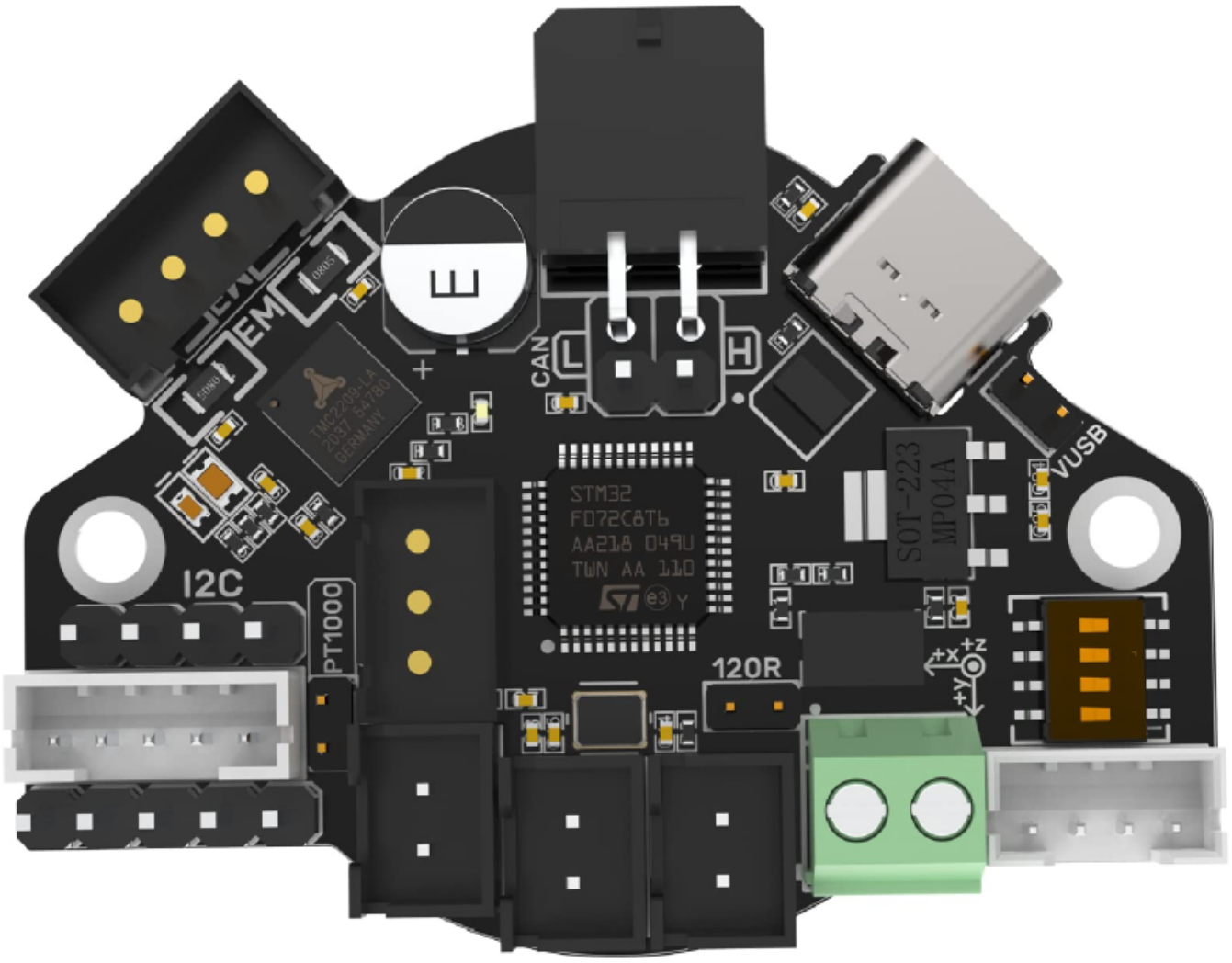
YTVI BTT EBB36 / EBB42 (USB)

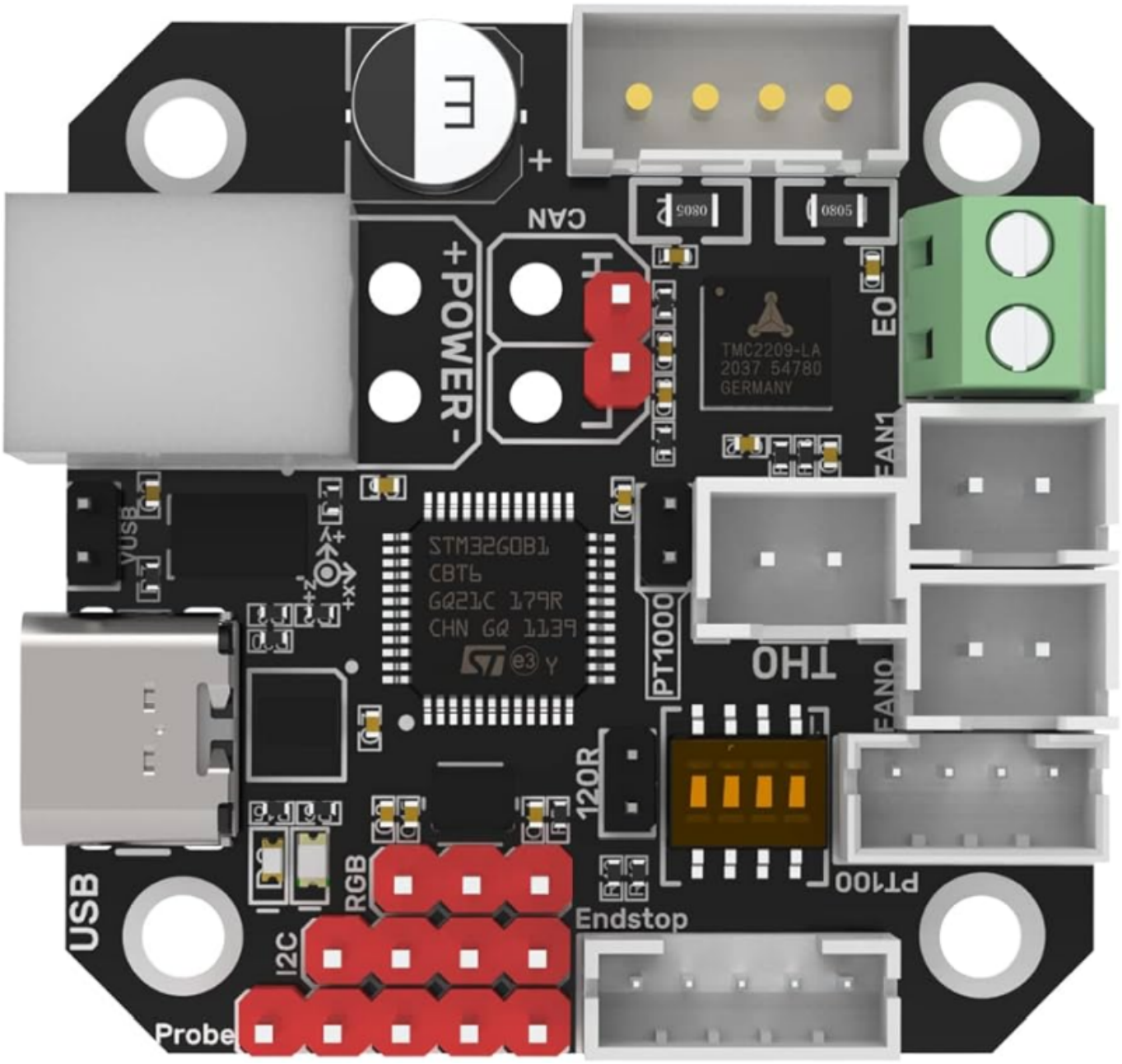
Schrittweise Anleitung, um das Bigtree Tech **EBB36 / EBB42** Kopfboard mit dem BTT Buskoppler **U2C** über **USB** in Betrieb zu nehmen.

> **Hinweise**
> 1) Diese Anleitung beschreibt beide EBB Boards (36 & 42) und das in allen derzeit verfügbaren Varianten (V1.0, V1.1, V1.2).
> 2) Variante 1.0 hat einen STM32F072C8T6 Controller. Die nachfolgenden Modelle einen STM32G0B1CBT6.
> 3) Die Modelle der Versionen (also als Beispiel EBB36 V1.0 & EBB42 V1.0) sind Pinkompatibel. Die verschiedenen Versionen sind nicht pinkompatibel!

EBB36



EBB42



YouTube Video #xx

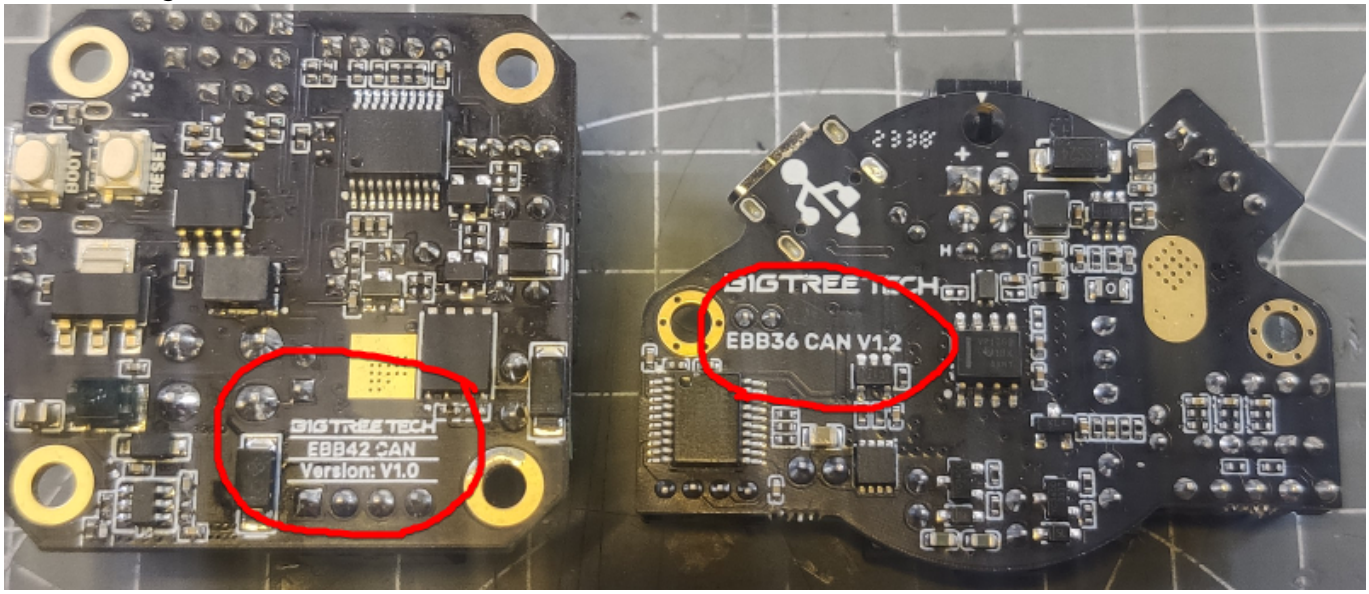


Hinweise

- **SBC** bedeutet in der Anleitung **S**ingle **B**oard **C**omputer. Also meistens wohl ein Raspberry Pi.
- Es wird davon ausgegangen das auf dem SBC Klipper und MainSail eingerichtet ist.
- Ein Zugang zum SBC über SSH ist notwendig!
- **Board mit Version 1.1**
  Hier wird der Pin PA2 beim Booten kurzfristig aktiviert. Da hängt in dieser Version die Ansteuerung für den Extruder Heizblock dran. Beim Flashen also lieber die 24V Versorgung abklemmen, um ein ungewolltes Aufheizen zu vermeiden!

Version ermitteln

Es ist wichtig die Version von dem Board zu ermitteln! Dazu einfach auf der Rückseite nachsehen:

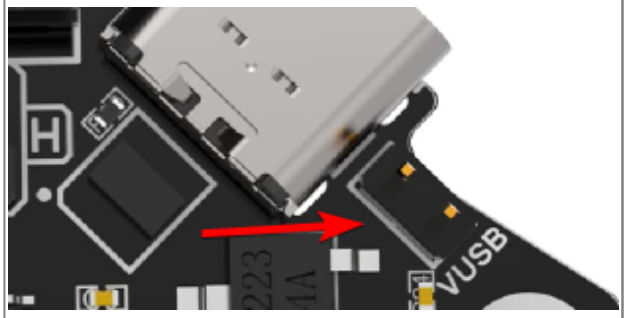
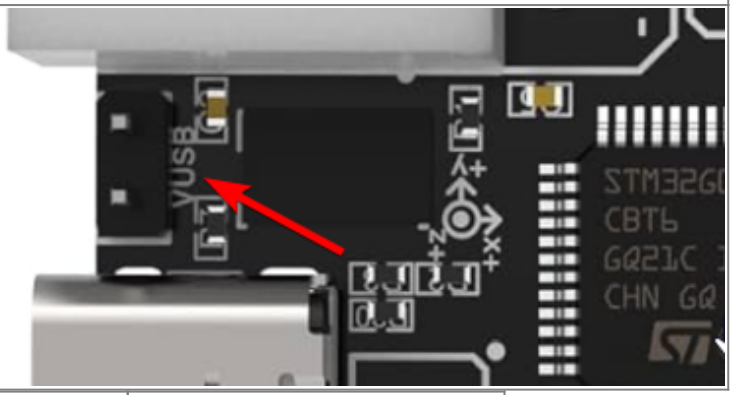


Links ist das EBB42, rechts ist das EBB36.

Die Versionen haben unterschiedliche Controller verbaut und teilweise unterschiedliche Pinbelegungen!

Stromversorgung

Auf dem Board gibt es einen Mikro Jumper, den man setzen kann:

EBB36	**EBB42**
	

Jumper	**Stromversorgung Controller**	**Verwendung**
gesetzt	5V Stromversorgung über USB Anschluss	Board liegt “auf dem Tisch”
NICHT gesetzt	5V Stromversorgung über 12/24V Stecker	Board ist im Kopf eingebaut

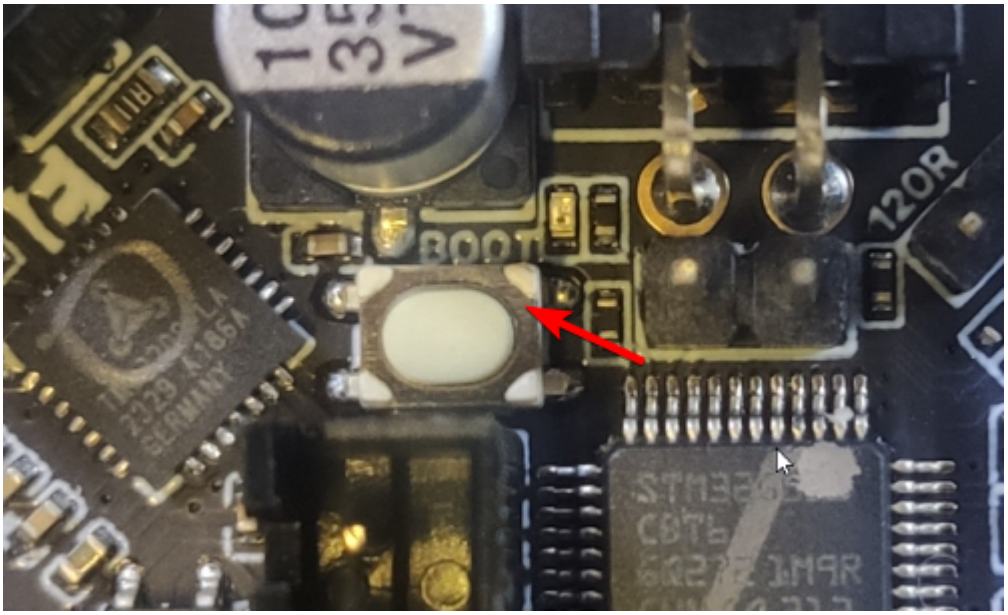
Als Faustregel im Betrieb den 5V Jumper immer abziehen!

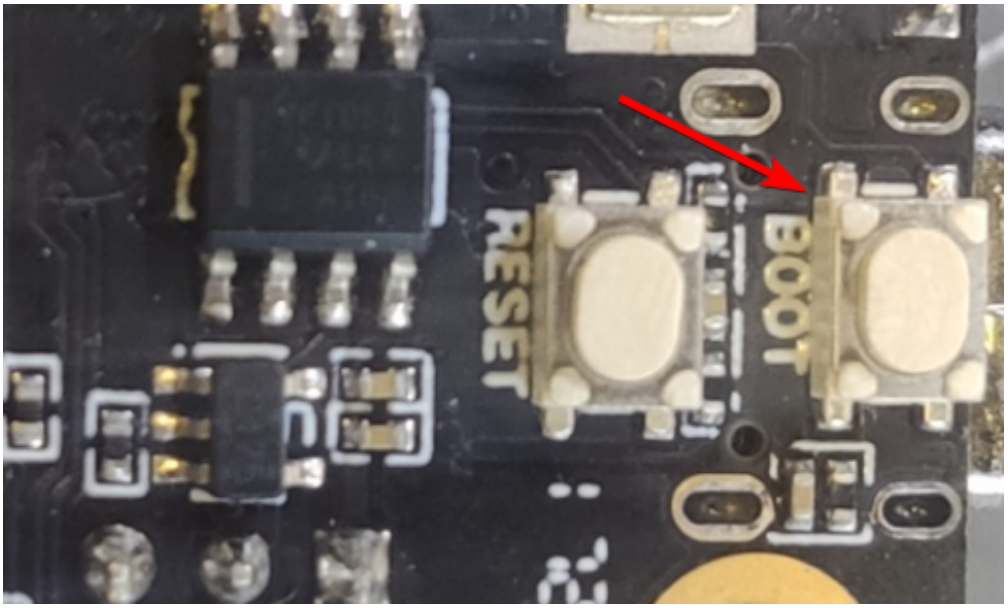
Verkabelung

- Ganz normale USB Verkabelung
- GND & 12 bzw. 24V extra zum Board führen

DFU Modus

- Das Board in den DFU Modus bringen
  - `dmesg -HW`
  - Die **Boot Taste** am Board gedrückt halten und dabei das Board über USB mit dem SBC verbinden

EBB36 : 

EBB42 : 

Wenn das Board schon über 12/24V angeschlossen ist geht auch folgendes
Boot Taste gedrückt halten, **RST Taste** einmal drücken und loslassen, dann die **Boot Taste** loslassen.

- Das Boiard meldet sich mit **`Product: DFU in FS Mode`**

```
pi@Pi3Test:~/katapult $ dmesg -HW
[Nov11 07:34] usb 1-1.4: new full-speed USB device number 17 using
dwc_otg
[  +0.133427] usb 1-1.4: New USB device found, idVendor=0483,
idProduct=df11, bcdDevice= 2.00
[  +0.000037] usb 1-1.4: New USB device strings: Mfr=1, Product=2,
SerialNumber=3
[  +0.000016] usb 1-1.4: Product: DFU in FS Mode
[  +0.000013] usb 1-1.4: Manufacturer: STMicroelectronics
[  +0.000012] usb 1-1.4: SerialNumber: 206A39785542
```

- STRG+C drücken, um die Meldungen zu beenden

Klipper flashen

> **ACHTUNG** Auf die Version achten bei `make menuconfig`!

- `cd ~/klipper`
- `make menuconfig` für **Version 1.0**

```
[*] Enable extra low-level configuration options
    Micro-controller Architecture (STMicroelectronics STM32)  --->
    Processor model (STM32F072)  --->
    Bootloader offset (No bootloader)  --->
    Clock Reference (8 MHz crystal)  --->
    Communication interface (USB (on PA11/PA12))  --->
    USB ids  --->
()  GPIO pins to set at micro-controller startup
```

- `make menuconfig` für **Version 1.1 & 1.2**

```
[*] Enable extra low-level configuration options
    Micro-controller Architecture (STMicroelectronics STM32)  --->
    Processor model (STM32G0B1)  --->
    Bootloader offset (No bootloader)  --->
    Clock Reference (8 MHz crystal)  --->
    Communication interface (USB (on PA11/PA12))  --->
    USB ids  --->
()  GPIO pins to set at micro-controller startup
```

  - beenden mit Q und Y
- Klipper kompilieren
  `make -j4`
- Klipper flashen
  `dfu-util -R -a 0 -s 0x08000000:mass-erase:force -D ~/klipper/out/klipper.bin`
  Das Ergebnis sollte sein `File downloaded successfully`

Port ermitteln

- Den USB Stecker abziehen
- `dmesg -HW` starten

```
pi@Pi3Test:~/klipper $ dmesg -HW
[Nov11 15:15] usb 1-1.4: new full-speed USB device number 8 using
dwc_otg
[  +0.134623] usb 1-1.4: New USB device found, idVendor=1d50,
idProduct=614e, bcdDevice= 1.00
[  +0.000034] usb 1-1.4: New USB device strings: Mfr=1, Product=2,
SerialNumber=3
[  +0.000016] usb 1-1.4: Product: stm32g0b1xx
[  +0.000012] usb 1-1.4: Manufacturer: Klipper
[  +0.000012] usb 1-1.4: SerialNumber: 3F003A000E50425539393020
[  +0.075794] cdc_acm 1-1.4:1.0: ttyACM0: USB ACM device
[  +0.000134] usbcore: registered new interface driver cdc_acm
[  +0.000010] cdc_acm: USB Abstract Control Model driver for USB modems
and ISDN adapters
```

  - Wir brauchen die Information mit **tty...** also in diesem Fall **ttyACM0**
  - STRG+C drücken, um die Meldungen zu beenden
- Den zugehörigen Link ermitteln
  `ls -lR /dev/ | grep -v '\->\s../tty' | grep -e 'tty[[:alpha:]]' -e serial`
  - Wir brauchen die Info unter `/dev/serial/by-id:` :
    `lrwxrwxrwx 1 root root 13 Nov 11 15:15 usb-Klipper_stm32g0b1xx_3F003A000E50425539393020-if00 → ../../ttyACM0`
  - Achte darauf das am Ende die gleiche tty Bezeichnung steht wie sie im vorherigen Schritt ermittelt wurde (hier also ttyACM0)
  - Was für für die Konfig brauchen ist dann am Ende:

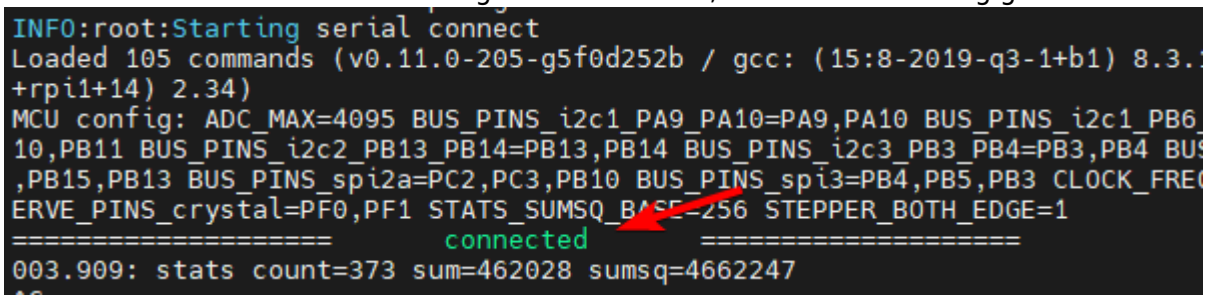
```
/dev/serial/by-id/usb-Klipper_stm32g0b1xx_3F003A000E50425539393020-if00
```

kurzer Test

Ob das Board korrekt mit Klipper läuft, lässt sich mit folgendem Befehl schnell testen:
`~/klippy-env/bin/python ~/klipper/klippy/console.py /dev/serial/by-id/usb-Klipper_stm32g0b1xx_3F003A000E50425539393020-if00`

Der Pfad am Ende muss natürlich mit dem übereinstimmen was ihr im vorherigen Schritt ermittelt habt!

Wenn ihr ein **`connected`** am Anfang des Textes seht, ist das Board richtig geflasht.



Konfig

- `cd ~/printer_data/config`
- **Version 1.0**
  - `wget https://raw.githubusercontent.com/bigtreetech/EBB/master/EBB%20CAN%20V1.0%20(STM32F072)/sample-bigtreetech-ebb-canbus-v1.0.cfg -O BTT_EBB.cfg`
- **Version 1.1**
  - `wget https://raw.githubusercontent.com/bigtreetech/EBB/master/EBB%20CAN%20V1.1%20(STM32G0B1)/sample-bigtreetech-ebb-canbus-v1.1.cfg -O BTT_EBB.cfg`
- **Version 1.2**
  - `wget https://raw.githubusercontent.com/bigtreetech/EBB/master/EBB%20CAN%20V1.1%20(STM32G0B1)/sample-bigtreetech-ebb-canbus-v1.2.cfg -O BTT_EBB.cfg`
- `nano ~/printer_data/config/BTT_EBB.cfg`

```
[mcu EBBCan]
serial: /dev/serial/by-id/usb-Klipper_stm32g0b1xx_3F003A000E50425539393020-if00
#canbus_uuid: 539892be834d
```

  - Die Zeile mit `serial` entsprechend mit dem ermittelten Pfad von oben anpassen
  - Die Zeile mit `canbus_uuid` auskommentieren

- In der `printer.cfg` noch die neue Konfig einbetten
  `[include BTT_EBB.cfg]`

TBD: wget Link falsch muss in '

Klipper Update

- Klipper Dienst stoppen
  `sudo systemctl stop klipper.service`
- `cd ~/klipper && make menuconfig`
  - Die Einstellungen sind genauso wie im Kapitel Klipper flashen
- `make flash -j4 FLASH_DEVICE=/dev/ttyACM0`
  - Wie man an den Port (hier ttyACM0) kommt, ist hier beschrieben
  - Am Ende kann es zu einem Fehler kommen. Davon nicht irritieren lassen. Wichtig ist diese Zeile:
    `File downloaded successfully`
- Einmal Reset-Taste drücken am Board hilft 🙂
- Klipper Dienst starten
  `sudo systemctl start klipper.service`

Links

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

From:
https://www.drklipper.de/ - **Dr. Klipper Wiki**

Permanent link:
https://www.drklipper.de/doku.php?id=klipper_faq:flash_guide:stm32g0b1:btt_ebb36_ebb42_usb

Last update: **2024/01/31 04:45**