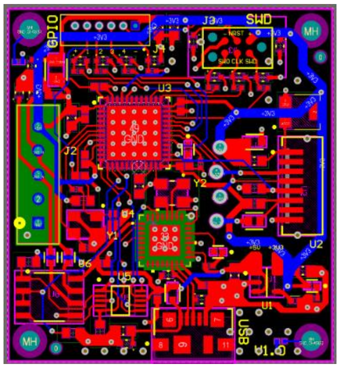
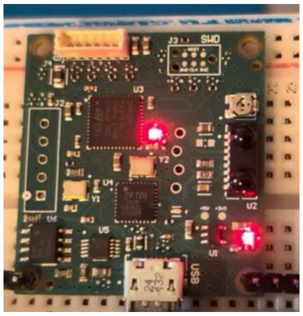
CHANUTH WEERARATNA - PORTFOLIO

✉ chanuth.weeraratna@uwaterloo.ca ☎ 647-773-0239 chanuth11 www.linkedin.com/in/chanuth

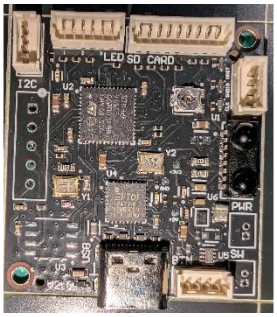
Product Development – Spectra | C, C++, Python, Altium, KiCAD 2025



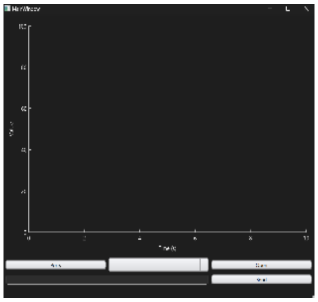
Altium Docking Station



Blink Sketch of IR/RS-485 Data



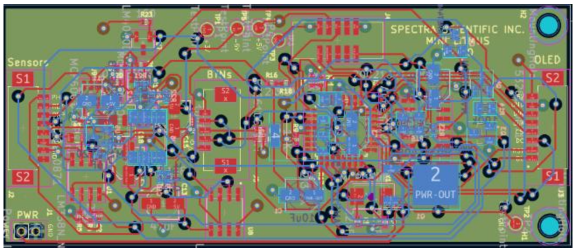
V2 Board Design



Real Time Data GUI

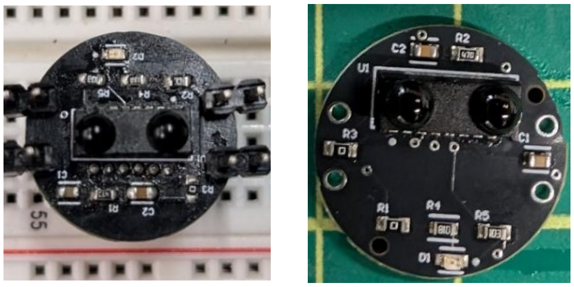
Board bring-up

Production Assembly



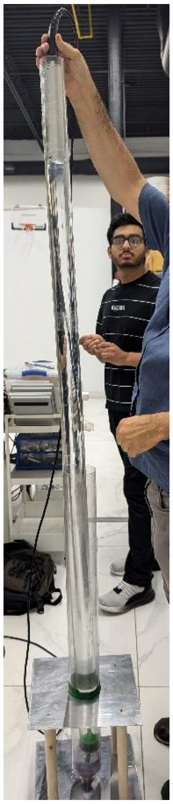
KiCAD Multi-sensor PCB

RS-485 Module w/ pogo pins



V1 and V2 of IR Module

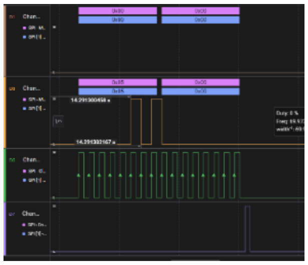
Assembled Prototype



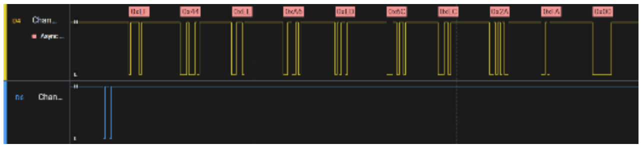
Datalogger Pressure Test Setup

- ❖ Executed full product lifecycle of a handheld multi-sensor meter, including prototyping, component selection, schematic capture, PCB layout/design, firmware test scripts, and field testing
- ❖ Created a custom 4-layer STM32 board in Altium to establish bidirectional UART, RS-485, and IR communication, incorporating copper puddles, stitching vias, ESD diodes, and differential pairs
- ❖ Designed IR and RS-485 PCBs in Altium with attention to DFM, water proofing, and size constraint
- ❖ Created test scripts in C to verify IR and RS-485 packets using cyclic redundancy checks (CRC)
- ❖ Created sensor calibration scripts in C++ using two-point calibration, regression, and Steinhart equation for NTC temperature sensors
- ❖ Created a front-end display in QT creator using Python and C++ to send firmware commands and receive and plot live data from pressure, temp, humidity sensors
- ❖ Designed and brought up a 4-layer ATMega328P board in KiCAD using an 12-bit MCP3208 ADC, GC4900A LCD, sensors (ORP, PH, Temp, Conductivity), and auto shutoff circuitry
- ❖ Integrated low power by using buck convertors for level shifting and trimmer POT for low IR mode Include the power circuitry layout, tight layout, jargon added in the muon application

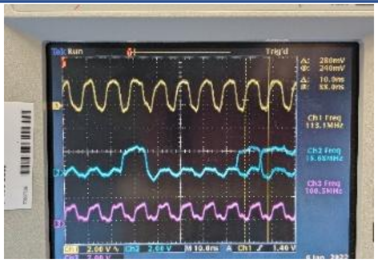
Production Test – KA Imaging | OrCAD, Python, C 2025



SPI Digital POT commands



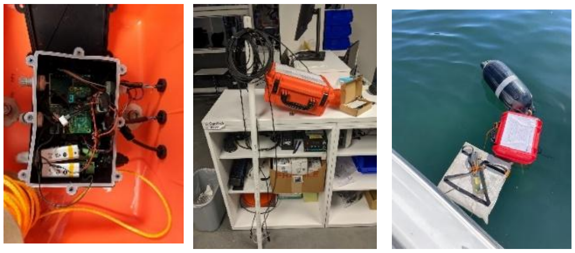
I2C Temp/Humid data



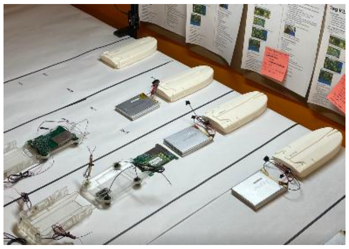
Rectified 100Mhz Clock Line (blue)

- ❖ Rectified 100Mhz clock line to increase X-ray image capture throughput using AC and series termination resistors to limit signal reflections at high speeds
- ❖ Decoded Motherboard I2C temp/humid sensor and dig pot High Voltage board (+/- 2000V) SPI settings using Logic analyzer

Test Jigs and PCB Designs – Project CETI | Fusion360, EasyEDA 2023

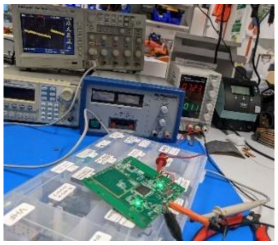


Directionality/Noise Test Fixture

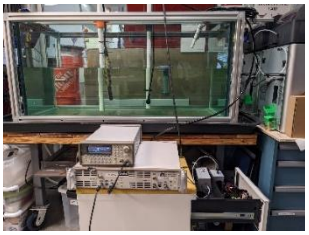


Full Tag Production

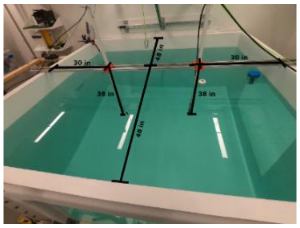
Potted Hydrophones



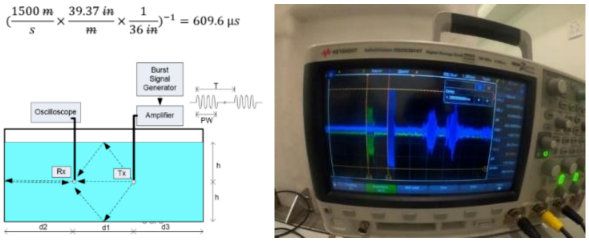
Tag Bring-up



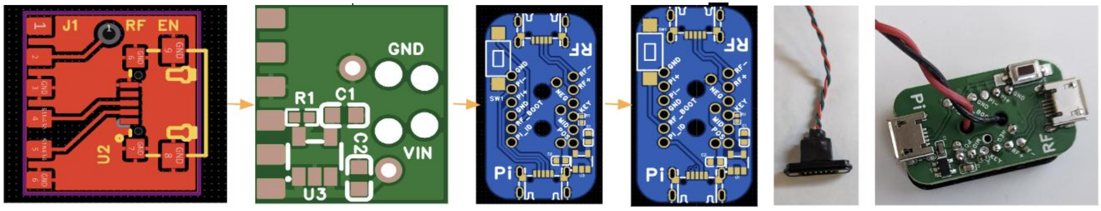
Small Water Tank Setup



Large Water Tank Setup

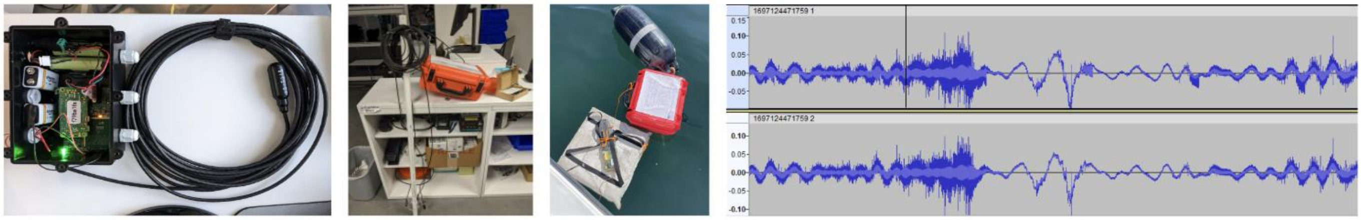

$$\left(\frac{1500\ m}{s} \times \frac{39.37\ in}{m} \times \frac{1}{36\ in}\right)^{-1} = 609.6\ \mu s$$

Time of Flight Delay Verification



Magnetic Connector PCB Progression

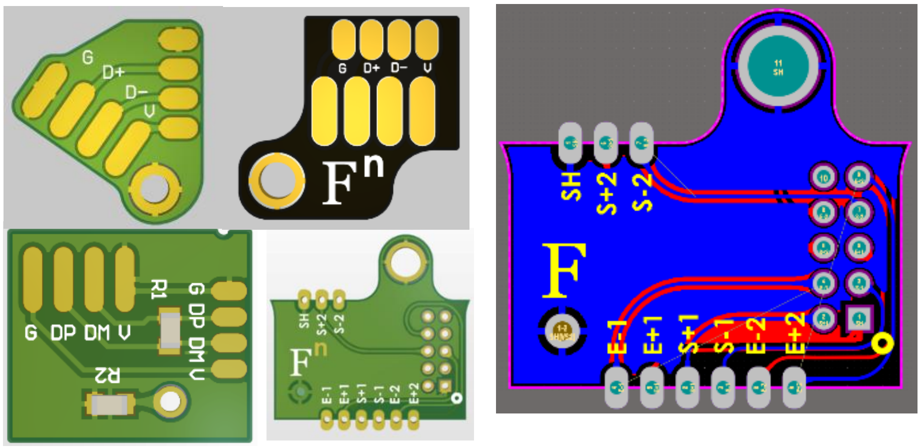
CMOY Amplifier



Boat Day Hydrophone Test and Ambient Noise Test (Audacity Audio waveform on the right)

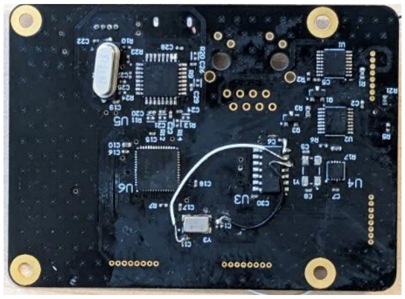
- Created a directionality and ambient noise testing jig using an acrylic laser-cut hydrophone stand, recording Tag, and waterproof casing to gather time of flight delay and ocean noise data
- Designed an underwater audio calibration test using adjustable pole mounts in Fusion360, analyzing the sensitivity and time of flight delay on oscilloscope and Audacity waveforms
- Created a waterproof CMOY amp using audio gain POTs and TLE2426 for virtual ground in EasyEDA
- Adjusted the Tag's LPF and HPF cutoff frequencies, audio gains, and ADC dynamic range
- Optimized Tag integration throughput by creating Bash and C testing scripts for RPi and STM32

PCB Design and Board Population - Forcen| Altium 2023

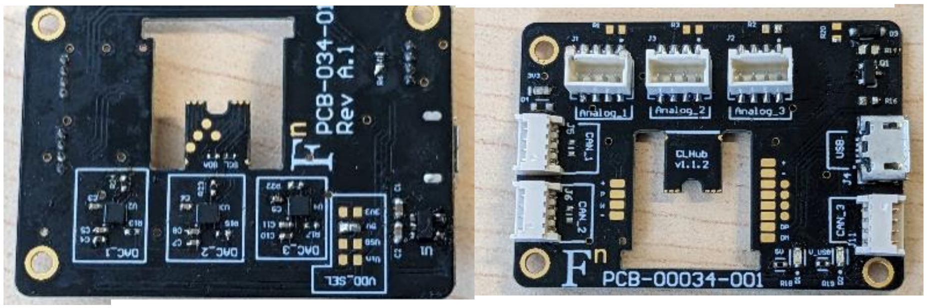


Digital and Analog COM Boards

Flex Film (analog flex not shown)



Test Jig to communicate with sensors

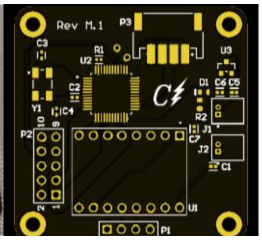


Sensor Breakout board (without sensor)

- Utilized Altium tools such as vias, castellated edges, polygon pours, teardrops, design rule checks, and soldering pads/masks to create rigid COM boards as well as flex films
- Designed numerous CAN to USB digital COM boards to convert analog strain gauge data from the force sensors to digital counts using an ADC
- Optimized signal strength and limited EMI by incorporating shielding pads, adding copper pours on the top and bottom layers, and ensuring the traces are routed appropriately for digital/analog PCBs
- Populated test jigs to communicate to the sensors through UART, USB, CAN, and Ethernet
- Procured and kitted components from Digi-key and then utilized hot air rework/soldering stations, microscopes, solder wicks, UV baths, and soldering techniques to populate the board
- Followed the board-bring up process to ensure the jig functions based on the Altium footprint by testing continuity on the DMM, oscilloscope probing SPI and clocks, and testing drivers for the ICs

STM32 Rotating Trophy Display | Altium 2023

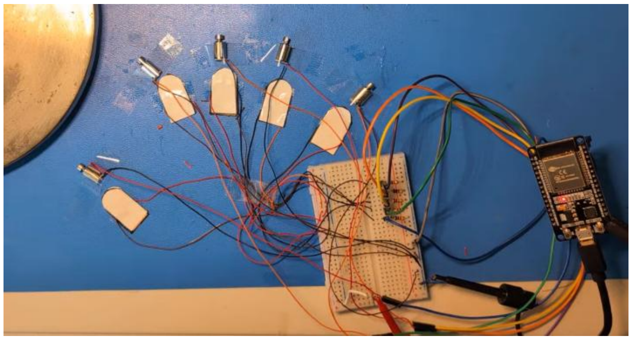
Populated Board



Footprint

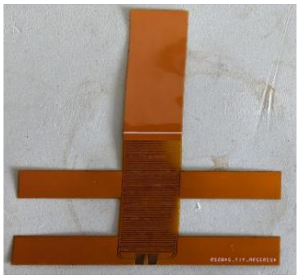
- Created a rotating trophy display using a NEMA17 stepper motor, A4988 motor driver, STM32, and ASM1117 regulator
- Designed the PCB in Altium using decoupling capacitors, shielding, pull-down resistors, and routing practices to limit noise
- Populated and procured components from Digi-key using soldering and hot air rework stations to meet cost constraints

Air Piano V1 | Soldering, Hot Air Rework, Hot Plate, EasyEDA 2024

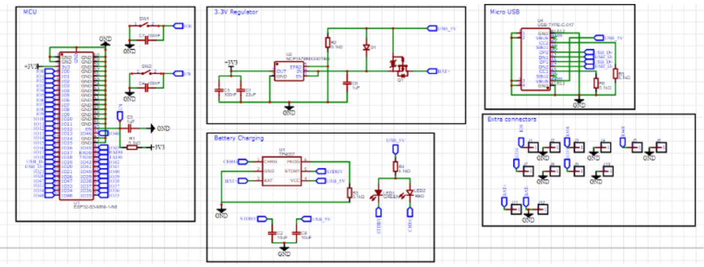


ESP32Wroom32D Initial demo

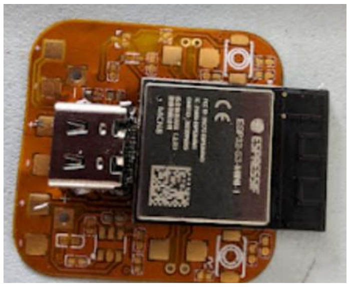
Arduino Nano 33BLE Pianist and Listener Interface



Force Sensitive Resistors



Pianist V2 Schematic



ESP32mini V2 PCB

- Tested wireless BLE and Wi-Fi COM using Teensy4.0, EP32Wroom32D, and Arduino Nano BLE33
- Researched and tested FSRs using copper tape, conductive polymer material, and insulators
- Custom designed and fabricated FSRs for fingers using interdigitated flex PCB and velostat on EasyEDA
- Experimented with Edge ML configurability to detect hand motions with MPU6050 IMU using model generated on Edge Impulse and Tensorflow Lite
- Created first prototype using FSRs, Arduino Nano BLE33, BJTs, 3.7 Lipo Battery, and perf boards
- Created PCB design for ESP32mini for V2 using 3v3 regulator with MOSFET control, TP4057 linear charging, and USB-C debugging port

Light Saber Pool Cue | Fusion360 2024

Electronic Housing

Fusion360 Assembly

Light Saber Pool Cue

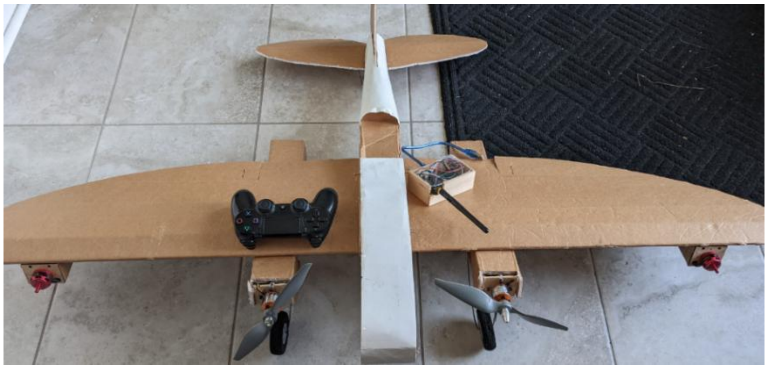
- Modded Pool Cue stick with light, audio, aim, and spring-loaded trigger using LEDs, laser module, and vibration sensor
- Designed and fabricated screw-adjustable encasing and trigger mechanism using Fusion360, Plexi glass, and repurposed hardware

Teensy 4.0 Audio Easy Button 2024

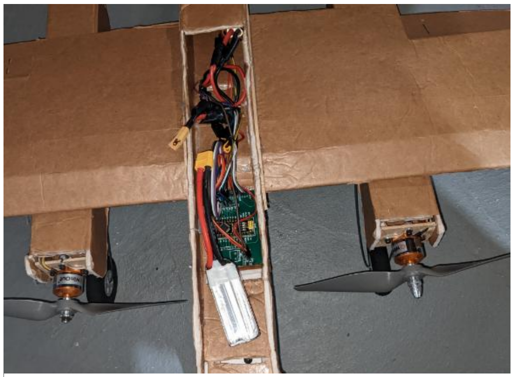
Button

- Integrated Easy Button with Teensy 4.0 Audio Sheild to play motivational audio on SD card for friends
- Created audio amplification circuit using two BJTs, adjustable POT, and signal conditioning capacitor

Quad-Engine Autopilot Plane | SolidWorks, AutoCAD 2022

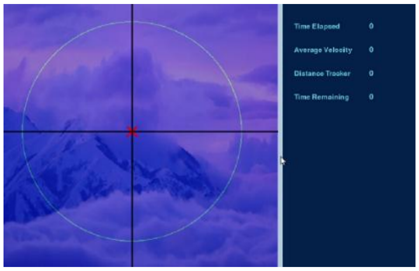


Quad-Engine Plane



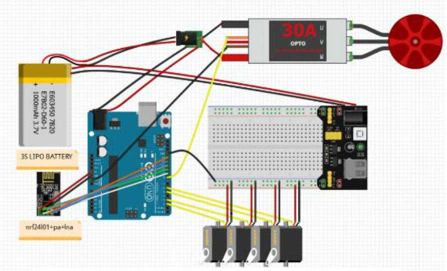
Electronic components for receiver

Transmitter for the Controller



Time Elapsed 0
Average Velocity 0
Distance Tracker 0
Time Remaining 0

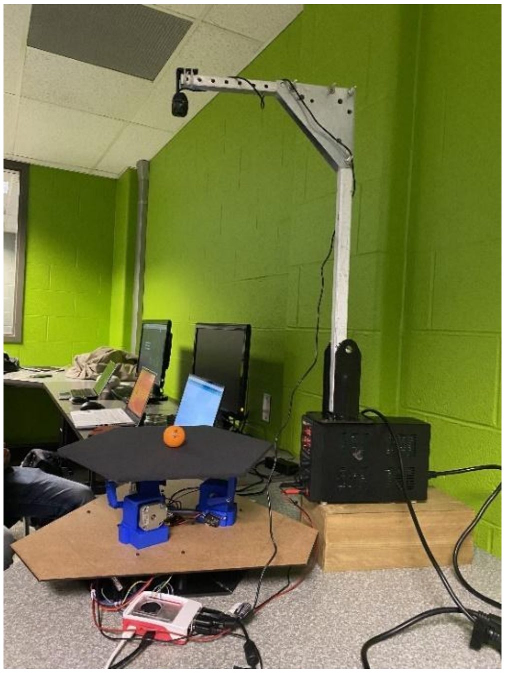
Radar tracking GUI



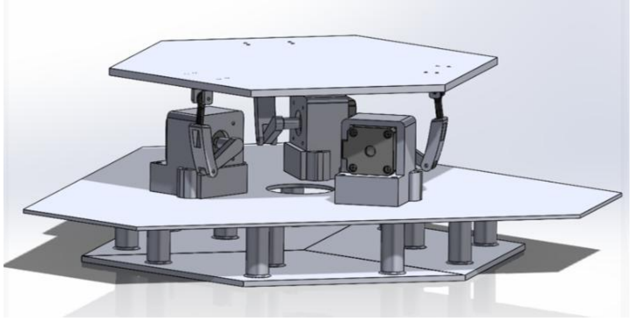
General Receiver Schematic

- Created an RC plane with an autopilot feature using Arduino, Python GUI, NRF24L01 module for radio communication, and a PS4 controller
- Utilized Altium and Fusion360 to create schematics for the electromechanical components which include GPS Beitian 880, NRF24L01, 30A ESC, micro servos, MPU6050 accelerometer, and Arduino
- Created an autopilot algorithm which takes location input and triangulates the destination distance using the haversine formula to maneuver the plane according to the bank and steer functions
- Communicated through SPI to transmit and receive PS4 controller, location, and gyro data

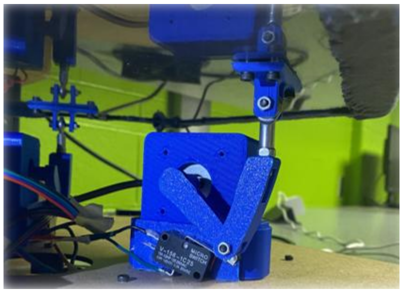
Stewart Platform | Python, C++, SolidWorks 2024



Stewart Platform Setup



SolidWorks Mechanical Assembly



Limit Switches for Homing Sequence

- Created a Stewart platform in SolidWorks using OpenCV to self-balance a ping pong ball under dynamic disturbances by integrating a RPi, Arduino, stepper motors, motor drivers, and webcam
- Optimized a PID controller in Python by fine-tuning Kp, Ki, and Kd parameters to increase the platform's response time and decrease overcorrection movement
- Designed a homing sequence in C++, utilizing limit switches to calibrate the initial arm position
- Tracked events using Python CSV logs to catch and debug issues such as connection of I2C line and errors in inverse kinematic calculations

Community Repair Hub | Fixer 2024

Fixed/Fixing Items

- Repaired electronics at Community Repair Hubs with skilled technicians to repurpose items
- Explored design and manufacturing of appliances for scale to learn more about consumer electronics by repairing fans, cameras, blenders, refrigerators, and heaters