Slow Wave Sleep Enhancement System Design Review Document

By

Aidan Stahl

Kavin Bharathi

Vikram Chakravarthi

Final Report for ECE 445, Senior Design, Spring 2026

TA: Hossein Ataee

12 February 2026

Project No. 36

Contents

1. Introduction

Many common neurological conditions like Alzheimer's disease, depression, and memory issues are associated with patients receiving lower quality sleep. Specifically, these issues often stem from a lack of a specific type of sleep known as slow wave sleep (SWS). As individuals age, sleep disorders and other sleep-related issues lead to a lack of overall sleep. As a result, the amount of time an individual spends in SWS and the quality of SWS they experience typically declines with age, contributing to many of the issues mentioned above. Our team is trying to improve sleep quality using a wearable device that is non-invasive and cost effective. This device will record EEG waves and then detect when the user is in Slow Wave Sleep (SWS) using the aid of specialized software. Once the user enters SWS, the system emits carefully timed bursts of pink noise through an auditory interface to enhance slow wave activity and extend its duration. The team that we're working with, Team 05 - Acoustic Stimulation to Improve Sleep, has cited studies that back up the effectiveness of pink noise bursts during SWS in improving overall memory.

1.1 High Level Requirements

- Pink noise should play within 300 ms of detecting slow wave sleep.
- The average comfort rating of the headset should be a ⅘.
- The entire design should be able to support 10 hours of consecutive sleep, meaning the battery should last at least 10 hours.

1.2 Visual Aid

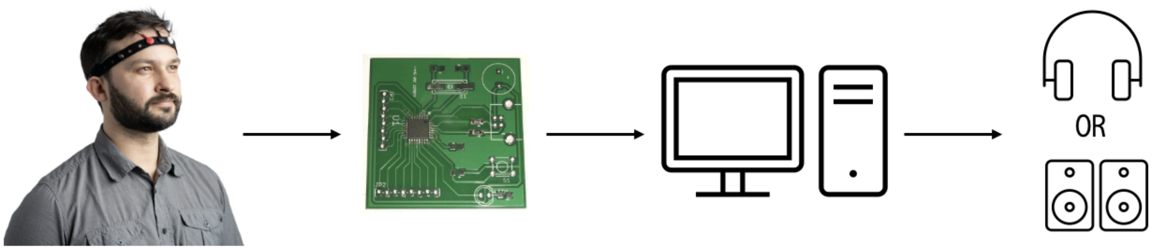

Figure 1: Visual Aid

2. Design

2.1 Block Diagram

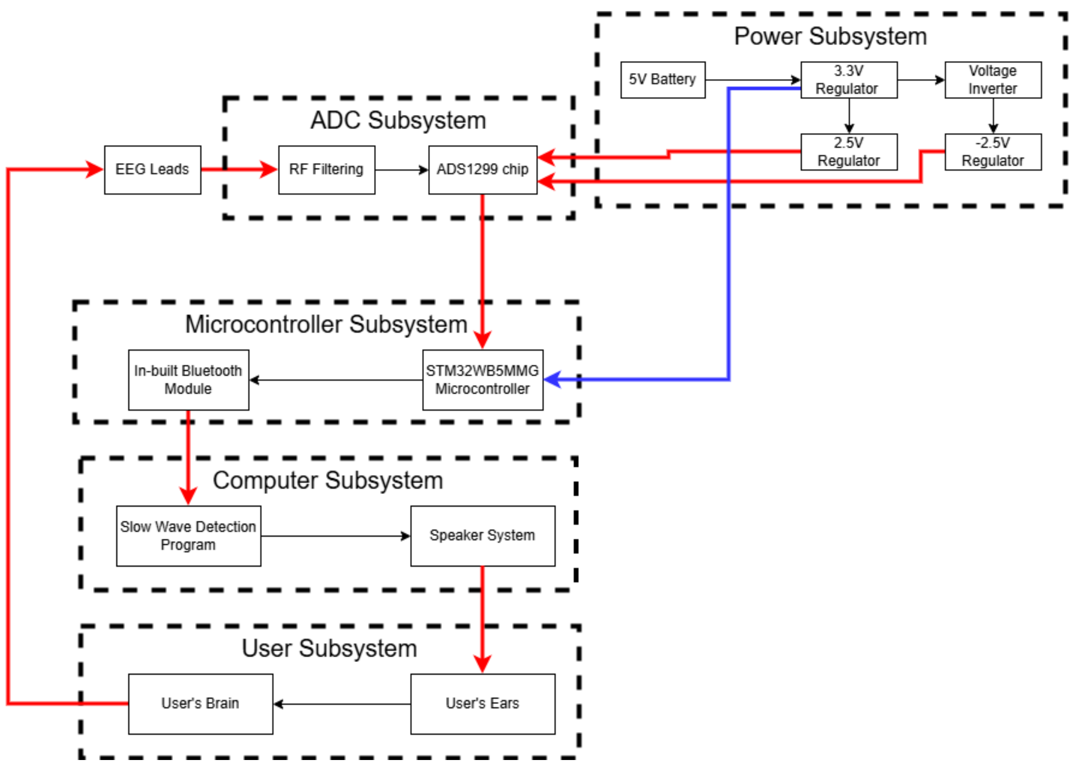

Figure 2: Block Diagram

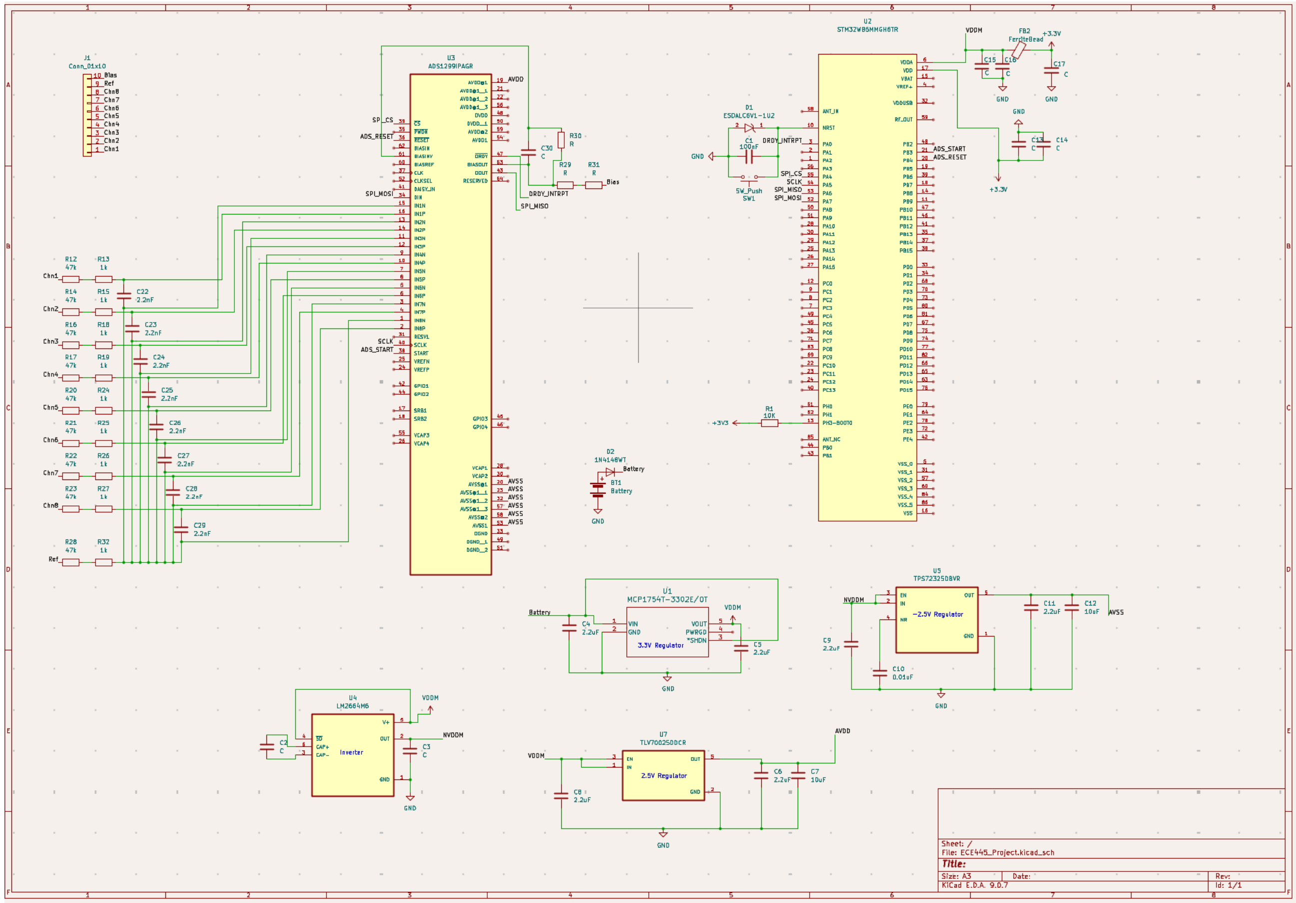

Figure 3: KiCAD Schematic for PCB components

2.2 Power Subsystem

To power our PCB, we used a power subsystem specifically designed to power the ADC and MCU. The MCU we used (STM32WB5MMGH6TR) accepts a power input voltage between 1.71 and 3.6 volts with a typical voltage of 3.3V. Based on these specifications, we chose to use a 6 volt lithium-ion battery to power a 3.3V voltage regulator. The 3.3V regulator is connected directly to the Vdd pin of the MCU. The ADC we used (ADS1299IPAGR) also utilized this 3.3V signal to power the digital logic (DVDD), which accepts a voltage between 1.8 and 3.6 volts. Additionally, we used -2.5V and 2.5V regulators to power the analog power supply pins of the ADC, providing a total range of 5 volts of input. The 2.5V regulator is powered by the 3.3V regulator. Additionally, we passed the 3.3V signal through an inverter to power the -2.5V voltage regulator with a -3.3V signal.

Requirements	Verifications
• Microcontroller and ADC must receive between 1.8V and 3.6V (ideally around 3.3V)	• Power the regulators with a 6V lithium-ion battery and measure the voltage received at the Vdd input pin of the MCU and the DVDD pins of the ADC. Confirm that the values are around 3.3V and within the required range of 1.8V and 3.6V.
• The analog positive power supply pins (AVDD) must receive a voltage between 2.475V and 2.625V (ideally around 2.5V).	• Power the regulators with a 6V lithium-ion battery and measure the voltage received at all AVDD pins of the ADC. Confirm that the values are around 2.5V and within the required range of 2.475V and 2.625V.
• The analog negative power supply pins (AVSS) must receive a voltage between -2.475V and -2.625V (ideally around -2.5V).	• Power the regulators with a 6V lithium-ion battery and measure the voltage received at all AVSS pins of the ADC. Confirm that the values are around -2.5V and within the required range of -2.475V and -2.625V.

Table 1: Power Subsystem Requirements and Verifications

2.3 ADC Subsystem

This subsystem includes both RF filtering and Analog-to-Digital Signal Conversion. RF filtering contributes by conditioning signals so only valid EEG frequencies reach the EEG. The electrode leads that are being used from the headset could potentially act as antennas and pick up radio-frequency signals from the Bluetooth and WiFi activity we are using to send our EEG signals. In order to filter these higher frequency signals out, we designed a low-pass filter consisting of a 1k Ohm resistor and 2.2nF capacitor. We chose these values based on researching the best components to filter out much higher frequency signals. With these components, using the following equation, we were able to find our cutoff frequency fc.

$$f_c = \frac{1}{2\pi RC} = \frac{1}{2\pi \cdot 1000 \cdot 2.2 \cdot 10^{-9}} = 72,343.2\ Hz$$

Since the cutoff frequency represents the frequency corresponding to an output power of -3 dB or ½ relative to the input power, we decided that this was an ideal cutoff frequency for a number of reasons. First, the typical frequencies of EEG leads are around 100-200 Hz at a maximum. This cutoff frequency is high enough where we won't be at risk of accidentally filtering out signals we want and the original EEG signals at this frequency will be maintained. Additionally, we expect much of our noise to result from the microcontroller's bluetooth module. Bluetooth transmission operates in the 2.4 GHz range, which is much higher than our cutoff frequency. This ensures almost all of the signals occurring at the bluetooth frequency range are filtered out. Based on the following calculations using the transfer function, we were able to determine that at bluetooth's frequency of 2.4 GHz, the input voltage seen at the beginning of the filtering circuit will realize an output voltage of around 3e-5*Vin or -90.416 dB relative to the input voltage.

$$|H(\omega)| = \frac{Vout}{Vin} = \frac{1}{\sqrt{(2.2 \cdot 10^{-6})^2 (2 \cdot \pi \cdot f)^2 + 1}} = \frac{1}{\sqrt{(2.2 \cdot 10^{-6})^2 (2 \cdot \pi \cdot 2.4 \cdot 10^{9})^2 + 1}} = 3.014 \cdot 10^{-5}$$

In the dB range this is:

$$20 log_{10}(3.014 \cdot 10^{-5}) = -90.4163\ dB$$

So, after completing these calculations, we are confident in the values we chose for the components of our low-pass filter. In addition, the low-pass filter will be used to remove higher frequency noise potentially from muscle activity and other sources that could potentially be recorded. This also prevents possible aliasing. The sampling frequency used by the ADC is finite and recording higher frequencies could cause aliasing if the Nyquist criterion is not satisfied. These components are used to ensure that the signals received at the ADC are as accurate as possible. Removing this would affect the sleep detection because this filters out the frequencies above 100Hz which would cause aliasing.

The Analog-to-Digital Converter contributes by converting conditioned analog EEG signals into synchronized digital samples. This is one of the most important parts of the PCB. The ADS1299 is an analog to digital converter (ADC) which has an eight-channels and a 24-bit resolution. It also contains programmable gain instrumentation amplifiers, a multiplexer, simultaneous delta-sigma ADCs, internal references generation, and digital filtering. The device is optimized for biomedical applications for microvolt measurements especially in the case of low-amplitude EEG signals. Specific to EEG signal measurements, the advantages are that the ADS1299 has 24-bit delta-sigma conversion and programmable gain amplifiers (PGA) which are low-noise. In delta sigma analog-to-digital-conversion, there is an oversampling modulator followed by a digital filter that produces a high resolution output. The oversampling and specific noise shaping techniques used in delta-sigma ensure that quantization noise is pushed out to higher frequencies which are not within the small bandwidth (of about 0-100 Hz for the EEG we are interested in). The digital filter removes the out of band noise. This in turn increases the resolution and the signal-to-noise ratio which is ideal for an application such as measuring EEG where the values are in microvolts and frequency bandwidth small in between 0-100 Hz. The PGA can be set to gains of 1, 2, 4, 8, 12, 2. This means that external amplifiers for the microvolt EEG voltages are not needed. Additionally, the channels are sampled simultaneously so as to maintain phase alignment so that it is more precise for applications such as working with brain activity and other medical applications. This ensures that there is no misalignment in the channels which could give inaccurate readings of EEG. This subsystem is essential as without it the analog signals will not be able to be converted to digital and thereby preventing any further processing to detect the slow wave sleep stage.

Requirements	Verifications
• At low-frequency inputs, the ADS must output the correct digitized measured voltage values within a standard error of ±1%.	• Use a signal generator to generate a low-frequency signal in the range of 1-100 Hz with a very low voltage and ensure the ADC's digitized output corresponds with the voltage set on the signal generator within a ±1% standard error. • Test with multiple frequencies in the range of 1-100 Hz and multiple voltages to ensure accuracy remains.
• High-frequency signals, such as bluetooth's 2.4 GHz frequency, should be filtered out.	• Use a signal generator to generate a high-frequency signal in the range of 2-3 GHz with a low voltage similar to what an expected EEG voltage would be. Ensure the ADC's digitized output is very close to zero with at least a suppression of - 90 dB or greater (for frequencies >= 2.4 GHz). • Test with multiple frequencies in the range of 2-3 GHz and multiple voltages to ensure signals remain suppressed.
• The sampling frequency should be greater than or equal to 200 Hz meaning the Nyquist frequency is at least 100 Hz, which ensures that there is no aliasing for signals less than or equal to 100 Hz.	• Use a signal generator to generate a sinusoidal wave. Test inputs at various frequencies from 10 Hz to 100 Hz and ensure the digitized output is the correct frequency within a ±1% standard error. • Plot the ADS's output while changing the voltage at a constant frequency and make sure that the output graph

	matches the frequency and voltages expected with a standard error of ±1. ○ Test with several frequencies to ensure accuracy is maintained.

Table 2: ADC Subsystem Requirements and Verifications

2.4 Microcontroller Subsystem

The STM32 Microcontroller receives the ADC samples from the ADC through the SPI interface. The role of the microcontroller is to package the multi-channel samples into Bluetooth packets which are timestamped and sent to the host computer. It is also responsible for configuring the ADS1299 registers, manages the start-up configuration, and maintains the sampling synchronization. The BLE protocol provides a low-power transmission which is continuous and therefore suitable for an overnight application such as this one.

The main reason behind choosing this particular STM32 microcontroller is that it is low-powered and also has in-built bluetooth communication. It has an Arm Cortex-M4 core up to 64 MHz, dedicated Cortex-M0+ CPU for radio communication, and integrated BLE 5.x radio. Importantly, the duo core architecture separates the ADC sampling and the radio communication.

In the SPI communication, there will be 8 EEG channels with 24-bit resolution for each channel. Let us set a sampling rate of 250 Hz, with 24 status bits each which in total is 24 status bits + 24 bits x 8 channels = 216 bits or 27 bytes per sample frame. Therefore, the data rate will be 27 bytes x 250 Hz = 6750 bytes/sec = 54 kbps. The sampling rate as established before is 200 Hz which is $\frac{1}{250}$ = 0.004 s = 4 ms. This means that the DRDY pin (Data Ready pin) is seen every 4 ms. If for example, a 8 MHz SPI clock is chosen on the STM32, then the time taken to shift 216 bits is $\frac{216}{8 \cdot 10^6}$ = 27 µs. So, the sampling period of 4 ms is greater than the SPI read time. Thus, the

CPU dedicated to ADC sampling does not have much burden thus allowing low power consumption. Additionally, the dual core nature would help in the Bluetooth and ADC acquisition and packetization to not interfere with each others' processes. Bluetooth supports 2 Mbps. As calculated earlier, the data rate is 54 kbps. This is well below the supported 2 Mbps.

For calculating the estimated latency, let us consider all of the delays. The ADC sampling delay is 4 ms. SPI transfer is 27 µs. Adding any MCU overhead and additional buffers, let us assume 0.3 ms overall. Assuming any additional delays of 15 ms, the overall approximated latency would be 4 + 0.027 + 0.3 + 15 = 19.157 ms. This is well within the 300 ms target latency and leaves sufficient room for the detection algorithm.

Requirements	**Verifications**
● Able to see digital signals corresponding to EEG on a receiver with sampling rate of 250 Hz and resolution of 24 bits.	● Use a signal generator to generate a specific frequency within the range of 1-100 Hz. Set the voltage at a low range, similar to what would be expected from an EEG lead. ● Check amplitude and received signals to see if they are the same as the signal from our SG within a ±1% standard error. ● Confirm that the sampling rate is within ±1% standard error of actual sampling rate of 250 Hz. ● Validate that amplitude corresponds to gain scaling. ● If Bluetooth communication is not working, try to see if EEG signals are received through wired UART connection. ● Test with multiple different frequencies from the SG within a range of 1-100 Hz and change the amplitudes as well to simulate representing multiple different voltage peaks.

• SPI communication works as expected at 8 MHz with no missed DRDY interrupts, no bit errors.	• Check if the digital information at the DRDY pin corresponds to the EEG received on the microcontroller output. • Check if proper MISO/MOSI transitions are made, SPI clock frequency matches. • Make sure that DRDY pins toggle at the correct interrupt times.
• Bluetooth operates with latency below 100 ms with packet loss rate less than 2% and throughput of 54 kbps.	• Timestamp the transmission and reception to compute actual latency. • Measure packet loss by streaming known data streams for a certain fixed time.

Table 3: Microcontroller Subsystem Requirements and Verifications

2.5 Computer Subsystem

This Slow Wave Detection Program contributes by transferring real-time EEG data to the host computer via Bluetooth. Uses an open source Command-line sleep analysis tool called YASA (Yet Another Spindle Algorithm) to read digitized EEG data and determine if the user is experiencing slow-wave sleep. We aim to use a lightweight Machine Learning algorithm to classify and detect EEG. Keep it lightweight for it to be deployable on mobile applications. Our Python script plays pink noise from the laptop that the sleep analysis tool is running on. The processing will run simultaneously to the data acquisition to allow real-time detection and response. If interrupts or transmissions fail, gaps occur and sleep detection cannot function.

The speaker system component contributes by playing an audio file with "pink noise" once slow-wave sleep is detected. Headphones or a speaker connected to the laptop's audiojack or via bluetooth play the pink noise MP3 file. If playback delay is excessive, the stimulation becomes ineffective.

The YASA algorithm allows for real-time slow-wave sleep identification. The tool is designed to detect several different cycles of sleep, including slow-wave sleep. Slow-wave sleep typically occurs when the sleep waves are delta waves in the range of 1 - 4 Hz, but can be less than 1 Hz and up to 15 Hz. The YASA algorithm will help us identify when the EEG waves are in this range. Since our project is designed to play pink noise quickly after the up-state of the wave, while the user is in slow-wave sleep we must identify when the up-state of the SWS wave occurs. Identifying the up-state by the preceding negative peak from the down-state makes the most sense, since the waves being analyzed by the computer subsystem are already delayed from when they were first detected by the EEG leads. Playing the noise as soon as a negative peak is detected allows us to play the pink noise as close as possible to the real-time up-state of the SWS wave of the user's brain. The YASA algorithm's available tools include negative and positive peak detection, so we will use this to detect a negative peak of the SWS wave and implement in Python additional software to play a specific MP3 file with pink noise after detection.

Requirements	**Verifications**
• Slow-wave sleep is detected correctly.	• Use sample EEG data to test if the algorithm is successful at identifying slow-wave sleep. If the algorithm identifies slow-wave sleep during the provided slow-wave sleep sample data and successfully identifies other sleep cycles or not sleeping, when non-slow-wave sleep sample data is provided, the algorithm is working as intended.
• Pink noise is played within 150 ms or less after the up-state of a slow-wave sleep wave is correctly identified.	• Use sample EEG data to test if the software correctly identifies when a slow-wave sleep up-state is occurring by making the program print a line

	indicating the up-state wave has been detected and look at the EEG sample data at this specific time to determine if an up-state is truly occurring. ● After the up-state of the wave has correctly been identified, make sure the pink noise MP3 file is played within 150 ms of this detection.

Table 4: Computer Subsystem Requirements and Verifications

2.6 User Subsystem

This subsystem represents the end goal of the system. The user hears pink noise while experiencing Slow-wave sleep and reaps its benefits. If any previous subsystem fails, the user will not receive stimulation during slow-wave sleep. It is the basis of our feedback loop, receiving pink noise while continuously supplying data.

Requirements	Verifications
● If given a survey to multiple users for how comfortable the device is, the average rating should be 4 out of 5. (1 is least comfortable, 5 is most comfortable)	● We will just have multiple users test the finished product ● Each user will complete an anonymous survey about their experience ● One of the metrics will be the comfortability of wearing the headset while sleeping

Table 5: User Subsystem Requirements and Verifications

2.7 Tolerance Analysis

Tolerance Analysis for this project can be conducted by taking into consideration the latency, power, and noise requirements of each physical component.

Meeting High-Level Requirements:

- Latency of each component DIRECTLY affects whether or not we meet the high level requirement of playing pink noise within 300 ms of detecting slow wave sleep.
- Understanding how much power each component draws DIRECTLY affects whether or not our project meets the 10 hour battery life requirement.
- Minimizing noise for EEG signals ensures ACCURATE identification of slow wave sleep

ADS1299 Signal Noise Tolerance Analysis:

- Input-Referred Noise: 1 μV_{PP}
- Input Voltage (EEG): 10 - 100 μV_{PP}
- Worst Case Signal to Noise Ratio: $SNR = 20 \cdot log_{10}(\frac{V_{in}}{V_{noise}}) = 20 \cdot log_{10}(\frac{10}{1}) = 20dB$

The worst case SNR shows that the EEG voltage will always be at least 10x larger than input noise for the ADS1299. Signal noise contributes minimally to amplitude and timing uncertainty.

ADS1299 Latency Analysis:

- Typical Data Rate for EEG Data (Digital Data Produced per Second): 250 SPS (Samples per Second)

$t_{dr} = \frac{1\ second}{250\ samples} = 4\ ms$

When the ADS1299 is given a signal to start sampling data, it takes $3 \cdot t_{dr}$ seconds to settle and give its first digital data output

Worst case delay from the ADC: $t_{ADC,\,worst} = 3 \cdot t_{dr} = 12\ ms$

$t_{ADC,\,worst} = 12\ ms$

Since we are sampling continuously, we can experience delays lower than 12ms. Worst case will stay at 12 ms.

ADS1299 Power Analysis:

- 5 mW per channel (8 channels ≈ 40mW)
- Analog Supply Current: $I_{AVDD} = 7.14\ mA$
- Digital Supply Current: $I_{DVDD} = 1\ mA$
- Current drawn from ADC: $I_{total} = I_{AVDD} + I_{DVDD} = 8.14\ mA$

STM32WB5MMG Power Analysis:

- Current Consumption when running: $I_{run} = 5000\ \mu A$
- Total Estimated Current drawn = 5 mA

Estimated required Battery Capacity =

$(I_{ADC} + I_{STM}) \cdot Battery\ Life = (8.14\ mA + 5\ mA) \cdot 10\ hr = 13.14\ mA \cdot 10\ hr = 131.4 mAh$

We know that our battery capacity is above 131.4 mAh, so 10 hour battery life is a reasonable target.

Bluetooth Low Energy Latency Analysis:

- The BLE module in the STM32 microcontroller DOES NOT transmit data continuously
- Periodic connection events are spaced out by Connection Intervals
- Range for Connection Interval time: $7.5\ ms \le T_{CI} \le 4\ sec$
- We can set the T_{CI} to our desired value when programming the microcontroller

- Lower T_{CI} results in lower latency but higher power consumption
- Higher T_{CI} results in higher latency but lower power consumption

3. Cost and Schedule

3.1 Cost Analysis

Part	Manufacturer	Retail Cost ($)	Bulk Purchase Cost ($)	Actual Cost ($)
Analog to Digital Converter (ADS1299IPAG (8-ch AFE, TQFP)) x 1	Texas Instruments	73.30	65.151	73.30
Microcontroller (STM32WB5MMG H6TR (BLE module)) x 2	STMicroelectronics	12.14	9.53	24.28
MCP1754 (3.3V regulator) x 1	Microchip	0.54	0.46	0.54
LM2664 (Voltage Inverter) x 1	Texas Instruments	1.08	0.776	1.08
TLV70025DDCR (2.5V Regulator) x 1	Texas Instruments	0.23	0.13760	0.23
TPS723 (-2.5V Regulator) x 1	Texas Instruments	3.45	2.03	3.45
Total				

Table 6: Cost Analysis/Bill of Materials

Team Member	Hours of Work	Hourly Rate ($)	Total ($)
Kavin	10	16.00	160.00
Vikram	10	16.00	160.00
Aidan	10	16.00	160.00
Total	10	16.00	480.00

Table 7: Weekly Labor Costs

3.2 Schedule

Week	Task	Person
3/2/2026	Finish Wiring PCB and passing DRC by 3/2	Aidan/Everyone
	Design Review with Yang Zhou and Hossein Ataee by 3/4	Everyone
	Incorporate wiring changes for PCB if needed after Design review and send in first order for PCB (expected arrival by March 19) by 3/5	Vikram/Everyone
	Get a working breadboard demo with demonstrable EEG	Everyone
3/9/2026	Arrival of PCB and begin soldering of parts by 3/12	Kavin/Everyone
3/16/2026	Spring break	
3/23/2026	Finish soldering PCB and testing for electrical correctness by 3/25	Everyone
	If there are any changes to be made to the PCB design, reflect changes on KiCAD and order a new PCB.	Aidan/Everyone

3/30/2026	Test PCB and test software to see correct identification of Slow Wave Sleep.	Vikram/Everyone
	Begin development of software detection algorithms.	Kavin/Everyone
4/6/2026	Make sure EEG signals are readable and corresponding to test data. Make sure that demo is read y	Everyone
4/13/2026	Test PCB and test software to see correct identification of Slow Wave Sleep.	Everyone
4/20/2026	Continue testing PCB and software and explore the possibility of developing an application if time permits.	Everyone
4/27/2026	Mock demo week	Everyone
5/4/2026	Final demo week	Everyone

Table 8: Schedule

4. Discussion of Societal Impact, Engineering Standards, Ethics, and Safety Considerations

The Societal impact of the Slow Wave Sleep Detector is improving widespread treatment for sleep and mental disorders. It has been proven through the studies discussed in this document that pink noise has many beneficial effects for improving sleep quality. The restorative outcomes from improved sleep are essential to improving widespread treatment. The non-invasive nature of this project allows it to be applied across a large group of users freely.

This project comes with several ethical considerations that we must take into account when designing. First, we must add voltage surge protection connected to the EEG output leads because if the voltage were to surge, the user could be harmed. Another potential ethical concern is patient information. Our device is going to be used in a medical/healthcare application. HIPPA cites established guidelines for protecting patient information that must be followed. Any use of patient information in our development and/or testing stages must abide by these guidelines. These safety and ethical guidelines are further enforced by following IEEE and ACM standards. More specifically, this project follows the IEEE Code of Ethics Principles 1 and 3, and aligns with the Association for Computing Machinery Code of Ethics Principles 1.2, 1.6, and 2.5. IEEE Principle 1 prioritizes the health, safety, and welfare of the public through ensuring that the EEG headband is non-invasive and ensuring that safe levels of pink noise are played. IEEE Principle 3 emphasizes being transparent with how effective this device is for sleep-enhancement. ACM Code of Ethics is followed by securing healthcare data according to widely accepted standards.

Electrical and mechanical safety have been considered in this project. We made sure that the system follows through with low-voltage battery operation. This allowed for an easier time to create circuit protection that will also protect circuit components. To address mechanical safety, the PCB will be situated in a protective, 3D printed enclosure. The purpose of this enclosure is to prevent accidental contact with outside environments that could damage solder joints or traces.

References

[1] *Hong-Viet V. Ngo, Thomas Martinetz, Jan Born, Matthias Mölle, Auditory Closed-Loop Stimulation of the Sleep Slow Oscillation Enhances Memory, Neuron, Volume 78, Issue 3, 2013, Pages 545-553, ISSN 0896-6273, https://doi.org/10.1016/j.neuron.2013.03.006.*

[2] *H.V. Ngo, & B.P. Staresina, Shaping overnight consolidation via slow-oscillation closed-loop targeted memory reactivation, Proc. Natl. Acad. Sci. U.S.A. 119 (44) e2123428119, https://doi.org/10.1073/pnas.2123428119 (2022).*

[3] *Marina Wunderlin, Marc A Züst, Elisabeth Hertenstein, Kristoffer D Fehér, Carlotta L Schneider, Stefan Klöppel, Christoph Nissen, Modulating overnight memory consolidation by acoustic stimulation during slow-wave sleep: a systematic review and meta-analysis, Sleep, Volume 44, Issue 7, July 2021, zsaa296, https://doi-org.proxy2.library.illinois.edu/10.1093/sleep/zsaa296*

[4] *"Cyton Data Format." OpenBCI Documentation, OpenBCI, 16 July 2025, docs.openbci.com/Cyton/CytonDataFormat/.*

[5] *Vallat, R., & Walker, M. P. (2021). An open-source, high-performance tool for automated sleep staging. eLife, 10. https://doi.org/10.7554/eLife.70092*

[6] *OpenBCI. (n.d.). OpenBCI EEG Headband Kit [Product page]. OpenBCI Shop. Retrieved Month Day, Year, from https://shop.openbci.com/products/openbci-eeg-headband-kit*

[7] https://www.build-electronic-circuits.com/printed-circuit-board-guide-beginners/ (image used from website)

[8] U.S. Department of Health and Human Services. (2025, March 14). *Summary of the HIPAA privacy rule*. HHS.gov; U.S. Department of Health and Human Services. https://www.hhs.gov/hipaa/for-professionals/privacy/laws-regulations/index.html

[9] Bluetooth SIG. *Part B — Link Layer Specification*, Core Specification (BLE), Section "connInterval". https://www.bluetooth.com/wp-content/uploads/Files/Specification/HTML/Core-54/out/en/low-energy-controller/link-layer-specification.html?utm_source=chatgpt.com.

[10] STMicroelectronics. (2024, October). *STM32WB55xx STM32WB35xx: Multiprotocol wireless 32-bit MCU Arm-based Cortex-M4 with FPU, Bluetooth 5.4 and 802.15.4 radio solution* (Datasheet DS11929 Rev 17). https://www.st.com/resource/en/datasheet/stm32wb55cc.pdf

[11] Garcia-Rill, E. (2009). Reticular Activating System. In L. R. Squire (Ed.), Encyclopedia of Neuroscience (pp. 137–143). doi:10.1016/B978-008045046-9.01767-8