ZeroPointModule von pkelectronics

YouTube Video #47

Nutzerkreis

Grundsätzlich kann das ZPM überall da eingesetzt werden, wo CAN mit einem Raspberry zusammen benötigt wird. Bei den Druckern passt es besonders gut in die kleinen High-End DIY Drucker wie den Voron V0, aber natürlich auch in den Micro oder den Salad Fork. Generell kann man sagen eignet sich das ZPM überall da, wo der Platz eher knapp ist und ein Fullsize Raspberry + CAN Adapter + 5V Netzteil nur schwer unter zubekommen ist.

Shop

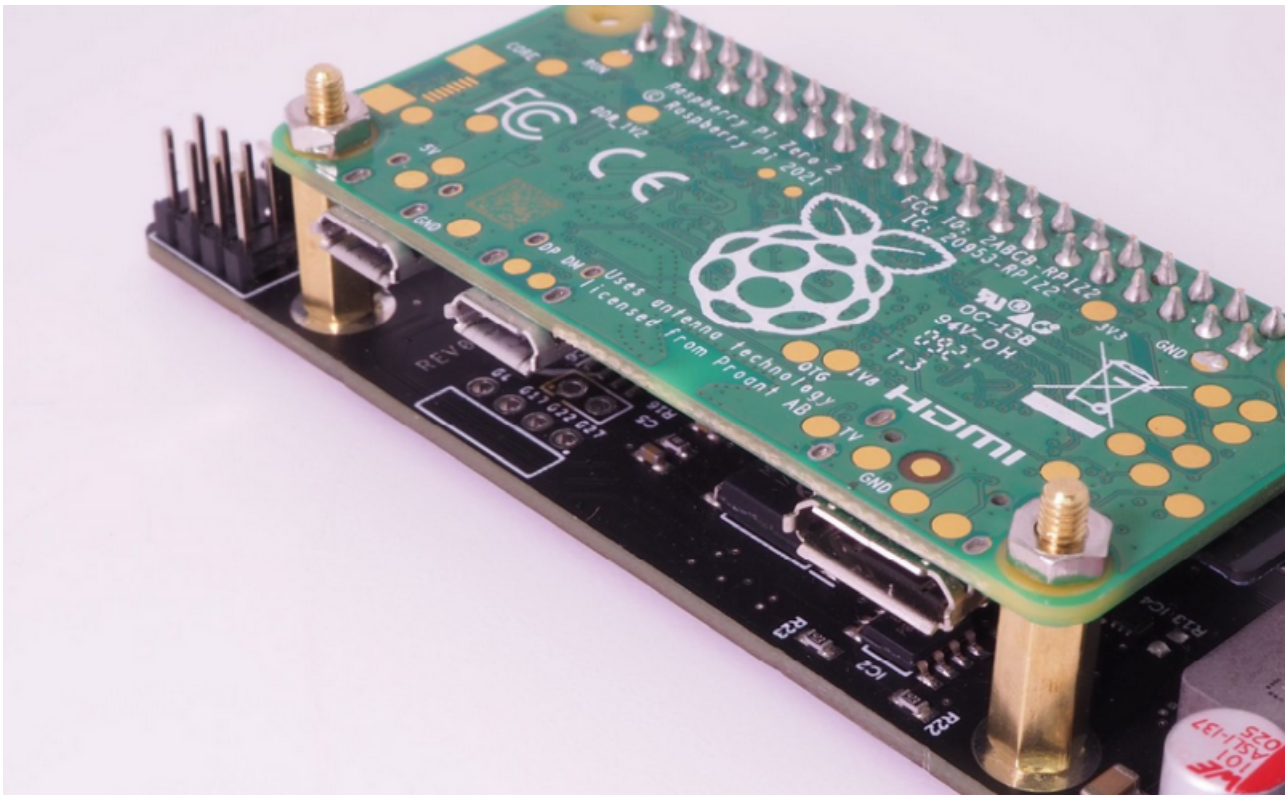
https://ib-wistinghausen.de/produkt/zeropointmodule/

Eigenschaften

- 5V 5A/3A DCDC converter powering the Pi and added peripherals
- MCP2515 CAN-Bus Controller and corresponding transceiver
- Two hardware PWM channels with low-side mosfet switches
- SPI, Uart and I2C connections broken out to dedicated connectors
- Two connectors for 4-pin PWM Fans (5V/24V selectable)
- low profile form-factor possible
- eeprom based autoconfiguration of IOs and kernel modules

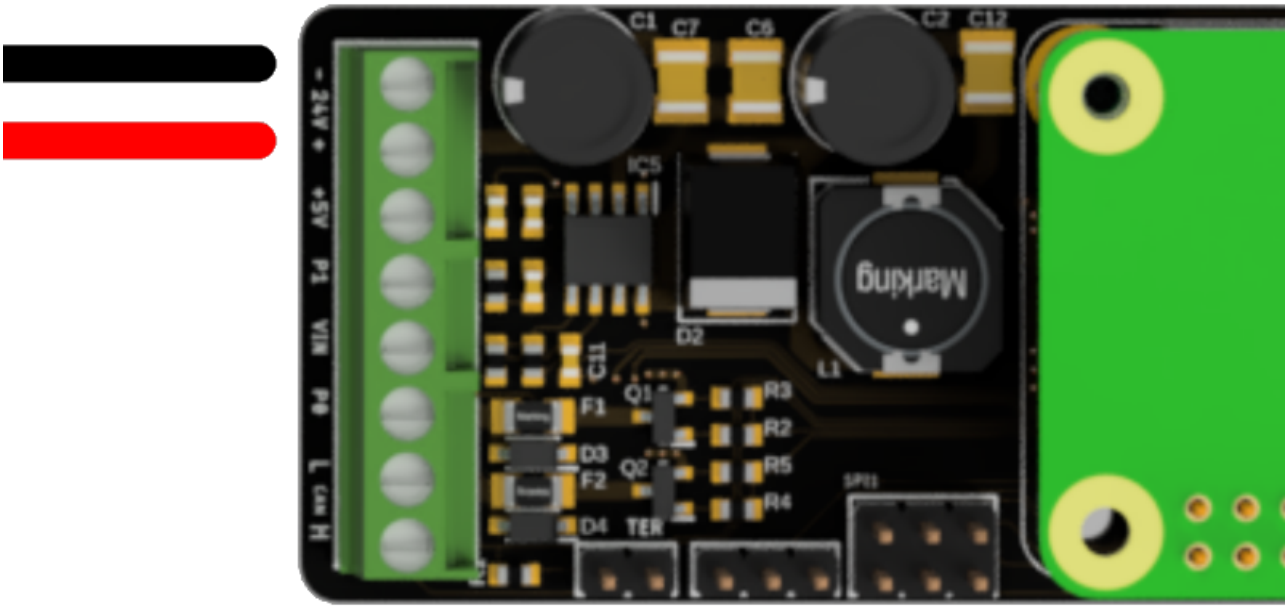
Grundinstallation

- Den Raspberry Pi Zero auf dem ZeroPointModul aufstecken (auf korrekte Einbaulage achten !)



Die Ports (HDMI, USB) zeigen Richtung ZeroPointModul.

- 24V Spannungsversorgung am Board anschließen



Hinweis

Die Platine kann auch mit weniger als 24V betrieben werde. 12V lief im Test auch stabil.

- SD-Karte vorbereiten
  - Installation über MainsailOS
  - Installation über kiauh
- SD-Karte in den Raspberry Pi einstecken und Spannungsversorgung einschalten
- per SSH einloggen und Updates / Tools installieren

```
sudo apt update && sudo apt upgrade -y && sudo apt install -y git
silversearcher-ag wavemon hexedit sudoku tcpdump iptraf mc htop dcfldd
nano usbutils ranger tldr ncdu can-utils multitail fd-find && mkdir -p
```

```
~/.local/share && tldr -u
```

- ungenutzte Dienste entfernen
  ```
  sudo apt autoremove -y modem* cups* pulse* avahi* triggerhappy*
  ```

Betrieb mit UART

- Serielle Schnittstelle aktivieren
  ```
  sudo raspi-config
  ```
  - → 3 Interface Options
  - → I6 Serial Port
  - Would you like a login shell to be accessible over serial? → No
  - Would you like the serial port hardware to be enabled? → Yes
- Bluetooth abschalten
  ```
  sudo nano /boot/config.txt
  ```
  - unter `[all]` folgenbden Eintrag hinzufügen `dtoverlay=disable-bt`
- `sudo reboot`
- Firmware mit Serial als Communication interface
  ```
  Communication interface (Serial (on USART1 PAXX/PAXX)) --->
  ```
  ```
  (Top) → Communication interface

  ( ) USB (on PA11/PA12)
  (X) Serial (on USART1 PA10/PA9)
  ( ) Serial (on USART1 PB7/PB6)
  ( ) Serial (on USART2 PA3/PA2)
  ( ) Serial (on USART2 PD6/PD5)
  ( ) Serial (on USART3 PB11/PB10)
  ( ) Serial (on USART3 PD9/PD8)
  ( ) CAN bus (on PA11/PA12)
  ( ) CAN bus (on PA11/PB9)
  ( ) CAN bus (on PB8/PB9)
  ( ) CAN bus (on PI9/PH13)
  ( ) CAN bus (on PB5/PB6)
  ( ) CAN bus (on PB12/PB13)
  ( ) CAN bus (on PD0/PD1)
  ( ) USB to CAN bus bridge (USB on PA11/PA12)
  ```
- Seriellen Port in Linux finden
  ```
  ls -lR /dev/ | grep -v '\->\s../tty' | grep -e 'tty[[:alpha:]]' -e
  serial
  ```
- Kommunikation Testen
  ```
  sudo systemctl stop klipper
  ~/klippy-env/bin/python ~/klipper/klippy/console.py /dev/ttyAMA0
  sudo systemctl stop klipper
  ```
- Seriellen Port in printer.cfg eintragen

Betrieb mit USB

- die Firmware mit USB Support kompilieren

```
(Top) → Communication interface

(X) USB (on PA11/PA12)
( ) Serial (on USART1 PA10/PA9)
( ) Serial (on USART1 PB7/PB6)
( ) Serial (on USART2 PA3/PA2)
( ) Serial (on USART2 PD6/PD5)
( ) Serial (on USART3 PB11/PB10)
( ) Serial (on USART3 PD9/PD8)
( ) CAN bus (on PA11/PA12)
( ) CAN bus (on PA11/PB9)
( ) CAN bus (on PB8/PB9)
( ) CAN bus (on PI9/PH13)
( ) CAN bus (on PB5/PB6)
( ) CAN bus (on PB12/PB13)
( ) CAN bus (on PD0/PD1)
( ) USB to CAN bus bridge (USB on PA11/PA12)
```

- USB Port verbinden
  - USB-Micro auf USB-A Adapter
  - USB-Hub für Raspberry Pi Zero
- Seriellen Port in Linux finden
  `ls -lR /dev/ | grep -v '\->\s../tty' | grep -e 'tty[[:alpha:]]' -e serial`
- Kommunikation Testen
  `sudo systemctl stop klipper`
  `~/klippy-env/bin/python ~/klipper/klippy/console.py /dev/ttyAMA0`
  `sudo systemctl stop klipper`
- Seriellen Port in printer.cfg eintragen

Betrieb mit CAN

- Die Firmware mit CAN Support und korrekter Geschwindigkeit compilieren und flashen

```
(Top)                                                                Klippe
[*] Enable extra low-level configuration options
    Micro-controller Architecture (STMicroelectronics STM32)  --->
    Processor model (STM32F446)  --->
    Bootloader offset (No bootloader)  --->
    Clock Reference (12 MHz crystal)  --->
    Communication interface (CAN bus (on PD0/PD1))  --->
(500000) CAN bus speed
()  GPIO pins to set at micro-controller startup
```

- Buskoppler prüfen
  `dmesg |grep '251\|can\|spi'`
- CAN erstmal down

```
pi@TestPi:~ $ ip a
2: can0: <NOARP,ECHO> mtu 16 qdisc noop state DOWN group default qlen
10
    link/can
```

- Interface UP
  `sudo ip link set can0 up type can bitrate 500000`
- `sudo nano /etc/network/interfaces.d/can0`

```
auto can0
iface can0 can static
    bitrate 500000
    up ifconfig $IFACE txqueuelen 128
```

-
```
pi@TestPi:~ $ ~/klippy-env/bin/python ~/klipper/scripts/canbus_query.py
can0
Found canbus_uuid=5b5a812a7283, Application: Klipper
Total 1 uuids found
```

- CAN UUID in der `printer.cfg` eintragen

```
[mcu]
restart_method           : command
canbus_uuid              : 5b5a812a7283
```

- Kommunikation Testen
```
sudo systemctl stop klipper
~/klippy-env/bin/python ~/klipper/klippy/console.py -c can0 5b5a812a7283
sudo systemctl stop klipper
```

Extra Hardware über RPi nutzen

- Hardware PWM einrichten
https://www.klipper3d.org/RPi_microcontroller.html
- Problem Neopixel → Timinig Problem
https://github.com/Klipper3d/klipper/issues/3366
- Problem ADXL345
https://forums.raspberrypi.com//viewtopic.php?t=149981

Raspberry PI MCU

- `cd ~/klipper/`
- `sudo cp ./scripts/klipper-mcu.service /etc/systemd/system/`
- `sudo systemctl enable klipper-mcu.service`
- `make menuconfig`

```
[*] Enable extra low-level configuration options
    Micro-controller Architecture (Linux process)  --->
()  GPIO pins to set at micro-controller startup
```

- `sudo usermod -a -G tty pi`
- `sudo service klipper stop`
- `make flash -j4`
- `sudo service klipper start`

Pin / GPIO Übersicht

PIN	Funktion	Position
GPIO2 / SDA	Generel IO / I2C	J1
GPIO3 / SCL	Generel IO / I2C	J1
GPIO4	Generel IO	J8
GPIO5	Generel IO	J5
GPIO6	Generel IO	J6
GPIO7	NC	-
GPIO12 / PWM0	Generel IO / PWM	J3 / J6
GPIO13 / PWM1	Generel IO / PWM	J3 / J5
GPIO14 / TXD0	UART	J1
GPIO15 / RXD0	UART	J1
GPIO16	LED	D6
GPIO17	Generel IO	J8
GPIO18 / SPI1_CS0	Generel IO / SPI	J4
GPIO19 / SPI1_MISO	Generel IO / SPI	J4
GPIO20 / SPI1_MOSI	Generel IO / SPI	J4
GPIO21 / SPI1_SCLK	Generel IO / SPI	J4
GPIO22	Generel IO	J8
GPIO23	Generel IO	J2
GPIO26	LED	D1
GPIO27	Generel IO	J8

Beispiel Konfig

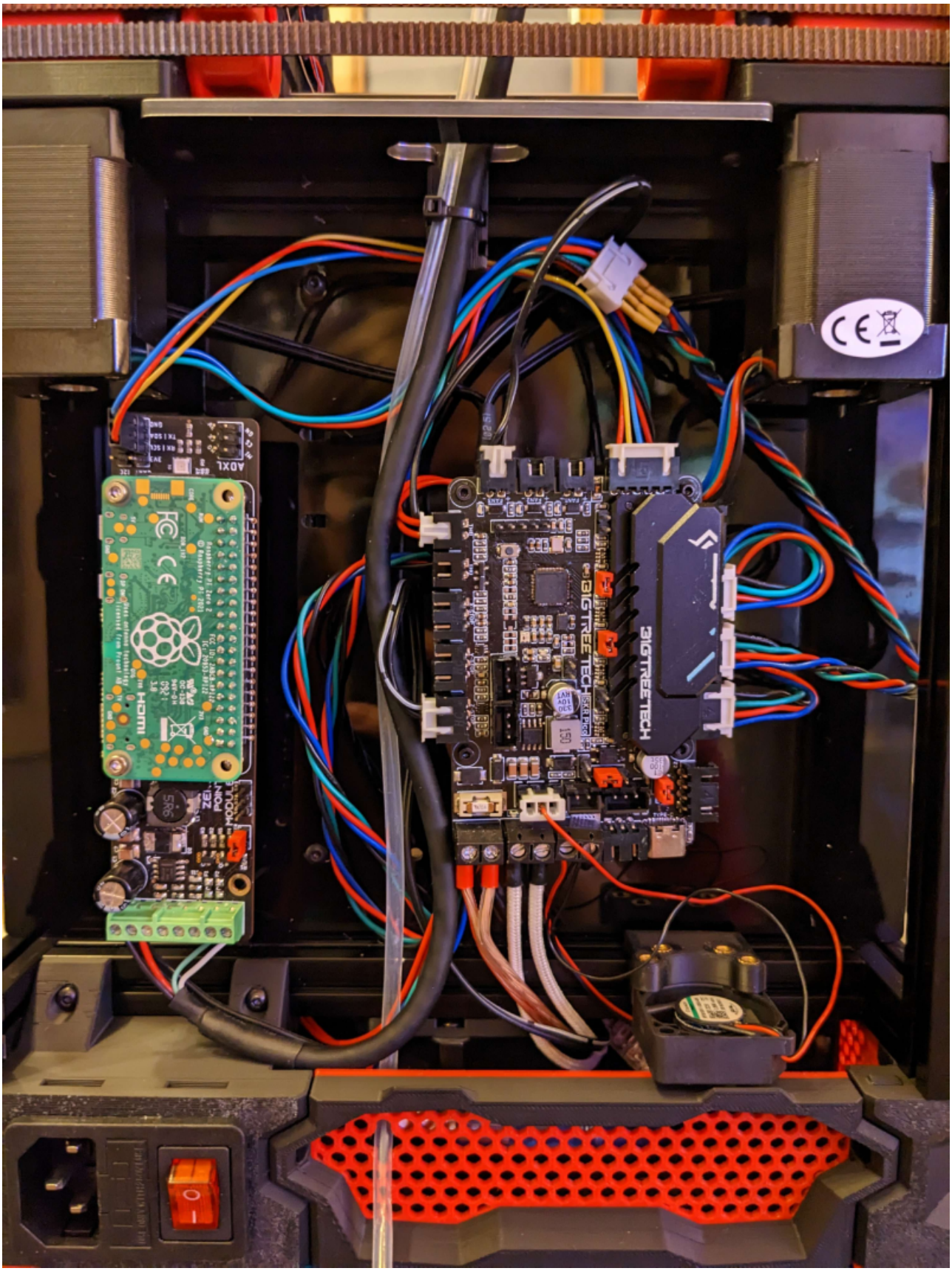
ZeroPoint.cfg

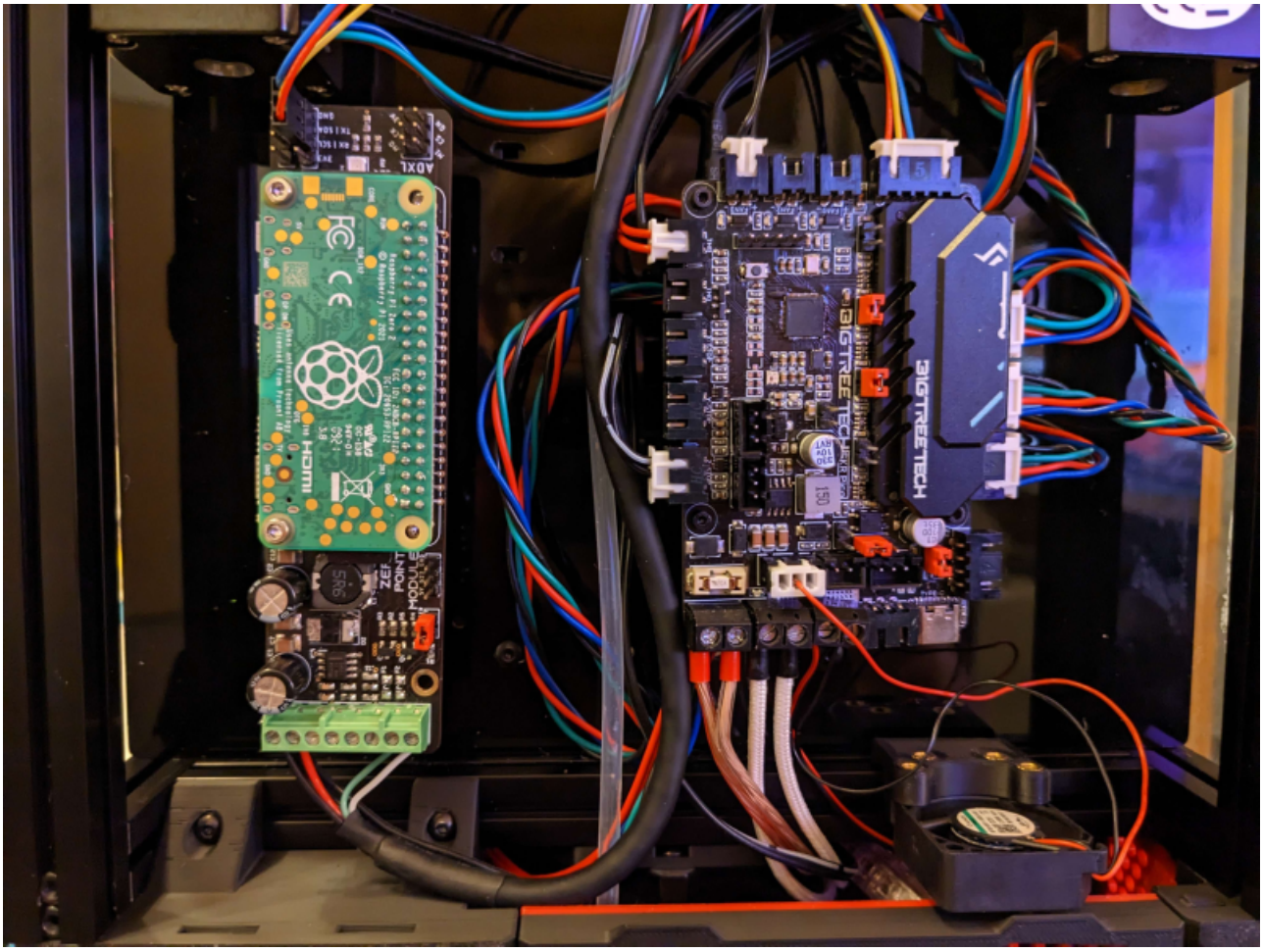
```
[mcu RPi]
serial : /tmp/klipper_host_mcu

[fan_generic RPiFan_0]
pin             : RPi:gpio12
max_power       : 1.0
cycle_time      : 0.0010
hardware_pwm    : false #true
kick_start_time : 0.100

[fan_generic RPiFan_1]
pin             : RPi:gpio13
max_power       : 1.0
cycle_time      : 0.0010
hardware_pwm    : false #true
kick_start_time : 0.100

[output_pin RPi_GPIO_4]
pin             : RPi:gpio4
```

```
[output_pin RPi_GPIO_5]
pin             : RPi:gpio5

[output_pin RPi_GPIO_6]
pin             : RPi:gpio6

[output_pin RPi_GPIO_16_LED]
pin             : RPi:gpio16

[output_pin RPi_GPIO_17]
pin             : RPi:gpio17

[output_pin RPi_GPIO_22]
pin             : RPi:gpio22

[output_pin RPi_GPIO_23]
pin             : RPi:gpio23

[output_pin RPi_GPIO_26_LED]
pin             : RPi:gpio26

[output_pin RPi_GPIO_27]
pin             : RPi:gpio27

[temperature_sensor BME_280]
sensor_type     : BME280
i2c_address     : 118
i2c_mcu         : RPi
i2c_bus         : i2c.1

[gcode_macro QUERY_BME280]
gcode:
    {% set sensor = printer["bme280 BME_280"] %}
    {action_respond_info(
        "Temperature: %.2f C\n"
        "Pressure: %.2f hPa\n"
        "Humidity: %.2f%%" % (
            sensor.temperature,
            sensor.pressure,
            sensor.humidity))}
```

Einbauvarianten

- Stefan Dej (meteyou) - Voron V0


BIGTREETECH
BIGTREETECH



Probleme

- sehr kleine Beschriftung auf dem Board
- SPI Port vom Zero nicht nutzbar (TBD)
- Schrauben übereinander → Fuckelig 🙂

Links

- ZeroPointModule auf Github
  https://github.com/pkElectronics/ZeroPointModule
- Schaltplan
  https://raw.githubusercontent.com/pkElectronics/ZeroPointModule/main/SRC/Schematic%20v20.pdf
- Pascal Wistinghausen - IB-Wistinghausen
  https://ib-wistinghausen.de/