

UCF Senior Design II

Smart Compact Scope for Hunting Rifles

Low Power Variable Optic with Automatic Distance Correction



Department of Electrical Engineering and Computer Science

University of Central Florida

Dr. Lei Wei

Group 13

Ahmed Al Busaidi CPE

Collan Dimitri PSE

Jacob Clevenger EE

Jose Vasquez Dickson EE

Contents

1	Executive Summary	1
2	Project Description	3
2.1	Motivation	4
2.2	Goals and Objectives	5
2.2.1	Lightweight and Slim	5
2.2.2	Inexpensive	5
2.2.3	Easy to Use	6
2.2.4	Accurate	7
2.3	Requirements and Specifications	8
2.4	House of Quality	9
2.5	Block Diagram	11
3	Research & Part Selection	12
3.1	Research	12
3.1.1	Similar Products	12
3.1.2	Lenses	14
3.1.3	Laser Rangefinder	16
3.1.4	Time-to-digital Converters	17
3.1.5	OLED Displays	18
3.1.6	Microcontrollers	18
3.1.7	Communication Interfaces	20
3.1.8	Batteries	22
3.1.9	Optical Calculations Research	26
3.1.10	Wireless Protocols	27
3.2	Part Selection	29
3.2.1	Laser Rangefinders	29
3.2.2	OLED Displays	33
4	Design Constraints & Standards	35
4.1	Constraints	35
4.1.1	Cost Constraints	35
4.1.2	Health, Environmental, and Safety	35
4.1.3	Time Constraints	36
4.1.4	COVID-19 Constraints	36
4.1.5	Programming Constraints	37
4.2	Standards	37

	4 .2.1	ANSI Z136.1	37
	4 .2.2	Python Programming Standards	38
	4 .2.3	Power Supply Standards	38
	4 .2.4	Unit Standards	40
	4 .2.5	Soldering Standards	41
5	Design		44
	5 .1	Design Considerations	44
	5 .1.1	Design Consideration I	44
	5 .1.2	Design Consideration II	45
	5 .1.3	Design Consideration III	45
	5 .1.4	Design Consideration IV	47
	5 .1.5	Design Consideration V	48
	5 .2	Optical Scope	49
	5 .2.1	Objective Lens	49
	5 .2.2	Main Eyepiece	50
	5 .2.3	Main Construction	51
	5 .3	Optical Housing	52
	5 .4	Targeting Lens	55
	5 .4.1	OLED Display	57
	5 .4.2	Display Board Circuit	57
	5 .4.3	Interfacing with the OLED Display	58
	5 .5	Laser Rangefinder	61
	5 .5.1	Calculating time-of-flight	62
	5 .5.2	Communicating with Microcontroller	62
	5 .5.3	Implementing the rangefinder	62
	5 .6	Set & Reset Switch	64
	5 .7	Power Supply	66
	5 .7.1	5V Voltage Regulator	66
	5 .7.2	12V Boost Converter	70
	5 .7.3	One-line Diagram Distribution	73
	5 .7.4	Interface Board Design	75
	5 .7.5	Power Supply Housing	80
	5 .8	Software	81
	5 .8.1	The Python Programming Language	81
	5 .8.2	Python Development	82
	5 .8.3	Tensor Flow Module	83
	5 .8.4	Bluetooth Connection	84
	5 .8.5	Data Visualization	85
	5 .8.6	Flow Chart	87
	5 .8.7	Examples of JSON	89
	5 .8.8	Function Breakdown	89
	5 .8.9	Main Functions Breakdown	91
	5 .8.10	Portenta Vision Shield	93
	5 .8.11	OpenMV Cameras	94
	5 .8.12	Android Studio	95

	5.8.13	nRFConnect	99
	5.8.14	GitHub	100
	5.8.15	Software Revision	100
	5.9	Design Summary	101
6		Overall Integration	104
	6.1	PCB Design	104
	6.1.1	Through Hole Components	105
	6.1.2	Surface Mount Components	106
	6.1.3	Layout	107
	6.1.4	Traces	108
	6.1.5	PCB Vendors	108
	6.2	Testing	110
	6.2.1	Overview	110
	6.2.2	Software	112
	6.2.3	Laser Rangefinder	114
	6.2.4	Targeting Lens	116
	6.2.5	Power Supply	116
7		Administration	118
	7.1	Division of Labor	118
	7.1.1	Scope Design	118
	7.1.2	Rangefinder	118
	7.1.3	Power Supply	118
	7.1.4	Targeting Lens	118
	7.1.5	Software	119
	7.2	Communication Tools	119
	7.2.1	Discord	119
	7.2.2	Zoom	119
	7.2.3	Overleaf	119
	7.2.4	OneDrive	119
	7.3	Budgeting and Financing	120
	7.3.1	Laser Rangefinder Cost	120
	7.3.2	Targeting Lens Cost	121
	7.3.3	Scope Cost	121
	7.3.4	Microcontroller Cost	121
	7.3.5	Power Supply	121
	7.3.6	Hantek 2D42 3 in 1	122
	7.4	Project Challenges	122
	7.4.1	Laser Rangefinder	122
	7.4.2	Power Supply	122
	7.5	Scheduling	123
8		Conclusion	125
B		Software	129

List of Tables

1	Specifications	8
2	Similar Product Comparisons	14
3	Microcontroller Comparison	19
4	Microcontroller Comparison Revised	19
5	Battery Comparison	23
6	State of Charge	25
7	Voltage Regulator Comparison [15]	26
8	Wifi vs. Bluetooth	28
9	Laser Rangefinder Comparisons	30
10	OLED Display Comparison	33
11	Circuit Classifications	39
12	Optical Lenses Comparison	52
13	Subset of SSD1309 Controller Commands	60
14	Laser Rangefinder Commands	64
15	LM7805 Specifications [14]	67
16	Absolute Maximum and Minimum of Vin [4]	69
17	Components Powered by Stepped Down 5 V Battery in Prototype	69
18	TPS55340RTER Specifications [10]	71
19	Components Powered by Stepped Up 12 V Battery in Prototype	72
20	Python vs. C	81
21	Rangefinder Error Codes	115
22	Parts List	120
23	Schedule	124

List of Figures

1	House of Quality	10
2	Block Diagram	11
3	Time-to-digital converter schematic [17]	17
4	DLEM 30 Laser Rangefinder	30
5	Connections to Pierced Terminals	42
6	Connections to Heel Filler	42
7	Interface board schematic between display and microcontroller	58
8	I2C Bus	59
9	Rangefinder Flow Diagram	61
10	Laser Rangefinder	63
11	Set/Reset Switch mounted on Weapon	65
12	5V Voltage Regulator	67
13	9.6V to 12V Boost Converter	71
14	Components Powered by Power Supply	73
15	Power Supply One-Line	75
16	Power and Display Interface Board	76
17	Board 3D Rendering	77
18	Overall Schematic	78
19	3D Model	79
20	USB-to-I2C Integrated Circuit	80
21	Pandas DataFrame	86
22	Matplotlib Graph Example	87
23	Software Flowchart	88
24	JSON Bullet Format	89
25	JSON Error Format	89
26	OpenMV Cameras	93
27	Block Diagram [12]	94
28	Application Stats Format	96
29	Application Camera Format	97
30	Application Setting Format	98
31	Full Schematic	102
32	PCB Design	104
33	Tensor Flow Model	114

1 Executive Summary

There exist many different optical elements for the firearms industry on the market, and most of them offer the same options with very little change. The price range for all of these can range from less than a hundred to multiple thousands, varying by company. The main difference with most of these companies, aside from quality and construction, is that they promote their own reticle that is seldom shared between brands. The main utilization for a reticle, however, requires the external information supplied by a rangefinder, a spotter who has one, or thorough knowledge of the terrain.

The main goal of this product is to provide the information and automatically adjust for it. In order to do that, there needs to be a rangefinder built on-board, which consists of just a laser emitter and detector. There needs to be a respectable scope system as well, as having function without clarity would defeat its own purpose. The way to couple these together are to have a reticle that can display in some form the compensation for the changing distance of the target, which will be accomplished by having an OLED screen as the reticle. The on-board electronics to make all this possible must also be compact enough that this scope is not bulky or overly heavy so that a hunter or a soldier is not fatigued by using this product.

Products similar to this one exist on the market to solve this problem, but the only option that does not have a separate rangefinder is both bulky and extremely expensive, where our product plans to keep the cost down, weight manageable, and user friendly by only displaying the information needed.

What our project aims to design is a product that incorporates digital assistance into an analog optical system. There are many different types of scopes on the market for the sporting shooter all the way to the military to utilize. Many of these products are simply passive optical elements that allow for magnified viewing of the shooter's target. There is generally a compact optic system that will often have the option for a variable magnification, but the most advanced option that is commonly available is for the illumination of the reticle by an LED light source. Not to belittle the efforts that went into the design of these scopes, as they required extensive optical knowledge as well as mechanical knowledge for the adjustable turrets and magnification of the scope. However, with our design, we want the scope to do most of the work for the user. There are generally notches in the reticle that correspond to angles, such as how wind might affect the bullet or how gravity will cause a drop over a specific distance, but the work of estimating or calculating that falls onto the user. What we want to incorporate to assist in those calculations is a range-finding laser and sensor system that will give an accurate read of exactly how far away the target is. With this information, a calculation for how gravity will affect the bullet at varying distances can be made to ensure that the user corrects their aim to accommodate. To assist in these calculations, there will be an onboard Arduino that will read information from the rangefinder and quickly and precisely calculate exactly how far the bullet will drop by the distance of the target. To display the

information, we intend to use a transparent OLED screen in place of the reticle, so that we can still see the object but are able to essentially overlay information onto the image. With the information supplied from the Arduino, we will be able to essentially point the user as to where they will need to aim to compensate for bullet drop that will happen over a specific distance. Currently, this project will only be able to account for vertical compensation, as that is calculable with the equipment we will have available, as horizontal adjustments will require knowledge of extra atmospheric readings, such as humidity or wind. Although there are lasing instruments that can detect these conditions, their utilization is out of the scope of this project, and their integration would likely not be financially feasible for us at this time.

With all the hardware in place to make this project something achievable, there also needs to be adequate work on the software side as well. One of the main components that we wish to implement alongside scope is application integration. The main issue with calculating bullet drop is that, not only is it specific to each individual bullet size, but even different grain weights for the same bullet would also require new calculations. We have the option of optimizing a scope for a specific bullet type and grain weight, such as if the military were to use a model that would not require multiple settings, but commercially, consumers would rather have multiple options for a scope, as well as the competing products all have options for multiple calibers. To allow for customization of the scope to the user, we would integrate an application that allows the user to choose their bullet type and grain weight, as well as what type of firearm they are using. The main goal with this product is to push an analog industry further into the digital world.

2 Project Description

The basis of our project that we aim to achieve is an optical system that can achieve a variable magnification from one to six times, housed in an aluminum tube with a diameter of 30 millimeters that provides rotating adjustments for the magnifying lens, as well as an extra adjustment for the focal lens to provide eye relief adjustments. To stay consistent in the industry, as well as allow for a more compact system, the objective lens will have a diameter of 24 millimeters. The shock absorption for the housing should be able to disperse 8.5 Joules (the average amount of recoil for a 5.56NATO round) without shifting the interior contents, so that the alignment of the lenses will not be affected by use of the scope. The housing should also have two adjustments for the windage and elevation of the reticle of the scope, the x-axis and y-axis respectively.

The reticle itself will have an embedded OLED display that will be programmable so that individual lights can be lit depending on information sent from the Arduino board. The information that the Arduino board receives will be sent from a rangefinder system that consists of a laser emitter and sensor. The emitter will produce a laser between 1300 and 1500 nanometers and have a range of 1000 meters while still being able to reflect and receive the signal into the sensor with a margin of error of at most 1 percent. The reason for the specific wavelength is so that not only is it out of range of visible light, but it would also be undetectable by infrared scanning equipment such as night vision, to add a level of stealth to the project. The goal for the processing speed of the unit is that from the laser being emitted to the processor adjusting the target display on the OLED display, there is an elapsed time of less than 5 seconds. For the power consumption, the goal is to have the battery life last for at least 50 hours continuously.

If possible, with the time and resources allotted include having upgraded shock absorption to disperse 20 Joules so that the scope can operate with more powerful calibers while still retaining the rigidity in the system. The effective range of the rangefinder would be increase from 1000 meters to 2000 meters with the same margin of error of at most 1 percent. The main upgrade that we would like to achieve if we have the time is to include integration with a phone application that allows for extra parameters to be input by the user about the specifications of the weapon being used, as things such as the barrel length, bullet weight, and type of weapon would all have noticeable differences on the trajectory of the bullet.

As is, the project would be calibrated to the specification most often used in the military, but application integration would allow for the optic to be more commercially marketable, and more versatile to include the use of different calibers. To make the acquisition of the target even faster, it would also be advantageous to decrease the processing time to within 2 seconds. To make the optic have more longevity as well, we would strive to increase the battery life to at least 1,000 hours.

The following couple subsections will cover our motivation, goals, and objectives for the overall design of this project. Then we will cover the products that are similar

to ours in the related works section. Following that we will go over the engineering specifications for this project as well as a block diagram of the project. Finally we will go over how we make decisions on what features and aspects of the project we want to prioritize. This will be done by building a house of quality which provides a visual representation of the correlation between different features.

2.1 Motivation

The main motivation behind this project is from a familiarity in the commercial optic industry and a desire to innovate in the market. Many products that exist currently operate on a basic principle of aiming a laser at a coated lens for red dot optics, and a slightly more complex optic system for magnified optics. Electro-optics utilize both concepts; however, they fall into the same category of simplicity. Electronics have progressed in many fields in recent years that aim to assist consumers, such as Tesla and their self-driving features, while most optics companies improve on either clarity or durability in newer products.

Innovation would be beneficial in military applications, as an assisted optic would allow for soldiers to perform at a higher capability than they are trained. In the hunting and sporting applications, the assistance allows for easier acquisition at longer range and fewer missed shots. There are options on the market that aim to achieve these goals, but they either operate on less of an automated level or are too expensive to promote ease of accessibility.

The purpose for designing this product is to introduce something that has been attempted, but whose previous iterations have been too exclusive or inefficient. The zoom lens system that we are integrating gives the user more options and has only been used on one similar product. These products will be given more in depth comparisons in a later section. The Sig Sauer BDX line is the only similar model that integrates the zoom system; however, the range-finding aspect of their system exists as a separate unit. The Trijicon CCAS provides the user with an unparalleled level of usability, however the price tag causes exclusivity that few would be able to attain. The reason that our design would benefit the market is because it gives a compact, self-contained package that allows the user to identify the range that their target is at and have an automatic correction system that informs the shooter of how to compensate vertically for the distance that they are shooting at. Even on a military basis where the budget is not something that has to be factored in nearly as heavily, it is still beneficial to have a product with a lower price point if it does not waver in quality. There exists rangefinders in the market that have a much longer range and are even more compact than the model that we are using, but in very few cases would someone require a distance greater than 1000 meters. The main utilization from this device will be from the 400 meters and onward range, however there are consumers who will want to test their limits, which is why we are aiming for the specific distance that we are. For the magnification limits, although it will not give the closest view at such great lengths, it will still allow identification of a

target and provide a challenge for a sport shooter. The stretch goal durability that we intend to achieve allows for a wider range of usability as well, such as if a user wants to utilize a longer-range cartridge without compromising the integrity of the unit or causing it to lose its reticle position, and therefore, become inaccurate.

2.2 Goals and Objectives

Below are the goals and objectives of this project. We want to build something that will be viable in the market and compete against other similar products. To be competitive you have to have features that people want while at the same time being inexpensive. This means we must find a compromise between features and cost while still competing with existing solutions on the market.

2.2.1 Lightweight and Slim

Our first goal is to keep the project as lightweight as possible. If the scope were to weigh too it would be less likely that people would buy it as it would be too inconvenient. Since rifles are already heavy tools, adding more weight wouldn't help. The main concern when it comes to weight added will be the battery as well as the laser rangefinder.

To help keep the design lightweight we will build our project out of lightweight materials. This includes using aluminum in some parts of the design as opposed to using a heavier material. Microcontrollers can also be bulky which may not add a lot of weight, but will defeat the "slim" goal of the project. The laser range finder will likely add the most bulk to the project as the very small ones are often times outside the budget of the project.

The targeting lens itself is unlikely to add much in terms of weight or bulk, but we will need to build a board that can interface with the targeting lens. The same board will also contain the power supply, which will be used to power the microcontroller, laser rangefinder, and targeting lens. Overall this board is also unlikely to add much in terms of weight and bulk. The primary concern will be the battery. We will need to find a battery that is small enough to not add much weight or bulk, but is also powerful enough to power each component.

With all of this in mind, this will give us a good idea of what type of components we should look for. In some instances it may not be possible to find a component that would be ideal, but these are compromises we will have to make due to our limited budget as students. Even so, keeping the design lightweight and slim is a high priority for this project.

2.2.2 Inexpensive

An inexpensive product is a big objective in this project. Many similar products on the market can often times end up costing several thousand dollars. Because of

this, these kinds of products are often inaccessible to a large amount of people. This is often why many of these types of designs are often only seen in the military or on the very high-end of the consumer market. The goal of the project is the keep the scope cheap enough that your average person in the target audience would be willing to invest in one. This means we would like to keep the total cost under \$1k.

The most expensive part of this project will likely be the scope and the laser rangefinder. The lens and housing of the scope will likely end up costing a few hundred dollars. The laser rangefinder is also expensive for the ranges we are looking to measure. These seem to run anywhere from \$200 all the way up to \$500. This is a cost that could likely be reduced if we were able to design and manufacture our own. But designing and building one is outside the scope of this project and would be far too much for a team of only four members to handle.

Microcontroller vary in cost anywhere from \$20 all the way up to a few hundred dollars. While buying the cheapest microcontroller would be very helpful in keeping the cost of the project down, many of the cheaper microcontrollers do not support all the features we need like bluetooth and several communication interfaces all while being small. This means we will end up reaching into the \$100 range for our microcontroller. It is possible that some optimizations could be made in the future to bring this cost down.

Finally the targeting lens and the power supply will end up only contributing around \$75 towards to total cost of the project. The targeting lens while seemingly complicated, is a surprisingly inexpensive part at around only \$30. The power supply and battery may end up costing around \$45 but overall this will not contribute much to the cost.

Considering all the previous factors it is feasible that we will be able to keep this project relatively inexpensive. The cost may seem high, but when compared to similar solutions on the market it ends up being very competitive. This is good for but the consumer as well as us since our budget is limited to what we can afford ourselves.

2 .2.3 Easy to Use

Being easy to use is an essential requirement for this project. Many products, especially military grade ones can be quite complicated to use. Of course this may often be because they support more features, but even at their base level they can end up being not very use friendly. This can introduce a barrier to entry for products like this as well as inconvenience people who are capable of using these types of products.

Because of this, we would like to create a product that is easy for your average consumer to use. This means the basic usage should be simple and easy to understand. We want to minimize that amount of input a user needs to give to the

system. The more inputs a system requires from the user, the more likely the user is to get confused and the less likely they are to use your product. To help prevent this we have limited the amount of inputs our project can receive from the user.

The first mode of input is used while the scope itself is being used. This is a single momentary push button. To operate the scope, the user can simply push a button and will instantly receive feedback on the targeting lens, either in the form of some sort of indicator letting the user know the scope is taking measurements or the result of the measurements being display on the lens. Each time the button is pressed, the scope will be able to retake the measurement and update the targeting lens accordingly. This simple button press is all that is need from the user to interact with the scope while in use.

The second form of input is via an app. This app will be able to connect to the scope via the Bluetooth module provided by the microcontroller. With this the user will be able to easily adjust different settings for the scope. This can allow the user to easily adjust to scope to their preferences as well as the type of rifle they are using. We will provide several presets that the user can choose from as well as an option to enter a custom projectile weight and size.

With the exception of a simple power switch and scope controls, these are the only inputs that our project should accept. This will allows us to keep the input from the user limited, benefiting both what we must deal with as designers as well as what the user must know to use the scope. Keeping it simple will help give us an advantage against our competitors as well as keeping the user of the scope happy.

2.2.4 Accurate

Being accurate is essential to the design. While it doesn't have to be perfect, it must be accurate enough that it increases the users accuracy to a greater amount than if they weren't using it. There is a very positive relationship between the cost of the scope and the accuracy. This means there must be some compromise between cost and accuracy. As it currently stands, the military grade scopes would be the highest in terms of accuracy. Of course, these are very expensive and outside the price range of your average consumer.

In our application, being accurate up the to centimeter would be nice to have, but would also be overkill. We have to analyse the cost of making it more accurate vs the value it will add to the scope making it more accurate. It can very quickly turn into a situation of diminishing returns, which means we should set a realistic goal of being accurate up to a few meters. The main application of this scope will be sports and hunting, so we also do not have the pressure of being pinpoint accurate like a military grade scope would.

The laser rangefinder will likely end up having the largest effect on the accuracy of

the scope. The longer the range the rangefinder can measure to larger the error. Ideally we want to have an error of ± 2 meters. Any more accurate would cause the cost of the rangefinder to increase significantly without much gain in terms of value to the user.

While the rangefinder will have to most significant affect on the accuracy, the targeting lens will have a significant affect on the accuracy. The accuracy of the lens is determined by the amount of detail we can get on the vertical axis. Since we are using an LED based solution, we will want to find a way to get a high density array of LEDs along the vertical axis. Depending on the method used, this could add to the cost, but not in a significant way like the rangefinder.

Overall, the goal is to have an accuracy of about 2 minutes of angle. Many aspects of this project may end up affecting accuracy, like the optical scope, the rangefinder, the targeting lens, as well as how sturdy the entire structure is. These are all concerns we have to keep in mind while designing and building the scope. At the end, we would like to have a scope that is accurate enough to compete with other solutions on the market all while beating them in terms of cost.

2.3 Requirements and Specifications

The following table contains a list of requirements for the project. These are goals that should be met when the project has been completed.

Requirement	Units
Variable Magnification Optical Set-Up	1x-6x Magnification. Will require 7-9 lenses
Bluetooth Module	Up to 100m
Microcontroller	At least 480MHz Processor Speed
Laser Rangefinder	500m to 2km
LCD Screen	20 x 4, I2C, 12V Power Supply
Tactile Button Pack	5V - 12V Supply
Connector Wiring	5V - 12V
Battery	5V - 12V

Table 1: Specifications

2.4 House of Quality

The House of Quality allows us to explore the relationships between different attributes in the projects. Based on the House of Quality, you can determine which attributes in your project that you want to prioritize. The up arrow indicates a positive correlation and the down error indicates a negative correlation. In the last row of the House of Quality matrix there are testing requirements which are set by both the customer and the engineers.

The cost of the project is a 1 to 1 relationship between the customer and the engineer. The more is costs to produce the more we would have to charge for the product. Ideally we want to keep the cost of the project as low as we can since we are trying to make a more affordable version of this type of scope.

The durability of the product can have the affect of causing it to cost more. Since the scope is mounted on top of a rifle, it will often be introduced into hazards environments. It will also be exposed to recoil when the gun fires. This means to be more durable we will have to invest in higher quality parts as well as measures to mitigate the effects of the recoil.

The power output and the battery life have a negative correlation. The more power the scope uses the shorter the battery life of the scope. The three major components using power in this product are the microcontroller, the laser rangefinder, and finally the targeting lens. The microcontroller will be always running meaning we will want to find a microcontroller that doesn't use a lot of power. The rangefinder will only be powered during the time of measuring the distance to the target. The targeting lens will be powered only during the period that the target has been set and the rifle is fired.

Accuracy has a high correlation with the cost of the product. The accuracy of the scope is mainly affected by two components. The first is the rangefinder. The more expensive the rangefinder the more accurate it is. The targeting lens also affects the accuracy. A high-density LED strip/or a high-density OLED display will allow us to be more accurate, but a higher density is also more expensive. There is also a positive correlation with ease-of-use and accuracy as the more accurate the scope is the easier it will be for the end user to trust it when aiming and won't have to compensate for inaccuracy.

Positive Relationship		↑	
Negative Relationship		↓	

House of Quality		↑				↓
		Cost	Durability	Battery Life	Ease of Use	Accuracy
		-	+	+	+	+
Cost	-	↑↑	↑↑			↑↑
Dimensions	-				↓	
Power Output	-	↓		↓		
Processing Speed	+	↑			↑	↑
Accuracy	+	↑↑			↑	↑↑
Tests for Engineering Requirements		< \$2000	8.5 Joules	50 Hours		2 Minutes of Angle

Figure 1: House of Quality

With this table we are able to get a pretty good idea of what we want to prioritize. Because our main constraint is money, we are going to aim to minimize the cost while still finding a good compromise on durability, battery life, ease-of-use, and accuracy. This is also beneficial to the customer as the product will be cheaper for them the cheaper it is for us making it more accessible.

2.5 Block Diagram

Below is block diagram that includes each component of the project design. It indicates who will be working on which parts. Components that have a gradient like the receiver and emitter are parts that will be worked on by multiple people. Components that are black are parts that will be work on by everyone. The block diagram also includes arrows that indicate power/data flow between components.

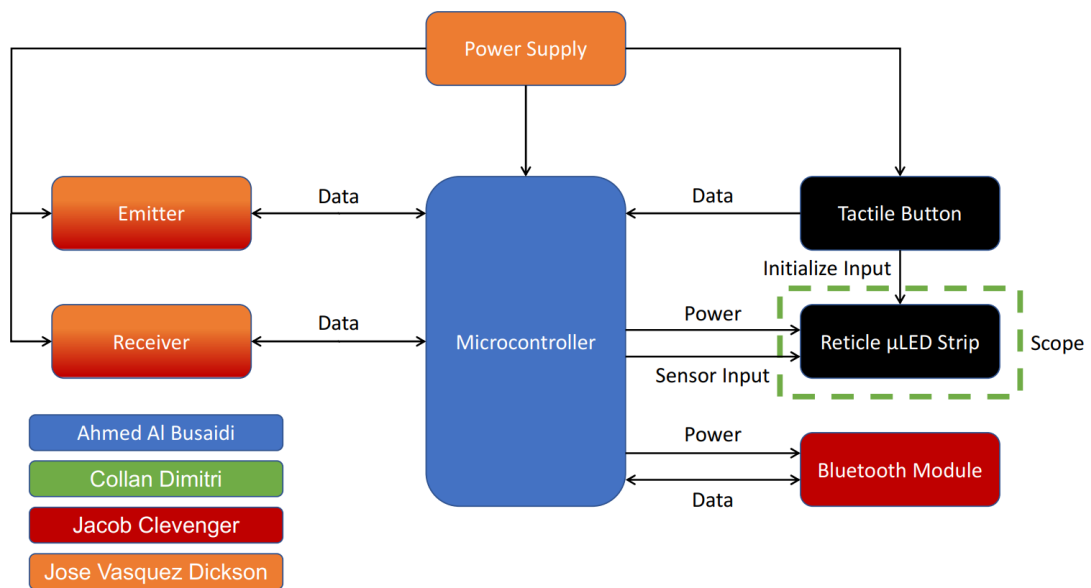


Figure 2: Block Diagram

Just above we have provided a block diagram to give an high-level overview of the entire technical portion of the project. This includes the emitter and receiver used with the rangefinder; the power supply the will be used to power all the components; the inputs used to give information to the microcontroller; the microLED strip to project the distance upon the lens of the scope, and finally the microcontroller which will coordinate all the inputs and outputs of the entire project.

3 Research & Part Selection

In this section we will go over the research that we did into the various parts of this project as well as the process of choosing parts. In the first half we will go over our research into what type of technology we might need to develop our project. The second half will cover the process of choosing the appropriate parts needed and why we chose them.

3.1 Research

We did quite a lot of research for this project. There were many different systems we need to research, ranging from optics, rangefinders, methods of creating a lens that can display information, the type of battery we want to power these systems, what type of microcontroller to use, and more. In this section we will cover each topic we looked into and which technology we decided to use.

3.1.1 Similar Products

Some of the more advance optics that exist in the market that are comparable include the Meprolight Foresight™, which is an entry level red dot optic that aimed to integrate display of extra information on the reticle. Alongside the reticle, which is interchangeable by the user through the use of an app, a leveler and a compass are projected onto the lens. Although this shares few similarities with our design, it helped inspire change from the original design of a non-magnified system that still integrates the rangefinder.

The Foresight™ also markets automatic zeroing through the app, however we confused that with the distance correction that we are trying to achieve. The company was actually referring to the ability to adjust the zero of the reticle through the included phone application, instead of using tools to adjust the laser on the unit itself. Our design was changed from what we thought were similarities to this product through the addition of the magnified system, however we do not plan to have any display that indicates distance or direction, our system will be more simplistic by only displaying necessary information: the vertically displaced reticle. The Foresight™ has an MSRP of \$699.99, which is a standard price of a mid-level optic in the industry.

The closest product on the market to our design is the line of scopes from Sig Sauer, the Ballistic Data Xchange (BDX) line. This product is usually sold alongside a rangefinder that has a built in Bluetooth transmitter that connects to the scope itself to communicate the distance that the user is trying to cover. There is a reticle inside the scope that uses the LED system we are trying to integrate that illuminates the appropriate LED to account for bullet drop. The MSRP of this line of scopes ranges from \$549.99 to \$1099.99, depending on the type of optical system, as well as if it is the combined package of the scope and rangefinder.

There exists many options in this line of scopes that offer many different magnification options, and there is an onboard computer that allows for the use of Bluetooth for the rangefinder, as well as a phone application that allows the user to build a profile for the scope depending on the type of firearm being used. Although gravity reacts the same to all projectiles, the velocity of a bullet will impact the distance traveled corresponding to a specific bullet drop. The profile on the application allows the user to preset the caliber, the barrel length of the firearm, and the type of firearm (semi-automatic or bolt action) to help factor in what the bullet will do at the specified distance. This is very similar to our project in the aspect that we also hope to have application integration to allow the user to adjust their settings, but we also want to start with a scope that is programmable so that the system is ready to go with no further adjustments. The mindset behind this choice is that in a military setting, a soldier will not be able to adjust options on a smartphone while in the middle of a firefight. However, this product is also made with commercial marketability in mind and is still aimed at hunters and sport shooters. The goal is to make a prototype that is aimed towards military applications, where a fully evolved version is designed for a shooter that is not in a life-or-death situation.

The product on the market that is the fully actualized version of what we are aiming for, with a plethora of other features, is the Trijicon Continuously Computed Aiming Solution (CCAS). This product utilizes an on-board rangefinder to gather information for the scope to correct its reticle for the distance, but it also can scan for the speed of the target, as well as the angle of the shot while factoring in atmospheric conditions to provide the best information for where the reticle must reside for a shot to hit its target, moving or stationary. The rangefinder also emits a laser at 1550 nanometers so that it falls out of the range of infrared light, meaning that the user will not be detected by standard night vision equipment. The extra information (the distance of the shot, the trajectory of the target, the type of ammunition being used, etc.) is projected onto the screen for the user using an LED.

The magnification of the scope is a fixed 6 times, where our goal is a system with an adjustable magnification of 1 to 6 times. This product is not integrated with an application; it is programmable to specific settings and calibers. This is intentional as the company designs most of their products with solely military in mind, factoring in that many decisions have a short response time and disarming oneself to change settings would not be preferred. The consumer side can still benefit from these products if they are comfortable with the steeper price tag. One of the improvements that we aim to achieve over this product, however, is cost effectiveness. There will be shared features, but we will attempt to keep our project under 10% of the cost of this unit, which has an MSRP of \$25,000. A product like this has a wide range of utility and versatility but a severe lack of cost effectiveness, so its utilization will be narrowed to only select situations, and in short supply.

	Magnification	Automation	Battery	Rangefinder
Mepro-light Foresight	Fixed 1x Magnification	App integration automatically changes reticle design, reticle position	50 hours minimum, rechargeable battery	No affiliated rangefinder
Sig Sauer BDX Line	Multiple options, Lowest 3.5x, Highest 30x	LED strip along reticle, automatic movement based on Bluetooth communication with external rangefinder	Battery life not stated, (2) 2032 Batteries	Rangefinder as separate unit, communication via Bluetooth
Trijicon CCAS	Fixed 6x Magnification	Automatic movement of reticle based off communication with onboard rangefinder, both stationary and moving	8 Continuous Hours, (4) CR123 Batteries	Onboard 1550nm Rangefinder, 2500m range
Group 13 Design	Variable 1-6x Magnification	Automatic movement of reticle based off communication with onboard rangefinder, stationary	50 Continuous Hours, battery operated	Onboard 1064nm Rangefinder, 1000m range

Table 2: Similar Product Comparisons

3.1.2 Lenses

One main aspect that had to be researched was for the optic system as a whole, because this was the first instance of our team not only designing their own optical system to be utilized for a specific purpose, but it also consisted of a zoom design for variable magnification. The basis of the system was founded on the zoom lens however, and that it only required three lenses minimum to achieve variable magnification. From there on, we found that there were multiple versions of the three lens systems that would work, consisting of a combination of positive and nega-

tive lenses, otherwise known as convex and concave, respectively. The difference in the systems is due to the extra elements they require or the aberration that is caused by them. Because of the issue with using three positive lenses having the highest degree of aberration, as well as it being uncorrectable, we have decided to go with the positive-negative-positive system. For naming convention, this three lens section is referred to as the eyepiece. This system, however, requires an extra lens to serve for reversal of the image. This is familiar from experiences in Geometric Optics, however it does also pose an extra challenge. The image reversal assembly requires that the image exist at twice the focal length of the lens itself, which requires much more space in the optical system. This was compensated for by choosing a lens with a smaller focal length, as the focal distance must be doubled in either direction. Thankfully, the focal point of the image reversal lens must only coincide with the focal distance of the objective lens, not the focal length for the eyepiece. This helped to shave about 60 millimeters off of the overall length of the system. The final lens in our assembly is the objective lens, whose job is primarily accepting light. This consists of gathering the rays from the object so that the image can even be formed, but as well as accepting more ambient light to illuminate the image even further and providing a wider field of view for the user, depending on the diameter of the lens.

Another aspect of the lens research was seeing each components relevance to the industry, and what would exist in the market but not necessarily in research. One major component is the targeting reticle in riflescopes. As an optics student focusing Gaussian profile beams into either a CCD camera or to a beam stop for analysis, there is no need for any sort of targeting reticle, and the closest we would have gotten to one would be passing the beam through some type of object to analyze the shape in the image or Fourier plane. However, in this situation, the object is placed at infinity and the image falls on the user's eye. The reticle is what allows the user to physically see the optical axis, and align it with their target, though it is only viable at a specific distance. That is what the tick marks on a reticle represent, the amount that a bullet would drop over a specific distance and is usually calibrated for a specific bullet size or grain weight. The reticle is generally in one of two places, the first being at the distance where the focal distance for the objective lens and twice the distance of the focal length of the image reversal lens meet. This is called the first focal plane. This reticle will increase in size as the magnification increases and allow the bullet drop coefficient of the reticle (the tick marks) to remain viable at any level of magnification. This is the preferred setup for our design.

The second option is between the eyepiece and image reversal assembly and is referred to as being in the second focal plane. This reticle retains its size to the perspective of the user regardless of magnification. The bullet drop coefficient for this reticle is viable only at a specific magnification, which is usually the highest magnification that's available. The reason one would use this configuration is for exclusively long-distance shots, as the lower magnification would only be used to help spot the target, which would then be zoomed in on. The benefit is that at

lower magnifications, the field of view is wide enough to acquire the target, but the magnification would increase to obtain a clearer image.

3.1.3 Laser Rangefinder

Being able to measure a large distance quickly and with moderate amount of accuracy is an essential requirement for this project. There are many telemeter devices out there on the market today, we are interested in the technology used in the firearms market. Typically, nearly all types of telemeter devices consist of an emitter device and a sensor. In this project, the Laser based telemeter is going to be used as it will allow us to quickly and accurately measure long distances.

Some of the current laser rangefinders on the market may fulfill our requirements when it comes to the distance we need to measure; but they are often far too large to properly fit the size constraints of the project. These products often times include components we don't need like builtin optical scopes. This is why we will be avoiding these and instead designing our own rangefinder. To do this, we are going to need to find a laser rangefinder emitter and laser rangefinder sensor that are able to measure within the range of our distance requirements.

One major difficulty is finding a laser rangefinder that will fulfill the range needs of the project. In the case of this project we may need to compromise for a rangefinder with a shorter distance to stay within the budget. Unfortunately, while rangefinders for shorter distances are rather easy to come across, finding one that can support up to at least 1km is rather challenging. They are typically sold by companies that specialize in them and they are not usually cheap.

Common uses of rangefinders

- Military
- Sports(golf, hunting, archery)
- Measuring tools
- 3-D modeling

In the case of this project, military grade would be the best one to use. They are highly accurate, measure long distances, and are very reliable. They are also very expensive, far outside the range of the projects budget. The next best option would be the rangefinders used in sports like golf and hunting. These rangefinders are less accurate than the military grade rangefinders, but are still accurate enough to be a good solution for this project. Another reason for choosing a sports grade rangefinder is that they are much more accessible and significantly cheaper.

3.1.3.1 Emitters & Receivers

The laser rangefinder is mainly composed of two main pieces. The emitter and the receiver. The emitter is responsible for sending out pulses of light while the receiver

is responsible for sending a single when it detects the light bounce back to it from the target.

3.1.3.2 Emitters

The emitter is responsible for, as the name suggest, emitting a laser that will travel to the target. These lasers are almost always class 1 or class 2 lasers. The classes are not heavily regulated as they are not very dangerous [9]. The class 1 laser used in the longer distance rangefinders we are using operate at a wavelength of around 905nm. This wavelength is infrared and thus cannot be seen by the human eye.

3.1.3.3 Receivers

The receiver is a sensor that is responsible for detecting the laser emitted by the emitter. When the laser reflects off of the target it returns in the direction of the emitter. Because of this, the receiver must be placed close to the emitter so it can pick up the reflected beam. The sensor is a photodiode that is tuned to have the correct frequency response at 905nm. Once the sensor detects the light it can then stop the timer to calculate the time of flight.

3.1.4 Time-to-digital Converters

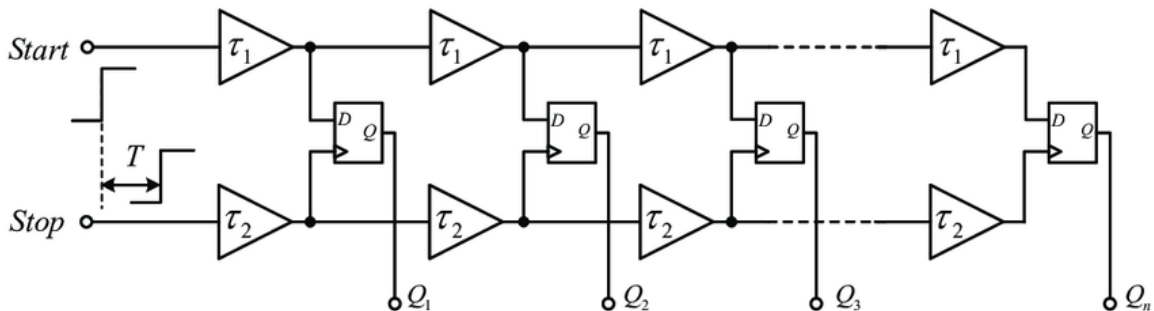


Figure 3: Time-to-digital converter schematic [17]

The time-to-digital converter is an important part of a laser rangefinder as it allows us to calculate a time-of-flight which is used to calculate the distance from the rangefinder to the target. Time-to-digital converters can be used to measure very short periods of time. They are often capable of measuring ranges from 1 nanosecond to a picosecond [2]. Being able to measure this level of detail is necessary when building something like a rangefinder as you are measuring something that travels at the speed of light.

The time-to-digital converter has two inputs, which we will call events. The start event and the stop event. The start event is used to tell the counter to start counting. The stop event, will of course, cause the counter to stop counting. The counter

counts the number of clock-cycles that passes when the clock is running. More often than not, these time-to-digital converters will use a crystal oscillator[3] to produce a high resolution time.

In general, for a time-to-digital converter, the higher the resolution the lower the range you are able to measure. This is perfectly fine for a time-of-flight calculation. In this project the maximum distance we would be measuring would be 4km, 2km to the target, and 2km to return. Since we are measuring photons, which travel at the speed of light, the range we need to be able to measure is between 0 nanoseconds and 13343 nanoseconds.

3.1.5 OLED Displays

While designing the project, we had to come up with a method to display information to the user without obstructing their view. We went through many different designs, but eventually came to the conclusion that an OLED display would be a good technology to use for this. The only problem was we had to find one on the market that would be suitable for our needs.

With most digital displays, such as an LCD it is almost impossible to have these displays be transparent. Of course, it has been done before, but their transparency is usually very limited. This is due to the fact that these displays need a bright back light to display an image. While there are methods of moving this back light away from the screen to create a transparent space between the display and the back light, the display itself is contained within a box that would end up blocking the view through our scope. This of course means we are not able to use any of these very experimental designs for our project.

This is where the OLED panel come in. Unlike the displays that require a backlight, the OLED emits its own light. This doesn't mean that all OLED panels are transparent, but in some cases, you can create a semi-transparent display. If the substrate of the OLED panel is transparent then we would have a high-density transparent display that we are able to display information on. The displays themselves are often times only 70% to 80% transparent, but this is enough for our use case. In the part selection we will go over what options we found on the market while searching researching this part.

3.1.6 Microcontrollers

Micro controllers are incredibly diverse and very flexible in their design and application, however that does not stop us from deciding on the best micro controller for our project. For our project having the option to connect to various different external modules is vital so that we may extend our project past our scope if need be. With that in mind we need a micro controller that has options to connect via ADC, i2c and a Bluetooth/WiFi module. Most micro controller have an integrated ADC and I2C converter so this was not an issue to find however a micro controller with

a Bluetooth/WiFi module that was small was slightly difficult to find. During my research process I recognized that we would never find a micro controller that would be perfect for what we want, there was always going to be downsides to each of them. I constructed a table illustrating the advantages and disadvantages of each of the micro controller that I have selected as a potential for our project.

Microcontroller	Price	Size(mm)	Features
Arduino Nano	\$20.70	18 x 45	ADC, UART
ESP-WROOM-32	\$10 - \$20	18 x 25.5	ADC, UART, I ² C, SPI
Raspberry Pi 4	\$35	85.6 x 56.5	Quad Core Processor
MSP430FR6869	\$20	16 x 16	I2C, ADC, SPI
Arduino Portenta H7	\$100	66 x 25	ADC, UART, DAC, USB-C, Bluetooth
Arduino Uno R3	\$23	68.6 x 53.4	ADC, UART, I2C

Table 3: Microcontroller Comparison

Microcontroller	Price	Size(mm)	Features
Arduino Nano	\$20.70	18 x 45	ADC, UART
ESP-WROOM-32	\$10 - \$20	18 x 25.5	ADC, UART, I ² C, SPI
Raspberry Pi 4	\$35	85.6 x 56.5	Quad Core Processor
MSP430FR6869	\$20	16 x 16	I2C, ADC, SPI
Arduino Portenta H7	\$100	66 x 25	ADC, UART, DAC, USB-C, Bluetooth
Arduino Uno R3	\$23	68.6 x 53.4	ADC, UART, I2C

Table 4: Microcontroller Comparison Revised

Revision

In the second semester we decided to switch to the Arduino Uno instead of the Arduino Portenta H7. One reason for doing this was to reduce the cost of the project. Another reason was the ATmega328P that the Arduino Uno uses is easier to integrate into our PCB than the MCU that comes with the Portenta H7.

3.1.6.1 Arduino Nano

The first potential micro controller that would be a potential for our project is the Arduino Nano. This is a very base level controller that would satisfy our requirements considering the price it would be a highly effective option as a backup as it only has what we need not what we might want in the future. The upside of this controller is that because it is very bare bones it serves as a very good baseline for the project to run however if we do change our scope to include more modules then it will not be compatible for those extra modules.

3.1.6.2 ESP-WROOM-32 ESP32

Second on the list is the ESP micro controller, this one is a great all around micro controller as it has multiple possible integration's for any kind of setup required. The only downside this controller has is that you need to purchase modules separately from the microprocessor increasing the total cost to over 150\$ which is vastly over the budget that we set for our micro controller.

3.1.6.3 Raspberry Pi 4

The Raspberry 4 is quite popular in the world of robotics as it is essentially a mini computer in which you could run an entire operating system in that small computer. This is an excellent test unit for our project as it allows us to treat our micro controller as a computer and make adjustment's on the fly. The downside to the Raspberry Pi is that it an overkill to our project in terms our power consumption and CPU power output.

3.1.6.4 MSP430FR6869

This TI micro controller has been used in my courses since my sophomore year, as such I have become highly comfortable using it and implementing it in whatever situation I would need to. However this micro controller is incredibly large in size as well as difficult to program as the TI library is all done in C which is very verbose to program in.

3.1.6.5 Arduino Portenta H7 / H7 Lite

My number one pick for this project is the Portenta H7 as it is brand new technology from arduino. Includes a dual processor CPU that utilizes a low power input. As well as this micro processor has the ability to run TensorFlow which could be adapted to run computer vision machine learning to take our project to the next level.

3.1.7 Communication Interfaces

There are many different communication interfaces standards out there. We are limited by the intersection of what our components support and what our micro-controller supports. Below we will cover each of the interfaces options we have

available and choose the ones that suite us best. Because we have multiple components we must interface with, we may end up using multiple different interfaces.

3 .1.7.1 ADC

Analog to Digital Converter is the first form of communication that we will be implementing in this project, the ADC we would be using is a direct conversion or flash ADC. This type of ADC is extremely quick and considering the high performance is needed, if this design is to be used in actual combat.

The Flash ADC has high performance mainly due to two reasons: the SNR (Signal to Noise Ration) and sampling rate. The Signal to Noise Ration in this ADC is reasonably high at 8 bits as the function of the ADC is to record a button input, then we do not need much accuracy, instead we need a very high sampling rate and a large number of comparators to read in input. The Flash ADC has a sampling rate in the gigahertz range which is extremely useless as data loss while using ADC is quite common mainly due to noise. Part of the reason the Flash ADC has very high performance is due to the comparators, this helps us read in data throughout any voltage range with reasonable accuracy.

3 .1.7.2 I2C

Inter-Integrated Circuit is a multi slave, multi master serial communication device. It is mainly used to connect to other micro controllers or devices on the board for a short range communication tool. Typically you would have multiple slaves and one master even though multiple masters are allowed, the master drives the clock and initiates the communication when needed.

An I2C integration in our project would not be incredibly useful as it required a clock to transmit data, which would not be an issue if other components required it however since it is the only component it does not make sense to utilize it. Additionally, an I2C bus is constructed using the Master/Slave circuit which is not the ideal approach to our project as it has more of a spider web approach.

3 .1.7.3 SPI

Serial Peripheral Interface is a bus communication device that utilizes clock and data lines to send data from and to external devices, it utilizes a start bit and end bit to initiate data transfer and then data transfer will start for every high there will be one bit transferred the data will be represented by a high for one and a low for zero.

In our design, we plan to use the SPI communication to communicate with an external LCD screen to display data, we plan on sending information to display the number instead of sending the data itself as the data on a LCD screen is done in data lines.

3.1.7.4 UART

Universal asynchronous receiver-transmitter is a device-communication tool, that allows us to send data in the form of packets. Much like it is done on the internet, it has a very simplistic design of two ports Tx and Rx which represent a transmitter and receiver respectfully. It sends bits on the wire using a clock and can send up to nine data bits in one data packet.

In our design, we plan to use this tool mainly as a universal communication device amongst our external devices, as it is incredibly versatile and allows for creating packets of data instead of a bit-by-bit approach.

3.1.8 Batteries

There is a great variety of batteries that can be chosen for any given power design for this project as it currently is or if when adding new components in the future. Batteries come in every shape and size that we can think of and their specifications are very customizable depending on the loads that you may have in the project. Batteries can be a single cell, or 8-cell, or even more and they can be NiMH or lithium-ion.

For our project, we needed a battery that will be able to supply enough mAh to power on the components sufficiently and have their width measurement to not exceed the width of the space below the scope as it will be placed there. This battery, or a set of batteries (depending on the power design), needs to be able to power on all the components with their required voltage threshold. It needs to be rechargeable to be able to continuously use this project over and over again. Also, the AWG of the wires from the battery needs to be able to support enough amperage depending on the estimated load from the components.

Most batteries have low mAh so this was crucial to find one with the most mAh while also considering other aspects such as voltage and size of the battery. While I noticed that all batteries are very similar in the AWG of the wires, I noticed in my research process that it would be hard to find a battery with wires of bigger AWG sizes bigger than 18 AWG. Most of the batteries carry 18 AWG wires which support up to 2.3A, according to the National Electrical Code. [5] Also, I noticed that there was an extensive selection of sizes for batteries because of the different ways that the cells are arranged, they can be two rows of four cells or four rows of two cells. Choosing the correct one will drastically change the size of the battery. Lastly, finding a battery that was rechargeable wasn't difficult as it is common for batteries to be rechargeable. Below on Table 10, it illustrates the advantages and disadvantages of each of the batteries that have been chosen to potentially be included in our project. The prices of the batteries below are just for the battery, it is not including the costs of battery chargers.

Battery	Voltage (V)	Size(mm)	mAh	Weight(g)	Price
Lithonia ELB-4865N	4.8	57.4 x 14.4 x 50.2	900	108.8	\$9
BGN-MHAA1800-10DWP-MINI-S110EC	12	113.0 x 48.5 x 14.2	1800	217.7	\$43
LAMPVPATH 3 AA Battery	4.5	68.0 x 48.5 x 18.2	2850	95.0	\$8
CO-RODE 8 AA Battery	12	127.0 x 71.1 x 20.3	2850	334.2	\$10
Tenergy Flat NiMH Battery	9.6	103.8 x 58.9 x 16.0	2000	220.0	\$17

Table 5: Battery Comparison

3.1.8.1 Lithonia ELB-4865N

This 4.8 V battery along with the 12 V battery below, BGNMHAA1800-10DWP-MINI-S110EC, was our first choice. These batteries work in a bundle to power on all the components. This battery would be a great choice as it satisfies our electrical requirements for our power design, but it lacks enough mAh to power on the components for a sufficient period of time. With the lowest at 900 mAh, it doesn't compare with the rest of the batteries which are closer to 2,000 and above.

3.1.8.2 BGNMHAA1800-10DWP-MINI-S110EC

This 12 V battery is bundled with the previous 4.8 V battery. This battery has a good rating of 1,800 mAh that would be sufficient for the components that it is powering. This battery costs \$43 which is expensive when comparing it to the rest of the batteries, it costs on an average of \$32 more than the rest of the batteries. So, choosing this battery goes against our goal to keep this project cost-efficient. Also, the weight combined with the previous battery would make this bundle of batteries too heavy for our goal coming at around 326.5 g.

3.1.8.3 LAMPVPATH 3 AA Battery

This 4.5 V battery along with the 12 V battery below, CO-RODE 8 AA Battery, was our second choice. These batteries work in a bundle to power on all the components. This battery would be a great choice with a lightweight design, cost-efficient,

small size, and the highest mAh at 2,850. On the other hand, the issue with this battery is that it delivers 4.5 V so if the batteries are a little drained it will not deliver 4.5 V anymore, it would be less. Or, if the load is drawing too much current, there will be a voltage drop which we don't want. These scenarios happen with any battery, but this battery is rated for 4.5 V and that is already the minimum recommended voltage that the MCU needs to be powered on. So, we would need a battery that is 4.8 V at a minimum.

3.1.8.4 CO-RODE 8 AA Battery

This 12 V battery is bundled with the previous 4.5 V battery. This battery has a great rating of 2,850 mAh and a cost-efficient price. The advantage of this battery and the previous one is that they are powered on by AA batteries so they are easily replaceable and you won't need to wait to charge your battery. The disadvantage of this battery is that the weight is the highest of them all at 334.2 g, the next heaviest battery is at 220.0 g which is quite a difference in respect to the weights of the batteries listed in the table above.

3.1.8.5 Tenergy Flat NiMH Battery

Our number one pick for our project is this 9.6 V battery. It has the best balance of all the specifications with 2,000 mAh, 220.0 g, and a cost-efficient price. This battery doesn't have the same voltage rating as the other batteries because we will be using this battery by itself to power on the entire power design. We will be supplying 5 V by stepping the voltage down and supplying 12 V by stepping it up. This battery is NiMH, but it doesn't matter what kind of battery it is for our project.

3.1.8.6 State of Charge

In this section, we will discuss the state of charge of the battery that will be used in this project. At best, the table below is an estimate of what the values really are. It will illustrate the voltage level for the 9.6 V, 5 V, and 12 V batteries. The 5 V and 12 V are derived from the 9.6 V battery as they are stepped down and stepped up from it, respectively. [6]

Battery	9.6V Battery	5V Battery	12V Battery	Specific Battery
100%	10.11	5.36	12.73	1.277
90%	10.03	5.30	12.62	1.258
80%	9.95	5.24	12.50	1.238
70%	9.87	5.18	12.37	1.217
60%	9.78	5.11	12.24	1.195
50%	9.69	5.01	12.10	1.172
40%	9.59	4.97	11.96	1.148
30%	9.50	4.90	11.81	1.124
20%	9.39	4.82	11.66	1.098
10%	9.29	4.74	11.51	1.073

Table 6: State of Charge

3.1.8.7 Voltage Regulators

In this section, we will discuss the difference between the linear regulator and the switching regulator and the advantages and disadvantages of them both. In our project, we need a voltage regulator for our power supply design in order to properly use the 9.6V battery as the only power supply in our system to then step it down to 5V to be properly fed to components and the MCU. We need to make sure we're choosing the correct choice in the voltage regulator because we want to be outputting the correct voltage, amperage, with the least amount of noise and ripples, and the most cost-efficient. Looking at these characteristics when choosing it is vital because our components could be permanently damaged if we supply the incorrect voltage to them.

Below in Table 7, you will see the difference between the linear regulator and the switching regulator. The most important aspects that we're going to be looking at are design flexibility, efficiency complexity, cost-efficient, and noise. These aspects will help us determine which one we should choose.

Characteristics	Linear Regulators	Switching Regulators
Design Flexibility	Buck	Buck, Boost, Buck-Boost
Efficiency Complexity	20% to 60% for low difference between voltages and it depends on the input voltage	80% to 95%
Cost	0.50¢ to \$2.00	\$3.00 to \$30.00
Noise (RMS)	10µV to 170µV (or 0.003 %Vo)	10mV to 35mV

Table 7: Voltage Regulator Comparison [15]

As you can see from the table above, the best choice for our project is the linear regulator. This is because they are cost-efficient which aligns with our goal, the noise generated is measured to be low, and the efficiency complexity is low to medium.

3.1.9 Optical Calculations Research

The calculations for the zoom section of this scope were all done using ABCD matrices. These matrices, with the help of MATLAB, allow us to easily calculate the required characteristics of the components. In a multiple lens optical system, the ABCD matrix will be split into two different matrices that are used consecutively. The first matrix will be a 2x2 matrix whose A and D values are equal to 1, the C value would be equal to zero, and the B value would be assigned the distance of the lens along the optical rail. The second matrix will also share the same A and D values of 1, but now the B value will be zero. The C value will represent the power of the lens and will include the focal length of the lens. The actual value for C will be $-1/f$, with the sign convention being important to identify if the lens is a positive, or convex, or negative, or concave lens.

These matrices are done consecutively to calculate the overall effective focal length of the system, as this can be used to identify the power. The values for all of the lenses were chosen to satisfy the distance between the focal lengths, so the values were all chosen almost arbitrarily, though the focal lengths and distances for the zoom lens had to be calculated. This was achieved by assigning an arbitrary value to the distance for the position of the magnifying lens and solving the system of equations for the value of the focal length of the magnifying lens when the power of the system as a whole is equal to 1, which is the lower power goal for our system. This focal length was assigned with slight trial and error to find a realistic focal length for the lens. After the focal length was assigned, the system of equations

was calculated again, but the goal was to solve now for the distance that would achieve a 6 times magnification of the overall system.

The initial choices for the focal lengths of the lenses were chosen with the idea of the focal points aligning with each other but were arbitrary otherwise. The main goal was to accomplish a compact system that achieved variable magnification. In this case, compact is relative but the goal was to remain under about 300 millimeters in length. Some larger scopes can fall even in the 350 to 400 millimeter length, but those are generally reserved for an extremely high magnification, where the maximum we expect to achieve is 6 times.

In our research for how to organize the distance between lenses and select the focal distance, one of the main aspects comes to the variable magnification eyepiece. In regard to focus, the focal length and orientation play major roles. However, when accounting for magnification, the primary factor is the distance between lenses. Keeping in mind that the first and last lenses of the eyepiece will be fixed, the only variable distance lens will be the magnifying lens. The bounds of which in can travel will be as close to the distance of the outside lenses as possible. Of course, the restraints for the picking the lenses come down to selecting something that is easily accessible, as our group lacks the resources or time to custom order lenses, so the variable of the system will be position along the optical axis.

In industry, there are more lens in a riflescope system, such as extra lenses in the eyepiece to ensure focus at a specific distance of eye relief, and the objective assembly is comprised of multiple lenses to account for aberration and field of view. Our design aims to be simpler and more streamlined, but one of the main factors for this does come down to the intricacy of that level of optical system, and our lack of experience with that complicated of a design. However, even if there are more complicated systems that exist, it does not necessarily imply that our system will not be sufficient. If we had learned to a higher-level optics prior to the start of the design, such an idea would be explored.

3.1.10 Wireless Protocols

Keeping the device easy to use is an essential requirement of this project. While we could always add more wires to connect every device we need to communicate, it can very quickly become inconvenient for the user. In the case of interacting with an existing device a user might have, such as an smart phone, avoiding wires is even more important. This is because many smart phones have various different connectors, like USB-C for modern android phones and the Apple Lightning Connector on iPhones. Not only would it be inconvenient for us as engineers to develop different interfaces to interact with all the different cable types; it would also be inconvenient for the user to have to deal with cable adapters to support whatever interface we support. Because of this, we want to find some method of communication that doesn't require any wires that tether the two devices together. The first thing that comes to mind is wireless communication that is already supported by

many existing smart phones on the market.

There are many different wireless protocols out there that are used for communicating between different devices. In our project, we need to be able to communicate between an app on either an Android or iPhone and our microcontroller. The wireless connection will need to support sending small amounts of data periodically in both directions between the app and the microcontroller. We will want something that does not require any equipment besides the phone and the microcontroller to communicate. The communication protocol must be supported by both devices, which will help narrow down what technologies we should research.

The two most popular wireless protocols supported by smart phones and a subset of microcontroller on the market are Wifi and Bluetooth. In the following section we will cover the differences between these two protocols and decide which one would suite our project best. A few key factors we will be looking for is ease-of-use for the end use, how difficult it will be to implement it on the engineering side, and what protocol is more widely supported by both devices.

Standard	Bluetooth	Wifi
IEEE Standard	802.15.1	802.11n
Frequency Band(GHz)	2.4	2.4 & 5
Distance(m)	10	90
Bandwidth(MB/s)	6.25	65

Table 8: Wifi vs. Bluetooth

3.1.10.1 Bluetooth

Bluetooth is widely supported across many devices. Based on the previous microcontroller we have looked at, the Bluetooth version we would be using is either Bluetooth Low Energy 5 or Bluetooth Low Energy 4.2. The iPhone supports Bluetooth 4.2 and so do many android phones. So we should opt to use 4.2 over 5 so we can support as many devices as possible. Bluetooth was invented in 1998, so it is a very mature and stable technology making it a good choice.

Along with being widely supported we must also consider how difficult it is to implement this through code. There are many libraries of interest when it comes to implementing a Bluetooth interface. On many microcontrollers the interface will end up being similar to any wifi interface provided, but in terms of app development, it can vary greatly. Android itself provides a standard library for interacting with a Bluetooth device. There are also many open-source third party libraries that simplify interfacing with a Bluetooth device.

3.1.10.2 Wifi

While in general Wifi is a faster, more reliable, and more secure networking interface, it comes with its own problems. In terms of using this in your home network to connect multiple computers, that is very sensible. But in our case, we will not be able to utilize this very effectively. Due to the nature of the project, the user will often likely be in places where they are unable to have access to a wifi connection. Unlike Bluetooth which will connect peer-to-peer, wifi will need an access point to connect. This means if the user is not connected to any wifi they will be unable to use the app to connect and control the microcontroller.

Along with limiting the locations a user is able to use the device, it also does not provide much benefit in terms of what it brings to the project. The higher speeds are nice, but it is very unlikely we will even come close to maxing out the bandwidth of Bluetooth, so we would not get any benefit out of using wifi here. Not only will it not provide much benefit, but developing on a wifi network will end up being more complicated. As mentioned earlier, Bluetooth has many open-source libraries for easily interfacing with Bluetooth devices, this is not the case when it comes to wifi. For these reasons, we decided against using wifi in this project and will instead use Bluetooth

3 .2 Part Selection

This subsection will cover how we decided on what parts we decided to use for our project. We will go over different components provided by different companies and why we chose the ones we did. Some components had many options, such as the microcontroller, while others like the component chosen for the targeting lens had very limited options.

3 .2.1 Laser Rangefinders

When it comes to laser rangefinders, there are many options on the market. Unfortunately, many of these options either can't measure a long enough distance, are too expensive, or are inaccessible due to the company that produces not selling to average consumers. Because of this our options were limited and we had to compromise on one that compromises between both distance and price.

Model	Distance	Accuracy	Power
H600A	600 Meters	±1 Meter	50W
HR50A	1.5 Kilometers	±1 Meter	50W
DLEM 20	5 Kilometers	±0.5 Meter	1.8W
DLEM 30	14 Kilometers	±1 Meter	2W
DLEM 45	20 Kilometers	±1 Meter	2W
LDM301	3 Kilometers	±60 Millimeters	5W

Table 9: Laser Rangefinder Comparisons

3 .2.1.1 DLEM 20, 30, and 45

The DLEM series was one of the first laser rangefinders we looked into. This rangefinder was almost perfect for our project. The model 30 measured up to 14KM which would have been an ideal range as it would allow to scope to work on many different types of rifles with different ranges and different situations. The size of it was 95mm x 25mm x 50mm which would be small enough that it would be negligible relative in terms of bulk and weight added to the scope.



Figure 4: DLEM 30 Laser Rangefinder

As far as size, power consumption, accuracy, speed, and range this rangefinder beat out all the other options several times over. The accuracy was significantly better than other rangefinders in terms of distance compared to error. The power usage was 25x less than other viable models that measured less than 1/20th of the range.

The manufacturer produces several different models of this rangefinder all at different measurement ranges. The shortest measurement range is the DLEM 20 which can measure up to 5 Kilometers. This is a very good range and will support most applications of the scope. The next model up, the DLEM 30 measures up to 14 Kilometers. This would cover nearly all applications and would probably be the best choice for this scope. Finally, the model with the longest measuring range, the DLEM 45, at 20 Kilometers was by far the most powerful one. This only would likely end up being overkill and would add unnecessary cost to the project.

Unfortunately, the company that produces this laser rangefinder, JENOPTIK, was not very accessible to your average consumer. After attempting to reach out to them and not hearing any feed back, we had to drop this from our list of feasible options. Price was also a concern since this company typically sells to industrial and military organizations.

3 .2.1.2 H600A

The H600A is a very affordable laser rangefinder, one of the cheapest we have found. But, the measuring range is only 600 meters. This means it would be quite a bit shorter than what we would want. A range of 600 meters might work in some applications such as an airsoft gun or a paintball gun, it would not fit the needs we would need for an actual hunting, sports, or military rifle.

The rangefinder provides a basic digital output, but as far as documentation is concerned, it is rather sparse. This means that it might be rather difficult to write the appropriate software for it. As far as dimensions go, the total size of the scope is 115MM X 58MM X 35MM. This is on the larger size of what we might want. While we could go a bit bigger, this rangefinder does not provide enough value to justify that amount of space.

Due to the short range, poor documentation, and rather large size for what is capable of we decided to not go with this one. While it may be fine in some contexts such as airsoft or paintball, for hunting and sport applications we will want at least 1 Kilometer. Because of this, we decided to keep looking.

3 .2.1.3 LDM301

This rangefinder seemed like it was going to be a good option when we first looked at it. The measurement range was all the way up to 3 Kilometers. The dimensions are pretty average at 136mm × 57mm × 104mm, meaning it was competitive with other rangefinder we have looked at. The power usage is quite low as well at only 5W during operation. The accuracy is quite impressive as well, the rangefinder is accurate up to 60 millimeters. This means it would be the most accuracy out of all the ones we have looked at so far.

As far as interfacing with the rangefinder, it provides a few different interfaces. These interfaces included RS-323, RS-422, and SSI 24 bit. Unfortunately, we

weren't familiar with any of the interfaces the rangefinder provided. Of course we could have just taken the time to learn these interfaces, but we discovered a few more issues with the rangefinder. Firstly, we found out that the rangefinder is only accurate at 3 Kilometers if the target is a special target meant for the rangefinder. Otherwise on natural surfaces that weren't made for the rangefinder, it is only able to measure up to 300 meters. Next we ran into the same issue as we did with the DLEM modles where the company we were attempting to buy from would only sell to industrial applications.

Because of these reasons we decided against going with this rangefinder. While the rangefinder seems like product that would work well for our application, that perception fades away when looking a little more into it. The requirements need for it to be able to measure up to 3 Kilometers would not exist in the context of our product. Being unfamiliar with the different interfaces provided by the would not be a deal breaker, but would count against it. Finally the company being unresponsive to our requests made us decide that it would be better to look elsewhere.

3.2.1.4 HR50A

As we kept searching we finally came across the HR50A. This rangefinder could measure up to 1.5 Kilometers. This means that it would fit the minimum requirements needed for the distances we are trying to measure. The pricing was comparable to the other products such as the H600A at around \$350. While this is a bit expensive, considering the limited options on the market that fit our needs, it was one of the best options we had come across.

To interface with this microcontroller we are able to use the UART protocol. This is fortunate because the team was familiar with this do to previous classes we had taken. Alongh with using a familiar interface, the documentation for running the rangefinder were, while not extensive, good enough that we could get a working rangefinder out of it. This meant that as long as all the other specifications checked out, this seemed like a good solution as far as laser rangefinder go.

When it comes to measurement range, this laser rangefinder was on the lower end of the previous components we have looked at. At a maximum of 1.5 Kilometers, it still fits within our target range, so while it is not perfect, it is one of the best we were able to find. For our target audience of hunting and sports, it should be sufficient enough to meet most of their needs. As for military it would likely fall short in terms of distance, but as long as have a known way of interfacing with a rangefinder, we would be able to swap between range finding modules without too much trouble.

The power consumption of the component peaks at around 50W during usage, which while being significantly more than the highest end rangefinder we looked at, is still feasible for us to power. The rangefinder itself will only be running for a few seconds during each measurement, so the amount of power it consumes overall will not be all that significant.

The dimensions of this rangefinder comes in at 135mm X 77mm X 40MM. This ends up being one of the larger rangefinders, but it is possible that we will be able to make modifications to the housing to reduce the size. While it might be heavier than the other models we have looked at, it is still more capable in terms of price, features, and accessibility. The total weight of the unit comes in at 270G which is not all that much. If we end up modifying the housing of this unit it would also reduce the weight which would be helpful in keeping the product as light weight as possible.

Overall this is a component that we had to make many compromises on. After weighing the pros and cons of each device, we came to the conclusion that this component will fit our needs the best. The price is within our budget and as far as buying it, the component is easily ordered online and we do not have to go out of our way to contact and negotiate with a company. The measuring range may not be the highest, but it meets our requirements and will be enough to build a product that can compete with others. The interface is a standard that everyone on the team is already familiar with meaning using the device will be easy enough for our team members to pick up. As for the dimensions of the device, while not ideal, we should be able to overcome any issues with a bit of ingenuity. For these reasons, we decided to go with this component as our choice for our laser rangefinder. While it is not perfect in every aspect, it meets all of our requirements all while being accessible enough that we can actually buy it and cheap enough, that while being a bit expensive, is still within our budget.

3.2.2 OLED Displays

When it comes to finding a proper OLED display that will work for our needs, our options were very limited. The only company that provided a display that works for our needs was Crystalfontz. The reason for this is that they are the only company who provides a display that is not only transparent, but also small enough to fit in our scope.

There were several other display which would have been interesting to look into if they were transparent as well. This includes rounded displays and higher resolution displays. All of these displays we looked at were OLED, but unfortunately had an opaque substrate making them completely opaque and therefore useless for our use case. On the off chance that the company were are buying from ever makes on of their rounded displays transparent, we would be able to simply swap it with our chosen display as it uses the same interface as the OLED display.

Display	Resolution	Size(mm)	Brightness
CFAL12856A0-0151-B	128x56	42.04 x 27.22 x 1.25	120 Nits

Table 10: OLED Display Comparison

In terms of what we are looking for in a display, the CFAL12856A0-0151-B meets a suitable amount of requirements. The display should be inexpensive. This display only costs us \$26.08, making it very affordable considering it is an essential part of the project. While the size of the display is not perfect it is good enough that it will fit our needs. Ideally the display would be a square and not a rectangle, but depending on how we decide to mount the display in the lens it will likely end up not being much of a problem.

Interfacing with the display is something to take into consideration. The fact that this display has a controller already built into it makes the development of the project considerably easier. This display uses the Solomon Systech SSD1309 which supports several different interfaces, including 8-bit parallel, I2C, and SPI. Since the microcontroller we chose supports both SPI and I2C will be very simple. To interface with the display, we will only have to build a board that takes the 24 pin cable attached to the display and convert it to an I2C signal for our microcontroller.

4 Design Constraints & Standards

In this section we will go over the different constraints and standards imposed upon us. The first subsection will be the constraints that our imposed upon us by our environments. This includes things such as the time we have to complete the project, our budget, as well as social and economic constraints. In the next subsection we will cover the standards imposed by various different organizations such as ANSI.

4.1 Constraints

There are several different constraints imposed upon us by factors within and outside of our control. The factors include things such as cost, health, environmental, safety, and time. We are also living in unique times meaning we have more constraints like the COVID-19 pandemic and the consequences that follow from that. In this subsection we will cover each of the constraints we faced while designing and building our project.

4.1.1 Cost Constraints

This project is completely funded by four students of the University of Central Florida. Because of this, we do not have the luxury of being sponsored by outside institutions meaning our funding is limited to what we are able to afford. This means we may have to opt for cheaper parts as well as parts that are less than what would be ideal for a product manufactured by a large cooperation.

While the cost should not constrain the overall idea of the project it might affect some parts. The most likely parts it will affect are the rangefinder and the targeting lens. Rangefinder are very expensive at long ranges, so using a shorting range rangefinder maybe be necessary. The targeting lens has two designs, one of which could wind up being very expensive to produce a single unit of. This means we may need to opt for our second design of the targeting lens, which would be significantly cheaper.

4.1.2 Health, Environmental, and Safety

There are few standards that exist regarding safety, the environment, and rifle-scopes, as the scope itself is made out of materials that are not damaging to the environment, and there are very few safety issues regarding just the scope itself (excluding the firearm). The expectation on the manufacturer is that the aluminum for the housing, as well as the glass for the optics, would be reliably sourced and handled. One main public safety standard that exists is safe handling of tritium, which is present in many different applications for firearms. Tritium is a self-luminous ion of hydrogen that emits beta particles, which is beneficial to the firearm industry as it allows for visibility of a reticle or sight system, even at night. One model of

riflescope that is famous for using tritium in their products is the Trijicon ACOG. When paired with a section of fiber optic cable with the tube of tritium placed at the end, the fiber optic cable will provide light for the reticle during the day, while the tritium will provide light in low light situations. The tritium that is placed within a component is generally protected by two aspects. The first is that the cannister itself is placed in a larger aluminum cannister, that not only protects the unit from damage, it also doesn't allow it to glow in all directions. The second is a layer of glass that covers the side where the light is intended to be emitted from. This piece of glass is usually subject to great force when the weapon is being used, so most products will use a high-quality crystal, such as sapphire crystal. This is the same scratchproof material that is used in higher end watches to ensure long life. Although there is not enough tritium in any one weapon component that can cause any sort of radiation hazard to an individual, there is enough possessed by the manufacturer that regulations must be put in place. These regulations, put in place by the Department of Energy, the Environmental Protection Agency, and the Nuclear Regulatory Commission. Generally, these regulations just govern how the tritium must be transported, stored, and accounted for.

One main design that is also considered for safety, though it has also been grouped with common sense, is that the eye relief must be sufficient in length that one must not put their eye so close to the scope that the recoil would cause them to be struck in the face. However, most designs do not call for an excessively long eye relief. Scout scopes, which are generally placed on pistols, do have a long eye relief, as the user is expected to be able to see through the optic when it is placed at arm's length, however 50 to 100 millimeters is generally acceptable for a riflescope.

4.1.3 Time Constraints

Several parts of the project are time sensitive to other parts of the project. We are able to develop some pieces of the project in parallel, like the scope and the rangefinder. But some pieces require designs from other pieces to be completed before they can be started. For example, the microcontroller relies on the laser rangefinder before the code can be completely developed for it. The power supply can only be developed after we need the power needs of all our other components. The targeting lens can be developed somewhat in parallel to the scope, but only after we know the dimensions of the scope.

4.1.4 COVID-19 Constraints

The COVID-19 pandemic has introduced many constraints into the development and construction of this project. Currently it is difficult to find many parts and get them shipped overseas due to backed up cargo ships. There are also many supply chain shortages when it comes to electronic devices. Due to this, we may not be able to implement every feature we want or implement them to the extent that we would want.

The pandemic has also introduced many price increases causing our budget to be restricted even further. Not only are the base prices of many products increasing, but their shipping prices are increasing significantly. Because of this we have to be more careful with our budget and timing.

Finally the last constraint the pandemic has imposed is meeting in person. While most of the state is opening up, it is still good to be cautious. Minimizing meeting in person while still being able to design and work on the project is important. If one of us were to become ill due to the virus, that could end up causing more problems than if we were to stay socially distanced.

4.1.5 Programming Constraints

The microcontroller that was selected has the ability to program in python and that is what will be used for the entire project, since microcontroller programming was mainly done in C up until recent, a package will be installed called microPython to access microcontroller functions such as UART connections. Initially we would need to separate our code into functions that contain similar functionalities so that our code can run efficiently and not have run times greater than order n squared.

4.2 Standards

The following is a list of standards and regulations imposed by different organizations. The standards relevant to this project are created by the IEEE and the ANSI. These standards are created for the intention of creating a safer and more accessible product.

The Institute of Electrical and Electronics Engineering, also known as the IEEE, is an institute that is responsible for producing at least 30% of Electrical Engineering literature [1]. The IEEE was formed in 1963 and has been promoting electrical engineering education ever since.

American National Standards Institute, also known as ANSI, is an American specific non-profit organization that facilitates the publication of voluntary standards. This organization provides standards that are relevant to this project, specifically regulations concerning the operation of lasers and safe use of power supply circuits and batteries.

4.2.1 ANSI Z136.1

The laser used in this product is a Class 1 laser. A Class 1 laser is considered safe and is exempt from following any beam-hazard control measures. [9].

The following list is a brief list of regulations taken from the ANSI Z136.1 [9]

- Safe for 100 seconds

- Considered safe for intentional exposure including the naked eye.
- May not exceed exposure for long duration of 100 seconds or 30000 seconds.
- May not emit laser radiation levels that may cause skin or eye injuries.

The use of the lasers in our project fall well within the safety standards created by ANSI. When operating the laser in this project, the laser will only be active for a short period of time while measuring the distance to the target. This will be less than 100 seconds of active time.

4.2.2 Python Programming Standards

Programming standards exist in order to create readable code and code that could be understood well at a glance, python has very similar standards with some variation.

Indentation is highly important in python as it details scope of for, while, and function blocks. "Indentation need to 4 spaces per indentation level", additionally due to large function headers when utilizing the indentation lines are limited to "79 characters per line". Blank lines are utilized sparingly in python as a method to identify similar blocks of code, functions should contain two blank lines before and after the function to separate them from other blocks.

Imports are extremely important in python as they allow us to access libraries that we would normally not have access to, python has standards when it comes to importing and naming the imports. Imports always need to be at the top of the file before any code has been written, they have a specific grouping depending on the kind of module being imported. Standard libraries need to be imported first at the top of the file, followed by "Related Third Party Imports", and followed by "Local Application/Library specific imports". Python standards also state that explicit imports be preferred over implicit as they "give less errors" as well as they are more readable.

Naming conventions are important to the programmer as they give an indication as what that variable/function is being used for. "Class names should normally use the CapWords convention", not to be confused with the CamelCase convention that does not capitalize the first letter of the first word. Variable names should also be using CapWords and generally they should be short and concise, e.g A variable used to store data for looping should be called i instead of loopingVariable, "Function names should be lowercase with words separated by underscores as necessary to improve readability."

4.2.3 Power Supply Standards

The standards for power supply equipment are categorized into three classes: Class I, Class II, and Class III. These classes are labeled from the least safe to the safest in regards to electrical shock from the equipment. Class I consists of

equipment that may expose the surroundings to an electrical shock but prevents it from basic insulation on the wires and protective grounding of the system. This class does not guarantee full protection from a hazardous voltage that the equipment may output. Class II provides the same protection as Class I but it is reinforced insulation so, it doesn't require grounding. Lastly, Class III is the most protected, the equipment in this class operates from a Safety Extra Low Voltage supply circuit. This allows the circuit to be fully protected from any possible electrical shocks and/or hazardous voltages that it may output from the equipment. In Table 11 below, the circuit classifications will be further explained which we will cover four of the five classifications, which are hazardous voltage, extra-low voltage, safety extra-low voltage circuit, and limited current circuits.

Circuit	Description
Hazardous Voltage	Any voltage that is exceeding 42.2 V AC peak or 60 V DC without a limited current circuit.
Extra-Low Voltage (ELV)	In a secondary circuit, a voltage that isn't exceeding 42.4 V AC peak or 60 V DC, that is separated from hazardous voltage by basic insulation.
Safety Extra-Low Voltage (SELV) Circuit	A secondary circuit that doesn't reach hazardous voltage levels within the equipment and protective ground under normal operation or during a single fault. If during a single fault, then the voltage within the equipment is to not exceed 42.4 V AC peak or 60 V DC for a period longer than 200 ms. The absolute maximum allowed is 71 V AC peak or 120 V DC.
Limited Current Circuits	A circuit that is designed to not reach hazardous voltage levels within the equipment even during a fault and is accessible if it reaches and surpasses the levels of SELV. For frequency, $f < 1$ kHz, the steady-state current is to not exceed 0.7 mA peak AC or 2 mA DC. For $f \geq 1$ kHz, the steady-state current is to not exceed 70 mA and it is equal to $0.7 \text{ mA} * f$ (kHz). For accessible parts not exceeding 450 V AC peak or 450 V DC, the absolute maximum capacitance is 0.1 μF . For accessible parts not exceeding 1,500 V AC peak or 1,500 C DC, the absolute maximum charge is 45 μC and the available energy is to not exceed 350 mJ.

Table 11: Circuit Classifications

According to table 11 above, our Smart Compact Scope is under the circuit classification for Extra-Low Voltage (ELV). It satisfies the requirement for this circuit

type as it does not exceed 42.4 V AC peak or 60 V DC. Our power system has a maximum voltage of 15 V DC being inputted to some components. The wires from the power supplies to the MCU and components will be protected with basic insulation that is Polyvinyl Chloride (PVC) insulation. This insulation will protect the surroundings from any possible hazardous voltages that the system may output as an electrical shock.

Furthermore, the power supplies must meet a minimum insulation requirement of basic insulation with a minimum dielectric strength of 1,500 V RMS for primary to ground. The ground, or protective earth, must be present for Class I equipments and it must be labeled with green and yellow, or transparent covering. Between the load resistors and the grounded parts in the system, there should not be more than 0.10 Ω present. To test this key insulation requirement, a current 1.5 times the current capacity of the possible hazardous circuit would be applied until the grounded parts in the system become live. In this test, an absolute maximum voltage of 12 V AC or DC is to be used and may not exceed 25 A.

These standards that we set to meet with our project are crucial to meet to not cause any electrical shock to the scope and surrounding metal material. If a hazardous voltage was to be delivered outside the equipment because of not sufficient protection this could cause an electrical shock to be delivered to the user. So, our design needs to comply with the power supply standards and keep the power supplies, components, and MCU well protected to prevent this event. Lastly, to note, the International Electrotechnical Commission (IEC) and the associated International Organization for Standardization (ISO) are the main agencies that are responsible for the listed electrical safety standards. [7]

4 .2.4 Unit Standards

The reason why we are going to be using MOA measurements as opposed to MRAD for the adjustments and measurements on the scope is because of standards in the firearm community. Although both are used, MOA is a more common measurement that is understood by the vast majority of the community, where MRAD is more specific, even if it is also useful in the educational community as well. MOA stands for minute of angle, which equates to about 1 inch in height for every 100 yards of distance. The actual figure is 1.0472 inches; however, most distances will not be so long range that the rounding creates an issue. 1 MOA equates to about 0.0167 degrees, or 1/60th of a degree, where 1 MRAD is equal to 0.0573 degrees. This means that the MOA measurement can achieve more precise adjustments than MRAD, however MRAD would be more suitable to shorter distances. If a shooter is aiming out at 500 yards, and the firearm they are using has an accuracy that is measured as 1 MOA, that means that the variance on their shot placement would fall within a circle with radius of 5 inches. The use of MOA is a SI measurement, which is another reason for its popularity in the United States, as it is less commonly measured by its angle but more by its estimation at a specific distance. However, the accuracy of the firearm is independent of the accuracy

of the scope, but a pair between the two is what allows a shooter to achieve an accurate long-range shot.

The reticle for the scope will have between 40-50 LEDs along the bottom half of its vertical axis, and the reason for the excess number is so that more distances can be accounted for. Also, the smaller size of the reticle will allow for the user to have a better sight picture of their target. Most reticles will be measured not in millimeters, but in MOA, to account for the fact that the size of the reticle will appear to grow or shrink depending on the distance of the target. If the dot is too small, then it can be washed out of the viewers sight in a brighter setting. This can be adjusted by increasing the intensity of the LED, but that would eat into the battery life of the unit. It can be useful, however, in low light conditions on long-range targets. Alternatively, if the dot is too large, then it could cover the target completely. A popular short-range target reticle is 13 MOA, but that means at 500 yards out, the reticle would be 68.07 inches, which would completely cover most targets.

4 .2.5 Soldering Standards

The standards we're following for soldered electrical connections in our project is according to the National Aeronautics and Space Administration Technical Standard. The standard number for this document is NASA-STD-8739.3. We will closely follow these guidelines by this document when making our soldering electrical connections on our PCB, wire connections, MCU, and anything else that we need to solder a connection. It is crucial we are aware of these standards to create a solid connection and don't trigger any hazardous conditions to the user or any other components or parts in the project. Also, soldering electrical connections are sensitive if they aren't placed correctly so following these standards will help us make solid connections based on how we need to do them, based on if it's a through-hole, wrap-around, side route, or any other style of connecting wires to solder to the board or pin connection. [11]

When inspecting the complete soldered connections, some acceptable criterias to look for are: the solder joint surface is smooth, the lead contour is visible, and absence of defects. [11]

One of the most common solder connections that we need to do in our project are the pierced terminal as shown on Figure 5 and the heel fillet connection as shown on Figure 6. These connections will be done on the MCU. When the wire is pierced through the opening on the terminal there shall be a minimum of 90 degrees to a maximum of 180 degrees turn. If the wire is turned outside this range of degrees, it may cause some physical damage to the insulation sleeve. An example of the correct way to solder these types of pierced terminal connections is shown below on Figure 5. [11]

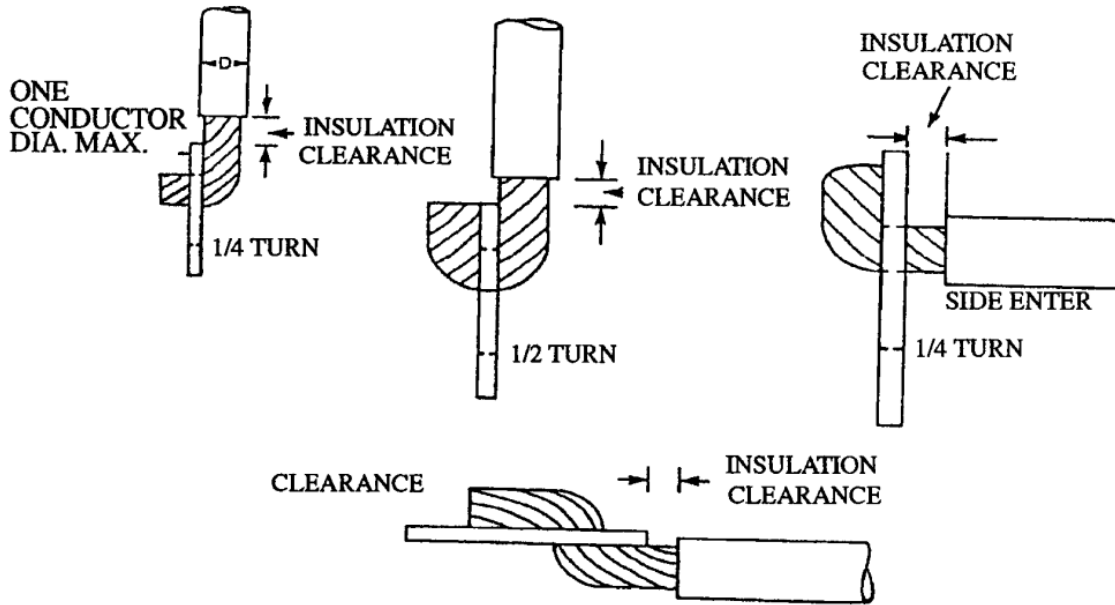


Figure 5: Connections to Pierced Terminals

As you can see in the figure above, there are many types of connections that can be done for the pierced terminal. Using this standard will help us determine the correct turn that we need to apply to the wire when soldering to the board. We can also determine from the figure that we are required to leave some insulation clearance to avoid the insulation from being in immediate contact with the board, or to what it is being soldered to.

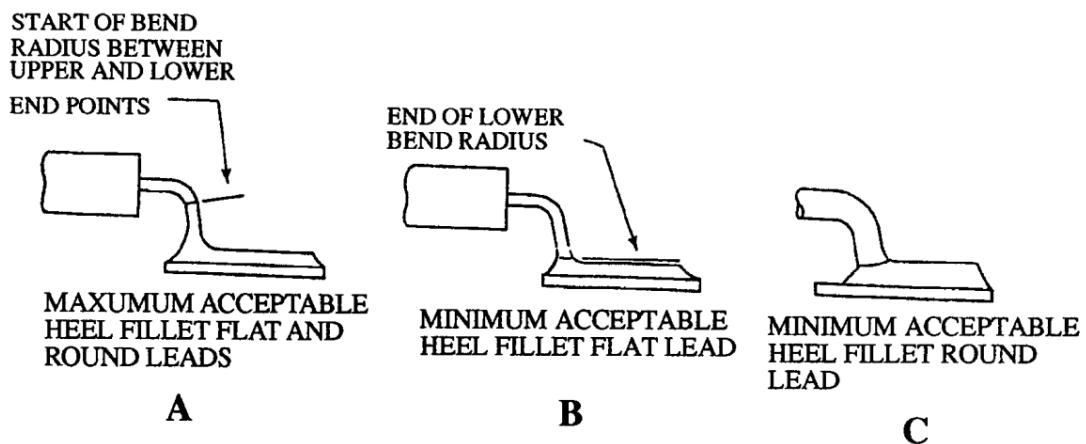


Figure 6: Connections to Heel Filler

In Figure 6 above, we can observe the minimum and maximum acceptable heel fillet that should be performed when soldering flat and round leads. According to

NASA, this connection is required for all single surface lapped solder joints. As seen in example B in the figure above, when performing this type of solder connection it is important to note that the connection between the heel and the pad needs to be continuous to comply with the standards. In other words, the connection between the beginning of the bend and the flat part needs to have a continuous smooth connection between them. [11]

5 Design

In this section we will take in depth look into the hardware and software designs of all the various systems. This first subsection will take a look at the overall design of all the systems. Next we will go into detail about the design of the optical scope lenses and the housing of the lenses. After that we will cover the design of the targeting lens and how it will be integrated with the microcontroller. Being able to accurately measure the distance to the target is important, so next we will cover the laser rangefinder. Then we will cover the design of the power supply that is going to be used to power the targeting lens, microcontroller, and laser rangefinder. To integrate all these systems together we need some software, so in the next subsection we will go over how the software works. Finally, we will take a look at the overall schematic and how each of these systems will interact with each other.

5.1 Design Considerations

During the design and development of this project, we made several design considerations that we will go over now. These considerations were affected by several different factors. One of the factors is cost, this is often the biggest factor when considering designs. Another factor is design difficulty and feasibility. Some parts we would have liked to build ourselves, but are not feasible either do to market constraints and manufacturing constraints. The last factor is time, some designs we would have liked to use would have taken too long to develop therefore we have to opt for something that can be designed and developed in a shorter period of time.

5.1.1 Design Consideration I

The first consideration we made while designing our project was the laser rangefinder. While it is easy and cheap to produce a short-range laser rangefinder, the long range ones are much more difficult. Finding individual emitters and receivers on the market that is able to measure in the 1km range proved to be impossible. After learning more about how laser rangefinders worked it became clear that producing these ourselves would not be feasible.

Seeing as it was not feasible for us to manufacture our own, the next step was to look for a laser rangefinder on the market. There are many options out there, but a majority of them are already in other products such as golf rangefinders and hunting rangefinders. Of course just buying a golf rangefinder and ripping out the laser rangefinder contained within might work, it is not practical or sustainable.

Since the previous two options were dead-ends, we decided to look for a company that specializes in manufacturing small long-range laser-rangefinders. Eventually we discovered a company called Jenoptik that does just that! But, in order to pur-

chase these rangefinders, you would have to contact them via email. Eventually we gave up on that as these rangefinders were likely to be far outside of our budget even though they were the perfect fit for what we are designing.

Finally after much more searching, we eventually came across a rangefinder on a site called AliExpress, a retail service based in China. The rangefinder was able to measure between 1k and 1.5k which was the range we were looking for. The price was also within our price range at around \$300. This is the laser we have decided to design our product around as it was the best compromise we could find.

5.1.2 Design Consideration II

Our next design consideration was the targeting lens. In our initial mock ups, we thought to use a single laser attached to a motor that would rotate along its vertical axis. With this we would be able to adjust the motor to point it on the targeting lens at the position we want the user to aim. After a bit more design and research though, we discovered this would not be a good solution. The recoil of the gun firing could easily cause the laser to shift and throw off the aim. It would also be more difficult to have a fine control over the laser and the code to program it would be rather difficult.

Our next idea was to have a glass or plastic lens with microLEDs along the vertical axis that we could control via our microcontroller. While this sounded great in concept and it probably feasible with the proper resources and time, it proved to be impractical for us. The first problem we ran into was finding microLEDs that you could easily purchase. Even if we were able to, we would need to be able to control each microLED individually. This is something that would have costed more than our budget would allow for and would not fit within our time constraints. We then thought we could reach out to a company that specializes in this type of product, but just like with the laser rangefinder we were unsuccessful.

Eventually we stumbled upon an unexpected solution. While searching for the microLEDs, we came across a small transparent OLED display. This display is produced by a company called Crystalfontz and they sell them both individually and in bulk. This was the perfect compromise for what we wanted. It was well within our budget and the only downside was that it is only 70% transparent; meaning that it will very slightly obscure the target. In the majority of scenarios this shouldn't be much of a problem. With this display all we had to do was design a board to power the display and facilitate the communication between it and the microcontroller.

5.1.3 Design Consideration III

One main consideration for this design is the basis of what type of firearm this scope is being made for. There are many factors that can impact the trajectory of a bullet at long range, aside from constants like gravity or environmental impacts such as wind, humidity, or the Coriolis Effect. Features of a firearm that impact the trajectory

generally include the barrel length and specifications, the gas system that operates the firearm, and the action type of the firearm. We will be standardizing the optic for a 5.56x45mm NATO round that is fired out of an AR platform with a 16-inch barrel that has a 1:7 twist and is semi-automatic with a direct impingement gas system. These are the details of the Colt M4 Carbine that is standard issue in the United States Army. Every aspect of this will be discussed as to its impacts, as well as how application integration (one of our stretch goals) will account for alternatives to the base model.

Starting with the basics, when a firearm is fired, the firing pin strikes the primer of the ammunition cartridge. This ignites the gunpowder, which transforms into gases that rapidly expand to push the bullet out of the barrel. In essence, the velocity of the bullet is dictated by the gases and their path. The barrel length impacts the velocity of the bullet as it gives the gases more room to push the bullet before the gas is dispersed into free space. Therefore, the longer the barrel length, the greater the muzzle velocity of the bullet. The twist rate of the barrel is what allows for a straighter bullet trajectory as a spiral pattern inside of a barrel rotates the bullet as it passes through the barrel. This acts very similarly to how a football spirals in air, allowing for a more accurate throw. The measurement of the twist rate is denoted by 1:X, which translates to 1 rotation of the bullet for every X number of inches traveled through the barrel. If the twist rate is too low (meaning that X is higher), then the bullet is not spinning enough and might tumble through the air, which greatly impacts trajectory. If the twist rate is too high, the centrifugal force caused by the rotations can spin the bullet off of its path. The next aspect that can make a difference on the trajectory of the bullet is the gas system that the firearm uses to operate. Whenever the rifle is fired, the same gases that push that bullet through the barrel also have a small amount that is diverted back through a port in the barrel, which returns to the bolt carrier group (the housing for the firing pin) and then pushes the bolt carrier group backwards. This movement then resets both the trigger and the firing pin, before returning to its original position through help of a spring. This is the system for an automatic or semi-automatic firearm. The alternative is a bolt-action firearm, which is operated through the manual ejection and rechambering of each round shot through the rifle. Because the movement is not done automatically, there is no need for a gas port, which allows for all of the gases to be used toward pushing the bullet through the barrel. This type of action allows for a much greater muzzle velocity, though the rate of fire is significantly decreased.

Although there are configurations that allow for a much flatter trajectory over longer range, we are choosing to use what is more popular in not only the consumer market, but also the military market. Most hunters will not use a set up similar to the one we have chosen (though it is still an option), but the application integration we plan to implement would account for these extra options. The 16-inch barrel is popular because of legal restrictions but also because of military standards, and the gas system allows for more firing options for the user. We are also using the standard 5.56x45mm NATO ammunition because it has a sufficiently long range and a large

popularity in many different sectors of commerce. Although this is the basis for our design, we have a stretch goal of application integration to broaden the usage of the scope to cover multiple different utilities, and even more specific scenarios that could be input by the user.

5.1.4 Design Consideration IV

Extra considerations that will have to take place if the application integration goes as planned will include not only programming changes, but the mechanical aspects of the scope will have to be built to better tolerances. The tolerance of the system refers to the overall machining process and gives a general sense as to how well the scope is built, but how well it will hold up in extreme conditions. A more specific definition of the tolerance of the system is the precision to which parts are built. Whenever two parts are touching, there has to be some sort of a gap that exists so the friction of the parts moving together would not completely hinder assembly. However, it is acceptable to have tight tolerances to the point that assembly would be arduous, as it infers that disassembly would be also as difficult. In our case, a scope is not something that the user would disassemble themselves to service or clean. In most cases, the bare minimum requires that a scope, with the assistance of rubber O-rings, would be able to maintain an air-tight seal so that the nitrogen, or in our case argon, gas would not be able to leak out. If that happens, the scope would be subject to water damage or fogging internally. This is an acceptable tolerance for a scope that is not expected to deal with the force of recoil from larger caliber rounds. However, to respond to the idea of our scope being user-adjustable for many different situations, it would require that the scope can take the extra force and not break its seal.

Another aspect that is directly affected by the tolerance would be the optical system of the scope. The design calls for specific distances between the lenses in the system so that the image remains in focus. If those distances are changed due to the recoil, then the user would no longer have a reliable scope. Especially if the lens that shifted on the axis were an interior lens, such as the magnifying lens or the image reversal assembly, then the user would not be able to attempt fixing it on their own, as it would have lost its vacuum and have been purged with normal air. A design flaw like this would fall onto the manufacturer and is generally covered in most warranties commercially. However, it is much harder for most lenses to be knocked out of place, as the interior of the scope tube generally holds the lenses in by a metal threading similar to a screw, and the individual housings for the lenses would be screwed to the correct distance. These housings are what allow lenses that are narrower than the diameter of the tube to still remain erect in the system. If necessary, a second, empty ring could be screwed into position to ensure that the housing cannot be adjusted any further to one direction. Most scopes commercially available are constructed of a one-piece housing of aluminum that helps to amplify shock absorption and ensure the safety of the optical assembly, but our team does not have the time or access to custom made systems, so the housing will still be

aluminum but coupled from varied tube diameters. With this, however, we do not foresee having and issues with the system retaining its specifications.

5.1.5 Design Consideration V

Continuing on with the application allowing for extra options when it comes to automatic distance correction, one of the main aspects would be with the reticle. Most reticles are designed with a single caliber in mind, and the horizontal tick marks that trickle down the vertical axis are calibrated specifically to that caliber. With our OLED design, we can attempt to set a different reticle for each specification that allows the user to accompany their settings with the proper sight picture, but at the very minimum we will only use a reticle that has the vertical and horizontal axis. As our scope has no means to provide information that would affect the windage, or allow us to predict movement in the x-direction, we only have to concern ourselves with the vertical axis. However, even with the change to use an OLED system instead of using the microLEDs, the only difference would be illumination of a different area that corresponds to the bullet drop of that specific caliber. In the few cases where the distance being targeted would correspond to a bullet drop that causes the OLED to illuminate below the bounds of the reticle, such as .300 Blackout that has a very quick drop-off after about 500 meters, we could program an illumination of set OLEDs that indicate to the user the equivalent of an error message.

Another consideration with the integration of an application that will require a fairly major design adjustment is the new information for the bullet drop coefficient of each different caliber. Testing for new calibers would be arduous and expensive, but also very time consuming, especially as we would have to acquire a different firearm and possibly hundreds of rounds of each different caliber that we plan to test. A solution that would facilitate the entire process would be to partner with an ammunition company that would already have the information for all of their bullets. Using the provided information, we could calculate the information pertaining to each different setting and adjust the programming accordingly. The only drawback is that the information that would be shared by the company will most likely only pertain to a specific set up, such as the muzzle velocity or bullet drop for a bolt action, or twenty-inch barrel, or some other setting that might not correlate to what the user is equipped with. However, this is still a step ahead of starting from scratch. A company that provides such information with all of their ammunition is Hornady, where they indicate the velocity at multiple differences, as well as the bullet rise or fall at a specific distance. They standardize their numbers as well, such as setting the zero value of their long-distance rounds to 200 yards, which would also help to ensure that the bullet drop remains in the bounds of the reticle even at long range.

5.2 Optical Scope

The design for the optical system is built with parameters in place to retain the scope's usability and marketability. Like most zoom scopes that exist on the market, it must be compact enough to fit on a rifle without hindering the user's movement. Also, the eye relief on the scope should not be so far that it is uncomfortable for the user, while it cannot be so close that the firearm's recoil has the capability to strike the user in the face. There will also be an adjustable aspect to the reticle of the scope, which will give it movement in the x-direction, which is referred to as the windage, and the y-direction, which is also known as elevation. This will be accomplished with a screw system that will push the housing of the lens in the specified direction to further align the reticle specifically onto the object of the scope, which would be done before use. This motion is independent of the automatic distance correction that is the main focus of our project. In essence, there should be a specific distance where the bullet drop will be in line with the main crosshair of the reticle. At this distance, the user would have no correction to do for their shot, as the crosshair will fall directly onto the target.

The main optical design will consist of five optical lenses, four of which will be double-convex, and the final remaining lens will be double-concave. With the selection of the lenses, the focal length will be approximately equal on both sides. This symmetrical system will allow for more simplified calculations to design the scope. For the optical axis that we made our calculations on, the object lens was placed at zero, meaning that the object is to the left, placed at infinity.

5.2.1 Objective Lens

The purpose of the objective lens is to gather light from the object, and a larger diameter lens allows for more light to be let in, as well provide a wider field of view. The diameter that we chose for the objective lens was 30 millimeters, so that the field of view is large enough that the user would not have issues acquiring a target at long distances, but not so large of a diameter that the lens would require accommodations to fit onto the rifle, such as a higher mount. The focal length of the objective lens will be 50 millimeters, which also ensures that an object will not be so close in most cases that it has not yet achieved focus. The focal lengths of the lenses were kept in mind when choosing options, as they can easily add considerable distance to the overall length of the system. The next lens in the system serves as the image reversal lens.

The goal of this lens is to reorient the image so that it is not inverted by the time it reaches the pupil of the user. The main design constraint we had to consider for this section was that an image reversal assembly requires that the light rays are able to travel twice the distance of the focal length of the lens before reaching the actual lens. This just meant that we had to choose a lens with a sufficient diameter that light could still travel and illuminate the image, but with a short enough focal length that the system not extend too far because of it. The lens that we chose has

a diameter of 25 millimeters, and a focal length of 25 millimeters. This means that this lens is placed at 80 millimeters along the optical axis, as the focal length of the objective lens requires 30 millimeters of distance, and the focal lens requires 50 millimeters to satisfy the 2F focal length requirement.

5 .2.2 Main Eyepiece

For the main eyepiece of the scope, we chose a three-lens system that incorporates two double convex lenses, one on either side, with a double concave lens in the center. The first lens is placed at 130 millimeters with respect to the objective lens and is spaced at the appropriate distance of two focal lengths from the image reversal lens. The lens we chose has a focal length of 50 millimeters and is only 5 millimeters wide at its center. This lens is followed by the double concave lens, which has a focal distance of -100 millimeters. In our calculations, we found that this lens has the greatest impact on the magnification of the system. If we were to make this have a smaller focal distance, for example one of the options we considered being only -50 millimeters, the magnification would jump to above 20. Although this can be beneficial in some scenarios, for our purpose we do not need to have an excessively large magnification, as the 1 to 6 times will suffice. This lens does not have a set location, as the variation of its position is what allows for us to have a variable magnification. With our current calculations, the double concave lens being placed at 133 millimeters with respect to the objective lens, which is only 3 millimeters away from the first lens in the eyepiece, we achieve a 1 times magnification. If our double concave lens is placed at 298 millimeters with respect to the objective lens, we achieve a 6 times magnification. All other magnifications between these two points will fall into this range of being between 1- and 6-times magnification. The final lens that will complete the eyepiece is another double convex lens that is placed at 300 millimeters with respect to the objective lens, this is only 2 millimeters away from the double concave lens when it is placed in the 6 times magnification location. The lens will have a focal length of 60 millimeters, which will also act as the focal distance for the eye relief of the user. This means that the user will be placing their eye at 60 millimeters away from the lens in the image plane, or 360 millimeters from the objective lens.

The overall system that we chose will allow for the desired magnification while still being fairly compact. The whole system falls in at a total length of about 300 millimeters, which is just under a foot in standard units. The largest distance that we use is the distance between the first eyepiece and the focal lens, as there is required to have a sufficient variable distance for the magnifying lens. This lens needs the full 170 millimeters off movement to achieve the entire range of magnification, as the concave and convex lenses will be almost touching on either side of the eyepiece. With a optic system, the focusing of the image is reliant on the lenses being the proper distance apart from each other, which is generally just the addition of their focal lengths. However, that is not the case exactly with a zoom magnification system. The focal lengths of the lenses can help to increase or decrease the effective

focal length of the system as a whole, but the range of magnifications that can be achieved relies on the distances between the lenses. Although we are still learning some of the specifics of this type of system, we found that increasing the magnification of the zoom system, which could be easily accomplished by altering the focal length of the double concave lens, the variable magnification still changes from a set range, not a proportional value of the magnifications. For example, the range we achieved was from 1 to 6 times, but if we altered the focal length to have a lower bound of 2 times magnification, the upper bound was not 12 times, but closer to 8 times. This realization during the trial and error of our calculations is what aided us in reaching our final values, as the focal length and distances between lenses had to both align to produce the desired zoom configuration. This overall realization, however, only applies to the focal lengths and the distances of the eyepiece lenses, as we did not vary the focal lengths of the image reversal assembly or the objective lenses. We believe that changing those lenses would have a proportional change on the magnification of the system, such as changing the 1 to 6 times magnification to a 3 to 18 times system. Although this is not our desired configuration, it was a learning moment if we ever decide to pursue different variations of this design. This realization also helped us to understand other products on the market, and how some products can have a shorter overall system while achieving a higher magnification, as their range is proportional to a smaller variation of magnification, such as an 8 to 24 times magnification only requiring the distance of the magnifying lens to triple the effective focal length of the system, where ours requires the magnitude to increase sixfold.

5.2.3 Main Construction

For the main construction of the scope, we will use an aluminum housing that consists of a 30-millimeter tube for the majority of the length, with a 40-millimeter section that houses the focal lens assembly and the magnifying lens. There will also be a slightly larger section that breaks the main tube into two pieces, this is where the adjustment turrets for the reticle will fall into place. The reason for a 30-millimeter tube is that it will allow for more room for adjustment of the reticle, but it also allows the Arduino board to sit underneath the unit and not have any section that reaches further out than the tube. This will help to ensure that if the unit were to drop, the electronics would have an added layer of protection.

For the 40-millimeter section of the assembly, there will be two more adjustments, one for the magnifying lens that allows the scope to have a variable zoom by changing the distance between the lenses, and one that allows the focal lens to translate laterally along the scope, so that the picture can have more clarity when the eye is within the range for proper eye relief. The rangefinder will be placed on the left side of the housing in the center of the main tube, as the top side will be populated by the turret that adjusts for elevation (vertical adjustment) and the right side will house the turret that adjusts for windage (horizontal adjustment). There will also be an input slot next to the Arduino board that allows for a toggle switch to be installed.

This switch will control the trigger for the rangefinder.

In a stretch rendition of the product, we can have a built-in mounting system so that there is more room for the electronic components. For the time being, however, we will use high-mount 30-millimeter rings that are constructed out of steel, or titanium if cost permitting. If we can get the slot machined properly and the housing for the battery sit inside the main turret, we will attempt to do so. Alternatively, if we must, we will put the battery underneath the unit, next to the Arduino.

To help ensure that this system can withstand the force from the recoil of the spent cartridge, there will be rubber gaskets that sit on either side of the lens for shock absorption. The main reason for these is so that there is no room for the lenses to move, which allows for more precise movements between though. These extra points also soak up recoil from the original. Finally, there will an input for the toggle switch that tells the rangefinder when to start. This will allow for the user to choose where the button will be placed according to personal preference.

Model	Focal Length	Diameter	Position (WRT Objective Lens)	Price
63-556	30mm	30mm	0mm (Objective Lens)	\$35.50
32-489	25mm	25mm	80mm (Image Reversal)	\$29.00
32-624	50mm	25mm	130mm (Eyepiece)	\$29.00
32-994	-100mm	25mm	Variable: 130mm-300mm (Magnifying lens)	\$29.50
45-167	60mm	40mm	300mm (Focal Lens)	\$39.50

Table 12: Optical Lenses Comparison

5.3 Optical Housing

Construction of the scope with concern to the housing elements will be just as important as the optical elements overall, as it is what allows the elements to remain securely in their position along the optical axis. The main tube will be comprised of aluminum, with a tube that is smooth on the outside but threaded on the inside. Most optics companies that design scopes use a single piece of aluminum that is custom molded to fit the specifications of the scope, but in our case, we will only be able to use separate pieces that are coupled together to achieve our system. This is a constraint due to lack of time or resources that these companies possess, however we expect more of just an aesthetic setback than a mechanical one. On

the interior of the tube, the lenses will be held in place by individual housing units which are threaded rings that individually hold in the lenses in the tube. These rings are generally black anodized aluminum, so that they do not reflect stray light, and they remain lightweight yet durable, so the scope has the ability to retain its specifications while still being lightweight. The main pitfall that befalls us when looking for the elements is that due to us not being able to have every part custom-made, we will have to make specific accommodations so that everything can fit. Many of the threaded lens holders than are offered by Edmund Optics, the same company we will be purchasing our lenses from, only go up to 1 inch, or 25.4 millimeters, in diameter. This means that we will have to ensure that whatever company we get the lens holder for the objective lens, which is 30 millimeters, will provide the same thread pattern as the M29 x 0.75 that most of Edmund's elements use.

The main aspect that we will have to get made custom will be the magnification section of the eyepiece. Most models in the market utilize a rotational stage that translates to a translation stage. This means that the rotation of the tube caused a z-direction movement of the lens. The upper and lower bounds for this will be the distances where 1 time and 6 times magnification are achieved. The reason we want to retain this standard that almost all other scopes on the market are using is because it requires more force to adjust the magnification. This force applies to both the force exerted by the user on the rotating stage as well as the recoil force from discharging the firearm. If we used a simple axial translation stage where the user is required to push or pull the stage to adjust the magnification, it could be more susceptible to unwanted adjustments from knocking the scope accidentally, or from increased recoil being applied.

With our scope having an OLED screen as our reticle, we are not necessarily required to use the same turret adjustment system as most scopes on the market. The general convention is that the main reticle, which is a non-magnified lens that falls at the distance of two focal lengths before the image reversal assembly, and one focal length after the objective lens, that has an engraved design of the crosshair that will be used aiming. This falls in a position called the first focal plane and it allows the scope to magnify the reticle as well as the image. This particular lens would have adjustments in both the x-direction and y-direction, called the windage and elevation, respectively, and is achieved by using a rotating knob that pushes a rod into the side of the lens, which has a spring on each axis to return the distance when the knob is rotated the other direction. The knob would not rotate uniformly like a standard screw, but rather would be incremental for distances corresponding to Minute of Angle (MOA) measurements that were covered in this paper. This allows the user to accurately judge the difference that their adjustment will be making. Also, if a shooter wants to adjust the reticle to match the location of their target as opposed to accommodating the crosshair based on external factors, it creates an easier environment in which to do so. For example, if the bullet will be pushed 5 inches at 500 yards due to force by the wind, and a 0.5 MOA adjustment is being used on the windage turret, a clockwise rotation of 2 clicks would center the target on the crosshair. 1 MOA is 1 inch at 100 yards, so that equals to 5 inches of

movement at 500 yards. This is not the convention that we will be recommending with our design, as the reticle itself will illuminate where the user should accommodate for aim. Also, as we are using a transparent OLED screen as opposed to a plain glass reticle, there would not necessarily be a required mechanical adjustment system. With a reticle that is essentially a single illuminated OLED on a screen, we could use a button system that allows the user to move the center of the reticle over by a single LED at a time. This could be accomplished by either a button system on the top and sides of the scope, or even integrate the idea implemented by the Meprolight Foresight and allows the user control with an application connected via Bluetooth that would adjust the reticle. With the implementation of a reticle that automatically corrects for distance, a zero adjustment would not require resetting on multiple occasions, the user would have the optic zeroed for a specific distance and the programming would accommodate accordingly.

The final adjustment feature that we will aim to achieve as a stretch feature would be a final focal adjustment that can be achieved by an axial translation of the first eyepiece. In similar products on the market, there are one or two more adjustment options to allow the user to accommodate for any sort of aberrations. The first one that we will attempt to achieve if time permits, is focal adjustment to allow for a more crisp sight picture at any distance. One issue with riflescopes that can exist is a lack of parallax adjustment, which is a phenomenon that causes the scope to fall out of focus for certain distances. This does not completely blur the sight picture that the user will be seeing, but it can cause a loss in resolution that would affect smaller details. When using a scope at its highest magnification settings, the small details do get exaggerated and could potentially cause misidentification, which would be a huge issue when something is on the other end of a firearm. This parallax adjustment would be an axial translation in at least one of two places. This first would be in the area of the reticle, which would be the first focal plane location for our design. The second would be in the focal lens of the eyepiece, the lens that is closest to the user's eye.

The adjustment for the reticle would require that the entire housing for the reticle, which includes the windage and elevation adjustments, to have the ability to be adjusted as a whole, as only moving the reticle would cause the rod and spring system discussed earlier to fall out of alignment. However, this would be easier to achieve with the digital system that we plan on implementing, as the adjustments are applied directly to the screen instead of lateral displacement of the reticle. As for the axial adjustment of the focal lens, that requires an additional lens be placed at the very front of the system. This would not affect eye relief as much however, as the image would be shifted in such small increments that it would still fall within the focal length of the human retina, and it would just allow for a more precise recombination of the optical rays on the focal point. The axial movement that takes place with the reticle would be achieved with essentially a third turret towards the center of the main tube of the scope, and rotation would cause either a forward or backwards motion. These turrets are usually placed on the left side of the scope and include an adjustment dial that denotes the distance that the object would be

placed at. The adjustment at the eyepiece would be very similar to how the adjustment for the magnifying lens would occur, however it looks far more similar to if you were to just unscrew the lens from the main tube. Forward or backward adjustments would be accomplished by either clockwise or counterclockwise rotations of the assembly, and you can visually see the lens moving to accommodate for the change.

The final product used a 3D printed PLA housing, essential a tough and durable plastic that was formed into place. The pieces were all designed using TinkerCAD online and were printed using a 3D printer. There were 3 main sections that composed the entire scope, the eyepiece, the reticle housing, and the electronics housing. The eyepiece ended up being about 180 millimeters long, and 50 millimeters wide throughout the entire section. At its thickest, the housing was about 10 millimeters wide, which is what provided the rigidity that we were hoping out of an aluminum housing. The magnifying lens was placed inside of a housing ring that was just slightly thinner than the interior diameter of the eyepiece tube, which, using magnets, allowed it to move back and forth. However, there was friction that could not be overcome, so it was not the most efficient system. The eyepiece lenses were placed on either end of the tube, with a 5-millimeter section left empty to allow the reticle housing to connect for easier assembly. The reticle housing was much shorter at only about 80 millimeters, with a thickness of 40 millimeters, except for a 45-millimeter section that connected to the eyepiece assembly. This housed the image reversal lens and the objective lens and clamped around the OLED display that was used as a reticle. It was broken into two sections at the location of the display, and screws were used to clamp the screen into place. This hole was to ensure that there was enough room for the reticle without having to force it into a cut out area, as the glass is fragile, and we did not want to risk breaking it. The final section is the electronics housing, which had a 40 millimeter by 80-millimeter section for the PCB and cut outs for the wires and the USB-C input. The underside had an indentation that allowed the section to fit snugly around a Picatinny rail mount that we ended up using to connect the entire unit to a rail section on an AR-15 handguard. The optical sections of the scope were connected to a lid that went on top of the electrical housing by a screw through a thicker section of the scope housing. The lid fit into place by aligning with posts to the bottom sections, but zip-ties were also used to ensure that nothing would move. The PLA housing was very sturdy for the material being used, and had an overall cost of only about \$8, though the printing process is very slow, and it took about 30 hours for all components to be printed. The main reason why we decided to use the 3D printing was because the machining process would have taken too long from the point when we attempted to make an order.

5.4 Targeting Lens

Designing a targeting lens at first glance seems like a rather daunting task. The lens is one of the most important features of this project. It is responsible for telling

the user where to aim and being accurate all while staying out of the users way. Due to these requirements, the design for the lens comes with a high requirements list.

- Must be mostly transparent and cannot obstruct the view.
- Should be able to quickly update based on new information.
- Needs to have enough detail to relay an accurate measurement back to the user.
- Has to be compact enough to fit within the scope housing.
- Should be stable and without moving pieces so its accuracy is not thrown off by recoil.

The short list above specifies the general requirements that will need to meet to develop a proper targeting lens. Several ideas come to mind when thinking about how this type of lens should be designed.

One of the first ideas was to have a laser mounted within the housing of the scope that is controlled along its vertical axis with a motor. The laser would then shine a beam upon a lens that would indicate to the user where they should aim. There were a few problems with this idea though. The first problem would be that the laser would be in the sight of the user which would be distracting. Another problem would be calibrating the laser. Due to the recoil of the rifle, the laser could shift and beam on the lens would not longer accurately represent where the user should aim. Another issue would be moving parts, the more moving parts the more prone to breaking the design is. Because of these reasons we decided not to peruse a design like this any further.

Our next idea was to design a lens which consisted of a glass or plastic lens with a high-density array of microLEDs along the vertical axis of the lens. This was a design that was semi-feasible, but we ended up not perusing due to the cost. While the array of microLEDs would have been cheap enough, we do not have the proper tools to affix them to a lens or wire them up. Another problem with this design would be that it limits the information we are able to provide to the user. If we wanted to display the battery charge or the distance to the target, we would not be able to in an easy to read fashion. We would have to somehow use the array of microLEDs along to vertical axis to display this information. This is not ideal, because of this and the difficulty in manufacturing a lens like this, we decided to keep looking.

Finally we stumbled upon something that would work! While searching for microLEDs for the previous targeting lens idea, we found a transparent OLED display that just happened to be small enough to fit without our scope housing. Due to the nature of it being a display, it has a high-density of OLEDs allowing us to accurately display any necessary information on the screen. Not only will it allow us to tell the user where to aim, we can also display other information. Some information we might want to display would be the remaining charge of the battery, an approxi-

mate amount of time left before the scope battery dies, the distance to the target, an accuracy rating indicating how sure the scope is of its accuracy, and much more information. This also solves the problem of moving parts, the OLED display is a solid-state module that as long as it is mounted secular, should not shift from the recoil of the rifle.

5.4.1 OLED Display

Our lens design uses a display created by a company by the name of Crystalfontz. This company produces many different displays intended to be integrated into other products. One display they produce is a transparent OLED display. This is a small display, that has a width of 42.04mm and a height of 27.22mm. The objective lens on the scope will be 24mm wide and the reticle will be around 20mm wide, meaning this screen is more than large enough to cover the entire lens. The display supports both I2c and SPI interfaces, meaning it will be easy to interface with our microcontroller.

The active display portion of the OLED module is 35.05mm by 15.32mm. To make use of the most amount of pixels provided by this display, we will want to mount it vertically. The display is 128x56 pixels, meaning if we mount it vertically, we will have a resolution of 128 pixels to work with along the y-axis of the lens. Since we will not be using the x-axis, this is preferable. Realistically, almost all the targeting lens points will be on the bottom half of the lens since the bullet will drop, rather than ascend.

The transparency of this display is one of the reasons it is not as good of an option as one of our previous design ideas. When an OLED pixel is turned on, it is an opaque light-blue color. This is fine, but when a pixel is turned off it is only about 70% transparent. This means the target you are looking out will be less visible than if you were to not have a targeting lens at all. It is highly unlikely that this will cause much of a problem, but it is something that should be noted.

5.4.2 Display Board Circuit

The display has a 24 position flexible-flat cable attached to it that can be used to interface and run the display. With this, we can build a circuit that is capable of running the display. We will use this cable with our display board to have it interface with our microcontroller. To control the display, we must interface with the built-in Solomon Systech SSD1309 controller.

To build this circuit we will use the data sheets provided by Crystalfontz to correctly interface with the display and power it. We will be using a software called EasyEDA to design both the circuit and the PCB. For this design we must take the interface provided by the OLED display and translate that interface into something that our microcontroller can interact with.

We are lucky because the display provides several different interfaces that we can interface with. Our microcontroller also has several different interfaces, so compatibility isn't much of an issue. Out of the different options, we decided to use I2C because our laser rangefinder was already using the ADC interface and I2C is a simple interface that we are familiar with. By default the I2C interface is not enabled for the display, so we will need to enable it using our circuit. We ended up achieving this by using a "tie low" resistor on several different pins. With this all we need to do is wire up the clock, data, and power pins from the microcontroller.

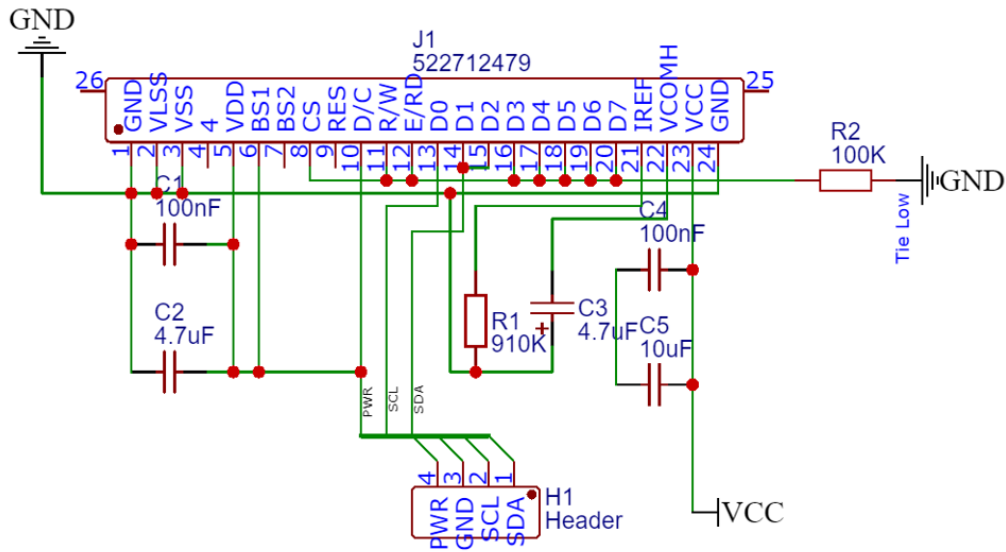


Figure 7: Interface board schematic between display and microcontroller

Just above in figure 7 is the schematic for the board that will facilitate communication between the microcontroller and the OLED display. The schematic contains a 24 pin surface mount Molex connector that the flexible-flat cable from the OLED display can connect to. Because our microcontroller supports I2C, the board has a header on it that takes an I2C connection which includes, power, ground, the clock signal, and the data signal.

5.4.3 Interfacing with the OLED Display

Interfacing the IC2 signal to the OLED display is relatively simple. There are 24 different pins on that are available on the display, but only two are used to actually display images on the screen. Below is a list of the pins and their purposes.

- **GND** is the ground pin.

- **VLSS** is the of the analog circuit.
- **VSS** is the ground of the logic circuit.
- **VDD** supplies power to the logic circuit. This is powered by the I2C signal from the microcontroller.
- **BS1/BS2** sets the communication protocol. We set BS1 high and BS2 low for I2C mode.
- **CS** enables MCU communication. Set to low since we are using the MCU to select I2C
- **D/C** Data/Command control pin. Tied low to act as SA0 in I2C mode for slave address selection.
- **R/W** Read/Write control pin. Tied low to enable write mode.
- **E/RD** Read/Write control pin. Tied low to enable write mode.
- **D0** Line for I2C clock when I2C mode is enabled.
- **D1/D2** Acts as SDA Out and SDA In when I2C mode is enabled.
- **D3 D7** Unused in I2C mode.
- **VCC** Supplies power to OEL panel. This is supplied by the external power supply.

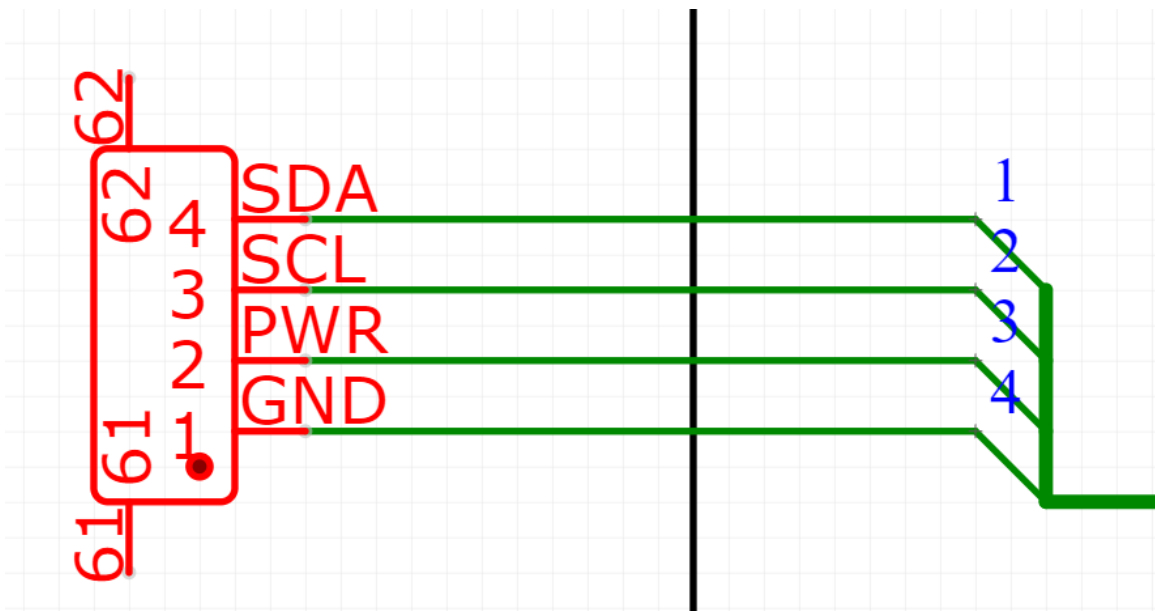


Figure 8: I2C Bus

In our design we have both an external power supply and a microcontroller that is able to produce power as well. For the logic circuit of the display we are going to

use the power provided by the microcontroller as it will provide what we need via the I2C input. For the display we need around 12.5v which the microcontroller. We could of course design a circuit that would boost the input from the microcontroller to 12.5v, but to avoid complicating this circuit we will relegate that to be handled by the power supply instead.

Our microcontroller is the Portenta H7 from Arduino. This board provides a high density connector with support for multiple interfaces. Among these connectors on the board, there is a ground pin and a 3.3v output. There are two pins that we are mainly interested in which are the data and clock pins. The I2C pins on the board are pins D11 for the serial data line and D12 for the serial clock line. These two pins are all we need to transmit what we want to display on the screen. The 3.3v output on the Portenta H7 will be used to power the logic circuit and the ground on the Portenta H7 will be used as a common ground for the display and the board used to interface the display and microcontroller.

The SSD1309 driver defines a set of commands for interacting with the display. This driver allows us to draw images to the display, adjust the intensity of the display, and many other things with a simple set of commands that can be easily programmed. In table 13 there is a list of the commands and what they do. These commands will be the most relevant ones to the targeting lens.

Hexadecimal	Description
0xFD	Sets command lock
0xAE/0xAF	Set OLED Display On/Off
0xD5	Set oscillator frequency
0xA8	Sets multiplex ratio
0xD3	Sets display offset
0xA0/0xA1	Sets Segment/Column Mapping
0xC0/0xC8	Sets row scan direction

Table 13: Subset of SSD1309 Controller Commands

While this is not an exhaustive list of all the commands these are the primary commands that will be used to interact with the display for our purposes. The interactions with the display will be expanded up on the software section of this document.

5.5 Laser Rangefinder

In this first part of this section we will go over how we might design a laser rangefinder for our project. Due to the constraints in finding parts we will not be able to execute this design ourselves and will have to resort to using a pre-built laser rangefinder that fits our needs instead.

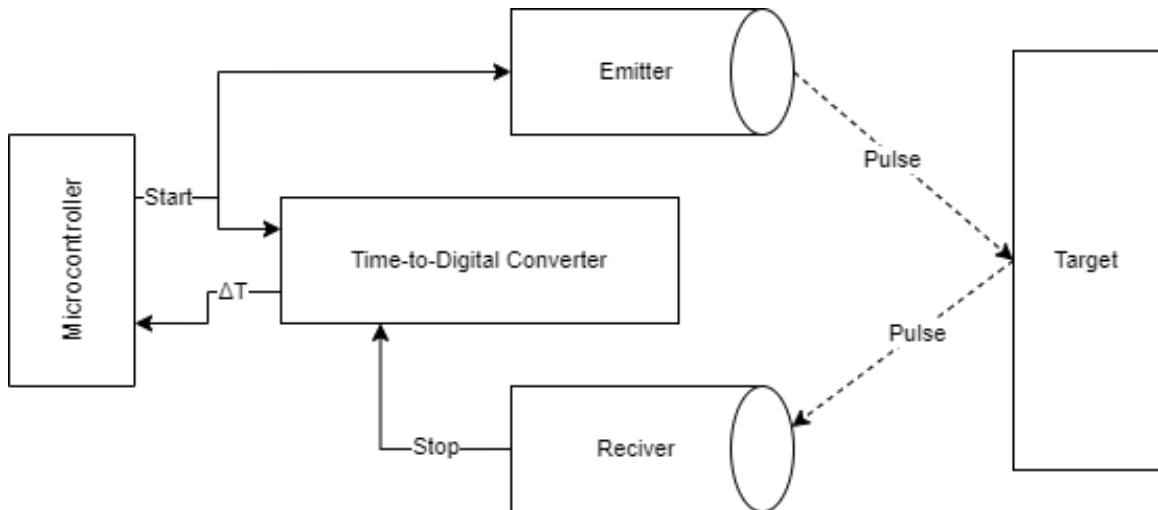


Figure 9: Rangefinder Flow Diagram

To design the rangefinder, we first needed to determine our constraints. The finder must be able to measure a large enough distance that the scope will be useful. According to our constraints listed previously we have decided to aim for a rangefinder with a range between 500 Meters and 1000 Meters. Since we are measuring a larger distance we are able to be less precise with our measurements than if we were measuring a very small distance. The design of our scope also limits how accurate we must be when measuring. Due to the amount of microLEDs we can fit on the lense, our range finder must only be accurate up to the point LEDs can indicate within the scope. For example if our LEDs can online measure in 20 meter intervales, then our rangefinder should be able to have a 10 meter error.

To measure the distance the rangefinder needs three essential components, the emitter, the receiver, and a timing circuit to measure the time of flight. The rangefinder works by first sending out a pulse of photons. As soon as the emitter is activated the timing circuit will start. The receiver will then wait to receive the photons. Once the receiver registers the photons emitted by the emitter, it can tell the timing circuit to stop. The timing circuit will now have to total amount of time it took for the photons to fly from the emitter, to the target, and back to the receiver. We can then calculate distance from the rangefinder to the target by multiplying the time of flight by the speed of light and then dividing by two. This should give us a relatively accurate measure of distance which will be essential in our scope design. [13]

5.5.1 Calculating time-of-flight

Getting the timing correct is a crucial aspect of the design. We cannot rely on timing the emitter and receiver just using the microcontroller, so we are going to need to use a special circuit designed for these exact situations. The circuit we are going to use is called a time-to-digital converter, or a TDC for short. This circuit will allow us to measure the time interval between two different events. In our case we will be measuring the difference between the time when the emitter is triggered and when the receiver is triggered. We can then use these as inputs for the time-to-digital converter to produce a delta time. The time-to-digital converter has two different inputs and a single output. The first input tells the circuit to start counting, the second tells it to stop. The output gives us our delta time that we then hook into our microcontroller and use in various calculations.

5.5.2 Communicating with Microcontroller

Our rangefinder will need to be able to interface with our microcontroller. Our microcontroller is able to support a few different types of communication interfaces, including ADC, I2C, and UART. We have already discussed previously the various different types of communication interfaces and why we chose to go with UART. So, we are going to need to build a circuit that can take the output from our time-to-digital converter and translate it into a UART signal that the microcontroller can use.

5.5.3 Implementing the rangefinder

The rangefinder that we decided to use for the project is the HIREED model HR600 rangefinder. This rangefinder is able to measure a distance of up to 1.5km. It uses the time-of-flight to calculate the distance as mentioned earlier. The rangefinder is accurate enough for the needs of this project with an accuracy of $\pm 1\text{m}$. Below in figure 10 is a top-down view of the rangefinder and its connections.

Connections

- **Red Line** - Power from 7V to 18V
- **Blue Line** - Ground
- **Yellow Line** - RXD, the data input line
- **Green Line** - TXD, the data output line



Figure 10: Laser Rangefinder

To power the rangefinder we will supply it with 12V at 200MA via our power supply. The red line is for power, the blue line is ground, the yellow line is receive data, and the green line is to transmit data. Since the rangefinder communications using the UART standard, we are able to easily connect it to our micro controller without the need for any sort of breakout board.

When communicating with the rangefinder via the micro controller there are several important commands that can be used to turn on, turn of, and start measurements. The following table displays the Hexadecimal value for each command followed by what the command does. These commands will be hard-coded into a JSON file so that we may pull whatever command whenever we need to, and since it is JSON format we may include descriptions into each JSON object for ease of use. This JSON file will be included into our micro-controller file structure with the title "Range Finder".

Hexadecimal	Command
0x4F	Turns on the laser module
0x4F	Turns off the laser module
0x4D	Measures the distance to the target. Slow but accurate.
0x46	Measures the distance to the target. Fast by may be inaccurate.

Table 14: Laser Rangefinder Commands

The laser rangefinder module provides a couple different modes for measuring. The first mode is "Fast Mode". This mode will quickly measure the distance but it may not end up being very accurate. Depending on the lighting and the surface of the target this may give results that are not acceptable. The second mode is "Slow Mode", this mode will not measure as fast as the other mode, but the measurements are more accurate. This is useful when the lighting is poor or the target is not very reflective.

Considering the use-case of our project, we will likely want to prefer using "Slow Mode" as we do not control what the target of the scope is. It would be better to give accurate measurements rather than quick especially for safety reasons.

5.6 Set & Reset Switch

When the user of the product is setting their target, the experience should be seamless. They should be able to set and reset the target without having to move their eye away from the scope. To do this the design will implement a switch button near the trigger of the weapon. The switch should be easily reachable by the trigger finger. We do not want the user to be able to push the button while their finger is on the trigger as this may cause them to accidentally fire the weapon. To do this it should be on the face side of the trigger guard. In figure 11 the switch is placed on the side of the trigger guard in front of the trigger. This will make it easy for the user to operate while also forcing the user to remove their finger from the trigger before using. This design is convenient yet sleek which adds to our design aesthetic and allows our project to have less of an impact on the gun, since most people do not prefer to have bulky items on their weapons we should be able to incorporate this with ease, and since the button is on the side of the we open we can make it as large or as small as it needs to be depending on the weapon the same button may not be as convenient for both a pistol and an assault rifle.

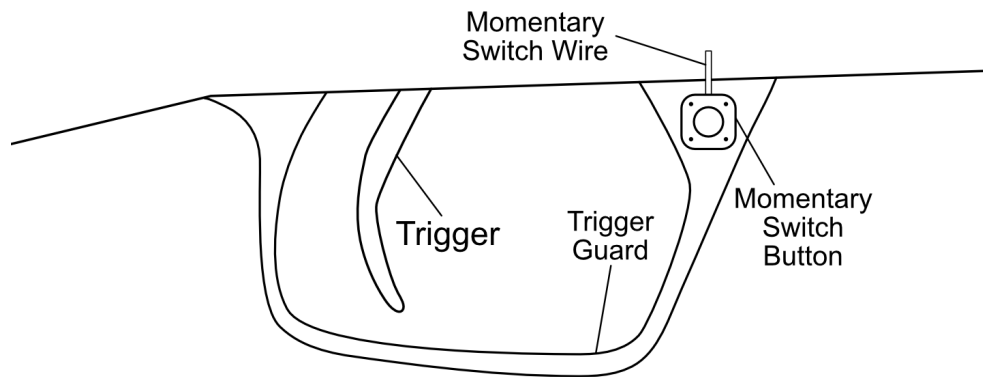


Figure 11: Set/Reset Switch mounted on Weapon

To attach the switch button to the trigger guard of the weapon, it should have a simple tape-like adhesive applied to the back. The button should be able to be affixed and removed without having to reapply the adhesive. On the top of the switch there should be two wires that run from the switch to the micro controller.

The first wire running from the switch should be connected to the pull-up resistor on the micro controller while the second wire should be connected to one of the available data pins. When the button is pressed the micro controller should be reading the state and react to it. If the target on the scope has already been set previously, then the targeting should be reset. If the target is not set, then the micro controller should initiate the targeting sequence. The switch should act like a toggle button, inverting the current state whenever it is pressed.

Our micro controller already has several pins in use by both the laser rangefinder and the targeting lens. This means our options are limited when choosing what pin we can connect our switch to. The following data pins are being used by other modules:

- **D13 / D14** - These pins are being used by the laser rangefinder for communication.
- **D12 / D11** - These pins are being used by the targeting lens to display graphics on the OLED display.

This leaves us with pins D0 through D10. For simplicity sake, we can just use the D0 pin on the micro controller to handle the signal from the switch. When the switch is closed, it will complete the signal between the pull-up resistor and the data pin. The pull-up resistor is used to give a known state of "1" to the input pin. Allowing us to detect via our micro controller when the switch has been closed.

5.7 Power Supply

Our Smart Compact Scope will contain one mobile power supply under the scope. The Portenta H7 (MCU) supports the connection of 5V directly to Vin, input pin on the MCU, to power on the microcontroller, and activate the power pins built on the microcontroller which are +5V and +3V3. The battery that we will be using is the Tenenergy Flat NiMH Battery, 9.6V, High Capacity 8-Cell, 2000 mAh, to power on the MCU and the other components needed for this system. We will be using this single battery to power on the MCU by stepping down the voltage to 5V and also powering on the OLED display and the rangefinder by stepping up to 12V. To be able to step down the 9.6V battery to 5V we will be using the LM7805 and to step up to 12V we will use TPS55340RTER. Also, this MCU supports two other methods to input power which are through a USB-C 5V power source or by connecting a single-cell 3.7 V Lithium-Ion/Lithium-Polymer battery via a 3-pin connector. This 9.6V battery would help us in our goal to keep the design compact under the scope, and as an additional perk, we will be able to supply all of the components in the system with the necessary nominal voltage with just this battery. [4]

It is important to choose this 9.6V battery to be able to reach the voltage threshold to power on the MCU and satisfy its circuit operating voltage of 3.3 V as well as the two other necessary components in the system that require 3.3 V, the OLED logic circuit and tactile button both require. These components will be powered on by the +3V3 output power pin built-in on the MCU that will have power sourced from the 9.6V battery that is powering on the MCU. Due to using this 9.6V to step up and step down to the corresponding voltage level that we need for the corresponding components, we won't be needing a second power supply. More details about this battery will be discussed below.

5.7.1 5V Voltage Regulator

In this section, we will be going over the design and procedure that the 9.6V battery will take to power on the microcontroller by stepping down from 9.6V to 5V. After stepping down the 9.6V input to 5V, it will then be able to be connected to the pinout on the PCB. This 5V pinout will be used to power the microcontroller which takes a 5V input. This is also necessary to power the logic circuit of the targeting lens as it will use a 3V output from the microcontroller. The microcontroller will also be responsible for providing power that the tactile button will use to indicate when the larger rangefinder should be activated.

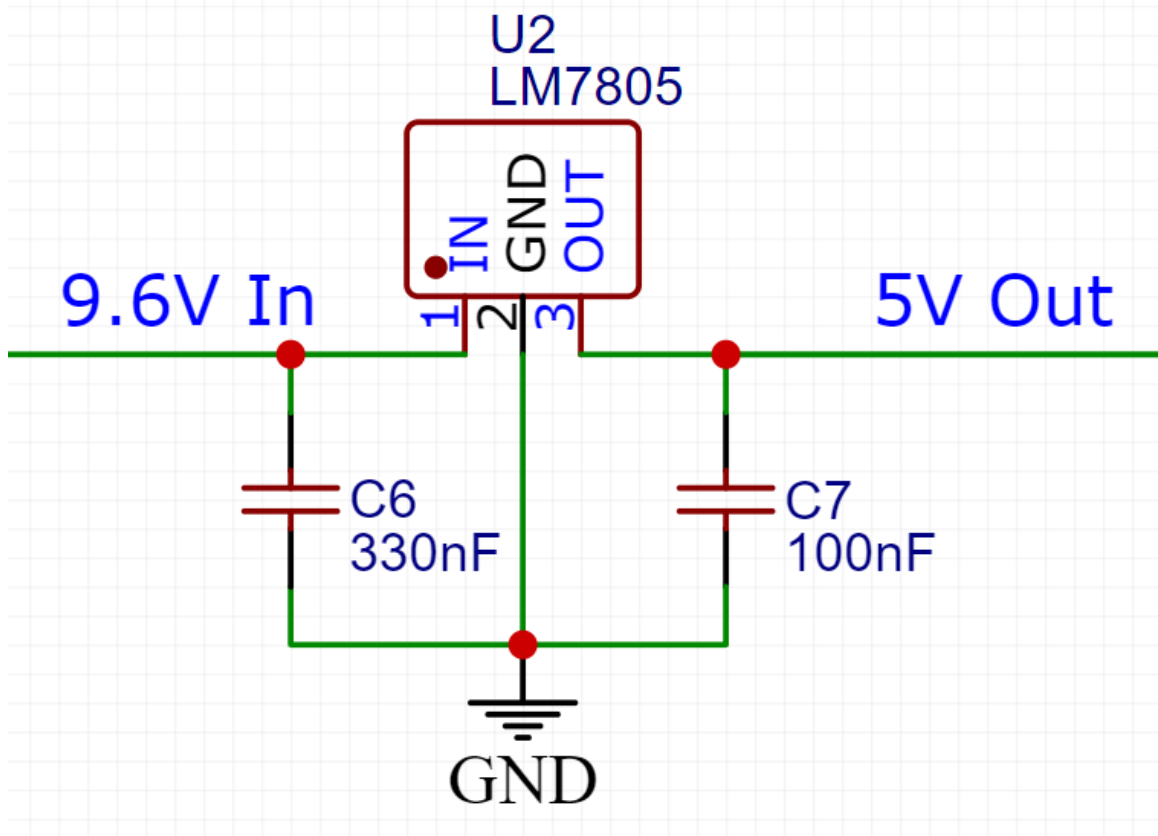


Figure 12: 5V Voltage Regulator

As you can see above in Figure 12, we are going to implement the voltage regulator LM7805 to get the initial 9.6V from the battery source and drop it to 5V so we can utilize this voltage for our necessary components as listed on Table 17 below. This linear voltage regulator is a through-hole mounting style and it features three pins that are the input, output, and ground. In Table 15 below are the specifications of this voltage regulator which are going to be crucial to make our power system work correctly. [14]

Symbol	Description	Min	Typ	Max
VinMax	Maximum input voltage	-	-	35V
Vin	Recommended operating input voltage	7V	-	25V
Vout	Output voltage	-	5V	-
Iout	Output current	-	1.5A	-
Tj	Operating junction temperature	0°C	-	125°C

Table 15: LM7805 Specifications [14]

We chose to power the MCU via the 5V input pin directly instead of the USB-C 5V connection for one important aspect: to save space. By connecting the 9.6V battery, and stepping down to 5V, with just two wires directly to the Vin, input pin, and GND we save space under the constrained space below the scope. With the USB-C 5V connection, we would need a cable adapter that will connect from the battery source which would be a USB-A to the MCU via USB-C. This cable can normally be inflexible and bulkier than just the set of thin strand wires that can be connected directly to Vin and GND. Choosing the 9.6V battery will help with the constrained space below the scope as we need to consider the size of the battery and if it will comfortably fit below the scope. The size of the battery in mm is 103.8x58.9x16.0 (4.09"x2.32"x0.63"). This size will be able to fit below the scope with the other components below the scope such as the OLED logic circuit and the OLED driving panel.

According to the datasheet of the MCU, the recommended operating conditions for the 5V input pin is to have a voltage between 4.5 and 5.5V. The battery we will be using for this pin will be stepping down to 5V, so it meets the recommended criteria. Moreover, the input voltage supplied through the 5V input pin will be unregulated. Meaning that the output is likely to fluctuate from the initially anticipated 5V from the battery. Moreover, the voltage supplied through the USB-C 5V connection will also be unregulated, so there's not an advantage with choosing it. We plan to fix this issue of noise and ripples in the input DC voltage with two methods: twisting the cables together and keeping cables short. We will twist the red (+) and black (-) wires together that come from the stepped down 5V and unwind them right before connecting it to the MCU to Vin, input pin, and GND. Twisting the cables helps minimize the effects of noise on the input voltage and it helps minimize the voltage drop when more current is drawn from the stepped down 5V. Also, keeping the cables short from the battery to the MCU will help us further prevent any significant voltage drops. Taking these measures to prevent noise and ripple will help to prevent any drastic changes in the voltage input and current load. Any drastic changes could cause components connected to the stepped down 5V to be damaged permanently if it exceeds from the recommended voltage criteria. [8]

Moreover, something important to note is that because of the absolute maximum and minimum that the Vin can hold, it would be very unlikely to damage the MCU if the input voltage changes drastically. As you can see in Table 16 below, the absolutes of the Vin are wide enough to give us more room for the input voltage to fluctuate. It shows that the fluctuations in the input DC voltage from the expected 5V battery won't damage the MCU and will still be enough to power the 3.3 V of circuit operating voltage.

Symbol	Description	Min	Max
Vin(Max)	Maximum input voltage from Vin pad	-0.3 V	21 V
P(Max)	Maximum power consumption	-	5000 mW

Table 16: Absolute Maximum and Minimum of Vin [4]

Even though it would be unlikely to damage the MCU we are still taking precautions to prevent further noise and ripple in the input voltage as the stepped down 5V is unregulated. Also, the expected reduction in voltage when a device is drawing too much current won't affect the system's functionality because the MCU only needs 3.3 V to operate correctly and the components will just need the 3.3 V connection on the MCU. So, if the voltage drops 0.5 - 1 V from the expected voltage it wouldn't affect the MCU or the components as their voltage requirements are still being met. [8]

The most sensitive components that we need to protect from noise and ripple are the OLED logic circuit and the tactile button because their recommended operating conditions have a smaller range and could easily be damaged permanently if the power supply delivers a voltage outside its range. In Table 17 below, we can see the recommended operating conditions for all the loads supplied by the stepped down 5V battery.

All the power conversions and battery management in the MCU will be handled by the PF1550 PMIC and the current drawn by the battery is according to rules specified by the PMIC. This MCU's DC to DC built-in design allows us to have two voltage sources on the board, one with a nominal voltage of 3.3 V (+3V3), and the second with a nominal voltage of 5V (+5V). This will be useful to power the various range of components with different input voltage ranges in case we would expand on this project and add additional components in the future. The values in the table below are taken from the datasheet of the respective component.

NOTE: The Tactile Button is subject to change and is only an initial design.

Part	Voltage Range (V)	Nominal Voltage (V)	Current (A)	Power (W)
OLED Logic Circuit	1.65 - 3.3	2.8	0.019	0.04256
Tactile Button	2.9 - 3.7	3.3	0.000075	0.0002475
MCU	3.3	3.3	1.51	10.89

Table 17: Components Powered by Stepped Down 5 V Battery in Prototype

As we can see from the table above, these components will be powered by the stepped down 5V battery and will be drawing an estimated cumulative current of 1.529075 A. To estimate how long this battery will be able to deliver power to the components we will use a simple formula, $\text{Hours} = (\text{mAh})/(\text{A} \times 1000)$. So, if the 5V battery is rated for 2,000 mAh and the total current drawn is 1.529075 A then this calculates to 1.31 hours. This means that if these components draw 1.529075 A constantly from the 5V battery, then the battery will last for 1.31 hours before it completely drains it.

Moreover, the loads shown on the table above will need to be powered by the MCU's +3.3V output pin. It is very important to connect the voltage source to the correct corresponding component to prevent damaging any components that are connected. The components are grouped by the nominal voltage required, to allow our Portenta H7 to power on these components without needing to tap onto the other +5V output pin. This will help reduce the number of cables under the scope which will help with the space constraint.

Grouping the components by the nominal voltage required allows us to not utilize the built-in +5V output pin because we don't have any components that need an input of 5V. This increases the allowable space below the scope for other components and parts. If we were to also use the +5V we would need around two to four extra wires and that will create a conflict with our space constraint as well as increase the possible damage the connections might incur with more cables tightly packed under the scope. This is a very important aspect of the power system because we still need to consider the step up 12V that will be used for two other components to complete the system and it will be discussed further below.

5.7.2 12V Boost Converter

In this section, we will be explaining the design and procedure that the 9.6V will do to power on the two other necessary components to complete our design, which are the rangefinder and the OLED driving panel. Both of these components require an input voltage of 12V, and we will reach that voltage by using the 9.6V battery and stepping up the voltage to 12V. Once that is complete, we will be able to deliver 12V directly to these components.

The entire schematic of the 12V boost converter is shown below on Figure 19. It shows on the left side the input of 9.6V and on the right side the output of 12V. In the middle it shows the TPS55340RTER voltage regulator that will be using resistors, capacitors, inductors, and diodes in parallel and/or series connections to create this circuit. In this circuit, we will be using four resistors, nine capacitors, one inductor, and one diode. The label and the values of each of the components are shown in the figure below.

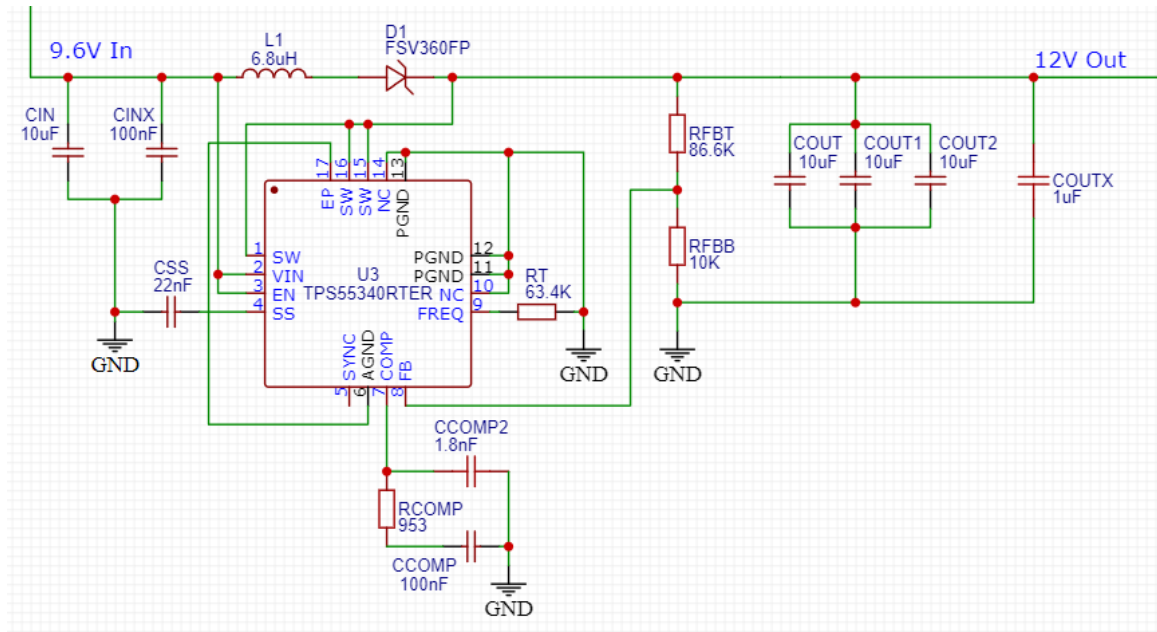


Figure 13: 9.6V to 12V Boost Converter

As you can see above in Figure 13, to complete our power system we are going to implement the voltage regulator TPS55340RTER to get the initial 9.6V from the battery source and step-up to 12V to feed the necessary components as listed on Table 19 below. This switching voltage regulator’s design flexibility is boost and it features 14 pins. In Table 18 below are the specifications of this voltage regulator which are needed to complete this power system design. [10]

Symbol	Description	Min	Typ	Max
VinMax	Maximum input voltage	-0.3V	-	40V
Vin	Recommended operating input voltage	2.9V	-	32V
Vout	Output voltage	Vin	-	-
Iout	Output current	-	-	-
Tj	Operating junction temperature	-40°C	-	150°C

Table 18: TPS55340RTER Specifications [10]

The Portenta H7 recommends an input voltage of 5.5V maximum and its two output pins are only 3.3 and 5V. So, the MCU will not allow us to power on the rangefinder and the OLED driving panel that requires an input voltage of 12V. We will need to design the power system with a separate power supply dedicated solely for the

range finder and the OLED driving panel, which will be powered by the 9.6V battery. [4]

This 12V battery is going to be stepped up from 9.6V to 12V to meet the voltage requirements on the components. The battery is arranged with 8 cells in four rows of two cells and this arrangement is best for our design because this size will fit best under the scope as mentioned in the previous section. The 12V battery includes two wires of 18 AWG and according to the National Electrical Code this gauge size is rated for a max current of 2.3 A. This amp rating will be sufficient to draw the amperage required by the components. [5]

The battery's output voltage is going to be unregulated, so, the voltage will likely fluctuate from the initially anticipated 12V. We plan to minimize the noise and ripple by utilizing the same tactics as discussed in the previous section, which were to twist the cables and keep the cables short.

This 9.6V battery that will be stepped up to 12V battery won't be adding any significant weight to the entire system, but either way it is essential for the system to work properly. Initially, our design was calculating to use two power supplies instead of one power supply. We thought our power design would work best if we use a 4.5V and a 12V power supply. Using two to power the components separately depending on their necessary voltage requirement. Using two power supplies wasn't the best for weight management and space constraint under the scope, so the new power design only includes one power supply of 9.6V. The initially expected weight of the two power supplies design was estimated to be around 292 grams, or 0.64 lbs. Now, with the new power supply design with just the 9.6V battery, the new weight of the power supply system is estimated to be around 220 grams, or 0.48 lbs. The new weight reduction is great to meet our goals of weight, but the new weight isn't significantly different from the previous weight.

As you can see from Table 19 below, we will only be powering the range finder and the OLED driving panel with the stepped up 12V battery. This battery will suffice the electrical requirements for these two components.

Part	Voltage Range (V)	Nominal Voltage (V)	Current (A)	Power (W)
Range finder	5 - 18	12.5	0.5	3.75
OLED Driving Panel	12 - 13	12.5	0.019	0.19

Table 19: Components Powered by Stepped Up 12 V Battery in Prototype

As we can see from the table above, these two components will be powered by the 12V battery and will be drawing an estimated cumulative current of 0.519 A. To estimate how long this battery will be able to source power to the components

we will use a simple formula, $\text{Hours} = (\text{mAh}) / (\text{A} * 1000)$. So, if the stepped up 12V battery is rated for 2,000 mAh and the total current drawn is 0.519 A then this calculates to 3.85 hours. This means that if these two components draw 0.519 A constantly from the 12V battery, then the battery will last for 3.85 hours before it completely drains it.

For the OLED display, this component will be powered on by two pins which are VDD and VCC. The pin VDD is for the OLED logic circuit and it will be using an input of 3.3 V. The pin VCC is for the OLED driving panel and it will be using an input of 12V.

5.7.3 One-line Diagram Distribution

The 9.6V battery will step down to 5V to power on the MCU to then distribute the power to the other two components, which are the OLED logic circuit and the tactile button, through the +3V3 output pin. Also, the 9.6V battery will step up to 12V to power on the other two components directly which are the rangefinder and the OLED driving panel. Below in Figure 14, it will be shown how the power is distributed from the 9.6V battery to their respective components for this power system design.

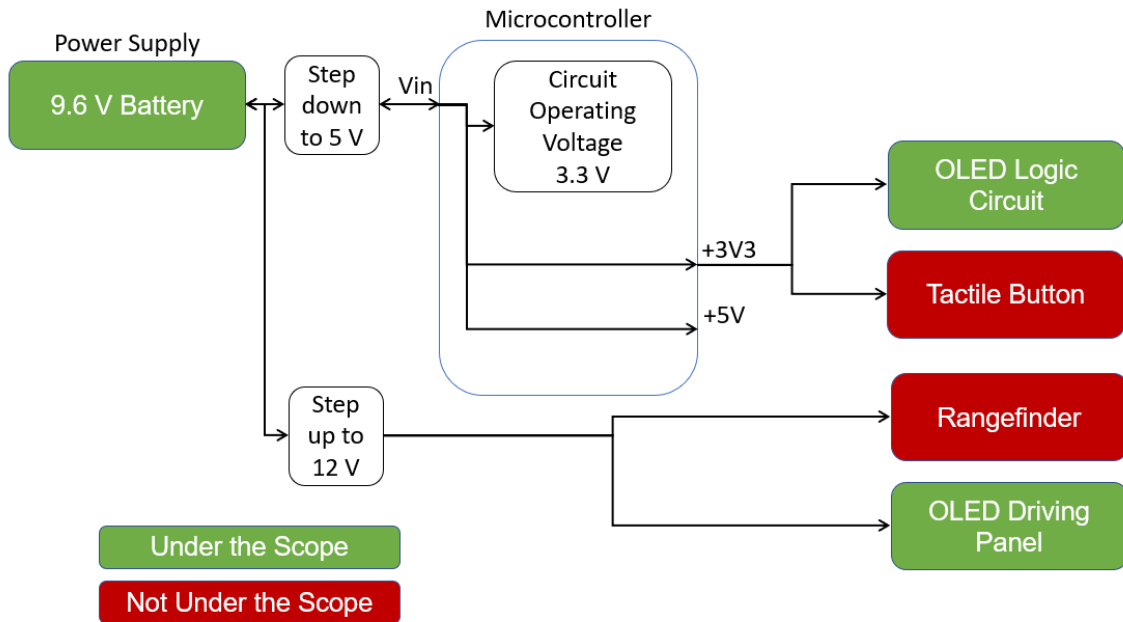


Figure 14: Components Powered by Power Supply

The power source for the power supply comes from an external battery source of a 9.6V battery. This power source is not regulated but we will minimize noise and ripple by tactics discussed in the 5V Step Down section above. These tactics will help us output the necessary voltage and current to the hardware components in

the system. The power source for the microcontroller is tapped at Vin after it has been stepped down from 9.6V to 5V. The pin Vin will be used to power the MCU's circuit operating voltage of 3.3 V, and after that, it will power the output pins +5V and +3V3.

Now, the rest of the components in the system will be powered by the same 9.6V that is stepped up to 12V. This stepped up 12V will be directly connected to the components because the MCU doesn't support the output of such a high voltage. The input to these components will be dependent on the 12V input to be regulated or unregulated. This battery isn't regulated but we will use similar tactics as done on the 5V battery to minimize noise and ripple. With these tactics we will try to assure that the components receive their voltage and current requirements smoothly and without a lot of drastic changes and spikes.

Revision

Initially, our battery design was to use two batteries for the power supply design. The first battery is 5V and the second is 12V. These batteries were going to feed into the corresponding components directly. The cons with this power supply design in comparison to our current one was that it was more expensive by almost double, it required more space which was an estimated of double the size, and the voltage from the batteries was unregulated. On the other hand, the new and current power supply design only consists of one battery. It is a 9V battery and it will use voltage regulators to step up or down and outputting regulated voltage to protect components in the system. The pros with the current power supply design is that it's less expensive by almost half, it requires less space as it's only one battery instead of two, and it outputs regulated voltage. This power supply design classifies under the Extra-Low Voltage power supply standards which we meet as our design doesn't exceed 60VDC and our wires are protected by basic PVC insulation.

The following diagram is a one-line diagram that demonstrates the distribution of power from the power supply, the 9V battery, to the different components in our project. We are using three voltage regulators to power all of our components from the 9V battery power supply. Moreover, our design has three voltage levels to choose from: 3.3V, 5V, and 12.77V. We'll be using the HT7533-1 for the 3.3V output, the HT7550-1 for the 5V output, and the AP3012KTR-G1 for the 12.77V output. These voltage outputs will be regulated as they are being converted from an unregulated power supply, 9V battery, to a regulated voltage through the voltage regulators. These three voltage levels that are available allow to power all the components within their recommended input voltage range. Furthermore, these three voltage levels, allows for any future expansion that might be required for any additional components that we might need if we add new features to our project.

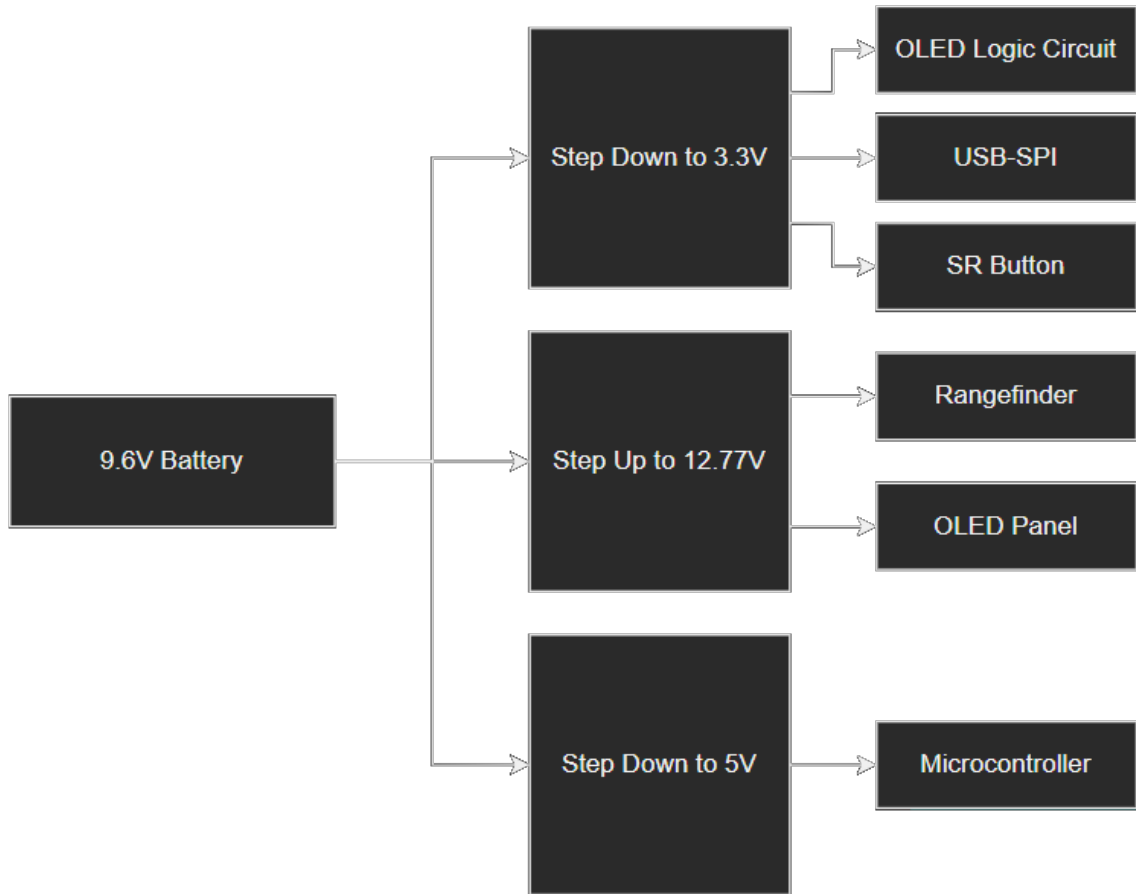


Figure 15: Power Supply One-Line

5.7.4 Interface Board Design

In this section, we're going to show the overall design layout of the power supply and how all the pieces work together in harmony. All the way from getting the necessary voltage threshold to delivering the power to the components and the MCU to the connections. Also, we will discuss the voltage regulators and the boost converters that make this system possible and efficient to reach our goal of saving space and weight.

The design of the power supply as shown below in Figure 16, was constructed with the software called EasyEDA. We were able to design this compound circuit with the help of this software and its tools to facilitate the process and procedures required to complete this successful circuit. By choosing the parts needed (resistors, inductors, capacitors, and semiconductors) for this project we were able to map and pinout our board. Next, we needed to adjust the components so the footprint fits better in the confined space as well as to have less overlap between connections.

In Figure 16, you can see the entire power and display interface board. This figure is showing the input +Vin of 9.6V on the top left of the figure and it is connected

directly to the voltage regulator that is going to transform it from 9.6V to 5V. Also, it is directly connected to the boost converter that is going to transform it from 9.6V to 12V. We will discuss these in greater detail below. After the voltage has been transformed to its respective voltage it is now ready to be connected to the MCU or the components in the system as you can see from the two output pins Vout12 and Vout5 on the right side of the figure. On the top right of the figure, we can see the OLED interface that is being connected directly to output pin Vout12 as that is the recommended input voltage it requires. The 4-pin header on the top left of the figure connected to the OLED interface is responsible for the I2C interface for the display.

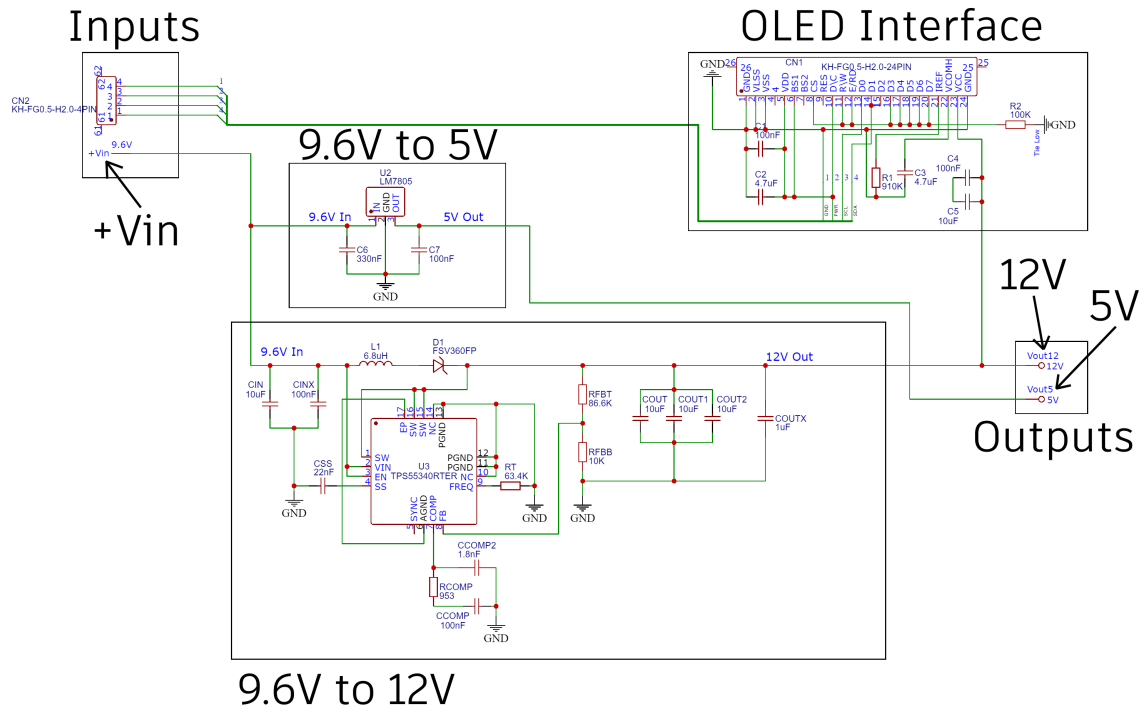


Figure 16: Power and Display Interface Board

First of all, the 9.6V from +Vin is taken by the voltage regulator circuit, LM7805, and stepping down to 5V. It is able to do so with the help of two capacitors that are connected to its terminals which GND is grounded. This regulator only accepts DC voltage and our battery is DC so it would be compatible. The maximum output current that the LM7805 can reach is only up to 1.5A. This is perfect as the MCU and the two components that are going to be connected to the output pin 5V will only require an estimate of up to 1.5A. This voltage regulator is also packed with internal thermal-overload protection which will help in the application of our project in case the user is out in a hot area, the component will turn off when excessive heat gets generated within it for any unknown reason. Also, the recommended

operating conditions for this voltage regulator are from 7V to 25V and we will be inputting 9.6V directly from the power supply so it will satisfy it. Moreover, as we can see from Table 6, the estimated lowest voltage that 9.6V battery will reach is 9.29V when it is at a 10 percent capacity. So, this voltage regulator will always be receiving the recommended operating conditions.

The second important part of this schematic is the TPS55340RTER. This device is a SEPIC converter and will allow us to take the initial 9.6V from the battery and step it up to 12V. This voltage will be necessary for the rangefinder and the OLED driving panel. With the help of a combination of circuits connected to this device, we are able to successfully step up the voltage to the desired 12V. This device allows for a maximum allowable output amperage of 5A which is plenty for our application. Our project will only be drawing an estimate of less than 1A for the two components connected to the Vout12 output pin. Also, the recommended input voltage range for this device is from 2.9V to 32V. We are going to remain inside this range as we're going to be connecting a 9.6V DC battery to the input pin.

Also, below in Figure 17, is illustrated a 3D rendering model of the board that we designed. It shows all the components that are being used with the same names as the schematic above in Figure 16. All the parts with the descriptions, quantities, and more will be listed in the BOM.

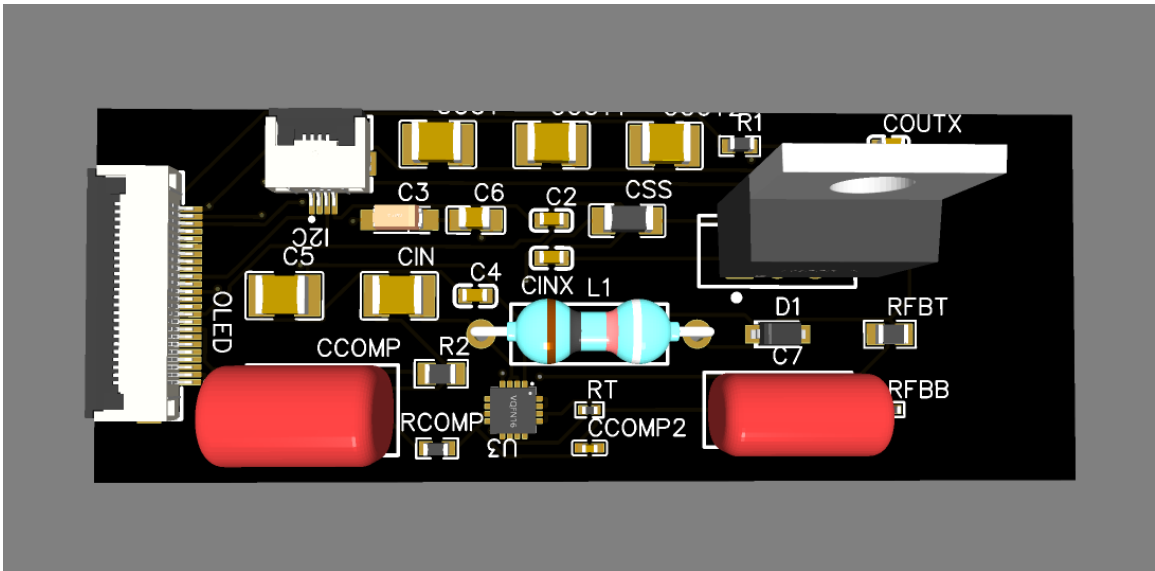


Figure 17: Board 3D Rendering

Revision

The overall schematic of the Power and Display Interface Board has been changed to accommodate for the components needed in the design as shown in Figure 18 below. The OLED Interface schematic and design has been kept the same but the voltage boosters have been changed from 9V instead of from 9.6V. We also added

a voltage regulator to drop the voltage of the power supply from 9V to 3.3V to feed some of our components. Lastly, we added a new USB-to-I2C integrated circuit for an app integration we needed; more on it below.

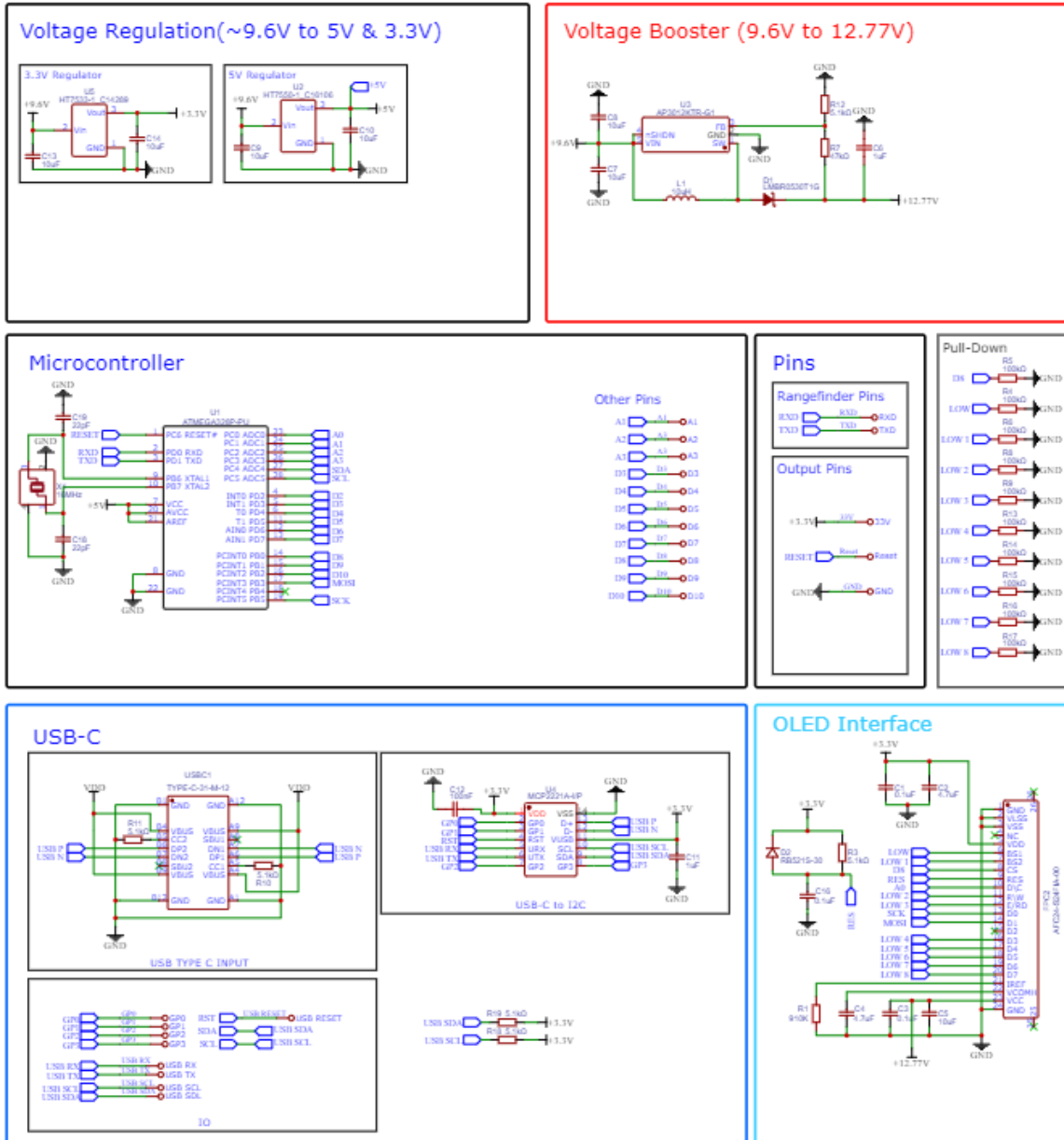


Figure 18: Overall Schematic

Below in Figure 19 it shows the PCB Interface Design in a 3D model rendered by EasyEDA. This is the PCB for all the components come together to communicate with each other. When designing this PCB we assured to not have any 90 degrees turns to ensure there is good solid connections between components. The other design rules we followed was to have a track width of 0.254mm with a clearance in between tracks of 0.152mm as this aligned with the design capabilities of our PCB

manufacturer. We also designed for the most components to be surface mounted as it kept our manufacturing costs of the PCB lower. Then some other components were through-hole to make sure it was mounted well and could withstand environmental stress. All the components on the PCB are clearly labeled and they are the Microcontroller, USB type C connector, the USB-to-SPI converter microchip, the OLED display connector, voltage regulators, the capacitors, resistors, inductors, the GPIO output pins, and the SPI output pins. The board thickness we chose is 1.6mm which is sufficient for a 2 layer PCB and it also aligns with our PCB manufacturer minimum requirements.

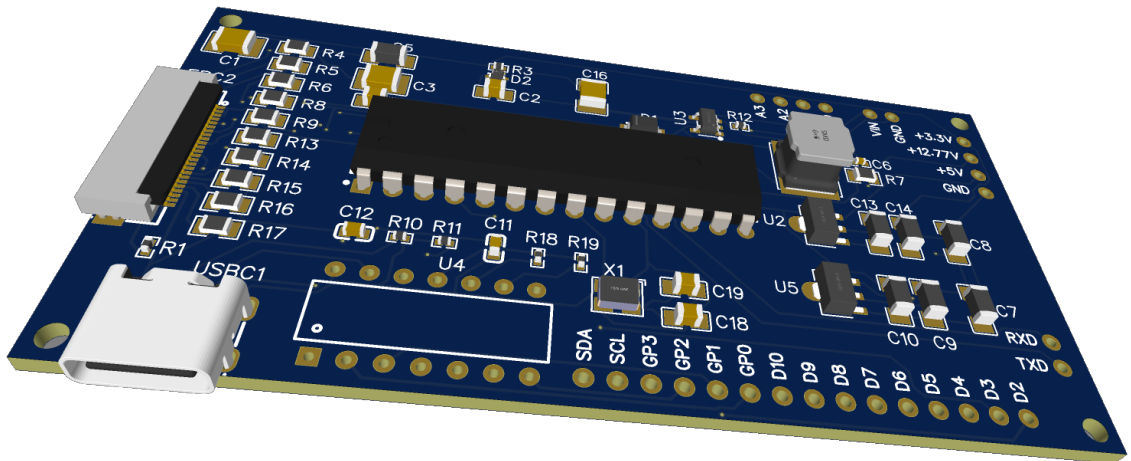


Figure 19: 3D Model

A USB-to-I2C has been added to the overall Interface Board Design which includes all the components. The USB-to-I2C integrated circuit on the PCB is the interconnection of the USB Type C connector and the MSP2210 with the GPIO pins. The Figure 20 includes the circuits for this USB-to-I2C interconnection. This integrated circuit is built into the PCB design for a seamless design with all the components to be signaled and powered from the same PCB. The USB Type C connector on the PCB powers the microchip MSP2210 directly. This input connector C-31-M-12 has 16 pins in a symmetrical fashion so that the cables can be reversed when connected. The USB Type C connector was mostly used for the I2C interface and input power, all the other unused input pins were connected with pull-down resistors to ground to ensure we keep those input signals LOW. The clock on this microchip is a ceramic resonator oscillator that runs at 12MHz which provides the proper frequency for the USB module to function properly. For the app integration with the smartphone, the USB Type C connector is required to have connectivity with the smartphone and convert the USB signals to I2C interface to be able to communicate with each other. The microchip is able to convert the USB signals to I2C legible signals via the data lines on the connector.

USB-C

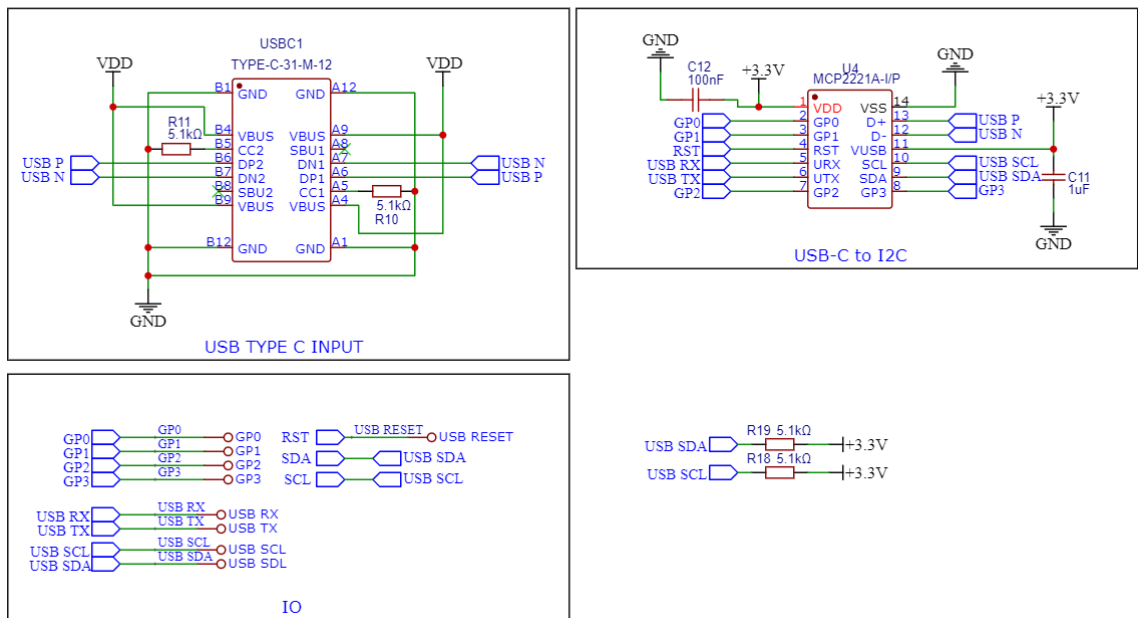


Figure 20: USB-to-I2C Integrated Circuit

5.7.5 Power Supply Housing

The housing for our power supply design will be the space allocated right below the scope. The space between the bottom of the scope and the top of the hunter's rifle will be the space that the power supply and all its connections will be located. Using this confined space correctly will allow to align with our goal to keep the design small and if packed all under the scope. This will allow for a sleek design won't change the center of gravity of the rifle when the hunter is holding it and it will be right under the scope which already adds some good weight to the rifle.

The measurements of the battery and all the components going under the scope, as shown in Figure 14, has been accounted for and will fit under the space below the scope. The measurements of the MCU will allow us to fit it under the scope as well. Also, the estimated measurements of the PCB are very similar to the MCU so it will also be able to fit under the scope. So, the MCU, the PCB, 9.6V battery, OLED logic circuit, and the OLED driving panel will all be able to fit under the scope with the estimated measurement that is allocated under the scope.

Something to note is that the power supply design won't need any additional parts to complete the housing. The scope will suffice to complete the necessary housing for our design and it will be able to keep the components, MCU, and board protected.

5.8 Software

In this subsection we go into detail about the software we are using to run this scope. We will start by going over the language we have chosen and the different modules we are using. Because this project makes use of a Bluetooth module we will also cover some aspects of developing an Android app to interact with the micro-controller.

5.8.1 The Python Programming Language

For the purposes of our project we were limited in programming languages as it needed to be supported by the micro-controller of our choice as well as have enough libraries to support our multi-external device project. We selected the Arduino Portenta H7 as our micro-controller of choice, and they support two high-level languages, C and Python (specifically micro Python). As the programmer I have programmed micro-controllers in C in the past and while C is an ultra-fast language it is outdated, hence I am using Python as my high-level language of choice. In the following paragraph I will be discussing what python is in detail as well as why it is a superior language for our project. The following figure describes the differences between Python and C.

Python	C
Uses Interpreter	Uses compiler
Simple syntax that resembles talking language	Not Simple
Can write Scripts in any environment	Scripting only allowed in IDE
Low Performance due to line by line interpretation	High performance due to complete file compilation
Allows exception handling	Does not allow exception handling
Python allows for no type declaration	Has to declare type on variable initialization
Use's automatic garbage collector for memory management	Programmer has to manage memory at all times
Pointers not available	Can use pointers

Table 20: Python vs. C

Python is an object orientated programming language that has an array of applications including machine learning, image processing and web development. It utilizes an interpreter which means that each line of code is ran and outputted as the

code works down in the file, this helps us detect errors if there is any as the code runs. Since Python is an object-oriented language the class of exceptions exists to aid us in narrowing errors down in code instead of a segmentation fault error in C, it would raise a Run-time error and show the error usually because of a high recursion loop in code. Python also creates memory automatically for objects which is essential as we have a very limited amount of space on our micro-controller, python also utilizes a Garbage Collector which automatically sweeps through our memory space and accurately releases memory heaps.

5.8.2 Python Development

For our file structure of the micro-controller that we chose is Portenta H7, it contains four files. The first file is the boot.py, second is main.py. Following those are our driver files and a read me file.

Our boot.py contains the file which will load when the board first boots up, for the purposes of this project. It will contain four functions. The first function will be used to make sure all connections are operational, by this I mean we will attempt to connect to all our external devices if any one throws an error it will be displayed either in the form of an LED flash, or in the form of the LCD screen if it successfully connects. The next function would test the capabilities of our emitter and receiver which would entail running it at least one time to make sure we receive a distance, initially I would set the distance to a negative number or "None" so that if we do not receive a number or it does not get assigned properly we would throw an error on the LCD and through the console in case it was in debug mode.

The next file is the main.py, this file would contain all the important functions that relate to determining range and displaying it on the display module. The first function would initialize all connections once again since it is a different file, this would include connecting to our devices via ADC, I2C, or UART. The following function would be used to calculate our fall off distance and all necessary elements to determine the changes to necessary to compensate for the bullet fall off distance. Following this function we will have our main loop, that would mimic the process to use the device in a real scenario. First element is to wait until the button is pressed to lock in our distance, a display message would appear to confirm the lock, and this is when we call our second function to determine the falloff, the return tuple would then be represented onto the display this process would then repeat for as long as the stop button is pressed to conserve power. Following this I would check if a Bluetooth connection has been established so that the data can be sent via the Bluetooth connection to the app of our choosing. Below is a flowchart displaying all the elements I have discussed above.

5.8.2.1 PYB Module

Historically coding with micro controller has been done with the C programming language, utilizing libraries that are created from the company that made the micro-

controller. Nowadays utilizing python is very common in many aspects of programming, even micro programming. In our project we will be using the PYB module that contains important assets used in micro-controller programming that include:

- PulseIO, for pulse related hardware, in our case we will be using it to power LED's to flash for different scenario's such as red for warnings and blue for processing.
- Time, very important as delaying functions is common-place in micro-controller programming.
- Busio, used to create connection objects that include
 - I2C
 - SPI
 - UART
- Bluetooth, talked about in depth below.
- Help, to aid us in finding quick documentation for the included functions.

All of these assets would be included to create a python project that would utilize important micro-controller capabilities that would aid us in creating a fully functioning a quick project.

5.8.3 Tensor Flow Module

Tensor Flow is a very popular machine learning package that gives access to a host of different tools to create examine and predict data. For our project I will be utilizing their neural network features specifically the convolution neural network. In short machine learning is an algorithm that will assign weights to various features that you give the model so if were trying to predict how fast someone can run, you could give it the features of height, weight, and athletic ability and the machine learning algorithm would utilize these features to obtain a formula to predict how fast someone is all you have to do is input a height, weight, and athletic ability. For our project we will be using a convolution neural network, this works in a very similar fashion to a normal neural network but instead takes features from an image instead and makes predictions based on that image. A Good example would be if we had pictures of animals and we trained our convolution neural network with the knowledge that a dog is a dog, cat is a cat and henceforth. The model should be able to tell you what animal it is based on the image you give it. In the next paragraph I will be explaining the general outline of a Convolution Neural Network using Tensor Flow and all its different attributes.

The first step in creating a model is too create a our training, testing, and validation set. This is important as we can use the same data set to test and validate in addition to train. Next thing is to decide on any preprocessing of data, this may include deleting empty entries and replacing outliers. Since we are collecting our

data through code we can decide what we can allow into the data-set, We will not be allowing NULL values and empty features in an entry so as far as we are concerned we do not need to perform any data preprocessing. Next step would be to determine how we would like our images to be displayed typically it would be in a 32 by 32 grid with grey scale filter to reduce the amount of data we turn it into grey-scale so we have less channels. Finally it is to build our model, this is a trial and error process as we do not know how each model will run until after we run it, but typically we will be using layers such as MaxPooling, Dropout, and Dense to achieve a good accuracy with minimal loss.

After creating our model now we need to display analytic and predict using that same model this is done in mainly two steps. First is to predict using the model, this is done in such a way that our output data is not what we want to see in the final dataset so we would need to do some data manipulation this would involve us to convert our output array into an array that matches the pixels on the OLED display. Once we have achieved the right amount of pixels we can then map pixels so that it is an accurate representation of the predicted output. Once we have our output we can then display analytic this could vary for what we need but it will mainly consist of box-plots to find outliers, scatter plots to determine relationships and finally another set of scatter plots which is found in the Tensor-Flow functions, that would show us how our accuracy changes with respect to epochs and how our loss changes with respect to epochs.

5.8.4 Bluetooth Connection

Utilizing an Android application to enable us to input key parameters into our fall off function is a feature that we have decided to implement as the convenience would be an important to the success of our project in a real world scenario. The Bluetooth will be done using a micro-Python library Bluetooth.BLE this library would allow us to connect to all devices via a stable Low Energy Bluetooth connection as well as being able to send data packets through that same connection. This connection would be stated as a peripheral connection in the python library, an event handler would be coded to configure the initial connection to another Bluetooth device, as well as handling retrieval and the sending of data packets through the right connection.

We will be using the BLE package to initialize our Bluetooth connection, this package is an extension to the Android library, for the most part Bluetooth connections are done through the default android package "android.Bluetooth" this package allows us to connect to a wide variety of Bluetooth devices including direct support for a Bluetooth headset. However the Bluetooth connection that our micro controller requires is not of the same kind it is a Low Energy Bluetooth device which requires a different connection to that of normal Bluetooth connections. To import this module we would first need to install it on our local computer then transfer it into our GitHub repository, this is done by installing it through a popular package manager Called Maven followed by utilizing the "implementation" command in Kotlin. This package

or module allows us to connect to low energy Bluetooth devices and utilize all of the normal commands and more. We would be using a Blemanager class to connect to our smart phone and call back functions to send data through the devices as well as get notifications of when data has been received from the micro-controller.

5 .8.5 Data Visualization

Python is a powerful language utilized for many topics that include scripting and image processing, one of these topics is data handling and visualisation in addition to machine learning. For the purposes of our project I would like to be handling data outputs to a csv file that would then be read into as a Pandas Data Frame this would allow us to organize our data and manipulate it to our liking, typically that includes Sorting and handling outliers. After this we can then utilize the Matplotlib module to visualize this data into an array of graphs such as scatter, bar and box plots this would give us an understanding of how well our design is performing and we can compare it to real world data to find our efficiency.

5.8.5.1 Pandas Module

This pandas module is a very common extension to python that is common for its Data Frame object, which is essentially a labeled table that has a lot of very important attributes of functions that allow us to manipulate our data and generate important information such as mean, median and mode. An Example of a Pandas Data Frame that contains data from the CDC regarding the Covid-19 Deaths.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 72864 entries, 0 to 72863
Data columns (total 15 columns):
#   Column                                     Non-Null Count  Dtype
---  -
0   id                                           72864 non-null  int64
1   Data As Of                                  72864 non-null  object
2   Start Date                                  72864 non-null  datetime64[ns]
3   End Date                                     72864 non-null  object
4   Group                                        72864 non-null  object
5   Year                                         72864 non-null  object
6   Month                                        13464 non-null  float64
7   MMWR Week                                   58608 non-null  float64
8   Week-Ending Date                           58608 non-null  object
9   HHS Region                                  72864 non-null  object
10  Race and Hispanic Origin Group             72864 non-null  object
11  Age Group                                   72864 non-null  object
12  COVID-19 Deaths                           57938 non-null  float64
13  Total Deaths                               48706 non-null  float64
14  Footnote                                    34493 non-null  object
dtypes: datetime64[ns](1), float64(4), int64(1), object(9)
memory usage: 8.3+ MB
None
```

Figure 21: Pandas DataFrame

5.8.5.2 Matplotlib

Matplotlib is a popular extension to python that allows for graphing of data objects and works extremely well with Pandas to graph its attributes. It serves a great purpose to visualize the data into a variety of different plots that gives a lot of insight into what exactly our project is doing and how well it performs in the real world. An example of a graphing feature that is the amount of deaths as time goes on versus the Race of the person.

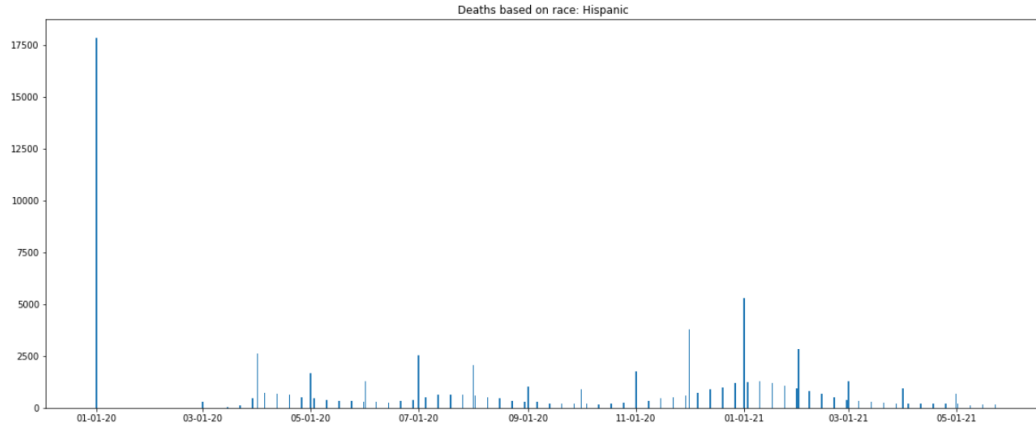


Figure 22: Matplotlib Graph Example

5.8.6 Flow Chart

The flow of the system is defined by the flowchart below in figure 23. This flow is important for the proper function of the system as it dictates the order in which operations should occur. The flow chart has 3 distinct states, the first it when the system is being initialized. Next is the waiting state, while we wait for input from the user, finally the last state is when we take the input from the our components and output data to the user.

At the start of the flow chart, we start by initializing several different components. The first thing we initialize is the connection to targeting lens. We want to connect to this first as it is the one way we are able to indicate critical errors to the user. Next we will initialize the laser rangefinder, when initializing we will check for any error codes; if there are error codes, we will display them to the user via the targeting lens, otherwise will will continue. Finally we will register our button listeners, so we can wait for input from the user. Once all these are setup we will move onto the next state.

This state is the loop state, which the system will be in for most of the time. The only job of this state is to wait for input from the user. To escape this state, the user must press the momentary push button. This will then exit the loop and move onto the next state in the system. During this state the microcontroller will also listen for updates from the app that it is connected to via Bluetooth. In this time it will be able to update the various settings like the targeting lens display as well as the bullet specifications which will affect the bullet drop calculations.

The final state of the system is processing the users input as well as the input from our components. The first thing that happens is the system will display a message on the targeting lens indicating to the user that a measurement is being taken. Next the microcontroller will send a signal to our laser rangefinder telling it to take a measurement. Once the measurement is taken, the microcontroller will update the targeting lens according to the distance that taken. The targeting lens will then stay

in this state until the user pushes the button again to take another measurement or the device is powered off.

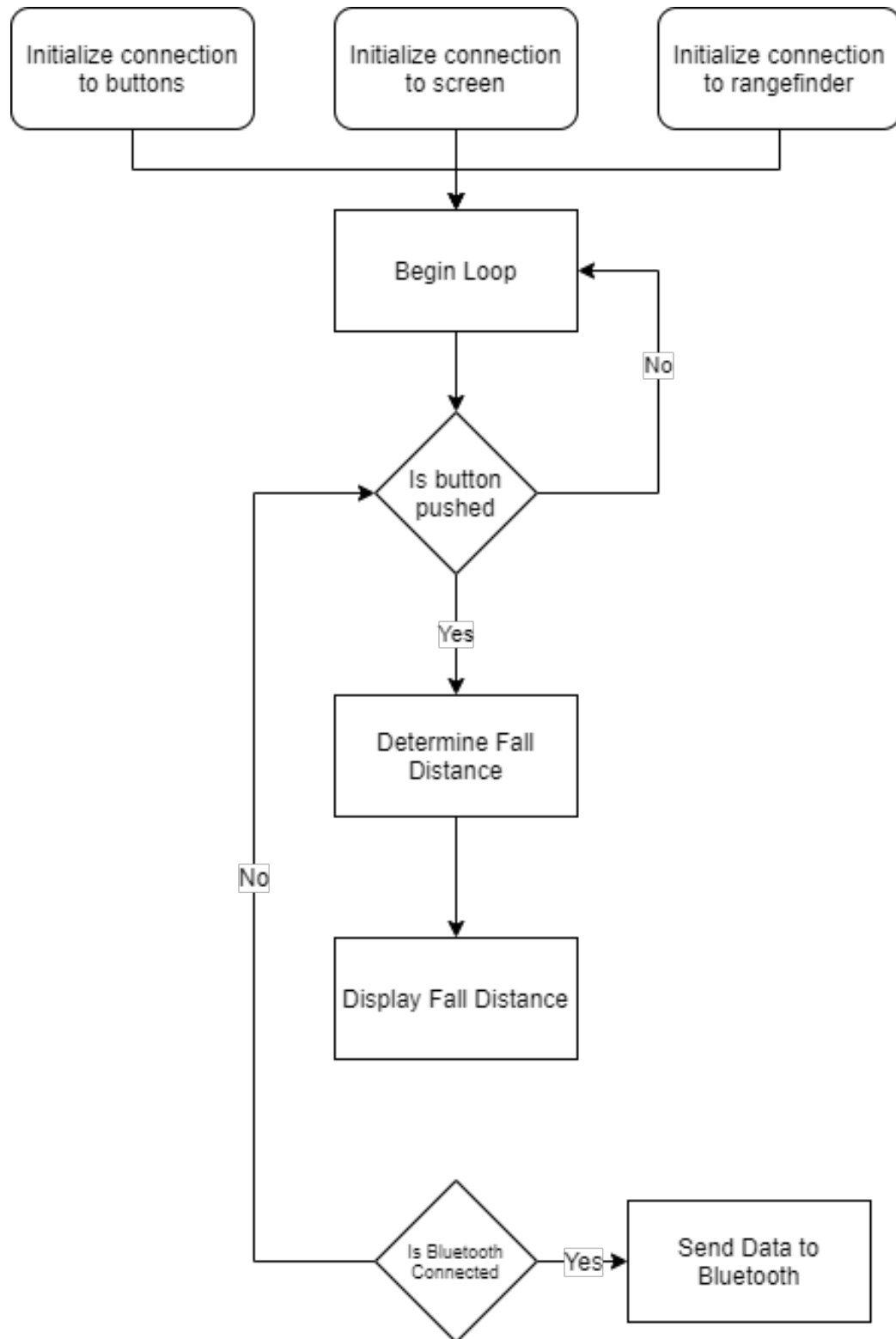


Figure 23: Software Flowchart

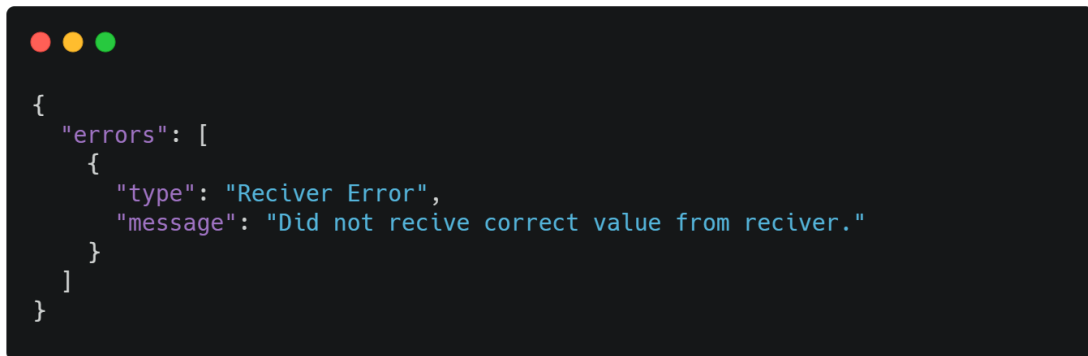
In addition to these files I will be adding two extra files for our convenience, the first JSON file will contain error messages instead of utilizing python errors I will be creating a JSON or Python Class to create my own errors so that we can determine errors faster. Additionally since we need some parameters to determine the fall off distance, I would create a JSON file with every gun and its parameters for easy access to the most popular guns as well as the one we are using to test. I have included examples of what each file might look like, in the following images.

5.8.7 Examples of JSON



```
{
  "7.62x93mm": {
    "_comment": "This is for a 7.62x39mm sized bullet",
    "weight": 8,
    "speed": 710
  }
}
```

Figure 24: JSON Bullet Format



```
{
  "errors": [
    {
      "type": "Reciver Error",
      "message": "Did not recive correct value from reciver."
    }
  ]
}
```

Figure 25: JSON Error Format

5.8.8 Function Breakdown

In this section I will be discussing all the different functions that will be implemented in our project including some basic coding objects as well as the entire function of the various code blocks, this section will explain in great detail all of the software aspects of our project.

5.8.8.1 Init Functions The first set of functions contained in our "boot.py" file in our initialization functions, these act as a first testing phase to make sure all of our code runs effectively without wasting battery attempting to perform all the major tasks, this would include initializing an I2C object in the following format `connection = I2C(*parameters*)` with the following parameters.

- Mode: Either being the slave or the master
- Addr: Address of our I2C, mainly used when I2C is a slave
- baudrate: SCL clock rate, used by the master
- dma: Allows for ultra fast transmission but does not handle errors

Additionally I will be initializing an SPI connection in the following format `connection = SPI(*parameter*)` with the following parameters.

- Mode: Slave or Master
- baudrate: SCK Clock rate
- Polarity: Can be 0 or 1
- Bits: 8 or 16, packet size
- first bit: Least or Most Significant bit

Finally I will be initializing an UART connection in the following format `connection = UART(*parameter*)` with the following parameters.

- baud rate: Clock Rate
- bits: Numbers of bits per character
- parity: Even or Odd
- stop: Number of stop bits
- flow: Flow control type
- read buff Len: Character length of the read buffer

Encompassing all of these declaration will be a try-catch block, which works to catch Errors as they appear in our code. Within the block the interpreter will run each line and if a connection has been failed that block of code will immediately jump to the catch block which will output a file displaying all of the errors information so we can track and work on errors in an effective manner. After all of these connections have been made we will test each connection to make sure on the other end of the connection the device is working as intended.

Our first external device to be tested would be our buttons since the buttons work as an input device only, the user would need to press down on the buttons to continue testing the system, following the button press the micro-controller would send a

signal to display an LED to let the user know that the button test has been successful. Following that test we would next test our LCD display, our LCD is structured into 8 pages or rows and 128 columns and within each column contains 8 bits that can be turned off or on at any time. We would test our display by turning on each individual pixel from each page however if one pixel does not turn on that does not cause us to abort our testing we would set up a counter so that if more than three columns worth of pixels is not operational we would abort our testing phase. Finally we would test our Emitter and Receiver, this would involve us to power on both devices and receive a distance through a connection of our choosing once we have received a distance we would then proceed to display it onto the display to accurately make sure that the distance acquired is correct, once it is displayed onto the screen the user would need to press the button to complete the testing phase.

5.8.9 Main Functions Breakdown

Our main.py would contain all the essential functions that is required to calculate the fall off display it and have all required functionality such as button presses and displaying LED's as well as app configuration. This section would work under the assumption that all essential connections are operational and work as expected however I would be doing a error catching on essential code breaking sections using my error.json file and outputting them into a csv file due to the severity of the errors.

Our first function would be constructed using pandas and the basic math library of python, its main goal is to output parameters that involve fall off distance based on parameters that we input into the function, parameters such as bullet size, bullet weight and bullet speed will be selected by the user before the program runs as those parameters cannot be determined by code, currently our application would determine all the parameters from a user selection. In the event that a smart phone is not available, my code will default to the M4/M4A1 as it is the United States Military standard issue rifle in case no gun parameter has been selected. The code will take the following input parameters:

- Bullet Speed
- Bullet Weight
- Gravity
- Drag (TBA)

The function would then output a tuple containing the fall off in the x-direction and then the fall off in the y-direction, our current scope of our project limits us to only output a correct value in the y-direction as we have no way of knowing information to determine a x-directional change, however future implementations of this same project may allow us to do so hence the reason for adding it to my function. That

concludes the function breakdown of calculating the bullet fall off, next I will be talking about display data onto our display.

Our next function is displaying data onto our transparent OLED display, one of the many benefits of utilizing python is its ease in coding this section of code, since we are using I2C the first step is to start the transmission using our slave address write a control byte and then write the data which must be as type unsigned 8 byte, however in python is to quite simple to cast variables as type declarations are not required. However the typical I2C transmission would work as follows:

- First Step: For the master to initiate a connection this step would always need to happen in the transmission pattern to send data over that I2C bus as that master
- Second Step: Is for the master to acquire the address of the receiver/sender depending on the scenario.
- Last Step is to send the data over that connection typically in a data packet design
- Typically the master ends the connection automatically but it is good practice to end the connection.

This would allow us to write data onto our display in an effective manner, this function does not need to return any values however it will be coded to return true in the case of success and false in case of failure this would allow us to error check in a considerably more faster way as well as to double check on both ends to verify success. Our next function is the function that would include our main loop which is essentially where our micro-controller will be in for the majority of the time it is running.

This function would contain our main loop which is the main attribute of our design, before we begin our loop we would need to set some parameters as once we are in our loop it is difficult to create and set variables that are required outside the scope of the loop. Firstly is to determine our button array which would essentially just set the state of our button since it is a single button it would work as a toggle which would switch the variable button state to either set or reset in either case we would need to alter where our code is running from in case of reset we would clear our LCD display and set our code to wait in an infinite loop until the state of the button has been changed this would be using an Interrupt service routine or ISR for short, what this does is it allows us to run multiple threads in our code and change variables and run functions as soon as a certain interaction has happened such as pressing a button in our case.

In addition to the main function, I will be including smaller functions that would run as a convenience to the programmer things such as resetting LCD's and variables that occur often would be bundled into a function for easy access, as well as outputting csv files or running data analysis on our data frames.

In addition to all of these functions I will be including a function that will allow us to display the battery life. Our micro controller does not natively support a battery life checker and since our battery will be made from scratch to accommodate all of our different pieces. Therefore we will be utilizing a custom function that would determine the battery life, we can determine this by incorporating a state of charge with our included battery. We first find out how many volts by reading in the value coming in through that port, we read that value then determine how much is left using a simple voltage divider equation. From there we can find out the percentage and display it accordingly onto our OLED screen.

5.8.10 Portenta Vision Shield

Future integration's of this part might include the use of the Portenta Vision Shield, this micro-controller board is an extension to the board we are currently using, Portenta H7, that includes a wide variety of features which include computer vision. For the purposes of our project we will be utilizing computer vision to improve upon our design. The first aspect that we would like to incorporate is a object detection, when the user sets the button to set the camera would take a picture, perform object detection on that picture and save it so that for every shot we can tell exactly what the weapon was aiming at. Additionally we can use the object detection to detect movements in the camera to enable features such as audio recording, and video recording this allows us to utilize the audio capabilities of the extension board and makes a very useful follow through into our app design. In our app we can have a direct feed through the camera and enable certain features such as Blob Detection, Histogram of Oriented Gradients (HOG) and Canny Edge Detection in our live feed this allows us to obtain key features and gives us the opportunity in using further computer vision concepts to break down the video and decipher objects. I have included key images of the different features that are available using this board.



Figure 26: OpenMV Cameras

5.8.11 OpenMV Cameras

In addition to the Portenta Vision Shield we have considered utilizing OpenMV cameras, OpenMV cameras are the cameras used in the Vision Shield just not natively supported on the Portenta H7, what this means is that we can not have one piece of code running both hardware devices simultaneously, instead we need to utilize a connection, such as I2C or UART, to send over data between the two devices. However this allows us to do many things such as utilize the computer vision extensions of OpenMV such as HOG, and Edge Detectors as well as their in built object detection system which will be useful to our implementation of these cameras. This camera would work just like the other camera with some minor changes. First thing is that to enable the picture we would need to send an enable bit to our camera micro-controller, this would allow the micro controller to exit out of an infinite loop, take a picture use computer vision to obtain multiple versions of the picture and upload it onto an SD card, this way we only send a single bit over to make sure it is working. Instead of sending an entire array of 32 by 32 images which would take a long time over the same connection. We may also index the folder by creating files that include the date and time as a long string as the folder name, and place the images in that folder. This would allow us to find images appropriately.

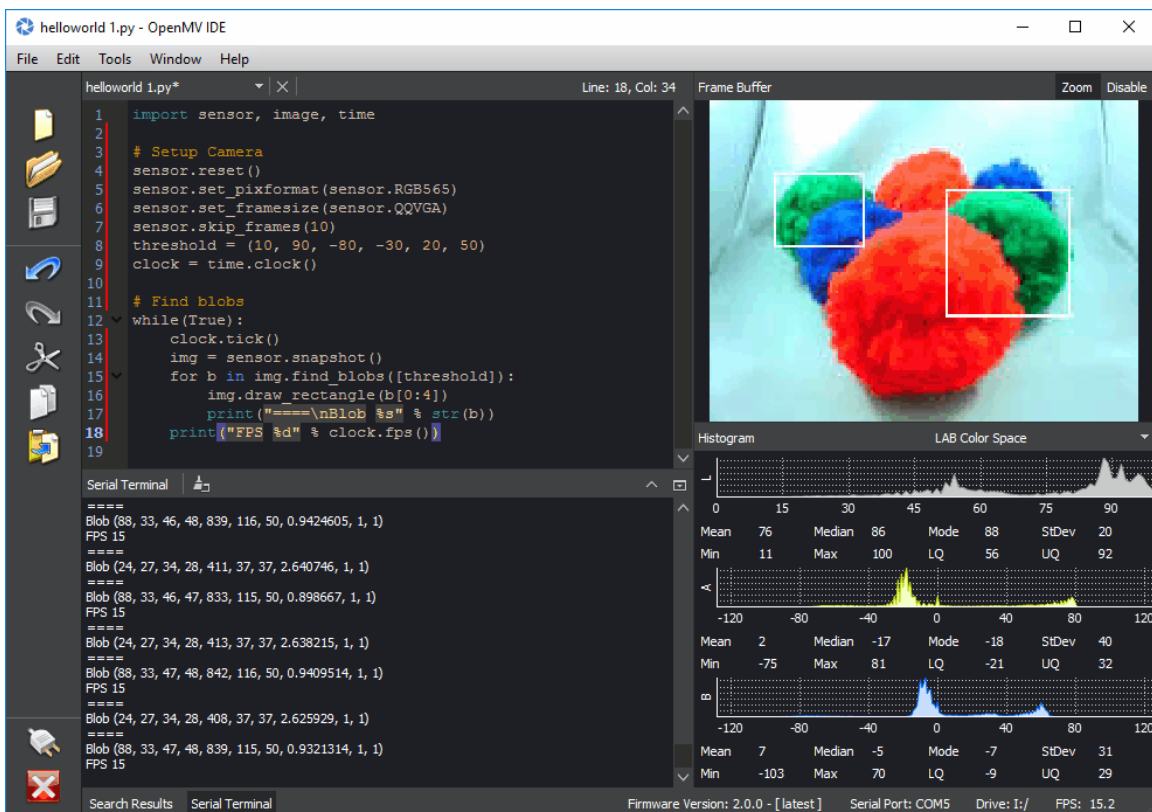


Figure 27: Block Diagram [12]

5 .8.12 Android Studio

Our app design will be fairly simple as it is not meant to have many features instead it will mainly be an add on to our design for convenience purposes only, our app will consist of three main tabs the first is a statistics page which will outline the different ranges and times each time a user clicks the set button the time, range, and bullet information will be relayed into the app in a list format, the reason for this is to outline the capabilities of our design as well as examples of what the user might be able to do with the scope. The second page will be a camera page this page will display the camera feed under the assumption that we incorporate the actual camera into our final scope. The Last Page will be a settings page, this page will allow us to configure our bullet data, set different formats and set different display settings for our camera feed. I have included images outlining each tab and its data.

The following images outline what our design should have looked like we designed it with the idea of it being simple easy to navigate and have an abundance of space to allow us to visualize as much data as we can because overall that's what the goal of the application is. We also designed it to be in either light or dark mode depending on your mobile device, this is a convenience change as most people prefer the option and could potentially save battery life. We detailed some of the things that might be included in the design however they may not end up in the design depending on time constraints, and the like. The android application is not a high priority as it only acts as a side-piece to the whole project so it is not in our scope currently and may not end up in the final rendition of our project.

Future integration's of this project could be include utilizing an Apple Application to our project, it would be coded in their native language named Swift it is an Object Orientated language that is very similar to Python. However when you code in swift the language automatically extends its support to their other devices and their respective operating systems so coding for a simple mobile application would automatically make it work for their wearOS systems such as Apple Watches. This is a very interesting application for our project since we can use the watch to connect to our project and it being used on the go is the most efficient way of using the Swift application.

Finally, it may be possible for us to extend this as a web application using the new Bluetooth apis built into the web platform. This would make it possible for us to have a single code base as the foundation for all of our apps. Of course, before we did anything like this we would first want to evaluate if an application like this would actually provide any benefit to the end user. In the vast majority of cases, the user will be interacting with the scope through their mobile device, so building out an entire web application may not serve much purpose.

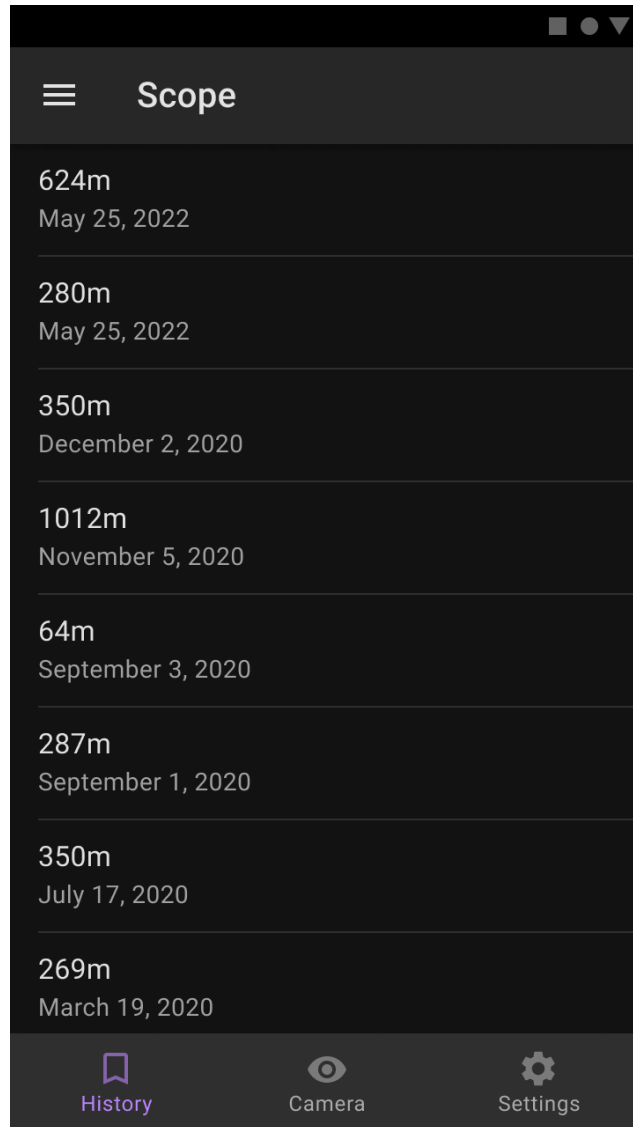


Figure 28: Application Stats Format

When the user first opens the app, they will be presented with this screen. The screen displays a list of items ordered by their date. When an item is clicked, a dialog will open with more information about the piece of data the user is interested in. This data will include the type of bullet that was used when the measurement was taken, the exact time the measurement was taken, as well as any user added metadata to that measurement.

The list will be a complete log of every measurement the user has taken. The data be stored on the users device and will not have any sort of cloud support. Each item will be able to be deleted, but the measurements will not be modifiable via the app. Along with storing measurements, we will also store errors that occur during a measurement. The errors can range from the power supply being too low to take the measurement, temperature errors in the case of the rangefinder being too cold

or how, and various other errors.



Figure 29: Application Camera Format

When the user opens this page on the app, they will be presented with a camera view if the scope contains a camera module. This camera view can either be used to view a live feed from the camera if active or it can be used to browse old recordings. Each recording will be displayed in the order it was taken by date. We will also be able to associate the recording back to a item from the users history.

These recordings will be stored on the users device. We have no plans to add support for any sort of cloud storage. Because the user may want to save their recordings on a different device, we will provide an option for the user to export all of their recordings. Because of the quality of the camera module, the file size will be relatively small, so this shouldn't be much of a problem.

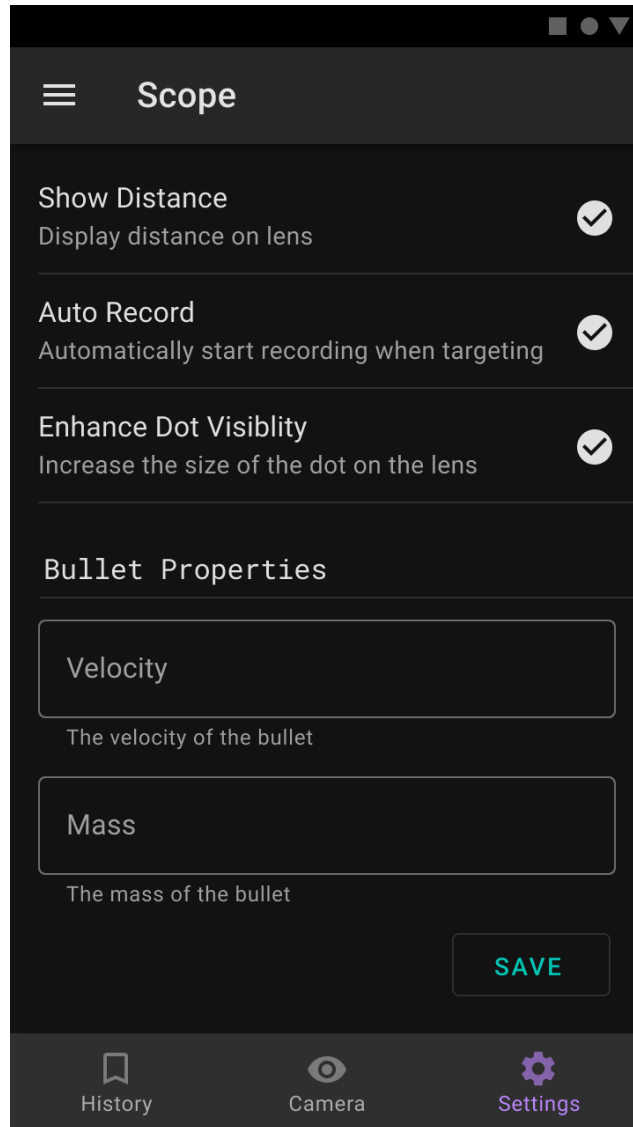


Figure 30: Application Setting Format

Figure 30 is the screen where the user will be able to configure several different settings for the scope. These settings will be stored on the users local device, but will also be synced to the microcontroller itself. This means that you will not be required to have your phone with you while using the scope. When the user makes a setting change, the app will tell the microcontroller about the change. The microcontroller will then update its current state and save the new state so it can be loaded later. Below is a brief overview of the available settings in the app.

General Settings

- **Show Distance** - This setting will tell the microcontroller to display the distance measured from the rangefinder onto the targeting lens. The distance will be display the last taken measurement by the rangefinder. We will dis-

play the distance in the upper corner of the display to prevent obstructing the users view.

- **Auto Record** - When enabled the camera will automatically start recording when the measurement is taken. There is also an option to set how long the recording should be.
- **Enhance Dot Visibility** - This setting will increase the size of the dot on the targeting lens. Of course this comes with the risk of decreasing the accuracy of the shot, which is why it will be disabled by default.

Bullet Properties

- **Presets** - This option will be a drop down list of different bullet types that have had their properties already set. The list will include the most popular types of bullets on the market. When an item is selected from the list, the microcontroller will be updated to use the properties of said item.
- **Velocity** - Velocity is a custom setting that will be used when there is not a preset bullet type defined by the app. This velocity will be the nominal velocity of the bullet which can be easily found online.
- **Mass** - Mass is also a custom setting that will be used when there is no presets. The mass will be used withing the bullet drop calculations much like the velocity. Also like the velocity, the mass of a bullet can quickly be found online.

While most settings will automatically be applied and synced to the microcontroller, the custom bullet type must be synced by using the save button. This will prevent the microcontroller from receiving partial information which may lead to inaccurate results. When the settings are saved, all options will go through a validator to ensure that we are passing correct information to the microcontroller. Any validation errors will be displayed to the user and they will be required to fix them to save.

5.8.13 nRFConnect

For the purposes of easy of understanding I will be talking about the nRFConnect application which is an exact copy of all the functionality, the basic understanding of the application is that it connects to low energy Bluetooth devices. As most if not all smart phones comes with Bluetooth having our only worry is being able to connect to a low energy Bluetooth device which is not the same kind of connection. This app will be a good backbone to fall on if our app does not work as intended. Android Studio is not a familiar development environment for members of this team so having a functioning app that we know works is very good for testing.

5.8.14 GitHub

GitHub is a version control website that is widely popular in software design, its primary function is version control however it exists to supply many more elements that allow different programmers to collaborate on various different projects. In our project we will be using GitHub as tool to have all programmers on the same coding data. Essentially what that means is that every programmer will be able to update a piece of code and have it update on every programmers personal computer this allows us to be on the same page and have no issues regarding looking at the same piece of code. However for the changes to be live and for the other programmers to receive them each programmer would need to push these changes onto the server after completing their tasks and provide some description of the task that is being completed, and prior to starting new tasks each programmer needs to pull from the server to have the update pieces of code on their local machine. The version control aspect of GitHub allows us to rollback changes after they are pushed as every version of the code is saved onto GitHub servers for our convenience and in the event of a fatal error happening we can roll back our code to a previous version.

5.8.15 Software Revision

As of the second semester of senior design we have made considerable changes to the initial plans as there have been unforeseen challenges, as software is very portable between languages the overall structure has not changed. However the elements outside of the software structure have, the first is the micro controller. Since the Portenta was considered a development board we could not incorporate that into our design as per instructor rules, hence we switched to the arduino uno for testing purposes and printed our own board to our uses, because of this change we had to switch to C++ in the arduino environment as Micro-python was not compatible with our board. This has worked out quite well thus far and since we had our Portenta H7 we decided to utilize it for a machine learning module if our scope allows it. The overall software design is identical the original plan barring some intricacies in programming between Python and C++, however it worked to the specification that is detailed in this article.

In addition to that we decided to utilize a full-stack approach to our app design, as the Android studio development proved too difficult for this team to implement, as it involved USB permissions which was a topic the team was not versed in, but since we have changed our design this allowed us to achieve a more stable and more feature packed app design as we were not focused on the intricacies of Android Studio and Instead we implemented features in our scope such as adjusting the bullet parameters and utilizing the display through the micro-controller and app. This app utilises Node.js to access the dynamic library files included with our I2C/USB combo converter to connect and pass information through to micro-controller, and the front end was designed using Vue.js as it can ported to a web-app if we deem

it possible considering USB permissions and all it entails.

5.9 Design Summary

A design aspect for the construction of that scope that we are going to utilize is that when we construct the scope, we will assemble the pieces under a chemical hood that is purged with argon gas. Purging the assembly area with a gas is not a new concept in the commercial optic world, however the industry standard is currently nitrogen. Both gases attempt to accomplish that same objective, which is to make a more consistent environment inside of an airtight system, and there are pros and cons to each of them.

Having the gas inside of the scope allows for a more inert environment, one that will help to reduce fog or condensation buildup that would hinder the users vision through the scope. Using high quality optics that provide clarity would be a useless point if the scope is not properly sealed to prevent against environment effects. Nitrogen is the industry standard as it accomplished the goals while still being economically viable to use in most products. Nitrogen also has a refractive index of $n = 1.000298$, which is small enough and close enough to air that it will have no substantial impact on the optical system. Argon is interchangeable as its refractive index is very similar, with $n = 1.000281$, there would not be any noticeable differences between usage of the gases. Argon is also more inert of a gas than nitrogen being that it is a noble gas, meaning that it will not react easily with water that manages to get inside the tube, and will not react to a temperature difference on the outside of the tube. Also, being a larger molecule than nitrogen, any leaks in the housing would have an added level of resistance as the leak would be slower.

Most companies use nitrogen as it is a more available gas but also because it is more cost effective. The benefits of either gas are similar, however, so the main reason one company uses argon over nitrogen would be for the marginal benefits that would still add to a better product, though commercially the price would reflect the increase in quality. The main reason why our group is going to use argon gas is because we have access to it, as it was a gift for the project, so we do not have to worry about extra costs. Even though the argon was a gift, we originally planned to flush the system with nitrogen to achieve all the same goals and attempt to keep up with industry standards.

We designed a complete circuit that will manage all of our power and bridge the gap between all of our components. With this circuit we have everything we need to interface our microcontroller and targeting lens, manage our power and supply power to both the microcontroller and rangefinder. All of this is on a single board which should be small enough to fit on the scope while keeping the entire design slim and lightweight.

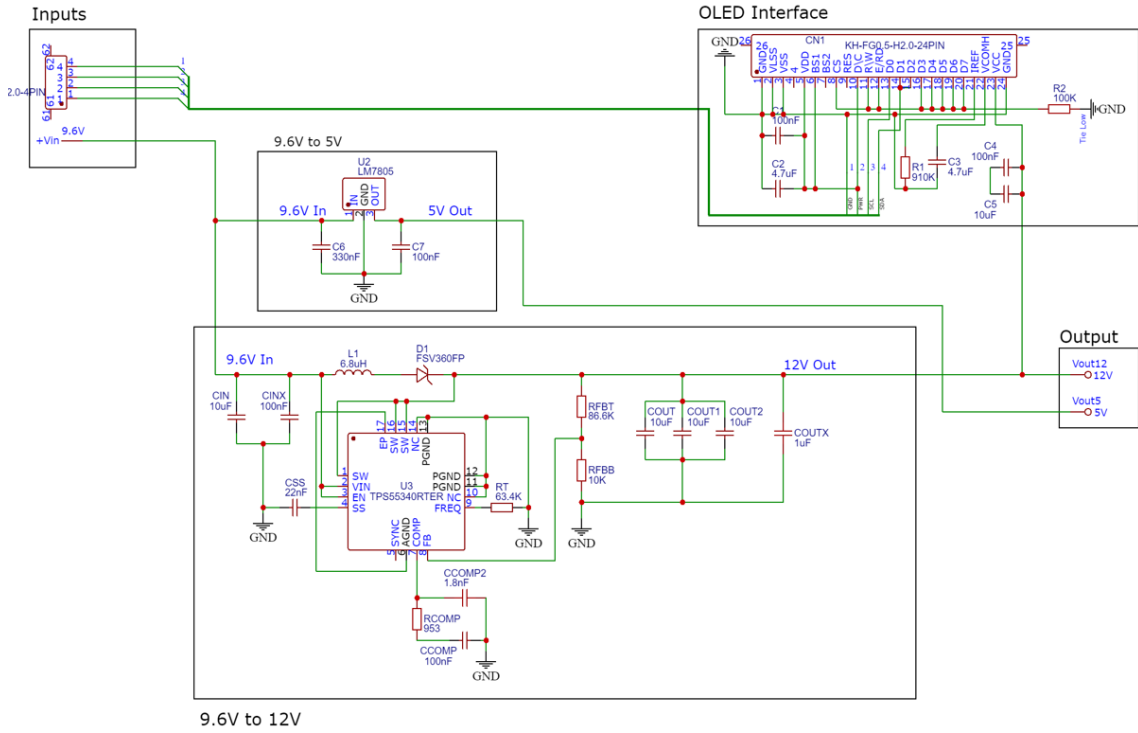


Figure 31: Full Schematic

Above in figure 31 is the schematic in full. This includes all the inputs, outputs, interfaces, power booster, and the voltage regulator. The total cost of the board will end up being less than \$10 in terms of components. It was important for us to make our inputs and outputs clear while keeping the design clean.

Finally we designed an app that will allow us to control the functionality of our micro-controller. This is important as there is no way to directly interface with the micro-controller other than plugging it into a computer via USB. So this app will allow us to configure settings, gather data, and bring features to the scope that would not have been possible without it. Some of these features include recording firing sessions, updating the dynamics of the bullet type, and configuring the display.

6 Overall Integration

This section will cover the PCB designs, the ways all the systems integrate with one another, as well as the testing of the subsystems and finally the entire system as a single unit. The first half will cover the PCB design for our primary circuit board. This board will contain both our circuit for our power supply as well as our the circuit we use to interface with the targeting lens. In the next section we will cover how we plan to test each system to ensure it works like expected.

6.1 PCB Design

When designing the PCB, we wanted to keep it as small as possible to keep the overall system compact. We would like for it to be comparable to the size of our microcontroller. Our microcontroller is the Portenta H7 which is 66mm x 25mm. This is a good size to fit within the size constraints of our project, so if possible the PCB should be the same size or smaller than our microcontroller.

To design our schematics and PCB, we used a free program called EasyEDA. This software is similar to the likes of EAGLE, which is a popular PCB and schematic design tool. With this software, EasyEDA, we were able to go directly from schematic to the PCB design view. It will also produce a Bill of Materials with links to all the parts we need to order as well as the total cost of the components.

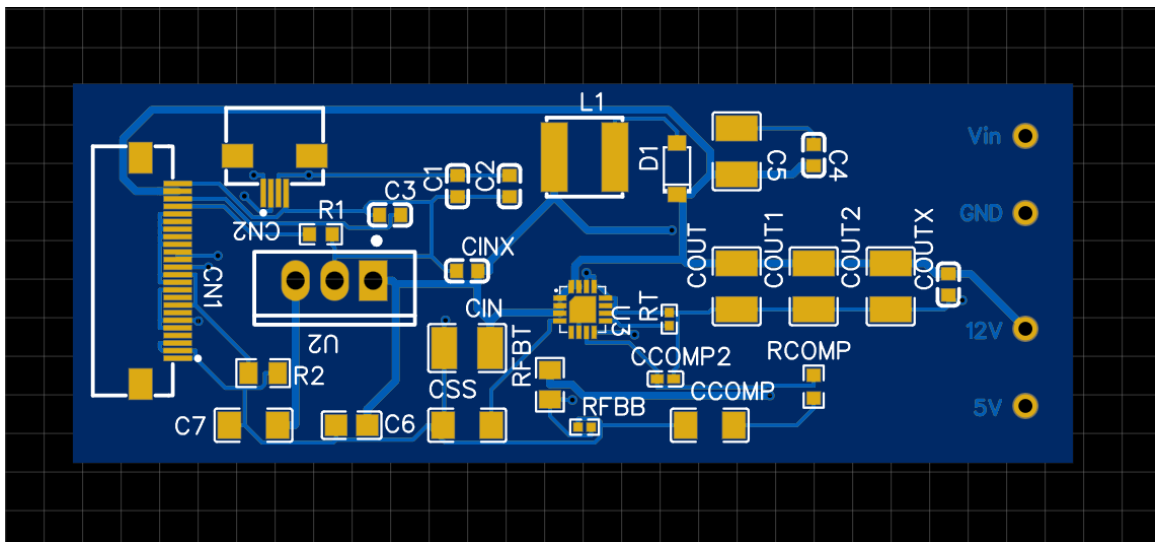


Figure 32: PCB Design

Above in figure 32 is the initial PCB design. Our total component footprint is fairly small, so we are able to exactly match the size of our microcontroller. We have two different ports on the board. On the left side of the board is the 24 pin connector that will be connected to our targeting lens. On the top of the PCB is an I2C connector

that will connect directly to the microcontroller. These are the only pin connectors that will be on the board.

As for pin outs, we will have a total of four pin outs. These pins outs will be located on the right side of the board. The first pin will be used as our ground for the entire board. The second pin will be used as our voltage in. Our battery will connect directly to these first two pins, the negative lead going to ground and the positive lead going into Vin. The next pin out is the 5V power that will be used to power the microcontroller. Finally the last pin is the 12V pin, which we will use to power the laser rangefinder component.

To connect the battery to the PCB we will use a female Venom Tamiya plug connector. The leads will be soldered directly to the GND and Vin pin outs on the board. The other end will be used to connect directly to the battery. We want to use this type of cable to the user can switch out the battery easily; allowing the user to have extra batteries on them and use a separate charger to charge the battery.

When designing the PCB we had to decide what type of components we wanted to use. We were given two options, through hole components and surface mount components. While this was a pretty easy choice to make for the final design, there were several considerations we had to make otherwise we would end up doubling our work and spending more money than we want.

6.1.1 Through Hole Components

Through hole components are component that are mounted on the board using hole drilled into the board. Each component has a footprint which indicates where the holes are to be drilled. Then to mount the component you place the leads of the component through the holes that were created for it and solder them in place.

There are two primary benefits to through hole components. The first one is testing. Mounting these components on a PCB yourself is quite easy. The most difficult part would be soldering these components in place, but even that is easy enough to do with a little practice. One reason we might want this is to make it easier to find and fix problems with our circuit. Not only would we be able to more easily spot problems and bad components, but fixing the components ourselves would be easier. Another advantage of through holes component is that they are more capable of putting up with mechanical stress. The connection formed by the solder between the leads and the PCB produces a very secure connection that is unlikely to become undone due to environmental stresses. In the use case of our scope, this seems like something we might want, but it is unlikely that it would outweigh the benefits of the surface mount components.

While there are many advantages to using through hole components, there are also quite a few disadvantages. One of the first disadvantages is the size of the components. Typically the size of a through hole component is several times bigger than its surface mount counter-part. This means having a compact design would be

difficult. In fact, the additional space we would need to support these components might make the board more prone to being hit against other objects due to the extra space it would take up. Along with being more bulky, the through hole components are not as common anymore. Most modern devices try to be as small as possible, this means most of the electronic parts being manufactured are smaller surface mount components. It would be more difficult to find all of the parts we need for our board as through hole, and we would like to avoid mixing through hole and surface mount components as much as possible.

6.1.2 Surface Mount Components

Surface mount components are components that are mounted on the surface of the PCB. Rather than mounting through holes drilled into the PCB, they have metallic pads on the PCB and the component that are then soldered together making the connection. Much like the through hole component, each surface mount component has a footprint indicating where the pads should be placed on the board and how it should be spaced.

Today most electronics are built using surface mount components. In fact the Arduino Portenta H7 microcontroller we are using for this project is using all surface mount components. The biggest benefit of a surface mount component is the footprint. In general these components are very small, several times smaller than the through hole components. This is very beneficial for us as we are trying to keep our design as compact as possible. While in some cases, surface mount component may be less stable than through hole components, they are able to withstand shakes and vibrations better than through hole components [16]. This is ideal for our case as the stresses the PCB will be put under will mostly be the forces of the gun recoil. Another reason we would want to use surface mount components is their availability. Nearly every component we want is available as a surface mount component. The only component we needed that was not available as a surface mount component was our voltage regulator. While it would have been preferable to find one that is, at the time of writing we haven't found one. Finally the last reason for preferring surface mount, is that it is generally cheaper and faster to produce a PCB board with surface mounts rather than through hole. Considering our constrained budget and time, cheaper and faster is important to us.

For a final product it seems that surface-mount components are better in just about every way. But, during development, it can introduce some inconveniences. The first inconvenience is the difficulty in replacing components that may have gone bad or were the wrong type. With through-hole this process is easy, just unsolder and replace. But, with surface-mount components, they components are significantly smaller meaning it will require more skill to place the component correctly. Secondly, when soldering a through-hole component, you can just place some solder on the leads of the component and the hole on the PCB and you are pretty much done. With a surface-mount component, you must place solder on the pads of the component which can be quite difficult to do, especially if you are not experienced

in soldering. So while it may be easier for a machine to solder on a surface-mount component, it is much more difficult for a human. Another difficulty introduced by surface-mount components is taking measurements across certain components. For example, if you wanted to measure the voltage between two components, with a through-hole component you can just probe the leads on the bottom of the board and take your measurements. With a surface-mount component, the leads will usually be covered by the component itself, making it impossible to take the measurement.

For these reasons we decided to start with ordering the PCB using mostly surface-mount components. In the future if we run into too many problems with the board, we may order a version that uses only through-hole components so we can test the board. But, we were able to do quite a bit of testing via our schematic and PCB design tool. So for this reason we will start out with just a surface-mount board.

6.1.3 Layout

When designing the board it was important to layout components in a way that made sense. It took several iterations before we landed upon a design that we were happy with. The tool we used to design our PCB would automatically take all the components and their footprints from the schematic and place them in the PCB designer. The first step to designing the PCB was to set the board size. We initially set it to the default size, which EasyEDA defines as 1.5x the sum area of all our component footprints. While we were able to make this size work, it wasn't really what we wanted in the end. So back to the drawing board.

In our next iteration, we took the board size of our microcontroller and decided that having our PCB be the same size as our microcontroller would make things easier. Because of course, designing our housing around components with the same dimensions would be easier than designing around components that had vastly different dimensions. So, for the board size we went with a size of 66mm x 25mm. Luckily enough, this was more than enough space for us to place all of our components in the way we wanted. In addition to having to place our components, we also needed to have pin ins and pin outs for our power lines.

We made several attempts to layout our components on the new board size of 66mm x 25mm. There were two components that needed to be in a specific location. The first one is our 24pin connector, this connector needed to be at the edge of the board as the connector is parallel with the PCB. Placing it anywhere besides the edge would block the cable from being able to connect to the board. The other component was our 4-pin connector for our i2c connection. This component also had the same issue as the 24-pin connector, so we also had to place it on the side of the board. While this was good, in our first couple iterations of laying out the board, we didn't put much thought into how the rest of components themselves should be placed. So we ended up just placing them where ever they would fit.

This proved to cause trouble later on when we would try to route our components. So to fix this, we decided to place components that were close in location on our schematic close to each other on the PCB. This proved to work out much better than our previous attempts and made it easy for the traces to be placed. There are probably better methods out there than how we did it here, but for our purposes this seems to work like expected.

6.1.4 Traces

Placing the traces was probably the trickiest part of designing this board. It took us several attempts and a couple board redesigns to get our traces placed properly. As mentioned above we went through several different component layouts, and this is what affected the placement of the traces the most. On our first attempt, our components were randomly placed on the board without much thought. We both manually placed our traces and used the EasyEDA built-in auto-router. Eventually with the randomly placed components, the board was able to be routed but then we noticed that we would need to increase the thickness of some traces. Traces that carry a lot of current need to be thicker than the other traces and have a larger gap between them and other traces. So after applying these updates, we were no longer able to find a route that would work. This was when we had to redesign our board.

After we spent some time redesigning our board, as discussed in the previous subsection, we once again attempted to route our board. This time we were able to successfully route every component with our new thicker traces. We then had to place our pin ins and pin outs. We initially thought that these were components that would transfer from our schematic to the PCB, but we were wrong. So using the PCB design we had to manually place these pin ins and pin outs. After that we were able to set their net labels and run the traces to them. With that, our board had all of its traces run and the only step left was to order the board, components, and test that it worked like expected.

6.1.5 PCB Vendors

To order our PCB we needed to choose a vendor to order from. There are many vendors out there, so we will narrow down our list to just three. The first one we will take a look at is Sunstone Circuits the second one is JLCPCB, and finally AIPBC. The main factors we will want to look at when ordering our PCB will be location, cost, and assembly. The location is important as the closer the manufacturer the faster we can get the board shipped. Cost is another important factor as we may need to order several boards, so the prices can add up quickly. Finally assembly is important, if we can find a manufacturer who will assemble the board that would make them preferable as mounting surface-mount components can be challenging. These are all considerations that will be taken into account before ordering the PCB from a vendor.

Sunstone Circuits

- **Location** - USA. This is good as it means the circuit board will be able to ship to us faster.
- **Cost** - Expensive. Ordering a PCB here would cost us around \$15 per board.
- **Assembly** - They will only print the board for us, we will have to assemble the board ourselves. This is not ideal as the board uses all surface-mount components which would be hard to mount.
- **Minimum Order** - 5 Circuit Boards

JLCPCB

- **Location** - China. Because it is in China shipping the PCB might take a while.
- **Cost** - Cheap. Ordering a single PCB here costs us only \$4.
- **Assembly** - JLCPCB will send a fully assembled PCB at no extra cost for assembly.
- **Minimum Order** - 5 Circuit Boards.

AIIPBC

- **Location** - China. Like JLCPCB, ALLPCB ships from China meaning it might take longer for us to get our board.
- **Cost** - Somewhat Cheap. Ordering a single PCB here costs us \$6.
- **Assembly** - ALLPCB provides PCB assembly, but unlike JLCPCB, they do not provide it for free.
- **Minimum Order** - 5 Circuit Boards.

The first option looked like a good choice due to being close to us, but the boards themselves are much more expensive and the actual shipping was not much faster than the other options we looked at. ALLPCB was similar to JLCPCB in many ways, but the boards were more expensive and assembly is not provided for free. For these reasons we did not choose Sunstone Circuits or ALLPCB. We decided to go with JLCPCB for several reasons. Compared to the other PCB manufacturers, they were by far the cheapest and provided PCB assembly. We are also able to buy all the board components from them as well. Finally, JLCPCB is tightly integrated with our design tool, EasyEDA. Because of this we are able to click a single button and go directly from PCB design to ordering the PCB without having to manage gerber files or BOM files ourselves.

When ordering the PCB we were given several options about how we wanted it built. The first option was the type of base material we wanted. We would have to choose between FR-4 and Aluminium. FR-4 is a fiberglass material. This is a pretty common base material to use for a PCB. Aluminium is better for are more durable and have better heat dissipation. Since heat is not of much concern to us

and our PCB will be secured in a housing, we went with FR-4 as it is cheaper than the Aluminum base. The next option was the assembly, the manufacturer allows you to choose how many boards you want them to assemble. We decided to have them only solder 2 of the 5 boards as this will allow us to make our own adjustments if there are problems with the assembled boards they send us.

6.2 Testing

This section covers the testing requirements and procedures to test that the product is functioning as expected. The first section will give a high-level overview of the various tests that will be run as well as the reasons for these tests. In the following section, each test that is run will be covered in more detail as well as describe the requirements for each individual test.

6.2.1 Overview

For the testing of our scope, we plan on doing it in stages before we get to a live fire test on a range. Making sure that the scope acts as intended before we get to what would be a field test would allow for us to save time and resources as we need specific spaces to be able to properly test the design to its full potential. First, we would want to make sure that the rangefinder is accurately gathering information, and that the pulse width of the beam is not being miscalculated for the distance calculations, as dealing with nanosecond response times could greatly throw off the results.

The distances first used as a basis to ensure accuracy and precision of the product will be known and short range. This will also give us a starting idea of what our margin of error is going to look like. Although a few inches at a starting distance of 100 yards would not have any great impact on the trajectory, a few feet at a thousand yards could cause missed shots at great distances. Because of the acceleration of gravity on a bullet at great distances, an error of 1-2% could miscalculate to a foot vertically, which could cause a miss for a hunter or a soldier. Although we might not have the capability to correct the error of the rangefinder that we plan on implementing, as it is a preconstructed unit, it would fall on us as the design team to denote that the error exists to ensure that the user is aware that greater distances lose their effectiveness, and there is an increased chance to miss a shot. To get an accurate margin of error, we plan to test the rangefinder at increments of 50 meters up to 1000 meters and will denote the measured distance compared to the actual distance along the way. We plan to repeat these measurements at least two times to find an average error in the system. The prediction is that there will be a percentage error as opposed to a given distance error, meaning that the distance measured will grow larger with each increment.

The second part of the preliminary tests to ensure that the product will behave as intended is to test the LEDs once they are installed on the reticle. The results we will

be looking for are that the correct LED will illuminate for a specific distance. We can have a spreadsheet that denotes what distance would illuminate which diode based off the bullet drop calculations. With this information, we can safely assume that the coding is reacting properly to ensure communication between the rangefinder and the Arduino board. Therefore, it is important that we ensure the accuracy of the rangefinder first, as it allows us to narrow down the problems that could possibly arise during the testing phase. If we are not systematic in our search for flaws in the system, we would not be able to narrow down the main problem and would not be able to fix the system. If we are sure that the rangefinder is working properly, the testing of this section of the design would not only be easy, but also very concise. Using the spreadsheet, we would be able to count the LEDs and ensure that the proper one is illuminated. Because each test would only take a few seconds, this one would easily have the most trials while being the most accurate. However, as this aspect of the design will be implemented by our group and dependent on a system that we cannot adjust, this aspect cannot have any margin of error. We plan to run at least 100 different trials of this section to see if there is any percentage error for a miscalculation in the coding. Although the coding will not have an error that we could easily adjust, we would attempt to get this percentage as close to zero if possible. If we managed to go 100 trials without a single error, we would play it safe and rerun the trials to ensure that the coding responds perfectly. If we do have a margin of error that relates to the coding that we cannot manage to fix, we will denote it for the sake of the user.

If both of these aspects are in perfect function order, or at least functioning with an appropriate level of error, we could finally take the product out to a long distance, outdoor range to conduct a live fire test. The reasons why this step is done last relates to not only that we would have to many variables to account for if we went straight to practical testing, but also live fire testing incurs additional costs to the overall project due to the current price of ammunition. For our live fire tests, we need to be consistent with the ammunition being used, whose selection will also be done meticulously, as we must use a cartridge that is readily enough available that we can do testing with at least 100 rounds (which is actually quite low for an industry standard) but also compatible with what would be a NATO standard round.

The projected round that we will use will be a 5.56x45mm NATO round, with a bullet grain weight of either 55 or 62, and a muzzle velocity of around 2900-3000 feet per second. The brand will be selected closer to time of testing, though our stretch goal would require us to test multiple different brands. The testing for the section will be similar to the dry fire test from the last section, where we would aim at a target and go based off of the illuminated LED, though we now must account for the distance off of the target that we are firing at. If the product reacts as intended, the bullet will strike the target within a radius of 6 inches as any distance. The error that could at this stage of testing, however, we predict to mostly be human. At greater distances, the LED will be large compared to the apparent size of the target, which can cause the tester to naturally aim off the center of the target. To minimize this error, we will set up the firearm with a bipod on a table, as well as

fire from a seated position so that natural human movement will not cause as great of an impact on accuracy. This testing will take the longest of all the tests, as the measurements taken after each shot will require us to travel to and from the firing line to the target each time, as for obvious safety reasons we will not allow anyone front of the firing line when the shot is being taken. This test will also take place at increments of 50 yards up to the maximum distance that we are able to test to, which will be dependent on the location we are able to use. Most outdoor ranges in Florida are not long enough to accommodate 1000 meters, so we will use as long of a range as we can find and compile the data to predict the behavior of any distances we could not test for. The conversion of distances is intentional as well, how we are testing in increments measured in yards, though we plan to have a maximum of at least 1000 meters. The reason for this is that standard measurements for the rangefinder, which determines the maximum length that our product can be utilized to the full extent, are recorded in metric, while most shooters in the market that we plan to introduce our product in will be using standard measurements. Our spreadsheets will have both denoted, though the conversions will be calculated, and the tests will be measured in standard.

6.2.2 Software

Software testing is done in mainly two ways: static and dynamic testing. This project will be incorporating both in the testing portion, but first I will be talking about the key differences between static and dynamic testing. Static testing is the kind of testing that makes sure that the code from the compilers or interpreters point of view is correct so this involves syntax, and exceptions being handled because a piece of code can work but not perform the actions required we need to incorporate dynamic testing. Dynamic testing, this kind of testing involves the usage of test cases, which is key in performing up to par in real world scenarios. I will be outlining the key test cases in following paragraph.

The first and most important test case is the receiver and emitter we will be doing this by measuring a distance and checking it through code, this is quite frankly the only way we would be able to test that the range finder works effectively in a timely fashion. Another way we can test this is by using two different range finders however this would cost us more money but could be a viable solution. Next is to test our display this is done through using an array of all alphanumeric characters, we would use this array to make sure we can display all values that is required. We will also be using a gradual addition approach in displaying them, so we will be displaying the letter "A" individually to begin with, then displaying "AB" and so on until we have covered all the elements of the array, once we have we will be working in the reverse order to make sure we can remove characters if needed so then we remove the character "A" then "AB" and so forth until our display is clear of all values. Next we will be testing our buttons to check the state that this will be done using the LED's on our board. If the state of the button is "set" then we will be displaying a green LED if the state of the button is set to reset we will be display a

red LED, the testing portion of the buttons will just check the code functionality of the buttons not the physical button itself as that button is being checked in our boot.py file. This would conclude a in-house dynamic testing event, however we will be conducting real life testing in various locations to confirm these test results.

Another aspect of the testing will involve testing the Android app for the scope. We will be looking for several aspects when testing. The first thing to test is the connection to the micro-controller over a Bluetooth connection. To verify the connection we will send a string back and forth between both the app and the micro-controller ensuring that we receive what we are expecting and that our data is not mangled. The next thing we need to test is that we are properly receiving data from events that occur on the micro-controller. These events will be a payload of data defining what type of data it contains as well as a timestamp of when the data was sent. We can test this by taking a measurement on the rangefinder and checking for a result on the app. Next we will need to test that the app is able to send data to the micro-controller to update the state on the app.

For the purposes of our project it will be difficult to test different ranges as that involves the usage of a live weapon which we do not have easy access to and even in the chance that we do utilizing a weapon in ranges of 1 kilometer would be difficult, because of this I will be utilizing data collection and machine learning to predict our outputs. The features we will be using are the features used to calculate the bullet fall off as well as time of day, current weather and battery percentage. These are all features that would contribute to how the scope operates and therefore can and will be used to predict different how our project would work under various conditions. Our target vectors in this model is the image that will be projected onto the OLED panel. This way we can predict how our project will function with reasonable accuracy without the need to go and test it which comes with a host of unknown variables such as weather and human error, another reason for this is because our project is not completely capable of working under every condition possible so having some sort of idea of how it operates before introducing the unknown variables will allow us to work towards creating the main goal of the project instead of accommodating the unknown variables.

Our model will consist of a convolution neural network or CNN for short, this type of model can predict images and it is fairly common in the machine learning industry. For our project we will be using Tensor Flow as our main module, using this we will construct a model that looks like the following image. This model will be run on a separate PC and output to confirm test results, which would allow us to save time in real life testing.

Main Graph

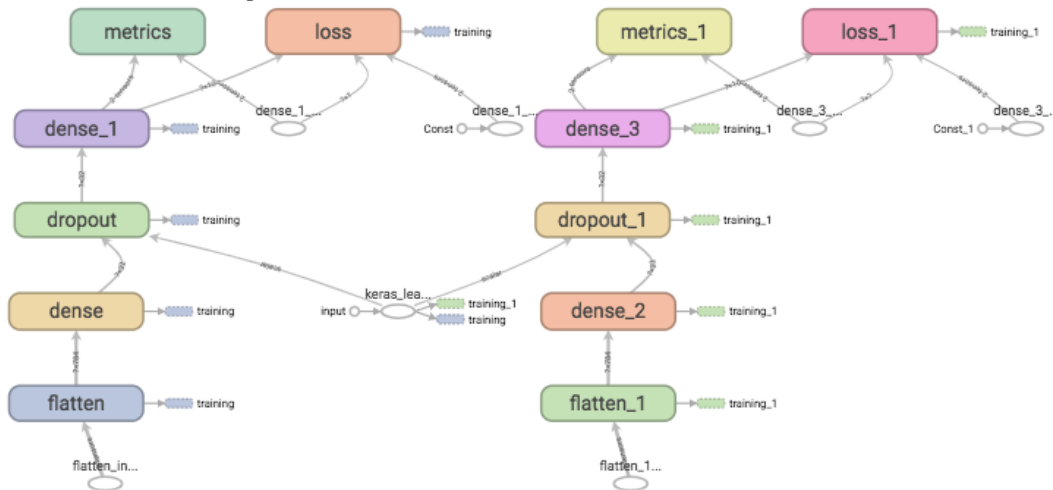


Figure 33: Tensor Flow Model

6.2.3 Laser Rangefinder

The laser rangefinder is arguably one of the most essential parts of the project. So it is important to ensure that this works and it is accurate to a certain degree. The range finder itself comes with several error codes that will enable us to test it in different conditions. We need to ensure we know what each of our expected outcomes are in each testing environment, this means both low-light and high-light environments. Weather may also play a big factor in the measurements that the rangefinder takes, so we will need to test various levels of rain which might change the accuracy or distance that we can measure. Finally we will need to test and handle low-power levels which may prevent us from taking a proper measurement.

Below in table 21 is a list of error codes provided by the laser rangefinder. These errors will help us define some of our tests. More specifically, error codes Er01, Er05, Er06, Er07, Er08, and Er09 will define a few of our tests. Other errors such as hardware failure, hardware malfunction, and temperature errors won't be tested as they are difficult to test and are not common scenarios we will run into. The error code are provided by the manufacturer via their data sheets. We will be able to receive these error code via the data lines connected from the rangefinder to the microcontroller.

Error Code	Description
Er01	VBAT Too low, the voltage should be $\geq 2.0V$
Er02	Internal module error
Er03	Module temperature is too low ($< -20^{\circ}$)
Er04	Module temperature is too high ($> +40^{\circ}$)
Er05	Target beyond the measuring range
Er06	Invalid measurements
Er07	Background light is too strong
Er08	Laser signal too weak
Er09	Laser signal too Strong
Er10	hardware failure
Er11	hardware malfunction2

Table 21: Rangefinder Error Codes

We will start by testing the accuracy of the rangefinder. To do this we will set out targets at known distances. These targets will be set from the minimum distance we are able to measure and increase in 100 meter increments all the way to 1500 kilometers. Each of these distances will be recorded and compared to the known distance. With this information we will be able to determine the accuracy at each of these distances. Not only will this give us a general accuracy, if we are able to determine a certain trend when it comes to measuring, we may be able to compensate for these errors in the software.

The tests for the weather will be similar to the previous test, but it will be carried out while it is raining. Due to the fact that we have limited funds and are not able to control the weather, we will have to wait for a rainy day to carry out this test. Our high-light level test will be done on a sunny day in the middle of the afternoon when it will be brightest, this test is important as it will determine how well this scope will work in a condition that will happen quite often. Our low-light level test will be done in both the evening and night so we can test ideal conditions. The success of these tests will not only be determined by the accuracy, but also the error codes returned by the rangefinder. Error codes Er08 and Er07 will be the codes we are looking for during these tests.

Finally we will test the range and low-voltage capabilities of the rangefinder. This test will first start out by testing the maximum range we can measure in ideal conditions at full battery charge. We will then reduce the total charge of the battery, which will in turn reduce the voltage of the battery. Around every 5% drop in battery

we will take a measurement and watch for the accuracy as well as the maximum distance we can measure. Once the battery is low enough we will get a error code of Er01. This error code will allow us to indicate to the user that they battery needs to be changed or recharged.

6 .2.4 Targeting Lens

To test the targeting lens, we will need to ensure all of our controls are working as expected. To do this we will have to make use of our microcontroller to send the appropriate commands. This will be tricky as we will have to be sure that our code works before actually being able to test it. We are fortunate enough that the company that produces our display has some open-source demo code that is known to work, so we will be able to use this as a baseline that we know works. Next once we confirm that our connections are working like expected, we will be able to test different commands. In table 13 there is a list of commands that we will be able to test out.

To start our tests, we will want to ensure that every pixel on the screen is working and that we don't have any dead pixels. To do this we will turn on every pixel on the display. We can then visually inspect the display to determine if there are any dead pixels. After confirming that there are no dead pixels will we adjust the brightness of the display through several levels. The reason we want to be able to adjust the brightness of the display is due to the fact that the screen will need to be very bright during the day to be visible. But at night, having the display be too bright may hurt the users eyes and make it hard for the user to see what is behind the display.

The next test we will need to preform is calibrating the display. The center of the display may not necessarily be the center of the display. So when calibrating we will need to tell the microcontroller what the center of the display is. Once the display is calibrated and we know the center, we can then check how accurate the targeting lens is. To do this we can calculate where the scope dot should be and compare it to where it currently is. If this position is off then we know there is either a bug in our code or the display was not calibrated correctly.

6 .2.5 Power Supply

For the testing of our power supply, we plan on using a multimeter to test out the output voltages of the batteries when they are fully charged and the change in its voltage as the battery starts to drain. Checking the output voltage of the 9.6 V battery will make sure that the battery is outputting the correct voltage that it is intended to on normal working conditions. In normal working conditions, the 9.6 V battery should output 9.6 V, but we need to remember that this number may fluctuate for various reasons such as physical damage, bad connection with the multimeter, overcharging, etc.

We will be connecting the probes of the multimeter to the corresponding wires of the

battery to test the voltage level. The red probe of the multimeter will be connected to the positive terminal of the battery or the red wire. The black probe will be connected to the negative terminal of the battery or the black wire. Making sure the probes are connected correctly will give us an accurate voltage measurement.

For the first part of the power supply, this 9.6 V battery will be stepped down to 5 V. To make sure the voltage regulator is converting the voltage correctly from 9.6 to 5 V we will measure the voltage of the output of the voltage regulator. We will connect the correct probes to the output pins of the voltage regulator and it should read 5 V.

For the second part of the power supply, this 9.6 V battery will be stepped up to 12 V. To confirm that the voltage regulator is working as intended we will measure the output voltage of the pins. We will connect the corresponding probes to the output pins of the voltage regulator and it should read 12 V because this 9.6 V was stepped up to 12 V.

These tests will only take a couple of minutes per section, so it is not a time-consuming procedure. However, it is a very important test to make sure it is done right as having the wrong voltage being connected to the MCU or the components could permanently damage it.

7 Administration

In this section we will cover the administrative parts of the project. This includes the division of labor, the tools we used to communicate and collaborate, the software we used to design the project, the budgeting and financing of the project, and how we scheduled our time. We won't provide any technical info here, this section just provides some general information about what went on in the background of this project.

7.1 Division of Labor

The following section will cover how we divided up the tasks for this project. Although each member has their own speciality, we often have to work closely and design in fields that are not our speciality since everything is intertwined and connects.

7.1.1 Scope Design

The scope was primarily designed by Collan Dimitri. Aside from general mounting design and targeting lens, this was outside the capability of the rest of the members in the project

7.1.2 Rangefinder

The rangefinder is a prebuilt module, so it has not been designed by any one team member. Though much of the research has been done by Jacob Clevenger. Due to the interactions with the microcontroller, Ahmed Al Busaidi has also worked closely on using this module.

7.1.3 Power Supply

The power supply for this project has mainly been designed and built by Jose Vasquez Dickson. Jacob Clevenger also collaborated on this design as they will be on the same board.

7.1.4 Targeting Lens

The targeting lens has mostly been designed and developed by Jacob Clevenger. Because Ahmed Al Busaidi is the programmer, he has collaborated closely with Jacob as the interactions with the microcontroller and the design of the targeting lens are closely intertwined. Jose Vasquez Dickson was also included on portions of this design as he was the primary designer of the power supply, so it was necessary for him to know certain aspect of the design.

7.1.5 Software

Software design and Implementation primarily done by Ahmed Al Busaidi, however Jacob Clever assisted with minor tasks when needed. Because this design interacted with both the laser rangefinder and the targeting lens, it will necessary for the computer engineer to work together with the electrical engineers on some aspect of this system.

7.2 Communication Tools

The following section describes the tools we used to communicate with each other as well as design and develop our project. These tools include communication, file sharing, and collaborative document editing.

7.2.1 Discord

Discord was the primary means of communication between team members. Discord is a free communication services that allows you to create "servers". These "servers" allow you to send text messages as well as talk over VoIP. A "server" allows you to create channels and orginaize your team in a sensible way. When it comes to meetings Discord has a built in feature for scheduling meetings and finding out who is available to join.

7.2.2 Zoom

Zoom was the primary form of communication with the professors at UCF. We were able to get feedback on our documents, designs, and ideas. The remote nature of this platform allowed us to stay safe during COIVD-19 times and was more convenient than meeting in person.

7.2.3 Overleaf

Overleaf was the platform this document was written. It allows you create documents based on the L^AT_EX programming language. This allowed us to have a single place where we could organize our document and share figures and tables with each other. Overleaf also has a built-in communication tool allowing us to write messages to each other while collaborating on the document. Along with writing messages to each other we can write comments in specific locations on the document if we think it should be modified.

7.2.4 OneDrive

OneDrive allowed us to share our designs and important documents with each other. This was very useful when it comes to sharing larger files as well as keeping

a history of all of our documents. This also served as a service to backup our files in the case of our local copies being corrupted.

7.3 Budgeting and Financing

This project will be completely financed by members of the group and each item that exceeds a reasonable value will be split amongst the members through a money sharing application such as Venmo, Paypal or Cashapp.

Table 22 is a list of the parts required for each module of the project. This list will only include major parts as listing off each individual resistor, capacitor, etc would not be appropriate here.

Part	Quantity	Cost
Portenta H7 Microcontroller	1	\$103.40
Crystalfontz Transparent OLED Display	1	\$26.08
Momentary Switch Button	1	Currently Unknown
Tenergy 9.6V Flat NiMH Battery	1	\$35.59
LM7805 Voltage Regulator	1	\$0.99
TPS55340RTER Voltage Converter	1	\$1.86
Primary PCB	1	\$30
HR50A Laser Rangefinder @ 1000m	1	\$284.40
Scope Assembly	1	Currently Unknown
Various Lenses	7	\$191

Table 22: Parts List

Total Cost (subject to change) = \$452.32

* This project will be self-financed by the group members *

7.3.1 Laser Rangefinder Cost

Both the laser rangefinder and the scope will be the most expensive parts of the project. As far as laser rangefinders go, this is on the cheaper end of the market. Many rangefinders of the same category are significantly more expensive. The cost of the laser rangefinder will be split among the group member making it much more affordable than if the cost was all on a single member.

7.3.2 Targeting Lens Cost

The targeting lens is a rather inexpensive part. While we had several designs that may have ended up costing several hundred dollars, we eventually came upon a design that would be inexpensive without degrading the quality of the product a significant amount. Relative to the rest of the parts, this part is likely one of the cheapest parts. The display itself costs less than \$30 and the PCB designed to run it will be less than \$100. Other than the cost to mount the lens and the cost added to the power supply to power it, it is unlikely that it will cause many issues when it comes to our budget.

7.3.3 Scope Cost

The scope will end up being another large portion of the cost of this project. The housing of the scope must be manufactured out of aluminium. Because of this it must be produced by special machines that are expensive to use. It is likely that we will have to hire a company to manufacture this, so the cost is still unknown.

The lenses for the scope are also very expensive. They can end up costing several hundred dollars. These as well must also be ordered from a company that is able to produce the lenses we need. Because our targeting lens is an OLED display, it will end up being one of the cheapest lenses we have to buy.

7.3.4 Microcontroller Cost

There are various cheaper microcontrollers we could have gone with but there are several reasons we went with the one we did. The first reason is the amount of data lines and interfaces it supports. Having support for multiple interfaces is important as our rangefinder uses UART and our targeting lens uses I2C.

Another reason for this choice was the built-in Bluetooth module. Because of the app that will be used with this product, we needed a way to communicate without requiring to connect to the scope via a cable. This means a way to communicate wirelessly, in this case Bluetooth, was essential for the board to have.

7.3.5 Power Supply

The power supply will not be a big cost for this project. It will be printed onto the same PCB that the targeting lens board is going to be printed on. We're going to be using only one battery for the entire design of the power supply so this decreases the budget on it drastically. The battery we're utilizing is only \$35.59 and the parts necessary to complete the system which are the voltage regulators comes at a total of \$2.85. The power supply comes at a grand total of \$38.44. This is relatively cheap for our project and goes well towards our goal to make the project cost-efficient.

7 .3.6 Hantek 2D42 3 in 1

We were fortunate enough in this project to receive the Hantek 2D42 3 in 1 thanks to UCF free of charge. While we are not able to keep this product or using it in our final design, it will be useful for testing and prototyping different designs.

7 .4 Project Challenges

While designing the project, we often had ideas that sounded cool at the start but turned out to be problems. There are also cases where an idea is completely feasible, but due to outside constraints, accomplishing those tasks were more challenging than we initially thought they would be.

7 .4.1 Laser Rangefinder

The rangefinder presented several challenges. The first was finding a rangefinder that would work for our project. There are many different rangefinders on the market, but a large majority of them are for very small distances. The small distance rangefinders are typically very cheap and easy to find. But, for our project we needed a long range laser rangefinder. This presented a big challenge as finding this rangefinder was a difficult task. To start, when looking for a rangefinder, most on the market are short distance rangefinder. When we did end up finding a long distance rangefinder, they were often several thousand dollars and only sold to special manufacturers. After much searching we eventually found several laser rangefinders that would work and were within our budget. Although the distance they measured was a little shorter than what we initially wanted, we were able to make them work.

After we found a laser rangefinder, the next challenge was figuring out how to make it work. The laser rangefinder is being shipped from overseas and would take several months to get here. So to deal with this, we had to find documentation about the rangefinder online. Attached to the product page was a small data sheet. It did not provide an excessive amount of information, but we managed to get all the information we needed to properly program the microcontroller to work with the rangefinder once it arrives.

7 .4.2 Power Supply

The current design that we're using for our power supply design wasn't the first one. Initially, our design was to include two batteries instead of one battery to supply power to all the components on our system. This design worked, but it wasn't as space-efficient when compared to our current design. Also, one of our goals was to make our project as lightweight as possible, so adding one more battery would certainly add a hefty amount of weight to the entirety of the project.

The two batteries that we wanted to initially use weren't NiMH and/or Lithium-Ion/Lithium-Polymer, they were battery holder case boxes. These are plastic and metal boxes that hold four or eight universal AA batteries. The advantage of this design is that we could easily recharge our power supply if it drains and AA batteries have high mAh close to 3,000mAh in comparison to batteries on the market which is 800mAh to 2,000mAh. This difference is significant when calculating how long our power supply design will last with all the loads connected and running at the same time which is what we took into account for our calculations.

On the other hand, the challenge with using this design of individual AA batteries is that one of the battery holder case boxes didn't have any type of proper cover or seal to prevent the AA batteries from falling and completely turning off the entire system. This is crucial to look out for because we don't want to cut off power to our components while they are operating and without safely turning them off; doing this can cause permanent damage to the components. Also, this would cause unnecessary challenges to the hunter if the batteries kept adjusting out of place or falling off completely.

Using this design will include two of these battery holder case boxes like mentioned before. One of them was a 12V with 8 slots for AA batteries and the second one was 4.5V with 3 slots for AA batteries. Including both of these batteries would require more work to secure it under the scope and more connections that are subject to possibly being damaged through use. For example, they would be subject to drastic and sudden movements when the hunter uses their rifle.

Lastly, the last challenge was trying to find a battery holder case box that would have an output that matches the recommended operating input voltage for the MCU and some of the other components which were 5V. The maximum and minimum recommended operating conditions for this 5V input voltage were from 4.5V to 5.5V. So, using the 4.5V battery holder case box would fall right on the minimum of the recommended input voltage. By using this battery, we would run the risk to have a lower output voltage from the 4.5V than what is expected once we connect all the loads required to this 4.5V battery. This could cause our components to not even turn on and our MCU to not function properly as it may or may not receive the correct recommended input voltage.

7.5 Scheduling

The following table breaks down our schedule into a week-by-week basis. This schedule starts on The first week of the 2021 Fall semester and ends on the last week of the 2022 Spring semester. While we may not end up following this schedule exactly as listed, it will provide a general outline of when milestones of the project should be started and completed.

Week	Goal
1	Syllabus week
2	Creating group
3	Deciding roles
4	Decide project idea
5-7	Start report, research project components
8	Finalize initial scope design
9-10	Research into specific parts
11	Finalize bill of materials
12-13	Order parts & design PCBs
14	Finalize project design
15	Finish Report
16	Submit Report
End Fall Semester	
17-19	Build rangefinder, scope, and program microcontroller
20-22	Finish scope casing and lens system
23-24	Add micro-led display to scope lens
25	Begin assembling components
26	Test systems together as a single unit
27-30	Fine tune and debug system
31	Cleanup project and polish
32	Present project

Table 23: Schedule

8 Conclusion

For the optical components of our project, we had many design changes along the way to our current decision. Initially, we had used the current convention of scopes as a reference, which use around 7 to 9 different lenses, but we settled around 5 to accomplish our goal. Although we were able to achieve a zoom system, our demonstration was in too small of an area for the overall scope to properly focus, so we have yet to show a true demonstration. However, with testing, we will be able to verify our work. The initial decision for a larger selection of lenses was altered after researching the inner workings of the scope and comparing it to our current knowledge of optics. In our previous classes, we have mostly covered symmetrical lenses, such as the double convex and double concave lenses that we ended up using, but many scopes on the market also utilize plano-convex lenses to assist in capturing collimated light. The reason we did not include one of these lenses in our design is due to unfamiliarity, however we are continuing research for personal benefit. The position of the plano-convex lens is generally forward of the objective lens on our current assembly, serving as the objective lens of most assemblies, which is why we want to continue our research for possible designs that could happen post-graduation, however we believe that even our current system is a testament to the knowledge we acquired even this semester.

The other lens that we did not current include in our design, that is present elsewhere in the industry, is a double convex lens that exists behind what we have as our focal lens. This takes place as the focal lens on these assemblies. The purpose of this lens is to adjust the eye relief for the user to ensure that an object is in focus at all magnitudes of magnification. Although the image is not so blurry that it becomes indistinguishable, the focusing of smaller objects is also more comfortable for the eye. This adjustment has to do with the parallax of the scope, which refers to the phenomena where objects do not focus on all distances. Adjustment for parallax does not exist on all commercial scopes, however, as it is usually reserved for higher-end models, as adjustment is accomplished by movement of both the image reversal assembly and focal lens in small increments. This is absent on lower-end scopes to keep prices down, as it is generally accomplished by an extra adjustment turret, which means more mechanical parts on the inside. We do not foresee adding the parallax adjustment of the image reversal assembly; however, we will add the new focal lens to make adjustments if our practical tests yield issues at longer distances.

Other changes that we decided would differ from the original idea involved overall length of the unit and the object lens side. We initially chose our design to be a 1-6x24 scope, which as we covered, means that the lowest magnification of the scope will be a 1 time, and the highest magnification will be 6 times, with an objective lens diameter of 24 millimeters. This change happened because the initial idea for designing the optical system was far more complex than the actual process needed to be as it was the first major design we had attempted in our studies. The original idea was to calculate a multi-variable equation that would have calculated

the effective focal length of an optical system when different focal lengths were used for each of the lenses. We were under the impression that this was the most efficient way to design a system to the parameters we had set. The main issue that would have come about from this method, however, was it would most likely have required us to have lenses custom made for the scope. With the time and resources that we have available, that would not have been a choice. After meeting with previous instructors, we realized that a design could have had essentially arbitrary focal lengths chosen that would satisfy the length requirement of our system, and we just had to follow the optics principles when decided the location of each lens, such as ensuring that each lens be placed at a distance either at the focal length or proportional to the focal length (such as the image reversal assembly requiring a $2F$ distance). Then, the only other design idea was that we could use double convex lenses but would use one double concave lens for the magnifying lens. The way that we finally decided which focal length lenses that we were going to use was by first searching the market to see what was available. We chose to use Edmund Optics as the company for the lenses, as it was in the price range and the quality was more than fitting for our design. We contemplated using other brands, such as Newport or Thorlabs as I have personally used their lenses in undergraduate laboratories prior this design, but the price was too steep for our project. We needed to ensure that the lenses were large enough to gather an appropriate amount of light for the system as well as having a small enough focal length that the system would be as compact as we would be able to achieve. The selection of lenses we had available was main contributor as to why we changed the initial 24-millimeter lens to a 30-millimeter lens.

The zoom system that we put in place is more advanced, as it stands, than we have learned in our previous classes, and we had to learn and consult our previous teachers to understand the concept. Prior experience has introduced us to the idea of a zoom lens system, however it was more a demonstration, where we were instructed in what lenses to use as well as where to place them along an optical rail. The difference between that experiment and this design project, aside from the fact that we were given jurisdiction over the focal length and size of the lenses, as well as their position in an optical system, is that we had to shrink the overall system down so that it could fit into a handheld device (for a size reference, as the actual product will be rail-mounted) and optimized for longer differences. Also, we needed a system that would capture light at infinity, while the lab had us placing the object the distance of one focal length away. Compared to the experiment as well, where we used 3 double-convex lenses, we chose to use 2 double-convex lenses and a single double concave lens. This helps with keeping a more compact system as the lenses will be able to move closer to each other while still providing the desired magnification, and we do not have to worry about lenses touching so they do not scratch or crack each other.

Future considerations of this project will include better processing on the software end, things such as multi threading and complete camera integration's are the things we can possibly do with more time and money. Multi threading is an ex-

cellent approach to make the system overall faster, this would allow us to compute the distance and work with the range finder at the same time. While this would require a stronger computer core to allow for multiple threads, it would reduce run times to about a third of the original run time. We could also do video recording and saving it to our drive while doing calculation or even machine learning. Another item to consider in future integration's is a complete camera integration onto our micro controller so that we don't have to declare connections before using them they would come natively onto our program thus allowing us to not rely on slow connections to send enable bits back and forth we could just use an in built function to enable the bit. This ties in well with our multi-threading discussion as if we do not have the same code running we may not be able to utilize an efficient multi-threading approach. Another item to consider is to utilize machine learning to completely account for any bullet falloff ahead of time instead of real time measurements, this would be done similarly to the guns.json each gun would contain a separate pre-trained machine learning algorithm that would allow us to predict using any kind of information such as wind speed, or elevation. This would require multiple hours of optimizing models but this could be done in proper environments to obtain all the results needed to configure and train the model.

A References

- [1] leee - about ieee.
- [2] Time to digital converters.
- [3] 10811a/b quartz crystal oscillator operating, 1980.
- [4] Arduino. Arduino portenta h7. page 22, 12 2020.
- [5] National Electrical Code. American wire gauge conductor size table. 2017.
- [6] Trojan Battery Company. State of charge.
- [7] Inc. CUI. Power supply standards.
- [8] Wavelength Electronics. Noise and ripple on power supplies.
- [9] Rockwell Laser Industries. Laser standards and classifications.
- [10] Texas Instruments. Tps55340 integrated 5-a wide input range boost/sepic/fly-back dc/dc regulator. 2021.
- [11] NASA. Soldered electrical connections. page 84, 01 1997.
- [12] OpenMV. Openmv ide, 12 2021.
- [13] Bill Schweber. Lidar and time of flight, part 3: Emitters, sensors, and scanners, 2019.
- [14] Shawn. Lm7805 voltage regulator.
- [15] John Teel. Linear and switching voltage regulators, 08 2021.
- [16] Telan. Through-hole vs. surface mount: Contrasting benefits and uses, 2017.
- [17] Hai Wang, Min Zhang, and Yan Liu. High-resolution digital-to-time converter implemented in an fpga chip. *Applied Sciences*, 7:52, 01 2017.

B Software

- **EasyEDA** - Used to design circuits, simulate circuits, and develop PCB designs.
- **Zemax** - Used for optical calculations and lens designs.
- **diagrams.next** - Used to produce flow diagrams and block diagrams.
- **Anroid Studio** - Used to program the phone app.
- **Visual Studio Code** - General Python Programming
- **Webench** - Used to design power supply.