Battlefield Effects Simulator Robot
Group 6 Senior Design CDR

16 September 2022

Jared Rymkos
Computer engineering

Julia Kemper
Computer engineering

Michael Rodriguez
Electrical engineering

Nicholas Nachowicz
Computer engineering

UNIVERSITY OF CENTRAL FLORIDA

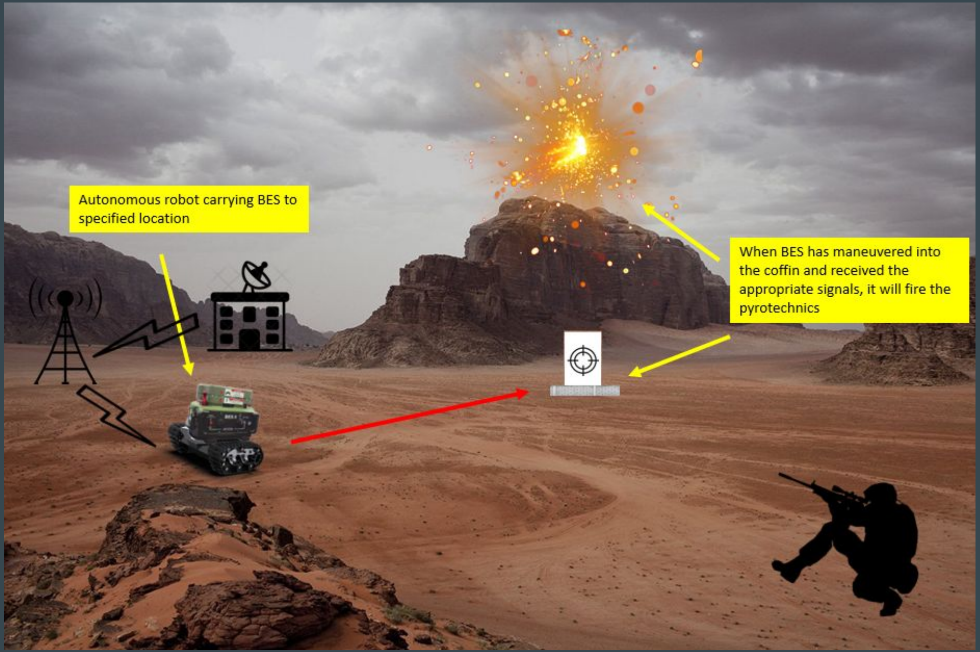
PEO STRI

Sponsors/Mentors:
James Todd
Jeremy Lanman
David Howard
Tom Waligora

BES ROBOT

Team:
Jared Rymkos
Julia Kemper
Michael Rodriguez
Nicholas Nachowicz

Sponsors/Mentors:
James Todd
Jeremy Lanman
David Howard
Tom Waligora



Project Overview

The Battlefield Effects Simulator robot is a small scale version of a robotic system that will be utilized on live fire ranges to support training realism and feedback, threat representation, and pyrotechnical handling.

This version of the robot will maneuver through rough terrain to and from desired locations to accurately fire pyrotechnics when proper safety signals are sent. The BES for this prototype will not be on top of the robot, but will still be in communication with it. The robot will interact with the BES to send the safety conditions for when it is clear to fire the pyrotechnics.

The safety conditions include:

1. The robot has arrived at the correct location on the map
2. There are no humans within a 2-meter vicinity
3. The robot is on a flat surface plus or minus ten degrees

Motivation

The motivation for this project comes from the governments derived need for a Modular Open System robotic platform that aligns with the Robot Operating System Military (ROS-M) framework and ecosystem.

The robot is intended to support increasing training realism and feedback and improve safety measures during training by removing the human element when it comes to pyrotechnics handling and replacement.

Goals and Objectives

Main goal: For this robotic system to be utilized on live fire ranges to support training realism and feedback, threat representation, and pyrotechnical handling.

Our objectives consist of the following:

- Have the robot navigate autonomously through rough terrain to its specified location
- Have the robot communicate with the control center to receive commands to fire
- Have the robot communicate with the BES to send the commands to fire the pyrotechnics

Constraints

- **Economic Constraints**
  - $1,000 Budget
- **Environmental Constraints**
  - Robot must be able to accurately navigate over rough terrain, hills, and holes
- **Social Constraints**
  - N/A
- **Political Constraints**
  - N/A
- **Ethical Constraints**
  - N/A
- **Health and Safety Constraints**
  - Senior Design team will not test with pyrotechnics due to the safety of the team
- **Manufacturability Constraints**
  - Arrival of parts on time
  - Arrival of parts in good condition
- **Sustainability Constraints**
  - N/A
- **Time Constraints**
  - Senior Design timeline
  - Sponsor timeline

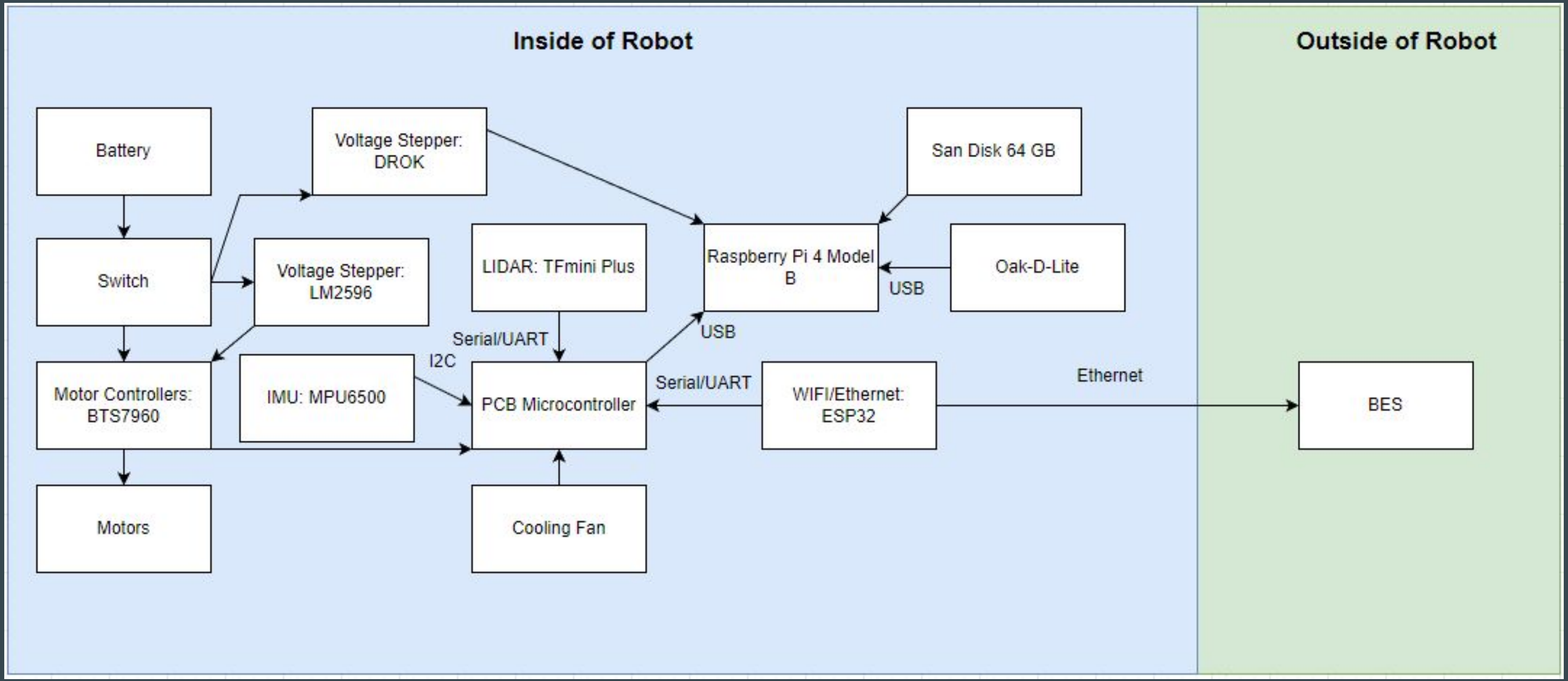
Standards

- IEEE/ISO/IEC 29148-2018 - Software Requirements Specification
  - Followed to create unambiguous, verifiable, and complete requirements
  - Avoid terms like “if possible” in requirements in order to make them clear
- ISO/IEC/IEEE 29919 - Software Testing
  - Followed to have a clear procedure when troubleshooting the code
  - Different test methods including Specification, Structure, and Experience based techniques
- PEP8 - Python Standard
  - Followed in order to have clear and cohesive code that mentors and team can follow
  - Allow smooth transition to future teams that build off our project
- IEEE 1725 - Rechargeable Battery Standard
  - Followed in order to maintain a safety requirement involving the battery
  - Lowers risks of accidents occuring with the battery and allows for all specs of the batteries to be available for the user.

Specifications

Component	Parameter	Design Specification
Battery	Discharge Time	1 hour
Battery	Charge Time	1 hour and 15 minutes
Speed	Accuracy	2 mph
Sensor obstacle detection	Accuracy	0.1 meters - 12 meters
Camera human detection	Accuracy	2 meters

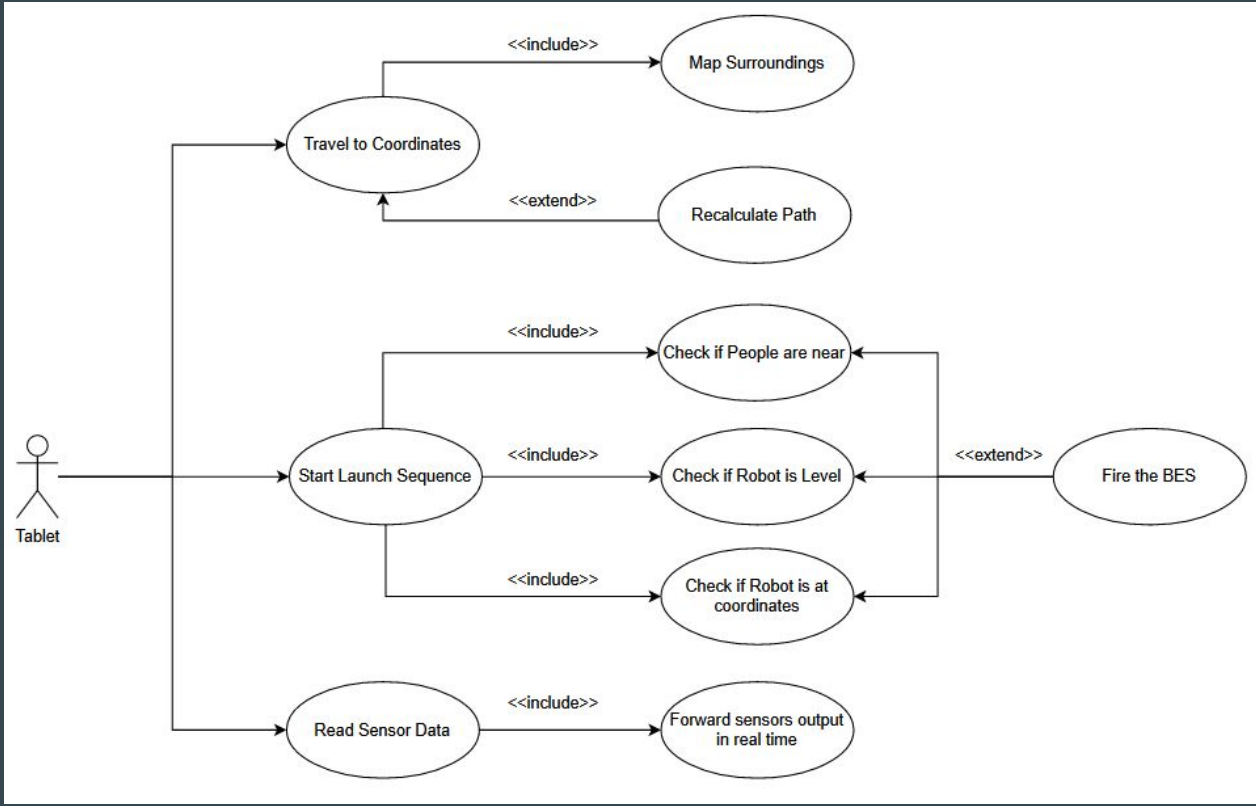
Overall system diagram



User interaction

- There will be a rocker switch on the robot that toggles the power
- The user will need to replace the battery with a charged one and plug it in
- The user will need to charge the batteries outside of the robot
- The user will need to interact with the GUI on the windows tablet

Use Case Diagram

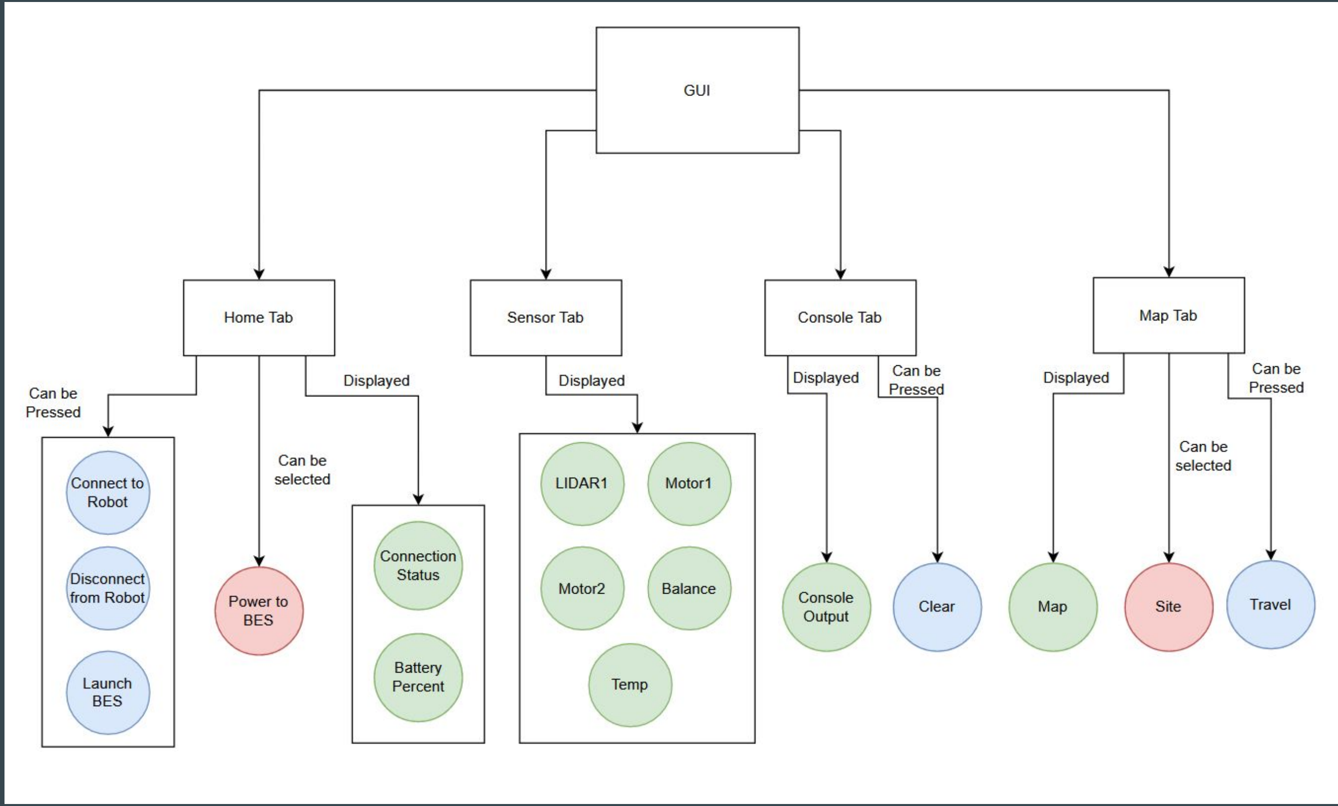


GUI

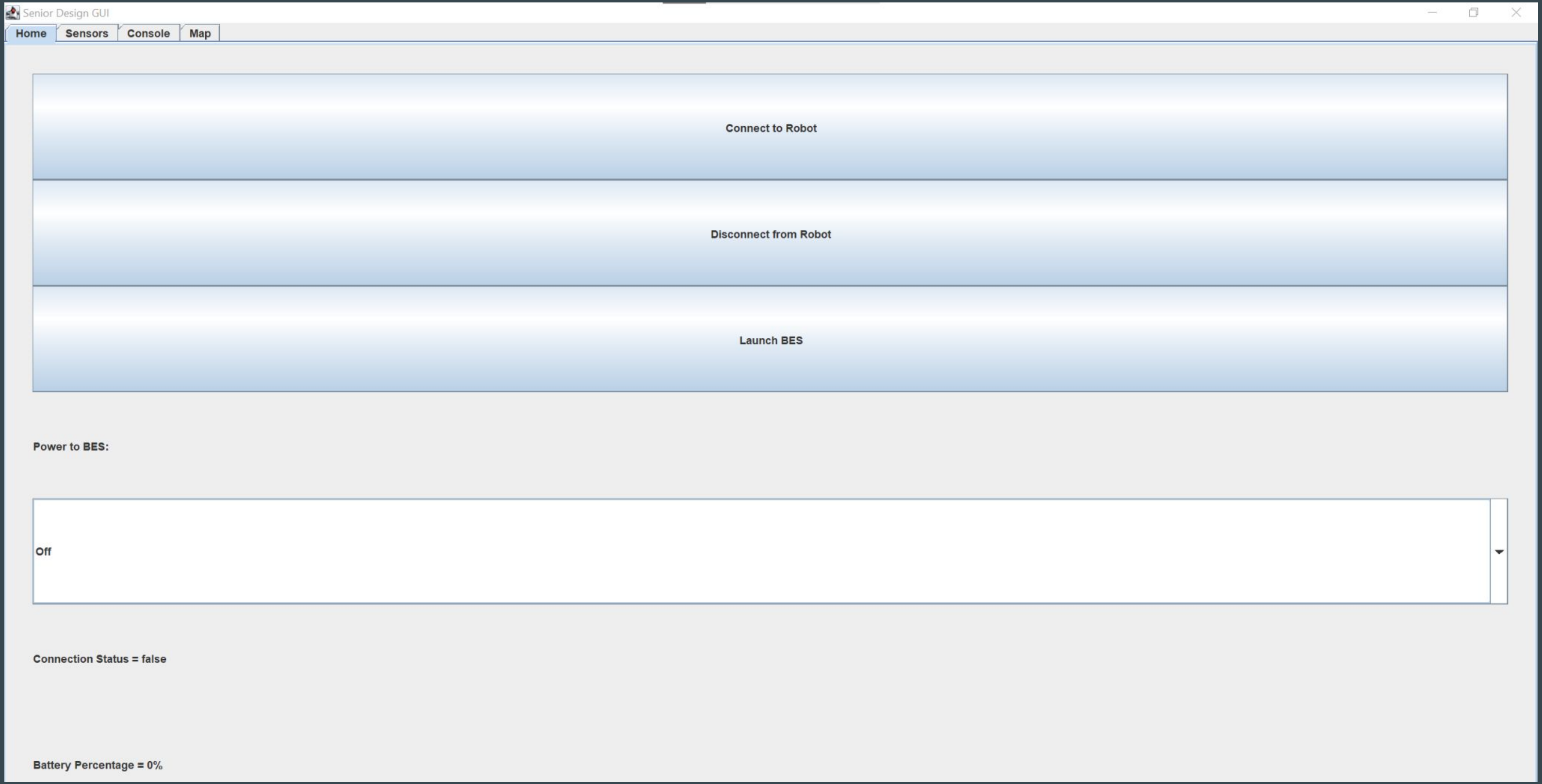
- Created in Java using the Swing library
- Basic functionality prototype is complete ready to start the bluetooth integration
- Size and formatting of the buttons and text will be completed later
- Four main tabs including home, sensor, console, and map
- Will have a 2D map that illustrates coordinates of the Robot and sites (in progress)
- Link to the code: https://github.com/jrymkos/SeniorGUI

1. **GUI Requirements**
   1.1. The robot will be turned on by a power button.
   1.2. An application will contain the UI used to control the robot.
      1.2.1. The application will be able to control the training zone.
      1.2.2. The application will have different destinations that a user can select for the robot to maneuver to.
      1.2.3. The application will tell the user the current battery life of the machine.

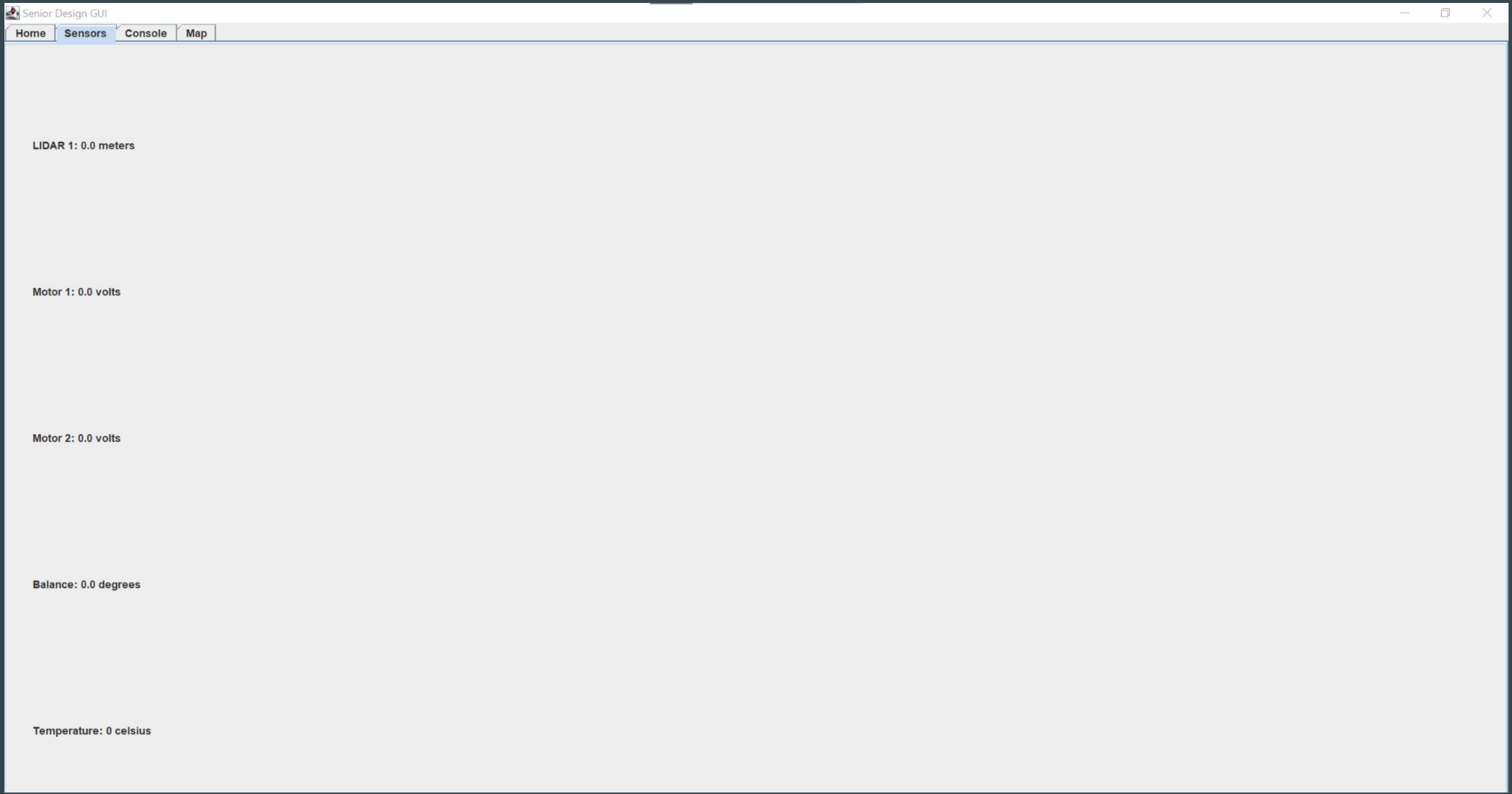
GUI - Component Diagram



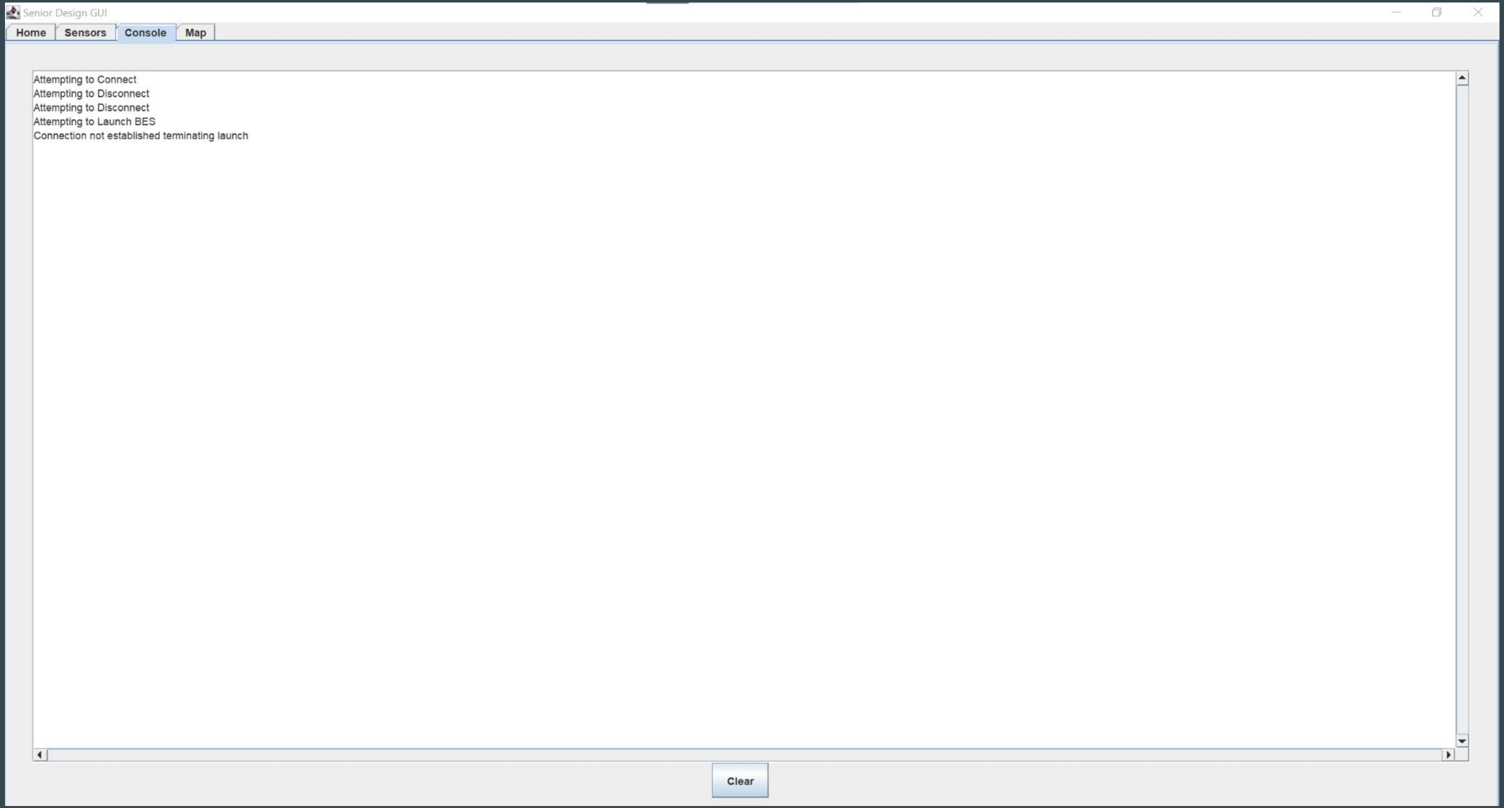
GUI - Home Tab


Senior Design GUI
Home
Sensors
Console
Map
Connect to Robot
Disconnect from Robot
Launch BES
Power to BES:
Off
Connection Status = false
Battery Percentage = 0%

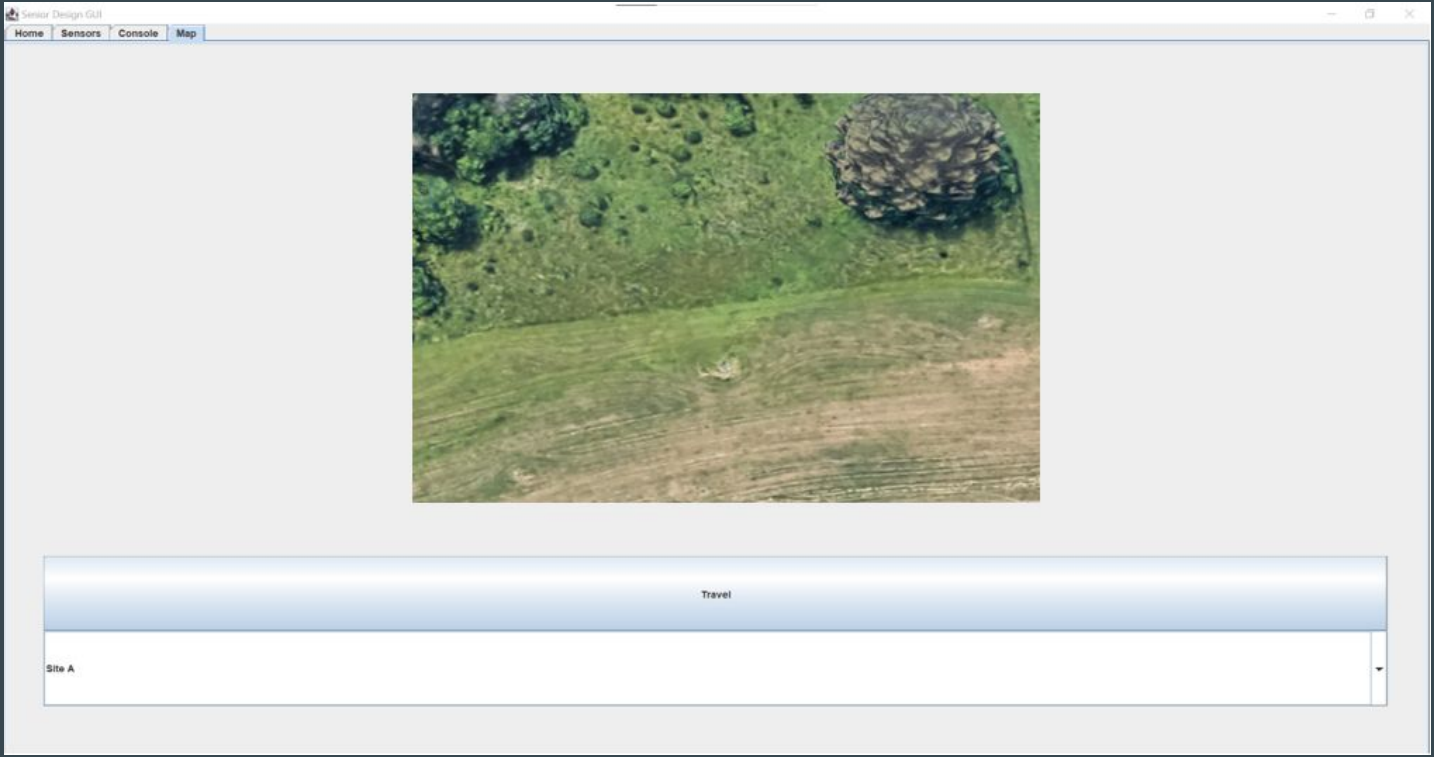
GUI - Sensor Tab



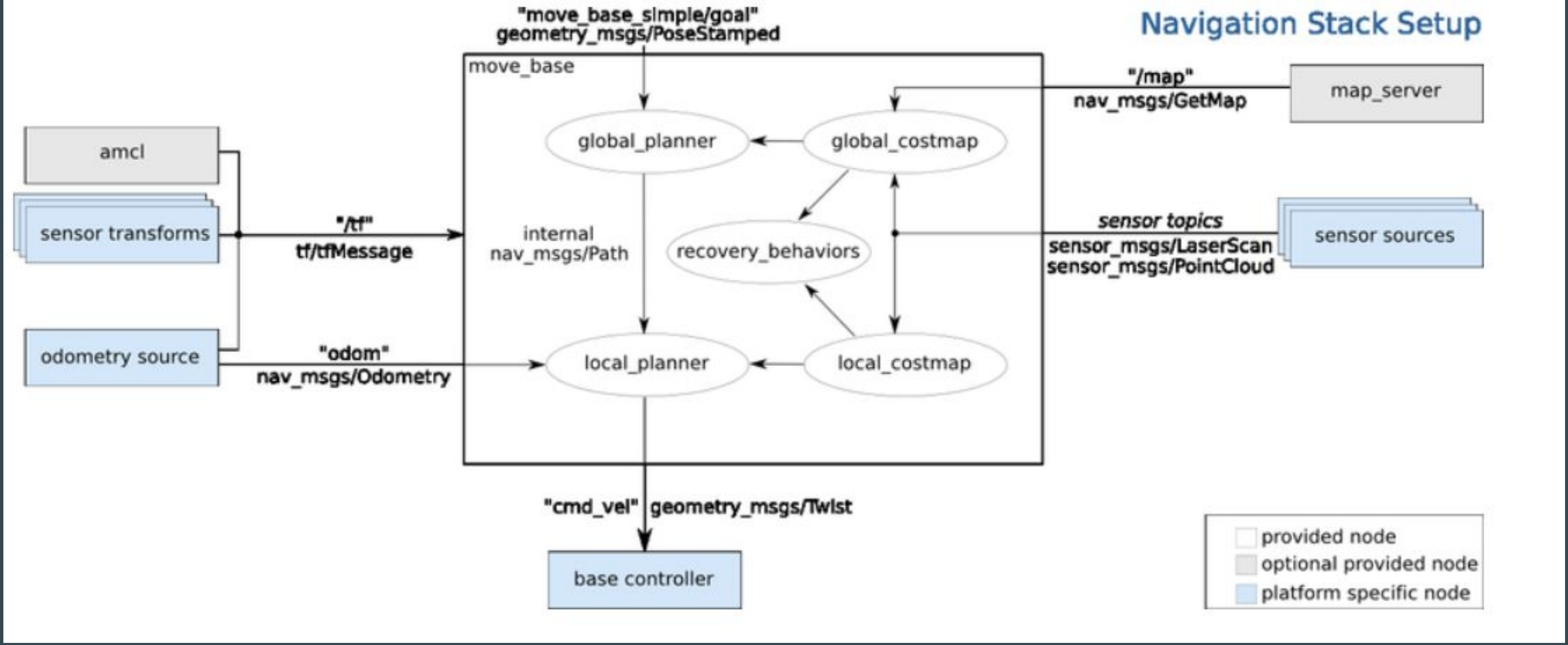
GUI - Console Tab



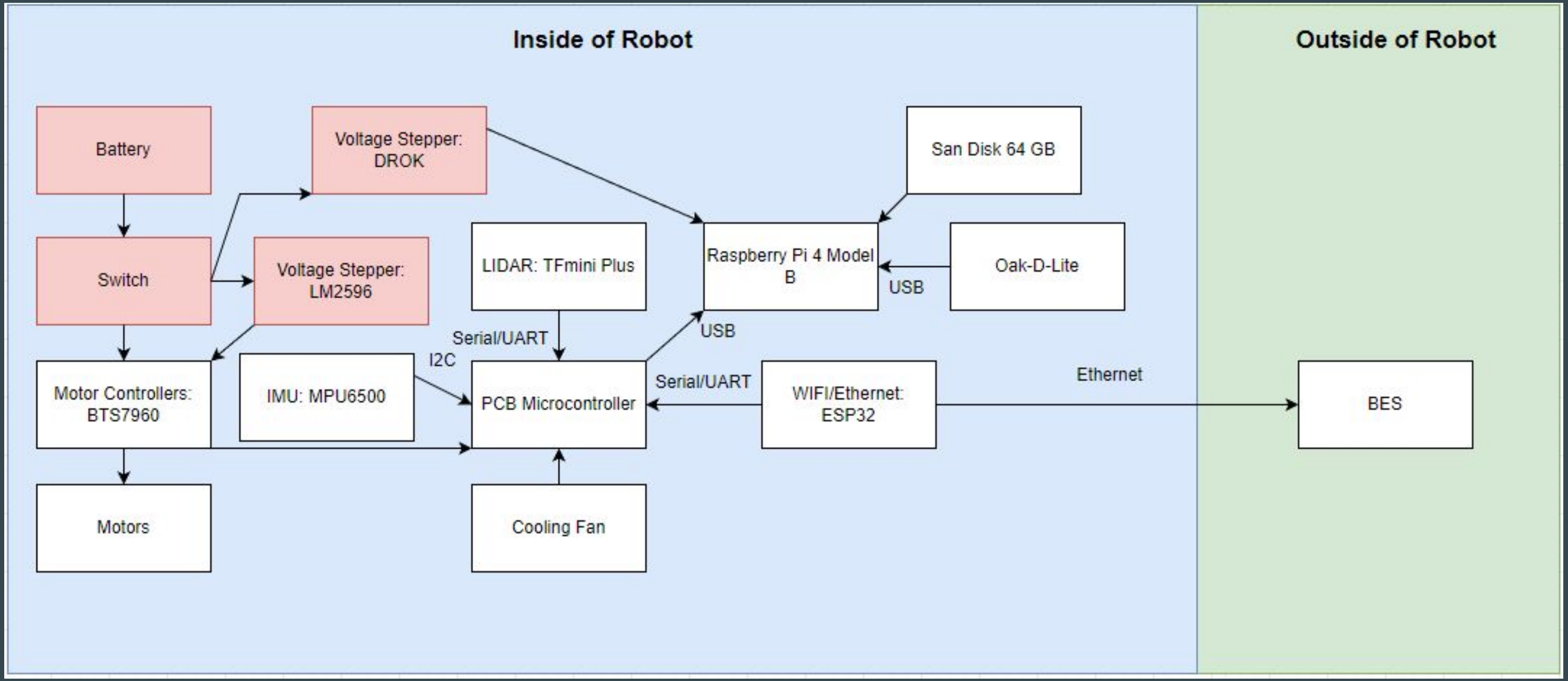
GUI - Map Tab



Robot Operating System (ROS)



Power System Diagram



Power System Requirements

2. Power Requirements

2.1 Powered by a rechargeable battery
2.2 The rechargeable battery will have a standard wall plug that can be plugged into an electrical outlet to charge.
2.3 Battery life will be determined by robot base purchased

Battery

Due to lack of finality with the motor of the robot, we estimate that in order to get at least 1 hour of battery life, we would need a battery within the range of 5 Ah to 20 Ah depending on the motors. The battery has to be 12V and should have some feasible way to recharge it below is the description of the battery of both ends of the expected range.

5 Ah Battery

For 5 Ah we have decided to utilize a SLA battery from the brand mighty max battery on amazon. The battery can handle 12 V and it is normally used for Garage door openers. The battery weighs only a little over 3 lbs so the weight shouldn't impact the robot too much. The battery is rechargeable and the initial current is 1.5 A. The only problem is that the max temp The battery can handle is around 50℃ or 122℉ based on the charts for the battery

20 Ah battery

For 20 Ah we would be utilizing an ExpertPower 12V Lead acid battery to operate the robot. This type of battery is normally used on wheelchairs and scooters and has an initial current of less than 6 A. A problem is that the battery weighs 13 lbs which is more than the base with the current motors can handle, but this battery should be able to power stronger motors with the larger capacity. The website does not contain any temp info so we don't know the max operating temperature.

Zeee 9000mAh RC battery

The Zeee 14.8V 9000mAh Lipo battery is the battery that will most likely be running our robot. This RC battery has enough power to fit within the battery range we are looking for. It is easily rechargeable through a AC/DC charger and can be recharged several times without damaging the batteries. The battery can also give a full range voltage output that our robot can use giving us faster motors to reach speed requirements if placed later.

The battery was a area in our requirements that had been deferred to later projects because of the cost it would ultimately add to the project.

Software:

- N/A

Future Updates:

- 10 Hour Battery Life That also Powers BES

Requirements Satisfied:

- 2

Direct Connections:

- Power Switch
- Buck Converters

Price:

- $83.69

Status:

- Purchased and Obtained

DC to DC Buck Converter

The Project required 2 DC to DC Buck Converters for 2 different voltages. The higher voltage would be 12V for the motors and 5V for the Raspberry Pi. We wanted to get a better buck converter for the 5V version to ensure that the Pi would be safe.

DROK Buck Converter

- USB output connection
- Volt and Amp Control
- Capable of 6V-32V to 1.5V-32V

Price

- $13.99

Status

- Purchased and obtained

Songhe LM2596S

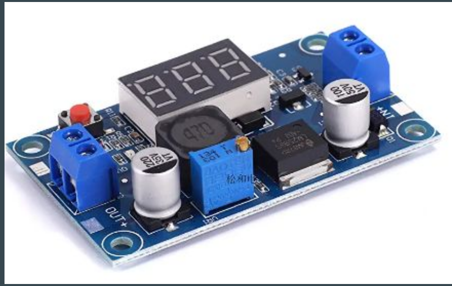
- Wire only connection
- V-40V to 1.25V-37V
- 2A

Price

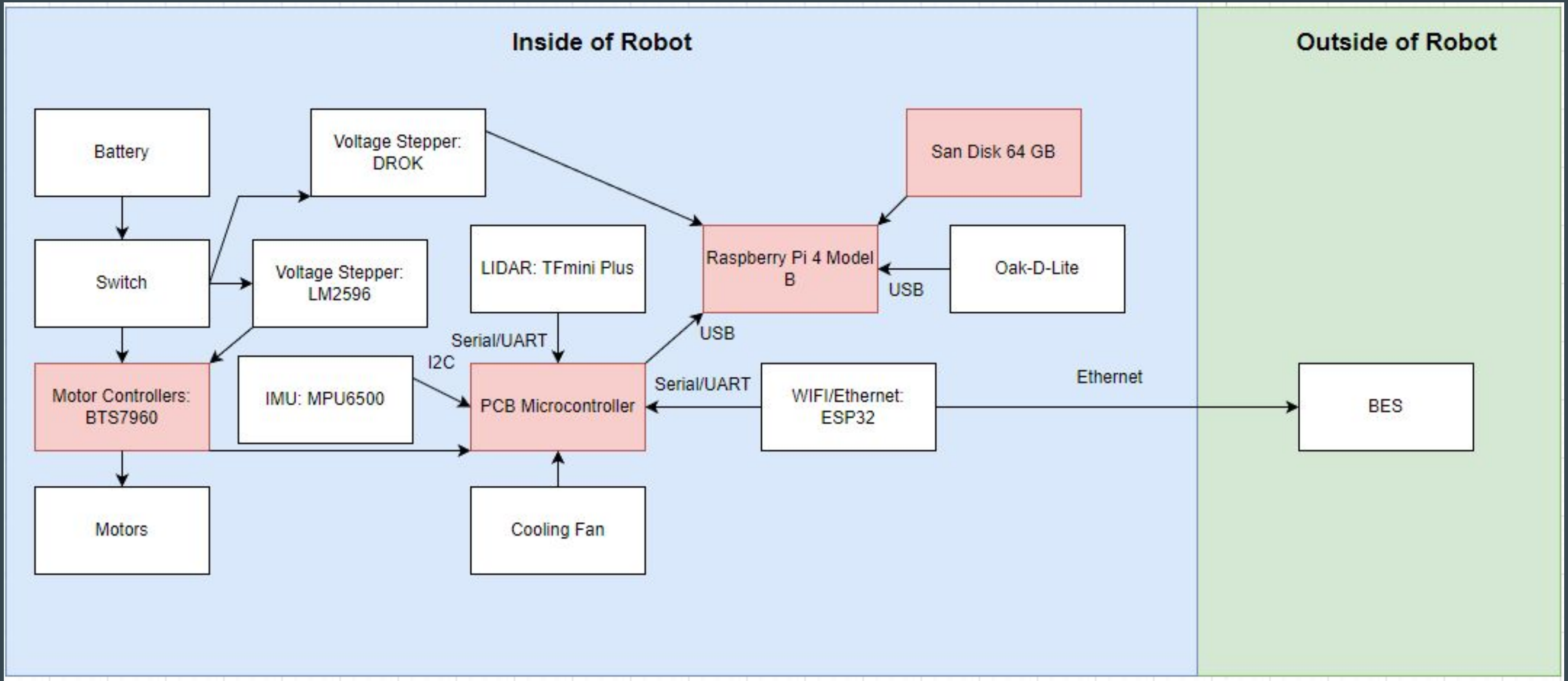
- $2.78

Status

- Purchase and Obtained



Control System Diagram



Control System Requirements

3. Control System Requirements

3.1 The robot will use open-source software aligning to the Robotic Operating System Military (ROS-M) framework and ecosystem.
3.2 Robot will be a small-scale prototype size of the product PEO STRI will implement in the future
3.3 Robot will move autonomously to its specified location
3.4 Robot will be able to detect and maneuver over rough terrain to arrive at its desired safe location
3.5 Robot will slow its speed when rough terrain or narrow paths are detected

Single Board Computers

Name	RAM (GB)	Processor	Input Voltage (V)	Price (amazon)
Raspberry Pi Model 4 B	2-8	4-Core cortex at 1.5GHz	5	Median price of $159.99
Odriod XU4	2	8-Core Cortex at 2 GHz	5	$84.75
Jetson Xaiver	8	6-Core NVIDIA Carmel Arm	5 or 9-20	TOO HIGH

Raspberry Pi 4 Model B

The Raspberry Pi 4 Model B is the brains of the robot. It will be running ROS with several packages tied to different components such as the arduino packages. This component will be focusing on obstacle detection and autonomous pathing. It will however control every other component in the system. There will be a 64GB Sandisk micro SD card for storing information about its surroundings.

Raspberry Pi was chosen because it is a company known for being reliable and there is an enormous amount of information on their single board computers.

Software:

- ROS

Future Updates:

- Update ROS software
- 8GB RAM version
- Larger memory

Requirements Satisfied:

- 3

Direct Connections:

- Arduino PCB Microcontroller (Micro USB)
- Buck Converter (USB)
- OAK-D-Lite (Micro USB)

Price:

- $134.99

Status:

- Purchased and Obtained

Microcontrollers

Texas Instruments		Arduino	
Pros	Cons	Pros	Cons
Previously worked with MSP430 Boards	More difficult to program	Easier to program	Cost
Cost	# of Pins	More Resources	Depleted market

ATmega 2560

For senior design we must make a custom PCB and we chose to do it with a microcontroller. ATmega2560 will be the powerhouse of our custom microcontroller. The reason we plan to use this chip is the number of pins that are not only available for our project now but the future. The ATmega2560 is often used on Arduino boards. The Arduino Mega2560 is the board we will be using in order to configure the robot and then test our PCB. The microcontroller will be teamed with the raspberry pi to work together in all functions of the project. The price may vary depending on if we get the first PCB correct.

Software:

- Rosserial_arduino Package
- Arduino IDE

Future Updates:

- Updating the PCB
- Adding a second

Requirements Satisfied:

- 3

Direct Connections:

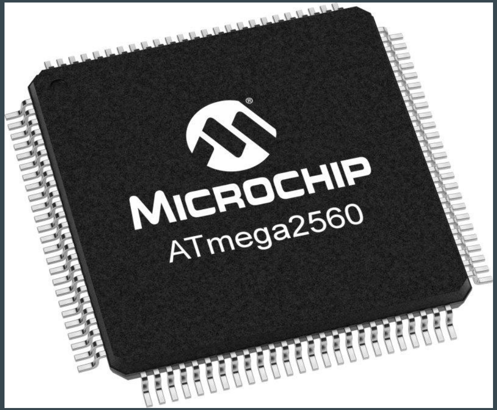
- Raspberry Pi Model 4 B (Micro USB)
- Motor controllers
- ESP32 (Serial/UART)
- TFminiPlus LIDAR (Serial/UART)
- IMU (I2C)

Price:

- Price May Vary

Status:

- Chip Obtained
- Arduino Mega2560 Obtained



Motor Controllers

While researching our project in senior design one we had already found good motor controllers for this project that had been used in previous projects. We did however keep looking for other options in case we could find something cheaper or considered a better product.

DC Motor Driver Module by Garosa

Very high power motor controller functioning up to 60A and 12V-30V

Price

- $60.06

Motor Controllers BTS7960

The BTS7960 is a high voltage motor controller. The main reason we are sticking with the BTS7960 is the how reliable it is. There are a lot of motor controllers that are available however there are a lot of duds. This is a motor controller made for Arduino. It is also a controller that could still be using in future versions of the robot.

Cons - Pins often come broken.

Software:

- Rosserial_arduino Package
- Arduino IDE

Future Updates:

- N/A

Requirements Satisfied:

- 3.5

Direct Connections:

- Custom Microcontroller
- Motors
- Buck Converter

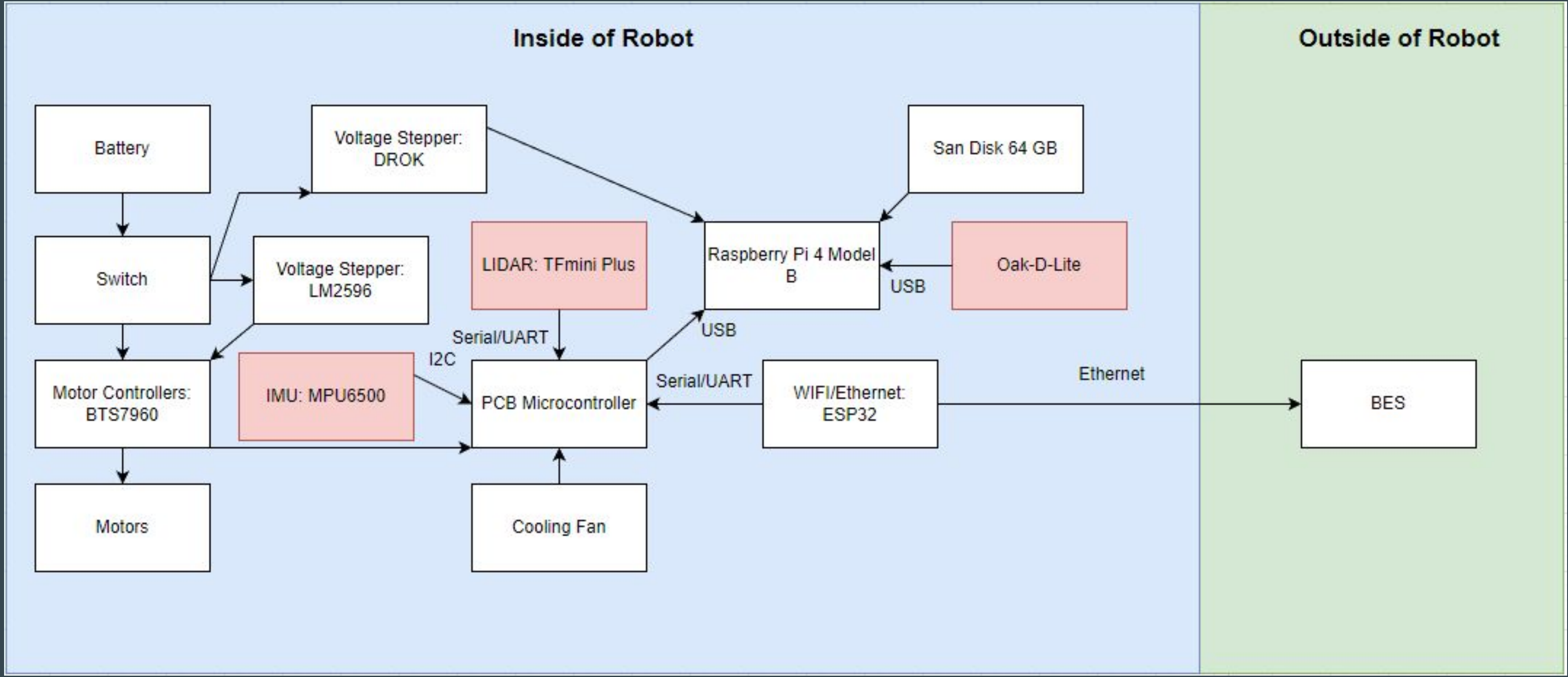
Price:

- $15.99 for 2

Status:

- Purchased and Obtained

Sensory System Diagram



Sensory System Requirements

4. Sensory Requirements

4.1. The robot will be in communication with the Windows tablet control system. A safety message will be sent when the following conditions are met:

- 4.1.1 The robot has reached its specified location
- 4.1.2 The robot has determined there are no moving elements within a 2-meter vicinity (depending on size of robot)
- 4.1.3 The robot is on a surface that is flat +- 20 degrees.

4.2 There will be a camera giving the robot a 360° view that will help in identifying people in the surrounding 2m area.

4.3 Robot will be able to detect any people within and 2 meter area

4.4 Robot will move into existing Stationary Armor Target coffin locations without interfering with any existing equipment

4.5 Robot will be able to detect collisions to avoid existing equipment or obstacles in its path

LIDAR Sensors

360° LIDAR Sensors				
Number	Name	Sensor radius Distance (meters)	Power Consumption (watts)	Cost (US dollars)
1	Slamtec RPLIDAR A1M8	12m	.5W	$99.99
2	Slamtec RPLIDAR Lidar SLAM A3	12m	3W	$595.99
3	Slamtec RPLIDAR Lidar SLAM S1	40m	> 2W	$649.99

LIDAR Sensors

Single-Point LIDAR Sensors				
Number	Name	Sensor radius Distance (meters)	Power Consumption (watts)	Cost (US dollars)
1	SmartLam MVR2EB	.3-14m	1.25W	$39.99
2	Stemedu Small TFmini-S	.1-12m	.12W	$43.90
3	Stemedu TFmini Plus	.1-12m	.55W	$49.99

TFmini Plus

The TFmini Plus is the LIDAR we will be using in order to determine if there are obstacles in our robots path. It will also help map it surroundings. This LIDAR is a single point LIDAR that is capable of working outside. It being capable of working in the outdoors is one of the biggest reason we went for this LIDAR. The LIDAR is also IP65 which was also the weather rating requirement we had before issues in budgeting

Because this LIDAR is a single point LIDAR we will need more than 1 on the robot and we have decided that the proper number should be 3. There will be 2 Lidars mounted in the front of the robot and 1 behind the robot.

Software:

- TFmini ROS package

Future Updates:

- 360° LIDAR
- Radar

Requirements Satisfied:

- 4.1.1, 4.4, 4.5,

Direct Connections:

- Serial/UART to PCB Microcontroller

Price:

- $49.99

Status:

- 1 Purchased and Obtained

Computer Vision Camera

Name	FPS (Max)	HFOV (Max)	Depth Perception	Resolution (Max)	Input Voltage (V)	Price (amazon)
OAK-D	60	69°	35m	4K	5	$249.99
OAK-D Lite	60	69°	19.1m	4K	5	$149.99
OAK-1	60	69°	N/A	4K	5	$189.99
Kinnect	30	57°	3.5m	1280x960	12	$21.99 (ebay)

Oak-D-Lite

The Oak-D contains three cameras with the middle being the primary and is what the Convolutional Neural Network (CNN) will use to determine if a person is present. The two stereo cameras are used for depth perception in order to measure the distance and determine if the person if far enough away.

The reason we picked this camera was because we needed to be able to measure the distance from the robot and person and the Lite version of the Oak-D does not contain a IMU (we have an external one) and has a lower resolution which is sufficient.

Software:

- ROS OpenCV Package
- CNN Model

Future Updates:

- 4 Cameras
- Updated CNN Model

Requirements Satisfied:

- 4.1.2, 4.2, 4.3, 4.4, 4.5

Direct Connections:

- USB to Raspberry Pi

Price:

- $157.94

Status:

- Purchased and Obtained

IMU

When searching for an IMU we understood that we could add this to the PCB Microcontroller. This would however add difficulty to an area that can already take a lot of time. Time is a big constraint in this project because of the difficulty in other areas such as ROS and that is why we decided not to implement it into our PCB.

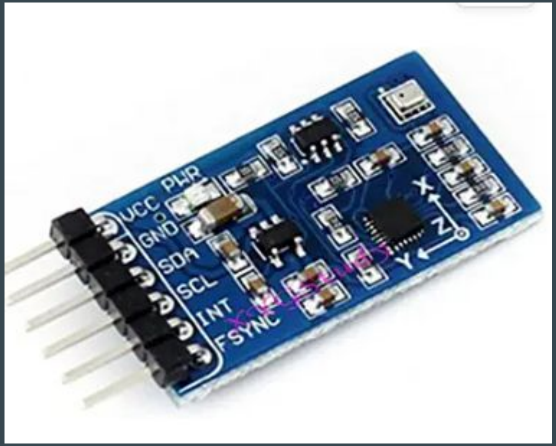
10 DOF IMU

Sensors

- 3-axis Gyroscope
- 3-axis Accelerometer
- 3-axis Magnetometer
- Barometric Pressure sensor
- Temperature sensor

Price

- $36.99



IMU MPU6500

The MPU6500 is the inertial measurement unit that we will be using in or system to understand the level and speed of the robot. This will be handled through the arduino with I2C connection. The reason for having the IMU will be to determine safety checks such as the robot isn't moving and the robot is level. The IMU consists of 3 axis accelerometer and 3 axis gyroscope

Software:

- Rosserial_arduino Package
- Arduino IDE

Future Updates:

- Implement into PCB

Requirements Satisfied:

- 4.1.3

Direct Connections:

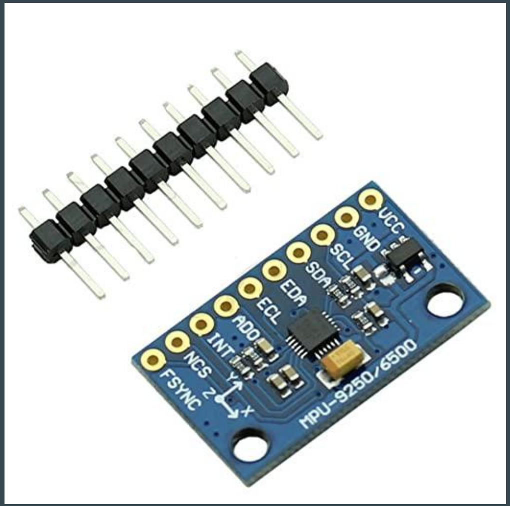
- Custom Microcontroller (I2C)

Price:

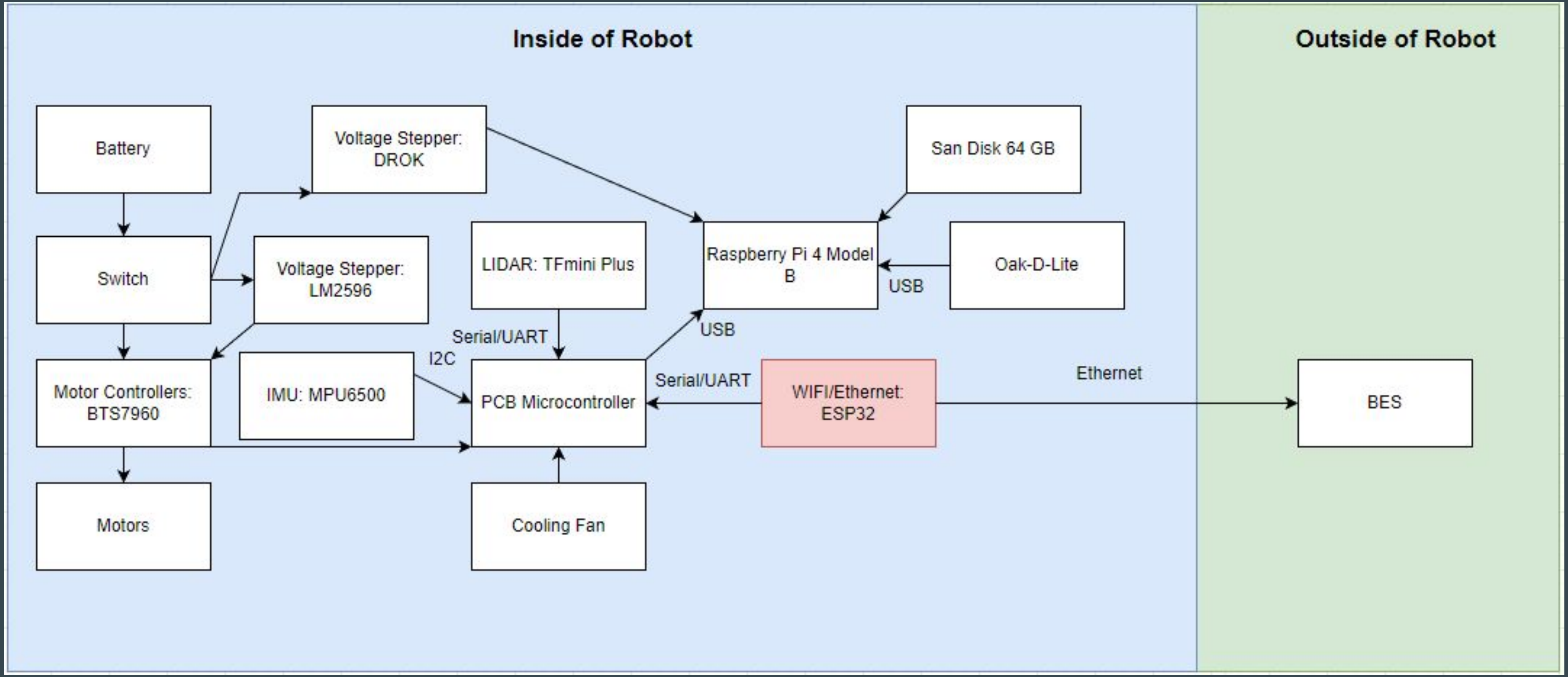
- $5.99

Status:

- Purchased and Obtained

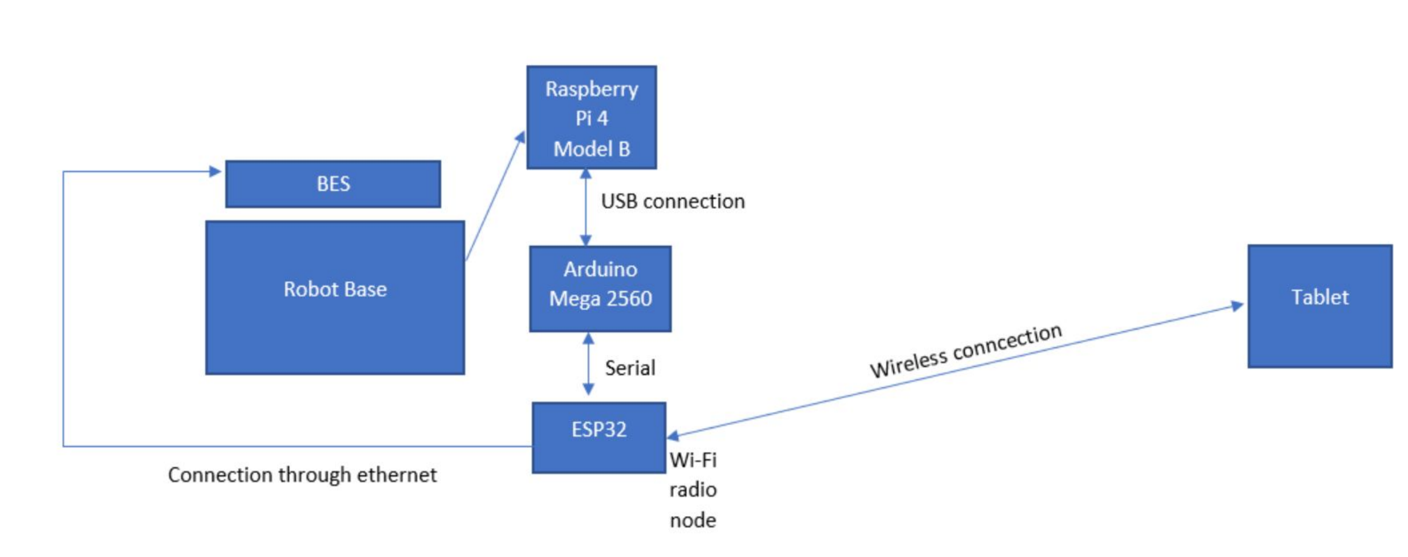


Communication System Diagram



Communication

- ESP32 will receive the signal wirelessly and communicate with BES wired
- The tablet will run Windows and the software to communicate with BES will be provided



Communication System Requirements

5. Communication System Requirements

5.1 The robot will be in communication with the Windows tablet control system. A safety message will be sent when the following conditions are met:

- 5.1.1 The robot has reached its specified location
- 5.1.2 The robot has determined there are no moving elements within a 2-meter vicinity (depending on size of robot)
- 5.1.3 The robot is on a surface that is flat +- 20 degrees.

5.2 The robot will send an activation signal to the Battlefield Effects Simulator once a safety message is received from the control system

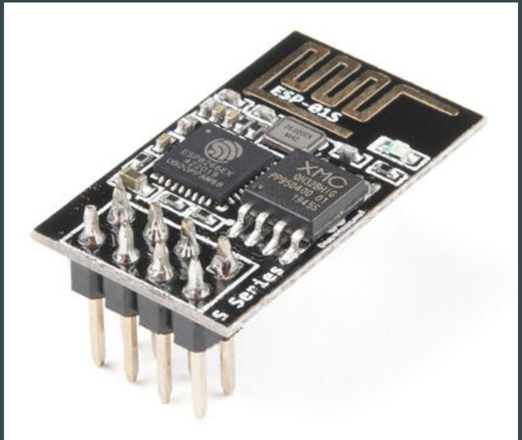
Communication Device

ESP8266

The ESP8266 was the initial communication device our group was planning on using. It would be a cheap and easy way to establish a TCP/IP connection with the control system. The main issue with this board is the ethernet capabilities. ESP8266 can work with ethernet but it is not a very typical set up and the efficiency can be poor. We need a connection tool that can send signals through ethernet to the BES easily. While researching we found that the board would not function very well with an off board ethernet connection.

Price

- $8.99



ESP32

The ESP32 will be a all in one communication expert. The ESP32 has both a WIFI antenna and ethernet port. This will make it easy for the robot to receive messages through TCP/IP from the control system and deliver proper messages to the BES through ethernet. This will be communicating back and forth with the control system the custom microcontroller and the BES.

Software:

- Rosserial_arduino Package
- Arduino IDE

Future Updates:

- DHCP Server

Requirements Satisfied:

- 5

Direct Connections:

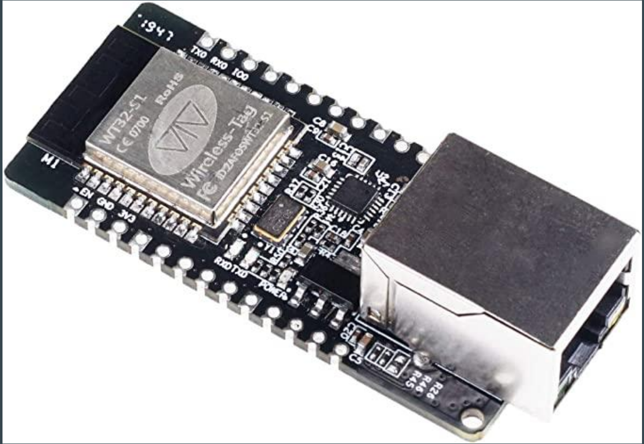
- Custom Microcontroller (Serial/UART)
- BES(Ethernet Cable)

Price:

- $16.95

Status:

- Purchased and Obtained



Drive System Requirements

6. Drive System Requirements

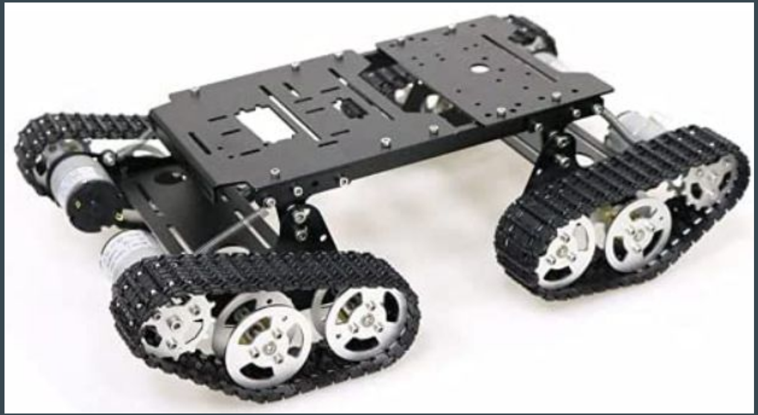
6.1 Connected components inside the frame of the robot: A custom printed circuit board (PCB), Battery bank, Motors, receiver, Receiver for accepting inputs from a person, Camera, Microcontroller, Range sensor
6.2 The robot will be utilizing motors for its treads to maneuver over rough terrain.
6.3 A custom PCB will be attached to a range sensor, receiver to receive data on the desired location, motors for the treads/wheels, external storage with AI subsystem, and rechargeable battery bank.
6.4 A range sensor will be attached to robot for it to avoid collisions while moving forward and backward.

Robot Base

The base that we are utilizing is a 4 wheel drive tank car chassis purchased from amazon. The brand of the chassis is DoHome and it utilizes 4 treads instead of 2 which was originally planned. Having access to 4 motors will allow us to have higher torque and maintain a good speed at the cost of using more energy but this gives us room in the event that there is a heavy modification added to the robot. The base with the motors that comes with the base can support a load of a little over 6 lbs, but this can be modified by utilizing differents motors on the robot. The frame of the chassis is made of aluminum alloy, the gears and bearings are metal while the treads are plastic which could cause problems down the line in outdoor testing. We can get different treads to attach to the frame so that the robot will be able to handle outdoor terrain better.

Motor parameters:

- 33GB-520 motor:
- Rated voltage: DC12V
- No-load current: 100mA
- Suitable voltage: 6-12V
- No-load speed: 170-350rpm
- Weight: 100g



MISC. or Minor Components

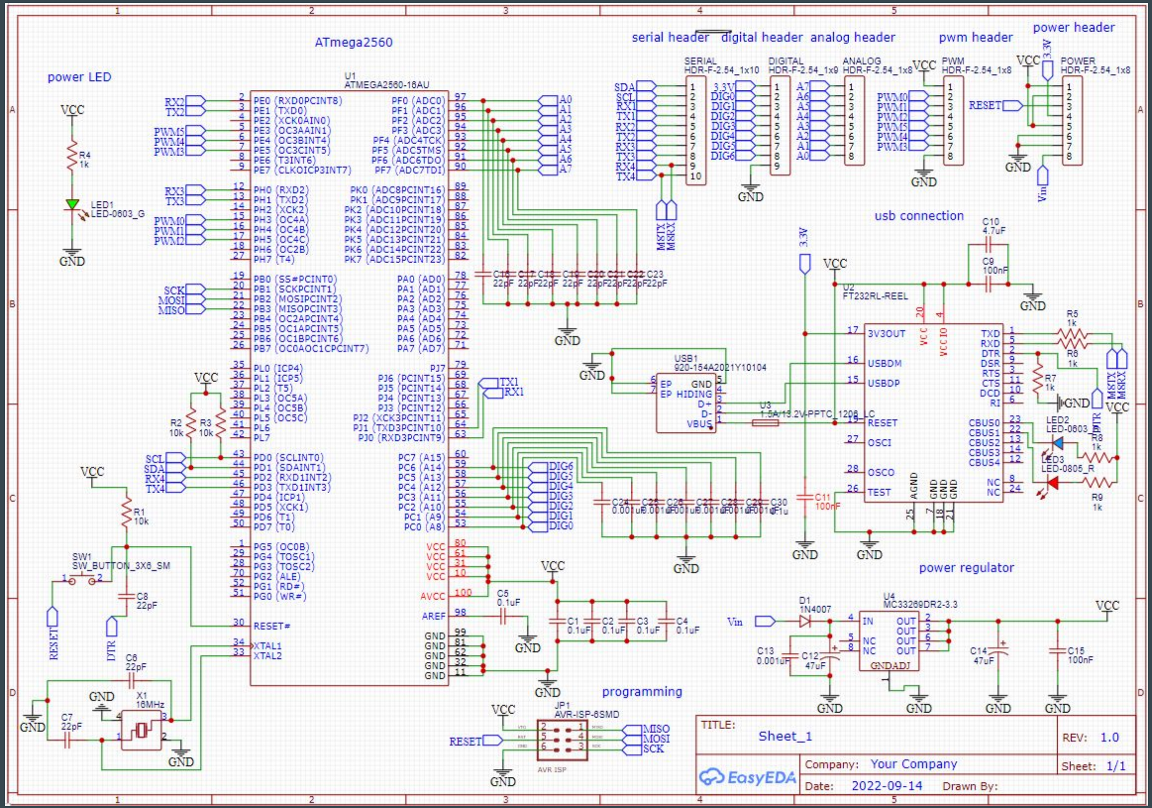
STORAGE: SanDisk Ultra 64GB microSDXC
The storage will be used mainly for ROS to understand and remember the terrain it is in. It will also be used for installing the operating system itself.

Satisfied Requirements: 1.2, 3.4

Cooling Fan:
Since the robot is going to be running outside we will need some sort of cooling. We think that the robot will be ok with 1 or 2 fans. We will just be using some extra fans we have from a previous PC build.

Red and Green LED:
These LEDs will be turned on based on whether the robot is safe to approach or it has determined the area is safe to fire in.

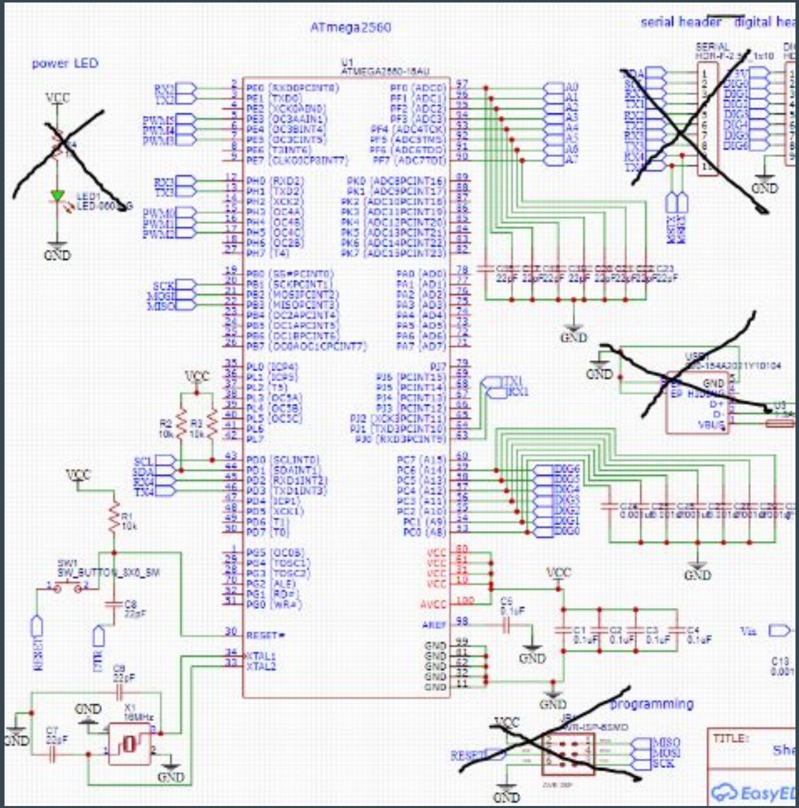
Schematics



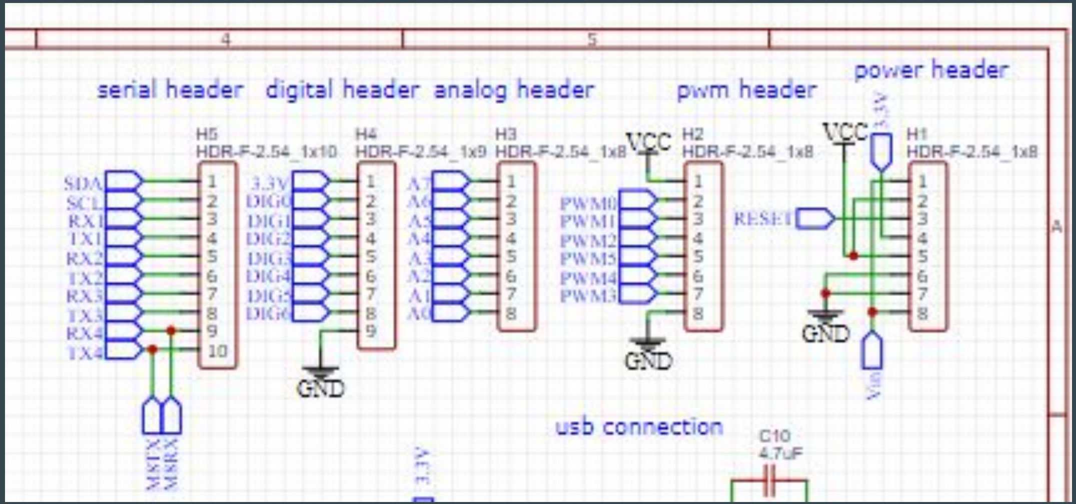
ATmega2560
power LED
serial header
digital header
analog header
pwm header
power header
usb connection
power regulator
programming
TITLE: Sheet_1
REV: 1.0
Company: Your Company
Sheet: 1/1
Date: 2022-09-14
Drawn By:

Schematic Power LED

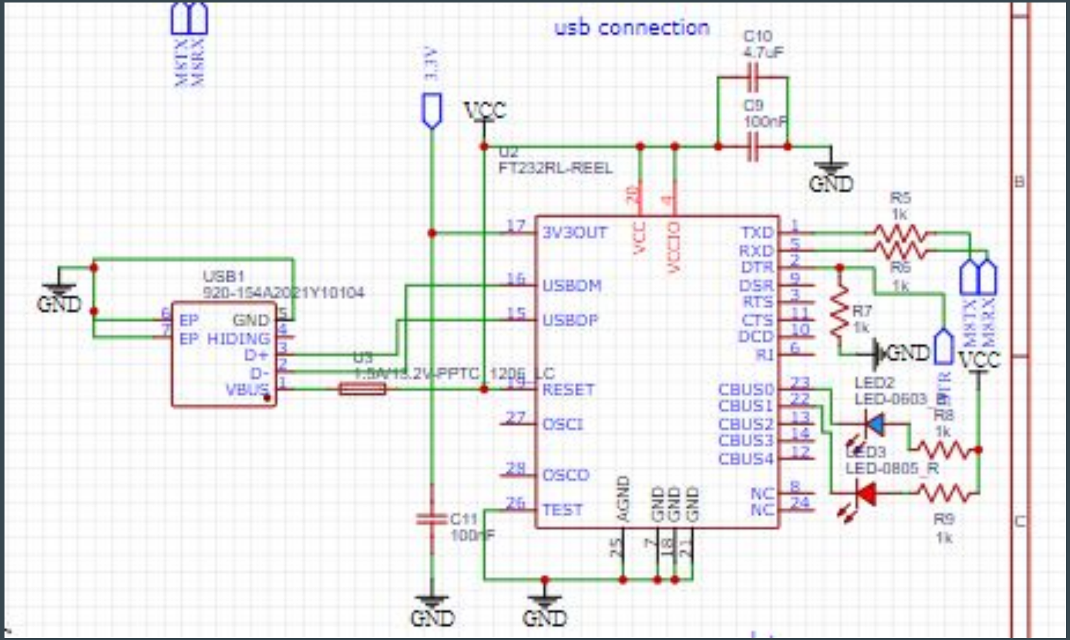
Schematic Microcontroller



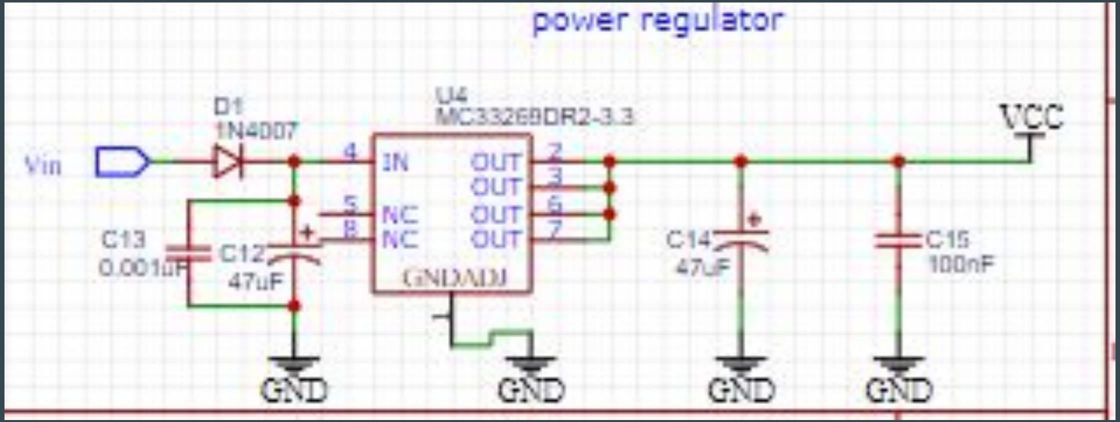
Schematic Headers



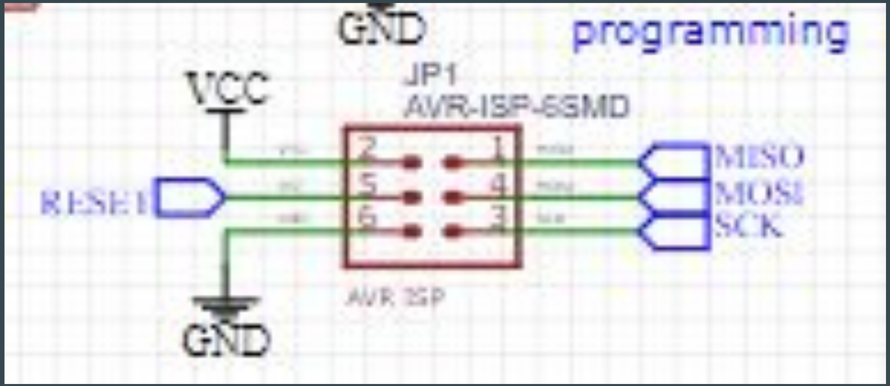
Schematic USB Connection



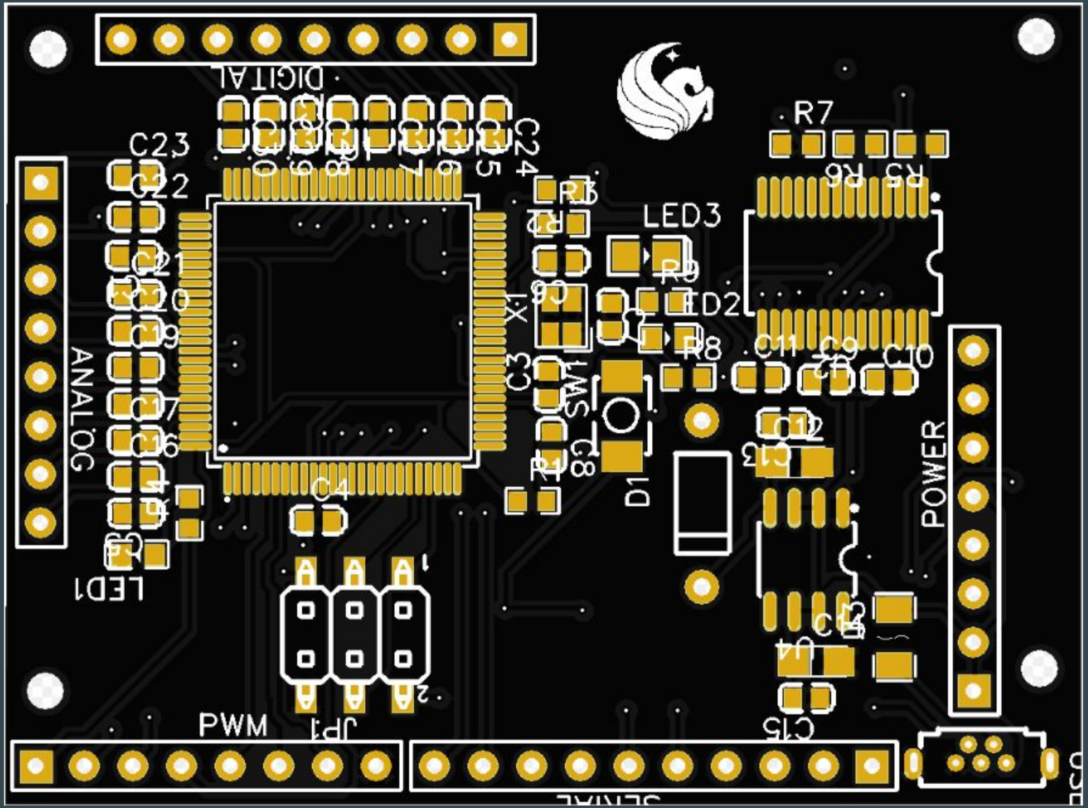
Schematic Power Regulator



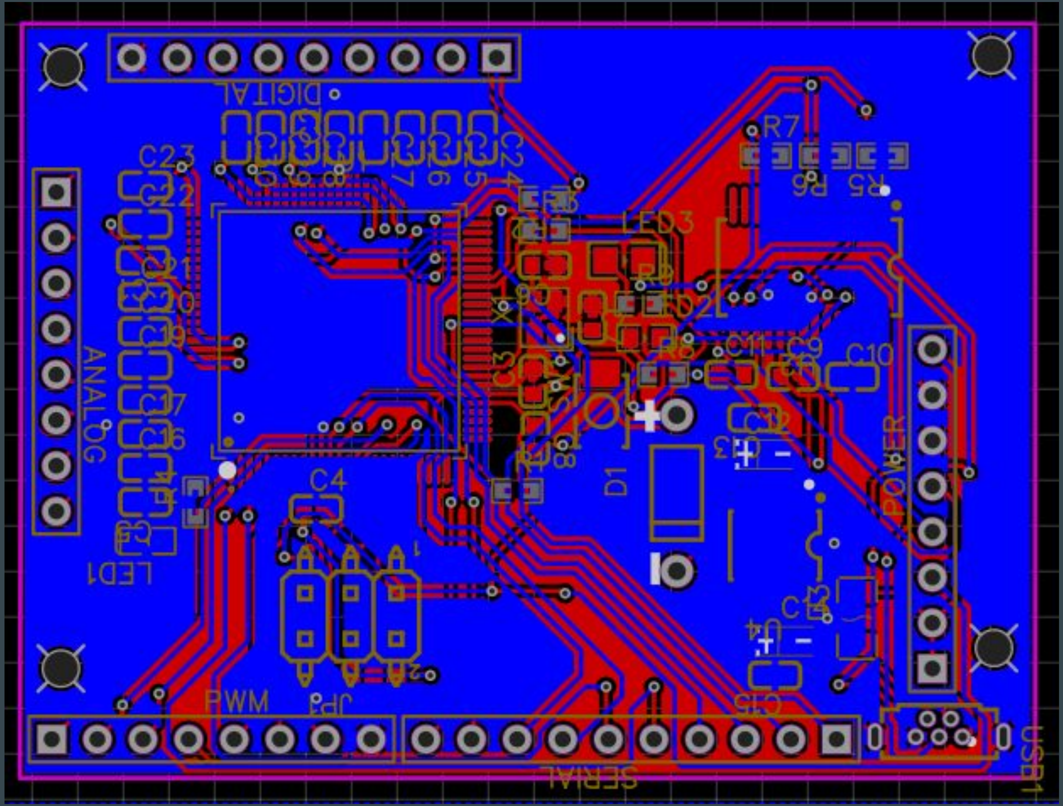
Schematic Programming



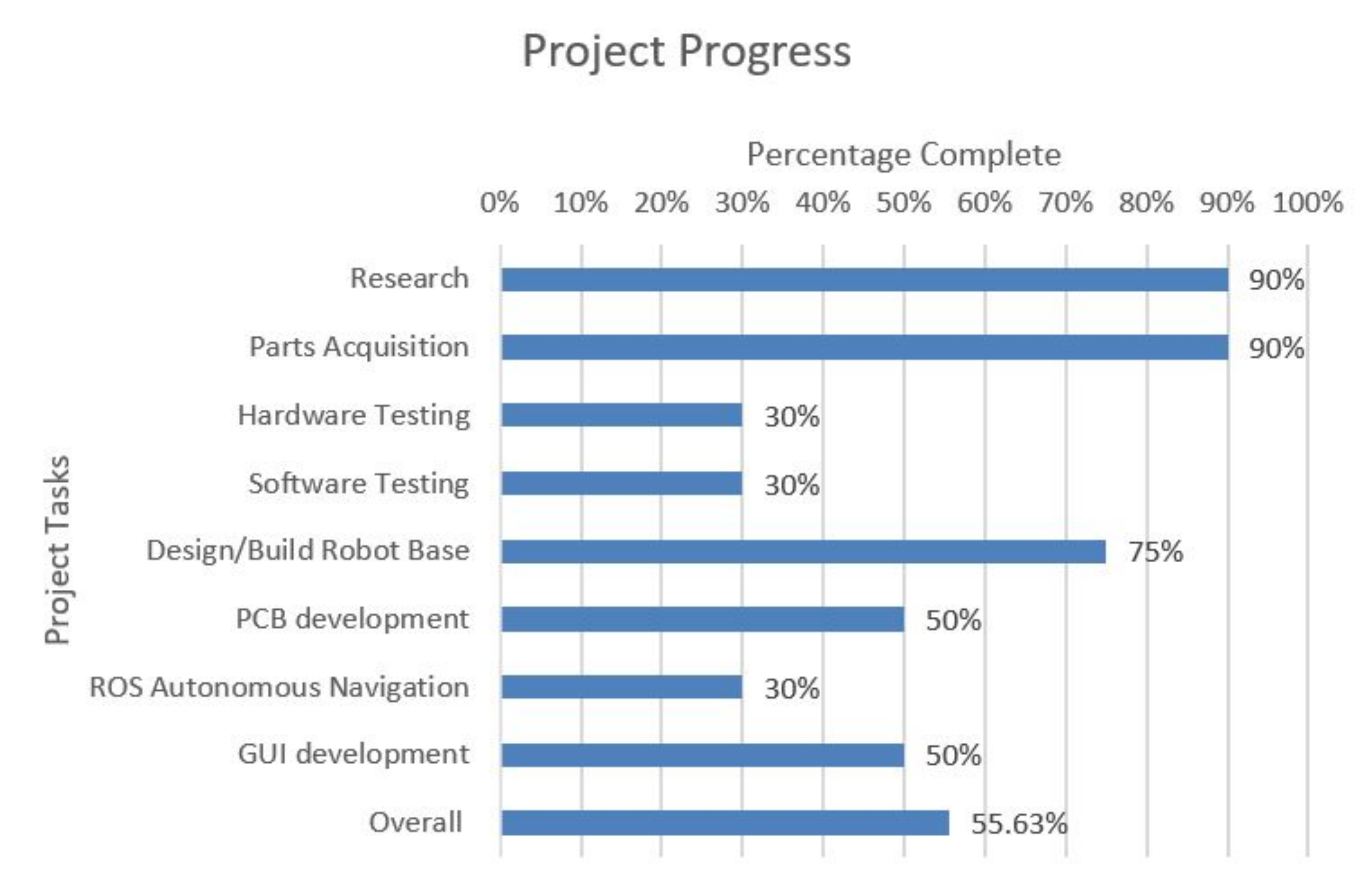
PCB Front



PCB Back


DIGITAL
ANALOG
PWM
JP3
SERIAL
POWER
USB1
LED1
LED3
D1
C4
C10
R7
R5

Progress



Project Progress
Percentage Complete
0%
10%
20%
30%
40%
50%
60%
70%
80%
90%
100%
Project Tasks
Research
90%
Parts Acquisition
90%
Hardware Testing
30%
Software Testing
30%
Design/Build Robot Base
75%
PCB development
50%
ROS Autonomous Navigation
30%
GUI development
50%
Overall
55.63%

Remaining tasks

- Soldering the PCB
- Program the microcontroller and the raspberry pi
- Connect the electronic components onto the frame
- Wire components together
- Improve on the frame to hold the components together
- Test the motors with the added weight
- Program with ROS
- Test the completed prototype
- Lots of testing
- Make any necessary changes based on test result

BUDGET	
NAME	**PRICE (TAX not included)**
CanaKit - Raspberry PI 4	$134.99
ESP32	$16.95
IMU	$5.99
TFmini Plus	$49.90
OAK-D Lite	$149.00
Robot Base	$109.99
Motor Controller	$15.99
Zeee 9000mAh	$83.69

BUDGET	
NAME	**PRICE (TAX not included)**
DROK Buck Converter	$13.99
LM2596 Buck Converter	$5.15
IMU	$5.99
Rocker Switch	$1.60
TOTAL	
SPENT	**$593.23**
REMAINS	**$406.77**

Thank you for listening to our presentation!

Any Questions?