Project Documentation
VERSION 2.0

Project
MINUTEMAN

Network of audio sensors capable of detecting gunshots.

University of Central Florida EECS
Senior Design 2
Dr. Lei Wei & Dr. Samuel Richie

GROUP 4

Nathan Thomas Cunanan - Computer Engineering - nathancunanan24@knights.ucf.edu
Nathaniel Dunn - Electrical Engineering - nathanieldunn992@knights.ucf.edu
Daniel Vicenti - Computer Engineering - dvicenti96@knights.ucf.edu
Eric Watson - Computer/Electrical Engineering - ewatson2@knights.ucf.edu
Adam Bush - Computer Engineering – adambush@knights.ucf.edu

Table of Contents

1. Executive Summary

For years, our country has witnessed countless massacres committed by mass shooters. The public response to these events has varied but have almost always been the same in terms of policy. Gun control is advocated for and pushed in legislation, but resistance is presented by those who oppose these measures, especially as they relate to gun rights. To be clear, this project does not attempt to answer which of these stances is correct.

However, it is clear that after all this time, it is very unlikely we are going to find effective legislation that everyone will all agree with. A technological solution to this problem is both more approachable, agreeable, and therefore necessary. Not just because most individuals can agree on it but because it can reliably respond to these events and save lives regardless of legislation issues.

Our society has advanced to a stage that technological solutions to difficult problems are becoming a reality almost every day. Already several solutions to the issue of mass shootings have been revealed such as door blocking systems [1], inflatable slides allowing occupants on the upper floor of a building to escape [2], and more. These solutions are not divisive, inexpensive, and are signs that the market is capable and willing to create more technological solutions to this issue in the future.

This project will aim to fulfill three primary goals. The first goal is to provide a solution that can provide rapid but accurate detection of a firearm discharge within an operational area. The second is to rapidly inform authorities about the discharge and allow a rapid response. It is the hope of this project that it will become a starting point that will enable other technological solutions to the threat of mass shooters.

This project was inspired by a previous Senior Design project called Project GLASS [3] which was an outdoor audio detection system that could accurately pick up the sound of a gunshot, discern the type of firearm used, triangulate the location of the gunshot, and then report that information to authorities.

2. Project Description

The features of our project were to create an audio sensor system that can be installed around a facility or onto a vehicle. This system is to accurately detect gunfire discharges, being able to discern them from other sounds in the environment such as engine backfires or glass breaking. It is to be able to discern the general direction of the gunfire with respect to the device. Then it is to inform the authorities.

This system in its basic form will be affordable for most facilities and organizations such as schools, public buildings, or places of large public gatherings. Specifically, the applications are for public safety, law enforcement, and military use. However, this can be an affordable addition to any private facilities who have concerns about active shooter situations.

2.1 Goals and Objectives

The project fulfilled the following goals and/or objectives as shown in Table 2-1:

A single device with several microphones to pick up nearby audio.
The device can be connected to a host computer and utilize any building or vehicle power supply.
A microcontroller to connect the sensors and stream audio to the host computer program.
A program/app that can be downloaded to a host computer to manage the device and performs the majority of the software functions.

Table 2-1: Goals and Objectives

2.2 Requirements

The project was aimed to fulfill the following requirements as shown in Table 2-2:

	Target	**Tested Within**	**Units**
Has a maximum range of	250	160	Meters
Determine the direction of a gunshot within	5	1	Degrees
Identify gunshots with greater than	95%	99.2%	Accuracy

Identify and display gunshot within	2	0.4	Seconds
Costs within	150	100	USD

Table 2-2: Requirements

2.3 Specifications:

The project fulfilled the following specifications as shown in Table 2-3:

The sensors are connected to a microcontroller, which is connected to the host computer program.
The microcontroller streams audio into a byte array on the host computer program.
The program checks the array for the muzzle blast of a gun using a convolutional neural network, searching for specific features in a scalogram created by a continuous wavelet transformation.
The neural network is trained from a dataset of gunshots and non-gunshot audio files of 80ms length.
The network provides the program with a prediction. If it is a gunshot, it will pass the audio array to multilateration program to determine the direction of the gunshot.
Within 2 seconds of the gunshot, the program will send an alert about the gunshot and its direction from the device’s location, as determined by GPS.

Table 2-3: Specifications

2.4 Constraints

The project was held within the following constraints as shown in Table 2-4:

Requires at least one array of audio sensors
All components must share the same clock rate when communicating
There is a required minimum sample rate required for analysis
The entire project cannot cost more than 1000 USD

Table 2-4: The Constraints

2.4.1 Ethical Constraints

Given the nature of the situations a gunshot detection system will be required in, there are some ethical constraints to consider. While a system failure is not always within the designer's ability to manage, at worst a failure in this case can potentially be catastrophic for several lives, especially if they are expecting the system to be reliable in this regard. At the least, it can create false alarms, wasting potentially thousands of dollars of resources in responding to a situation that is not in fact dangerous. Therefore, making the system as reliable as possible in correctly identifying shootings and doing so as rapidly as possible is a design constraint.

There are also other ethical constraints to consider with regards to privacy. As this is a device that nearly always listens, efforts should be made to prevent individuals from using the device to spy on others.

3. Project Management

Within our group, we planned on which programs we were going to use to produce organizational duties within each person and make sure we do our priorities right and have a to-do list for each one. The team was split up into hardware and software, so we utilize apps like Trello or Google Drive to store our documents and keep updated with each other pertaining to research tools and references regarding our project. We used Google Drive mainly to store links to articles we have found all throughout the depths of the internet. Trello was also used to organize our priorities, milestones, and deadlines for any major priority we had to attend to in Senior Design 1 and Senior Design 2.

We had weekly meetings on a certain day like Friday so we make sure everyone in the team is well prepared for each milestone laid out for us and no one misses a deadline or else there will be a consequence. We also decided on having core values between the group because without these values, everyone will be in disarray or someone has the potential to be lost or he doesn't know what to do a certain stage of the implementation of the design. Trello is a useful tool as it is one of the most popular organizational modes that is also accessible easily and portable because users can use in their mobile phones. Google Drive is a sure shot go-to for document storing and document/spreadsheet creation since it is the most popular out of the bunch that is used by millions of people around the world.

We also used Discord as a means of communication between each other since it has its handy feature of voice chat and screen sharing. We meet up on discord every week to discuss the necessary details that we to deal with for our next milestone. There is constant agreement and we make sure that nothing gets passed through without the approval of everyone in the group. It's expected that everyone has their own fair and share of other obstacles and we expect each other to remind the whole team if something unexpected pops up we have to understand all the circumstances regarding the situation if ever one arises.

There were in-person meeting times that were also arranged prior to the final deadline to work on the design and then the real implementation of the Minuteman system. If a meeting could not be arranged or if members couldn't partake arrangements were made to assess each possible solution to make our flow better as a team. Eventually we agreed to meet up in the senior design laboratory daily to work on the project. Continuous communication and trust were key in maintaining a smooth and successful project.

There are numerous key points that were well discussed in setting up a precise and well-resourced project. Especially given the constraints and requirements of the project. Groups were divided by major to focus on hardware and software elements of the project. Each member was given individual parts to focus on and assist others with.

One of the major risks encountered in the planning for this project was reducing tension between the team due to the reduced time allocation to develop the project. Tension between team-members would have resulted in poor and mismanaged productivity. Planning especially was key, as research and debugging time needed to be allocated in order to compensate for the lack of overall practical experience in the team. There were a number of setbacks in this process, however. The team ran into issues coordinating times and miscommunication. This was somewhat relieved by spending more time working on the project while physically together.

There is a preeminent structure that breaks down project development and planning. A Work Breakdown Structure is a hierarchical tree structure of deliverables and tasks that need to be performed in order to complete a project. This was useful as a foundation for project planning and management. The lowest level of the work breakdown structure is the Work Package. The WP is the allowance of a certain package that can be cost analyzed, resourced, and can be easily scheduled. The WP contains a list of tasks that can be performed that form a basis for the schedule created by the team themselves. This package also allows the fair assignment of the roles for a team. WBS allows build-up of costs and schedules in the form of a hierarchy. This facilitates strong management in project execution. This kind of process was used for other applications such as document and risk management in our project.

While we intended and planned for an excess of time for each stage in case of complications, a number of issues were met that delayed other aspects of the project. Interdependencies, especially including the hardware, which was required for much of the testing and debugging, presented obstacles that prevented progress in other areas. If the project was redone, it's likely that planning around inter-dependencies would be a requirement, and better communication for assisting others in clearing these obstacles.

3.1 House of Quality Table:

Table 3-1 properly illustrates our discussed customer requirements along with the desired technical equivalents into a relationship matrix showing how strong of a relationship two different requirements relate to. There is also a row called Direction of Improvement where it shows how much relative improvement we need for each field and a certain weight on how important each field is for our project.

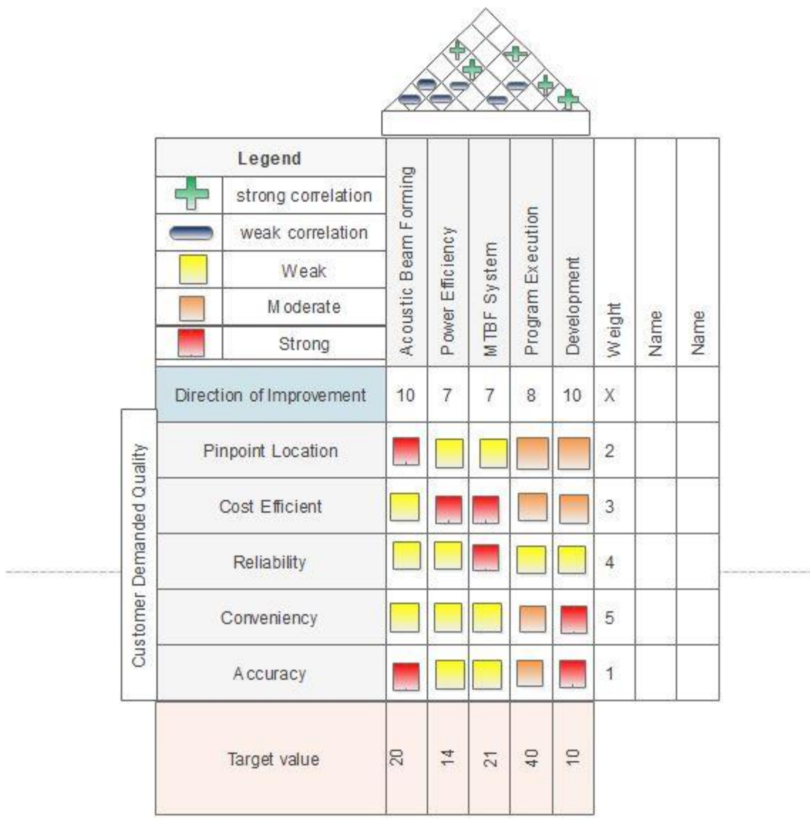

Table 3-1: House of Quality

In our HOQ table, this is intended to compare, and contrast then analyze with determination in both the engineering requirements and marketing requirements for our Minuteman device. The HOQ table is a crucial part of our product development process which is also widely used in the world of engineering. A House of Quality table is a conceptual map that allows for inter-functional planning and communications where it produces a dynamic system of goals to determine which goals are more important than the others and to achieve these goals where in other goals would be affected.

3.1.1 Understanding the House of Quality Table:

There are exactly 5 engineering requirements and 5 customer centered requirements laid out in a matrix like structure. These will then be compared to each other to determine which pairs of relationship extract the most attention and which do not. As can be seen from the House of Quality table above, there are different color-coded regions in the mapping of the table which corresponds to how strong a relationship is between two requirements. The red color stands for a strong relationship which means a pair of requirements correlate with a stronger bond. The orange color stands for a middle ground between requirements meaning that this has a decent relationship while not as robust as a pair with a perfect correlation. Lastly, the yellow color stands for a weak relationship meaning that they do not correlate whatsoever by any means necessary.

With these correlations, these provide the necessary information for customers alike in how our priorities are aligned. For example, Pinpoint location and Acoustic Beamforming have such a tight relationship that there is no more reason to worry about how to stoutly perform that certain task. Compared to the weaker relationships, they tend to be the ones that need more critical attention because they usually are going to be harder to deal with and, in effect, would need to do more focused research into.

The Direction of Improvement row in our HOQ table signifies which pairs would be put in the priority queue first and which come in last. From the table there is a digit that represents the priority sequence for our project. A 10 corresponds to more improvement in a part where a lower digit would correspond to a lesser worry for improvement. For example, Power Efficiency and Pinpoint Location got a 7 which means that those two won't be in need of improving since they really do not offer much relationship with each other because they don't relate to each other at all.

The top side of the HOQ table corresponds to the need for execution in parallel to desired prioritized actions regarding the requirements chosen. There are symbols like a plus and minus sign to fully express the need for prioritization for all the requirements decided upon by the team. A plus symbol designates a strong need for action and a better mindset for future reference because this is needed for necessary enhancements down the line when something bad happens with the product creation. A minus symbol is the ultimate opposite which provides weaker need for attention for future use. The weights in which these aligned is very crucial because these serves as the engineering specifications just perfect for the customer needs and wants or else the relationship with the specific customer will crumble when it matters most. We have decided on which requirements are more suitable for our project extensively so that we have our minds set for what's to come in this journey of creating our Minuteman system.

3.2 Project Block Diagram:

For this project, it was definitely necessary that there was a widespread approval of everyone else's roles in the development and the design stages alike. We have carefully decided on which sectors that we encapsulate for the project so it is properly shown that what every member is doing whether it might be in the power module, central module where a microcontroller is included, sensor module or even the software side of things.

What is a block diagram? A block diagram constitutes a "quick, high level view of a system to rapidly identify points of interest or trouble spots. With this kind of structure, it has the chance to not offer much detail that is required for the likes of comprehensive planning or the implementation of the design. Block diagrams are essentially focused on specific parts of a project such as input or output of a design. This principle is called "black box" in the world of engineering. Block diagrams are similar to making flowcharts but, in this case, it is shown that every block in the diagram is color coded and, according to the professors, that each of the initials of the members of the team so it is clear that the roles are laid out properly. Our block diagram is spread properly as though it looks to be categorized in different modules. There are arrows that indicate where each module traverses into. For example, the Power Control Unit (PCU) module flows through the FPGA / Microcontroller that goes into the user interface module that is led by a variety of combinations of students.

There are varying practices for making these block diagrams enticing and better looking for the readers to have a look at in the future. Firstly, we have to identify the system that we are working on. We have to determine the system of roles that needs to be assigned so that each of the members have a clear look of what needs to be done because of the potential consequences that emerge if everyone thinks they have a concise view of their tasks that need to be completed. We also have to determine which kinds of components that are required to be used, what are the flow of the different modules that are included in the system and what the outputs are for each of the inputs.

Next, we have to create and also label the block diagram. We had to determine carefully what kind of modules are needed for the minuteman system. It didn't take long for us to be sure of what the different pieces are in designing the system because all that really needed to be done was to identify the core components that blends in properly in creating a flaw free prototype for presentation come next semester. The arrows are a crucial part of the diagram because, without them, we wouldn't know what the other modules of the project relate to in terms of the other modules. The relationships between the modules decided on by team is vital in the development process of the project because it will be the deciding factor where the team can process into themselves what the system will look like for the final product. While this is the case, the team has to make sure there is ample accuracy from the professors so as though there will not be risks taken by the group and that each of the components are set properly.

Figure 3-1 shows the diagram, which shows the modules and division of tasks given to each individual member of the group. The blocks were made by looking at the objectives and requirements of the project. Each module represents a necessary function of the system. Each block in the diagram contains a description of the function and an abbreviation of initials. This divulges each group members' tasks for the project and the level of priority they have in that area. The priority is given from right to left.

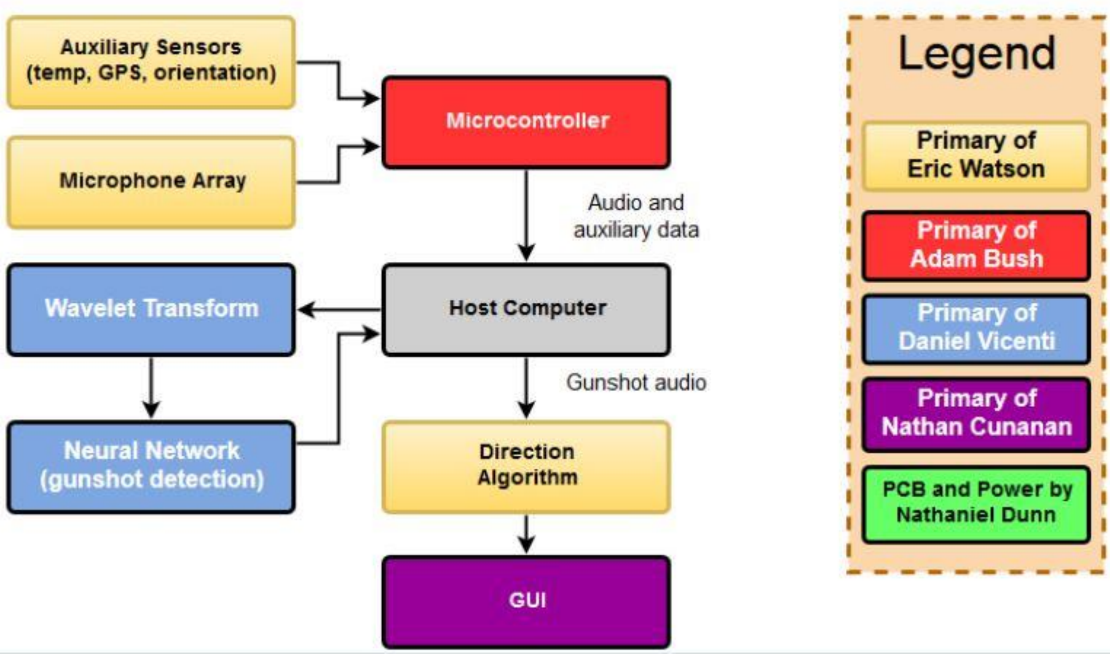

Figure 3-1: Project Block Diagram of Modules and Division of Tasks

3.3 Project Budget and Financing:

For the financing and budget of the project, each part was carefully analyzed and reviewed upon approval for each member of the group so that there is full dependency for everyone. This was one of the main factors in creating the project: money. As students, money is an aspect in current life where it isn't flourishing since all the focus is finishing and getting a degree. The whole team knows that this is going be a big focus in managing the project properly and that money would be an immediate issue because of the lack of it. We had the ideas of implementing sponsors for our project so that we don't have to worry about where to get the financing from but we have decided to stick with getting the finance naturally so, in that way, once our product gets finalized come summer, we will have the path of managing our own business in regards to our Minuteman system without having to depend or give back to our sponsors especially in the share in profits between the companies end and our end as well. No project plan is complete without a concrete plan for our budget. Figuring out the bottom line is what matters and we have to extensively process as a unit on what we are willing to spend based on the parts that we decide to purchase.

With a proper budget, it is important to come up with detailed estimates because it is essential that there is a foundation for where we start from in terms of which parts to get and where to source them as well. While the work is ongoing, it is definitely possible to track the project even if there possible changes that can appear during the design implementation. Our plan was to purchase all the materials needed for the project before the second semester starts because if we procrastinate and get our parts late, we can possibly lose time in the process since we don't have much leeway for our Senior Design 2 time span. The estimate of these costs will vary as the development of the system gets deeper and deeper. We will get a more proper view of what the costs really are as we learn more about the project.

There is an inclusion of resource cost rates. Since there is work needed for the project to be done, there are also rates that need to be calculated. All the materials that is used for the project also are counted for cost rates. The rate of the materials that are purchased will account for the labor needed by the members as well. While there material cost rates, there are also costs of quality. The quality related activities come into play in this kind of project in a way where the quality of the work process of each member is monitored so that the group doesn't get affected by a lack of performance which can result in a bad product. The cost of quality is just a way of tracking the cost of each activity that the team will encounter throughout this whole process. With this, it is super important to keep track of the estimates needed. In this way, we will know the assumptions made when we decided on making the numbers.

With all these steps, we can concur that our project creation will be smoother because we don't have to worry about where to get the money and plus we will learn from the initial makeup of the design that we can eventually change for the better later.

Table 3-2 shows the material costs and quantity of our project, providing some perspective on what will be necessary for the project’s budget. This budget will change as the project gets denser and more as we go further along the journey of making this project. We hope to have a targeted budget of $1000 or less by the time we must purchase the parts.

Material/Part	**Quantity**	**Price**	**Total**
Arduino Due	1	$48.00	$48.00
AT91SAM3X8E	2	$11.54	$23.08
Power Adapter	2	$2.99	$5.98
V Regulator	2	$17.31	$34.62
Inverter	4	$1.43	5.72
Microphone	12	$2.12	$25.44
Gyroscope	1	$34.34	$34.34
GPS Modules	1	$15.99	$15.99
Temp Probe	1	$8.20	$8.20
Amplifier	25	$0.15	$4.20
PCB Costs	20	$5.00	$100.00
Miscellaneous		$50.00	$50.00
		Total	$355.57

Table 3-2: Material Costs and Quantity

3.4 Project Milestones:

Knowing the do’s and don'ts of a senior design project is very hefty at times. In order to have a successful project development and design, the team needs to have cohesive and strict schedule in terms of determining when specific milestones have to be done and what kind of requirements are needed that would eventually result in a hassle free environment wherein each member of the team is able to work on their own sectors of the project as smooth as possible. If it wasn’t for specific milestones being implemented in our project design, our team will be in complete disarray and chaos due to not knowing what the real task in hand is

Having no milestones will produce confusion amongst the members of the team that will possibly lead to more problems in terms of creating the software or hardware because procrastination has the major effect of affecting a person's well being and ability to progress in the project plus being able to overcome all the unexpected obstacles that may arise at the time of creation of the Minuteman system.

The main point of these milestones are to be responsible for the duties possible and to be aware of the circumstances whether it may be good or bad depending on the situation of the person. The question is: how are project milestones decided upon? These milestones are decided upon critical events that the team might resolve as something that relays good progression and not disorganized information that can potentially disturb the team's growth and rate of progress. The correct use of project milestones is at the start of a critical phase of work and at the end of the critical phase of the work as well. Another time to use project milestones are in the likes of deadlines. Milestones basically act as deadlines in the sense that it is referenced in by the team whenever there is confusion between members. By definition, a deadline is states as the latest time or date that a task should be completed" With this definition, it is imperative that all of these priorities be finished at the specified date. In our team, however, we will try and implement our deadlines that have cushion in between the real deadline date and a personal deadline date for the team as whole. In this way, there will be a leeway in between these two end points just in case there may be changes that are needed to be displaced for better results in the design.

Another question is: how frequently do we really need to put milestones in our project development? The answer is for every planning/ developing cycle it is important that a milestone is laid out before the team so the team understands properly what duties are aligned for them each time. With how humongous this project is, there should be consistent inclusion of these milestones at least 2 a week especially in regards to senior design 2 where the main purpose is to actually create the Minuteman system. It doesn't apply to Senior Design I where it is purely research. There will be mandatory meetings each week to discuss potential changes to the plan or if there are changes that arise that is worth noting for future use. It isn't guaranteed that there are changes that will affect the flow of the development cycles, but there should be critical review for each of the next priorities so that the whole team is unified in what to do. Table 3-3 displays the full listing of our project milestones, as known currently. As shown, some dates are to be determined.

Table 3-3 illustrates the always updated milestone chart that the group initially decided upon when the Senior Design 1 and were just hypothetical decisions on when each component for both semesters were desired to be accomplished.

Milestone	Time Duration	
Task	**Start**	**End**
Research	1/10/2019	2/15/2019
Group Formed	1/10/2019	1/10/2019
Divide and Conquer	1/10/2019	1/30/2019
Task Division	1/10/2019	2/1/2019
Project Meeting	2/7/2019	
Update Divide and Conquer	1/31/2019	2/13/2019
FPGA/Microcotroller	1/21/2019	2/15/2019
Power	2/1/2019	2/15/2019
Communication	2/1/2019	2/15/2019
Sensor	2/1/2019	2/15/2019
Application	2/1/2019	2/15/2019
GPS	2/1/2019	2/15/2019
Software-Hardware Integration	2/1/2019	2/15/2019
First Draft Senior Design	2/15/2019	3/27/2019
Design	4/10/2019	6/28/2019
Microcotroller	4/10/2019	6/28/2019
Power	4/10/2019	6/28/2019
Communication	4/10/2019	6/28/2019
Sensor	4/10/2019	6/28/2019
Application	4/10/2019	6/28/2019
GPS	4/10/2019	6/28/2019
First Design Revise	6/15/2019	6/15/2019
Second Design Revise	6/26/2019	6/28/2019
Final Design Revise	7/7/2019	7/8/2019
Verification/Implementation	6/15/2019	7/31/2019
Microcotroller	6/15/2019	7/10/2019
Power	6/15/2019	6/26/2019
Communication	6/15/2019	7/29/2019
Sensor	6/15/2019	7/8/2019
Application	6/15/2019	7/31/2019
GPS	6/15/2019	7/29/2019
First Test Revision	6/15/2019	6/25/2019
Second/Final Test Revision	6/26/2019	7/7/2019
Final Test Revision	7/7/2019	7/31/2019
Final Project Presentation	8/2/2019	

Table 3-3: The Project Milestone Listing

3.5 Decision Matrix:

The project was decided through meetings that took place after class and at certain meeting times. After the first three meetings, we created a priority project list shown below. We weighted our individual desires and what resources the project would cost. We ranked what we wanted with one being the best and 5 being the worst. The score is added up in the Decision Matrix below in the Group Ranked tab. After this, each project was analyzed with how much time it would consume. The cost of the project and capacity of knowledge that we would want to expand.

Table 3-4 shows the decision matrix of the group, including the ranking each member of the group gave to each of the project ideas presented between us.

Project Idea	Nathan	Nathaniel	Daniel	Eric	Added Lowest Score	Group Ranked
Master/Slave RC Car	3	4	2	4	13	3
Warning Bracelet	2	1	3	2	8	2
Gunshot Detection	1	2	1	1	5	1
Laser Guided Drone	4	3	4	3	14	4
Mini Surveillance	5	5	5	5	20	5

Time	Cost	Knowledge	Resource Cost	Total	Total Rank
3	3	4	10	23	3
1	2	3	6	14	2
2	1	2	5	10	1
5	4	1	10	24	4
4	5	5	14	34	5

Table 3-4: The Decision Matrix

The ranking show with the Group Ranked and the Total that the Gunshot Detection project was the best option for our group. This project was taken well by all members in our group with the desire to help each other in time of need. This project will not only cost the least, but it will also help expand the knowledge of the group.

3.6 Methodology for Design:

Developing the necessary code for the Minuteman project, our crew was focused on the usage of agile software development paradigm. This kind of methodology adapts unique way of planning how to start developing code in different stages. As our device is being created, it was expected that there will be new constraints or requirements that will arise as the deadline gets closer and closer or when new information is gathered. There were times when the team started to build the system but would encounter some roadblocks along the way whether it may be a piece that needs to be replaced because of unforeseen circumstances that will pop up and we find out that a part is not compatible with the system as a whole or it is not robust enough when the software application is running. With agile movement, it was a very perseverant movement that forces users to readjust with their developmental tactics or plan when obstacles appear along the way of development. Continuous improvement and early delivery were key in making this system work without any dead ends so that each developer knew the main plan of attack.

This is compared to the traditional linear system of planning, designing, implementing, careful testing and eventual release. With an agile system, it will follow a regular cyclic schedule where testing passes through the planning phase and back again cycling every single stage until it has a definite signal of a thumbs up for a release. What this kind of scheduling did is that it constantly accompanied any new changes or ideas to the system whether it's a concrete change or a minor change that doesn't need much thought put into it.

The kind of framework that the team used was Scrum. Using the Scrum process, the development of the Minuteman was divided into various levels of sprints throughout all methods of project development, that varied within the amount of difficulty that a task desires or needs. A scrum framework is led by scrum master to monitor each stage of development and decide on whether progress has been made and appeal any modifications that may be needed for the system from everyone in the team. Since our whole team has experience in software development, half of us targeted microcontroller programming and the other half handled the task of programming the UI and the Arduino boards.

4. Research and Theory

4.1 Existing Projects

There are a number of pre-existing projects to Minuteman that had similar goals. They are typically gunshot detectors designed for large cityscapes or smaller facilities such as school campuses. These projects include a previous Senior Design project, several products for the private marketplace, and several products for military/government use. These are typically made with various more specific applications in mind, but they generally follow the same idea of using audio sensors to detect gunfire and respond accordingly.

4.1.1 Project GLASS

Project GLASS was a senior design project created in 2014. GLASS was created with similar circumstances in mind to Minuteman, attempting to breach the political divide created by mass shootings in the American public by creating a technological solution to the issue. Although GLASS is more focused on outdoor applications, such as in cities, it is relatively similar in that it detects gunshots, determines the type of firearm used, and locates where the gunshot originated, and then outputs this data to an application that can then inform authorities.

GLASS determines a gunshot from an event by looking for specific conditions to confirm that a gun was fired; a decibel level exceeding 140 decibels, peak frequency in the event indicating the gunpowder discharge in the barrel, and the subsonic frequency made by the projectile flying through the air. The former filters out most events not involving a firearm (although GLASS admits that suppressed weapons could have a lower decibel level), the second condition allows GLASS to also determine the caliber of the projectile, and the latter allows GLASS to confirm if it was definitely a firearm and not something similar like a recording.

GLASS ultimately didn't produce their theoretical product, and instead had to create a prototype to demonstrate it's abilities using the limited resources their team had available. The theoretical design involved four microphones with individual processors to filter out events that don't meet the first condition. Once an event does, it passes it on to a processing unit to analyze the event and filter it through the two remaining conditions. Then it passes this information on to an android application. Their prototype ended up using a simpler audio sensor that instead communicated with a normal computer, but demonstrated what the project attempted to do. [3]

4.1.2 ShotSpotter

ShotSpotter is a company that specializes in providing outdoor gunshot detector products to law enforcement, as well as public and private facilities. Although they have several products currently, they are all based off of their ShotSpotter system. Over a large area, such as a neighborhood or campus, the system detects gunshots and timestamps when they detected it. Knowing the speed of sound, and the direction of the sound, the system is able to use these timestamps to determine the distance, and pinpoint where the gunshot occurred [4].

ShotSpotter offers four products utilizing their technology, although we will focus on two of these products since they are more relevant to Minuteman.

4.1.2.1 Site Secure

Site Secure is a general product provided for facilities, whether they be businesses, hospitals or school campuses. It's unclear how it would differentiate from SecureCampus on the latter. The detector is able to determine the number of shooters and shots fired, whether the weapon is automatic, and the direction of travel. It can accurately pinpoint the location of a gunshot to within 25 meters Then it sends this information to authorities and on-site personnel through alerts. It can also be integrated with other on-site security systems. It has a successful detection rate of about 90%, and can send out alerts within a minute. Audio samples are verified by a ShotSpotter Incident Verification Center [4].

SiteSecure is also advertised to energy infrastructure sites, such as power stations or transformer sites. This describes them as fence or pole mounted systems. It also denotes that ShotSpotter can detect gunshots from up to 500 meters away. It also claims that it is able to detect subsonic projectiles, and can locate them even if the gunshot is not toward the direction of the device [4].

4.1.3 Raytheon Boomerang III

Boomerang is a military shot detection system designed to detect ballistic projectiles heading toward it, determine the direction they came from, then deliver this information to its users. It filters out non-ballistic sounds, as well as gunfire going out away from the device. It can be mounted onto protected buildings or facilities, or on vehicles such as a HUMVEE.

The system works through a multi-directional microphone array, as shown in Figure 4-1. It passively detects acoustic events and filters them through signal processing. Boomerang audibly calls out that a shot was detected, and states the rough clockwise direction and distance, while presenting more accurate data to an LED display. It is also able to broadcast the data to an outside facility or center through an Ethernet connection.

The system is able to operate on vehicles moving up to 60 miles-per-hour in any outdoor terrain. It has a detection rate of greater than 95%, and is able to pinpoint the direction to within 2.5 degrees and distance with about a 10% margin of error. It requires no calibration, just activation. It can be easily incorporated with a separate system or third-party systems through an integration kit and interface. [6]

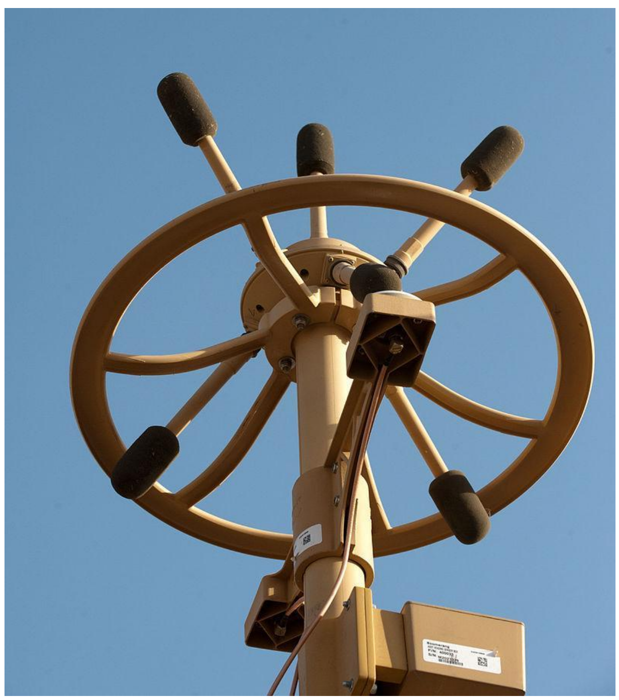

Figure 4-1: Boomerang III microphone cluster. Open Government License. Obtained from wikipedia.

4.1.3.1 Raytheon Boomerang Warrior-X

The Boomerang Warrior-X is a derivative product of the Boomerang III system. It is designed to be worn over the shoulder of soldiers in active combat areas. It features a separate display and like the Boomerang III, it detects incoming gunfire and returns where it is coming from. It is then able to relay this information through audio played aloud or through an earpiece.

Like the Boomerang III it has a detection rate of greater than 95%, although the margin of error is greater for directions, at less than 7.5 degrees. The margin of error for range detection is also greater at 20% rather than 10%. No information is available about it's power supply but it is able to operate for about 12 hours. [7]

4.1.4 Guardian Indoor Active Shooter Detection System

The Guardian Indoor Active Shooter Detection System is created by a company called Shooter Detection Systems. As the name describes, the system is designed to detect indoor shootings within a building or facility. It detects a gunshot by both picking up the infrared light emitted by the muzzle flash of the weapon and the audio generated by the gunshot. It is able to use both to locate the shooter within a provided floorplan of the building. Upon detection, it is able to alert security through its own program or application, and is able to integrate with third-party systems.

The devices themselves take the appearance of a normal wall-mount, similar to a power outlet. Due to it using both infrared and audio detection methods, it is able to confirm gunshots with a high degree of accuracy. Although no statistics were provided, it is stated that the system has a range of a 40m radius from the mount. It draws about 1 watt of power through a power over ethernet relay, which connects to a computer containing the software needed to run the system. [8] [9]

4.1.5 SenseShot

SenseShot is a series of products provided by Information Systems Technologies Incorporated that offers both indoor and outdoor gunshot detection systems. Specifically, they are designed to be low cost, and low power solutions. For their indoor application, it consists of a network of MEMS microphone sensors similar in appearance to a fire-alarm system. Much like Guardian it utilizes a Power-over-Ethernet connection to a relay that connects to a base station that is hosting the software for the system. At this system, the software filters out real events from false alarms through analysis, like GLASS. It is also able to estimate the shooter's location and the trajectory of the gunshots detected.

The SenseShot indoor detection system can pick up low-powered firearms to an estimated 100 meters, while being able to pick up high-powered firearms at 200 meters. It is recommended they are spaced out every 200 meters, and the network itself can handle up to 252 sensors. It can be set up to take 120 volt A/C power, and has a wireless Wi-Fi option.

Their outdoor detection system operates through a network of audio sensors mounted onto municipal street lights. This network is managed through a server operating their SensorAgent software. Multiple sensors are used to both confirm and triangulate the shooting, and report it to authorities, even directly to on-duty officers through dispatch. The system requires at least 2 arrays. It is able to detect a low-powered weapon within about 500 meters, and a high-powered weapon within 1km of the sensor. It can utilize either ethernet, fiber optic, or wireless signals for each sensor. [10]

4.1.6 Observations

In conclusion, we have decided that it might be cheaper to integrate an outdoor audio system. While an indoor system would be greatly useful, due to reverberation, relying on audio confirmation can make the results inaccurate. While incorporating a secondary detection system, such as infrared, can make the results far more accurate, the implications is that the range and visibility of the sensors requires that more sensors are provided. Since our goal is to provide an affordable platform, more sensors would increase the costs of our system.

The Boomerang III and Warrior-X provide interesting examples of just how accurate an audio detection system can be in the outdoors, even while on a moving platform such as a vehicle or individual. It is likely that our system will be similar, but will use triangulation between multiple sensors to more accurately locate the shooter.

ShotSpotter offers an interesting take on facility-based shot detection systems, even if technical specifications aren't all that available. However, SenseShot and GLASS are able to provide more technical angles on such things, especially with ideas such as Power-over-Ethernet connections, multiple condition filters, such as decibel level and muzzle-blast shockwaves. These are likely to influence the design of Minuteman, in that it will be an outdoor gunshot audio detection system. Likely for facilities, rather than large scale environments.

4.2 Gunshots

Detecting acoustic events relating to gunfire requires understanding the nature of these events. When a gun is fired, it is propelling a led projectile (sometimes with other materials, such as copper) out through a barrel, by means of gaseous expansion of ignited gunpowder. Upon leaving the barrel, the gun generates a muzzle blast as the gases escape after the bullet. The bullet itself, due to the velocities it is propelled at, usually creates a supersonic shockwave behind it as well. [11] The only exceptions to these outcomes is whether the firearm is equipped with a suppressor and if the ammunition itself is designed to propel the bullet at subsonic speeds.

Excluding those exceptions, and despite other varying characteristic which will be explored, a gunshot will always generate a sound wave above 130 dB within a certain vicinity of the gunshot (depending on the caliber and intensity of the muzzle blast). The muzzle blast will generate a spherical sound wave from the location of the blast, whereas the shockwave of the bullet will follow it in a conical pattern, following the doppler effect, expanding from the trajectory of the bullet as it soars through the atmosphere. [11]

The rest can vary depending on the firearm and ammunition. Factors such as gas system, barrel length and rifling, to projectile shape and consistency of the primer and gunpowder can alter these characteristics in varying ways. This is why, although a gunshot can be fairly distinct from other sound events, it is important to be able to roughly make a distinction between the characteristics of these events for the purposes of confirmation.

When the shockwave is picked up by a sensor, it will appear as a thin “N” shaped waveform. This is the result of the shockwave compressing air particles at a high-velocity (represented by the upper amplitude) with a trailing rarefaction wave of low-velocity particles (represented by the lower amplitude). This is distinct to ballistic projectiles. This can be used to confirm that a gun is being fired by determining if the sensor detects a shockwave following the projectile. [12]

The amplitude of the N-wave is weakly dependent on the Mach velocity of the projectile [12]. Thus, so long as the bullet is supersonic, it will emit a very clear shockwave that can be detected by a sensor. However, the amount of time it takes to receive the shockwave after the muzzle-blast depends on the position of the sensor, due to how the shockwave follows the bullet in a conical pattern, rather than just releasing in a circular pattern from the muzzle-blast. Thus any method to detect the shockwave will first need to respond to the muzzle-blast, and need to be able to pick up the shockwave within a set amount of time.

Figure 4-2 displays a two-channel recording of a gunshot as an example. A noticeable complication to any software is the addition of reflection. This when the sensor receives a similar signal from reflections of the original signal off of other surfaces, such as the ground. This can seriously disrupt efforts to locate the signal.

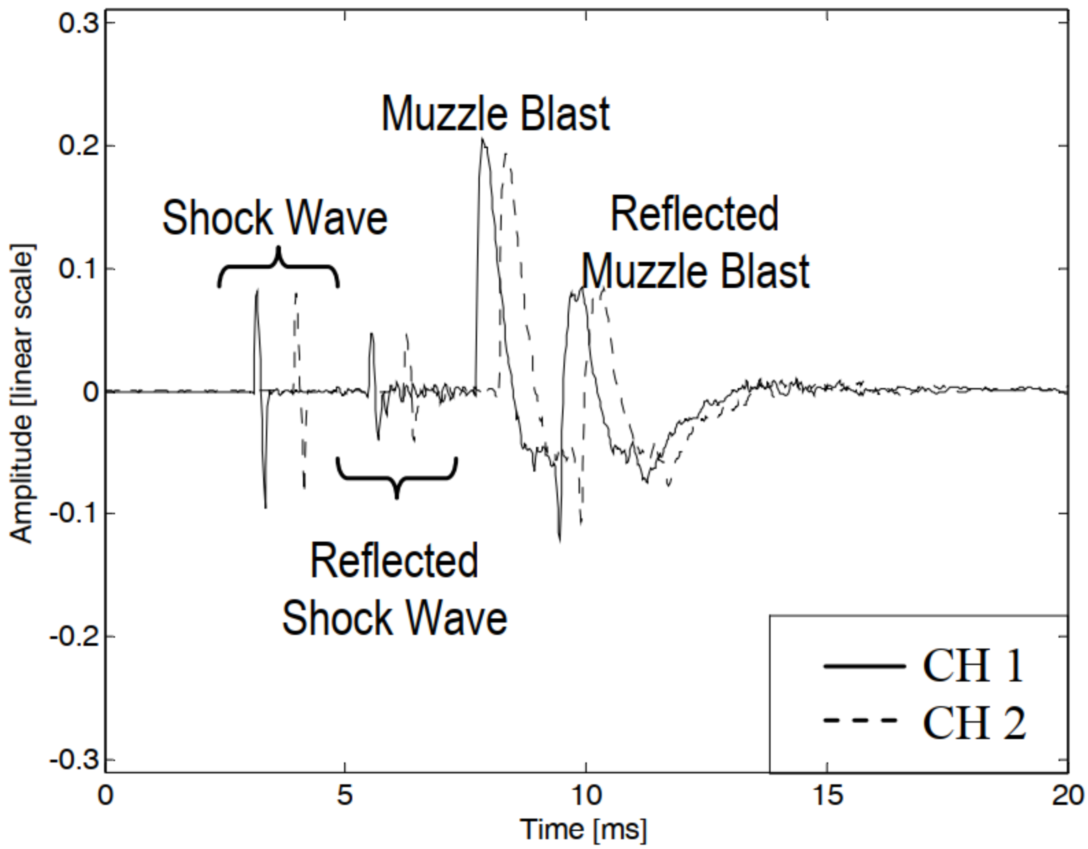

Figure 4-2: An example of a gunshot recording with two channels [38].

4.3 Triangulation

For our project, triangulation was one of the options considered for determining where the sound of a discharged weapon is coming from. Triangulation uses trigonometry in order to determine the position of an object by using angles to get the direction where it's coming from.

2D Triangulation - For using 2D triangulation, two arrays of three microphones would be needed, which would be shaped in the form of an equilateral triangle. Using two arrays, the angle obtained from each array can be used to find a point of intersection, which is where the source where the sound is coming from.

For this approach to work, the microphones would need to be close together, and the sound be far enough so it can be considered a line that is perpendicular to the origin. Sound must be picked up far away anyways, as the microphones that will be used have a decibel peak of 130 dB for the standard pressure level. If a firearm is discharged near the array, this would lead to distortion as most firearms have a decibel peak of around 140 dB to 150 dB, so the sound picked up from the microphone would not be able to be compared to other firearm sounds due to this distortion.

When the array of microphones picks up sound from a discharged firearm, the time difference between the first microphone and the second microphone to pick up the sound can be used. This is represented as (t_B - t_A), where t_A is the time of the first microphone, and t_B is the time of the second microphone. Knowing the speed of sound, which is represented by C, the distance can be calculated by multiplying the speed of sound by the time difference, which is represented as Δx in the following equation:

$$\Delta x_{BA} = C \times (t_B - t_A)$$

The equation for speed of sound used in our project is the ideal speed of sound equation related to temperature in a dry environment. This equation was used, as the pressure and humidity did not have much of a significant impact of the speed of the sound. A website [16] with a handy calculator was used to perform the calculation of the speed of sound using data from a local weather report, where the temperature (T) was equal to 28℃, pressure (p) was equal to 101.8 kPa, and relative humidity (φ) was equal to 36%, which resulted in a speed of sound of 348.74 m/s. The calculation for the ideal speed of sound with the temperature (T) equal to 28℃ is 348.1 m/s, which uses the equation shown down below. Using this comparison, only temperature is necessary to be measured for our project, so no barometer or humidity sensor is needed for our project, which helps save cost on components.

$$C = 331.3\sqrt{1 + \frac{T}{273.15\ K}}$$

Knowing the calculated distance Δx and each side of the equilateral triangle is the same value, which can be represented as S, trigonometry can be used to calculate the angle from these values. Since the sound is considered perpendicular, a right triangle is formed on the side of the equilateral triangle, where the side of the equilateral triangle is considered the hypotenuse of the right triangle, and the distance Δx is considered the adjacent side of the right triangle, so the cosine function of the adjacent over the hypotenuse is used, which is then inverted to solve for the angle, which is represented as θ. After θ is calculated, the angle α can be calculated by adding the angle θ to the 60° angle of the equilateral triangle next to it. Figure 4-3 shows a visual representation of this.

$$\theta = \text{Cos}^{-1}(\Delta x \div S) \qquad \alpha = \theta + 60°$$

Even if solving for the angle α is more than enough to get the angle the sound source is coming from, error in the degree of the angle calculated can still occur. This amount of error can be determined by using two equations. The first equation is solved by getting the time difference of the two microphones that picked up the sound after the first microphone, which for this equation, can be assumed to be $(t_B - t_C)$. The difference is then multiplied by the speed of sound, which is represented by C. This distance can be represented as Δy_{BC}, which is the side of another right triangle that is formed. Knowing the angle α is the addition of θ and the equilateral triangle angle of 60°, a straight line can be formed if another angle is added so it is 180°. This can be used for the second equation of Δy, which uses the trigonometric function where the cosine of (180° - α) is set equal to the adjacent side of the triangle (Δy) divided by the hypotenuse of the side of the equilateral triangle (S). The equation is then rearranged in which the value for Δy is solved for. If both equations of Δy are not equal to each other, then that means there is an error in the angle calculated. The equations are shown below. Figure 4-3 also shows the visual representation of sound moving directly to the front microphone A and sound coming from the right side of the microphone array.

$$\Delta y_{BC} = S \times \cos(180° - \alpha) \qquad \Delta y_{BC} = C \times (t_B - t_C)$$

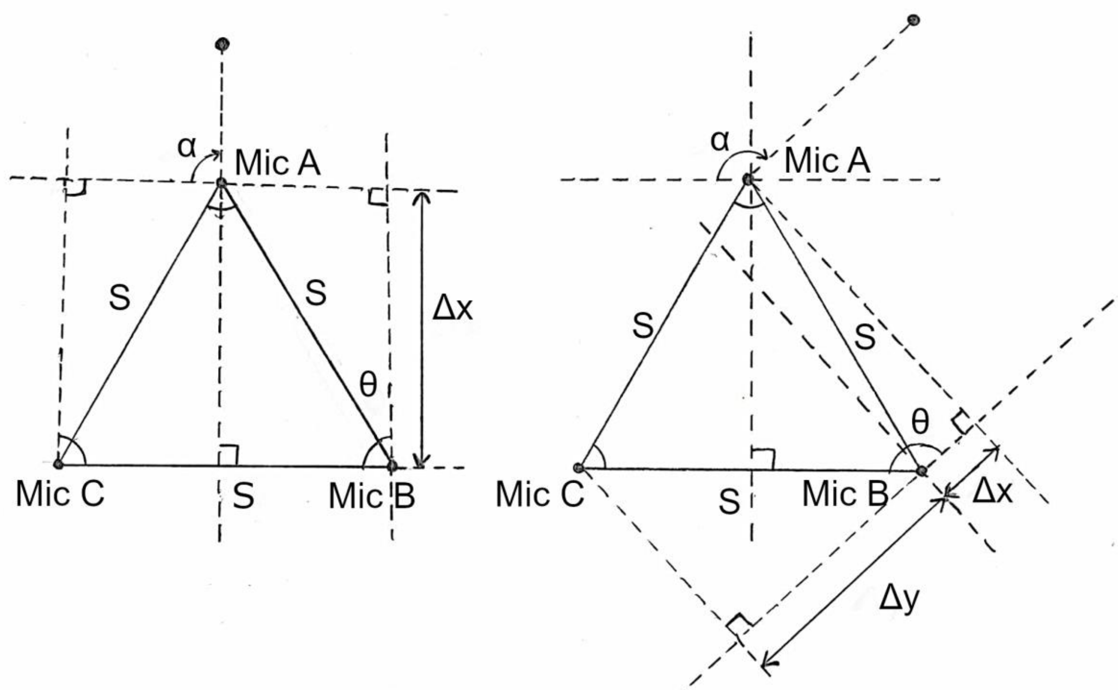

Figure 4-3: Shows how the direction of sound is located from the mic array

Since the angle of the direction of the sound can be located, the distance can be determined to the source if a secondary microphone array picks up the sound of the discharged firearm. Since the arrays would be a distance apart, the length between the microphones of both arrays that picked up the sound first would be known. This would be done by using a GPS module that would give the coordinates of where the array is. Since a GPS module is used, a compass would also be needed to get the orientation of the microphone array, so the values of θ and α can be adjusted accordingly. With these adjustments, the position of the sound source should be able to be obtained. The direction angles of α_1 and α_2 of the two microphone arrays are greater than 90°, so two new angles of β_1 and β_2 would be calculated by having the angle α subtracted from 180°. Given both of these are less than 180°, the angle of β_3 can be calculated by having $(\beta_1 + \beta_2)$ subtracted from 180°. Since the distance between both microphone arrays are known, which can be represented as D_3, and all angles of the triangle formed between both microphone arrays are known, the law of sines can be used to solve the distance of one of the microphone arrays from the source. The sine of β_3 divided by the distance D_3 can be set equal to either the sine of β_1 divided by the distance D_1 and solve for D_1 or the sine of β_2 divided by the distance D_2 and solve for D_2. The equations are shown below and Figure 4-4 represents the 2D triangulation.

$$\beta_1 = 180° - \alpha_1 \qquad \beta_2 = 180° - \alpha_2 \qquad \beta_3 = 180° - (\beta_1 + \beta_2)$$

$$D_1 = [\mathrm{Sin}(\beta_2) \times D_3] \div \mathrm{Sin}(\beta_3) \qquad D_2 = [\mathrm{Sin}(\beta_1) \times D_3] \div \mathrm{Sin}(\beta_3)$$

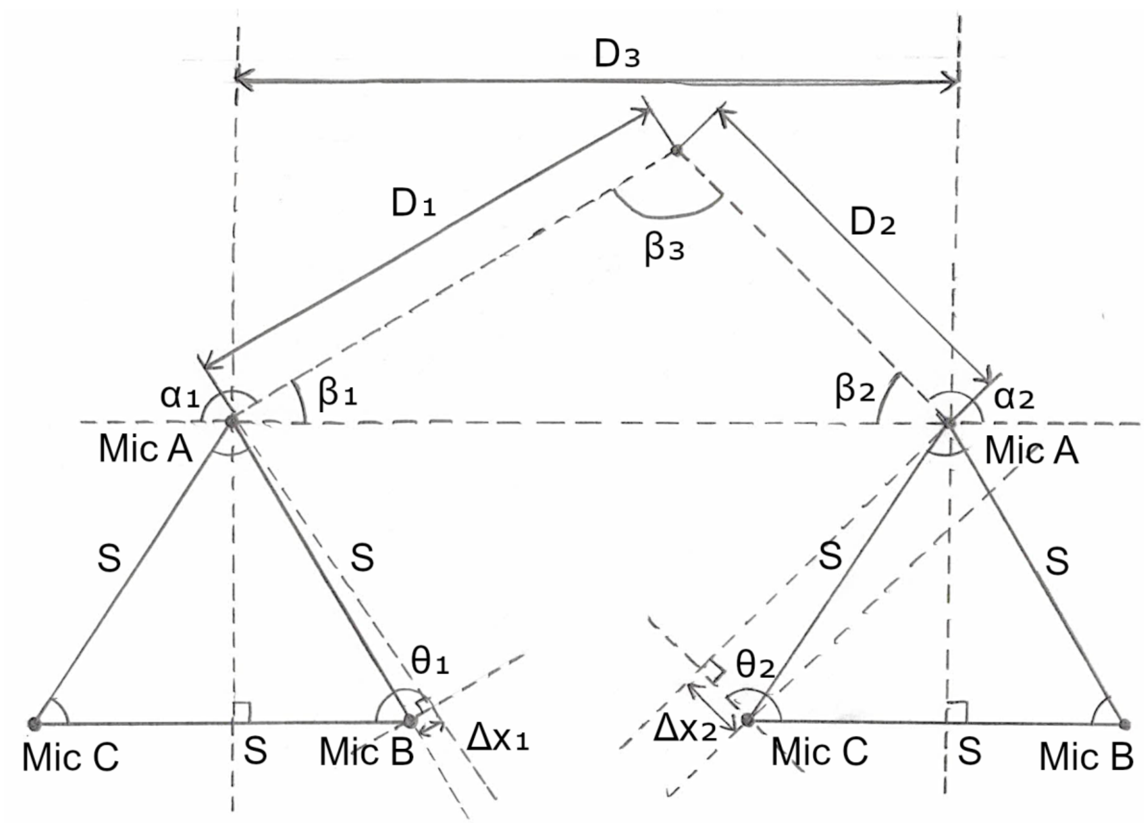

Figure 4-4: Diagram showing two microphone arrays locating sound source

As the distance can now be calculated using the law of sines, the position of the sound source can be located, but the distance needs to be split into x and y components. To get the x component of the distance D_1, the cosine of β_2 is multiplied by D_1, and to get the y component, the sine of β_2 is multiplied by D_1. The same procedure is used to split the components of D_2, but D_{x2} will be negative as it is moving to the left as shown in Figure 4-4. The equations to split the components are shown below:

$$D_{x1} = \mathrm{Cos}(\beta_1) \times D_1 \qquad D_{y1} = \mathrm{Sin}(\beta_1) \times D_1$$
$$D_{x2} = -\mathrm{Cos}(\beta_2) \times D_2 \qquad D_{y2} = \mathrm{Sin}(\beta_2) \times D_2$$

3D Triangulation - GPS utilitizes latitude and longitude as it's coordinate system, so the microphone array in this project would need to be 3D, so an additional microphone would need to be added. This would then form an equilateral triangular pyramid for its shape, as shown in Figure 4-5 for the side view and top view. The fourth microphone in the array would use the same calculations as done for the 2D coordinates. When observing the figures below, it is found that the coordinates can become spherical, where α_2 in the Figure 4-5 side view can be represented as θ for spherical coordinates if α_2 is subtracted from 90°, as θ starts from the z-axis and moves clockwise to a maximum of 180°.

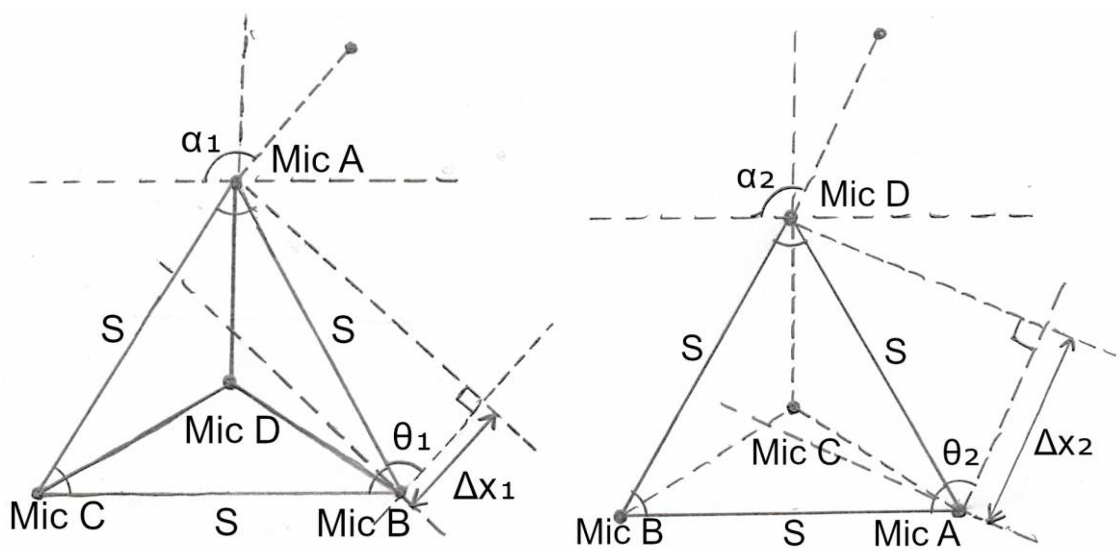

Figure 4-5: Diagram of the top (left) and side (right) view of the 3D microphone array

In the Figure 4-5 top view, $\alpha 1$ can be considered φ, but would need the have the value of $\alpha 1$ subtracted from 270°, as φ starts from the x-axis and rotates to a maximum of 360° counterclockwise. The radius of spherical coordinates would then be the distance of the source of the sound to the first microphone that picked up the sound, which is the length of D. Knowing this, the coordinates calculated should be able to be easily used with the GPS coordinates of the microphone array.

With triangulation, having the requirements of two or more microphone arrays can be problematic, as if only two microphone arrays are used, and there is a scenario in which one breaks down, then this can lead to the failure of detecting a discharged firearm's location, but the usage of using one microphone array can still be used to report it to authorities. When multiple firearms of the same type are being discharged, each microphone array could potentially get sound from two different firearms. Having this occur, the location of the sound source calculated would be wrong, as both sounds would be from two different locations, compared to one location using both arrays from two different sounds.

4.4 Multilateration

Multilateration is another known method that can be used to locate a transmission of a source. A real-life application of this is known as WAM (wide area multilateration). For WAM to work, an aircraft would send a transmitted signal, which is received from several antennas at known positions from each other. Each antenna would be synchronized at the same clock and would receive the transmitted signal from the aircraft, which would be at a different time from each other. The antennas would then compare their TDOA (time distance of arrival) and would use this to form hyperboloids which are then intersected with each other. When the hyperboloids intersect, they would give the position of the aircraft. [17]

2D Multilateration - For WAM, the transmitted signal would have the speed approximately of the speed of light, which is approximately 3×10^8 m/s. This method can also be applied for sound as well, if the speed of sound is used instead of speed constant of a signal sent from an emitter. Knowing this, each microphone can be used instead of antennas and the same principle of the intersection of hyperboloids can be used to determine the location of a discharged firearm for our project. Figure 4-6 shows a visual representation of how hyperboloids are used to determine the location of a source. The left side of the figure shows how the Loran-C hyperbolic navigation works, in which a flat line of a hyperbola is determines the possible location of the source of the signal. The right side of the figure shows how intersection of hyperbolas between multiple receivers can be used to determine the location of a source.

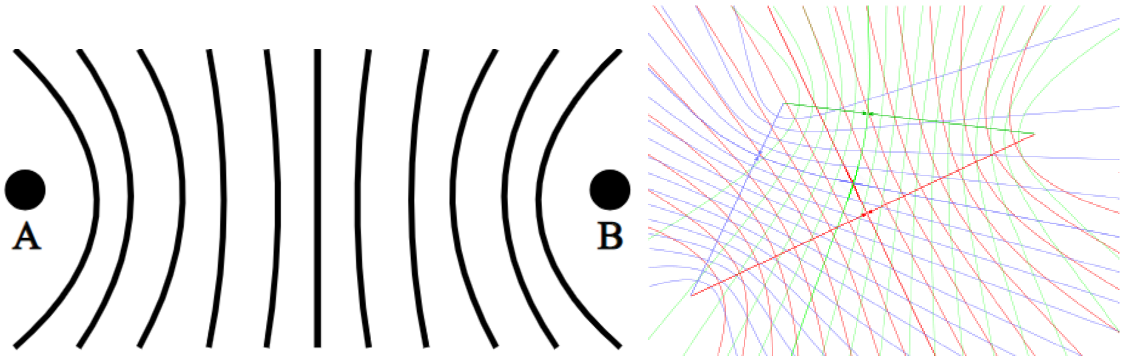

Figure 4-6: (left) Loran-C hyperbolic navigation [18], (right) 2D Multilateration [19]

Multilateration would still require the same components as used for the triangulation method. The GPS module would provide the origin of the microphone array, so the coordinates of each microphone relative to the GPS module are known. The compass would also provide the orientation of the array, so adjustments can be made when the location of the source is obtained.

2D Multilateration would use the same three microphones used for 2D triangulation, with the exception that it does not need to be in the form of an equilateral triangle. Knowing this, the distance equation can be utilized for multilateration, which uses the Pythagorean theorem. Using the distance equation, the axis values of x and y are both the power of 2, which are both added, and set equal to the distance of D which is also the power of 2. For a microphone, represented by 'i' that receives the sound, the values of x and y are replaced by $(x - x_i)$ and $(y - y_i)$, where x_i is the distance of microphone 'i' from the origin of the x axis, and y_i is the distance of microphone 'i' from the origin of the y axis. The value of D is also replaced with D_i for the microphone that receives the sound input.

$$(x - x_i)^2 + (y - y_i)^2 = D_i^2$$

The microphone array would pick up the sound of the discharged firearm the same way as done using triangulation. The value of D_i is calculated using the distance formula, in which the speed of sound C is multiplied by the time of t_i, when it detected the sound of the discharged firearm. The microphone array would also use the temperature probe that would be used for the triangulation method, which would then be plugged into the equation for the speed of sound, which can then be applied to the distance equation.

$$D_i = C \times t_i \qquad\qquad C = 331.3\sqrt{1 + \frac{T}{273.15\ K}}$$

Knowing the value for D_i can be calculated, the distance equation using the Pythagorean theorem can be squared for the values of $(x - x_i)^2$ added to $(y - y_i)^2$

and D_i^2. The value of D_i will also be replaced by the speed of sound C multiplied by t_i.

$$\sqrt{(x-x_i)^2+(y-y_i)^2}=D_i \quad \rightarrow \quad \sqrt{(x-x_i)^2+(y-y_i)^2}=C\ x\ t_i$$

Knowing there are three microphones for the microphone array in 2D, then there are three equations that can be written, which would be represented as A, B, and C.

$$\sqrt{(x-x_A)^2+(y-y_A)^2}=C\ x\ t_A$$
$$\sqrt{(x-x_B)^2+(y-y_B)^2}=C\ x\ t_B$$
$$\sqrt{(x-x_C)^2+(y-y_C)^2}=C\ x\ t_C$$

In the current form, all equations have the values for x and y being unsolvable. However, by using the difference of time from the first microphone that picks up the sound from the discharged firearm and the other two microphones, two equations can be written. Knowing there are two unknowns, and two equations, the values for x and y can be solved for.

$$\sqrt{(x-x_B)^2+(y-y_B)^2}-\sqrt{(x-x_A)^2+(y-y_A)^2}=C\ x\ (t_B-t_A)$$
$$\sqrt{(x-x_C)^2+(y-y_C)^2}-\sqrt{(x-x_A)^2+(y-y_A)^2}=C\ x\ (t_C-t_A)$$

This equation can also be simplified further by having the origin changed as the first microphone that receives the sound from the discharged firearm. The relative distance values of the other microphones, represented as 'i', has x_i and y_i changed to x'_i and y'_i, which are set equal to the value of (x_i - x_0) and (y_i - y_0), where x_0 and y_0 are the relative position of the microphone that first received the sound.

$$\sqrt{(x-x'_B)^2+(y-y'_B)^2}-\sqrt{x^2+y^2}=C\ x\ (t_B-t_A)$$
$$\sqrt{(x-x'_C)^2+(y-y'_C)^2}-\sqrt{x^2+y^2}=C\ x\ (t_C-t_A)$$

3D Multilateration - Knowing three hyperbolas are utilized by 2D multilateration to solve the x-axis and y-axis location of the discharged firearm, then three hyperbolas would be utilized for 3D multilateration, as shown in Figure 4-7. This would add a fourth microphone similarly to how 3D triangulation is performed but differs in which the fourth microphone can be located anywhere on the z-axis instead of needing to form the shape of a pyramidal triangle. Using the same four equations, the time difference between the first microphone that received the sound and the other three microphones can be used to solve for the unknown values of x, y, and z.

$$\sqrt{(x-x'_B)^2+(y-y'_B)^2+(z-z'_B)^2}-\sqrt{x^2+y^2+z^2}=C\ x\ (t_B-t_A)$$

$$\sqrt{(x - x'_C)^2 + (y - y'_C)^2 + (z - z'_C)^2} - \sqrt{x^2 + y^2 + z^2} = C\ x\ (t_C - t_A)$$
$$\sqrt{(x - x'_D)^2 + (y - y'_D)^2 + (z - z'_D)^2} - \sqrt{x^2 + y^2 + z^2} = C\ x\ (t_D - t_A)$$

Knowing these equations are nonlinear, they are considered too computationally expensive to solve if a cheap microcontroller is used for our project. However, given there are three values of x, y, and z, it is possible to linearize this equation and remove any square roots. This linearized equation can then be used with a 3x3 matrix, and the method of gaussian elimination will solve the values of x, y, and z.

The equation would be linearized by first using a value to represent the square root of the distance equation, where R_i can represent the square root of $(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2$. Using the TDOA between the microphone of 'i' and the first microphone '0' to pick up the sound, the square of R_i to the power of 2 can be set equal to $[(D_i - D_0) + R_0]^2$. The equation can then be further simplified and be set equal to 0, as shown below.

$$[(D_i - D_0) + R_0]^2 = R^2_i \qquad \rightarrow \qquad (D_i - D_0) + 2R_0 + [(R_o^2 - R_i^2) \div (D_i - D_0)] = 0$$

Since $2R_0$ is the square root of the problem that needs to be eliminated, then another equation can be set equal to the simplified equation above. This can be done using any of the other microphones in the array that is not the first microphone. The microphone of 1 will be used to be set equal to this, which will then remove the value of $2R_0$.

$$(D_i - D_0) + 2R_0 + [(R_o^2 - R_i^2) \div (D_i - D_0)] = (D_1 - D_0) + 2R_0 + [(R_o^2 - R_1^2) \div (D_1 - D_0)]$$
$$(D_i - D_0) - (D_1 - D_0) + [(R_o^2 - R_i^2) \div (D_i - D_0)] + [(R_o^2 - R_1^2) \div (D_1 - D_0)] = 0$$

After the square root is removed, $R^2{}_i$, $R^2{}_1$, and $R^2{}_0$ can have their x, y, and z values foiled. These would then be substituted to replace $(R_o^2 - R_i^2)$ and $(R_o^2 - R_1^2)$.

$$-x_i^2 - y_i^2 - z_i^2 + 2[(x)(x_i) + (y)(y_i) + (z)(z_i)] = (R_o^2 - R_i^2)$$
$$-x_1^2 - y_1^2 - z_1^2 + 2[(x)(x_1) + (y)(y_1) + (z)(z_1)] = (R_o^2 - R_1^2)$$

After the equations above are substituted, the only values of x, y, and z are unknown, while the rest are constant values. Knowing this, the equation can be written in the format of Ax + By + Cz + D = 0, where each constant of A to D can be calculated as shown below. The constant of D is not to be confused with the distance of D_i. D_i is substituted with the distance equation as shown below.

$$A_i = \frac{2}{C}\left(\frac{x_i}{t_i - t_0} - \frac{x_1}{t_1 - t_0}\right),\ \ B_i = \frac{2}{C}\left(\frac{y_i}{t_i - t_0} - \frac{y_1}{t_1 - t_0}\right),\ \ C_i = \frac{2}{C}\left(\frac{z_i}{t_i - t_0} - \frac{z_1}{t_1 - t_0}\right),$$

$D_i = C(t_i - t_1) + \frac{1}{C}\left(\frac{-x_i^2 - y_i^2 - z_i^2}{t_i - t_0} + \frac{x_1^2 + y_1^2 + z_1^2}{t_1 - t_0}\right)$, $A_i x + B_i y + C_i z + D_i = 0$

Since two microphones are already used in these equations, then in order to solve for three unknowns, a fifth microphone would be needed to be added to the microphone array. With the fifth microphone added, three equations can be solved. The values of x, y, and z can then be solved using gaussian elimination, using the equation of Ax = B. B is a 3 x 1 matrix that contains the constants of D_2 to D_4. A is a 3 x 3 matrix where the first row is A_2, B_2, C_2, second row is A_3, B_3, C_3, and the third row is A_4, B_4, C_4. The 3 x 1 matrix of 'x' contains the values to solve for of x, y, and z. In order to solve for x, y, and z, the matrix of A is inverted, which is then multiplied by the matrix of B, which then solves the values of x, y, and z. Given these, the distance of a discharged firearm with respect to the microphone array is known, and the GPS and compass can be used to get the correct position of the firearm used.

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} A_2 & B_2 & C_2 \\ A_3 & B_3 & C_3 \\ A_4 & B_4 & C_4 \end{bmatrix}^{-1} \begin{bmatrix} D_2 \\ D_3 \\ D_4 \end{bmatrix}$$

Multilateration, compared to triangulation, offers advantages that make it easier to utilize for our project. This includes reducing the amount of materials needed for this project, as only one array of microphones would be needed instead of using two microphone arrays. This can also allow it to become mobile, as it would be possible to mount it on a vehicle given only one microphone array is needed. However, if the microphone array is mobile, this can affect the results of the location of the sound, as the vehicle moving can cause differences in the time of arrival between microphones that receive the sound, which would reduce the accuracy of finding the location of a discharged firearm.

Sample rate is also a design concern, as having a lower sample rate would reduce the time difference between microphones, which will affect the accuracy of the location of a discharged firearm. This can be alleviated by spacing microphones further apart, but this can eventually lead to issues with the stability of the structure. Having microphones that are spaced further away can also eventually lead to the point where it would not be able to be easily transported by hand, and would also go over the weight restriction of 15 to 10 lbs.

If the microphone array is designed to be mobile, the compass utilized would require an accelerometer and gyroscope added with the magnetometer. A new compass with a magnetometer, accelerometer, and a gyroscope together would be expensive. However, if only one microphone array is needed, then only one GPS module and compass module is needed, which from this offset, would be around the same cost using just one GPS module and a more advanced compass. Factoring the cost of the structure for the microphone array, using only one would save cost as well. From these observations, utilizing multilateration is more beneficial than triangulation, and will allow more features for how the sensor module can operate.

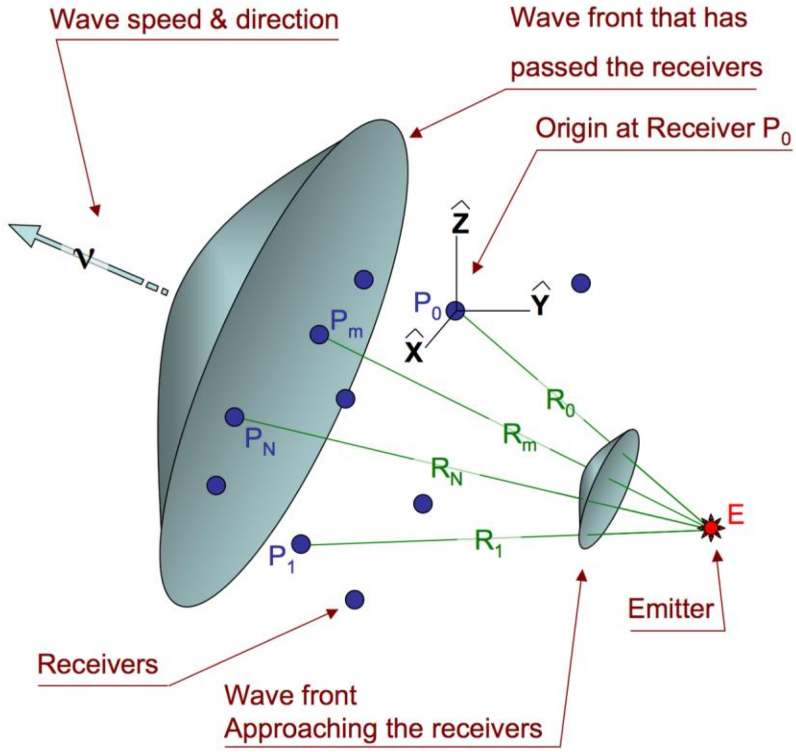

Figure 4-7: Diagram representing 3D hyperboloids [20]

4.5 Machine Learning:

Machine learning is one of the core learning points that we decided to insert into our project. It is vital that we master this aspect of our project because this is the basis of our detection algorithm for this project. By definition, Machine Learning is "scientific study of algorithms and statistical models that computer systems use to effectively perform a certain task". What this is saying that computers will learn stage by stage until it gets used to, most importantly, be as efficient of a system as it possibly can. This process is used worldwide. For instances, it is being utilized for filtering emails, computer vision, where usually it is very inconvenient to be able to learn algorithms properly for performing a specific task that the computer is being asked to do.

One crucial aspect of machine learning is the mathematical optimization which transmits methods or application domains and gets spread all throughout the field. There are subsystems of machine learning. One of which is supervised learning.

Supervised learning consists of a broad library of terms that makeup or explain what machine learning there is to have purpose for. Two types of supervised learning are classification and regression algorithms. For classification algorithms, it is considered as where outputs are restricted to a set of values that vary to be limited. For example, when filtering emails, the input would take the role of the incoming email from one client to another, wherein the output will be the folder where the email is classified into like the Inbox or the Sent Items. Meanwhile, Regression algorithms are named for the continuous outputs it produces. This matters in the fact that may have an arbitrary value within a random range. For example, we measure the temperature, a length of an object, or a price.

Another is unsupervised learning. Unlike supervised learning, unsupervised learning tends to build a model from a set such data that only comprises of inputs whereas the output is not needed. The main purpose of the unsupervised learning is creating a structure of a data. It can also discover patterns in a set of data which can group inputs in a set of classification.

Next we have Active learning. These algorithms access outputs which can be desired that is related to a input set that can be limited in size. When used, this type of algorithm is given to a user for labeling. A type of learning algorithm that closely sewed in active learning algorithms are Reinforcement learning algorithms. These algorithms are feedback based in a moving environment that presents feedback in the form of positive and sometimes negative. These algorithms are mainly used for autonomous vehicles or sometimes in playing a game with a human.

Machine learning as a whole is in good relations to optimization or the varying set of ideas that algorithm optimization explains for software developers. This is due to the different kinds of problems arising in creating these machines which can result into loss function to a set of examples. While this is the case, generalization is what separates the two from being in the like. With optimization, it is only concerned mainly to the minimize the loss on a set while machine learning is concerned with minimizing loss on samples that haven’t been discovered yet.

A useful machine learning technique is the utilization of an artificial neural network. Artificial neural networks are computational models inspired by the human brain. They consist of layers of neurons which summarize a set of weighted inputs before passing the result through an activation function to produce an output. The output of the neurons connects to the inputs of the next layer of neurons. Learning is achieved through backpropagation. An error value is calculated for the outputs of the neurons that provide an incorrect output, which then dictates how much the weights for each of its inputs are adjusted. Then the process is repeated for each neuron it is connected to in previous layers.

Neural networks are excellent for handling large amounts of data and has a high potential accuracy in real world situations. Categorical neural networks produce outputs that label a sample of data with a “category”. In supervised training, samples are manually provided labels which denotes the correct category for that sample. The network then produces a category for the sample and error is determined by comparing the category to the correct label.

5.4.1 Convolutional Neural Networks

A convolutional neural network consists of convolutional neuron layers with pooling and fully-connected neuron layers. Convolutional layers are considered to be more efficient at detecting repeating features across large multidimensional arrays [23]. The rough architecture of a convolutional neural network is shown in Figure 4-8.

A convolutional layer works by taking it’s input several in smaller subsets of neurons called receptive fields, rather than the whole thing at once in a fully-connected layer. This receptive field filters data through the weights of its inputs. This receptive field convolves over the data, utilizing data from local features to learn about other areas. Unlike a fully connected layer, there are less neurons and connections to adjust for repeating features, and repeated or similar weights are reduced significantly. The final output of this receptive field once it has passed over the entire input is called a feature map. A convolutional layer generates several feature maps for several receptive fields.

Pooling down-samples the data received in the feature maps by merging neighboring cells in blocks. Typically, this is either done by taking the average of the data, or the maximum value. This is important as, along with receptive fields, it increases the robustness of the analysis against data translation (changes of the sensor’s perspective of the same data) [23].

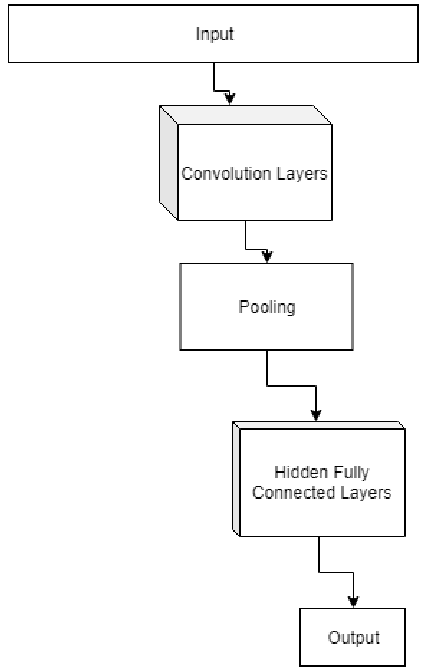

Figure 4-8: A simplified view of a convolutional neural network.

4.5.2 Activation Function

In an artificial neuron the sum of the combined weighted inputs for each neuron is passed through that neuron's activation function to produce its output. The activation regulates the output for the rest of the network. Some of the basic activation functions are step and piecewise functions, although these tend to limit the number of potential outputs to just one or zero, or on or a limited function in between. Logistic sigmoid or hyperbolic tangent functions offer a softer, asymptotic output, more variables, and were thus were more commonly used activation functions [23]. A Tanh activation function is shown below:

$$Tanh(x) = \frac{\sinh(x)}{\cosh(x)} = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

However, more recently deep neural networks have started to replace such models with alternatives that offer greater efficiency. One such common alternative is the Rectified Linear Unit (ReLU) and Leaky ReLU activation functions. They offer faster computation and resistance to saturation. It also offers this simplicity while still avoiding insufficient data discrimination. Leaky ReLUs also offer greater resistance to "dead zones" where the weights of the inputs will consistently force the output to produce zeroes, by providing a softer positive gradient to negative output instead [23]. A standard and leaky ReLU are shown in two equations below respectively.

$$f(x) = max(0, x)$$

$$f(x) = x\ (if\ x > 0);\ 0.01 * |x|\ (otherwise)$$

4.5.3 Loss and Optimization Algorithms

There are two algorithms at the core of learning in neural networks; the loss and the optimization functions. In training, the loss function calculates the error or loss of the network, while the optimization algorithm is designed to reduce the loss of the loss function by determining the parameters of the back-propagation. Usually they have a learning rate which adjusts how much the weights are changed every time back-propagation occurs.

A Cross-entropy (or log) loss function is common for categorical neural networks. If these networks have outputs with probability values (usually between 0 and 1), this function will determine the loss to be the difference between the predicted probability and the actual probability.

Stochastic Gradient Descent is a common optimization function for neural networks. The "gradient descent" part explains that the algorithm is attempting to find an error gradient, or slope, then "descend" down that gradient until the model reaches the lowest point. Stochastic implies something that is randomly determined. In normal Batch Gradient Descent, the optimization takes the error gradient of all samples and sums them together to form a final gradient descent. This can be problematic as it means collecting potentially millions of gradients and then summing them all together. Instead, stochastic takes the gradient of one random example.

Adam optimization function is an extension of the stochastic gradient descent based on two other extensions. Normal stochastic gradient descent maintains a single learning rate for the whole of the model. Adam creates per-parameter learning rates, and an exponential moving average of the error gradient and squared error gradient which is used in adjusting these learning rates. Adam is more efficient for non-stationary (dynamic/changing) situations [37].

4.5.4 Dropout Learning

While deep neural networks have proven to be more efficient than shallow neural networks, especially in classification, there is a notable drawback to this approach. As data is passed through several layers, the output becomes more complicated than it needs to be which creates inaccuracies. This is called oversampling. Dropout learning is an attempt to address this by randomly eliminating neurons within a layer. These random disruptions (of a predefined probability) keep the network focused by cutting out unnecessary connections. They also make it simpler over time [23].

4.6 Digital Signal Processing

In order to properly present the data in a way that's appropriate to the neural network, and provides the most amount of information, the audio has to be processed to provide more information about the signals it receives. Computers record audio digitally over time. It takes a sample of the audio magnitude at a specific point in time, then takes another sample. This is done in a sample rate of a certain number of samples per second. Therefore, audio files are simply presented as arrays of these samples in chronological order.

This information is of course necessary but it's not complete. More information can be learned from the frequency of information in the sample. Specifically, this details how often a particular measurement is recorded over the recording. This helps to inform about patterns in the signal.

The goal is to present this information together as a single "image" for processing. The more data that can be collected from each data sample, the more that can be learned from it by the machine learning algorithm. This will increase accuracy.

4.6.1 Fourier Transform

The Fourier transform is a common method of either obtaining frequency domain information from a time domain source, or time domain information from a frequency domain source. The former is called analysis, while the latter is called synthesis. Since we already have audio files in time domain format, we will mostly be looking into analysis [26].

There are two analysis functions, since the frequency domain is represented with both real and imaginary graphs. That is because frequency is represented as the sum of a real number and an imaginary number.

The real frequency analysis function (depicted as $Real\ of\ X()$) is the sum of the product of the time domain amplitudes times the cosine wave of the frequency index b [26]:

$$Real\ of\ X(a) = \sum_{a=0}^{N-1} x(a) * cos[(2 * \pi * b * a)/N]$$

Likewise, the imaginary frequency analysis function (represented as $Imaginary\ of\ X()$) is the sum of the product of the time domain amplitudes times the sine wave of the frequency index b [26]:

$$Imaginary\ of\ X(b) = -\sum_{a=0}^{N-1} x(a) * sin[(2 * \pi * b * a)/N]$$

Since a frequency cannot be more than half the total number of samples, the frequency index b runs from 0 to half of the total number of samples.

These functions represent the rectangular form of the frequency domain. A cartesian coordinate graph, where the x-axis is frequency and the y-axis is amplitude. We can convert this into polar coordinates. First, we find the magnitude, which is the square-root of the sum of the squares of the real and imaginary analysis functions [26]:

$$Magnitude(b) = \sqrt{ReX(b)^2 + ImX(b)^2}$$

And the phase analysis function is the arctangent of the real analysis function divided by the imaginary analysis function [26]:

$$Phase(b) = arctan[ReX(b)/ImX(b)]$$

A big drawback to using Fourier transform is that it relies on the signal being stationary. As in, a certain frequency is expected to be equally present across the entire signal recording. This is not true in reality, which often presents non-stationary signals and data.

4.6.1.1 Short-time Fourier Transform

A simple approach to this problem is short-time Fourier transform. It divides the signal recording into "windows" of samples, and applies Fourier transform to each window instead of across all of the samples at once. This information can then be depicted as an image. The value of each pixel is the magnitude, but the location refers to time in the x-axis and frequency in the y-axis. This allows the frequency information to be depicted over time.

However, this approach produces a drawback of its own in the form of data resolution (the amount of data to work with). The size of these windows affects both the frequency and time resolution. The larger the windows, the higher the frequency resolution but the lower the time resolution and vice versa [25]. This information is still useable, but the window size is fixed, and cannot adjust to the changes throughout the signal, where more frequency or time resolution would be needed.

4.6.2 Wavelet Transform

The wavelet transform is another approach to signal analysis. It attempts to resolve the drawbacks of short-time Fourier transform by providing a time-frequency analysis that can adapt based on the resolution needed. If there is a high-frequency part, the window shifts for high time resolution and low frequency resolution. If there is a low-frequency part, the window shifts for low time resolution and high frequency resolution. Thus, this addresses the dynamic/non-stationary signal issue, while also addressing the resolution problem [25]. This is shown in Figure 4-9.

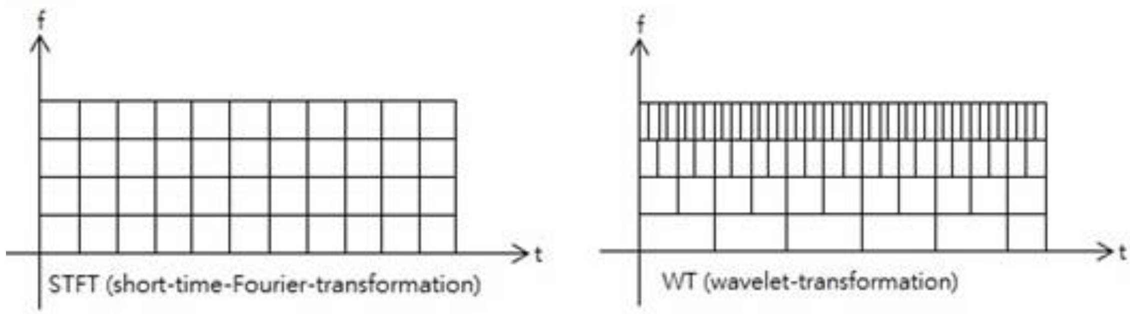

Figure 4-9: An example displaying the differences in implementation of frequency windows over time. Obtained from Wikipedia, free to share.

The continuous wavelet transform is a function, $W(a,b)$, with parameters defining the expansion factor, a, and the shift factor, b. It is equal to the integral over negative and positive infinity, of the signal function, $f(t)$, times the complex conjugate of the wavelet sequence $\psi_{a,b}(t)$. This is shown in the following equation [25]:

$$W[a,b;\ f(t),\psi_{a,b}(t)] \ = \int_{-\infty}^{+\infty} f(t) \ * \ \underline{\psi_{a,b}(t)}\, dt$$

The wavelet sequence is the result of the expansion and shift of the mother wavelet function $\psi(t)$. It is the product of one over the square root of the expansion factor times the mother wavelet function, with parameters of time minus the shift factor divided by the expansion factor. This is shown below [25]:

$$\psi_{a,b}(t) = \ 1/\sqrt{a} * \psi(\frac{t-b}{a})$$

As these are examples of continuous wavelet transform, the formula must be discretized in order to be used in our program. In discrete form, the wavelet sequence changes. Although it is still a result of the shift and expansion factors applied to the mother wavelet function, the process is somewhat different. It is now the product of two to the negative expansion factor divided by two, times the mother wavelet function with parameters two to the negative expansion factor, multiplied by time minus the shift factor. This is shown in the equation below [25]:

$$\psi_{a,b}(t) = \ 2^{-a/2} * \psi(2^{-a} * t - b)$$

This results in a final wavelet transform analysis function by bringing in the new discretized coefficients. Where the function is now equal to two to the expansion factor divided by two, times the integral of the signal function times the mother wavelet function with parameters two to the negative expansion function, multiplied by time minus the shift factor. This is shown in the equation below [25]:

$$W[a,b;\ f(t),\psi_{a,b}(t)]\ = 2^{a/2}\int_{-\infty}^{+\infty} f(t)\ *\ \psi(2^{-a} * t - b)\, dt$$

5. Design

From the research gathered, it was possible for the Minuteman to be designed. This process included a multitude of design tasks that were completed. Everyone was responsible for their own subject while designing the Minuteman. This helped in the design flow and allowed those individuals to fully understand its application. If each individual is becoming an expert in their own part it will help in integration and testing. With possible failures and troubles later in this program, troubleshooting will become easier if individuals take responsibility for their part.

It is important to maximize the quality of our initial design. Detail shall be taken when designing the Minuteman. Then when the integration of the Minuteman's modules comes, it will have either minor errors or none at all. This is to avoid redesign in the integration section of this project. The responsibility of each individual depends on the designation of their roles. If something fails in the system, then it is the group's responsibility to fix the issue. Like stated above, it is important that the members know their design so they can provide detailed support in the troubleshooting effort.

5.1 Standards

Standards are a means of a providing common guidelines for products and services across an industry, nation, or globally. They are utilized so that new products or services can be introduced into a market in a way that it can be made compatible with other products/services or to keep these products/services safe. Standards can either be self-imposed/regulated, or enforced by law depending on the nation or supranational entity.

5.1.1 802.11 Standards

With 802.11, it is a group of standards that consists wireless techniques that include the same protocols that indicate Layer 2 protocols. The first standard we have is the 802.11a which utilizes a 5 GHz frequency band while the next standards like the IEEE 802.11b and 802.11g use the 2.4 GHz frequency band. The 802.11b and the 802.11g can operate in an unstable and unregulated frequency band, there has a chance of interference or noise from phones and other devices. The IEEE standards only sets specs and does not particularly test certain equipment for compatibility. Just like in Wi-Fi, it can mean 802.11a, 802.11b, or 802.11g, and also does not include the security standard protected access or the WPA or even the WPA2. The Wi-Fi products should note the frequency in which a device runs. The maximum rate of data varies enormously between the different set of standards. Table 5-1 shows the wireless standards below.

IEEE standard	**802.11a**	**802.11b**	**802.11g**	**802.11n**	**802.11ac**
Frequency	5 GHz	2.4 GHz	2.4 GHz	2.4/(5) GHz	5 GHz
Data Rate	54 Mbps	11 Mbps	54 Mbps	600 Mbps	1 Gbps
Typical Range(Out)	100 ft	100 ft	125 ft	225 ft	90 ft
Typical Range (In)	400 ft	450 ft	450 ft	825 ft	1000 ft

Table 5-1: Table of Wireless Standards

5.1.4 I^{2}C

The I2C or as it's called, I squared C, is a "de facto world standard" that is implemented in over 1000 ICs all around the globe that is manufactured by 50 plus companies. The I2C-bus is used in various designs such as a System Management Bus, Power Management Bus, Intelligent Platform Management Interface, Display Data Channel, etc.

When it comes to designing I2C compatible chips, all specifications and proper standards must be met in order to ensure a successful creation of these chips. There should be careful review with the right protocols before making these and always refer to component data sheets.

Wires like the serial date (SDA) and the serial clock (SCL) transmit relative information between devices that are connected to this I2C bus. Every device is noticed by a unique address which can act as a transmitter or a receiver depending on the function. Devices can be applied as masters or slaves when transferring data between devices. The master is the device that initiates a data transfer to the bus which also generate clock signals. The one addressing this transmission is the slave.

The generation of certain clock signals on the bus is assigned to the master devices, the master devices generates its own clock signals whenever it transfers data on the bus itself. The clock signals coming from the bus can be modified when they are stretched by a slow slave that is holding down the line or when another master device when an arbitration occurs.

Table 5-2 shows the features of a I2C bus protocol.

Feature	**Single Master**	**Multi-Master**	**Slave**
Start condition	Mandatory	Mandatory	Mandatory
Stop condition	Mandatory	Mandatory	Mandatory
Acknowledge	Mandatory	Mandatory	Mandatory
Synchronization	Not applicable	Mandatory	Not applicable
Arbitration	Not applicable	Mandatory	Not applicable
Clock Stretching	Optional	Optional	Optional
7-bit slave address	Mandatory	Mandatory	Mandatory
10-bit slave address	Optional	Optional	Optional
General Call Address	Optional	Optional	Optional
Software reset	Optional	Optional	Optional
START byte	Not applicable	Optional	Not applicable
Device ID	Not applicable	Not applicable	Optional

Table 5-2: I2C protocol features

Table 5-2 illustrates the different characteristics of the I2C protocol that is required to be looked at by people who are willing to be implementing these buses into their projects. This table serves as a foundation or mini manual for developers and what the required standards are like for the I2C bus.

5.1.5 C# programming

The goal in terms of programming in C# language is to get the right practices that will result in good programming by the developers in the team. With this kind of extensive project, it is required to have knowledge of what is considered good in coding such programming languages because of how clean and flow smoothly it should be in the first place. There are numerous amounts of conventions that a C# developer should remember when coding up programs.

First, we have naming conventions. In this aspect of C# programming it is considered good practice to name the variables in c programming to something that can be easy to understand. If the developer knows that a particular namespace is imported by default in a project, we don’t have to qualify the names from that same namespace. The priority is with the owner of the cs file to abide the rules of the naming conventions which can be interpreted by the programmer and/or others as well. It can be seen as very inefficient to traverse through various files without having the trouble of finding the right files you’re looking for. The next one in naming conventions is that you don’t have to change the names of objects that were created in Visual Studio tools.

The next priority standard are layout conventions. With layout conventions, it is without a doubt, a high stakes priority when setting up code. With good layout comes out a better-looking structure for the code, not only for the creator to understand properly but also for the reader who would be willing to read the code and get help from later. First, the developer only has to write one statement per line. Next, you can only write one declaration per line as well. Additionally, if continuation lines are not indented automatically, you have to indent such with a one-tab stop. Next, add at least a blank line between method definitions and property definitions. With this, the reader knows which is in analyzing such code. Lastly, necessarily use parentheses to make clauses in an expression apparent:

```
if ((val1 > val2) && (val1 > val3))
{
   // Take appropriate action.
}
```

Next we have commenting conventions. Just like in C or C++, commenting can be done using ‘//’ or ‘/* */’ for larger chunks of code. For better organization, placing the comment on a separate line, not at the end of a line so it looks cleaner. Next, we must begin a comment with an uppercase letter. Additionally, we end comments with a period to show courtesy in creating code.

Some other good practices in C# programming is for variables. Use implicit typing when initializing local variables. Do not use var when the type is not precisely apparent from the right side of an assignment. Developers cannot rely on variable names to be able to specify the type of the variable; it isn't always correct.

5.1.6 Python Programming

Outside of other programming languages that are being used around the world, we can consider the fact that some languages are easier to understand than the others. While others can be more direct, like doing C++ or C programming, we have a more feasible approach such as Python. Python isn't necessarily a programming language but a script. Which is simple in developing because of the simple function conventions, variable naming, less unnecessary added distractions in the code and such.

With Python, we use single quotes for string literals. Examples like 'hello' or 'goodbye'. We use double quotes for single quote characters or strings that contain natural language. Single quotes are easier to read with the human eye. With triple single quotes, we use those for docstrings.

Usually the way to start a python script is to declare numerous imports. When importing from a module, we used parenthesis around the names that we are importing so it looks more organized. Python is less tedious compared to the other development mediums developers use out there, it is more straightforward in the sense that you don't have to think about outside matters when coding such script. The simpler the structure of the language is, the easier it is to understand for the newer and up and coming software engineers. It is not as known as the more popular ones but it is still in use today especially for data.

5.1.7 USB 2.0

USB 2.0 Hi-Speed is the natural choice for low-latency, inexpensive communications. 512-byte bulk transfers at 480 Mbps offers a latency around 8.5 microseconds, which fits without disturbing the 44.1 kHz sample rate even after the microsecond of USB overhead.

5.1.7.1 USB Electrical, Mechanical and Environmental Compliance Standards

The USB is one of the most prominently used devices in the computer world nowadays being used for data storage by people connecting them to their laptops or desktops to be able to store valuable information that they can bring around wherever they want to in a small holdable device that they can put in their pockets for convenience. Like all produced apparatus in the technological society today, there has to be implemented standards that pave way to how a device like the USB is supposed to be worked on and how the safety procedures relay in information for the users to fully capture the essence of this instrument wherein there will be no mere consequences that arise when these things are used by the consumers. The following table below illustrates the different criterias in determining the standards for USB cables, how assemble these cables and the connectors that are connected in the sources.

Test Description	**Test Procedure**	**Performance Required**
Visual and Dimensional Inspection	Visual,dimensional and functional inspection in accordance with the USB quality inspection plans	Must Meet or exceed the requirements specified by the most current version of the USB specs
Insulation Resistance	This test procedure is to detail a standard method to assess the insulation resistance of USB connectors. To determine the resistance offered by the insulation materials and various seals of a connector to a DC potential	1000 MegOhms minimum
Dielectric Withstanding Voltage	Detail a test method to prove that a USB connector can operate safely at its rated voltage and withstand momentary over-potentials due to switching, surges	The Dielectric must withstand 500 V AC for one minute at sea level
Random Vibration	A USB connector has to function during vibration or to survive conditions that of vibration should be clearly stated by the detailed product specifications.	No discontinuities of 1 microseconds or longer duration when mated USB connectors are subjected to 5.35 Gs RMS.
Thermal Shock	To determine the resistance of a USB connector to exposure at extremes of high and low temperatures and to the shock of	10 cycles -55 C and +85 C. The USB connectors under test must be mated

	alternate exposures to these extremes, simulation worst case conditions.	
Solderability	To detail the uniform test method for determining USB connector solderability. The test procedure contained utilizes the solder dip technique.	USB contact solder tails must pass 95% coverage after one hour steam aging as specified in Category 2.

Table 5-3 : Standards for USB 2.0

Table 5-3 displays some of the important safety measures laid out by the people who created Universal Serial Bus'. If it wasn't for these standards, there will be major consequences in the end of it all. No one knows how severe these complications will be and that's why there should be applicable solutions to each and every part of the traits a USB possesses. There are many more standards to apply to USB's but we have implemented some of the really important and most well-noted ones for our report.

5.1.8 American Wall Outlets

In the world, there are many different kinds of outlets that show users where they can connect their devices into to gain power such as appliances like the television, stoves, smartphone chargers, etc. Without these sockets/outlets there wouldn't be any source of power to make these tools work. There are numerous amounts of outlets that are offered all around the world which offer a unique design and structure for each of them. Some of the different outlets offer 2 pins while there are sockets in which there are 3. As for this project, we will only focus on the american made outlets because that is what we will be using to provide power for our Minuteman system which enable our audio sensors as a whole and power up all the necessary inputs to be able for our project to work properly. The the standards for these outlets are the following:

TYPE A	**TYPE B**
Mainly used in the USA, Mexico, Canada, Japan	Mainly used in USA, Mexico, Canada, Japan
2 pins	3 pins
Not grounded	grounded
15 amps	15 amps

Almost always 100 to 127 V	Almost always 100 to 127 V
Socket compatible with plug type A	Socket compatible to plug type A and B

Table 5-4 : Standards for American Outlets

Table 5-4 above illustrates a few of the details that incorporate outlets in america in specific. To go beyond further what these offer, Type A plugs and outlets are as known as NEMA 1-15. The Type A plug has two flat 1.5 mm thick blades, which are polarized and can be only inserted in a particular way because the two prongs don’t have the same shape. The blades are neutral to 7.9 mm wide. In contrast, Type B systems are designated as standard NEMA 5-15. Like the Type A, it also has 1.5 mm thick blades wherein the device is grounded before it even it connected. In addition, it has a 4.8 mm diameter round or u-shaped pin. Both Type A and B plugs are not insulated.

5.1.8.1 Electrical Safety Standards

For our project, it is critically required for our system to be safe for use by others later on even after the team graduates so that we have a good reputation from numerous consumers out there. Safety mechanisms such as enclosures for systems that can prevent any electrical contacts from being shorted and such. Other important Occupational Safety & Health Administration (OHSA) standards when it comes to safety for specific electrical related subsystems in terms of interactions with numerous users serve as the following:

- Electrical Power Generation, Distribution - what this piece of informations relays is that it regulates operation and the keen care for electrical power generation, control, transmission, and distribution lines.
- Electrical Protective Devices - This regulates the design requirements in relation to different types of equipment for electrical systems that include such as rubber insulating covers, rubber insulating gloves, and sleeves which have to meet the certain project requirements listed in by the team in this report
- Wiring Design and the means of Protection - This standard illustrates the labeling of device wiring and the aspect of protection in the design of the Minuteman system. This also allows proper grounding terminal connections are perfectly labeled all throughout the project.
- Hazardous locations - This standard relay in requirements for hazard equipment that are particularly classified whether it may be flammable vapors, liquids, gasses, combustible dusts and fibers that may show the possibility of a combustible concentration quantity

- Use of equipment - regulates the use of plug and cords connected in major equipment used for the project. This includes flexible cord like extension cords.
- Personal Protection - regulates the use of personal equipment needed for ample safety in case there is a troublesome event that may happen during the design stages of the project. Anyone working possible electrical hazards have to be very careful in which parts can go with others and not potentially short circuit and burn needed parts. Everyone in the team has to use protective equipment that is appropriate for parts of the body that needs to be protected.

In summary, safety has to be the number one priority for our team from start to finish. We cannot let ourselves get too loose with the equipment that we are working with because someone can easily just accidentally burn or destroy a part that may be too expensive for student funds that will be hard to replace later on. It is also imperative that we look out for each other in case something unfortunate happens during the implementation stage that may affect our states mentally and physically. This only cannot harm our education as a whole but it can also potentially destroy our self being and put our bodies in harm as well. Dealing with these kinds of tools are no joke. That's why only experienced professionals usually handle these devices. There will be strict embodiment of rules between the team to always make sure every piece of equipment works properly and that nothing is connected incorrectly.

5.2 Hardware

The design of the hardware in the project had constraints from the cost and the components available for the project. The parts were exchanged based on further research done in the project. There was careful consideration and debate where the group understood each person's opinions on what the right parts to get was. There was need to implement new parts later, because of the complexity of the Minuteman project. Table 5-5 further shows the features and specifications of the hardware.

Feature	**Hardware Specifications**
Microcontroller	AT91SAM3X8E
Power Adaptor	12V DC 1 Amp Power Adapter Supply
Switching Regulator	Analog Device LTM4622A Dual Ultrathin 2A or Single 4A Step-Down DC/DC µModule Regulator

	Texas Instruments LMR23610ADDA Buck Switching Regulator IC Positive Adjustable
Voltage Inverter	MAX1044 CMOS Switched-Capacitor Voltage Converters
Microphone	CMC-6035-130T, CUI, Mic Omni-Directional 2200Ohm -42dB 2VDC Round Solder Pad
Gyroscope, Accelerometer, Geomagnetic Sensor	BNO055 9 DOF Absolute Orientation IMU Fusion Breakout Board
Global Positioning System (GPS)	u-blox NEO-6M GPS Module
Amplifier	Texas Instruments TL072
ADC	Texas Instruments ADS8586SIPMR

Table 5-5: Hardware Specifications

5.2.1 Central Module

The Central Modules purpose is to be the connector between the Sensor Module and the software program on the computer. In the initial design of the Minuteman, it was planned to use triangulation. There would be two sensor modules with 5 microphones each, both connecting to the Central Module. This changed when multilateration became feasible. With this development the Central Module and Sensor Module could be compartmentalized into one hub. This type of compartmentalization could provide a high rate of reliability, since large functions of the Minuteman were modulated.

Due to the use of multilateration, the Central Module housed almost all the hardware except for the microphones. These microphones were spread out from the Central Module. The central housing unit is large enough for our design to be able to contain the PCB and assortment of sensors. The device was not able to be waterproof; with more time and resources a water-resistant device housing could have been completed. The Central Module device housing is a rectangular box with enough room for a sizeable PCB. The module has an input and output port for the microphones, the USB, and the wall adapting power supply.

5.2.2 Microcontroller

The Microcontroller unit (MCU) is a less sophisticated computer. It usually contains RAM, Program Memory, Microprocessor, Analog to Digital Converter, Input/Output Pins, and a clock generator. All these functions are combined on an integrated circuit in various ways to make many different microcontrollers. The microcontroller is a very compact device able to electrically manipulate analog and digital information. These devices all have different uses depending on their configuration/design. To understand the use of these devices you must read the datasheet and know which configuration you need. For this project the group researched to pick the best microcontroller that would fit our requirements.

5.2.3 Development board

The boards below are a selection of development boards. These development boards have microcontrollers integrated inside of them. With other functions integrated within those microcontrollers. The development boards make it easier for designing, because there is a large support from the development board designers. One of these boards will be used for easy integration for our project. This will also help in the testing and design. When it is required to create our own PCB it will be easy because, whatever development board that was chosen, the microcontroller that is inside that development board will be used in the Printed Circuit Board (PCB).

5.2.3.1 Arduino Due

The Arduino series of boards are one of the most favored of development boards. One such reason is the integrated development environment (IDE) that Arduino has. Their IDE is very similar to the C language and so it is very easy to pick up if you have previous coding experience, including having learned the syntax and implementation. During the initial glimpse, this board was very impressive with a 3.3VDC operating voltage and 54 Digital Input/Output Pins and 12 Pins with ADC capabilities providing PWM output. This is exactly what was needed for the design. Though some of the specification may not be needed, they might be used later on if the initial design is expanded upon later in the development of this project. This board is also fiscally responsible, costing $38.50 and including the development information that Aduino provides.

Microcontroller	AT91SAM3X8E
Operating Voltage	3.3VDC
Input Voltage (recommended)	7VDC to 12VDC
Digital Input/Output Pins	42

Digital Input/Output Pins capable of Pulse Width Modulation	12
Analog Input Pins	12
Analog Output Pins using a Digital to Analog Converter (DAC)	2
Flash Memory	512 KB
SRAM	96 KB
Clock speed	84 MHZ
Universal Asynchronous Receiver/Transmitter (UART) communication	Yes
USB Protocol	Yes
Reset Protocol	Yes
Overvoltage Protection	Yes
Overcurrent Protection	Yes

Table 5-6: ARDUINO DUE Ideal Specifications

5.2.3.2 ARDUINO MEGA 2560 REV3

Like the Arduino Due, the Arduino MEGA 2650 REV3 is a capable board for this project. During the initial inspection, it has no issue with the amount of Input/output Pins needed. This board provides extra protection as well with more pins for PWM. These two board are very similar and contain all necessary aspects for this project. This is a comparable board to the Arduino Due. The Arduino MEGA 2650 trades off some hardware for different configurations like more coding pins, but less memory for coding compared to the Arduino Due. This board also has a smaller clock speed than the Arduino Due. Costing the same as the Arduino Due at $38.50, the Arduino MEGA 2650 REV3 is a different board with similar capabilities with minor changes for different applications. Just like the Arduino Due, the Arduino MEGA 2650 REV3 comes with extra hardware that might not be needed. This hardware could be used later if extra or unseen qualities are added to the project.

Microcontroller	ATmega2560
Operating Voltage	5VDC

Input Voltage (recommended)	7VDC to 12VDC
Digital Input/Output Pins	38
Digital Input/Output Pins capable of Pulse Width Modulation	16
Analog Input Pins	16
Flash Memory	256 KB
SRAM	8 KB
Clock speed	16 MHZ
Universal Asynchronous Receiver/Transmitter (UART) communication	Yes
USB Protocol	Yes
Reset Protocol	Yes
Automatic Reset (Software capable to reset Arduino MEGA in code)	Yes
Overvoltage Protection	Yes
Overcurrent Protection	Yes

Table 5-7: ARDUINO MEGA 2560 REV3 Ideal Specifications

ARDUINO UNO REV3

Looking at the Arduino Uno REV3 it is noticeable the difference in its capabilities. The Arduino Uno has a noticeable drop in hardware capabilities. With only 14 Input/Output Pins, this board has considerably less capable than the previous two. The hardware in this board is half the specs of the Arduino MEGA and Arduino Due. It does not have enough Pins for this project and would require multiple boards if used. Each individual sensor module would need its own board. Though this board is less capable, its price is cheaper than both the Arduino Due and Arduino MEGA costing $22.00, a little over half the cost of the other two boards.

Microcontroller	ATmega328P
Operating Voltage	5VDC

Input Voltage (recommended)	7VDC to 12VDC
Digital Input/Output Pins	8
Digital Input/Output Pins capable of Pulse Width Modulation	6
Analog Input Pins	6
Flash Memory	32 KB
SRAM	2 KB
Clock speed	16 MHZ
Universal Asynchronous Receiver/Transmitter (UART) communication	Yes
USB Protocol	Yes
Reset Protocol	Yes
Overcurrent Protection	Yes

Table 5-8: ARDUINO UNO REV3 REV3 Ideal Specifications

5.2.3.3 BeagleBone Black

The BeagleBone Black is just or more capable than any of the other Arduino boards. Beagle board gives out a large amount of information about their project, which helps in our development process. This is the same as the Arduino boards. The BeagleBone Black might be more than what is needed for this project. With 92 Input/Output Pins this board is not only able to do the job required, it might be to much even if the project were to be added on to. With having a AM335x 1GHz ARM Cortex-A8 processor. This board is made for a more complex project with video and other functions. When comparing its price, it is cost effective for what you get in functionality, but it is much more than what this project requires. Costing $62.38, this could be a great test board for our project.

Microprocessor	Sitara AM3358BZCZ100
512MB DRR3L	512MB DRR3L
Operating Voltage	1.5VDC, 3.3VDC, and 5VDC

Video Out HDMI (Graphic capability)	Video Out HDMI (Graphic capability)
Digital Input/Output Pins	92
Flash Memory	2 GB
Clock speed	24.576 MHZ
Universal Asynchronous Receiver/Transmitter (UART) communication	Yes
USB Protocol	Yes
Reset Protocol	Yes
Overcurrent Protection	Yes
Reliability	Yes

Table 5-9: BeagleBone Black Ideal Specifications

5.2.3.4 Field Programmable Gate Array (FPGA)

The Field Programmable Gate Array (FPGA) is a simple product to understand, but a complex device to implement. The reason why it is easy to understand is that it is a device with no hardware until it is coded. This mean that the FPGA has no physical hardware until someone codes the hardware onto it. If you look at Figure 5-1 you can see the top down representation of what a FPGA looks like.

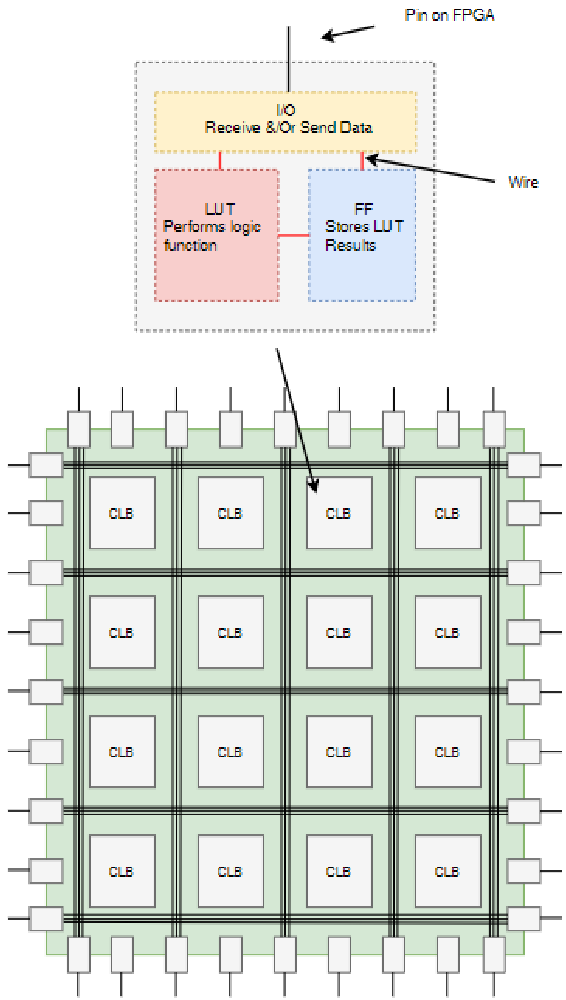

Figure 5-1: FPGA Diagram

The board is an array of blocks that are coded to form many different tasks. These blocks are called Configure Logic Blocks (CLB). Each one of these blocks has a assortment of internal components. The internals of these block are what is programmed. The CLB can perform Analog to Digital Conversion, Digital Signal Processing, and other assortments of logic configurations. The Internal of the CLB are also present on Figure 5-1. The LookUp Table (LUT) is a truth table where for every set of inputs you have outputs. The LUT is what is programmed to have different functions and results. Those results are sent to the Flip Flops to be stored. The stored value are either transferred to other CLB or sent to an output pin at the borders of the FPGA diagram.

These devices are very powerful and can come in a large range of configurable specs. With the more complex FPGA needing more Input/Output Pins. This plays a large part of how much the FPGA will cost and how much it is able to do. Because the FPGA is programmable hardware, it is able to reconfigure many times. This mean in one configuration a group of CLB could be an ADC and another configuration it could work as a switch. That is why it has ‘Field Programmable’ at the start of it name. Some provide a limited number of reconfigurations and others provide an unlimited amount of reconfigurations.

The FPGA has a steep learning curve, even with enough sources online. It requires you not only to understand the hardware you are creating, but you must also understand the process of creating it as well. The user is required to learn one of the Hardware Descriptive Languages. The two most common languages are VHDL and SystemVerilog. For both languages, they have a process of designing and creating the hardware in the system as well as a step for verification and testing. These two sections are broken up into design and verification. For this project, SystemVerilog is being studied for application.

Just like the Microcontroller the choice of FPGA are depended on our need and what the complexity its job will be. The best choice will be dependent on the need for the project and complexity that the designer wants.

5.2.3.5 Mpressions Odyssey MAX 10 Evaluation Kit

The choice of using a MAX 10 Intel FPGA was because of the easy of access to the development application provided by Intel. In this project using, a Evaluation Kit for the FPGA integration would save time and reduce complexity. The MAX 10 Intel FPGA was the a deciding factor when looking at different boards, with a large amount of user support from Intel that will help in the integration part of this project.

This board is capable of providing a lot of utility with an audio monitor able to connect straight to a microphone to process the data. The Mpressions Odyssey is also able to connect with the Arduino boards for integration. The Mpression has an extensive user guide as well.

Microcontroller	EFM32 (USB Device)
FPGA	Intel MAX 10
Arduino Compatibility	Yes
Accelerometer	Yes
Light Sensor	Yes
Temp/Humidity Sensor	Yes
Audio Input	Yes

Operating Voltage	3.0VDC
Digital Input/Output Pins	14
SRAM	512 KB
Flash Memory	2 MB
Clock speed	50 MHZ
USB Protocol	Yes
Bluetooth Protocol	Yes
Overcurrent Protection	Yes
Reliability (Test Ports)	Yes

Table 5-10: Mpressions Odyssey MAX 10 Evaluation Kit Ideal Specifications

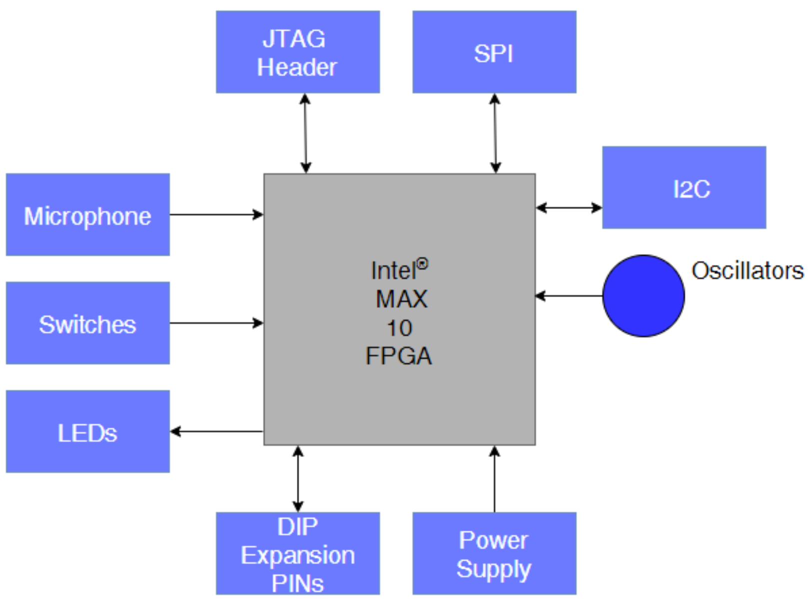

Figure 5-2: Mpressions Odyssey MAX 10 Evaluation Kit Diagram

5.2.3.6 MAX® 10 FPGA Evaluation Kit

The MAX 10 FPGA Evaluation Kit is a board made by Intel using their MAX 10 FPGA. This kit was made so that the FPGA could be tested with many different application. Intel provides an extensive amount of information in the user guide with videos and different design application for this board.

When comparing Mpressions Odyssey to this board, the MAX 10 FPGA Evaluation Kit has more connection ports. It allows for the user to manipulate it better, since it has less already integrated hardware. This will be better for our project since in these stages it might change in the design. The Mpressions Odyssey is initially more capable than the Intel Kit, because it has hardware already interfaced to it. While the intel kit has very little, this means that the Intel Kit is more flexibility for new hardware integration.

FPGA	Intel MAX 10
Arduino UNO Rev3 Compatibility	Yes
Potential Potentiometer	Potential Potentiometer
Operating Voltage	5VDC
Digital Input/Output Pins	40
Arduino Input/Output Pins	14
Analog Input/Output Pins	8
Flash Memory	172 KB
Clock speed	50 MHZ
Bucked Power to Filtered	3.3VDC
Mini USB Protocol	Yes
USB Protocol	Yes
Reset Protocol	Yes
Overcurrent Protection	Yes
Reliability (Test Ports)	Yes

Table 5-11: MAX® 10 FPGA Evaluation Kit Ideal Specifications

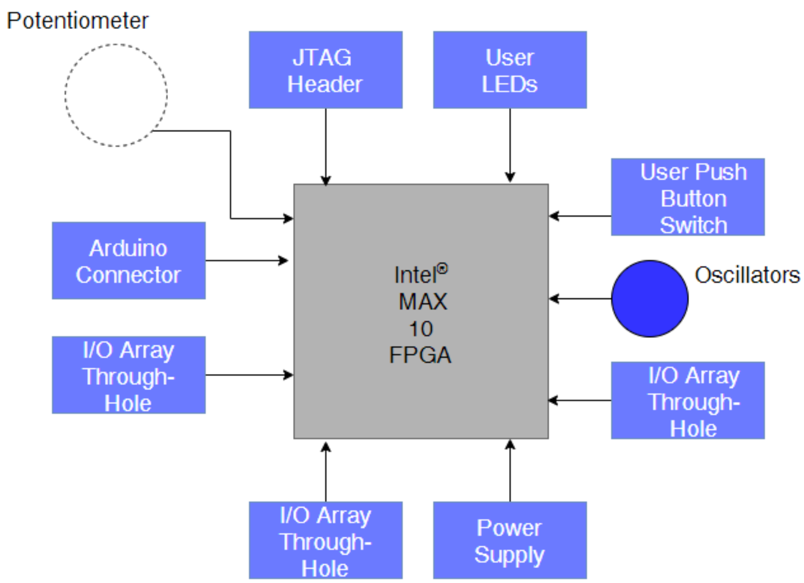

Figure 5-3: MAX® 10 FPGA Evaluation Kit Diagram

The results of the search for the right board has come to a decision. One is to take the Arduino Due by itself to do the majority of the signal analysis. Other designs were to take the Arduino UNO REV3 and combine that with the MAX® 10 FPGA Evaluation Kit. Both these decisions do not change the results of the power supply, but it does change a lot of the hardware design and how the modules work. The choice of this project depends on the requirements of the sensors and the function that the development board must complete. The Arduino Due is the best development board for this project. This also means that with later development of our PCB the AT91SAM3X8E microcontroller will be used for the project.

5.2.3.7 AT91SAM3X8E

The AT91SAM3X8E microcontroller designed by Microchip Technology was chosen from the Arduino Due. The AT91SAM3X8E microcontroller shall provide the best results of the development boards that were screened. One of the main functions that was required for our design was a USB compatibility. This will help with our hardware to software integration. As well as are large amount of GPIO pins and a large library for software.

Processor	ARM Cortex-M3
Operating Voltage	3.3VDC & 1.8VDC (self regulated)
Voltage Regulator	1.8VDC
Programmed input/output (PIOs)	103

Flash Memory	256 to 512 KB
SRAM	32 to 100 KB
ROM	16 KB (bootloader routine/startup)
Clock speed	84 MHZ
ADC	12-bit, 16 channel
DAC	12-bit, 2 channel
Timer	32-bit, 9 channel
I2C	2
Serial Peripheral Interface (SPIs)	6
USART/UART	3/2
Low-Power Mode	2.5 µA
Power-on Reset (PoR)	Yes
USB Protocol	Yes
Ethernet Protocol	Yes
HSMCI Interface	Yes
Dimensions (L x W x H) mm | in	22 x 22 x 1.4 | 0.866 x 0.866 x 0.055

Table 5-12: AT91SAM3X8E Ideal Specifications

With the size of this device at 484 mm^2 it is the largest device that's on the PCB. The placement of this device was important because it took up so much space. When designing the PCB this microcontroller was placed in the most optimal station so the traces could be short, and it can be sectioned for functional purpose. This also helped in the development of our PCB because the analog and digital sections could also be divided. With this it was also very important that the USB was isolated since it has a high data transfer rate. The USB provides communication between the Central Module and the computer to interface with the software application.

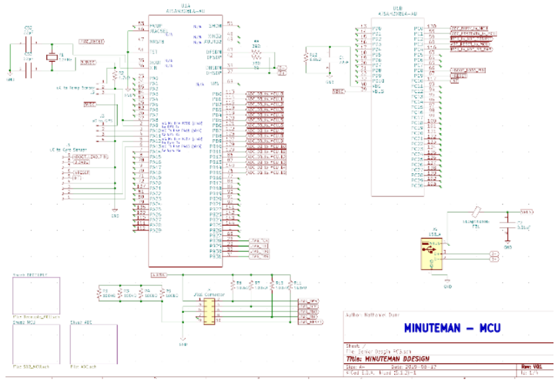

Figure 5-4: MCU Schematic Layout

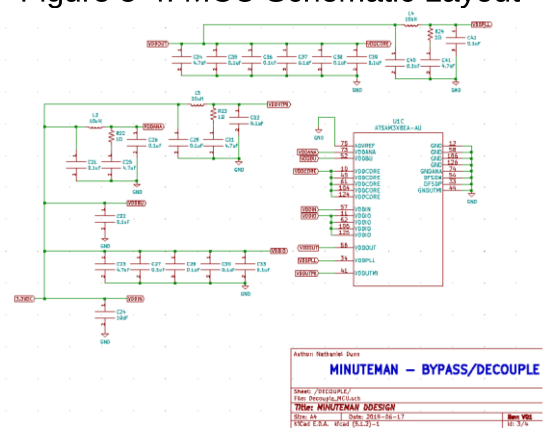

Figure 5-5: MCU Power Schematic Layout

5.2.4 Printed Circuit Board

The Printed Circuit Board (PCB) was to be designed in the Central Module of this project. The hardware requirements that the Minuteman needs to function determined the size of the PCB itself. With the initial change from triangulation to multilateration the size of the PCB has increased. With more components being added to the PCB, it is a beneficial to separate the PCB into functions. Just like how the Minuteman was designed with modules, it would be an asset to the project design if the PCB was sectioned as well. Both the central module PCB and microphone PCB were designed using the KiCad software because of its large component library and its zero monetary cost. These boards were manufactured by JLCPCB with a central module PCB size of 91mm*114mm and the microphone PCB size of 57mm*38mm. The size of both PCBs designs gave ample room for tracing and correct sectioning without being oversized. The central PCB was divided into six sections. Two power sections, one for each integrated circuit, so minimal interference would affect our other sections. The ADC section is split between the analog input of the microphone connector and the digital output going into the microcontroller (uC). The uC section being placed in the middle of the board so that the ADC section and the USB section could be separated from the power sections. The USB section was placed close to the uC, because it requires minimal tracing, corners and vias to mitigate interference with high speed data transfer. The final JTAG and sensor section at the bottom of the PCB. The sectioning of the PCB makes it easy to troubleshoot and solder. When designing a PCB, there are certain standards and choices that need to be made. These choices include a two-layer board that is 1.6 mm thick. The central PCB also is 1 oz copper weight. The weight and the current load change the width of the traces. To calculate the trace width, I used ADVANCED CIRCUITS 4PCB Trace Width Calculator. I entered the values that are for our board and got the trace width for each application. With 32 mils for power and 8 mils for most other connections.

It was important to follow the manufactures limits on bored construction. With 4 mils being the minimum between traces. Trace width was set at 8 mils to lessen interference. Room was not an issue because of the trace width size, distance between traces, and using 603 surface mounted devices

The PCB had three versions. Each improving the boards functionality. With sectioning of the board, troubleshooting was easy, and the board could be fixed with little effort.

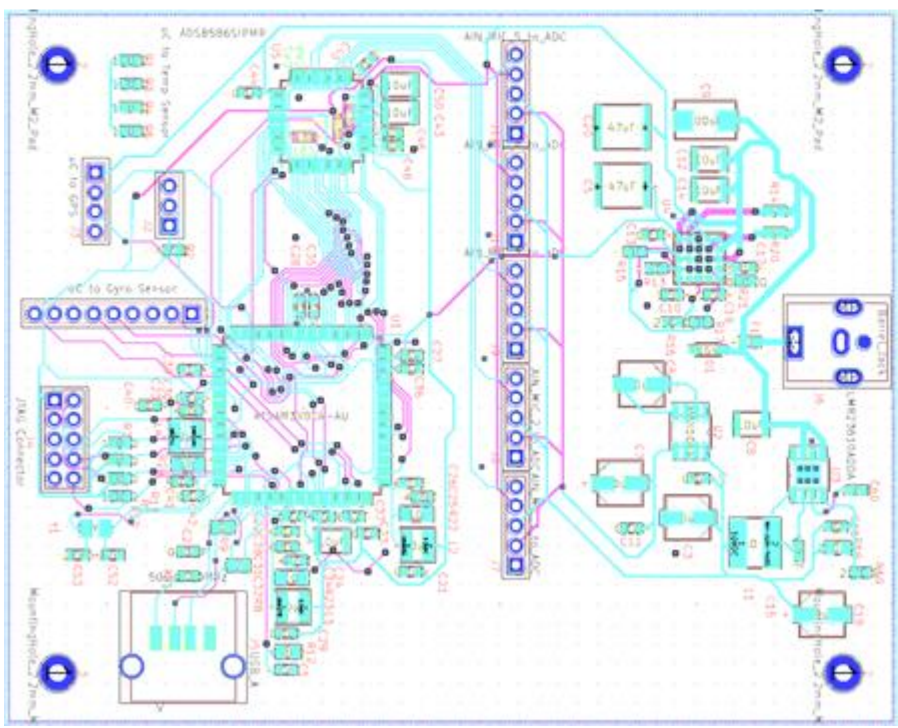

Figure 5-6: Main PCB Board Layout

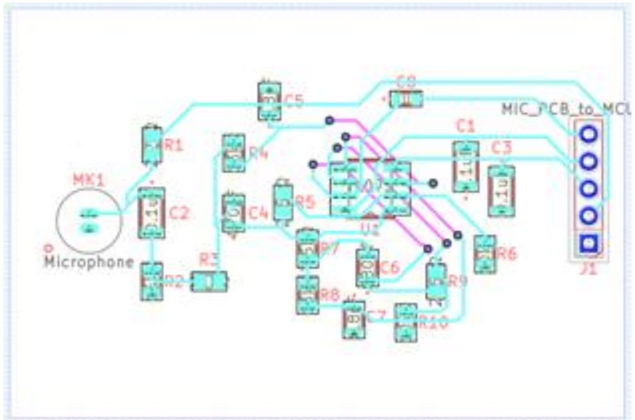

Figure 5-7: Mic PCB Board Layout

5.2.5 Power Supply

The design of a power supply is based on the requirements that is needed to implement Multilateration. This process took time to research. One could take the power supply design and use a simple regulator, but for this project we wanted to pay attention to the development of power supply. The use of Texas Instruments Webench Power Designer program was initially used to help in the development of our board. It helped simulate the different power sources. In this project we compared the use of different sources and the power consumption that was needed.

When the design was considered for this project it was to take the American standard 120VAC at 60 Hz outlet power to a 12VDC adapters. The 120VAC 60 Hz would convert from a wall outlet to a filtered 12VDC. This our supply voltage for the DC to DC buck down voltage regulator. This creates a voltage that is powering our sensor module. In the design there is one central model with 5 microphones in an array around it. It is essential to ensure that the voltage delivered is consistent and with limited noise for good reliability. Through overvoltage and overcurrent sensing, our project has achieved that goal.

The circuit design for this project was done on LTspice. Using this program did help us in development, since it was developed by Analog Devices. This meant that LTspice had many related integrated circuits that were used.

5.2.5.1 Power Requirements

The power requirement is strict and dependent on the load of the circuit design. Each individual integrated circuit does consume a different amount of current and need a specific voltage to work. If these values are not provided then failure will occur in the system. To meet the requirements, it must first be understood what is required. To effectively gauge how much current is needed the whole circuit design was complete. Then the current consumption was considered for each voltage regulator. The range of current consumption for each regulator is within the needed requirements of the circuitr.

5.2.5.2 12V DC Power Adapter Supply

Like stated this is the power supply adaptor that will take the Unite States standard wall outlet of 120VAC at 60 Hz to acquire a steady 12VDC at 2A. This will proceed to the power supply and bucked down with the voltage regulator chosen below. This product was chosen to because of its capabilities and the prices. It was the cheapest selection, which meant buying multiple would not be a problem. For reliability and test assurance three of these devices were bought in case the faults with the initial product or failure later in the Minuteman design cycle.

Input Voltage Range	120VAC 60 Hz
Output Voltage	12VDC -/+5%
Output Voltage	Single
Current Range	2 A
Output Port	2.1 mm
Cost	$4.59

Table 5-13: 12V DC Power Adapter Supply Ideal Specifications

5.2.5.3 TPS62125

This Texas Instrument TPS62125 Buck DC to DC converter is optimized for low power conversion. Still applicable for our project taking in a 12VDC supply and outputting to a range of 1.7VDC to 10VDC. This product is much smaller than the Analog Devices LTM4622A, which would be better for our printed circuit board. This also means it is a much simpler design, but with little to no sensing capabilities compared to the LTM4622A. Another issue that this product has is the low output current. This can be a problem since this product is optimized for small integrated circuit design and not for multiple sensor modules with many integrated circuits drawing power from it.

Input Voltage Range	3VDC - 17VDC
Output Voltage Range	1.2VDC - 10VDC.
Output Voltage	Single

Current Range	0 mA - 300 mA
Output Ripple Voltage	Low
Power efficiency	High
Reliability	Yes

Table 5-14: TPS62125 Ideal Specifications

5.2.5.4 LTM4622A

The LTM4622A Switching Regulator is able to step down a 12VDC supply to a desired 0.3VDC to 6VDC for our microphones and the microcontroller. This DC to DC regulator is a small enough size for our PCB and is able to connect using a Land Grid Array (LGA). Looking through the datasheet, this product has reliability capabilities with overvoltage protection, overcurrent protection, and overtemperature protection. This is a big difference from TPS62125 where it ops for a smaller device size, but with less capabilities. The utility that the LTM4622A is greater and more diverse. The LTM4622A device is also able to output a larger current range from 0 Amps to 2.0 Amps. This will allow us to supply our sensor module, ADC, and microcontroller with as much current as needed. Not only can it supply a large current, but it is able to regulate two different voltages with better filtering capabilities so that the regulated voltage is consistent and within the desired voltage range. This will ultimately fill our requirements for the design of the project.

Input Voltage Range	3.6VDC - 20VDC
Output Voltage Range	0.3VDC - 10VDC.
Current Range	0 A - 2.0 A Dual to single 4 A
Output Voltages	Dual
Output Ripple Voltage	Low
Power efficiency	High
Overvoltage Protection	Yes
Overtemperature Protection	Yes
Power Good Protection	Yes

Table 5-15: LTM4622A Ideal Specifications

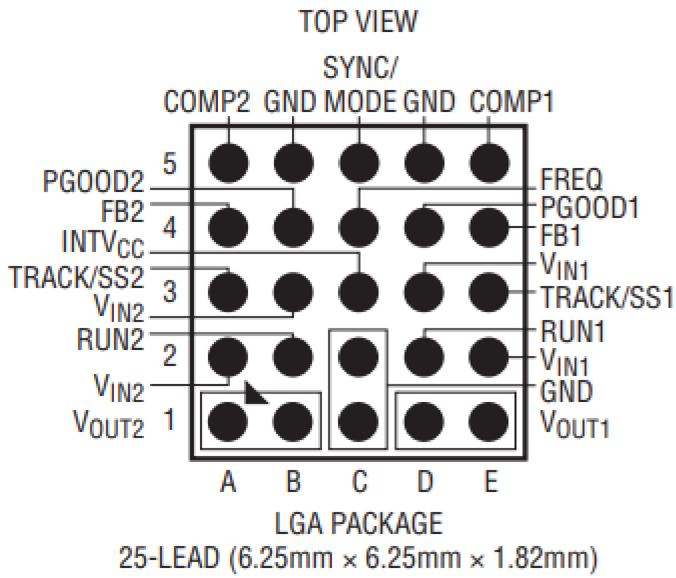

Figure 5-8: Analog Device LTM4622A Pin Layout [30]

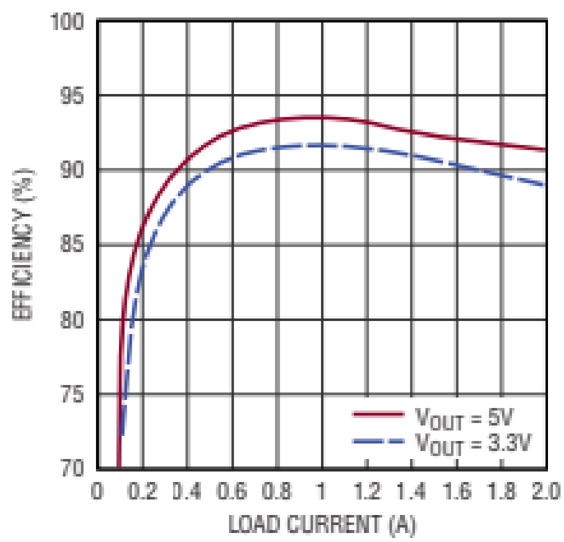

Figure 5-9: Output Efficiency vs Load Current LTM4622A [30]

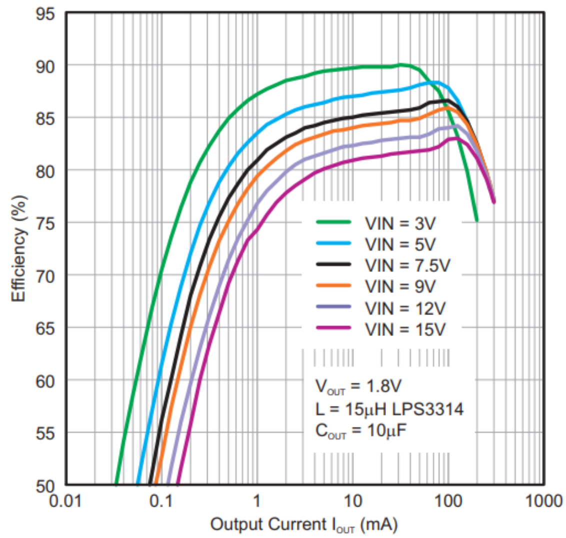

Figure 5-10: Output Efficiency vs Load Current TPS62125 [33]

When comparing these two devices we see that they have many different functions, but some functions are the same. Both having the range of output voltages that we want, but the LTM4622A is able to have dual output voltages. In this case our sensor module will have more than just the microphone so the LTM4622A can supply two different output voltages for different sensors in our sensor module. When comparing the current supplied from both integrated circuits, it is noticeable that the current from the TPS62125 is much smaller than the current supplied by the LTM4622A. The efficiency of the load current is also distinctly different, with LTM4622A peak efficiency at 93% with 12VDC input at 1 A (Figure 5-9) when compared to TPS62125 peak efficiency at 84% with 12VDC input at 130 mA (Figure 5-10).

This means that the LTM4622A can output a higher current with more power efficiency. With the LTM4622A higher current, more power efficiency, and reliability utility it is a much better option for our design. With the great amount of utility that the LTM4622A has when comparing it to the TPS62125, the LTM4622A costs over ten times the price. LTM4622A costs $17.31 in contrast to $1.60 for the TPS62125. The LTM4622A still outshines the TPS62125, even with the cost in mind, which makes the LTM4622A the choice for this project.

In the design of the power supply there are two output voltages. With Vout1 being 5VDC and 200 mA and Vout2 with 3.3VDC and 600 mA. The values can be changed with the FB1 and FB2 pins. By tying an 8.11 kΩ and 13.3 kΩ resistor to ground it create the 5V and 3.3V. The values for the resistors were found in the application section of the datasheet with the table below giving direction of the chosen resistors values.

V_{OUT} (V)	1.5	1.8	2.5	3.3	5.0	8.0	10.0	12.0
R_{FB} (k)	40.2	30.1	19.1	13.3	8.25	4.87	3.83	3.16

Figure 5-11: Power Supply Design LTM4622A FB1 & FB2 Resistor Values [30]

To simulate the current consumption, a load of 25 Ω and 5.5 Ω were use simulate the maximum current draw of 200 mA and 600 mA. Both the voltage output and current draw can be seen in Figure 5-12 and Figure 5-13. Because this is a switching regulator the voltage will switch on and off until it reaches the set voltage by the feedback pins. The simulation was stopped at 5 ms so that the startup sequence could be shown as well as steady state at normal operation. From the transient response shown it can be seen that at 3.5 ms the operation changes from startup to normal operation.

An important part of this project was to get a high efficiency for both of our loads. It is important to do this for our power design. Specifically, the efficiency of the power supply is dependent on the regulator. With the LTM4622A our efficiency with a 12VDC input is 87% for our 5VDC supply and 91% for our 3.3VDC supply. For this project these numbers suffice for our needs. For more power efficiency, burst mode can be used. This allows higher efficiency with lower currents. For this project burst mode was not used.

The use of coupling capacitors helps with filtering out AC voltage from the DC line. This mean that if you have a capacitor tied to ground on a DC line, then as it should any small amount of AC voltage that might interfere with the supplies will be shorted to ground. Nullifying the AC aspect of the voltage and providing noise protection to the voltage supplies. Using multiple capacitors with different Farad values so that the circuit could be coupled at different frequencies. This provides even better immunity to noise, which could cause critical failures in downstream circuit.

This is a switching regulator. So, it must switch on and off for the desired voltage. The FREQ port is the design port for the frequency that the regulator switches on and off. To implement the desired frequency a resistor tied to ground is used. The equation below is used to get the desired frequency by setting the resistor value.

$$f(Hz) = \frac{3.2e11}{324k || R_{FSET}(\Omega)}$$

The resistor value for R_{FSET} for our project is 215K Ω, which gives 2.5 MHz. This was chosen in the midrange of the power supplies capabilities of 800 KHz to 4

MHz. With suggestions from the datasheet a 2 MHz frequency should be used to reduce switching current ripple when the voltage is above 3.3VDC.

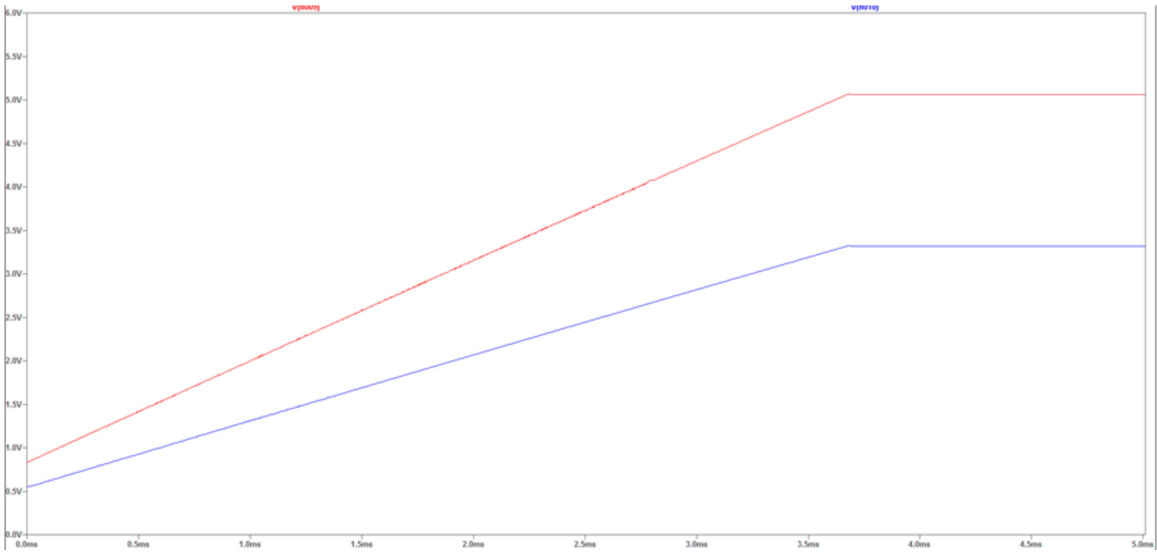

Figure 5-12: Power Supply Design LTM4622A Voltage vs Time

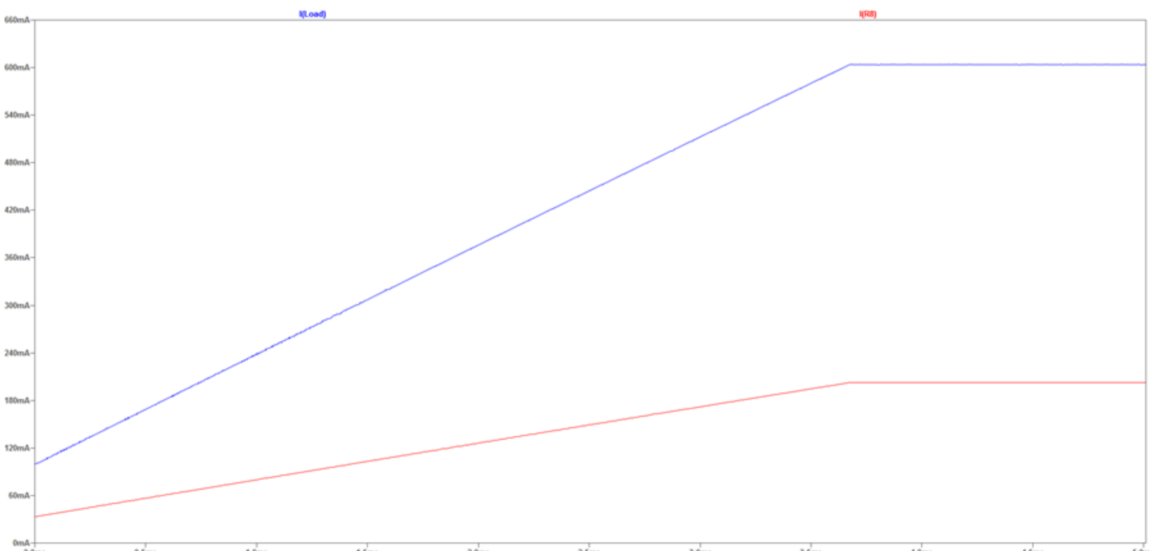

Figure 5-13: Power Supply Design LTM4622A Current vs Time

5.2.5.5 LMR23610ADDAR

The LMR23610ADDAR is a switching buck regulator with a single output voltage at 1 A. This device will take in the 12VDC power supply and convert it to a 10VDC supply. This will be used for the rails on our TL072 op amps so that the microphones millivoltage output can be increased to +/- 5V range for the ADC conversion. More important specification can be found on the datasheet. WEBENCH was used as an initial reference for the design.

Operating Voltage	4VDC to 36VDC
Output Voltage	10VDC
Current Consumption	75 uA
Power Efficiency	95.6%
Dimensions (L x W x H) mm	5.00 x 6.20 x 1.70

Table 5-16: LMR23610ADDAR Ideal Specifications

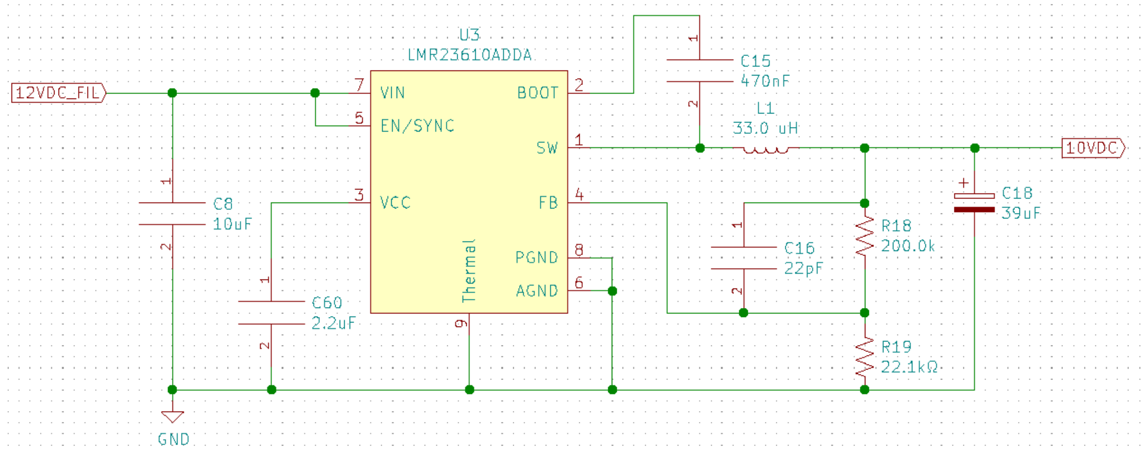

Figure 5-14: LMR23610ADDAR Buck Switching Regulator Design

This part was chosen because of its high-power efficiency and low cost at $3.12.

5.2.5.6 MAX1044/ICL7660

The MAX1044/ICL7660 is a CMOS switched-capacitor voltage converter. It is used to invert the 10VDC power supply. This would give a -10VDC supply, which is used for the rails on the Texas Instruments TL072. There were different methods of doing this, but through searching for the best method this was chosen. Previously this was not needed because a single rail operational amplifier was used, but with new designs this component became necessary. For this project the MAX1044/ICL7660 was used as the basic negative voltage converter as shown in the datasheet. The specification for why this device was chosen can be found in the table below.

Operating Voltage (LV pin open)	3VDC to 10VDC
Output Voltage	-5VDC
Current Consumption (LV pin open)	200 uA
Power Efficiency	98%
Output Efficiency	99.9%
Dimensions (L x W x H) mm	1.930 x 1.930 x .630

Table 5-17: MAX1044 Ideal Specifications

With a high output efficiency of 99.9% this product provided a stable rail for the Texas Instruments TL072. The design of this product is simple and is shown below.

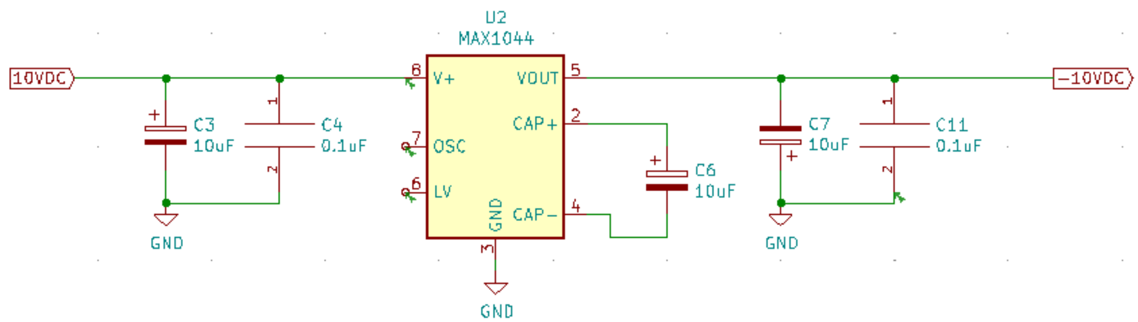

Figure 5-15: MAX1044 CMOS Voltage Converter Design

As stated in the datasheet the LV pin should be open for voltages above 3.5VDC as well as the Boost pin and the OSC pin. These pins should be used for lower voltages. The capacitance values were taken from the application section and are used for coupling. This product was implemented into the PCB board within the power section as shown in the diagram for the PCB layout.

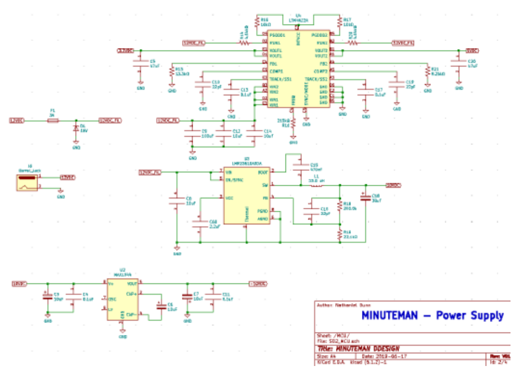

Figure 5-16: Power Schematic Layout

5.2.6 Overvoltage and Overcurrent protection

To make sure the power supply has the best reliability, it was important to add circuit that protects the downstream circuit so minor fluctuations do not cause catastrophic effects to the circuit. One way this was done is adding overvoltage and overcurrent protection to our 12VDC 1 A wall outlet power supply. This can be done in many ways with very complex circuits depending on the results that are required. For this project the design of the overvoltage and overcurrent protection will be simple, since it was used only for the main supply.

To design our overvoltage and overcurrent protection circuit a zener diode and a fuse was used. This design is straight forward, since it only contains two components. The overvoltage and overcurrent protections depends on the ratings chosen for the zener diode and the fuse. Because the overvoltage protection for the main power supply is 12VDC at 1 Amp our design will use a zener diode with a turn on voltage of 16VDC. This means that any overvoltage that goes across the zener will be dissipated. For this design any current above 3 Amp will be considered overcurrent and our fuse is rated to this. The circuit below is the design of the overcurrent and overvoltage protection.

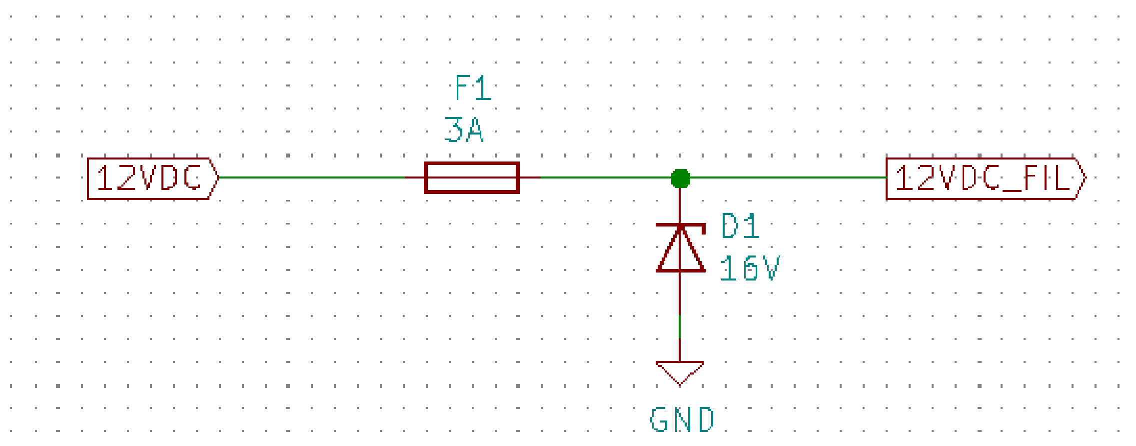

Figure 5-17: Overcurrent and Overvoltage Protection Design

The fuse is in series with the main power source so that if current exceeds 3 Amp, then the fuse can blow to protect downstream circuitry. The zener diode is in parallel to the fuse, so that the circuit can still get the 16VDC and dissipate any voltage that will trigger the turn on voltage of the zener diode. It is also important that the current of the zener diode is within range of the fuse. The fuse that was chosen for the design is SF-0603F125 and the zener diode is the MMSZ5242B_H2.

5.2.6.1 478-2868-1-ND
The SF-0603F125 is a fuse for the overcurrent protection of the power supply. This table below shows the specs for why this component was chosen.

Rate Current	3.00 Amp
Rated Voltage	32VDC
Fast Acting	Yes

Table 5-18: 478-2868-1-ND Ideal Specifications

5.2.6.2 MMSZ5246B-FDICT-ND
Zener diode is used for overvoltage protection for power supply circuit. The table below are the specification of why the part was chosen.

Zener Voltage Range	16VDC
Power Dissipation	500mW

Table 5-19: MMSZ5246B-FDICT-NDIdeal Specifications

5.2.7 Sensor Module
The sensor module for the project is the main component that was used to receive sound input signals, and have the signals sent to the central module. The sensor module utilized multiple hardware components, such as microphones, GPS, a compass, temperature probe from the compass, and amplifiers to power the microphones. The microphones used required to be of decent quality so distinct sounds of gunshots can be compared, but also inexpensive so the project did not go over budget. The research done for finding the right microphone to use for this sensor module is discussed in the hardware initial design section of this project. Two methods of finding the location of the sound from the microphones are discussed in research section of this project, which is triangulation and multilateration.

When this module was initially proposed, we suggested that it should be wireless for this project. This seemed ideal at first and would allow it to be set up without dealing with the hassle of cable management, but since this module is designed as security device to protect people, transmitting any signal wirelessly can be compromised. By using Wi-Fi or any other wireless communication method, this can lead to an exploit where the signal could be jammed by the attacker, or either congestion of the network can cause the transmitted signal sent to the receiver to be significantly delayed. Due to this, the design of the sensor module was changed so the signals would be sent using wires to the central module, which reduces the risk of the wireless transmitter being exploited by an attacker.

The sensor module for this project was initially to be designed such that it would be mounted on a tripod, so it can be set up in a short period of time. Due to time constraints, the module was set up using PVC pipes in a fixed shape instead, as shown in Figure 5-18. Without time constraints, the sensor module would have been mounted on the top of a tripod, which would have had the array design such that it could fold like a tripod stand, where each arm that extends out contains one microphone. When folded out, the microphones would have the same distance between each other by around 50 centimeters and would have one microphone at the top of the main pole the arms are attached to. The sensor module was also designed so it is less than 10 lbs, as using light material for the tripod will reduce cost and make it easier to transport. The PVC pipe used made the weight around 7 lbs for the final design of the array.

A GPS module was also used for the sensor module that was designed. This allowed ease of use when setting up the sensor module, as having no GPS module used would mean manual calculations would need to be done to get the location, which would then be set as constants for the program to run the location algorithm. This means the sensor module can be more portable and be set up in around a minute. A compass is also used alongside the GPS module to give the orientation of the array, so necessary adjustments can be made for the location algorithm. An individual temperature probe was initially going to be installed on the array but was removed as the compass provides the temperature measurement from its built-in probe.

Figure 5-18: Sensor Module

5.2.7.1 Microphone

A microphone is a device that converts a sound input into an electrical signal output. Microphones can generate electrical signals using multiple types of principles but utilize one thing in common: a diaphragm. The diaphragm is a thin piece of material that vibrates when sound waves collide with it. When the diaphragm vibrates, a small electrical current is generated, which is then amplified by the microphone. The types of microphones we considered for this project were dynamic and condenser microphones, as other types were too expensive or not needed for our project. [13]

Dynamic microphones operate by utilizing a coil wrapped around a permanent magnet. When the diaphragm vibrates, the coil would then vibrate over the magnet and generate an electromagnetic field, which would then produce a current output through the wires connected to the coil. Due to its simple design, dynamic microphones are more durable and can handle sound waves with high dB compared to condenser microphones. [13]

Condenser microphones operate by utilizing the diaphragm as the front plate of the capacitor. When the diaphragm vibrates, the distance between the front and back plate changes, which would change the value for its capacitance. When the plates of the capacitor are close, a charge occurs, and when the when they are apart, the charge would then be discharged as current. In order for this microphone to operate, a voltage would need to be supplied to the capacitor. Since condenser microphones utilize the diaphragm as the front plate of the capacitor, they are more sensitive to sound compared to dynamic microphones. [13]

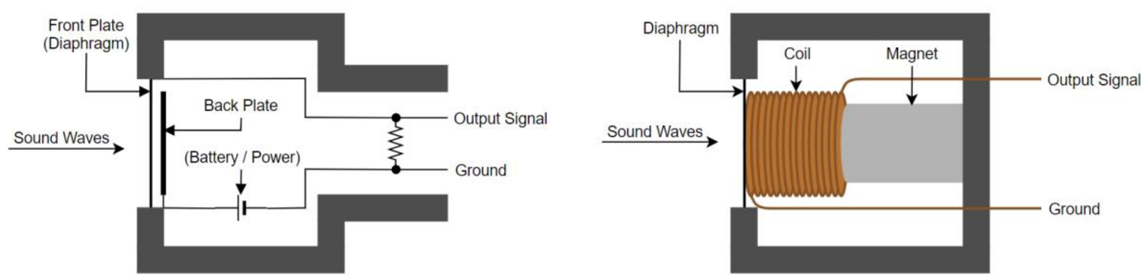

Figure 5-19: Diagram of dynamic (left) and condenser (right) microphone

Microphones are designed with certain patterns for how they pick up sound around them. Cardioid patterns are shaped like a heart and pick up sound around the front of the microphone, but sound outside the heart shape is lower in amplitude. Unidirectional patterns only pick up sound from the front of the microphone, and Omnidirectional patterns pick up sound all around the microphone. For our project, any microphone we chose had to be omnidirectional, since collecting sound from all direction would increase the accuracy of pinpointing the location of a discharged firearm.

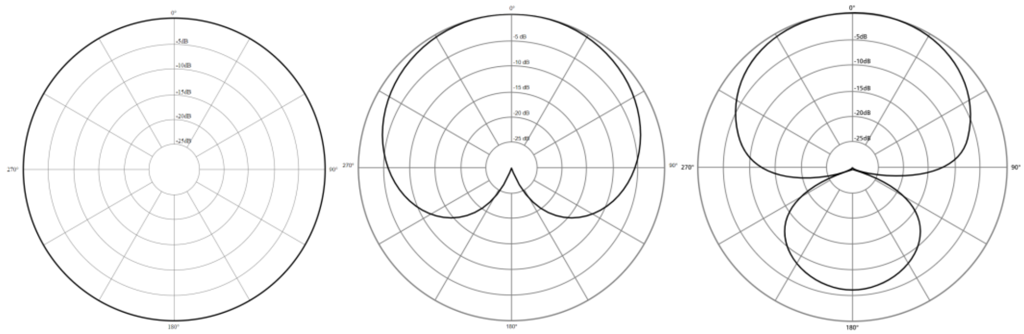

Figure 5-20: Mic polar patterns (Omnidirectional, Cardioid, Supercardioid) [14]

For our project, we needed a microphone that would meet the following requirements, such that it has high sensitivity, is omnidirectional, has a good signal to noise ratio, has a flat frequency response, and is also cheap. The following microphones were considered for our project:

Knowles SPW0690LM4H-1 - This was the first microphone we considered, which uses MEMs technology. This uses an etched diaphragm on a silicon wafer to generate an electrical signal output, which works similarly to a condenser microphone. This microphone was considered initially as it had the following characteristics as shown in Table 5-20.

Directionality (Polar Pattern)	Omnidirectional
SNR (Signal to Noise Ratio)	66.5 dB
Sensitivity	-41 dB
AOP (Acoustic Overhead Point)	135 dB SPL (Sound Pressure Level)
Frequency Range	20 Hz to 10 kHz
Output Type	Digital PDM (Pulse Density Modulation)

Table 5-20: Knowles SPW0690LM4H-1 Specifications

We chose this initially as it already had a built-in analog to digital converter, a high resolution, and a high AOP of 135 dB SPL, which is good since firearms have a decibel level around at least 140 dB, so the microphone should not distort as much as the sound is picked up from a distance. However, we later decided not to use this microphone due to the way it needs to be soldered, and also because we realized its built-in ADC is a constraint. If the ADC ended up needing to be redesigned later in the project, this meant the entire mic would need to be replaced because of it, so we decided to only use an analog mic instead.

CUI CMC-6035-130T - This is an electrostatic condenser microphone that was used for our project. The reasons we chose this is because of its following characteristics as shown in Table 5-21.

Directionality (Polar Pattern)	Omnidirectional
SNR (Signal to Noise Ratio)	70 dB
Sensitivity	-42 dB
High Maximum Input	130 dB SPL
Frequency Range	100 Hz to 20 kHz
Output Type	Analog

Table 5-21: CUI CMC-6035-130T Specifications

The microphone was also designed to be waterproof, which was ideal for our project as the microphones would be exposed to the outside environment, where it will be exposed to rain and moisture in the atmosphere. The frequency response is also ideal for our project, as we will be only using lower frequency ranges, which are within the flat line as shown in the Figure 5-21. Figure 5-22 also shows the diagram of how the power module interfaces with the microphone. A 2.2 kΩ load resistor is added to match the impedance of the microphone, which is 2.2 kΩ. This ensures maximum power transfer for the microphone. A coupling capacitor is added between the microphone and the amplifier, so DC noise from the power module is removed.

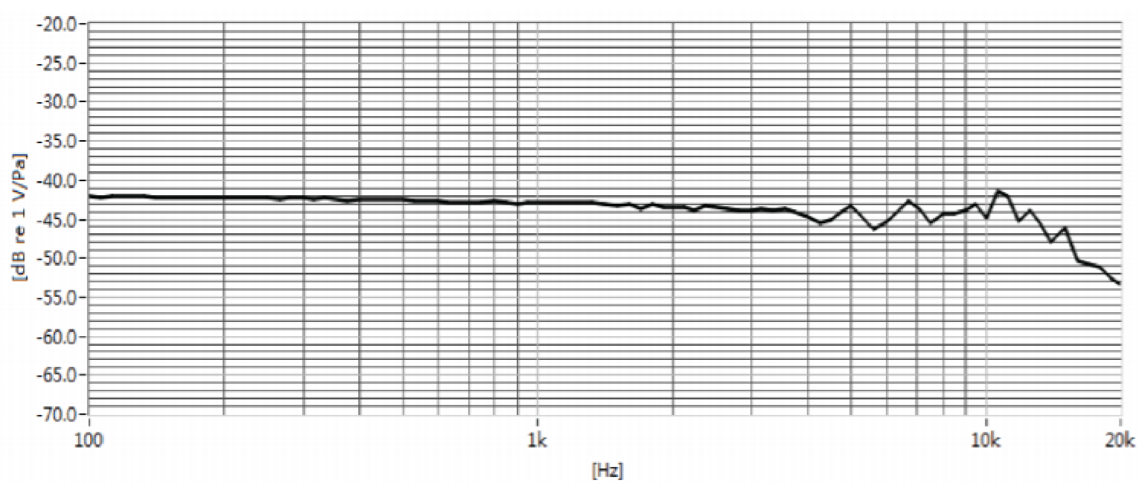

Figure 5-21: Frequency response of CUI CMC-6035-130T [15]

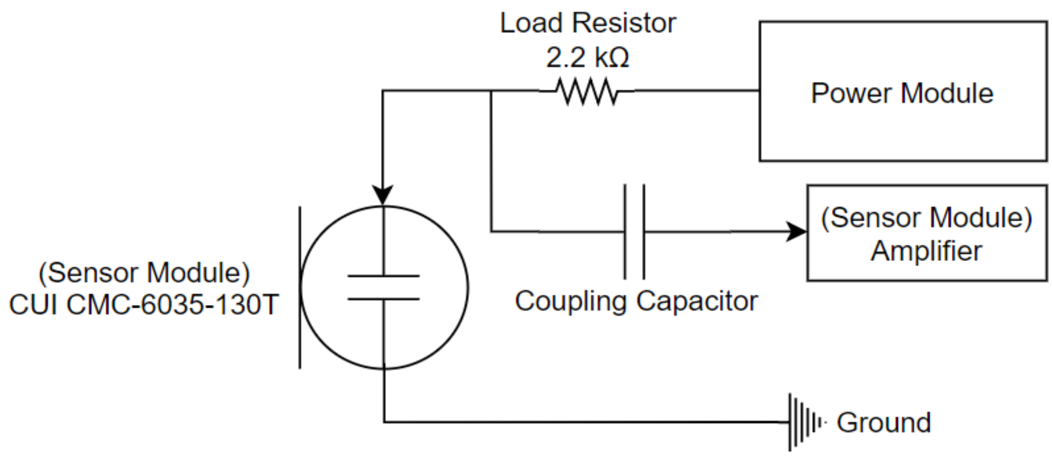

Figure 5-22: Microphone Interface Diagram

5.2.7.2 Thermometer

Knowing the speed of sound changes with the respect of temperature, based on the equation of $C(m/s) = 331.3\sqrt{1 + \frac{T}{273.15\ K}}$, where T is in degrees Celsius, a thermometer is needed to get the temperature of the environment where the microphones are located. This will ensure that the speed of sound calculated would be close to the real value as possible instead of just using the ideal speed of sound of 343 m/s, which could affect the accuracy of pinpointing the location of a discharged firearm. The following temperature probe will be used for our project:

Maxim Integrated DS18B20 - The microphone array used in this project would be also be exposed to the environment, which contains moisture and other elements that can damage electrical components that aren't protected. Due to these conditions, the DFRobot DFR0198 thermocouple thermometer was initially chosen, as it provided a waterproof enclosure for the DS18B20 temperature sensor. This thermocouple was initially chosen as it has the following specifications that are ideal for our project as shown in Table 5-22.

Temperature Accuracy	±0.5℃ for -10℃ to 85℃
Digital Output Resolution Range	9 bits to 12 bits
Temperature Range	-55℃ to 125℃
Query Time	Less Than 750 ms
Voltage Range	3V to 5.5 V

Table 5-22: DS18B20 Specifications

Since the thermometer has a voltage range of 3V to 5.5V, this allows any microcontroller with a 3.3 V and 5 V voltage output to power this thermocouple. Knowing the temperature accuracy is around ±0.5℃, the speed of sound calculated would also be accurate as possible in the environment it will be used in. The query time being less than 750 ms means that it will be able to update it's current temperature more than 1 second, which is more than enough to ensure the speed of sound calculation will be accurate. Figure 5-23 shows how the temperature probe interfaces with the other modules, where the power module supplies 3 V to the sensor, and the data from the sensor is sent to an I^2C port of the microcontroller. A 4.7 kΩ resistor is added between the voltage input and data output as suggested from the specifications of the temperature probe.

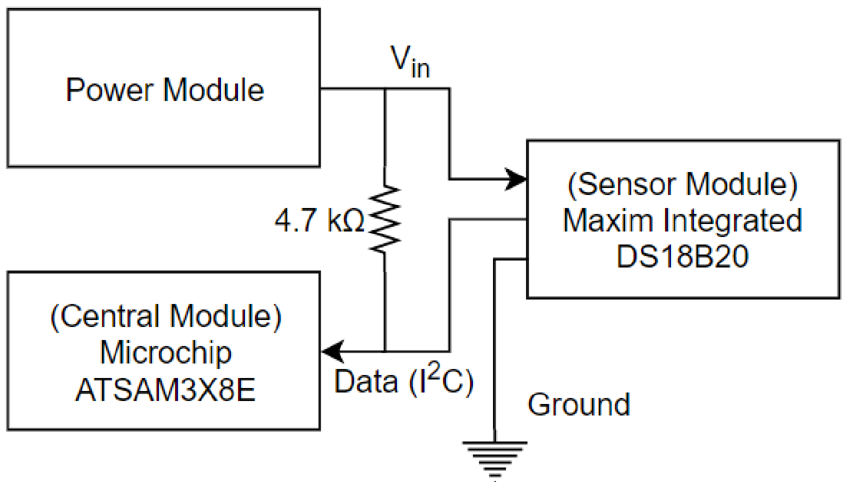

Figure 5-23: Temperature Probe Interface Diagram

Reviewing specifications of other hardware chosen, it was determined that the compass chosen below for the project was found to have a built-in temperature sensor, which had a temperature accuracy of 1℃. The DS18B20 was then rendered obsolete, as it was unnecessary to have the probe used when the compass could already measure temperature.

5.2.7.3 Compass

Knowing the microphones in this project are used to pick up the location of a discharged firearm, that means they will be deployed as an array on a stand, which could have it's orientation at any direction where it's set up. Due to this, a compass would need to be used, as it faces towards the strongest magnetic field. This would face towards the Earth's north magnetic pole, as it will not be exposed to any other strong magnetic fields in our project. Using the compass, it will be able to provide a reference of the orientation of the microphone array which would then be used with the GPS coordinates for triangulating the location of a discharged firearm.

QMC5883L 3-Axis Magnetic Sensor - This magnetic sensor module was considered for our project, as it is compact, provides reasonable accuracy, and has a low power consumption. The QMC5883L magnetic sensor has the following specifications that are ideal for our project as shown in Table 5-23.

Update Rate Range	10 Hz to 200 Hz
Operating Temperature	-40℃ to 85℃
Voltage Range	2.16 V to 3.6 V
Heading Accuracy	1°
Output Type	I^2C Digital Output

Table 5-23: QMC5883L Specifications

The QMC5883L was considered if the array was designed to be stationary, as the update rate selection of 10Hz of this sensor was ideal if it was unnecessary to continuously update when the microphone array isn't going to move at all when it is deployed. The voltage range of 2.16 V to 3.6 V would allow this to be used with a microcontroller that provides 3.3 V and the I^2C digital output provides an easy interface to work with. The heading accuracy of 1° would provide a reasonable offset orientation of the microphone array that is used for pinpointing the location of a discharged firearm.

Bosch BNO055 9-Axis Absolute Orientation Sensor - This sensor was chosen for the project, and is the fusion of an accelerometer, gyroscope, and a magnetometer. This sensor is more expensive than the QMC5883L, however it provides absolute orientation compared to the QMC. The Bosch BNO055 has the following specifications that are ideal for our project as shown in Table 5-24.

Operating Temperature	-40℃ to 85℃
Voltage Range	1.7 V to 3.6 V
Gyroscopic Range	±125 °/s to ±2,000 °/s
Acceleration Range	±2 g or ±4 g or ±8 g or ±16 g
Compass Heading Accuracy	±2.5°
Max Update Frequency	400 Hz
Output Type	I^2C Digital Output

Table 5-24: BNO055 Specifications

This sensor is ideal as it uses Euler angles to provide the pitch, yaw, and roll, which makes the orientation easy to comprehend. The sensor also separates the acceleration due to gravity and acceleration due to movement. The maximum voltage range allows this to easily be used with a microcontroller that provides 3.3 V, the digital output allows easy interface with the microcontroller, and the clock frequency of 400 Hz is useful as having a higher update rate allows the results to be more accurate if the microphone array is in motion. The ±2 g range of the accelerometer is also more than enough to be suitable for a vehicle in motion, if the microphone array is mounted on top. Figure 5-24 shows a diagram of how the compass interacts with other components. The power module supplies around 3 V for the compass, while the data output of the compass is sent to an I^2C port of the microcontroller. The sensor also was determined to contain a built-in temperature probe, which was ideal and reduced the cost of an extra component needed.

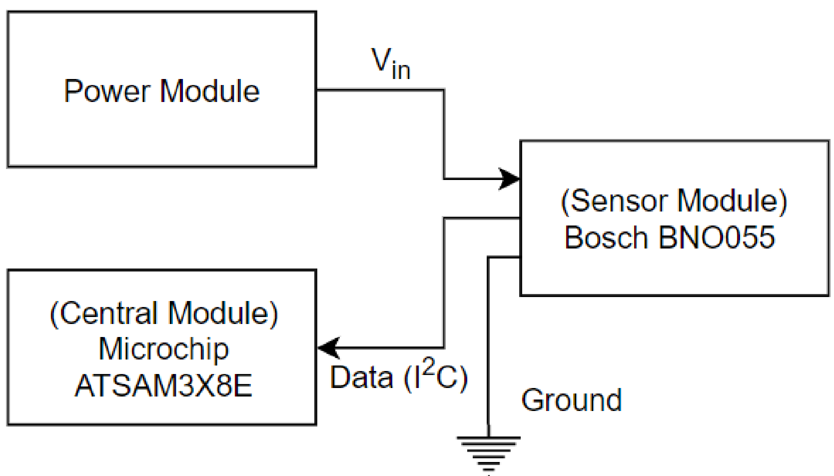

Figure 5-24: Compass Interface Diagram

5.2.7.4 GPS

GPS (Global Positioning System) is necessary for our project, as a reference is needed for the location. A GPS module would be attached to each microphone array, which would also be combined with a compass to provide a reference frame. Using the coordinates from the GPS module for each array, the coordinates can be used to perform the calculations necessary to determine the location of a discharged firearm.

u-blox NEO-6M GPS Module - The DIYmall 6M GPS Module was chosen for this project and uses the u-blox NEO-6M, as it is inexpensive. The GPS module has the following specifications that are ideal for our project as shown in Table 5-25.

Maximum Voltage	3.6 V
Highest Sensitivity	-161 dBM
Cold Start Time	27 seconds
Accuracy	2.5 m

Table 5-25: NEO-6M Specifications

Since the module has a cold start of 27 seconds, this would allow the microphone array to be deployed quickly and have its position accurate less than 1 minute. The maximum voltage of 3.6 V would allow this to be used with the power module that would supply 3 V of power, and the 2.5 m horizontal accuracy of the GPS module would also allow the location of the discharged firearm to be in a tolerable position. Figure 5-25 shows an interface diagram of how the GPS module interacts with the other modules in the project.

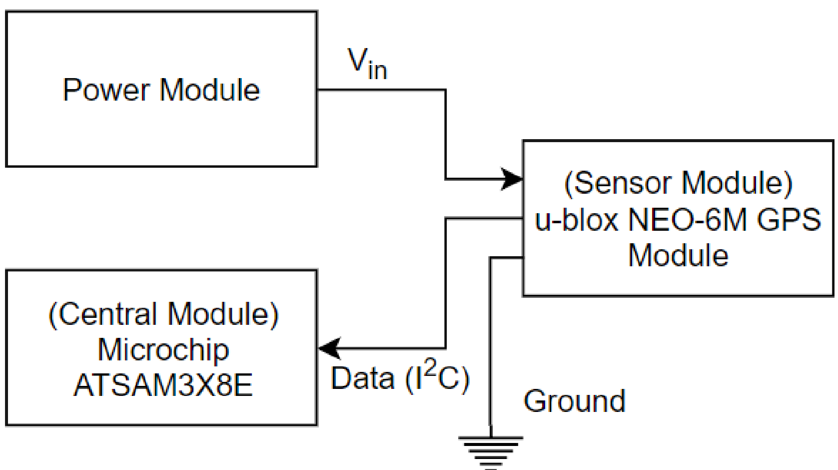

Figure 5-25: GPS Interface Diagram

5.2.7 Analog to Digital Converter

The analog to digital converter, also known as ADC, is a critical component required for this project. The analog to digital converter is to convert an analog input with a continuous time and continuous voltage level to a discrete time and discrete voltage level digital output. The discrete time of the analog to digital converter is known as the sampling period, which is the time difference between two consecutive samples. The frequency of the analog to digital converter would be the inverse of the sampling period, which is also known as the sampling rate.

The performance of an analog to digital converter is determined mainly by the bandwidth of the signal it's converting and the signal to noise ratio (SNR). The bandwidth is the frequency range of the analog input, which then determines the sampling rate that is needed. This is done by using the Nyquist Shannon sampling theorem. The Nyquist Shannon sampling theorem requires the sampling rate to be greater than twice the bandwidth of the analog input for a wave to be reconstructed. For instance, if a frequency is a range of 500 Hz to 4000 Hz, then a sampling rate greater than 7000 samples per second is needed for the minimum sampling rate.

The signal to noise ratio of an ideal analog to digital converter can be determined by the quantization error. The quantization error is the difference between the digital signal level that is closest to analog signal level. The digital signal level is determined by the resolution the analog to digital converter uses. The resolution is the value of 2^N, where N is the number of bits the analog to digital converter can encode. The quantization error then uses the equation as shown below:

$$\text{Quantization Error} = -20\log(2^N)\ \text{dB}$$

Given this equation, the ideal signal to noise ratio can be determined knowing the number of bits used. For example, a 16-bit analog to digital converter has an ideal signal to noise ratio of -96.3 dB using this equation, while an 8-bit analog to digital converter has a signal to noise ratio of -48.2 dB. The example here shows that having a higher resolution is ideal for our project, as having more noise would be less suitable for comparing gunshot samples.

In order to determine the right amount of sampling to use for an analog to digital converter, the time difference between microphones are needed. This was done using a scenario when a gun was discharged 500 meters in front of the first microphone, as shown i. The array for this scenario utilizes multilateration, has five microphones, and each microphone is separated by 50 centimeters from the origin point of the microphone array, as shown in Figure 5-26.

Knowing the microphone A picks up the sound first, the distance from the sound source to it would be 500 meters. The second microphone, microphone D, is chosen, which is 50 centimeters behind from microphone A in the x-axis, and 50 centimeters above microphone A in the y-axis. Knowing the x-axis length, and the y-axis length, the hypotenuse of the triangle can be calculated using the x-axis distance of 500 meters added by 50 centimeters and the y-axis distance of 50 centimeters, which is shown below.

$$Length = \sqrt{500.5^2 + 0.5^2} \cong 500.50025\ m$$

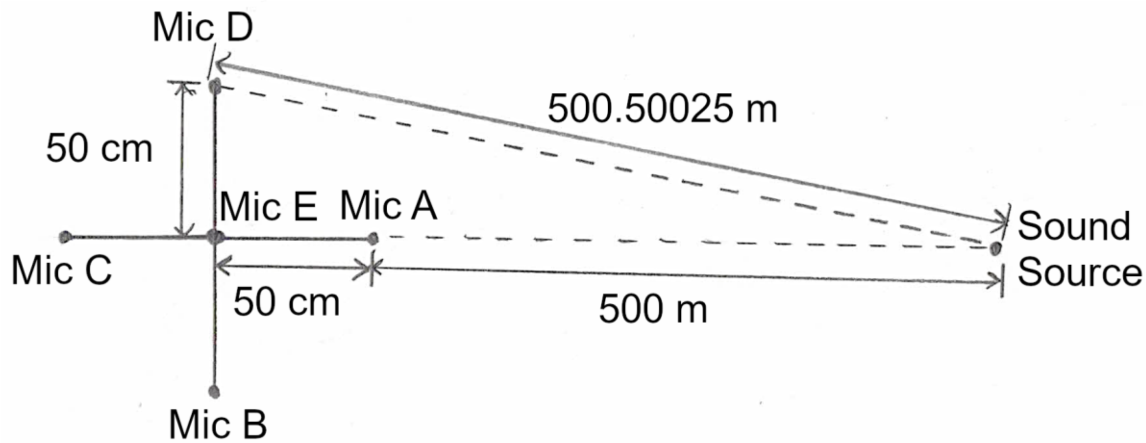

Figure 5-26: Microphone array with a sound source 500 meters in front of mic A

Knowing both distances between microphone A and microphone D, the time can then be calculated by having both lengths divided by the ideal speed of sound constant. The distance of 500 meters would be subtracted from 500.50025 meters and would then be divided by 343 meters per second, which would then equal 1.458 milliseconds. Knowing the time difference, the sampling period can be obtained, which would be the inverse of the time. This would be equal to 686 samples per second

$$\text{Time Difference} = [(500.50025 - 500)\ \text{m}] / 343\ (\text{m/s}) = 1.458\ \text{ms}$$

$$\text{Sampling Rate} = (1 / 1.458\ \text{ms}) = 686\ \text{samples/second}$$

Given the Nyquist Shannon Theorem, the sample rate is the twice frequency range of the analog input for it to be faithfully reproduced. The microphones in the microphone array do not require the entire frequency spectrum, so only a frequency range of around 300 Hz to 5000 Hz would be needed for this project, as only lower frequency sounds are required for comparing gunshots. A high pass filter at 300 Hz is used so low frequency sounds, such as the human voice spectrum can be removed, where voice of men is normally at 85 Hz to 180 Hz and women are normally at 165 Hz to 255 Hz. The low pass filter is located at 5 kHz as when viewing the frequency response of the microphone, the frequency response after 5 kHz becomes more unstable and starts drops off after 10 kHz, where 20 kHz is the half cutoff of the microphone.

Knowing the bandwidth chosen is 4.7 kHz, then the sampling rate must be greater than 9400 samples per second. Knowing the sampling rate is greater than the frequency for the time difference between microphones A and D, a sampling rate of 9500 samples per second would be more than enough to ensure the microphone array calculates the position of sound accurately.

The Arduino Due is a development board that has a microcontroller known as the ATSAM3X8E, which is made by Microchip. This microcontroller has a built-in analog to digital converter that has a 12-bit resolution, which has 4096 voltage levels. This gives a maximum ideal quantization error of -20log(4096), which is -72.3 dB for the signal to noise ratio. However, when looking through the data sheet of the microcontroller, the signal to noise ratio has a range that goes from -60 dB to -73 dB. This is expected as the noise would be higher, especially if noise is generated from other components used as well.

The ATSAM3X8E microcontroller has a theoretical maximum analog to digital sample rate of 1 MHz. This is only for one input, as the total amount of inputs the analog to digital converter of the microcontroller is 16. Given the microphone array has a minimum amount of five microphones, this sample rate would be limited by the time it takes for the microcontroller to have the samples read.

After looking through specs of the ADC of the ATSAM3X8E, it was determined to be unsuitable for the project, as it used a multiplexed data interface. Since the ATSAM3X8E is multiplexed, data would not be able to be sent simultaneously, which leads to issues where samples from microphones would be lost when another is having its data received. The Texas Instruments ADS8586SIPMR ADC was instead chosen for the project, as it has the following specifications as shown in Table 5-26.

Resolution	16 bit
Input Type	Single Ended
Data Interface	Parallel, Serial
Channels	6
Sampling Rate	250 kSPS

Table 5-26: ADS8586SIPMR Specifications

The ADS8586SIPMR has a resolution of 16 bits compared to 12 bits of the ATSAM3X8E, which is useful as it also helps reduce the quantization error. The 6 channels also allow it to work with five microphones of the array, and high sampling rate allows the anti-aliasing filter to be designed better. The high sampling rate is also ideal, as it was later found out that a minimum sampling of 20 kHz would be needed to get the time between microphones. A sampling rate of 44.1 kHz was used for the project, which is a standard sampling rate commonly used for audio recordings. Figure 5-27 Shows the schematic layout of the ADC used for the project.

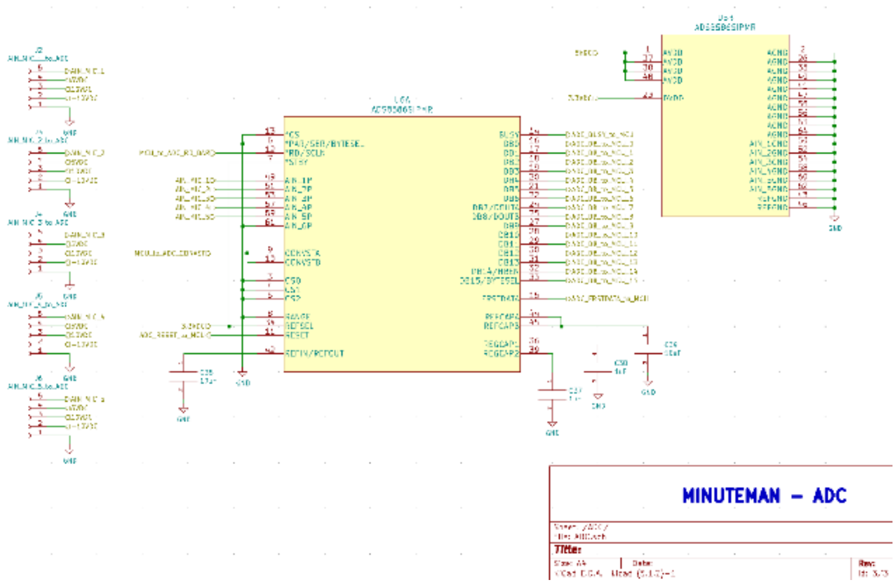

Figure 5-27: ADC Schematic Layout

5.2.8 Amplifier

Since an electrostatic microphone was chosen for the project, an amplifier would be required to increase the gain. The amount of gain needed is determined by the voltage required for the analog to digital converter. For the ATSAM3X8E microcontroller, the voltage range for the conversion from analog to digital is 0 V to 3.3 V or 5 V, depending upon the design chosen. Given a range starting from 0 V for the analog to digital converter, the amplifier would require a DC bias offset in order to keep the input signal from being clipped.

Knowing the sensitivity of the microphone used, the voltage output can be determined. The decibel equation related to voltage is equal to -20log(V). Taking the inverse of the equation, the voltage would be equal to $10^{(-db/20)}$. Knowing the microphone chosen is the CMC-6035-130T, the maximum sensitivity of -39 dB can be plugged into the equation as shown below:

$$V_{out} = 10^{(-39/20)} = \pm 11.2\, mV$$

Knowing the maximum output voltage, the gain can be determined by having the voltage needed for the analog to digital converter divided by the voltage of the microphone. The range chosen for the analog to digital converter is 0 V to 3 V with DC bias, then the waveform voltage range without the bias would be ±1.5 V. Knowing the voltage needed, the gain would need to be around 133.7, which can be rounded up to 140, to make designing the amplifier easier.

Texas Instruments TL084 - When choosing an amplifier, the TL084 is a consideration. The TL084 has been used in labs from previous classes, so given the familiarity with it, it will be easy to use for the project. The TL084 has the following specifications as shown in Table 5-27.

Supply Voltage Range (V_{CC})	±5 V to ±15 V
Maximum Differential Input Voltage (V_{IN})	±30 V
Maximum Input Voltage (V_I)	±15 V
Common Mode Rejection Ratio (dB)	86 dB
Total Harmonic Distortion (%)	Average of 0.003%
Input Noise Voltage Density (nV/√Hz)	Average of 18 nV/√Hz
Slew Rate (V/µs)	Average of 13 V/µs
Input Resistance (Ω)	10^{12} Ω
Number of Amplifiers	4

Table 5-27: TL084 Specifications

The TL084 has a total of four amplifiers, which is useful for this project as it can reduce the space taken on the PCB. Other specs listed, such as the common mode rejection ratio, low noise voltage, and a small harmonic distortion percentage is useful for our project. To simulate the amplifier, National Instruments Multisim application was used. A non-inverting feedback amplifier design was chosen for the simulation, so the general equation is the following:

$$A_V = [(R_2 \div R_1) + 1]$$

Given the equation, the resistors can be picked so a gain of 140 is obtained, such as R_2 being set to 139 kΩ, and R_1 being 1 kΩ. However, by using only one amplifier, noise and distortion would be amplified as well, so the design can be made such that two amplifiers are used to obtain a gain of 140. This can be done by using an amplifier that has a gain of 10, and a second amplifier that has a gain of 14. The equations for solving the gain of both amplifiers are shown below:

First Stage ($A_v = 10$)

$$R_1 = 1\ \text{k}\Omega, \quad R_2 = 9\ \text{k}\Omega, \quad A_v = [(9\ \text{k}\Omega \div 1\ \text{k}\Omega) + 1] \rightarrow A_v = 10$$

Second Stage ($A_v = 14$)

$$R_1 = 1\ \text{k}\Omega, \quad R_2 = 13\ \text{k}\Omega, \quad A_v = [(13\ \text{k}\Omega \div 1\ \text{k}\Omega) + 1] \rightarrow A_v = 14$$

Given the fact that the analog to digital converter requires a range of 0 V to 3 V, a DC offset bias is also needed so the AC signal from the microphone is shifted. If this offset is not used, then half of the waveform from the microphone would be cut off when it goes through the analog to digital converter. This can be supplied by a 3 V, which has its voltage divided by half, which is 1.5 V for the offset. Figure 5-28 shows the simulated circuit of the microphone amplifier using Multisim. Figure 5-29 also shows the gain of the amplifier by comparing an input voltage of ±10 mV and the output voltage, which is around 2.8 V with an offset as shown.

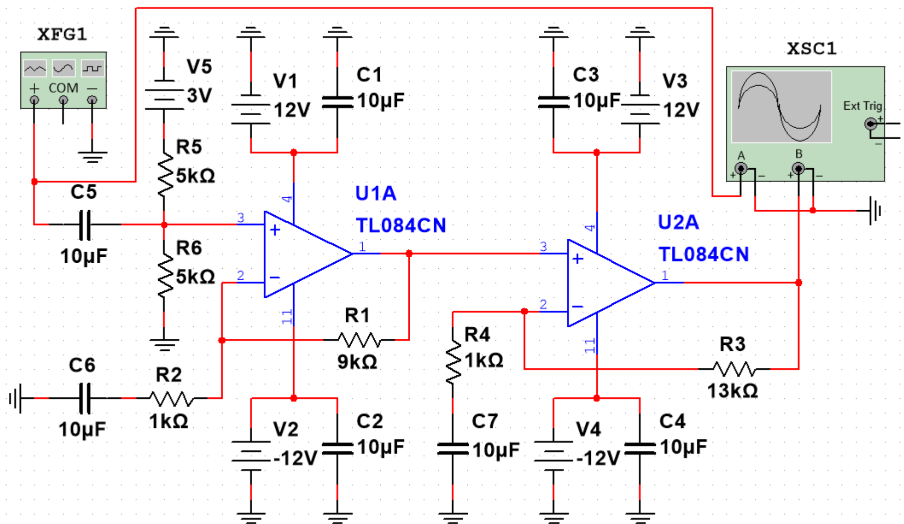

Figure 5-28: Microphone Amplifier Circuit

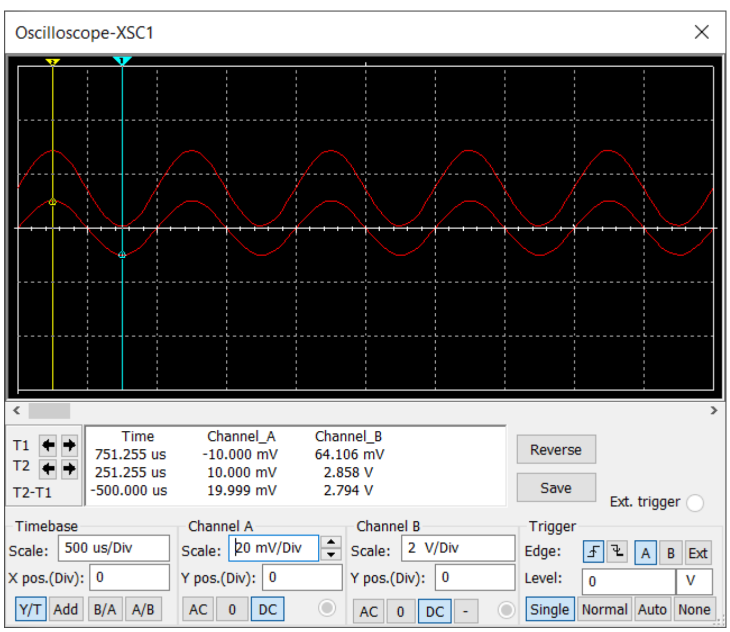

Figure 5-29: Gain of Microphone Amplifier

In Figure 5-28, the capacitor C5 is used to block any DC voltage from the microphone source. Since only two terminals exist for the microphone, the DC voltage from the power module is connected to the output of the microphone. This will cause noise and add an unwanted DC offset if a capacitor is not used. This capacitor is known as a coupling capacitor, which forms a short for AC signals, but is an open loop for DC signals, so the DC signal from the power module is blocked.

Coupling capacitors can also be used with the voltage supply for the amplifiers, as AC noise can be picked up as well. By forming a short as shown for capacitors C1, C2, C3, and C4, the AC signal picked up from the voltage supply would be sent to ground instead of the amplifiers. Figure 5-30 shows the frequency response of the amplifier and the phase as well, which was done using the AC sweep simulation in Multisim. The dB value of the flat line is around 43 dB, so the half power frequency would be 40 dB, which is at 25 Hz. The response starts to smooth out around at 100 Hz.

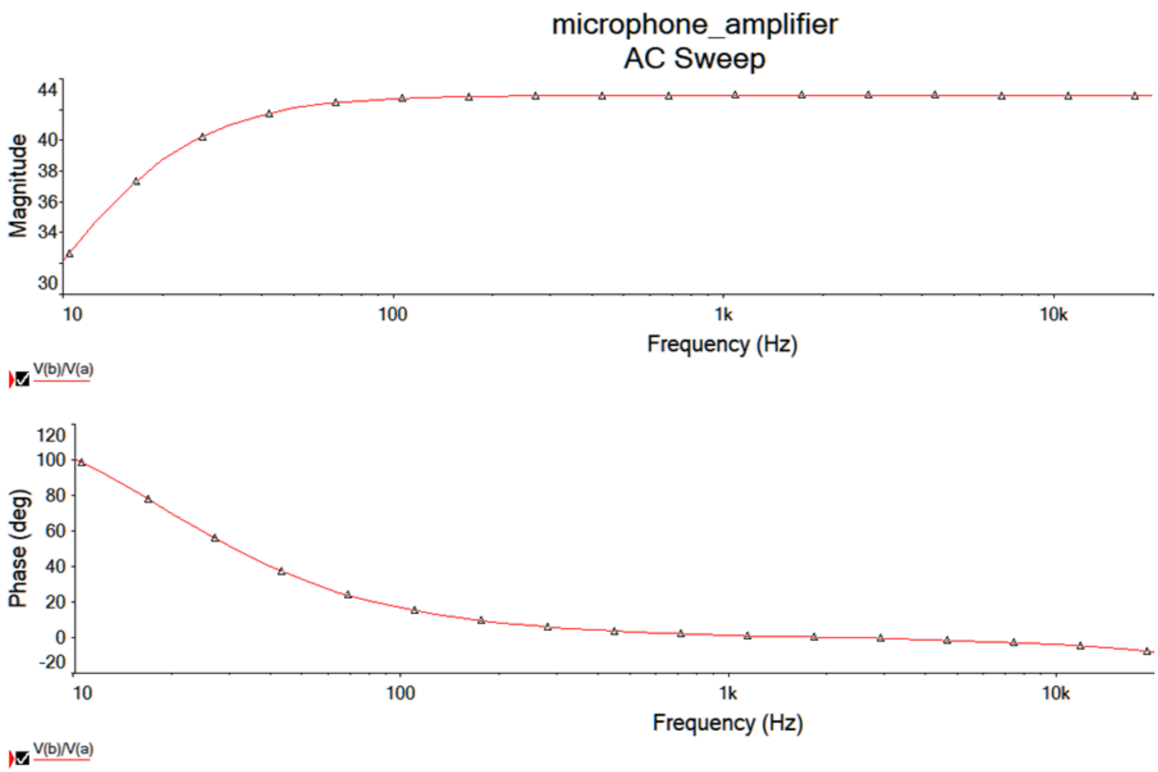

Figure 5-30: Microphone Amplifier Frequency Response

After looking through other amplifiers, the TL084CN might not be ideal. One reason is since it has 4 amplifiers within it. By having multiple microphones outputs connected, this can cause potential noise between them, which is not ideal for our project. Another reason another amplifier is needed is due to the power module design. By selecting an amplifier that uses only a positive voltage input, this would make designing the rails to supply the voltage for the microphones easier, as other devices used do not require negative voltage.

Multiple alternative amplifiers were searched, such as the LM386 operation amplifier made by Texas Instruments. This amplifier was found to have an adjustable gain of 20 to 200 by using a resistor and capacitor in series to adjust it. However, when looking through other specifications, it was found the have a total harmonic distortion of 0.3%, which is much worse than the total harmonic distortion of the TL084 amplifier. The amplifier's design is also old, and dates back from the 1980's, so more suitable amplifiers could be used today that are the same price level and are of higher quality.

Maximum Integrated MAX4466 - The maximum integrated MAX4466 is an audio amplifier designed specifically for electrostatic microphones. This amplifier is used in breakout boards, which amplify the audio of the electrostatic microphone embedded in it, which provides a 3 V to 5 V output depending on how much voltage is added to the input port. This amplifier also has a rail to rail design, and only requires a positive voltage input, which is ideal for the power module design. Table 5-28 shows specifications that are ideal for the project.

Supply Voltage Range (V_{CC})	+2.4 V to +5.5 V
Common Mode Rejection Ratio (dB)	126 dB
Total Harmonic Distortion (%)	Average of 0.03%
Input Noise Voltage Density (nV/√Hz)	Average of 80 nV/√Hz
Slew Rate (V/µs)	Average of 300 mV/µs
Open-Loop Gain (dB)	95 dB (R_L = 10 kΩ)

Table 5-28: MAX4466 Specifications

The amplifier was considered for the project, as it had an ideal gain, which means one stage can be used for the design. Knowing this is used with breakout board designs, this makes it easier to use other designs as a reference for designing the amplifier to be used for the PCB. The MAX4466 is also designed to be used only for one microphone each, so this also removes the issue where the noise between multiple microphones are picked up from each other.

Texas Instruments TL072 - After reviewing the MAX4466, the amplifier was determined to be of a low-quality design. So, after looking through multiple amplifiers available, the TL072 was considered, as it has good specifications like the TL084, as shown in Table 5-29. This amplifier is a dual low-noise JFET operation amplifier, which has a high slew rate, high input impedance, low harmonic distortion percentage of 0.003%, and also has a low noise voltage density.

Supply Voltage Range (V_{CC})	±5 V to ±15 V
Maximum Input Voltage (V_I)	±15 V
Common Mode Rejection Ratio (dB)	Average of 100 dB
Total Harmonic Distortion (%)	Average of 0.003%
Input Noise Voltage Density (nV/√Hz)	Average of 18 nV/√Hz
Slew Rate (V/µs)	Average of 13 V/µs
Input Resistance (Ω)	10^{12} Ω
Amount of Amplifiers	2

Table 5-29: TL072 Specifications

Given this is also a dual operational amplifier, this means that for one microphone, both amplifiers on each TL072 can be used for one microphone, instead of two microphones like the TL084, which could cause interference with both potentially. The operational amplifier however still requires a positive and negative supply voltage, but this was easily solved by using an inverter on the main board. With having two stages, it also allowed a filter to be used along with amplification.

5.2.8.1 Filter

Since Nyquist's sampling theorem needs to be satisfied, a filter is needed so the desired bandwidth is achieved. Initially, this would be a filter of a range of 300 Hz to 5 kHz, however designing a bandpass is not necessary, so only a low pass filter is needed. This filter is also known as an anti-aliasing filter, which is designed to reduce aliasing in analog to digital conversion. An example of aliasing is when a bandwidth of 5 kHz is used, but the frequency is 6 kHz. This frequency would be sampled to 4 kHz, which then causes aliasing, in which high frequencies cannot be distinguished from low frequencies. By using an anti-aliasing filter, the frequencies higher than the cutoff frequency would be attenuated, which reduces the gain and prevents aliasing.

The options for using a low pass filter involve using a passive design or an active design. A passive design just uses a resistor connected to the voltage output and a capacitor after the resistor connected to ground. This has the high frequencies go to ground, so only low frequency signals pass through. For a first order low pass, the equation of $f_c = 2\pi RC$ is used, where given the known cutoff and the known value of the capacitor or resistor chosen, the value of the other part can be determined. However, this is not the case, as when designing a passive low pass filter, input impedance, and output impedance need to be considered, which makes designing passive filters more complicated the higher the order is. By using an active filter, this utilizes an operational amplifier which can be used to provide high input impedance and low output impedance. There are also calculators available online which make designing active low pass filters easier.

Since designing active low pass filters are easier, the filter would be designed this way with an order that is desirable for our project. By using a fourth or fifth order filter, two operational amplifiers can be used, which can also provide gain as well. If fifth order is used, then a first order low pass can be added before the first operational amplifier. Given this, it is possible to design the amplifier and have the frequency cutoff designed on the same PCB used for each microphone. For the gunshots to be compared easily, it will be desirable to have the frequency response flat as possible, so a Butterworth filter is needed for our design, as it fulfills this criterion.

The circuit would be designed using a Sallen-Key configuration, where each stage is a second order low pass filter, which combine would become a fourth order filter. Figure 5-31 shown below shows the frequency response of low pass filters with different orders. A first order low pass has a drop-off of 20 dB/decade, while a fourth order has a drop-off of 80 dB/decade and a fifth order has a drop-off of 100 dB/decade. Given the higher the order, the drop-off becomes sharper and allows more low frequencies to be passed through the ADC compared to a first order filter. A fourth order filter is desirable, since having the drop-off being 80 dB/decade is more than enough to filter out unwanted high frequency waves and allows fewer parts to be used, which simplifies the PCB design compared to a sixth order filter that would take up more space.

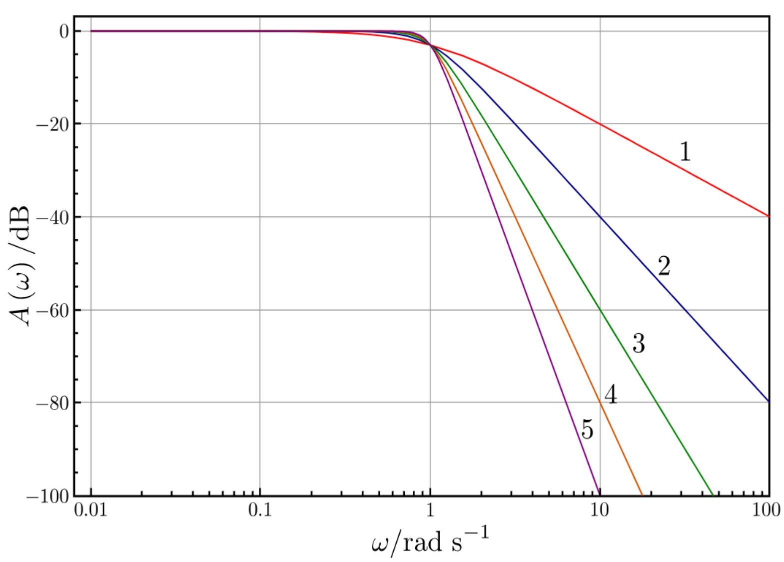

Figure 5-31: Frequency response for low pass filter with different orders

In order to make the filter design easier, an online calculator was used to determine the resistor and capacitor values used [27]. The circuit was designed to be a Butterworth filter such that the passband is flat for the frequency response. The cutoff frequency was chosen to be 7.6 kHz, which is the new bandwidth of the microphone. The gain of the first stage was chosen as 3.73 and the second as 21.5, so a total gain of 80.27 is obtained for the output. The capacitors for the calculator are chosen based on the tolerance set, such as E6 capacitors have a tolerance of 20%, which E12 capacitors has a tolerance of 10%. For the calculator, E6 capacitors were chosen, as they're the most abundant available and would work well enough for the filter design.

Initially for the filter design at the beginning, the gain was around 144. The gain was lowered, as the pressure for the sound going through the microphone caused clipping for the ADC input, so the second stage was lowered in order to compensate for this. The reason this occurred was because the datasheet of the microphone showed the sensitivity only for audio at 1 kHz. Frequencies above 1 kHz can cause a different response, normally higher in value, which was the reason the clipping occurred before the gain adjustment.

Using the given values from the calculator, the filter was then simulated using National Instrument's MultiSim circuit simulator. The circuit is designed using a non-inverting amplifier for each stage. The gain uses the equation of $A_V = 1 + (R_1 \div R_2)$, in which the value of R_1 from the calculator is 205 kΩ and the value of R_2 from the calculator is 10 kΩ, which gives a gain of 21.5 for the first stage. Figure 5-32 shows the fourth order Sallen-Key Butterworth filter, and Figure 5-33 shows the amplification of the output compared to the input. The left side of Figure 5-33 has an input frequency of 1 kHz with a peak amplitude of 10 mV. The right side of Figure 5-33 has an input frequency of 10 kHz, which has an attenuated amplitude which is close to the original input amplitude, showing the filter is working as intended.

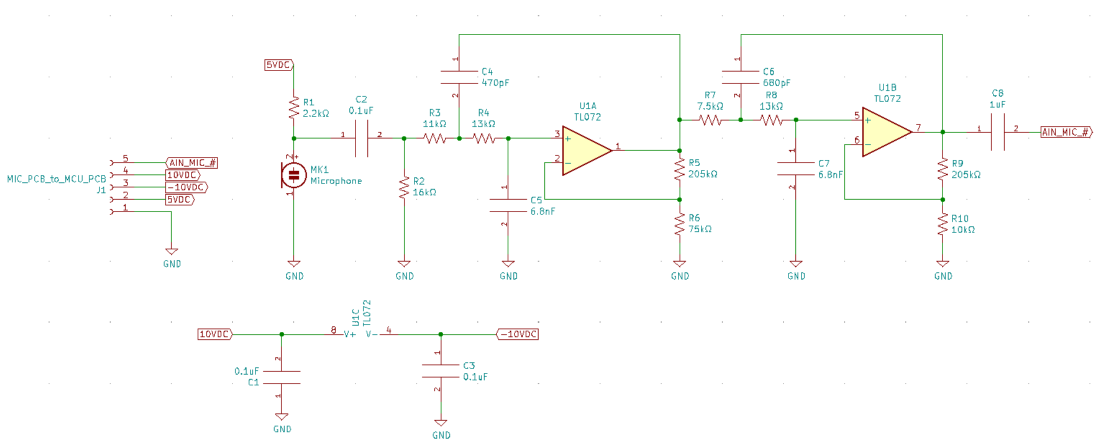

Figure 5-32: Microphone 4th order filter

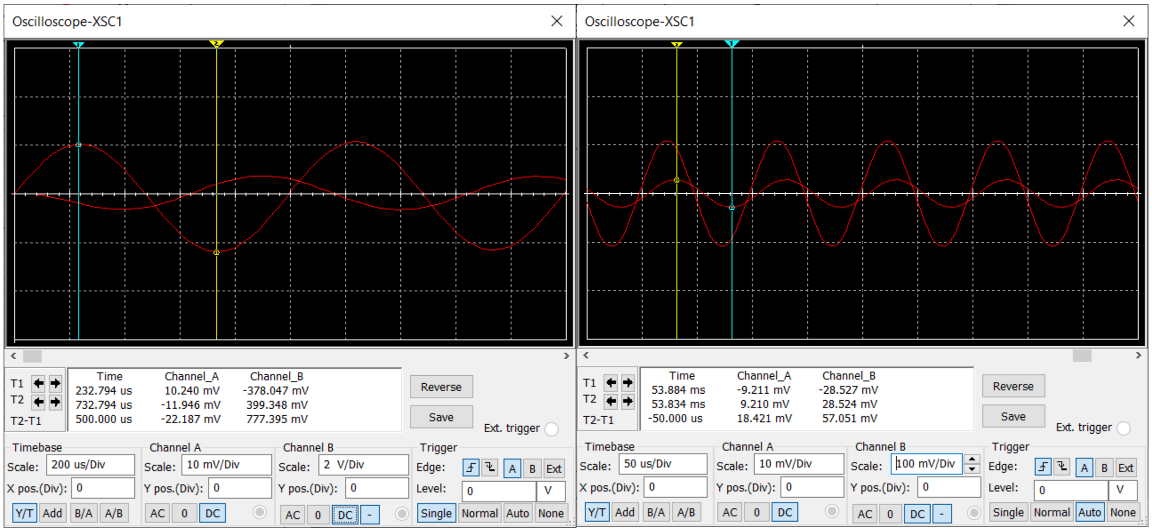

Figure 5-33: Microphone filter amplitude 1 kHz input (left) and 10 kHz input (right)

The filter would be designed to utilize a single supply voltage source that uses a virtual ground. Another either use another design where the input is offset to work only in positive voltage, while the negative supply of the amplifier will be attached to ground. The amplifiers also utilize coupling capacitors like the older circuit design for the amplifier, which is C6 and C7. Resistors of R3, R4, R7, and R8 are used to for the gain of each stage, which supplies 12 for each one, while the remaining resistors and capacitors are used for the second order low pass filter of each stage, which when combined together would form the fourth order filter. Figure 5-34 also shows the frequency response and phase response of the fourth order filter.

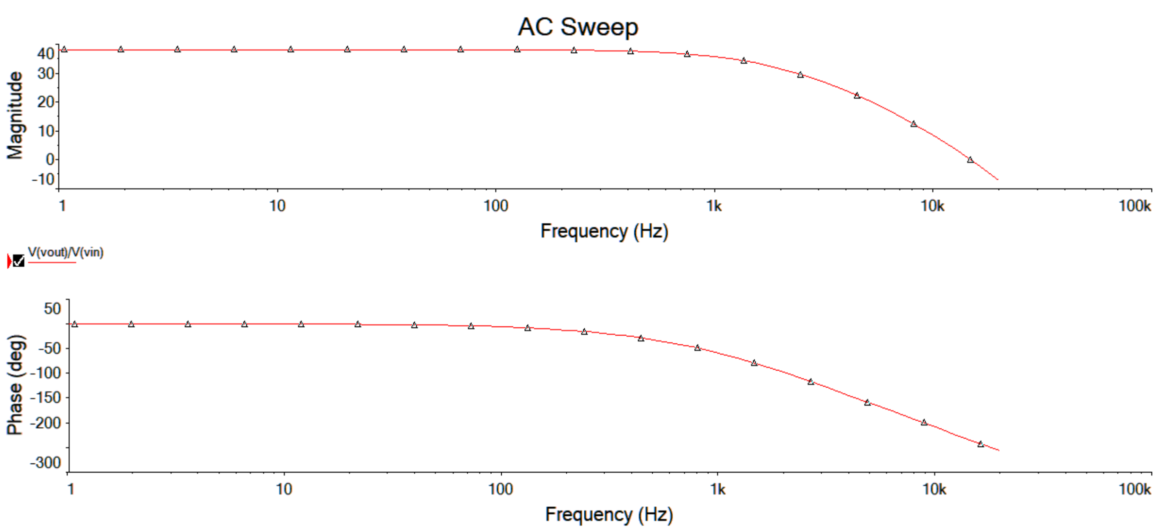

Figure 5-34: Microphone 4th order filter frequency response

5.3 Software

For this project, there were three programming languages used, which are C, C#, and Python, that were very important in programming the microcontroller, designing the Minuteman UI, setting up the Neural Net, etc. Table 5-30 shows the software specifications of the features. Figure 5-35 displays a block diagram of the overall software design.

Feature	**Specifications**
Gunshot Detection	This feature was conducted with Python programming language
Multilateration, Sound Recognition, and GUI	C# language was used in developing code using the Windows OS
Operating System	Windows was used as the primary operating system in this project
USB data transfer	A USB 2.0 was used to be the medium for transmission of data between the hardware and the software.

Table 5-30: Software Specifications

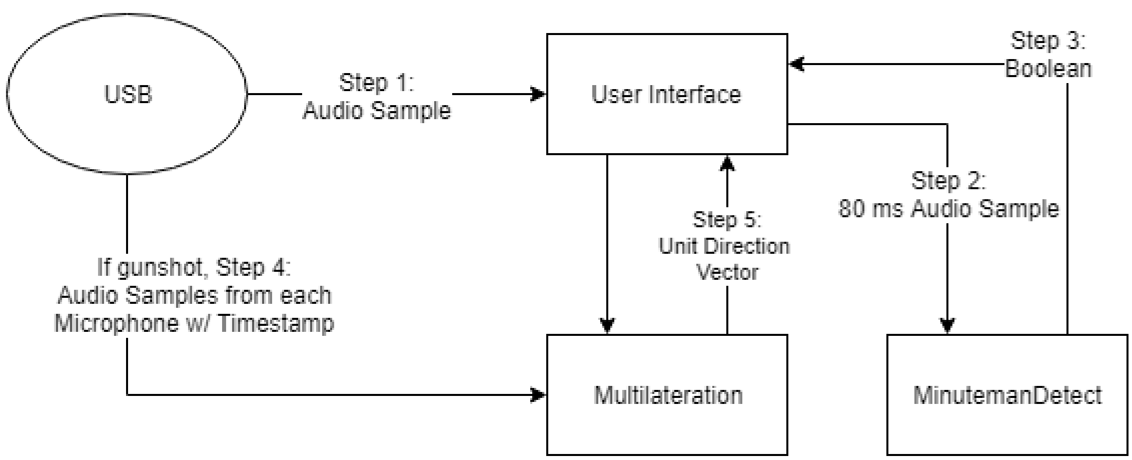

Figure 5-35: A simplified view of the software architecture of project minuteman.

The software aspect may not be included in the physical creation of our project, it still steps in as an integral part to be able to make each piece of the puzzle work or function properly because, without software implementation, the device is basically paper weight. The software and hardware sections for the project vary in difficulty whether it comes to creating the necessary power modules or finding out how to properly manipulate to varying parts or whether it may be figuring out the suitable program to run using Arduino based language into an Arduino Uno board which is also very hefty in detail because setting up and creating a feasible and efficient piece of program takes time to mold into a smooth working process for our audio system to work properly.

While the first set of prototypes or even the initial hardware design is being made, it is imperative, there must be a set in stone decided on algorithms that coincide with the hardware being made and has to be tested in its compatibility with the architecture being built in parallel. The constant modification of the source code goes well with the subsequent constant change in design externally. Something like having an operating system as intimidating as Linux, with software implementation, it is easy for the compiler to transmit some language like c programming into assembly code, Linux will have a hard time figuring out where to place in the memory these custom designs will reside.

It is with intense priority that the software testing must be done at each stage of development for all the hardware parts that is being planned at a specific span of time when implementing the blueprint. One of the most critical points in making software code is its reliability. Without reliability, everything that has been done by the team will be for nothing because if a program crashes and ruins the flow of all the implementation that's been inserted into the sensors, all code must be redone in a careful manner. In the way of testing the code, there will be pre-recorded input samples that will be inserted to the already made algorithms in which we will know if the source code holds up to its known standards. With the built in User Interface created for the ease of the user, it must be handled with care and be tested to see if it can coexist with all the other factors as a union. If one aspect fails, there will be much forethought consequences that, not only the team will face, but will have to resort into restarting a specific field of the creation process. One stage of the development cannot be finalized unless all important source code and software is implemented concretely.

When it comes to transmitting every software fully tested, there will be expected alterations in which there might be required changes by the customer for a better flow of events in regards to the transition of each stage of the process. Our primary objective in this project is to optimize well thought code into our Minuteman architecture, without any errors that may arise. Faster access times is crucial in getting the most optimized final design as possible when the time comes. There will be multiple testing modes when attempting to get ranges and locations for the final product which will consider environmental variables such as surrounding noise and temperature peaks in a specific setting.

5.3.1 Development Tools

Regarding writing software, it’s very critical to take the advantages that every tool offers. We will be using applications such as an IDE or VCS to build our source code for the software implementation part of our project.

An Integrated Development Environment is an application that consists of a simple text/code editor, a code compiler and a handy debugger to debug all the flaws of the source code. Applications like the Arduino software that will be used for the project is built in with Arduino related products. This type of IDE is open source which means that everyone gets a good grasp of what is going on when the code is being inserted into the application and the software development community is free to add in add-ons that can certainly aid in making the process easier for developers.

VCS of Version Control System is a tool that is made by Git. It is a system that manages changes to files over a span of time in that way it is possible to recall previous versions of the project later. With this mechanism, it speeds up the runtime just in case there is a mistake in producing source code where users can backtrack into a version that wasn’t stained yet and refresh because of the notion that there are backups available for use even if the previous versions were used months ago. The use of the version control system improves team collaboration in the sense that people sharing ideas to make the code run more efficiently will come in very handy. The repositories are stored in a cloud system where files can be cloned to their local hub so that the original files will not get touched and will be available to be accessed by other members of the team.

One useful tool when it comes to developing software code is an application called Visual Studio. It is a Windows accompanied software that has many benefits in terms of ease and accessibility for the developers. With Visual Studio Code it can illustrate a specific folder a developer is managing and bring to life by listing down all the various directories and files that may be embedded into a certain project. There is a side tab that shows exactly all the files you need to access when creating your code and you have access into creating new folders and new files that may be necessary in building your program. Another aspect of VSC is the embedded command terminal where you can run JS, Python, C programming codes straight from the application. This is one of the main tools we aspire to use for our Minuteman system because of its simple nature and uncomplicated terms of use. One major factor that separates the VSC program is the git command being included already in the system. With the git command, you can easily push commits into your repository somewhere like Github which is the best in the world when it comes to a stable version control system wherein it detects who is doing what so as to avoid stepping over each other. The user has the choice of staging the code where he/she can set up a new version of the file or revert to the older version. This makes accessibility very easy for our team because everyone knows exactly which aspects of the project gets updates and which still need work.

Another development tool that is vital for our project is the MATLAB application. MATLAB is a useful resource that has the function to design in scientific and computing in engineering aspects. MATLAB can run with C, JAVA, C#, Python, SQL, and many more programming languages out there. MATLAB helps to prototype code into C code for simplicity which can work in real-time operating systems. In relation to our Minuteman project, this is mainly going be used in digital signal processing testing to test different signals for our audio sensors. It allows to be a simulation for adaptive filtering and also be able to act like noise algorithms that can be assured with efficient work and minimal latency. This kind of tool can assist in the process of testing and simulate processes needed by the user where it particularly analyzes the algorithm of noise cancellation sequences before it transmits to the processor itself.

The last candidate for a developmental tool is Kicad. The main function of Kicad is it is used to design a custom PCB that strives with a modular approach that insists in delineating every aspect of the PCB. All users are provided with various number of schematics and this program checked all the schematics for any possible errors that popped up when testing. The Kicad user interface helps everyone in the team in regard to design the circuit board with smooth progress which was then implemented to the project.

5.3.2 Embedded Systems Design

For our Minuteman design, it consists of connected components that work as a unit. We have the power module, the UI module, the sensor system, and the central module which consists of a set of microcontrollers. The custom board that we will be implementing serves as the bridge between the audio sensors and the UI module that our software developers will be working on. It will basically process the necessary data that is accepted through the Audio sensors which will transmit this critical information into the User Interface. There will be a clear signal where the location of the gunshot has been sourced from inside the UI that will alert the user specifically.

To be able to fully access the right data from its buffers, our software has to have clean processing while it is being read from the UI. From the board, it will read the time frame that a gunshot has occurred and, using a specified counting system that will access the max value that each of the microphones pick up during the event.

One of the best ways to reducing the memory time is using multithreading that accompanies the data retrieval and to also reduce the processing stage of the microphones picking up sound from the gunshots. There will be an alternate process in which the buffers be accessed. While there can only be a maximum of two arrays being accessed, one of the processes will wait for the other to retrieve the time frame needed for the data. We plan to halve the tasks between threads so while a specific thread is in the processing stage, the other one will be accessing the sample time frame for the other microphones that need data as well.

For the Audio sensor system, this will be where the sampling and then passed into a conversion function reside. All the microphones that are laid out in an arbitrary location has its output signal relayed into our Field Programmable Gate Array (FPGA) that is inserted into our custom board mechanism.

For the custom board, the microcontroller runs Linux that is put into a mode of low power, which is key to observing one of customer's requirements in power consumption. If the main system has unnecessary power usage, it won't be looked at as much of a profitable product for the consumers. Having low power consumption is much better because it will definitely help users in keeping their usage in check wherever they might be using the device. When the interrupt gets received by the Audio module, the Operating System (Linux) will then fork the threads for algorithms in our software such as the recognition of the gunshots and the algorithm of triangulation which picks up the max value of a signal produced by a weapon.

Next is the User Interface (UI), it is where the data is fed through that gives up an output alerting the user the specifics of the gunshot sounds being heard by the microphone or audio sensors. Bluetooth will come in handy for this module because our Minuteman design is designated for components to be in range of one another and not send long range transmitted signals to even farther distanced locations. The purpose of the User Interface is to filter out duplicate signals, which is then next forwarded to reliable connections afterwards. After that, there will be an alert displaying for the user to check the interface itself for the required data/ information. Upon entering the interface, there will be a designated map that will be seen coming from a node which sent the alert and the location where the gunshot occurred relative to the node. Every event recorded in the UI will have a date and time so it can be sorted out pretty easily later for future use. It is with imperative discussion that the UI be reliable and it be unidirectional wherein changes to the sensors will be made through the system directly.

5.3.3 Operating System

For the creation of this project, we decided on using Windows OS because the team decided on not using the Linux influenced system anymore. Windows OS has the tendency to be the go-to OS system for a lot of software developers out there because of the necessary access to most of these products. Windows is one of the most popular operating systems out there in the world today, so we decided in just using the easiest one for a faster execution. With the exclusion of the mobile application, a Linux influence system is not needed since we decided on just creating a user interface that is connected in a host computer that has a Windows operating system.

5.3.5 Reading Data From USB

For the Neural Network to compare audio samples received from the microphones, data from the ADC needs to be sent from USB to the C# program running the Minuteman UI. The C# extension known as LibUSBDotNet utilizes the LibUSB driver to access the USB CDC interface. The USB is accessed using the known vendor ID and product ID of the USB for the microcontroller. Once the USB is connected from the UI program, a separate thread is called to read data from the USB.

Once the thread is called, a byte is written to the USB of the microcontroller, which would respond and sends the data to the program. The microcontroller would send packets of a maximum of 512 bytes, which would then be parsed through each time when read. The first byte of each packet has a value that defines which microphone the data is being sent from. The value of 128, 64, 32, 16, and 8, represent the microphone index variables of 0 through 4.

When the data is read, its stored in a buffer for each microphone, that has a length of 3528 bytes, which is the size of a 40ms time window for a sample rate of 44.1kHz. Each microphone has a total of 10 buffers, which stores the previous data so the Neural Network can concatenate the current sample and the previous sample to create an 80ms time window for the data read. Once the buffers are full for all microphones, the Neural Network function is called in a separate thread, which then determines if the sound is a gunshot, for the Multilateration function to be called.

5.3.6 Gunshot Classification through Machine Learning

Identifying gunshots is an important task for our software. Once a file is received from the sensor, it must be able to confirm the presence of a gunshot in the sample. While alternative software and hardware options are available for detecting gunshots are available, for this project we will seek a machine learning solution to this aspect of the project.

As discovered in our research, gunshots consist of a muzzle blast sound that radiates from the source, and a shockwave caused by the supersonic projectile. This shockwave produces a sharp, “N” shaped waveform that is depicted in the audio amplitude. However, this is somewhat of a distraction, since the shockwave follows a doppler path along the bullet’s trajectory, the angle of device can vary the time when the wave is received. This means analysis of the sound has to focus on the features of the muzzle blast, and it will be the primary thing to look for with our machine learning algorithm.

Due to the potential for accuracy, a convolutional neural network will be used for detection. It will do binary classification for this task, since we are only looking to detect a gunshot at this time, not specify the weapon. Python offers a wealth of libraries that are useful for generating, training, and utilizing neural networks.

While training a learning algorithm can be time and resource consuming, the final model can be collected from the training as a file containing each layer and their respective weights. This can be extracted and established in the actual program when it is started on the host computer. So long as this model has a sufficiently fast runtime and high classification accuracy, it will work sufficiently. This also means that the network can be easily updated later by simply replacing this file.

5.3.7 Gathering a Dataset

Given the nature of classification through artificial neural networks, and the required accuracy for this project, the convolutional neural network will need to be given supervised training before use. This requires a large dataset of samples that are accurately labeled. The convolutional neural network will be able to train by classifying each sample and comparing the results with the ground truth labels.

Freely available datasets for training are difficult to obtain. This is because the process of manually recording and labelling a sufficiently large enough dataset of samples for the convolutional neural network is expensive in both time and resources. Each sample has to be manually recorded and labelled, and done so correctly so as not to deceive the convolutional neural network. There is at least one, publicly available dataset available on the internet. The Mivia Audio Events Dataset has 6,000 recorded events, consisting of glass breaking, screams, and gunshots. They are recorded at different translations (different angles, distances, elevation), and are recorded with background ambience and sounds that add variation [24].

While such a dataset would be optimal for this project, unfortunately inconsistencies in hardware and file specifications make it difficult to incorporate into the project. Therefore, it was a necessity for this project to collect original samples for use in training. A safe gun range with an appropriate backdrop (a strong physical barrier such as a hill to stop the ammunition) and isolation was selected to fire the weapon and collect audio samples. For the purposes of creating a prototype within time constraints of this project, a .22lr pistol was fired to develop this dataset.

Samples were taken from a number of different positions, varying in angle and distance, with several samples recorded at each position. Additional samples involving loud noises, such as conversation, rocks smashing together, and background noise such as traffic and airplanes were also collected. These were used to provide the network with examples of audio that does not contain a gunshot. After collecting the samples, each of them was manually evaluated and labelled with a ground truth.

Accuracy is important, since detection of gunshots is such an important part of this project. That is why we needed to collect a large number of samples, even if it is for one gun.

5.3.8 Data loading

The process of extracting and preparing data for training and/or analysis is called data-loading. Each sample in the dataset is labeled in their file name. To make sure the entire dataset doesn't consume too much memory, the dataset is divided into batches. There are two sets of batches for the purposes of training and testing. These are separate so that the test data isn't used in testing and so the outcome of the network analyzing these samples is genuine. During either mode, the program will go through each file name, identifying the label of the file, before running it through a transformation function and the network. When the algorithm predicts the outcome of the file, it will compare that with the label to determine an error if any.

Audio files consist of samples, taken over time, that provide a simple number representing the amplitude taken at that sample. The number of samples gathered is determined by the sample rate. This information can be translated into frequency information, and a format that allows for frequency-time analysis in the form of an image.

In this image, each pixel value, similar to a grayscale image, depicts the amplitude. These amplitudes are arranged in windows of frequency, each window corresponding to a certain set in time. As stated in the research, short-time Fourier transform is limited in that it is unable to adapt window size to display the appropriate frequency or time resolution. Instead, however, this can be done with a continuous wavelet transform.

The wavelet transform only requires the data, a range of scales, and a mother wavelet function. The PyWavelet library was used for its wavelet transform function. An example scalogram of a transform of a gunshot wav file is shown in Figure 5-36.

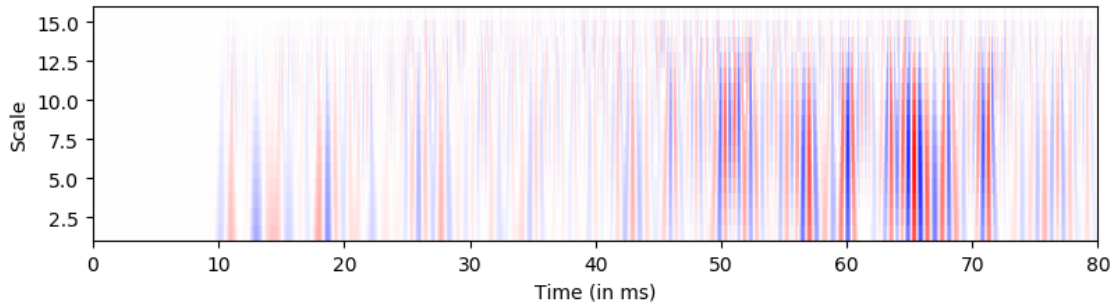

Figure 5-36: Scalogram of a gunshot

5.3.9 Training and Testing the Model

A separate Python program was used to create, train, and test the model. The final program will only be using the final model generated by this program. The training program includes a data loading file, that will extract and load the data for each batch.

The Keras library, utilizing TensorFlow as the backend, was used in training, testing, and then implementation of the neural network. The network is made by creating an object with each layer added to the object. Then the network uses a fit generator, which takes a data generator in the form of the data loader class. The loss function is binary cross-entropy, and the optimization function is Adam set to a learning rate of 0.001. We run the training through 15 epochs, the amount of times, the network is trained over the entire dataset. Once all epochs have been run through, testing is done over the testing data. Which generates a mean accuracy based on loss.

A diagram displaying the architecture of the minuteman neural network is shown in Figure 5-37, and a summary of the layers and parameters is shown in Figure 5-38.

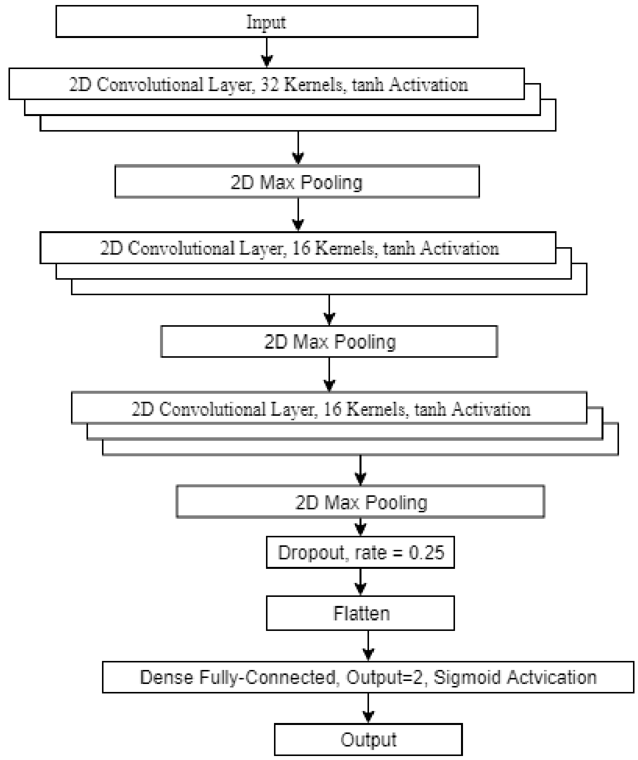

Figure 5-37: Diagram of the Minuteman Convolutional Neural Network architecture.

Layer (type)	Output Shape	Param #
conv2d_1 (Conv2D)	(None, 14, 3527, 8)	40
activation_1 (Activation)	(None, 14, 3527, 8)	0
max_pooling2d_1 (MaxPooling2	(None, 7, 1763, 8)	0
conv2d_2 (Conv2D)	(None, 6, 1762, 16)	528
activation_2 (Activation)	(None, 6, 1762, 16)	0
max_pooling2d_2 (MaxPooling2	(None, 3, 881, 16)	0
conv2d_3 (Conv2D)	(None, 2, 880, 32)	2080
activation_3 (Activation)	(None, 2, 880, 32)	0
max_pooling2d_3 (MaxPooling2	(None, 1, 440, 32)	0
dropout_1 (Dropout)	(None, 1, 440, 32)	0
flatten_1 (Flatten)	(None, 14080)	0
dense_1 (Dense)	(None, 2)	28162
activation_4 (Activation)	(None, 2)	0

Total params: 30,810
Trainable params: 30,810
Non-trainable params: 0

Figure 5-38: A summary overview of the Minuteman Convolutional Neural Network.

5.3.10 Neural Network Implementation

Once the network is trained, the training program saves a .h5 file. The file is then used later in a python program accompanying the main C# Program. The MinutemanDetect.py script is started from the C# program using process commands. Then a named pipe is used to connect to the two programs. The C# program sends an array of bytes, containing the data for 3,528 audio samples (about 80ms of audio) to the python script.

The python script loads the network model first, then it starts to receive information from the pipe. The array of bytes sent is converted into an integer array. Then that array is passed into the continuous wavelet transform. That will generate a 2D array which is the scalogram of the file. A neural network classifies the file, producing either a 0 or 1. If 0 the data is not a gunshot, if 1, the data is a gunshot. An if statement generates a boolean variable based on that answer (false for no gunshot, true for gunshot) which is then sent back to the C# program through the named pipe.

5.3.11 Determining Direction of Gunshots

After an audio sample is successfully classified, the minuteman system must be able to determine the direction of the gunshot with respect to the device. Multilateration used to achieve this. This works by observing features of one recording of the gunshot, and then comparing the difference in time between when these features were recorded by several, spaced microphones.

For performing Multilateration, the Math.NET extension was utilized for C# to perform matrix operations, creating vectors, and a rotation matrix. After a gunshot is recognized, a thread would be created that would perform Multilateration. This function would copy the bytes of the 80ms samples sent to the Neural Network function, along with samples of the other four microphones. A peak detection algorithm is used to get the sampling time distance (TDOA) between the microphones. Once the TDOA is calculated, the TDOA with the smallest value would be chosen as the first microphone, and the second microphone chosen is the next minimum TDOA.

These two microphones are then used the create the system of linear equations derived in the Multilateration research, where the third to fifth microphones are used to get the three equations with three unknown variables. The origin of the microphone array is set as the first microphone with the TDOA value of zero, and the speed of sound is calculated using the measured temperature from the compass. Once the TDOA, position, and propagation of sound is calculated, the matrix equation of $Ax = b$ is performed using QR decomposition, which returns the vector with the x, y, and z coordinates. Before sending the data to the UI, a rotation matrix is created using the Euler angles provided from the compass. The rotation matrix is then multiplied to the vector result of the matrix operation, which gives the direction based on the orientation of the array. The result is then added to a queue which the UI accesses to display the location of the gunshot.

5.3.12 User Interface

While the group had numerous options regarding the user friendly interface that a person is able to handle when running the Minuteman system it was without a doubt we go and connect the hardware into a host computer that the user can interact such as buttons and different features for access such as a create account and a log-in/log-off component so that the user gets their own workspace in the future. Another trait that the UI has is that it’s able to run in the background for convenience as it acquires the necessary information for the user.

There were many options considered in developing the code for the User Interface that will allow for ease for the group and with a certain programming language that the group is already experienced on. There were options such as C, C++, Java, Python, VB.NET, of C#. In the end, we ended up using C# using Windows Forms. We determined it to be a feasible option in creating the interface itself. Visual Studio was a crucial part in gluing all the software together because each component of the software was developed in C# while the neural network was developed in Python script. The integration of the Python script was done through named pipes which means that a program can be in relation with as many processes as possible or different threads. Between the Multilateration code, USB transfer code, Neural Network, it all worked using a pipe system.

As for the main component for the Interface, we decided on going for a Google Maps Satellite View to properly display data transferred into the interface such as the direction of the audio captured by the microphones. Our initial decision was to just display info such as the direction and distance of the gunshots. We scrapped the idea of the gunshots because of the changes in code of the multilateration algorithm. In return, we wanted to properly show the user by a legitimate map where the general direction of the gunshot was in relation to the main system. There is also a table that, in return, displays info such as the timestamp in which it gets the time wherein the gunshot was captured by the system and text that states that if a gunshot is found. There are two buttons in which controls the interface, a Record button and a Stop button. These two buttons control the whole process that starts the different data transmissions throughout the system itself. The record starts while the stop button closes the connection between everything. We decided to have proper displays so that it easy on the eye for the user in getting the results he/she wants. The table on the left of the interface shows vital information that expresses the needed data by the user such as the cartesian coordinates and the timestamp of when it gets captured.

6. Integration and Testing

6.1 Integration

Integration describes how the various components of the project comes together. For hardware, this describes how the various hardware components are connected. For software it describes how various functions and algorithms work together.

6.1.1 Software Integration

The main minuteman program is written in C# and implements several algorithms and functions key to the project. It features the USB class which takes data from the USB and passes it on to the other programs. A user interface to manage the program and present it to the user. A neural network class to start up, send and receive data to and from the MinutemanDetect python script. And a multilateration class to calculate the direction of the gunshot with respect to the device.

Minuteman required drives to implement the USB functionality. It required the MathNet Spatial and Numerics libraries for the multilateration class. The python script required the Keras library with TensorFlow backend, as well as the PyWavelet library for the wavelet transform. All classes are worked on and tested using the Visual Studios IDE.

6.2 Testing

Testing each component of our project is important for ensuring that the entire system works. Due to the nature of the task put upon our design reliability is important. Failure could mean false alarms, misdirecting authorities to the source of a shooting, or worse, failing to identify and alert authorities of the incident at all. Hardware needs to be able to work at expected, and software must be accurate.

6.2.1 Hardware

The testing section is important to be detailed. Knowing that each individual component will have its own individual set of failure modes and failure rates can cause issues for the design of the system and any troubleshooting that will take place. This device is a prototype and proof of concept product. This means it is different in the how it is produced, since it will never reach a full product lifetime. Compared to how products are created in the business world, the proof of concept and design would be just one part of the initial processes in the development stage. Testing the system was completed when both the hardware and software were integrated. The initial testing looked for hardware fidelity and individual component design.

6.2.2 Installation

Each individual component was tested to make sure that they worked before initial integration. The individual component did have a power test, where it was powered with the desired voltage. This type of testing did mitigate any manufacturer faults that could occur in the project. This did help in troubleshooting, since any initial issues would not be from the individual components but caused by faults in the design or integration.

After individual component testing of the hardware, it was important to test the integrated hardware as well. This means, each individual section of the PCB was tested before implementation. The power supply did have all of its parts on a breadboard and then tested to make sure that desired voltages and currents are supplied. It was important to test the sensor modules, so that if any sensors are different from what is expected, it could have been redesigned. The microcontroller section did have similar implementation where after initial fidelity testing, it was supplied with the voltage required and tested to see if any fault in the design occur. Testing the main PCB was important, because the main PCB processes the information that comes from the sensor module and deliver it to the software. By separating this from the software testing, it did provide more knowledge of any faults if it lies within the hardware design.

6.2.2.1 Component

The individual components were tested for any manufacturing errors. The tests did make sure that stated results either in the circuit design or datasheet are the results that are received. The table below shows each of the component test and the results.

Component/Function	Testing	Operating Result
Arduino Due	12VDC adaptor supplied to the Arduino Due input power port.	Operating correctly within datasheet direction and no deviation of desired results.
AT91SAM3X8E	Tested with 3.3VDC & 1.8VDC.	Operating correctly within datasheet direction and no deviation of desired results. Full PCB integration
ADS8586SIPMR	Tested with 3.3VDC & 5VDC.	Operating correctly within datasheet direction and no deviation of desired results. Full PCB integration
CMC-6035-130T	Tested with 5.0VDC & +/- 10VDC.	Operating correctly within datasheet direction and no deviation of desired results. Full PCB integration
BNO055 9 DOF Absolute Orientation IMU Fusion Breakout Board	Tested with 3.3VDC.	Operating correctly within datasheet direction and no deviation of desired results. Full PCB integration

u-blox NEO-6M GPS Modules	Tested with 3.3VDC.	Operating correctly within datasheet direction and no deviation of desired results. Full PCB integration
Power Adapter Supply	The value should be 12VDC	All three adaptors were within 12VDC +/-5% Operating correctly within datasheet direction and no deviation of desired results. Full PCB integration
Analog Device LTM4622A	Tested with 12VDC	All Voltages within 3VDC & 5VDC +/-1% Operating correctly within datasheet direction and no deviation of desired results. Full PCB integration
Texas Instruments LMR23610ADDA	Tested with 12VDC	Voltage within 10VDC +/-3% Operating correctly within datasheet direction and no deviation of desired results. Full PCB integration
Maxim Integrated MAX1044	Tested with 10VDC	Voltage within -10VDC +/-3% Operating correctly within datasheet direction and no deviation of desired results. Full PCB integration
Texas Instruments TL072	Tested with +/-10VDC	Microphones within requires accuracy

		Operating correctly within datasheet direction and no deviation of desired results. Full PCB integration

Table 6-1: Testing table

All hardware in the device are working as directed except the ADC, this was corrected by the software. The ADC issue was a DC offset that most likely came from the wall outlet or interference from the power supply. To avoid this interference, the PCB should have separated grounding plans like illustrated in the ADC datasheet. Bypass capacitors should have also been placed on the front without any vias connecting them. Other than what is stated there was no other hardware choice that were unsolved.

6.2.2 Software

Testing the software was admittedly a very difficult task. The program consisted of two primary functionalities; the neural network-based detection system, and the multilateration algorithm. There is also the graphical UI, and a program to handle data incoming from the USB. The software was accessible and useable on reasonably capable desktops or laptops.

6.2.2.1 Installation

Installation was an important part of our implementation. Our initial design wanted to be an easy to install program. While a simple installation process hasn't been implemented yet, we still have a plan for installation testing.

1. The user or individual in charge of setting it up should be able to easily download the software installer from the internet.
2. Upon execution, the installer will present the user with the option to choose the installation path, recommending a fast hard-drive or solid-state drive.
3. Upon clicking "Next", the software should begin installation automatically.
4. Once installation is done, a final screen should notify the user that the program is installed and offer to open the program upon click "finish".
5. Clicking "finish" should close the installer.

To test this, several computers, ranging within a set of performance requirements, will download and install the software through the steps stated above.

6.2.2.2 Usage security

To prevent unwanted individuals from accessing and tampering with the system, the user is asked to create an account upon the first execution of the program. They are then required to log-in from then on. For account creation we had the following requirements.

1. The user is directed to an account creation page on the program.
2. They enter a valid username and password.
3. Upon entering, the program should validate that the account was created for the user, before proceeding to the control panel.

This was tested upon start of the main program from the IDE. It prevented access to the program until an account was created. It stored a username and password and it validated that an account was created once that entered.

6.2.2.3 Logging into an account

Once an account is created and/or once the user has logged out of the system, they need to log back in through a login page before getting access to the program.

1. The user is presented with a username and password field.
2. The user is redirected to the control panel if they enter the correct credentials.
3. The user is denied if they enter the wrong credentials.

Like account creation this was tested from the IDE upon start of the program for regular testing. If there is already an account, entering the username and password for the account allows for successful entry into the program.

6.2.2.4 Logging out

The program automatically logs out upon closing the control panel. This was tested by closing the program and re-opening it. Each time the program requested a login as expected.

6.2.2.5 Connecting to the sensors

If the system is connected to the computer, the program is able to automatically connect to the USB connection. No user involvement was necessary to test this, as it was recorded as coming online so long as power was supplied to the hardware, and the program was there to receive it.

6.2.2.6 Detection

Detection is a major functionality of this program. The USB software passes data in 80ms chunks from one of the microphones to a python script, which reads the data, transforms it, and classifies if it is a gunshot. If it is, then the data from all five microphone sensors is passed to the multilateration and that is allowed to run before detection continues. The following a list of how the detection script runs.

- The program receives an audio file from the USB software.
- Upon receiving a sample, the script transforms it using a wavelet transform.
- The program passes the new data through the artificial neural network, which classifies if it is a gunshot.
  - If it is not a gunshot, the script returns a boolean set to false to the main program.

    - If it is a gunshot, the script returns a boolean set to true to the main program.

This is tested by activating the “record” function on the GUI, which establishes a connection with the python script and starts to analyze the incoming data from the device. Console printing is used to confirm the accuracy of data being sent from one end to the other. Another print to display the transformed data after running it through the wavelet transform. And a final print to confirm that neural network’s prediction lines up with the returned Boolean sent to the main program. Console printing is removed after sufficient testing.

6.2.2.7 Displaying in Gunshot Detection

A key function of the GUI is displaying the direction of a gunshot and additional data to the user once the device has detected a gunshot and calculated the direction. The following describes what GUI is to do.

- Once the record button is pressed, all the main functionality of the software is activated.
- Once a unit vector is received from the multilateration algorithm, the GUI displays the direction of the gunshot with respect to the device in the form of an arrow pointing in the gunshot’s direction.
- Threading is used in acquiring data from the multilateration algorithm wherein it updates the User Interface whenever a gunshot is detected. The text box displays the timestamp in which the gunshot is picked up upon with the cartesian coordinates.
- With the textbox right at the left, it will constantly update when the whole system is running until the user stops the program itself.

Figure 6-1 illustrates the home page for the whole Minuteman UI where the user gets prompted to create an account or log in.

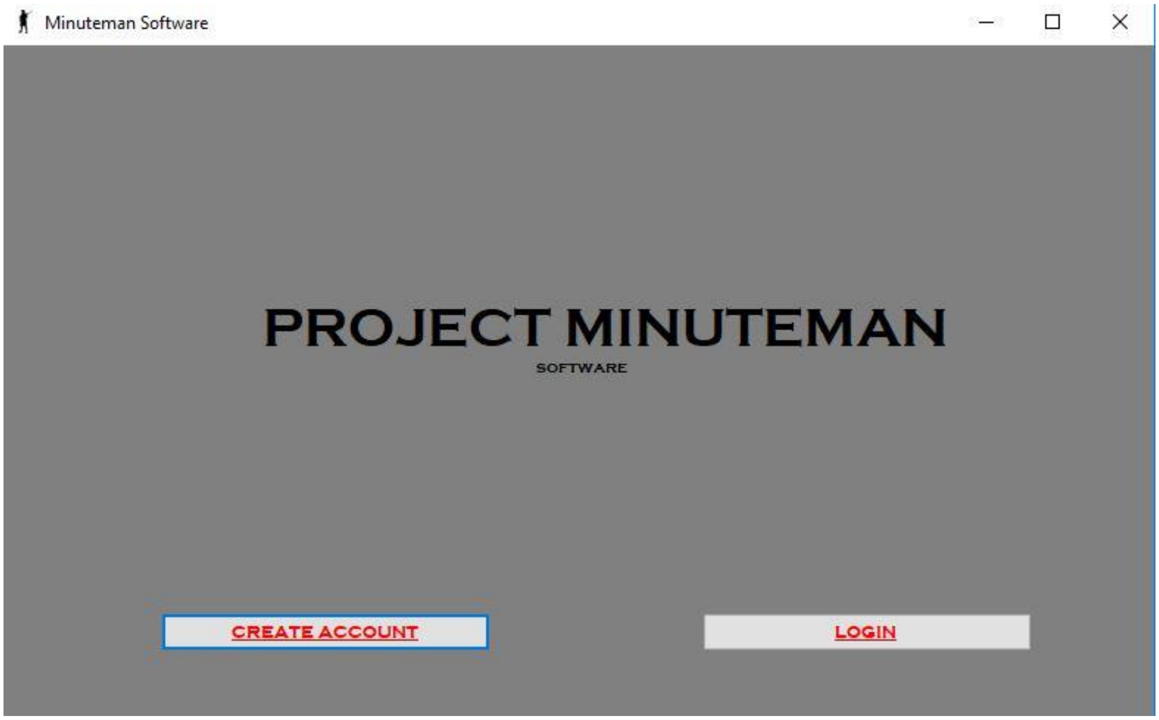

Figure 6-1 Main User Interface Home

Figure 6-2 illustrates where the user gets to implement his username and password once he/she gets to create an account.

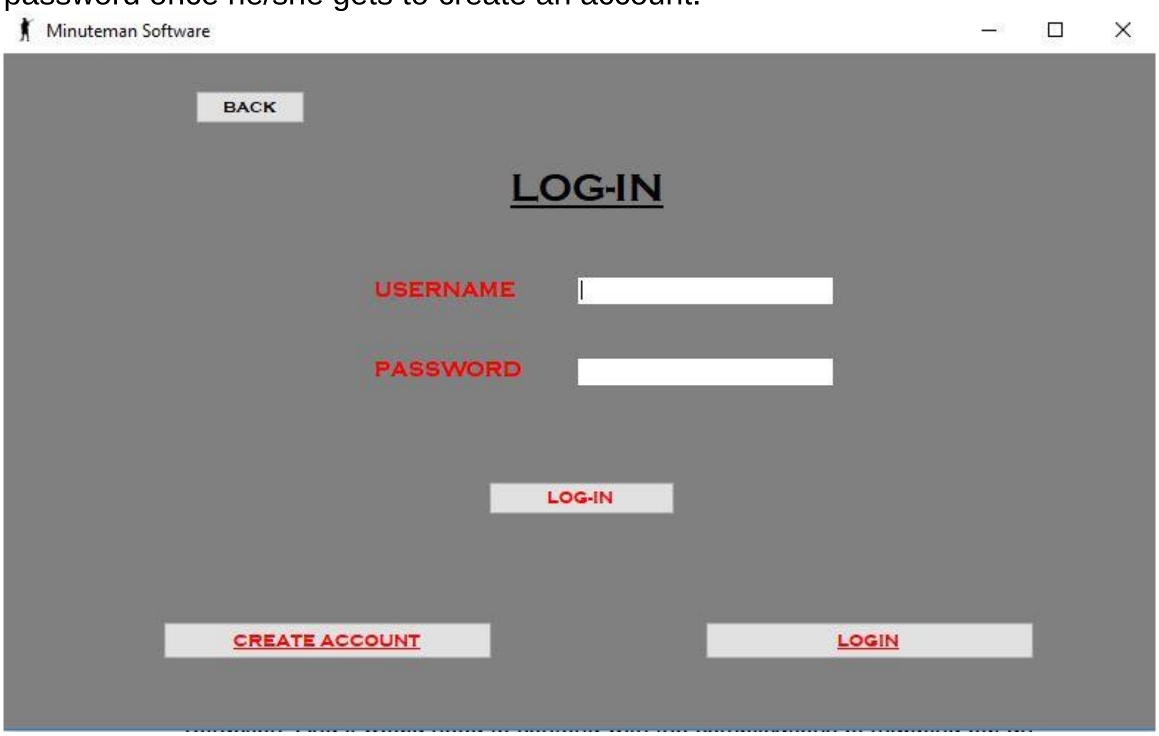

Figure 6-2 below illustrates what the Minuteman graphical interface looks when the main program is running. This is tested by starting the program, logging in, and then interacting with the interface to activate the program. Then audio is fed into the device to pick up an event and display the direction from which it originates.

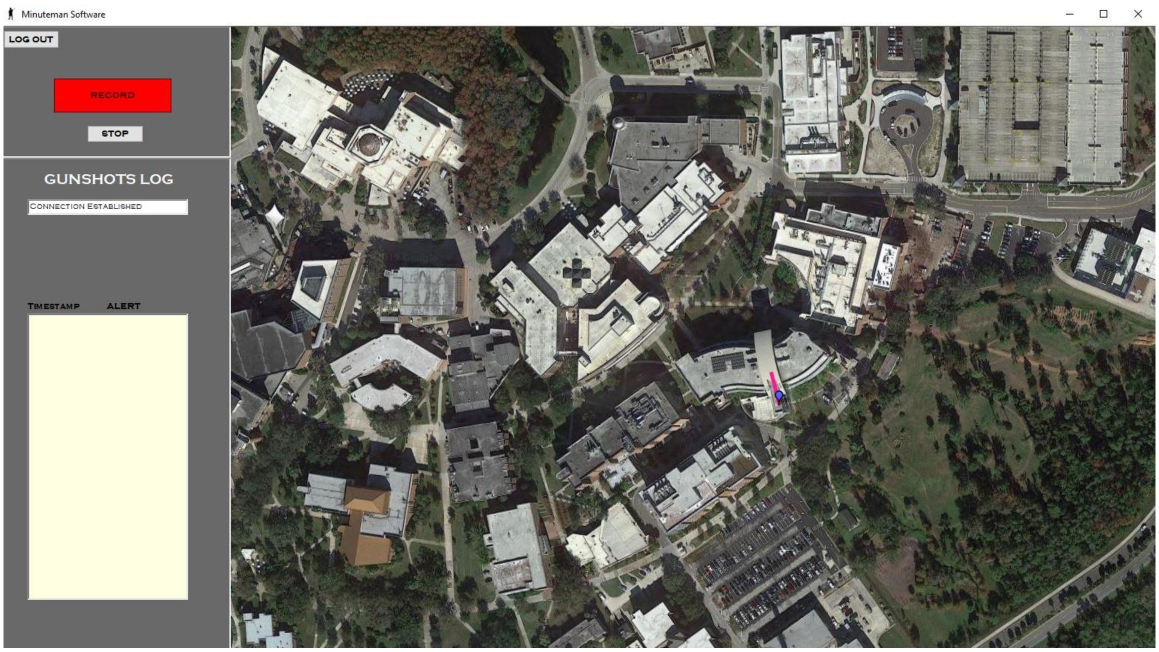

Figure 6-2: Minuteman User Interface

7. Usage Guide

The software suite must be downloaded from our project page and unzipped to a common directory, and both Python 3.7 and the most current .NET Framework must be installed on the host computer. Set the body of the device in an unobstructed location or mount to maintain its best performance.

After plugging in the Arduino and main PCB's USB cables to the host computer, allow the hardware 30 seconds to initialize. Then open the executable in the installation directory and you will see the surrounding area visible through satellite display. The controls will be presented on your left - press "Record" to begin tracking nearby gunshots.

While recording, all detected gunshots will be analyzed and their direction of origin will be displayed textually, on the left, and visually, on the map to the right. Pressing

“Stop” will end the capture session and clear the view for future use. Closing the program at any time will end all data capture and close the companion application.

8. Conclusion

Research and development of this project has drastically improved our understanding of the concepts and functionality of this project. Our original plan for this project involved creating an indoor gunshot detection system, utilizing triangularization with multiple connected sensors to locate a gunshot. While this was realized with systems like SenseShot, it came to our understanding that an indoor system was not realistically within the limitations of our capabilities. Such a system would need to compensate for the obstructions within the interior of the building. It would likely have issues locating events accurately with audio refraction. And it would need to contend with the complications of mapping out an internal space to coordinate locating the event. All of which would require many sensors, good device communication, and would likely cost more than other approaches.

It was the team’s decision to create an outdoor system designed primarily around one device rather than a network of devices or sensors. It could be mounted onto a building or vehicle like the Boomerang system and can be far more affordable and accurate than an indoor system. It also comes with the added benefit that an outdoor detector can easily point to a building if a shooting is taking place within.

Our research into multilateration, and into the Raytheon Boomerang III system, also revealed that locating the shooter, and identifying what firearm was being used, did not require a large number of sensors. Instead, all that was required was one device within a specific area of operation. It was then necessary to add a GPS module, along with a compass for the purposes of the multilateration program and for displaying the direction of gunshots with respect to the device.

While not conventional, machine learning was selected as our primary detection method. Originally it involved several steps. The first was where the peak in an audio sample was measured and ignored if it did not exceed a minimum threshold. While it is true that guns can generate audio exceeding 130dB, this can drastically lower over certain distances, especially for smaller caliber ammunition such as .22 long rifle.

There was a second layer involving a machine learning detection system that was supported by the third layer which attempted to analyze and detect the distinct shockwave of the bullet as it travelled through the air. However, the latter was deemed too complicated and inconsistent, since the listener will always receive the shockwave at different times depending on their position with respect to the bullet’s trajectory. In some cases, never receiving the shockwave. There were also a number of time constraints, and analyzing for both detecting the muzzle blast audio and the bullet shockwave would waste precious time. Therefore, it was simplified to a single layer utilizing machine learning to detect a muzzle-blast.

Artificial neural networks, especially convolutional neural networks proved to be worthwhile investment in research. While not conventional, the minuteman CNN can classify samples with a high degree of accuracy and very quickly. There was notable drawback in the time cost of needing to gather a dataset and train the program. However, as online dataset proved to be insufficient, an original one proved the be the most optimal, especially as our choice of microphone sensors for the array would dictate what the network classifies.

Audio files aren't very descriptive on their own. Much of the detail is in the frequency of waveform magnitudes. While Fourier transform methods, such as fast-time Fourier transform, were considered, it was ultimately considered that a wavelet transform would offer more dynamic and valuable information, especially for a data hungry neural network.

C# was ultimately selected for the program's graphical user interface, as it provided easy to use tools and functionality for the purposes of the graphical interface. It was also enough for the USB software and multilateration algorithms. Python proved to be the most valuable resource for training, testing, and implementing the neural network and wavelet transformation functions. The script was run by creating a new process from the main program. Both programs sent and received data through a named pipe with the main program acting as the server.

Audio is received from five microphone sensors on the main array. The audio is processed in 80ms buffer arrays. This is processed by the array's MCU, which sends it through a USB to the host computer. The minuteman program then detects gunshots within the audio samples. If a gunshot is detected it passes it to the program's multilateration algorithm, which determines the direction from which the gunshot originated. Then this information is displayed to the user through the program's GUI.

The end result is a detection system, using just one device—although several can be employed—that is not only affordable, but accurate, and can quickly be employed in situations that require it. Although a facility might not be equipped with such devices in the event that it might need it, a vehicle, possible a police patrol car, can be brought to such events.

Appendices

Appendix A: References

[1] “Introducing the Barracuda™ Intruder Defense System.” School Kids Healthcare Blog RSS, www.schoolkidshealthcareblog.com/2015/introducing-the-barracuda-intruder-defense-system.html

[2] White, Blair. “Home.” Slide To Safety, Slide To Safety, 13 Jan. 2019, slidetosafety.com/

[3] Alvarado, D., Kon, C., Babamir, Z., & Salazar, L. (n.d.). *Gunfire Location and Surveillance System*(Rep.). Retrieved from http://www.eecs.ucf.edu/seniordesign/fa2013sp2014/g03/project-documentation.html
Previous Senior Design Project

[4] “Home.” ShotSpotter, www.shotspotter.com/.

[5] Nikolaos, Tsiatis E. “Recording and Calculating Gunshot Sound—Change of the Volume in Reference to the Distance.” AIP Conference Proceedings, vol. 1203, no. 1, Jan. 2010, pp. 846–851. EBSCOhost, doi:10.1063/1.3322568.

[6] “Boomerang Mobile Shooter Detection System.” Mil, milcom-security.com/project/boomerang-mobile-shooter-detection-system/.

[7] “Warrior-X Individual Worn Shooter Detection System.” Mil, milcom-security.com/project/warrior-x-individual-worn-shooter-detection-system/.

[8] “INDOOR DETECTION.” The Guardian Indoor Active Shooter Detection System, shooterdetectionsystems.com/indoor-detection/.

[9] “Active Shooter and Gunshot Detection Provided by STANLEY Security.” STANLEY Convergent Security Solutions, Inc, www.stanleysecuritysolutions.com/shooterdetection.

[10] SenseShot® Indoors Gunshot Detection and Localization System. “SenseShot Indoors Outdoors.” Home - SenseShot® Indoor and Outdoor Gunshot Detection and Localization Systems, senseshot.com/.

[11] Pathrose, Nimmy, et al. “Analysis of Acoustic Signatures of Small Firearms for Gun Shot Localization.” 2016 IEEE Annual India Conference (INDICON), 2016, doi:10.1109/indicon.2016.7839135.

[12] Snow, W. "Survey of Acoustic Characteristics of Bullet Shock Waves." IEEE Transactions on Audio and Electroacoustics, vol. 15, no. 4, 1967, pp. 161–176., doi:10.1109/tau.1967.1161921.

[13] *Microphone Tutorials*, www.mediacollege.com/audio/microphones/.

[14] "Microphone." *Wikipedia*, Wikimedia Foundation, 16 Mar. 2019, en.wikipedia.org/wiki/Microphone.

[15] "CMC-6035-130T." *CUI Inc*, www.cui.com/product/audio/microphones/electret-condenser-microphones/cmc-6035-130t.

[16] Sengpiel, Eberhard. Calculation Speed of Sound in Humid Air and the Air Pressure Humidity Moist Air Water Vapor Density of Water Atmospheric Pressure - Sengpielaudio Sengpiel Berlin, www.sengpielaudio.com/calculator-airpressure.htm.

[17] "ADS-B." FAA Seal, 27 Oct. 2016, www.faa.gov/nextgen/programs/adsb/atc/wam/.

[18] Wolff, Christian. "Radar Basics." Radar Basics - Multilateration, www.radartutorial.eu/02.basics/rp52.en.html.

[19] "Loran-C." Wikipedia, Wikimedia Foundation, 11 Mar. 2019, en.wikipedia.org/wiki/Loran-C.

[20] "Multilateration." Wikipedia, Wikimedia Foundation, 7 Feb. 2019, en.wikipedia.org/wiki/Multilateration.

[21] "Comparing Python to Other Languages" python, Python Software Foundation, 17 June. 2015, www.python.org/doc/essays/comparisons/

[22] Naidu, Naveen. "Software Engineering | Agile Software Development" GeeksforGeeks, www.geeksforgeeks.org/software-engineering-agile-software-development/

[23] Piczak, Karol J. ENVIRONMENTAL SOUND CLASSIFICATION WITH CONVOLUTIONAL ...Institute of Electronic Systems Warsaw University of Technology, http://karol.piczak.com/papers/Piczak2015-ESC-ConvNet.pdf.

[24] Pasquale Foggia, Nicolai Petkov, Alessia Saggese, Nicola Strisciuglio, Mario Vento, Reliable Detection of Audio Events in Highly Noisy Environments, Pattern

Recognition Letters, Available online 9 July 2015, ISSN 0167-8655, http://dx.doi.org/10.1016/j.patrec.2015.06.026.

[25] Jiang, Xiao-ping, and Guan-qiang Cao. “Belt Conveyor Roller Fault Audio Detection Based on the Wavelet Neural Network - IEEE Conference Publication.” Belt Conveyor Roller Fault Audio Detection Based on the Wavelet Neural Network - IEEE Conference Publication, IEEE, https://ieeexplore.ieee.org/document/7378120/.

[26] Smith, Steven W. The Scientist and Engineers Guide to Digital Signal Processing. California Technical Pub., 1997, www.dspguide.com/.

[27] Beis, Uwe. Active Low-Pass Filter Design and Dimensioning, www.beis.de/Elektronik/Filter/ActiveLPFilter.html.

[28] “HOW TO DESIGN A PCB LAYOUT.” Circuit Basics, http://www.circuitbasics.com/make-custom-pcb/.

[29] “How to Build an Overvoltage Protection Circuit.” Learning about Electronics, http://www.learningaboutelectronics.com/Articles/Overvoltage-protection-circuit.php.

[30] “LMT4622A.” Analog Device, https://www.analog.com/en/products/ltm4622a.html.

[31] “Selecting a Fuse Value.” In Compliance, https://incompliancemag.com/article/selecting-a-fuse-value/.

[32] “Max1044 Switched-capacitor Voltage Converters.” maxim integrated, https://www.maximintegrated.com/en/products/power/charge-pumps/MAX1044.html.

[33] “TPS62125.” Texas Instruments, http://www.ti.com/product/TPS62125.

[34] “Arduino Due.” Arduino, https://store.arduino.cc/usa/due.

[35] “INTEL® MAX® 10 FPGA DEVELOPMENT KITS AND BOARDS.” intel, https://www.intel.com/content/www/us/en/products/programmable/fpga/max-10/kits-and-boards.html.

[36] “Trace Width Calculator.” AC 4PCB, https://www.4pcb.com/trace-width-calculator.html

[37] Kingma, Diederik P, and Jimmy Lei Ba. “Adam: A Method for Stochastic Optimization.” *ICLR 2015*.

[38] http://www.montana.edu/rmaher/publications/maher_aesconf_0608_1-8.pdf

Appendix B: Permissions

Arduino Due Permission

Sara Therner (Arduino)
Apr 9, 11:26 CEST

Dear Nathaniel,
Thank you for your email. Photos of boards and other Arduino references shall be used for explanatory or descriptive purposes only. Being a school project we understand that you need to be able to present your project and the knowledge you gathered during so you may use the image of DUE for your senior design project. Please note that other rules may apply in case you work with Arduino related products in the future (outside school) so it is always good to check with us first.

Best of luck with the design project.

Best Regards,

Sara Therner
Trademark & Licensing Manager
Arduino Customer Support

Nathanieldunn992
Apr 9, 02:10 CEST

Name: Nathaniel Dunn
Email: Nathanieldunn992@knights.ucf.edu
Subject: Permission

Message: Dear Arduino,

My name is Nathaniel Stence Dunn and I am in my senior year at UCF getting my Electrical Engineering degree. I am emailing you for permission to use the images below for my senior design project.

https://store.arduino.cc/usa/due
Use the first image of the Arduino Due.

Best regards,

Nathaniel Dunn

Nathanieldunn992@knights.ucf.edu