ToqueCAN

GitHub: https://l.isiks.tech/ToqueCAN

ToqueCAN Features

- Pi toque, sits on a Raspberry Pi with GPIO passthrough
- USB C to Pi's front USB
- 3 port USB hub (+1 used for CAN)
- All USB connectors are ESD protected
- STM32F072 MCU with SN65HVD232 transceiver for 1M CAN
- 2x CAN outputs with jumpers for proper termination, expansion connector for more
- Supports Candlelight and Klipper (in bridge mode) firmware
- If Klipper firmware is used, optional 5V Neopixel control
- Powers the Pi and USB ports with its 5V 5A converter, can be turned off
- USB power limited to 2.1A so Pi always gets enough power
- 24V power input for CAN connectors and 5V regulator, fused
- Cutout for a 30mm fan for cooling the Pi, 5V or 24V
- Optional PWM RPM control for the fan, from Pi's GPIO or MCU if running Klipper in bridge mode

Parts Needed

These parts may or may not come with your PCB if you buy them from a vendor. Isik's Tech purchases will include these, except the fan, fan fasteners and the USB Cable.

| Part	US	AliExpress
ToqueCAN PCB	Check GitHub	
ToqueCAN USB Adapter (Optional)		
2x10 Pin Tall Long Legged Female Pin Header (Might be soldered on the ToqueCAN already)	Amazon	AliExpress
2x Molex Micro-Fit 2x2 Male Connector	Mouser	AliExpress
8x Molex Micro-Fit Female Pins	Mouser	
2x 3-Pin JST-XH Male Connector	Amazon	AliExpress
1x 4-Pin JST-XH Male Connector		
10x JST-XH Pins		
5x Pin Header Jumpers	Amazon	AliExpress
3x M2.5 14mm Male/Female Standoffs	-	AliExpress
4x M2.5 6mm (or similar) Screws	Amazon	AliExpress
4x M2.5x4.2x3.6 Threaded Inserts (Optional, for Pi mounts)	-	AliExpress
4x M3x5.0x4.0 Threaded Inserts (30mm Fan, 8x for 40mm Fans)	Amazon	AliExpress
4x M3 Screw (Fan) (Fan height + few mm long) (+4x M3 6mm screw for 40mm Fan)	Amazon	AliExpress
30mm (or 40mm) PWM Fan	Amazon	AliExpress
Short USB A to C Cable	Amazon	AliExpress

Note: AliExpress affiliate / Amazon associate links above. Purchases made using the above links help me continue developing open-source projects like this without additional cost to you. Thanks!

Step 1: Soldering Pins

Depending on how/where you sourced your ToqueCAN, the GPIO pins of your ToqueCAN may or may not be pre-soldered. If they aren't, solder the pins marked on the picture below. You can solder the rest of the pins if you prefer, but they aren't used by the ToqueCAN and some of them can be difficult to reach with your soldering iron.

Long legs on the male side of the GPIO pins are easy to accidentally bridge and cause a short, so check your solder joints before powering your ToqueCAN.
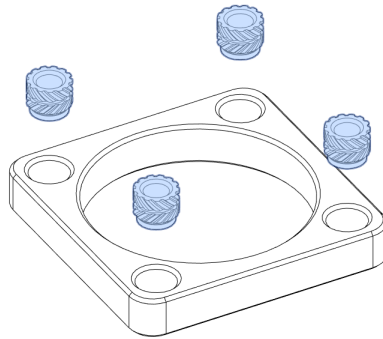
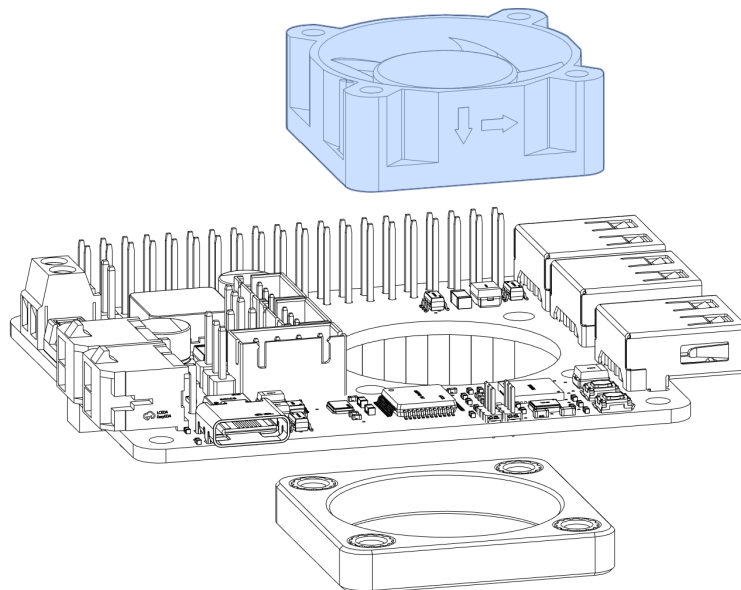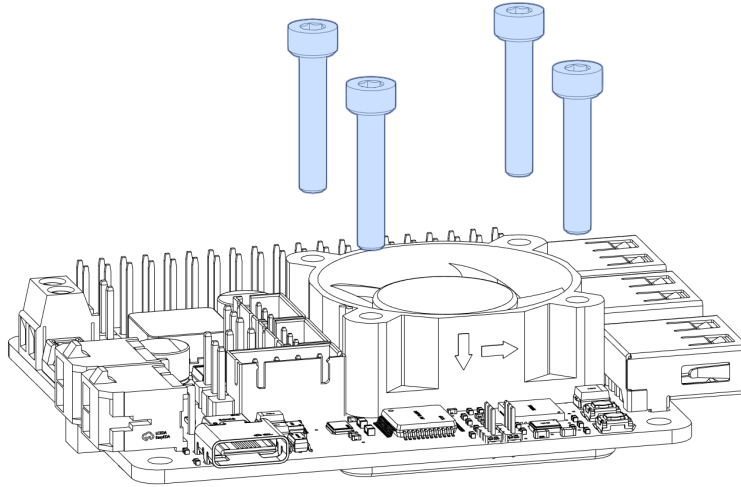Solder the pins marked below:
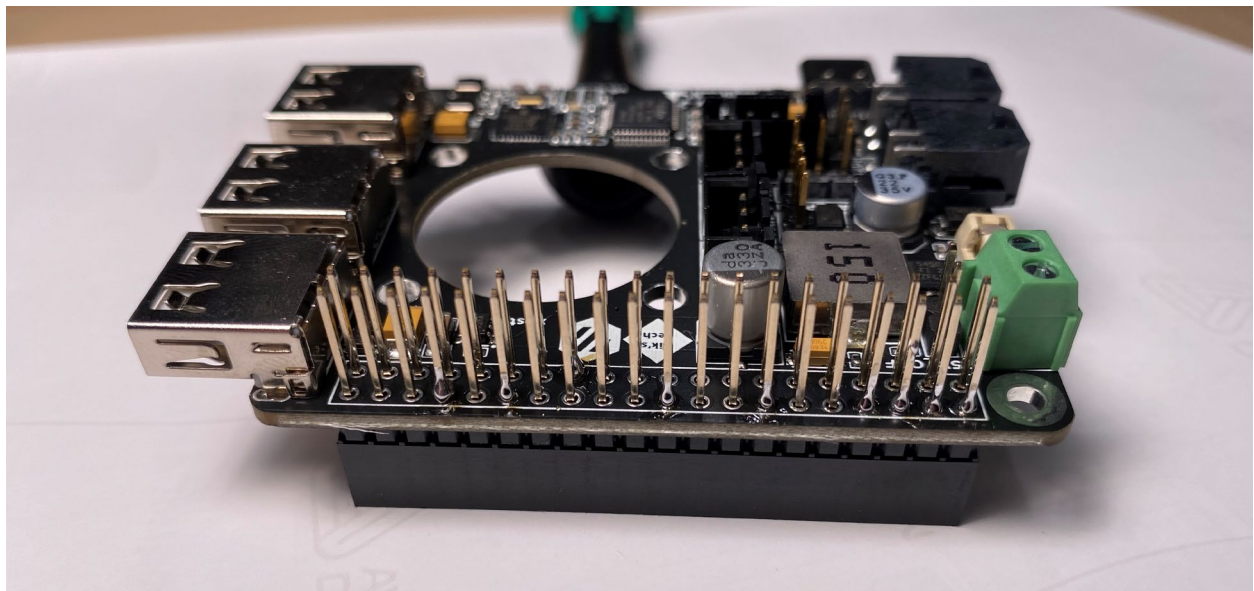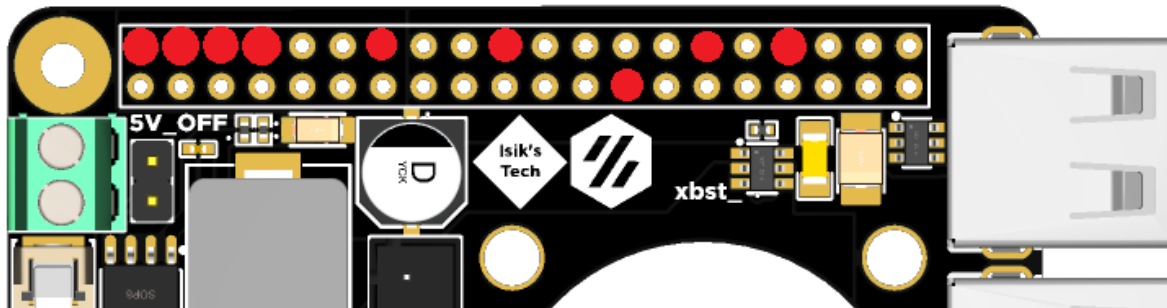
Step 2: Fan Installation

30mm

1. Print the 3D printed part from the GitHub repo.
2. Insert 4x M3x5x4 (Voron size) threaded inserts.

3. Align the printed part behind the PCB, place the 30 mm fan above the PCB.
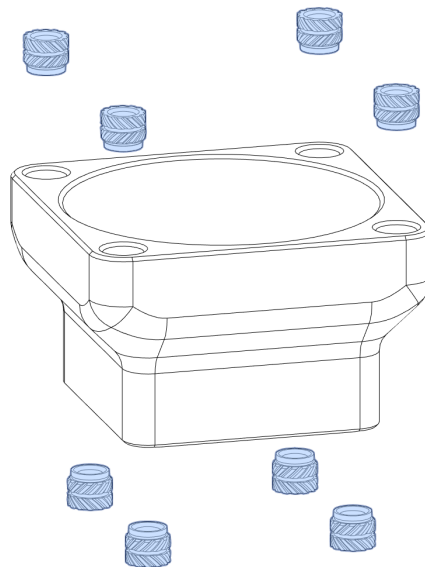
4. Use the appropriate size M3 screws (usually 12 mm for 3007 fans, 16mm for 3010 fans) to screw the fan to the printed part with the PCB in between.

40mm

1. Print the 3D printed part from the GitHub repo.
2. Insert 8x M3x5x4 (Voron size) threaded inserts.

ToqueCAN Manual

3. Align the printed part on the PCB, screw 4x M3 6mm screws with plastic washers from the other side of the PCB to secure it.

You can use nylon washers if you have them. If not, printable washers are available on the GitHub repo.

4. Place the 40 mm fan above the printed part. Use the appropriate size M3 screws (usually 16 mm for 4010 fans, 25mm for 4020 fans) to screw the fan to the printed part.
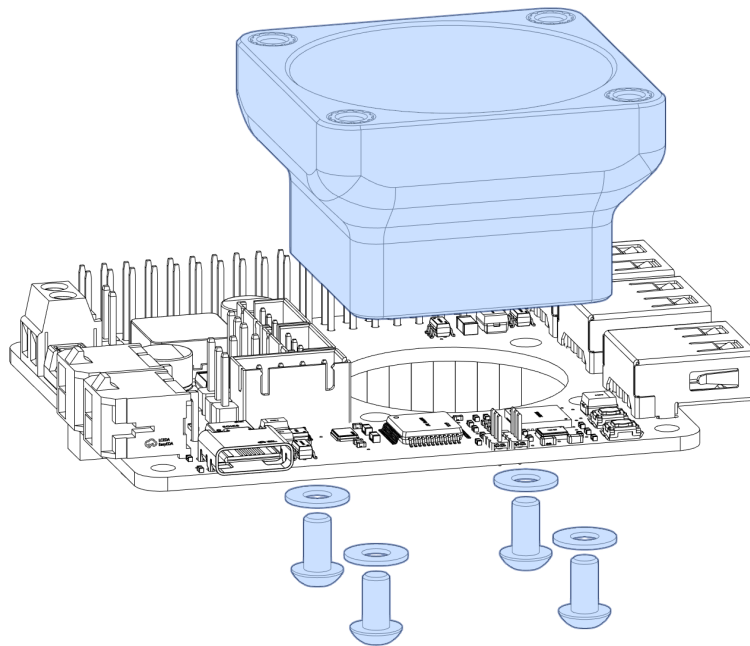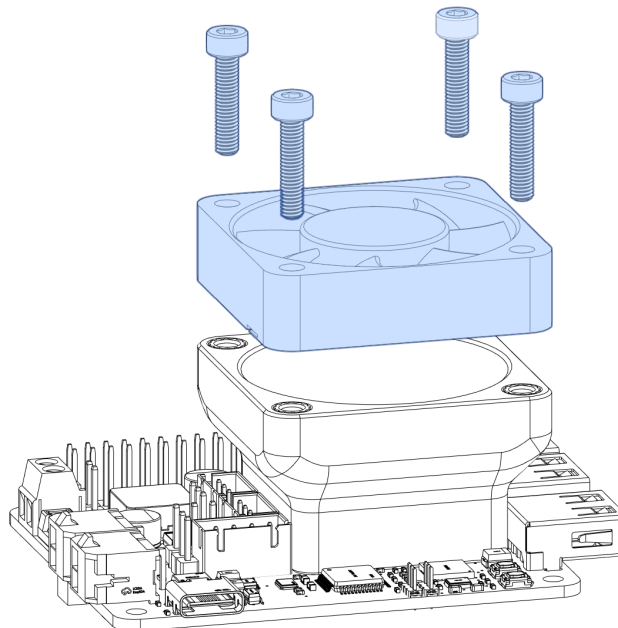
Step 3: ToqueCAN on Pi Installation

1. Insert the M2.5x4.2x3.6 inserts to your mount if your mount supports it.
2. Screw a M2.5 6mm screw in the corner with GPIO and USB/Ethernet (depending on your Pi's version) on your Raspberry Pi.
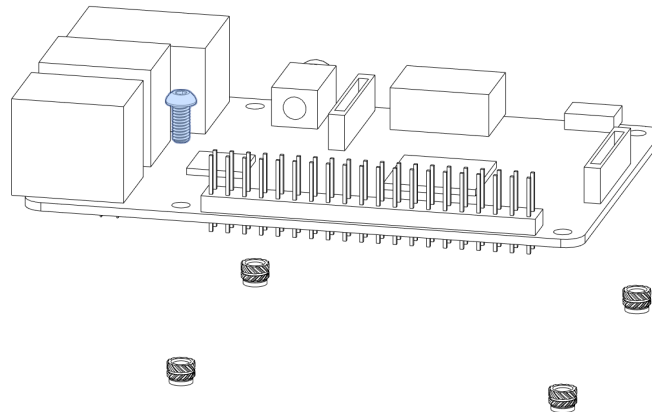
3. Screw 3x M2.5 14mm standoffs in the other screw holes.

ToqueCAN Manual
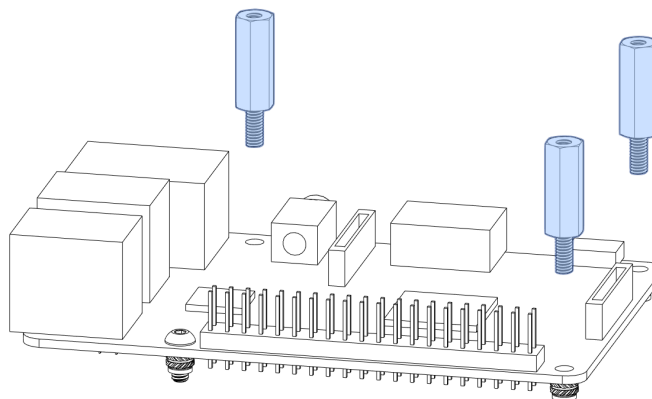
4. If you're using the 180° USB Adapter, connect the USB cable to the Pi, route it between the USB / Ethernet connectors to where the USB connector of the ToqueCAN will be.
5. Mount the ToqueCAN on the GPIO pins of your Pi.
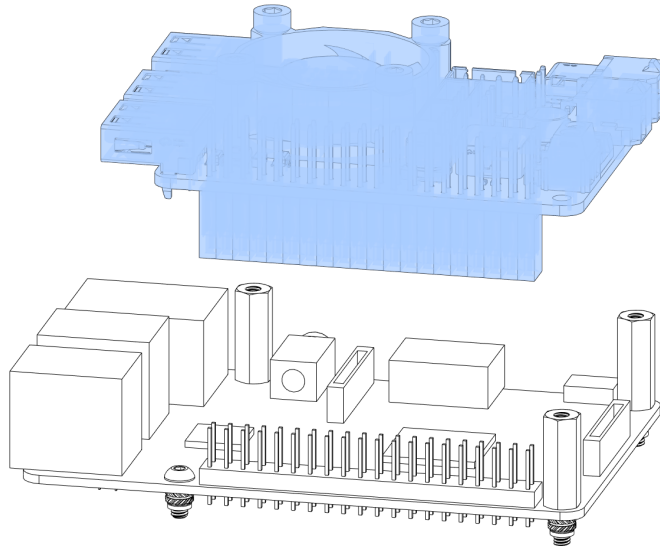
6. Screw 3x M2.5 6mm screws in the corner holes. Connect the USB cable.

ToqueCAN Manual

Step 4: Jumpers

There are 5 sets of jumpers on the ToqueCAN. 3 more commonly adjusted ones are highlighted in the image below:
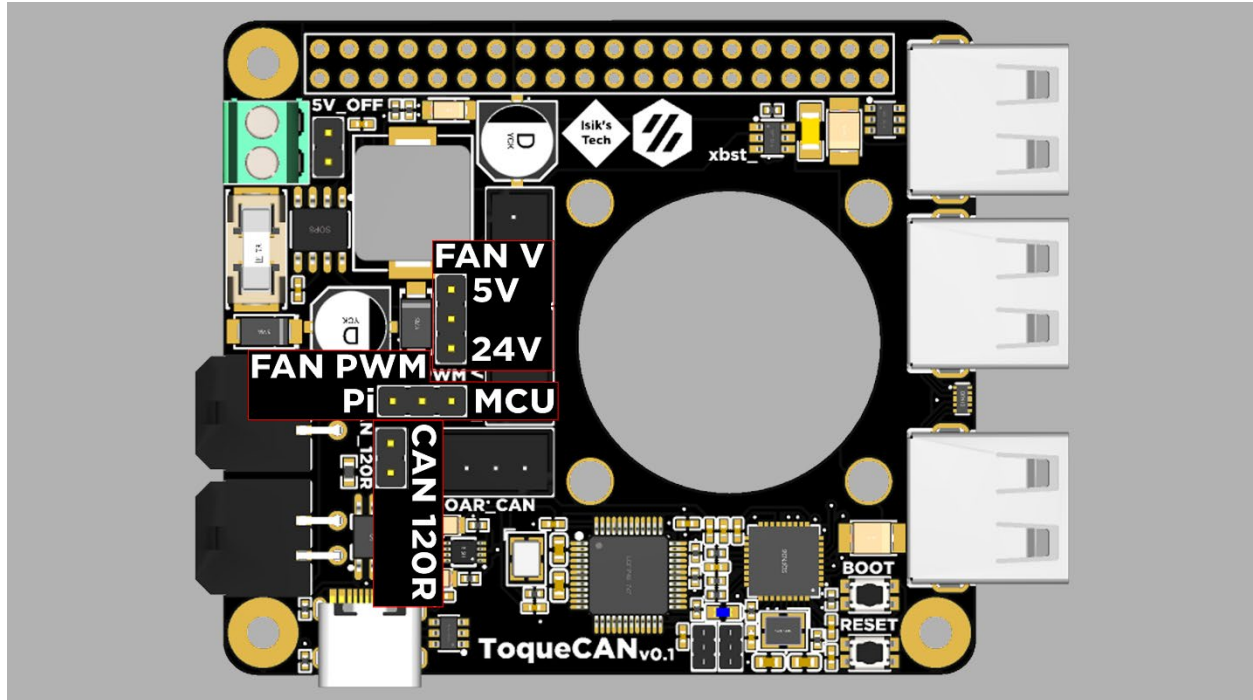
Jumpers:

- 5V_OFF: Turns off the 5V regulator on the PCB. Leave unpopulated for ToqueCAN to generate 5V for the Raspberry Pi, USB devices and other devices powered by the 5V rail.
- MOAR_CAN: This is a jumper header and an expansion connector in one. Its usage is explained later in this document. Populate with 2x jumpers for using the Microfit CAN outputs only.
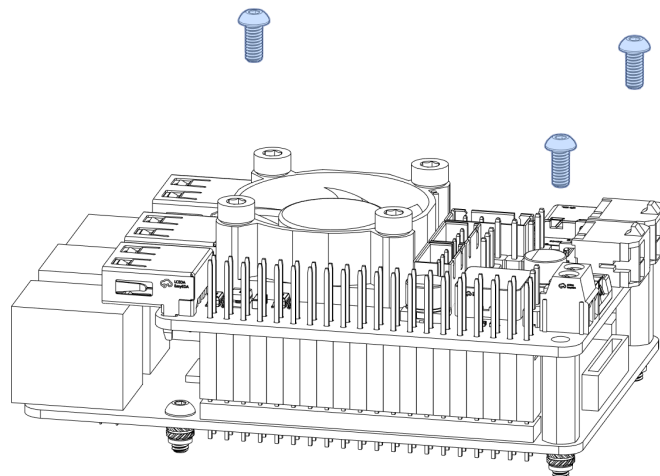- CAN_120R: The 120Ω termination resistor for CAN bus. Populate it if you are connecting 1 CAN device to the Microfits, don't populate if you're using both. For its use with MOAR_CAN, refer to that section of this document.
- FAN_PWM: The PWM source for the 30/40mm fan, if you are using a PWM capable fan. This lets you control the fan's speed. You need to use Pi's control if you are using Candlelight as your CAN firmware, with Klipper in bridge mode you can use the Pi or the MCU.
- FAN_V: Voltage of the 30/40mm fan. 5V from the regulator or 24V from the power input.

Step 5: Connecting the FAN & (Optionally) Neopixels
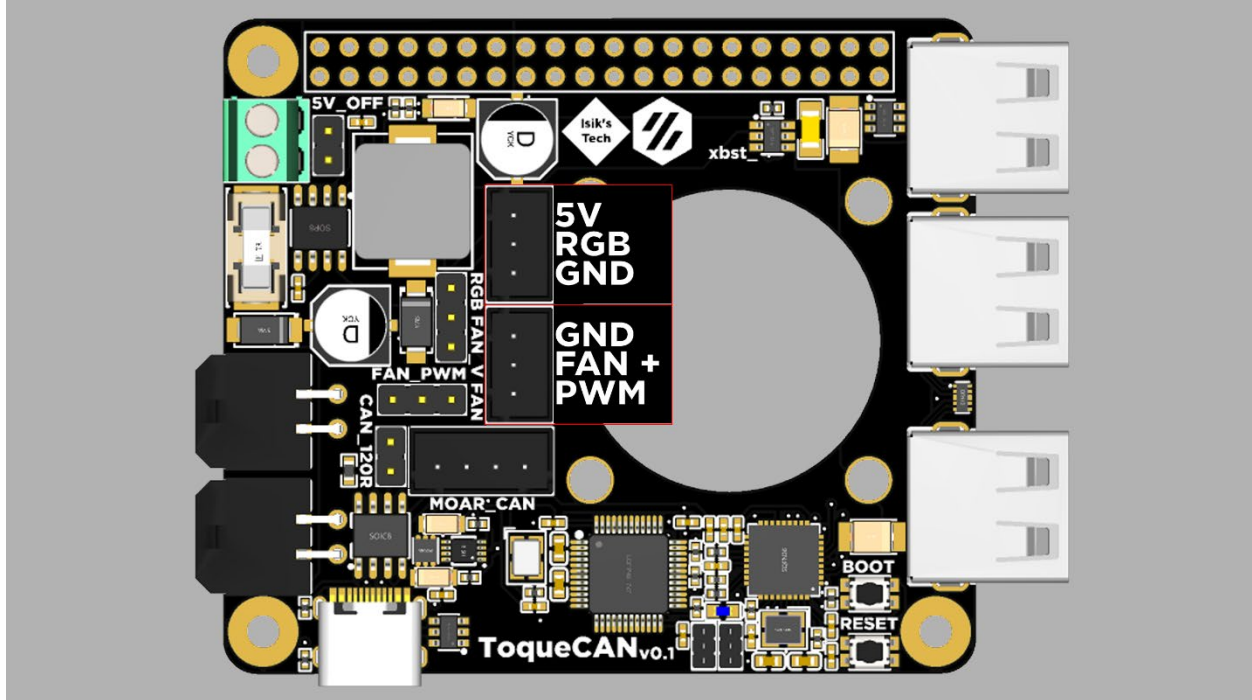
Neopixels are only compatible with Klipper (in bridge more) firmware, you can't use them with Candlelight.

Step 6: Powering ToqueCAN

ToqueCAN is designed for 24V power. There is a 10A fuse on the PCB to protect the ToqueCAN and attached devices. The maximum TOTAL power draw available through the CAN connectors is 8A, the rest is used for the 5V regulator and some safety margin.

Power the ToqueCAN using the screw terminal. Ferrules are recommended.

Pinout:
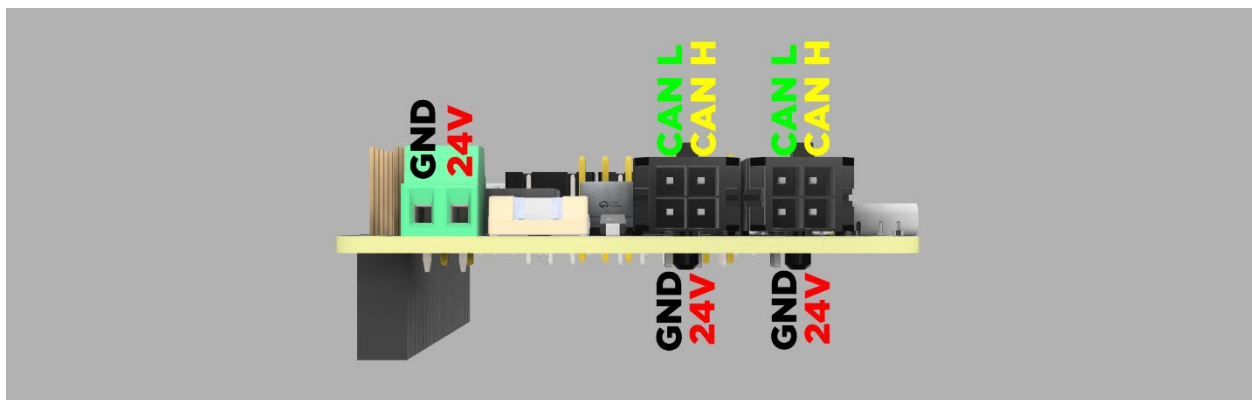
Step 7: Firmware Flashing

Candlelight

1. SSH into your Raspberry Pi.
2. Install dependencies:
```
sudo apt-get install cmake gcc-arm-none-eabi git dfu-util
```
3. Download the firmware: (By Lab4450)
```
cd ~
git clone https://github.com/lab4450/u2c
cd u2c
```
4. On your ToqueCAN, hold down the BOOT button, while holding it down press the RESET button, then release the BOOT button. The blue LED should turn on.
5. Use `lsusb` to verify that the ToqueCAN's MCU (STM32F072) is in DFU mode.
6. Flash the firmware:
```
sudo dfu-util -D candleLight_fw.bin -d 0483:df11 -a 0 -s
0x08000000:leave
```

Klipper (Bridge Mode)

1. SSH into your Raspberry Pi.
2. Navigate to the Klipper directory: `cd ~/klipper`
3. Clean old build data: `make clean`
4. Configure the build settings: `make menuconfig` Use these settings:

```
[*] 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 to CAN bus bridge (USB on PA11/PA12))  --->
    CAN bus interface (CAN bus (on PB8/PB9))  --->
    USB ids  --->
(1000000) CAN bus speed
()  GPIO pins to set at micro-controller startup (NEW)
```

5. Press `Q` to exit, `Y` to save.
6. Plug in your ToqueCAN while holding down the BOOT button.
7. Use `lsusb` to verify that the ToqueCAN's MCU (STM32F072) is in DFU mode.
8. Flash the firmware:
```
make flash FLASH_DEVICE=0483:df11
```

Step 8: Klipper Can Bus Prep & CAN PCBs

1. If this is your first-time enabling CAN Bus on this printer, you need to enable the CAN network on your host device (Pi). To do this, you need to create a file. Use this command: (SSH)
   ```
   sudo nano /etc/network/interfaces.d/can0
   ```

   Add this to the file you're creating:

   ```
   allow-hotplug can0
   iface can0 can static
        bitrate 1000000
        up ifconfig $IFACE txqueuelen 1024
   ```
   Press `CTRL + X`, then `Y` and `ENTER` to save & exit.
2. Shutdown your host using: `sudo shutdown now`
3. When the LEDs on the Pi stop blinking, turn off your power supply to disconnect your ToqueCAN from power.
4. Crimp the microfits on the wires of your CAN devices. Insert them to the microfit connector according to this pinout:
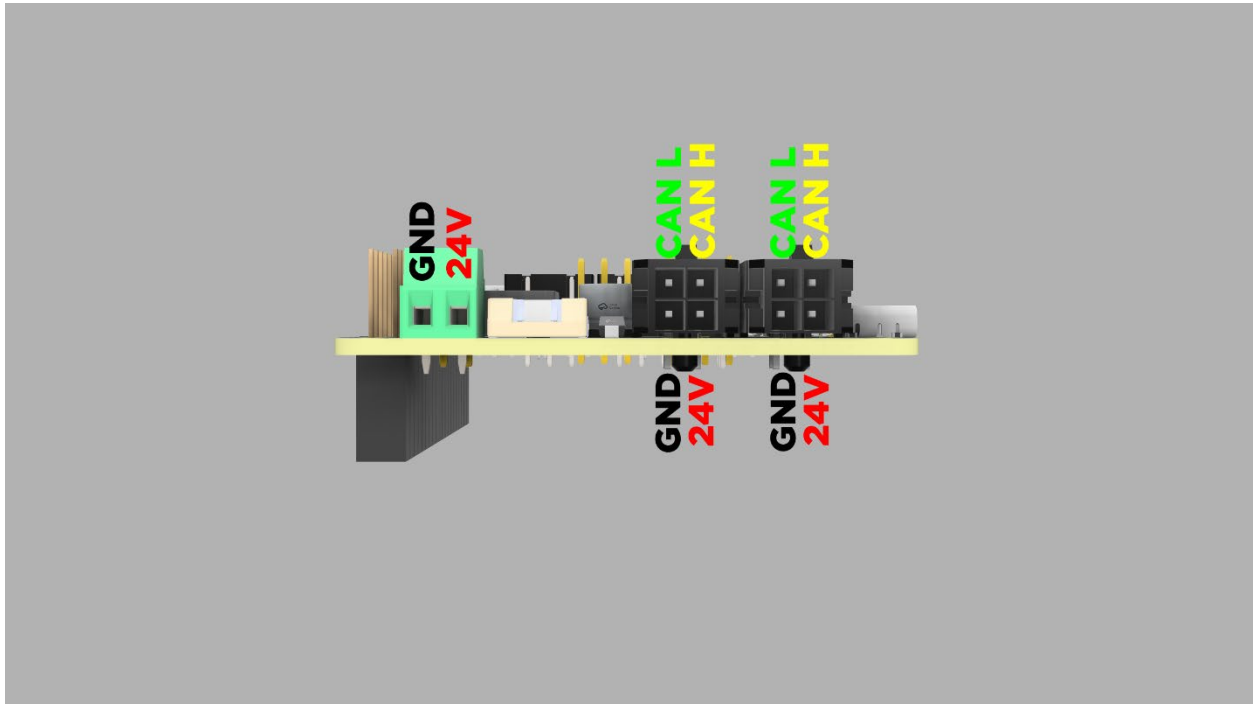


5. Plug in your CAN devices.
6. Turn your power supply back on and SSH into your Pi.
7. If your CAN devices aren't flashed with CanBoot/Katapult and/or Klipper yet, use their instructions to flash them.
8. Use this Klipper script to check if your CAN devices are detected and what their UUIDs are:
   ```
   ~/klippy-env/bin/python ~/klipper/scripts/canbus_query.py can0
   ```

ToqueCAN Manual

## Step 9A: Pi PWM Fan Control

This step is optional. If you want to use your Raspberry Pi to control your PWM fan, and configured the jumpers correctly for this, follow these instructions:

1. SSH into your Raspberry Pi.
2. `sudo raspi-config`
3. Enter `Performance Options`
4. Enter `Fan`
5. Select `Yes`
6. Enter `14`
7. Adjust the trigger temperature.
8. When done, exit with `Finish`, and reboot.

## Step 9B: Klipper Bridge Mode Config

This step is optional. You only need to follow this step if you are using Klipper in bridge mode as your ToqueCAN's firmware and want to use Klipper to control the PWM fan and Neopixels.

1. Download the config file from the ToqueCAN [GitHub repo](#).
2. Upload this to your config folder
3. Open it and edit your `canbus_uuid`. You can use the script from the last step of Step 8 to find what your UUID is. Save and close (don't restart yet).
4. Open your `printer.cfg`. Add this line:
   `[include toquecan.cfg]`
   Save and close. Restart Klipper.

🍪

# MOAR_CAN Connector

This is an optional connector for adding more CAN bus devices. This requires an external PCB or similar solution to wire all the CAN bus devices together. Currently there isn't a ToqueCAN solution for this. This section explains how it's intended to be used.

Here is a diagram of how the CAN connectors, termination resistor, transceiver and the MOAR_CAN connector are wired on the PCB:
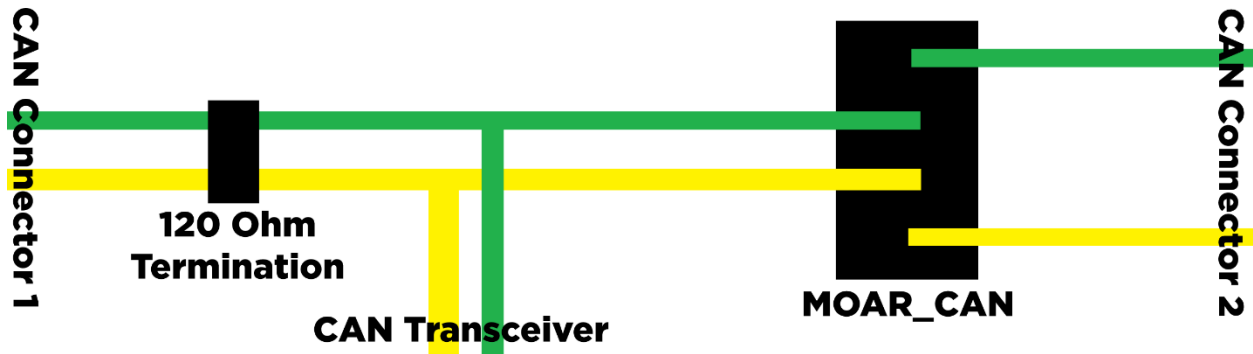


As you can see, the MOAR_CAN connector is between the CAN connectors. Populating it with 2 jumpers joins the CAN wires of both connectors so they're on the same bus. If using both connectors, the termination resistor header isn't jumped, since the transceiver of the ToqueCAN is in the middle so the CAN bus doesn't need termination. If using only one CAN device, the termination resistor header should be jumped as now the ToqueCAN is at one end of the CAN bus.

But what if you want moar CAN bus devices? That's what this connector is for. You simply need to wire it in a way that goes to your extra CAN devices and comes back.

For example:
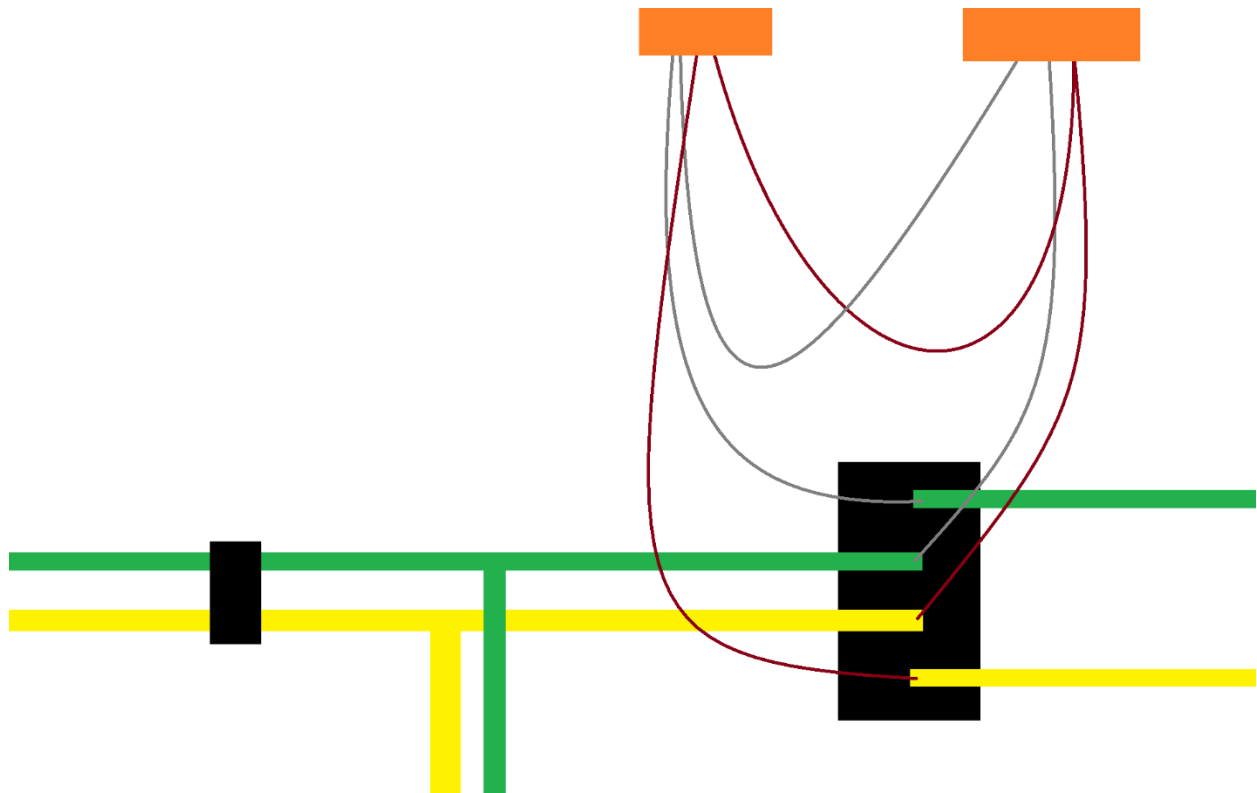


In this diagram 2 more CAN devices are connected to the same bus. Wires are connected to the middle pins of the MOAR_CAN connector, a CAN bus device to these wires. Then daisy chained from that device to another device. Finally, it's daisy chained again to the outer pins of the MOAR_CAN connector on the ToqueCAN PCB. In this example you connected 4 devices to the CAN bus of the ToqueCAN, so in total there are 5 devices on the bus (including ToqueCAN). Theoretically you can add even more devices, but you will eventually hit a limit where your devices can't communicate well.

A simple breakout PCB with some 4-pin JST-XH connectors can act as a hub for this to make daisy chaining easier.

Also remember to make sure the first and last CAN devices on the bus should have a 120Ω termination resistor, and none of the devices in between should have that.

Another thing to keep in mind is, your other CAN devices should share the same GND, so you either need to power them from the same source or tie the GNDs together.

# Chips & Specs

| ToqueCAN | |
|---|---|
| Power In | 24V |
|     Fuse | 10A ([Replacement](#)) |
|     Powers | CAN Outs<br>5V Regulator<br>Optionally (Jumper), Fan |
| 5V Regulator | TI TPS5450 |
|     Power Out | 5V 5A |
|     Powers | Raspberry Pi (Through GPIO)<br>3.3V Regulator<br>USBs (Limited to 2.1A, D.I. AP22653)<br>Optionally (Jumper), Fan<br>Neopixels |
| 3.3V Regulator | TI LP5912-3.3 |
|     Power Out | 3.3V 500mA |
|     Powers | USB Hub<br>CAN MCU<br>CAN Transceiver<br>TI SN74LVC2G125 |
| USB Hub | Microchip USB2514B-I/M2 |
|     USB In From | USB C |
|     USB 1-3 To | USB A Connectors |
|     USB 4 To | CAN MCU |
| CAN MCU | STM32F072C8T6 |
|     Firmware | Candlelight or Klipper in Bridge Mode |
|     Connects To | CAN Transceiver (TX: PB9 RX: PB8)<br>USB Hub (PA11/PA12)<br>Neopixel (PA3, Through SN74LVC)<br>Fan PWM (Jumper, PA2, Through SN74LVC) |
| CAN Transceiver | TI SN65HVD232 |
|     Speed | 1M |
| PCB Size | 56.5 x 65.0 mm |
| PCB Layer Count | 4 |

# Debugging
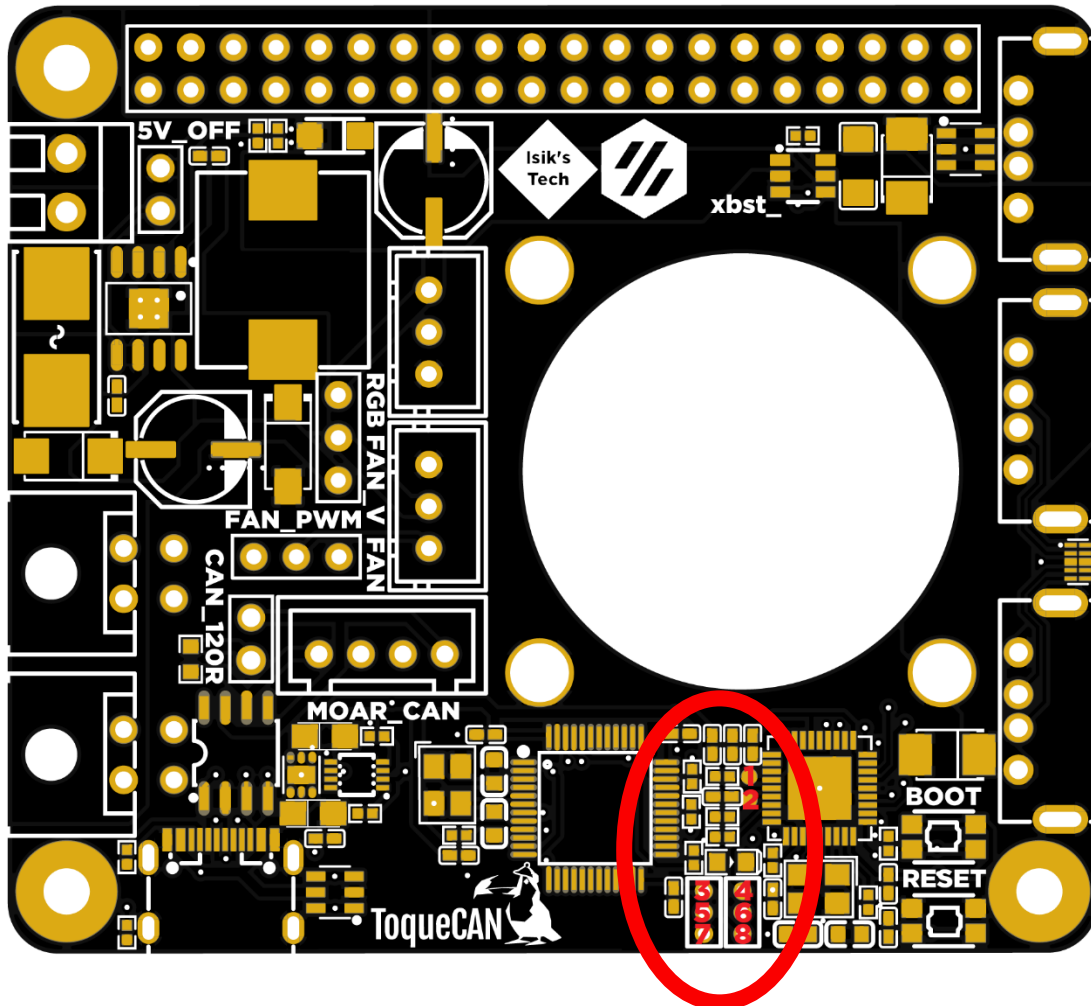


| Pin | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|
| Chip | USB2514 | USB2514 | - | USB2514 | STM32 | STM32 | - | STM32 |
| Function | SDA | SCL | 3.3V | Reset | Reset | SWDIO | GND | SWCLK |

Pins 3,5,6,7,8 can be used to flash new firmware to bricked (DFU not working) STM32F072 MCUs, or to access other ST-Link features. You can use the buttons for BOOT0 and RESET.

Pins 1,2,3,4,7 can be used to flash new firmware to the USB hub controller (USB2514B-I/M2) via $I^2C$, or to reset it.

# Schematic
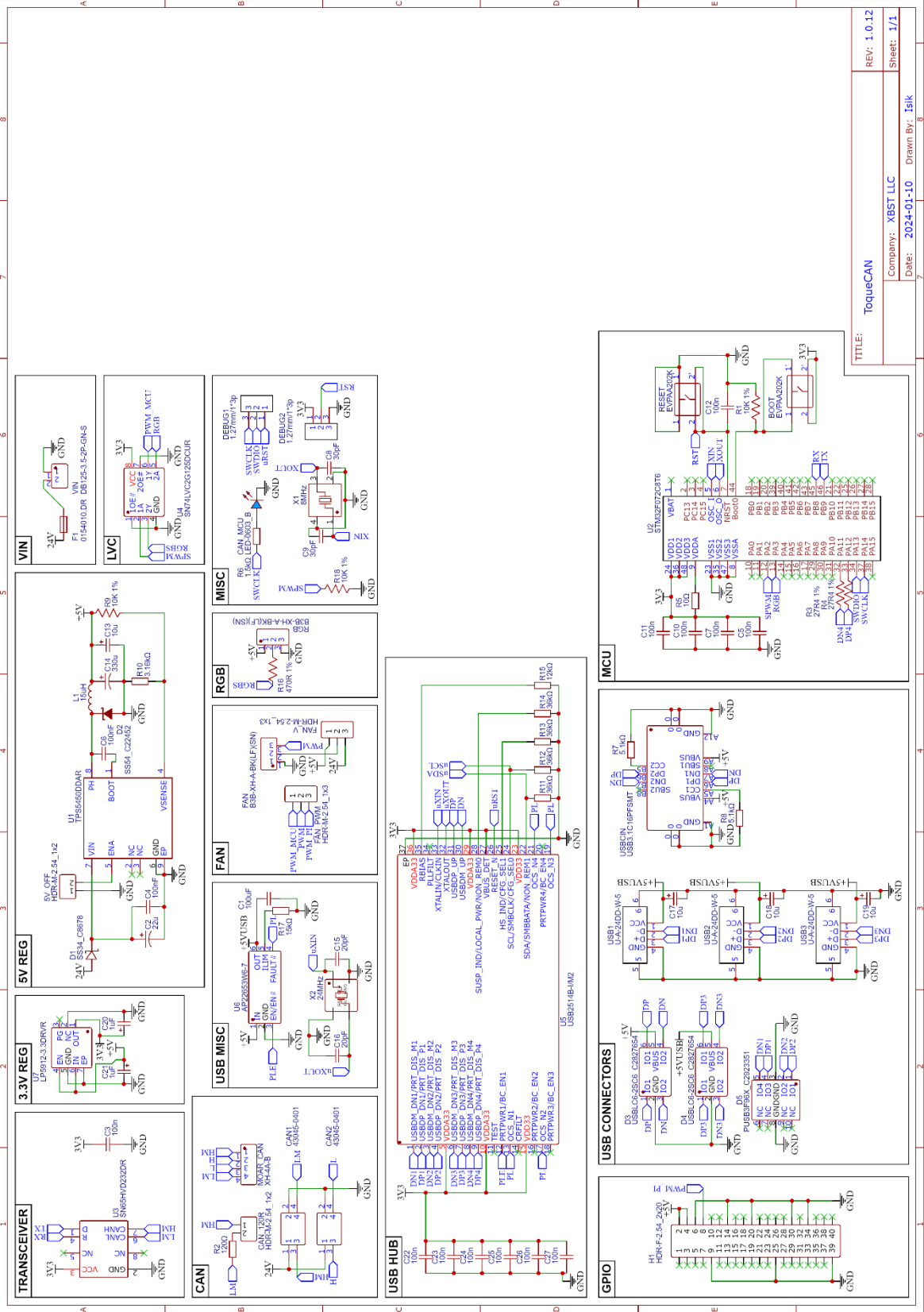
## More Resources

- Klipper CAN Bus Docs: https://www.klipper3d.org/CANBUS.html
- Klipper CAN Bus Troubleshooting: https://www.klipper3d.org/CANBUS_Troubleshooting.html
- Lab4450 U2C Candlelight Firmware: https://github.com/lab4450/u2c
- GitHub Repository of the Project: https://l.isiks.tech/ToqueCAN

## Thanks

- Doc
- Le0n (logo)
- Thebrakshow
- Esoterical
- alexz
- Kyleisah
- whoppingpochard

**Isik – xbst_**