Multichannel EEG-Based Brain-Computer Interface for Real-Time Computer Control

ECE 49022 Group 11

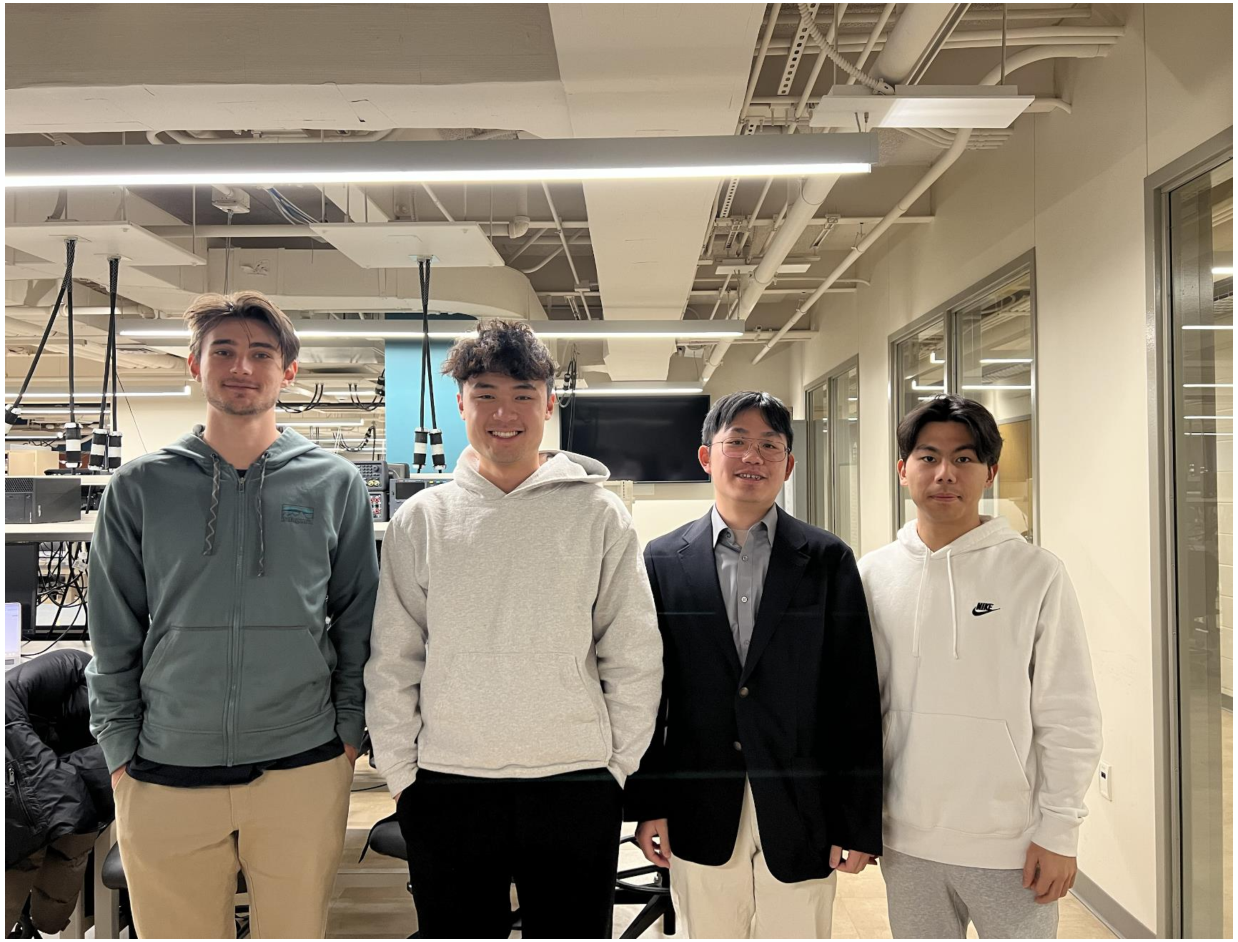
4/29/2025

PURDUE
UNIVERSITY

Team Member Introduction

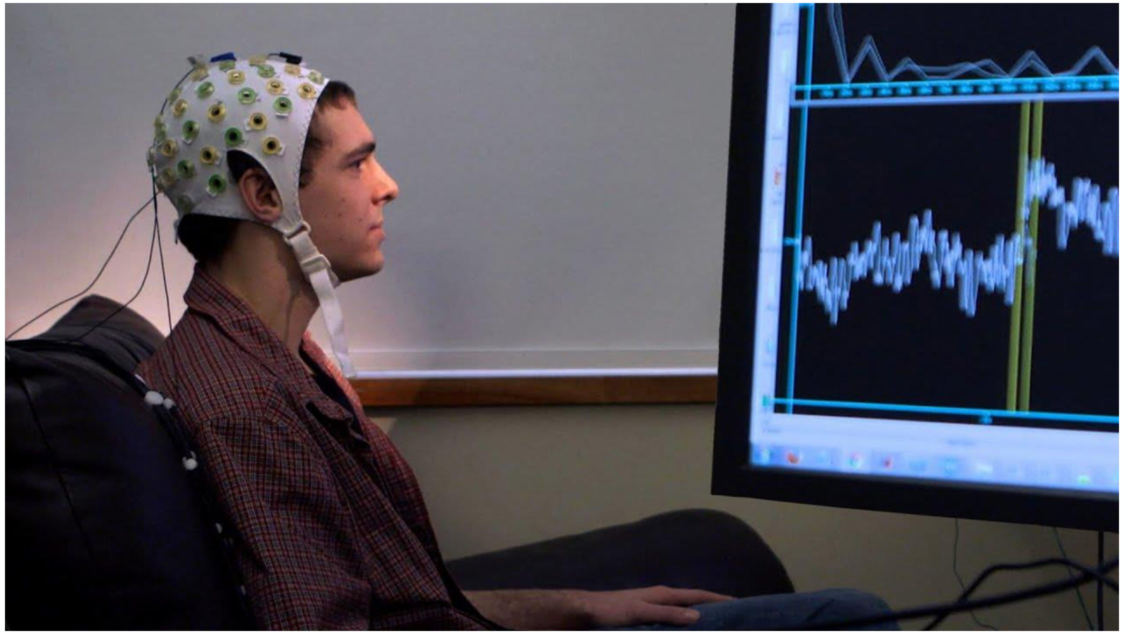
The Team(From left to right):

- Jasper Koliba (Team Leader)
  - Computer Engineering
  - Software algorithms and UI
- Alex Cheng (Purchasing)
  - Electrical Engineering
  - EEG Circuit and Headpiece
- Hao Yang (Communication)
  - Electrical Engineering
  - EEG Circuit and System power
- Yoshiki Takeuchi (Facilitator)
  - Electrical Engineering
  - Microprocessor



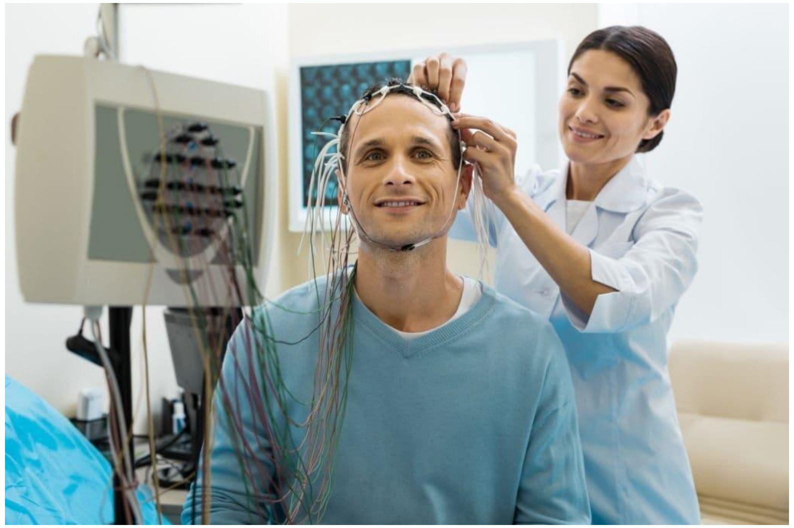
Project Pitch

Develop a brain-computer interface (BCI) using EEG to assist individuals with physical disabilities. Electrodes capture brain voltage fluctuations, a microcontroller processes signals, and algorithms translate sensory-motor activity into computer control, enabling hands-free device interaction for enhanced accessibility and independence.

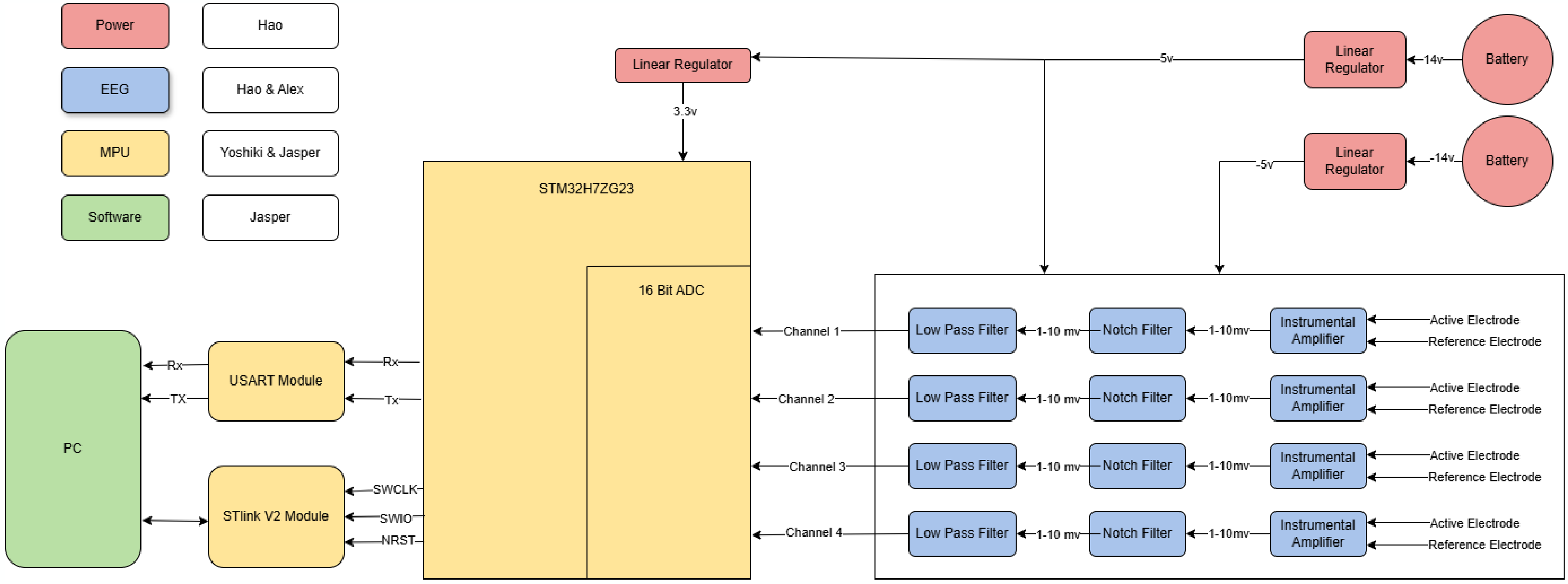


Users

- **Individuals with disabilities**: Individuals who are paralyzed, missing limbs, or have limited hand mobility face challenges in interacting with a computer.
- **For entertainment**: Individuals are looking for a new way to play games or interact with a computer.



System Diagram



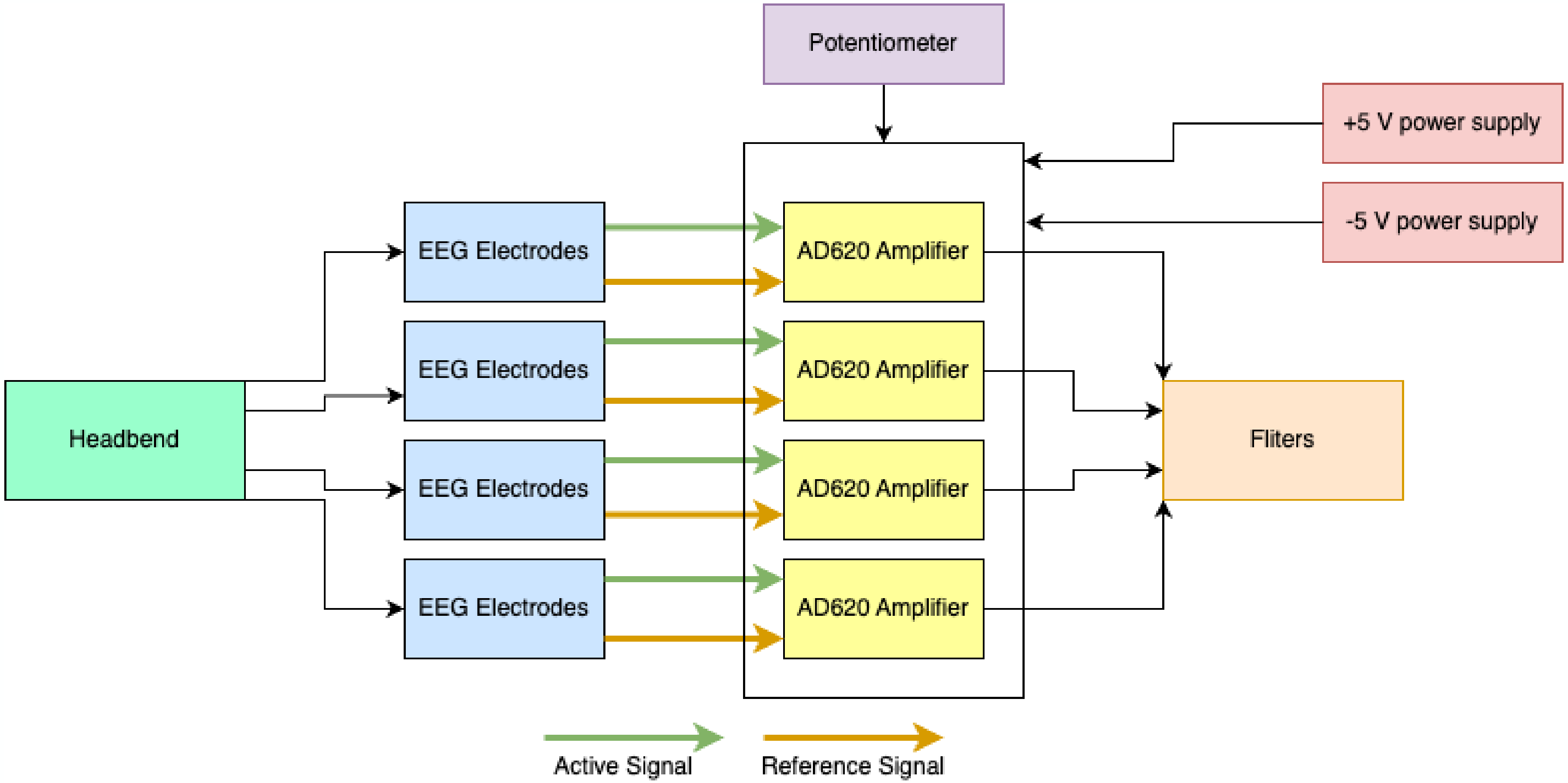
System Requirements

Requirements	Status
4 channels or working EEG circuits	
Delivery power to entire system via Batteries	
Send 4 channels of data to computer via USART	
Real time blink detection	
Real time mouse movement	
Aid disabled person communicate and gain functionality	

PURDUE UNIVERSITY

Headpiece and EEG Amplifier

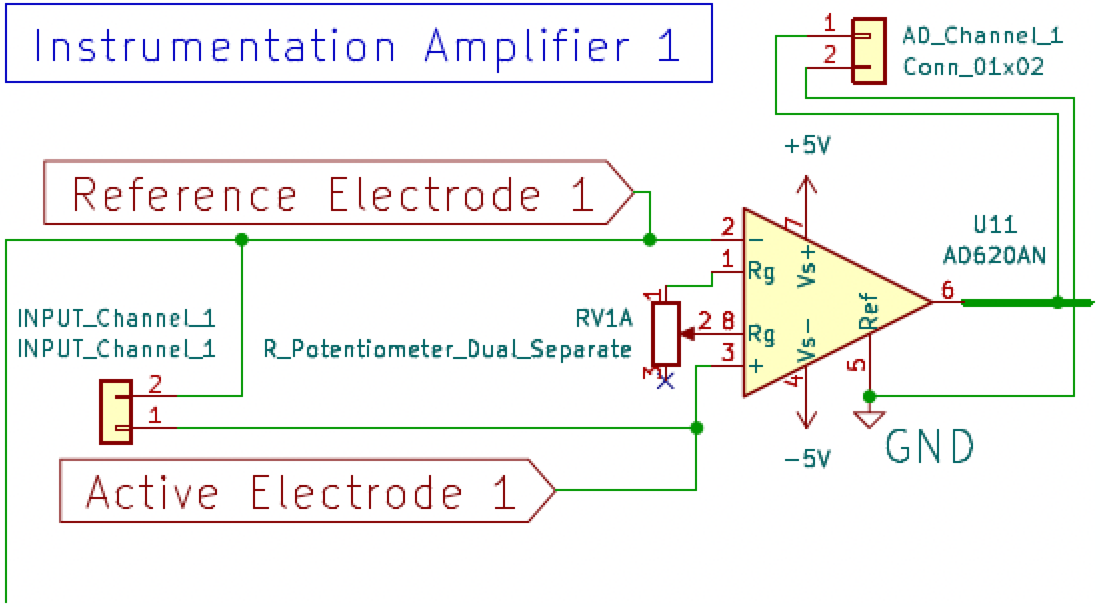
System Diagram

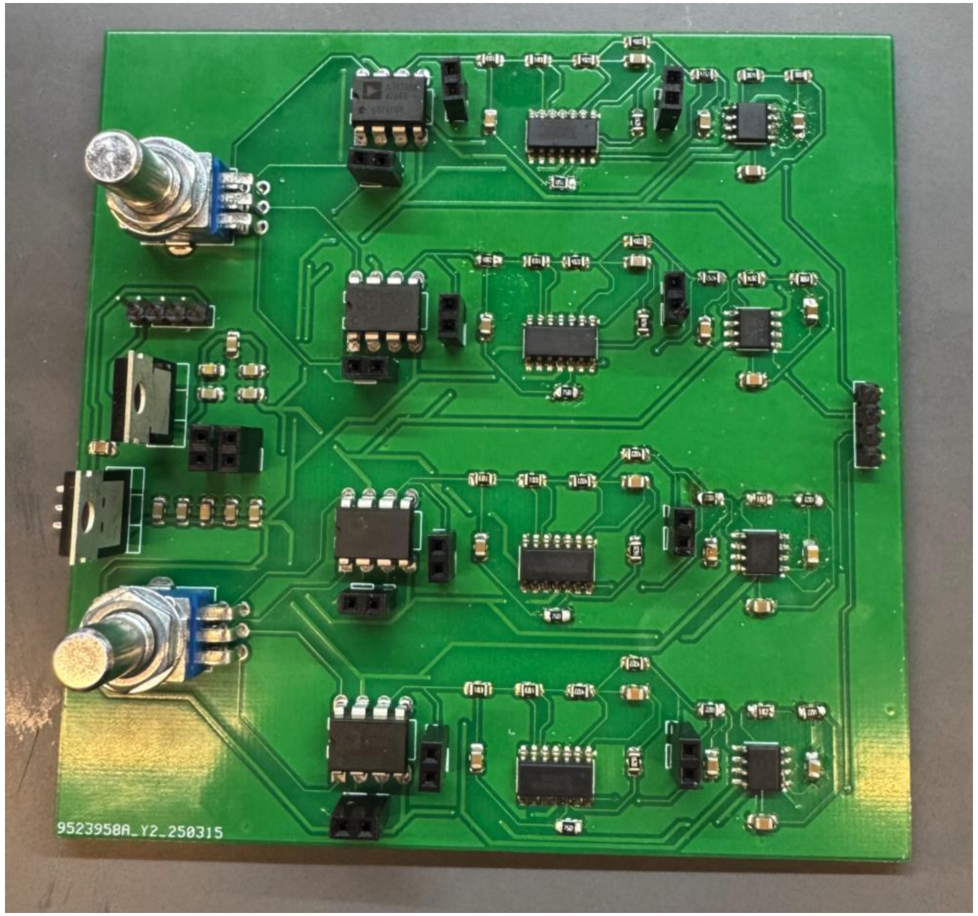


Headpiece and EEG Amplifier

Amplifier Circuit Design

- 2 potentiometer with max resistance of 10k
- 4 AD620AN instrumentation amplifier

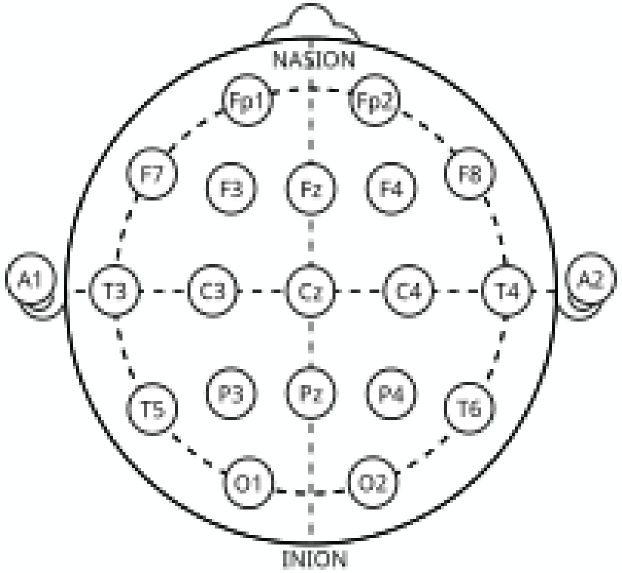


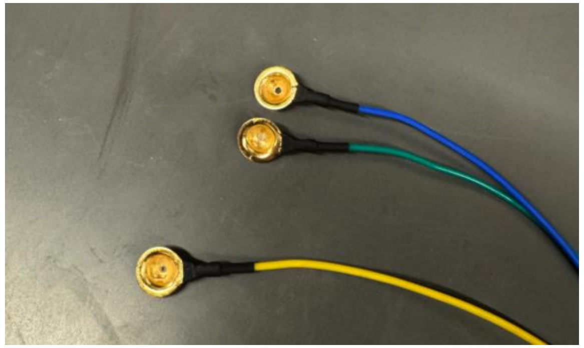


Headpiece and EEG Amplifier

Headpiece Design

- Wet Electrodes
  - Electrodes with Conductive paste
    - Better signal reception
- 4 Channel Electrodes
  - Four Active (Fp1, Fp2, O1, O2)
  - Reference (Earlobe)
  - Fp1 and Fp2 for Blink Detection
  - O1, O2 for SSVEP
- Headband
  - Sports headband
  - Ultra-low cost





Headpiece and EEG Amplifier

Requirements	Status
The headband securely position electrodes at O1, O2, Fp1, and Fp2 (10-20 system) and earlobe with consistent scalp contact	
Earlobe electrodes must provide a stable reference signal	
Active electrodes provide a stable active signal input	
Potentiometer can provide resistance between 10 to 10k Omega	
Each AD620AN amplifier amplify the differential signal (active vs. reference) with a gain of at least 20	

Headpiece and EEG Amplifier

Potentiometer value Justification

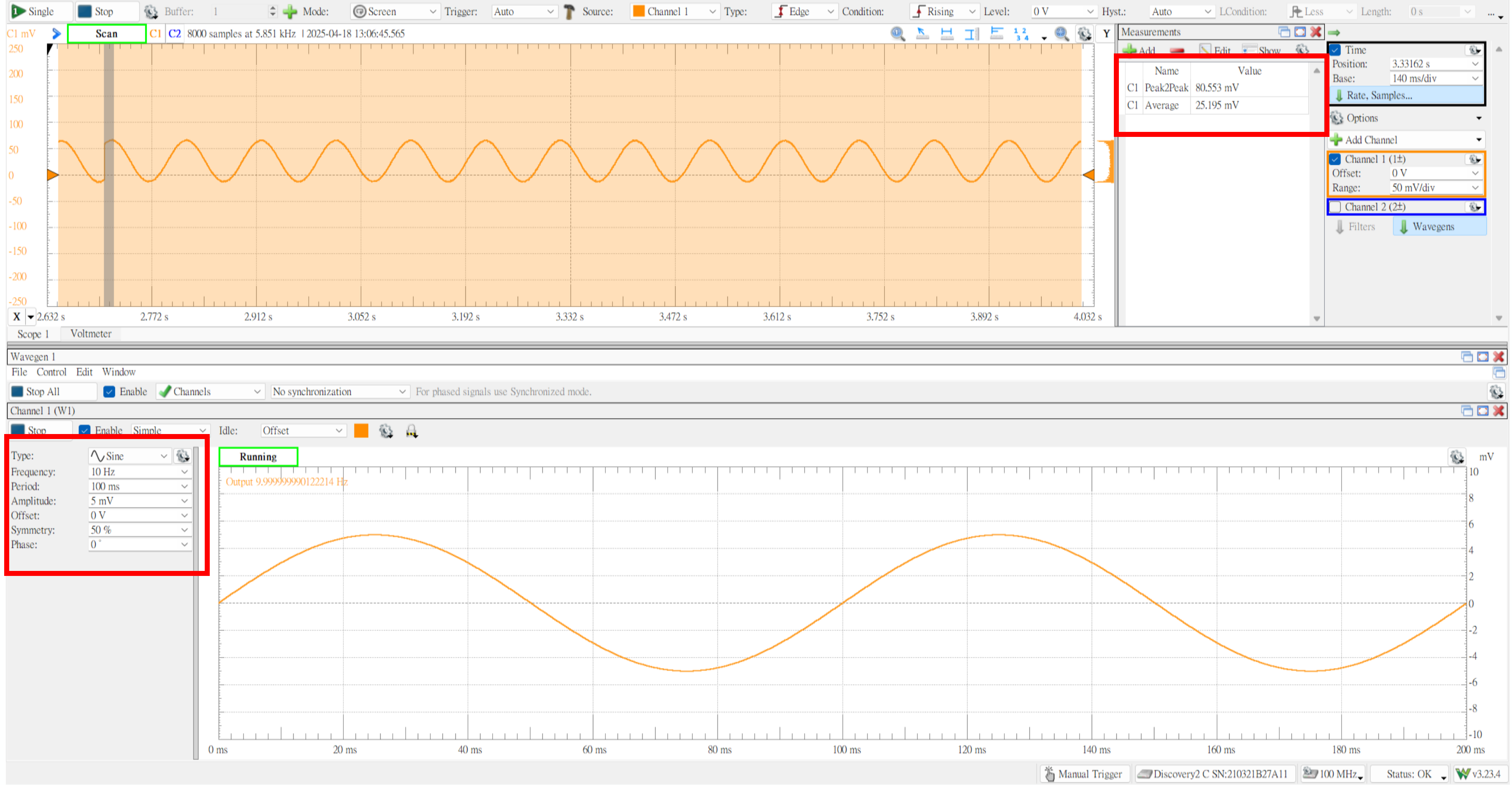
- Gain Calculation for Ad620AN: $$R_G = \frac{49.4k\Omega}{G-1}$$
- This Value Ensure amplifier have enough gain for EEG signal
- Max and min Resister Value of the Potentiometer

PURDUE UNIVERSITY

Headpiece and EEG Amplifier

Amplifier Gain Justification

- Most of the EEG input signal that we want is between 5Hz to 10Hz
- Amplifier output with Sine wave input @10Hz, 5mV with 2.94k Resistor Value
- Expected Gain: 15.8
- Actual Gain: 16.1

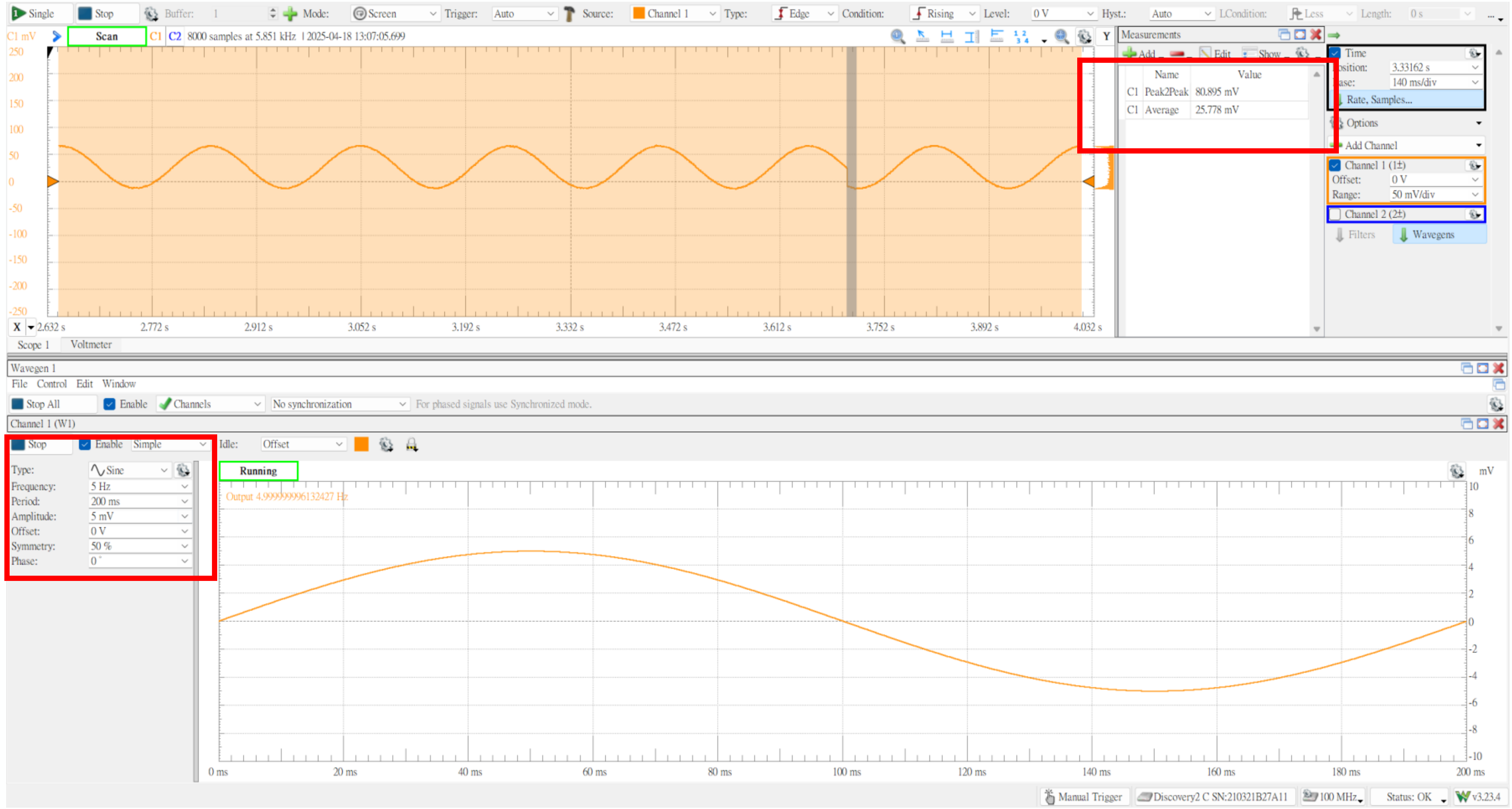


PURDUE UNIVERSITY

Headpiece and EEG Amplifier

Amplifier Gain Justification

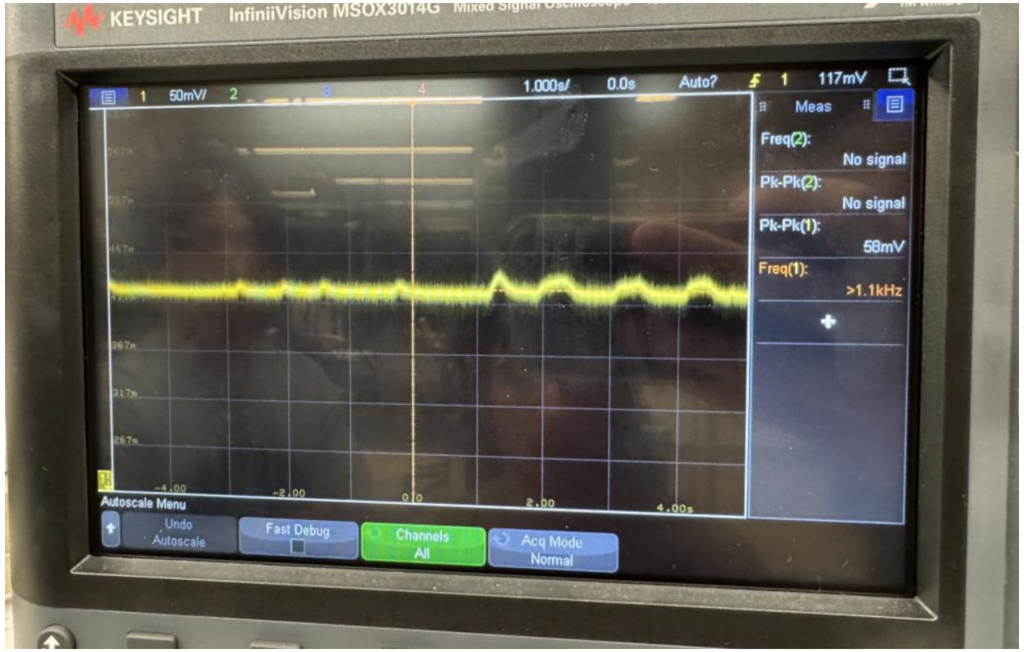
- Amplifier output with Sine wave input @5Hz, 5mV
- Expected Gain: 15.8
- Actual Gain: 16.2



Headpiece and EEG Amplifier

Electrode Input Justification

- Input waveform straight from electrodes
    - One reference and one active

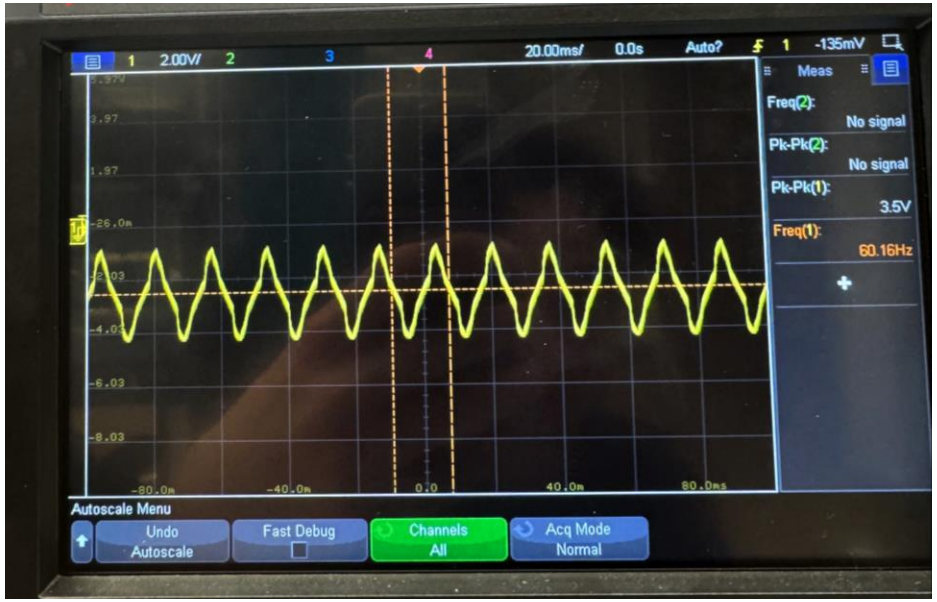


PURDUE UNIVERSITY

Headpiece and EEG Amplifier

System Intergration

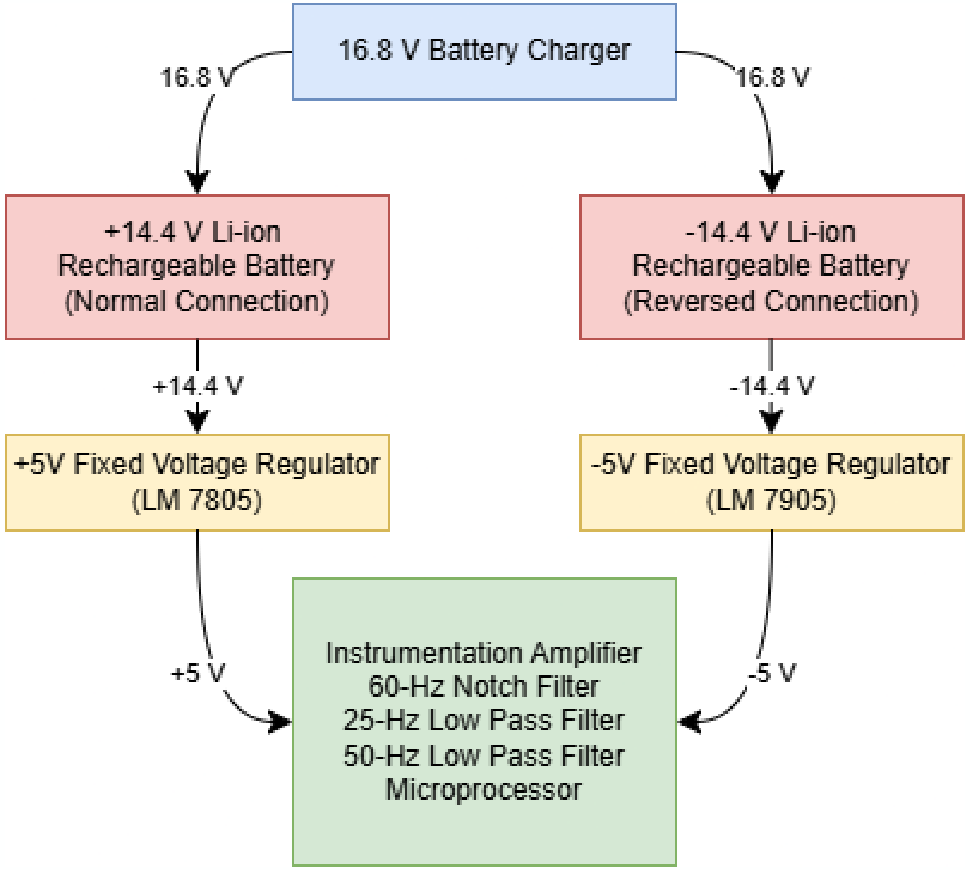
- Output with amplifier compare with Raw data input from electrodes
- Amplifier have proper gain on EEG signals
- Expected Gain: 19.1
- Actual Gain: 20.7



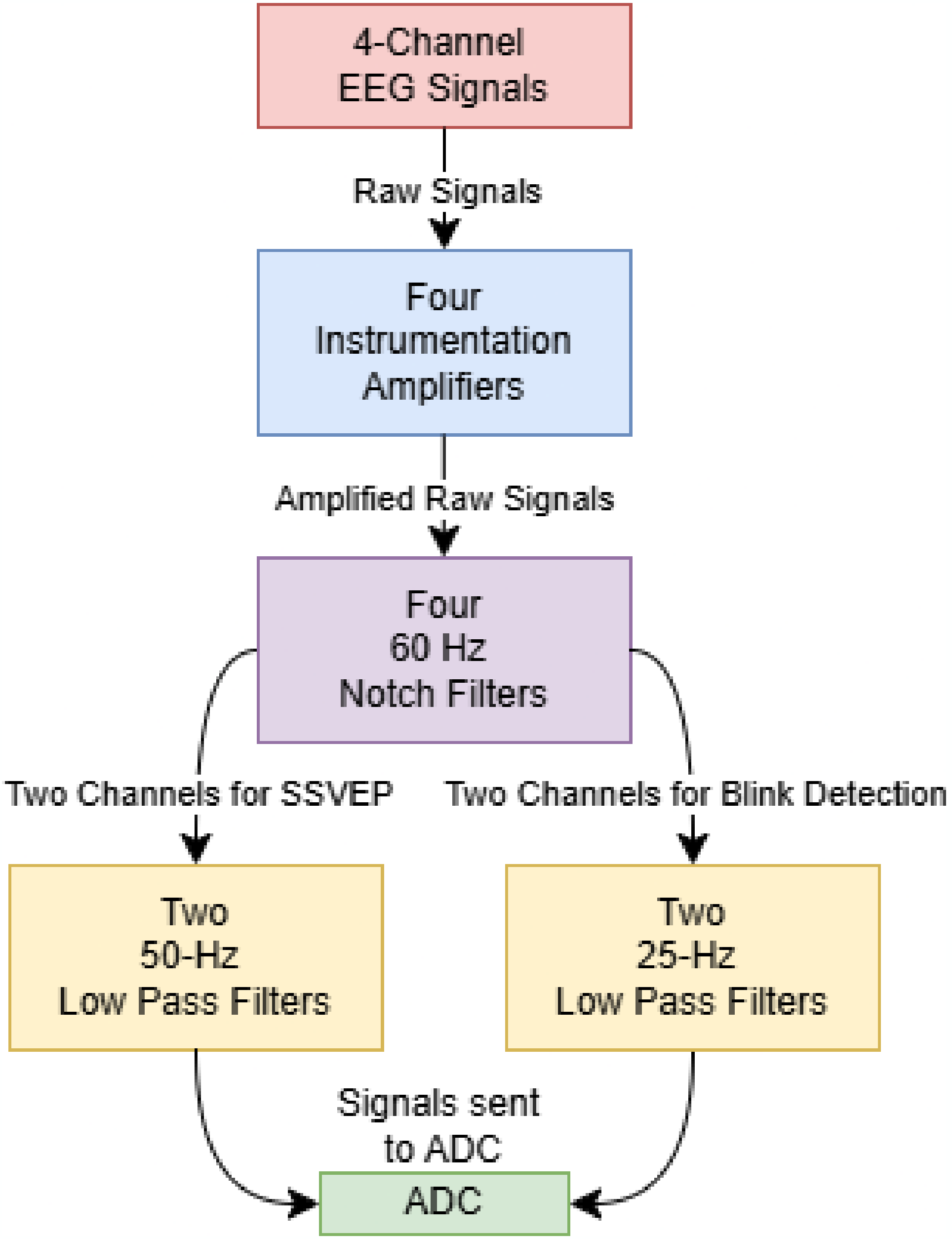
PURDUE UNIVERSITY®

Filters and Power

System design for power and filters



Power Subsystem Block Diagram

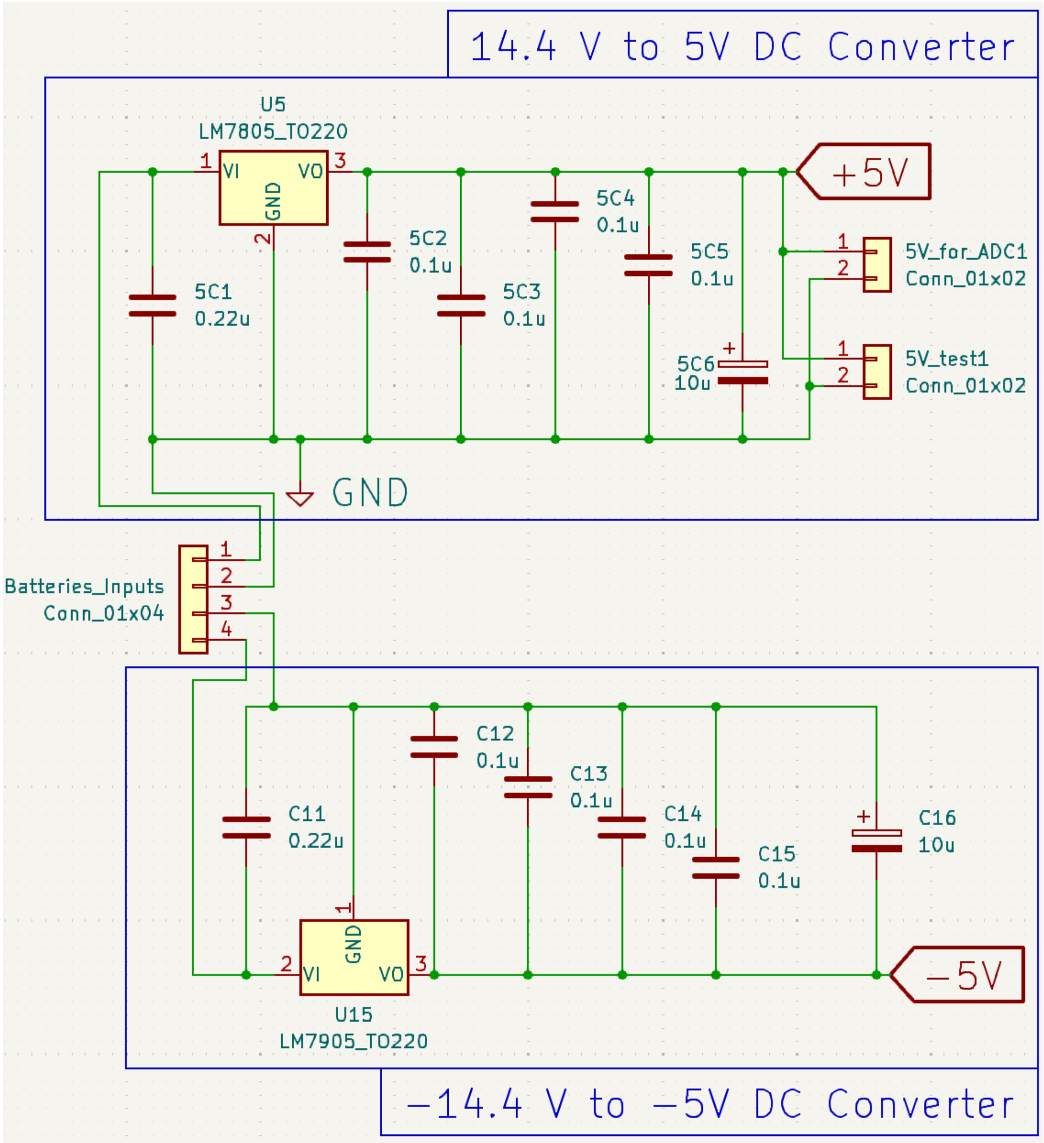


Filters Subsystem Block Diagram

PURDUE
UNIVERSITY®

Filters and Power

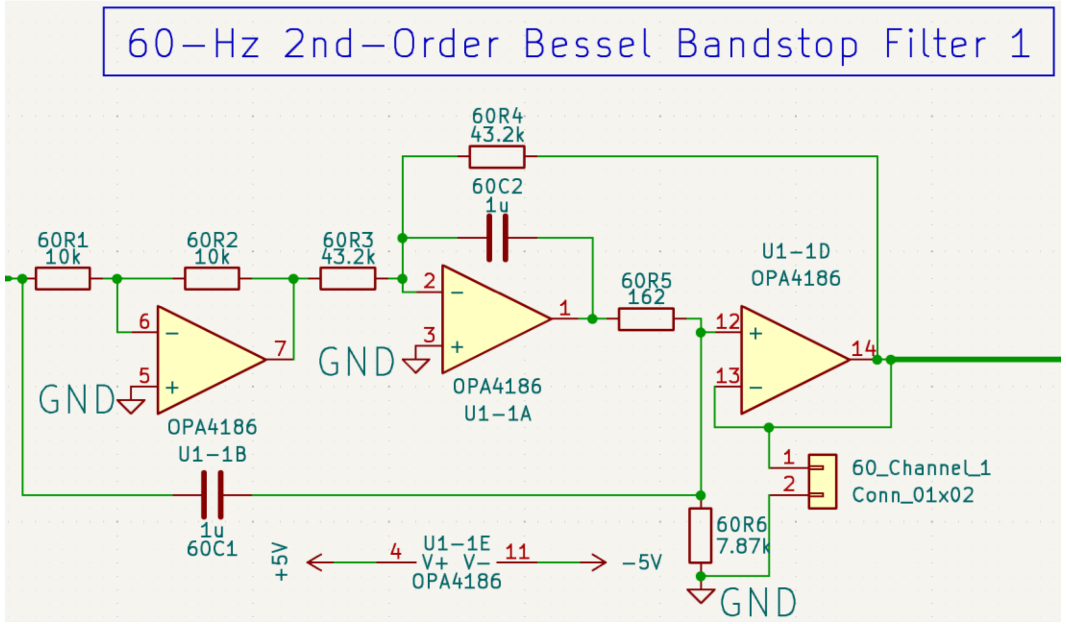
Schematic design for Power



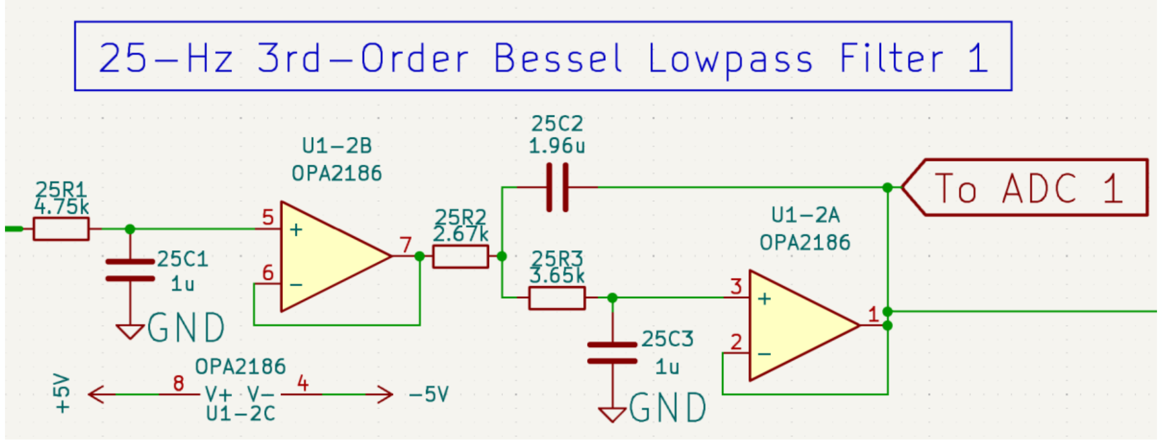
Power module with decoupling capacitors

Filters and Power

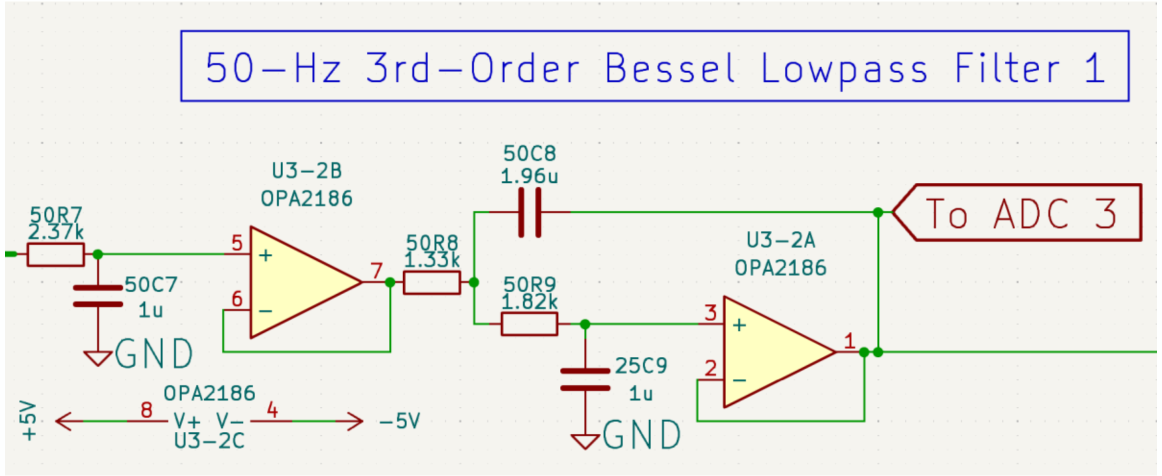
Schematic design for Filters



60-Hz Notch Filters



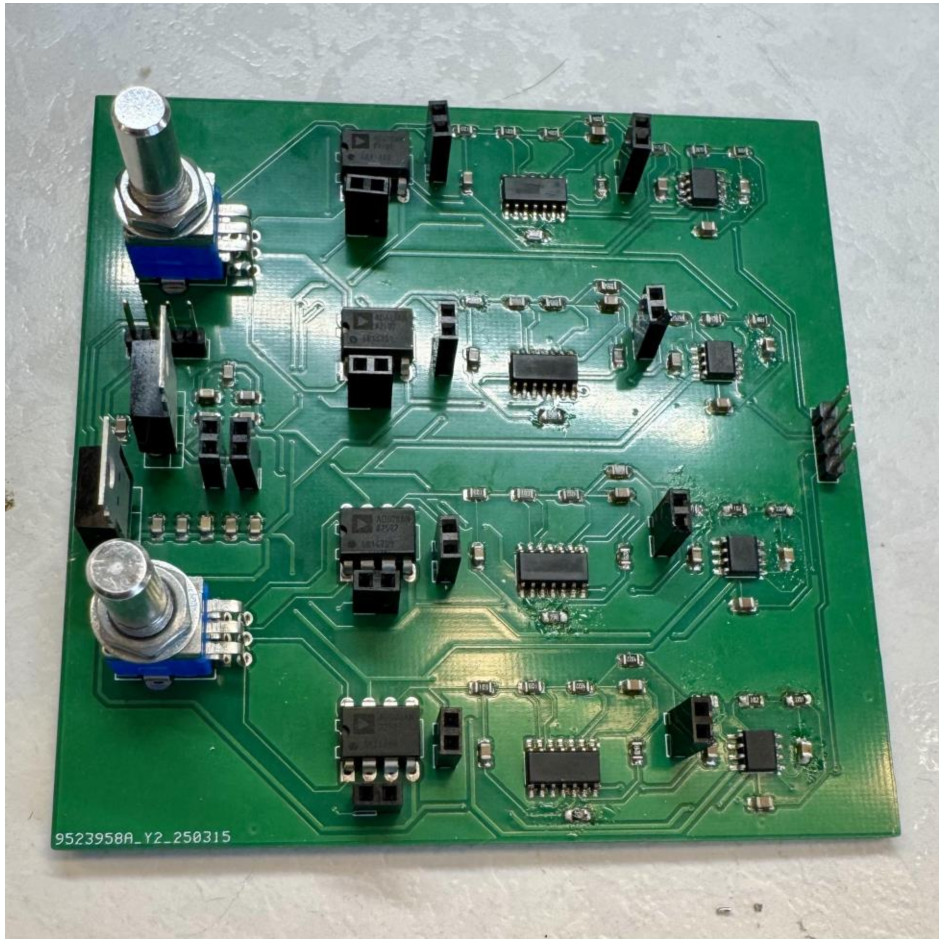
25-Hz Low Pass Filter



50-Hz Low Pass Filter

Filters and Power

PCB Design for Power and Filters



Filters and Power

Subsystem Functionality for Power

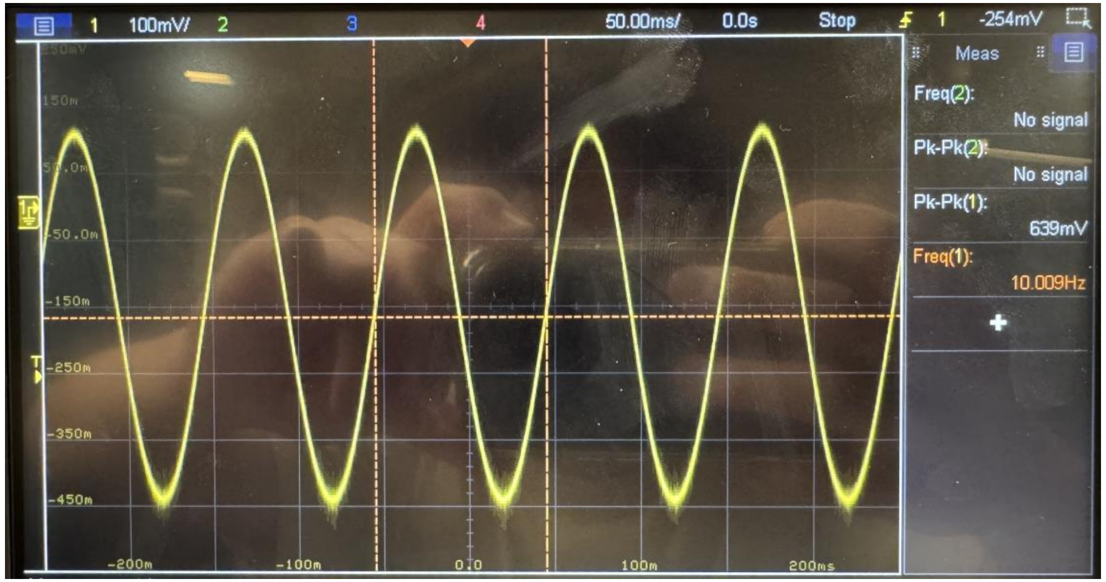
Input Voltages for Channel Four Instrumentation Amplifier

Input Voltages for Channel Four OPA4186DR (Notch Filter IC)

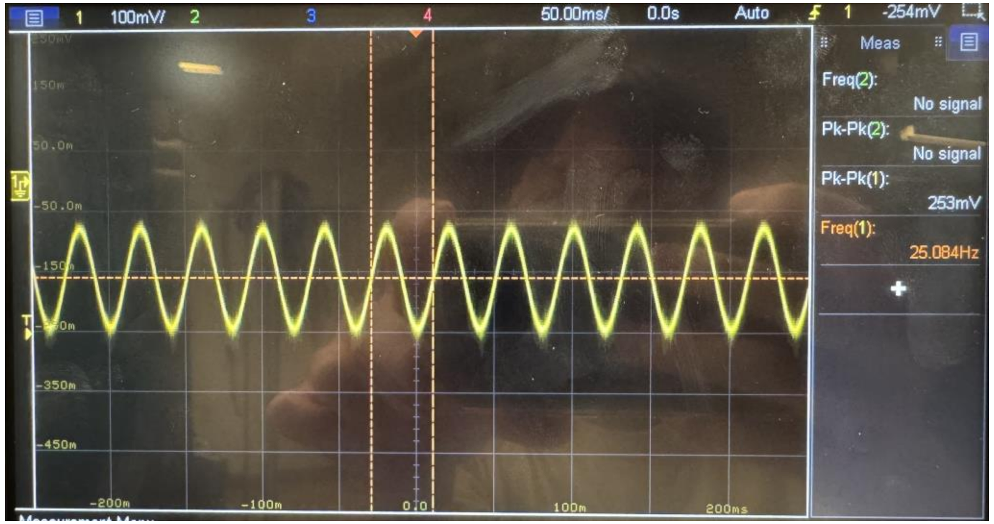
Input Voltages for Channel Four OPA2186DR (50-Hz Low Pass Filter IC)

Filters and Power

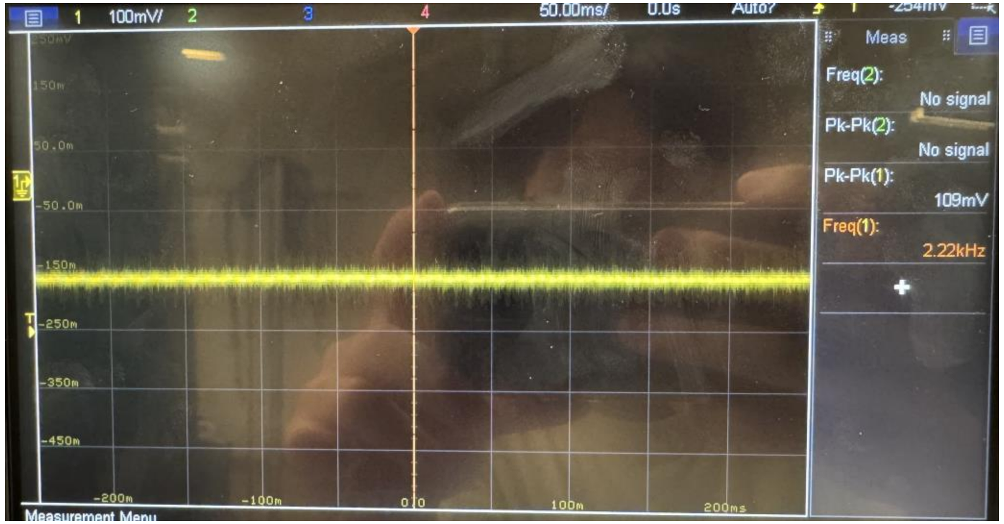
Subsystem Functionality for Filters – 1st and 2nd Channel



Input: 10Hz & 5mVpp
Output: 10Hz with 639mVpp



Input: 25Hz & 5mVpp
Output: 25Hz with 253mVpp

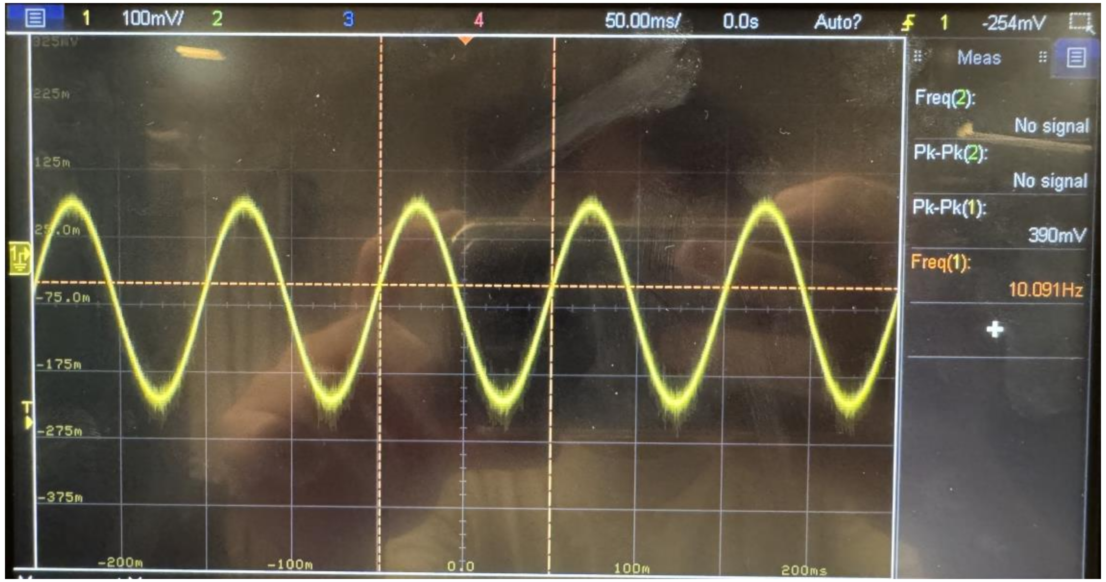


Input: 60Hz & 5mVpp
Output: 60Hz with 109mVpp

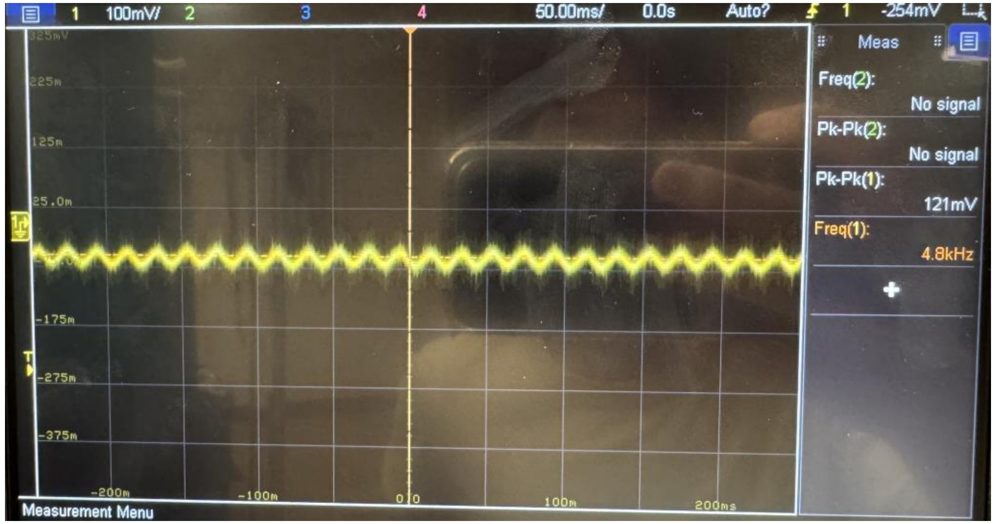
PURDUE
UNIVERSITY

Filters and Power

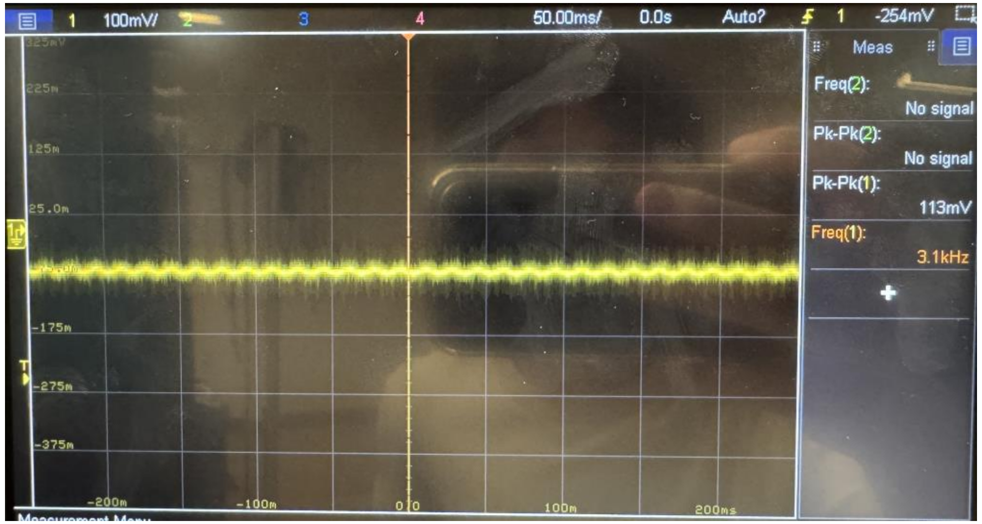
Subsystem Functionality for Filters – 3rd and 4th Channel



Input: 10Hz & 5mVpp
Output: 60Hz with 390mVpp



Input: 50Hz & 5mVpp
Output: 50Hz with 121mVpp



Input: 60Hz & 5mVpp
Output: 60Hz with 113mVpp

PURDUE
UNIVERSITY®

Filters and Power

Specification Relevance and Justification

Subsystem	Specification	Value	Justification
Power	Battery Voltage	>9V under load	Ensure LM7805 & LM7905 operations
Power	LM7805 Output Voltage	5V, ±5%	Powers 5V components
Power	LM7905 Output Voltage	-5V, ±5%	Supports dual-supply op-amps
Power	Efficiency	>60% for both linear regulators	Reduces power loss
Filter	60-Hz Notch filter	Center at 60 Hz, ±10 Hz	Rejects power line noise
Filter	25-Hz Low Pass Filters	-3 dB point at 25 Hz on FRA	Pass low frequency signals
Filter	50-Hz Low Pass Filters	-3 dB point at 50 Hz on FRA	Pass low frequency signals

Filters and Power

Specification Evidence for Power System

Battery One Voltage

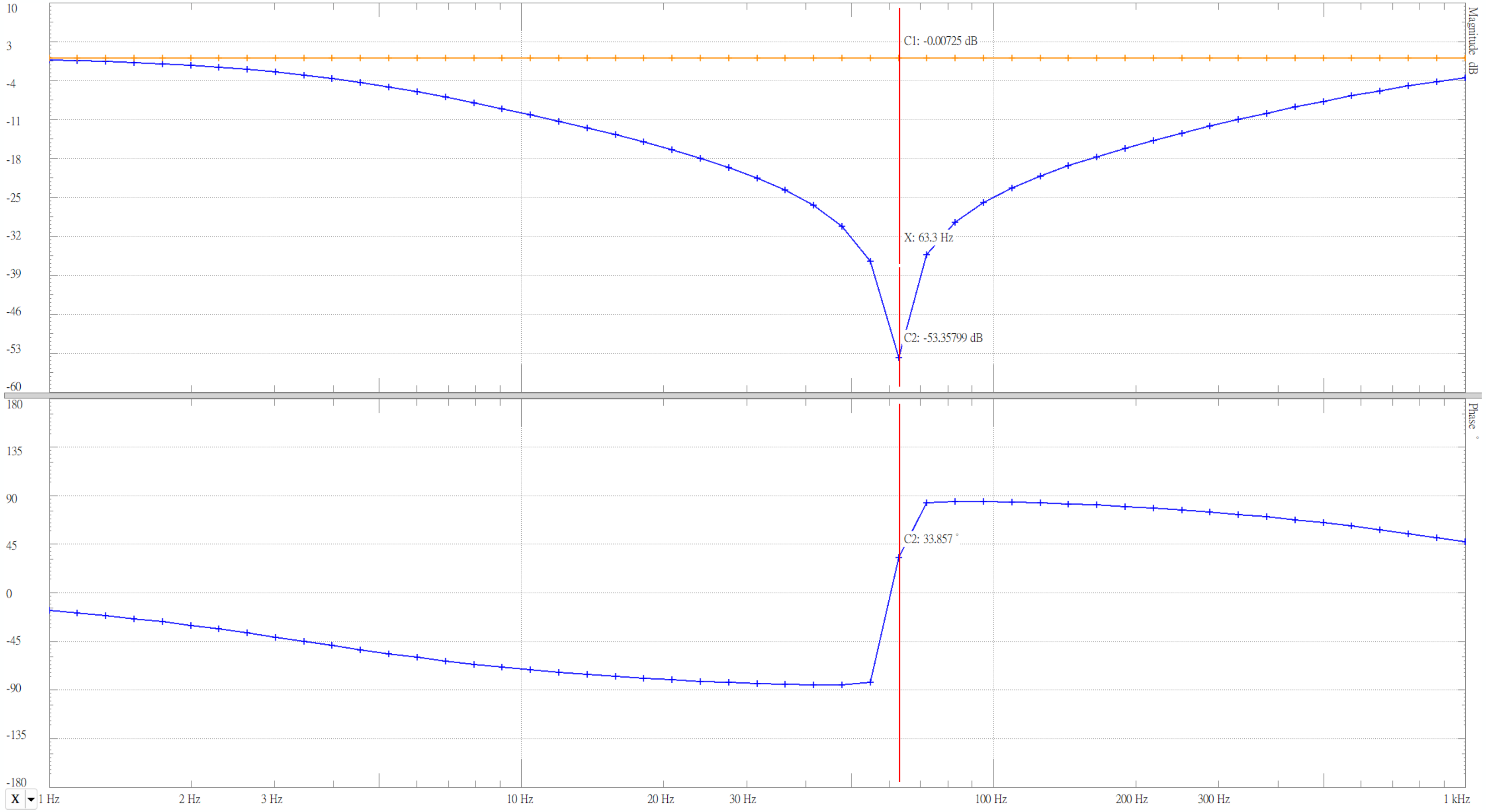
Battery Two Voltage

LM 7805 Linear Regulator Output

LM 7905 Linear Regulator Output

Filters and Power

Specification Evidence for 60-Hz Notch Filters – Frequency Response Analysis

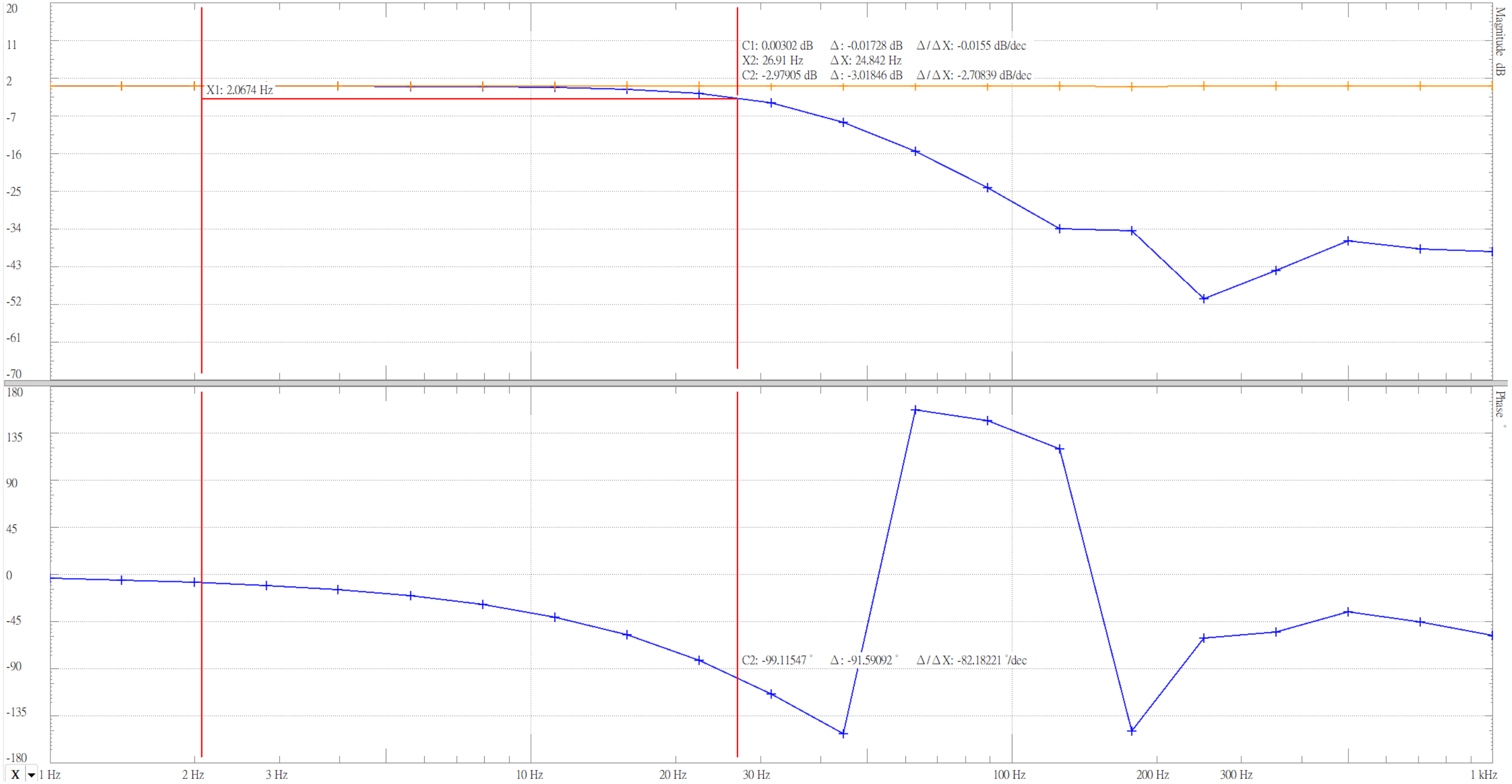


63.3 Hz

PURDUE UNIVERSITY

Filters and Power

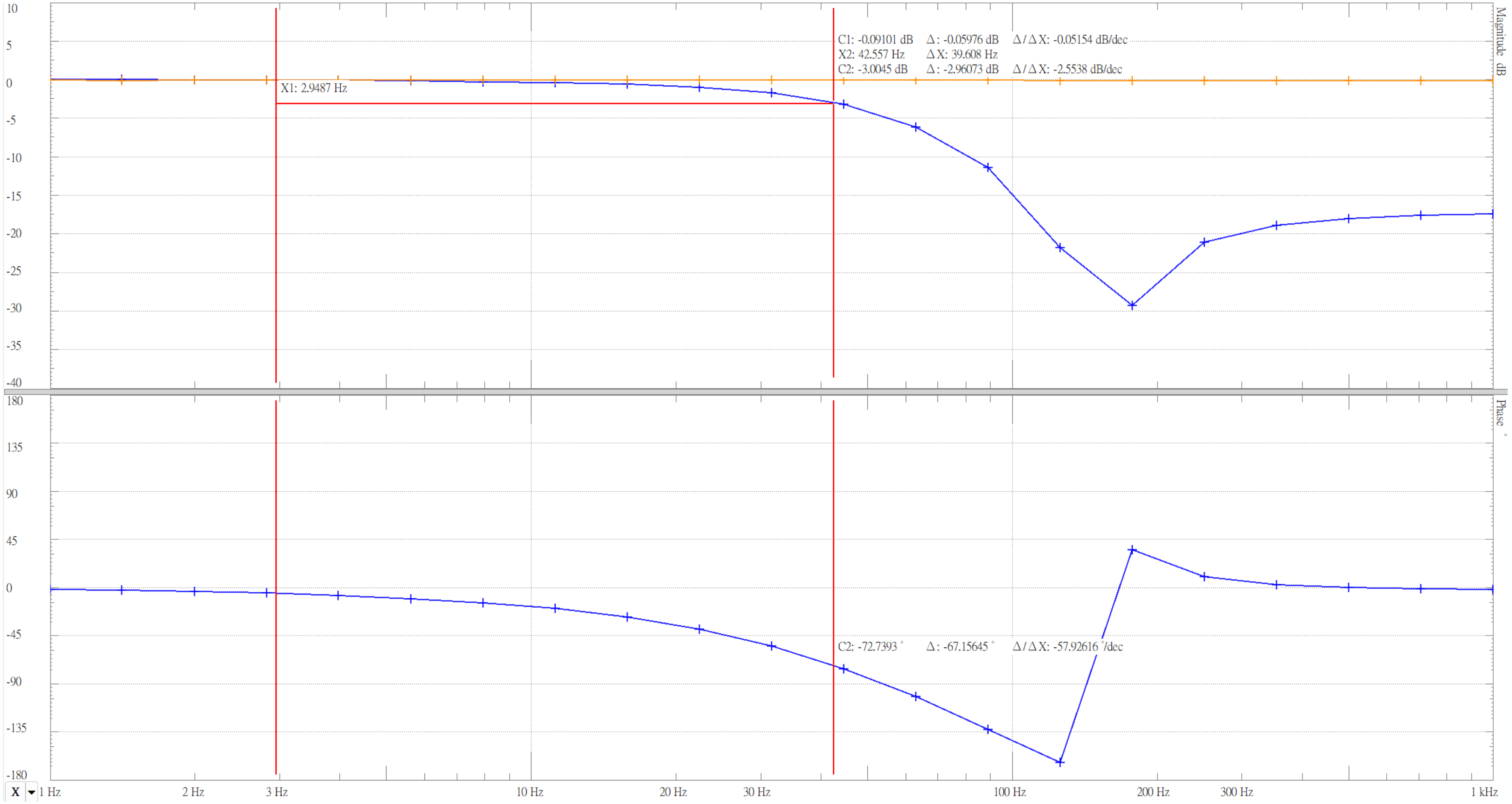
Specification Evidence for 25-Hz Low Pass Filters – Frequency Response Analysis



26.91 Hz

Filters and Power

Specification Evidence for 50-Hz Low Pass Filters – Frequency Response Analysis



42.56 Hz

PURDUE UNIVERSITY

Filters and Power

Specification Evidence for both power and filter

Subsystem	Specification	Value	Met/ Not met; was completed/ Not completed
Power	Battery Voltage	15.3230V & 15.2861V	Met
Power	LM7805 Output Voltage	4.9658V	Met
Power	LM7905 Output Voltage	-5.02731V	Met
Filter	60-Hz Notch filter	63.3 Hz	Met
Filter	25-Hz Low Pass Filters	-3 dB point at 26.91 Hz	Not met; was completed
Filter	50-Hz Low Pass Filters	-3 dB point at 42.56 Hz	Not met; was completed

PURDUE UNIVERSITY

Filters and Power

Specification Evidence for both power and filter

Subsystem	Specification	Value	Met/ Not met; was completed/ Not completed
Power	Efficiency	32.4% and 32.89%	Not met; was completed

$$P_{Out} = V_{Out} \cdot I_{Out} = 4.9658 \cdot 0.010824 = 0.05375\ W$$

$$P_{In} = V_{In} \cdot I_{In} = 15.3230 \cdot 0.010824 = 0.16586\ W$$

$$\eta = \frac{P_{Out}}{P_{In}} \times 100\% = \frac{0.05375}{0.16586} \times 100\% \approx 32.40\%$$

Efficiency for LM 7805

$$P_{Out} = |V_{Out}| \cdot I_{Out} = 5.02731 \cdot 0.011056 = 0.05558\ W$$

$$P_{In} = V_{In} \cdot I_{In} = 15.2861 \cdot 0.011056 = 0.16900\ W$$

$$\eta = \frac{P_{Out}}{P_{In}} \times 100\% = \frac{0.05558}{0.16900} \times 100\% \approx 32.89\%$$

Efficiency for LM 7905

PURDUE UNIVERSITY

Filters and Power

Progress and Evidence Clarity

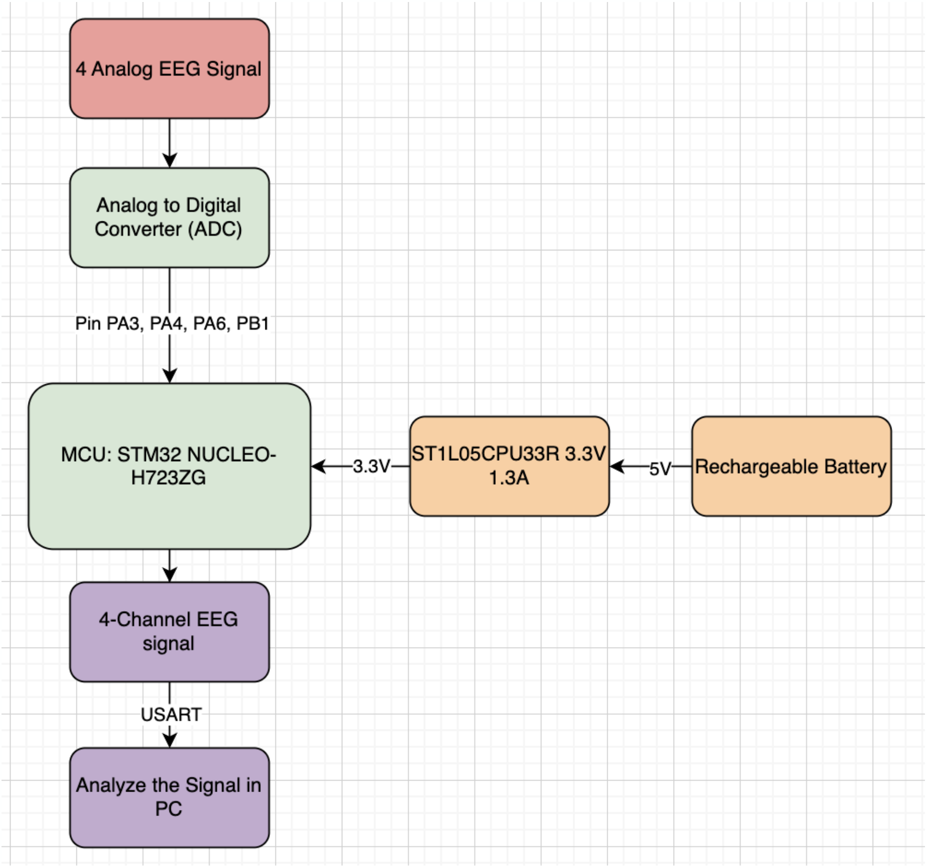
- Power subsystem: Complete
- Filter subsystem: Functional
- Achievement: Stable power supply despite high 15.3V input.
- Challenge: Tuned 25 Hz and 50 Hz filter to achieve 26.91 Hz and 42.56 Hz due to component tolerances (E96 and E12).

- All plots (oscilloscope, FRA) and photos are high-resolution, labeled

Microprocessor

Subsystem design

- 5V from the battery and in the PCB it converts to 3.3V
- Analog signal from EEG circuit
- Analog to Digital Converter
- Send signal to PC using USART
- Analyze singal in the PC



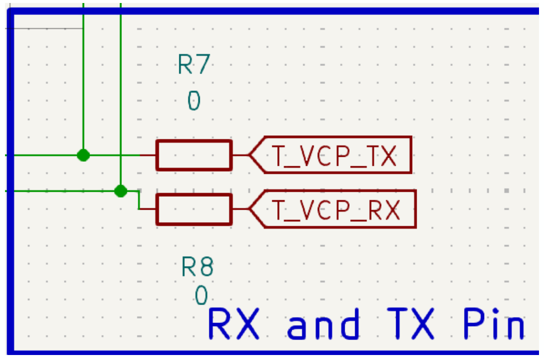
PURDUE UNIVERSITY

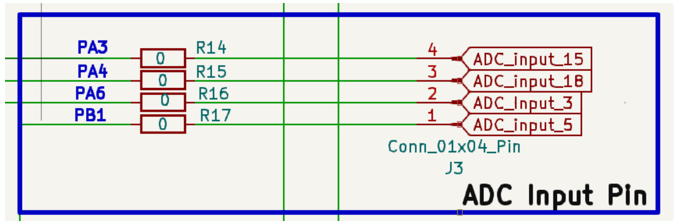
Microprocessor

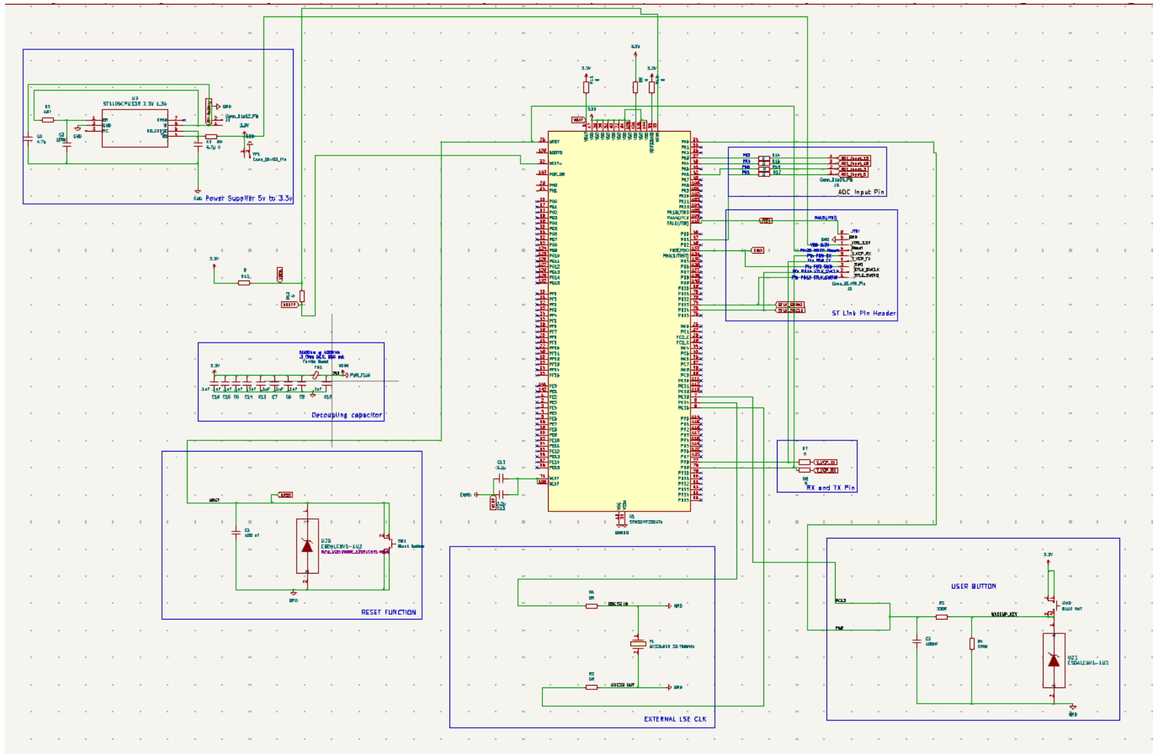
Requirements	Status
The device features a 4-channel Analog-to-Digital Converter (ADC) for precise signal processing and data acquisition.	
The device is powered by a battery, ensuring portable and reliable operation.	
Using our own designed PCB	
The device is equipped with a 16-bit microcontroller for high-precision processing and performance.	
The device's microcontroller unit (MCU) supports a sampling rate of 250 samples per second for accurate and efficient data acquisition.	

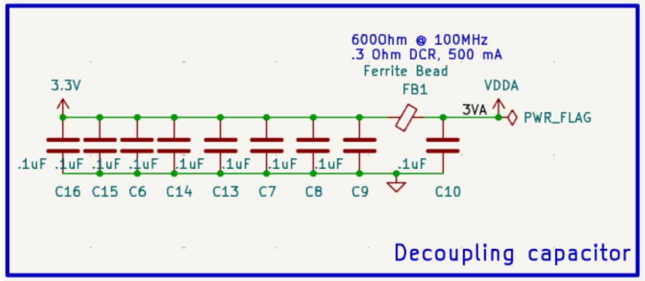
Microprocessor

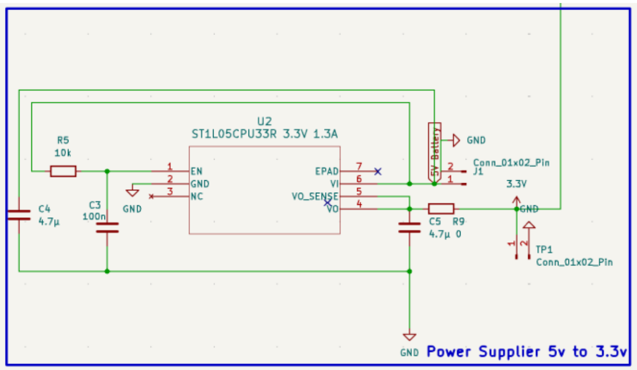
Schematic design

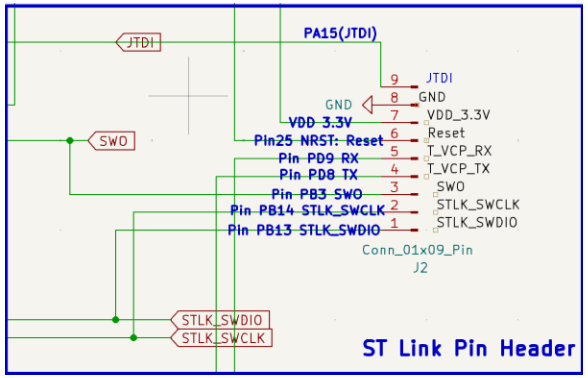












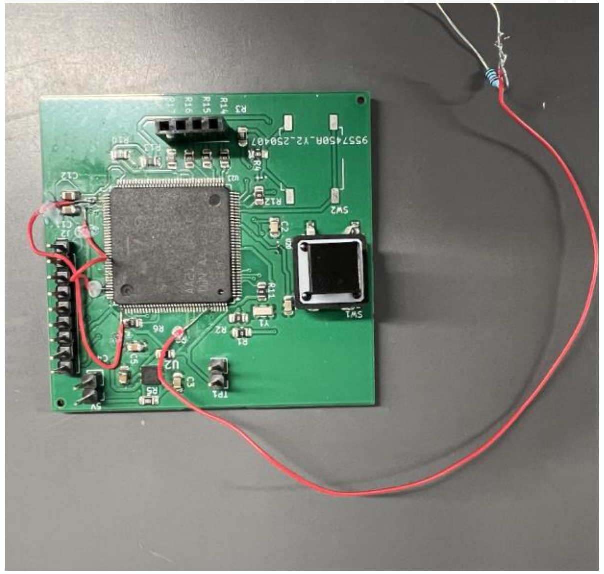
PURDUE
UNIVERSITY®

Microprocessor

Subsystem Functionality

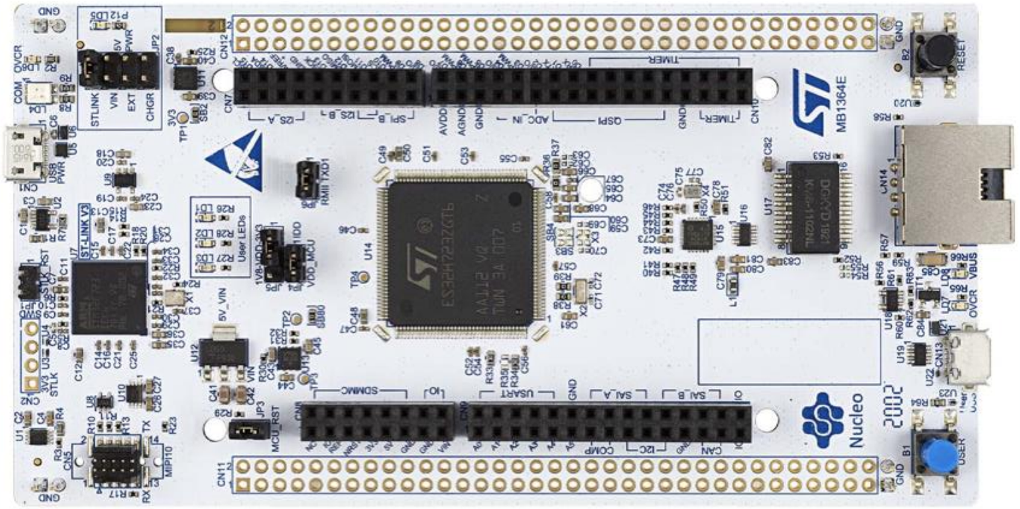
After design our own PCB, we found some issues.

- Wrong pin for STLK_Clock, SW_DIO
    - We resolder with the correct pin
- Not connecting BOOT0 to GND
    - We resolder a 10k pull down resistor with GND
- ST-Link V2 could not detect in our STM32 IDE Cube system.
    - After debugging and ask Professor Niraj for help it still not working.



Solution

- Using the original microprocessor which is STM32H723ZG.

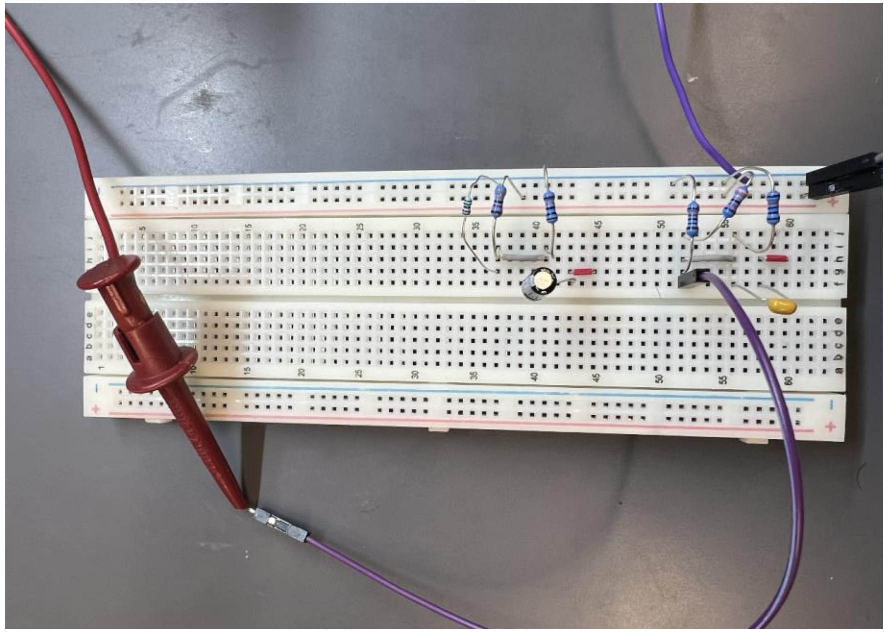


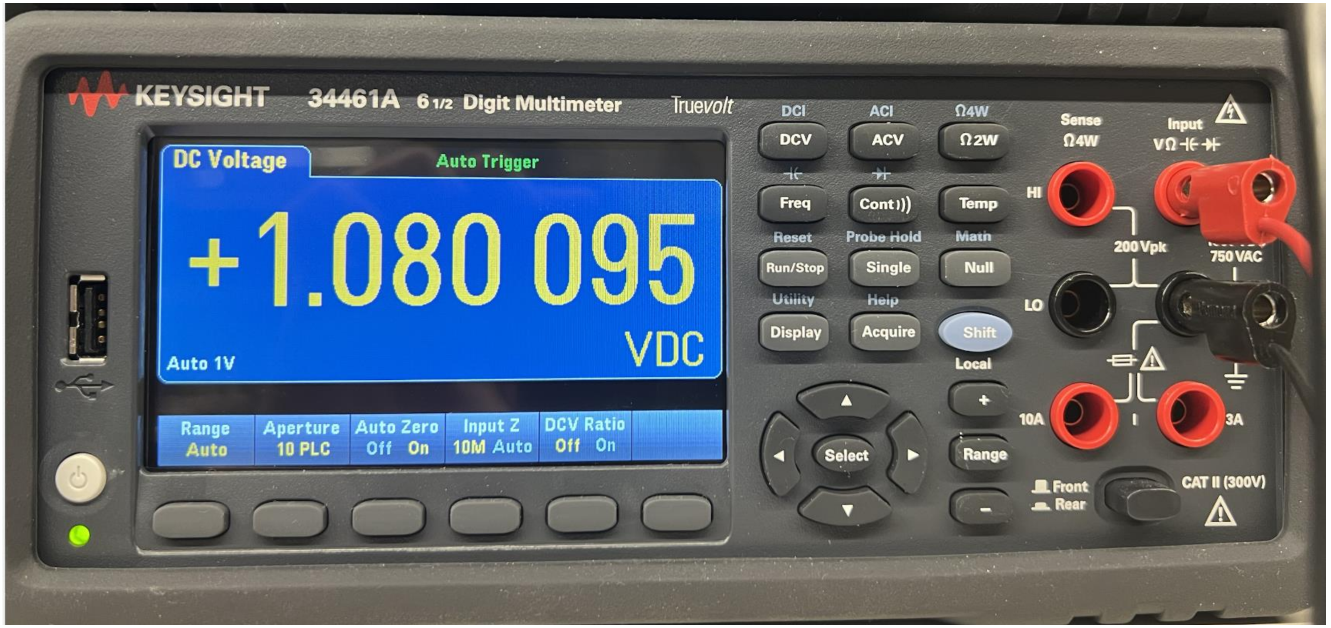
Microprocessor

Subsystem Functionality

Issue: We identified an issue with a negative DC offset in the PCB's output signal, which is incompatible with the microcontroller's requirements, as it cannot process negative voltages.

Solution: Implemented a high-pass filter and a voltage divider, successfully shifting the DC offset to a positive 1V, aligning with the microcontroller's operational specifications.





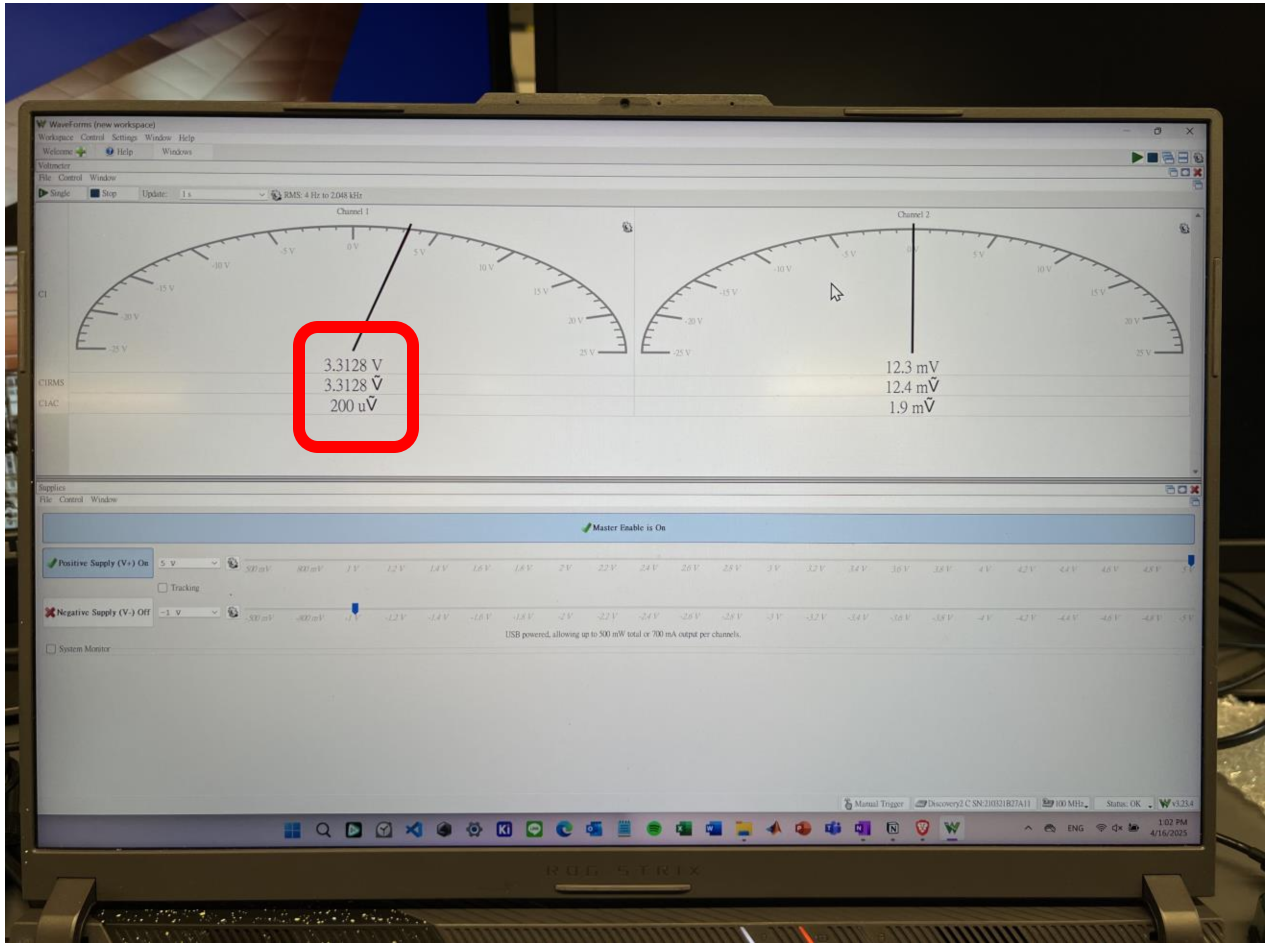
PURDUE UNIVERSITY

Microprocessor

Specification Relevance and Justification

5V to 3.3V

- We have a stable 3.3V for our microprocessor using **ST1L05CPU33R** 3.3V.



Microprocessor

Specification Relevance and Justification

Sampling Rate

- To show our ADC's Sampling rate we samples 4 channels in the for loop and sends those channels of data via USART.
- Results
  - 250 samples/sec for each ADC

```
for (int i = 0; i < 4; i++) {
        HAL_ADC_Start(&hadc1);
        HAL_ADC_PollForConversion(&hadc1, HAL_MAX_DELAY);
        adcValues[i] = HAL_ADC_GetValue(&hadc1);
    }

snprintf(buffer, sizeof(buffer), "%u,%u,%u,%u\r\n",
            adcValues[0], adcValues[1], adcValues[2], adcValues[3]);
HAL_UART_Transmit(&huart3, (uint8_t*)buffer, strlen(buffer), 10);

if (HAL_GetTick() - startTick >= 1000) {
    sprintf(samplingMsg, "Sampling Rate: %u samples/sec\n", adcConversionCount);
    HAL_StatusTypeDef status = HAL_UART_Transmit(&huart3, (uint8_t*)samplingMsg, strlen(samplingMsg), 10);
    adcConversionCount = 0;
    startTick = HAL_GetTick();
}
```

```
PS C:\Users\jaspe\BCI_Senior_Design> python .\src\data_input\test_in.py
Sampling Rate: 250 samples/sec
Sampling Rate: 250 samples/sec
Sampling Rate: 250 samples/sec
Sampling Rate: 250 samples/sec
Sampling Rate: 250 samples/sec
Sampling Rate: 250 samples/sec
```

Microprocessor

Specification Relevance and Justification

Bit Rate

- The microcontroller operates with a reference voltage (Vref) of 3.3V and features a 16-bit analog-to-digital converter (ADC). This configuration provides a resolution of 2^16, or 65,536 levels, for converting analog signals to digital values.

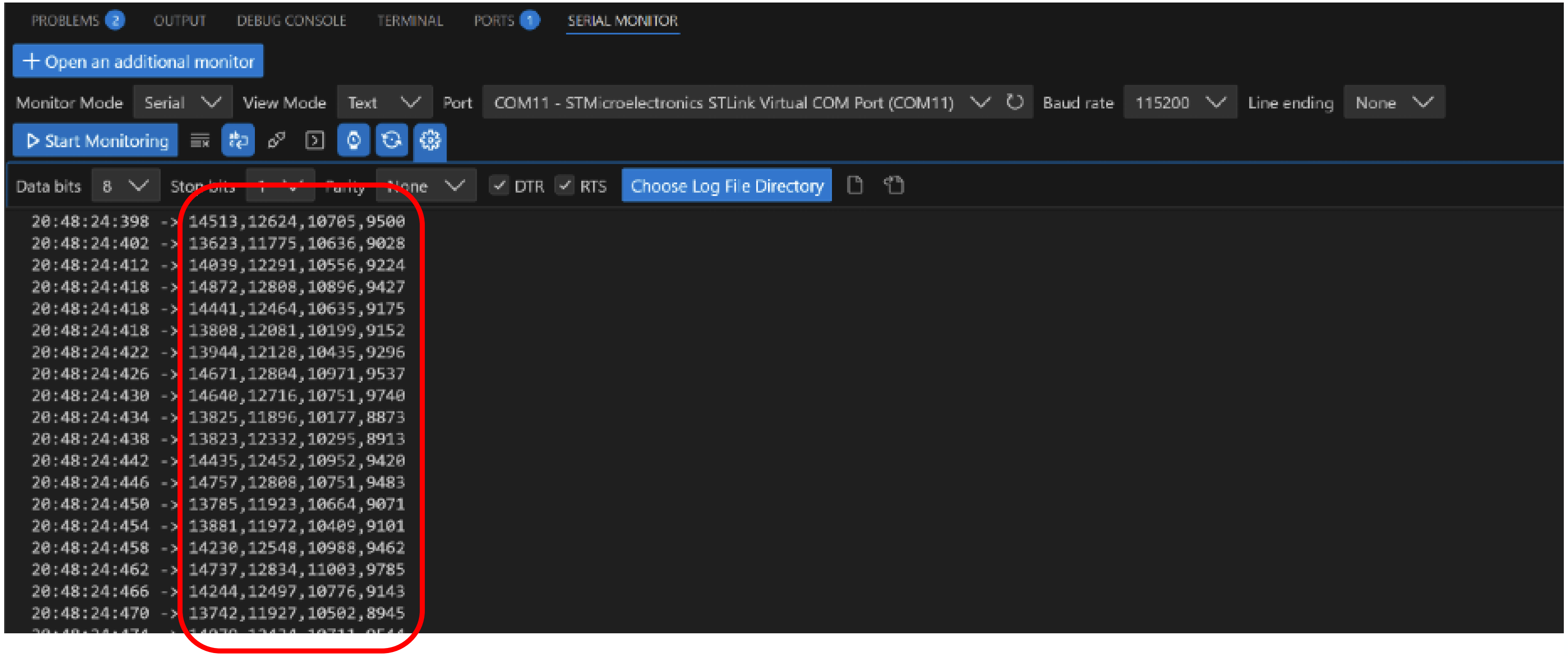
```
PS C:\Users\jaspe\BCI_Senior_Design> python .\test_bitrate.py
ADC Value for 4 Channels: 65536, 65536, 65536, 65536
Bit-Rate for 4 Channels: 16.000, 16.000, 16.000, 16.000

ADC Value for 4 Channels: 65536, 65536, 65536, 65536
Bit-Rate for 4 Channels: 16.000, 16.000, 16.000, 16.000

ADC Value for 4 Channels: 65536, 65536, 65536, 65529
Bit-Rate for 4 Channels: 16.000, 16.000, 16.000, 16.000

ADC Value for 4 Channels: 65237, 65536, 65536, 65536
Bit-Rate for 4 Channels: 15.993, 16.000, 16.000, 16.000

ADC Value for 4 Channels: 65536, 65536, 65005, 65536
Bit-Rate for 4 Channels: 16.000, 16.000, 15.988, 16.000

ADC Value for 4 Channels: 65536, 65536, 65536, 65141
Bit-Rate for 4 Channels: 16.000, 16.000, 16.000, 15.991

ADC Value for 4 Channels: 65536, 65121, 65536, 65536
Bit-Rate for 4 Channels: 16.000, 15.991, 16.000, 16.000

ADC Value for 4 Channels: 65158, 65536, 65536, 65536
Bit-Rate for 4 Channels: 15.992, 16.000, 16.000, 16.000

ADC Value for 4 Channels: 65536, 65536, 65536, 65093
Bit-Rate for 4 Channels: 16.000, 16.000, 16.000, 15.990
```

Microprocessor

Specification Relevance and Justification

4-Channel ADC via Rx & Tx

- USART send 4 data channels via Rx and Tx from the MCU to the PC



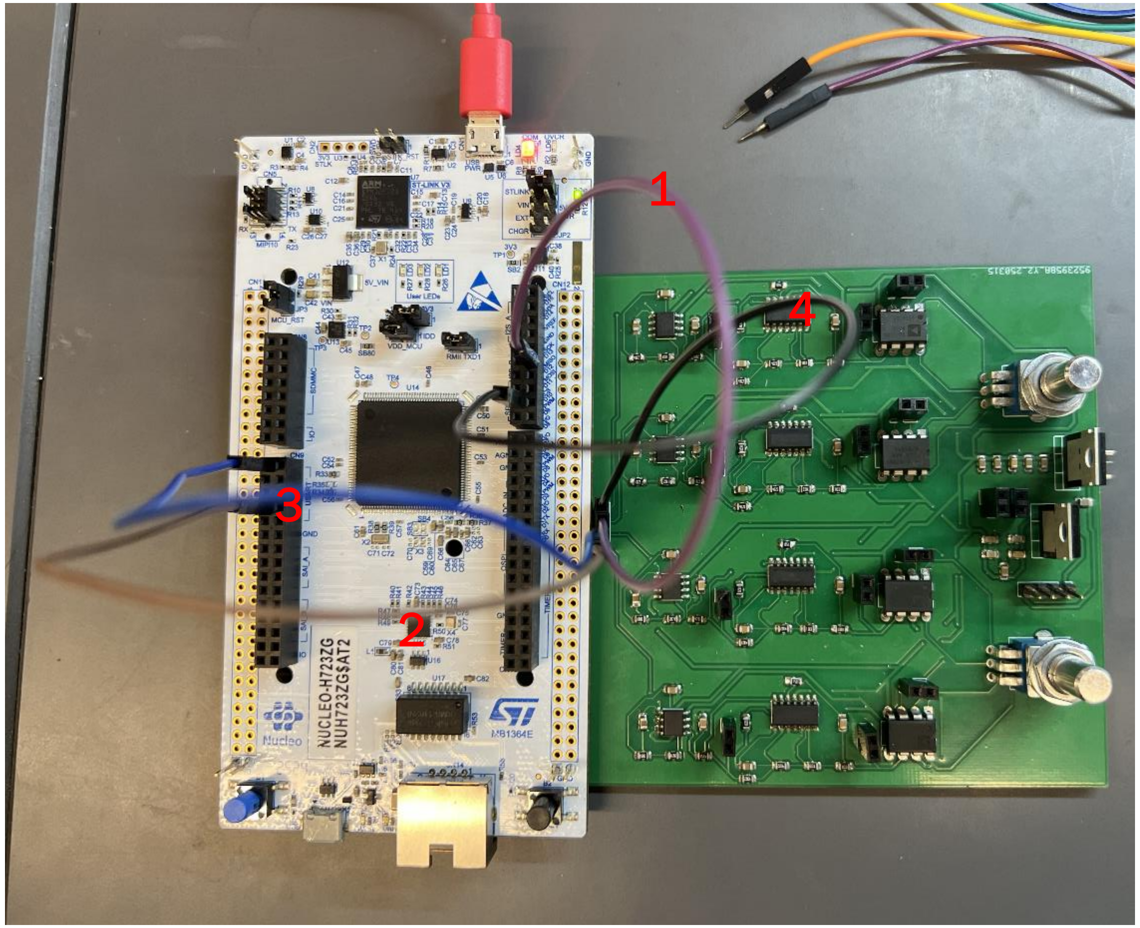
PURDUE UNIVERSITY

Microprocessor

Integration

This is how we combine the MCU with our EEG circuit

- 4 EEG circuit output to ADC input in MCU



Microprocessor

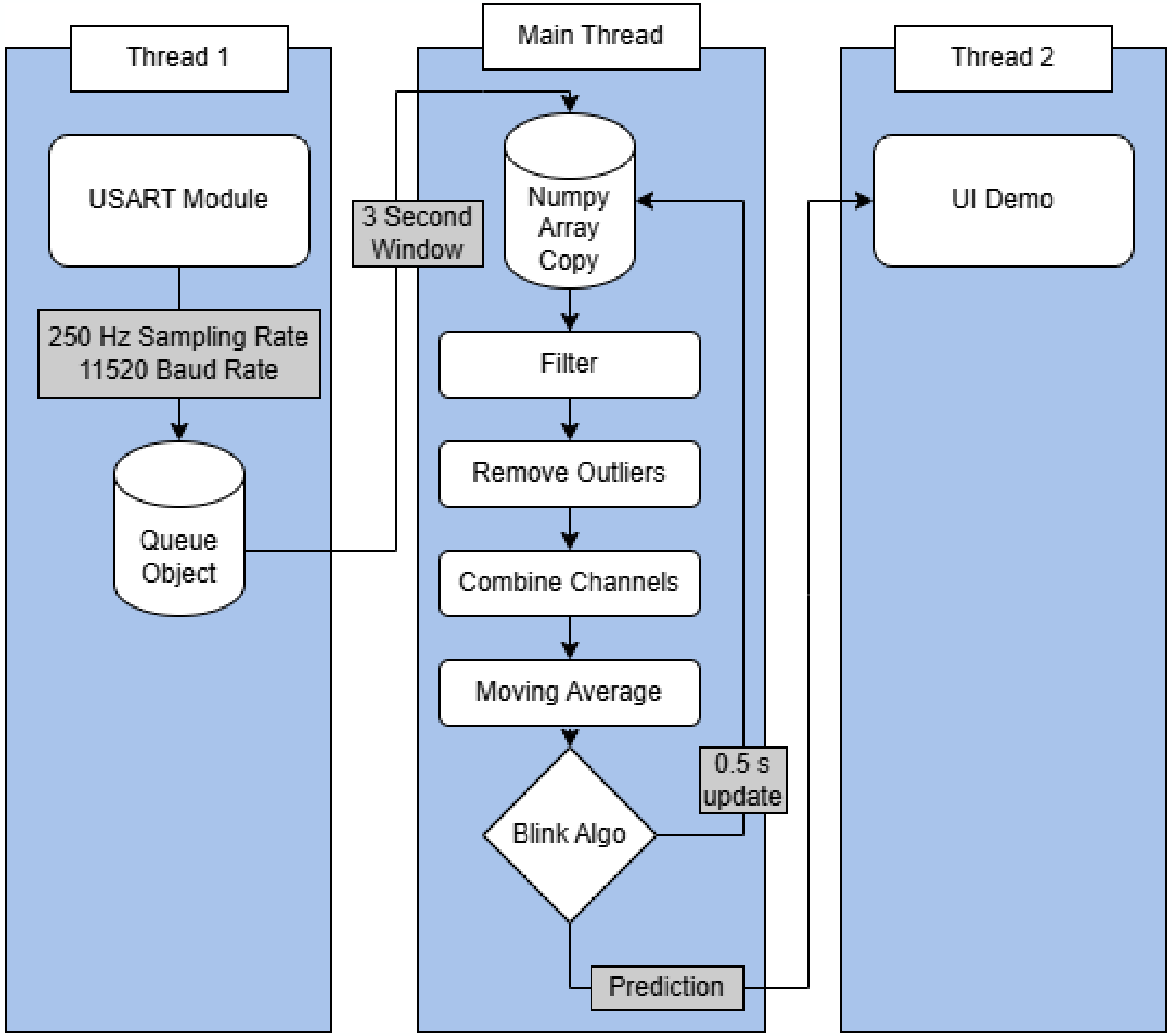
Progress

- Working on stabilizing the ADC for 4 channels.

Software and Algorithms

Subsystem Design

- Data Input
  - Reading Data from serial port
- Data Processing
  - Data transformations and predictions
- UI Control
  - Demo
  - Navigation to different features
    - Experiment
    - Visualization
    - Demo



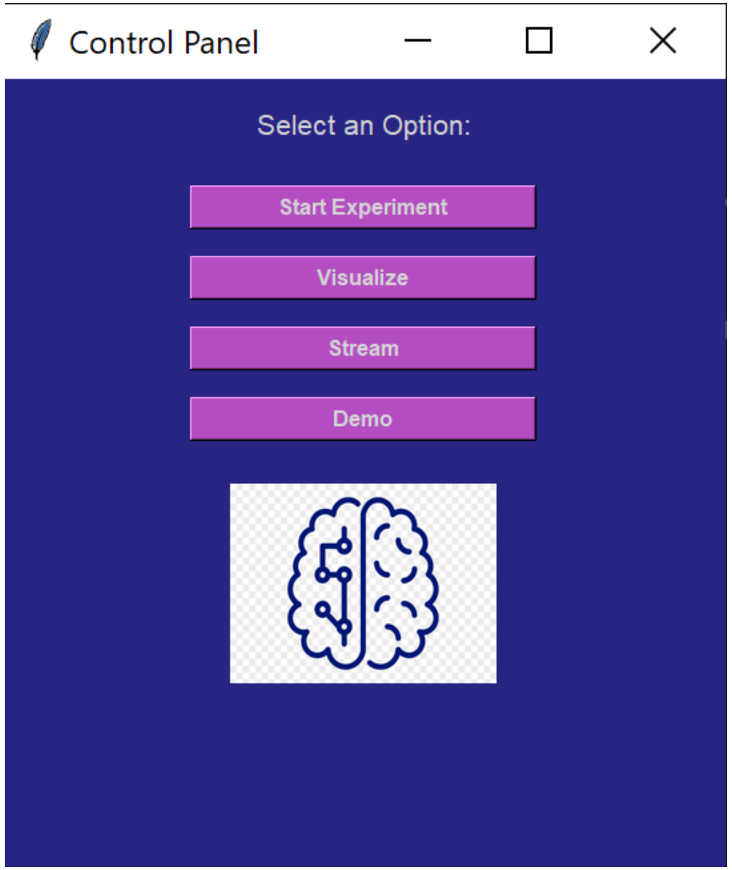
PURDUE UNIVERSITY

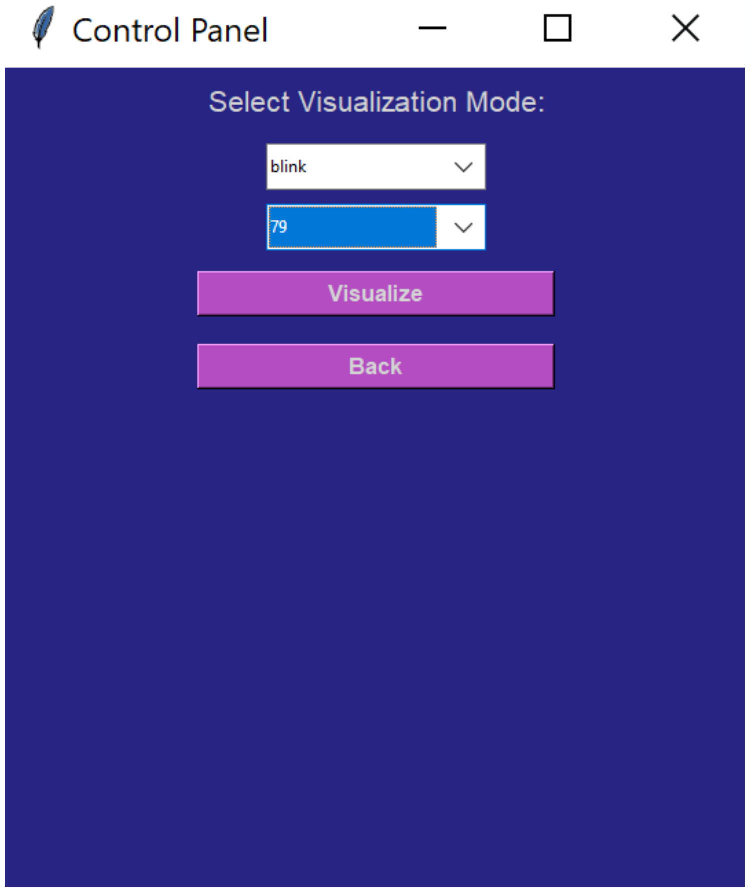
Software and Algorithms

Subsystem Functionality – UI Navigation

Main Menu

- Experiment
- Visualization
- Stream
- Demo
    - Dino Game
    - Assistant
    - Keyboard



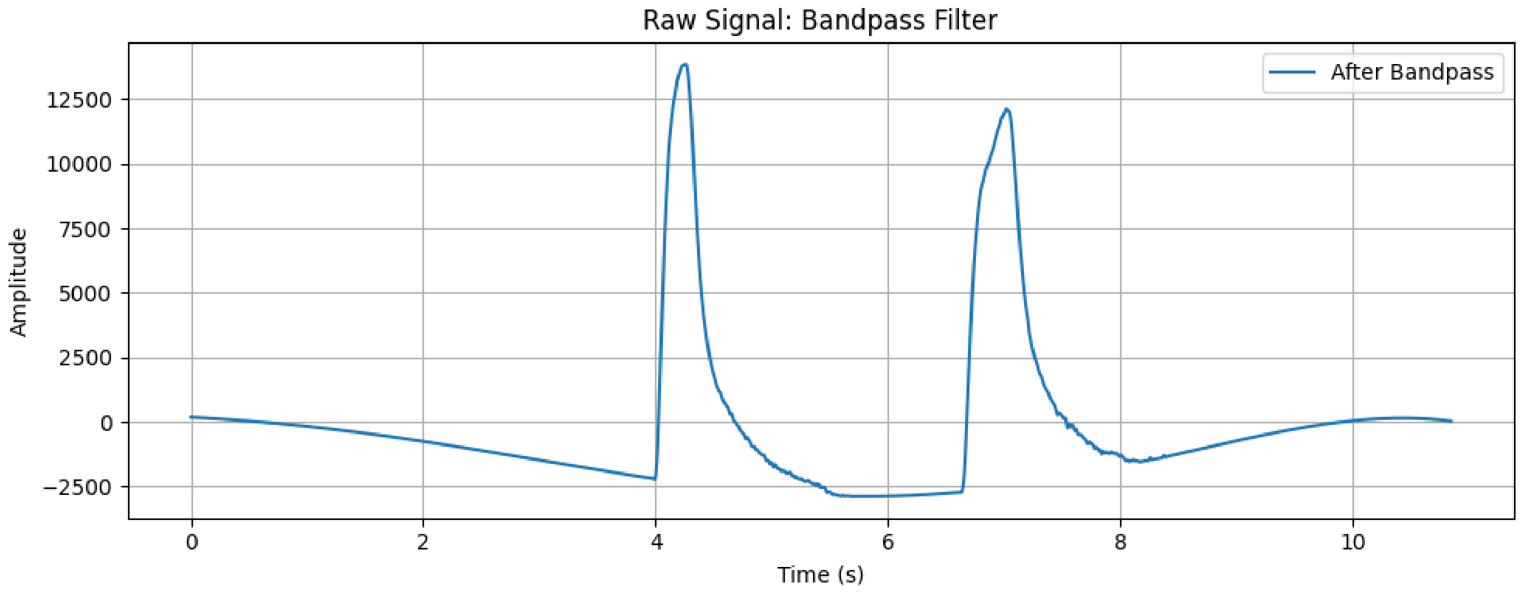


Software and Algorithms

Subsystem Functionality – Blink Algorithm

Threshold Based Algorithm

- Find peaks with SciPy based on STD of data
- Take derivative of signal to find where the peak flattens out
- Check that blink is within certain length
- Check that blink is not counted twice due to windowing



Software and Algorithms

Integration

- Fully Integrated with EEG and Microprocessor
- Reads 4 channels of EEG data in via USART module in MCU
  - Sample Rate is correct
  - Handles missing packets, too many packets, or no packets

```
PS C:\Users\jaspe\BCI_Senior_Design> python .\src\data_input\test_in.py
Testing sampling rate over 1 seconds
Duration of Measurment: 1.0018200874328613 seconds
Number of samples recorded: 250
Sampling rate 250 / 1.0018200874328613 = 249.5458048167298


Testing sampling rate over 2 seconds
Duration of Measurment: 2.0007100105285645 seconds
Number of samples recorded: 502
Sampling rate 502 / 2.0007100105285645 = 250.91092530065234


Testing sampling rate over 3 seconds
Duration of Measurment: 3.003199815750122 seconds
Number of samples recorded: 753
Sampling rate 753 / 3.003199815750122 = 250.73256732733248
```

Software and Algorithms

Specification, relevance, and justification

Specification	Status
Blink Accuracy > 90%	
Mouse Accuracy > 80%	
Latency < 1 s	
Data Processing (filter, outlier, smoothing)	
Command output demos	

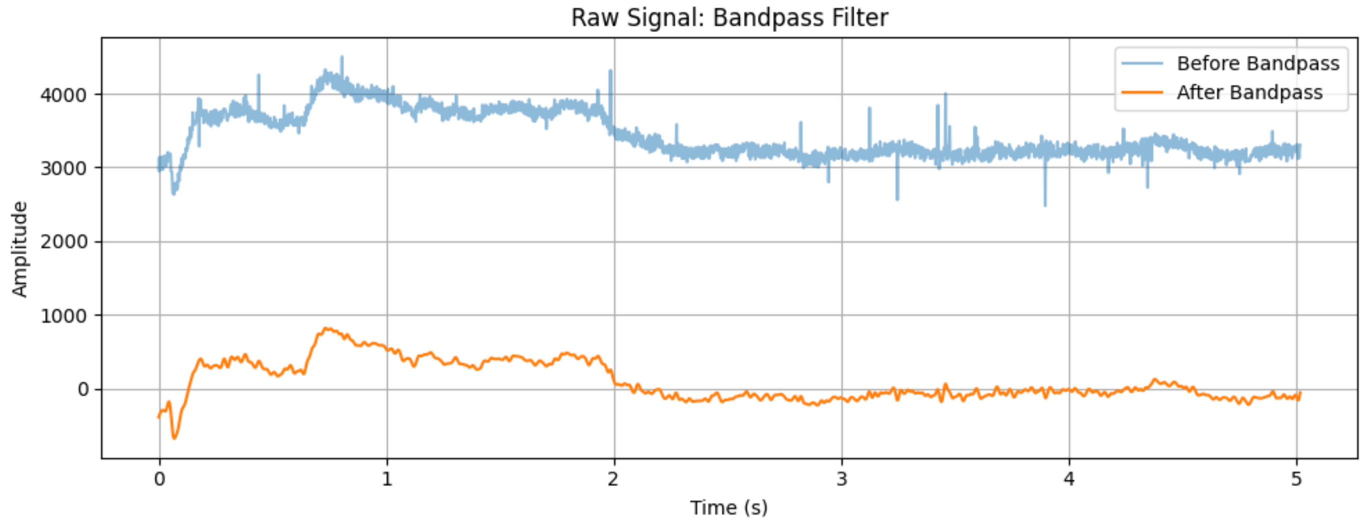
Software and Algorithms

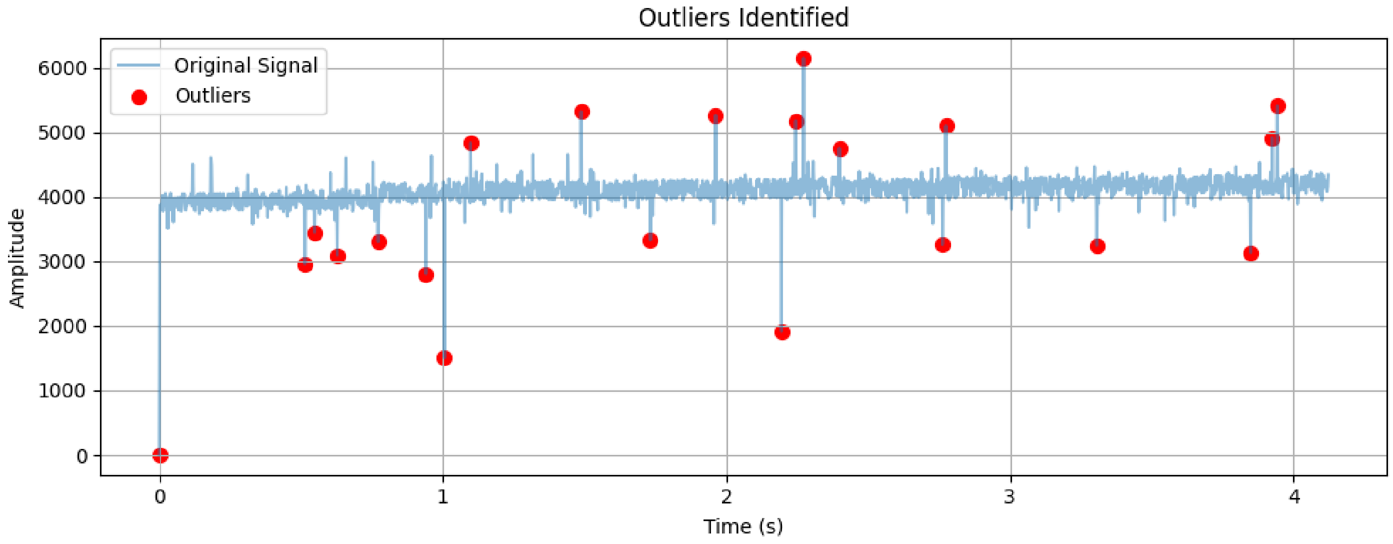
Specification Evidence – Data Processing

Bandpass filter:
- 0.1 - 20 Hz

Notch Filter:
- 60 hz

Outlier detection:
- Z Score





PURDUE UNIVERSITY

Software and Algorithms

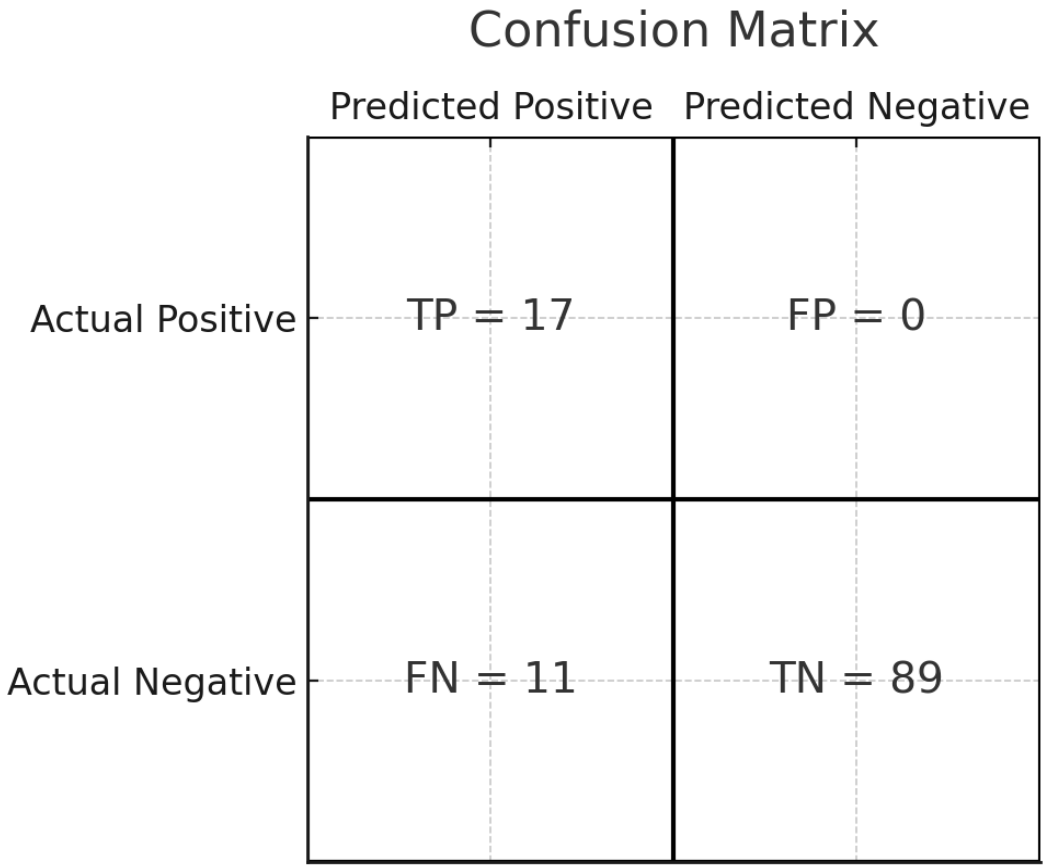
Specification Evidence – Blink Classification Accuracy

- **Experiment setup**
  - 2 full minutes of real time blink detection
    - Num samples = (total_secs – window) / (update_cycle_time)
    - 117 = (120 – 3) / (1)
  - Logged predictions for each window
  - Metrics computed after experiment

Metrics:
Precision: 1.0000
- TP / FP + TP

Recall: 0.6071
- TP / TN + TP

Accuracy: 0.9060



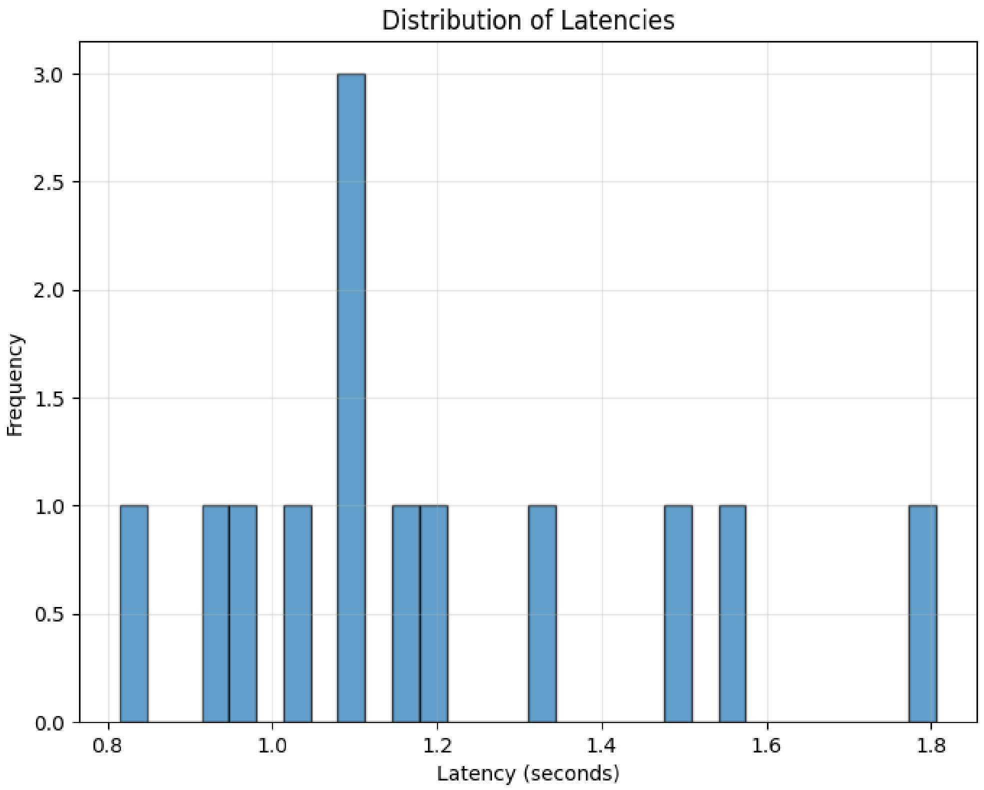
Software and Algorithms

Specification Evidence – Latency Experiment

- **Experiment setup**
  - 60 seconds of data
    - Log blink time with keyboard
    - Log blink detected time

Experiment Results

- Reason for Variance
  - Threading
  - Background local processes
  - Windowing of data
  - Inconsistent ground truth



Metrics:

Mean Latency: 0.72 s

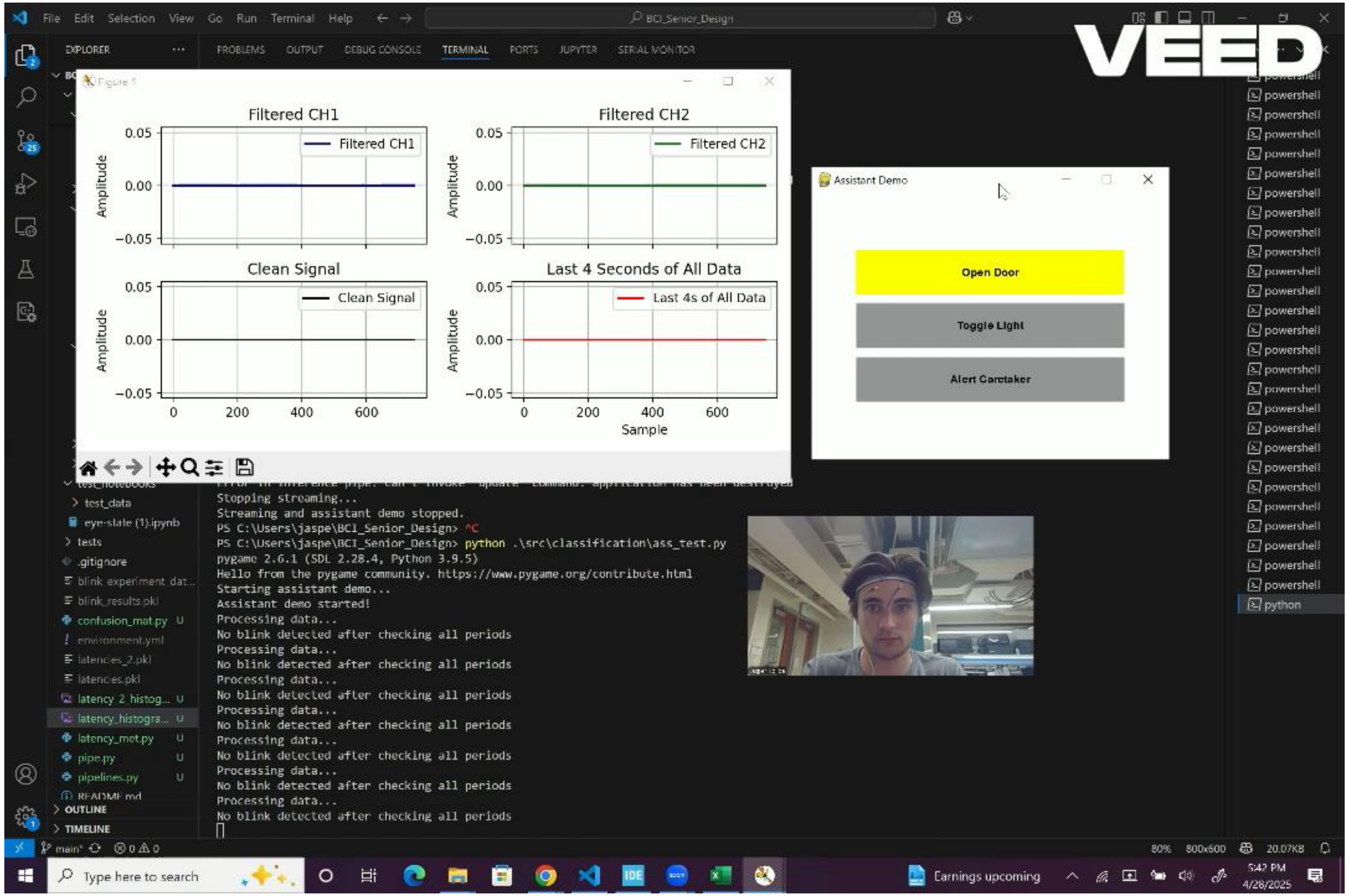
Var of Latency: 0.1 s^2

Median Latency: 0.73 s

PURDUE UNIVERSITY

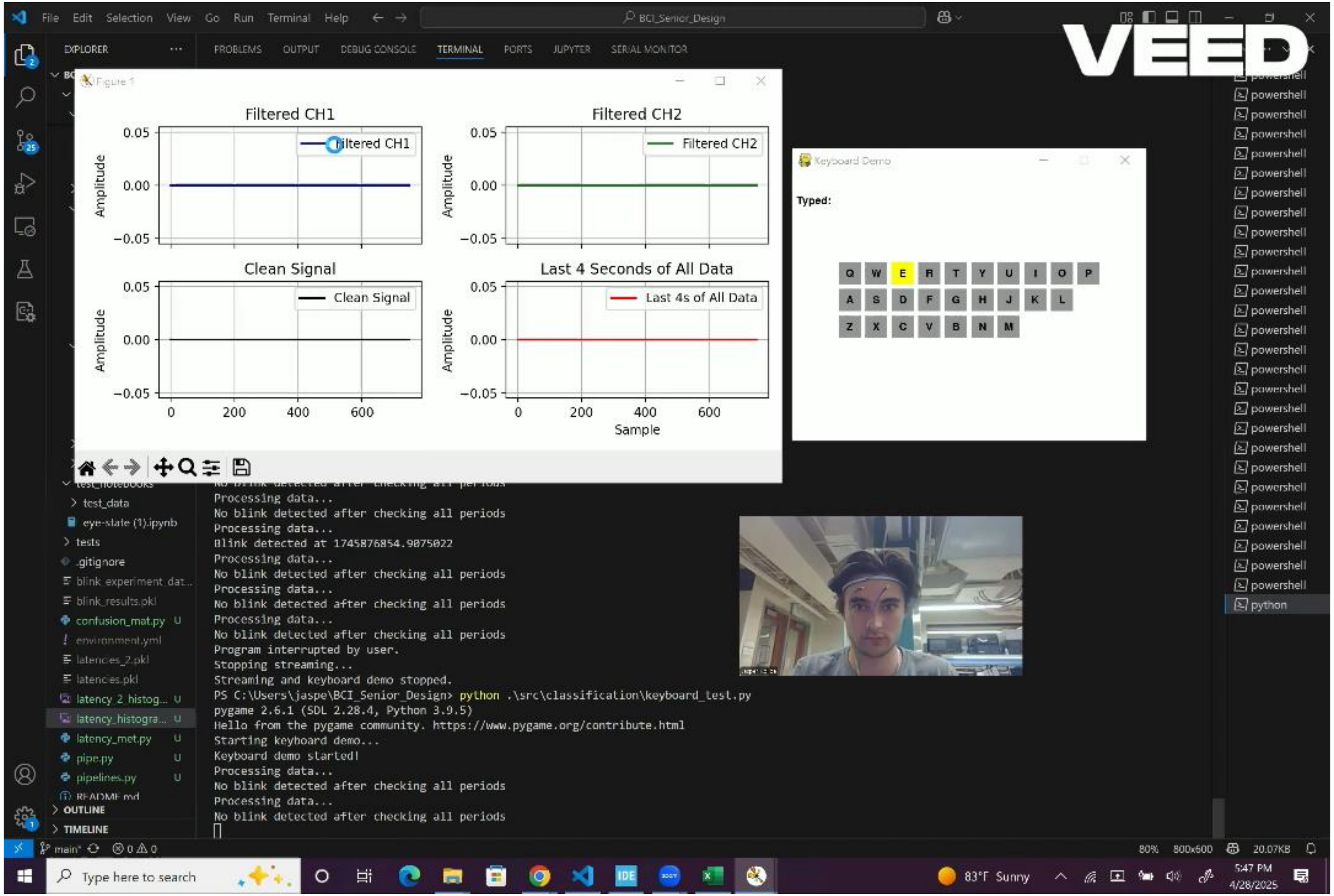
Software and Algorithms

Assistant Demo



Software and Algorithms

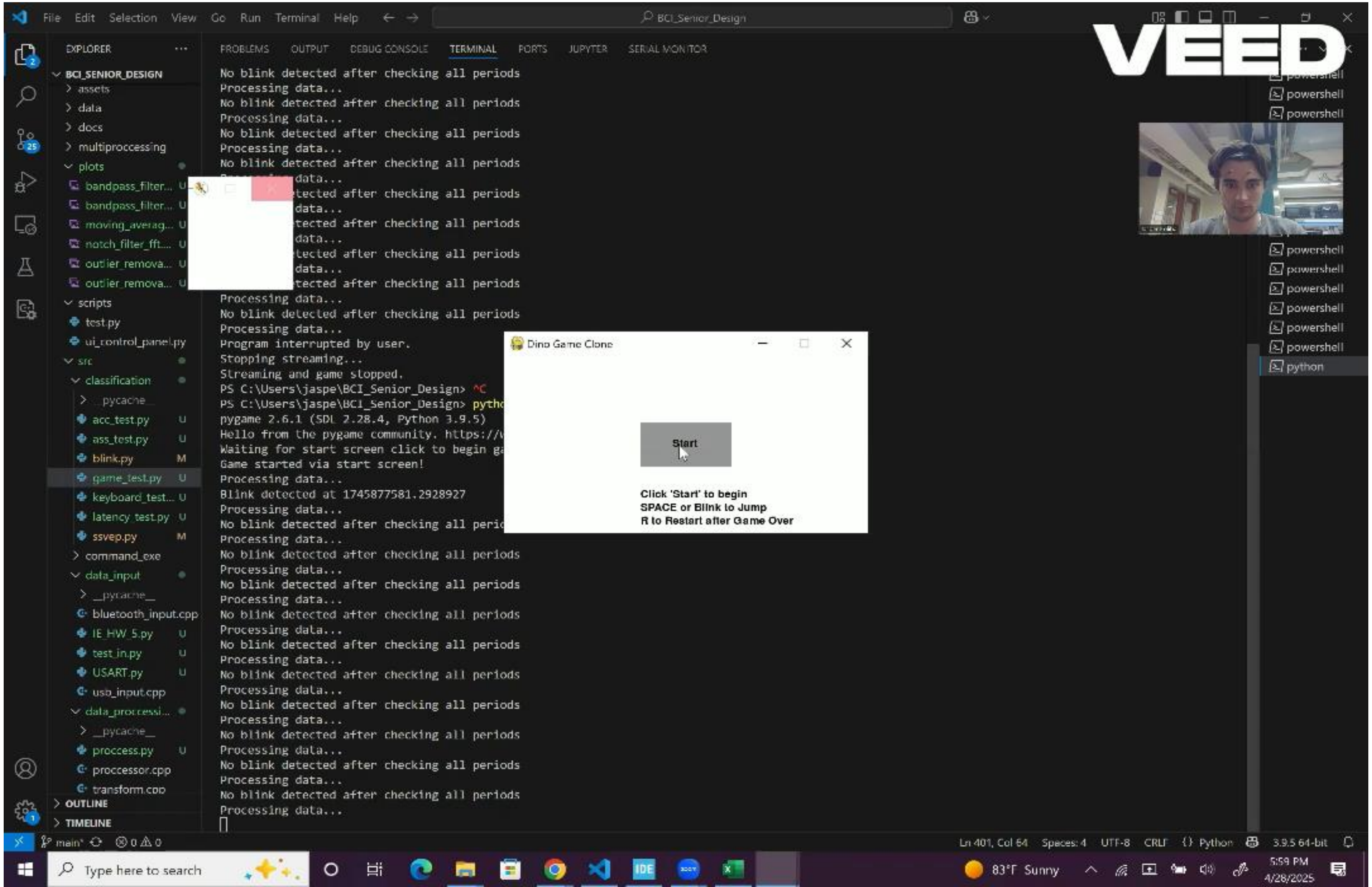
Keyboard Demo



PURDUE
UNIVERSITY®

Software and Algorithms

Game Demo



Software and Algorithms

Progress

- Blink Detection
  - Working with desired functionality
- Mouse Movement
  - Not working with desired functionality

Why Mouse Movements didn't work:

- Indented Approach – SSVEP (Steady State Visual Evoked Potentials)
  - Core Idea: By looking at a given frequency that frequency is present in visual cortex
  - Why it didn't work:
    - Electrodes not making good contact with scalp
    - Noise in circuit
    - Not optimized

Thank you, Any Questions?