Acoustic Stimulation to Improve Sleep

ECE 445 Final Report – Spring 2026

Team 18

Sid Gurumurthi

Bakry Abdalla

John Patrick Ludeke

Professor: Arne Fliflet

TA: Mingrui Liu

Abstract

This report presents the design and implementation of a closed-loop auditory stimulation system intended to enhance slow-wave sleep (SWS) using real-time EEG analysis and phase-aligned acoustic stimulation. The system acquires EEG data through an OpenBCI Cyton board and performs single-channel SWS classification using machine learning and signal processing techniques. Detected slow-wave activity is used to trigger precisely timed pink noise stimulation aligned with predicted slow-wave peaks. The final system integrates EEG acquisition, embedded signal processing, real-time stimulation control, and audio output within a unified hardware and software platform. This report discusses the design evolution, subsystem implementation, verification procedures, performance evaluation, costs, and ethical considerations associated with the project.

Table of Contents

1. Introduction

Slow-wave sleep (SWS), also referred to as deep sleep or N3 sleep, is an important stage of sleep associated with memory consolidation, neural recovery, metabolic regulation, and overall cognitive function. However, the amount of SWS naturally decreases with age and is often further reduced in individuals with sleep disorders. Research has shown that individuals younger than 55 years old average approximately 9.1% SWS as a percentage of total sleep time, while individuals older than 55 years old average approximately 2.1% SWS. Additionally, patients with sleep apnea average approximately 3.7% SWS compared to approximately 9.1% in healthy individuals. Reduced SWS has been associated with memory impairment, hypertension, immune dysfunction, psychiatric disorders, and increased risk of metabolic disease [7].

One proposed method for enhancing SWS is closed-loop auditory stimulation, where acoustic stimuli are delivered in synchronization with ongoing slow-wave oscillations measured through electroencephalography (EEG). Prior work has demonstrated that phase-aligned auditory stimulation can enhance slow-wave oscillation activity and improve memory consolidation during sleep [7]. This motivated the development of a system capable of detecting SWS in real time and delivering low-latency, phase-aligned audio stimulation.

The final system acquires EEG data using an OpenBCI Cyton board and performs real-time signal processing on an ESP32 microcontroller to classify SWS and estimate slow-wave timing. Once slow-wave activity is detected, the system generates phase-aligned pink noise stimulation through a DAC, amplifier, and speaker subsystem. Figure 1 shows the high-level architecture of the final system.

The project underwent several architectural changes throughout development. The original design centered around a custom PCB integrating the ADS1299 analog front end, signal processing microcontroller, audio subsystem, and power circuitry into a single platform. However, as development progressed, the team determined that a more modular architecture based around the OpenBCI Cyton board would significantly reduce hardware risk while allowing greater focus on the closed-loop signal processing and stimulation objectives of the project. The

final implementation therefore used the Cyton board for EEG acquisition and a separate custom PCB for embedded processing, power regulation, and audio output generation.

1.1 Background and Motivation

Slow-wave sleep plays a significant role in neurological and physiological recovery. During SWS, the brain exhibits high-amplitude, low-frequency oscillations typically within the 0.5-4 Hz range. These oscillations are associated with synaptic homeostasis, memory consolidation, and restoration processes that contribute to cognitive performance and long-term health [7].

Despite its importance, SWS decreases substantially with age and in patients with sleep disorders. Individuals younger than 55 years old average approximately 9.1% SWS during sleep, while individuals older than 55 years old average approximately 2.1% SWS. Patients with sleep apnea similarly exhibit reduced SWS percentages of approximately 3.7% compared to approximately 9.1% in healthy individuals. Reduced SWS has been linked to impaired memory formation, hypertension, diabetes risk, psychiatric disorders, and weakened immune function [7].

Because SWS is characterized by identifiable slow-wave oscillations in EEG signals, it is possible to detect and analyze this sleep stage using noninvasive EEG acquisition techniques. Prior research has demonstrated that auditory stimulation delivered in phase with ongoing slow-wave oscillations can strengthen these oscillations and improve memory consolidation performance [7]. This creates the need for a closed-loop system capable of accurately detecting SWS and delivering low-latency stimulation aligned with the timing of slow-wave activity.

The goal of this project was therefore to develop a real-time EEG-based auditory stimulation system capable of:

- acquiring EEG signals during sleep,
- detecting slow-wave sleep using machine learning and signal processing techniques,
- estimating slow-wave oscillation timing in real time, and
- generating phase-aligned pink noise stimulation with low latency and reliable operation.

1.2 System Overview

The final system consists of three primary subsystems: the EEG acquisition subsystem, the embedded signal processing and control subsystem, and the audio stimulation subsystem. Figure 1 shows the high-level block diagram of the final implementation.

The EEG acquisition subsystem consists of an EEG headband connected to the OpenBCI Cyton board. EEG signals are measured from the user during sleep and digitized by the Cyton board using the ADS1299 analog front end [1], [4]. The Cyton board streams digitized EEG data to the ESP32 microcontroller through a UART communication interface using the OpenBCI packet format [3].

The embedded subsystem running on the ESP32 performs real-time signal processing on the incoming EEG data. This includes causal filtering, slow-wave frequency estimation, trough detection, and SWS classification using features derived from zero-crossing point analysis [9]. When slow-wave activity is detected, the ESP32 predicts the timing of upcoming slow-wave peaks and schedules stimulation events accordingly.

To generate auditory stimulation, the ESP32 communicates with an MCP4822 DAC through an SPI interface. The DAC converts digitally generated pink noise samples into an analog waveform, which is amplified using the PAM8302 audio amplifier [8] and played through a speaker positioned near the user. A dedicated battery-powered 3.3 V rail powers the embedded and audio subsystems to support portable operation and reduce external electrical interference. Additionally, the ESP32 transmits real-time EEG and stimulation data to a phone application through Bluetooth communication for visualization and debugging purposes. This enabled live monitoring of filtered EEG signals, predicted stimulation peaks, and stimulation timing during system testing.

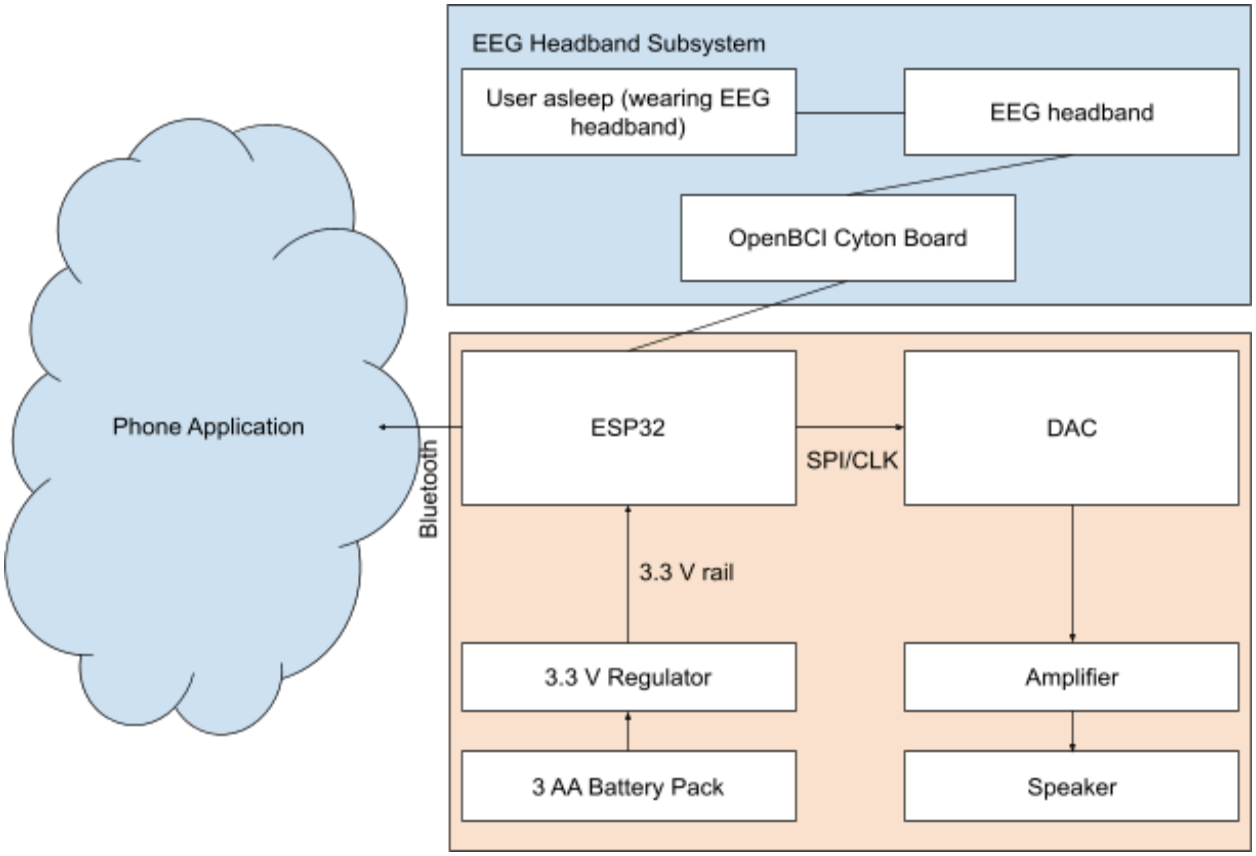

Figure 1. High-level Block Diagram

1.3 Initial vs Final Design

The original system architecture was designed around a fully custom PCB integrating the ADS1299 analog front end, microcontroller, power delivery subsystem, and audio subsystem into a single platform. The intent of this architecture was to replicate the functionality of existing EEG acquisition hardware while maintaining complete control over the signal acquisition and processing pipeline.

During development, the team realized that the original PCB design introduced substantial hardware complexity within the limited project timeline. The design required integration of many small surface-mount components, sensitive analog circuitry, multiple power domains, high-resolution EEG acquisition hardware, and embedded audio circuitry simultaneously. Additionally, fabrication and assembly timing became a concern after learning

about PCB manufacturing and turnaround constraints later in the semester, after the fourth-round PCB order deadline had already passed.

At the same time, the team had already begun prototyping using the OpenBCI Cyton board and an external audio breadboard while waiting for the original PCB fabrication process. This prototype architecture proved effective for iteration and debugging because it leveraged an already validated EEG acquisition platform while allowing development effort to focus on the real-time signal processing and stimulation pipeline.

As a result, the final system architecture transitioned toward a modular implementation using the OpenBCI Cyton board for EEG acquisition and a separate custom PCB containing the ESP32, DAC, amplifier, and power circuitry for audio generation and embedded processing. This significantly reduced hardware risk while preserving the project's core functionality and design objectives.

Another architectural refinement involved the visualization interface used for real-time monitoring and debugging. The final implementation used Wi-Fi-based communication between the ESP32 and the mobile visualization interface because it provided more reliable continuous data streaming and simplified real-time visualization integration compared to Bluetooth-based communication.

To support the final architecture, a new audio and power PCB was designed and fabricated independently to interface directly with the Cyton-based acquisition system. This final implementation more closely matched the prototypes developed throughout the semester and enabled reliable end-to-end demonstration of EEG acquisition, SWS detection, and phase-aligned auditory stimulation.

2. Design

The final system combines EEG acquisition, embedded signal processing, real-time stimulation control, and audio generation into a closed-loop auditory stimulation platform. The design was divided into multiple subsystems to simplify implementation and debugging while allowing independent development of EEG acquisition, signal processing, and audio output functionality.

The system acquires EEG signals using the OpenBCI Cyton board and processes the incoming data on an ESP32 microcontroller. The ESP32 performs real-time filtering, slow-wave sleep (SWS) classification, and slow-wave oscillation timing estimation. When slow-wave activity is detected, the ESP32 generates phase-aligned pink noise stimulation through a DAC and audio amplifier chain. Wi-Fi communication was additionally implemented to support real-time visualization and debugging during development.

2.1 System Architecture

The final architecture consists of three major subsystems:

1. EEG acquisition subsystem,
2. embedded processing and control subsystem, and
3. audio stimulation subsystem.

The EEG acquisition subsystem measures neural activity using scalp electrodes wired to the OpenBCI Cyton board. The Cyton board digitizes EEG signals using the ADS1299 analog front end [1], and transmits EEG packets through the OpenBCI USB dongle using the Cyton packet protocol [3].

The embedded processing subsystem is centered around the ESP32 microcontroller. Incoming EEG packets are parsed in real time and passed through multiple signal processing stages including causal filtering, frequency estimation, trough detection, and SWS classification. The ESP32 also manages timing synchronization and stimulation scheduling to support phase-aligned audio delivery.

The audio subsystem converts digitally generated pink noise samples into analog audio output using an MCP4822 DAC and PAM8302 amplifier [8]. The amplified signal is then played through a speaker.

The system additionally supports Wi-Fi communication between the ESP32 and a web application. Initial visualization and debugging were performed using a Python script that parsed serial monitor output from the ESP32 and generated real-time plots of filtered EEG signals and stimulation events. After validating this workflow, equivalent functionality was implemented through the Wi-Fi-based web application interface.

The modular architecture significantly simplified testing and debugging because EEG acquisition, processing, and stimulation functionality could each be validated independently before full system integration.

2.2 EEG Acquisition Subsystem

The EEG acquisition subsystem is responsible for measuring and digitizing neural activity during sleep. EEG data was acquired using the OpenBCI Cyton board, which uses the ADS1299 analog front end for low-noise biopotential acquisition [1], [4]. The ADS1299 performs amplification and 24-bit analog-to-digital conversion of incoming EEG signals before transmitting digitized samples through the Cyton communication interface.

The system primarily used a single EEG channel referenced between the C3 and M2 electrode locations. This configuration was selected because prior work demonstrated that single-channel EEG is sufficient for SWS classification [9], while also simplifying signal acquisition and processing requirements.

The Cyton board samples EEG data at 250 Hz and transmits data packets through the OpenBCI USB dongle using the Cyton packet format [3]. To interface directly with this data stream, the RX and GND pins of the ESP32 were connected to the corresponding RX and GND pins of the Cyton dongle. This allowed the ESP32 to intercept the raw serial bitstream generated by the Cyton system.

To verify correct communication and packet integrity, all incoming serial data was initially printed directly to the ESP32 serial monitor. Packet parsing was then implemented according to

the OpenBCI Cyton data format specification [3], including synchronization using packet start and stop bytes and extraction of channel sample values from each packet. The Cyton data format consists of 33-byte packets beginning with a header byte of 0xA0 and ending with a footer byte of the form 0xCX, where the intermediate bytes contain the sample number, 24-bit signed EEG channel values for eight channels, and filler data fields [3]. Figure E5 in Appendix E shows correctly packetized EEG data printed from the ESP32 serial monitor, confirming successful synchronization to the Cyton packet structure and proper extraction of incoming channel values.

After packet parsing was implemented, incoming EEG channel values were visualized using the Arduino serial plotter and compared against signals displayed in the OpenBCI GUI. Similar signal fluctuations and waveform behavior were observed in both interfaces, confirming successful packet transmission and correct reconstruction of EEG channel data on the ESP32.

The Cyton platform additionally performs several hardware-level functions internally, including amplification, differential measurement, common-mode rejection, and high-resolution digitization through the ADS1299 [1]. This provided a stable and validated EEG acquisition front end while allowing development effort to focus on the embedded signal processing and stimulation pipeline.

2.3 Signal Processing and SWS Classification

The signal processing subsystem performs real-time EEG filtering and SWS classification using features derived from the incoming EEG signal. The classification pipeline was developed and validated in Python using the Sleep-EDF Expanded dataset [5], [6] before being adapted for embedded execution.

The implemented classification approach was based on the single-channel SWS detection method proposed by Su et al. [9]. EEG recordings were segmented into 30-second epochs, and a zero-crossing point (ZCP)-based feature vector was extracted from each epoch. The feature vector is defined as:

$$x = [x_1, x_2, x_3] \quad (1)$$

where:

- x_1 is the mean zero-crossing segment length,

- x_2 is the standard deviation of zero-crossing segment lengths, and
- x_3 is the sum of each zero-crossing segment length weighted by the area under the absolute value of the EEG signal within that segment [9].

This feature vector captures the temporal characteristics associated with slow-wave activity while remaining computationally lightweight enough for embedded implementation. Figure E3 in Appendix E illustrates the detected zero-crossing locations within a representative EEG epoch used for feature extraction.

Before feature extraction, EEG signals were filtered using causal bandpass filters to isolate relevant EEG frequency content while preserving real-time operation. Multiple machine learning classifiers were evaluated during development, including logistic regression, random forest, and multilayer perceptron (MLP) models. The MLP model was ultimately selected due to its strong classification performance and improved balance between sensitivity and specificity.

The implemented MLP consists of a feedforward neural network containing an input layer, hidden fully connected layers, and an output layer used for binary SWS classification. Nonlinear activation functions within the hidden layers allow the model to learn more complex decision boundaries between SWS and non-SWS EEG feature patterns than simpler linear classifiers. The resulting confusion matrix and classification performance metrics for the final MLP model can be seen in the MLP section of Figure B1 in Appendix B.

The final classification pipeline continuously updates using a rolling 30-second EEG window and produces real-time SWS predictions used by the stimulation subsystem.

2.4 Phase-Aligned Audio Stimulation

In addition to detecting SWS, the system was designed to deliver audio stimulation aligned with the phase of ongoing slow-wave oscillations. Prior work demonstrated that stimulation delivered near the up-state of slow-wave oscillations can strengthen slow-wave activity and improve memory consolidation [7].

To estimate slow-wave timing in real time, the EEG signal was first passed through a causal 0.5-4 Hz bandpass filter to isolate slow-wave oscillations. The dominant slow-wave

frequency was then estimated using autocorrelation over a rolling EEG buffer. The oscillation period was computed as:

$$T = 1/f \tag{2}$$

where f is the estimated dominant slow-wave frequency.

Real-time trough detection was then performed on the filtered EEG signal. Troughs were detected as local minima below -25μV with a minimum spacing of 0.7 seconds between detections. Once a trough was detected, the system predicted the timing of the next slow-wave peak by offsetting the detected trough time by approximately half of the estimated oscillation period calculated from Equation (2):

$$t_{peak} = t_{trough} + T/2 \tag{3}$$

This predicted peak time, calculated from Equation (3), was used to schedule playback of a short pink noise burst through the audio subsystem. Figure E1 in Appendix E illustrates the trough detection and peak prediction process used for phase-aligned auditory stimulation.

The use of causal filtering and time-domain peak prediction allowed the system to operate in real time without requiring future EEG samples, which is critical for closed-loop stimulation systems.

2.5 Audio Subsystem

The audio subsystem converts digitally generated stimulation signals into audible pink noise bursts delivered to the user during sleep. The subsystem consists of an MCP4822 DAC, PAM8302 audio amplifier, and speaker output stage [8].

The ESP32 generates pink noise samples digitally and transmits these samples to the MCP4822 DAC using an SPI communication interface. Compared to PWM-based audio generation approaches explored earlier in development, the DAC-based implementation provided cleaner analog output and reduced high-frequency switching noise.

The DAC output is routed into the PAM8302 amplifier, which amplifies the analog audio signal to a level sufficient for speaker playback. The amplifier output is then connected to a small speaker positioned near the sleeping user. Figure C2 in Appendix C shows the final schematic for the audio subsystem, while Figure D2 in Appendix D shows the corresponding PCB layout implementation.

The audio subsystem was powered using a dedicated 3.3 V rail generated from a battery-powered supply. Battery operation reduced dependence on external power sources and helped minimize electrical interference coupling into the system.

During testing, the subsystem successfully generated clearly audible pink noise bursts with low latency and stable operation.

2.6 Embedded Implementation

The embedded processing pipeline was implemented on an ESP32 microcontroller due to its integrated Bluetooth functionality, sufficient computational performance, and support for real-time peripheral interfaces.

The ESP32 continuously receives EEG packets from the Cyton dongle through UART communication at 115200 baud. Incoming packets are parsed and converted into signed EEG sample values according to the Cyton packet format specification [3].

Once decoded, EEG samples are passed through the real-time signal processing pipeline consisting of:

- causal filtering,
- rolling buffer management,
- slow-wave frequency estimation,
- trough detection,
- feature extraction,
- SWS classification, and
- stimulation scheduling.

To support real-time visualization during debugging, the ESP32 additionally transmitted filtered EEG values, stimulation timing information, and event markers through the serial monitor, which were then parsed and displayed by a Python visualization script connected to the same serial port. As seen in Figure E6 in Appendix E, this allowed live observation of EEG signals and predicted stimulation events during testing.

The same real-time data and stimuli stream was later transmitted over Wi-Fi to the web application interface, where matching signal behavior and event timing relative to the Python visualization confirmed proper wireless data transmission.

Careful timing management was necessary to ensure stimulation events remained within 300ms of detected slow-wave activity. Because the system operated continuously on streaming EEG data, all processing stages were designed to execute causally and with minimal latency.

2.7 PCB Design and Hardware Integration

The final hardware implementation included a custom PCB integrating the ESP32 microcontroller, DAC, amplifier circuitry, and power regulation subsystem. The PCB was designed in KiCad and served as the central hardware platform for embedded processing and audio generation. Figure E7 in Appendix E contains the fabricated final audio and power PCB used for embedded EEG processing and auditory stimulation output.

The PCB includes:

- ESP32-WROOM-32E microcontroller,
- MCP4822 DAC,
- PAM8302 audio amplifier,
- voltage regulation circuitry, and
- peripheral interface connections for the Cyton Dongle and Speaker.

The layout prioritized modularity and clean routing of audio and power signals. Decoupling capacitors were placed near power pins to improve supply stability, and audio routing was separated from high-speed digital traces where possible to reduce coupling noise.

The final PCB design differed substantially from the original architecture proposed earlier in the semester. The original design attempted to integrate EEG acquisition hardware, including

the ADS1299 analog front end, directly onto the PCB alongside the embedded and audio subsystems. However, due to the complexity of integrating sensitive analog EEG circuitry within the project timeline, the final design instead leveraged the validated OpenBCI Cyton platform for EEG acquisition while retaining custom embedded and audio hardware.

This final architecture reduced hardware risk while still preserving the project's primary design objectives and enabling reliable end-to-end operation of the closed-loop stimulation system.

3. Verification

3.1 SWS Classification Performance

The SWS classification subsystem was evaluated using the Sleep-EDF Expanded dataset [5], [6] with expert-labeled sleep stages. Multiple models were tested, and a multilayer perceptron (MLP) was selected due to its strong performance.

The final model achieved accuracy and sensitivity exceeding the 75% requirement defined in the design review. The confusion matrix (Appendix B) shows reliable separation between SWS and non-SWS, with relatively few missed SWS detections. Some misclassification occurred during transitions between sleep stages, which is expected due to overlapping EEG features.

3.2 Real-Time EEG Processing Verification

The EEG processing pipeline was verified to ensure correct packet parsing, timing, and signal reconstruction from the OpenBCI Cyton stream. The system successfully parsed 250 Hz data and maintained continuous, correctly ordered samples.

Verification was performed by comparing ESP32-processed signals with OpenBCI GUI outputs and known input waveforms. The reconstructed signals showed consistent frequency and amplitude, confirming correct parsing and timing.

No dropped packets or synchronization issues were observed during extended tests. Filtered outputs also matched offline implementations, validating the embedded signal processing pipeline.

3.3 Phase Alignment Verification

Phase alignment was tested using synthetic low-frequency signals to emulate slow-wave activity. This allowed controlled evaluation of trough detection and peak prediction.

The delay between predicted peaks and audio output was measured using an oscilloscope. The system consistently produced stimulation within the required 300 ms window, meeting the design requirement.

Timing jitter was minimal across trials, and stimulation occurred near the intended slow-wave phase. Minor offsets due to frequency estimation did not significantly impact performance. Example measurements are shown in Appendix E.

3.4 Audio Output Verification

The audio subsystem was verified for signal quality and output level. Oscilloscope measurements confirmed clean DAC output with reduced noise compared to earlier PWM designs.

Acoustic testing was performed using an SPL meter at 10 cm from the speaker. The system produced an average output within the required 50 dB ±2 dB range, satisfying the specification (see Figure E4 in Appendix E).

Output levels were consistent across trials, and the generated pink noise was clearly audible without being disruptive.

3.5 Requirement Verification Table

System performance was evaluated against the requirements defined in the design review (Appendix A). All major requirements were successfully verified.

The classification subsystem exceeded accuracy targets, the processing pipeline maintained reliable real-time operation, and the stimulation subsystem met latency constraints. The audio subsystem satisfied SPL requirements, and system-level tests confirmed expected runtime and data integrity.

No major failures were observed. Minor deviations, such as small timing offsets, were within acceptable limits. Full verification details are provided in Appendix A.

4. Costs

4.1 Labor Costs

Using best estimates from our semester, along with not having to take into account the extra cost of others working besides the 3 team members. Using knowledge of internship salaries for engineering students from UIUC, we came up with an estimate of $50 per hour. Additionally, we assumed that each person was contributing around 15 hours of work each for 14 weeks.

$$14\,weeks \times 15\,hours/week \times 50\,\$/hour = \$10,500$$

4.2 Parts and Manufacturing Costs

Table 1 - Physical Parts Costs

Part	Manufacturer	Quantity	Retail Cost ($)	Total Cost ($)
ESP32-WROOM-32E (microchip)	Espressif Systems	3	4.99	14.97
LM1117T-3.3 (regulator)	Texas Instruments	3	1.61	4.83
2464 (3×AA PCB holder)	Keystone Electronics	1	1.77	1.77
HTSW-103-07-T-S (1×3 2.54 mm vertical header)	Samtec Inc.	2	0.11	0.22
HTSW-106-06-T-S (1×6 2.54 mm vertical header)	Samtec Inc.	2	0.44	0.88
1N5817 (diode)	Diodes Incorporated	3	0.22	0.66
1301.9306 (buttons)	SCHURTER Inc.	2	0.58	1.16
AS02508MS (speaker)	PUI Audio, Inc.	1	3.39	3.39
MCP4821-E/MS (DAC)	Microchip Technology	2	3.05	6.1
CL05B104KP5NNNC (0.1 µF caps)	Samsung Electro-Mechanics	12	0.1	1.2

CL21A106KOQNNNE (10 µF caps)	Samsung Electro-Mechanics	8	0.1	0.8
RC0603FR-7W10KL (10 kΩ resistor)	YAGEO	4	0.1	0.4
OpenBCI EEG Headband Kit	OpenBCI	1	349.99	349.99
MCP4822 DAC	Microchip	1	4.28	4.28
PAM8302 Audio Amplifier	Diodes Incorporated	1	0.5	0.5
			Total	**389.22**

Table 2 - Software Components Costs

Part	**Manufacturer**	**Retail Cost ($)**	**Actual Cost ($)**
React Native	Same Sky	0.00	0.00
Firebase	Google	0.00	0.00
FFT / ML Libraries	Open source	0.00	0.00
Total		0.00	0.00

4.3 Parts and Manufacturing Costs

We can find the total by adding all of our totals together.

$$\$10,500 + \$389.2 + \$0.00 = \$10,889.20$$

5. Conclusions

5.1 Executive Summary

This project demonstrated a closed-loop auditory stimulation system capable of detecting slow-wave sleep (SWS) in real time and delivering phase-aligned acoustic stimulation. The final system successfully integrates EEG acquisition, embedded signal processing, and audio output into a functional prototype that meets key performance requirements.

5.2 Technical Conclusions and Accomplishments

The system successfully achieved real-time EEG acquisition, SWS classification, and stimulation control. The classification algorithm exceeded required accuracy, and the embedded pipeline maintained reliable operation at 250 Hz.

Phase-aligned stimulation met latency requirements, and the audio subsystem satisfied output specifications. Overall, the integrated system operated reliably and validated the core closed-loop stimulation approach.

5.3 Limitations and Future Work

Despite meeting key requirements, several limitations remain. The SWS classification model was validated primarily on offline datasets, and performance may vary under real-world conditions with noise and motion artifacts. Additionally, phase alignment accuracy is dependent on frequency estimation, which introduces small timing errors.

Future work could include testing with human subjects in controlled sleep studies, improving artifact rejection, and refining phase prediction algorithms. Hardware improvements could include integrating EEG acquisition into the custom PCB and optimizing power consumption for extended operation.

5.4 Ethical and Societal Considerations

EEG data introduces privacy concerns, and the system must ensure secure handling of sensitive information. In line with the IEEE Code of Ethics, the design prioritizes safety and transparency and is not intended for clinical use. This technology has potential to improve sleep and cognitive health, but responsible use and accessibility should be considered for broader impact.

References

[1] Texas Instruments, “ADS1299 Data Sheet,” [Online]. Available: https://www.ti.com/product/ADS1299. [Accessed: Apr. 1, 2026].

[2] M. M. Can, “Open-Source DIY EEG: Brainwaves on a Custom PCB,” Hackster.io, Sep. 18, 2025. [Online]. Available: https://www.hackster.io/mustafamelihcan/open-source-diy-eeg-brainwaves-on-a-custom-pcb-99146e.

[3] OpenBCI, “Cyton Data Format,” Jul. 16, 2025. [Online]. Available: https://docs.openbci.com/Cyton/CytonDataFormat/.

[4] OpenBCI, “Cyton Specs,” Jul. 15, 2025. [Online]. Available: https://docs.openbci.com/Cyton/CytonSpecs/.

[5] B. Kemp, “Sleep-EDF Database Expanded,” PhysioNet, Oct. 24, 2013. [Online]. Available: https://physionet.org/content/sleep-edfx/1.0.0/.

[6] B. Kemp, “Sleep-EDF Database,” PhysioNet, Dec. 12, 2002. [Online]. Available: https://www.physionet.org/content/sleep-edf/1.0.0/.

[7] H.-V. V. Ngo et al., “Auditory closed-loop stimulation of the sleep slow oscillation enhances memory,” *Neuron*, vol. 78, no. 3, pp. 545–553, 2013, doi: 10.1016/j.neuron.2013.03.006.

[8] Diodes Incorporated, “PAM8302A Datasheet,” Apr. 2013. [Online]. Available: https://cdn-shop.adafruit.com/datasheets/PAM8302A.pdf. [Accessed: May 5, 2026].

[9] B.-L. Su et al., “Detecting slow wave sleep using a single EEG signal channel,” *Journal of Neuroscience Methods*, vol. 243, pp. 47–52, 2015, doi: 10.1016/j.jneumeth.2015.01.023.

Appendices

Appendix A – Requirements and Verification Table

Table A1: Requirements and Verification Table

Requirement	Verification Procedure	Equipment	Data / Presentation
The speaker shall produce 50 dB ±2 dB measured at 10 cm distance in ambient noise <40 dB during continuous stimulation.	1. Place a calibrated SPL meter 10 cm directly in front of the speaker. 2. Enable continuous pink noise output. 3. Record SPL for 10 seconds and compute the average value.	Calibrated SPL meter	Table of measured SPL values and average
Artifact Suppression: Over an 8-hour recording, the subsystem shall exhibit < 10% artifact-contaminated data. An artifact is defined as samples flagged by ADC saturation or lead-off detection.	1. Run an 8-hour overnight recording with the full system enabled. 2. Log raw EEG, lead-off status bit. 3. Compute artifact fraction = (time flagged artifact) / (total time). 4. Repeat for at least one additional trial if feasible.	Fully assembled device, PC/phone receiver + logging software, analysis script, optional accelerometer calibration jig.	Timeline plot showing artifact flags over 8 hours, excerpt plots (EEG segment before/during/after artifact).
The SWS classification algorithm shall achieve ≥ 75% accuracy and	1. Obtain labeled sleep dataset. 2. Run implemented ML model on the same dataset. 3. Compute overall accuracy: defined as how many	Laptop, Python environment, labeled dataset	Table of predicted vs. actual labels, calculated

sensitivity when compared against an 8-hour sample of labeled reference sleep data.	predictions were correct / total predictions (based on a classification of SWS or Not SWS). 4. Compute overall sensitivity: defined as true predictions of SWS / (true predictions of SWS + false predictions of SWS) 5. Generate a confusion matrix based on SWS vs Non SWS using the ML algorithm and the actual data.		accuracy, calculated sensitivity, confusion matrix
The system shall produce an audio response within 300 ms of detecting SWS.	1. Feed a synthetic slow-wave waveform into the PCB (e.g. a 1Hz sine wave) 2. Use an oscilloscope to capture the time difference between audio output and the audio trigger signal from the microcontroller.	Function generator, oscilloscope, laptop.	Oscilloscope view
The signal processing subsystem shall correctly parse 1-channel EEG data streamed from the OpenBCI Cyton Board at 250 Hz, producing 24-bit	1. Configure the OpenBCI Cyton Board to stream data at 250 SPS over UART. 2. Capture the raw data stream using the implemented packet parsing algorithm to extract sample frames, 3. Compare parsed output against expected waveform	Function generator (or DAC), resistor divider/attenuator, oscilloscope, PC logger (Python/MATLAB).	Table with measured DRDY frequency, plots of injected vs recorded waveform

digital samples with accurate channel extraction and timing.	characteristics (frequency, amplitude, and phase consistency).		
The system must support continuous operation for at least 10 hours on a full charge.	1. Write down the system's battery capacity in mAh. 2. Measure the average current draw over a short period of time during operation. 3. Divide battery capacity by average current draw to get the number of hours that the system should last.	Ammeter, fully integrated system.	A simple math calculation on a slide or piece of paper.
The application shall display EEG signal values that match the data transmitted from the headset within 10% error.	1. Stream EEG data from your headset to the app. 2. Simultaneously log the raw transmitted data to a laptop. 3. Compute the error between the App values and the raw data.	Laptop for raw EEG data, any device to access the App	Ground truth data, App data, error calculation.

Appendix B – SWS Classification Results

```
logreg
Balanced Accuracy: 0.9371227867399514
Confusion Matrix:
[[6737  799]
 [  19  944]]
              precision    recall  f1-score   support

           0     0.9972    0.8940    0.9428      7536
           1     0.5416    0.9803    0.6977       963

    accuracy                         0.9038      8499
   macro avg     0.7694    0.9371    0.8202      8499
weighted avg     0.9456    0.9038    0.9150      8499

==================================================
mlp
Balanced Accuracy: 0.9281829358228995
Confusion Matrix:
[[7197  339]
 [  95  868]]
              precision    recall  f1-score   support

           0     0.9870    0.9550    0.9707      7536
           1     0.7191    0.9013    0.8000       963

    accuracy                         0.9489      8499
   macro avg     0.8531    0.9282    0.8854      8499
weighted avg     0.9566    0.9489    0.9514      8499

==================================================
rf
Balanced Accuracy: 0.9363813818282835
Confusion Matrix:
[[6718  818]
 [  18  945]]
              precision    recall  f1-score   support

           0     0.9973    0.8915    0.9414      7536
           1     0.5360    0.9813    0.6933       963

    accuracy                         0.9016      8499
   macro avg     0.7667    0.9364    0.8174      8499
weighted avg     0.9451    0.9016    0.9133      8499

==================================================
```

Figure B1: SWS algorithm performance using three different models

Appendix C – Schematics

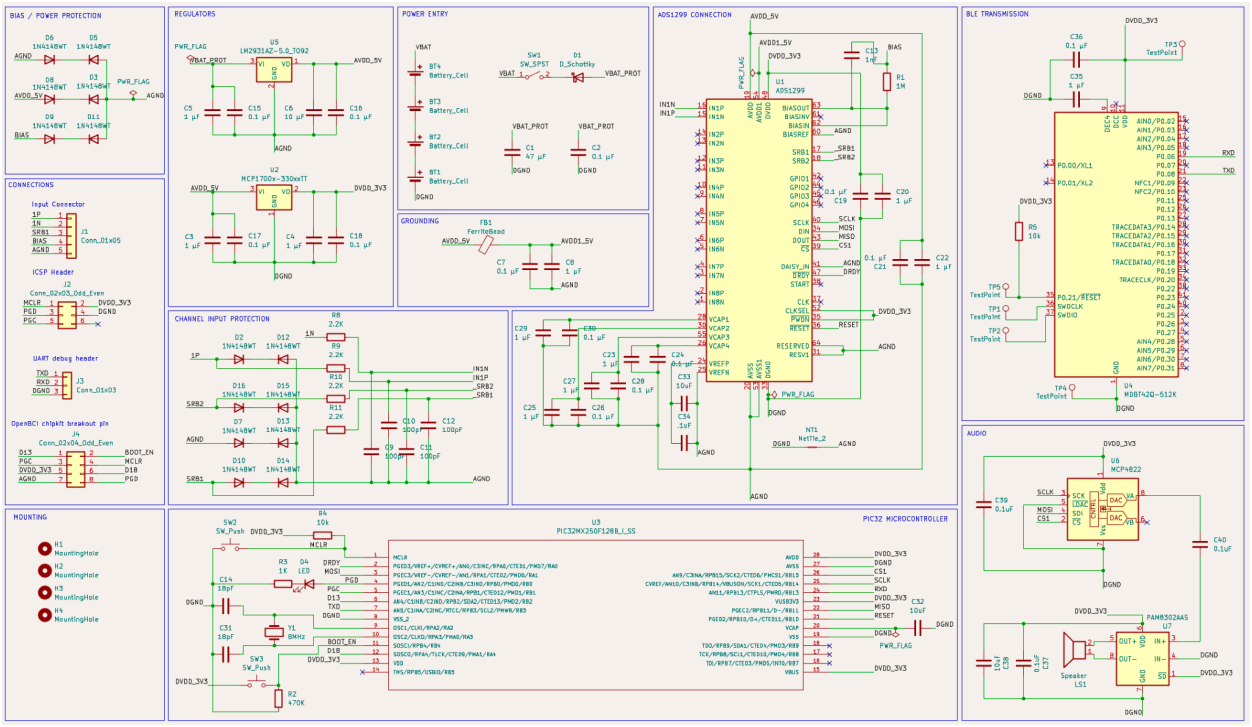

Figure C1: Initial Schematic of the PCB Design

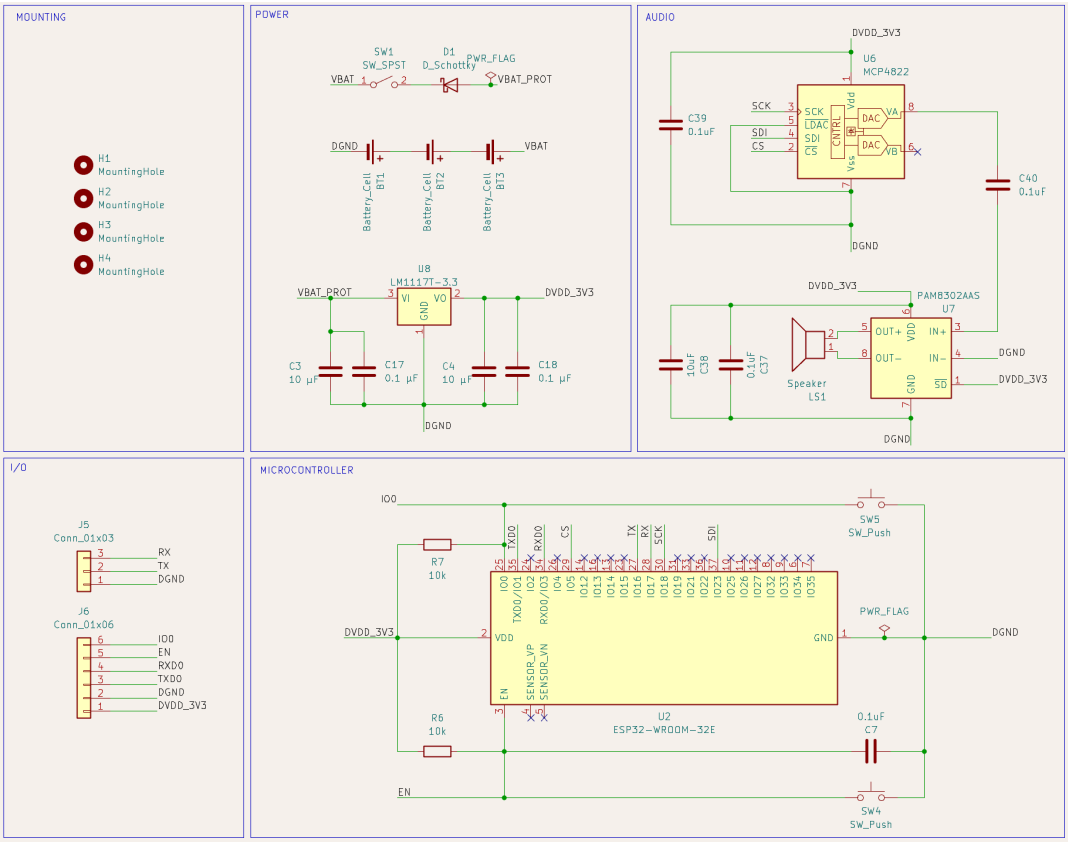

Figure C2: Final, Simplified Schematic of the PCB Design

Appendix D – PCB Layouts

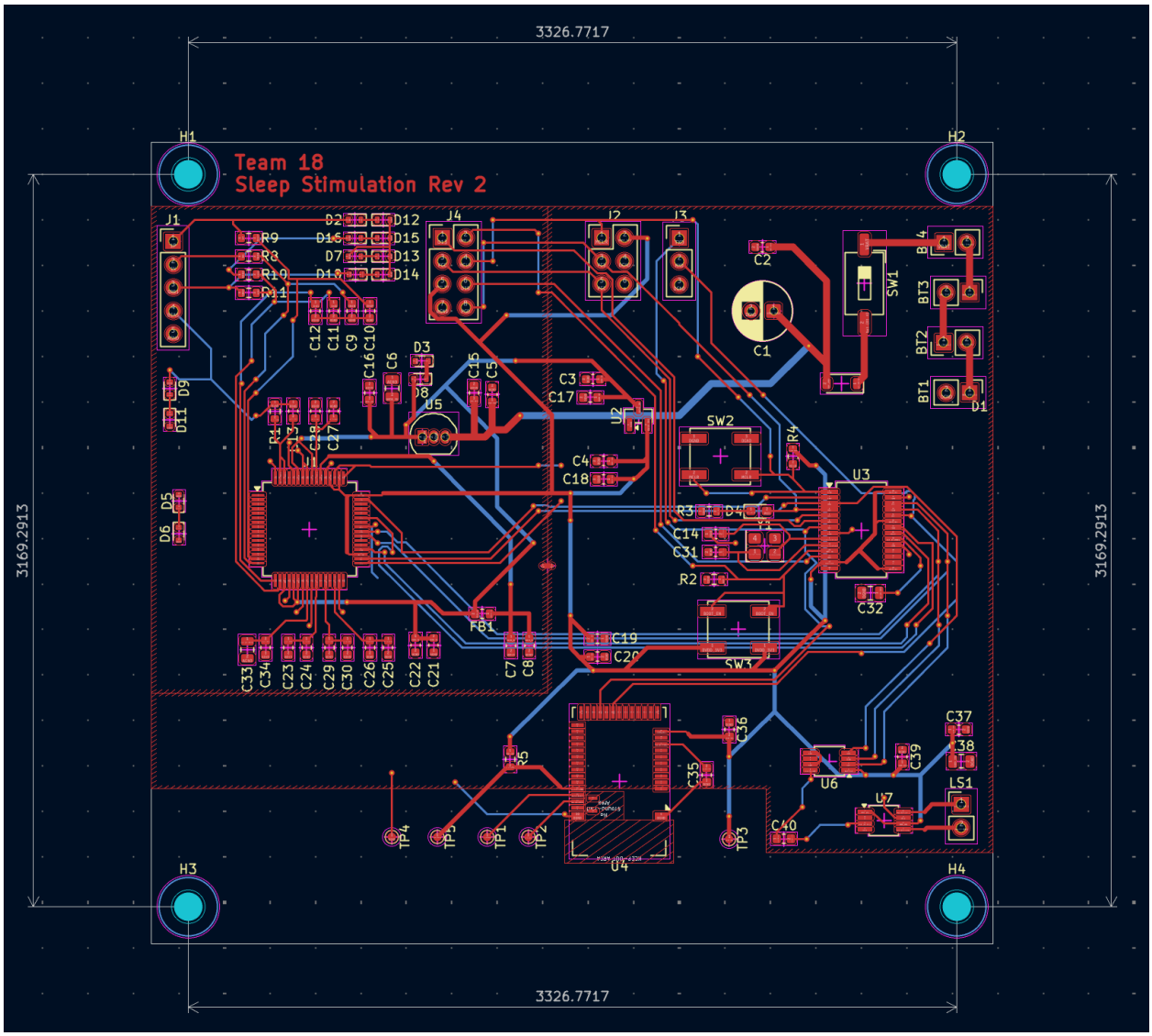

Figure D1: Initial PCB layout

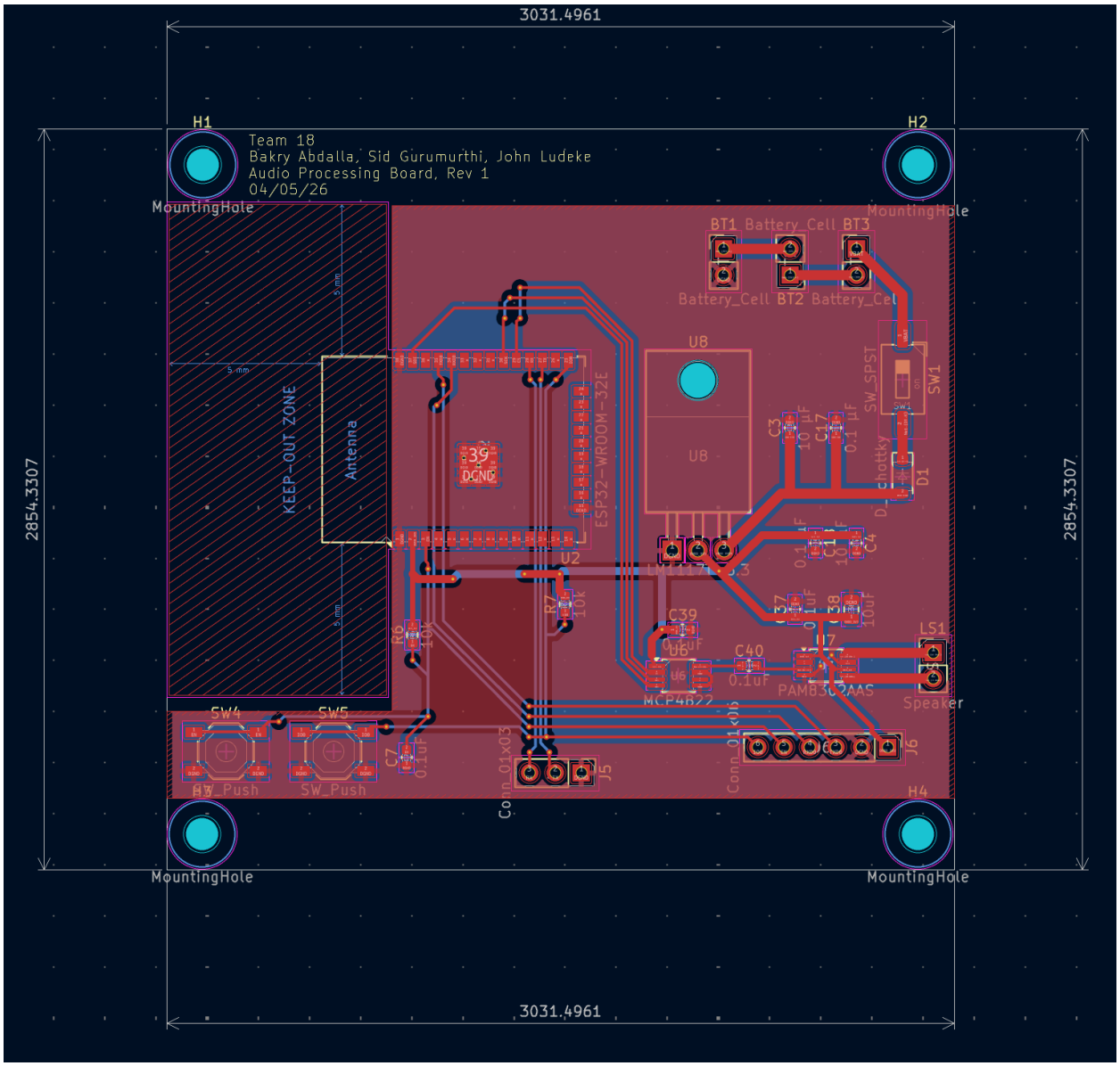

Figure D2: Final PCB Layout

Appendix E – Additional Figures and Plots

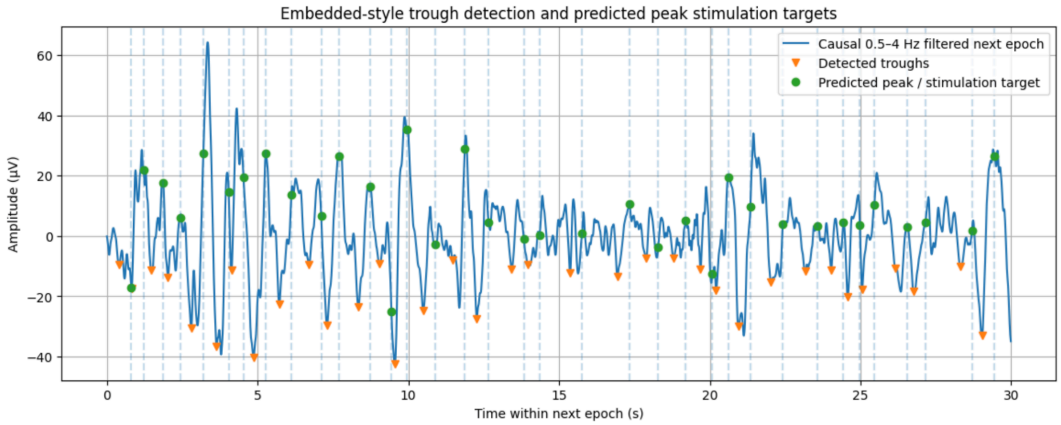

Figure E1: SWS Algorithm Visualization with Trough and Peak Detection

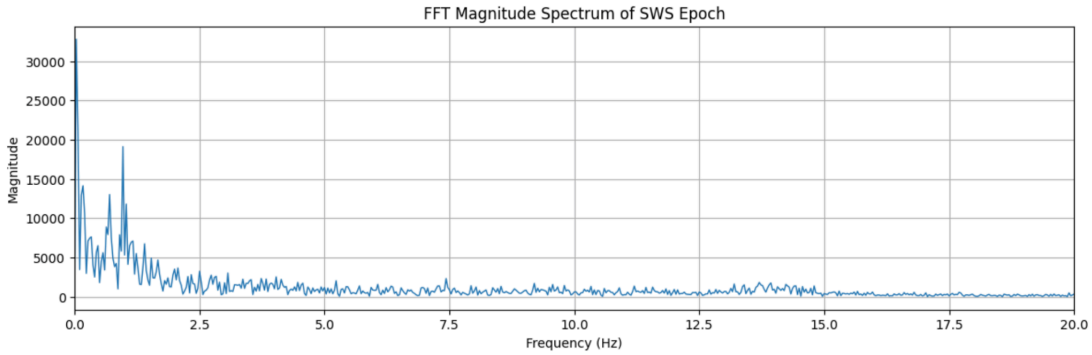

Figure E2: FFT Magnitude Spectrum of a 30 Second SWS Segment

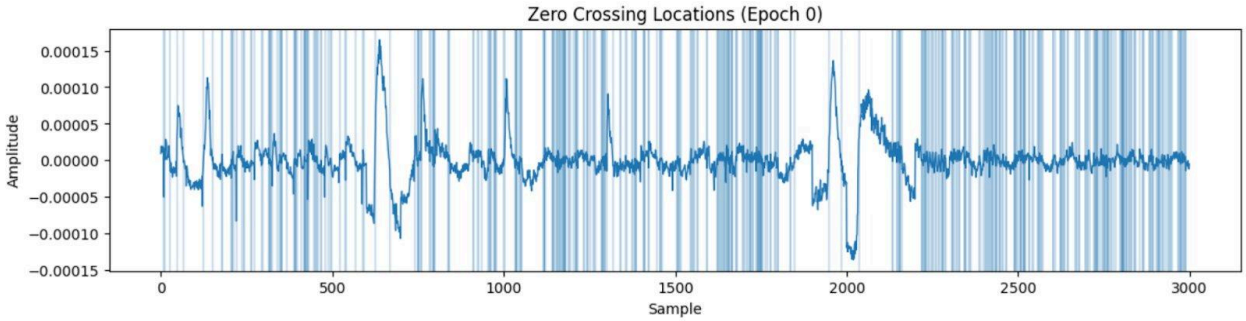

Figure E3: Zero Crossing Locations of Example EEG Data

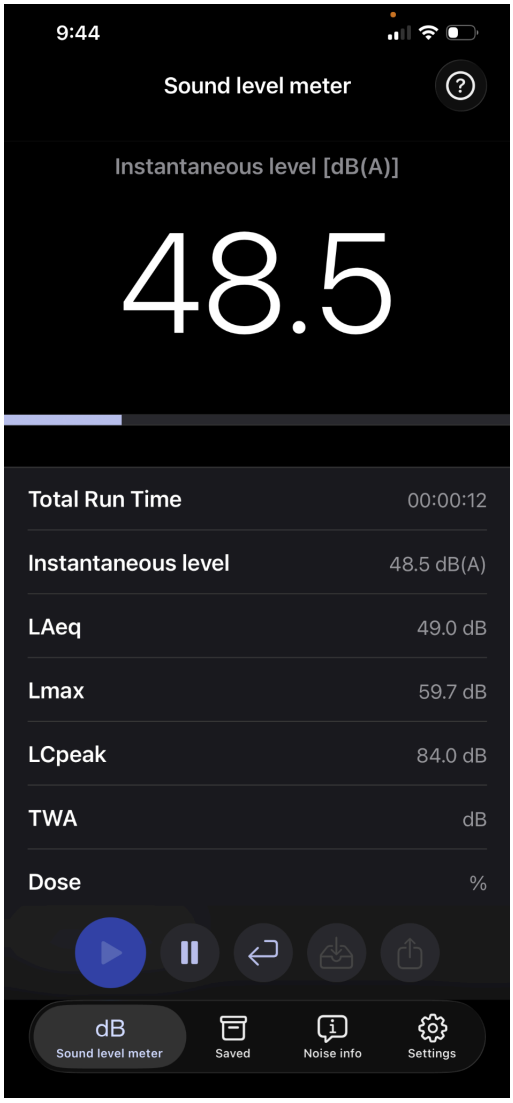

Figure E4: Speaker Volume Tested Using an SPL meter

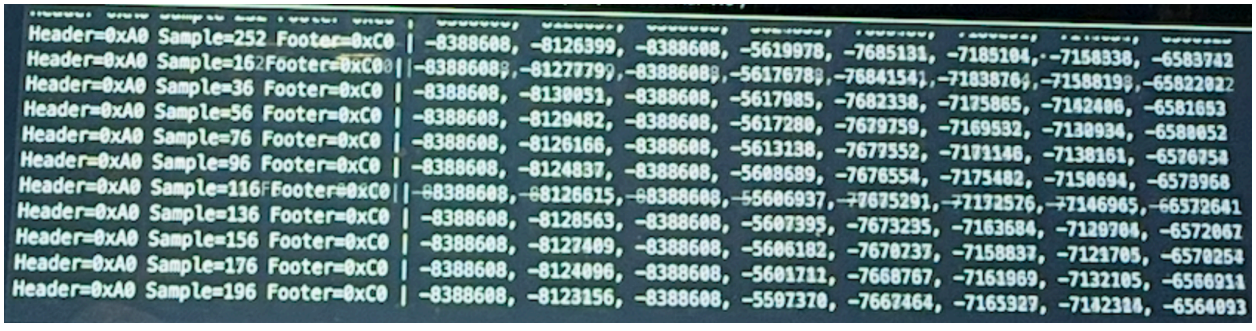

Figure E5. Cyton Packet Parsing and Serial Output Verification

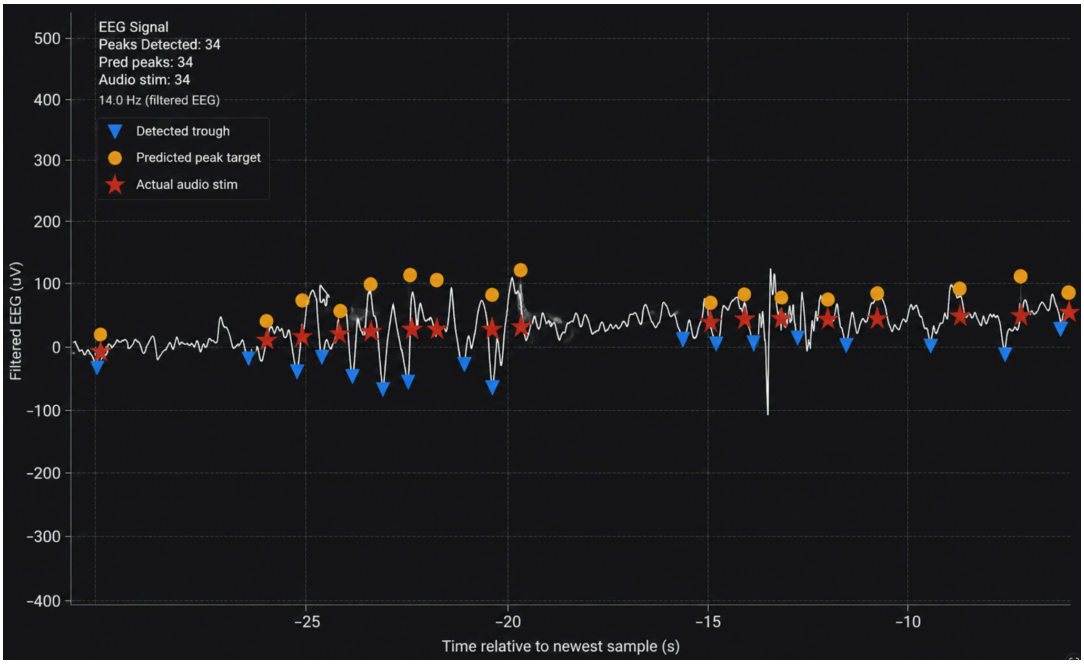

Figure E6. Real-Time EEG Visualization and Stimulation Debugging Interface

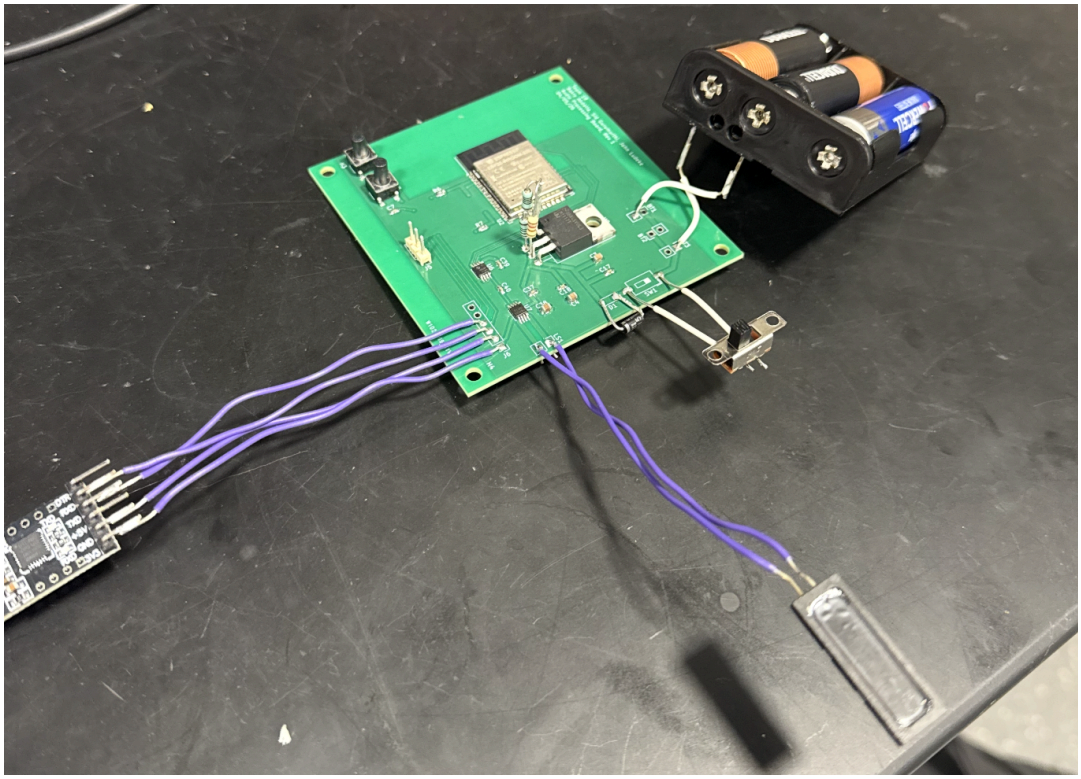

Figure E7. Final Audio and Power PCB