Educational Coordinated Robotics

By

Kan, John

Lu, Louis

Null, W David

Design Document for ECE 445, Senior Design, Fall 2018

TA: Dongwei Shi

20 September 2018

Project No. 6

Contents

1 Introduction

1.1 Objective

Coordinated autonomous technologies are currently at the forefront of research efforts in the field of robotics. They enable novel and effective solutions including but not limited to search and rescue operations, distributed sensing applications, and agricultural tasks. However, communication between multiple robots is difficult to implement in educational environments because of the cost of materials and complexity of the systems. As a result, coordinated robotic applications are infrequently taught in high school engineering classes or after-school robotics programs, and students miss out on learning about this exciting new class of robots.

Our project seeks to address this issue through a PCB that enables students to develop their own network of robots with the ability to communicate. The PCB will include an overall system with three subsystems: RF (developed in the ISM band under Part 15) [1], power electronics with motor control, and sensors. Additionally, there will be a battery and motor interface.

We propose to develop three robots that are equipped with our PCB, a Raspberry Pi for computation, and a camera. We will develop a protocol that allows any robot to talk to any other robot in the network using concepts from Mesh networking and swarm robotics. Our primary goal is to develop a PCB and accompanying software that enables a student to easily experiment with a network of coordinated robots.

1.2 Background

The motivation for this project arises from the Office of Naval Research (ONR). Currently, ONR funds a national underwater ROV competition each year called SeaPerch [2]. The purpose of this competition is to introduce middle school and high school students to various STEM fields through thinking about problems and building technology of Naval relevance. However, because of shifting applications for robotic technologies in the Naval context [3], ONR is interested in funding the development of a new competition concept. Our team has ties with the Naval Research Laboratory in Stennis Space Center, MS and the lab is very interested in the outcome of our project. The hardware and software libraries we develop this semester may lead to further development to create a coordinated robotics competition.

1.3 High-Level Requirements

- We must design and build three robotic systems capable of actuation and environmental sensing.
- Each robotic system must be capable of connecting to a communication network with other robotic systems in order to transmit data and receive commands.
- Each robotic system must contain at least one marked code package which can be modified for customized control of the coordinated robotic network.

2 Design

2.1 Block Diagrams

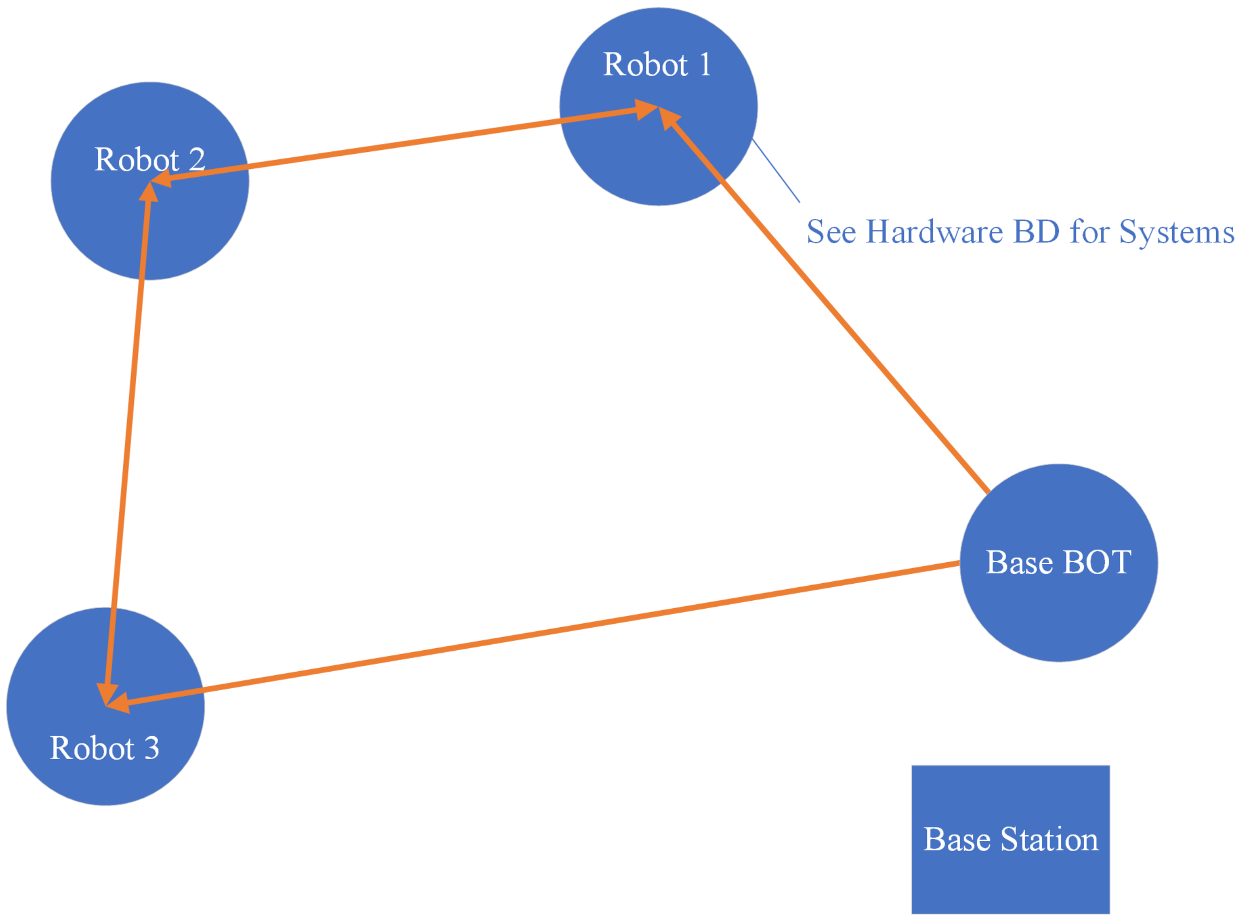

Figure 1: Overall System Block Diagram

Figure 1 shows the desired operational configuration for our system. We will build three robotic systems which will have the ability to communicate with each other and a base station. The robotic systems and the base station will be nodes in a network, and the user has direct access to the base station. This network is represented by the orange arrows and can be reconfigured for various applications.

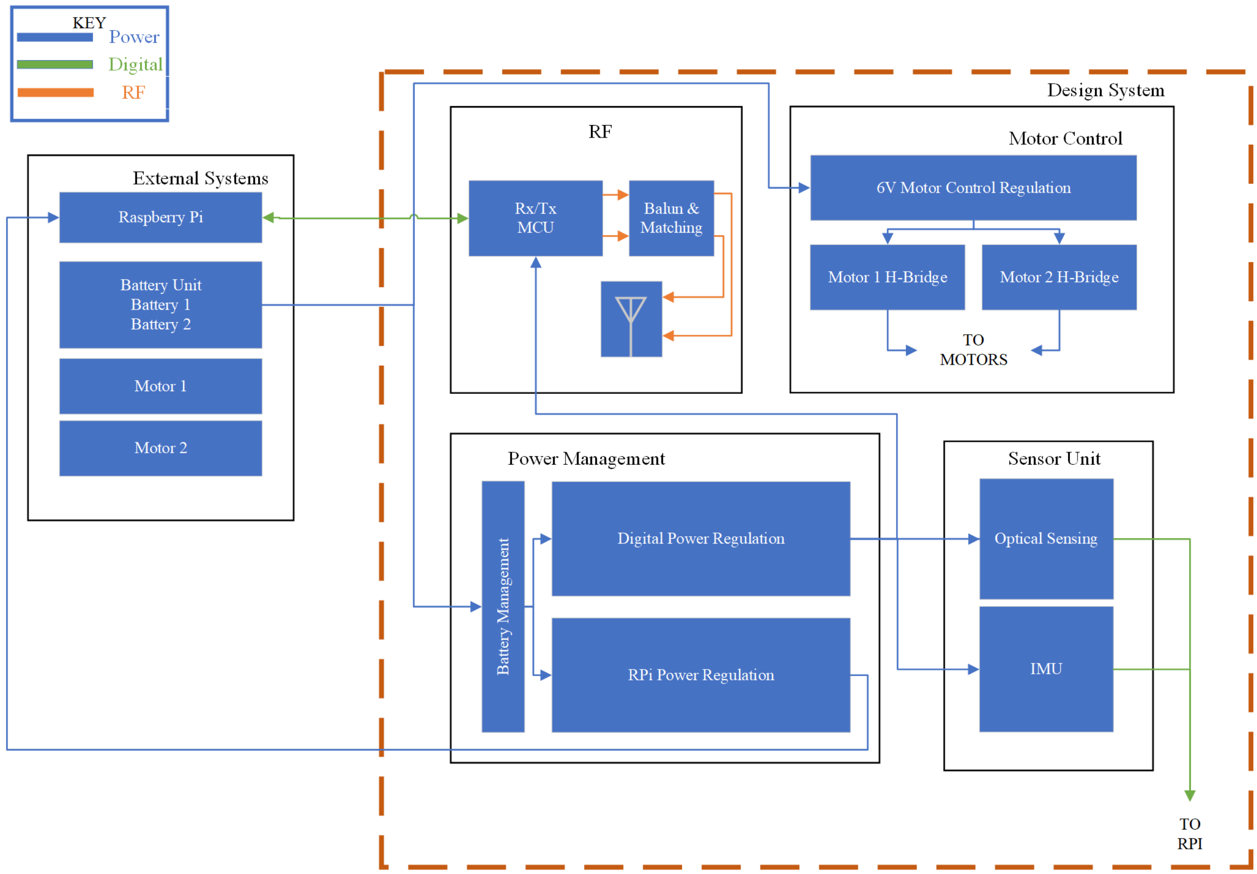

Figure 2: Single Robot Hardware Block Diagram

The block diagram shown here in Figure 2 represents the hardware design that will be present on each robotic system. The "External Systems" are off the shelf components which will integrate with our PCB. The "Design System" will be the modules that we design, print, and solder. Initially, we will design each module on its own PCB for testing purposes. However, once each module has been sufficiently verified, we will attempt to consolidate all Design Systems onto one PCB.

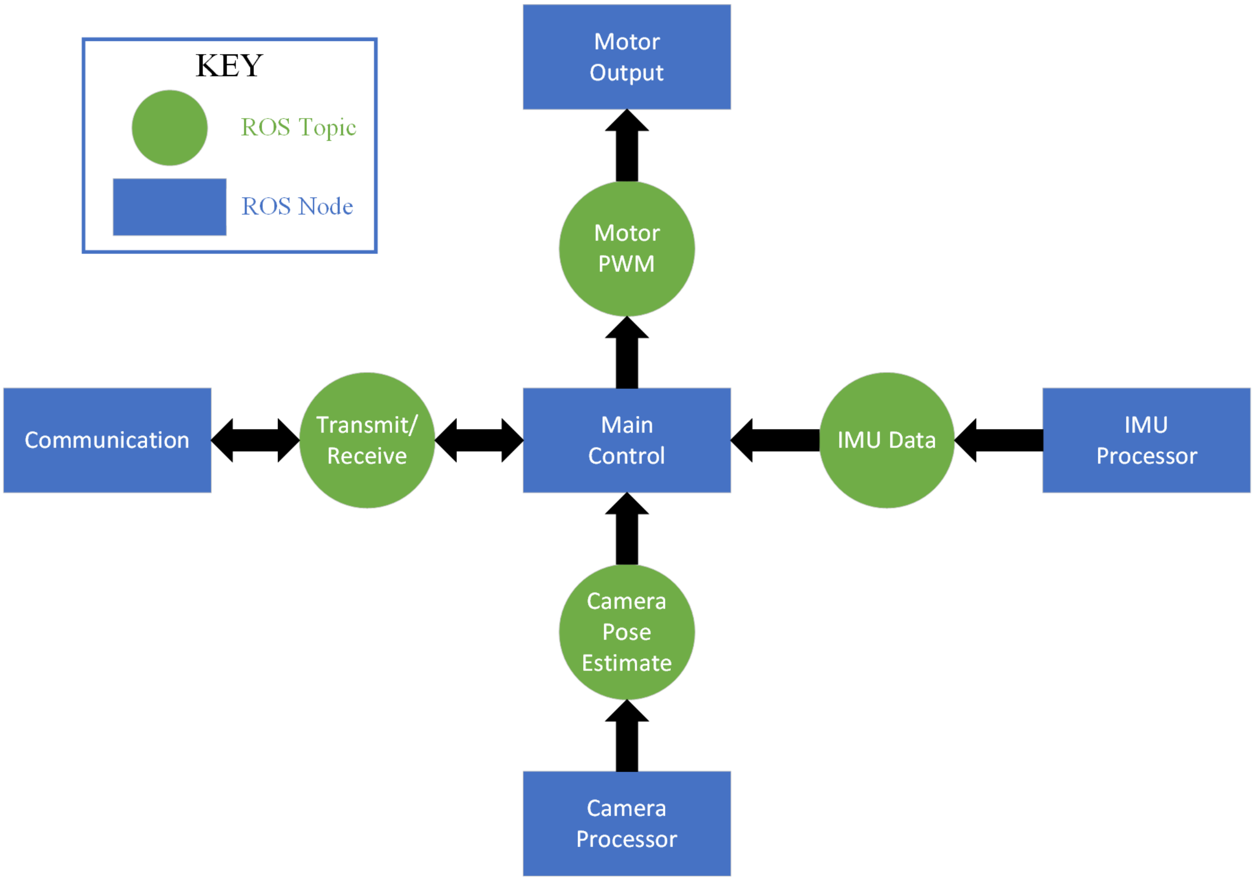

Figure 3: Single Robot Software Block Diagram

Each robotic system will be controlled through software packages that will interface with the hardware we design. The block diagram shown in Figure 3 represents a high level organization of our software design which will be present on each Raspberry Pi in the network. We will use ROS (Robot Operating System) [4] in coordination with C++ and Python to implement our to control algorithms. The ROS nodes listed are executable files that can publish messages and receive messages from the ROS topics listed. An arrow leading from a node to a topic signifies that the node is publishing messages to the topic. Conversely, an arrow leading from a topic to a node signifies that the node is subscribing to that topic. In the "Main Control" node, we will have sections of code clearly marked that can be modified in order to meet our 3rd high level requirement.

2.2 Physical Design

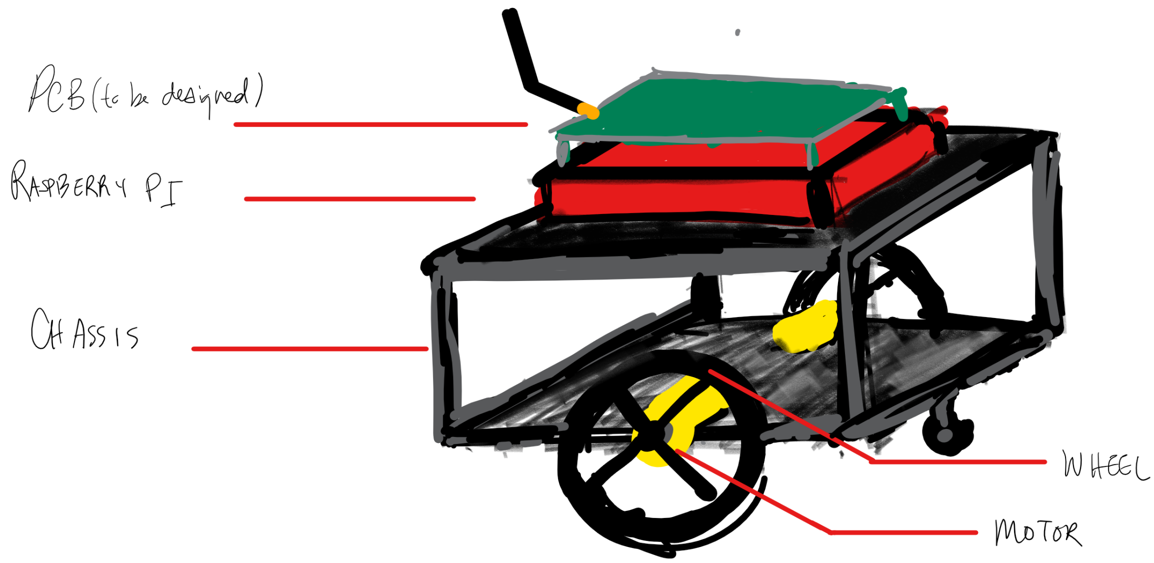

Figure 4: Single Robot Physical Design

The physical design shown in Figure 4 represents one robotic system in our network. The body will be a simple chassis with two actuated wheels on either side, and a platform with space for a Raspberry Pi, our circuit, and a power supply. Our circuit will connect to the pin-outs of the Pi.

2.3 Functional Overview and Block Level Requirement

Essentially there are two different main sections: (1) external connections with peripheral devices and (2) the main system. The external systems include two DC motors, a Raspberry Pi, and batteries. The main system is broken into four main sections: RF, power management, sensor control, and motor control circuit. The main system will be controlled by a digital system run through the Raspberry Pi.

2.3.1 Base Station (2.5 points)

The base station will be a randomly chosen robot among the robot group that the user connects to. The user can issue commands through this base station to all other robots.

Requirements	**Verification**
The user needs to connect to the base station robot through local network	Connect the Pi to a WiFi network, verify commands are issued to the Pi through SSH

Table 1: RV Table for Base Station

2.3.2 Radio Frequency (RF) Communication Module (7.5 points)

The RF section of our board will ultimately allow communication and connection between a large group of robots. To communicate between the robots, an MCU IC will be used to receive and generate the proper analog signals using the ISM band of 2.4 GHz. An external antenna will be used to gain the proper range necessary to transmit through a group of devices and proper matching circuits will be designed between the IC and the antenna. Additionally, in order to simplify the system, Bluetooth protocols will be used for communication between the robots. Use of a well-documented protocol is necessary to ensure the wireless system is able to withstand a variety of problems, from issues with noise to signal integrity issues. Thus, the RF section is critical for communication between robots.

Requirements	**Verification**
Module must be able to transmit data to another device	Check if data can be received on another device
Must be able to send at 1Mbps	send a large file over and check time required
Antenna must provide general directivity between each device	Verify with robots in different directions and orientations
Allow multicast and multiple device communication	Send bytes from one robot to multiple others and see if they all received it
(1) RF circuit must be properly matched for each module (2) Antenna should have a 50 Ohm input impedance (3) Matching circuit should bring input impedance of transceiver IC to 50 Ohm	(1) Use Network Analyzer to ensure that the s-parameters (and by extension, input impedance) match – if not matched, adjust the values of the matching network until s-parameters are matched (2) Check input impedance of antenna using a network analyzer, find s-parameters, and note input impedance

Table 2: RV Table for Battery Unit

2.3.3 Robot Communication Routing Protocol (2.5 points)

Each robot should be able to send commands to another robot connected to the network even if the the other robot cannot be directly reached. For example, if robot A is too far from robot C to communicate with it, robot A can use robot B that is connected to both to relay commands to robot C. We will accomplish by implementing a routing protocol at the software level at each robot. The routing protocol we will use is a variant of Path Vector Routing. Path vector routing is a reliable extension to distance vector routing algorithm that can easily avoid problems such as count-to-infinity[5]. It is scalable and gives optimal path between two machines. It is a proven protocol and widely used in industry such as the Border Gateway Protocol (BGP)[6]. Each of our machines will broadcast a heartbeat message to all machines it knows at a fixed interval. If one robot A fails to receive another machine's (robot C) heartbeat message in time, it will consider robot C dead or out of range, and broadcast that status. If another robot B knows the path to robot C, robot B will append itself to the path vector to robot C and send the entire path back to A. This way, A still knows how to find C even when there is no direct link between the two.

Requirements	Verification
Robot needs to communicate with each other directly	Send messages from one robot to another
Robot needs to relay message between two robots if necessary	Using software to kill a link between two robot, and test to see if the third robot can relay messages.

Table 3: RV Table for Routing Protocol

2.3.4 Power Management (5 points)

The main power section of this project comprises of various dc-dc converters to ensure everything is powered correctly. Because we need to provide power to the Raspberry Pi, we will map one pin out on our board carrying 5.1V to the Vin port on the Pi. Additionally, power for digital circuits will be converted using DC-DC converters to a voltage level of 3.3V.

Requirements	Verification
(1) Distribute 3.3V (+/- 5%) power to sensor and RF modules (2) Ensure all rise times are less than 50 ms (3) Ensure that shutdown will occur within 50 ms	(1) Verify voltage output and ripple with oscilloscope readings (2) Verify with an oscilloscope that the rise time at start up is 50 ms – if not in 50 ms time frame, turn off (3) Induce a shutdown (pull out power source) and note the shutdown time
(1) Distribute 5.1V (+/- 5%) power to Raspberry Pi (2) Ensure all rise times are less than 50 ms (3) Ensure that shutdown will occur within 50ms	(1) Verify voltage output and ripple with oscilloscope readings (2) Verify with an oscilloscope that the rise time at start up is 50 ms – if not in 50 ms time frame, turn off (3) Induce a shutdown (pull out power source) and note the shutdown time

Table 4: RV Table for Power Management

2.3.5 Sensor Unit (2.5 points)

The sensor unit will include an IMU to give each robot a relative pose estimate. We will integrate the IMU onto our PCB. It will also communicate acceleration and orientation data with the Raspberry Pi.

Requirements	Verification
Detect acceleration and orientation of robot and relay data back to Raspberry Pi on two axes up to 1 g of force	Run programs on Pi to take IMU reading from GPIO and verify position and acceleration estimates

Table 5: RV Table for IMU Unit

2.3.6 Camera Distance Detection (2.5 points)

To obtain each robot's absolute position within the environment, We will stick fixed sized colored square markers (6cm by 6cm) around the robot's environment. The distances between the markers are known, so when each robot look around its environment using its small camera, it will compare the size of the marker it sees in the camera, the real size of the marker, and the camera's focal length to calculate the distance between the marker and camera. Because we plan to place eight markers around the environment equi-dsitant from each other, the robot can rotate itself and point the camera at various angles to calculate its distance a few markers, therefore obtaining the absolute position inside the defined environment.
To calculate the distance from the camera to a marker, we will use the pinhole project formula[7]

$$\frac{y}{Y} = \frac{f}{Z}$$

where y is the height of the object on the sensor (camera), Y is the real object height, f is the focal length, and Z is the distance between the camera to the object. All four of these measurements are in the same unit. So beside the focal length of the camera, we also need to estimate the pixel size in a unit like millimeter. So the formula then becomes

$$Z = \frac{f \times Y \times H_{image}}{y \times H_{sensor}}$$

where H_{image} is the image height in pixels, y is the object height in pixels and H_{sensor} is the sensor height in millimeters.

Requirements	Verification
Camera on robot needs to calculate its distance to any arbitrary marker.	Place the marker at a known distance away from the camera and verify that the camera estimates a reasonable value.
Rotate on its spot to calculate distances with multiple markers.	See below
Infer its absolute position given distances to several markers.	Two markers will be setup at known distances from the robot in two directions, the robot will rotate and infer its absolute location.

Table 6: RV Table for Camera Distance Calculation

2.3.7 Motor Control (5 points)

A simple open-loop system using an H-bridge will allow the motors to be controlled from the GPIO pins on the Raspberry Pi. A separate battery will be connected to this section, and the proper step-down DC-DC converter will be used to provide power to the two motors. A feedback loop could potentially be implemented to provide precise speed control for better pose estimates.

Requirements	**Verification**
Control Motors using two H-bridge drives	Feed various PWM duty cycles into the SN754410NE chip and observe motor speeds driven by a power supply.

Table 7: RV Table for Motor Control

2.3.8 Motors (2.5 points)

The motors must be low-cost but must provide enough torque in order to move the robot. The motors will interface with the H-bridge unit, and will draw power from the 6V source in the battery unit. The motors will be part of the external interface devices, and will be bought from a supplier.

Requirements	**Verification**
Motors need to provide enough torque for the robot	Install motors on the robot and see if the robot moves
Motors need to operate at 5V ($\pm$10%)	Use a power supply to test that motor work at 5V
Motor draws 1.5A($\pm$10%) with load	Take multi-meter reading of current through motors with load.

Table 8: RV Table for Motors

2.3.9 Raspberry Pi

The Raspberry Pi is not a part of the development for this project, but is a necessary external input and output device. Each robot will have a Raspberry Pi in order to collect data, make decisions, and process commands from internal sensors and from other robots in the network. The Raspberry Pi will be connected in some way to every module designed for this project, including the RF chip, sensors, power management, and motor drive modules.

Requirements	**Verification**
Must have enough GPIO and SPI interfaces	Pi 3B+ satisfies such requirement
Have a processor strong enough to complete some simple computer vision algorithms at 1 fps	Run desired algorithms on Pi to measure performance

Table 9: RV Table for Raspberry Pi

2.3.10 Battery Unit (5 points)

The battery unit will contain two separate power sources. This is because of the broad range of devices our system will need to support. Bluetooth Low Energy devices such as the Nordic Semiconductor chip, typically draw around 13.5 mA of current at 3.3V in an on state and 26 μA of current in a low power state [8]. This power need contrasts with the power needs of the motors, which generally require 6V and can consume up to 1.2 W of power per motor[9]. These different power needs require various voltages and dc-dc converters. Thus, we will use 2 power sources for the scope of this project.

Requirements	**Verification**
Supply enough power (8W) for approximately 30 minutes (4 Wh)	(1) Use a resistive load to discharge batteries until completely drained (2) Analyze power measurements of voltage and current using an oscilloscope to battery (3) Note the point at which the voltage drops lower than 20% of rated voltage (4) Verify that the time to discharge was 30 minutes

Table 10: RV Table for Battery Unit

2.3.11 Software System (15 points)

Our software packages will be written in C++ and Python using the ROS programming environment. There will be a copy of our software system deployed on each robot in the network. It will interface with the the sensors, motors, and communication system in our circuit. We will have executable files designated to gather data, transmit and receive communication, process computations, and drive motors.

Requirements	**Verification**
Interface with RF module	Send and receive bytes to and from another robot
Take in raw data from IMU and camera units	Sanity check on values read in
Output correct soft PWM signal from GPIO pins at 100Hz	Use oscilloscope reading to verify

Table 11: RV Table for Battery Unit

2.4 Risk Analysis

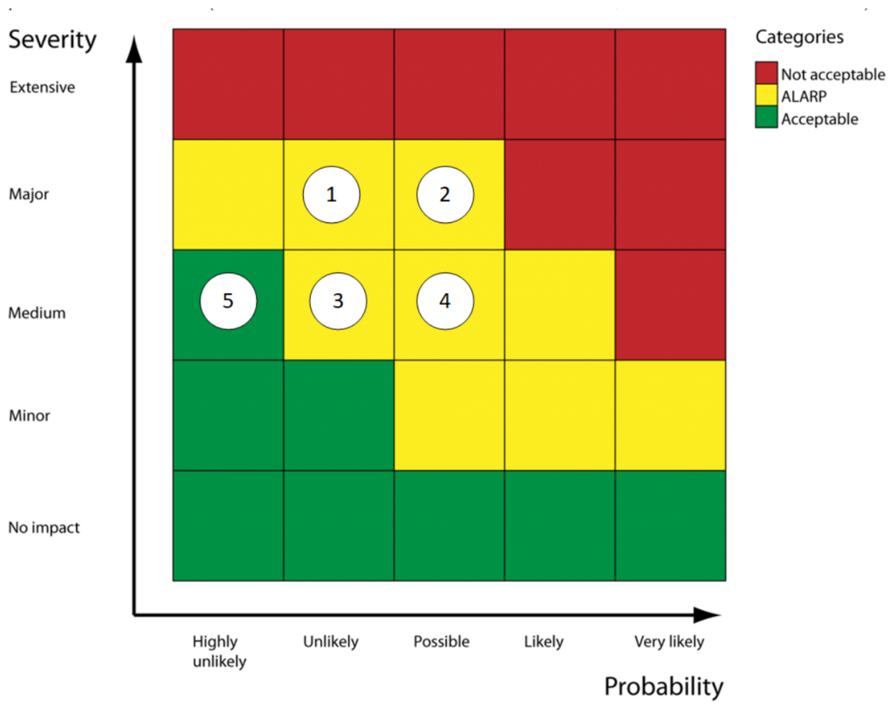

Figure 5: Risk Assessment

For the above figure 5, the numbered bubbles represent:

1. Sensors
2. RF Communication
3. Power Distribution
4. Motor Control
5. External I/O

A couple sections of this project have some risk, but overall, the risk is either ALARP or acceptable. The sections with the greatest amount of risk include RF communication modules and the sensor modules. The power management system, while important, has low risk as they are generally simple to construct and debug. The sensor system is most likely lower risk than the RF communication module, because the entire idea of coordinated robots hinges upon the communication between multiple robots using an RF signal. Thus, in order to ensure that one of the basic requirements are met, an RF communication system must be completed. Thus, it is the most high-risk section of this project. Following the RF system, the positioning and sensor unit is the next highest risk area as the movement of the individual robots depends upon the sensory data. Finally, while integral to the operation of the project, the power section of this PCB is a lower risk area than the communication and sensing section of the robot.

2.5 Supporting Material

2.5.1 Circuit Schematics

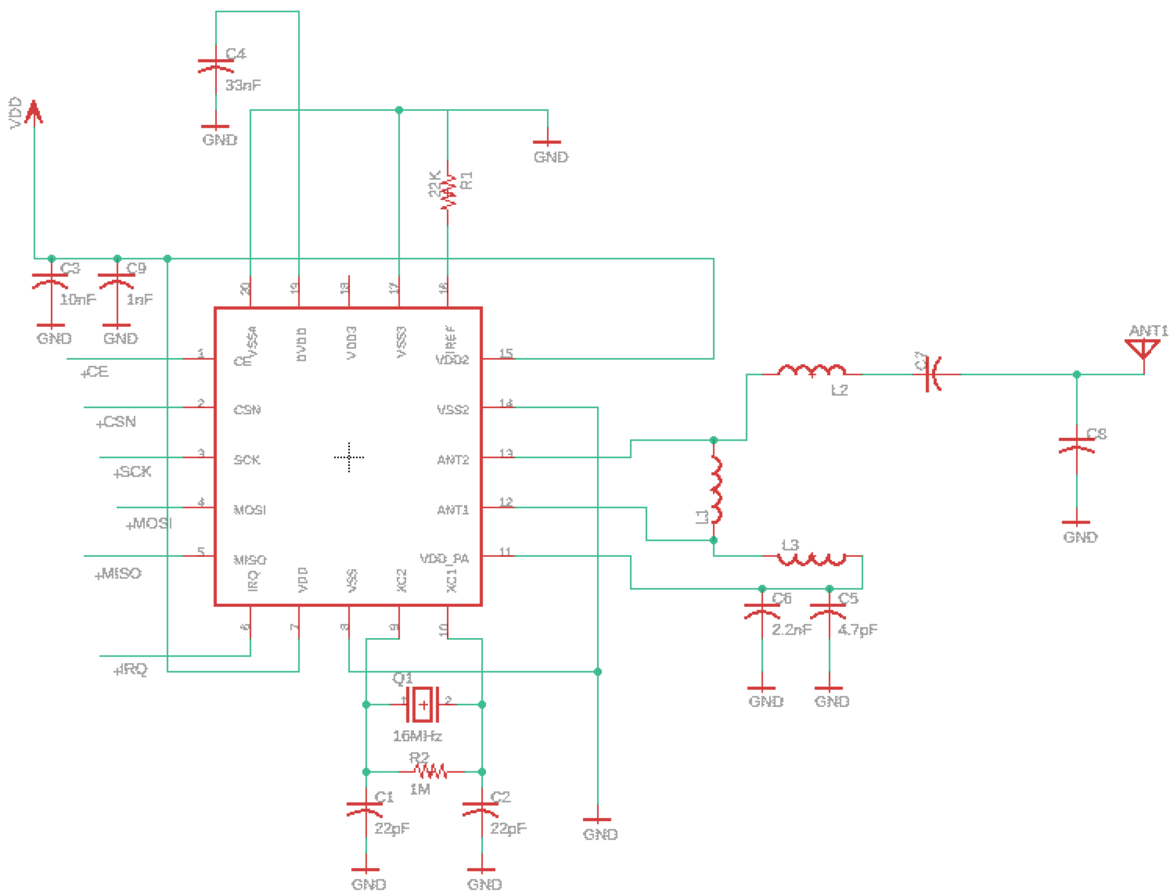

Figure 6: RF Transceiver Circuit

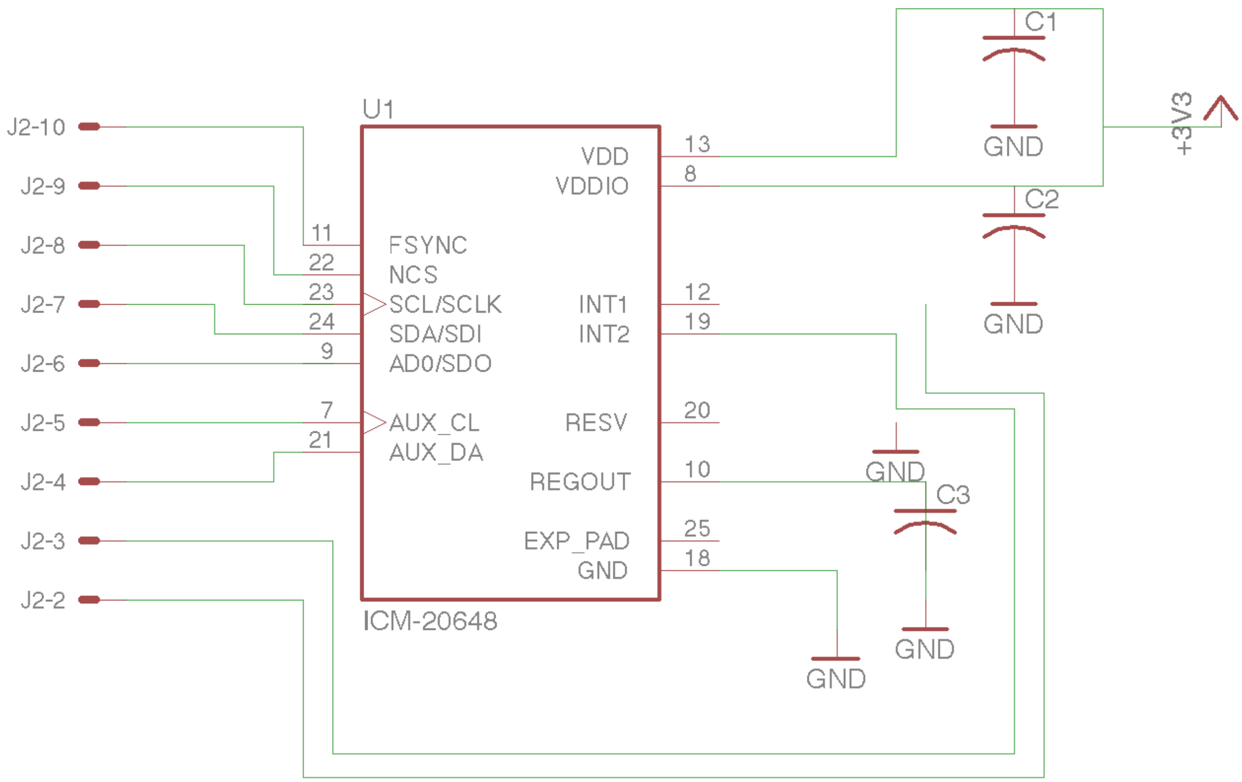

Figure 7: IMU Circuit

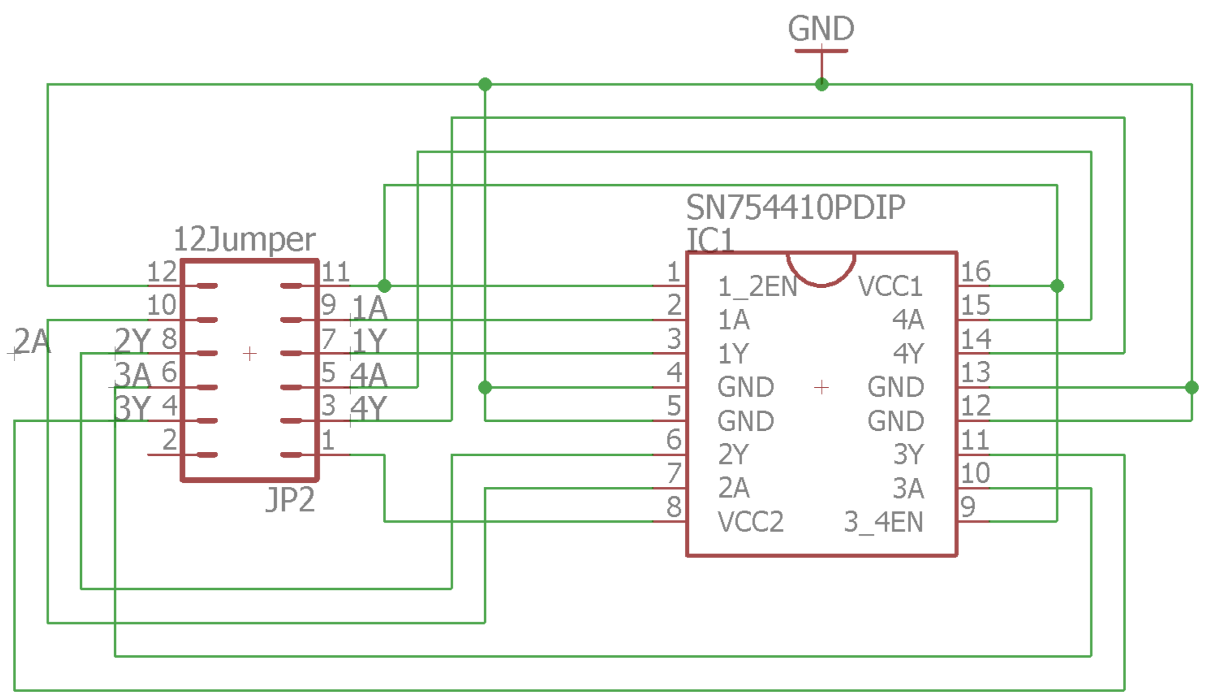

Figure 8: Motor Module Circuit

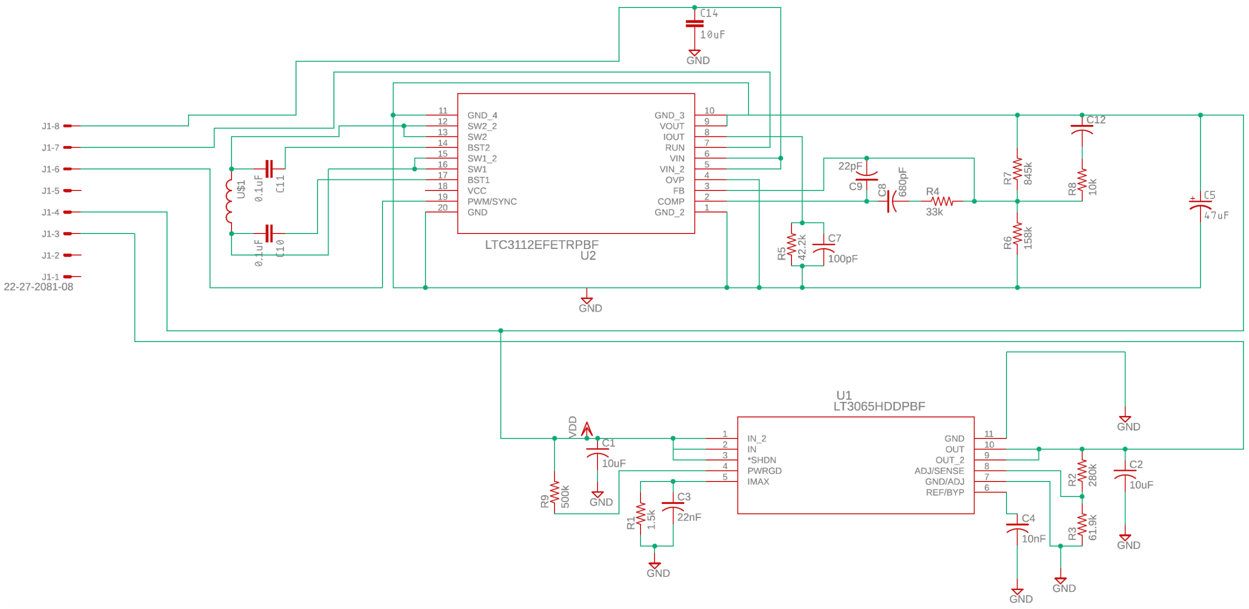

Figure 9: Power Distribution Circuit

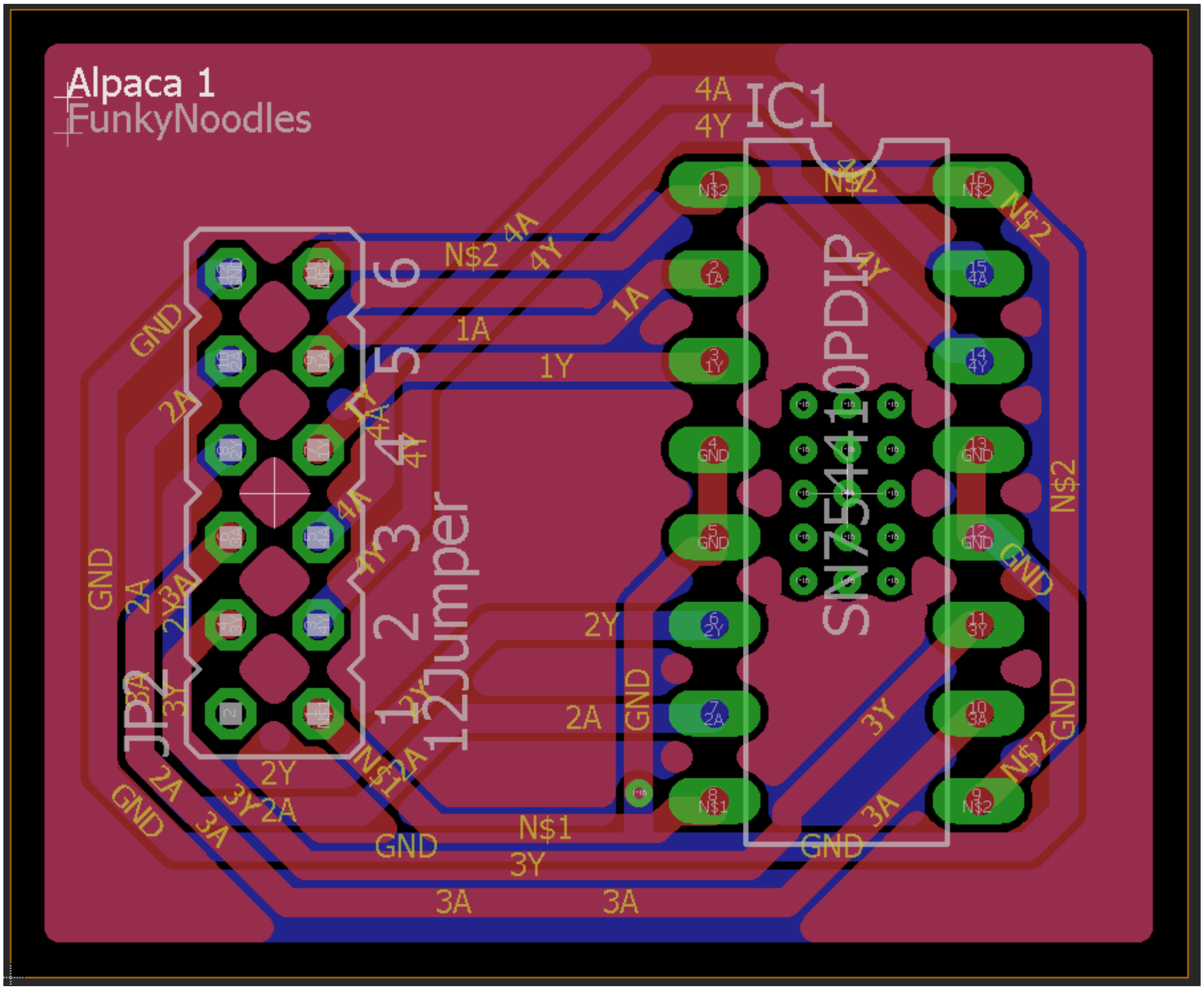

Figure 10: Motor Module Layout

3 Tolerance Analysis

3.0.1 Introduction

Matching networks are extremely important components of RF systems. If two different components in any electrical system are not properly matched, power and information losses will occur between components. Due to the extremely low power and high signal integrity requirements of RF systems, matching networks are of utmost importance. In our project, one of the key areas where tolerances may cause a great deal of problems is in the RF module of the PCB. Within the RF module of the board, the main area where these matching networks are necessary is in the transition between the NRF24L01+ chip and the antenna used to broadcast the signals. This section will analyze and determine how tolerances of the inductors and capacitors will impact the transfer of information and power level between these two different components.

Matching networks are comprised of a collection of three or four inductive and capacitive components [10]. Because of the nature of physical components, inductors and capacitors have tolerances depending on frequency of operation as well as temperature and voltage factors. The overall impedance of the input and output ports of the matching networks depend upon the actual impedances of the components. Thus, if the tolerances cause the inductors and capacitors to have a change in value, then the impedance of the input and output ports of the matching network will change, and there will be losses incurred in the system. Thus, in order to ensure that the operation of the circuit is proper, then the impact of the matching network on the power losses will be analyzed.

3.0.2 Motivation

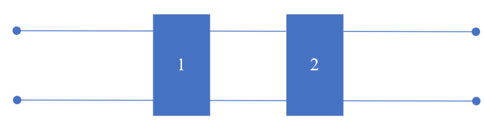

Figure 11: Two Port

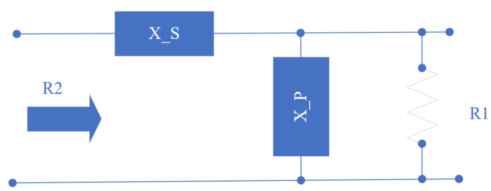

Figure 12: General Matching Network

An example picture of the interface between a two port network is shown in Fig. 11. Note that if the input impedance of device one (1) is 100Ω, while the input impedance of port two is 50Ω, then a matching network, as shown in Fig. 12 is necessary. While input impedances are useful, they are not as useful as looking at the reflection coefficient. Reflection coefficients [10] are found to be

$$\Gamma = \frac{Z_1 - Z_0}{Z_1 + Z_0} \tag{1}$$

In this case, the goal of the matching network is to minimize the reflection coefficient at the interface of the two two-port modules. Thus without the impedance matching network, if the impedance of device 1 is 100Ω and the impedance of device 2 is 50Ω, the reflection coefficient will become:

$$\Gamma_1 = \frac{Z_1 - Z_0}{Z_1 + Z_0} \tag{2}$$

$$\Gamma_1 = \frac{100 - 50}{100 + 50} \tag{3}$$

$$\Gamma_1 = \frac{50}{150} \tag{4}$$

$$\Gamma_1 = \frac{1}{3} \tag{5}$$

Similarly, for the terminating load, which is already at 50, the reflection coefficient is zero. It is known that the mismatch factor is related to the reflection coefficient comparison of both the source and the load.

$$MF = \frac{(1 - |\Gamma_1|^2)(1 - |\Gamma_2|^2)}{|1 - \Gamma_1\Gamma_2} \tag{6}$$

$$MF = \frac{1 - |1/9|}{1} \tag{7}$$

$$MF = \frac{8/9}{1} \tag{8}$$

From the mismatch factor [10], the losses can be found:

$$ML = -10log(MF) \tag{9}$$

$$MF = 0.511dBm = 1.1mW$$

The impact of having such a reflection coefficient of $\Gamma = \frac{1}{3}$ is especially seen when an analysis is done on the power. A typical RF signal in a 2.4 GHz line may have a power level of -20 dB. If this is a transmission going from the antenna to the MCU, and there is a reflection coefficient of $\frac{1}{3}$, then there will be a loss of 0.511dBm. This is enough to put it outside of the necessary power level for the MCU. However, with a reflection coefficient of 0, it is quite possible to have a power transfer of around 100%. This means that the signal stays well within the power level necessary for the input of the MCU and that the signal can be properly received. Thus, the matching network is absolutely necessary for the matching between the MCU and the antenna.

3.0.3 Tolerance Analysis

Note that the components have tolerances and these tolerances have a direct impact on the impedances, as seen in the derivation in Example 1. Assuming that the tolerances are limited to 10% for inductors and and the capacitors also have a tolerance of 10%, then the worst case scenario that occurs is when the inductances and capacitances swing the farthest away from their predicted value. Tables 12 and 13 indicates that the worst case is when the capacitor and inductors are farthest away from their labeled value, the value of the impedances are also listed. Impedance is related to frequency and to the value of the component [10].

$$Z_L = j\omega L \tag{10}$$

$$Z_C = \frac{1}{j\omega C} \tag{11}$$

A lossless matching network will be used in order to transform impedances. While the exact values and setup of the network is still being analyzed, based upon the impedance of the IC chip, found through scattering parameters, a matching network will be created. If the input to the IC has a value of 100 Ω, then the matching network must cause the antenna, which is at 50 Ω to appear to be 100 Ω. In this case, a sort of high pass topology is formed.
Without further derivation, the value of the components, X_s and X_p shown in 12 will be chosen such that the values will be matched. Z_in relies to a great extent upon the values of X_p and X_s, and when they are summed together, Z_in becomes:

$$Z_in = jX_s + j\frac{X_pR_1^2}{R_1^2 + X_p^2} + \frac{X_p^2R_1}{R_1^2 + X_p^2} \tag{12}$$

If, for example, X_p is already known and selected based off of a capacitance of 4.2pF so that X_p = -100 Ω, while X_s is a value of 20.8nH so that X_s = 50Ω, each with tolerances of 10%, the values of Z_in are listed in 13, with maximum and minimum values of Z_in based off of tolerances. Note that there is ML related even with the tolerance of the impedances, and thus, it is key that high-quality, low impedance devices are chosen and that the matching network is accurately constructed. This can also be seen in the graph Fig. 13, which indicates that as the input impedance moves away from being matched to 50Ω, there are losses. Even with tolerances nearing 30%, there can be significant power loss. Additionally, note that this impedance matching network, even with issues of tolerances in the components, still has a much better mismatch loss (.551 dBm versus approximately 0.02 dBm) than if there was no matching. Thus, impedance matching is a key component of this project and must be handled well.

Component	Value (Real)	Tolerance	Max Value	Min Value
Inductor	3.3nH	10%	3.64nH	2.97nH
Capacitor	0.66pF	10%	0.729pF	0.594 pF

Table 12: Worst Case Loss Analysis for Inductor and Capacitor Tolerances

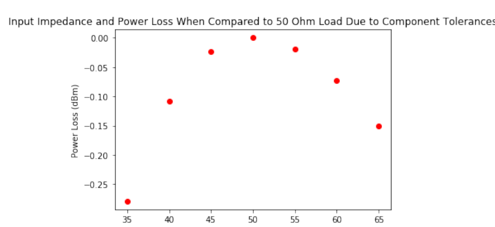

Figure 13: Losses due to Tolerances in a Matching Network

Case	Inductance (nH)	Capacitance (pF)	Impedance(Ω)	Mismatch Loss (dBm)
Lower Bound Worst Case	2.97	0.7239	45.24-j4.98	0.0227
Upper Bound Worst Case	3.63	0.594	55.2485 + j5.015	0.0206

Table 13: Worst Case Loss Analysis for Inductor and Capacitor Tolerances

4 Costs and Schedule

4.1 Costs

4.1.1 Labor

Assuming each of gets paid 30 dollars an hour, and we would on average each spend 27 hours a week. For there are 16 weeks this semester, then the labor of all three of us will be

$$30 \times 27 \times 16 \times 3 = \$38,880$$

4.1.2 Parts

Part	Count	Cost for Each
Raspberry Pi 3 B+	1	\$40.00
NRF24L01+	1	\$2.53
LT3665	2	\$4.00
RCL Components	50	\$0.005
SN75441NE	1	\$1.95
Motors	2	\$1.00
Robot Kit	1	\$15.00
IMU	1	\$2.50
Camera	1	\$21.99
PCB Orders	3	\$5.00
Total	N/A	\$109.22

Table 14: Motor Module Circuit

Table 14 illustrates the cost of each parts for a single robot. Each robot will cost \$109.22 to make; since we are making at least three, and ideally four to five. Our project parts in total will cost \$546.10 for five robots.

4.2 Schedule

	David
	John
	Louis

Task	Start Date	End Date	Sept W1	Oct W2	W3	W4	W5	W6	Nov W7	W8	W9	Dec W10	W11	W12
Design RF board	9/28	10/5												
Design Sensor Board	9/28	10/5												
Design Motor control board	9/28	10/5												
Send out initial board orders	10/6	10/11												
Test initial boards	10/12	10/16												
Order dev computer	10/3	10/8												
Acquire raspberry pis	10/3	10/8												
Acquire Motors	9/29	9/30												
Test motors	9/30	9/30												
Buy Development Board for NRF24L01+	10/3	10/3												
Work on Design Document	9/29	10/4												
Buy Raspberry Pi (x2)	10/3	10/3												
Identify, buy battery	10/3	10/3												
Research sensors	10/1	10/3												
Design Power Board	10/1	10/4												
Create Software Algorithms	10/1	10/24												
Test Development Board, algorithms	10/6	10/28												
Breadboard Motor Drive, test logic circuit	10/9	10/13												
Integrate Motors with PCB	10/3	10/10												
Test Battery	10/5	10/8												
Integrate battery power	10/8	10/9												
Send out second, revised board	10/5	10/22												
send out third, revised board	10/13	10/31												
Match RF networks	10/27	11/5												
Document RF metrics	10/28	11/6												
Document Power metrics	10/28	11/6												
Document sensor metrics	10/28	11/6												
Tolerance metric analysis	10/28	11/11												
Integrate RF and sensor module	11/2	11/8												
Integrate RF, sensor, and power module	11/5	11/13												
Integrate battery, RF, sensor and power	11/12	11/17												
Work on Final Demo	11/13	12/1												
Record Final Demo	11/17	12/4												
Final Presentation Preparation	11/27	12/9												
Final Project analysis	12/2	12/10												
Final Paper	12/5	12/13												
Submission, clean up, etc.	12/6	12/14												

Figure 14: Fall 2018 445 Gantt Chart

Figure 14 shows our proposed schedule for each task.

5 Ethics and Safety

Since we are using RF bands for communications, we are making sure we do not violate FCC regulations on the frequency and transmitter power. Because we are buying an MCU using a well-established Bluetooth technology specified in IEEE 802.15 [1], our RF usage is legal, safe, and will conform to the IEEE Code of Ethics Article 1. It is important to use radio bands that will not interfere with sensitive devices or with restricted bands, such as the aircraft communication band. Thus, using the RF band specified in IEEE 802.15 [1], and used in this project allows for the safety that is required in Article 1.

We are considering using NiMH nine-volt batteries to power our systems. NiMH batteries, according to Energizers guide [11], are cheap, made of environmentally friendly materials, only contain mild toxins, and are recyclable. These conditions are perfect for our use case. NiMH batteries are also fairly safe to use. However, one needs to watch out for overcharging and potential failed valves on the battery causing gas to build up inside rupturing the cell.

Additionally, this project has a variety of moving parts, which may cause a hazard to human health. In order to comply with the IEEE code of ethics part 9, "to avoid injuring others"[12], each module will have a software turn-off switch (in order to remotely shut down the robots), as well as a hardware switch readily available to power down the robot. This ensures that if the robots are found to be headed to an area where they could potentially cause harm, such as with pets and young children, or even sensitive equipment, that the robots can be shut down remotely or through manual shutdown.

Additionally, each robot will have a number of sensors that will collect data about the outside world. All data can be used in a way that is harmful to other people, and since a camera may be used for object detection and local position sensing, this camera may have data that could be used in a malicious manner. To conform to Article 9 of the IEEE Code of Ethics,"to avoid injuring others...[or] their reputation..."[12], a key detail is that the the information collected by these robots will not be stored in a long-term manner, that is, all video sensing data will not be stored except in temporary memory for object detection. All data that is collected will be clearly marked and indicated to all users so they realize the privacy constraints with these devices.

Finally, this project is intended to be used by young people and students in order to understand more about coordinated robotic systems. Because of this education based focus, there will be an emphasis on upholding the IEEE Code of Ethics Article 5, which seeks to "improve the understanding of individuals and society of the capabilities and societal implications of conventional and emerging technologies, including intelligent systems",[12]. In order to properly educate the users of these robots, a great deal of discussion in the final project and documentation of these robots will detail the science, engineering, and ethics of networks of robots.

References

[1] "IEEE 802.15 Wpan Document Archive," Web page, accessed September 2018. [Online]. Available: http://grouper.ieee.org/groups/802/15/pub/Download.html

[2] "SeaPerch ROV Competition," Web page, accessed September 2018. [Online]. Available: https://www.seaperch.org/index

[3] "Unmanned Systems Roadmap," Department of Defense, 2007. [Online]. Available: https://www.globalsecurity.org/intell/library/reports/2007/dod-unmanned-systems-roadmap_2007-2032.pdf

[4] "Robot Operating System," Web page, accessed October 2018. [Online]. Available: http://www.ros.org/

[5] J. L. Sobrinho, "Network routing with path vector protocols: Theory and applications," in *Proceedings of the 2003 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, ser. SIGCOMM '03. New York, NY, USA: ACM, 2003. [Online]. Available: http://doi.acm.org/10.1145/863955.863963 pp. 49–60.

[6] "Border gateway protocol," Wiki Page, Cisco Systems, Inc. [Online]. Available: http://docwiki.cisco.com/wiki/Border_Gateway_Protocol

[7] X. G and Z. Z, *Epipolar Geometry in Stereo, Motion, and Object Recognition*, Dordrecht, The Netherlands: Kluwer Academic Publishers, 1996.

[8] "nRF24L01+ Single Chip 2.4GHz Transceiver," Nordic Semiconductor, accessed September 2018. [Online]. Available: https://www.nordicsemi.com/eng/Products/2.4GHz-RF/nRF24L01

[9] "6v 250 ma brushed dc motor," Robot Shop, accessed September 2018. [Online]. Available: https://www.robotshop.com/media/files/pdf/datasheet-711.pdf

[10] S. J. Franke, *Wireless Communication Systems*, Champaign, IL, 2018.

[11] "Nickel Metal Hydride (nimh) Handbook and Application Manual," Web page, accessed September 2018. [Online]. Available: http://data.energizer.com/pdfs/nickelmetalhydride_appman.pdf

[12] "IEEE Code of Ethics," Webpage, IEEE, accessed September 2018. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html