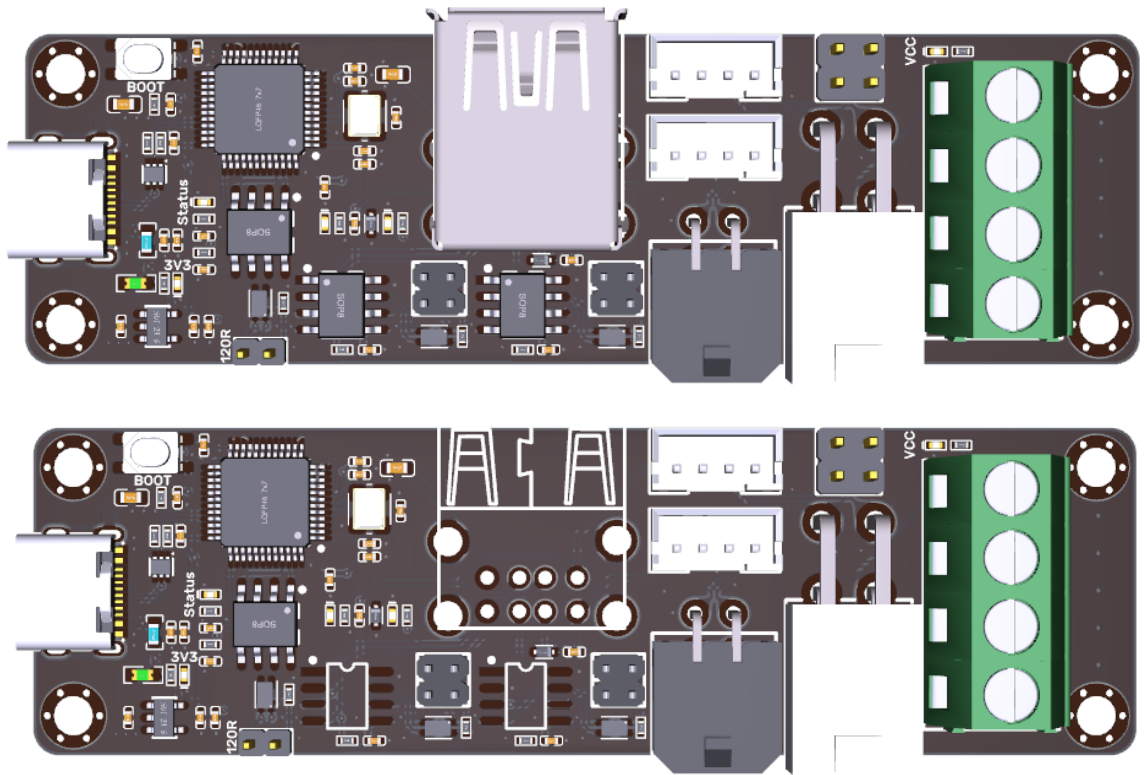


BIGTREETECH

U2C

使用说明



目录

目录	2
修订历史	3
一、产品简介	4
1.1 产品特点	4
1.2 产品尺寸	4
二、外设接口	5
2.1 Pinout	5
2.1 接口说明	6
三、 可选的接线图	7
四、图片展示	8
五、固件说明	10
5.1 固件源码	10
5.2 固件更新	10
5.3 Klipper 配置	10
六、注意事项	11

修订历史

版本	修改说明	日期
01.00	初稿	2022/04/23
01.01	添加更详细的 pinout 指示图	2022/11/25

一、产品简介

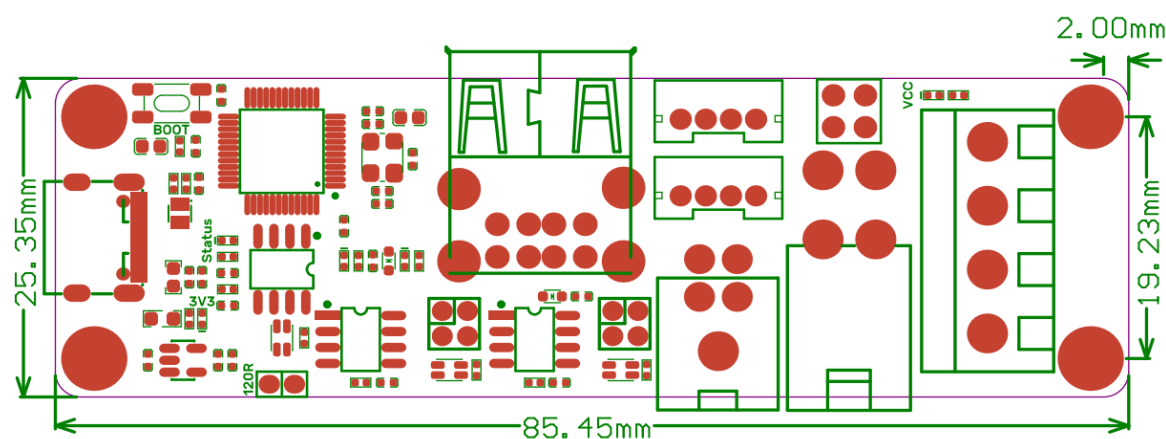
BIGTREETECH U2C 是一款 USB 转 CAN 总线的模块，可以通过它将 Raspberry Pi 的 USB 端口转为 CAN 总线，通过 CAN 总线连接 3D 打印主板、BIGTREETECH EBB35/36/42 CAN 等产品。板上具有多种不同形态的 CAN 接口，有助于连接到不同的板子上。

由于 CAN 总线在工业控制领域的诸多优势，所以我司推出了此 USB 转 CAN 模块。CAN 总线传输距离较远，并且总线上可以挂载大量的节点，所以主板、EBB、其他扩展板可以同时挂载到同一个 CAN 总线上。

1.1 产品特点

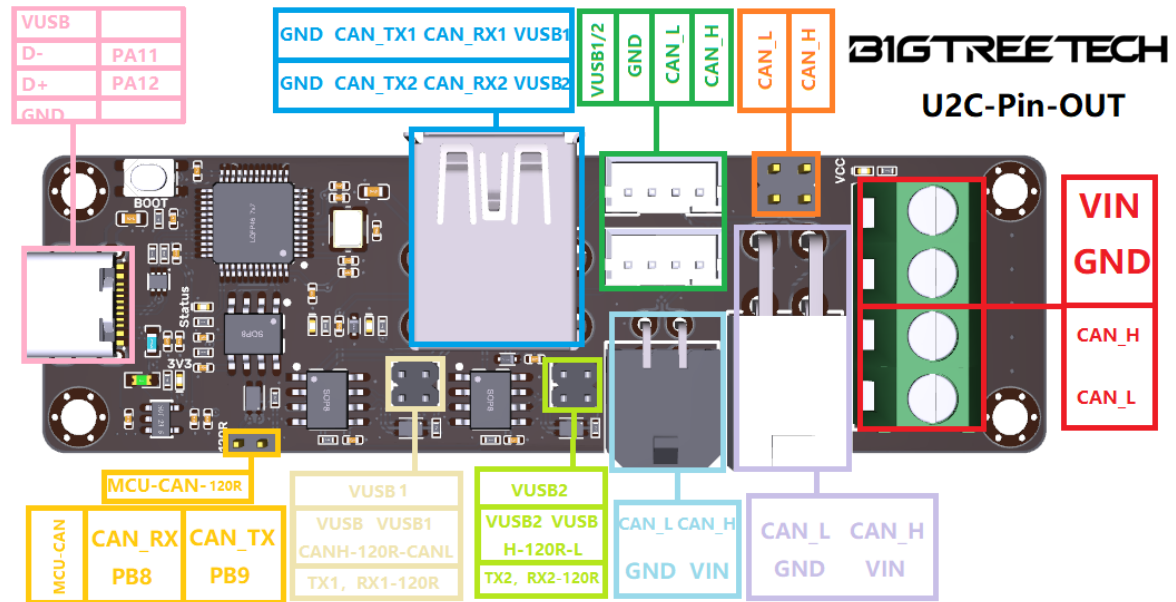
1. 支持 CAN 总线连接，数据传输较远、抗噪声能力强、实时性强、可靠性高
2. PCB 表面采用沉金工艺
3. TYPE-C 口 ESD 保护，防止静电损坏硬件
4. 支持通过 SWD 或者硬件 DFU 方式给主控更新固件
5. 丰富的 CAN 接口

1.2 产品尺寸

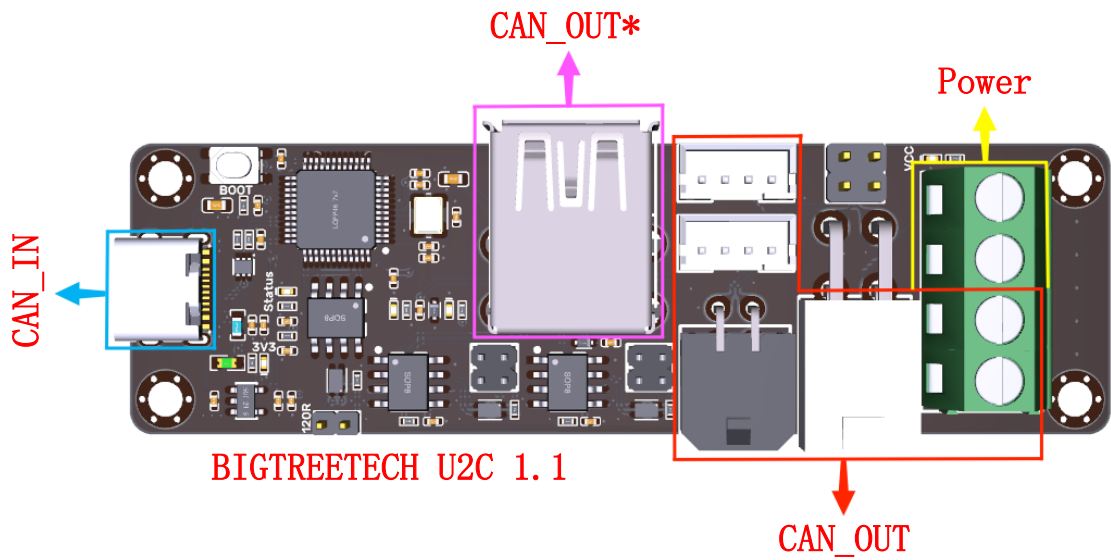


二、外设接口

2.1 Pinout



2.1 接口说明

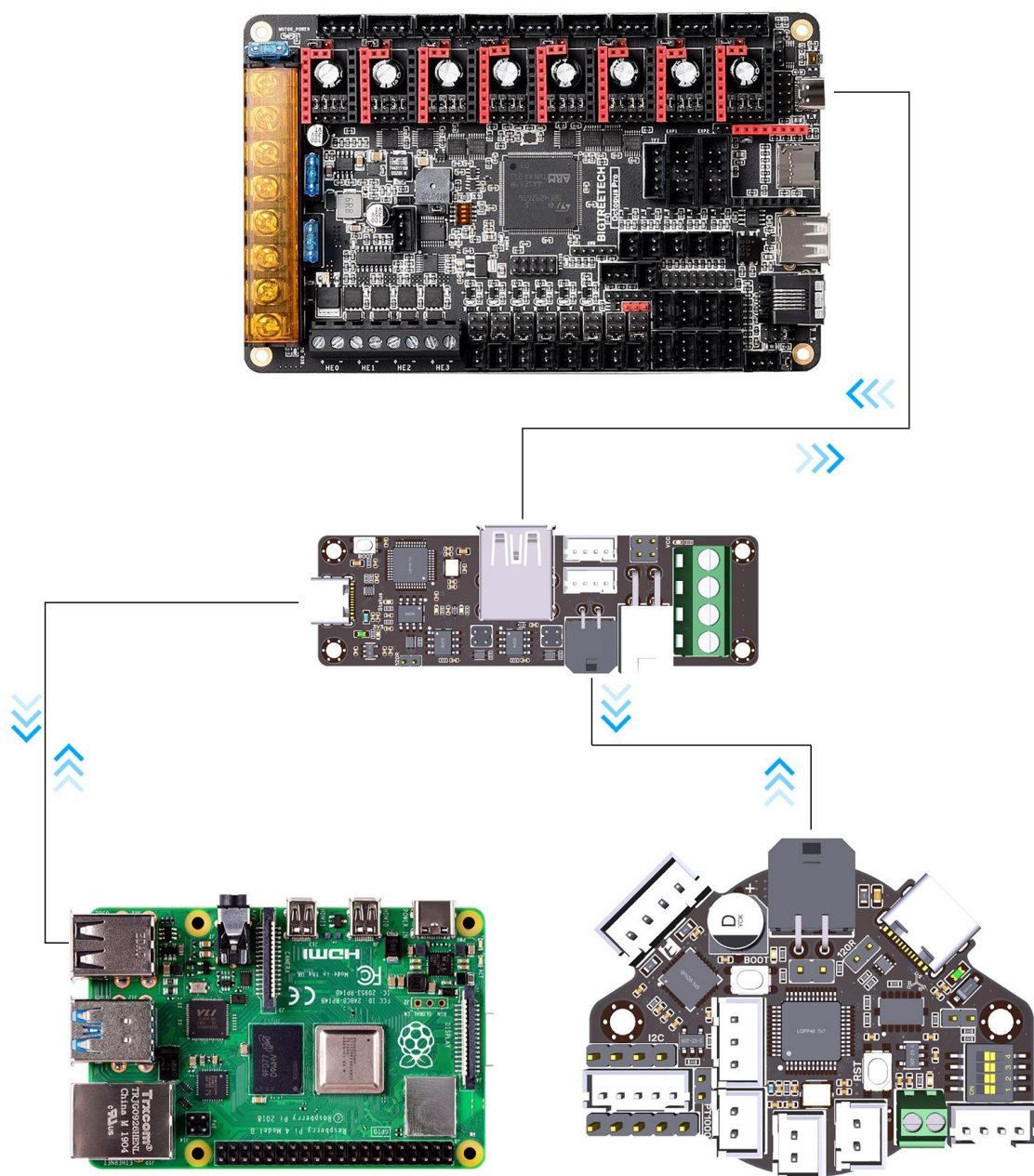


CAN_IN: USB 转 CAN 接口，用于连接树莓派。
 CAN_OUT*: 用于连接无 CAN 收发器但具备 CAN 功能的接口，比如连接主板的 USB 口，其中，主板 USB 口与主控连接的引脚要具备 CAN 功能。例如：

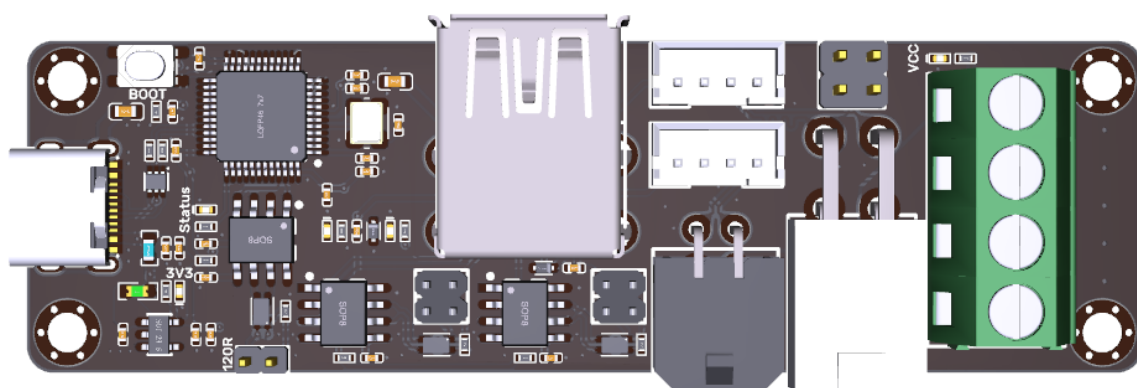
B12	70	C8	44	32	C1	PA11	I/O	FT	(3)	CAN_RX, USART1_CTS, TIM1_CH4, COMP1_OUT, TSC_G4_IO3, EVENTOUT	USB_DM
A12	71	B8	45	33	C3	PA12	I/O	FT	(3)	CAN_TX, USART1_RTS, TIM1_ETR, COMP2_OUT, TSC_G4_IO4, EVENTOUT	USB_DP

CAN_OUT: 用于连接 BIGTREETECH EBB CAN 或其他具备 CAN 收发器的板子。
 Power: 电源。

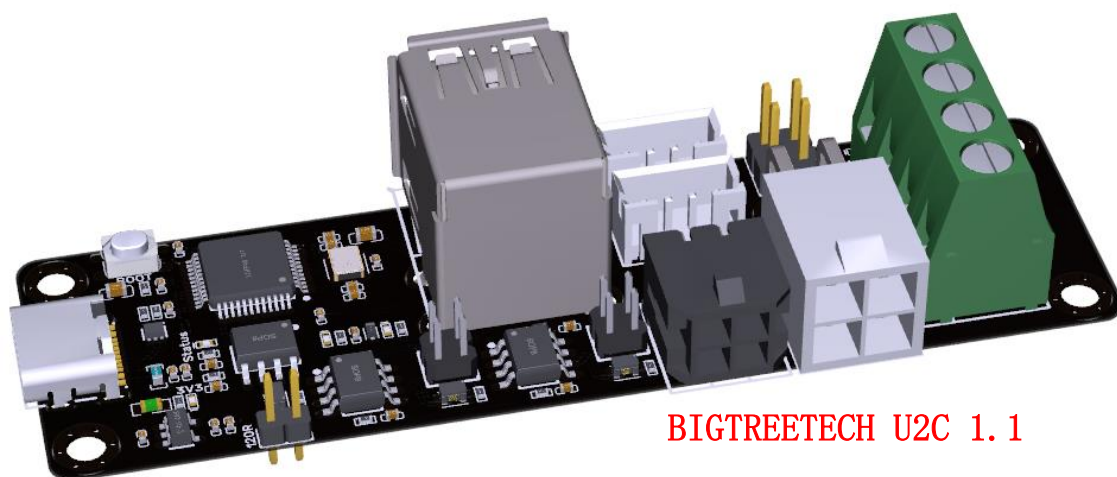
三、可选的接线图



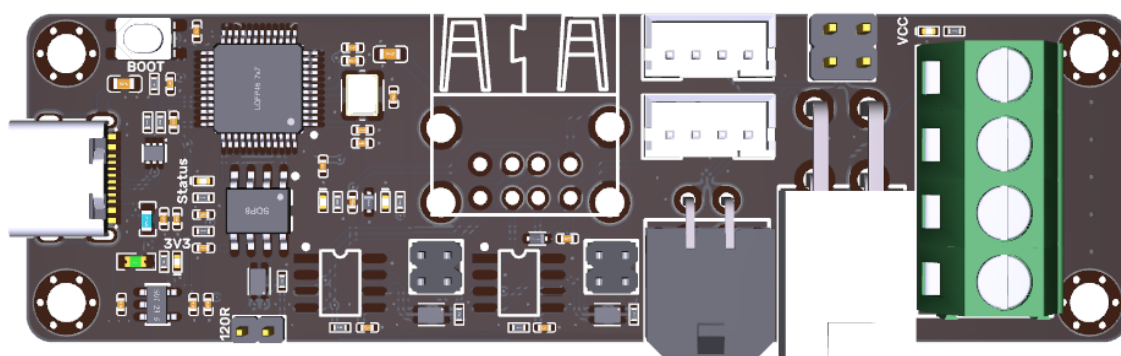
四、图片展示



BIGTREETECH U2C 1.1

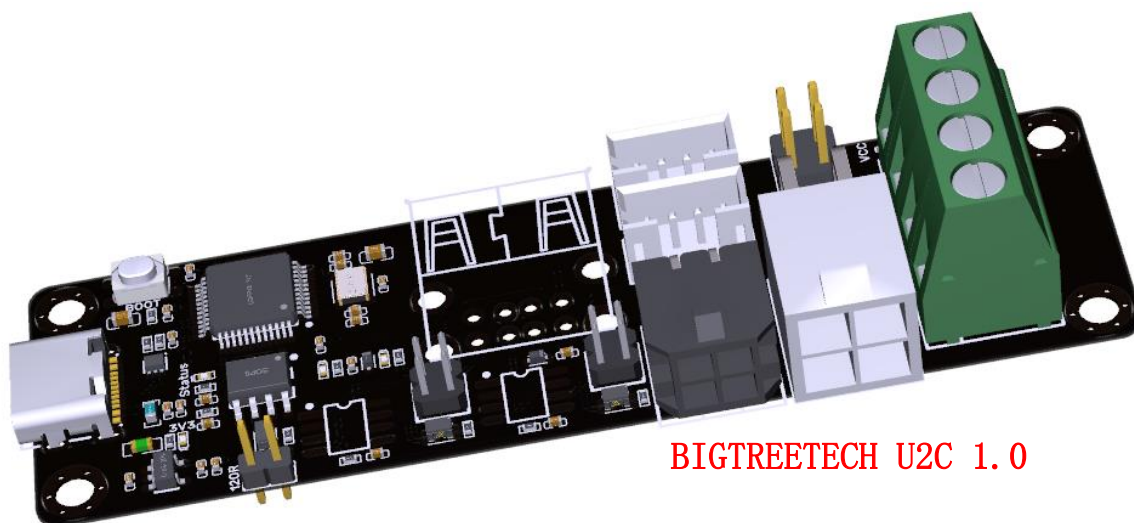


BIGTREETECH U2C 1.1

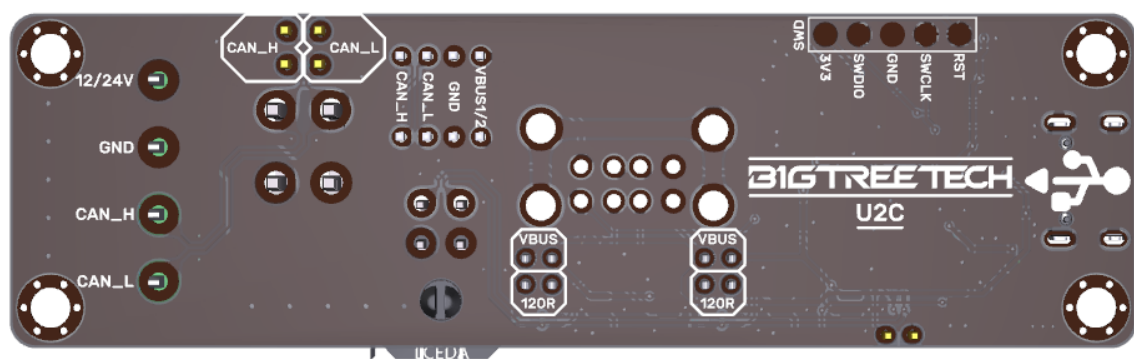


BIGTREETECH U2C 1.0

深圳市必趣科技有限公司
BIGTREETECH



BIGTREETECH U2C 1.0



BOTTOM

五、固件说明

5.1 固件源码

V1.0/1.1 使用的是开源的 candlelight 固件: https://github.com/candlelight-usb/candleLight_fw

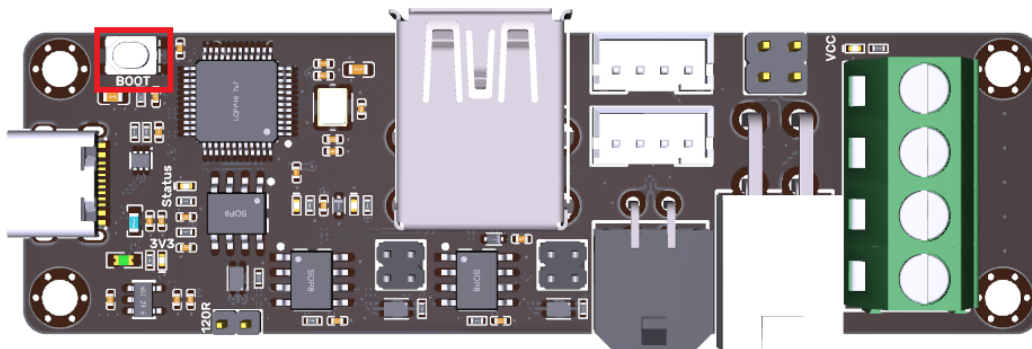
MCU 型号为 STM32F072C8

V2.0/2.1 在开源的 candlelight 基础上增加了 STM32G0 芯片的支持:
https://github.com/bigtreotech/candleLight_fw/tree/stm32g0_support

MCU 型号为 STM32G0B1C8

5.2 固件更新

模块完全断电，按住 Boot 按钮，插入 Type-C 数据线给模块供电，进入 DFU 模式，使用 STM32CubeProgrammer 软件更新固件



5.3 Klipper 配置

固件配置参考: <https://www.klipper3d.org/CANBUS.html>

1. 在 ssh 终端中输入 `sudo nano /etc/network/interfaces.d/can0` 命令并执行

```
auto can0
iface can0 can static
    bitrate 500000
    up ifconfig $IFACE txqueuelen 1024
```

将 CANBus 速度设置为 500K（必须与固件中设置的速度一致 **(500000) 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

六、注意事项

1. 12/24V 和 GND 不要接反。
2. 使用 CAN 通讯时，需要看哪个 CAN 节点用作终端，如果是终端，必须将 120R 位置插上跳线帽；