Mobile LiDAR Scanning System

University of Central Florida
College of Electrical Engineering and Computer Science
CREOL, The College of Optics and Photonics

Group 2

Dr. Wei Lei
Dr. Samuel Richie
Dr. Aravinda Kar

Senior Design I - 120-Page Draft

Mateo Cuesta - Photonic Science Engineering
Dannah Dolorfino - Computer Engineering
Arturo Martin Jimenez - Photonic Science Engineering
Troy Morgan - Computer Engineering

Table of Contents

1.0 Project Narrative

Reconnaissance is crucial and necessary to most military efforts. Reconnaissance, also referred to as recon, is when soldiers do a thorough search of a particular area and make sure there are no adversaries in the area. This is challenging because one may not have a good sense of the layout of the room. To reduce risks of injury and even casualties, it is most helpful to know what to look out for in an area before entering. Our group thought of a product that is able to produce a 3-dimensional model of an area using LiDAR technology. This will allow soldiers to better understand the layout of what they are dealing with.

LiDAR, which stands for Light Detection and Ranging, is a method of remote sensing that uses lasers to measure distances. The reflected light from the target can be stored and processed to produce a 3-dimensional model of the area. LiDAR differentiates from conventional remote sensing due to the fact that it uses light in the near and mid-infrared sections of the electromagnetic spectrum (NIR/MIR). LiDAR technology is also capable of working in low ambient light settings, over long ranges depending on the optical setup, and can represent both stationary and moving objects.

The goal of this project is to create a remote-control car fitted with a LiDAR scanning system on top. With this configuration, we want to be able to drive the remote-controlled or RC car into a room and produce a 3-dimensional model of that room. This idea was suggested because recon in the military can be a dangerous job. To make it easier on our soldiers and to keep them safe, we plan on creating a device that goes into enclosed areas and scans using LiDAR technology. The data from the scans are sent via a wireless connection to a remote user interface. The data is then processed to produce a 3-dimensional model of the area.

The team is made up of two Photonic Science Engineers and two Computer Engineers. The project workload will be divided into two sections, the LiDAR system, and the software/hardware system. The two Photonic Engineers will ensure that the LiDAR system will scan and image the room as well as meeting OSHA laser safety standards. The two Computer Engineers will ensure that the electronic components are able to integrate the LiDAR system as well as the image rendering. The programming language that will be used in this project is python. We selected this language due to the vast libraries that exist for image processing. The Mobile LiDAR Scanner will be remotely controlled via a controller. The user will be able to guide the car into any area that they want to model. The image processing will be done on a computer that is not located in direct proximity to the Mobile LiDAR Scanner.

To summarize, the Mobile LiDAR Scanner will be able to render a 3-dimensional model of a scanned area. The data will be transferred wirelessly to a remote computer to process and display the data. This application of LiDAR technology can be used in military applications such as recon. Having technology that can produce a 3-dimensional model of an area, without the need for on-site human supervision gives military personnel an advantage when planning out missions. Being able to have a visual of an area or room alleviates some danger from entering blindly into the room.

2.0 Project Description

2.1 Project Motivation

The motivation for the creation of our project was mainly due to the group's interests and our strong suits for our major. This group is composed of two Photonic Science Engineers and two Computer Engineers. Photonics is the study of the transmission of protons, this is also shown in fiber optics. The application of Photonic Science is used across multiple parts including lighting and displays, optical data communication to imaging, which is used in many aspects of society like health care, security safety or even in manufacturing sector to life sciences. Computer Engineering is the combination of computer science and electrical engineering to develop computer hardware and software. The application of computer engineering is most used in the development of today's inventions. Embedded systems are a token of what computer engineering symbolizes. The device used both hardware and software to distribute a desired outcome. With the combination of the two majors, we decided on attempting to create a device that used LiDAR.

When brainstorming many ideas that are geared towards the majors of this group, we stumbled upon many different design proposals. Many of the designs were trying to implement the LiDAR and Radar technologies for the device. The first invention was to aim towards using LiDAR in an autonomous car that would be able to detect specific objects located in a room. The vision was to have the LiDAR device scan a particular object, or upload the dimensions of the specific object, then when the object is downloaded and rendered in the system, we were to let it roam a room, then automatically detect the item that we set out for it to look for. After intensive research on learning how to develop a product of that caliber, we began to lose hope in the development of this idea. Many of the constraints dealt with the cost of this product, knowledge in particular fields with the development of this product and time constraints. Although this project would be constructed over two semesters, the group didn't feel too confident in manufacturing the idea.

The second idea that the group began to dwell on was instead of having the autonomous car roam around the room looking for specific objects, we decided to make somewhat of a security system that was still autonomous in nature, but it scanned the room periodically to look for any changes in the area. If a particular item was set in a place for a reasonable time, the LiDAR car, after sending multiple scans, should've been able to detect the missing items. Since LiDAR sends multiple beams in the area, we felt like this could be a good way to distinguish a change. This idea also had some constraints after researching its development. We began to realize that a lidar scan may not be the best when deciphering objects. As an example, if this LiDAR machine was set up in a public place such as an airport, the device would have a hard time identifying between a person or object. The LiDAR sends multiple beams, but the beams aren't discerning enough to be able to tell the difference. Again, this was an invention the group was confident in, but when the computer engineers were thinking of the software development in deciphering objects from people, it would be nearing impossible to create, unless we used a different form of detecting.

Which brings us to our final design on the topic. Understanding the constraints of our previous designs, we decided on creating a LiDAR machine that's remote controlled to a room or area, then with LiDAR, it can create a 3D image of the room and send this back to the user. The scan would begin to take a series of images, then the return value would be of what the libraries of Laspy can

read, Laspy of which would be discussed in further detail in the following sections. The proposal came from understanding the harsh situations military soldiers have to deal with when performing reconnaissance in a room or area. One deems a room to be dangerous before entering if the soldier doesn't know what can be on the opposite side of the door.

The scan is sent to the user's computer via Bluetooth, so instead of trying to get the information back from the remote-controlled car, we can get it automatically once the scans complete. Implementation should be quite simple knowing the groups strong suits on particular areas in their studies. After the LiDAR scan that is going to be assembled by the Photonic Science majors, when it sends data, this can be directly read into a program the Computer Engineers could read. They would then use that information to render a 3D image of what was recorded on their laptop. Although this may be a tedious task for the group, especially in the LiDAR mechanics and software issues, the group felt more confident in producing this device as opposed to the other ideas that were mentioned.

2.2 Objectives and Goals

The goal of this senior design project is to design and build a mobile LiDAR detection system that will map an area of interest and provide information about the surroundings to the user. The dimensions for an area are defined in Table 2, scanning volume. In Table 1, the goals are broken down and analyzed with their respective objectives.

Table 1: Goals and Objectives

Component	Goal	Objective
Optical	Build a LiDAR system to scan an area	Design, build, and integrate an illumination and detection system.
Mechanical	Mobile LiDAR	Integrate LiDAR system into RC car for mobility.
Software	Render 3D model of scanned area	Process data in an outside CPU to render a 3D model for the user.
Hardware	Transmission of data	Integrate a Bluetooth module into the LiDAR system to transmit data for image rendering.

Software: As discussed above, this team would be using Python alongside libraries that are associated with it to be able to properly display the image of the scanned area to our screen. We would also be using C to configure the settings for the microcontroller to be able to use Bluetooth to have connection from our device to the machine we're getting the readings on. When both

programs are run on the system, it should be able to produce a LAS file that outputs the data and points to the screen. Depending on the size of the LAS file that is received, then it would relate to the image that's being processed.

Hardware: The devices used in this project would be a microcontroller, laser diode, photodetector. That's on the full LiDAR machine which should be battery powered on an independent remote-control device. We would also use a Bluetooth module to be able to communicate to and from the user. The remote-control device would be a simple RC car that we assemble and attach the peripheral devices on.

Power: We would be using a generic battery, the battery would power the RC car, the microcontroller, and the LiDAR system. The amount of time the device would be operating at would be about 20-30 minutes. We would internally have all the components with set voltages and currents to be able to output the required amount. To be able to keep the device running for at least 30 minutes, the power distribution would play a key role in development.

Connection: The communication used would be the Bluetooth module that would be connected to the microcontroller. The user would output specific characters from the keyboard to be able to perform different functions on the device. Bluetooth was implemented over Wi-Fi due to the better security and no requirement for an internet connection to be present. Bluetooth can be point-to-point and can be accessible through multiple devices. The Wi-Fi standards can be somewhat robust to implement knowing the issues that can arise when establishing a secure connection

2.3 Requirement Specifications

Project specifications are statements that address the technical need of an engineering design. These specifications further describe what needs to be done in order for our project to be a solution to our problem. Table 2 below summarizes the specifications needed to be addressed as our team designs and builds the LiDAR scanner. Some of these specifications are imposed by the nature of the application. Since our system is to be used for reconnaissance purposes, the horizontal field of view (FOV) of the full scan should be 360° around the system to provide full coverage of the room or area of interest. Similarly, the system should be able to optimize resolution for the area or room to be scanned, changing the illumination and sensing FOV between 10° and 45°, accordingly, to achieve the desired optical power distribution on the target and to detect the entirety of the illuminated area.

Table 2: Project Specifications

Specification	Description	Unit
Time to generate a 3D model	A 3D model of the environment should not take too long to generate	≤ 15 minutes
Horizontal Field of view	The LiDAR system is able to scan horizontally across an area	360°
Vertical Field of view	The LiDAR system is able to scan vertically across an area	10° - 45°
Scanning Volume (Open Area)	Volume to be scanned and mapped in an open area.	9000 - 20000 m^3
Scanning Volume (Room)	Volume to be scanned and mapped in a closed room.	300 - 2000 m^3
Detection range for LiDAR	How far the LiDAR system is able to read distances.	1 - 50 m
LiDAR device volume	The LiDAR system will take up this amount of space on the vehicle itself.	< 2500 cm^3
LiDAR device weight	The LiDAR system itself should not weigh more than this weight.	< 5 lb
Total vehicle weight	Must be able to be carried along during missions.	≤ 12 lb
Control range	Be able to receive LiDAR data and execute movement instructions from the user in this range.	1 - 260 ft
Operation lifetime	Vehicle must have enough power for movement and LiDAR scans for this amount of time.	30 minutes
Speed of System	This would determine the operating speed thar we intended to use on our system.	< 2 mph
Turn Radius	The degree at which our machine could turn, dependent on the types of wheels we use in our setup	0°
Bluetooth Range	The range at which the Bluetooth module can receive data to transmit to our LiDAR system	< 20 ft

The generation of the 3D scan is a key feature of our LiDAR scanner. Thus, it is important that the creation of the 3D point-cloud should not take too much time. The scanner must provide a 3D point-cloud representation of the scanned environment after the LiDAR scan is complete. This allows the user to quickly visualize an unknown environment and make decisions accordingly. In addition to render time, the power on the scanner must be able to support the mechanical parts of the LiDAR system itself and the transmission of data to the application.

In addition, we also have to include areas of physical functions regarding the RC car. Distinct features such as the speed, turn radius or even the range at which the Bluetooth can operate are all key features in the design of our project. The speed of the LiDAR system has been thought of significantly. We would need to create a system that moves at a steady pace while still managing to perform efficiently. Too slow the operations could be time consuming, and too fast could cause permanent damage to our car due to crashes. The turn radius is also an important feature we plan to use. This could be an option into rotating the LiDAR system for a full 360-degree scan. The efficiency could be improved, and the cost of the product could decrease knowing we wouldn't need to include a servo motor for rotation of the LiDAR scanner. The range the Bluetooth modules operates is also a key ingredient for production in our system. We would need to understand the range at which our system can operate. If this system is driven in a room or area, one would need to understand the areas at which it would work. More details regarding these features are present in following sections below.

The area of interest scanned by the LiDAR is variable to the vertical FOV that we can achieve. Likewise, the maximum range of the LiDAR, that is, the maximum distance that can be detected, depends on the vertical FOV as the power from the illumination system needs to be more tightly focused to reach longer distances and receive enough optical power at the detector. The maximum range also depends on the reflective properties of the target object. Table 3 below shows the variance between the maximum range and the maximum object height for increasing vertical FOV. Note that the target object is assumed to be retroreflective, thus more light is reflected back at the detector so the maximum range would be shorter for material with lower reflectivity. A more detailed description including targets with different reflectivity's will be included in further documentation of the project.

Table 3: Vertical FOV vs. Maximum Detection Range

Vertical FOV (º)	Max. Range with Retroreflector (m)	Max. Object Height (m)
5	50	4.37
12	28	5.89
25	15	6.65
45	6	4.97

The device itself is meant to be mobile and portable. The mobile LiDAR scanner needs to be able to be carried by its users as easily as possible. Once the LiDAR scanner is placed, it can begin its

scan when prompted by the user. The LiDAR scanning system on top of the vehicle itself will take up more weight compared to the embedded components. The embedded components on board the LiDAR system itself will mostly The LiDAR system itself is planned to weigh less than 5 pounds. We will have to account for the weight of the LiDAR system to ensure that it does not affect. Thus, we plan that the entire mobile LiDAR scanner to weight under 12 pounds.

Regarding the control range of the LiDAR scanner, there are two aspects that we must consider as we build the mobile LiDAR system: the movement control range and the data transmission range. Data transmission range is restricted by the range of the Bluetooth module. The movement will be controlled remotely.

2.4 House of Quality

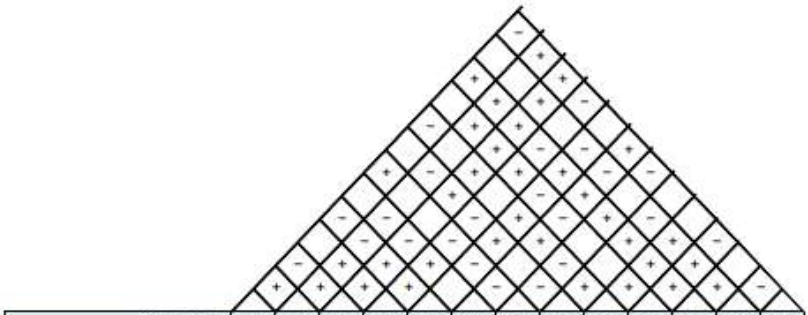
The House of Quality is a matrix for planning a way to show how customer requirements relate directly to how companies can achieve said requirements. Figure 4 shows a correct correlation from the user to how we chose to implement it. The specific qualities of our device are compared to the components each of our users would experience. The focus of the parts in our device was taken and compared using the house of quality.

The Mobile LiDAR Scanning System created many thoughts on how we decided to implement it. Gearing more towards consumer satisfaction, the team displayed key features alongside the engineering requirements. These features determine the safety, efficiency and how user friendly it is when utilizing our product. The team began to understand what variables may be affected and how we were able to resolve it. This House of Quality design was able to show us a visual representation of how our design would be created.

In the below diagram, the house of quality is divided into two parts. It shows the Engineering requirements and the Consumer requirements. The Consumer requirements show us different variables that are assessed when dealing with the device. The Engineering requirements are different aspects associated with how well we administered our product. The variables show important features that are used to configure our device.

The bottom row of the House of Qualities are our target values when designing our product. We listed values such as the cost, the FOV, and even how we plan on communicating our data. To be able to build this product, these target values are necessary for our design. These are considered mostly due to knowing what we should expect, and also understanding what's going on. The parameters listed are the basis of the device and values we take into account.

2.4.1 House of Quality Diagram



Correlations	
Positive	+
Negative	−
No Correlation	

Relationships	
Strong	●
Moderate	○
Weak	▽

Column #	1	2	3	4	5	6	7	8	9	10	11	12	13
Customer Requirements (Explicit and Implicit) \ Engineering Requirements	Size (cm)	Cost	CPU Processing	Laser Power	Operational Lifetime	Hardware communication	weight	Microcontroller Processing	Power consumption	Laser Wavelength	Sensor	FOV	Operational Range
Afforable	▽	●	▽	●	○	▽	○	○	▽	○	○	▽	○
Portability	●	○	▽	▽	▽	▽	●	▽	▽	▽	▽	▽	▽
UI Interface	▽	○	●	▽	●	●	○	○	●	▽	▽	▽	●
Vision Range	▽	●	●	●	○	○	▽	●	●	●	●	●	▽
Power Consumption	●	●	●	●	●	○	○	●	●	●	▽	●	○
User Friendly	○	▽	▽	○	▽	○	●	▽	▽	▽	▽	●	●
Performance	○	○	●	●	●	●	●	●	●	●	●	●	●
Efficiency	●	○	●	●	○	●	○	●	●	●	●	●	●
Mechanical runtime	▽	○	●	○	●	●	○	●	●	▽	▽	▽	●
Safety	○	▽	○	●	▽	▽	○	▽	▽	●	▽	▽	○
Control Range	▽	●	▽	▽	▽	●	▽	●	▽	▽	▽	▽	●
Target	28 x 25 x 30	≤ $400	3 GHz	< 500 mW	≤ 30 mins.	Bluetooth	≤ 12lbs.	16 MHz	TBD	940 nm	CCD Sensor	360°	≤ 50m

3.0 Technology Investigation

For our senior design project, we chose a project that would allow us to take the concepts that we have learnt from different classes and merge it all into one functioning system. Our project can be broken down into various subsystems, which can be further broken down into other subsystems. For example, the LiDAR component can be decomposed into two main subsystems, the illumination and sensing subsystems. Taking the sensing subsystem, it can be decomposed into the optical component and the sensing component. These two subsystems are core classes taught throughout our college career, but they are taught individually. It is important to conduct technology investigation into similar technologies that are currently in the market to understand how our product should work. The workload was divided up depending on what subsystem each member wanted to contribute too, this can be seen in the block diagram fig #. It is each team member's responsibility to conduct technology investigation into their subsystem.

This section will explore similar products that are currently in the market by describing the application of their technology. We will present the similarities and differences in our features that will make our product unique compared to the existing technologies. This section will also discuss the important components of the Mobile LiDAR Scanner and the technology investigation done for them.

3.1 Existing Relevant Technology and Similar Projects

3.1.1 LiDAR Technology

LiDAR technology is becoming increasingly popular in the field of range finding. From autonomous vehicles to geographical mapping to military applications, the need for a precision mapping device is in high demand. LiDAR technology allows the operators to take a large 3-dimensional space and project it onto a 2-dimensional screen for analysis.

LiDAR technology in the automotive industry has brought autonomous vehicles to our daily lives. LiDAR in combination with machine learning and computer vision allow cars to think for themselves. Since LiDAR collects data as points in space, it allows engineers to input that data into a machine learning algorithm to teach computers how to interpret the data. LiDAR technology in autonomous vehicles also allows for velocity detection as well as object recognition. The velocity detection is calculated through the time of flight (TOF) calculation. LiDAR with machine learning is also capable of performing object recognition. This is because the collected data points, when plotted out, will have a distinctive shape in a 3-dimensional space. Repeated testing can allow researchers to teach the computer to interpret the data and be able to distinguish a car from a human by the input data.

Another common application of LiDAR technology is in the field of geographical mapping. This application is mostly done by mounting a LiDAR system onto an airplane to scan from a top-down view of the area. This technology also incorporates ToF technology to measure the surface of the landscape. Geographical mapping with LiDAR allows us to view locations and areas from a bird's eye perspective. Although other methods of imaging allow us to do the same, LiDAR allows the users to collect information from individual points to reconstruct the landscape for studying.

Applications of this technology are currently being used in the amazon to create a map and study the vegetation of the amazon. They are comparing measurements taken at various times of the year and analyzing how the amazon is constantly changing over time and to measure for deforestation. Another application of airborne LiDAR is to map floods and change of terrain. This is being done by the National Weather Service, where they are collecting data of the elevation in cities to analyze for potential floods in the future.

Another existing technology similar to the Mobile LiDAR Scanner is a previous senior design project done in the 2017 - 2018 academic year. This project done by Group 10 of that year shares a similar goal as our project but incorporates the technology in a distinct way. Both groups share the same goal of creating a 3-dimensional model of an area. The main features that distinguish our project from theirs are the rendering time, the range of sensing, and mobility of our project.

3.1.2 SLAM

SLAM stands for simultaneous localization and mapping [1]. A SLAM robot is usually exploring an unknown, static environment. Using only its own controls and the observations of features of the environment, the robot must create a map of the area and draw its own path in that map. This process often has the robot considering information captured from its previous position. SLAM is quite the difficult problem because in the real world, mappings between the observations and landmarks is unknown. Thus, making the wrong data associations would prove to be quite disastrous. This would result in the map being drawn from different perspectives aggregated together. There are methods to improve the maps but, they are often very expensive machine learning processes and may require more computational power.

Initially, our idea for this project had similar premise to a SLAM robot, where the LiDAR scanner would create a map of its surroundings and be able to find itself in that map. Using the live distance data from the LiDAR, the robot would have some level of autonomous movement. If the robot came a certain distance between a wall or obstacle, it would move away in the opposite direction. Of course, this type of robot would need lots of processing power and machine learning techniques be able to map its surrounding and be able to avoid obstacles. The development time needed for this type of project would be way beyond the two-semester time limit for Senor Design for students of our skill.

Our team ultimately decided on a LiDAR scanner that was built the vein of a SLAM robot. Rather that simultaneously mapping the environment as the robot moved forward, the robot would move into position designated by the user and begin a static scan from that position.

3.1.3 Similar Products

To add onto the existing technologies of LiDAR systems that have been developed. The team came across a device that has a similar design. The device is remote controlled and also has a LiDAR system attached to it. The device uses wireless communication through a smartphone app to control the car, this would give the user full control on where they'd want to position it. Also using wireless connection, the device is able to start and stop. The device is able to drive to the desired location, make a three-dimensional scan of the area, then return the image that was processed. The company XiaoR Geek created a machine that properly depicts a view the team is headed towards.

Their machine is equipped with a Raspberry Pi and a 4GB motherboard, which is used to make simple computations. The software they used was Python and C language to be able to perform its desired functions. The built in built-in XR-ROS graphical human-computer system makes it easier for them to teach fast operations. This Raspberry Pi ROS SLAM navigation system can understand the different functions of LiDAR for building map and have neat positioning systems. The robot supports the SLAM algorithm, which is basically having map awareness. SLAM stands for Simultaneous Localization and Mapping. Thus, feature is a process of creating a map it generates, then also being able to explore the map it created. For the camera that captures the images, it has a 1080P HD camera with RpLiDAR A1 radar. The specs of the camera help make map building easier.

To explain more of their software implementation, XiaoR Geek provides somewhat of a manual to help the user understand the specifications of the robot itself as well as teaching the user basic SLAM techniques. It gives a better understanding of the autonomous navigation using the Raspberry Pi. Their application is widely used in artificial intelligence institutions. Although this system is running autonomously, our team decided to make our device remote controlled. This is all due to the fact of giving the user more control, for in military situations, they can pinpoint exactly what they need a scan of.

3.2 Relevant Technology

Zoom lenses: One relevant technology that can be implemented to the illumination system is a zoom lens optical layout. This zoom lens system is most commonly implemented in cameras because it allows the user to adjust the focal lens of the system. By adjusting the focal length, the user is able to attain a better image resolution. With respect to LiDAR systems, adjusting the focal length of the collecting optics can either increase or decrease the FOV. This allows the user to detect at various ranges with the best resolutions at each range. This concept of adjusting the focal length of the detection system is described in more detail in Section 5.2.3.2 (Detection Optical Design).

Photodiodes: Photodiodes are semiconductors that are able to convert light to electricity by photon absorption. The photodiode is created on the principle of having two distinct sides, one doped positively with holes and the other doped negatively with electrons. The positively doped side is denoted with the letter p and the negatively doped side is denoted with the letter n. When the p-n sides are joined, a depletion region is formed. This is the area of the semiconductor that allows recombination between the holes and electrons. For current to be generated from photodiodes, the holes and electrons need to recombine. In order for recombination to happen an external stimulus needs to give the electrons enough energy to go from the valence band to the conduction band, this bandgap energy is inversely proportional to the wavelength of light. This is the basic working principle of photodiodes. The relationship between the bandgap energy and the wavelength of light can be seen in the equation below.

$$E_g = \frac{hc}{\lambda} \quad (1)$$

When a light whose wavelength matches that of the bandgap energy of the photodiode electrons has sufficient energy to recombine with the holes in the conduction band and current flows. When there is a light constant incident on the photodiode, there is a constant current that flows through the system which can be detected. Measuring this current gives us an understanding of the incident light. For modulated light that is incident on the photodiode, the current that is produced is also modulated and can be detected as well. The following image, Figure 1, gives an understanding of how current is generated in a photodiode.

On-going research in the field of semiconductors has led us to the development of the avalanche photodiode. The avalanche photodiode functions with the same principles that are described above for the standard p-n photodiode with one key difference. The avalanche photodiode operates at a high reverse bias condition. By operating at this condition, the electric field of the depletion region will cause the electrons to be pulled away with a high velocity. This causes them to collide with other electrons in the lattice producing a domino or avalanche effect. This is where the term avalanche photodiode is derived from. This also allows for avalanche photodiodes to have a higher quantum efficiency than p-n photodiodes.

Since their quantum efficiency is also much higher, avalanche photodiodes have a much lower electronic signal-to-noise ratio. This is because of the larger responsivity at the operating wavelength than p-n photodiodes. One drawback of the avalanche photodiodes is that they are susceptible to quantum noise, which is the noise that arises from quantum fluctuations. The primary quantum fluctuation that is seen with avalanche photodiodes is in the change of the electric field of the depletion region. Since this is the area that light interacts with the semiconductor. One way to balance this out is to select the operating reverse voltage such that the quantum noise is equal to the electrical noise, this lowers the overall noise of the diode.

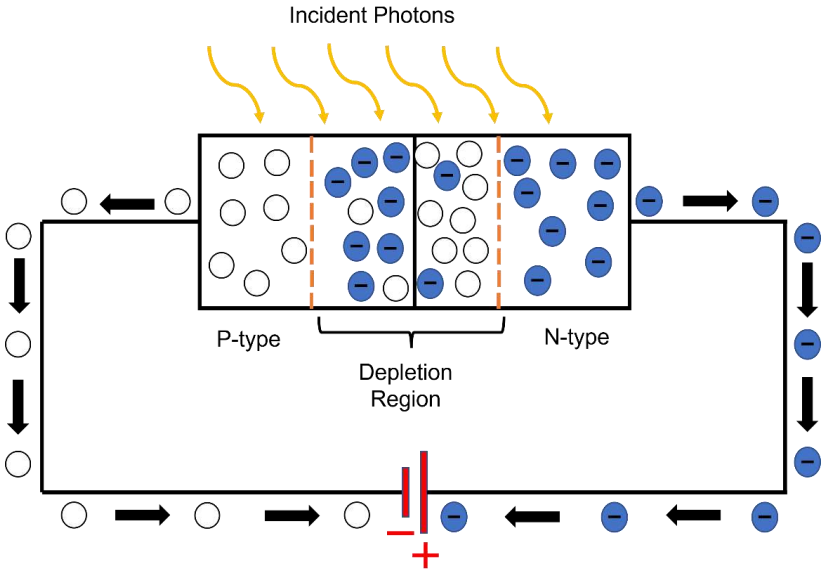


Figure 1: Photodiode

Charged Coupled Devices: Similar to the photodiodes that were described above, a charged-coupled device (CCD) sensor is also a semiconductor with the distinction that it is a metal-oxide-semiconductor as opposed to a p-n junction semiconductor. This metal-oxide-semiconductor or MOS is a field-effect transistor that undergoes an oxidation process to increase the electrical conductivity of the device. When integrating individual MOS's into arrays they act as capacitors that can transfer electric charge. Integrating this array into an external circuit creates a CCD sensor that is able to convert optical signals into electrical signals. A CCD sensor is a passive pixel sensor because the image processing has to be done externally.

A complementary metal-oxide-semiconductor (CMOS) sensor is a CCD sensor that works as an active pixel sensor. As opposed to the passive pixel sensor described above, the CMOS sensor is much faster than the CCD because the image processing can be done on the same integrated circuit. This is done because each photosensor has a photodiode that allows the pixel signal to be amplified automatically reducing the signal-to-noise ratio. The CMOS sensor also has CMOS transistor switches that allow the information to be accessed directly from the photosensor and sequentially rather than having the signal travel through an array of MOS capacitors.
The CMOS sensor also consumes less power than the CCD sensor.

PointNet++: PointNet++ is a computational approach that was developed in Stanford University by Dr. Charles R. Qi in 2007 [2, 3]. This computational analysis is based on a deep hierarchical neural network for point cloud-based systems. With the use of metric space distances PointNet++ is capable of learning local features with increasing contextual features. In Figure 2, we can see a software diagram of the PointNet++ system.

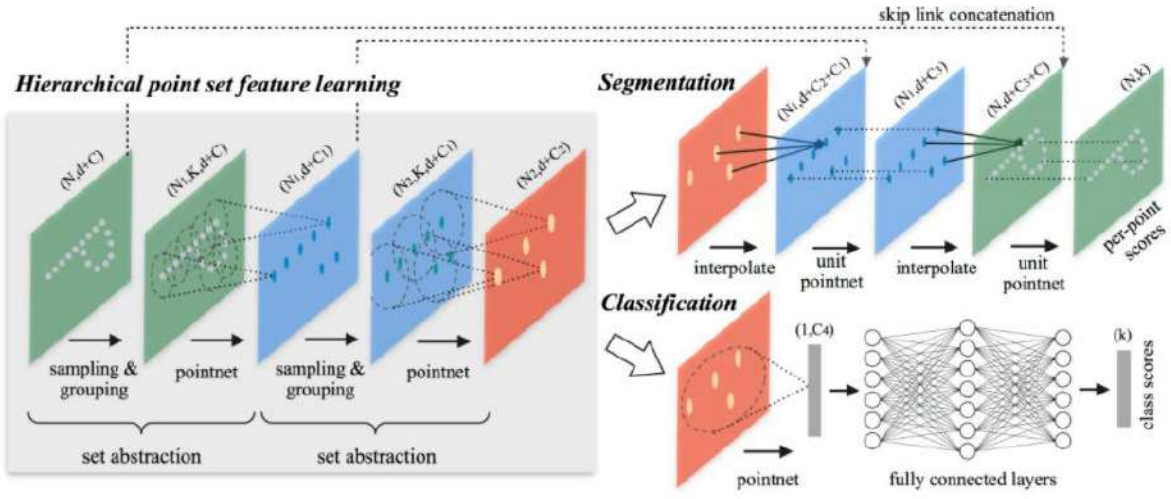


Figure 2: PointNet++ Architecture [2]

One of the most interesting applications of PointNet++ is in a research project conducted at Davidson College in North Carolina, where they are using PointNet++ to present, classify, and track identification of nuclear reactions [4]. In this research project they use a 3-dimensional space of inside the chamber and the collected data is the recorded energy or terminal velocity of the

particles after nuclear reactions have occurred. From this information PointNet++ is capable of mapping out the path of the particles as well as identifying different particles in that reaction. PointNet++ is applicable to our senior design project as it is a machine learning algorithm whose data storage and representation is a point cloud [3]. If time permits PointNet++ could be introduced into the Mobile LIDAR Scanner as stretch features, where the Mobile LiDAR Scanner can be sent into an area for recon where it can automatically identify any potential threats such as weapons, explosives, and other forms of dangerous objects.

3.3 Strategic Components and Part Selection

In this section, we will present the decision process that the group used to decide on specific components for the Mobile LiDAR Scanner. The components that required strategic selection include the detection system, the illumination system method, and the embedded system.

3.3.1 Sensing

This section goes over the decision-making process for the group to decide on a sensor for the detection system of the LiDAR. The sensor is the main component in the detection system. Different sensors serve different purposes, and some are better than others depending on their purpose. The two different sensors that the group is considering are a CCD sensor and a photodiode sensor. When selecting a sensor, the most important factors are outlined in Table 4.b. These factors are given a specific weight that is specific to the engineering requirements and specifications of this project. The CCD sensor and photodiode are compared in Table 4.a.

The criteria that are of the highest weight in the decision matrix are the FOV and active region, the quantum efficiency (QE), the operating wavelength of the sensor, and the resolution of the sensor. The FOV, as described in section 2.2 Project Features, is what dictates the range that the Mobile LiDAR Scanner will be able to detect. In order for the group to meet the specification of a maximum of 50 m, we will need to have a FOV that ranges from 10° - 45°. From the table below we see that the CCD sensor outweighs the photodiode in that category. This is because the active region of the CCD sensor is much larger.

Another important criterion is the pixel size of the sensor. As previously mentioned, a larger active region leads to more incident light that is detected leading to a larger FOV. For LiDAR applications, it is important that we are capable of capturing the most amount of light possible. Since we are measuring the reflection of objects at a far distance, the larger our active region is, the easier it is to collect far away objects that are off axis. The CCD sensor outweighs the photodiode in this category because CCD sensors are built in a 2-dimensional shape. Photodiodes are naturally 1-dimensional or just a spot to detect single photons. The CCD sensor that is being considered by the group has an active region of 160×60 pixels. Converting this to real dimensions is an active region of 3.2 mm × 1.2 mm and an active area of 3.8 mm^2. In comparison to a photodiode of similar price and same peak wavelength which has an active area of 1.5 mm^2. The group had considered a 1-dimensional photodiode array with an active region of 3.9 mm^2. Although the photodiode array has a larger active region which leads to a larger FOV than the selected CCD sensor. This photodiode array is 5 times more expensive than the selected CCD sensor and would exceed the budget constraint set by the group.

The next two criteria that will be discussed are the QE and operating wavelength. Since the illumination source that we are using is a 940 nm laser diode, we ideally want the sensor to have the highest responsivity at this wavelength. The QE of a sensor is defined as the amount of photons incident on the sensor that can be converted to electrons. A high QE at a particular wavelength means that almost every incident photon is able to be converted to electrons for electronic detection. The datasheets of the photodiode array that was in consideration give us the responsivity of the photodiode array as a function of the operating wavelength. The responsivity of a sensor is a measurement of how many electrons are created from incident photons. To calculate the QE of the photodiode array from the responsivity we can use the equation,

$$QE = (1240\ W \cdot nm/A\ ) \times \frac{R}{\lambda} \tag{2}$$

Where R is the responsivity of the photodiode and λ is the operating wavelength. For an operating wavelength of 940 nm, the responsivity is approximately 0.68 A/W, as seen in Figure 3 [5]. This results in a QE for the photodiode array of around 90%. Looking at Figure 4 [6], which is a plot of the QE for the CCD sensor with respect to operating wavelength. At the same operating wavelength of 940 nm, the QE of the CCD sensor is approximately 68%. CCD sensors typically have lower QE in the NIR spectrum than silicon photodiode arrays, but in turn, are much cheaper and can contain a higher number of elements within the same active area to achieve higher resolution. This is one of the tradeoffs that had to be considered when deciding on a sensor for the detection system for the Mobile LiDAR Scanner.

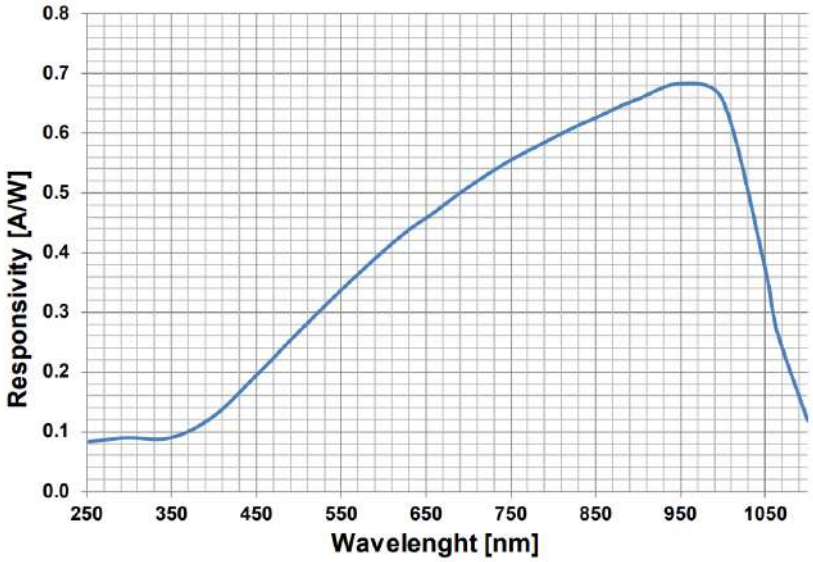


Figure 3: Typical photodiode array responsivity [5]

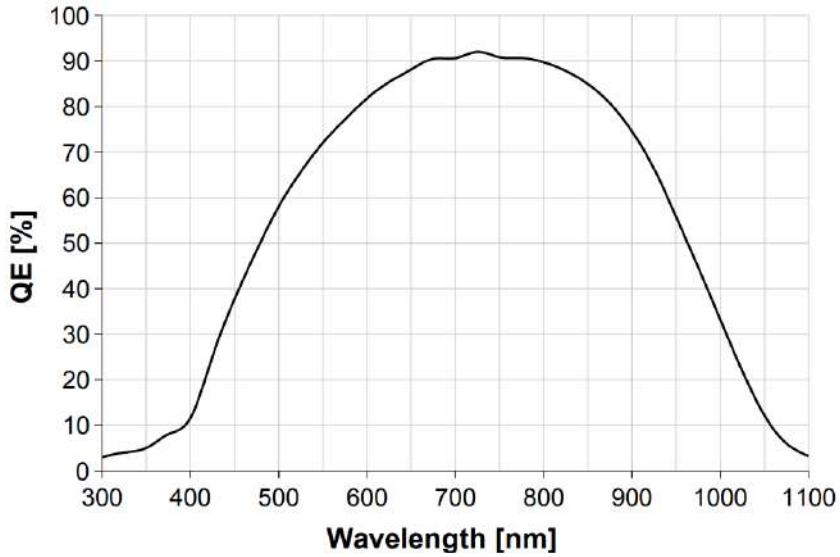


Figure 4: Typical CCD quantum efficiency [6]

The resolution of the sensor as described in Table 4.b, is the active pixel area where photons can be detected. Here we can analyze the physical construction of both sensors and determine that the resolution of the CCD sensor is much better than that of a photodiode. The CCD sensor is constructed in a rectangular figure where the active pixel area is 160 × 60 pixels which translates to physical dimensions of 3.2 mm × 1.2 mm, where the active area is 3.84 mm^2. When compared to the photodiode that the group was considering, which was a photodiode array of 1 × 16 pixels which translates to physical dimensions of 1.57 mm × 1.22 mm, where the active area is 1.92 mm^2. Another major issue that came up when deciding for the sensor selection is the shape of the beam that is produced from the collecting optics. Since we are designing the optics to have an ideally spot size radius that will be focused on the sensor, the larger our active pixel area is the more of the light that we can detect. Therefore, since the photodiode array is a thin rectangle, the spot size radius might end up being larger than the width of the array and we would lose a lot of the information. Since the CCD sensor is rectangular, even if the beam is focused off-center the beam spot size radius will still fall in the active area and all the information will be collected.

The final criteria to be discussed is the physical size of the sensor. This criterion is important to meet the design constraint of the physical dimension of the LiDAR. This constraint is listed in Table 2, where the physical size of the LiDAR is to be kept under 2500 cm^2. The photodiode outweighs the CCD sensor in this category because the photodiode, as mentioned above, can be a spot size or a 1-dimensional array. Although the CCD sensor is physically larger than a reasonably priced photodiode, the CCD sensor outweighs the photodiode overall 9 to 6 in the decision matrix. Therefore, from the decision matrix below the group has decided to use a CCD sensor as the detection device for the Mobile LiDAR Scanner.

Another crucial component when designing the detection system of the LiDAR Scanner is an optical filter. Since our project will be able to function in a dark room, in a room with light, and outside during the day and night, we need to ensure that the light that is being detected by the sensor is strictly from the reflection of our illuminating source.

Table 4.a: Decision matrix for sensor

Decision Factors		CCD Sensor	Photodiode
Criteria	Wt.		
Physical Size	1.00	1.00	2.00
FoV	3.00	3.00	1.00
Active Region	3.00	3.00	1.00
Cost	2.00	2.00	1.00
Quantum Efficiency	3.00	2.00	2.50
Wavelength	3.00	3.00	3.00
Resolution	3.00	3.00	1.50
Weighted Scores		17.00	12.00

Table 4.b: Criteria definition for Table 4.a.

Criteria	Definition
Physical Size	Determines how compact the system will be
FoV	Field of view of a single source/receiver configuration
Pixel Size	Determines the size where photons will be collected
Cost	Cost of a single source/receiver configuration
Quantum Efficiency	The amount of incident photons on a sensor that can be converted to electrons
Wavelength	Wavelength where the sensor operates at
Resolution	Active pixel area where photons can be detected
Notes: The weights used are subjective to the project's goal and application	

Using a filter allows us to reduce our noise level by eliminating background light. Three distinct forms of optical filters exist which fall in the same category as electrical filters: low-pass, high-pass, and bandpass filters. For our specific project, it is appropriate to use a bandpass filter. The

selected filter needs to have a center wavelength that is equal to the wavelength of the illumination source. This will allow us to collect all of the reflected light from the illumination source that enters the collecting optics. Figure 5 below shows the frequency response of the three types of filters mentioned above [7]. From the image, it is clear to see why we decided on selecting a bandpass filter. The selected filter has a linewidth of 20 nm and a full-wave half max (FWHM) of 10 ± 2 nm. The FWHM is the width of the spectrum calculated at half the maximum value.

One consideration that must be taken into account is that there will be a power loss from the incoming signal when we introduce the filter into the collecting optics. By looking at Figure 6, which is the transmission vs wavelength curved from Thorlabs [8], we can see the maximum amount of light transmitted at the center wavelength is only 70%. This is due to the nature of how the optical filter is constructed. If this was an ideal thin film filter that only allows one wavelength to go through the transmission would be 100% at that specific wavelength. Commercial bandpass filters are made by combining thin-film filters and the superposition of the transmission peaks at each wavelength combine to generate the transmission plot see in Figure 6 [8].

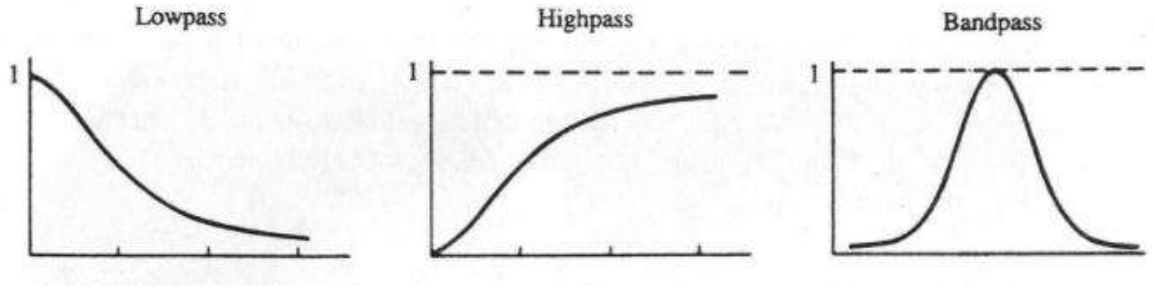


Figure 5: Low-pass, High-pass, and Bandpass filters [7]

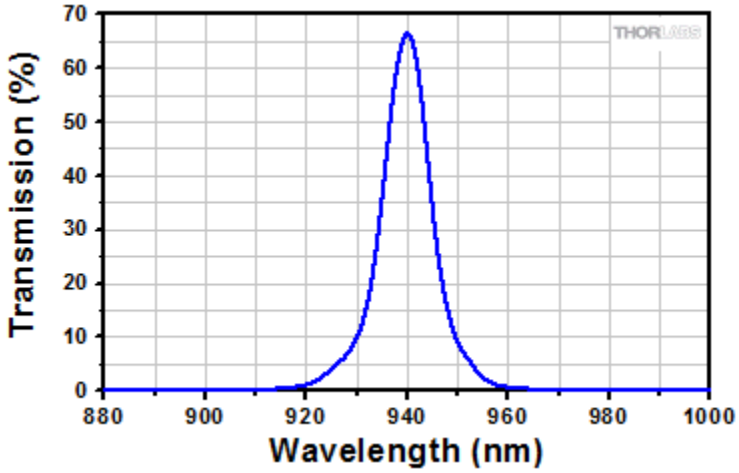


Figure 6: Transmission vs Wavelength NIR Bandpass filter [8]

3.3.2 Illumination

Wavelength: LiDAR systems typically use wavelengths in the near infrared (NIR) and infrared (IR) bands for several reasons. Firstly, longer wavelengths are safer to the eye as they are more easily absorbed by the retina, thus allowing for applications where people might be the target to be detected such as is for LiDAR-assisted autonomous vehicles. This susceptibility to absorption by water also proves beneficial for outdoor applications where solar radiation becomes a source for background noise. The sun emits light in a broad spectrum, including wavelengths in the NIR and IR bands, which, although considerably less than the emission in the visible spectrum—still could prove to be a significant source of noise. Fortunately, these wavelengths are often absorbed at different layers of the atmosphere as well as by clouds, creating dips in the solar irradiance at ground level. Figure 7 [9] shows the solar irradiance spectrum.

There are noticeable minima at various wavelengths in the spectrum, more predominantly in the NIR and IR bands. The wavelengths typically used for different LiDAR technologies are labeled. Not all are wavelengths corresponding to minima in the spectrum, however. For indoor applications, where solar irradiance does not constitute an issue, the wavelength does not need to be restricted as such. Likewise, for long-range detection where signal attenuation due to humidity represents a bigger problem than noise introduced by solar irradiance, shorter wavelengths, like the labeled 850 nm, would be more appropriate. For our system, however, solar irradiance needs to be considered as it would be used outdoors as well as indoors. Therefore, we will be focusing on the 905 nm, 940 nm, and 1550 nm wavelengths.

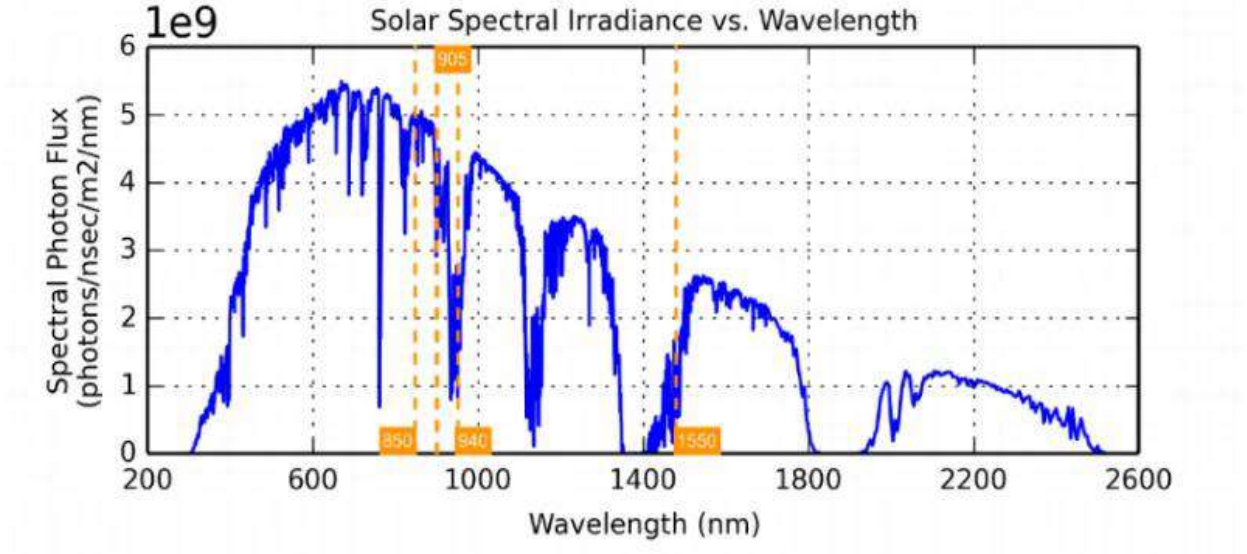


Figure 7: Solar Spectral Irradiance [9]

As discussed previously, when applications require detection of targets in humid environments, using longer wavelengths reduces the power that can be delivered to the target and detected from it. In these cases, water vapor can absorb the light from the LiDAR illumination system before it can be detected or before it can even hit the target. Therefore, special considerations need to be made when selecting the operation wavelength for our LiDAR detection system. This system must be able to detect targets outdoors, at a maximum distance of 50 m. Using the shorter wavelength of 850 nm would significantly reduce attenuation from water absorption, but it would also

introduce significant noise from solar irradiance as was concluded before. While noise reduction algorithms could be applied to diminish the noise through image processing, for high levels of noise such methods can distort the data. Since our detection range is limited to 50 m, the attenuation due to humidity can be managed by increasing illumination power. Thus, a more appropriate approach would be using a longer wavelength to reduce noise and monitoring signal strength to increase power when necessary.

Consequently, our system will require more illumination power than. Thus, another factor important to consider is the minimum power needed depending on the responsivity or quantum efficiency of our sensor at the given wavelength. The wavelength used for illumination will influence our decision when selecting a sensor, since the minimum power needed to detect an object of given reflectivity is determined by the sensitivity of our sensor to the illumination wavelength. We will discuss power considerations in further detail in the corresponding section below.

Choosing 940 nm as the operating wavelength for our LiDAR system. At this wavelength, the solar spectral irradiance is at a local minimum as seen in Figure 7, the water absorption is relatively low compared to longer wavelengths, and laser eye safety standards can be met for low intensities of light.

Illumination Sources Overview: Three illumination sources are considered in this section, Light Emitting Diodes (LEDs), Vertical Cavity Surface-Emitting Lasers (VCSELs), and Edge-Emitting Lasers (EELs). These three sources offer different emission irradiances, emission spectrums, and beam characteristics, providing various advantages for different aspects of our project. The goal of this section is to describe the different features of these sources and outline our selection process to decide which source is most suitable for our project.

LEDs are best suited for indoor applications where noise introduced by solar irradiance is negligible. For flash LiDAR applications where a broad area needs to be illuminated simultaneously, they offer wide beam illumination with a Gaussian profile, thus extra optical elements are not needed to shape the beam. In addition, LEDs are the most inexpensive of the three sources discussed here, thus many can be combined at low additional cost to obtain higher brightness if needed.

VCSELs are suited for outdoor detection applications, since narrow emission spectrums can be achieved, allowing for filtering of the light at the sensor to minimize noise from solar irradiance. The medium-wide illumination array of typical VCSEL arrays also makes it appropriate for Flash LiDAR applications, although the angular illumination of VCSELs is neither as wide nor as evenly distributed as that of LEDs.

EELs offer similar properties to VCSEL arrays. They are also suited for outdoor applications due to its narrow emission spectrum; however, its illumination area is narrower than that of a VCSEL array and the elliptical shape of the beam does not provide uniform illumination. EELs are therefore better suited to scanning LiDAR applications where a medium-long detection range is required.

Power: Under perfect environmental conditions, power considerations are mainly guided by two parameters: object reflectance of the illumination light, and sensor sensitivity to illumination wavelength. The first parameter is not one that can be controlled, so it is considered a design constraint. The detection range of the LiDAR system is therefore defined for specific object reflectances. The first parameter is one that, as discussed previously, needs to be considered when selecting the illumination wavelength on which the LiDAR will operate. The quantum efficiency (QE) of a CCD sensor from Espros Photonics Corporation (EPC) for ToF measurements is shown in Figure 4 [6] in Section 3.3.1 (Sensing), reflecting around 60% QE at 940 nm, while the QE of a typical photodiode array for the same wavelength is in the order of 85%. The light source selected should therefore provide the minimum amount of power required by the sensor of the system. For our system, as will be discussed in Section 3.3.1, a CCD sensor will be used, which suggests that our illumination source should provide about 25% more power than would be required for a photodiode array.

LED: LED powers can vary significantly, from only a few milliwatts to hundreds of milliwatts. The power is distributed over a broad area and a broad emission spectrum, thus for long-distance outdoor detection applications, they are not the best choice. Nonetheless, LED emitters can be obtained for low costs and can provide decently high powers if several LEDs are used. Figure 8 shows the emission spectrum and Figure 9 shows the angular irradiance distribution (right) for an IR LED emitter with peak wavelength of 940 nm from Würth Elektronik [10]. This LED has a maximum radiant intensity of 30 mW/sr. The sr stands for steradian, which means that the LED delivers 30 mW of power for a FOV of about 66º with most of the power concentrated in the zero-degree axial direction. As can be seen in the angular irradiance distribution, only about 50% of that power is contained within the inner 34º. In addition, the emission spectrum shows only about 50% of the power contained within a 40 nm bandwidth from the center 940 nm wavelength.

Therefore, using a 40 nm bandpass filter to reduce solar irradiance noise would cut approximately 50 % of our power, and if long distances need to be scanned, less than 50% of the power will reach targets directly in front of the system, leading to an overall power loss of about 75%. Part of this loss could be reduced by using lenses to focus the light more tightly, and by using more LEDs as their low costs allows for higher numbers to be used.

VCSEL: For higher power and narrower emission spectrums, VCSEL arrays are the appropriate choice over LEDs. Although slightly more expensive, VCSEL arrays can deliver significantly higher powers with emission spectrums as narrow as 20 nm in linewidth. The angular illumination distribution is comparable to that of LEDs, providing wide illumination areas and divergence angles over 15º. While this is an attractive feature for Flash LiDAR systems, for out project it is necessary to detect objects at long distances which requires the power to be more tightly focused, and our scanning design requires the illumination beam to be nearly collimated near the emission point to be able to spread the beam in a vertical slit with negligible horizontal width. These requirements of course can be with the appropriate optical system, as will be discussed in the corresponding section for optical design.

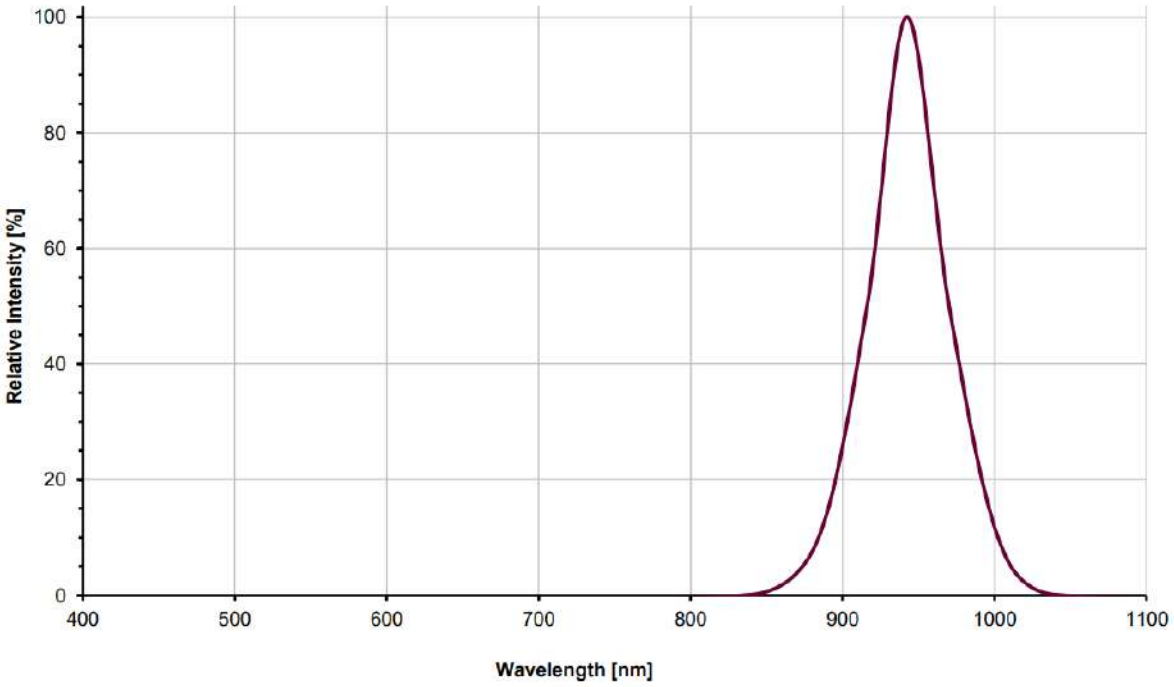


Figure 8: Emission spectrum for IR LED [10]

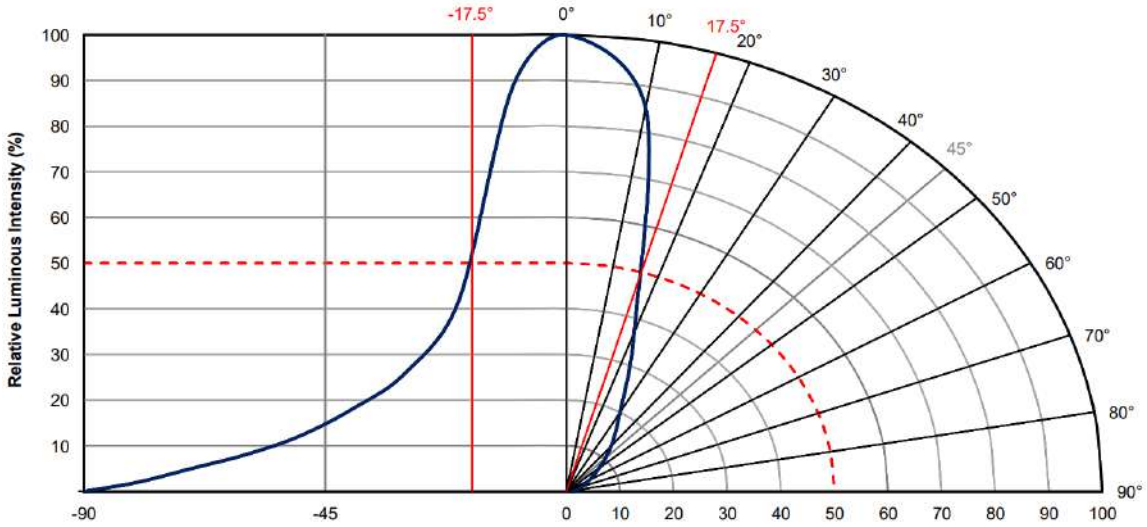


Figure 9: Angular distribution of IR LED radiant flux [10]

EEL: EELs offer the highest power out of the three sources discussed here. Varying with wavelengths, for ToF sensor purposes, EEL powers can go as high as 80 W. The emission spectrum bandwidth is comparable to that offered by VCSEL arrays, and illumination beams are typically narrower, which combined with the higher powers allow for long-range detection. Consequently, EELs are more expensive and require extra safety considerations for outdoor operation.

Light Source Modulation: Our detection system relies on time measurements between emission and collection of light pulses to calculate the distance traveled by the light and determine the object distance. For this purpose, our light source needs to be modulated. The modulation frequency (i.e., the time between pulses) affects the precision of the distance measurements and determines the maximum distance that can be detected. Higher modulation frequencies yield better precision but limit the maximum detection range to shorter distances. For our required max. distance of 50 m, the maximum modulation frequency can be determined as:

$$f_{max} = \frac{c}{2D_{max}} = \frac{3 \times 10^8 m/s}{100\ m} = 3\ MHz$$

This measurement is assuming a 50% duty cycle of the laser pulse train. As discussed previously, attenuation of the signal also decreases the maximum distance that can be detected, and higher powers are necessary to sense the target. One way of achieving these higher powers is by using short pulses. The time between pulses is limited by the detection distance at 3 MHz, but the pulse width can be made narrower by electrically modulating the source with very short on-periods. LEDs can typically withstand modulation frequencies of up to 40 MHz, thus at a 50% duty cycle, it would fit our application. However, LEDs do not provide very high powers with high directionality, therefore with such long pulses, the signal attenuation would not allow for detecting objects at 50 m away from the system. In addition, the long rise and fall times of LEDs do not allow for short pulsing, thus their modulation capabilities do not satisfy the max. detection distance requirements. VCSELs and EELs, on the other hand, can be modulated at frequencies higher than 40 mHz, and therefore allow for short-pulse modulation. Both types of lasers offer similar modulation characteristics, but they differ in temperature stability.

While pulsing a laser can reduce the heating effects, other factors like cavity structure of the laser affect the thermal stability of the device. High temperatures can cause wavelength shift, decrease in Wall-Plug-Efficiency, and shortening of device life. As discussed in the previous section, our applications require moderately high powers, for which temperature stability becomes a significant factor to consider. While VCSELs are a better choice for low-power applications, when more power is needed, EELs begin looking like better candidates.

Overall ToF Performance: We can evaluate the performance of each source discussed for ToF measurements by considering their characteristics pertaining the different parameters analyzed thus far. For determining the illumination wavelength, we considered noise introduced by solar irradiance, eye safety guidelines, and signal attenuation in humid environments, finding a balance for these parameters in the 940 nm wavelength, where there was a minimum in the solar spectrum, eye safety standards could be met for relatively low powers, and the attenuation was reduced. With this wavelength selected, the subsequent decisions are guided by this factor as well.

Considering that for accurate distance measurements in outdoor applications the solar irradiance can be minimized by using this wavelength, it is therefore important to evaluate the power offered by each source at said wavelength. LEDs provide high total radiant flux but have a wide emission spectrum and thus the total power is not entirely concentrated in the wavelength of interest. VCSELs and EELs, on the other hand, have narrow emission spectrums, and can offer a more directional optical power delivery. Further considering power, VCSELs have a wider angular

power distribution than EELs. EELs offer high powers with high directionality. For short-range detection, VCSELs are the better candidates, while for long-range detection, EELs offer the highest performance. Since our system requires both long and short-distance measurements, a hybrid of both could be used, tailoring the system for each distance. However, this is not a cost-efficient approach. For our scanning design, it is best to have a well-collimated beam that can be expanded into a vertical fan. This can be best achieved with an EEL source. Both the hybrid choice, and the scanning choice require optics to manipulate the beam; however, the former requires extra optics for collimation of the VCSEL beam in addition to the FOV varying optics. For a final decision, the parameters in Table 5 summarize the characteristics of each source at the 905 nm and 940 nm wavelengths.

Table 5: Illumination Source Comparison

Parameters	LED		VCSEL Array		EEL	
Peak wavelength [nm]	905	940	905	940	905	940
Typical radiant flux [mW]	4	4	3500	1100	8500	250
Typical radiant flux at max [mW]	1	1	2800	940	7600	230
Typical fwd. voltage [V]	1.3	1.3	3.2	2.8	3	2.2
Max. fwd. current [mA]	100	100	2600	1200	3500	230
Max. fwd. current pulsed [mA]	120	120	3000	1500	3800	260
Spectral width (FWHM) [nm]	70	45	10	9	10	10
Max. modulation freq. [MHz]	<40	<40	>40	>40	>40	>40
Pulse Duration [ns]	--	--	100	100	100	100
Duty Factor [%]	--	--	0.1	0.1	0.1	0.1
Illumination angle (full) [deg]	60	60	40	30	20	10
Wall Plug Efficiency (WPE) [%]	44	46	35	33	45	40
Operating temp. range [C]	-40 to 100	-40 to 100	-40 to 85	-40 to 80	-40 to 100	-40 to 100
Cost	Low	Low	Med	Med	Med-High	Med-High

3.3.3 Microcontroller

This section goes over how the decision over the microcontroller was decided for this project. When choosing which microcontroller to use, our team focused on ensuring that the controller would be able to handle the influx of LiDAR data that is transmitted to the computer. Our group looked at two main microcontrollers to choose from the ATmega328 microcontroller from Arduino and the MSP430FR6989 microcontroller from Texas Instruments. Table 6.a shows the decision matrix used to decide which microcontroller to use for this project. Table 6.b gives more detail on each of the criteria seen in Table 6.a.

The most heavily weighted criterion when considering both of these microcontrollers was familiarity with the product. Considering that our team must have a working prototype after two semesters of design, construction, and evaluation, we decided to heavily weigh familiarity with our choice of microcontroller. Two members of the team had previous experience with the MSP430FR6989 microcontroller from a previous class, extensively using it in lab settings. The ATmega328 was another microcontroller that was considered mainly due to the compatibility with the Bluetooth module we decided for communication. To be able to construct a working microcontroller for the Bluetooth module, diverting the pins was a task we performed to do so. The diverting of the pins would be further explained in the following sections.

The second largest weighted criteria are tied between processing power and memory. Both the MSP430FR6989 and the ATmega328 both run a RISC Architecture with a clock that can go up to 16 MHz. We concluded that either microcontroller could be used to handle the data transmission from the LiDAR scanner to the application on the computer. They also both have a memory size of 32KB, which wasn’t a great factor in choosing. With more memory that can be added to either microcontroller would then permit us to have somewhat more functionality when performing. But with more memory for more components to be working, that would then have a negative effect on the power consumption. The 32KB of memory is a standard amount for our group to work with.

Finally, the least weighted criteria are tied between power efficiency, cost, and size. Power efficiency is the ratio of the total output power to the input power expressed as a percentage. Both microcontrollers give a reasonable amount of power. The MSP430FR6989 has an invigorating property associated with it. This property is controlling the amount of power that's getting outputted. We can use the low power modes to be able to conserve power for specific tasks. The ATmega328 has a power consumption of around 26 micro amps when run at 5V, then when it runs at 3.3V it can produce 21 micro amps. Understanding the flexibility of the lower power modes, we able to change the output power depending on the functions. The microcontroller would be able keep itself in a low power mode in a idle state and use full power for more complex computations. This could be used to achieve a far better power consumption than that of the ATmega328. As mentioned in the table below, the MSP430FR6989 is the microcontroller we feel would be best suited for this device. Knowing the microcontroller would have multiple components connected to it such as the Bluetooth module, LiDAR and possibly the car for movement. Distributing the power would be a best practice to be able to have a running machine for at least 30 minutes.

The cost associated with the two aren’t a deciding factor, but it was taken into account when adding to the total cost of the system. Discussed in the table, the MSP430FR6989 was a bit cheaper at a cost of just $20. This cost was reduced when using the discount name INNOVATION. The price

when the discount was not applied was just the price of shipping, which in the end would be a deciding factor knowing the shipping was less than $10. Searching for the price associated with the ATmega328, the prices ranged from $30 - $40. This price is listed well over the MSP430FR6989 microcontroller due to ATmega328 including additional components in the package. The listed weight for the MSP430FR6989 was at 1.0 knowing the price was low for an embedded system performing many different operations in a given time. The ATmega328 would be somewhat more compatible with the Bluetooth module, but the other microcontroller can be configured in the same instance.

Size is something that has to be considered when building the embedded system of the mobile LiDAR scanner. The majority of the weight and volume of the scanner will be mostly taken up by the LiDAR scanner system itself, but to compare the sizes of both microcontroller we intend to use, we noticed the MSP430FR6989 is significantly larger than the ATmega328. The dimensions of the ATmega328 are about 35.5 mm × 8.6 mm × 3.33 mm. As compared to the other microcontroller MSP430FR6989 the dimensions are 76.2 mm × 25.4 mm × 50.8 mm. The dimensions of the second microcontroller are considerably larger, but this is expected knowing it offers more in the production stages. Also, the power distribution is more of what we can control. The effects of implementing this to our design wouldn't be a considerable constraint knowing the sizes aren't too overwhelming. The construction of our device is geared mostly towards the size of the LiDAR component; therefore, the size of the slightly larger microcontroller isn't a determining factor.

Table 6.a: Decision matrix for microcontroller

Decision Factors		MSP430FR6989	ATmega328
Criteria	Wt.		
Familiarity	2.5	5.0	1.25
Power Efficiency	1.0	4.0	2.0
Cost	1.0	1.0	2.0
Processing Power	2.0	3.0	3.0
Memory	2.0	4.0	3.0
Size	1.0	1.0	2.0
Weighted Scores		18.00	13.25

Table 6.b: Criteria definitions for Table 6.a

Criteria	Definition
Familiarity	How familiar we are operating this microcontroller
Power Efficiency	Ratio of total output power to input power
Cost	The price of each microcontroller
Processing Power	Ability to manipulate power
Memory	Stored information for immediate use
Size	The dimensions of the circuit board
Notes: The weights used are subjective to the project's goal and application	

3.3.4 Wireless communication

Some sort of wireless connection between the rendering application and the microcontroller in charge of the scanning systems is a requirement to keep our LiDAR scanner mobile. Two options that were considered were Bluetooth and Wi-Fi. Both are popular ways used by devices to connect to one another. Both options also had modules that could be connected to microcontrollers (HC-05 for Bluetooth and ESP8266 for Wi-Fi), meaning either could have been used.

Size and cost are the least weighted criteria when comparing the Bluetooth and Wi-Fi modules. The HC-05 and the ESP8266 have similar dimensions and are both relatively small. The HC-05 module has the dimensions of 26.9 mm × 13 mm. The ESP8266 has dimensions of 20 mm × 18 mm. This means that these modules will not take up too much space on the vehicle compared to other components on the scanner itself.

For cost, the Wi-Fi module typically is half the cost of the Bluetooth module, at about $5 versus the Bluetooth's typical price of $10. While the Wi-Fi module does offer a better price, the Bluetooth module offers more specifications that outweigh this advantage.

Ease of configuration, security, and input voltage range are the next weighted criteria. Ease of configuration refers to how much effort is needed to configure the settings of each module. For the HC-05 module, it can be switched to either slave mode or master mode using an on-board button. When in master mode, the module will be able to read attention (AT) commands to set the baud rate or the pairing password used between devices. These AT commands would be transmitted using a device connected to the module through a serial terminal or a Bluetooth terminal for a smart device. For the ESP8266 module, it can be configured in a similar manner to the Bluetooth module. The configuration mode on the Wi-Fi module is done through a serial terminal on a smart device as well.

Input voltage range refers to the range of supply voltage needed to power these modules. Typical microcontrollers offer two voltages to power peripherals: 3.3 V and 5 V. Depending on the specific ESP8266 module, it may come with an internal divider allowing it to have an input voltage range from 0 to 3.3V. The HC-05 module has an input voltage range of 3.3 V to 6 V. The specific HC-05 module that we are using in development mentions that Logic 0 and Logic 1 are defined as 0 V and 3.3 V respectively. In the case of this Bluetooth module, where we must use the 5 V rail from the microcontroller, the appropriate voltage divider can be used to supply the correct voltage for logic purposes.

Security is a criterion that must be considered. We found the Wi-Fi module to pose too much of a security risk, because there can be multiple devices on a Wi-Fi network that could potentially intercept the ESP8266's transmission. With the HC-05, we know that connection is between the smart device and the module.

Table 7.a: Decision matrix for wireless communication module

Decision Factors		Wi-Fi	Bluetooth
Criteria	Wt.		
No requirement of internet connection	3.0	0.0	6.0
Data rate	1.75	3.5	1.75
Security	1.5	1.5	3.0
Input voltage range	1.5	1.5	1.5
Ease of configuration	1.5	1.5	1.5
Size	1.0	1.0	1.0
Cost	1.0	2.0	1.0
Weighted Scores		11.25	15.75

Data rate and the requirement of an internet connection were the most important factors to consider when choosing between the two modules. The volume of LiDAR data read correctly by the LiDAR scanner heavily influenced the data rate of the modules. The data rate of the Bluetooth module is 3 Mbps and the data rate of the Wi-Fi module is 7 Mbps. This means that the Wi-Fi module has more than double the data rate of the Bluetooth module, thus making it the preferred communication module.

However, this criterion is overruled by the most weighted criterion: the requirement of internet connection. We discussed the possible applications of this mobile LiDAR for military reconnaissance or archeological exploration. More than likely, the locations for these applications would not have a stable internet connection that would allow a ESP8266 to function. The HC-05 module however does not require an internet connection to function.

Table 7.b: Criterion definitions for Table 7.a

Criteria	Definition
No requirement of internet connection	Whether an internet connection is required for the module to operate
Data rate	How fast the data can be transmitted
Security	How likely is the data to be seen by non-users
Input voltage range	Amount of voltage
Ease of configuration	Effort required to initially configure the module
Size	The dimensions of each module
Cost	The price of each module
Notes: The weights used are subjective to the project's goal and application	

3.4 Possible Architecture and Software Designs

3.4.1 Collecting Optics

This section goes over the decision-making process for deciding on an optical schematic for the detection system. The optical components and design are what will drive the LiDAR as it is how light will be distributed out as well as how the reflected light will be collected.

For the detection optics, the two optical designs that were considered are two or three-lens systems. Both options would include the same components with the major difference being that the two-lens system consists of two positive focal length lenses. The three-lens system consists of a negative focal length lens sandwiched in between two positive focal length lenses. The criterion with the largest weight to it is cost. Since we have a limited budget discussed in section 4.2 Design Constraints, when we add more optics to the system, we are slightly altering the other parameters by a couple of millimeters or milliradians, when we can increase the cost by almost 33 %. In order to satisfy the other parameters and create an optical system that is capable of capturing light from the other parameters, we need to sacrifice cost and look at the other criteria.

The next important criteria discussed in the decision matrix for the detection optics Table 8 are FOV, total focal length, and resolution. The FOV and total focal length are two criteria that are dependent on one another. In order to reach a FOV outlined in Table 2, we need to calculate the needed focal length. This mathematical calculation is done in the section 5.2.3.2 detection system. The desired total focal length that will give us a maximum FOV of 45° is 3 mm focal length. For the FOV category, the three-lens system is more favorable as the inclusion of the third lens allows us to capture all the light even from the maximum FOV angle of 45°. This is also due to the fact that in the three-lens system the diameter of the negative focal length lens is much larger and therefore can collect light that is otherwise lost in the two-lens system. With a larger lens diameter, the physical numerical aperture (NA) of the system increases.

When looking at the total focal length of the system, the two-lens system outweighs the three-lens system, but only by a relatively small amount. This is because the two-lens system allows us to get the theoretical focal length ± 0.5 mm. The three-lens system has a larger error for the total focal length of the system by ± 2.5 mm. This is a limitation of the commercially available concave optics. The smallest lens available that has a negative focal length is one with a diameter of 6 mm and a focal length of -6 mm. In order to bring the total focal length of the collecting optics down, we would ideally want a smaller focal length for the concave optics. Since none are available, we would need to get one custom made which would make us exceed the budget by too much. The final category of the detection optics for the Mobile LiDAR Scanner is the physical size of the optics. This category is important for creating the housing of the LiDAR and the NA as mentioned above. One limitation that will be present when creating the housing using a 3D printer is the precision of the printer itself. The smaller the optics the smaller the housing will need to be to keep the optics stationary. Larger optics require a larger housing and therefore it would be easier to overcome the limitations of the 3D printer precision.

Table 8.a: Decision matrix for detection optics

Decision Factors		Two-Lens System	Three-Lens System
Criteria	Wt.		
FoV	2.00	1.00	2.00
Total Focal Length	2.00	2.00	1.75
Cost	2.50	2.00	1.00
Robustness	1.00	1.50	1.00
Physical Size	2.00	0.50	2.00
Numerical Aperture	2.00	1.00	2.00
Weighted Scores		8.00	8.75

Table 8.b: Criteria definitions for Table 8.a

Criteria	Definition
FoV	Field of view of a single source/receiver configuration
Total Focal Length	From theoretical calculations, this is the focal length that will give us the desired FOV
Cost	Cost of the total optics system
Robustness	Ease of creating a 3D printing housing for the optics, This is considered from the diameter of the optics and the distance of the optical optics.
Resolution	The spot size of light incident from different FOV angles
Notes: The weights used are subjective to the project's goal and application	

3.5 Part Selection and Summary

In this section, we will go over the selected components for the Mobile LiDAR Scanner by summarizing the sections presented above. In this section we will describe specific components that will help us construct the Mobile LiDAR Scanner. Table 9 presents the specific components that were selected from the decision matrices created by the groups as well the specific part number of each component.

Table 9: Selected Components

System	Component	Part Number
Sensor	CCD Sensor	EPC635-CSP44-002
Illumination Source	Laser Diode	TBD
Microcontroller	Texas Instruments	MSP430FR6989
Wireless Communication	Bluetooth	HC-05
Lens 1	Biconvex	ED #67-595
Lens 2	Plano Concave	ED# 67-982
Lens 3	Biconvex	ED #67-595

4.0 Standards and Design Constraints

4.1 Standards

As engineers, it is important to be aware of the standards and adhere to standards set by the greater engineering community. These standards are put in place to protect the well-being of the public that interacts with the product as well as holds the engineers that design the product accountable. Table 10 below shows what standards that we plan to adhere to while designing the Mobile LiDAR Scanner.

4.1.1 Laser standards

Laser safety standards are important to follow to prevent harmful injuries to the eye and skin for both operator and bystanders. Table 10, which is shown above, gives an outline of the major laser standards that the group will follow. IEC 60825-1:2021 is a standard set by the International Electrotechnical Commission to ensure that the lasers that are being operated and used follow safety guidelines. These guidelines include how to properly guide a laser and what is considered to be correct personal protective equipment (PPE) needed to operate a laser.

The next two laser standards are given by the American National Standard Institute (ANSI) and are backed by the Laser Institute of America (LIA). These standards will provide guidelines for the group to create a safe product. The main laser standard is ANSI Z136.1, this standard is the way to classify lasers. Different lasers are classified in different classes depending on the power level that can be achieved as well as the level of harmfulness to the human body. ANSI Z136.6 is considered to be the parent standard in lasers as it outlines the protective housing needed for specific laser classes, the standard operating procedures for specific laser classes, and the duties and responsibilities of the laser safety operator (LOS).

The second standard that is important for the application of the Mobile LiDAR Scanning System is ANSI Z136.6. This standard put forth by ANSI and LIA states the safe use of lasers in an outdoor environment. Since the project will be used in a common area where bystanders may or may not be present. This requires the group to select a laser in a non-hazardous class such as a class 1 or 2 laser. ANSI Z136.6 will also give guidelines to the nominal ocular hazard distance and specular reflection that can occur from scanning the area.

4.1.2 Electrical component standards

IEEE 802.15.1-2005 is a Bluetooth standard set by the Institute of Electrical and Electronics Engineers (IEEE). This standard updates a previous standard, IEEE 802.15.1-2002, which gives the standard for low-power, low-complexity wireless connections between devices in a Personal Operating Space (POS). The POS is defined as 10 meters in all directions around a person that is stationary or moving. For the Mobile LiDAR scanner, communications between the PC application and the LiDAR system are done wirelessly using Bluetooth. We will need to adhere to this standard to maintain interoperability between the two devices.

IPC J-STD-001 is a standard given by IPC, a global association that aims to assist electronics manufacturers (PCBs, OEMs, and EMS) and give better electronics to the suppliers. This standard gives descriptions about what materials and methods are appropriate to use. It also set the acceptance critters for soldered electrical and electronic products. Given that a PCB is needed to be designed for this project, we will need to keep these standards in mind.

Table 10: Table of standards

Standard	Description
IEC 60825-1:2021	Safety of laser products
ANSI Z136.1	Laser classification
ANSI Z136.6	Requirements for outdoor laser operation.
IEEE 802.15.1-2005	Provides standards for low complexity, low-power wireless connections for devices in a POS; defining the POS
IPC J-STD-001	Requirements for Soldered Electrical and Electronic Assemblies

4.2 Design Constraints

In this section we present the design constraints that the group will face.

4.2.1 Time and economic constraints

Arguably, the two biggest constraints that the team faces are time and money. Since the project is to be designed and built in two semesters (about eight months, not counting semester breaks), it is extremely difficult to include all three types of features listed above.

The team is also financially responsible for one hundred percent of the project. Therefore, the group must come up with a design that allows the Mobile LiDAR Scanner to always have optimal resolution and computational time with a budget constraint. The team has set a budget of $1200.

4.2.2 Environmental, social, and political constraints

From an environmental standpoint, the product does not cause any harm to living beings and to the environment it encounters. It is important that our scanner does not leave parts as it moves about. The scanner's components should be attached such that they do not fall off during a scan.

Our team envisioned our LiDAR scanner to assist reconnaissance during military operations by providing a 3D scan of an area. This 3D area scan can be used to better plan out movement of soldiers. While our LiDAR scanner does provide helpful information that can reduce the number of casualties, some may say this product contributes to the "gamification" of war. Microsoft had made a deal with the US Army in 2021 to create augmented reality glasses to serve as a combat

assistant and training tool for $22 billion dollars [11]. The glasses provided a soldier a view similar to one seen in first-person shooter video games, complete with 3D terrain maps, a heads-up display, and even outlining fellow soldiers. With the addition of our own mobile LiDAR scanner, the 3D scans could be used in tangent with the glasses, providing 3D environment scans for missions inside unknown buildings immediately to soldiers.

Following the announcement of the deal, many Microsoft employees expressed their concerns of the technology that they developed to help the military and "gamifying" war [12]. The gamification of war refers to how with these new technologies, it becomes easier for soldiers to remove the "human" aspect of hostiles, seeing them as only targets to eliminate.
It is always important to realize the ramifications of your product, even if these consequences were intentional or not. Our team sees the mobile LiDAR scanner simply as a way to get 3D scans of an environment that may or may not be accessible to humans. Applications for this technology could be used for military reconnaissance or archeological exploration. However, the main purpose of the scanner is to ensure the safety of human lives, whether they are soldiers or archeologists.

4.2.3 Ethical, health, and safety constraints

When looking at the ethical, health, and safety constraints that our group faces is the laser light that is being used. Since the Mobile LiDAR Scanner will be operating outdoors and indoors, it is important to consider how the laser light will affect possible bystanders. The group needs to ensure that the selected laser is not harmful in the case that the laser light was to enter a bystander's eye. We need to ensure the proper laser safety standards, described in Table 10.

4.2.4 Manufacturability and sustainability constraints

Table 11: Project Constraints

Constraint	Description	Unit
LiDAR size	LiDAR must be able to sit on top of RC car	< 2500 cm^3
Image Resolution	Due to the power of the laser, lens sizes, and sensor size	TBD
Laser Safety	Since the project will be used in places with bystanders, lasers cannot cause damage to bystanders' eyes.	N/A
Project budget	Project is entirely funded by group members.	$1200
Working product	Be able to have a product that meets the specifications and basic features.	One working prototype

The two other constraints summarized in Table 11 are engineering constraints. These include the overall size of the LiDAR system and the power distribution along the components. LiDAR size is a constraint because we need the LiDAR to be small enough to be portable on top of an RC car. The final constraint deals with the resolution of the final image. This feature is a constraint because of all the factors that need to be taken into consideration. These factors include the output power of the laser, the size of the lens and sensor, and even the reflectivity of the material. Table 11 highlights the constraints that the group has encountered.

5.0 Project Hardware and Software Design

5.1 Initial Design Architectures and Related Diagrams

In this section, we will present the schematic design of the entire Mobile LiDAR Scanner as a block diagram and the hardware architecture of the Mobile LiDAR Scanner. Furthermore, we break down the main subsystems of our project and discuss their specific roles within the product. Here we will also present mathematical and software simulation that was used in the design process of the Mobile LiDAR Scanner. Finally, this section will present the software libraries that will help the group complete the objectives of this project.

5.1.1 Overall Project Diagram

Here we present the block diagram of the entire Mobile LiDAR Scanner and the tasks that each member has taken the responsibility for. Figure 10 shows the overall design of the project. The Mobile LiDAR scanner consists of three main parts: the LiDAR scanning system, the RC car itself, and the PC application used to interface with the LiDAR scanning system. This block diagram is further broken down into the roles that each member of the group will take lead on. Each block also has a label to describe what stage of the design process it is on.

Figure 10 breaks down the systems that each member of the group is responsible for. Beginning outside of the LiDAR box, Troy Morgan (CpE) will be taking charge of designing the power distribution for the entire Mobile LiDAR Scanner. The power supply is what will allow the laser to emit light, the sensor to collect the information, the rotation stage for the LiDAR to scan, and the Bluetooth module to transmit information for data processing. Dannah Dolorfino (CpE) will be in charge of transferring the collected information from the sensor to the user interface where the image processing will occur, and the final 3-dimensional model will be rendered and displayed.

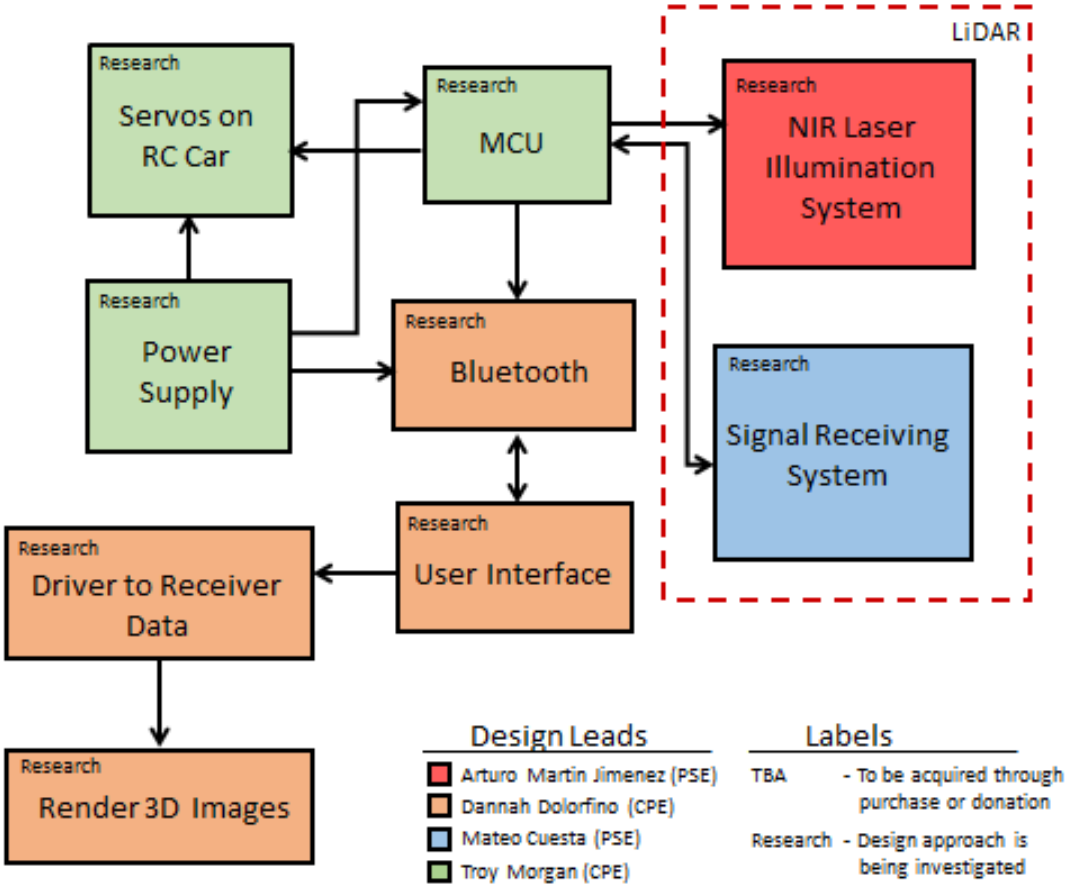


Figure 10: Overall project block diagram

The LiDAR system was broken down into its two major subsystems: the illumination system and the detecting or connecting system. Arturo Martin (PsE) will be in charge of designing and building the illumination system. This system is composed of the emitting optics, which are the optical components that will expand the beam to scan the room, and the illumination source. The detection system of the LiDAR will be designed and built by Mateo Cuesta (PsE). This system consists of the collecting optics, which will collect the reflected light from the illumination source. The second part of the system is the sensor which will convert the photons to electrons to create the 3-dimensional model of the area.

5.1.2 LiDAR System Schematic

In this section, we provide a schematic of the LiDAR system that will be used for the Mobile LiDAR Scanner. Figure 11 shows the optical layout of the Mobile LiDAR Scanning System. This layout is composed of two subsystems: an illumination system and a detection system. As shown

in Figure 10, Arturo Martin and Mateo Cuesta will take the lead on this design distributing the workload as mentioned in the section above and visualized in Figure 10.

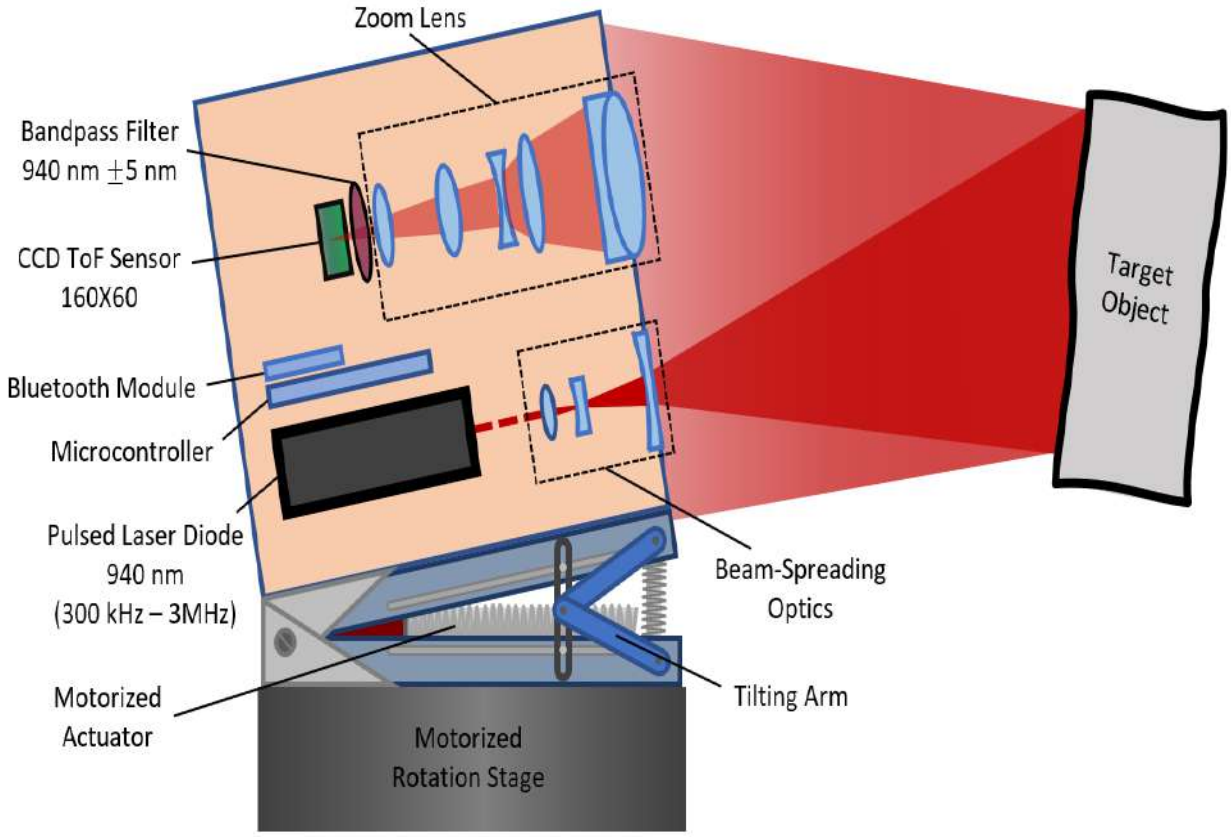


Figure 11: Scanning LiDAR Diagram

5.2 Hardware Design

In this section the group will decompose the Mobile LiDAR Scanner into its various subsections headed by a group member as described in Figure 10. Each subsection will describe the design challenges and solutions to the challenges for each individual subsection of the Mobile LiDAR Scanner.

5.2.1 Illumination System

The light source is the heart of the illumination system. For our LiDAR we need to pulse the illumination light to illuminate our target. The scattered pulsed signals from the target will then be captured by the detector to measure the distance. As discussed previously, the pulse width and frequency determine the maximum distance that can be detected, and the pulse width determines the longitudinal resolution of the system. In this section we will discuss how the illumination source is driven and modulated for different detection distances and illumination FOVs.

The longitudinal resolution is determined by the pulse width of the illumination light, as well as by the gating settings of the sensor. We will try to optimize the resolution firstly considering only pulse width, and then for data acquisition the gating settings of our detection system will be optimized around the pulse characteristics. Longitudinal resolution is the range of distances in the axial direction of the propagating beam that can be distinguished from a single pulse measurement. Considering a single pulse of width Δt is sent out, the light will propagate a certain distance Δd in that time. Since the distance measurements depend on the time difference between the launch of the pulse and its detection, no discrimination can be made for time intervals shorter than the pulse width, assuming our sensor shutter is permanently open. Therefore, the minimum distance that can be resolved is Δd. Equation 3 can be used for a simple approximation of the longitudinal resolution of our system, neglecting any sensor settings.

$$\Delta d = \Delta t \frac{c}{2} \tag{3}$$

This equation suggests that shorter pulse widths yield better resolution; however, there are limitations to how short a pulse we can achieve based on our laser properties. In addition, as will be later discussed, the pulse width should be compatible with the sensor shutter time. A pulse too short could be missed in the closed period of the shutter.

The pulsing frequency, or the time between pulses, in the simplest case determines the maximum distance that can be detected. The illumination power and reflectivity of the target, however, are the main limiters of the detection range. Equation 3 can also be used to determine the maximum modulation frequency for a given detection range. For example, for a detection range of 50 m, a maximum modulation of approximately 3 MHz is allowed. If the pulses are modulated at higher frequency, light reflected by objects located within the travel distance of a pulse in the period of modulation will reflect the light before objects farther away, and therefore the pulses reflected by farther objects will not be correctly associated with their time of flight.

Let us then consider the limitation for the laser modulation. EELs can typically be modulated at high rates, which allows us to get very short pulses. What limits this modulation is the relaxation oscillation. The modulation frequency cannot be greater than the frequency of the relaxation oscillation, which can be determined using Equation 4.

$$f = \frac{1}{2\pi} \frac{1}{(\tau_{sp}\tau_{ph})^{1/2}} (I/I_{th} - 1)^{1/2} \tag{4}$$

Here, τ_{sp} and τ_{ph} are the stimulated carrier lifetime and the photon lifetime, respectively. The stimulated carrier lifetime τ_{sp} is typically in the order of 10 ps while the photon lifetime τ_{ph} depends on the refractive index of the cavity and the threshold gain g_{th} as shown in Equation 5. For typical laser diodes, the oscillation relaxation frequency limit is a few gigahertz.

$$\tau_{ph} = g_{th} \frac{c}{n} \tag{5}$$

It is important to consider the temperature effects on the output power of the laser when operating it. Equation 6 shows that the threshold current increases with temperature, and consequently the output power decreases as well. Maintaining the temperature within the operational range of the laser is important for efficiency and will allow us to achieve higher illumination powers. This will be especially important when detecting targets at longer distances, where more power is required.

$$I_{th} = I_z \, exp(T/T_0) \tag{6}$$

In this equation, I_z is the threshold current at 0 K, and typically determined through extrapolation. T_0 is the threshold temperature coefficient. The higher T_0 the less the threshold current increases with temperature.

5.2.2 Detection system

The second subsystem of the LiDAR Scanner is the detection or sensing system. This system is characterized by the collecting optics and the sensor. The detection system is what defines the FOV of the LiDAR system. The FOV is defined by the focal length of the collecting optics with Equation 7.

$$FOV = 2 \times arctan\left(\frac{H}{2f}\right) \tag{7}$$

Where H is the height of the sensor and f is the focal length of the collecting optics. For our project one of our design objects, listed in Table 2, is to be able to have a vertical FOV that can vary from 10° to 45°. There are two methods that can be implemented to vary the FOV. The first is by keeping the height of the sensor fixed and varying the focal length of the system. Since LiDAR is collecting distance measurements rather than an actual image, having a defocused system will give the same information as a focused system. The second method to vary the FOV is to change the height of the sensor while keeping a fixed focal length. Since the sensor has a set height the way to implement this is by placing an aperture in front of the sensor. This aperture would most likely be an iris that can be opened or closed depending on the user's preference.

The group opted for a design that implements both of these methods in order to meet budget constraints. In order to change the focal length of the collecting optics would require either a zoom lens system or mechanical components that would move the collecting optics. The second method of adding an electrical iris to the system is not possible as we would need to custom order an iris that meets our specification for the size of the LiDAR. This would not be possible with the set budget. Therefore, in order to meet the engineering specifications in Table 2, the collecting optical design will have a fixed focal length and the casing will act as an aperture stop that will set the height of the sensor.

Using Equation 7, the calculated focal length for a 10° FOV is 18 mm and for a 45° FOV is 3 mm. Through the design process, we simulated the collecting optics through Zemax. Zemax is an optical design simulation software that models the optical path taken by the light through a system of lenses. In this design, it was concluded that if the effective focal length (EFL) of the collecting optics is 3 mm and the entrance pupil is proportional to the height of the sensor then, light can be captured for the entire range of FOV as described in Table 2.

After discussing the FOV the second important factor of the sensing system is the collection of photons. Since we are measuring reflectivity of objects, our importance is not the resolution of the sensor but rather capturing enough photons. The more photons that can be captured, the easier it is to distinguish the reflectivity from the ambient noise. One design method to limit the ambient noise entering the system is to place a bandpass filter with a maximum at the operating wavelength. This will block atmospheric light and other sources that can interfere with our measurements.

The resolution of the LiDAR system is dependent on the FOV, the distance of the measurement, and the number of points that are being illuminated by the source onto the object. One of the unique features of the Mobile LiDAR Scanner is the user selective scanning distance. For different distances of the scan, the Mobile LiDAR Scanner will use different methods of illumination. Depending on the distance of the scan the resolution will vary by a small factor. The reason for having two different methods of illumination depending on the selected scanning distance is such that the rendered image of the area has an optimal resolution. Our goal is that for the selected distance the Mobile LiDAR Scanner can illuminate as much of the object in each scan. Another design component that will help the group improve the accuracy of the Mobile LiDAR Scanner is the filter. By filtering out the majority of the noise, we are making sure that the optical signal that is present on the filter is from reflections that were illuminated by the source.

5.2.3 Optical System

For the Mobile LiDAR Scanner, there are two distinct optical designs, one for the illumination system and one for the detection system. These two designs will work cohesively, the illumination optics providing the specifications around which the sensing optics are designed. The FOV of the illumination system determines the area that will be scanned as well as the detection range targeted. Although the way the light is reflected or scattered from the object cannot be controlled, the power delivered by the illumination system to the area of interest will be proportional to the number of photons that reach the detector after hitting the target. With this in mind, it is important then that the optical properties of the illumination system are complementary to the optical requirements of the sensing system.

When working with a lens system with multiple optics the distance between the lenses changes the way the beams propagate through the system. To calculate the distance between the lenses the group will be calculating the ABCD matrix for the three-lens system. The ABCD matrix is a 2 × 2 matrix that describes the effect that the optical surface has on the propagating beam. One important criterion that must be considered when analyzing the optical system with the ABCD matrix is that the measurements have to be done in paraxial calculations. To calculate the ABCD matrix we need to understand the reflection and translation matrix of an optical system.

$$\mathbb{R} = \begin{bmatrix} 1 & 0 \\ -\frac{1}{f} & 0 \end{bmatrix} \tag{8}$$

$$\mathbb{T} = \begin{bmatrix} 1 & \frac{t_i}{n_i} \\ 0 & 1 \end{bmatrix} \quad (9)$$

The first matrix, Equation 8, represents the refraction matrix of the optical system. The second matrix, Equation 9, is the translation matrix of the optical system. Where t_i is the thickness of the lens and n_i is the refractive index of the lens. For light propagating through free space, n_i is the refractive index of air which is 1. The translation matrix becomes the translation matrix for free space where t_i, is the separation between the lenses. By knowing the focal lengths of the lenses and the final image height that we need, we can solve for the distance between the three lens systems.

5.2.3.1 Aberration Correction

For a LiDAR system, aberrations are not as crucial as for classical imaging systems. Chromatic aberrations, specifically, are not a relevant issue as the LiDAR relies on a single illumination wavelength with narrow linewidth. The main focus of the optical systems in LiDAR detection is typically to collect as much of the scattered light from the target as possible. Single-point scanning systems have the lowest optical requirements, in this sense, as they rely on detecting the distance of objects with point-inspection (illumination). For Flash LiDAR, however, the optics at the sensor become more important, directing beams from various angles to a corresponding pixel on the sensor in order to create a 3D matrix of information where the where the first two vector elements represent the coordinates of the point on the sensor (x,y) and the last element represents the measured distance at that point, rather than intensity. In this way, Flash LiDAR can create 3D maps using a 2D sensor. Consequently, Flash LiDAR need to account for certain aberrations in their sensing optics. Aberrations like field curvature, astigmatism, spherical aberrations, and distortion can affect the image formed at the sensor and need to be minimized. Our design, while not a Flash LiDAR design, employs similar imaging methods to create the 3D map of the scanned area. While scanning one vertical line at a time reduces the effect of aberrations, their impact on the image is still not negligible.

In this section we will discuss the methods and formulas used to calculate and minimize the relevant aberrations. These aberrations are all wavefront aberrations, which are additive, and can thus be corrected through their total sum throughout the system. Using Seidel's formulas for 3rd order aberrations, we can determine the parameters in the optical system that affect each aberration, optimizing them to achieve the minimum value possible of total aberrations. As these formulas will show, most wavefront aberrations depend on the height of the marginal and chief rays, which are at maximum and minimum, respectively, at the aperture stop. Therefore, elements around the aperture stop will influence the total aberration sum the most, so designing our optical systems around the stop, with some level of symmetry, will help us reduce aberrations.

Spherical Aberrations: As the name suggests, spherical aberrations are caused by spherical surfaces. Rays incident on spherical lenses at high or low off-axis points are not focused at the same distance as are rays closer to the axis. This creates an extended focal length and the image blurred decreasing resolution. In a LiDAR system, this reduction in resolution will mix distance information from several points in the object space depending on the severity of the aberration as

the root mean square (RMS) spot sizes are increased potentially to be greater than the pixel size leading to under-sampling. Equation 10 shows the Seidel coefficient for spherical aberration.

$$W_{040} = \frac{S_I}{8} = -A^2 \mathrm{y} \Delta \left(\frac{u}{n}\right)/8 \tag{10}$$

In this equation, y is the height of the marginal ray at the surface, u is the angle of the incident ray with respect to the optical axis, n is the refractive index where the incident beam propagates before reaching the surface, $\Delta\left(\frac{u}{n}\right)$ is the aplanatic deviation calculated as:

$$\frac{u_{i+1}}{n_{i+1}} - \frac{u_i}{n_i},$$

where $u_i$and u_{i+1} are the marginal ray angles before and after the surface, respectively, and $n_i$and n_{i+1} are the refractive indices before and after the surface. See Figure 11 for reference. A is the ray refraction invariant calculated as:

$$A = n(yc + u), \tag{11}$$

where c is the curvature of the surface, defined as the inverse of the radius of curvature. This refraction invariant is defined using Snell's Law in the paraxial approximation, then A/n is the angle of incidence on the surface with respect to the normal to the surface.

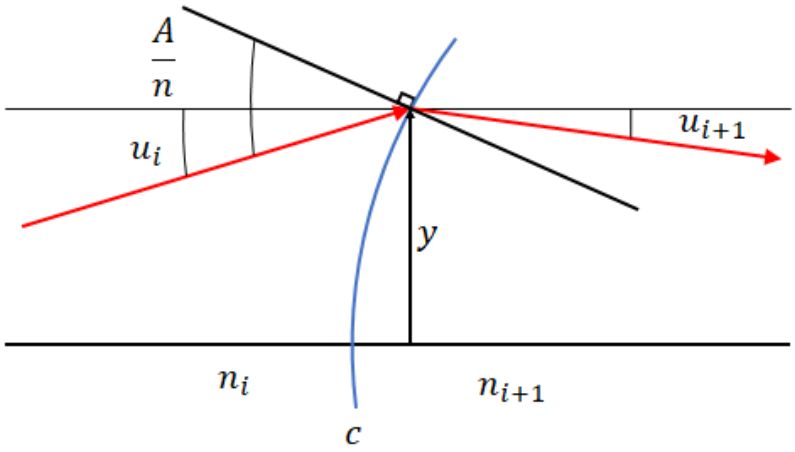


Figure 12: Aplanatic deviation and refraction invariant

As can be seen in Equation 10, spherical aberration is highly dependent on the angle of the marginal ray with respect to the surface normal. This angle varies with height and for rays parallel to the optical axis, spherical aberration increases with height. This relationship suggests that to decrease spherical aberration on a single surface, the height of the marginal ray can be decreased, or the curvature of the surface, to decrease the angle refraction invariant. This might not always be beneficial since decreasing the marginal ray height indicates decreasing the NA, and if we want to capture more light, the NA must be as large as possible. Nonetheless, this provides a basic

principle of power splitting, to divide the total power of the system among several lenses, reducing the surface curvatures, and minimizing spherical aberrations at each surface. More importantly, however, is the effect of the aplanatic deviation since it can change the sign of the aberration index at each surface. Making some indices negative and others positive minimizes the total aberration when added. This idea will be applied to the other aberrations as well.

Astigmatism: When rays propagating in different optical planes focus at different distances from the lens, this is called astigmatism. For our application, ideally, this should not be a significant issue, as we are scanning objects on a vertical line. Astigmatism would be negligible if all the light would be reflected from the target along the same optical plane; however, this is not always the case, as the scattered light from the target could propagate in many random directions, hitting the collector lens from the detection optics from various optical planes. Therefore, we need to minimize astigmatism in our system.

$$W_{222} = \frac{S_{III}}{2} = -\bar{A}^2 y \Delta \left(\frac{u}{n}\right)/2 \tag{12}$$

Equation 12 shows the Seidel coefficient for astigmatism. Notice that it is very similar to the equation for the spherical aberration Seidel coefficient, with the exception that instead of the marginal ray refraction invariant, now the chief ray refraction invariant is used. This suggests that a similar approach to that used to correct spherical aberrations can be used to reduce astigmatism. Chief rays originate from the edge of the object at an off axis point and have a height of zero at the aperture stop, thus without decreasing the FOV of the system, astigmatism correction relies mostly on power splitting to reduce surface curvatures and optical symmetry around the stop to cancel out coefficient values.

Coma: One way in which spherical aberrations can be corrected for rays parallel to the optical axis is by using parabolic (aspheric) surfaces. With these surfaces, however, off-axis rays experience strong coma. Coma also occurs in systems with spherical surfaces, and it causes parallel off-axis rays to focus at different heights on the image space, creating a vertical shift on the focal points that can appear as a smear or a tail in the image. The Seidel coefficient for coma aberrations is shown in Equation 13. In this one, both the marginal ray and chief ray refraction invariants are used, so there are more degrees of freedom when it comes to minimizing the total coma through the sum of the coefficients of individual surfaces. Typically, the chief ray in a system crosses the optical axis only at the aperture stop if there are no images of the stop within the system. Therefore, the chief ray changes sign after crossing the axis and so do the signs of the coefficients for the subsequent surfaces. This allows for cancellation of the comma by elements on both sides of the aperture.

$$W_{131} = \frac{S_{II}}{2} = -A\bar{A} y \Delta \left(\frac{u}{n}\right)/2 \tag{13}$$

Field Curvature: For our system, due to the wide FOV requirements at short-range detection with a FOV of 45°, field curvature and distortion are most likely the most severe aberrations. Field

curvature is described by the Petzval sum, Figure 13 [13, 14], which is part of the corresponding Seidel coefficient shown in Equation 14.

$$W_{220} = \frac{S_{III} + S_{IV}}{4} = S_{III}/4 - H^2 c\Delta\left(\frac{1}{n}\right)/4 \tag{14}$$

Here, H the Lagrange invariant, which is calculated as:

$$H = n(\bar{h}u - h\bar{u}), \tag{15}$$

where $\bar{h}$ and $\bar{u}$ are the height and angle of the chief ray, respectively, and h and u are the height and angle of the marginal ray, respectively.

The first half of the Seidel coefficient for the field curvature is simply the coefficient for coma. We will disregard this part for our analysis of the field curvature since it will be minimized with the coma. In the second part, the Lagrange invariant does not change throughout the system, and the Petzval sum is only dependent upon the curvatures and refractive index of the lenses. Thus, the position of the elements in the system does not affect the field curvature; only the power of the lenses does. With this in mind, however, the same principle of power splitting can be applied to reduce field curvature. In addition, a field flattener lens can be introduced near the image plane to correct the field curvature. For our project budget, additional optics are not the best solution to aberration correction, although it is preferred over customizing lenses for this purpose. Nonetheless, budget considerations are made when designing the optical systems for both illumination and sensing. For the sensing optical system, specifically, a thin field flattener lens is a promising solution for small correction of the field curvature. Most of the correction, however, is done in the bulk of the optical system by power splitting.

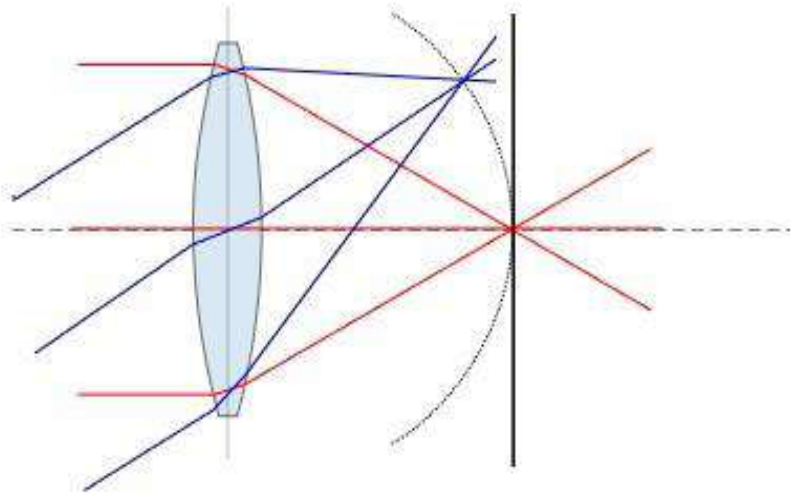


Figure 13: Field Curvature [13]

Distortion: In our system, distortion will likely be severe at wide FOVs. Although some methods exist for correcting distortion optically, like using Fresnel lenses and other aspheric surfaces, the

most efficient for our application is digital correction of the aberration through image processing. Since distortion is only a geometric mapping error of points on the sensor, the information on each pixel, depending on the image resolution, can be recovered and the points can be remapped to fit a rectangular grid free of distortion.

5.2.3.1 Illumination Optical Design

A peculiar characteristic of our LiDAR system is its varying illumination and sensing FOV. In order to accurately map objects at short as well as long ranges, the illumination power has to be distributed accordingly. To reach longer distances, the power needs to be focused more tightly, and thus a narrower FOV is required. Conversely, to scan objects closer to the system, the FOV needs to be wider. To achieve the FOV requirements for our illumination system, we designed our optical system for the purpose of beam expansion and de-collimation. Certain assumptions were made regarding laser beam divergence and beam diameter. Our optical design has some mechanical considerations, as some of the lenses need to be translated inside their housing. Details regarding the mechanical design of our project are discussed in section 5.2.4.

Our LiDAR illumination system requires the laser beam to be spread into a vertical fan with varying FOV from 10° to 45°. For a fixed angle, the simplest design employs a single concave cylindrical lens whose focal length can be determined using thin lens approximation as follows:

$$f = -\frac{d}{2tan(FOV/2)} \tag{16}$$

Here, d is the collimated laser beam diameter. Using this equation for the limits of our FOV, assuming d = 4 mm, we get $f = -22.86\ mm$ and $f = -4.828\ mm$ for our FOVs of 10° to 45°, respectively. This formula hints at one possible solution, which is using a zoom lens design to vary the focal length of the system.

Beginning with a simple design with two thin lenses separated by a distance t,we can use the effective focal length equation (Equation 17) to calculate the distance t required assuming certain focal lengths f_1 and f_2.

$$\frac{1}{EFL} = \frac{1}{f_1} + \frac{1}{f_2} - \frac{t}{f_1 f_2} \tag{17}$$

If we let $f_1 = f_2 = -10\ mm$, then the required distance between the lenses to obtain an EFL of 22.86 mm is -15.63 mm and to obtain an EFL of 4.828 mm is 0.7125 mm. The former is negative and not conceivable, and the latter is much too small; the curvature of the lenses would not allow for such a small separation. To obtain only positive separation values, the sum of the inverse of the focal lenses must be smaller than the inverse of the EFL. For a real solution, we can select $f_1 = f_2 = -50\ mm$ to obtain $t = 9.36\ mm$ and $t = 417.8\ mm$ for the 22.86 mm and 4.828 mm EFLs, respectively. However, notice that the separation to obtain the 4.828 mm EFL is incredibly long, which makes for an impractical approach. We would encounter a similar issue if we used a combination of two convex lenses, or a convex and a concave lens. This suggests that two lenses are not sufficient to create a practical zoom lens for our system. Using a similar approach an

additional convex lens is introduced into the system, and Equation 17 can be extended to two lenses as follows:

$$\frac{1}{EFL} = \frac{1}{f_1} + \frac{1}{f_2} + \frac{1}{f_3} - \frac{t_1}{f_1 f_2}\left(1 - \frac{t_2}{f_3}\right) - \frac{t_2}{f_3}\left(\frac{1}{f_1} + \frac{1}{f_2}\right) \quad (18)$$

Equation 18 shows that many degrees of freedom are added by introducing a third lens. Just looking at the equation a new term can be seen with the second lens separation, and the middle term is dependent on both separations. Given a set of lenses and their focal lengths, t_1 and t_2 can be determined by setting a restraint on their sum to keep the cavity length at a maximum specified value. This value can be defined considering total system size, as well as constraints for the translational mechanics needed to move the lenses. For our design, we will initially set $t_1 + t_2 = 60\ mm$. The ideal focal lengths of the lenses are somewhat more difficult to determine, but the basic principle used for the two-lens system of the sum of their inverses being slightly smaller than the inverse of the EFL still holds. This means that the sum of the last terms must remain negative for the separations to be positive.

Another thing to consider when selecting the initial focal lengths for each lens is that we do not want any rays focusing within or outside the system. Since we will be working with relatively high-power pulsed laser light, it is preferable not to focus the light before expanding it. Focusing the light inside lenses could damage them. Since for the zoom lens design there need to be moving optical elements, it is not prudent to have light focused between elements either as it could cross a lens while moving it to change the FOV. It is definitely not prudent to have the light focus outside of the system as it could be hazardous for the user. Therefore, another condition for the optical design is that the beam should be vertically spread without focusing.

This can be achieved with the simple two-lens design discussed above. The first lens can be moved closer or farther from the laser to bring the spread of the marginal rays at a higher or closer point on the second fixed lens which will cause the FOV to increase or decrease, respectively. Thus, moving the first lens closer to the laser increases the FOV, while moving it farther from the laser decreases the FOV without focusing the light. However, this design as seen previously, requires a cavity length that exceeds the limit for practical dimensions for the optical system.

A more appropriate design is therefore the three-lens zoom design. Using a fixed biconvex lens to bring the collimated marginal rays towards the optical axis followed by a movable concave cylindrical lens with high negative power to spread the beam, and finally a second concave cylindrical lens with low negative power fixed at the end of the system. This design option is more expensive than the one described previously, since it adds an extra lens, however, it allows for spreading of the light with reasonable cavity lengths and without focusing, as can be seen in Figure 15. To begin our design, we select a long focal length of 58 mm for the convex lens to ensure the light is not focused. The subsequent concave cylindrical lenses were given focal lengths of -12 mm and -40 mm, respectively. The first concave cylindrical lens has the highest negative power because this will be the lens that will be translated inside the cavity. Having more power, this lens can affect the EFL of the system more significantly. These focal lengths were selected after iteratively calculating t_1 and t_2 with several combinations of focal lengths to satisfy the cavity length constraint for an EFL of 22.86 mm. With these focal lengths, the power $K$of the lenses are

found as their inverse, and the radii of curvature R for each surface can be determined using Equation 18 below assuming N-BK7 glass of refractive index 1.5168.

$$K = \frac{1}{f} = R^{-1}(n-1) \tag{18}$$

Let the first lens be biconvex, and the second and third lens both be plano-concave. The surface radii for each lens are then as shown in Table 12.

Table 12: Illumination Optics Prescription

Lens	Bi-Convex		Cylindrical Plano-Concave 1		Cylindrical Plano-Concave 2	
Surface	1	2	3	4	5	6
Radius (mm)	60	-60	-6.2	Infinity	-20.67	Infinity

With this initial prescription, we used Zemax to simulate and optimize the optical design for our illumination system. Using the slide bar function to vary the curvatures of surfaces 3 and 5 and varying the position of the cylindrical plano-concave lens 1, we obtained the distances for achieving the desired 10° and 45° FOV. Figure 14 and Figure 15 show the optical layout for the 10° and 45° FOV, respectively, after optimization. The prescription for the optimized optical system is shown in Table 13. The next step will be selecting real lenses with similar properties from distributors like Edmund or Newport and adjusting the distances for each FOV value.

Table 13: Illumination Optics Prescription

Lens	Bi-Convex		Cylindrical Plano-Concave 1		Cylindrical Plano-Concave 2	
Surface	1	2	3	4	5	6
Radius (mm)	60	-60	-6	Infinity	-20	Infinity
Thickness (mm)	2		2		2	
Diameter (mm)	6		6.4		26	

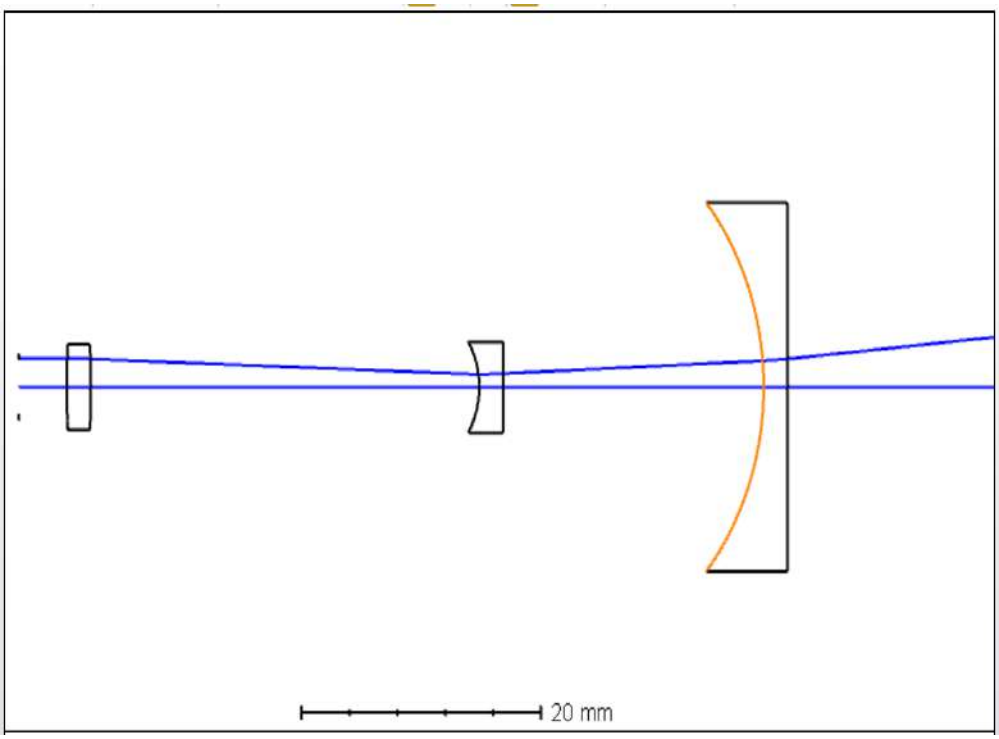


Figure 14: Illumination Optical Setup for 10° FOV

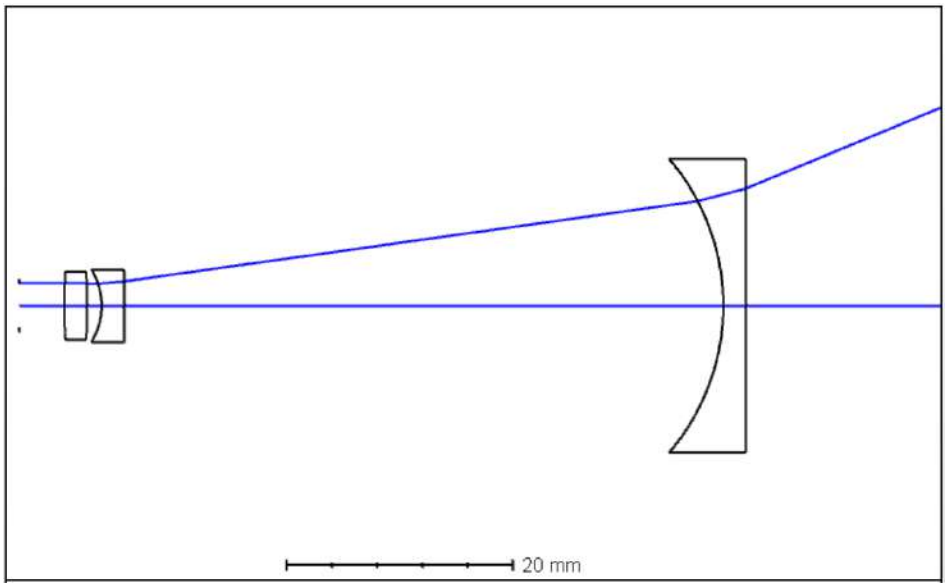


Figure 15: Illumination Optical Setup for 45° FOV

5.2.3.2 Detection Optical Design

For the detection system of the Mobile LiDAR Scanner, the group opted for the three-lens system. This decision came from section 3.4.1 Optical system. When modeling a three-lens system on Zemax we are fully capable of capturing the majority of light from FOV ranging from 0° - 45°. To calculate the total focal length of the optical system we used the following equation.

$$f_{total} = \frac{f_1 f_2 f_3}{f_1 f_2 + f_1 f_3 + f_2 f_3} \tag{19}$$

Using Zemax, which is a simulation software for optical design, we are able to insert the lenses that we plan to use and see how they will be incident on the sensor itself. This software helps us analyze the effect that the optics will have on the propagating light beam by looking at important parameters such as the RMS spot size and the field curvature that is modeled by the chief ray on Zemax. Figure 16 gives a layout of the current optical design that is being considered for the detection optics of the LiDAR Scanner. In Table 14, we present the prescription of the three lenses used in the collection optics. This Table 14 shows the focal lengths and radii of curvature of the lenses used for the Zemax simulation in Figure 16. Within Table 14, we show the thickness and diameter of the lenses.

From our current design, there is a high loss of information entering from the largest FOV angle. This issue is not of major concern because the rays of light that we are losing can be recaptured from other FOV angles. In Figure 16, the pink rays depict light coming from an FOV of 45°, some of these rays will not enter the collecting optics. This is a loss that we can afford because the information from those rays can be captured from the red rays of light which are incident on the sensor. Another consideration to be discussed is where the pink rays are being focused at. Currently, they lie in a curved imaging plane, this aberration was discussed in section 5.2.3.1 Aberration corrections and further analyzed below. This issue can be neglected as well since the majority of the information that we hope to capture with clear resolution is located at the smaller FOV angles. The final design problem we are trying to resolve is to lower the focusing point for the FOV of 45°. The focus point of these rays is just outside the active region of the CCD sensor. This issue will be worked on to optimize the system.

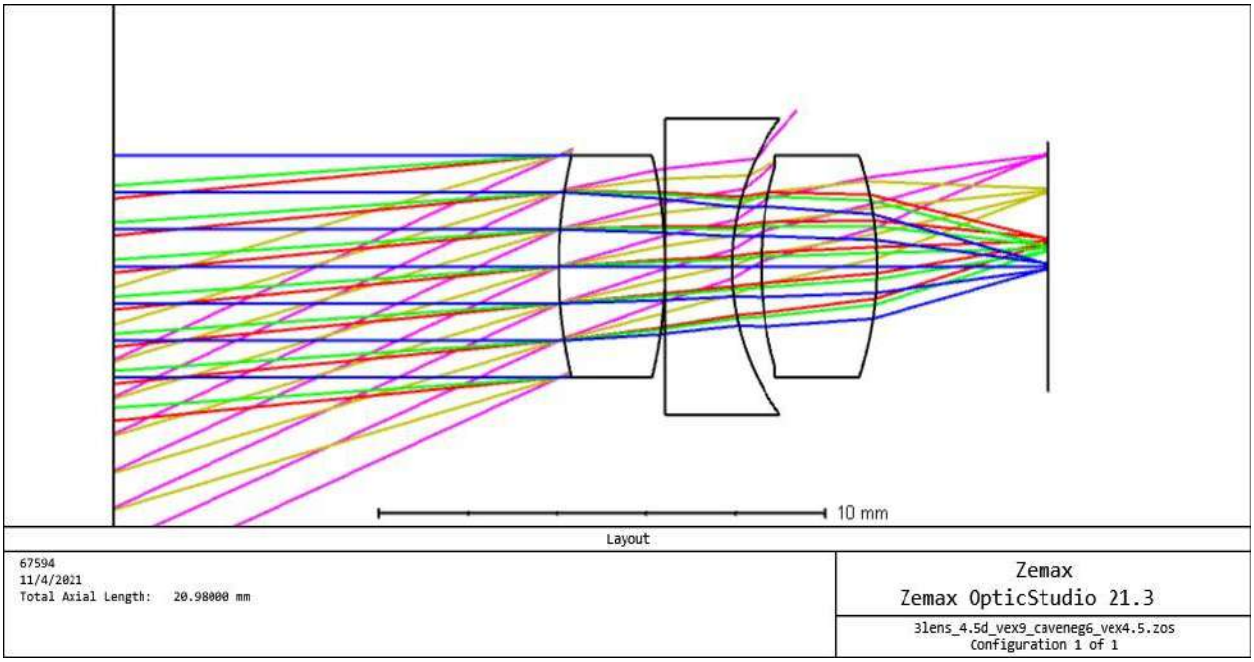


Figure 16: Detection Optics Layout

Table 14: Detection Optics Prescription

Lens	Bi-Convex		Cylindrical Plano-Concave 1		Cylindrical Plano-Concave 2	
Surface	1	2	3	4	5	6
Radius (mm)	8.88	-8.88	Infinity	4.71	6.58	-6.58
Thickness (mm)	2.38		1.50		2.60	
Diameter (mm)	4.5		6.0		4.5	

In optics when light passes through any form of aperture diffraction will occur. Since the aperture of our collecting optics is circular, the resulting diffraction pattern that will be observed on our imaging plane is an Airy disk, Figure 17 [15] . This diffraction pattern consists of a circular center with concentric rings around it that show the energy distribution of the propagating light. The formed Airy disk is a measure of the wavelength and NA of the optical system. Since our three lenses are circular lenses, this is a parameter that is of high importance to us. Having a large Airy disk focused on our sensor could result in a loss of information from our LiDAR. On Zemax the Airy disk is measured as the RMS spot size radius. In our current optical design, the RMS radii as seen from different angles that are described in Table 2, can be seen in Figure 18. Here the RMS radii values are measured in micrometers (μm) and the FOV angles (θ) are the half angles of the FOV as described in Table 2. From Figure 18, the focused light will be fully focused onto the active area of the sensor, meaning that we will be able to capture all the information that is being reflected from the LiDAR Scanner.

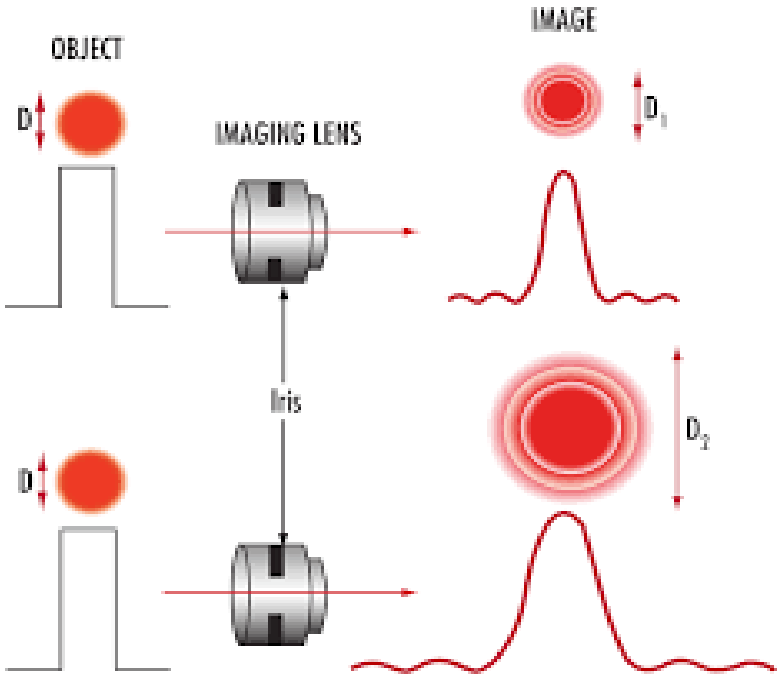
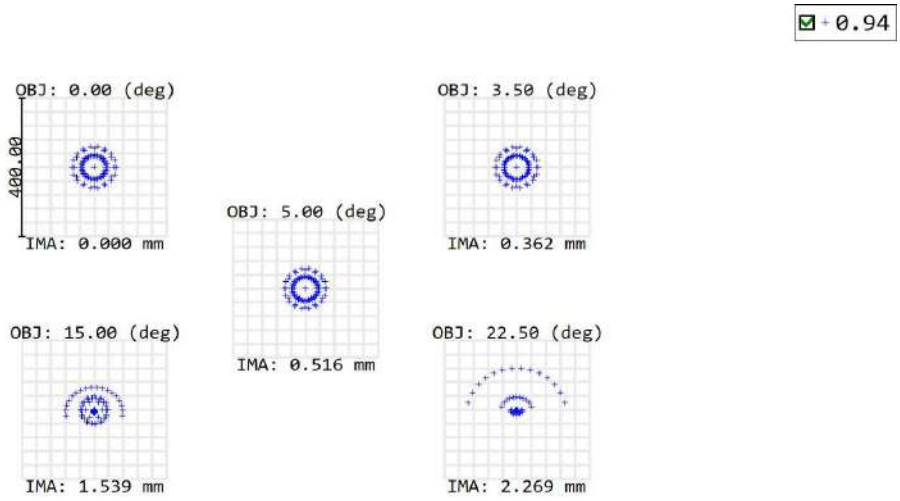


Figure 17: Airy Disk [15]

The second important parameter when designing the collecting optics is the field curvature of the collecting optics. Field curvature is an optical aberration that causes light to be focused on a curved imaging plane rather than on a flat imaging plane. Figure 19 visually describes field curvature and its effect on the imaging plane. This aberration is caused by the curved nature of the lens system. The mathematical description of field curvature is explained above in section 5.2.3.1 Aberrations. In our optical design, we can see a large field curvature for rays of light that are incoming from a FOV of 45°.

In order to meet the requirement specifications for the project, the focal lengths of the collecting optics are relatively very small, and this causes a large effect of field curvature. One method that is commonly used to correct field curvature is to use an iris that removes the rays of light that focus on the curved plane. For our project, this is not an option because it will prevent us from capturing the entire FOV outlined in Table 2. Figure 19 shows the current field curvature that is present in our optical system. From Figure 19, we can see that there is a maximum field curvature of 22.5°, this is the point where the incoming rays with a FOV of 45° are incident on the imaging plane. Another method that the group has devised to fix the current field curvature issue is to create a defocused imaging system. A defocused imaging system will decrease the resolution of our illumination system but will allow the rays of light from the various FOV to be incident on the sensor in the imaging plane at the same time. The other consideration to correct the field curvature of the collection optics is to introduce more lenses. Increasing the number of lenses of our system will give us more control of where the optical rays will focus at. This design will consist of a fisheye lens to collect the majority of light following by a zoom lens section similar to the one that is design in Figure 16, and finally a focusing lens to focus the light right onto the sensor.



Surface: IMA

Spot Diagram	
67594, 11/3/2021 Units are µm. Legend items refer to Wavelengths Field : 1 2 3 4 5 RMS radius : 43.712 43.381 43.101 44.298 71.968 GEO radius : 60.915 60.091 59.022 79.641 135.572 Scale bar : 400 Reference : Chief Ray	Zemax Zemax OpticStudio 21.3 3lens_4.5d_vex9_caveneg6_vex4.5.zos Configuration 1 of 1

Figure 18: RMS Spot Size

When designing the detecting optics another important consideration is the filter. As described above in section 3.3.1. As described in the referenced section the purpose of the optical bandpass filter is to prevent light that is not of the same wavelength from the illumination source from reaching the detector. Including a filter in our optical design is of high importance because this will allow us to reduce the signal to noise (S/N). By reducing the S/N, we are able to ensure that high peaks of light are reflections from the object rather than noise. This will improve the resolution of the rendered image. For our design, the filter can be located anywhere in the design shown in Figure 16. Since the distance between the lenses dictated the location of the focused light and the RMS spot size and field curvature parameters discussed above, locating the filter between the lenses is not a consideration for the group. This leaves us with two locations that the filter can be placed at to complete the optical design. Both of these locations have their potential benefits and drawbacks that will be discussed below.

The first considered location to have it at the entrance of the collecting optics. Looking at Figure 16, this would be on the left-hand side of the first lens as the light travels from left to right in that figure. The benefit of locating the filter at this position is that it would act as protection for the collection optics. This means that the filter would prevent dust particles or any other form of debris forming on the front mirror that can cause issues when collecting light. Another benefit of locating the filter in front of the collecting optics is that when designing and building the housing of the collection optics we can make the housing more uniform. This is because the size of the selected filter is much larger than that of the lenses. The filter that has been selected for our project has a diameter of 25.4 mm. The largest diameter of the selected lenses is the concave lens with a diameter of 6 mm. Since the diameter of the filter is much larger than that of the lenses, the NA of the system does not change, and the entrance pupil of the system is still the first lens on the left of Figure 16. This limitation is due to the sizes that are available for purchase. If we were to custom design either the lenses or the filter to have uniform size the budget constraint that was set by the group would be surpassed.

A drawback of placing the filter in front of the collecting optics is how much the intensity of the reflected light drops before entering the lenses. As mentioned in section 3.3.1 sensing and visually represented in Figure 6, 100 % of the light will not be transmitted through the filter. At our operating wavelength of 940 nm, ideally only 70 % of the incoming light will be transmitted through the filter. From our current optical design, we are losing light that comes in from the FOV of 45°.

The second place that is being considered for the filter to be located is in front of the sensor. On Figure 16, this would be all the way to the right just in front of the black line. The benefits of having the filter located in this position is that there is no loss of intensity before entering the collecting optics. This means that with sufficient power the majority of information will enter the collecting optics and be blocked off by the bandpass filter before being incident on the sensor. The main limitation of placing the filter in this location is that the housing for the optics would have to be modified to fit the filter. This would introduce a big constraint when designing and constructing the housing for the LiDAR due to the difference in diameter. This constraint and the one mentioned above about the housing for the LiDAR will be discussed in further detail below in section 5.2.4.1 Component Housing.

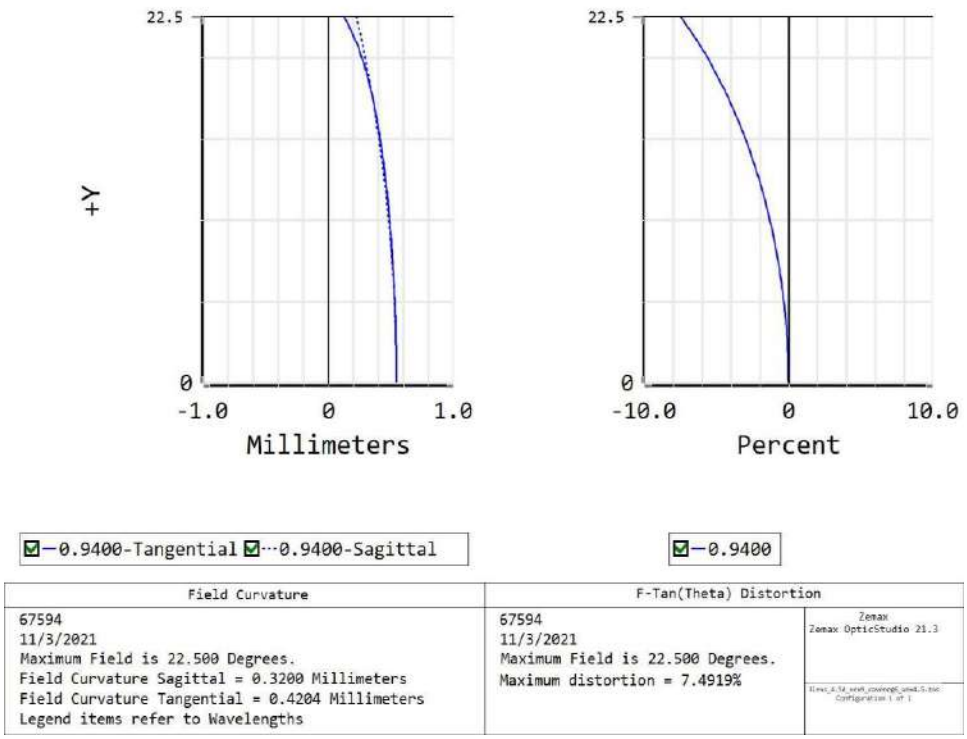


Figure 19: Field Curvature of Collecting Optics

This design limitation can only be decided once prototype testing is being done so that the group can outweigh the pros and cons of the location of the filter. The deciding factor for the filter location will be based upon power drop. The location in which the filter shows the least amount of intensity drop for the light incident on the sensor is the location that will be selected. If there is no significant difference in the intensity of the light incident on the sensor then the group will select the location where the component housing design is the most practical and easiest to execute.

5.2.4 Mechanical Design

For the mechanical design of the Mobile LiDAR Scanner, we can break it down to three major sections: component housing, mobility mechanics, and rotational mechanics. Design considerations will be presented in these three sections as they are going to be tackled in senior design II. Since the mechanical design of the Mobile LiDAR Scanner is dependent on having the optical designs optimized, the group has placed a higher level of importance in accomplishing these tasks prior to completing the mechanical design.

5.2.4.1 Component Housing

The mechanical design of the Mobile LiDAR Scanner will be designed to meet the requirement specifications of Table 2. In order to meet this constraint and requirement specification the group has decided to 3D print the housing for the optics and electronic components. The materials that can be used to 3D print the housings is a lighter option than building a metallic housing for these components. There is a tradeoff when choosing this method which is the structural integrity of housing. By selecting to 3D print the housings we are limiting the environment that our project can operate in. Climate limitations have to be taken into account. A 3D printed housing exposes

our components to weather such as rain. In comparison to a metallic housing that even under rainy conditions ensures that our optical and electronic components stay dry.

Another consideration that has to be taken into account when 3D printing the component housing is the precision of the printer itself. Since the optical design for both optical systems, the detection and illumination, will be precisely designed. The design is made such that the lenses are in fixed position for the most optimized system. Designing and printing a housing will require a very precise printer. Another constraint for the component design is that the housing needs to be constructed in a way for the lenses to be inserted to their designated location and at the precise distances from one another and from the sensor. A design feature that could be used to accomplish this is to create a lens holder with a lock and key mechanism. This can be easily designed on SolidWorks. Another design consideration when 3D printing the housing of the optics is to print each set distance individually and screw shut the entire system. This allows us for a more secure and compact housing that is less susceptible to external stress.

Another important constraint that the group needs to research is the different materials that are used for 3D printing. Different materials are used for different purposes, and this is research that the group plans on working on over the winter break. The final constraint that needs to be analyzed is the 3D printer itself. Depending on the type of printer that is available in the senior design lab is the design that will ultimately be selected. In order to select the final design for the component housing, the group needs to analyze how precise the 3D printer is and select what is the appropriate material that needs to be used to meet our weight constraint as well as be structurally hard enough to not fall apart. Additionally, since our team want to be able to work with the LiDAR system outside, the house will have to be weather-proof even waterproof to a certain degree. Of course, environmental conditions such as precipitation (rainfall, snow) and visibility conditions (fog or smog) will have to be considered. The embedded system is the most vulnerable to wet conditions as the excess moisture will cause a short circuit on the PCBs, rendering the LiDAR scanner useless.

The Mobile LiDAR Scanner will be housed in a 3-dimensional printed encasing. The group opted for this form of encasement as described above. The encasing will be printed in the CREOL senior design lab using a 1.75 mm PLA filament on the Original Prusa i3 machine, this printer is shown in Figure 20. The PLA filament or the Polylactic Acid filament is different from that of previously used filaments because it is made from renewable resources such as sugar cane or corn starch. PLA filaments have been shown to be more resistive to impacts and tensile strength [16]. Due to its high resistivity to impact, the group can be confident that this material will keep the optical components safe as the Mobile LiDAR Scanner transits through various terrains. For printing, the group will use the Original Prusa i3 machine. Since the Mobile LiDAR Scanner requires very high precision for the optical components, the group requires a machine that can print details in the millimeter regime. The Original Prusa i3 machine is capable to print details with 0.05 - 0.35 mm precision.

For printing the components for the Mobile LiDAR Scanner, the printing and assembly will be done by subsystems. Both optical systems will be printed separately from the main encasing and then combined in the final assembly. In the following section, we will describe how each system will be printed and then assembled. Figures 21-23, 25 are not drawn to scale since the finalization of the design on SolidWorks is still being completed.

The main encasing will be where all the electronic components will be housed in. This section will hold the microcontrollers that are needed for the Mobile LiDAR Scanner to operate, the EPC635 CCD sensor, and the battery to power all the internal components. When printing the main hosing for the Mobile LiDAR Scanner the group plans to print this component in two parts. This is because we need the interior of the main housing to be accessible for inserting all the internal components. Another feature that will be included into this design is a lock and key mechanism between the top half and bottom half of the main housing such that the internal components of the housing are accessible. This will allow the group to adjust or in the needed situation replace the internal components of the Mobile LiDAR Scanner. In order to attach the collection and illumination housing, there will be groves with a lock and key mechanism that will keep these two housing components connected to the main housing. Since each subsection of the component housing require access for either wiring or adjusting and aligning the optical components the design of the entire housing needs to be accessible into each individual section.

Figure 20: Original Prusa i3

This will also be the section that the servo motor is attached to for achieving the requirement specification of the 360° FOV, as outlined in Table 2. Figure 21 gives a visual representation of the main housing with dimensions. The dimensions for the main housing of the Mobile LiDAR Scanner are 12 cm × 15 cm × 15 cm. Within the main housing all of the electronic components will be placed. The microcontrollers, the rotational mechanics for the tunable FOV of the illumination optics, the CCD sensor (since it needs to be mounted on a PCB as described in section 6.7.4 60-pin breakout PCB design), and the Bluetooth module. Since the main housing will sit directly on the servo motor and the RC platform, the weight limitation for the main housing is governed by the wight limit of the servo motor and RC platform.

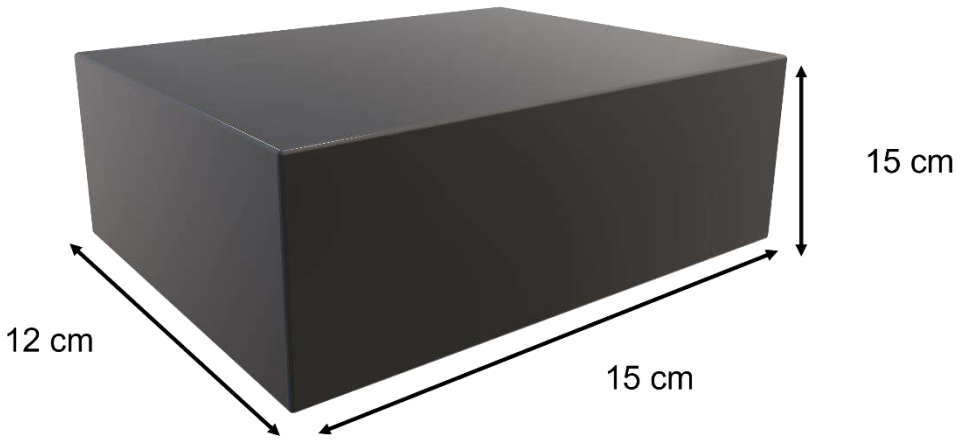


Figure 21: Main housing design

Collection Optics Housing: The collection optics housing will be a cylindrical tube that will be attached to the main housing with a lock and key mechanism as described above and aligned with two metallic or PLA rods to the main housing. The purpose of the metallic or PLA rods is such that all the components lie on the optical axis. The rods start at the CCD sensor and follow throughout the collection optics. The design for the rods in a PLA material can be seen in Figure 22. These rods are also present in the illumination optics and serve the same purpose. This can be seen in Figure 23 and Figure 25. The two prints will be done in the longitudinal direction, such that the optics can be aligned to the design specifications before they are closed off. There will also be some sort of lock and key mechanism that will allow the two longitudinal sections to be separated so the internal optical components may be accessed. Another important design configuration for the collecting optics is the location of the NIR bandpass filter. As mentioned in the previous section the collection optics housing will be printed in two parts. This filter is currently much larger in diameter than most of the lenses used for the collection optics, therefore there must be modifications made to the collection optics housing such that the NIR bandpass filter is capable of fitting into the subsystem. Figure # (collection optics housing) visually represents the collection optics housing with the relative dimensions that are not drawn to scale.

Figure 22 gives a visual demonstration of how the collection optics will be held into place within the collection optics housing design in Figure 23. In this design, we can see the supporting rods that ensure that the center of the optical components is aligned with the center of the CCD sensor. This is crucial for the design such that the simulated optical designs that are shown in Figure 16 hold true. In Figure 23 one can clearly see that the filter is placed behind the optical components right in front of the CCD sensor, by the bulge in the design.

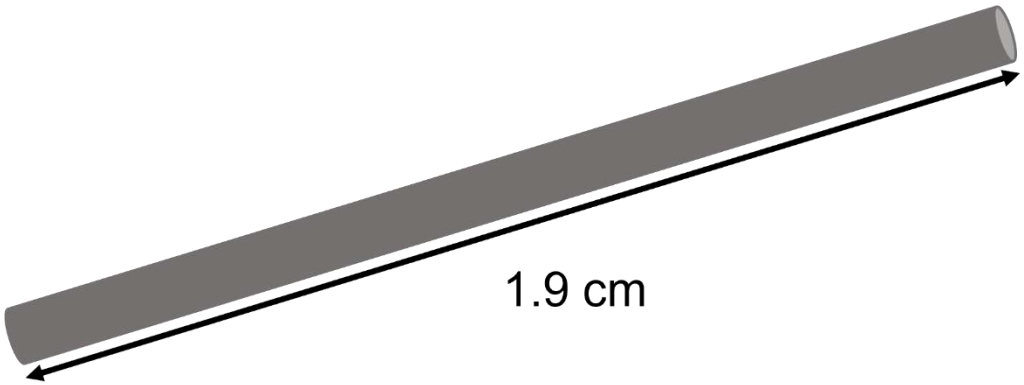


Figure 22: PLA rod design collection optics

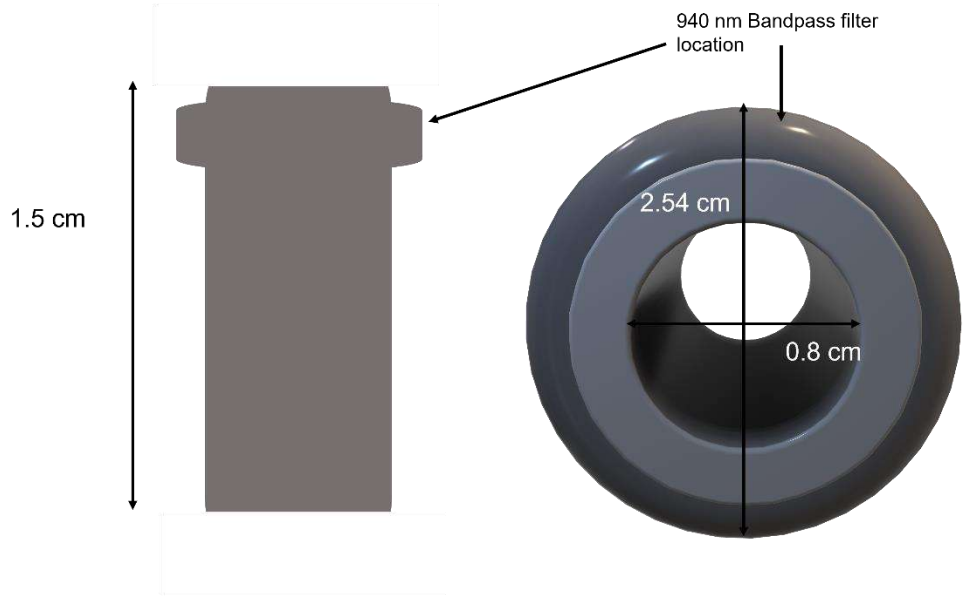


Figure 23: Collection optics housing design

From Figure 23 we can see the dimension of the current design for a 3-lens system is 1.5 cm long, 2.54 cm in diameter at the location of the NIR bandpass filter, and 0.8 cm in diameter at the location of the optical components. Since the effective focal length that needs to be achieved is 3 mm, as described in section 5.2.2 Detection optics, the housing will be very short. Although this may seem like a relatively short housing for the optical components, commercially available moderately inexpensive LiDAR scanners with similar operational distances as the Mobile LiDAR Scanner have similar dimension sizes.

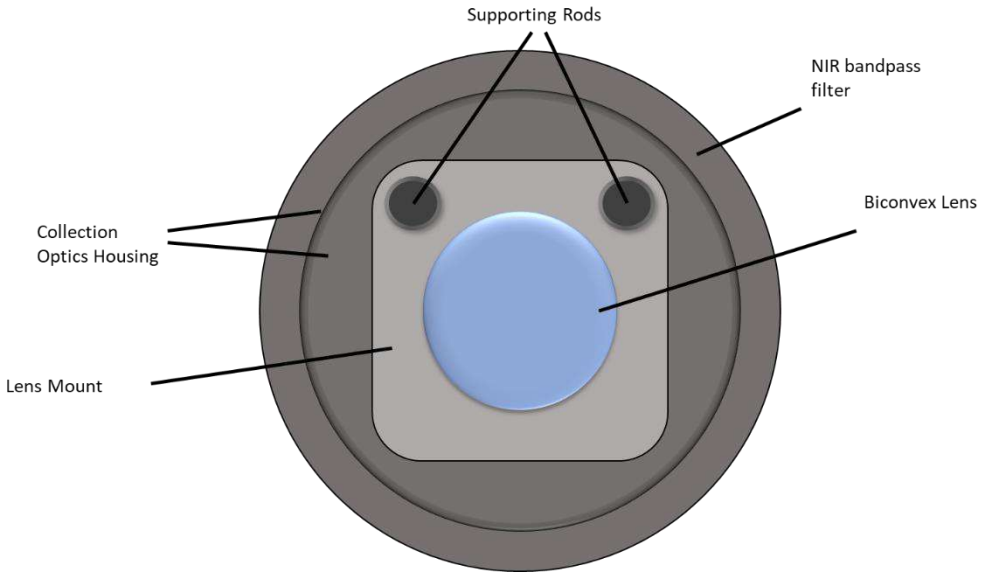


Figure 24: Collection optics mount

In this design, the group needs the NIR bandpass filter to be completely stationary, this is why we have designed the collection optics housing to have two different diameters. By only having one section of the collection optics housing to be the diameter of the NIR bandpass filter we eliminate any potential tilting of the filter due to external stress such as mobility. Since the NIR bandpass filter is much larger than the optical components of this subsystem, the NIR bandpass filter must be placed at the entrance of the collection optics housing such that the rods that keep the center of the lenses on the optical axis can be extended from the CCD sensor outwards. One consideration that can be assumed from having the NIR bandpass filter at the entrance of the collection optics housing is that the NIR bandpass filter will serve as a protection to the lenses from dust and other small contaminants.

Illumination Optics Housing: The group identifies the illumination housing to be the most complicated design that needs to be created. This is because the design will include a rotational motor and rotational rod within the illumination housing to move the illumination optics. Apart from this the optical components that are used for the illumination optics are cylindrical lenses rather than circular lenses. Therefore, when designing the illumination optics housing the group needs to consider the shape of the optics. Since the middle cylindrical lens will be translating throughout the illumination housing the group needs to consider designing sufficient space for this translation while still maintaining the optical alignment of the lenses. Another special consideration for this design is that the two cylindrical lenses are different sizes. This introduces another complication to design the housing in a way that the optical components lie on the optical axis. One design feature that is being considered by the group is to introduce the metallic or PLA rods that are used in the collecting optics. This will allow the group to have more control over the optical alignment for the illumination optics. Figure 25 gives a visual representation of the design for the illumination optics housing.

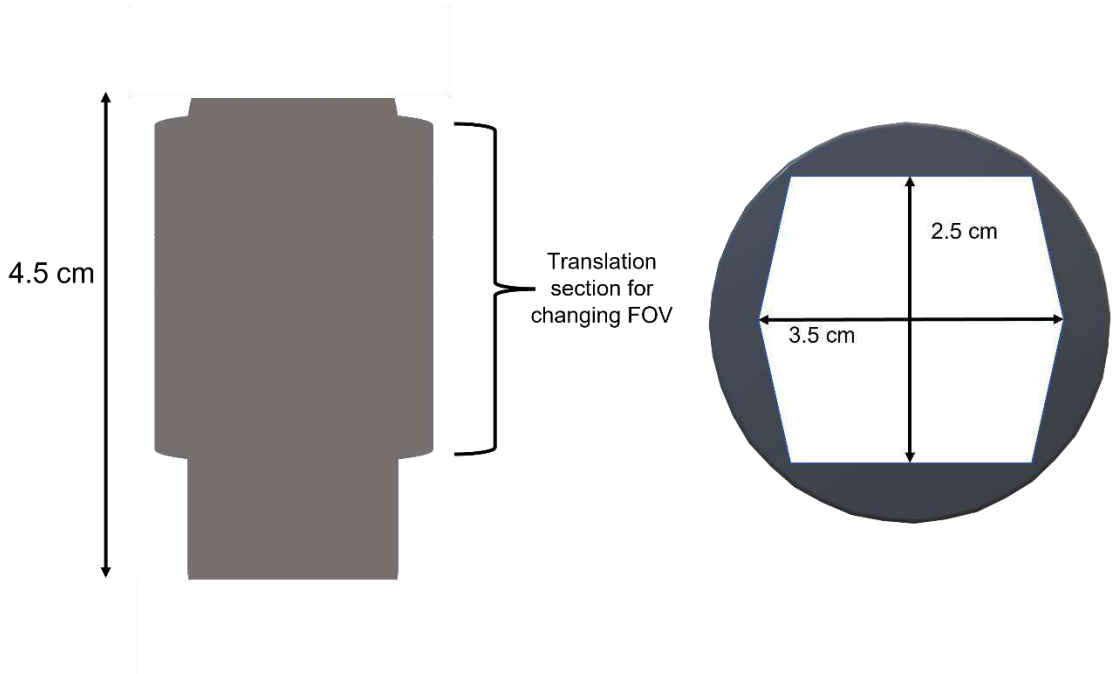


Figure 25: Illumination optics housing design

From Figure 25, the dimensions of the illumination housing tube are 4.5 cm in length, 3.5 cm wide and 2.5 cm tall. The illumination housing has different dimensions for the inside because the two cylindrical lenses are different in size, and since the middle lens will be movable for the tunable FOV, the group has decided that the simplest design would be to make the illumination optics housing the size of the largest lens.

5.2.4.2 Locomotion, Tilting, and Lens Adjusting Mechanics

Locomotion mechanism for Scanner transportation: For the locomotion mechanics, the group is purchasing an RC car and modifying it. Since one of the main features of our product is the mobility of the LiDAR Scanner, we will be attaching it on top of an RC car. To accomplish this, we will be removing the outer cover of the car and leaving the chassis of the car. This will allow us to mount the LiDAR onto the car itself and be able to be controlled by the user. In order for us to do this we need to meet the weight constraints of the LiDAR, which is discussed above. By designing and building the LiDAR separate from the RC car, gives us the flexibility as to what mode of transport the LiDAR can be mounted on. This enables the Mobile LiDAR Scanner to be placed on a boat or a drone as examples but are outside the scope of this project.

Tilting mechanism for optimization of illumination power delivery: Another mechanical design aspect that needs to be considered is how well can the Mobile LiDAR Scanner perform with its height off the ground. Since the Mobile LiDAR Scanner will be placed on top of an RC car, this can lead to a loss of relevant information due to the low height of the scanner with respect to the target objects. Relevant information is considered to be information that is at eyesight level of an average human. With prototype testing we will be able to discuss more about this mobility design and consider other approaches to achieve this. This, for example, can be solved by creating a stand that elevated the Mobile LiDAR Scanner such that the FOV of the system is more at eye level to capture relevant information.

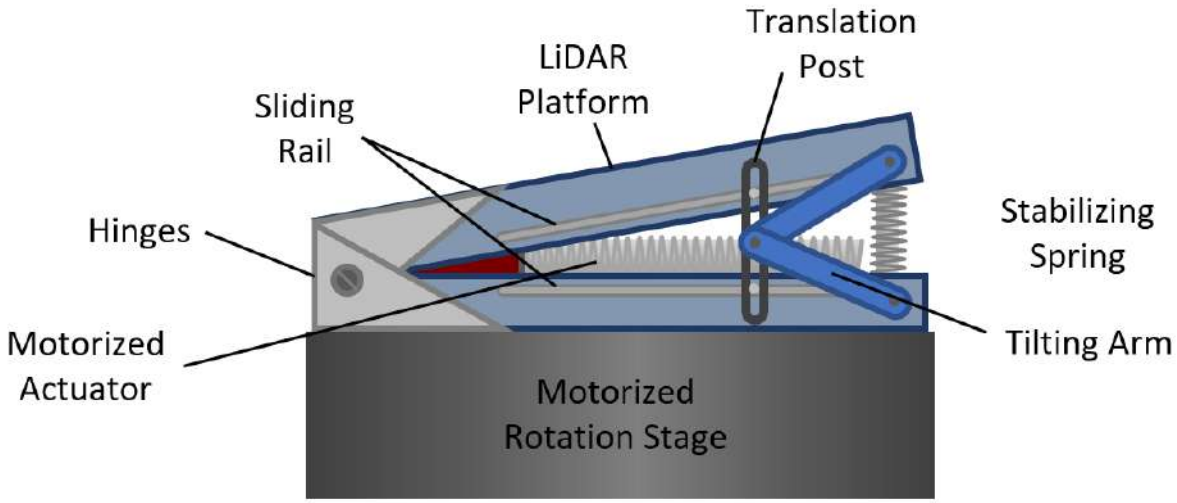


Figure 26: Possible Tilting Mechanism

Figure 26 shows a possible design for the tilting mechanism in which a motorized actuator is used to extend a tilting arm that increases the incline of the LiDAR platform. A spring is used to provide stability along with the tilting arm while the platform is suspended. As this mechanism needs to function during scanning, it is crucial that the platform remains stable while rotating the entire LiDAR system.

The threading in the motorized actuator needs to provide enough precision for 1º steps of the tilting. This will allow for precise tilting of the platform, tuning the illumination height to each detection range. The mechanism shown in Figure 26 has a non-linear relationship between the lateral translation of the actuator and the increase in the angular inclination. This makes it more complicated to find the minimum threading space for the required precision, as the rate of increase of angle of inclination will vary with the lateral position of the actuator. Figure 27 shows the trigonometric relationships of this possible design. The distance d is the position of the translation post shown in Figure 26. Notice that the path (in red in Figure 27) of the tilting arm is circular, and the junction of both tilting arms moves along the circumference of this path. The displacement in the x-direction of this junction is caused by the motorized actuator and its minimum displacement will be determined by the threading of the rod attached to it. Equation 20 shows how the translation post position d is related to the tilting angle θ.

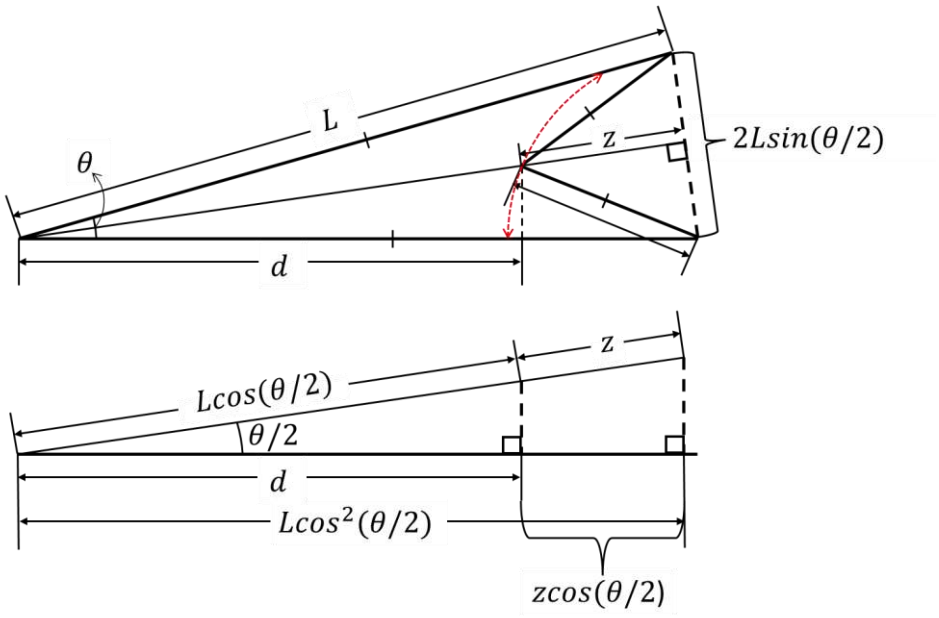


Figure 27: Trigonometric Relationships of Tilting Mechanism

$$d = Lcos^2(\theta/2) - \cos(\theta/2)\sqrt{l^2 - L^2\sin^2(\theta/2)} \qquad 0 \le d \le L - l \qquad (20)$$

Using a slightly different approach leads us to a more condensed equation (Equation 21), where the limits for the values of d and the corresponding limits for the inclination angle θ is more obvious.

$$d = L - l\cos\left\{\arcsin\left[\frac{\mathrm{L}}{l}\sin\left(\frac{\theta}{2}\right)\right] - \frac{\theta}{2}\right\} \qquad \begin{matrix} 0 \le d \le L - l \\ \text{or} \\ 0 \le \theta \le 2\arcsin(l/L) \end{matrix} \qquad (21)$$

To find the minimum lateral step needed for a tilting precision of 1º, we need to find the first partial derivative of d with respect to θ ($\delta d/\delta\theta$) and then solve for a discrete change in the angle $\Delta\theta = 1^0$. However, since the relationship between the lateral displacement and the angle of inclination is clearly non-linear, the derivative will be quite complex, as seen in Equation 22

$$\frac{\delta d}{\delta\theta} = \frac{L^2cos^2(\theta/2)\sin(\theta/2)}{2\sqrt{l^2 - L^2\sin^2(\theta/2)}} + \frac{1}{2}\sin(\theta/2)\sqrt{l^2 - L^2\sin^2(\theta/2)} - \frac{L}{2}\sin(\theta) \qquad (22)$$

Figure 28 shows a plot of the position d and its derivative with respect to the angle of inclination θ for a tilting 1-inch-long tilting arm and a 4-inch-long platform. Using $\Delta\theta = 1^0$ and finding where the second partial derivative with respect to θ is equal zero, we can get an equation for the minimum displacement of the actuator as a function of the platform length and tilting arm length. However, since the equation is periodic, it will also have a periodic derivative. Deriving Equation 22 is not necessary if we know not all values of d are valid, as seen in Equation 20 and Equation 21. Figure 29 shows that $\delta d/\delta\theta$ can be approximated as tangent curve for which a change in inclination angle of 1^0 can be obtained for a varying displacement depending on the inclination of

the tilting arm. The low values of the second derivative for low angles indicate that for small inclination angles small horizontal displacement of the tilting arm will cause large increases in the inclination angle. For example, for a tilting arm 1-inch long, tilting a 4-inch-long platform as the values used for the plots in Figure 28 and Figure 29, a small horizontal displacement of 200 µm of the translation post with a 0º inclination of the platform will bring the platform to an inclination angle of 4º. For small displacements, the relationship is fairly linear, thus suggesting that a minimum precision of 50 µm is needed to increase the inclination angle from 0º to 1º.

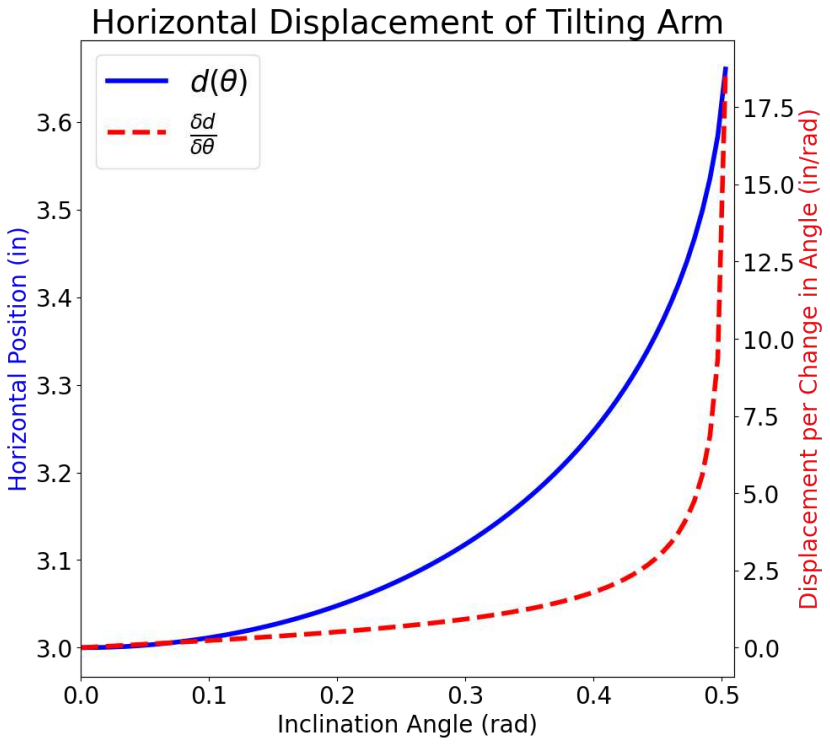


Figure 28: Sample Curves of Horizontal Displacement of Tilting Arm and its Partial Derivative with Respect to The Inclination Angle (arm length = 1 in, platform length = 4 in)

The stage should allow a maximum tilting angle large enough to direct most of the illumination power to nearby targets in front of the scanner. For short detection ranges, the illumination FOV can be as large as 45º, thus, to deliver most of that power to objects in front of the scanner, the beam should be tilted so that at least 50% of the illumination is above the horizontal. In other words, tilting the angle too high (>22.5º) will not allow for any scanning of objects at a height lower than the scanner. The minimum detection distance of our scanner is 1 m, so our beam should be directed in a way that light can hit objects on the ground 1 m away from the scanner. For a 25-cm-tall scanner, the maximum angle of inclination should be 8.4º, which also allows for over 68% of the light to be above the horizontal. To allow for a margin of error and more freedom of movement, the maximum tilting angle will be set to be 15º, to allow the user to detect objects higher off the ground if necessary. Nonetheless, the default settings of operation for the scanner will set the maximum tilting angle to be 8.4º when scanning objects 1 m away from the detector.

With the tilting arms fully extended, the maximum tilting angle can be calculated as the upper limit shown in Equation 21. It is not ideal however to have the tilting arms fully extended since it could cause damage to them. The arms will most likely be made of metal, but they will be supporting a substantial portion of the weight of the LiDAR, thus it is best to not place excessive strain on them. The tilting arm length l will be determined by Equation 23 as twice length required for a given platform length L to achieve a maximum tilting angle of 15º when fully extended. For a platform 4-inches long, the arm length is roughly 1 inch, thus the minimum horizontal displacement precision is still valid.

$$l = 2Lsin(\theta_{max}/2) \tag{23}$$

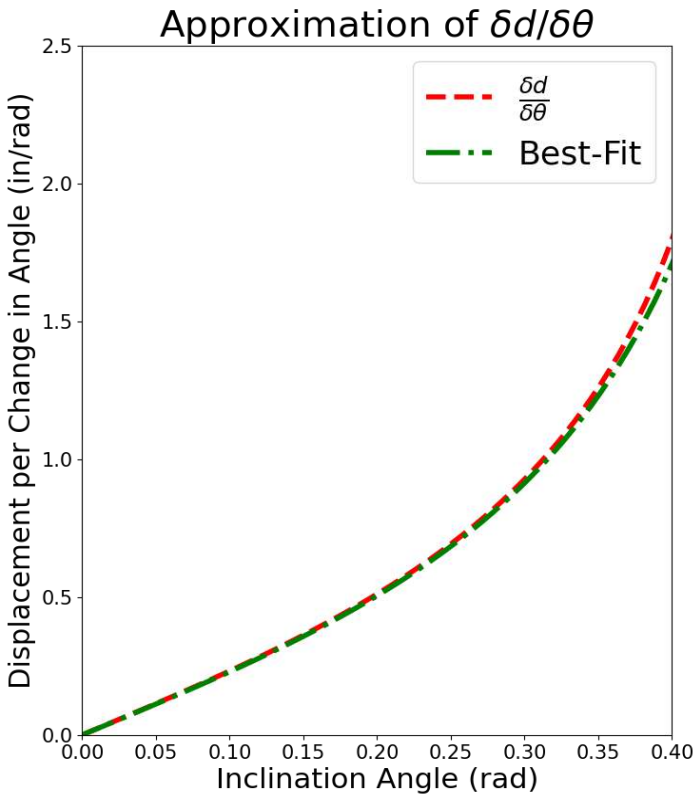


Figure 29: Best-Fit Curve with Tangent Model of Partial Derivative

After determining the platform and tilting arm lengths that satisfy the required maximum inclination angle, following the previous analysis for minimum horizontal displacement, we can now determine the threading pitch needed for the actuator to provide the required precision. Since we determined that a minimum horizontal displacement Δd of 50 µm is needed to achieve the required angular precision for the tilting mechanism, this means that our motorized actuator must provide such precision. Using a step motor with a minimum step rate α of 1.8º, Equation 24 tells us that the minimum threading pitch p required would be of 10 mm. However, this assumption relies on a perfect step precision from the motor, which is not realistic. With a 0.5 mm threading pitch we can achieve the required displacement precision with a rotation angle of 36º from the step motor, which should be well above the margin of error for the minimum angular step of 1.8º.

$$p = \frac{\Delta d * 360^0}{\alpha} \tag{24}$$

Lens translation mechanism for FOV tuning: A main feature of our scanner design is its tunable FOV for optimized power distribution at various detection distances. As previously discussed, for targets up close to the scanner (within 10 m) our illumination FOV must be wide enough to illuminate targets of considerable size; however, when detection range is increased, the illumination beam must be narrower in order to deliver more power to distant targets. This can be accomplished as discussed in Section 5.2.3.1 in the discussion of the illumination optics design, where a zoom lens design was constructed in order to achieve FOV variation from 10º to 45º (full angle). This design, however, depends on a mechanical system to translate the adjustable cylindrical lens along the optical axis within the lens encasing. The challenge of this design is to have the lenses protected within the encasing while still being able to move one of them within. To ensure the longevity of our system, the lenses should not be entirely exposed to the environment, this can cause dust deposition on the lenses, scratches, and other damages that would reduce the system performance. In addition, having the lenses completely enclosed provides better support against unwanted movement during operation. Since we are working with lenses of 3 mm diameters, the handling and housing become a challenge as well. We need to have encasings with small enough features to be able to restrain the fixed lenses and to allow for smooth translation for the adjustable lens. The encasing for the fixed lenses can be printed using a 3D printer. A polymer material, as the ones typically used in 3D printer, provide the necessary support and protection for the fixed lenses, and small features can be easily printed with the 3D printer. For the adjustable lens, however, the encasing cannot be solely printed. The polymer materials are not ideal for moving parts as any friction could easily damage the encasing. Metal, on the other hand, is smoother and allows for implementations were adjacent parts slide on each other. However, metal is also more difficult to manipulate.

The manufacturing expenses to obtain the metal pieces with the features required for our design are over our budget; therefore, an alternative approach is needed. Using thin metal rods (1/8 in diameter) as railing system provides additional support and a mechanism for sliding the movable mount along the optical axis. Figure 30 below shows a layout of a possible design implementing the metal rod support mechanism and a motorized actuator composed of a 12 V servo motor with shaft attached to a threaded rod. The motor rotates the threaded rod and the movable mount slides across the optical axis, effectively changing the FOV of the illumination system. To accomplish this, the movable mount will have a metal nut with the same threading as the rod attached to the motor, as seen in Figure 32, so the mount will be moved when the motor spins. The supporting rods above will ensure the lens is translated perpendicularly to the optical axis and provide stability to the system. As seen in Figure 31 the fixed lens mounts will not have the threaded nuts, so they will not move as the motor spins, instead, they will have bearings to assist the rotation of the threaded rods.

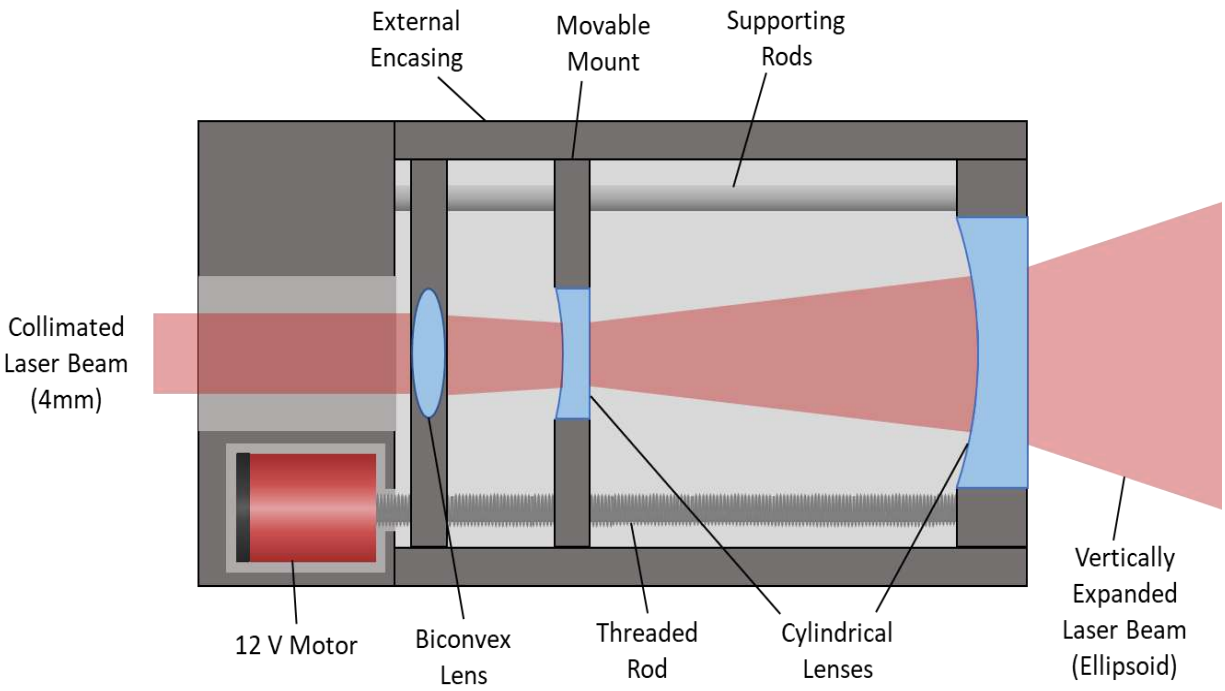


Figure 30: Profile View of FOV Tuning Mechanism.

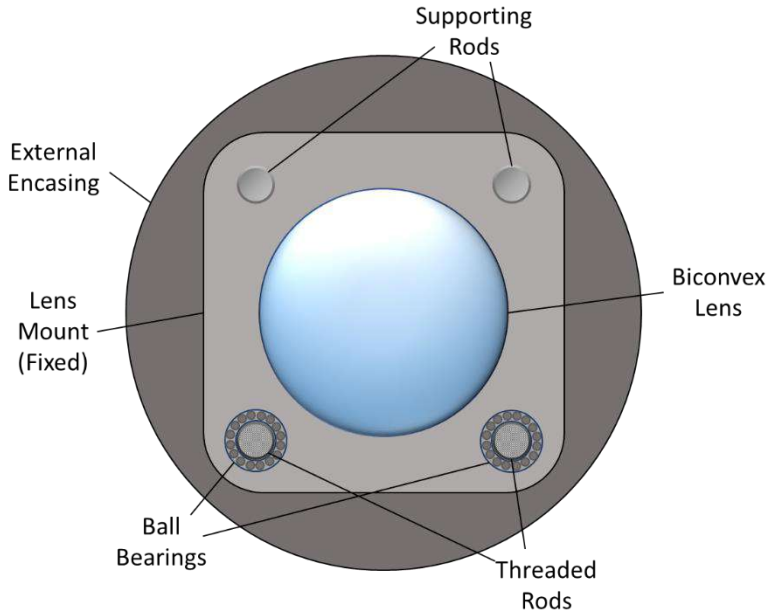


Figure 31: Fixed Lens Mount in Encasing

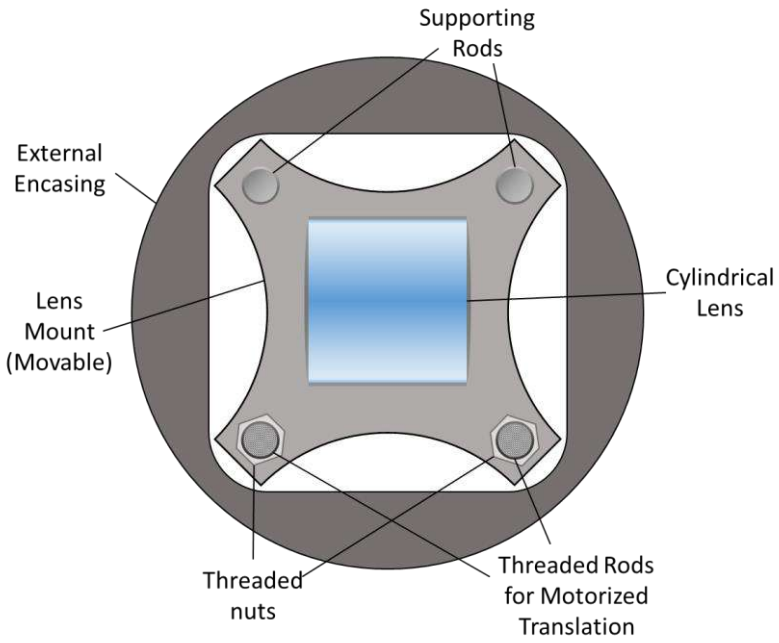


Figure 32: Movable Lens Mount For FOV-Adjusting Cylindrical Lens

5.2.4.3 Rotational Mechanics

The group is defining rotational mechanics as the mechanical design that will rotate the LiDAR Scanner to achieve a 360° scan of the area. In this section we will discuss the design that will be used as well as the scanning speed of the LiDAR. Since we are purchasing a rotation motor that will be powered by the system, there are not many design considerations that are needed other than for it to be able to rotate the whole system. One considered design for the rotational mechanics is to purchase the rotation motor and create a rotation stage ourselves. The rotation stage will be designed to fit the LiDAR and the electronic components to dimension. In order to mount our own built stage, we must consider the rotation speed of the motor, the torque that can be generated from the motor, and the power that needs to be delivered to the motor.

The rotational speed for the LiDAR Scanner will be based on the speed that we collect information from the area. This speed needs to be slower than the detection speed to ensure that we are not losing information before it is even detected. The main constraint for the rotational mechanics is that the rotation stage cannot rotate faster than the speed at which the light travels from the source to the sensor. If the rotation stage is rotating at a higher frequency than the calculated frequency of the modulated laser pulse, then the LiDAR will not be able to collect the reflected light. The laser maximum frequency that the laser can be modulated to scan at a maximum distance of 50 m is 3 MHz. This frequency is calculated in section 3.3.2 Illumination under the section describing the modulation. Figure 33 gives a visual representation of the integration of the servo motor into the Mobile LiDAR Scanner design. Figure 33 shows how the main housing will be mounted onto a rotation stage that will allow the Mobile LiDAR Scanner to achieve the requirement specification of the 360° FOV, which has been outlined in Table 2.

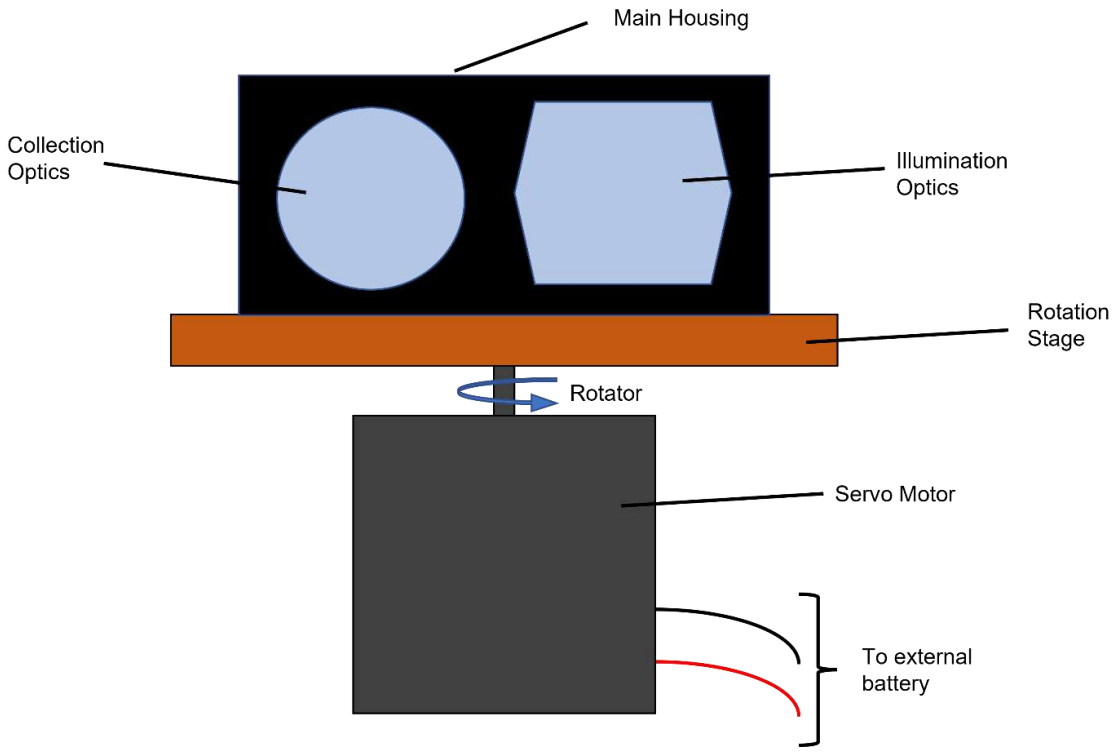


Figure 33. Servo motor & rotation stage design

The CCD sensor used for ToF measurements has a minimum shutter time of 250 ns, which for the maximum laser modulation frequency of 40 MHz it offers means that the shortest illumination pulse of 25 ns can be easily sampled within one shutter opening. For longer illumination pulses, the shutter speed will of course need to be decreased, which indicates that our scanning speed will vary with the detection distance defined by the user. We can define an equation for the scanning speed of our system based on modulation frequency. To accurately sample our backscattered pulse, the shutter opening time must be at least twice the width of the pulse. For example, a pulse of 333.33 ns corresponds to the maximum modulation frequency f_{LED} of 3 MHz allowed for an unambiguity distance of 50 m for our maximum detection range, thus when scanning at 50 m (r), our shutter opening time $t_{shutter}$ should be at least 667 ns (see Equation 25). This means that our scanner needs to stay relatively still for that time.

$$t_{shutter} = \frac{2}{f_{LED}} = \frac{4r}{c} \tag{25}$$

To determine the maximum allowed transverse displacement allowed for our scanner, we need to consider the optical magnification of our detection system. Since we are implementing a variable FOV system, our magnification will change with the detection distance as well. For detecting objects at the maximum range (r) of 50 m our angular FOV is 5º which gives a magnification (M) of 0.73 for our sensor of 3.2 mm in height (H) (see Equation 26). Our transverse displacement will shift the horizontal position of our sensor, so we need to consider the ratio of pixel to object size. The pixel size of the sensor is 20 µm, thus each pixel will take 27.288 µm of the object space.

$$M = \frac{H}{2 * r * \tan\left(\frac{FOV}{2}\right)} \tag{26}$$

Ideally, our sensor would not be displaced by more than a pixel on the object space during scanning, to prevent significant jitter and smearing of the image. This means that the linear speed of the displacement needs to be lower than 27.288 µm / 667 ns = 41.8 m/s, which translate to a rotational speed of 0.836 rad/s or 7.98 rpm. This is a relatively low speed; for averaging the data to create a detailed 3D map of the scanned area at the maximum detection range, it would take a couple of minutes. Equation 27 condenses all the work to calculate the maximum scanning speed v_{scan_max} for a given detection distance, where r is the detection range, s is the pixel size (20 µm), and M is the magnification obtained from Equation 26.

$$v_{scan\_max} = \frac{Pixel\ size\ at\ object}{r * t_{shutter}} = \frac{15}{\pi} \cdot \frac{c}{2r^2} \cdot \frac{s}{M} \quad [rpm] \tag{27}$$

5.2.5 Embedded System

An embedded system is basically a minicomputer. It is a micro processing hardware system with software that's designed to perform a dedicated task that's either part of a larger system or as an independent system. At the core of an embedded system, the microcontroller is used to carry out real-time computations. There are many different types of microcontrollers, and they could be found in just about every machine that's used in today's society. Due to the variety of different microcontrollers, to be able to gear it more to our project, and the microcontroller that we were most familiar with on the team, we decided to use the MSP430FR6989. Although using the ATmega328 microcontroller for the Bluetooth module connection would be easier. Diverting the pins and programming in the C language were the key factors in producing a working device for our project.

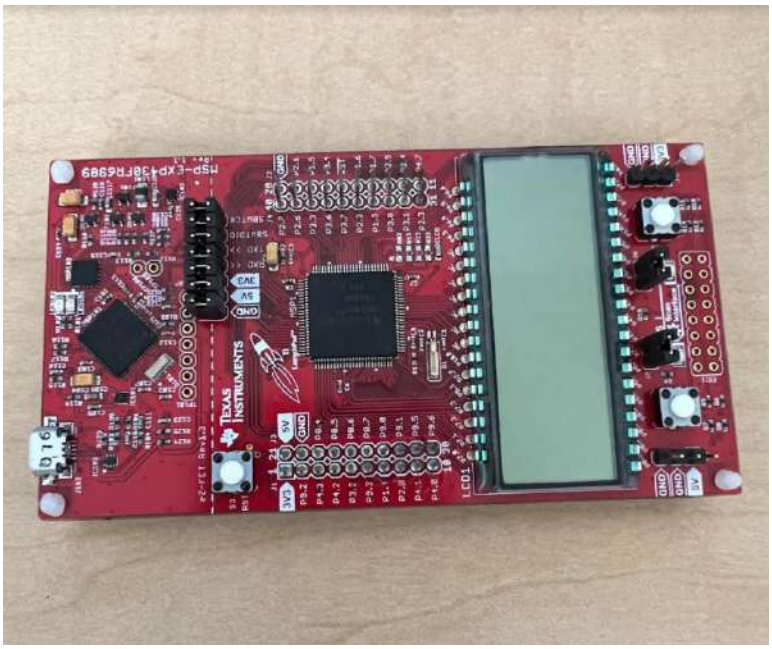


Figure 34: MSP430FR6989

5.2.5.1 Microcontroller

The microcontroller is the heart of any embedded system. Choosing a microcontroller is a very critical decision. It acts as the main communication between the different peripherals of the system used throughout the system and the calculation of data points from the LiDAR scanner itself.

The most crucial aspect of the microcontroller that needs to be considered is the baud rate offered by the microcontroller. This is important for the microcontroller to communicate back to the PC and to handle the influx of LiDAR data given by the scanner. Data collected from the LiDAR system is captured at a very high speed, and the microcontroller will have to be able to capture the data from the detection system, convert it to an encoded point, and transmit it back to the PC to be rendered in a 3D point cloud.

UART (Universal Asynchronous Receiver/Transmitter) is one of the most popular hardware communication protocols for device-to-device communication. This protocol was decided to be used for this project. UART uses a full-duplex transmission mode, having two signals (transmitter and receiver) to communicate between devices. UART typically transmits data as frames as bytes of data. Both distance and direction will need to calculate the data points. Calculated data points from the scanner can also be encoded as ASCII text. Characters like a comma can be used to delimit different data points during transmission.

5.2.5.2 Bluetooth

To handle communications between the scanner and the computer, our group knew that the communication would have to be wireless to account for the mobility of the scanner. A wired connection in this case would severely reduce the range of mobility of the scanner, essentially tethering it to the computer used to render the 3D scans. Our group first considered using a Wi-Fi module to interface the vehicle with the computer. However, this would require an Internet connection to present to operate the vehicle. Finally, the team opted for a Bluetooth connection, using an HC-05 Bluetooth module.

Bluetooth refers to a wireless, short-range technology standard used for data transmission between a mobile device and a fixed device using low-power high-frequency radio waves. These radio waves can fall between 2.400 GHz and 2.480 GHz [17]. Two devices must be constantly connected in order to transmit and receive data over Bluetooth. Currently, there are two types of Bluetooth technologies: Bluetooth Low Energy (LE) and Bluetooth Classic (known formally as a Bluetooth Basic Rate/Enhanced Data Rate or BR/EDR). As the name suggests, Bluetooth LE is a more popular option because it requires much less energy to operate but the Bluetooth Classic offers a slightly better data rate. Bluetooth Classic offers data rates of 3 Mbps compared to the Bluetooth LE's 1 - 2 Mbps [18]. In the context of our project, we are focusing on LiDAR data transmission and a high data rate is coveted.

Within our project, Bluetooth communication will be used for transmitting LiDAR data from the scanner to the computer and sending commands to the LiDAR system using the UART communication protocol. This allows the LiDAR scanner to move about freely and takes away the heavy calculations from the scanner itself. To be able to establish this communication in development, the team decided to use an HC-05 Bluetooth module from DSD Tech attached to our microcontroller. This module is shown in the figure below:

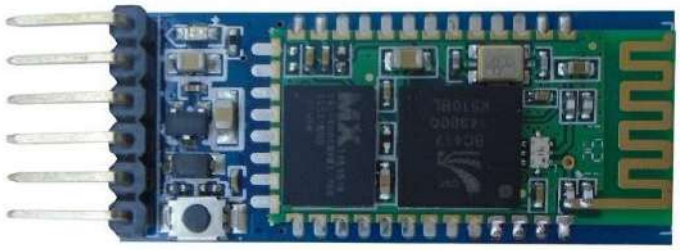


Figure 35: DSD Tech HC -05 Bluetooth module

We chose this module because it was cheap and readily available on Amazon. As the LiDAR data is measured by the LiDAR system, the data will be streamed back to the computer. This raw data will be collected by some sort of driver. The raw data will be encoded into a LAS file and read by our program on the computer. Additionally, the computer will be able to send commands to the LiDAR system, telling the scanner to start or stop a scan.

Our team decided to use a Bluetooth connection rather than a Wi-Fi connection to connect to our microcontroller because we did not want our scanner to be restricted because it needs an internet connection to be able to transfer the data properly. The mobile LiDAR scanner will be used for both reconnaissance and exploration, like out on missions or enclosed areas for archaeological purposes. An Internet connection is not always readily available in these areas and the scanner cannot connect to the rendering computer, rendering the LiDAR scanner useless.

5.2.5.3 CCD Sensor Pin Layout

For the selected CCD sensor for the Mobile LiDAR Scanner, the pin layout can be seen in Figure A.3.1 [6] in Appendix 3. Although the purchased sensor has a total of 44 pins, not all of them are required for our project. The required pins for the Mobile LiDAR Scanner are 1,2,4-9,10-13,18,19,29. Pins 1 and 2 are the modulator and clock of the CCD sensor. Since the CCD sensor comes with a built-in modulator and TOF clock, these two pins will be connected to the laser diode to modulate the signal as well as read the time it takes for the pulses to return. This will help the group in the TOF calculations by allowing us to have a baseline measurement of when the pulse leaves and when it returns. Pins 4,5,7,8,10-13 are the data pins, these are the pins that will transfer the converted optical signal to an electrical system and send it to the Bluetooth module for data transmission. Pin 6 is the power pin that supplies the digital voltage. Pin 9 is the power pin to the IO supply. Pins 18 is the pin that controls the I^2C clock input. Pin 19 is the pin that controls the I^2C data input and output. Since the CCD sensor is configured for I^2C, we need to use this pin to communicate the other electronics to the sensor. Pin 25 is the ground pin for the modulated laser diode. Pin 26 is the modulator for the laser diode, this is what will supply the modulated current or voltage that will modulate and pulse the laser diode. Pin 29 is the pin that regulates the shutter speed of the CCD sensor. This will allow the modulation of the optical pulses and collecting the

needed information. The two remaining pins, 37 and 39 are the ground pins for pin 9 and 6 respectively.

In order to integrate the CCD sensor into the Mobile LiDAR Scanner, the group bought a CCD sensor that comes pre-mounted on a PCB board carrier. In this configuration, all of the CCD sensor pins are functional, but the group will only use the ones listed in Table 15 below. The remaining pins will be grounded. The group also purchased a connector that the group will use to attach the CCD sensor onto the main PCB board design. One added benefit of this is that the CCD sensor will be orientated orthogonal to the main PCB and parallel to the illumination source. Table 16 will give an explanation for the abbreviations used in Table 15.

Table 15: Pin list for EPC635 CCD sensor

Pin #	Pin Name	Supply Class	Pin Type	Description
1	DCLK	V_{DDIO}	DIO	TCMI data clock output
2	MODCLK	V_{DDIO}	DI	Modulator/ demodulator for external clock input
4	DATA0	V_{DDIO}	DIO	TCMI high-speed output bit 0
5	DATA1	V_{DDIO}	DIO	TCMI high-speed output bit 1
7	DATA2	V_{DDIO}	DIO	TCMI high-speed output bit 2
6	VDD	V_{DD}	PWR	Digital supply VDD
8	DATA3	V_{DDIO}	DIO	TCMI high-speed output bit 3
9	VDDIO2	V_{DDIO}	PWR	IO supply VDDIO
10	DATA4	V_{DDIO}	DIO	TCMI high-speed output bit 4
11	DATA5	V_{DDIO}	DIO	TCMI high-speed output bit 5
12	DATA6	V_{DDIO}	DIO	TCMI high-speed output bit 6
13	DATA7	V_{DDIO}	DIO	TCMI high-speed output bit 7
18	SCL	V_{DDIO}	DIOD	I^2C clock input
19	SDA	V_{DDIO}	DIOD	I^2C data input/output
25	VSSLED	V_{DDLED}	GND	Ground pin for laser modulator
26	LED	V_{DDLED}	AOD	LED driver open drain output
27	LED2	V_{DDIO}	DO	LED driver push-pull output

Table 16: Pin Table Key

Pin Type	Meaning
DI	Digital Input
DO	Digital Output
DIO	Digital input/output (bidirectional)
DIOD	Digital input/output (bidirectional), open drain
AOD	Analog Output, open drain
GND	Ground

5.2.6 RC Platform

5.2.6.1 RC Platform 1

To be able to make the LiDAR machine move as well as the microcontrollers that are going to be used to control it, we need to create a platform that the LiDAR scanner will sit on top of. After trial and error of looking for multiple ways to implement a way to make the scanner travel, the group decided to use a Remote-Control Platform. The Remote-Control platform is exactly what it sounds like, we intend to use a NodeMCU with two high torque DC motors to make the whole device move. The two motors would be connected to a continuous track that are similar to what an army tank uses to travel through different terrain. The platform we will use would vary depending on the size of all the components we have after assembled. Discussing the dimensions of all the components we plan on using in previous sections listed above, the platform will be directly proportional to it.

The DC motor associated with the remote-controlled device would be required to have a torque that's more than the weight associated with the fully completed piece. The motors we would be using are the high torque DC motors that come in the Arduino kit associated with this product. Although the motors we will be using may have a hard time when the LiDAR scanner is assembled on the platform, the motors could be easily replaced with higher torque motors that would make the device move through the given terrain more smoothly. Depending on the number of motors we use for this design, we would add to the price and document it in our design process. The motor would be connected to the NodeMCU, this is what the remote would be communicating to for the user to perform any action they desire. The way in which it would be controlled would be on our smartphone device. By downloading the app created for the NodeMCU, this would make it easier to use the same device to send signals to the MSP430FR6989 for startup and use the same device for controlling the remote-controlled car.

The platform we intend to use would be closely related to the LiDAR scanner. Knowing the LiDAR scanner is the heaviest piece, the team would need to devise a plan to have a platform for

the LiDAR scanner to sit on top of, then be able to perform at the rate we expect it to. The platform would either be made up of a thin piece of wood that we would cut for the accurate dimensions, or we may use the back plastic piece that was also in the Arduino kit associated with the remote-controlled car. As we begin to develop our device, the plan for constructing our build will be clearer. Figure 36 [19] is a visual representation of the described RC platform.

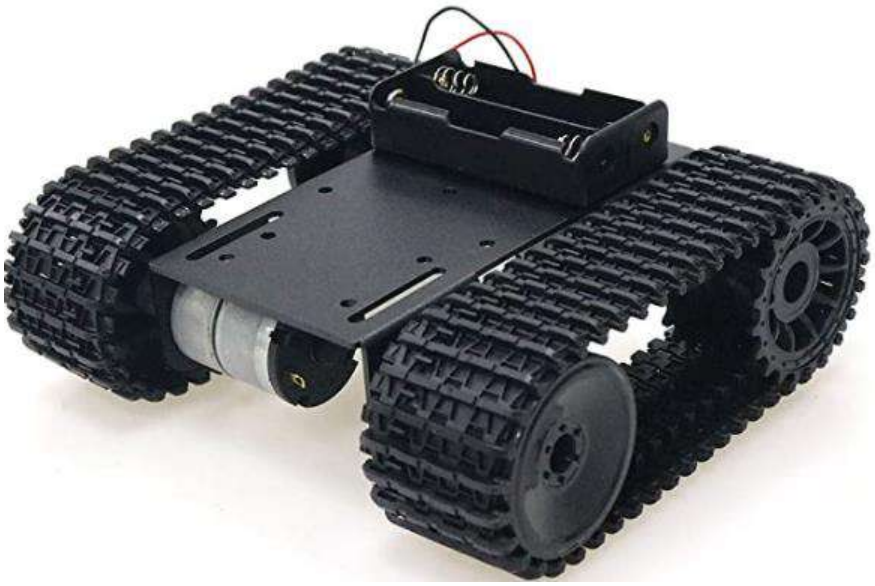


Figure 36: RC Platform 1 [19]

5.2.6.2 RC Platform 2

To further expand on the development of the platform we intend to use for transportation. We also looked into possible replacements if our original design for the RC car failed. The first design focused more on the group obtaining a cheap yet fully functional device that would be able to transport the LiDAR system as well as the microcontrollers associating with them. The weight we plan on delivering for the total device was to be less than twelve pounds. The weight that would be on top of the platform, the groups intended target weight is at most six pounds. Although the LiDAR system we plan on using is sought to be lightweight, we also have to make sure the dimensions/specs and the load are directly proportional to the platform we use. RC Platform 1 was a design the team took a look at to be cheaper, the price was about $30. This would be beneficial to the final price on implementation. RC Platform 2 is more expensive, the listed price from the seller was listed at $82. The difference in price would increase the total budget of our LiDAR scanning system by an amount of $52. Despite the fact that the total price is shown to be a significant increase, the performance of our system would perform better. The LiDAR system and microcontrollers associated would have more room for placement, and the movement that would be dependent on the motors could prove to be more efficient.

The motors associated with it are 2WD motors that are attached to sprocket type wheels. The sprocket wheels help it to have a full 360 degree turn while staying in place. This was looked at in the development on how we plan on capturing the full 360-degree view from the camera. We either can decide on using the tank to make the full 360 degree turn or we can plan on using servo motors to make the complete spin on the LiDAR system separately. This design would come more to fruition as we develop the product. The servo motors would be an additional price added to the total price of our system but can be a positive addition. The sprockets have a significantly slower spin speed than servo motors, also to take into consideration is the stability when turning. Again, these are all theories we have to accept, careful testing through all of our products would come as we get closer to developing the full device.

In the relevant technologies section, we make comments about similar devices other groups and/or companies made that replicate our design. Many of the designs include particular aspects we've considered in developing ours. The XiaoR Geek RC car we plan to use is compatible with many controllers on the market. These controllers include Raspberry Pi, UNO R3 and even Jetson Nano. The controller will be the key in making our device move. It would deal with swiftly maneuvering the device in a set location to expel scans. Figure 37 [20] gives a visual representation of the described RC platform.

Figure 37: RC platform 2 [20]

5.2.7 Power

The distribution of power is a key role in making sure our device performs at expected levels. Understanding the rates of how many voltages and the amount of current being transferred through each line would result in a working system. Too much voltage or current going through the wire to an element could result in overheating or burning components. After creating a final PCB design, this would be crucial, destroying pieces in the PCB board could cause a delay in our final product, due to reordering pieces. Too little power would have the system perform at far lower

levels than expected. The device for the most part wouldn't work. The importance of power distribution in specific lines would help produce a working product. We also need to make sure power isn't being wasted, this could have negative effects on the longevity of the running time for our LiDAR scanning system. Below we talk about many different components associated with requiring specific voltage and current levels to produce a working product for our system.

CCD Sensor: In order to integrate the CCD sensor, we must deliver the necessary power to this sensor. To turn on the EPC635 CCD sensor a voltage of up to 2.0 V can be applied into the chip. Once the chip has been configured to a specific mode of operation, the power consumption of the chip will vary depending on the mode of operation that the EPC635 CCD sensor is at. In the rest state when all actions are pending, and nothing has been initialized the EPC635 CCD sensor will consume 40 mW of electrical power. The mode of operation that consumes the most amount of electrical power is the conversion mode, this is when the integration has been completed and the rows are converting the data. The power consumption during this mode of operation is 350 mW. In the readout mode, when each pixel is sending out the captured information via the I^2C bus train consumes roughly the same amount of electrical power, 320 mW. Since the group plans to modulate the laser diode with the onboard modulator on the EPC635 CCD sensor, we need to make sure that the sufficient current and voltage are biasing the laser diode such that lasing can occur.

Servo Motors on RC Car: To achieve working statistics for the servo motors controlling the movement of the RC Platform 2, we would need to be able to operate at the levels that it's set to. After reading the manual that came with the RC Platform car for XiaoR Geek, the voltage listed is 12 V, these 12 V would be too great for what the microcontroller outputs. This came up in discussion, the use of involving an amplifier would produce this preferred level. The amplifier would be connected to the 5 V pin on the microcontroller, therefore creating an output of 12 V. The 12 V would be altered on either side of the car, this is where the team would perform left or right turns respectfully. The current listed for the servo motors is rated at 2000 mA. The techniques used to bring the voltage up would also be considered when planning to have the rated current. Since the microcontroller we plan on using would be separate from the rest of the machine, we could focus more on the production of this piece more, this would allow us to have more flexibility when increasing the output power.

Servos for Rotating Scanner: In our design, instead of using the RC car to create a rotation for the scanner, the group decided to use a motor to rotate it on its own. The motor we plan on using, or one like this, was the OVU00212. This component has different ratings depending on the amount of torque needed to perform its duties. The rated voltage associated with this system is about 12 V. The current related based on this device's datasheet, is 510 mA at max torque. The device uses DC voltage, since the current isn't alternating and rated at a set level, we could also investigate providing rechargeable batteries. This would also give us flexibility when initializing the power distribution throughout our system. Knowing this would be a separate function, this could be avoided when discussing the other components. The maximum speed of rotation of the OVU00212 can be measured up to 720 degrees per second. To ensure our system performs in a way we intend to, speed plays an important role as well. Although we may not need to perform at high rotation levels due to a compromise in stability, we need the rotation to be at a level that is set for the scanning of our device in harmony with the rays at which the data is being collected at.

Depending on the amount of power getting distributed to the system would help establish a better understanding of the amount of output power needed in production.

Bluetooth Module: The module we plan on using is the HC-05, this module is used to establish working communication between the microcontroller and the LiDAR system. Depending on the ascii value we send via UART along this connection, would make our device perform specific operations. With this design, we connected the HC-05 directly to the MSP430FR6989. The rate voltage it can operate at is in the range of 1.8 V to 3.6 V. Fortunately enough, the microcontroller we use has voltage outputs of 3.3 V or 5 V. Being able to connect to this directly gives us a plus in production. Knowing the output power isn't at a level that requires the use of voltage principles such as voltage divider, verifying the output power at all times would result in our system performing at the level we intend to use it at.

Vertical Cavity Surface Emitting Laser (VSCL): When dealing with the LiDAR scanner, this may be the most important. This type of laser is responsible for emitting light or an optical beam vertically for the top surface. Since the light is being charged perpendicular to the surface of where the laser is emitting from, multiple VCSELs can all be processed in a wafer at one time. The amount of power that can be associated with this mechanism can exceed power levels of about 4.3 W. Too little power in this system would result in the system not performing at the levels we intend to use it at. If the power is at levels that are too high, then it would be at risk of compromised and faulty pieces.

Above are a few parts needed in our design. The distribution of power throughout the entire system plays an important role, discussing the aspects in understanding the amount of power being dissipated, guaranteeing proper flow between the systems would make certain unnecessary power loss. When discussing the total power being produced by each system, many things are being considered. Moving closer to the final design of our product, we would have a better understanding of the power distribution we intend to want. The sections following this, we dive deeper into what each of these components do. Knowing the attributes these systems perform would give us a better understanding of proper power flow through our system.

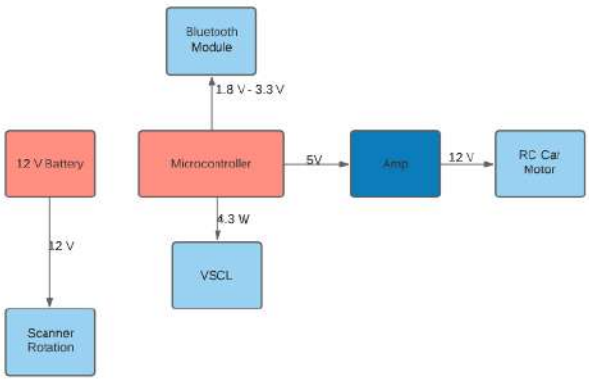


Figure 38: Power distribution Block diagram

5.3 Software Design

In this section the software design of the Mobile LiDAR Scanner will be discussed. The software needed for this is quite extensive as the hardware required for this project. Programming the microcontrollers to their data transmission policies and setting the configurations for the CCD sensor will need to be set before the scanning begins. As the data is transmitted from the Bluetooth module, the data will be pre-processed before any scans can be made. The distance data from must converted into a (X, Y, Z) format so that the data can be plotted in a 3D plane. Additionally, a GUI (graphical user interface) will be used to interface the user with the Mobile LiDAR scanner. Through this GUI, the user is able to set configurations for a particular scan, visualize the 3D point-cloud, and save the scan as a .las file for future viewing.

5.3.1 Laspy

Laspy is a public python library used for creating, reading, and modifying [21, 22] LAS/LAZ files. LAS files are generally organized into three main parts. It consists of the Header which carries certain information such as the point format and the version number. The header would be deemed as one of the important aspects because it stores the different dimensions for each particular point. The second main part is the VLRs or the Variable-Length Record. VLRs store any additional information like the Spatial Reference System, or SRS for short. VLRs also have components embedded into it. It contains a header and a payload, the payload which consists of and is limited to 65,535 bytes since the length of the payload is stored as uint16. The last part in a LAS file would be the Point Records. Point Records are also usually the biggest chunk of data, the format describes the dimensions stored for each point in the specified record. As the LAS file versions began to get updated, they added different types of formats to be used when reading the data. In one of the latest versions, LAS added EVLRs. EVLRs are similar to VLRs but have a higher payload associated with it. The length of the payload using EVLRs can be stored in uint64.

The basic manipulation when using Laspy is opening and reading. Laspy uses a function called `laspy.open()` to get the header and VLRs. This function is usually used when grabbing the metadata that's contained in the header. Although it doesn't read points in the file, this can simply be used for acquiring the specifications and dimensions of the file. The reading function when using the basic manipulation techniques for Laspy files is `laspy.read()`. If one wants to collect all of the information from the files including the data, the function would do so. The return from this function would be an object that can be used to access the data.

In the case of the file being too big to read for the RAM that we'd be using, we can parse the information in chunks to be able to read the file that contains all the data points. The file is `LasReader.chunk_itertor()` that reads chunk by chunk. The function allows the user's machine to process large files (filtering and splitting).

Writing data is the key component in this project. With the use of the LiDAR machine that's been discussed in early sections. The goal is to be able to use the microcontroller to input specific read functions to the LiDAR machine, then the data sent back should be LasData. The data returned would then be processed with the function `lasData.write()` to write to a file or to write to a stream.

5.3.2 Code Composer Studio

Code Composer Studio is a platform we used to be able to divert the pins for the microcontroller to read the information processed by the Bluetooth module. This platform is a software that assists in the development of embedded and debugs those applications. The C/C++ compiler, project build environment, editor, and debugger were key features in developing the communication for our system. Our code consists of multiple functions, some include the connections via UART and also coding the specified LEDs for error detection or confirmation of the received data bits, all of which give unique ingredients for the working application. In the section Hardware Specific Testing below, we give a thorough review on the specifications on the diverted pins and also an insight on the different components used to establish our communication. To be able to divert the pins correctly we needed to visit the microcontrollers Chip Data sheet and Family User's guide, which is explained in further details in the following sections.

In development, we used three four functions to be able to establish Bluetooth communication from our computer to the microcontroller. As mentioned in previous sections, our purpose was to be able to turn the LiDAR system on and off when we send different ASCII values. "1" is meant t trigger it on and "2" is meant to turn it off. We were able to create this with the different functions. The functions are `Initialize_UART()`, `uart_write_char()`, `uart_read_char()` and the main function. Gearing towards embedded systems, Code Composer Studio was able to help establish these functions to produce communication via Bluetooth. The first function `Initialize_UART()`, is a function that was meant to help divert pins. When first receiving the MSP430FR6989, the microcontroller has preset value on each pin. One needed to visit the microcontrollers Family's Users Guide and the Chip Data sheet to read which pins needed for transmission and receiving. The second function `uart_write_char()` writes characters from the buffer, then outputs it to the screen. The last function is `uart_read_char()`, reads the character into the buffer, then gets processed from the write function.

5.3.3 Embedded System

CCD Sensor: For the CCD sensor, it must be configured to meet our project's needs. This component of the scanner acts as the bridge between the photonic system and the embedded system. The sensor that was chosen for this project (epc635) is configured using I2C. Luckily, the epc635 has a predefined C-like functions that allow us to interface with the I2C master CPU on the epc635. In these functions, our microcontroller would send configurations using
Serial Data (SDA) and Serial Clock (SCLK) ports on the epc635 itself.

The epc635 sensor performs ToF measurements by averaging the data from 512 frames. There are two options for averaging, 2x DCS, where the number of frames is divided in two and each group is categorized as a DCS or Differential Correlation Sample, and 4x DCS where the number of frames is divided into 4 groups. In the 4x DCS mode, each DCS is matched by a π-delay with respect to each other. This phase shift is introduced at the laser modulation, and it is used to perform the ToF calculation. How the phase delay affects the accuracy of the measurements will be discussed in more detail in Section 5.3.7. For our application, during scanning of the area of interest, we will be using 4x DCS mode and the sensor in continuous measurement control (auto-run) mode by enabling bit 1 in the shutter control register 0xA4 (See Appendix A Section A.1.2). This will allow us to measure the distances of objects continuously as we perform a 360º scan. The

images taken will then need to be registered to render the 3D map by using simple, intensity-based image correlation algorithms.

For a single shutter pulse, 4 DCS readings are produced providing intensity readings of the received light pulse at 4 different points in time with a phase shift of $\pi/2$ (see Figure 38). This intensity reading with a π-delay matching offers an intrinsic correction for distance errors due to pixels offsets. Equation 28 is used to calculate the distances from the ToF reading, defining an offset from experimental test runs. In Section 5.3.7 we discuss alternative methods for establishing an offset for improving axial resolution.

$$D_{ToF} = \frac{2}{c} \cdot \frac{1}{2\pi f_{LED}} \cdot \left[\pi + atan2\left(\frac{DCS3 - DCS1}{DCS2 - DCS0}\right)\right] + D_{offset} \tag{28}$$

Here, f_{LED} is the modulation frequency of the laser diode or LED, c is the speed of light, $DCS0$ through $DCS3$ are the DCS least significant bit (LSB) readings, D_{offset} is the offset distance determined experimentally, and $atan2$ determines the phase between $DCS3 - DCS1$ and $DCS2 - DCS0$.

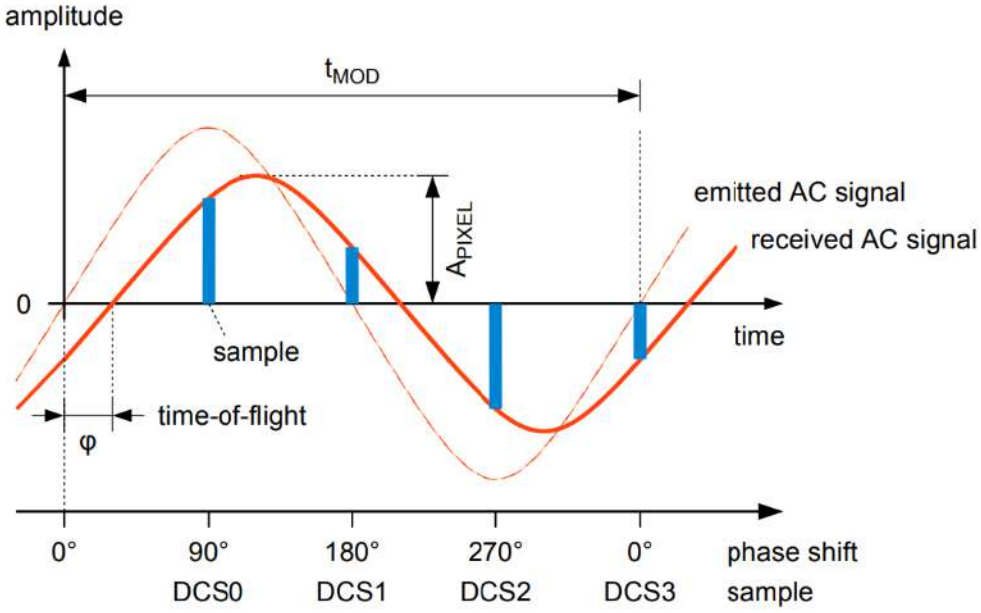


Figure 39: DCS Phase Shift. From [6]

In auto-run mode with 4x DCS, the times it takes to perform one ToF distance measurement for 8-bit and 12-bit DCS data are 6.108 ms and 7.789 ms, respectively. Although we are performing a horizontal scan with vertical cross sections, these cross sections are several pixels thick, thus each ToF measurement comprises a 160 × 60 × 9 (8-bit) or a 160 × 60 × 13 (12-bit) data matrix; therefore, after each measurement is performed, the scanner should be rotated to displace the observed area only a few pixels before the next measurement is taken. The time it takes for one measurement, however, is not the time the sensor needs to be exposed to the incoming beam. In fact, the shutter open time is much shorter (about 250 ns), thus we can begin to rotate our scanner

to the next position much sooner. The short shutter time will also minimize any jitter in the images from movement. The scanning speed is discussed in more detail in Section 5.2.4.3.

Microcontroller: The MSP430FR6989 is planned to be used for the data transmission of the LiDAR data from the CCD Sensor to the host computer where the 3D point-cloud representations will be rendered. As discussed in the previous section, the ToF calculations will be handled by the CCD sensor. This frees up the MSP430FR6989 to handle transmitting the captured data from the CCD sensor to the attached Bluetooth module. The information is then transmitted to the host computer from the Bluetooth module

The distance data transmitted by the CCD sensor, ranging from 8 to 12 bits of DCS distance data on each pixel, in a parallel fashion. This means that 8 to 12 pins have to be set as a GPIO (general purpose input/output) pins to read the pins on the sensor. The MSP430FR6989's clock only goes up to 16 MHz, as a result our CCD sensor will have to capped at the same frequency. This is to minimize data loss due to new data overwriting stale data that needs to be sent through the host computer. If our team decides to use the 8-bit (byte) format due to the fact that are more interested in distance data, the byte is transferred to the GPIOs at every clock cycle. Software-wise that means that the data from GPIO pins is captured in different variables and concatenated together into a single byte variable.

To transfer the data, the microcontroller will transmit the data using the Bluetooth module. An example of the process of using UART with the MSP430FR6989 are discussed Section 6.5.1. The Bluetooth module has two ports, a transmitter and receiver, that are attached to two pins on the MSP430FR6989's receiver and transmitter respectively. In a function called `Initialize_UART()`, two pins have to be diverted initially to UART functionality acting as the transmitter and receiver. The clock used for the UART transmission is set, in our case we will use the sub-master clock (SMCLK) and set the modulators and dividers to get the baud rate to 9600 Hz (same frequency used by the Bluetooth module). Oversampling may not be an option since the timing is so tight between the CCD sensor and the microcontroller. Finally, we exit the reset state to allow for UART transmission/reception. We build the functions `uart_write_char()` and `uart_read_char()` to transmit and receive bytes respectively. The `uart_write_char()` function takes an unsigned character `ch` as an argument and waits for any ongoing transmission to complete using a while-loop that checks if the transmission flag on the UART register equals to 0. If it does, the it copies `ch` to the transmission buffer, where it is transmitted to the Bluetooth module. The `uart_read_char()` does not take any arguments and checks to see if a byte was received by checking the receive flag on the UART register. If it finds the flag raised, the function will capture and return the received character. If the flag is not raised, it will return a NULL character.

In the main() function, we first called the `Initialize_UART()` function to divert the pins to UART functionality and configure our UART settings in a way that is compatible with our Bluetooth module's own UART function (9600 baud, 8-bit data, no parity, 1 stop bit, LSB first, no flow control). Inside of an infinite loop, the 8-bit distance data is captured into a variable using the `uart_read_char()` function. The distance data in a variable is transmitted to the Bluetooth module using the `uart_write_char()` function. Some interrupt system may have to be implemented to avoid a race condition.

5.3.4 PC Application

The operation of taking the data from the LiDAR system and drawing the 3D point-cloud representation of the data is a simple operation when looking top-down. Figure 18 above shows how the software interacts between the microcontroller and the personal computer held by the user. The LiDAR system is controlled by the microcontroller, which is on board the RC platform. The user's personal computer must have Bluetooth capabilities to interface with the Bluetooth module that is connected to the on-board microcontroller. To turn on the LiDAR system, a start/stop button will be available to the user on the application. A previously determined string of bits is transmitted to the Bluetooth module connected to the microcontroller. Based on those command bits, the LiDAR will begin its scan or cease the current scan.

As the LiDAR system scans the area, the raw LiDAR data will be transmitted back to the PC application through the Bluetooth module. The application must have some sort of driver to be ready to receive the raw data. The data from the LiDAR will have to be converted into a form that is useful for us. Using a Python library called *laspy*, an open-source library used for reading and editing .las files, the raw data has to be converted into a .las file.

With the .las file created from the raw data, the application will use the *laspy* library to plot out the points of the scan. These points then can be used in conjunction with the OpenGL library to create the 3D point-cloud mappings of the scanned area.

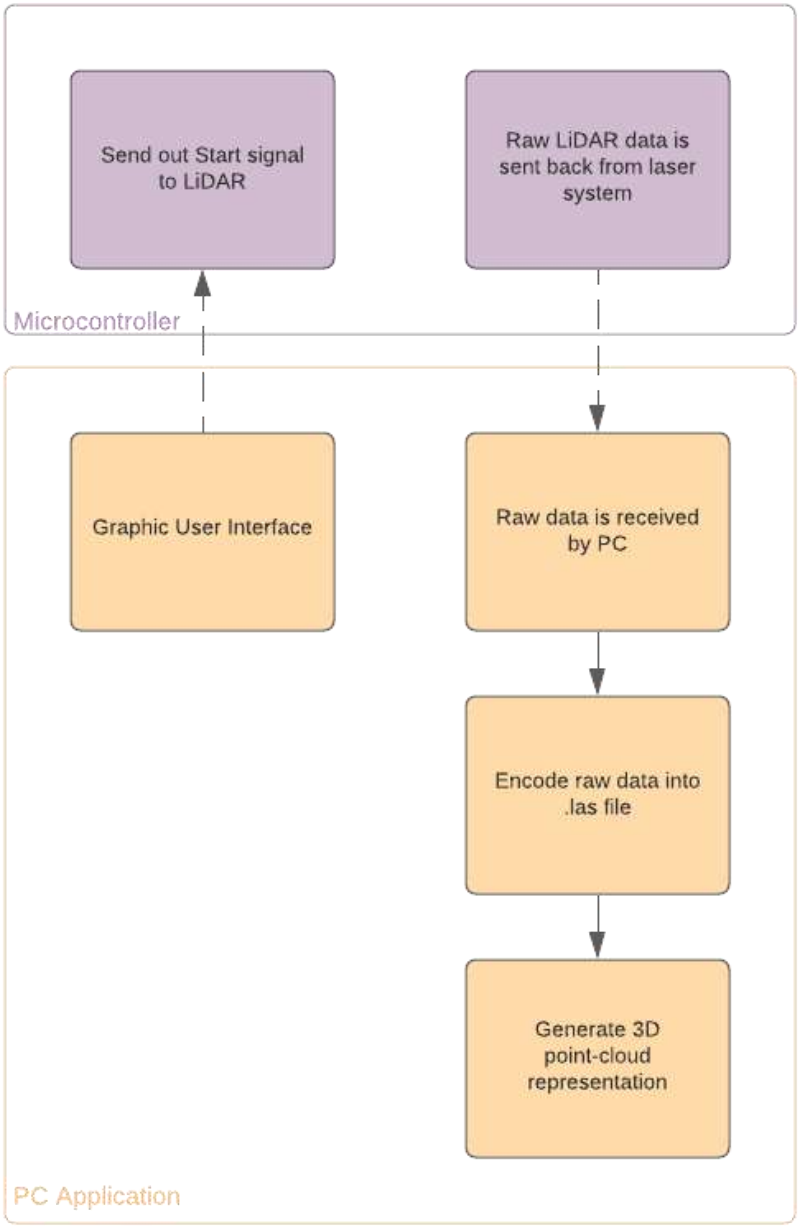


Figure 40: Software block diagram for Application

5.3.5 Graphic User Interface

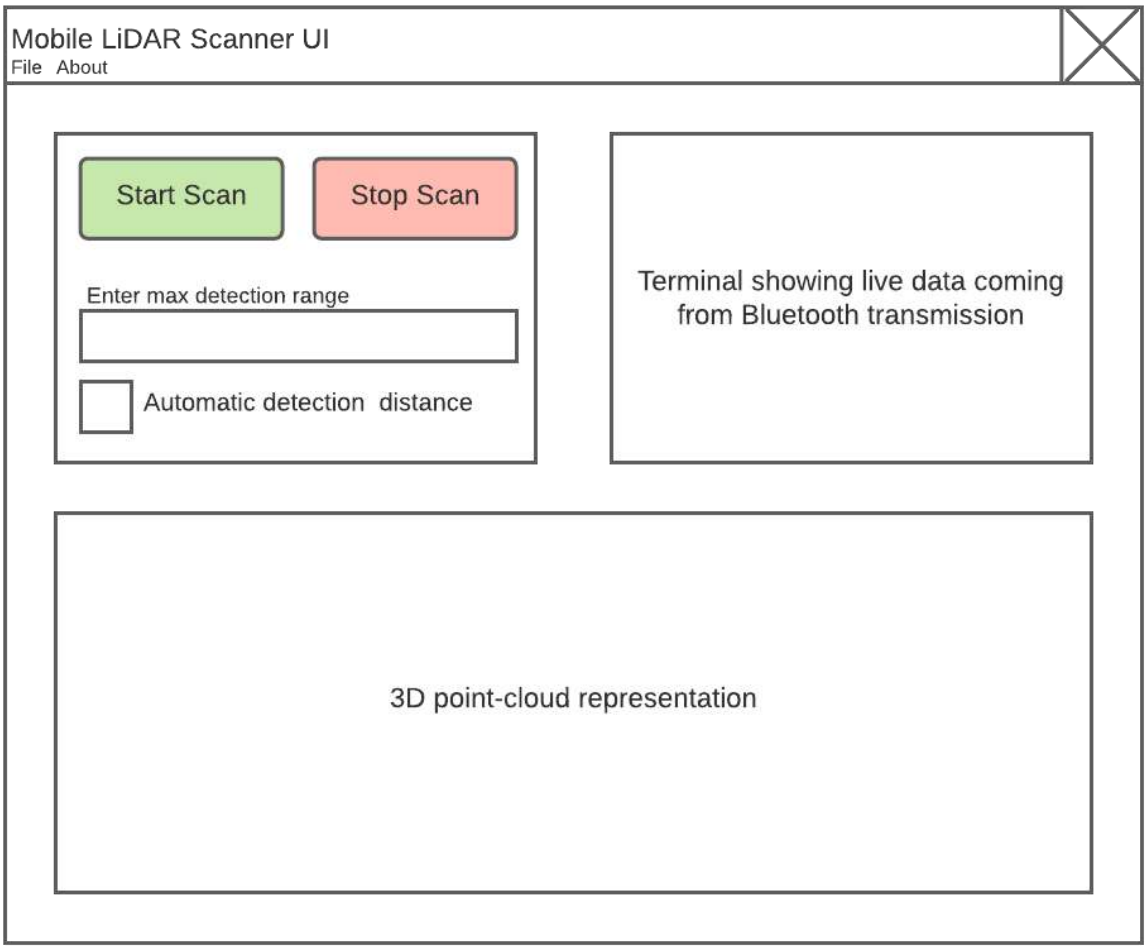


Figure 41: Basic GUI mock-up

The figure above (Figure 41) shows a simple representation of what the final GUI will look like for the PC application that will accompany the mobile LiDAR scanner. The user will be able to start and stop the scanner using two buttons on the top. When the stop button is pushed, the current scan will be canceled. Options are given for custom maximum detection distance.

A terminal-like display will be shown in the upper-right corner of the screen will list out the data received from the Bluetooth module. This data should be coming in the format (x, y, z) coordinates. Finally, the display is at the bottom of the screen. When not displaying 3D point cloud, it will tell the status of the program. Depending on the status, the display will read “No active scan”, “Scan in progress…”, or “Cancelled previous scan”. Once a scan has been completed, the 3D point cloud will be displayed. We may want to include the option to allow the user to save and open and view their own .las files from previous scans

To build this this application, two Python libraries will be used: PyBluez and PySimpleGUI. We have tested with the HC-05 Bluetooth module previously by connecting directly through the

settings window in our computer. Using the PyBluez library, we are able to establish a connection in the program itself. PyBluez first creates a `BluetoothSocket` object to represent the endpoint of a Bluetooth connection. In our case, this will be the Bluetooth module handling the data transmission between the mobile LiDAR scanner and the computer. The `BluetoothSocket` object will use its `accept()` method with a `BluetoothSocket` and Bluetooth address as parameters. Once the connection is made, we are able to read data and send commands using the `listen()` and `send()` methods respectively. While the data is being read When the application is closed, the Bluetooth connection will be closed using the `close()` method.

The graphical user interface is planned to be built using the PySimpleGUI library. This library is a wrapper library that encompasses other GUI packages to make them easier to use: Tkinter, PyQt, wxPython, and Remi. In PySimpleGUI, widgets like windows, labels, and buttons are referred to as `Elements`. Each of these elements act as an object and have their own methods and variables exclusive to them, so object-oriented programming is essential in this aspect. Currently for development, we have been using the matplotlib library to test the sample LiDAR scans and printed them onto a 3D space. However, for the final user interface, we are hoping we are able to use the PyOpenGL library for a better look of the 3D point-cloud representation. While matplotlib provides a nice three space to look it can be a bit slow to move around and view the whole point cloud.

Unfortunately, the majority of our team is developing software and conducting tests on Microsoft Windows machines. The PySimpleGUI library also only as documentation concerning packaging the application for Windows machine. As a result, the resulting application that will accompany our mobile LiDAR scanner may only be available to Window machine. Mac and Linux users will either that to compensate by using a virtual machine running with the Windows OS or invest in another computer.

5.3.6 Point-cloud representation

The end product of our mobile LiDAR scanner is the 3D point-cloud representation of the area that was scanned. The data coming in from the Bluetooth module will be either 8 or 12 bits long, transmitted using UART, measuring distances from the LiDAR system to the object of interest. This distance data needs to be converted into a form that able to plot in a 3D plane. Thus, it will need to convert to data points with an x-, y-, and z- component. The angle of the mirror will have to be considered to convert the distances to usable points.

The mobile LiDAR scanner will be represented as a circle with a dot. Once the data points are calculated, these points can be stored into an array of vectors. Using this array, we can draw each of the points into a 3D plane. As the points are being drawn on the 3D plane, the laser is represented as the scan is drawn as a line from the center of the LiDAR scanner. Figure 42 shows how this would look like. The final scan may not have the same scale as seen in the figure. The vectors can be saved as a .las file for future viewing.

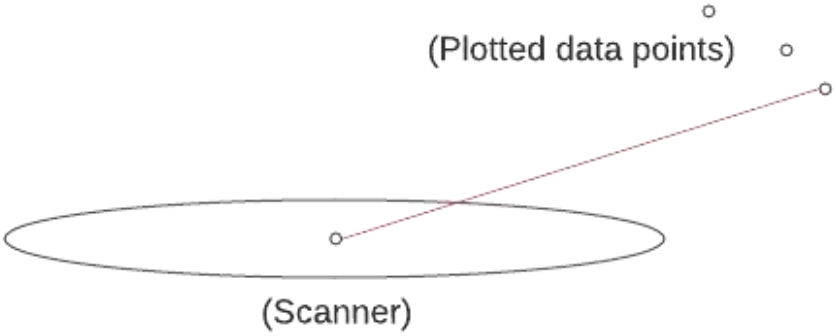


Figure 42: Example of point-cloud creation process.

5.3.7 Distance Calculation and Error Correction

In this section, we focus on discussing how the distance calculation accuracy from the ToF sensor and resolution can be improved through various techniques implemented through the programming of the image acquisition and data processing. The accuracy of the distance calculation is of course affected by several physical parameters which will be monitored during the scanning and the program will determine the optimal values of these parameters to achieve the best reading. In addition, the axial and transverse resolution of the system can be improved through image processing and averaging of overlapping scans to provide data redundancy and improve the signal to noise ratio.

Monitoring and optimizing illumination parameters: A main factor in the quality of the ToF reading is the illumination. Of course, there are many parameters that comprise the illumination of our system. Mainly, our program will focus on determining optimum illumination power, FOV, tilting angle, modulation frequency, and phase shift with respect to the ToF detection demodulator. These parameters are intertwined and for a given detection range they often must be analyzed simultaneously. Let's take, for example, a detection range of 10 m with a retroreflective target. At this range, the maximum modulation frequency in the simplest of cases is determined by Equation 2 to be 15 MHz. Our program should take the desired maximum detection distance from the user and adjust the modulation frequency accordingly using Equation 2. In the case of the detection distance is set to automatic, the system will set the modulation frequency accordingly after performing the diagnosis scan and determining the appropriate detection range. Another parameter that needs to be adjusted to the defined detection range is the FOV of both the illumination beam and the detector. Along with the FOV, we need to consider the adjustment of the inclination angle of our scanner, since it will also affect the height of the objects that can be detected at each distance. The adjustment of the FOV and tilting angle, however, will not be continuous with the detection range, but rather discretized to accommodate for the precision limitations of our FOV tuning mechanisms. The discretization of the FOV and the tilting angle will be defined as shown in Table 17, with smaller steps for shorter ranges, since changes at short distances affect the height of objects that can be detected more significantly than at longer distances.

Table 17: FOV and Tilting Angle Settings

Detection Range (m)	Full FOV (deg)	Tilting Angle (deg)	Max. Height Detected Range (m)
[1, 3)	45	8.4	[0.85, 2.05)
[3, 6)	43	16.7	[2.61, 4.97)
[6, 10)	40	10	[3.71, 6.02)
[10, 15)	35	6.4	[4.68, 6.89)
[15, 20)	25	4.6	[4.86, 6.40)
[20, 30)	15	3.1	[3.99, 5.86)
[30, 40)	10	1.8	[3.83, 5.02)
[40, 50)	5	0	[1.99, 2.43)

For each distance to be detected, the FOV and tilting angle is defined until the next distance in the table. For example, for detection distance in range [1, 3) the distances are between 1 m (including) and 3 m (not including) and the full FOV and tilting angle are defined as 45 and 8.4, respectively. The full FOV for each range is defined arbitrarily, considering that much narrower FOV values are required for detection distances longer than 10 m than it is needed for detection distances from 1 m to 10 m. The inclination angle is then determined roughly as the angle at which the outermost ray for the defined FOV will hit the ground at the defined detection distance. However, considering that the illumination beam is Gaussian, for longer detection distances we want to have the center of the beam closer to the desired target height, thus the inclination angle obtained from the previous analysis is reduced to match an approximate height of 1.5 m of the center of the beam at the upper limit of the detection range. Finally, the maximum detected height can be calculated using Equation 29 where y is the height of the sensor (estimated to be 25 cm), r is the detection distance, FOV is the full FOV angle, and α is the tilting angle. Figure 43 shows how the illumination beam FOV determines the height of the detected object at a specific detection distance.

$$h = r * tan(FOV/2 + \alpha) + y \tag{29}$$

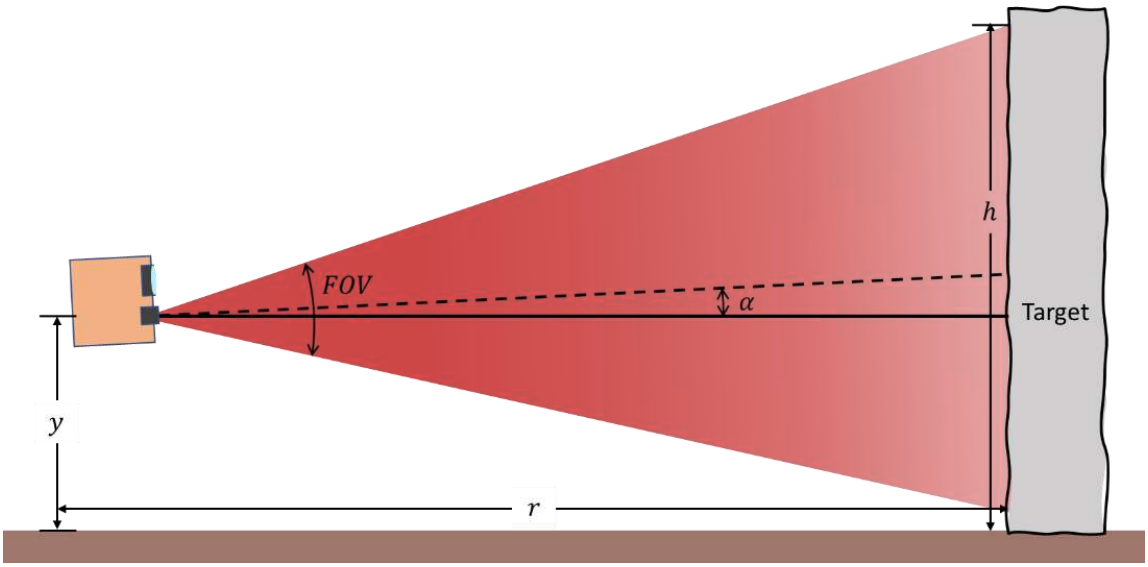


Figure 43: Adjusting Illumination FOV for Given Detection Distance

With the FOV, tilting angle, and modulation frequency optimized for a given detection range, the next parameter to adjust is the illumination power. More power is required for targets farther away, as well as for targets with lower reflectivity. Since in a given detection range there can be different kinds of target who reflect different percentages of the illumination power, it is not efficient to define a specific output power for each different scenario. Instead, it is more appropriate to monitor the detected power and adjust the illumination power accordingly. If for a given detection range, the detected signal has a low signal to noise ratio, the output power needs to be increased. The challenge with this is that our ToF sensor performs the distance measurements automatically so the data we receive is not of the detected intensity. However, for ambient light calibration, the ToF chip takes a grayscale picture which can be used to determine if the SNR is high enough or if the output power must be increased. With the grayscale image, to determine the SNR, we need take the average pixel value throughout the image. If the average intensity is below an experimentally determined threshold, the output power is increased gradually while continuously monitoring the average pixel intensity until it is above the threshold. The issue with this method is that taking the average intensity means that outliers can affect the SNR reading. Since the illumination beam is Gaussian, the intensity will be higher at the center, dropping exponentially towards the edge. Therefore, the average should not be taken for the entire image but rather of a region of interest selected towards the corners of the image. Then the intensity of the marginal light rays will be high enough to detect objects at the edges of the FOV with high SNR. In the case of pixel saturation, power needs to be decrease. If the power is not too high as to cause damage to the sensor, which would only be likely for retroreflective targets, the saturation can be remedied by decreasing the integration time for each frame. Fortunately, the epc635 chip has an automatic process for adjusting integration time to reduce pixel saturation (see Appendix A, Section A.2).

Using the epc635 CCD chip for ToF measurement, a phase delay can be introduced to the laser modulation with respect to the demodulation of the backscattered light collected at the detector. To add this delay, bit 2 in register 0xAE (see Appendix A.1, Section A.1.2) needs to be set to enable Delay Locked Loop DLL [6]. For different detection ranges, the phase delay must be adjusted so the ToF values for each DCS reading are accurate within the defined range. This occurs when the DCS values closely match (see Figure 44). The delay of the laser modulation can be

administered in discrete steps t_{DLL} whose width vary with temperature and device and can be determined using the value of register 0xE9 in Equation 30 (see Appendix A.1, Section A.1.2).

$$t_{DLL} = [(\text{register 0xE9} - 128) \cdot 0.003 \text{ ns}] + 2.1 \text{ ns} \quad (30)$$

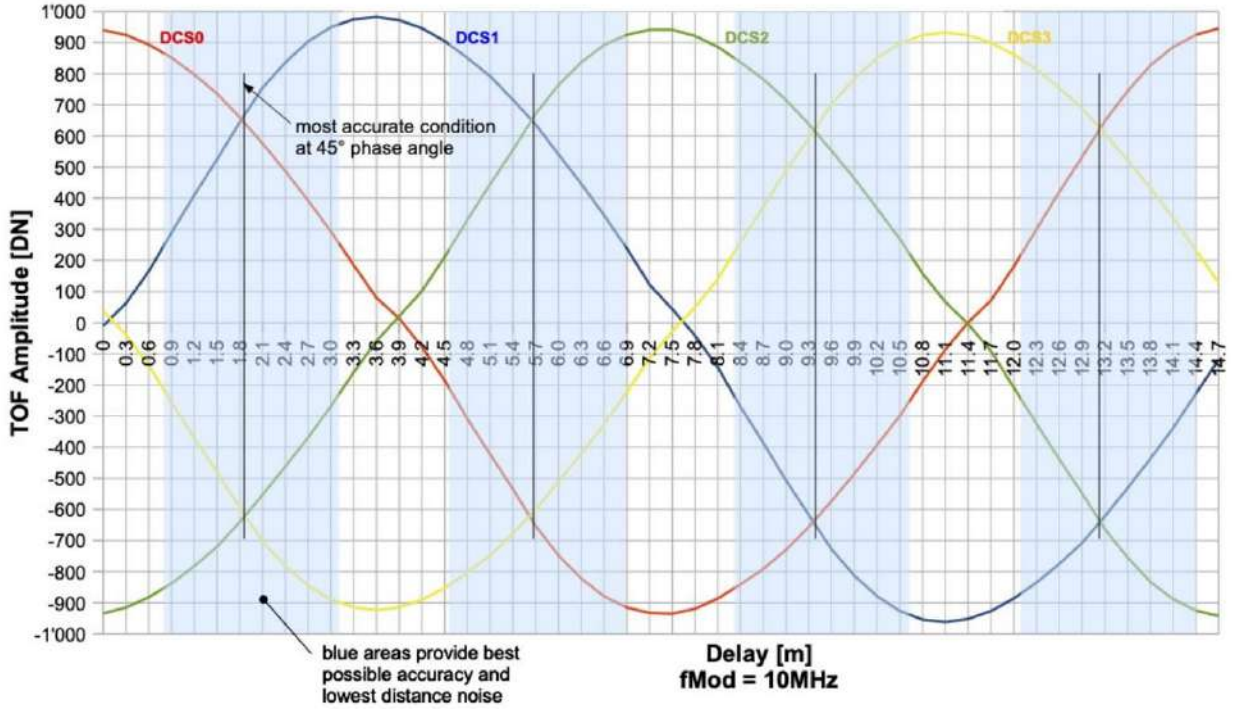


Figure 44: DCS amplitudes for the 4 DCSx (measurement data) [6].

The total delay time is given by $n \cdot t_{DLL}$, where n is the number of steps defined by the value of register 0x73 which goes from 0 to 49 (see Appendix A, Section A.1.1). From Figure 44 we see the ToF amplitudes for each DCS measurement at a modulation frequency of 10 MHz. The blue sections represent distance ranges where the ToF measurement would be most accurate. Notice that the unambiguity distance for 10 MHz modulation is about 15 m, which means that if if we wanted to detect objects from 1 m to 15 m, the phase and modulation settings shown in Figure #201 would leave gaps of low accuracy measurements in between the detection range of interest. This can be fixed by doing three scans, each with a different modulation phase delay separated by a third of the unambiguity distance or the pulse width. For example, for the 15 m detection distance with modulation frequency of 10 MHz, figures 45 to 47 show the DCS measurements for phase delays of 0, T/3, and T/6, respectively. Notice that the gaps in Figure 45 with no phase delay can are filed in in figures 46 and 47 by the phase delays with increments of one third of the pulse width. Figure 48 shows the sections of best accuracy for each phase delay. Notice now there are no gaps, thus theoretically, the full range of distances from 0 m to 50 m should be able to be measured with good accuracy.

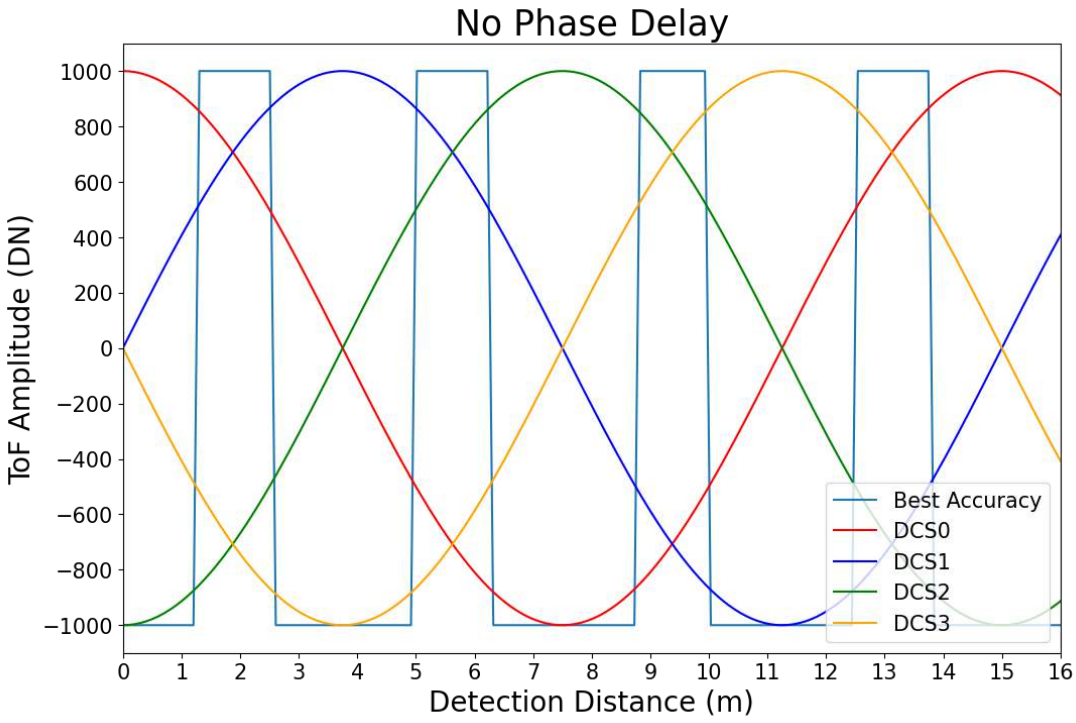


Figure 45: DCS Data, 10 MHz, No Phase Delay

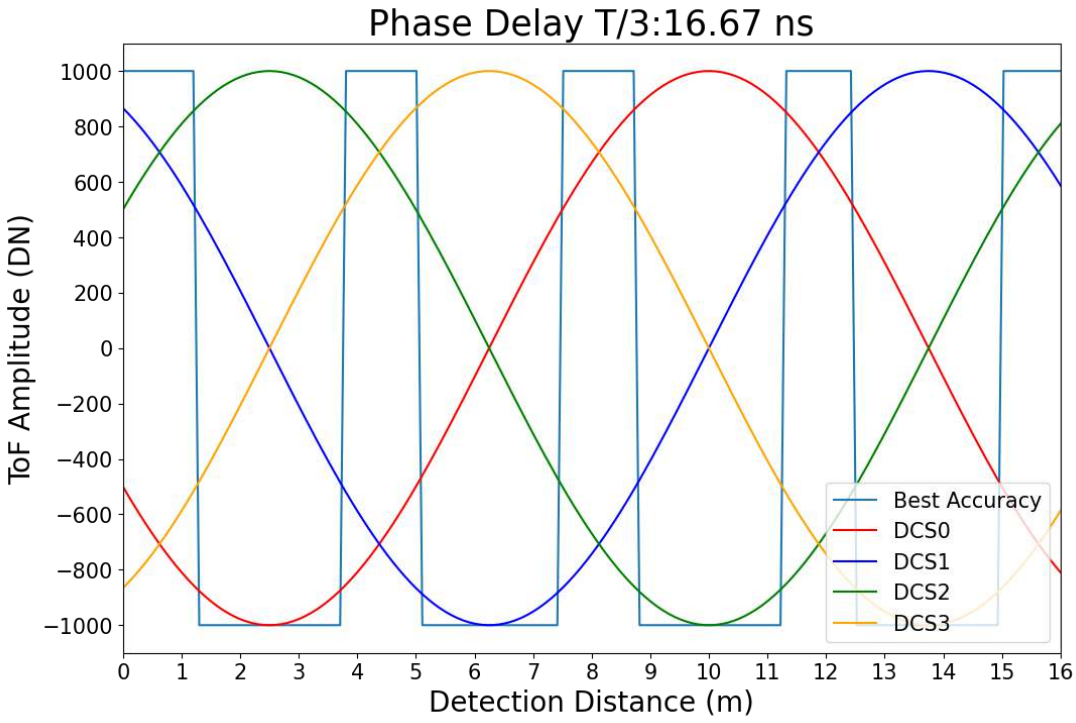


Figure 46: DCS Data, 10 MHz, 5 m Delay

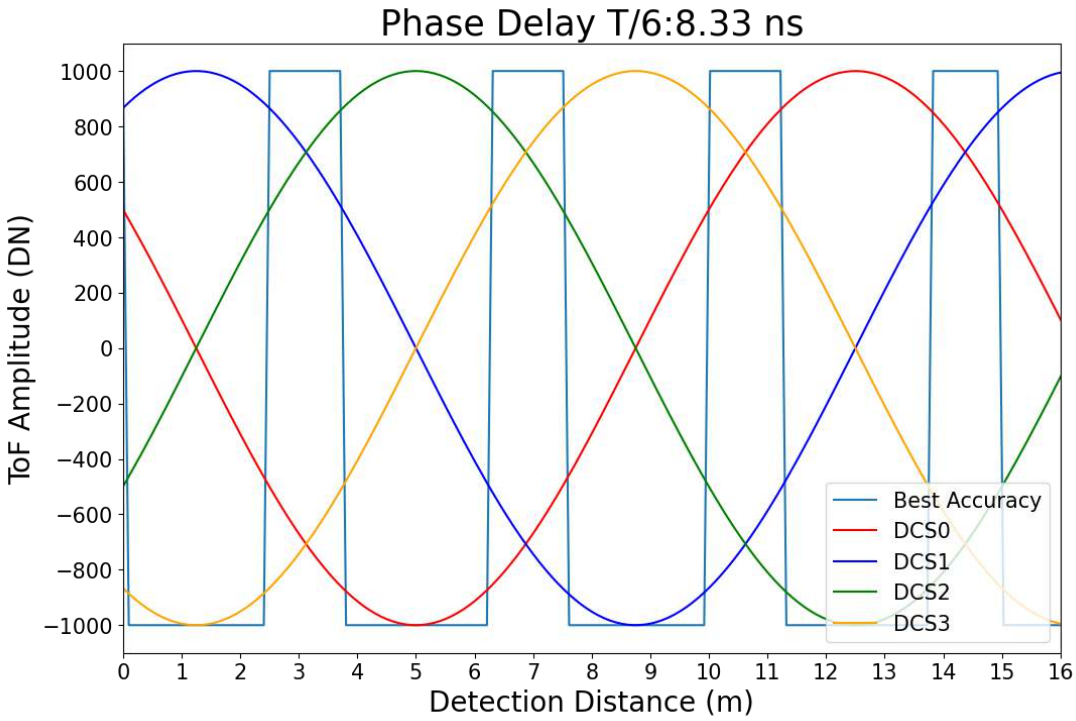


Figure 47: DCS Data, 10 MHz, 2.5 m Delay

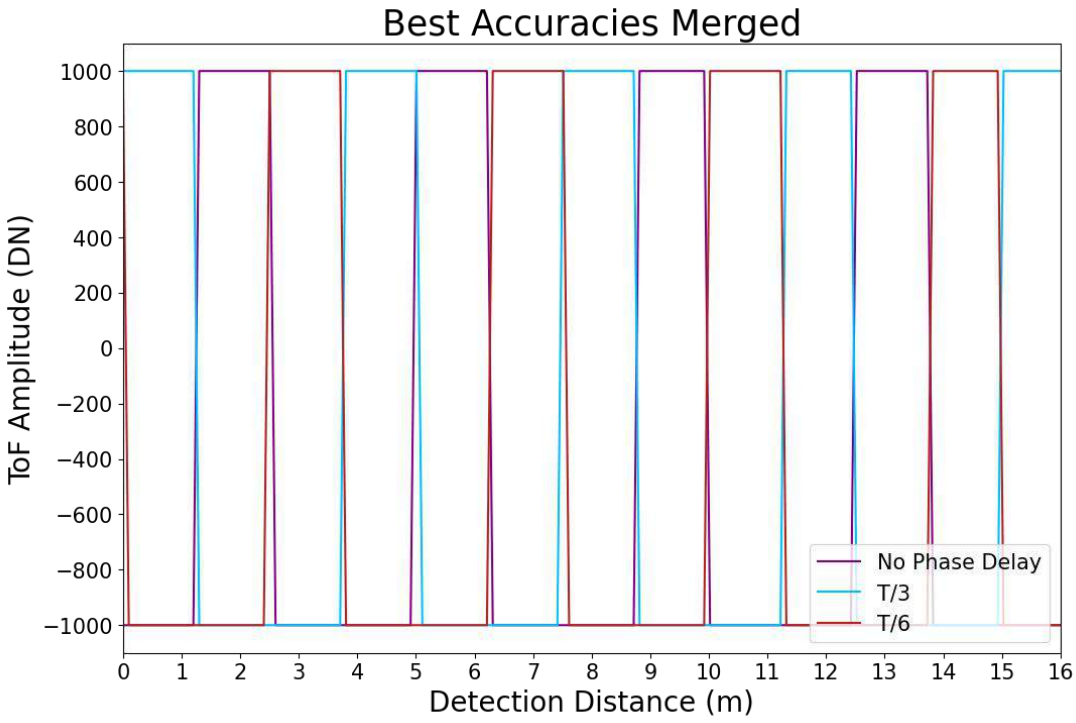


Figure 48: Best Accuracy Regions, 3 Scans, 10 MHz

The maximum delay that can be applied is of $50 \cdot t_{DLL}$, which is approximately 100 ns, corresponding to a 15 m delay. This means that for detection ranges with unambiguity larger than 15 m a different method must be used. For our maximum range of 50 m, for example, a combination of phase delay and different modulation frequencies can be used to detect objects closer to the scanner. If the user defines a detection range that does not include nearby distances, less scans need to be made and the phase delay does not need to be applied as much. Ultimately, for detection distances shorter than 15 m, a phase delay can be applied to successfully cover any range of distances within the unambiguity distance; for detection distances larger than 15 m, the number of scans needed to detect all objects in wide detection ranges increase significantly. That, combined with slower scanning speeds (see section 5.2.4.3), increases the required time for complete data collection and 3D map rendering for longer distances.

5.4 Summary Design

In this section we summarize the main hardware and software designs for the Mobile LiDAR Scanner. Table 18 presents the designs that have been selected and will be implemented into the Mobile LiDAR Scanner. Table 18 will also contain hardware designs that are still under consideration such as component housing, mobility mechanics, and rotational mechanics.

Table 18: Design Summary

Design	Outcome
Illumination Optics	3 lens optical system composed of a biconvex lens, plano-concave lens, and a cylindrical lens.
Detection Optics	3 lens optical system composed of a biconvex lens, plano-concave lens, and a biconvex lens. This design also includes a NIR bandpass filter.
Component Housing	3D printed housing whose design is still under consideration.
Mobility Mechanics	Modified RC car where the LiDAR and rotation stage will be placed on top of.
Rotational Mechanics	Rotational motor with a flat base on top where the LiDAR will sit on. The rotation stage will allow the LiDAR to scan 360°
Microcontroller data transmission protocol	UART, I2C
Wireless connection module	HC-05 Bluetooth module
Software	Python language, Laspy python library

6.0 Project Prototype Testing

In this section, we will go over testing that has been done as well as plans for testing the prototype once it has been built. The section discusses the procedures and plan we have for each subsystem for our Mobile LiDAR Scanner. This would also include the many tests we've done, our faults and what we did to overcome then. After all further testing in each of our subsystems are complete, then we should be able to develop a working prototype.

The illumination system of our scanner must meet several criteria, most of which were considered in the design process. Testing our system to make sure these criteria are met is important. In addition, some performance parameters are best determined during testing of the system to define the limits of our system. The main parameters that need to be tested in our illumination system are optical power for various pulse frequencies and FOVs using both the laser and the LEDs, performance of optics and mechanical system for beam expansion and FOV tuning, and system robustness during scanning. These tests will be done independently of the remaining of the scanning system. Upon integration, several other tests of the mobile LiDAR system as a whole will be carried out to determine several parameters of the general system performance. Optimization of tilting angles and FOVs for specific target distances will be done through a series of tests of the entire system to determine the best combinations. In this section, however, we will only discuss tests of the illumination system independently of the rest of the system.

6.1 Testing of the Illumination System

Illumination power and modulation: For mid-range and long-range detection, the illumination system relies on the VCSL. Determining the maximum optical power that can be obtained for various modulation frequencies and pulse widths will allow us to find the optimum frequencies for different detection ranges. Firstly, the maximum power of the VCSL will be measured for continuous wave (CW) operation. This value will serve as the minimum operational power for single-target detection. The illumination power can be increased by pulsing the laser, as the peak power will increase for shorter pulse times. To test the illumination power with different modulation frequencies and duty cycles we use a function generator and electrically modulate the VCSL directly. The VCSL will be placed in the circuit shown in Figure 49, using a transistor to allow for faster modulation with lower electrical strain on the laser. The design of the circuit is similar to the one that will be used in the integrated system. Another aspect of this testing procedure is to optimize the circuit to achieve higher modulation frequencies and shorter pulses. This can be done through simulation, however, and it is more convenient and practical than varying the capacitance or amplification factor of the circuit experimentally.

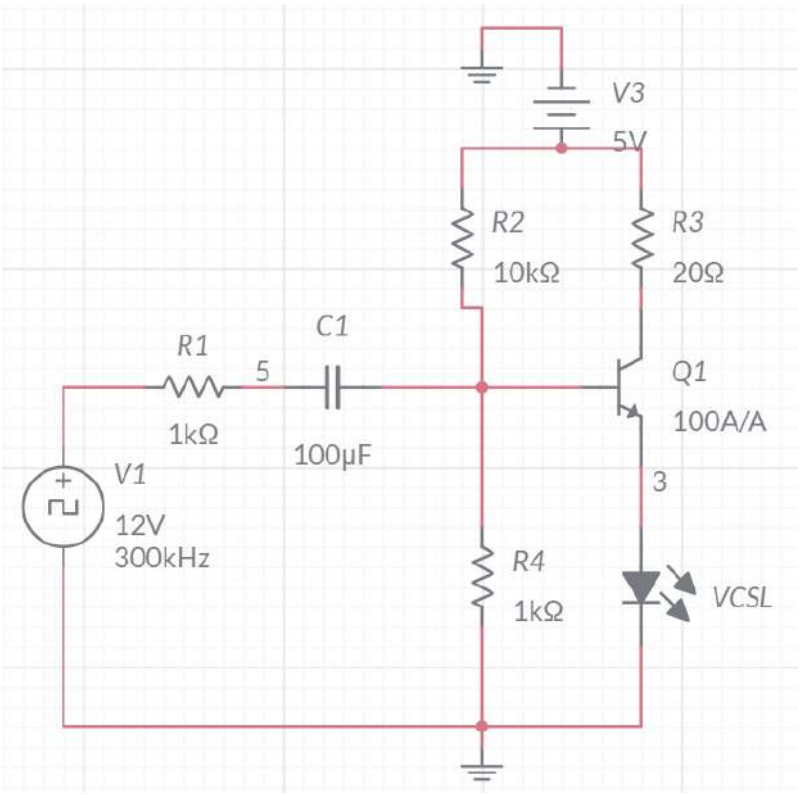


Figure 49: VCSL Modulation Circuit

Once an optimized circuit is realized for our VCSL, the maximum power will be measured for frequencies of 3 kHz, 300 kHz, 600 kHz, 1 MHz, and 3 MHz. At each frequency the duty cycle will be changed between 50% and 1% (e.g., 50%, 30%, 10%, 1%), and the power will be measured for each duty cycle. The frequency and duty cycle combinations that yield the highest peak powers and average power above the CW power will be used for detection at the distances allowed by the modulation frequency. The optimal powers determined here will be used for later tests of the required illumination power for target of different reflective and scattering properties at different distances. For the maximum distance of 50 m, the illumination power required for a black target with high absorption will likely be substantially high, thus it is important to know what the limitations of our systems are before defining for which target the maximum 50 m detection range is applicable.

In the case of short-range detection, LEDs can be used as the illumination source since less power and less coherence of the beam are needed. With an array of LEDs around the sensor, a wide area can be illuminated simultaneously as used in Flash LiDAR, and the resolution of the measurements can be improved by averaging the values of overlapping data values. The power required for this scenario is lower than that required for the VCSL; nonetheless, it is important to understand the limits of our system even at short-range detection. Therefore, similar testing procedures as the ones carried out for the VCSL can be applied to the LEDs. A similar circuit to that showed in Figure 49 can be used for the modulation, measuring the illumination power at various modulation frequencies and duty cycles. It is important to note that LEDs have much longer response times than VCSL, thus the duty cycles that can be achieved for high frequency modulation will not be as small as those possible for a VCSL. In addition, for better comparison of the results, the

Irradiance in power per unit area will be recorded for both VCSL and LED measurement at the operational wavelength of our scanner (940 nm). Since LEDs have a wider emission spectrum, constraining the power measurements to the peak wavelength will allow for a more accurate comparison. The goal of these tests is to determine if certain frequency and duty cycle combinations of the LED illumination is more efficient for short-range detection than the VCSL's, as well as to examine the versatility of our system for varied detection range selection.

Beam spreading optics: A key feature of our system is its capability of adjusting the illumination for optimal power delivery to the target. Tuning the FOV of the scanning fan tailors the intensity profile of the illumination beam to the size of the target at the specified distance. It is therefore of crucial importance to make sure the tuning of the FOV is accurate, precise, and efficient. The accuracy of the FOV tuning system will be largely dependent on the sliding mechanism for the adjustable lens. This mechanism will most likely be composed of a motorized actuator with a long, threaded rod that will move the lens along the axis by rotating the rod. The threading of the rod and the minimum rotation step of the motor in the actuator will determine the precision of the system. Small threading and small rotation steps will allow for more precise translation but will decrease the speed of the tuning. The determination of the positioning for each FOV is the first step in this experiment. By measuring the size of the beam on a target at a fixed distance, the position of the adjustable lens for each FOV can be determined. These positions are mapped to specific number of rotations of the motor to be able to change the FOV to the necessary values. This mapping method is however a bit restrictive as it only allows for quantized positions and values of the FOV. A better approach is determining the relationship between lens translation and FOV from the measurements and using that relationship to determine the position needed for any given FOV within the permitted range. Once the relationship is established, the system can be tested by repeating the tuning process for a random input sample of FOVs and finding the error between output and input values (the output values can be measured in the same way as in the previous experiment for determining the translation-FOV relationship). With this method, both the accuracy and precision of the system can be determined.

The robustness of the tuning mechanism against repeated use is a factor that affects both the precision and the lifetime of the system. When changing the FOV in quick successions, the accuracy and precision of the tuning mechanism will decrease, thus it becomes important to understand the maximum speed and rate at which the FOV can be changed while maintaining acceptable error. This can be tested using the same method described above for rapid changes of the FOV by feeding the input FOVs at a quicker rate.

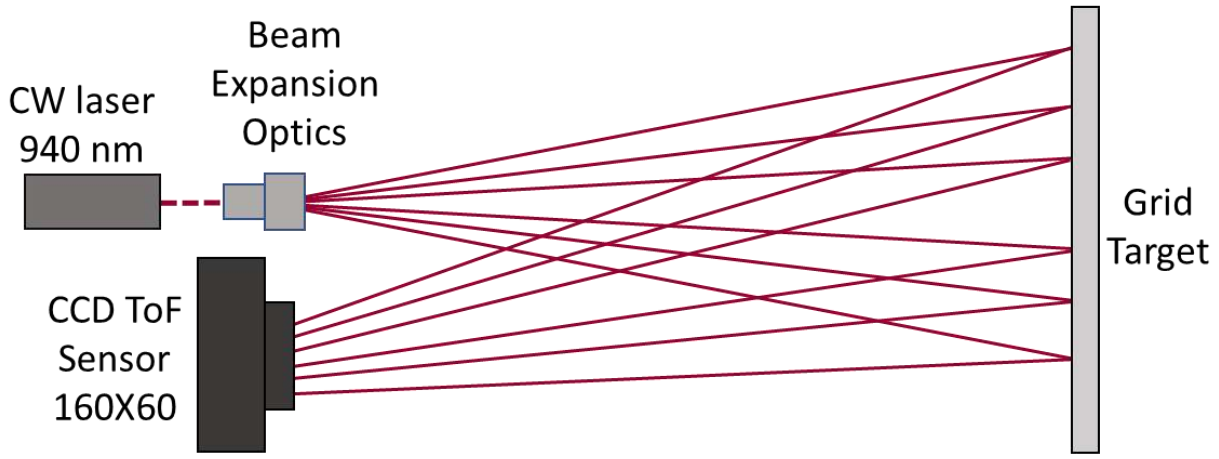


Figure 50: FOV Testing Setup

The challenge is measuring the output FOV. Since we are operating at NIR wavelengths, the measurements need to be done using a camera. Figure 50 shows a basic setup to carry out the measurements. The observed beams on the target will appear as shown in Figure 50. The ruler on the target can be used to determine the FOV of well aligned beams with respect to the target where the precision of the ruler can provide an accurate measurement. This occurs for beams centered at the center of the target and with edges right at the tick marks. For intermediate beams where the precision of the target is no longer enough for an accurate measurement, the magnification of the camera can be used, with the initial FOV measurement as a reference, to calculate the FOV more accurately. The distance of the target to the CCD camera should be constant. Since our system should be capable of varying the FOV from 5º to 45º, the beam size can be quite large for long target distances, thus, for a target 10 cm by 10 cm in size, it is best to keep it within 9 cm from the beam expansion optics. This method allows us to accurately measure the output beam FOV to determine the error rate of our beam expansion system. However, it requires that our scanner remain static pointing in a fixed direction.

Our system is required to render a 3D map of a 360º scan, thus our illumination system must be rotated to scan the area of interest. This raises the issue of ensuring that our FOV tuning mechanism is robust against the rotation during scanning. The testing method described preciously does allow us to measure the output FOV while scanning, thus a different method must be implemented. The first robustness factor that should be tested in the FOV tuning system during scanning is any unwanted movement or translation of the adjustable lens during rotation. While the other lenses in the system could potentially move as well during rotation, affecting the output FOV, the adjustable lens is mounted onto a movable rail, and thus is more susceptible to unwanted movement. Fortunately, to test this, only a few adjustments to the method previously described are needed. The first set of measurements can be done with the same setup, only this time the illumination system will be rotating at a set scanning speed. The FOV will be measured at the target which will be fixed in position with respect to the scanner and the camera, which will not rotate either. While the illumination system rotates, the beam will scan the target at some instance and the camera will record the projection of the beam on the target to measure the FOV. To prevent under sampling of the measurements in the case of a periodic error, the target will be moved to different radial positions with respect to the origin of the beam as shown in Figure 51. Taking the FOV measurements at 0º, 90º, 150º, 225º with respect to the initial position, with the target at equal

distance from the beam origin in each position, will allow us to measure any errors throughout the rotation.

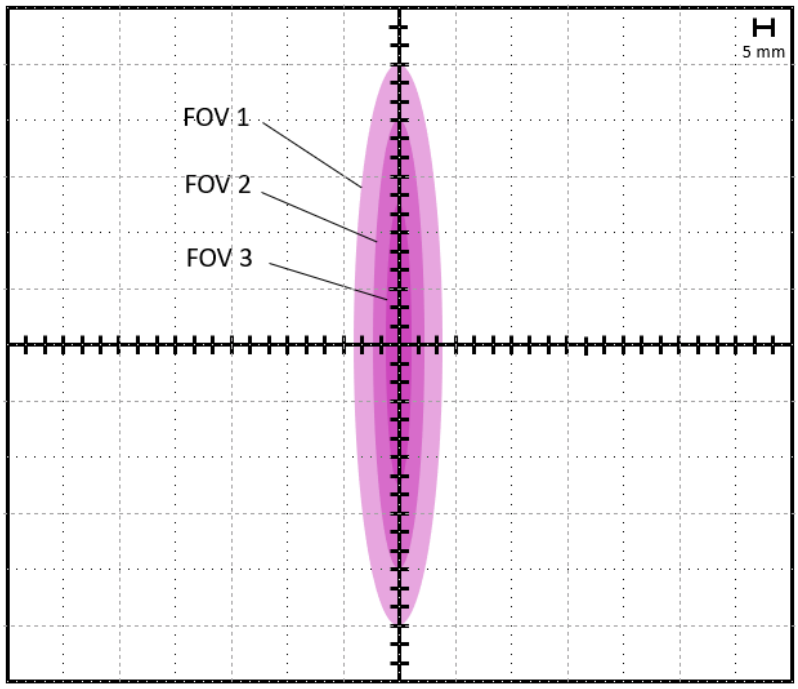


Figure 51: Beam Observed on Grid Target

Tilt and rotation of illumination system: We have thus far discussed some methods for testing the precision, accuracy, and efficiency of our FOV tuning system during scanning and repeated rotation at a fixed speed. Having determined the errors of the FOV tuning mechanism at such fixed scanning frequency, a new constraint is presented for which the maximum scanning speed needs to be adjusted. Therefore, it is important that we test, for a given FOV, the errors introduced by various scanning speed and define a new maximum speed for which said errors remain acceptable. The maximum scanning speed was previously determined by the pulsing rate and the detection range. In order to capture an emitted pulse at the CCD sensor while rotating the system, the angular translation of the sensor during the round-trip time of the pulse must be smaller than the width of the sensor pixel array, otherwise, the part or all of the pulse will be lost. However, because we are discussing light pulses, the round-trip time are incredibly small, even for the maximum detection distance of 50 m, the rotation speed for a radius of 6 cm would have to be around 300 krad/s for the pulse to be lost. This is, of course, not a conceivable speed for our system.

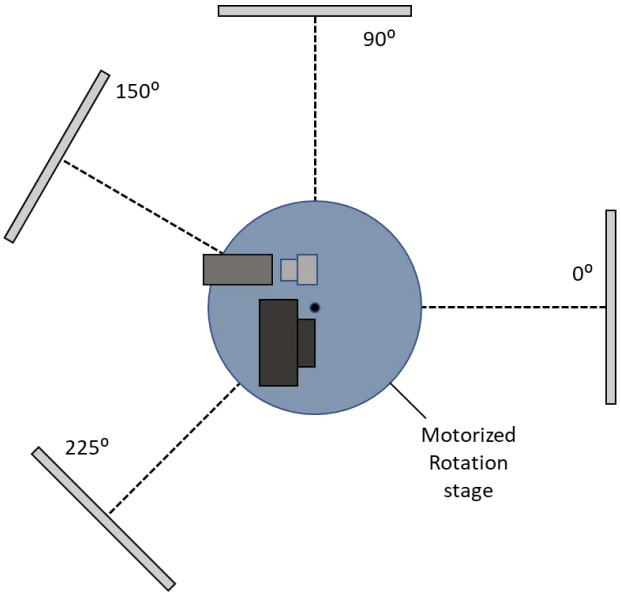


Figure 52: FOV Testing Setup During Scanning

Another parameter that placed a constraint on the rotation speed was the pulsing frequency. The rotation speed and the pulsing frequency determine the lateral resolution during scanning. For example, for a pulsing frequency of 100 kHz at maximum detection range of 50 m, to obtain a lateral resolution of 1 cm the maximum scanning speed allowed is 20 rad/s. While still quite a high scanning rate, this is a more plausible constraint for our system. The purpose of this section, however, is not to discuss the effect of rotation speed on the resolution of the system directly, that is discussed in more detailed in Section 5.2.4.3. Instead, we want to test the effect of rotation speeds on the FOV tuning mechanism and determine if it could represent a more pressing constrain than the system lateral resolution. This can be done once more using a very similar method as those previously described. Since all we are doing is varying the speed of rotation, the same setup represented in Figure 50 can be used. While the illumination system is rotated, the beam projected onto the target at a fixed distance will be detected on the CCD sensor and the FOV can be measured for that position. The target can then be moved to different radial positions as shown in Figure 52 to average the measurements for a given scanning speed. The procedure can then be repeated for different scanning speeds, while monitoring the FOV. An issue with this method is that we cannot monitor any actual movement in the optical system. Any discrepancies in the output FOV are assumed to originate from unwanted movement of the optical elements due to the presence of movable parts; however, a more appropriate method would test the strain on the optical system due to rotation. This is a more challenging task as the closed encasing of the lenses does not allow for easy access or monitoring of the parts inside. For our application, fortunately, is sufficient to attribute the errors in the FOV to the optics and use scanning speeds low enough to maintain the errors at an acceptable level.

The final mechanical process to be tested in our illumination system is the tilting mechanism. Since our scanner is close to the ground, it becomes difficult to scan nearby objects with a horizontal optical axis. For farther targets, however, an optical axis close to horizontal may be preferable to detect objects on the ground. Therefore, it is necessary to implement a tilting mechanism that adjusts the inclination of the illumination and sensing system for different detection ranges. Detailed descriptions of this mechanism are found in Section 5.2.4.2. The purpose of this section

is to outline the steps and methods to test this tilting mechanism. The tilting should work independently of the rotation during scanning and allow for adjustment of the inclination angle to achieve optimal image quality. If during the initial scanning rounds, the image is too dim, after regulating the output optical power the tilting should also be adjusted to improve the illumination distribution onto the target. While testing this optimization process requires testing all integrated systems, the tilting alone can be tested independently prior to integration. The main parameters of interest to be tested are the tilting accuracy, precision or repeatability, and stability during rotation. The first two parameters can be tested simultaneously in a similar fashion as they were tested for the FOV tuning system. With the beam set to a FOV of 5° for a narrow beam, the inclination angle can be calculated through simple trigonometry using the setup shown in Figure 53. In this way, the accuracy of each inclination can be determined. Repeating this process for a randomized input values of inclination angles provides statistical data for the precision of the tilting system.

To test the stability of the tilting mechanism during scanning, the system is rotated while the height of the beam projection on the target is measured. Once again, the setup in Figure 51 can be used for this procedure. After having measured the inclination angles at each radial position of the target, the scanning speed can be varied as well, and the testing process repeated to evaluate the performance of the tilting mechanism at different rotation rates. This will likely represent another constraint for the scanning frequency.

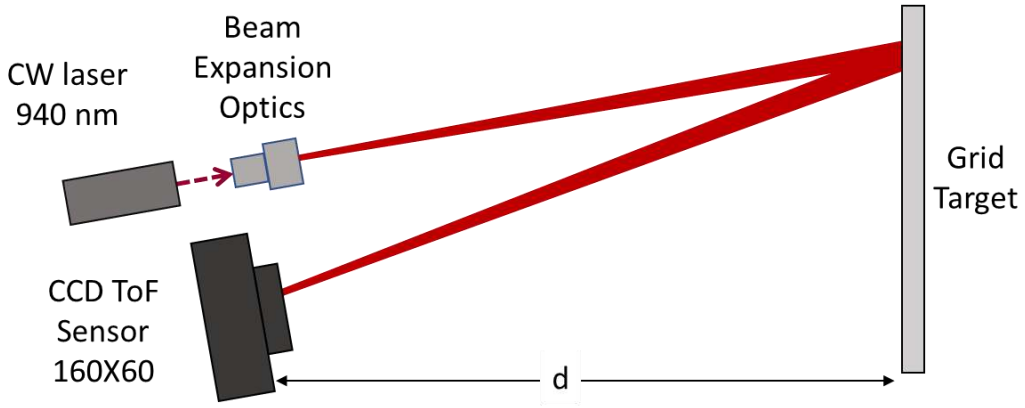


Figure 53: Tilting Accuracy and Precision Testing Setup

Having tested all the aforementioned elements of the illumination system, we should have a better understanding of the performance of this system, providing insight into its performance as part of the integrated system as a whole. The power at different modulation frequencies provides us with the maximum power that can be delivered at different frequencies and for targets at different distances, as the modulation frequency determines the maximum distance for a target which can be detected. Testing the FOV tuning system is crucial for our projects, since one of the highlighted features of our design is its variable FOV. Finally, testing the tilting mechanism allows us to establish the best inclination angles to deliver the most illumination power to the target, as well as to define additional constraints to the maximum scanning speed of our system.

6.2 CCD Sensor Prototype

For prototype testing of the CCD sensor the group has outlined essential tasks that will allow us to make sure that the sensor is working as well as properly set us up for integrating the CCD sensor into the entire system. As described in a previous section 5.2.5.3, for our application we will not be using all of the pins that come on the epc635 CCD sensor. The pins that will be used are described in Table 15. Therefore, one of the first things that we must test for the CCD sensor is that the appropriate pins are connected and that the rest are grounded. To achieve this, we can simply connect the CCD to a power source and use a voltmeter or ammeter to measure the current or voltage at the desired pins. If there is a reading, we know that they have been properly connected. This will also allow us to make sure that the pins that are not used are properly grounded.

The next prototype testing that is crucial for integrating the detection system with the illumination system is for us to measure the experimental results of the CCD sensor responsivity for our operating wavelength. Although the responsivity of the sensor is outlined in the datasheet, we must ensure that the actual value is close to what is described in the datasheet. If the value is slightly lower, we know that the output power of the laser diode must be increased in order to compensate for that loss. From the responsivity tests, we can then test the optical power and appropriate shutter speed that is required for different distance measurements such that optimal resolution is always achieved. Being able to understand the power and detection accuracy of our measurements will allow us to understand the operating conditions that are required for the Mobile LIDAR Scanner to achieve the best axial resolution for the selected scanning range.

To begin integrating the CCD sensor to the collection optics, the group must ensure that the incident optical power onto the CCD sensor is sufficient for detecting the reflected light. Since there will be a power drop in through the lenses and through the NIR bandpass filter. Although that the datasheet for the epc635 CCD sensor says that for the time-of-flight measurements at a wavelength of 940 nm the typical sensitivity for a modulated light source at 12 MHz is 0.8 nW mm^{-2} LSB^{-1}. Sensor sensitivity is inversely related to the optical modulation of the incident light [23]. Since our operating modulation frequency is 3 MHz, then the detected optical irradiance needs to be greater than the one given in the datasheet.

The final testing required to fully integrate the illumination and detection system is to test the modulation frequency of the laser diode with the internal modulator of the CCD sensor. As described in Table 15, the modulator pin of the CCD sensor is pin number 27. As described in the datasheet the CCD sensor is capable of modulating an external laser diode or LED to a modulation frequency of up to 10 MHz. From section 3.3.2, it is described that the modulation frequency that is required for a maximum scanning range of 50 m is 3 MHz, therefore according to the data sheet the CCD sensor is more than capable of modulating at this frequency. We would still need to test the internal modulator to ensure that there is enough power being supplied to the CCD sensor to properly modulate the laser while still being able to operate for sufficient time to complete and render the scanned area.

To integrate the CCD sensor into the entire system, we must test that the response time and shutter speed of the CCD are proportional to the rotational velocity of the entire Mobile LiDAR Scanner. This will be one of the final tests to be done on our CCD sensor to finalize the project. This test is crucial to meeting the requirement specification of a 360˚ FOV, because if the system is being

rotated at a speed faster than the response time and shutter speed of CCD sensor that data that is collected will be incorrect the output image will show overlapping data and missing data points since the Mobile LiDAR Scanner was rotating at a faster rate than the time it takes for light to return to CCD sensor. The integration time of the epc635 CCD sensor is t_{INT} = 1 μs, the read-out time for one row of the epc635 CCD sensor is t_{HSYNC} = 16 μs or 8 μs for either 12-bit or 8-bit respectively. The total time for one 3-dimensional ToF measurement is 7.789 ms or 6.108 ms for either a 12-bit or 8-bit respectively. These time values presented in the datasheet for the epc635 CCD sensor are given for a frequency of 20 MHz.

6.3 Data Processing

Being able to understand the information being processed from the LiDAR machine is a crucial aspect of our project. The LiDAR machine's task is to be able to scan a room or area then return the specific X, Y, and Z points for coordinate plotting. The purpose of this test was to be able to properly collect the correct values to be able to display on the UI screen. To demonstrate this test, the group needed to be able to perform a test on the LiDAR machine to make sure the components are running efficiently. Thereafter, the group understood that any data, whether it be correct or incorrect would rely solely on the software. Obtaining the LAS files and using the functions associated with it, the team performed many tests on gathering the information.

The first test we performed to properly collect the data was we utilized the function `laspy.open()`. The functions assignment was to read the header and VLRs of a file, but not the points inside. While the points are being processed and the file is collecting the bits, the function collect the data and stores it in a file. As mentioned in the section of Laspy, this collects the metadata contained in the header. This is a sense of importance because we would need to make sure the file that's being read is the correct one, and also to make sure the file is opening correctly. Following the function to initially open, we then have to read the incoming data from the file. `laspy.read()` is desired to read all information in the file. This includes the header, VLRs and the point records. The data in the file is crucial for the creation and rendering of our 3D image. If the point in the file become corrupted or the data hasn't been transferred properly, then the 3D image that we plan to render on the UI, will be flawed. Ensuring this particular function is important for development. The last important function required for testing would need to be `lasData.write()`. The las data that's collected would need to be stored in a las file. To achieve this, this function converts las data into an object. While performing tests, writing the data to a las file would be the only way we can read what is getting returned from the LiDAR machine.

Software testing in this sense would be different than that of hardware testing. It's important for the team to test each piece of code to ensure consistency. This can be tested by using prototype software and checking for the validity of algorithm used or functions. To prototype certain section, we will have specific section operate under specified conditions. By no means is this going to be a final product when testing, but to make certain of the persistent data we expect, it would need to be tested on all circumstances. Furthermore, to test specific features in our software design, we can input hard coded elements. Doing this would give us a better understanding on the outputs we may receive. With hard coded elements, we would be able to test different stages of software development, however it would still need to be tested when gathering data from the LiDAR system. Although it may be easy to establish a working environment with values and parameters that were

hard coded, this would still create an environment where we have algorithm, functions, and variable set-in place for when we receive actual data. Prototyping in this way will allow the events to be simulated in an environment that we can control.

To discuss the differences when we're prototyping, we also want to make sure we depict prototyping and testing. In this case of prototyping, this is to make sure when we're going through trial and error, we can input certain values to expect a desired answer. When we test, this would get us a step closer in producing our full outcome. Once we see after inputting hard coded results in our software implementations, when we test with actual data from our LiDAR system, we should achieve a desired result. While we continue to test our system, as we inch closer to a final outcome, the team would need to constantly have a prototype, so we don't interfere with our testing. As we begin to get closer to the outcome, we plan to continuously update our final product, then build upon it until we achieve our desired result.

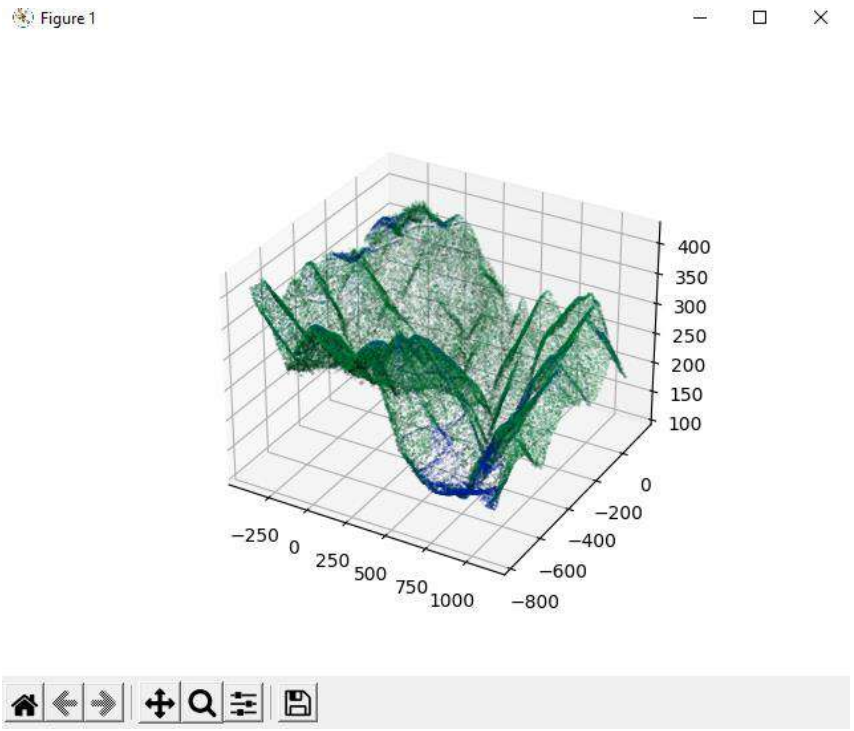


Figure 54: First Test

Figure 54, the image shows the first analysis we ran with some sample LiDAR data of a mountain range. The data we used was just to make sure the program ran correctly, now after receiving an image that properly rendered an image, we can now push forward into our prototype. As mentioned previously, the hard coded data that was used to ensure a working program can now be tested further. The prototype we have is now updated and we will continue to make changes. While we continue to test, the group also needs to understand how we can collect multiple sets of data. This image is shown for just one scan, but if the user would want an additional scan, we would need to handle the requests as so. This image is also just a beginning, following more tests, the image would get updated to resemble that of a scanned room or area.

6.4 Remote Control Prototyping

In order for our device to maneuver around a given plane correctly, and also figuring out how it can be incorporated into our design, we need to guarantee a working remote-control car. This remote-controlled car also has to be tested for the time of operation, speed at which this remote-controlled car can move and even the amount of weight this car can carry without risking performance. To begin testing, we would need to build the piece altogether, the piece the group has decided on using is RC platform 2 from the section we discussed earlier. In the package we received, assembling the car would be quite simple but we would also need to make sure every part of the car is working like it's supposed to. Although this may be an unnecessary task, this is still useful to make sure our design runs as we anticipate.

Figure 55: Unassembled parts

The kit that we received came with many parts for us to create the platform. As I introduced in the previous paragraph, we need to also ensure each component is working properly. The kit comes with two-wheel sprockets, two motors and the wires to connect then, and also the gears used to rotate the sprockets. When developing, the two motors are key into making sure the machine moves. Each motor is to be tested on the number of rotations per second. Both motors would need to have the same rotation speed, when we plan to create the software system for turning, this would be done by slowing one of the motors down to produce an accurate spin. With further testing, we can conclude on the remote-control car to be in final stages of prototyping.

Furthermore, after we complete the assembly of the remote-control car, testing the speed, time of operation and weight capacity is what can be in the next stages of testing. Testing the speed of the car is an important factor because this would give us a better understanding of how long it'll take the user to reach a desired spot, then we can calculate the time it takes for the LiDAR system to scan and return to us the data. The speed also plays a big part for users that may crash. Since this

is a prototype system, the team would need to make sure when we are navigating through the area, crashing is the most of our worries. Unintentional damage to the car could result in many components failing and us buying more. This could result in the amount of money we spend for this project to increase.

The speed could also be directly proportionate to the time at which it would operate. Speed usually requires more power, and with more power, it would take time of operating lower overall. Earlier we discussed our system to run for at least 30 minutes. This time frame should be more than enough time to drive the remote-controlled car to a location and wait for the system to make a scan of the room. The time it takes for the scan to upload to the UI and render its data as a 3D image, we should still have plenty of time to gather more scans if needed to. This can be testing by using the most power the car can operate at, then recording the time it has been operated for. Weight could also play a big role in the performance of our device. If the maximum weight carried by this device is exceeded, the performance could be reduced significantly. To start, we would first test to see if the weight of our LiDAR system is compatible for the RC Platform we intend to use. We will do this by getting weight that's equal to the weight of the LiDAR system, then running the operation for as long as we can. The amount of display time should be at least 30 minutes. If it shows to operate at a time less than that, then different adjustments would be made.

Figure 56: Motors used for Speed of device

As we continue to test our product, we will continue to update our prototype. The prototype would be the last successful attempt of running tests. After running multiples tests, the prototype would give us a cushion on the last operations we handle successfully. In terms of dealing with the RC

car alone, this is running separately from the rest of the system, therefore conducting changes to the RC car wouldn't affect the whole prototype. We felt like this should be implemented separately because we would have more control of the hardware side of things for the car itself and have more flexibility to changes that happen on the LiDAR system.

6.5 Data Transmission

In this section we present the methods that the group is using and will use to test the hardware components of the Mobile LiDAR Scanner.

6.5.1 Bluetooth module test

Establishing a connection between the computer and the microcontroller is needed for the transfer of incoming data from the LiDAR system to the Python application that generates the 3D renderings of the LiDAR scans. The purpose of this test was to familiarize ourselves with connecting the MSP430FR6989 microcontroller to the computer using Bluetooth during development. Using the Bluetooth-enabled smart device (computer or phone), we sent ASCII characters using UART to the microcontroller. These ASCII characters are interpreted by the microcontroller to turn on the on-board green LED.

The Bluetooth module we used has six pins to interface with the microcontroller. Going down the module the six pins are named STATE, RX, TX, GND, VCC, and EN. For this hardware test, we only need to use four of the pins since the Bluetooth module will only be in slave mode, thus the STATE and EN pins are not needed. Being in slave mode means that the HC-05 module was only to be used to send bytes of information to the microcontroller.

The HC-05 module interprets 3.3 V as Logic 1 and 0 V as Logic 0. Logic 1 on the MSP430FR6989 is 5 V, so a voltage divider is needed to reduce the voltage for the module. Following an example in the module's user manual, the RX pin on the module is connected to the output of the voltage divider. The divider was made of a 1 kΩ resistor and a 2 kΩ resistor on a solderless breadboard.

The code for the experiment first defines constants for the green LED, the UART flags, and the UART buffers. The constant definitions and their meanings are in the Table 19 below:

Table 19: Constant definitions for Bluetooth hardware test code

Bit mask/Flag/Register	Defined name	Description
BIT7	Green LED	The on-board green LED is at P9.7; defined bit mask to access green LED easily
UCA0IFG	FLAGS	A register that contains transmit and receive flags
UCRXIFG	RXFLAG	A flag that tells if data has been received; 0: no new data; 1: byte received
UCTXIFG	TXFLAG	A flag that tells if data in currently being transmitted; 0: transmission in progress; 1: ready to transmit
UCA0TXBUF	UCA0TXBUF	A register acting as a transmit buffer that holds the byte that will be transmitted
UCA0RXBUF	UCA0RXBUF	A register acting as a receiver buffer that holds the received byte

The transmission model used for communication is UART. On both the microcontroller and the module, a transmitter (TX) and receiver (RX) is needed to communicate. The MSP430FR6989 microcontroller is embedded into an evaluation module called the LaunchPad Development Kit. It offers two eUSCI modules: eUSCI_A0 and eUSCI_A1. eUSCI_A1 (or eUSCI module #1 Channel A) connects to the backchannel UART. This module won't be useful since it directly communicates to the LaunchPad Board over the USB connection. Instead, we have to use eUSCI_A0 (or eUSCI module #0 Channel A) which offers a separate UART channel connected to the 20-pin BoosterPack connector. Using the MSP430FR6989's data sheet, ports P4.2 and P4.3 were used to act as the transmitter and receiver signals respectively. These UART signals on the microcontroller will be referred to as TXD and RXD respectively. The block diagram shown in Figure 56 below depicts this system:

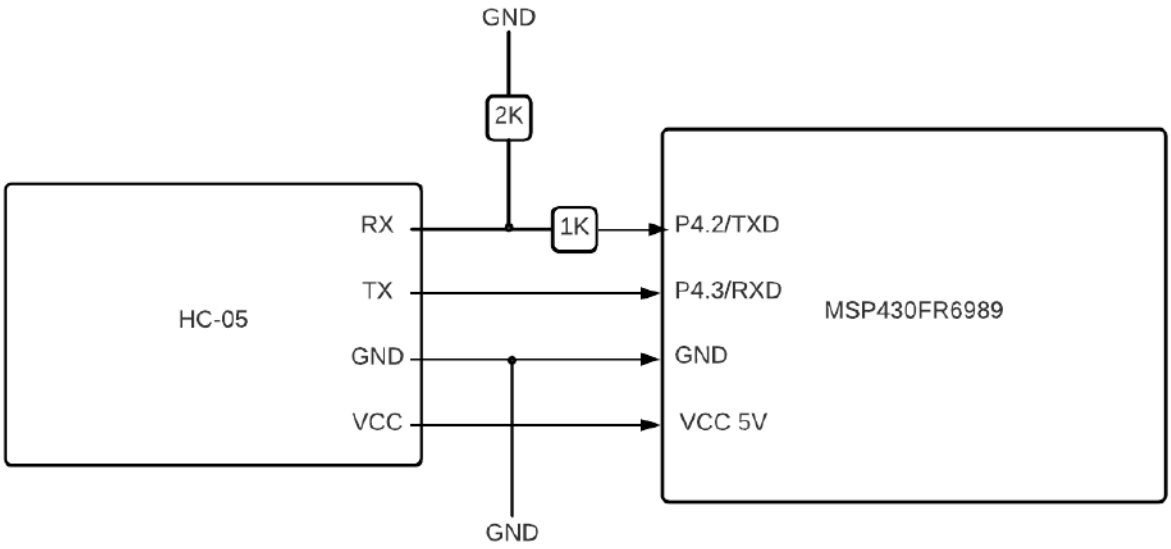


Figure 57: Bluetooth hardware test block diagram

The code then diverts the pins on P4.2 and P4.3 to UART functionality [24]. It then initializes the UART settings using the UART configuration registers. In this test, we use the most popular UART configuration (9600 baud, 8-bit data, LSB first, no parity bits, 1 stop bit, no flow control). The HC-05 module itself is configured to 9600 baud by default. Functions to write and read characters are written using the constants we defined earlier. After initializing the UART and setting up the needed functions, the program enters an infinite loop.

In this loop, the microcontroller checks the RXFLAG to see if it has received any bytes. If the flag is raised, whatever byte was currently in the receive buffer is captured. If the flag is not raised, it simply returns the null character "\0". If the captured data byte from the buffer is read as "1", the green LED is switched on. If the captured data byte from the buffer is read as "2", the LED is turned off. If no data is captured, the code checks the RXFLAG again and the loop repeats.

In order to transmit data from a smart device to the microcontroller, a Bluetooth terminal application was used to transmit ASCII characters. When a "1" or "2" was sent to the microcontroller, the green LED turned on or off respectively. Toggling the LED indicated that the byte was successfully transmitted and received using UART and Bluetooth and interpreted by the microcontroller.

6.5.2 Quality of data transmission

In addition to testing the Bluetooth module responsible for communication, our team also wanted to test the quality of transmission. The data will have to go through three major transmissions: (1) CCD sensor to microcontroller, (2) microcontroller to Bluetooth module, and (3) Bluetooth module to computer. The CCD sensor is the component of the LiDAR system that will supply the data from the scan. The CCD sensor needs to be able to send back data to the microcontroller. The data is transmitted back to the computer wirelessly via the Bluetooth module. There will be a large

amount to data transmitted between the CCD sensor and the computer. It is important for us to recognize the limitations of our product and see if they meet the requirement specifications, we had previously established in Table 2 These different aspects of quality are discussed below.

Max distance: The first aspect needed to be tested is max distance that data can be sent. Bluetooth is a provides wireless data transmission between two devices for a rather short range. As the distance between the two devices grow, the latency of the data (the time it takes for data to be stored or received) gets larger. If the data takes to be transferred by the Bluetooth module, it creates a bottleneck effect. This may result in loss of data and other errors. To test this, we plan to use the previous Bluetooth module test and use different distances between the Bluetooth module and a Bluetooth-enabled device, ideally another computer. Since, the Bluetooth terminal reads out what is received by the Bluetooth module, we can use a separate timer to time how long it takes for data to be received. We will test different distances (10 m, 20 m, …etc.) and record the time it takes for the LED to light up on the microcontroller, indicating that the data from the other device has been received.

Max speed: The next aspect that needs to be assessed is the maximum speed at which the data can be sent. LiDAR data is calculated by the CCD sensor at very high speeds. It is imperative to ensure that the data is being transmitted to the microcontroller quickly. The Bluetooth module as stated previously runs at frequency of 2.4 GHz, allotting us some leeway for wireless transmission. We are slightly limited to receiving data at a clock rate of 16 MHz, since the MSP430FR6989 is limited to that clock rate. This puts a slight cap on reading the data from the CCD sensor. The CCD sensor can configure its own clock frequency from 10 to 80 MHz. We may also have account for the fact that two different data transmission protocols are being used. Between the CCD sensor and the MSP430FR6989, the data is captured on the sensor, calculated to find distance using ToF, and sent out in parallel as 8 bits at every clock cycle. Between the MSP430FR6989 and the HC-05 module, UART is used, meaning data is sent out one bit at a time. Our team will have to play around with different frequency settings for the CCD sensor, MSP430FR6989, and the HC-05 to see which combination of configurations gives the fastest speed for data transmission.

Bit-error rate: With any sort of data transmission, there are bound to be errors. In the context of our project, these errors manifest as missing points or misplaced points on the final point-cloud representation. To test this, we plan to connect a Bluetooth-enabled computer to the HC-05 module. Then, we will create a program that will send large pieces of data to the HC-05. Then, the microcontroller will simply print that it receives onto a terminal on Code Composer Studio. Finally, we can compare the received data to the sent data and count the errors. This can be done a number of times and we will average the error rate.

6.5.3 Data transmission in different environments

The first application that our team envisioned for this robot was to be able to make 3D point-cloud scans of enclosed areas without the user needing to be inside of the room. So, our scanner will have to transmit data between walls and other obstacles. To test for a single wall, we will first need a control run for comparison. So, we measure how long it takes to scan a room with the computer in the same room. Next, we will place the scanner in the middle room and have the user's computer be outside of the room. Then, we will measure the length of time it takes to complete the scan.

Our team also considered making scans in open areas. In open areas, there is a lot of open air that may or may not show up in a 3D point cloud. So, our team will have to deal with the max distances not being read properly or even resulting in an error. Even with max distances being recorded, we must ensure that the LiDAR scanner is still able to show the environment close to it.

6.5.4 Parallel transmission

While the LiDAR data transmission is crucial to our project, the scanner must also be able to handle transmission coming in from the computer application. Some transmissions from the computer may include a start/stop LiDAR system to start a scan or cancel a current scan. Additionally, our team wanted to implement two different configurations for short-distance and long-distance scans, changing the FOV accordingly.

6.6 LiDAR Prototype test

When testing the LiDAR component of this project, we plan to conduct various measurements in different environments to understand the working conditions of our product. Initial testing will be done in a dark room that has no external illumination source such that the only light that is scanning the area is the light from the illumination system. This experiment will allow us to understand how the LiDAR will detect light in an ideal condition with no noise. Experiments under these conditions will give us a sense of the working conditions of our sensor and how it collects light. This will allow us to have a baseline for collecting optics.

The following test will be done in a room whose illumination source is light generated from light bulbs. Introducing a broadband light source will allow us to measure the effectivity of the filter within the collecting optics. White light from lightbulbs is broadband within the visible light spectrum which will introduce noise to the measurements. If the sensor is still able to easily detect the reflected light, then we can move to the final prototype test.

The final prototype test for the LiDAR is in an outdoor environment, where the illumination source is the sun. The sun is the largest broadband source that exists because it emits light outside of the visible spectrum. This means that the noise that is being introduced into the sensor could be light of the same wavelength as the illumination source. Conducting experiments under this condition will allow us to know what is the power level that will distinguish a reflection from the LiDAR scan from the noise.

6.7 PCB Vendor and Assembly

After testing all components to create a working prototype, we now need to look into design. Although we may have created a system that works efficiently, we would need to look closer into developing our own PCB. Printed Circuit Boards are a means of shrinking components that we wired from a breadboard. Upon creation, we can develop all input and output components for easy installation. Switching from a breadboard to a PCB has many pros. The connections are permanent, and also offer greater current carrying capacity. We decided to create our own PCB, this would allow us to have more control on specific components needed in our design. This would also add onto making our system more compact, it could be downsized because all unnecessary components would be removed. To add into producing our PCB, we also have to consider the software we plan

to use when creating our PCB, the company that would print it, the best layout of how the board would be displayed as well as mounting the components. Further into development, we would examine these attributes for designing our final product.

6.7.1 Eagle

Eagle is a software that is used to create printed circuit boards also known as PCBs. It also adds a sense of familiarity since this was the application used for Junior Design. To add on, this is one of the most popular programs while also being easy to use. The app offers many utilities to help ensure a proper established PCB. One would begin designing all components needed, while also collecting the footprint associated with it. Then when all components are wired on the sheet, one could transfer it as a PCB design to rearrange components in desired locations. Upon completion, the result would show a 3D image of our printed circuit board, this can then be processed for us to use in our design. Figure 58 shows an image of how Eagle software is handled when creating a PCB.

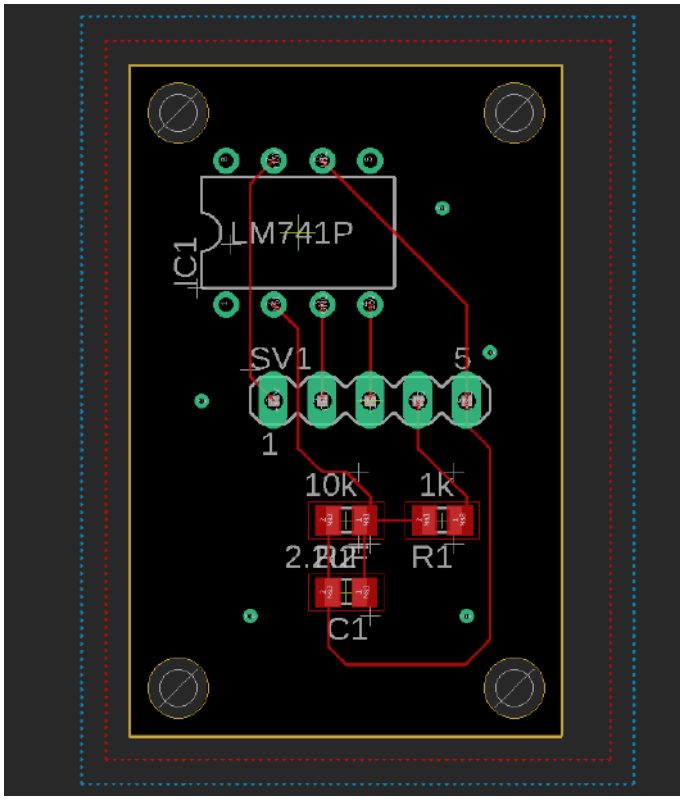


Figure 58: Junior Design PCB

Typically, the process begins with searching for each component we intend to use in our setup. This setup may include resistors, capacitors, pins, switches etc. These components would then need to be tracked by its digital footprint. Sites like Mouser and a branch from Texas Instrument are platforms used to track them. Once we collect the digital footprint, the components are laid on the sheet for connections. The team would need to establish all connections on the board layout before we transfer the setting on a PCB layout board. Configuring errors and following circuit protocols, we would develop a working product, that would have precise current and voltage

regulation throughout the board. Moreover, the last procedure would be to make sure wires aren't interrupting each other. Airwires are key attributes to making sure the design is complete. The placement of wires is necessary for board development. Upon completion, our board would then be sent for a company to produce it.

6.7.2 KiCad

KiCad is another software we could use to develop our board, although Eagle offer familiarity, we can also look into other options in terms of establishing our board. This software offers up to 32 layers of PCB layout while still giving us a 3D image working on it. It provides a better view when we plan on implementing it. Figure 59 [25] shows how the 3D view for better depiction of components. Like Eagle, KiCad features an integrated environment for the PCB layout, schematic capture while also being equipped with a file view. The two software applications can be used to help the group create a PCB that's unique to our project.

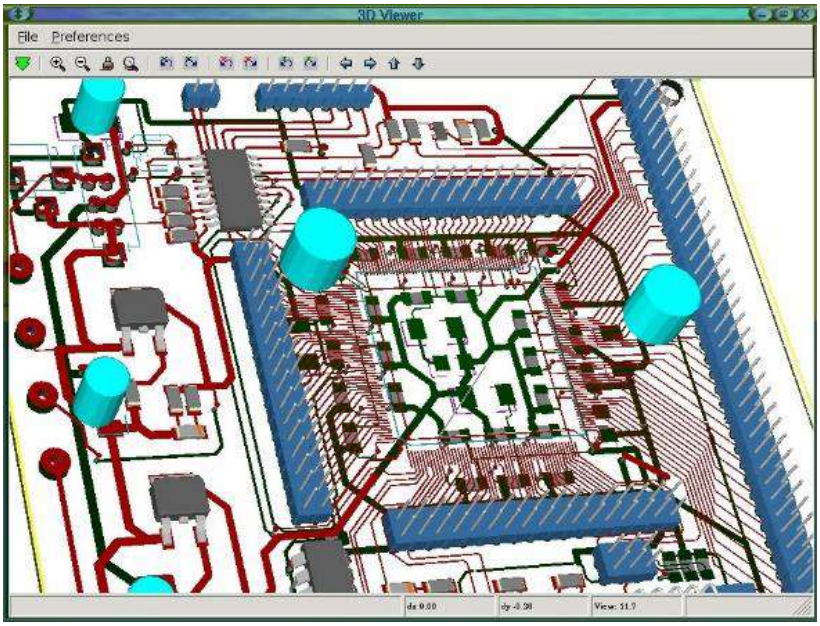


Figure 59: Image from Wikimedia Commons [25]

6.7.3 Test/Development PCBs

As we have been developing our scanner, we need to configure the different portions of our system. The CCD sensor are made more accessible with the addition of the card carrier and 60-pin connector, the pins on the connector are too small for us to attach jumper wire and connect to a solderless breadboard. Thus, we will need to create test PCB used to develop and test code that will configure the CCD sensor to what we need for the project, configuring the which clock rate to use and making sure that microcontroller is able to read the changes in the data before it is overwritten. The pins require are discussed in Section 5.2.5.3 in detail, where we only need to use pins 1, 2, 4-9, 10-13, 18, 19, 29. Any other pins will just be grounded to the final PCB board. For this test PCB, we will just use three 6-pin headers to breakout the pins from the CCD sensor. Figure 60 shows a simple representation of what the test PCB for the CCD sensor is planned to look like.

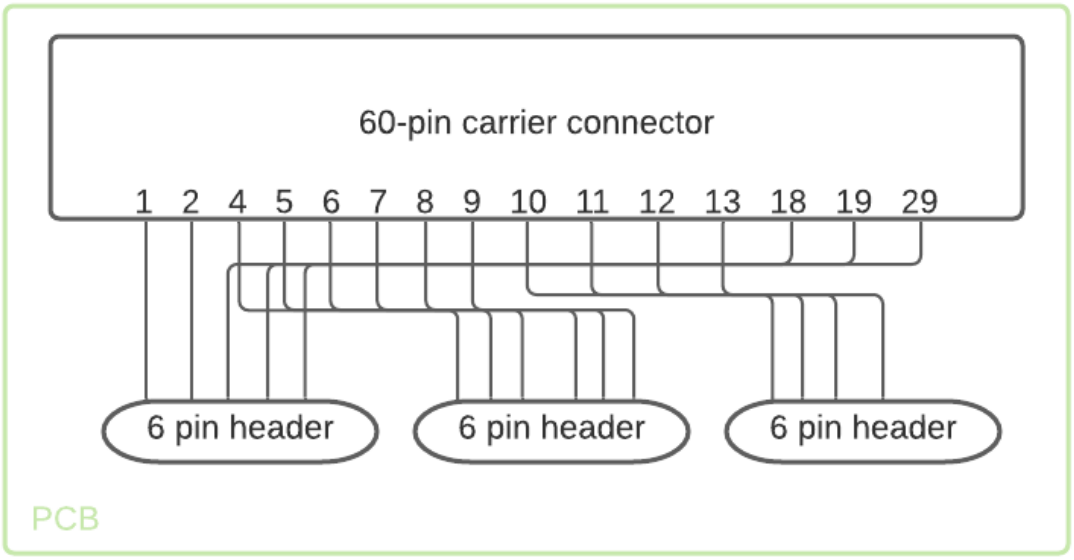


Figure 60: Block diagram of test PCB for CCD sensor

When the settings and pin configuration for our MSP430FR6989 are finally set, we will need to flash these configurations onto a raw MSP430FR6989 before it goes onto our final PCB. In the case of the establishing the Bluetooth communications, our microcontroller will need P4.2 and P4.3 diverted to UART transmitter and receiver respectively. Configuring the CCD sensor requires I2C, thus a SCLK and SDA will be needed. In this case we must use P4.0 and P4.1 to act as our serial data and serial clock respectively.

6.7.4 60-pin breakout PCB design

The epc635 is a chip needed to establish a resolution of 160 x 160 pixels. This chip is used to have complete control of the logic to operate the device. Depending on the illumination power and the specified optical design, adding a few additional components could create a 3D camera. But to be able to use this chip and get the values associated with the 60-pin connector. The group needed to establish a way of obtaining all 60 pins. The use of a breadboard with wires was a bit too robust to initiate a strong connection. The team investigated creating our own PCB board that had a female connection for the 60-pin epc635.

The creation of the PCB board was quite simple, since the epc635 had 60 pins, we first needed to find a casing for the 60 pins to be connected to. After searching multiple footprints that were compatible with this imager, we decided to use DF12-3.0-60. This was a standard 60 pin connector that was needed for the insertion of our imager. Alongside this 60-pin connector, we needed to possess areas to link our male wires for the connection to our MSP430FR6989. The PCB consisted of three 20 header pins. The header pins we used were JBK20. Originally, we wanted to have one header pin that had 60 pins on it, but the configuration of wires to establish secure connection through the PCB was nearly impossible. This is what the group decided to look into after knowing the fact. Although we use three 20 count header pins, after trial and error of figuring out the best

layout of the board, this was the most significant and seems to be deemed as the best fit for our case.

The layout of the PCB design for this particular board was simple. We wanted the 60-pin connector to be in the middle, then on opposite ends of the board, we placed each of the 20 headers. The connection from the 60 pins to each header pin was connected as follows. For the first 20 header pin labeled as J1. We connected the number of the pin to the corresponding header pin. The numbers should align. The second header pins we continued the trend and labeled each of the remaining pins. The last 20 header pins labeled J3 are connected in resemblance to the others. Starting from pin 60, we connected the header pin number 20 then connected the follow pins in descending order. The labeling of the pins are all in the manual we created for the PCB, this would help the group determine which header pins to use to produce our product.

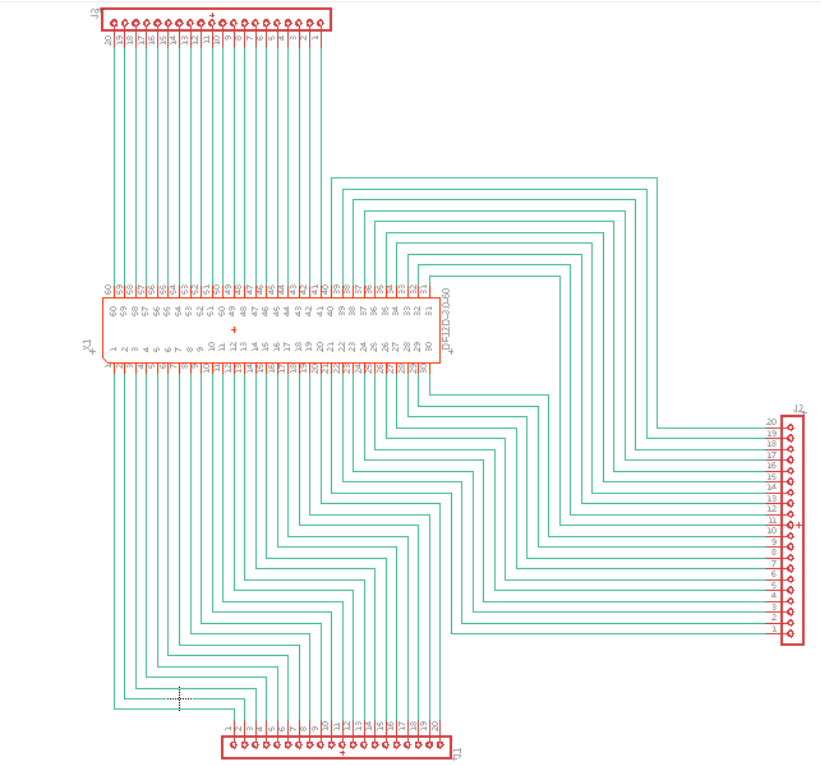


Figure 61: Schematic design of 60-pin breakout PCB

As seen in Figure 61, this was the general layout the team decided. It may be confusing to decipher upon first look, but the layout of each component and the wires are completely necessary to make the connection on the board design simpler. The software used to create this was on Eagle. Eagle made it easier to find part that we need associated with our design, it also made it easy to transfer this to a board design. The board we design that shows the connection of all the wire above and below the PCB is shown below. The image may be slightly blurry, but this was just to show the connections and the general layout of what we created.

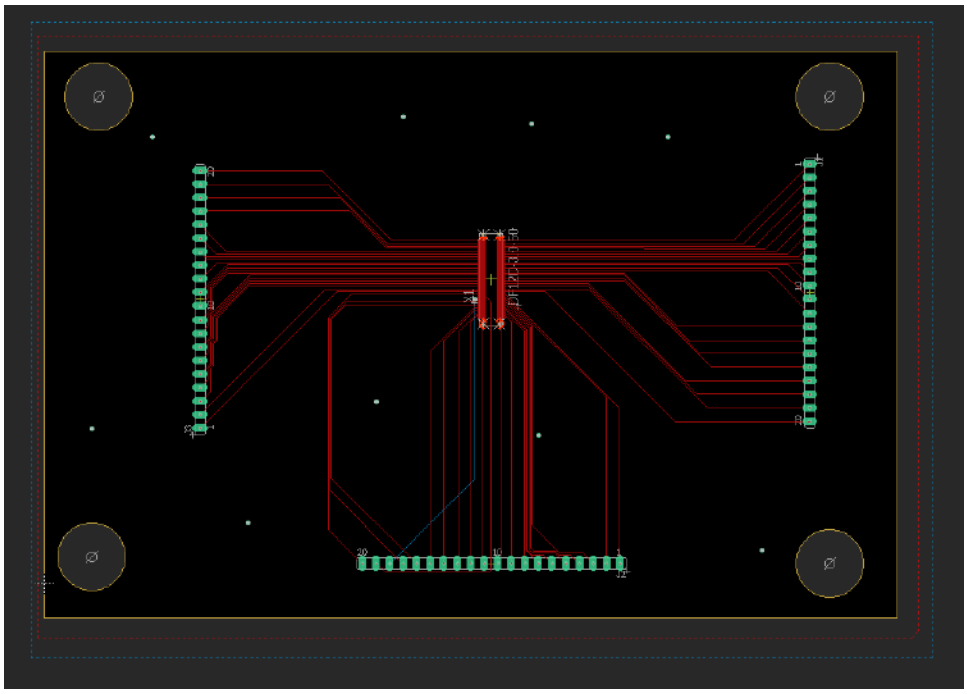


Figure 62: 60-pin breakout board design

Knowing since we finally have a final PCB Board, the obstacle we seem to have come across is the manufacturing of our board. There are many different manufacturers to develop our boards. Some of the sites we've come across and have been labeled in our senior design sites are PCB-Pool, OSJ Park, Express PCB, and 4PCB. These different locations we can use to implement our piece are a few top links we could use for our development. The one we seem to have a liking of is the first name. PCB-Pool is a website used to create PCB boards that we've created. To be able to develop a board on their site, it seems all we would need to do is send the .pcb and .sch file. After giving a thorough review of the contents in our file, the board should be able to be made and shipped to our location. Knowing this, the development of the many boards we plan on making should make it easier for us to produce a working machine.

6.7.5 PCB Design for Peripheral Components

Our Mobile LiDAR scanner is a machine that requires many different components to be connected. Components such as resisters, capacitors, the CCD sensor, our microcontroller etcetera. To be able to properly connect each component with an elite fashion, the group decided on using a PCB. The PCB we make it simpler for us to have secure connections throughout the device. As mentioned in previous sections, the RC Platform 2 has a bigger platform for us to connect multiple pieces to, with this attribute, we could have our design connected with breadboards in wires for our prototype version. But in our final design, we plan on designing a PCB that'll perform the same as the breadboard in our prototype. Using a PCB would have a far greater benefit than that of a breadboard.

The components we plan on using in our second PCB board design are: connections from the servos on the motors and the RC platform, establish a connection to our microcontroller to the Bluetooth module, which is what we plan to use to communicate with our machine, a relationship

with the FOV tuning mechanism and the tilting mechanism also plays a part in our PCB design as well. Having the ability to be able to set specific constraints for our mechanics would provide us with more functionalities of our machine. We also plan to have a secure connection to The LiDAR scanner from a separate microcontroller. Upon implementation of our prototype, we plan on adding onto the second PCB board design layout. The digital footprints for all the associated components would be listed and found on either Mouser or Digikey. The two sites are used to locate digital footprints, then we could make a connection on the software application Eagle. Eagle would provide us with a platform to construct our view on how we would like each element to be placed.

Since this design of the PCB would be ever changing, the schematic for this design isn't established yet. The Divide and Conquer for Senior Design II would give a better layout on how we felt we needed to construct our piece. So far, the structure we came up with for the final PCB design is listed in the next section. This gives us a better representation of how we plan on building things on our breadboard. The connections being made would make the information flow from one side of the system to the other. Up to this point, the group has made advancements in the communication aspect of our design. The next step is to build our system on the RC Platform, this would give us a better understanding of how we plan to operate the system. It'll give us insight into the many possibilities we plan on using and plans that we need to withhold.

6.7.6 Final PCB design

For the final product, our final PCB will act as the bridge between the optics and the accompanying computer application.

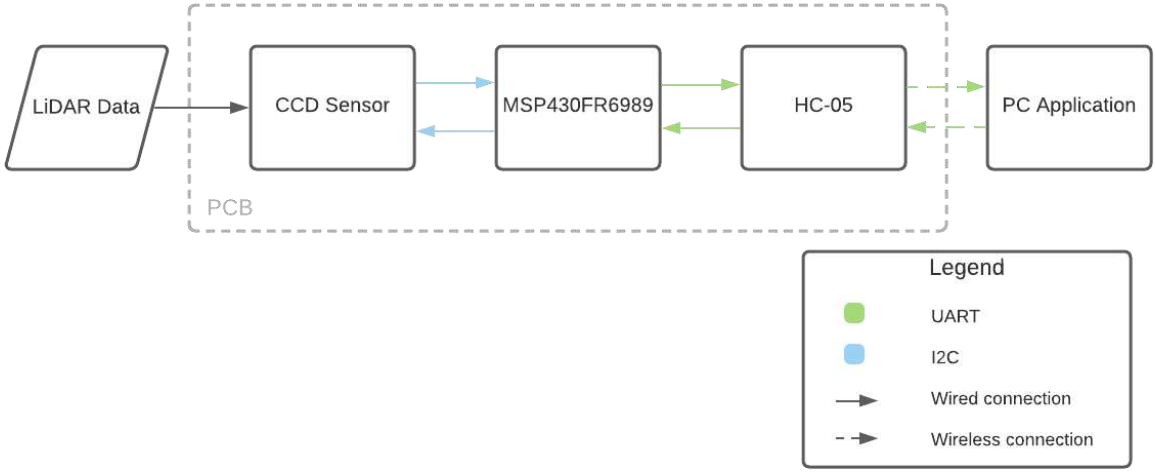


Figure 63: High-level function of final PCB

Figure 63 shows how the PCB will fit into the final design. The final PCB will consist of three main components: the CCD sensor, the MSP430 microcontroller, and the HC-05 Bluetooth module.

Communications between the microcontroller, the HC-05, and PC application are all handled using UART. Wire connections are needed to connect the transmitter and receiver of the microcontroller

to the receiver and the transmitter of the Bluetooth module respectively. Recall from the Bluetooth module test in 6.5.1 that Logic 1 on the HC-05 is seen as 3.3 V. Thus, a voltage divider will have to connected to the receiver of the HC-05, possibly using a 1 kΩ resistor and 2 kΩ resistor embedded on the PCB itself.

Communications between the CCD sensor and the microcontroller use I2C. However, it should be reminded that these communications are strictly to configure the CCD sensor upon boot. The data is instead sent out in parallel on data pins. Serial clock and serial data will have to be connected between the CCD sensor and the microcontroller to allow data transmission using I2C. Configurations for the CCD sensor will be encoded into the program that will be flashed onto the MSP430FR6989.

On the custom PCB, it will hold the CCD sensor's card carrier, the card edge connector, the female connectors that connect to the MSP430FR6989 LaunchPad. It should be noted that the custom PCB will be mounted on top of the MSP430FR6989 LaunchPad Development Kit using female header pins. The pins of the MSP430FR6989 chip are already broken out into male header pins on the LaunchPad board. This would allow use to avoid the cost of having to buy a lone MSP430FR6989 chip, building a separate PCB to breakout the pins in order to flash a program onto the chip. It will make final assembly much easier since we would just have to connect the header pins of the LaunchPad to the female connectors on the custom PCB.

6.8 Locomotion

The LiDAR scanner that our team plans to build is mobile. The RC platform that we plan to use, mentioned in Section 5.2.6.2, has uses a continuous track as its primary way of locomotion. This platform is powered by two motors, two driving wheels, and 10 driven wheels, each moving their respective continuous track. To control the platform, we need to control the speed and the direction in which the two continuous tracks are going. To move forward, both tracks will have to increase their speed forward. To move backward, both tracks will increase their speed in the opposite direction. To turn left, the left track will have to move forward at a quick speed while the right track will have to run backwards. This is vice versa for the platform to turn right. The first thing to test is what speeds are needed to be set for different movements. This will take a bit of trial and error in order to accomplish this.

Our team will need to figure out the top and lowest speed of the platform on different terrains, since our LiDAR scanner will be able to be used both inside and outside. Some of the different terrains that our team had in mind for the outside included asphalt, dirt, sand, and grass. Since the RC platform uses continuous tracks, we are hoping that the LiDAR scanner will be able to perform well. It is also important to consider the terrain of inside environments as well. This includes carpet and linoleum flooring. For each terrain that we wish to test, we will allow the RC platform to go forward a set distance (10 feet or so). If the RC platform is able to reach the distance without getting stuck or tipping over, we can increase the speed and try again. The data from this test will allow us to be mindful of the top speeds for each of the terrains while in development.

Operational lifetime is a very important aspect of locomotion that needs to be considered. Our team wanted to observe the operational lifetime of the mobile LiDAR scanner when idle and active.

To the test the idle operation lifetime, our team plans to simply record the time it takes for the battery to run out when the scanner is idle. Active operational lifetime is slightly different, where we have to consider when the scanner is only moving and when the scanner is only scanning. To test the active operational lifetime for movement, we can let the scanner move back and forth and record the time it takes until the battery runs out. Similarly, we can let the scanner scan an area until the battery runs out and record the time it takes. In our requirement specifications, our team had stated that we aimed for an operational lifetime of 30 minutes for both movement and scanning. To test this, our team plans to set up a test area where the scanner will move to a designated position and then begin a scan. Once the scan is complete, we record the time.

6.9 Calibration

While working on the hardware and software aspects of our design, we need to make sure we include boundaries. These boundaries are going to be set limits or in other words, we would need to calibrate our design. Using a specified point of interest, the team could create a way for the machine to set certain deviations from the referenced calibration. For example, the lasers we use could be deemed as harmful if we don’t set them to a particular frequency, we would also need to make sure we adjust the FOV of the laser, so it does not focus on one point. We would want our laser to spread across in a range. This would allow us to get more precise readings from the LiDAR system.

Another calibration technique we could use in our design could be, making certain the data that is received from the LiDAR system is in tune with how the software collects the data. Certain variable, metadata and even algorithms we establish in the creation of our code solely dependent on how we expect the data to be returned. Any off data could ruin the resulting 3D point-cloud representation. When plotting the data in the application itself, it is important to establish a point of origin in the point-cloud representation itself. This will act as a placeholder for the LiDAR scanner system in the world of the point-cloud representation. Based on this point of origin, the data taken from the LiDAR scanning system will be measured from here. The final point-cloud scan will be calibrated to this single spot and the data will not be shifted.

Environment factors (such as heat, humidity, evaporation etc.) can all have a negative effect on the readings we receive from the system. Depending on the type of risk we face, adding calibration techniques to parts of our design could reduce errors. These calibration techniques would have check conditions that would adjust the calibration, and it would be decided on each user. Environmental concerns could be wide ranging, so it's important for us to create presets, then add calibration techniques that would gear more towards the user and setting. These calibration techniques are also important because it can help the user get a better understanding on how each aspect is used, when determining speed, it could give an output of how long the system is set to last. If the user sets the intensity of the scanner, that could also have a negative effect on the duration the device would operate. These functions would help the user have a better understanding of the product.

7.0 Administrative Content

In this section we will describe the two administrative tasks that the group has set to follow throughout the completion of the project. The first that is discussed is an outline for both semesters that the group put together to make sure that the project is completed on time. The second is the tentative budget for the selected products in order to meet the budget constraint that the group has set.

7.1 Milestone Discussion

Table 20.a: Project milestones for Fall 2021 semester

Senior Design I - Fall 2021		
Milestone	Begin	End
Team formation and project ideas	08/23/21	09/03/21
Project selection and role assignment	09/03/21	09/07/21
Project proposal - Divide and Conquer	09/07/21	10/01/22
Research	10/01/21	11/01//21
Buying parts	09/19/21	01/20/22
Optical design simulation	11/01/21	12/02/21
Designing and building optical system	10/15/21	12/02/21
Proof-of-concept test	11/01/21	12/02/21
Initial documentation	09/27/21	12/07/21

Table 20.a above, shows a breakdown of the project as milestones for the first semester. The fall semester mostly consists of technology investigation and system designs. During the first few weeks, the team was formed, and the project was selected. As the weeks progressed and through weekly meetings, the group was able to create a more well-defined problem. Having a well-defined problem allowed us to begin our technology investigation for the sections that each member of the group took charge on, Figure 8. This schedule has allowed us to stay on task with the writings that have to be in this class as well as the major design milestones that we look to be at.

Table 20.b is the tentative milestone schedule that the group has created. Since we have not begun yet with the building of the Mobile LiDAR Scanner, we have selected vague dates on where we believe we should be. For the final documentation we do not have an end date as that will be specified to us in senior design II. The group has set to begin prototype building on January 10th. This is the first day of the spring semester and the first day back from the holidays. The group

plans to have a finished prototype 1 no later than February 10th. This allows us to begin testing and optimizing the design for the prototype no later than March 3rd. Testing will be done throughout since through trial and error we will get ideas on how to better optimize the prototype. Testing will be done until April 1st, and the finishing touches on the final prototype will be done until April 10th. The rest of the time will be used to prepare for the committee presentation and real-life demo.

Table 20.b: Project milestones for Spring 2022 semester

Senior Design II - Spring 2022		
Milestone	Begin	End
Final Documentation	01/10/22	TBA
Prototype Building	01/10/22	03/01/22
Prototype 1 finish	01/10/22	02/10/22
Testing and optimization	02/10/22	04/01/22
Finishing touches on project	04/02/22	04/10/22
Final Presentation	TBA	TBA

7.2 Budget and Finance Discussion

This section will describe the projected budget for the group. Since the project is self-funded, the budget is a constraint that is present for the group as described in Table 21. The majority of the budget has been allocated to the optical and photonic components of the Mobile LiDAR Scanner. Table 21 breaks down the projected budget that the group hopes to meet. It should also be noted that the budget shown here only accounts for the material used in the final product and does not include any duplicate parts used for development. Table 18 has been separated into the two major components of the Mobile LiDAR Scanner, the optical components, and the electrical components.

The most expensive component of the entire project is the collection optics for the detection system. This system contains a large number of wavelength-specific lenses, that will help us correct some of the aberrations that have been previously explained and losses of power at our operating wavelength. Another factor for contributing to a large expense in the collection optics is that we need to include a lot of optical components in order to properly guide the reflected light from the object to the sensor. The second most expensive component of the Mobile LiDAR Scanner is the illumination optics, the highest expense of this subsystem is the cylindrical lens that needs to be used in order to achieve the fan of light that we require to scan the area or room. The other expensive components that are seen in the Mobile LiDAR Scanner are the servo motors to create the rotation stage and the bandpass filter required. The group needs to invest in high-quality servo motors to be able to rotate the LiDAR to get the desired scan. The purchased motors must

also be able to produce the required torque for a time that is close to or greater than the operational lifetime of the Mobile LiDAR Scanner.

The total projected budget for the Mobile LiDAR Scanner is now $1144.00. From updates to our original design and the group has had to update the project budget constrain of $900 to $1200. Although the team was set on creating this project in under $1000, this increased budget will better the performance of the Mobile LiDAR Scanner.

Table 21: Breakdown of project budget

	Item	Price/Unit	Number of Units
Optics	CCD Sensor	$73.00	1
	940 nm Laser diode	$70.00	1
	Collector Optics	$200.00	1
	Illumination Optics	$180.00	1
	Bandpass filter	$100.00	1
	Lidar Rotation Stage	$160.00	1
	Lidar Tilting Motors	$25.00	2
	Optics Subtotal	$833.00	8
ECE	Resistors Pack	$13.00	1
	Capacitors Pack	$13.00	1
	Transistors Pack	$15.00	1
	Electronics Kit	$30.00	1
	Batteries	$20.00	1
	Microcontroller	$30.00	2
	RC car	$100.00	1
	Wiring Kit	$20.00	1
	Bluetooth module	$10.00	1
	PCB	$10.00	3
	ECE Subtotal	$311.00	13
	Estimated Total Expenses	$1144.00	21

As mentioned throughout this report in section 4.2.1 Time and Economic Constraints as well as in this current section the Mobile LiDAR Scanner is currently being funded by the four group members. However, the College of Optics and Photonics has announced that there will be some funding allocation for certain groups in Senior Design 1. This decision is to be made by Dr.

Arvinda Kar and Dr. David Hagen, Dean of the College of Optics and Photonics. If group 2 is selected to have their budget funded the financial sponsor for the Mobile LiDAR Scanner will be the College of Optics and Photonics rather than the group. This gives the group some financial freedom to improve the design and construction of the Mobile LiDAR Scanner. Some of these design elements have been introduced in previous sections as design considerations. In order to introduce these extra features, the group has decided that if the extra funding is received then we will allow for this funding to cover the current costs of the materials and use our own personal finances to cover the difference for adding the extra features.

In addition to the breakdown of the project budget, a bill of materials (BOM) is shown below in the Table 22. The bill of materials shows what parts we have currently (as of the end of the Fall 2021 semester) decided to use, and which distributor used bought/plan to buy the part.

Table 22: Current BOM

Quantity	Name	Parts Number	Manufacturer	Price per unit	Distributor	URL (if applicable)	Purchased
1	Bluetooth module	HC-05	DSD Tech	$9.99	Amazon	Link	Yes
1	TI LaunchPad with MSP430FR6989	MSP-EXP430FR6989	Texas Instruments	$20.00	Texas Instruments	Link	Yes
1	XiaoR Geek Tank Mobile Platform	(Not given)	XiaoR Geek	$82.99	Amazon	Link	Yes
1	Industrial High Power Flood Illuminator	ATBX-00	ams	$12.31	Digikey (through CREOL)	-	Yes
1	High-power IR VSCEL	INV-C33CTOHIR	Inolux	$9.94	Digikey (through CREOL)	Link	Yes
1	epc635 Card Edge Connector Carrier	EPC635-002 CC CHIP CARRIER	ESPROS Photonics AG	$72.58	Digikey (through CREOL)	Link	Yes
1	Vertical Card Edge Connector	MEC6-130-02-L-DV-A	Samtec Inc.	$9.20	Digikey	Link	No
3	20-pin headers	TSM-121-02-T-SH-013-P	Samtec Inc.	$3.66	Digikey	Link	No

References

[1] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*. MIT Press, 2005.

[2] C. R. Q. L. Y. H. S. L. J. Guibas, "PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space," presented at the Conference on Neural Information Processing Systems (NIPS), Stanford University, 2017. [Online]. Available: http://stanford.edu/~rqi/pointnet2/.

[3] *pointnet2*. (2018). Github. [Online]. Available: https://github.com/charlesq34/pointnet2

[4] A. R. a. A. Davis, "Point Cloud Based Machine Learning for Event Classification and Track Identification of Nuclear Reactions," presented at the Southeastern Section of the APS, Tallahassee Florida, 2021, H01.00036.

[5] O. S. Company. *Planar Diffused Silicon Photodiodes*, p. 5. [Online]. Available: http://www.osioptoelectronics.com/Libraries/Datasheets/Multi-Element-Arrays.sflb.ashx.

[6] E. P. Corporation. *DATASHEET – epc635 3D TOF imager 160 x 60 pixel*, ESPROS Photonics Corporation, EPC635 2020, p. 60. [Online]. Available: https://www.espros.com/downloads/01_Chips/Datasheet_epc635.pdf.

[7] S. M. K. Hasan and T. Chowdhury, "Face Recognition Using Artificial Neural Networks," 2004.

[8] ThorLabs. *NIR Bandpass & Laser Line Filters: 700 - 1650 nm Center Wavelength* ThorLabs. [Online]. Available: https://www.thorlabs.com/newgrouppage9.cfm?objectgroup_id=1000.

[9] A. Pacala, "How Multi-Beam Flash Lidar Works," in *OUSTER Blog* vol. 2021, OUSTER, Ed., ed, 2018.

[10] W. E. e. G. C. KG. *Infrared LED 940 nm* 2018, p. 8. [Online]. Available: https://www.we-online.com/katalog/datasheet/15400394A3590.pdf.

[11] J. Novet. "Microsoft wins U.S. Army contract for augmented reality headsets, worth up to $21.9 billion over 10 years." CNBC. https://www.cnbc.com/2021/03/31/microsoft-wins-contract-to-make-modified-hololens-for-us-army.html (accessed November 3, 2021).

[12] J. Fingas. "US Army delays Microsoft's $22 billion HoloLens deal." Yahoo! Finance. https://finance.yahoo.com/news/us-army-delays-microsoft-hololens-deal-202233405.html?guccounter=1&guce_referrer=aHR0cHM6Ly93d3cuZ29vZ2xlLmNvbS8&guce_referrer_sig=AQAAABUqP0ubpjWjyrdH0hTvfQXmpIvJHtGJtdbyjR3A_k9KXSKasRxU-RU8HxAEu21Flk0TWFkVpr2_vi7eMCDOaxvArzKZrnQRR44Zjt2H560B16grsxm5tYiVkvR72l0DSqjZ5orxEWtMse_J54d8CGkfviMOZWCpwdGMmNbPz1Kz (accessed October 25, 2021).

[13] "Petzval field curvature." Wikipedia. https://en.wikipedia.org/wiki/Petzval_field_curvature (accessed November 4, 2021).

[14] M. J. Riedl, *Optical Design Fundamentals for Infrared Systems*, Second ed. (Tutorial Texts in Optical Engineering). Bellingham, Washington USA: SPIE PRESS, 2001, p. 202.

[15] "The Airy Disk and Diffraction Limit." https://www.edmundoptics.com/knowledge-center/application-notes/imaging/limitations-on-resolution-and-contrast-the-airy-disk/ (accessed October 27, 2021).

[16] K. Lambert. "Printing with the PLA filament 1.75 mm: A Complete Guide for You." https://www.makeshaper.com/2020/03/27/printing-with-the-pla-filament-1-75-mm/ (accessed November 29, 2021).

[17] "Bluetooth® Wireless Technology." Bluetooth SIG, Inc. https://www.bluetooth.com/learn-about-bluetooth/tech-overview/ (accessed November 2, 2021).

[18] C. F. C. Pollette. "How Bluetooth Works." howstuffworks https://electronics.howstuffworks.com/bluetooth.htm (accessed November 2, 2021).

[19] "SZDoit Professional Smart Tracked Car Chassis, Remote Control Robot Tank Platform for Arduino / Raspberry pi / NodeMCU with High Torque DC Motor, DIY RC Crawler Caterpillar Steam Learning Kit." Amazon. https://www.amazon.com/dp/B07BHFM9PJ/ref=cm_sw_r_cp_api_glt_fabc_GPQVQXBSRHZE50AM183Z?_encoding=UTF8&psc=1 (accessed November 17, 2021).

[20] "XiaoR Geek Smart Robot Car Chassis Kit Aluminum Alloy Big Tank Chassis with 2WD Motors for Arduino/Raspberry Pi DIY Remote Control Shock Absorbing Tracked Robot Car Toys - Learning Kit (Black)." Amazon. https://www.amazon.com/dp/B09C7TK9Y9/ref=sspa_dk_detail_0?pd_rd_i=B08QZB5MFR&pd_rd_w=9I4l3&pf_rd_p=9fd3ea7c-b77c-42ac-b43b-c872d3f37c38&pd_rd_wg=T4jSF&pf_rd_r=S3T0D7TZQW0J1PQ4E96B&pd_rd_r=3b37893e-26bd-4c3d-88b9-afd197accd9b&spLa=ZW5jcnlwdGVkUXVhbGlmaWVyPUEyOFY4NVY5U0pZWVo2JmVuY3J5cHRlZElkPUEwNDg5Nzc1MzUyN0g5QkkzWlo3NiZlbmNyeXB0ZWRBZElkPUEwOTk3MzY3MlJUMk82Ukw1N041MyZ3aWRnZXROYW1lPXNwX2RldGFpbCZhY3Rpb249Y2xpY2tSZWRpcmVjdCZkb05vdExvZ0NsaWNrPXRydWU&th=1 (accessed November 17, 2021).
[21] tmontaigu. "What is a LAS file?" https://laspy.readthedocs.io/en/latest/intro.html (accessed November 4, 2021).
[22] *laspy/laspy*. (2021). Github. [Online]. Available: https://github.com/laspy/laspy
[23] Z. Cheng, J. Zeng, D. Liang, C. Chang, and B. Wang, "Influence of Initial Phase Modulation on the Sensitivity of the Optical Fiber Sagnac Acoustic Emission Sensor," *Applied Sciences*, vol. 9, no. 5, p. 1018, 2019. [Online]. Available: https://www.mdpi.com/2076-3417/9/5/1018.
[24] T. Instruments. *MSP430FR698x(1), MSP430FR598x(1) Mixed-Signal Microcontrollers*, Texas Instruments 2018, p. 184. [Online]. Available: https://www.ti.com/lit/ds/symlink/msp430fr6989.pdf?ts=1638841159712&ref_url=https%253A%252F%252Fwww.ti.com%252Fproduct%252FMSP430FR6989.
[25] Bitsrc, "Kicad Pcbnew3D screenshot," vol. 229 KB, K. P. D. screenshot.jpg, Ed., ed. Wikimedia Commons 2009.

Appendix A

A.1 EPC635 Register Map

A.1.1Control page 0x00 ~ 0x7F

<table>
<tr><th>Addr.</th><th>Type</th><th>Default</th><th>Description</th></tr>
<tr><td>0x00</td><td>R</td><td>---</td><td>IC type for device family identification. For chip type refer to register 0xFA.</td></tr>
<tr><td>0x01</td><td>R</td><td>---</td><td>IC version for device mask identification. For chip version refer to register 0xFB.</td></tr>
<tr><td>0x11</td><td>R/W</td><td>---</td><td>Address register for indirect read/write access to EEPROM (refer to 15.5 and 15.6)</td></tr>
<tr><td>0x12</td><td>R/W</td><td>---</td><td>Data register for indirect read/write access to EEPROM (refer to 15.5 and 15.6)</td></tr>
<tr><td>0x1C</td><td>R/W</td><td>0x1E</td><td>TCMI ESM FS: frame start label (refer to 6.5)</td></tr>
<tr><td>0x1D</td><td>R/W</td><td>0xE1</td><td>TCMI ESM FE: frame end label (refer to 6.5)</td></tr>
<tr><td>0x1E</td><td>R/W</td><td>0xAA</td><td>TCMI ESM LS: line start label (refer to 6.5)</td></tr>
<tr><td>0x1F</td><td>R/W</td><td>0x55</td><td>TCMI ESM LE: line end label (refer to 6.5)</td></tr>
<tr><td>0x20</td><td>R</td><td>0x00</td><td>Strap scan register high (refer to 5.6.3):
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0..4</td><td>reserved</td><td>0</td></tr>
<tr><td>5</td><td>Strap input 0: I²C address A0</td><td>0</td></tr>
<tr><td>6</td><td>Strap input 1: I²C address A1</td><td>0</td></tr>
<tr><td>7</td><td>reserved</td><td>0</td></tr>
</table>
Default start-up values of these registers are only valid until end of reset phase. Values might be over-written by external pull-up resistors during strap scan phase when reset is released.</td></tr>
<tr><td>0x22</td><td>R/W</td><td>0x30</td><td>DCS selection for 1st frame (refer to the chapter 11.8):
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0</td><td>mgx0 modulator (mga0, mgb0)</td><td>0</td></tr>
</table></td></tr>
</table>

			1	00: DCS 0 01: DCS 1 10: DCS 2 11: DCS 3	0
			2	mgx1 modulator (mga1, mgb1) 00: DCS 0 01: DCS 1 10: DCS 2 11: DCS 3	0
			3		0
			4..7	reserved	0x3
0x25	R/W	0x35	DCS selection for 2nd frame (refer to the chapter 11.8):		
			Bit	**Function**	**Default**
			0	mgx0 modulator (mga0, mgb0) 00: DCS 0 01: DCS 1 10: DCS 2 11: DCS 3	1
			1		0
			2	mgx1 modulator (mga1, mgb1) 00: DCS 0 01: DCS 1 10: DCS 2 11: DCS 3	1
			3		0
			4..7	reserved	0x3

Table A1.1: Address map of the control page (0x00 ~ 0x7F)

<table>
<tr><th>Addr.</th><th>Type</th><th>Default</th><th>Description</th></tr>
<tr><td>0x3A</td><td>R/W</td><td>0x00</td><td>Readout mode for grayscale
0x10: single-ended readout (negative numbers are clipped)
0x00: differential readout. Select this mode by the user application, refer to chapter 9.3 and 10.1</td></tr>
<tr><td>0x3C</td><td>R/W</td><td>0x26</td><td>Modulation control in grayscale mode:
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0</td><td>reserved</td><td>0</td></tr>
<tr><td>1..2</td><td>reserved</td><td>1</td></tr>
<tr><td>3</td><td>reserved</td><td>0</td></tr>
<tr><td>4</td><td>0: LED/LD modulated
1: LED/LD on during integration</td><td>0</td></tr>
<tr><td>5</td><td>0: LED/LD modulated
1: LED/LD off during integration</td><td>1</td></tr>
<tr><td>6..7</td><td>reserved</td><td>0</td></tr>
</table></td></tr>
<tr><td>0x60</td><td>R</td><td>---</td><td rowspan="2">Temperature sensor, refer to chapter 10</td></tr>
<tr><td>0x61</td><td>R</td><td>---</td></tr>
<tr><td>0x71</td><td>R/W</td><td>0x00</td><td rowspan="2">Number of fine DLL delay steps to delay the LED output by approx. 10ps per step. Valid only if bit 2 in register 0xAE is enabled. Refer also to register 0xAE and chapter 5.8. Max. value is 799 (0x31F).
Note: Delay is sensitive to VDD variations and noise.</td></tr>
<tr><td>0x72</td><td>R/W</td><td>0x00</td></tr>
<tr><td>0x73</td><td>R/W</td><td>0x00</td><td>Number of coarse DLL delay steps to delay the LED output by approx. 2ns per step. Valid only if bit 2 in register 0xAE is enabled. Refer also to register 0xAE and chapter 5.8. Max. value is 49 (0x31).
Note: Delay is sensitive to VDD variations and noise.</td></tr>
<tr><td>0x7D</td><td>R/W</td><td>0x04</td><td>Mode control:
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0..1</td><td>reserved</td><td>0</td></tr>
<tr><td>2</td><td>Enable PLL
0: disable
1: enable</td><td>1</td></tr>
<tr><td>3..7</td><td>reserved</td><td>0</td></tr>
</table></td></tr>
</table>

Cont. 35: Address map of the control page (0x00 ~ 0x7F)

A.1.2 RAM page (0x80 ~ 0xEF)

<table>
<tr><th>Addr.</th><th>Type</th><th>Default</th><th>Description</th></tr>
<tr><td>0x80</td><td>R/W</td><td>0x3F</td><td>Clock control:
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0..5</td><td>reserved</td><td>1</td></tr>
<tr><td>6</td><td>Modulation clock source
0: Internal modulation clock
1: External clock from MODCLK input</td><td>0</td></tr>
<tr><td>7</td><td>reserved</td><td>0</td></tr>
</table></td></tr>
<tr><td>0x85</td><td>R/W</td><td>0x01</td><td>Modulation clock divider:
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0</td><td rowspan="5">Modulation clock divider provides clock to the LED/Pixel-field modulator/demodulator circuits by integer division of the internal PLL clock or external MODCLK:

f_{mod_clk} = 80MHz / (modulation clock divider + 1)
Default: 80MHz / (0x01 + 0x01): f_{mod_clk} = 40MHz
Maximal value of modulation clock divider = 0x1F: f_{mod_clk} = 2.5MHz
Note: The LED modulation frequency is 4 times lower than f_{mod_clk}</td><td>1</td></tr>
<tr><td>1</td><td>0</td></tr>
<tr><td>2</td><td>0</td></tr>
<tr><td>3</td><td>0</td></tr>
<tr><td>4</td><td>0</td></tr>
<tr><td>5..7</td><td>reserved</td><td>0</td></tr>
</table></td></tr>
<tr><td>0x89</td><td>R/W</td><td>0x03</td><td>TCMI clock control:
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0</td><td rowspan="5">TCMI clock divider:
f_{tcmi_clk} = 80MHz / (TCMI clock divider + 1)
Default: 80MHz / (0x03 + 0x01) = 20MHz
Maximal value of TCMI clock divider = 0x1F = 2.5MHz</td><td>1</td></tr>
<tr><td>1</td><td>1</td></tr>
<tr><td>2</td><td>0</td></tr>
<tr><td>3</td><td>0</td></tr>
<tr><td>4</td><td>0</td></tr>
<tr><td>5..6</td><td>reserved</td><td>0</td></tr>
<tr><td>7</td><td>DCLK skew enable:
0: disable
1: enable
Used to delay DCLK edge (typ. 2ns) to compensate PCB delays.
Might be particularly useful when TCMI clock divider = 0 (divided by 1).
When set normal, DCLK edge is centred with respect to other TCMI *SYNC*, DATA[7:0] outputs.</td><td>0</td></tr>
</table></td></tr>
<tr><td>0x8B</td><td>R/W</td><td>0x01</td><td>Number of PLL clock periods delay of the demodulation signal path (all modulation modes). It can be used to insert a phase shift between modulation (LED) and demodulation (pixel). 1 PLL clock cycle is 12.5ns @ 80MHz PLL clock. This is equivalent to a distance shift of 1.875m independent of the LED modulation frequency. Note: This phase shift is temperature independent.
0: no delay
1: 1 clock
2: 2 clocks
…
12: 12 clocks (max. value)</td></tr>
</table>

Table 36: Address map of the RAM page (0x80 ~ 0xEF)

<table>
<tr><th>Addr.</th><th>Type</th><th>Default</th><th>Description</th></tr>
<tr><td>0x90</td><td>R/W</td><td>0xC4</td><td>LED driver control:
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0</td><td>reserved</td><td>0</td></tr>
<tr><td>1</td><td>Inverts output signals LED and LED2 if drivers are enabled
0: not inverted, e.g. LED = 0, not active: Pin LED non-conductive, LED2 = VSSIO.
1: inverted, e.g. LED = 0, not active: Pin LED conductive, LED2 = VDDIO.</td><td>0</td></tr>
<tr><td>2</td><td>LED output select:
0: LED driver is disable. Pin LED is non-conductive.
1: LED driver is enabled.</td><td>1</td></tr>
<tr><td>3</td><td>reserved</td><td>0</td></tr>
<tr><td>4</td><td>LED/LD permanently on (torch function, no modulation) if drivers are enabled:
0: off
1: on (Refer to IMPORTANT NOTE chapter 5.7)</td><td>0</td></tr>
<tr><td>5</td><td>LED2 output select:
0: LED2 driver disabled. Output is in Tri-State with termination resistor to VSSIO.
1: LED2 driver enabled.</td><td>0</td></tr>
<tr><td>6..7</td><td>reserved</td><td>1</td></tr>
</table></td></tr>
<tr><td>0x91</td><td>R/W</td><td>0x03</td><td>Sequencer control:
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0..1</td><td>reserved</td><td>1</td></tr>
<tr><td>3..5</td><td>reserved</td><td>0</td></tr>
<tr><td>6</td><td>If enabled, avoids readout rollover when using slower DCLKs e.g. DCLK < 10MHz. Stretches HSYNC for slower TCMI interface. Causes reduced DCS frame rate due to additional 2µs per ADC conversion (t_{conv}/2 + 2µs).
0: disable for DCLK > 10MHz (default)
1: enable for DCLK = 10MHz or lower</td><td>0</td></tr>
<tr><td>7</td><td>reserved</td><td>0</td></tr>
</table></td></tr>
<tr><td>0x92**</td><td>R/W</td><td>0x34</td><td>Modulation select:
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0..1</td><td>reserved</td><td>0</td></tr>
<tr><td>2</td><td>reserved</td><td>1</td></tr>
<tr><td>3</td><td>Dual integration time mode – acquisition with 2 integration times per DCS frame using additionally integration length 2 registers 0x9E and 0x9F:
0: disable
1: enable
Needs register 0x94 set to 0x80, otherwise it is not effective (see 39)</td><td>0</td></tr>
<tr><td>4</td><td rowspan="2">Number of DCS readouts select:
00: Grayscale mode, DCS0 only
01: Dual phase mode, DCS0, DCS1 or DCS2,DCS3
10: reserved
11: Full resolution mode or dual int. mode, DCS0, DCS1, DCS2, DCS3</td><td>1</td></tr>
<tr><td>5</td><td>1</td></tr>
<tr><td>6</td><td rowspan="2">Modulation select:
00: TOF mode
01: reserved
10: reserved
11: Grayscale mode</td><td>0</td></tr>
<tr><td>7</td><td>0</td></tr>
</table></td></tr>
<tr><td>0x94**</td><td>R/W</td><td>0x00</td><td>Pixel operating and readout control. Refer to chapter 8.2 and 11.5:
0x00: Default, TOF mode: full resolution mode
0x80: Dual phase or dual integration time mode</td></tr>
</table>

Cont. 36: Address map of the RAM page (0x80 ~ 0xEF)

<table>
<tr><th>Addr.</th><th>Type</th><th>Default</th><th>Description</th></tr>
<tr><td>0x9D**</td><td>R/W</td><td>0x50</td><td>ADC resolution control, refer to chapter 8.4:
0x50: 12 bit resolution
0x34: 8 bit resolution</td></tr>
<tr><td>0x9E**</td><td>R/W</td><td>0x07</td><td rowspan="2">Integration length 2: Number of modulation clock periods for the second integration time in the dual integration time mode (refer to 8.2.3, default: 2'047). See registers 0xA2 and 0xA3 for functional definition details.
Bit 3 in register 0x92 has to be set to 1 to enable this integration time for the even rows. The odd rows operate with the integration length 1 set in registers 0xA2 and 0xA3.</td></tr>
<tr><td>0x9F**</td><td>R/W</td><td>0xFF</td></tr>
<tr><td>0xA0**</td><td>R/W</td><td>0x00</td><td rowspan="2">Integration time multiplier (10 bit value) for integration lengths set with the integration length registers (default = 1, min. value = 1). This multiplier is active on both settings integration length 1 and 2.</td></tr>
<tr><td>0xA1**</td><td>R/W</td><td>0x01</td></tr>
<tr><td>0xA2**</td><td>R/W</td><td>0x07</td><td rowspan="2">Integration length 1: Number of modulation clock periods for the (first in the dual integration time mode) integration time (16 bit value, default = 2'047, min. value = 7 which is integration time 200ns @ 10MHz).
Integration time = Integration time multiplier * (Integration length +1) * t_{mod_clk}
e.g. for defaults @ 10MHz modulation clock = 51.2µs
Note: (Integration length + 1) should be evenly divisible by 4.</td></tr>
<tr><td>0xA3**</td><td>R/W</td><td>0xFF</td></tr>
<tr><td>0xA4</td><td>R/W</td><td>0x00</td><td>Shutter Control:
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0</td><td>Shutter release. Refer to chapter 6.2
0: disable
1: enable. In single shot mode: Starts acquisition and is auto cleared.
Note: Shutter release is not auto-cleared when multiple frames is enable.</td><td>0</td></tr>
<tr><td>1</td><td>Multiple frames (auto-run or video mode). Refer to chapter 6.2
0: disable. Single shot mode.
1: enable. Multiple frame mode active if shutter enabled. Refer to chapter 6.2.2.</td><td>0</td></tr>
<tr><td>2..7</td><td>reserved</td><td>0</td></tr>
</table></td></tr>
<tr><td>0xA5</td><td>R/W</td><td>0x07</td><td>Power control (Refer also to11.6)
0x00: Power off
0x07: Power on</td></tr>
<tr><td>0xAE</td><td>R/W</td><td>0x01</td><td>DLL control (Refer also to register 0x73 and chapter 5.8):
0x01: no delay
0x04: delay manually set by register 0x73
Note: The change of register 0xAE from 0x01 to 0x04 generates also a delay, even if register 0x73 is set to 0x00.</td></tr>
<tr><td>0xCA</td><td>R</td><td>0x20</td><td>
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0</td><td rowspan="2">reserved, I²C address A1, A0 of 7-bit I²C device address. Programmable only during reset via strap pins using external pull-up resistors.</td><td>0</td></tr>
<tr><td>1</td><td>0</td></tr>
<tr><td>2</td><td rowspan="5">I²C device address A6 … A2 of 7-bit I²C device address.
Programmable via direct access from I²C or from EEPROM during start up, followed by an I²C general call “Device address reload” to take it into effect.</td><td>0</td></tr>
<tr><td>3</td><td>0</td></tr>
<tr><td>4</td><td>0</td></tr>
<tr><td>5</td><td>1</td></tr>
<tr><td>6</td><td>0</td></tr>
<tr><td>7</td><td>reserved</td><td>0</td></tr>
</table></td></tr>
</table>

Cont. 36: Address map of the RAM page (0x80 ~ 0xEF)

<table>
<tr><th>Addr.</th><th>Type</th><th>Default</th><th>Description</th></tr>
<tr><td>0xCB</td><td>R/W</td><td>0x23</td><td>I²C and TCMI control:
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0</td><td>I²C clock stretching
0: disabled
1: enabled</td><td>1</td></tr>
<tr><td>1</td><td>I²C pins input spike filter
0: disabled (> 1MHz)
1: enabled (≤ 1MHz, FM+)
When I²C pins input spike filter = 0, SDA and SCL pins can be used up to 10MHz as inputs (driven rail-to-rail by a real CMOS driver, no pull-up) and up to 2MHz as outputs.</td><td>1</td></tr>
<tr><td>2</td><td>TCMI ESM mode enable (refer to 6.5)
0: disabled
1: enabled</td><td>0</td></tr>
<tr><td>3</td><td>TCMI DCLK mode select (refer also to6.1)
0: continuous
1: gated</td><td>0</td></tr>
<tr><td>4</td><td rowspan="2">TCMI data format (refer also to 6.4).
01: 8 bit mode. Transfers the 8 MSB bits of the pixel data with 1x DCLK.
10: lsb/msb split mode: Transfers 12 bit pixel data with LSByte leading and MS-Byte trailing with 2x DCLK. Data are MSB aligned (default). The optional SAT bit is on the LSB.
11: msb/lsb split mode: Transfers 12 bit pixel data with MSByte leading and LS-Byte trailing with 2x DCLK. Data are MSB aligned. The optional SAT bit is on the LSB.</td><td>0</td></tr>
<tr><td>5</td><td>1</td></tr>
<tr><td>6</td><td>When split modes selected (= 11 or 10), forces bit DATA[0] of the LSByte = 1 when the pixel is saturated. Not effective with other TCMI data formats.
0: disabled
1: enabled</td><td>0</td></tr>
<tr><td>7</td><td>reserved</td><td>0</td></tr>
</table></td></tr>
<tr><td>0xCC</td><td>R/W</td><td>0x00</td><td>TCMI polarity settings:
<table>
<tr><th>Bit</th><th>Function</th><th>Default</th></tr>
<tr><td>0</td><td>DCLK edge select to align all other TCMI outputs
0: falling edge
1: rising edge</td><td>0</td></tr>
<tr><td>1</td><td>HSYNC polarity
0: HSYNC active low
1: HSYNC active high</td><td>0</td></tr>
<tr><td>2</td><td>VSYNC polarity
0: VSYNC active low
1: VSYNC active high</td><td>0</td></tr>
<tr><td>3</td><td>XSYNC polarity
0: XSYNC active low
1: XSYNC active high
Only effective when bit 6 is set to 0</td><td>0</td></tr>
<tr><td>4</td><td>DATA[11:0] unsigned/signed TCMI data output format
0: unsigned integer, subtract 2'048 to get correct value (Default)
1: two's complement signed integer (-2'048 ... 2'047).</td><td>0</td></tr>
<tr><td>5</td><td>reserved</td><td>0</td></tr>
<tr><td>6</td><td>Select XSYNC / SAT output pin function
0: XSYNC
1: SAT</td><td>0</td></tr>
<tr><td>7</td><td>Force DATA[11:0] = 0xFFF (unsigned) / 0x7FF (signed, two's complement) during data-out operation when corresponding pixel is saturated
0: disabled
1: enabled</td><td>0</td></tr>
</table></td></tr>
</table>

Cont. 36: Address map of the RAM page (0x80 ~ 0xEF)

Addr.	Type	Default	Description
0xCD	R/W	0x13	ADC control, refer to chapter 8.4: 0x13: 12 bit resolution 0x1B: 8 bit resolution Registers 0x9D must be set according chapter 8.4
0xE8	R/W	---	Temperature offset correction for the calculation according the formula in chapter 10 by the application SW. Range approx. -27 … +27ºC with around 0.2ºC steps. The reference temperature is +27ºC. 0x7F (127) corresponds to 0ºC offset. 0xFF: Function is not supported.
0xE9	R/W	---	DLL step. Supported for Wafer IDs 40 or higher. Refer for details to register 0x73 and 21. The exact value is t_{DLL} = ((register 0xE9 -128) * 0.003ns) + 2.1ns (at +27ºC, V_{DD},V_{DDPLL} = 1.8V). 0xFF: Function is not supported.

Cont. 36: Address map of the RAM page (0x80 ~ 0xEF)

A.1.3 EEPROM page, indirect data access section (0xF0 ~ 0xFF)

Addr.	Type	Default	Description
0xF0	R/W	0x00	User register for user data. Do not write the register during frame acquisition. The number of WRITE cycles into the EEPROM should not exceed 100 WRITE operations.
0xF5	R	0x00	Customer ID
0xF6	R	---	Wafer ID
0xF7	R	---	
0xF8	R	---	Chip ID
0xF9	R	---	
0xFA	R	0x04	Chip and part type: 0x04 = epc635
0xFB	R	---	Chip and part version (release) e.g. 0x01 for version -001

Table 37: Address map of the EEPROM page (0xF0 ~ 0xFF)

A.2 EPC635 3D TOF Distance Measurement Flow

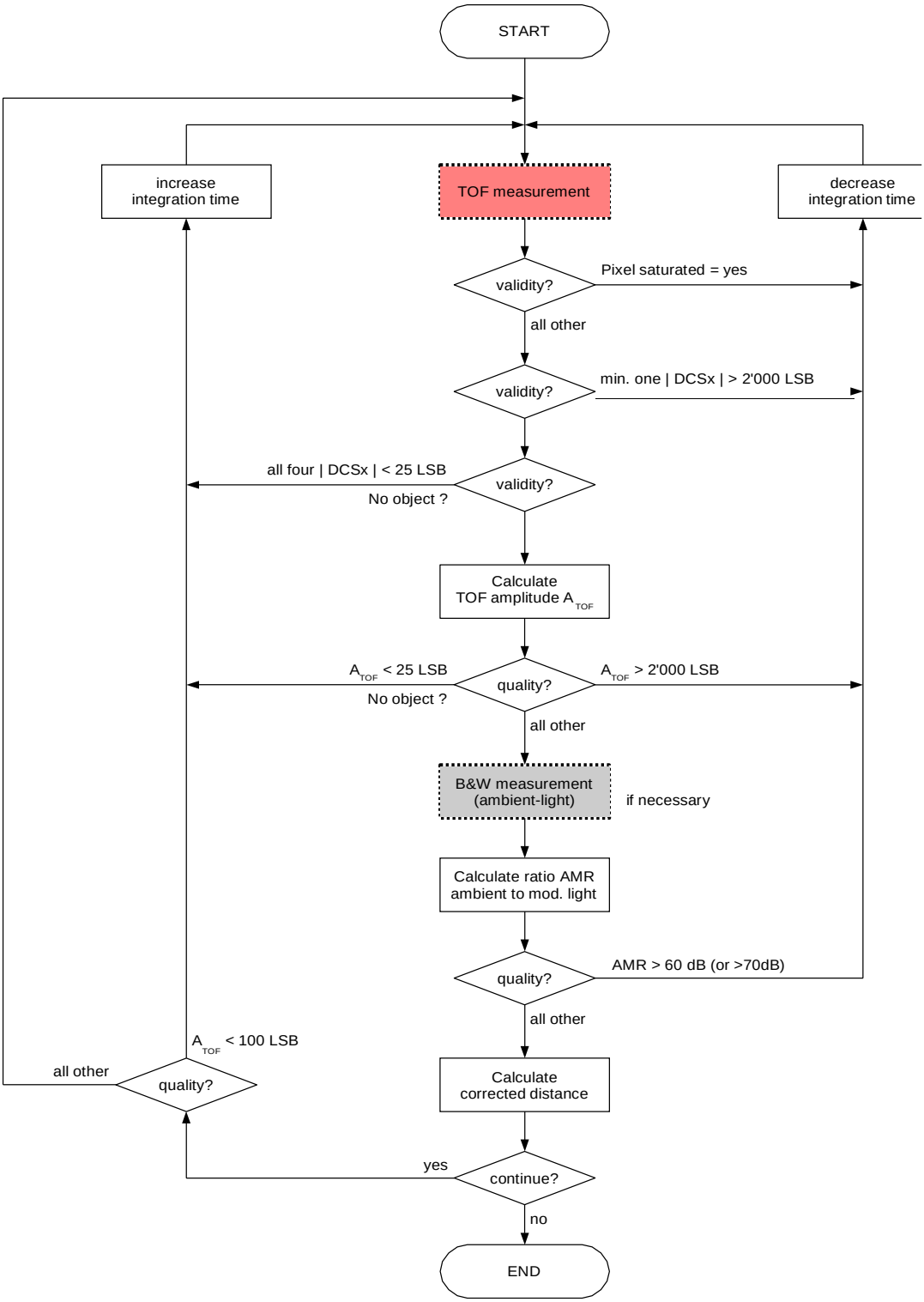


Figure A.2.1: Generic validity and quality flow chart for a single pixel

A.3 EPC635 Pin Layout

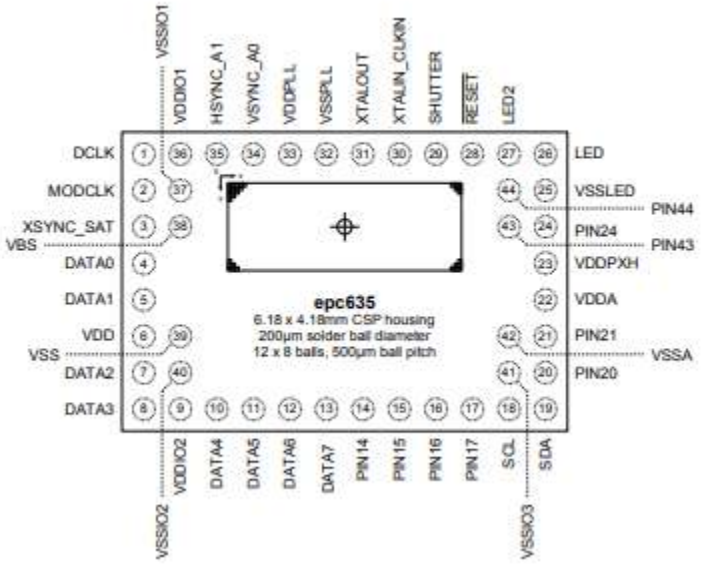


Figure A.3.1. Pin Layout of CCD Sensor [6]