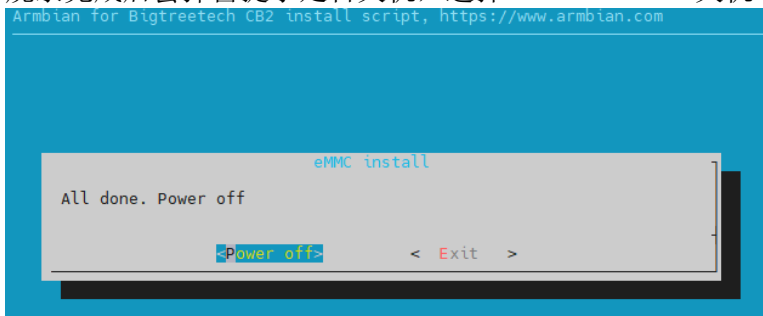
- (3) 选择文件系统为 “1 ext4”，然后选择 “OK”



- (4) 等待烧录完成



- (5) 烧录完成后会弹窗提示是否关机，选择 “Power off” 关机



- (6) 关机后断电，然后拔出 MicroSD 卡，重新再通电即可从 eMMC 启动

4.4 擦除 eMMC

当不使用 eMMC，而使用 MicroSD 卡作为系统卡时，最好将 eMMC 的数据擦除，以免主板错误的从 eMMC 启动。

4.4.1 使用 UMS 擦除 eMMC (Windows)

若 eMMC 中已经烧录过 V3.0.1 及其之后版本的系统，电脑会将 eMMC 识别为 UMS 设备（类似 U 盘一样的设备）。UMS 模式相对于 Loader 模式有以下好处：

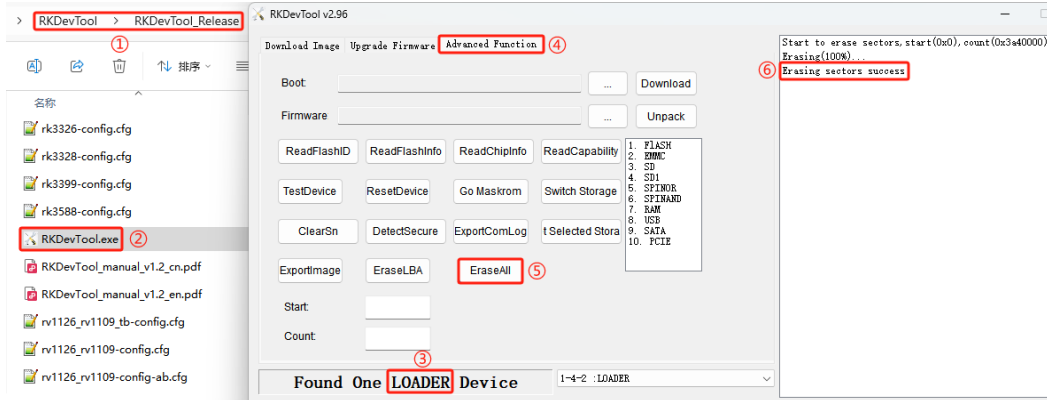
- 可以直接修改 /boot/ 分区中的配置信息
- 可以直接像 Micro SD 卡那样，直接烧录系统到 eMMC
- 可以通过软件擦除 eMMC 中的所有内容

- 参照“4.3.1 使用 RKDevTool 烧录系统到 eMMC”中的步骤，将主板连接到电脑。
- 电脑会将 eMMC 识别为 UMS 设备。
- 安装 [SD Card Formatter](#) 软件，格式化 eMMC 的 UMS 设备。（请不要直接使用 windows 系统提供的格式化功能，因为它无法完全擦除 eMMC 中的数据）

4.4.2 使用 RKDevTool 擦除 eMMC (Windows)

若 eMMC 中已经烧录过 V2.0.0 及其之前版本的系统，电脑会将 eMMC 识别为 Loader 模式的设备。

1. 参照“4.3.1 使用 RKDevTool 烧录系统到 eMMC”中的步骤，将主板连接到电脑
2. 打开“RKDevTool”软件



- ① 找到下载的工具所在的路径
- ② 打开 RKDevTool 工具
- ③ 软件会识别出一个“LOADER”的设备，如果是“MASKROOM”则说明 eMMC 中没有数据，不需要擦除
- ④ 点击“Advanced Function”
- ⑤ 点击“EraseAll”开始擦除 eMMC 中的数据
- ⑥ “Erasing sectors success”擦除完成

4.4.3 从 MicroSD 卡启动系统后擦除 eMMC

1. 参照“4.3.2 使用 MicroSD 卡烧录系统到 eMMC”中的步骤，登录到系统终端
2. 运行 `sudo mkfs /dev/mmcblk1` 命令，然后输入“y”确认。

```
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:~$
```

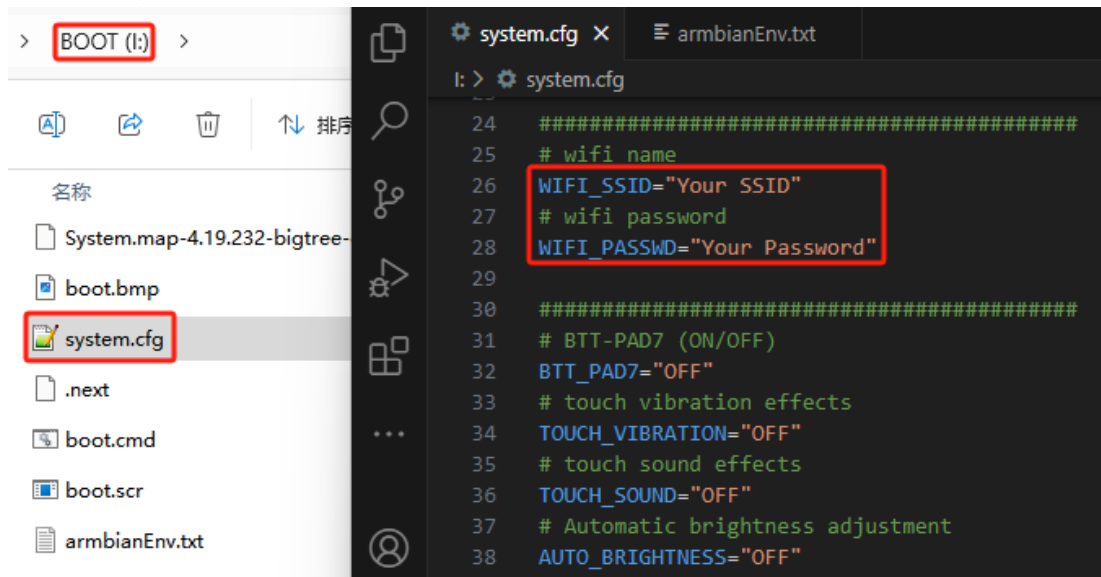
五、系统配置

5.1 使用网线

网线即插即用，不需要额外的设置

5.2 设置 WiFi

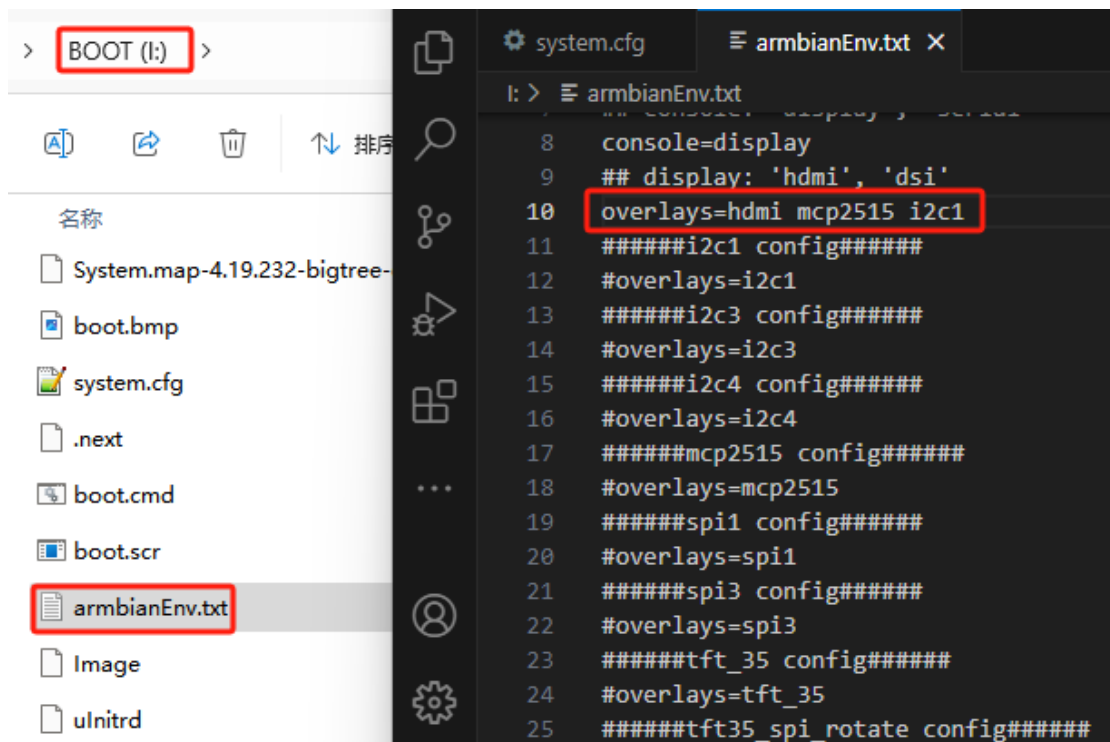
系统镜像烧录完成后，MicroSD 卡会有一个被电脑识别的 FAT32 分区，此分区下有个名为“system.cfg”的配置文件，打开后将 Your SSID 替换为实际的 WIFI 名称，Your Password 替换为实际的密码



5.3 配置 overlays

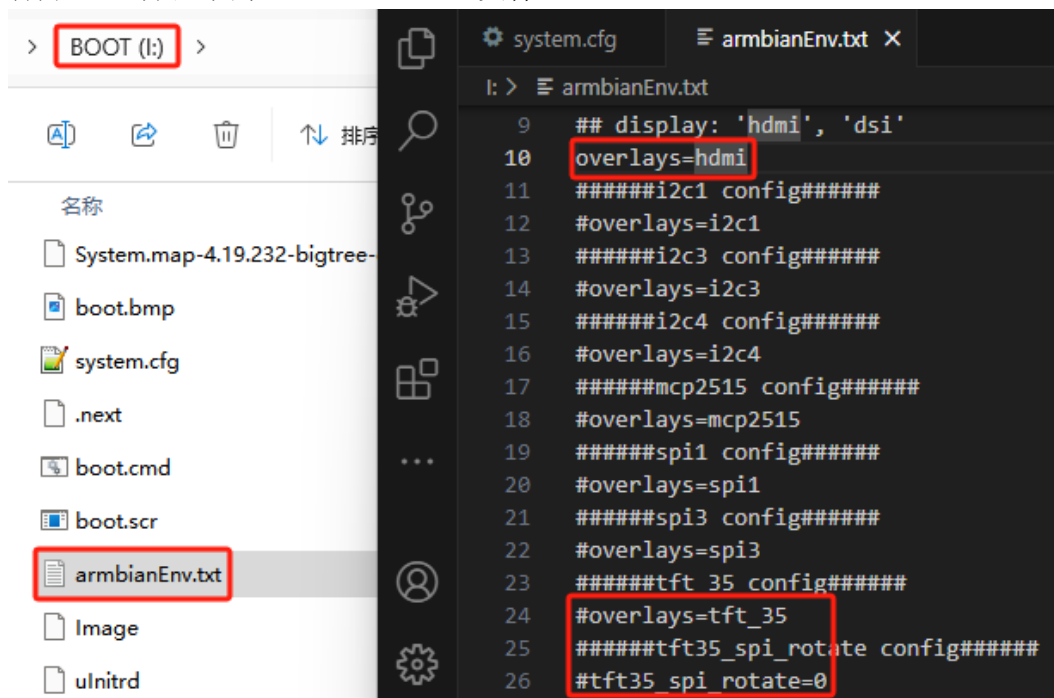
打开 BOOT 分区下的 armbianEnv.txt 文件, 设置 overlays 的值。配置文件中同一时间仅支持打开一行 overlays, 如果打开了多行 overlays 的配置, 只会生效最后一行的配置。如果有打开多个 overlays 配置的需求, 可以将多个配置的内容放在同一行 overlays 后面, 并且多个配置中间用一个空格隔开。例如我们需要同时使用 DSI 屏幕、mcp2515 SPI 转 CAN 模块, 和 I2C1:

```
overlays=dsi mcp2515 i2c1
```



5.4 配置显示屏

1. 打开 BOOT 分区下的 armbianEnv.txt 文件

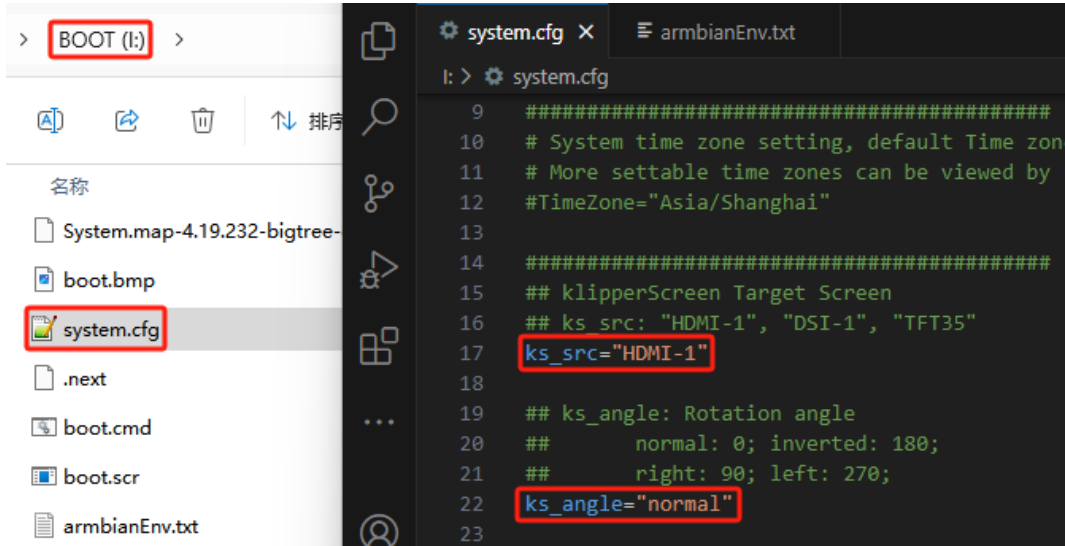


2. overlays 默认设置为 hdmi，代表系统默认使用 hdmi 屏幕。可以将其修改为实际使用的屏幕，可设置的选项如下：
 - “hdmi” : [HDMI 接口的屏幕](#)
 - “dsi” : [DSI 接口的屏幕](#)
 - “tft_35” : [SPI 接口 3.5 寸屏幕](#)

其中“tft_35”还有一个参数“tft35_spi_rotate”在系统级旋转显示界面，默认的“0”代表不旋转，可使用的参数还有“90”，“180”，“270”。

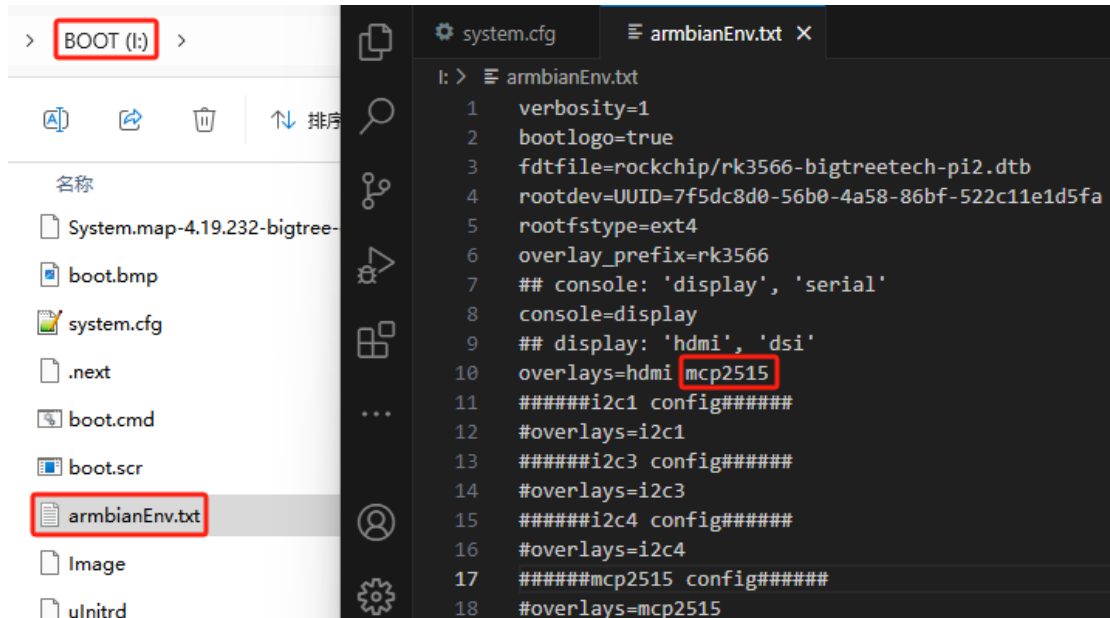
注意：屏幕只能选择使用其中的一个，无法同时使用多个屏幕

3. 设置 KlipperScreen，打开 BOOT 分区下的 system.cfg 文件，设置屏幕的类型“ks_src”，和旋转角度“ks_angle”



5.5 SPI 转 CAN 的使用

打开 BOOT 分区下的 armbianEnv.txt 文件，将“mcp2515”添加到 overlays 的配置中



5.6 CSI 相机使用及 crowsnest 配置

无论是 rpi v1.3 的 ov5647 还是 rpi v2 的 imx219 均不需要在 armbianEnv.txt 文件中配置 overlays，即插即用。

crownsnest.conf 文件中的配置如下图所示：

device: /dev/video0 # CSI 相机的节点固定为 video0

custom_flags: --format=UYVY # 当前系统 CSI 相机不支持默认的 YUYV，需要设置为支持的 UYVY 格式

```
[crownsnest]
log_path: /home/biqu/printer_data/logs/crownsnest.log
log_level: verbose # Valid Options are quiet/verbose/debug
delete_log: false # Deletes log on every restart, if set to true
no_proxy: false

[cam 1]
mode: ustreamer # ustreamer - Provides mjpg and snapshots. (All devices)
# camera-streamer - Provides webrtc, mjpg and snapshots. (rpi + Raspi OS based only)
# If camera-streamer is used, this enables also usage of an rtsp server
enable_rtsp: false # Set different ports for each device!
rtsp_port: 8554 # HTTP/MJPEG Stream/Snapshot Port
port: 8080 # See Log for available ...
# widthxheight format
device: /dev/video0 # If Hardware Supports this it will be forced, otherwise ignored/coerced.
resolution: 640x480 # You can run the Stream Services with custom flags.
max_fps: 15 # Add v4l2-ctl parameters to setup your camera, see Log what your cam is capable of.
custom_flags: --format=UYVY
#v4l2ctl:
```

5.7 蓝牙的使用

1. 扫描蓝牙设备，输入如下命令，出现如下列表的蓝牙设备，如下图

bluetoothctl --timeout 15 scan on

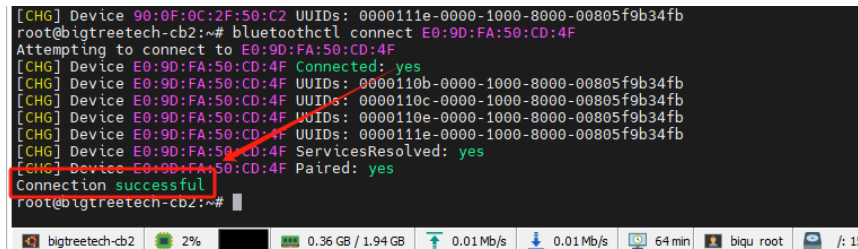
```
TX errors 0 dropped 0 overruns 0 carrier 0 collisi
root@Hurakan:~# bluetoothctl --timeout 15 scan on
Discovery started
[CHG] Controller 50:41:1C:F1:1B:DD Discovering: yes
[NEW] Device 61:81:3F:1B:B0:79 61-81-3F-1B-B0-79
[NEW] Device 67:06:15:E1:7A:62 67-06-15-E1-7A-62
[NEW] Device 78:77:40:B5:D8:02 78-77-40-B5-D8-02
[NEW] Device 61:C5:14:23:27:CC 61-C5-14-23-27-CC
[NEW] Device 7B:42:68:66:19:07 廖金花
[NEW] Device 6F:D8:78:63:4F:CD 6F-D8-78-63-4F-CD
[NEW] Device 4C:E8:2E:37:02:CE 4C-E8-2E-37-02-CE
[NEW] Device 51:22:49:FC:CF:C1 51-22-49-FC-CF-C1
[NEW] Device 73:B9:DB:2D:F1:08 73-B9-DB-2D-F1-08
[NEW] Device 54:8B:65:88:88:85 54-8B-65-88-88-85
```

2. 找到自己的蓝牙设备，比如我的蓝牙设备名字是 HONOR xSport PRO，在设备列表中找到对应的蓝牙 MAC ID 如下图

```
[CHG] Device 42:70:F4:03:91:BA ManufacturerData Value:
10 07 7a 1f 3b 4d ef 5c 68 ..z.;M.\h
[CHG] Device 4E:B0:A9:B4:33:11 RSSI: -75
[CHG] Device 45:69:88:0D:E0:7B RSSI: -92
[CHG] Device 7F:E1:35:CF:F8:A3 RSSI: -77
[CHG] Device 4E:B0:A9:B4:33:11 RSSI: -88
[CHG] Device 04:7A:0B:19:E7:AF ManufacturerData Key: 0x038f
[CHG] Device 04:7A:0B:19:E7:AF ManufacturerData Value:
0a 10 ff ff 64 93 15 36 c3 5c de 20 11 08 08 .....d..6.\. ...
10 17 25 34 ..%4
[NEW] Device E0:9D:FA:50:CD:4F HONOR xSport PRO
[CHG] Device 04:7A:0B:19:E7:AF Class: 0x000a0110
[CHG] Device 04:7A:0B:19:E7:AF Icon: computer
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 0000fdaa-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 00001105-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 0000110a-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 0000110e-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 00001112-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 0000111f-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 00001200-0000-1000-8000-00805f9b34fb
[NEW] Device 90:0F:0C:2F:50:C2 XIAOYONG
root@Hurakan:~#
```

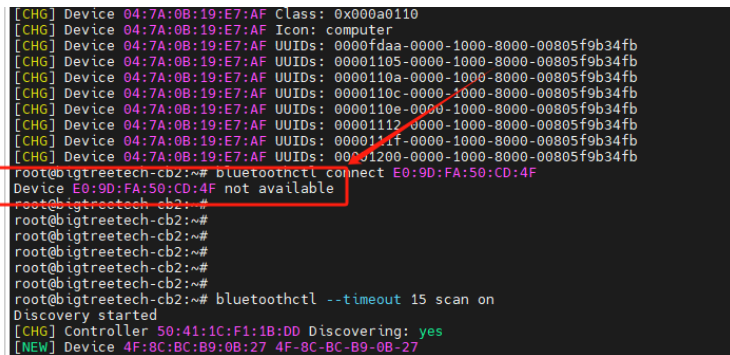

3. 连接蓝牙设备，输入如下命令，连接成功如下图

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



```
[CHG] Device 90:0F:0C:2F:50:C2 UUIDs: 0000111e-0000-1000-8000-00805f9b34fb
root@bigtreotech-cb2:~# bluetoothctl connect E0:9D:FA:50:CD:4F
Attempting to connect to E0:9D:FA:50:CD:4F
[CHG] Device E0:9D:FA:50:CD:4F Connected: yes
[CHG] Device E0:9D:FA:50:CD:4F UUIDs: 0000110b-0000-1000-8000-00805f9b34fb
[CHG] Device E0:9D:FA:50:CD:4F UUIDs: 0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device E0:9D:FA:50:CD:4F UUIDs: 0000110e-0000-1000-8000-00805f9b34fb
[CHG] Device E0:9D:FA:50:CD:4F UUIDs: 0000111e-0000-1000-8000-00805f9b34fb
[CHG] Device E0:9D:FA:50:CD:4F ServicesResolved: yes
[CHG] Device E0:9D:FA:50:CD:4F Paired: yes
Connection successful
root@bigtreotech-cb2:~#
```

(1) 若出现如下图输出，请重新打开蓝牙设备，然后重新按 1 和 2 的步骤连接蓝牙设备



```
[CHG] Device 04:7A:0B:19:E7:AF Class: 0x000a0110
[CHG] Device 04:7A:0B:19:E7:AF Icon: computer
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 0000fd00-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 00001105-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 0000110a-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 0000110e-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 00001112-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 0000111f-0000-1000-8000-00805f9b34fb
[CHG] Device 04:7A:0B:19:E7:AF UUIDs: 00001200-0000-1000-8000-00805f9b34fb
root@bigtreotech-cb2:~# bluetoothctl connect E0:9D:FA:50:CD:4F
Device E0:9D:FA:50:CD:4F not available
root@bigtreotech-cb2:~#
root@bigtreotech-cb2:~#
root@bigtreotech-cb2:~#
root@bigtreotech-cb2:~#
root@bigtreotech-cb2:~# bluetoothctl --timeout 15 scan on
Discovery started
[CHG] Controller 50:41:1C:F1:1B:DD Discovering: yes
[NEW] Device 4F:8C:BC:B9:0B:27 4F-8C-BC-B9-0B-27
```

(2) 若如下图输出，请输入如下命令，然后重新进行 1 和 2 步骤

```
bluetoothctl remove E0:9D:FA:50:CD:4F (您的蓝牙设备对应的 MAC ID)
```

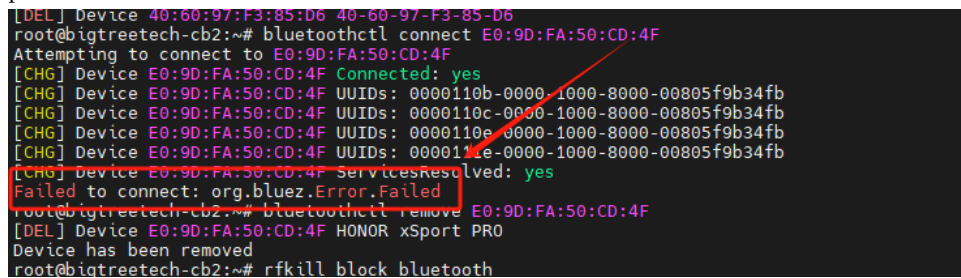
```
rfskill block bluetooth
```

```
sleep 3s
```

```
rfskill unblock bluetooth
```

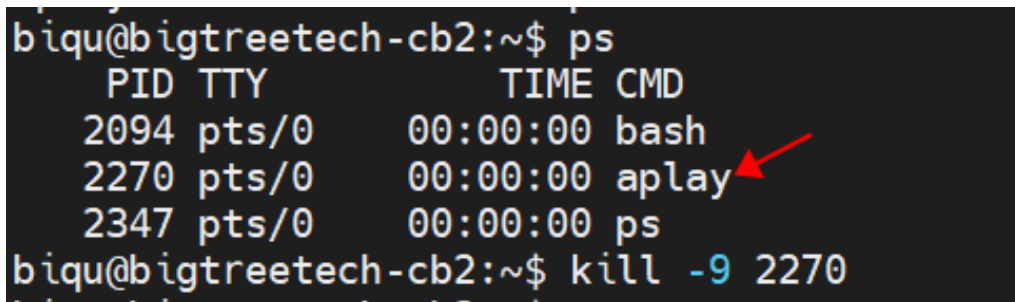
```
pulseaudio -k
```

```
pulseaudio --start
```



```
[DEL] Device 40:60:97:F3:85:D6 40-60-97-F3-85-D6
root@bigtreotech-cb2:~# bluetoothctl connect E0:9D:FA:50:CD:4F
Attempting to connect to E0:9D:FA:50:CD:4F
[CHG] Device E0:9D:FA:50:CD:4F Connected: yes
[CHG] Device E0:9D:FA:50:CD:4F UUIDs: 0000110b-0000-1000-8000-00805f9b34fb
[CHG] Device E0:9D:FA:50:CD:4F UUIDs: 0000110c-0000-1000-8000-00805f9b34fb
[CHG] Device E0:9D:FA:50:CD:4F UUIDs: 0000110e-0000-1000-8000-00805f9b34fb
[CHG] Device E0:9D:FA:50:CD:4F UUIDs: 0000111e-0000-1000-8000-00805f9b34fb
[CHG] Device E0:9D:FA:50:CD:4F ServicesResolved: yes
Failed to connect: org.bluez.Error.Failed
root@bigtreotech-cb2:~# bluetoothctl remove E0:9D:FA:50:CD:4F
[DEL] Device E0:9D:FA:50:CD:4F HONOR xSport PRO
Device has been removed
root@bigtreotech-cb2:~# rfskill block bluetooth
```

4. 蓝牙使用中途退出语音播放功能，如果不能再次使用蓝牙，需要手动删除对应的播放进程，用 ps 命令查看播放的进程号，然后用 kill -9 进程号 删除对应的播放进程。如下图所示



```
biqu@bigtreotech-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@bigtreotech-cb2:~$ kill -9 2270
```

5.8 3.5mm 圆口耳机设置

1. 输入命令:

aplay -l

查看对应的声卡，如下图所示：（由图所示耳机口的声卡对应的是 card 0）

```
[ General system configuration (beta): armbian-config ]
Last login: Wed Apr 10 02:18:28 UTC 2024 on tty1
root@bigtreetech-cb2:~#
root@bigtreetech-cb2:~#
root@bigtreetech-cb2:~#
root@bigtreetech-cb2:~# aplay -l
**** List of PLAYBACK Hardware Devices ****
card 0: rockchiprk809co [rockchip,rk809-codec], device 0: fe410000.i2s-rk817-hifi-rk817-hifi-0 [fe410000.i2s-rk817-hifi-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
card 1: rockchipbt [rockchip,bt], device 0: fe420000.i2s-bt-sco-pcm bt-sco-pcm-0 [fe420000.i2s-bt-sco-pcm-0]
  Subdevices: 1/1
  Subdevice #0: subdevice #0
root@bigtreetech-cb2:~#
root@bigtreetech-cb2:~#
root@bigtreetech-cb2:~#
root@bigtreetech-cb2:~#
```

2. 输入命令:

amixer -c 0 contents (0 表示的上述的 aplay -l 所找到的 card 0)

查看播放通道和录音通道设置，如下图所示：

```
root@bigtreetech-cb2:~#
root@bigtreetech-cb2:~#
root@bigtreetech-cb2:~#
root@bigtreetech-cb2:~# amixer -c 0 contents
numid=3,iface=MIXER,name='PCM'
; type=INTEGER,access=rw---R--,values=2,min=0,max=252,step=0
; values=255,255
| dBscale-min=-95.00dB,step=0.37dB,mute=0
numid=2,iface=MIXER,name='Capture MIC Path'
; type=ENUMERATED,access=rw-----,values=1,items=2
; Item #0 'MIC OFF'
; Item #1 'Main Mic'
; values=0
numid=4,iface=MIXER,name='Capture Volume'
; type=INTEGER,access=rw---R--,values=2,min=0,max=255,step=0
; values=255,255
| dBscale-min=-95.00dB,step=0.37dB,mute=0
numid=1,iface=MIXER,name='Playback Path'
; type=ENUMERATED,access=rw-----,values=1,items=11
; Item #0 'OFF'
; Item #1 'RCV'
; Item #2 'SPK'
; Item #3 'HP'
; Item #4 'HP_NO_MIC'
; Item #5 'BT'
; Item #6 'SPK_HP'
; Item #7 'RING_SPK'
; Item #8 'RING_HP'
; Item #9 'RING_HP_NO_MIC'
; Item #10 'RING_SPK_HP'
; values=0
root@bigtreetech-cb2:~#
```

3. 输入命令:

amixer -c 0 cset numid=1 3

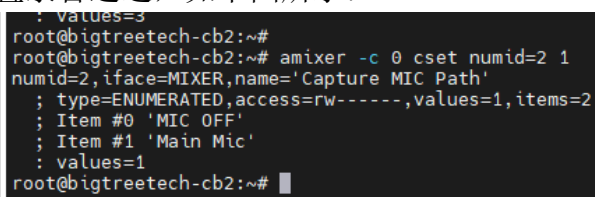
设置播放通道，如下图所示：

```
root@bigtreetech-cb2:~#
root@bigtreetech-cb2:~#
root@bigtreetech-cb2:~# amixer -c 0 cset numid=1 3
numid=1,iface=MIXER,name='Playback Path'
; type=ENUMERATED,access=rw-----,values=1,items=11
; Item #0 'OFF'
; Item #1 'RCV'
; Item #2 'SPK'
; Item #3 'HP'
; Item #4 'HP_NO_MIC'
; Item #5 'BT'
; Item #6 'SPK_HP'
; Item #7 'RING_SPK'
; Item #8 'RING_HP'
; Item #9 'RING_HP_NO_MIC'
; Item #10 'RING_SPK_HP'
; values=3
root@bigtreetech-cb2:~#
```

4. 输入命令:

amixer -c 0 cset numid=2 1

设置录音通道，如下图所示:

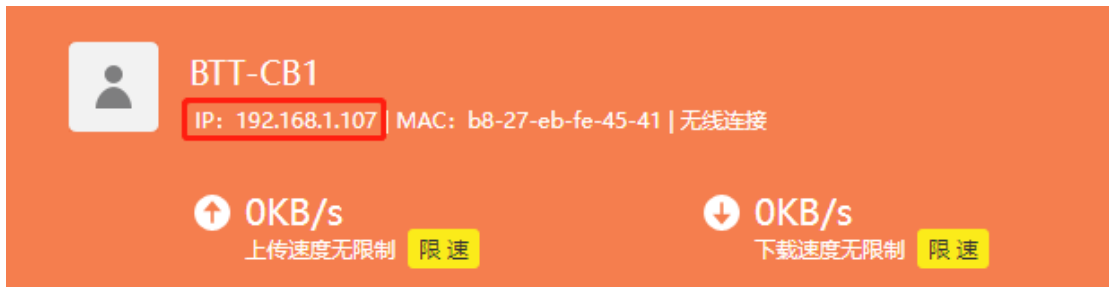


```
: values=3
root@bigtreotech-cb2:~#
root@bigtreotech-cb2:~# amixer -c 0 cset numid=2 1
numid=2,iface=MIXER,name='Capture MIC Path'
; type=ENUMERATED,access=rw-----,values=1,items=2
; Item #0 'MIC OFF'
; Item #1 'Main Mic'
: values=1
root@bigtreotech-cb2:~#
```

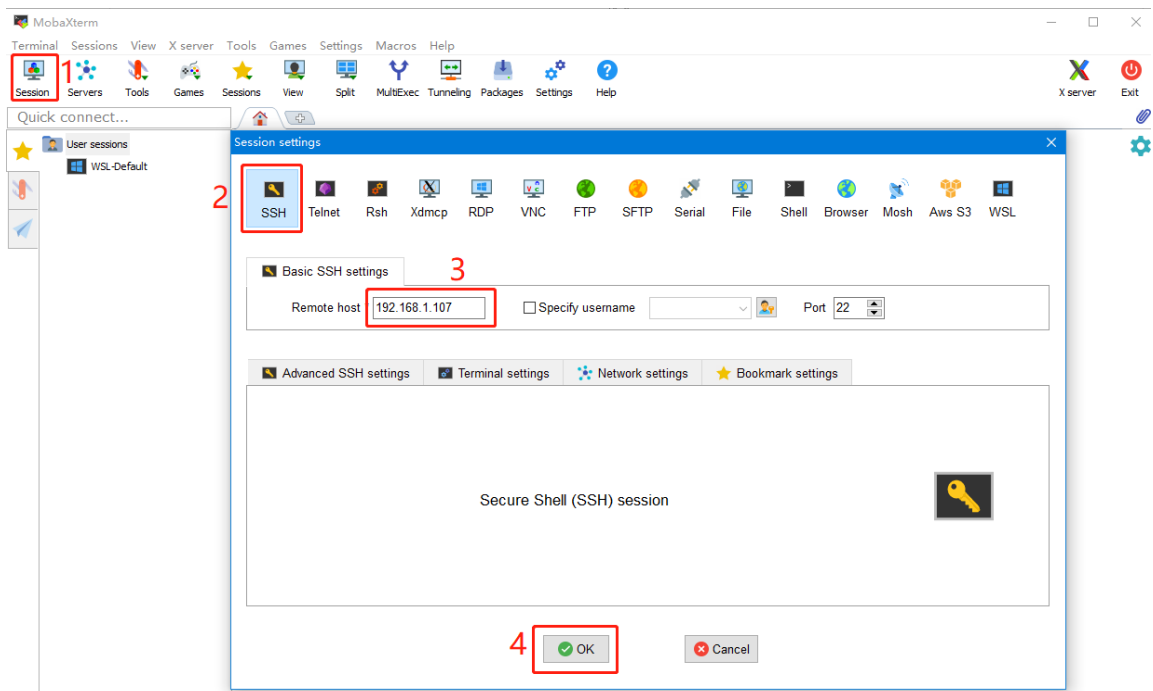
5. 输入如下命令播放音频，音频文件目录 xxx 加音频文件名 xxxxx.wav
- aplay -D plughw:0,0 /xxx/xxxxx.wav**
6. 输入如下命令录音（其中 10 表示录音 10 秒），录音存放的目录是 xxx，文件名 xxxx.wav
- sudo arecord -Dhw:0,0 -d 10 -f cd -r 44100 -c 2 -t wav /xxx/xxxx.wav**
7. 输入如下命令播放录音
- aplay -D plughw:0,0 /xxx/xxxx.wa**

六、SSH 连接设备

1. 安装 ssh 软件 MobaXterm: <https://mobaxterm.mobatek.net/download-home-edition.html>
2. 通电后等待系统启动, 大概 1~2 分钟
3. 设备连上 WIFI 或者插上网线后, 会被自动分配一个 IP
4. 进入路由器管理界面找到设备的 IP (这里应为 BTT-CB2)



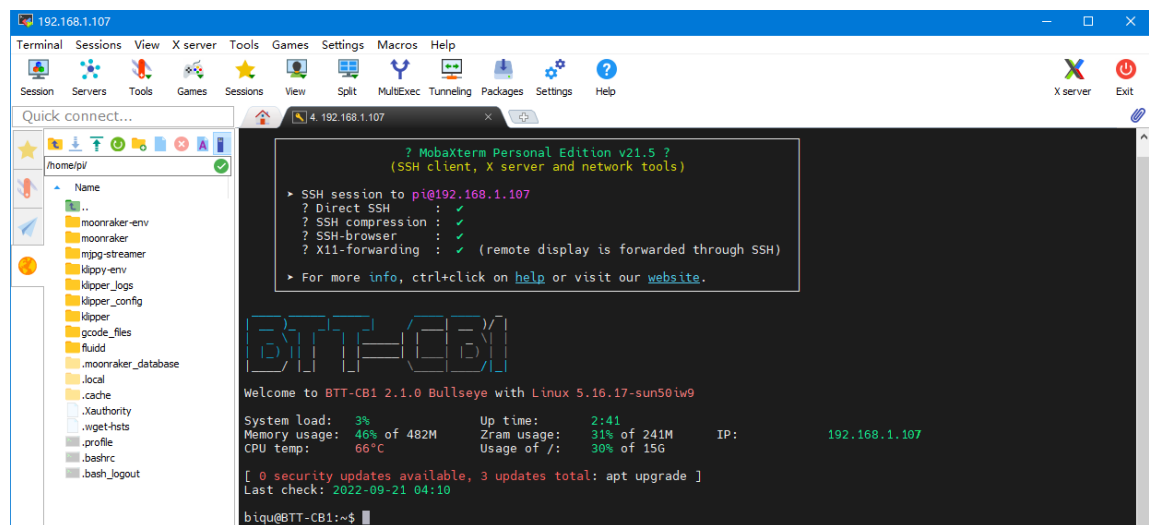
5. 打开已经安装的 MobaXterm 软件, 点击 “Session”, 在弹出的窗口中点击 “SSH”, 在 Remote host 一栏中输入设备的 IP 地址, 点击 “OK” (注意: 电脑和设备必须要在同一个局域网下)



6. 输入登录名和登录密码进入 SSH 终端界面

登录名 login as: biqu

密码: biqu



七、注意事项

1. 上电后大概 10s 左右，系统进入 kernel 阶段。此时蓝灯常亮，绿灯会不断的闪烁，代表系统在正常运行。

2. Klipper 系统

root 管理员:

login: root

password: root

biqu 普通用户:

login: biqu

password: biqu

Minimal 系统

root 管理员:

login: root

password: root

Minimal 系统是标准的 Armbian 启动流程，只有管理员帐户“root”。首次启动后，系统会引导用户在终端中创建自己的普通账户。

3. PCIe M.2 接口不支持热插拔，需要预先插上固态硬盘才能识别到设备。
4. 使用 eMMC 启动时，不要插 MicroSD 卡。使用 MicroSD 卡启动时，需要将 eMMC 中的数据擦除。

如果您还需要此产品的其他资源，可以到 <https://github.com/bigtreetech/> 上自行查找，如果无法找到您所需的资源，可以联系我们的售后支持（service005@biqu3d.com）。

若您使用中还遇到别的问题，欢迎您联系我们，我们会细心为您解答；若您对我们的产品有什么好的意见或建议，也欢迎您回馈给我们，我们也会仔细斟酌您的意见或建议，感谢您选择 BIGTREETECH 制品，谢谢！