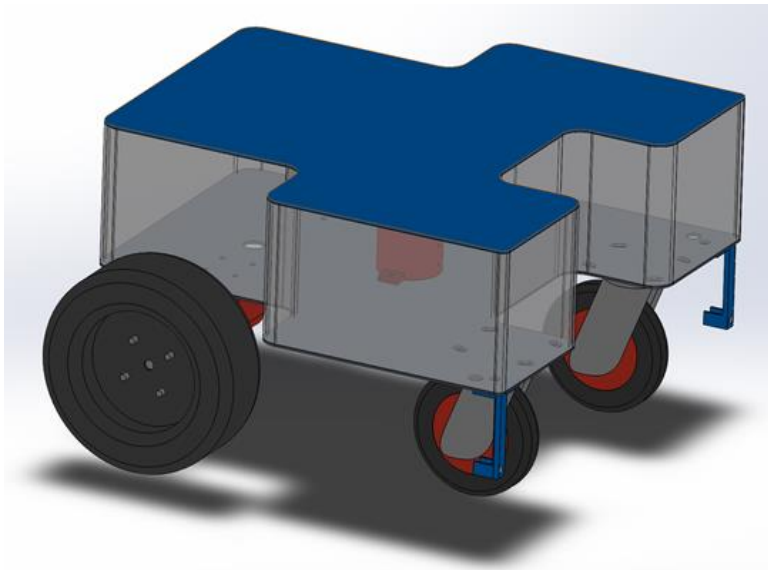
E-Goat Robotic Solar Farm Grass Cutting System



Department of Electrical Engineering and Computer Science

University of Central Florida

Dr. Lei Wei & Dr. Samuel Richie

Sponsors: OUC & Duke Energy

April 21, 2020

Senior Design II

Group 26

Steven Cheney	Computer Engineer
Jordan Germinal	Computer Engineer
Eduardo Guevara	Computer Engineer
Davis Rollman	Computer Science
Jonathan Smith	Electrical Engineer

Table of Contents

Table of Figures

Table of Tables

1 Executive Summary

Solar Panels Farms require more maintenance than what many people may think. As with a large grassy area or field for any kind of use, the grass in the area must be kept in order to keep a comfortable working environment outdoors. Solar Panel Farms typically use traditional grass fields in order to keep the cost of construction and infrastructure low. However, according to Duke Energy and the Orlando Utility Commission, keeping the lawn serviced and cut adds an additional cost to annual maintenance expenses of around $150,000 to $200,000 per 500 acres. In addition to the cost expenditure, the mission focus of producing and using renewable sources of energy is tampered by using services operating fossil fuel equipment. Most lawn services use either diesel or fuel powered trimmers, mowers, and blowers to effectively reduce vegetation.

Duke Energy and the Orlando Utility Commission have decided that an effective way to reduce such expenditures and carbon footprint would be to create an electric powered autonomous grass cutting rover. The rover will be sponsored as a joint partnership between Duke Energy and the Orlando Utility Commission set budget of $1500. The project will consist of several engineering disciplines, including Electrical Engineering, Mechanical Engineering, Computer Science, and Computer Engineering students in order to create an all-around effective solution. As part of a demonstration to the sponsored companies and extra effort to gather input from prototypes, a competition will be held at a Solar Panel Farm within the Walt Disney World property managed by Duke Energy. The competition will be in between not only University of Central Florida teams, but also teams from the University of Western Florida, with similar engineering disciplines consisting in their teams.

Two teams will work together to create an end solution assigned with different tasks and roles. The purpose of the Electrical Engineering, Computer Science, and Computer Engineering teams will be on the hardware stack and software stack of the overall system. The hardware stack will include actuators, sensors, peripheral sensors, power systems, printed circuit boards, cabling and wiring placement and so on. The software stack will consist of raw sensory data and actuator I/O, low-level serial communication processing, actuator commands, high-level path finding and object recognition algorithms, and so on. The Mechanical Engineering team will create the physical unit structure of the rover. This includes but not limited to body framework, wheels, dampening equipment, cooling solutions, electronics housing, shielding elements, and so on. After work from both teams have been completed, the teams will have produced a fully functional high-performance working prototype to fit Duke Energy's and the Orlando Utility Commission's requirements and constraints.

The purpose of this documentation is to explicitly layout: the projects summary as a whole, the goals we aim to achieve, the research that brought us to our solutions, and engineering specifications in our final designed product. This information will be accompanied by several tables, block diagrams, and images in organizationally keep record of high-level and low-level concepts. Within this documentation, the technical requirements and specifications of our sponsors Duke Energy and the Orlando Utility Commission (the customer) are presented here. The project will be broken down into a series of task checkpoints starting with creating engineering requirements and specifications based on customers specifications and constraints. We will be starting off with a research phase, where we will be looking more in depth into the types of terrain that the solar farms are built on. We will then begin research into the hardware that can aid in a potential success for this project design. This step will require more research and experimentation to see what does and what does not work. Finally, we will be in the construction phase, where a prototype will be designed through trial and error processes based off of the parts that we decide have made it to the final round.

2 Project Description

This chapter of the document provides background information and a large picture idea of the motivation and goals that end solution will be based upon. The following sub-chapters include project motivation, goals, existing project and products, engineering specifications, and a house of quality study.

2.1 Project Motivation

Solar panels "harvest" energy from the sun using photovoltaic panels by allowing photons, or particles of light, to detach electrons from atoms. This in turn generates a flow of electricity. These solar panels are usually installed and set into an array on a large grassy field in order to produce more energy, avoiding obstruction from large objects such as buildings, trees, etc. in order to have the highest yield of renewable energy. These arrays of solar panels are commonly referred to as a solar panel farm, where private companies, and even today the common house-owner, collect the sunlight to reduce carbon footprint emissions and produce energy through a cleaner, more environmentally friendly way. By converting the free energy of the sun into power, it seems that there is only profit to be gained. After an initial large fee to set up and install these panels, of course.

Although this source of energy becomes a "free" source of energy over time, there are still severely expensive upkeep costs. For example, most of these farms are built over grass, as the grass neither absorbs nor reflects heat as severely as a flat concrete pad. However, as all grass does in the world, it grows every day. The maintenance on such fields can be a major expense to the companies supplying the services and impacts the cost of energy users in the end. This requires constant lawn services to be contracted to keep the grass cut and low, in order to prevent any obstruction of view of the sunlight reaching the solar panels. Because of the way solar panels are built, this makes it very difficult for people and mowers to fit under and between the giant support beams and electronics that run these panels. Even worse, as we are all human, the potential for error exists when one is needed to cut near such expensive equipment. Accidents may occur and increase the risk of equipment damage and replacement.

Our solution is the "e-GOAT": an AI-assisted autonomous solar powered rover-based robot that can cut the grass and reduce the cost of maintenance of the solar farm. The "e-GOAT" robot will be a high-functioning autonomous lawn mower that will move through terrain in order to cut the grass. The "e-GOAT" will be able to autonomously identify areas of grass that need attention, avoid obstacles, and provide motion and navigation to the land mower bot. The low-cost bot will also be friendlier to the environment than traditional lawn service equipment; the bot will be electrically powered, unlike most regular conventional mowers, which in turn will allow it to reduce the carbon footprint to almost, if not completely, to zero. As an added benefit, it will be able to operate at day or night, alleviating the need for extra lighting around the plant.

2.2 Goals

The goals of this projects as specified by the customer are as follows:

1. To produces a rover-based robot and provide the articulated motion needed to move the weed whacker across the terrain and cut grass.
2. To produce a Navigation and Control solution to identify grass areas that need attention (i.e., cutting), avoid any obstacles such as solar panel structures, and provide overall motion control of the rover-based robot.
3. To produce a power supply unit for the rover. Grass cutting may take place at night and charge during the day. To accommodate such a scenario the system must use a defined battery storage technology with charging capability.
4. To complete the challenge competition at during the month of April in 2020. The main objective is to complete the 10-foot-wide, by 50-foot-long course working around obstacles within a 15-minute time constraint.

2.3 Existing Projects and Products

This section lays out similar commercial and previous UCF projects that have been implemented and meet similar design requirements to goals of our project. Since this project is the second generation of its kind, our development begins with a reference project from last year's 2018-2019 competition. Not only did we rely on the previous rendition of the project for references, but we also looked at similar commercial solution available to purchase online. With these different products, our team was able to solidify our requirements from the ideas and ingenuity of these similar products. Although these products and projects are similar in function to our current project, these created solutions do not meet the specifications and goals that we are aiming to achieve.

2.4 Articulated Autonomous AI-Assisted Solar Farm Grass Cutter

As previously stated, this project is a second-generation update of a previously held competition held by UCF and USF students in 2018-2019 academic year. For reference to this year’s project, our sponsors, OUC and Duke Energy, have given us permission to use and review the previous competing teams’ autonomous rovers. We also have the choice of choosing one physical unit to keep as reference, test, and design for our second-generation rover.

The physical rover we received was from a team of UCF Students on the “Blue Team” last year. The team consisted of four ECE students; Brendaei Dieter, Chris Entwistle, Mario McClelland, and Daniel Warner. The other ME students included Dalton Cone, Vince Cloyd, Luke Davis, Kyle Izbicky, and Sylvester Quezada. The Green Team E-Goat consists of a large, four wheeled rovers using a string based cutting system in a combination of wood, frame, and plastic body. This rover uses off the shelf batteries, motors, and electrical components to create an autonomous grass trimming rover for solar farm maintenance. Through the use of two controller boards (an Arduino and ODROID-XU4) and various sensors, the rover is able to map its surroundings avoid hitting obstacles while cutting the grass.

On the electric side, the rover consisted of three lithium polymer batteries that where connected to three separate subsystems. These subsystems are the motor control, the ODROID-XU4 power supply, and the GPS beacon power supply. Five motors powered the mowing head spindles and the wheels for locomotion. The mower was also equipped with a kill switch to the back of the unit and LiDar sensor on the top. Although there is some debate as to whether the LiDar sensor had any use in the project due to the four ultrasonic sensors on the outside of the unit and the lack of mention within the documentation and code.

There are many things with this design that we learned from and incorporate into our own project. We designed our own our project and learn from the shortcomings of the previous generation. One of these shortcomings is the organization of the internal components. There were many missing labels and unshielded wiring within the electronic compartment. The lower bottom of the rover may have been susceptible to short circuits due unshielded aluminum frames. There is also no supporting or internal documentation within the team to lay out the high-level design of the wiring.

Further shortcomings included I/O and computational capabilities of the ArduinoMega and the use of various redundant sensors. The ArduinoMega was used in this project to control the motor drivers through a series of PWM outputs. The microcontroller is an economically good choice to fit within the constraints of the project budget. However, ramp up time needed to program and wire the ArduinoMega is a large detriment to the timing constraints of the team. The use of a professional hobbyist board would be better much alternative with pre-layer I/O pins with useful markings, labels, and LED indicators.

The Blue Team rover is a successful attempt in completing the challenge put on by OUC and Duke. It was only two of the four UCF senior design rovers that was able to successfully compete in the field competition. Great capabilities such as a strong visual sensor, strong processing power, and good coding led this rover to be third place within the competition. With this in mind, we look forward to working on this past integration as a foundation for the new second generation Blue Team E-Goat rover.

2.4.1 Husqvarna Auto Mower Series

Autonomous lawn mowers have been a commercially available product for the past twenty years. In fact, the first US patented design of an autonomous lawn mower was patented by a company named Mowerbot in 1969. However, the first profitable line of mowers come from a Swedish company named Husqvarna. Husqvarna created a series of different autonomous lawn mower rovers capable of mowing large lawns up to over an acre for their top line mowers. These mowers can cost in a range between $1,600 USD MSRP to $3,500 USD MSRP depending on specifications and performance. The lowest tier mower, the A 310, fits the range of the project budget with the most core specifications and requirements met. Although this is a commercial product and details are limited to online marketing visuals and details, we can still gather some useful information into its capabilities, maintenance, and operation.

At the lower-class rovers in the Husqvarna mower series, a competitive edge is already offered to everyday conventional human operated lawn mowers. These lower-class mower rovers can already offer a multitude of capabilities including but not limited to a cutting area of approximately half an acre, a sixty-minute recharge cycle, and a tolerance of up to 30% inclinations. The build of the lower end models offers a hard-exterior ABS plastic cover that will guard well against different outdoor conditions. The Husqvarna series also features wireless smartphone connectivity to monitor activity and send simple control commands to the rover while active. This feature uses Bluetooth connectivity and can send messages within a thirty-meter range. The professional end models also feature a patented design of a GPS navigation system used to understand its position within a field.

One of the main drawbacks of the Husqvarna autonomous mower is the path planning capabilities it lacks. Essentially, the mowers approximately are able to cover 100% of their specified lawn by randomly selecting a heading and position to travel too while avoiding obstacles. It is assumed that if the rover is utilized every day of the week for a certain number of hours a day depending on the lawn size, the rover will cut the whole lawn. We implement a smarter path planning procedure that is able to know areas of grass that need attention. We also implemented a rover that is able to maintain a heading and create lawn patterns for consistent cuts. Another feature that Husqvarna lacks is lack of easy setup. Manual installation of boundary wire on the lawn is required before putting the rover to use. This boundary keeps the rover from passing over certain areas of lawn or plant beds. To avoid this, a device with very high precision GNSS or location system must be used in order to avoid certain areas.

Overall the Husqvarna autonomous mowers are a good source of reference. When it comes to physical design, actuator control, and connectivity, the Husqvarna specifications meet the requirements and constraints our project is trying to achieve. On top of that, the Husqvarna rover is also our competitor to create a more functional autonomous lawn mower for a lower cost.

2.4.2 iRobot Terra t7 Robot Mower

A familiar name to indoor and commercially available robotics, iRobot is an American company working on robotics since 1990. The company has been a popular brand for robotic vacuum and wet mop cleaners. Recently, iRobot has been developing a product called the Terra to take its robotic indoor capabilities to mow lawns outdoors. The Terra is commercially available in Germany while it is still in a beta program in the United States as of 2019. The Terra resolves some of the major shortcomings of current lawn rovers, including replacing boundary wire with sensor arrays to avoid plant beds, lawn mapping, and advanced patterned grass cutting. With limited details on specifications through its website, we are not able to pick out details on different sensors and specific technologies in use. However, we are able to decipher some unique features through its marketing facade.

As mentioned, the Terra diverts from boundary wires for avoiding areas and obstacles to using wireless beacons that are used to find the position of the rover through the yard. This technology is not discussed nor reported in detail in any of iRobots product or documentation pages. However, through several new sources and public domain records, iRobot has filed an FCC application in 2015 to ultra-wideband beacons. The robot also uses an advanced planner to mow the lawn in a fixed pattern. The robot is also able to detect large objects and avoid them using two frontal sensors. The Terra also features a safety key to start the mower, an external emergency stop button, and a joystick controller to manually take control of the rover.

The Terra has many features that we would like to incorporate into our project. To satisfy the boundary constraints of the project, a precise navigation system like the Terra should be used. We would also like to incorporate the precise path planning that the Terra to cover as much grass as possible within one cut. However, the team needed to decide whether these technologies fit within the budget and timing constraints of the project as some of these technologies are complicated to implement and expensive to purchase specialized equipment. For this reason we chose to not implement a complex solution while still delivering a minimum viable product.

2.4.3 Existing Rover Location and Navigation Technology

Based on the previous year's team performance, the customer had expressed an interest to extend the autonomy of the second-generation solar farm rovers. It is then highly important to research and distinguish different technologies to navigate and plan an autonomous agent on an open field. After reviewing the location technologies of the rovers above, there are three main methods that position and location can be determined: Inertial Navigation Systems (INS) Sensors, GNSS & Boundary Wire technology, or Hybrid Solutions. The table below, Table 1, describes the key features of each of these technologies.

Table 1: Existing Location and Navigation Tech on Rovers

	INS	GNSS & Boundary Wire	Hybrid
Key Features	Gather relative position from a relative point of origin using velocity and heading through the use of filtering algorithms	The use of satellites to provide a geo-positioning	Use of different sensors such as Ultrasonic, RF, Infrared, etc. to create a fusion to triangulate distance from base to beacons

2.4.3.1 INS

Inertial Navigation Systems are a navigational device that use a combination of locomotive sensors and computational power in order to continuously calculate local current position. The sensors inside an INS system may consist of accelerometers, gyroscopes, and magnetometers, usually all included in an Inertial Measurement Unit (IMU). Through IMUs sensor in combination with a computational board, the sensor readings can be used to determine position using a frame or reference point of origin. This system allows the device to not be dependent on external references or navigation systems such as Geo-Satellites or external beacons. Even though the sensors on these systems can be extremely accurate, devices using INS are subject to drift and noise error. As the sensor over time accumulates error, the true location of the locomotive will not accurately reflect the real-world state. This can be a large issue in our implementation of our product since we need to accurately define where we are in large environment for over one-hour.

2.4.3.2 GNSS & Boundary Wire

Global Navigation Satellite Systems (GNSS) is the second form of location awareness that uses satellite navigation in order to determine the approximate location of a receiver on the globe. A GNSS capable system can be used virtually anywhere on the surface of the globe provided ideal environment and weather conditions available such as clear skies and no obstruction of large land masses or objects. In these ideal conditions, GNSS systems are capable of typically giving 5 to 10-meter accuracy from a specific receiver. To constrain a rover with this margin of error, a Boundary wire is usually set around the perimeter. This wire carries a certain frequency that can be sensed by the rover in order not to cross over it. The accuracy of GNSS technology can also be greatly increased by using more than one receiver to estimate the margin of error and at times get 50mm to 10cm accuracy. If our project is to solely use GNSS technology, this differential technology should be investigated and used to produce the best location data to navigate the lawn.

2.4.3.3 Hybrid Solution

Hybrid location awareness uses several different technologies and algorithms in order to geo-locate an agent receiver in an environment. Examples of certain technologies include but are not limited to: Ultrasonic Sensors, Radar, Infrared Technologies, LoRa for data communication, etc. Typically, hybrid solutions set fixed location points for a frame of reference for a moving agent. These location points act as a “light-house” or “beacon” to allow the free moving agent to calculate its position. Usually three or more beacons are necessary in order to produce accurate results within a state space. Depending on the environment and design constraints, certain sensors will have an advantage over other technologies. In our design constraints, line of sight to a beacon is not guaranteed and our rover performs outdoors, an Ultrasonic sensor would be favored in the design of the system. This system, however, can become extremely complex to design and test due to the integration of multiple technologies and extra subsystems such as beacons to add in addition to the scope of the project.

2.5 Engineering Specifications

The engineering requirement specifications section shows the customer specifications given by the sponsor (OUC and Duke Energy) of what they desire this project to achieve. In turn, we construct the requirement specifications given the customers wants and constraints.

2.5.1 Customer Specifications & Constraints

Through official documentation given to us by OUC and Duke energy, the customer has laid out the focus and plan of a successful product they would like to implement onto their solar farms. In the broadest generalization provided by the customer, they are requesting “an AI-assisted, autonomous solar powered rover-based robot to articulate and guide electric weed whackers.” The customer also articulates certain aspects and features it would like to see through the second generation of these rovers such as more navigation and motion features. The following list, depicted in Table 2 and Table 3, breaks down the system requirements given by the customer.

Table 2: Customer Specifications & Constraints Part 1

Designation	Description
CSC-01	Robotic rovers must use an off-the-shelf battery powered trimmer (no metal blade must be string-based) to cut grass.
CSC-02	The rover may consist of an off-the-shelf remote-controlled system that is modified for the application but should still have an autonomous mode or capability.
CSC-03	The robotic grass cutting rovers must be equipped with a remote kill switch that can turn off the cutting system and locomotion at a distance of approximately 50 feet.
CSC-04	An off-the-shelf battery and charger must be utilized.
CSC-05	The rover must be capable of safely navigating in uneven terrain (~ 3-inch terrain differential over ~ 2-foot span in any direction) without capsizing while avoiding a series of obstacles.
CSC-06	No part of the system must be of a height no taller than 20 inches from the ground

Table 3: Customer Specifications & Constraints Part 2

Designation	Description
CSC-07	The system must operate independently and have no attachments to existing solar farm array structures.
CSC-08	Total system materials and assembly cost target: $1500.
CSC-09	The ability to cut grass at an acceptable height (3 to 6 inches) is considered a plus. It is expected that teams will adapt and mount a commercially available string trimmer head to their devices.
CSC-10	The system must traverse the large areas and maneuver around PV support structures.
CSC-11	Avoid any damage to surrounding infrastructure, the environment and humans
CSC-12	Provide a math model to estimate how much grass area the robot can cut per hour.
CSC-13	System to provide a secondary safety protocol to deal with rogue objects, in addition to the remote kill switch.
CSC-14	The System also to include location beacon with independent power supply for a defined period of time.

2.5.2 Requirement Specifications

Aiming to deliver the customer specifications within their constraints, our team has created engineering requirement specifications in order for this project to be successful. These requirements have been divided into our General Requirements Specifications and given a difficulty rating on a scale of 1 to 5 with 1 being the easiest and 5 being the hardest. Tables 4 and 5 further explain the requirements specified to include in this project.

Table 4: Requirement Specifications Part 1

Designation	Description	Difficulty
GRS-01	The rover will cost no more than $1500 to assemble	2
GRS-02	The rover will consist of a remote kill switch that can be activated from 50ft	2
GRS-03	The rover will consist of an off the shelf battery and charger	1
GRS-04	The rover will use an off-the-shelf battery powered trimmer	3
GRS-05	The rover will travel at a speed of 2.5 mph.	3
GRS-06	The rover will not exceed a height of 20” from the ground.	2
GRS-07	The rover dimensions will be 2’ x 2’ x 1.5’	3
GRS-08	The rover will weigh 20lbs	1
GRS-09	The rover will run at a safe temperature of 98°	3
GRS-10	The rover will run for 1 hour until the power is consumed.	4
GRS-11	The rover will stop one foot away from the solar poles	5
GRS-12	The rover will stop 0 feet away from the boundary	5
GRS-13	The rover will be able to navigate in a 50’ x 10’ field	4

Table 5: Requirement Specifications Part 2

Designation	Description	Difficulty
GRS-14	The rover will contain a two-layer printed circuit board with a footprint of 7”x9”	4
GRS-15	The rover will cut the grass to a maximum of 3”	4
GRS-16	The rover electric components will be shielded by a plastic casing	1
GRS-17	The rover will have a maximum voltage consumption of 12v	4
GRS-18	The rover will be able to detect foreign objects from 3’	3
GRS-19	The rover will be capable of navigating uneven terrain at an incline differential of 20°	3

2.5.3 Team and Product Constraints

Laid out in this section are several sections the project team has identified where challenges and hardships can occur. It is good to generally distinguish these constraints in order to appropriately create technical solutions or change approach of a problem.

2.5.3.1 Terrain

There are several ways to structure a Solar plant farm dependent upon the terrain given in the area. We are lucky that Florida is a predominantly flat state, however, other states, and countries, have a much more treacherous terrain to navigate. For the constraint of this project, we will be navigating a majorly flat surface area. We are able to design and plan around a more mountainous terrain, but this would most likely require a different design of wheels than the schematics for the wheels we have in mind.

2.5.3.2 Time

As with any project, there is a deadline for when everything must be fully built and fully operational. Although we are quite an organized group and very efficient in completing all the tasks set before us, not all problems have easy solutions. Dependent on the difficulty of the task set before us, it could potentially hinder the precious time we have to work on realizing this robotic lawn mower.

2.5.3.3 Malfunction

As with the build of any robotic, there is always the chance for an electrical or mechanical part to fail, whether it be on arrival from the manufacturer, or from user error from a bad connection etc. The risk is always prevalent for hardware to malfunction.

2.5.3.4 Monetary Expenditure

Although this project is sponsored, the majority, if not all, of our costs should be covered while researching and constructing this project. However, if for whatever reason we reach the maximum amount allotted to Group 26, and extra parts become astronomically expensive, problems could arise for our group if a solution is not made.

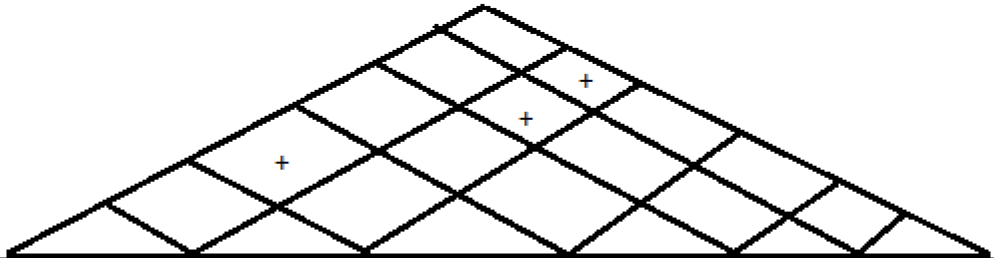
2.5.3.5 Applicable Standards

The first official ANSI standard for robotic lawn mowers was approved September 5, 2019. In this standard, the ANSI/OPEI (Outdoor Power Equipment Institute) robotic lawn mower standard states: “ANSI/OPEI 60335-2-107-2019 (Standard) for Outdoor Power Equipment - Household and similar electrical appliances - Safety - Part 2-107: Particular requirements for robotic battery powered electrical lawnmowers (national adoption with modifications of IEC 60335-2-107).” This standard should be published sometime this fall. This standard was a joint effort creation between consumers, manufacturers, and commercial equipment users. There will be more standards that will be created in the very near future as autonomous lawn mowers grow in popularity and become more mainstream. At the end of the day, no one wants to be mowing their own lawn. However, as more competition arises from competing companies attempting to make a foothold in this market, it will be inevitable that the number of standards for these products increases in order to make it a safer product.

2.6 House of Quality

To ensure the highest possible quality, our product must achieve a balance between our engineering requirements and marketing requirements. The following House of quality diagram in Table 6 allows us to allocate a level of importance and devise a strategy on how to achieve our requirements.

Table 6: House of Quality



		Budget	**Mowing Capability**	**Remotely Controlled**	**Setup Period**	**Mowing Time**	**Form Factor**
		-	+	+			-
Cost	-	↑↑		↓	↑	↑	↑
Safety	+	↓	↓	↑↑	↑		↑↑
Efficiency	+	↓↓	↓		↑↑	↓	↑↑
Autonomy	+			↑	↑	↓	
Size	-	↑	↓		↓		↑↑
Durability	+	↓			↓		
Target for Engineer Requirements		$1500	10’x50’ lawn	50’	>=5min	>=15min	h-20”

Legend

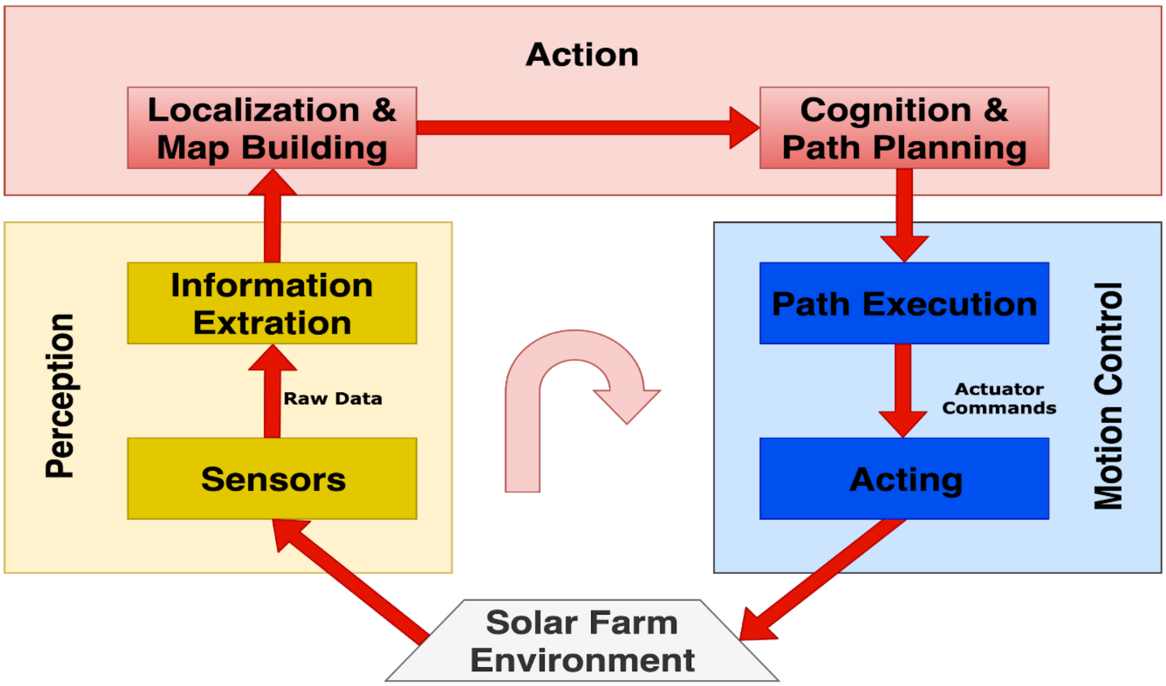
(+ Positive Polarity, - Negative Polarity, ↑↑ Strong Positive Polarity, ↑ Positive Polarity, ↓ Negative Polarity, ↓↓ Strong Negative Polarity)

As depicted in the graph, the stronger the polarity, the more important that topic is considered within the realm of this project.

2.7 High Level Control Scheme

Based on the research and previous products reviewed, we have overlaid a high-level control scheme that the rover will implement and follow along its path. This path is shown in the following table, Image 1.

Figure 1: Control Scheme



3 Applicable Standards & Design Constraints

Standards are constructed and designed methodologies, technologies, and inventions that set an established way of creating or operating new systems or technologies. Standards ensure that projects and products work smooth, efficiently, and safely with other technologies and development teams. Within this project, our team will be using a variety of standards that be compliant to domestic and international laws and regulations. We will also use standards in this project to ensure the safety of every team member and create an easy to understand technologies.

The standards we will put into place will be gathered from various Standard Development Organizations (SDOs) including but not limited to; The Institute of Electrical and Electronics Engineers (IEEE), The National Resource for Global Standards (NSSN), The American National Standards Institute (ANSI). These institutions provide guidance and resources towards the applicable standard towards the development of products and projects. The standards will be acquired through the use of visiting the databases and gathering information through documentation created by the SDOs.

Throughout the project, we will also be looking into the design impact and design constraints provided by the standards institutions mentioned above. The project aims to create a lawn rover solution that has the optimal outcome for social, economical, health, and safety reasons.

3.1 Applicable Standards

This section will describe the applicable standards that will be used in the design and implementation of the project.

3.1.1 Serial Communication Standard

To communicate between the microcontroller and the sensors and actuators, the use of several serial protocol standards will be used within the project. These standards will be discussed in detail here.

UM10204 (I2C Serial Protocol)

The UM10204 I2C Bus serial standard is a serial data protocol format that can be used in a wide variety of consumer products, telecommunication devices and industrial electronics, and so on. This form of communication takes advantage of the similarities between devices such as containing a microchip and containing some of intelligent control, and uses it to create a simplistic form of communication between devices. The IC2 standard was created by Philips Semiconductors which is now NXP Semiconductors and the standard is still held by the company.

The IC2 Bus standard aims to be simplistic as to minimize the design complexity within the communication aspect itself. A simple 2-wire bus is used and connected between the desired devices. One line acts as a serial data line (SDA) while the other line works as a serial clock line (SCL). Each connected device to the bus is addressable by a unique address given by the manufacturer and acts as a simple master/slave relationship to all connected devices at the same time. All devices can act as transmitters or receivers.

The IC2 Bus standard uses bi-directional 8-bit oriented digital serial communication. Data transfers can be executed at up to 100kbit/s within its standard limits and up to 400kbits/s within its Fast mode with options for several other higher-speed modes available. The standard also includes a collision detection and arbitration to prevent loss of data and corruption when two devices acting as master devices decide to transmit information at the same time. Noisy data information is also filtered out using on chip filtering to reject signal spikes that can destroy the integrity of the data.

The IC2 Bus standard is open to use for any project without the use of a license or intellectual property fee. For this project, we will be able to use the standard given proper accreditation to the technologies use, which this section aims to do.

3.1.1.1 Universal Serial Bus (USB) Standards

The Universal Serial Bus standard is a hardware industry standard created by the Universal Serial Bus implementers Forum (USB-IF) and enforced by the International Electrotechnical Commission (IEC) as IEC 62680 series of standards[12]. The USB standard is a serial transmission hardware protocol that allows peripheral devices with using the standard to connect to an intelligent computing system or power system. The USB was first invented in 1994 to replace the common place parallel port at the time as a means of serial communication to peripheral devices, most commonly the printer. USB provides a fast, easy, and universal way of interfacing many technologies, making one of the most commonly known used standards to date. For our project, we will also be using this standard in order to interface with several subsystems of our device.

There have been several USB specification releases since its initial release in 1996. For brevity of the standard, we will highlight the usage of the standards that will pertain to our project. The project will utilize the USB 2.0 standard created in and enforced by the IEC in the year 2000. The data throughput allowed at this standard rate is up to 480Mbits/s through a serial line bus and is available in several pin out forms. For this project, we will be using several features of the standard and two forms of the output.

The compute module and the microcontroller module of the rover project will use the USB standard to connect both the single board computer and the PCB designed by the team. In this way, we will be able to communicate through a serial protocol while easily being able to connect devices. The Single board computer utilizes the USB 2.0 and 3.0 standards with a type-A connection. This connection will then lead to another USB 2.0 connection with a type-B connection connected to the selected microcontroller. The connection used will be used for both serial communication and power supply for the microcontroller and sensory devices.

3.1.1.2 UART

Another form of communication is Universal Asynchronous Receiver/Transmitter, or UART. This type of communication is not a communication protocol like SPI. This form of communication is a physical circuit that is implemented inside devices, such as microcontrollers and stand-alone Integrated Circuits (ICs). UART can be applied to applications through hardware, enabling the transmission and receiving of data from the microcontroller or IC it is attached to. The integrated circuit can transmit serial data between two UART devices connected to each other. Because UART does not rely on a clock signal to transmit data, there has to be a program to indicate when to begin sending and/or receiving information. UART signals utilize start and stop bits in the data packets that are communicated between devices. This will indicate when to start and stop receiving or transmitting data. The data transmitted through the UART devices originates from the data bus on the circuit the UART is initially implemented on. In order to receive the data, the two UARTs in communication must be utilizing the same baud rate. This will ensure the data is not sent too rapidly, which could result in incorrect data being recorded. This means that if sent too quickly, the data could contain the risk of recording data with errors. This data would therefore not be consistent with the receiver or transmitter. UART follows the IEEE communication standards. These components will be used to communicate with the development boards and microcontrollers to each other or certain electrical components that are applicable.

3.1.1.3 NMEA 0183 Serial and Power Standard[14]

As described in section 4 and 5 of this documentation, the rover will use a GPS module to localize itself within an environment. The NMEA 0183 Standard an serial protocol and electrical standard created for localization technologies used in land a maritime uses. The NMA 0183 is a standard defined and controlled by the National Marine Electronics Association and is a replacement for the NMEA 0180 and NMEA 0182.

For our project, we will use the NMEA Output packet sequence used over a UART protocol. The NMEA output defines information and serial bit strands that represent different parts of information from the GPS module to be communicated towards other devices. This serial package is transmitted through the transmission lines from the NMEA compliant package and interpreted by the master devices connected to it. This bit “sentence” is created within the module and then transmitted over the bus. The NMEA sentence consists of several parts:

Table 7: NMEA Sentence Types

Option	Description
GGA	Time, position and fix type data.
GSA	GPS receiver operating mode, active satellites used in the position solution and DOP values
GSV	The number of GPS satellites in view satellite ID numbers, elevation, azimuth, and SNR values.
RMC	Time, date, position, course and speed data. Recommended Minimum Navigation Information.
VTG	Course and speed information relative to the ground.

These sentences start with the first word defining the data type of which they carry[10]. Each data type has its own unique interpretation within the NMEA standard and can be used for a broad range of implementations. Table X.X defines the data types that will be used within our GPS module chosen. An example of GGA sentence for the GPS NMEA is given as:

Figure 2: NMEA Sentance

```
$GPGGA,123519,4807.038,N,01131.000,E,1,08,0.9,545.4,M,46.9,M,,*47
```

With each string separated by a comma indicating a certain information field:

Table 8: GGA Sentence Description

Name	Example	Units	Description
Message ID	$GPGGA		GGA protocol header
UTC Time	64951.000		hhmmss.sss
Latitude	2307.1256		ddmm.mmmm
N/S Indicator	N		N=north or S=south
Longitude	12016.4438		dddmm.mmmm
E/W Indicator	E		E=east or W=west
Position Fix Indicator	1		See Supplementary Table
Satellites Used	8		Range 0 to 14
HDOP	0.95		Horizontal Dilution of Precision
MSL Altitude	39.9	meters	Antenna Altitude above/below mean-sea-leve

Table 9: GGA Sentence Description

Name	Example	Units	Description
Units	M	meters	Units of antenna altitude
Geoidal Separation	17.8	meters	
Units	M	meters	Units of geoids seperation
Age of Diff. Corr.		second	Null fields when DGPS is not used
Checksum	*65		
<CR> <LF>			End of message termination

3.1.2 Surface Mount Package Standards

In creating printed circuit boards and using surface mount components for this project, our project will comply with current surface mount standards that are used in industry today. There are man package types and technologies for semiconductor devices used in a broad range of different electronic solutions. The JEDEC Solid State Technology Association is United States SDO that standardizes sizes and package shapes in order to efficiently allow companies and hardware manufactures to easily choose package solutions to fit their requirements and constraints [25]. The JEDEC Solid State Technology Association standardizes part numbers in order to distinguish size and package within documentation and physical packages.

Technologies and parts that use these standards include but are not limited to: resistors, capacitors, inductors, computational chips, Op-Amps, and timing crystals. There are several standard package forms these components come in including: flat chips, cylindrical components, small outline integrated circuits (SOIC), and other forms. For example, given the part package for flat resistor of 1206, in the United States, that will correspond to a resistor size of .12 inches in length and .10 inches in height. In our project, we will be using these package numbers in order to create surface mount placements to fit the package size without interfering with other placements and electronics on board.

3.1.3 IEEE Wireless Standards

The Institute of Electrical and Electronics Engineers, aka the IEE, is an association that regulates and mandates numerous standards on the world of engineering. One of these standards set into place is standard 802.11. This standard is extremely important to this project as most of our communication with the autonomous robot will be done wirelessly. Products that implement this standard must pass tests in order to be considered, or deemed, “Wi-Fi certified.” The Wi-Fi most commonly used is of the 2.4 GHz wireless spectrum.

3.1.4 Economic Constraints

The cost of designing and implementing the required components to realize the project at hand, an autonomous lawn mower, can be slightly daunting. There are many pieces of the puzzle that must fit and work together as a perfect mesh of engineering in order to achieve an autonomous robot that can move on its own (without the use of any external aid), as well as cut grass effectively, while being able to navigate and avoid obstacles. These feats require equipment that can cost quite a bit of money. Luckily, this project is sponsored up to $1500. With this in mind, there is a bit of leeway in the parts list category. This means that we can pick and choose different sensors, motors, wheels, motorheads, printed circuit boards (PCBs), etc.

However, even with a large budget on our senior design project, we must still be mindful that parts can be dead on arrival (DoA), or even break during the testing or construction phases. Because of this, we have to budget the parts required accordingly in order to keep a safety, or back-up, budget in the event any part breaks or is bad. This research paper alone will go through and list out all the parts we have looked at as a group, as well as pit them against each other for scores on what will be the most beneficial for our specific project design.

Another economic constraint to consider is that because we are Phase 2 of a three-phase project, next year's senior design candidates that partake on this project will be able to build off our current project and hopefully design something even more effective than our design. This furthers into our client being able to mass produce these autonomous lawn mowers for use on the desired solar farms as posed by the client's request in this project. Alternatively, this project can be used by the client much further than just for mowing solar lawns, but for their own business properties, as grass is everywhere. It is for this reasoning, being able to mass produce these autonomous lawn mowers, as well as the reasoning for the potential of parts breaking, that we are trying to keep the cost as low as possible while designing, building, and testing our autonomous lawn mowers.

3.1.5 Environmental Constraints

As a group, we are trying to keep this project as "green" as possible. The entire theme of this project is to aid in the ease of cutting grass in solar farms, a very "green" source of infinite energy. With that in mind, we will not be using any gasoline or fossil fuel to propel our autonomous lawn mower, nor give it the energy needed to move or cut grass. Instead, we will be using a battery pack, which will be able to be recharged at any outlet or a charging station (design pending). The energy being used to recharge this battery pack will most likely be pulled from a building's power grid and converted into the energy our autonomous lawn mower will need. However, this energy being used from the buildings, will most likely be "dirty" energy. This means that the energy being siphoned will most likely be supplied from a city's coal or nuclear power plant. It is our wish to try and improve on the design by the end of next semester to achieve a solar-powered-recharging autonomous lawn mower, in order to not only prevent the use and production of more dirty energy, but to also make our autonomous lawn mower full self-sustaining.

Furthermore, the wheels and motorhead being used to navigate the autonomous lawn mower and cut the grass respectively, will require greasing or oiling from time to time in order to ensure smooth movement for clean cuts on the grass. This means that over time, there will eventually be some grease or oil that will run off onto the grass, and with precipitation, will eventually run off into nearby rivers, soils, streams, and/or sewer systems, contaminating local water sources. The contaminated grass could even be ingested by animals such as birds, worms, etc. that will be ingested by larger animals and eventually end up on our tables in the meat that we eat, aka a butterfly effect. We are doing our best to try and limit the amount of oils required to operate the autonomous lawn mower, but at the end of the day, only so many measures can be taken to prevent such an effect from happening.

3.1.6 Social Constraints

The primary goal of the autonomous lawn mower system is to eliminate the need for lawn service companies to have to come out to solar farms, and struggle to cut the grass around the cornucopias amounts of support beams that hold up the solar panels themselves, and also help route the cables of power conversion to the generators. Because of this, there will be less of a need for lawn service companies, reducing potential jobs from these people. As lawn service is seen as more of an entry level position in the current day, requiring no prior work experience to operate a lawn mower, this will eliminate the need for these entry level job positions. Once these entry level positions are terminated, it will be harder for individuals to seek a higher-level job, as in order to acquire a higher tier position, one must obtain lower tier work experience first. Less jobs available will create a higher unemployment rate as well.

3.1.7 Political Constraints

As mentioned above, the political implications of this project being completed and realized will mean the reduction of entry level job positions. This will result in higher unemployment rates for any state that has a solar farm that wishes to utilize this technology.

3.1.8 Ethical Constraints

As stated previously, the main goal of this project is to create an autonomous lawn mower that will cut grass for solar farms. As we are trying to aid in green energy consumption, we as a team take remaining ‘green’ very seriously. As such, we are ensuring that no unethical methods are being done in the building of this project, such as insecure power connections to save money and cut costs. Everything will be built securely and safely. The autonomous lawn mower being built will not use any potentially toxic products, nor will corners be cut in the process of building this robot. We as a team are making sure that not only the autonomous lawn mower system will work and maintain the client's wishes, but also that it will be designed to work and operate safely in accordance with all the standards put in place for the parts required to build and realize this device.

As far as patent protection goes, extensive research is being done to avoid any infringement on any existing patents on other autonomous lawn mower devices already on the market. Any protected concepts or designs used in the implementation of the design and operation of our autonomous lawn mower will properly attribute any applicable credit to the respective owner of that design.

3.1.9 Health and Safety Constraints

Health and safety typically coincide with one another. It is difficult to have one without the other. One of the largest and most important safety constraints within this project is one given by the client. This constraint states that the autonomous lawn mower must be able to kill the power if for whatever reason a kill switch is hit. This means that if anything is wrong, or if the autonomous lawn mower does not detect an important obstacle in its path, it must be able to shut off completely once this remote button is pressed. This single safety constraint will lead to a plethora of safety checks, as instantly killing the power on the autonomous lawn mower will ensure safety in any situation. With this in mind, we are ensuring that the switch location for the logic on the PCB is in a safe location, hidden and protected from any potential water damage that could occur, as if the power kill switch is faulty, it could potentially result in a hazardous result for the equipment on the solar farm, or worse, injury for a human.

Another very important health and safety constraint for the autonomous lawn mower is to have some sort of sensor to be able to detect where it is and where it is going. As the robot will be moving on its own, it is mandatory that it have some form of vision attached to its body. Without vision, it has the potential to run into objects that it should not be getting close to. This could end up creating very hazardous situations for the client and its employees. With a sensor, or camera, attached to the body, the autonomous lawn mower will have a far superior sense of direction and will result in few, if not eliminate, the potential risk of collisions with the objects around it.

Additionally, the autonomous lawn mower will be powered by electricity. As with all electrical equipment, this always implies certain implications to the operation of this device to ensure a safe work environment. It is imperative that one operates this device safely in order to avoid potential electrocution or loss of limbs, as the autonomous lawn mower will indeed have a motor fast enough to cut grass with a zip tie. This means that the autonomous lawn mower has the potential to severely injure someone if they misuse or misappropriate the autonomous lawn mower.

If operated correctly, and with the addition of some security features that we are programming into the autonomous lawn mower, the risk factor in operating and controlling the autonomous lawn mower will be drastically reduced. However, it is still imperative that those operating this device maintain a safe distance from the blade and maintain correct usage of the device in order to avoid injury.

3.1.10 Time Constraints

Time constraints on this project are very closely monitored, as there is a deadline on this project. That deadline is to be fully completed with both research and building of the autonomous lawn mower by the end of Senior Design 2. This project is meant to last two semesters, Fall and Spring semester. Constant work is required to be done both researching and prototyping. A design that takes a long time to implement or debug could prove catastrophic to the timeline and success of this project. There are many additions and interface options that could be added to the activation and controlling of the autonomous lawn mower design, however, many of these features would extend outside the scope of this project, as they are not critical and may take too long to implement into the system we have designed and are building for the client. Should time permit, we will add as many features as possible into this project without jeopardizing any key features and functionalities. This means the core functions will take priority in being implemented.

By following the strict project timeline, we will have more time to address any issues that may arise during the prototyping stage. This will give us a more relaxed experience in dealing with issues, eliminating any stress from any last-minute implementations of parts. The timeline of this project can be found in Section 7 of this document, found on page 94. This section will explicitly describe the path the project took, as well as the number of days dedicated to each member of the group’s task at hand. We have already ordered the majority of the parts needed to begin prototyping the autonomous lawn mower and are therefore ahead of our time constraints.

3.1.11 Testing/Presentation Constraints

The primary presentation constraint regarding the autonomous lawn mower will be the competition that will take place between all the teams competing. There will be a course designed by the client for each of the autonomous lawn mowers to compete in. This involves a multitude of tests and terrains to traverse, while providing a smooth and clean cut of the grass. There will also be a pre-competition held at the school of UCF, most likely to occur in front of the HEC building (location currently decided as of 12/1/2019). This pre-competition will be hosted by UCF and the teams competing will be all of the UCF teams, whereas the official client’s competition will be hosted at the Disney Solar Farms and will contain teams from multiple schools. Once prototyping is completed, our group will be performing our own tests on the autonomous lawn mower prior to the pre-competition and the official competition itself.

4 Research

Prior to building a prototype, research must be conducted in order to find the technologies and solutions that will satisfy our requirements. In doing so, we consolidate all the possible solutions that meet the functionality that we desire and compare the solutions in between one another. After this process, parts are selected and documented. This chapter of the document details the process of exploration and comparison between different parts and solutions that have been researched for this project.

4.1 Relevant Technologies

In this section, we will discuss the relevant technologies that are incorporated into our rover. The rover will use several mechanical, electrical, and computational technologies that we are familiar with today. We will learn more and dive deeper into the use of these technologies through this section.

4.1.1 Motors

There are many important components required to realize a fully autonomous lawn mower. One of these core components include, but are not limited to, the motors. Motors are required to pass a multitude of restraints. The most notable of usages motors will grant, is the ability to move. Each individual wheel has to be attached to a motor. For example, the very first robotic project most engineering students at UCF will work on is called the “Bo-Bot.” This robot introduces students to the servo motors used to spin the wheels and allow the robot to navigate, eventually leading up to an autonomous robot that can navigate an obstacle course, similar to this project. A motor is also required to spin the string that will be cutting the grass. If the motor does not have a high enough cycles per minute, the chances of the grass being cut uniformly will be low. Motors are an integral portion of fully realizing an autonomous lawn mower. This section will explain the research behind the top motors we looked at and compared as the final motors to be used in our project.

4.1.1.1 Trimmer Motors

The first thing we had to figure out when it came to the trimmer motors was if we wanted to buy a commercial weed-whacker head (we would need at least two) or if we wanted to buy the motor ourselves and attach the line. After doing a comparison as seen in the figure below, we decided to go with a motor that we bought ourselves. It would be cheaper, and we could get exactly what we wanted out of the motor. Once that was decided, we let the mechanical engineers decide which ones to pick. For trimmer motors it's pretty simple, you just need a motor head that is capable of spinning at least 6,000 rpm, which is the minimum speed necessary to be able to cut grass with weed whacker line. So that's exactly what we picked, If you go much faster than what is sufficient to do the job then you are just wasting battery and that's not good practice.

Table 10: commercial vs custom mower head

Criteria	Weight	Evaluation			Score		
		Ryobi one+ 18v 	Jiaruixin RS550 	MD5-2450 	Ryobi one+ 18v 	Jiaruixin RS550 	MD5-2450 
Speed	(5)	1	2	3	5	10	15
Cost	(3)	2	3	1	6	9	3
Size	(2)	1	3	2	2	6	4
Total score					**13**	**27**	**22**

Table 11: comparing custom trimmer motors

Criteria	Weight	Evaluation			Score		
		RS550 	JD3-24135-CVC 	MD5-2450AS-AA-C 	RS550 	JD3-24135-CVC 	MD5-2450AS-AA-C 
Speed	5	2	1	3	10	5	15
Size	3	3	1	2	9	3	6
Cost	2	3	1	2	6	2	4
Total Scores					**25**	**10**	**25**

4.1.1.2 Wheel Motors

For the wheel motors, we as a group are less interested in speed, and more interested in torque. Since the mower is only going 2-4mph but has to pull around 30lbs. Again, for these motors, we let the mechanical engineers decide which ones we needed. They did another comparison and found the one that matched our torque needs, along with our budget limitations. This motor will also be accurate enough to use for navigation. The wheel motors will be attached to the frame of the mower and will be driven by the motor driver which will in turn be driven by the microcontroller.

Table 12:Comparison of Driving Motors

Criteria	Weight	Evaluation			Score		
		A28-150	E30-150	F30-150	A28-150	E30-150	F30-150
Power	3	3	1	2	9	3	6
Cost	5	1	3	2	5	15	10
Size	2	1	3	2	2	6	4
Total Scores					**16**	**24**	**20**

4.1.1.3 Lidar Motor

The Lidar motor is simply used to rotate the head. We can adjust the rotation of the head to get the lidar to the level of resolution we want as well as saving power by not having the lidar head spin full speed when doing so is unnecessary.

4.1.2 Power

In this section, we will review the power sub systems and how it will be distributed through out to each individual component. We will also see the power consumption needs of all of the devices and what batteries we will use to power them.

4.1.2.1 Power Requirements

The power requirements for this mower will be a summation of the needs of all the individual components. For simplicity sake I have broken it up into three different sections: wheels, trimmers, and accessories. The wheels will be the most difficult to account for power consumption because they will be taking different levels of power depending on what maneuver they are doing. To keep things simple, we will just assume the worst-case scenario in power consumption and say that they are in 100% motion the whole time. The trimmers will be easy to calculate because they will be drawing the same amount of power the whole time they are on (which will be a good amount but easily calculable). The last of the three sections is the accessories, which includes the Lidar, Camera, Microcontroller, and Odroid. These components are much lower power than the other ones and will be at a different voltage so they will get their own battery to run off of.

4.1.2.2 Battery Selection

When it comes to battery selection, we obviously want something small and compact but also something power dense too with a good amount of amp hours. An obvious choice is Lipo batteries as they are very power dense, and we can fit multiple batteries in the mower. In addition to the Lipo batteries, we will also have to buy a charger for them and a fire sleeve for them to charge in. The voltages we will be working in will only be 12v and 5v so we can easily find Lipo batteries at those voltages. As we figure out how much each section draw in current, we can combine the batteries in parallel to double the amp-hours. It just so happens that the previous year's rover had batteries in it which are perfect for our application. At 11.5V and 8000mAh, they will be perfect to power the wheel motors and trimmer motors. We also have another 12V battery that is very large and could be used to power the trimmer motors or the wheel motors all by itself. There is no serial number or other information on the outside of the battery but going by size and manufacturer, it is fair to estimate that its amperage hours is somewhere in the ballpark of 15 or so.

4.1.2.3 Distribution

So, in total, we will have three batteries. One for Wheels, One for trimmers, and one for all the electronic accessories. We plan on making the inside of the mower clean with good cable management (the last years was a rat's nest). We plan on using wire looms were applicable and using zip ties. We want to be able to easily identify what parts are connected in the case of something wrong happening with the mower so we can easily remedy the problem. In terms of wire size, we will have to see what the current needs are for each particular connection and then select the correct gauge wire accordingly. We are also considering using Velcro in order to secure all of the electronics and wiring to the frame of the mower while still allowing us to move and change parts out. This should help with the organization and safety. The wiring being well labeled and easy to see where connections are being made will help us easily see where problems may be occurring as well as be able to easily replace parts.

4.1.3 Single Board Computer

This section describes the use of the research and part selection of the single board computer subsystem within the rover project. In general, this subsystem is used for high-level software implementation, image processing, and path finding and planning. The single board computer is critical to the autonomy of the rover and care in research and part selection is crucial to the success of this project.

4.1.3.1 Single Board Computers Overview

A Single Board Computer (**SBC**) is a computer system built on a single circuit board. They are made of microprocessors, inputs and outputs, memory and several other components. Single Board Computers are used across a wide variety of different industries, like educational systems, development systems, or used as in embedded computer controllers. They are used in different homes and devices that we use every day.

A Single board computer can allow us to implement code to be able to control the hardware devices of the autonomous lawn mower. One of the good things about owning a single board controller is that it can hold various kinds of Operating systems (OS) like Embedded LINUX/Windows, Desktop LINUX/Windows, RTOS, UNIX, Sun Solaris and more. Engineers can use SBCs to do embedded type of applications using the C programming language without the need of an OS.

A single board configuration reduces the overall cost of the system by reducing the required number of circuit boards and getting rid of the use of bus driver connectors and circuits in other cases. One of the advantages of using Single Board Computer is that a number of different components are built into it. They normally come with Bluetooth, WIFI, HDMI, Ethernet, and an audio adapter. This facilitates the user in many different ways. With a Single Board Computer engineers can attach it to an external monitor and use it as an interface to control devices. The Bluetooth/Wi-Fi capability also helps with controlling our devices over the airways, without the need to use a physical wire.

Most Single Board Computer also have internet capability that allows the user to connect to devices and be able to control without having to be in the vicinity, giving us the ability to control it anywhere in the world. With all those attributes supported by the Single Board Computers, we can have simple interface for navigating, organizing and executing programs.

4.1.3.2 Single Board Computers Needed

For the purpose of our project a really good Single Board Computer will be needed. We will need a Single Computer Board that can support high processing power demands of computer vision and image processing and also Wireless communications. We will need one that work in synced with our chosen microcontrollers allowing us to test the microcontroller that will be implemented onto the custom-made PCB for the overall grass cutter system. We will be able to test several attributes before we go ahead in applying it to our final PCB design.

The Single Board Computer can support testing of a wide variety of electrical components, like motor driver chips, GPS module, camera, ultrasonic sensors, and voltage regulators. We will be implementing the code needed to operate the hardware components using familiar programming languages like C, C++, and Python.

For the navigational system, a camera will be used. The Single Board Computer selections will be displayed below, where I will elaborate more on the different types of Single Board Computers that are on the market. Because of all the different types of SBCs on the market we decided to come up with criteria of different sorts to help narrow down the selection of options that are available. We narrowed it down to 4 important criteria with different weights in importance.

One of the criteria that we really care about is the Single Board Computer's processing power. This criterion was our most important criteria as we needed one with really good processing power to be able to handle the amount of work that'll be needed for this project. There will be several different tasks that will need to be done ranging from controlling the motor, electrical components, computer vision, and many more of other tasks.

The second criteria that was the second most important was the power consumption of the Single Board Computer. We will be needing Single Board Computer that will not consume a lot of power as we will not utilizing a built in solar on the lawnmower, so we will need to be careful about our power consumption, as the autonomous lawn mower will be expecting to run for quite some time.

The third criteria we had was about the connectivity and storage. We want a really good Single Board Computer that can be used to connect several different inputs devices to it without any trouble. The Single Board Computer should be able to handle and process such devices all at the same time. It should also have great memory storage capacity to accompany its arrays of ways to connect to it. Our ideal board would allow to process the required demands from the various sensors we will be using without any troubles or lags. This is huge for us if we want to obtain any kind of success in this project.

The last criteria and least important criteria of our must-haves are the cost. The cost of the Single Board Computer will play a huge factor in our decision making. With the limited funds given to us, we have to make to keep in mind the cost of components. We have allocated funds to different disciplines, leaving us with not a lot of money to waste. The table below will show the 3 different Single board computers that made the cut and I will also go in details about their respective specifications.

4.1.3.3 Raspberry Pi 3 B+

The Raspberry Pi 3 B+ runs at 1.4 GHz with 1GB of memory. The cost of a Raspberry Pi is relatively cheap at a price of only $35. It has the capability to support the camera as an input, as well enough Inputs and Outputs to support the different sensors are required for this complex project. The Raspberry Pi 3 B operates at 5V and contains 4 USB 2.0 ports.

The Raspberry Pi 3 B + supports Wi-Fi and BLE protocols. The Raspberry Pi 3 B+ has a Quad Core Broadcom BCM2837 64-bit ARMv8 processor. The Raspberry Pi 3 B + has an integrated GP-Video Core IV that runs at 400 MHz. That's a really good processing speed which would make this single Board Computer a really great selection for this project as the robot will have a demanding computer vision architecture to make important navigational decisions. The power consumption is relatively good as it does not consume too much power for the task that it would handle.

As you would expect from a Single Board Computer, it can be programmed in numerous different programming languages including C, and C++. It comes with its own operating system, called Raspbian OS. It also has the ability to support other operating systems as well, which will be discussed in the next subsection.

4.1.3.3.1 Operating System consideration for the Raspberry Pi 3B+

There are a lot of different operating systems choices available for the Raspberry Pi 3 B+. We narrowed it down to 7 different operating systems that we initially would take into considerations in case we choose to go with the Raspberry Pi 3B+. The choices of Operating systems were:

- Raspbian
- Ubuntu Mate
- Snappy Ubuntu Core
- OSMC
- LibreElec
- Risc OS
- Windows 10 IoT Core

4.1.3.3.2 Raspbian

Raspbian is a Debian-based computer OS specifically for Raspberry Pi. The Raspbian is known for its fast performance in floating point arithmetic operations. It is used to help with calculations and aids in the optimization of code for the Raspberry Pi 3B+.

4.1.3.3.3 Ubuntu Mate

Ubuntu Mate is a community developed operating system that uses Linux as its core system, while at the same time being able to provide a desktop computer-like Graphical User Interface (GUI).It would work perfectly with the Raspberry Pi as Ubuntu Mate’s original design was to support for computers like the Raspberry Pi.

4.1.3.3.4 Snappy Ubuntu Core

Snappy Ubuntu Core is the smallest Ubuntu ever. It is a minimalized version of Ubuntu which enables machines like the Raspberry Pi to run it without any problems. It is cloud based so it has the ability to run remotely upgradeable Linux app packages known as snaps.

4.1.3.3.5 OSMC

OSMC (Open Source Media Center) is a free and open source media player based on Linux that can be used on the Raspberry Pi. After doing some more research on this operating system, we have concluded that it would not be a good fit for our specific project.

4.1.3.3.6 LibreElec

LibreElec is a Linux based operating system that eliminates several of the features in order to maximize the hardware potential and at the same time having enough performance to get the job done. It focuses on multimedia services, which is not a target that we are looking for, so we would not select it.

4.1.3.3.7 Risc OS

Risc OS is a computer operating system originally designed by Acorn Computers Ltd. The primary purpose of its creation was for it to be a fast and easily configurable operating system for ARM microprocessors.

4.1.3.3.8 Windows 10 IoT Core

Windows 10 IoT Core is an operating system built for your internet of things. It was designed by Microsoft to be utilized in embedded systems. After doing further research we found that the Windows 10 IoT Core would not be a great fit as it requires licensing for a project of this magnitude that we are doing.

4.1.3.4 ODROID-XU4

The ODROID-XU4 is powerful Single Board Computer and energy efficient hardware. It is one of the fastest of the three Single Board Computer that we are considering. The ODROID-XU4 support open source, which allows to run various versions of Linux including the Ubuntu 15.4 and Android 4.4 KitKat and 5.0 Lollipop. The ODROID-XU4 has extremely amazing data transfer speed that complements its already powerful processing power of 8 arm CPU cores and. This is a very important aspect for this project, which will contain a lot of computing and data transfer for the navigation and wireless communications.

The ODROID-XU4 has HDMI 1.4a to allow it to connect to external monitors. The ODROID-XU4 has an output resolution of 1080p. This is remarkable output specs and may be helpful to us in the future. The ODROID-XU4 has a pretty good GPU, which is ARM Mali T628. It will be more enough for the needs that we may have for it if we choose the ODROID-XU4.

The ODROID-XU4 comes out on top when compares with other eight cores Single Board Computers out the market. It is one of the mature Single Board Computers, and the ODROID-XU4 is able to run the mainline Linux kernel. Documentations and supports us widely available for the ODROID-XU4 as it has a really big community of people that is willing to help and get beginners started on their project. So, if we get stuck, we would have plenty of people to help us out.

In table 10, we compared the amount of power consumption that the ODROID-XU4 requires. This comparison is made in both terms of Amps and Watts. This helps explain how much power we can be expected to require if we use this board. We have limited about of power available and we do not want to spend all of our power on the Single Boad Computers

Table 13: ODROID-XU4 Power Consumption

ODROID XU4 power	Amps	Watts
Idling – login	0.422	2.1
Idling – desktop	0.838	4.2
Chromium – desktop	1.280	6.4
Chromium – Roar 720p	2.633	13.2
Kodi YouTube – Roar 720p	1.696	8.5
Kodi Veevo – Roar 1080p	1.914	9.6

The power jack supports up to 5v at 4A. As you can see from Table 10. Under normal operations the ODROID-XU4 consumes about 2.1W of power and can go up to 10W depending on the number of USBs attached and computing loads is being done at the time. This is a very interesting aspect as we would want a Single Board Computer that would not consume a lot of power. The ODROID-XU4 is also equipped with a cooling fan which prevents from overheating.

4.1.3.5 Nvidia Jetson TX2

One of the aspects that made us consider the Nvidia Jetson TX2 was its powerful computing power. It is the most power-efficient embedded AI computing device on the market currently. The Nvidia Jetson contains a Dual-Core NVIDIA Denver 2 64-Bit CPU, alongside with an 8GB 128-bit LPDDR4 Memory, making a really supercomputer like device. Obtaining a device like the Nvidia Jetson TX2 would be more than capable of handling the tasks needed. In comparison with the other Single Board Computers we were considering, it should be in a class of its own when it comes to computational power.

The Nvidia Jetson TX2 consumes about 7W of power when idle but can go up as high as 15W of power. It does consume a lot of power as we would expect from a SBC of this magnitude. This makes Nvidia Jetson TX2 a really bad choice for us when we are considering the amount of power consumption as an important criterion. Due to its powerful CPU, the Nvidia Jetson TX2 comes at a really hefty price of $450. Another area where the Nvidia Jetson TX2 had a disadvantage, as the funds for this particular project is limited. Overall the Nvidia Jetson TX2 would have been a really great fit for our processing needs but is too expensive and consumes too much power for this project.

4.1.3.6 Asus Tinker Board

The Asus Tinker Board is a really interesting board. Its appearance resembles that of the Raspberry Pi 3, but it is quite different especially in the software aspect of it. The Asus Tinker Board is consisting of icons depicting its various functions, in order to avoid mixed up between camera and display connectors. The inputs are color-coded to facilitate the distinctions of the different pins. An example of that is having a red pin for the 5V and black for the ground.

The Asus Tinker Board is equipped with a really good system-on-chip (SOC); it has a quad core Rockship RK3288 running at 1.8GHz. The Asus Tinker Board has 2GB of RAM, which is really good for us since our chosen Single Board Computer will need to be able to handle large amount of data at the same time. It has a really good output display resolution. It can support 2160p resolution. With this kind of resolution, we can get a lot of things done. The GPU performance is truly remarkable and one of the things that stood out to us when we decided to do research on it. It also has several USB ports to help us connect to it the various sensors that we will be utilizing.

Tinker OS is on the best attribute about the board. As you will see in later testing, that the software is just as important as the hardware when it comes performance. A really fast software can make up for its hardware limitations by having a really fast software. The Tinker OS comes with several pre-installed that we would find useful if we are to choose this specific board. It is based on a Linux so that’s a great plus to us for us to transition to. It does have some drawbacks to it . For example, the security password initially given to it is easily found online, making it easy to get hack, and until you change the password it leaves you at a great risk of being hack. Another drawback I found was that it is in its infancy stage and new, so it is not as easy to be found really good documentation for beginners.

The power consumption is relatively good as you can see in Table 11. The Asus Tinker Board while Idle does not consume any power. When the Asus Tinker Board is using chrome casually without watching videos, it uses a pretty significant amount of power. When watching high quality videos, for example 720p videos, the power consumption ramps up to 6.7 watts. We can see in the table below the maximum power consumption while handling high task is 7.1 W. This makes the Asus Tinker Board a pretty good candidate when we take power consumption into consideration.

Table 14: Power consumption of Tinker

Tinker power	Amps	Watts
Idling – login	0.340	1.7
Idling – desktop	0.625	3.125
Chromium – desktop	1.1	5.5
Chromium – Roar 720p	1.340	6.7
Kodi YouTube – Roar 720p	1.397	6.985
Kodi Veevo – Roar 1080p	1.420	7.1

4.1.3.7 Single Board Computer Comparisons

In this section we will compare the Single Board Computers and see how they will hold up against each other. We will be discussing the several different hardware components and also the software side and their differences. I will show the different results I have obtained as own the three different that I will be going over in this section. We decided not to use the Nvidia Jetson for the section as I do not own one. The Nvidia Jetson is too expensive and after wanting to narrow it down even more, we choose to go with the Asus Tinker Board, Raspberry Pi 3, and the ODROID-XU4.

4.1.3.7.1 Price and Power

We will discuss the cost of the three Single Board Computers that made the list. As you can observe in the figure below, the ODROID-XU4 is priced at $59, Asus Tinker Board at $55, and the Raspberry Pi 3 at $35. As you can the Raspberry Pi 3 is priced at very low of $35. The ODROID-XU4 and the Asus Tinker Board are around the same price, of around $4 of each other. The Asus Tinker Board and the ODROID-XU4 is twice as much as the Raspberry Pi 3, which makes us think what we are getting extra in the ODROID-XU4 and Asus Tinker Board.

Figure 3: SBC Price Comparison

The Raspberry Pi 3 comes with 4 1.2 GHz A53 ARM Cortex. The Asus Tinker Board comes with 4 1.8 GHz A17 ARM Cortex. There is more power coming from the Asus Tinker Board than the Raspberry Pi 3. The ODROID-XU4 is made up 8 cores, 4 of them are 2.0GHz A15 ARM Cortex, and the other 4 are 1.6 GHz A7 ARM Cortex. The ODROID-XU4 has a lot more processing power than compare to the Asus Tinker Board and the Raspberry Pi 3. How we will perform in actual test will be determine later in our comparison section.

In terms of graphics, all of the chosen Single Board Computer have a GPU on a System on a Chip. The ODROID-XU4 has an ARM Mali T628. The Raspberry Pi 3 has Video core IV. The Tinker has an ARM Mali T760.The ODROID-XU4, Asus Tinker Board, Raspberry Pi 3, all have an HDMI sockets to connect to an external display. On the ODROID-XU4 and the Raspberry Pi 3 can output at up 1080p. The Asus Tinker Board can output at an incredible 2160p, which is considered as 4K. So, if high resolution is the goal for a project, the Tinker would steal the show.

In terms of memory, we have quite the variety. The Raspberry Pi 3 has a 1GB DDR2. The ODROID-XU4 has 2GB DDR3. The Asus Tinker Board have a 2GB DDR3 as well. This gives us a basic idea of the power of these Single Board Computers. Table 12 compares the boards previously listed, depicting the price, CPU spec, GPU spec, Video Output resolution, and the memory equipped on the board.

Table 15: Single Board Computer Comparison

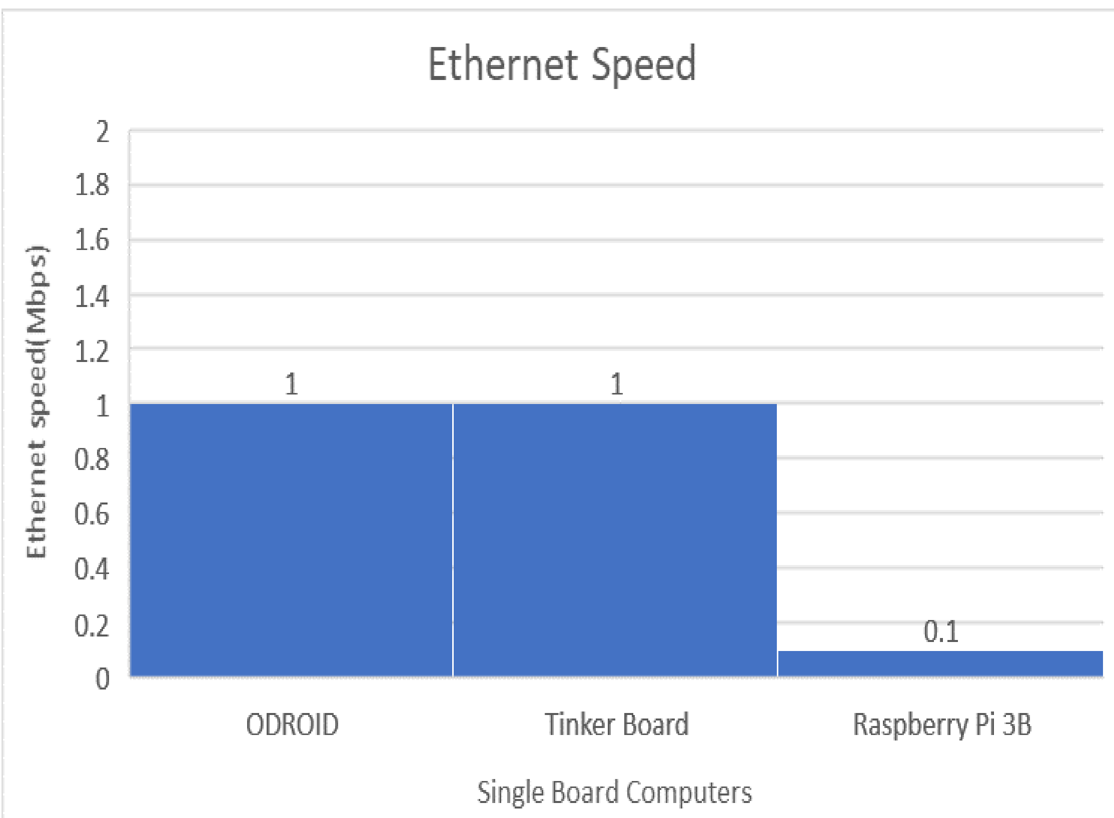
FEATURE	**ODROID-XU4**	**Asus Tinker Board**	**Raspberry Pi 3**
PRICE	$59	$55	$35
CPU	Exynos 5422 4x2.0GHz A15 4x 1.6GHz A7 ARM Cortex	Rockship RK3288 4X1.8GHz A17 ARM Cortex	Broadcom BCM2837 BCM2837 4x 1.2 GHz A53 ARM Cortex
GPU	ARM Mali T628	ARM Mali T760	Videocore IV
Video Output Resolution	1080P	2160P	1080P
Memory	2GB DDR3	2GB DDR3	1GB DDR2

4.1.3.7.2 Connectivity and Storage

So, having discussed about the processor power, graphics power and RAM, we will now be discussed connectivity. On the Raspberry Pi 3 we have 4 Universal Serial Bus 2.0 ports. We also have on the Asus Tinker Board 4 Universal Serial Bus 2.0 ports. As for the ODROID-XU4, 1 USB 2.0 port and 2 USB 3.0 ports. That's a really good thing about the ODROID-XU4, it has a Universal Serial Bus 3.0 port , that's is crucial to have on a Single Board Computer. It came at a cost because on the ODROID-XU4, we only have 3 Universal Serial Busses, compared to the 4 Universal Serial Busses that is on the Asus Tinker Board and the Raspberry Pi 3.

All the board have Ethernet sockets on them. The ethernet sockets are meant or the user to connect to the internet and transmit data to the Single Board Computer. The rate of transmission of the different boards are important to take in consideration if we ever to use ethernet for this project. The Raspberry Pi 3 has a 100 Megabit Ethernet. The ODROID-XU4 has 1 Gigabit per second Ethernet speed. The Asus Tinker Board has the same ethernet speed of 1 Gigabit per second.

Figure 4: Single Board Computers Ethernet Speed Comparison



We might also want wireless connectivity to help with controlling of the autonomous lawn mower. The Asus Tinker Board has WIFI capabilities which is a good thing to have on Single Board Computer. The Raspberry also has WIFI in it to facilitate us with the controlling and sending data. The Asus Tinker Board has Bluetooth capabilities in it. The Raspberry Pi 3 also has Bluetooth built in. The ODROID-XU4 does not support WIFI which is a downside to it, as we might want to send data to it using WIFI.

The ODROID-XU4 also does not support Bluetooth, which we found to be a major drawback as we wanted that feature to be included. As we know Bluetooth would allow us to pair with lawn mower and allow a strong and reliable connection. If we wanted to connect by wireless signal for the ODROID-XU4, we would have to purchase extra accessories to connect to it. This will cost us to use a Universal Serial Bus ports which is bad since the ODROID-XU4 only has 3 Universal Serial Bus ports, limiting us to use 2 Universal Serial Bus ports. With the other two Single Board Computers, we would still have 4 Universal Serial Bus ports available if we decided to use WIFI.

If we were to add a keyboard and a mouse to the ODROID-XU4 while using the adapter for wireless communication, we would be left with no available USB ports for us to use. With the Asus Tinker Board and the Raspberry Pi 3 presented the same scenario would still have 2 USB ports available. The amount of USB ports matter to us as we will have a lot of sensors connecting to the Single Board Computer at the same time. Having no WIFI and Bluetooth capabilities and only 3 Universal Serial Bus ports is marked against for the ODROID-XU4.

We will now discuss the Single Board Computers and how they boot up and how they store their respective operating systems. The Raspberry Pi 3 stores its operating system on the Micro SD card. The Tinker also holds its operating system on the MicroSD card. On the ODROID-XU4 we have some flexibility on how we would want to boot up our operating system. We can use the MicroSD as well as the eMMC. It is a lot faster to boot up from eMMC than it is on the MicroSD card. With the ODROID-XU4 allowing us to make a choicer is a plus. The ODROID-XU4 has a switch to decide where you want to boot the operating system from This should give the ODROID-XU4 a really big advantage in certain times of performance tests.

Table 13: Single Board Computer Connectivity Capabilities

FEATURE	ODROID-XU4	Asus Tinker Board	Raspberry Pi 3
USB 2.0 ports Support	Yes 2 available	Yes 4 available	Yes 4 available
Number USB 3.0 ports	Yes 2 available	NO	NO support
Ethernet speed	1Gbit	1Gbit	100Mbit
WIFI support	No support	Yes	Yes
Bluetooth support	No support	NO	NO
OS (manufactured)	Ubuntu Mate	Tinker OS 2.0.3	Raspbian

As you can see from Table 12, you can see how the different Single Board Computers stacked up against each other, when it came to discussing their connectivity capabilities. We also discussed the various way to connect to the board and how some may limitations when you want to add additionally to it. We also discussed the various

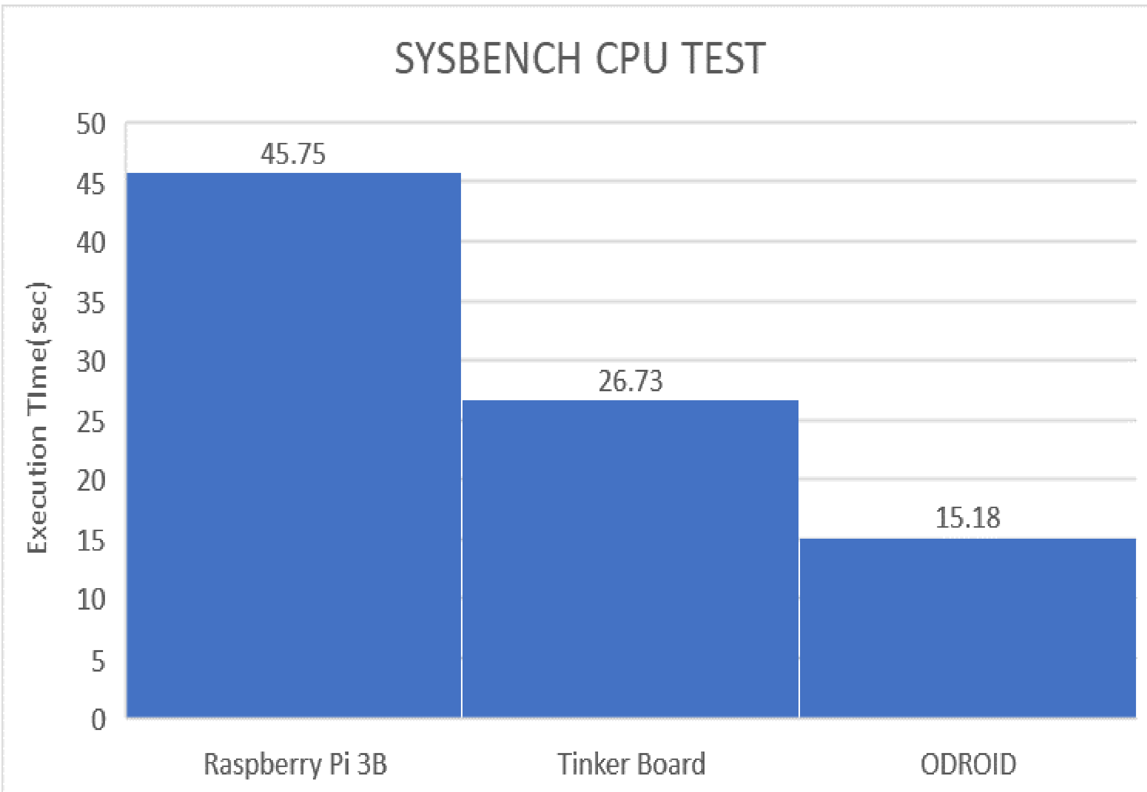
4.1.3.7.3 Benchmark CPU Test

In this section we will discussing the results of the Single Boards Computers with different benchmarking and test. We will be displaying how fast the Single Board Computers really are. The testing will be down to check out and test real world use. We will use different kind of test display the data and analyze it.

The first form of testing will be done with the Sysbench. I downloaded Sysbench on all of the Single Board Computers to well compared with each other. SysBench is a benchmark suite which allows to quickly get an impression of system performance which is important if you plan to run a database under intensive load. We will put the CPUs to the test and see how their processing power really work.

On the ODROID-XU4, you will see in the figure below how its CPU respond with the task being. We will see the execution and we will analyze it. The ODROID-XU4 has 8 cores, so we will use 8 threads corresponding to each core. The test to be done by the CPU is to hit all the prime factor up to value of 10000 matches is a good way to test as it will stress out all the cores and see how the CPU operates at high capacity. The Tinker uses 4 cores to do its computations and the Raspberry Pi 3 is the same way. In the figure below I show how the Single Board Computers did on this test. Their respective execution times are listed below.

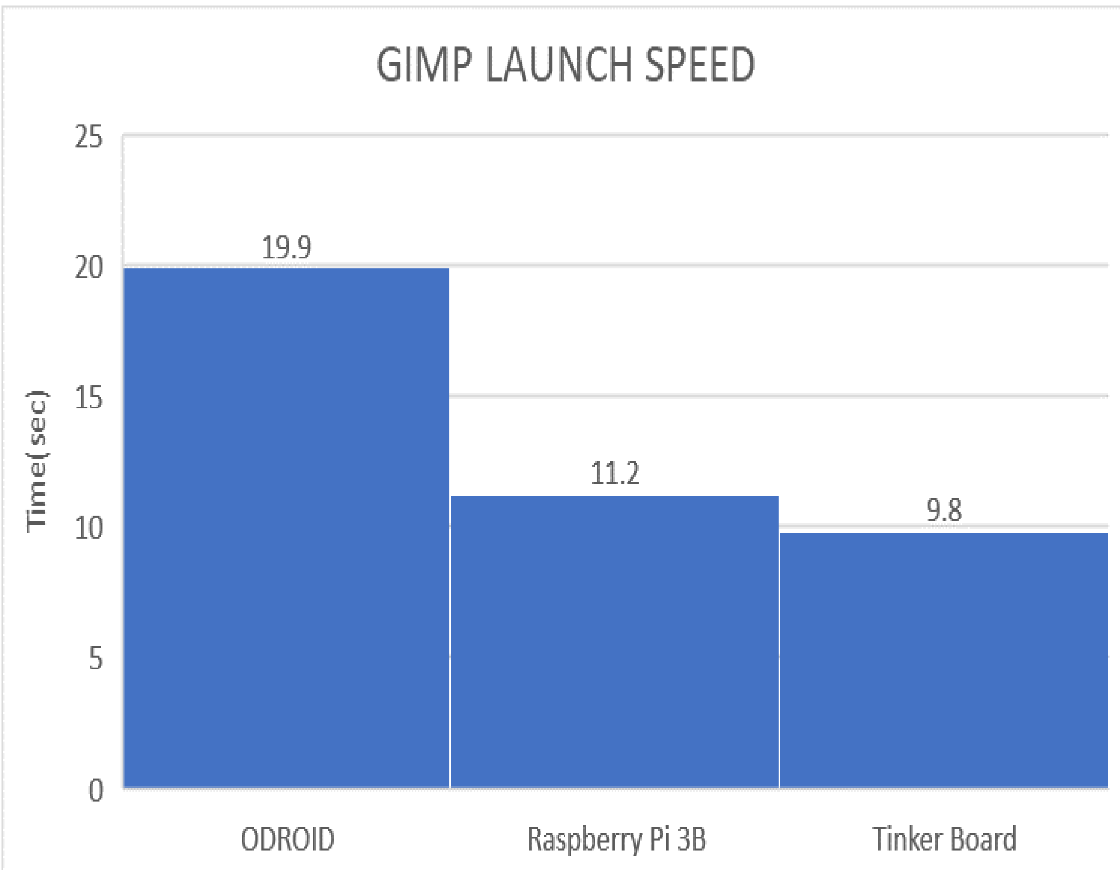
Figure 5: SYSBENCH CPU Test



As you can see from the table above the ODROID-XU4 did really well as expected. The ODROID-XU4 at an execution of 15.18 sec. The Tinker Board had an execution time of 26.73 secs, and the Raspberry Pi 3 had 45.75 secs. The ODROID-XU4 did better than the Tinker Board, and the Raspberry Pi 3. As expected, the ODROID-XU4 took this round on testing, proving its faster CPU specs.

The second Test we did on the Single Board Computers was opening a really heavy program. We mention heavy in terms of how big the program and how it needs a lot of packages to load. This test will display how the Single Board Computers can handle large applications and how fast the software and the hardware are. This test will also show us how the processing and storage speed are. The program that I will be using to test is GIMP image editor. I tested every single SBC individually and obtain the values. You can the data obtained in the figure below.

Figure 6: GIMP Launch Speed



The results obtained from the testing is intriguing. We expected the ODROID-XU4 to win this one as well, since it has a faster CPU. It was not the case, because the Tinker Board won surprisingly. This illustrates and confirms the power of software. The Tinker OS is a lot better and smooth, compare to the ODROID-XU4 which has a quite delaying feel of Ubuntu Mate.

4.1.3.8 Chosen Single Board Computer

After carefully evaluating the different Single Board that we were considering and that would fit our project, we chose to go with the ODROID-XU4. As you can observe from the table below the ODROID-XU4 has a slight advantage in the metrics that we need for our project. Despite it being a slightly higher price than the raspberry pi 3B+, it has all the features we need and has better processing power while keeping its power consumption as low as the Raspberry Pi, which is remarkable how it is able to do that. Overall the ODROID-XU4 won the race as it will be the best Single Board Computer for our project needs. The ODROID-XU4 is overall the best options for us at the moment. We rated the comparisons of all the boards previously mentioned on a scale of 1 to 5, with 1 being the lowest and 5 being the highest. These scores were then multiplied by a factor of 3 in order to give us a closer comparison on the effectiveness of each potential product. These evaluations can be seen in Table 14 below.

4.1.3.9 Switching to the Raspberry PI 4

In the implementation stage we ran into a little trouble with the ODROID-XU4. We weren't able to get ODROID-XU4 to start the required services on boot up. The services were things like ROS topics, ROS nodes , and ROSCORE. Being able to start these components was vital if we ever wanted to put the rover on the ground and for it to move. After many unsuccessful attempts we tried switching to the Raspberry PI 4, to see if we could start the services on start up. It worked marvelously with the Raspberry PI 4, as it worked on the first attempt. After that we decide the switched everything to Raspberry PI 4 instead. The transition was pretty great and smooth, as the code base we had was compatible with the Raspberry PI 4.

Table 16:Single-Board Computer Options Weighted

Criteria	Weight	Evaluation			Score		
		Odroid XU4	Raspberry Pi 3 B+	Nvidia Jetson	Odroid XU4	Raspberry Pi 3 B+	Asus Tinker Board
Processing Power	3	5	4	3	15	12	14

Power Consumption	2	5	5	1	10	10	10
Cost	1	4	5	1	4	5	4
Total Scores					**29**	**27**	**28**

4.2 Microcontroller Control Board

This section describes the use of the research and part selection of the microcontroller subsystem within the rover project. In general, this subsystem will be used for actuator control and low-level commands, serial communication between the single board computer, and Analog and Digital input and output for the various sensors used on the rover. The single board computer is critical to the control and sensor subsystem of the rover and care in research and part selection is crucial to the success of this project.

4.2.1 Microcontroller Overview

A microcontroller is a small single integrated computer on a chip that is usually used for specific operations in an embedded system. A microcontroller is usually comprised of I/O connections/peripherals and memory storage. Microcontrollers can be argued to be in virtually every aspect of electronics in today's world. From being placed in simple appliances such as coffee makers and microwave ovens, to larger machinery such as vehicles and many IoT devices. Microcontrollers function primarily for one or several specific functions without the overhead of extra computational power or memory capacity. This makes solutions for single use purposes easy and affordable. Microcontrollers are also efficient when it comes to development and testing. Modern microcontroller usually come with supplementary documentation, software, hardware, and even live service centers for in-depth questioners. Programming with microcontrollers are much simpler through supplied serial interfaces and compilers supported by the manufacturer. Microcontrollers can also make low-level hardware configuration easier to access than other higher end CPUs.

4.2.2 Microcontroller Need

For our project, the microcontroller is functioning as a control board subsystem within our hardware stack. It is used as a means of processing and communication between several peripheral I/O devices and the single board computer. Due to the low cost of and specific functionality, the microcontroller can be used to process our sensory data and actuator commands at an affordable price. Examples of the I/O devices include sensors such as IMUs, stepper motors, and heat sensors. Since it is also easy to test the software needed to support our desired functionality, integrating into our final printed circuit board will be fairly simple without the use of complex programming protocol hardware in our final PCB.

4.2.3 Microcontroller Board Options

As a team, we have consolidated the commercially available Microcontrollers Boards on the market and have researched into which controller will be the best for our solution. Using these boards could have been very helpful for us since all the hardware and in some instances the software is bundled into a solution ready for development. It also saved us time to choose the microcontroller we would like to use based on the open hardware boards since they reveal many of the features and connections available to use. Several of the most notable controller boards we researched into are the Arduino UNO, ArduinoMega 2560, the Pixhawk X4, the MSP430, and the MSP432.

4.2.3.1 Arduino UNO

The Arduino UNO is a microcontroller board made for a wide variety of consumers for educational purposes to commercial start-up development or prototyping. It houses the ATmega328P microcontroller that is a high performance 8-bit processor. This board has been extensively used and tested for a variety of applications. The Arduino UNO has many software and community resources available online and in many how-to manuals in text. The Arduino UNO may be the simplest microcontroller we have researched although it could be useful in our application.

4.2.3.2 ArduinoMega 2560

The ArduinoMega 2560 is a microcontroller similar in many aspects to its smaller counterpart the Arduino UNO but has many more useful features. Like the Arduino UNO, the ArduinoMega 2560 is also used in a variety of applications and prototyping. Where the ArduinoMega shines however, is the increased number of I/O pins and connection points. This is due to the use of the ATmega2560 microcontroller which on the board is a surface mount device component (SMD) soldered onto the board. Previous generation of the E-Goat projects have used the ATmega2560 as their choice of Microcontroller to control the sensory Ultrasonic input and use the motor control.

4.2.3.3 Pixhawk 1

The Pixhawk 1 is a specific microcontroller board that is used primarily in the fields of autopilot hobbyist and autonomous developers that is called a flight controller board. The Pixhawk 1 is created from an independant open-hardware project that was originally based in 2009 on the ATmega series but grew as hardware capabilities needed to increase. The Pixhawk 1 has several major advantages over other boards including its many built in I/O breakout pins for easy access and several software compatibilities to easily integrate off the shelf peripherals. The Pixhawk uses a much larger 32-bit ARM Cortex M4 and houses many onboard sensors such as an accelerometer, gyroscope, and magnetometer. One thing to consider about the Pixhawk 1 is the utility towards the project and the cost. The biggest advantage of the Pixhawk 1 is the software compatibility with ArduPilot, an autopilot software that can be easily used with a Mission Planner software. However, even with the plethora of resources, there is a steep learning curve and some peripherals we plan to use may not be suitable for this board.

4.2.3.4 MSP430 5529LP

The MSP430 5529 LaunchPad is a microcontroller board specifically used to demo the onboard MSP430 5529 microcontroller. This microcontroller board uses a 16-bit architecture with 63 I/O pins, similar to the ArduinoMega. This microcontroller board is also easy to integrate with other technologies and prototype with its included 40-pin I/O interface that can be used with Texas Instruments Booster Pack, a family of peripheral boards. These booster packs may include technologies such as RF or WIFI communication. This LaunchPad also contains easy interface capabilities to easily program the board via its onboard USB hub. One of our team members is also familiar with the board as it has been used in previous projects so the hardship of learning are greatly reduced. However, documentation and software support for this board is not the same as it is for the Pixhawk and the Arduino series boards as the online communities are much smaller in its use.

4.2.3.5 MSP432P401R high-precision ADC LaunchPad

The MSP432P401R is a successor to the MSP430 line of Texas Instruments Microcontroller series. Unlike the MSP430, the MSP432 family of microcontrollers are based on the ARM Cortex-M4F with 32-bit processing capabilities. This allows for better precision accuracy and computation for specific use cases. The MSP432P401R launchpad is an extension of the MSP430 development boards that include the same I/O interfaces and compatibility with TI booster packs much like the MSP430 5529LP. Along with the increased computation capability, the MSP432 series also comes with a boost in memory address space at a size of four gigabytes and higher clock frequency at 48MHz. At the same cost as other LaunchPad boards with higher performance rating, the MSP432P401R LaunchPad is a good consideration within our lineup of microcontroller boards.

4.2.4 Microcontroller Board Comparisons

With our options consolidated into four choices, we display a table to layout the technical details of each board, showing at a glance what each board is capable of through a weight scale. For our purposes, the microcontroller must be capable of handling many peripherals through and data variety such as serial communications and PWM signals for actuator control. We also needed to find a microcontroller board solution that will be able to handle and easily programmable to interface between the computational board (the single board computer) and the peripheral output.

Table 17: Board comparisons Part 1

Criteria	Weight	Evaluation			Score		
		Arduino Uno	Adruino Mega 2560	Pixhawk 1	Arduino Uno	Adruino Mega	Pixhawk 1
I/O Pins Used	5	9	4	4	45	20	20
Libraries	6	9	9	9	54	54	54
MC Programming	7	8	7	7	56	49	49
Cost	4	10	8	2	40	32	8
Hardaware Overhead	3	7	3	5	21	9	15
Total Cost:					216	164	146

Table 18: Board comparisons Part 2

Criteria	Weight	Evaluation		Score	
		MSP430 5529LP	MSP432P401R	MSP430 5529LP	MSP432P401R
I/O Pins Used	5	7	9	35	45
Libraries	6	5	4	30	24
MC Programming	7	6	4	42	28
Cost	4	8	9	32	36
Hardaware Overhead	3	6	7	18	21
Total Cost:				157	154

4.2.5 Microcontroller Selection

The microcontroller that we selected needed to create PWM signals for motor controllers, interface with digital and analog signal input and output, and support serial communication between the Single Board Computer through serial communication. Due to the limited time constraints of the project, the microcontroller must also needed to be simple enough to program through provided IDE software and when programming the final PCB design. Based on the review and research done on the different microcontrollers, we have narrowed our selection and chose the Arduino Uno as our choice of a development board for this project. As mentioned before the Arduino Uno uses the ATmega328P microcontroller chip.

Table 19: Microcontroller Comparisons

Features	Atmega328P	ATmega2560
Clock Speed	16MHz	16MHz
I/O's	14	54
Analog Inputs	6	16
PWM's	6	6
Operating Voltage	5v	5v
Supply Voltage	7-12V	7-12V
Flash Memory	32KB	128KB
SRAM	2KB	8KB
EEPROM	1KB	4KB

This microcontroller met the required functions desired for the subsystem. Through the use of the simplistic Arduino IDE, we will be able to develop the solution we need through the plethora of C/C++ libraries available to us through the community. The other important factor is that both of these microcontrollers are capable of analog input and digital I/O. This allows us to use several communication protocols such as I2C, UART, and SPI. Both microcontrollers are capable of low power consumption to not draw from the battery resource of the other subsystems.

4.2.5.1 Atmega328P[4]

The Atmega328P is the smaller of the two ATmega microcontrollers we researched into. The microcontroller has 32 lead pins that can come in a variety of different packages such as quad flat package (QFP) or a dual in-line package (DIP). Of the 32-lead pins on the package, 23 of these lines are programmable I/O lines. Of these 23, 14 are digital I/O lines, 6 are PWM capable lines, and 6 are analog input lines. The Atmega328P can operate between 2.7v – 6v. The Atmega328P also contains two 8-bit Timer/Counters and one 16-bit Timer/Counter with separate prescaler and compare mode. The microcontroller will also be able to withstand temperatures outdoors with a maximum operating temperature at 125° Celsius. The microcontroller has 32K bytes of in-system self-programmable flash program memory with 1Kbytes of EEPROM and 2Kbytes internal SRAM. The Atmega328P is also capable of creating SPI, I2C, and UART serial interfacing. [4]

The package we will be using for this specific microcontroller is the DIP package. For rapid development and less complexity on the final PCB design, the DIP package of the Atmega328P will be extremely useful. By using a dip socket on the final PCB design, we will be able to easily transfer our Atmega328p to from our development board the Arduino Uno to the created PCB.

The ATmega328p is also a cost-effective microcontroller for this project. With a cost of approximately $2.00[5] the ATmega328p is a viable option to stay within the budget constraint of our project. If deemed necessary, we will split the subsystems such as Motor control and Sensor readings to two individual control Atmega328p microcontrollers. This will allow for distinction and ease of integration and stress testing in the final stages of project development.

4.2.5.2 Revision to Atmega2560

During the development process of the microcontroller code, we came to realize that several library dependencies to communicate with the single board computer required a substantial amount of space in the microcontroller. This library called ROS Serial implemented for the Atmega2560 using the Arduino IDE and Arduino boot-loader on the chip caused our memory to overflow when creating stream buffers for the multiple messages used in our code. The Atmega328p would not be able to handle over four message sizes with over a 256kb buffer size. Several other libraries including the Adafruit AHRS library which provides sensor fusion for the IMU and the Adafruit GPS library greatly increased our need for flash storage on the system. Therefore, we decided to switch our microcontroller to the Atmega2560 around the end of January 2020.

4.3 Obstacle Avoidance Sensors

Part of having an autonomous lawn mower means that it must be “autonomous.” This means that the robot being constructed must be able to move, or navigate, around by itself, without the aid of any remote control, or human interaction. In order to achieve this feat, we are planning to use a sensor that will be able to detect objects around it. Although several sensors were looked at in comparison with the Lidar sensor, we found that there was ultimately no better sensor out there on the public consumer market that could compete. The following section will delve deeper into the reasoning behind why we chose the Lidar sensor.

4.3.1 Lidar

The Lidar we will be using for this project is a Slamtech RPLIDAR A2. Our reason for choosing this specific lidar is simple - the team before us used it so we get it for free! In all seriousness, It is a very expensive and top of the line lidar with very good resolution and customization. Lidar is very simple in concept. The head of the lidar will rotate at a certain rpm and one of the lidars ports transmits a laser, the second port acts a receive and catches all the reflected light and with the combination of knowing its rpm, when the light was sent, and when it was received, you begin to be able to build a 2d map of your surroundings. This is very useful for things like obstacle avoidance and navigation. We will use the lidar in addition with the camera to be able to more accurately know where we are and be able to better adjust our position to avoid obstacles.

4.3.2 Camera

In this section we will discuss all the needs that we have for a camera. We will also discuss the various types of cameras on the market. We will also discuss briefly how a camera in conjunction with computer vision can create a realistic view of the world for the robot to navigate smoothly and safely. The camera choice in this project will play in important role in the success of the robot. We will dive deep into the research for cameras so that we can have a good understanding of the best camera that will interact with our Single Board Computer. I will show the various criteria that we have in place, to show how we are choosing the best camera that fit our initial requirements.

4.3.2.1 Cameras for Map Building

The camera chosen will help with the implementation of the computer vision aspect of map building. The camera will be a digital one that will create images containing pixels that can be processed using the open source computer vision and OpenCV libraries. Certain cameras have really useful features that can help with map building applications like 3D imaging,3D image reconstruction, path planning, deep learning, obstacle detection, object recognition and tracking.

Some of the techniques of map building that we are looking to implement are the Simultaneous Localization and Mapping (SLAM) and Large Scale Direct Simultaneous Localization and Mapping (LSD-SLAM). These techniques can be used in partnership with a camera. This combination will enable us to have a fully running autonomous lawn mower using AI.

3D reconstruction is the ability to create a 3D map of the environment. A camera that have that feature can help the robot understand and interact with the world.3D reconstruction is very important when it comes to collision avoidance, motion planning, and realistic integration of the real and virtual world as seen in the figure below.

Giving the robot the ability to see is a really crucial aspect for it to be autonomous, which is the reason for the camera on it. The use sonar, lidar or infrared sensors are obsolete when cameras are in play, but can be added as secondary protection elements.

Cameras are perfect cost-efficient way to implement a smart tracking, mapping and detection for this project. The robot should be able to start at any random environment and gracefully be able to build a map of its surroundings and be able to localize where on the map itself is located. An example of that would be for the autonomous lawn mower to enter any farm in the world and be able to efficiently map the area and learn about the environment around it. This type of feature is in its infancy stage in research but plays a great role in our success of achieving a truly autonomous lawn mower. The camera for map building is essential given the design constraints and initial requirements that we set in place.

3.5.2.2 Camera Needed

For this project we will a camera that can help the robot be able to see the world and map its surroundings. The Single Board Computer will be receiving data from the camera as input, so we need one that is compatible with it. Another criterion that we will take into consideration is the efficiency of the camera in terms of handling the demanding task that the robot will need in order for it to navigate and help to determine any obstructions in its path. We will need a camera of high resolution that will help with the implementation of the computer vision software onto the system. There are many good varieties of cameras on the market, but we decided to narrow it down to a couple good ones based on the criteria that was most important to the team. The camera chosen will be transmitting data to the Single Board Computer via USB, so it is important to have a camera that can support USB 3.0 for fast data transmission. USB 3.0 is 10 times faster than USB 2.0. This allows a smoother and easier transmission, which is perfect for us as our project requires fast data transfer.

Table 20: Camera Comparisons

	Raspberry Pi 3 Model B Night Vision Camera	Raspberry Pi Camera Module V2 [A25]	Zed camera
Manufacturer	Unistorm	Raspberry Pi	NVidia
Price	$25.99	$24.90	$449
Size	4.5x2x1.2 in	4.7x0.9x3 in	6.89 x 1.18 x 1.3 in
Key Elements	-Night Vision capability -Low costs -5 MP Fisheye Camera Lenses -1080P sensor resolution - 72 degrees field of view -infrared or flash LED compatibility -Compatible with Raspberry Pi -LEDs automatically turn on when the environment around the camera gets dark	-8 MP native resolution high quality Sony IMX219 Image sensor -Capable of 3280x2464 pixel static images -Compatible with Raspberry Pi -Maximum of 1080P30 in Raspberry Pi Board	-Night Vision capability - High-Resolution and High Frame-rate 3D Video Capture - Spatial Mapping - Depth Perception indoors and outdoors at up to 20m

4.3.2.2 Raspberry Pi 3 Model B Night Vision Camera

The Raspberry Pi 3 Model B Night Vision Camera has a 5 Megapixel camera. The Raspberry Pi 3 Model B Night Vision Camera has night vision accompanied with a flash on both sides. It has sensors that automatically detect if there is low light in the room and turns them on if they do. This feature allows the camera to get a better accurate picture of its environment. The current of this device is at $25.99.

Figure 7: Rasberry Pi 3 Model B Night vision Camera with permissions from miuzeipro.com

This camera has the potential to be an excellent pick for us as it will have the night vision option and will fit our specification requirements that we initially set. This particular camera was chosen last semester to be the eyes for the robot. It was connected through a Camera Serial Interface, which meant that the camera had to be compatible with the chosen Single Board Computer. At the time they used Raspberry Pi 3 Model B as the Single Board Computer of choice, so it combined perfectly. The Raspberry Pi 3 Model B Night Vision Camera has a really good resolution at 1080p.It also has a focal length of 2.1, diagonal angle of 130 degrees and provides 3.3V power output.

4.3.2.3 Raspberry Pi Camera Module V2

The Raspberry Pi Camera Module V2 has 8 MegaPixel camera a range of resolutions and frames that can be utilized depending on the need. The camera is made by Raspberry Pi, which is a respectable company. The Raspberry Pi Camera Module V2 cost is at $24.90. The reason why the Raspberry Pi Camera Module V2 made the list was due to its amazing specs. Raspberry Pi Camera Module V2 has 8MP Sony IMX210 image sensor with a fixed focus lens The Raspberry Pi Camera Module V2 can support 3280x2464 pixel static images along with 1080p at 30frames/sec, 720p at 60frames/sec and 640x480p60/90 videos. The Raspberry Pi Camera Module V2 has really great quality video and would serve as a good visual input for the autonomous lawn mower. One of the important features that it is missing is the night vision. The night vision is crucial for this project as is one of the specifications given in the initial document. In order for us to be able to use this camera we would have to buy a flashlight and find a way to attach it to it so the Raspberry Pi Camera Module V2.This one of the drawbacks that may disqualify this camera in our final selection of cameras.

Figure 8: Raspberry PI Camera used with permission from pishop.us

4.3.2.4 Zed Camera

The Zed camera is one of the best cameras on the market. The ZED camera has stereoscopic view. The ZED can capture depth in every single pixel. We, humans use two eyes to see the world perfectly. The ZED camera uses two lenses to view the world just like a human would. The ZED camera has a dual 4Megapixel Camera that captures 2K 3D videos. It also has low-light sensitivity which would allow it to view even with the most challenging of environments. The ZED is the fastest camera in the world when it comes to depth. The ZED camera captures videos at 1080p at 30 frames/sec. This is incredible specifications, and with this type of camera, we would in great position to easily implement our computer vision side of the project as it would give the lawn mower a great set of eyes for them. One of the most intriguing features that sets the ZED camera apart from the other is its 3D mapping ability. The ZED camera is able to accurately map its environment using binocular vision and high-resolution sensors. With this camera, the autonomous lawn mower would be able to successfully know how far objects are itself and know where it is on the map.

Figure 9: Zed Camera with permission from zedcamera.com

4.3.2.5 Image 6: Intel® RealSense™ Camera ZR300

The Intel® RealSense™ Camera ZR300 is tremendous camera. The Intel® RealSense™ Camera ZR300 does a lot of the same thing as the Zed Camera but for the fraction of the price. The Intel® RealSense™ Camera ZR300 catch our eyes due to its wonderful specs that stood out to us. The Intel® RealSense™ Camera ZR300 can be used Indoors or outdoors. This is important for us to know as we want a camera that works anywhere without problems, allowing the autonomous lawnmower to not be limited to a certain environment. The Intel® RealSense™ Camera ZR300 is equipped with depth sensors, which is perfect for what we are looking for in a camera. The depth technology that they are using is the Active IR stereo. The Intel® RealSense™ Camera ZR300's stereo images resolution and frame rate is at 60 frames per second fixed focus using 2 VGA's. The Intel® RealSense™ Camera ZR300 RGB camera resolution is 2MP, up to 1080p at 30 frames per second, with 16:9 aspect ratio, rolling shutter, fixed focus. The Intel® RealSense™ Camera ZR300's depth output resolution and frame rate is able to support 628x468 at 60 frames per second. The minimum depth distance support by the Intel® RealSense™ Camera ZR300 is 0.6m, and the maximum range is dependent on the light conditions This will allow us map out terrains and be able to have a good navigation for our lawn mower to use. The tracking module of the Intel® RealSense™ Camera ZR300 is equipped with Fisheye camera resolution using VGA at 60 frames per second. The Fisheye camera field of view (FOV) is 166° × 100° × 133°. That is pretty good for our demand for this project. The maximum power consumption for the Intel® RealSense™ Camera ZR300 is only 1.9 W, which is really good as we do not want a camera consuming all of our power.

Figure 10: Intel® RealSense™ Camera ZR300 used with permissions from amazon.com

Figure 19: Comparing Sensors

Criteria	Weight	Evaluation			Score		
		RPLidar (LiDar)	Ultrasonic Sensor	Computer Vision	RPLidar	Ultrasonic Sensor	Computer Vision
Accuracy	5	4	1	3	20	5	15
Ease-of-Use	3	2	4	3	6	12	9
Cost	2	1	4	3	2	8	6
Total Scores					**28**	**25**	**30**

4.4 PCB

Printed circuit boards, or PCBs, are probably the most important portion of this project. Without a PCB, it would be extremely messy in terms of wiring, and would risk potential fire hazards with so many exposed wires as well as the risk of wires getting caught in wheels.

4.4.1 PCB Software

When it comes to PCB software, there are many ones to choose, but we've narrowed it down to three: Eagle, Kicad, and EasyEDA. These three we have determined to have everything we need such as decent libraries, a schematic editor, and a PCB layout tool. With that being said they do have advantages/disadvantages. We will start with Eagle. Eagle is a well-known software with a great user interface as well as a pretty intuitive system. Eagle is also well used in industry and is able to achieve a very complex board. However, Eagle is not free and If we choose to use it, we would be using the free version which has set limitations such as board size, layer size, and other things of that nature. Next, we have Kicad. The special thing about Kicad is it is an open source program and completely free to the user. This is nice because there are no limitations to the program, and you can use it fully. The Major downside to Kicad is its user interface. It's not as user friendly or easy to use as Eagle and that can make a steep learning curve for a beginner. The last one we will go over is EasyEDA. EasyEDA is interesting because it's an online software with all the advantages of Kicad and Eagle with the extra advantage that it can be used in a team setting where multiple people can log onto and work on the same project. This is great for keeping everyone in a group up to date and allowing others to get to work on their portion of the project as well as being able to see the whole thing come to fruition. For this reason, we have chosen EasyEDA. Full disclosure I have actually already used Eagle and Kicad and I think it's a good idea to branch out and become familiar with other software. EasyEDA will hopefully allow us to keep up to date with each other's work and really help things move forward quickly.

4.4.2 PCB Sourcing

For PCB sourcing, there are a few things you need to take into consideration. One being if you want the board to be populated by the company after they make the board. Another being if you want it made in America or from a foreign country like China. An American made board might be 2-4x more expensive than one bought overseas however, some grants and contracts stipulate that all the work you need done/things need produced must come from an American company. The last thing we want to consider is complexity. Some PCB manufacturers do not have the machines or resources to do certain trace sizes/layers/vias. We need to look at our needs and match the appropriate manufacturer who is up to the task of making what we need. When it came down to choosing a PCB manufacturer, we went with who we thought would give us the least amount of problems at the most reasonable cost. We had to use an evaluation chart in (seen in Fig. Xx) to decide between our three main choices: Oshpark, PCBWay, and JLCPCB. Now we the three main concerns when it came to choose a manufacturer were cost, time, and reliability. We gave more weight to what we thought would be the most important factors of choosing. The most important being reliability, what good is it if the PCB only cost $2 and comes in one week if it is not made correctly and has to be debugged and then hopefully fixed/ remade. Our second two factors (cost and time) were still important so we weighted them the same. We decided the scores in each category from reviews and word of mouth of the three companies. Once we totaled the scores. We found that JLCPCB has the highest score with PCBWay at a close second. It was surprising to see how far behind Oshpark fell (mainly because word of mouth on board quality was terrible). The only thing to consider now is that when it comes to ordering, the company were making the order from must come from the United States. That means that technically we can't use the sponsors money to order from JLCPCB because it comes from China. The simple solution to this problem is to just pay for the PCB ourselves. We shouldn't need too many boards and at their prices, it shouldn't hurt the bank to much either. It will be worth it for peace of mind and confidence in the product were receiving.

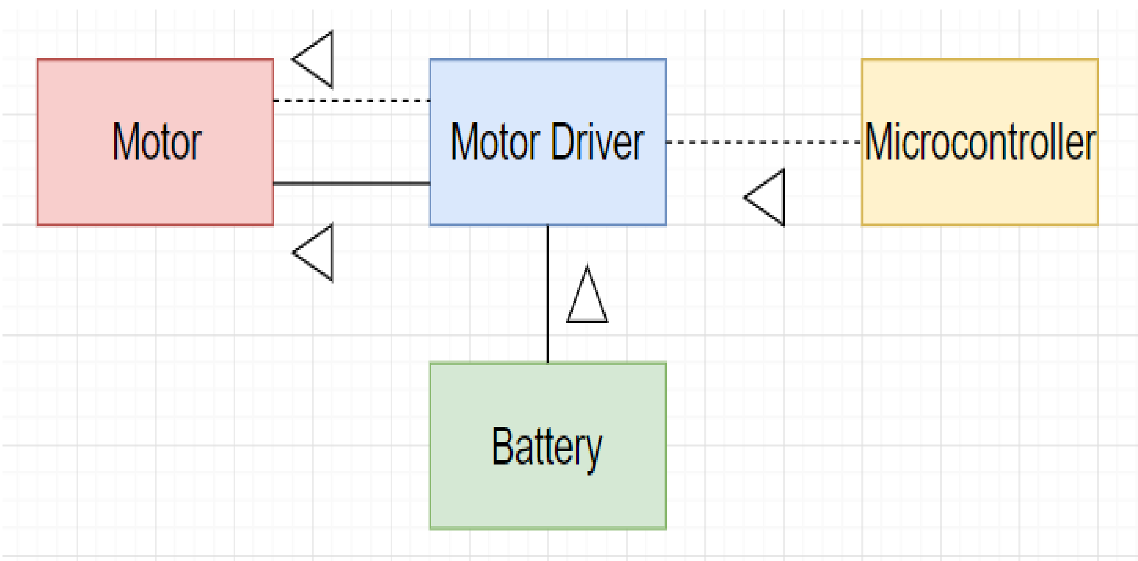
Figure 20: PCB manufacturer comparison chart

Criteria	Weight	Evaluation			Score		
		Oshpark	PCBWay	JLCPCB	Oshpark	PCBWay	JLCPCB
Cost	2	1	2	3	2	4	6
Time	2	2	2	2	4	4	4
Reliability	3	1	3	3	3	9	9
Total Cost:					9	17	19

4.4.3 Motor Driver

The Motor driver is one of the necessary things we must build for this project to work. In short, a driver allows things to talk to one another, and since the microcontroller will not be able to directly talk to the motor, you need the driver. The driver will work by getting a control signal from the microcontroller. This signal will let the driver know what kind of voltage/current the motor actually needs in order to make the maneuver that the microcontroller wants it to make. The driver will need an outside voltage in order to actually make the motor move because the signal from the microcontroller is just a control signal and very low in power. We will call this the VCC. The driver will regulate the VCC to give the motor the correct power based on what the microcontroller is telling it to. Each wheel motor will need one of these in order to function properly. The reason we don't need a driver for any of the trimmer motors is because they will simply be on or off at full power. There is no need to control direction or speed. We will pick the speed we want the trimmers to go and then give them that power, and that's it.

Figure 11: Motor Driver

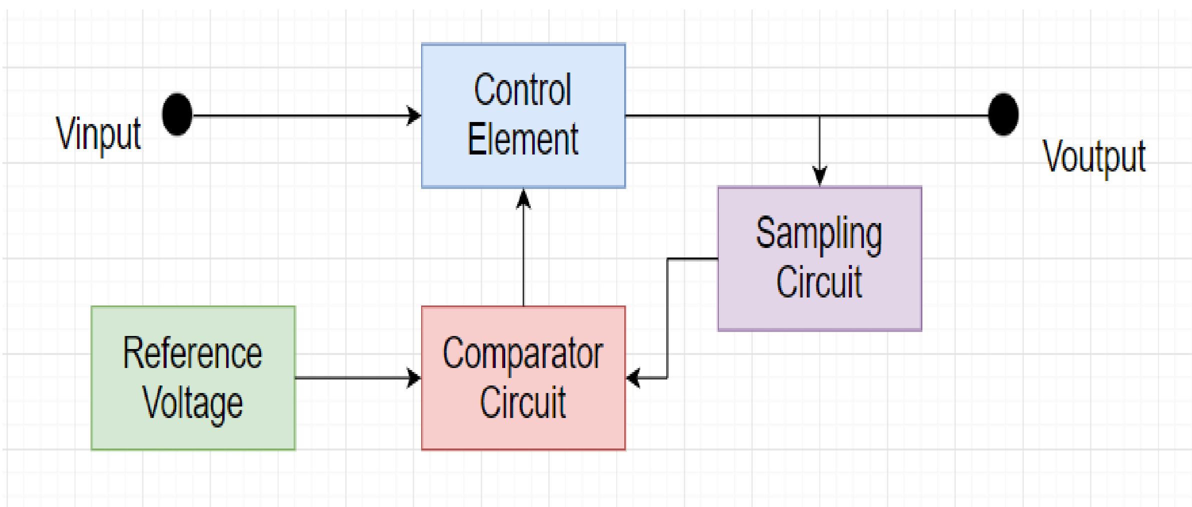


4.4.4 Voltage Regulator

The Voltage regulator is very simply a device that changes to the voltage that you put into it another voltage(whether it be lower or higher).We will only need lower voltage to power our 5V items such as the Camera, Lidar, Odroid. Since our

Voltage regulator only needs to step down the voltage from 12v to 5v, it's going to be pretty simple to make. The main component of the voltage regulator will be the potentiometer. The potentiometer we are going to buy will have a small adjustment screw on it that will change its resistance as it is turned one way or the other. To make it 5V you simply take your multimeter, put the leads on the output and adjust your potentiometer screw until you get to your desired voltage. This should be a simple board to make, however, we will need to attach a heat sink to it, as voltage regulators tend to produce a good amount of heat and we don't want it ruining anything in the mower. The reason we will be using a 12v battery and stepping it down to 5v instead of just using a 5v battery in the first place is the fact that we already have in our possession 12v batteries from last year's project and it makes more fiscal sense to use what we already have. Another thing to keep in mind is that the Arduino Uno microcontroller that we will be using already has on board a 3.3V and a 5V source that we could tap into. However, we would not want to use this voltage source for anything that needs a large amount of current because the Uno is not equipped to handle that.

Figure 12: voltage regulator block diagram

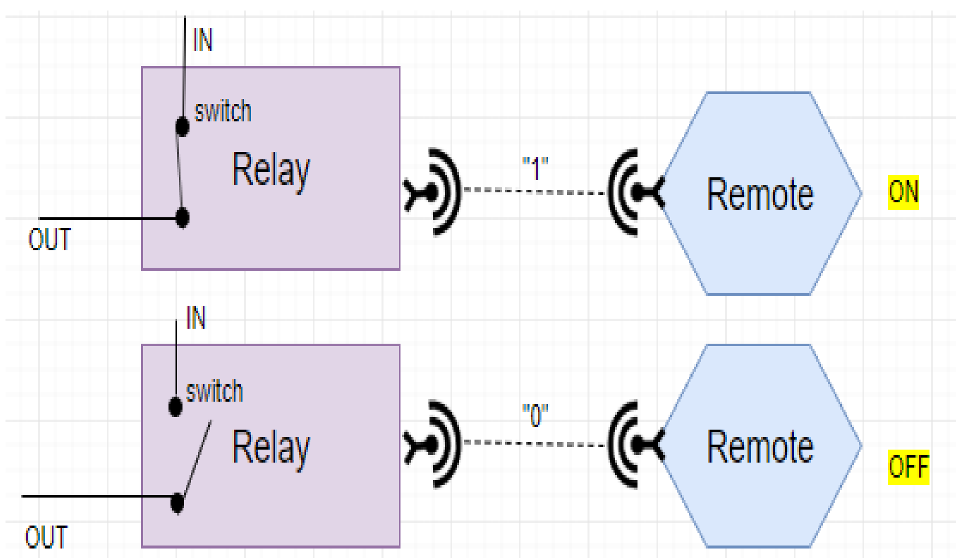


4.4.5 Remote Relay

One of the given requirements for this project is we have to be able to kill the mower with a remote switch from up to 50ft away. So, it's obvious we will need a remote that connects to a switch that powers both the trimmers and the wheel motors. The relay works like a tri-state buffer with no control signal the input (12V from battery) will be the output. However, once the PCB receives the control signal from the remote, the board will open the trace between the input and the output and that will put the output at 0v. Effectively stopping all power from getting to it.

Because the trimmers and wheel motors will run off of two separate batteries, were going to need two separate ones. Because this particular board would require some complex PCB and antenna design, we have decided that we will just buy a commercial one rated for the amount of power we will have going to the motors. The ones we have looked at and intend to purchase, also have two channels along with a remote that has two different buttons. So, we will just label one button “trimmers” and the other button “wheels” and hook up the power to the corresponding channel. Once we have that setup, we will have all we need in one remote and will be able to easily independently shut of the power to the wheels and trimmers at a distance well over 50ft. It will also be nice to be able to kill the trimmers without killing the wheels so we can work on the mowers navigation without draining all the power from the trimmer’s battery. Another safety feature we are going to add to the toggle switches to the wheel and trimmer motors. This is just another safety precaution to make sure that the wheels/trimmers do not turn on if something were to go wrong or the relay button was accidentally pressed.

Figure 13: block diagram of remote relay



4.5 Navigation Sensors

As part of our specification from the customer, the second generation of the E-Goat rover is more autonomous than its predecessor. To fulfill this requirement, the rover navigates in an open field and avoid obstacles. As previously stated in the pre-existing technologies section, there are several technologies that past senior design and commercial products have implemented. These technologies have been used as a reference to research navigation solutions to our own product. There are several parts to navigation in the scope of our project needs. The rover needs to:

1. Locate objects and react
2. Localize its position within an environment
3. Receive or create direction to reach new positions
4. Avoid certain environment cases such as illegal positions

With consideration to monetary and time constraints, the team has procured several solutions to the navigation portion of the project. The most notable technologies that we believe would be extremely helpful are boundary wire, GPS, Inertial Navigation, and object detection sensors.

4.5.1 Boundary Wire

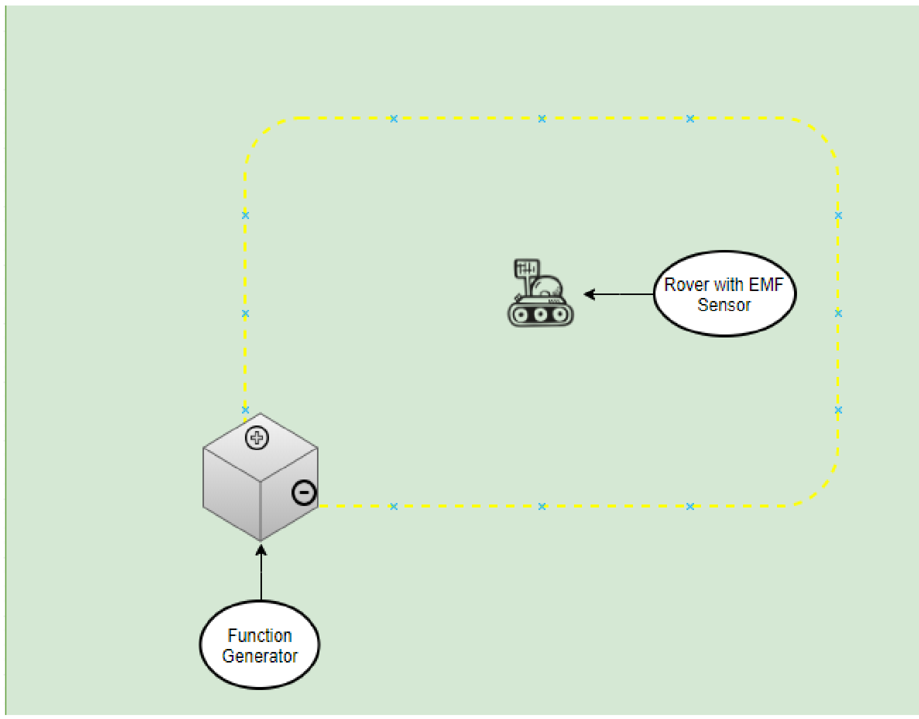
As the rover moves within the environment that it is trying to mow the lawn, the boundaries of the grass need to be identifiable. Boundary wire offers a simple solution to creating an invisible barrier around a perimeter of land. As previously stated, it is used in almost all commercially available autonomous lawn mower today. It is also used for other applications such as invisible fences for pets and other commercial guidance cases. Boundary wires use two different subsystems that comprise of the larger solution: a perimeter wire with function generator attached in a circuit, and a receiver with an electromagnetic field (EMF) sensor.

4.5.1.1 Boundary Wire Overview

The perimeter wire is laid out around the edges of a certain area that creates a kind of fence from keeping the object in our out. This wire completes a circuit that is connected to a function generator. The generator produces an alternating current that is typically at a frequency of 10.4khz in several commercial applications. The wire is usually dug under the ground between two or three inches. However, due to the customer constraints of no modifications to the field, the wire in our application would need to lay on top of the grass. This would still keep functionality while not destroying the solar panel property.

The EMF sensor is then placed on the object that is to be constrained within the boundary. When the object crosses over the boundary, the EMF sensor senses the electromagnetic field from the current going through the wire. This trips the sensor triggering the appropriate response for the use case. This has been a a great implementation to our rover to avoid the strict boundary constraints. With multiple sensors on board, the rover is able to edge along the boundary still cutting grass while making sure not to cross over.

Figure 14: Boundary Wire



There are currently many on the market solution for the boundary wire issue, some even as low as $100 dollars in online retailers. This solution would be low cost in components and resources. It may be a viable option to purchase an on the market solution for a boundary wire. However, the complexity of these two subsystems fit within the scope of this project and can be produced at a similar to less expensive rate than commercially available options.

4.5.2 Boundary Wire Part Selection

With help from the previous generation of the project, we are able to easily research the parts and processes that the first generation sought out to attempt. The Signal generator is based on an integrated precision circuit timer the NE555[26]. This component is able to create precise oscillations based on the circuitry design in order for our sensor to find the frequency. Based on previous projects creating a perimeter wire with the Arduino sensor, we have decided to use a frequency range between 32Khz and 44Khz. Additional parts such as resistors, capacitors, DC barrel connections, and wiring where acquired for the function generator.

The sensor is comprised of 2 simple LC circuits and two Operational Amplifiers that output two analog signals to the microcontroller when the resonance frequency from the function generator is detected. The Op-Amps that we used come in the LM324 DIP chip package that contains four operational amplifiers. We also be used resistors, capacitors, and inductors to create our LC circuit and to set the gains of the Op-Amps.

For testing and prototyping, the rover cuts a lawn area of 50x10 feet. Assuming that the lawn area is rectangular, this requires us to supply the perimeter with approximately 120' worth of perimeter wire. The wire is weatherproof shielded for outdoor use. Several brands of perimeter wire can be found for on the market autonomous lawn mowers can be used for our purposes. We also needed a case housing for the function generator and an additional LiPo 11.1v DC battery to power the function generator while on the field. A modified plastic container found in an office supply stores is able to shield electronics from mist and rain. A full parts list for both subsystems of the boundary wire can be found here.

Table 21: Perimeter Boundary Part Selection

Item	Description
Perimeter Wire	100m Automower Boundary Wire
Screw Terminals	3xED10561-ND, 125V, 6A, 1x02 2.54mm Screw Terminals
Timer	TI NE555P Timer
Op-Amps	TI LM324N
Resistors	4x1MΩ(1206), 4x10kΩ(1206), 1x3.3kΩ(1206), 1x12kΩ(1206), 1x47Ω(2W Axial-0.6)
Capacitors	2x22nF(1206), 1x100nF(50V 1206), 1x100nF(1206), 1x1µF(1206), 1x1.2nF(1206)
Inductors	2xRLB0914 Inductors 1mH
Housing	10x5cm Plastic Container

4.5.3 Localization Technology

Global Positioning System Technology is an extremely common tool used today for almost every practical navigation sense. Whether it may be in different environments such as land, air, or see, GPS is an adaptable solution for localization and positioning. For our purposes, we researched into using GPS to localize the rover within the environment and understand its distance from other reference positions or points. We also wish to use GPS as a means to recover the rover in the event that the battery is drained, and the rover is left stranded. While researching different GPS systems, sensors, and transmitters, we came across certain solutions that could be beneficial to our rover.

4.5.3.1 Standard GPS

Standard GPS is the technology is the most fairly common and most familiar with. Standard GPS from our definition signifies the use of one receiver on the ground in communication with orbiting satellites. The receiver is able to determine its position by receiving a constant transmitting radio signal from an orbiting satellite. The satellite is equipped with an extremely accurate atomic clock and precision on its current position in its orbit. Through this, a receiver can pick up on the signal and calculate the distance since the speed of the waves are constant. Typically, a common GPS receiver needs to be able to be in communication of at least four GPS satellites in order to make the correct calculations.

This technology is accessible to all civilians with the proper equipment. A standard GPS module is easy to implement in projects and hobby receivers sell for as little as $35 dollars. Surface mount modules and microcontrollers with GPS already integrated is also common. The low cost of adaptation of the technology is a great advantage to other localization methods. It also fits into the customer constraint of having a user accessible GPS on board the rover unit.

Although GPS receivers that we view on a day to day basis seem fairly accurate on smartphones and GPS devices, there are some limitations to standard GPS that where a hindrance for us in the project. First, standard GPS technology is susceptible to accuracy and precision loss. This may be due to many reasons including but not limited to elevation, weather, buildings, rogue radio frequencies, blocking obstacles, etc. This precision loss accounts for most GPS receivers to be accurate to about ten meters of true position.

In our project, we used the standard GPS to approximately locate where we are in a set environment. Given our current position, we understand at what position objects are by integrating it with SLAM technologies or create missions to travel to one point from current position or elsewhere. The GPS may also be used as a “soft boundary” to notify the user if in the rover is out of its designated mowing or service area. Since standard GPS is accurate to ten meters as stated before, it is not a practical option to determine true position through this form. As our boundary is constricted to only fifty feet by ten feet of lawn space, it is much harder to determine if the rover is truly at a position within the rectangle. Greater accuracy and precision are needed for it to be a data point to make action decisions on.

4.5.3.2 RTK GPS

Real Time Kinematic (RTK) GPS is technological technique based on standard GPS as discussed in the previous section. At a high-level overview, the difference between standard GPS and RTK GPS is the greater positional accuracy achieved with RTK GPS. Since standard GPS is calculated by calculating the distance between the satellite and the receiver using the constant speed of the RF wave, it is received with some delay as data is transmitted through the atmosphere. With RTK positioning, a fixed base station is placed in an environment in close proximity to where the rover's current position is. The rover still uses the GPS satellites to calculate for position. However, correctional data is then sent to the rover via the base station. This allows for much higher precision that can be as accurate in many cases to the centimeter.

With the complexities involved with RTK GPS technologies, affordable solutions are hard to come by for a project with a budget constraint such as hours. However, RTK GPS technologies are widely used in hobby enthusiasts and commercial application for remote and autonomous drones, planes, and rovers. There are several GPS modules created by U-Blox such as the NEO-M8P2 that are capable of using RTK technology to work as either a base station or normal receiver. However, the NEO-M8P2 module is much more expensive than its other GPS counterparts starting at $200 versus several of its on GPS modules and its competitors modules averaging around $30.

For true localization position within the confined lawn at the solar panel field, RTK will be the only option available at an affordable cost. There are other alternatives as mentioned with the investigation of the Terra robot from iRobot, however creating a hybrid solution is beyond the range of the scope of this project and in itself a challenging feat. If we were to buy this module, we would not only need one but two for the base and the rover, making a $400 dollar expenditure on a $1500 dollar project, close to a third of the budget.

4.5.3.3 Inertial Navigation

An Inertial Navigation system is a combination of sensors and computer hardware that allow a device to continuously measure its current position through its dead-reckoning position. Inertial Navigation was first used in maritime and aircraft applications before the use of GPS in order to track a ships movements and position within the ocean. Inertial Navigation systems use accelerometers, gyroscopes, and magnetometers in combination to calculate acceleration, pitch, yaw, and locomotive degrees of freedom.

Simple Inertial Navigation systems can be created easily through a combination of the sensors mentioned before. All the sensors in conjunction form an Inertial Measurement Unit (IMU). Many IMU development boards can be easily purchased with these sensors that have easy access to readout pins to be connected to control boards. The hardest aspect of the Inertial Navigation Systems is the differential drift that happens with the accumulation of error in calculations. This makes INS positions fairly inaccurate as time goes on if not corrected by some other means of localization.

For our project, Inertial Navigation is a simple and cost-effective solution to a project scope this large. Although Inertial Navigation is far worse in precision than RTK or standard GPS, there may be workarounds to gain more accuracy through other methods. We can use a standard GPS system to continuously correct the INS readouts through software such as the use of a Kalman Filter.

4.5.4 Localization Technology Part Selection

After research into Standard GPS, RTK GPS, and Inertial Navigation, we have decided to use a combination of some of the technologies mentioned above. Although all technologies could be implemented within the final solution, cost and time constraints limit us to what technologies we can take on and reasonably be able to implement. We are able to satisfy the customers constraints and needs through using Standard GPS and Inertial Navigation. The use of RTK GPS did not meet the need of the budget constraint for the project and required more time invested to learn technical implementation to our solution. Although for future implementation of this solution, RTK GPS would be preferred to standard GPS.

4.5.4.1 MTK3339 GPS module[11]

The MTK3339 GPS module is a SMD GPS chipset with a high sensitivity level and low power consumption for precise GPS signal processing to get precise location in sub-optimal conditions. The module comes with pin input for external antenna I/O and comes with an automatic antenna switching function with short circuit protection. The GPS module is capable of a high update rate of up to 10Hz with proprietary self-generated orbit prediction for instant positioning fixing. The modules have a cold startup time of 34 seconds. This GPS utilizes the NMAE 0183 communication standard help by the National Marine Electronics Association[11]. The GPS is capable of keeping track of up to 22 satellites through its 66 RF channels of communication. Another useful feature of the MTK3339 is the ability to log data position every 15 seconds for a total of 16 hours of position history. The log data can be accessed through the module itself with built in storage and will be appended during power loss.

The module has been developed with the Adafruit Ultimate GPS Breakout V3 board[9]. The board has been useful for testing and implementation purposes. The breakout board contains an external antenna support with a uFL connector. This has been converted to an SMA attachable antenna using an SMA to uFL adapter[site]. The development board comes with 9-pin out connections. Two digital I/O for serial communication, and two VIN pins for redundancy. This has been used for communication with the microcontroller to gain positional coordinates.

Table 22: Microcontroller Positioning Information

Features	MTK339
Cold Start	34 sec
Satellites	22
Dimensions	16x16x5mm
Velocity Accuracy	±0.1m/s
Update Rate	1 to 10Hz
Acquisition sensitivity	-145dBm
Operating Current	20-25mA
Supply DC Voltage	3.0-3.4V
Tracking sensitivity	-165dBm

4.5.4.2 GPS External Active 28dB Antenna

An external antenna is used to connect to the MTK339 GPS module board in order to gain better signal from GNSS satellites to the rover. This antenna draws around 10mA of current and gives an additional 28dB of gain. The antenna is connected to the uFL to SMA adapter connected to the MTK339 on the PCB. The GPS is mounted to the exterior of the rover through a hole mounted within the body.

4.5.4.3 Adafruit Precision NXP 9-DOF Breakout Board[16]

For ease and simplicity of interfacing and development for both modules mention above, the Adafruit Precision NXP 9-DOF Breakout Board is used for testing and implementation within the product. The breakout board features both modules the FXAS21002C and the FXOS8700 sensor packages all on a custom PCB design with some additional power features. The breakout board accepts either 3.3Vin voltage or 5Vin voltage that is stepped down through an onboard regulator to 3.3v. The board also comes with Arduino libraries that make interaction and programming with the two modules simpler [11].

Table 23: Sensor Package Information

Module Package	Sensors	Degrees of Freedom		ADC Resolution	Output Data Rate
FXOS8700	Accelerometer	3-Axis DOF		14-bit	1.563 to 800 Hz
	Magnetometer	3-Axis DOF		16-bit	1.563 to 800 Hz
FXAS21002C	Gyroscope	3-Axis DOF		16-bit	12.5 to 800 Hz

4.5.4.4 FXOS8700 6-Axis sensor Accelerometer & Magnetometer [6]

The FXOS8700 sensor package is a small, low power p 3 axis accelerometer and Magnetometer that is combined to be a small CBD package. The package is supplied any voltage between 1.62 and 3.6v with a low current draw of 240µA while only one sensor is active and a current draw of 80µA when both sensors are active. This component is used to measure the acceleration and direction of the rover to localize itself. This accelerometer is equipped with a dynamically selectable acceleration scale range from ±2g /±4g /±8g to fully take advantage of the precision and range appropriate for use. The magnetometer sensor also contains a range of ±1200µT. The sensor is capable of either I2C or SPI serial communication with 14bit accelerometer data resolution and 16bit analog to digital resolution. The FXOS8700 is also capable for outdoor conditions with guarantee operation temperature between -40° and 85° Celsius.

Table 24: Gyroscope Comparisons

Features	FXOS8700CQ	FXAS21002C
Supply DC Voltage	1.95-3.6V	1.95-3.6V
Scale Ranges	±2 g/±4 g/±8 g	±250/500/1000/2000°/s
Current Draw	240 µA - 80 µA	2.7 mA
Temperature Rate	–40° to +85 °C	–40° to +85 °C

4.5.4.5 FXAS21002C 3-Axis Digital Angular Rate Gyroscope[7]

The FXAS21002C sensor package is a small yaw, pitch and roll rate gyroscope that is packaged as a CBD surface mount component with good accuracy. The sensor is capable of emitting a serial communication signal of 16bit analog to digital conversion and can support both I2C and SPI communication interfaces. The FXAS21002C can be supplied a voltage between 1.95v and 3.6v with a current draw of in active mode of 2.7mA. The sensor is also equipped with a programmable low pass filter to limit the digital output data bandwidth. One useful feature of the sensor is Low power standby mode which when activated can conserve power consumption which decreases the current consumption to an average of 2.8μA. The FXAS21002C is also capable for outdoor conditions with guarantee operation temperature between -40° and 85° Celsius.

4.6 Computer Science and Computer Vision

Computer vision is a field of computer science. It includes many problems regarding cameras, robots, and artificial intelligence. Computer vision is a way for computers to understand and process the outside world. This is very important to our project, as the e-Goat rover must explore its surroundings and react accordingly. There are many unsolved problems in this field, and many more problems that are much more difficult than they may seem. In this field it is often difficult to determine which problems are easy, and which are hard.

The e-Goat rover needs to collect information about the outside world in order to determine where to go and what paths to take. Computer vision algorithms and techniques could be used in conjunction with robot localization techniques to navigate and provide additional input to the system. The robot navigation and autonomous control, including navigating around obstacles and staying within bounds, are two examples that could be driven by computer vision. The main problem we need to solve is finding out where we are in the world and finding out what we need to do next. We can take input from the outside world to estimate each of these, but we have to come up with a design that will allow us the best shot at accurately deciding where we need to move to. There are many ways of designing such a system, and as we learn more about the field, we will better know which direction to take the Computer Vision component of this project.

4.6.1 Board Selection

Depending on what type of computer vision algorithms we wish to run, we could end up having to choose the NVIDIA Jetson, as described in section 4.3.1. While, it is much more expensive, if we find that we do need any type of neural network computation, such as Object Detection, Tracking, or Segmentation, a smaller power single board computer will not be able to handle the task. The NVIDIA Jetson has dedicated hardware to handle neural network computations, and common Computer Vision libraries such as OpenCV have compiled versions available that take advantage of this additional hardware. While we do not think this is necessary yet, we need to do more real experimentation with our planned design to see if it is doable.

4.6.2 Programming Languages

In this section we will discuss the different programming languages that we have available for us to use for our Computer vision. We took the previous year's work and we expanded on it by doing more research and writing more on the topic.

4.6.2.1 C Programming Language

The C programming language is a really interesting and useful language. It can be used to support many technical needs. The C programming language in nearly all fields. The C programming language can used to help with embedded application, memory management and other things. The C programming language is many used for implementing operating systems and embedded systems. It was built in 1972 by Dennis Ritchie. We all know how to program in C since it is part of the curriculum at the University. The downside to the C programming language is the lack of support for higher ordered data structures. This makes the task of handling unique situations arduous and time consuming. Memory management is very important in the C language. Developers must properly use it, or it can be degenerative to the system, by creating long runtimes with poorly written programs.

4.6.2.2 Java Programming Language

Java is a really great programing language. The Java programming language is an object-oriented language. Object oriented programming is a programming style which is associated with concept of class and objects. With the Java programming language, we do not have to worry about memory management as the programmer. Unlike C programming language where we have to worry about the memory management. So, with that in mind it gives a way to be more efficient when writing code as we are able to write less lines. The Java programming language is relatively simpler than C programming language, and more beginner friendly. The Java programming language is easy to implement but lags in performance compared to the C programming language and C++.

4.6.2.3 Python Programming Language

Python is one of the most popular programming languages at the moment. Python programming language is a scripting language that used in many scientific and commercial applications. Python programming language is an interpreted, high-level, general-purpose programming language. Python is an intuitive language that can be used to write code with little experience in it, however, the language itself is inherently slower than C programming language or C++ programming language. For computer vision Python programming language would be a great selection and would work wonderfully with our Single Board Computer.

4.6.2.4 C++ Programming Language

C++ programming language is another type of Object-Oriented Programming, along with Python programming languages and Java programming languages. The C++ programming language is very similar to the C programming language. The C++ programming language and the C programming language share mostly the same syntax with a little change in them. Most of us on this team is capable of coding in C++ programming language without any troubles, which will make learning the computer vision aspects of it easier on us. The C++ programming language is also great for optimal performance on the hardware. We would get the maximum performance of our ODROID-XU4. The C++ programming language has a robust standard library (STL) and a quick processing and compilation mechanism. This is beneficial to this project due to the higher performance and high capabilities with the components that will be used such as the ODROID-XU4 and camera.

3.8.3 Software Middleware

Robot Operating System (ROS) may be used as a software stack, handling various software that we wouldn't have to write ourselves. This provides a massive benefit of code reuse and plug and play functionality provided out of the box. ROS is not an operating system, but a middleware software system that can host user supplied "packages" and handles all inter process communication. ROS's primary library api's are in c++ and Python, and intended to be used on Linux operating systems, such as the one we will have on the Odroid. Many robots have been built with ROS, and there are many community packages available for all sorts of uses. With ROS, we can build a hardware abstraction and focus instead of higher-level motion planning and control, rather than building a system ourselves for controlling our wheel motors in a low-level way. A ROS system would be run on the Odroid XU4, or the main processing board, not a lower level board like the Arduino or Pixhawk. We can communicate from the computation board to the lower level control board through an interface. There are interfaces to work with ROS in most languages. It is language agnostic.

The abstraction of software and hardware in ROS allows developers to write useful packages that many robots can take advantage of. Reuse of code and portability to new hardware designs is one of the core tenets of ROS.

There are a few packages that may be useful to us. Instead of using an outside SLAM algorithm (discussed in section 4.7.4), there are a few packages that provide a method of SLAM using ROS. All that's needed is to provide the point cloud coming from the LIDAR sensor. One is gmapping. We can also generate an occupancy map with gmapping, which would be useful for pathfinding, explained in section 4.7.5.

ROS is not absolutely necessary, and we need to investigate whether it would be beneficial to us to use ROS over using a library like OpenVSLAM on its own. This will be determined once we have tested a few methods and can test with a working robot.

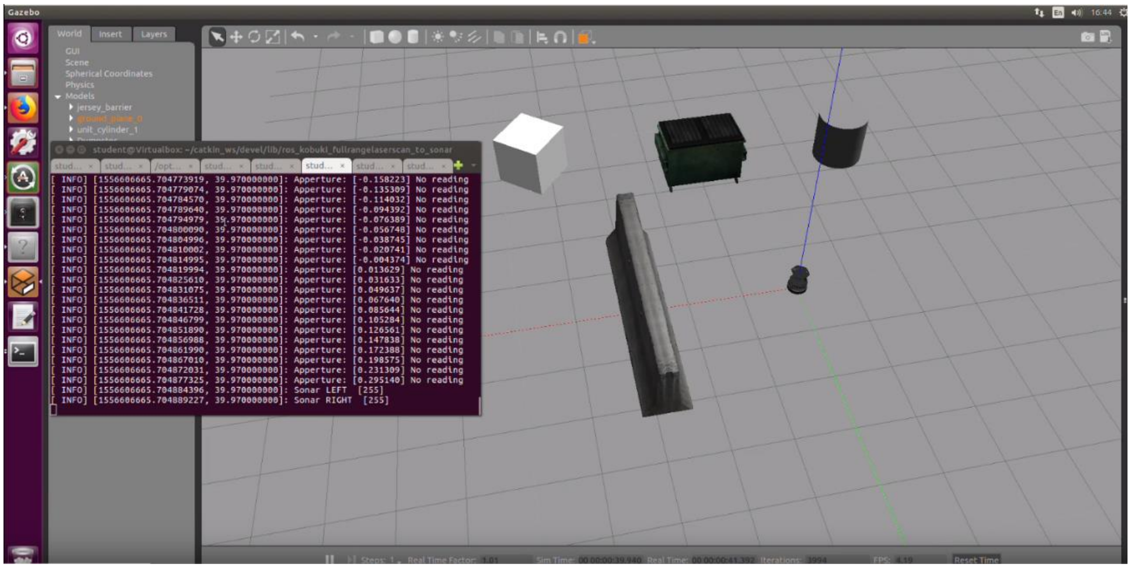
Because we will likely be programming in Python using OpenCV for other miscellaneous computer vision tasks, it would be benenficial to use the rospy APIs to take advantage of ROS in Python. Keeping as few languages as possible would be great so we can streamline the developer load. Python does come with some drawbacks, most apparent being slower runtime performance. C++ connections to ROS would likely be faster, but we do not have any experienced c++ developers on our team, and that would take additional learning curve and ramp up time.

4.6.3 Gazebo Simulation

Gazebo is a simulation framework for modern robots. It includes all sorts of features such as a physics engine, 3d graphics engine, and all sorts of sensor inputs. The goal of Gazebo is to provide a simulation that is as close to the real world as possible, to provide a complete input and output system for a robot. With Gazebo, it’s possible to rapidly build and test algorithms for performing robot tasks, without physical hardware.

It is possible to integrate Gazebo with ROS. There is a package called gazebo_ros_pkgs which provides wrappers for Gazebo. There is some manual work that has to be done ourselves, providing interfaces and setting up the system to simulate our exact hardware systems.

Figure 15: Gazebo simulation with a subscription to a sonar topic



4.6.4 Edge Detection

If a RTK-GPS solution is used, it may be beneficial to just have a smaller computer vision component, that’s sole purpose is to detect and avoid obstacles. Because our set of possible obstacles is small (people and solar panel frames, which on the robot level are just cylindrical poles mounting into the ground), the problem is somewhat simplified. We may be able to get away with a simple edge detection technique and avoid expensive computation. Edge detection using either Sobel or Canny algorithms can be a simple convolution, and in the case of Canny, some additional computation. The Canny Edge Detector involves non maximal suppression and some levels of thresholding. Canny edge detection does have some major benefits. It is much better at defining real edges than Sobel, which it is built on top of. Edge detection in this case can be used with a video feed from a camera to find the vertical edges that each of the poles would have. Usually, this is used in conjunction with a neural network for learning features about edges and detecting certain objects. We could potentially have a very simple classifier to find the poles and depending on which side of the image it is on, in addition to the distance to the pole, start turning the robot around the pole. This method could also involve using the LIDAR SLAM data (described in section 4.8.4).

4.6.4.1 Sobel Edge Detection

Sobel edge detection is one of the most important edge detection algorithms in computer vision. The Sobel edge detection uses 1st derivative of the intensity information. The idea behind Sobel edge detection algorithm is that the algorithm takes in an image which gets convolve with two tables x-masks and y-mask. The Two tables are below show the convolutional filters used.

Figure 16: Sobel edge detection filter

X-mask

-1	0	1
-2	0	2
-1	0	1

Y-mask

1	2	1
0	0	0
-1	-2	-1

After the picture get convolve with the tables. the two output tables of the masks and image are combined using the magnitude formula. This gives us a smoothened gradient magnitude output. We obtained the magnitude of the gradient with the x-mask and y-mask outputs. The Gx is the horizontal answer and the Gy is the vertical answer. The magnitude vector formula is shown below:

$$G = \sqrt{(Gx)^2 + (Gy)^2}$$

After obtaining the vector gradient. We compare that value with the threshold. The algorithm check to see if that magnitude is greater than the threshold, if it then it marks as an edge. The Sobel edge detection check neighborhood pixels and see any jumps in brightness and assumes that these pixels are edges. It also uses the arctan in other the direction of the gradient.

4.6.4.2 Canny Edge Detection

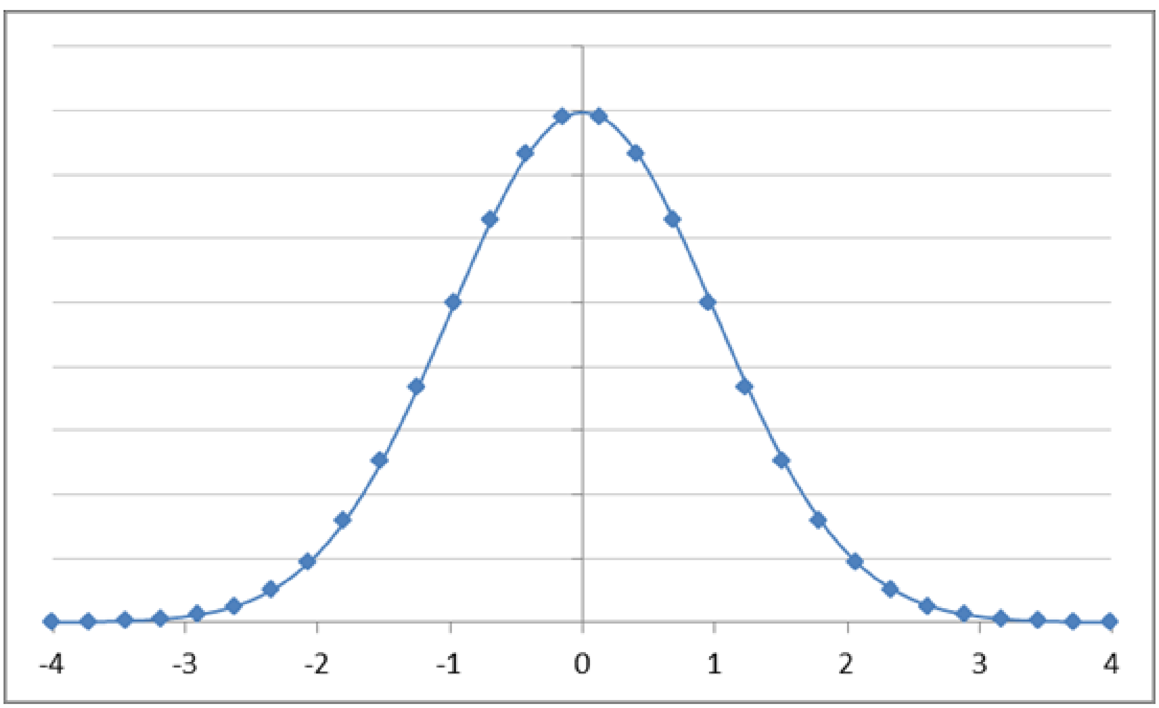
The Canny edge detection is a very useful edge detection. The Canny edge detection is an edge detection that uses a multi-stage algorithm to detect a wide range of edges in a given picture. The Canny edge detector is somewhat similar to the Sobel edge detector as it uses two tables to convolve the images with. The Canny edge detector uses the derivative of the Gaussian to be used as mask, which is different from the tables of -1/1 that the Sobel edge detector used. The derivative formula of the Gaussian is shown below:

$$G(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{x^2}{2\sigma^2}}$$

$$G(y) = \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{y^2}{2\sigma^2}}$$

AS you can both from the equation above, the Gaussian has x and y in the equation. We used the first derivative of each of the equations and we used them as the masks. Depending on the dimension of interested, i.e. 1D or 2D, we can use the appropriate formula and do the same step of taking the first derivative and assigning it to the mask. The same procedure is done as the Sobel edge detector to detect the edges in a particular picture.

Figure 17: Gaussian Bell Curve



The bell curve of the Gaussian function makes it ideal to smoothen an image, without having a sharp twist or curve in the output.

4.6.5 SLAM Techniques

SLAM, or Simultaneous Localization and Mapping, is a field of algorithms for generating a map of the world based on sensor input over time. Two of the most common algorithms are LIDAR based or camera-based SLAM. We will likely be using a LIDAR based solution, as provided by the open source software OpenVSLAM. OpenVSLAM is one of the main commonly used SLAM libraries and has been used on similar low powered hardware by others. OpenVSLAM also has a ROS package available, which will be handy if we decide to use ROS. While we may be restricted in the processing of the LIDAR input, our LIDAR sensor is not a particularly advanced sensor anyway, and depending on what else we run, we will likely be able to keep up with the point rate input. Once we have LIDAR input working, we can build an estimate of the world with a point cloud.

4.6.6 Reduction of the problem

If we can reduce the problem to the problem of finding a shortest path with the max coverage we can do, where we define coverage as the total area the robot moves through. The robot is not just moving in a line but takes up additional area (the grass that the blades will cut if the robot is standing still). There are numerous papers available to solve this common problem. One interesting one uses a modified A-star algorithm to solve this problem [2]. If we can transform our input into a useable occupancy grid map, we can implement this algorithm as is to solve our path planning issue. The output of the algorithm is a list of waypoints for the robot to follow. It is still necessary, however, to have a good localization method, so we can know where we are in relation to each waypoint. The motion planning then becomes trivial robot motion from waypoint to waypoint. The following image is from the paper, illustrating a potential map and solution:

This could work very well with a higher-level design using RTK-GPS, as described in section 4.6.3.2, or any method with an accurate Localization technique. Some other motion planning techniques we've discussed would not work well with this, such as a pure computer vision approach, avoiding obstacles with something like a camera, combined with Object Detection or Edge Detection and obstacle avoidance, because we would not have a full map of the world along with a precise position of where we are in that world at the current time.

In order to deliver a working prototype as fast as possible (which itself is a large improvement from all of last year's groups, which functionally did not do any autonomous grass cutting, even though they planned it), we have come up with a simple method for cutting. The method will be expanded after we have it working. This method is very similar to how an indoor IRobot Roomba works, and does not have any special planning or mapping. We need three main sensory components for this method:

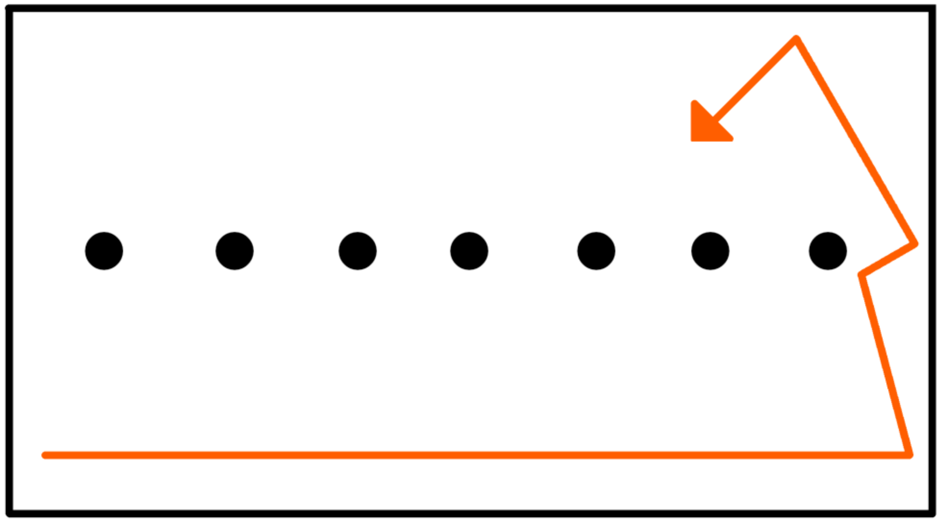
- Boundary wire (discussed in section 4.5.1)
- Lidar (discussed in section 4.3.1)
- Camera (discussed in section 4.3.2)

With these three components, we can adequately create a minimal prototype that fulfills all the project requirements. The system will work on a random turn when it runs into an obstacle, just like a Roomba. The boundary wire detection and Lidar obstacle detection components will be the two signals for the robot to back up and rotate, before proceeding ahead. The camera will be used exclusively for person detection in order to meet the requirement of shutting down the trimmer motors when a person is detected in front. As discussed earlier, we will use a pre-trained neural network such as YOLO or a similar CNN to perform object recognition. Because we are not using an NVIDIA Jetson, and our chosen Odroid board is weaker (but stronger than the Raspberry Pi), this will likely only be able to run at a few frames per second. Early testing has confirmed this to be true. This is perfectly acceptable to us.

When a prototype is ready, testing will be done to find a best suitable rotation angle. It might be the case that a slightly different rotation angle could significantly change our average grass cut in the competition time, because this method does significantly rely on randomization. For example, a very small rotation angle will cause the robot to only venture around the outer perimeter, never going near the panel poles themselves. Even though the robot will be crossing over paths it's already mowed, there will be an ample amount of time for the robot to cut a lot of grass. The competition area is 10'x50', so 500 sqft. The competition time is 15 minutes, so on average we should cut 33 sqft per minute, or ½ sqft per second. Our robot will be able to cut a line of approximately 1 ½ feet wide. It works out to be less than 1 mph necessary to cut the entire 500 sqft, if the robot is moving optimally and not cutting any grass area twice. This leaves us with plenty of room to spare.

This method will be used in order to minimize the time to a working project. Once we have this prototype in place, our goal is to use existing sensors and additional environmental data the robot will have to improve the method. That's when true localization and mapping will be necessary. The team does not want to spend lots of time and money on an extremely fancy (and CS heavy) solution which may not work out in the end. This method will guarantee us some good results, and we can focus on improving those results with a more computer vision heavy approach after.

Figure xx: Illustration of robot motion path



The above figure is a demonstration of how we expect our rover to move using this method.

4.6.7 Mobile App

In addition to including a semi-long-range remote off switch (for safety reasons, as per the requirements), we will be able to connect to the rover using a mobile app, through a yet undecided channel (one of the technologies discussed in section 3.9). Bluetooth communication is the likely choice, as it doesn't require being on a WIFI network, and can be easily implemented in a mobile app, as all modern phones have Bluetooth. Because the ODROID does not have Bluetooth built in, we can use a USB Bluetooth dongle, which would work out of the box with Linux. It is our goal to have a mobile app that has a few different main functionalities:

1. Remote manual control of the rover. Including direction, speed, and other controls for turning on/off mowing, etc.
2. Remote debug capabilities. We should have our compute single board computer send any useful debugging information to the mobile app, so we can easily check the status and what the rover is doing at this moment.

With these two main goals of the mobile app, we should be able to more effectively and efficiently develop and work with the rover. We have a lot of flexibility in designing the app. We plan on having at least two main screens, one for remote control access, and one for viewing remote debug information. There will be a way included to switch between the two screens. The app should allow easy connection to the rover. Bluetooth APIs allow an app to manually connect to a Bluetooth device. We can have a selection UI to view any rovers available on Bluetooth to connect to.

4.6.8 React Native

React Native is a toolkit made by Facebook. It is inspired by the React JavaScript web framework, also made by Facebook. React Native is essentially a NodeJS runtime on mobile. It provides a React-like interface for programming. It also allows you to deploy to both IOS and Android, with only one codebase. It could be decided that we only need to target Android devices to simplify the project, as this mobile app is not an integral part of the project. Android would be a good choice as the openness of the device APIs (notably Bluetooth) is However, it is not just a web rendering engine, the components made in React Native are true native UI components that you'd get using the platforms native toolkits. There are many 3rd party React Native component libraries that contain all sorts of premade components. A basic component library, NativeBase, is discussed in section 3.8.7.1. Because all our functionality can be done with React Native, it would ease development to go with this solution. There is a very easy way to get started with React Native, and that's through Expo. With Expo, you can generate everything you need and get started with a project right away. There is also an app to immediately view changes as you write them. It recompiles when you save a file and transfers the changes to your phone immediately for testing.

4.6.8.1 React Native Extensions

If it's decided to just target an Android device, there are many React Native modules available to interface with the Android native Bluetooth APIs. There is a React Native package called react-native-ble-plx which uses RxAndroidBle as a native library under the hood on Android, and RxBluetoothKit on IOS. Extensions like these will help keep development workload down, as one codebase can deploy to both IOS and Android, even using native APIs such as Bluetooth.

One of the bigger problems people seem to have with React Native is the built-in components. The components that come with it are very basic and don't look very good. There is a library called NativeBase, which supports IOS and Android. NativeBase provides React Native components for many common UI elements seen on both platforms, as well as many utilities to make development easier. With the components provided by NativeBase, we should easily be able to get up and running with a prototype app.

4.6.8.2 GPS Localization Visualization

As per the requirements, the robot needs to be able to be localized with GPS precision at any time. We can utilize the GPS module that will be in the robot for this task. Because the robot will not have a wireless data connection (connection to the internet), local wireless will have to do. We can have a simple server running to be queried by the mobile app connection, using whichever wireless protocol we end up using. This mobile app (also containing many other remote features, described in section 4.6.7-4.6.8) will allow a user to view the location of the robot overlaid onto a Google Maps-like interface. The app will also show a track of some previous location points, so the user can view where the robot has been previously and where it is headed. There will be a separate tab for this feature.

4.7 Wireless Communications

Wireless communications are an integral portion of the project. This is because there are many components to the autonomous lawn mower that is being designed that can utilize the capabilities and flexibilities that a wireless transmitter permit. Some examples of the benefits of wireless communication include boundary zone implementation, operating system application, video streaming of live time cutting the grass, and remote-control access to the autonomous lawn mower. The most important aspect of wireless communication is the ability to create a “kill switch,” both fulfilling a customer requirement and enabling a range of at least 50 feet to be able to power down the autonomous lawn mower. This switch will be an established communications link between the autonomous lawn mower and an application utilizing an operating system to send a command to the autonomous lawn mower to cease all functions, and essentially “power down.” For example, a remote kill switch function can be implemented by sending a “kill” command from an operating system application to the autonomous lawn mower through an established wireless communications link. The “kill” command would ultimately be received by the microcontroller, proceeding to send signals to the rest of the components to enter an immediate shutdown procedure. Utilizing wireless communications, it is also possible to program the autonomous lawn mower to send signals to the same operating system application (used to send the kill switch) to output statistical data, such as GPS location, battery information, troubleshooting information, and operating statistics. Modern wireless communications consist of mostly two standardized protocols: there is IEEE 802.11 specification of wireless local area networks (LAN) and Wi-Fi, and Bluetooth. Of all the available wireless communication formats, the technologies being discussed for use in this project are Bluetooth, Radio Frequency (RF), SPI, and Wi-Fi.

4.7.1 Bluetooth

Bluetooth is capable of providing wireless communication, similar to the ability of Wi-Fi. The core differences between the two is how they are designed to provide this communication and how they are used. Bluetooth is a wireless technology standard that is used to exchange data over short distances, typically no more than 30 feet. This technology is typically used between personal mobile devices, and is now beginning to see increased usage in personal computer devices as well, such as headsets, microphones, keyboards, etc. This means that a Bluetooth-enabled device is able to communicate with other devices, such as the mentioned headsets, and even printers. Bluetooth allows for a personal area network (PAN) to be created between devices.

Spun off from IEEE wireless standards, Bluetooth uses 2.4 gigahertz (UHF) radio frequency to transmit and receive. There is a class of Bluetooth that allows for a much higher range than the previously mentioned 30 feet range, however, the cost comparison to the other methods listed for wireless communication puts it at a negative price-point-ratio. Due to the customer's kill switch requirement requiring a minimum of 50 feet to function, Bluetooth would not be an acceptable technology to use for implementation of the required kill switch on the autonomous lawn mower. The range limitation of Bluetooth provides a significant drawback when comparing the power consumption comparison of Bluetooth to Wi-Fi. The following table, Table 25 provides a comparison between Wi-Fi and Bluetooth.

Table 25: Wireless Communication Comparison – Bluetooth vs. Wi-Fi

	Wi-Fi	**Bluetooth**
Key Elements	- Operates at 2.4-5GHz - High Cost - Up to 11Mbps bandwidth - 600Mbps Bit Rate - High Power Consumption - High Range	- Operates at 2.4GHz - Low Cost - 800Kbps bandwidth - 2.1Mbps Bit Rate - Low Power Consumption - Low Range

4.7.2 Radio Frequency

Wireless Radio frequency, or RF, is any of the electromagnetic wave frequencies that range from approximately three (3) kilohertz to 300 gigahertz. This range is the range of frequencies used for radar signals, or communications. The most common devices that utilize radio frequency fields are cordless phones and cell phones, radio and television broadcast stations, satellite communications systems, certain Bluetooth modules, Wi-Fi, and two-way radios. Other appliances also use RF for external communications, such as garage-door openers and microwave ovens. Because the spectrum for RF is so wide, it is also possible to communicate on a much smaller scale, such as the RF that television remote controls use, or some cordless keyboards use.

Radio Frequency is measured in hertz, or Hz, which is the number of cycles per second when a radio signal is transmitted. As the frequency is enhanced on the RF spectrum, electromagnetic energy changes from infrared to ultraviolet, eventually reaching gamma rays. Radio frequency can be applied to different types, sizes, and shapes of small electronic circuit boards. In order to operate one of these modules, a custom printed circuit board (PCB) is required, with an antenna soldered on in order to transmit or receive data with the host processor. The different types of RF modules include Transmitter module (TX), Receiver module (Rx), transceiver module, and System on a Chip (SoC) module. Radio frequency spectrum is separated into numerous ranges, or bands. Each band denotes an increase of frequency. The table below, Table 26 displays the RF spectrum of frequency ranges and accompanying bandwidths.

Table 26: Radio Frequencies and their ranges

Designation	Abbreviation	Frequencies	Wavelength Range (Distance)
VLF	Very Low Frequency	3 kHz – 30 kHz	100 – 10 km
LF	Low Frequency	30 kHz – 300 kHz	10 km – 1 km
MF	Medium Frequency	300 kHz – 3MHz	1 km – 100 m
HF	High Frequency	3 MHz – 30 MHz	100 – 10 m
VHF	Very High Frequency	30 MHz – 300 MHz	10 – 1 m
UHF	Ultra-High Frequency	300 MHz – 3 GHz	1 m – 10 cm

SHF	Super High Frequency	3 GHz – 30 GHz	10 – 1 cm
EHF	Extremely High Frequency	30 GHz – 300 GHz	1 cm – 1 mm

Frequencies above 1 GHz and above are conventionally called microwave, while frequencies of 30 GHz and above are considered millimeter wave. By increasing the power of the transmitter, a larger communication distance can be achieved. However, this will result in a higher electrical power drain on the transmitter device. This consequently also opens up the system to being susceptible to interference with an extra RF device. Increasing the sensitivity of the receiver will boost the communication range but will also be open to potential interference with additional RF devices. Radio frequency devices have the best success when being used in an open-air line of sight view. However, due to the nature of the customer's request to operate in a Solar farm, where there are cornucopias amounts of interferences in the form of support beams, the solar panels itself which could potentially bounce the radio frequency signal around, etc, radio frequency is not the best method for transmitting data, or even a kill switch, as reliability becomes an issue. Even though RF tends to be a low power consumption method of wireless communication, the low average range of approximately 30 meters does not support the customer's requirement on the kill switch range. At most, RF could be used for aiding in determining the positioning of the autonomous lawn mower at any given point.

4.7.3 SPI Protocol

SPI stands for Serial Peripheral Interface. SPI was first introduced by the company Motorola. SPI is used to communicate a single master device output to a multitude of slave devices. The easiest implementation of SPI is by using it on an Arduino system. There are four wires required to be used to implement an SPI. The first wire is SCK, or Serial Clock wire. This is the wire that propagates the clock signal from the master device to the slave device(s). The second wire is MOSI, or Master Out-Slave In. This wire is used as the output from the master device and an input to the slave device. The MISO line, or Master in-Slave Out wire, is used as input to the master device, and output from the slave device. The last wire is the SS, or Slave Selection. This wire will send the signal to the slave that will be selected for data transmission.

4.7.4 Wi-Fi

Similar to Radio Frequency, Wi-Fi is a type of radio frequency technology used for wireless local area networking (WLAN). This version of wireless communication is based on the IEEE 802.11 standards. There have been several versions of Wi-Fi that have been created since its conception in the late 1990's. Each standard of Wi-Fi has its own version name, denoted by the standard version and specifications. Modern Wi-Fi most commonly uses 2.4 gigahertz (UHF) or 5.8 gigahertz (SHF) radio bands to communicate between devices. Table 27 below lists the most common formats of Wi-Fi that are used today, and the specs that will be used for comparison within this project.

Table 27: Wi-Fi versions able to perform in the Autonomous Lawn Mower

802.11 Protocol	**Stream Data Rate (Mbit/s)**	**Outdoor Range**
802.11b	1, 2, 5.5, 11	140m (460ft)
802.11g	6, 9, 12, 18, 24, 36, 48, 54	140m (460ft)
802.11n	Up to 288.8	250m (820ft)
802.11n (5 GHz)	Up to 600	250m (820ft)

The frequency range, or type, for all of the Wi-Fi 802.11 protocols in Table Y range from one to six gigahertz (1-6GHz). Wi-Fi protocols 802.11b, 802.11g, and 802.11n were chosen to be compared over all the other Wi-Fi protocols that are capable of higher data rates because most of the data sent will likely be within a 2.4 gigahertz (UHF) range. The autonomous lawn mower will most likely be communicating parameters wirelessly that enable the functions of a kill switch at the most basic level, while at the highest of data transfer levels the autonomous lawn mower will be communicating location, or other base level data. The plans for a live video stream from the lawn mower are currently not in the realm of desired data transfer, therefore higher protocols of Wi-Fi are not required, even though with the current selections the possibility of video streaming is indeed possible. The most likely scenario for the autonomous lawn mower is the autonomous lawn mower transmitting and receiving over a 2.4 gigahertz (UHF) radio frequency band, unless the 5.8 gigahertz (SHF) is chosen under the 802.11n version.

Using Wi-Fi creates a wireless local area network (WLAN), which is possible to be connected to a local area network (LAN) that can be connected to a metropolitan area network (MAN), or even connected to a wide area network (WAN). With the flexibility of connections over Wi-Fi, it is possible to establish communications to the autonomous lawn mower from remote locations as far as from another state, or even farther. Although possible, establishing this wide area network would require additional networking equipment, but the possibility is still there if desired by the customer. It is doubtful that the WLAN created for the autonomous grass cutter will connect to a WAN or MAN through the solar farm's networking equipment, because network security problems could arise, or the solar farm could potentially not support outside internet access. However, using the WLAN created for the autonomous lawn mower enables other devices the opportunity to connect within a range of 140 to 250 meters. Since the kill switch requirement is of at least 50 meters, using a kill switch implemented with Wi-Fi will meet the requirements requested by the customer.

5 System Design

After reviewing and researching different technologies for parts and development for the project, final decisions were made on the system design and structure. These decisions will be shaped and molded into the final integration of the prototype. It will also give us a ground and source for on the final integration and testing of the system in order to be successful. In this section, the team will identify and describe the decisions created in order to achieve the goals set out for the project.

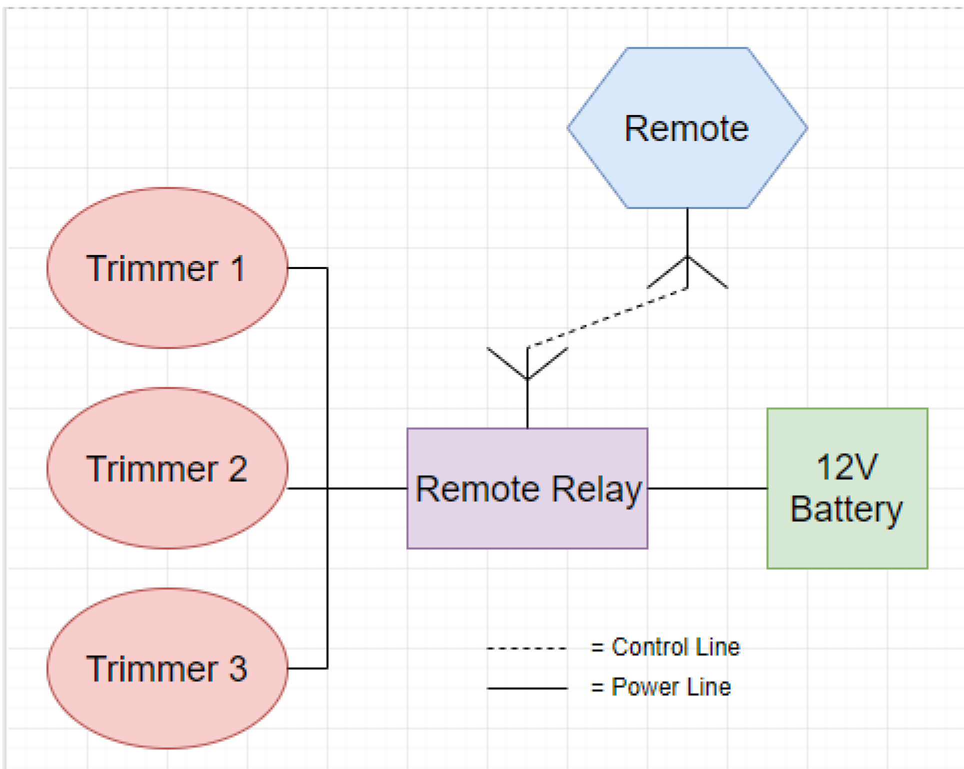
5.1 Power Systems

All total power system will be made up of all 12 volt batteries. Four Ovonic 8000mAH Lipo batteries, and one GTK lead-acid 15000mAH battery. The plan is to use three of the Lipo batteries for the trimmer motors, the lead-acid battery for the wheel motors, and then the last Lipo will be used for the single board computer/ microcontroller PCB/ other electronic accessories. This should more than make up for our goals as far as operation time of the mower is concerned.

5.1.1 Trimmers

The figure below shows the power management system for all three trimmers. They will all be tied to the same power source, however, there will be remote switch in between them acting as an open whenever it receives a signal from the remote control.

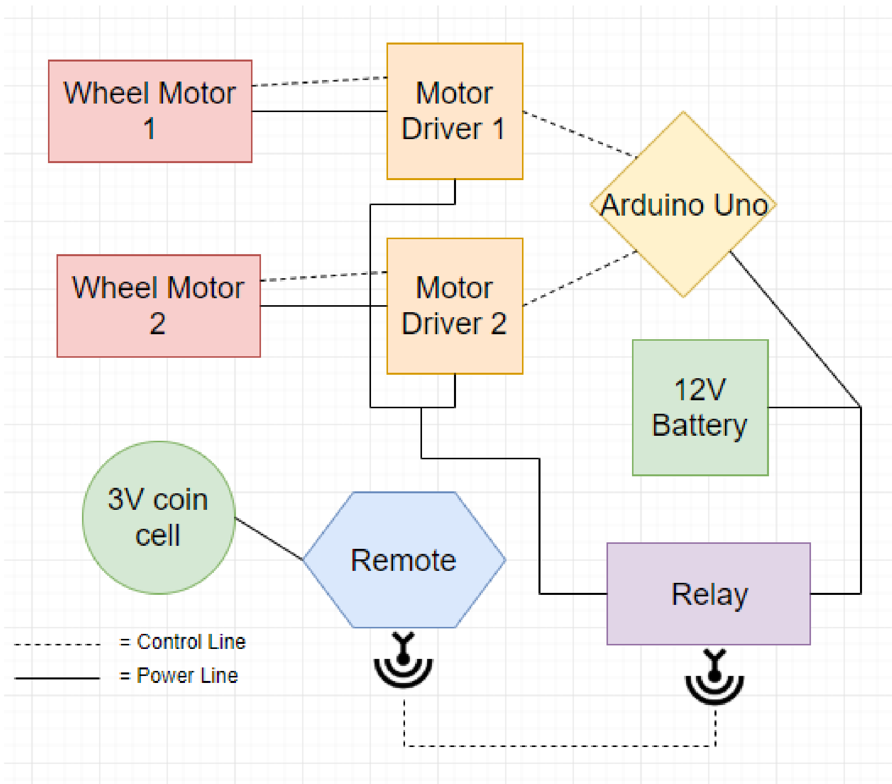
Figure 18: Power system for trimmers



5.1.2 Wheels

This figure shows the power management system for the wheel motors, which is a little more complex. Basically, just like the trimmer system, there is a remote relay in between the battery and the drivers that will act as a kill switch (which should work anywhere from 30-50 meters). Also included is the control line which will initially come from the Arduino Uno microcontroller, which will in turn tell the driver what to do so that it can take however much Vcc it needs to control the wheels for that given command.

Figure 19: power system for wheel motors



5.1.3 Accessories

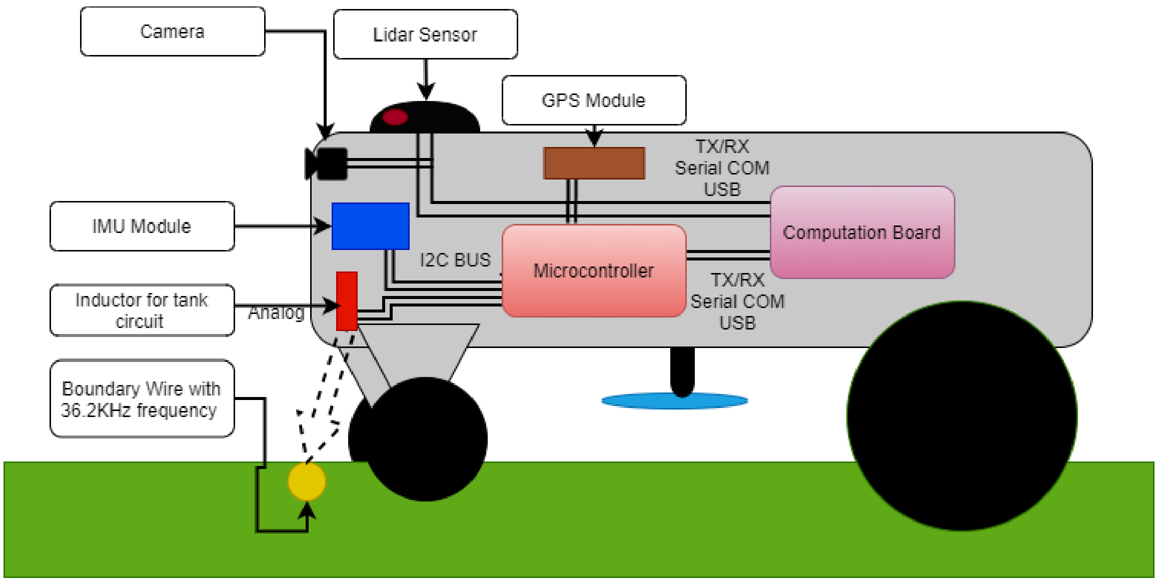
The last power system is the accessories. All it includes is a 12V battery being taken to a voltage regulator to drop it down to 5V. And then sending it to all the accessories that take that voltage such as the MIC, camera, lidar, and Odroid.

5.2 Navigation System

Based on the navigation technologies selected in section 4.7 Navigation Sensors, these subsystems are incorporated in the project in order to meet the projects goal. The electric fence subsystem keeps the rover out of the boundary and the IMU and GPS assist internal navigation within the lawn. 5.2.1 Boundary Wire Subsystem Design.

The boundary wire subsystem is comprised of two critical sectional parts: the function generator with perimeter wire and EMF sensor attached the physical rover unit for data processing. It is critical that both systems work properly in order for the boundary constraints to be held and the rover not end up in an illegal position that could harm or damage its users or equipment.

Figure 20: Concept drawing of Autonomous Lawn Mower



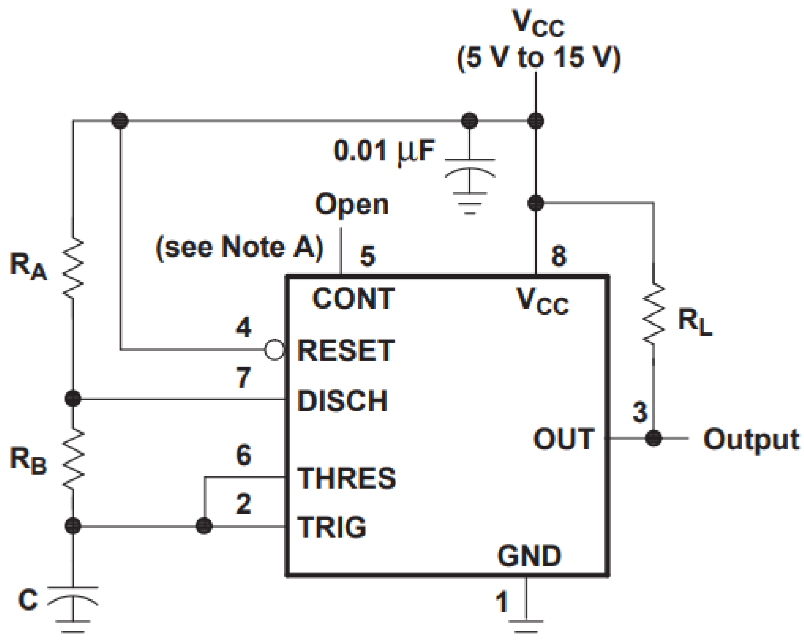
5.2.1 Function Generator Design

Discussed in this section is the design of the Function generator section as part of the Boundary wire subsystem. This is technical detail calculations, schematics, and PCB trace created and procured by the team.

5.2.1.1 Function Generator Design Calculations

Using the parts selected for the function generator, an initial prototyping sketch has been gained from the NE555 manual to understand the connections for the NE555 Timer Circuit[30].

Figure 21: Function Generator Design



To set the duty cycle and frequency of the square wave, we use the following equation:

$$(1) \quad f = \frac{1.44}{(Ra + 2 * Rb) * C}$$

Where f is the frequency desired, Ra is the resistor in series with Rb and C. We would like a frequency between 32KHz and 44KHz as experience in other projects, these frequencies should not interfere with our GPS, Wi-Fi, or other modules onboard the rover. We also use a potentiometer to change the frequency to match the resonance frequency of the receiver circuit. The potentiometer has a resistance of 12Kohms with an additional choice of + 4.7Kohms. Using equation (1) we calculate our upper bound and lower bound frequency fL and fU using the max and min resistance of the potentiometer:

$$fL = \frac{1.44}{(3.3 + 2 * (12 + 4.7)) * 1.2e10\hat{}(-9)} \approx 32.698KHz$$

$$fU = \frac{1.44}{(3.3 + 2 * (12 + 0)) * 1.2e10\hat{}(-9)} \approx 43.956KHz$$

When we increase resistance of Rb, we reduce the frequency of the timer while keeping the gain constant. When we decrease the resistance of Rb, we increase the frequency of the timer while also keeping the gain constant.

5.2.1.2 Function Generator Schematic and PCB Design

With these calculations in place, we can create a schematic and PCB design to reflect our design. Using a schematic provided by RobotShop.com and using provided documentation from previous year's generation 1 PCB design, we are able to create our own reference PCB design[cite].

Figure 22: PCB Schematic Design 1

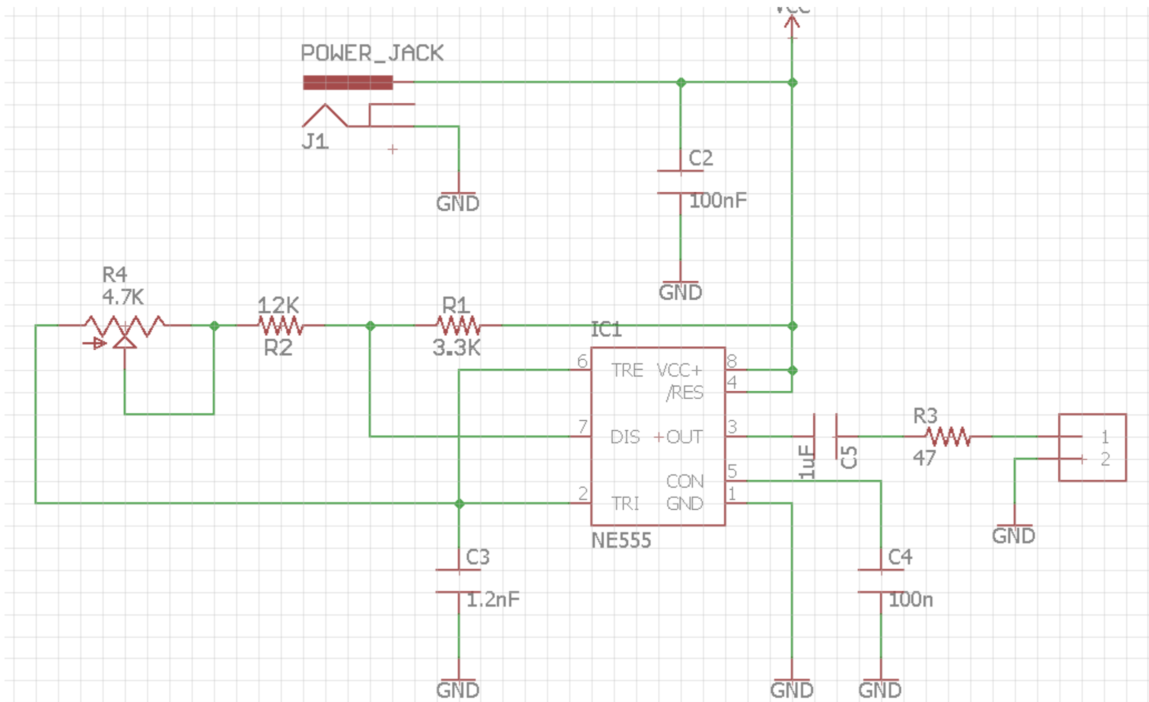
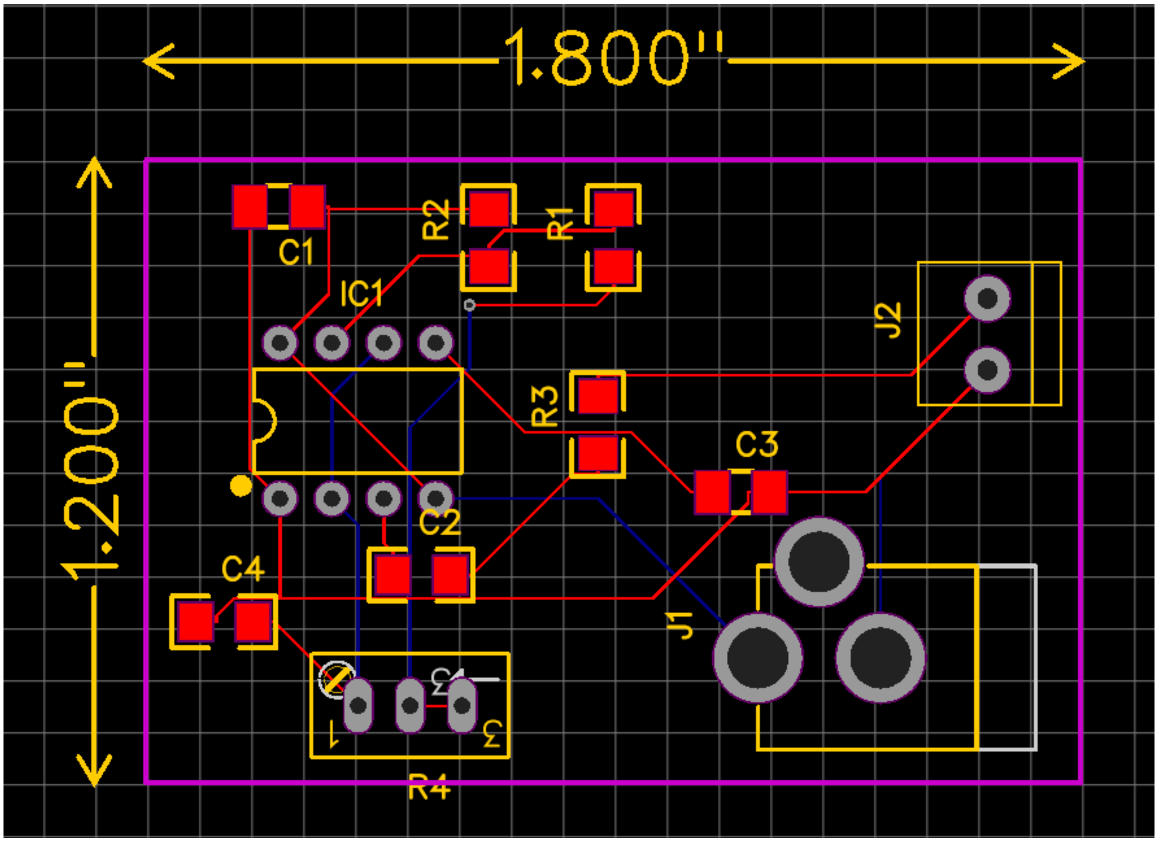


Figure 23: PCB Schematic Design 2



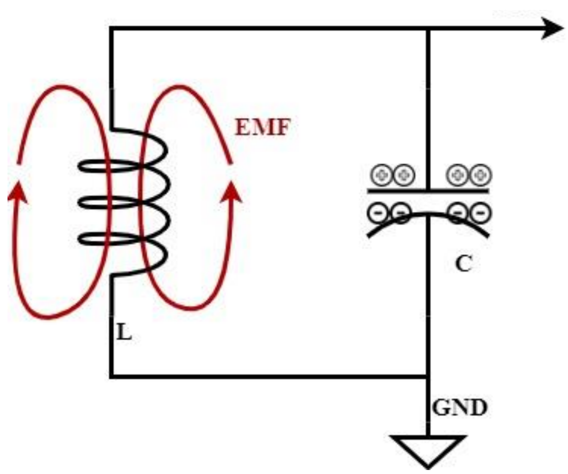
5.2.2 EMF Sensor Design

Discussed in this section is the design of the EMF Sensor section as part of the Boundary Wire subsystem. This is technical detail calculations, schematics, and PCB trace created and procured by the team.

5.2.2.1 EMF Sensor Design Calculations

To detect our square wave produced by the function generator in the previous section, we needed a circuit that picks out the created frequency and determine if we have tripped the wire or not. This way. It can act as an electrical resonator to store the energy of the frequency being emitted and we can measure this as an analog signal into the microcontroller. This kind of filter needed would be an LC filter or so called a resonant filter. A resonant filter consists of an inductor and a capacitor in parallel or series that through Faraday's Law and the drop of magnetic field, the capacitor is charged up at the resonance frequency of the circuit.

Figure 24: EMF Sensor



Calculating the resonance frequency based on the frequency outputted by the function generator for parallel LC circuit, we get:

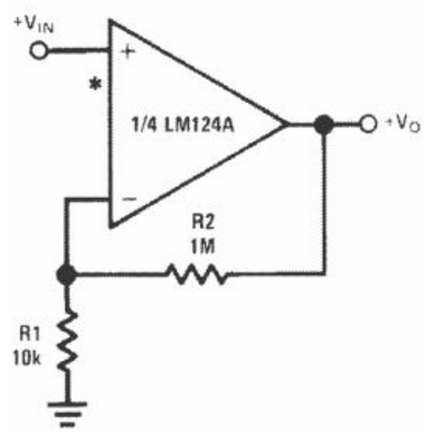
$$f0 = \frac{1}{2 * \pi * \sqrt{L * C}}$$

Where L is the inductance value of the coil in Henry (H) and C is the capacitance value measured in Farads (F) Fixing the capacitance for a small capacitor at 22nF, we get an inductance value of L = 1mH at the resonance frequency of 33.932KHz.

The voltage amplitude of the capacitor is fairly small due to the change in the magnetic field. With such a small change, the microcontroller chosen is not capable to detect a small voltage amplitude. To increase the amplitude, we also use an Op-Amp to increase the signal that can be read to the microcontroller. The chosen Op-Amp, the LM324, contains four Op-Amps but we only make use of two. To achieve a non-inverting gain of 100, we connect two resistors in series with the Op-Amp to specify the desired gain. Using the data sheet provided [cite]we get:

Where $R1 = 10KOhms$ and $R2 = 1MOhms$ for a non-inverted gain of 100.

Figure 25: Op Amp Design



5.2.2.2 EMF Sensor Design Schematic and PCB

With these calculations in place from the data sheet and LC circuit, we create a schematic and PCB design to reflect our design. Using a schematic provided by RobotShop.com, and we create our own reference PCB design[cite].

Figure 26: EMF Sensor Design 1

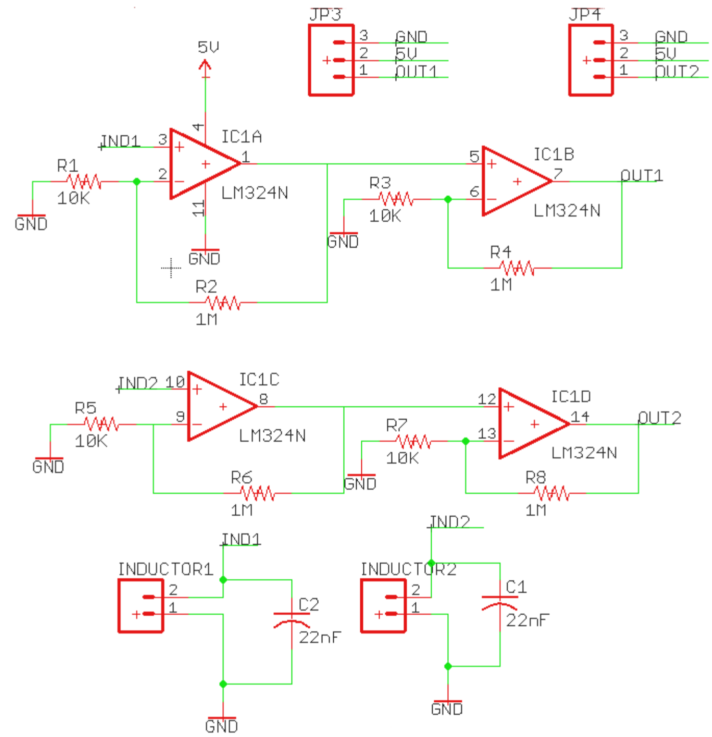
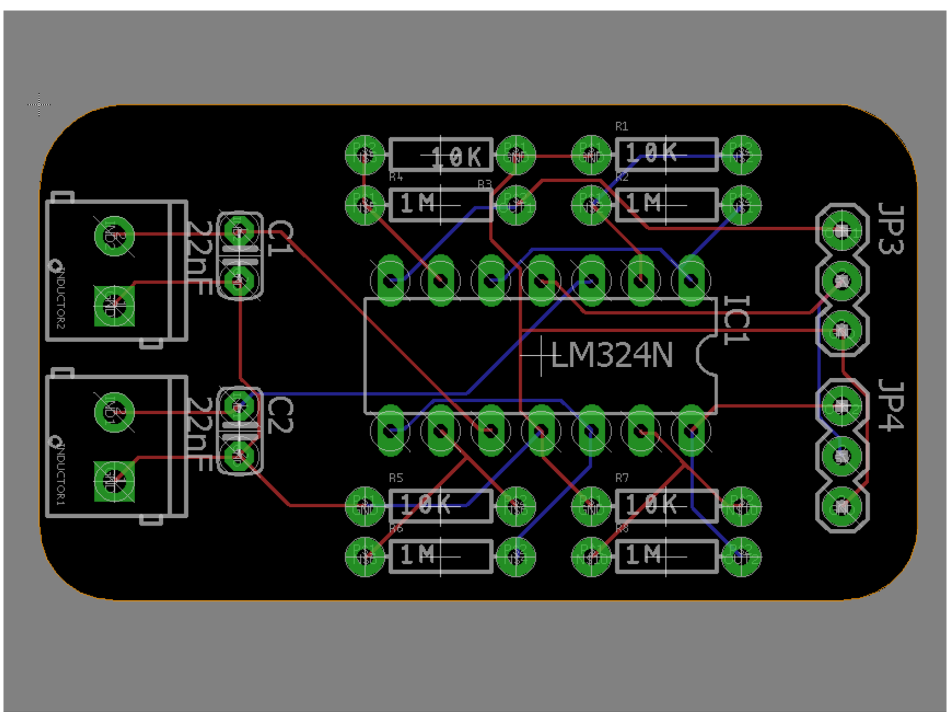


Figure 27: EMF Sensor Design 2

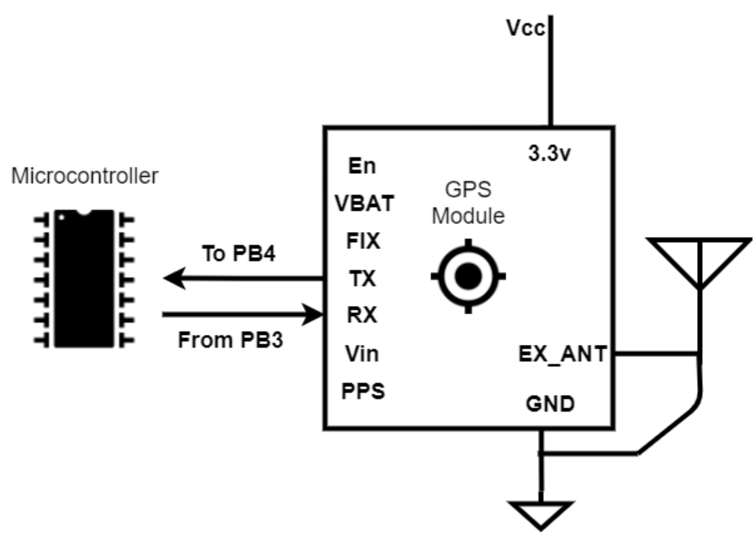


5.2.3 GPS Design and Schematics

The GPS module chosen for this project as mentioned in section 4.7.4.1 is the MTK3339 GPS module. This module needed its own serial communication pins connected to the microcontroller board in order to use the NMAE packet communication protocol. We also needed a Vin of 3.3v regulated from either the microcontroller itself or an external regulated power supply (for example from the on board LiPo batteries regulated through another device)

To increase the signal coverage that the GPS module can receive, the module is extended with a large antenna. The antenna is mounted on the back of the rover outside of the main body to reduce interference within the electronic chamber and reduces interference from the shielding of the rover. The GPS module maintains an internal UTC clock within its module and a data log of previous locations every ten seconds. In order to decrease the warmup time and always have an accurate UTC time clock on chip, a constant Vcc of 3.3v will need to be supplied to the module itself while it is on low power mode. Based on the development board, we put a 3.3v CR1220 coin battery on the bottom of the created PCB in order to keep this log and time.

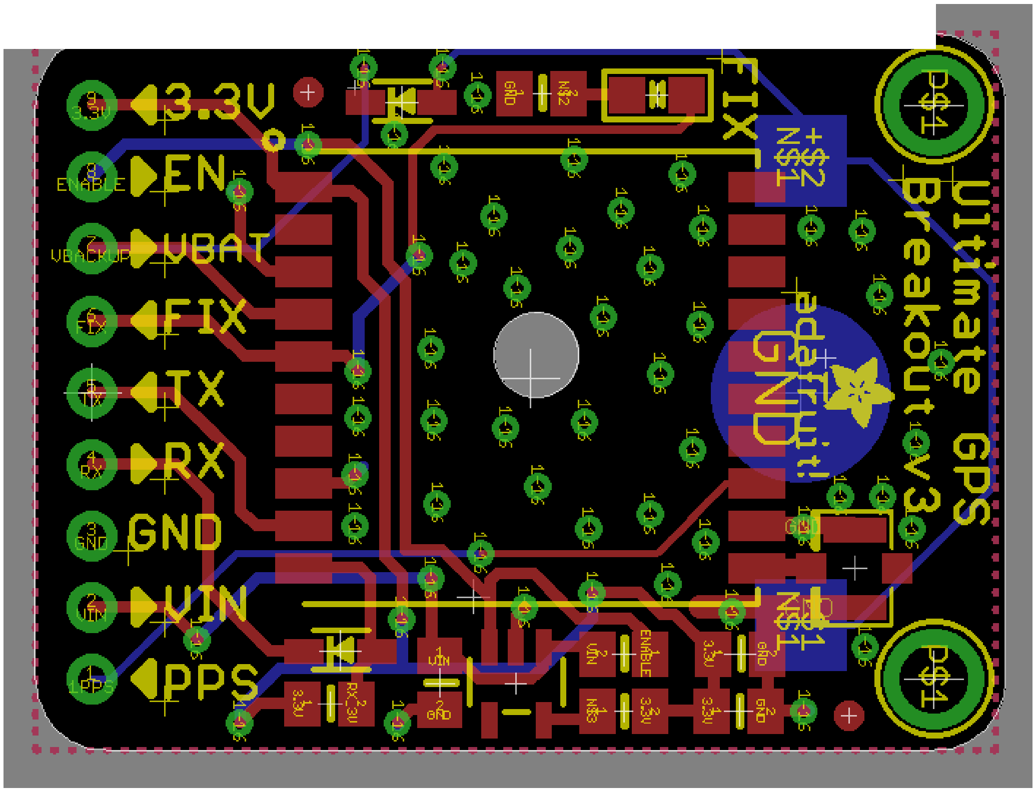
Figure 28: GPS placement Design



5.2.3.1 GPS Schematic and PCB Design

Based on the Adafruit Ultimate Breakout for the MTK3339 GPS Module, we created a PCB design that implements the above design to receive GNSS signals and communicate the position with the MCU.

Figure 29: GPS Schematic and PCB Design



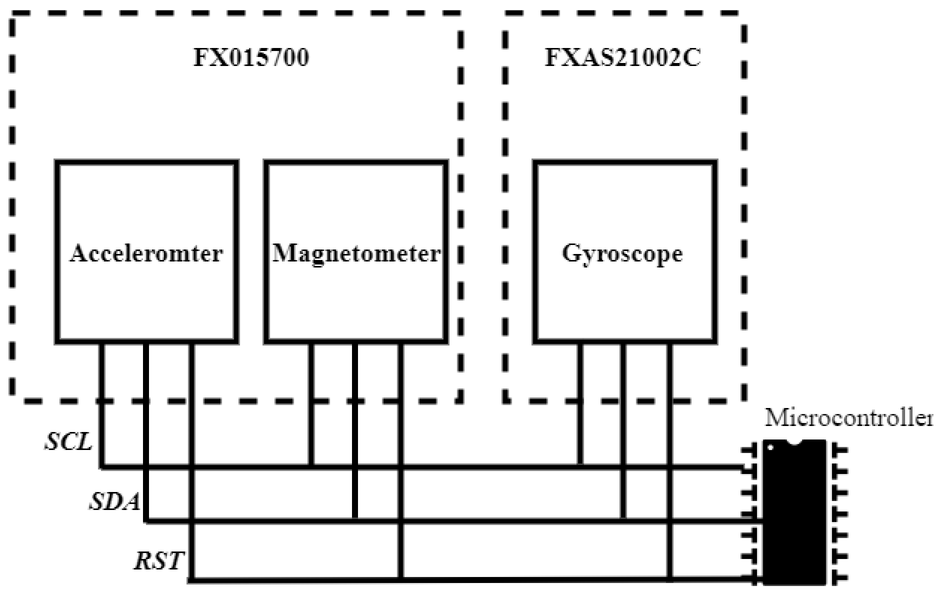
5.2.4 IMU Design and Schematic

Discussed in this section is the design of the IMU Sensor modules as part of the microcontroller Localization subsystem. This is technical detail calculations, schematics, and PCB trace created and procured by the team.

5.2.4.1 IMU Design

The IMU module for this project is the FXOS8700 6-Axis sensor Accelerometer & Magnetometer and the FXAS21002C 3-Axis Digital Angular Rate Gyroscope as discussed in both sections 4.7.4.3 and 4.7.4.3. The IMU sensor consists of three distinct chip modules that can be interfaced using the IC2 standard. When working with the sensors, it is important to correctly wire the devices in order to enable proper communication through IC2 to the microcontroller.

Figure 30: IMU Wire Diagram



5.2.4.2 IMU Communication

As previously stated, the IMU modules will communicate via the IC2 standard through both the FMX015700 and the FXAS21002C. Both modules contain a unique 8-bit address that will be used to address the modules from the controlling master unit. This ensures that the data received and called for from the units are the correct packet readings and are properly handled by the microcontroller. The FMX015700 accelerometer and magnetometer is capable of single-byte read and write with an additional feature of multiple-byte write. There are several slave addresses that can be assigned to the modules through the SA1, SA0, GA1, or GA0 ports assigned on the modules by raising them high or low.

5.2.4.3 IMU Schematics and PCB Design

Based on the Adafruit NXP-Precision 9dof Breakout Board, we have implemented our physical designs. We used the physical IMU sensory breakout board provided by Adafruit for this project due to the modularity and easy integration within our project. The board consists of both the FMX015700 and the FXAS21002C with several pin in and outs for the IC2 communication line and 3v3 DC power in.

Figure 31 IMU Schematic

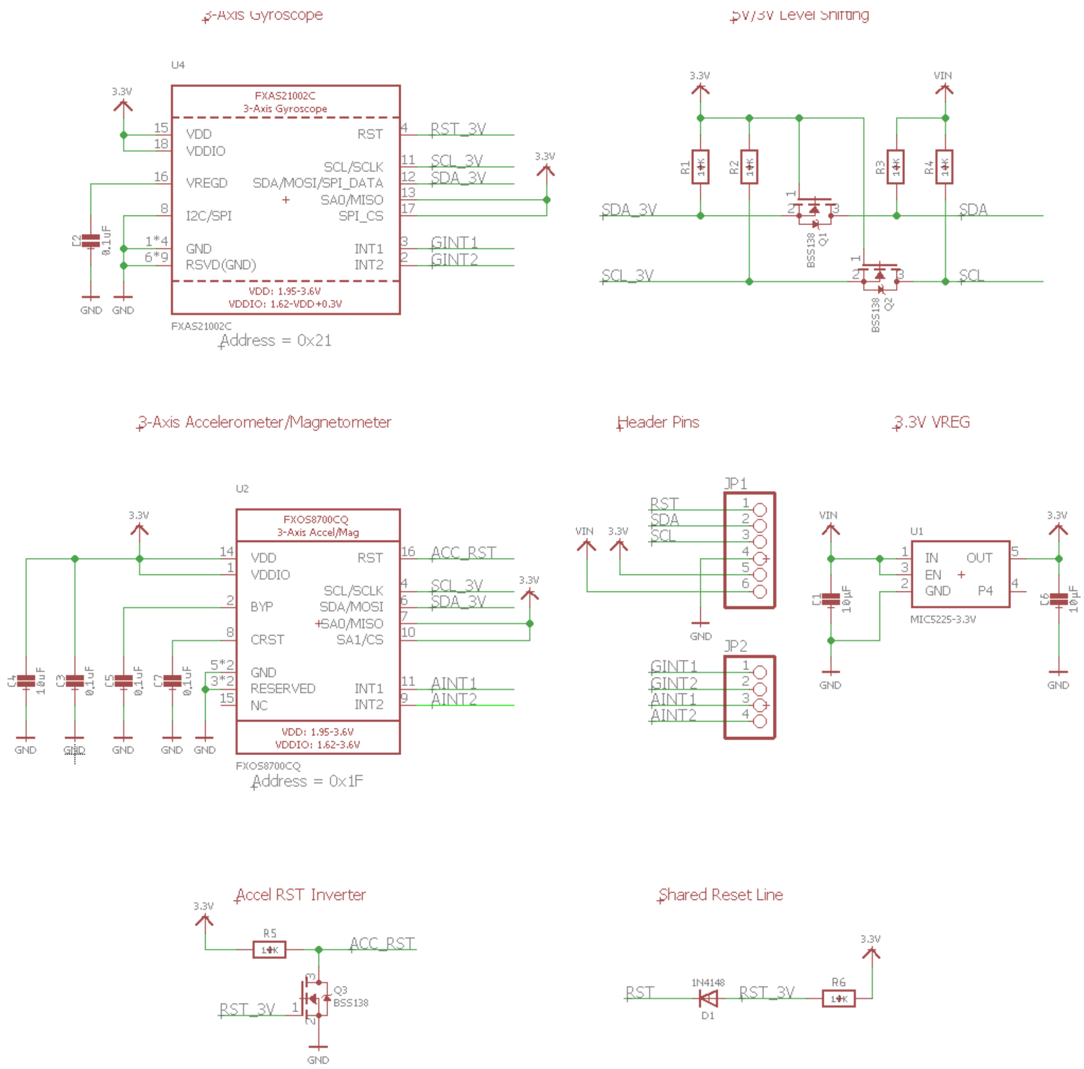
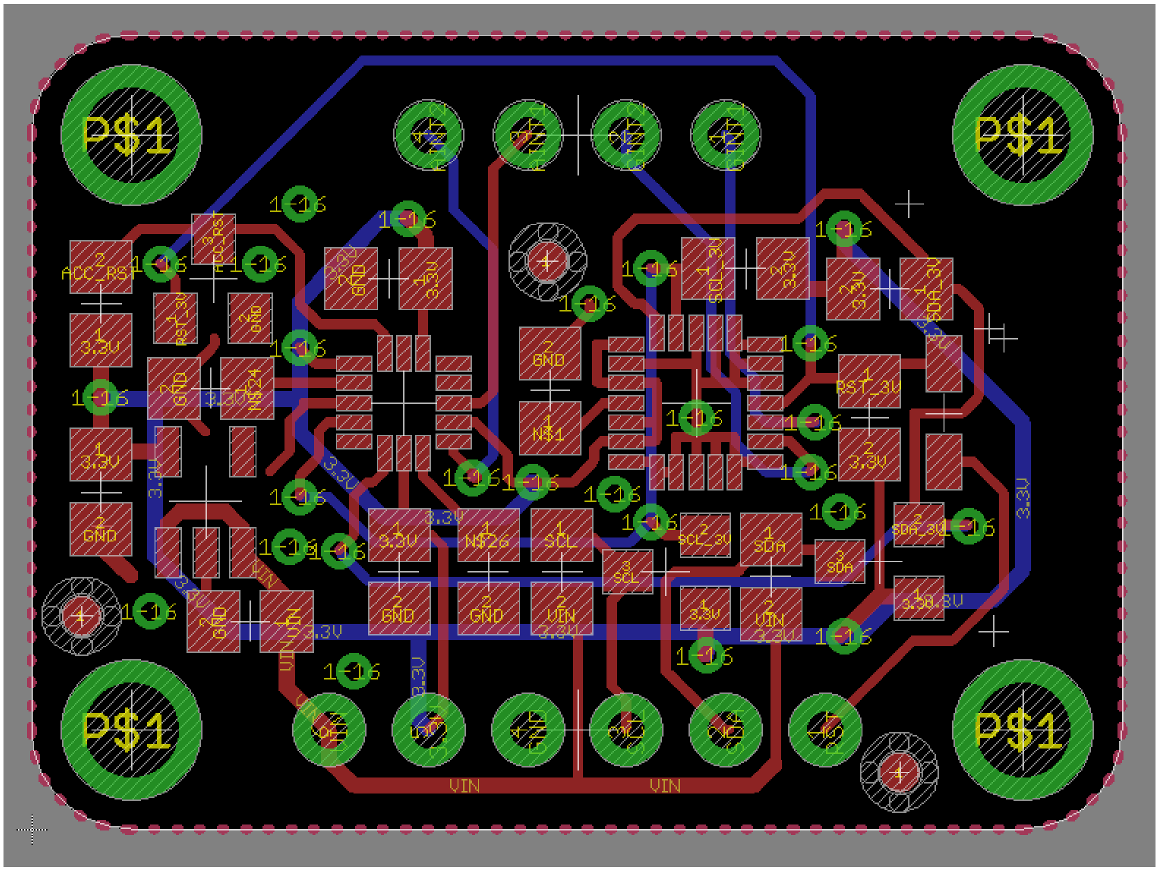


Figure 32: IMU Board Layout



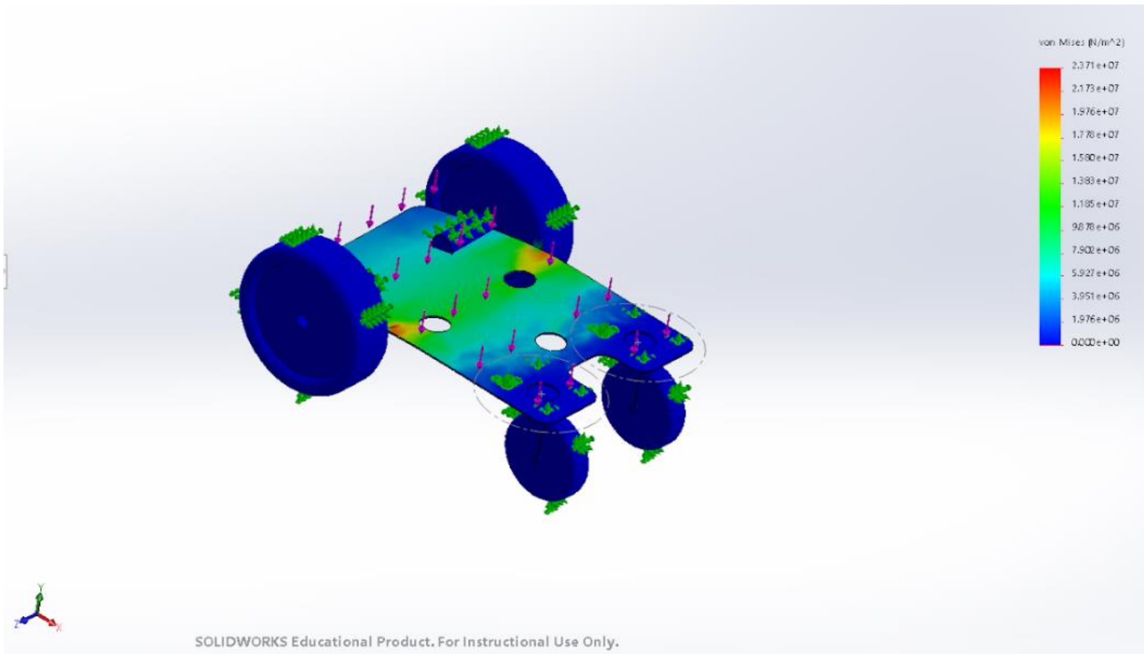
5.3 Physical Design

In collaboration with the Mechanical Engineering team through the department of Mechanical Engineers at UCF, physical design for key elements of the rover such as wheels, casters, body, and overall chassis design will be contributed by this team. The Physical design will be created using AutoCAD software and sourced from various shops and manufacturing facilities including University of Central Florida’s Department of Engineering Machine shop. In order to fulfill the physical standard requirements by the customer, the autonomous vehicle must be in a form or shape of a rover that will satisfy the grass cut criteria and conform to the constraints of height to ensure no damage comes to the solar panels.

5.3.1 Base Assembly

The frame of the mower is going to be cut from aluminum. The frame needs to be sturdy and strong enough to allow all the components to be mounted to it as well as be able to bear the weight of all its contents. UCF has a special machine shop that will be perfect for getting the sheet of aluminum cut. The frame has been designed by the Mechanical Team through SolidWorks and will include variables such as stress testing and material comparison.

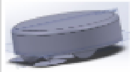
Figure 33: CAD Stress Testing



5.3.2 Shell

For the shell of the mower, we are planning on having it 3d printed. UCF has 3d printers open to students to be used for just an occasion, but the dimensions of the 3d printer limit us on size so if we end up using UCFs printer, we will have to print it into parts and then combine the parts. Another option would be to get in contact with some of the members of a FamiLAB who have a space here in Orlando. Some of the members have access to much larger 3d printers capable of printing much larger things and would most likely print the shell of the mower in one print which would make it structurally much more sound and more aesthetically pleasing.

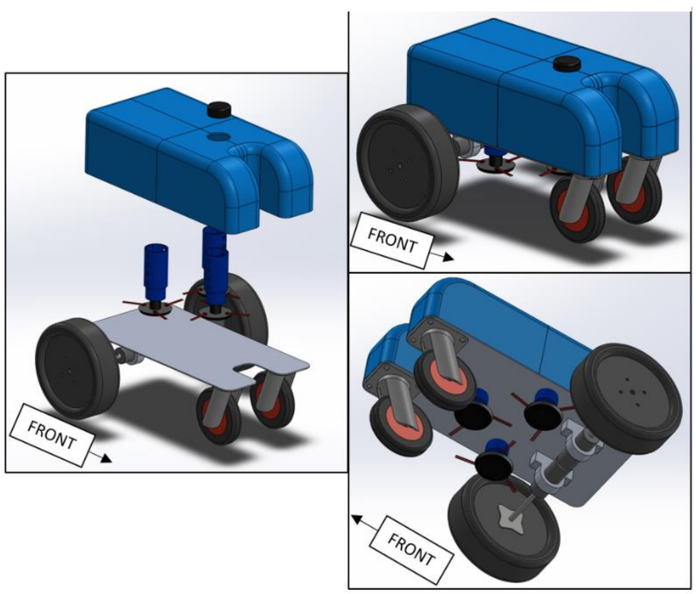
Table 28: Body Shell Comparison

Criteria	Weight	Evaluation			Score		
		Concept 1 	Concept 2 	Concept 3 	Concept 1 	Concept 2 	Concept 3 
Cover Height	3	2	3	3	6	9	9
Appearance	2	3	2	2	6	4	4
Manufacturability	5	2	2	3	10	10	15
Total Scores					**22**	**23**	**28**

5.3.3 Wheels

The wheels are put into two different categories: The driving wheels and caster

Figure 34: CAD rendering



wheels. The driving wheels need to be big and robust and have deep treads to be able to get better traction on whatever terrain it's driving over. The caster wheels on the other hand need to just support the body. They need to be smooth and easily roll over whichever terrain they're on as to cause the least amount of drag to the system. There were many options to choose from so once again we went to the comparison chart to make the decision. The final choices we made for both were great decisions that had everything we were looking for as well as coming at a reasonable price.

5.3.3.1 Drive Wheels

The caster wheels will help support the rover to be more stable and aid with maneuvering while performing the task. Contenders were chosen and a winner was carefully selected from the three. With the advantage of having a prior caster wheel, the selection process was much easier since the main criteria for this was to keep the rover balanced. This was already met by the prior caster wheel.

Table 29: Driving wheel comparison

Criteria	Weight	Evaluation			Score		
		TD-164-010 	TD-161-006 	Oregon 72-108 	TD-164-010 	TD-161-006 	Oregon 72-108 
Tread	5	3	2	1	15	10	5
Cost	2	1	3	2	2	6	4
Size	3	3	1	2	9	3	6
Total Scores					**26**	**19**	**15**

5.3.3.2 Caster Wheels

The driving wheels are a crucial portion of the rover. They need to support and transport the rover to different surfaces and terrains. Therefore, the drive wheels must have good traction and rigidity to withstand these tasks. The three contenders were selected and a winner of the three was chosen that best fit the criteria.

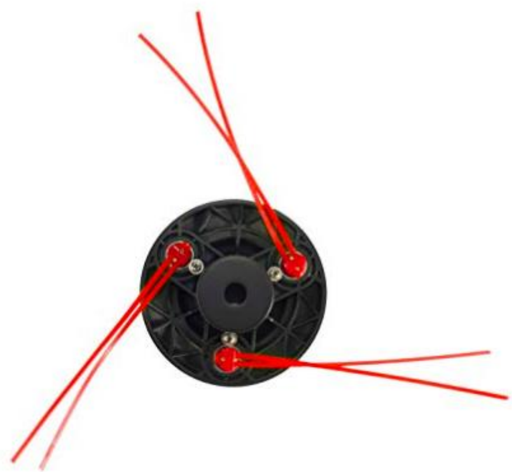
Table 30: Caster wheel comparison

Criteria	Weight	Evaluation			Score		
		Steelex D2580 	TPR 	Service Caster Brand 	Steelex D2580 	TPR 	Service Caster Brand 
Height	5	3	3	2	15	15	10
Durability	3	3	3	2	9	9	6
Cost	2	3	2	1	6	4	2
Total Scores					**30**	**28**	**18**

5.3.3.3 Trimmer Head

For the rover to properly cut the grass, a trimmer head was needed and coupled with the trimmer motor to work in unison. After carefully researched every best possible options, the Mechanical team has concluded with the proper trimmer head to be equipped with the rover. The MaxPower Pivot Trim is a universal trimmer head that is powerful enough to accept both 0.080” and 0.095” line thickness for the trimmer string. There are three pivoting lines to prevent breakage and is easily able to use. The head weighs 1.28 ounces and includes 6 pivoting lines. It is also an economical choice for around $20.11 each head.

Figure 35: Trimmer Head



5.4 Microcontroller System

Once we figured out the needs of the overall system, we determined that the ATMega328P would be more than sufficient for the goal we wished to accomplish with it. Below are the schematic and the board layout for the PCB that we will mount our microcontroller chip onto, as well as other components to compliment the needs of the microcontroller itself. This microcontroller, along with the single board computer will be used to receive data from the sensors and then process it, and then send signals to the motor drivers to let them know what the wheels should be doing.

Figure 36: Microcontroller PCB Schematic

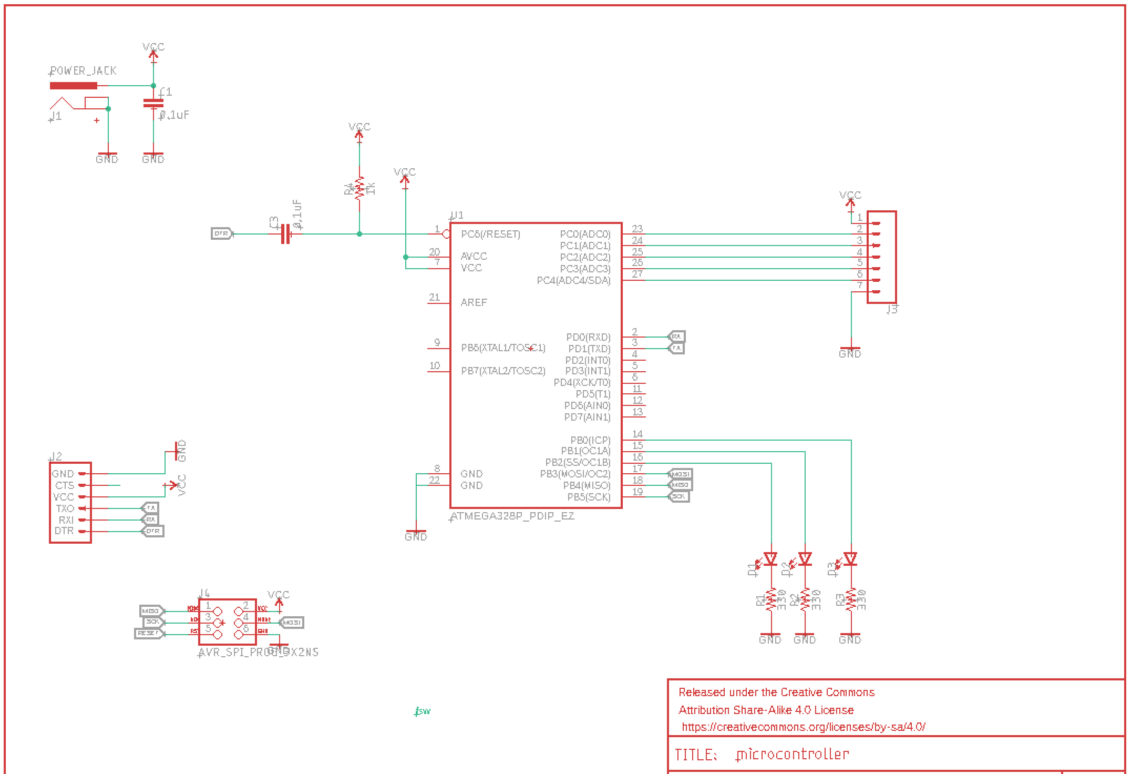
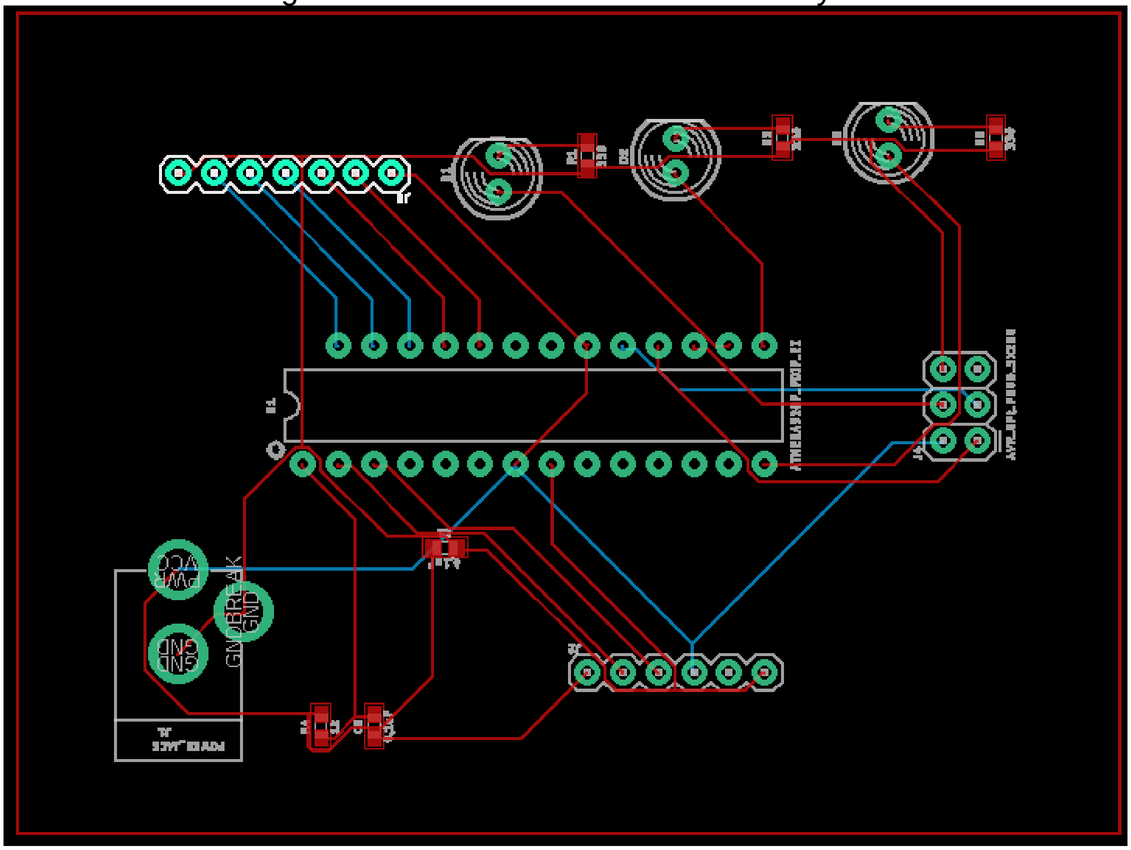


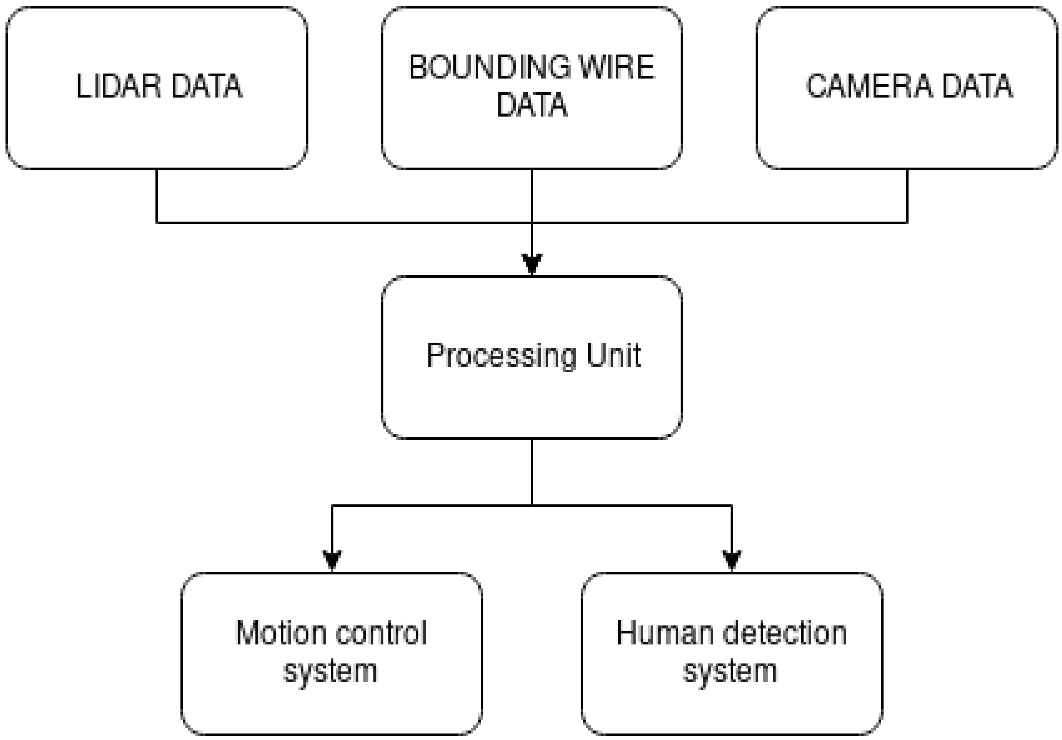
Figure 37: Microcontroller PCB board layout



5.5 Software Systems

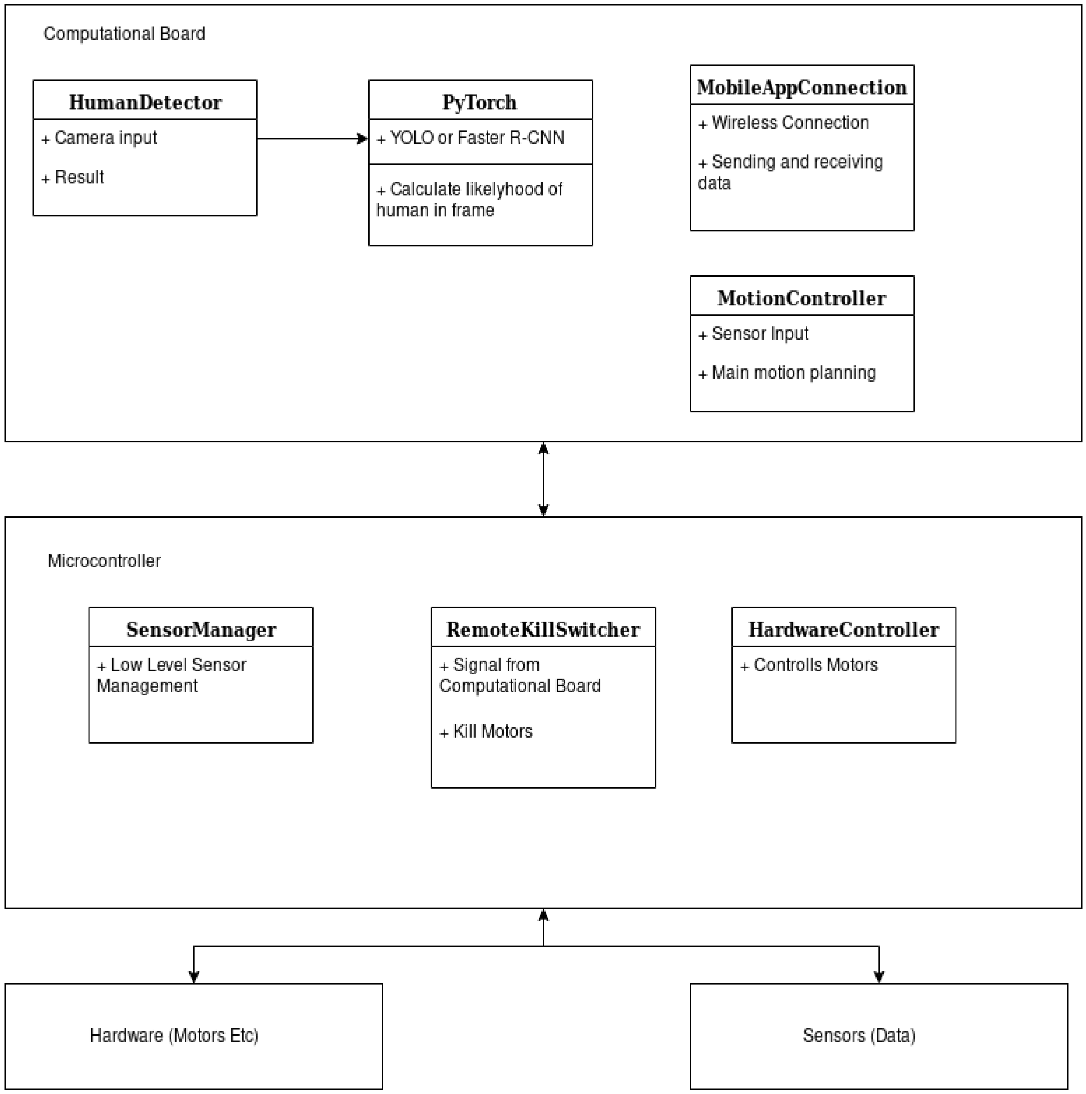
At a high level, as discussed previously, the low level microcontroller controls the hardware components, motors, and sensors. This microcontroller will be communicating with a higher level processor board (Odroid).

Figure 38: Software System Design



The Computational Board will be communicating and controlling the microcontroller. All information going to the outside world (Mobile App and Remote Relay) will be going through the computational board.

Figure 39: Software System Design 2



6 Demonstration and Testing

To show the customer that we have ensured and satisfied of their requirements and met the constraints of the program, demonstration and testing are key. The on-going process during development and integration is to demonstrate and test the capabilities of the product in order to show the internal team how much has progressed while at the same time showing the customer the end product and the satisfaction of their investment. Throughout the development phase, internal testing is important for the team to show strengths and build upon the incremental stages so that others are able build upon each other's work. When a flaw, bug, or failure is encountered, it a good sign of progress and recovery can be made and delegated to the appropriate team member. This section will go into detail and describe the various test methods, plans, and demonstration techniques that will be put into play in order to successfully show completion of the project.

6.1 Testing Plans

Testing plans involve testing the individual components in the lab first to make sure they function correctly such as the wheel motors, the drivers, and the microcontroller. Once we check all the individual components, we can start to test smaller subsystems to make sure they work well together correctly and check to see we have all the parts we need to make the smaller systems work correctly. Once all the subsystems have been tested and proven to work how we intend them to, we will begin to assemble the mower as a whole and slowly put all the parts together starting with the frame/wheels and then shifting to the motors and from there the more specific things such as the single board computer and the microcontroller. Once everything is gathered and assembled correctly, we need to take the mower outside to do field testing. We will see how the mower behaves and reacts in certain situations and then tweak whatever we need to until it is acting how we see fit. The goal is to get the mower to be as efficient as possible in its locomotion as to not go over the same area twice and lower the time it takes to mow a certain area. Outside of the goals we have internally, there will also be a formal competition between groups at UCF and groups at USF. The groups will compete to see who can mow a certain area of grass most efficiently in the least amount of time. There will be judges from Duke Energy as well as Orange County Utility. The competition will be combined with a presentation and after all the information has been taken into account, the judges will choose a winner. Last year's winner was the green team from UCF. Looking at their mower, it is clear that they went with a robust mower with large wheels and two commercial cutting blades attached to the front. Our goal will be to win the competition!

Figure 40: Competition site at Disney

6.2 Testing Goals

Testing goals include simple things first, like having the motor and the driver be able to talk to each other or having the code be able to turn the motor when we want it to. Once we get the little things in place it will be much easier for all the big things to come together. Our hope is to have the mower operational by February. We will be using the senior design lab to do most of our testing and also any open fields on the UCF campus.

6.2.1 Cutting Rate Test

To test the cutting rate of the rover a 10’ x 50’ area of grass will be marked out and the rover will be placed in the area to be timed to see how long it takes to complete cutting the area. Problems that might occur for this function may be due to the speed of the drive motors being too low, or the inefficiency of the path of navigation that the rover takes to cover the area. To address the issue, if it occurs, the speed of the drive motors could be increased or the path of navigation could be improved.

6.2.2 Cutting Height Adjustability Test

To test if the rover can have an adjustable cutting height of predetermined heights the rover will be placed on level ground and a ruler will be used to measure the height of the trimmer line from the ground. The problem that might occur is the measurement is incorrect. To help eliminate this issue the rover frame should be constructed before the height adjusting mechanism is made in order to measure the dimensions needed for the correct trimming heights.

6.2.3 Rover Stability Test

To test the stability of the rover, the rover will be placed on uneven ground to check if it will tip over. The wheels might be too close together, causing the stability of the rover to be compromised. To avoid this problem the design of the frame is to have the wheels as far apart from each other as they can be.

6.2.4 Turning Radius Test

The turning radius of the rover needs to be as small as possible for the best maneuverability. To test the rovers turning radius, the rover will be placed on level ground and controlled to turn around in the smallest radius the rover can complete. Some problems that might cause the rover to have greater than a zero-degree turning radius include the drive motors not being calibrated correctly with each other. This can be fixed with the motor controller. Other sources of error include the caster wheels interfering with each other or the trimmer heads interfering with each other. To avoid this problem the frame will be designed with consideration for the measurements of all components around the front steering wheels. These components will have sufficient room for the caster wheels to be able to spin completely and without interference.

6.2.5 Obstacle Avoidance Test

To test the rover’s obstacle avoidance, the rover will be placed in an area enclosed in boundary wire and with objects for the rover to navigate around. The rover might hit the objects if the location sensors and cameras are not calibrated correctly. To avoid this problem a different component may be selected or a different method will be implicated.

6.2.6 Autonomy Test

To test the autonomous abilities of the rover, the rover will be placed in a simulated test area and must perform cutting the entire area’s grass without harming people or the environment. This will also be completed without the assistance of a person. This feature may have many problems do to the complexity of this task. If this problem occurs it may be addressed by utilizing different components or a different method may be implicated.

6.2.7 Day/Night Run Test

To test if the rover can operate during the day and night conditions, the rover will be tested once during the day and once during the night. This feature may have many problems do to the complexity of this task. If this problem occurs, it may be addressed by utilizing different components or a different method may be implicated.

6.2.8 Weather Resistance Test

To test if the rover will be weather resistant, the cover will be lightly sprayed with water and checked to see if any water leaks through. Some problems may occur with this feature due to the cover being made in four pieces then glued together to make one shell. This problem may be avoided with a different design of the cover or the use of a larger 3D printer that can print the entire cover at once.

6.2.9 Battery Life Test

The rover needs to complete the task of cutting the area with one fully charged battery. To test this, the team will mark out an area of 500 square feet and have the rover trim the entire area without changing or recharging the battery.

6.2.10 Manual Kill Switch Test

The manual kill switch will be tested by the rover running in a test area, and a team member walking up to the rover and pressing the button to test if the kill switch stops locomotion and trimming system. Problems with this function may originate from poor connection from the switch to the drive and cutting system. If this problem occurs it may be addressed by utilizing different components or a different method may be implicated.

6.2.11 Remote Kill Switch Test

The remote kill switch will be tested by the rover running in a test area and at a distance of at least 50 feet a team member will press the remote kill switch to test if the kill switch stops locomotion and trimming system. Problems with this function may originate from poor connection from the switch to the drive and cutting system. If this problem occurs, it may be addressed by utilizing different components or a different method may be implicated.

6.2.12 Perimeter Wire Test

To test that the rover will remain within the bounds of the allocated cutting area, we will conduct sensory boundary tests for the perimeter wire subsystem within the rover. Outside of the final integrated rover, we will conduct breadboard tests in order to accurately measure the desired frequency and signal strength we desire. After implementing the subsystem with the software for feedback response, we will test the rover on the perimeter wire in order to ensure that the rover does not go out of bounds. We will also test subsystem components with large copper grade wire that will border the cutting area over 120ft.

For testing, we measured out 150ft of boundary wire in order to have extra room around the perimeter. We then tested the tank circuit within the lab to find the optimal frequency that the inductor would pick up the frequency from the wire. The following are two images depicting the testing phase of the perimeter wire.

Figure 42: Measurement of Sensning

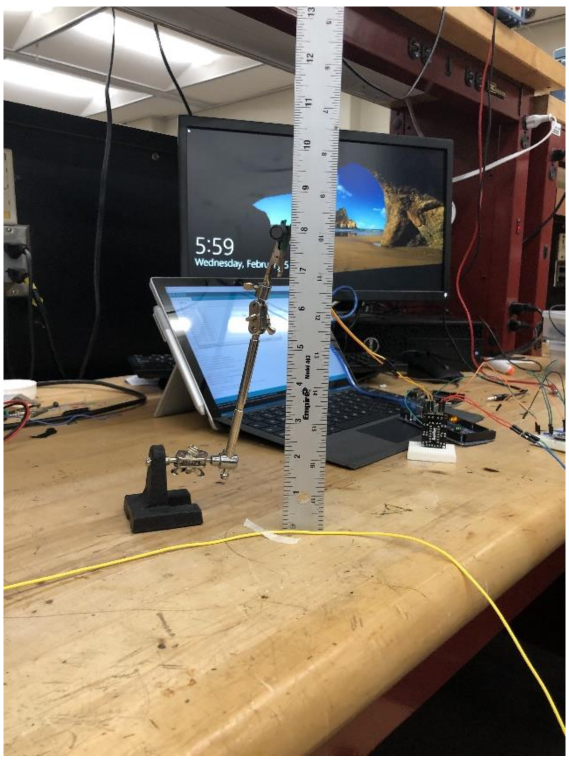


Figure 41: Measurement of Wire

6.3 Integration

This section describes the integration process of the final prototype of the rover. Due to the COVID-19 Pandemic that occurred during the production of this prototype and UCF closure starting March 9th, integration of the project was an extreme challenge for all members. All UCF students and many faculty where directed by the University to stay at home and work remotely, this order preceded the Orange County and state of Florida stay at home mandate, which as to this day in late April, is still undetermined when the ordinance will be lifted. In these circumstances, members made their best effort into integrating the final rover.

6.3.1 Tank Circuit Integration

The rover has been integrated with two inductors on the front left and front right of the rover. The circuitry is secured onto the body of the rover with tape and fast holding velcro to main its position while its in motion and allow easy troubleshooting when needed to be removed. The inductors are held in place on the bottom of the rover body through 3D printed parts. These parts where designed by our Mechanical Engineering team. The blue “antlers” as we informally call them allow the inductor to maintain a position low to the ground while not being vulnerable to the wheels on the rover or the elements from environment.

Figure 43: Tank Circuit Secured with Velcro

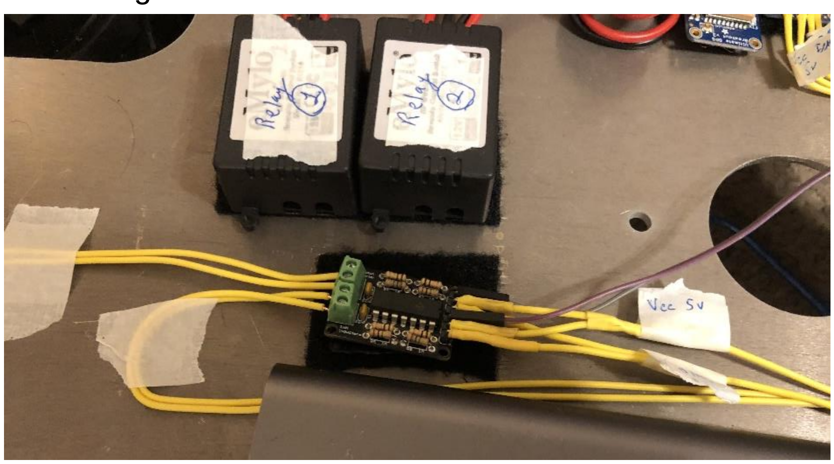
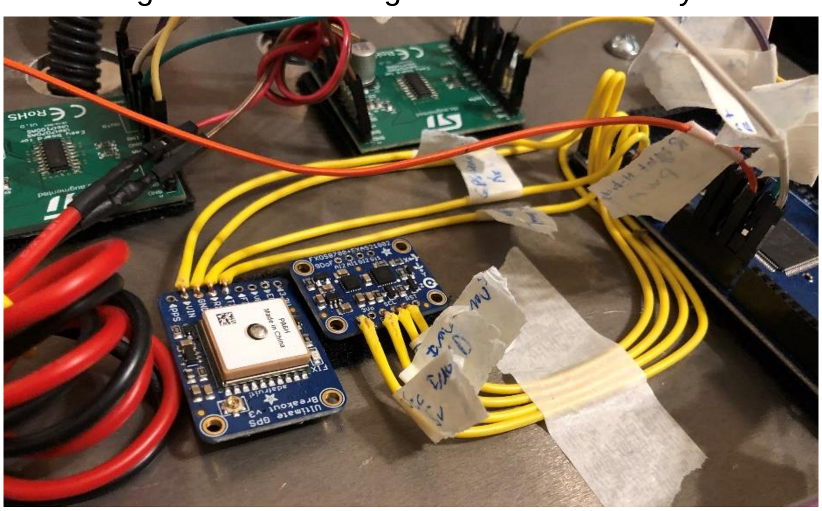


Figure 44: Inductors fastened to 3D printed holsters

6.3.2 IMU Integration

The IMU was integrated to the body of the rover close to the center of gravity in order to accurately determine the correct orientation and heading of the rover. The IMU is secured with velcro in order to keep it securely fastened to the body of the rover while allowing us to easily take it of to debug. The IMU is also connected to the MCU with labeled wire neatly organized on the rover.

Figure 45: IMU Integrated on Rover Body



6.3.3 GPS Integration

The GPS was integrated along side the IMU. The GPS is located right next to the IMU on the same velcro strip as the IMU in order to make good use of space. The GPS antenna is also connected to the GPS module and uses a magnet in order to maintain its position onto the aluminum body. While not a strong force, it is sufficient to keep from sliding around the body.

Figure 47: GPS Integrated on Rover

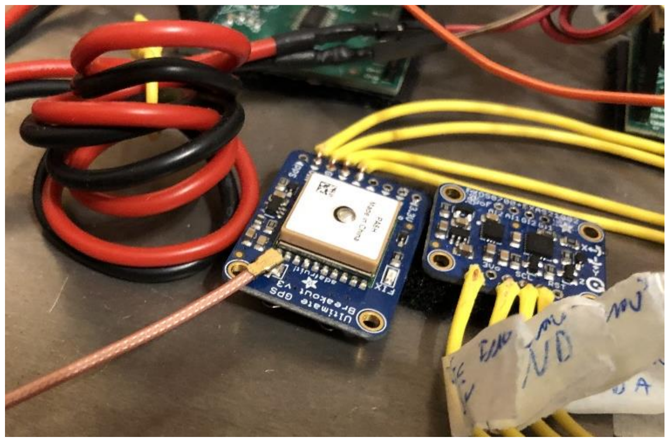
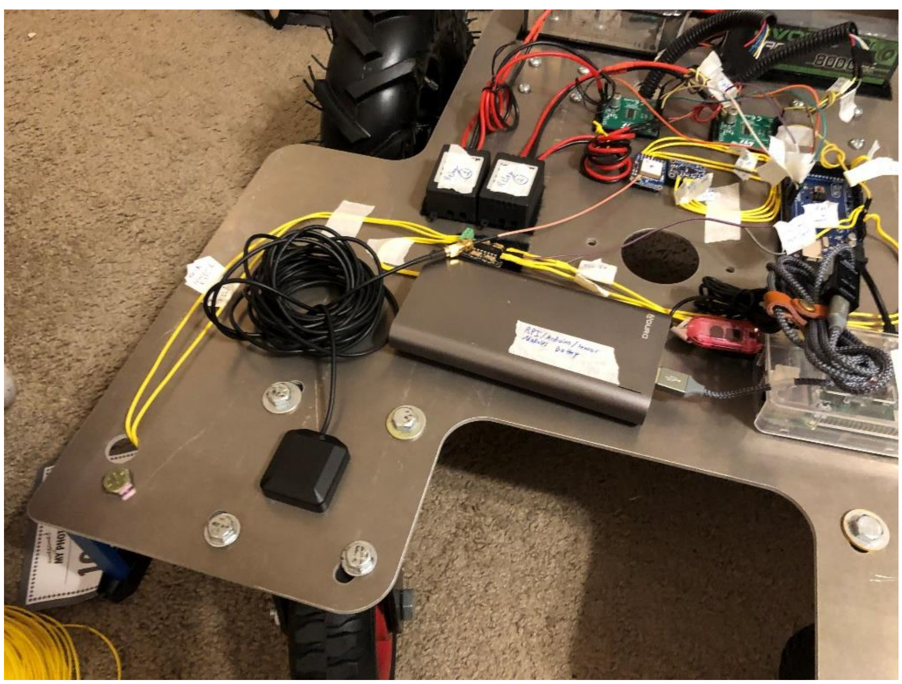


Figure 46: Integrated GPS Module and Antenna on Rover



7 Administrative

In this section, the managerial, budget, and delegation aspects of the project will be discussed and presented. To understand each team members required goals and marks to complete this project successfully, transparency and understanding of duties and tasks are necessary. In this section we will discuss how using the AGILE method of software development and tasking will help us remain on track. We will also talk about financial constraints in detail and overall budget of the project. We will also present the team members and discuss their overall duties and positions within the project.

7.1 Safety Precautions

Due to the handled equipment and nature of the project, it is important that all safety measures are taken place. The rover and the surrounding environment in which it is purposed to serve can be hazardous and cause injury if one is not properly equipped and understand how to handle the rover properly. In order to reduce risk during development, research, and testing of the project, the following guidelines should be followed:

1. When working on the rover, ensure a large clear area away from possibly valuable equipment that could be damaged.
2. When working on the rover, ensure that no food or drink items are near the rover that could possibly damage the equipment or electronics inside.
3. When working within the rover electronic compartment, make sure that all power subsystems are physically disconnected before manual work or inspection is done.
4. If power systems seemed to have failed, always check with a multimeter to review subsystems are turned off completely
5. When rover is operational, stand approximately 15 feet away from the operational rover.
6. When rover is connected to power subsystems, do NOT place hands under the rover or lift rover from front end.
7. Ensure that closed toed shoes, long pants, luminescent vest, and hard hat are worn on Duke Energy solar farm at all times during site visit.

7.2 Jira and AGILE

The AGILE method of development is being used for this project. The team has already purchased cloud hosted Atlassian services, including Confluence, Jira, and Bitbucket for hosting git repositories. Confluence is used for editing documents and storing internal team information, as well as Google Drive and Google Docs for main document editing. Jira is used for our sprint planning and backlog. The team has a backlog of action items to take. Sprints last 1 week each, and the first sprint tasks are as follows:

Figure 48: Jira and AGILE Task Distribution

Front-End Application Research	Autonomous Research	DR	EGAM-14	-
Bluetooth/Wireless Technology on Logic Board Research	Research	S	EGAM-15	-
Location and Positioning Research	Autonomous Research	EG	EGAM-12	-
LiDar and Computer Vision Research	Autonomous Research	JG	EGAM-13	-
Power Research	Research	JS	EGAM-21	-
PCB Design and Sourcing Research		JS	EGAM-27	-
Research Actuator and Motor Control		S	EGAM-26	-
ROS Research and Libraries		EG	EGAM-28	-
Document Writing			EGAM-29	-

7.3 Internal Documentation & Communication

Throughout the program, it is necessary to maintain information and documentation within our group in order to ensure that all information is easily accessible to everyone. There are several ways that our team communicates between one another as a main channel and different repositories for version control. Access to these resources is available upon request within the team. Here all the platforms we use in collaboration between all team members.

- SharePoint by Microsoft
- Google Drive by Google
- GroupMe by Microsoft
- Confluence by Atlassian Software
- Discord by Discord Inc.

7.4 Version Control

To keep and maintain progress practicing good Agile methodology, software source control software is needed. This allows individual developers within the team to associate tasks with current development branches, make pull requests, and peer review team members code before being merged into the final release branches. Throughout this project we will be using Bitbucket by the Atlassian

Software sweet in order to keep track of pull requests, review, comment, and manage development and release branches of our code.

7.5 Budget

For our budget, we were allotted $1500. Between the ME, EE, CPE, and CS. Most of the parts for our project have been purchased and / or awaiting arrival or purchase status through the department of Mechanical Engineering. We are also using some of the components from the original build to help cut costs as well as using comparison charts that use cost as one of the points of evaluation. The team is allotted $1500 since this is a sponsored project by Duke Energy and Orange County Utilities. The sponsor sponsors this project every year to help student do their best to fill an actual need.

Table 31: ME & ECE/CS Conjoined Budget

Part	Price	Quantity	Total
Caster Wheel	$15.86	1	$15.86
Drive Wheels	$107.52	1	$107.52
Driving Motors	$84.33	2	$168.65
Trimming Motors	$13.58	2	$27.16
Trimmer Head	$17.69	1	$17.69
18V Ryobi 1.3Ah	$69.97	1	$69.97
18V Ryobi 4.0Ah	$119.00	1	$119.00
32 Qt latching box	$7.98	1	$7.98
Boundary Wire	$18.95	1	$18.95
GPS Module	$39.95	1	$39.95
SMA Adapter Cable	$3.95	1	$3.95
GPS Antenna	$14.95	1	$14.95
IMU Sensor	$14.95	1	$14.95
Lidar	$300	1	$300
Perimeter Wire Kit	$9.95	1	$9.95
Odroid XU4	$77.85	1	$77.85
Bearing Pillow	$15.95	1	$15.95
USB-Hub Adapter	$6.99	1	$6.99

Table 32: ME & ECE/CS Conjoined Budget Continued

Part	Price	Quantity	Total
Intel RealSense Camera	$175.99	1	$175.99
Wireless Relay	$28.69	1	$28.69
Ovonic 11.1V 8000mAh 3S 50C LiPo Battery	$87.99	1	$87.99
B3 RC LiPo 2S-3S Battery Balancer Charger 7.4-11.1V	$10.99	2	$21.98
5pcs Male T-Plug	$9.98	2	$19.96
Lipo Bag	$7.77	1	$7.77
Velcro Strips	$8.99	1	$8.99
Buck Converters (5pcs)	$26.59	1	$26.59
Wire loom	$11.99	1	$11.99
Wire	$15.99	1	$15.99
Connectors	$11.99	1	$11.99
Connectors (robust)	$10.99	1	$10.99
Wire (robust)	$16.89	1	$16.89

Table 33: ECE/CS/MS BOM

Department	Cost Estimate
Mechanical Engineering	$551.70
Electrical & Computer Engineering	$653.30
Total	$1205.00

7.6 Purchasing

Since this project is being funded through our sponsors Duke Energy and the Orlando Utility Commission, funding will be provided through the Department of Mechanical Engineering who hold the funds from the sponsor. Each team will be allocated $1500 for spending power that can be used to purchase through online retailers with certain restrictions. The team members through their designated tasks and needs will be free to choose parts to purchase that will contribute towards the project. However, each individual team member goes through one central figure and documents the purchase and price in order to keep records on budget constraints and time management for critical system design implementation.

Our Mechanical Team member, Nick Delligatti, has volunteered to handle and create all the purchase requests for the group. Purchase requests work by using the Department of Mechanical Engineering's purchase order request through their website. The part name, part numbers, quantity, price, vendor, and link to the part will be sent to through the website. Once that is completed it is up to UCF purchasing department.

7.7 Trouble Shooting

While creating the E-Goat rover, we faced many challenges and issues when setting up software and hardware components for this rover. Here are the most helpful troubleshooting guides that we can support each other with.

7.7.1 Sensor Modules and Arduino Not Powered

1. Make sure all devices are connected to power through either a 5v port.
2. Enable Relay switches to ensure that power is being supplied to components
3. Make sure all batteries are fully charged and Power Bank is fully charged.

7.7.2 Sensor Modules not Receiving Communication

1. Ensure that all connections are properly made on the Arduiono board and to the Single Board Computer (RaspberryPi/ODROID)
2. Try to reset the PCB or Arduino by using the onboard reset button.
3. Log into the Single Board Computer (RaspberryPi/ODROID)
4. Open a command terminal prompt
5. Type "/rostopic list" and make sure the following topics are listed
    a. /cmd_vel
    b. /tank_circuit
    c. /position
    d. /orientation
6. Type "systemctl list-units" and check if the egoat-service.service is running in the list of services
7. Type "sudo systemctl stop egoat-service.service"
8. Type "sudo systemctl start egoat-service.service"
9. Re-Do step 5 to verify topics are active.

7.7.3 Uploading Code into Arduino Mega

*Note that this code as is specifically designed for the Arduino Mega and will not work on other dev boards.

Pre-Requisites: Linux Distro with ROS Kinetic OR Melodic installed. Arduino IDE installed. ROS-Serial library installed with the Ros-Serial Arduino Library. A ROS workspace environment.

1. Clone the arduino_code repository into your packages fold in the workspace (_nameOfWorkSpace_/src)
2. In the root of your workspace open a terminal prompt
3. Type “source develop/setup.bash”
4. Type “catkin_make”
5. Type “catkin_make install”
6. Type “catkin_make arduino_code_firmware_firmware”
7. Make sure that the Arduino Mega is connected to you computer via USB and the communication port is connected to /dev/ttyACM0
8. Type “catkin_make arduino_code_firmware_firmware-upload”

7.8 Project Milestones

The Fall and Spring 2019-2020 Senior Design 1 milestones are to keep the large scope of the project on task and on track for this semester. In order to keep track of both the Mechanical, Electrical, Computer Engineering and Computer Science members of this team, detailed spreadsheets and charts have been created to keep track of mission critical points in time. The Mechanical Engineering group have distinct project and documentation deadlines that are not shared with the Electrical, Computer Engineering and Computer Science members. Through special permissions with the Computer Science section of Senior Design and the nature of Interdisciplinary teams, the computer science member shares the same deadlines as Electrical and Computer Engineering members in this team. Table 32 shows the detailed dates of track points.

Table 34: Milestone Dates

Task	Due Date	People
Form Group	8/30/2019	Group 26 ECE
Project Idea	9/11/2019	Group 26 ECE
1st Meeting with Dr. Steiner	9/11/2019	ECE
Initial Report	9/20/2019	ECE
Initial Documenation Meeting	9/24/2019	ECE
Miletone 1 -Idea Finalized		
Project Status Presentation	10/3/2019	Everyone
Product Requirements Explored	10/21/2019	ECE and CS
Technology Memo	9/26/2019	Mechanical
Team Contract	10/26/2019	Mechanical
Miletone 2 -Full requirements and specifications defined		
Prelimiary Midterm Presentation	10/14/2019	Everyone
Purchase Test Equipment and Components	10/22/2019	ECE and CS
Midterm Presentation	10/28/2019	Everyone
Concept Design Report	10/29/2019	Mechanical
60-page Draft Design Documenation	11/1/2019	ECE and CS
Test equipment and components	11/3/2019	Everyone
Competition Site Visit	11/4/2019	Davis & Natalie
Order Parts	11/12/2019	Nick
100-page Draft Desgin Documentation	11/15/2019	ECE and CS
Final Presentation	11/25/2019	Everyone
Final Documentation SD 1	12/4/2019	ECE and CS
Milestone 3 - Research and Final Document completed		
Build Prototype	2/1/2020	ECE and CS
Testing and Redesigning	2/15/2020	ECE and CS
Finalizing Prototype	3/1/2020	ECE and CS
Peer Presentations	TBA	ECE and CS
Internal Competition	TBA	Everyone
Final Report	TBA	ECE and CS
Final Presentation	TBA	Everyone

7.8.1 Mechanical, ECE, and CS Gantt Charts

To visualize our goals and time used throughout the project, all members of the team have contributed into creating a gantt chart. Gantt charts are a productive way to visualizing projected schedule and achieved milestones.

Figure 49: Gantt Chart 1

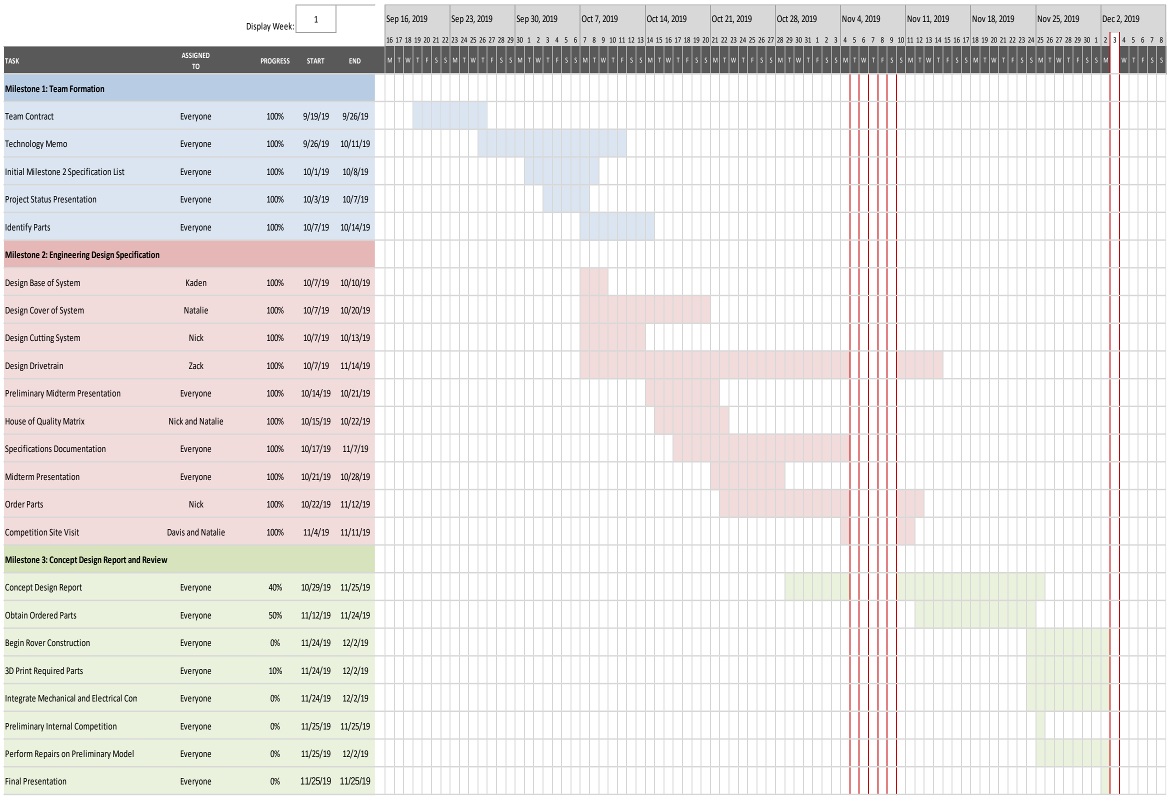


Figure 50: Gantt Chart 2

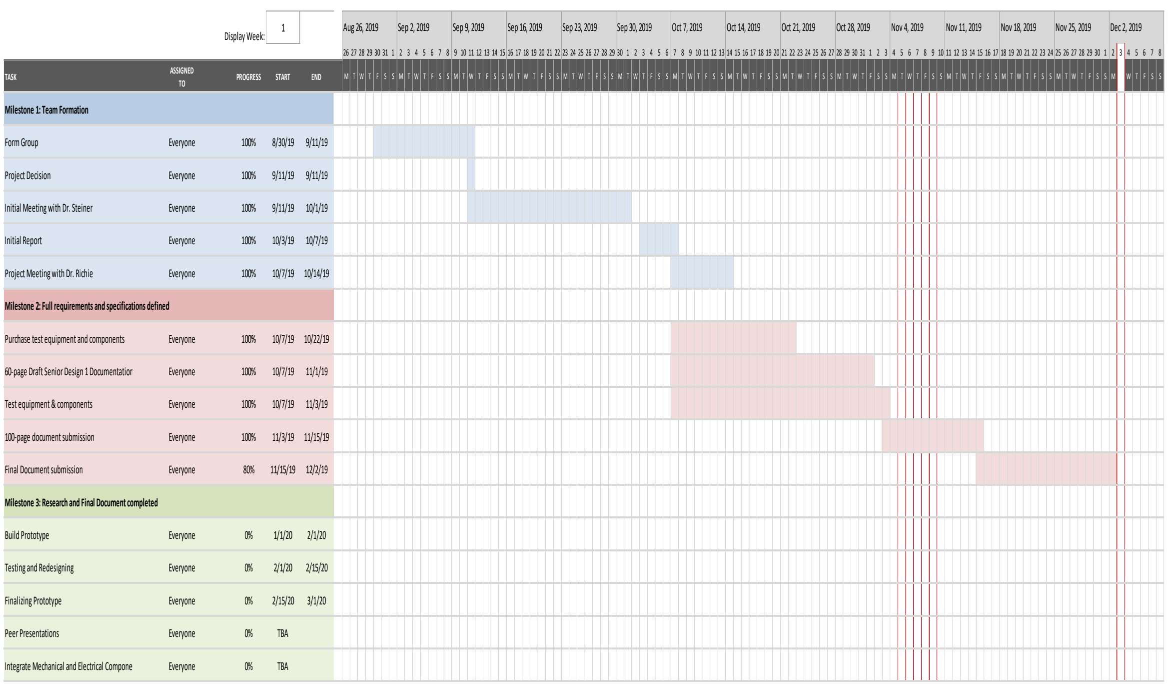
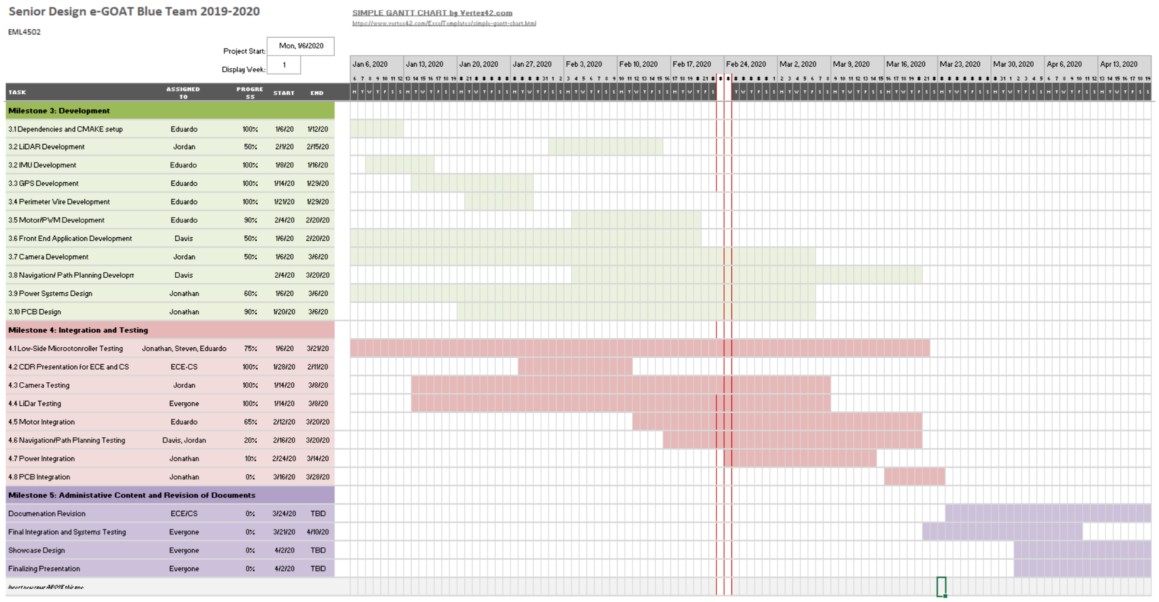


Figure 51: Gantt Chart 3



7.9 Technical Roles

In this subsection, the technical roles and strengths of our team members will be briefly summarized and documented in order to maintain a clear understanding of high level function within the project.

7.9.1 Steven Cheney - Computer Engineer:

Steven Cheney is one of the three Computer Engineers on this EGOAT team. He is a jack of all trades, having experience ranging from hardware assembly and design, to having programming experience. Although less experienced in the form of hardware design, he is actively researching how to best aid the team and lend a hand in whatever area of the project that is needed, whether it be technical or physical. Steven has dabbled with both sides of the degree extensively, and currently prefers to work with the hardware. Not only will he be aiding with wireless communications on the autonomous lawn mower, but he will also be helping with the assembly process of all components used in this project, primarily aiding Jordan Germinal and Davis Rollman with any programming assistance needed.

7.9.2 Jordan Germinal - Computer Engineer:

Jordan Germinal is one of the three Computer Engineers working on this project. He is primarily working on the autonomous lawn mower's robot vision with the Computer Science major on this team, Davis Rollman. While the core of my classes at UCF have been structured around a split of hardware and software, the majority of my electives have been in software programming, allowing me a greater understanding of what the programming aspect of this project will require for the robot vision. I have even taken a robotics course at UCF.

7.9.3 Eduardo Guevara - Computer Engineer:

Eduardo Guevara is one of the three Computer Engineers within the project that will be helping with several subsystems towards the final solution. Eduardo will be focused in working on the Navigation and Localization subsystems closely related to several parts of the hardware stack. Within these stacks, work will compromise mostly of the Navigation System in designing, implementing, and testing the Navigation system with the microcontroller. As previously stated, this system includes the Boundary Wire System, the GPS Localization system, the IMU sensor. These modules and systems will communicate closely with the microcontroller selected. Eduardo Guevara will also offer consulting and general help with source control, ROS, and integration between the mechanical team.

7.9.4 Davis Rollman - Computer Science:

I am the only computer science major on the team, but not the only coder. I will be working with Jordan Germinal on Computer Vision aspects of the project. We will be using the LIDAR sensor to gather data about the world around the robot. We need to come up with good solutions for navigation. I have some hardware and electrical background from an internship, but mostly just enough to understand what the others on the team are doing at a high level.

7.9.5 Jonathan Smith - Electrical Engineer:

Since Jonathan Smith is the only electrical engineer on the team, his main focus is going to be on power and PCBs. This project will need quite a few PCBs including the microcontroller, motor drivers, voltage regulators, remote relays, and emf sensor. Now, not all of these need to be made by him, some can be purchased commercially. However, some of them will need to be made by him, which shouldn't be too difficult because he has experience in PCB design. When it comes to the batteries and power consumption, he doesn't have too much experience in it, but this project should help with that. Overall this project will be a great learning experience and we're glad we chose Egoat as our project.

8 Bibliography

8.1 Copyright Permissions

The copyright permissions requests and approvals are shown below. This is to ensure used information, data and images get the proper credit and follow copyright laws in this project documentation.

Figure 52: The permission request to Pishop is shown to use their information and images of their products on their website.

PS

support.77496.d25484bb4d4f5540@helpscout.net on behalf of
PiShop.us - Support <support@pishop.us>
Tue 12/3/2019 10:26 AM
Jordan Germinal

Hello Jordan,

Thank you for contacting us! We really appreciate you taking the time to ask us.
Feel free to use any descriptions and pictures from our website.

How would you rate my reply?
Great Okay Not Good

Best regards,

Anneline Kraemer
PiShop.us Support Team
support@pishop.us

Figure 53: The permission request to Stereolabs is shown to use their information and images of their products on their website.

Copyright Permission

Guillaume Cortes (Stereolabs)
Dec 3, 00:54 PST

Hi Jordan,

Thank you for reaching out.

Yes you are free to use the content available on our website.
We'll be glad to look at the documentation one it is done. Would you mind giving us access to it?

Best.

Best,
Guillaume
Stereolabs Support

Jordan Germinal
Dec 2, 14:53 PST

To whom it may concern,

I'm currently writing a documentation for my Senior Design 1 Project that includes the use of your products.

May I have permission to use the information and images of your products from your website?

Thank you

Sincerely

Jordan Germinal

Figure 54: The permission request to Miuzeipro is shown to use their information and images of their products on their website.

Copyright Permission

S

support@miuzeipro.com
Mon 12/2/2019 8:33 PM
Jordan Germinal

Yes, it doesn't matter

support@miuzeipro.com

...

Ok, thanks. | Thanks! | Ok, sent.

Are the suggestions above helpful? Yes No

JG

Jordan Germinal
Mon 12/2/2019 7:02 PM
support@miuzeipro.com

To whom it may concern,

I'm currently writing a documentation for my Senior Design 1 Project that includes the use of your products.

May I have permission to use the information and images of your products from your website?

Thank you

Sincerely

Jordan Germinal

Jean010@knights.ucf.edu

University of Central Florida

Figure 55: the permission request to Amazon is shown to use their information and images of their products on their website.

Copyright Permission

JG

Jordan Germinal
Mon 12/2/2019 7:04 PM
cis@amazon.com

To whom it may concern,

I'm currently writing a documentation for my Senior Design 1 Project that includes the use of your products.

May I have permission to use the information and images of your products from your website?

Thank you

Sincerely

Jordan Germinal

Jean010@knights.ucf.edu

University of Central Florida

8.2 References

[1] AIR 802, "IEEE 802.11 a/b/g/n Wi-Fi Standards and Facts," [Online]. Available: https://www.air802.com/files/802-11-WiFi-Wireless-Standards-and-Facts.pdf. [Accessed 14 Oct 2019]

[2] Anh Vu Le, Veerajagadheswar Prabakaran, Vinu Sivanantham and Rajesh Elara Mohan, "Modified A-Star Algorithm for Efficient CoveragePath Planning in Tetris Inspired Self-ReconfigurableRobot with Integrated Laser Sensor [Online]. "https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6111563/pdf/sensors-18-02585.pdf." [Accessed: 3-Oct-2019]

[3] Agarwal, Tarun. "Wireless Radio Frequency Technology working and Applications." https://www.edgefxkits.com/blog/rf-wireless-radio-frequency-working-application/ [Accessed 16 Oct 2019]

[4] "ATmega328P Datasheet," 2015. [Online]. Available: http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-7810-Automotive-Microcontrollers-ATmega328P_Datasheet.pdf. [Accessed: 27-Nov-2019].

[5] "ATMEGA328P-PU," *DigiKey*. [Online]. Available: https://www.digikey.com/product-detail/en/microchip-technology/ATMEGA328P-PU/ATMEGA328P-PU-ND/1914589&?gclid=Cj0KCQiA2vjuBRCqARIsAJL5a-IkzHE5nuyJn4uLVwQ2Qq-9FwTQ6D_YqY72cAW7F6AP3GV43FC_3f8aAssPEALw_wcB. [Accessed: 27-Nov-2019].

[6] Cdn-learn.adafruit.com. (2019). 3-Axis Digital Angular Rate Gyroscope. [online] Available at: https://cdn-learn.adafruit.com/assets/assets/000/040/671/original/FXAS21002.pdf?1491475056 [Accessed 27 Nov. 2019].

[7] Cdn-learn.adafruit.com. (2019). FXOS8700CQ 6-axis sensor with integrated linear accelerometer and magnetometer. [online] Available at: https://cdn-learn.adafruit.com/assets/assets/000/043/458/original/FXOS8700CQ.pdf?1499125614 [Accessed 27 Nov. 2019].

[8] Circuit Basics, "Basics of UART Communication," Stanford, [Online]. Available: https://web.stanford.edu/class/cs140e/notes/lec4/uart-basics.pdf. [Accessed 22 Oct 2019]

[9] Dell'Acqua, Alexa A., and John F. Mazzaferro. "Wireless communications." *Telecommunications*, Mar. 1996, p. S1+. *Gale OneFile: Computer Science*, https://link.gale.com/apps/doc/A18190291/CDB?u=miam50083&sid=CDB&xid=084e79f7. [Accessed 27 Oct. 2019]

[10] D
iffen.com, "Bluetooth vs. Wifi," Diffen, [Online]. Available:

https://www.diffen.com/difference/Bluetooth_vs_Wifi. [Accessed 31 Oct 2019]

[11] “

Foundation Strategy 2016-2019” (PDF). *Raspberry Pi*. Raspberry Pi Foundation. pp. 3–5. [Accessed 2 November 2019]

[12]

GCI Staff. “OPEI announces first ANSI standard for robotic lawn mowers.” September 20, 2019. https://www.golfcourseindustry.com/article/opei-robot-mower-standard/. [Accessed 1 Nov 2019].

[13]

“GlobalTop Technology Inc. FGPMMOPA6H GPS Standalone Module ...” [Online]. Available: https://cdn-shop.adafruit.com/datasheets/GlobalTop-FGPMMOPA6H-Datasheet-V0A.pdf. [Accessed: 27-Nov-2019].

[14]

“Hands-on with the New Raspberry Pi camera module” https://www.zdnet.com/article/hands-on-with-the-new-raspberry-pi-camera-module/ [Accessed November 13, 2019]

[15]

I2C Info, "I2C Info - I2C Bus, Interface and Protocol," 30 Nov 2018. [Online]. Available: http://i2c.info/

[16] “

IEC,” *IEC 62680-2-1:2015* | *IEC Webstore* | *energy, multimedia, USB, LVDC*. [Online]. Available: https://webstore.iec.ch/publication/23313. [Accessed: 27-Nov-2019].

[17]

K. Townsend, “NXP Precision 9DoF Breakout,” *Adafruit Learning System*. [Online]. Available: https://learn.adafruit.com/nxp-precision-9dof-breakout/downloads. [Accessed: 27-Nov-2019].

[18]

L. Ada, “Adafruit Ultimate GPS,” *Adafruit Learning System*. [Online]. Available: https://learn.adafruit.com/adafruit-ultimate-gps/downloads. [Accessed: 27-Nov-2019].

[19]

M. Grusin, "Serial Peripheral Interface (SPI)," SparkFun Electronics, 2018. [Online]. Available: https://learn.sparkfun.com/tutorials/serial-peripheralinterface-spi/all. [Accessed 3 Oct 2019]

[20]

NMEA Revealed. [Online]. Available: https://gpsd.gitlab.io/gpsd/NMEA.html#_nmea_0183_physical_protocol_layer. [Accessed: 27-Nov-2019].

[21]

“Jetson TX2 Module.” https://developer.nvidia.com/embedded/jetson-tx2. [Accessed October 16, 2019]

[22]

“ODROID-XU4 OCTA-CORE COMPUTER WITH SAMSUNG EXYNOS

5422” https://www.antratek.com/odroid-xu4-octa-core-computer-with-samsung-exynos-5422 [Accessed October 30, 2019]

[23]
“ODROID-XU4 Special.” https://www.hardkernel.com/shop/odroid-xu4-special-price/ [Accessed November 11, 2019]

[24]
“Raspberry Pi 3 Model B” https://www.raspberrypi.org/products/raspberry-pi-3-model-b/ [Accessed November 9, 2019]

[25]
“Raspberry Pi Camera Board – Night Vision and Adjustable-Focus Lens (5MP)” https://thepihut.com/products/raspberry-pi-night-vision-camera [Accessed October 28, 2019]

[26]
"Single Board Computer Peripherals". Newmicros. [Accessed October 16, 2019]

[27]
“Spatial Mapping - Overview.” *Stereolabs*, https://www.stereolabs.com/docs/spatial-mapping/

[28]
Stryuk, Tara. “What is the difference between Bluetooth and Wi-Fi?” Apr. 2017. https://www.techopedia.com/2/27881/networks/wireless/what-is-the-difference-between-bluetooth-and-wi-fi. [Accessed 31 Oct. 2019]

[29]
Topline, “Surface Mount Nomenclature and Packaging.” [Online]. Available: https://www.topline.tv/SMT_Nomenclature.pdf. [Accessed: 11-27AD].

[30]
TI. (2019). xx555 Precision Timers. [online] Available at: http://www.ti.com/lit/ds/symlink/se555.pdf [Accessed 27 Nov. 2019].

[31]
Intel,“Specifications for the Intel® RealSense™ Development Kit Featuring the ZR300” [online] Available at: https://www.intel.com/content/www/us/en/support/articles/000023521/emerging-technologies/intel-realsense-technology.html [Accessed 27 Nov. 2019].