Software Setup

Please follow the software setup instructions in our GitHub to update the drivers on the Pico properly, install PlatformIO, and run the code on the Pico. Be sure to **update the drivers** on the Raspberry Pico. Without this step, the project code will not work.

Hardware Assembly

Parts list for the BLDC Motor Driver Project:

- 1x AS5047 Magnetic Absolute Encoder
- 1x GM5208 Brushless Gimble Motor
- 1x Raspberry Pico
- 1x Brushless Bros Inc. PCB
- 1x 8-pin Wire Harness
- 1x 3-pin Wire Harness
- 1x power supply
- 1x 3D Printed Stator-Encoder Adapter
- 1x 3D Printed Rotor Magnet Adapter
- 1x 3D Printed "Tuning Fork"
- 1x 3D Printed PCB mount
- 8x 3M Bolts

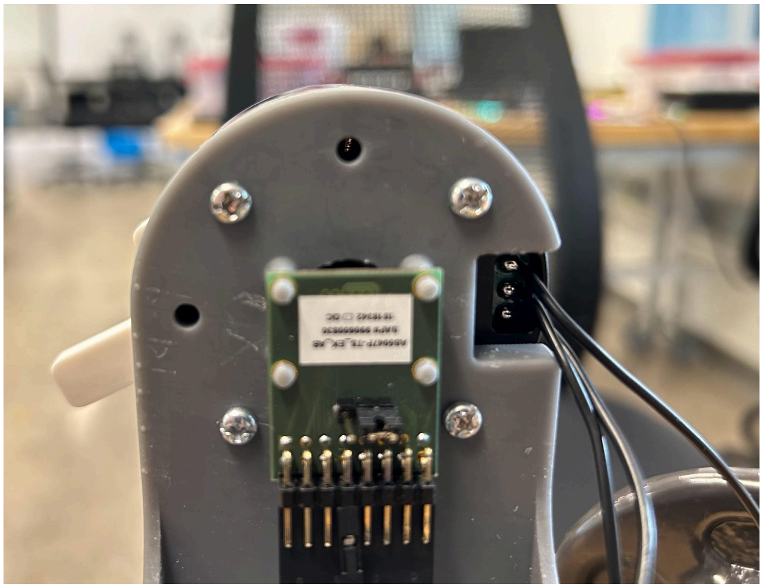
Tools:

- Phillips Head Screwdriver
- Clamps (To clamp the motor mount to a table)

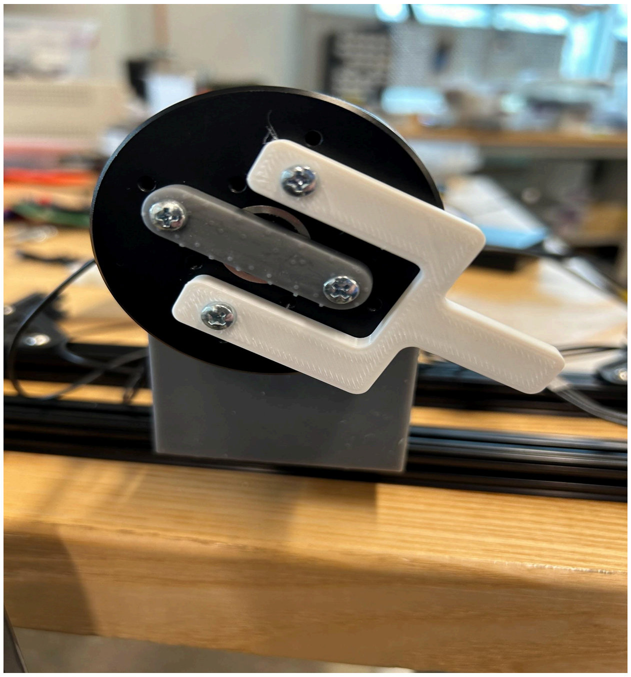
Instructions:

- 3D Print all necessary parts
    - All CAD files can be found on our Github here
    - All parts except the PCB mount and tuning fork were printed on a Form 2 Formlabs resin printer for dimensional accuracy
    - The PCB mount and tuning fork were printed on an FDM Creality Ender 3

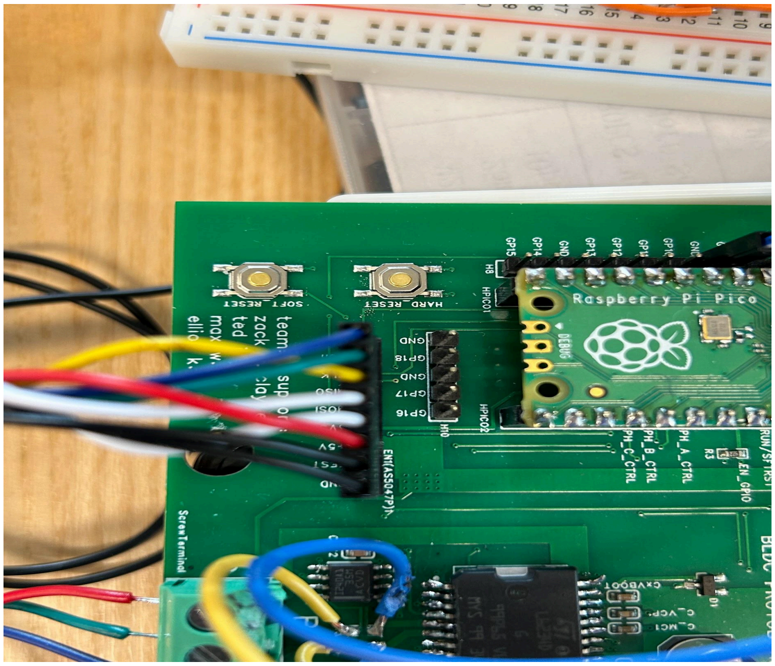
- Mount the Stator-Encoder Adapter to the GM5208 Motor's stator using 4 10mm 3M bolts



- Place the magnet disk for the encoder in the Rotor Magnet Adapter (press fit)
- Mount the Rotor Magnet Adapter to the GM5208 Motor’s rotor using 2 3M bolts
- Mount the Tuning Fork to the GM5208 Motor’s rotor using 2 3M bolts

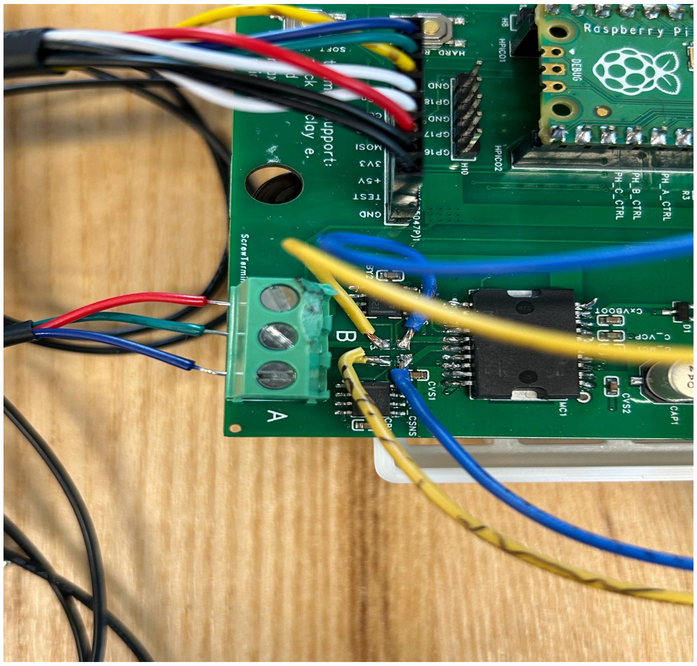


- Place the AS5047 Encoder chip on the Stator-Encoder Adapter by sticking the placement stubs through the mounting holes on the encoder chip
- Place the PCB board in the PCB mount
- Connect one side of the 8-pin wire harness to the bottom row of pins on the encoder chip (should have CSn, CLK, MISO, and MOSI pins along with power/ground pins)
- Connect the opposite side of the 8-pin wire harness to the PCB 8-pin female header pin connector



- Place the Raspberry Pico in the 20 female header pin connectors on the PCB
    - Note about the PCB: our photos include the use of a breadboard wired to the PCB to replace resistors that had incorrect values during PCB production. If the shunt resistors R4 and R5 are 5 megaohms, this change will need to be made. If not, this change is not necessary. For the sake of brevity, we have left this change out of this document as it should not be necessary
- Connect the connector side of the 3-pin wire harness to the GM5208 wires

- Connect the opposite side of the 3-pin wire harness to the PCB by securing it in the 3-pin screw terminal



- Plug in the Power Supply (USB Battery) the barrel jack connector
- Connect micro USB cable to the Raspberry Pico for programming

Quick Startup

Once you’ve completed the software and hardware setup, pull the code from our GitHub to your local machine and set it up as a PlatformIO project. Follow the instructions in the software setup section to build the code and flash it onto the Raspberry Pico. The library of code will give you the ability to do velocity control or position control with this setup. In velocity control mode, you can set a speed setpoint and have the motor spin at the desired speed in (rad/s). In position control mode, you can set positional setpoints (in radians) and have the motor turn to those setpoints.