Mesh Network Headphones

ECE445 Final Document

Team 12
Aditya Bawankule
David Hickox
Alex Ortwig
Abigail Starr
TA: Dhruv Mathur
05/08/2020

Table of Contents

Abstract

Currently, audio splitting is only available in AUX format, and a Bluetooth signal cannot be split into two. The originally proposed solution involved a puck with one Bluetooth receiver and two transceivers to split the audio stream. The solution presented in this document is a Bluetooth Headset, that can connect to similar headsets over proprietary connection in order to share its audio. Unlike the original solution, no additional device is required (ie the puck), and more than two devices can be connected to the network.

1. Second Project Motivation

1.1 Updated Problem Statement

When using an audio system, it is very difficult to come up with an organized and presentable solution that offers a quality listening experience. Such a system is even more challenging when there is a desire to make it portable and shareable with friends and family. Currently, there are a few ways to listen to audio with others, including speakers, sharing one set of earbuds, an AUX splitter, etc. However, speakers are loud and don't fit all occasions, sharing earbuds is unsanitary [1], and AUX as a technology is being phased out.

Bluetooth has made its mark on recent technology as the dominant close-range wireless communication protocol in the electronics industry. However, Bluetooth is typically limited to one connection for audio, which means users who wish to share audio with friends are left unable to do so. A portable, modular, and shareable audio system that lets consumers take their music where they want to and share it with who they want to, all without the clutter of cables and other hardware restrictions is what is needed.

1.2 Updated Solution

The original solution involves a puck which the user would carry around. Each puck has two Bluetooth transceivers and one Bluetooth receiver. An audio playing device, such as a cell phone or other Bluetooth enabled listening device would connect to the puck's receiver, which would then pass the signal to both of the transceivers. Each transceiver would be connected to a unique listening device, such that essentially both listening devices are connected to the same playing device. This solution is sub optimal because each puck not only has to be carried around, but also can only connect to two devices, there is no room for three or more listening devices to be connected.

Instead of being in a puck, the solution presented in this document is a set of headsets. Each headset has the capability to connect to the playing device over bluetooth. Then, in order to get more than one headset connected to the same audio stream, any additional set will be able to connect to the first one over a 5GHz proprietary connection. Each additional headset that joins the area will automatically connect to the network of existing headsets. It will connect what it determines to be the best headset to connect to, based off of signal integrity. The key advantages of the new solution over the old solution include the lack of need of a separate device and the ability to connect more than two distinct devices in total.

One example for the use case of this solution would be a classroom environment. For example, a teacher may want to play a video but for whatever reason could not play the audio aloud. The existing way to solve this is to have each student play it individually on their own device, but that brings up its own complications, of each student not having their own device or students getting distracted while on their own device. The proposed solution would allow the teacher to play some audio and have each student listen in a silent environment with minimal setup. Once the master headset is paired to a Bluetooth device, the slaves will connect automatically, and all play the same audio.

Other products exist in the marketplace that function using the same theory as the original group's implementation. Examples would include the Monoprice 109722 [2] and the Anker Soundsync [3], both of which are available commercially and utilize the Bluetooth 5.0 protocol. These commercially available devices do not have the capability to connect to more than two Bluetooth audio devices at the same time. Our product intends to extend the quantity of Bluetooth devices that can be connected to the host audio source by implementing a mesh network instead of utilizing the two-device feature seen in commercial products.

1.3 Updated High-Level Requirements

- Can support listening for at least 3 slave devices in addition to the master
- Can provide audio at full volume for at least 8 hours on battery charge
- Can support a distance of at least 3m from the Bluetooth device to the acting master headset, as well as from the acting master headset to any one of the acting slave headsets.
- System must support high data rate PCM signals (192khz PCM)

1.4 Updated Visual Aid

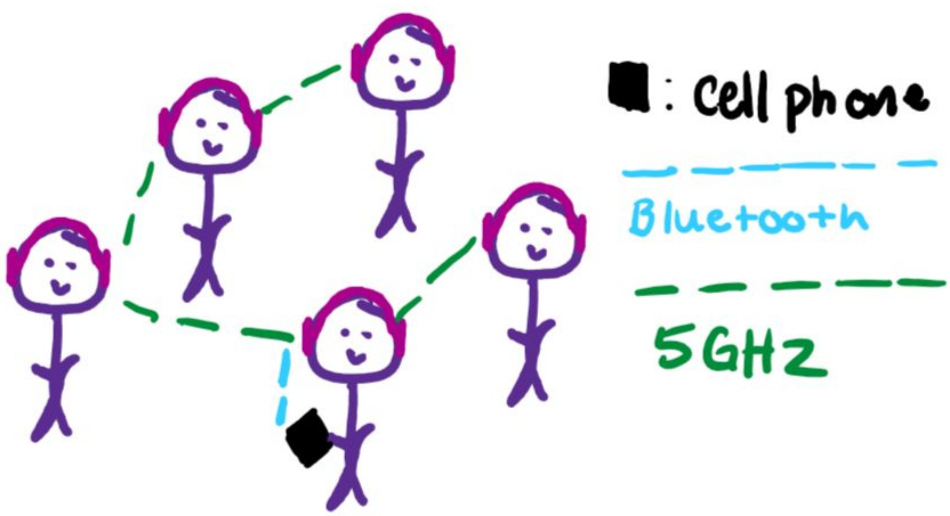

Figure 1. Pictorial Representation One headset is connected to cell phone via bluetooth and the other connected to the original via 5GHz connection. Additional headsets can be connected through 5GHz

1.5 Updated Block Diagram

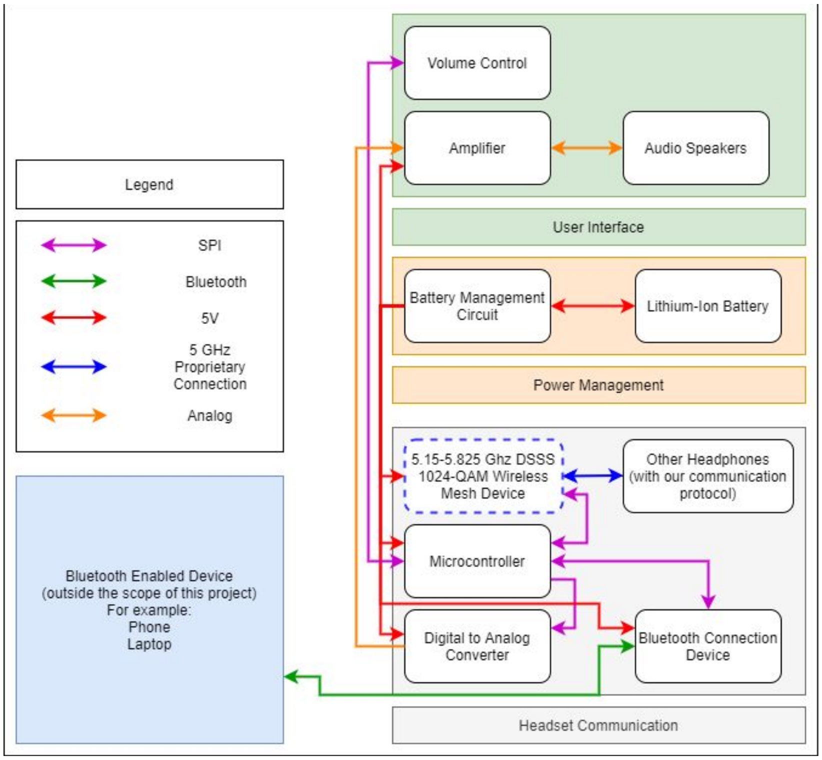

Figure 2. Block Diagram A summary of how different sub-blocks connect to each other

2. Second Project Implementation

2.1 Hardware Development

A PCB was designed according to the details represented above. Pictures of the layout can be found in Appendix C Section 1. This is a two layer board designed to fit inside of a headset. Dimensions were obtained from a headset laying around, in which the circular portion was approximately a circle with diameter of 3". This was then made the shape of the board, with a flat side so that it can be placed in the headset without less worry as to the orientation placed. Two M2 screw holes are also provided for security on one side. The other side has the outputs for the speakers, so it can be glued to the housing along with the wires.

This PCB has passed Design for Manufacturability checks by JLCPCB, so the PCB could theoretically be ordered if chosen to continue with this project. A breakdown of the parts checked in the DFM check can be seen in Section 2 of Appendix C. In addition, the Bill of Materials is included in Section 3 of Appendix C, with every part for the PCB only. The cost of the PCB in total would be the $2 JLCPCB order and the $45.83 Digikey order.

Overall, this board will accomplish all of the submodules except for the battery charging and speakers. Because there is a connector open for batteries it is still possible to have rechargeable batteries with this implementation. Additionally, there are connectors to connect the speakers needed on this board, so this will accomplish the speaker module assuming those are connected.

Given the maximum power draw of 278.22 mW by drivers, a 95 percent efficiency on the amplifier with a base operating power of 6mW making the driver power draw 278.22/.95 + 6 = 298.86 mW. The power draw of the DAC is a maximum of 210mW. The maximum draw from the microcontroller given the expected data rate is 348.47 mW. The total power of the headphones comes out to a maximum of ~855.33mW , a required endurance of 8 hours at full volume, and a battery voltage of 3.7V, with a buck boost efficiency of 95%, the minimum battery capacity needed is equal to $8/95\% * \frac{P}{V} = 1.947Ah$ or 7.20 Wh. This has made us choose the Lithium Ion square battery 103450 (standard size) it is 3.7v and 2Ah and some models come with overcharge protection which makes our circuit cheaper. This battery can be attached to the PCB and charged separately.

The Bluetooth and wireless mesh radio are built into the chosen microcontroller, the ISM43907. This is a different microcontroller than the one specified in the design doc, for the very reason that it has built in bluetooth and wifi (5GHz) connectivity. This new microcontroller fits the same specs as the old one, but the overall device count is smaller because of the built in capabilities. The SPI line on the microcontroller is connected to the DAC in addition to the I2C line. Because the Bluetooth and radio are built in, there is no need for additional SPI lines.

New boost and buck converters were chosen to go from the battery voltage to 3v3 and then from 3v3 to 5v. The new boost converter, the TPS61022, operates at 1.03MHz and its approximate efficiency is 94.7%. Similarly, the new buck converter, the TPS56637, operates 560.68KHz and its approximate efficiency is 92.9%. Both of these calculations were done using TI's webench program. These were chosen for their efficiency. One of the drawbacks for both of these chips is their nonstandard footprints. This increased the time to design with them because it was necessary to make the footprint by hand. Pictures of these footprints can be seen in Appendix C, Section 1, in the picture of the front of the layout.

2.2 Mesh Network Software

Mesh Networks can operate in two modes, flooding and routing. Our typical scenario of listening to audio will involve the flooding mode, in which every device sends the information it receives to every device except for the one that it received it. We will be using time stamped audio samples, and our microcontrollers are fast enough to sort through all the extra data. A few ms of delay while processing may occur, this is within tolerances for comfortable listening and avoiding desync. It should be noted that the processing occurs after packet rebroadcast not before so it will not affect daisy chain latency. It is also possible to do a hybrid data flood mode entailing a little bit more time between receiving and sending data which increases daisy chain latency slightly but prevents the packet sorting latency which with many nodes can quickly overburden the processors and start affecting even daisy chain latency.

```
while 1
    Wait for Packet
    if incoming packet's timestamp is already past or is already in the system
        discard packet
        break
    else
        rebroadcast packet to all neighbors
        add packet to audio playback stack, where it can be sorted by timestamp and played back
```

Figure 3. Psuedocode for mesh network operating in Flood mode

In the routing mode, packets have a destination device, and have to be routed through the network so they can reach their destination. Our device will use this mesh network mode during certain important events such as authentication and delivering software updates.

For optimal results, packets need to be directed through the network so they can reach their destination in the minimum amount of time. Each device will have to locally store and update a file of every connection it has, and pass that file onto other devices. Each device will have an overview of the entire network, and can use that to make intelligent network routing decisions. This is known as a Link-state algorithm.

```
neighbors [] = getConnectedDevices //stores neighbors (with signal strength) of local device
networkconnections[thisDeviceID] = neighbors //store connections of all devices on network
broadcast(networkconnections)
if (incomingPacket.isOfType(networkconnections))
    updateConnections(incomingPacket) //updates this devices local view of all devices
```

Figure 4. Psuedocode for mesh network operating in Routing mode

The device can use the network information to construct a weighted graph of the network, with the weights being calculated using the signal strength and latency between the devices. The signal strength and latency will be measured by the accuracy of the QAM signal with respect to its symbol page and expected error and with the round trip ping time respectively. Our Mesh Network will use the A* graph traversal algorithm in order to determine optimal connections between devices on the network.

```
open[]
close[]

open.insert(node.start)
while(open.isNotEmpty)
    find(node_current with lowest f)
    if (node_current == node_goal)
        return node_current //found our soln
    node_sucessors = generateSucessors(node_current)
    foreach node temp in node_sucessors
        temp.cost = g(node_current) + w(node_current, temp)
        if(open.contains(temp))
            if g(temp) <= temp.cost
                continue
            else if(closed.contains(temp))
                if g(temp) <= temp.cost
                    continue
                open.add(temp)
                closed.remove(temp)
            else
                open.add(temp)
                node_goal.distance = h(temp)
        closed.add(temp)
    if (node_current != node_goal)
        break //means open list is empty
```

Figure 5. Pseudocode for A* routing used for determining optimal path through the network

2.3 Connection and Latency Research

Despite the lack of ability to test with Bluetooth and our 5GHz connection, research has been conducted on the difference between the two connections. One of the biggest concerns is latency, because that would be noticeable to users if the audio is supposed to be synced between devices, but the latency causes it not to be synced. In order to compare this solution to something available in the real world, the assumption has been made that the 5GHz proprietary signal presented in this solution would be better than a 5GHz wifi signal. Since this is for a more specialized purpose than wifi, it will have a lighter protocol, leading to lower latency. However, due to the lack of ability to test this, it is unknown whether there will actually be lower latency and if so, how much lower. Therefore the assumption has been made that it will, worst case, have the same latency as the 5GHz wifi signal. The median latency on a 5GHz wifi signal is .9ms [5].

It is important to also notice that the median latency on a 2.4GHz wifi signal is 6.22ms [5], as this is much higher than the 5GHz counterpart. One of the tradeoffs between these two signals, besides latency, is range. According to CenturyLink, the least range between either 2.4GHz or 5GHz is 62ft [6], and this is well within the high level requirement of 3m for this project. Therefore, because the latency is lower and \the range does not take an impact on this project, using the 5GHz is preferable over using a 2.4GHz connection.

Similarly, it is now necessary to consider using a Bluetooth mesh network. The Bluetooth protocol used here, aptX codec, has a latency of 150-200ms [7], which is around 200x worse than 5GHz wifi on average. Because the minimum range on bluetooth is 100m [8], it does not provide any distinct advantage over 5GHz wifi, and is significantly worse in latency, so using the 5GHz for the connection is the best option.

2.4 Specific Operations of 5 GHz Broadcast

The system will broadcast and receive using 1024-QAM or lower signals (Quadrature Amplitude Modulation with the number indicating number of distinct amplitude levels, eg 1024 is equivalent to 10 bits per period). Each pair of headphones will take a wireless band of a maximum width 20MHz between 5.150 GHz to 5.850 GHz. The headphones will optimize the mesh to allow smaller bands depending on signal clarity and the lack of other interferences. Best case scenario is a 1024-QAM signal hopping over 1MHz-2MHz bands. Worst case would be a 64-QAM signal over a 20 MHz band. Both of these signals would have the data rate capable of transmitting the audio signals needed but the wider band signals are much more resilient to interference and are easier for other competing 5GHz systems to detect and avoid. The smaller bandwidth signals provide less of an obstacle for other 5GHz systems causing less disruption to existing infrastructure, it also allows a larger and more dense mesh network. In the case that interference is problematic this system will also be capable of DSSS (Direct Sequence Spread Spectrum) which allows them to widen the signal by intentionally introducing pseudorandom noise making it extremely hard to interfere with. This does at minimum double the required bandwidth. Though this is still well within our tolerable range for bandwidth.

3. Second Project Conclusions

3.1 Implementation Summary

The hardware design is done, but not the implementation, as that would need the ability to purchase parts. First, came the choice of which parts to use by David. The design of the schematics and the power calculations for the battery were done by Alex. Abby then took these schematics, and ran the calculations for the buck converter and boost converter, and then ran the PCB through the DFM checks. On the hardware side, everything is ready to be purchased and assembled if we were to have that ability. As far as software goes, Aditya and David have been working on the mesh network algorithms.

3.2 Unknowns, Uncertainties, Testing Needed

On the hardware side, there exists a lot of work to be done yet. The first step would be to order the PCB and associated parts, as well as assembly. Once that is done, it would allow us to test the subsystems as well as the system as a whole. Unfortunately due to the circumstances, parts are unable to be ordered so none of this testing can be carried out.

Once hardware testing is able to happen, unit tests are laid out in Appendix A below for each sub component. The only blocks that do not require connection to the 445 lab are: volume control, the microcontroller, and the bluetooth connection. The rest of the blocks would require lab equipment such as the Oscilloscope to test. Even the ones that can be tested outside of the lab require the lab equipment to solder the board together, as they are small smd components which are not easy to solder.

Without access to radio equipment, we are unable to test a large part of the software required for this project, specifically most of the software related to mesh networking. While certain portions could be simulated, testing for interference and other unintended side effects will be very difficult without hardware.

3.3 Ethics and Safety

We intend to follow the IEEE Code of Ethics 7.8.1-7 and 7.8.9-10 [9]. Some of the Code of Ethics is not applicable due to the nature and methods of our project. In the case of any violations of this code we will take them seriously. Due to the physical nature of our device we do not foresee any sort of physical injury possible without extreme misuse. For this reason we will focus on the nonphysical component when designing and operating our device according to the Code of Ethics 7.8.1 [9].

It is important to ensure that our device will not harm the user's ears, or emit excess radiation as per FCC regulation. Our device is classified as an intentional radiator by the FCC [10]. The chosen frequency is well within the FCC regulations, so any radiation emitted is within the healthy limit [11] that will not negatively impact the users quality of life any more than any other consumer devices. Additionally, as with any battery charging, it is necessary to ensure that the battery will not catch on fire, either when the battery is charging or during normal use. To solve the charging issue, a standard off the shelf charging system will be used that is UL rated in order to limit the amount of power that can enter the system at any one time.

To prevent harming the users ears we will ensure that our internal amplifier is limited to being able to drive any common off the shelf headphone speaker to a maximum of 130dB (roughly the SPL of a rock concert) with a warning at 80dB to alert the user that long term effects could occur. This works out to anywhere from 3mW on earbuds to 16,000 mW for the highest impedance large form factor headphones. This exact value will be chosen based on our voice coil selection specifically [12]. We will also not exceed the maximum voice coil rated power and current in order to prevent destruction of our products and burns to the customer.

3.4 Project Improvements

One improvement that can be made is having the battery charge over USB instead of having to take it out to charge it. This would drastically improve the usability of the headphones, as well as the durability because there are less moving parts.

A second improvement would be to make our own headset, instead of buying existing headset like the current plan is. This would allow for the reduction of cost as commercially available headsets are expensive and marked up due to not buying them wholesale. It would also be easier to then design the headset around the electronics, to be more space efficient and lightweight this way.

A third improvement would be to increase the number of headsets in the mesh. Currently it is software limited at 8 in order to keep the mesh network algorithm light and fast. In order to increase the maximum number of headsets, we would need to increase the size of the data structures holding the network connections, potentially causing slowdowns and latency issues if the processor is not able to update the data structures as quickly. Adding more devices will increase the number of incoming streams when in flood mode, which could overload the processor if not handled cleanly. While the additional devices would increase the size of the node-graph of the network, only the building of the graph may present issues as traversing it is not typically performance-intensive, but building it using the list of devices and their connections will take additional time when more devices are present.

4. Progress Made on First Project

- Main Hub Board
  - Broke the PCB into two boards
  - Pictures of the new PCBs are included in Appendix D Section 1
  - Results from DFM check from Bay Area Circuits also in Appendix D Section 1
- Received and Assembled at least one locker daughter pcb
  - Communicated to the ADC using the STM32's SPI interface
  - Took voltage readings using the ADC
- Quoted, Ordered, and received all necessary parts.
  - Pictures of Parts in Appendix D Section 2
- Assembled Lockers
  - Pictures of Lockers in Appendix D Section 3
- Started working on main locker CAN and RTOS code as well as setup compiler to compile for our board layout
  - Code Screenshots in Appendix D Section 4

5. References

[1] B. Gibson, “Here's Why You Should Never, Ever Share Earbuds with Anyone,” The Healthy, 26-Nov-2017. [Online]. Available: https://www.thehealthy.com/ear-nose-throat/never-share-earbuds/. [Accessed: 03-Apr-2020].

[2] Suellen, “Monoprice 109722 Bluetooth Transmitter and Splitter - Other Bluetooth,” Walmart.com, 19-Feb-2017. [Online]. Available: https://www.walmart.com/ip/Monoprice-109722-Bluetooth-Transmitter-and-Splitter-Other-Bluetooth/39457824. [Accessed: 04-Apr-2020].

[3]https://www.amazon.com/Anker-Soundsync-Transmitter-Synchronization-Connection/dp/B07BR3FFTB/ref=sr_1_16?dchild=1&keywords=bluetooth+splitter&qid=1585967638&sr=8-16

[4] M. Mansouri, M. Meghdadi, and R. A. Moghaddam, “Achieving high-dependability optimal multicast in Wireless Mesh Networks,” 2011 IEEE 13th International Conference on Communication Technology, 2011.

[5] I. Grigorik, “Performance of Wireless Networks: WiFi - High Performance Browser Networking (O'Reilly),” High Performance Browser Networking, 27-Apr-2016. [Online]. Available: https://hpbn.co/wifi/. [Accessed: 09-May-2020].

[6] “What's the difference between 2.4 GHz and 5 GHz WiFi?,” CenturyLink. [Online]. Available: https://www.centurylink.com/home/help/internet/wireless/which-frequency-should-you-use.html. [Accessed: 09-May-2020].

[7] “Audio latency with Bluetooth technology,” Audio latency with Bluetooth technology. [Online]. Available: https://www.silabs.com/community/wireless/bluetooth/knowledge-base.entry.html/2015/08/06/audio_latency_withb-7BGG. [Accessed: 09-May-2020].

[8] J. G. Sponås, “Things You Should Know About Bluetooth Range,” Welcome to our blogs. [Online]. Available: https://blog.nordicsemi.com/getconnected/things-you-should-know-about-bluetooth-range. [Accessed: 09-May-2020].

[9] “IEEE Code of Ethics,” *IEEE*. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html.

[10] “Equipment Authorization – RF Device,” *Federal Communications Commission*, 20-Mar-2018. [Online]. Available: https://www.fcc.gov/oet/ea/rfdevice. [Accessed: 04-Apr-2020].

[11] “Wireless Devices and Health Concerns,” *Federal Communications Commission*, 21-Feb-2020. [Online]. Available: https://www.fcc.gov/consumers/guides/wireless-devices-and-health-concerns. [Accessed: 04-Apr-2020].

[12] “Headphone Sensitivity / Power Requirements Compared” *Head-Fi*, 28-Feb-2003. [Online]. Available: https://www.head-fi.org/threads/headphone-sensitivity-power-requirements-compared.476345/. [Accessed: 04-Apr-2020].

Appendix A: Requirements and Verification Tables

Submodule	Requirement	Verification
Volume Control	Ensure that the volume button buttons will increase and decrease the volume of the device’s sound.	1. Press the increase volume button and verify that the output of the media is louder than before. (Provided that the system did not start at maximum volume) 2. Press the decrease volume button and verify that the output of the media is quieter than before. (Provided that the system did not start at minimum volume)
Amplifier	Have a Total Harmonic Distortion of no worse than -80 db	1. Send several frequencies into the amp from the DAC 2. Test the integrity of the signal with an oscilloscope and use it to programmatically calculate the Total Harmonic Distortion
Audio Speakers	Have a +-3db response from 20hz to 20khz	1. This can be tested using a calibrated microphone and a noise isolation box. This would require a frequency sweep to identify the sound levels. 2. This is not necessary for the most part due to the drivers we purchased guaranteed from the manufacturer to be

		responsive within 1db for 3hz-30khz.
Battery Management	Successfully charges the device without causing harm to the battery, device, or user.	1. Connect the linear regulator to a 5V input, and a resistive load on the output. 2. Ensure the linear regulator maintains a constant 1A (+/- 0.2A) current output through the load.
Lithium-Ion Battery	Starting from full charge, the battery is able to output 860mW (our system maximum power draw) for 8 hours continuously without sagging below 3.3V, the battery's minimum allowed voltage.	1. Connect a fully charged battery to an electronic load set to draw 860mW 2. Allow the battery to discharge into the electronic load for 8 hours. 3. Ensure that the battery has not fallen below its minimum rated voltage with a multimeter while under the above load.
Proprietary Connection	Can join a mesh network	1. Pair a second pair of headphones to a bluetooth device so it enters broadcast mode 2. Turn on the pair of headphones and verify enters into the mesh network and mirrors the audio when told to enter the mesh network
	Can broadcast a mesh network for other headsets to join	1. Pair this set of headphones to a bluetooth device 2. Turn on a second set and verify that it can enter the network and correctly mirrors audio

Microcontroller	Ensure that the microcontroller works.	1. Connect a programmer to the JTAG breakout for the microcontroller. 2. Verify that the microcontroller can be successfully programmed.
	Ensure that the microcontroller can communicate with each part of the system correctly.	1. Send a test program that validates each connection independently 2. Send final product program that demonstrates all systems working together.
Digital to Analog Converter	The DAC can convert a digital signal to an analog output with a signal to noise ratio of at least 80 db and within 7 bits of accuracy (0 - reference current) including overshoot.	1. Send a known digital signal into the DAC. 2. Use an Oscilloscope to observe that the DAC outputs the correct analog waveform within expected error and accuracy.
Bluetooth Connection Device	Can connect to a modern smartphone (Bluetooth device)	1. Turn on headphones and enter pairing mode 2. Verify the device appears in the smartphone's pairing menu, and select it 3. Verify the device plays Bluetooth audio from the smartphone

Appendix B: Schematics

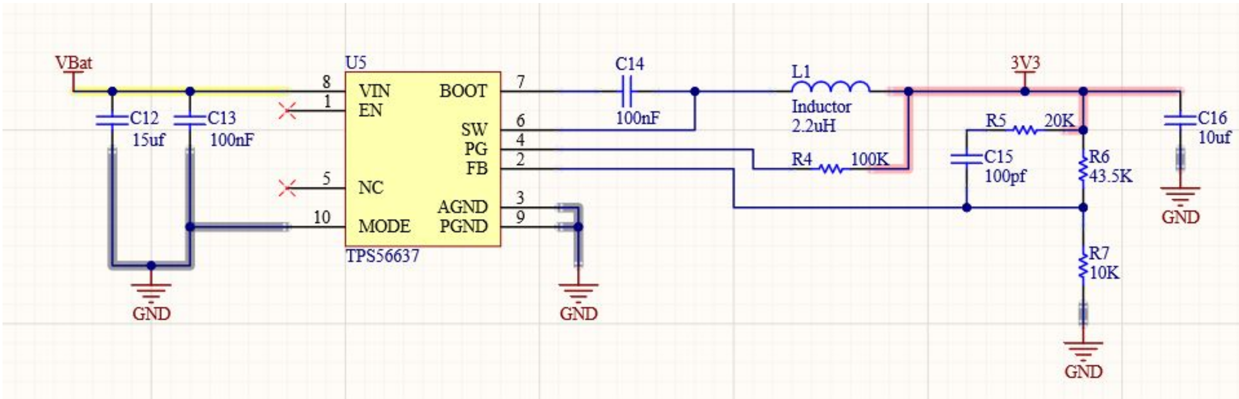

Figure 6. Buck Converter Takes the battery voltage down to operating voltage of 3.3V.

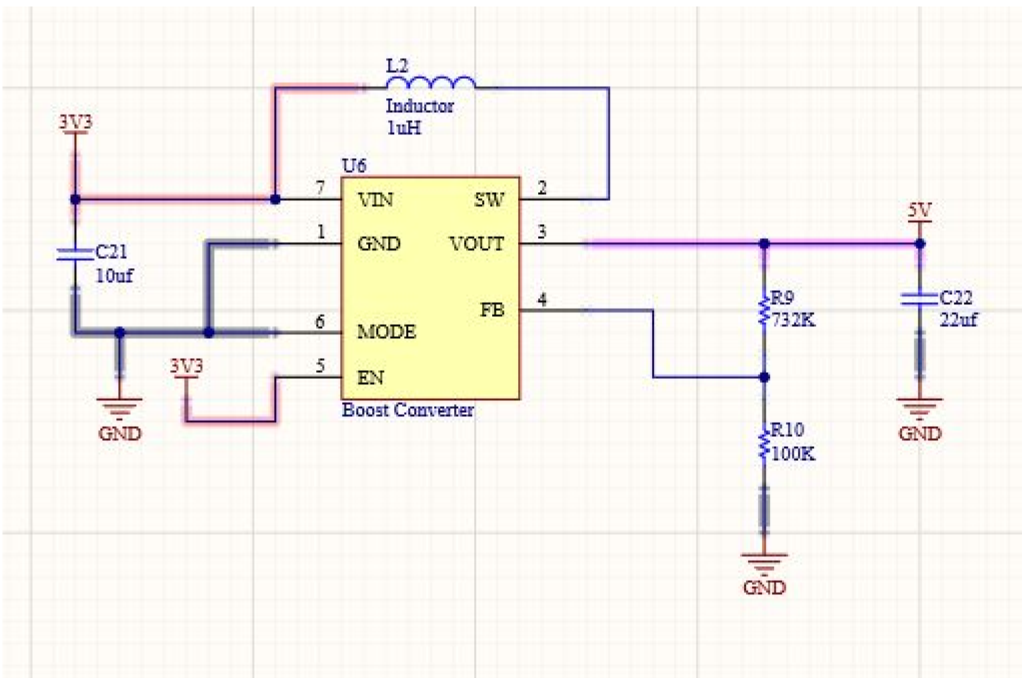

Figure 7. Boost converter This goes from 3.3v to 5v. It is a boost instead of a buck converter due to the fact that if the 3v3 line is not working, then the 5v line should not work either.

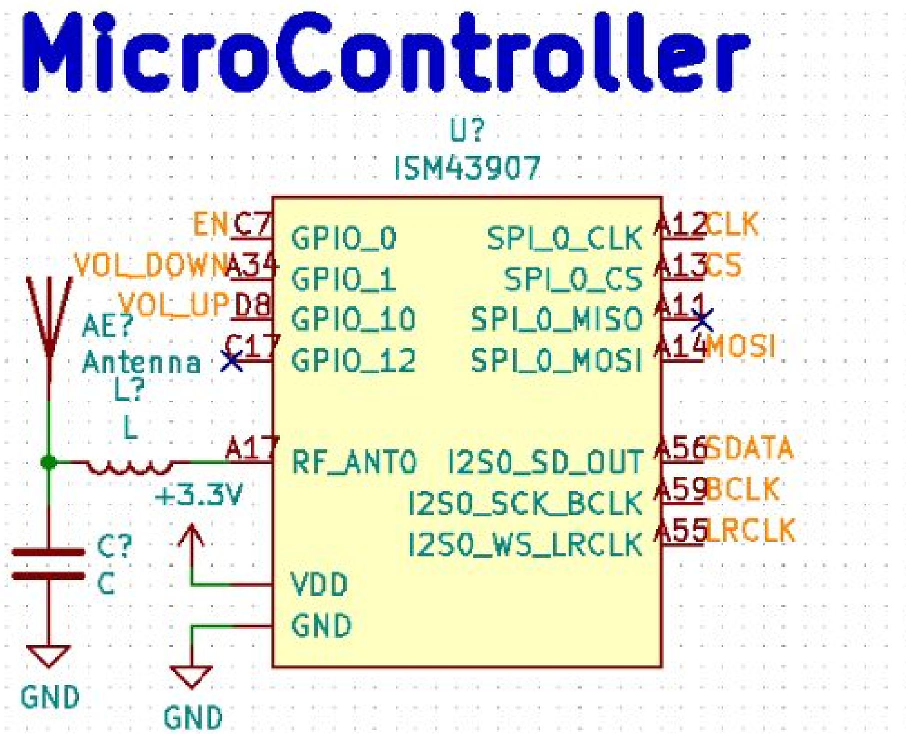

Figure 8. ISM43907 Microcontroller with built in bluetooth and wifi transmission

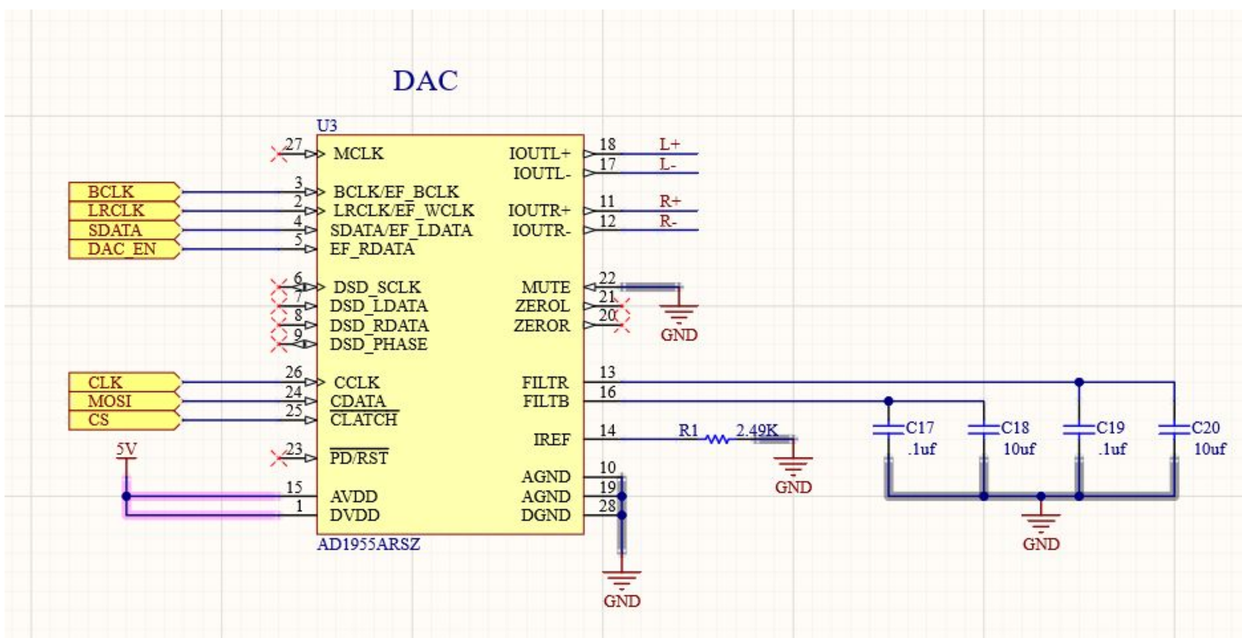

Figure 9. DAC The digital to analog converter will take the data from the amplifier, and convert it to an analog waveform to give to the speakers.

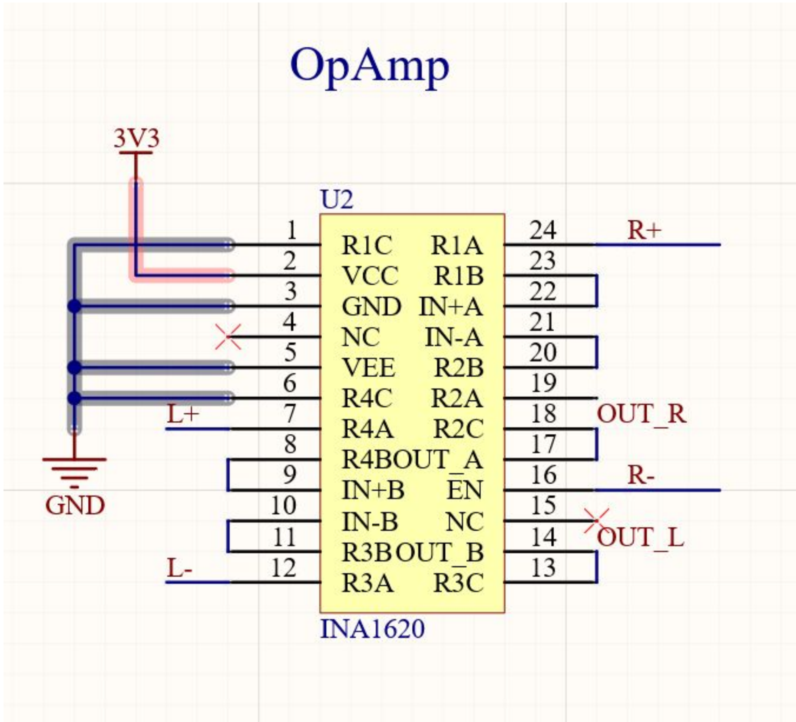

Figure 10. Op Amp The Op Amp serves as the amplifier and takes its data directly from the DAC and from the microcontroller move to the correct volume

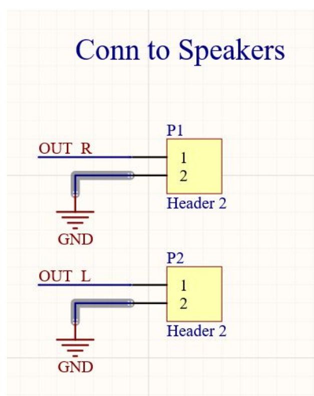

Figure 11. Speaker Connectors Because the speakers are not surface mount parts, there must be connectors going off the board to connect to.

Appendix C: Printed Circuit Board

Section 1: PCB Pictures

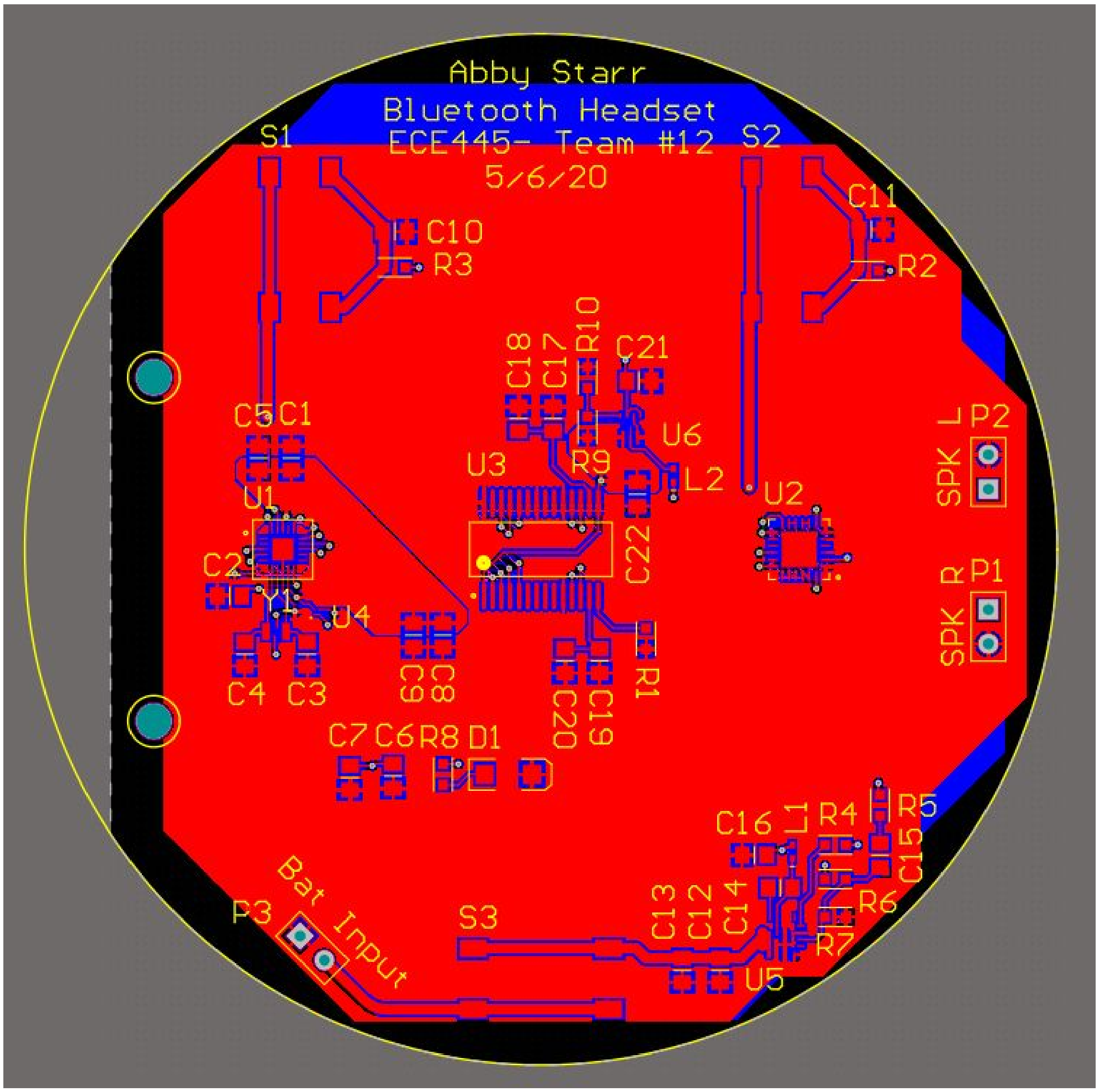

Figure 12. 2 layer PCB with the schematics above.

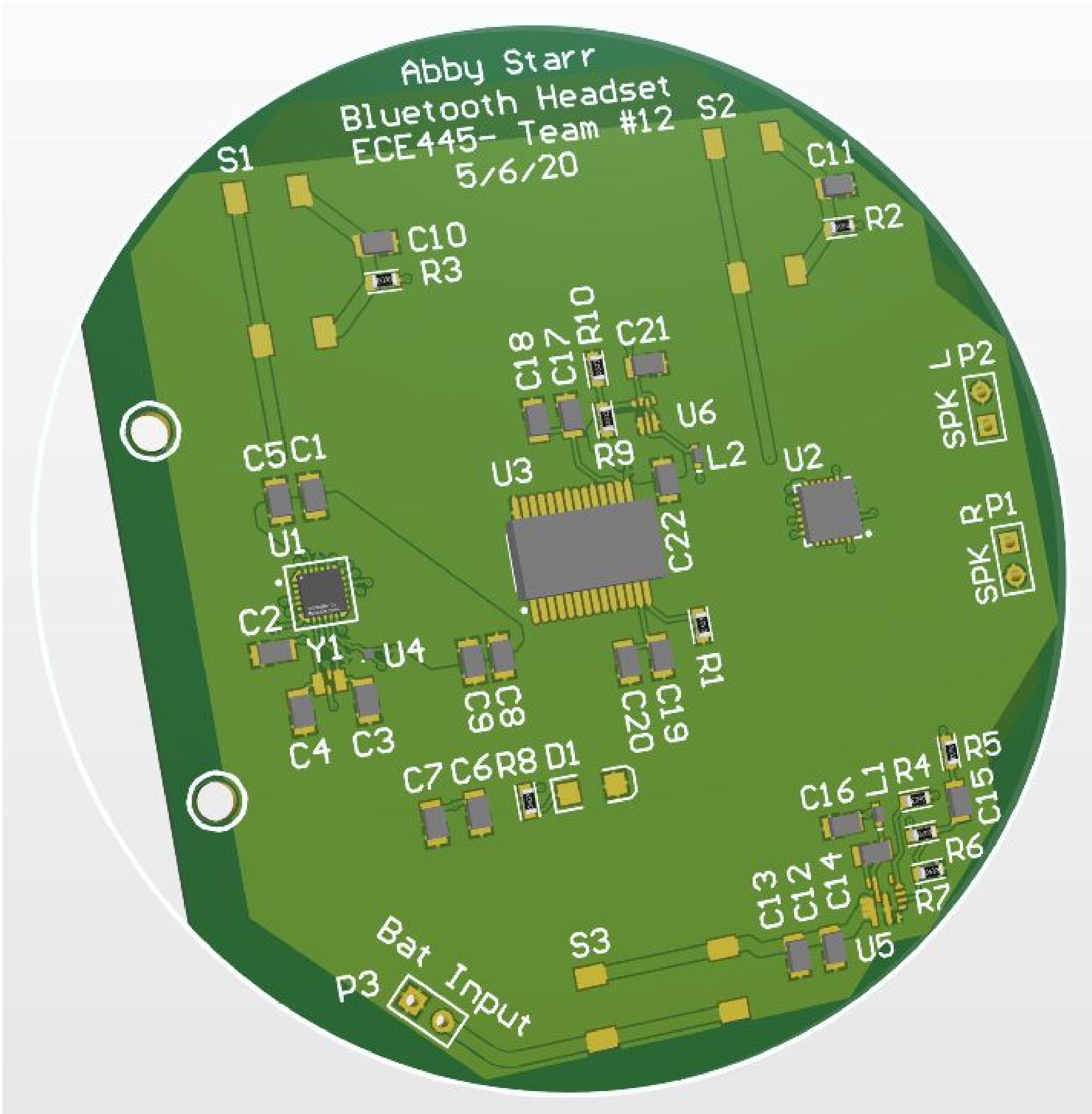

Figure 13. 3D representation of the board in the headset

Section 2: DFM Checks

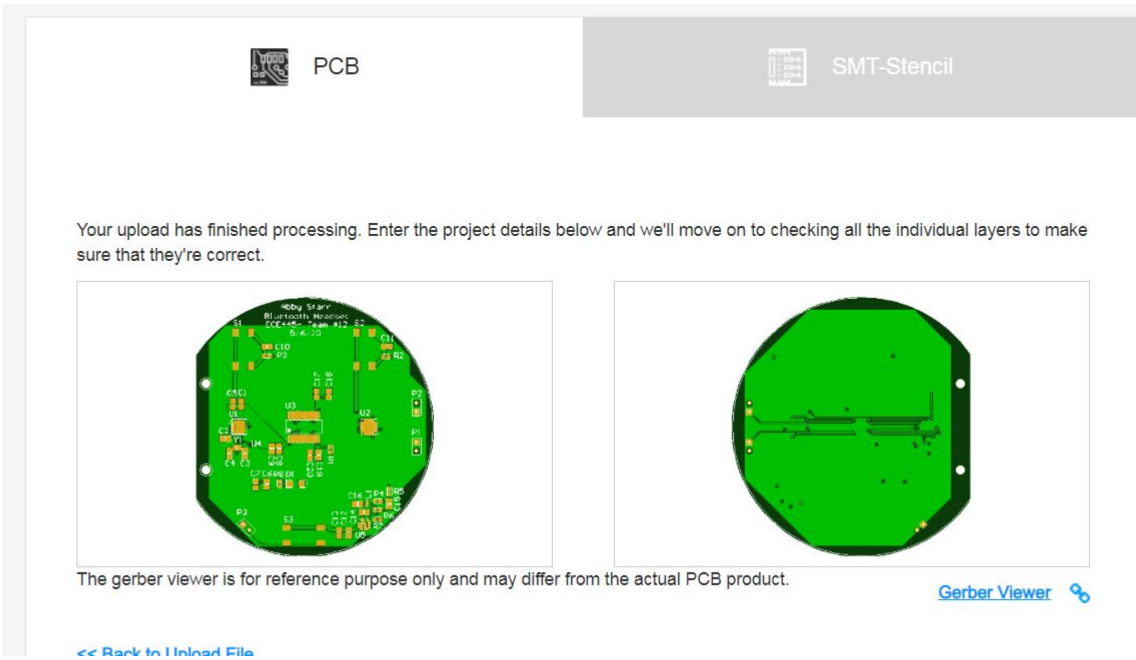

Figure 14. Screenshot of Board in the JLCPCB order screen, proving its ready to order

Figure 15. Screenshot of DFM analysis on board done in JLCPCB

Section 3: Bill of Materials

Description	Value	Quantity	Price
Capacitor	1uf	4	0.1
Capacitor	10pf	2	0.1
Capacitor	.1uf	5	0.12
Capacitor	10uf	4	0.17
Capacitor	4.7uf	1	0.1
Capacitor	15uf	1	0.1
Capacitor	100nF	2	0.1
Capacitor	100pf	1	0.1
LED		1	0.36
Inductor	2.2uH	1	0.45
Resistor	2.49K	1	0.24
Resistor	100	2	0.36
Resistor	100K	1	0.17
Resistor	20K	1	0.1
Resistor	43.5K	1	0.1
Resistor	10K	1	0.13
Resistor	330	1	0.28
Push Button		3	0.7
ISM43907		1	12.37
INA1620		1	6.41
AD1955ARSZ		1	14.85
LP5907		1	0.54
TPS56637		1	3.5
SiT1533		1	1.13

Figure 16. Bill of Materials for the PCB This only includes things that are directly soldered to the board. The total price for all of these components is $45.83

Appendix D: Original Project

Section 1: Keypad/LCD PCBs

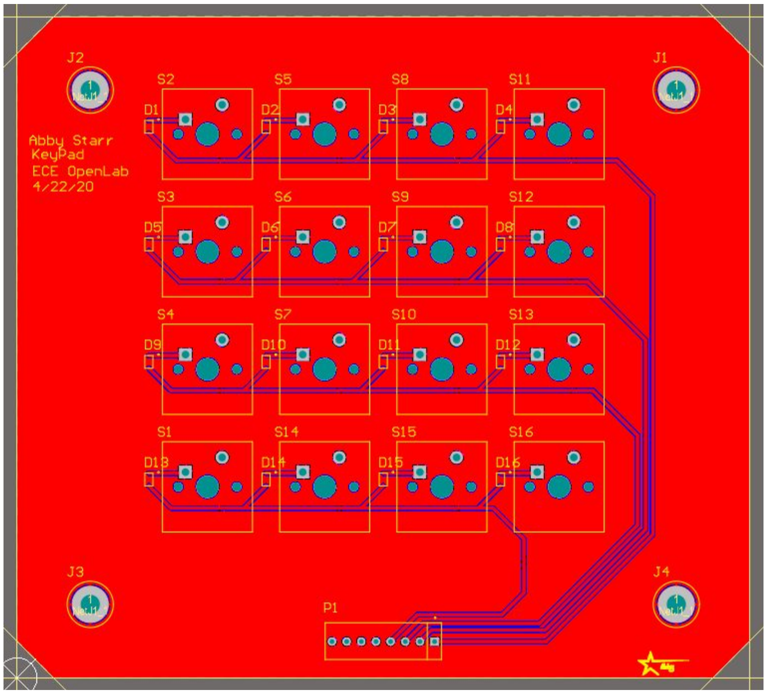

Figure 17. Keypad PCB 2D 2-Layer PCB for the just the keypad portion. Four holes around the outside fit with main micro board to mount together

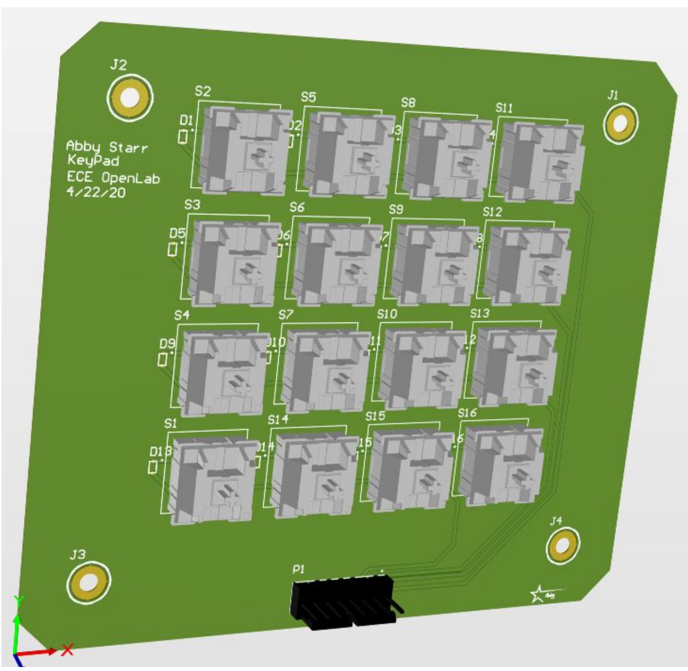

Figure 18. 3D Keypad Board Simulated keypad board to show spacing of keys

DFM Checks

	Min. Distance	Bay Area Circuits Standard Capability	Bay Area Circuits Advanced Capability
copper to board edge:	40.0 mil (1.02 mm)	10 mil	5 mil
trace to plated hole:	15.83 mil (0.4021 mm)	10 mil	5 mil
trace to non-plated hole:	5.0 mil (0.13 mm)	8 mil	4 mil
copper to copper:	5.0 mil (0.13 mm)	6 mil	2 mil
copper ring:	10.83 mil (0.2751 mm)	6 mil	1 mil
track width:	10.0 mil (0.254 mm)	6 mil	2 mil

Min. Distance values are color-coded according to Bay Area Circuits' Standard & Advanced capabilities. Green = Standard Capability, Orange = Advanced Capability, Red = Not Within Capability

Figure 19. DFM for Keypad Board Board fits within the advanced capabilities of Bay Area Circuits

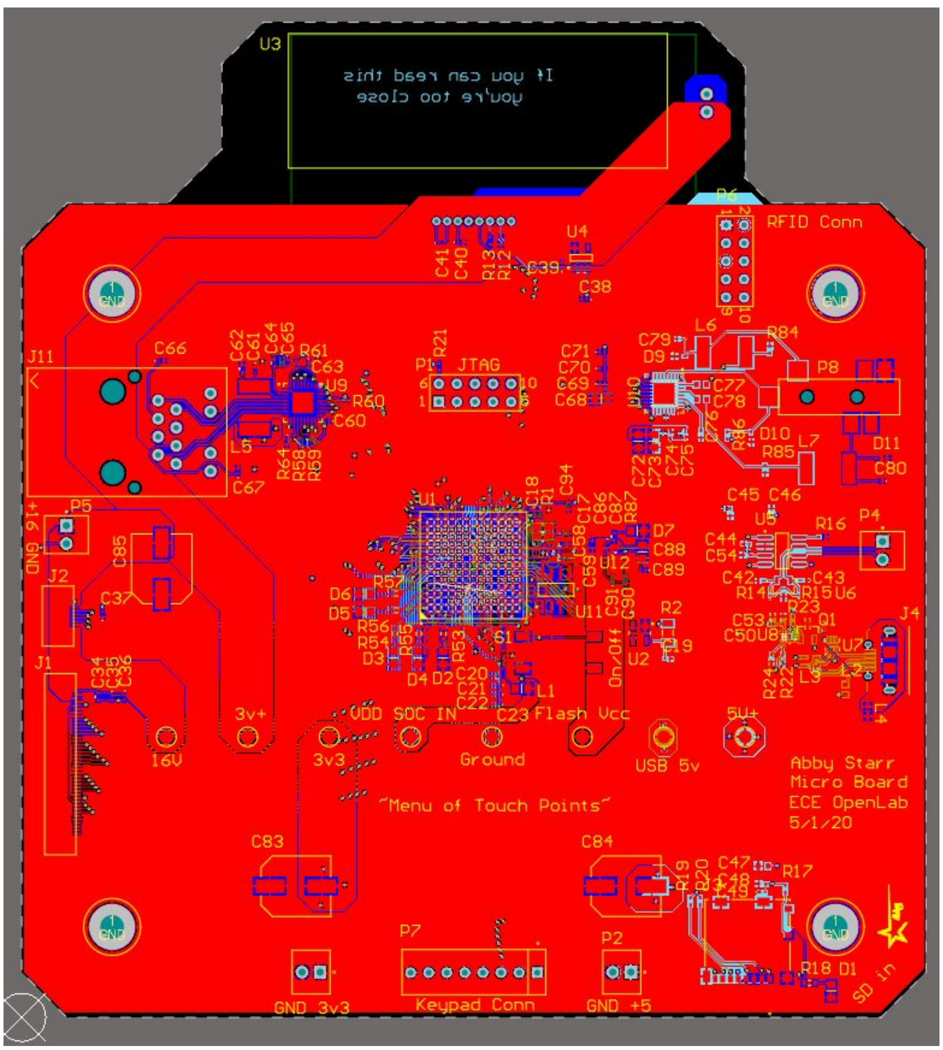

Figure 20. Main Microcontroller Board View from top down of 4 layer board. Holes around the outside fit with the keypad board to mount together

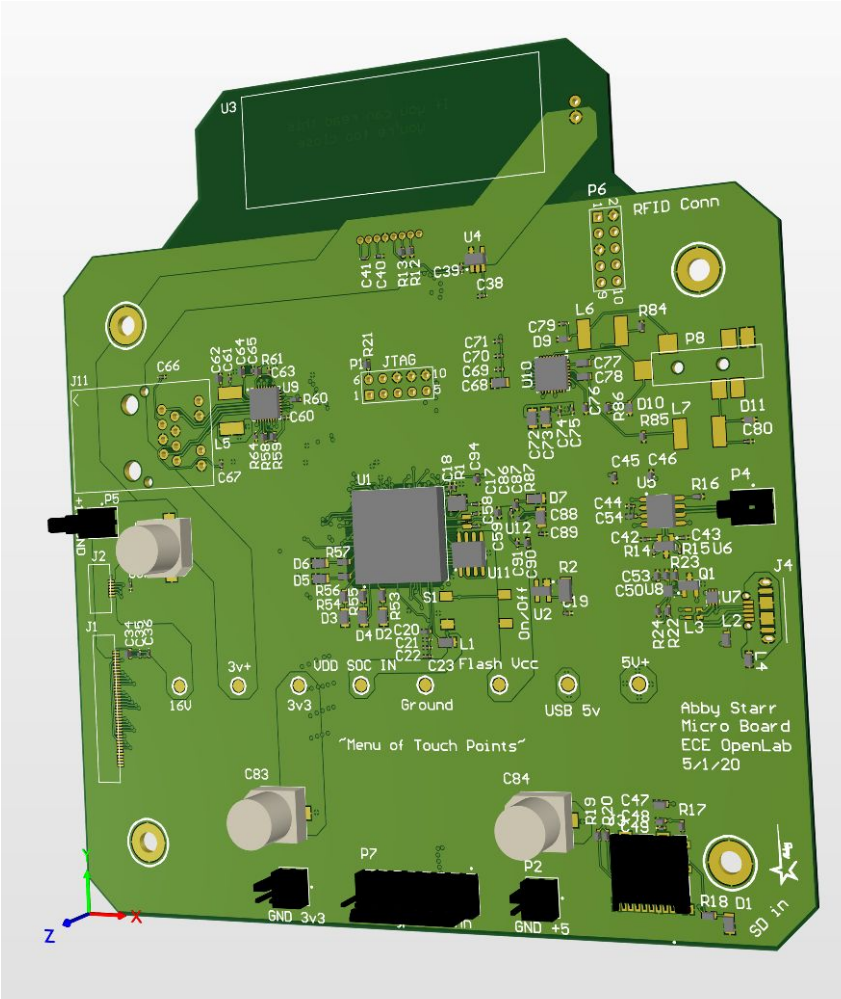

Figure 21. 3D Main board 3D representation to see spacing and physical design

DFM Checks

	Min. Distance (outer layers)	Min. Distance (inner layers)	Bay Area Circuits Standard Capability	Bay Area Circuits Advanced Capability
copper to board edge:	39.0 mil (0.991 mm)	13.0 mil (0.330 mm)	10 mil	5 mil
trace to plated hole:	10.0 mil (0.254 mm)	10.0 mil (0.254 mm)	10 mil	5 mil
trace to non-plated hole:	-	-	8 mil	4 mil
copper to copper:	5.0 mil (0.13 mm)	5.0 mil (0.13 mm)	6 mil	2 mil
copper ring:	3.94 mil (0.100 mm)	3.94 mil (0.100 mm)	6 mil	1 mil
track width:	5.19 mil (0.132 mm)	5.93 mil (0.151 mm)	6 mil	2 mil

Min. Distance values are color-coded according to Bay Area Circuits' Standard & Advanced capabilities. Green = Standard Capability, Orange = Advanced Capability, Red = Not Within Capability

Figure 22. DFM Check for Main Micro Board Passes BAC advanced capabilities

Section 2: Locker Board PCB’s

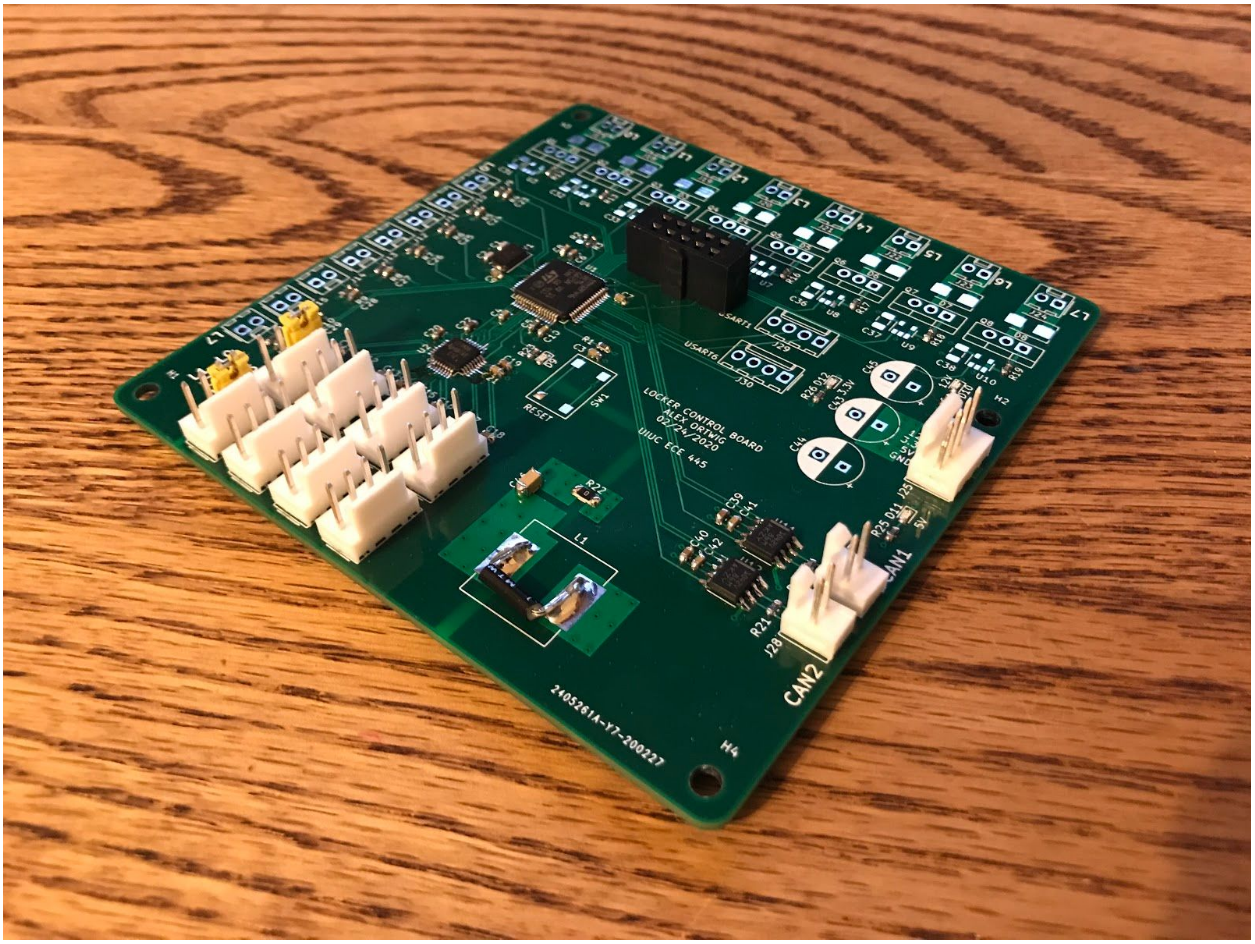

Figure 23: Soldered Locker Control Board

The locker control board was ordered and partially assembled. Initial testing included programming the microcontroller, interfacing with the ADC over SPI, and taking voltage measurements using the ADC.

Section 3: Pictures of Lockers and Components

Figure 24. Locker PCB Components

Figure 25. Locker Components

Figure 26. Small Locker Assembled

Figure 27. Large Lockers Assembled

Figure 28. NXP Dev Boards for Testing Code

Section 4: Locker Code

```
  // foreground loop
  while(1)
  {
    // Operate on CANopen protocol stack, slave
    MCO_ProcessStack();
    // Operate on application
    USER_ProcessApp();

    if (MY_NMT_STATE == NMTSTATE_OP)
    { // Only work on manager when we are operational
      // Operate on CANopen protocol stack, manager
      MGR_ProcessMgr();
      // Operate on custom CANopen manager application
      USER_ProcessMgr();
    }

    if (do_nmtreset && MCOHW_IsTimeExpired(nmtreset_delay))
    { // after delay, reset all slave nodes
      // restart all nodes
      MGR_TransmitNMT(NMTMSG_RESETAPP,0);
      do_nmtreset = FALSE;  // only do it once
    }

  } // end of while(1)
} // end of main
```

Figure 29. Basic Can Open test implementation that works on our board. There is much more code behind this but this is the main loop.

```c
int main(void)
{
    /* Init board hardware. */
    BOARD_ConfigMPU();
    BOARD_InitPins();
    BOARD_BootClockRUN();
    BOARD_InitDebugConsole();
    if (xTaskCreate(hello_task, "Hello_task", configMINIMAL_STACK_SIZE + 100, NULL, hello_task_PRIORITY, NULL) !=
        pdPASS)
    {
        PRINTF("Task creation failed!.\r\n");
        while (1)
            ;
    }
    vTaskStartScheduler();
    for (;;)
        ;
}

/*!
 * Task responsible for printing of "Hello world." message.
 */
static void hello_task(void *pvParameters)
{
    for (;;)
    {
        PRINTF("Hello world.\r\n");
        vTaskSuspend(NULL);
    }
}
```

Figure 30. Test implementation of Amazon FreeRTOS. Once again much more code behind this, this is just the task and area that task is scheduled from.

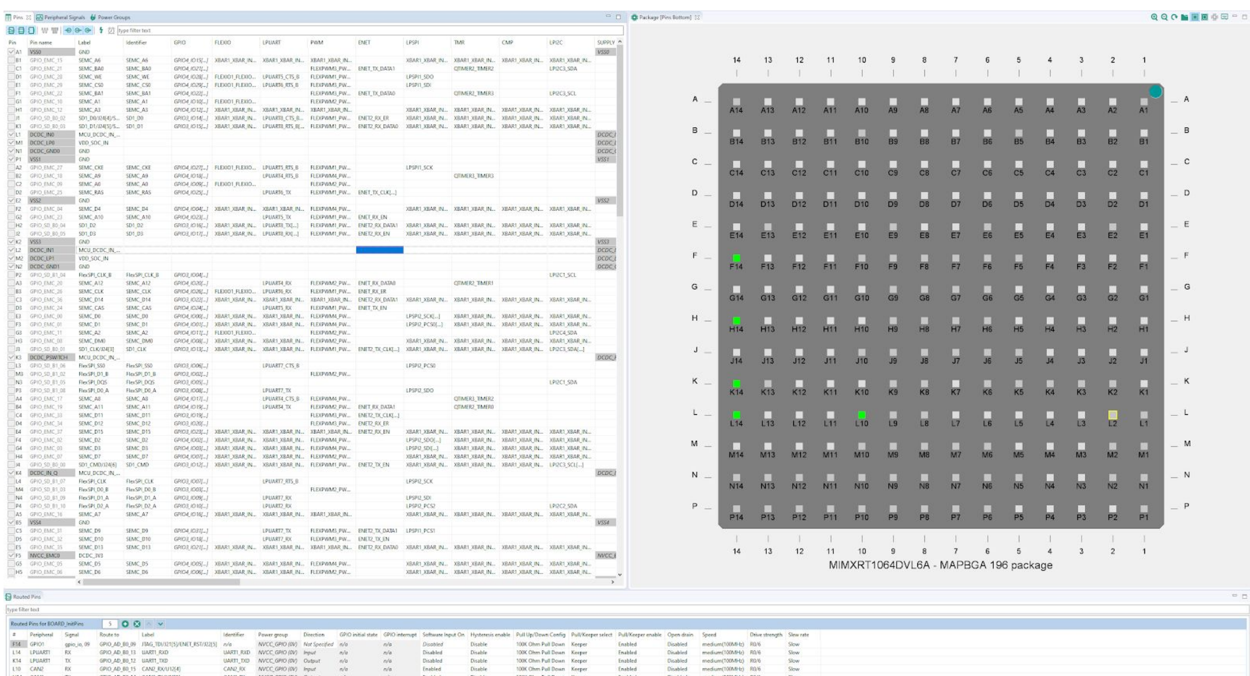

Figure 31. Pin Config for our MCU. In accordance with our preliminary PCB design, currently highlighting CAN pins