Low-Cost Bone Conduction Hearing Aid Development Platform

ECE4012 Senior Design Project

Group sd20p04, The EnginHears
Project Advisor, Dr. David Anderson

Lillian Anderson, Electrical Engineer, landerson65@gatech.edu
Cynthia Baseman, Electrical Engineer, cbaseman3@gatech.edu
Bradley Blaho, Electrical Engineer, bblaho@gatech.edu
Brooke Brennan, Computer Engineer, brookebrennan@gatech.edu
Caitlin Choate, Computer Engineer, cchoate6@gatech.edu
Aaron Green, Electrical Engineer, agreen77@gatech.edu
John Landers, Electrical Engineer, john.landers@gatech.edu
Megha Tippur, Electrical Engineer, mtippur3@gatech.edu

Submitted

2020 April 29

Table of Contents

Table of Figures

Table of Tables

Executive Summary

This project intended to develop a low cost, socially destigmatized hearing aid design. Despite hearing loss being so prominent, the average cost of hearing aids is $2,300 per unit, putting hearing aids out of reach of many low- to middle-income families [1]. In addition, many people who need and can afford hearing aids choose not to buy them, as their bulky aesthetic increases social stigma [2]. To combat these current market limitations, the EnginHears developed a Low Cost Bone Conduction (LCBC) Hearing Aid Development Platform.

The hardware system of the LCBC consists of battery powered acoustic sensing and acoustic driving stages. The sensing and driving stages provide acoustic frequency filtering, and contribute appropriate gain for signal processing and transducer driving. All the hardware components are designed to be housed within a custom designed headband case, while the signal processing hardware, including DSP development board, are designed to be housed within a custom designed belt compartment. The sum of all hardware components is currently limited to $31.38 when purchased in bulk, which closely aligns with our goal of around $30.

The software system architecture is designed to consist of real-time acoustic signal sampling, filtering, and driving stages within a DSP development board. Filtering shapes the frequency response of incoming signals, tuning the relative gain of acoustic frequency bands independently to compensate for frequency dependent hearing loss.

The final hardware deliverables include system schematics, circuit simulation results, 3D CAD models of a headband case and belt compartment, and PCB layout files. Final software deliverables include code for execution on a DSP evaluation board as well as batch processing and real-time software simulations demonstrating filtering capabilities. Finally, documentation substantiating design choices, a video explaining system design choices, and a video demonstrating system performance is also submitted.

Nomenclature:

ABS - Acrylonitrile Butadiene Styrene

A/D - Analog to Digital Converter

ADC- Analog to Digital Converter

ANSI - American National Standards Institute

BAHA - Bone Anchored Hearing Aid

BPF - Bandpass Filter

BTE - Behind The Ear

CFR - Code of Federal Regulations

CIC - Completely In the Canal

COVID-19 - Disease caused by SARS-CoV-2 coronavirus

D/A - Digital to Analog Converter

DAC - Digital to Analog Converter

dB - Decibels

DMA - Direct Memory Access

DSP - Digital Signal Processing/Processor

FCC - Federal Communications Commission

FDA - Food & Drug Administration

HL - Hearing Level

Hz - Hertz

IEC - International Electronics Commission

ITE - In The Ear

kg – kilograms

LCBC – Low-cost Bone Conduction

mm - millimeters

ms - milliseconds

N - Newtons

OTC - Over The Counter

PERT - Program Evaluation Review Technique

PLA - Polylactic Acid

PSAP - Personal Sound Amplification

PTA - Pure Tone Average

SAR - Specific Absorption Rate

SNR - Signal to Noise Ratio

SPL - Sound Pressure Level

WHO - World Health Organization

The Low Cost Bone Conduction Hearing Aid Development Platform

1. Introduction

Team EnginHears spent $926.30 to develop the Low-Cost Bone Conduction (LCBC) Hearing Aid prototype to combat age-related hearing loss. The LCBC hearing aid has been designed and partially prototyped even with the constraints arising from the COVID-19 outbreak.

1.1 Objective

The objective of this project was to design and model the LCBC Hearing Aid design for those with age-related hearing loss. This design uses analog audio amplification and DSP algorithms to create filters that amplify desired frequencies and increase speech clarity. The team has tested these algorithms in their environments. The device's design is discreet and comfortable to minimize social stigma and maximize the user's experience. The final form factor of the hearing aid, which is 3D modeled in SolidWorks, is a self-contained unit that does not need to be linked to any other devices. It was desired that the product could be easily charged using standard cables. The user should only need to charge the device overnight or at most once during the day. Because the end user of this product is someone with general age-related hearing loss, this bone conduction hearing aid is not designed for individuals with total hearing loss or unique hearing problems, although it may still offer relief in these cases. This device is designed to be used in one's daily life when the user is engaged in a conversation or listening to a speaker and will help the user better comprehend speech.

1.2 Motivation

Despite the prominence of hearing loss, the average cost of hearing aids is $2,300 per unit, placing hearing aid technology out of reach for many individuals in low-to-middle income countries [1]. The primary focus of this project is to decrease the cost of hearing aid technology, so it is more accessible to these lower income countries. A set goal price of roughly $30 would make this technology more attainable. Social stigma is the second focus of the project, given that many individuals who could benefit from hearing aid technology refrain from an aid purchase because of the device's bulky aesthetic, which increases social stigma [2]. A bone conduction hearing aid with headset form factor that emulates the design of bone conduction headphones will make the hearing aid technology less noticeable, minimizing social stigma associated with hearing loss.

1.3 Background

Bone conduction hearing devices use vibration to send signals directly to the inner ear, making them ideal for those who suffer hearing loss due to middle or outer ear damage. Some current solutions that use bone conduction may require surgical implants, which are both costly and invasive. The global technology company MED-El offers a bone conduction hearing aid that does not require surgery, but it is marketed more towards children who are too young for surgical procedures. Nonetheless, their devices show that there is a market for less invasive bone conduction hearing aids. Additionally, there are other companies like Aftershokz that have further modernized bone conduction listening devices with a design that resembles headphones similar to the design proposed in **Figure 1**. These devices range in price and functionality, but the cheapest start at $60, which is more expensive than our proposed solution.

2. Project Description and Goals

The goal of the LCBC hearing aid development platform was to design an inexpensive, easily manipulated, discreet, and aesthetically appealing hearing aid design platform to lower-income individuals. The proposed hearing aid design consists of the following components:

- Analog Hardware
    - 2 Microphones
    - 2 Bone Conduction Transducers
    - 2 Instrumentation Amplifiers
    - 2 Low Pass Filters (microphone to microcontroller interface)
    - 2 Band Pass Filters (microcontroller to bone conduction transducer interface)
    - 1 Rechargeable Battery
- Cases
    - Headband Case
    - Belt Case
- Human Interfacing Hardware
    - 2 pushbuttons for digital volume control**
    - 1 pushbutton for on/off switch**
- Digital Hardware
    - 1 Microcontroller
    - 2 Class D Amplifiers
    - 1 Battery Charger

** Future goals: feature of form factor design, but not included in schematic

A microphone captures all incoming acoustic energy for signal processing and amplification. Utilizing two microphones would also allow for increased directionality through beamforming, reducing ambient noise. The final design is for the microcontroller to sample the audio frequency data from the microphones and apply filtering to compensate for age-related hearing loss, based on a general age-related hearing loss audiogram. The microcontroller then outputs the digitally filtered signal to an analog BPF. The output signal of the analog BPF then propagates to a class D amplifier, then to the bone-conduction transducers. Shown in **Figure 1**, all the electrical components are designed to be housed inside a custom-built case, which resembles the Aftershokz bone conduction headphones. This hearing aid was designed to be self-contained and not need interfacing with any other products.

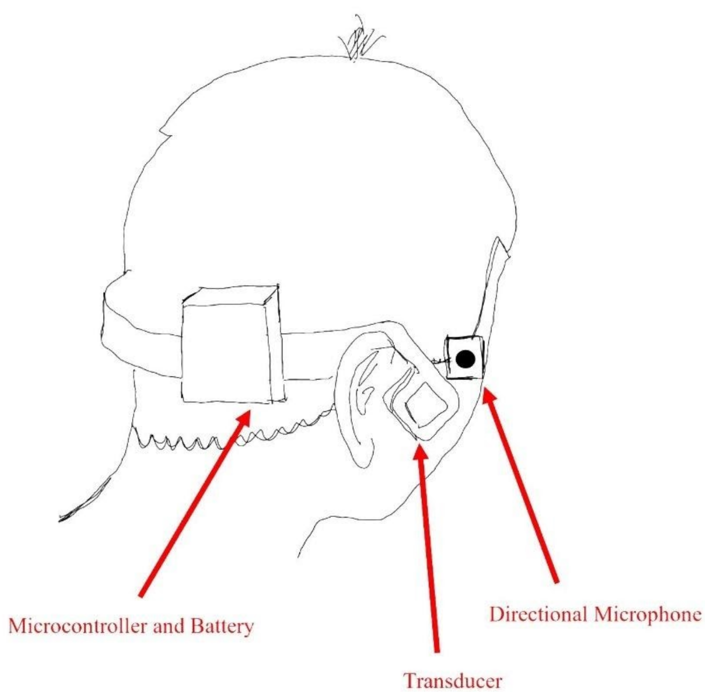


Figure 1: Anticipated Design Prototype

User considerations and satisfaction were a top priority when designing this product. Because this device was designed to be used in day-to-day life, it was important that all of the user requirements were addressed. The top customer needs and requirements for the device are shown in the QFD diagram in **Figure 2**. The final product is designed to be discreet and comfortable with a long battery life. The customer should be able to use the device easily and comfortably without worrying about the technical details. This is what would mark the greatest customer satisfaction. The important part of design is achieving these customer requirements while also meeting all technical requirements. It was desired that the device function well with a low time and long battery life while also being small and lightweight. **Figure 2** below outlines the relationships between these customer and engineering requirements.

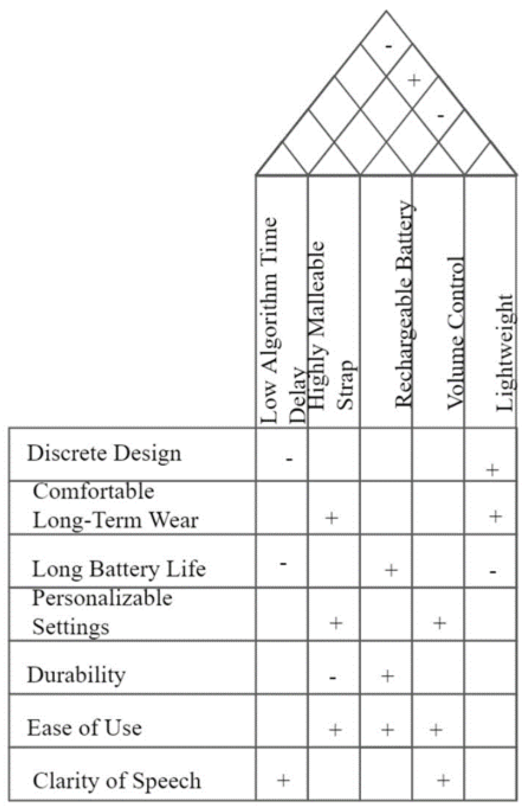


Figure 2: Original QFD Diagram

This hearing aid was designed to be self-contained and does not need to interface with any other products. There are not any constraints on materials or manufacturing processes. The design is workable and can be adjusted depending on what manufacturing works best.

The goals and requirements of the LCBC Hearing Aid were adjusted given the current situation involving COVID-19. The "Lightweight" technical requirement has been removed as a current priority from **Figure 2**, as a completed prototype of the LCBC hearing aid is no longer feasible. Additionally, many of the customer requirements from **Figure 2** are no longer verifiable given that a completed prototype is not realistically attainable. Specifically "Durability," "Ease of use," "Clarity of Speech," and "Comfortable Long-Term Wear" cannot be tested without a prototype. Compliance with the remaining customer requirements: "Discreet Design," "Long Battery Life," and "Personalizable Settings," can be estimated without a prototype, so these requirements will continue to be tracked:

3. Technical Specifications & Verification

Table 1 shows the technical specifications for both the hardware and software components of the LCBC Hearing Aid.

Table 1. System Specifications

Item	Specification	Measured Value
Analog Hardware		
Supply voltage	3.3 - 5.0 V	3.9V - 5.0V
Battery life	6 - 12 hours	6 - 10 hrs (SPL dependent)
Frequency Response	200 - 10,000 Hz	200 - 10,000 Hz
Sampling rate	20kHz - 25kHz	25 kHz
Analog BPF f_{cu}	10kHz	10 kHz
Analog BPF f_{cl}	200Hz	200 Hz
Algorithm / DSP		
Maximum Delay (ms)	3 - 10ms	10.24 ms
Computation speed	Number of instructions * sample rate < Fclk of dsp device * instructions/cycle	Number of instructions * 25,000 Hz < 200 MHz * instructions/cycle

4. Design Approach and Details

4.1 Design Approach

Significant portions of the originally proposed LCBC design have been prototyped, however the COVID-19 outbreak has forced the team to simulate performance of the remaining designs.

4.1.1 Case Development

A central component of the LCBC hearing aid development platform is a wearable case. The originally proposed case had several important design specifications as discussed in Section 3. More generally speaking, the following requirements were proposed for the case:

1. Must house all required electrical hardware: including a power source, all analog amplification and filters networks, a microcontroller, microphones, and bone conduction transducers
2. Must provide appropriate static coupling force to the head for effective transmission of the mechanical accelerations of bone conduction transducers for sound perception
3. Must be fully wearable
4. Must have a discreet aesthetic, to facilitate a reduction of the social stigma surrounding hearing aid technologies

Case development began with a very rough sketch of the intended design, which was eventually converted into a 3D CAD model in SolidWorks, see **Figure 3** [3]. Throughout the semester, the case design underwent nine iterative updates, primarily to tune dimensions to accommodate electrical components, improve user comfort, as well as to tune the compression force vector.

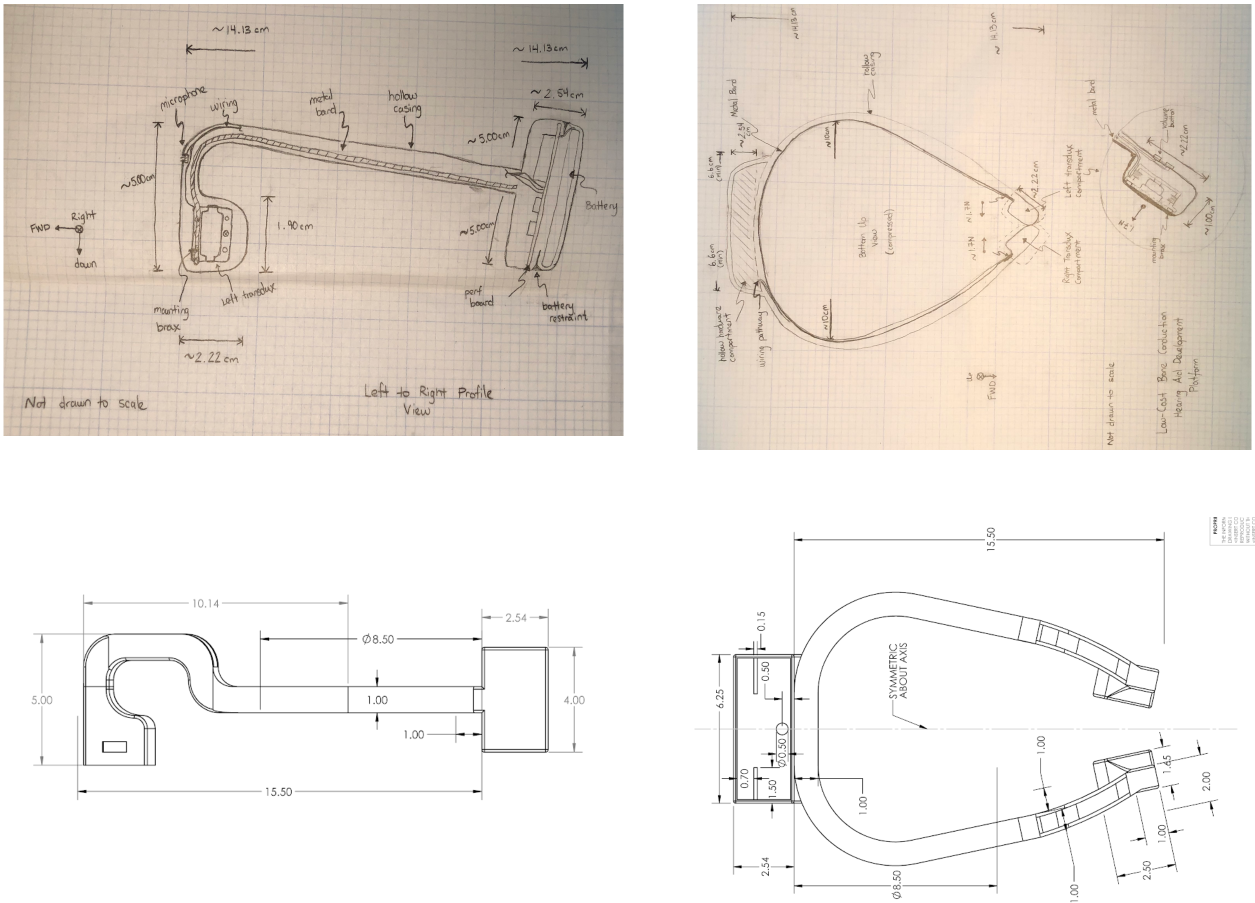


Figure 3. The LCBC case progression.
Case concept sketches: side view (top left) and top view (top right).
Iteration 9 of the combined case: side view (bottom left) and top view (bottom right).

In the interest of modularity and rapid prototyping, case design was subdivided into components, each having relatively simplistic and static interface with mated components to reduce inter-component dependence. In this way, designs for individual components could be updated based on user feedback or dimensioning issues without disturbing the design parameters of other components. Additionally, all components were designed in SolidWorks to facilitate 3D printing. Finally, components were designed to be printed with commercially available 3D print filaments, reducing cost and complexity.

With these constraints, the specifications originally proposed, and the case requirements in mind, iterative development began. Due to a lack of mechanical design experience, the first few iterations were relatively simplistic. After the CAD of each iteration had been completed, it was 3D printed in a rigid material (ABS) and subsequently evaluated for appropriate dimensioning and adequate aesthetics. Updates were recorded and then implemented in the next iteration.

In order to satisfy the requirement that the case be fully wearable in the designs agreed upon by the team, it was determined that the headband case and transducer compartments must be capable of flexibility beyond what is capable with 3D print filaments such as ABS or PLA. After researching 3D print filaments, Gizmo Dorks 2.85mm TPU was selected, and all subsequent prints of the headband case and transducer compartments were printed with this filament [4].

After iteration 5, dimensioning and aesthetic design had stabilized and development focus shifted to generation of the necessary compression force vector for the bone conduction transducers. It was decided that the simplest and most effective solution would be to insert a band inside the headband case. As the arms of the headband case, which are printed in a flexible TPU, deflect outwards, they generate minimal resistance and consequently a minimal compression (restoring) force. However, introduction of a band inside of the headband case that has greater rigidity than the TPU headband case increased resistance to outward deflection and consequently increased the net compression (restoring) force of the headband case. Discussions revolved around 3 potential materials for the band: spring steel, plastic, or rigid 3D print filament such as ABS or PLA. Ultimately, it was decided that the compression band should be 3D printed to facilitate rapid design updating, since many of the parameters of the compression band would be heavily influenced by the parameters of the headband case driving compression band updates after each headband case update. The first iteration of the headband case and compression band (case iteration 6) can be seen in **Figure 4**.

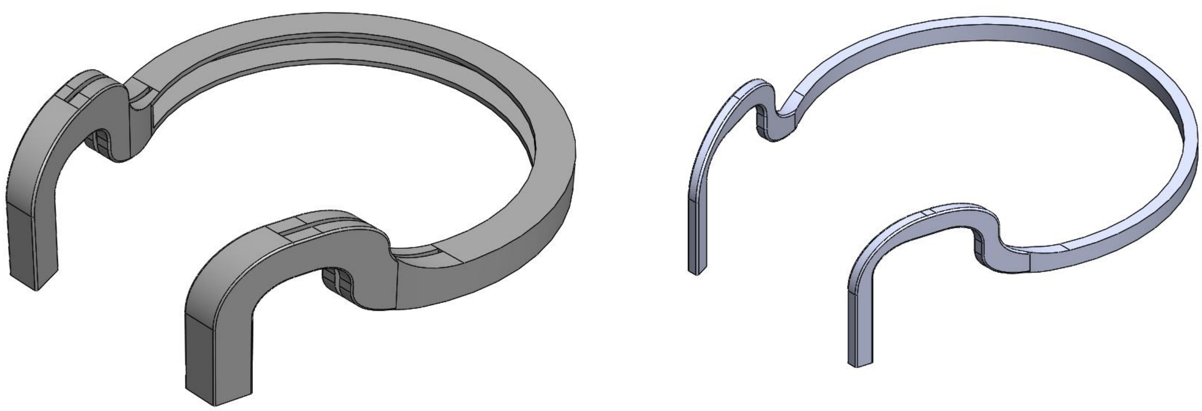


Figure 4. Case iteration 6: 3D CAD model of the headband case in TPU (left) and 3D CAD model of the compression band in ABS (right)

The compression force experienced by the wearer of the LCBC case is the sum of the restoring forces exerted by both the headband case and the compression band. In theory, the absolute restoring force of the composite headband unit can be approximated, as it is a function of design geometries, material properties, anticipated deflection displacement, and other known or calculable parameters. However, calculation of these values is non-trivial, and so the team leveraged the iterative design approach to refine the compression force magnitude.

The final iterations of the headband case focused on tuning the compression force vector. It was discovered after the introduction of the compression band that contact between the users' heads and the headband was not concentrated at the zygomatic arch, instead there was contact along much of the "ear arch" of the headband. This left the bone conduction transducer compartments relatively uncoupled from the users' heads, resulting in ineffective transmission of the mechanical accelerations necessary for sound perception. The final iteration of the headband case, iteration 9, implemented two design updates meant to combat this issue. By introducing a curved contour in the sagittal plane of the "ear arch" portion of the headband case and compression band as seen in **Figure 5**, the case achieves a reduction in the contact area of the headband case assembly. Additionally, the bone conduction transducer compartments were modified to project the transducers inward, further limiting the contact area to the driving surface of the transducer compartments.

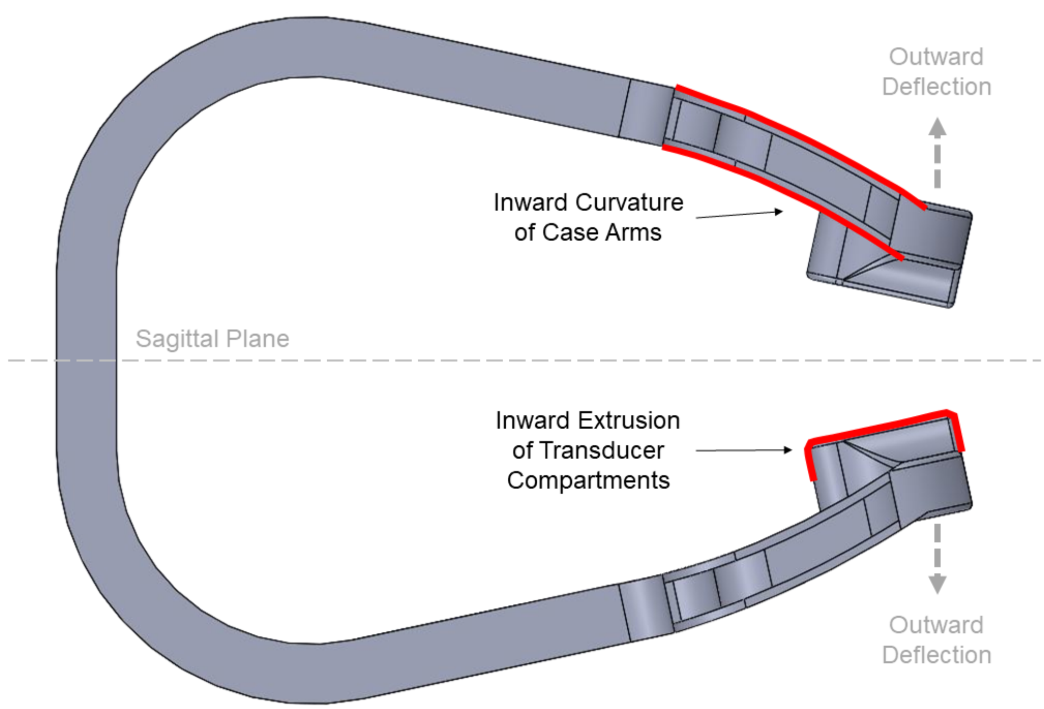


Figure 5. Top-down perspective of iteration 9 of the headband case. Changes are in red.

In addition to headband case development, compartments for holding electrical components were necessary to satisfy the proposed requirements. Original concepts drawings only included one compartment which would be affixed to the back of the headband. An underlying assumption with this design choice was that the dimensions and design feasibility of the compartment would be constrained by the size and weight of the battery. In reality, the primary determinant of the feasibility of this design choice was the microcontroller form factor. The selected prototype DSP development board (TI TMS320F28377S) is comparable in weight to the selected battery (Adafruit 328), and is significantly larger than the selected battery [5, 6]. It was assumed that putting both of these within an electrical compartment affixed to the rear of the headband case would have generated an unacceptable design aesthetic and torque. An alternative solution involved the development of a microcontroller compartment model in SolidWorks, complete with a handle that could be attached to a belt loop. Thus, the design approach was refined to create these two electrical compartments: one exclusively for the TI DSP development board, and another for the analog hardware components and battery, as seen in **Figure 6**. Future work will involve selecting a smaller DSP development board, or integration of all digital and analog hardware on a custom PCB to reduce size and weight, facilitating integration with the headband case, and removing the necessity for the DSP development board compartment.

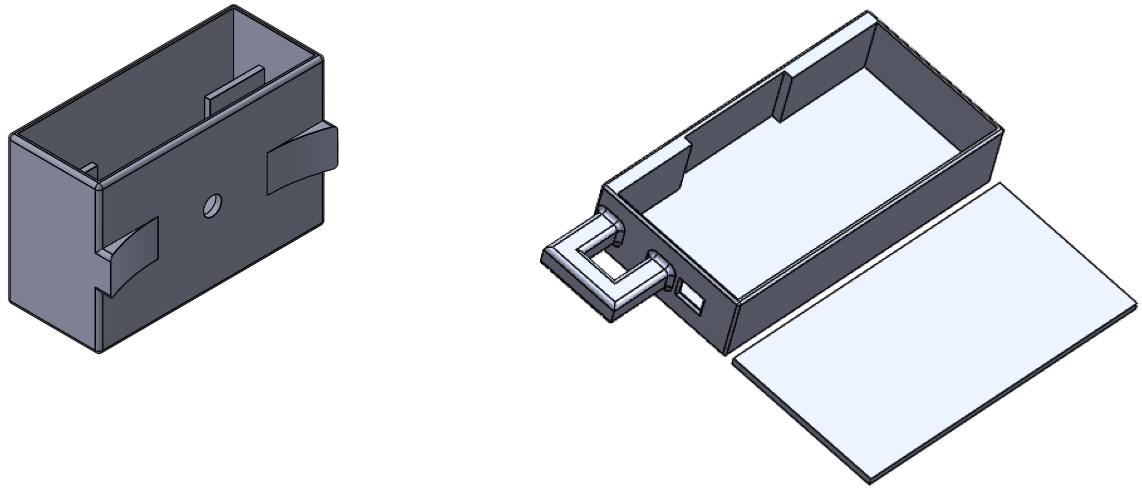


Figure 6. Left: headband electrical compartment; right: DSP development board compartment

4.1.2 Analog System Development

The proposed analog system of the LCBC hearing aid can be divided into two separate stages. The first, a sensing stage, facilitates effective transduction of incident acoustic energy and transfers the resulting electrical signal to a DSP development board. The second, a driving stage, effectively transduces an audio frequency electrical signal output from a DSP development board to mechanical accelerations.

The development of the proposed analog system was also constrained by the specifications addressed in Section 3, bulk cost of required components, and an implicit requirement to minimize system complexity. Size and weight of the majority of the analog components used for prototyping were not priority due to constructive time-management as well as the planned development of a custom PCB with use of surface mountable components.

Analog system development began with research and selection of appropriate transducing components. After discussing microphones with several advisors, the Invensense ICS-40740 and ICS-52000 microphones were selected as potential candidate microphones as they were reasonably priced at under $4/unit and advertised high performance characteristics [7]. The ICS-40740 microphone is an analog differential output MEMS microphone, while the ICS-52000 is a PDM (digital) output MEMS microphone. After discussion with the software team, it was decided that an analog signal input to the DSP development board would be preferable to a PDM signal as sampling PDM signals would require external libraries and sampling rates on the MHz order. Thus the ICS-40740 was selected as the prototype microphone. The Adafruit 1674 transducer advertises a wideband frequency response, is relatively cheap at under $9/unit, and is widely available making it an ideal bone conduction transducer for the prototype LCBC hearing aid [8].

Characterization of the transduction components was the next step in analog system design, as the performance of the microphone and bone conduction transducer would drive amplification and analog filtering parameters.

Microphone sensitivity was the primary variable of interest, as this parameter would determine the necessary gain to utilize the full scale range of the microcontroller ADC. Despite not having precision audio testing equipment, the team utilized several different speaker sources, minimizing irregularities in frequency response of individual speakers, at a relatively constant volume to characterize the output voltage of the ICS-40740 microphone. The results of this experiment, shown in **Figure 7**, were utilized as reference data for determining gain values. The team noted that variations over frequency are likely not characteristic of the microphone's actual performance as speaker directionality varies with frequency and acoustic driver parameters. The "ICS Data Sheet Converted" trace represents a scaled differential output voltage based upon the frequency response advertised by the ICS-40740 datasheet [7].

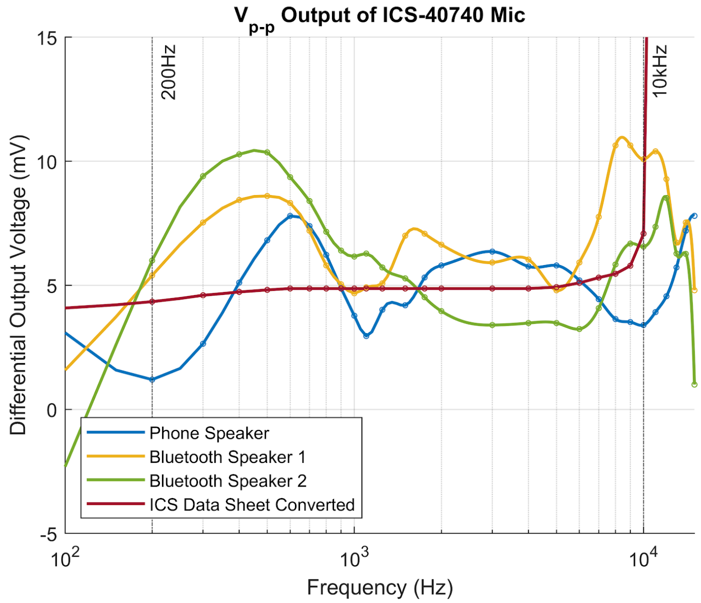


Figure 7. Differential output voltage of ICS-40740 vs. frequency

Frequency response of the microphone was also of interest, as this would theoretically drive both analog and digital filter parameters; however, confounding variables due to COVID-19 prevented further characterization of the microphone frequency response.

The primary characteristics of interest for the bone conduction transducer were frequency response and sensitivity. Due to time management and a lack of publicly available information, these characteristics were not heavily investigated. **Table 2** contains the specifications publicly available for the Adafruit 1674 bone conduction transducers [8].

Table 2. Adafruit 1674 advertised specifications

Specification	Value
Impedance	8 Ω
Frequency Response Bandwidth	300 - 19,000 Hz
Resonance Frequency (f_r)	1.6 kHz

In order to better understand the magnitude response of the bone conduction transducer the team created an experimental setup where a signal of known constant V_{pk-pk} was injected into the input of a class D amplifier (Adafruit PAM8302A) driving a bone conduction transducer [9]. The bone conduction transducer was glued to an accelerometer (ADXL355) [10]. Due to limitations with the accelerometer, the sampling frequency was set at 3kHz, effectively limiting the characterizable bandwidth to 1.5kHz. All data was put through a bandpass filter in order to reduce aliasing effects and high frequency noise content. The full experimental setup is shown in **Figure 8**, while **Figure 9** shows the frequency response of the transducer assuming the Class D amplifier exhibits a relatively linear response across the frequency region of interest.

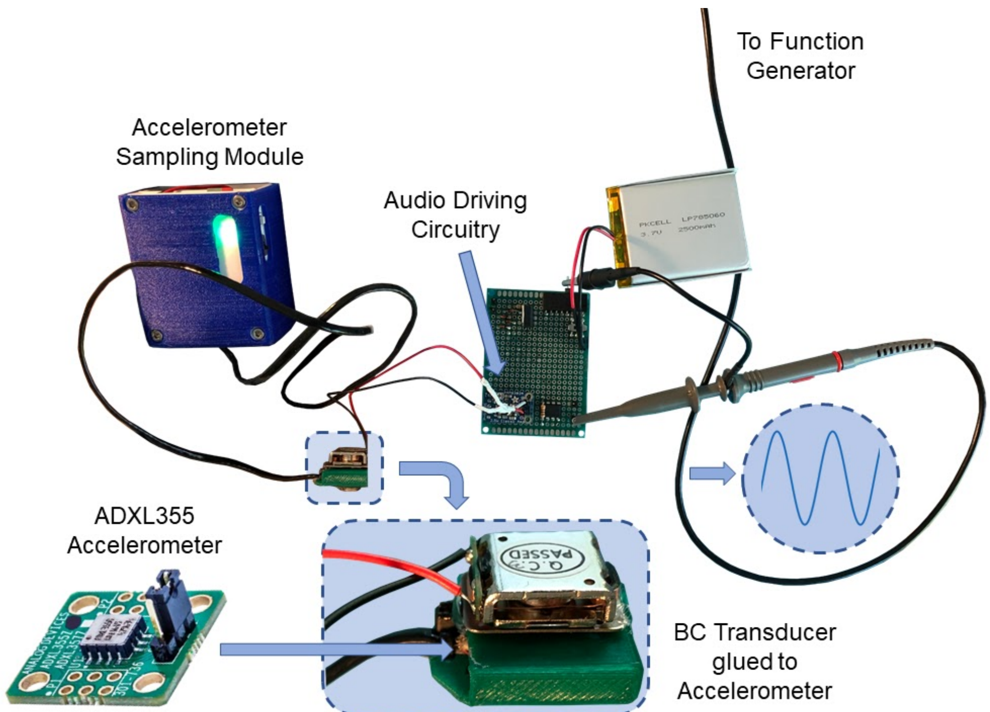


Figure 8. Experimental setup for measuring output acceleration magnitude of the Adafruit 1674 bone conduction transducer.

An important consideration of any characterization experiment of a bone conduction transducer is the mechanical impedance of the test rig as this will contribute a non-trivial impact on the acceleration magnitude response. Effective transfer of mechanical energy can only result with system coupling between the actuator and the sensor. Thus, the total system magnitude response is a function of both the actuator (bone conduction transducer) and the sensor (the human head in this case) mechanical parameters. Ideally the mechanical impedance of the accelerometer package would approach the mechanical impedance of the human skull, however, creating a mechanical model of the human skull was not prioritized during system development as this is outside of our expertise. Consequently, the magnitude response plotted in **Figure 9** should be used only for reference information on general low frequency trends of the Adafruit 1674.

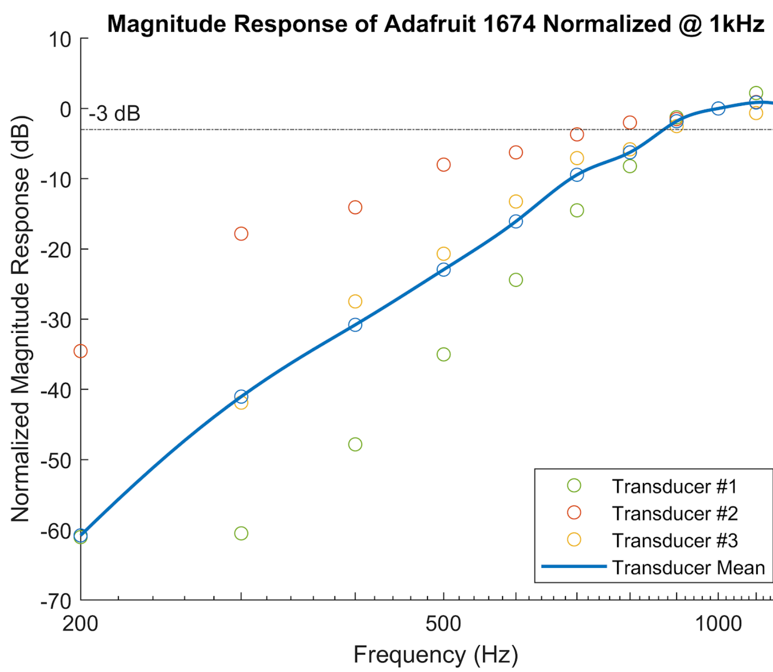


Figure 9. Acceleration magnitude response of Adafruit 1674 bone conduction transducer normalized at 1kHz

Complete characterization of the transduction components did not occur as development deadlines dictated priorities. However, output magnitude averages gleaned from the ICS-40740 experiment, datasheet parameters, and the ADC characteristics were used to determine an appropriate gain value for the analog system front end. Ultimately, it was determined that the advertised sensitivity of ICS-40740 and the full scale ADC range of the selected DSP development board (TI TMS320F2837xS) would be the primary constraining variables for the calculation of the ideal gain.

$$Sensitivity_{@1kHz,\,94dB\,SPL} = -37.5dB$$
$$V_{full-scale} = 3.0V$$
$$\alpha = \frac{V_{full-scale}}{V_{diff-out}} = \frac{3}{10^{-37.5/20}} = 225 = 47dB$$

At this stage of development, the team began to construct analog system topologies that would achieve the specifications listed in **Table 1**, as well as minimize both system cost and complexity. The first revision of the system block diagram is shown in **Figure 10**.

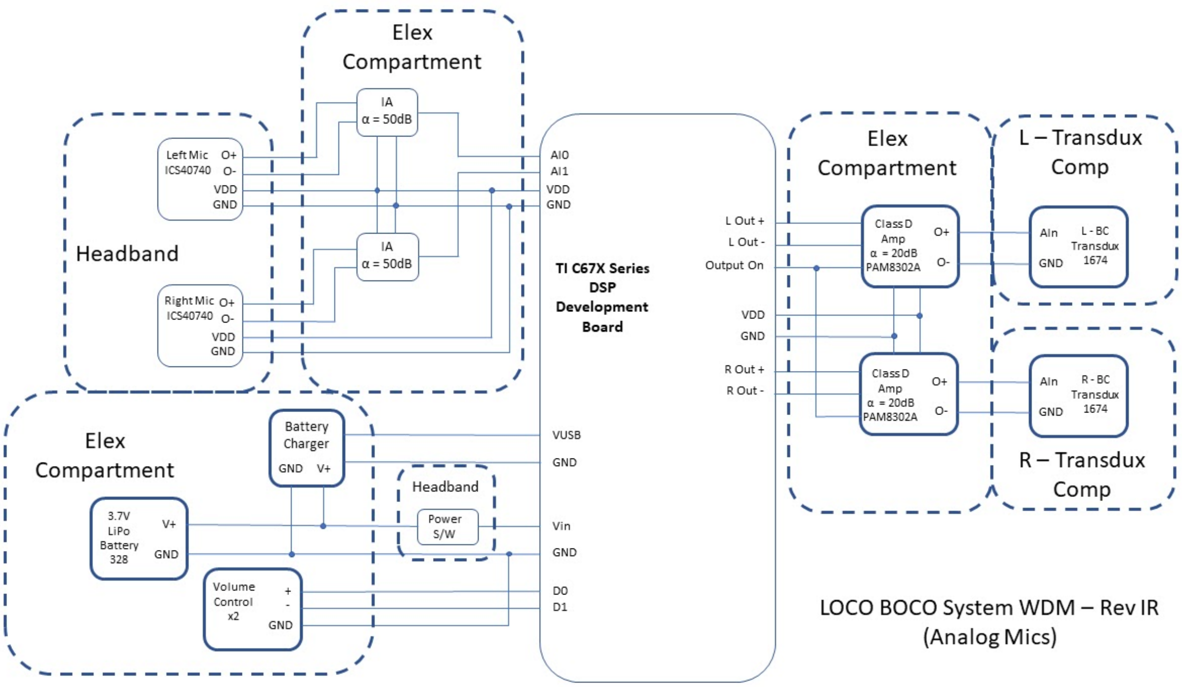


Figure 10. Rev IR of the system level block diagram for the LCBC analog circuit.

As the output of the ICS-40740 microphones was a differential analog voltage, a high gain low noise instrumentation amplifier was required to achieve the specified gain value. Because the LCBC Hearing Aid is battery powered, the amplifier also had to be low voltage and single-supply. The AD623AN instrumentation amplifier was selected as it met these requirements, was relatively low cost at under $6/unit and had a highly configurable gain parameter ($\alpha = 1 - 1000$) [11].

The next stage of analog system development involved designing a network to effectively transfer the audio frequency electrical signal within the sensing and driving stages. It was also decided that the total number of amplifiers should be minimized to reduce cost and complexity, and consequently, all filters would be passive. As the ADC of the selected microcontroller had a input voltage range of $0 - 3.0V$, the sensing stage had to maintain appropriate DC offset of ideally $1.5V$. Thus, the sensing stage analog network was designed to include only a LPF and to utilize the REF input of the AD623AN amplifier to introduce the DC offset. The output of the microcontroller's DAC was also DC offset, and likely contained quantization noise as well as any other noise introduced by the microcontroller filtering functions. Additionally, the input of the selected class D amplifier (Adafruit PAM8302A) accepted a differential analog voltage. Thus a BPF, to reduce noise by constraining bandwidth, was added to the system architecture.

Another critical addition to the system architecture was a microphone bias network. It was quickly determined that the ICS-40740 microphones were extremely sensitive to voltages over 3.3V and thus were incompatible with the selected battery voltage of 4.0V. These updates can be seen in revision B of the system block diagram in **Figure 11**.

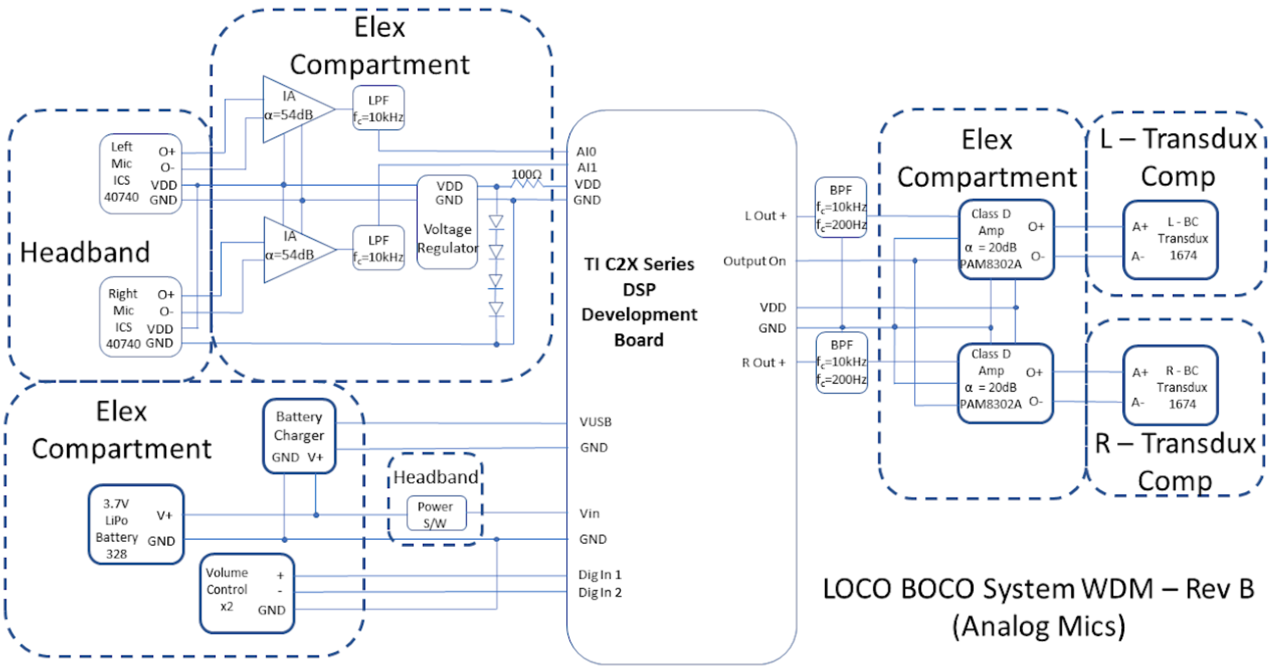
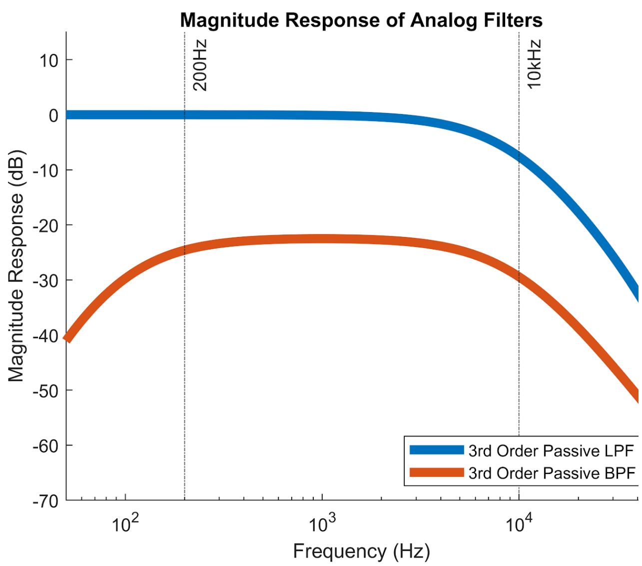


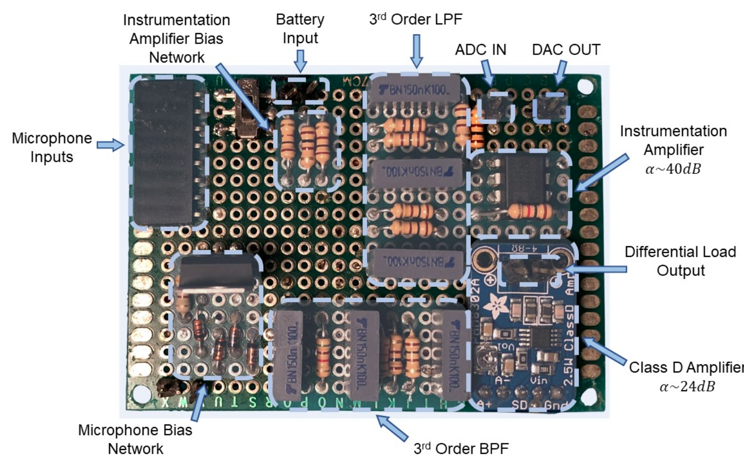
Figure 11. Revision B of the LCBC analog system block diagram.

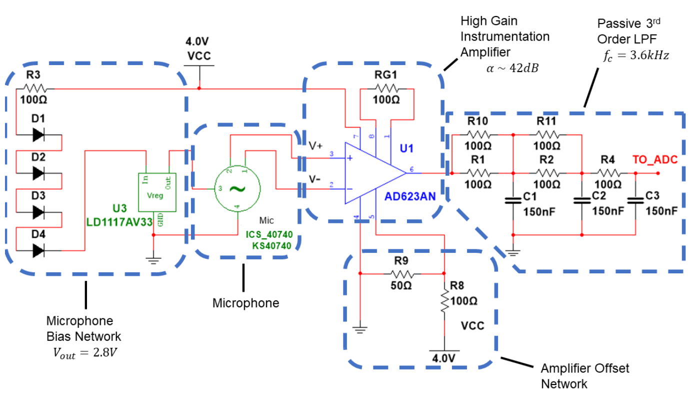
At this stage in analog system development, the team focused on developing a miniaturized prototype of the proposed system architecture for integration with the case. The first stage of this process was implementation of the currently proposed circuit on a perf board. Ultimately though, a custom fabricated PCB would be designed in order facilitate mass production and miniaturization.

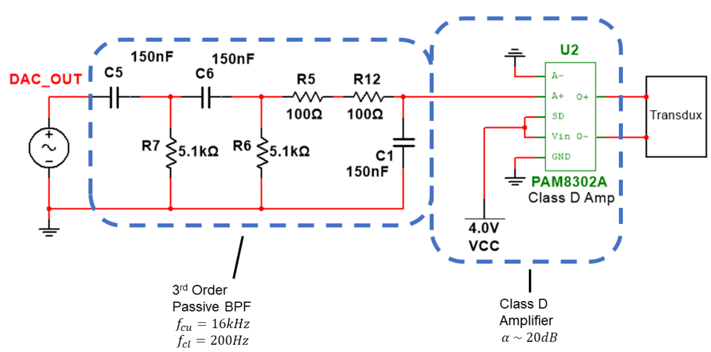
Unfortunately, it was also at this time in the development process that the COVID-19 outbreak occurred, forcing the team to reevaluate priorities. It was determined that the team should focus on simulation of the analog circuit, to ensure proper functioning. In addition, the team would prioritize generation of PCB Gerber files which could be printed and populated after it was safe to do so again. Completion of the circuit on the perfboard would be pursued but not prioritized.

Ultimately, the team was able to obtain all three of these goals. Plots of the optimized LPF and BPF are shown below, in **Figure 12a**. These passive filters are designed using just 33nF capacitors to minimize complexity. The PCB Gerber files of the completed system schematic are shown in **Figure 12d**. These files will allow the team to achieve seamless integration with the case, and reduce size and weight by using surface mount components. **Figures 12b and 12c** showcase the current prototype system implemented on a perfboard. Unfortunately, the team was only able to get 100Ω and $5.1k\Omega$ resistors, which limited the effectiveness of the filters to be designed.









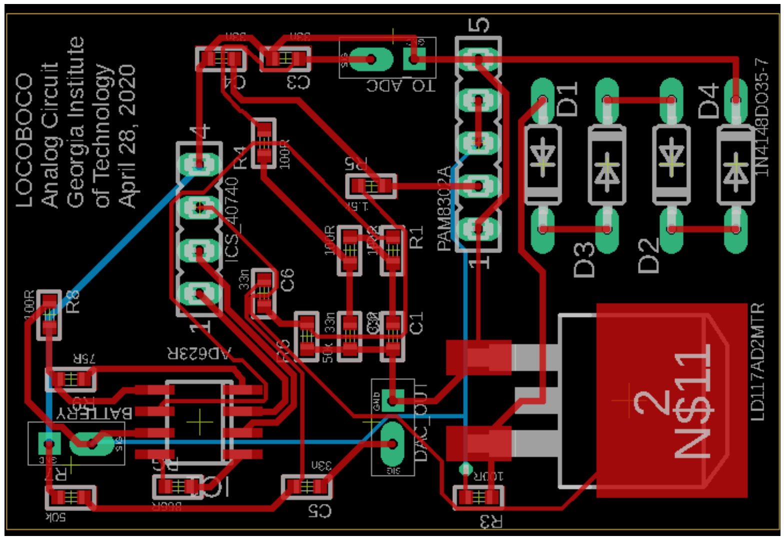


Figure 12. Post COVID-19 deliverables. (a) Top left: Magnitude response plots of simulated passive analog filters. (b) Top right: perfboard implementation of analog circuit under COVID-19 constraints. (c) Middle: Schematic of perfboard implementation. (d) Bottom: PCB Gerber file of analog system

4.1.3 Software Development

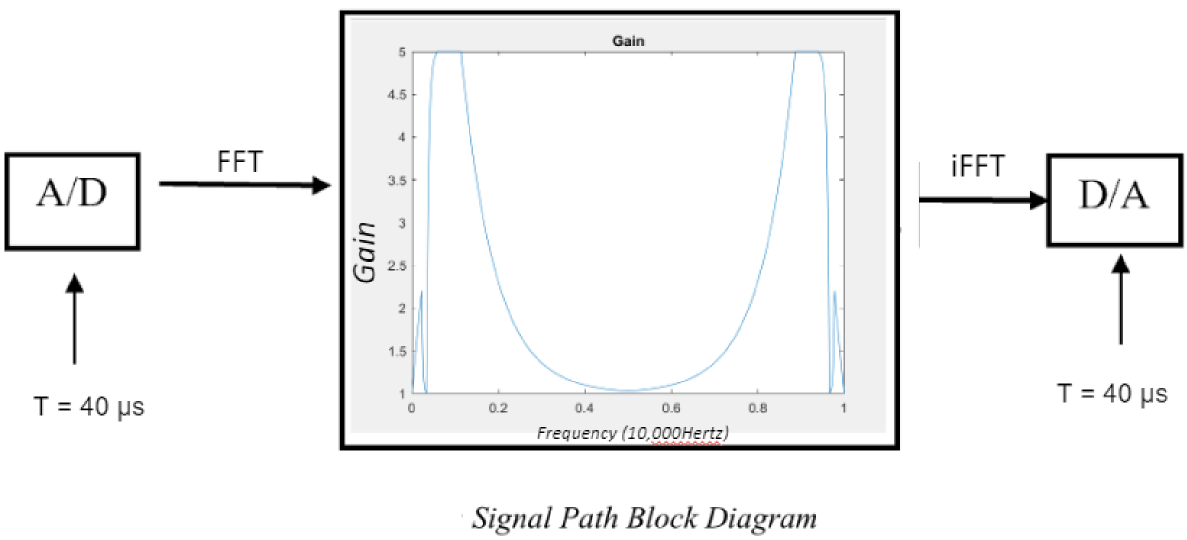


Figure 13. Signal path block diagram for software's overall filtering approach.

Figure 13. provides the software team's overall approach to filtering the input signal. The filtering will begin with sampling the signal at 25kHz, which is above the Nyquist frequency. The samples will then be converted from analog to digital values with an a/d converter so that the data can be manipulated using a microcontroller. An FFT will be applied to the digital values so we can apply our digital filter in the frequency domain. The filter's objective is to shape the frequency response so that frequencies that have a lower gain are amplified. The selection and explanation of the filter displayed in **Figure 13.** will be discussed in further detail below.. After applying the filter, an iFFT will be applied to the data which will be converted back into analog values with a d/a converter. The a/d and d/a conversions are handled by the microcontroller.

The first step for software development was to identify a filtering algorithm to assist with speech amplification and comprehensibility. This filter would largely be dependent upon the frequency response of a person's hearing, so the team ran a MATLAB simulation designed to graph the frequency response of one's hearing based upon a tonal hearing test [12]. The resulting frequency response can be seen in **Figure 14**. The shape of the individual's frequency response for hearing inspired the frequency transfer function in **Figure 15**, which serves to model a more general frequency response for hearing. With the general frequency response for hearing defined, a corrective filter could be designed to amplify the frequencies that have a lower gain, increasing compression of lower and higher frequency sound. The ski-slope filter the team designed can be seen in **Figure 16.**

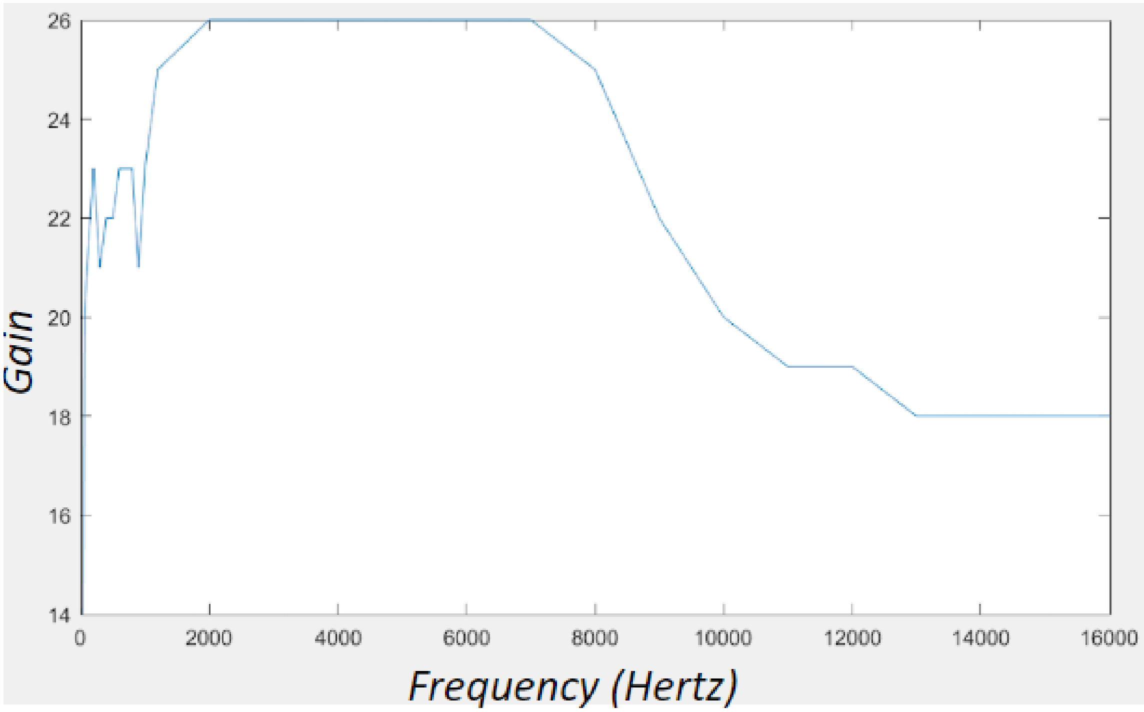


Figure 14. Graph depicting the frequency response for an individual's hearing based upon a tonal hearing test.

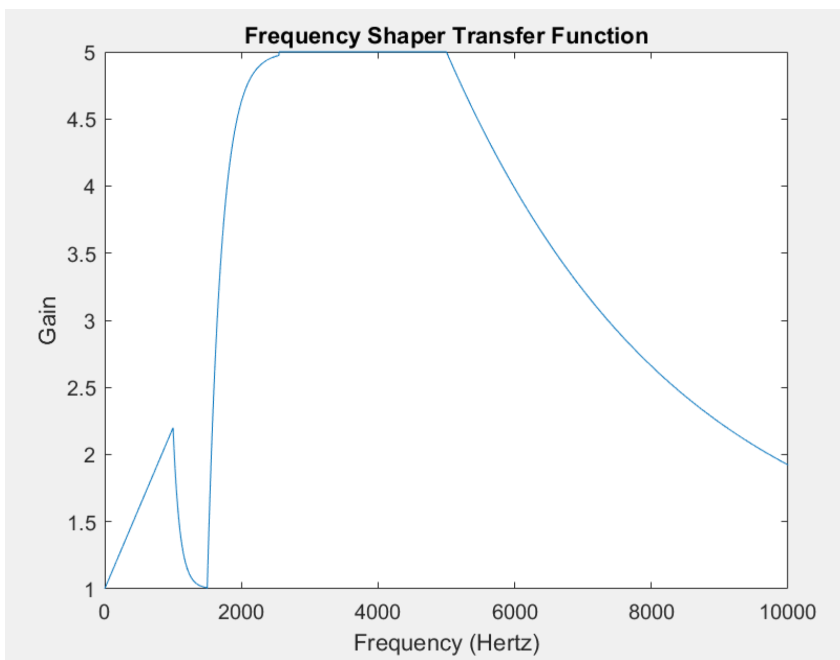


Figure 15. Graph depicting the general frequency response for hearing.

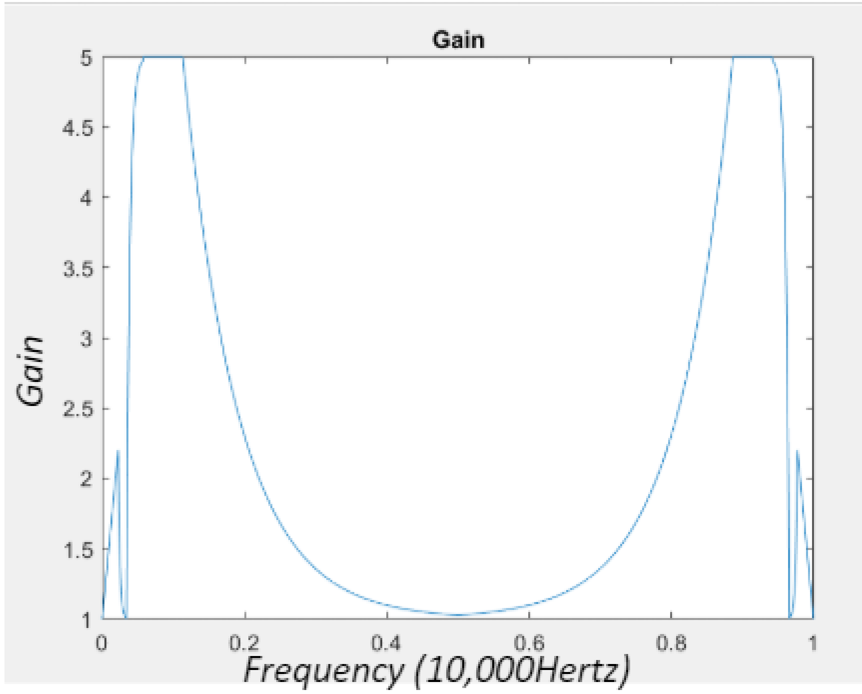


Figure 16. Graph depicting the ski-slope filter designed to amplify low and high frequencies.

Taking advantage of the symmetric properties of FFTs, the ski-slope filter was constructed by isolating the first half of frequency values and separating those values into five different ranges. The transition between these ranges can be seen in **Figure 15** as the slopes of the gain alternate. The first range is defined by a linearly increasing slope. The second range is defined by a decreasing exponential function. The third range is defined by an increasing exponential function. The fourth range is defined by a constant set by the maximum gain parameter. The fifth range is defined by a decreasing exponential function. These five ranges are repeated in a symmetrical manner on the latter half of the frequency values.

To test and apply the filter design, the team began development in MATLAB for its ease of use for DSP applications. A program was first developed to apply the ski-slope filter on audio files that simulate conversations for mild and moderate hearing loss. The program reads in an audio file and stores the data in an array. An FFT is applied to the data so that the frequencies can be manipulated based upon the ideal gain calculated by the ski-slope filter. The gain is multiplied by the audio signal in the frequency domain, and then the signal is converted back into the time domain with an iFFT. The adjusted audio is then played by the program so the modifications can be heard. Before filtering, the mild and moderate audio files have speech that sounds muffled. After the filter is applied, the speech is clarified and more comprehensible. The value for gain can be adjusted as a means to increase or decrease amplification.

Another MATLAB simulation was used to test the ski-slope filter in a real-time application. In this simulation, the benefits relative to hearing loss can not be heard, but it was useful for debugging and identifying limitations with the filter. In real-time, noise is amplified with the applied filter, thus it is important that the input signal is pre-filtered to limit noise.

Due to COVID-19, there was limited access to labs and hardware, so the team focused on developing modular software. As the final system will be run in C, the team focused on implementing the filter design in C. Key functions are written and tested separately from the hardware to demonstrate their functionality in the target language. Those functions include an FFT, iFFT, and the ski-slope filter algorithm. **Figure 17** shows an example output from the C program testing the combination of these functions and shows that the functions work correctly. For further clarity in analysis, results of this test output have been graphed in **Figure 18** and **Figure 19**. **Figure 18** displays that the data returned reflects the original data correctly after an FFT and iFFT are applied. **Figure 19** shows that the data was amplified after the ski-slope filter was applied and demonstrates that all of the functions work together.

```
Orig (dim=8):  0.13, 0.00   0.09, 0.09   0.00, 0.13  -0.09, 0.09  -0.13, 0.00  -0.09,-0.09  -0.00,-0.13   0.09,-0.09
 FFT (dim=8): -0.00, 0.00   1.00,-0.00   0.00, 0.00   0.00, 0.00  -0.00, 0.00   0.00,-0.00   0.00, 0.00   0.00, 0.00
iFFT (dim=8):  0.13, 0.00   0.09, 0.09   0.00, 0.13  -0.09, 0.09  -0.13, 0.00  -0.09,-0.09  -0.00,-0.13   0.09,-0.09


Orig (dim=8):  0.13, 0.00   0.09, 0.09   0.00, 0.13  -0.09, 0.09  -0.13, 0.00  -0.09,-0.09  -0.00,-0.13   0.09,-0.09
 FFT (dim=8): -0.00, 0.00   1.00,-0.00   0.00, 0.00   0.00, 0.00  -0.00, 0.00   0.00,-0.00   0.00, 0.00   0.00, 0.00
Gain Added (dim=8): -0.00, 0.00   5.00,-0.00   0.00, 0.00   0.00, 0.00  -0.00, 0.00   0.00,-0.00   0.00, 0.00   0.00, 0.00
iFFT (dim=8):  0.63,-0.00   0.44, 0.44   0.00, 0.63  -0.44, 0.44  -0.63, 0.00  -0.44,-0.44  -0.00,-0.63   0.44,-0.44
```

Figure 17. Test output showing functionality for the modular FFT, iFFT, and ski-slope filtering algorithm.

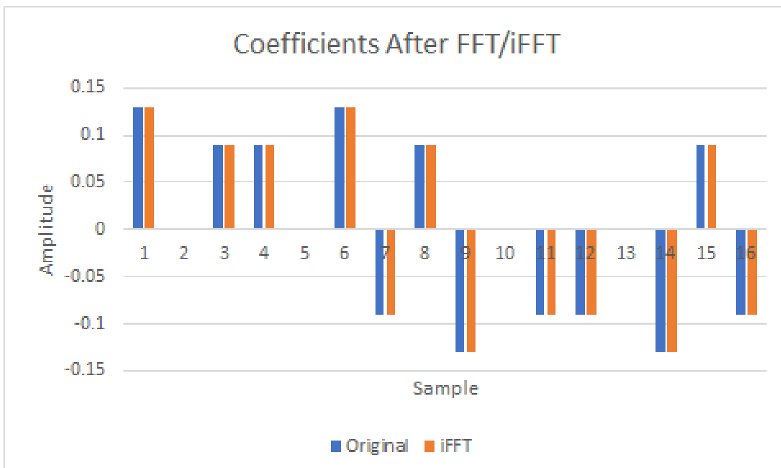


Figure 18. Graphed output showing that the data returned reflects the original data after the application of an FFT and iFFT are applied.

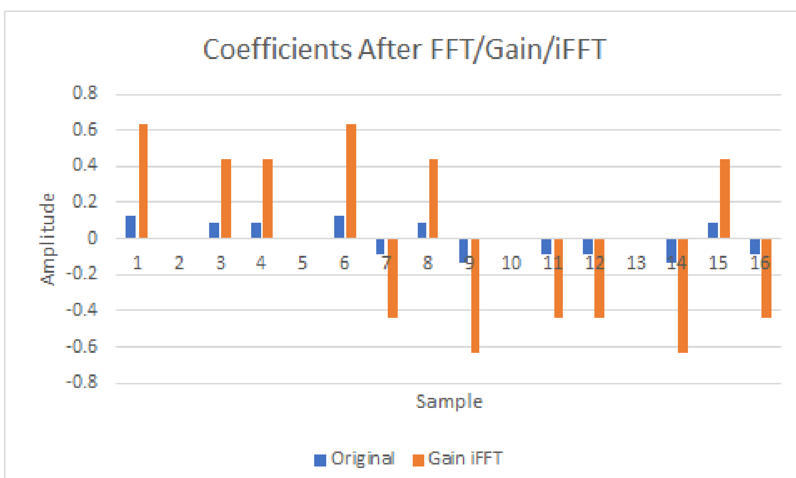


Figure 19. Graphed output showing that the data was amplified after the ski-slope filter was applied.

The team also had the task of selecting a microcontroller to carry out the computations. We wanted a device that was specifically designed for signal processing, while still being relatively small and cheap. In the end, the TMS320F28377S C2000TM 32-bit microcontroller, displayed in **Figure 20**, was selected because of its ease-of-use and ample documentation [5]. This device is floating-point and coded in C, and was specifically designed for real-time signal processing applications. The F28377S microprocessor is soldered onto a Launchpad, which allows for easy debugging and interfacing during these earlier iterations.

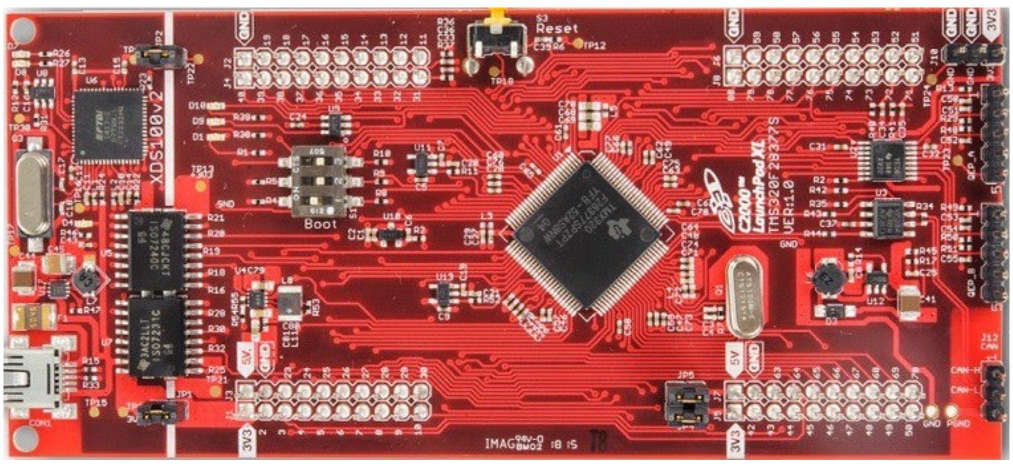


Figure 20. Overhead view of F28377s Microcontroller.

The C55 and C67 Digital Signal Processors from Texas instrument were also heavily considered; however the software team ran into major problems with these boards [13, 14]. The C55 uses Fixed-Point representation, and the software team wouldn't have had the time to code an algorithm by the deadline. The Floating-Point C67 seemed like a great option; however, there were no example codes or documentations on how to even initialize the board.

The team also started programming on the microcontroller. The clocking frequencies, interrupts, a watchdog timer, and other initializations have been set up. The ADC samples the analog microphone data on pin J7-4 and the DAC outputs an analog signal on pin J3-9 for the transducer, both at a sampling frequency of 25 kHz. **Figure 21** below shows the results of a successful initialization test.

Figure 21. Microcontroller Initialization Test Block Diagram.

The board was able to read a sinusoidal signal from a microphone on pin J7-4, displayed in **Figure 22**, apply a notch filter in the interrupt, and output the desired signal on pin J3-9, displayed in **Figure 23**.

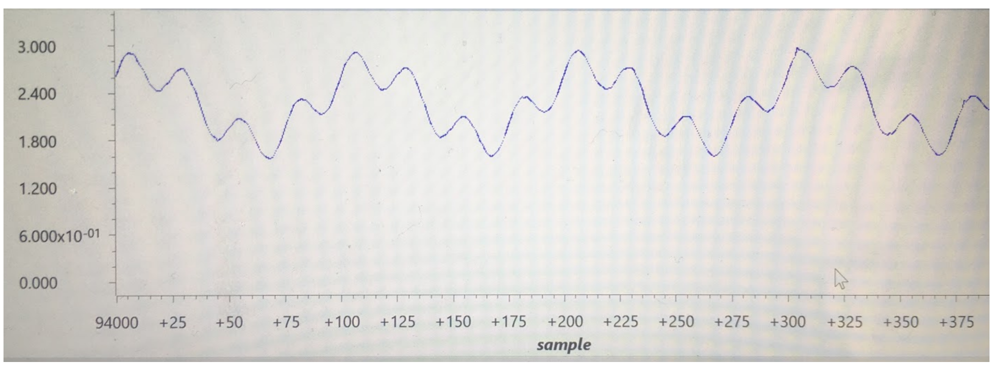


Figure 22. Microphone Input Signal of Sinusoid and Noise.

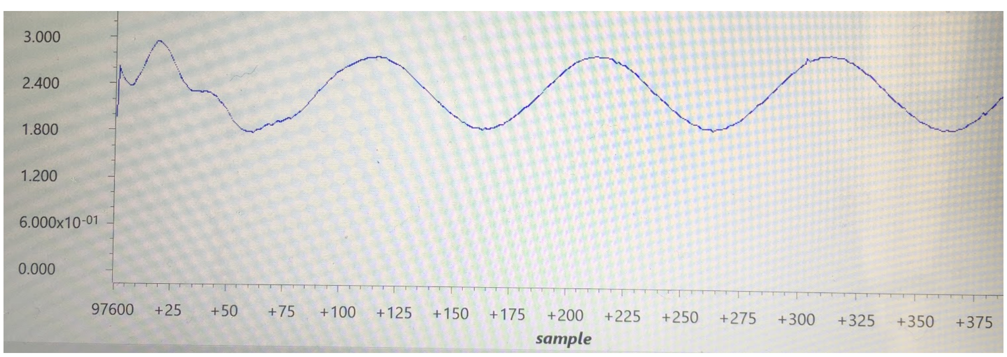


Figure 23. Output Signal of Sinusoid with Noise Removed.

The software team was also working on implementing Direct Memory Access to create a buffer system for interacting with the already described filter. There will be three 128-sample buffers that rotate between ADC, computation, and DAC. The buffers will be rotated in an interrupt that is triggered each time the ADC has sampled 128 times. We selected a working C2000Ware demo that sets up an ADC channel to transfer the results by the DMA into a buffer in RAM. This demo contains the relevant initialization and code structures necessary to implement DMA in our project. Unfortunately, due to COVID-19, this DMA implementation, as well as integrating the filter functions onto the board, have not yet been tested.

4.1.4 Discussion and Future Work

The development of the LCBC was an incredibly intensive process and was also impacted by the COVID-19 outbreak, forcing the team to postpone certain development objectives. The team intends to continue work on the LCBC, focusing on the following items first.

The next major development milestone to be accomplished will be integration between all three components: analog system hardware, software, and the case. Several system updates need to take place for this to be achieved. First the PCB Gerber files need to be printed and populated, so that a dual channel version of the planned analog system can be implemented. The case must also be updated to facilitate effective coupling of the bone conduction transducers to the users head, allowing transmission of mechanical energy to the cochlea. The final step will be to deploy the software algorithms developed in C on the microcontroller so that DSP frequency shaping can take place.

After achieving complete system integration, development efforts will stop so that a comprehensive evaluation of the system can be accomplished. This will ensure that the current system design is adequate for shaping the transfer function of incoming acoustic signals and ultimately improving speech comprehensibility by the user. Professional grade equipment will be required for effective system evaluation, so the team will rely on the facilities of Dr. Bhamla’s contacts for LCBC testing.

Future improvements have also been planned for the LCBC. Future algorithm work will include incorporating acoustic beamforming into the code executing on the microcontroller utilizing the array of microphones installed on the LCBC hearing aid. Acoustic beamforming will increase the directionality of microphones and allow for common mode cancellation of background noise. This is highly desirable for a hearing aid, as those affected with mild to moderate hearing loss often lose directionality within their own hearing capabilities and often have a reduced effective SNR, as HL decreases relative to the noise floor. Implementing a beamforming algorithm can alleviate this.

An additional future development objective will include developing and incorporating a calibration procedure for fine tuning the overall frequency response of the system to address the variability in the audiograms of the customers. As the LCBC is targeted towards those in low- to middle- income countries and regions, the calibration procedure will not require any specialized equipment. In addition, the calibration should be able to be accomplished by someone with limited experience, and for those whose audiogram might not be known. To accommodate these requirements, the calibration procedure will need to play back pure tones through the LCBC hearing aid and allow users to select appropriate gain values in response to their fluctuating HL’s.

A final addition will be incorporating left vs. right balance adjustment for the LCBC hearing aid. It is often the case that those with mild to moderate hearing loss have variability in HL between their right and left ears. The LCBC should be able to address this potential disparity so that users do not continue to have HL imbalance.

4.2 Codes and Standards

As the LCBC Hearing Aid will not communicate with any other devices and will only require a DSP board interfacing with analog components, there are not any codes and standards necessary to specify for the technical design. However, the LCBC Hearing Aid will be subject to several FDA regulations, and will be tested in accordance with American National Standards Institute (ANSI), the International Electrotechnical Commission (IEC), and World Health Organization (WHO) standards and guidelines for hearing aid performance.

According to the FDA (as of Nov 2019), a Hearing Aid (proper) can only be purchased after evaluation from a licensed physician [15]. The intended users of the LCBC Hearing Aid are low-to-middle-income individuals who potentially may not have access to a physician who could diagnose hearing loss and prescribe hearing aids. As such, the LCBC Hearing Aid will be classified as an over the counter (OTC) Personal Sound Amplification Product (PSAP), which relieves the LCBC Hearing Aid of most federal regulation. However, as a PSAP, the LCBC Hearing Aid must conform to the Radiation Control for Health and Safety Act of 1968, specifically 21 CFR 1000.15, 21 CFR Part 1003, and 21 CFR Part 1004 [15]. These regulations protect consumers against defective electrical products, and will require LCBC to both fail gracefully in the event of internal defect and only emit acoustic frequency vibration.

Performance of the LCBC Hearing Aid should also be evaluated in accordance with the ANSI S3.6, ANSI S3.13, ISO 389-3 1994, IEC 60318-5 2006, IEC 60318-6 2007, and ISO 1683 2008. These standards contain requirements on the calibration, testing, and characterization of audiological emitting equipment [16]. ANSI S3.13, ISO 389-3 1994, IEC 60318-6 2007, and ISO 1683 2008, specifically provide guidance on testing of bone vibration audiological devices as the most prominent direct acoustic measurement, SPL, cannot be directly obtained from a bone vibration device. Performance specifications of the LCBC Hearing Aid have been selected from ANSI/CTA-2051 and WHO recommended values.

4.3 Constraints, Alternatives, and Tradeoffs

4.3.1 Constraints

The two main constraints on the design are cost and size. The average cost of a hearing aid is currently $2,300 per unit [17]. The ultimate goal is to create a hearing aid under $30, a 99.35% percent decrease in cost. In order to achieve this goal, the design eliminates customization of parts and algorithms for an individual’s specific needs. Instead, a general speech enhancement algorithm will be used along with a minimal hardware design that has a specialized processor allowing for signal manipulations at the software level. Minimizing hardware components simplifies the design and decreases possible part failures.

The other main constraint on the project is size. The size of the final design needs to fit within a headband. Typically, decreasing the size of a design decreases the performance as less transistors can be implemented in the design. In order to meet this constraint, both the hardware design and software complexity need to be minimized. Battery size must also be selected to fit the size constraints while also being rechargeable with a battery life of 6 to 12 hours.

A latent constraint in the iterative design process introduced by COVID-19 was a reduction in the development time allowed. As of Friday March 13th, 2020 hardware design iteration and improvement was frozen as access to necessary prototyping tools and resources was cut-off. Software development has continued, but at a greatly reduced pace, as group access to DSP development boards and testing equipment is no longer possible. In order to combat this, the software team will implement a multi-tier design to separate the algorithm from the hardware interface. This will allow the algorithm development to progress remotely.

4.3.2 Alternatives and Tradeoffs

Throughout the design process the team was confronted with many choices which required a tradeoff evaluation. The team formalized as many specifications as reasonably possible and created implicit constraints to guide to decision decisions. The primary tradeoffs that existed for all design decisions were system cost, size, and ease of development. These formed the fundamental tradeoffs for the team.

For hardware design, the focus on low system cost, small size, and ease of development, drove the implicit constraint to reduce system complexity. This manifested itself in limiting amplifier count and thus utilization of passive filters. Additionally, the system was designed using minimal component variation. The case was designed to be compact and 3D printable in readily available filaments thus reducing production costs.

The software team put early emphasis on understanding the fundamental tradeoffs of different types of development boards. In order to compute the DSP algorithms, a general purpose microprocessor, microcontroller, FPGA, or DSP specific processor could have been used. While a general-purpose microprocessor offers flexibility in its capabilities and could be used for DSP applications, it is too large and expensive to be used for the final design. Microcontrollers are more specific than general-purpose microprocessors, allowing the device to be simpler, cost effective, and lower in power consumption. If a microcontroller were chosen it would need to be able to do multiple multiply and accumulate (MAC) in a single cycle. Some offer this ability but are typically slower and may be less parallelizable compared to other devices like an FPGA. As timing is vital for this design, a device specialized in DSP applications was preferable. FPGAs are comprised of logic gates that a user specifically programs allowing great customization. The logic cells can perform simple calculations efficiently and in parallel increasing output performance compared to microcontrollers and DSP specific processors. One drawback is the programmability of an FPGA. Fewer DSP algorithms are defined for FPGAs causing the code development to be much more complex and time consuming to develop. If the needed performance can be achieved with a small enough DSP processor, a DSP processor was preferable because of its ease in use and pre-defined specialization for DSP specific calculations. DSP microprocessors or DSP microcontrollers are optimized for signal processing and can provide fast instruction sequences. It is also flexible in its programmability allowing various languages including MIPS assembly and C to be used for development. Also, some include RTOS for real time programming while maintaining low power consumption. The selection of a DSP microprocessor or DSP microcontroller is prefered.

The team originally planned to develop software for either a Texas Instrument C55 or C67 digital signal processor. Though these boards are highly specialized for DSP applications, they lacked documentation and required extra debugging components to begin running any software on the boards. The process of starting development on these boards was deemed too time consuming for selection. Although the TMS320F28377S C2000™ 32-bit Microcontroller was not marketed specifically as a DSP microcontroller, it is specialized for real-time and DSP applications and was readily available for development, suggesting that a DSP microprocessor or DSP microcontroller was preferable [5].

For implementation on the selected DSP development board, there are two main language families that were originally considered for this project: C and assembly. When determining which language is better, efficiency, portability, time, specific hardware, and the engineers' proficiencies were considered. C offers higher level syntax that is more portable and easier to develop compared to assembly languages, but the hardware device must have enough memory so that the program's efficiency is not greatly impacted. Additionally, there are more examples of DSP and real-time algorithms in C that decrease time spent on code development. Assembly can be more beneficial when execution time and memory are of high concern as assembly can directly address specific registers and take advantage of specific hardware designs. Furthermore, all processors support the use of an assembly language, but not all processors can support C. Because C is easier to develop in and all team members have experience in C, C is the development language of choice. This programming language preference influenced the decision in selecting the TMS320F28377S C2000™ 32-bit Microcontroller.

5. Schedule, Tasks, and Milestones

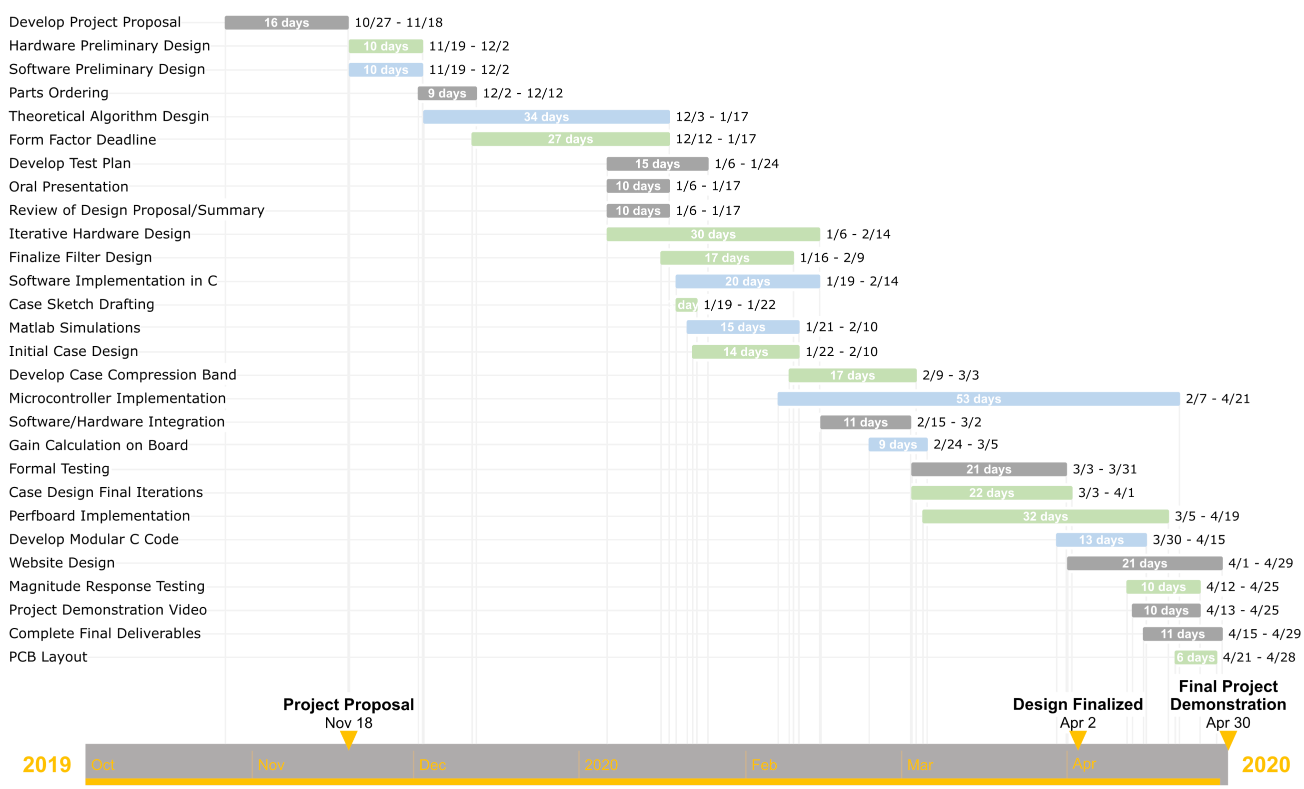


Figure 24: Gantt Chart

The Gantt chart in **Figure 24** shows the project's projected timeline, broken into hardware and software tasks so that maximum parallelism can be achieved throughout the semester. The overall duration of the project was approximately 185 days.

The Hardware sub-team, composed of Aaron, Bradley, and Cynthia, was responsible for completing the green tasks in **Figure 24**. The GANTT Chart Tasks include hardware preliminary design, form factor deadline, iterative hardware design, finalize filter design, case sketch drafting, initial case design, develop case compression band, case design final iterations, perfboard implementation, magnitude response testing, and PCB layout.

The sub-team completed the majority of analog system design in bi-weekly team meetings to make sure that the team was in agreement on design decisions. Together the sub-team outlined the preliminary design for the analog system and the desired form factor for the headband case and belt assembly. The sub-team iteratively designed the hardware schematic, including the filter design, and PCB layout. Although all design decisions were made collaboratively, Bradley was responsible for the modeling of the headband case assembly and the magnitude response testing, Cynthia for the belt case model, and Aaron for the PCB Gerber file generation.

The Software sub-team, composed of Caitlin, Megha, Lillian, Brooke, and John, was responsible for completing the blue tasks in **Figure 24**. The GANTT Chart Tasks include software preliminary design, theoretical algorithm design, software implementation in C, matlab simulations, microcontroller implementation, gain calculation on board, and develop modular C code.

The final preliminary software design is depicted in **Figure 13** as an overview of the software approach. This design was sketched out with input from Lillian, Caitlin, and Megha and was deemed an important but not difficult task. With the preliminary design in mind, Caitlin researched past hearing aid designs that the software team could draw inspiration from when designing the amplification filter. These findings concluded the theoretical algorithm design and led to the MATLAB ski-slope filter simulations for recorded audio file processing and the real-time simulation. This preliminary research not only had low difficulty but reduced future development difficulty as the team did not need to design a filtering algorithm from scratch. The MATLAB simulation on recorded audio files was completed through collaborative work between Caitlin and John. With a working filtering function in MATLAB, Caitlin added this function to the real-time MATLAB simulation and did some filter debugging. These tasks posed moderate difficulty as getting the simulations to work involved testing multiple noise reduction strategies and associated debugging for each attempt.

The software team researched different boards to use for development. After initially selecting the Texas Instrument C67 digital signal processor, Lillian and John installed Code Composer Studio and attempted to run several demos on the board. After switching to the F28377S microcontroller, Lillian helped install the correct version of Code Composer Studio and C2000Ware, which contains software examples and documentation for easy development, on everyone's computer, as well as explaining an introduction on how to use CCS and the microcontroller. Lillian then led efforts to initialize the microcontroller (clocking frequencies, watchdog timer, interrupts, ADC, DAC, DMA, etc.) and set up early stages of integration tests with hardware. Megha worked on using SIMULINK to create a filter for the microcontroller without the assistance of Code Composer Studio [18]. Caitlin worked with rewriting the ski-slope filter's gain calculation in C on the microcontroller.Interfacing with the board's hardware proved very challenging due to insufficient documentation, limited memory, and the Code Composer Studio API restrictions.

Working FFT and iFFT functions in C were developed as a result of more collaboration between John, Caitlin, and guidance from Dr. Anderson. This task had moderate difficulty as setting the specific properties for C to work in Visual Studios proved tedious. Due to uncertainty about the functions operating properly, Caitlin researched another FFT and iFFT option that employed FFTW, a free DSP library for C. An example program using the library's functions was developed as a back-up plan if the individual FFT and iFFT implementations were not successful. Once a working FFT and iFFT were implemented in C, Caitlin added the necessary functions to implement and test the ski-slope filter with the FFT and iFFT functions. Due to previous implementations of the ski-slope filter, this task was only mildly challenging due to some differences in the API environment.

As most of the work was done in software, the technical risk was found most in tasks requiring multiple implementation attempts, while development time was the resource lost on unsuccessful implementations.

The GANTT Chart Tasks include project proposal, parts ordering, develop test plan, oral presentation, review of design proposal/summary, software/hw integration, formal testing, project demo video, and completed final deliverables. All members of the team worked collaboratively on the project proposal, oral presentation, and final deliverables design. Weekly reports were sent out by Brooke, Bradley, and

Caitlin. All formal parts ordering was completed by the documentation lead, Bradley. Brooke was responsible for registering for the Senior Design Expo and editing the final videos. The team's webmaster, Megha, was responsible for website design and the inclusion of all necessary deliverables. These tasks had low difficulty as they were largely collaborative and straightforward.

6. Final Project Demonstration

Table 1 lists the originally proposed system specifications for the hearing aid and the measured values. Each specification has been achieved. The supply voltage of 3.9V - 5.0V was created using a commercial power supply, which is tested by the company. The 6-10 hour battery life was tested by playing white noise on a speaker and having the device send the signal from the microphone through a transducer, and then checked every hour to see when the battery died. The battery died after 10 hours of testing; however, the battery life is SPL dependent, so the lower limit of 6 hours was added for margin of error. The frequency response of 200 - 10,000 Hz was created by implementing a band pass filter before the signal reaches the transducer, using an upper cutoff frequency of 10,000 Hz and a lower cutoff frequency of 200 Hz. This was verified by a simulation in Multisim [19]. The 25 kHz sampling frequency was created by setting clocking frequencies correctly in the microcontroller implementation. This was verified with an oscilloscope reading the DAC output. The maximum algorithm delay was calculated by multiplying the buffer size by the number of buffers between ADC and DAC and then dividing everything by the sampling frequency. Our buffer size was 128 samples, there were two buffer lengths in between input and output, and the sampling frequency was 25 kHz. Finally, the computation speed had a sample rate of 25 kHz and a clock frequency of 200 Mhz. These values were manually set during microcontroller initialization.

This project can be separated into three functional modules: analog hardware, case, and software. For hardware, prototype testing of the analogy hardware components was accomplished by measuring frequency response characteristics, as this characteristic was the primary developmental concern. The exact method of interrogation varied for each component. For example, microphones were stimulated with speakers while the bone conduction transducers were measured with accelerometers. All stimuli utilized pure tones sweeping the system bandwidth and all responses were recorded as peak to peak magnitudes of the components response. Due to limitations previously discussed, some of the data is not completely reliable due to confounding variables, however general trends can still be used as reference information.

Throughout the development of the case, qualitative testing and user feedback was gathered allowing the team to refine dimensions and other imperfections. These sessions allowed the team to tune the compression magnitude and vector as previously discussed. Unfortunately, due to restrictions on the testing facilities available to the team post COVID-19, rigorous testing of the case prototype was not able to be conducted.

For the software module, the real-time simulation in MATLAB was used to test the filter's effect on noise by adjusting the gain and the input signal. When the gain was increased, background noise was increased, showing a positive correlation between the two. For integrating the software and hardware modules, the microcontroller was initialized and used to connect various parts of hardware to the algorithm in C. Various aspects of hardware and software could still be isolated, and this form of testing allows for hardware and software compatibility verification. The notch filter test displayed in **Figures 21-23** shows an example of this prototyping testing in more depth.

Due to the cancellation of the Senior Design Expo, the final project demonstration includes two videos documenting the team's progress in development, both available on the completed project website at: https://ece4012y202002.ece.gatech.edu/sd20p04/. One video is a recording of a final presentation documenting the team's research, documentation, and thoughts on future work. The presentation highlights the problems that inspired this project and the team's main objectives that were considered throughout development. Market alternatives, development cost, and bulk market cost are presented. The presentation overviews the technical specifications as well as the software and hardware design approaches. A current status is given and leads to the discussion of future work. The second video includes a simulated demonstration of the system's functionality after current contingency plans have been completed. The final project demonstration highlights:

- An overview of the entire analog system
- A demonstration of the MATLAB simulation showcasing the implementation of the filter on audio files
- A demonstration of hardware and software initial integration
- An overview of the case design

Additional files including hardware and software simulations, program solutions, and Gerber files can be found at the github link, https://github.com/cchoate6/EnginHears.git, and in the Appendix.

7. Marketing and Cost Analysis

7.1 Marketing Analysis

Despite a proliferation of hearing aid and PSAP products on the market today, there is not a low-cost (<$50) singular bone conduction hearing aid available. **Table 3** provides a comparison of the mean cost of several prominent hearing aids types:

Table 3. Hearing aids by cost

Hearing Aid Type	Conduction Modality	Cost
Bone Anchored Hearing Aid (BAHA)	Bone Conduction	$9,500 [20]
Signia Behind The Ear (BTE)	Air Conduction	$3,598 [21]
Signia Completely In the Canal (CIC)	Air Conduction	$3,598 [22]
Heartech CIC	Air Conduction	$2,000* [23]
Heartech BTE	Air Conduction	$2,000* [23]
Bonein BN-802	Bone Conduction	$75 [24]
Low Cost Bone Conduction (LCBC)	Bone Conduction	$30

* Non-promotional price

There are some low cost bone conduction hearing aids currently available on the market today. The most well-reviewed of these is the Bonein BN-80X Series [24]. These products are functionally very similar to the LCBC Hearing Aid, however their form factors are different. The BN-802 requires a wired connection to a separate microphone/processing unit, a design component that is not a part of the LCBC hearing aid. Additionally, these comparable products are still significantly more expensive (2.5x - 6x) than the proposed cost of the LCBC hearing aid. The LCBC hearing aid's form factor is also significantly larger than many of the current market air conduction hearing aids (BTE, ITE, CIC), allowing elderly users to more easily manipulate controls, handle, and apply / remove the LCBC hearing aid. The form factor of the LCBC hearing aid also presents an aesthetically refreshing alternative to other current market hearing aids, which are easily recognizable as hearing aids. The LCBC hearing aid does not stand out as a hearing aid, as its form factor is sleek and very similar to other popular bone conduction headphones [25]. This subtly gives improved hearing capabilities to users who may not wish to advertise their hearing impairments with a conventional hearing aid.

7.2 Cost Analysis

The total development cost breakdown of the LCBC hearing aid is summarized in **Table 4**. Cumulative project development labor is tallied in **Table 5**.

Table 4. Total development costs

Category	Cost
Labor	
Engineer I - $75k (8 Engineers)	$42,900.00
Development Parts	
Microphones	$394.62
DSP Boards	$141.77
Bone Conduction Transducers	$53.70
Batteries	$101.60
Miscellaneous (wire, passive components, etc.)	$46.22
Bone Conduction Headphones (Aftershokz)	$79.95
Class D amplifiers	$53.50
3D print filament	$54.94
Development Subtotal:	$926.30
Indirect Costs	
Development Facilities	$51,759.00
Fringe Benefits	$8,010.00
Total	$103,595.30
Total Real Cost	$926.30

Table 5. Total project development labor

Labor Category	Individual Hours	Total Hours
Weekly Meetings	8.5	68
Software Development	88	440
Hardware Development	114	342
Integration (Software + Hardware)	3	24
Testing/Simulations	22	110
Presentation Development	20	160
Total	**255.5**	**1,144**

Table 6 provides a breakdown of the projected budget compared to the teams actual expenditures on prototyping components.

Table 6. Requested funding vs. actual spending

Requested		Actual	
Category	**Cost**	**Category**	**Cost**
Development Parts		**Development Parts**	
Microphones	$50.00	Microphones	$394.62
DSP boards	$75.00	DSP boards	$141.77
BC transducers	$50.00	BC transducers	$53.70
Batteries	$35.00	Batteries	$101.60
Misc components	$35.00	Misc components	$46.22
Aftershokz headphones	$75.00	Aftershokz headphones	$79.95
Unanticipated Costs	$100.00	Class D amplifiers	$53.50
Demonstration Prototypes (5)		3D print filament	$54.94
Required Components	$325.00		
Total:	**$745.00**	**Total:**	**$926.30**

As the LCBC is primarily marketed towards low- to middle- income countries, a bulk cost analysis was also completed to ensure that the final cost of the hearing aid would still be near the original $30 price point. A component breakdown of the analysis of the bulk cost is presented in **Table 7**. For the purposes of the analysis presented in **Table 7**, only the necessary to build the LCBC in its current prototype state were analyzed. Bulk is defined as 10,000 units, and bulk cost pricing was used when available from manufacturers. However, when bulk pricing was not available, it was assumed that a discount rate would still be applied. The mean discount rate of 0.288264 was calculated from the available discount rates and applied to components that did not advertise a bulk pricing discount. The 3D print quantity is specified as 0.144 as this is the fraction of filament used to print a case assembly per 1kg of material. Ultimately the per unit bulk component cost was determined to be $31.38.

Table 7. Component bulk cost analysis

P/N	Description	Qty	Bulk Cost	Total Cost
LAUNCHXL-F28377S	TI MCU Development Kit	1	$8.67 [26] *	$8.67
1674	Adafruit Bone Conduction Transducer	2	$2.58 [8] *	$5.16
328	3.7V LiPo Battery (2.5Ah)	1	$4.31 [6] *	$4.31
ICS40740	Invensense Analog MEMS Microphone	2	$1.65 [7]	$3.30
AD623ANZ	Instrumentation Amplifier	1	$2.92 [11]	$2.92
2130	2.5W Class D Amplifier	2	$1.14 [9] *	$2.28
1304	LiPo Battery Charger	1	$1.72 [27] *	$1.72
TPU-3.00mm-Black	3D Print Filament (3mm - 1kg)	0.144	$8.62 [4] *	$1.24
N/A	Misc Expendables (Cable, Solder, etc.)	1	$1.00	$1.00
B3F-1022	SPST Tactile Pushbutton	2	$0.15 [28]	$0.30
9B-10.000MAAJ-B	Crystal Oscillator	1	$0.18 [29]	$0.18
LD1117AV33	Linear Voltage Regulator	1	$0.16 [30]	$0.16
CC0402KRX5R7BB154	Capacitor - 150nF	6	$0.01 [31]	$0.07
1N4148	Diode	4	$0.01 [32]	$0.04
RK73H1JTTD1000F	Resistor - 100 Ohms	13	$0.00 [33]	$0.03
RK73B1ETTP512J	Resistor - 5.1k Ohms	2	$0.00 [34]	$0.00
			Total	**$31.38**

* Bulk cost estimated

As the target market of the LCBC is low- to middle- income users, the group does not intend to have a large profit margin, and would reinvest all profits into the future development of the device. An analysis of the final sale price of the LCBC in its current state is presented in **Table 8**. This analysis assumes that the LCBC is to be developed at scale in a manufacturing facility, so additional costs such as labor, benefits, and shipping must be examined.

Table 8: Production and Sales Cost Analysis

Expenses & Income	Amount
Expense	
Assembly Costs	
Parts (Bulk Cost)	$31.38
Labor	$4.17
Fringe Benefits on Labor	$1.25
Overhead	$2.00
Testing	
Labor	$1.81
Facilities Lease	$3.00
Assembly Subtotal	$27.23
Marketing	$1.00
Shipping	$1.00
Total Expenses	$45.61
Income	
Sale	$50.00
Total Income	$50.00
Net Income	$4.39
Profit Margin	8.8%

Despite the above cost analysis being over budget, this LCBC still beats out other relevant competitors such as the Bonein BN-802 which is priced at $75 [24]. Additionally, a $50 price point still represents a 97.8% decrease in price from the average cost of hearing aids [1].

8. Conclusion

The development of the LCBC was an incredibly educational process immensely broadening the team’s engineering design experience. The goals of the project were ambitious, given the available development time and the proposed scope. Development required the team to quickly learn about processes and software that they had never used, such as materials analysis, SolidWorks, and Code Composer Studio. Additionally the team's ingenuity and resolve were put to the test as the COVID-19 outbreak greatly reduced the teams ability to collaboratively develop, test, and work on the project in a lab. Despite these challenges the team was able to accomplish a significant amount throughout the semester, delivering a prototype implementation of the necessary hardware, simulations of functionality, and modular c functions ready for deployment.

The design process forced the team to overcome several other significant obstacles. As the team began the iterative design portion of the development process, it became apparent that the scope of the project should be somewhat reduced to increase feasibility of project completion. This involved moving beamforming, certain case development goals, and construction of frequency response configuration software objectives to reach goals so that the team could focus on the essential development requirements. In addition, the team was forced to continue system development post COVID-19 outbreak. In response, the team quickly reassessed and redefined deliverable requirements. The team was also still able to complete a prototype analog hardware system facilitating simulations of the magnitude response of the system. In addition, modular C functions ready for deployment to microcontrollers, software simulations, and the final case assembly design iteration were all accomplished post COVID-19 outbreak. Unfortunately, formal system testing as originally proposed was not able to be accomplished, as the team did not have enough development time or adequate lab space. Throughout this semester, the team learned the value of planning, and the importance of flexibility and contingency plans when obstacles become reality.

Currently, the team has developed a software approach that tests and simulates a ski-slope filter in MATLAB and C. To reflect software modularity, individual functions for an FFT, iFFT, and ski-slope filter have been developed in C. A microcontroller has been initialized with features such as a watchdog timer, I/O, and ADC and DAC conversions. The team has also made significant headway on the originally proposed analog hardware objectives. A functioning analog system prototype has been designed, implemented, and tested on a perfboard, proving the feasibility of the proposed design. In addition, improvements to this design, largely filter related, have also been developed and simulated in Multisim, allowing the team to improve the implementation when development can be resumed. Finally, Gerber files have been generated with the updated filter components, further miniaturizing the system and reducing cost by allowing use of surface mount components. Currently the case assembly is on its 9th developmental iteration, and has been printed in ABS for the project demonstration. A compression solution has been created to generate a coupling force at the temples of the user's head, and compartments for all electrical components have been designed and implemented. All SolidWorks files have been delivered to facilitate rapid revisions as required.

For future work that can be done on this project, it is advised to review the 'Future Work' section of the final report. To have a clear understanding of where the project left off, it is also recommended to watch the final presentation and demo. To begin working where the team left off, the team has compiled all of the useful files into a team github repository at https://github.com/cchoate6/EnginHears.git. There are simple README files in each folder to describe what can be found. If there are any additional questions, the team can be reached at enginhears@gmail.com.

9. Leadership Roles

Table 9 shows each leadership role with the respective team member in charge.

Table 9. Leadership Roles

Position	Name
Algorithms Lead	Lillian Anderson
Audio Lead	Bradley Blaho
Documentation Lead	Bradley Blaho
Embedded Systems Lead	Cynthia Baseman
Expo Coordinator	Brooke Brennan
Hardware Lead	Aaron Green
Integration & Test Lead	Megha Tippur
Software Lead	Caitlin Choate
Software Co-Lead	John Landers
Systems Lead	Brooke Brennan
Webmaster	Megha Tippur

10. References

[1] C. Cassel and E. Penhoet, "PCAST Recommends Changes to Promote Innovation in Hearing Technologies," *National Archives and Records Administration*, Oct-2015. [Online]. Available: https://obamawhitehouse.archives.gov/blog/2015/10/26/pcast-recommends-changes-promote-innovation-hearing-technologies. [Accessed: Nov-2019].

[2] A. Mccormack and H. Fortnum, "Why do people fitted with hearing aids not wear them?," *International Journal of Audiology*, vol. 52, no. 5, pp. 360–368, Nov. 2013.

[3] "3D CAD Design Software: SOLIDWORKS," *Product Categories*. [Online]. Available: https://www.solidworks.com/. [Accessed: 29-Apr-2020].

[4] "Gizmo Dorks 3mm (2.85mm) Flexible Filament (TPU), 1 kg for 3D Printers, Red," *amazon.com*. [Online]. Available: https://www.amazon.com/Gizmo-Dorks-Flexible-Filament-Printers/dp/B00T7P1MZG. [Accessed: 29-Apr-2020].

[5] "TMS320F28377S ACTIVE This product has been released to the market and is available for purchase. For some products, newer alternatives may be available. C2000™ 32-bit MCU with 400 MIPS, 1xCPU, 1xCLA, FPU, TMU, 1024 KB Flash, EMIF, 16b ADC," *TMS320F28377S C2000™ 32-bit MCU with 400 MIPS, 1xCPU, 1xCLA, FPU, TMU, 1024 KB Flash, EMIF, 16b ADC | TI.com*. [Online]. Available: https://www.ti.com/product/TMS320F28377S?utm_source=google&utm_medium=cpc&utm_campaign=epd-c2x-null-prodfolderdynamic-cpc-pf-google-wwe&utm_content=prodfolddynamic&ds_k=DYNAMIC+SEARCH+ADS&DCM=yes&gclid=Cj0KCQjwy6T1BRDXARIsAIqCT

XoNXWtnfm0lm7uwLw3y9S_Wev-NymUXNGxviwzCEJorVNsQ5UziFs0aArAKEALw_wcB&gclsrc=aw.ds. [Accessed: 29-Apr-2020].

[6] "328," *DigiKey*. [Online]. Available: https://www.digikey.com/product-detail/en/adafruit-industries-llc/328/1528-1840-ND/5054542. [Accessed: 27-Apr-2020].

[7] "ICS-40740 Datasheet," *InvenSense*. [Online]. Available: https://invensense.tdk.com/download-pdf/ics-40740-datasheet/. [Accessed: 29-Apr-2020].

[8] "1674," *DigiKey*. [Online]. Available: https://www.digikey.com/product-detail/en/adafruit-industries-llc/1674/1528-1948-ND/6827067. [Accessed: 27-Apr-2020].

[9] "2130," *DigiKey*. [Online]. Available: https://www.digikey.com/products/en/development-boards-kits-programmers/evaluation-boards-audio-amplifiers/789?k=PAM8302. [Accessed: 27-Apr-2020].

[10] "EVAL-ADXL354/EVAL-ADXL355 User Guide." [Online]. Available: https://www.analog.com/media/en/technical-documentation/user-guides/EVAL-ADXL354-355-UG-1030.pdf. [Accessed: 29-Apr-2020].

[11] "AD623ANZ Analog Devices: Mouser," *Mouser Electronics*. [Online]. Available: https://www.mouser.com/ProductDetail/Analog-Devices/AD623ANZ?qs=/ha2pyFadug5IpovjBszOn14xFG6pX81o7KL7Zrk9VQ=. [Accessed: 27-Apr-2020].

[12] "MATLAB," *MathWorks*. [Online]. Available: https://www.mathworks.com/products/matlab.html. [Accessed: 29-Apr-2020].

[13] "TMS320C5535," *TMS320C5535 data sheet, product information and support | TI.com*. [Online]. Available: https://www.ti.com/product/TMS320C5535. [Accessed: 29-Apr-2020].

[14] "TMS320C6727," *TMS320C6727 data sheet, product information and support | TI.com*. [Online]. Available: https://www.ti.com/product/TMS320C6727. [Accessed: 29-Apr-2020].

[15] E. A. Mann, "Regulatory Requirements for Hearing Aid Devices and Personal Sound Amplification Products ," *U.S. Food and Drug Administration*, Feb-2009. [Online]. Available: https://www.fda.gov/media/75418/download. [Accessed: Nov-2019].

[16] G. Popelka and R. Margolis, "Bone-Conduction Calibration," *Seminars in Hearing*, vol. 35, no. 04, pp. 329–345, 2014.

[17] "Deafness and hearing loss," *World Health Organization*, Mar-2019. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/deafness-and-hearing-loss. [Accessed: Nov-2019].

[18] "Code Composer Studio (CCS) Integrated Development Environment (IDE) CCSTUDIO This product has been released to the market and is available for purchase. For some products, newer alternatives may be available.," *CCSTUDIO Code Composer Studio (CCS) Integrated Development Environment (IDE) | TI.com*. [Online]. Available: http://www.ti.com/tool/CCSTUDIO. [Accessed: 29-Apr-2020].

[19] "Multisim Live Online Circuit Simulator," *NI Multisim Live*. [Online]. Available: https://www.multisim.com/. [Accessed: 29-Apr-2020].

[20] G. Watson, S. Silva, T. Lawless, J. Harling, and P. Sheehan, "Bone anchored hearing aids: a preliminary assessment of the impact on outpatients and cost when rehabilitating hearing in chronic suppurative otitis media," *Clinical Otolaryngology*, vol. 33, no. 4, pp. 338–342, 2008.

[21] "Signia Pure Primax," *HearStore*. [Online]. Available: https://hearstore.com/signia-pure-primax/. [Accessed: Nov-2019].

[22] "Signia Silk NX," *HearStore*. [Online]. Available: https://hearstore.com/signia-silk-nx/. [Accessed: Nov-2019].

[23] "Now You Can Hear 85-90% Better Without Paying $5,000 For Hearing Aids!," *Heartech labs*. [Online]. Available: https://us.heartechlabs.com/02/. [Accessed: Nov-2019].

[24] "BN-802 Headset," *Bonein.net*. [Online]. Available: http://www.bonein.net/en/product/1.html. [Accessed: Nov-2019].

[25] "Bone Conduction Headphones," *AfterShokz*. [Online]. Available: https://aftershokz.com/. [Accessed: Nov-2019].

[26] "LAUNCHXL-F28377S," *Texas Instruments | Programmers, Development Systems | DigiKey*. [Online]. Available: https://web.archive.org/web/20160604030248/digikey.com/product-detail/en/texas-instruments/LAUNCHXL-F28377S/296-42484-ND/5404239. [Accessed: 26-Apr-2020].

[27] "1304," *DigiKey*. [Online]. Available: https://www.digikey.com/product-detail/en/adafruit-industries-llc/1304/1528-1865-ND/6618716. [Accessed: 27-Apr-2020].

[28] "B3F-1022 Omron Electronics: Mouser," *Mouser Electronics*. [Online]. Available: https://www.mouser.com/ProductDetail/Omron-Electronics/B3F-1022?qs=sGAEpiMZZMsgGjVA3toVBBU0c2ex/tqc0UD9K90M980=. [Accessed: 27-Apr-2020].

[29] "9B-10.000MAAJ-B," *DigiKey*. [Online]. Available: https://www.digikey.com/product-detail/en/txc-corporation/9B-10-000MAAJ-B/887-1010-ND/2118694. [Accessed: 27-Apr-2020].

[30] "LD1117AV33," *DigiKey*. [Online]. Available: https://www.digikey.com/product-detail/en/stmicroelectronics/LD1117AV33/497-1485-5-ND/586006?utm_adgroup=Supplier_STM&utm_source=google&utm_medium=cpc&utm_campaign=Product_SKU_E&utm_term=ld1117av33&utm_content=Supplier_STM&gclid=Cj0KCQjwj7v0BRDOARIsAGh37ip6v11OztsuujGJsniOcwKeKQEv1EZskOcGLbLMkP27kktkzdLzIpkaAoHiEALw_wcB. [Accessed: 27-Apr-2020].

[31] "CC0402KRX5R7BB154 Yageo: Mouser," *Mouser Electronics*. [Online]. Available: https://www.mouser.com/ProductDetail/Yageo/CC0402KRX5R7BB154?qs=sGAEpiMZZMsh%2B1woXyUXj8PMfHfoZZHBoH6OWBunObY=. [Accessed: 27-Apr-2020].

[32] "1N4148," *DigiKey*. [Online]. Available: https://www.digikey.com/product-detail/en/on-semiconductor/1N4148/1N4148FS-ND/458603?utm_adgroup=Diodes - Rectifiers -

Single&utm_source=google&utm_medium=cpc&utm_campaign=Shopping_Discrete Semiconductor Products_NEW&utm_term=&utm_content=Diodes - Rectifiers - Single&gclid=CjwKCAjw4pT1BRBUEiwAm5QuR5vMQDzD0nAYfM7GPGfLaoVyDF8K79TOC_vG2V72eYiLvLebtMPS8RoClqkQAvD_BwE. [Accessed: 27-Apr-2020].

[33] "RK73H1JTTD1000F," *DigiKey*. [Online]. Available: https://www.digikey.com/product-detail/en/koa-speer-electronics-inc/RK73H1JTTD1000F/2019-RK73H1JTTD1000FTR-ND/9844846. [Accessed: 27-Apr-2020].

[34] "RK73B1ETTP512J," *DigiKey*. [Online]. Available: https://www.digikey.com/product-detail/en/koa-speer-electronics-inc/RK73B1ETTP512J/2019-RK73B1ETTP512JTR-ND/9843991. [Accessed: 27-Apr-2020].

Appendices

Included is a summary list and links to all additional files that are stored in the LOCO BOCO project's Github repository, https://github.com/cchoate6/EnginHears.git. These files can be used to reproduce or continue the project.

Software Deliverables

- Audio File Simulation
    - This is a MATLAB simulation that demonstrates adding the ski-slope filtering algorithm on conversations that simulate hearing loss.
    - The file 'enginHearDemo.m' is the main file to run the simulation. The conversations pre- and post- filtering will be played aloud.
    - Several graphs analyzing the characteristics of the signal will be output for analysis.
        - controlVolume.m
        - createPowerSpectrum.m
        - enginHearDemo.m
        - freqshape.m
        - viewAudio.m
- Audio Files
    - audio files used to test the effectiveness of the ski-slope filter
    - _Filtered is added to files post-filtering and _50 indicates the max gain was set to 50
        - Conversation-Mild.mp3
        - Conversation-Moderate.mp3
        - Conversation-Normal.mp3

        - Conversation_Mild_Filtered.wav
        - Conversation_Moderate_Filtered.mp4
        - Conversation_Moderate_Filtered.wav
        - Conversation_Moderate_Filtered_50.wav
        - Converation_Normal_Filtered.wav

- <u>DMA</u>
    - This folder contains documents that discuss relevant example files to set up DMA, as well as the final desired set-up.
    - 'adc_soc_continuous_dma_cpu01.c' is the main file that will be used as a template to set up DMA.
        - adc_soc_continuous_dma_cpu01.c
        - buffdac_sine_dma_cpu01.c
        - DMA_custom_setup.c
- <u>FFT_IFFT</u>
    - This folder contains the Visual Studio Project Solution that includes the modular C functions.
    - Example implementations of an FFT, iFFT, and ski-slope filter are demonstrated and output to the console.
    - The example input data was selected because they had known FFT values, ensuring the FFT algorithm operated correctly.
        - FFT_IFFT.c
- <u>Initialization_Integration</u>

    - This folder only included the main C file used to test the initializations for the microcontroller. It contains a simple notch filter that filters out signal noise. This is the file that was run in the demo of hardware and software integration. To run this file, replace 'watchdog_cpu01.c' with this file in the Microcontroller_IO folder. They require the same additional libraries.
        - Initialization_Integration_Test.c
- <u>Microcontroller_IO</u>
    - This folder houses the initialization code needed to make the microcontroller operational.
    - 'watchdog_cpu01.c' is the main file that has the I/O initializations and the general layout for the filter implementation. The gain calculation is included in this file, but requires the addition of an FFT/iFFT and DMA to be operational.
        - watchdog_cpu01.c
- <u>TMS320F2837sX_Microcontrollers_Documents</u>
    - This folder contains the TMS320F2837xS Microcontrollers datasheet
        - tms320f28377s.pdf
- <u>Real Time Simulation</u>
    - This folder contains the MATLAB simulation for real-time filtering applications. This simulation was mainly used for debugging purposes as a way to test noise, but it does show the filter applied in real-time.
        - HearingAid.m
        - freqshape.m

Hardware Deliverables

- <u>Case</u>
    - This folder contains 3D design files for the LCBC case assembly.
    - All of the 3D SolidWorks files for the 9th Iteration of the case assembly are in this directory. Mechanical drawings of the CAD files for the 9th Iteration are in the Drawings directory STL files necessary to recreate the 9th iteration of the case assembly (ready for 3D printing) are available in the STL directory.

- <u>Circuit Simulation</u>
    - This directory contains files describing the proposed analog filter implementations as well as their simulated magnitude plots. In addition, plots of the magnitude responses of the transduction components are also included.
    - Schematics of the proposed analog circuit are available in the Schematics directory. Plots of the magnitude responses of the proposed analog filter networks, and transduction components are included in the plots directory.

- <u>Gerber Files</u>
    - This folder contains all of the necessary files in order to have the hardware system printed circuit board manufactured. This includes drill files, copper layers, silkscreen layers, board profile and more. These Gerber files were created using Autodesk Eagle.