Radio Mobile Foxbot

Brady Lagrone
Sophia Panagiotopoulos
Miguel Segura

Concept of Operations

Concept of Operations
for
Radio Mobile Foxbot

Team <51>

Approved by:

Project Leader Date

Prof. Kalafatis Date

T/A Date

Sophia Pana

Change Record

Rev.	Date	Originator	Approvals	Description
-	[date here]	[person's name]		Draft Release

Table of Contents

List of Figures

1. Executive Summary

Radio fox hunting now more commonly known as Amateur Radio Direction Finding (ARDF) has always involved a hunter, with a directional antenna and a radio receiver, and a fox, which is a hidden stationary transmitter. However, a stationary transmitter limits the potential of ARDF training. Rather than a person acting as a mobile fox transmitter, we aim to provide an autonomous system that will transmit while still mobile. The Mobile Radio Foxbot will address this issue with a mobile robot chassis which will detect its environment and develop a path to follow while transmitting. The Mobile Radio Foxbot will provide different modes of training, each with varying degrees of difficulty. We also plan to have direct communication with the Mobile Radio Foxbot so that we can locate it and change the type of training. This will drastically increase the potential of ARDF training.

2. Introduction

This document is an introduction to the Radio Mobile Foxbot, an autonomous training tool that facilitates and encourages the practice of radio communication and transmission. The “fox” will be able to orchestrate different paths and place itself in discrete locations while providing different schedules of radio transmission for added difficulty that the user can determine.

2.1. Background

There are many different interest groups of radios out there that have been developing “Fox Hunts.” One of these interest groups is our own Amateur Radio Club at A&M, the W5AC. The W5AC was founded in 1912 and is [1] “one of the oldest student organizations [at] Texas A&M.” Throughout the year there are many events and meetings that W5AC will host, but one of their specific events is their weekly “Fox Hunts.”

The mobile radio Foxbot ushers in a new generation of "Fox Hunts" for W5AC and other radio clubs. This procedural-leveled Foxbot will allow further practice with a brand-new system of automation. The autonomous pathing structure that the Foxbot will follow allows a more dynamic and engaging experience for Fox Hunts. This system is not only for W5AC but other Radio Clubs interested in a fully automated hunt. [2] However, the user may need an FCC License at the technician level to operate the Foxbot based on FCC regulations and standards.

Civilian, military, and commercial interests are all served by the Foxbot’s capabilities, which offer an adaptable platform for ARDF training. There can be active usage of the Foxbot within emergency situations to simulate a real-world search. Furthermore, military personnel could benefit from a field challenge by using the Foxbot as a new tool to develop ARDF skills and techniques.

The Foxbot would allow more practice for self-interested users who want to enhance their radio receiver skills/knowledge. Those who are interested can obtain an [3] FCC License to utilize and manage their own Foxbot.

2.2. Overview

The system will transmit radio signals in the 446.050 MHz frequency range using a Baofeng UV-5R Radio for internal communication. Scheduling and autonomous operations are managed by an ESP32 microcontroller, programmed with C and the ESP-IDF Framework. The MCU controls the motor drivers through a UART system, facilitating serial data exchange, and also interfaces with the radio to deliver transmission signals to the user. To power all of our components we will be using two LiPo batteries - one for each motor - along with an additional power source to power the MCU and the Baofeng UV-5R itself. The Foxbot will be able to autonomously move and provide a chosen transmission signal to facilitate practice/Fox Hunts.

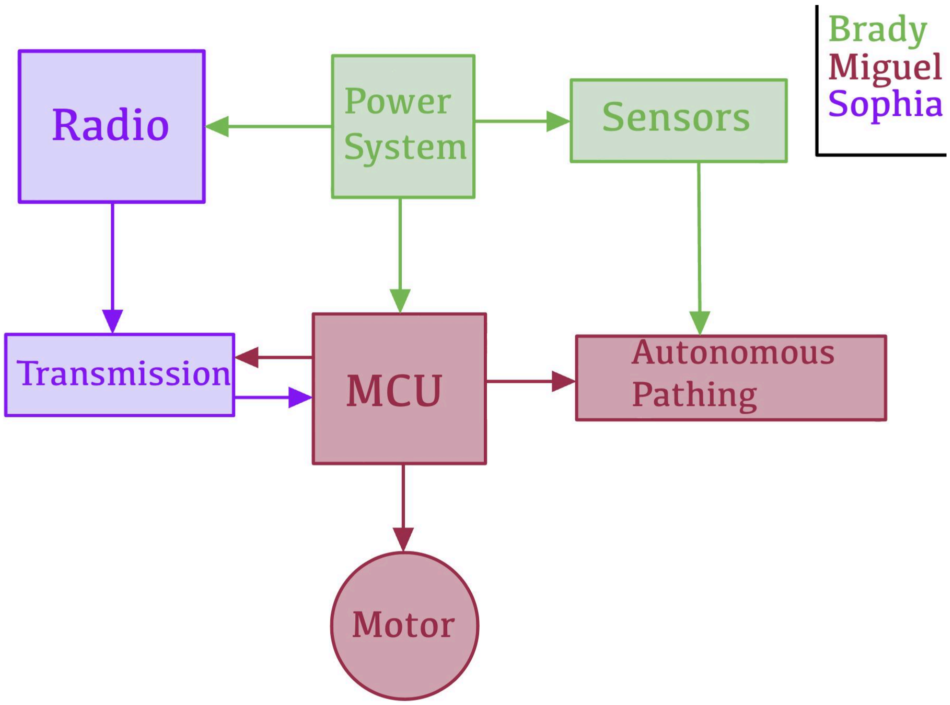


Figure 1: *Foxbot Description Block Diagram*

The block diagram above illustrates how the components of the subsystems interact with each other to complete the Foxbot's design and functionality. With a color correspondence of each individual working on their different subsystem.

2.3. Referenced Documents and Standards

[1] Texas A&M Amateur Radio Club (W5AC) - https://w5ac.tamu.edu
[2] FCC Standard Policies - https://www.fcc.gov/media/radio/public-and-broadcasting
[3] FCC License - https://www.fcc.gov/obtaining-license

3. Operating Concept

3.1. Scope

The radio Foxbot system is designed as a tool for amateur radio hunting and developers. This system will allow fox hunters to practice tracking transmissions and for developers working with transmission signals to better engineer their equipment. The Foxbot shall be able to avoid detection from a foxhunter.

3.2. Operational Description and Constraints

3.2.1 Operational Description

To operate the system, the user must turn on the bot in the desired starting location. The bot will formulate the route it will take based on the level of difficulty selected by the user. The way in which the user will select different modes will be determined by DTMF tones that will come from a repeater radio held by the operator. The repeater will send a tone which the radio on the foxbot will decode. The bot will begin moving in its path while simultaneously following transmission patterns. The FoxBot will operate until the duration of the hunt is over or until it has been caught by the HoundBot. Since this is a relatively new method of ARDF, the exact operation specifics will be determined based on the measured performance of the FoxBot . The necessary rules/guidelines will be formulated so that beginner to expert users can operate the system.

3.2.2 Operational Constraints

The bot will only be able to operate in a generally open environment with no stairs or sharp inclines. The ground must be generally easy to traverse as mud, rocks, and water will disturb the integrity of the electronics.

3.3. System Description

The Foxbot is divided into 3 main subsystems: Power, MCU/Autonomous Pathing, and Transmission.

Power Subsystem: Responsible for driving the motors, MCU, and radio systems. There will be batteries specific to the motor and another lipo battery system solely for the MCU and transmission system. Voltage regulators will be monitoring batteries to ensure the system is sustained at safe operating levels. In the event that voltage levels become unsafe, the power will shut off. Also responsible for providing power to all the sensors, and how the sensors will be connected to the other subsystems.

MCU/Autonomous Pathing Subsystem: Responsible for creating the basis on which the bot will formulate its path. Flight Controller and Mission planner will be used to program the operation that the foxbot will follow. Based on the location and difficulty of the operation, the MCU will control what and when the radio will transmit.

Transmission Subsystem: Responsible for emitting a signal for the foxhunter to track. Based on the location of our foxbot, the radio will be in either transmission or reception mode. When a signal needs to be emitted, the radio on the foxbot will listen for a DTMF signal from a repeater. This signal will be decoded and correspond to a certain pattern of signals to emit.

3.4. Modes of Operations

Training: The Foxbot generates waveforms for faulty line training and can be calibrated for technician training.

Deployed Modes: Includes continuous radio transmission with the bot being stationary, intermittent radio transmission with the bot being stationary, continuous transmission with the bot being mobile, and intermittent transmission with the bot being mobile.

3.5. Users

Users of this system must hold an FCC license at the technician level or higher to legally transmit signals.

One target user for the Foxbot is the user controlling the “fox”. This user will be responsible for configuring the signals that will be transmitted, the mode that the fox will be in (ie. training, sedentary, or active), and the starting location of the “fox”. A general understanding of radios and computer literacy is needed to operate the system.

Another user is the adversary trying to find the location of the Foxbot. They will be able to test their receivers and must have a good understanding of radio waves to operate the system effectively. The Foxbot system can also be used by antenna engineers trying to read certain frequencies under certain conditions.

3.6. Support

To aid the user, an instruction manual will be included with the bot. This will walk the user through the setup and maintenance of the bot. The code will be neatly and thoroughly commented on, so the user can follow the thought process of the program.

4. Scenario(s)

4.1. Fox Hunt

The Fox will move and hide in different locations for the User/Users to search. A starting location must be predetermined before the Fox Hunt can begin. After, the fox will move from space to space creating a more diverse and heightened difficulty Fox Hunt. The User who reaches the bot first will be crowned the winner. Only one person must have an FCC License to start the Fox for the Fox Hunt.

4.2. General Radio Transmission Search Practice

Users can place the Foxbot in a predetermined location with a scheduled radio transmission system for the Fox to operate. This setup allows the practice of learning how to receive transmission signals and how to properly locate the Fox.

5. Analysis

5.1. Summary of Proposed Improvements

- The system will have on board navigation so that it can move around autonomously, or we can have it follow a predetermined route.
- Different modes of operation will allow for different types of ARDF training, which can all be controlled remotely.
- System will be able to send status reports back to us.

5.2. Disadvantages and Limitations

- System will be battery operated, thus the duration of training could lead to system failure in the sense the battery could discharge and not provide enough voltage to operate.
- Doesn’t know when it has been caught
- Because communication with our system will be done via radio transmission, our system won't be able to provide a visual image.
- Conditions of the environment have to be viable, meaning our system will struggle or won't be able to go up steep or muddy pathways, and won't work in precipitation weather conditions.

5.3. Alternatives

- A person can act as the mobile transmitter
- The system could be powered via solar cells, which would allow it to train much longer
- This system could be mobile via air flight
- Controlled via RC controller

5.4. Impact

- Not hurting environment
- Popularize hunting robots

Radio Mobile Foxbot

Brady Lagrone
Sophia Panagiotopoulos
Miguel Segura

FUNCTIONAL SYSTEM REQUIREMENTS

Functional System Requirements

for

Radio Mobile Foxbot

Author Date

Approved by:

Project Leader Date

John Lusher, P.E. Date

T/A Date

Change Record

Rev.	Date	Originator	Approvals	Description
-	[date here]	[person’s name]		Draft Release

Table of Contents

List of Tables

List of Figures

1. Introduction

1.1. Purpose and Scope

ARDF (Amateur Radio Directional Finding) groups will be able to use the Foxbot as a practice tool to develop further their skills in radio finding. Through the autonomous design nature of the Foxbot, the scenarios at hand are left to the users' discretion. The Foxbot itself will provide the basis of its template for radio searching capabilities with its multi-tiered level system.

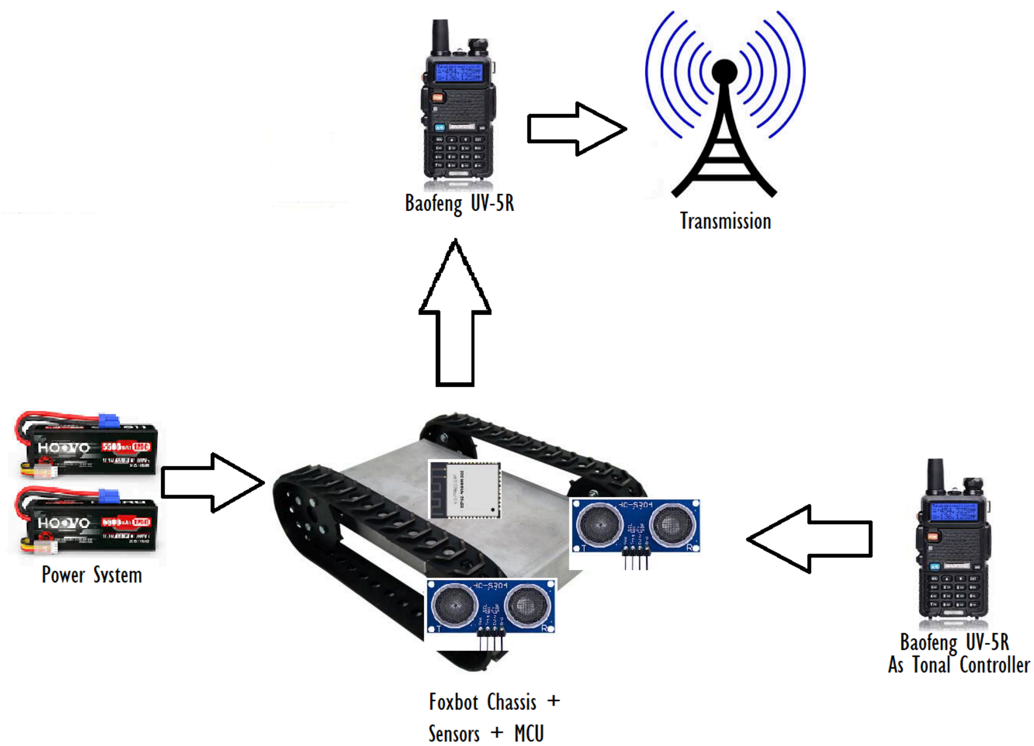


Figure 1. Your Project Conceptual Image

The following definitions differentiate between requirements and other statements.

Shall:	This is the only verb used for the binding requirements.
Should/May:	These verbs are used for stating non-mandatory goals.
Will:	This verb is used for stating facts or declaration of purpose.

1.2. Responsibility and Change Authority

The team leader, Miguel Segura, will be responsible for verifying all requirements of the project are met. These requirements can only be changed with the approval of the team leader and Professor Stavros Kalafatis.

Subsystem	Responsibility
Power Management/Sensors	Brady Lagrone
Radio Transmission	Sophia Panagiotopoulos
Autonomous Pathing/MCU	Miguel Segura

Table 1: Subsystem Leads

2. Applicable and Reference Documents

2.1. Applicable Documents

The following documents, of the exact issue and revision shown, form a part of this specification to the extent specified herein:

Document Number	Revision/Release Date	Document Title
C95.1	2019	IEEE Standard for Safety Levels with Respect to Human Exposure to Radio Frequency Electromagnetic Fields, 3 kHz to 300 GHz
ANSI C119.6	2011	American National Standard for Electric Connectors—Non-Sealed, Multiport Connector Systems Rated 600 V or Less for Aluminum and Copper Conductors
MIL-STD-1275	August 29, 2006	DEPARTMENT OF DEFENSE INTERFACE STANDARD CHARACTERISTICS OF 28 VOLT DC ELECTRICAL SYSTEMS IN MILITARY VEHICLES

Table 2: Applicable Documents

2.2. Reference Documents

The following documents are reference documents utilized in the development of this specification. These documents do not form a part of this specification and are not controlled by their reference herein.

Document Number	Revision/Release Date	Document Title
FCC-89-18	Jan 31, 1989	Amateur Radio Services, Amendment of the rules Part 97
IEC 62133-2:2017	Feb, 2017	Safety requirements for portable sealed secondary cells
IEC 60950-1	May, 2013	Information technology equipment - Safety

Table 3: Reference Documents

2.3. *Order of Precedence*

In the event of a conflict between the text of this specification and an applicable document cited herein, the text of this specification takes precedence without any exceptions.

All specifications, standards, exhibits, drawings or other documents that are invoked as “applicable” in this specification are incorporated as cited. All documents that are referred to within an applicable report are considered to be for guidance and information only, except ICDs that have their relevant documents considered to be incorporated as cited.

3. Requirements

In the following section, “Radio Mobile Foxbot” is referring to the entire system for which the proof of concept is being developed.

3.1. *System Definition*

The Radio Mobile Foxbot is an autonomous training tool that facilitates and encourages the practice of amateur foxhunting and radio communication. It allows users to select different modes of training, each with varying degrees of difficulty. The Radio Mobile Foxbot is composed of three main subsystems: Autonomous Pathing, Power, and Radio Transmission/Reception.

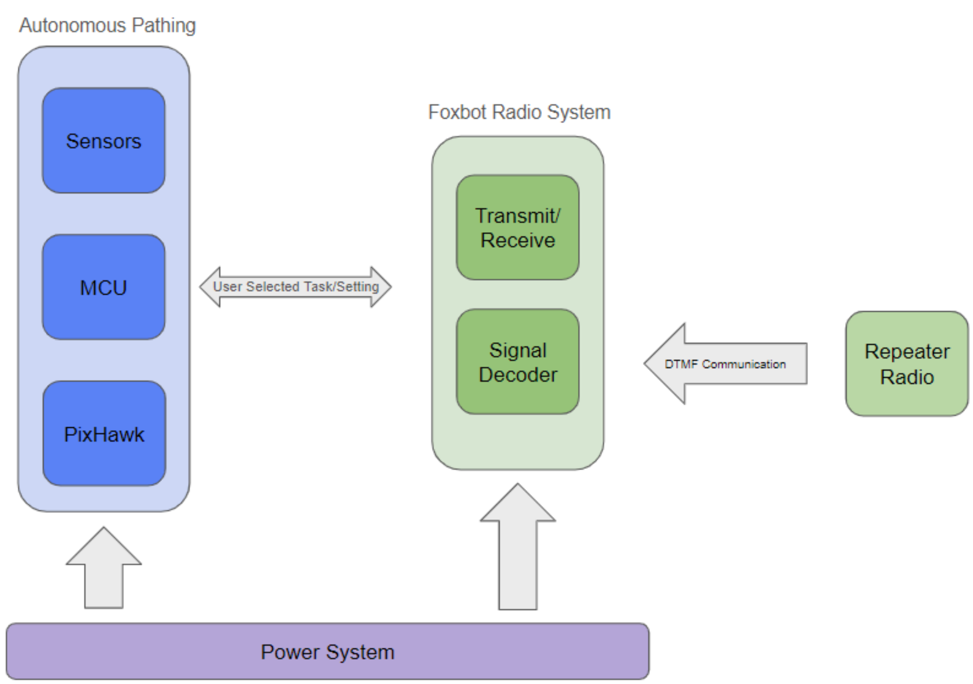


Figure 2. Block Diagram of System

The autonomous design of the Radio Mobile Foxbot will be determined by sensors, a Pixhawk, and MCU communication between one another. Pixhawk will be used with Mission

Planner to provide the GPS path that the Foxbot will take. This will then go to the MCU which will control the motors. The sensors will serve as object detection triggers, sending a signal to the MCU to reconfigure the path. A repeater radio will send DTMF signals to the radio on the Foxbot. These signals, in the form of frequencies, will be decoded from analog to digital and sent to the MCU. Based on the value of the digital signal, the Foxbot will transmit at a certain pattern and frequency. The user will be able to choose the setting that the Foxbot is in using these DTMF tones. The power subsystem will be in charge of regulating the voltage for the radio, motors, sensors, and MCU.

3.2. Characteristics

3.2.1. Functional / Performance Requirements

3.2.1.1. Transmission Frequency

The Radio Mobile Foxbot will have a transmission frequency of 446.050MHz and will be in the FM simplex mode.

> *Rationale: This combination is in accordance with the American Radio Relay League (ARRL) Amateur Radio Band Plan for 70 cm.*

3.2.1.2. Path Planning

The Radio Mobile Foxbot shall be able to avoid obstacles, modifying its path if one is sensed.

> *Rationale: The Foxbot is trying to avoid being caught by the Houndbox, so it needs to constantly be evaluating its surroundings in order to stay moving.*

3.2.1.3. Battery Operating Time

The Radio Mobile Foxbout shall be able to function on battery power for a period of 2 hours.

> *Rationale: Amateur radio foxhunting competitions last on average for 2 hours.*

3.2.1.4. DTMF Signaling

The repeater shall take no longer than 10 seconds to transmit the DTMF signal for the foxbot to decode.

> *Rationale: The purpose of the Foxbot is to transmit signals for directional radio systems to track. There must be a signal*

3.2.2. Physical Characteristics

3.2.2.1. Mass

The mass of the Radio Mobile Foxbot shall be less than or equal to 5 kilograms.

> *Rationale: This is so that the motor drives can efficiently move the foxbot at the desired speed to avoid capture.*

3.2.2.2. Antenna Size and Orientation

The radio antenna must be placed vertically on the Foxbot with the antenna itself being at least 148 mm.

Rationale: This is a requirement specified by our customer due to constraints of the Houndbot.

3.2.2.3. System Area

The system shall operate over any area as long as it is in range of the repeater for receiving DTMF tones.

Rationale: The foxbot must be able to receive signals from the repeater in order to know what functions to carry out. If the box goes too far from the user, no signals will be able to reach the system.

3.2.2.3. Inputs

The presence or absence of any combination of the input signals in accordance with ICD specifications applied in any sequence shall not damage the Foxbot System, reduce its life expectancy, or cause any malfunction, either when the unit is powered or when it is not.

Rationale: By design, should limit the chance of damage or malfunction by user/technician error.

3.2.2.3.1 Electrical Characteristics

3.2.2.3.2 Power Consumption

The maximum peak power of the system shall not exceed 620 watts.

Rationale: This is a requirement specified by our customer due to constraints of total time in which the Foxbot is operational.

3.2.2.3.3 Input Voltage Level

The input voltage level for the Foxbot shall be +9.5 VDC to +12.6 VDC for logistic control, and +18.3 VDC to +22.2 VDC

Rationale: Landcraft specification compatibility, MIL-STD-1275.

3.2.2.3.4 Vehicle Power Supply System

The Foxbot shall have DC voltage and current generating equipment, storage batteries, and distribution equipment normally fitted to the vehicle comprise the power supply system.

Rationale: Customers require fully autonomous design, capable of operating without direct power

3.2.2.4. Outputs

3.2.2.4.1 Data Output

The Foxbot shall include an interface compatible with the transmission system of the radio.

Rationale: User commands are understood based on the function selected by the user.

3.2.2.4.2 Diagnostic Output

The Foxbot shall include a diagnostic interface for control and data logging.

Rationale: Data will be converted between digital and analog forms, as well as going through different controllers, so having a system to track the data as it progresses through the system will be essential.

3.2.3. Environmental Requirements

The Search and Rescue System shall be designed to withstand and operate in the environments and laboratory tests specified in the following section.

Rationale: The foxbot will need to be tested in terrain similar to that found in field work.

3.2.3.1. Pressure (Altitude)

The Radio Mobile Foxbot shall be able to operate in altitudes ranging from sea level to 400m above sea level.

Rationale: Transmission can be better at slightly higher altitudes for longer transmission distances.

3.2.3.2. Thermal

The Radio Mobile Foxbot shall be able to operate in temperatures ranging from 5℃ and 40℃.

Rationale: The battery of the radio needs to be in moderate thermal conditions during peak transmission.

3.2.3.3. Rain

The Radio Mobile Foxbot shall operate under light rain conditions.

Rationale: Heavy rain will prevent the Foxbot from performing optimally.

3.2.3.4. Humidity

The Radio Mobile Foxbot may be able to function properly in an environment with humidity ranging from 0% to 100%.

3.2.4. Failure Propagation

The Radio Mobile Foxbot shall not allow propagation of faults beyond the Radio Mobile Foxbot interface.

3.2.5 Transmission Errors

The foxbot will determine if the DTMF signals fall in the range of the accepted frequencies. Incorrect frequencies/requests will not pass to the microcontroller and the user will have to change their tone. Transmissions received and output will stick to the specifications outlined by the FCC.

Rationale: This will maintain the integrity of the system.

3.2.4.1.1 Built In Test (BIT)

The Mobile Radio Foxbot shall have an internal subsystem that will test signals and evaluate if there is a failure.

3.2.4.1.1.1 BIT Critical Fault Detection

The BIT shall be able to detect a critical fault in the Foxbot system 90 percent of the time.

Rationale: This requirement would allow the foxbot to stop before possibly inducing damage to the system/user .

3.2.4.1.1.2 BIT False Alarms

The BIT shall have a false alarm rate of less than 5 percent.

Rationale: This is a requirement specified by our customer due to constraints of their system in which the Search and Rescue System is integrating.

3.2.4.1.1.3 BIT Log

The BIT shall save the results of each test to a log that shall be stored for retrieval and clearing by maintenance personnel.

Rationale: This would be helpful for the user to analyze the data and apply it to their needs.

4. Support Requirements

The Radio Mobile Foxbot needs to be located in areas without radio transmission interference for proper usage. Users must also provide a safe way to provide recharging to the Foxbot after usage as well as the external Baofeng UV-5R that is provided. Users need to tend to proper maintenance and clear out any debris that the Foxbot may encounter inside the Foxbot’s tracks.

Appendix A: Acronyms and Abbreviations

ARDF	Amateur Radio Directional Finding
BIT	Built-In Test
CCA	Circuit Card Assembly
EMC	Electromagnetic Compatibility
EMI	Electromagnetic Interference
EO/IR	Electro-optical Infrared
FOR	Field of Regard
FOV	Field of View
GPS	Global Positioning System
GUI	Graphical User Interface
Hz	Hertz
ICD	Interface Control Document
kHz	Kilohertz (1,000 Hz)
LCD	Liquid Crystal Display
LED	Light-emitting Diode
mA	Milliamp
MHz	Megahertz (1,000,000 Hz)
MTBF	Mean Time Between Failure
MTTR	Mean Time To Repair
mW	Milliwatt
PCB	Printed Circuit Board
RMS	Root Mean Square
TBD	To Be Determined
TTL	Transistor-Transistor Logic
USB	Universal Serial Bus
VME	VERSA-Module Europe

Appendix B: Definition of Terms

Radio Mobile Foxbot

Brady Lagrone
Sophia Panagiotopoulos
Miguel Segura

INTERFACE CONTROL DOCUMENT

Interface Control Document

FOR

Radio Mobile Foxbot

Author Date

Approved by:

Project Leader Date

John Lusher II, P.E. Date

T/A Date

Change Record

Rev.	Date	Originator	Approvals	Description
-	[date here]	[person's name]		Draft Release

Table of Contents

List of Tables

List of Figures

1. Overview

This document is provided to detail how the microcontroller, power system, and radio will interface. The ICD will include the physical descriptions of the Radio Mobile Foxbot this includes the electrical components, sensors, and other various inputs. This document will also explain how each of the subsystems will interface alongside each other to complete the objectives described in the ConOps and FSR documents.

2. References and Definitions

2.1. References

American National Standard for VME64 (ANSI/VITA 1-1994 (R2002))
4 Apr 1995

American National Standard for VME64 Extensions (ANSI/VITA 1.1-1997)
7 Oct 1998

2.2. Definitions

DTMF	Dual Tone Multi-Frequency
FCC	Federal Communications Commission
mA	Milliamp
mW	Milliwatt
MHz	Megahertz (1,000,000 Hz)
TBD	To Be Determined
RF	Radio Frequency
Ah	Amp-Hour
FFT	Fast Fourier Transform

3. Physical Interface

3.1. Weight

Component	Weight

Chassis	7 lbs
Motors	360g each (2x)
Lipo Batteries (Motor Control)	134g
Lipo Battery (Control Unit)	376g
Baofeng UV-5R	130g
Pixhawk	93.9g
GPS	36g
Ultrasonic Sensor	2.89oz each (2x)
Microcontrollers	0.348645oz each (2x)
Sabertooth Dual 12A Motor Driver	2.20oz
Miscellaneous	

Table 1: Radio Mobile Foxbot Weight Specifications

3.2. Dimensions

3.2.1. Dimensions of MCU Subsystem

Component	Length	Width	Height
Pixhawk	84.8mm	12.4mm	44.0mm
ESP32-WROOM	18.0mm	19.22mm	3.20mm
Sabertooth Dual 12A Motor Driver	64.0mm	75.0mm	16.0mm

Table 2: MCU Subsystem Dimension Specifications

3.2.2. Dimensions of Transmission Subsystem

Component	Length	Height	Width

Baofeng UV-5R	58mm	110mm	32mm
PCB	TBD	TBD	TBD

Table 3: Radio Subsystem Dimension Specifications

3.2.3. Dimensions of Power/Sensor Subsystem

Component	Length	Height	Width
Lipo Battery (Control Unit)	155mm	46mm	23mm
Lipo Batteries (Motor Control)	105mm	33mm	17mm
Ultrasonic Sensor	33.02mm	10.16mm	30.48mm
GPS	54mm (diameter)	N/A	14.5mm (thickness)
PCB	TBD	TBD	TBD

Table 4: Power/Sensor Dimension Subsystem

3.3. Mounting Locations

3.3.1 Placement of Foxbot

The Foxbot will need to be placed on a level piece of land determined by the user. There should be no stairs, water, ditches, or environmental factors that could interfere with the movement of the foxbot.

3.3.2 Mounting of Sensors

The sensors will be placed on the front two corners of the chassis to allow for optimum surveying of the environment. A component that is 3D printed will be required to mount the sensors to the chassis to protect it from the elements and for secure attachment. Screws will be utilized to connect the component to the chassis itself.

3.3.3 Mounting the Radio

The body of the radio will be mounted under the chassis and held in a 3D-printed box. Screws will be needed to securely attach the box component to the chassis. The antenna will be mounted on the top of the chassis in a vertical position for ample signal transmission. There will need to be some component holding the antenna up and protecting it from environmental factors.

3.3.4 Mounting the Control Unit

The control unit will be held under the chassis in a 3D-printed box. This box will be securely attached to the chassis via screws. The control unit will consist of the Pixhawk, ESP32, and GPS components.

3.3.5 Power Supply

The battery will be connected to the robot chassis via velcro. This velcro will allow for the battery to be secured, but can still be removed in case of recharge.

4. Thermal Interface

The microcontroller will be constantly running calculations to keep the Foxbot system going, leading to the possibility of it overheating. The radio could also experience heating issues due to the 2 hour period of the Foxhunt. Environmental factors could also impact the thermal effects of the system. To combat these, a heat sink will be implemented for the main MCU subsystem in order to keep the system operating efficiently.

5. Electrical Interface

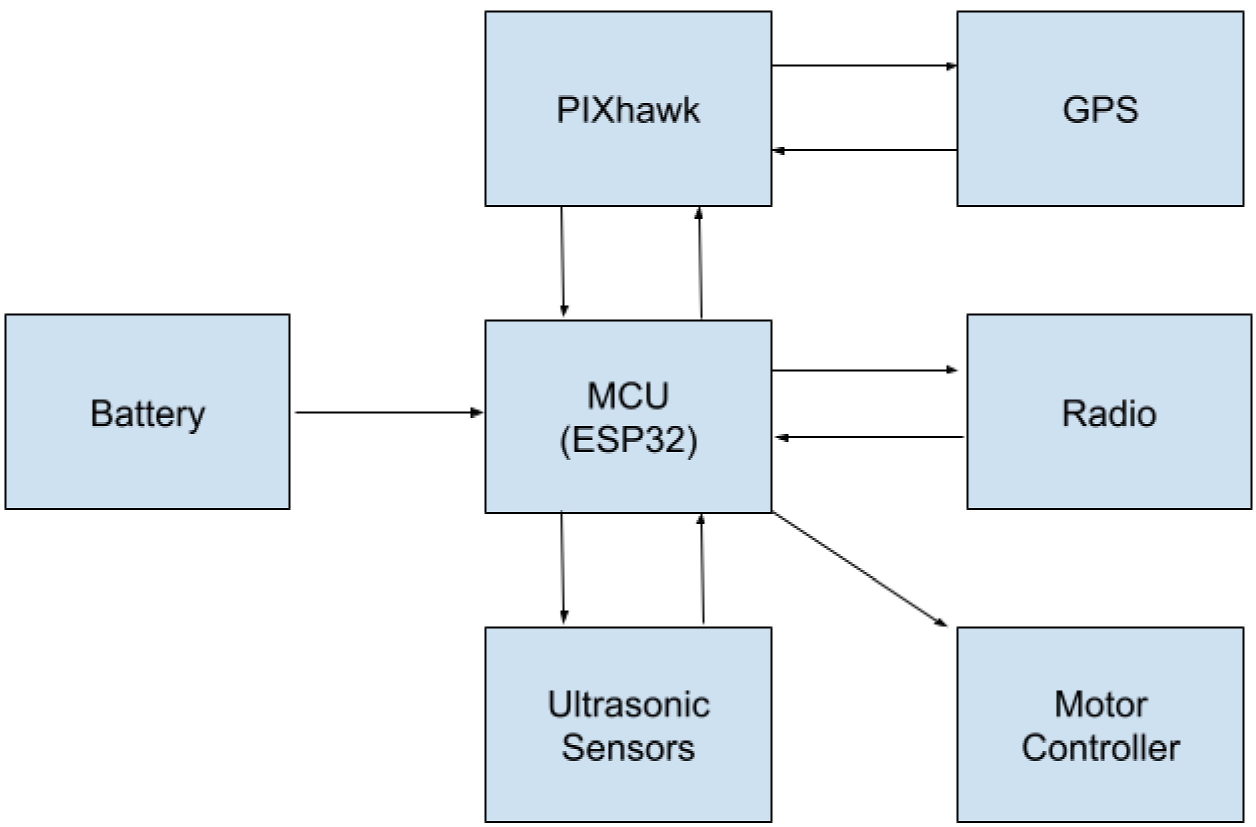


Figure 1: Electrical Interface Diagram

5.1. Primary Input Power

5.1.1. Control Unit Power

This system will contain all the components and their respective drop down voltage regulator, this system will be powered via the 11.1v 5500 mAh Lipo battery. This battery will have enough capacity to run the system for a minimum of 2 hours while the system is at peak consumption. The control unit will also track the voltage of said lipo battery.

5.1.2. Motor Control/ Motor Power

The SaberTooth motor controller and the two motors on the robot chassis will be powered via the two 7.4V 2200 mAh Lipo batteries. The control unit will monitor the voltage of these batteries.

5.2. Voltage and Current Levels

5.2.1. Control Unit Operational Values

Component	Voltage [V]	Current [mA]	Power[mW]
Baofeng Radio	7.4	1780	13172
ESP32-WROOM (Qty = 2)	3.3	160	528
HC-SR04 (Qty = 2)	5.0	30	150
M8N	5	150	7500
Radio Transmission System	TBD	TBD	TBD

Table 5: Control unit operational Voltage and Current Levels

The values of table 5 allow for accurate gauging of average power consumption. The values represented in this table allow us to estimate the total power consumption.

5.2.2. Motor Controller and Motors Operational Values

Component	Voltage [V]	Current [A]	Power[W]
Sabertooth Dual Motor Driver	24	24	576

IG42 24VDC 122 RPM Gear Motor (Qty = 2)	24	1	24

Table 6: Control unit operational Voltage and Current Levels

The values of table 6 allow for accurate gauging of average power consumption. The values represented in this table have come from SuperDroids website and allow us to estimate the total power consumption.

5.3. Signal Interfaces

5.3.1 Motors Interfacing

The MCU will communicate with the motors through the UART pins. Data will also be gathered from the Pixhawk as the Foxbot formulates its path. The motors will move based off of the Pixhawk and MCU data.

5.3.2 Sensor Interfacing

Data from the sensors will communicate with the MCU by reading in pulses to measure if objects are in the path of the Foxbot. The path of the Foxbot will be autonomously updated based on the signals received.

5.4. User Control Interface

The interface that will be used is a secondary Baofeng UV-5R utilizing the DTMF system in place. This radio will act as a remote controller and allow the user to interact with the system utilizing the numerical keypad. Each key will represent a different tone and allow level selection and more.

6. Communications / Device Interface Protocols

6.1. Wireless Communications (DTMF)

The transmission system will be controlled by DTMF signals from the repeater. These tones will be decoded by the microcontroller to produce transmissions selected by the user.

6.2. Device Peripheral Interface

The Sabertooth 12A Motor Driver is controlled through a serial port using UART and the ESP32 UART pins. Other devices that will be connected to the ESP32 UART pins will be the GPS and Pixhawk

Radio Mobile Foxbot

Brady Lagrone
Sophia Panagiotopoulos
Miguel Segura

Schedule

	9/9/2024	9/23/2024	9/30/2024	10/7/2024	10/14/2024	10/21/2024	10/28/2024	11/4/2024	11/11/2024	11/18/2024	11/25/2024	Date
Subsystem Research												
Conops Report									Not Started			9/15/24
FSR									In Progress			9/17/24
ICD									Complete			9/24/24
Validation Plan									Behind			9/26/24
Online schematic testing												9/30/24
Midterm Presentation												9/30/24
Order Power System Components												9/30/24
ESP32 Flashed with Foxbot												10/7/24
Foxbot is moving												10/1/24
Radio transmits tones												10/8/24
Subsystem Introduction Project												10/14/24
PCB design completed												10/14/24
PCB order												10/14/24
Foxbot path testing in environment												10/21/24
PixHawk Mission Planner												10/21/24
Battery Voltage Dividers												10/21/24
Project Update Presentation												10/23/24
Power to PixHawk and GPS												11/4/24
Power to ESP32 and HC-SR04												11/4/24
PCB interfacing with MCU												11/4/24
Final Presentation												11/11/24

Radio Mobile Foxbot

Brady Lagrone
Sophia Panagiotopoulos
Miguel Segura

VALIDATION PLAN

MCU/Autonomous Pathing Validation Plan:

Task	**Success Criteria**	**Result**	**Owner**
General MCU Research	Purchased MCU for Capstone Project	Complete	Miguel
ESP32 IDF Installation	Installation parameters working with VSCode	Complete	Miguel
Research ESP32 Code + Dummy Examples	Have a functioning UART design with Dummy Examples	Complete	Miguel
MCU Sub-System Introduction Project	Sub-system introduction is flashed to the dev kit. Both “Hello World” and “Blinky” are functional	Complete	Miguel
ESP32 Devkit Flashed	No Errors are caused when DevKit is flashed	Complete	Miguel
ESP32 connected serially (UART) with Sabertooth 12A	Motors are moving and functional	Complete	Miguel
PixHawk Connection with ESP	ESP32 Detects the Pixhawk	Not Tested	Miguel
Mission Planner Algorithm Connected with ESP + Pixhawk	Mission Planner can be interpreted with the PixHawk and ESP32	Not Tested	Miguel
Baofeng interface with ESP	ESP32 receives signals from Baofeng	Not Tested	Miguel
Baofeng signals are translated with ESP	Baofeng functions through the ADC and	Not Tested	Miguel

	DAC		
Autonomous Design Semi-Functional	ESP recognizes the path laid our and attempts to move properly	Not Tested	Miguel

Power System plus PixHawk Validation Plan:

Test Name	**Success Criteria**	**Result**	**Owner**
HC-SR04 Converter operating for 2	Buck converter successfully has a Vout = 5V, and Iout = 37.15mA	Not Tested	Brady Lagrone
ESP32 WROOM converter operating for 2	Buck converter successfully has a Vout = 3.3V and Iout = 200mA	Not Tested	Brady Lagrone
Baofeng radio converter operating	Buck converter successfully has a Vout = 7.4V and Iout = 1.78 A	Not Tested	Brady Lagrone
Radio Control PCB converter operating	Buck converter successfully has a Vout = 5V and Iout = 1 mA	Not Tested	Brady Lagrone
PixHawk and GPS converter operating	Buck converter successfully has a Vout = 5.5V and Iout = 375 mA	Not Tested	Brady Lagrone
PixHawk Navigation	PixHawk is successful with connecting to laptop and mission planner	Not Tested	Brady Lagrone
Battery monitoring	Voltage Divider for both main control unit and robot chassis divides battery to voltage < 3.6V	Not Tested	Brady Lagrone
Main control unit design	Main control unit has the correct pins connected to the right points on board	Not Tested	Brady Lagrone
Total power Consumption	Run system at max consumption for 2 hours	Not Tested	Brady Lagrone

Radio Validation Plan:

Test Name	Success Criteria	Result	Owner
Radio are functioning properly	Repeater and radio turn on and can transmit and receive signals	Complete	Sophia
DTMF tones programmed onto the radio	Using the programming wire and CHIRP, set the DTMF tones based on the bandpass filter frequencies. When keys are pressed, tones should be emitted from the repeater	Complete	Sophia
DTMF tones received by the radio	Can hear the tones through the ear piece. Means that the radio picked up the tones from the microphone and they are now being played out of the headphone jack.	Complete	Sophia
DTMF tones are recognized by the FFT chip	Tones are given as an input to the FFT chip (analog), and through a measuring device, the digital conversion can be displayed and read. Each analog tone will have a corresponding digital integer that should be read.	Complete	Sophia
MCU decoding	MCU will be able to read the digital signal and generate a transmission pattern based on the received values.	Not Tested	Sophia and Miguel
Transmission	Signal from the MCU is read by the	Not Tested	Sophia

Radio Mobile Foxbot

Brady Lagrone
Sophia Panagiotopoulos
Miguel Segura

Subsystem Reports

Subsystem Reports
for
Radio Mobile Foxbot

Author Date

Approved by:

Project Leader Date

John Lusher II, P.E. Date

T/A Date

Table of Contents

List of Tables

List of Figures

1. Introduction

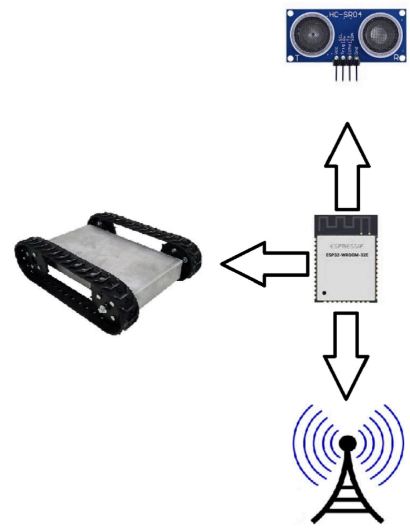
Our system is the Radio Mobile Foxbot, an autonomously designed robot that will be sending scheduled radio signals dependent on the user. The system will drive itself to different scheduled locations using Mission Planner and will then send a radio signal to be located by the user. This is a device made to practice ARDF (Amateur Radio Directional Finding) due to the current limited training tools/abilities. This system has 3 different components that will be further integrated together in 404. These consist of the microcontroller/sensor subsystem, the radio subsystem, and the power subsystem. Provided the proper validation and requirements needed for integration, our Radio Mobile Foxbot system is ready for full integration with all of our components.

2. Microcontroller and Sensor Subsystem Report: Miguel Segura

2.1. Subsystem Introduction

- This subsystem is being handled by Miguel Segura of Team 51, of the Radio Mobile Foxbot. In this subsystem there are two main components that are being used alongside the rest of the system. This includes the microcontroller and the ultrasonic sensors, both of which will be used to navigate the bot effectively.\

- The task at hand for each particular component (intended for ECEN 403 Development) is to develop the appropriate firmware to control the Foxbot utilizing a Sabertooth 12A Motor Driver. Readily using the aforementioned microcontroller and ultrasonic sensor.



- Further to be integrated is the connection between the MCU and the Radio Subsystem. The MCU will decode specified DTMF tone signals sent by the Radio subsystem to further control the Foxbot. This communication includes telling the Foxbot what level selection is being chosen, what scheduling system is being used, and more that is discussed in the Radio Subsystem introduction.

- The MCU will also be integrating with the Power Subsystem in terms of reading voltage/current values to provide proper values to the user. This will also be needed to navigate the motors on/off usage and to inform the user that the Foxbot's battery level is low. More information can be found in the discussion of the Power Subsystem introduction.

2.2. Microcontroller

2.2.1 Microcontroller and Specific IDE

- Utilized was an ESP32-WROOM-32E
- Programmed using the Espressif-IDE and Espressif IoT Development Framework

2.2.2 Functionality of MCU in Foxbot Design for 403 Demonstrations

- Connect ESP32 to Sabertooth 12A Motor Driver to control motors through a Simplified Serial connection with the assistance of UART

- Receive data from ultrasonic sensors to direct motors and the Foxbot
- Construct an autonomous path design that mimics a proposed design path made by Mission Planner

2.2.3 Microcontroller Operations

The ESP32 serves to control the motors and thus the Foxbot itself by creating an autonomous path that will mimic what Mission Planner will provide. The motors themselves are controlled using a simplified serial communication with the assistance of UART. Programmed onto the MCU was functionality that allowed UART to be used (for current and future integration), motor commands that directed the bot to move in different directions at given speeds, and configurations to allow the ultrasonic sensors to work through the RX and TX GPIO pins. Further firmware will be developed that will integrate the Radio Subsystem and Power Subsystem to the MCU subsystem accordingly.

2.2.4 Microcontroller with Motor Validation

The validation plans that go along with the microcontroller were testing the firmware to make sure that everything properly flashes onto the ESP32, the motor driver (Sabertooth 12A) connecting through UART, and showcasing motor speeds/angular turns that the Foxbot will deliver. The following two tables provided showcase the motors functioning at different speeds through a designated serial value that is being sent to the motor driver. Level 1 corresponds to a serial value of 15 (64 is the starting base to provide a forward function for the bot, so 64 + 15 provides the “Level 1” speed functionality), Level 2 is 30, Level 3 is 45, and finally Level 4 is 63 with the changed value being due to setting the motor driver to its maximum corresponding value of 127 (64 + 63).

Levels 1, 2, and 3 all functioned fairly consistently with the tests while Level 4 provided inconsistent distances travelled over set time spans. Our particular design also faced challenges concerning battery life which would reduce distances and overall speeds produced by the Foxbot.

In Table 2 are the time intervals that it takes for the Foxbot to perform angular turns. Typically set for either a 90° or 360° degree turn. For lower level motor speeds, the angular turning time equates to 4 times its 90° turn, however, for higher motor speeds there were inconsistencies and not simply 4 times their respective 90° turn.

Each of these validation plans were created by designing firmware using the ESP-IDF and programming specific modules to validate the design process. The largest concern of this validation process is generating connectivity values that demonstrate control over the Foxbot's system design.

Test/Level	Level 1	Level 2	Level 3	Level 4
Test 1	13.51 cm	33.02 cm	52.83 cm	68.58 cm
Test 2	13.34 cm	32.51 cm	52.07 cm	71.12 cm
Test 3	13.67 cm	33.02 cm	52.32 cm	63.50 cm
Test 4	13.34 cm	32.77 cm	52.07 cm	64.77 cm
AVG Speed	0.133 m/s	0.330 m/s	0.521 m/s	0.669 m/s

Table 1: Distance traveled and average motor speeds in m/s

Angle/ Level	Level 1	Level 2	Level 3	Level 4
90°	6850 ms	2675 ms	1800 ms	1125 ms
360°	27400 ms	10700 ms	6960 ms	4720 ms

Table 2: Time to rotate in milliseconds

2.3. Ultrasonic Sensor

2.3.1 Ultrasonic Sensor Specifications

- Utilized was two HC-SR04 ultrasonic sensors
- Validated for 1 meter range

2.3.2 Functionality of Ultrasonic Sensors in Foxbot Design

- Design firmware that allows connection between ESP32 and motors
- Connect to ESP32 and provide obstacle interference data to properly navigate the Foxbot to its desired location

2.3.3 Ultrasonic Sensor Operations

The ultrasonic sensors main purposes are to redirect and control the Foxbot during its designed path. They are designed to be working in a 1 meter diameter range and acquire distance away from obstacles/other path interferences. There will be two ultrasonic sensors placed onto the Foxbot that will be used to check if there is any object that the Foxbot may collide with. If an obstacle is detected, the ultrasonic sensor gathered information will allow the MCU to redirect the Foxbot accordingly.

2.3.4 Ultrasonic Sensor Validation

The ultrasonic sensors were connected to the ESP32 using its TX and RX GPIO pins (transmit pin and receive pin), which were then programmed to calculate the distance measured whenever the Trigger Pin is set to high (i.e. an obstacle is detected). Connectivity between the MCU and the ultrasonic sensors was validated by measuring distances between the Foxbot's front design and adjusting the object being detected randomly to secure appropriate validation values. Alongside these measurements, the ultrasonic sensors moved the Foxbot accordingly for demonstration purposes and provided appropriate feedback to the motors/motor driver.

```
I (296) main_task: Calling app_main()
This is the distance away from object for HCSO4 sample 1 - 0.891800
This is the distance away from object for HCSO4 sample 2 - 0.823200
This is the distance away from object for HCSO4 sample 3 - 0.840350
This is the distance away from object for HCSO4 sample 4 - 0.840350
This is the distance away from object for HCSO4 sample 5 - 0.806050
I (10346) main_task: Returned from app_main()
```

Figure 1: Ultrasonic Tests in meters

2.4. Subsystem Conclusion

The microcontroller and sensor system worked together after developing the appropriate firmware to connect everything together. Most of the development was done using the Espressif-IDE with some aid of the provided example templates to learn how to develop code for things like the ESP32. Many of the faults that appear to occur in the system are related to the battery depletion due to the high current drawn from the motors from the lipo batteries. This leads to some inconsistencies with our measurements, but should be resolved whenever the Power Subsystem is properly integrated with the MCU onboard.

However, everything functionally works together (i.e. the MCU, motor driver, ultrasonic sensors) and can perform the necessary functions that were needed for 403 development. The largest issue at the very beginning was properly developing firmware that could operate the bot with the ESP32, but after this was completed developing further firmware was made much easier.

Test Name	Success Criteria	Methodology	Status
Pathing Accuracy	The path will stay in range and be autonomous	Place the Foxbot in a location and it should be able to determine its path while staying in a preprogrammed range	Passed
Sensor Sensitivity	Obstacles will be avoided including trees, ditches, and moving objects	The Foxbot will be able to reroute its path if any obstacles are sensed by the sensors	Passed

Table 3: Validation Results

3. Radio Subsystem Report: Sophia Panagiotopoulos

3.1 Subsystem Introduction

The radio subsystem is designed to provide a means of communication between the foxbot and the user. The user will be able to control the transmission of foxbot radio using DTMF signals, which will be decoded and sent to the MCU for further processing. The radio subsystem was tested to confirm that the radios were interfaced, DTMF signals were being sent, the signals were properly decoded, and that transmission from the foxbot radio of a single frequency was achievable.

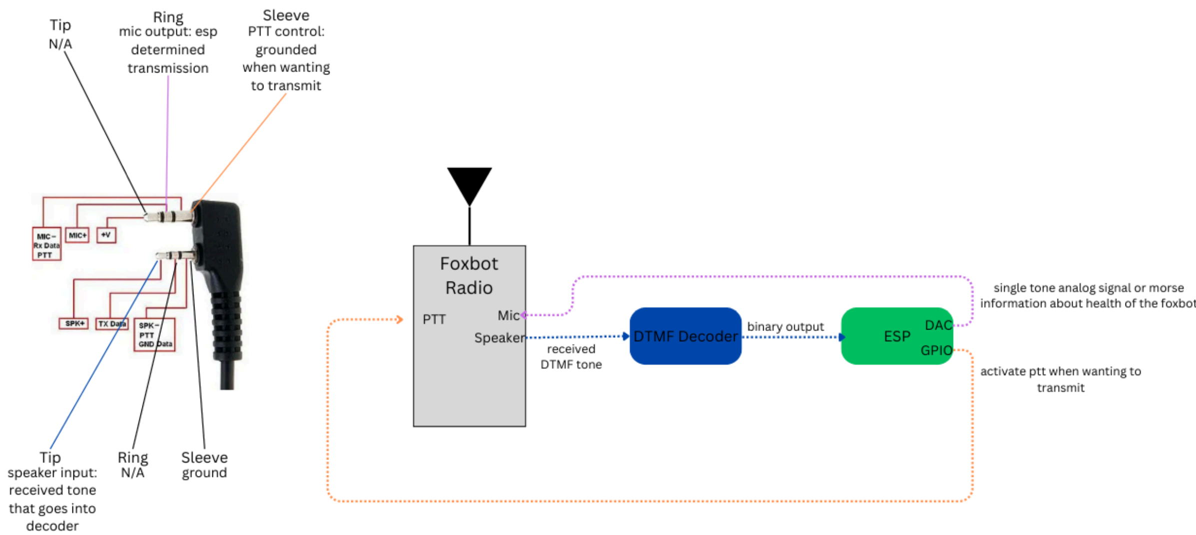


Figure 2: Diagram of the Radio Subsystem

3.2 DTMF Decoder

3.2.1 Operation

A DTMF signal is based on a matrix of audio frequencies where two frequencies are added to correspond to each button on a phone keypad. The Baofeng UV-5R contains a speaker and microphone TRS audio jack which can be used to extract and input signals to and from the radio. For user control of transmission, a DTMF signal can be sent from the user radio to the foxbot radio where the signal will be pulled from the 'tip' of the speaker audio jack and be sent into the decoder. Based on the binary output of the decoder, the MCU will be able to formulate what kind of transmission the user is requesting from the foxbot radio, whether that be change of transmission pattern or foxbot health updates.

3.2.2 Altium Design

For the radio subsystem, a PCB needed to be created to decode the received DTMF signal. The input of the decoder (J1) is coming from the speaker jack of the foxbot radio and the output of the decoder (J2) will be an input for the MCU.

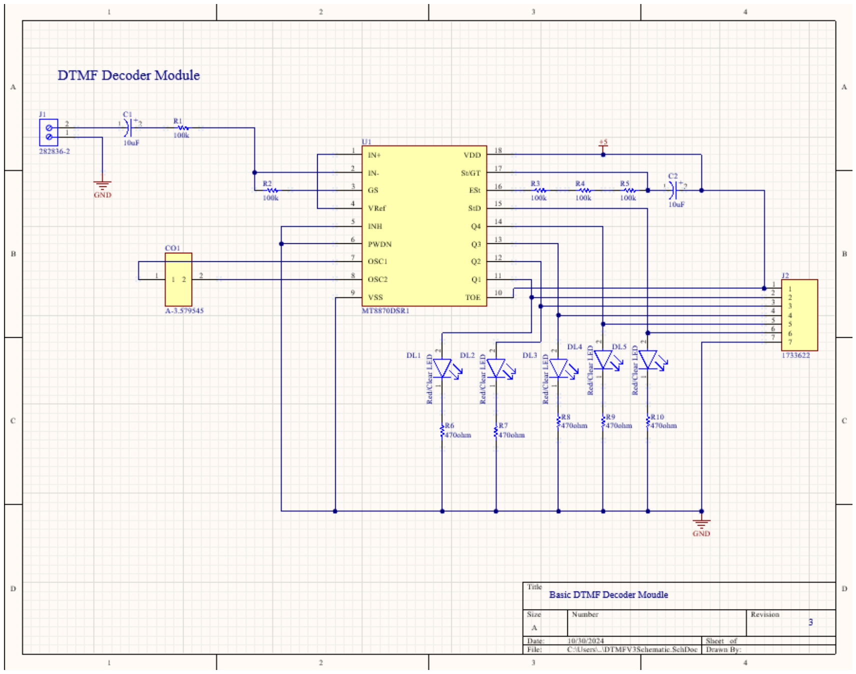


Figure 4: DTMF Decoder Schematic Design

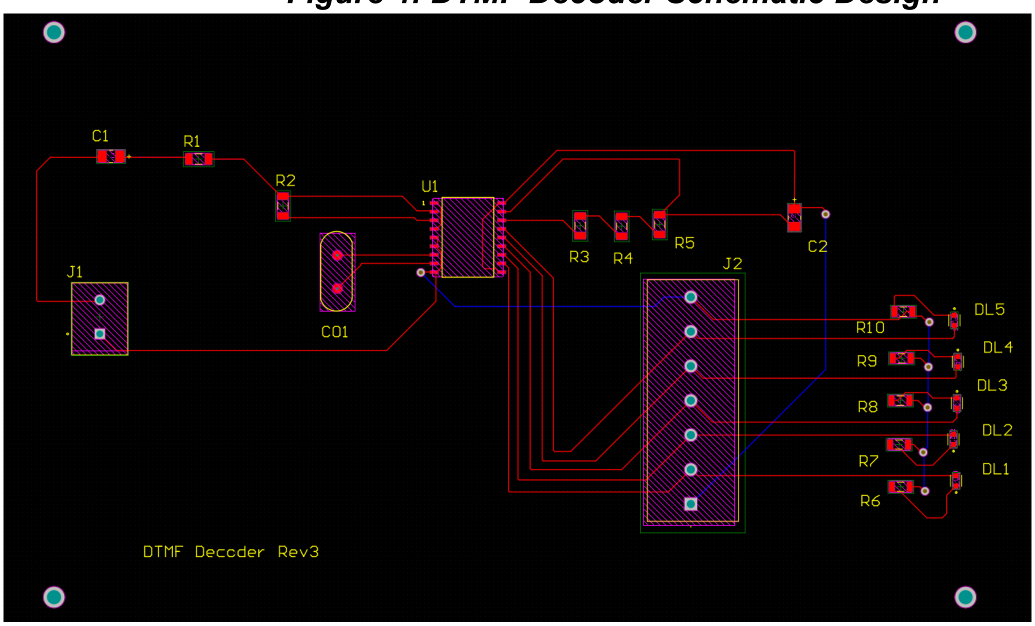


Figure 5: DTMF Decoder PCB Layout

3.2.3 Validation

The radios were programmed to receive and send signals between one another by putting them on the same frequency band with the appropriate offset. To validate radio interfacing, a signal from one radio was sent and heard on the other radio. This confirmed that if the user were to send a DTMF tone, the foxbot radio would be able to pick up the signal.

In order for the decoder to function properly, a certain signal of the right frequencies must be recognized as a DTMF tone. The user radio was tested for accurate DTMF signal transmission using an oscilloscope. Using the fast fourier transform (FFT) function, the received signal could be validated as a DTMF tone by analyzing the frequencies comprising the tone . Each number on the keypad was sent from the user radio and pulled out of the speaker jack of the foxbot radio. The peaks of the FFT were validated against the standardized DTMF signal matrix to ensure proper signals were being sent.

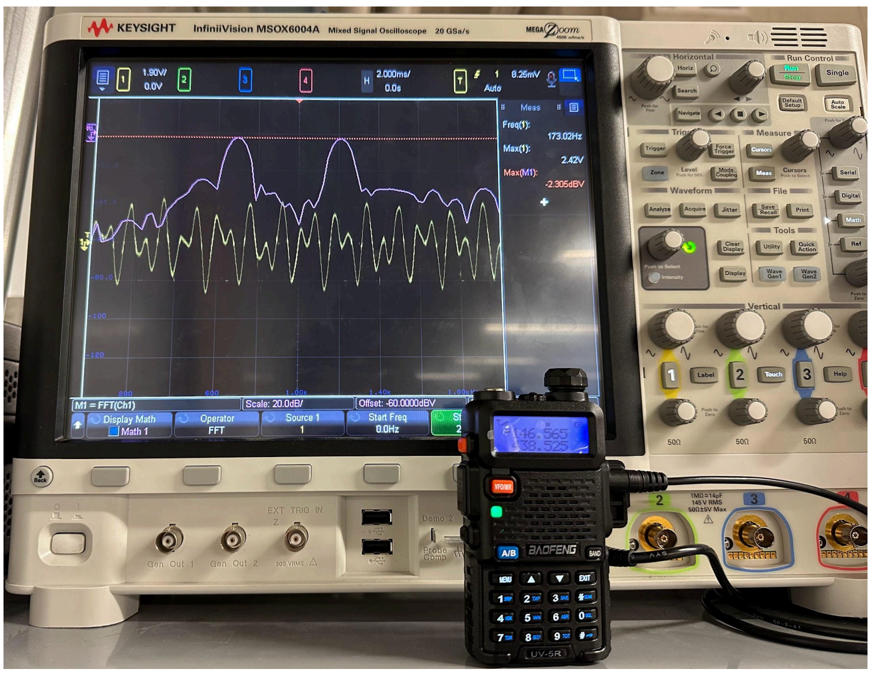


Figure 6: Oscilloscope reading the DTMF tone received by the foxbot radio.

In *Figure 6* the FFT shows the frequency response of the signal received by the transmitting radio. The tone sent was a 1 which corresponds to 697 Hz and 1209 Hz, validated by the two peaks in purple. An FFT was produced for each of the sent DTMF tones to ensure that all numbers on the keypad would be recognized by the decoder. The yellow signal is the sinusoidal wave showing the addition of the two frequencies. The FFT shows that the only accepted frequencies are those comprising the DTMF signal.

DTMF Tone Validation		
Keypad Number	Frequencies (Hz)	Received Tone Accuracy
1	1209 + 697	Accurate
2	1336 + 697	Accurate
3	1477 + 697	Accurate
4	1209 + 770	Accurate
5	1336 + 770	Accurate
6	1477 + 770	Accurate
7	1209 + 852	Accurate
8	1336 + 852	Accurate
9	1477 + 852	Accurate
0	1336 + 941	Accurate

Table 4: DTMF Tone Validation Using FFT

Once it was validated that the radio was sending accurate DTMF signals, the decoder module was ready to be tested. A 2.5 mm TRS male audio jack was plugged into the corresponding female port on the receiving radio. The wires associated with the 'tip' and 'ground' of the signal were the inputs of the decoder and attached to the PCB (the left two wires in *Figure 5*). 5 LEDs are present on the board, 4 corresponding to the binary output (DL1-DL4) and 1 indicating the latch of the IC has been updated (DL5). The number pressed on the keypad, of the user controlled radio sending the signal, will be output as a binary value from the decoder. Each tone was sent by the user radio and decoded by the foxbot radio.

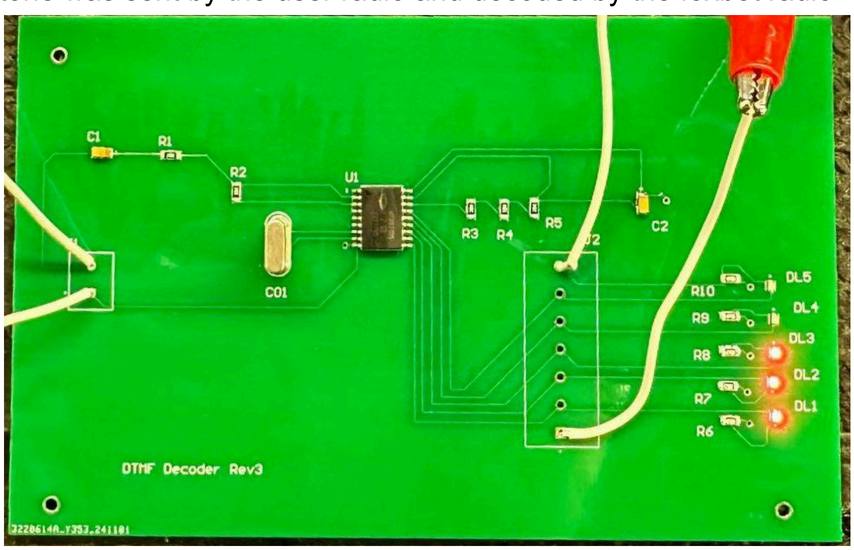


Figure 7: Decoder Output of Received Signal

In *Figure 7* the output LEDs are showing a binary output of 7 (with the DL1 being the least significant bit) when a 7 was pressed on the keypad of the user radio to send the DTMF tone. DL5 indicates when the tone has been received and turns on when the latch has updated. In *Figure 7* DL5 is not on because the steering voltage has fallen below the threshold indicating that the latch is ready to be updated by a new received tone pair. DL5 is important for designing the timing parameters of the decoder. The decoder will only update the output when DL5 is no longer on, so the timing of sending signals to the foxbot radio will depend on the length of time it takes the steering voltage to fall below the threshold. Each tone pair was sent from the user radio to the foxbot radio to ensure that all numbers could be decoded. *Table 5* shows the results from those tests.

DTMF Decoder Validation		
Keypad Number Sent	Expected Binary Output	Decoder Binary Output
1	1	1
2	2	2
3	3	3
4	4	4
5	5	5
6	6	6
7	7	7
8	8	8
9	9	9
0	10	10

Table 5: DTMF Decoder Validation

The decoder was functioning as intended and from this a DTMF library can be formulated, allowing the user to fully customize the intended output.

The range at which the decoder was working did not seem to align with the specifications needed for the project, however. The radios needed to be close together and at the correct angle in order for the decoder to read the signal. In order to rule out that it was a signal strength issue, the receiving radio was hooked up to the oscilloscope and an FFT was produced to validate that the radio was indeed receiving the correct signal even at varying distances. The user radio sent DTMF tones to the foxbot radio at different locations around zach where the FFT still showed the correct tone pair. Once the signal could be ruled out as a possible problem, it then became a PCB debugging issue. The ratings of the capacitors and resistors were tested to ensure they read the correct values. This would eliminate possible signal cutoff within the decoder as the combinations of the resistors and capacitors are used to filter out unwanted frequencies not recognized as the two tone pair. The voltage of each pin of the IC was measured to validate that each pin was performing at the correct rating, eliminating that there was a malfunction with the IC. DL5 can be used to see when the latch of the IC is high or low, indicating when the decoder is ready to receive a new signal. When the decoder first receives power, this LED goes high meaning that the steering voltage has gone above the threshold and the latch has been updated. Once fallen, the decoder is ready to be tested. When the decoder was not reading the signals being sent, the latch LED still went high when it received power meaning that the IC was still functioning as intended. Ultimately, the range of the decoder is still an ongoing issue that will be worked on immediately in 404.

3.3 Transmission

3.3.1 Operation

Based on the decoded signal, the MCU will formulate the signal to be transmitted from the foxbot radio. The DAC pin of the MCU will interface with the radio through the 'ring' section of the microphone audio jack. When PTT on the foxbot radio is enabled, the signal can be transmitted and received by the user radio.

3.3.2 Validation

The radios were already validated to be interfaced for communication between one another. Pulling a signal from the MCU is a task to be completed in 404 when subsystem interfacing begins, so for transmission validation a frequency on the audio scale was generated from an app and sent into the microphone port of the radio using a 3.5 mm TRS jack. When PTT was pressed, the audio signal could be heard from the user radio indicating that the foxbot radio can send a signal to be picked up by the user radio.

3.4 Subsystem Conclusion

Each part of the radio subsystem was shown to behave properly. The radios are able to send and receive signals between one another and the decoder is able to accurately decode the different DTMF tones that will be sent between the radios. The range of transmission will be resolved within the first few weeks of integration so that whole system testing can begin. When interfaced with the other subsystems, it will allow the foxbot to be user controlled through radio communication.

Once the subsystems are integrated sufficiently, an additional feature of morse transmission and decoding will be implemented between the user and foxbot radios. Morse is a tool that can be used to get battery and location updates from the foxbot, allowing the user to gain additional data about the health of the foxbot. The morse code will be generated from the MCU and will be transmitted similarly to the single tone audio frequency mentioned previously. A morse decoder will be used to determine the information that the foxbot radio is trying to relay. A certain DTMF signal will be sent that will be recognized as the user seeking updates on the bot.

4. Power Subsystem Report: Brady Lagrone

4.1 Subsystem Introduction

The main point of the power subsystem is to provide power to each of the components on the system, as all of the components will be powered via a battery. Another purpose of this subsystem is to monitor each of the batteries on the Foxbot, this includes the two 3S lipo batteries on the rover chassis and then also the 3S battery that will be powering the components on the Foxbot. When monitoring the batteries the system is responsible for gathering data about the batteries and returning the data to the ESP32WROOM. This subsystem is also responsible for connecting the components together on the PCB board, for example the ESP32 ports where they need to be connected. And finally this subsystem is responsible for the PixHawk, which is going to act as the autopilot on the FoxBot, and develop a path that the robot can take.

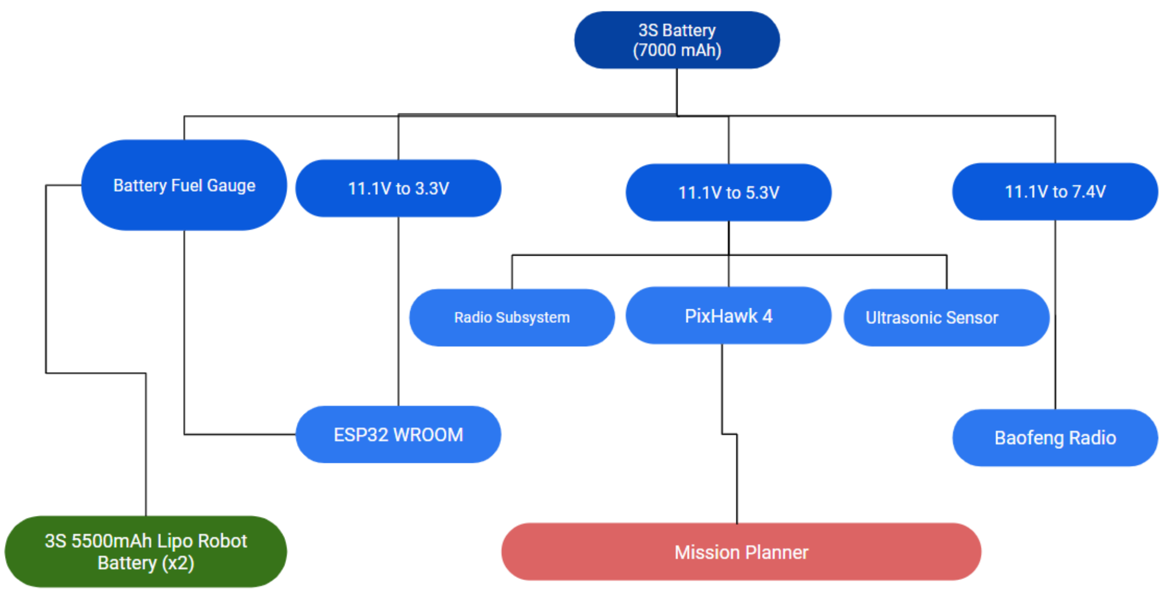


Figure 8: Power Subsystem Break Down

4.2 DC to DC Converters

Component	Operating Voltage	Min Voltage	Max Voltage	Operating Current(mA)	Current designed for (+25%)	Power(W)	Total Power(W)
ESP32	3.3	3	3.6	500	625	1.65	20.126
ESP32	3.3	3	3.6	500	625	1.65	
HC-SR04 (Ultrasonic Sensor)	5		5.5	15	18.75	0.075	
HC-SR04 (Ultrasonic Sensor)	5		5.5	15	18.75	0.075	
M8N (GPS)	5	2.7		150	187.5	0.75	
Pixhawk PX4 Flight Controller	5.3			150	187.5	2.75	
Baofeng(Radio)	7.4			1780	2225	13.172	
MT8870DSR1(Radio transmission PCB)	5	2.7	5.5	0.8	1	0.004	

Table 6: Operating Conditions of Each of the Components

Table 6 describes a great importance of this subsystem as the components that are being used have varying operating voltages and currents. For example the ESP32 requires 3.3V to operate and 500mA, it is important to note as seen in the Current Designed for (+25%) column that this was designed with an additional 25% current draw available. This is to provide appropriate currents to the components even when there is a greater load. Next, this also provides how much capacity our battery will need, because a requirement of the system is that it has to operate for 2 hours. Another note is the Foxbot will be powered via a battery and thus it will discharge over time and not provide a constant voltage. The 3S Lipo battery will range from 9.6V at fully discharged to 11.1V nominal to 12.6V at fully charged.

4.2.1 11.1 to 3.3V DCDC Validation

The 3.3V DCDC buck converter will be responsible for providing 3.3 Volts to both of the ESP32 ROOMS, which require 625 mAmps to operate.

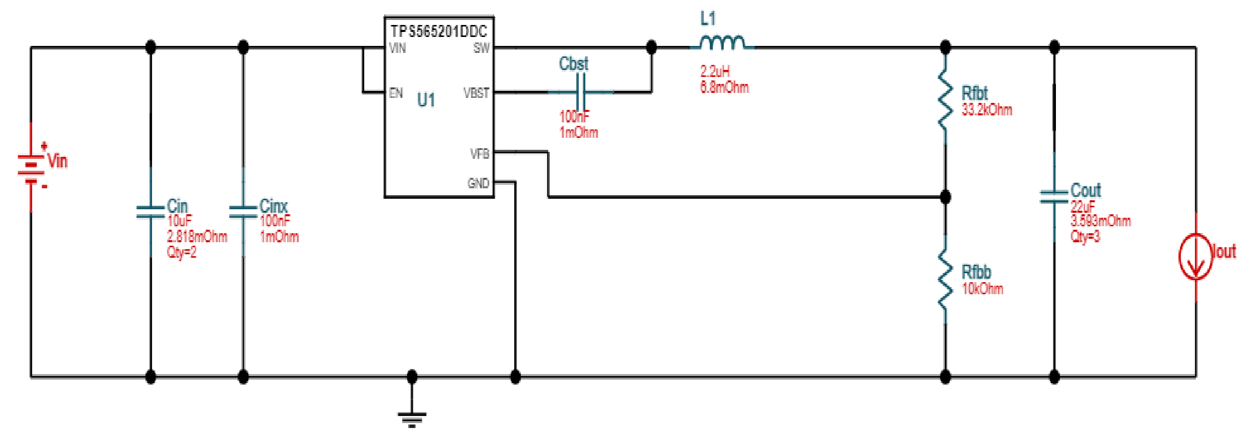


Figure 9: Schematic of 11.1V to 3.3V Buck Converter

Changing input (300mA)		
Input Voltage	Output	Line Regulation
12.6	3.274	0
12.3	3.273	0.03054367746
12	3.273	0.03054367746
11.7	3.273	0.03054367746
11.4	3.273	0.03054367746
11.1	3.272	0.06108735492
10.8	3.272	0.06108735492
10.5	3.272	0.06108735492
10.2	3.272	0.06108735492
9.9	3.271	0.09163103238
9.6	3.271	0.09163103238

Table 7: Line Regulation Test 3.3V Buck Converter

Table 7, was completed using a DC load that was drawing 300mA then the input voltage was changed from 12.6V to 9.6V, and output was collected. The change is seen in the column labeled Line Regulation. The converter is operating as expected.

Current Output		
Current	Voltage	Load Regulation(%)
0	3.301	0
10mA	3.288	0.3938200545
20mA	3.286	0.4544077552
50mA	3.283	0.5452893063
100mA	3.279	0.6664647077
200mA	3.273	0.8482278098
400mA	3.266	1.060284762
600mA	3.254	1.423810966
800mA	3.239	1.878218722
1A	3.227	2.241744926
1.1A	3.224	2.332626477
1.2A	3.216	2.57497728

Table 8: Load Regulation Test 3.3V Buck Converter

Table 8, was completed using a DC load that was drawing from 0A and all the way up to 1.2A while the input voltage was set to 11.1V. This is supposed to represent the maximum load that the converter should receive. The change is seen in the column labeled Load Regulation. The converter is operating as expected.

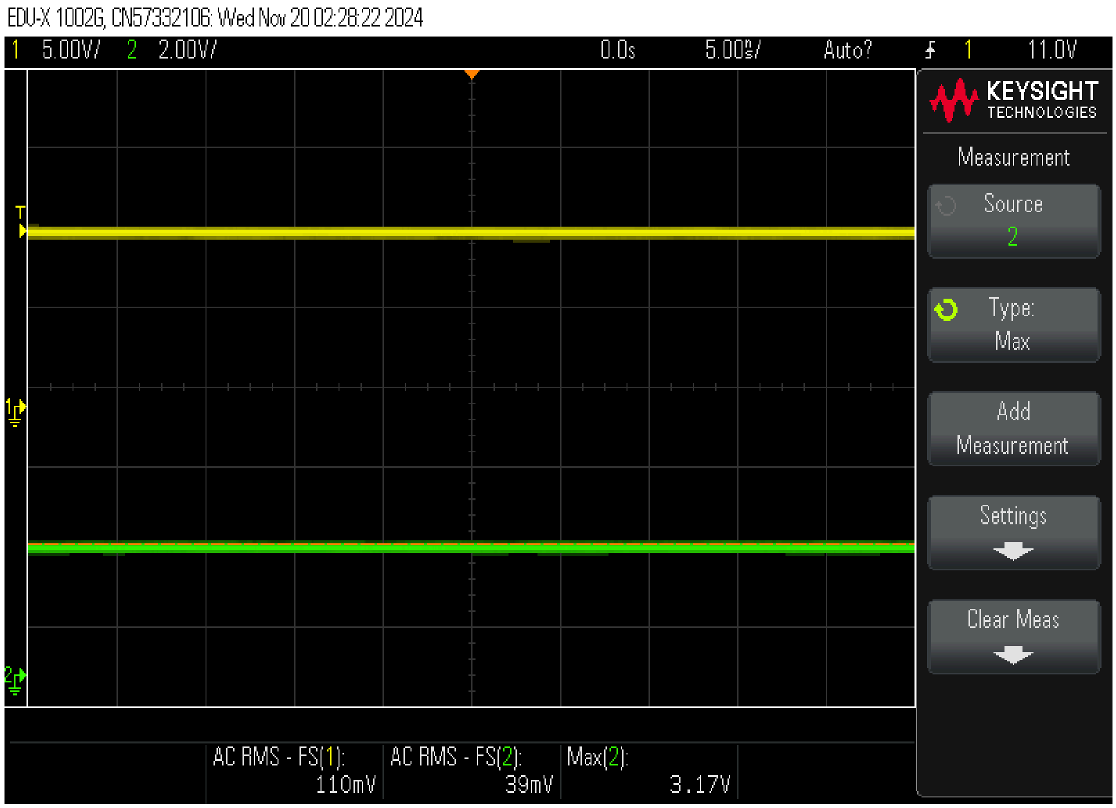


Figure 10: Scope of 3.3V Converter Noise

Figure 10 provides a measurement of the noise that is being inputted and outputted of the 3.3V Buck Converter. The yellow line represents the voltage input at 11.1V which has a noise of 110mV, while the green line represents the voltage output of the converter, measured with a noise of 39mV. Thus the converter is operating correctly in this aspect.

4.2.2 11.1 to 7.4V DCDC Validation

The 7.4V DCDC buck converter will be responsible for providing 7.4 Volts to the Baofeng radio for transmission and receiving and will have to provide 2.225A to operate.

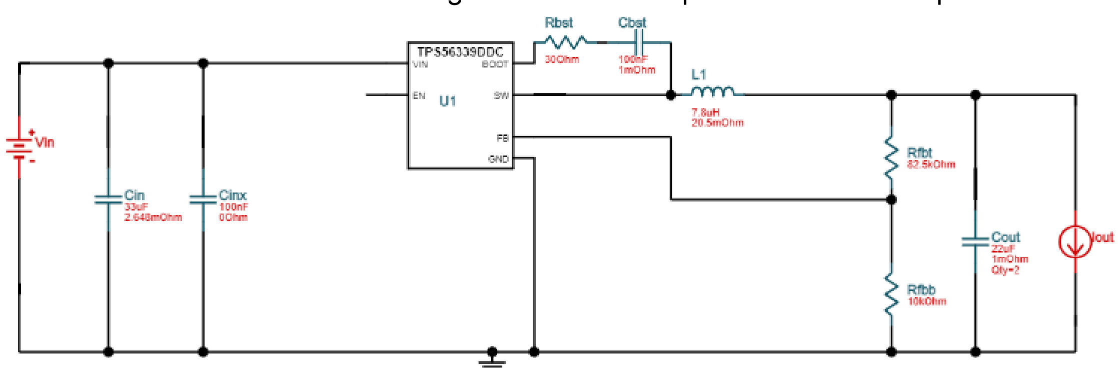


Figure 11: Schematic of 11.1V to 7.4V Buck Converter

Changing input(1 A)		
Input Voltage	Output	Line Regulation
12.6	7.324	0
12.3	7.325	0.01365374113
12	7.326	0.02730748225
11.7	7.327	0.04096122338
11.4	7.327	0.04096122338
11.1	7.328	0.0546149645
10.8	7.33	0.08192244675
10.5	7.331	0.09557618788
10.2	7.332	0.109229929
9.9	7.333	0.1228836701
9.6	7.336	0.1638448935

Table 9: Line Regulation Test 7.4V Buck Converter

Table 9, was completed using a DC load that was drawing 1A then the input voltage was changed from 12.6V to 9.6V, and output was collected. The change is seen in the column labeled Line Regulation. The converter is operating as expected.

Current Output		
Current	Voltage	Load Regulation(%)
0	7.434	0
200mA	7.344	1.210653753
400mA	7.367	0.9012644606
600 mA	7.357	1.035781544
800 mA	7.343	1.224105461
1 A	7.329	1.412429379
1.2 A	7.314	1.614205004
1.4 A	7.32	1.533494754
1.6 A	7.31	1.668011838
1.8 A	7.282	2.044659672
2 A	7.25	2.47511434
2.2 A	7.25	2.47511434

Table 10: Load Regulation Test 7.4V Buck Converter

Table 10, was completed using a DC load that was drawing from 0A and all the way up to 2.2A while the input voltage was set to 11.1V. This is supposed to represent the maximum load that the converter should receive. The change is seen in the column labeled Load Regulation. The converter is operating as expected.

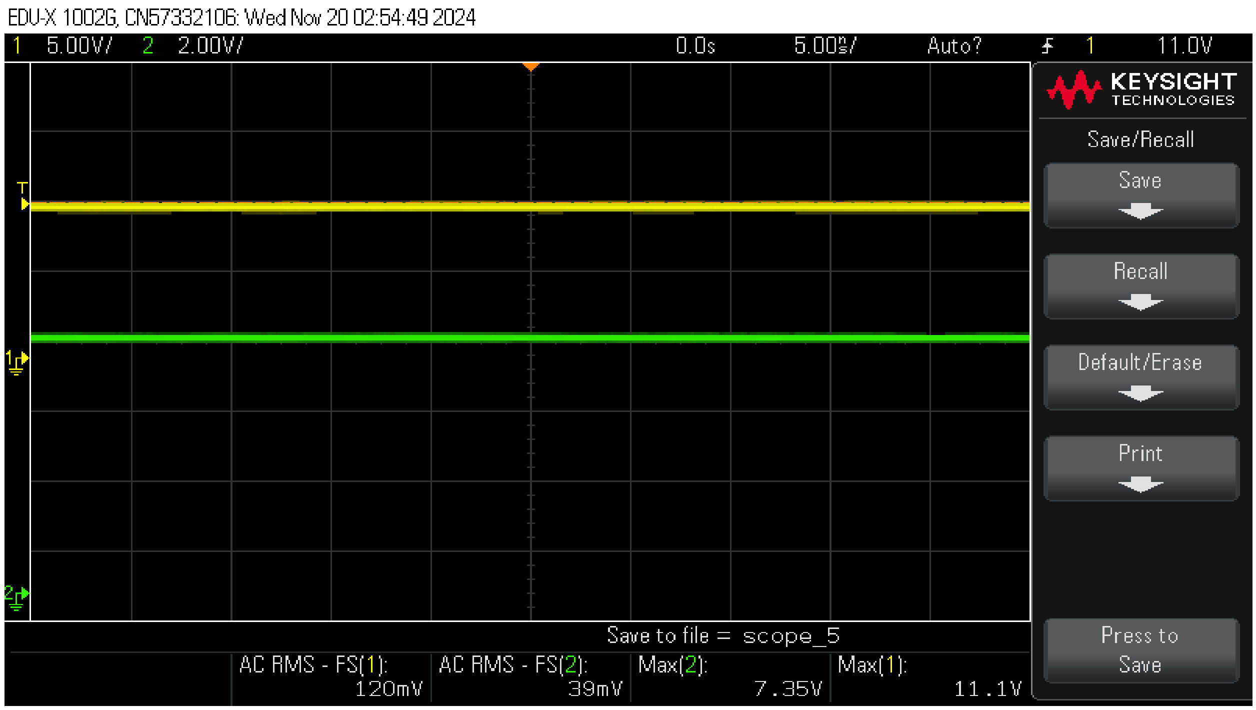


Figure 12: Scope of 7.4V Converter Noise

Figure 12 provides a measurement of the noise that is being inputted and outputted of the 7.4V Buck Converter. The yellow line represents the voltage input at 11.1V which has a noise of 120mV, while the green line represents the voltage output of the converter, measured with a noise of 39mV. Thus the converter is operating correctly in this aspect.

4.2.3 11.1 to 5.3V DCDC Validation

The 5.3V DCDC buck converter will be responsible for providing 5.3 Volts to the two ultrasonic sensors, radio subsystem, PixHawk 4 and will have to provide 0.4125A to operate.

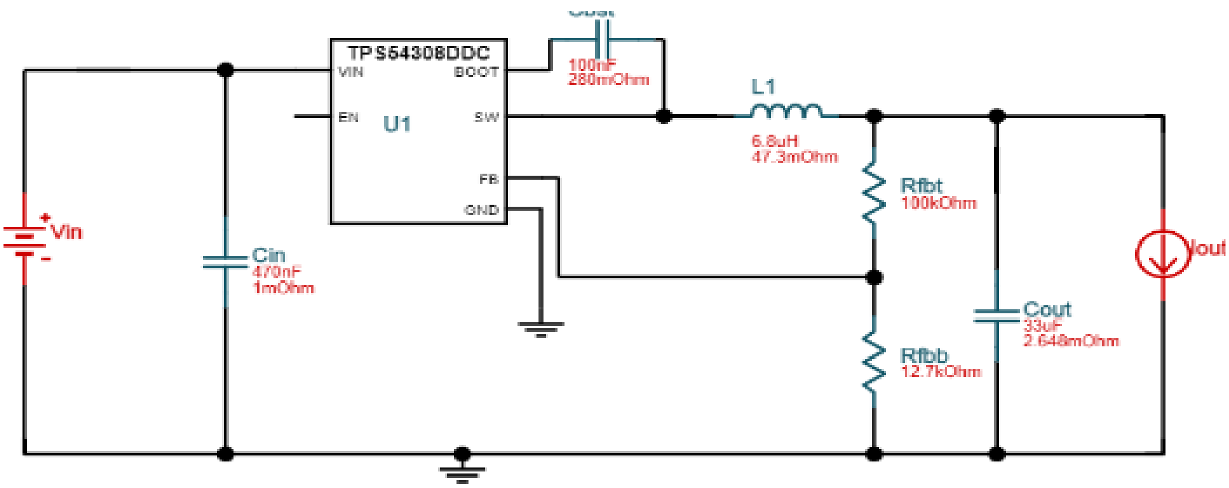


Figure 13 : Schematic of 11.1V to 5.3V Buck Converter

Changing input(200mA)		
Input Voltage	Output	Line Regulation(%)
12.6	5.222	0.7978723404
12.3	5.216	0.9118541033
12	5.215	0.9308510638
11.7	5.215	0.9308510638
11.4	5.213	0.9688449848
11.1	5.222	0.7978723404
10.8	5.221	0.8168693009
10.5	5.22	0.8358662614
10.2	5.22	0.8358662614
9.9	5.22	0.8358662614
9.6	5.21	1.025835866

Table 11: Line Regulation Test 5.3V Buck Converter

Table 11, was completed using a DC load that was drawing 200mA then the input voltage was changed from 12.6V to 9.6V, and output was collected. The change is seen in the column labeled Line Regulation. The converter is operating as expected.

Current Output		
Current	Voltage	Load Regulation(%)
0	5.264	0
50mA	5.25	0.2659574468
100mA	5.237	0.5129179331
150mA	5.222	0.7978723404
200mA	5.21	1.025835866
250mA	5.197	1.272796353
300mA	5.186	1.481762918
350mA	5.179	1.614741641
375mA	5.171	1.766717325
400mA	5.166	1.861702128

Table 12: Load Regulation Test 5.3V Buck Converter

Table 12, was completed using a DC load that was drawing from 0A and all the way up to 400mA while the input voltage was set to 11.1V. This is supposed to represent the maximum load that the converter should receive. The change is seen in the column labeled Load Regulation. The converter is operating as expected.

Judging by both *Table 11* and *Table 12* it is evident that this 5.3V buck converter did not output the desired 5.3 volts ever. However this slight (<0.1V) deviation will not cause any issues when powering components.

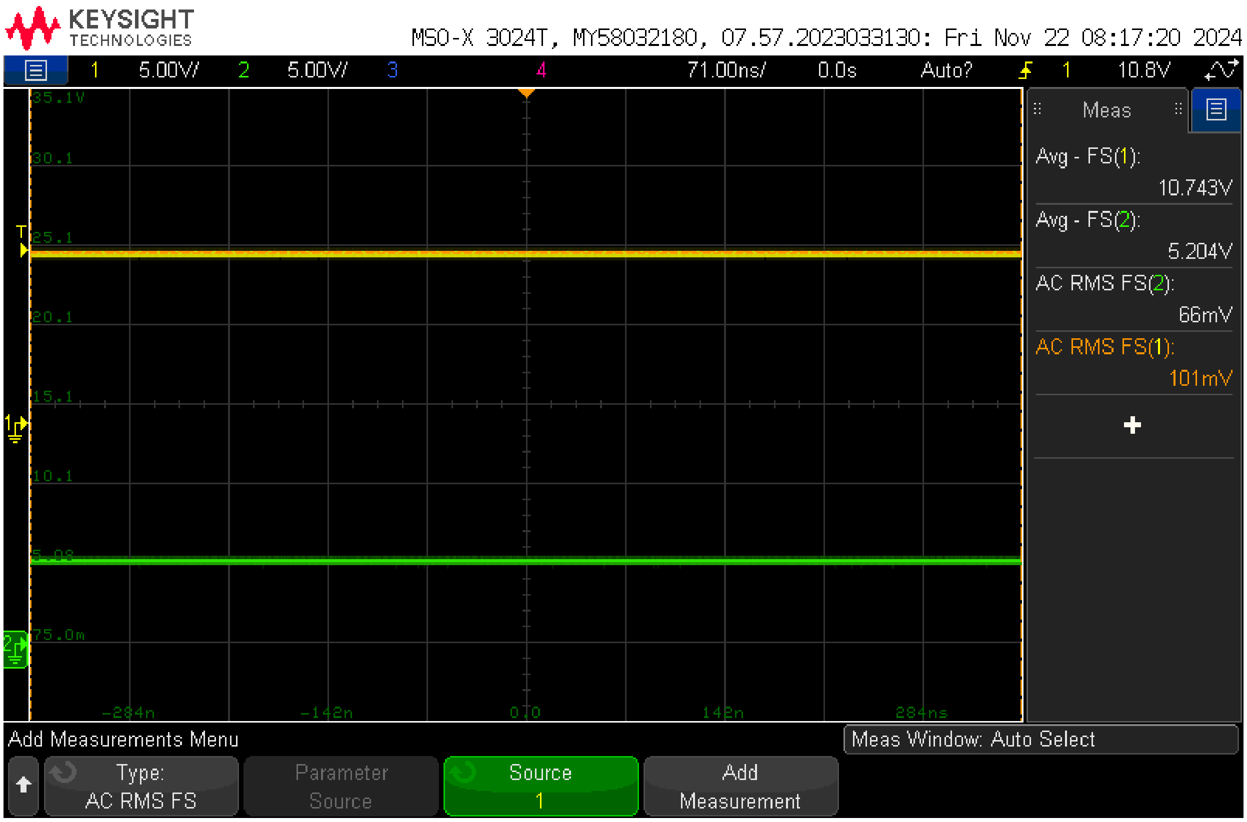


Figure 14: Scope of 5.3V Converter Noise

Figure 14 provides a measurement of the noise that is being inputted and outputted of the 7.4V Buck Converter. The yellow line represents the voltage input at 11.1V which has a noise of 101mV, while the green line represents the voltage output of the converter, measured with a noise of 66mV. Thus the converter is operating correctly in this aspect.

4.3 PixHawk Mission Planner Validation

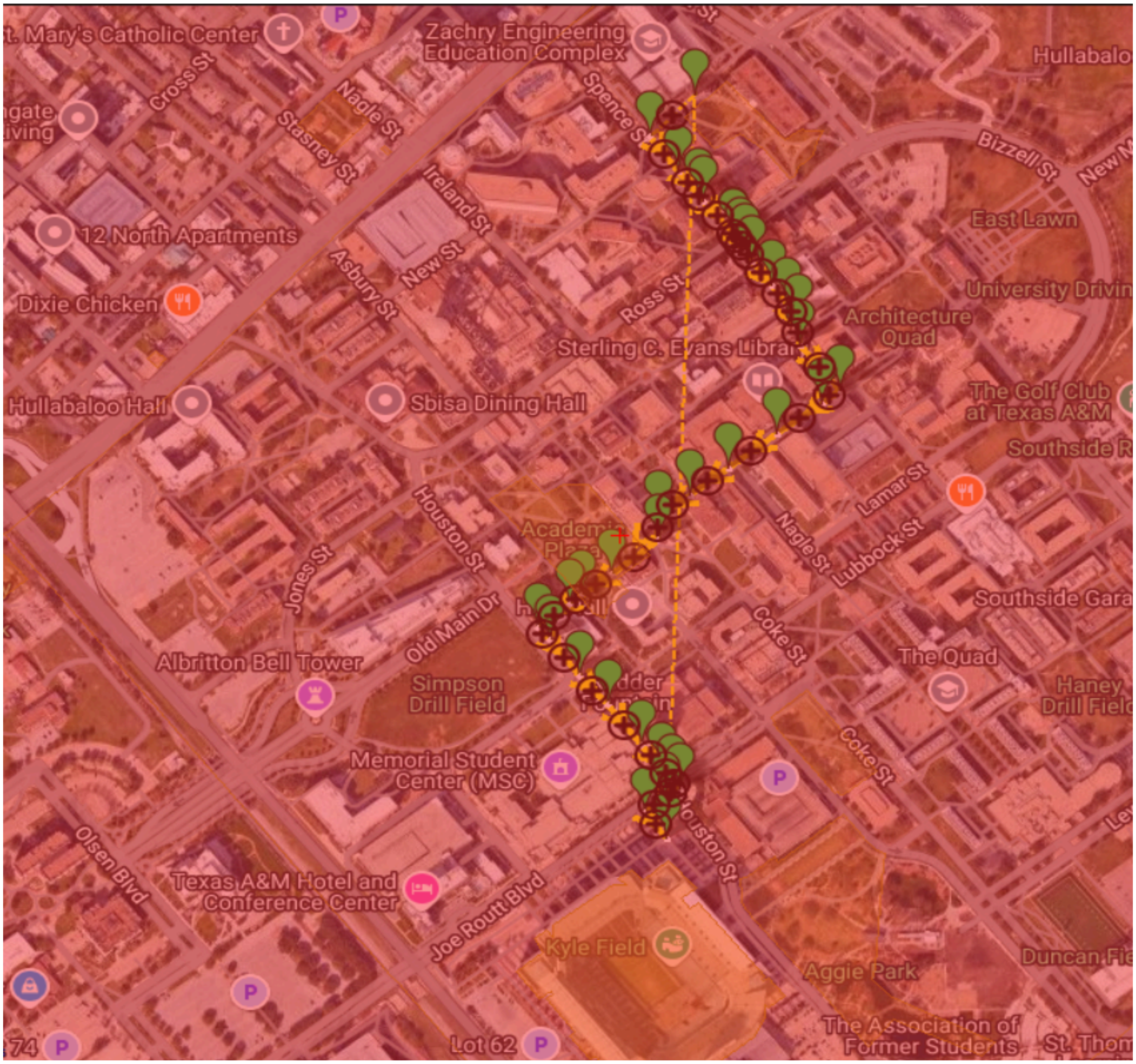


Figure 15: Path from EQUAD to Kyle Field on Mission Planner

The requirement from our sponsor was to have a path in front of Zach (EQUAD) to Kyle Field that the FoxBot can follow. An additional requirement was to have points where the FoxBot can pause and do an action(like stop transmitting); this is represented by the green points in *Image 9*. Additionally based on the speed at which the FoxBot can move it would take about 40 minutes for it to make this mile track. This is less than the 2 hour time line thus this is a pass and this will work.

4.4 Battery Fuel Gauge

The battery fuel gauge is responsible for monitoring the three 3S Lipo batteries aboard the FoxBot, thus there will be three of them designed for the three batteries. These battery monitors are responsible for alerting the ESP32 with a signal when there is a low battery level detected, there is also additional functionality required, so the user should be able to acquire additional data about the batteries, for example battery life, current draw, or voltage output.This is going to be done using the BQ7692003PWR. The goal for this semester was to design this system and test it next semester.

4.4 PCB Design

Part of this subsystem was to develop and construct the PCB board that would connect everything together.

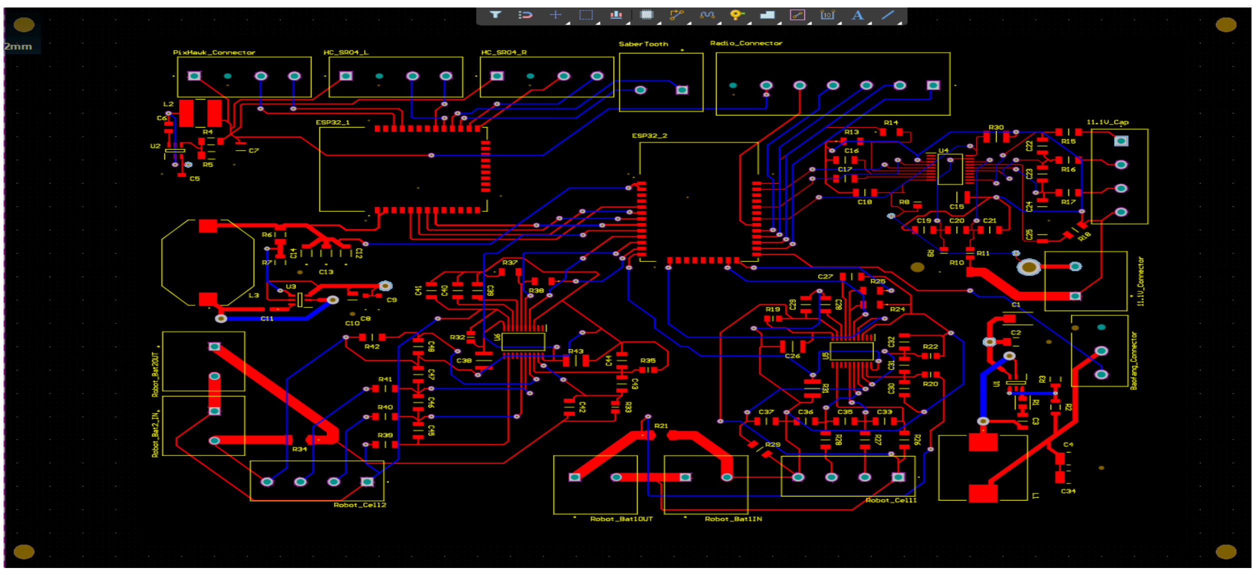


Figure 16: Altium PCB Design

4.5 Subsystem Conclusion

Overall the subsystem came together and operated very successfully. Of the parts that were tested everything worked correctly under all the conditions that were tested. However the converters all seem to drop in voltage very slightly, which is most likely due to a not solid soldering somewhere.

While the subsystem operated correctly this came with some “gerry rigging” to operate correctly, which of course will need to be fixed for future integration with other subsystems. For example when looking at *Image 10* looking at the 11.1V connector on the right side of the board. One of the through holes was not a hole to begin with, and was actually just a pad on the board. This caused the connector to not be able to be soldered to the PCB. And had to solder a wire to the board's pad directly. Another error or maybe just a quality of life thing is to have a connector that connects to the 3.3V Buck Converter on the left side of the board. This would all allow for easier testing of the converters, and had to just solder a wire directly to the inductor of that circuit.

Another error that was very quickly realized is an error with the battery monitors on the bottom of the PCB design. These batteries need to have a common ground with the ESP32 that is actually responsible for reading the battery monitors. Right now would require the three batteries being used to have an external ground between them outside the PCB board. This would also decrease the complexity of the battery monitors on the bottom of the PCB board. This error will need to be fixed so that the PCB board is at full functionality.

The power subsystem worked great, with each of the converters performing well under the line regulation test, output their respective voltages. With the greatest change in voltage only being about 1.02% for the 5.3V buck converter, which is about a 62mV variance in the expected output. Which occurred at the lowest voltage of 9.6V. This will of course still be in the operating voltages of the components. Next, in terms of the load regulation test the greatest output change was 2.52% for the 3.3V converter, which is about 81 mV variance in the expected output. This will of course still be in operating voltages of the components.