

BIGTREE TECH

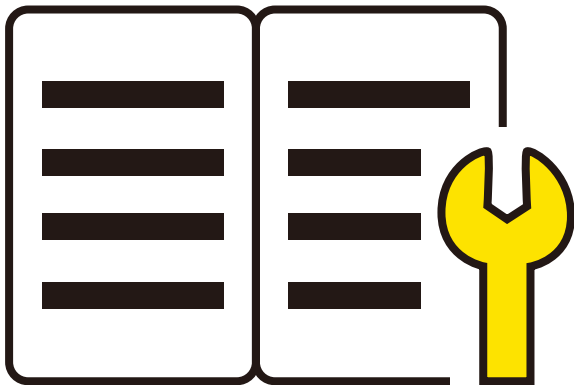


EBB SB2209 CAN V1.0 (RP2040)

用户手册

版本 2024-06-26

版本	修订日期	修改说明
v1.0	2023.11	初稿
v1.2	2024.02	- 更正：4线PT100 DIP开关选择的引脚图。 - 新增：固定CAN bus线材的钢丝直径及长度建议。 - 更新：手册中所有'CanBoot'的提及均更名为'Katapult'。
v1.3	2024.02	调整BLTouch和MicroProbe的连接方法。



感谢由 **CHAOTICLAB** 提供的Voron官方风格手册制作指导, 以及为 EBB SB2209 CAN V1.0 (RP2040) 设计安装所需的打印件模型。

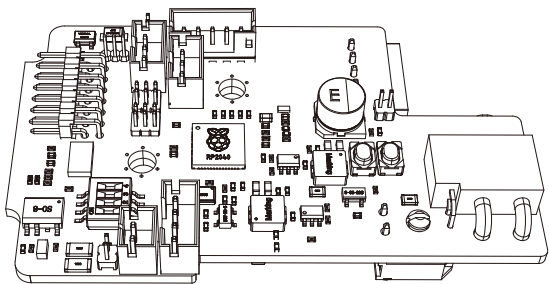
- 蓝色突出显示部件：包含于该EBB SB2209 CAN V1.0 (RP2040) 配件清单中；
- 红色突出显示部件：Voron StealthBurner的配件，需自行准备。

目录

配件清单

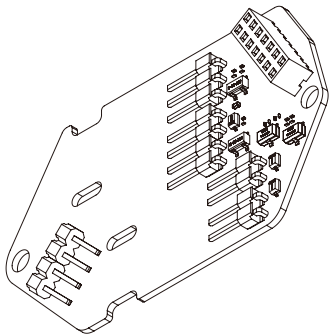
StealthBurner	1
EBB SB0000 CAN v1.0 概览	1
涡轮风扇	3
轴流风扇	4
PWM风扇	5
LED	6
ClockWork2	7
EBB SB2209 CAN V1.0 (RP2040) 概览	7
安装 EBB SB2209 CAN V1.0 (RP2040)	8
CAN bus 120R 终端电阻	10
挤出机电机	11
热敏电阻	12
PT100/1000	13
加热棒	14
X 滑车	15
接近开关	15
X轴限位开关	16
Voron Tap	17
BLTouch & MicroProbe	18

StealthBurner 最后的组装	19
CW2 电线支架	20
固定CAN / USB-C线材	22
StealthBurner 连接主板	24
Klipper	25
烧录Katapult	25
编译固件	30
通过Katapult进行固件更新	31
通过DFU进行固件更新	33
CAN bus配置	37
配置Klipper	38



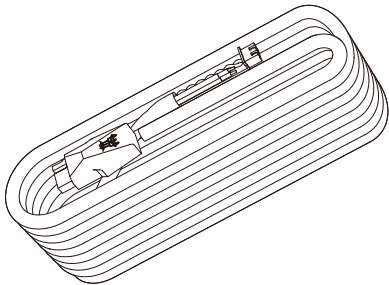
BIGTREETECH EBB SB2209 CAN V1.0 (RP2040)

1pc



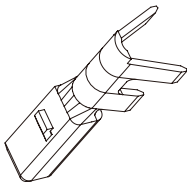
BIGTREETECH EBB SB0000 CAN V1.0

1pc



CAN bus Cable

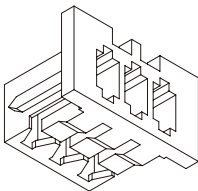
1pc



簧片

间距 2.0mm
间距 3.0mm
间距 2.54mm

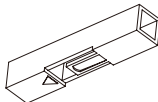
30pcs
5pcs
20pcs



间距 2.0mm 胶壳

2-pin
3-pin
4-pin
5-pin

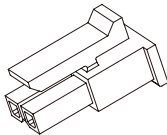
1pc
2pcs
1pc
1pc



间距 2.54mm 胶壳

1-pin

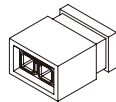
15pcs



间距 3.0mm 胶壳

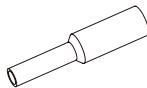
2-pin

1pc



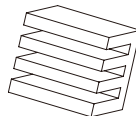
跳线帽

15pcs



管形接线端子

2pcs



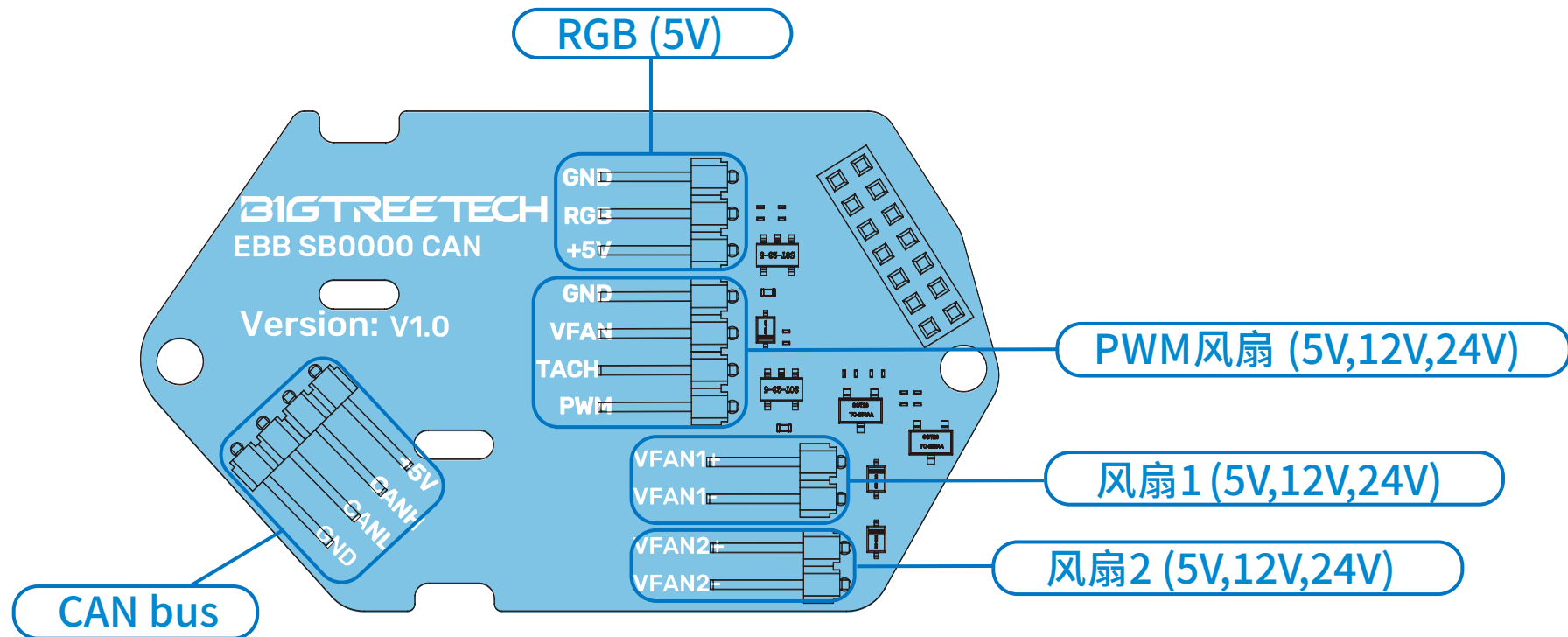
散热片

1pc



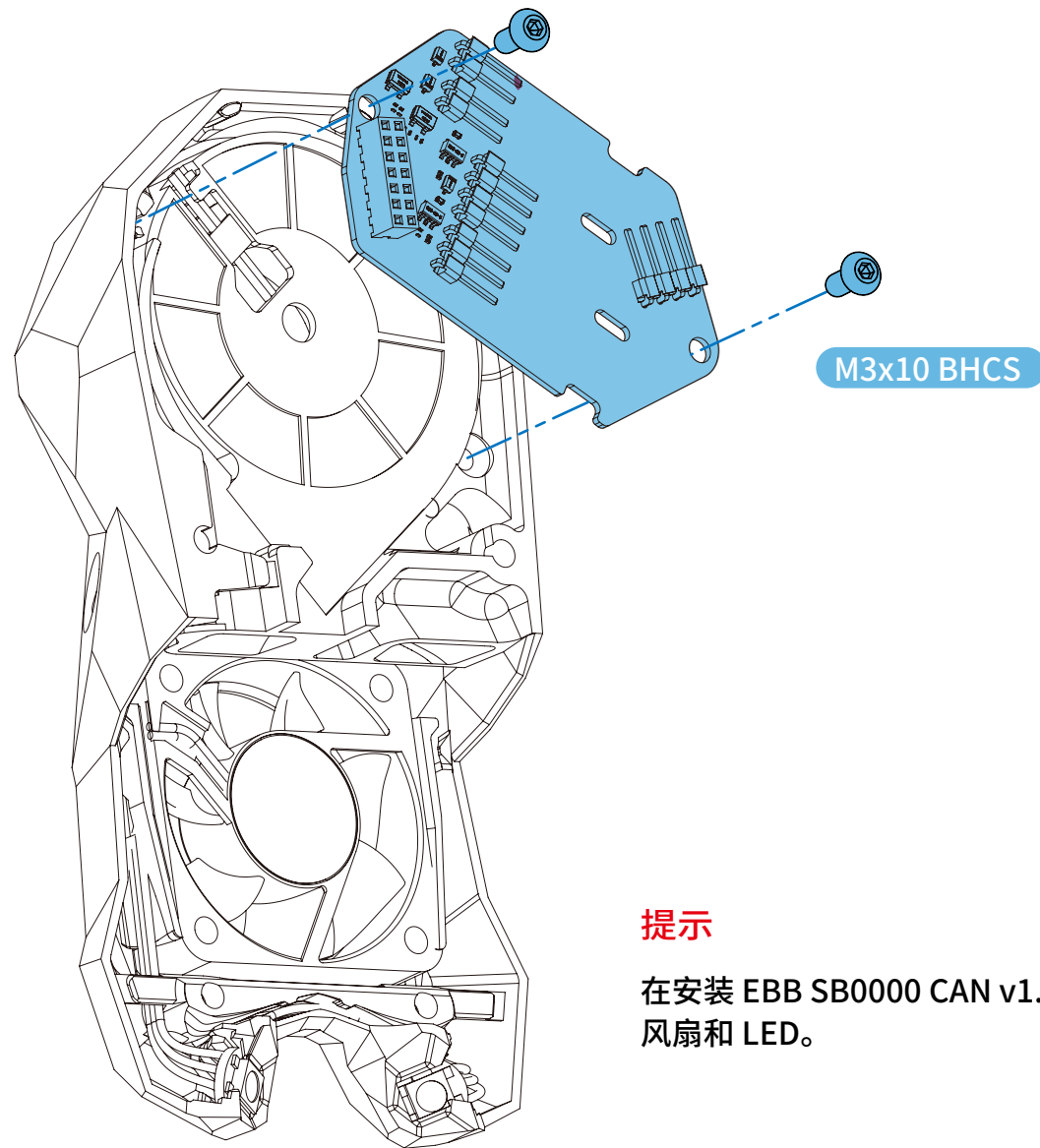
M3 x 10 圆头内六角螺丝 (BHCS)

4pcs



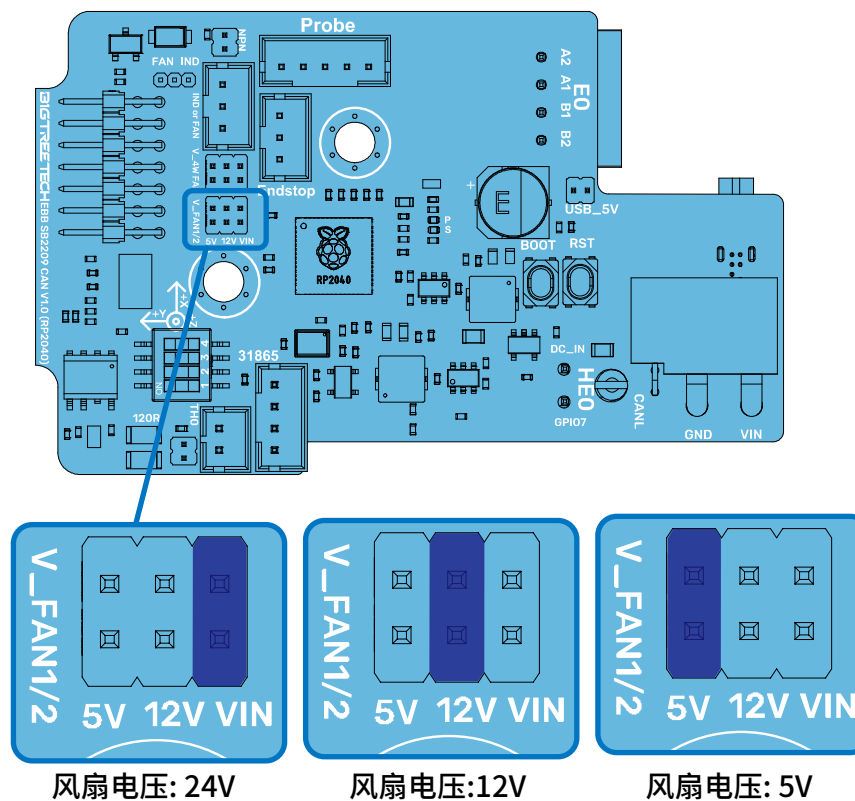
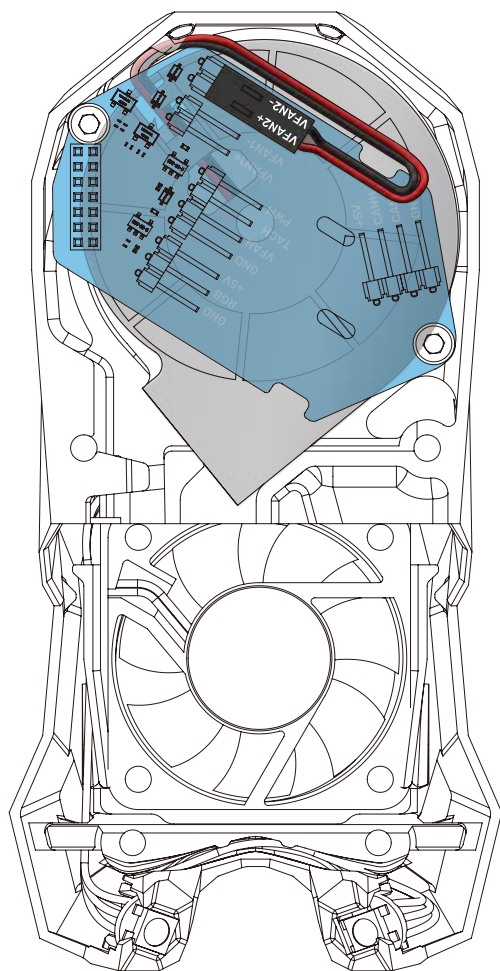
接线端子

EBB SB0000 CAN v1.0使用的是杜邦2.54mm间距的插座，因此，StealthBurner上的所有风扇和 LED必须也使用2.54间距的杜邦连接器。



提示

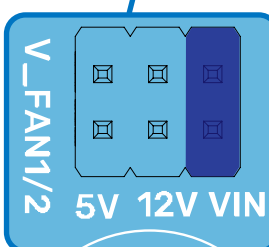
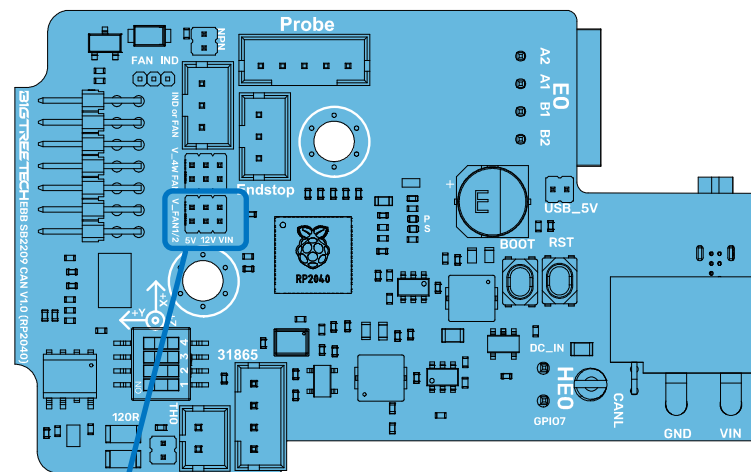
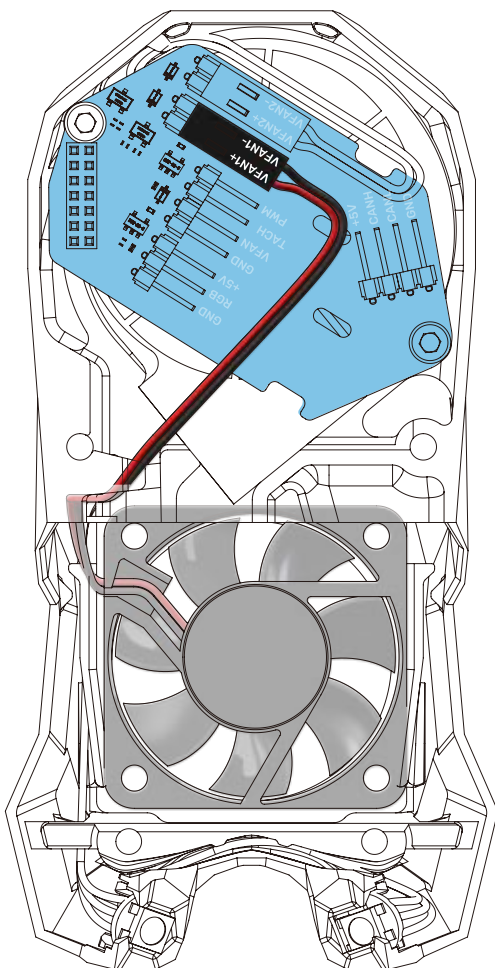
在安装 EBB SB0000 CAN v1.0前，请先安装5015涡轮风扇，4010轴流风扇和 LED。



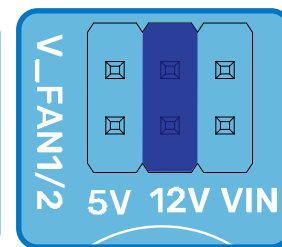
风扇电压

EBB SB0000 CAN v1.0的所有风扇均支持5/12/24V的电压选择，使用跳线帽参考上图选择对应的电压。

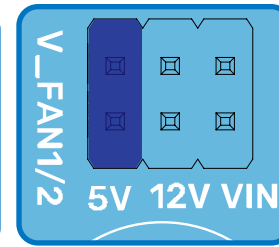
注：涡轮风扇和轴流风扇的电压将被同时选择为5/12/24V。



风扇电压: 24V



风扇电压: 12V

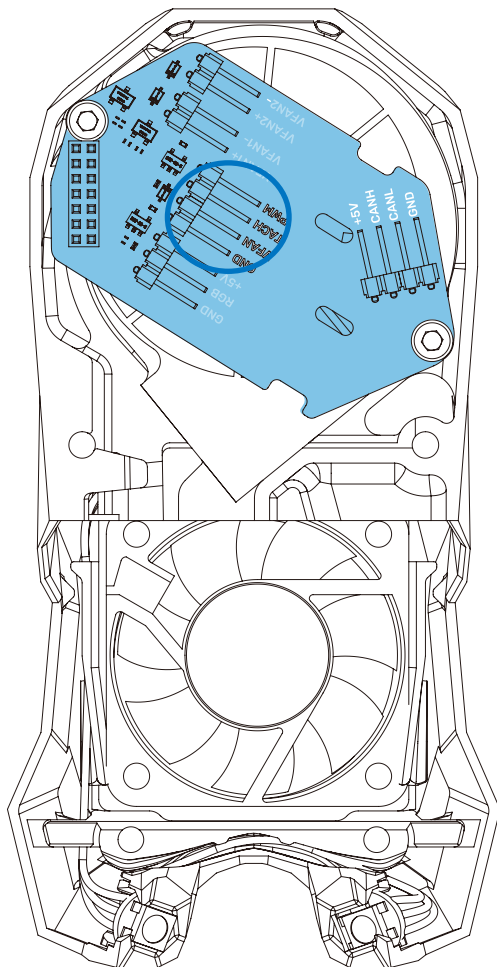


风扇电压: 5V

风扇电压

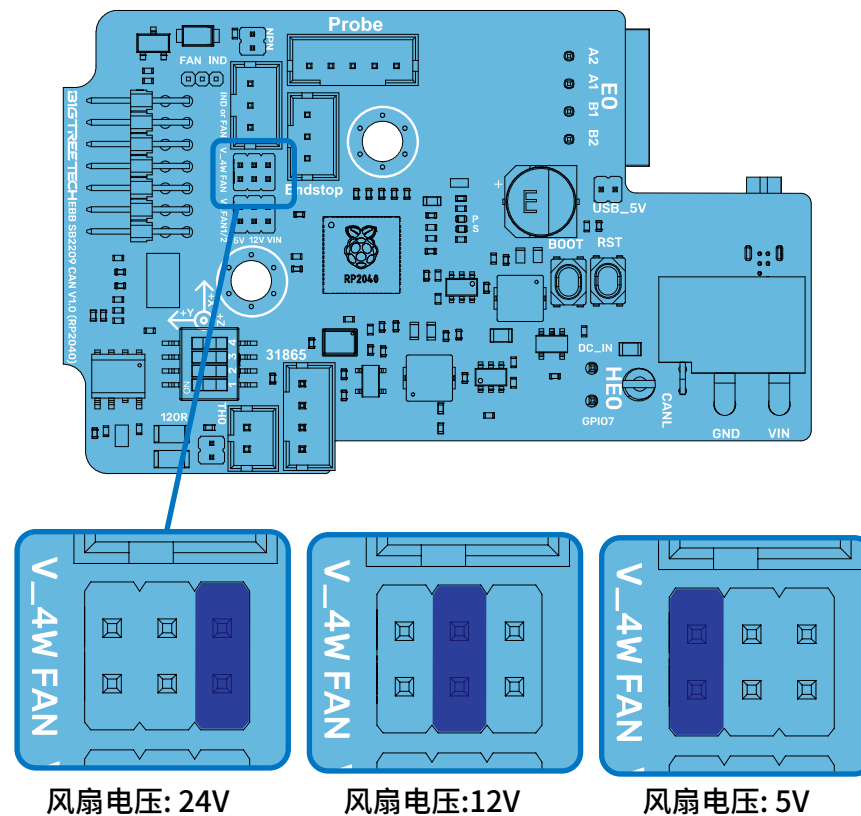
EBB SB0000 CAN v1.0的所有风扇均支持5/12/24V的电压选择，使用跳线帽参考上图选择对应的电压。

注：涡轮风扇和轴流风扇的电压将被同时选择为5/12/24V。



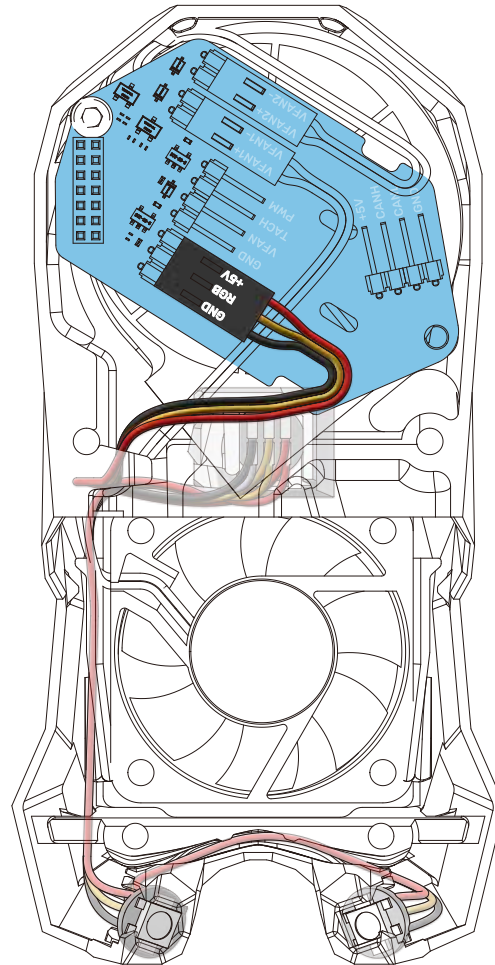
PWM风扇

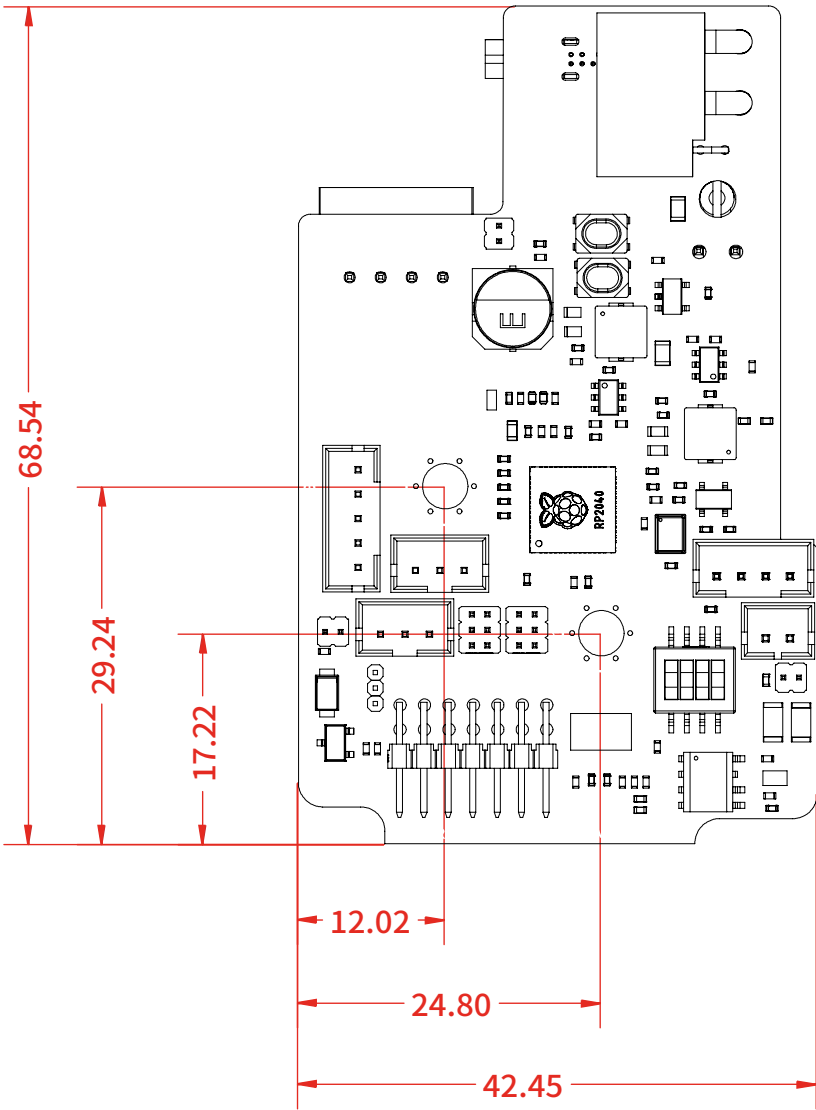
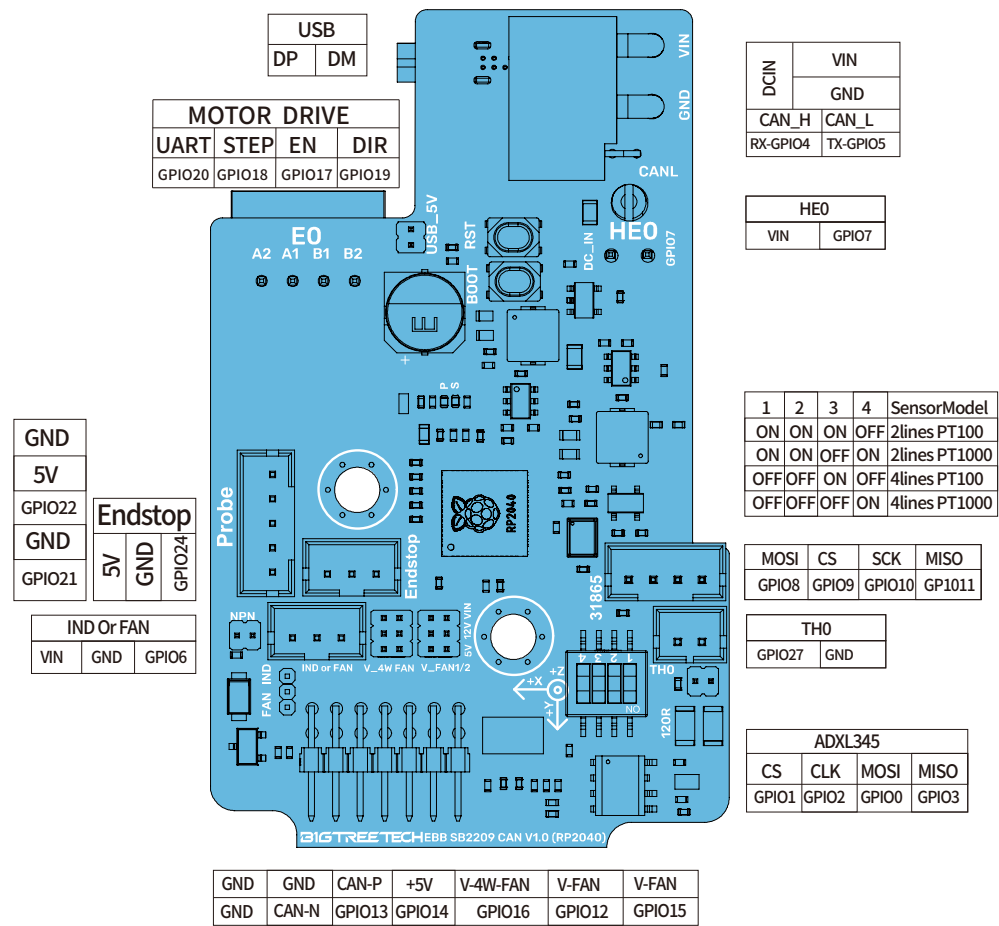
EBB SB0000 CAN v1.0设有一个2.54mm 间距4p杜邦针座，支持PWM风扇。

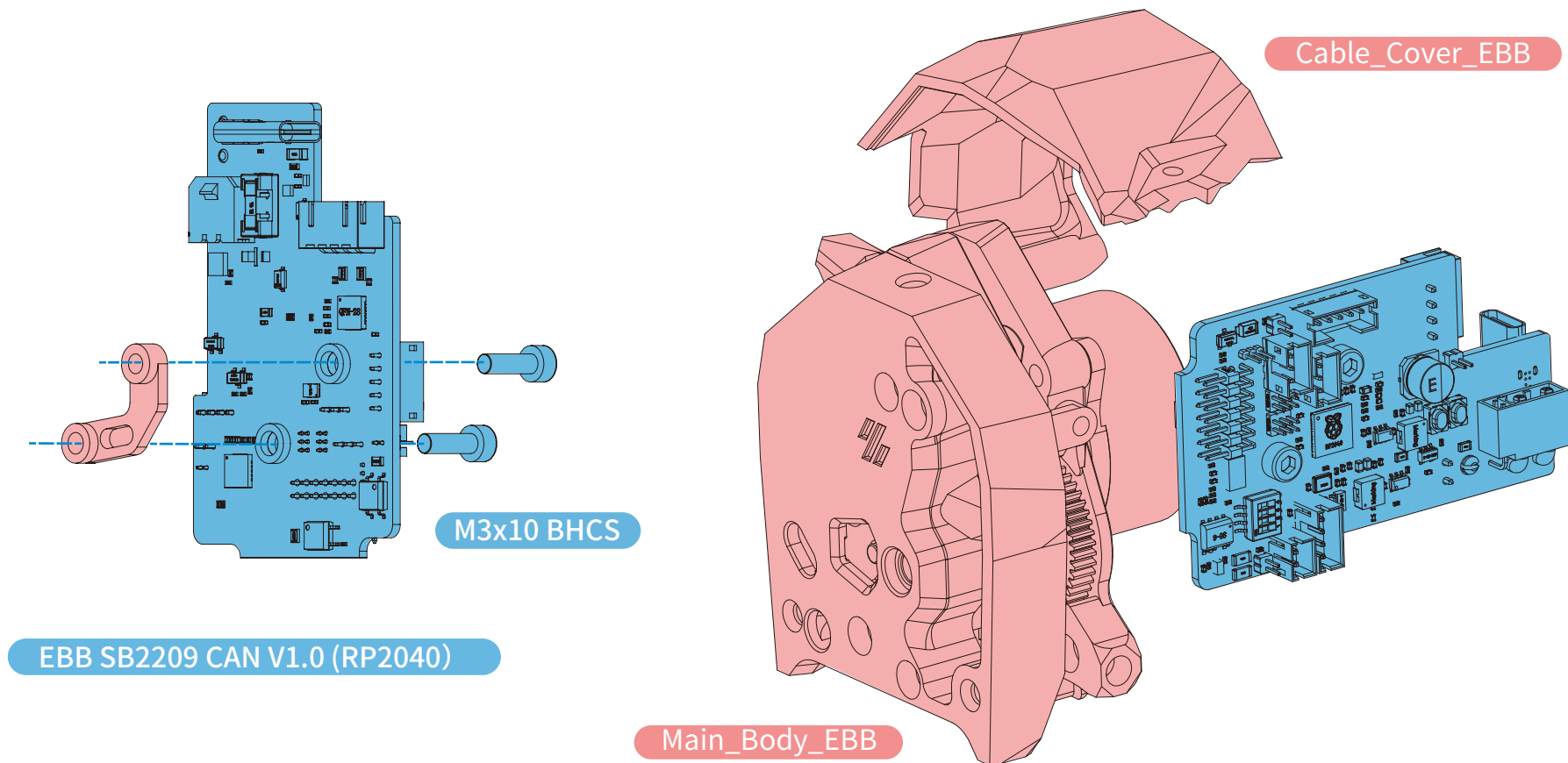


风扇电压

EBB SB0000 CAN v1.0的所有风扇均支持5/12/24V的电压选择，使用跳线帽参考上图选择对应的电压。

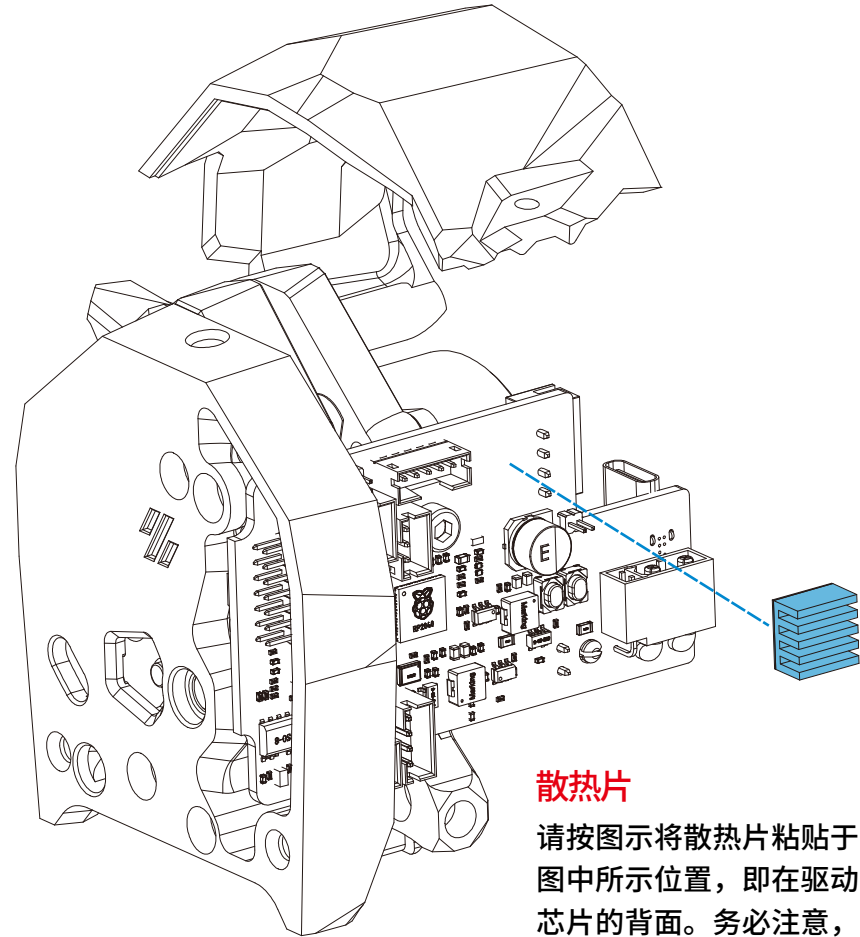
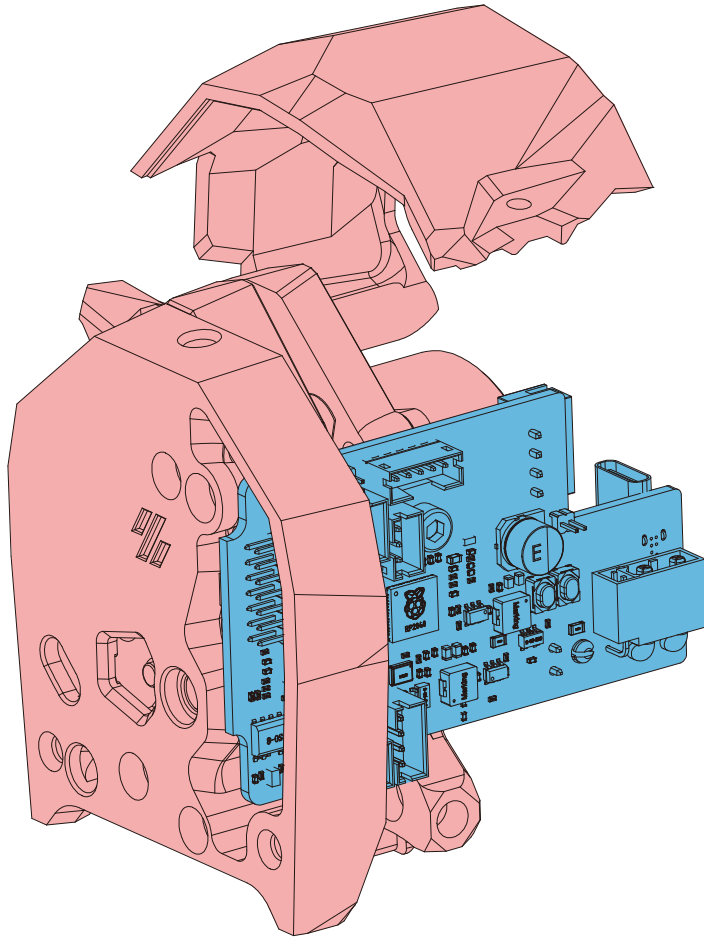






提示

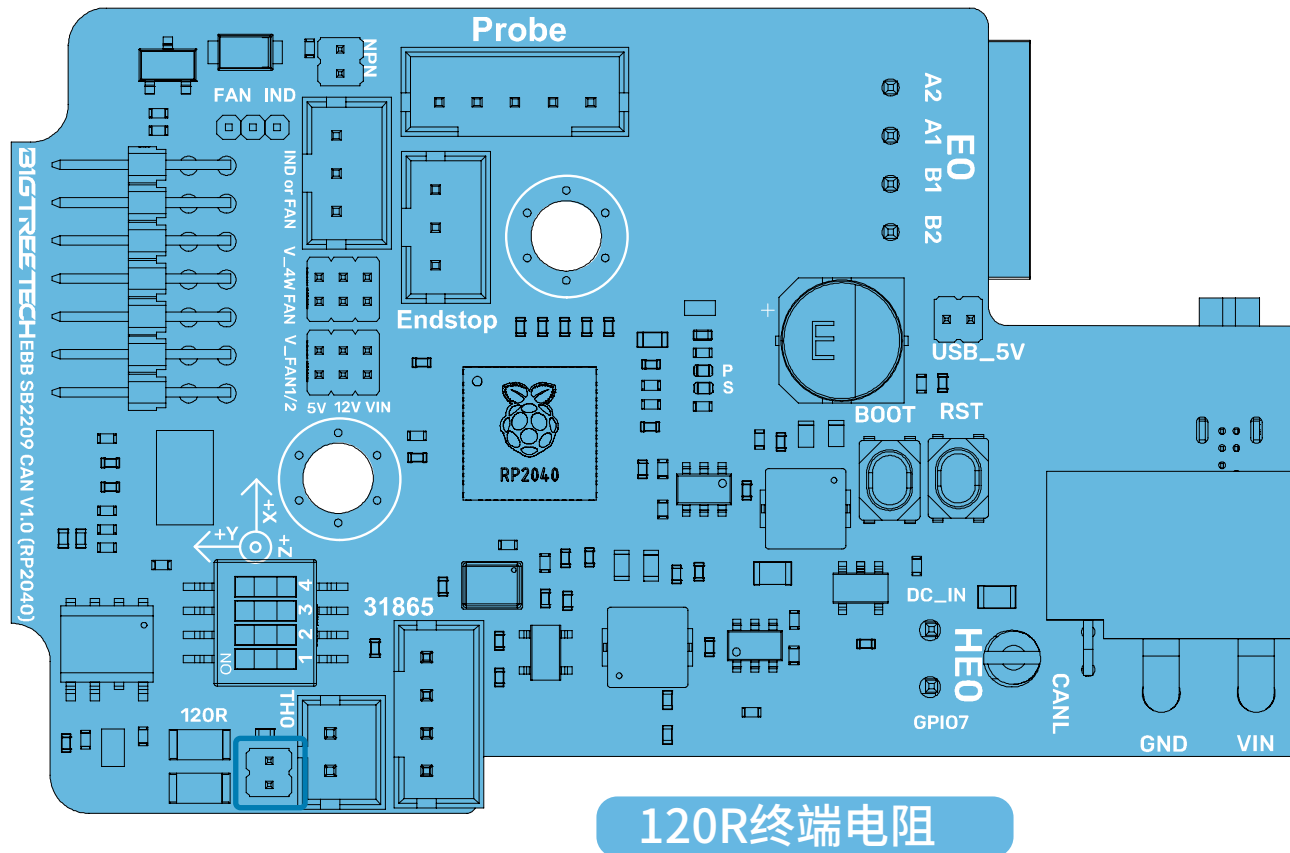
Cable_Cover_EBB 和 Main_Body_EBB 基于VORON官方打印件修改与官方打印件兼容但略有不同且更便于接线。

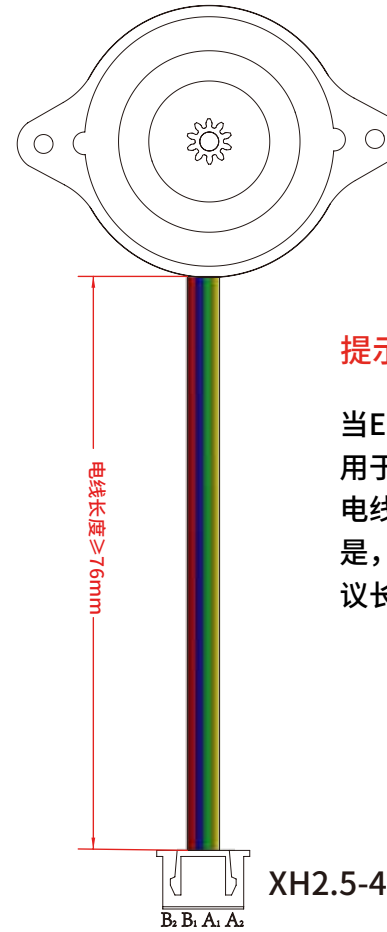
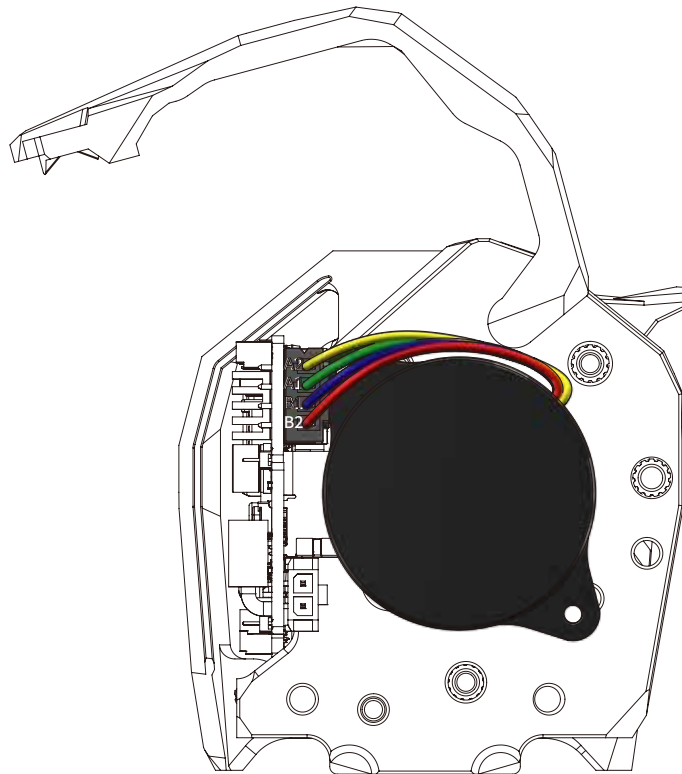


散热片

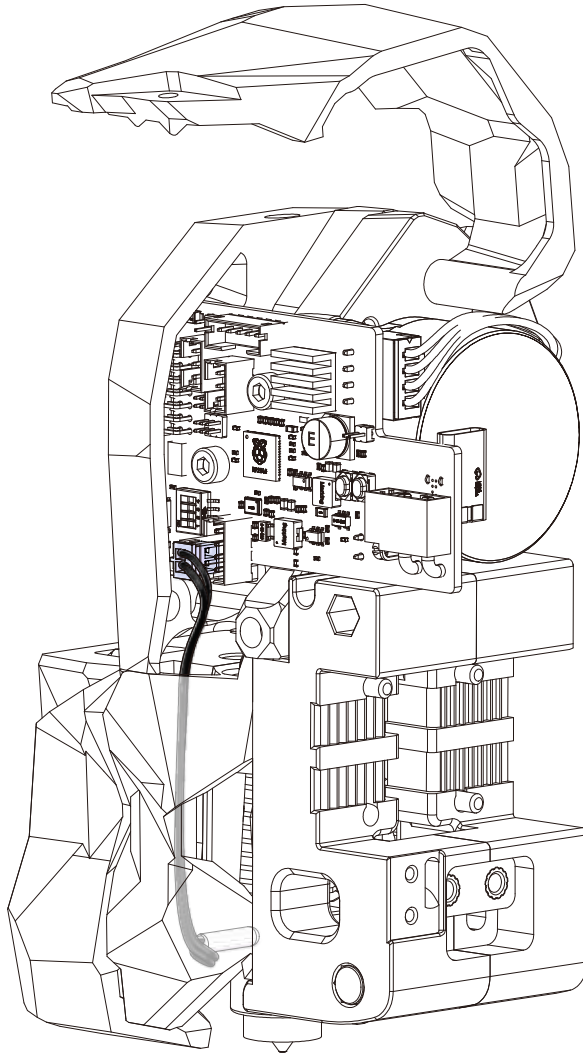
请按图示将散热片粘贴于图中所示位置，即在驱动芯片的背面。务必注意，安装散热片时避免接触到电机的连接器引脚。

EBB SB2209 CAN V1.0 (RP2040) 使用CAN bus通信时，如果作为终端必须将120R位置插上跳线帽。



**提示**

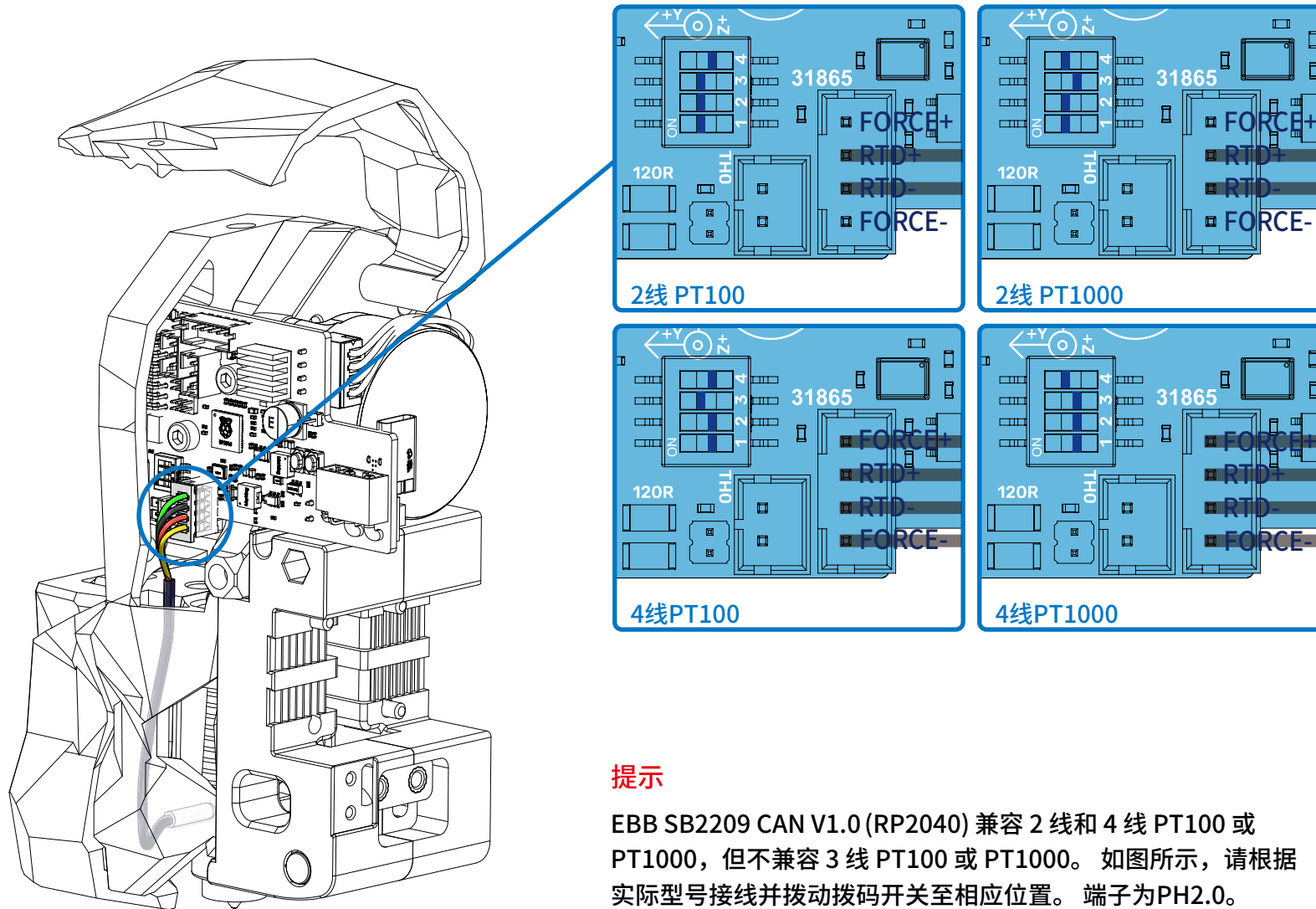
当EBB SB2209 CAN V1.0 (RP2040) 用于StealthBurner时，挤出机电机电线长度大于等于76mm即可。但是，为了后续接线和组装方便，建议长度最好为100mm或110mm。

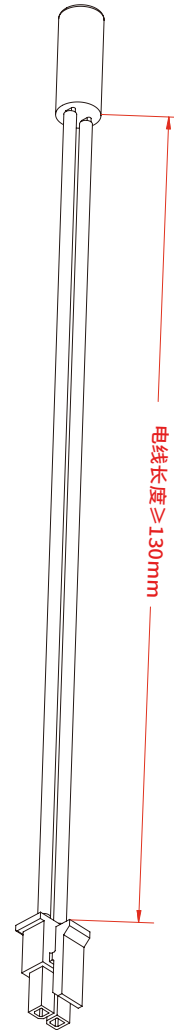
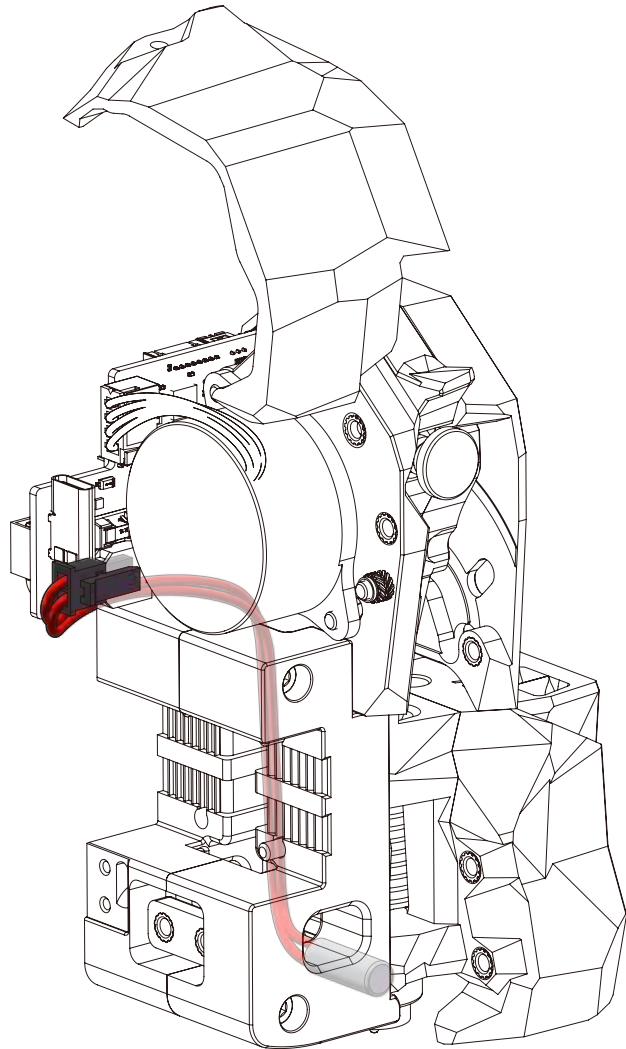


PH2.0-2PIN

提示

当EBB SB2209 CAN V1.0 (RP2040) 用于StealthBurner时，热敏电阻电线长度大于等于103mm即可。但是，为了后续接线和组装方便，建议长度最好为120mm或130mm。



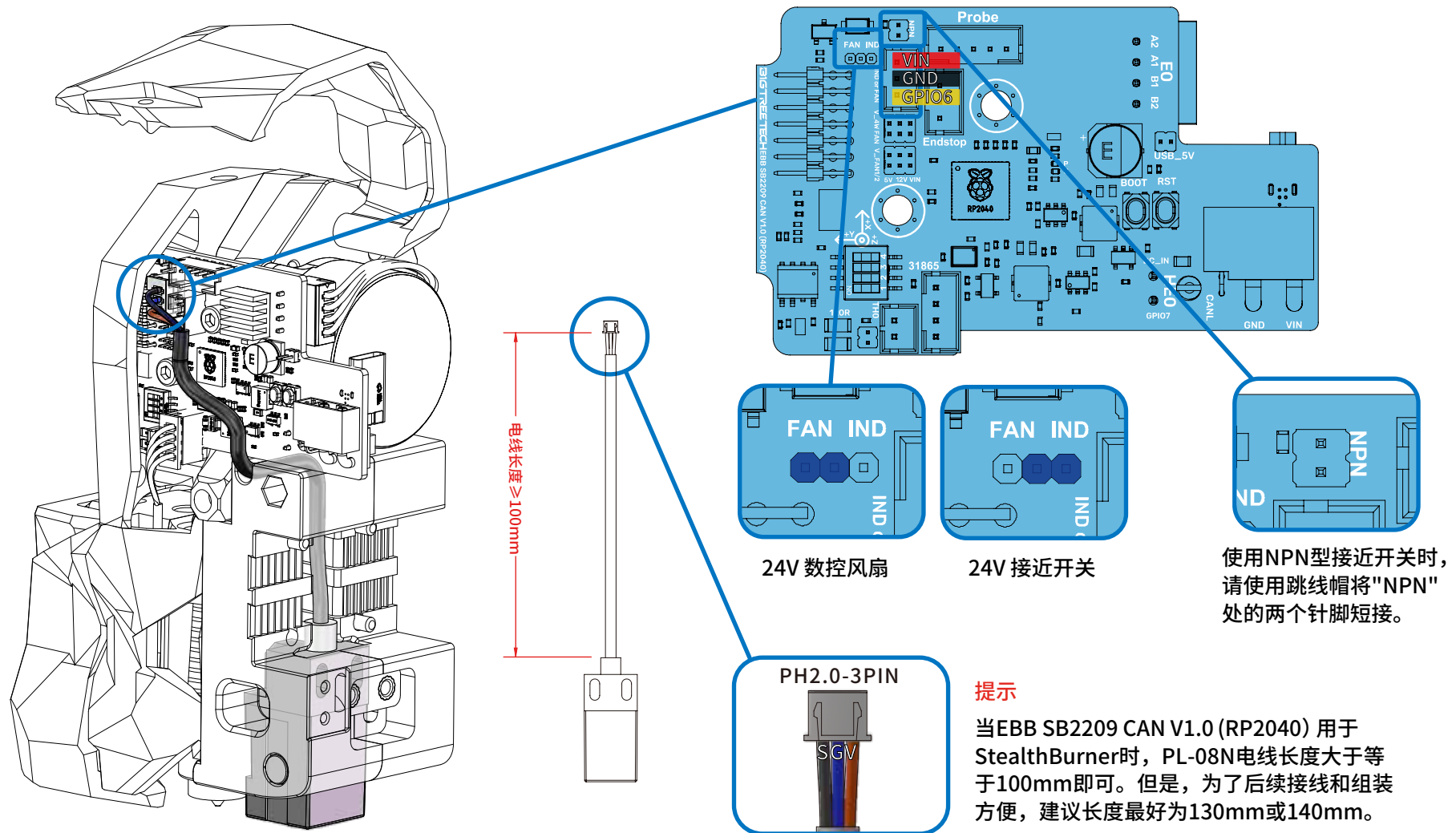


电线长度 $\geq 130\text{mm}$

MX3.0-2PIN

提示

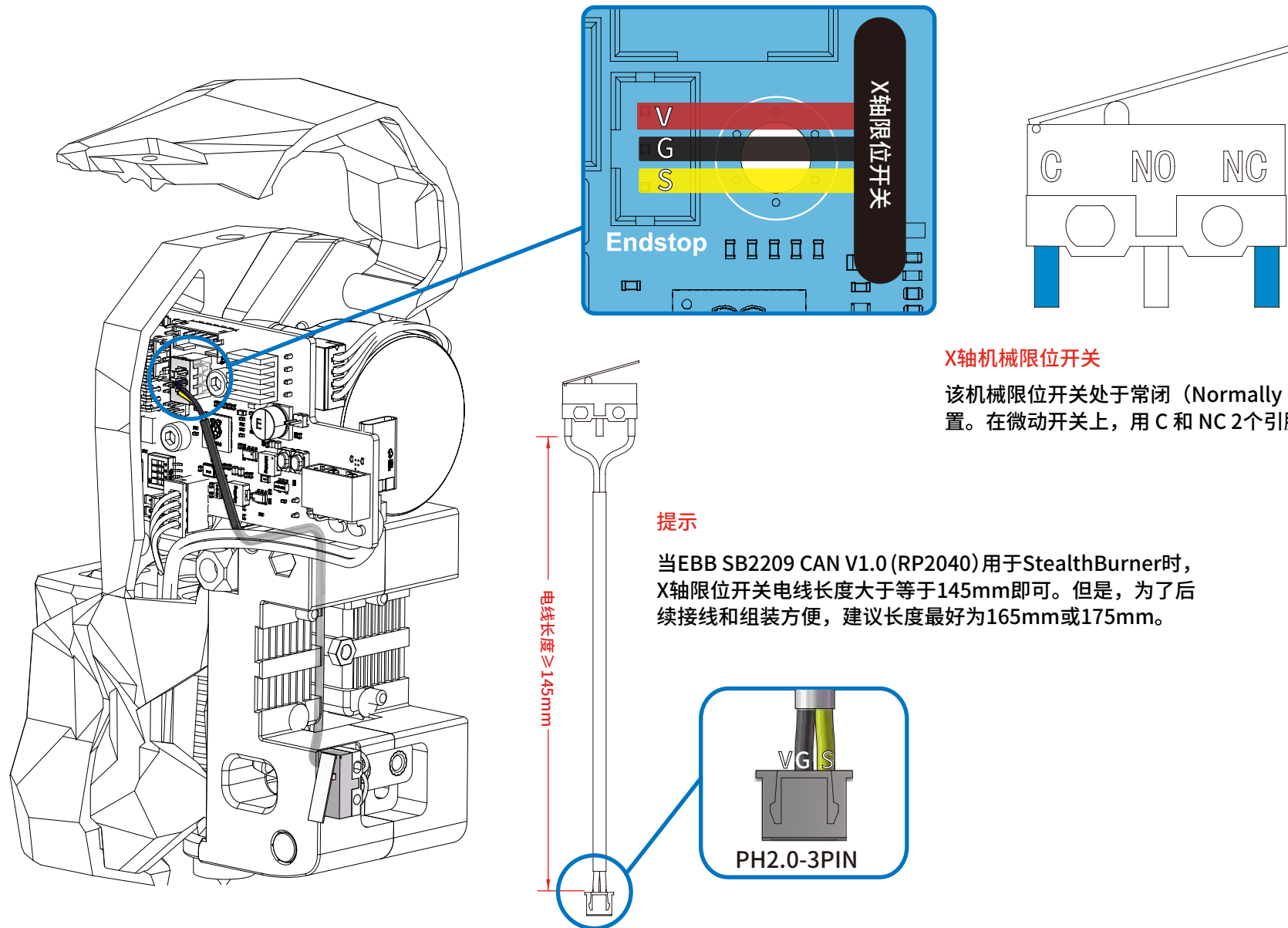
当EBB SB2209 CAN V1.0 (RP2040) 用于StealthBurner时，加热棒电线长度大于等于130mm即可。但是，为了后续接线和组装方便，建议长度最好为150mm或160mm。

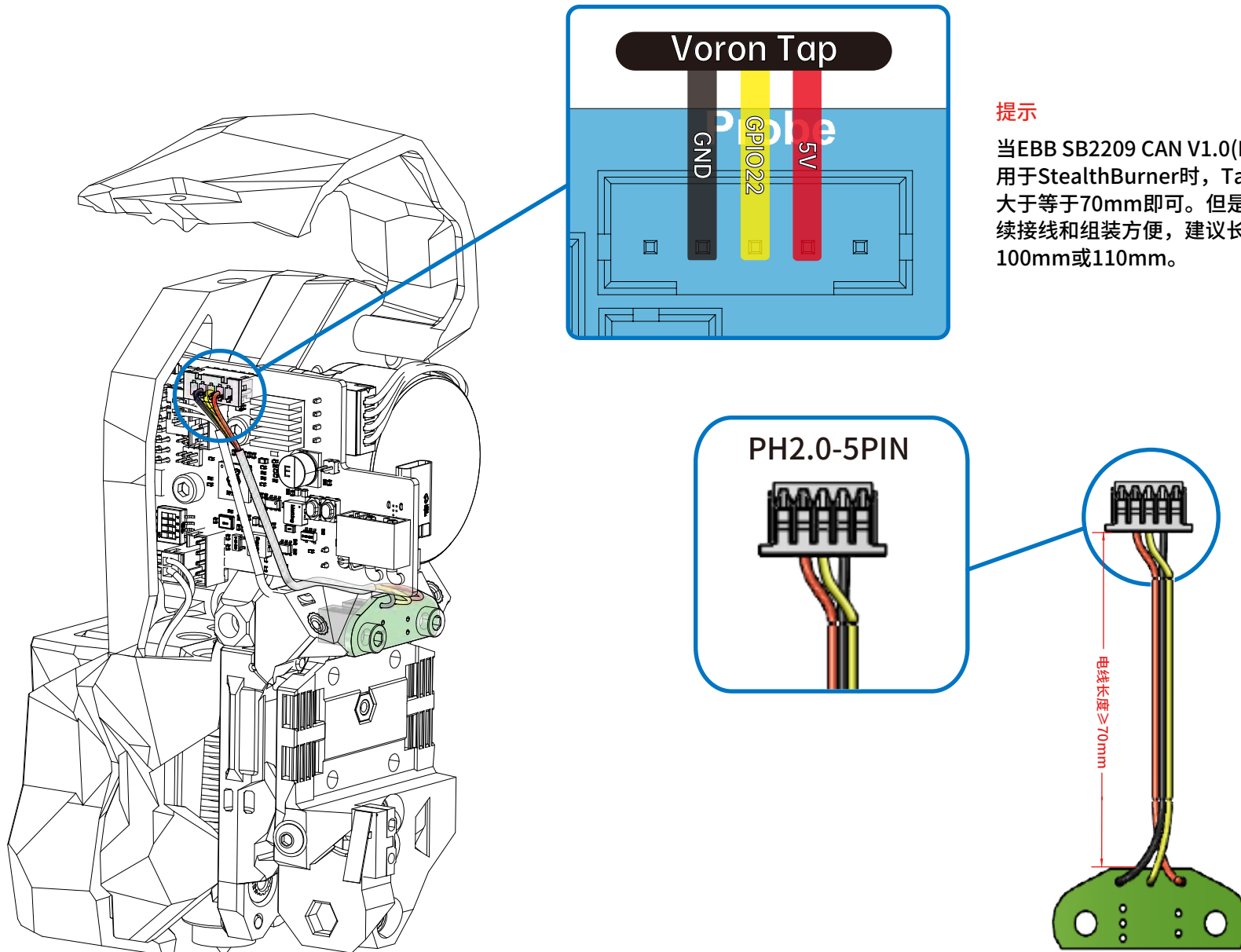


接近开关

支持NPN型和PNP型接近开关。对于NPN型接近开关, 请使用跳线帽将"NPN"处的两个针脚短接; 而对于PNP型接近开关, 则需要移除跳线帽。

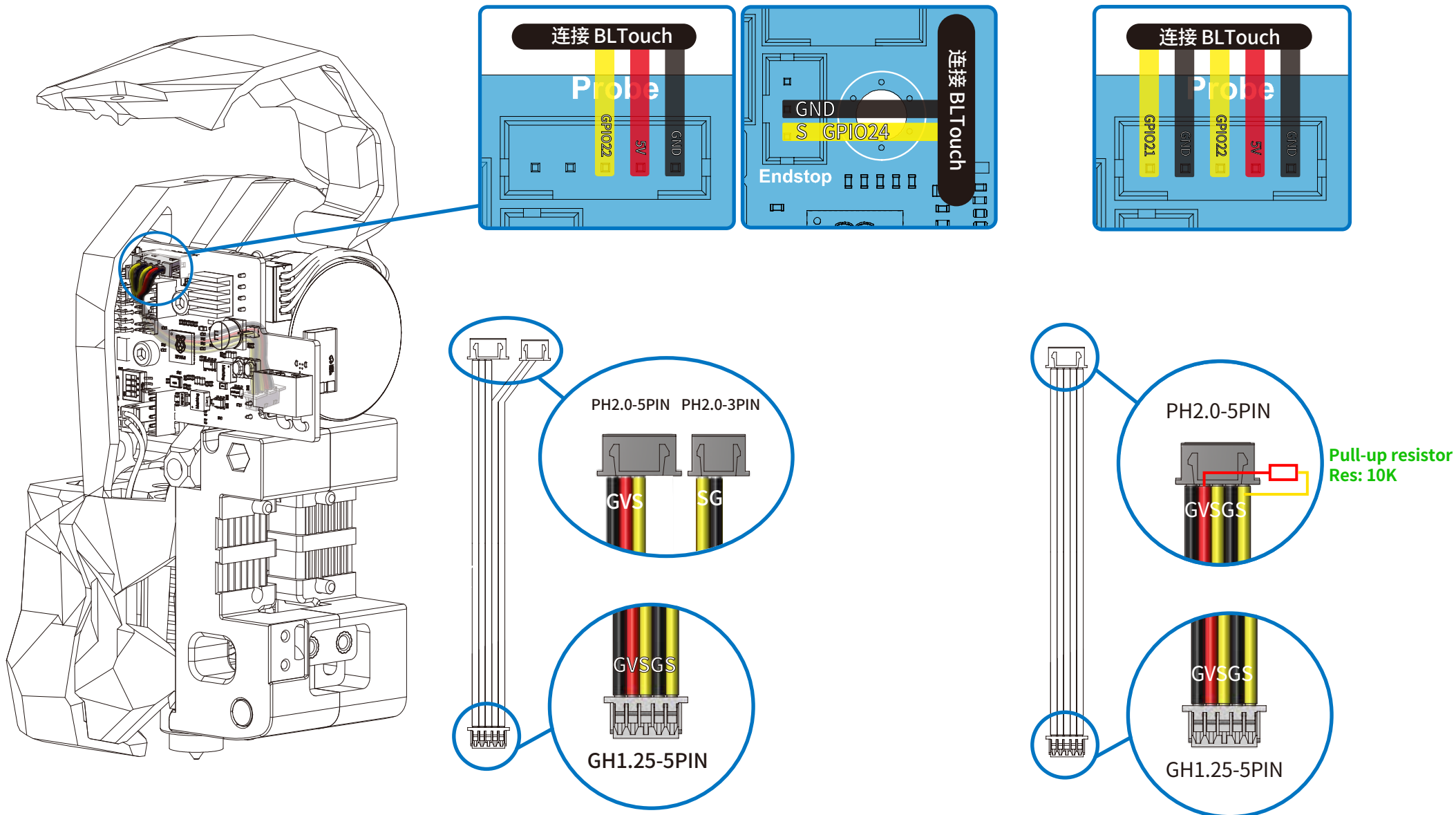
注: 该接口亦可为24V数控风扇使用, 当左侧二脚插入跳线帽(FAN), 该接口可为24V数控风扇使用; 当右侧二脚插入跳线帽(IND), 该接口可为24V接近开关使用。

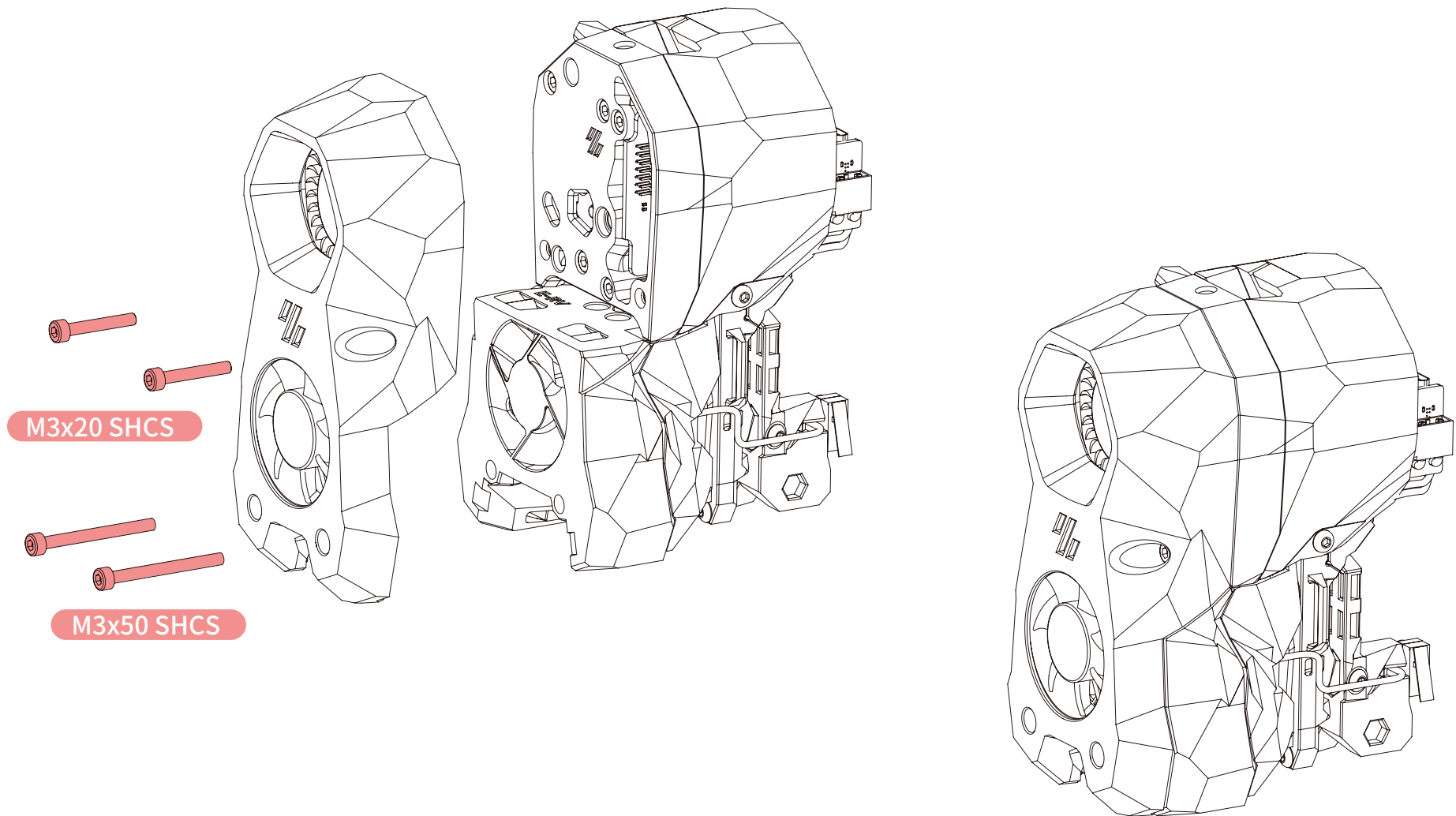


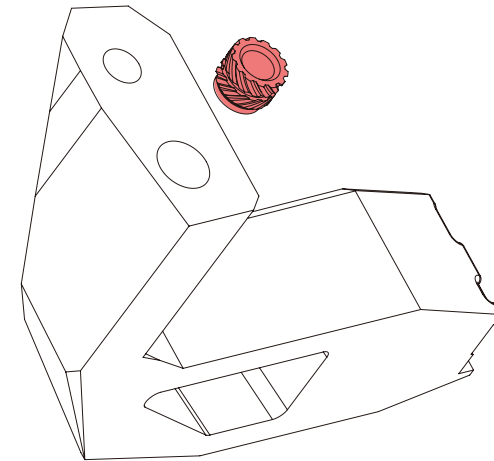
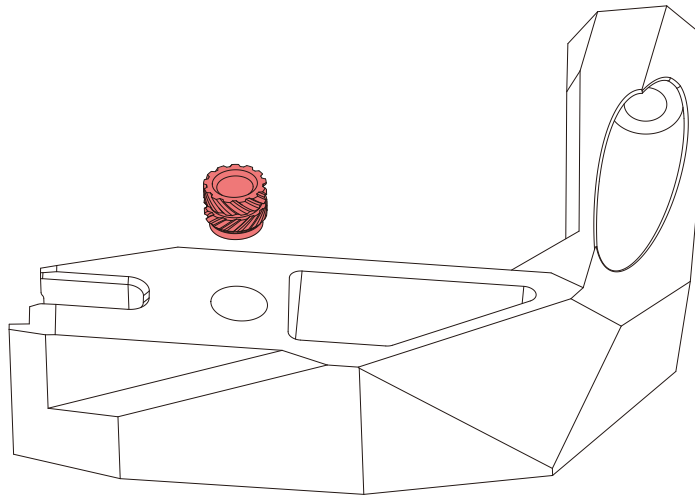


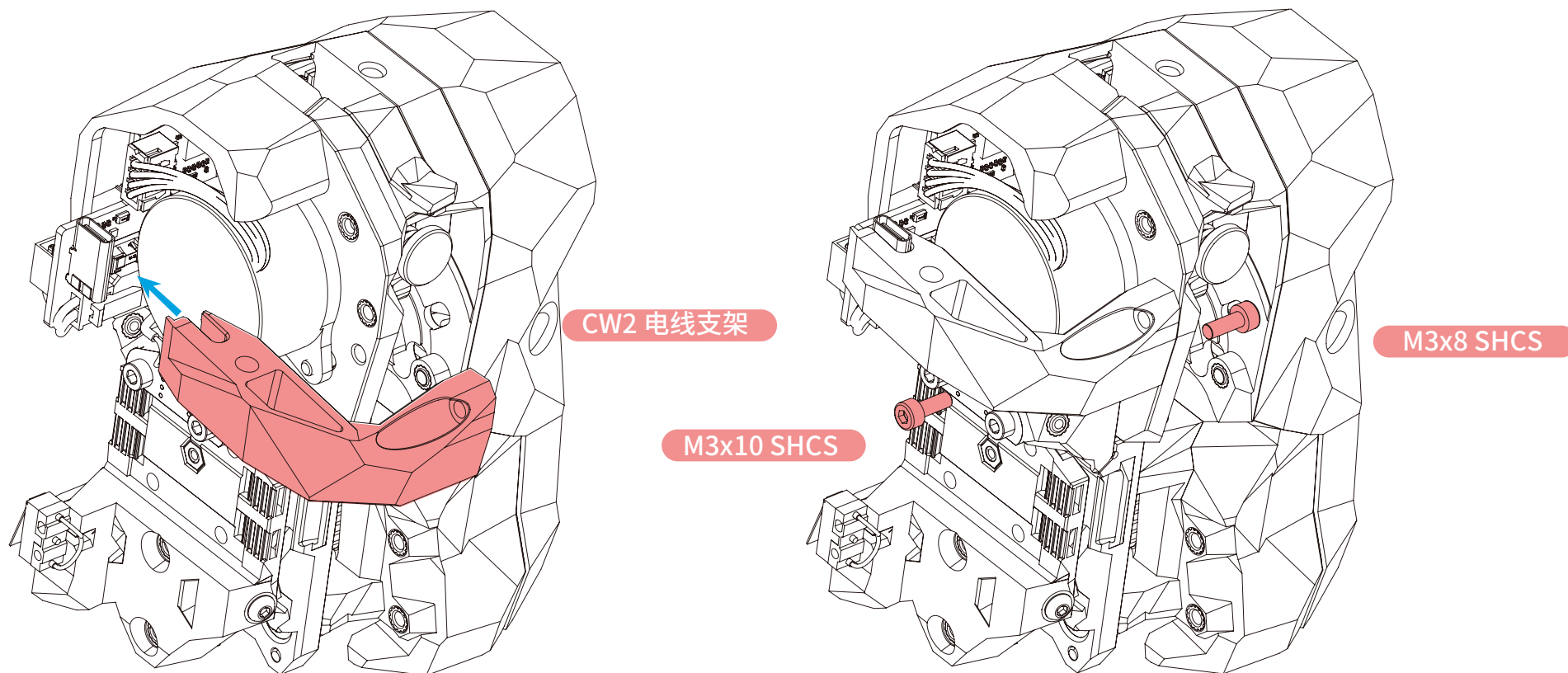
方法一：

方法二：



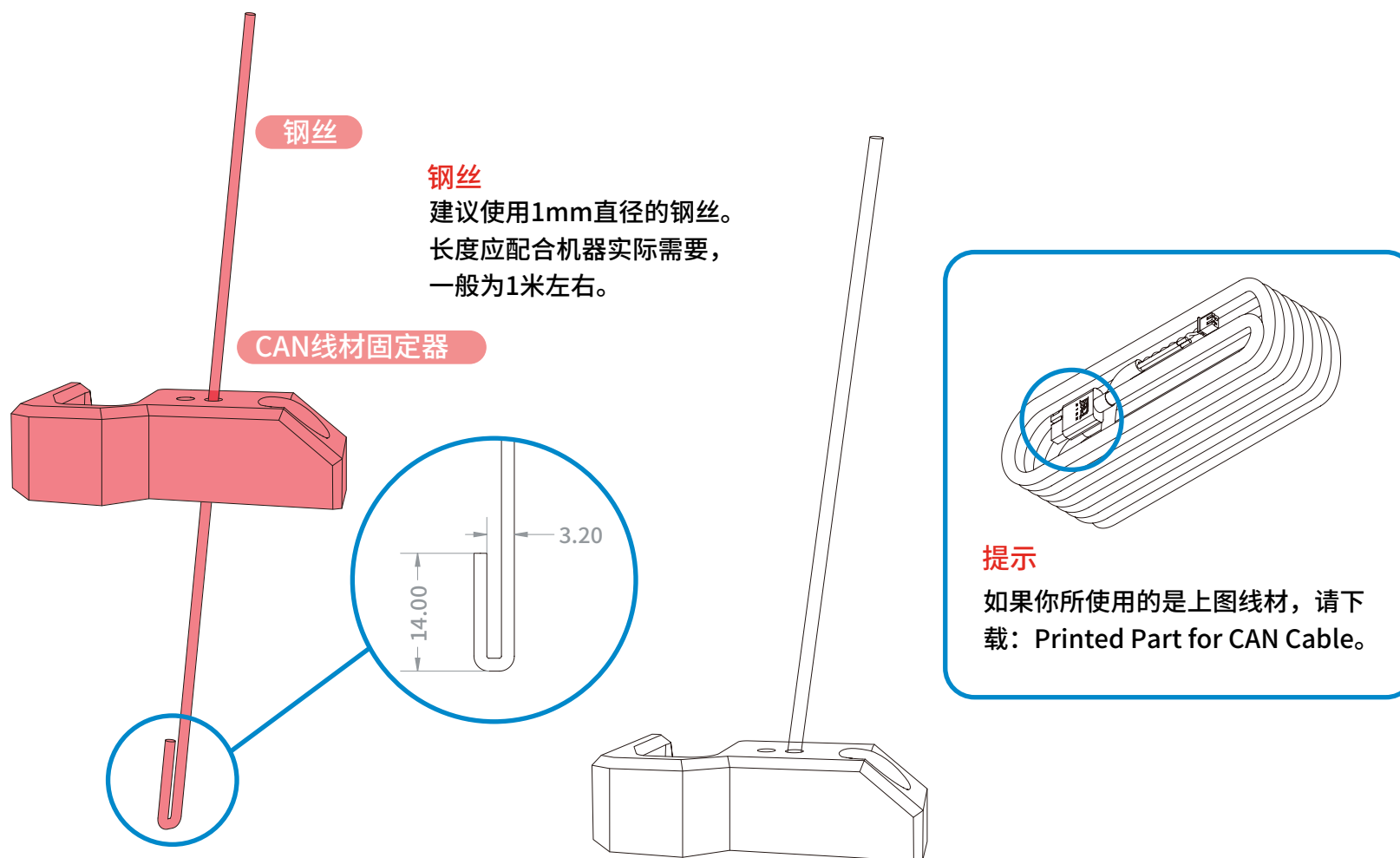






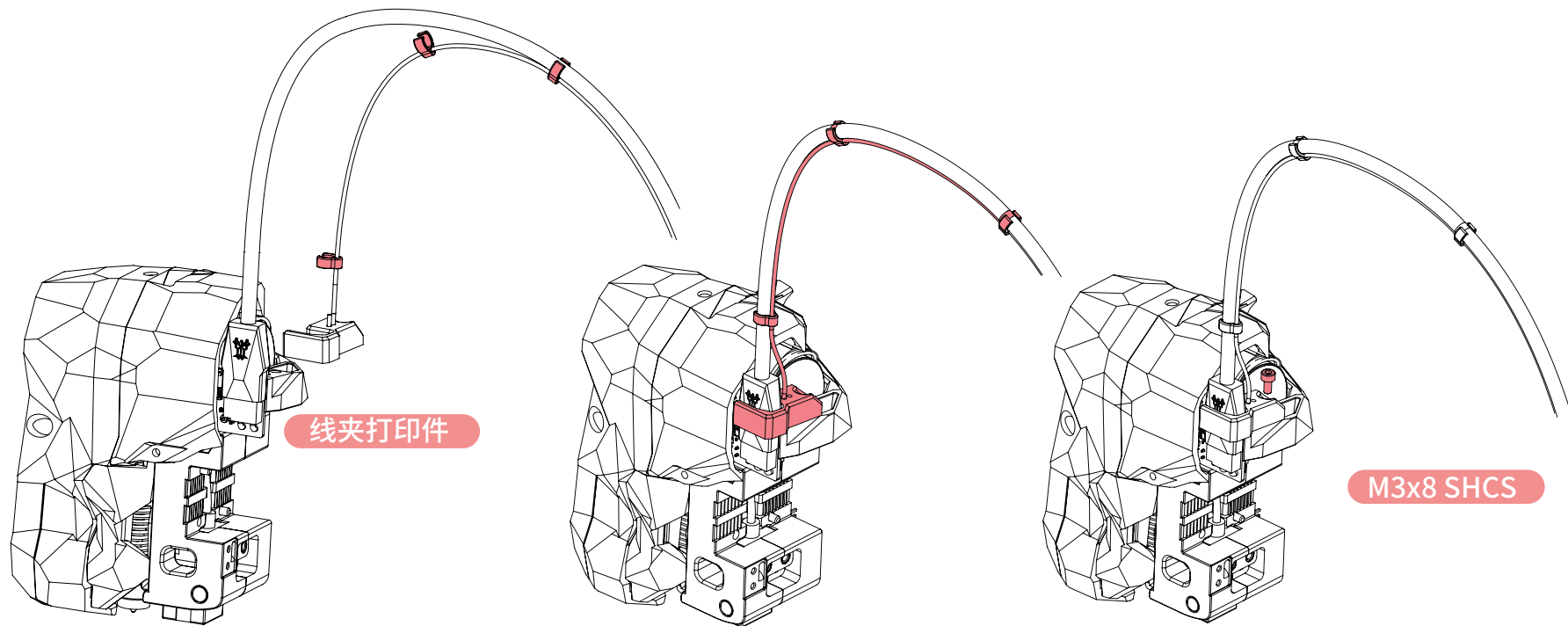
提示

CW2电线支架(CW2 Cable Bridge)与CW原版略有不同, 可在BIGTRETECH GitHub的EBB\EBB SB2240_2209 CAN\CAD 或 STL 目录中找到并下载打印。



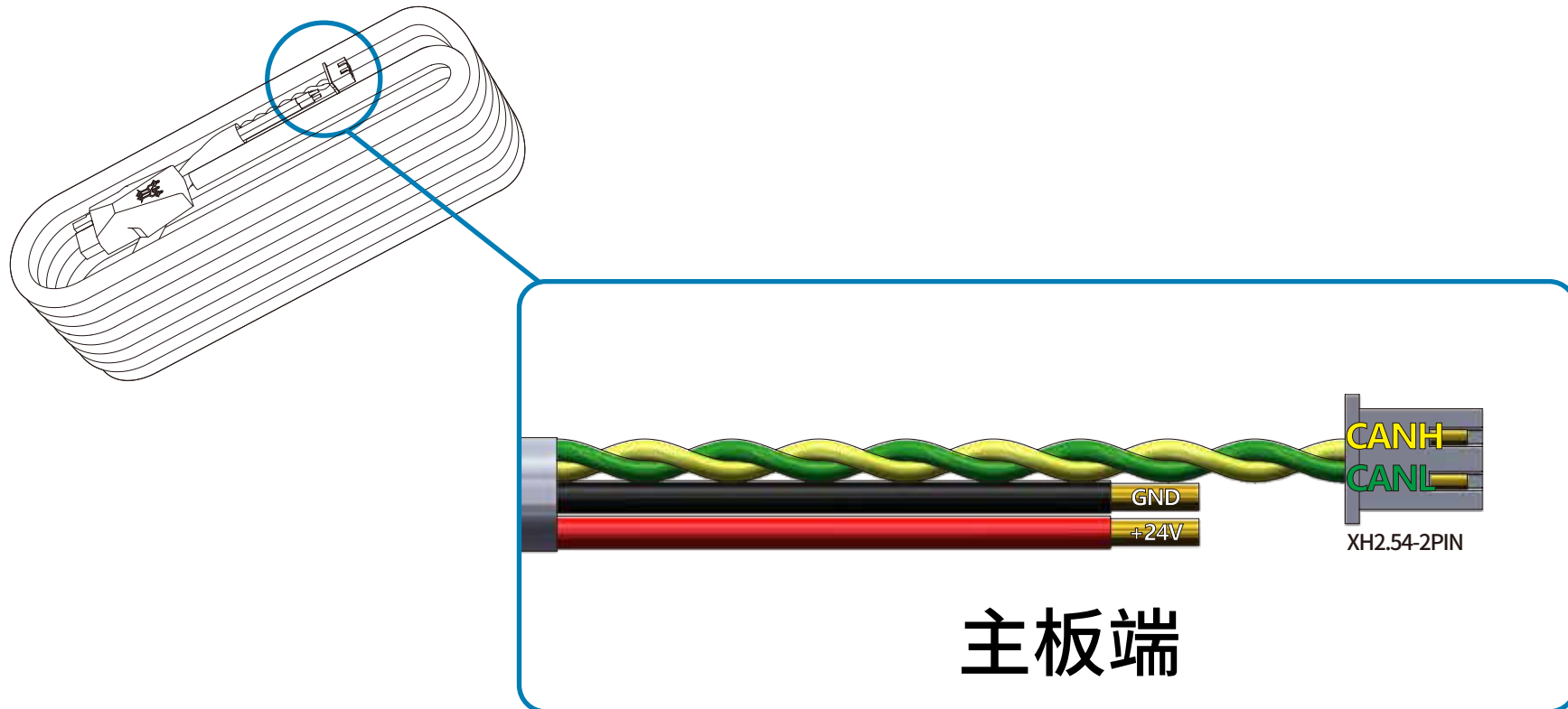
提示

CAN/USB-C线材固定器 (Printed Part for CAN Cable, Printed Part for USB-C Cable) 乃定制打印件，可在 BIGTREETECH GitHub的EBB\EBB SB2240_2209 CAN\CAD 或 STL 目录中找到并下载打印。



线夹

线夹打印件(Cable Clip Printed Part)乃定制打印件，可在BIGTREETECH GitHub的EBB\EBB SB2240_2209 CAN\CAD或 STL 目录中找到并下载打印。或者，也可以使用扎带固定线材。



之前被称为「CanBoot」的引导加载程序软件已更名为「Katapult」。

注意: Katapult旨在通过 CAN bus 接口直接更新 MCU 固件。

若您更倾向于使用 DFU 更新方法, 请跳过此步骤。

“树莓派或CB1烧录Katapult”

参考此处说明下载Katapult工程

<https://github.com/Arksine/Katapult>

1. 输入

```
cd ~
```

跳转到主目录, 输入

```
git clone https://github.com/Arksine/Katapult
```

下载Katapult工程,

输入

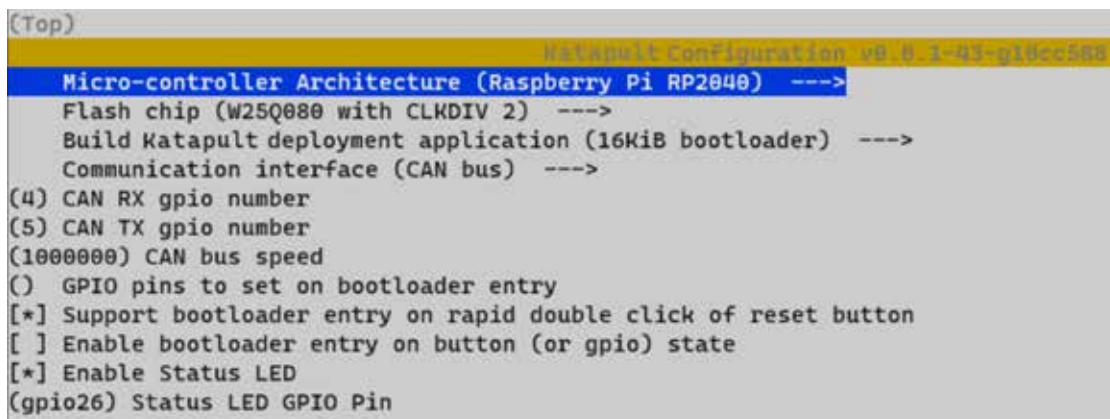
```
cd Katapult
```

跳转到Katapult目录中。

2. 输入

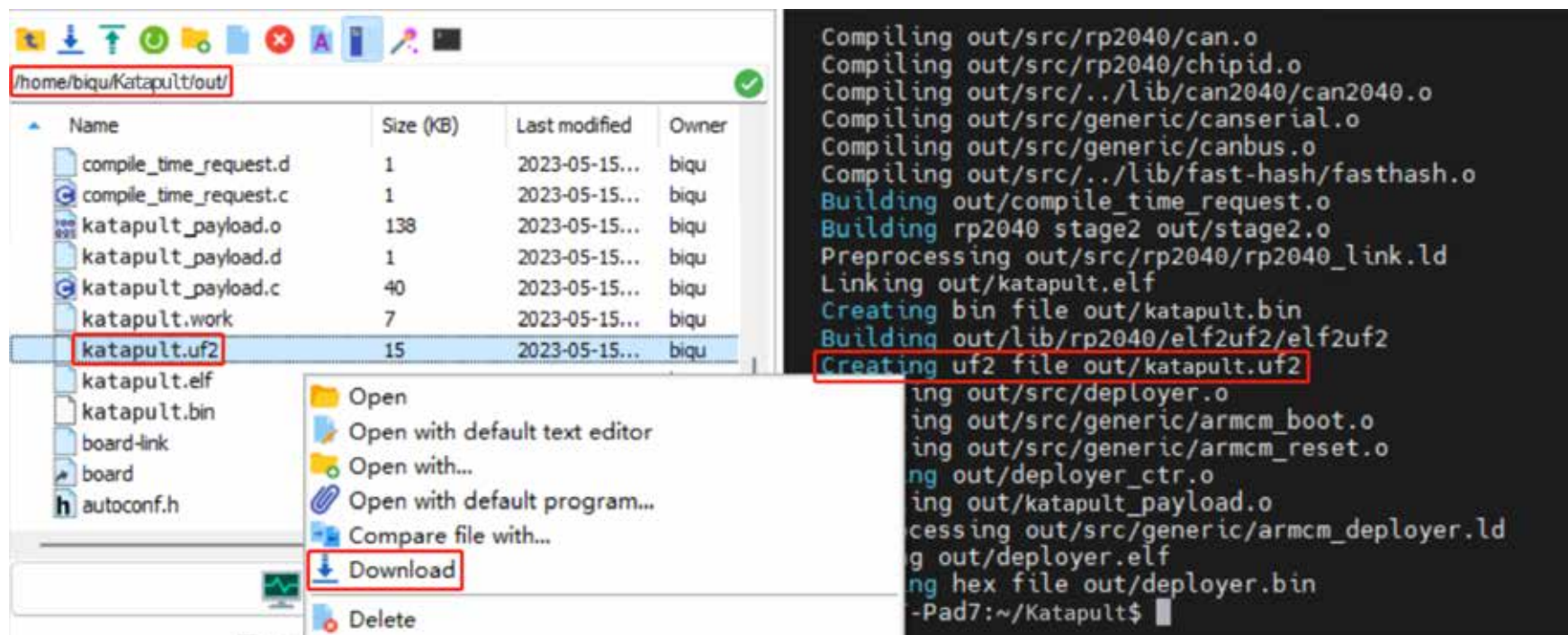
```
make menuconfig
```

并按照下图配置

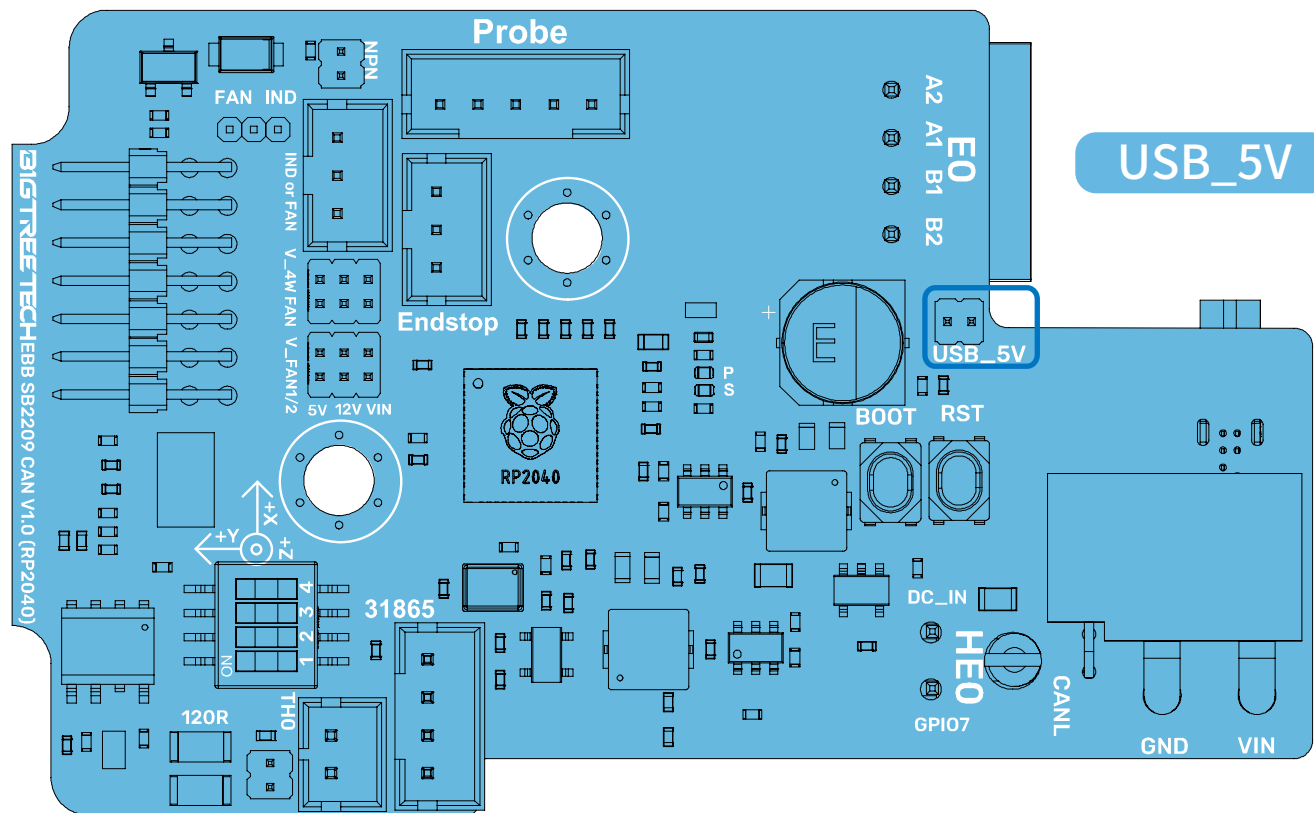


```
(Top)
Katapult Configuration v0.0.1-43-g10cc508
Micro-controller Architecture (Raspberry Pi RP2040) ---->
Flash chip (W25Q080 with CLKDIV 2) ---->
Build Katapult deployment application (16KiB bootloader) ---->
Communication interface (CAN bus) ---->
(4) CAN RX gpio number
(5) CAN TX gpio number
(1000000) CAN bus speed
() GPIO pins to set on bootloader entry
[*] Support bootloader entry on rapid double click of reset button
[ ] Enable bootloader entry on button (or gpio) state
[*] Enable Status LED
(gpio26) Status LED GPIO Pin
```

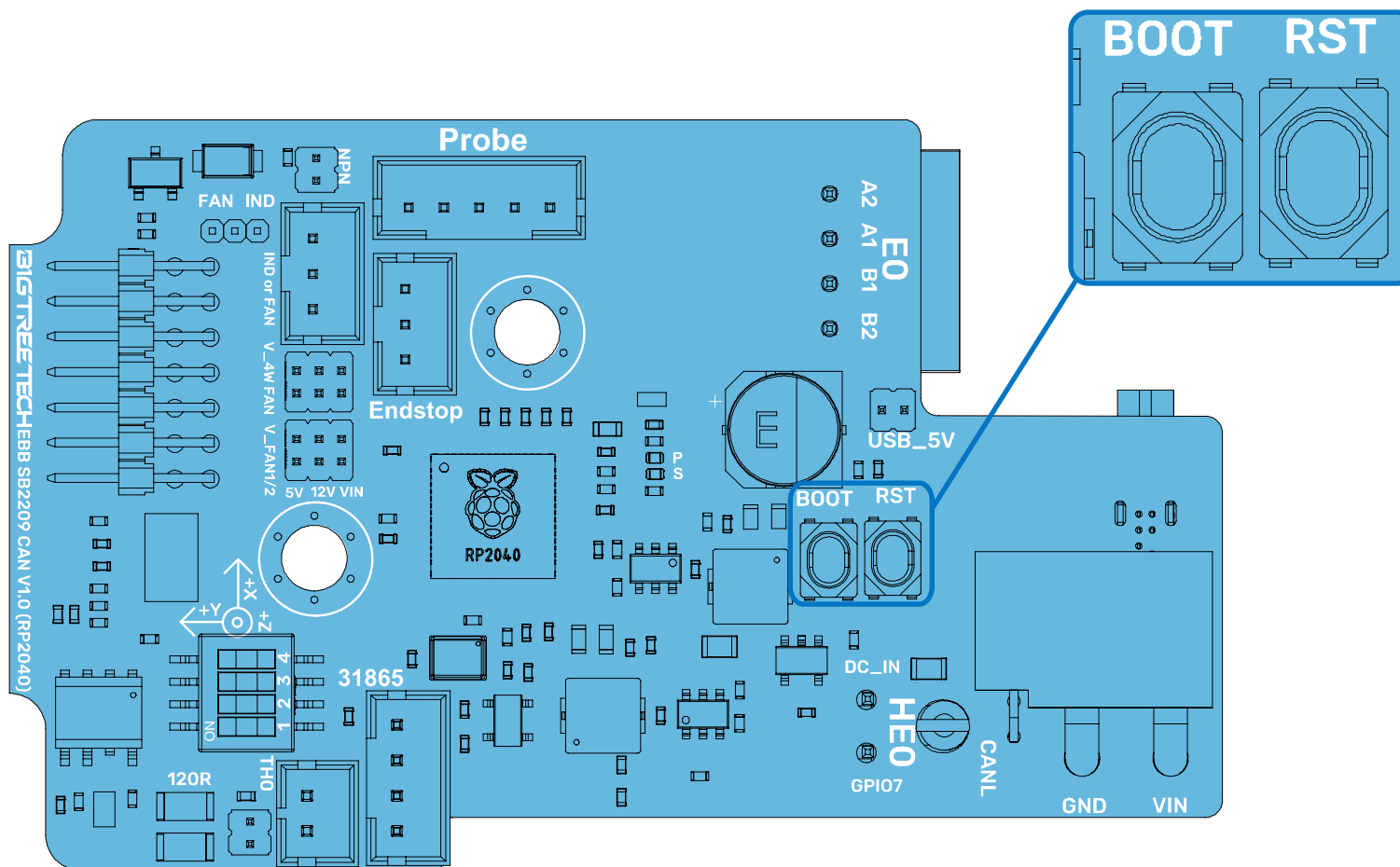
3. 输入make编译固件，当make执行完成后会在 `home/biqu/Katapult/out` 文件夹中生成我们所需要的 ‘katapult.uf2’ 固件，在SSH软件左侧可以直接下载到电脑中；



4.1. 请使用Type-C连接线将 EBB SB2209 CAN V1.0 (RP2040) 连接至树莓派/CB1，并确保已连接USB_5V跳线帽，以便通过Type-C为EBB SB2209 CAN V1.0 (RP2040) 提供电源。



4.2. 按住Boot按钮，然后单击一下Reset按钮进入DFU模式；



4.3. 在SSH终端命令行中输入

```
lsusb
```

查询 DFU设备ID

```
pi@fluidpi: ~$ lsusb
Bus 001 Device 005: ID 2e8a:0003 Raspberry Pi RP2 Boot
Bus 001 Device 004: ID 1d50:6061 OpenMoko, Inc. Geschwister Schneider CAN adapter
Bus 001 Device 003: ID 0424:0c00 Microchip Technology, Inc. (formerly SMSC) SMC9512/9514 Fast Ethernet Adapter
Bus 001 Device 002: ID 0424:9514 Microchip Technology, Inc. (formerly SMSC) SMC9514 Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
pi@fluidpi: ~$
```

4.4 请输入以下命令以烧录Katapult:

```
make flash FLASH_DEVICE=2e8a:0003
```

其中“2e8a:0003”需替换为第4.3步中查询到的实际设备ID。

4.5 烧录完成后，请拔下 USB_5V 跳线帽和Type-C数据线。

1.) SSH连接到树莓派后，在命令行输入：

```
cd ~/klipper/
make menuconfig
```

使用下面的配置编译固件(如果没有下列选项，请更新Klipper固件源码到最新版本)；

```
[*] Enable extra low-level configuration options
Micro-controller Architecture (Raspberry Pi RP2040) --->
```

如果不使用Katapult

```
Bootloader offset (No bootloader) --->
```

```
Flash chip (W25Q080 with CLKDIV 2) --->
```

如果使用Katapult

```
Bootloader offset (16KiB bootloader) --->
```

如果使用Type-C上的USB通信

```
Communication interface (USB) --->
```

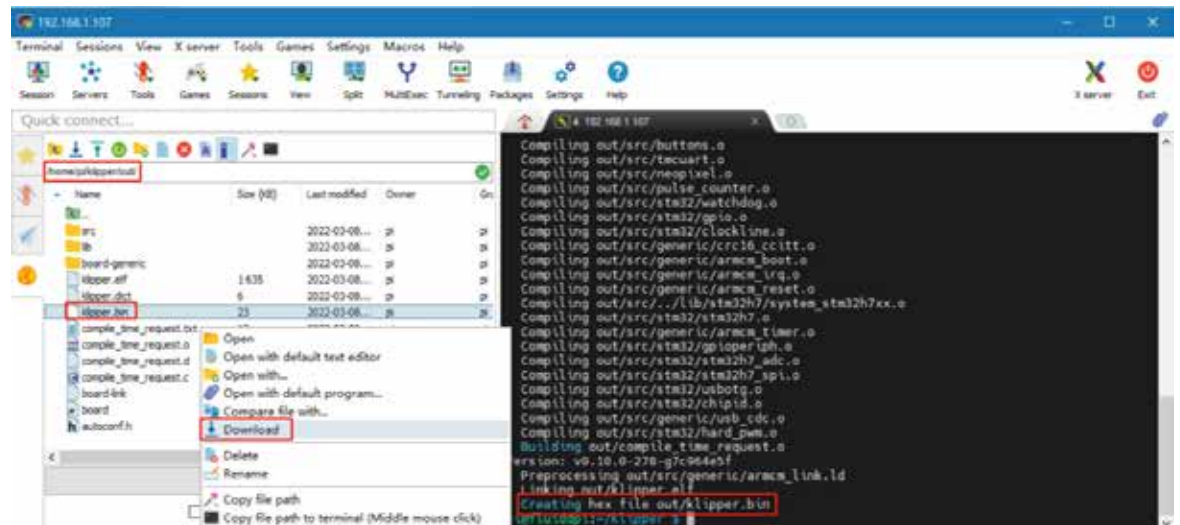
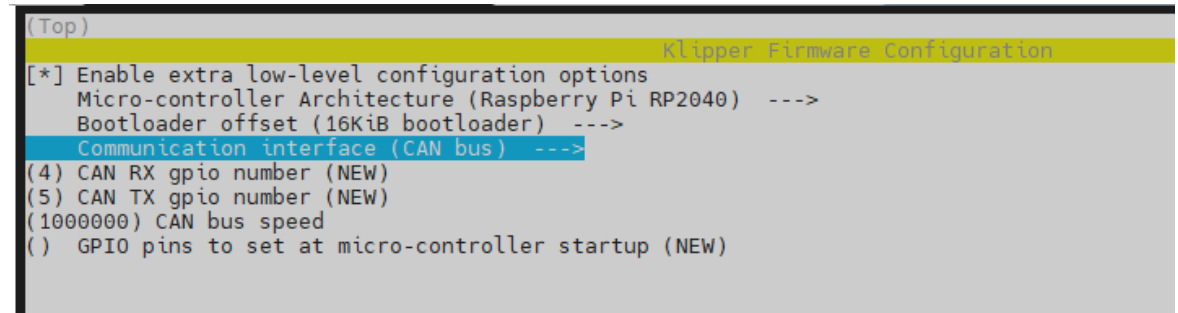
如果使用CANBus通信

```
Communication interface (CAN bus) --->
```

```
(4) CAN RX gpio number
```

```
(5) CAN TX gpio number
```

```
(1000000) CAN bus speed
```



2.) 配置选择完成后，输入 ‘q’ 退出配置界面，当询问是否保存配置时选择 “Yes” ；

3.) 输入 **make** 编译固件，当 **make** 执行完成后会在 **home/pi/klipper/out** 文件夹中生成我们所需要的 ‘klipper.bin’ 固件，在SSH软件左侧可以直接下载到电脑中。

1.使用CAN bus需要接好CAN bus线缆以及插上120R终端电阻的跳线帽。

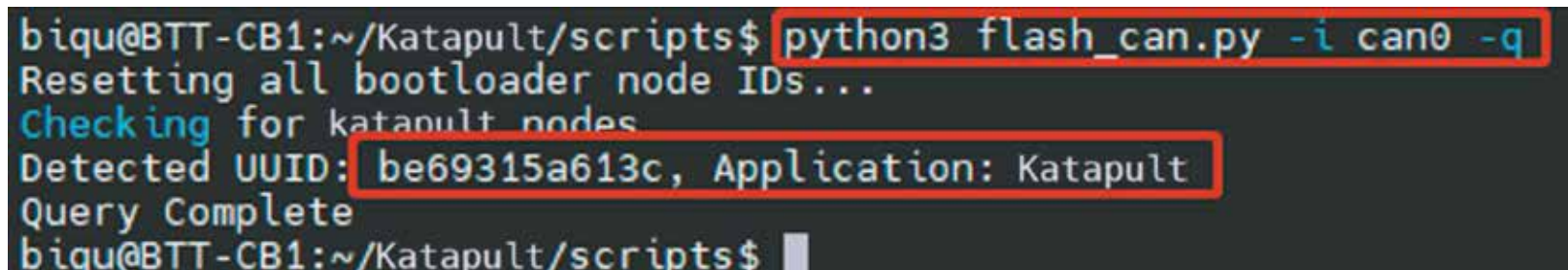
2. 输入

```
cd ~/Katapult/scripts
```

然后输入

```
python3 flash_can.py -i can0 -q
```

查询canbus ID（需提前接好CAN线并通电），
如下图已找到设备的UUID。



```
biqu@BTT-CB1:~/Katapult/scripts$ python3 flash_can.py -i can0 -q
Resetting all bootloader node IDs...
Checking for katapult nodes
Detected UUID: be69315a613c, Application: Katapult
Query Complete
biqu@BTT-CB1:~/Katapult/scripts$
```

3. 输入

```
python3 flash_can.py -i can0 -f ~/klipper/out/klipper.bin -u be69315a613c
```

替换为实际的UUID，注意：klipper.bin需要提前make生成出来，并且Katapult的Application start offset 为16KiB offset，所以Klipper的menuconfig中Bootloader offset 也要为16KiB bootloader，如下图已经烧录成功。

```
biqu@BTT-CB1:~/Katapult/scripts$ python3 flash_can.py -i can0 -f ~/klipper/out/klipper.bin -u be69315a613c
Sending bootloader jump command...
Resetting all bootloader node IDs...
Checking for katapult nodes...
Detected UUID: be69315a613c, Application:Katapult
Attempting to connect to bootloader
KatapultConnected
Protocol Version: 1.0.0
Block Size: 64 bytes
Application Start: 0x8002000
MCU type: stm32g0b1xx
Verifying canbus connection
Flashing '/home/biqu/klipper/out/klipper.bin'...

[#####]

Write complete: 13 pages
Verifying (block count = 414)...

[#####]

Verification Complete: SHA = C3B1F96A8FCE706587BF4A9119D95D80465875A3
CAN Flash Success
biqu@BTT-CB1:~/Katapult/scripts$
```

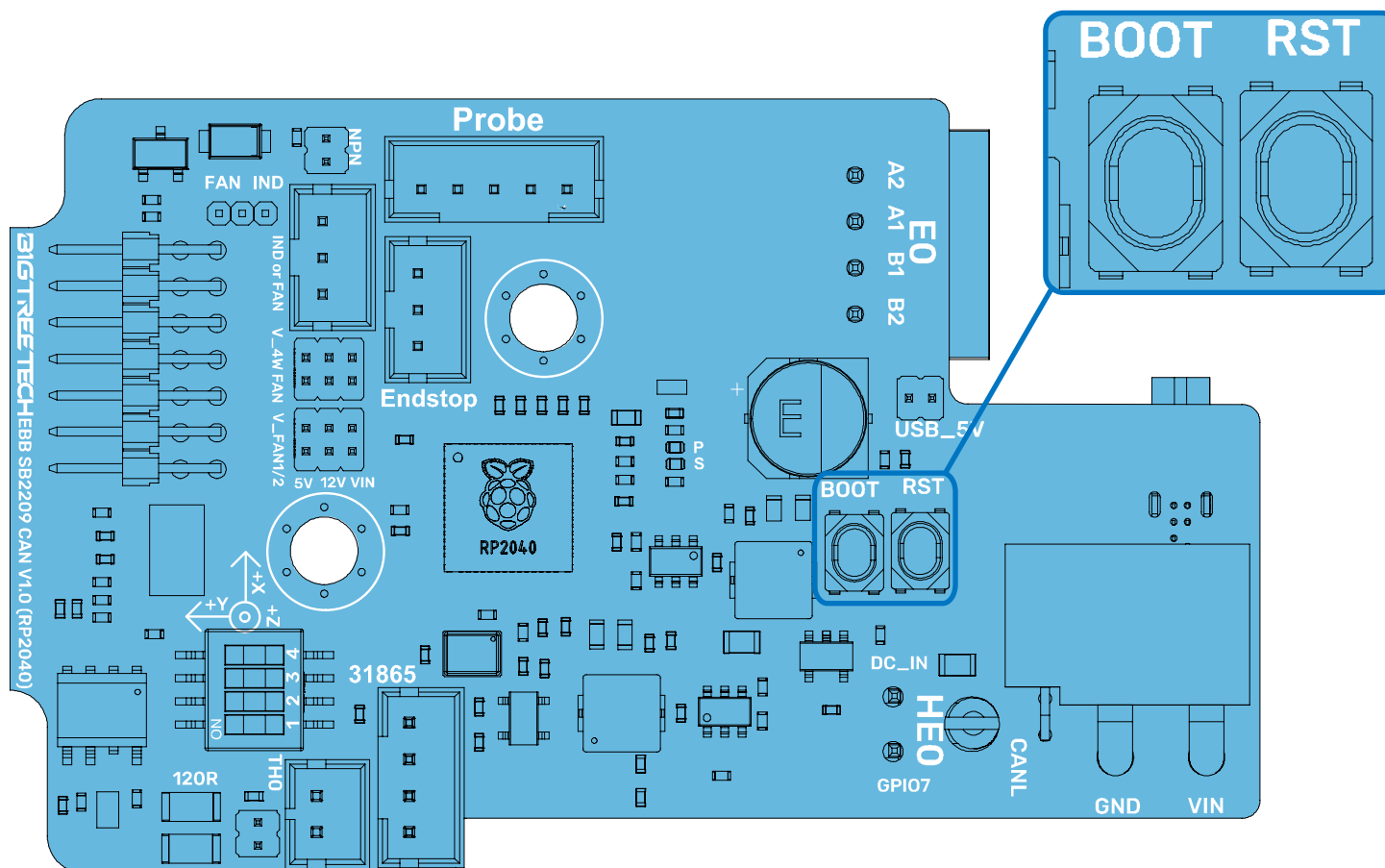
4. 再次输入

```
python3 flash_can.py -i can0 -q
```

查询，此时Application由之前的Katapult变为Klipper，代表Klipper已经正常运行。

```
biqu@BTT-CB1:~/Katapult/scripts$ python3 flash_can.py -i can0 -q
Resetting all bootloader node IDs...
Checking for katapult nodes...
Detected UUID: be69315a613c, Application: Klipper
Query Complete
biqu@BTT-CB1:~/Katapult/scripts$
```


2. 按住Boot按钮，然后单击一下Reset按钮进入DFU模式；



3.在SSH终端命令行中输入

```
lsusb
```

查询 DFU设备ID

```
pi@fluidpi: ~$ lsusb
Bus 001 Device 005: ID 2e8a:0003 Raspberry Pi RP2 Boot
Bus 001 Device 004: ID 1d50:6061 OpenMoko, Inc. Geschwister Schneider CAN adapter
Bus 001 Device 003: ID 0424:0c00 Microchip Technology, Inc. ( formerly SMSC ) SMC9512/9514 Fast Ethernet Adapter

Bus 001 Device 002: ID 0424:9514 Microchip Technology, Inc. ( formerly SMSC ) SMC9514 Hub
Bus 001 Device 001: ID 1d6b:0002 Linux Foundation 2.0 root hub
pi@fluidpi : ~$
```

4.输入

```
cd klipper
```

跳转到klipper目录下，输入

```
make flash FLASH_DEVICE=2e8a:0003
```

 开始烧录固件

(注意：将2e8a:0003更换为上一步中查询到的实际的设备ID)。

5. 固件烧录完成后，输入

```
ls /dev/serial/by-id/
```

查询设备的Serial ID（只有通过USB通信的方式才会有此ID，CANBus方式忽略此步骤）。

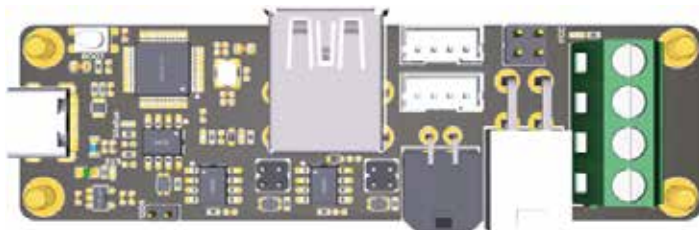
6. 如果使用USB通信，第一次烧录完成之后，再次更新时无需手动按Boot和Reset按钮进入DFU 模式，可以直接输入

```
make flash FLASH_DEVICE=/dev/serial/by-id/usb-Klipper_rp2040_4550357128922FC8-if00
```

烧录固件（注意：将/dev/serial/by-id/xxx更换为上一步中查询到的实际的ID）。

烧录完成后，请拔下 USB_5V 跳线帽和Type-C数据线。

搭配BIGTREETECH U2C 模块使用



1. 在SSH终端中输入

```
sudo nano /etc/network/interfaces.d/can0
```

命令并执行

```
allow-hotplug can0
iface can0 can static
    bitrate 1000000
up ifconfig $IFACE txqueuelen 1024
```

将 CAN bus 速度设置为 1M（必须与固件中设置的速度一致(1000000) CAN bus speed），修改后保存（Ctrl + S）并退出（Ctrl + X），输入

```
sudo reboot
```

重启树莓派。

2. CANBus上的每个设备都会根据MCU的UID生成一个 canbus_uuid，要查找每个微控制器设备 ID，请确保硬件已通电并正确接线，然后运行：

```
~/klippy-env/bin/python ~/klipper/scripts/canbus_query.py can0
```

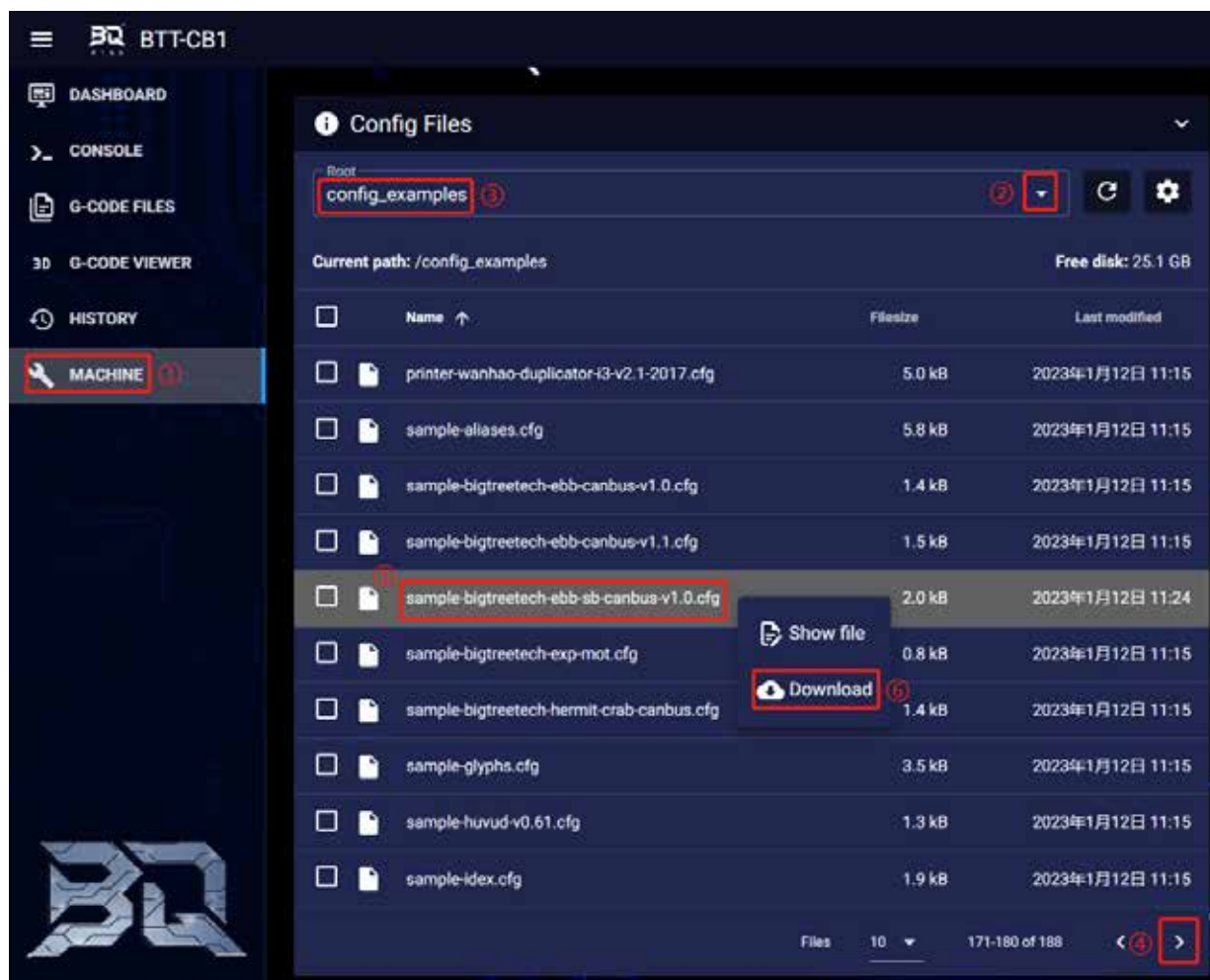
3. 如果检测到未初始化的 CAN 设备，上述命令将报告设备的 canbus_uuid

```
Found canbus_uuid=0e0d81e4210c
```

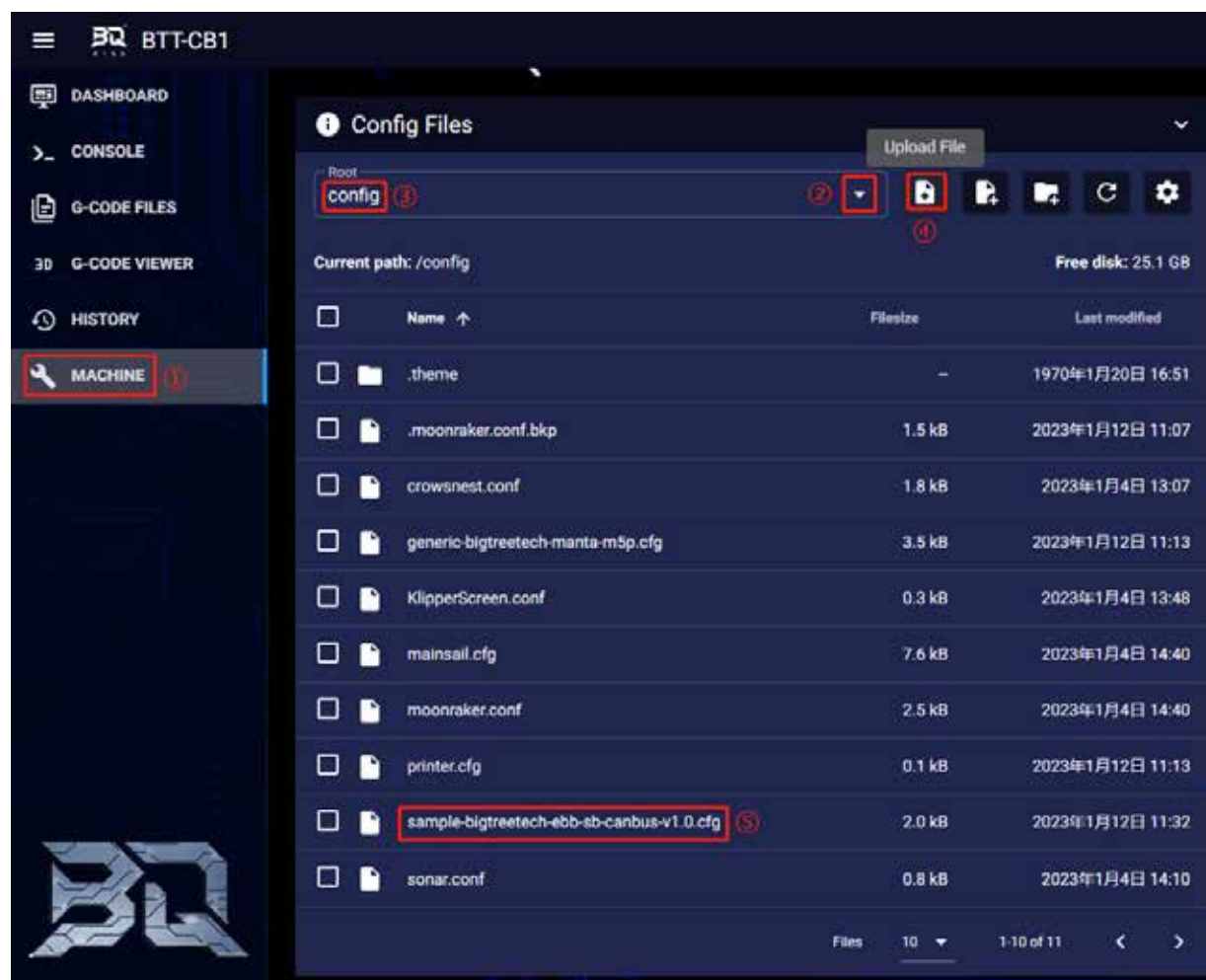
4. 如果 Klipper 已经正常运行并且连接到此设备，那么 canbus_uuid 将不会被上报，此为正常现象。

1. 在电脑的浏览器中输入树莓派的IP访问，如下图所示的路径中下载主板的参考配置，如果找不到此文件，请更新Klipper固件源码到最新版本，或者到GitHub下载：

<https://github.com/bigtree-tech/EBB>



2. 将主板的配置文件上传到Configuration Files中;



3. 并在“printer.cfg”文件中添加此主板的配置

```
[include sample-bigtreetech-ebb-sb-canbus-v1.0.cfg]
```

```
X      printer.cfg
11
12      [include sample-bigtreetech-ebb-canbus-v1.0.cfg]
13
```

4. 将配置文件中的ID号修改为主板实际的ID（USB serial 或者 canbus）

```
X      sample-bigtreetech-ebb-sb-canbus-v1.0.cfg
8      [ mcu EBBCan]
9      serial: /dev/serial/by-id/usb-Klipper_firmware_12345-if00
10     #canbus_uuid: 0e0d81e4210c
```

5. 按照下方链接的说明配置模块的具体功能：

```
https://www.klipper3d.org/Overview.html
```

Website

www.bigtree-tech.com

GitHub

www.github.com/bigtreetech

Discord

www.discord.gg/5jdwbyYZuv

