OpenSense v3

An Open Source, Adaptable Sensor Platform

James Eyler
System/Power Design
Project Management
Computer Engineering
BSCpE

Brandon Collins
Software Design
Computer Engineering
BSCpE

Noah Wilfond
Photonics Sensor Design
Photonics Science and Engineering
BSPSE
Electrical Engineering
BSEE

Daniel Gonzalez
Sensor Design
Computer Engineering
BSCpE

Motivation and Background

Problem: Many modern sensor platforms are expensive, closed-source, and difficult to customize, creating barriers to learning, especially in K-12 education where accessibility and transparency matter most.

Educational Focus: Drawing from our own experience as students, we aim to lower technical barriers and make circuit design and embedded systems more approachable for educators and learners.

Foundation: OpenSense, developed by **DRACO Labs**, is an open-source, low-cost, and scalable sensor platform designed specifically for educational use.

Demonstration & Vision: By developing a custom optical spectrometer, we showcase our sensor platform's flexibility and customization, advancing DRACO Labs' mission to empower the next generation of innovators.

Our Contribution: We build on OpenSense by expanding its capabilities, improving usability, and supporting a broader range of sensors.

Basic Goals and Objectives

Design a data logger capable of acquiring and recording outputs from multiple sensors simultaneously.

- Implement a data acquisition system capable of sampling each connected sensor at a minimum rate of 10 Hz.
- Support simultaneous communication between a central MCU and at least 5 sensors over I²C.
- Establish a baseline battery life that will guarantee a minimum of 2 hours of battery life.
- Be able to store data from sensors onto an SD card, inside of a .csv file.
- Ensure that sensor readings achieve a minimum accuracy of 85% in relaying correct values.

Develop a user-configurable conversion interface to present the information in user-readable units.

Maintain a minimum of 1 hour of onboard storage for collected data.

Publish the system and associated documentation under an open-source license to enable community access, collaboration, and further development..

Design an optical spectrometer module to establish proof-of-concept integration with the sensor platform, focusing on accessibility and educational use.

- Implement a diffraction-grating-based spectrometer with a single detector to verify integration of custom sensors with the platform and produce basic spectra for educational demonstration.
- Adjust the angle of the diffraction grating with a servo motor to vary the detected wavelength.

Advanced Goals and Objectives

Integrate a power supply subsystem capable of sustaining continuous operation for a minimum of two hours.

- Establish a baseline battery life that will guarantee a minimum of 6 hours of battery life.
- Data collection will be able to store up to 4 continuous hours of data.

Utilize a microcontroller to control the data acquisition subsystem and distribute power to connected components as required.

Create a housing that enables system portability by minimizing weight, reducing volume, and ensuring robustness during transport and deployment.

Implement the system for straightforward setup to minimize deployment time and required user training.

Develop a software framework that supports user-defined sensor models, increasing customization and expanding supported sensor options.

- Extend the data-acquisition system to include an intuitive interface that simplifies onboarding new sensors.

Demonstrate the platform’s flexibility by using it to process data from an optical spectrometer and resolve distinct spectral peaks.

Incorporate GUI support for visualizing data in real time, enabling intuitive interpretation for educational purposes

Demonstrate the platform’s flexibility by constructing an optical spectrometer capable of resolving distinct spectral peaks.

- Utilize a photodiode within an optical spectrometer setup to demonstrate the customizability of the sensor platform.
- Evaluate resolution performance by measuring and distinguishing multiple spectral lines from calibration sources.

Achieve a minimum wavelength coverage of 400-700 nm (visible spectrum) to ensure relevance for education and demonstration purposes

- Employ reflective optics to ensure broadband compatibility and minimize chromatic aberrations.

Stretch Goals and Objectives

Enable wireless data transmission for automated upload to external devices or storage systems.

Enable the USB port to power the system and also recharge batteries (Lithium).

- Establish a baseline battery life that will guarantee a minimum of 24 hours of battery life.
- Data collection will be able to store up to 8 continuous hours of data.

Incorporate modular sensor ports to support “plug-and-play” integration and replacement of multiple sensor types.

- Incorporate a custom library that will allow users to input parameters of their own I²C sensor - effectively making the platform functional with any sensor that utilizes I²C communication Protocols.

Ensure that sensor readings achieve a minimum accuracy of 95% in relaying correct values.

Develop a cross-platform interface for the visualization of data to ensure broad accessibility and usability.

Implement spectrometer data visualization tools into the GUI to make optics more approachable for students and educators.

- Develop interactive plotting features to display real-time spectral data.

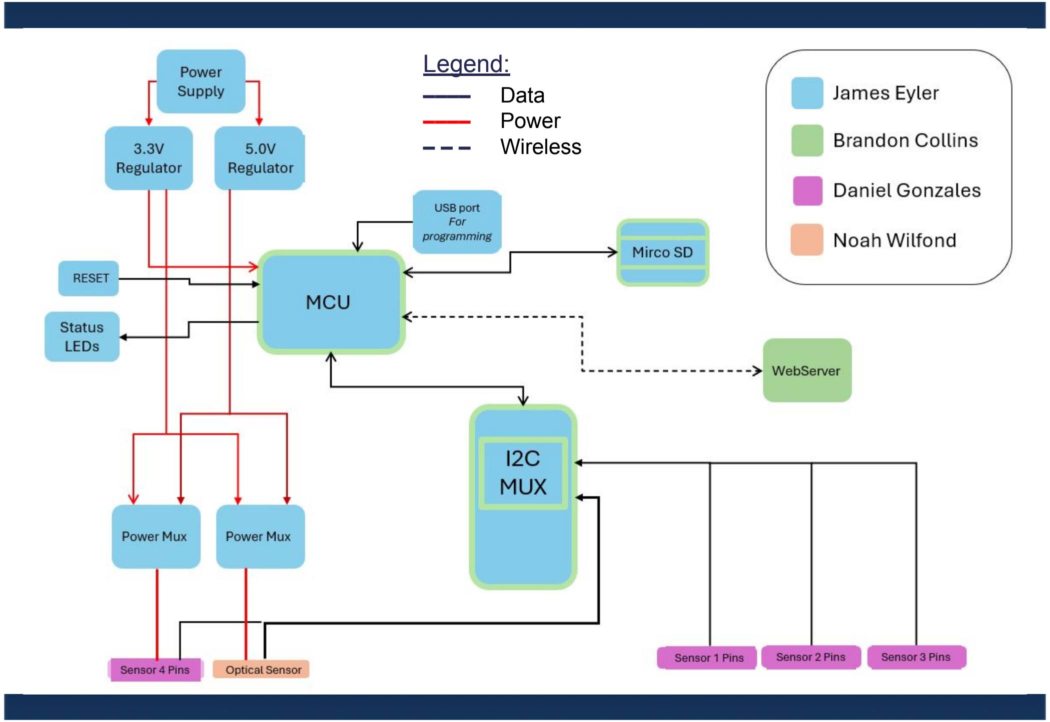
Extend the wavelength of the spectrometer subsystem into the near-UV or near-IR to broaden applications.

- Modify optical components to ensure compatibility with UV/IR operation.

Table of Engineering Specifications

System Specification Table		
Specifications	**Description**	**Target Values**
Unit Cost	The platform should have a low cost per unit for volume	< $60
System Lifespan	The device should be guaranteed to support data collection over a set period of time	≥ 2 hours
Sampling Rate	The platform should be able to record multiple points of data over time	At least 10 Hz per sensor
Data Storage	The device should have enough storage to gather significant data for analysis	≥ 1 hour storage
Sensor Accuracy	The sensor should collect data accurately	Accuracy ≥ 85%
PCB Dimensions	The PCB should be made to fit in a compact/portable housing	≤ 200cm2
Sensor Support	The platform should be able to support 5 or more sensors	≥ 5 Sensors
Spectrometer wavelength range	The spectrometer should be able to plot a spectrum consisting of at least the visible wavelength as well as some UV wavelengths.	300 - 800 nm
Spectral Resolution	To be able to distinctly measure different wavelengths, a spectral resolution of at least 5 nm must be met.	≤ 5 nm

Hardware Diagram



- Sensors connect to MCU
  - Two of the 5 sensor paths will have a switching path
- Sensors send data to MCU
- MCU then transfers data to the SD card
- Other side items
  - LEDs - for life/heartbeat
  - Reset/Boot button

Choosing an MCU

ESP32-WROOM

- Core/Speed
  - Dual Core
  - Up to 240MHz
- Memory
  - 384kB ROM
  - 512kB SRAM
- Connectivity
  - Wifi / Bluetooth
- Languages
  - ESP-IDF
  - Arduino

STM32UO

- Core/Speed
  - Single Core
  - 56MHz
- Memory
  - 256kB Flash Memory
  - 512kB SRAM
- Connectivity
  - Nothing Listed
- Languages
  - C
  - C++

MSP430

- Core/Speed
  - Single Core → 16-Bit
  - 16MHz
- Memory
  - 15.5kB FRAM
- Connectivity
  - Nothing Listed
- Languages
  - C

- The ESP32 allows for the implementation of FreeRTOS tasks making the use of concurrency easier to implement

Choosing an I^2C Multiplexer

TCA9548APWR	PCA9547PW	LTC4305IGN
• Number of Channels ◦ 8 • Clock Frequency ◦ 0 to 400kHz • Voltage Logic ◦ 1.8V, 2.5V, 3.3V, 5.0V	• Number of Channels ◦ 8 • Clock Frequency ◦ 0 to 400kHz • Voltage Logic ◦ 1.8V, 2.5V, 3.3V, 5.0V	• Number of Channels ◦ 2 • Clock Frequency ◦ 0 to 400kHz • Voltage Logic ◦ 1.8V, 2.5V, 3.3V, 5.0V

- TCA9548APWR is a more popular IC, because of this there is more available documentation on circuit orientation for the device
    - There are development breakout boards (with schematics) for this IC specifically

Choosing a Level Shifter

NTB0102DP	TXS0102DCUR	74LVC1T45GW
• Translation Speed ○ 100Mbps • Voltage A (VCCA) ○ 1.2V ~ 3.6V • Voltage B (VCCB) ○ 1.65V ~ 5.5V • Direction ○ Auto Bidirectional sensing	• Translation Speed ○ 2Mbps • Voltage A (VCCA) ○ 1.65V ~ 3.6V • Voltage B (VCCB) ○ 2.3V ~ 5.5V • Direction ○ Auto Bidirectional sensing	• Translation Speed ○ 420Mbps • Voltage A (VCCA) ○ 1.2V ~ 5.5V • Voltage B (VCCB) ○ 1.2V ~ 5.5V • Direction ○ Bidirectional (non-auto, would not work with I^2C)

- A major thing with Level Translators is the speed, it is necessary to find an IC that is fast and operates using an Open-drain directionality

Choosing a Regulators (3.3V)

LM2596SX-3.3/NOPB	MP2315	LM2623
• Switching Frequency ○ 150kHz • Vin_{MIN} ○ 4.5V	• Switching Frequency ○ 500kHz • Vin_{MIN} ○ 4.5V	• Switching Frequency ○ 300kHz - 2MHz • Vin_{MIN} ○ 0.8V

- Switching Frequency is very important
  - Regulators with lower switching frequency are less sensitive to board design than those with higher
  - <= 1MHz is recommended (Dr. Weeks)

Choosing a Regulators (5.0V)

LM3481MMX/NOPB	TPS563201	LM27402
• Switching Frequency ○ 100kHz ~ 1MHz • Vin_{MIN} ○ 2.97V ~ 48V	• Switching Frequency ○ 580kHz • Vin_{MIN} ○ 4.5V	• Switching Frequency ○ 200kHz - 1.2MHz • Vin_{MIN} ○ 3V

- Switching Frequency is very important
  - Regulators with lower switching frequency are less sensitive to board design than those with higher
  - <= 1MHz is recommended (Dr. Weeks)

Choosing a Humidity Sensor

SHTC3	SHT31-DIS	HDC1080
• Accuracy (RH) ○ ±2%RH • Supply ○ 1.62V - 3.6V • Package ○ 2.0 x 2.0 mm	• Accuracy (RH) ○ ±2%RH • Supply ○ 2.15V - 5.5V • Package ○ 2.5 x 2.5 mm	• Accuracy (RH) ○ ±2%RH • Supply ○ 2.7V - 5.5V • Package ○ 4.0 x 4.0 mm

- Since the RH for these sensors are all the same, we chose to go with the SHTC3 due to being more low powered as well as having a smaller footprint to make the PCB layout easier.

Choosing a Temperature Sensor

LMT87DCK	TMP36	MCP9700A
• Accuracy (25°C) ○ ±0.4°C • Supply ○ 2.7V - 5.5V • Quiescent Current ○ ~5–10 μA	• Accuracy (25°C) ○ ±1°C • Supply ○ 2.7V - 5.5V • Quiescent Current ○ ~50 μA	• Accuracy (25°C) ○ ±1°C • Supply ○ 2.3V - 5.5V • Quiescent Current ○ ~6 μA

- We chose to go with the LMT87DCK because it had the greatest accuracy of of ±0.4°C while also having a low quiescent current as a lower quiescent current is always better.

Choosing a Pressure Sensor

7BB-27-4CL0 (Piezo Disk)	FSR 402 (Force Sensitive Resistor)

7BB-27-4CL0 (Piezo Disk)

- Best for
  - Measuring dynamic pressure such as taps or knocks
- Power
  - Requires essentially zero senor power other than the op-amp

FSR 402 (Force Sensitive Resistor)

- Best for
  - Measuring static pressure such as steady presses
- Power
  - Divider current draws current whenever powered

- We chose to go with a piezo disk for our sensor as we felt like for educational purposes it could be better for our sensor platform to work with a sensor that detects dynamic pressure rather than only static and it also consumes less power overall than a force sensitive resistor.

Choosing a Photodiode

Vishay TEMD5080X01	OSRAM SFH 2716 A01	Thorlabs FD11A

Vishay TEMD5080X01

- Spectral Range
  - 350 - 100 nm
- Active Area
  - 7.7 mm
- Cost
  - $1.8

OSRAM SFH 2716 A01

- Spectral Range
  - 350 - 1000 nm
- Active Area
  - 0.35 mm
- Cost
  - $0.86

Thorlabs FD11A

- Spectral Range
  - 320-1100 nm
- Active Area
  - 1.21 mm
- Cost
  - $1.20

- We chose to go with the TEMD5080 photodiode due to it having the necessary spectral range to meet our project requirements while also having a good cost and the largest active area making it the easiest to work with.

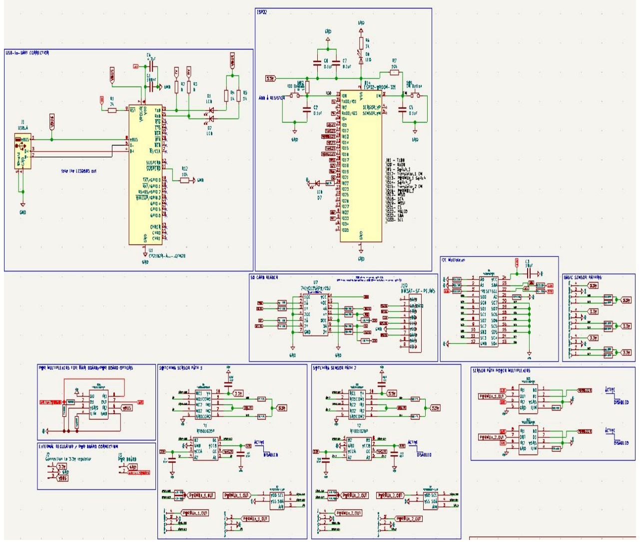
Choosing a Motor

Adafruit 1404	Adafruit 154	DFRobot SER0046
• Operating Voltage ○ 4.8 - 6 V • Position Feedback ○ Yes • Cost ○ $14.95	• Operating Voltage ○ 5 - 6 V • Position Feedback ○ No • Cost ○ $11.95	• Operating Voltage ○ 5 V • Position Feedback ○ No • Cost ○ $6.90

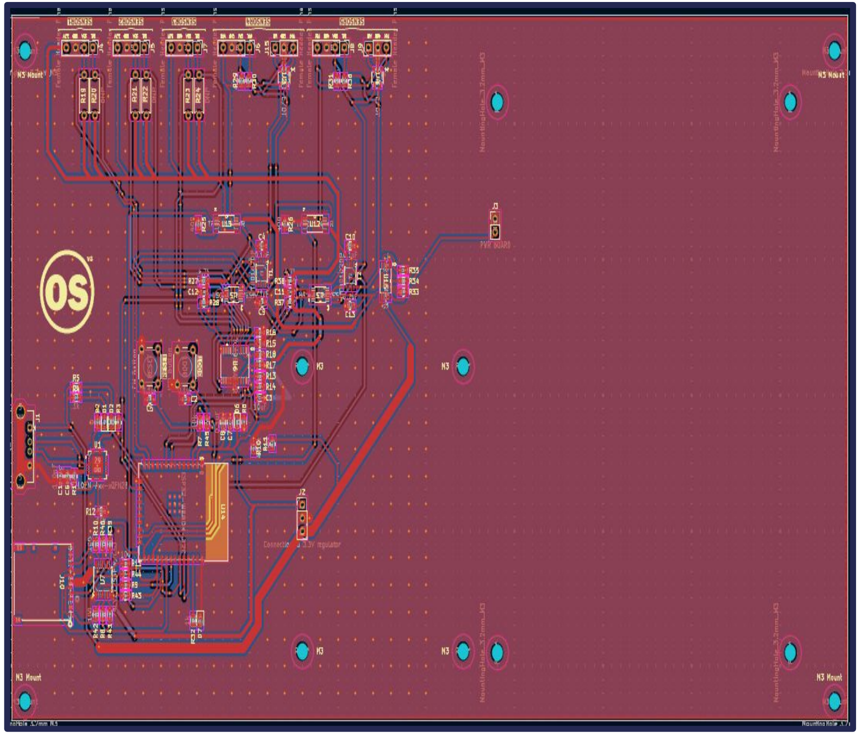
- We chose to go with the Adafruit 1404 servo motor since it has a built-in rotation feedback line. This line will allow us to measure the rotation of the servo, which is needed to determine the corresponding wavelength.

Main PCB (MCU PCB) Design

- Main Points of the MCU Board
  - 2 of the 5 sensors have a switching path
  - Main board connects to battery board and 3.3V regulator
  - Everything is design to be modular



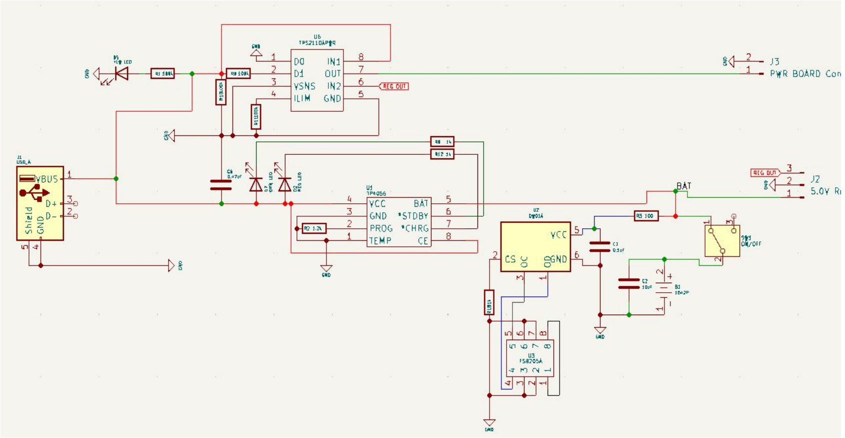
Caption Placeholder (also maybe split into multiple slides)



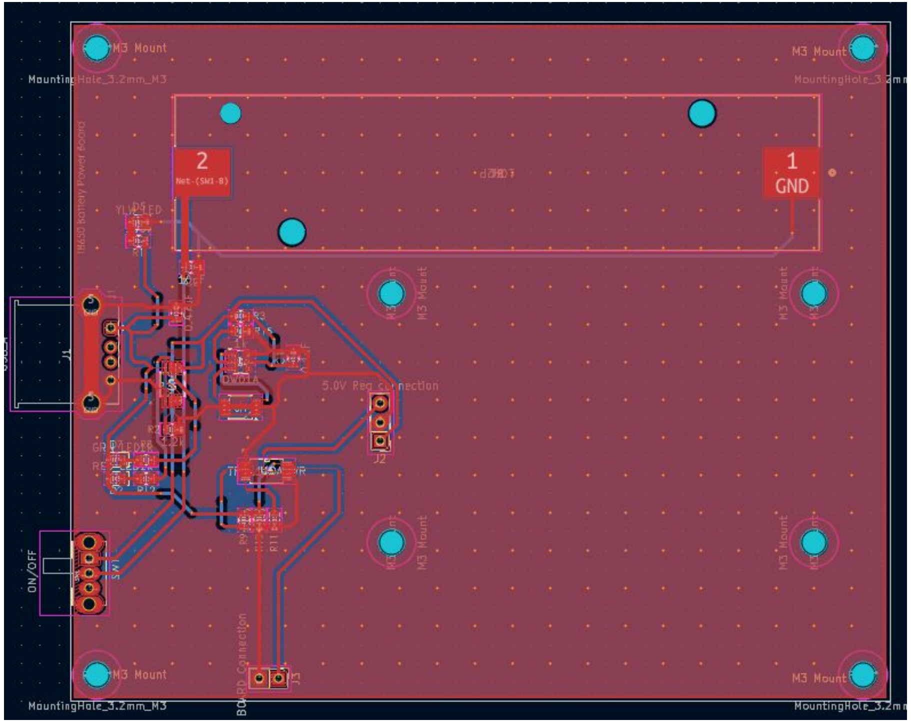
Caption Placeholder

Battery Power + Charging PCB Design

- References
  - TP4056 module board
- Important safety features
  - Overcharge/temperature sensitive IC
  - On/Off switch



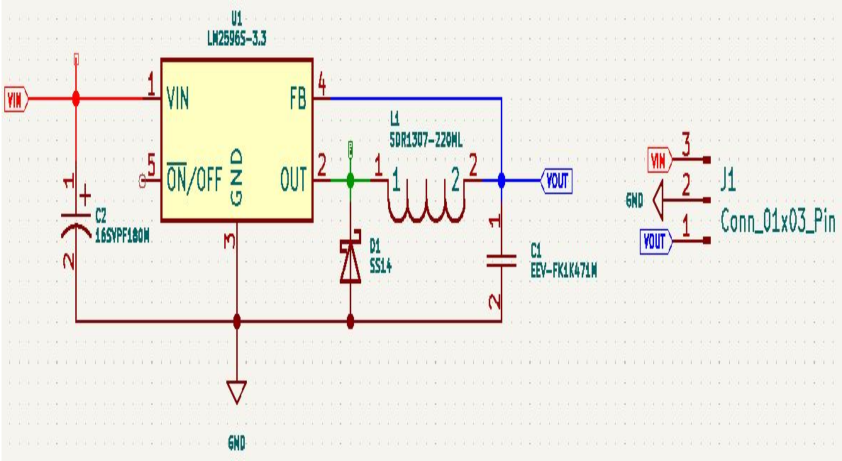
Caption Placeholder



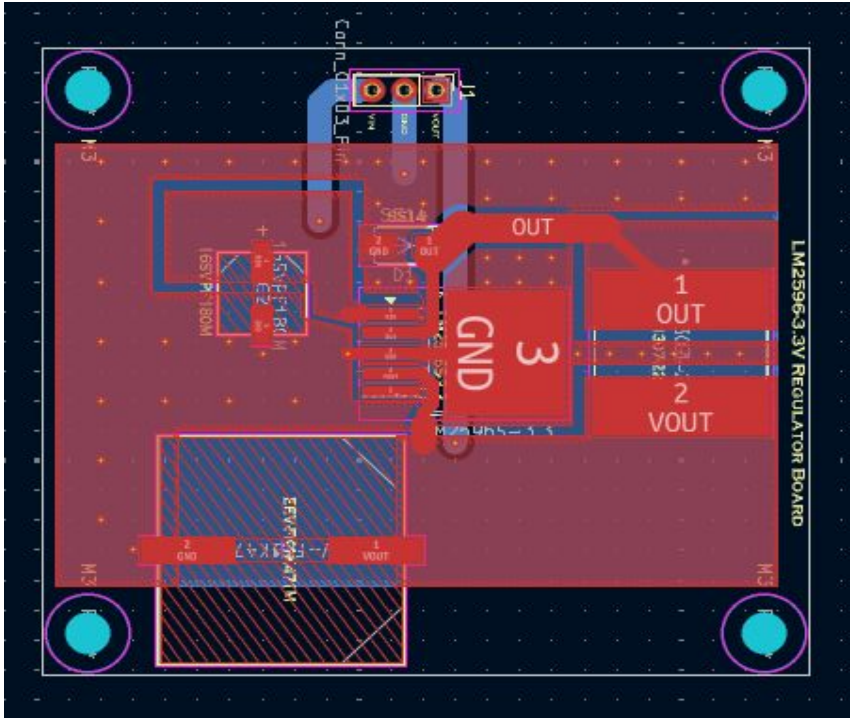
Caption Placeholder

3.3V Regulator Design

- Things to keep in mind
  - Vin/Vout need pwr planes
  - Use PCB layout from datasheet



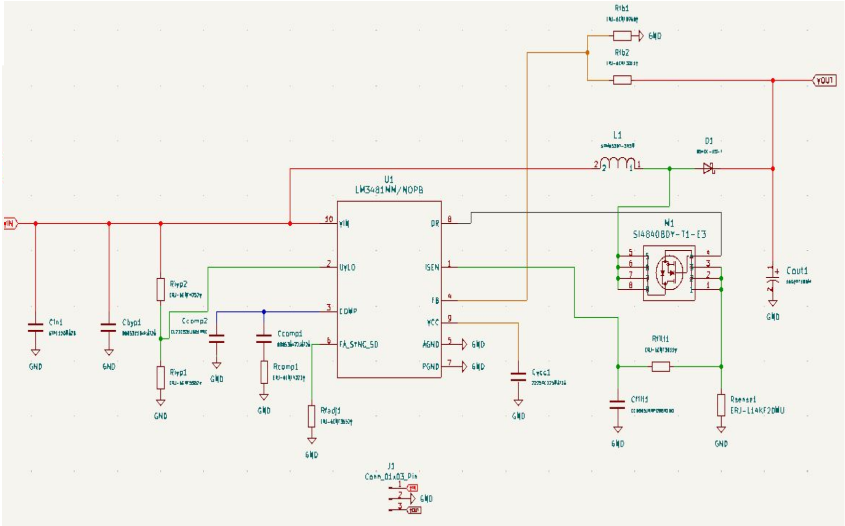
Caption Placeholder



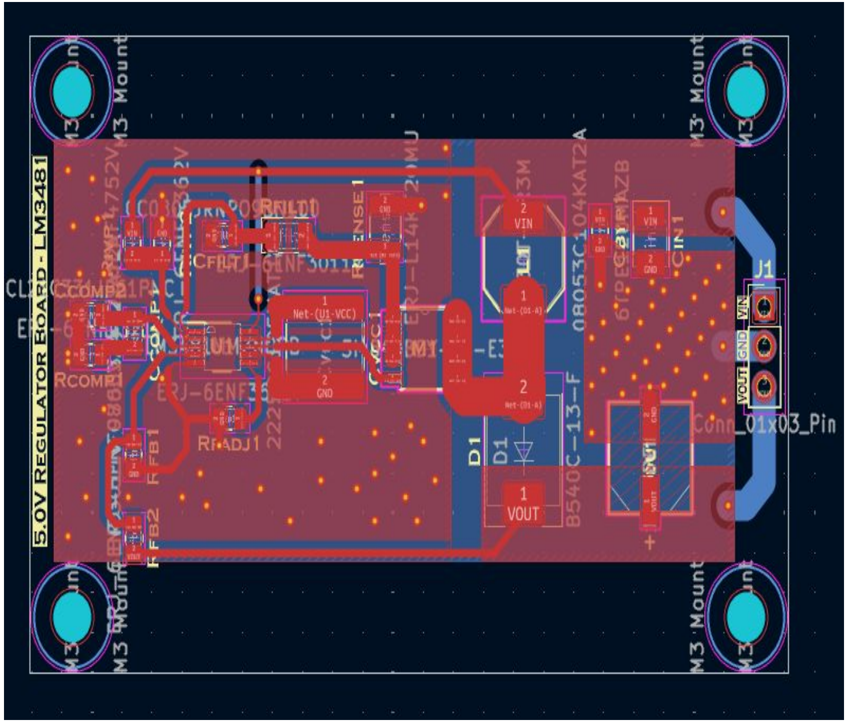
Caption Placeholder

5.0V Regulator Design

- Things to keep in mind
  - Vin/Vout need pwr planes
  - Use PCB layout from datasheet

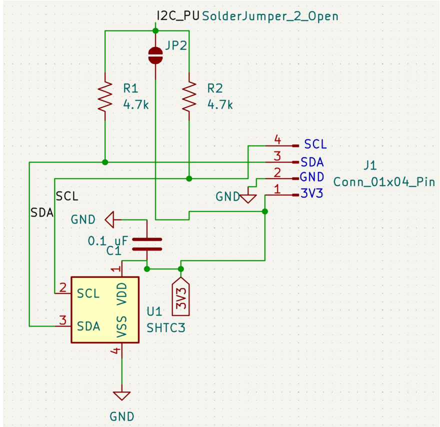


Caption Placeholder

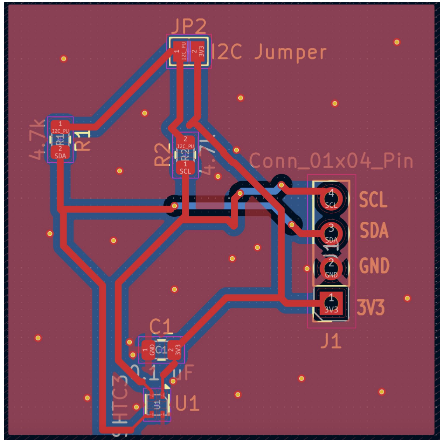


Caption Placeholder

Humidity Sensor Design



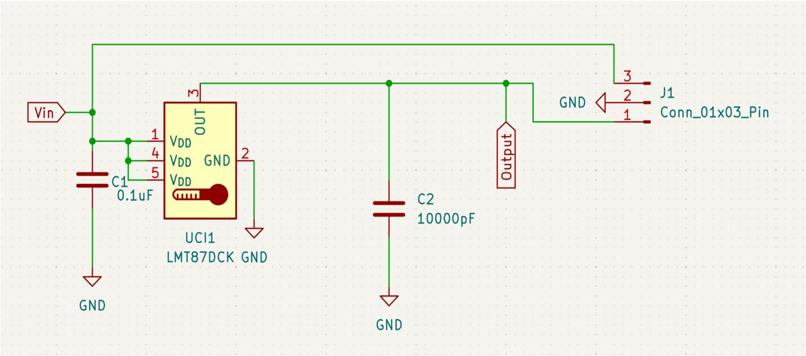
Caption Placeholder



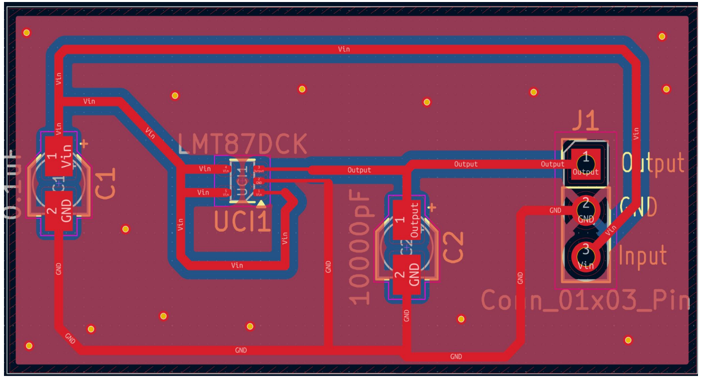
Caption Placeholder

- This design makes use of the SHTC3 humidity sensor along with a jumper, pull up resistors, and a decoupling capacitor.
- This design is makes use of an I2C connection to the main board.
- The PCB was properly routed using the proper thickness and spacing.
- Ground nodes were placed next to components connected to ground to try and make it more solid.
- The second layer was also used to make some connections smoother.

Temperature Sensor Design



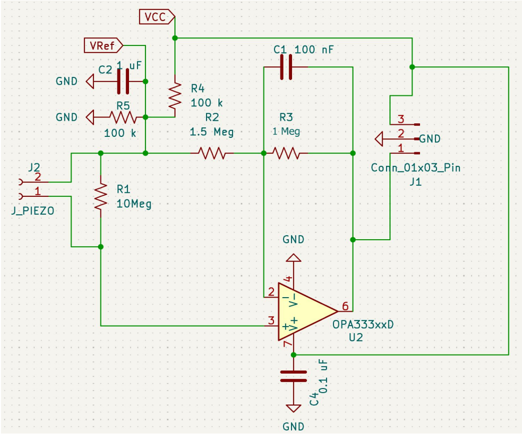
Caption Placeholder



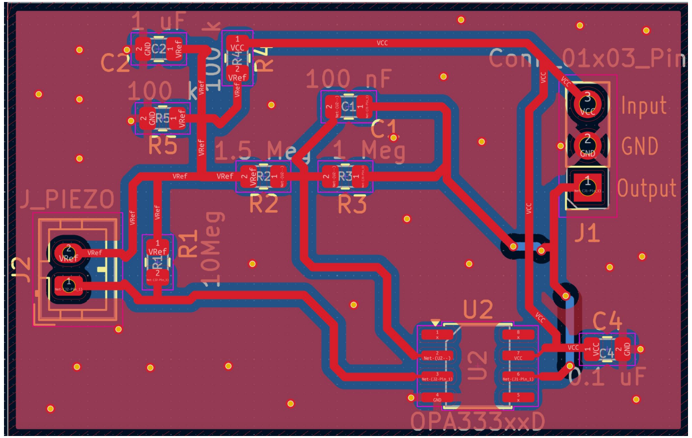
Caption Placeholder

- This design makes use of the LMT87DCK temperature sensor along with a decoupling and an output capacitor.
- This design makes uses of an analog connection to the main board.
- The PCB was properly routed using the proper thickness.
- Components were properly spaced out to avoid interference.

Pressure Sensor Design



Caption Placeholder

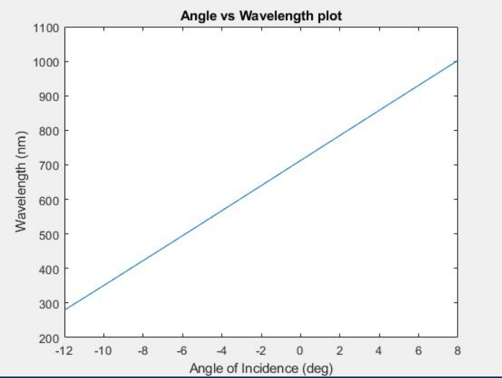


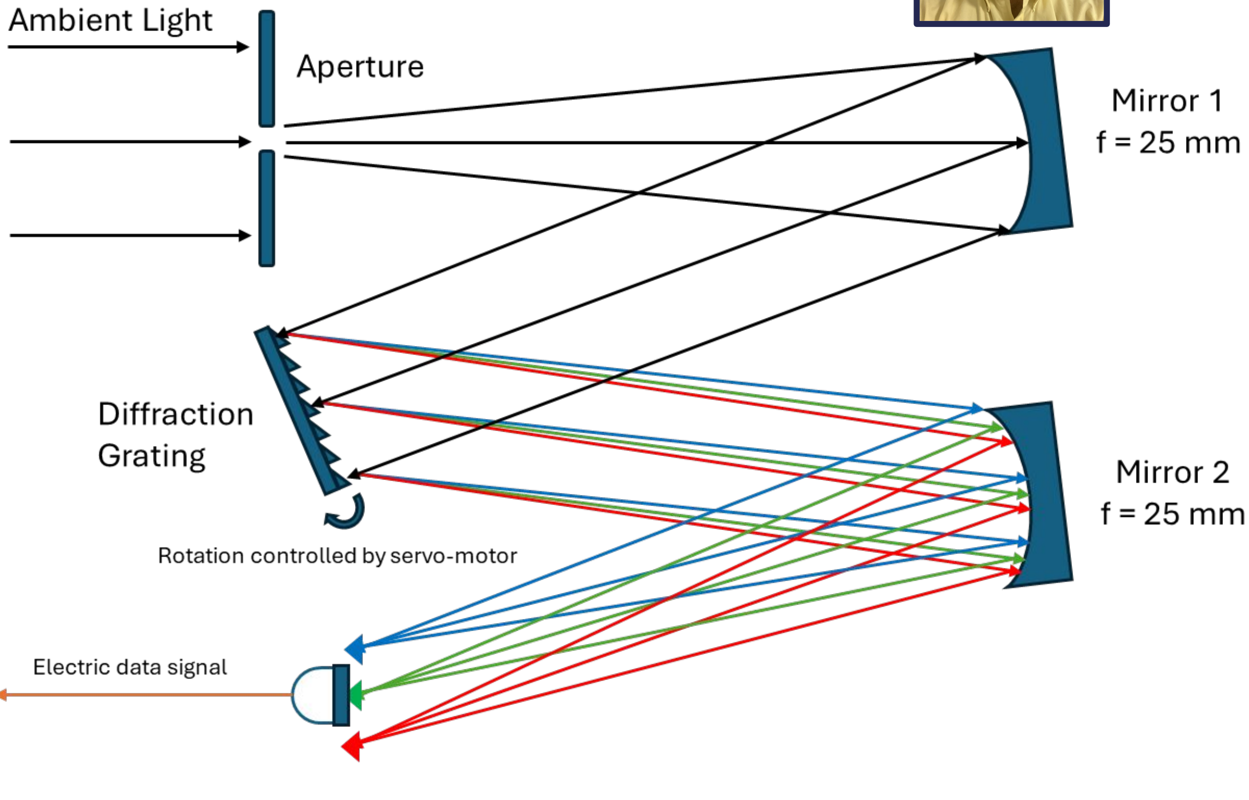
Caption Placeholder

- This design makes use of the 7BB-27-4CL0 buzzer serving as our Piezo disk as well as multiple decoupling capacitors and resistors.
- To ensure stability, the schematic and PCB are connected to a pin connector that will then connect to the Piezo disk rather than direct connection.
- The PCB was properly routed using the proper thickness and spacing.
- The second layer was used when needed to help make the connections smoother.

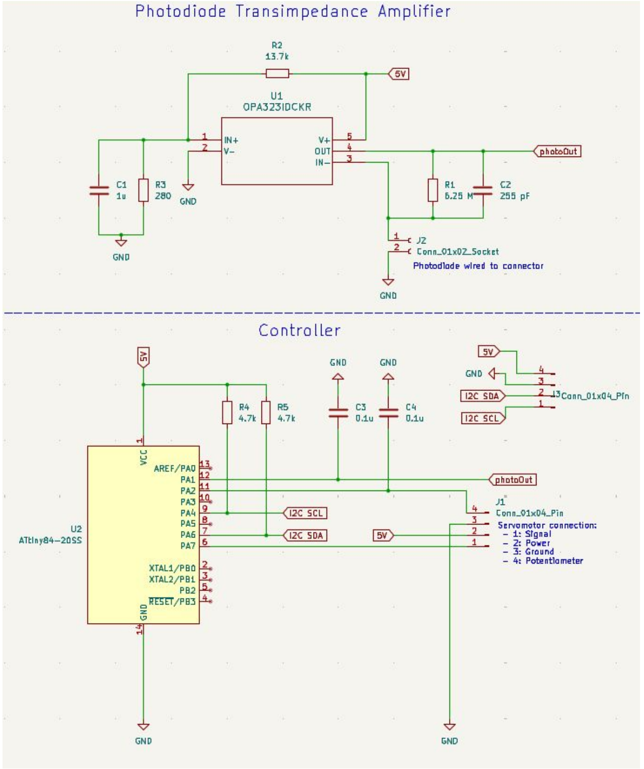
Monochromator - optical design

MATLAB Calculations

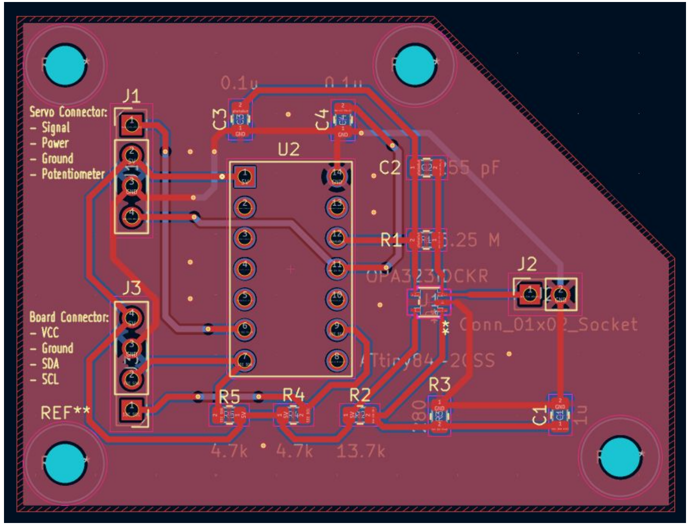




Monochromator - electrical design



Schematic for the monochromator PCB. Top half shows the transimpedance amplifier, and bottom half the motor controller.



PCB Layout for the monochromator.

- This design includes two main processes: the transimpedance amplifier and the controller.
    - The transimpedance amplifier converts the photocurrent from the photodiode into a measurable voltage.
    - The controller controls the rotation of the servo motor as well as records the output from the transimpedance amplifier and the potentiometer of the servo motor to transmit both optical and rotation data via I2C to the sensor platform for processing.

Choosing a Programming Environment

ESP-IDF
Native Framework

- C/C++ implementation
- Full system control
- Built-in HTTP, FAT, NVS
- Production-ready stability
- FreeRTOS foundation

ESP-IDF

Arduino Core
Rapid Development

- C/C++ with abstraction
- Extensive libraries
- Fast prototyping
- Community support
- IDF compatibility

ARDUINO™

MicroPython
Scripting Option

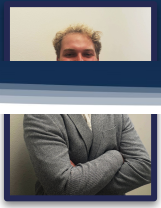
- Python implementation
- Fastest iteration
- Readable code
- Less deterministic
- Lower throughput

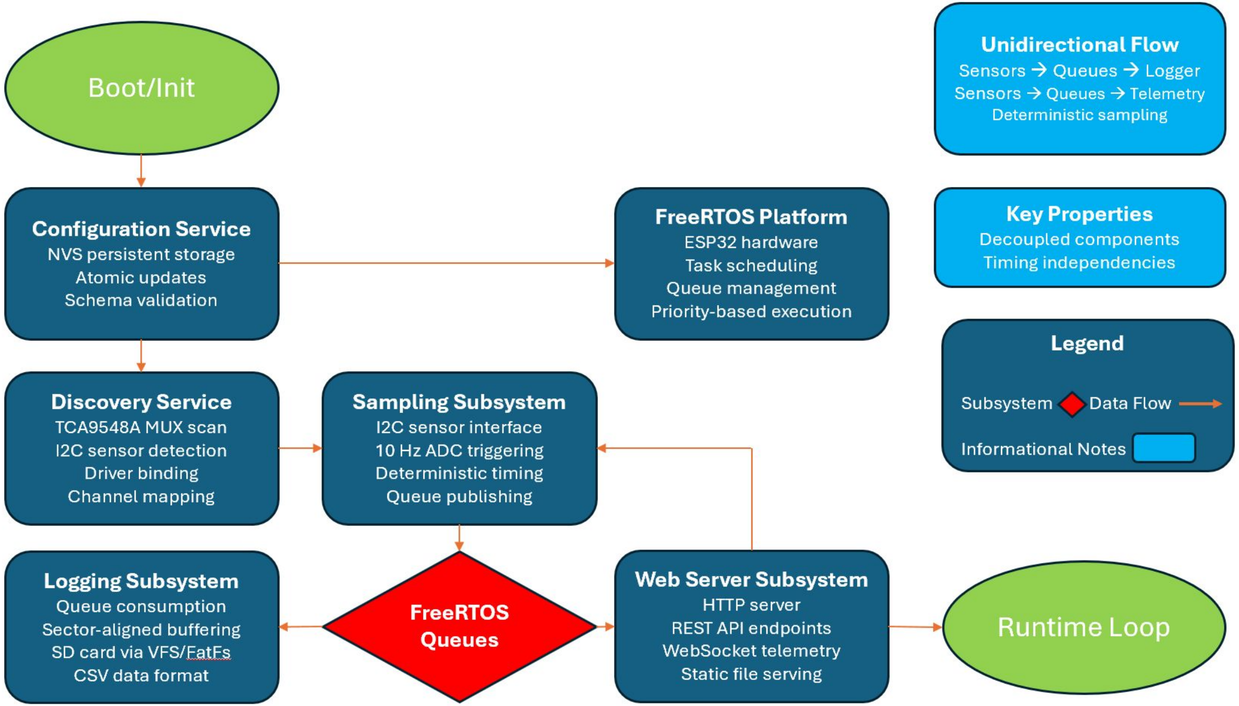
MicroPython

Programming Environments Technical Characteristics

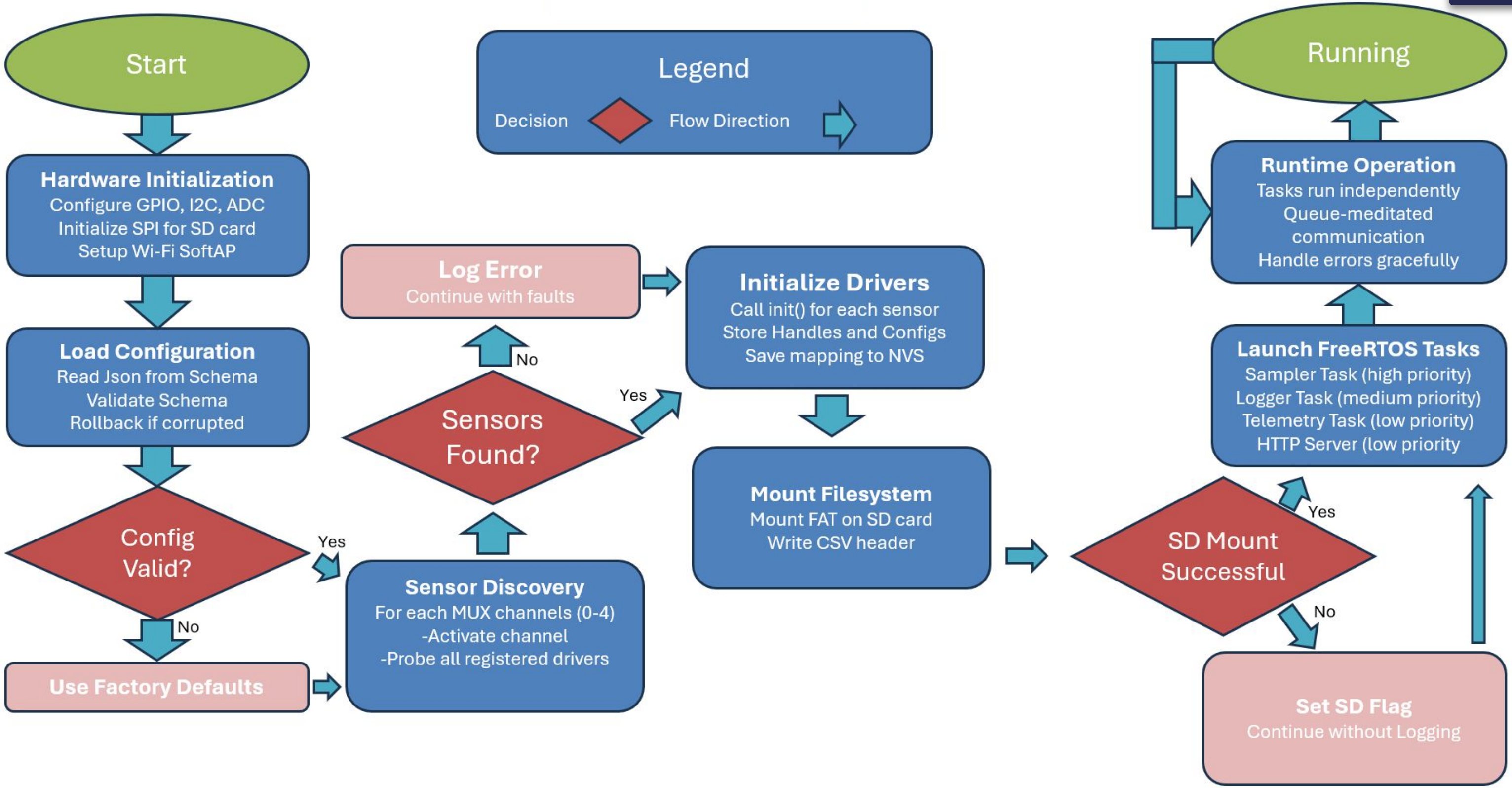
	ESP-IDF	Arduino Core	MicroPython
Language	C/C++	C/C++	Python
Library Support	Good Amount	Most Extensive	Limited
System Control	Full Control	Good Access	Moderate
Prototyping Speed	Medium	Fast	Fastest
Development Friction	Some	Minimal	Most
Performance	Highest	High	Lower
Determinism	Excellent	Excellent	Limited

Software Architecture

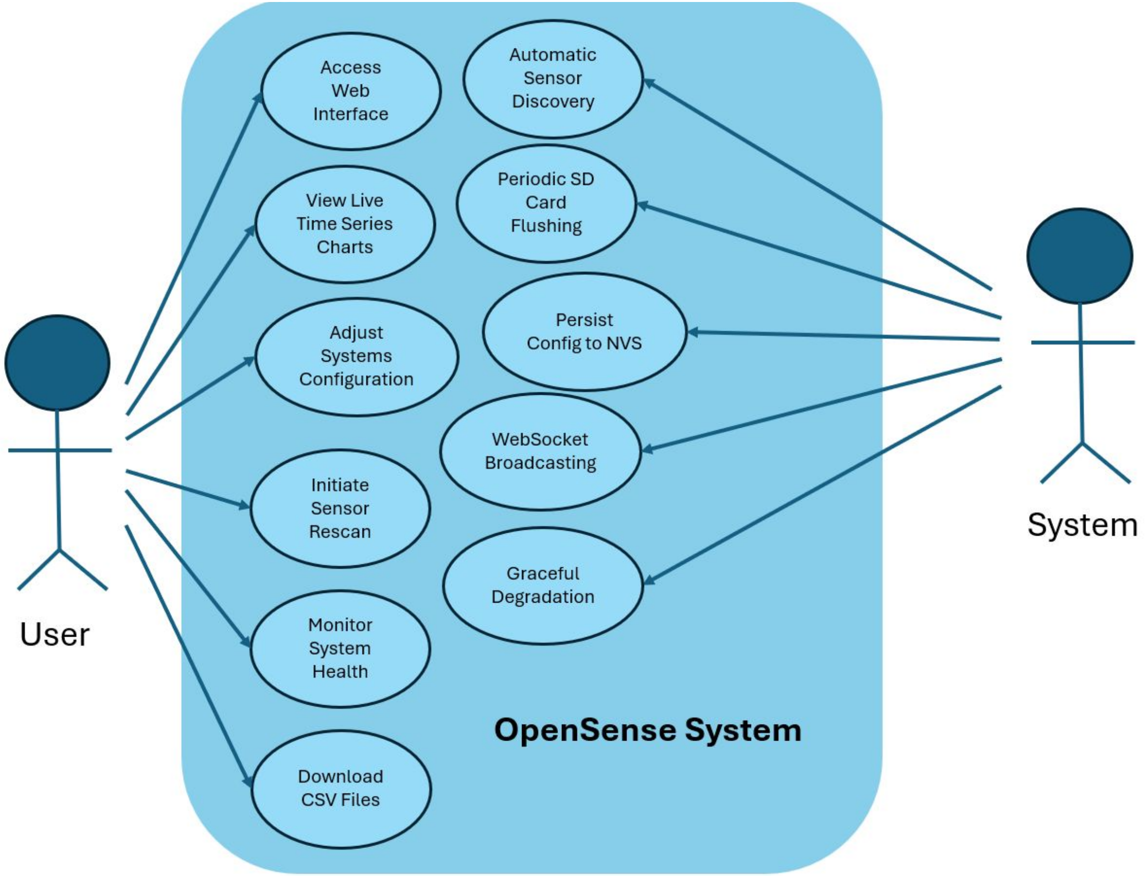




Main Program Flow

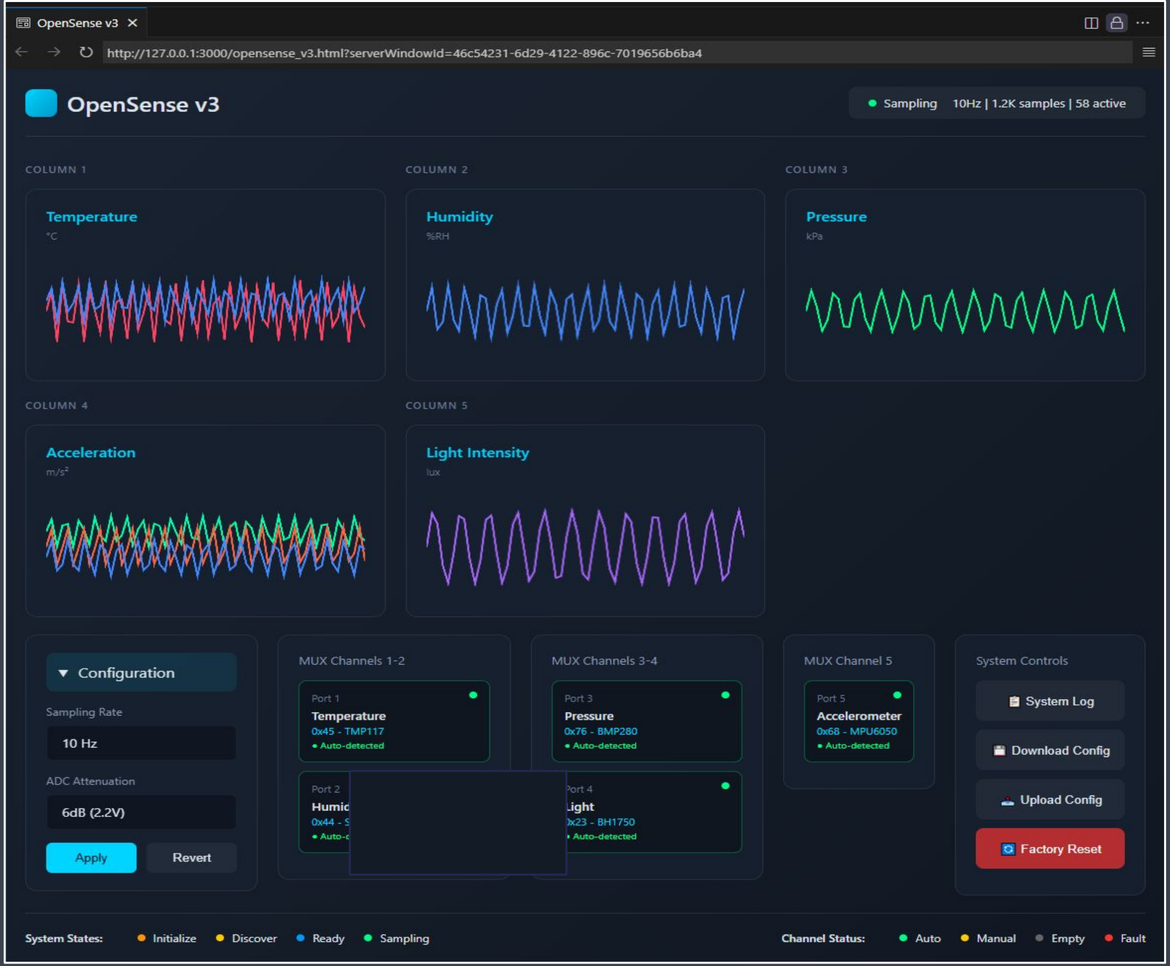


Use Case Diagram



- Description of diagram

Generated Concept User Interface

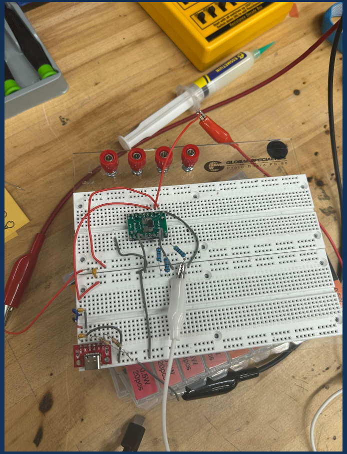


- Description of image

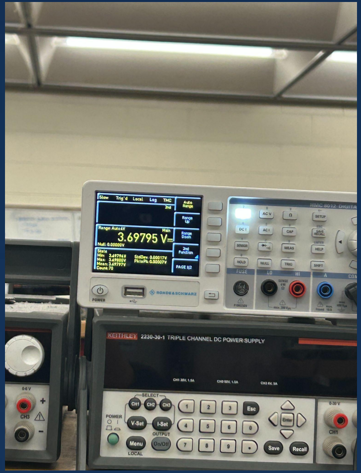
Prototyping and testing

Power System

- One of the tests we performed was testing our power source system. Our board is designed so that when it is plugged in to the charger it will automatically use that as its power source and if it is disconnected then it will immediately switch to the battery. In order to test this, we set up our initial design on a breadboard and hooked it up to a Power Supply to simulate the charger and the battery to see how the output would change based on which of them was connected and we got the results we expected.
- When connected solely to the battery the output was 3.69 which matches the expected output of just the battery and when connected to just the charger or to both the battery and charger the output was 4.89 showing it took priority.



CAPTION PLACEHOLDER



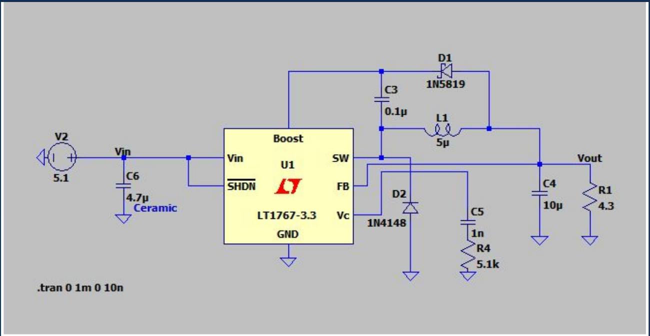
CAPTION PLACEHOLDER

CAPTION PLACEHOLDER

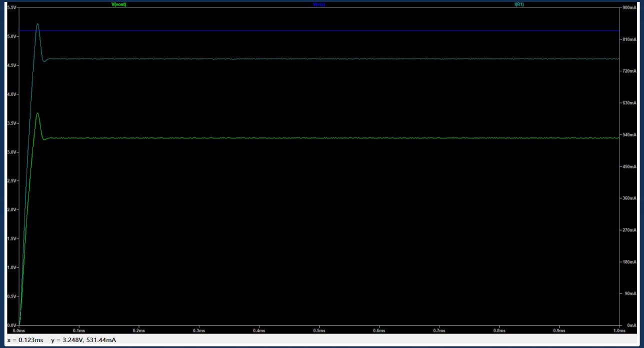
Prototyping and testing

Regulators

- When we bought our initial regulators, we attempted to test them on a breadboard along with the power circuit we would be using but after multiple failed attempts we learned that due to the nature of switching regulators it was very unlikely to work.
- This is because they require a tight layout and high frequency operation and the parasitic capacitance and the stray inductance on the vreadboards created a stable operation environment.
- As a result, we instead ran the regulators through a virtual simulation to ensure they would work since we couldn’t physically test them.
- We ended up adjusting our regulator choice since then with Dr. Week’s advice but below are the results from one of the simulations.



CAPTION PLACEHOLDER



CAPTION PLACEHOLDER

Prototyping and testing

Monochromator

- Two prototypes of the spectrometer system were conducted in order to verify the feasibility of the optical system.
  - The first prototype (left) contained a simplified system which uses lenses as substitutes for the mirrors.
  - The second prototype (right) contained all of the selected components for the optical system.
- These tests both used an optical breadboard to align components and manual rotation of the diffraction grating.
  - The tests verified the calculations showing the correlation between rotation and wavelength at the detector which our sensor is measuring.
  - The first prototype showed the importance of proper alignment of components – the diffraction grating wasn't directly above the axis of rotation causing a shift in position of the reflected beam. This was fixed in the second prototype.

The first prototype of the monochromator system.

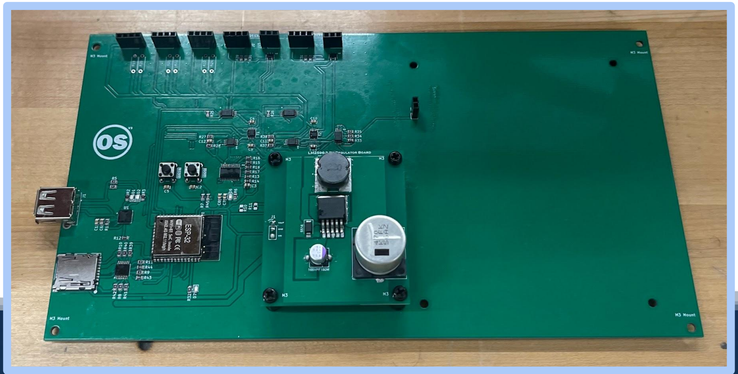
The second prototype of the monochromator system.

Prototyping and testing

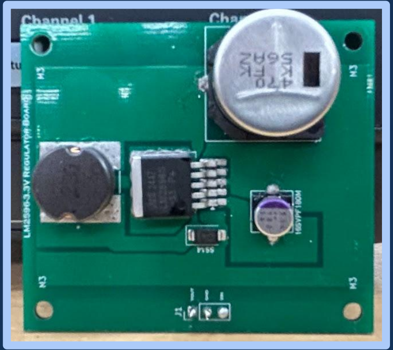
Board Fabrication

Top → Main (MCU) Board
Bottom Right → 3.3V Regulator Board
Bottom Left → 5.0V Regulator Board

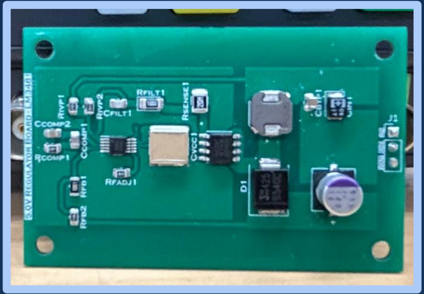
- Both Regulator boards output the desired voltage/amperage
- We have successfully uploaded test code to the Main Board showing a heartbeat LED working as intended
- Battery Board not yet fabricated due to a mislabeled footprint from Ultra Librarian



CAPTION PLACEHOLDER



CAPTION PLACEHOLDER



CAPTION PLACEHOLDER

Administrative Content

Work Distribution

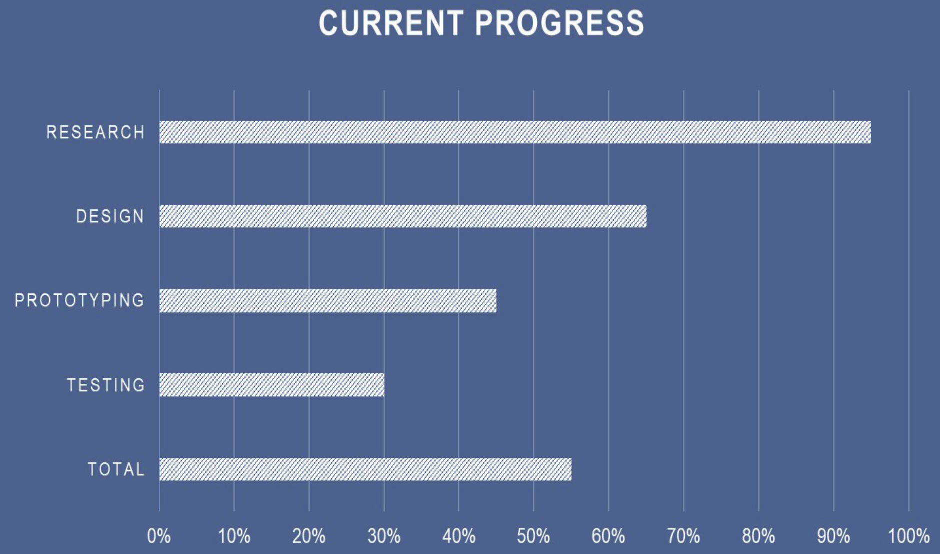
Name	MCU Board	Power system	Sensors	Software
Brandon				X
James (PM)	X	X		
Daniel			X	
Noah			X	

Administrative Content

Budget

Item	Cost
Main platform	$128.09
Power System	$53.24
Sensors	$197.71
Monochromator	$225.64
PCBs	$463.84
Total	$1058.52

Administrative content



Plan for Completion

Present

All PCBs approved and ordered

1-2 Weeks

PCB Assembling and Testing

3-4 Weeks

Programming and testing

5-6 Weeks

Midterm Demo

9 Weeks

Final Demo and Showcase