Frequency-Modulated Continuous-Wave Radar

Group 16
Casey Sanchez (EE)
Fahim Ahmed (EE)
Mikayla Peña (EE)
Sebastian Peralta (EE)

Sponsor: Dr. Xun Gong, UCF Professor

December 6, 2022

Table of Contents

1.0 Executive Summary

Radar systems have become a prevalent technology in modern times. Due to their versatility, they can be found in a variety of applications, from tracking global weather patterns to mapping terrain. Parameters such as frequency, antenna size, antenna number, power level and mobility all play a part in determining a particular radar's capabilities and what kind of information can be extracted from its received signal. Although large, high-power radar systems have their place, particularly in the defense and aerospace industries, small-scale radar are oftentimes better suited for everyday consumer use. These systems can be used for applications where high power and large range are not necessary, but relatively small dimensions, ease of use, low cost, and direct access to the necessary information is a priority.

One piece of information that radar systems are often used to find is the velocity of objects in the radar's field of view; this system is often referred to as a Doppler radar given that it uses the Doppler effect to calculate the velocities. This velocity information can be used simply to analyze an environment, or it could be used to control an additional subsystem. Although radar data could be utilized post-collection, a more versatile system would have software that could process the collected data in real-time. Our team's project is to design such a system. We plan to build a portable, frequency-modulated continuous-wave (FMCW) radar module and implement a user-friendly application programming interface to accompany it. The API will allow the user to easily extract different information from the raw radar data in real time.

The system will be divided into three parts: the radar module itself (purely hardware), the API (purely software) and the USB interface between them (a combination of hardware and software). A signal of a certain frequency will be transmitted from the radar module, and that signal will bounce off objects in the environment and return to the radar receive antenna. The signal is then filtered and amplified in hardware and sent through the USB interface to the host computer. The host computer will then use our API to extract the desired information from the raw data.

One of the main concepts of our project is that it is a multi-purpose consumer product. The radar module and accompanying software can be used by a consumer for a variety of applications, and any particular use of the system does not define the limits of the system's capabilities. In order to demonstrate just one application of the system, our team will position the radar module beside Pegasus Drive on UCF's main campus. We will use our system to track, in real-time, the maximum velocity of vehicles on that street.

2.0 Project Description

2.1 Motivation

Radar system design is highly diverse from a technical perspective, spanning topics from wireless communication to programming digital user interfaces. A project such as

our radar system therefore involves knowledge of several electrical engineering subdisciplines. Design of the FMCW radar itself requires knowledge of microwave signal propagation and radio frequency signal processing. The USB interface consists of an analog to digital converter and microcontroller, which requires knowledge of embedded systems. And implementing the API involves programming and digital signal processing. This means that each of our team members will be able to demonstrate technical ability in the area related to their academic subfield. Additionally, due to signals traveling through the subsystems sequentially, team members will also need to have a basic understanding of subsystems interacting with their own. This affords an excellent opportunity for the team to expand our knowledge beyond the subfield-specific courses we have chosen on our academic track. The diversity in our system's technical requirements and the subsystem interconnectivity presents a challenge that our team is ready and willing to take on.

Another motivation our team has for completing this project is the wide range of possibilities for application of our system. We will only be demonstrating one application of our system for academic purposes, but in reality, our system could be used for many other things. The main advantage of our system is that the radar module is portable, meaning it is ideal for aiming at different targets while maintaining its wide field of view. Tracking the speed of vehicles, measuring terrain height, and monitoring activity in parking garages are just a few examples of how our system could be used.

Lastly, as aforementioned, one of our main focuses is creating a consumer product. Given our constraints, our end product for academic purposes will likely leave much room for future improvement from a marketing standpoint. Consumer needs would ultimately drive further development of the system design, both on the hardware and software side. The software capability could be expanded to extract additional parameters from the radar data or potentially present the information pictorially rather than numerically. The hardware could be modified to allow for the radar frequency or power to be set by the user, according to what is required of the specific application. The size and aesthetics of the system could also be improved. Completing our project is just the first step to creating a highly successful, marketable product.

2.2 Goals

There are three goals for the FMCW radar project. The primary goal aligns with our system's minimum expected level of functionality, and it is to measure the velocity of vehicles on Pegasus Drive on UCF's main campus. Successfully completing this goal will demonstrate that the radar module and API are capable of detecting movement in a dynamic environment. Pegasus Drive is a convenient location for the demonstration because vehicles often accelerate and decelerate in a short span of time due to foot traffic across the drive. This will allow our team to verify that our system can accurately detect velocity even with nonzero acceleration present. For the demonstration, the radar and host CPU will be positioned beside Pegasus Drive, pointed toward oncoming traffic. Either direction of traffic would do, but ideally the radar should be positioned in a way as to minimize interfering readings from background movement. During the demonstration,

the radar will remain on, and at a minimum, the API should output the maximum detected velocity at any given moment. For instance, if a vehicle turns the corner slowly, speeds up, and then comes to a stop, the real-time readings output by the software should begin low, increase and then decrease to 0, with all readings remaining within the allowed margin of error. As an extension of this demonstration, the system can be taken to alternative locations to test for higher velocity readings (Gemini Boulevard would be a good alternative).

The second goal for our project is to demonstrate the ranging functionality of our radar. This is designated as a stretch goal because achieving it will require additional work on the software side of the design. It will be worked toward if the projected completion date of the primary goal is well before the scheduled deadline. For the ranging demonstration, the radar should be connected to the host computer in a way that allows the module to be pointed in different directions. To verify that the system can accurately detect range, the radar should be pointed at stationary, relatively large objects that span a range of distances away. A convenient place to perform the demonstration would be in the UCF arboretum, where there are many large trees and very few moving objects to interfere with the readings. The API output should update in real-time to show the minimum distance detected by the radar.

The final goal for our project, and the second stretch goal, is to use the data output from our API as the input to a feedforward control system. The output of the control system could be used as an alternative way to visualize the velocity or ranging data being collected from the radar. A relatively simple example would be to feed the velocity output of the API to a microcontroller, which then controls a series of LEDs. The maximum velocity detected by the radar could linearly relate to the speed at which the LEDs blink, or perhaps the velocities could map to a color spectrum, where blue corresponds to slow speeds and red to high speeds. Whatever implementation is chosen, this final goal will require the most design work for both the hardware and software. It will be attempted after goals one and two are completed, and if the schedule allows for it.

One important characteristic of these three technical goals is that they all operate in real time. The real-time signal processing is entirely dependent on the success of the USB interface and data retrieval aspect of the software, which is the most complex part of the project design. In addition to defining stretch goals, our team has also defined a “bare minimum” goal. This bare minimum goal eliminates the need for real-time data processing. It simply requires that one (or more) of the aforementioned goals be achieved without the need for the information to be processed in real-time. For instance, rather than the API output velocity readings from cars in real-time, the received radar signal can be recorded for a certain length of time, then loaded into the API for post-processing. A corresponding visual of what would have been seen during real-time operation would be the acceptable output of the API for this bare minimum goal.

In addition to the team’s technical goals, there are also project management goals the team will strive to achieve. The first is closely related to the technical goals, and this is

keeping an accurate record of the project design iterations. Every concrete design change the team makes from start to finish should be well documented. This includes modifications to existing designs from external sources and any changes made from our original designs for each subsystem. Changes should be accompanied by a reason the change was made. The design documentation should be updated on a monthly basis.

A second project management goal is to perform and document the results of our final demonstration. Regardless of which technical goals are ultimately achieved or which design iteration of the project subsystems is used, the results of the system demonstration need to be well-documented. In general, the project results should include an externally measured input parameter (such as the actual distance to an object), the radar measured parameter (being the output of our software), the calculated error and the time delay (if the demonstration is performed in real-time). These four parameters should be measured and recorded for at least four targets per technical goal. Documentation of the final demonstration is important because it shows the proven capability achieved by the system at its last design iteration. The more technical goals we can achieve, the better, because more demonstrations will show the full extent of the final system's capabilities.

Lastly, our team strives to meet on a regular basis and keep to our project schedule. The best way to ensure the team meets its deadlines, both those from the Senior Design course and those created by our team, is to track progress towards objectives at least twice a week. Our team will meet with this frequency and record what progress each team member has made toward their individually assigned goals since last meeting. This will make it clear whether the team is making sufficient progress towards meeting our deadlines.

2.3 Objectives

In order to achieve our goals, steps must be taken to design and complete each of the subsystems in the project. The three subsystems are the radar module, the USB interface, and the API. Each of these subsystems varies in the amount of effort needed to design and construct it, and the effort generally increases as we aim to achieve each of the successive goals outlined in the previous section. We will begin with a discussion of the first subsystem, the radar module.

Our radar module is centered around the MIT Lincoln Laboratory laptop-based radar, which anyone can build as they work through the corresponding MIT OpenCourseWare material. The details of that project will be discussed in further detail in Section 3.1 "Similar Products." There are two categories in which the components for the radar can be placed: those that operate at higher, microwave frequencies, and those that operate at lower, radio frequencies. There are also mixers that convert from one frequency to another, but for the purposes of this discussion, the mixers will be placed in the microwave frequency category.

Dr. Xun Gong, professor at the University of Central Florida and the primary advisor for our project, has provided our group with a partially built version of the MIT radar for us to start with. The part of the radar he provided includes all the high-frequency components; they are assembled and fully functional. What is left to be completed is the low-frequency circuits, which are the low-pass filter, gain stage, and modulator. MIT OpenCourseWare provides schematics for these circuits, but the course is dated and details regarding the design of the circuits is limited. In order to complete the radar, our team will need to update the designs for each circuit and test them to verify proper operation. Once the circuits have been tested, the team can design printed circuit boards for use in the final module prototype. Work regarding completion of the radar module constitutes our team's first objective.

The second objective is to design an operational USB interface. A variety of interfaces can be used to communicate between the radar module and the host computer, including wireless communication or the audio interface already used in MIT's project design. Our team has decided that a USB interface would be a fair compromise between feasibility and ease of use. The input of the circuit is the analog output of the radar module. The output of the interface is a digitized version of the radar output, sent through USB2. The interface should transmit the digitized signal quickly and with minimal error introduced.

The third objective is to design a functional and easy-to-use API. There is much room for flexibility in the implementation of this API. Further details will be discussed in a later section, but the most beneficial characteristic of the API is that it will be aligned with radar signal processing standards. This is not a necessity, but using a standard will ensure that the functions and architectures are easily understood by certain end users who wish to slightly modify the functionality of the system. For the purposes of our team's project, the code developed should not only be able to extract the necessary information from the radar module signal but should also be somewhat adaptable. Certain parts of the final code may ultimately go unused in our demonstrations, but they will be available for later use.

Because the three goals for the project each have their own requirements for the API, depending on which goals the team has the resources to aim for, the built-in capabilities of the API will be changed or added upon as needed. For our primary goal, the only necessary capability of the API is that it takes in a digitized received radar signal and outputs the maximum instantaneous velocity detected in real time. For our second goal, the API must also be able to extract range information, and for the final goal, the API must use a standard communication protocol to output either velocity or ranging information to an external device such as a microcontroller.

The fourth objective is to design the PCBs for the project. Two things will be taken into consideration to design the PCBs. The first is testing. The more components that are placed on a single PCB, the harder it is to troubleshoot issues. The circuitry for the project will therefore be broken up into multiple PCBs to facilitate the testing process. The second consideration is the size and shape of the PCBs. Although the overall size

of the radar module will be largely dictated by the size of the antennas, which have already been provided by Dr. Gong, it is still important to ensure the shape of the PCB makes it easy to install on the final system. Additionally, the components to be soldered onto the PCB should be large enough for easy assembly, but not so large as to make the entire PCB footprint unnecessarily large.

The fifth and sixth objectives are to acquire the necessary materials for the project and assemble the prototype. This applies only to the USB interface and radar module. Once the subsystems have been assembled, another phase of testing will be required. This is to verify all the parts are working as they should prior to the final assembly. Once the radar, USB interface and software have all been independently tested, the final objective is to integrate the system and attempt to reach the primary goal of the project.

It is possible that the final testing and integration stage may reveal shortcomings or issues with regard to system design. Our team's approach will be to meet the final objective as quickly as possible to leave time for circling back and making modifications to the system.

2.4 Requirement Specifications

2.4.1 Desired System Qualities

2.4.1.1 Accuracy Requirements

There are certain qualities in a product that will guide our requirement specifications. Accuracy is one important quality; our primary goal of measuring car velocities will be used as an example. If a car comes at 30 meters per second and the radar system reports 10 meters per second as its measurement, the radar is essentially useless. If the system reports 30 meters per second but only after a long delay, the radar is not much use either unless the results are adjusted for the time delay (this is not very useful for real time analysis). If the radar reports 28 meters per second with a very short delay, there is still error present, but the system is suitable for general analysis of the environment. In summary, our system needs to prioritize accurate results, but not so much as to cause excessive strain on the technical design.

The first accuracy requirement specification will be imposed on our system that says the output measurement must be between plus and minus 10 percent of the actual parameter value. This means that the system error is allowed to increase slightly as the magnitude of the measured quantity is increased, such as for very high velocities or very long distances. Verification of this requirement involves taking an accurate, external reading of the parameter in question and comparing it to the measured parameter generated by our system. This is fairly straightforward in the case of the ranging goal, for instance, where the distance can be easily measured between the radar module and a stationary object. For velocity error calculation, a calibrated commercial off-the-shelf device with a known error can be used to measure the velocity

of the target object. Alternatively, our team may choose to acquire a target that we can control the velocity of, so the accurate velocity will always be known.

The second accuracy requirement specification to be imposed on our system says that the output measurement must be generated and displayed by the software no longer than three seconds after the measured parameter occurs in the environment. This means that if a velocity of 5 meters/second appears in the environment, and then immediately disappears, it should be no longer than three seconds before the system displays 5 meters/second. In actuality, this requirement is somewhat lax, as real-time system outputs often appear nearly instantaneously. For the sake of establishing an absolute maximum lag time for our software and taking into account data transmission complexity in our anticipated subsystem architectures, three seconds will be used.

The final accuracy requirement specification has to do with the transmitted frequency. In general, as frequency is increased, accuracy of the radar increases as well. Frequency constraints will be discussed more later on, but for now, we will impose a requirement that the system must transmit at no less than 2 GHz for sufficient radar accuracy.

2.4.1.2 Operating Distance Requirement

Maximum operating distance is another system parameter that must be established. This is the longest length at which the radar output will remain within the maximum error requirement outlined above. The desired maximum operating distance is largely dependent on the specific application of the radar. Since our radar module is meant to be multi-purpose for the general consumer, it can be assumed that the targets are within 50 meters of the user. This means that at the very least, our radar should be able to sense targets up to 50 meters away. Being able to accurately sense anything further is not necessary, but it would be an improvement.

2.4.1.3 System Runtime Requirement

A characteristic similar to maximum operating range is minimum system runtime. The necessary runtime for the system is largely dependent on the application. In general, the system should remain on long enough to allow for sufficient data collection in an environment. If the system loses power soon after powering on, not only would the user have to consistently replace the batteries, but the user would not be able to collect very much successive information before having to put the module down. A minimum total runtime of 30 minutes will be the system requirement. While this does mean the radar module would not be ideal for long-term, continuous environment surveillance, it will work nicely for measuring environment parameters in short sessions.

Verification that the runtime requirement has been met is slightly tricky, since in actuality the runtime is not entirely dependent on the system characteristics. It also depends on the type of batteries that are used. Several batteries with the same voltage may have different capacities, which will then translate into different system runtimes. For the sake of catering to the general consumer, our team will use batteries that can be easily

acquired in a local hardware shop or general store. The capacities and characteristics of those batteries will be used as the standard to gauge the runtime performance of our radar module.

2.4.1.4 Transmitted Power Requirement

How much power the system consumes is closely related to runtime. The more power the system consumes, the shorter the run time will be, regardless of what type of (presumably finite) power supply is used. Additionally, high-powered electrical systems have the tendency to generate excessive heat, which will be discussed further in the Health and Safety section of the paper. The primary element of our system that affects power use is the transmitted power of the radar; the other elements of the system have reasonably low current draw that will not raise a power-consumption concern. Prior knowledge and research has shown us that our desired radar accuracy can be achieved by transmitting on the order of 10 milliwatts of power. This will be set as the lower threshold for transmitted power. Although lower transmitted power is better for runtime, as will be illustrated in the House of Quality figure, on its own a high transmitted power is actually better, in general, for a radar system. For this reason, a lower threshold for transmitted power is established, and the proper balance must be found between using a higher power and compromising other elements of the system.

2.4.1.5 Module Size Requirements

Portability is an ideal characteristic for our system. The basis is simple: the end-user should be able to pick up the radar module and point it in different directions. This makes it easier to take multiple measurements in an environment without the hassle of moving a large piece of equipment. Two parameters that affect portability are weight and dimensions. The requirements for these two parameters will be set to less than 3 kilograms and less than 0.5 cubic meters, respectively.

2.4.1.6 System Cost Requirement

The final characteristic to be considered is cost. Radar systems are relatively expensive when compared to other systems in electrical engineering due to the high cost of RF signal processing components. Our budget for the project, which is almost entirely dependent on the cost of the hardware, will take into account both hardware costs and each team member's maximum contribution to the project. The requirement for cost therefore states that in total, the system must cost no more than $800.

2.4.2 Requirements Summary

Table 1 below summarizes the quantitative requirements discussed in the previous section. Note that measurement error is the only requirement with both an upper and a lower bound, denoted this way for clarity. Another way of stating it is that the absolute value of the measurement error must be no greater than ten percent of the actual parameter value.

Table I
FMCW Radar Requirements

FMCW Radar Requirements		
Requirement	Minimum	Maximum
Measurement Error	-10%	+10%
Max Operating Distance	50 meters	---
Runtime	30 minutes	---
Cost	---	$800
Size	---	0.5 cubic meters
Frequency	2 GHz	---
Output (TX) Power	10 milliWatta	---
Software Lag Time	---	3 seconds
Weight	---	3 kilograms

2.4.3 House of Quality

Designing a marketable consumer product involves finding a good balance between conflicting consumer desired qualities. Every desired quantity cannot be maximized, because increasing one will inevitably decrease another. For instance, if the cost of the system is very low, it will inevitably cause the system accuracy to become low as well. High accuracy means purchasing high-quality parts, and high-quality parts are more expensive than low-quality parts. It becomes a balancing act between desired qualities.

The house of quality for our system, shown in Figure 1, shows the relationship between all the desired system qualities and the technical characteristics of the system. On the left side of the figure, each quality is listed with a plus or a minus sign, depending on whether the quality should be as high as possible or as low as possible, respectively. The top of the rectangular portion of the figure shows the technical characteristics of the radar system, and whether they should ideally be as high as possible (indicated with a plus sign) or as low as possible (indicated with a minus sign).

The intersection between a quality and a technical characteristic shows how moving the technical characteristic in the ideal direction would impact the quality parameter. The “roof” of the house of quality shows the interrelationship between the technical characteristics.

Note that some intersection points in the figure are blank. This indicates that there is no relationship between the intersecting characteristics. Technical characteristics with many plus signs in their corresponding intersection points can be easily moved in the desired direction without much impact on other parameters. For our specific project, measurement error and software lag time are two technical characteristics whose intersection points with other parameters are mostly blank squares and plus signs. This means the software lag time and measurement error can both be decreased as desired, and other system parameters will either be benefitted or remain unaffected.

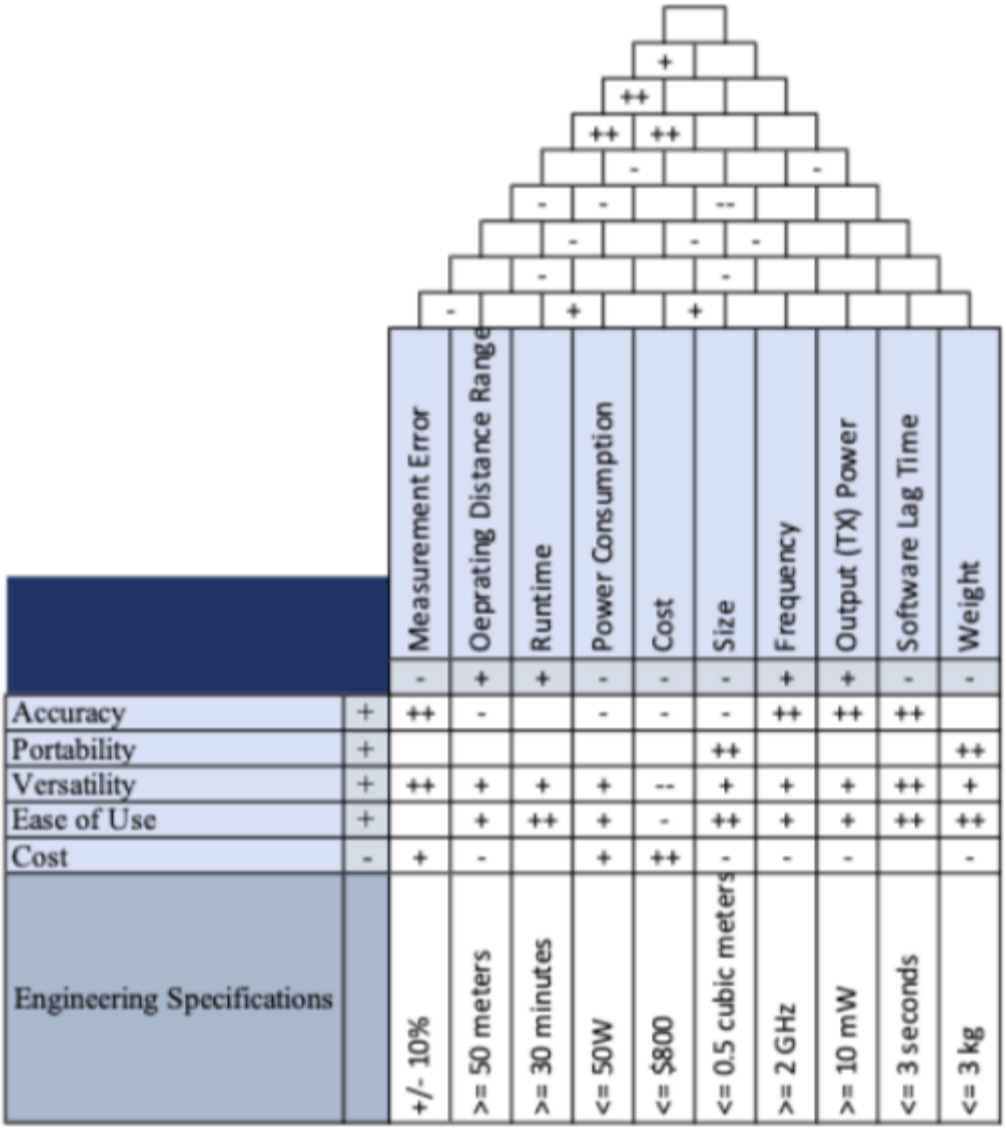

Fig. 1. FMCW radar project house of quality.

3.0 Project Research

3.1 Radar Background Theory

3.1.1 Radar Transmission Theory

The doppler in our system consists of an oscillator that after some attenuation amplification, gets split in two, where transmitted, the other half is fed into a frequency mixer which performs a multiplication function. The transmitter emits a microwave spherical wavefront, as shown in Figure 3. We could imagine this spherical wavefront

propagating from the emitter and reaching a target downrange. Some of the wavefront intersects with the cross-section of the target and is scattered back to the emitter.

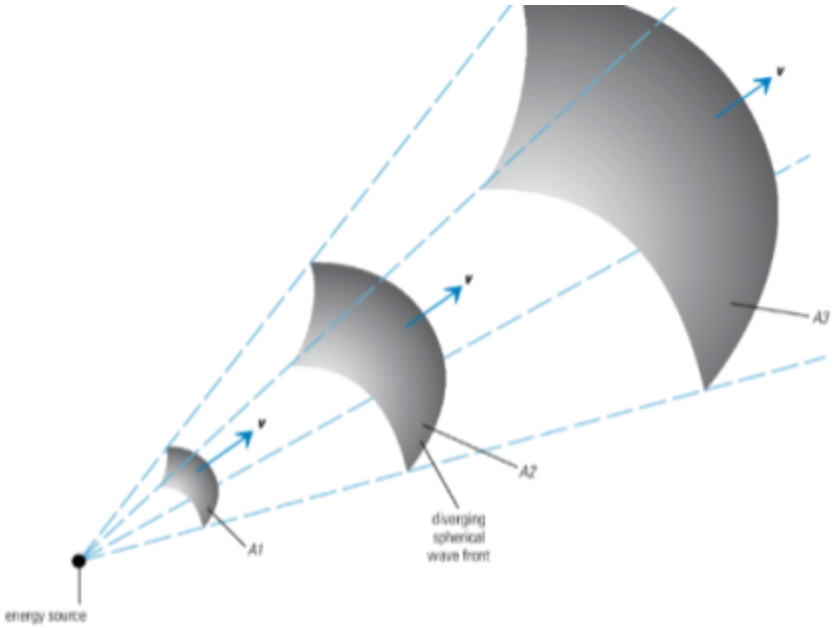

Fig. 3. Spherical wavefront.

In order to reduce the coupling between the transmitter and receiver and avoid deadening the received signal we have separated the two. In conventional radar systems one usually would have only one.

The received scattering is then amplified and fed into the multiplier. The difference comes out of the mixer and is amplified with three op-amps. The three op amps consist of an amplifier and a fourth-order low-pass filter.

The range to the object is calculated by determining the round-trip distance that light would have traveled in that amount of time. Note that signal power drops by the distance to the object raised to the fourth power. This tells us that the farther an object is, the more difficult it is to distinguish the received power pulse from the noise.

Below is the equation for range in a two-way (roundtrip) radar.

$$Range \simeq \frac{P_t G_t G_R \lambda^2}{(4\pi)^3 S_{min}} \quad (1)$$

$$P_t = transmitted\ power$$
$$G_t = transmitted\ antenna\ gain$$
$$G_R = receiver\ antenna\ gain$$

$$\lambda = wavelength\ of\ signal$$
$$S_{min} = minimum\ detectable\ signal$$

The gain of antenna is given by

$$G = \frac{4\pi A_e}{\lambda^2} \quad (2)$$

$$A_e = equivalent\ area \text{ of aperture}$$

Substituting into Equation (1) results in

$$\text{Range} \simeq \left(\frac{P_t \sigma \pi r^2}{2\lambda^2 S_{min}}\right)^{1/4} \quad (3)$$

We can observe based on the above equation that in order to double the range of a range system we would need to increase its power by a factor of 16. We could also increase the frequency by a factor of 4, or double the radius of the aperture. Understanding the technical background of our product lets us have a deeper insight into the design choices and how to improve the product in future iterations.

To minimize range ambiguity we want to ensure the distance to the target is greater than or equal to the speed of light divided by two times the pulse repetition frequency.

$$Distance \geq \left(\frac{c}{2*(Pulse\ Repetition\ Frequency)}\right) \quad (4)$$

3.1.2 Microwave Theory

There are fundamental differences between traditional circuit and network theory and microwave engineering theory. In the traditional circuit model a circuit is described by how much energy is guided through interconnected components. The sizes of the components do not matter in their function, and the voltage is the same along a transmission line. In network theory the ratio between voltage and current is always a constant. We are able to use Kirchhoff's laws to solve for unknown values in a straight-forward way. The ratio of voltage to current is called impedance. We can model resistors, inductors, and capacitors using impedance. In order to model sinusoidal time variance complex impedance can be used. When signals are traveling through a line, only one mode can propagate.

We can model an infinitely small cross sectional area of a transmission line with the generalized lumped-element model of a transmission line, as shown in Figure 4. This model allows us to calculate characteristic impedance, phase velocity, and both the real and imaginary parts of the propagation constant.

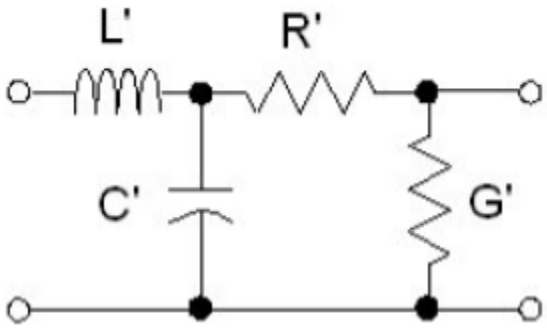

Fig. 4. Generalized lumped-element model.

The relationships between the wavelength, phase velocity, and phase constant are useful to understand. The relationship between wavelength and the phase velocity is given by

$$\beta = \frac{2\pi}{\lambda} \text{ (radians/length)} \tag{5}$$

where $\beta = \omega\sqrt{L'C'} = 2\pi f\sqrt{L'C'}$

The series impedance and shunt admittance are given as

$$Z' = R' + j\omega L' \tag{6}$$

$$Y' = G' + jwC' \tag{7}$$

The propagation constant is given as

$$\gamma = \sqrt{Z'Y'} \tag{8}$$

The characteristic impedance of a transmission line can be found understood using the following equations.

$$Z_0 = \sqrt{\frac{R' + j\omega L'}{G' + jwC'}} \text{ (low-loss)} \tag{9}$$

For lossless transmission lines the following wave equations can be used to derive the phase velocity.

$$-\frac{\partial v(z.t)}{\partial z} = L\frac{\partial i(z.t)}{\partial t} \rightarrow \frac{\partial v^2(z.t)}{\partial z^2} - \frac{1}{v_p^2}\frac{\partial v^2(z.t)}{\partial t^2} = 0 \tag{10}$$

$$-\frac{\partial i(z.t)}{\partial z} = C\frac{\partial v(z.t)}{\partial t} \rightarrow \frac{\partial i^2(z.t)}{\partial z^2} - \frac{1}{v_p^2}\frac{\partial i^2(z.t)}{\partial t^2} = 0 \quad (11)$$

This leads to

$$v_p = \frac{1}{\sqrt{L'C'}} \quad (12)$$

The general solution for a lossless transmission line is given as

$$v(z,t) = v^+(t - \frac{z}{v_p}) + v^-(t + \frac{z}{v_p}) \quad (13)$$

$$i(z,t) = i^+(t - \frac{z}{v_p}) + i^-(t + \frac{z}{v_p}) = \frac{v^+(t-\frac{z}{v_p}) - v^-(t-\frac{z}{v_p})}{Z_0} \quad (14)$$

$$Z_0 = \sqrt{\frac{L}{C}} \text{ (characteristic impedance)} \quad (15)$$

For time harmonic transmission lines (illustrated in Figure 5)

$$\frac{dV(z)}{dz} = -(R + j\omega L)I(z) \rightarrow \frac{dV^2(z)}{dz^2} - (R + j\omega L)(G + j\omega C)V(z) = 0 \quad (16)$$

$$\frac{dI(z)}{dz} = -(G + j\omega C)V(z) \rightarrow \frac{dI^2(z)}{dz^2} - (R + j\omega L)(G + j\omega C)I(z) = 0 \quad (17)$$

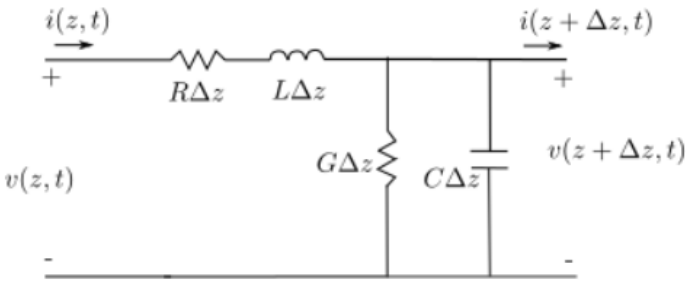

Figure 5: Time Harmonic Transmission Line.

$$Z_0 = \sqrt{\frac{L'}{C'}} \quad (18)$$

The standing wave ratio measures the impedance matching of the loads to the characteristic impedance of the transmission line. Standing waves will be created when the impedance is mismatched. The standing wave ratio is given by

$$SWR = \frac{V_{max}}{V_{min}} = \frac{1+|\Gamma|}{1-|\Gamma|} \quad (19)$$

Voltage on a transmission line is described by

$$V(z) = V_0^+(e^{-j\beta z} + \frac{V_0^-}{V_0^+}) = V_0^+(e^{-j\beta z} + \Gamma_L e^{+j\beta z}) \quad (20)$$

$$I(z) = \frac{V_0^+}{Z_0}(e^{-j\beta z} - \frac{V_0^-}{V_0^+}e^{+j\beta z}) = V_0^+(e^{-j\beta z} + \Gamma_L e^{+j\beta z}) \quad (21)$$

For reflection, a transmission line terminated in a short or open reflects all power back to the source. When considering input impedance (illustrated in Figure 6),

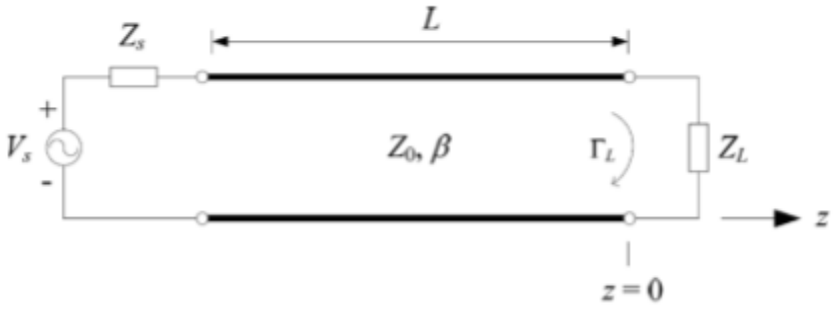

Fig. 6. Input impedance.

$$Z_{in} = \frac{V(z)}{I(z)} = Z_0\frac{1+\Gamma_L e^{+j2\beta z}}{1-\Gamma_L e^{+j2\beta z}} \quad (22)$$

$$Z_{in} = \frac{V(z)}{I(z)} = Z_0\frac{1+\Gamma_L(l)}{1-\Gamma_L(l)} \quad (23)$$

This gives,

$$\Gamma_L(l) = \frac{Z_{in}-Z_0}{Z_{in}+Z_0} \quad (24)$$

Note that L' can be thought about as the tendency of a transmission line to oppose a change in current. Conversely, C' can be thought about as the tendency for a transmission line to oppose a change in voltage. The phase velocity can also be found if we know L' and C'.

$$v_p = \frac{1}{\sqrt{L'C'}} \quad (25)$$

3.1.3 Antennas and Electromagnetics

An antenna can convert a signal voltage on a transmission line into a transmitted electromagnetic wave. A time varying signal applied to the transmit antenna induces an electrical current on the antenna which in turn produces electromagnetic radiation. Antennas are dielectric or metal structures that are specifically designed or chosen to provide an efficient launch of electromagnetic waves into space. Physically, we can conceptualize an antenna as providing a sea of free electrons which can be influenced by external forces and that offer zero resistance to movement.

We can derive from Maxwell's equations that electromagnetic radiation occurs when electric charge accelerates. We cannot directly see electromagnetic waves. In order to prove they exist, we can formulate Maxwell's equations as a wave equation in the form

$$\nabla^2\Psi - \frac{1}{v^2}\frac{\partial^2\Psi}{\partial t^2} = g \tag{26}$$

where Ψ is the field quantity. Next, we take the curl of Faraday's law to obtain the new vector field as

$$\nabla x \nabla x E = -\nabla x \frac{\partial B}{\partial t} = \frac{\partial}{\partial t}(\nabla x B) \tag{27}$$

$$\nabla x \nabla x E = -\nabla x \frac{\partial B}{\partial t} = -\epsilon\mu\frac{\partial^2 E}{\partial t^2} - \mu\frac{\partial J}{\partial t} \tag{28}$$

The right hand side of the above equation yields some important observations. In free space $J = 0$ and $\epsilon = \epsilon_0$. Therefore

$$v = c = \frac{1}{\sqrt{\epsilon_o\mu_0}} \tag{29}$$

As the electromagnetic wave traveling at the speed of light reflects off of an object, the reflected energy illuminates the receiving antenna which in turn induces a signal that travels through a coaxial cable. For an isotropic antenna, the electromagnetic wave propagates spherically from the excitation point. A directional antenna (which is the type our system uses) produces a gain radiation pattern as shown in Figure 7.

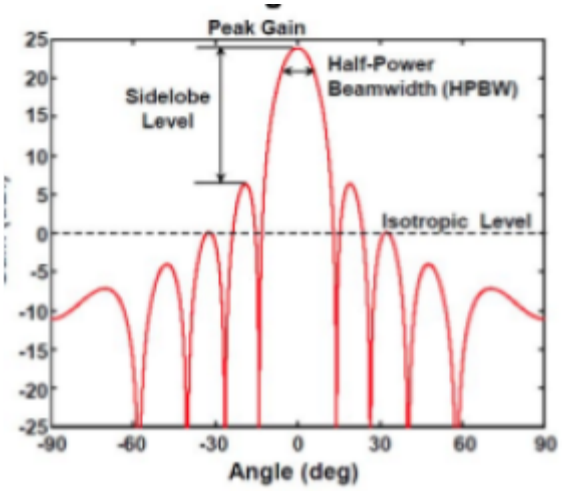

Fig. 7. Gain pattern for circular aperture.

Note that a directional antenna has a main beam that is pointed in a direction and sidelobes that radiate away from the beam. The electromagnetic waves are not sent in a perfectly circular beam. As the antenna diameter increases, the bandwidth gets narrower and the main beam's gain increases. The gain G of an aperture is given by the expression.

$$G = \frac{4\pi A_e}{\lambda^2} \tag{30}$$

The effective isotropic radiated power is a function of the transmitted power and the gain of the transmitting antenna.

When considering calculating the gain from measured power coupling between two identical antennas we can utilize the following relation.

$$G_{dBi} = \frac{1}{2}\left[10log\left(\frac{P_r(\theta,\varphi)}{P_t}\right) + 20log\left(\frac{4\pi r}{\lambda}\right)\right. \tag{31}$$

The voltage reflection coefficient tells us how much of the signal is reflected by the antenna. Note that for a well-designed antenna the magnitude of the voltage reflection coefficient should have a low value. The power transmission coefficient tells us how much power is transmitted relative to the incident power.

If a wire is placed next to a metal wall, the antenna radiation will have a phase shift of 180 degrees. If we want our radiated energy to efficiently be directed in a direction, we would place our antenna a quarter of a wavelength away from the metal wall, causing the phase shift to now be 360 degrees (see Figure 8). In a circular waveguide TE_{11} mode will be used.

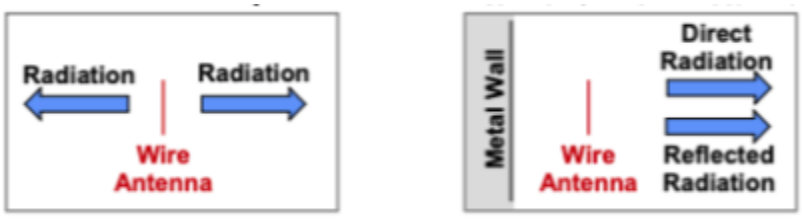

Fig. 8. Wire antenna near a PEC barrier.

In summary we have described some of the fundamental characteristics of antennas. We have also described the design of a circular waveguide antenna that can be used as a transducer, as shown in Figure 9.

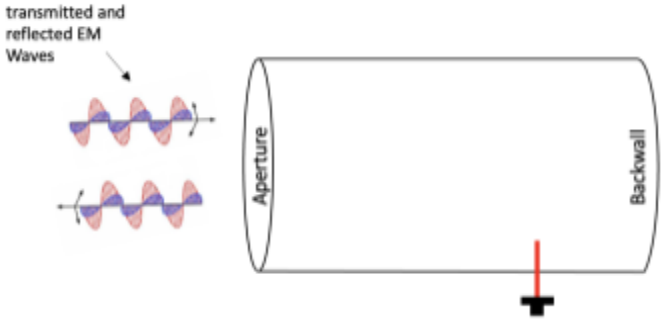

Fig. 9. Diagram of a “cantenna” antenna.

Here are the dimensions of the coffee cans we are using:

- Metal Can Length = 5.25”
- Metal Can Diameter = 3.9”
- Monopole Wire Length = 1.2”
- Spacing from monopole wire to back wall = 1.8”

An analog to digital converter will need to be implemented to digitize the phase differences for further signal processing.

The modulator will be able to select what mode the radar is in. The square wave mode will select velocity mode for the radar. The square wave is sent into the VCO. This creates a tone signal, this is then sent to the transmitter and bounced off an object moving at a certain velocity. This reflected signal will come back into the receiver antenna with a frequency shift. This signal is then mixed with the original transmitted signal. The lowpass filter selects the difference frequency. This is then going to be sent to the laptop or microcontroller where it will undergo a fast fourier transform. Our note that our FMCW radar is an active system, being powered by 9V batteries.

3.2 Similar Products

During our market search we found the following relevant products available.

3.2.1 MIT Small Radar System

Our initial product design used the MIT Small Radar System as a reference point for the initial design of our product. The setup uses a radar module to interface with a laptop via a two-channel audio cable, which has a 3.5mm audio jack on the laptop side. Key characteristics of the radar module itself are its FMCW architecture, coffee-can receive and transmit antennas, and S-band range of operation. The primary area of the project our team will be improving upon is the communication interface. Firstly, the MIT setup uses post-data acquisition processing. A .wav file is recorded while the radar is running, and the entire .wav file is analyzed digitally after the radar module has been turned off. Our project will provide real-time data processing capabilities, eliminating the need to record and store large files on the host device. Additionally, the audio cable used in the MIT laptop-based radar is effective but not ideal, as the majority of devices that interface with laptops do so via USB. Our project will increase flexibility by changing the audio cable to a USB2 cable. Elements of the MIT design that will remain largely similar in our design include the active gain stage, active low pass filter, and applications (doppler and ranging).

3.2.2 Police Radar Systems

Police radar systems are devices used to track the velocity of vehicles. Though such devices are available on the market, they are typically quite costly, ranging from $500 to $2000. Our system will be able to track velocity at a much lower cost. This will allow our product to reach an untapped market for consumers who wish to have the functionality of a velocity radar system without the steep cost. One difference between our system and police radar systems is that police radars typically operate in several frequency ranges including the X band(≈10.5GHz), K band(≈24GHz), and Ka band(≈33-36GHz) [18]. The primary reason for using these bands would be due to the fact that for higher frequencies, smaller antennas can be used, therefore maximizing space efficiency. To facilitate testing of our project, we will operate in a lower band where design components are a bit easier to obtain. Additionally, specific bands require licensing to use, and this must be taken into consideration for the purposes of our project demonstration.

3.2.3 Automatic Doors

Automatic doors operate by using a sensing system to send signals to a control system that will open the door for oneself. These sensing systems come in many different forms, but at a fundamental level sensing systems will be detecting disturbances within an electromagnetic field.

Our system will encode the received information in a digital format. This allows the potential for digital signal processing and integration with other systems. In theory one will be able to use our system to interface to a control system such as an automatic door. Automatic door installation on the market will run one on the order of thousands. Our system would allow savvy consumers the ability to use the data from our radar for various sensing and controls applications.

3.3 Relevant Technologies

3.3.1 Remote Sensing

When discussing technologies such as remote sensing, electromagnetic energy travels in the form of waves through free-space. The wavelengths can vary from the order of millimeters to kilometers, as illustrated in Figure 10. We are all familiar with visible light, which sits somewhere towards the middle of the electromagnetic spectrum. Visible light only covers a very small portion of the electromagnetic spectrum. We need instrumentation to observe any other forms of electromagnetic energy.

Remote sensing is the process of measuring the reflected and emitted electromagnetic energy at a distance; without contact. These measurements can be used to detect and monitor the physical characteristics of an object of interest. The required spectral, radiometric, and spectral resolutions will vary based on the desired application.

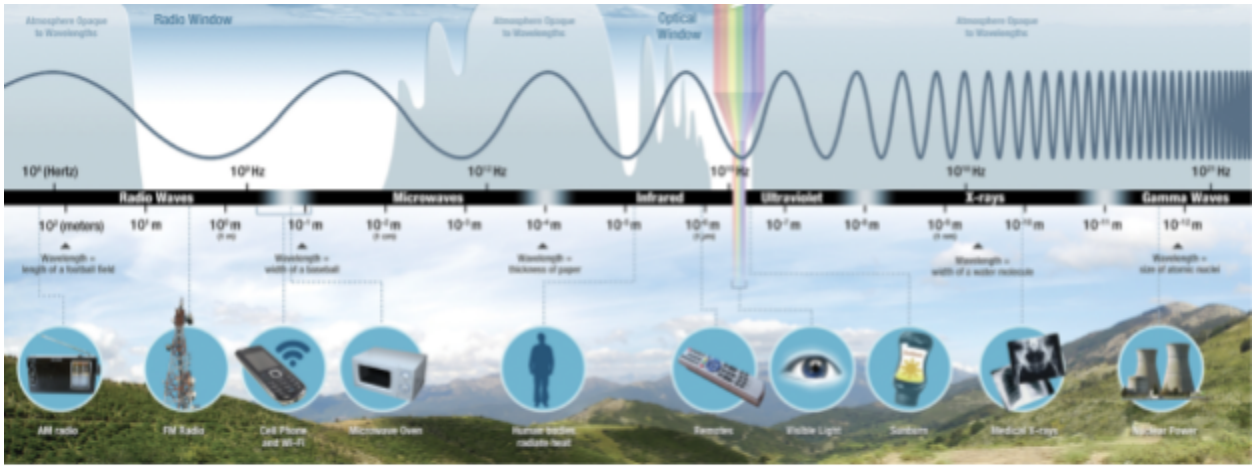

Fig. 10. Diagram of electromagnetic spectrum. Credit: NASA Science.

3.3.2 Pulse-Doppler Signal Processing

Our system will be able to determine the range of an object using Pulse-Doppler Signal Processing. This process utilizes the Doppler effect which describes the change in frequency of a wave in relation to an observer who is moving relative to the wave source. We frequently hear this phenomenon in our daily lives when we hear a car horn or police siren change pitch as it drives away or towards us.

It is important to recognize that we are not directly measuring the frequency shift, but are measuring the phase shift between pulses.

As the object that is reflecting the electromagnetic waves moves in position, the returned signal carries a phase difference relative to the previous pulse. This principle allows us to translate the information to electrical signals and ultimately compute range.

3.3.3 Transceiver Architectures

A basic radio system at a fundamental level, has a transmitter to propagate the electromagnetic wave carrying the signal and also a receiver to obtain these waves being propagated within the air. The transmitter will generally have a modulator that will combine a low frequency signal with the high frequency signal, using a mixer. The receiver will serve the opposite function of demodulating the signal and obtaining the signal at hand and then provide the system information sent over from a transmitter. There are 2 popular types of radio systems, which are the following: the Time-Domain Duplexing (TDD) and the Frequency-Domain Duplexing (FDD).

3.3.3.1 Time-Domain Duplexing Model

The TDD (shown in Figure 11) is a progressing area in the current area of 5G communication since it provides better quality signals for this range. The system on chip, is based on complementary mosfet technology (CMOS) which includes the modem and transceiver of the overall system. The RF frontend where we primarily work with high frequency signals is at a very base level composed of the power amplifier (PA), switch (III-V semiconductors, MEMS, etc.), and filter. This system is nice and convenient to implement since we are able to use a switch to swap from transmit or receive mode.

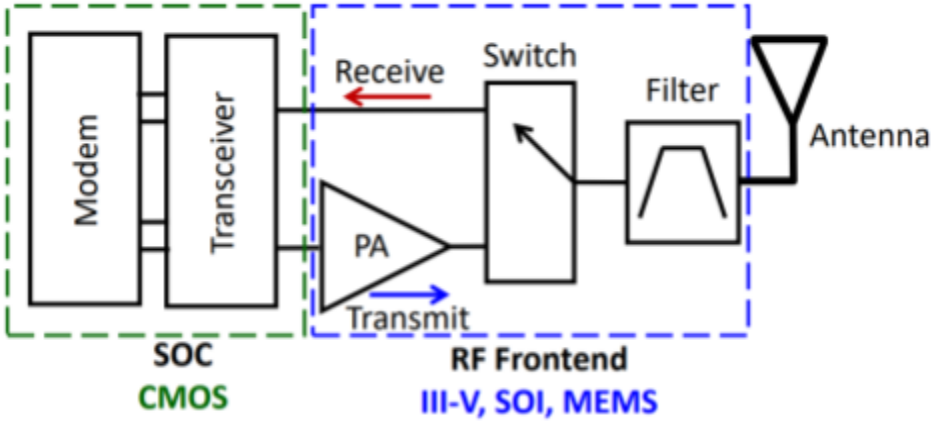

Fig. 11. The time-domain duplexing model.

3.3.3.2 Frequency-Domain Duplexing Model

The Frequency-Domain Duplexing (FDD) shown in Figure 12 has both transmission and receiving operating at the same time. It similarly has the CMOS technology for the SOC and the III-V compounds & MEMS for the RF frontend. 4G-LTE bands generally depend on this system to operate.

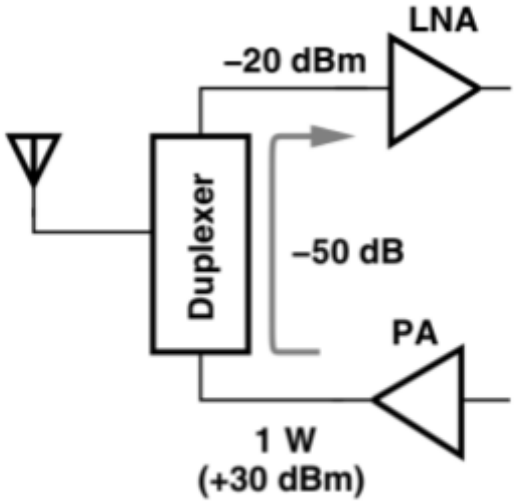

Fig. 12. The frequency-domain duplexing model.

Above we discussed a very high level understanding of the 2 techniques of the transceiver architectures. If we go further in detail, the system becomes increasingly complicated. The receiver system and transmit system have their own block diagrams, and the receiver is generally much more complicated than the transmit. This would because the receiver must deal with interferers, noise, non-linearity, dynamic range, etc. Generally, there are 3 different receiver architectures that we would consider. These are the Direct Conversion (RF to Low Frequency), Heterodyne (RF to IF [Intermediate Frequency] to Low Frequency), and the Super Heterodyne (RF to IF2 to IF1 to Low Frequency).

3.3.3.3 Direct Conversion Receiver

A direct conversion receiver, shown in Figure 13, demodulates the radio frequency signal using synchronous detection with a local oscillator that has a frequency close to the carrier frequency of the signal. This simple system is good, for overall simplicity of circuit design. Though the issue using this technique would be dynamic range. This technique was initially useless in its initial conception due to errors you may receive in the signal. Modern technology has allowed this to be considered again because of corrections that can be made with the aid of software.

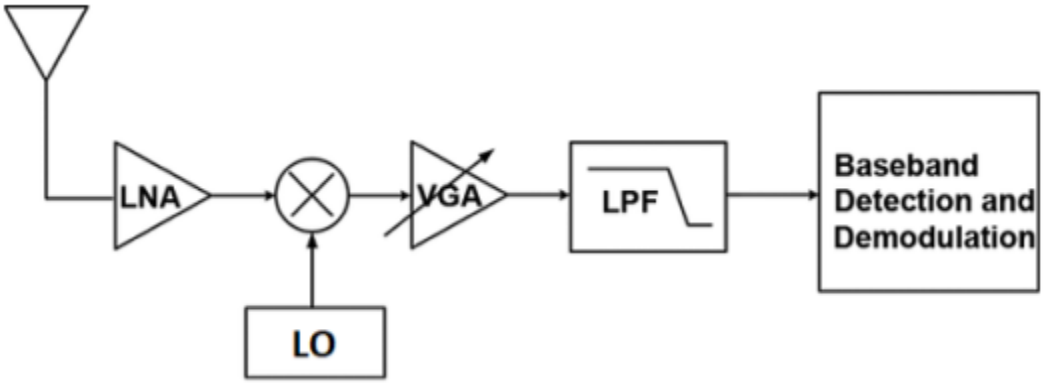

Fig. 13. A direct conversion receiver.

3.3.3.4 Heterodyne Receiver

A heterodyne receiver, shown in Figure 14, makes the use of an intermediary frequency signal and then into the low frequency territory. We are able to make use of this by using one mixer to bring a modulated RF signal to the modulated IF signal, which when demodulated brings us to baseband signal. The main reason we would want to use the intermediate frequency is to improve the frequency selectivity. Since radio frequency is really used in some form of communication circuits, we generally would like to easily pick out the desired signals and this would require filtering. The better the frequency selectivity, the better it is to pick out our desired signal. Finally, there is the super heterodyne receiver which makes use of 2 different intermediate frequency conversions. This would therefore cause us to use 2 mixers in the system actually. The primary reason we would want to use a system like this is to better tune the frequency selectivity than with just 1 IF conversion. We would first go from RF to the first IF which is “high” then to the second IF which is “low” and then finally to the low frequency signal.

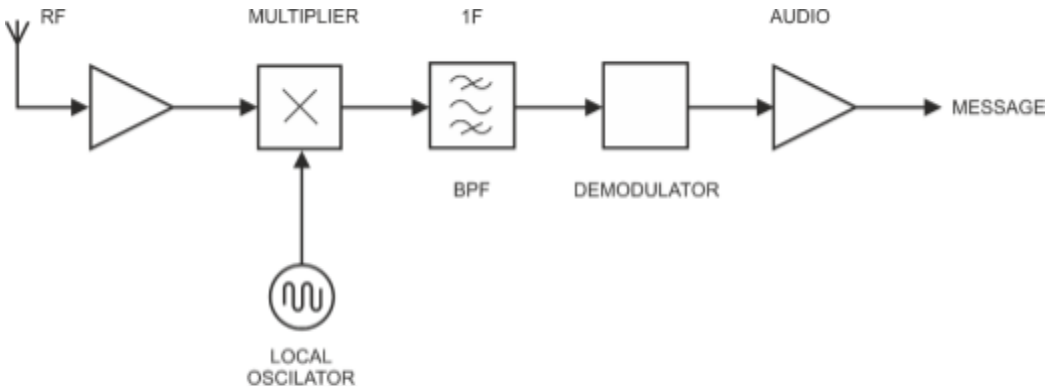

Fig. 14. A heterodyne receiver

3.4 Strategic Components and Part Selections

In order to strategically select our components and part selections, it is important to have some understanding of transmission line theory, as at high frequencies circuits behave much differently than at low-frequencies as seen in linear circuits theory.

Also, in order to properly understand the operation of the radar, it is important to make sure we are informed of the relevant theory that would help us visualize how the various components are working together in order to make the system function.

3.4.1 RF Signal Processing Components

Our RF signal processing components were chosen to meet the following parameters.

- ➢ Frequency = 2.4Hz
- ➢ Bandwidth = 80MHz
- ➢ Antenna Isolation = 50dB
- ➢ DC Power < 1W
- ➢ RF Power < 1W (EIRP)

Some realization constraints were inherent in our design. We wanted the system to be able to use off the shelf parts, use connectorized components, and run on 9V batteries. On the transmitted side we needed a modulated source, amplification, distribution, and an antenna for radiation. On the receiver end we needed an antenna to receive aperture, amplification, and a demodulator.

The transmitting RF component chain is made up of a cascaded two-port network model that consists of a VCO, attenuator, amplifier, splitter, and finally transmitting antenna. The receive chain consists of the receiving antenna, amplifier, and finally the mixer.

The receiving and transmitting antennas are dubbed the cantenna's due to the fact that cans were used as the circularly polarized waveguide antennas. These antennas are surprisingly highly efficient and suit our purposes very well. In order to understand the RF and antenna subsystem, it is worthwhile to describe some of the fundamental characteristics of antennas. We want to develop a stronger understanding of gain radiation patterns, power density, beamwidth, reflection coefficient, transmission coefficient, antennas arrays, measurements. We will use this understanding to follow the design, fabrication, and testing of an antenna that can be used as a transducer.

3.4.2 RF Design Description

The RF section of our hardware design will consist of a voltage controlled oscillator that is fed into an attenuator the signal is then amplified and sent through a beam splitter. Half of the signal will be transmitted, and the other half will be sent to the mixer. Once the signal is received the received signal will be amplified and then multiplied with the

reference signal that was sent into the mixer. The signal is then sent through a gain stage and then low-passed to prevent aliasing. This signal will be sent for further signal processing. This architecture is illustrated in Figure 15.

The system will be powered with a 9V battery pack, producing +9V and +18V. This powers the RF components and provides a reference voltage for the analog circuits The +5V will power analog circuits including the modulator, gain stage, and fourth order low pass filter.

Our next steps will include building the system on a breadboard and starting the testing phase.

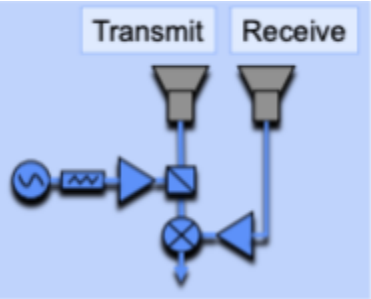

Fig. 15. RF hardware setup.

We will be using the ZX95-2536C+ Voltage Controlled Oscillator, the ZX60-272LN-S+ Low Noise Amplifier, the ZX10-2-42+ 0.1dB insertion loss splitter, and the ZX05-43MH-S+ Mixer.

3.4.3 Modulator

The modulator is what will be driving our transmitted signal. We will use an integrated chip that would be able to produce various waveforms such as ramp, sine, and square. For the purposes of range and velocity sensing we will be using a ramp waveform. Our modulator would modulate oscillator 1 yielding a voltage that is proportional to the transmit frequency. The modulator also provides a syncing function.

The Vtune voltage in the modulator will be proportional to the transmit frequency. The linear ramping of the voltage of the modulator will cause the oscillator to produce a linear FM chirp that is used for transmitting and receiving. The ramp time will be approximately 20ms with a triangular wave period of 40ms. The magnitude of the ramp will modulate the desired transmitted bandwidth.

For the high frequency modulation, we are using a ramp generator to modulate the low frequency information. The XR-2206 is a monolithic function generator which we will use to create our modulating signal in the low frequency range. This component is capable of creating high quality sine, square, triangle, ramp, and pulse signals that have

high stability and accuracy. This specific application we would like the XR-2206 to produce a ramp signal, which is popular to use in communication systems. Ramp generators are very popular to use since they provide a quick response and provide the capability to change the start-up & return flow time. The “Sync Pulse Inhibit” is a feature of the circuit since we would like to have a functionality that allows the converter to be turned off and on without having the cycle power. This would help up therefore, save power, reducing inrush current, help prevent input impedance problems and delay the converter turning on before the voltage stabilizes.The frequency space portion of the ramp generator is meant to help us adjust the frequency range we would like to use on the circuit based upon what range we would like to create the ramp at and will be manually done here with a potentiometer. The chirp rate adjust part is what the name implies, adjusts the rate of the chirp seen on the low frequency end. A chirp is generally defined as a signal where the frequency either increases or decreases over time. This specific part of a circuit is common to sonar, radar and laser systems since they are generally sending signals out to be transmitted and received. We also see the use of decoupling by the XR-2206 which is a common component to put near any integrated circuit since it protects the IC from any instantaneous changes in voltage and filters out unwanted noise that interferes with the IC outside of the system.

3.4.4 Mixer

Mixers are a popular IC to use in RF communication systems. Applications of interest include military radar, cellular base stations, etc. A RF mixer can be a 3-port passive or active device that is integrated in the modulator or demodulator. The ideal goal of the mixer is to change the frequency of the electromagnetic signal, preserving the signal as much as possible (phase and amplitude). As we have discussed earlier, a mixer is a fundamental part for the heterodyne receiver models that converts RF signals to IF.

There are 2 primary conversions that the mixer is used for, downconversion and upconversion (see Figure 16). Downconversion is best described as the frequency conversion process where a radio frequency signal is mixed with the frequency coming from a local oscillator to receive the intermediate frequency to better obtain the signal integrity of the system. This process is typically done when we are interested in demodulating the desired signal at hand. Up-conversion is the reciprocal process where you combine the IF signal with the signal from the local oscillator in the frequency domain and convert it to an RF signal.

Passive mixers generally are more popular to use since they are simple in design, have a relatively large bandwidth and have good intermodulation distortion performance. Active mixers are really only used in RFIC design. Generally, active mixers provide conversion gain, good isolation between ports and do not need as much power to drive the local oscillator. They are easily able to be integrated with signal processing circuits & are not as sensitive to load matching.

Mixer parameters to be considered are conversion loss/gain, 3rd order intercept, spurious behavior, isolation, noise figure, and dynamic range. We consider conversion

loss/gain to measure the active gain or loss in a mixer. This is defined as the relationship of the IF output power to the RF input power. Conversion loss is a very important parameter for passive mixers. 3rd order intercept is the RF input power where the unwanted intermodulation signals equal to the desired IF output power. Spurious behavior is seen as external signals that create interference in the IF range. Isolation looks into the amount of power leaked from one port to another. Noise figure is a characteristic where we see how much noise (or undesired random signal) is seen from the input to the output of the mixer. Dynamic range is the power range which the mixer operates ideally.

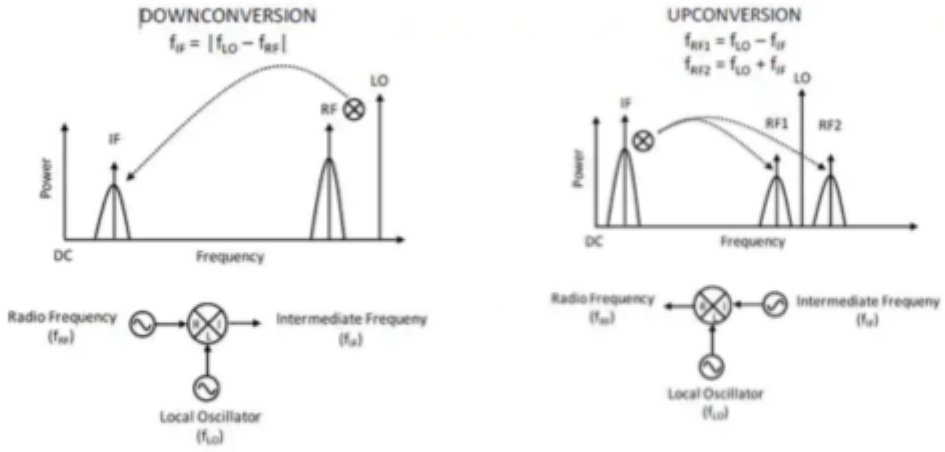

Figure 16: (Left) Down Conversion Using a Mixer;
(Right) Up Conversion Using a Mixer

In order to choose a mixer for our project, there are a few specifications that must be established. The first is the operating frequency of the mixer. Since there are three ports on a mixer, there are three frequency specifications that should be established (typically in terms of ranges). Our mixer needs to be able to handle an RF port frequency of our radar's receive signal, which may be anywhere between 2 GHz and 2.5 GHz. It is entirely dependent on the transmit frequency the team decides on. A wider range is ideal, but the primary goal is a lower operating threshold of at least 2 GHz.

Next, the IF port frequency must be established. We wish to convert the RF signal down to the order of 10 kHz. This means our mixer must be capable of outputting this frequency on the IF port. Consequently, the mixer's LO port must be able to take an input frequency nearly identical to that of the RF port for the differential to be on the order of only 10 kHz. Another important parameter is the required power of the LO port. Mixers often have specified minimum power levels at the LO port for the mixer to operate. If this power level is excessively high, it may strain the power source of the system. A lower LO power level is generally more convenient for small-scale applications such as our project. Additional considerations when choosing a mixer is the versatility of the module, which typically refers to whether the module can work both as an upconverter and a downconverter, and the conversion loss, which should remain as low as possible.

We have decided to use the mixer suggested by the MIT OpenCourseWare course, which is the ZX05-43MH-S+ frequency mixer by MiniCircuits. Its LO and RF port frequency range is 824 MHz to 4200 MHz, which covers the 2000 MHz to 2500 MHz range needed. Its IF port ranges between DC (zero frequency) and 1500 MHz, which is more than enough for our purposes. The LO signal power is +13 dBm, which is a bit high, but ultimately this will work out nicely considering the power levels already being generated by our circuit (as will be seen in the hardware design portion of the paper). The ZX05-43MH-S+ also has a 6.1 dB typical conversion loss, which is acceptable, and very good port isolation. All around it is a great choice for our frequency mixer.

3.4.5 Power Amplifier

An RF power amplifier is best defined as an electronic amplifier that is designed to take a low power RF signal and boost it up to a high power RF signal. Power amplifiers are known to drive the antenna of a transmitter generally. Important parameters for the design of a power amplifier are typically: gain, power output, bandwidth, power efficient, linearity, input/output impedance matching and heat.

Power amplifiers are the most power consumption heavy components in a RF transceiver and provide the most difficult design challenges within the system. Basic power amplifiers include Class A, Class B, Class C, as shown in Figure 17.

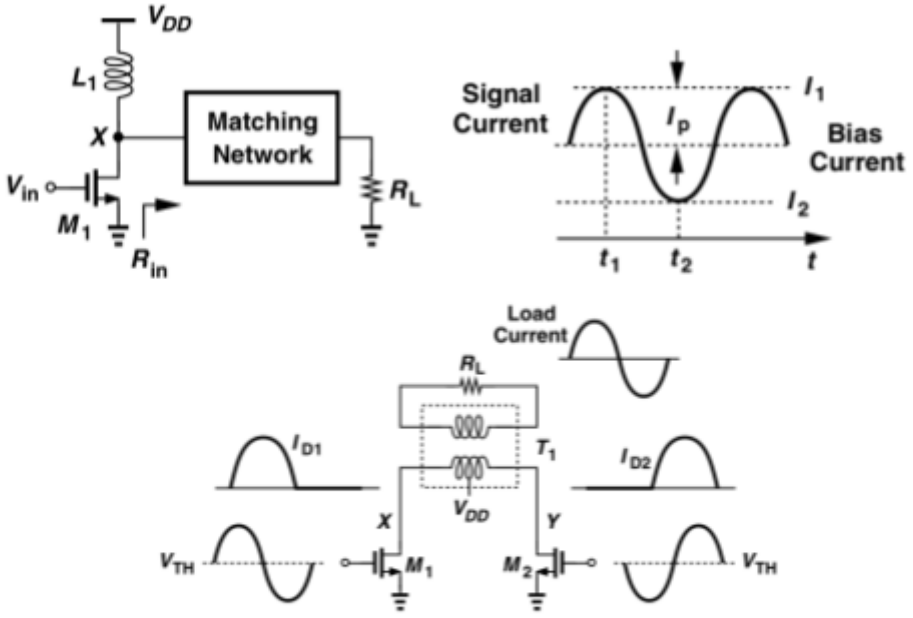

Fig. 17. Top left: Class A power amplifier. Top right: Class B power amplifier. Bottom: Class C power amplifier

Power amplifiers are often placed just before a radar's transmit antenna to give the signal a final boost, and that is the use of the power amplifier in our design. Like many RF signal processing components, the most important parameter of a power amplifier is its frequency range. In our project, we need to amplify the outgoing signal at its radio frequency, which at a minimum will be 2 GHz as previously mentioned. Anywhere in the 2 GHz to 2.5 GHz range would be ideal. The second most important parameter is the amplification factor of the power amplifier, which is expressed in decibels. The MIT reference documents have suggested that the required amplification is at least 10 dB. Not only must the amplifier have the desired gain, but it must also have the ability to actually output the desired power level. A power amplifier with 10 dB of gain may not necessarily be able to output the expected 20 dBm of power from a 10 dBm input. It is important to determine the necessary output power before selecting a component.

We will use the MIT suggested MiniCircuits low noise amplifier ZX60-272LN-S+. This device has an operating frequency of between 2300 MHz and 2700 MHz, and an average gain of 14 dB, which is plenty for our purposes. One important characteristic is that this is a low noise amplifier, which helps to set a low noise factor for the entire system. This is better than a typical power amplifier, which is designed primarily for its gain characteristics and tends to have poorer noise characteristics. Though this component is called a low noise amplifier, it is functioning as a power amplifier in terms of where it is placed in the radar system.

3.4.6 Oscillator

Oscillators are a fundamental part of the RF transceiver architecture. They are extensively used in both the transmit and receive paths of the system. An oscillator used in a RF transceiver must satisfy 2 primary sets of requirements: system specifications and interface specifications. System specifications relate to the frequency of operation and the signal integrity of the output. Interface specifications relate to the drive of the oscillator or the output swing. Figure 18 shows three different oscillator topologies.

RF oscillators should ideally be designed such that the frequency range can be tuned across a set range. The two sets of criteria to fit the frequency range would be, the system specification of the oscillator needed to go into the mixer and second would be an additional margin to cover process and temperature variations and errors due to modeling inaccuracies. Output voltage swing is a strong consideration as well for oscillators since they must be able to produce sufficient swings on the output to guarantee the switching of transistors for the consecutive stage. A buffer could be used to amplify/drive to drive the stages. Driving is also an important feature for oscillators as well since we would need them to be able to drive a large load capacitance. Capacitive loading is more serious in transmitters, since the input capacitance of a power amplifier can propagate to a local oscillator of upconversion mixers.

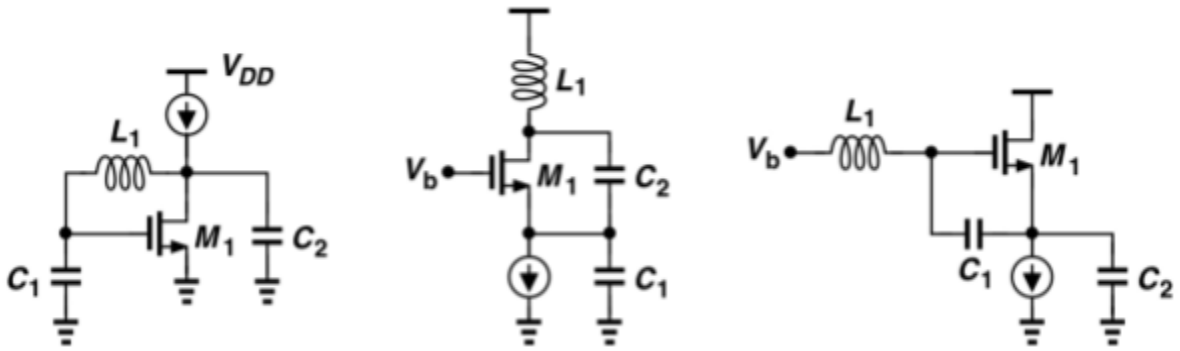

Fig. 18. Topology of 3 Different 3-Point Oscillators

Regarding the basic principles of an oscillator, an oscillator generates a periodic output. This creates a necessity for the circuit to have a system in place that allows its own noise to grow and then become a periodic signal. From the controls system point of view, an oscillator can be viewed as a system that is a “poorly designed” negative feedback amplifier. Baurkhausen’s criteria states that the following condition must be satisfied to ensure stability of the oscillator:

$$|H(s = j\omega_1)| = 1$$
$$\angle H(s = j\omega_1) = 180^{\circ}$$

The oscillator in our circuit may be the most important component in the radar module. Since it generates the transmitted signal, it is crucial that the correct component be selected. The frequency range must, of course, contain the desired transmit frequencies. The oscillator should also be able to output voltages across the desired range. The harmonics created by the oscillator should be kept to a minimum, as well as the DC operating power. Our team has decided to use the MIT recommended part, MiniCircuit’s ZX95-2536C+. This component has a frequency range of 2315 MHz to 2536 MHz, which meets the primary requirement of being higher than 2 GHz. However, the operating bandwidth of this device is the narrowest of all the RF signal processing components at only 221 MHz. This imposes a restriction on what frequency bands can be chosen for the radar’s operation. Though this component will suffice for project development, this would be an excellent area for the project to be improved in. Increased frequency range is generally a positive when it comes to radar modules.

3.4.7 Low Noise Amplifiers

A low noise amplifier (LNA), is an RF amplifier that is designed to amplify a low power signal without degrading the signal-to-noise ratio. In an ideal since it is what it should do, but it will add noise to the signal due to the noise figure of the integrated circuit itself. An LNA generally adds about 2dB or 3dB to the signal in noise. Gain is another critical parameter for the low noise amplifier since it must be large enough to minimize the noise of the following stages. This becomes a specific issue in downconversion mixers. The choice of this gain largely depends on a compromise between noise figure and then the receiver linearity. Common topologies for an LNA include those shown in Table 2.

The most commonly used topology is the common-source stage with cascode, inductive load and inductive degeneration. Table 2 and Figure 19 show some common topologies for a low noise amplifier.

Table II
LNA Topologies

Common–Source Stage with	Common–Gate Stage with	Broadband Topologies
▪ Inductive Load	▪ Inductive Load	▪ Noise–Cancelling LNAs
▪ Resistive Feedback	▪ Feedback	▪ Reactance–Cancelling LNAs
▪ Cascode, Inductive Load, Inductive Degeneration	▪ Feedforward	
	▪ Cascode and Inductive Load	

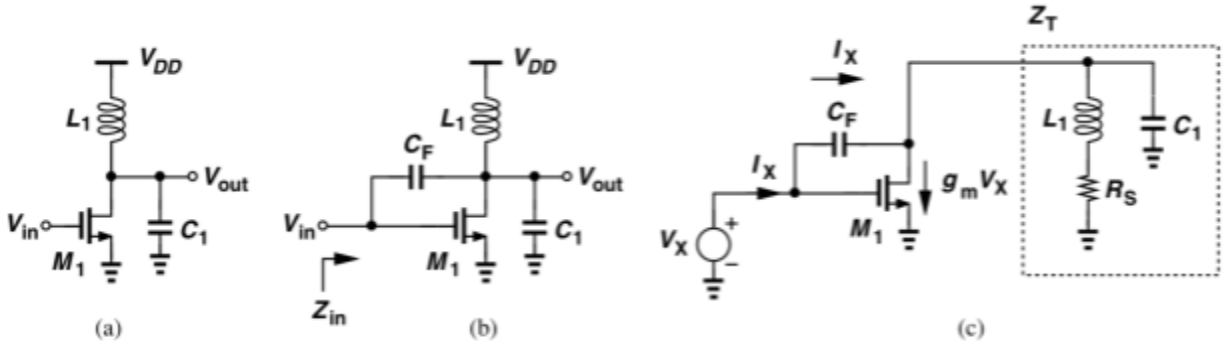

Fig. 19. LNA Topologies

There is a second amplification that often occurs in a radar system, and that is after the signal has been received. After traveling a distance, hitting targets and losing power upon reflection, the signal that was originally transmitted will have lost nearly all of its power by the time it arrives back at the receive antenna. Often there is just enough power to detect the most important characteristics of the signal. In order for the signal to actually be processed and analyzed, it needs to be amplified back up to a more detectable level. This is the purpose of placing a low noise amplifier directly after, or nearly directly after, the receive antenna of a radar. The purpose of using a low noise amplifier as opposed to a power amplifier is that at this point in the system, the focus is no longer on boosting the signal high enough to get the desired range or accuracy. The focus is creating a large but clean signal for proper analysis. Low noise amplifiers are ideal because they minimize the noise passed onto subsequent system components.

For our system, will we use a second ZX60-272LN-S+ amplifier as our low noise amplifier.

3.4.8 Analog-to-Digital Converters

The received analog RF signal must be converted to a digital representation in order to extract meaningful information from the signal. To achieve this an analog-to-digital converter must be utilized, which samples the signal and converts the voltage of the sample to a sequence of bits to be interpreted by the microcontroller and subsequently the host device. Key properties to be considered when choosing an analog-to-digital converter are the following: its full scale value (the maximum signal amplitude that the device can measure), its bit resolution (the number of bits used to represent the value in binary), and its sampling frequency (the rate at which samples are acquired). These properties may be visualized in Figure 20 below, which may have a full scale value of 7, a bit resolution of 3, and a sampling frequency of 1 Hz. In this case, the minimum representable value by the analog-to-digital converter would be $7V \,/\, (2^3 - 1) \;=\; 1V$.

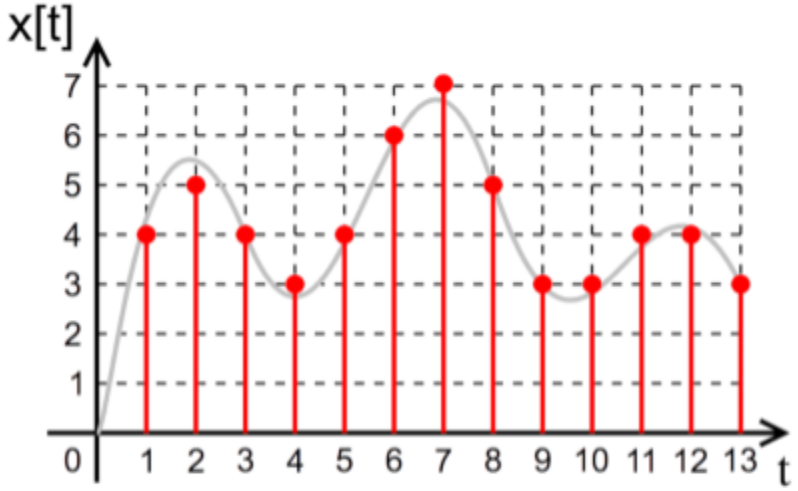

Fig. 20. The discretization of a time domain signal at 1 second intervals.

The analog to digital converter in our system will be used to turn the analog receive radar signal into a signal we can analyze in the host computer. The primary concern when it comes to converting analog signals to digital is having a sufficient sampling rate. If the sampling rate is too low, the signal will not be accurately represented in the digital domain. As a general rule, the sampling frequency should be twice that of the highest frequency component present in the signal. In the case of our system, the frequency of

the signal being processed will be no more than 20 kHz. This means the analog to digital converter must have a sampling rate of at least 40 kHz.

The original MIT implementation of our radar module used the sound card within a laptop to digitize the radar signal. To mimic this process outside of a laptop, we will be using a 24-bit audio analog to digital converter, the PCM1802 by Texas Instruments. It uses a delta-sigma conversion method and includes internal filters to prevent aliasing. Best of all, the sampling frequency can go up to 96 kHz, allowing for input signals of up to 48 kHz.

3.4.9 Microcontroller

The microcontroller is the gateway between the analog radar-side of the project and the digital host-side of the project. It is responsible for accumulating the digital samples generated by the analog to digital converter and passing those packets of information onto the host computer. The microcontroller also allows the radar module to be controlled by the host. For instance, capabilities such as frequency changes, switching the radar module on and off, or selecting alternative modes of operation requires that digital commands be converted in analog equivalents. Microcontrollers are a great option for this type of interface, as they are often equipped with analog and digital input and output pins. They can easily receive digital instructions from the host and translate those instructions into analog electronic signals that affect the radar operation.

There are many considerations when selecting a microcontroller, and like many electronic parts, there are often several possible options that will suffice for a given application. We have decided to use the Arduino Zero for our project. This development board is equipped with Atmel's SAMD21 microcontroller. The Zero was chosen for several reasons, including its number of general input/output pins, 48 MHz clock speed, 32-bit ARM Cortex M0+ Core and small size. All of these had an effect on the final decision, though should it be necessary, the Arduino Mega has been identified as a suitable replacement and can be used in the same way as the Zero for our project.

3.4.10 Operational Amplifiers

Both the gain stage and the low-pass stages consist of operational amplifiers, typically called op-amps. Op-amps have five terminals: the non-inverting terminal, the inverting terminal, the positive power rail, the negative power rail, and the output terminal. The positive and negative power rails make the op-amp an active device. The voltages at the positive and negative rail are the limiting thresholds of the output; some op-amps are not even capable of reaching those limits. Typically the absolute value of the power rails are equal, such as +12V and -12V. However, this is not necessary, and in our applications we will use ground as the negative rail.

Several different classes of circuits can be constructed using op-amps, including filters, summing circuits and amplifiers. These circuits could certainly be constructed from passive devices, but the design process is often lengthy due to the interdependency of the circuit stages. One major advantage of using op-amps is that one stage has no

direct effect on the following stage, aside from the signal being passed between them. The behavior of each circuit remains independent, allowing for quick design and integration of large circuits. An additional advantage of op-amps is their low cost compared to other devices that provide the same capabilities.

The two types of op-amp circuits used in our design are a gain circuit (or amplifier) and low pass filter. Although these two circuits are cascaded, their behaviors are independent of each other. The gain circuit has an output that is greater in amplitude than the input. Some op-amp gain circuit configurations are called inverting amplifiers because the gain is actually negative, meaning the output is larger in magnitude but the negative or flipped version of the input. In our design, we will be using a non-inverting amplifier configuration, so the gain is positive.

Filters designed using op-amps have a different focus; they are used to change the frequency spectrum of the input signal. Different types of basic filters are low-pass, high-pass, bandpass and band-reject. Each of these passes frequencies according to either one or two cutoff frequencies. Low pass filters allow frequencies lower than the cutoff, high pass filters allow frequencies higher than the cutoff, bandpass filters allow frequencies between two cutoffs, and band-reject filters allow frequencies outside two cutoffs. In addition to the type, filters can have different names according to their mathematical characteristics. Types of filters in that sense include such as Butterworth, Bessel, and Chebyshev, where each of these has a characteristic shape of the filter's frequency response.

Cascading filters is an excellent way to improve the accuracy of the filtering. Cascading two low-pass filters with the same performance will decrease the presence of frequencies beyond the cutoff more so than if a single filter had been used. In our design, two second-order low-pass filters will be cascaded to create an effective fourth-order filter.

The operational amplifier used in our design will be the LM324N, a quad operational amplifier, meaning there are four separate op-amps powered by identical power rails. Using this device will reduce the space needed on the PBCs. The power rail for the device can span from +3V to +36V, which is more than sufficient for our application.

3.5 Complete System Block Diagram

Figure 21 shows a block diagram of our entire system. The design shown can achieve every goal outlined previously, though certain elements may not be needed for any given goal.

The design has been divided into three different subsystems: the RF system, the RF data processing, and the implementation. The RF subsystem is entirely analog. It begins with input DC voltages, which are converted to electronic signals with a defined shape, amplitude, and frequency. This signal is converted to an RF signal that passes through a series of devices via SMA-to-SMA connectors. The signal is sent out into the

environment through the transmit antenna and reenters the system through the receive antenna. It proceeds through two additional RF signal processing components before entering two electronic circuits, the gain stage and low pass filter.

The oscillator, attenuator, amplifiers, mixer, and splitter are all single, pre-assembled block components with internal circuitry. The modulator, gain stage and low pass filter are all circuits that will be designed by our team and assembled on PCBs.

The next subsystem is the RF data processing. There are two steps to the RF data processing. The first step is to convert the output signal of the RF system from analog to digital so it can be sent to the host device. The use of a USB2 connection, implemented through the microcontroller (MCU), is a part of the signal transmission to the host device. The second step in the RF data processing is implemented in software. The raw data is extracted from the digital, time-domain representation of the signal by using Fourier frequency analysis techniques. The resulting analysis can then be used for the implementation.

The concept of the implementation subsystem is the display or use of the information gathered in the RF data processing subsystem. For the first two project goals of acquiring velocity and ranging readings, the only necessary aspect of the implementation is to display the readings on the host device monitor. For the third project goal of using the information gathered to drive a control system, hardware may be used in the implementation subsystem such as motors, LEDs, or other devices.

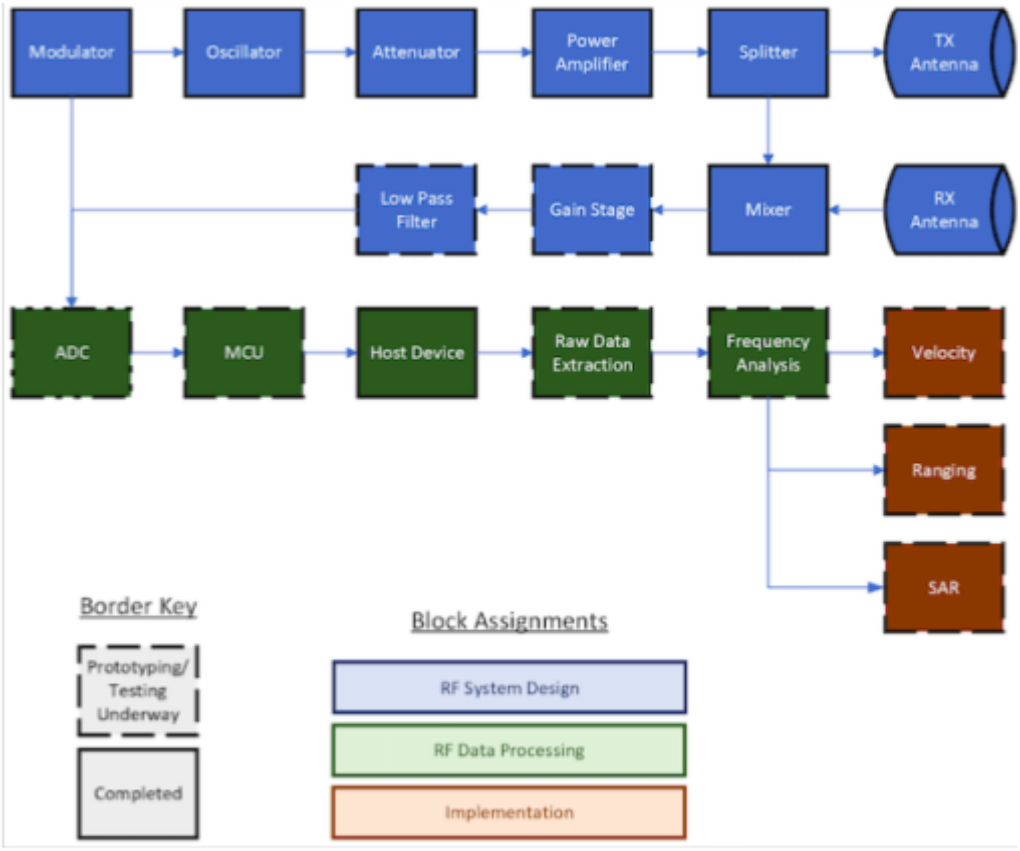

Fig. 21. Overall project block diagram.

In summary, our RF system design chain will consist of a two-port cascaded network model containing modulator, oscillator, attenuator, amplifier, and splitter. This signal is then sent through the transmitting antenna. The received signal is then sent through a low noise amplifier and finally through a mixer, where it is compared with the transmitted signal. Our part selection for this RF component chain will be centered around our requirements for the above sub-systems.

Once the signal is sent through the mixer, it is sent through the gain stage and the low pass filter. This filtered, amplified signal is then sent out to the ADC. The ADC sends the digitized signal through the MCU to the host device via USB. The host device processes and analyzes the signal. The information obtained can then be either displayed or utilized in an external system.

One area in which our project may be improved in the future is the host device. As of now, the host device is a laptop like that used in the MIT radar system. Ideally the host device would be included in the radar module itself; this could easily be accomplished through the use of a microprocessor.

4.0 Project Constraints

4.1 Standards

Aligning with standards ensures our project will be easily compatible with existing products and components on the market, as well as remain safe and reliable during operation. Though not entirely complete due to the vast number of standards in existence, the following list covers the standards that are most relevant to our project design.

4.1.1 Hardware Standards

4.1.1.1 IEC 60086-1 Primary Batteries - General

Our radar module will be portable, so our team has decided to power it using 9V batteries. The IEC Primary Batteries standard outlines several important characteristics that should be consistent among general primary (single-use) batteries. Some of these characteristics are physical, such as the dimensions of the battery, while other characteristics are electrical, such as the maximum voltage each battery cell should supply. The most important aspect of this standard is that it allows for batteries from different manufacturers to be used interchangeably, eliminating the need to consider differences in function or form [1].

There are a few implications this standard has with respect to our system design. The first is that the physical size of the batteries is predetermined. Because 9V batteries have standard dimensions, the battery holder we select must be the correct size and shape. The way we choose to physically assemble our final integrated system must also take the standard size of 9V batteries into account. Another implication is that the input

voltage levels we have to work with are finite. Each new 9V battery will have a nominal voltage of +9V, so the only available input voltages are k*+9V, where k is any positive integer. Because of the major implications this has on our power supply circuit design, it has been taken into account prior to selecting the type of batteries the system runs on. The section on the power supply circuit will further discuss the reasoning behind choosing 9V batteries for our system.

One additional thing to note with regard to our system's batteries is that our system can run on either primary or secondary batteries. If secondary batteries are used, there are additional standards that would apply to their safe use and storage.

4.1.1.2 ANSI C18.1M – Portable Primary Cells and Batteries with Aqueous Electrolyte

This standard is similar to IEC 60086-1. It outlines several standards for portable primary battery cells, including information on safety standards. An interesting subsection of the standard is consideration of potential misuse. Things such as installing batteries backwards, dropping batteries and over-discharging batteries are all addressed [2].

Almost all batteries, even common household batteries such as 9V batteries, come with instructions on how to properly use them. Our team must be sure to abide by these rules because the consequences of misuse have already been tested for and established.

4.1.1.3 IEC 61169-15 – Radio Frequency Connectors (SMA)

There are many types of radio frequency connectors. All of the connectors are used to transmit RF signals, but equipment such as spectrum analyzers and components such as RF frequency mixers come with certain connectors already embedded. If a device has a certain connector already on it, the complementary connector must be used to securely attach a cable or another device. For instance, if a device has a male N-type connector, a cable with a female N-type connector must be used to access the device port.

The IEC 61169-15 standard outlines the physical dimensions and quality parameters for the SMA connection [3]. The high frequency components of our radar module all come with female SMA connections, so in order to connect them, our team needs to ensure we use male SMA to male SMA cables or barrels. Using other types of connectors will not ensure a solid electrical connection.

The SMA standard is also important for the design of our PCB. One PCB in our design will use an SMA connector for one of its input signals since it must interface with a device that uses an SMA connector. Knowing the size and construction of a standard SMA connector will partially dictate the design of the PCB, as the size and electrical characteristics of the SMA connector must be taken into account.

4.1.2 Software Standards

Programming language and other software standards are a bit different from hardware standards, especially when it comes to the consequences of diverging from them. Imagine trying to insert an SD card into a microSD card port; it simply would not fit, and the lack of adherence to the standard would be quite obvious before the system is constructed and tested. But in software, failing to align with standards may have consequences that are not apparent until after system implementation.

An example of this is programming language compilers. They typically will either give an error or a warning when users diverge from standards, such as not using correct syntax. But computers are not mind-readers. Even if code technically aligns with standards, it is possible that the functionality of the code is not what the user intended because the user did not align with standards before attempting to use them.

Our project has two subsystems that use software, the USB interface and the API. Standards specific to USB have been listed in the Hardware/Software Hybrid Standards section, but because our USB interface uses a programmable microcontroller, standards related to the language used to program the microcontroller have been included in this section.

4.1.2.1 ISO/IEC 14882:2020(E) - Programming Language C++

Programming language standards are essential to the development of our project. The most important element in our system that utilizes a programming language is our API for processing the digitized received radar signal. There is no one programming language that is required for our API implementation; we are free to use whichever language best suits our needs in terms of computation speed, ease of use, architecture and available libraries. However, once a programming language is selected, there are standards such as ISO/IEC 14882:2020(E) that dictate how code in that programming language should be written. Which programming language we select may also dictate which available standards we can (or should) align to with respect to our specific application (which is radar signal processing).

We chose to use C++ for our API primarily because of its speed and streamlined architecture. According to the ISO website, the ISO/IEC 14882:2020(E) C++ standard covers the following: lexical conventions, basics, expressions statements, declarations, modules, classes, overloading, templates, name resolution, exception handling, library introduction, among several other more complex topics [4].

Everything covered in the C++ standard can be thought of as a toolbox with a set of rules for using the tools. As long as we align with the rules, we can design our code to do almost anything we like.

4.1.2.2 ISO/IEC 9899:2018 – Programming Language C

The standard for C is exactly like the standard for C++ in the sense that it defines a toolbox with rules for using the tools [5]. To a certain extent, the code we develop in one language can be modified to align with the standards of the other language. It is important to note that processing capability is often added between iterations of programming languages, such as from C to C++ or from Python 1.0 through Python 3.0. Selecting a more advanced programming language may imply the standards are more extensive in content, but it likely means more capability as well.

Our team could choose to utilize the C language in an alternate implementation of our API, but this standard is included here primarily because of the language used in the Arduino IDE.

4.1.2.3 Arduino Language Standard API

In our project, the microcontroller used in our USB interface will be programmed using the Arduino IDE. This IDE utilizes an unofficial language that is a hybrid of C and C++. The Arduino website provides the functions, variables and structure elements that are used in Arduino code, with many aspects of the code aligning closely with either C or C++ [6].

This standard affects how we write the code to program our microcontroller. We could opt to use an alternative tool to program the microcontroller but using the Arduino IDE would simplify the development immensely and make it easier to modify the functionality later on.

4.1.2.4 Ripple Radar API Standard

One way to ensure our API is easily adaptable by the end-user is to use a standard API as the baseline code. The Consumer Technology Association (CTA) hosts Ripple, an open radar API standard released in January 2022 [7]. According to the CTA website, Ripple has been especially designed for general purpose consumer radar and is meant to enable hardware and software interoperability for radar applications. Ripple can be thought of as a more specialized version of a software standard; it defines special functions and structures that must be used in a certain way, but which can be combined to create certain functionality in a project. Choosing to align with this standard facilitates software development for our project by making it more straightforward to extract information from our radar module. Ripple also makes it easier to adapt code for other radar modules or for other software designers to understand and later modify our API.

4.1.3 Hardware/Software Hybrid Standards

4.1.3.1 IEC 62680-2-1 – USB Specification

Our team wanted an interface for our project that would be compatible with most laptop computers. There are a limited number of port types used for inputting data to a laptop,

the most common being USB, audio, ethernet, and SD/microSD. We chose USB because it is easiest to implement from a technical perspective and also a common port for laptops.

Standard IEC 62680-20-1 is a very extensive document, covering nearly all aspects of a USB interface both on the hardware side and the software side. According to the standard preview provided by IEC, topics discussed include USB architecture, the data flow model, considerations for isochronous transfers, the mechanics of USB assemblies, electrical characteristics, protocol layering, device framework, USB host requirements, and hub specifications [8].

This list alone indicates that the USB standard imposes many restrictions on our project interface design. Manufacturers have already aligned their laptops with the standard both in hardware and in software, so even a slight change such as a slightly bigger wire housing will render our interface essentially unusable. Though these restrictions leave little room for freedom of design, aligning to the standard actually facilitates the development of our system's USB interface. So long as we align with the standard, any laptop with a USB input port will be compatible with our interface, and the laptop can therefore be used to receive and process signals from our radar module.

4.1.3.2 SCPI-99 – Standard Commands for Programmable Instruments

The Standard Commands for Programmable Instruments (SCPI) standard provides a structured means of controlling electronics test equipment and automatic test equipment [9]. SCPI originally sat atop the IEEE 488.2 General Purpose Interface Bus (GPIB) specification which dictated how bytes of data were communicated between these pieces of test equipment and computers, but did not provide a rigid command syntax for making measurements. In this sense, it was possible for two different devices to employ the IEEE 488.2 specification to communicate with a computer, but they may have had completely different sets of commands to make measurements. This greatly complicated the testing process, where many pieces of test equipment may be connected to a single computer by way of bus topology. Examples of such standardized commands include the following (where capitalized letters are required and lower case letters are optional): "MEASure:VOLTage?" to measure the average voltage at a test point with a multimeter; "SOURce1:FREQuency 100" to set the frequency of channel 1 on a function generator; "MEASurement1:MAIN FREQuency" to select the frequency as the measurement of interest of channel 1 on an oscilloscope; and "MEASurement1:STATistics:VALue:ALL?" to retrieve all of the statistics of the aforementioned measurement, which includes the peak & trough values, the standard deviation value, and the average value.

4.1.3.3 VISA - Virtual Instrument Software Architecture

The aforementioned SCPI standard was originally intended to work with the IEEE 488.2 GPIB interface, however newer bus interface developments have occurred throughout the years. The Virtual Instrument Software Architecture (VISA) [10] specification defines an Application Programming Interface (API) that enables communication with a wide

variety of pieces of test equipment using a multitude of different interface standards, such as GPIB, USB, and TCP/IP. Users can leverage this API to write programs that communicate with test equipment by issuing SCPI commands.

4.2 Constraints

4.2.1 Federal Communication Commission

Any project involving wireless transmission should verify compliance with Federal Communication Commission (FCC) regulations. The FCC regulates what, when, where, at what power, and at what frequency we are allowed to transmit signals in the U.S. The reason for wireless transmission regulation is that devices are tuned to receive at certain frequencies. If signals other than the expected signal are being broadcast at the same frequencies, interference is created, and depending on the application, the consequences can range from being merely inconvenient to having a drastic effect on national security. To ensure our project adheres to federal regulations, we must understand exactly how the FCC regulates wireless transmission.

The basis for regulation is dividing the frequency spectrum into bands. Bands are then allocated for specific purposes such as radio stations, military communication, or cell services [11]. In order to broadcast frequencies within a certain band, permission must be granted by the FCC. In most cases a license must be obtained, but ISM bands have also been allocated. ISM bands are for Industrial, Scientific and Medical use license-free. Consequently, the bands are often used for hobbyists' projects and for short-range communications such as Bluetooth [12]. Our project will use the 2.4 GHz ISM band to avoid going through the tedious process of obtaining a license from the FCC.

Although a license is not required to transmit in the 2.4GHz band, there are still restrictions that must be adhered to. These requirements include parameters such as maximum transmitted power and number of channels for channel-hopping devices. The restriction we are most concerned with is the transmit power. According to the FCC, the maximum transmit power allowed in the 2.4GHz band is 1W [20]. This must be taken into consideration when designing our RF subsystem. A higher transmit power is ideal according to our project's house of quality, but the maximum power is capped by the FCC regulation.

4.2.2 Time Constraints

A major constraint in our project is time. The course policies have outlined at what time the final project should be completed, which is April 2023. There are also several reports due periodically up until the project completion date, at which time certain components of our project should be completed. The planning and design work, and corresponding documentation, should be completed by December 2022, and the prototype construction and testing should be completed next semester. But simply

looking at the deadlines set forth by the course policies is insufficient to completely understand the time constraints placed on the team.
An important factor to consider in any project is the dependencies between tasks. Completing three independent tasks that each take one month does not imply the same time constraint as completing three tasks of the same length, but with the condition that one task must be done after the other. This shows that additional time constraints are created by the dependencies between tasks in a project. Our project's subsystems can be designed and developed independently up to the point of integration, at which point a setback in one aspect of the system will cause a setback for the entire integration process. Ultimately this can extend the anticipated completion date of crucial, end-of-project tasks such as the final demonstration. The team must work to not only get tasks done in a timely fashion but also take into consideration task dependencies. It is these dependencies that will drive prioritization.

In addition to time constraints created directly by the course calendar, everyone on the team has other commitments outside of Senior Design. We may technically be allowed four weeks to complete a certain task according to the course guidelines, but in reality, our team members may only be able to commit two to three weeks' worth of time actually working towards completing that task. Additionally, unexpected setbacks such as supply chain issues may delay task completion. This is why it is imperative that our team aim to complete tasks at least a week before the official deadline; two weeks before the deadline would be ideal for more important tasks.

In order to keep to a plan that is well ahead of schedule, our team should consider the effects supply chain issues may have on our hardware acquisition. Certain suppliers may have long lead times for crucial elements in our design, or they may not have our preferred item available at all. Shipping times can also be quite lengthy when ordering hardware or PCBs. It is the team's responsibility to check lead times, shipping times and product availability for multiple suppliers to determine which can deliver our items as quickly as possible. This is especially important if changes are made to the design late in the project.

4.2.3 Economic Constraints

Because our team has been donated a partially completed radar, we have been relieved of the bulk of our project's cost. The radar signal processing hardware such as the mixer, splitter and oscillator, in addition to the many SMA connections, are the most expensive components of the design, especially considering that nearly half of our project is implemented purely in software using free applications for development. However, in preparation for the worst -case scenario in which we no longer have access to the donated parts, our team must consider the maximum extent of our economic resources.

We want to keep the overall cost of our project below $800. This includes the cost of parts for the prototype, taxes, shipping, handling and any additional parts purchased for the sake of testing our initial designs (such as for breadboard tests). Because we have

no sponsor aside from Dr. Gong's part donation, our budget implies a maximum contribution of $200 per team member to complete the project. We must keep costs within these bounds, and if necessary, we can modify our design or source parts from suppliers who offer cheaper prices than our preferred suppliers.

The supply chain issues mentioned in the Time Constraints section are, at their root, economic constraints. In our current market, there is still a shortage of certain electronic components. When these items have high demand and low availability, it drives prices much higher than they would be under normal conditions. The team's first choice of parts or suppliers may turn out to be unreasonably expensive, so alternatives may need to be selected.

In addition to the cost of the actual project itself, there may be costs associated with equipment. Our team will use many different pieces of equipment to develop and test our subsystems, particularly when it comes to verifying that our circuits are operating as desired. However, test equipment for electronics is generally quite expensive. Considering the cost of RF systems is already high, it is unlikely the team will have the budget to purchase test equipment specifically for the purposes of our project. We should minimize our costs by utilizing the test equipment provided to us by the University of Central Florida, or at the very least evaluating such equipment before deciding to purchase new equipment for our project.

4.2.4 Health and Safety Constraints

There are a few things that must be taken into consideration when performing RF transmission experiments. Electromagnetic radiation can cause biological harm under certain conditions if the proper precautions are not taken. There are three parameters that determine the safety factor of a particular RF experiment. These are the transmitted power, the transmitted frequency, and personnel proximity.

From a technical standpoint, high-power RF systems are used to achieve high resolution or to reach targets at extended distances. However, the introduced risk is that organisms in the presence of the high-power energy could potentially be harmed. The amount of energy absorbed by a person or animal in the presence of RF energy is typically measured in watts per kg (W/kg) or milliwatts per gram (mW/g) [13]. If this measurement is too high, damage to body cells may occur causing serious medical issues.

Another significant parameter in radar systems is the frequency used. Much like power level, frequency is catered to the specific radar application. High frequencies are better suited for speed and high resolution, while low frequencies are best for range. According to the FCC, humans absorb RF energy at a maximum rate when the signal frequency is 70 MHz; this is why RF safety standards typically apply to frequencies between about 30 to 300 MHz. When in these ranges of operation, it is crucial to consider the effects the transmitted signal may have on nearby people or animals.

Note that personnel proximity is an important factor when considering safety implications of both frequency and transmitted power level. Even if an RF system is transmitting at around 70 MHz and at a relatively high power, if there is nobody within the range of the radar system, there poses very little if any risk of causing harmful biological effects.

Currently, our project is designed to operate at 2.4 GHz (in an ISM frequency band) and will only transmit 10 mW (10 dBm) of power. This means there is very low safety risk, and pretty much any location away from sensitive lab equipment or people with high health risks should be suitable for conducting our tests. However, it is very important that we verify that our system is operating as intended before performing field tests. If the system is actually transmitting at a different frequency or higher power level than desired, there are significant safety implications.

Aside from the RF safety factors of our system, there is a safety concern regarding the thermal characteristics of our design. Electronic systems, especially poorly designed power systems, have the potential to heat up very quickly. Normally the worst-case scenario in these situations is component failure and ultimately damage to the device, but for our project, the PCBs and components will be mounted onto a wooden board. A potential consequence of allowing any of our components to overheat is the module catching fire! Not only would this be unacceptable from a technical perspective, but it would put the consumer in danger. In future iterations it may be a good idea to research alternative materials to construct the module frame out of, but as our team has been given a partially completed system with the wooden board already integrated, our objective must be to design our system in such a way that prevents components from overheating. PCBs often have heat sinks added for components that tend to heat up, and selecting circuit elements wisely can prevent components from heating up in the first place. These points are especially important when it comes to designing our power supply circuit, which is the circuit with the most dynamic thermal characteristics.

4.2.5 Manufacturability Constraints

If our team were to design the physical form of our radar module entirely from scratch, we could choose between 3D printing the module body, ordering a pre-existing frame or using other approaches. We would then need to decide where all of our system components would be placed and how they would be secured to the module frame. However, because a partially built radar has already been provided to us, our task is somewhat simplified.

The radar module consists of two antennas and several electronics components, some of which are circuits packaged in small, box-shaped housings and some of which are “loose” electronic components such as resistors, capacitors and operational amplifiers. Ultimately, all of these components need to be mounted on the radar module and connected together. The module we have already has several components, including the antennas, mounted onto a wooden board, so now our team’s only task is to mount the remaining “loose” elements of the design onto the pre existing frame.

It is a course requirement that our project involves a PCB, so more likely than not, all the circuits in our system will be placed on a PCB. The task is then simplified; we just need to make sure the PCBs are of the appropriate dimensions to fit on the pre-existing module frame. If we so choose, rather than securing our PCBs directly to the wooden board, we could 3D print customized housings for the PCBs to fit snugly into and just secure the housings to the board. In this case, we would need to consider the additional weight and/or added dimension from the frames. As it stands, though, the weight and dimensions of the final system will be little changed from the weight and dimensions of the partially built radar we have been provided.

Fitting the PCBs onto the radar board will not just be a matter of putting puzzle pieces together. Each circuit on a PCB has certain inputs and outputs. It would not be wise to place a PCB all the way to the left of the board if its input is coming from the right side of the board. Additionally, the power supply circuit will be used to power every other circuit on the board, so it needs to be placed strategically. These things will be considered during the PCB design rather than after the fact. It will avoid an unnecessary complex assembly process when it comes to constructing our prototype.

5.0 Hardware Design

5.1 Power Supply Circuit

There are three supply voltages required for our system to operate: +12V, +5V, and +3.3V, all DC generated from alkaline batteries. The +12V supplies the modulator, which pulls 12mA. The +5V supplies the op-amp positive rails in the gain stage and low pass filter circuits, the oscillator, the power amplifier and the analog-to-digital converter, which collectively pull about 125mA. +3.3V supplies the microcontroller in our system, and its current draw is highly dependent on the application and which pins are being used. Although our current design has the MCU running on power from the host CPU, the power supply will be designed for the worst case scenario in which the microcontroller pulls 200mA from the batteries. +3.3V is also used for the analog-to-digital digital supply voltage and the offset voltage in our op-amp circuits, though these contribute negligible current draw. These current and voltage requirements will drive the design of our power supply circuit.

5.1.1 Battery Selection and Configuration

Because we know the necessary supply voltages for our system, we can select which type of batteries we want to design for. There are three important considerations when selecting the batteries. The first is that the battery type should be easily accessible to the consumer. The second is that the number of batteries in total should be kept to a minimum. The third is that battery depletion must be kept in mind. Our team has decided to use buck converters for efficiency purposes, so choosing batteries that add up to exactly the needed voltage levels would be insufficient. The battery voltages would soon deplete to just below their nominal levels, and then the buck converters

would not be able to supply the voltages needed to power the components in our system. In consideration of all these points, our team has chosen to design around the use of common 9V alkaline batteries.

9V batteries are affordable and accessible to the general consumer, which is one of our desired system characteristics. In order to generate the three voltages required from 9V batteries, our design uses 4 batteries. First, two batteries are placed in parallel to increase the total capacity. Each battery has approximately 600mAh of capacity alone, and when batteries are placed in parallel their total capacity is equal to the sum of their individual capacities. Our design then generates a higher voltage by placing another pair of +9V batteries in series with the first pair. This doubles the available voltage to +18V. This means our design contains four +9V batteries in total. There is one +9V node, one +18V node, and approximately 1200mAh capacity for each node. The battery configuration is shown in Figure 22.

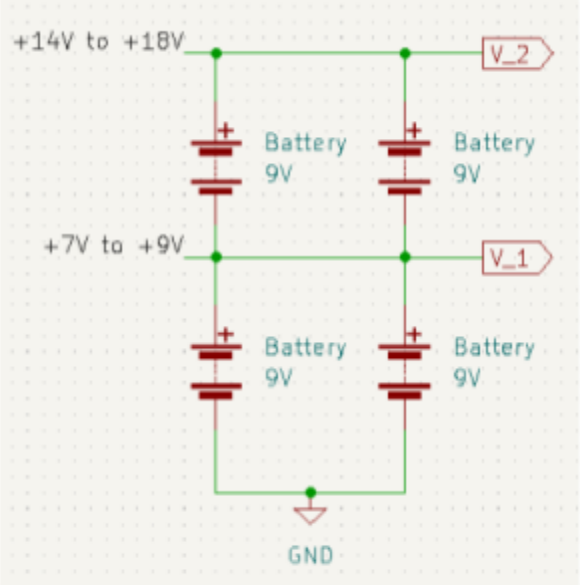

Fig. 22. Power supply battery configuration.

Now that our batteries and their configuration have been established, the circuit design can be discussed. Our goal is to achieve an output of +12V, +5V, and +3.3V regardless of fluctuations in the batteries' voltages, at least up to the point where the batteries would simply be considered dead. The tool our team used to design our power supply is Texas Instruments' WEBENCH tool. It takes in the desired output voltage, desired output current and the range of input currents, and outputs a variety of possible power supply circuits that could be used to achieve the desired goals. The chosen design is a balance of cost, PCB footprint and efficiency.

5.1.2 +3.3 V Power Supply

For our +3.3V circuit, the input voltage may range between +9V (when the battery is full) and +7V (when the battery is nearly dead). And as mentioned before, the output current at +3.3V needs to be 200mA in the worst possible case. Figure 23 below shows the chosen design for the generation of +3.3V from an input voltage of V_1. The key component in the design is the TPS54233 by Texas Instruments, a +3.5V to +28V input, 2A step-down DC to DC converter. It takes in the input voltage of between +9V and +7V produced by the first pair of batteries and reduces it down to +3.3V for our system components.

The two resistors tied to the PH and VENSE pins of the converter determine the output voltage of the circuit. Equation X gives the relationship between the circuit's input and output voltages. V_{ref} is +0.8V, R_1 is the resistor tied to PH, and R_2 is the resistor tied to ground.

$$V_o = V_{ref}\left(\frac{R_1}{R_2} + 1\right)$$

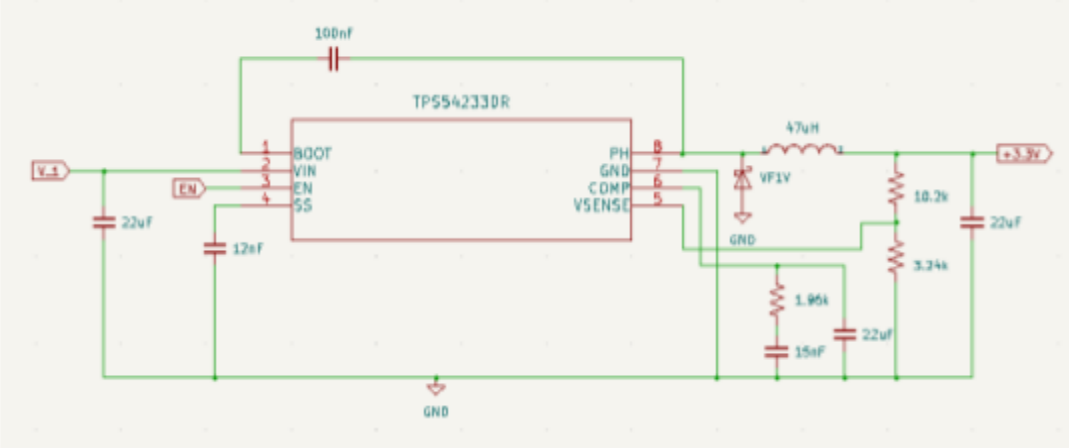

Fig. 23. +3.3V power supply circuit.

The circuit uses an input decoupling capacitor to smooth out any ripples that may be present in V_1. The capacitor connected to the SS pin is used for setting the start-up time for the circuit; using the equations available in the converter datasheet, a 12nF capacitor results in a start-up time of 5 milliseconds, which is within the recommended range of 1 millisecond to 10 milliseconds. The components at the COMP pin function as external compensation for the converter. TI's WEBENCH tool automatically generates the appropriate values for these components, though they can be manually calculated using the corresponding datasheet equations. An additional inductor and capacitor are used for output filtering, and the remaining components are required per the design of the converter.

5.1.2 +5 V Power Supply

A similar approach will be used to design the +5V supply voltage circuit as was used for the +3.3V circuit (see Figure 24). Note that the input voltage is still coming from the first pair of +9V batteries, so the input is still potentially varying between +9V and +7V. The only difference is that, at a maximum, the amount of current being drawn at this voltage level is only 150mA rather than a potential 200mA for the +3.3V supply. Regardless, because the desired output voltage and current are still within the range of the TPS54233DR, it will be used again for the +5V power supply.

The only differences between the components in the +5V supply and +3.3V supply circuits are the value of the resistor tied from VSENSE to ground (which sets the output voltage), the value of the output capacitor, and the components at the COMP pin.

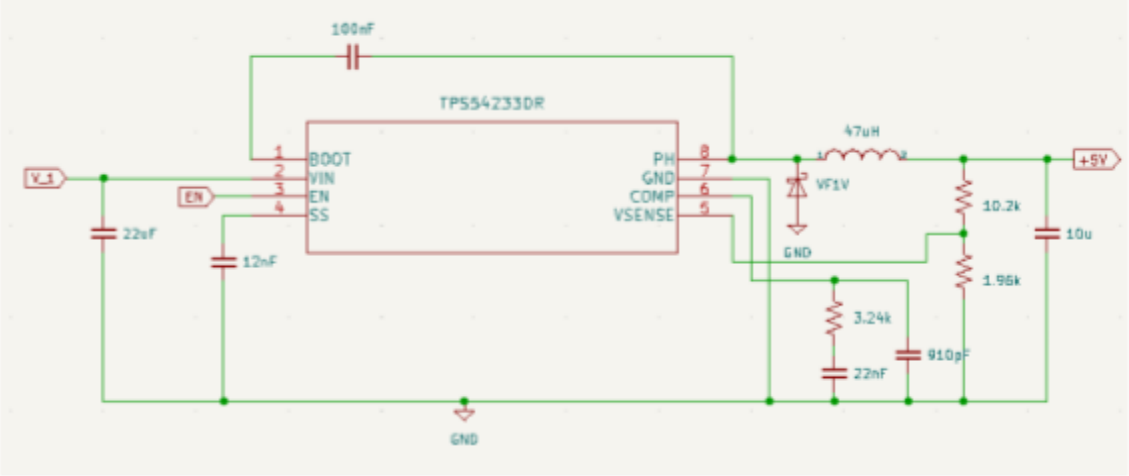

Fig. 24. +5V power supply circuit.

5.1.3 +12 V Power Supply

Lastly, there is the circuit for the +12V supply. This circuit will use the voltage generated from using both pairs of batteries and requires only 12mA to be output. Because two pairs of parallel batteries are being used instead of one, the voltage can vary more as each pair of batteries begins to deplete. If each pair reduces to only +7V, then the total voltage supplied will only be +14V. Though this particular scenario is unlikely considering the small current being drawn at the +18V node, it must be taken into account when designing the circuit. Figure 33 shows the results of the WEBENCH design search. The key component is the LMZM23600 step-down power module. It has a higher output voltage capacity of +35V but a lower maximum output current at 0.5A. This is perfectly suited for the +12V supply since the primary concern is the output voltage, not the output current. Note that in Figure 25, the input voltage is coming from the V_2 battery node.
x

The TPS54233DR DC to DC converter is used once again as in the +3.3V and +5V power supply circuits, and the same components are modified as discussed in the +5V power supply design. The voltage divider at VSENSE and PH pins are adjusted to generate a +12V output, and other components are adjusted as necessary to account for the change in output voltage and current.

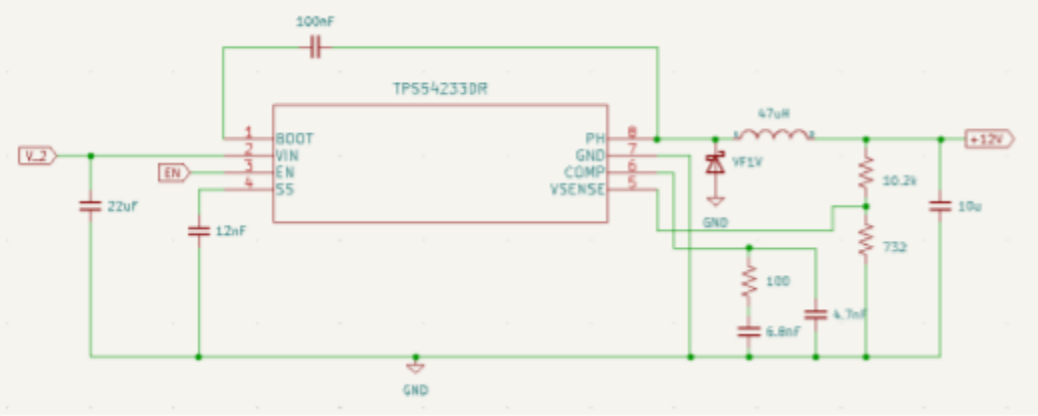

Fig. 25. +12V power supply circuit.

5.1.4 Complete Power Supply Circuit

The complete power supply schematic is shown in Figure 26.

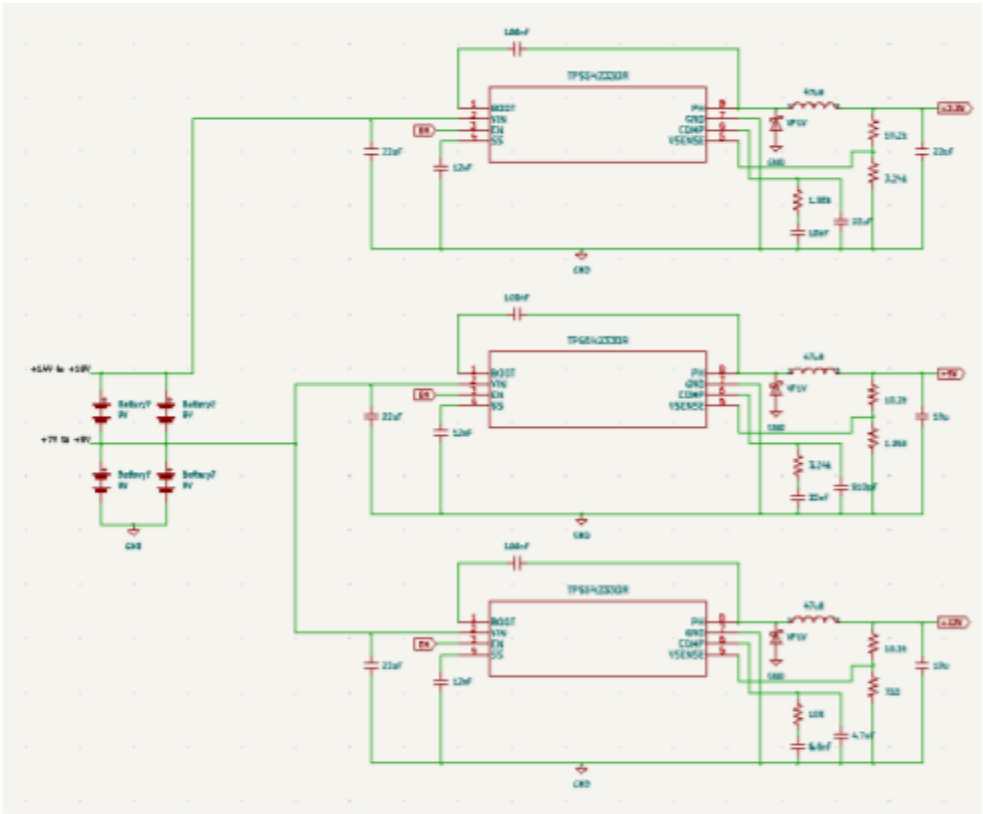

Fig. 26. Complete power supply circuit.

5.1.5 Power Enable Functionality

One additional capability of our system is a power-down function controlled by the host device and implemented through the MCU. Note that there are pins denoted as “EN" on each of the three voltage regulation devices used in the power supply circuits. The EN pins are active high, which means that the voltage regulation device only generates an output when there is a high voltage on the EN pin. For the TPS54233DR, voltages over +1.25V are considered sufficiently high. Alternatively, the pin may be left floating to keep the device enabled. If the EN pin’s voltage is less than +1.25V, the device generates no output. This feature will be used to power the radar module on and off via host control.

The MCU used to process the ADC data in our system will also be used to drive the EN pins in the power supply. When the host device gives the command for a system power-up, the MCU will change the output of a designated digital pin to high. For powering down, the MCU digital output will remain low.

Note that in our particular implementation, the MCU is running off of the host device’s power supply. In the case that the MCU is running off of the +3.3V battery supply, the +3.3V voltage regulation device would need to be constantly kept on. Its EN pin could be kept floating or tied to the V_IN pin of the device.

5.2 Modulator Circuit and Oscillator

There are two basic applications of our radar module: doppler (for measuring velocity) and ranging. For doppler, the transmitted signal only needs to be a single frequency. For ranging, the frequency is linearly swept in a triangle-wave pattern. In our project, the software can handle both applications using the triangle-wave pattern, but for the purpose of explaining the oscillator operation, using both a constant frequency and a varying frequency will be discussed.

The oscillator has a linear input-output relationship between the voltage at its Vtune pin and its output pin. If the effects of other parameters such as temperature are ignored, there is only one possible oscillator output frequency for any given Vtune voltage. This means that for a doppler application where only one output frequency is needed, Vtune will remain constant. The correct voltage would be determined from Figure 27, which shows the relationship between Vtune and the oscillator output voltage at 25 degrees Celsius.

A higher frequency is ideal for maximum resolution in doppler applications. The highest frequency in the ISM band our project operates in is 2.495 GHz, so for a simplified configuration that achieves the highest possible velocity resolution, Vtune could be tied directly to +3.2V. For our design, a frequency modulated signal is used primarily for the ranging application, but the software is able to account for the frequency modulation and still produce velocity readings. As a result of this design choice, the oscillator’s output signal will always be frequency modulated, meaning the voltage at the Vtune pin

must always be varying linearly. To achieve this, the Vtune pin will be driven by the modulator ramp signal.

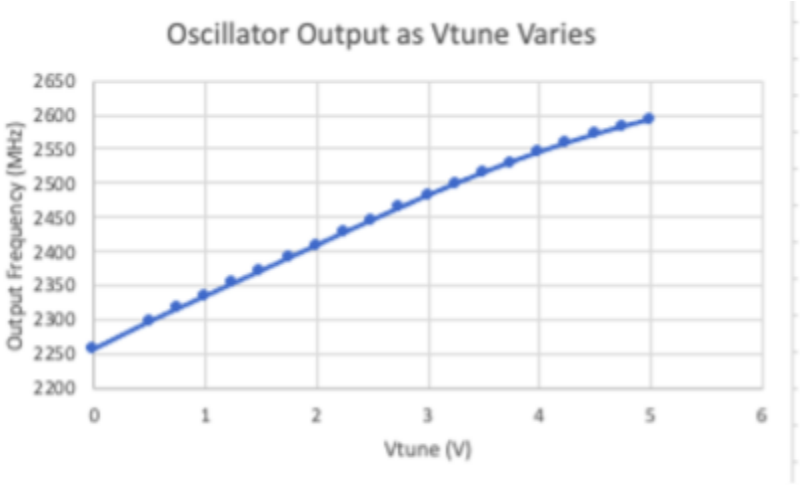

Fig. 27. Oscillator output characteristics.

The signal will begin at a certain frequency, increase linearly until it reaches a stop frequency, then linearly decrease at the same rate until it returns to the start frequency again. This increase and subsequent decrease in frequency constitutes a single period of the signal, which is called a chirp in the context of our RF transmitter.

Before the modulator circuit can be designed, we must specify the desired transmit frequencies and the desired length of a chirp in seconds. The frequencies will be dealt with first. Our system will operate in the entire 2.402 GHz to 2.495 GHz ISM band, resulting in a 93 MHz bandwidth. Figure 27 shows that the corresponding voltage sweep should be from +2V to +3.2V. This provides two specifications that must be dealt with separately in the modulator circuit design: the triangle wave peak-to-peak voltage is 1.2V and the offset voltage is +2.6V. The resistance at the modulator pin 3 is used to control the triangle wave amplitude. A series capacitor and voltage divider at the modulator pin 2 output sets the DC offset. The series capacitor filters out any DC offset already present in the signal, and the voltage divider creates the new DC offset.

$$2.6 = 5\frac{R_1}{R_1 + R_2} \rightarrow R_2 = 0.923R_1$$

The next step is to set the period of the triangle wave. The total period, consisting of the signal linearly increasing and subsequently decreasing back to its start value, is called a chirp. The chirp rate will be set to 40 milliseconds. The signal period of the modulator's output signal is set by two components, the timing resistor connected to either Pin 7 or Pin 8 (our design uses Pin 7) and the external timing capacitor across pins 5 and 6. See Equation X.

$$f_0 = \frac{1}{RC} Hz$$

Though there are an infinite number of combinations that would result in our desired signal period, there are also recommended values for the resistor and the capacitor. The modulator datasheet states that the resistance should be within the range of 4kΩ and 200kΩ, while the capacitor value should remain within 1000pF and 100μF. Our design will use a 100kΩ resistor in series with potentiometer for the resistance, and the capacitor will be 0.47μF. This allows the signal period to be varied if desired, but for a 40-millisecond period, the potentiometer resistance would be set to 84kΩ, for a total resistance of 85kΩ.

Lastly, it should be noted that the modulator has an additional output, a square wave which has a rising edge aligning with the beginning of the ramp and a falling edge aligning with the halfway point of the ramp. This signal can be used for de-chirping recorded ranging profiles, since the rising edge of the square wave would denote the start of a new chirp. For our real-time applications, this isn't necessary because the number of samples being processed at any given time is controlled through software. This signal will go unused in our design.

Figure 28 shows our modulator circuit. The inputs are +12V to power the circuit and create a standard initial DC offset of +6V and +5V to create the output DC offset voltage through the voltage divider. The outputs are the ramp signal, which is sent to the oscillator for both doppler and ranging applications, and the square wave signal, which is unused.

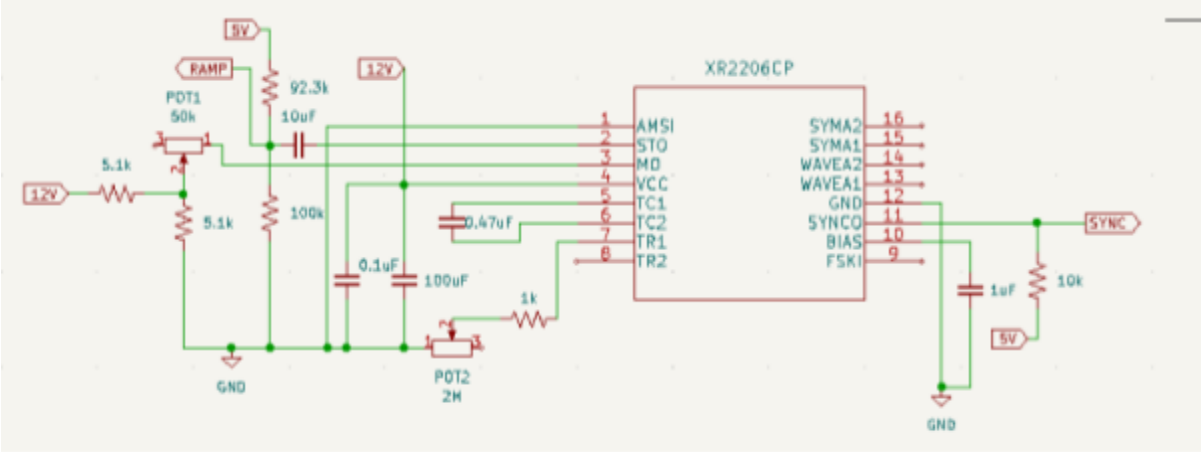

Fig. 28. Modulator circuit.

5.3 RF Subsystem Design

The RF subsystem is a series of components that changes either the frequency or power of the input signal. The input signal to the series comes from the modulator circuit and oscillator discussed in the previous section. The oscillator is the first component in which the signal power analysis switches from traditional DC power analysis using resistance, current and voltage relationships to using RF power analysis. In RF power analysis, signal power is measured in decibels per watt (dB), or in the case of a low powered system like our radar module, decibels per milliwatt (dBm). This change in

signal type is accompanied by a change in connector; rather than using jumper wires, the RF subsystem will use SMA to SMA connectors between its components.

Figure 29 shows how the components of our RADAR module are interconnected. For sensing the environment, a signal is generated at Modulator 1 and moves through the system, out the transmitting antenna, back in from the environment into the receive antenna, and then continues until it exits through the video amplifier. This final signal, in addition to another output signal that is generated by the modulator, holds the information that is subsequently digitized and analyzed by our API.

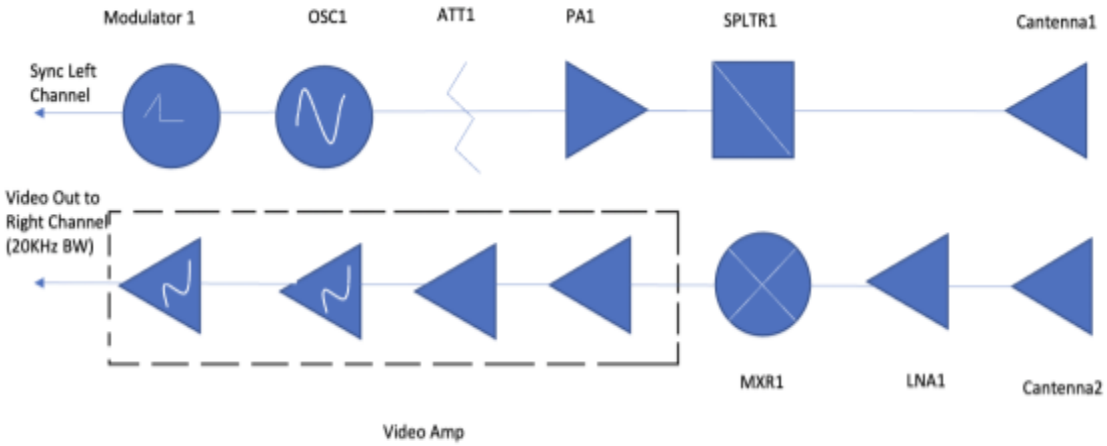

Fig. 29. RF subsystem block diagram.

The RF system may be viewed as a series of power increases and decreases as the signal moves through each component. The oscillator's output power is +6dBm; this is the starting signal power. The next component in the system is the attenuator. Attenuators have multiple functionalities. The first is to decrease the power of an incoming signal. The second is to improve the impedance matching of the system. Here, the attenuator is used directly before a power amplifier, so its primary functionality is impedance matching rather than decreasing the signal power. For the MiniCircuits VAT-3+ attenuator used in our system, the attenuation is dependent on the frequency of the signal. Our signal is below 3 GHz, so the attenuation is only 0.2dB. When working in decibels, scaling is done by addition rather than multiplication, so the signal power is now +6dBm minus 0.2dBm, which is +5.8dBm.

Next comes the power amplifier. It boosts the signal power by 11.5dB at a minimum, bringing the signal power up to at least 17.3dBm. After the amplifier comes the splitter. The splitter has two effects on the incoming signal. The first is that it sends identical copies of the signal to two separate output ports. But as the result of splitting the signal, the power of each signal is only half that of the original signal. This corresponds to a decrease of 3dB. So after the splitter, two identical signals of 14.3dBm are created, one of which is routed to the mixer and the other which is routed to the transmitting antenna.

The transmit antenna has an associated gain of 8.1 dBi, so the final power of the signal at transmission is 22.4dBm, which is about 150 mW. The transmitted signal loses power as it propagates through the air, hits targets in the environment, and comes back through the receive antenna. The power of the received signal cannot be directly calculated. It is dependent on many factors, the most relevant to our application being the distance and velocity of targets in the environment. The material composition, size and shape of the target also have a significant effect on received power. However, the received signal power can be approximated as being between -150dBm and -50dBm. -100dBm will be used in the discussion moving forward.

The receive antenna has the same gain as the transmit antenna, so the signal is increased by 8.1dB upon reentry into the system. Its power is now -91.9dBm. Immediately after the signal comes through the receive antenna, it goes through a low noise amplifier that increases the power by 11.5dB at a minimum (this amplifier is the same as that used before the splitter). This results in a signal power of -80.4dBm. The final component of the component in the system that uses RF power analysis is the mixer. Aside from any activity in the environment, the mixer is the only element of the RF subsystem that alters the frequency of the signal. The original signal is fed into the mixer's LO port. The received signal from the environment is fed into the mixer's RF port. The frequency of the signal at the IF port is equal to that of the original signal minus that of the received signal, effectively producing a signal whose frequency directly reflects how the signal's frequency changed while it was propagating through the environment. In addition to a change in frequency, the input signal to the mixer's RF port sees a maximum drop in power of 6.3dB at the output port. The signal power is then -86.7dBm by the time it reaches the gain stage. Figure 30 below shows a diagram of the power flow through the RF subsystem, from the oscillator to the mixer.

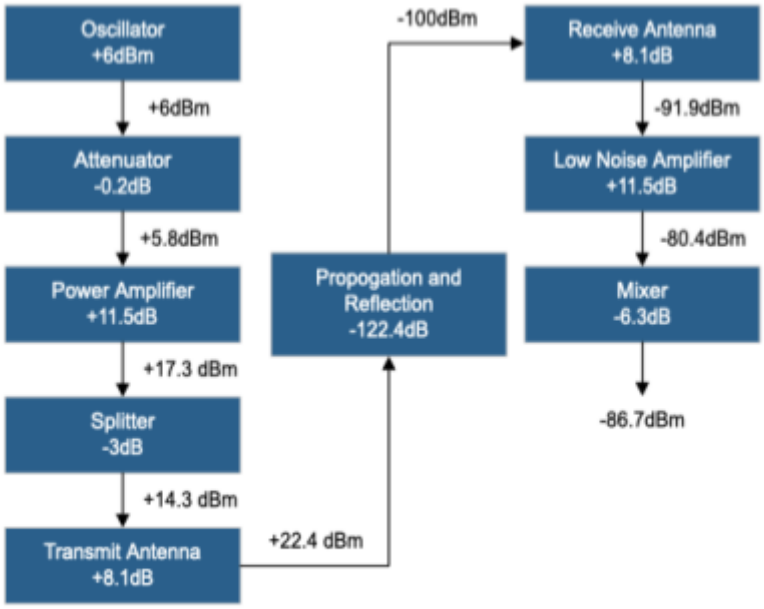

Fig. 30. Example power flow diagram for the RF subsystem.

For close-range applications, the loss due to propagation and reflection is likely to be much lower, so the final signal power will be much higher.

5.4 Gain Stage

The gain stage is the first circuit coming after the RF signal processing components. The SMA connector from the mixer leads to an SMA connector on the gain stage PCB, and that SMA connector feeds two traces (ground and signal voltage) into the gain stage circuitry. The power analysis will now be converted back to a DC analysis. The incoming power is on the order of -10dBm. Three concepts are used to determine the input voltage to the gain stage: the relationship between power in dBm and power in watts, the equation to calculate voltage from watts and resistance, and the impedance of the RF signal processing components, which is 50Ω. An input RF power of -80 dBm will be used as an example to illustrate the calculations.

-80 dBm is equivalent to 10^{-11} watts. The corresponding voltage into the gain stage is then the square root of 10^{-11} times 50Ω, which is 22μV. This means that the order of the voltage going into the gain stage is on the order of microvolts. This is much too small for accurate signal analysis, so the signal needs to be amplified further. An adjustable active non-inverting amplifier will be used for this purpose.

The active amplifier consists of two resistors, a capacitor, a potentiometer and an operational amplifier. The capacitor and resistor tied to the non-inverting terminal of the op-amp serve to create a DC offset for the incoming signal. As it is, the signal is sometimes positive and sometimes negative; to make the signal positive all the time, it must be offset by some DC value. Our team has decided on an offset of +3.3V, which is what the other end of the resistor is tied to. On the inverting terminal of the op-amp, there is one resistor going to +3.3V and a potentiometer tied to the output terminal. These two resistances determine the gain of the circuit as given in Equation X, where the resistor is R_1 and the potentiometer is R_2. .

$$Gain = \frac{R_2}{R_1} + 1$$

Because one of the resistances is variable, the gain of the circuit is variable. Depending on the particular application of the radar and the measured power ratings of the RF signal processing components, the gain of the circuit will need to be adjusted. The gain can vary from unity to 45.45. This is enough to get the signal voltage up to the order of millivolts instead of microvolts. Figure 31 shows the gain stage schematic.

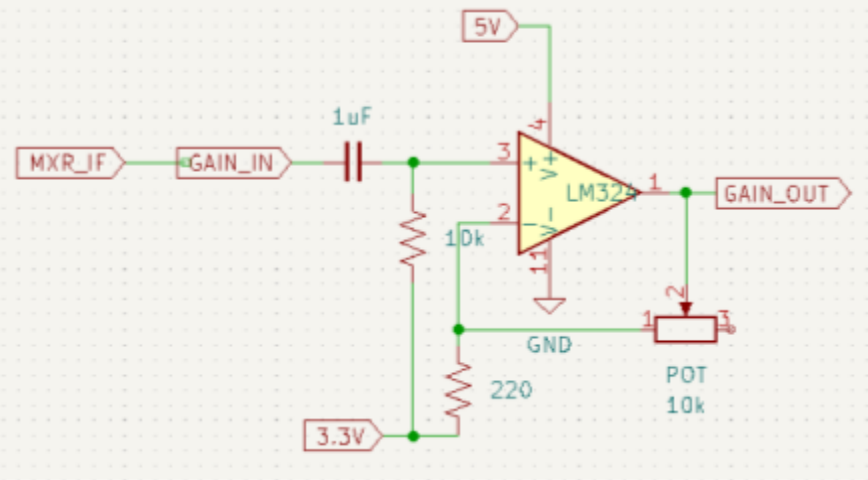

Fig. 31. Gain stage schematic.

Multisim is used to simulate the gain stage circuit design and verify that it is operating as desired. The primary concern is the gain of the circuit. We want to make sure the actual gain aligns closely with the theoretical gain for a given value of the potentiometer. For demonstrative purposes, the potentiometer will be set to its maximum value and represented as a resistor in the Multisim circuit. Figure 32 shows the circuit being simulated, with input and output probes.

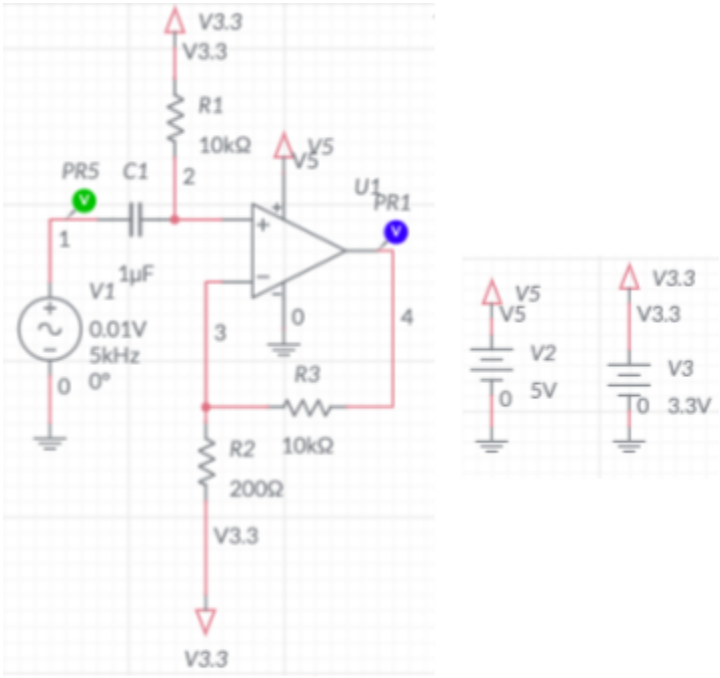

Fig. 32. Multisim gain stage simulated circuit.

Figure 33 shows the input and output signals of the circuit. Note that the input is on the order of millivolts, whereas the realistic input would be on the order of microvolts. A larger input is used to make the output easier to visualize. Note the +3.3V DC offset of the output.

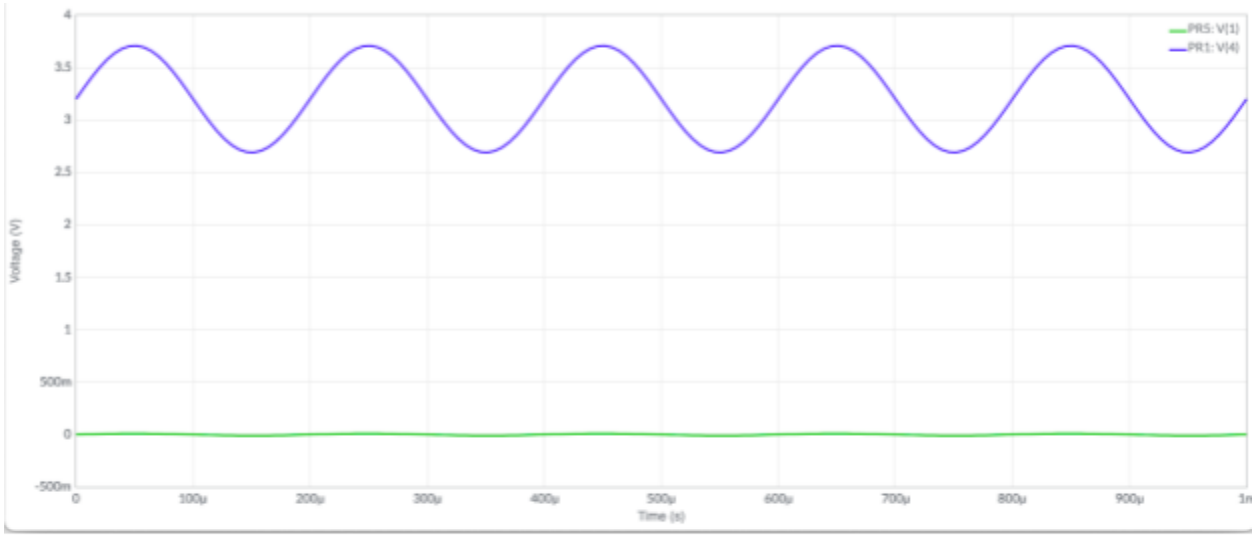

Fig. 33. Multisim gain stage simulated input and output.

A second characteristic that needs to be verified is the frequency response of the circuit. Our gain stage needs to maintain the expected gain for the entire frequency range of the input signal, which for our application is up to 20kHz. Figure 34 shows the gain stage output as the input frequency is varied. The graph shows that the circuit greatly exceeds the frequency response specification. Note that below 20 Hz, the gain of the circuit drops off. This is not a concern, as frequencies this low are not essential to the data analysis.

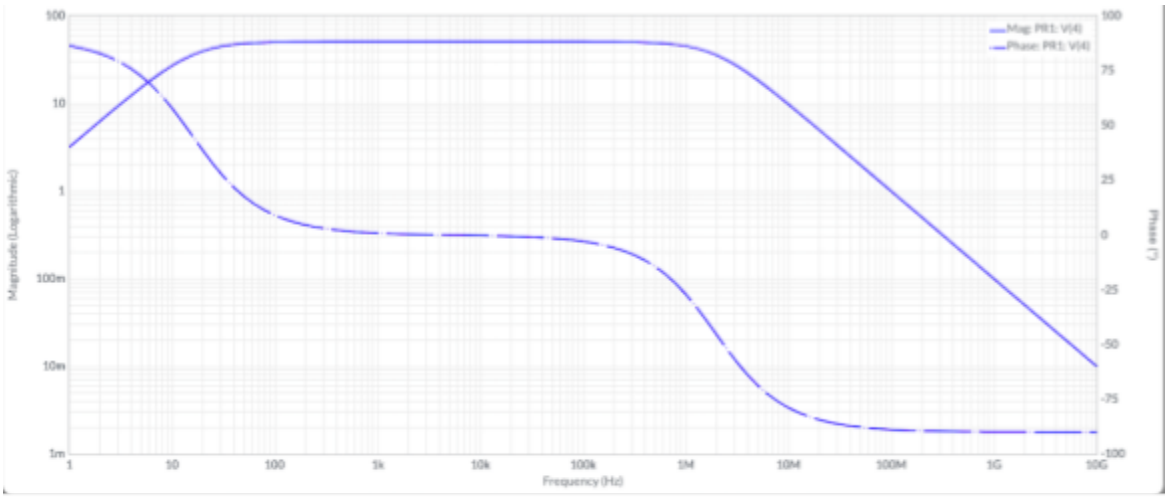

Fig. 34. Multisim gain stage frequency sweep.

5.5 Low Pass Filter

A filter is a circuit that removes unwanted frequencies from a given signal. There are various kinds of filters, denoted by what frequencies they filter and whether they are active or passive. For this project, we are using an active low pass filter. A low pass filter as a circuit that is designed to reject frequencies higher than a certain cutoff frequency. This can both come in the package of a passive or active filter. A passive filter is made up of resistors, inductors and capacitors and primarily rely on the resonance of capacitors and inductors to achieve the required values. It is relatively hard to get a passive filter to operate in general how you would want it to, due to the large capacitance or inductance that would be needed for it to properly operate. Inductance is a particular concern as well since it introduces magnetic interference within the system and can affect performance. Typically it is ideal to stay away from using inductance unless absolutely needed, such as radio frequency or power systems applications. The advantage of a passive device is that we would not have to provide biasing for the device to function. Thus, we would save costs on power consumption if we can prioritize using passive components unless the benefit of an active device outweighs the cost. An active filter is ideally used in most situations since it is much easier to control due to the transistors in an operational amplifier that enable the adjustment of the gain and therefore, change the frequency range being used by the transfer function. For our desired functions the active components will consume a relatively small amount of power. Operational amplifiers do also provide the nice benefit of helping us eliminate the need for inductors on a circuit.

The low pass filter topology being used here is that of a Sallen-Key filter cascaded with other filters. The Sallen-Key is a good topology to use because of its simplicity. Sallen Keys are a voltage-controlled voltage-source (VCVS) filter topology that uses a voltage amplifier with high input impedance and near 0 output impedance. Input Impedance is best defined as the measure of opposition to a given current, into a given load. Having high input impedance means that we receive as much of the voltage signal as we possibly can, ultimately reducing the loss of the signal so we can provide, in this case, maximum gain from the operational amplifier. Similarly, we would want “0” output impedance from the circuit to ensure that we have maximum gain from the operational amplifier. This becomes especially important in a cascaded system, since we would want the amplifiers to have as minimum loss as possible as they transition throughout the chain. Generally, if the input impedance were to be high we could “fix” it by cascading additional parts onto the circuit. Though doing this would not be ideal at all, since it would create higher costs with the increased number of ICs, complicate the circuit further & reduce layout space for either the printed circuit board (in this case) or the integrated circuit that would be fabricated on a silicon wafer. We need to have a 15 kilohertz low pass filter here to filter out everything above that range.

In order to understand the design decisions surrounding our fourth-order sallen-key low-pass filter design we will first explore the behavior of a first-order low-pass filter and then move up to a fourth-order filter. Consider the circuit in figure 35. When the

frequency of the source is varied, only the impedance of the capacitor is affected. At lower frequencies the capacitor acts as an open circuit.

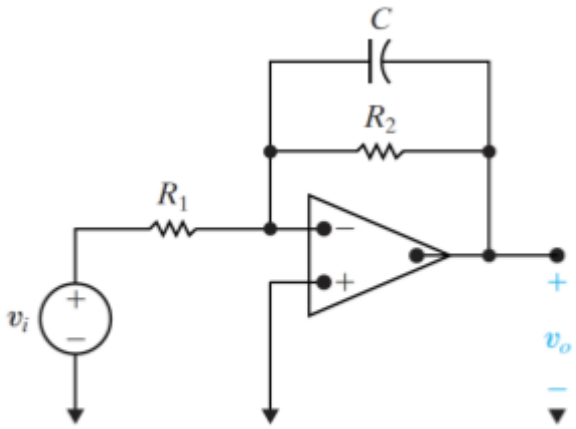

Fig. 35. Example low pass filter circuit.

The impedance of a capacitor is given as

$$X_c = \frac{1}{2\pi f C}$$

As we can see from the above equation, as the frequency decreases the impedance of a capacitor increases. Therefore, a capacitor does not act as an open circuit, but rather has a very high impedance. Recall that an open circuit has an impedance of ∞. The frequency of a DC current is effectively zero, which is why a capacitor acts like an open circuit for DC currents.

So, at low frequencies the op-amp circuit acts as an amplifier with a gain of $\frac{-R_2}{R_1}$. At very high frequencies the capacitor acts as a short circuit, which connects the output of the op-amp circuit to ground. This is why the circuit in figure # acts as a low-pass filter with a pass-band gain of $\frac{-R_2}{R_1}$.

We can also confirm this by computing the transfer function $H(s) = \frac{V_0}{V_{in}}$.

$$H(s) = \frac{-Z_f}{Z_i}$$

$$H(s) = \frac{-R_2 \,||\, (\frac{1}{sC})}{R_1}$$

$$H(s) = -K\,\frac{\omega_c}{s+\omega_c}\text{ ; where } K = \frac{R_2}{R_1} \text{ and } \omega_c = \frac{1}{R_2 C}$$

To calculate a specified cutoff frequency we can therefore use the following equation.

$$C = \frac{1}{R_2 \omega_c}$$

The above equations imply that the op-amp filter will allow us to independently specify the cutoff frequency and passband gain based on the component values used.

The cutoff frequency of a filter is defined as the frequency at which the maximum magnitude of the transfer function has been reduced by -3dB.

In order to construct a circuit with a sharper but continuous frequency response at cutoff frequency we can utilize a higher-order op amp filter.

As more filters are cascaded, the transition from the passband to the stopband becomes sharper.

In order to compute the transfer function for a cascade of n-prototype low-pass filters we simply multiply the individual transfer functions.

$$H(s)' = \left(\frac{-1}{s+1}\right)\left(\frac{-1}{s+1}\right)\ldots\left(\frac{-1}{s+1}\right) = \frac{(-1)^n}{(s+1)^n}$$

One must note that as the order of the low-pass filter is increased by adding prototype low-pass filters to the cascade, the cutoff frequency also changes.

If we are able to calculate the cutoff frequency of the higher order filters formed in the cascade of the first-order filters, we can use frequency scaling to calculate component values that move the cutoff frequency to its desired location. Once we start with a cascade of n prototype low-pass filters, we can compute the cutoff frequency for the resulting nth order low-pass filter.

For a Sallen-key filter we can choose our components for a desired cut-off frequency using the following formula

$$f_c = \frac{1}{\sqrt{R_1 R_2 C_1 C_2}}$$

Our values were chosen such that $f_c = 15kHz$.

Figure 36 below shows the schematic for the low pass filter. The input comes directly from the gain stage, and the output goes to the analog to digital converter. The low pass filter is the final processing of the analog signal before it is digitized. The two operational amplifiers shown are only two within the quad operational amplifier package we are

using. They have shared power rails, but their inverting, non-inverting, and output terminals are unique.

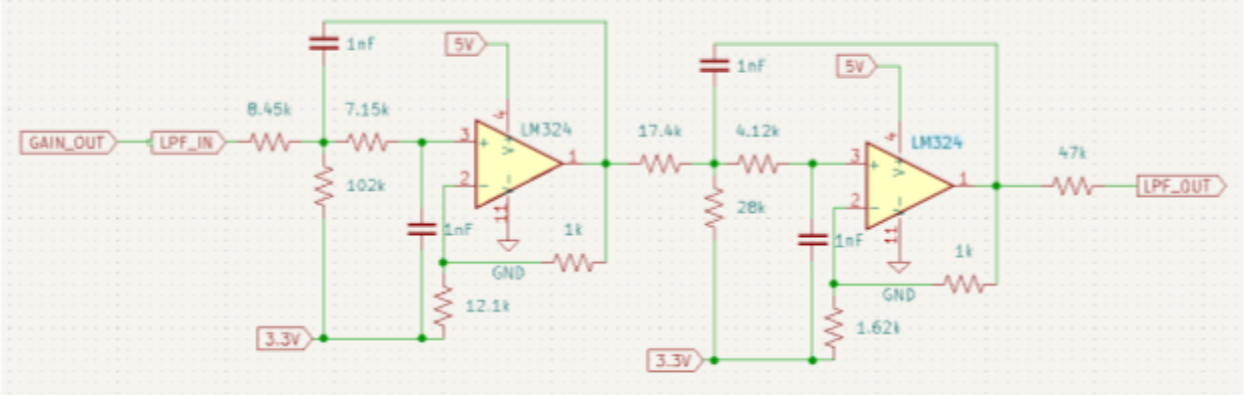

Fig. 36. Low pass filter schematic.

Multisim is used to simulate the low pass filter circuit design and verify that it is operating as desired. The primary concern is the frequency response of the circuit, so first a frequency sweep will be performed. Figure 37 shows the circuit being simulated, with a single input probe and two output probes, one after each filter stage. The second output probe shows the frequency response of the entire circuit, not just the second filter stage.

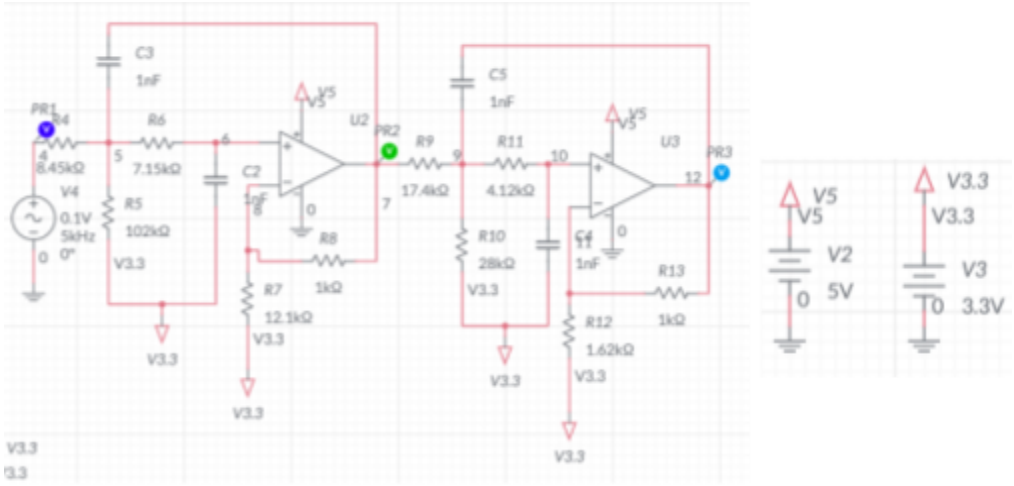

Fig. 37. Multisim low pass filter circuit.

Figure 38 shows the AC sweep for the circuit. Notice how the cutoff frequency of the first stage (in green) and the cutoff frequency of both stages combined (the blue) are identical at 15 kHz. The difference is how steep the drop-off is after the cutoff frequency. After the first stage, the cutoff is not very steep, meaning the output signal still contains quite a bit of the frequencies beyond the cutoff. After the second stage is added, the cutoff becomes much steeper, meaning much more of the frequencies beyond the cutoff are eliminated from the signal.

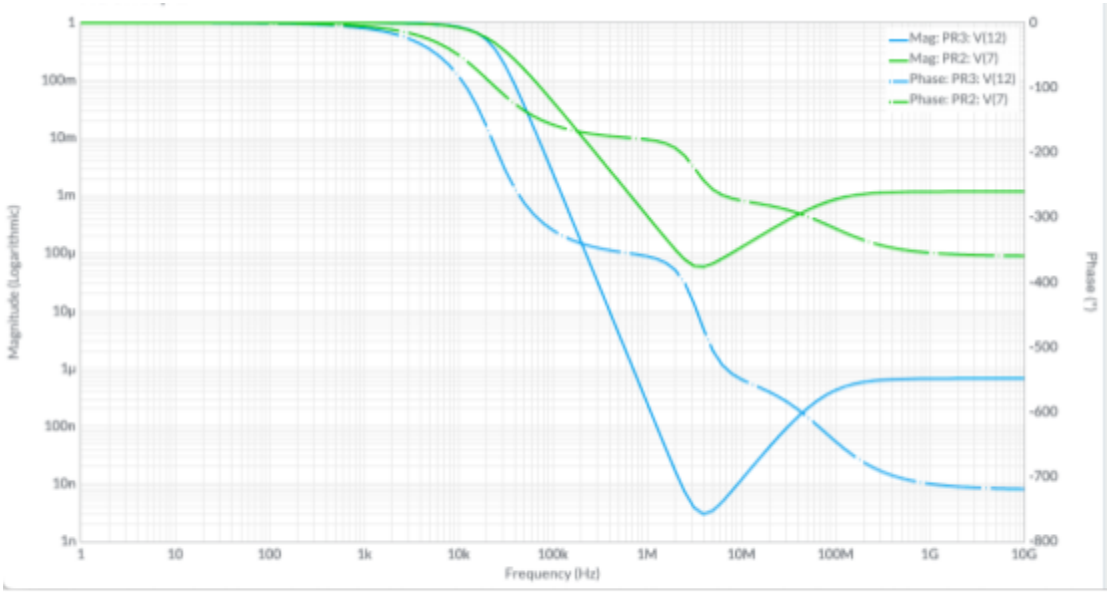

Fig. 38. Low pass filter AC sweep. The green shows the output of the first stage. The blue shows the output of both stages combined.

An example transient response will be shown to demonstrate the removal of unwanted frequencies. In Figure 39, the input signal is at 20kHz, which is beyond the filter's cutoff frequency. Notice how the magnitude of the output signal is already half that of the input signal. The output phase aligns with that shown in the AC sweep for 20kHz.

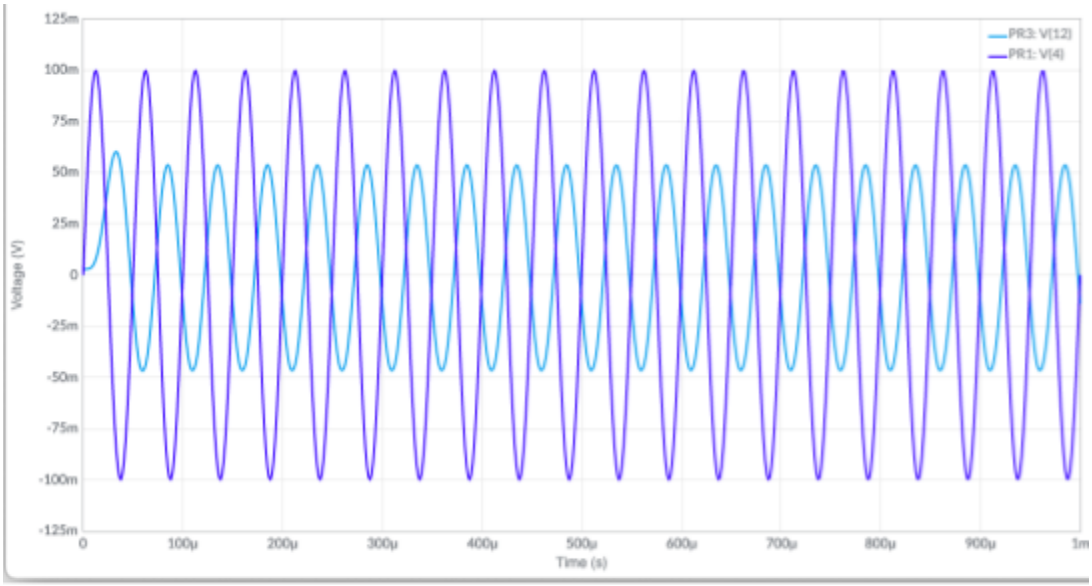

Fig. 39. Low pass filter simulated input (indigo) and output (light blue).

5.5 Analog-to-Digital Converter Circuit

We have opted to use the PCM1802 dual-channel 24-bit 96kHz audio analog-to-digital converter from Texas Instruments for our project. While two channels are available with this model, the same signal will be fed to both input channels which effectively makes it indifferent from a single-channel model. This device is driven by the SCKI, or system clock input, and it can operate in either “slave” mode or “master” mode. In “slave” mode the device has its clock signals of LRCK (left/right clock) and BCK (bit clock) as inputs that the user must provide in order to retrieve a bit output on DOUT. In “master” mode the device provides the clock signals of LRCK and BCK, which can then be used to trigger interrupts and collect digitized bit data from DOUT. We have opted for SCKI to be driven by a 49.152MHz oscillator in order to achieve the maximum sampling frequency of 96kHz. For the device to operate as “master” and have a sampling frequency of 96kHz we must set “INTERFACE MODE” to “Master mode (512 fS)” by setting the “MODE0” pin to HIGH and the “MODE1” pin to LOW. The “FORMAT” is set to “24-Bit, MSB-First, Left-Justified” by setting both “FMT0” and “FMT1” pins to LOW. From the interface timings shown below, the following observations can be made: a sample begins at the rising edge of FSYNC and ends at the falling edge of FSYNC; a new bit is ready to be read from DOUT at the rising edge of BCK; and the channel of the associated input is indicated by LRCK, with HIGH indicating the left-channel and LOW indicating the right-channel.

Fig. 40. PCM1802 interface timings. Credit: Texas Instruments.

Figure 41 below illustrates how the various power supplies, input signals, and output signals are to be inter-connected within the circuit. Our designed power supply provides the regulated +5V and +3.3V DC voltages, our received RF signal is tied to VIN, and our XLH535049.152000X 49.152MHz oscillator provides the clock signal tied to SCKI. The 1uF capacitor C1 gives an 8 Hz cutoff frequency for the input high pass filter. The two 0.1uF capacitors C2 and C3 that are connected to VREF1 and VREF2 are used for power supply noise reduction purposes. The two 0.1uF capacitors C4 and C5 that are connected between VCC and GND as well as between VDD and GND are used as bypass capacitors, which inhibit noise from entering the analog-to-digital converter by bypassing to GND, yielding a cleaner DC voltage which may be present on the switching voltage regulators used by our power supply.

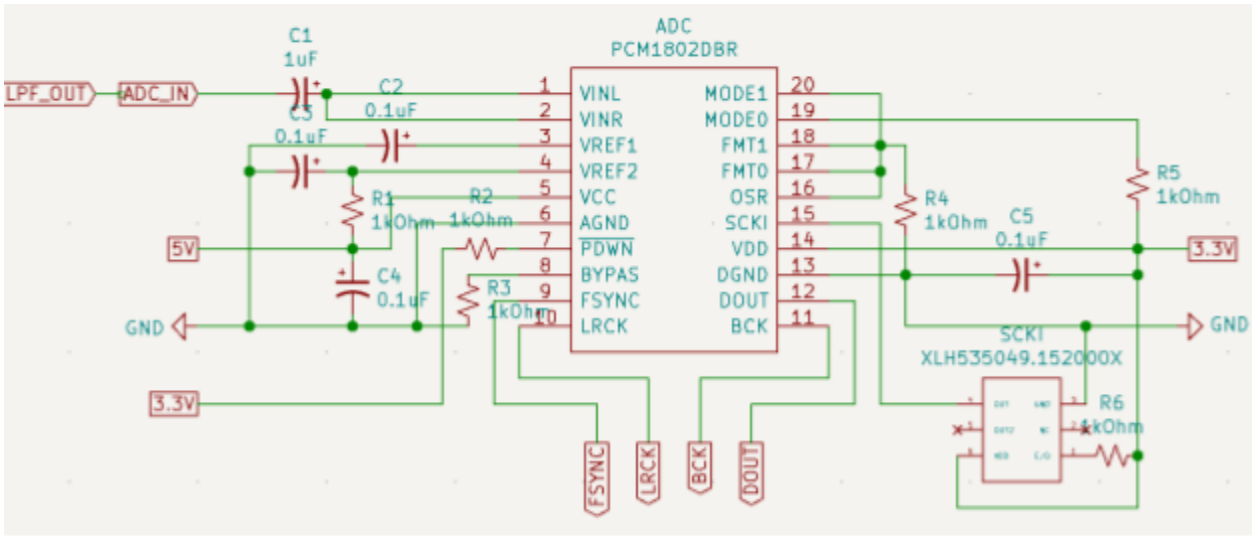

Fig. 41. PCM1802DBR analog-to-digital converter schematic.

One crucial component in the circuit is the crystal clock oscillator driving the ADC. We need a clock signal in order for the ADC to collect samples of the data at a high enough frequency to satisfy the Nyquist Sampling Theorem which states if a system uniformly samples an analog signal at a rate that exceeds the signal's highest frequency by a factor of at least two, then the original signal can be perfectly recovered from the sampled discrete values.

A crystal oscillator uses a piezoelectric crystal as a frequency selective element. Piezoelectricity is the ability of certain crystalline materials to convert mechanical energy into electrical energy. We will be utilizing the inverse piezoelectric effect which turns electrical energy into mechanical energy. This allows us to provide a stable clock signal to the analog to digital converter. A crystal oscillator relies on the slight change of shape of a quartz crystal under an electric field.

A crystal oscillator can be modeled using capacitors and inductors where the series resonance is given as

$$f_r = \frac{1}{2\pi\sqrt{L*C}}$$

And the parallel resonance is given as

$$f_p = \frac{1}{2\pi*\sqrt{L*C_s}} * \sqrt{1 + \frac{C_s}{C_p}}$$

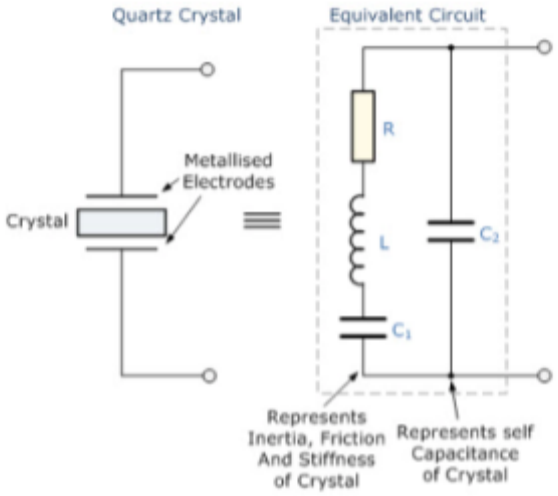

Fig. 42. Quartz crystal equivalent circuit.

Crystals must be designed to provide a load capacitance.

Usually bipolar transistors are used in the construction of crystal oscillator circuits. Since crystal oscillators are highly stable, have a high Q-factor, and are small, they are a more favorable choice for timekeeping over other types of resonators such as LC circuits.

In order to have a functional analog-to-digital converter we need a clock source. Originally we planned to build a crystal clock oscillator from scratch to drive the analog to digital converter. This will allow us to collect discrete samples from the continuous analog signal. During our market search we found a Renesas XL Low Phase Noise Quartz-based PLL oscillator that would provide this function for us. This is convenient as it would eliminate the need to design an entire crystal oscillator circuit. The Renesas XL is a ultra-precision crystal oscillator with 750 to 890fs phase jitter over a 12kHz - 20MHz bandwidth. The noise from this device is comparable to traditional bulk quartz and SAW oscillators. This device was an ideal choice due to its low cost, low noise, and wide frequency range. This is an active device and we would be supplying 3.3V from the power supply circuit to power the device.

In terms of physical implementation this device is directly integrated onto the circuit board for the analog to digital converter.

5.6 PCB Vendor and Assembly

For our printed circuit board, we will be outsourcing the official PCB manufacturing to JCLPCB due to their ability to provide cheap, efficient and effective manufacturing for our end product. For prototyping we will be looking towards using the PCB Milling machine in the senior design lab to prototype a few of our boards.

Assembly will be done using the reflow oven and solder paste. We will need to obtain a solder mask, especially if our boards get more complicated as the project progresses. Normally we will use thin stainless steel as the solder mask. It is easy to clean and relatively easy to cut. The disadvantage of using stainless steel is that it is not transparent. This makes aligning later in the process more difficult. For thinner designs a special type of thin adhesive tape can be used. This makes aligning the mask with the milled sections easier, as you can see under it. With most PCB milling designs there is a very small room for error in regards to alignment. If the melted solder gets past an isolation line the device will be shorted and will not function correctly. If all else fails we will be hand soldering all the components onto the board carefully. We will try to stick to obtaining surface mount components for the PCB, ie. op amps, resistors, capacitors, etc. Unfortunately we will be unable to obtain a surface mount equivalent of the XR-2206 IC, since the only model available is throughhole and we are unable to find a similar chip with similar functionality.

The benefit of having surface mount components will be that we can use relatively small components on the circuit board, which would therefore reduce the overall size of the PCB maximizing functionality with cost. One error of beginner PCB designers is not optimizing the board to the smallest size possible. This would mean that the designer would have to effectively orient the parts to reduce its overall layout, and also factoring in the proper routing for each component. Milling a double sided board will not be much of an issue for us. The Quick Circuit J5 in the Senior Design Fabrication lab can easily do double-sided boards if one knows how to operate it. At a high level it simply amounts to flipping the board over from left to right and then using the “mirror” function in Isopro. We can also easily create extremely precise through-holes to fit our components competently.

For our prototype, two of our teammates will be using the PCB milling machine in the senior design lab. For this version, we will need to be very careful drilling the mounting holes and interconnects especially if we attempt to use a compact design of the board. We must be careful to keep the electrical connectivity functional, since otherwise it will make the board useless and potentially fry our components. For any through-hole components it is important to correctly isolate the underside of the board.

6.0 Software Design

6.1 Real-Time Processing

The timings of the generated signals are relatively fast, with sampling frequencies in the megahertz range and wait periods in the microsecond range. As such the burst data must be accumulated directly on the microcontroller that is ingesting the samples from the analog-to-digital converter prior to transmission to the host device in lieu of transmitting individual samples. Each burst contains a number of chirps that is specified by the “chirps-per-burst” parameter, each chirp contains a number of samples that is specified by the “samples-per-chirp” parameter, the inter-chirp delay is specified by the

“chirp period” in microseconds, and the inter-burst delay is specified by the “burst period” in microseconds. An example of such burst data is visualized in the waveform of Figure 43 below.

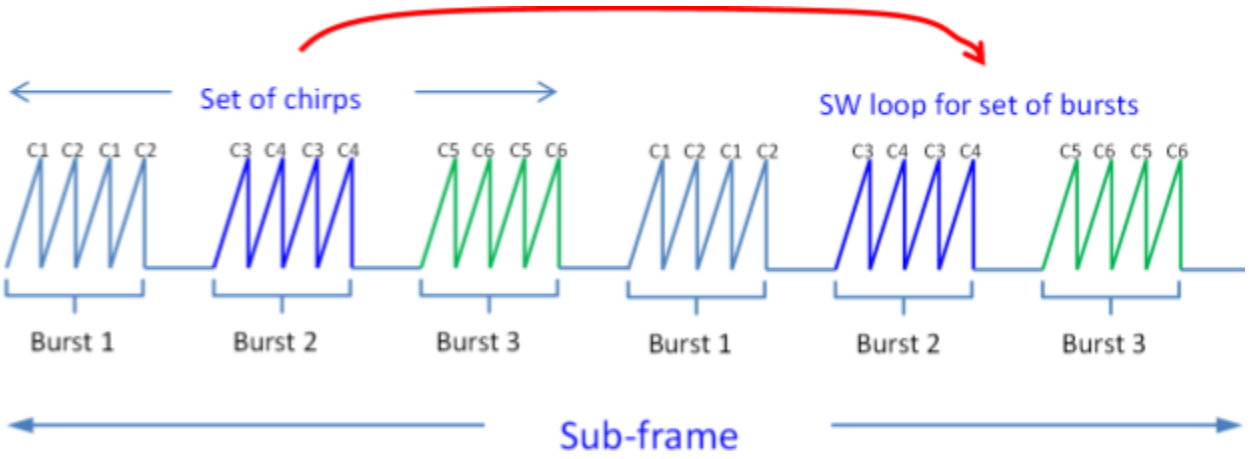

Fig. 43. Waveform of an FMCW radar burst. Credit: Texas Instruments [14].

These samples are obtained as serial data through an external interrupt-attached digital pin where a total of n interrupts would be triggered for an n-bit analog-to-digital converter. Each bit of the n-bit data sample is to be stored in a buffer with a size of $\lceil n / 8 \rceil$ bytes. For each of the n subsequent interrupts, the bits are accumulated in the buffer by left-shifting the buffer over by 1 bit then OR’ing the digital pin reading to the buffer. The process of filling the buffer with the n-bit analog-to-digital converter reading is visualized in the flow chart of Figure 44 below.

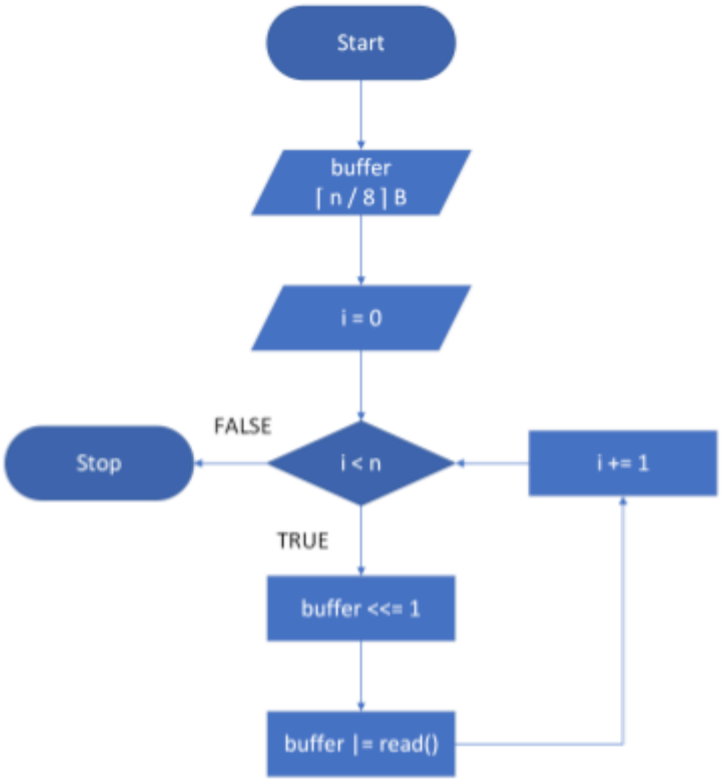

Fig. 44. Flow chart demonstrating the accumulation of serial analog-to-digital converter bit values into a buffer.

The minimum possible non-zero value that each sample can represent is constrained by the analog-to-digital converter's bit resolution and maximum input voltage. For example, for an analog-to-digital converter with a bit resolution of 24-bits and a maximum input voltage of +5V, the analog value would be represented as a floating-point number in the range of 0.0 to 5.0, with a minimum possible non-zero value of $5 / 2^{24} \approx 2.98 * 10^{-7}$, or a resolution of about +0.3μV.

These samples are then accumulated in the burst buffer whose size is determined by the size of each sample, the number of samples-per-chirp, and the number of chirps-per-burst. This size is constrained by the Static Random-Access Memory (SRAM) available to the microcontroller. For example, using an SRAM size of 32 kilobytes, reserving 50% of that SRAM for burst data, a sample size of 24 bits ($[24 / 8] = 3$ bytes), and a chirps-per-burst of 5, each burst could contain a total of $\lfloor SRAM * 0.5 / (sample\ size * chirps\text{-}per\text{-}burst) \rfloor = \lfloor 32 * 10^3 * 0.5 / (3 * 5) \rfloor = 1066$ samples-per-chirp. At the sampling frequency of $f_s = 96kHz$, the total amount of time to collect these 1066 samples would be $1 / f_s * samples - per - chirp = 1 / (96 * 10^3) * 1066 \approx 11.1ms$. With the

inter-chirp period of 40ms as set by the modulator, the total amount of time to collect a complete burst would be approximately $11.1ms + 40ms * 5 = 211.1ms$.

6.2 Ripple Interface

The software design is critical to introducing aspects of modularity by offering a Software Development Kit (SDK) with our frequency-modulated continuous-wave radar module. This SDK is designed to expedite the application development process for end-users by providing an Application Programming Interface (API) that conforms to the Consumer Technology Association's (CTA)® Ripple standard. This standard was chosen for its emphasis on enabling inter-operation between hardware and software for radar systems. This SDK is to be run on a host device that must connect to our module via a USB2 interface.

At its core the API implementation of the Ripple standard offers a layer hardware abstraction to interface with our module. This includes a state machine specification that controls the states of the system and the configuration of the system's parameters that determine its modes of operation.

The state machine is extraordinarily simple in that it consists of four states and six transitions. The four states are the following: "OFF", "IDLE", "ACTIVE", and "SLEEP". The six transitions are the following: "TurnOn", "TurnOff", "StartDataStreaming", "StopDataStreaming", "GoSleep", and "WakeUp". The sensor is initialized to the "OFF" state where it consumes the least amount of power and does not produce any amount of data. Calling the "turn_on" method triggers the "TurnOn" transition that transitions from the "OFF" state to the "IDLE" state, which indicates that the sensor is prepared for operation. It is from within this state that configurations may be activated and deactivated. Calling the "go_sleep" method triggers the "GoSleep" transition that transitions from the "IDLE" state to the "SLEEP" state, where power consumption is minimized but configuration settings are retained, and calling the "wake_up" method triggers the "WakeUp" transition that transitions from the "SLEEP" state back to the "IDLE" state. Calling the "start_data_streaming" method triggers the "StartDataStreaming" transition that transitions from the "IDLE" state to the "ACTIVE" state, where the sensor begins transmitting and receiving radio waves using the activated configuration settings, and calling the "stop_data_streaming" method triggers the "StopDataStreaming" transition that transitions from the "ACTIVE" state back to the "IDLE" state. To discard all configuration settings and deactivate the device, the "turn_off" method must be called to trigger the "TurnOff" transition that transitions from the "IDLE" state to the "OFF" state. This state machine is visualized in the diagram of Figure 45 below.

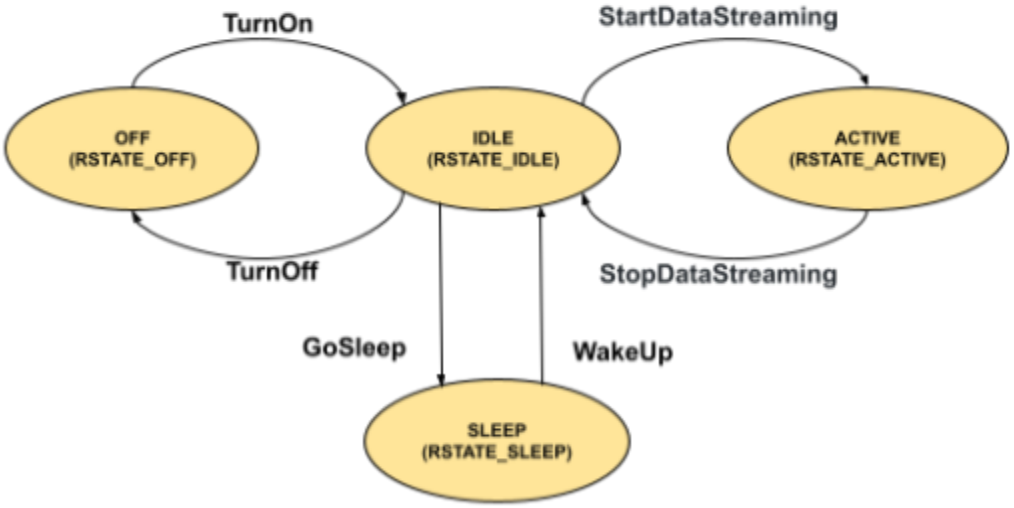

Fig. 45. The Ripple state machine diagram. Credit: Consumer Technology Association.

The API allows for the definition of up to 256 configurations via unsigned byte slot identifiers, however we will enable only one slot (one configuration) for our project. These configuration parameters consist of the three following groups: main parameters, transmitter parameters, and receiver parameters.

The main parameters group consists of the following parameters: the afterburst power mode determines the power mode of the module following the burst period; the inter-chirp power mode determines the power mode of the module between chirp periods; the burst period specifies the period between subsequent bursts in microseconds; the chirp period specifies the period between subsequent chirps in microseconds; the chirps-per-burst specifies the number of chirps contained within a single burst; the samples-per-chirp specifies the number of analog-to-digital converter samples contained within a single chirp; the lower frequency specifies the frequency at which the transmitter antenna begins the emitting of its signal; the upper frequency specifies the frequency at which the transmitter antenna ends the emitting of its signal; the transmitter antenna mask specifies which transmitter antennas are currently enabled; the receiver antenna mask specifies which receiver antennas are currently enabled; and the analog-to-digital converter sampling frequency specifies the rate at which analog samples are digitized.

The transmitter parameters group consists of the following parameters: the power specifies the transmitter antenna power emission in decibels.

The receiver parameters group consists of the following parameters: the variable gain amplifier specifies the signal gain in decibels; the high pass filter gain specifies the filter gain in decibels; the high pass filter cutoff specifies the cutoff frequency in kilohertz.

The Ripple standard defines two interfaces that the developer must implement to facilitate bi-directional module-to-host communication, which consists of a sensor interface and a sensor observer interface. The sensor interface directly communicates with our board via USB2 to modify & retrieve parameter values, transition between states, and retrieve burst readings. The sensor observer interface contains callbacks to be notified of modified register values and be notified of new burst readings to be processed.

The sensor interface stores the aforementioned configuration parameters to be communicated with the module. These configuration parameters of a slot can be modified and retrieved with “set” and “get” methods, which include the following: “set_main_param”, “get_main_param”, “set_tx_param”, “get_tx_param”, “set_rx_param”, and “get_rx_param”. Once the specified slot has its configuration parameters set, the parameters can be communicated to the module with the “activate_config” method if it is in the “IDLE” state. With the configuration activated data streaming may begin by calling the “start_data_streaming” method. When this method is called a thread is spawned independently of the main thread in order to communicate with the module to retrieve new bursts. This allows for seamless data transmission and data processing.

Prior to turning on the sensor the user may register a sensor observer, which contains the “on_burst_ready” callback. As the name indicates, it is called when a burst is ready to be read from the sensor. As such, the user may call “read_burst” on the sensor to retrieve the burst format as well as the raw data as a sequence container of bytes. This data contains the measured power of the received signal which may be used for a variety of purposes, such as velocity measurement, ranging measurement, and synthetic-aperture radar mapping.

6.3 Module-to-Host Data Transmission

Module-to-host data transmission is facilitated by the USB2 serial communication standard employing the request-response message pattern. In this pattern the host populates and sends a request structure to the module, which in turn processes the request, populates a response structure, and sends the response back to the host. The request structure consists of an 8-bit unsigned integer “command” field that designates the command to be processed, a 16-bit unsigned integer “size” field that specifies the size of the subsequent buffer field in bytes, and an 8-bit unsigned integer array “buffer” field allocated to the size specified by the “size” field. Similarly, the response structure consists of an 8-bit unsigned integer “status” field that specifies if the processed request was either a success or a failure, a 16-bit unsigned integer “size” field that specifies the size of the subsequent buffer field in bytes, and an 8-bit unsigned integer array “buffer” field allocated to the size specified by the “size” field. The “command” field values of the request are chosen by the programmer and the “status” field values of the response of “SUCCESS” and “FAILURE” are 1 and 0, respectively. Alternatively, the user may

implement custom status values. These structures and their fields are more clearly visualized in the Table 3 below.

Table III
Request and Response Data Structures

Request Structure	
Type	**Field**
uint8_t	command
uint16_t	size
uint8_t []	buffer
Response Structure	
Type	**Field**
uint8_t	status
uint16_t	size
uint8_t []	buffer

Our project will implement the following request commands to be processed by the module: “SET_MAIN_PARAMS”, “SET_TX_PARAMS”, “SET_RX_PARAMS”, “SET_STATE”, and “READ_BURST”. The values corresponding to these commands are stored in an enumeration with the following values: “SET_MAIN_PARAMS” is 0; “SET_TX_PARAMS” is 1; “SET_RX_PARAMS” is 2; “SET_STATE” is 3; and “READ_BURST” is 4.

The “SET_MAIN_PARAMS” command sets the main parameters on the module. The request is expected to contain the following data: a command field storing the value of the “SET_MAIN_PARAMS” enumeration element; a size field storing the value of $4\ bytes\text{-}per\text{-}parameter\ *\ 15\ parameters\ =\ 60\ bytes$; and a buffer field storing the values of the parameters “AFTERBURST_POWER_MODE”, “INTERCHIRP_POWER_MODE”, “BURST_PERIOD_US”, “CHIRP_PERIOD_US”, “CHIRPS_PER_BURST”, “SAMPLES_PER_CHIRP”, “LOWER_FREQ_MHZ”, “UPPER_FREQ_MHZ”, “TX_ANTENNA_MASK”, “RX_ANTENNA_MASK”, and “ADC_SAMPLING_HZ”. If this size expectation is met then the parameter values stored in the buffer are applied and digital pin output values associated with the parameters are modified. On success a response is generated containing the following data: a status field storing the value of “SUCCESS”; a size field storing the value of 0; and a buffer field that is empty. On failure a response is generated containing the following data: a status field storing the value of “FAILURE”; a size field storing the value of 0; and a buffer field that is empty.

The “SET_TX_PARAMS” command sets the transmitter parameters on the module. The request is expected to contain the following data: a command field storing the value of the “SET_TX_PARAMS” enumeration element; a size field storing the value of $4\ bytes\text{-}per\text{-}parameter\ *\ 1\ parameter\ =\ 4\ bytes$; and a buffer field storing the value of the parameter “POWER_DB”. If this size expectation is met then the parameter values stored in the buffer are applied and digital pin output values associated with the parameter are modified. On success a response is generated containing the following data: a status field storing the value of “SUCCESS”; a size field storing the value of 0; and a buffer field that is empty. On failure a response is generated containing the following data: a status field storing the value of “FAILURE”; a size field storing the value of 0; and a buffer field that is empty.

The “SET_RX_PARAMS” command sets the receiver parameters on the module. The request is expected to contain the following data: a command field storing the value of the “SET_RX_PARAMS” enumeration element; a size field storing the value of $4\ bytes\text{-}per\text{-}parameter\ *\ 3\ parameters\ =\ 12\ bytes$; and a buffer field storing the value of the parameters “VGA_DB”, “HP_GAIN_DB”, and “HP_CUTOFF_KHZ”. If this size expectation is met then the parameter values stored in the buffer are applied and digital pin output values associated with the parameter are modified. On success a response is generated containing the following data: a status field storing the value of “SUCCESS”; a size field storing the value of 0; and a buffer field that is empty. On failure a response is generated containing the following data: a status field storing the value of “FAILURE”; a size field storing the value of 0; and a buffer field that is empty.

The “SET_STATE” command directly assigns the state of the module. The state values are stored in an enumeration with the following values: “OFF” is 0; “IDLE” is 1; “SLEEP” is 2; and “ACTIVE” is 3. The request is expected to contain the following data: a command field storing the value of the “SET_STATE” enumeration element; a size field storing the value of $1\ byte$; and a buffer field storing the value of the state enumeration element of either “OFF”, “IDLE”, “SLEEP”, or “ACTIVE”. If this size expectation is met then the state is set to the value that is stored in the buffer and the following events will occur for each of the states: in the “OFF” state the module is completely powered down; in the “IDLE” state the module is powered on, is ready to have configurations set, and is ready to be activated; in the “SLEEP” state the module enters a low-power mode where the configuration settings are retained; and in the “ACTIVE” mode the module begins actively transmitting and receiving data. On success a response is generated containing the following data: a status field storing the value of “SUCCESS”; a size field storing the value of 0; and a buffer field that is empty. On failure a response is generated containing the following data: a status field storing the value of “FAILURE”; a size field storing the value of 0; and a buffer field that is empty.

The “READ_BURST” command reads the burst current stored within the module’s internal burst buffer. The request is expected to contain the following data: a command field storing the value of the “READ_BURST” enumeration element; a size field storing the value of 0; and a buffer field that is empty. On success a response is generated

containing the following data: a status field storing the value of “SUCCESS”; a size field storing the size of the burst buffer that is $\lceil n / 8 \rceil * samples\text{-}per\text{-}chirp * chirps\text{-}per\text{-}burst$ bytes where n is the amount of bits provided by the analog-to-digital converter; and a buffer field storing the analog-to-digital converter samples. On failure a response is generated containing the following data: a status field storing the value of “FAILURE”; a size field storing the value of 0; and a buffer field that is empty.

6.4 Ranging and Velocity Computation

Implementations such as velocity extraction, range extraction, or synthetic aperture radar mapping must be responsive if they are expected to provide real-time updates. In order to achieve this the highly-optimized Eigen linear algebra library is leveraged to handle these computations hastily. Using Eigen’s Fast Fourier Transform (FFT) class it is possible to extract both the velocity and ranging information from a radar burst. The Fast Fourier Transform computes the Discrete Fourier Transform but many orders of magnitude faster by taking advantage of the periodicity of sinusoids. By identifying where these signals overlap it is possible to reduce the overall number of multiplications necessary to compute the Fourier transform. This reduction in multiplications is substantial yielding a complexity of $O(N\ log_2 N)$ compared to the $O(N^2)$complexity of the DFT. For large data sets this reduction is immense as the growth of N^2 rapidly outpaces the growth of $N\ log_2 N$. For example, for an $N = 100$ the ratio of computations of the DFT compared to the FFT is $(100^2)/(100 * log_2 100) \approx 15$, and for an $N = 1000$ the ratio of computations of the DFT compared to the FFT is $(1000^2)/(1000 * log_2 1000) \approx 100$. These discrepancies between $O(N^2)$ and $O(N\ log_2 N)$ are more clearly visualized in Figure 45 below.

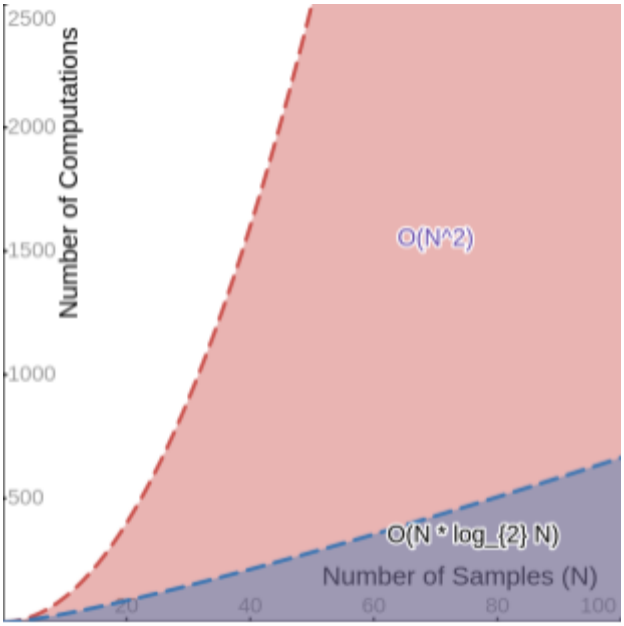

Fig. 45. FFT versus DFT number of computations.

A separate class for ranging and velocity computation will ingest burst readings from the sensor observer. Applying the FFT to the columns of the burst data matrix whose columns denote the chirp index and whose rows denote the sample index it is possible to extract the ranging information. Then, by applying the FFT a second time to the rows of the previous FFT-transformed data matrix it is possible to extract the velocity information. The final result is a matrix whose rows contain ranging data and whose columns contain velocity data. It is then possible to refine this data by applying the Constant False Alarm Rate (CFAR) algorithm to filter out noise. The CFAR algorithm averages neighboring cells to a test cell to establish an interference baseline, where if this baseline is exceeded then noise is assumed to be present and ignored. This ranging and velocity extraction implementation is visualized in the Figure 46 below.

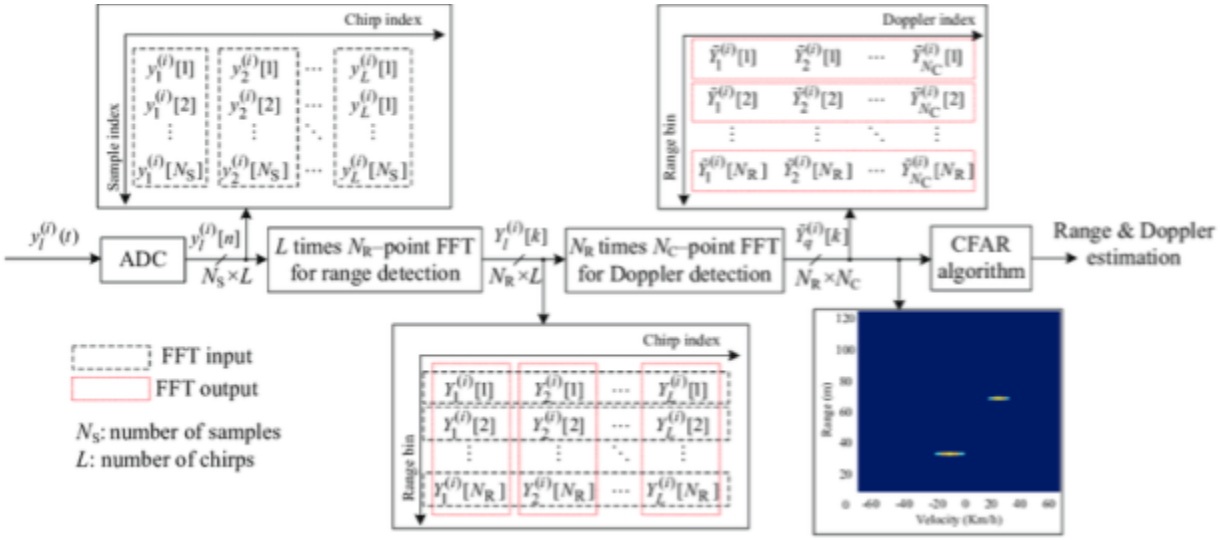

Fig. 46. Ranging and velocity extraction from FMCW burst data. Credit: Kim, Bong-seok & Kim, Sangdong & Jin, Youngseok & Lee, Jonghun [15].

6.5 Graphical User Interface

A Graphical User Interface will be implemented to simplify interacting with the FMCW radar module. The Visualization Toolkit will be used to visualize the ranging & velocity data and to implement widgets such as buttons & text boxes. It will have buttons to change between the “OFF”, “IDLE”, “SLEEP”, and “ACTIVE” states of the state machine, and display which of the states is currently active. It will also have sliders to modify parameter values constrained within their ranges, which can only be modified when not in the “ACTIVE” state. Given this behavior, the user will be locked out from making modifications to the parameters, but can still see what their values are currently set to.

The ranging & velocity data will be visualized in a 2-dimensional graph that is updated in real-time. The ranging values will be in units of meters and represent values along the Y-axis and the velocity values will be in units of meter-per-second and represent values along the X-axis. These power values will be color-coded with dark blue at the low end of the spectrum and bright yellow at the high end of the spectrum for easy visualization. Additionally, the GUI may implement a third dimension of time with units of seconds. This would allow for 3-dimensional visualization over a set period of time by maintaining a first-in first-out queue of range & velocity data. This would allow for easy identification of vehicles that may be accelerating or decelerating as they pass by.

An example rendering of the aforementioned layout and its features is demonstrated below in Figure 47. This rendering is merely an example and not reflective of the final product.

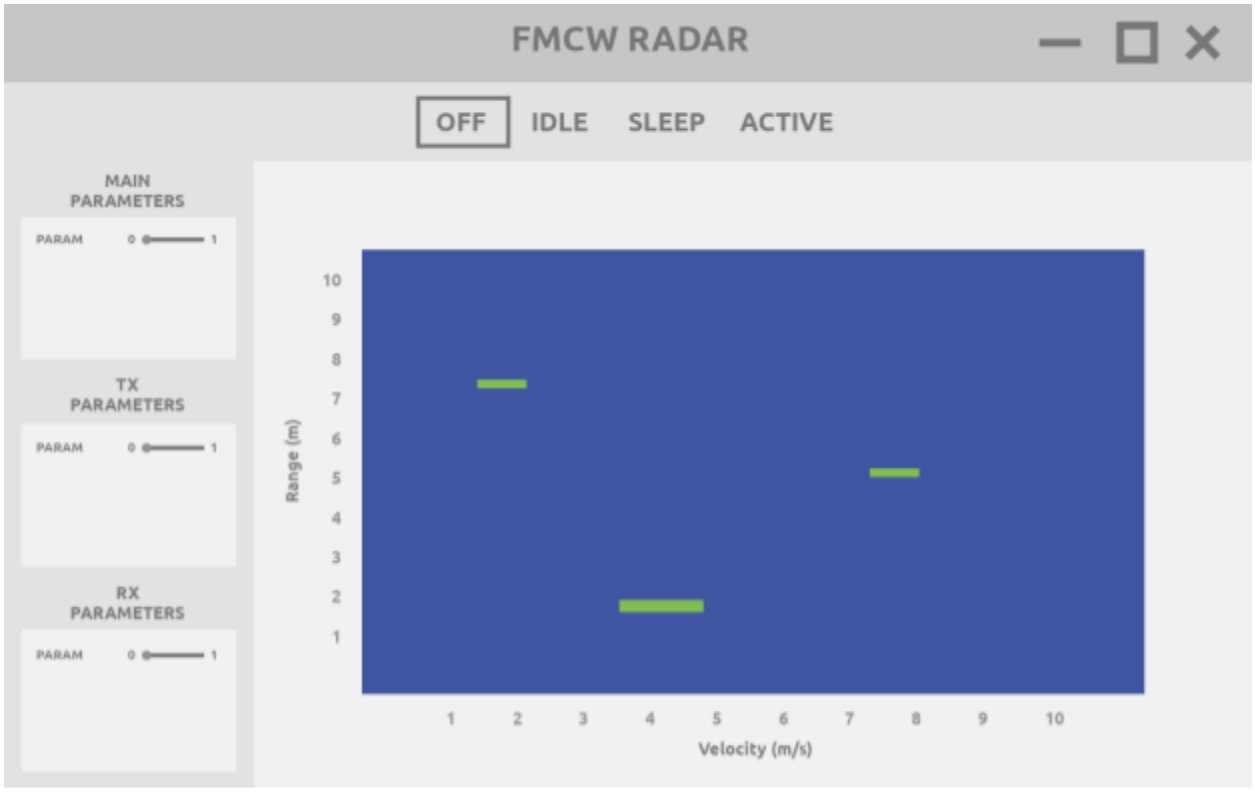

Fig. 47. The GUI concept for interacting with the module.

7.0 Project Prototype Construction

7.1 PCB Design Standardization

Our project consists of 5 different PCBs. The four PCBs our team will design are the power supply PCB, the PCB consisting of the low pass filter and gain stage circuitry, the ADC PCB, and the modulator PCB. The fifth PCB is the Arduino Zero, which comes already assembled and will be connected to our PCBs via Dupont wires. In order to maintain design consistency across our PCBs, a set of basic standards will be outlined.

The first standard has to do with component size. The footprints for specialized integrated circuits such as the ADC and modulator are already established, though their sizes were taken into consideration when the components were selected. Other common components such as resistors and capacitors come in a variety of sizes, and an appropriate size must be selected that takes into consideration the possibility of needing to test, replace or assemble the final circuits. Our team has opted to use 0805 components for resistors, capacitors, and inductors. They will be large enough to allow for hand-soldering if necessary, but not so large that they unnecessarily increase the dimensions of the PCBs. Additionally, because the 0805 components are surface mount (SMD), the PCB will be smaller and easier to assemble than if the components were all thru-hole.

The next standard is that our PCBs will have at most two layers. This design choice is primarily driven by the 2-layer limitation of the milling machine available to our team; additional details about the milling process are discussed in the next section. Though our PCBs will ultimately be manufactured and assembled by a professional supplier, it is good to have designs that can be made using the milling machine if necessary. In light of this, all of our PCB designs will have two layers at most.

The next standard defines how our PCBs will be interconnected. The Arduino Zero uses female 2.54mm pin sockets to connect to external signals; this is the socket that will be used for all of our input and output signals between PCBs. Male-to-male Dupont wires will then be used to connect nodes. In the case of RF signal processing components that need to be powered from a PCB, the connection would be a soldered lead on the component side and a male Dupont connector on the PCB side.

Additional standards are defined for the edges of our PCB. The outer shape of our PCBs will be rectangles, with the exception that the shape can be enlarged by combining two rectangles into a composite shape if needed. Mounting holes will be added at the corners of the PCB that measure 3mm in diameter. The distance between any edge of the PCB, whether this is the outer edge or the edge of a mounting hole, must be at least 5mm.

Finally, our PCB designs will adhere to a set of requirements for trace length, component spacing and via size. The traces for our PCBs will all be 0.025mm in width. The milling machine previously mentioned has spacing limitation which restricts how close components may come to traces. To allow enough space for traces to run between components, pads will all be at least 1.25mm apart. Lastly, all via points will be 0.8mm in diameter with a 0.4mm hole diameter.

7.2 PCB Prototyping

In order to build a functioning prototype, we will need to mill PCBs in-house for quick testing and prototyping. We will use the Quick Circuit J5 shown in Figure 48 for this purpose.

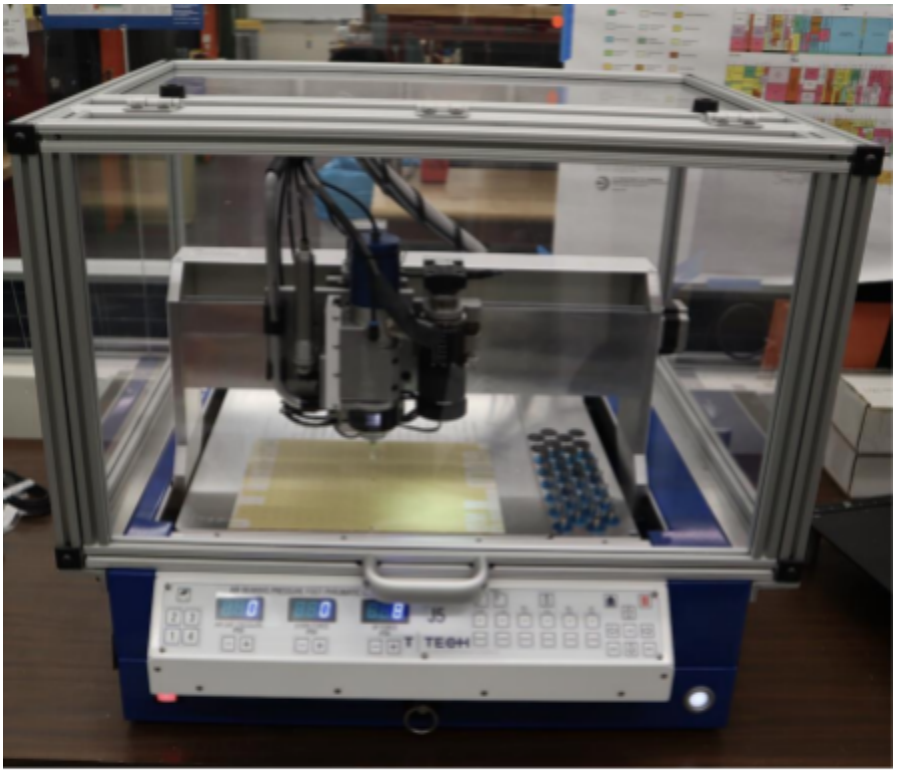

Fig. 48. Quick circuit J5 PCB prototyping machine

The Quick Circuit QCJ5 is a three axis micro-milling and drilling machine. The main purpose of this machine is to quickly create cost effective prototype printed circuit boards. This allows us to greatly reduce design-to-test cycle times. Micro-milling is distinct from conventional milling due to increased precision and damping which leads to the capability to more accurate milling paths and drill holes. This also means that the components inherently have a tighter relationship with each other. For example, a small quality issue with the spindle may have a significant negative effect on the quality of the milled part. The revolutions-per-minute is much higher in micro-milling than in conventional milling. The size of the cutting tool can be extremely small (diameter of 25 micrometers is possible). This means that the length-to-diameter ratio is often high. This increases the likelihood of tool breakage. In addition, the stiffness, attenuation, and accuracy of the micro-milling machine have a significant influence on the quality of the parts. It is also very important to constantly monitor the micro-milling process because the cutting tool can easily damage the surface of the material or break. The chip removal mechanisms are primarily influenced by the material type and its properties followed by the tool geometry, scale of machining, and the primary process parameters (cutting speed, feed rate, depth of cut). The thickness of the removable material layer is limited; this is defined as the minimum chip thickness $\hbar m$.

Isopro is a mill-path generator and the CAD/CAM interface for the Quick Circuit QCJ5. Isopro is made specifically to be a user-friendly interface to drill, mill, route, and edit our

circuit board designs. An understanding of computer aided manufacturing and Gerber files is required to use the Isopro software.

The T-Tech has a complete set of precision fine-grained carbide tools. The tool kit includes milling tools, a contour router, and various sized drill bits. We will be cutting out our prototype PCB boards on a FR-4 copper substrate board (see Figure 49). FR-4 is a composite material composed of woven fiberglass cloth and an epoxy resin binder that is flame resistant. The copper board we will be using is an 8 mil 1.0 oz 12” x 9” double sided FR-4.

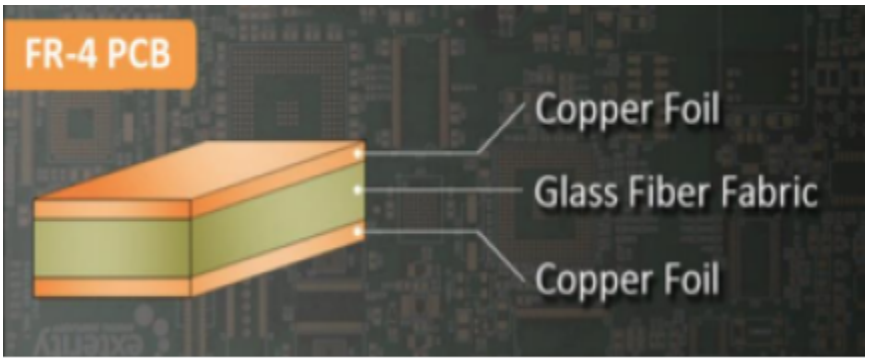

Fig. 49. FR4 Copper Substrate

The four available drill bit sizes that we carry in the milling lab as of date are 0.0320”, 0.400”, 0.0595”, and 0.125” as shown in Figure 50 from left to right respectively. We can clearly see the wide range of drill bit sizes we have at our disposal.

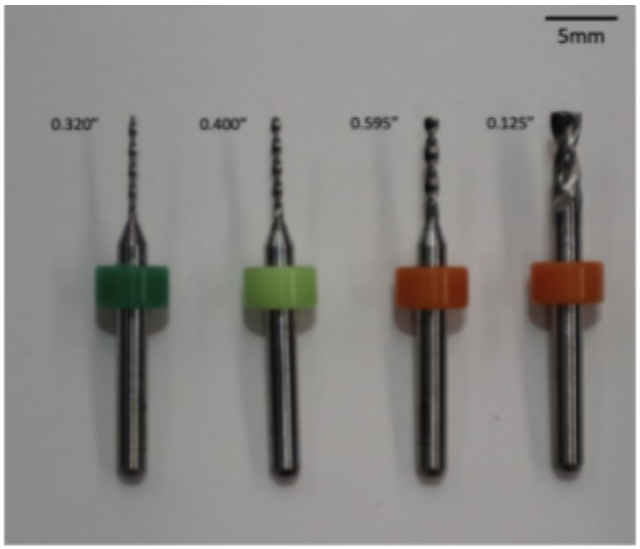

Fig. 50. Drill bits available in the lab

It is important to note the milling tool's pointed tip, as shown in Figure 51. This is significant because due to the pointed nature of the tool, we can control the width of our milled cut by the cutting depth of our tool in the negative z-direction.

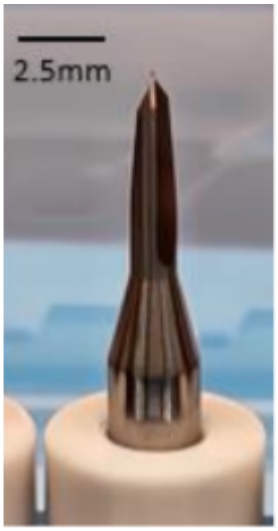

Fig. 51. T1 Milling Tool

In order to illustrate the process to go from a low-pass filter design I will provide an example of the process from a schematic all the way to a functioning prototype. In order to successfully test our design we need to be able to prototype a similar design first.

Before we use the Quick Circuit QCJ5, we must prepare the files we plan to import into Isopro properly. This will either involve converting a file from AutoCAD (*.dxf) to Isopro (.gbr) or from Eagle (.sch) to Isopro (.gbr).

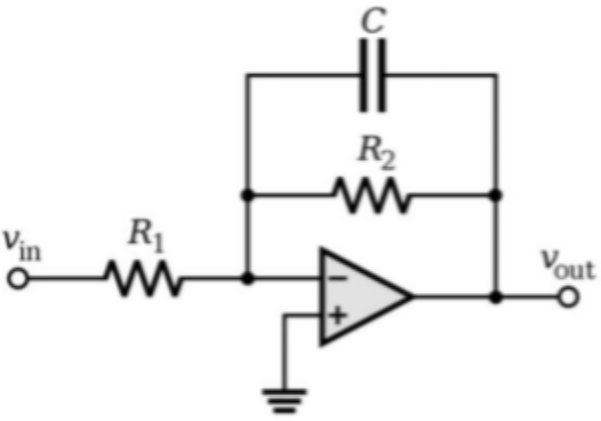

Fig. 52. Generic low-pass filter.

First, we would need to simulate the circuit in Multisim in order to confirm that the circuit will be behaving as expected. Below we verify that the design is indeed a low-pass filter by using an AC sweep.

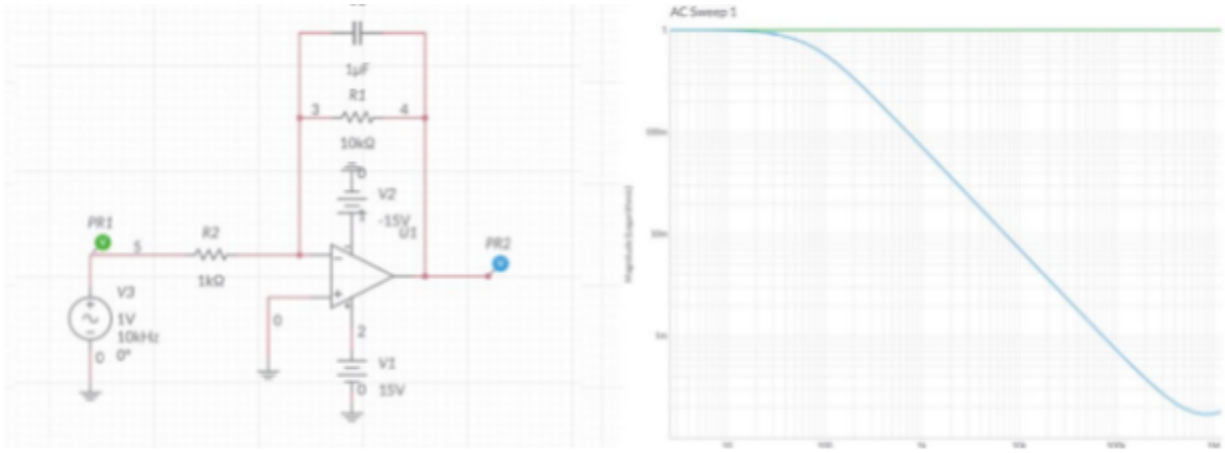

Fig. 52. Simulating the low-pass filter.

Next, we would need to design the circuit schematic in Autodesk Eagle. The LM741 op amp was used for this design.

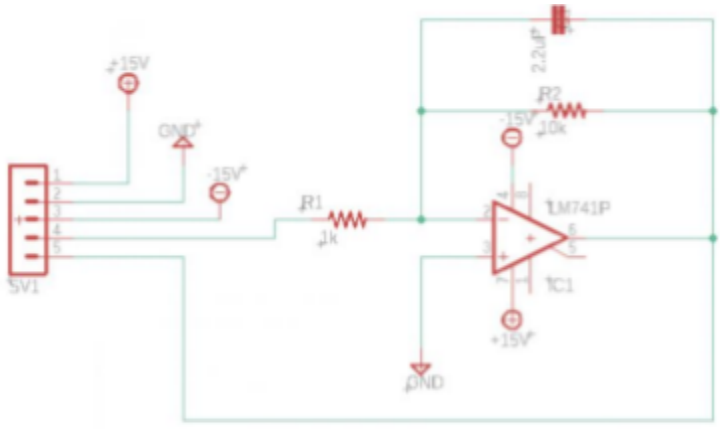

Fig. 53. Circuit design in Eagle.

Next, we proceed to convert the low pass filter design in Eagle to a .brd file. After arranging the board as desired, it should look like Figure 48. For more detail of how to correctly create the low-pass filter, reference the week 3 document for EEL 3926L.

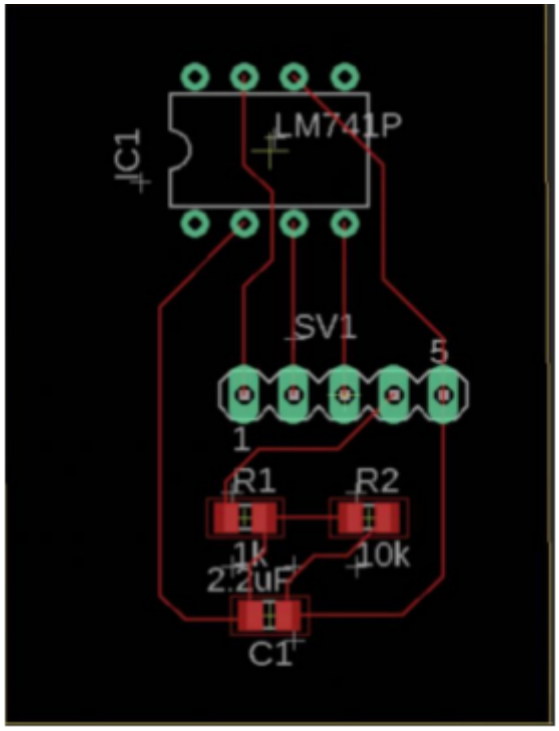

Fig. 54. Circuit in board form.

Once we have a satisfactory board layout that has passed the ERC check, we can now select File > Cam Processor > Process Job.

Now that we have properly prepared our design for importation into Isopro, we are now ready to start using the Quick Circuit J5 and Isopro to mill our design to completion.

Now that we are ready to use Isopro to prepare our design for milling we first will open Isopro on the computer in the milling lab. We should see the opening screen as shown in Figure 55.

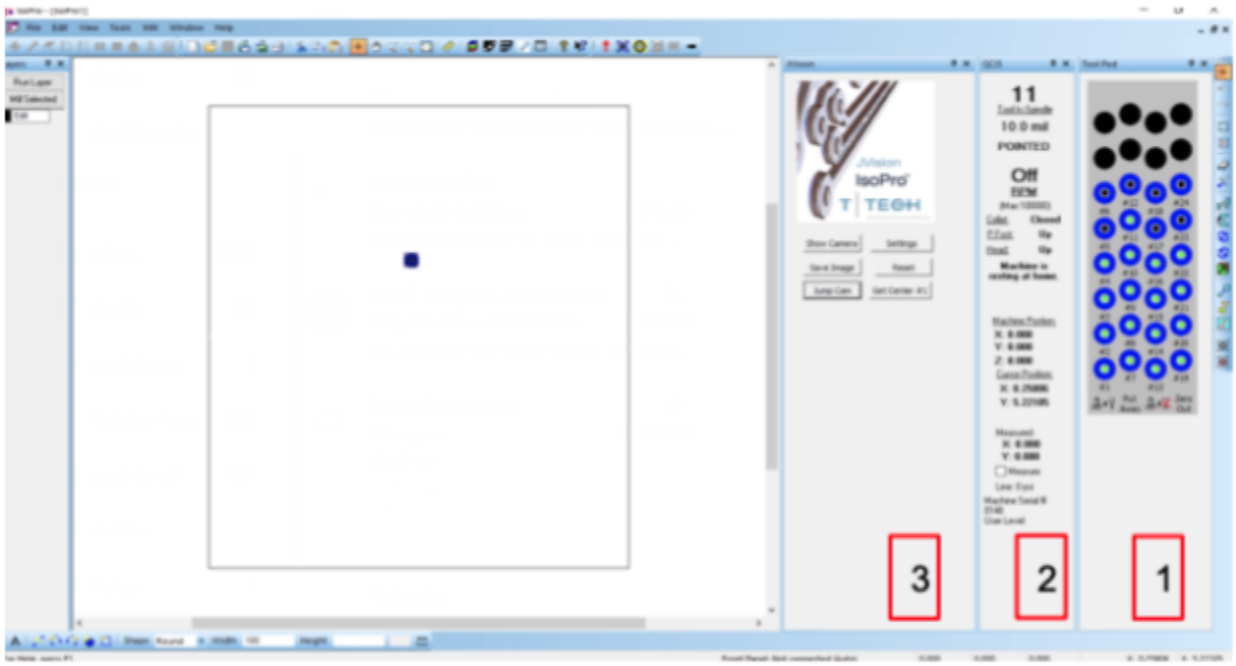

Fig. 55. Isopro home screen.

Once we are at this home screen we can go ahead and select “Mill” > “Initialize”. The Quick Circuit J5 should now initialize itself and place the spindle over the home position. While the machine initializes, let us quickly go over some of the prominent features that we see on the Isopro home screen. The tool pod, informational window, and camera window are marked by the numbered boxes from 1 to 3 respectively.

After opening Isopro, we can now select File > Import > Auto-Detect File(s), and select the Gerber files associated with our board. We can now see our board layout in Isopro.

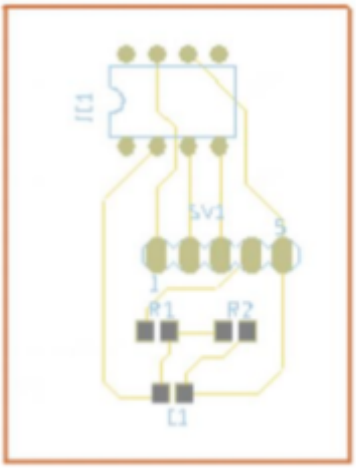

Fig. 56. Low pass filter opened in Isopro.

We next need to change the layer where the holes should be to a drill layer. We should now see the drill icon where the pins of the op-amp should be. It is useful here to hide any unnecessary layers to have a smooth workflow.

Fig. 57. Drill holes added.

Notice here that the Isopro does not insert drill holes for the 5-pin header. After testing multiple types of pin headers, it seems like Isopro does not register the holes needed for pin headers. We will have to put in the holes manually here. The most efficient way to do so is simply copy the holes created for the op-amp and place them where the holes for the 5-pin header should be. We also need to delete the drill holes that are located where the resistors and capacitors will be placed, as we will be using surface mounted components. We also need to delete the solder pads located on the 5-pin header.

Fig. 58. Drill holes added to 5-pin header.

Next, we want to isolate the copper_top.gbr layer (yellow layer). I have selected a 10-mil isolation. We then will hide the original layer leaving only the isolated layer (green). We can also create our board outline by creating a new layer and then using the rectangular tool. Our filter in Isopro should now look like Figure 59.

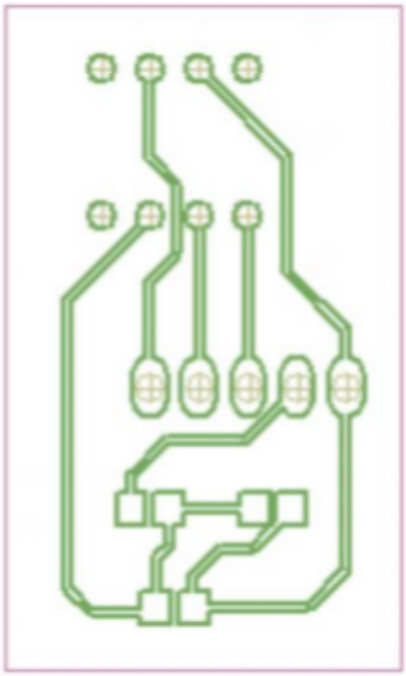

Fig. 59. Isolation created for components.

We now need to position the board in the desired location. Put the target over the circuit and do a material height check. Be sure to use the “Jump to Cursor” function described in the last section to make sure that the design is within the bounds of the copper board. We are now ready to run our layers. At this stage insert the drill bit into the collet. We will first run the drill layer. Once you reach the “Manual Tool Change” window, simply press the “OK” button. For the depth of cut, we will set the drill to about 2 mm which is around 78 mil. We now can run our isolation layer. We will unload the drill bit and the QCJ5 will automatically pick up the milling tool.

Now we can proceed to cut out our board using the tool cutter located to the right of the Quick Circuit J5 and place our components. In order to guide our PCB design creation it could be useful to understand the limits of the milling machine. Notice the milled copper traces shown in Figure 60. It is important to observe how the pointed tool behaves as it cuts through the materia

Notice the milled copper traces shown in Figure 60. It is important to observe how the pointed tool behaves as it cuts through the material. The figure demonstrates the up-milling face (highlighted by the red arrow) and the down-milling face (highlighted by

the green arrow), where the direction of the feed is shown by the magenta arrow, where “f” is the feed rate, “a_e” is the width of cut, and “h_m” is the average chip thickness. As the tool moves forward while rotating clockwise, the cutting lip will engage the material. This begins the formation of the chip with a chip load of zero. As it moves forward the chip load increases until it reaches the center of the crescent. The tool then decreases its chip load back to zero leaving the green portion of the crescent (down-milling) minimum quality copper trace width that we can make with our currently available tools which is approximately 125µm. The red and green arrows correspond to the up milling and down milling respectively. The magenta arrow depicts the motional direction.

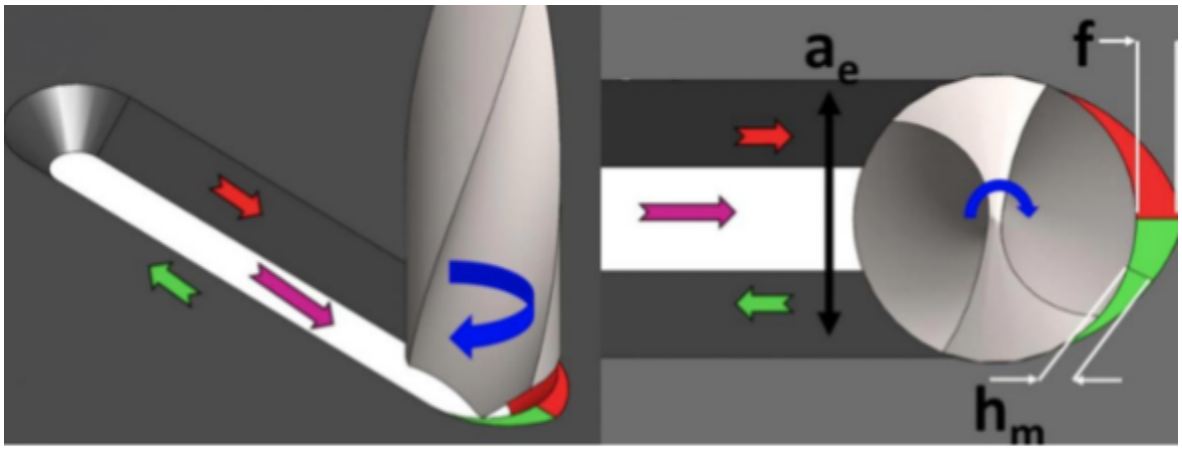

Fig. 60. Illustration of milling in detail.

Fig. 61. Highest resolution of the Quick Circuit J5.

7.3 Power Supply PCB

The power supply PCB includes the following elements: electronic components (those discussed in the hardware design section), 9V battery pack holders, and female pin headers. The electronic components are organized into three identical blocks, as the circuits for all three supply voltages are physically the same. The PCB incorporates four 9V battery pack holders arranged side-by-side. These holders are thru-hold components, so the positive and negative terminals of each battery are electrically connected to traces on the PCB, eliminating the need for additional jumper cables or wires.

Finally, the PCB includes 2.54mm female socket headers. There are a total of 21 headers, arranged into one four by five block plus one additional header. Each row in the four by five block corresponds to a certain voltage level generated by the circuitry, these being ground, +3.3V, +5V and +12V. Five pin headers are used for each voltage because the voltages must be sent out, via male-to-male Dupont wires, to all electronic components in the system that require a power supply. Each supply voltage level does not necessarily have five components connected to it, but having more headers than necessary allows for additional capabilities to be added or for the current design to be modified if desired. Additionally, the last single header is used for the enable signal. This is a digital signal from the MCU that will control all three EN pins in the power supply circuit, therefore only one pin header is needed.

Figure 62 below shows a close-up 3D rendition of the power supply PCB. Six holes per battery holder have been created, four for mounting and two for the thru-hole leads. Two additional holes have been added near the female header pins for mounting the PCB to the radar module. Note that in the figure, the battery holders themselves are the only elements missing.

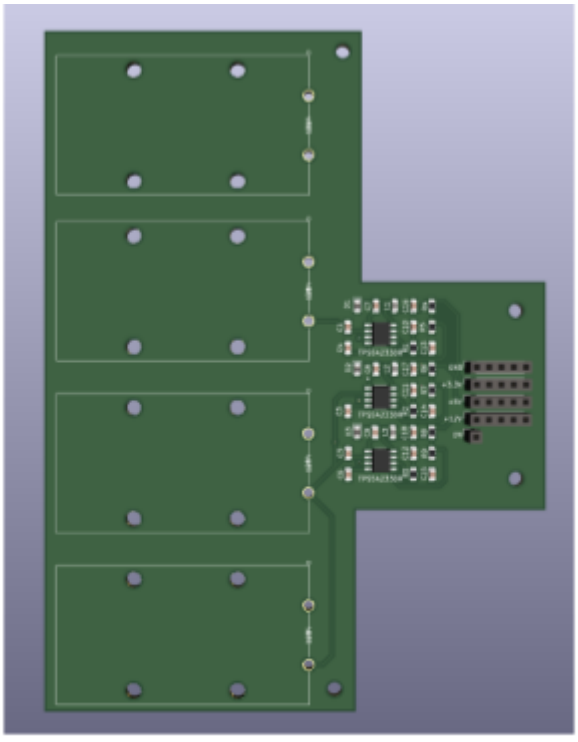

Fig. 62. Power supply PCB 3D rendition.

7.4 Modulator PCB

Our modulator PCB consists of the electronics for the modulator circuit and a row of pin sockets for the input and output signals. The modulator device is a thru-hole component. The potentiometers are thru-hole as well (Bourn 3299 packages), and their resistance is adjusted by a small screw at the top of the package. Holes are added at the corner of the PCB to allow for mounting on the radar module. Figure 63 below shows a 3D rendition of the circuit board.

The five in sockets have been labeled independently for this PCB. This is because each socket is a different signal or voltage level, unlike in the power supply PCB where all sockets in a row were the same node. Although the SYNC output signal from the modulator is not needed for our purposes, a pin socket has been allocated for it in the case that the design is modified later on.

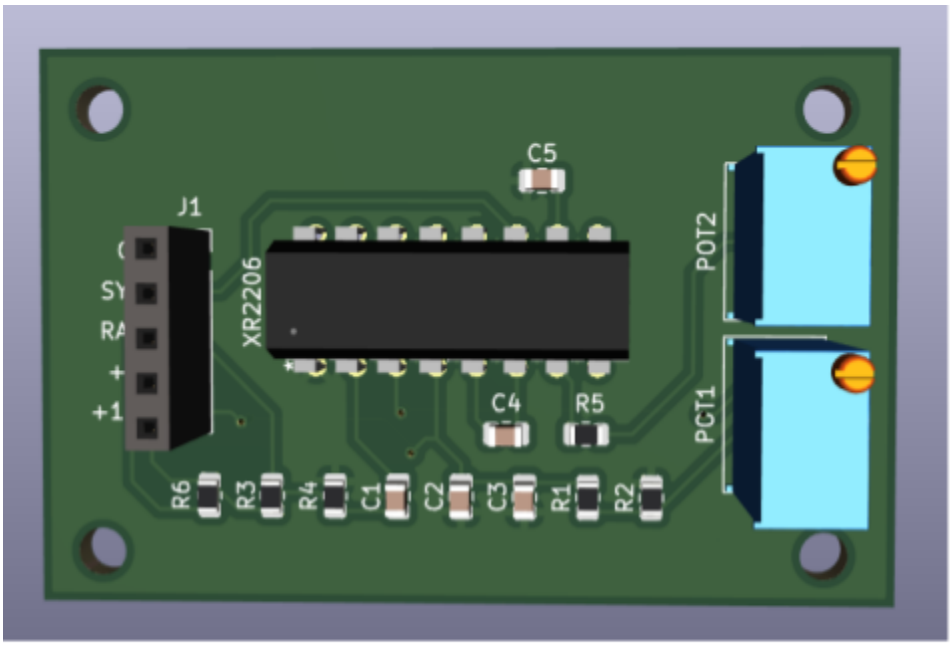

Fig. 63. Modulator PCB 3D rendition.

7.5 Gain Stage and Low Pass Filter PCB

The gain stage and low pass filter will be combined into a single PCB. The primary reason for this is that the LM324N includes the operational amplifiers needed for both circuits. The PCB includes all the circuit elements, one row of pin sockets for the input voltages and a single pin socket for the output signal. One important difference between this PCB and the others is the inclusion of an SMA connector. The input to the gain stage, which is the input to this PCB, is coming from the mixer. The mixer's ports have SMA female connections. The easiest way to connect the mixer's output signal to the

PCB would be with a male-to-male SMA cable. In order to do this, a female SMA connector must be integrated onto the board.

Figure 64 shows the cross-section of an SMA connector. On the left, there are three distinct metal pieces: the top, the bottom, and the middle. The top and bottom pieces (equivalently, the outer pieces) are the ground leads. The middle piece is the signal lead. When integrated into the board, the two outer pieces will be soldered to pads that are connected to the ground plane. The signal lead will be connected to a trace that feeds into the gain stage electronics. The SMA connector used for our PCB is a horizontally-oriented female SMA connector.

Figure 65 shows a 3D rendition of the low pass filter and gain stage PCB. The only missing element is the SMA connector..

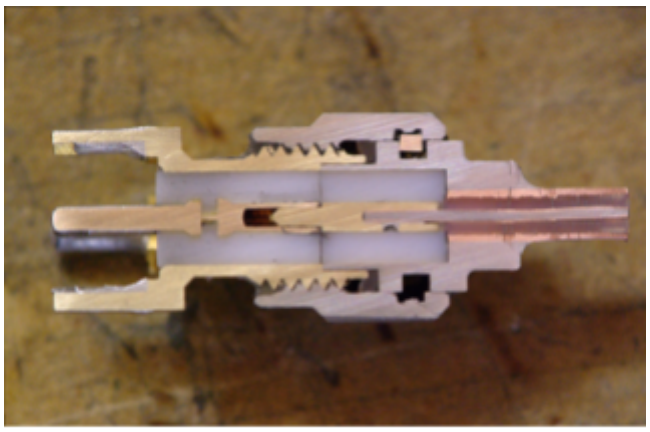

Figure 64: Cross-section of an SMA connector.
Credit: TubeTimeUS.

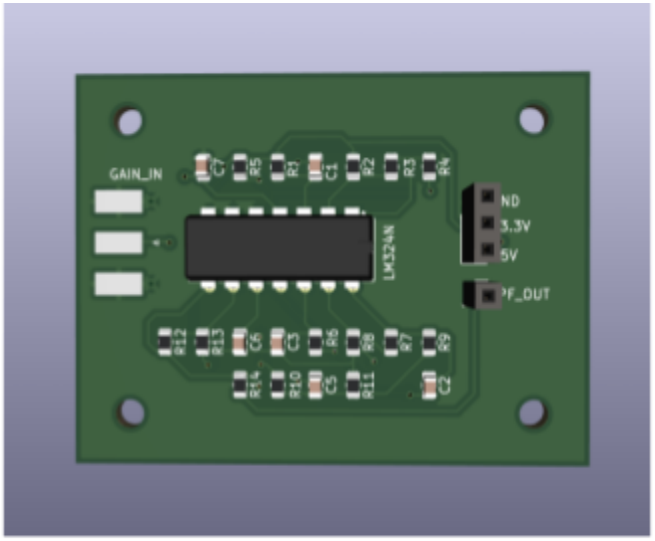

Figure 65: Low pass filter and gain stage PCB 3D rendition

7.6 ADC PCB

The analog-to-digital converter PCB consists of the following two ICs: the PCM1802DBR (24-bit 96kHz ADC) and the XLH535049.152000X (49.152MHz clock oscillator). The XLH535049.152000X was chosen specifically to target the PCM1802DBR's maximum sampling frequency of 96KHz, or 512 times greater than that of the sampling frequency. This oversampling is performed to improve both the resolution of the analog-to-digital converter and the signal-to-noise ratio, yielding a more accurate measurement.

At the "ADC_IN" a 1µF capacitor is placed in order to target an 8Hz cutoff frequency of the high-pass filter of the analog-to-digital converter, which is computed given that the device has an input impedance of approximately 20kΩ. This input signal node is tied to both pins "VINL" and "VINR" as we are only concerned with the one signal. Two 0.1µF capacitors are placed at both "VREF1" and "VREF2" to ascertain that the references have low source impedances. Additionally, two 0.1µF bypass capacitors are placed at both "VCC" and "VDD" to filter out any noise from our power supply. The mode select pins of "~PDWN" and "MODE0" are tied to a LOW digital signal via a 1kΩ resistor and the pins of "BYPAS", "OSR", "MODE1", "FMT0", and "FMT1" are tied to a HIGH digital signals via a 1kΩ resistor. The XLH535049.152000X has a 0.01µF bypass capacitor placed at its "VDD" to mitigate power supply line noise.

Three female connectors are used to allow for input and output connections between our various PCBs. A 1x1 female header is used to connect the "LPF_OUT" and the "ADC_IN" nodes, a 1x3 female header is used to connect the +5V, +3.3V, and GND of the power supply, and a 1x4 female header is used to connect the analog-to-digital converter's output signals of "FSYNC", "LRCK, "BCK", and "DOUT" to the Arduino Zero for digitization.

A 3D rendering of the analog-to-digital converter PCB can be visualized in Figure 66 below.

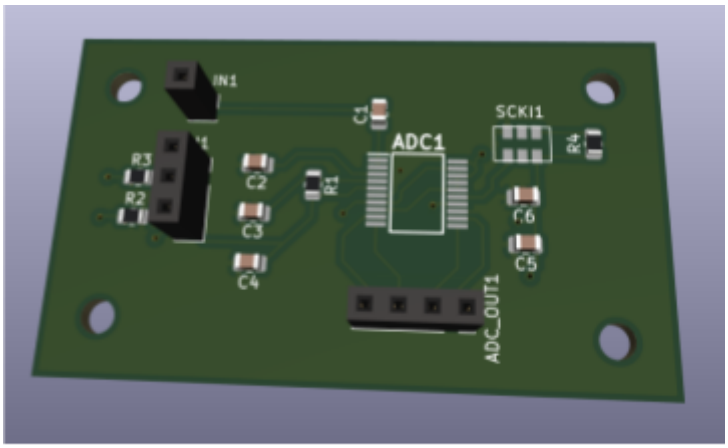

Fig. 66. ADC PCB 3D rendition

8.0 Project Prototype Testing Plan

8.1 Required Equipment

There are several pieces of equipment that will be necessary when testing our subsystems. Each piece of equipment either acts as an input to our system, measures an output of our system or simply measures characteristics of components. Calibrated lab equipment is necessary to verify that our subsystems are operating as desired.

8.1.1 Multimeter

The first piece of equipment needed is a multimeter. A multimeter is a device that can be used to measure several different parameters including DC voltage, AC voltage, capacitance, resistance, and current. The most important function of the multimeter is voltage measurement, as the voltage at different points of our circuits are known and should be at a certain level for the circuit to operate correctly. Multimeters will also be used to verify that our components have the correct characteristics. Circuit elements such as resistors and capacitors come with a nominal value, but their actual value can and should be measured. This helps better understand the operation of the constructed circuit, which will deviate at least slightly from the theoretical performance.

Multimeter will also be used during our PCB testing process. One of the biggest issues in circuit design is lack of connectivity, which is when two points are not connected when they should be. Using a multimeter will help verify that all points that should be connected are actually connected.

We will primarily be using the HMC 8012 Digital Multimeter that is available in the senior design lab. This multimeter is highly accurate and highly reliable.

8.1.2 Network Analyzer

Network analyzers are especially useful in RF system testing. They can display the frequency spectrum of an RF signal, which is the graph of the frequencies present in the signal versus the magnitude of each frequency. Network analyzers can also be used to measure characteristics of our high-frequency signal processing components; some example parameters are S-parameters, voltage standing wave ratio (VSWR), and return loss. Much like circuit element nominal values, these parameters are often stated in a component's datasheet, but it is a good idea to verify them. All components will have characteristics that vary slightly from the designed nominal value.

For RF testing we will be using an Aglient PSA Series Spectrum Analyzer. This device offers high-performance spectrum analysis up to 50GHz.

8.1.3 DC Power Supply

DC Power Supplies are used to generate constant voltages for use as inputs to a circuit. In our circuits, DC power supplies take the place of batteries in our design. The voltage generated by a power supply such as a battery will eventually decrease as the supply is used, so a DC power supply is used in a lab setting to ensure the voltage is constant. In our project, DC power supplies are used to generate the power needed for our active filters and create a DC offset for the circuit outputs.

Like many pieces of lab equipment, one primary advantage of using a DC power supply is the ability to quickly and easily generate different voltages for application to the circuit being tested. It may be advantageous to test different power supplies for a circuit to determine which one creates the best output signals. If batteries were used, not only would the voltage not be constant, but the voltage levels available would be limited to integer multiples of the voltage for an individual battery. It would be a hassle to have to generate additional voltages for testing purposes.

We will be using the Keithley 2230-30-1 Triple Channel DC power supply. The Keithley power supply provides two channels capable of producing up to +30V and up to 1.5A; with one additional channel capable of producing up to +6V. Each channel can be independently controlled to be on or off at any time. To aid with device-under-testing (DUT) the device has a timer capability that allows us to set up unattended tests that turn off the channels after a programmed time-interval. A USB can also be connected to facilitate instrument control, data logging, and analysis.

8.1.4 Oscilloscope

An oscilloscope is a device used to measure output voltages from a circuit. An important characteristic of the oscilloscope is that it can measure both the time domain and frequency domain versions of a signal. In the time domain, an oscilloscope is used to display and measure the shape of a signal. For a DC signal an oscilloscope would simply show a straight horizontal line in the time domain and a straight vertical line in the frequency domain. In this case, the device acts more like a multimeter. For AC signals, the oscilloscope allows the user to view the period and frequency composition of the signal being analyzed. There are a plethora of useful features on an oscilloscope, such as cursors for measurement, multi-signal computations, and the ability to toggle between AC and DC coupling, which either ignores the DC offset in a signal or leaves it in, respectively. For our applications, the oscilloscope is used to verify the gain of our gain stage, the frequency response characteristics of our low pass filter, and the signal shape generated by our modulator.

We will be using the Rohde & Schwarz RTM3004 Oscilloscope. This device not only provides us with an oscilloscope but also with a logic analyzer, protocol analyzer, and digital voltmeter. This will allow us to quickly and efficiently debug our electrical systems. We can also do frequency analysis using this device which may be useful when testing our system.

8.1.5 Function Generator

Function generators can create output signals of various shapes, frequencies and amplitudes. These signals can be used as input signals for our circuits during design verification. Nearly every circuit in our design modifies an input signal in a certain way; it does not just generate a signal. However, if each circuit required the output signal from the previous subsystem in order to be tested, we would have to verify the functionality of each subsystem in a sequential fashion, which would be a very lengthy process. Alternatively, we choose to use the function generator to simulate the output signal from the previous subsystem. Another advantage of initially using a function generator rather than using the output from the previous system is that the signal from the function generator can be easily modified. At the press of a button, the frequency, magnitude or shape of the signal can be changed to acquire a better understanding of how the circuit being tested operates. A more complete idea of the circuit performance can be developed this way.

We will be using the Tektronix Dual Channel Arbitrary Function Generator. The AFG3022B has 2 analog channels and has a 25MHz bandwidth. It has a maximum output frequency of 25MHz. Its maximum amplitude is $20V_{pp}$ into 50Ω. It also conveniently has a USB connector on the front panel for waveform storage. This functionality was actually used in this report.

8.2 Hardware Specific Testing

Hardware testing is to be programmatically implemented using Python and the “pyvisa” library that serves as a wrapper for the NI-VISA application programming interface. NI-VISA allows for communication with test equipment such as the network analyzer, DC power supply, function generator, and oscilloscope via USB or GPIB. By leveraging the capabilities of NI-VISA it is possible to issue SCPI commands to these devices, which allows for the setting of values and getting of measurements. By interfacing with these devices in this manner it allows us to design tests that require minimum interaction beyond initial composition while accumulating relevant data samples as quickly as possible. These tests will be verified against simulated data wherever possible.

8.2.1 Antenna Test

Required equipment for executing the antenna test would include the following: a reference antenna with known characteristics (gain, directivity, pattern, et cetera); a receiver system to determine how much power is received from the reference antenna; a radio frequency power transmitter to direct energy into the antenna under test; and a positioning system to rotate the test antenna relative to the source antenna to measure the radiation pattern as a function of the angle.

To test the antenna we must prepare the network analyzer and connect it to the antenna as is directed. The Voltage Standing Wave Ratio (VSWR) of the antenna must then be measured to evaluate how effectively power can be sent to or received by the antenna to ensure acceptable losses. The return loss of the antenna is then measured and is expected to be relatively high. The insertion loss of the antenna is finally measured and is expected to be relatively low.

Testing of antennas is not a trivial matter and is a fundamental skill for those looking to be well-versed in antenna theory. All the theory in the world will not help us if the antennas that we may be testing do not work as we would like. One of the most important devices for antenna testing is a vector-network analyzer (VNA). This device will allow us to measure the impedance of the antenna.

Network analyzers are crucial devices that help us characterize devices under test. This can include networks for radio frequency circuits or even microwave components as well such antennas. Using ‘black boxes’ we can help define networks as ‘black boxes’. Most commonly we use 2-port networks due to simplicity, but 3-port, 4-port, and n-port networks do exist based on how much we need to characterize from what we are testing. Parameters of interest in these networks are impedance, admittance, and scattering parameters. Network analyzers are more interested in looking for scattering parameters since we are taking radio frequency measurements. For our case, we will be using a 2-port network and define the following scattering parameters by how we are using this.

A narrow band signal is swept through the DUT through a calibrated reference plane. This allows direct measurement of the scattering parameters. Note that multiport networks are measured two-ports at a time terminating all ports.

Calibration is very important when using a network analyzer. This is because the very definition of S-parameters requires that the ports be matched. Calibration will also remove systematic errors caused by the measuring device and will promote accuracy. One should re-calibrate whenever settings are changed on the analyzer such as the frequency band and sweep time. If the test set-up was changed, or if someone else handed the analyzer. Electronic Calibration is usually the standard method.

In order to properly test our antenna we will need a couple of things situated first.
We will need a reference antenna, for which we know the characteristics, a RF power transmitter to inject energy into our antenna, a receiver system to determine the received power, and a positioning system to rotate the antenna to measure the radiation pattern as a function of θ. Dual-polarized horn antennas are usually used as reference antennas. We will have access to such equipment through Dr. Gong’s ARMI Lab.

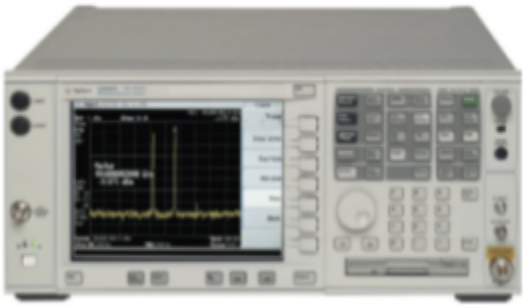

Fig. 67. Agilent 50 GHz VNA.

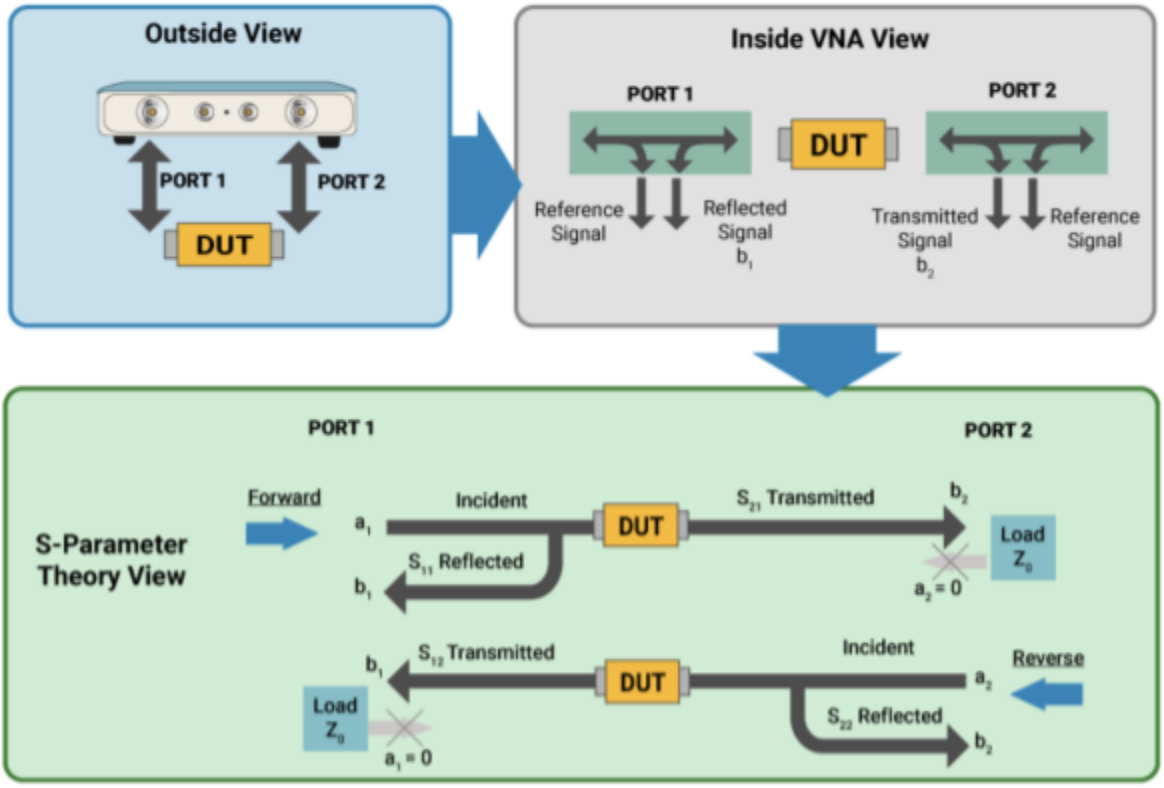

Fig. 68. Vector network analyzer diagram.

8.2.1.1 S-Parameters

The transmitting system will need to be able to quantify how much power is received from the test antenna. This will most likely be done using a transmission line connected directly to the antenna terminal using an SMA connector. We can also use a VNA and find the S-parameters. The behavior of a 1-port network is specified by one scattering parameter which is the reflection coefficient Γ.

$$\Gamma = \frac{Z_L - Z_O}{Z_L + Z_O}$$

The behavior of a 2-port network is specified by four scattering parameters where two are usually more relevant to us. The S-parameters of interest in a 2-port network are denoted S_{ij}, where S is a 2-by-2 matrix, i is the row index, and j is the column index. S contains the following values: the input port voltage reflection coefficient S_{11}; the reverse voltage gain S_{12}; the forward voltage gain S_{21}; and the output port voltage reflection coefficient S_{22}. If the system is linear and made entirely of reciprocal material then the network is reciprocal.

For a symmetric lossless 2-port network the following equations hold:

$$|S_{11}|^2 = |S_{22}|^2 = \Gamma^2$$

$$S_{12} = S_{21}$$

$$|S_{21}|^2 = 1 - \Gamma^2$$

For a 2-port network as shown in Figure 60, the following equations hold:

$$S'_{11} = S_{11} e^{-j\beta 2l}$$

$$S'_{12} = S_{12} e^{-j\beta 2l}$$

$$S'_{22} = S_{22}$$

$$S'_{21} = S_{21} e^{-j\beta 2l}$$

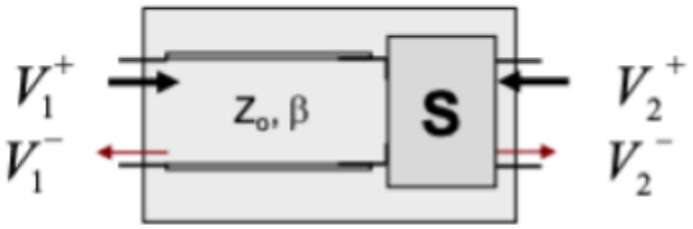

Fig. 69. A two-port network.

The beauty of these networks is that we can measure the electrical characteristics at points of interest without direct contact. In other words, the thing in the black box is completely determined by what it scatters. This allows precise measurement at points too small or delicate to be measured with a multimeter.

The S11 parameter is best defined as the input port voltage reflection coefficient. Essentially we are looking at the Voltage Standing Wave Ratio (VSWR), where we are focusing on how much of the signal is lost in transmission and how much is reflected. The value for the voltage standing wave ratio is generally in the range of [1, ∞) where the lower the value, the better the performance. Additionally for the reflection coefficient, if we have a value of 0 it means that no signal was reflected at all in the system. The input return loss should ideally be high as well, to best see that no signal was lost when the network was under test. This would refer to the impedance matching of the system (we will discuss more in detail about later). S_{11} and S_{22} are arguably the most important parameters for testing most radio frequency networks.

The S_{21} parameter refers to the forward transmission which looks into the gain and loss of the device under test. The transmission coefficient is best described as the ratio of the amplitude of the complex transmitted wave to the incident wave at the discontinuity of the transmission line. This essentially helps us in testing to see how the network behaves under test whether there are active networks within it or passive networks. For example, we are measuring an active network for let us say a receiver system with a low noise amplifier. We would expect to see an increase in gain from the forward transmission coefficient defined by S_{21} based on the measurement. If we do not, then we should be concerned something has gone wrong in the circuit (impedance mismatching, the low noise amplifier not being properly activated with DC biasing, et cetera).

The S_{12} parameter essentially looks at the reverse isolation coefficient. The reverse isolation coefficient is essentially the reciprocal of the forward transmission. We ideally

would look more into the same principles as described for S_{21} but it would be much easier to determine the isolation loss from here.

The S_{22} parameter has a similar functionality to the S_{11} parameter. S_{22} more focuses on return loss than S_{11}'s focus on insertion loss. Ideally here we will want the return loss to be very low, so we lose the minimal amount of the signal transmitted. This would also mean that we would want the reflection to be very minimal as well during transmission with the voltage standing wave measurement we discussed previously.

8.2.1.2 Impedance Testing

Testing the impedance of the antenna is of vital importance. If the impedance of our antenna is not close to the normalized 50Ω then we will have very little power transferred or received. The VNA testing will be able to give us the impedance readings as a function of frequency. If we are not able to access lab quality VNAs (they are quite expensive otherwise) then there are now cheaper handheld options that would suffice if necessary. A quality handheld VNA can be purchased for around a hundred dollars. Usually VNAs are on the order of thousands or tens of thousands of dollars, so this is one testing capability we may want to consider down the line.

The magnitude of the impedance mismatch is measured using the Voltage Standing Wave Ratio, or VSWR. The Voltage Standing Wave Ratio is a function of the magnitude of the reflection coefficient. This gives us a quick way to gauge how much power is being reflected by the antenna.

Impedance matching in the radio frequency domain is best described as a method or practice of tuning the input or output impedance for a specific value. We want this specific value to actually assist us to maximize the power transfer or make the signal reflection as small as possible. A smart way for us to actually do this would be to make Z_s and Z_L equivalent to each other, to maximize power transfer. There are two forms of impedance matching, single stub (most commonly used for passive radio frequency devices and circuits) and then lumped element matching (which are used for active radio frequency circuits commonly). More details are as follows:

Single stub impedance matching is largely done by inserting another transmission line (also known as a stub) with reference to the load. When we do this type of impedance matching, we must always also consider the length of the stub line as well. When using this technique, the realization of the stub is normally terminated by a short/open circuit on the system. Using a short circuited stub is helpful in that it is less prone to leakage of the signal and easier to physically realize. An open circuited stub can be more practical with transmission lines such as microstrips since we would have to drill into insulating substrate to short circuit the two conductors of the line.

In lumped element impedance matching we use capacitors and inductors. This technique is popular with RFICs and active circuit microwave circuits where this becomes much more prevalent. Otherwise, all similar traits exist as we have discussed before. The key differences will arise when we attempt to use a Smith Chart which is what we will discuss in the next section.

8.2.1.3 The Smith Chart

Smith Charts are an important tool to help RF engineers create matching circuits of transmission lines. These charts appear as shown in Figure 70 below.

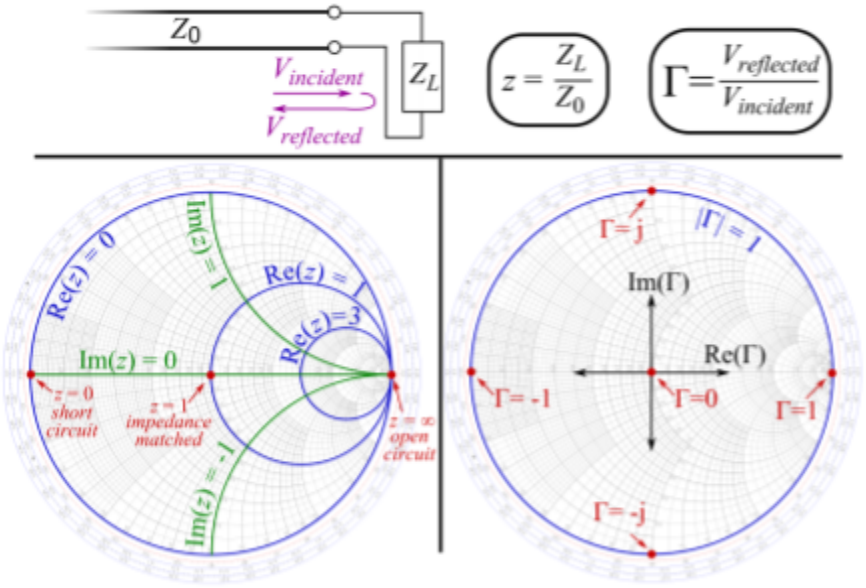

Fig. 70. A Smith chart. Credit: Sbyrnes321.

The upper portion of the graph is called the “inductive region” where impedances are of the form $Z = sL$ and the lower portion of the graph is called the “capacitive region” where impedances are of the form $Z = 1 / (sC)$. On the right side the impedance tends toward infinity where the transmission line becomes an open circuit. On the left side the impedance tends toward zero where the transmission line becomes a short circuit. Impedance matching is achieved near the middle where impedance tends toward one. The chart can also be flipped vertically to look at the admittance of the microwave circuit. The impedance or admittance must be normalized prior to using the Smith Chart.

8.2.1.4 Radiation Pattern & Antenna Gain

Once the equipment and the antenna range are set in motion, we must next look into using the source transmitting to the antenna under test with a plane wave from a given direction. It is important to note that the polarization of the antenna and the gain should be known. Reciprocity lets us know that the radiation pattern for both receive and transmit mode will be identical. We like to measure peak gain using the Friis Transmission Equation and a “Gain Standard” antenna. A Gain Standard is a test antenna that has an accurate gain and polarization (preferably linear).

Due to our wave being circularly polarized the wave will have equal amplitude components in both orthogonal directions. The received power can be found accordance with the Friis Transmission Equation below:

$$P_R = (P_T G_T G_R \lambda^2)/(4\pi R)^2$$

Once these tests are completed, we can compute efficiency/directivity from the radiation power (gain may not be needed) as follows:

When doing test measurements, it is important to know that we are considering doing measurements within the Far Field Region. The radiation pattern does not change the shape given the distance of “R”. The far field region is dominated by radiated fields, with the electric and magnetic fields perpendicular to each other and in the direction of propagation. The following equations must be satisfied for operation in the far field region with $R = Distance$, $D = Linear\ Dimension\ of\ the\ Antenna$ and $\lambda = Wavelength$.

$$R > \frac{2D^2}{\lambda}$$

$$R \gg D$$

$$R \gg \lambda$$

8.2.2 Power Supply Test

The power supply has two +9V alkaline batteries in parallel that are connected in series with another two +9V alkaline batteries in parallel. This arrangement allows for approximately 1200mAh of charge as well as two nodes at +9V and +18V. The +9V node is supplied to two separate buck converters that regulate the voltage to +5V and +3.3V, whereas the +18V node is supplied to a buck converter that regulates the voltage to +12V. In order to verify the fidelity of these switching regulators, their output voltages

and their ripples must be analyzed. Ripple is the result of converting an AC voltage to a DC voltage where variation is not completely suppressed following rectification. In the case of an ideal regulator there is no ripple. This ripple can be quantified with the “ripple factor,” where

$$\gamma = \sqrt{{V_{DC}}^2 + {V_{AC,RMS}}^2} / V_{DC}$$

and a value of $\gamma = 1$ indicates that no ripple is present.

To test the power supply the DC power supply must be connected to supply +9V to both the +5V and +3.3V regulators and it must be connected to supply +18V to the +12V regulator. The oscilloscope must be connected to the outputs of the +3.3V, +5V, and +12V regulators for monitoring. The root-mean-square voltage can be measured from the oscilloscope by issuing the SCPI command of “MEASure1:MAIN VOLTage:RMS”.

8.2.3 Modulator Test

EVT testing is a fundamental portion of our testing procedure for the modulator circuit, since we are required to make sure the circuit is functional at the base level. There are two main types of testing, which are Acceptance Testing Plan (ATP) and Design Verification Testing (DVT). The point of EVT is to make sure our circuit is working at a basic level that performs design goals and specifications.

Acceptance Testing Plan is base level testing to ensure minimum specifications or contract specifications are met for a specific design. In the case we have here, we are doing physical testing to ensure we are receiving the proper and adequate signals that are being sent once we turn on the signal on the PCB with the given power supply. For the Acceptance Testing Plan, we are running multiple portions over a set number of time so we are able to send out a signal to ensure the circuit is working properly. In the case of the modulator circuit, we are focusing on the reliability for the circuit to turn on and off without any latch-up occurring into the XR2206 function generator integrated circuit occurring at all. Latch-up is a big concern here, since this occurs very commonly in CMOS circuits which are a very common transistor to use in modern circuits because of the ease of fabrication and versatility of the circuit. Since we are not are not doing integrated circuit design for this we cannot go attempt to prevent the latch-up behavior so if this circuit does have a significant latch up behavior, we would need to find a suitable replacement for this integrated circuit instead. Another thing that should be considered for the acceptance testing plan, we would like to factor in whether we are getting the correct outputs from the modulator to see if we are getting the ramp function desired from it.

Design Verification Testing (DVT) is traditionally a much more intensive program that is designed after the objective of the specific design in mind, verifying that all the product specifications have been met completely. Some important considerations that are typically considered are reliability, function, climatic, electromagnetic compatibility & mechanical testing. Out of all typical testing done for design verification; we believe reliability, functional & mechanical testing are the ones we should be considering given our project. Our project is meant to be a proof of concept more so, so we are not having consideration of commercial or industrial use in mind. Functional testing will include if our modulator circuit actually works as intended, performance testing will relate more towards if the circuit is capable of actually making an adequate single that will not have too much margin of error & finally a mechanical testing would involve us physically dropping the PCB and shaking it around so we can go ahead ensure there is no electrical failure on the PCB due to mechanical stresses on the circuit.

To test the modulator we must first ensure that all of the components have been assembled properly onto the circuit and verify that there are neither short nor open components. The Keithley 2230-30-1 Triple Channel Power Supply must then be connected and turned on for DC biasing. The biasing for our XR- 2206 Waveform generator was +12V. Then the outputs of the device can be measured using the multimeter for DC measurements and the oscilloscope for AC measurements & waveforms. For our purposes the expected waveform should be that of a ramp signal with a period of 40ms.

In order to generate our desired waveform within our low-frequency circuit design we needed a waveform generator. We chose the XR-2206 for this purpose. The XR-2206 is a monolithic function generator integrated circuit that can produce multiple waveforms such as sine, ramp, pulse, and square waves with both high stability and high accuracy. The amplitude and frequency of the waveform can be modulated with an external voltage. Instead of directly modulating the voltage we can use potentiometers to change the resistance within the circuit. Due to the equation we are all fond of (V=IR) we can change the voltage and therefore the amplitude of frequency of our generated waveform. The oscillator has a maximum frequency of 1MHz and has a sweep range of 2000:1.

We were able to physically test our modulator section of the low-frequency design using a breadboard, as shown in Figure 71.

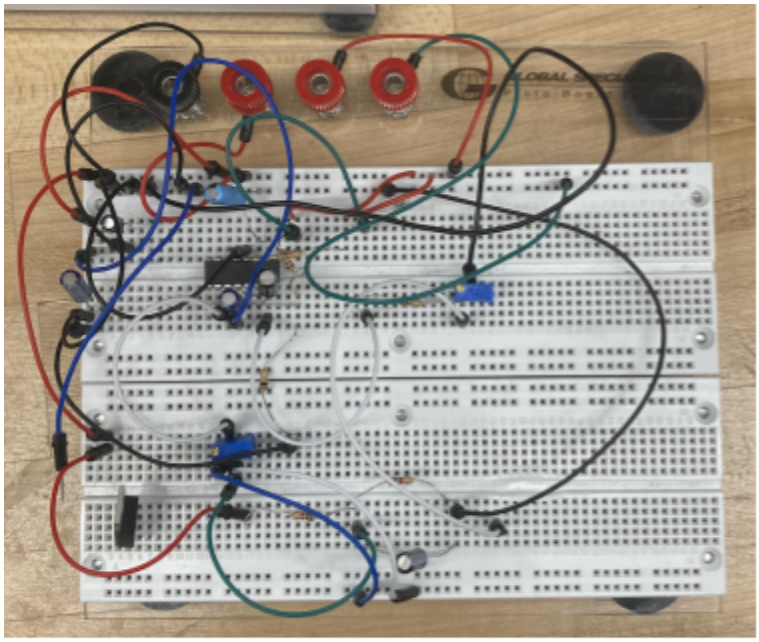

Fig. 71. Waveform generator testing.

After tuning the circuit by adjusting the potentiometers we were able to see the exact output of desired; as shown in Figure 72. The period of our generated ramp was 40ms as designed.

Fig. 72. Ramp output of modulator.

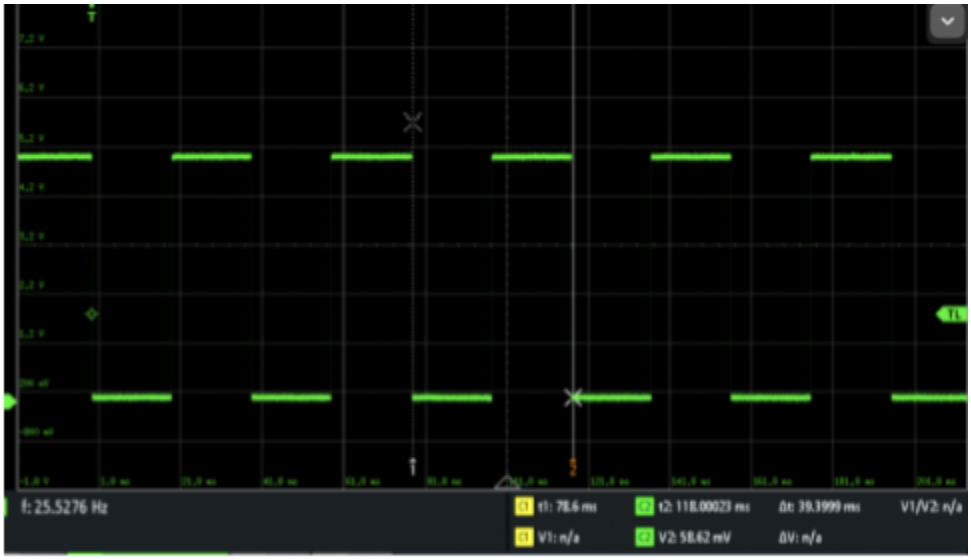

Fig. 73. Square wave with same period.

The next steps would be to build and test the gain stage and the two stage 15kHz lowpass filter. We then would be able to start to physically run tests on our radar once all low frequency sections are operational.

After initial testing we will proceed to design PCBs for the low-frequency sections of our circuitry. Using KiCAD we will design a double-facing PCB. We will minimize the space taken up by the design in order to facilitate implementation.

We will then be able to mill prototypes in the Senior Design Fabrication Lab. Once designs are verified we can then order professionally made PCBs for our final design.

8.2.4 Gain Stage Test

Engineering verification testing is a fundamental portion for the gain stage circuit, since we need to assess if the stages are actually capable of properly boosting the necessary signals on the low frequency end for the video amplifier. Tests for this will include, basic functional testing, power measurements, signal quality, conformance, electromagnetic interference, thermal and basic parameter measurements to ensure that the gain stages will be operating as intended. These tests will ideally be split between acceptance testing plans and device verification testing.

Acceptance testing plan for the stages, will largely include the base level functionality of the gain stages without going too in depth of the actual ability for the device to perform particularly well. One thing we must check, would be if the printed circuit board was fabricated properly and all the traces and routing was done to requirements without harming electrical integrity of the circuit. Once that is done, we must also go in depth on

if the components have been manufactured properly to be used on the board. Passive components such as capacitors or resistors are not typically of concern especially at low frequency operation, but we must really dive into it for our active components (which would primarily be the operational amplifier for this circuit). Another thing we would want to ensure operates properly onto this circuit, would be that the correct biasing appears onto the integrated circuits. The integrated circuits in question here are the operational amplifiers. DC biasing is an important component of any active network, since active networks have to (in most cases) amplify a signal for the gain to increase. Likely the final component of our acceptance testing plan here would probably include if the correct voltage and current value appear throughout the circuit. This would include us taking a multimeter device and physically measuring the current and voltage values through each component. We would want to comprehensively look at how each cascaded circuit operates properly and go along that path. This would be because is we look at it randomly, then we could take a long time troubleshooting since this design was done with a specific technique in mind. These values will be cross referenced the simulation we have done in Multisim comparing each specific value and figuring out if we have differences, and if so why, and additionally if these different values will affect operation.

Design Verification Testing (DVT) is traditionally a much more intensive program that is designed after the objective of the specific design in mind, verifying that all the product specifications have been met completely. Some important considerations that are typically considered are reliability, function, climatic, electromagnetic compatibility & mechanical testing. Out of all typical testing done for design verification; we believe reliability, functional & mechanical testing are the ones we should be considering given our project. Our project is meant to be a proof of concept more so, so we are not having consideration of commercial or industrial use in mind. Functional testing will include if our low pass filter circuit actually works as intended, performance testing will relate more towards if the circuit is capable of actually filtering the intended signal that will not have too much margin of error & finally a mechanical testing would involve us physically dropping the PCB and shaking it around so we can go ahead ensure there is no electrical failure on the PCB due to mechanical stresses on the circuit. In the specific case of the gain stages, it would be important to look into the reliance of the interconnects on the PCB to ensure they will not fail on us at all once the circuit goes into some potential mechanical deformation. A possible scenario for this would be once the automobile we are tracking the velocity starts coming towards the physical radar, how would the PCB handle the mechanical deformation for the gain stages? Given the scale of this project and its relative sensitivity, we would rightly guess that radar would not function as well once hit by the car. Though it would provide interesting insight on

what can be reusable for the system which would help reduce cost for a different iteration.

To test the gain stage we must first ensure that all of the components have been assembled properly onto the circuit and verify that there are neither short nor open components. Then the function generator is connected to the input, the oscilloscope is connected to the output, and the power supply is connected and turned on for DC biasing. The function generator and the oscilloscope are to be connected to the computer executing the test via USB2.0 Type A to USB2.0 Type B cables. The function generator must first have its function set to generate a sine wave with the SCPI command of “SOURce1:FUNCtion SIN”. An AC sweep is performed by implementing a loop that iterates through signal frequency values starting at 1kHz and stopping at a value that is greater than 1MHz, where each subsequent signal frequency value is multiplied by 1.5. At each iteration the function generator is to have its frequency set with the SCPI command of “SOURce1:FREQuency [frequency]” where “[frequency]” is the current frequency value. Upon sending this SCPI command to the function generator the testing computer must sleep the thread for approximately double the settling time in order to allow the signal to settle prior to reading measurements from the oscilloscope. Once settled, the frequency is measured from the oscilloscope with the SCPI command of “MEASure1:MAIN FREQuency” and the amplitude is measured from the oscilloscope with the SCPI command of “MEASure1:MAIN VOLTage:AMPLitude”. This data will be stored in a DataFrame for easy exporting to external file formats such as CSV or XLSX which will allow for graph visualization where attenuated frequencies can be easily verified. The measured AC sweep is expected to be comparable to the simulated AC sweep as shown in Figure 74 below. Between the frequencies of 20Hz and 1MHz the signal is expected to be amplified by an approximate gain of 50.

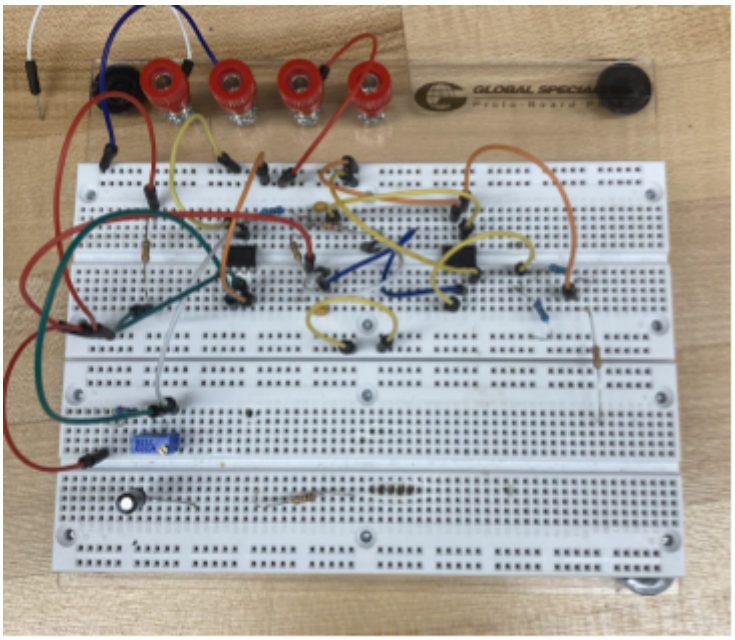

Fig. 74. Gain stage testing.

We are currently working on testing the Gain stage. We plan to build the stage on a breadboard and test its functionality. We will then design a PCB and mill a prototype. We then can finalize our design and order a board from a manufacturer.

Using the potentiometer we are able to modulate the voltage being supplied to the waveform generator and therefore change the amplitude and frequency of the waveform.

8.2.5 Low Pass Filter Test

Engineering verification testing is a fundamental portion for the low pass filter, since we need to assess if the filter is actually capable of properly filtering out the necessary signals that we do not need for the radar system. Tests for this will include, basic functional testing, power measurements, signal quality, conformance, electromagnetic interference, thermal and basic parameter measurements to ensure that the low pass filter will be operating as intended. These tests will ideally be split between acceptance testing plans and device verification testing.

Acceptance testing plan for the filter, will largely include the base level functionality of the low pass filter without going too in depth of the actual ability for the device to perform particularly well. One thing we must check, would be if the printed circuit board was fabricated properly and all the traces and routing was done to requirements without harming electrical integrity of the circuit. Once that is done, we must also go in depth on if the components have been manufactured properly to be used on the board. Passive components such as capacitors or resistors are not typically of concern especially at low frequency operation, but we must really dive into it for our active components (which would primarily be the operational amplifier for this circuit). Another thing we would want to ensure operates properly onto this circuit, would be that the correct biasing appears onto the integrated circuits. The integrated circuits in question here are the operational amplifiers. DC biasing is an important component of any active network, since active networks have to (in most cases) amplify a signal for the gain to increase. Likely the final component of our acceptance testing plan here would probably include if the correct voltage and current value appear throughout the circuit. This would include us taking a multimeter device and physically measuring the current and voltage values through each component. We would want to comprehensively look at how each cascaded circuit operates properly and go along that path. This would be because if we look at it randomly, then we could take a long time troubleshooting since this design was done with a specific technique in mind. These values will be cross referenced the simulation we have done in Multisim comparing each specific value and figuring out if we have differences, and if so why, and additionally if these different values will affect operation.

Design Verification Testing (DVT) is traditionally a much more intensive program that is designed after the objective of the specific design in mind, verifying that all the product specifications have been met completely. Some important considerations that are typically considered are reliability, function, climatic, electromagnetic compatibility & mechanical testing. Out of all typical testing done for design verification; we believe reliability, functional & mechanical testing are the ones we should be considering given our project. Our project is meant to be a proof of concept more so, so we are not having consideration of commercial or industrial use in mind. Functional testing will include if our low pass filter circuit actually works as intended, performance testing will relate more towards if the circuit is capable of actually filtering the intended signal that will not have too much margin of error & finally a mechanical testing would involve us physically dropping the PCB and shaking it around so we can go ahead ensure there is no electrical failure on the PCB due to mechanical stresses on the circuit. We would ideally like to have an input source to form a function generator to test this before actually putting this into the radar for a system integration.

To test the low pass filter we must first ensure that all of the components have been assembled properly onto the circuit and verify that there are neither short nor open components. Then the function generator is connected to the input, the oscilloscope is connected to the output, and the power supply is connected and turned on for DC biasing. The function generator and the oscilloscope are to be connected to the computer executing the test via USB2.0 Type A to USB2.0 Type B cables. The function generator must first have its function set to generate a sine wave with the SCPI command of “SOURce1:FUNCtion SIN”. An AC sweep is performed by implementing a loop that iterates through signal frequency values starting at 1kHz and stopping at a value that is greater than 1MHz, where each subsequent signal frequency value is multiplied by 1.5. At each iteration the function generator is to have its frequency set with the SCPI command of “SOURce1:FREQuency [frequency]” where “[frequency]” is the current frequency value. Upon sending this SCPI command to the function generator the testing computer must sleep the thread for approximately double the settling time in order to allow the signal to settle prior to reading measurements from the oscilloscope. Once settled, the frequency is measured from the oscilloscope with the SCPI command of “MEASure1:MAIN FREQuency” and the amplitude is measured from the oscilloscope with the SCPI command of “MEASure1:MAIN VOLTage:AMPLitude”. This data will be stored in a DataFrame for easy exporting to external file formats such as CSV or XLSX which will allow for graph visualization where attenuated frequencies can be easily verified. The measured AC sweep is expected to be comparable to the simulated AC sweep as shown in Figure 63 below. Between the frequencies of 3Hz and 15kHz the signal is expected to have an approximate gain of unity.

8.3 Software Specific Testing

8.3.1 Ripple API Test

In order to validate the performance and stability of the software it will be fuzzed and monitored using the "doctest" testing framework. "Fuzzing" is the testing process of deliberately inserting invalid data to various inputs and monitoring the result to verify that errors are handled appropriately. For example, the parameters specified by the Ripple sensor implementation have specified ranges, and should those parameters fall outside of the specified ranges calls to methods that modify them will fail and yield a return value of "BAD_INPUT". Similarly, in order to transition from one state to another state the state machine must be in the former state to begin with. If it is not in the former state then the method that is attempting the transition will fail and yield a return value of "BAD_STATE". Test cases for the Ripple API will be composed for each of the following methods: "turn_on", "turn_off", "go_sleep", "wake_up", "set_fifo_mode", "activate_config", "deactivate_config", "get_main_param", "get_main_param_range", "set_main_param", "get_tx_param", "get_tx_param_range", "set_tx_param", "get_rx_param", "get_rx_param_range", "set_rx_param", "start_data_streaming", "stop_data_streaming", "check_country_code", and "get_register". These methods may have multiple points-of-failure, as such may be composed of multiple sub-cases to assess the fidelity of these methods.

8.3.2 USB Interface Test

USB communication testing will be done independently of the Ripple API, as the USB communication is performed in a multi-threaded manner within the sensor interface. The request and response data structures will be fuzzed and communicated, where good requests are expected to receive a response with a "SUCCESS" status and bad requests are expected to receive a response with a "FAILURE" status. Additionally, the integrity of the communication channels must be assessed, such as timeouts or unforeseen data losses.

9.0 Administrative Material

9.1 Project Milestones

One of the most important elements of any project is the management. Having a solid plan with regard to technical design must come with an effective plan for executing each of the steps needed to achieve the team's goals in a timely fashion. We are already nearly at the halfway point of our project development, so a brief overview of our current progress will first be discussed.

9.1.1 Progress Recap

Since August of this year, our team has brainstormed different ideas for our project, narrowed down our project scope, and chosen a model on which we would base our

design. We eventually decided on a radar project that involved actively gathering velocity and ranging data from an environment. Acquiring the partially completed radar module from Dr. Gong accelerated our project timeline significantly by reducing the amount of work needed on the radar module itself. In fact, our team has made significant progress on the design of the radar circuitry, so the radar subsystem of the project is ahead of schedule compared to the software design and USB interface design. But in general, we are at the point where we have a solid understanding of exactly what subsystems should be in our project, so we need to design them.

To recap, here are the objectives that were outlined at the beginning of the paper: update and verify the MIT radar module circuit designs, design an operational USB interface, design a functional API, design the PCBs for the project, order hardware and materials, assemble the prototype and test the prototype. In addition to these objectives related to the technical aspect of our project, there are also administrative tasks that need to be handled. We will need to document all of the technical and financial information for our project, so time must be taken to write those documents. The remaining schedule for our project will now be outlined by month. Note that there are often tasks in parallel as different team members work on their respective subsystems, as well as handle any administrative tasks they have been assigned.

9.1.2 Design Verification and Part Acquisition

For the remainder of November, our team will be designing the USB interface, testing our radar module circuit designs and designing PCBs. The USB interface is a more complex subsystem in the sense that it has a serial communication constraint in addition to being implemented in a real-time system, so it requires more research than was done for the components in our radar subsystem circuitry. For the first two weeks of November, the components needed to implement the USB interface will be selected and acquired. For the last two weeks of November, the design will be tested and modified as necessary before solidifying the design. For the radar subsystem, the design is essentially complete. We are in the testing phase to ensure that our designs are performing as desired. Our gain stage design has already been verified, and the second week of November will be used to test the modulator and low pass filter circuits. Any necessary changes will be made to the circuits according to the results of the tests. Once all designs are tested and verified for accurate performance, the PCB design will begin. The PCB design is in progress now and should be completed by the end of November. Additionally, time will be spent documenting our design progress for a submission by November 4, 2022. It is especially important to make as much progress in November as possible considering that some days will be taken off for the Thanksgiving holiday.

Two weeks in December will be committed to active progress on the project, while the latter two weeks may have slower progress due to winter break. By the end of November, all of our designs should be completely tested, and the corresponding PCBs should be designed. December will primarily be committed to ordering hardware and materials for the project. Additionally, there is another draft of our documentation that

needs to be submitted. Because the designs for all of our subsystems likely went through some iterations since our previous drafts, it will likely be necessary to make modifications to the previous content as well as add more material to reflect any additional tests that were performed.

There is another aspect of our project that has yet to be mentioned: the potential of running out of time. Senior Design requires the project to be fully operational by April of next year. In the case that the complexity of our project creates too many setbacks to be completed in a reasonable amount of time, our team has selected a completely different project that we may default to. This project was one of the initial project ideas that was considered, and its general concept was very interesting, but it was much simpler than our radar project and would be less diverse from a technical standpoint. Our team eventually chose to use it as a back-up plan, and although our team is making excellent progress towards our current goals, it is still important to develop a back-up plan in case something unexpected happens to disrupt our scheduling. Research for this system has already been completed. December, being a partially inactive month as far as project progress, will be used to order the necessary parts for the back-up plan. The parts will be there in case our team ultimately decides to go with our secondary project.

9.1.3 Prototype Construction and System Integration

January marks the first month that our team will have all the required materials to begin constructing our prototype. “Constructing” is a term used fairly loosely in this sense, as part of our project is technically software based. The remaining subsystems’ circuits (USB interface, modulator, low pass filter and gain stage) will all be placed on a PCB. The first two months of January will consist of assembling each PCB that was designed, and the last two months of January will consist of testing those PCBs. Each PCB should be tested separately to better identify any issues there may be with the boards. It may be necessary to redo or reconstruct some of the PCBs in the case that the boards are damaged during the assembly phase. On the software side of things, January will be committed to developing the software API. Since it is purely software and its tests are somewhat independent of the PCB tests, the software API can be developed in parallel with the PCB construction and testing.

More likely than not, the testing in January will be slower than expected or require design modifications. For the sake of accurate scheduling, the end of January and the beginning of February will be dedicated as a “catch up” period of time. Any repeated tasks, such as reordering parts, reconstructing PCBs, and redoing tests will be done at this time. Again, this is more relevant to the hardware side of the project than the software side, although any failed software tests from January can and should be redone in the catch-up period to bring the software design up to requirements. This leaves the end of February for final testing and design verification. In the case that the design verification goes smoothly in January, then the schedule will skip the catch-up period and go directly to system integration, which is discussed next.

By the end of February, each of the individual subsystems should be completely tested and verified. The next step is system integration, where the output signal from one system is sent to the input port of the next subsystem. The best approach to system integration is intersystem test points. Even though each of the systems works alone, there may be effects created by connecting the systems. Testing the signal (or information) after each system allows us to verify whether the individual subsystems are still working after integration. An analog test point can be added after the radar module, and a digital test point can be added after the USB interface to ensure the information was correctly digitized. A similar approach would be to create larger subsystems from the defined subsystems. First, the USB interface can be connected to the software API, and a test input analog signal can be used (such as a single-tone note with a known frequency) to verify that those two subsystems are working correctly together. Next, the radar module may be connected to the USB interface and the output signal of those two modules can be tested. A complete system integration would follow.

Much like the assembly process, the integration process will likely take more time than expected. Time will be allocated in March to allow for last-minute design changes, hardware acquisition and further testing. Even in light of potential setbacks, the system should be completely operational by the end of March.

One thing to note is that the system integration phase is typically where the project timeline is evaluated. If the system tests imply that the project is unlikely to be completed by the required deadline, this would be the time to switch over to one of the back-up plans. The first fall back would be to simplify the project we are already working on. Simplification of the subsystem of the project that is not operating as desired may be the short-term solution needed to ensure the project deadline is met. If simplification does not solve the issues, then construction and system integration for the secondary project should begin. This should begin as early as possible after it is evident the primary project will not be completed on time. If personnel time can be properly allocated, the secondary system construction and integration may even be worked on in parallel with the primary system integration. This can be done if there is uncertainty as to whether the primary project issues will be resolved. The necessary parts for the back-up plan will have already been acquired, so the only steps would be to build the system and write the necessary software. In conclusion, either the designed primary project, a simplified version of the primary project or the secondary project needs to be completed by the end of March.

9.1.4 Prototype Demonstration

April will be the final month of the project. This is the month in which the system is demonstrated. The primary objective of the project is to track the velocity of cars on Pegasus drive, so taking the system out for a field demonstration is the objective for April. The results should be recorded even in the event that the system does not perform quite as well as expected. Ultimately, project success is contingent upon accurate results within a certain delta, so if possible, slight modifications to the hardware or software settings can be made to achieve better results. If the secondary

plan was ultimately used, then April would be used for a demonstration of that system instead. All results recorded should include the testing conditions, such as the system settings, the actual environment parameters (such as how fast the cars were actually going), and any other relevant information that would give a more complete description of the system performance.

9.1.5 Documentation and Further Development

Like the first semester of Senior Design, the second semester should involve consistent documentation of our progress. There will be significantly different content in our paper from the progress taken in January through April than for the progress made in August through November. It is generally a good idea to record both failures and success when performing system tests and integration. From an administrative standpoint, it helps to document exactly what changes were made to the project since the submission of past papers. From a technical standpoint, it helps to document exactly what issues were discovered from initial designs and potential solutions to such issues. The more details on the changes made to the project between the semesters, the easier it will be to understand the project development if the system needed to be upgraded or modified in the future.

Lastly, the best-case scenario should be considered. If our team makes significantly more progress on our project than expected, we will have time to focus on our other goals, specifically the ranging and control system applications. It is important not to jump ahead to other goals before it is verified that the projected primary goal completion date will be at least two to three weeks before its initial expected completion date. The process for achieving the other goals will not involve as much work as for the primary goal, as the changes to the system will primarily be in software. Once the necessary changes are made, additional project demonstrations can be completed, and the results for those demonstrations can be recorded.

9.2 Budget and Finance Discussion

As our team has gone through the project so far during the semester, we have realized that working with projects within the radio frequency domain tends to be more costly, especially on the RF component side of things. We therefore had to keep a relatively reasonable budget to ensure that our project did not break the bank, especially since we are financing the endeavor out of pocket.

Based on the MIT course project, we have been given a project bill of materials of $361.22 for all of our components. This thankfully was reduced even more, due to a kind donation for Dr. Xun Gong with all of the radio frequency modules we would need for our project, with the condition that we will be returning a fully functional radar back to him once we are done with Senior Design for him to use for educational purposes for future students of his to use or learn from.

Even with that, we will likely require more than just that amount of money to ensure everything is working functionally. We have enough components to test and prototype in case one variation works better than another or one variation gets destroyed in the process. A safe budget to keep in mind for now would be about $500 split amongst our 4 members which would be $125 per team member. This would ideally provide us enough resources to build a fully functional project within a margin of freedom to ensure our success with some leeway.

To develop the budget of the project, the team needs an estimated cost for all potential materials and parts. Since we have already been donated some parts, the actual current costs of those models will be used as the cost estimate. It is important to remember, however, that just because the part has been acquired, it does not mean the part will remain functional or ultimately be included in our design. It may turn out that a cheaper but higher performance version of the device becomes available, so the costs reflected in the following tables will represent the class of material or device (e.g., the approximate cost of 2 GHz to 4 GHZ modulators).

The materials for our project can be divided into groups, roughly based on which subsection a particular material belongs to. Our team has already been given the frame of the radar module, but in anticipation of the potential need or desire to redo the frame, the associated costs will be recorded here. Note that any hardware required to secure components to the frame is included in the group of frame materials, and dimensions for the hardware is negligible in the context of cost (all of the hardware is standard and quite small, so the cost should not be inflated). Table 4 shows the parts needed for the radar module frame. Prices are skewed to the high side if costs tend to vary with supplier.

Table IV
Budget for the Radar Frame

Part Name	Description	Unit Cost	Quantity	Subtotal
Screws	50 pack of screws	$4.00	1	$4.00
Nuts	50 pack of nuts	$2.00	1	$2.00
Bolts	50 pack of bolts	$2.00	1	$2.00
Washers	50 pack of washers	$2.00	1	$2.00
L-brackets	Single L Bracket	$1.00	2	$2.00
Wooden Board	Single wooden board for mounting radar	$20.00	1	$20.00
			Subtotal:	$32.00

The next category is RF hardware. To connect the electronics of the radar, certain connectors will be required. Their approximate costs are in Table 5.

Table V
Budget for the Radar Connectors

Part Name	Description	Unit Cost	Quantity	Subtotal
SMA M-M Barrels	Single male-to-male SMA connector	$6.00	4	$24.00
SMA F bulkhead	Female SMA bulkhead connector	$5.00	2	$10.00
SMA M-M Cables	SMA male-to-male cables	$10.00	3	$30.00
			Subtotal:	$64.00

The next category of parts is the radar module electronics and antennas. This includes everything needed to generate, transmit, receive and process the analog radar signal. Costs for common, minor components such as capacitors and single-valued resistors have been excluded from the list of electronic parts in Table 6.

Table VI
Budget for radar electronics and materials

Part Name	Description	Unit Cost	Quantity	Subtotal
Oscillator	Generates transmit signal	$50.00	1	$50.00
Attenuator	Reduces signal strength	$10.00	1	$10.00
Amplifier	Increases signal strength	$90.00	2	$180.00
Splitter	Divides an RF signal into multiple paths	$40.00	1	$40.00
Mixer	Changes an RF signal's frequency	$50.00	1	$50.00
Wooden Board	Single wooden board for mounting radar	$20.00	1	$20.00
Can	Can for antenna	$5.00	2	$10.00
Modulator	Generates waveform	$5.00	1	$5.00
Op-amp	Quad operational amplifier	$15.00	1	$15.00

The next category, shown in Table 7, includes everything related to the system power supply. This includes the batteries and the circuitry used to regulate the voltages. For the cost of batteries, rather than round up to the high end of potential cost, such as if we were to go directly to a manufacturer specializing in batteries, the average cost of typical, widely-available alkaline batteries will be used. Additionally, we will account for needing to run tests on the system runtime by allocating budget for two sets of batteries. Equivalently, a set of good-quality rechargeable batteries could be purchased.

Table VII
Budget for Power Supply

Part Name	Description	Unit Cost	Quantity	Subtotal
9V battery	9V alkaline battery	$5.00	8	$40.00
9V battery connectors	Wires with caps for connecting to 9V batteries	$0.50	4	$2.00
Buck Converter	Brings voltage down to regulated level	$2.00	3	$6.00
			Subtotal:	$48.00

This concludes the cost for the radar module components, connectors and frame. In addition to the radar module, our system has a real-time, USB interface that uses an analog to digital converter and a microcontroller. The estimated costs for these are $6.00 and $50.00, respectively.

There are two more categories that have yet to be discussed. The first is the final category of physical materials, and that is the PCB construction category. Costs associated with PCB construction are much more variable than the costs outlined previously. It is highly dependent on what materials we used, how we ultimately choose to assemble the PCBs, whether we will need to reorder or redo certain boards, which supplier we decide to use, among other factors. Though many of these decisions have been made, things may change as the team works through the system integration process. For the sake of creating a somewhat accurate idea of how much the PCBs for our project will cost will use the approximate costs given by a major supplier of PCBs, and estimate that at the worst, we will have to remake every PCB in our design at least once (so twice as many PCBs will be accounted for than necessary per or design). In total this is approximately ten dollars per board, times three boards, resulting in thirty dollars allocated for PCB construction.

The final category of costs includes things such as shipping, taxes, and other fees that add on to baseline costs. This will create a slight increase in the costs previously stated and may have an even greater impact if more expensive options are chosen (such as rush shipping). To account for these costs, the subtotal of $580.00 up to this point will

be rounded up to an even $650.00. This can also serve to account for more expensive components being chosen than originally planned for. So in conclusion, the team must remain under $800 in total for the project according to the requirement specifications, and the anticipated cost is only $650. This fact, in conjunction with the fact that many components have already been given to the team, gives us a positive outlook with regard to the project finances.

9.0 Conclusion

Our team has made significant progress since the beginning of Senior Design. We began with the general concept of a radar application. Our desire to demonstrate our knowledge of multiple electrical engineering sub-disciplines and create a quality consumer product led us to solidify our project as a frequency-modulated continuous-wave radar module with multiple applications and a customizable software interface. This project uses RF engineering, integrated circuits, electronics, digital signal processing and coding, which covers many different areas of electrical engineering. This allows each member of the team to demonstrate their understanding and capabilities in their area of specialization.

After obtaining a donated, partially-built radar module from our sponsor Dr. Xun Gong, our team outlined a series of goals to aim for. Our primary goal is to use our radar module to measure the velocity of vehicles driving on a street near the engineering building at UCF. One of our stretch goals is to expand the capabilities of the software and output range readings. Lastly, we have a stretch goal of using the velocity readings, the range readings, or both as the input to a control system with external hardware.

In addition to technical goals, our team has a set of administrative goals that will facilitate the technical design, test and integration processes. We will meet regularly, provide updates between members with regard to progress on the technical aspects of the project, document our design changes, and keep a good record of our project remonstration results.

To reach these goals, our team will need to finish designing the radar module provided to use by Dr. Xun Gong, design a USB2 interface between the module and the host device in our system, develop a user-friendly API, construct and test our prototype, make any necessary modifications, and finally demonstrate our project.

A set of requirement specifications has been outlined to guide our system design. These requirements are related to the size, power consumption, accuracy and versatility of the system. In order to meet the specifications, our design will need to be balanced, as improving the system in one area may worsen the system in another area. The project's house of quality shows the interdependency between various characteristics of the system.

Our project is based on MIT's laptop-based radar, and our team performed extensive research to understand the operation of the baseline system. Background information

related to radar technology, remote sensing, microwave propagation, antennas, transceiver architectures, and pulse-doppler processing are all relevant to our project. Additionally, research was performed to investigate how MIT's baseline system design could be improved, particularly by implementing real-time signal processing and using a USB interface between the host device and radar system. All of the research combined resulted in a set of components that would be used in our design, including a microcontroller, analog-to-digital converter, mixer, splitter, amplifier, oscillator and modulator. A complete system block diagram was then established which outlined how signals and information would move through our system. The diagram was easily divisible into three distinct subsystems: the RF module, the RF data processing, and the implementation.

The design of each subsystem is subject to various constraints. Some of these constraints are industry standards, such as the C++ programming language, the physical characteristics of alkaline batteries, and the dimensions of coaxial connectors. Other constraints are specific to our project, such as time constraints related to our course deadlines, health constraints related to RF transmission safety, and financial constraints created by the current economic environment. These constraints must be taken into consideration during the system design, from the initial design and through subsequent design iterations.

The hardware design process is broken up into the following subsystems and circuits: the power supply, the modulator circuit, the RF subsystem, the gain stage, the low pass filter, and the analog-to-digital converter circuit. Each system or circuit has its own functionality and specifications. The circuits being designed will be assembled on PCBs, while the majority of the RF subsystem consists of antennas and pre-assembled circuits within block housings.

The software design consists of receiving and processing the signal from the RF subsystem. Real-time processing is accomplished by receiving a set amount of data per period and performing computations on just that dataset. The Ripple radar standard is used to develop the software. The software graphical user interface enables easy control of the radar module and acquisition of the computed velocity and range values at any given instant.

The project prototype is already partially complete. The only remaining elements are the PCBs. Two versions of these PCBs will be constructed during the testing phase of the project. The first version will be manufactured using a milling machine. The PCBs from the machine will be used for quick, small-scale design verification. Once designs are verified, PCBs will be manufactured by JLCPCB. The majority of components such as resistors and capacitors will be assembled prior to our team's receipt of the board, and our team will complete the assembly phase by soldering the remaining components.

Each element of the system will need to be tested, with the tests corresponding to the parameters that need to be verified. Several pieces of lab equipment will be used to verify that our subsystems and circuits are working as desired, including multimeters,

oscilloscopes, DC power supplies, and network analyzers. Software-driven test procedures will be used to generate complete analysis reports and provide concrete, quantitative test results. The software testing consists of verifying that the USB interface is operational and the computations in the software are accurate.

Once our individual subsystems have been verified, the integration process will begin. There may be difficulties in the integration process that require design changes. The integration phase often encompasses intermittent redesign phases until the system is capable of functioning as a whole unit. At this point, it is important to return to the requirement specifications and verify that the system has met them. If a requirement has not been met, the system may need further modification or the requirement itself may need to be updated.

The final step in the process is the demonstration. During our demonstrations, it will be important to record as many details as possible about the performance of the system. The requirement specifications can be used as the guidelines for measuring system performance. Ultimately the system will be rated based on how many specifications it is able to meet during demonstration.

Although the focus of this project is primarily on the technical aspects, our team has made an intentional effort to expand our focus beyond this. Engineering in the industry encompasses aspects of business as well, and our team's goal is to produce a quality, marketable consumer product. Our ultimate goal by the end of Senior Design is to have a product rather than just a device. It will be user-friendly, accurate, reliable, and versatile. Further development in the future can lead to an even better product.

Our team will continue to track our progress as we complete the design phase of our project this December and move into the testing phase starting in January. We will constantly evaluate the quality and performance of our system to ensure our end result is satisfactory.

11.0 Appendices

11.1 References

[1] “IEC 60086-1:2021,” *IEC Webstore,* June 17, 2022. [Online]. Available: *webstore.iec.ch/publication/60968*. [Accessed Nov. 3 2022].

[2] “ANSI C18.1M, Part 2-2019,” *ANSI Webstore*. [Online]. Available: *webstore.ansi.org/Standards/NEMA/ansic181mpart2019*. [Accessed Nov. 3 2022].

[3] “IEC 61169-15:2021,” *IEC Webstore*. [Online]. Available: *https://webstore.iec.ch/publication/62310*. [Accessed Nov. 3 2022].

[4] “ISO/IEC 14882:2020 Programming Languages - C++,” *ISO*. [Online]. Available: *www.iso.org/standard/79358.html*. [Accessed Nov. 3 2022].

[5] “ISO/IEC 9899:2018 Information Technology - Programming languages - C,” *ISO*. [Online]. Available: *www.iso.org/standard/79358.html*. [Accessed Nov. 3 2022].

[6] “Language Reference,” *Arduino*. [Online]. Available: *www.arduino.cc/reference/en/.* [Accessed Nov. 3 2022].

[7] “Ripple,” *Consumer Technology Association.* [Online]. Available: *www.cta.tech/Resources/Standards/Ripple.* [Accessed Nov. 3 2022].

[8] “IEC 62680-2-1,” *IEC.* [Online]. Available: *webstore.iec.ch/preview/info_iec62680-2-1%7Bed1.0%7Den.pdf.* [Accessed Nov. 3 2022].

[9] “Standard Commands for Programmable Instruments (SCPI),” *IVI Foundation*. [Online]. Available: *https://www.ivifoundation.org/docs/scpi-99.pdf*

[10] “VPP-4.3: The VISA Library,” *IVI Foundation*. [Online]. Available: *https://www.ivifoundation.org/downloads/Architecture%20Specifications/vpp43_2022-05-19.pdf*

[11] “FCC Online Table of Frequency Allocations,” *Federal Communications Commission,* July 1, 2022. [Online]. Available: *transition.fcc.gov/oet/spectrum/table/fcctable.pdf.* [Accessed Nov. 3 2022].

[12] “What are the ISM Bands, and What Are They Used For?,” *Military-Aerospace Electronics,* Feb. 19, 2019. [Online]. Available: *www.militaryaerospace.com/directory/blog/14059677/what-are-the-ism-bands-and-what-are-they-used-for.* [Accessed Nov. 3 2022].

[13] “RF Safety FAQ,” *Federal Communications Commission.* [Online]. Available: *www.fcc.gov/engineering-technology/electromagnetic-compatibility-division/radio-frequency-safety/faq/rf-safety#Q6.* [Accessed Nov. 3 2022].

[14] “Programming Chirp Parameters in TI Radar Devices,” *Texas Instruments*. [Online]. Available: https://www.ti.com/lit/an/swra553a/swra553a.pdf

[15] “Low-Complexity Joint Range and Doppler FMCW Radar Algorithm Based on Number of Targets,” *Kim, Bong-seok & Kim, Sangdong & Jin, Youngseok & Lee, Jonghun*. (2020). Sensors. 20. 10.3390/s20010051.

[16] “Transmission Line Model,” *Microwaves101.* [Online]. Available: *www.microwaves101.com/encyclopedias/transmission-line-model.* [Accessed Nov. 22 2022].

[17] “Build A Small Radar System Capable of Sensing Range, Doppler and Synthetic Aperture Radar Imaging,” *MITOpenCourseware.* [Online]. Available: *ocw.mit.edu/courses/res-ll-003-build-a-small-radar-system-capable-of-sensing-range-doppler-and-synthetic-aperture-radar-imaging-january-iap-2011/pages/projects/.* [Accessed Nov. 4 2022].

[18] Vortex. “Police Radar: How it Works & How to Beat It,” *Vortex Radar, Nov. 9, 2017.* [Online]. Available: *www.vortexradar.com/2017/11/police-radar/.* [Accessed Nov. 4 2022].

[19] Shane Cloude, *An Introduction to Electromagnetic Wave Propagation & Antennas.* Springer New York, NY. 1996.

[20] “Publication Number: 558074,” *Federal Communications Commission, April 2, 2019.* [Online]. Available: *https://apps.fcc.gov/oetcf/kdb/forms/FTSSearchResultPage.cfm?switch=P&id=21124* . [Accessed Dec. 6 2022].