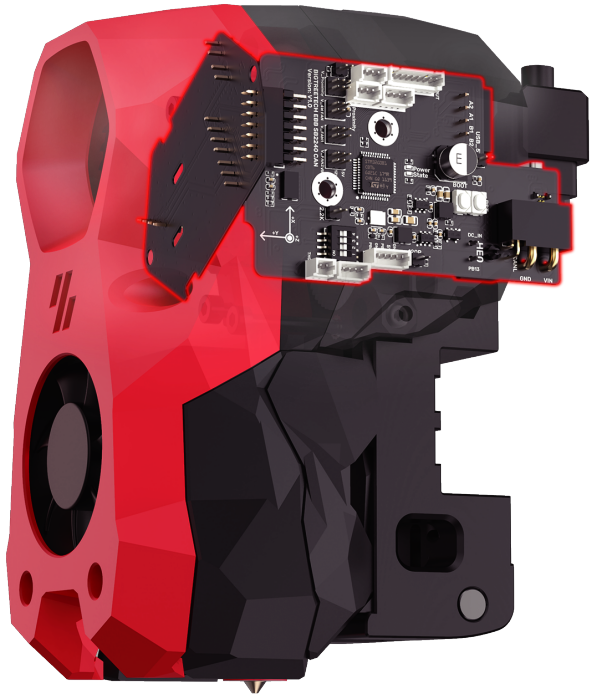
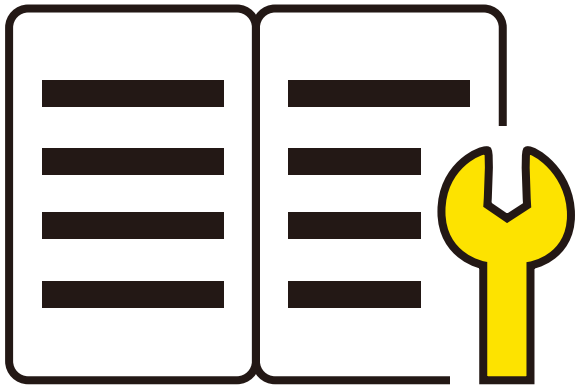
BIGTREETECH



EBB SB2240\2209 CAN V1.0
BUILD GUIDE

VERSION 2023-01-13



Thanks to CHAOTICLAB for providing guidance on Voron's official style build guide, and designing the printed part model for EBB SB2240\2209 CAN.

- Highlighted in blue are included in this EBB SB2240\2209 CAN v1.0.
- Highlighted in red are other accessories of the Voron StealthBurner, which are not included in this EBB SB2240\2209 CAN v1.0 and will need to be purchased by yourself.

Table of Contents

Packing List

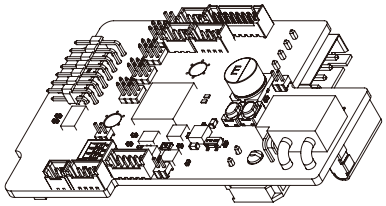
StealthBurner Body 1
- EBB SB0000 CAN v1.0 Overview 1
- Turbine Fan 3
- Axial Fan 4
- PWM Fan 5
- LED 6

ClockWork2 7
- EBB SB2240\2209 CAN v1.0 Overview 7
- Install EBB SB2240\2209 CAN v1.0 8
- Extruder Motor 10
- Thermistor 11
- PT100/1000 12
- Heater Cartridge 13

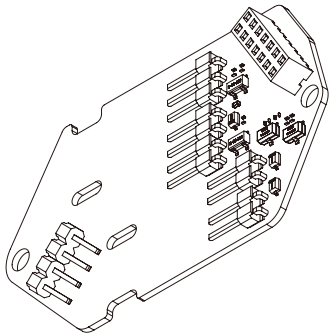
X-Carriage 14
- Proximity Switch 14
- X Axis Limit Switch 15
- Voron Tap 16
- BLTouch 17

StealthBurner Final Assembly 18
- CW2 Cable Bridge 19
- Fix CAN / USB-C Cable 21
- StealthBurner to Mainboard Wiring 23

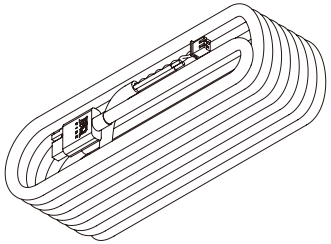
Klipper 24
- Compile Firmware 24
- Firmware Update (via STM32CubeProgrammer) 25
- Firmware Update (via DFU) 28
- Firmware Update (via Canboot) 31
- CAN bus Configure 34
- Klipper Configure 35



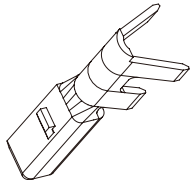
BIGTREETECH EBB SB2240/2209 CAN V1.0 1pc



BIGTREETECH EBB SB0000 CAN V1.0 1pc

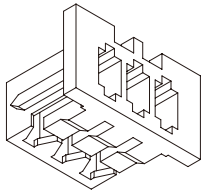


CAN bus Cable 1pc



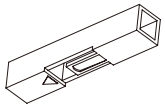
Crimp Pin

1.25 mm pitch	40pcs
3.0 mm pitch	5pcs
2.54 mm pitch	20pcs



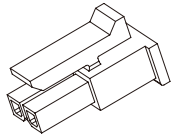
1.25 mm Pitch Pin Connector Housing

2Way	1pc
3Way	2pcs
4Way	2pcs
5Way	1pc
8Way	1pc



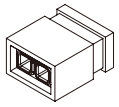
2.54mm Pitch Pin Connector Housing

1Way	15pcs

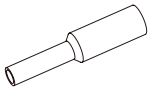


3.0 mm Pitch Pin Connector Housing

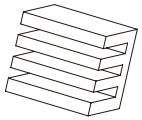
2Way	1pc



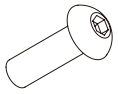
Jumper Cap 15pcs



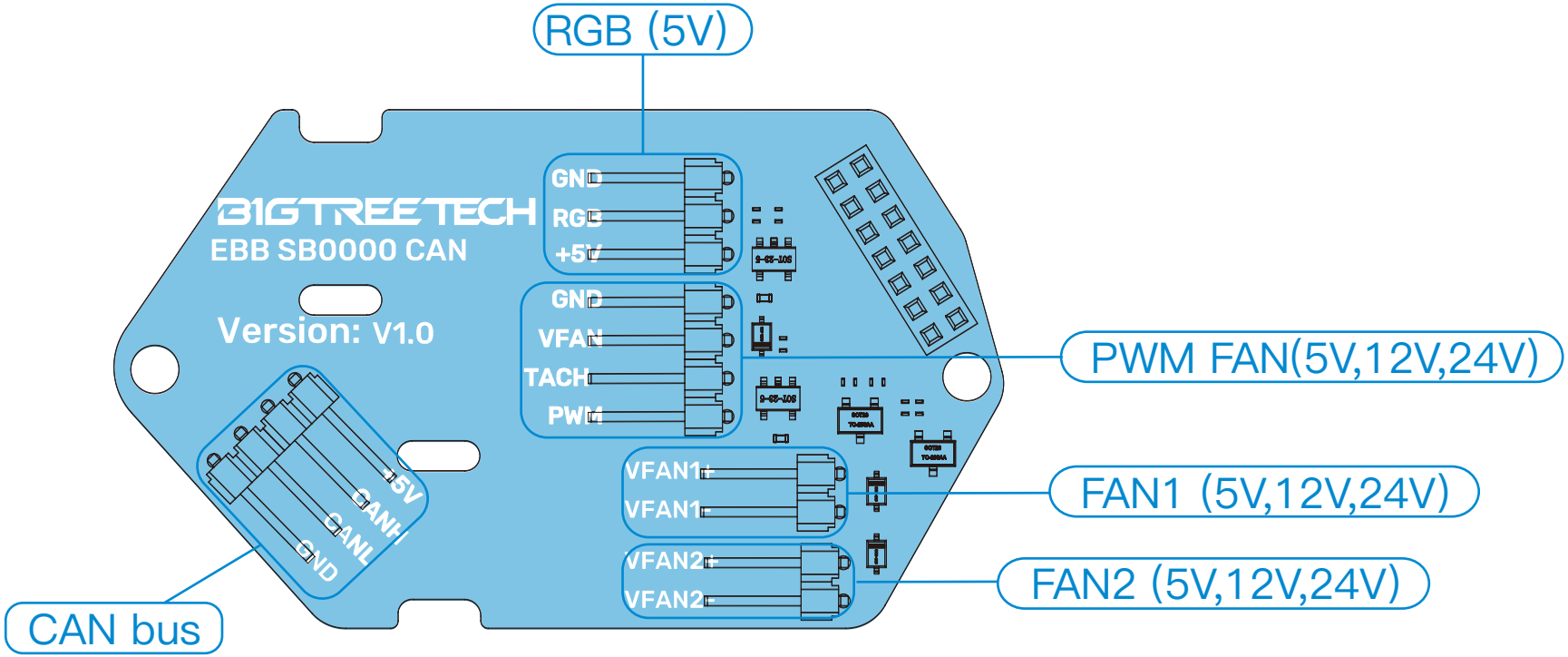
Insulated Wire Ferrule 2pcs



Heat Sink 1pc

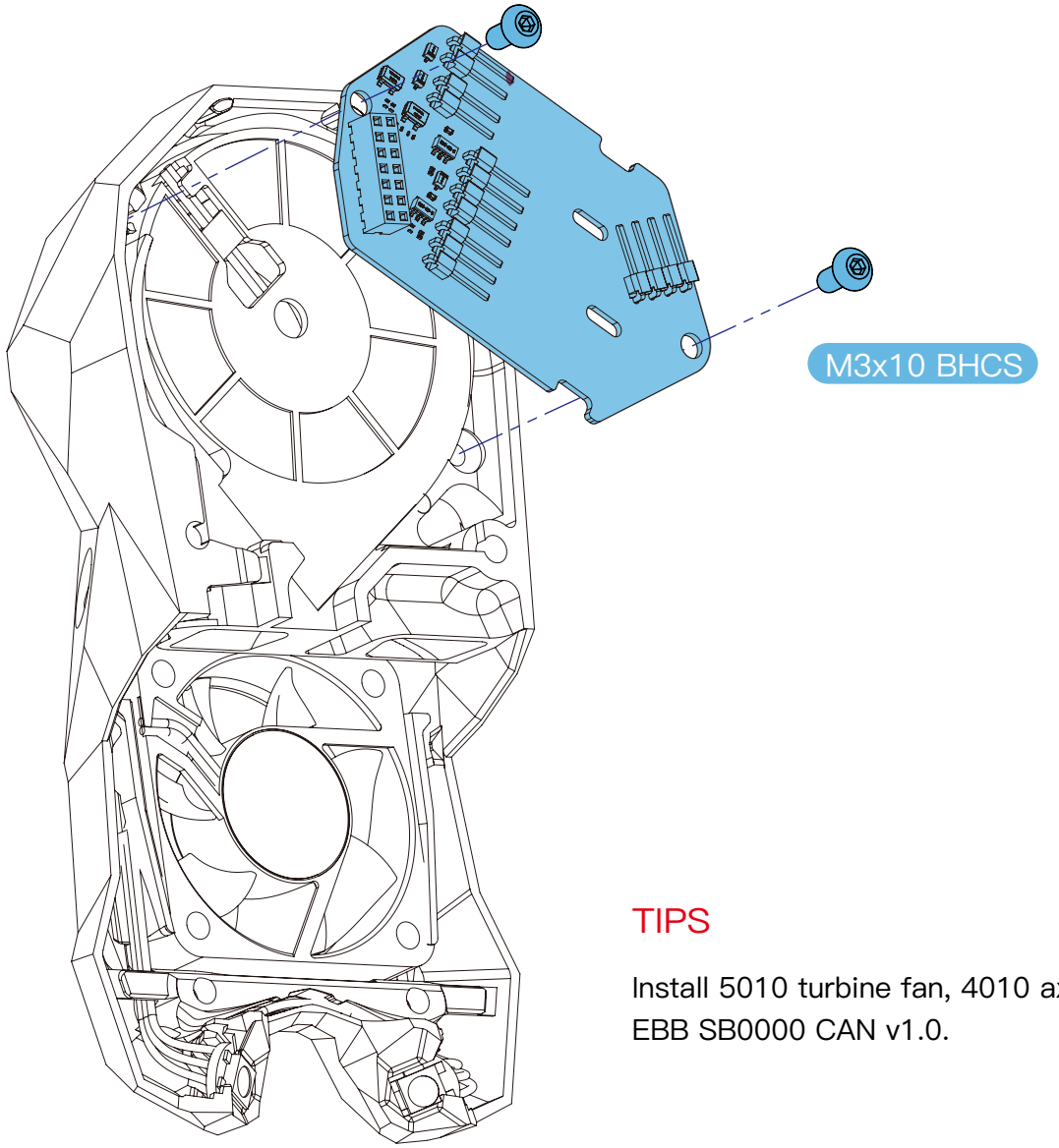


M3 x 10 Button Head Cap Screw (BHCS) 4pcs



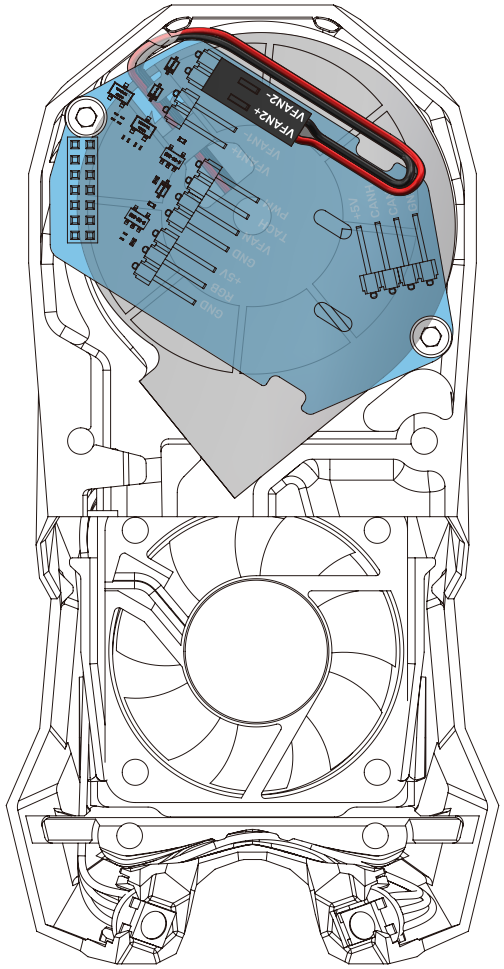
WIRE TERMINAL

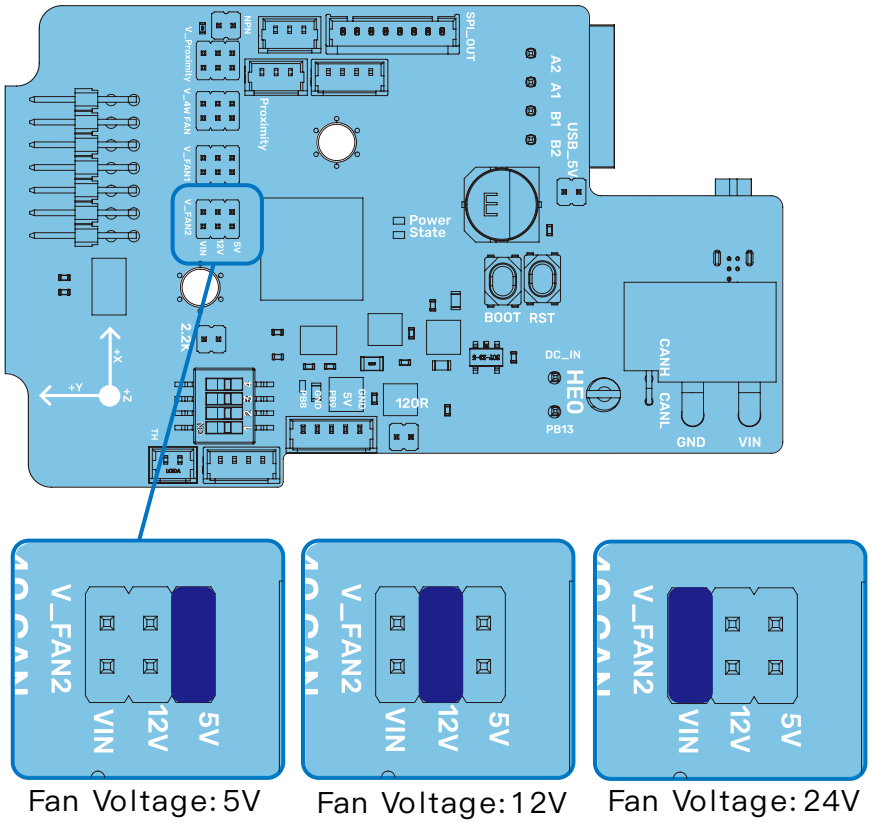
EBB SB0000 CAN v1.0 use 2.54 pitch DuPont connector. Therefore, all fans and LED lights on StealthBurner Body should be 2.54 pitch DuPont connector.



TIPS

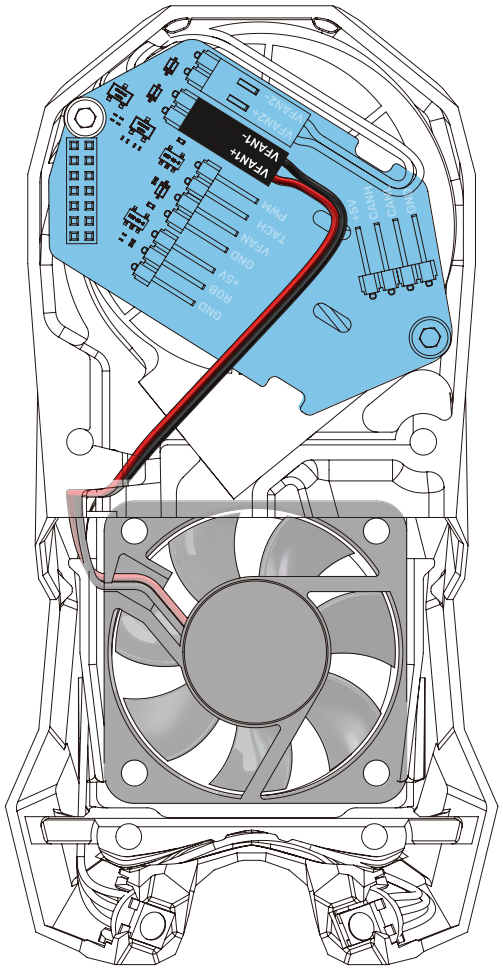
Install 5010 turbine fan, 4010 axial fan, and LED lamp before installing EBB SB0000 CAN v1.0.

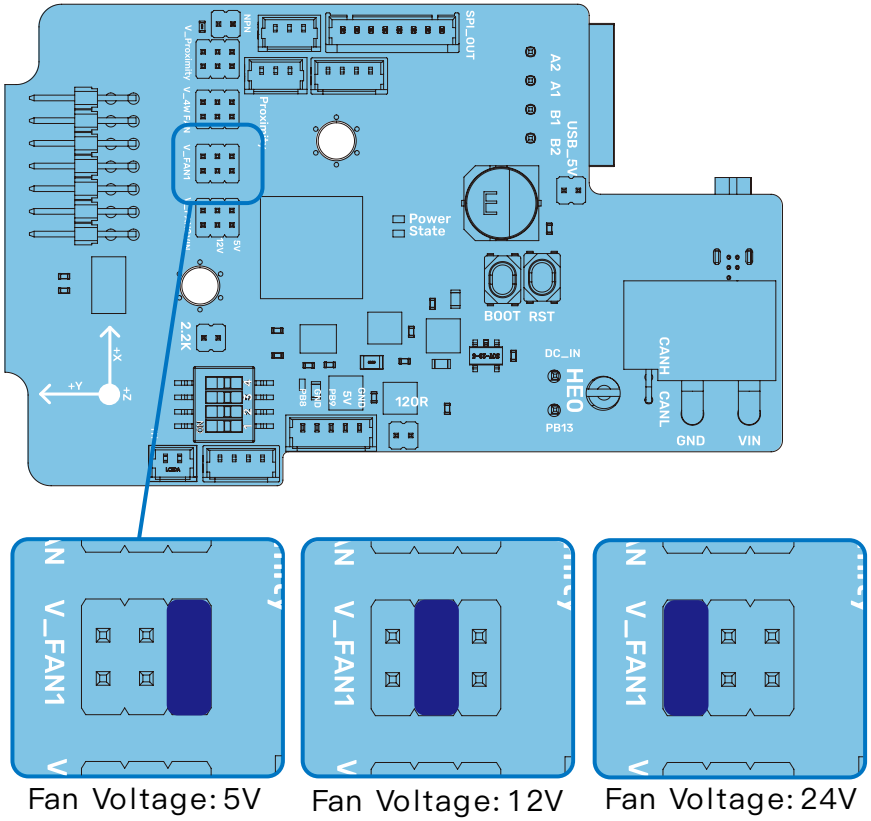




FAN VOLTAGE

All fans on the EBB SB0000 CAN v1.0 support multi voltage selection, as shown in the figure, and jumper caps are inserted or removed according to the actual voltage used.





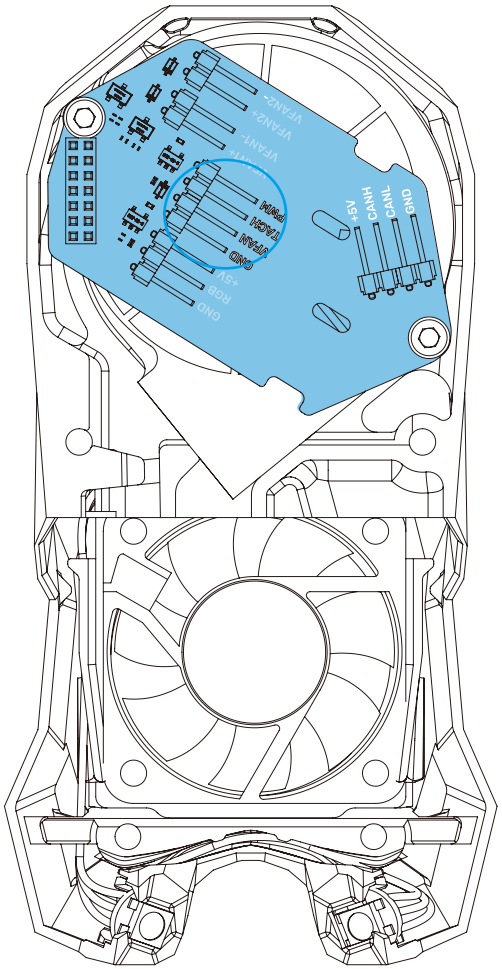
Fan Voltage: 5V

Fan Voltage: 12V

Fan Voltage: 24V

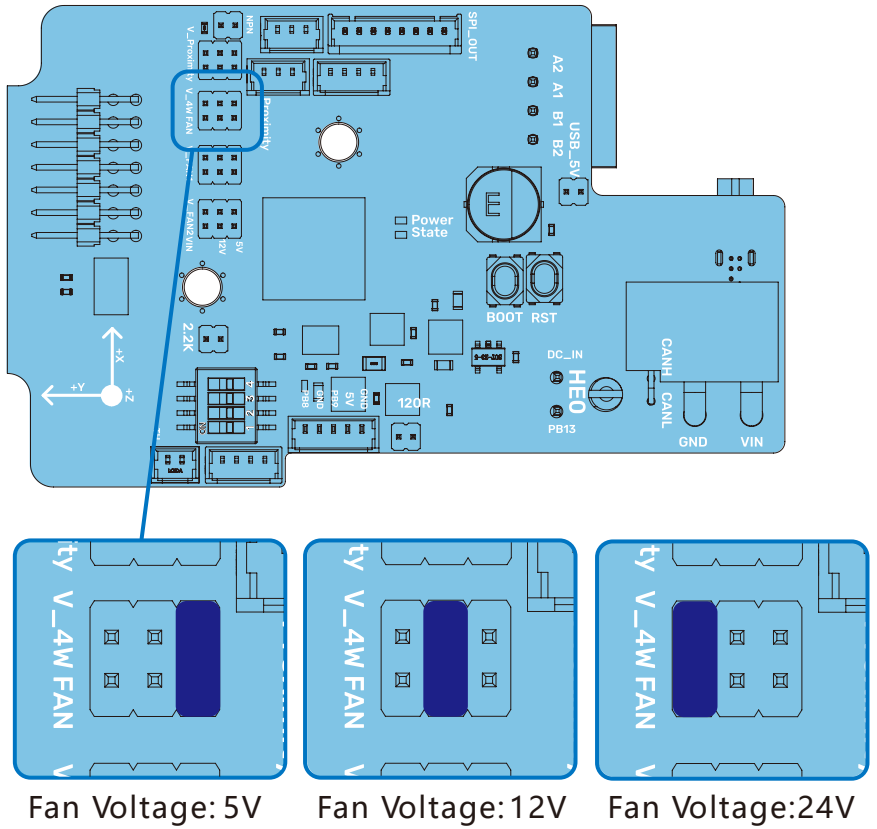
FAN VOLTAGE

All fans on the EBB SB0000 CAN v1.0 support multi voltage selection, as shown in the figure, and jumper caps are inserted or removed according to the actual voltage used.



PWM FAN

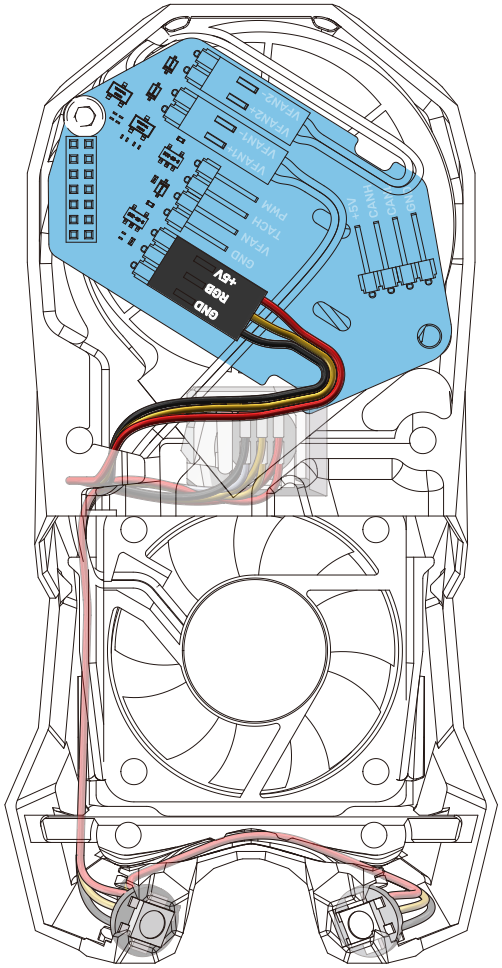
You can also use PWM fan on EBB SB0000 CAN v1.0. As shown in the figure, it contains a 4 pins PWM fan header (2.54 pitch and 4 pins DuPont connector).

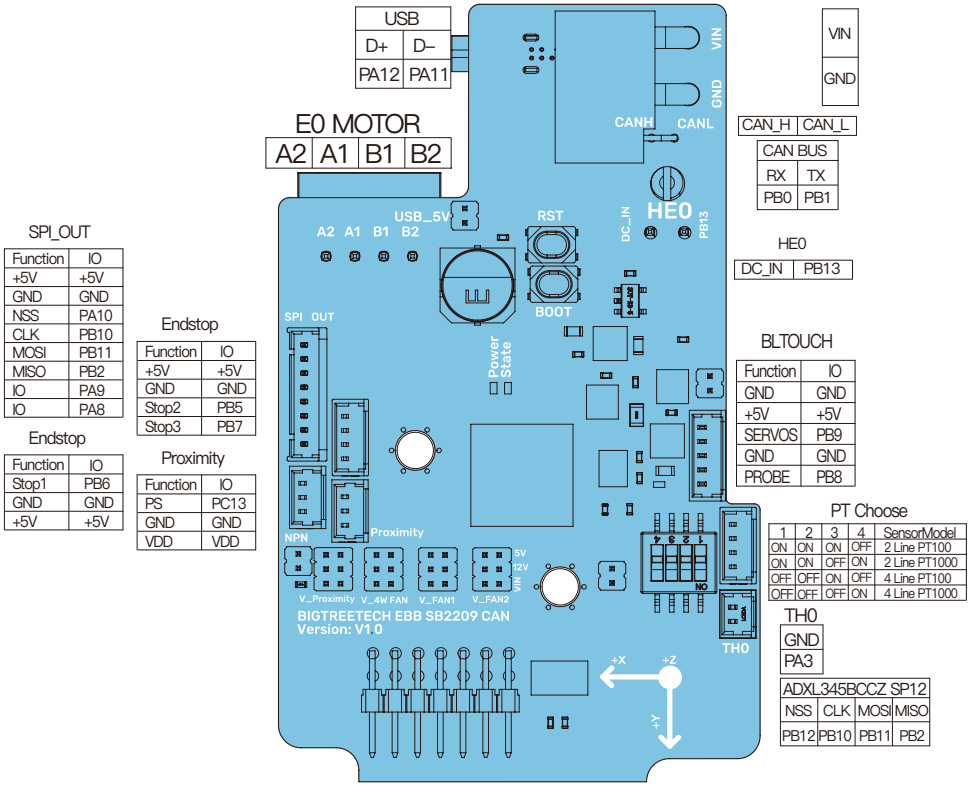


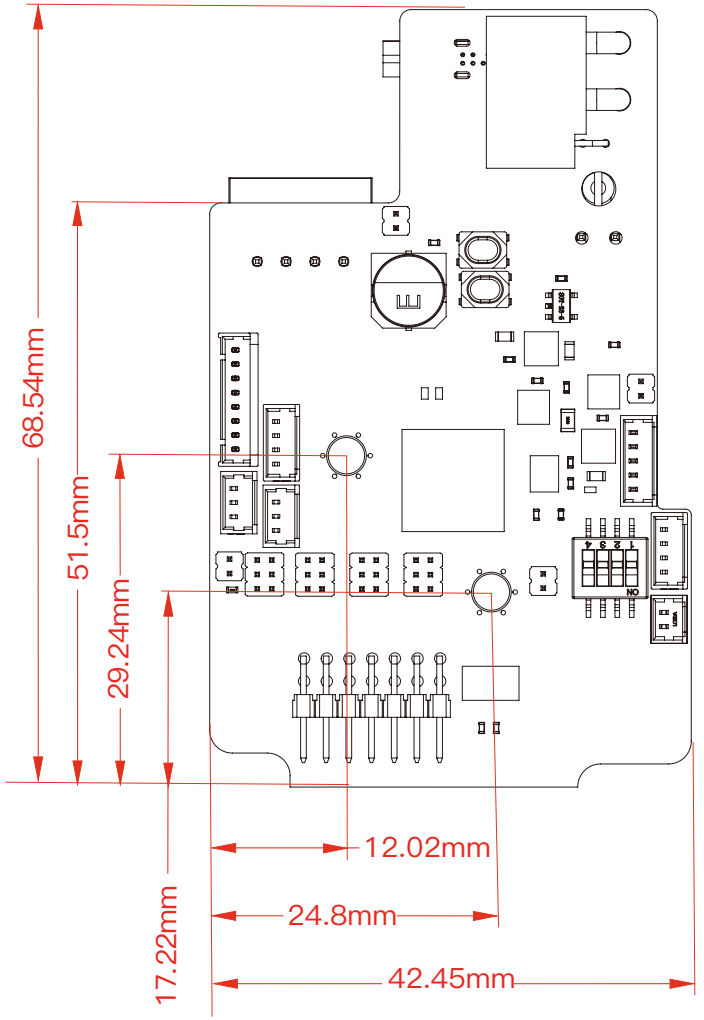
Fan Voltage: 5V Fan Voltage: 12V Fan Voltage:24V

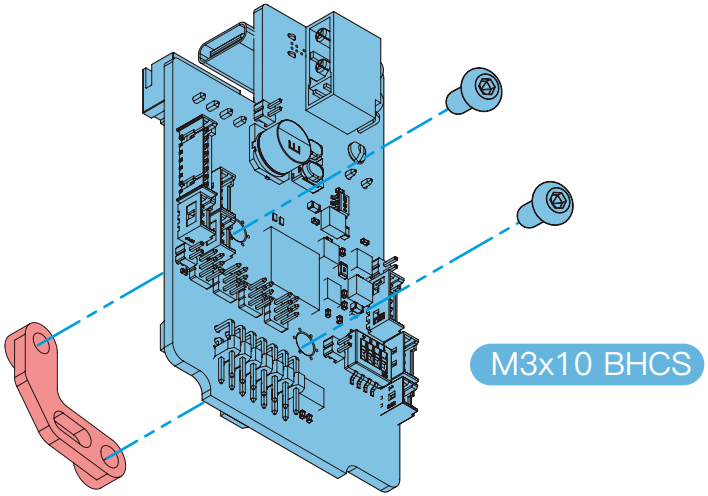
FAN VOLTAGE

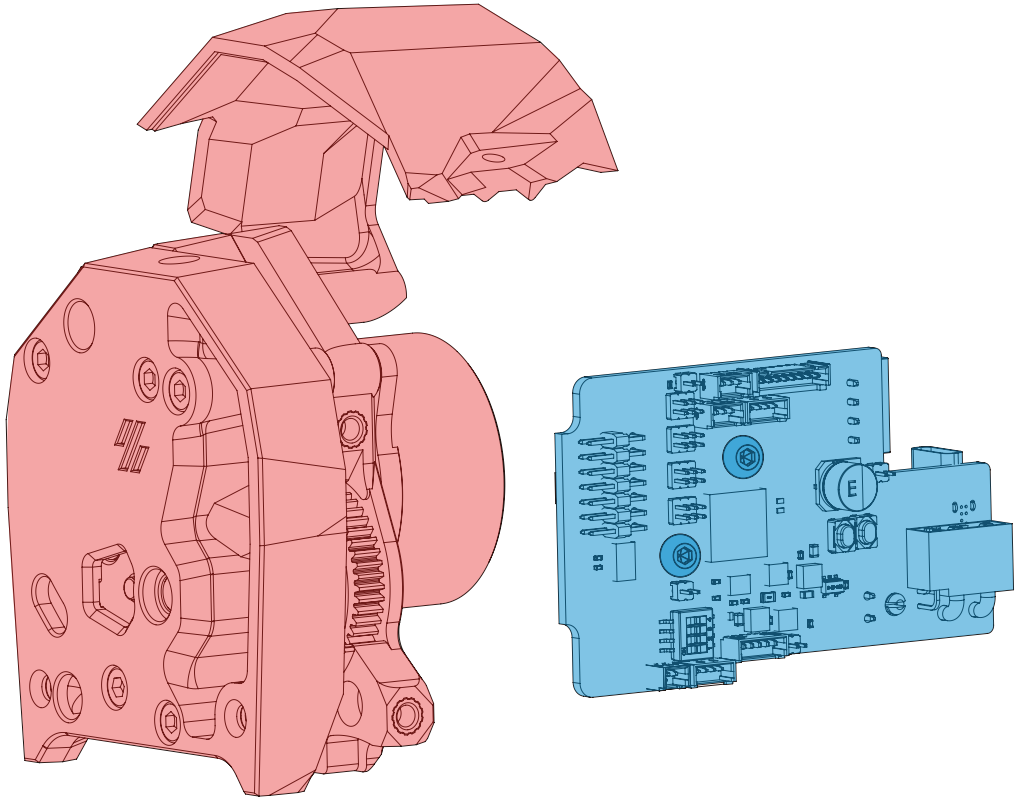
All fans on the EBB SB0000 CAN v1.0 support multi voltage selection, as shown in the figure, and jumper caps are inserted or removed according to the actual voltage used.

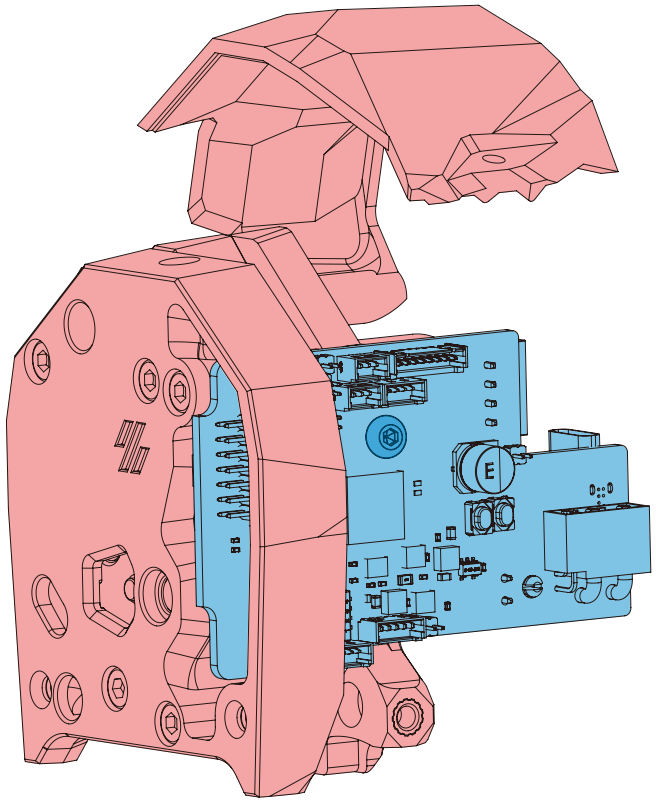

GND
RGB
+5V

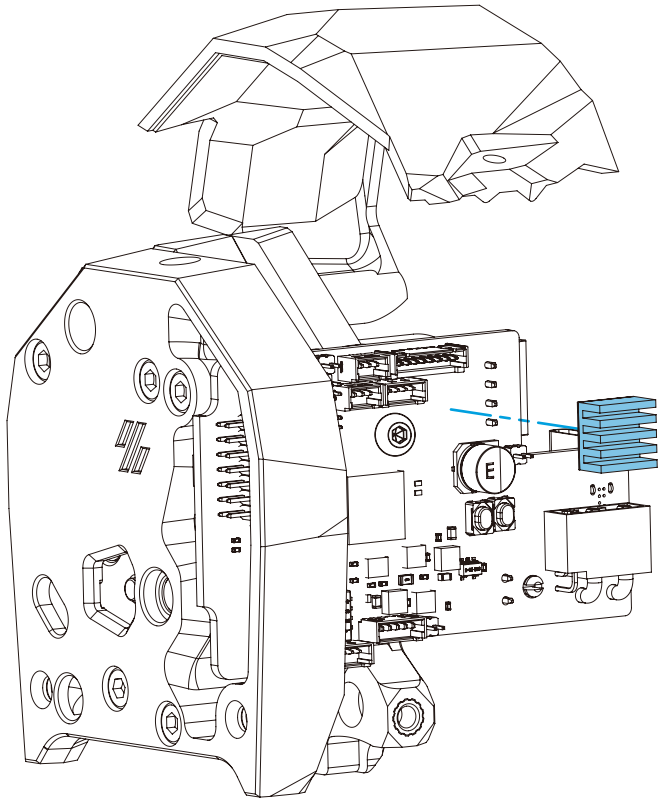

USB
D+ D-
PA12 PA11
E0 MOTOR
A2 A1 B1 B2
VIN
GND
CANH
CANL
CAN_H CAN_L
CAN BUS
RX TX
PB0 PB1
HE0
DC_IN PB13
RST
BOOT
USB_5V
SPI_OUT
Power State
BLTOUCH
Function IO
GND GND
+5V +5V
SERVOS PB9
GND GND
PROBE PB8
PT Choose
1 2 3 4 SensorModel
ON ON ON OFF 2 Line PT100
ON ON OFF ON 2 Line PT1000
OFF OFF ON OFF 4 Line PT100
OFF OFF OFF ON 4 Line PT1000
TH0
GND
PA3
ADXL345BCCZ SP12
NSS CLK MOSI MISO
PB12 PB10 PB11 PB2
SPI_OUT
Function IO
+5V +5V
GND GND
NSS PA10
CLK PB10
MOSI PB11
MISO PB2
IO PA9
IO PA8
Endstop
Function IO
Stop1 PB6
GND GND
+5V +5V
Endstop
Function IO
+5V +5V
GND GND
Stop2 PB5
Stop3 PB7
Proximity
Function IO
PS PC13
GND GND
VDD VDD
NPN
Proximity
V_Proximity V_4W FAN V_FAN1 V_FAN2
BIGTREETECH EBB SB2209 CAN
Version: V1.0
GND GND CANH +5V VDD 4WFAN VDD FAN1 VDD FAN2
Interface To EBB SB0000
Function GND CANL FAN2 PWM FAN1 PWM RGB 4WFAN TACH 4WFAN PWM
IO GND CANL PA1 PA0 PD3 PB15 PB14

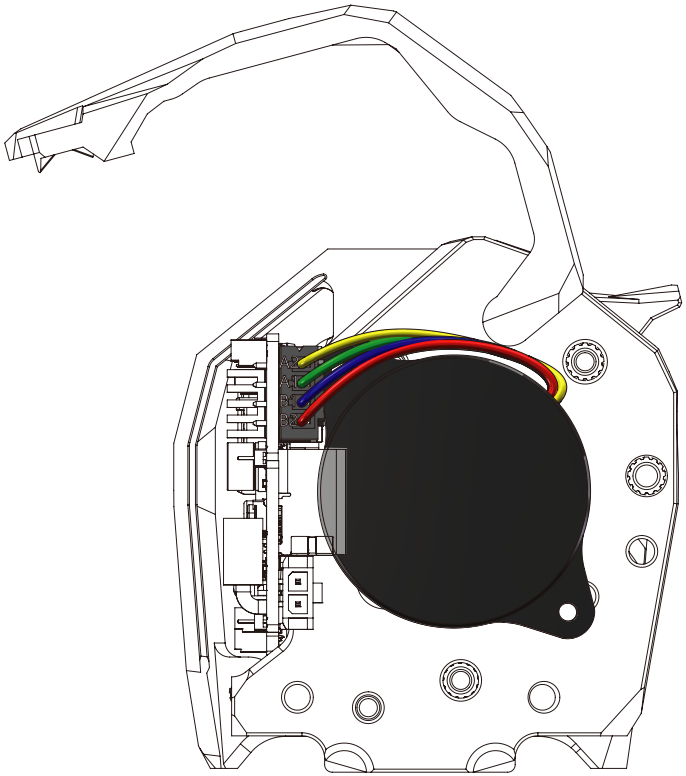

68.54mm
51.5mm
29.24mm
17.22mm
12.02mm
24.8mm
42.45mm

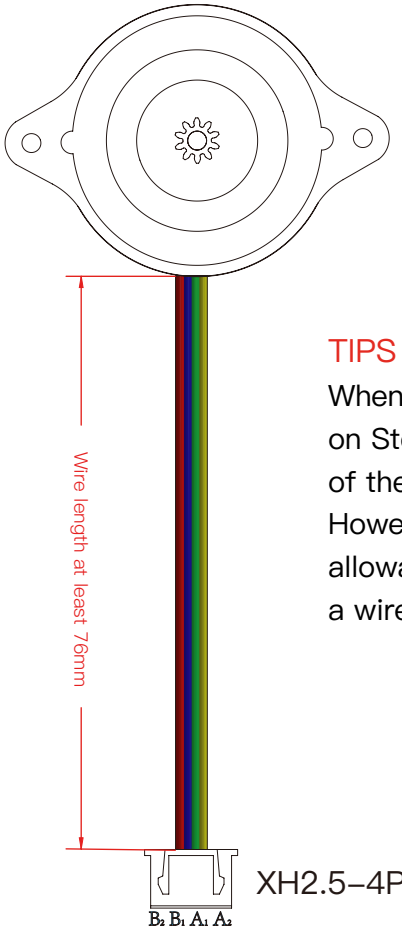

M3x10 BHCS
EBB SB2240\2209 CAN









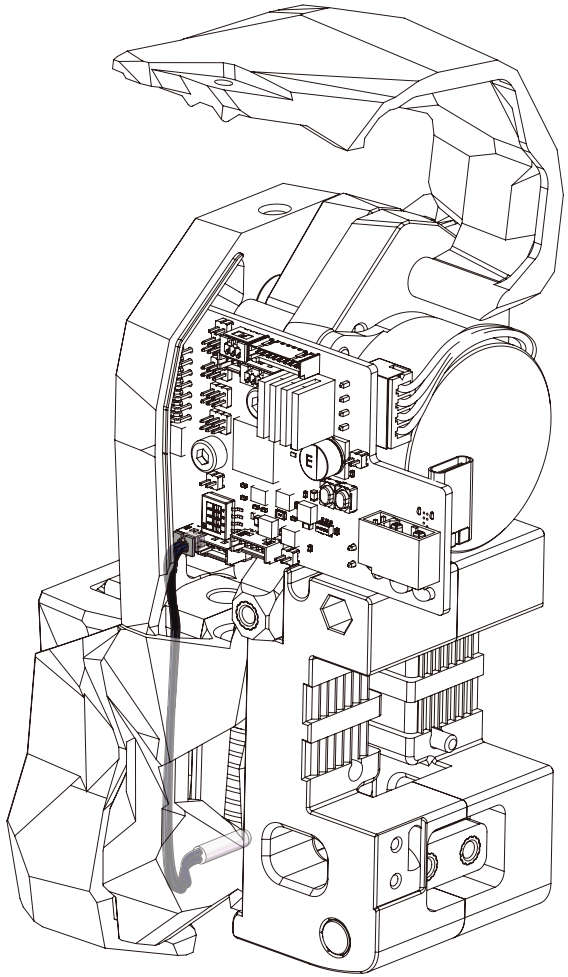


XH2.5-4PIN

TIPS

When EBB SB2240\2209 CAN v1.0 is used on StealthBurner, the minimum wire length of the motor is 76mm.

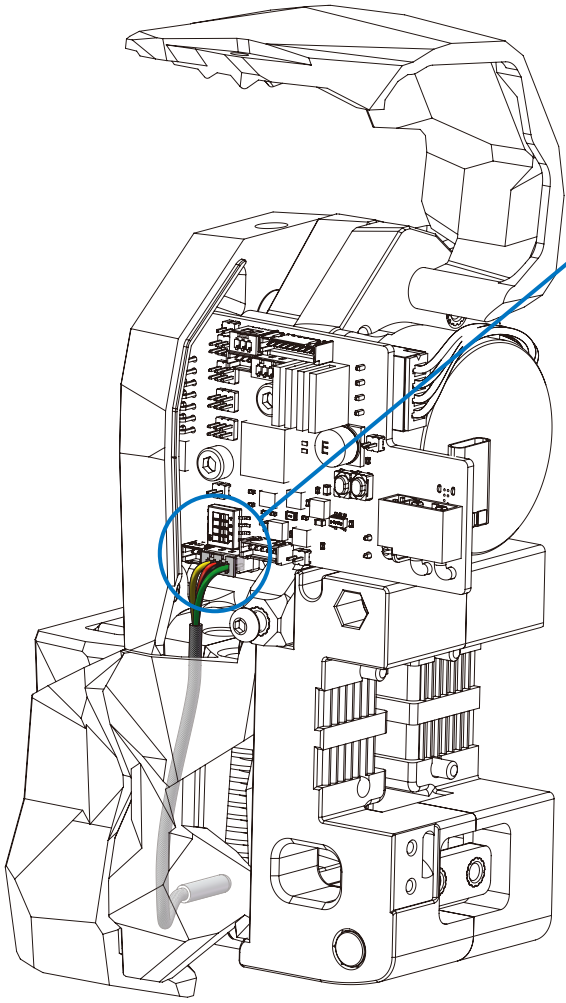
However, it is recommended to leave some allowance for wiring and assembly, such as a wire length of 100mm or 110mm.

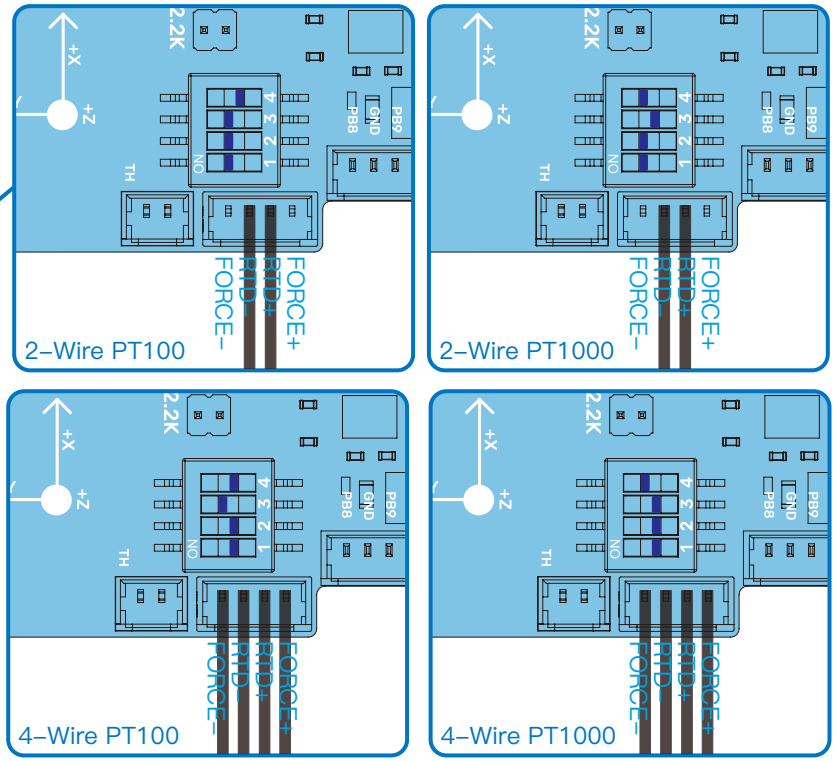


PH1.25-PIN

TIPS

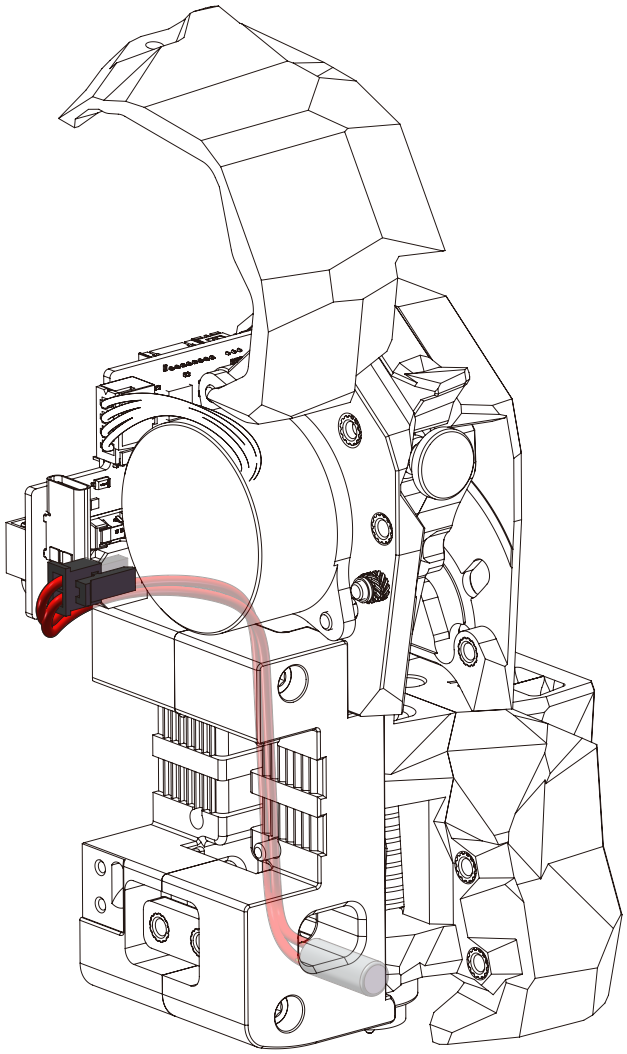
When EBB SB2240\2209 CAN v1.0 is used on StealthBurner, the minimum wire length of the thermistor is 103mm. However, it is recommended to leave some allowance for wiring and assembly, such as a wire length of 120mm or 130mm.

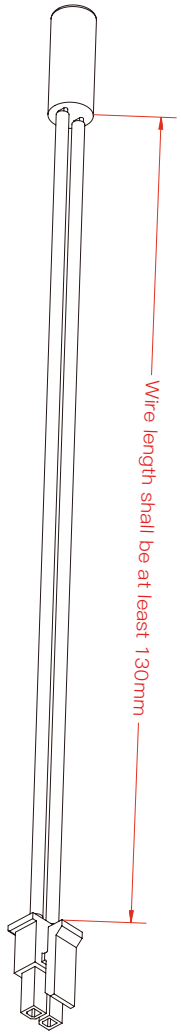




TIPS

EBB SB2240\2209 CAN v1.0 is compatible with 2-wire and 4-wire PT100 or PT1000, but not 3-wire PT100 or PT000. As shown in the figure, wire and adjust the dial switch according to the actual model. And the terminals are PH1.25.

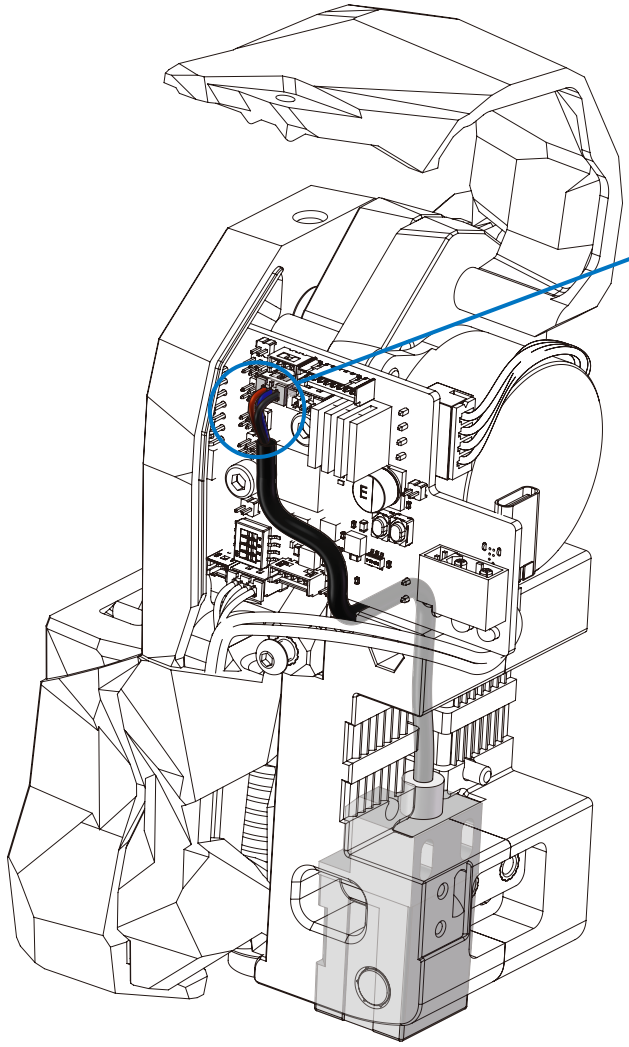


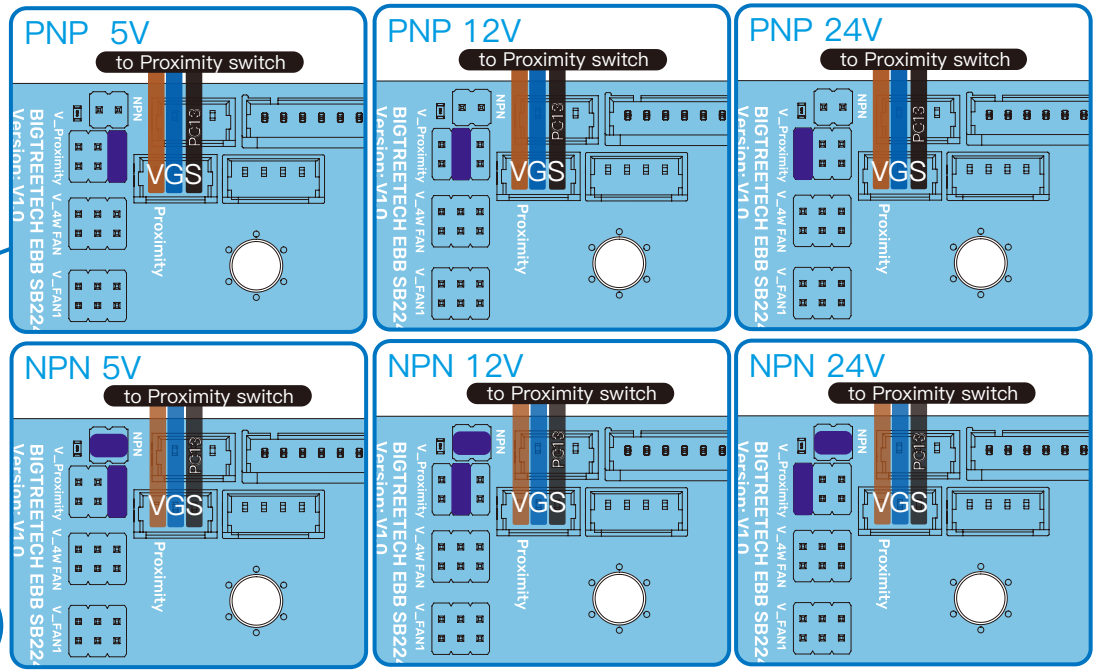


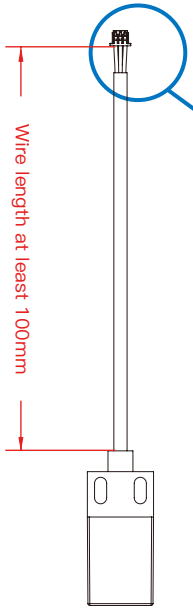
MX3.0-2PIN

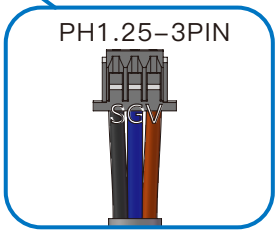
TIPS

When EBB SB2240\2209 CAN v1.0 is used on StealthBurner, the minimum wire length of the heater cartridge is 130mm. However, it is recommended to leave some allowance for wiring and assembly, such as a wire length of 150mm or 160mm.







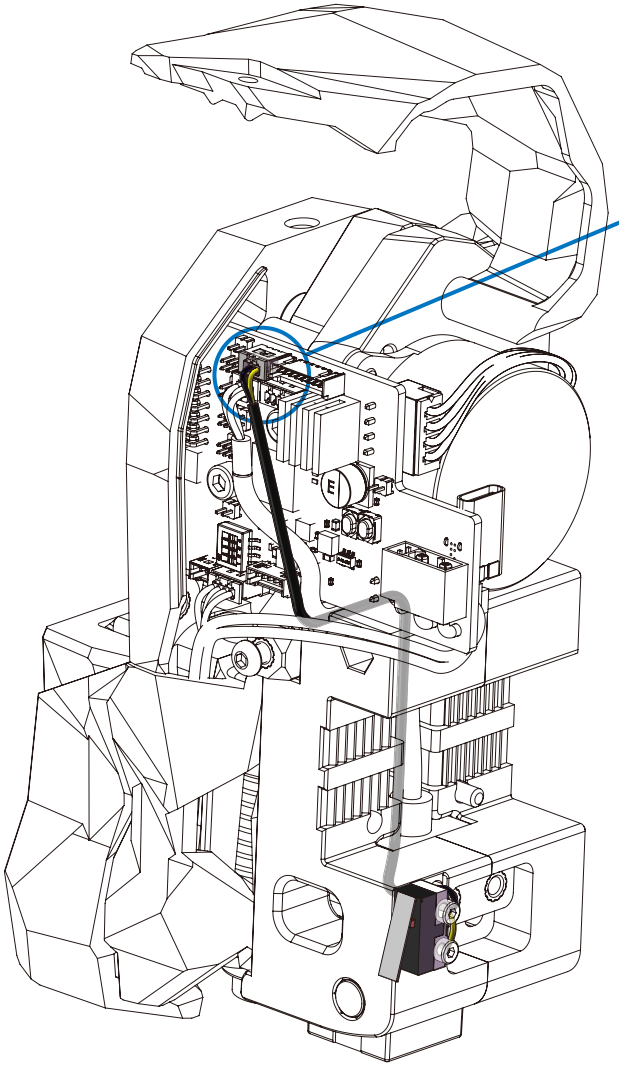


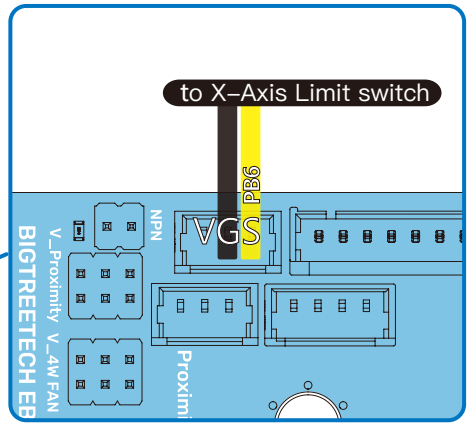
PROXIMITY SWITCH

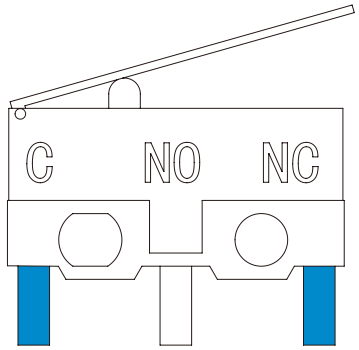
The proximity swith is divided into PNP and NPN and has various voltages. As shown in the figure, the jumper cap is inserted or removed at the corresponding position according to the actual model.

TIPS

When EBB SB2240\2209 CAN v1.0 is used on StealthBurner, the minimum wire length of the PL-08N is 100mm. However, it is recommended to leave some allowance for wiring and assembly, such as a wire length of 130mm or 140mm.

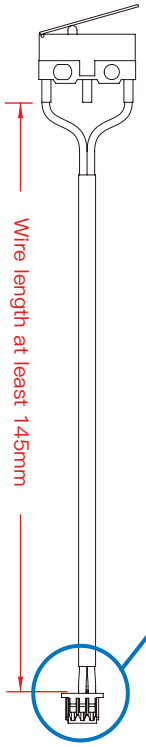






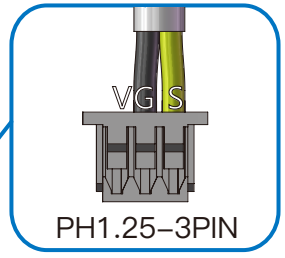
END-STOP SWITCHES FOR X

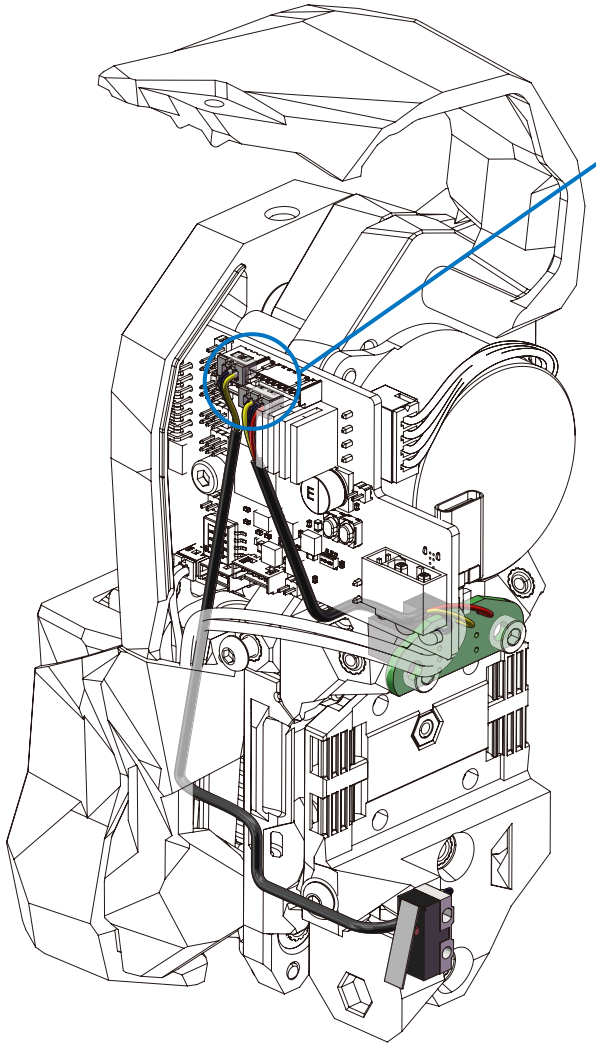
End-stops are wired in a "Normally Closed" configuration. On microswitches those are the 2 outer terminals indicated by C and NC.

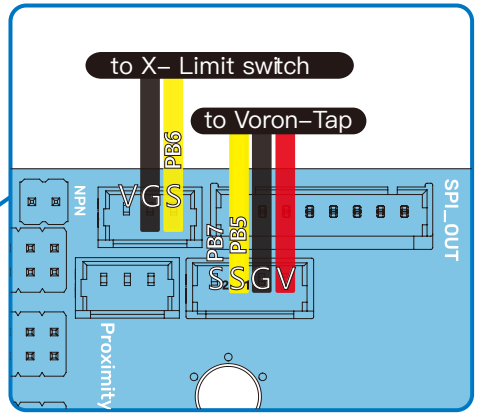


TIPS

When EBB SB2240\2209 CAN v1.0 is used on StealthBurner, the minimum wire length of the X Axis Limit Switch is 145mm. However, it is recommended to leave some allowance for wiring and assembly, such as a wire length of 165mm or 175mm.

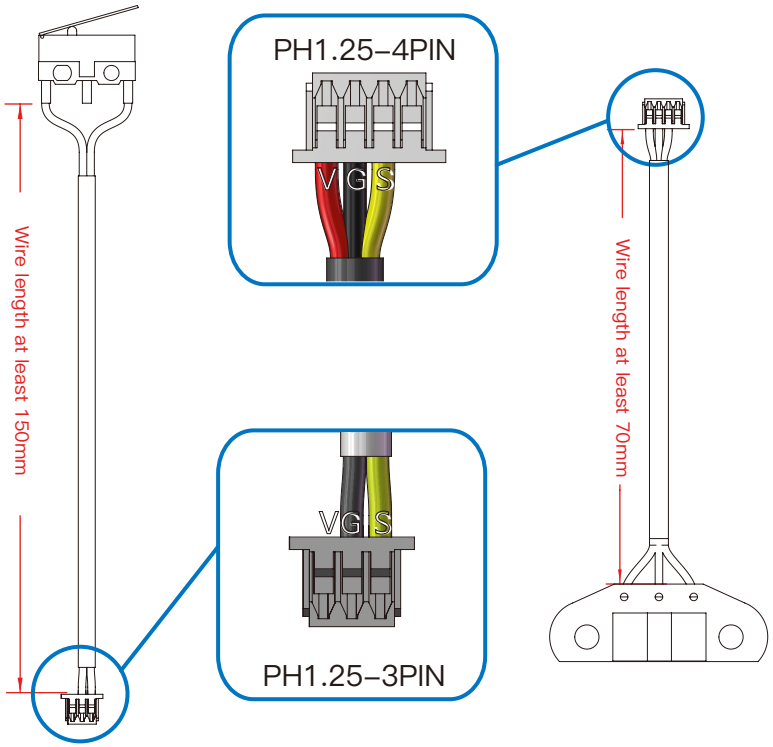


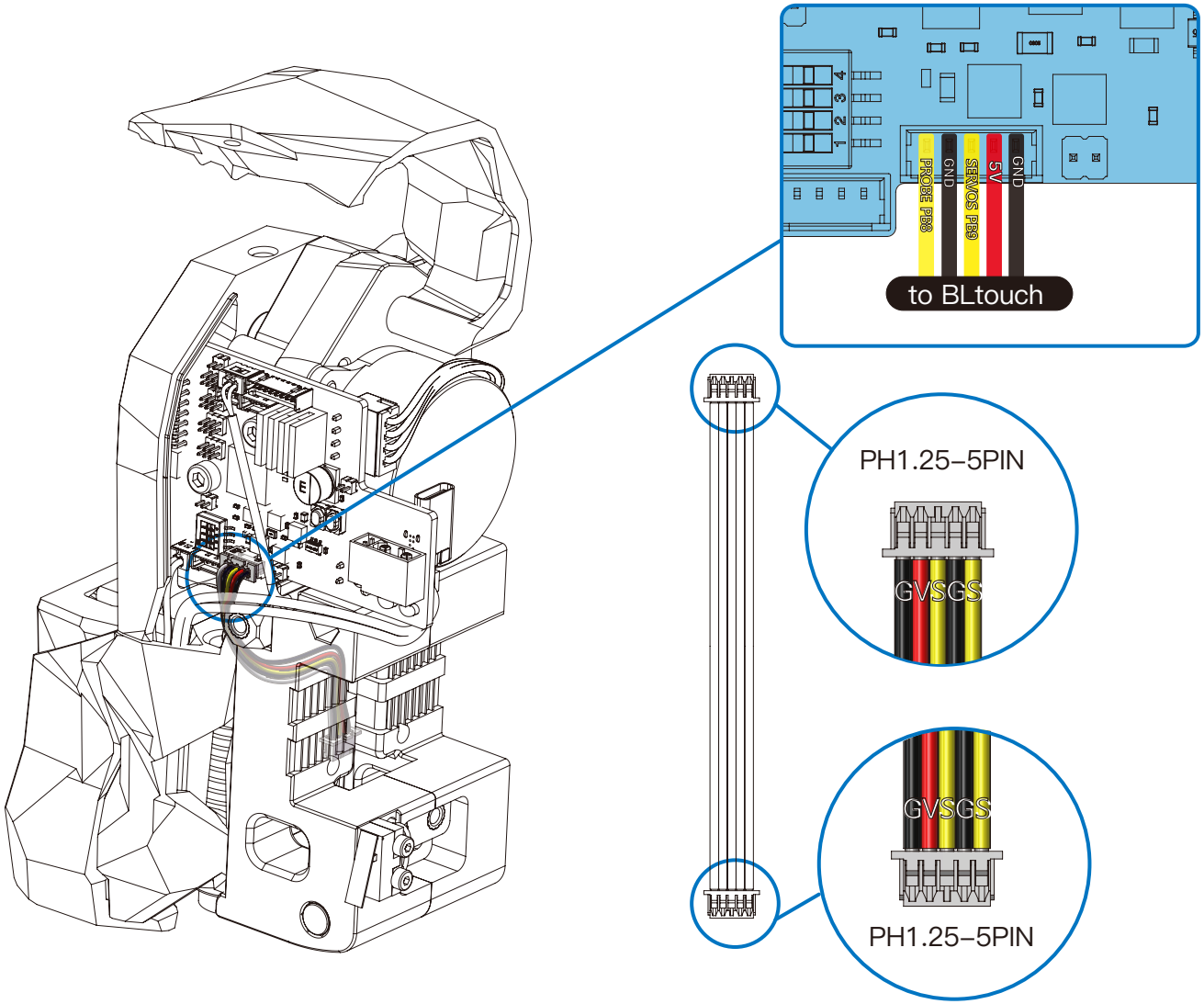


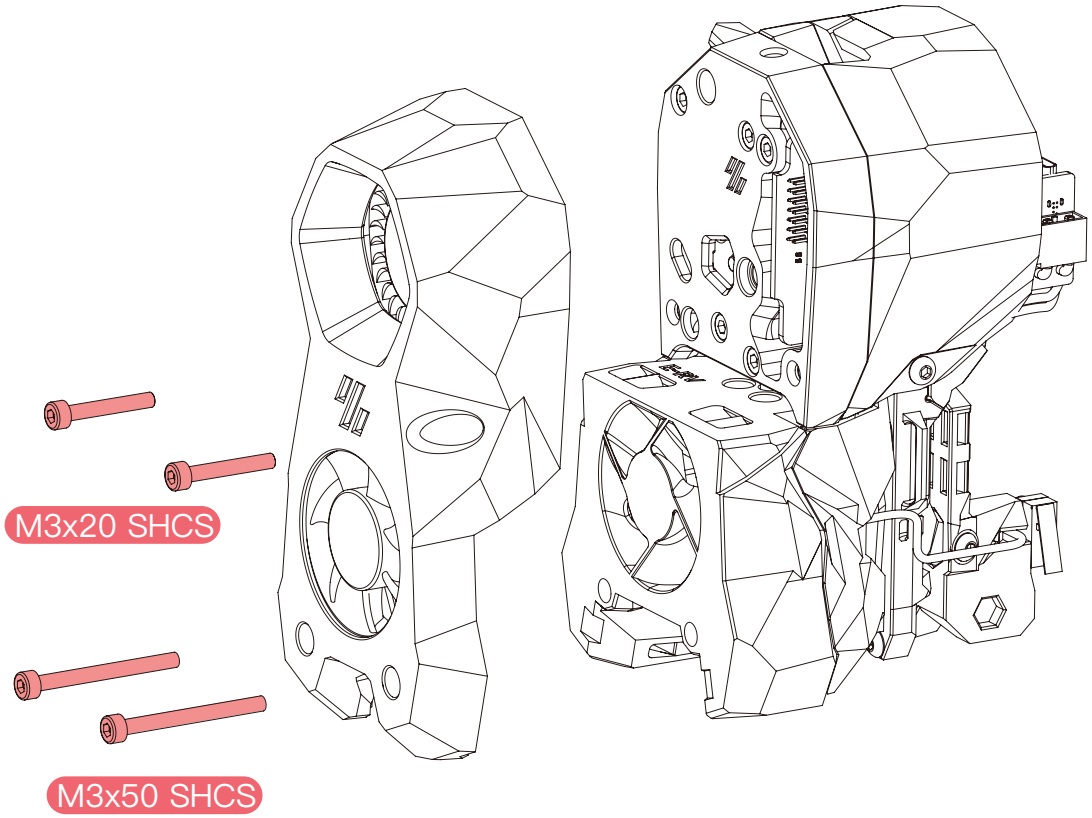


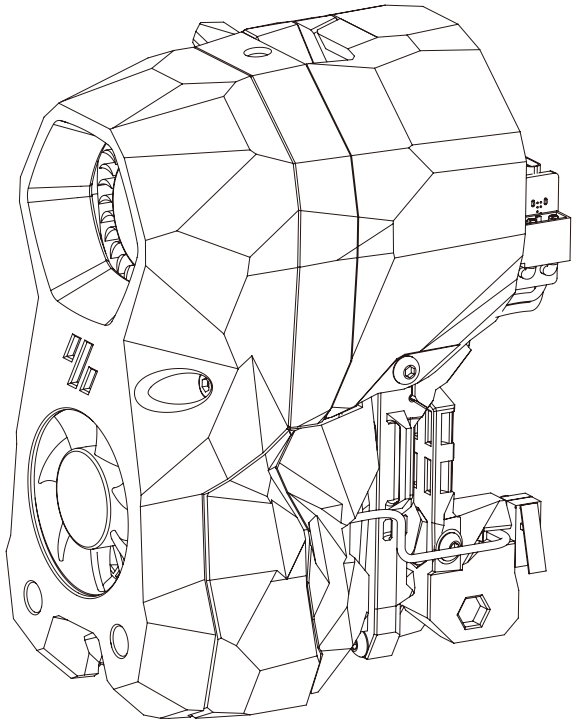
TIPS

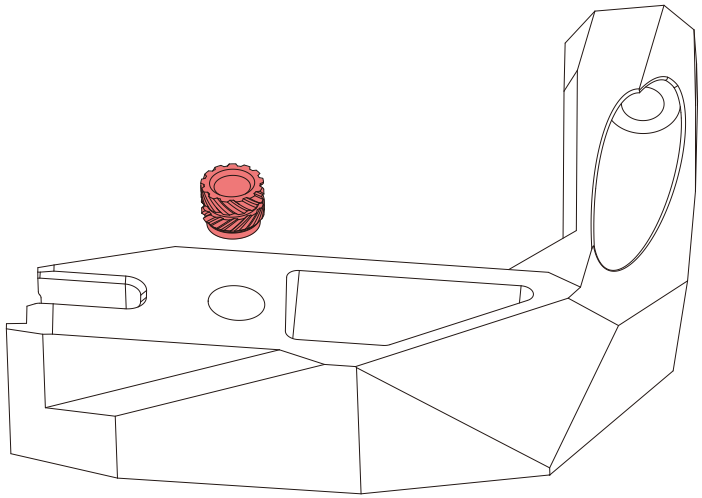
When EBB SB2240\2209 CAN v1.0 is used on StealthBurner, the minimum wire length of the X Axis Limit Switch is 150mm, Tap cable is 70mm. However, it is recommended to leave some allowance for wiring and assembly, such as a wire length of 170mm or 180mm for X Axis Limit Switch and 100mm or 110mm for Tap.

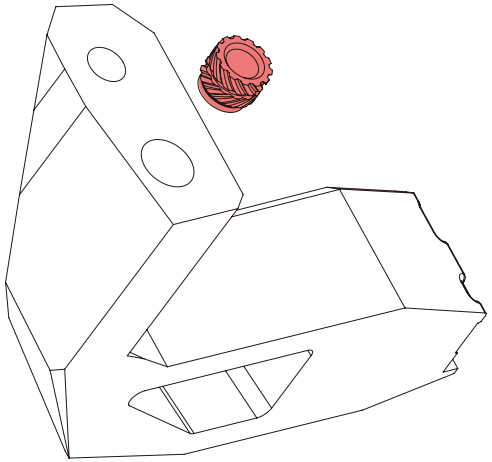


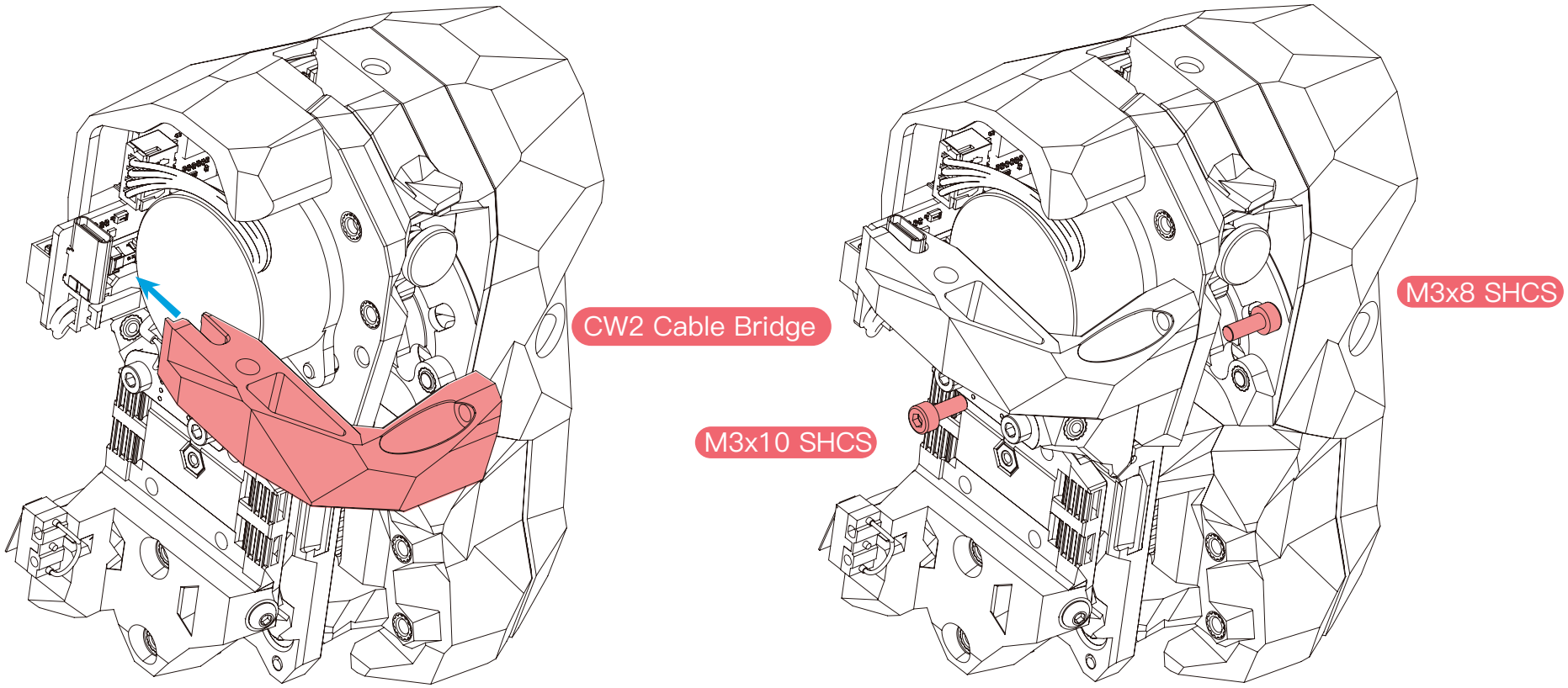

1 2 3 4
PROBE PB8
GND
SERVOS PB9
5V
GND
to BLtouch
PH1.25-5PIN
GVSGS
GVSGS
PH1.25-5PIN


M3x20 SHCS
M3x50 SHCS



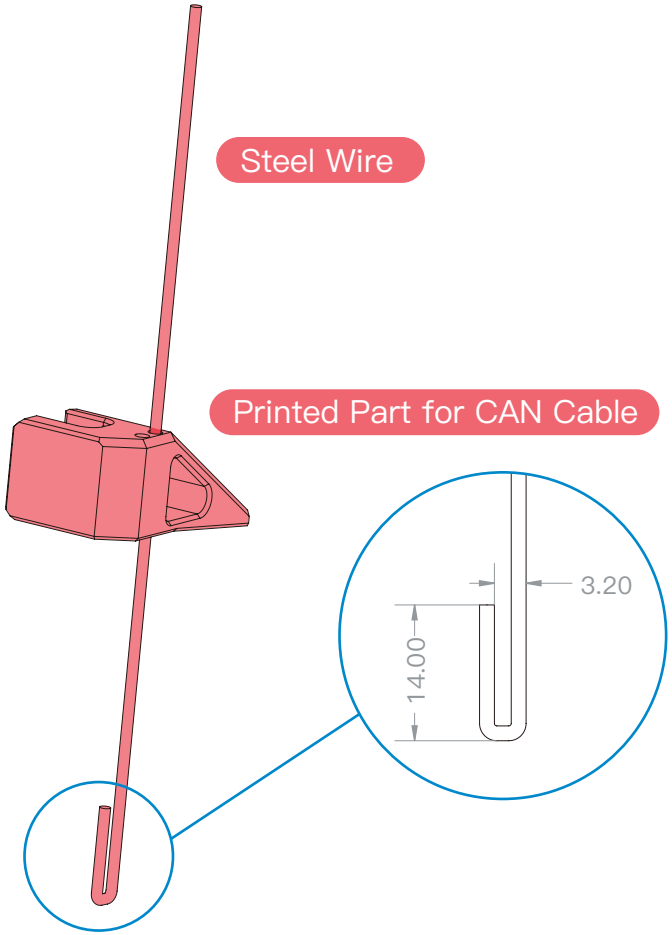


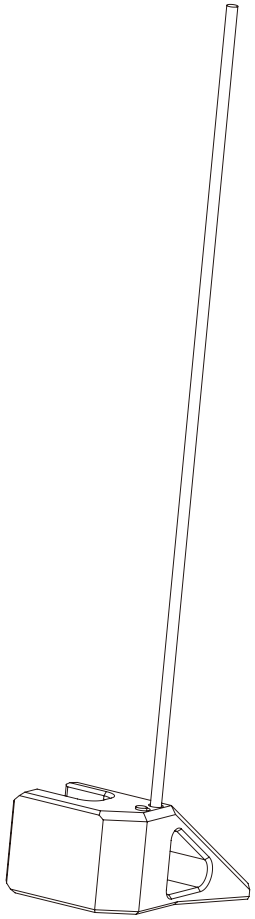




TIPS

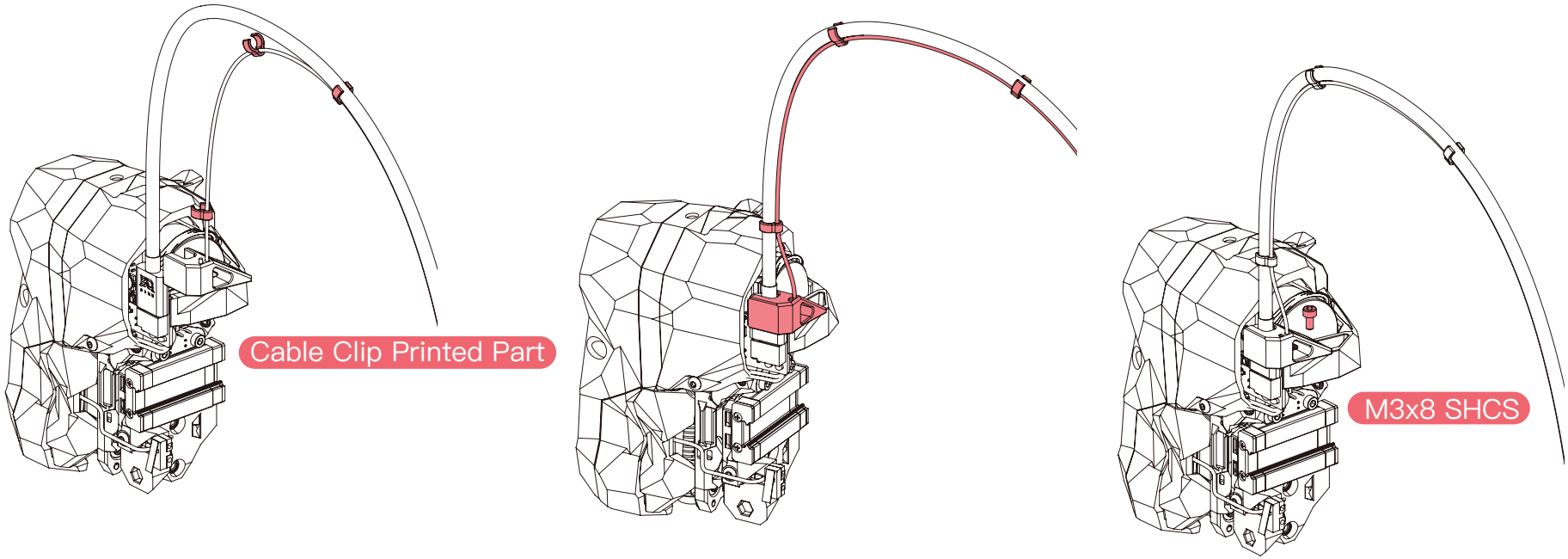
The CW2 cable bridge is a custom printed part that can be downloaded directly from our GitHub repository, in EBB\EBB SB2240_2209 CAN\Custom Printed Parts.





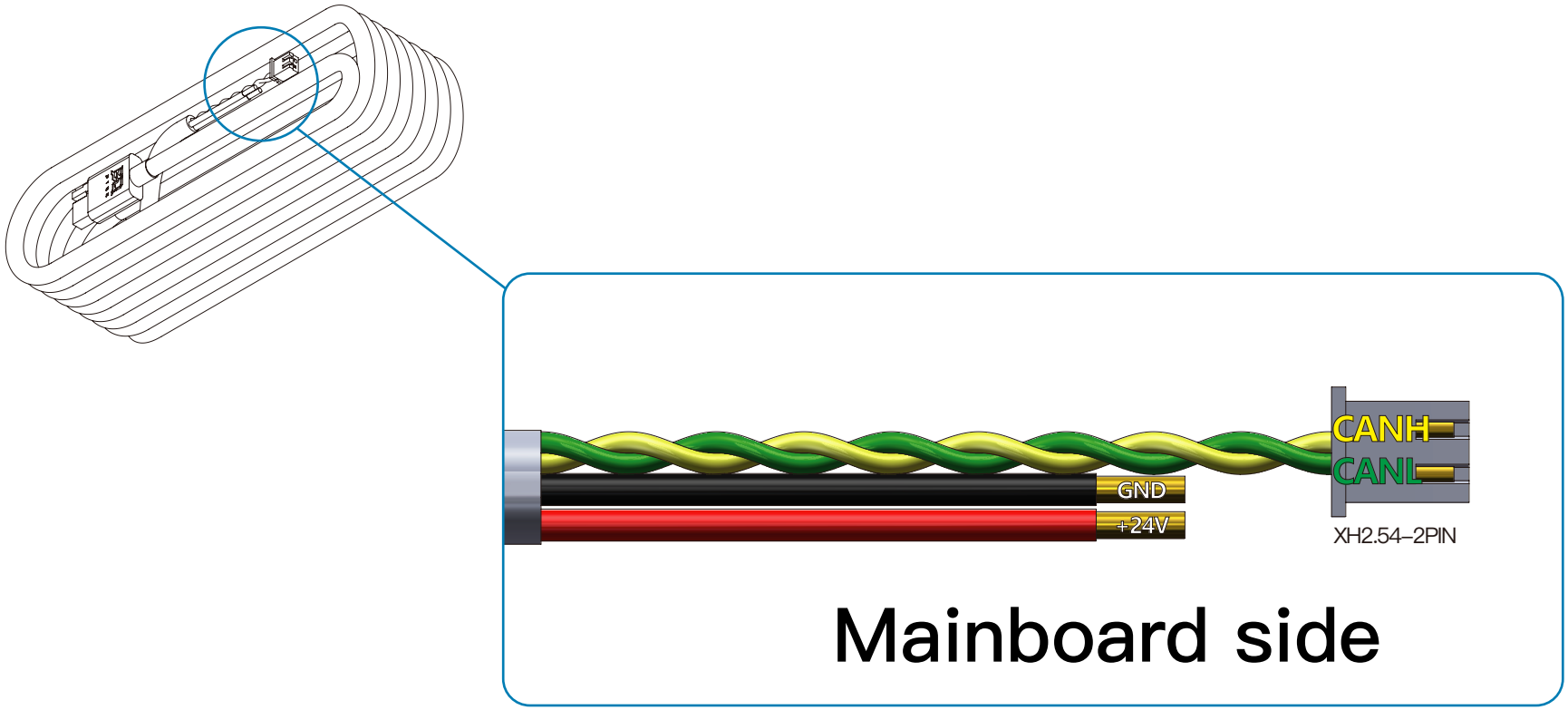
TIPS

The printed part for CAN cable is a custom printed part that can be downloaded directly from our GitHub repository, in EBB\EBB SB2240_2209 CAN\Custom Printed Parts, as well as the printed part for USB-C cable.



CABLE CLIP

The cable clip printed part can be downloaded from our GitHub repository, in EBB\EBB SB2240_2209 CAN\Custom Printed Parts, or you can use zip ties instead.

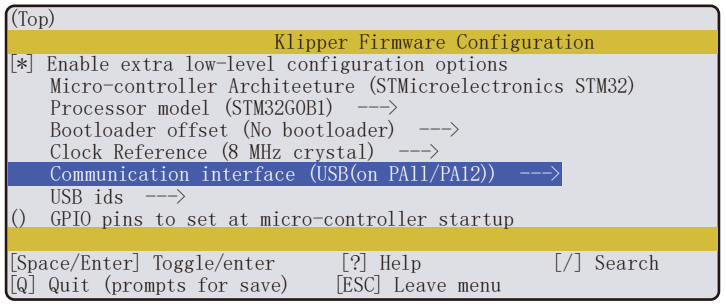

CANH
CANL
GND
+24V
XH2.54-2PIN
Mainboard side

1.) **After SSH is successfully connected to Raspberry Pi, run**

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

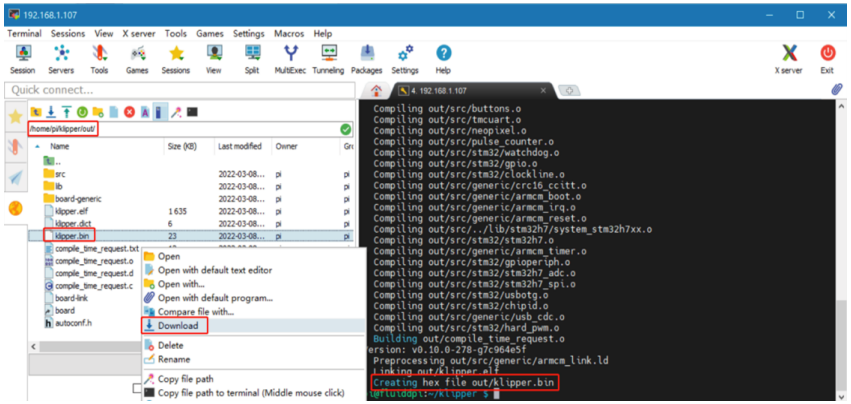
Compile the firmware with the following configuration (if the options below are not available, please update your Klipper source code to the newest version).

[*] Enable extra low-level configuration options
    Micro-controller Architecture (STMicroelectronics STM32) --->
    Processor model (STM32G0B1) --->
If you do not use CanBoot
    Bootloader offset (No bootloader) --->
If CanBoot is used
    Bootloader offset (8KiB bootloader) --->
    Clock Reference (8 MHz crystal) --->
If USB communication on Type-C is used
    Communication interface (USB (on PA11/PA12)) --->
If CAN-Bus communication is used
    Communication interface (CAN bus (on PB0/PB1)) --->
    (1000000) CAN bus speed



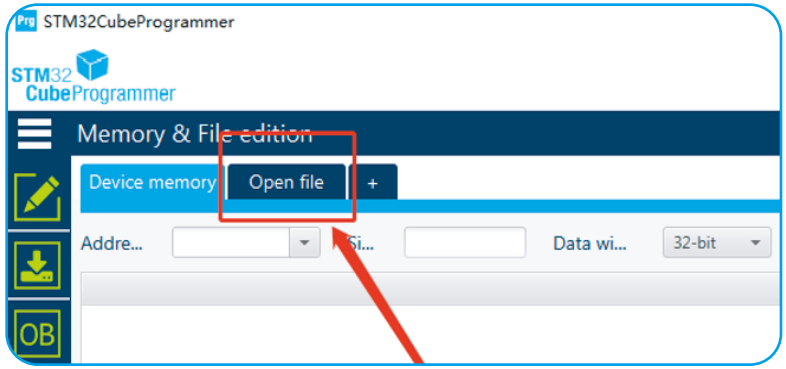
2.)Press 'q' to exit, and "Yes" when asked to save the configuration.

3.)Run `make` to compile firmware, "klipper. bin" file will be generated in `home/pi/klipper/out` folder when `make` is finished, download it onto your computer using the SSH application.

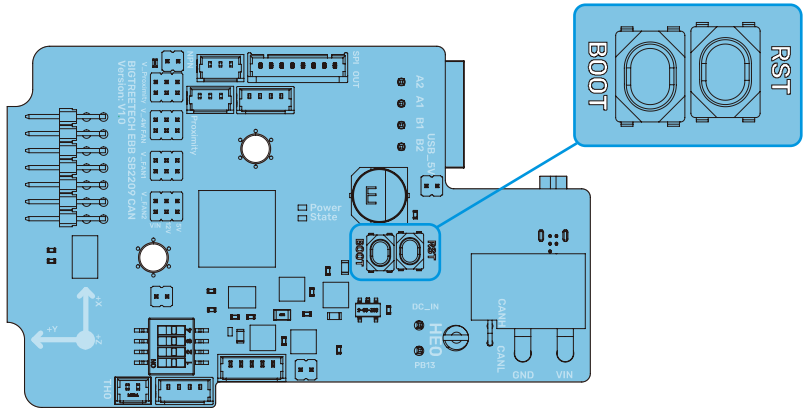


Update directly through computer.

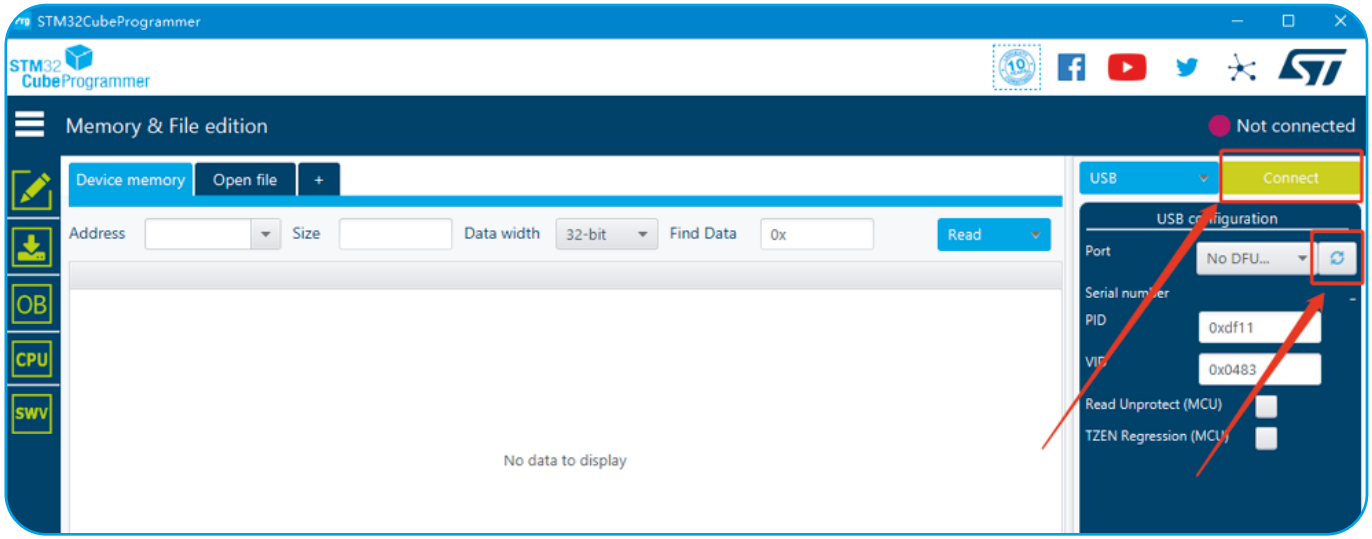
1.Open the STM32CubeProgrammer and select the firmware to download (klipper. bin).



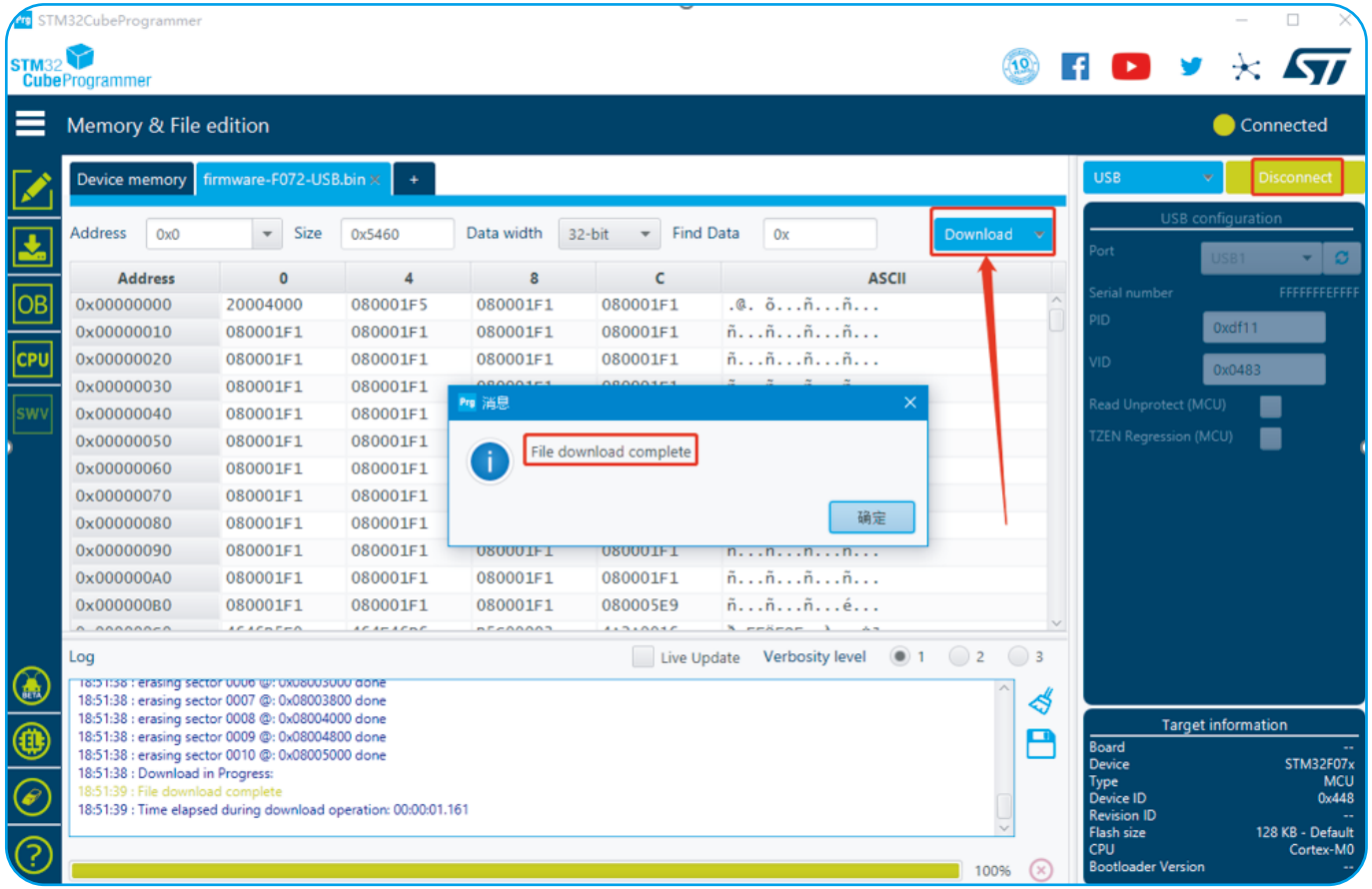
2.Press and hold the **Boot** button, and then click the **Reset** button to enter the DFU mode.



3. Click the **Refresh** icon in STM32CubeProgrammer until the Port changes from "No DFU d..." to "USB1", and then click **Connect** to connect to the chip.

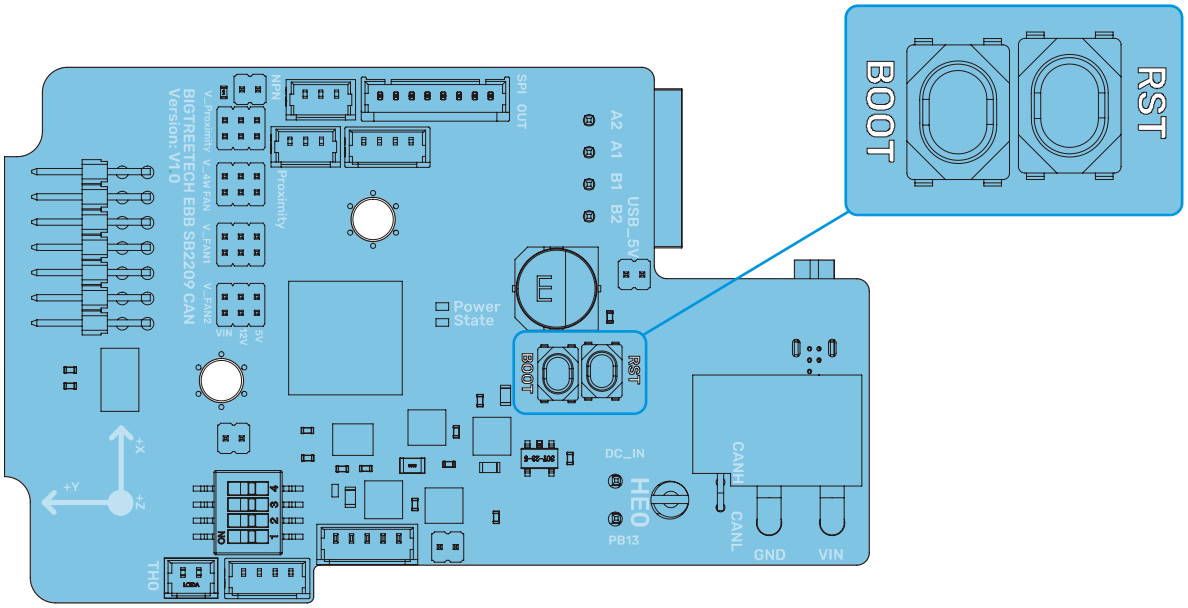


4. After the connection is successful, **Connect** will become **Disconnect**, and then click **Download** to start downloading the program. After the download is completed, a pop-up window of "File download complete" will appear, indicating that the writing is successful.



Raspberry Pi or CB1 update via DFU.

1.Press and hold the **Boot** button, and then click the **Reset** button to enter the DFU mode.



2.Enter in the SSH terminal command line

```
lsusb
```

Query DFU device ID

```
pi@fluiddpi:~ $ lsusb
Bus  001  Device  005:   ID   0483:df11  STHicroolactronics STM Device in DFU Mode
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@fluiddpi:~ $
```

3.Run:

```
cd klipper
```

to enter to the klipper directory, then run the following command to write the firmware:

```
make flash FLASH_DEVICE=0483:df11
```

Note: Replace 0483: df11 with the actual device ID found in the previous step.

4. The prompt "`File downloaded successfully`" indicates that the writing is completed.
After the writing is completed, there will be an error message: **dfu-util: Error during download get_status**, just ignore it.

5. After the firmware is written, run

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

to query the serial ID of the device (this ID can only be

found in the USB communication mode, and this step is

ignored in CANBus mode).

6. If USB communication is used, after the first writing,

it is not necessary to manually press the Boot and

Reset buttons to enter the DFU mode when updating again.

You can directly enter

```
make flash FLASH_DEVICE=/dev/serial/by-id/usb-Klipper_stm32g0b1xx_0F0033000C504B4633373520-if00
```

to write the firmware (Note: replace/dev/serial/by id/xxx with the actual ID found in the previous step).

7. The prompt " `File downloaded successfully` " indicates that the writing is completed.

After the writing is completed, there will be an error message: **dfu-util: Error during download get_status**, just ignore it.

Raspberry Pi or CB1 updated via CanBoot.

Refer to the instructions here to download the CanBoot project
https://github.com/Arksine/CanBoot

1.Run:

```
cd ~
```

to enter the home directory, then run:

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

to download CanBoot project.
run:

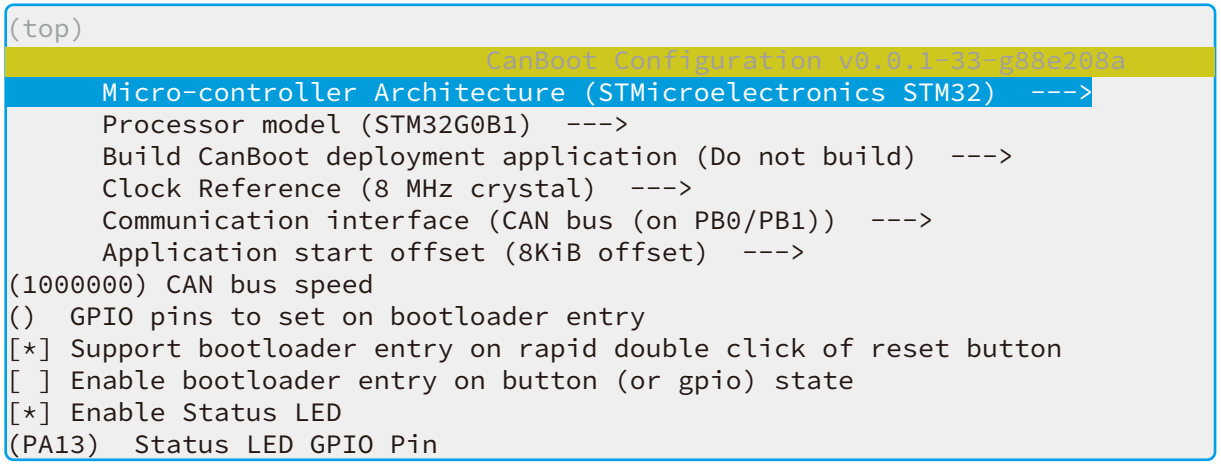
```
cd CanBoot
```

to enter the CanBoot directory.

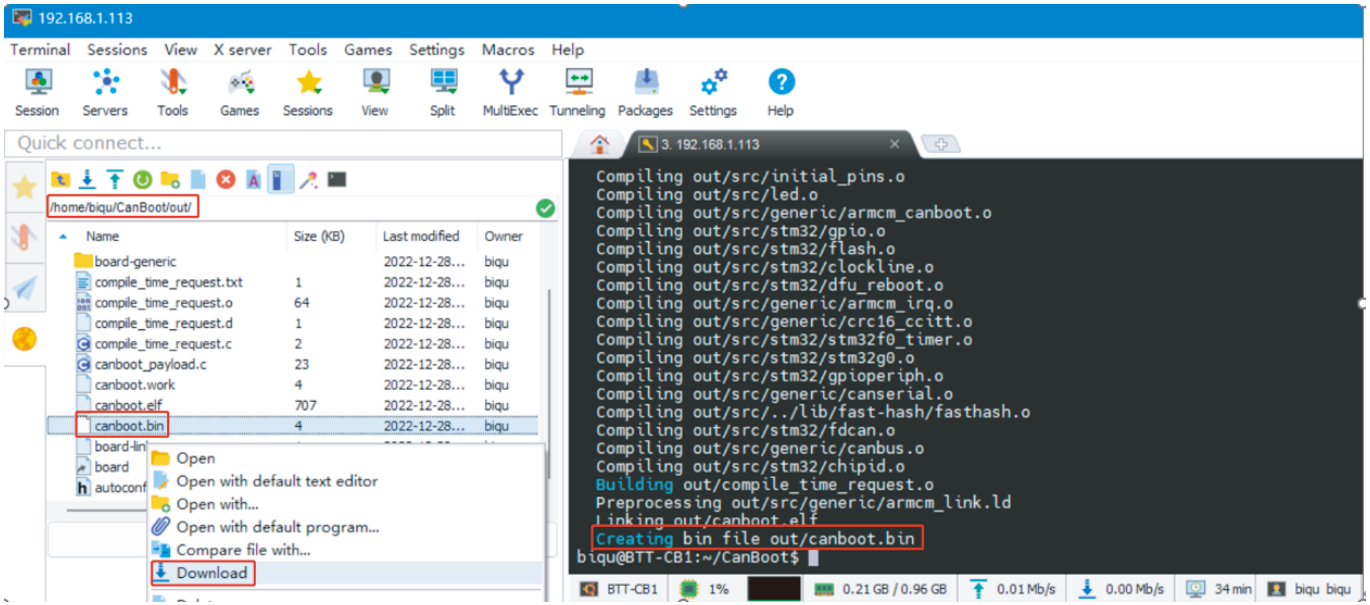
2.Run:

```
make menuconfig
```

and configure according to the following figure

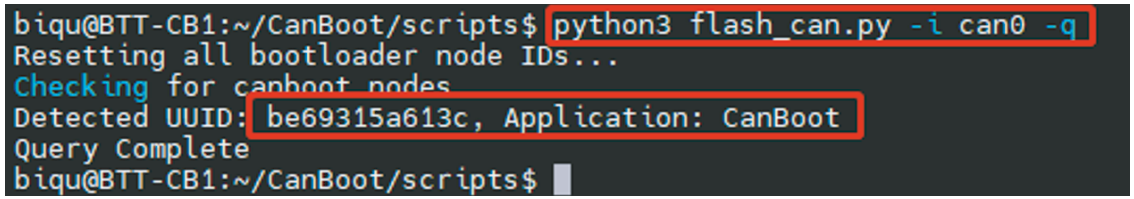


3.Run
`make`
to compile firmware, 'canboot.bin' file will be generated in `home/pi/klipper/out` folder when `make` is finished, download it onto your computer using the SSH application.



4.Use STM32CubeProgrammer software to write 'canboot. bin' to the chip.

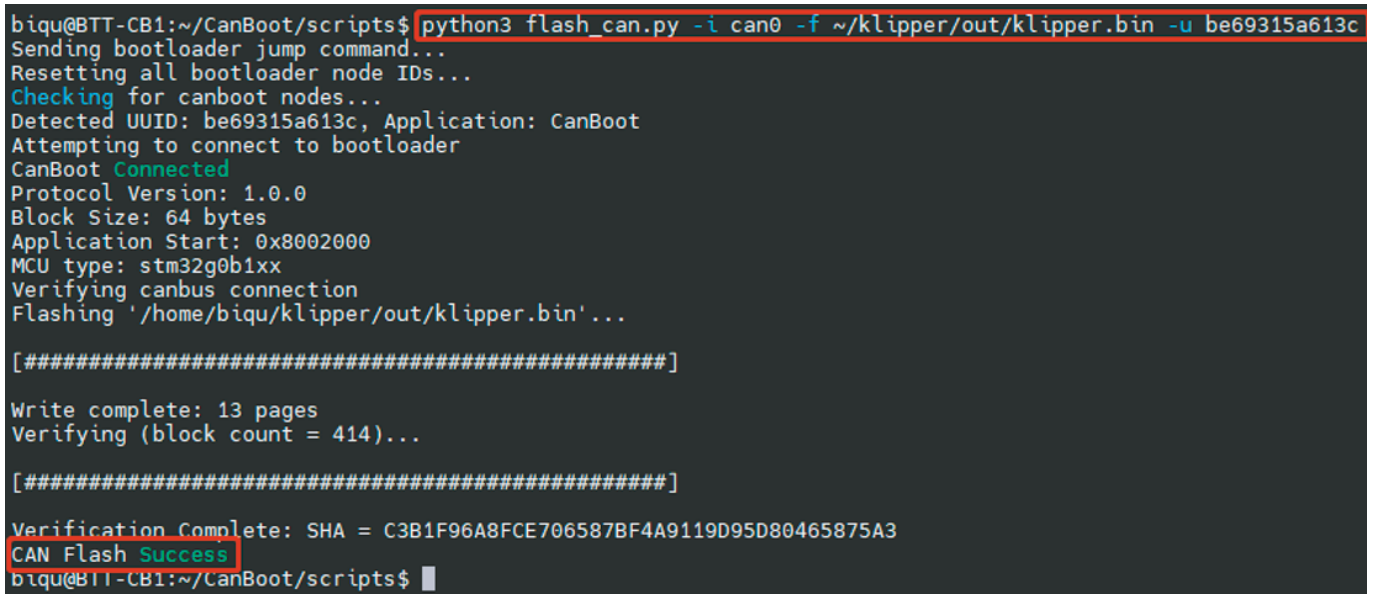
5.Run:
`cd ~/CanBoot/scripts`
then run:
`python3 flash_can.py -i can0 -q`
to query the canbus ID (connect the CAN cable and power it on in advance), as shown in the figure on the right, the UUID of the device has been found.



6.Run:

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

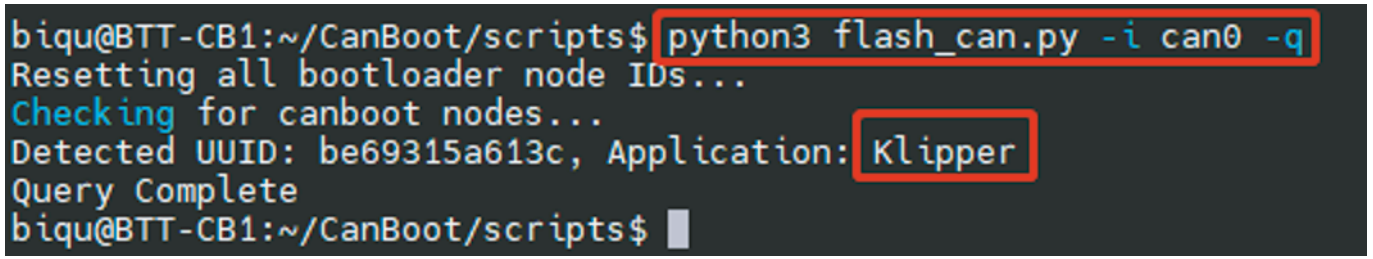
The be69315a613c is replaced with the actual UUID. Note: klipper.bin needs to be made in advance, and the application start offset of CanBoot is 8KiB offset, so Klipper's menuconfig Bootloader offset should also be 8KiB bootloader, as shown in the following figure.



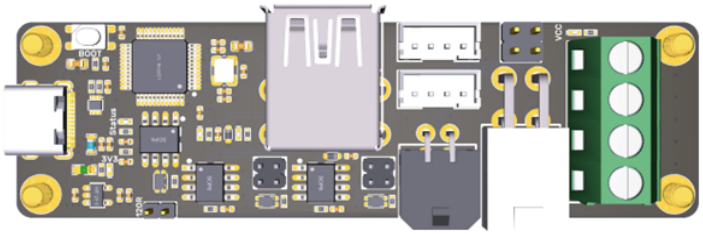
6.Run:

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

to query. At this time, the Application changes from CanBoot to Klipper, indicating that Klipper has been running normally.



Used with BIGTREETECH U2C module.



1. Enter the command

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

in the SSH terminal and execute

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

Set the CAN-BUS speed to 1M (it must be consistent with the speed set in the firmware (**1000000) CAN bus speed**), save (Ctrl+S) and exit (Ctrl+X) after modification, and enter

```
sudo reboot
```

to restart Raspberry Pi.

2. Each device on CAN bus will generate a canbus_uuid according to the UID of MCU, to find each microcontroller device ID, make sure the hardware is powered on and wired correctly, and then run:

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

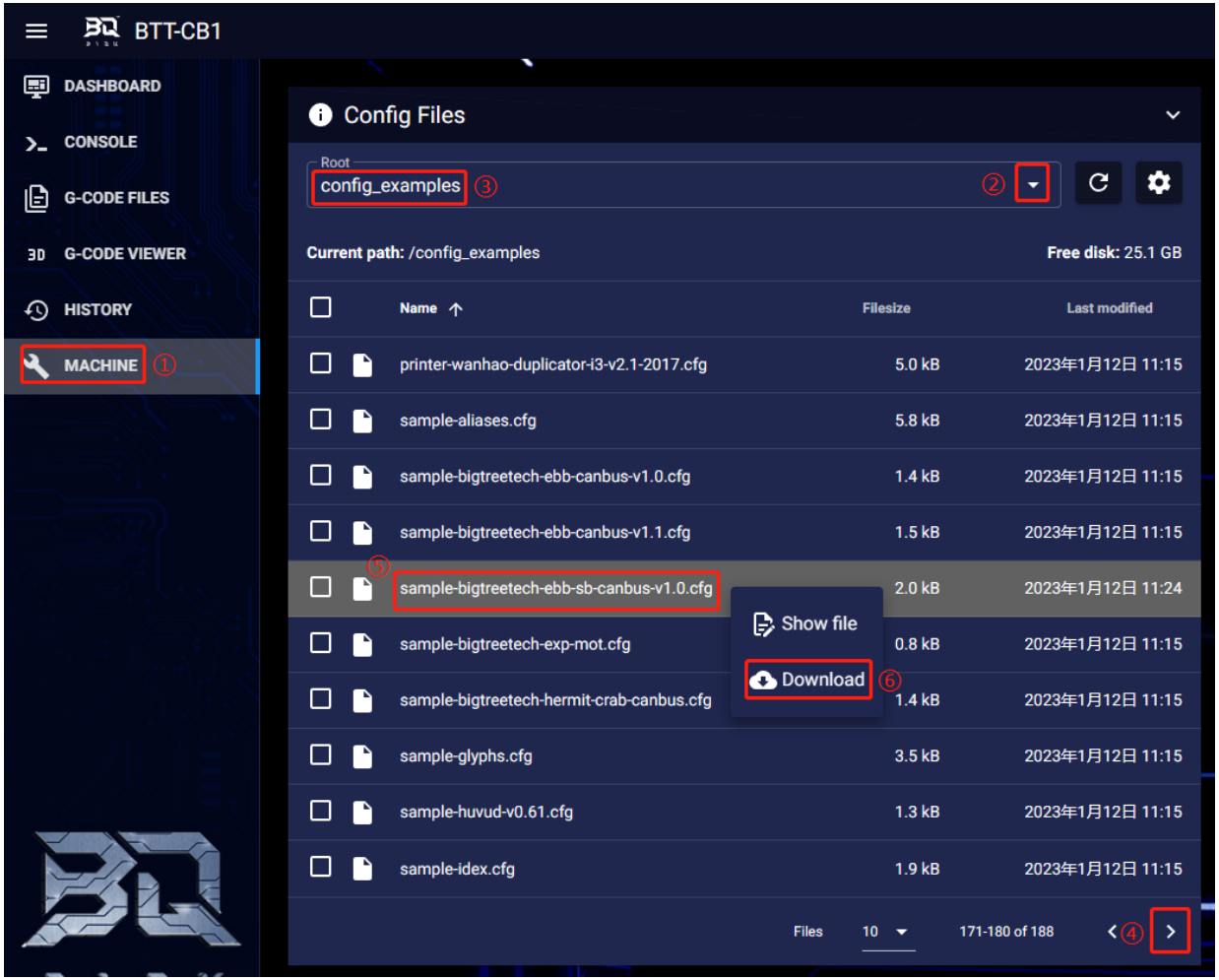
3. If an uninitialized CAN device is detected, the above command will report the device's canbus_uuid.

```
Found canbus_uuid=0e0d81e4210c
```

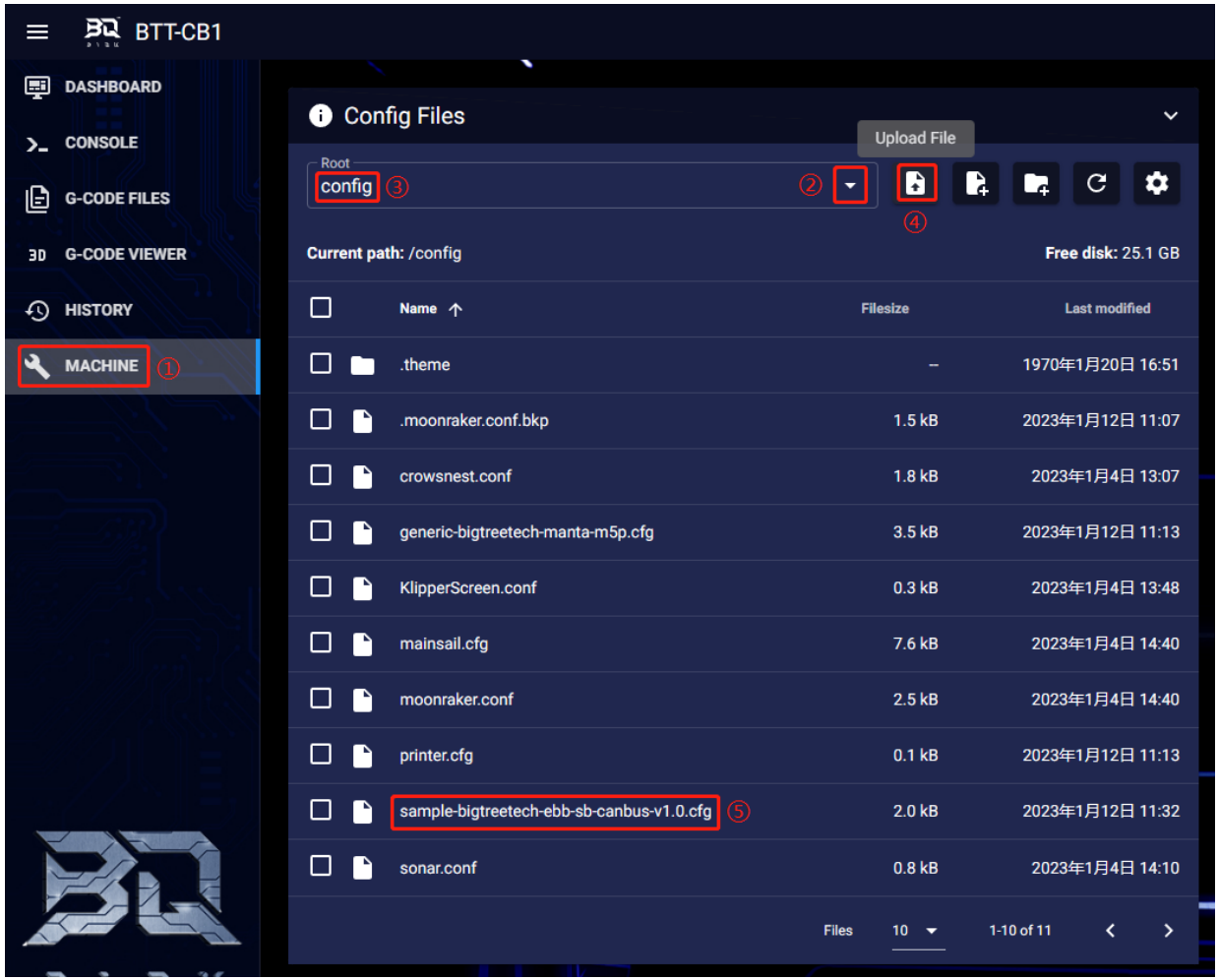
4. If Klipper has been running normally and connected to this device, then canbus_uuid will not be reported, which is normal.

1. Enter Raspberry Pi IP address into your browser, and find the reference config for the motherboard in the directory shown below, if there is no such config available, update your Klipper source code to the newest version or download it from GitHub:

`https://github.com/bigtreetech/EBB`



2. Upload the configuration file of the motherboard to Configuration Files.



3. Add the configuration of this motherboard in the "printer. cfg" file:

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

printer.cfg

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

4. Enter the correct ID (USB serial or canbus).

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. Configure the module's specific functions according to https://www.klipper3d.org/Overview.html

Website
www.bigtree-tech.com

GitHub
www.github.com/bigtreetech

Discord
www.discord.gg/5jdwbYYZuv