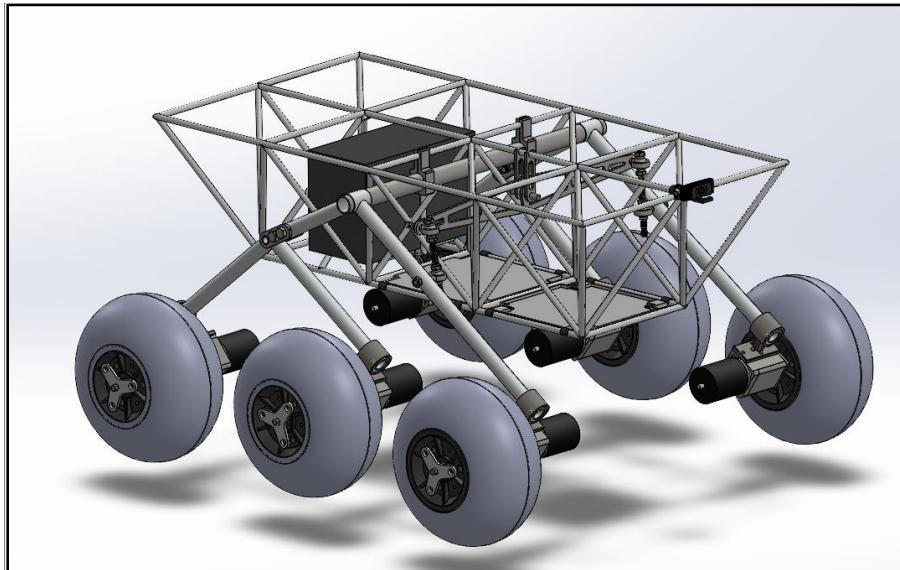


Planetary Rover for Rough Terrain

Tristan Hughes (ME), Michael Cannon (ME), Creighton Cathey (ECE), Patrick Maloney (EE), Kyle Pitre (ME), Etheranan Smith (EE)



L.O.U.I.S. (*Land-Optimized Upgradable Interplanetary System*)

I. Abstract

The "Planetary Exploration for Autonomous Rovers in Rough Terrain" project aims to design, fabricate, and test a prototype planetary rover that can be operated in various terrains on Earth. Sponsored by NASA and the LSU Electrical Engineering Department. This rover serves as an experimental platform for faculty, students, and future capstone teams. The project is inspired by existing technologies, such as NASA's Curiosity and Perseverance rovers. The primary objectives are to design a rover capable of driving on various terrains, operating remotely, running on an onboard power supply, monitoring system statuses, and transmitting real-time data. Future expandability is a core design consideration, as it will be modified by future capstone teams. Attempts to design, fabricate, and test the prototype planetary rover were made while adhering to the following functional requirements, constraints, and deliverables. The functional requirements include drive on diverse terrains, primarily loose, deformable such as sand and gravel, operate via remote control and onboard camera, utilize onboard rechargeable batteries for power, monitor system statuses through sensors, transmit real-time data for remote operation, and facilitate expansion for future capstone teams. Quantitative constraints include a weight limit, traveling speed, dimensions of rover, carrying capacity, minimum operating time, a budget of \$6,000, ability to traverse a minimum slope, and a maximum latency. Qualitative constraints include portability, durability, and ease of operation. The deliverables are a functional rover adhering to the given constraints and functional requirements, a method for remote operation, and a comprehensive final report and presentation detailing the technical and financial aspects of the project. By balancing performance requirements with engineering and budget constraints, this project aims to advance the understanding of mobile robotic systems for planetary exploration.

Table of Contents

I.	Abstract.....	2
II.	Executive Summary	12
III.	Engineering Specification	13
III.A.	Objective Statement	13
III.B.	Introduction	13
III.B.1.	Background Information	13
III.B.2.	Problem Description and Motivation for the Project	13
III.B.3.	Existing/Competing Technologies	14
III.B.4.	Potential Customers	14
III.C.	Functional Requirements.....	14
III.D.	Qualitative Constraints	16
III.E.	Measurable Engineering Specifications.....	16
III.F.	Deliverables	17
IV.	Embodiment	18
IV.A.	Functional Breakdown - Objective Tree(s).....	18
IV.B.	Concept/Solution Generation, Evaluation and Selection.....	20
IV.B.1.	Concept/Solution Generation Method(s) Used	20
IV.B.2.	Concept/Solution Evaluation Method(s) Used	20
IV.B.3.	Concept/Solution for Function F1 - Drive on Various Terrains	20
IV.B.4.	Concept/Solution for Function F2 - Facilitate Expansion.....	21
IV.B.5.	Concept/Solution for Function F3 – Distribute Power to Electronics	22
IV.B.6.	Concept/Solution for Function F4 – Operate Safely.....	23
IV.B.7.	Concept/Solution for Function F5 – Transmit Data	24
IV.B.8.	Concept/Solution for Function F6 – Operate Remotely.....	24
IV.B.9.	Concept/Solution for Function F7 – Monitor System Statuses	24
IV.B.10.	Concept/Solution for Function F8 – Contain Autonomous Features	25
IV.C.	System Description/Product Architecture.....	25

IV.C.1.	Description of Sub-System SS1 – Frame	27
IV.C.2.	Description of Sub-System SS2 – Drivetrain	28
IV.C.3.	Description of Sub-System SS3– Legs	29
IV.C.4.	Description of Sub-System SS4 – Differential Suspension	30
IV.C.5.	Description of Sub-System SS5 – Electric Propulsion	30
IV.C.6.	Description of Sub-System SS6 – Power Supply.....	31
IV.C.7.	Description of Sub-System SS7 – Power Distribution	31
IV.C.8.	Description of Sub-System SS8– Controller	31
	Description of Sub-System SS9 – Object Detection and Reaction.....	31
IV.C.9.	Description of Sub-System SS10 – Communication	32
V.	Engineering Analysis and Materials Selection	32
V.A.	Engineering Analysis for SS1 - Frame	32
V.A.1.	Types of Eng. Analysis Conducted:	32
V.A.2.	Eng. Analysis & Materials Selection for SS1-P1 - Frame: Reaction forces, stresses, and failure	32
V.A.3.	Eng. Analysis & Materials Selection for SS1-P1 – Center of Mass.....	32
V.A.4.	Eng. Analysis & Materials Selection for SS1-P2 – Electronics Platform: Impact...33	33
V.A.5.	Eng. Analysis & Materials Selection for SS1-P3 – Frame Axle: Stress, deflection, friction	33
V.B.	Engineering Analysis for SS2- Drivetrain	33
V.B.1.	Types of Eng. Analysis Conducted:	33
	Error! Bookmark not defined.	
V.B.2.	Eng. Analysis & Materials Selection for SS2-P2 – Motor Attachment: Stress	34
V.B.3.	Eng. Analysis & Materials Selection for SS2-P4– Torque on Wheel.....	34
V.B.4.	Eng. Analysis & Materials Selection for SS2-P3 – Axle: Stress.....	34
V.B.5.	Eng. Analysis & Materials Selection for SS2-P5-Motor Attachment Bolts: Shear Failure	35
V.C.	Engineering Analysis for SS3- Legs	35
V.C.1.	Types of Eng. Analysis Conducted:	35

V.C.2. Eng. Analysis & Materials Selection for SS3-P1/P2 – Rocker-Bogie: Reaction forces, bending moments, stress.....	35
V.C.3. Eng. Analysis & Materials Selection for SS3-P3 – Rocker-Bogie Bracket: stress, deflection.....	35
V.D. Engineering Analysis for SS4- Differential Suspension.....	36
V.D.1. Types of Eng. Analysis Conducted:	36
V.D.2. Eng. Analysis & Materials Selection for SS4-P1 – Differential Bar: Support, Stress, and deflection.....	36
V.D.3. Eng. Analysis & Materials Selection for SS4-P2- Differential Axle: Stress and deflection.....	36
V.D.4. Eng. Analysis & Materials Selection for SS4-P3 – Differential Bar: Stress and deflection.....	36
V.D.5. Eng. Analysis & Materials Selection for SS4-P4 – Differential Strut: Stress and buckling.....	37
V.D.6. Eng. Analysis & Materials Selection for SS4P1-P5 – Differential System: Kinematics	
37	
V.E. Engineering Analysis for SS5- Electric Propulsion	37
V.E.1. Types of Eng. Analysis Conducted:	37
V.E.2. Eng. Analysis & Materials Selection for SS5-P1 - Motor	37
V.E.3. Eng. Analysis & Materials Selection for SS5-P2- Motor Controller	38
V.F. Engineering Analysis for SS6 - Power Supply.....	38
V.F.1. Types of Eng. Analysis Conducted:	38
V.F.2. Eng. Analysis & Materials Selection for SS6-P1 - Battery	38
V.G. Engineering Analysis for SS7- Power Distribution	39
V.G.1. Types of Eng. Analysis Conducted:.....	39
V.G.2. Eng. Analysis & Materials Selection for SS7-P1 – Voltage Converters	39
V.G.3. Eng. Analysis & Materials Selection for SS7-P2– Wires	39
V.H. Engineering Analysis for SS8- Controller.....	40
V.H.1. Types of Eng. Analysis Conducted:	40

V.H.2.	Eng. Analysis & Materials Selection for SS8-P1 – Digital Controller.....	40
V.H.3.	Eng. Analysis & Materials Selection for SS8-P2– Digital Kill Switch.....	40
V.I.	Engineering Analysis for SS9- Object Detection and Reaction.....	41
V.I.1.	Types of Eng. Analysis Conducted:	41
V.I.2.	Eng. Analysis & Materials Selection for SS9-P1 - Servomotor	41
V.I.3.	Eng. Analysis & Materials Selection for SS9-P2 – Ultra-Sonic Sensors	41
V.I.4.	Eng. Analysis & Materials Selection for SS9-P3 - Algorithms.....	41
V.I.5.	Eng. Analysis & Materials Selection for SS9-P4 – Single Board Computer	41
V.I.6.	Eng. Analysis & Materials Selection for SS9-P5 - Microcontroller	42
V.I.7.	Eng. Analysis & Materials Selection for SS9-P6 - Gyroscope/Accelerometer.....	42
V.I.8.	Eng. Analysis & Materials Selection for SS9-P7 - Camera.....	42
V.J.	Engineering Analysis for SS10- Communication	42
V.J.1.	Types of Eng. Analysis Conducted:	42
V.J.2.	Eng. Analysis & Materials Selection for SS10-P1 – Wi-fi- module	43
V.J.3.	Eng. Analysis & Materials Selection for SS10-P2 - Serial Communication	43
VI.	Manufacturing & Assembly	43
VI.A.	Manufacturing	43
VI.B.	Assembly.....	44
VII.	Safety Considerations	46
VII.A.	Operational Safety.....	46
VII.B.	Safety During Manufacturing, Assembly and Testing	46
VIII.	Environmental Impact Assessment.....	47
VIII.A.	Manufacturing Related Environmental Impact	47
VIII.B.	Operation/Usage Related Environmental Impact.....	47
VIII.C.	End-Of-Life (Disposal) Related Environmental Impact.....	47
IX.	Design-Phase Testing	47
IX.A.	Design-Phase Testing Objectives, Rationale & Brief Description.....	47
IX.B.	Design-Phase Testing Outcomes & Conclusions.....	48

X.	Testing, Validation, and/or Implementation.....	48
X.A.	Testing & Validation of Function F1 - Drive on Various Terrains	48
X.A.1.	Objective, Rationale & Brief Description - F1	48
X.A.2.	Testing & Validation Summary of Results & Conclusions – F1.....	49
X.B.	Testing & Validation of Function F2 - Facilitate Expansion.....	50
X.B.1.	Objective, Rationale & Brief Description - F2	50
X.B.2.	Testing & Validation Summary of Results & Conclusions – F2.....	50
X.C.	Testing & Validation of Function F3 – Distribute Power to Electronics	50
X.C.1.	Objective, Rationale & Brief Description – F3.....	50
X.C.2.	Testing & Validation Summary of Results & Conclusions – F3.....	51
X.D.	Testing & Validation of Function F4 – Operate Safely	51
X.D.1.	Objective, Rationale & Brief Description – F4.....	51
X.D.2.	Testing & Validation Summary of Results & Conclusions – F4.....	51
X.E.	Testing & Validation of Function F5 – Transmit Data	51
X.E.1.	Objective, Rationale & Brief Description – F5.....	52
X.E.2.	Testing & Validation Summary of Results & Conclusions – F5.....	52
X.F.	Testing & Validation of Function F6 – Operate Remotely	52
X.F.1.	Objective, Rationale & Brief Description – F6.....	52
X.F.2.	Testing & Validation Summary of Results & Conclusions – F6.....	53
X.G.	Testing & Validation of Function F7 – Monitor System Statuses.....	53
X.G.1.	Objective, Rationale & Brief Description – F7	53
X.G.2.	Testing & Validation Summary of Results & Conclusions – F7	53
X.H.	Testing & Validation of Function F8 – Contain Autonomous Features	53
X.H.1.	Objective, Rationale & Brief Description – F8.....	53
X.H.2.	Testing & Validation Summary of Results & Conclusions – F8.....	54
X.I.	Testing & Validation of Qualitative Constraint Q1 - Portability	54
X.I.1.	Objective, Rationale & Brief Description - Q1	54
X.I.2.	Testing & Validation Summary of Results & Conclusions – Q1	54

X.J. Testing & Validation of Qualitative Constraint Q2 - Durability	54
X.J.1. Objective, Rationale & Brief Description - Q2.....	54
X.J.2. Testing & Validation Summary of Results & Conclusions – Q2	54
X.K. Testing & Validation of Qualitative Constraint Q3 – Ease of Operation	55
X.K.1. Objective, Rationale & Brief Description – Q3.....	55
X.K.2. Testing & Validation Summary of Results & Conclusions – Q3	55
X.L. Testing & Validation of Qualitative Constraint Q4 – Aesthetically Pleasing	55
X.L.1. Objective, Rationale & Brief Description – Q4.....	55
X.L.2. Testing & Validation Summary of Results & Conclusions – Q4	56
X.M. Testing & Validation of Quantitative Constraint M1 - Weight.....	56
X.M.1. Objective, Rationale & Brief Description - M1.....	56
X.M.2. Testing & Validation Summary of Results & Conclusions – M1	56
X.N. Testing & Validation of Quantitative Constraint M2 - Latency	56
X.N.1. Objective, Rationale & Brief Description - M2.....	56
X.N.2. Testing & Validation Summary of Results & Conclusions – M2.....	57
X.O. Testing & Validation of Quantitative Constraint M3 - Budget.....	57
X.O.1. Objective, Rationale & Brief Description – M3	57
X.O.2. Testing & Validation Summary of Results & Conclusions – M3	57
X.P. Testing & Validation of Quantitative Constraint M4 – Carrying Capacity	57
X.P.1. Objective, Rationale & Brief Description – M4.....	57
X.P.2. Testing & Validation Summary of Results & Conclusions – M4.....	57
X.Q. Testing & Validation of Quantitative Constraint M5 - Speed	58
X.Q.1. Objective, Rationale & Brief Description – M5	58
X.Q.2. Testing & Validation Summary of Results & Conclusions – M5	58
X.R. Testing & Validation of Quantitative Constraint M6 - Dimensions	58
X.R.1. Objective, Rationale & Brief Description – M6.....	58
X.R.2. Testing & Validation Summary of Results & Conclusions – M6.....	58
X.S. Testing & Validation of Quantitative Constraint M7 – Operation Time	58

X.S.1.	Objective, Rationale & Brief Description – M7	58
X.S.2.	Testing & Validation Summary of Results & Conclusions – M7	59
X.T.	Testing & Validation of Quantitative Constraint M8 - Slope	59
X.T.1.	Objective, Rationale & Brief Description – M8.....	59
X.T.2.	Testing & Validation Summary of Results & Conclusions – M8.....	59
X.U.	Testing & Validation of Quantitative Constraint M9 – Detection Distance	59
X.U.1.	Objective, Rationale & Brief Description – M9.....	60
X.U.2.	Testing & Validation Summary of Results & Conclusions – M9.....	60
XI.	Project Management	60
XI.A.	Schedule and Milestones	60
XI.B.	Budget.....	62
XII.	Summary and Conclusions.....	62
XIII.	Appendix	63
XIII.A.	Quality Function Deployment (QFD) - HoQ	63
XIII.B.	Concept/Solution Evaluation and Selection Supplement	65
XIII.B.1.	Concept/Solution for Function F1 – Drive on Various Terrain	67
XIII.B.2.	Concept/Solution for Function F2 – Facilitate Expansion.....	67
XIII.B.3.	Concept/Solution for Function F3 – Distribute Power to Electronics	68
XIII.B.4.	Concept/Solution for Function F4 – Operate Safely.....	70
XIII.B.5.	Concept/Solution for Function F5 – Transmit Data	70
XIII.B.6.	Concept/Solution for Function F6 – Operate Remotely.....	71
XIII.B.7.	Concept/Solution for Function F7 – Monitor System Statuses	72
XIII.B.8.	Concept/Solution for Function F8 – Contain Autonomous Features	73
XIII.C.	System Description/Product Architecture Supplement.....	75
XIII.C.1.	Sub-System SS1 - Frame.....	75
XIII.C.2.	Sub-System SS2 - Drivetrain.....	88
XIII.C.3.	Sub-System SS3 - Legs	105
XIII.C.4.	Sub-System SS4 – Differential Suspension.....	120

XIII.C.5.	Sub-System SS5 – Electric Propulsion.....	131
XIII.C.6.	Sub-System SS6 – Power Supply	137
XIII.C.7.	Sub-System SS7 – Power Distribution	142
XIII.C.8.	Sub-System SS8 - Controller	149
XIII.C.9.	Sub-System SS9 – Object Detection and Reaction	153
XIII.C.10.	Sub-System SS10 - Communication.....	162
XIII.D.	Engineering Analysis and Materials Selection Supplement	166
XIII.D.1.	Eng. Analysis Details for SS1 - Frame.....	166
XIII.D.2.	Eng. Analysis Details for SS2 - Drivetrain.....	180
XIII.D.3.	Eng. Analysis Details for SS3 - Legs	186
XIII.D.4.	Eng. Analysis Details for SS - 4 – Differential	191
XIII.D.5.	Eng. Analysis Details for SS5 – Electric Propulsion.....	211
XIII.D.6.	Eng. Analysis Details for SS6 – Power Supply	226
XIII.D.7.	Eng. Analysis Details for SS7 – Power Distribution	237
XIII.D.8.	Eng. Analysis Details for SS8 - Controller	241
XIII.D.9.	Eng. Analysis Details for SS9 – Object Detection and Reaction	241
XIII.D.10.	Eng. Analysis Details for SS10 - Communication.....	243
XIII.E.	Manufacturing & Assembly Supplement	246
XIII.E.1.	Manufacturing Processes	246
XIII.E.2.	Assembly Processes	250
XIII.F.	Design Phase Testing Supplement.....	254
XIII.F.1.	Design Phase Testing Methods and Details	254
XIII.F.2.	Design Phase Testing Results.....	254
XIII.G.	Testing & Validation Supplement.....	254
XIII.G.1.	Testing & Validation of Function F1 - Drive on Various Terrains.....	254
XIII.G.2.	Testing & Validation of Function F2 - Facilitate Expansion	258
XIII.G.3.	Testing & Validation of Function F3 – Distribute Power to Electronics	259
XIII.G.4.	Testing & Validation of Function F4 – Operate Safely	268

XIII.G.5.	Testing & Validation of Function F5 – Transmit Data.....	269
XIII.G.6.	Testing & Validation of Function F6 – Operate Remotely	271
XIII.G.7.	Testing & Validation of Function F7 – Monitor System Status	272
XIII.G.8.	Testing & Validation of Function F8 – Contain Autonomous Features	273
XIII.G.9.	Testing & Validation of Qualitative Constraint Q1 - Portability	291
XIII.G.10.	Testing & Validation of Qualitative Constraint Q2 - Durability.....	291
XIII.G.11.	Testing & Validation of Qualitative Constraint Q3 – Ease of Operation	295
XIII.G.12.	Testing & Validation of Qualitative Constraint Q4 – Aesthetically Pleasing .	295
XIII.G.13.	Testing & Validation of Quantitative Constraint M1 - Weight	296
XIII.G.14.	Testing & Validation of Quantitative Constraint M2 - Latency.....	297
XIII.G.15.	Testing & Validation of Quantitative Constraint M3 - Budget.....	299
XIII.G.16.	Testing & Validation of Quantitative Constraint M4 – Carrying Capacity	302
XIII.G.17.	Testing & Validation of Quantitative Constraint M5 - Speed.....	304
XIII.G.18.	Testing & Validation of Quantitative Constraint M6 - Dimensions.....	305
XIII.G.19.	Testing & Validation of Quantitative Constraint M7 – Operation Time	305
XIII.G.20.	Testing & Validation of Quantitative Constraint M8 - Slope	319
XIII.G.21.	Testing & Validation of Quantitative Constraint M9 – Detection Distance....	320
XIII.H.	Project Management Supplement.....	322
XIII.H.1.	Schedule and Milestones Details.....	322
XIII.H.2.	Budget Details.....	324
XIII.I.	Reference Materials	328
XIII.I.1.	Codes and Standards.....	328
XIII.I.2.	References and Bibliography	328

II. Executive Summary

Objectives and Problem Definition: Team 57 is tasked with creating a mobile, durable, and simultaneously lightweight autonomous planetary rover with room for expansion for future capstone teams, the LSU Electrical Engineering Department, and the LSU Mechanical Engineering Department. The goal of the project is to produce a rover that can help educate students on robotics and expand the research horizons of the departments. |

Key Engineering Specifications: | The rover must have a battery life of 1 hour, weight less than 100 kg, support an additional 35 kg, move at a top speed of at least 1.25 m/s, detect objects at a distance of 2 m, react to those objects, and travel up inclines of at least 5 degrees. |

Product/System Description: | The rover utilizes a lightweight skeletal frame complete with a 6-wheel rocker bogie suspension mechanism and drivetrain. The legs and suspension system together allow for the rover to maintain six points of contact with the ground at all time, and proved to be capable of navigate rough terrain without issue. The rover is also equipped with a lightweight lithium-ion battery that allows the rover to run for over an hour without issue, a webserver that allows for easy connection and usage, and two ultrasonic sensors that allow it to detect objects at a distance. |

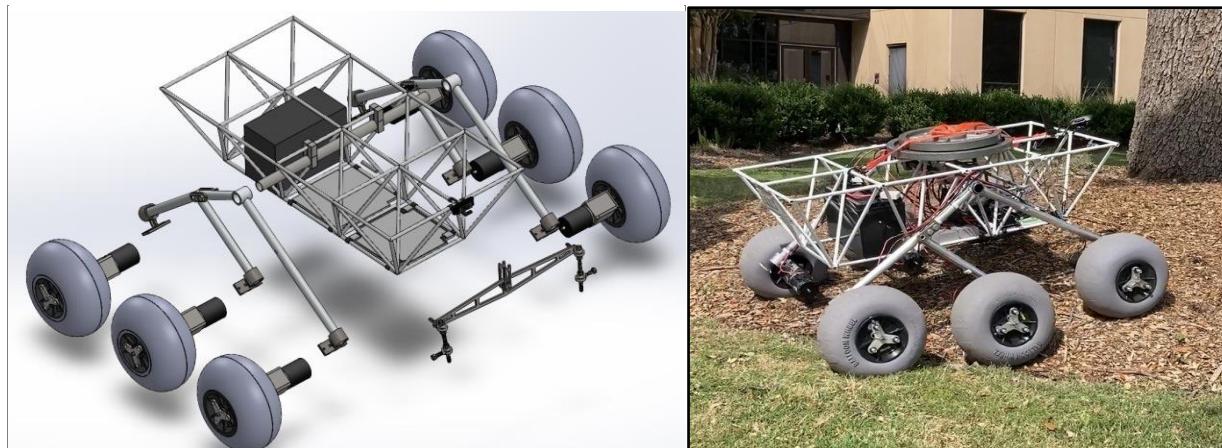


Figure II.1: SolidWorks model of exploded view (left) and final physical prototype (right)

Engineering Analysis Result Summary: | The results of the engineering analysis conducted throughout the semester for the electrical portion of the team were useful in selecting appropriate components as well as giving an idea for how algorithms will be written in the spring. Motor and battery current calculations were done to size both components correctly. Flowcharts were made and pseudocode was written for the motor controller, AI implementation, and other components that required it. FEA analysis was utilized for parts generated in SolidWorks that were too complex to model by hand. |

Manufacturing/Assembly Summary: | Manufacturing for the rover was done in stages, with each individual subsystem being manufactured, assembled, and tested (where applicable) before being put together. This was done with a variety of manufacturing techniques, including but not limited to: MIG welding (steel and aluminum), Lathe, Mill, Waterjet, 3D printing, soldering etc. |

Safety Considerations: | During operation of the rover, fuses were in place to provide overcurrent protection and a digital kill switch were implemented into the digital controller. All electrical

connections must be checked before pushing the rover to avoid burning the motor controllers. Proper PPE will be worn at all times, including safety glasses, closed toed shoes, and long pants. It will also be important to comply with OSHA standards and ensure that no one student is lifting above the maximum recommended load of 50 lbs. When testing, it is also important to ensure that the rover is not being tested on a slope that is overly steep so as to avoid the rover's motors potentially stalling and slipping.

Design-Phase Testing Results: Team 57 did not conduct any testing during design phase.

Validation Test Plan & Results Summary: Testing was conducted for several of the subsystems before the full assembly of the rover. This included torque testing, strain-gauge testing, and impact testing. Once fully assembled, the rover completes various terrain test, with different surfaces, slope and additional loads being added. All constraints were met after testing was completed and the rover passed every test without fail.

Statement of Prototype Performance: The rover was able to pass every test without fail. This included gravel, grass, and sand slopes with a slope up to 20 degrees and with an additional payload of 36 kg. The original object of this project included an autonomous element, which was implemented at a low level, but the architecture is there for future capstone teams to build off of.

III. Engineering Specification

III.A. Objective Statement

Team 57 was tasked with creating a mobile, durable, and simultaneously lightweight autonomous planetary rover with room for expansion for future capstone teams, the LSU Electrical Engineering Department, and the LSU Mechanical Engineering Department. The goal of the project is to produce a rover that can help educate students on robotics and expand the research horizons of the departments.

III.B. Introduction

III.B.1. Background Information

NASA landed the first planetary rover on Mars in July of 1997. Ever since, it has been a mission of theirs to gather various information and possible proof of life on the planet. There have been five successful rover missions to Mars by NASA since 1997 and with the Artemis missions to the moon coming in the near future, the plan is to use information gathered there to send more missions to Mars. NASA is interested in finding signs of life on Mars and rovers are their primary way of doing this. Rovers will be an important part of the future of space exploration, and it is necessary to expose young engineers to their specific design and manufacturing processes.

III.B.2. Problem Description and Motivation for the Project

The motivation for this project is to have a working rover that can be experimented with and built upon by LSU engineering faculty and students. The rover will be used for research in areas such as: computer vision, solar panels, GPU computing, and wireless communication. All of these areas will not be implemented by our team but should be implemented by future capstone

teams. The final motivation for this project is providing a rover that can be used for teaching robotics and control-related courses.

III.B.3. Existing/Competing Technologies

NASA Curiosity Rover - The Mars Curiosity Rover was launched in November of 2011. It is one of two planetary rovers currently in operation on Mars. Curiosity has a mass of 899 kg and is 3 m x 2.7 m x 2.2 m. It has been recorded climbing a slope of up to 23 degrees. This rover uses a plutonium dioxide generator for power, but they have two backup lithium-ion batteries in reserve. It is also considerably larger, heavier, and more powerful than the rover we intend to build, but it is the main inspiration for our design.

Ripping It Outdoors Rover - This is an example of an Earth rover that operates similar to how we want ours to operate. It successfully fulfills all the functions listed in our functionality section. It has a price range near our budget (\$4,000-\$6,000 depending on options) and a weight of 210 lbs. While it is not directly competing with our rover, it is beneficial to use it as a reference for weight and budgeting purposes

III.B.4. Potential Customers

Primary: NASA - This project is sponsored by NASA. LSU Electrical Engineering Department - The department would be getting an operational rover to experiment with and build upon.

Secondary: Future Capstone Students - This is a multi-year project that will continue to be built upon by capstone teams in the coming years using the rover we built as a base to start from. Biologists/Geologists - A major function of fully operating rovers is to be able to take samples of rock and possible organic material to be studied further. Emergency Response Teams - This rover could also potentially be used on Earth in response to natural disasters in areas where people may not be able to reach because of difficult terrain.

III.C. Functional Requirements

[Table III-1: Required Functions]

#	Function	Explanation	Type(s) of Validation Test(s)
F1	Drive on Various Terrains	The rover must move forward and backwards, change and maintain speeds, and turn as desired. It will perform these operations effectively on various terrains.	Rover driven on concrete, gravel, grass, and sand. Average speeds, turning rates, and turning radii recorded. Obstacles of known height driven over to test suspension system.
F2	Facilitate Expansion	The project is intended to be continued by future capstone teams in the coming years. The design must have additional internal space and weight capacity to incorporate future add-ons.	Extra weights to simulate future components added during driving tests.
F3	Distribute Power to Electronics	The rover will be powered by an onboard rechargeable lithium-ion	Testing individual electrical components as necessary to verify

#	Function	Explanation	Type(s) of Validation Test(s)
		battery. This battery, in combination with bus bars, voltage converters, and appropriately sized wiring, will be responsible for providing power throughout the rover.	they work as advertised, testing of subsystems as they are constructed to verify the components work in collaboration as expected to power all electrical systems on the rover. A torque test was also conducted.
F4	Operate Safely	The rover will utilize a kill-switch to halt all operations to prevent hazards. Overcurrent protection will also be implemented in the design to prevent damages.	Field-test of physical and digital kill switches while operating the rover.
F5	Transmit Data	The data gathered by the onboard system will be transmitted in real-time to be utilized by the rover operator. The transmission of this data will also detect and help prevent onboard issues.	Camera operation testing and ultrasonic sensor tests.
F6	Operate Remotely	The rover's movement and speed will be controlled by a remote control that is not physically connected to the rover with guidance from an onboard camera. A movable camera will allow the operator to control the rover from a remote location.	Controller distance test.
F7	Monitor System Statuses	Sensors throughout the rover will observe and collect data relating to different components. Statuses like remaining battery life, internal temperatures, speed, and incline are of importance and will need to be tracked to ensure uninterrupted operations.	Test of individual sensors
F8	Contain Autonomous Features	The rover will autonomously detect and react to objects by utilizing the onboard sensors and camera.	Tests of ultrasonic sensors and autonomous mode

III.D. Qualitative Constraints

[Table III-2]: Required Qualitative Constraints

#	Qualitative Constraint	Explanation	Type(s) of Validation Test(s)
Q1	Portability	To facilitate easy transportation, the rover should be small and light enough to be moved by a team without disassembly	Rover carried by team over 500 ft. Rover transported by pickup truck between testing locations.
Q2	Durability	Rover needs to withstand external Baton Rouge environment and resist excessive wear from use in sandy and rocky environments	Drive testing, Strain-gauge testing
Q3	Ease of Operation	Rover operation should be intuitive and easy to learn	Anonymous survey conducted
Q4	Aesthetically Pleasing	This project is a representation of both the student's work and the LSU engineering department as a whole, and as such it is important to be presentable.	Anonymous survey conducted

III.E. Measurable Engineering Specifications

[Table III-3]: Required Measurable Engineering Specifications

#	Name	Symbol	Units	Value(s)	Explanation	Type(s) of Test(s)
M1	Weight	W	kg	100	Rover needs to be light enough to be carried by a capstone team.	Weighing the rover on a scale
M2	Latency	L	ms	50	For the ease of use for the operators of the rover, it is important for latency to be minimized.	Latency testing for each part of the communication system and adding all latency together.
M3	Budget	B	\$	6000	A budget of \$6000 was decided upon by the sponsor	N/A
M4	Carrying Capacity	P	kg	35	The project is intended to be built upon by future capstone teams, so capacity for extra weight is incorporated.	Extra weight test during normal rover operation
M5	Speed	V	m/s	1.25	Typical strolling speed. Chosen so operator can easily keep up with rover.	Speed Test
M6	Dimensions	D	m ³	1.25	This is a 1/3rd scale of the last two NASA Rovers.	Dimensions Test

#	Name	Symbol	Units	Value(s)	Explanation	Type(s) of Test(s)
M7	Operation Time	T	min	60	An operating time of 60 minutes was picked by the sponsor.	Battery depletion test
M8	Slope	θ	$^{\circ}$	5	A 5° slope was decided upon by the sponsor to fit the criteria of the rover being mobile in rough terrain.	Driving rover on known/measured slopes
M9	Detection Distance	S	m	2	As part of the 'autonomous' the rover will be able to detect objects at a distance.	Ultrasonic distance tests

III.F. Deliverables

Upon completion of this capstone project, the team was capable of designing and manufacturing a fully operational planetary rover meeting all the required constraints. The rover utilizes a lightweight but simultaneously strong skeletal frame that provides future capstone teams the ability to expand upon this project. The 6-wheel rocker-bogie suspension system with all-wheel drive allows the rover to be capable of climbing a series of obstacles while operating on various terrains at a maximum speed of 1.3 m/s. With an onboard lithium-ion battery, it is capable of doing this well over the desired 60-minute operation time. The rover was designed to be easily disassembled in order to allow future teams to break apart how all components work together and make any redesigns deemed necessary. This also provides easy means of transportation should the rover be needed to be moved over a long distance in a smaller area than the 1.25 m^3 size limitation given by the sponsor. Taking all this into consideration, the project is considered a success by both the team members as well as the sponsor, Dr. Meng.]

IV. Embodiment

IV.A. Functional Breakdown - Objective Tree(s)

The functional requirements have been previously explained in Section III.C; there are several sub-functions needed to satisfy these main functions. Under the "drive on various terrains" function, the "drive straight & turn" sub-function enables forward movement by receiving and responding to user controller inputs, transferring power to the correct motors, and ensuring sufficient traction to prevent slipping. Additionally, it utilizes independent wheel rotation for desired directional control by rotating wheels on the left and right side at different rates for turning in the appropriate directions or at the same rate for forward and backward motion. In the "operate remotely" function, the first sub-function involves converting user input into an electrical signal, transmitted through the rover's electrical systems via the microcontroller. This signal is then translated into mechanical motion by the necessary controller. For the "distribute power to electronics" function, the sub-functions involve providing a space for a battery and administering power as needed. To achieve the "facilitate expansion" function, the rover must support additional weight and allow space for future add-ons by other teams. Under the "contain autonomous features" function, two sub-functions are crucial; "detect objects" and "react to objects." There are ultrasonic distance sensors mounted to the front of the rover for allowing objects to be detected at a distance, and the rover adjusts its path based on the identified objects following a Bug-0 algorithm. For "monitor system statuses," the sub-functions include "utilize sensors" to integrate onboard sensors into the rover's design and "gather sensor data" to acquire information about various rover statuses. The "transmit data" function is realized through sub-functions for sending and receiving gathered data. Finally, the "operate safely" function includes sub-functions for emergency shutoff and safe power distribution, utilizing fuses, overcurrent protection, as well as physical and digital kill switches to shut down all operations. |

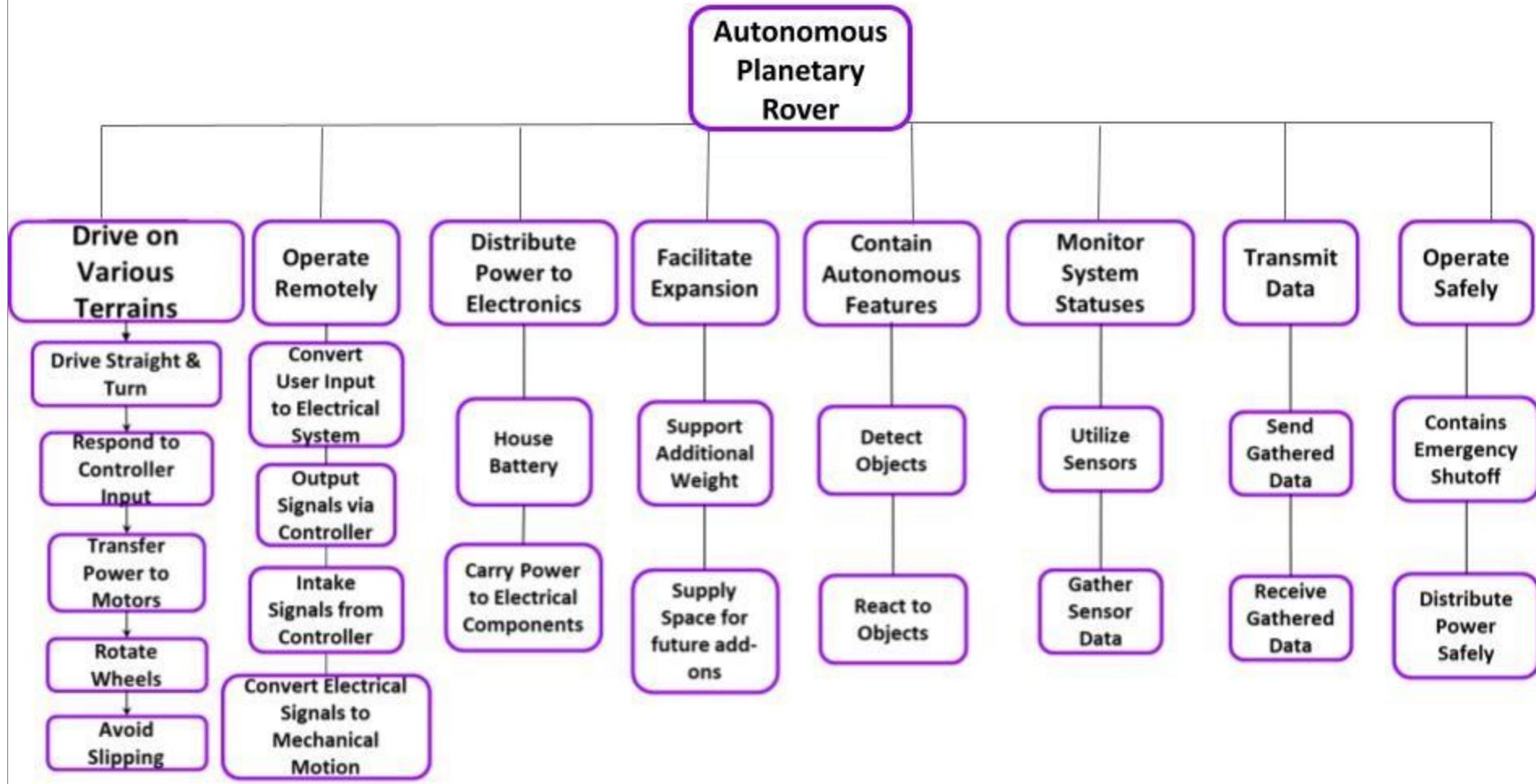


Figure IV-1: Project Objective Tree

IV.B. Concept/Solution Generation, Evaluation and Selection

IV.B.1. Concept/Solution Generation Method(s) Used

To generate concepts for the rover design, the functions were first divided into mechanical and electrical categories. The group members then broke into groups with their respective disciplines to generate concepts to accomplish the mechanical and electrical objectives. This was achieved by having each member brainstorm individually; then, the groups met and discussed ideas to see if any could be combined or were generated by multiple members. This also allowed for further brainstorming as a group to generate the final list of concepts for each function. With the concepts established by each discipline, the two groups then discussed all ideas created for both mechanical and electrical objectives to determine if certain concepts from one discipline would help concepts in the selection of concepts from the other discipline.

IV.B.2. Concept/Solution Evaluation Method(s) Used

The criterion required to achieve each function was first determined, so weights could be assigned to each criterion. The weights were determined by assigning values to depict the importance of each criterion when compared to the other criterion in an AHP. Examples of the AHPs are shown in Appendix VII.B. A one implies the two criteria are equally important, values greater than one imply the row is more important than the column, and values less than one implies the row is less important than the column. These values were discussed as a group to assign values from 0.1 – 9 in order most accurately represent the function being considered. With these values assigned, the geometric average for each row was then found by summing the values in the row and taking the respective root. The geometric averages were then used to assign weights for each criterion by dividing the average of each row by the sum of the averages of all the criteria.

With the weights for each criterion determined, a decision matrix could then be made to compare the concepts for each function against their respective counterparts. A datum design was established amongst the various design concepts, which was used as a middle ground to compare the other concepts to. For the datum, a value of 0 was assigned for each criterion, and the remaining concepts were given -1, 0, or 1 for each criterion depending on if it was worse, the same, or better than the datum in the respective aspect. For decision matrices where extensive data was available, values ranging from -9 to 9 were assigned rather than -1 to 1. These values were multiplied by the respective weights determined by the AHP and summed to determine the concept weights. The concept with the highest weight was chosen for the design for each function.

IV.B.3. Concept/Solution for Function F1 - Drive on Various Terrains

IV.B.3.a. Original Concept/Solution Selected for Function F1

A motor that can provide the correct amount of torque at the correct speed for our application is required for the rover to be capable of driving on various terrains. We selected brushed DC motors to achieve this. A pros and cons list was generated in order to come up with this selection. Brushed DC motors can provide the torque we need and even greater given a gearbox attached to it. It can do this while staying relatively inexpensive, which was a major factor in our decision. The major downside to brushed DC motors is that the brushes will degrade over

time and require replacement. This is not an issue for us as we expect the motors will likely be improved upon by future capstone teams anyway. The torque was transmitted to the wheel using a driveshaft extender that keyed into a hubcap. A rocker-bogie mechanism was designed to allow relative motion between the wheels. A differential stabilizer arm was added to provide stability and flexibility while driving.

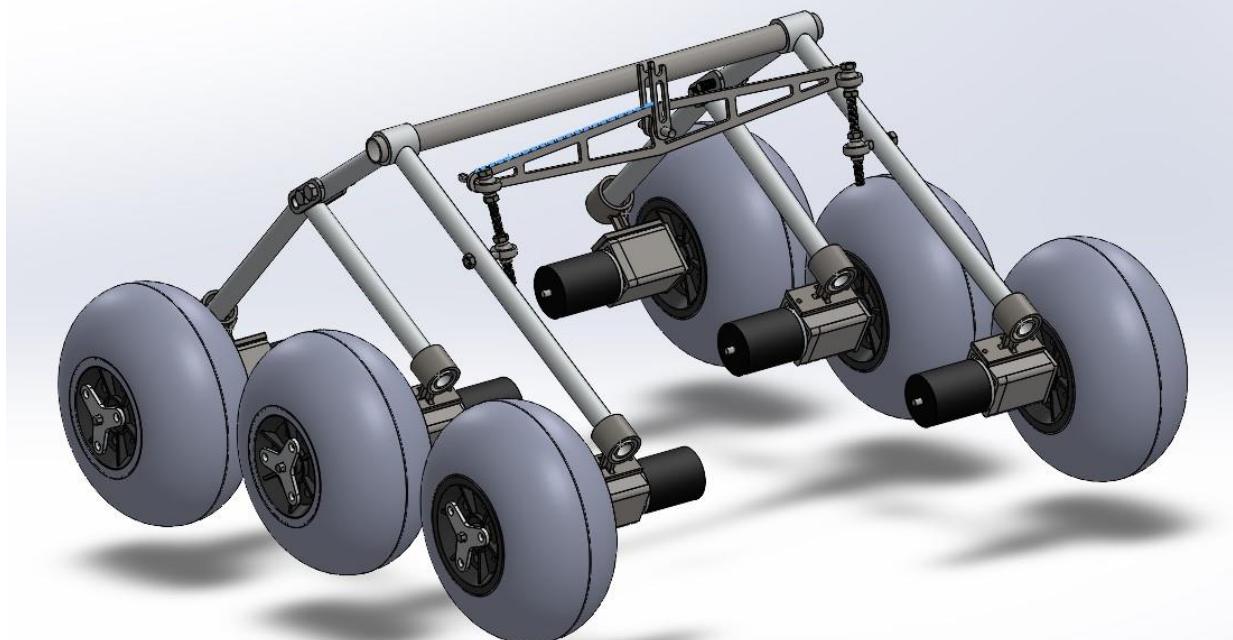


Figure IV-2: SolidWorks model of drivetrain and suspension systems.

IV.B.3.b. Re-design Concept/Solution for Function F1

No major system wide redesign was needed for this function.

IV.B.4. Concept/Solution for Function F2 - Facilitate Expansion

IV.B.4.a. Original Concept/Solution Selected for Function F2

The skeleton frame had several critical advantages over other potential concepts that led to its selection. One: The truss structure resulted in the best strength to weight ratio of potential concepts. Two: The numerous frame struts provide many convenient attachment points for a wide variety of potential future equipment. Three: The open nature of the frame allows easy access to components mounted in the frame's interior.

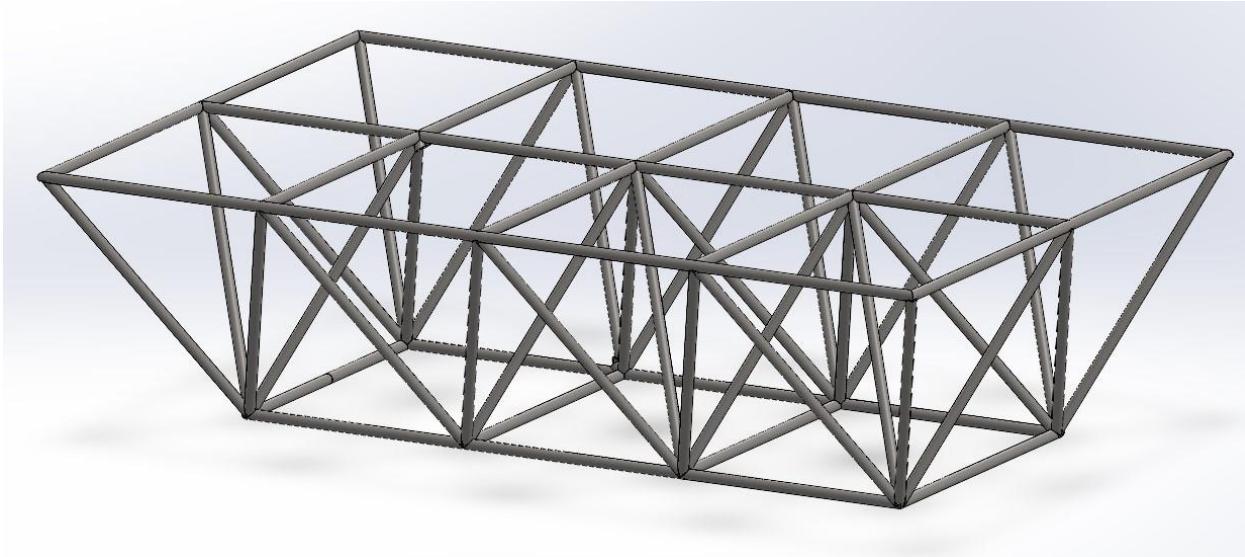


Figure IV-3: Skeleton frame used the primary mounting fixture.

IV.B.4.b. Re-design Concept/Solution for Function F2

No redesigns were needed for this function.

IV.B.5. Concept/Solution for Function F3 – Distribute Power to Electronics

IV.B.5.a. Original Concept/Solution Selected for Function F3

To supply power to all onboard systems of the rover that require it, we utilized a lithium-ion battery, bus bars, and voltage converters. The bus bars facilitate the distribution of power to various components from a central location, while the voltage converters step down the voltage to a level safe for our electrical components. The battery, of course, provides the power to be utilized throughout the rover. The ratings for all these components are displayed in Figure IV-4 in the appendix. Additionally, the decision matrices and AHP tables that were constructed when selecting between concepts that were considered for the power supply and for power distribution can be found in the appendix.

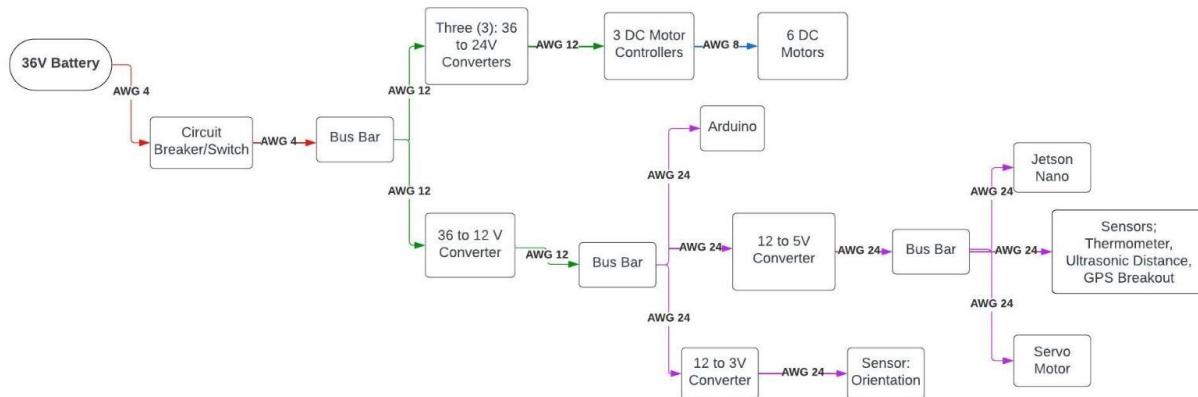


Figure IV-4: Illustration of Original Concept for Function F3

IV.B.5.b. Re-design Concept/Solution for Function F3

When testing the rover, we discovered that when power was supplied from the battery to the entire rover, the motors would 'jump' three times. This means that the motors would briefly run at full speed in short bursts (no more than one second) a total of three times when the circuit breaker/switch from the battery to the bus bar was closed. After troubleshooting, we determined that this occurred because the onboard controllers of the rover did not have sufficient time to initialize. Consequently, the motors ran at full voltage without any control since all components of the rover were powered when the switch was closed. To address this issue, it was decided that we would power the controllers before activating the motors and allow enough time for the controllers to initialize. Below is the circuit we devised to resolve this problem.

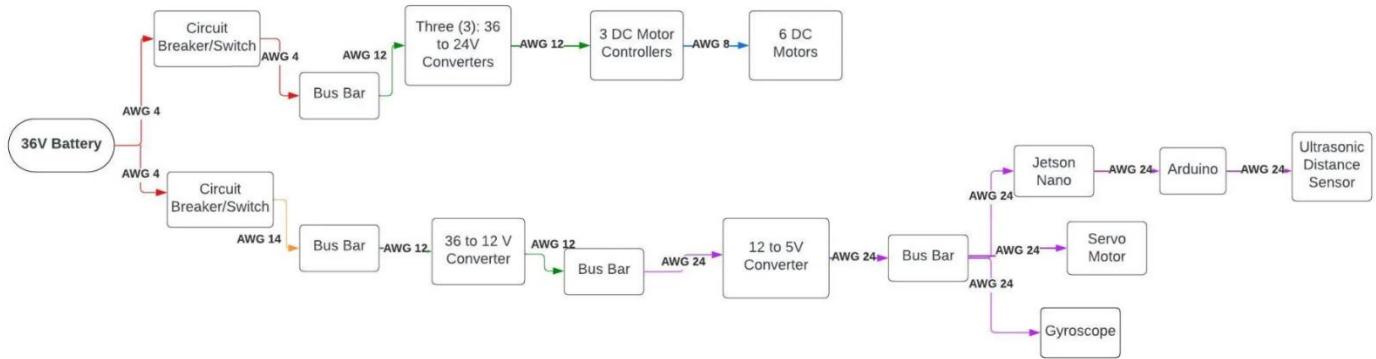


Figure IV-5: Illustration of New Concept for Function F3

IV.B.6. Concept/Solution for Function F4 – Operate Safely

IV.B.6.a. Original Concept/Solution Selected for Function F4

The most important aspect of this function is making sure that overcurrent protection is in place so that components do not overheat and explode or melt. To ensure this is the case for our rover, we determined that we would be using fuses to protect the motors. Fuses were the logical choice and were chosen for the following reasons: they are small and easily integrated into a circuit, they are inexpensive, and they provide guaranteed protection given an unexpected increase in amperage. We are also choosing to implement a digital kill switch in order to immediately stop the rover whenever the user desires.

IV.B.6.b. Re-design Concept/Solution for Function F4

While not really a redesign, this semester we decided to add some more solutions for the function of "operate safely". The first is circuit breakers/switches placed on top of the battery that allow the rover to be easily turned off in the case of emergency and easily turned on without any potential harm to the user. The first switch is used to turn on the computer systems and then, once the computer systems are initialized, the second switch is used to turn on the driving system of the rover. The second solution to improve safety is 3D printed covers for the battery terminals to avoid a scenario where someone could be touching both terminals at the same time.

IV.B.7. Concept/Solution for Function F5 – Transmit Data

IV.B.7.a. *Original Concept/Solution Selected for Function F5*

This function was to be realized using a Wi-Fi module attached to our single-board computer to transmit signals to the controller. The team ultimately decided on Wi-Fi based on a few factors. The required range of the rover lies within the specifications of a Wi-Fi module; a Wi-Fi link is reliable if the rover and controller are the only devices on the signal, Wi-Fi has a low cost of implementation, Wi-Fi has a low average latency, Wi-Fi has built-in security and encryption features, and Wi-Fi has minimal power consumption compared to other options. Transmitting data is an integral function of the project, and Wi-Fi is the best option, given our perimeters and constraints. We anticipated using a Wi-Fi module that connects to our single-board computer via a network interface card (NIC).

IV.B.7.b. *Re-design Concept/Solution for Function F5*

This function did not require a redesign; the team decided to move forward with a Wi-Fi module and the function was accomplished.

IV.B.8. Concept/Solution for Function F6 – Operate Remotely

IV.B.8.a. *Original Concept/Solution Selected for Function F6*

Only two concepts were considered for the type of control selection: a digital controller and a physical controller. The digital controller was selected as our concept that would realize the function because it is less expensive, works well with a Wi-Fi communication system, and is easier to iterate by future capstone teams.

IV.B.8.b. *Re-design Concept/Solution for Function F6*

This function did not require a redesign; the team decided to move forward with a digital controller, and the function was accomplished.

IV.B.9. Concept/Solution for Function F7 – Monitor System Statuses

IV.B.9.a. *Original Concept/Solution Selected for Function F7*

For Function F7, "Monitor System Statuses," we employ various sensors to track critical aspects of the rover. We also make use of the Endure Power app to track the current drawn from the battery, state of charge of the battery, and temperature of the battery. Our other sensors include a temperature sensor for the electronics platform, a gyroscope to measure the slope that the rover is driving on, and ultrasonic sensors to be used in autonomous mode for detecting objects in the path of the rover. The real-time battery monitoring app is crucial to gauge operational range, especially in remote operations, as well as overall life of the battery. Temperature sensors are vital for both safety and preserving the electrical components in hot environments.

IV.B.10. Concept/Solution for Function F8 – Contain Autonomous Features

IV.B.10.a. Original Concept/Solution Selected for Function F8

This function involved implementing an autonomous subsystem with a Single-Board Computer (SBC) that could be utilized in future applications of computer vision and obstacle avoidance. The chosen SBC, NVIDIA's Jetson Nano, excels in GPU performance for computer vision tasks with its 128 CUDA cores. This SBC, in conjunction with a camera and ultrasonic sensors, provides image and depth information for effective object avoidance. More information can be found in the appendix.

IV.B.10.b. Re-design Concept/Solution for Function F8

This concept did not have to be redesigned; however, we did choose a specific algorithm for object detection in avoidance. We implemented a Bug-0 algorithm, which acts as a simple obstacle awareness and avoidance algorithm that will turn away from objects seen by the ultrasonic sensors. We decided on a rudimentary algorithm to allow for future expansion and iteration. This will be expanded upon by future capstone design teams. |

IV.C. System Description/Product Architecture

The rover is driven by six AmpFlow DC brushed motors. The torque is stepped up by a built-in 32:1 gearbox. This torque is then transmitted to the wheels via a driveshaft extender which keys into a hubcap. A series of bolts passing through the hubcap transmits the torque to the wheel. Each wheel system is bolted to the end of one of the leg struts. The legs are made up of a rocker-bogie mechanism, one on each side of the rover. Each side holds three wheels. This system allows for the three wheels on the same side to move independently, allowing the rover to move over rough terrain while maintaining six points of contact with the ground. The legs rotate about an axle mounted in the middle of the frame. To keep the frame from rotating too far about the center axle, a differential stabilizer provides a second point of contact between the frame and the legs. The stabilizer can rotate relative to the frame allowing the left and right wheels to move in opposite directions.

On the electrical side, at its lowest level, the rover will take controller input and power, and be capable of outputting sensor data, a video feed, and executing desired driving and turning maneuvers. The selected lithium-ion battery will supply power to the bus bar, which will, in turn, provide power to the microcontrollers, servo motors, sensors, and DC motor controllers. Subsequently, the microcontrollers will establish communication with the camera, servo motors, and DC motor controllers. Finally, the microcontroller will output sensor data and a live video feed through the WIFI module. Additionally, the DC motor controller will operate the DC motors as directed by the microcontroller, enabling the rover to drive and turn effectively. The functional decomposition can be found in figures IV-6 & IV-7.

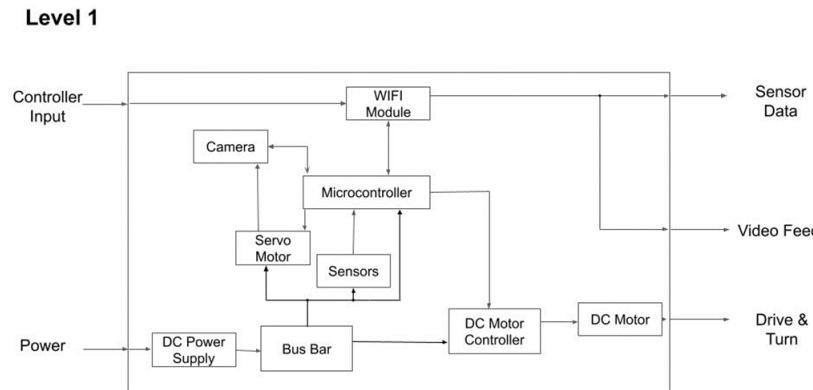


Figure IV6: Level 1 Functional Decomposition

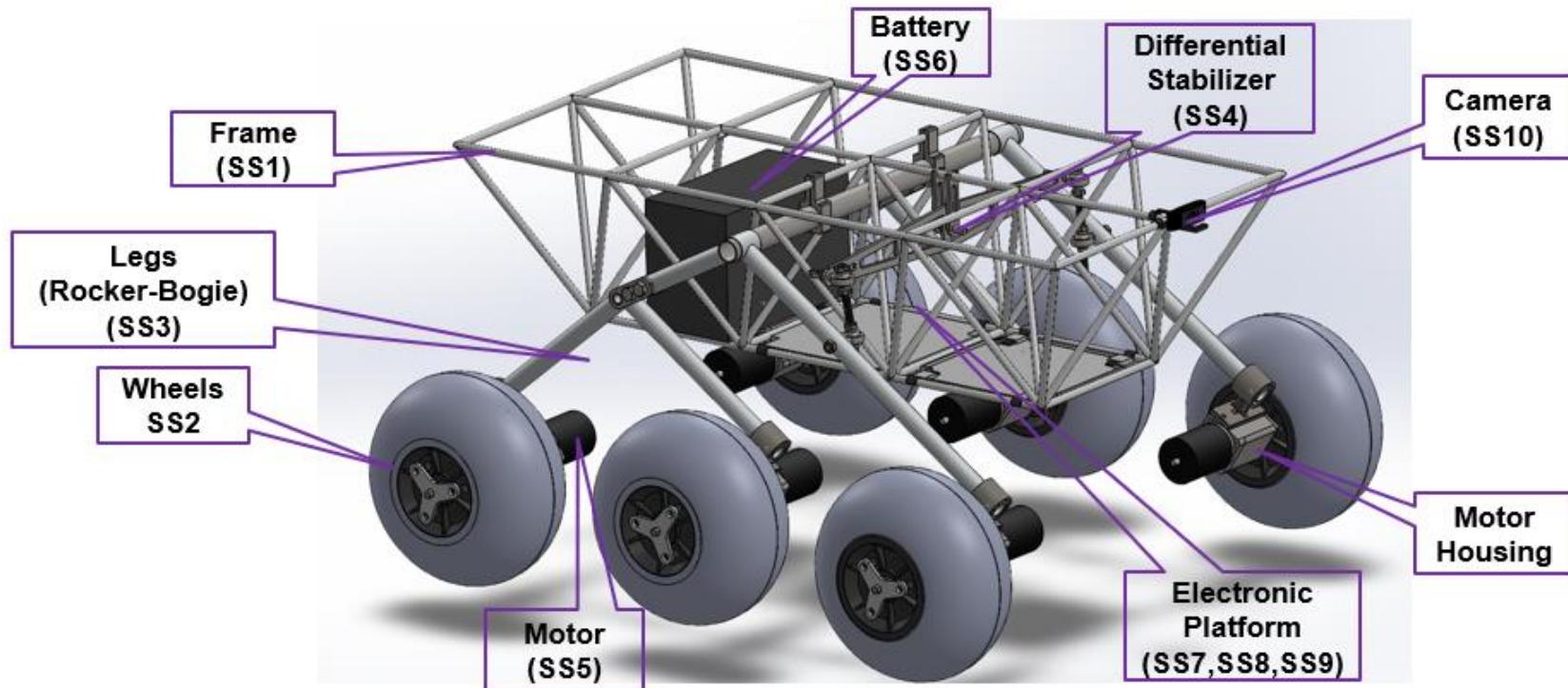


Figure IV-7: SolidWorks model of full rover assembly with key components labeled.

Table IV-1: List of Sub-Systems

#	Name	Brief Description of Sub-System Functionality
SS1	Frame	The main body of the rover that everything will connect to. Supports the electronic components, sensors, and any future equipment.
SS2	Drivetrain	The method for turning the wheels which translates to forward motion.
SS3	Legs	The legs allow the rover to move over terrain while maintaining all-wheel contact with the ground.
SS4	Differential Suspension	Prevents the frame from rotating around the frame axle while still allowing for separate motion of the legs. This insures all 6 wheels can remain in contact with the ground.
SS5	Electric Propulsion	The motors and motor controllers are responsible for the movement of the rover in the desired direction.
SS6	Power Supply	The battery provides power to everything on the rover.
SS7	Power Distribution	The bus bars distribute power to all necessary electrical components onboard the rover
SS8	Controller	How the rover receives input from a separate computer
SS9	Object Detection and Reaction	The algorithm detects objects and provides the rover with a reaction to them.
SS10	Communication	The method by which the rover communicates and sends data collected by the sensors offboard.

IV.C.1. Description of Sub-System SS1 – Frame

The main body of the rover consists of a skeletal frame measuring 4 ft in length and 2 ft in width. It is made from series of 1 ft, 0.25 in, hollow, 1010 steel bars welded into a truss structure. These components were recycled from an older rover project. It strikes a good balance between strength and still maintaining a relatively light weight. The open nature of the frame allows for easy access to internal components and versatile selection of attachment points or future components. Two brackets will be welded to support a central axis that will connect to the legs (SS3). Two more attachments will be welded to the frame to support the differential subsystem (SS4). The camera, ultrasonic sensors, and other electronics will be mounted on custom 3D printed attachments. A part-by-part breakdown will be available in the appendix.

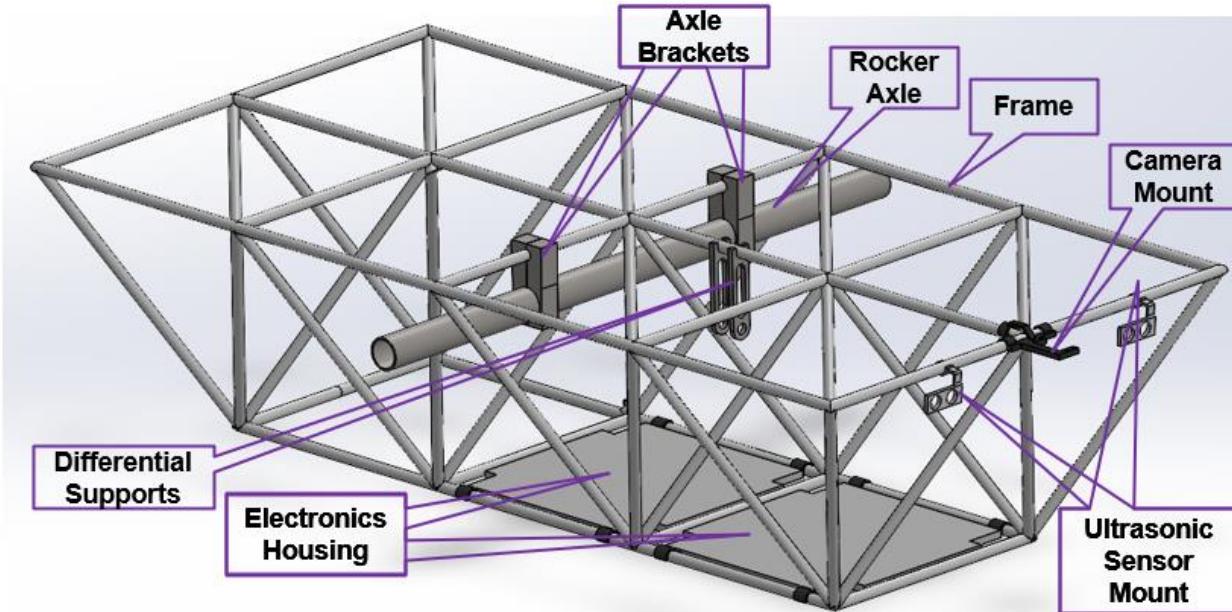


Figure IV-8: SolidWorks Model of rover frame system (SS1).

IV.C.2. Description of Sub-System SS2 – Drivetrain

The rover contains six wheels, each with its own motor. The motors have internal 32:1 gearboxes to step up the torque. Since the built-in driveshaft on the motors is too short to reach the wheel, driveshaft extenders key onto the motors and stretch through the axle to key into the hubcap. The hubcap is bolted to the wheel with three $\frac{1}{2}$ inch bolts which transmit the torque to the wheels. The wheels themselves are 13-inch diameter balloon wheels. These are off the shelf components meant for sandy terrain applications. These wheels were selected as the only terrain specifically requested by our sponsor was sand. Each motor is contained in a square steel housing. The motor housings prevent the motor itself from rotating during operation while providing a contact point to the leg system. Each housing contains two threaded holes through which the whole wheel system can be bolted to the legs. A part-by-part breakdown will be available in the appendix.

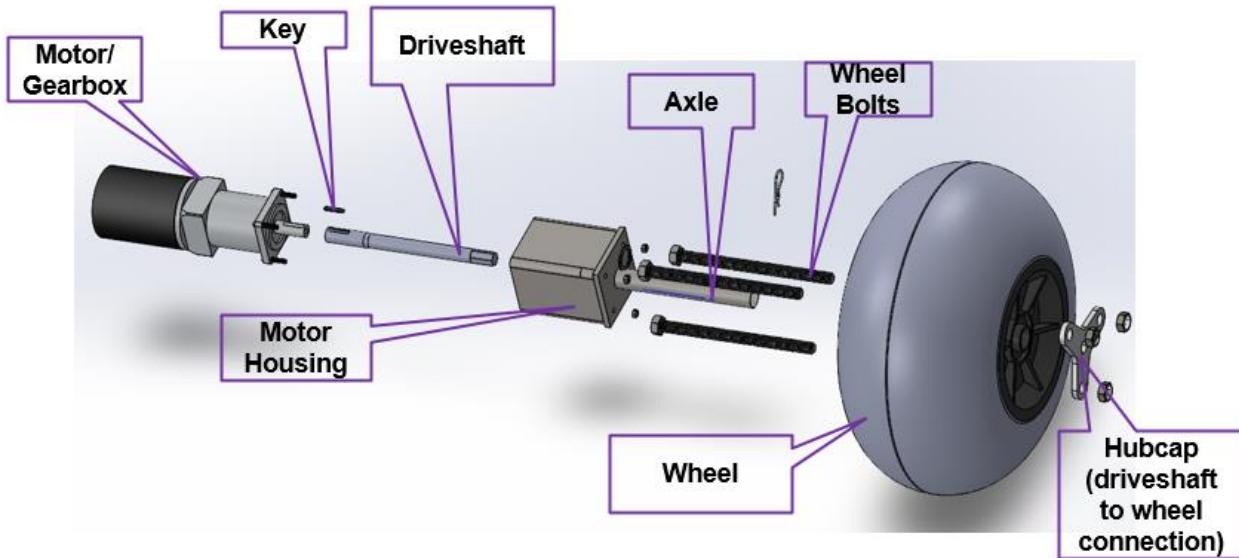


Figure IV-9: Exploded view of rover drivetrain (SS2)

IV.C.3. Description of Sub-System SS3– Legs

The leg system utilizes a rocker-bogie mechanism. This allows all wheels to maintain contact with the ground when driving over uneven. The rocker rotates about the frame axle and holds the front wheel. The bogie holds the two back wheels. It is attached to the rocker and can also rotate freely giving the system an extra degree of freedom. The rover uses two of these systems, one on the left and one on the right, to hold a total of six wheels. The rocker is held to the frame with removable retaining rings. The wheel housing brackets are held in place with a set screw and are capable of rotating when loosened. The wheel systems (SS2) bolted to the wheel housing brackets. A part-by-part breakdown will be available in the appendix.

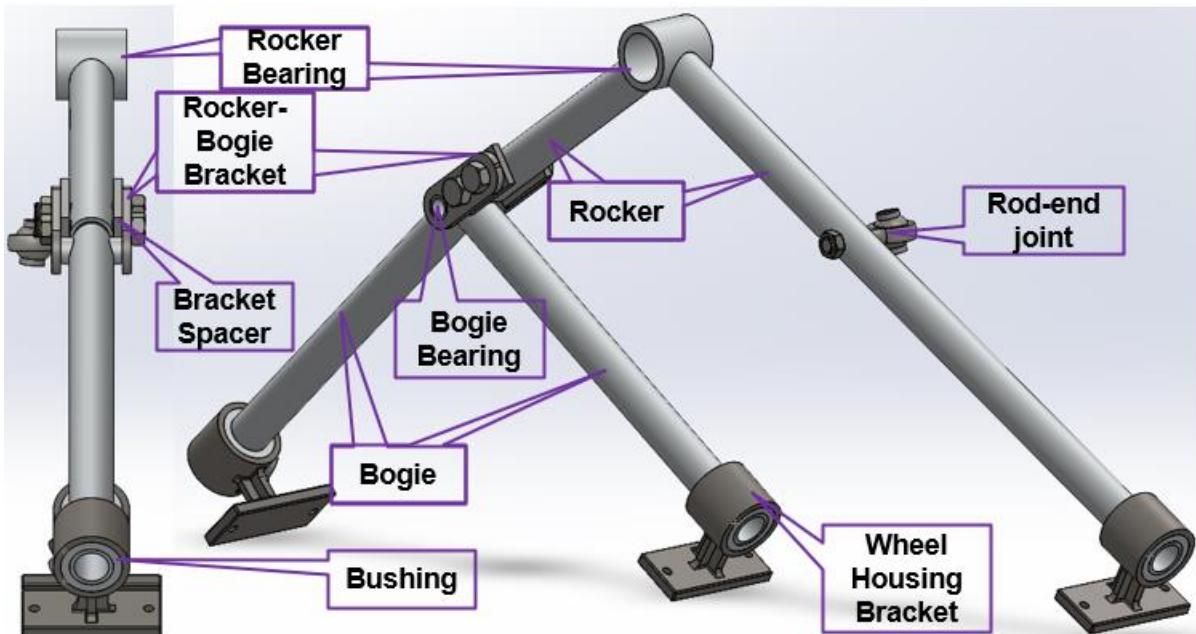


Figure IV-10: SolidWorks model of rocker-bogie leg system with major components labeled.

IV.C.4. Description of Sub-System SS4 – Differential Suspension

The primary function of the differential is to prevent the frame system from freely rotating about the frame axle. It does this by providing a second point of contact between the frame and the legs. If this system were static, it would prevent the legs from rotating in opposite directions. This is necessary when moving over uneven terrain. As such, the differential bar is capable of pivoting back and forth around an axle. Each leg is connected to one end of the bar by a strut. The rod-end joints that connect these components allow the legs to support the frame while still being capable of relative motion. The struts are threaded and held to the joints by nuts. This allows the legs to detach and the system to be adjusted. A part-by-part breakdown will be available in the appendix.

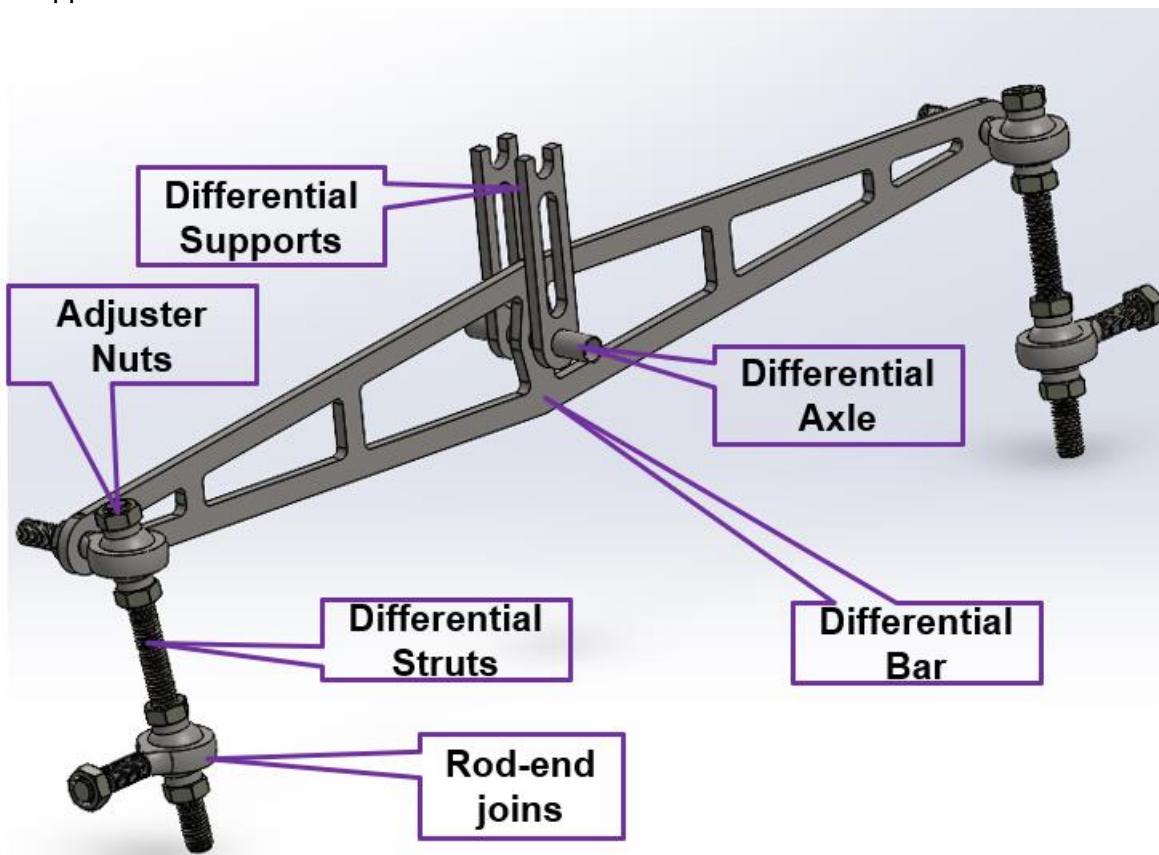


Figure IV-11: SolidWorks model of differential system with major components labeled.

IV.C.5. Description of Sub-System SS5 – Electric Propulsion

This subsystem consists of electric motors, motor controllers and gearboxes that are used to drive the rover in the desired direction at the desired speed. We have three motor controllers and 6 gear motors. Each controller is responsible for two motors with independent control of each motor. The motor controller takes input voltage from the battery as well logic from one of the two onboard microcontrollers. These inputs go through an algorithm and an output voltage is produced that is applied to the motor in order to turn it at the desired speed and in the desired direction. The motor is coupled to the gearbox via a shaft and the gearbox has an output shaft that connects to the wheel in order to make the rover move.

IV.C.6. Description of Sub-System SS6 – Power Supply

For the power supply subsystem, it was determined that a lithium-ion battery would be used to provide power to all electrical components throughout the rover. We identified a 36V 50Ah lithium-ion battery from a company called Enduro as the best possible option for this subsystem. The main advantages of this battery are the fact that it is relatively light in weight (34.14 lbs), while also being small enough in dimension to fit in our rover without protruding from the frame. Even with the power supply chosen being small and light, it still has the nominal voltage and capacity ratings to adequately distribute power to all electrical components over the desired hour of operation. The datasheet for the selected battery can be found in the Appendix.

IV.C.7. Description of Sub-System SS7 – Power Distribution

This subsystem consists of bus bars, DC-DC voltage converters, wires, and fuses that will be used to safely distribute power to all electronic components on the rover. There will be five total DC-DC voltage converters. The five converters are as follows: three 36 to 24V, one 36 to 12V, and one 12 to 5V. We will use no more than 3 bus bars, whose purpose will be to serve as a central point for the connection of our electrical components. The bus bars will be used in combination with the different voltage converters to safely distribute power based on the ratings of the components that need to be powered. More specifically, the subsystem will utilize wires of AWG gauge 4, 8, 12, 14, and 24 as conductors through the system. A flowchart of the power distribution subsystem can be seen in appendix.

IV.C.8. Description of Sub-System SS8– Controller

The Digital Controller subsystem is a pivotal rover component, primarily responsible for its remote operation. This subsystem is built around the idea that any digital device can be a controller. In our testing, we used a laptop, a tablet, and a phone. The controller of choice connects to the rover's Wi-Fi access point, ensuring a secure and responsive communication link for controlling the rover. A web application on the controller features an intuitive user interface, allowing operators to effortlessly send control signals for precise maneuvering and operation of the rover. These signals, processed by the controller, are transmitted wirelessly to the rover's onboard microcontrollers, enabling real-time control and feedback. Designed for compatibility with various digital devices, the subsystem offers flexibility in the choice of hardware for the control system. Additionally, it incorporates safety protocols and redundancy measures to maintain reliable operation under different circumstances. A figure that illustrates this subsystem can be found in the appendix.

Description of Sub-System SS9 – Object Detection and Reaction

The Object Detection and Reaction subsystem is a forward-looking aspect of our rover, primarily focused on recognizing and responding to environmental obstacles. This subsystem leverages a combination of ultrasonic sensors and a camera to detect objects in the rover's path. The ultrasonic sensors provide reliable distance measurements, alerting the system to the presence of nearby objects, while the camera offers a visual perspective, enabling more complex object recognition. This subsystem includes the implementation of a Bug-0 algorithm. This algorithm was mainly chosen for its efficiency in dynamic environments and future expansion. A significant aspect of this subsystem is its developmental nature. It is designed to fulfill its

immediate functional requirements and serve as a foundation for future capstone teams. Providing a robust and adaptable framework allows subsequent teams to further enhance and build upon this technology. This approach ensures that the subsystem remains relevant and evolves over time, adapting to new challenges and integrating advancements in object detection and navigational algorithms. A figure that illustrates this subsystem can be found in the appendix.

IV.C.9. Description of Sub-System SS10 – Communication

The Communication subsystem is a crucial component of our rover, designed to facilitate robust and reliable information exchange between the rover's various systems, its controller, and the integrated sensors. Central to this subsystem is the Wi-Fi module, which acts as the primary conduit for wireless communication. This component establishes and maintains a stable and secure Wi-Fi network through which data and control commands are transmitted between the rover and its controller. This subsystem is engineered to implement efficient and effective communication protocols. The communication subsystem is integrated with the rover's onboard processing unit. This integration allows for the preprocessing and encoding of data before transmission, enhancing the efficiency of data exchange and reducing latency. A figure that illustrates this subsystem can be found in the appendix.]

V. Engineering Analysis and Materials Selection | |

V.A. Engineering Analysis for SS1 - Frame

V.A.1. Types of Eng. Analysis Conducted:

- Reaction force analysis
- Stress analysis
- Failure analysis
- Tipping analysis
- Buckling analysis
- Impact analysis

V.A.2. Eng. Analysis & Materials Selection for SS1-P1 - Frame: Reaction forces, stresses, and failure

| Michael Cannon (ME)

The largest unknown in this calculation is the location of the applied load from additional equipment. The first part of this analysis was to implement worst case scenario assumptions and locate the loading condition which resulted in the highest reaction forces under design loads. These forces were then used to find the greatest stresses and failure limits. The largest reaction force was 1114 N. The greatest stress was 44.5 MPa. This resulted in a safety factor of 6.85. See Appendix XII.E.1.a for details.

V.A.3. Eng. Analysis & Materials Selection for SS1-P1 – Center of Mass

Tristan Hughes (ME)

The center of mass for the rover is important for consideration of tipping. Ideally, the center of mass would be as close to the center of the frame was possible on the X and Y axis, but lower than the frame on the Z axis. An estimate was taken using the SolidWorks model generated, however this number was revisited upon full assembly of the rover. The rover is symmetrical about the Y axis so only the X and Z were considered. It was found that the center of mass on the x axis was 0.65 m from the front of the rover, placing it very near the center, and 0.35 from the ground on the Z axis, placing it well below the frame.

V.A.4. Eng. Analysis & Materials Selection for SS1-P2 – Electronics Platform: Impact

Kyle Pitre (ME)

An impact analysis was conducted to determine the velocity in which the rover could collide with an obstacle before component failures. From observing the integrity of the various components on the rover, it was decided the most likely parts to fail first upon impact would be the electronic components and their platforms. Because of this, the point at which the PLA platform would break was analyzed. It was assumed the platform was dropped vertically with the weight of the electronics secured to it. Under these assumptions, the maximum impact velocity was found to be 10.6 ft/s after applying a factor of safety of 1.5 resulting in a maximum drop height of 1.76 ft. A free-body diagram and more detailed explanation can be found in the appendix (Section XII.D.1).

V.A.5. Eng. Analysis & Materials Selection for SS1-P3 – Frame Axle: Stress, deflection, friction

Michael Cannon (ME)

The axle will use A513 for the material so that it may be weld compatible with the frame. Using worst case scenario assumptions, a largest expected loading state was derived. A MATLAB code was developed to vary the tube size and compare against the yield stress. This was used to select the pipe size. Under maximum loading under the relevant assumptions the factor of safety was found to be 4.8 See Appendix XII.D.1.b for details.]

V.B. Engineering Analysis for SS2- Drivetrain

V.B.1. Types of Eng. Analysis Conducted:

- Stress analysis

V.B.2. Eng. Analysis & Materials Selection for SS2-P2 – Motor Attachment: Stress

Kyle Pitre (ME)

A stress analysis was conducted to determine the safety of the motor attachment component. This component is secured by screwing into the motor housing as well as into the rover legs. Because of this, it was assumed the circular section was fixed (see the free-body diagram in Section XIII.D.2 for reference). It was also assumed the reaction force from the wheel is acting parallel to the wheel axis. Thought this orientation is not likely to occur, it is a worst-case scenario as applying the load in this direction exposes the motor attachment to stresses in the direction where it is the weakest. For the reaction force, the maximum reaction force found during the leg analysis (Section V.C.2) was added with the maximum impact force found during the impact analysis (Section V.A.3). Under these assumptions, the maximum bending stress was found to be 107.4 MPa. This gives a factor of safety of 4 when considering the maximum allowable stress of 445.6 MPa as provided by the manufacturer. These results were also confirmed through the use of an ANSYS model. Similar boundary conditions were applied for this FEA as for the hand calculations, and the convergence of this model can be seen in the appendix (Section XIII.D.2).

V.B.3. Eng. Analysis & Materials Selection for SS2-P4 – Torque on Wheel

Tristan Hughes (ME)

In order to size the motors a torque calculation was performed. This was done by considering three forces on the wheel, the force required to overcome rolling resistance, the force required to climb a grade and the force required to accelerate to a final velocity. Numbers were picked for a 10-degree angle on sand which was considered our worst case, and a factor of safety of 1.15 applied from there. The torque required was found to be much too high for most motors within our price range, so this was stepped with a gearbox. Details are available in section XIII.D.2.

V.B.4. Eng. Analysis & Materials Selection for SS2-P3 – Axle: Stress

Kyle Pitre (ME)

A stress analysis was also conducted on the axle connecting the wheel to the motor housing. This component is hallowed, to allow the motor shaft to run through it and connect to the wheel hub. Designing in this fashion ensures there is no stress experienced on the motor shaft, and solely on the axle. It was assumed the axle is fixed on the side connecting to the motor housing and the force is the maximum wheel reaction force from Section V.C.2. The yield stress for steel, aluminum, and zinc were considered for determining the maximum allowable length of the axle. Under these assumptions, the maximum lengths were found to be 0.99, 0.78, and 0.63 meters for steel, aluminum, and zinc respectively to ensure the axle will not fail under maximum loading conditions. From these results, steel was the choice of material for this component as it provides the largest maximum length while not having a large cost or weight due to the

component's small size. A more in-depth discussion of this analysis can be seen in the appendix in Section XIII.D.2.

V.B.5. Eng. Analysis & Materials Selection for SS2-P5-Motor Attachment Bolts: Shear Failure

Michael Cannon (ME)

Shear failure analysis done on motor attachment bolts. For this analysis, a worst-case scenario was used where the entire weight of the rover was loaded onto a single bolt. The maximum shear force was found to be 4.99 MPa which resulted in a factor of safety of 26.4.

V.C. Engineering Analysis for SS3- Legs

V.C.1. Types of Eng. Analysis Conducted:

- Reaction Force
- Stress analysis

V.C.2. Eng. Analysis & Materials Selection for SS3-P1/P2 – Rocker-Bogie: Reaction forces, bending moments, stress

Kyle Pitre (ME)

The reaction forces acting at each wheel were first determined by assuming the force from the frame and all of its components are concentrated at the joint where the axle is connected on a single side of the rover. Assuming these forces are acting on a single set of legs serves as a worst-case scenario, as these weights will realistically be distributed between both sets of legs rather than just one. We assumed the rover was on level ground for this analysis in order to obtain the angles of the reaction forces, and aluminum was the chosen material in order to reduce weight. The reaction forces were found to be 232.2, 194.7, 97.4, and 127.4 newtons for the front, top, middle, and back locations respectively. A free-body diagram has been placed in the appendix (Section XIII.D.3) for a reference of these force locations.

After solving for the reaction forces, we were able to determine the bending moments acting on the rocker separately by finding the moment on the front and back segments. These were then added together to obtain the bending moment about the middle section. We also assumed the reaction force from the wheels was acting through the center of the cross-section of the leg. This was repeated to find the moment acting on the bogie component of the legs as well. These moments were then used to find the bending stress acting on both the rocker and the bogie, which resulted in bending stresses of 130.8 MPa on the rocker and 42.1 MPa on the bogie.

V.C.3. Eng. Analysis & Materials Selection for SS3-P3 – Rocker-Bogie Bracket: stress, deflection

Michael Cannon (ME)

The material for the rocker-bogie bracket will be A36 carbon steel. The primary factor in this decision is that there will be an extra A36 scrap from the differential bar and so significant costs can be saved by not buying additional material. The bracket was analyzed using FEA in

Ansys. The maximum stress observed in the part was 5.3 MPa. This resulted in a safety factor of 47.

V.D. Engineering Analysis for SS4- Differential Suspension

V.D.1. Types of Eng. Analysis Conducted:

- Stress analysis • Failure analysis
- Kinematic analysis • Finite Element Analysis (FEA)

V.D.2. Eng. Analysis & Materials Selection for SS4-P1 – Differential Bar: Support, Stress, and deflection

Michael Cannon (ME)

The differential bar support was analyzed to determine if it could withstand the reaction forces it was exposed to. The applied load was selected as the maximum reaction force at the differential (see Appendix XII.D.4.a). Due to the Parts complex geometry, and FEA simulation was conducted in Ansys to determine the maximum stresses. The maximum Von-Mises stress was found to be 19.4 MPa. This yields a factor of safety of 12.9. Deflection was negligible. See Appendix XII.D.4.a for detailed assumptions and analysis. The material chosen was A36 steel. This material has good manufacturing characteristics, weld compatibility with the frame, high strength, and lower cost.

V.D.3. Eng. Analysis & Materials Selection for SS4-P2- Differential Axle: Stress and deflection

Michael Cannon (ME)

The differential bar axle was analyzed to determine if it could withstand the reaction forces it was exposed to. Due to the part's simple geometry, the analysis was conducted by hand calculation. The part experienced a maximum bending stress of 38.7 MPa which results in a factor of safety 7.9. The deflection is negligible. The part will be manufactured from an extra strut that will be cut off the frame. This will save on cost and sets the material at 1010 steel. See Appendix XII.D.4.b for details.

V.D.4. Eng. Analysis & Materials Selection for SS4-P3 – Differential Bar: Stress and deflection

Michael Cannon (ME)

The complex geometry of the differential bar necessitated the use of FEA to determine if it could withstand the local maximum reaction forces. The FEA was carried out using ANSYS. The maximum VonMises stress was found to be 93.5 MPa. This yields a factor of safety of 2.7. The maximum deflection was 0.00041 m (0.016 in) which was acceptable. The selected material

was A36 steel. This was done to utilize the same base part as the differential bar support. This saves cost and reduces scrap. See Appendix XII.D.4.c for details.

V.D.5. Eng. Analysis & Materials Selection for SS4-P4 – Differential Strut: Stress and buckling.

Michael Cannon (ME)

The differential struts were analyzed for both tensile and buckling failure. The maximum tensile stress was found to be 1814 psi (12.5 MPa) leading to a factor of safety of 24.8. The critical buckling force was found to be 23400 lbf (104 kN) leading to a factor of safety of 101. The material chosen for this part was 18-8 stainless steel. Primary factors in this decision were part availability, price, corrosion resistance, and strength. See Appendix XII.D.4.c for details.

V.D.6. Eng. Analysis & Materials Selection for SS4P1-P5 – Differential System: Kinematics

Michael Cannon (ME)

A kinematic analysis was conducted on the differential system to ensure that the system could complete its full range of desired motion without surpassing the mechanical limits of individual components. This process was carried out iteratively in MATLAB. The desired system output was a 10 in (0.254 m) difference in front wheel height. The limits of the system were $\pm 13^\circ$ for the differential bar angle and $\pm 32.5^\circ$ for the swivel angle on the rod-end joints. Iterative adjustments to the geometry driven by this analysis led to a configuration that conformed to these standards.

V.E. Engineering Analysis for SS5- Electric Propulsion

V.E.1. Types of Eng. Analysis Conducted:

- Motor Current
- Current Drawn from Source
- Motor controller code flowchart
- Thermal management
- Motor Voltage
- Motor Power
- Voltage difference when turning

V.E.2. Eng. Analysis & Materials Selection for SS5-P1 - Motor

Patrick Maloney (EE) and Tristan Hughes (ME)

In order to properly select the motors, the speed required by the motor was calculated and used with the torque calculation mentioned in SS2. The current drawn from the source was also determined using equations shown in the corresponding section of the appendix in order to size the battery properly. The peak armature current of the motor was calculated and compared to the rated current of the motor. The peak current is above the rated current of the motor, but this only happens a very short period of time. The maximum continuous armature current was calculated and also compared to the rated current of the motor. The maximum armature current is less than the rated current of the motor. The motor voltage was also calculated as well as the electrical and

mechanical power. The motor voltage will be important as this is the output voltage of the motor controller. Electrical and mechanical power can be used to find the efficiency of the motor which can be used to verify that the model is accurate. When picking a specific motor to work with the range of operational temperatures was considered. The motor will naturally generate a certain amount of heat when operating, and as such a heat transfer analysis was performed to find the convergence temperature of the motor for worse case. The resistor method was utilized, and code was written to find the convergence temperature for a wide range of T_{in} . Notably, when this analysis was performed the motor housing had been designed to fit around the entire motor. Since the redesign, the motor housing instead fits around the gearbox only, leaving this analysis a overestimate.

V.E.3. Eng. Analysis & Materials Selection for SS5-P2- Motor Controller

Patrick Maloney (EE)

An algorithm flowchart was created for the logic behind the motor controller code. The difference between the output voltage for a generic turning scenario was also calculated in order to give an idea of expected voltages while turning the rover. Sample code was available from the manufacturer but not really used in actual implementation. The code used in testing and presentation of the rover is available in the corresponding section of the appendix.

V.F. Engineering Analysis for SS6 - Power Supply

V.F.1. Types of Eng. Analysis Conducted:

- Battery
- Thermal Management

V.F.2. Eng. Analysis & Materials Selection for SS6-P1 - Battery

Ethernan Smith (EE) and Tristan Hughes (ME)

Before selecting the lithium-ion battery as the power supply for the rover, a duty cycle was constructed to determine the required capacity. This involved assessing the motor's current draw under various scenarios, analyzed by Patrick. The resulting duty cycle, considering all electrical components during an hour of operation, was modeled in Simulink. The chosen 36V 50Ah lithium-ion battery from Enduro was then analyzed to ensure it could power the rover adequately. The analysis, performed in Simulink, considered the state of charge and voltage over the hour of operation, confirming the battery's suitability. The battery, equipped with a built-in battery management system, meets voltage requirements and avoids excessive discharge. Additionally, a compatible 36V 18A battery charger from Enduro was selected as it is compatible with the battery selected. The datasheet for the battery charger can be found in the appendix, with additional facts presented in the appendix. Extreme temperatures are known to cause the operational effectiveness of batteries to decline, and as such thermal management was taken into consideration when designing this subsystem. A convection heat transfer analysis was performed to see how increasing the freestream velocity of incoming air may help to cool the battery. The rover is planned to move at 1.25 m/s and so the worst case of the freestream velocity was taken

to be that. The manufacturer gives an operating temperature of -4 – 130 F however also recommended to keep it as close to 70 F as possible. As such, it was taken that the surface temperature of the battery to be 70 F and the freestream temperature to be a wide range of temperatures. A MATLAB code was generated that allows this range to be taken easily, and together with a heat generation of the battery, a temperature convergence plot was generated.

V.G. Engineering Analysis for SS7- Power Distribution

V.G.1. Types of Eng. Analysis Conducted:

- Voltage converter selection
- Wire sizing
- Bend radius of wires
- Voltage drop

V.G.2. Eng. Analysis & Materials Selection for SS7-P1 – Voltage Converters

Ethernan Smith (EE)

After selecting the electrical components such as sensors, motors, corresponding controllers, and the battery for the rover, the voltage ratings were utilized to determine the placement and sizes of voltage converters. These voltage and current ratings can be found in the appendix and are marked in bold. A 100A circuit breaker will function as on/off switches from the battery to the motors. It will feed a bus bar rated for 48V and 100A which serves as a connection point from the battery to three voltage converters. Each of the three motor controllers (each controlling two motors) will use a 36V to 24V converter rated for 30A to step down voltage from the bus bar. Additionally, a 70A circuit breaker will be connected to a bus bar. This circuit breaker will operate as an on/off switch specifically for the onboard controllers which need to be powered before the motors are. The circuit breaker will then be connected to a 10A voltage converter, stepping down 36V to 12V. This converter will then feed another bus bar (also rated for 48V and 100A). This bus bar, in turn, will be connected to a 12V to 5V converter and will also directly connect to the Arduino. The 12V to 5V converter will supply power to a final bus bar that distributes power to many electrical components, as shown in the appendix. In total, 5 converters will be used with 3 bus bars to distribute power at the proper voltages throughout the rover

V.G.3. Eng. Analysis & Materials Selection for SS7-P2– Wires

Ethernan Smith (EE) and Patrick Maloney (EE)

After completing the selection of electric components, the wires required to supply current throughout the system could be properly sized. When sizing the wires in the electrical system, we referred to the AWG wire gauge chart pictured in appendix. We primarily based our wire sizes on the chassis wiring column in this chart because the wires are going to be in free space with ambient air. This is as opposed to the power transmission column, as the power transmission ratings are conservative and for wires that are bundled together and do not have access to open air. This criterion applies only to the wiring from the motor controllers to motors. The wire sizes are properly labeled in the flowchart shown in appendix. Summarizing the flowchart in writing and using the chassis wiring column unless otherwise noted, we used 4 AWG for battery to bus bar,

12 AWG from DC-DC converters to motor controllers, 8 AWG from motor controllers to motors (power transmission column), and 24 gauge for all electronic components. The voltage drop for each size of wire was calculated based on the longest length of wire. This is not expected to be a factor but we just want to be sure. The voltage drop is realistically minimal with the largest drop being 4.36% for the servomotor connection. This is an overly conservative estimate as it considers the wire being run half the length of the rover and the servomotor at stall current. For the 8-gauge wires being run through the legs, we wanted to know if it is even possible to bend them through the legs to get to the motor. To do this we needed to calculate their bend radius as well as the maximum available bend radius when its bending at angles through the legs. We found that it is possible to bend those wires through the legs but due to potentially weakening the legs system we decided against it.

V.H. Engineering Analysis for SS8- Controller

V.H.1. Types of Eng. Analysis Conducted:

- Flowchart Analysis
- Programming (HTML, CSS, JavaScript, Python)

V.H.2. Eng. Analysis & Materials Selection for SS8-P1 – Digital Controller

Creighton Cathey (EEC)

In selecting the controller for our rover, we focused on user accessibility and ease of integration. We chose a digital controller operated through a web app, accessible via Wi-Fi once connected to the rover. This decision was backed by an analysis of the app's functionality, using flowcharts and pseudocode, particularly for its implementation in JavaScript. The web-based approach ensures a flexible and user-friendly interface, allowing operators to control the rover seamlessly. This control method aligns with our aim of developing an efficient, intuitive system suitable for various operational contexts. This decision also stems from keeping our sponsor's primary objective in mind – to have a system that upcoming capstone teams can build on in the future. Keeping the controller digital will allow for many iterations and changes without completely refabricating a new controller.

V.H.3. Eng. Analysis & Materials Selection for SS8-P2– Digital Kill Switch

Creighton Cathey (EEC)

Implementing a digital kill switch is a critical safety feature in our rover design. This button, integrated into the digital controller's user interface, serves as an immediate stop mechanism for the rover. Our decision was backed by a flowchart analysis of the manual driving mode, highlighting how the kill switch functions as an interrupt. HTML, JavaScript, CSS, and Python code were developed for the kill switch. This feature not only enhances safety but also provides an essential control tool in case of unexpected issues during rover operation

V.I. Engineering Analysis for SS9- Object Detection and Reaction

V.I.1. Types of Eng. Analysis Conducted:

- Flowchart Analysis
- Programming (JavaScript, Python, C++)
- Compatibility Analysis
- Torque for Servo Motor

V.I.2. Eng. Analysis & Materials Selection for SS9-P1 - Servomotor

Patrick Maloney (EE)

In order to ensure that the servo motor selected is capable of turning the camera at our desired rate, the moment of inertia of the camera had to be calculated. From this calculation the torque required to accelerate the camera was calculated. Because the camera does not provide any opposing torque, this is the only torque that needs to be overcome assuming negligible friction in the motor.

V.I.3. Eng. Analysis & Materials Selection for SS9-P2 – Ultra-Sonic Sensors

Creighton Cathey (EEC)

An analysis of range, power, and compatibility requirements guided our selection of ultrasonic sensors. The chosen sensors excel within a 4-meter range, delivering optimal performance up to 2.5 meters, thus satisfying our minimum 2-meter operational criterion. These sensors are versatile, functioning with 3V and 5V power supplies, making them compatible with various microcontrollers and microcomputers. This adaptability meets our current needs and allows for easy integration and expansion by future capstone teams.

V.I.4. Eng. Analysis & Materials Selection for SS9-P3 - Algorithms

Creighton Cathey (EEC)

The Object Detection and Reaction subsystem, a key component of our rover, employs the Bug-0 algorithm for its efficiency in dynamic environments. This algorithm is integral to processing data from ultrasonic sensors, enabling the rover to identify and navigate around obstacles. Our analysis included flowcharts outlining the algorithm's functionality and code for its implementation. This subsystem is designed to be developmental, serving its immediate purpose and providing a foundational platform for future enhancements by subsequent capstone teams.

V.I.5. Eng. Analysis & Materials Selection for SS9-P4 – Single Board Computer

Creighton Cathey (EEC)

For our rover's computational needs, we selected the Jetson Nano Dev Kit, a single-board computer known for its 128 CUDA cores and Quad-Core Processor with 4 GB Dual Channel memory. This choice was made considering the computer's capability to efficiently handle our object detection and avoidance tasks. We conducted a compatibility analysis with the Wi-Fi module and a power analysis linked to the battery, ensuring that the Jetson Nano can support

current functionalities and future expansions planned by subsequent capstone teams. The Jetson Nano's implementation can be seen in the sensor data flowchart in the appendix, and the complete system code flowchart is based on code running on the Jetson Nano single-board computer.

V.I.6. Eng. Analysis & Materials Selection for SS9-P5 - Microcontroller

Creighton Cathey (EEC)

The Arduino Due microcontroller was chosen for its array of 54 Digital GPIO ports and 12 analog GPIO ports. The need for extensive sensor integration and future expandability influenced this choice. Our analysis included compatibility assessments with our current sensor array and power consumption evaluations with the battery system. The Arduino Due's capabilities meet our current requirements while providing ample scope for future enhancements. The Arduino Due's implementation can be seen in the sensor data flowchart in the appendix.

V.I.7. Eng. Analysis & Materials Selection for SS9-P6 - Gyroscope/Accelerometer

Creighton Cathey (EEC)

To track the rover's velocity and orientation in all circumstances, the Adafruit 9-DOF Absolute Orientation IMU Fusion Breakout was chosen for its comprehensive data output capabilities, including orientation, angular velocity, acceleration, magnetic field strength, and temperature. Our selection process involved verifying compatibility with the Arduino Due and ensuring the voltage requirements matched our system's specifications. Including a 3.3V regulator and logic level shifting for various pins made this gyroscope/accelerometer module a suitable and versatile choice for our rover.

V.I.8. Eng. Analysis & Materials Selection for SS9-P7 - Camera

Creighton Cathey (EEC)

We used a Logitech C920x HD Pro Webcam attached to the Jetson Nano for video feed to our digital controller and object detection and avoidance. This camera, offering Full HD 1080p resolution, was chosen after ensuring its compatibility and assessing its field of view for effectiveness in object detection and avoidance. Our analysis focused on how well this camera would integrate with our system and whether its specifications would meet the demands of our object detection subsystem.

V.J. Engineering Analysis for SS10- Communication

V.J.1. Types of Eng. Analysis Conducted:

- Link Budget Analysis
- Flowchart Analysis

- Programming (Python, Bash, C++)
- Compatibility Analysis

V.J.2. Eng. Analysis & Materials Selection for SS10-P1 – Wi-fi- module

Creighton Cathey (EEC)

Our analysis encompassed link budget analysis and path loss evaluations for the Wi-Fi module. We opted for a dual-band Wi-Fi module connected to the Jetson Nano, which is configured as a Wi-Fi access point. The 5 GHz band, with a link budget of 19.2 dB (9.2 dB with a 10 dB fade/error margin), will be prioritized over the 2.4 GHz band due to its higher data rate of 867 Mbps (compared to 300 Mbps in the 2.4 GHz band). Despite the 2.4 GHz band having a slightly higher link budget of 22.6 dB (12.6 dB with fade/error margin), the superior data transfer rate of the 5 GHz band makes it more suitable for our rover's communication needs. The link budget calculations can be found in the appendix.

V.J.3. Eng. Analysis & Materials Selection for SS10-P2 – Serial Communication

Creighton Cathey (EEC)

For the connection from the Jetson Nano to the Arduino, we are using two serial communication channels, USB to Micro USB. We are using the programming port on the Arduino to send motor controls and the NativeUSB port to send sensor data. With this, we did an analysis on the correct baud rate and the latency expectations.]

VI. Manufacturing & Assembly

[The manufacturing process undertaken by the team can be divided into three main categories: component fabrication, subsystem assembly, and full system assembly. While the process proceeded in this order, it is important to note that there was some overlap as manufacturing issues occasionally came up that had to be dealt with.]

VI.A. Manufacturing

[The majority of the components were manufactured by hand in an effort to cut down on cost. The first component to be manufactured was the frame. The team utilized a section of a frame from an older project to save manufacturing time. The frame was cut to size using a reciprocating saw and an angle grinder. Leftover notches and rough edges were sanded down using the belt grinder. The legs struts, motor housings, axles and axle brackets were all cut out of larger components using the bandsaw. The lathe was used primarily when making the driveshafts. The cutting tool was used to decrease the outer diameter and the drill bit was used to cut out the hole for the motor. The end mill was used frequently for a variety of components. Each driveshaft needed a hex head milled into the end and a slot to fit the key. Cutouts in the axle brackets to fit the frame and axle were drilled on the mill. Holes for the hitch pins, rod-end joints, and motor jousting bolts were drilled on the mill. A circular cutting mill tool was used to cut out the semi-circular shapes on the leg struts. The wheels had to be disassembled and altered to fit the

wheel bolts. Holes were drilled through the rubber wheel core with a power drill. The differential struts were shortened to size with a hacksaw. MIG welding was also utilized for every mechanical subsystem. While the frame was recycled, it was added onto substantially, namely with the central axle that would attach to the legs and the supports for the differential bar. The legs were aluminum, and the MIG welding machine in the shop was utilized for this as well. Due to the student's relative inexperience, spare parts were used for practice welding.

Some components had more complex shapes that would have been expectantly difficult to make by hand. The differential bar, differential supports, rocker-bogie brackets, and motor mounts were all cut from a sheet of steel using the waterjet. The hubcaps were water jetted from an aluminum sheet. All components used to attach the electronics to the frame were 3D printed using FDM printers.

No electrical devices or equipment was manufactured, all of it was purchased and then assembled by subsystem and eventually integrated together. When necessary, wire to wire connections were soldered together to allow for the wiring to reach to all places that power needed to be routed to. Each time a subsystem was constructed, it was tested to ensure that the connections made were sound and that the system operated as expected. |

VI.B. Assembly

Because a majority of components are not permanently attached, there are a number of steps involved with the assembly of the rover. This was a design choice intended to make transportation, repair, and modification easier. Since there are too many components to discuss every assembly step in the main report, this will just be an overview on assembling the major subsystems together.

The drivetrain subsystem (SS2) is attached to the leg subsystem (SS3) through a series of bolts which are hand tightened with a hex wrench. The legs (SS3) are attached to the frame (SS1) by sliding the rocker onto the frame axle and securing it in place with a retaining ring. The differential system (SS4) can be detached from the legs (SS3) by unscrewing the bolt on the rod-end joint. The differential (SS4) can be removed from the frame (SS1) by removing the retaining rings from the differential axle and sliding it out of the differential supports. The electronics are attached to the electronics platforms with strips of Velcro. The battery can be lifted out of the frame after removing the ratchet strap.

The positive battery terminal has two different wires connected to the appropriate step-down converters after going through circuit breakers that act as switches. Wires are run throughout the body of the rover to the appropriate electric devices. Connections were made by either solder or screw terminals. Code was written to control the motors, servomotor, and to send data back to the user. Code was also written to realize the digital controller and to communicate with the rover remotely. Details on the assembly of individual subsystems can be found in the appendix. |

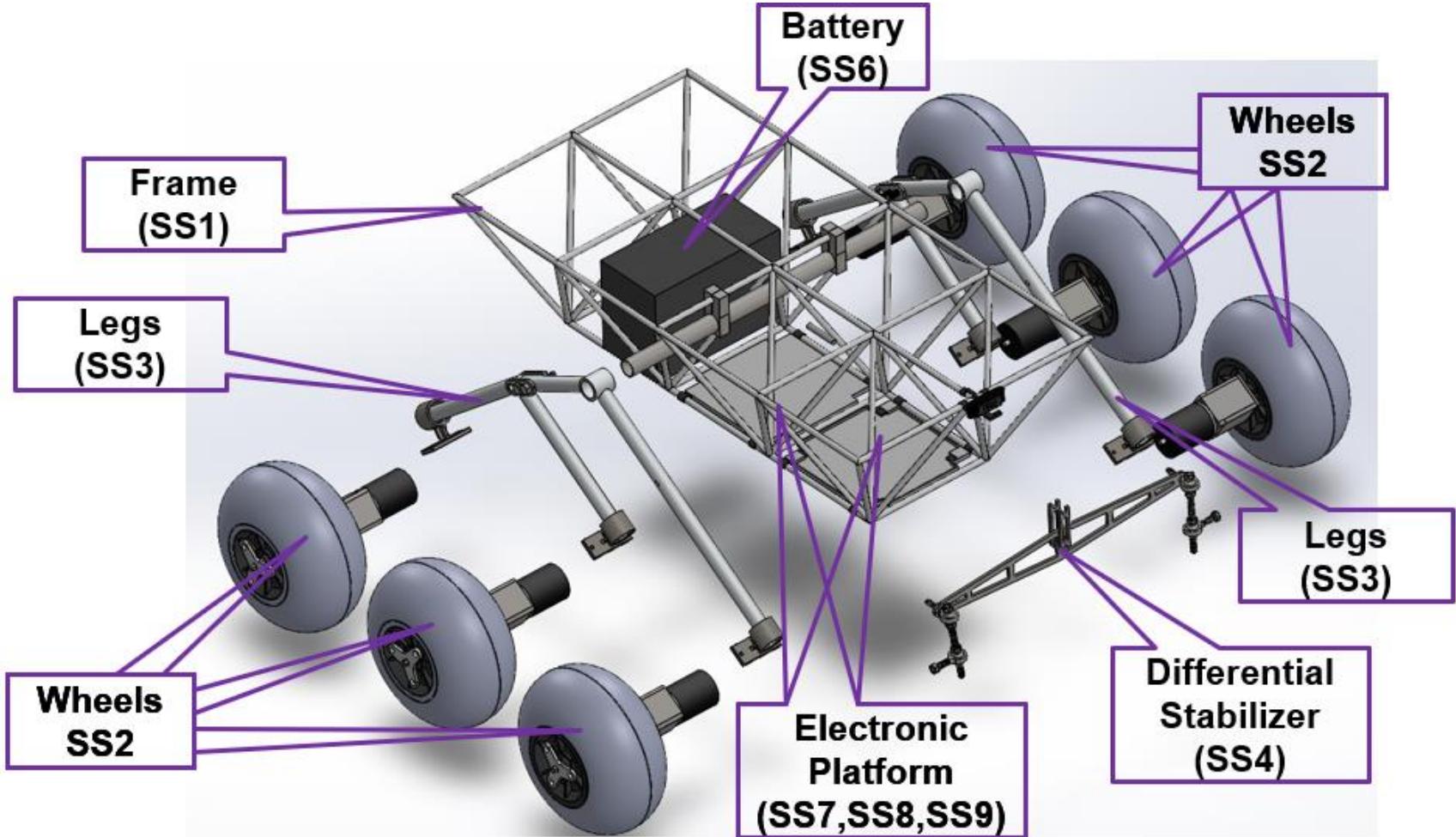


Figure VI-1: SolidWorks model of exploded view of the major rover subsystems.

VII. Safety Considerations

VII.A. Operational Safety

Operational safety is paramount in ensuring the secure handling and functioning of the rover. Key considerations include adhering to the National Institute for Occupational Safety and Health (NIOSH) guidelines regarding the rover's weight, to prevent any strain-related injuries during transportation or manipulation. Additionally, a maximum speed limit of 1.25 m/s is imposed to ensure the rover's movements remain safe and controllable. A digital kill switch is integrated into the controller, allowing for an immediate halt of the rover in case of emergencies. This feature is crucial in preventing accidents and ensuring quick response to unforeseen situations. This also includes a digital stop when the rover loses connection to the controller or if the temperature sensors sense an issue with the battery or motors. The use of fuses in the electrical system is essential to protect against overcurrent, thereby safeguarding the electronic components from damage and reducing the risk of fire. Specific guidelines are in place to prevent pushing or manually maneuvering the rover. This precaution is designed to protect the motor controllers from potential damage due to undue physical force. When handling the rover, it's important to follow these guidelines strictly to maintain the integrity of the electrical and mechanical systems.

VII.B. Safety During Manufacturing, Assembly and Testing

During the manufacturing, assembly, and testing phases, safety takes on a different aspect, focusing on the protection of the team members involved. Adhering to proper Personal Protective Equipment (PPE) standards is imperative. This includes the use of gloves, safety glasses, and appropriate footwear to mitigate the risk of personal injury. The work area will be well-organized and free of hazards, with tools and materials stored safely when not in use. In addition to PPE, there are safety considerations specific to the assembly and testing of the rover. Electrical safety is crucial, particularly when working with the rover's power systems. This includes ensuring all electrical connections are secure and insulated, and that team members are trained in handling electrical components safely. Battery safety is also a key concern, requiring proper handling and storage of the battery to prevent leaks or explosions. During testing, the rover should be operated in a controlled environment, away from uninvolved personnel or potential hazards. This controlled environment allows for the safe observation of the rover's performance and the identification of any issues in a contained setting. Emergency procedures should be established and understood by all team members, ensuring preparedness for any situation that might arise during testing. These safety considerations form an integral part of the project, ensuring the well-being of both the team members and the integrity of the rover. By adhering to these guidelines, risks are minimized, and a safe working environment is maintained throughout the project lifecycle.

VIII. Environmental Impact Assessment

VIII.A. Manufacturing Related Environmental Impact

The rover is made from primarily steel and aluminum parts. Steel production involves resource extraction and emits greenhouse gases but benefits from high recyclability and durability. In contrast, aluminum's production is energy-intensive, contributing to greenhouse gas emissions, and requires bauxite mining. However, aluminum's lightweight nature can improve vehicle fuel efficiency during operation⁽¹⁾. Sustainable manufacturing depends on optimizing materials, recycling, and energy sources to minimize environmental impact throughout a rover's lifecycle. The primary manufacturing techniques used were basic hand tools, welding, and CNC/Waterjet cutting. Of these, welding is the most impactful on the environment. The primary concern lies in the emissions generated during the welding process. Welding operations often involve the use of gases, such as argon and carbon dioxide, which can contribute to air pollution if not properly controlled. CNC/waterjet are power electronically, and as such as relatively little environmental impact outside of emissions create during the power generation.

VIII.B. Operation/Usage Related Environmental Impact

Operation of this rover will require charging the battery within the rover which will be done via energy supplied by a utility company. The energy required to charge the battery and the energy required to charge the digital controller are the only two aspects of this project that will be directly drawn from the utility. This energy will be coming from either a natural gas power plant or a nuclear power plant. No part of this rover will be emitting anything into the atmosphere during operation.

VIII.C. End-Of-Life (Disposal) Related Environmental Impact

At the end of the life of the lithium-ion battery that we choose to use, it is critical that the battery be properly recycled. Mismanagement of lithium-ion batteries at the end of their lives can lead to fires or explosions. The materials in the battery also can be reused to manufacture new batteries.⁽²⁾ The motors and electronics should also be recycled at the end of their life as they are made of metal and can be used to manufacture new devices.⁽³⁾ The structural components of the rover will be made of steel or aluminum. Both materials can be recycled either as components upon disassembly or as raw materials.

IX. Design-Phase Testing

Team 57 conducted no design-phase testing.

IX.A. Design-Phase Testing Objectives, Rationale & Brief Description

N/A

IX.B. Design-Phase Testing Outcomes & Conclusions

N/A

X. Testing, Validation, and/or Implementation

X.A. Testing & Validation of Function F1 - Drive on Various Terrains

Michael Cannon (ME), Kyle Pitre (ME), Tristan Hughes (ME)

The following tests were performed: Suspension test, Turning test, Terrain test. These tests were done during a series of test drives which were used to simultaneously test a wide variety of functions and constraints. Due to a limited testing window, test drives were restricted to location on campus. For a more information on tests and results, see appendix XIII.F.1.

X.A.1. Objective, Rationale & Brief Description - F1

Suspension Test

Objective: The rocker-bogie leg mechanism and differential suspension system are designed to allow the rover to drive over uneven terrain and obstacles while maintaining six points of contact with the ground. The goal of the test is to find the tallest low-radius obstacle that the rover can transverse. Low-radius obstacles refer to obstacles that have radius to height ratios close to one. These obstacles have extremely high slopes and induce the most extreme angles in the rover's legs. Examples include tree roots, curbs, and parking blocks.

Test description: Rover driven over low-radius obstacles of increasing height until suspension limit was reached. Tests were filmed so that any visual data could be extracted and recorded.

Turning Test

Objective: Measure turning rates and radii for the two turning methods on several terrains. The turn-in-place method is only suited for sand, smooth concrete, and tile. The moving turn can be conducted on any surface.

Test description: Rover was directed to execute the desired type of turn on the chosen terrain. Tests were filmed so that turn rates and radii could be calculated at a later time. Some turning tests were conducted with an additional 80 lbf of load attached. These are considered a part of the carrying capacity and can be found in Section X.P.

Terrain Test

Objective: Drive over selected terrain (concrete, grass, gravel, sand) and verify acceptable rover performance. Acceptable performance is defined as: ability of rover to maintain six points of contact with the ground, ability of the rover to make turns, ability to maintain at least 0.5 m/s with motors at 25% of maximum speed, minimal wheel slipping.

Test description: Rover was driven and maneuvered on concrete (outside AMMF and PFT), grass (outside PFT), gravel (outside PFT), and sand (UREC volleyball courts). Tests were filmed so that speed could be determined by tracking wheel movement and other test requirements observed.

X.A.2. Testing & Validation Summary of Results & Conclusions – F1

Table X-1: Suspension test results

Test #	Obstacle Description	Obstacle Height (in)	Pass/Fail
Test 1	Traffic cone on its side	3	Pass
Test 2	Curb	3.75	Pass
Test 3	Curb	4	Pass
Test 4	Parking Block	7	Pass (Bogie tilt angle maxed out)



Figure X-1: Maximum bogie tilt angle reached while overcoming 7-inch parking block.

Table X-2: Terrain test results

Terrain	Average speed (m/s)	All 6 wheels maintain contact?	Noticeable wheel slipping during straight drive?	Turning types possible.
Rough Concrete	0.56	Yes	No	Moving Turn
Smooth Concrete		Yes	No	Turn-in -place, Moving Turn
Gravel	0.56	Yes	No	Moving Turn
Grass		Yes	No	Moving Turn
Sand		Yes	No	Turn-in -place, Moving Turn

Table X-3: Turning test results

Terrain	Turn Type	Turn Radius (ft)	Turn Rate (deg/s)	Direction
Smooth Concrete	Turn-in-place	0	23.4	CW
Smooth Concrete	Turn-in-place	0	25.7	CCW
Sand	Turn-in-place	0	15	CW
Sand	Turn-in-place	0	10.8	CCW
Grass/Concrete	Moving Turn	7	11.3	CW
Sand	Moving Turn	3.5	15	CW

X.B. Testing & Validation of Function F2 - Facilitate Expansion

Michael Cannon (ME), Kyle Pitre (ME), Tristan Hughes (ME)

X.B.1. Objective, Rationale & Brief Description - F2

The sponsor, Dr. Meng, plans for this rover to be a multi-year project. As such, it is expected for the rover to support additional weight for whatever projects future capstone teams may have. We have designed around the rover supporting an additional 35 kg as it drives. The additional load test was applied during the navigate various terrain tests. This additional weight was placed in what is considered the ‘worst’ place on the rover, which is near the front.

X.B.2. Testing & Validation Summary of Results & Conclusions – F2

While the rover was performing the test for the “Drive on various terrain” function, additional payloads were added to the frame of the rover to simulate additional equipment that may be added in the future. This payload was 36 kg, and while it had an impact on the performance of the rover, namely on the speed, the rover was still able to successfully navigate each terrain without issue. These tests were filmed so that speed data and general performance of the rover could be documented. The extra load was also implemented on the slope test which can be found in Section X.T

Table X-4: Extra load test results

Terrain	Able to drive and navigate?	Speed (m/s)	Unloaded Speed (m/s)
Concrete	Yes	0.62*	0.55
Grass	Yes	0.50	0.56
Sand	Yes	0.50	0.56

*motor operating speed increased 5% before test

X.C. Testing & Validation of Function F3 – Distribute Power to Electronics

Ethernan Smith (EE) and Patrick Maloney (EE)

X.C.1. Objective, Rationale & Brief Description – F3

Various tests were performed to ensure proper distribution of power to the electronics. Voltage converters were individually tested by applying a DC power supply voltage to their high end and measuring the output voltage with a multimeter on the low side, repeating the process five times for accuracy. Applied voltages were carefully selected based on anticipated levels for each converter, ensuring thorough validation of their voltage conversion capability. The bus bars were tested by applying a 40V voltage from the DC power supply using clamps. Wires were then connected from the bus bars to previously tested voltage converters, with temporary connections made. Output voltages were measured on the low side of the converters using a multimeter. Since the converters had already been verified to function correctly, matching voltage on the low side indicated proper functioning of the subsystem, including the bus bars.

Torque Test:

Patrick Maloney (EE), Michael Cannon (ME)

To validate the current analysis used to size the battery and motors, a torque test was performed using known weights to get known torques. A motor was connected to a pulley with weights from

15 to 25 pounds in increments of 5 at the end of it and current and speed were recorded as it pulled the weight up. The objective was to compare the current observed to the current from the model to ensure that the battery and motors were sized properly and to ensure that these measurements were being done at least at the speed we set our constraint at.

X.C.2. Testing & Validation Summary of Results & Conclusions – F3

As there were many tests performed for this function as discussed above, I will be putting the results in the appendix for each test conducted. These can be found in appendix.

As seen below, all tests done were completed at a speed greater than the minimum required speed. The predicted steady-state current is consistently slightly lower than the experimental current but still within the uncertainty for all three tests so the analysis is verified. Statistical analysis used to get the uncertainties is in the appendix.

Table X-5: Torque Test Result Data Table

Load (lbs)	Torque (N*m)	Experimental Steady-State Current (A)	Predicted Steady-State Current (A)	Average Motor Speed (rad/s)	Minimum Required Motor Speed (rad/s)
15	4.68	2.6±0.3 2.6±0.3	2.3	9.03	7.57
20	6.23	3.3±0.3 3.3±0.3	3.0	8.02	7.57
25	7.79	3.9±0.4 3.9±0.4	3.7	7.93	7.57

X.D. Testing & Validation of Function F4 – Operate Safely

Patrick Maloney (EE) and Ethernan Smith (EE)

X.D.1. Objective, Rationale & Brief Description – F4

During operation of the rover, there are two methods of bringing the rover to a stop in an emergency: a digital kill-switch and a physical kill-switch. Both need to be tested for functionality in order to ensure safety when driving the rover. The digital kill-switch was tested during malfunction of the code of the control of the rover and the physical kill-switch was tested during normal operation of the rover. The results are seen in the next section.

X.D.2. Testing & Validation Summary of Results & Conclusions – F4

The digital kill-switch passed the test and stopped the rover during a code malfunction. The physical kill-switch also passed the test and stopped the rover while it was driving. The physical kill-switch is also the switch used to turn the rover on and off during normal operation, so it is frequently in use and therefore frequently tested.

X.E. Testing & Validation of Function F5 – Transmit Data

Creighton Cathey (EEC)

X.E.1.Objective, Rationale & Brief Description – F5

The capability to transmit data reliably from the rover to a remote operator is crucial for monitoring the rover's status and controlling its operations in real time. This function is centered around the use of a NVIDIA Jetson Nano equipped with a Wi-Fi module, serving as a hotspot to enable data transmission to a tablet with a digital controller interface. The testing of this function ensures the stability and reliability of the data transmission link under various conditions and distances, allowing for effective rover operation and data feedback. The testing was segmented into three primary categories: connection establishment, data transmission reliability, and range and interference testing. A baseline test initially verified the connection between the tablet and the Jetson Nano over Wi-Fi. Data transmission tests simulated control commands from the tablet to the rover and feedback data from the rover to the tablet under various environmental conditions. Lastly, range and interference tests assessed the reliability of the Wi-Fi connection at different distances and in the presence of potential sources of interference.

X.E.2.Testing & Validation Summary of Results & Conclusions – F5

We performed a Wi-Fi ping Test to test the latency of the Wi-Fi connection from our web server to the digital controller. This was performed using a MacBook Air (M2, 2022) as the controller and we used the ping command in the terminal to test the latency to the Jetson Nano's IP address. The Wi-Fi's average latency was 4.96 ms. The full test can be found in the appendix.

We performed a serial monitor latency test to test the latency of the serial communication from our Jetson Nano to the Arduino. This was performed by writing a program that sent a command to the Arduino and from the Jetson Nano and back to the Jetson Nano from the Arduino. This accounts for processing time on the Arduino as well.

The average latency for the serial connection resulted in an average of 16.08 ms round trip. The full test can be found in the appendix. It is important to note that this would be approximately half if no data needs to be sent back to the Jetson Nano. However, we tested the latency this way to account for object detection latency in sending data from the Arduino to the Jetson Nano.

X.F. Testing & Validation of Function F6 – Operate Remotely

Creighton Cathey (EEC)

X.F.1.Objective, Rationale & Brief Description – F6

In order to verify that the rover can be operated from a controller that is not physically connected to it, we tested the controllers involved and their ability to interact with one another. The main goal of this test was to get the digital controller and the motor controllers to accurately control the speed of the motors. First, the motor controller was tested prior to integration with the digital controller. A series of inputs ranging from full speed forward to full speed reverse were applied from the Arduino IDE to the motors via the motor controller. The speeds were recorded. Then the same test was repeated with input from the digital controller to ensure that the connection was made, and the controllers were accurately tuned. Once the rover was fully assembled, the same test was repeated a third time to ensure that the speed of the rover was acceptable.

X.F.2. Testing & Validation Summary of Results & Conclusions – F6

We performed an operating distance test, in which we tested the rover's remote operation across the entire length of a testing field. The field was approximately 250 feet long, and we successfully operated the rover across the entire length of the field with no noticeable added latency. There was no drop in bars in the Wi-fi signal, and the camera feed still operated at a high framerate. We believe the rover could be operated at a much further distance; however, we ran out of testing space.

We performed testing on the digital controller that included each team member controlling the rover and giving feedback about the ease of operation of the controls. This information was then used to iterate on the controller.

X.G. Testing & Validation of Function F7 – Monitor System Statuses

Creighton Cathey (EEC) and Patrick Maloney (EE)

X.G.1. Objective, Rationale & Brief Description – F7

The various sensors being used to monitor different aspects of the rover during operation must be tested to see if they work correctly or not. The gyroscope was tested for accurate slope by placing it on a phone with a level app open and reading the values output by the gyroscope and comparing to the phone. The temperature sensor (also part of the gyroscope) was tested by exposing it to room temperature air and reading the output of the sensor to ensure that the reading made sense.

X.G.2. Testing & Validation Summary of Results & Conclusions – F7

As shown below, the gyroscope is accurate enough for our purposes as it is being compared to a slope that is only accurate to 2 decimal places. The temperature sensor also is accurate as it consistently output a temperature of 21.5 degrees Celsius in a room that was approximately 21.1 degrees Celsius.

Table X-6: Monitor system statuses test results

Level App Slope (degrees)	Gyroscope Slope (degrees)
5	4.98
10	10.06
12	12.10
15	14.89

X.H. Testing & Validation of Function F8 – Contain Autonomous Features

Creighton Cathey (EEC)

X.H.1. Objective, Rationale & Brief Description – F8

We performed a test of the Bug-0 algorithm by placing the rover on blocks so that the wheels were not touching the ground and then placed objects in front of the rover's ultrasonic sensors to

simulate objects that should be avoided. We placed objects at different distances to see if the rover would accurately react. This information was used to iterate on the algorithm for how far the detection distances should be set and how fast the wheels should turn. We did not get to a point where we were comfortable testing the object avoidance in a field test.

We performed a test of the ultrasonic sensors to see at what range they would accurately detect objects. To do this, we connected the two ultrasonic sensors to the Arduino via a breadboard and placed objects at varying distances from the sensors. This information was used to iterate on the code. We found that the sensors send outlier information in some of our tests, so we changed the code to take multiple readings every second and take the average of those readings after getting rid of the highest and the lowest outliers, and this gave us the most accurate readings.

X.H.2. Testing & Validation Summary of Results & Conclusions – F8

X.I. Testing & Validation of Qualitative Constraint Q1 - [Portability]

[All Team Members]

X.I.1. Objective, Rationale & Brief Description - Q1

The objective of this test was to ensure that it is possible for a number of people between four and six to pick up the rover without violating OSHA's lifting limits per person. The rover was weighed upon its full assembly, and it was lifted to ensure it is not too heavy to be lifted by four to six people. Additionally, a long-distance transportation test was conducted, where the rover was secured and transported in the bed of a pickup truck. |

X.I.2. Testing & Validation Summary of Results & Conclusions – Q1

Rover was successfully carried by two people and transported by pickup truck. |

X.J. Testing & Validation of Qualitative Constraint Q2 - [Durability]

[Kyle Pitre (ME)]

X.J.1. Objective, Rationale & Brief Description - Q2

The primary way we plan on testing this is by using strain gauges to ensure that the legs and frame were durable enough to bear the maximum expected load. Strain gauges were placed along the wheel axle under the maximum expected loadings to ensure that deformation is within expected parameters. There was also an impact test conducted on the electronics platform. This was done by dropping the platform at increasing heights with a spare motor controller secured to the platform until failure. Additionally, the rover underwent a visual inspection for damage and wear. |

X.J.2. Testing & Validation Summary of Results & Conclusions – Q2

See Section X.P.2 for strain gauge testing results.

The maximum drop height was found to be 1.5 ft with an impact velocity of 8 ft/s.

Some wear found on driveshafts and internals of wheel. All components still appear functional.

X.K. Testing & Validation of Qualitative Constraint Q3 – Ease of Operation

Creighton Cathey (EEC), Tristan Hughes (ME)

X.K.1.Objective, Rationale & Brief Description – Q3

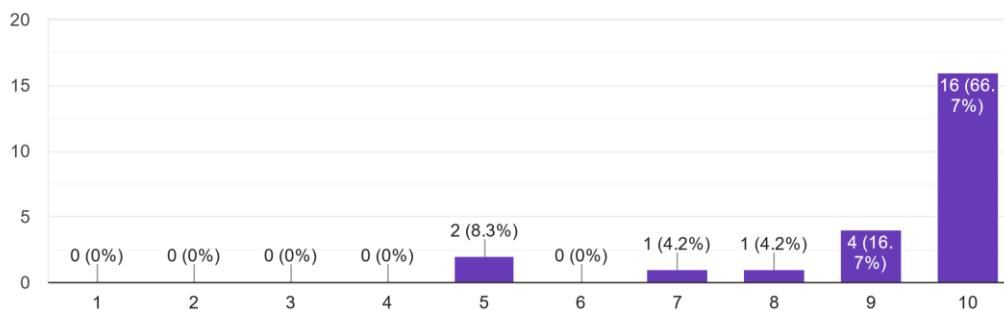
The rover must be easy to operate in order for it to be useful to anyone. We tested that someone with no prior experience with the rover can easily operate it. A poll was taken to see how intuitive the rover's setup and user interface is to random users.

X.K.2.Testing & Validation Summary of Results & Conclusions – Q3

Figure X-2: Test Result Data Table

How intuitive are the controls for the rover?

24 responses



X.L. Testing & Validation of Qualitative Constraint Q4 – Aesthetically Pleasing

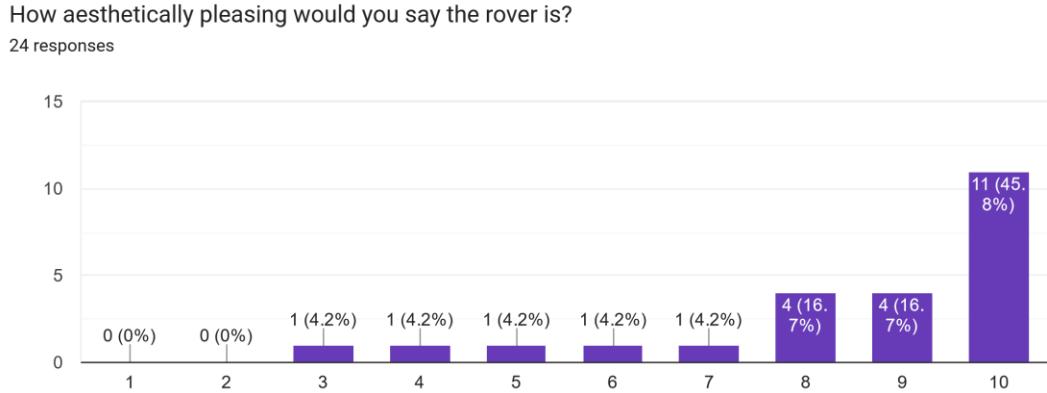
Tristan Hughes (ME)

X.L.1.Objective, Rationale & Brief Description – Q4

To test how aesthetically pleasing the rover is to the general population, we conducted a random survey asking questions about the appearance of the rover.

X.L.2. Testing & Validation Summary of Results & Conclusions – Q4

Figure X-3: Test Result Data Table



X.M. Testing & Validation of Quantitative Constraint M1 - [Weight]

Tristan Hughes (ME)

X.M.1. [Objective, Rationale & Brief Description - M1]

The rover is expected to be 75 kg after manufacturing has been completed. The weight of the rover was set to under 100 kg for portability and for operation of the motors/battery to be within expected parameters. The rover was weighed upon final assembly to measure that it remained within this constraint.

X.M.2. Testing & Validation Summary of Results & Conclusions – M1

The weight of the rover was found to be 77.5 kg after fully assembled.

X.N. Testing & Validation of Quantitative Constraint M2 - [Latency]

Creighton Cathey (EEC)

X.N.1. [Objective, Rationale & Brief Description - M2]

This constraint aims to ensure minimal latency in the Wi-Fi communication between the digital controller and the rover's single-board computer. This testing aims to measure the delay between inputs from the controller and outputs executed by the rover's systems. The tests were conducted under various conditions to simulate real-world operating environments, including different distances between the controller and the rover, and in the presence of potential sources of signal interference.

X.N.2. Testing & Validation Summary of Results & Conclusions – M2

We performed latency testing for the entire communication system and added up the latency of the subparts to get the complete system latency. The average network latency was found using a wi-fi ping test and was found to be 4.96 ms. The web server processing latency was found by timestamping before and after processing commands and was found to be 1.23 ms. The roundtrip serial communication latency between the Jetson Nano and Arduino was an average of 16.08 ms. All of these together gave us an average latency of 22.27 ms, well below our 50 ms latency constraint.

X.O. Testing & Validation of Quantitative Constraint M3 - Budget

Kyle Pitre (ME)

X.O.1. Objective, Rationale & Brief Description – M3

The budget for this project was set at \$6,000. The purchases for all components, manufacturing, and testing were tracked to ensure this maximum cost was not exceeded.

X.O.2. Testing & Validation Summary of Results & Conclusions – M3

See Section XI.B for a more detailed breakdown of the project spending.

X.P. Testing & Validation of Quantitative Constraint M4 – Carrying Capacity

Kyle Pitre (ME)

X.P.1. Objective, Rationale & Brief Description – M4

The rover is required to be capable of carrying an additional 35 kg for future teams to add equipment. This was tested by placing this weight in the least ideal location while conducting the drive on various terrains test. Strain gauge tests were also performed on the wheel axle, as this component gave the lowest factor of safety during the stress analyses performed, to ensure this component can withstand these loads. This constraint was also validated in the field by the extra load test for function F2 (see section X.B)

X.P.2. Testing & Validation Summary of Results & Conclusions – M4

Table X-7: Strain Gauge Test Results

Weight	Strain	Stress (psi)	Deflection (in)
0	0.0000	280	0.001
5	0.0001	1959	0.005
10	0.0001	2799	0.008
15	0.0001	3799	0.010
20	0.0002	6438	0.018
30	0.0003	7837	0.021
35	0.0003	8677	0.024

40	0.0004	9797	0.027
50	0.0005	12596	0.034

X.Q. Testing & Validation of Quantitative Constraint M5 - Speed

Michael Cannon (ME), Patrick Maloney (EE)

X.Q.1. Objective, Rationale & Brief Description – M5

The rover is expected to be able to navigate various terrains, and speed is an important consideration in navigation. The constraint set at the beginning of the project is a minimum top speed of 1.25 m/s. To test this, the rover was driven a known distance and timed while doing so. The speed was then calculated using the distance traveled and the time taken to travel.

X.Q.2. Testing & Validation Summary of Results & Conclusions – M5

The average speed between the two trials was found to be 1.355 m/s, over satisfying our requirement.

Table X-8: Speed test results

Trial #	Terrain	Average Speed (m/s)
1	Smooth Concrete	1.39
2	Smooth Concrete	1.33
3	Rough Concrete	1.35
4	Grass	1.35
Average	-	1.355

X.R. Testing & Validation of Quantitative Constraint M6 - Dimensions

Tristan Hughes (ME)

X.R.1. Objective, Rationale & Brief Description – M6

The rover is expected to fit within a 1.25 m^3 box as part of the portability qualitative constraint. Once the rover was fully assembled the dimensions were measured.

X.R.2. Testing & Validation Summary of Results & Conclusions – M6

The dimensions of the rover were found to be 1.25 m X 1.25 m X 0.625 m once fully assembled.

X.S. Testing & Validation of Quantitative Constraint M7 – Operation Time

Ethernan Smith (EE)

X.S.1. Objective, Rationale & Brief Description – M7

The rover was intended to be able to operate for an hour without a battery recharge. As the output of the variable load tester was not large enough to have any impact on the size of our relatively large battery, we instead operated the rover for as long as possible on various terrains.

The state of charge and current drawn were then monitored utilizing the built-in battery management system via the Enduro Power application, to verify the goal for operation time was met.

X.S.2. Testing & Validation Summary of Results & Conclusions – M7

In total, we conducted three separate extended full rover tests, each of which provided the opportunity to monitor the battery and observe how the state of charge was impacted by traversing various terrains. These tests lasted 37, 21, and 15 minutes, respectively. Unfortunately, we were unable to test for a full hour at any time due to various failures that prematurely ended the extended tests. However, the battery was only briefly charged once between the second and third tests. At the conclusion of the final extended test, the rover operated for a total of 4405 seconds (approximately 73 minutes), and the state of charge dropped from 95% to 81%. Each of these tests individually can be found in Appendix, providing more detail about the terrains the rover operated on, the currents drawn from the battery while traversing these terrains, and the reasons for prematurely ending the tests.

Table X-9: Operation time test results - Overall

SOC Initial	SOC Final	Time total (s)	Time Total (m)
95	81	4405	73.42

X.T. Testing & Validation of Quantitative Constraint M8 - Slope

Michael Cannon (ME)

X.T.1. Objective, Rationale & Brief Description – M8

Objective: Test the rover's ability so climb slopes. Minimum required slope of 5degrees was selected by the sponsor. Test slope of 20 degrees selected by team.

Test description: The testing location was selected was the Mississippi River levee which has a slope ranging from 15 to 20 degrees. The rover was driven directly up the side on a path of uneven dirt and short grass. The test was filmed to that the rover's average speed could be calculated. This test was also conducted with the extra 35 kg load attached and the motors set to 25 percent of maximum speed.

X.T.2. Testing & Validation Summary of Results & Conclusions – M8

Table X-10: Slope test results

Slope (degrees)	Terrain	Load (kg)	% of Max motor speed	Average rover speed (m/s)
15-20	Uneven dirt and short grass	35	25	0.33

X.U. Testing & Validation of Quantitative Constraint M9 – Detection Distance

Creighton Cathey (EEC)

X.U.1. Objective, Rationale & Brief Description – M9

This is to ensure the rover's ultrasonic sensors and camera system can detect objects at specific distances. This testing aims to verify that the sensors and camera meet or exceed the required detection range specifications under various conditions. Detection distance testing involved measuring the maximum and minimum distances at which the rover's ultrasonic sensors and camera can reliably detect objects. The tests were conducted in a controlled environment to ensure accuracy and repeatability. Different sizes, shapes, and materials of objects were used to simulate a range of potential obstacles the rover might encounter.

X.U.2. Testing & Validation Summary of Results & Conclusions – M9

We tested the ultrasonic sensors to see at what range they would accurately detect objects. To do this, we connected the two ultrasonic sensors to the Arduino via a breadboard and placed objects at varying distances from the sensors. This information was used to iterate on the code. We found that the sensors send outlier information in some of our tests, so we changed the code to take multiple readings every second and take the average of those readings after getting rid of the highest and the lowest outliers, and this gave us the most accurate readings. We found that the optimal distance to get accurate measurements was up to 2.5 m, which is within our 2 m constraint.

XI. Project Management

XI.A. Schedule and Milestones

Last semester, we started with concept generation and selection, which led to defining all preliminary systems by the beginning of October. We then moved into conducting the required analyses for the remainder of the fall semester to prove our chosen designs were feasible. All major components for the project were then ordered by the beginning of February. This allowed us to confirm the costs for buying the required parts, so a budget could be established for using the various machines in the shop. All individual components were manufactured by March 20. As the individual components were completed, they were also tested to ensure they work as expected. This gave us sufficient time to test and assemble all sub-systems together by April 1. Once the rover was fully assembled, we began full system testing to identify any modifications that need to be made. This gave us the month of April to be solely for making these modifications to ensure a fully operational model by April 30. |

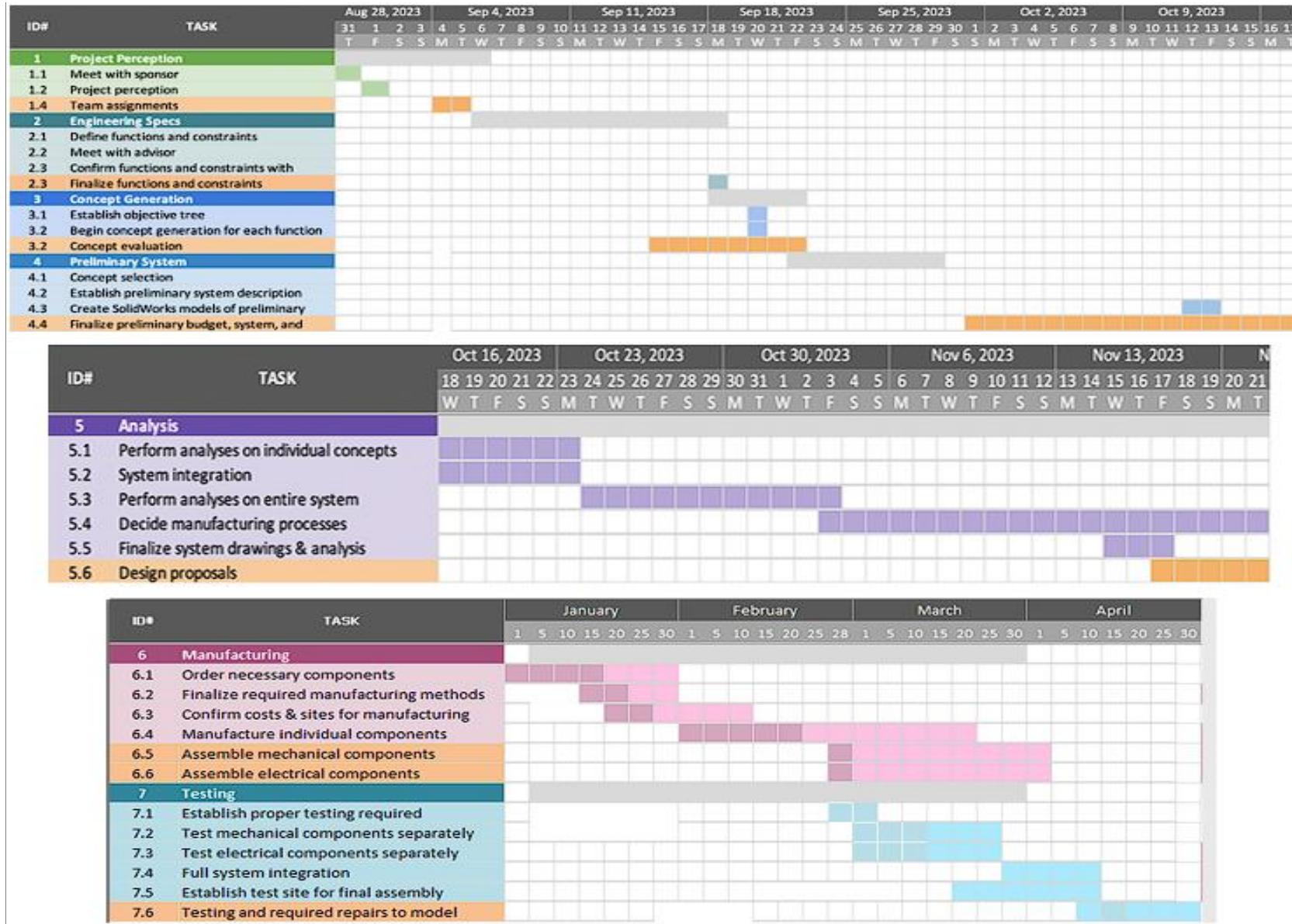


Figure XI-1: Gantt Chart with Project Time-Line

XI.B. Budget

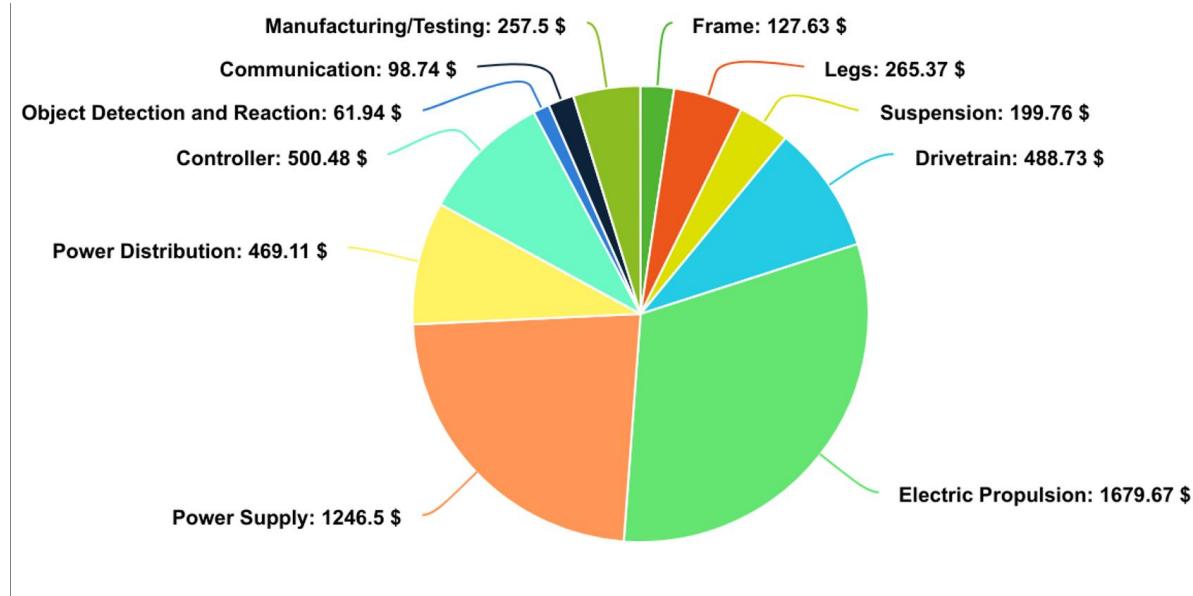


Figure XI-2: Pie Chart of Budget

The actual budget by major expense category is above. More details in Appendix XIII.H.2 |

XII. Summary and Conclusions

With the allotted budget of \$6000, team 57 was able to design, manufacture, and assemble a mobile, durable, but simultaneously lightweight rover with room for potential expansion for future capstone team. It fit within 1.25 cubic meters, weighs 77 kg, and is equipped with a 6-wheel drivetrain and rocker-bogie suspension mechanism that ensures durability in rough terrain. The rover is capable of driving at a maximum speed of above 1.25 m/s for well over 60 minutes while holding 36 kg of extra weight. It also can traverse slopes of up to 15 to 20 degrees. The average latency between the rover's onboard Wi-Fi module and the digital controller is 22.27 ms and the ultrasonic sensors can detect objects at distances up to 2.25 m. The skeletal frame is lightweight while still being very strong and gives several points for future teams to expand upon it. The rover is also equipped with a camera and multiple sensors, such as temperature, gyroscopic and ultrasonic, the latter of which allows the rover to detect objects in its path while in "autonomous" mode. The rover also has a Jetson Nano and Arduino that allow for higher levels autonomy to be added in the future.

A compilation of footage taken during test drives of the rover can be found here:
<https://www.youtube.com/watch?v=AAzgz0E7lkk> |

XIII. Appendix

XIII.A. Quality Function Deployment (QFD) - HoQ

The House of Quality shown in Figure XIII-1 depicts the importance of the different functions for the rover and the means to accomplish these goals. Weights for these functions were established by assigning values ranging from 0 to 9 to the relationship between each of the functions. Higher values correlate to functions that are strongly related and vice versa. Each value in the summation column was divided by the sum of all the summations from each row to obtain the percentage representing the importance weight for each function.

Quantitative constraints were established to give physical values that could be measured to determine the success of the project. The sponsor established the top two quantitative constraints that the entire project should remain under a budget of \$6,000, and the rover should be capable of driving up a 5-degree slope. Constraints for the dimensions and weight were derived through researching previous Mars rover designs. It was decided to create a model at approximately half the scale of the Perseverance and Curiosity rovers when considering dimensions, which is how the target length, width, and height of 1.25 meters was determined. This would also allow the rover to fit into elevators and through doors if it is required to be worked on in labs at Patrick F. Taylor. Considering there will not be additional equipment included, it was decided an appropriate maximum weight of 75 kilograms was an appropriate amount when considering the dimension scale down. This would allow for 25 kilograms of additional equipment to be added by future teams in order to remain within safety regulations stating 23 kilograms per person carrying an object is allowed. This is represented by the carrying capacity constraint. The speed of the rover should be a minimum of 1.25 m/s. Lastly, a latency of 50 milliseconds was established for the transmission of data in order to safely have time to shut down the rover if needed.

When increasing the maximum slope, the rover is capable of navigating, a negative response will result from the remaining constraints. This would require more money to be spent for upgrades, the operation time to decrease due to increased power requirements, and the dimensions, weight, and speed to all require reductions. Similarly, reducing the price of the rover would require negative responses of the other constraints. The rover would have to be smaller to reduce material and with less expensive components like batteries and wheels. As the weight of the rover is reduced, the operation time will increase, which results in a positive correlation between these constraints. But this will have an opposite effect on the maximum speed of the rover because as the speed increases, the operation time will decrease due to more power being required. The more weight is reduced, the more allowable weight for extra components and the faster the rover will be capable of moving. Lastly, there is a negative relationship between carrying capacity and speed. The rover can be designed to carry larger loads or move faster, but it cannot accomplish both of these goals at the same time. A middle ground between the two will need to be decided. |

Title: Project #57 - Planetary Exploration Rover

Author: _____

Date: _____

Notes: _____

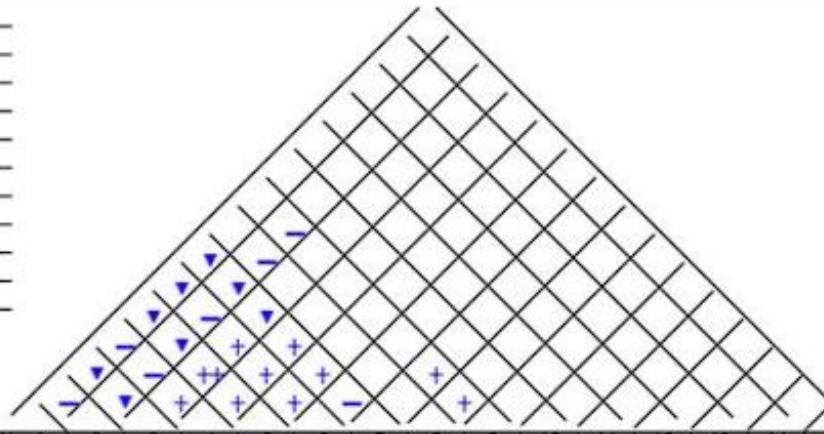
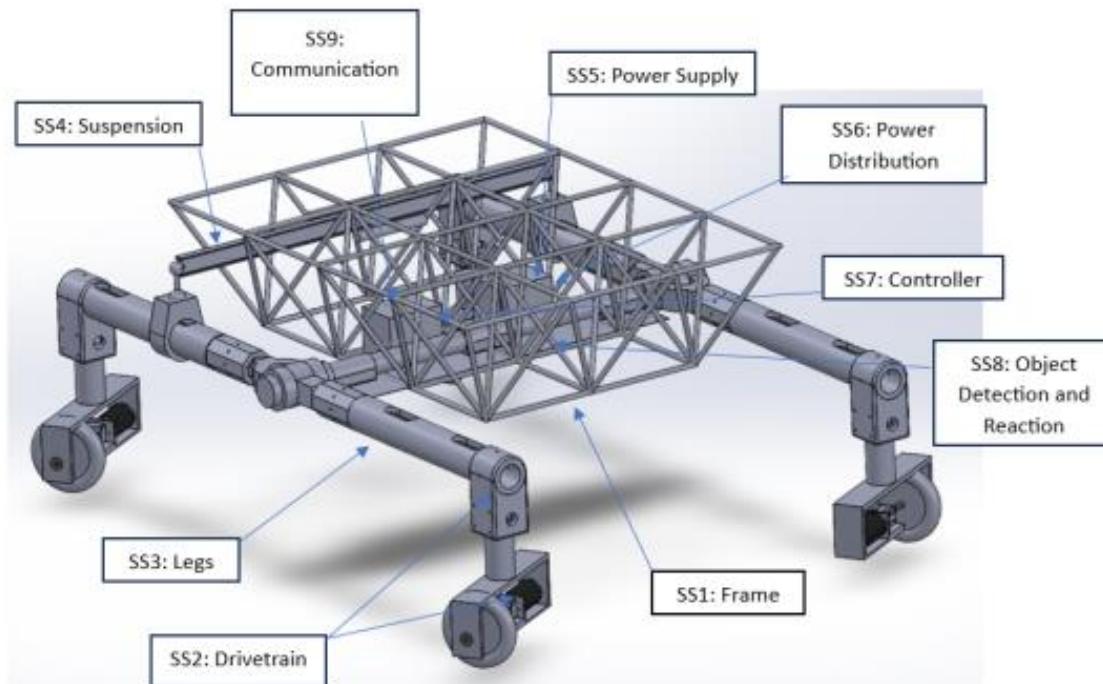


Figure XIII-1: The House of Quality for the project.

XIII.B. Concept/Solution Evaluation and Selection Supplement

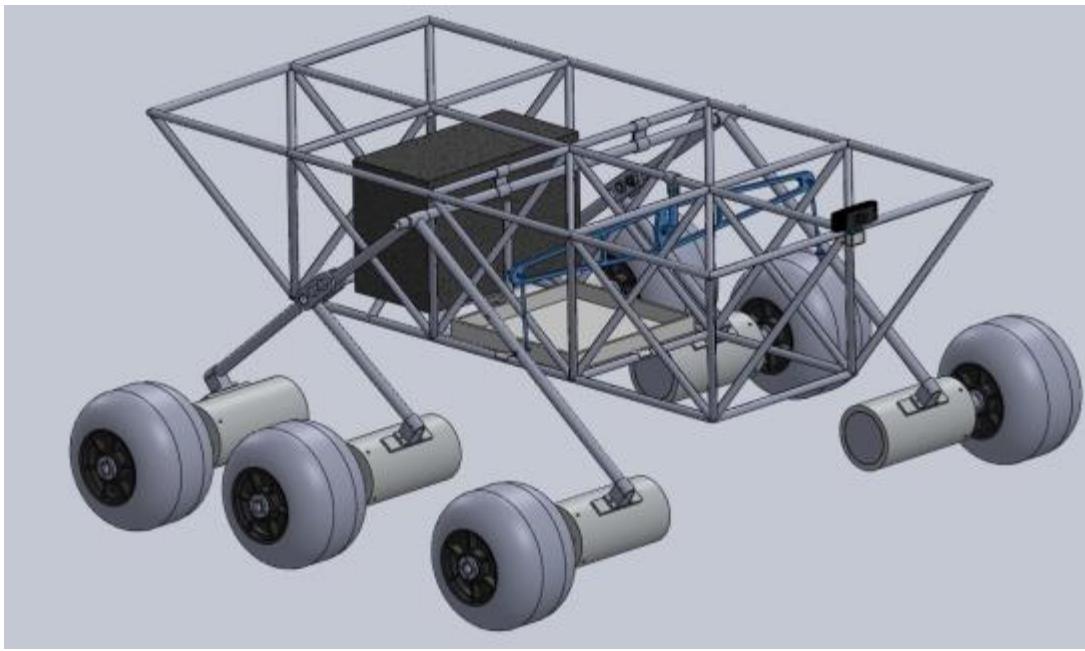
Evolution of overall rover concept and design.



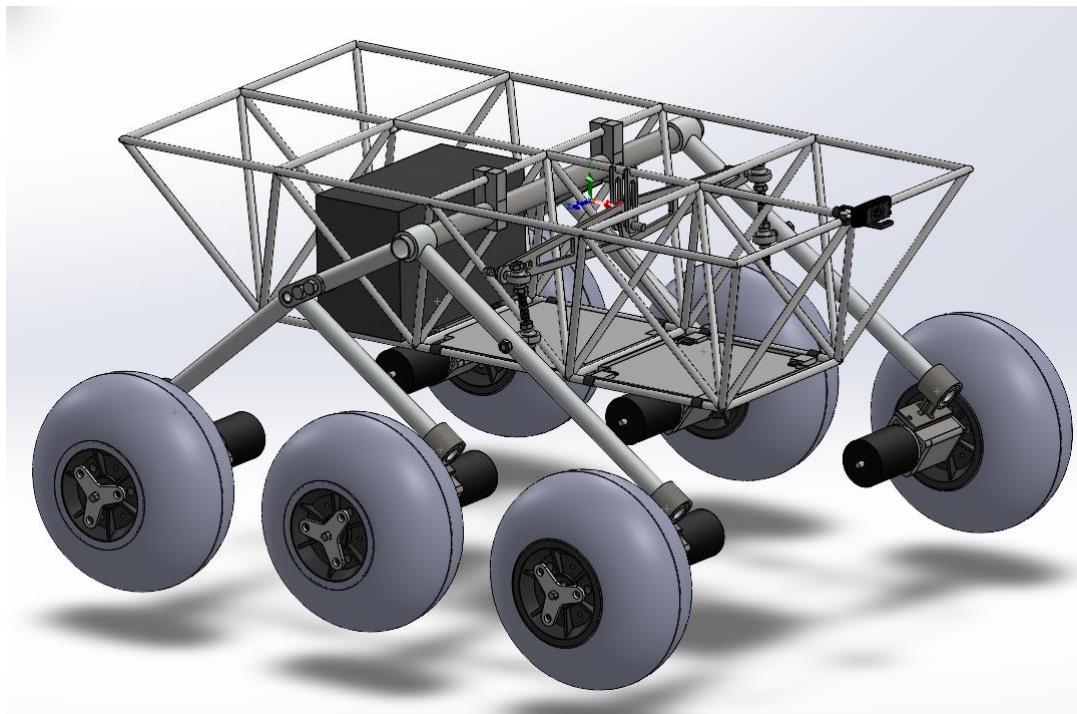
XIII.B-1: Full rover SolidWorks model used for fall midterm presentation.



XIII.B-2: SolidWorks model of frame and legs circa late October.



XIII.B-3: Full rover SolidWorks model used for fall final presentation.



XIII.B-4: Full rover SolidWorks model at the time of writing the spring final report.

XIII.B.1. Concept/Solution for Function F1 – Drive on Various Terrain

XIII.B.1.a. Original Concept Selected – Details

Criterion	Weights	Concepts			
		AWD	4WD	FWD	RWD
Traction	20.98%	0	1	-1	-1
Carrying Capacity	8.26%	0	1	-1	-1
Climb Capability	20.98%	0	1	-1	-1
Braking	2.22%	0	-1	1	1
Turning	2.22%	0	-1	1	1
Required Power	5.89%	0	-1	1	1
Maintenance	10.81%	0	-1	1	1
Cost	19.70%	0	0	1	1
Weight	8.94%	0	0	1	1
Weighted Total	100.00%	0	0.2908	-0.0044	-0.0044

XIII.B.1.b. Re-design Concept Selected – Details

[N/A]

XIII.B.1.c. Original and Re-designed Concepts Considered but not Selected.

	Weights										
	Traction	Carry Capacity	Climb Capability	Braking	Turning	Required Power	Maintenance	Cost	Weight	Geometric Average	Normalized Weights
Traction	1	3	1	7	7	5	3	1	5	2.81	22.65%
Carrying Capacity	1/3	1	1/3	3	3	3	1	1/3	1	1.00	8.05%
Climb Capability	1	3	1	7	7	5	3	1	5	2.81	22.65%
Braking	1/7	1/3	1/7	1	1	1/3	1/5	1/7	1/5	0.29	2.31%
Turning	1/7	1/3	1/7	1	1	1/3	1/5	1/7	1/5	0.29	2.31%
Required Power	1/5	1/3	1/5	3	3	1	1/3	1/5	1	0.58	4.71%
Maintenance	1/3	1	1/3	5	3	3	1	1/3	3	1.20	9.63%
Cost	1	3	1	7	7	5	3	1	3	2.66	21.40%
Weight	1/5	1	1/5	5	5	1	1/3	1/3	1	0.78	6.31%

XIII.B.2. Concept/Solution for Function F2 – Facilitate Expansion

XIII.B.2.a. Original Concept Selected – Details

The selected concept for the frame was the skeleton design. The relevant features selected were weight, cost, expandability (how easy it would be to add more stuff in the future), strength, and ergonomics (how comfortable/easy would it be to transport). An AHP table was used to score the weights of each of the features. The two most important were found to be weight and cost. It should be noted that at the time of concept selection the projects budget was 33% lower than final budget. The two other concepts considered were the platform and box designs. Due to the truss construction, the skeleton frame was estimated to be the better choice for both strength and weight. The box was expected to be slightly cheaper due to using fewer materials. The platform was estimated as the easiest to transport due to its thin shape. After running these concepts through a decision matrix, the skeleton frame came out on top, followed by the platform and then the box.

Frame			Concepts		
	Criterion	Weights	Skeleton	Platform	Box
	Weight	0.294	0	-1	-1
	Cost	0.236	0	-1	1
	Expandability	0.149	0	1	-1
	Strength	0.294	0	0	-1
	Ergonomics	0.027	0	1	-1
	Sum total +'s	-	0	2	1
	Sum total -'s	-	0	-2	-4
	Overall Total	-	0	0	-3
	Weighted Total	-	0	-0.3546156	-0.52734064

Frame	Weight	Cost	Expandability	Strength	Ergonomics	Average	Nomal Weight
Weight	1	3	5	1	7	17	0.294
Cost	0.333	1	5	0.333	7	13.666	0.236
Expandability	0.2	0.2	1	0.2	7	8.6	0.149
Strength	1	3	5	1	7	17	0.294
Ergonomics	0.14	0.14	0.14	0.14	1	1.56	0.027
SUM						57.826	1

XIII.B.2.b. Re-design Concept Selected – Details

N/A

XIII.B.2.c. Original and Re-designed Concepts Considered but not Selected.

See discussion in XII.B.2.a.

XIII.B.3. Concept/Solution for Function F3 – Distribute Power to Electronics

XIII.B.3.a. Original Concept Selected – Details

The datum was selected to be lead-acid batteries. As the research done by the time the decision matrix was constructed was general and without analysis, values ranging from -1 to 1 were used for the comparisons made in the matrix. Lithium batteries, by all accounts, were better than lead-acid batteries in all aspects besides the fact that they are more expensive. As a result, they were assigned a 1 in all categories except cost, where they were given a -1 rating. Solar power was found to be comparable in quality to the lead-acid datum in terms of durability, mass, and maintenance, which earned it a 0 rating for those criteria. However, the cost and dimensions of the panels were determined to be worse than the lead-acid battery, earning it a mark of -1 in those categories. Nickel metal hydride (NiMH) batteries were the final option to be considered. They tend to be both lighter and smaller than lead-acid batteries. This earned them a 1 rating in

both the mass and dimension criterion. Even though they cost about the same as lead-acid batteries, they tend to be less durable and require more maintenance. As a result, they received a 0, -1, and -1 rating in these categories, respectively. Overall, lithium batteries were illustrated to be the best power supply choice, followed by lead-acid, NiMH, and solar power.

Criterion	Weights	Concepts			
		Lead-Acid	Lithium	Solar Power	NiMH
Cost	44.09%	0	-1	-1	0
Durability	31.47%	0	1	0	-1
Mass	12.60%	0	1	0	1
Maintenance	8.12%	0	1	0	-1
Dimension	3.73%	0	1	-1	1
Weighted Total	100.00%	0	0.1182	-0.4782	-0.2326

XIII.B.3.b. Re-design Concept Selected – Details

N/A

XIII.B.3.c. Original and Re-designed Concepts Considered but not Selected.

Function F3 is the “Distribute Power to Electronics” main function. The most relevant and important criteria to be considered were to be cost, mass, dimension, durability, and maintenance. Due to our modest budget (\$6,000) in comparison to similar projects, we determined that cost was the most important criterion we would be considering and thus it should be weighted more heavily than any of the other criteria. Regardless of how good a power supply option is, if it does not fit our budget, it cannot be selected. As our project is one that future Capstone teams will be building upon, the durability of power supply is the next most important criterion from our list. Without a durable power supply, future capstone teams will need to revisit and potentially replace our original power source, which will in turn hinder their ability to simply expand on our base design. As a result, the durability criterion is weighted higher than all other criteria besides cost, which it is rated as moderately less important than. For the three remaining criteria, there was no one criterion that was significantly more important than the rest. Because the mass of the power supply has a direct impact on the speed and demand of the motors amongst other functionality related impacts, mass was assigned a weight that indicated it is moderately more important than dimension and maintenance. As all the power supply options will require almost equal levels of maintenance, so long as they are durable, the maintenance criterion was determined to be less important than all other criteria besides dimension. Maintenance was considered instead to be ‘more important’ than dimension is, indicated by the weight in that spot. The dimensions of the power supply are the least important criterion for the same reason as for maintenance as described above, but it is less important than maintenance because the dimensions of the power supply is something that can be worked around. With the criteria being weighted as described above, the best power supply option was identified to be a lithium-ion battery. The AHP for these concepts can be seen in Figure XII-7 in Appendix XII.B.3.b. Additionally, custom power distribution boards were ruled out as an option to supply power throughout the system due to the system’s current and voltage demands. Consequently, bus bars were selected to distribute power to the rover’s components as needed.

	Cost	Mass	Dimension	Durability	Maintenance	Geometric Average	Normalized Weight
Cost	1	3	7	3	5	3.16	44.09%
Mass	0.33	1	3	0.2	3	0.903	12.60%
Dimension	0.14	0.33	1	0.14	0.2	0.267	3.73%
Durability	0.33	5	7	1	5	2.255	31.47%
Maintenance	0.2	0.33	5	0.2	1	0.582	8.12%

XIII.B.4. Concept/Solution for Function F4 – Operate Safely

XIII.B.4.a. Original Concept Selected – Details

We utilize time-delay fuses for the motors. The time-delay fuses ensure that the motors can draw enough current during a transient without blowing the fuse immediately. We determined that a decision matrix was not necessary for the concepts for this function as using fuses was the overwhelmingly obvious choice for our application.

XIII.B.4.b. Re-design Concept Selected – Details

The circuit breakers that primarily act as manual switches were added out of necessity in order to ensure a safer startup for the rover. There was no decision process here as the only option we had available to us was to use circuit breakers as switches.

XIII.B.4.c. Original and Re-designed Concepts Considered but not Selected.

The other concept originally generated for this function was using circuit breakers in their normal function, rather than just as switches. This option does not offer much when compared with fuses, specifically. The advantage that circuit breakers have is that they can be closed and opened without having to be replaced like fuses do. Circuit breakers, however, are larger than fuses and more expensive. Since we are trying to minimize weight and cost in the rover, we decided that circuit breakers would not be as good of a choice as fuses are. Circuit breakers also would not fit into the physical circuits as easily as fuses would. Fuses also are not very expensive so having to replace them is not as big of a downside as it could be if they were more expensive. This decision was mostly about using them in motor protection. Because of the battery's size, the circuit breakers that we use as switches fit nicely on top of it.

XIII.B.5. Concept/Solution for Function F5 – Transmit Data

XIII.B.5.a. Original Concept Selected – Details

Function F5 “Transmit Data” consists of transmitting control instructions to the rover and transmitting sensor data back to the user interface of the controller. This function was realized using a Wi-Fi module attached to our single board computer that will transmit signals to the controller. The team ultimately decided on Wi-Fi based on a few factors. The required range of the rover lies within the specifications of a Wi-Fi module, a Wi-Fi link is reliable if the rover and controller are the only devices on the signal, Wi-Fi has a low cost of implementation, Wi-Fi has a low average latency, Wi-Fi has built in security and encryption features, and Wi-Fi has minimal power consumption compared to other options. Transmitting data is an integral function for the project, and Wi-Fi is the best option given our perimeters and constraints. We used a Wi-Fi module that connects to our single board computer via a network interface card (NIC).

Criterion	Weights	Concepts			
		RF	Bluetooth	WiFi	Satellite
Range	17.47%	0	-1	0	1
Reliability	38.76%	0	-1	0	1
Cost	8.66%	0	0	0	-1
Average Latency	22.79%	0	0	1	-1
Complexity	2.95%	0	1	0	-1
Security	5.27%	0	-1	1	1
Power Consumption	4.10%	0	1	0	-1
Weighted Total	100.00%	0	-0.5445983	0.280646112	0.23010617

XIII.B.5.b. Re-design Concept Selected – Details

N/A

XIII.B.5.c. Original and Re-designed Concepts Considered but not Selected.

For the data transmission there were four main concepts considered: RF, Wi-Fi, Bluetooth, and Satellite communication, of which the aforementioned Wi-Fi was deemed most viable for the project. Through the team's research we determined that Bluetooth communication did not have the necessary range, reliability, and security features that were required for the project. Satellite communication did not have the necessary cost, average latency, and power consumption that was required for the project. RF communication was closely behind Wi-Fi; however, because of Wi-Fi's superior latency and security features it was ultimately determined to be the best choice for our use case. Wi-Fi is going to allow us to use the rover as an access point and connect the digital controller to the rover.

	Weights								
	Range	Reliability	Cost	Average Latency	Complexity	Security	Power Consumption	Geometric Average	Normalized Weight
Range	1.00	0.20	5.00	0.33	5.00	5.00	7.00	1.79	17.47%
Reliability	5.00	1.00	7.00	3.00	7.00	3.00	7.00	3.97	38.76%
Cost	0.20	0.14	1.00	0.20	5.00	3.00	5.00	0.89	8.66%
Average Latency	3.00	0.33	5.00	1.00	5.00	5.00	3.00	2.33	22.79%
Complexity	0.20	0.14	0.20	0.20	1.00	1.00	0.20	0.30	2.95%
Security	0.20	0.33	0.33	0.20	1.00	1.00	3.00	0.54	5.27%
Power Consumption	0.14	0.14	0.20	0.33	5.00	0.33	1.00	0.42	4.10%

XIII.B.6. Concept/Solution for Function F6 – Operate Remotely

XIII.B.6.a. Original Concept Selected – Details

Only two concepts were considered for the type of control selection for the "Operate Remotely" function: a digital controller, and a physical controller. For the decision matrix, the digital controller was set as our datum, with the physical controller being evaluated in relation. Values used to evaluate the physical controller in comparison to the digital controller datum ranged from -1 to 1 due to our general understanding of the two concepts. Since the reliability of the controller depends on our design, regardless of the type, the physical controller received a rating of 0 for the reliability criterion. A physical controller can only use its built-in buttons, whereas a digital controller can have buttons and features added as needed. Thus, the physical controller scored a -1 in the customizability criterion in relation to the datum. When it came to evaluating the

ease-of-use criteria of the physical controller, we considered again the set layout of a physical controller and inability to change how easy it is for someone unfamiliar with the project to use it, thus restricting the ease of use of it. So, again it scored a -1 in relation to the datum. Overall, the physical controller was found to be not as good an option as the digital controller datum, as it scored a negative total when it was all said and done. As a result, a digital controller is the type of control we chose for our rover.

	Concepts		
Criterion	Weights	Digital Controller	Physical Controller
Ease of use	28.10%	0	-1
Reliability	58.44%	0	0
Customizability	13.46%	0	-1
Weighted Total	100.00%	0	-0.415579698

XIII.B.6.b. Re-design Concept Selected – Details

N/A

XIII.B.6.c. Original and Re-designed Concepts Considered but not Selected.

For this function, there were three main characteristics that were determined to be most important: ease of use, reliability, and customizability. Out of these three criteria, the most important to our project is reliability. Without being able to reliably operate the rover remotely, all our work will be for nothing. So, reliability is weighed as more important than both the ease-of-use criterion and the customizability criterion. The ease-of-use criterion was considered to be the second most important criterion for this function, as a complex method of operation would greatly hinder the rest of what is accomplished with our rover when it is controlled by somebody unfamiliar with the project. As a result, ease of use is weighed as more important than customizability. Although customizability is certainly a relevant aspect to be considered when picking our type of control, it is simply not as important as the other criteria and thus was weighed as such. With the criteria being weighted in this fashion, it was determined that a digital controller was the best control type option.

Weights						
Criterion	Ease of use	Reliability	customizability	Geometric Average	Normalized Weight	
Ease of use	1.00	0.33	3.00	1	28.10%	
Reliability	3.00	1.00	3.00	2.080083823	58.44%	
customizability	0.33	0.33	1.00	0.479141986	13.46%	

XIII.B.7. Concept/Solution for Function F7 – Monitor System Statuses

XIII.B.7.a. Original Concept Selected – Details

For Function F7, "Monitor System Statuses," we will employ various sensors to track critical aspects of the rover. These include the Endure Power app to monitor battery life and current draw in real-time, a temperature sensor for the electronics housing, and a gyroscope to measure the rover's incline or angle. The real-time battery life sensor is crucial to gauge operational range, especially in remote operations. Temperature sensors are vital for both safety and preserving the electrical components in hot environments. The gyroscope monitors the rover's angle so that the

operator always know the slope that the rover is on. This can be useful when trying to expand the capabilities of the rover by driving over steeper and steeper slopes.

XIII.B.7.b. Re-design Concept Selected – Details

N/A

XIII.B.7.c. Original and Re-designed Concepts Considered but not Selected.

We determined that a humidity sensor is not critical for the operation of our rover as we will not be operating in any wet environment that could damage electronics. Some temperature sensors have built in humidity sensors, however, so we may still pursue this option depending on future need and budget. No decision matrix is necessary. These three sensors were the only ones we seriously considered as they were the most important to our project. There are no competing concepts for each sensor but there are other concepts for sensors that we came up with that were not deemed necessary for our rover. These concepts were humidity sensors and an accelerometer. These concepts realistically are still available as options if a future team decides they want them further into this project but for now they are not seen as critical.

XIII.B.8. Concept/Solution for Function F8 – Contain Autonomous Features

XIII.B.8.a. Original Concept Selected – Details

Function F8 “Contains Autonomous Features” is a complex function with many different concepts used to realize the solution. To achieve this function, using extensive research and decision matrices, the team decided to implement a subsystem for autonomy which includes a single board computer (SBC) that utilizes data from a camera in addition to an ultrasonic sensor to employ computer vision and obstacle avoidance algorithms, while tracking the rover’s location in a map of the environment. To achieve this, we needed to find a SBC with a powerful CPU and GPU. Through our research and a thorough decision matrix we decided to use the Jetson Nano – a SBC made by NVIDIA that lets you run multiple neural networks in parallel for applications like image classification, object detection, segmentation, and speech processing. It has 128 CUDA cores that help with machine learning and image processing. Out of the SBCs considered it was the most advanced in terms of GPU performance for computer vision algorithms. We chose a camera with an ultrasonic sensor to give us both image and depth information to more accurately avoid objects. We will implement simple Vector Field Histogram (VFH) algorithms for real time obstacle avoidance; this is because VFH uses real time sensor data to create a histogram grid of surroundings and calculates the best path to avoid obstacles. We will not be completely implementing the utilization of the algorithm as our sponsor wants future capstone teams to dive into the object avoidance algorithms. This was chosen because it is also easier for further teams to add additional algorithms for obstacle avoidance without needing to alter the existing VFH algorithms. For object avoidance the team has chosen to use The BendyRuler algorithm. The BendyRuler algorithm probes around the vehicle in many directions looking for open spaces and then tries to pick the direction that is sufficiently open while also moving the vehicle towards the final destination. We chose the BendyRuler algorithm because it is a real-time object detection and avoidance system that facilitates future expansion for future capstone teams. If we require further designation of objects we plan to implement Convolutional Neural

Networks (CNNs) for object recognition. If we require mapping we will utilize the addition of simultaneous localization and mapping (SLAM) algorithms for creating a map of the environment while tracking the rover's location.

	Weights											
	Processing Power	Memory	Power Consumption	I/O	Connectivity	Software Support	Cost	Security	ML/AI Capability	Geometric Average	Normalized Weight	
Processing Power	1.00	3.00	7.00	3.00	1.00	3.00	7.00	3.00	7.00	3.12	0.271375223	
Memory	0.33	1.00	7.00	5.00	1.00	1.00	5.00	3.00	1.00	1.78	0.154540889	
Power Consumption	0.14	0.14	1.00	0.20	0.14	0.33	1.00	1.00	1.00	0.39	0.03368533	
I/O	0.33	0.20	5.00	1.00	1.00	3.00	3.00	0.33	3.00	1.13	0.098362697	
Connectivity	1.00	1.00	7.00	1.00	1.00	1.00	5.00	3.00	3.00	1.89	0.164970776	
Software Support	0.33	1.00	3.00	0.33	1.00	1.00	5.00	3.00	3.00	1.35	0.117623409	
Cost	0.14	0.20	1.00	3.00	0.20	0.20	1.00	0.33	0.33	0.42	0.036300591	
Security	0.33	0.33	1.00	3.00	0.33	0.33	3.00	1.00	1.00	0.78	0.068200885	
AI Capability	0.14	1.00	1.00	0.33	0.33	0.33	3.00	1.00	1.00	0.63	0.054940199	

XIII.B.8.b. Re-design Concept Selected – Details

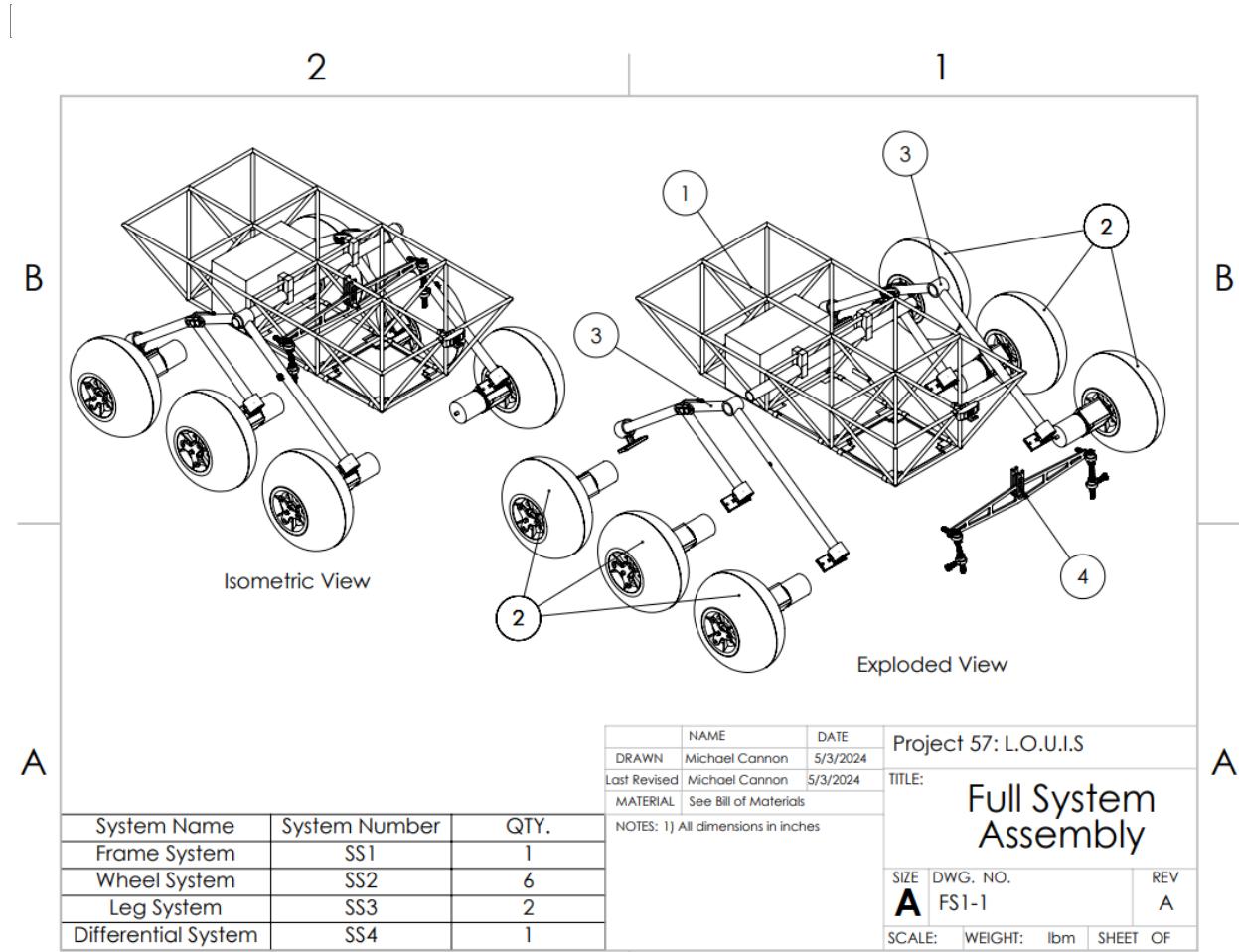
N/A

XIII.B.8.c. Original and Re-designed Concepts Considered but not Selected.

Many concepts were considered for the “autonomous features” for the project. The main concept the team chose for the autonomous features was the single board computer. The team considered different types of microcontrollers and SBCs; the four options considered in the final decision matrix were the Raspberry Pi 4, the Arduino line of SBCs, the Jetson Nano, and the Google Coral Dev Board. The Jetson Nano was much more powerful than the Raspberry Pi and Arduino, which was important for our implementation of computer vision. The Jetson Nano was very similar to the Google Coral; however, we ultimately went with the Jetson Nano due to its onboard memory to run multiple applications and the 128 CUDA cores that would help with efficiency future machine learning and computer vision implementations. The Jetson Nano also had the most robust I/O and software support when compared to the other options. The team considered using Infared sensors instead of Ultrasonic sensors in conjunction with the camera to give us depth information. However, infrared sensors have a shorter range and can be affected by changing light conditions. The team considered Single Shot MultiBox Detector for the object detection algorithms and “you only look once” (YOLO) algorithms for object detection and avoidance. However, the BendyRuler algorithm gives us the most room for expansion by future capstone teams, which is our sponsors highest priority.

Criterion	Weights	Concepts				
		Raspberry Pi	Arduino	Jetson Nano	Google Coral	
Processing Power	27.14%	0	-1	1	1	
Memory	15.45%	0	-1	1	0	
Power Consumption	3.37%	0	1	-1	-1	
I/O	9.84%	0	-1	0	0	
Connectivity	16.50%	0	-1	0	0	
Software Support	11.76%	0	-1	0	0	
Cost	3.63%	0	1	-1	-1	
Security	6.82%	0	-1	0	0	
ML/AI Capability	5.49%	0	-1	1	1	
Weighted Total	100.00%	0	-0.828127864	0.447170982	0.292630093	

XIII.C. System Description/Product Architecture Supplement



XIII.C-1: Exploded view of mechanical subsystems

XIII.C.1. Sub-System SS1 - Frame

The frame consists of a series of skeletal struts which form a strong and lightweight frame. The open nature of the frame allows for easy access to the internal space. The struts also allow for components to be bolted at many points along the frame. These two features allow for additional equipment to be easily added to the frame. Additional equipment could include robotic arms, scientific sampling equipment, solar panels, etc. The axle sits underneath the top level of the frame. This allows the frame to rest on the axle without extra joints. To keep the frame from sliding relative to the axle, the axle will be welded to the struts that contact it. Since this does not offer a lot of contact space, two additional brackets will be welded to the frame and axles. This also holds the frame to the axle when frame is lifted and the forces are reversed so that the frame is no longer resting on the axle. The legs slide onto the axle and are held in place by retaining rings. This means that the legs can be relatively easily removed from the frame system. The electronics are held in 3D printed cases. Since these parts do not need to hold any significant weight, detailed analysis is not required. Engineering drawings are not needed at this time also

since the exact geometries are not yet known. Rather, these parts will be custom designed and printed to fit the exact geometry and configuration encountered during the manufacturing process.]

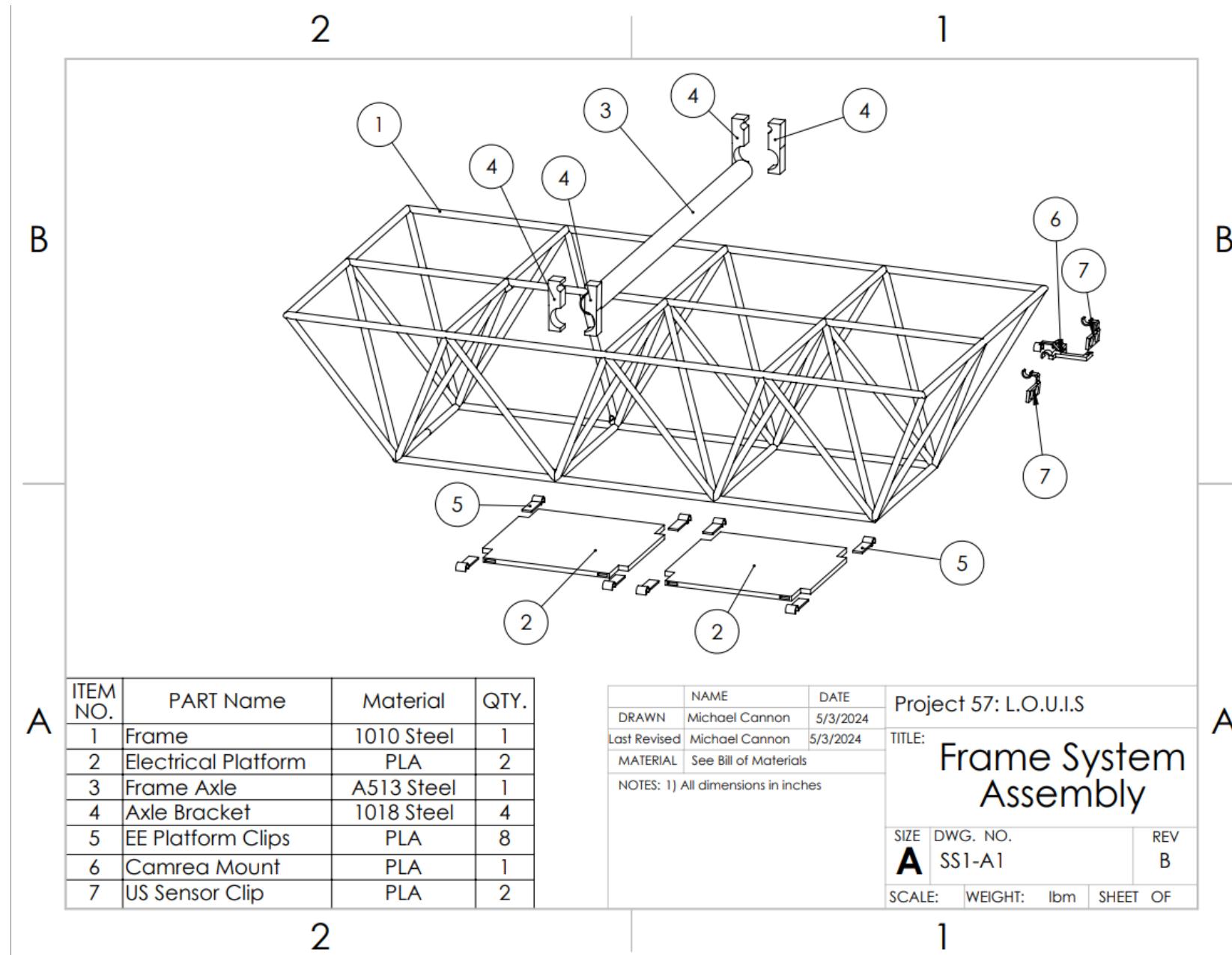


Figure XIII-2: Exploded View Assembly Drawing of Sub-System SS[1] - Frame

XIII.C.1.a. *Comprehensive Parts List for SS1 - Frame*

[Table XIII-1: List of Parts for Sub-System SS1]

Part #	Quantity	Material	Name
SS1-P1	1	1010 Steel	Frame
SS1-P2	2	PLA Plastic	Electronics Platform
SS1-P3	1	A513 Steel	Frame Axle
SS1-P4	4	1018 Steel	Axle Bracket
SS1-P5	8	PLA Plastic	EE Platform Clips
SS1-P6	1	PLA Plastic	Camera Mount
SS1-P7	2	PLA Plastic	US Sensor Clip

Engineering (Manufacturing) Drawings of All Parts of SS1 - [Frame]

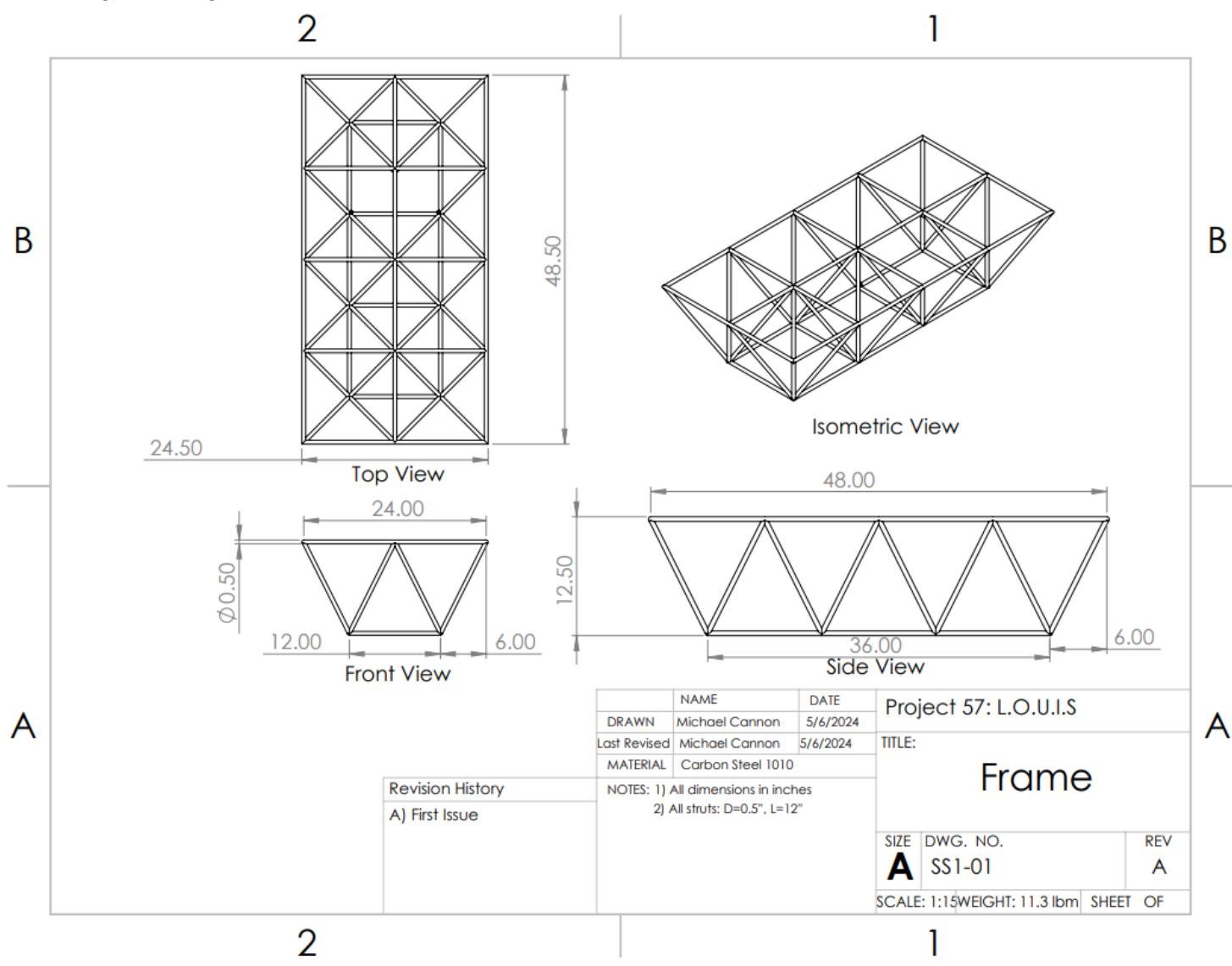


Figure XIII-3: Manufacturing Drawing of part SS1-P1 - [Frame]

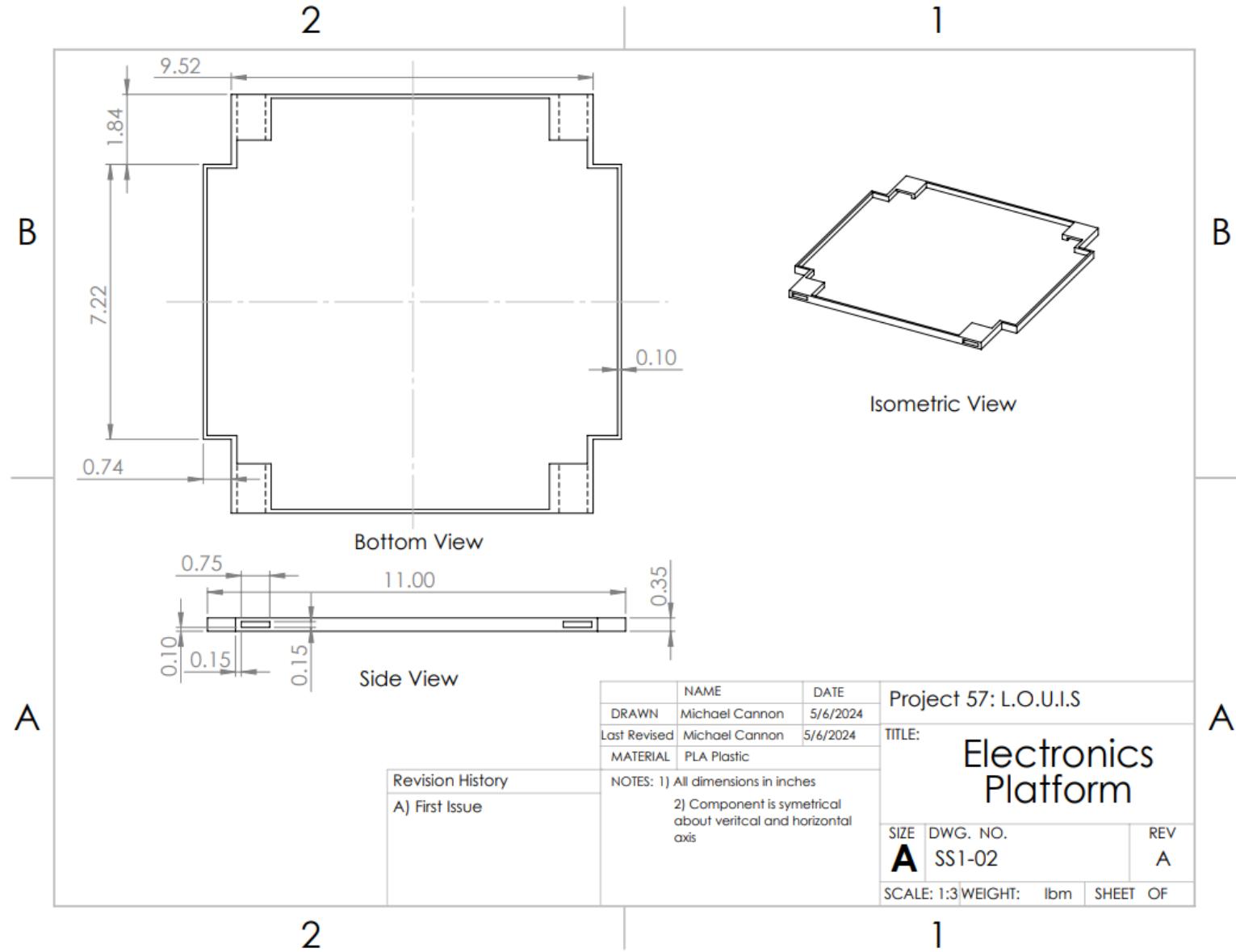


Figure XIII-4: Manufacturing Drawing of part SS1-P2 – Electronics Platform

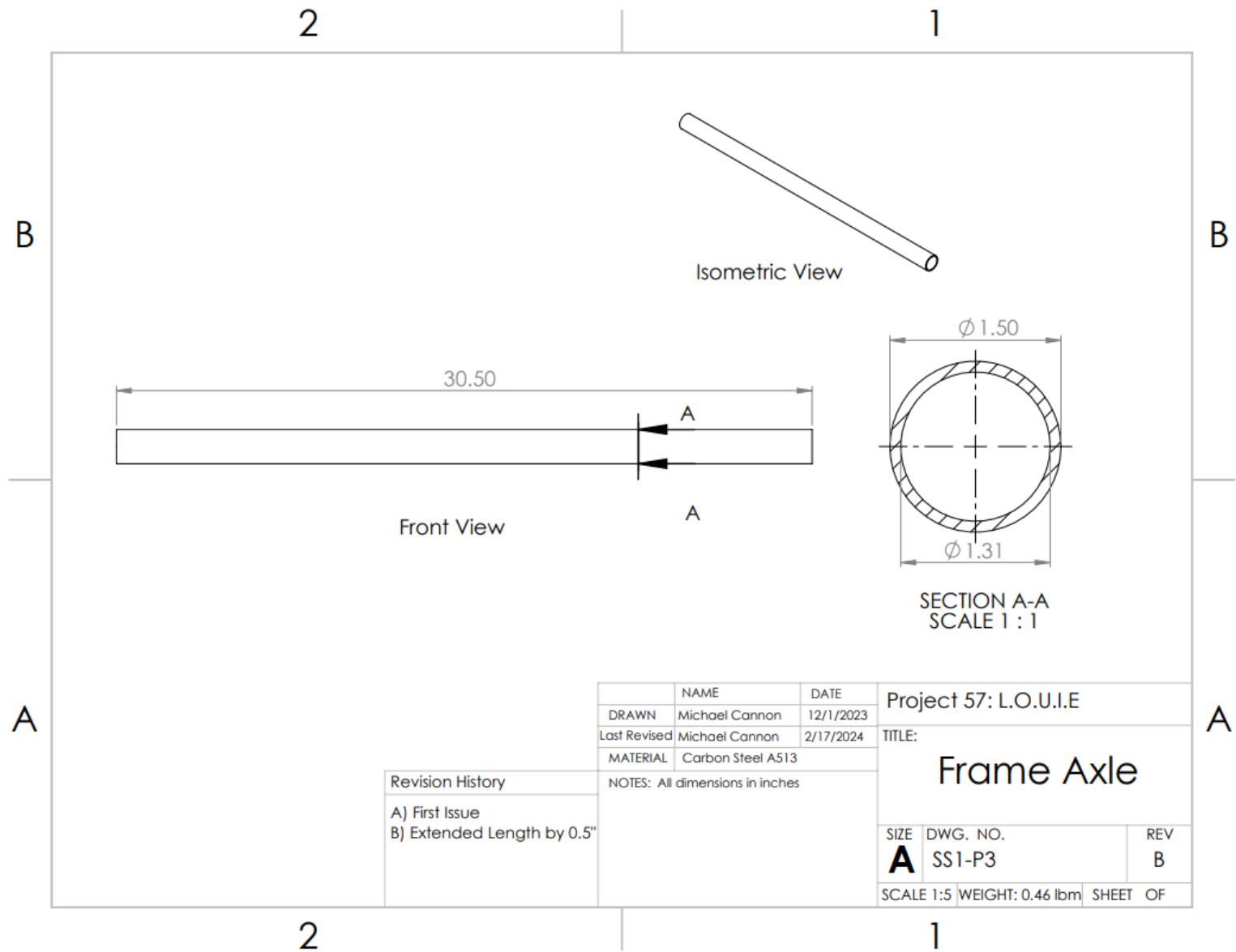


Figure XIII-5: Manufacturing Drawing of part SS1-P3 – Frame Axle

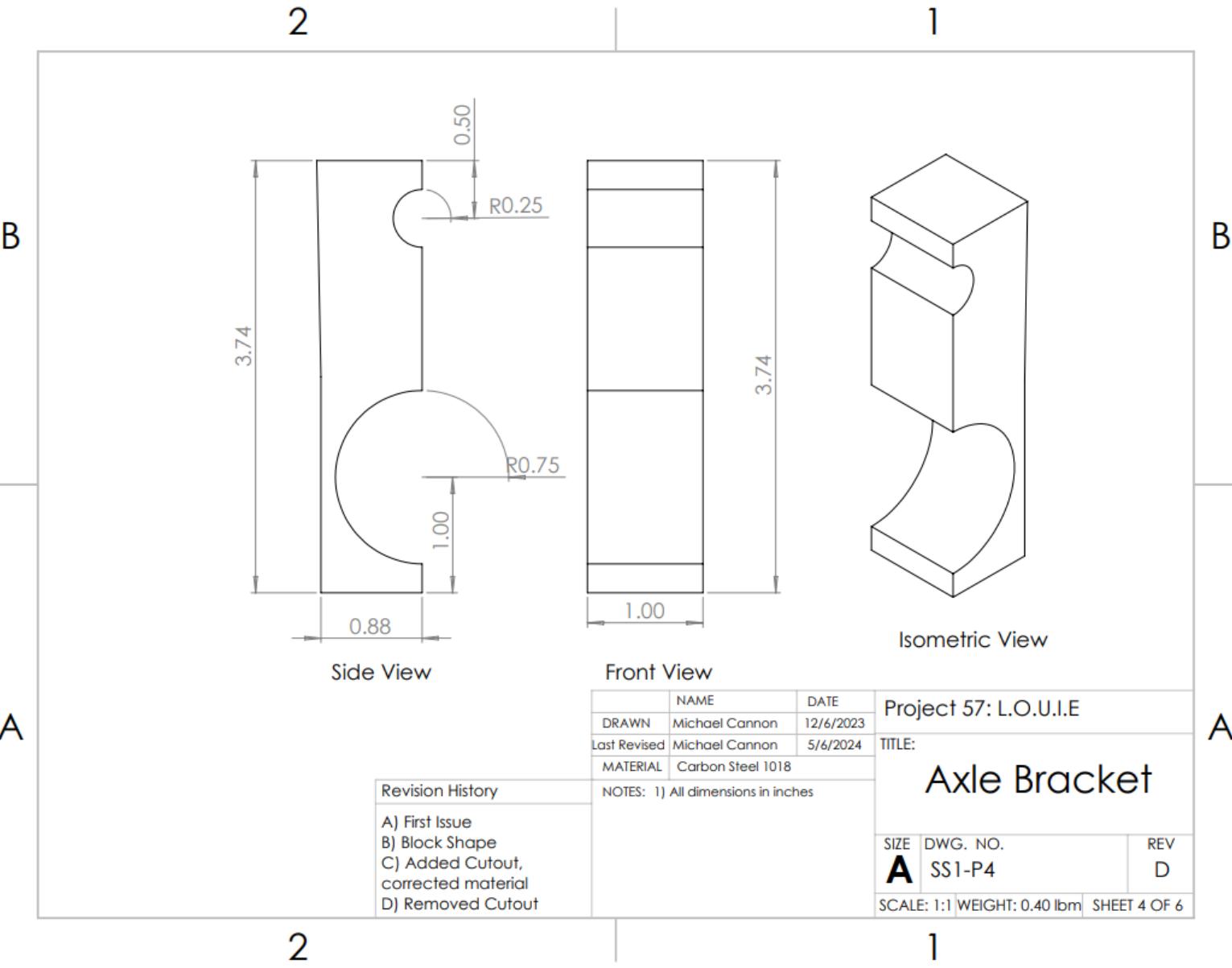


Figure XIII-6: Manufacturing Drawing of part SS1-P4 – Axle Bracket

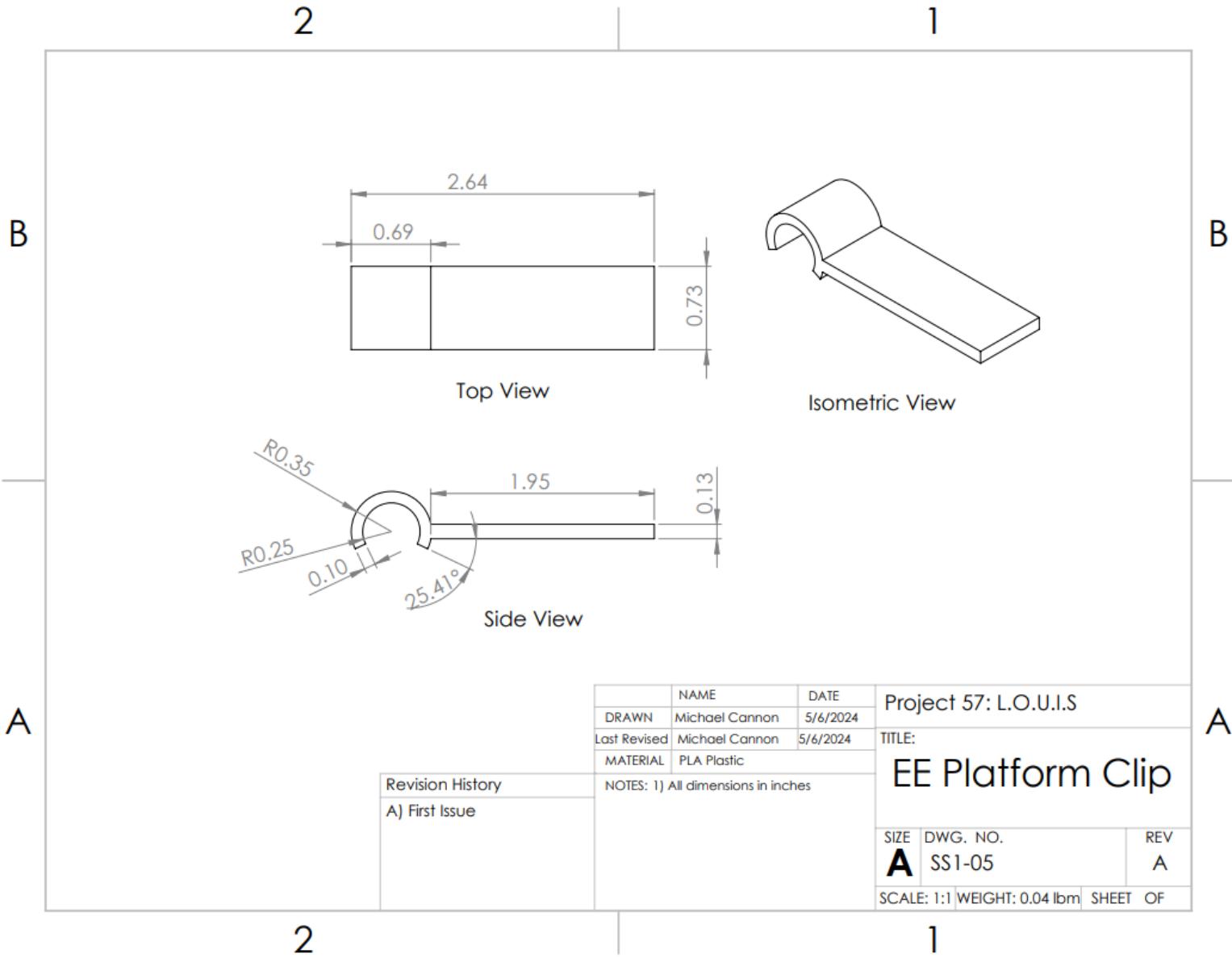


Figure XIII-7: Manufacturing Drawing of part SS1-P5- EE Platform Clip

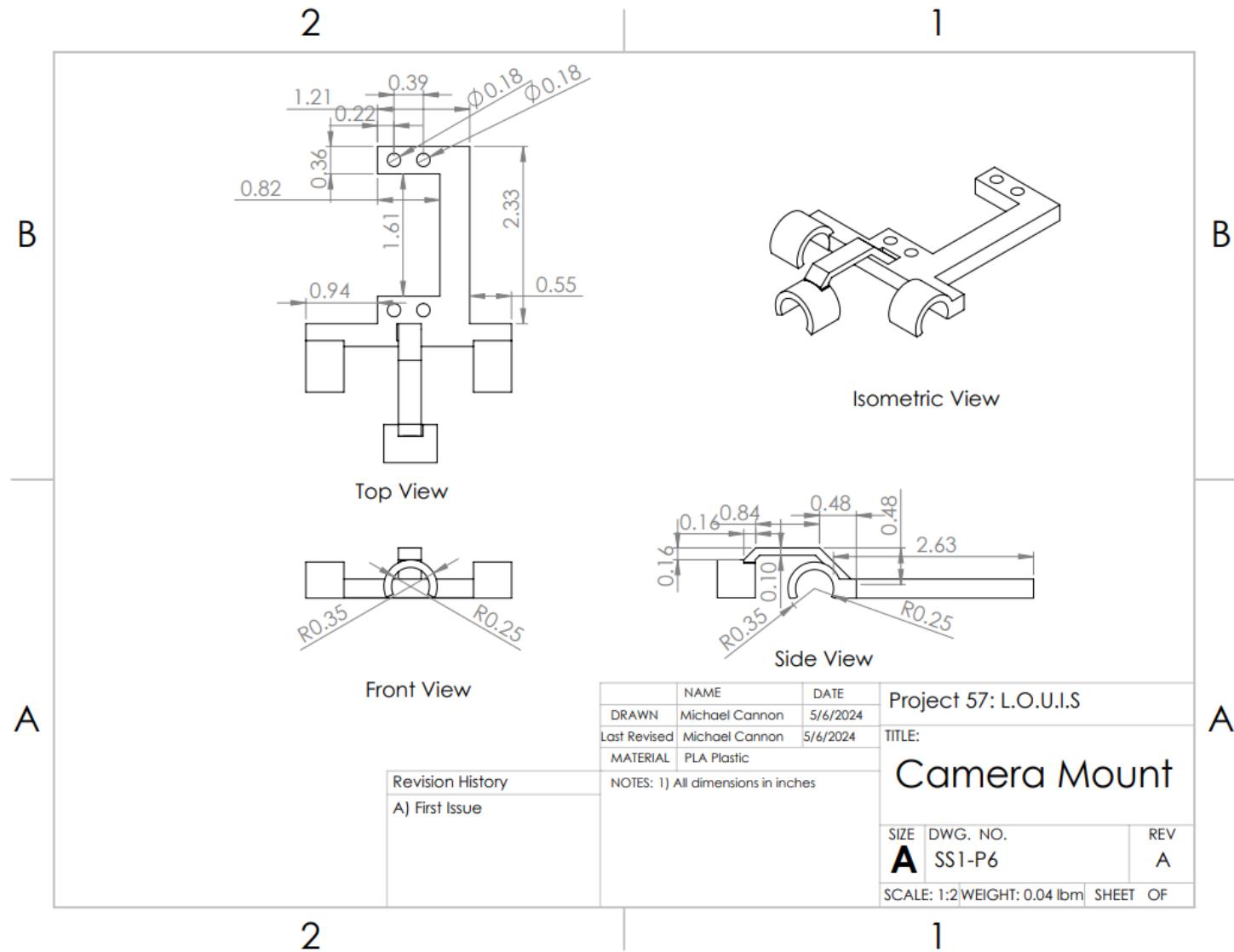


Figure XIII-8: Manufacturing Drawing of part SS1-P6- Camera Mount

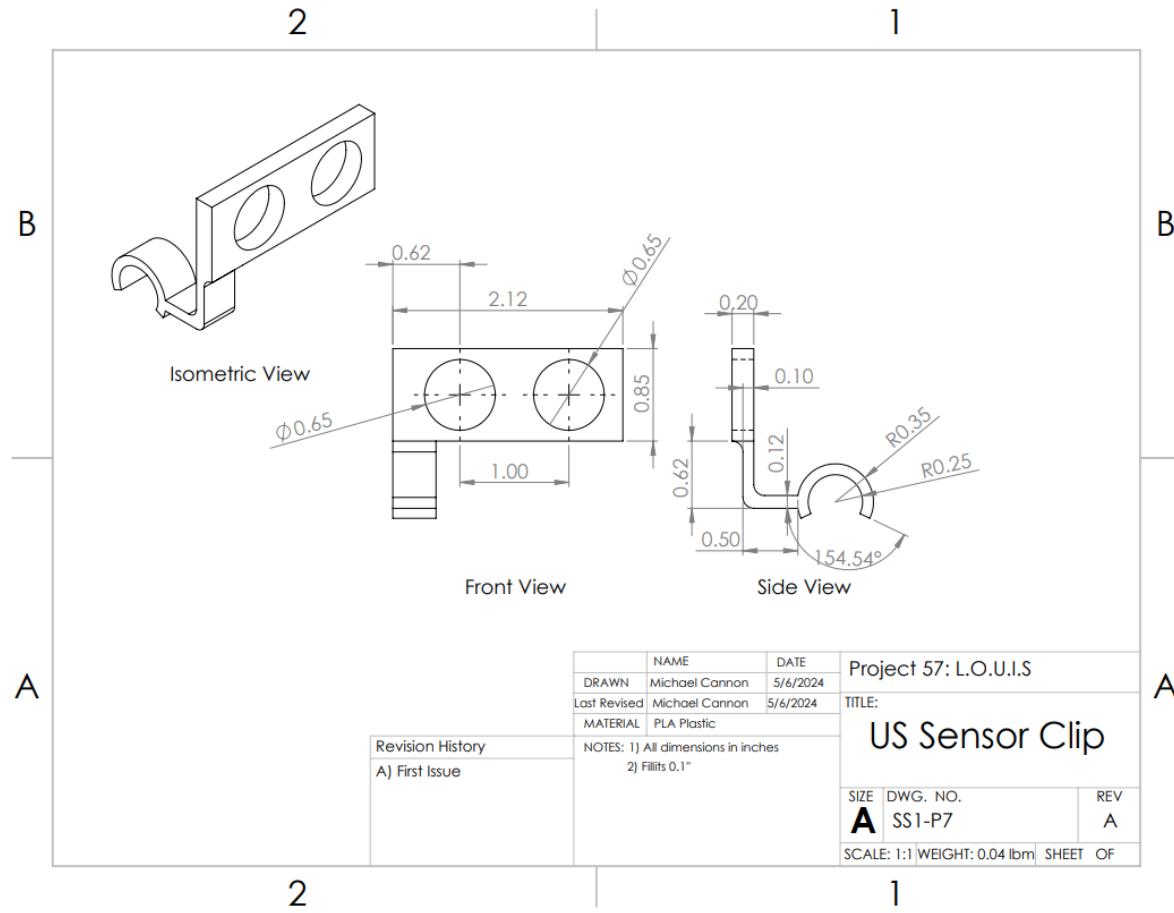


Figure XIII-9: Manufacturing Drawing of part SS1-P7– US Sensor Clip

XIII.C.2. Sub-System SS2 - Drivetrain

The drivetrain starts at the motor. The motor is held on a fixed motor housing which is bolted to the legs. A fixed axle is welded to the motor housing. The wheel sits on this axle and rotates about it. It is important to note that this axle does not provide any torque to the wheel and only supports the rover structurally. The end of the motor has a short driveshaft with a key. This keys into our longer driveshaft which passes through the fixed axle and keys into the hubcap using a hex key. This driveshaft does not provide any structural support and can fully focus on transmitting torque to the wheel. Three half-inch bolts pass through the hubcap and the wheel and are bolted into place. These bolts transmit the torque to the wheel. To keep the wheel from sliding along the axle, there is a retaining ring on the motor side and a hitch pin on the hubcap side. Both of these components are removable. The motor is secured to the housing by 4 bolts. The Bracket that attaches the motor housing to the leg is bolted to the motor housing and contains a set screw to tighten it to the legs.

This design has several advantages. Since most parts are attached through impermeant means, the system can be disassembled into almost all of its separate parts. Since the keyway on the driveshaft is a hole instead of a slot, the driveshaft is very secure on the motor. Since the driveshaft and axle are two separate components, the driveshaft theoretically does not have to withstand any bending forces. The wheel is an off the shelf component designed specifically for sand. As such, it performed very well on sandy terrain during testing. The system can quickly be unbolted from the leg, and the angle the wheel sits at can also be adjusted by adjusting the set screw.

There are also a few disadvantages. The axle itself is a simple bearing with no ball bearings or lubrication. Over long periods of use it could be susceptible to wear. The size of the axle hole in the wheel limited the size of axle and therefore the size of the driveshaft. Due to these tight tolerances, some friction wear has also been noted on the driveshaft. While the keyway design has proven to be very secure, the driveshaft cannot be easily removed unless it breaks at the keyway joint. The motor screws are oriented with the heads facing in and the nuts on the outside of the housing. Unfortunately, there is no access to the screw heads once the motor is in place so the screws have to be tightened by gripping the threads with vice grips. This means to remove the motor each screw has to be sawed off with a hack saw. The solution to this is to mill hex wrench slots into the motor housing to provide access. This would take a full day of manufacturing since there are 24 slots that would need to be milled which could not be fit into the schedule during our limited testing windows.

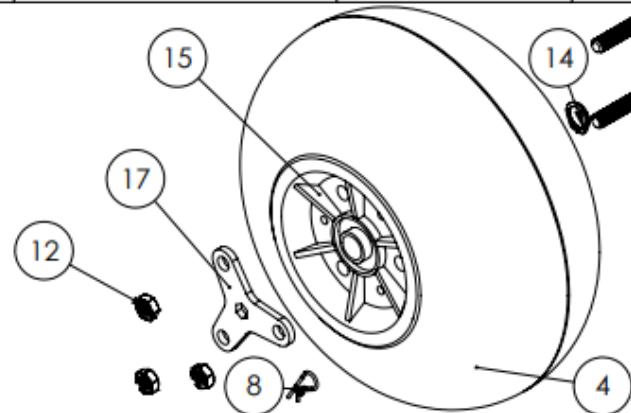
2

1

ITEM NO.	PART NAME	Material	QTY.
1	Motor Housing	A513 Steel	1
2	Motor Attachment	-	1
3	Wheel Axle	304 Steel	1
4	Tire	-	1
5	Motor Attachment Bolt	-	1
6	Motor Mount	A36 Steel	1
7	0.5" bolts, long	-	3
8	Hitch Pin	-	1
9	Driveshaft	Steel	1
10	Key	-	1
11	Motor	-	1
12	0.5" nut	-	3
13	Motor Screw	-	4
14	Wheel Retaining Ring	-	1
15	Wheel Core		1
16	Set Screw	-	1
17	Hubcap	6061 Aluminum	2
18	Motor Nut	-	4

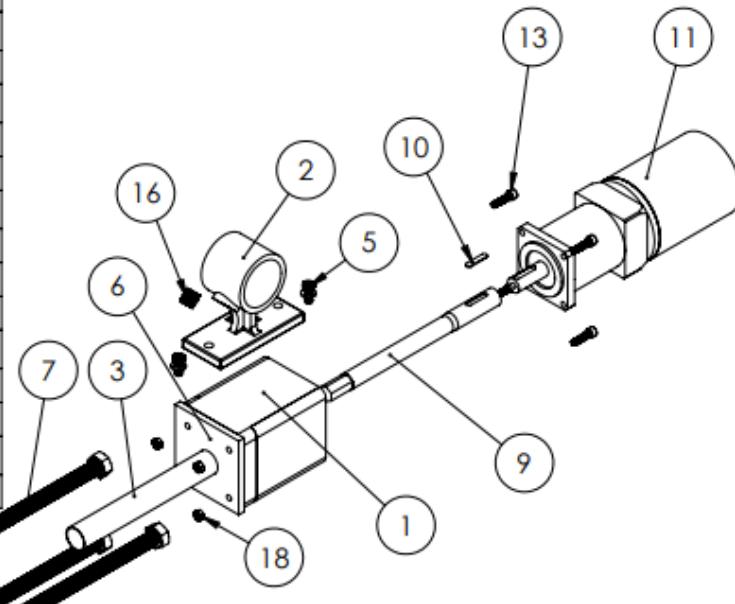
B

B



A

A



DRAWN	Michael Cannon	DATE	Project 57: L.O.U.I.S	
Last Revised	Michael Cannon	5/3/2024		
MATERIAL	See Bill of Materials			
NOTES: 1) All dimensions in inches				
2) Material for off the shelf parts not listed				
SIZE	DWG. NO.		REV	
A	SS2-A1		B	
SCALE:	WEIGHT:	lbm	SHEET	OF

2

1

Figure XIII-3: Exploded View Assembly Drawing of Sub-System SS2 - Drivetrain

XIII.C.2.a. *Comprehensive Parts List for SS2 - Drivetrain*

Table XIII-2: List of Parts for Sub-System SS2

Part #	Quantity	Material	Name
SS2-P1	1	A513 Steel	Motor Housing
SS2-P2	1	Cast Iron	Motor Attachment
SS2-P3	1	304 Steel	Wheel Axle
SS2-P4	1	Rubber	Tire
SS2-P5	1	Steel	Motor Attachment Bolt
SS2-P6	1	A36 Steel	Motor Mount
SS2-P7	3	Steel	0.5" bolts, long
SS2-P8	1	Steel	Hitch Pin
SS2-P9	1	Steel	Driveshaft
SS2-P10	1	Steel	Key
SS2-P11	1	Various	Motor
SS2-P12	3	Steel	0.5" nut
SS2-P13	4	Steel	Motor Screw
SS2-P14	1	Steel	Wheel Retaining Ring
SS2-P15	1	TPU Plastic	Wheel Core
SS2-P16	1	Steel	Set Screw
SS2-P17	2	6061 Aluminum	Hubcap
SS2-P18	4	Steel	Motor Nut



About this item

- 2PCS x 13" Beach Balloon Wheels: The 2PCS x 13" beach wheels make it easier than ever to transport your beach umbrellas, folding chairs, and kayaks to the beach. Its large and inflatable wheel design allows your cart to glide on the sand effortlessly. This wheel is suitable for axles with a 0.9-inch diameter. Please measure your cart's axle before purchase, then select the proper size you need.
- Robust TPU Construction: With heavy-duty TPU measuring at 13-inch diameter for the sand wheels, your sand cart will move easily and transport smoothly. Benefits also include less rolling resistance and better performance in crossing sand. It makes no noise when rolling up and down the ground, perfect for family beach trips, gardens, and camping. Its carrying load capacity is 121 lbs.
- Flexible Movement on All-terrain: With the help of our beach balloon wheels, your cart ensures a smooth pull and easy transportation on multiple ground surfaces, such as sand, gravel, and grass lawn. Our 2-3PSI low pressure balloon beach wheels will never go flat and easily roll over sand.
- Easy & Quick Air Inflation: We will considerably provide you with a hand air pump for saving labor, ensuring quick inflation. Before inflation, the balloon tire is 11.8-inch in diameter and 5.9-inch in tread, while after inflation, the diameter is 13-inch and the tread is 7.1-inch. Its small and light size makes it convenient to carry and store.
- Wide Application: These heavy-duty beach tires for carts are the perfect accessory for your next adventure. Whether you are going to the beach, having a picnic in the park, or off on a camping trip, the wheels will be sure to turn heads.

VEVOR Beach Balloon Wheels, 13" Replacement Sand Tires, TPU Cart Tires for Kayak Dolly, Canoe Cart and Buggy w/Free Air Pump, 2-Pack

Figure XIII-4: Specifications for part SS2-P4/P15 – Tire/Wheel Core

Material	Alloy Steel
Drive System	Internal Hex
Head Style	Dome
Item dimensions L x W x H	0.75 x 0.25 x 0.25 inches
Exterior Finish	Black Oxide

About this item

- Package Included: 20pcs
- Thread Length: 5/8" (not including head height) ; Head Height : 0.132 inch; Head Diameter: 0.437 inch .
- Thread Diameter: 1/4" (0.25") , Thread Pitch: 20 (0.05") .
- Allen Hex Drive Size: 5/32" (0.156")
- Material: Alloy Steel 10.9 Grade Black Oxide

KOSJETHAS 20Pcs 1/4-20 x 5/8" Button Head Socket Cap Screws Bolts, Allen Hex Drive, Black Oxide Alloy Steel 10.9 Grade, UNC Machine Fully Thread

Figure XIII-4: Specifications for part SS2-P5 – Motor Attachment Bolts

Material	Steel
Grade Rating	Grade A
Thread Size	1/2"-13
Exterior Finish	Zinc
Manufacturer Grade	Grade A

About this item

- Coarse Thread
- Superior Corrosion Resistance
- Hot Dipped Galvanized Finish
- Grade A
- Full Thread

Technical Details

Material	Steel
Grade Rating	Grade A
Thread Size	1/2"-13
Exterior Finish	Zinc
Manufacturer Grade	Grade A
Head Style	Hex
Color	Silver/Grey
Fastener Type	Hexagon Bolt
Brand	FASTENER DEPOT
Thread Type	Coarse Thread
Size	1/2"-13 x 7"
Number of Pieces	25
Manufacturer	Fastener Depot, LLC
Part Number	FD495953
Finish	Hot Dipped Galvanized
Tool Tip Description	Flat
Measurement System	Inch
Certification	Astm A307
Batteries Included?	No
Batteries Required?	No

1/2"-13 x 7" Hot Dipped Galvanized Hex Bolt, Grade A, Full Thread, Quantity 25 - by Fastener Depot, LLC

Figure XIII-5: Specifications for part SS2-P7 – 0.5” bolts, long

Thread Size: **1/2"-13**

Item Width String: **3/4 inches**

Item Height String: **7/16 Inches**

Number of Items: **50**

Material Steel

Fastener Type Hex

Thread Size 1/2"-13

Exterior Finish Hot-Dipped Galvanized

Metal Type Alloy Steel

About this item

- Hex nut tightens from the side with a wrench, enabling use when there is limited space above the nut
- Grade 2 steel is often used in applications where toughness is the primary consideration
- Hot-dipped galvanized finish offers excellent corrosion resistance
- Meets ASME B18.2.2 specifications

Steel Hex Nut, Hot-Dipped Galvanized Finish, Grade 2, ASME B18.2.2, 1/2"-13 Thread Size, 3/4" Width Across Flats, 7/16" Thick (Pack of 50)

Figure XIII-6: Specifications for part SS2-P12 – 0.5" nut



Roll over image to zoom.

Structural Pipe Fitting: Support, 1 1/4 in For Pipe Size, For 1 5/8 in Actual Pipe Outer Dia, Pipe

Item 30LX62 Mfr. Model 30LX62

Product Details

Catalog Page [1268](#)

Brand **APPROVED VENDOR**

Finish **Zinc-Plated**

For Pipe Size **1-1/4 in**

Item/Primary Noun **Rail Support**

Component Type **Pipe**

Overall Length **4 in**

Inside Diameter **1-3/4 in**

Manufacturer Part Number **30LX62**

Fitting Shape **Closed Rail Support**

For Actual Pipe Outer Diameter **1-5/8 in**

Material **Cast Iron**

UNSPSC **30191601**

Color **Gray**

Country of Origin **India (subject to change)**

Corrosion Resistant **Yes**

Product Description

Cast iron mounts secure slide-on railing and framing structures to a surface such as a floor or wall. Cast iron fittings have a higher tensile strength than aluminum fittings. These fittings have a galvanized finish that provides some rust resistance, making them suitable for indoor or outdoor use.

Figure XIII-7: Specifications for part SS2-P2/P16 – Motor Attachment/Set Screw



binifiMux 1/4-20 x 1" Hex Bolts Screws, with 1/4-20 Nylon Inserted Lock Nuts / 1/4 Washers / 1/4-20 Hex Nuts, 304 Stainless Steel /A2-70, Silver Tone, 50pcs

Size	1/4-20" x 1" (10-Pack)
Color	Silver
Material	Stainless Steel
Brand	binifiMux
Fastener Type	Hexagon Bolt, Lock Nuts, Hex Nuts, Washers

About this item

- [Heart] Name: 1/4-20 UNC Hex Bolts; [Heart] Material: 304 Stainless Steel ;
- [Heart] Full Thread; [Heart] Coarse Thread: 1/4 Inch;
- [Heart] Package: 10pcs Hex Head Screws, 10pcs 1/4-20 Hex Nuts, 10pcs 1/4-20 Nylon Lock Nuts, 10pcs 1/4 Flat Washers, 10pcs 1/4 Spring Washers, all store in one storage box;;
- [Heart] Hex Face Dia.: 11mm; [Heart] Hex Part Thickness: 4mm;
- [Heart] Typically used to create a bolted joint, in which a threaded shaft exactly fits a corresponding tapped hole or nut.

Figure XIII-8: Specifications for part SS2-P13/P18 – Motor Screw/Motor Nut



Highlights

- 1-piece per pack
- Steel construction
- Zinc-plated
- 1/2 in.

Product Information

Internet # 204276264
Model # 809598
Store SKU # 570497

Specifications

Dimensions: H 2 in, W 1 in, D 1 in

Dimensions

Product Depth (in.)	1 in	Product Height (in.)	2 in
Product Width (in.)	1 in		

Details

Color Family	Metallics	Fastener Type	Specialty Fastener
Package Quantity	1	Product Weight (lb.)	0.01 lb
Returnable	90-Day	Type	Safety Pin

Figure XIII-9: Specifications for part SS2-P8 – Hitch Pin

Retaining Ring Type	External
Retaining Ring Style	Standard
System of Measurement	Inch
Material	1060-1090 Spring Steel
Finish	Black Phosphate
For OD	7/8"
For Groove	
Diameter	0.821"
Diameter Tolerance	-0.003" to 0.001"
Width	0.086"
Width Tolerance	0" to 0.005"
Ring	
ID	0.804"
ID Tolerance	-0.01" to 0.005"
Thickness	0.078"
Thickness Tolerance	-0.003" to 0.003"
Min. Hardness	Rockwell C47
Thrust Load Capacity	10,500 lbs.
Magnetic Properties	Magnetic
Specifications Met	ASME B18.27.2
RoHS	RoHS 3 (2015/863/EU) Compliant
REACH	REACH (EC 1907/2006) (06/14/2023, 235 SVHC) Compliant
DFARS	Specialty Metals COTS-Exempt
Country of Origin	United States
USMCA Qualifying	No
Schedule B	731824.0000
ECCN	EAR99

Figure XIII-10: Specifications for part SS2-P14 – Wheel Retaining Ring

	High Performance Economy Motors					
Voltage Range	6 Volts to 18 Volts		12 Volts to 36 Volts		24 Volts to 72 Volts	
Model	E30-400-12	E30-150-12	E30-400-24	E30-150-24	E30-400-48	E30-150-48
Diameter (inches)	3.1	3.1	3.1	3.1	3.1	3.1
Length (inches)	5.8	4.0	5.8	4.0	5.8	4.0
Peak HP	1.6	0.6	2.1	1.0	2.7	1.1
Stall Torque (oz-in)	970	420	1500	710	1860	750
Efficiency	78%	72%	79%	76%	82%	77%
Nominal Voltage	12V ¹	12V ¹	24V ¹	24V ¹	48V ¹	48V ¹
RPM at Nominal Voltage	6500	5400	5700	5600	5800	5600
Shaft Diameter (inches)	1/2	1/2	1/2	1/2	1/2	1/2
Shaft Length (inches)	2.0	2.0	2.0	2.0	2.0	2.0
Keyway (inches)	1/8	1/8	1/8	1/8	1/8	1/8
Capacitors	No	No	No	No	No	No
Magnet Type	Ferrite	Ferrite	Ferrite	Ferrite	Ferrite	Ferrite
No Load Amps	5.5	3.5	3.2	2.1	1.4	1.0
Resistance (Ohms)	0.030	0.081	0.089	0.190	0.284	0.708
Kt (oz-in/Amp)	2.46	2.93	5.63	5.70	11.1	11.4
Kv (RPM/Volt)	549	461	240	237	122	118
Weight (pounds)	5.9	3.6	5.9	3.6	5.9	3.6
Voltage Range	6 Volts to 18 Volts		12 Volts to 36 Volts		24 Volts to 72 Volts	
Model	E30-400-12	E30-150-12	E30-400-24	E30-150-24	E30-400-48	E30-150-48
Peak Efficiency (PE)	78%	72%	79%	76%	82%	77%
RPM at PE	5800	4700	5100	5000	5300	5000
Torque at PE (oz-in)	105	57	145	85	150	80
Horsepower at PE	0.60 HP	0.26 HP	0.75 HP	0.40 HP	0.80 HP	0.40 HP
Current at PE	48 Amps	23 Amps	29 Amps	17 Amps	15 Amps	8 Amps

Figure XIII-11: Specifications for part SS2-P11/P10 – Motor/Key

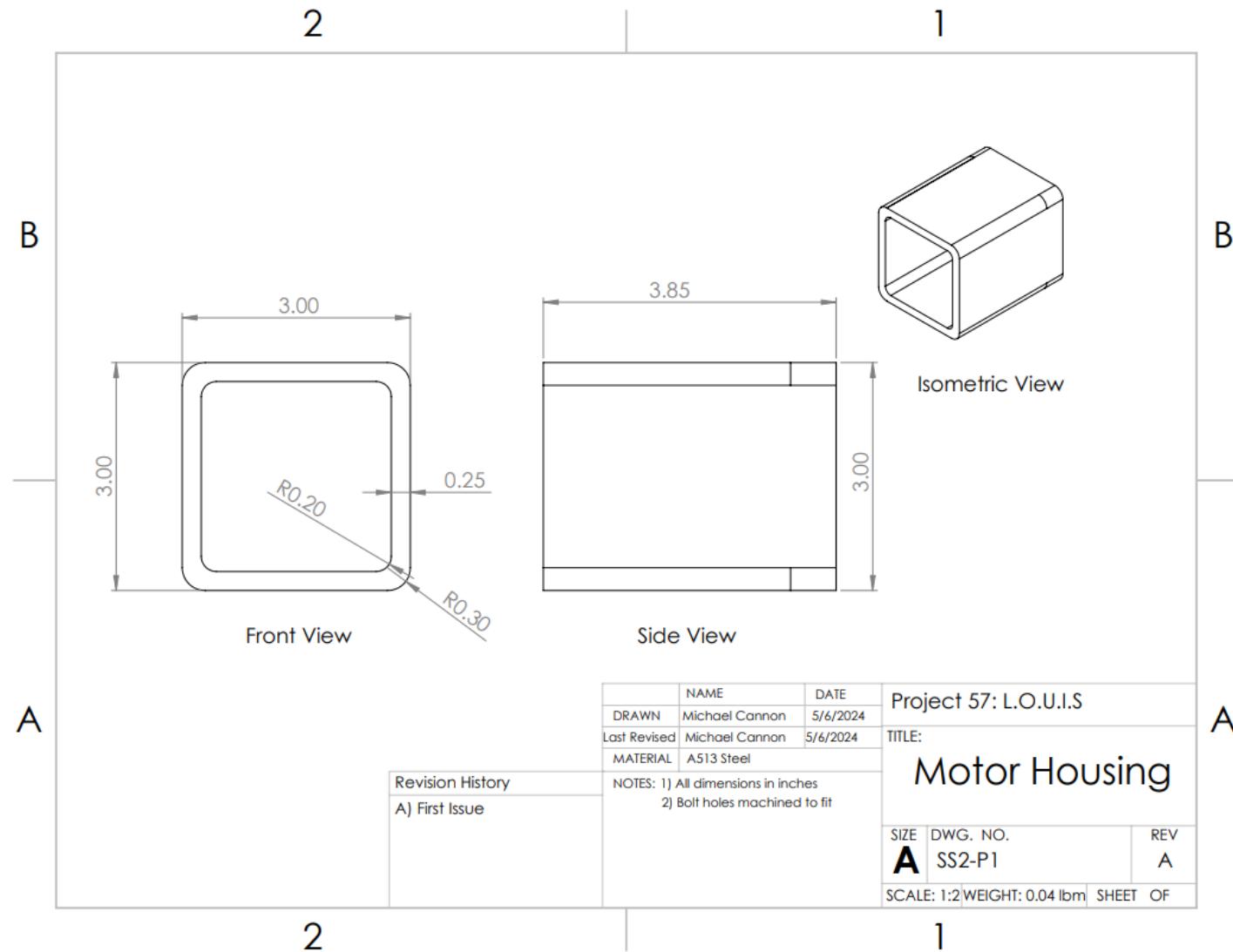


Figure XIII-12: Manufacturing Drawing of part SS2-P1- Motor Housing

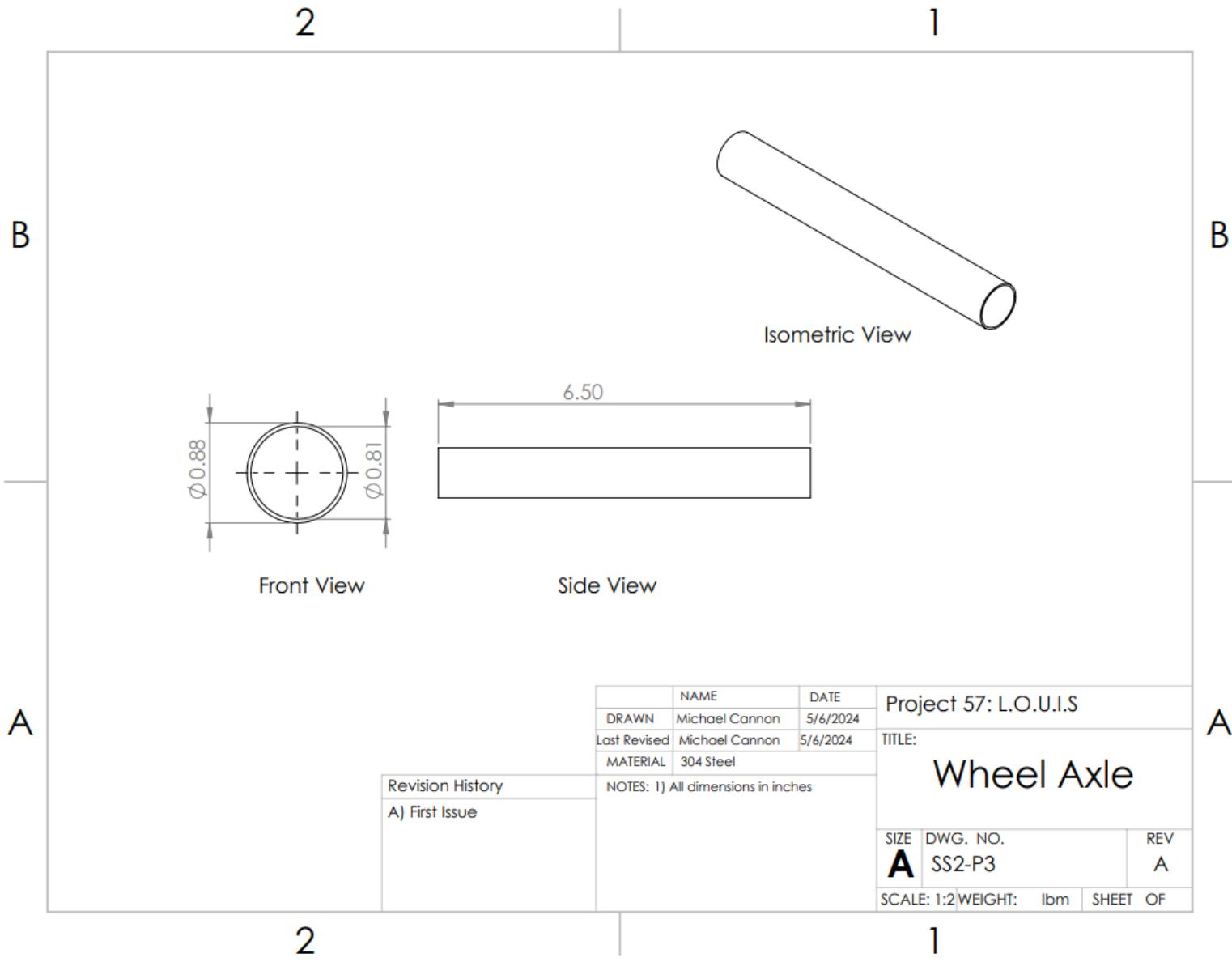


Figure XIII-13: Manufacturing Drawing of part SS2-P3- Wheel Axle

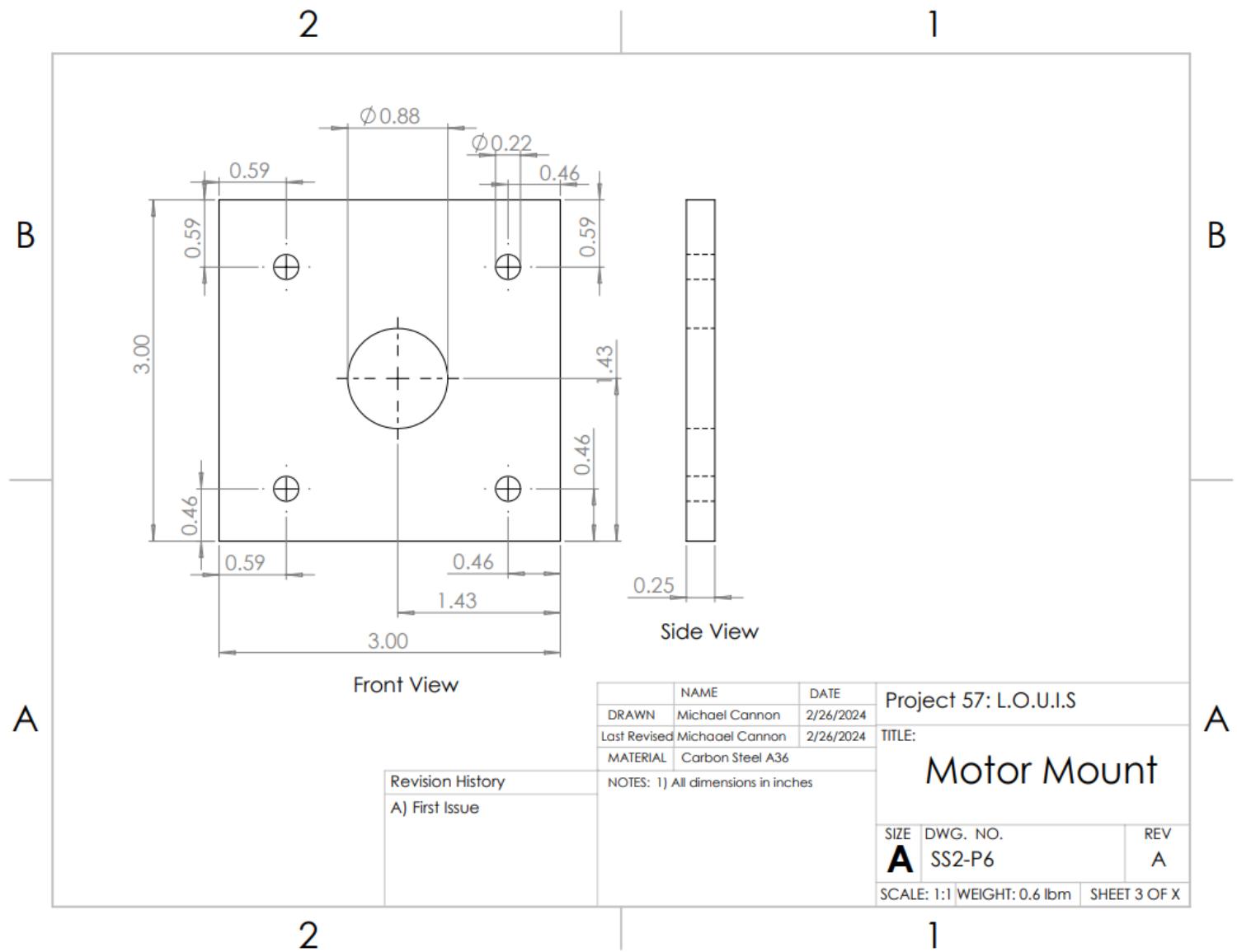


Figure XIII-14: Manufacturing Drawing of part SS2-P6- Motor Mount

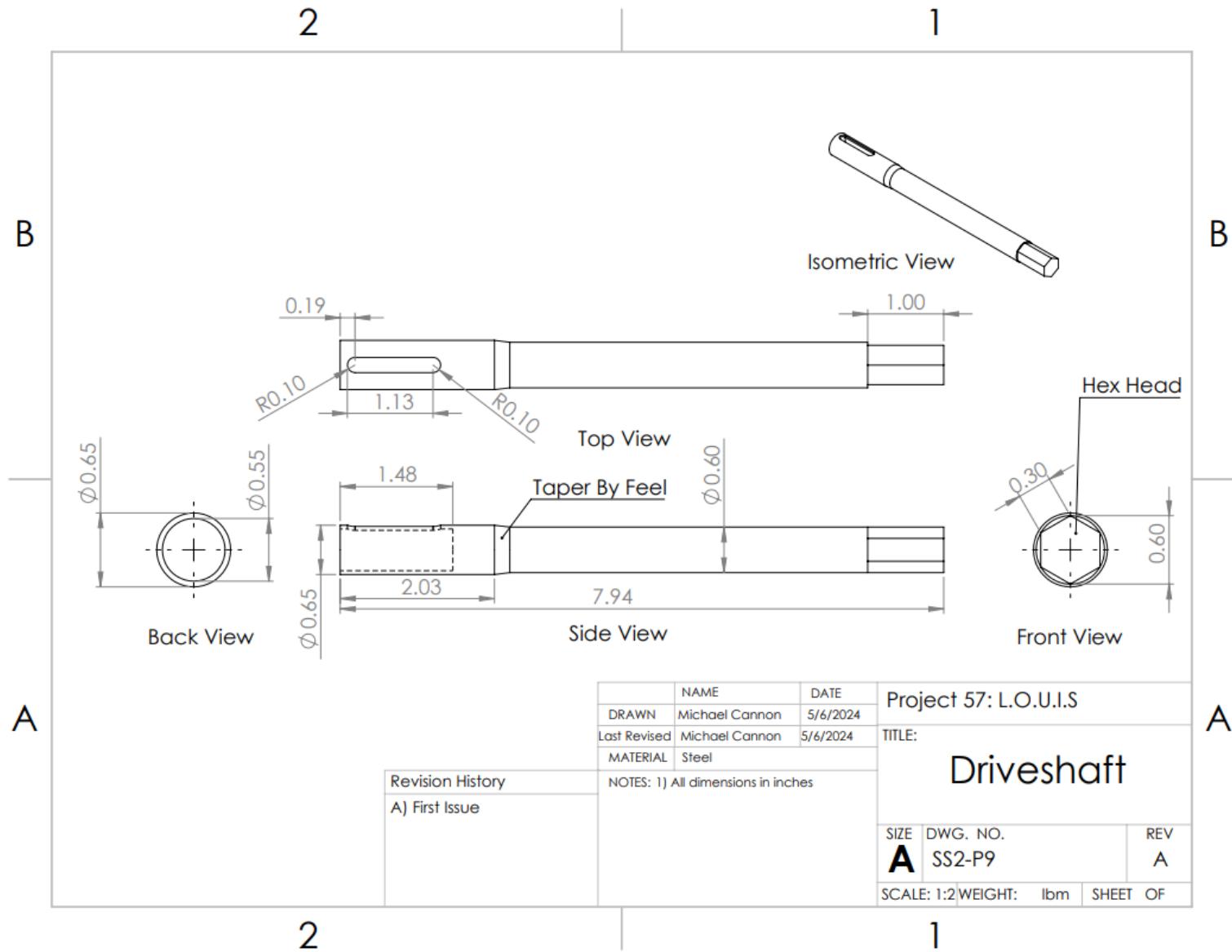


Figure XIII-15: Manufacturing Drawing of part SS2-P9- Driveshaft

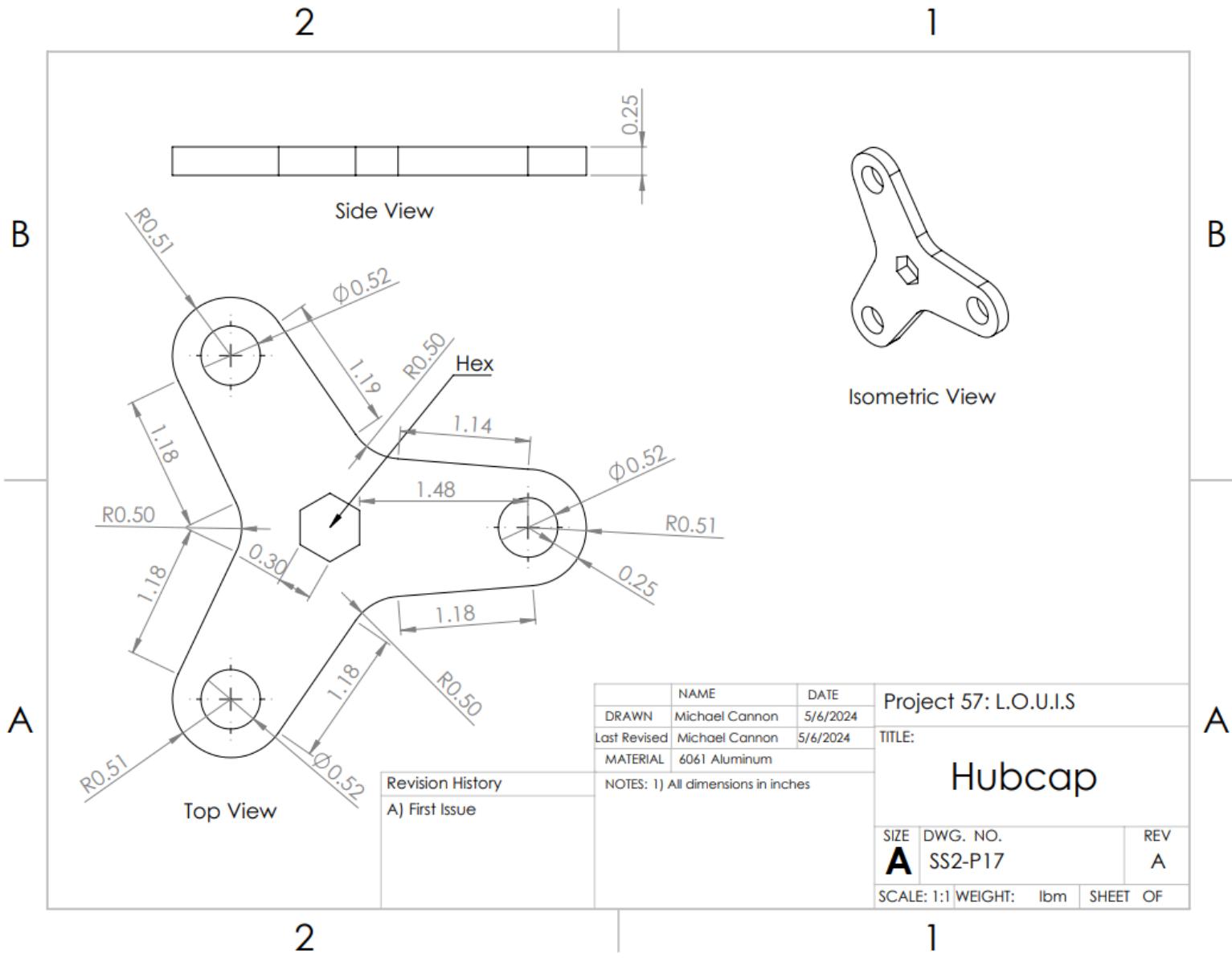


Figure XIII-16: Manufacturing Drawing of part SS2-P17- Hubcap

XIII.C.3. Sub-System SS3 - Legs

The legs utilize of a rocker-bogie design. This is a 2 DOF system. The rocker rotates about the main axle and holds the front wheel and the bogie. The bogie rotates around a secondary axle at the end of the rocker and holds the back two wheels. This system allows all the wheels to remain in contact with the ground while moving over rough terrain. The differential system screws into the side of the front leg. Two brackets bolted to the end of the rocker form the bearing that holds the bogie axle. The advantage of keeping all wheels in contact with the ground is that they can all contribute to forward or rotational motion. At the ends of the legs are brackets which attach the leg system to the wheel system.

The legs are held in place on the frame axle by retaining rings. Removing the outer retaining ring allows the full rocker-bogie to be removed from the axle. The rod-end joint will also have to be unbolted and removed from the rocker. Note: To completely disconnect the legs, the motor wires will have to be unscrewed from the motor controllers. To disconnect the bogie from the axle, remove the two half-inch bolts and nuts. The two Rocker-bogie brackets and spacers can be taken off the bogie axle. The wheel system (SS2) can be removed by unscrewing the motor attachment bolts. The Leg-motor bracket can be removed or adjusted by loosening the set screw.

Several additions were made during the manufacturing process. Due to a build up of weld material on the bogie, a spacer was designed to move the RB (rocker-bogie) brackets further away from the bogie. The ID of the Leg-motor brackets was larger than expected. This necessitated the manufacturing of a series of bushings to fill up the extra space. The rocker bearing was replaced with a thicker version after the original had a brittle fracture due to post-weld weakening. This was done on both rockers just to be safe. It was discovered part was through manufacturing that the waterjet cuts holes with a slight taper. This, combined with excessive grinding on the bogie axle, causes the rocker-bogie connection to be slightly looser than desired. This led to the back four wheels fanning out slightly while driving. While not optimal, this did not impede the function of the legs or drivetrain.

The team has come up with a potential fix but has not had time to implement it. By loosening the set screw, the wheel assemblies can be rotated about the legs. This lets the wheels sit straight relative to the ground but misaligns them with the direction of travel. By offsetting the leg-motor bracket with the motor housing and drilling a new screw hole at the new location, the wheel could be angled in the right direction.

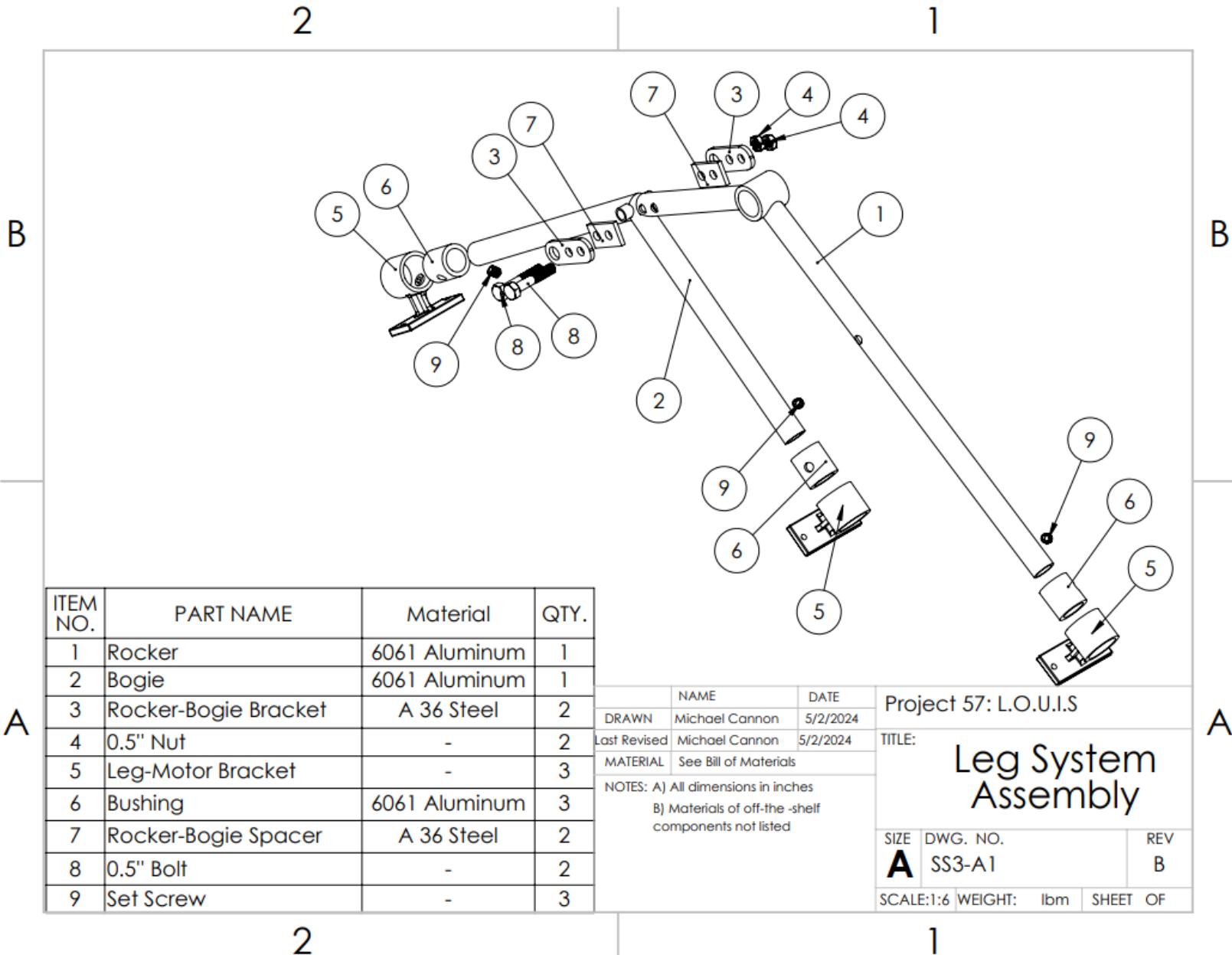


Figure XIII.C.3-1: Exploded View Assembly Drawing of Sub-System SS3 - Legs

XIII.C.3.a. *Comprehensive Parts List for SS3 - Legs*

Table XIII.C.3-3: List of Parts for Sub-System SS3

Part #	Quantity	Material	Name
SS3-P1	1	6061 Aluminum	Rocker
SS3-P2	1	6061 Aluminum	Bogie
SS3-P3	2	A36 Steel	Rocker-Bogie Bracket
SS3-P4	2	Steel	0.5" Nut
SS3-P5	3	Cast Iron	Leg-Motor Bracket
SS3-P6	3	6061 Aluminum	Bushing
SS3-P7	2	A36 Steel	Rocker-Bogie Spacer
SS3-P8	2	304 Steel	0.5" Bolt
SS3-P9	3	Steel	Set Screw

XIII.C.3.b. Off-The-Shelf Parts and Component Specifications for SS3 - Legs



Roll over image to zoom.

**Structural Pipe Fitting: Support, 1 1/4 in For Pipe Size,
For 1 5/8 in Actual Pipe Outer Dia, Pipe**

Item 30LX62 Mfr. Model 30LX62

Product Details

Catalog Page 1268

Brand APPROVED VENDOR

Finish Zinc-Plated

For Pipe Size 1-1/4 in

Item/Primary Noun Rail Support

Component Type Pipe

Overall Length 4 in

Inside Diameter 1-3/4 in

Manufacturer Part Number 30LX62

Fitting Shape Closed Rail Support

For Actual Pipe Outer Diameter 1-5/8 in

Material Cast Iron

UNSPSC 30191601

Color Gray

Country of Origin India (subject to change)

Corrosion Resistant Yes

Product Description

Cast iron mounts secure slide-on railing and framing structures to a surface such as a floor or wall. Cast iron fittings have a higher tensile strength than aluminum fittings. These fittings have a galvanized finish that provides some rust resistance, making them suitable for indoor or outdoor use.

Figure XIII.C.3-2: Specifications for part SS3-P5/P9 – Leg-Motor Bracket/Set Screw

Thread Size: **1/2"-13**

Item Width String: **3/4 inches**

Item Height String: **7/16 Inches**

Number of Items: **50**

Material Steel

Fastener Type Hex

Thread Size 1/2"-13

Exterior Finish Hot-Dipped Galvanized

Metal Type Alloy Steel

About this item

- Hex nut tightens from the side with a wrench, enabling use when there is limited space above the nut
- Grade 2 steel is often used in applications where toughness is the primary consideration
- Hot-dipped galvanized finish offers excellent corrosion resistance
- Meets ASME B18.2.2 specifications

Figure XIII.C.3-3: Specifications for part SS3-P4 – 0.5" Nut

Material	Stainless Steel
Thread Size	1/2"-13
Exterior Finish	Stainless Steel
Manufacturer Grade	Grade 304
Head Style	Hex

About this item

- Thread Size: 1/2-13; Thread Pitch: 13 TPI; Thread Type: Coarse Thread/ UNC; Thread Style: Right Hand
- Screw Length: 2 inch or 50mm (measured excluding the head height); Head Height: 0.323 inch (8.2mm); Width Across the Flat Head: 0.75 inch (19.1mm)
- Hex Drive Size: 3/4"; Package Includes: 10 PCS
- Strong enough for indoor and outdoor projects. Hex bolts generally requires washers and nuts to distribute the pressure, as well as preventing pull through. Eastlo provides stainless steel nuts and washers separately or in sets with good quality and reasonable price.
- Made of 304 Stainless Steel (no coating outside) which is superior to regular steel and any zinc plated fasteners, provides good strength and corrosion resistance in bad environments

Product Description

SIZE: 1/2-13 x 2"

Specification:

Thread Diameter 0.5"
Head Width Measured From Flat to Flat 3/4"
Screw Length 2 inch

Package Included:

10 PCS Hex Head Cap Screws

About Hex Bolts

Hex head cap screws are precision threaded fasteners with an External socket drive. They are known for their efficient application performance and endurance widely used in a variety of industries such as construction, manufacturing, engineering, assembling, aircraft and so on.

The high quality 304 stainless steel provide perfect strength and good corrosion resistance in bad environments like high temperature, cold weather and high humidity areas without the need for time consuming and costly maintenance.

Eastlo fastener hex cap screws meet the specific tolerance as described under ASME B18.2.1-1996 and guarantee to out live and out last all other grades of hardware where exposed to outdoor elements.

Figure XIII.C.3-4: Specifications for part SS3-P8 – 0.5" Bolt

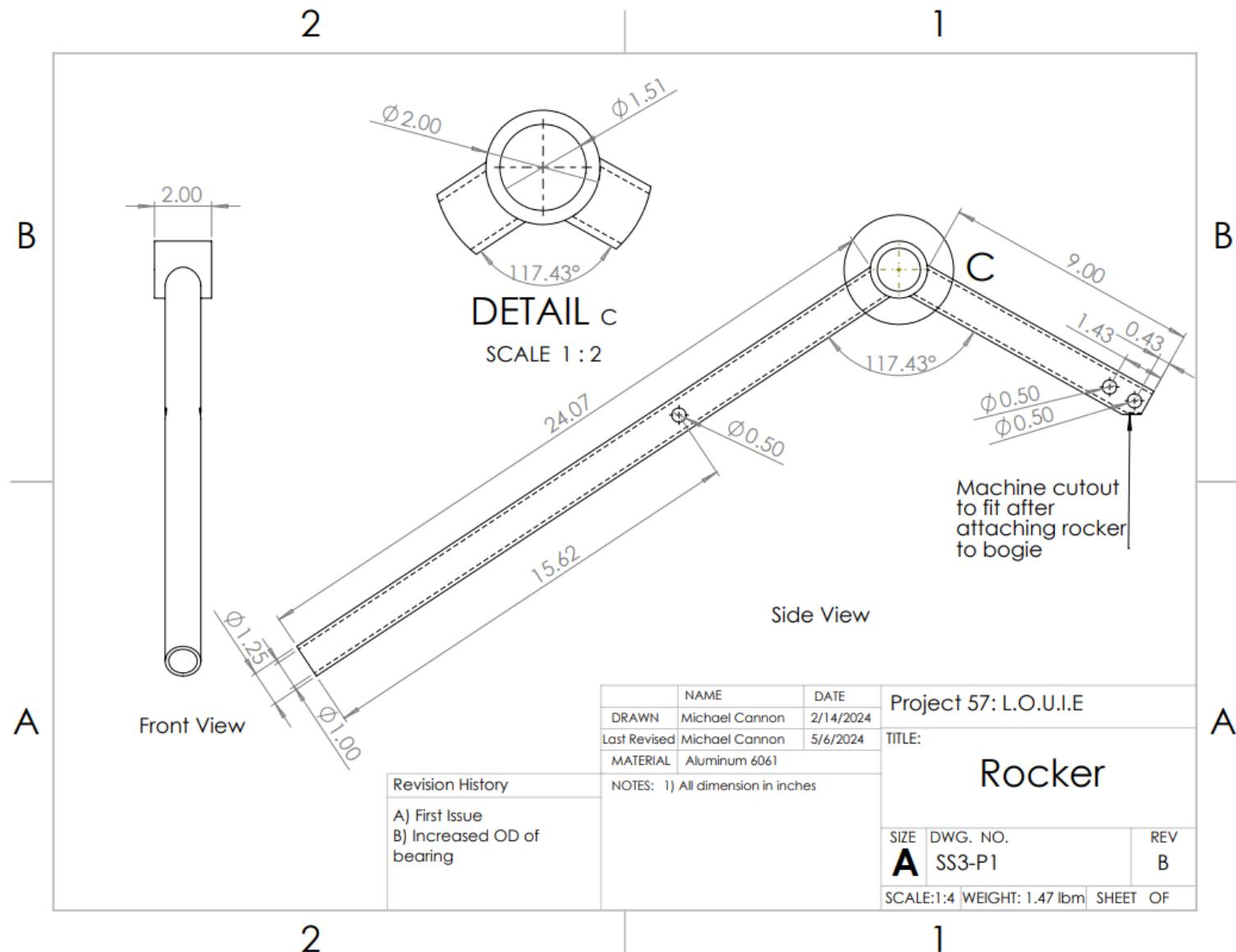


Figure XIII.C.3-5: Manufacturing Drawing of part SS3-P1- Rocker

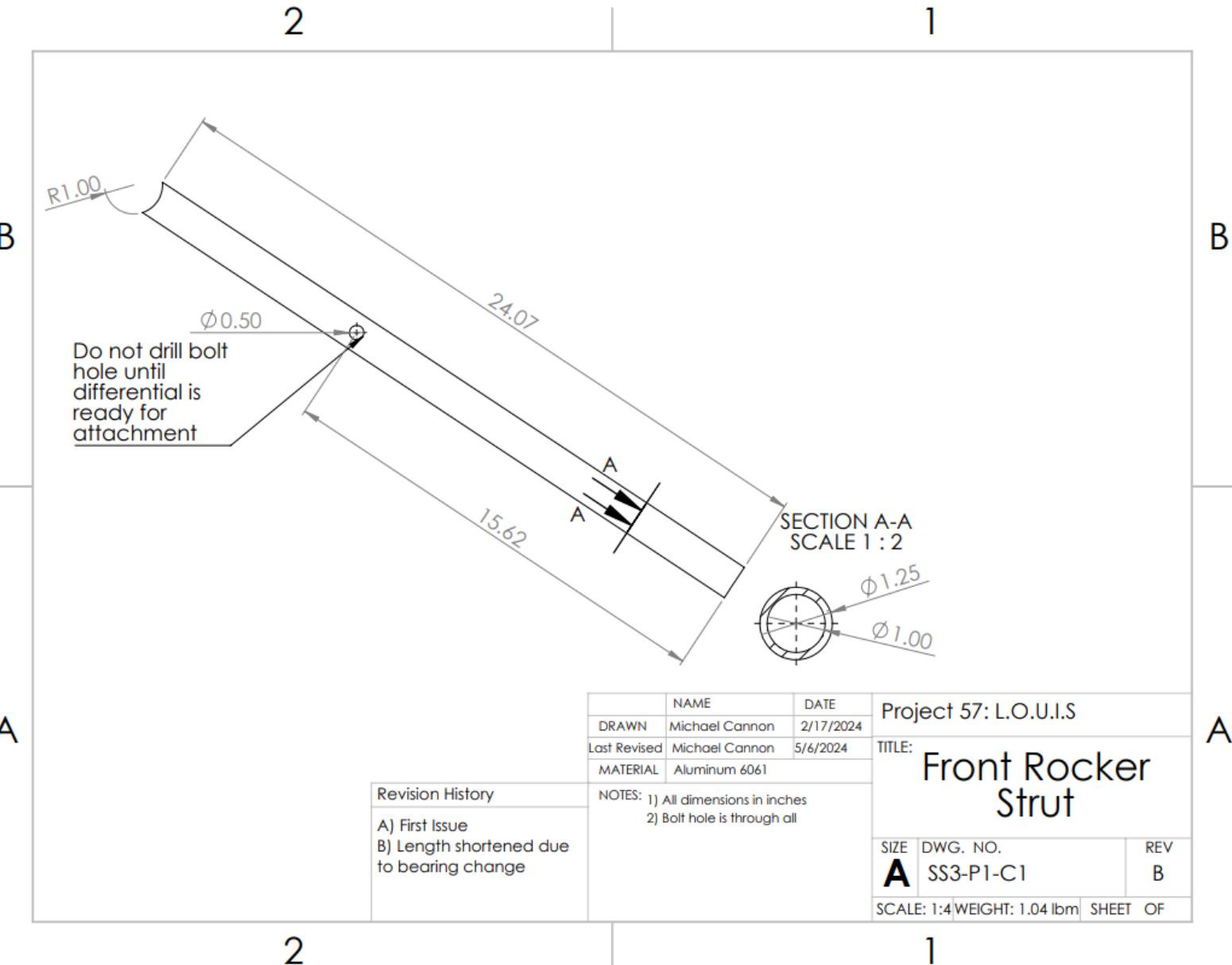


Figure XIII.C.3-6: Manufacturing Drawing of part SS3-P1-C1- Rocker Front Strut

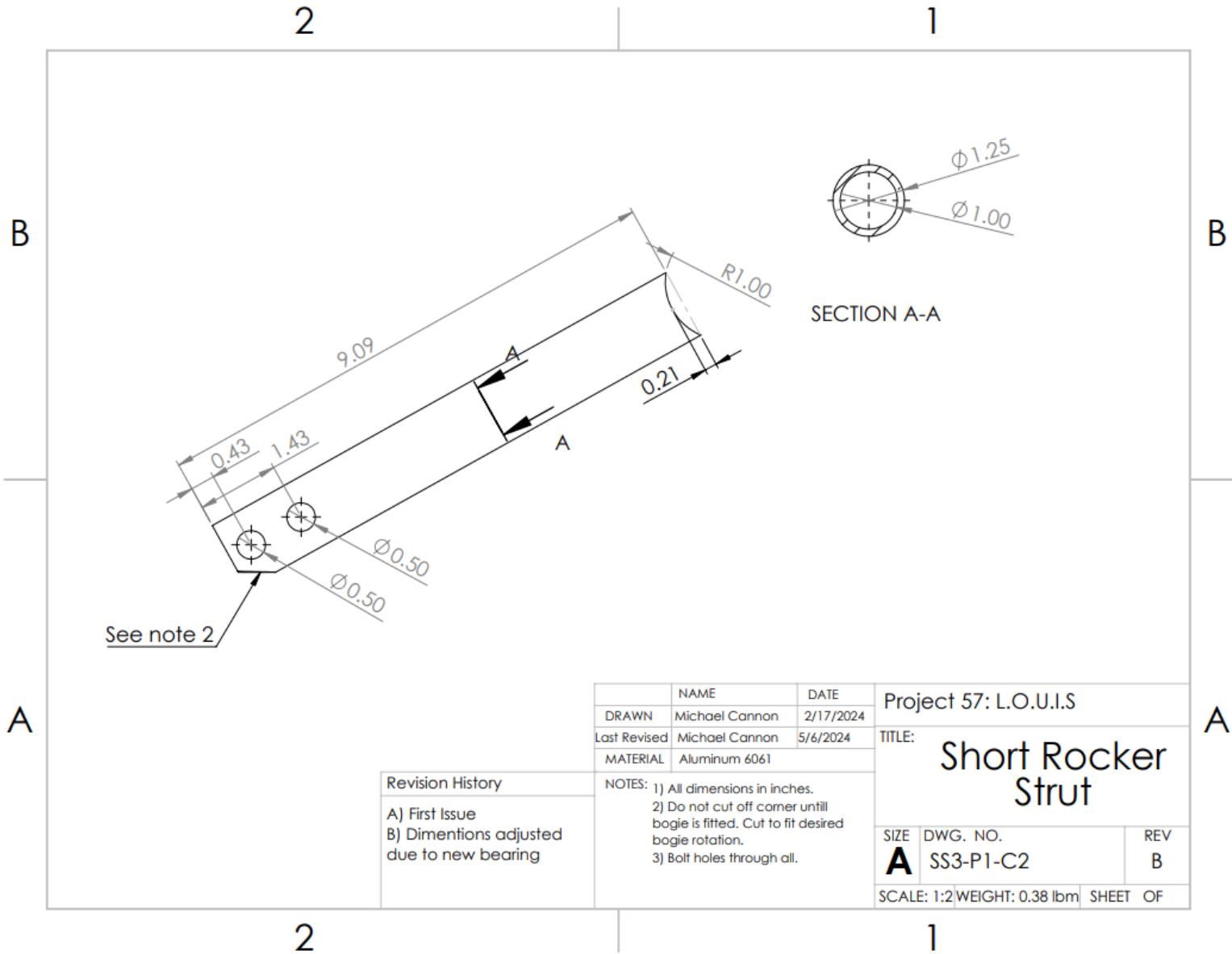


Figure XIII.C.3-7: Manufacturing Drawing of part SS3-P1-C2- Rocker Short Strut

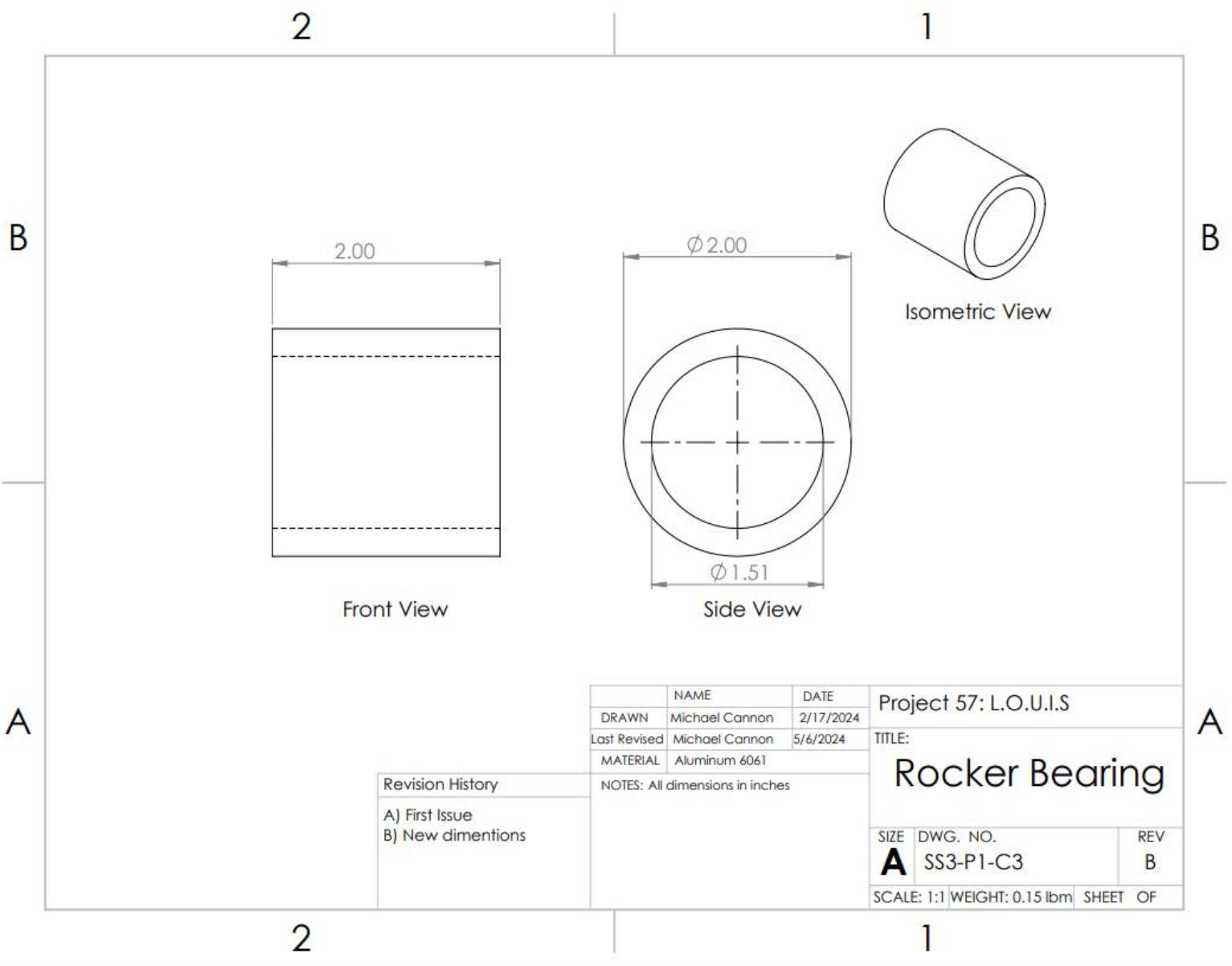


Figure XIII.C.3-8: Manufacturing Drawing of part SS3-P1-C3- Rocker Bearing

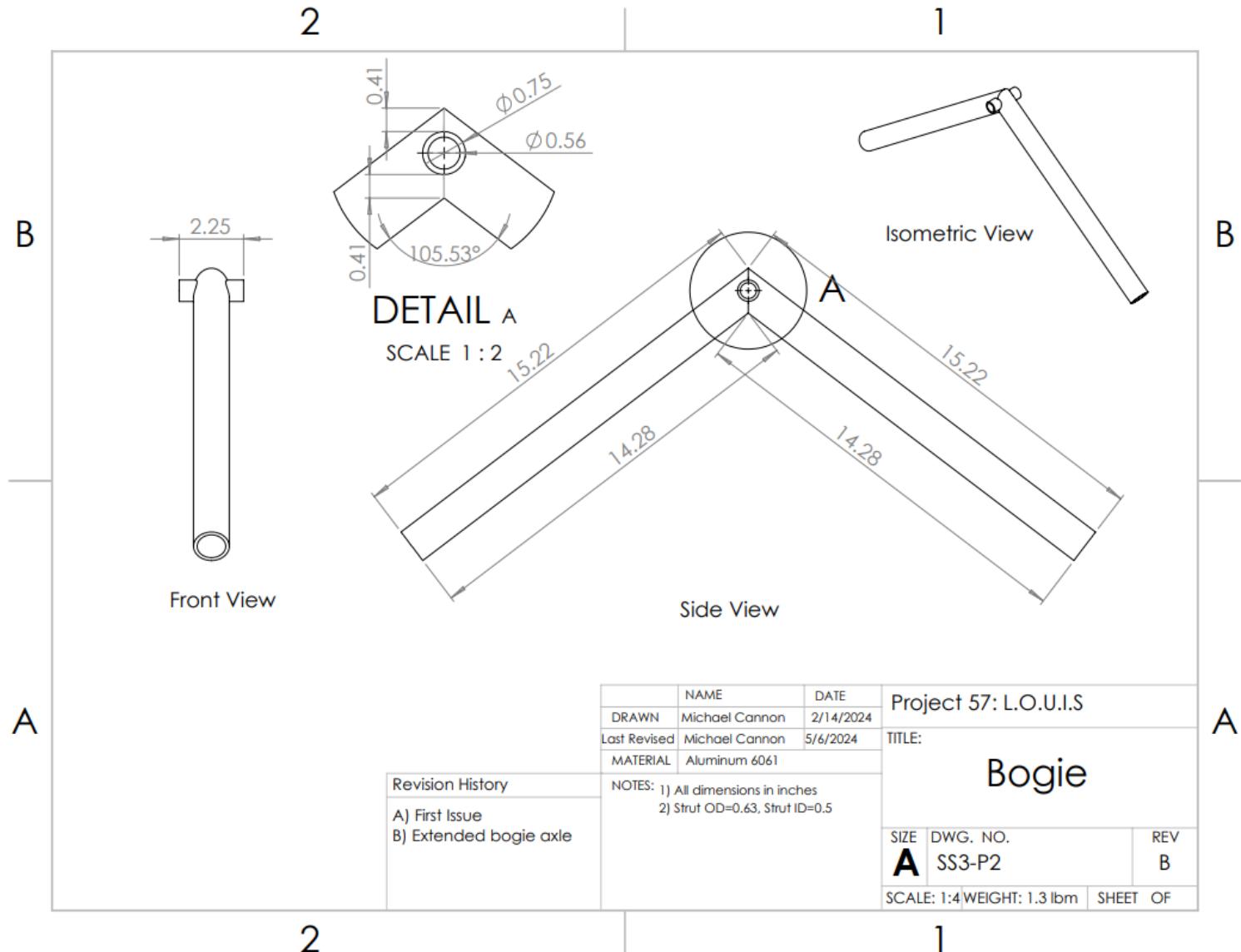


Figure XIII.C.3-9: Manufacturing Drawing of part SS3-P2- Bogie

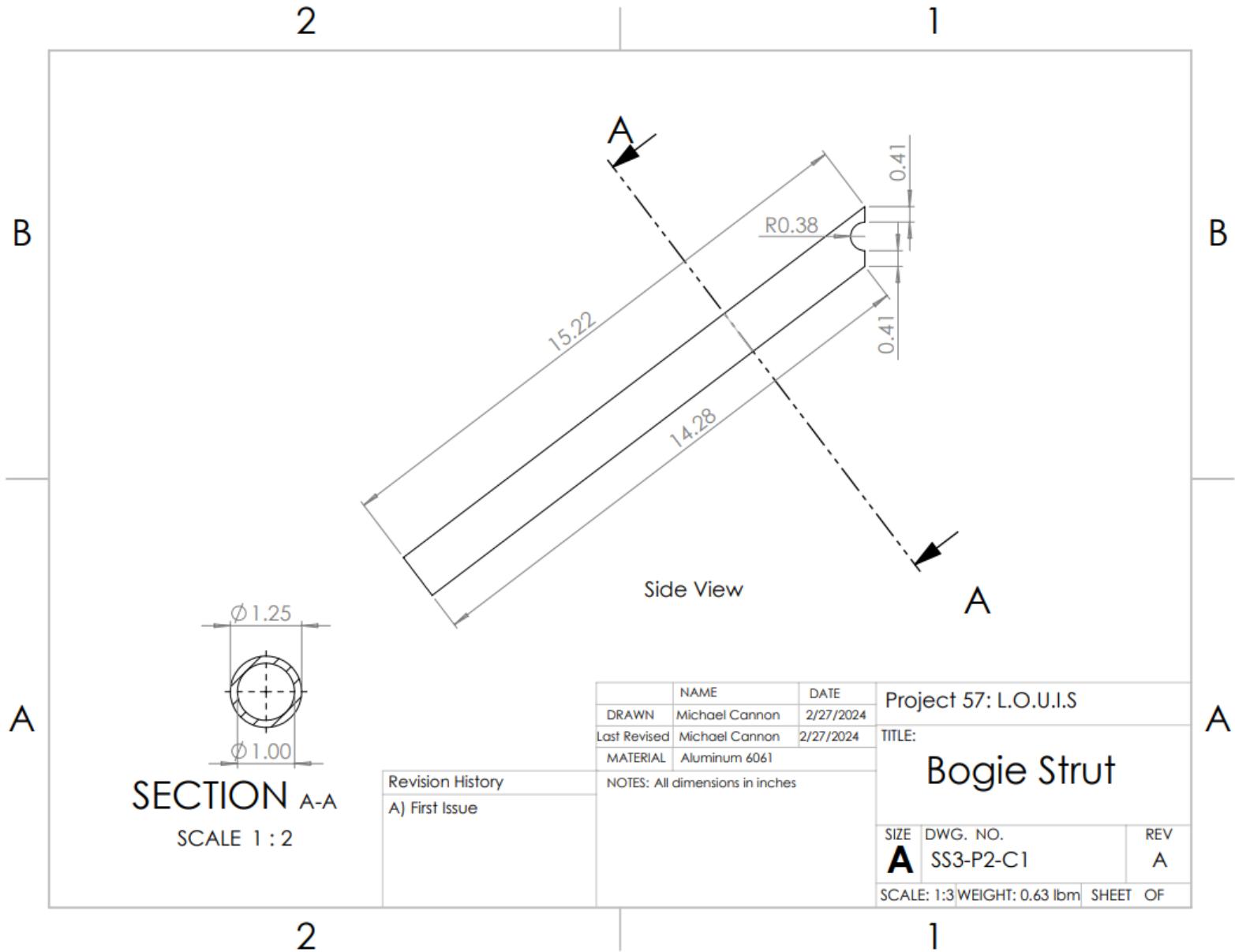


Figure XIII.C.3-10: Manufacturing Drawing of part SS3-P2-C1- Bogie Strut

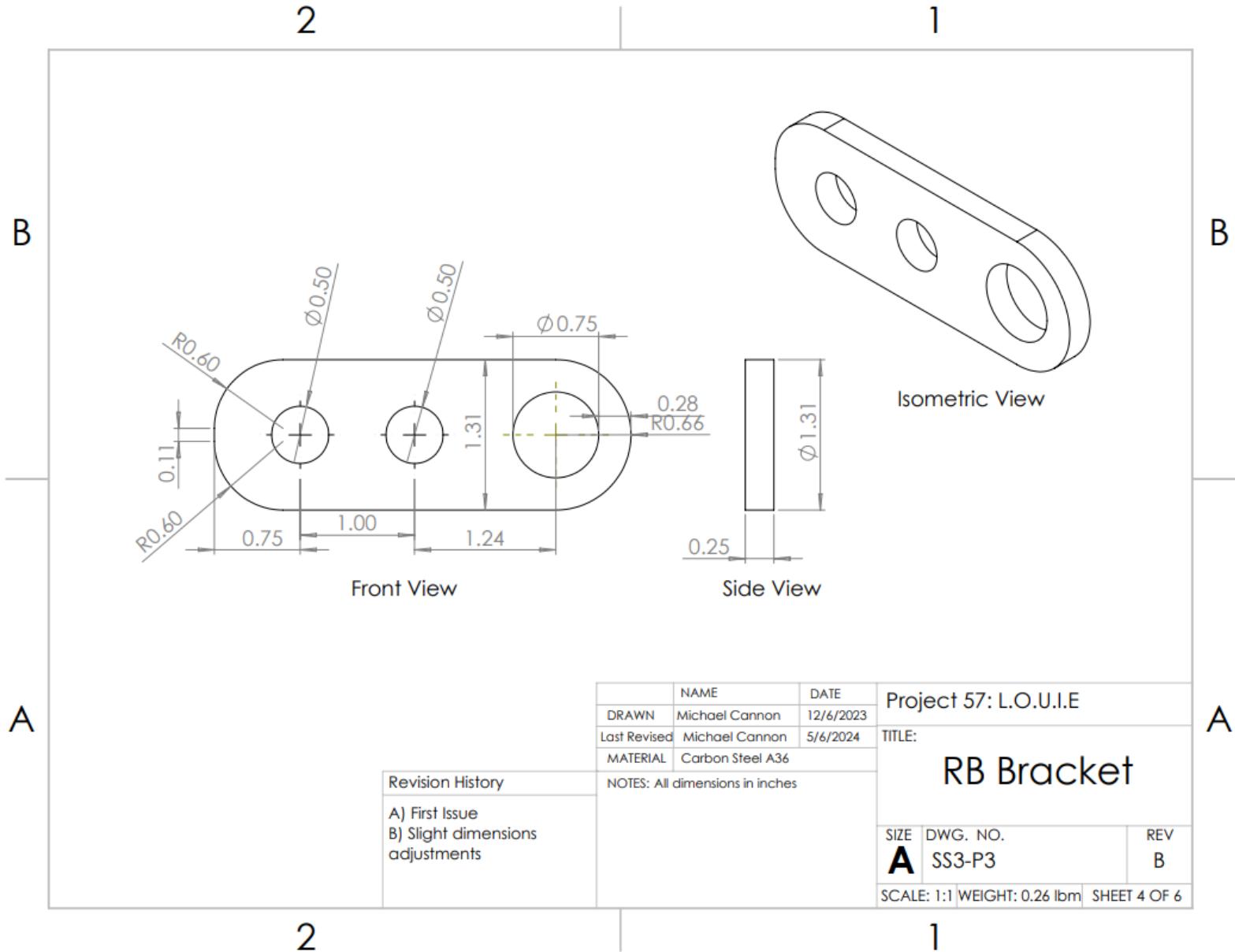


Figure XIII.C.3-11: Manufacturing Drawing of part SS3-P3- Rocker-Bogie Bracket

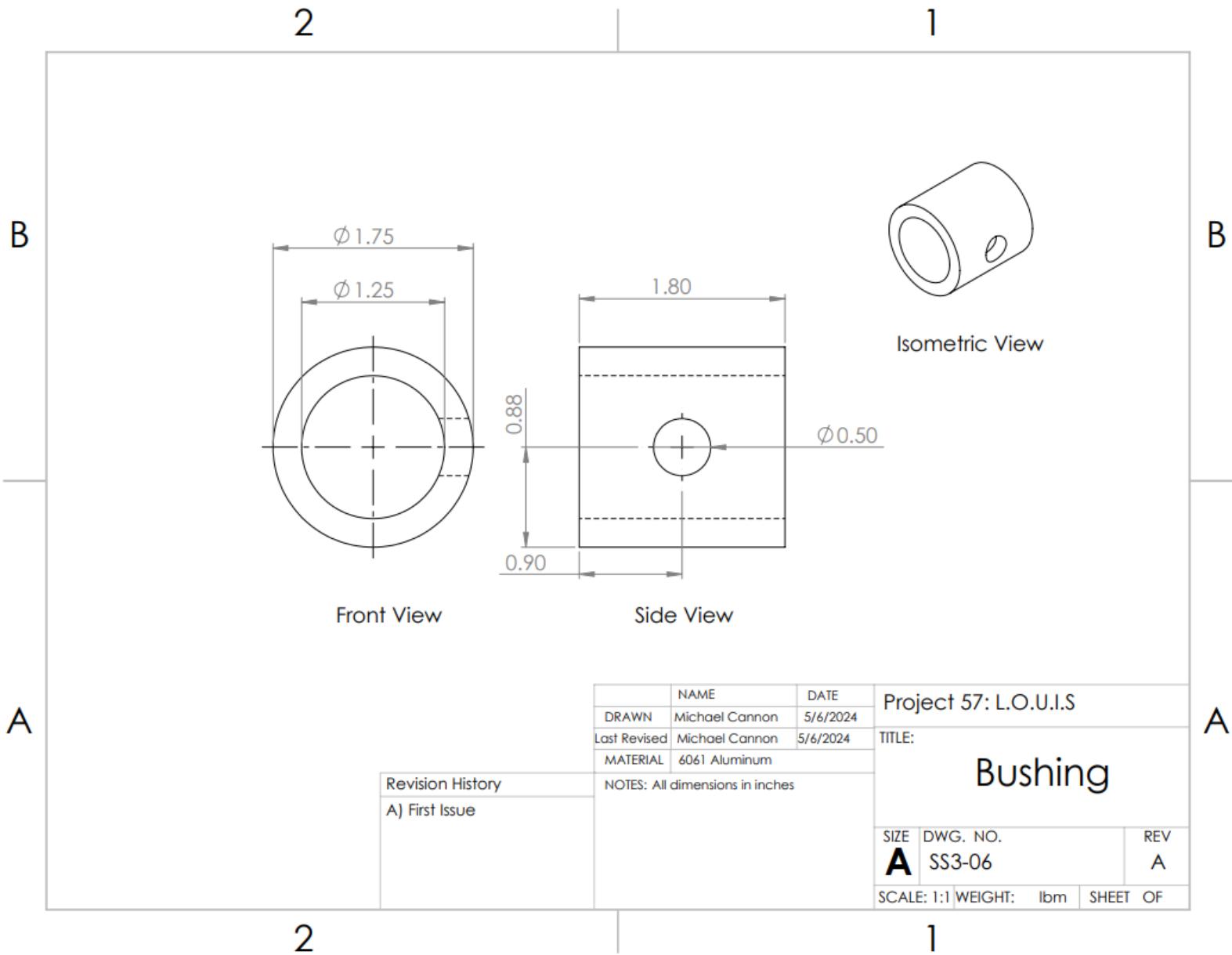


Figure XIII.C.3-12: Manufacturing Drawing of part SS3-P6- Bushing

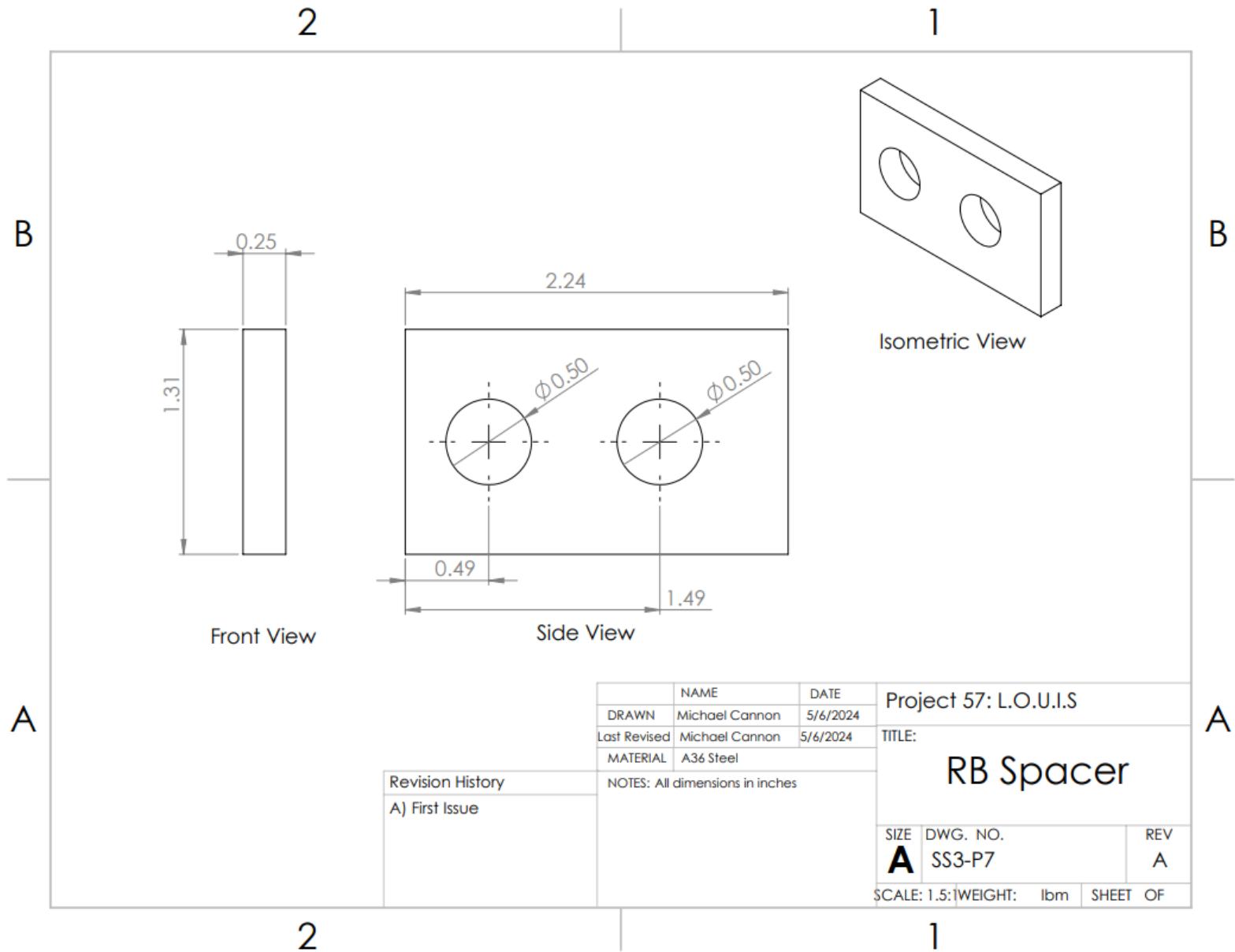


Figure XIII.C.3-13: Manufacturing Drawing of part SS3-P7- Rocker-Bogie Spacer

XIII.C.4. Sub-System SS4 – Differential Suspension

The primary function of the differential is to provide a second point of contact between the frame and the legs. Without the differential, the frame would freely rotate about the axle whenever the center of mass was off center. The differential is connected to each of the legs by a strut. If this were a fixed system, then the differential would prevent the legs from rotating independently. This would cause one or more wheels to come out of contact with the ground. To solve this the differential bar rotates about an axis and the struts are connected on each end with ball joints. This allows for the legs to move counter to one another as the rover moves across rough terrain. The differential is a one DOF system. The limits of the system are determined by the limits of rotation of the differential bar and the max ball swivel angle. These values are discussed in the analysis section. The maximum difference in front wheel height allowed by the differential system is 10 inches.

A few notes on the differential performance. Overall, the differential performed its function exactly as desired. The bolt holes on the differential bar were cut with the waterjet. Unlike holes cut on an end mill, the waterjet cuts with a taper that makes tight fits hard to achieve. This gave the differential the ability to rotate forward and backward by a few degrees. This made the system sit at an awkward angle when under compression but also gave it more degrees of freedom and therefore more range. The nuts on the struts can loosen under repeated vibration. This can be solved by replacing the regular nuts with lock nuts.

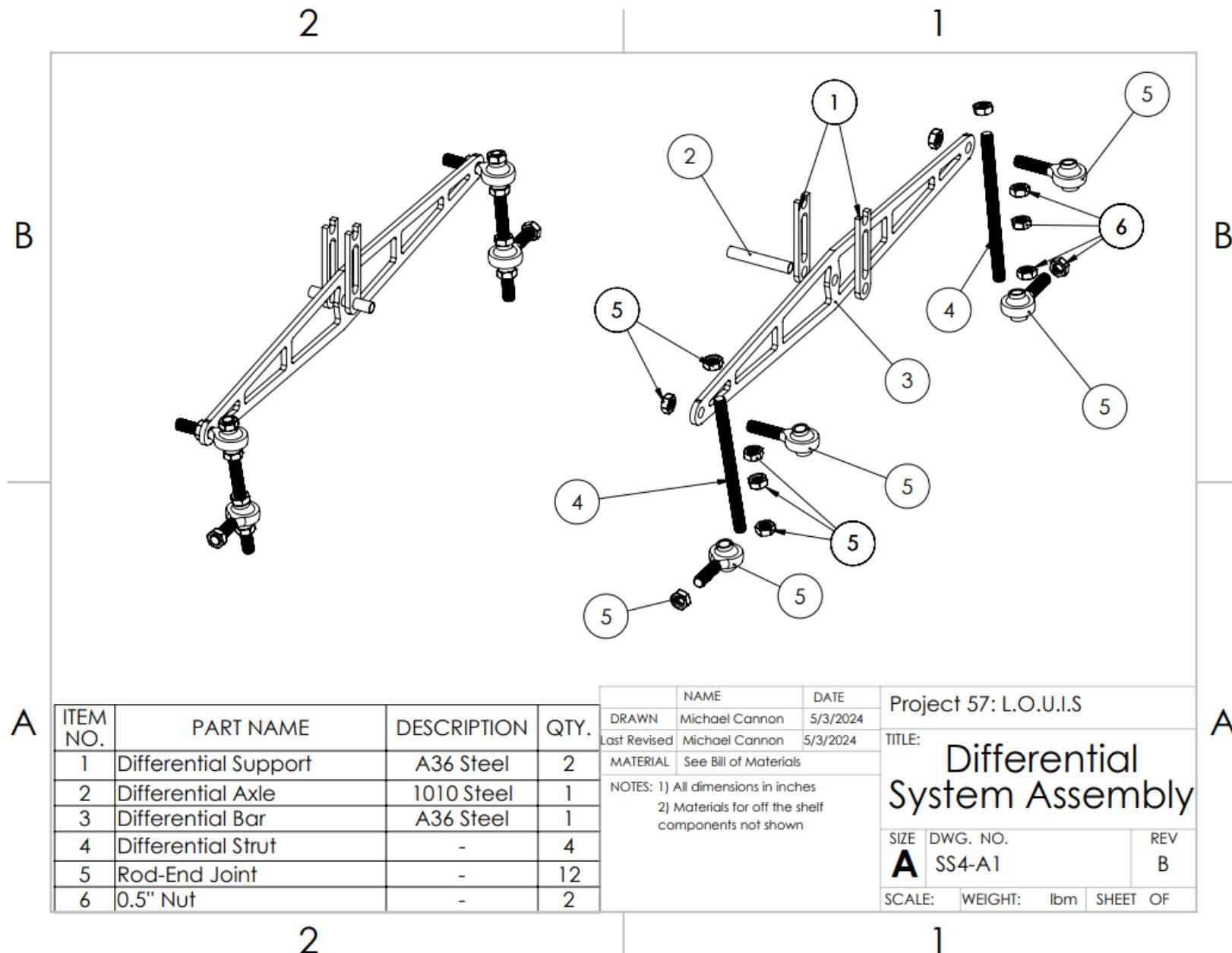


Figure XIII.C.4-1: Exploded View Assembly Drawing of Sub-System SS4 – Differential Suspension

XIII.C.4.a. *Comprehensive Parts List for SS4 – Differential Suspension*

Table XIII.C.4-4: List of Parts for Sub-System SS4

Part #	Quantity	Material	Name
SS4-P1	2	A36 Steel	Differential Support
SS4-P2	1	1010 Steel	Differential Axle
SS4-P3	1	A36 Steel	Differential Bar
SS4-P4	4	Stainless Steel	Differential Strut
SS4-P5	12	Steel	Rod-end Joint
SS4-P6	2	Steel	0.5" Nut



Fully Threaded Rod: 1/2"-13 Thread Size, Stainless Steel, 18-8, Plain, 12 in Overall Lg

Item 10W668 Mfr. Model U51070.050.1200

Product Details

Catalog Page 2044

Brand APPROVED VENDOR	Removable Yes
Color Silver	Resistance Properties High Temperature Resistant
End Type Straight Cut	RoHS Compliant Yes
Finish Plain	Rockwell Hardness B80 Min
Includes Nuts No	Standards ASTM A354BD
Material Stainless Steel	System of Measurement Inch
Material Fabrication Cold Forged	Tensile Strength 85,000 psi
Material Grade/Class 18-8	Thread Direction Right Hand
Maximum Temperature 302 °F	Thread Fit 2A
Minimum Temperature -58 °F	Thread Size 1/2"-13
Minimum Yield Strength 45,000 psi	Manufacturer Part Number U51070.050.1200
Overall Length 12 in	UNSPSC 31161618
REACH Compliant Yes	Country of Origin Varies (subject to change)

Product Description

These rods and studs provide high strength when mounting and securing components in assemblies or structures. They have greater corrosion resistance than steel rods and studs making them well-suited for use in wet environments.

Figure XIII.C.4-2: Specifications for part SS4-P4 – Differential Strut

Shank Thread Size	1/2"-20
ID	1/2"
ID Tolerance	-0.0005" to 0.0015"
Maximum Ball Swivel	65°
Overall Width	1 1/2"
Overall Thickness	1 1/4"
Shank Center Length	2 5/8"
Shank Thread Length	1 5/8"
Static Radial Load Capacity	16,200 lbs.
Material	Zinc-Plated Alloy Steel
Ball Material	Chrome-Plated Bearing Steel
Liner Material	PTFE Plastic
Lubrication	Not Required
Rod End Type	Ball Joint
Shank Threading	Fully Threaded
Shank Gender	Male
RoHS	RoHS 3 (2015/863/EU) Compliant
REACH	REACH (EC 1907/2006) (01/17/2023, 233 SVHC) Compliant
DFARS	Specialty Metals COTS-Exempt
Country of Origin	Peoples Republic of China
Schedule B	848330.8055
ECCN	EAR99

Figure XIII.C.4-5: Specifications for part SS4-P5 – Rod-End Joint

Material	Steel
Fastener Type	Hex
Thread Size	1/2"-20
Exterior Finish	Zinc
Finish Type	Sae

About this item

- Hexagonal shaped nut with internal thread
- Hex nuts are for general applications and are good when working at tight angles
- Zinc-plated for moderate corrosion resistance

The Hillman Group 3866 1/2-20 Grade 5 Hex Nuts SAE Zinc Plated (10-Pack)

Figure XIII.C.4-4: Specifications for part SS4-P6 – 0.5" Nut

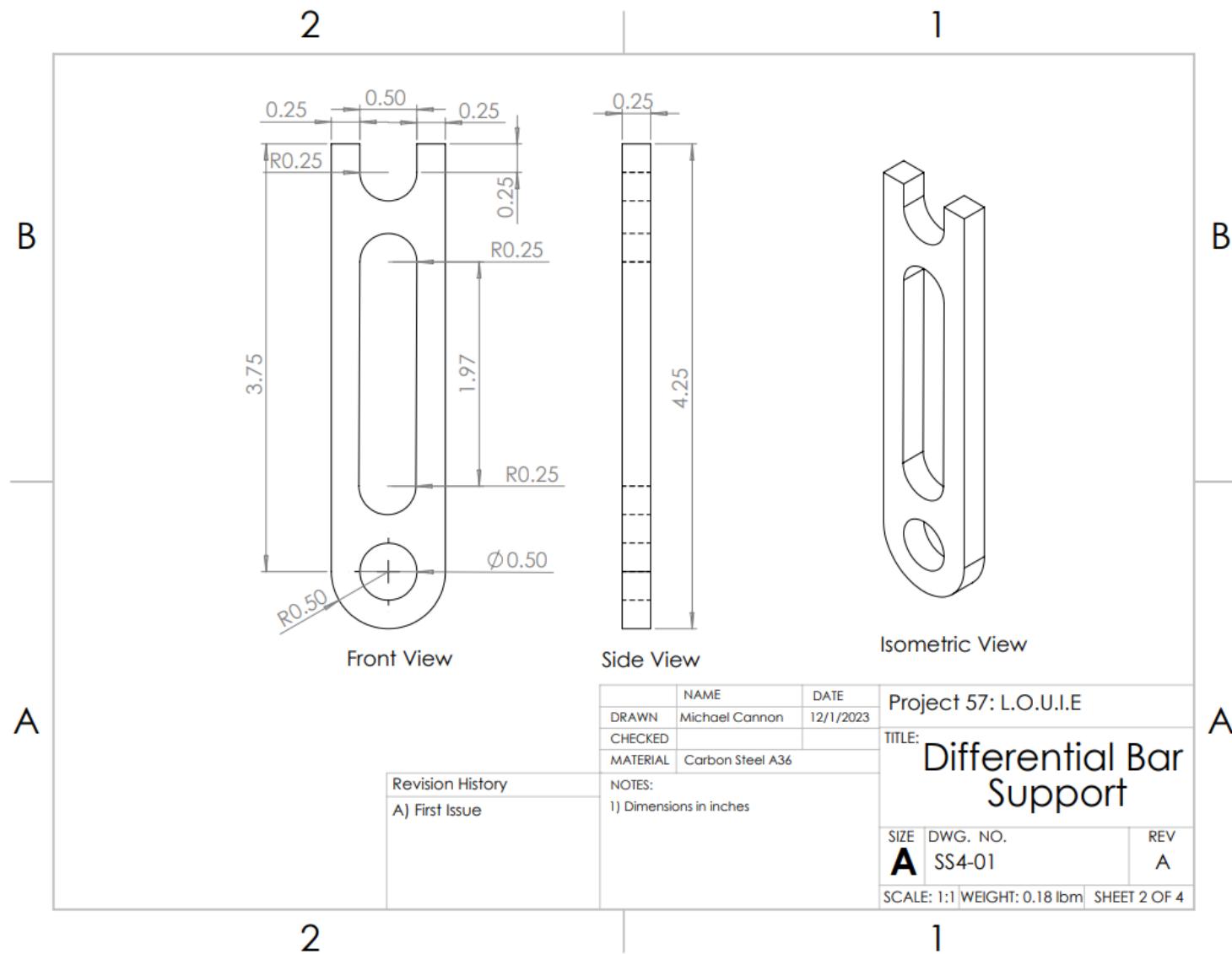


Figure XIII.C4-5: Manufacturing Drawing of part SS4-P1- Differential Support

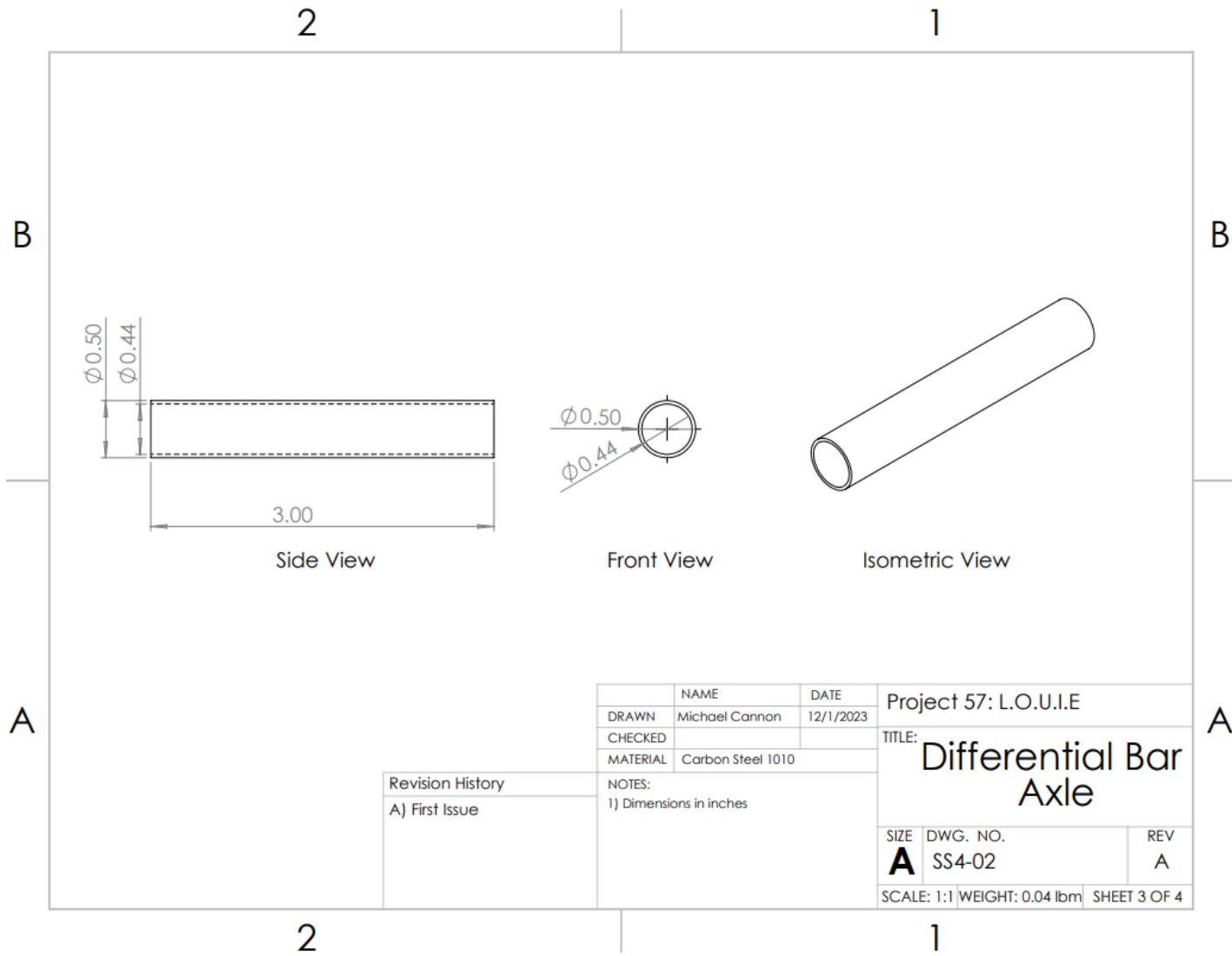


Figure XIII.C4-6: Manufacturing Drawing of part SS4-P2- Differential Axle

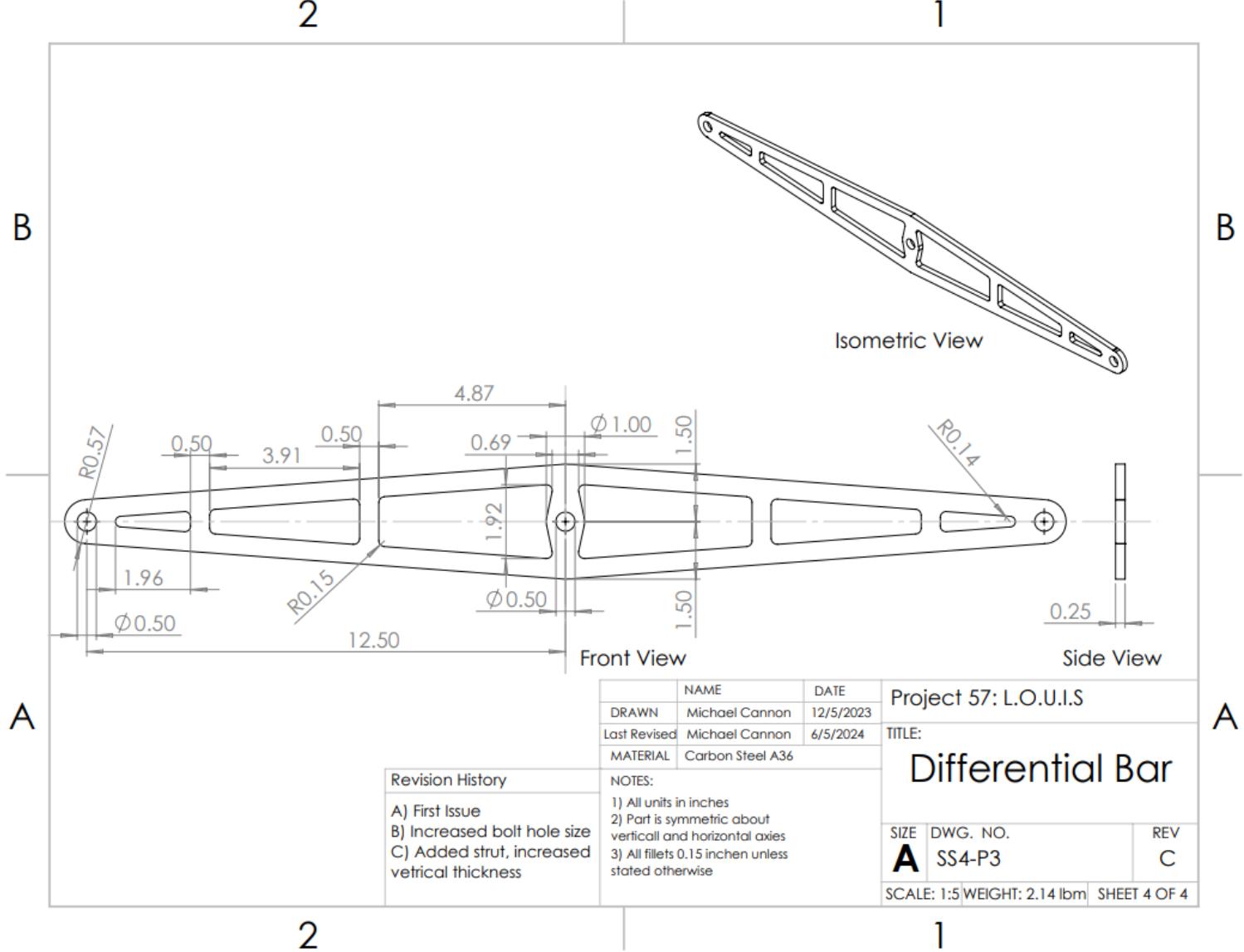


Figure XIII.C4-7: Manufacturing Drawing of part SS4-P3- Differential Bar

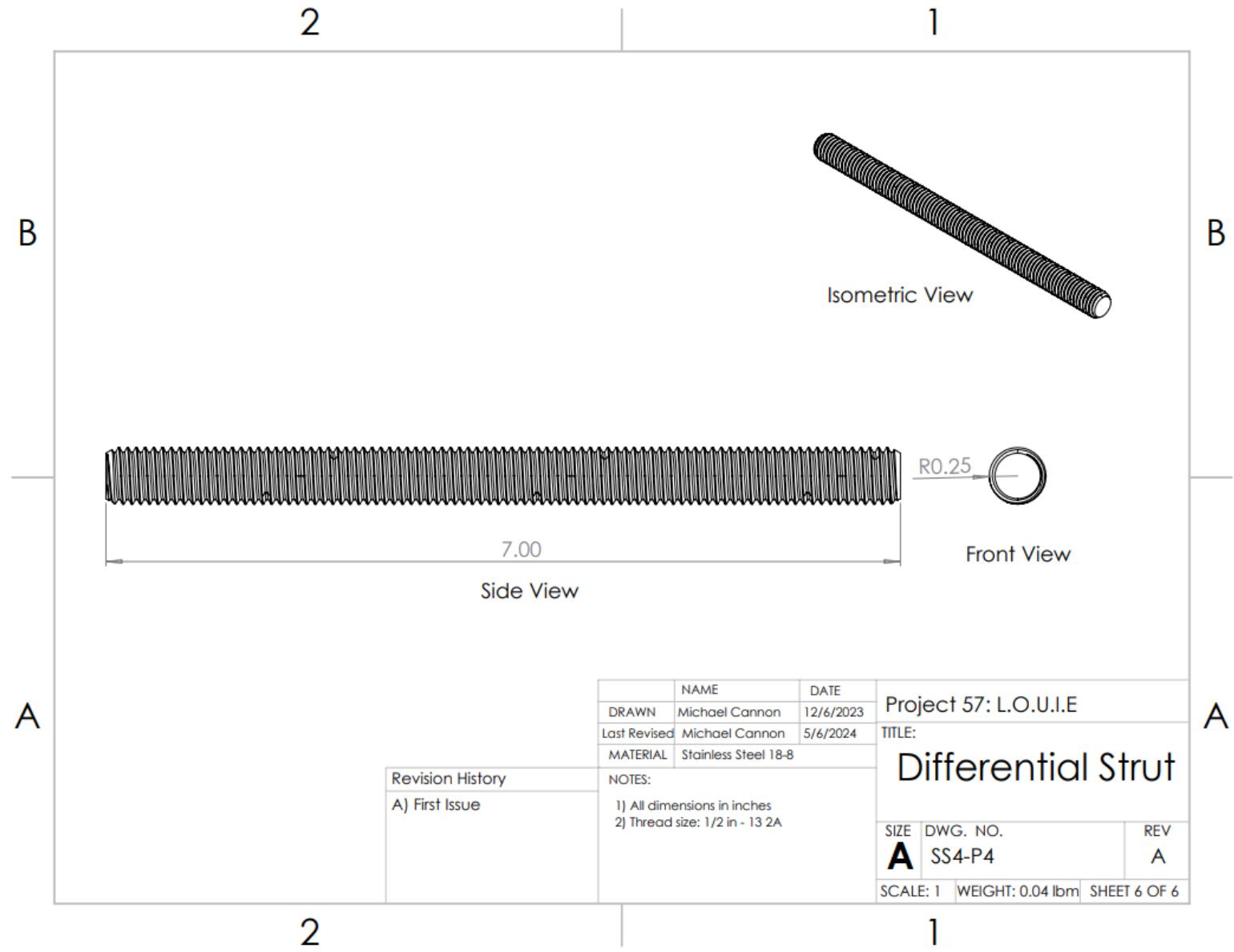
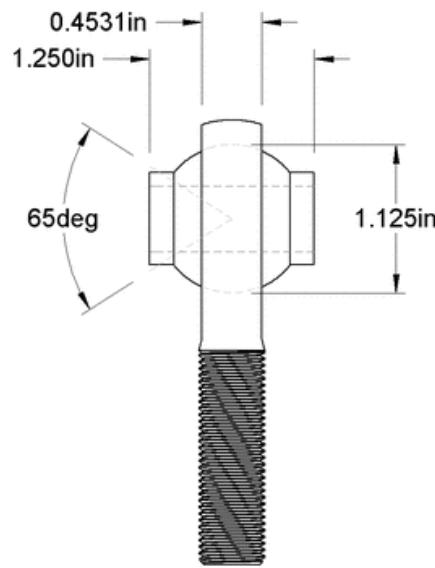
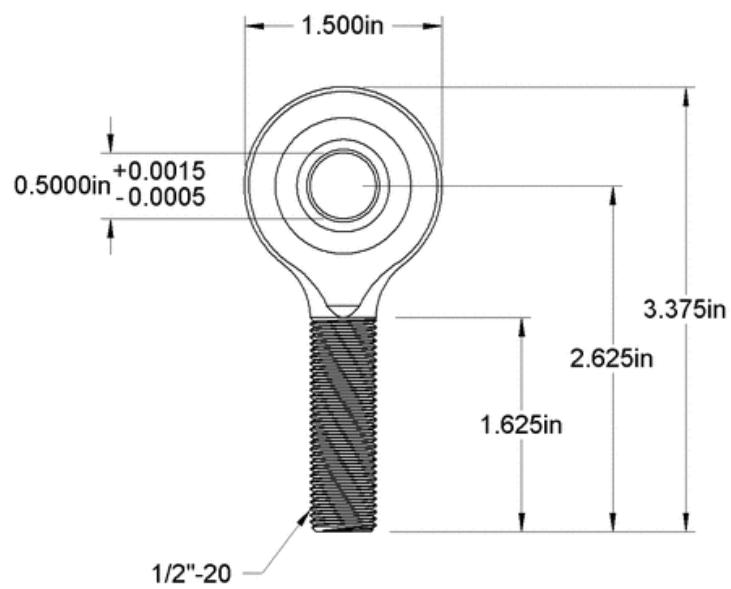


Figure XIII.C4-8: Manufacturing Drawing of part SS4-P4- Differential Strut



Shank Thread Direction: Right Hand

Figure XIII.C4-9: Manufacturing Drawing of part SS4-P5- Rod-End Joint



PART
NUMBER

6960T251

Super-Swivel Ball
Joint Rod End

XIII.C.5. Sub-System SS5 – Electric Propulsion

This subsystem consists of electric motors, motor controllers and gearboxes that will be used to drive the rover in the desired direction at the desired speed. We will have three motor controllers and 6 gear motors. Each controller will be responsible for two motors with independent control of each motor. The motor controller will take input voltage from the battery as well logic from one of the two onboard microcontrollers. These inputs will go through an algorithm and an output voltage will be produced that will be applied to the motor in order to turn it at the desired speed and in the desired direction. The motor will be coupled to the gearbox via a shaft and the gearbox will have an output shaft that will connect to the wheel in order to make the rover move.

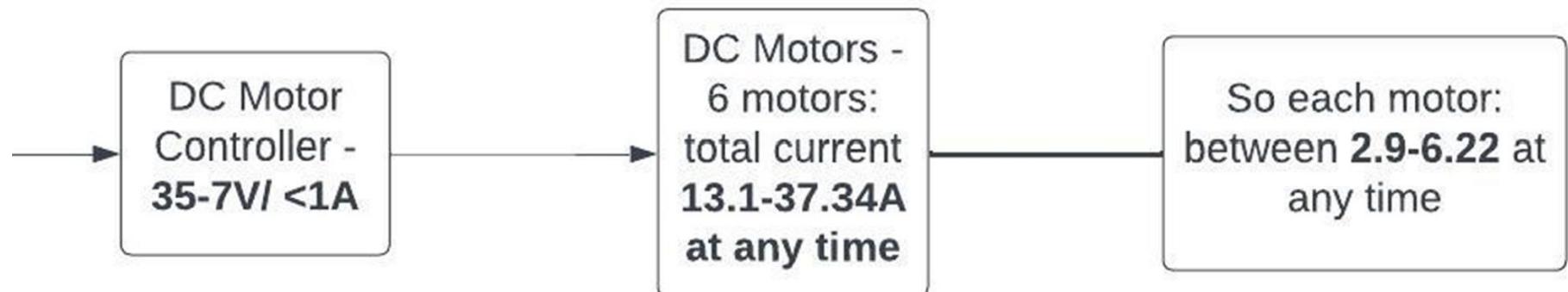


Figure XIII-6: Manufacturing Drawing/Wire Diagram SS5 – Electric Propulsion

XIII.C.5.a. Comprehensive Parts List for SS5 – Electric Propulsion

Table XIII-5: List of Parts for Sub-System SS5

Part #	Quantity	Material	Name
SS5-P1	6	N/A	Motor
SS5-P2	3	N/A	Motor Controller

XIII.C.5.b. Off-The-Shelf Parts and Component Specifications for SS5 – Electric Propulsion

	High Performance Economy Motors					
Voltage Range	6 Volts to 18 Volts		12 Volts to 36 Volts		24 Volts to 72 Volts	
Model	E30-400-12	E30-150-12	E30-400-24	E30-150-24	E30-400-48	E30-150-48
Diameter (inches)	3.1	3.1	3.1	3.1	3.1	3.1
Length (inches)	5.8	4.0	5.8	4.0	5.8	4.0
Peak HP	1.6	0.6	2.1	1.0	2.7	1.1
Stall Torque (oz-in)	970	420	1500	710	1860	750
Efficiency	78%	72%	79%	76%	82%	77%
Nominal Voltage	12V ¹	12V ¹	24V ¹	24V ¹	48V ¹	48V ¹
RPM at Nominal Voltage	6500	5400	5700	5600	5800	5600
Shaft Diameter (inches)	1/2	1/2	1/2	1/2	1/2	1/2
Shaft Length (inches)	2.0	2.0	2.0	2.0	2.0	2.0
Keyway (inches)	1/8	1/8	1/8	1/8	1/8	1/8
Capacitors	No	No	No	No	No	No
Magnet Type	Ferrite	Ferrite	Ferrite	Ferrite	Ferrite	Ferrite
No Load Amps	5.5	3.5	3.2	2.1	1.4	1.0
Resistance (Ohms)	0.030	0.081	0.089	0.190	0.284	0.708
Kt (oz-in/Amp)	2.46	2.93	5.63	5.70	11.1	11.4
Kv (RPM/Volt)	549	461	240	237	122	118
Weight (pounds)	5.9	3.6	5.9	3.6	5.9	3.6

Figure XIII-7: Specifications for part SS5-P1 - Motor

Voltage Range	6 Volts to 18 Volts		12 Volts to 36 Volts		24 Volts to 72 Volts	
Model	E30-400-12	E30-150-12	E30-400-24	E30-150-24	E30-400-48	E30-150-48
Peak Efficiency (PE)	78%	72%	79%	76%	82%	77%
RPM at PE	5800	4700	5100	5000	5300	5000
Torque at PE (oz-in)	105	57	145	85	150	80
Horsepower at PE	0.60 HP	0.26 HP	0.75 HP	0.40 HP	0.80 HP	0.40 HP
Current at PE	48 Amps	23 Amps	29 Amps	17 Amps	15 Amps	8 Amps

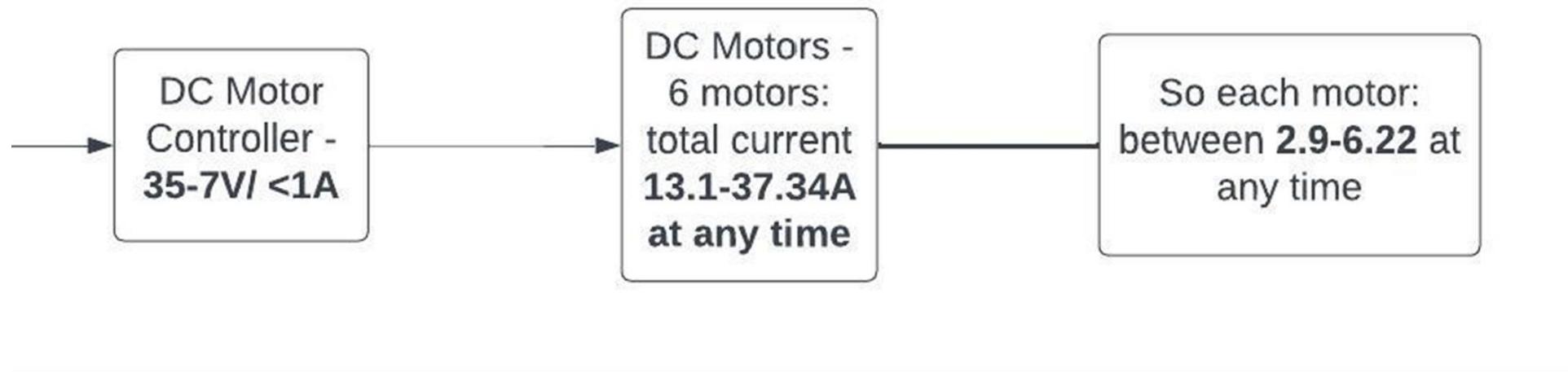
Figure XIII-8: Specifications for part SS5-P1 - Motor

- Operating voltage from 7V to 35VDC
- Rated Current <= 30A
- Peak Current <= 80A

Features:

- Bi-directional control for dual brushed DC motor
- Support motor voltage from **7V to 35VDC**
- Maximum current up to 80A peak (1 second), 30A continuous, each channel
- On board MOSFETs are switched at **18 KHz** for quiet operation
- Regenerative Braking
- On board Error LEDs to indicate: (each channel)
 - Input Error (Blinks 2 times)
 - Under Voltage Warning (Blinks 3 times)
 - Over Voltage Protection (Blinks 4 times)
 - Over Temperature Protection (Blinks 5 times)
- Thermal protection.
- Current limit protection
- Multiple input modes:
 - RC (Radio Control)
 - Analog voltage from potentiometer/variable resistor/joystick
 - PWM and DIR (sign-magnitude and locked-antiphase) from microcontroller
 - Serial Simplified or Serial Packetized from microcontroller
- GROVE compatible connectors for control input
- RC (Radio Control) friendly connectors
- On board push buttons for fast test and manual operation
- On board LED indicators for Error, RUN, Over current, motor output, for each channel
- **NO REVERSE POLARITY PROTECTION**
- Dimension: 81.28mm(W) x 101.60mm (L) x 42mm (H)

Figure XIII-9: Specifications for part SS5-P5 – Motor Controller



Manufacturing Drawing/Wire Diagram

XIII.C.6. Sub-System SS6 – Power Supply

For the power supply subsystem, it was determined that a lithium-ion battery would be used to provide power to all electrical components throughout the rover. We identified a 36V 50Ah lithium-ion battery from a company called Enduro as the best possible option for this subsystem. The main advantages of this battery are the fact that it is relatively light in weight (34.14 lbs), while also being small enough in dimension to fit in our rover without protruding from the frame. Even with the power supply chosen being small and light, it still has the nominal voltage and capacity ratings to adequately distribute power to all electrical components over the desired hour of operation. The datasheet for the selected battery can be found in Appendix.

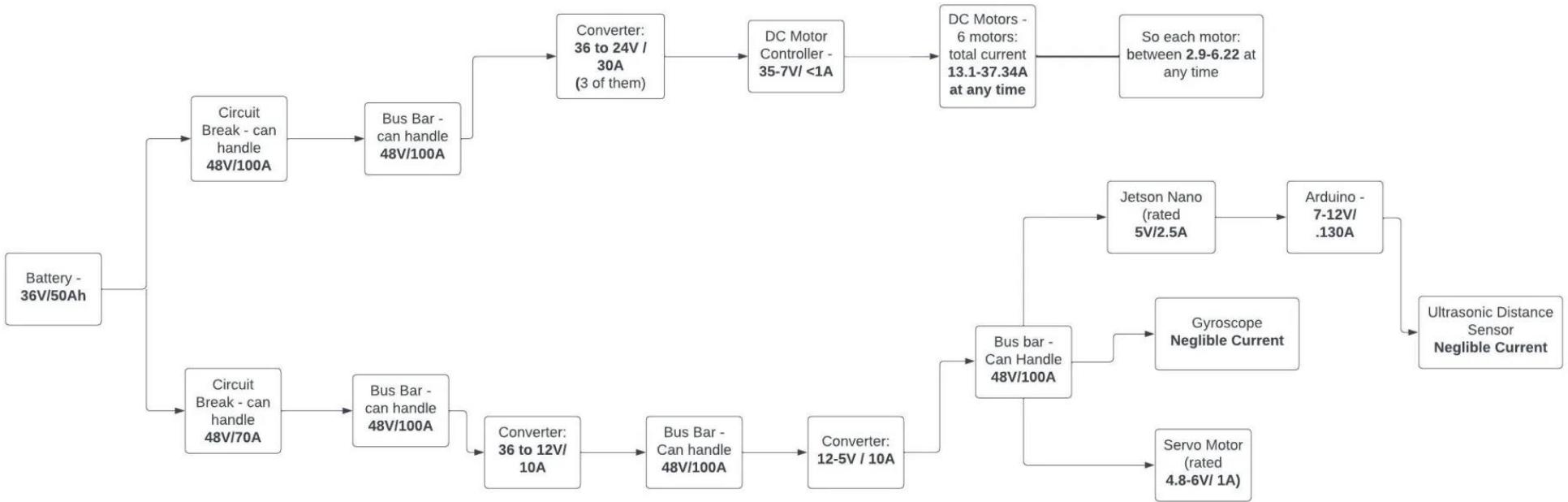


Figure XII-14: Exploded View Assembly Drawing of Sub-System SS6 – Power Supply

XIII.C.6.a. Comprehensive Parts List for SS6 – Power Supply

Table XIII-6: List of Parts for Sub-System SS6

Part #	Quantity	Material	Name
SS6-P1	1	N/A	36V 50Ah Lithium Ion Battery from Enduro
SS6-P2	1	N/A	36V 18A Lithium Ion Battery Charger from Enduro
SS6-P3	1	N/A	3 Pack 100 Amp Circuit Breakers 12V-48V
SS6-P4	1	N/A	1 70 Amp Circuit Breaker 12V-48V



BATTERY SPECIFICATION SHEET

Enduro Power ProConnect Series 36V 50Ah LiFePO4 Battery
EP3650BT



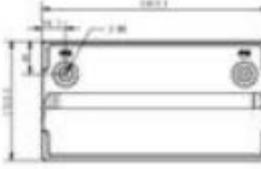
Product Specifications		BMS Overview	
Basic Electrical Specification			
Nominal Voltage	36.4 V	Over Current Protection	yes
Nominal Capacity	50 Ah	Under Voltage Protection	yes
Energy	1920 Wh	Over Voltage Protection	yes
Resistance	≤ 30mΩ @1kHz AC	Overload Protection	yes
Efficiency	99%	High Temperature Protection	yes
Self Discharge	≤3% per month	Low Temperature Protection	yes
Parallel Connection	4+ [call us if over 4]	Short Circuit Protection	yes
Series Connection	No Series Connections		
Charge Specification		Certifications	
Recommended Charge Current	10 A - 40 A	MSDS Datasheet	MSDS
Maximum Charge Current	60 A	UN38.3	DG61
Recommended Charge Voltage	44 V	CE Certification	EN61000-6-1, EN61000-6-3
BMS Charge Voltage Cut-Off	45 V (3.75 ±0.025 vpc)	ROHS Certification	ROHS
Reconnect Voltage	36.0 V	FCC Certification	Part15B
Balancing Voltage	3450mV (cell)	IEC62619 documentation or 49 CFR	IEC62619 / 49 CFR 173.185
		IMDG Code	UN3840 / Class 9
Discharge Specification		Mechanical Specifications	
Maximum Continuous Discharge Current	100A (up to 30 mins)	Dimensions (L x W x H)	13 x 6.8 x 9.43" (330 x 172 x 214 mm)
Peak Discharge Current	350 A (3s)	Weight	34.14 lbs (15.49 kg)
BMS Discharge Current Cut-Off	110A / (30s)	Terminal Type	M8
Recommended Low Voltage Disconnect	33.6 V	Terminal Torque	10 -12 N·m
BMS Discharge Voltage Cut-Off	30 V	Case Material	ABS
Reconnect Voltage	36 V	Enclosure Protection	IP54
Short Circuit Protection	1000 µs	Cell Type - Chemistry	Prismatic - LiFePO4
Temperature Specification			
Discharge Temperature	-4 to 130 °F (-20 to 55 °C)		
Charge Temperature	32 to 130 °F (0 to 55 °C)		
Storage Temperature - 1 month	-4 to 113 °F (-20 to 45 °C)		
Storage Temperature - 1 year	32 to 95 °F (0 to 35 °C)		
BMS High Temperature Cut-Off	176 °F (80 °C)		
Reconnect Temperature	122 °F (50 °C)		
BMS Low Temperature Cut-Off (Charging)	32 °F (0 °C)		
Reconnect Temperature (Charging)	41 °F (5 °C)		

Figure XII-15: Specifications for part SS6-P1 – Battery –: 36V 50 Ah

Features

- Advanced high frequency switching power supply technology.
- DC output isolated from AC input.
- Single Input 100-120VAC, 50Hz-60Hz.
- Charger compensates for AC input line voltage variations.
- Constant Current (CC), Constant Voltage (CV).
- 2 LEDs: LED1 Red (power on), LED2 Red/Green (charging/full).
- Short circuit, over voltage, over-temperature, reverse polarity protections.
- Forced ventilation with fans.

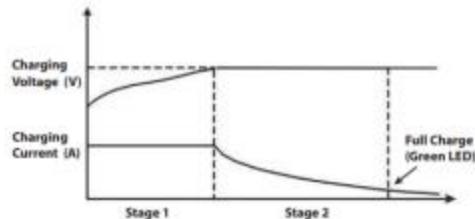
Operation

- Check to make sure charger is working properly.
 - Check if your local power supply (grid) conforms to the charger's input voltage 100-120V. Then connect the charger to a wall socket and turn on.
 - LED1 RED confirms AC power is on, and the LED2 GREEN confirms charger is operating properly.
- Unplug charger from wall socket prior to connecting DC output.
- Securely connect the DC Output cord Red (+) positive and then Black (-) negative to the battery terminals.
- Plug charger into wall socket and turn on.
- Charging Status: LED2 RED confirms the charger is charging, LED2 GREEN confirms the battery is fully charged. Once charging is complete disconnect the charger from your Enduro Power lithium battery within 24hrs.

Note: The charger has DC output once AC power is on. For Reverse Polarity Protection the charger has a replaceable Fuse.

Charging Mode (2-stage)

Stage 1: CC (Constant Current) Stage 2: CV (Constant Voltage)



Precautions

- Sparking is normal when making connection to battery due to high current.
- This charger is designed for charging LITHIUM LiFePO4 type batteries only.
 - Avoid charging incompatible batteries.
- The battery might be damaged if LED2 does not change to green after ample charge time.
- Always place the charger in well-ventilated and dry environments.
- The charger is designed with an aluminum case as its heat sink. Do not cover it to avoid the case/charger from overheating while charging.
- Pay close attention to DC terminal placement:
 - RED/Brown → (+) Positive Terminal, BLACK/Blue → (−) Negative Terminal**
- DO NOT disassemble charger. Call Enduro Power with technical concerns.

Packing List

- | | |
|------------------------------|------------------|
| 1. Charger | 3. AC Power Cord |
| 2. DC patch cable w/ M8 Lugs | 4. User Manual |

Figure XII-1716: Specifications for part SS6-P2 – 36V 18A Battery Charger

EPLC36V18: Enduro Power 36V 18A Lithium Battery Charger

✓ Charger Ratings

Input 110-130 Volts, 50/60Hz

Output: 43.8 Volts, 18 Amps.

Charging Mode: Constant Current(CC) and Constant Voltage (CV).

✓ Lithium Iron Phosphate (LiFePO4) Optimized

Optimized for charging all types of Lithium Iron Phosphate (LiFePO4) batteries.

Utilize only lithium compatible chargers with your LiFePO4 battery to obtain a 100% charge that will maintain the health of your lithium battery.

✓ Safe And Secure

Automatic shutdown when fully charged. Charger is programmed to guard against Overcharge, Over-Voltage, Overheat, Excessive Charging and Short Circuit protection.

✓ LED Charge Indicator

Red = Charging

Green = Fully Charged

Figure XII-1817: Specifications for part SS6-P2 36V 18A Battery Charger

XIII.C.7. Sub-System SS7 – Power Distribution

This subsystem consists of bus bars, DC-DC voltage converters, wires, and fuses that will be used to safely distribute power to all electronic components on the rover. There will be six total DC-DC voltage converters. The six converters are as follows: three 36 to 24V, one 36 to 12V, one 12 to 5V, and one 12 to 3V. We will use no more than 3 bus bars, whose purpose will be to serve as a central point for the connection of our electrical components. The bus bars will be used in combination with the different voltage converters to safely distribute power based on the ratings of the components that need to be powered. More specifically, the subsystem will utilize wires of AWG gauge 5, 9, 12, and 24 as conductors through the system. A flowchart of the power distribution subsystem can be seen in appendix.

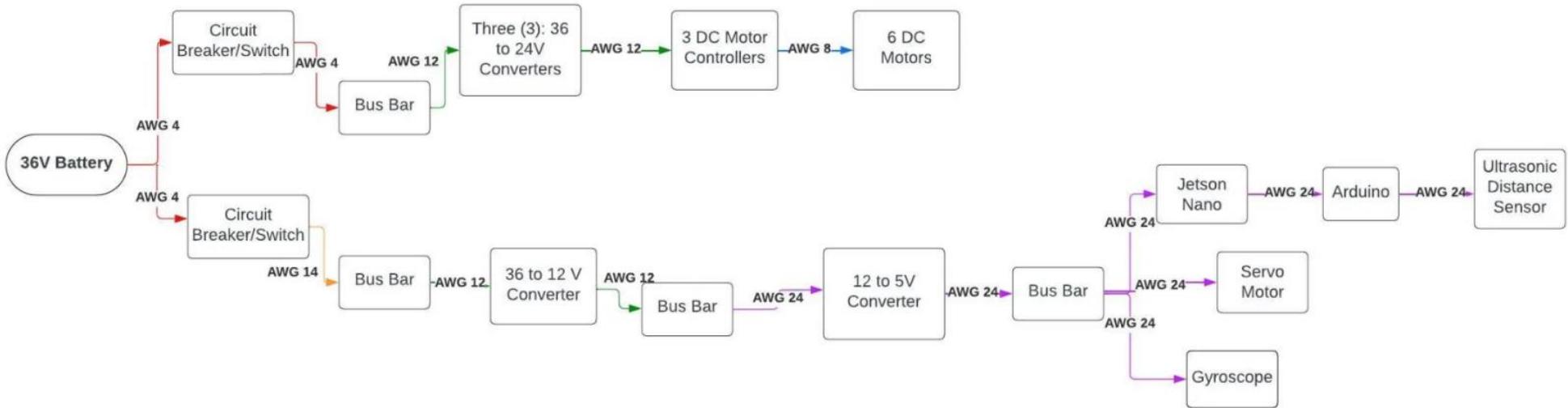


Figure XIII-10: Exploded View Assembly Drawing of Sub-System SS7 – Power Distribution

XIII.C.7.a. Comprehensive Parts List for SS7 – Power Distribution

Table XIII-7: List of Parts for Sub-System SS7

Part #	Quantity	Material	Name
SS7-P1	3	Tin-Plated Copper	Bus Bar
SS7-P2	3	N/A	DC 36/48V to 24V 30A Step Down Converter
SS7-P3	1	N/A	DC 36/48V to 12V 20A Step Down Converter
SS7-P4	1	N/A	DC 12V to 5V 10A Step Down Converter
SS7-P5	1	N/A	DC 12V to 3V 3A Step Down Converter
SS7-P6	2 feet	N/A	5 Gauge wire
SS7-P7	20 feet	N/A	9 Gauge wire
SS7-P7	20 feet	N/A	12 Gauge wire
SS7-P7	20 feet	N/A	24 Gauge wire

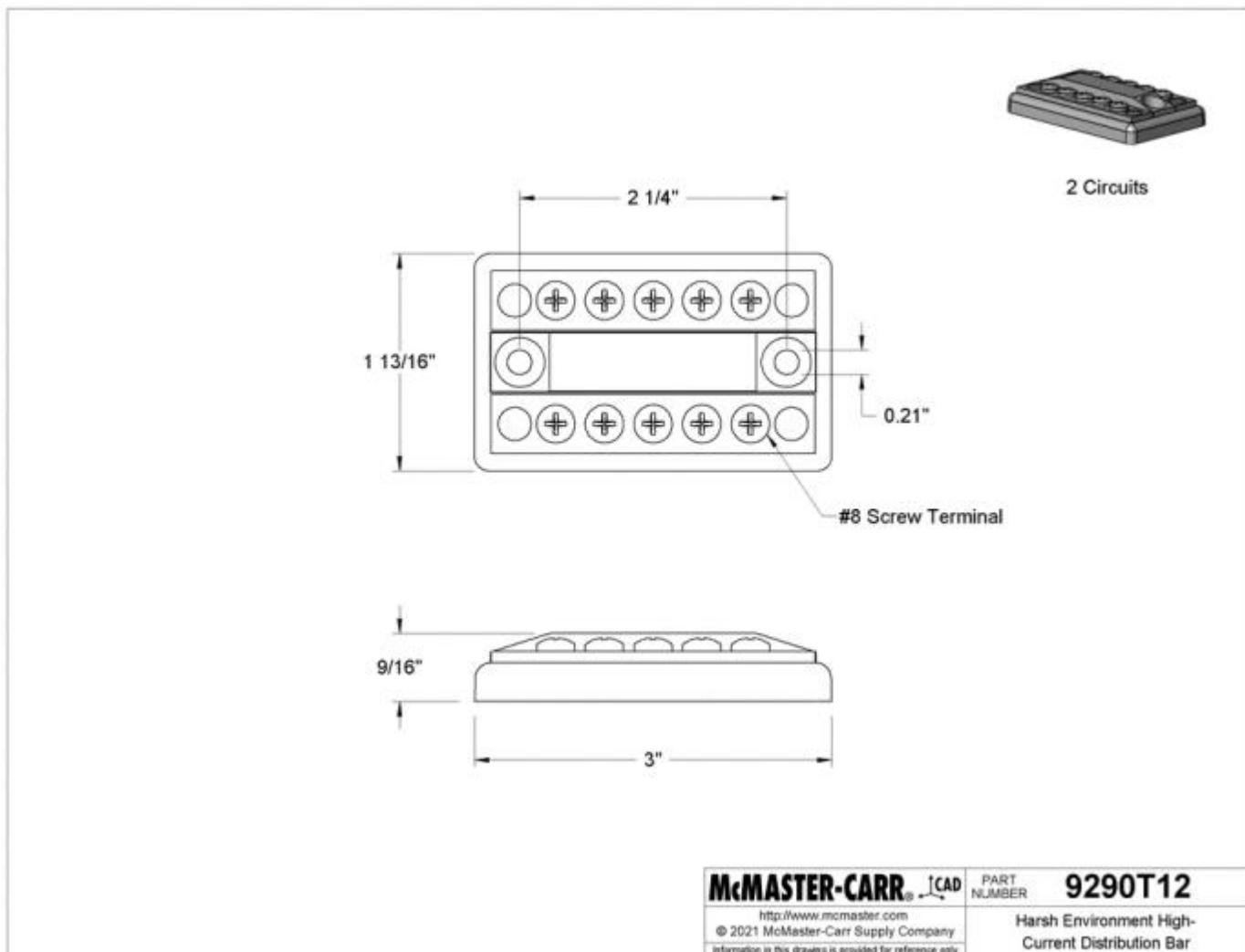


Figure XII-2019: Specifications for part SS7-P1 – Bus Bar

Number of Circuits	2
Current	
300V AC	100A
48V DC	100A
Wire Connection Type	Screw Terminals
Terminals	
Number Per Circuit	5
Size	No. 8
Length	3"
Width	1 13/16"
Height	9/16"
Material	Tin-Plated Copper
Terminal Material	Stainless Steel
Min. Temperature	Not Rated
Maximum Temperature	176° F
Features	Sealed Underside
Mounting Location	Surface
Mounting Fasteners Included	No
Mounting Holes	
Diameter	0.21"
Number of	2
Center-to-Center Length	2.25"
Specifications Met	CE Marked
RoHS	Not Compliant
REACH	Not Compliant
DFARS	Specialty Metals COTS-Exempt
Country of Origin	Mexico
USMCA Qualifying	No

Figure XII-21: Additional Specifications for part SS7-P1 – Bus Bar



Figure XII-2220: Specifications for part SS7-P1 – 36/48 to 24V 30A Converter

Wiring Diagram

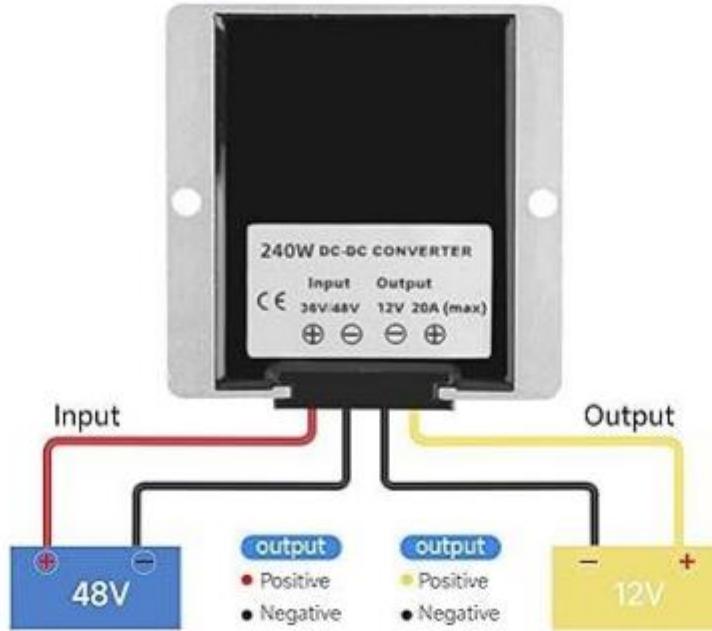


Figure XII-2321: Specifications for part SS7-P1 – 36/48 to 12V 20A Converter



Figure XII-2422: Specifications for part SS7-P1 – 12 to 5V 10A Converter



Figure XII-2523: Specifications for part SS7-P1 – 12 to 3V 3A Converter

XIII.C.8. Sub-System SS8 - Controller

The Digital Controller subsystem is a pivotal component of the rover, primarily responsible for its remote operation. This subsystem is built around a digital device, such as a tablet, which is equipped with a web application. The tablet establishes a connection to the rover's Wi-Fi access point, ensuring a secure and responsive communication link for controlling the rover. The web application on the tablet features an intuitive user interface, allowing operators to effortlessly send control signals for precise maneuvering and operation of the rover. These signals, processed by the controller, are transmitted wirelessly to the rover's onboard microcontrollers, enabling real-time control and feedback. Designed for compatibility with various digital devices, the subsystem offers flexibility in the choice of hardware for the control system. Additionally, it incorporates safety protocols and redundancy measures to maintain reliable operation under different circumstances.

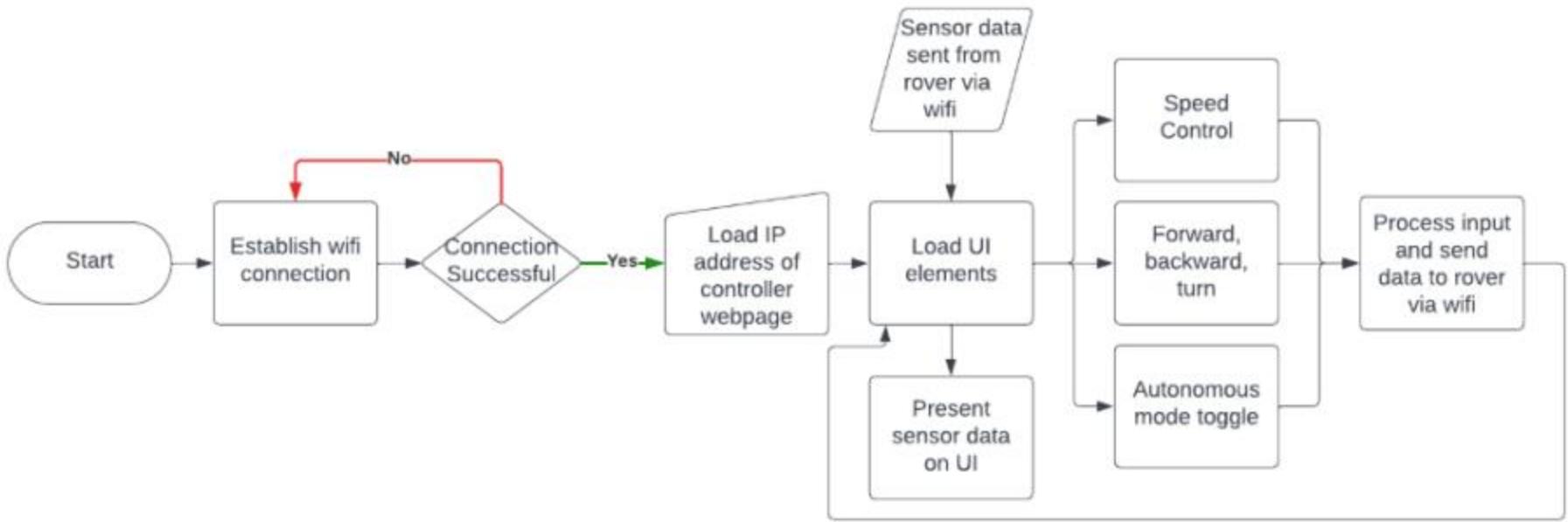


Figure XII-24: Manufacturing Drawing of part SS8-P1- Digital Controller

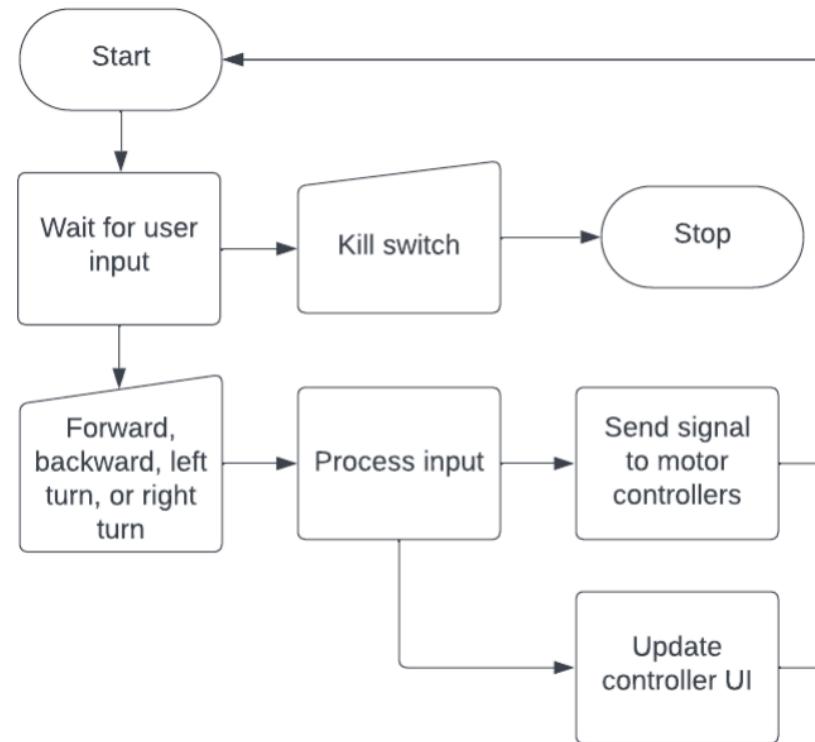


Figure XII-25: Manufacturing Drawing of part SS8-P2- Digital Kill Switch

XIII.C.8.a. Comprehensive Parts List for SS8 - Controller

Table XIII-8: List of Parts for Sub-System SS8

Part #	Quantity	Material	Name
SS8-P1	1	N/A	Digital Controller
SS8-P2	1	N/A	Digital Kill Switch

XIII.C.9. Sub-System SS9 – Object Detection and Reaction

XIII.C.9.a. Comprehensive Parts List for SS9 – Object Detection and Reaction

Table XIII-9: List of Parts for Sub-System SS9

Part #	Quantity	Material	Name
SS9-P1	1	N/A	Servo Motor
SS9-P2	2	N/A	Ultrasonic Sensors
SS9-P3	1	N/A	Object Detection Algorithms
SS9-P4	1	N/A	Jetson Nano Single Board Computer
SS9-P5	1	N/A	Arduino Due Microcontroller
SS9-P6	4	N/A	Temperature Sensors
SS9-P7	1	N/A	Gyroscope/Accelerometer
SS9-P8	1	N/A	Camera

- Electrical Specification (Function of the Performance): Operating speed (at no load): 0.09 ± 0.01 sec/ $60^\circ(4.8V)$ 0.08 ± 0.01 sec/ $60^\circ(6V)$
- Running current (at no load): 400 ± 30 mA ($4.8V$) 500 ± 30 mA($6V$)
- Stall torque (at locked): 2.0 ± 0.20 kg·cm ($4.8V$) 2.2 ± 0.20 kg·cm($6V$)
- Stall current (at locked): 1300 ± 40 mA ($4.8V$) 1600 ± 50 mA($6V$)
- Idle current (at stopped): 6 ± 1 mA ($4.8V$) 6 ± 1 mA($6V$)
- Running life(at no load): >350000 Turns($4.8V$) >320000 Turns($4.8V$)

Figure XII-26: Specifications for part SS9-P1 – Servo Motor

TECHNICAL DETAILS

- Voltage: 3V – 5.5V
- Current: 2.2mA
- Measuring Range: 2cm – 450cm
- Working Temperature: 0°C to 70°C
- Compatible with HC-SR04 Ultrasonic Sonar Distance Sensor

Product Weight: 6.0g / 0.2oz

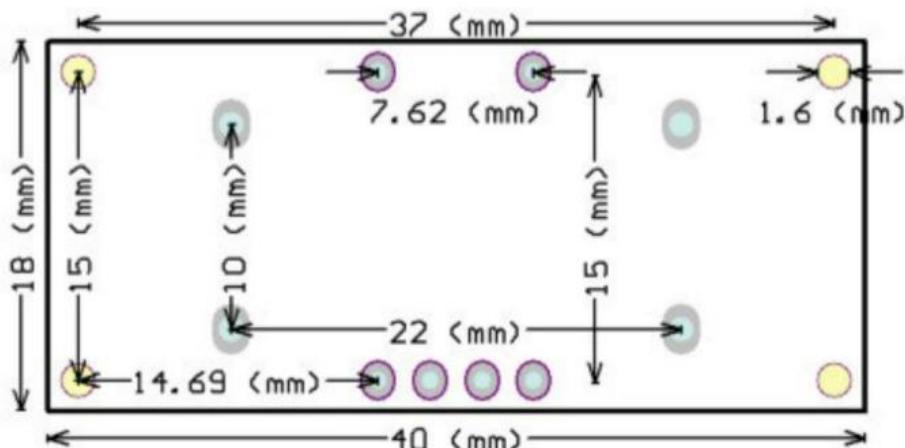


Figure XII-27: Specifications for part SS9-P2 – Ultrasonic Sensors

Maxwell GPU

128-core GPU | End-to-end lossless compression | Tile Caching | OpenGL® 4.6 | OpenGL ES 3.2 | Vulkan™ 1.1 | CUDA® | OpenGL ES Shader Performance (up to): 512 GFLOPS (FP16)
Maximum Operating Frequency: 921MHz

CPU

ARM® Cortex® -A57 MPCore (Quad-Core) Processor with NEON Technology | L1 Cache: 48KB L1 instruction cache (I-cache) per core; 32KB L1 data cache (D-cache) per core | L2 Unified Cache: 2MB | Maximum Operating Frequency: 1.43GHz

Audio

Industry standard High Definition Audio (HDA) controller provides a multichannel audio path to the HDMI interface.

Memory

Dual Channel | System MMU | Memory Type: 4ch x 16-bit LPDDR4 | Maximum Memory Bus Frequency: 1600MHz | Peak Bandwidth: 25.6 GB/s | Memory Capacity: 4GB

Storage

eMMC 5.1 Flash Storage | Bus Width: 8-bit | Maximum Bus Frequency: 200MHz (HS400) | Storage Capacity: 16GB

Boot Sources

eMMC and USB (recovery mode)

Networking

10/100/1000 BASE-T Ethernet | Media Access Controller (MAC)

Imaging

Dedicated RAW to YUV processing engines process up to 1400Mpix/s (up to 24MP sensor) | MIPI CSI 2.0 up to 1.5Gbps (per lane) | Support for x4 and x2 configurations (up to four active streams).

Operating Requirements

Temperature Range (T_j): -25 – 97C* | Module Power: 5 – 10W | Power Input: 5.0V

Display Controller

Two independent display controllers support DSI, HDMI, DP, eDP:
MIPI-DSI (1.5Gbps/lane): Single x2 lane | Maximum Resolution: 1920x960 at 60Hz (up to 24bpp)
HDMI 2.0a/b (up to 6Gbps) | DP 1.2a (HBR2 5.4 Gbps) | eDP 1.4 (HBR2 5.4Gbps) | Maximum Resolution (DP/eDP/HDMI): 3840 x 2160 at 60Hz (up to 24bpp)

Clocks

System clock: 38.4MHz | Sleep clock: 32.768kHz | Dynamic clock scaling and clock source selection

Multi-Stream HD Video and JPEG

Video Decode

H.265 (Main, Main 10): 2160p 60fps | 1080p 240fps
H.264 (BP/MP/HP/Stereo SEI half-res): 2160p 60fps | 1080p 240fps
H.264 (MVC Stereo per view): 2160p 30fps | 1080p 120fps
VP9 (Profile 0, 8-bit): 2160p 60fps | 1080p 240fps
VP8: 2160p 60fps | 1080p 240fps
VC-1 (Simple, Main, Advanced): 1080p 120fps | 1080i 240fps
MPEG-2 (Main): 2160p 60fps | 1080p 240fps | 1080i 240fps

Video Encode

H.265: 2160p 30fps | 1080p 120fps
H.264 (BP/MP/HP): 2160p 30fps | 1080p 120fps
H.264 (MVC Stereo per view): 1440p 30fps | 1080p 60fps
VP8: 2160p 30fps | 1080p 120fps

JPEG (Decode and Encode): 600 MP/s

Peripheral Interfaces

xHCI host controller with integrated PHY: 1 x USB 3.0, 3 x USB 2.0 | USB 3.0 device controller with integrated PHY | EHCI controller with embedded hub for USB 2.0 | 4-lane PCIe: one x1/2/4 controller | single SD/MMC controller (supporting SDIO 4.0, SD HOST 4.0) | 3 x UART | 2 x SPI | 4 x I2C | 2 x I2S: support I2S, RJM, LJM, PCM, TDM (multi-slot mode) | GPIOs

Mechanical

Module Size: 69.6 mm x 45 mm | PCB: 8L HDI | Connector: 260 pin SO-DIMM

Figure XII-28: Specifications for part SS9-P4 – Jetson Nano Single Board Computer

Board	Name	Arduino® Due
	SKU	A000062
Microcontroller	AT91SAM3X8E	
USB connector	Micro USB	
Pins	Built-in LED Pin	13
	Digital I/O Pins	54
	Analog input pins	12
	Analog output pins	2
	PWM pins	12
Communication	CAN	Yes (ext. transceiver needed)
	UART	Yes, 4
	I2C	Yes
	SPI	Yes
Power	I/O Voltage	3.3V
	Input voltage (nominal)	7-12V
	DC Current per I/O pin (group 1)	9 mA
	DC Current per I/O pin (group 2)	3 mA
	Power Supply Connector	Barrel Plug
	Total DC Output Current on all I/O lines	130 mA
Clock speed	Processor	AT91SAM3X8E 84 MHz
Memory	AT91SAM3X8E	96KB SRAM, 512KB flash
Dimensions	Weight	36 g
	Width	53.3 mm
	Length	101.5 mm

Figure XII-29: Specifications for part SS9-P5 – Arduino Due Microcontroller

Technical specs:

- Usable temperature range: -55 to 125°C (-67°F to +257°F)
- 9 to 12 bit selectable resolution
- Uses 1-Wire interface- requires only one digital pin for communication
- Unique 64 bit ID burned into chip
- Multiple sensors can share one pin
- $\pm 0.5^\circ\text{C}$ Accuracy from -10°C to +85°C
- Temperature-limit alarm system
- Query time is less than 750ms
- Usable with 3.0V to 5.5V power/data

Figure XII-30: Specifications for part SS9-P6 – Temperature Sensors

OPERATING CONDITIONS BNO055						
Parameter	Symbol	Condition	Min	Typ	Max	Unit
Start-Up time	T _{Sup}	From Off to configuration mode		400		ms
POR time	T _{POR}	From Reset to Config mode		650		ms
Data Rate	DR	s. Par. Fusion Output data rates				
Data rate tolerance 9DOF @100Hz output data rate (if internal oscillator is used)	DR _{tol}			±1		%
OPERATING CONDITIONS ACCELEROMETER						
Parameter	Symbol	Condition	Min	Typ	Max	Units
Acceleration Range	g _{FS2g}	Selectable via serial digital interface		±2		g
	g _{FS4g}			±4		g
	g _{FS8g}			±8		g
	g _{FS16g}			±16		g
OUTPUT SIGNAL ACCELEROMETER (ACCELEROMETER ONLY MODE)						
Parameter	Symbol	Condition	Min	Typ	Max	Units
Sensitivity	S	All g _{FSXg} Values, T _A =25°C		1		LSB/mg
Sensitivity tolerance	S _{tol}	T _A =25°C, g _{FS2g}		±1	±4	%
Sensitivity Temperature Drift	TCS	g _{FS2g} , Nominal V _{DD} supplies, Temp operating conditions		±0.03		%/K
Sensitivity Supply Volt. Drift	S _{VDD}	g _{FS2g} , T _A =25°C, V _{DD_min} ≤ V _{DD} ≤ V _{DD_max}		0.065	0.2	%/V
Zero-g Offset (x,y,z)	Off _{xyz}	g _{FS2g} , T _A =25°C, nominal V _{DD} supplies, over life-time	-150	±80	+150	mg
Zero-g Offset Temperature Drift	TCO	g _{FS2g} , Nominal V _{DD} supplies		±1	+/-3.5	mg/K
Zero-g Offset Supply Volt. Drift	Off _{VDD}	g _{FS2g} , T _A =25°C, V _{DD_min} ≤ V _{DD} ≤ V _{DD_max}		1.5	2.5	mg/V
Bandwidth	bw ₈	2 nd order filter, bandwidth programmable		8		Hz
	bw ₁₆			16		Hz
	bw ₃₁			31		Hz
	bw ₆₃			63		Hz
	bw ₁₂₅			125		Hz
	bw ₂₅₀			250		Hz
	bw ₅₀₀			500		Hz
	bw ₁₀₀₀			1,000		Hz

Figure XII-31: Specifications for part SS9-P7 – Gyroscope/Accelerometer

	Supports multiple resolutions, including 1080p (Full HD) @30 fps, and 720p (HD) @ 30fps to best support the quality offered by your application and monitor
Video	78° diagonal fixed field of view (dFOV) 1x digital zoom (Full HD) available Built-in HD autofocus ensures you're seen clearly throughout video calls RightLight 2 auto light correction for clear image in various lighting environments ranging from low light to direct sunlight
Audio	Dual omni-directional mics, optimized to capture audio clearly from up to one meter away, can be enabled via the Logi Tune app
Connectivity	Easily connects via USB-A; cable length of 5 ft (1.5 m)
Privacy Shutter	Switch to privacy mode in an instant with the attachable privacy shade
Mounting Options	Universal clip and 1/4" thread for tripod mounting ²
Logi Tune Support	Download Logi Tune at www.logitech.com/tune to activate mics plus control zoom, adjust color, set manual focus and easily update firmware
Certification	Certified for Microsoft Teams® and Zoom™
Compatibility	Works with other popular applications such as BlueJeans, Cisco Webex®, Fuze, Google Meet™, GoToMeeting™ and Microsoft DirectShow to ensure compatibility and seamless integration in the workplace
	Part number 960-001360
General	Including clip: Dimensions & weight Height x Width x Depth: 1.70 in (43.3 mm) x 3.70 in (94 mm) x 2.80 in (71 mm) Weight: 5.71 oz (162 g) Cable Length: 5 ft (1.5 m)
	What's in the box Webcam with attached 5 ft (1.5 m) USB-A cable Privacy shutter User documentation
	Warranty 3 years

Figure XII-32: Specifications for part SS9-P8 – Camera

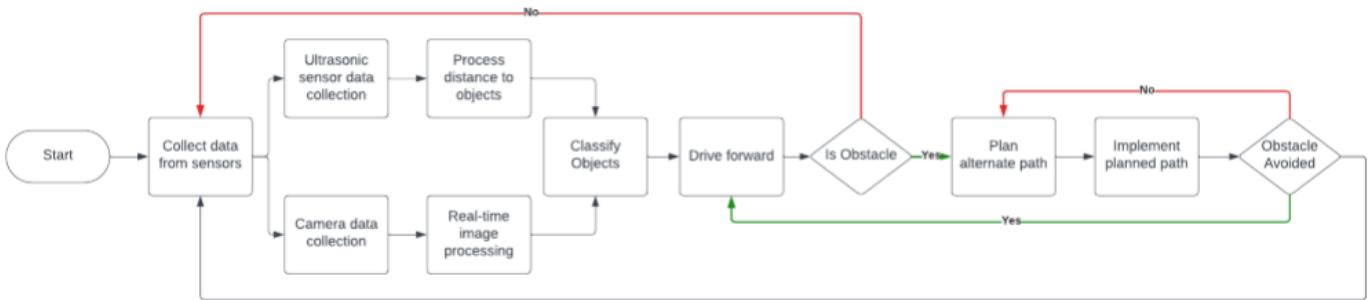


Figure XII-33: Manufacturing Drawing of part SS9P3- Algorithms

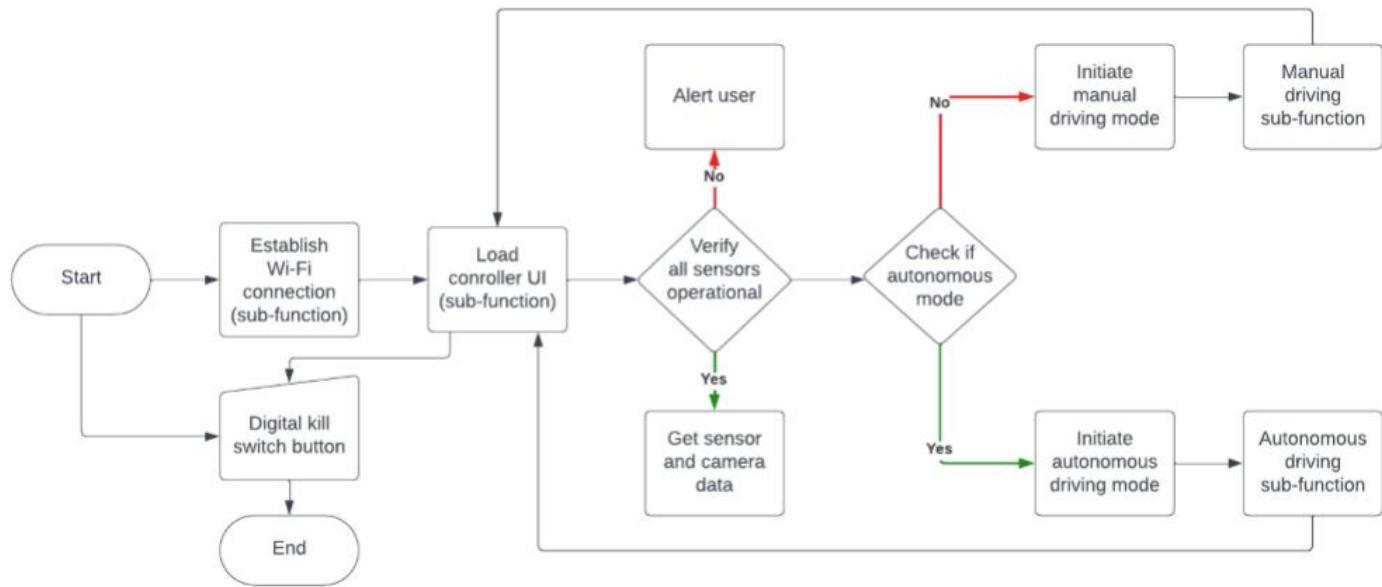


Figure XII-34: Manufacturing Drawing for part SS9-P4 – Jetson Nano

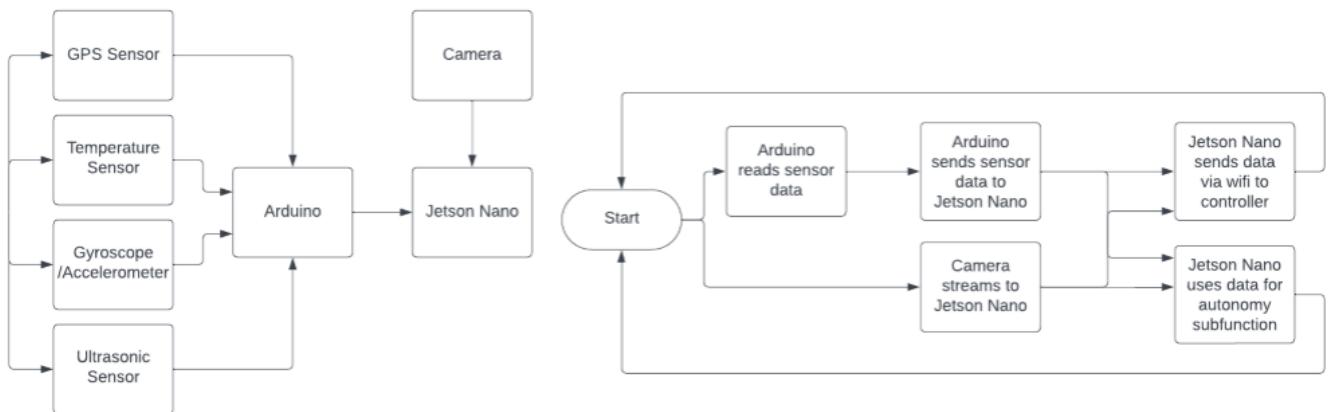


Figure XII-35: Manufacturing Drawing of part SS9P5 – Arduino Due

XIII.C.10. Sub-System SS10 - Communication

The Communication subsystem is a crucial component of our rover, designed to facilitate robust and reliable information exchange between the rover's various systems, its controller, and the integrated sensors. Central to this subsystem is the Wi-Fi module, which acts as the primary conduit for wireless communication. This module is responsible for establishing and maintaining a stable and secure Wi-Fi network, through which data and control commands are transmitted between the rover and its controller. This subsystem is engineered with a focus on implementing efficient and effective communication protocols. These protocols are to ensure seamless and uninterrupted data flow. They govern the transmission of critical information, such as telemetry data from the rover's sensors, operational status, and real-time environmental feedback, all of which are essential for informed decision-making and control. The Communication subsystem is integrated with the rover's onboard processing unit. This integration allows for the preprocessing and encoding of data before transmission, enhancing the efficiency of data exchange and reducing latency. The subsystem's design also encompasses error-checking and correction to ensure the integrity and reliability of the data transmitted, crucial in maintaining the operational accuracy of the rover.

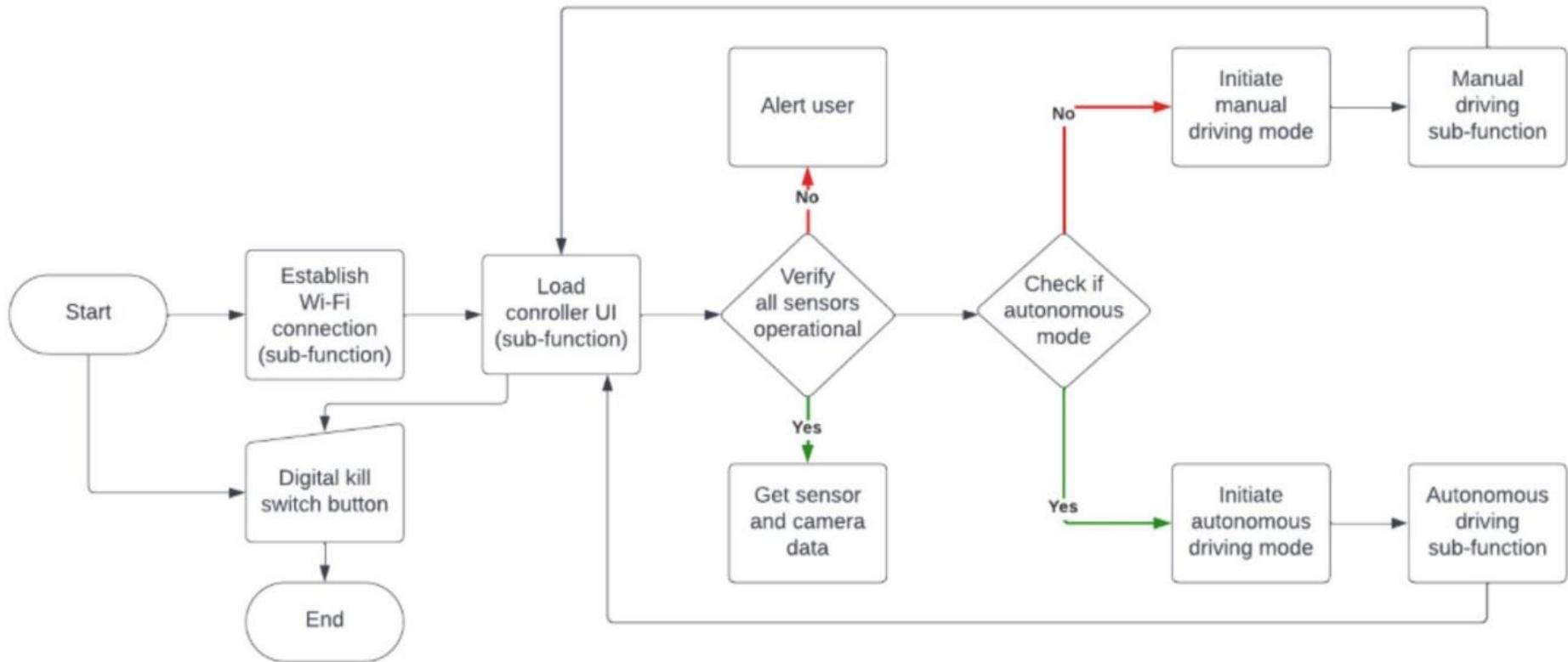


Figure XII-36: Exploded View Assembly Drawing of Sub-System SS10- Sub-Sys Controller

XIII.C.10.a. Comprehensive Parts List for SS10 - Communication

Table XIII-10: List of Parts for Sub-System SS10

Part #	Quantity	Material	Name
SS10-P1	1	N/A	Wi-fi Module
SS10-P2	1	N/A	GPS Breakout

- Supports 2.4G/5GHz dual-band, the speed up to 300Mbps in the 2.4GHz band and 867Mbps in the 5GHz band
- Supports Bluetooth 4.2 standard
- Adopts NGFF (M.2 A/E Key) interface
- Supports Linux, Windows 10/11, etc.

Figure XII-37: Specifications for part SS10-P1 – Wi-fi Module

- -165 dBm sensitivity, 10 Hz updates, 66 channels
- 5V friendly design and only 20mA current draw
- Breadboard friendly + two mounting holes
- RTC battery-compatible
- Built-in datalogging
- PPS output on fix
- Internal patch antenna + u.FL connector for external active antenna
- Fix status LED

Figure XII-38: Specifications for part SS10-P2 – GPS Breakout

XIII.D. Engineering Analysis and Materials Selection Supplement

The frame was recycled from a previous rover project, so 1010 carbon steel was already decided for us. The electronics platform were decided to be PLA plastic to benefit from the ease of manufacturing that comes from 3D printing.

XIII.D.1. Eng. Analysis Details for SS1 - [Frame]

XIII.D.1.a. Eng. Analysis and Materials Selection Details for SS1-P1 - [Frame]

Michael Cannon (ME)

Materials Selection

The frame material is 1010 carbon steel. This material has good machineability and weldability. It also has high strength making it a good choice for supporting loads and for use in the frame. However, the primary factors in this decision were based in frame design and cost rather than materials selection. The decision was made to use a component from an old project that was available in the machine shop. This saved time, cost, and manufacturing complexity. The details of that decision are expanded upon in Appendix XII.C.1. That locked the material at 1010 carbon steel. Since the old part was employed as a load bearing frame it was determined by a go-no-go decision that the material would be suited for this application.

Reaction force analysis

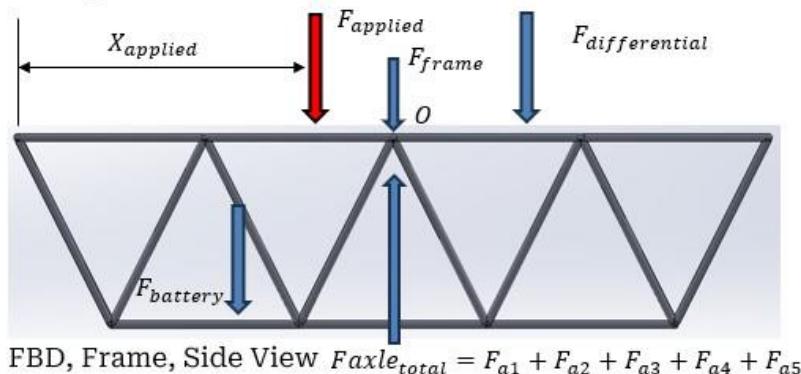
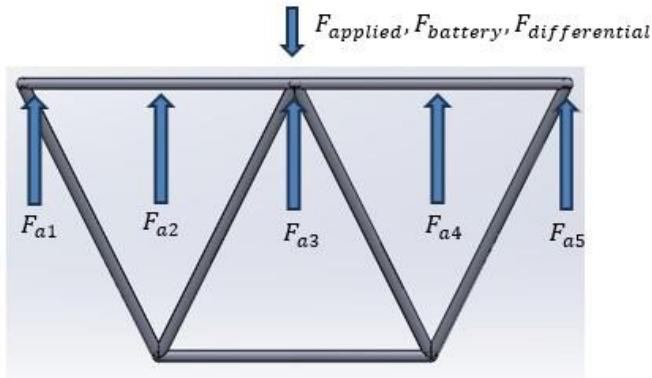


Figure XIII.D.1-1: FBD of frame. Side view.



FBD, Frame, Front View

Figure XII.D.1-2

Assumptions

- Entirety of applied load concentrated at one location
- Each horizontal strut modeled as a beam with a fixed support (for stress analysis)
- Supporting struts modeled as unsupported at the top (for buckling analysis)
- Rover on level ground
- Rover in static equilibrium

Reaction forces concentrated along centerline of frame.

$$\sum F_y = 0, \quad \sum M_O = 0$$

$$F_d = F_{app}(X_a - X_{app}) + \frac{F_b(X_a - X_b)}{X_d - X_a}$$

$$F_a = F_f + F_d + F_b + F_{app}$$

In MATLAB, Xapp ranged from 0 to 48 in (1.22 m)

Results

For $M_{app}=35$ Kg, $F_a_{max}=1695$ (381 lbf) N, $F_d_{max}=1114$ N (250 lbf)

These forces were taken as the worst case scenario encountered when operating under design conditions.

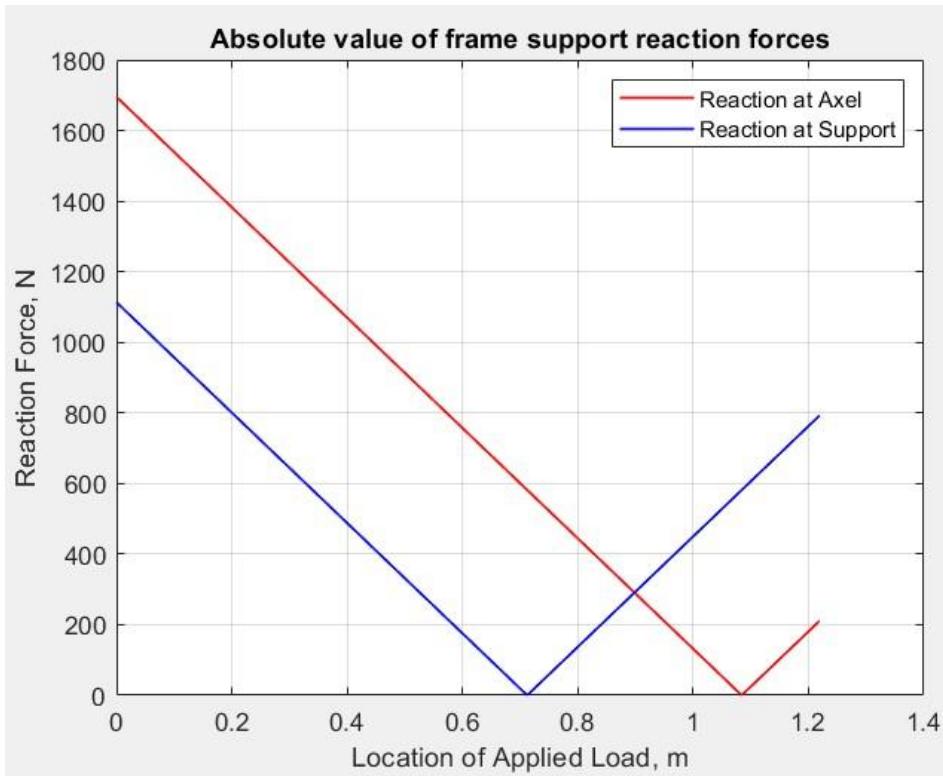


Figure XII.D.1-3: Absolute value of frame reaction forces.

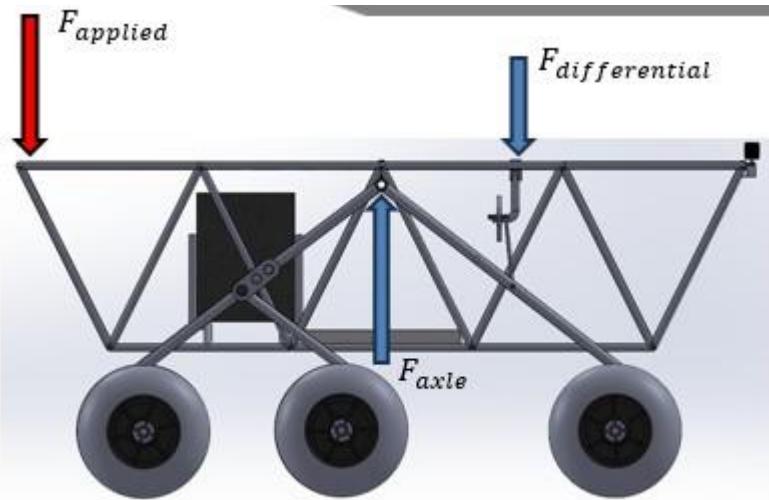


Figure XII.D.1-4: Location of applied load to generate greatest reaction forces.

Strut FBD

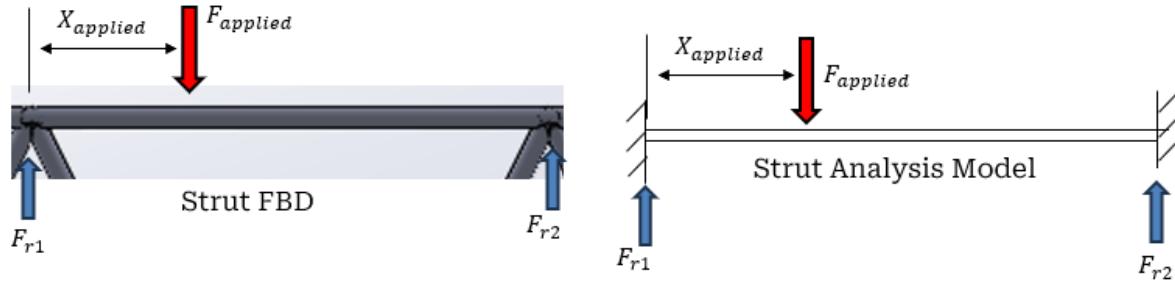
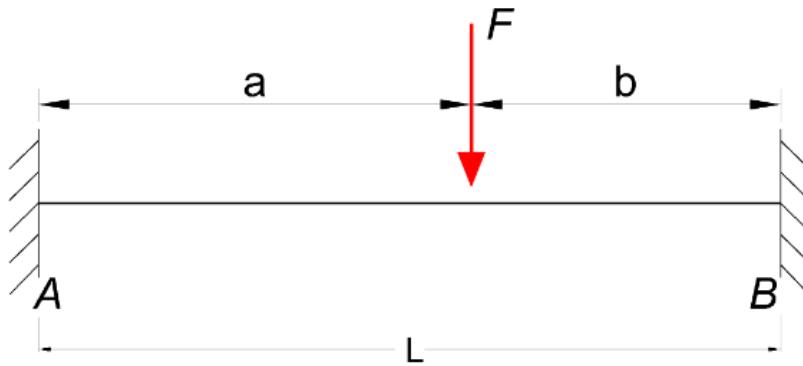


Figure XII.D.1-5: FBD of frame strut (left), FBD of model used for analysis (right)

For the stress analysis, each strut was modeled as a beam with fixed supports and the entirety of the applied load or reaction force located at the point to generate the highest moment. This point comes out to be 1/3 along the length of the beam. This will be explained in the Appendix XII.D.1.b.



XII.D.1-6 Terminology used in analysis.

For from the applied load of 75 lbf (334 N) and taking $a= 4$ in (0.1016 m), $b=8$ in (0.2032 m), and $L=12$ in (0.3048 m):

$$M_{bending} = \frac{Fab^2}{L^2} = 334 * 0.1016 * \frac{0.2032^2}{0.3048^2} = 0.7528 N * m$$

$$I = \frac{\pi(r_2^4 - r_1^4)}{4} = \frac{\pi(0.0063^4 - 0.0055^4)}{4} = 5.45 * 10^{-10} m^4$$

$$y = r_2 = 0.0063 m$$

$$\sigma_b = \frac{M_b y}{I} = 0.7528 * \frac{0.0063}{5.45 * 10^{-10}} = 8.76 MPa$$

$$SF = \frac{\sigma_y}{\sigma_b} = \frac{305}{8.76} = 34.8$$

For deflection:

$$\delta = \frac{Fa^3b^3}{3L^3EI} = 334 * 0.1016^3 * \frac{0.2032^3}{3 * 0.3048^3 * 305 * 10^9 * 5.45 * 10^{-10}} = 1.36 * 10^{-5} m (0.00053 in)$$

For from the reaction force from the differential of 250 lbf (1114 N) and taking $a= 4$ in (0.1016 m), $b=8$ in (0.2032 m), and $L=12$ in (0.3048 m):

$$M_{bending} = \frac{Fab^2}{L^2} = 1114 * 0.1016 * \frac{0.2032^2}{0.3048^2} = 2.51 N * m$$

$$I = \frac{\pi(r_2^4 - r_1^4)}{4} = \frac{\pi(0.0063^4 - 0.0055^4)}{4} = 5.45 * 10^{-10} m^4$$

$$y = r_2 = 0.0063 m$$

$$\sigma_b = \frac{M_b y}{I} = 2.51 * \frac{0.0063}{5.45 * 10^{-10}} = 29.2 MPa$$

$$SF = \frac{\sigma_y}{\sigma_b} = \frac{305}{29.2} = 10.4$$

For deflection:

$$\delta = \frac{Fa^3 b^3}{3L^3 EI} = 1114 * 0.1016^3 * \frac{0.2032^3}{3 * 0.3048^3 * 305 * 10^9 * 5.45 * 10^{-10}} = 4.53 * 10^{-5} m (0.0018 in)$$

For from the reaction force from the axle of 381 lbf (1695 N) and taking a= 4 in (0.1016 m), b=8 in (0.2032 m), and L=12 in (0.3048 m):

$$M_{bending} = \frac{Fab^2}{L^2} = 1695 * 0.1016 * \frac{0.2032^2}{0.3048^2} = 3.82 N * m$$

$$I = \frac{\pi(r_2^4 - r_1^4)}{4} = \frac{\pi(0.0063^4 - 0.0055^4)}{4} = 5.45 * 10^{-10} m^4$$

$$y = r_2 = 0.0063 m$$

$$\sigma_b = \frac{M_b y}{I} = 3.82 * \frac{0.0063}{5.45 * 10^{-10}} = 44.5 MPa$$

$$SF = \frac{\sigma_y}{\sigma_b} = \frac{305}{44.5} = 6.85$$

For deflection:

$$\delta = \frac{Fa^3 b^3}{3L^3 EI} = 1695 * 0.1016^3 * \frac{0.2032^3}{3 * 0.3048^3 * 305 * 10^9 * 5.45 * 10^{-10}} = 6.9 * 10^{-5} m (0.0027 in)$$

For buckling:

The supporting struts were analyzed for buckling failure under compressive loads. For this analysis, the strut was modeled without the horizontal strut supporting it at the top. This will result in very conservative numbers since much of the load would be absorbed by the horizontal struts through bending. If the component passes analysis under these assumptions it will pass under actual conditions. This also assumes that all of the applied load is concentrated over just one strut. If it passes under this assumption, it can be assumed to pass when positioned over two, three, or four struts.

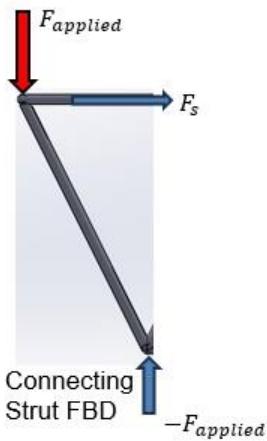


Figure XII.D.1-7: FBD of frame support strut.

Case	1	2	3	4	5
Constraints					
<i>k</i>	4	1	.25	2.046	1

Figure XII.D.1-8: Worst case scenario method of buckling, unsupported, chosen for analysis

The following equations were used to determine the critical buckling force. Length was found in SolidWorks.

$$I = \frac{\pi(r_2^4 - r_1^4)}{4} = \frac{\pi(0.0063^4 - 0.0055^4)}{4} = 5.45 * 10^{-10} m^4$$

$$F_{cr} = \frac{\pi^2 EI}{KL^2} = \frac{\pi^2 * 305 * 10^9 * 5.45 * 10^{-10}}{0.25 * 0.3696^2} = 3005 N (675.6 lbf)$$

$$SF = \frac{F_{cr}}{F} = \frac{3005}{334} = 9.0$$

XIII.D.1.b.

Eng. Analysis and Materials Selection Details for SS1-P1 – Center of Mass

[Tristan Hughes (ME)]

Center of Mass:

Assumptions: 9 heaviest objects will be a close approximation.

Consider frame, battery, drivetrain, and electronics

Take front of rover as 0 on X axis and ground as center on Z axis

Given:

$$m_{frame} = 30 \text{ lbs}, m_{battery} = 30 \text{ lbs}, m_{drivetrain} = 15 \text{ lbs}, m_{electronics} = 10 \text{ lbs}$$

$$x_{frame} = 24 \text{ in}, x_{battery} = 30 \text{ in}, x_{drivetrain} = 5, 18, 30 \text{ in}, x_{electronics} = 12 \text{ in}$$

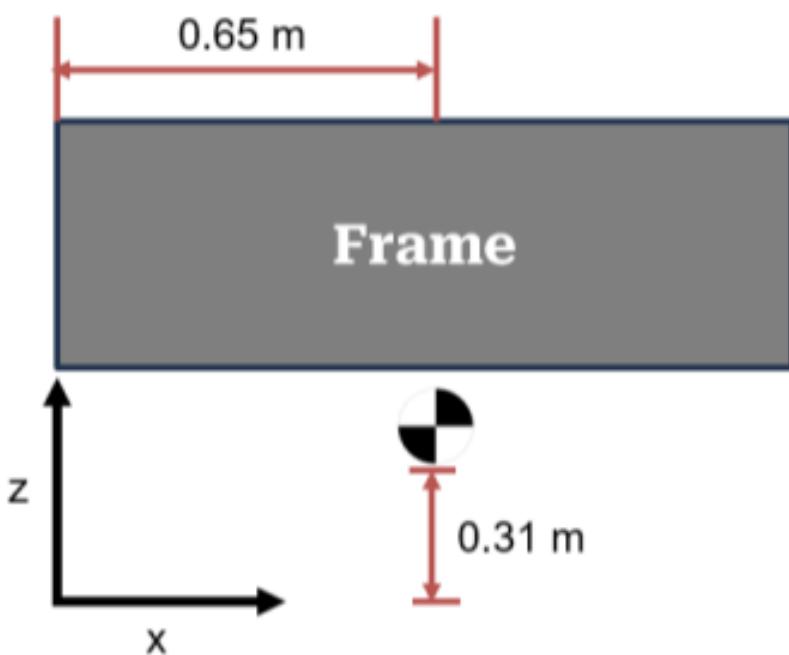
$$z_{frame} = 18 \text{ in}, z_{battery} = 16 \text{ in}, z_{drivetrain} = 6 \text{ in}, z_{electronics} = 12 \text{ in}$$

$$x_{cm} = \frac{\sum m_i x_i}{m_{total}}, z_{cm} = \frac{\sum m_i z_i}{m_{total}}$$

Converting to metric and plugging in yields

$$x_{cm} = 0.65 \text{ m}, z_{cm} = 0.31 \text{ m}$$

Side View



Materials Selection

Since this part needs to be welded to the frame of the rover, which is steel, this component also needs to be steel. The selected material was A513 carbon steel. This was the only material available by any vendor ate a reasonable price what also fit our geometrical requirements.

Assumptions

1. Force applied to the axle is the largest axle reaction force found in Appendix XII.D.1.a.
2. Force is applied at a single location on the axle at a point that will generate the greatest bending moment. (Another worst-case assumption. The axle is actually supported at 5 separate locations.)
3. Axle is modeled as a beam with fixed supports.
4. The assumptions above represent the highest stress encountered by the beam when operating under design conditions. Therefore, if the axle passes this analysis, it will also pass under all other loading conditions when operating under design loads.
5. L=24 in (0.6096 m), E=200 GPa (31910 ksi)

Analysis

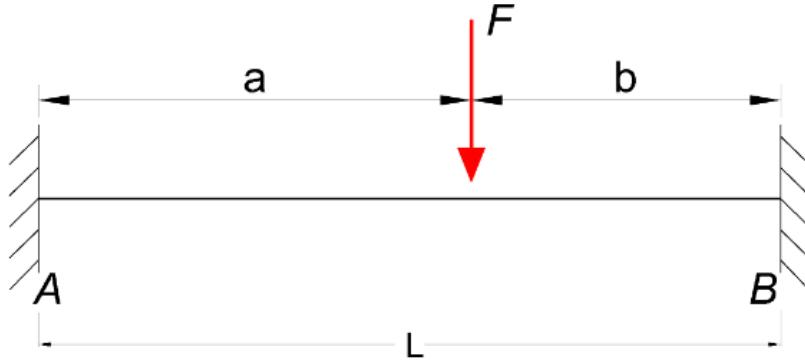


Figure XII.D.1-9: Terminology used in analysis.

$$M_{bending} = \frac{Fab^2}{L^2}$$

To determine the force location that would generate the greatest moment, the equation was inserted into MATLAB and solved for the full range of force location.

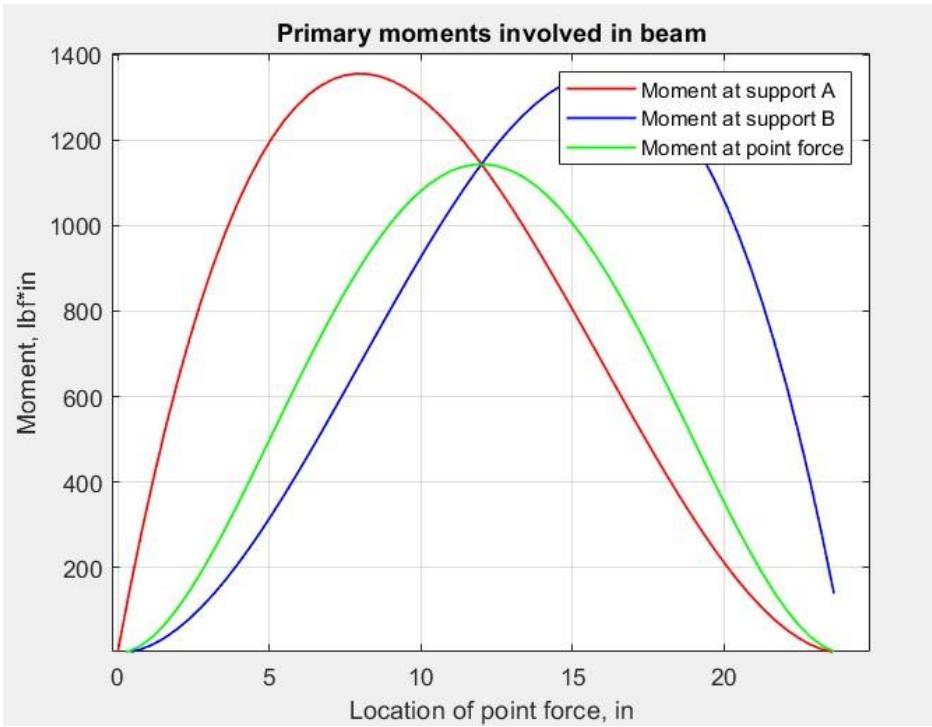


Figure XII.D.1-10: Peak moments experienced by the beam for a range of applied force locations. (MATLAB)

As seen in Figure XII.D.1-10, the bending moment is highest when the force is located at 1/3 and 2/3 of the total beam length. The location of 1/3 the length of the beam is therefore chosen as the location of the axle force, and the greatest moment will be at the reaction force at A.

For maximum bending moment:

$$a = \frac{L}{3} = \frac{0.6096}{3} = 0.2032 \text{ m}, \quad b = \frac{2L}{3} = 2 * \frac{0.6096}{3} = 0.4064 \text{ m}$$

$$M_{b,max} = \frac{Fab^2}{L^2} = 1695 * 0.2032 * \frac{0.4064^2}{0.6096^2} = 153 \text{ N m (1345 lbf in)}$$

Bending stress in the axle is therefore found as:

$$\sigma_b = \frac{M_b y}{I}, \quad I = \frac{\pi(r_2^4 - r_1^4)}{4}, \quad y = r_2$$

Deflection:

$$\delta = \frac{Fa^3b^3}{3L^3EI}$$

Entering these equations into MATLAB and solving them for a range of pipe sized yields the following plots. The outer diameter was ranged from 1/2 in (0.0127 m) to 1 1/2 in (0.0381 m) tube size. The thickness was held constant at 0.095 in (0.002413 m).

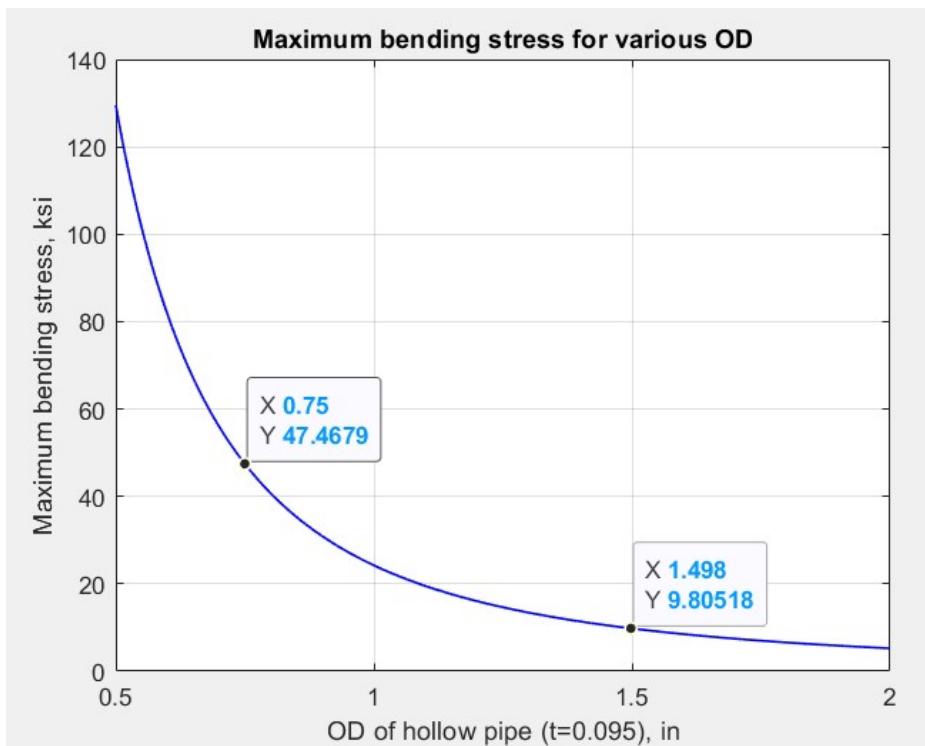


Figure XII.D.1-11: Maximum bending stress for various OD (MATLAB)

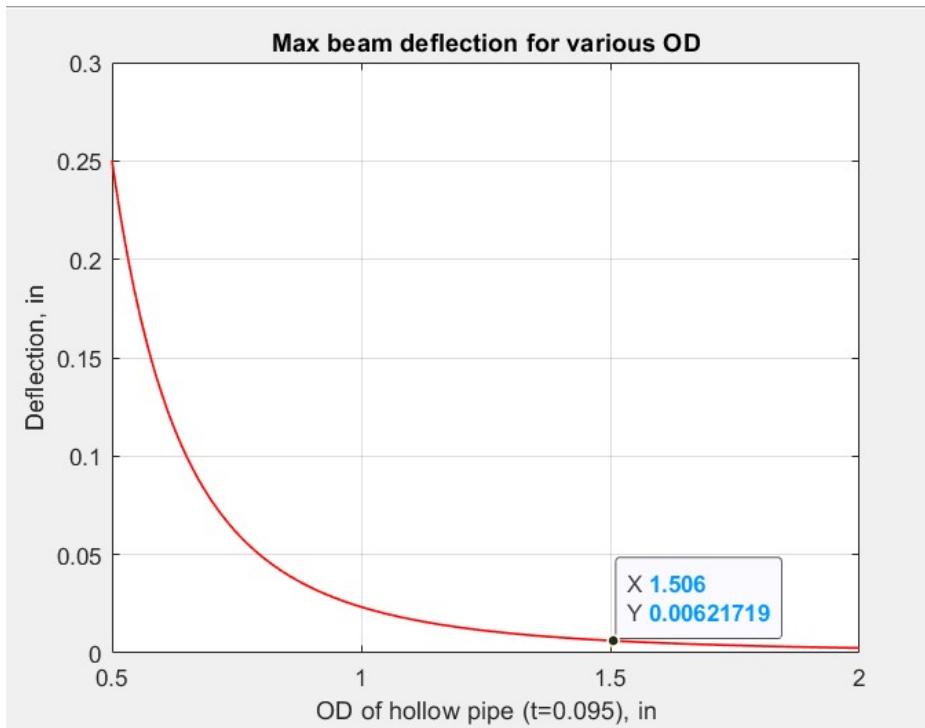


Figure XII.D.1-12: Maximum deflection for various OD (MATLAB)

The first point in figure XII.D.1-12 shows the OD near the yield strength. This sets the lower limit of the tube size. The plot can help size the axle. The second point shows the OD near the OD of

the selected part. This gave the part an acceptable factor of safety. Other considerations when choosing an axle were maximizing diameter to increase contact area with the axle bearing for welding, reducing cost, and what was available from vendors.

According to the Von-Mises failure theory, failure occurs when the Von-Mises stress (σ_{VM}) exceeds the yield strength (σ_y). The Von-Mises stress is given as:

$$\sigma_{VM} = \sqrt{\sigma_1^2 - \sigma_1\sigma_2 + \sigma_2^2}$$

Under the loading conditions laid out in this analysis, there is only axial stress present.

$$\sigma_x = \sigma_b, \quad \sigma_y = 0, \quad \tau_{xy} = 0$$

The principal stresses are:

$$\begin{aligned}\sigma_1 &= \frac{\sigma_x + \sigma_y}{2} + \sqrt{\tau_{xy}^2 + \left(\frac{\sigma_x - \sigma_y}{2}\right)^2} = \frac{\sigma_x}{2} + \frac{\sigma_x}{2} = \sigma_x = \sigma_b \\ \sigma_2 &= \frac{\sigma_x + \sigma_y}{2} - \sqrt{\tau_{xy}^2 + \left(\frac{\sigma_x - \sigma_y}{2}\right)^2} = \frac{\sigma_y}{2} + \frac{\sigma_y}{2} = \sigma_y = 0\end{aligned}$$

The Von-Mises stress reduces to,

$$\sigma_{VM} = \sqrt{\sigma_b^2 - \sigma_b * 0 + 0^2} = \sigma_b$$

The safety factor is therefore:

$$SF = \frac{\sigma_y}{\sigma_b} = \frac{47}{9.8} = 4.8$$

Original MATLAB Code

```
% Rover Axel Bending Stress and Failure
```

```
% Michael Cannon
```

```
% Nov 1, 2023 (started)
```

```
% Dec 6, 2023 (last edited)
```

```
clear, close all, clc
```

```
% Modeling axel as beam fixed by leg-axel brackets (wall supports) with all
% force from frame concentrated at point to generate the highest moment.
```

```
%% Using current axel
```

```
% also locating the point of maximum moment
```

```
r1=0.655*0.0254;
```

```
r2=0.75*0.0254;
```

```
L=24*0.0254;
```

```
x=0:0.01:L;
```

```
a=x;
```

```
b=L-x;
```

```
P=381*0.4536*9.81;
```

```
E=200*10^9;
```

```

Ma=(P.*a.*b.^2)/L.^2;
Mb=(P.*(a.^2).*b)/L.^2;
Mp=(2*P*(a.^2).*(b.^2))/L.^3;
Mmax=max(Ma);
X_max=[0.33,0.66]; % highest moment generated by force concentrated at 1/3
% or 2/3 of the beam length

```

```

figure(1)
plot(x/0.0254,Ma*8.85,'r','LineWidth',1)
hold on
plot(x/0.0254,Mb*8.85,'b','LineWidth',1)
hold on
plot(x/0.0254,Mp*8.85,'g','LineWidth',1)
grid on
title('Primary moments involved in beam')
legend('Moment at support A','Moment at support B','Moment at point force')
xlabel('Location of point force, in')
ylabel('Moment, lbf*in')

```

```

disp('For a beam held by fixed supports at both ends, the maximum bending moment in the beam')
disp('occurs when the force is concentrated at 1/3 or 2/3 of the beam length. (see figure 1)')
disp(' ')

```

```

y_max=r2;
I=(3.14159/4)*(r2^4-r1^4);
SB_max=(Mmax*y_max/I)/1000000; % MPa

```

%% Alternate Axels

```

r2=(0.25*0.0254:0.001*0.0254:1*0.0254);
r1=(r2-0.095*0.0254);
L=24*0.0254;
x=0.33*L;
a=x;
b=L-x;
P=381.0652*0.4536*9.81;
E=200*10^9;

```

```

Ma=P.*a.*b.^2/L.^2;
Mb=P.*(a.^2).*b/L.^2;
Mp=2*P*(a.^2).*(b.^2)/L.^3;
Mmax=Ma;

```

```

I=(3.14159/4).*(r2.^4-r1.^4);
d_max=P*(a^3)*(b^3)./(3*(L^3)*E*I);
y_max=r2;
SB_max=(Mmax*y_max./I)/1000000; % MPa

figure(2)
plot(2*r2/0.0254,SB_max*145/1000,'b','LineWidth',1)
title('Maximum bending stress for various OD')
xlabel('OD of hollow pipe (t=0.095), in')
ylabel('Maximum bending stress, ksi')
grid on

figure(3)
plot(2*r2/0.0254,d_max/0.0254,'r','LineWidth',1)
title('Max beam deflection for various OD')
xlabel('OD of hollow pipe (t=0.095), in')
ylabel('Deflection, in')
grid on

```

disp('Figures 2 and 3 show that smaller axels are still well under failure conditions')
 disp('')

%% Using 1.5 OD Axle (english units)

```

r2_in=0.75; % in
r1_in=0.655; % in
r2=r2_in*0.0254; % m
r1=r1_in*0.0254; % m
L_in=24; % in
L=L_in*0.0254; % m
x=8*0.0254; % m
a=x;
b=L-x;
F_lbf=381; % lbf
F=1695; % N
Mb=(F.*a.*b.^2)./L.^2; % N*m
I=3.14159*(r2^4-r1^4)/4; % m^4
E=190*10^9; % Pa
ymax=r2;
SB_max=(Mb*ymax/I)/1000000; % MPa
SB_max_in_lbf=SB_max*8.85;
Sy_psi=47000; % psi
Sy=324; % MPa

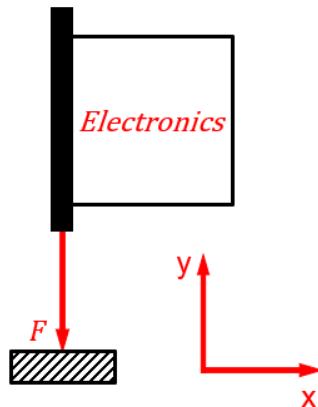
```

$$SF = Sy/SB_{max}$$

XIII.D.1.d. Eng. Analysis and Materials Selection Details for SS1-P2 – Electronics Platform

Kyle Pitre (ME)

Free-Body Diagram:



Assumptions:

- The platform in which the electrical components are attached to will be the first to fail upon impact
- The material of the platform is made of PLA as this was 3D printed
- It is assumed to be dropped vertically (no movement in the x-direction)

Knowns:

- $F_{yield} = 8,700 \text{ psi}$ (platform yield strength)
- $E = 551,143 \text{ psi}$ (platform Young's Modulus)
- $\rho = 78 \frac{\text{lb}}{\text{ft}^3}$ (platform density)

Unknowns:

- Maximum impact velocity (V_{max})
- Maximum drop height (y_{max})

Solving for the maximum velocity:

$$V_{max} = \frac{F_{yield}}{\sqrt{\rho E}} = \frac{(8,700) \frac{\text{lb}}{\text{in}^2} (144) \frac{\text{in}^2}{\text{ft}^2}}{\sqrt{(78) \frac{\text{lb}}{\text{ft}^3} (551,143) \frac{\text{lb}}{\text{in}^2} (144) \frac{\text{in}^2}{\text{ft}^2}}} = 15.9 \frac{\text{ft}}{\text{s}}$$

Solving for the maximum drop height:

$$y_{max} = \frac{V_{max}^2}{2g} = \frac{(15.9)^2 \frac{\text{ft}^2}{\text{s}^2}}{(2)(32) \frac{\text{ft}}{\text{s}^2}} = 3.95 \text{ ft}$$

Applying a factor of safety of 1.5:

$$\frac{V_{max}}{1.5} = 10.6 \frac{\text{ft}}{\text{s}}$$

$$\rightarrow y_{max} = \frac{(10.6)^2}{(2)(32)} = 1.76 \text{ ft}$$

Impact Force:

$$F_{max} = VA\sqrt{\rho E} = (10.6) \frac{ft}{s} (0.06) \text{ ft} (1) \text{ ft} \sqrt{(78) \frac{lb}{ft^3} (551,143) \frac{lb}{in^2} (144) \frac{in^2}{ft^2}} = 50,040 \text{ lb}$$

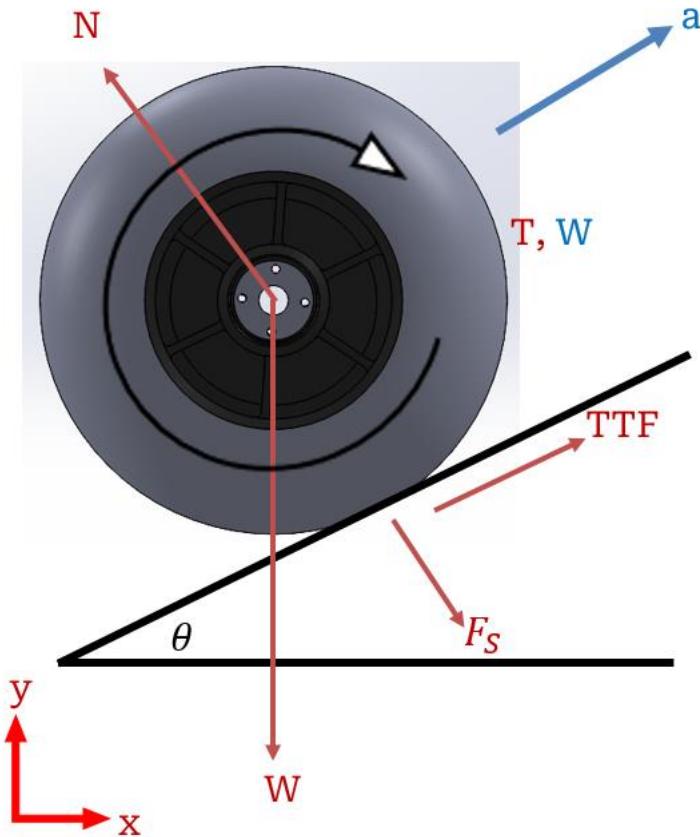
It is important to note that this analysis is conducted assuming the worst orientation in which the rover will experience impact. The rocker-bogie system ensures the rover is always in contact with the ground to reduce the chances of the rover free falling from this height. Additionally, this impact velocity is greater than the maximum speed of the rover, so running into objects will not bring upon component failures.

XIII.D.2. Eng. Analysis Details for SS2 - Drivetrain

Eng. Analysis and Materials Selection Details for SS2-P5 – All Terrain Wheel

Tristan Hughes (ME)

Free-Body Diagram:



Assumptions: No slipping occurs, angle remains constant, μ is known and constant, weight is equally distributed between wheels

Knowns: $W = 220.73 \text{ N}$, $\theta = 10^\circ$, $C_s = .15$, $R = 13 \text{ in}$, $\text{FOS} = 1.15$, $t_a = 3 \text{ s}$, $v_{max} = 1.25 \frac{m}{s}$

Unknowns: T, N, F_s, TTF

$$TTF = RR + GR + F_a$$

TTF = Total Tractive Force (N)

RR = Force to overcome rolling resistance (N)

GR = Force required to climb a grade (N)

F_a = Force required to accelerate to final velocity (N)

Numbers picked were for a 10° angle on sand which was considered the worst case.

$$RR = W_{wheel} \cos(\theta) * C_s = 220.73 \text{ N} * .15 * \cos(10^\circ) = 32.60 \text{ N}$$

$$GR = W_{wheel} * \sin(\theta) = 220.73 \text{ N} * \sin(10^\circ) = 38.33 \text{ N}$$

$$F_a = \frac{W_{wheel} * V_{max}}{g * t_a} = \frac{220.73 \text{ N} * 1.25 \text{ m/s}}{9.81 \frac{\text{m}^2}{\text{s}} * 3 \text{ s}} = 9.37 \text{ N}$$

where: W = weight acting on wheel, C_s = surface friction coefficient, g = gravity, t_a = time to max acceleration, V_{max} = max velocity

$$TTF = 80.31 \text{ N}$$

FOS = 1.15

$$T_w = TTF * R_w * FOS = 80.31 * .1651 * 1.15 = 15.24 \text{ N*m}$$

Torque may be stepped up with a gearbox, so torque required by motor may be designed around that.

A gear ratio of 32:1 was picked, so final torque required by motor goes to:

$$T_F = \frac{T}{GR} = 0.47 \text{ N*m}$$

```

TorqueRequirements_ForceAnalysis_SI.mlx  ConvectionHT_on_BatteryV1.mlx  +
1 % The following is code for the forward motion torque requirements
2 % Written by Tristan Hughes for Capstone Team #57 Fall 2023 - Spring 2024
3
4 % FOS = Factor Of Safety
5
6
7 %%%% Variables
8 M = 135 % taken from max weight of 100 kg + the 35 kg carrying capacity
9 Wtotal = M * 9.81 % N
10 Wwheel = Wtotal / 6 % N, weight per wheel
11 C = .15 % unitless, from friction on sand dunes (worst case)
12 alpha = 10 % degrees, max slope is 5 degrees, doubled for FOS
13 Vmax = 1.25 % m/s
14 ta = 3 % s, time to max speed
15 RF = 1.15 % Resistive Factor (friction on bearings)
16 RW = .1651 % m
17
18 %%%% Equations
19
20 % Rolling Resistance
21 RR = Wwheel * C * cosd(alpha)
22
23 % Grade Resistance
24 GR = Wwheel * sind(alpha)
25
26 % Acceleration Force
27 Fa = (Wwheel * Vmax) / (9.81 * ta)
28
29 % Total Tractive Effort
30 TTE = RR + GR + Fa % in lbf
31
32 % Torque required per wheel
33 Tw = TTE * RW * RF % in lbf*in
34
35 T_motor = Tw / 32
36
37 %%%% Kinematics
38 a = Vmax/ta % m/s^2
39
40
41
42
43
44
45

```

M = 135
Wtotal = 1.3244e+
Wwheel = 220.7250
C = 0.1500
alpha = 10
Vmax = 1.2500
ta = 3
RF = 1.1500
RW = 0.1651

RR = 32.6058

GR = 38.3285

Fa = 9.3750

TTE = 80.3092

Tw = 15.2479

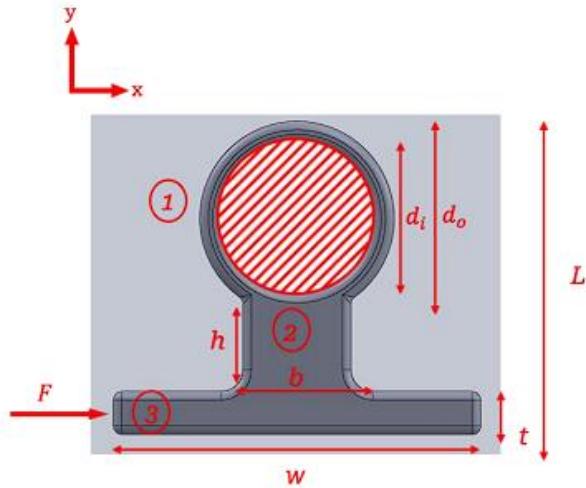
T_motor = 0.4765

a = 0.4167

XIII.D.2.a. Eng. Analysis and Materials Selection Details for SS2-P2 – Motor Attachment

Kyle Pitre

Free-Body Diagram:



Assumptions:

- The component is fixed about the circular section
- The force on the wheel is acting parallel to the axle (in the x -direction)
 - This is taken as the maximum wheel reaction force plus an impact force

Knowns:

- $F = 53,635 \text{ N}$ (maximum reaction force plus impact force)
- Material is steel (taken from manufacturer)
- All dimensions:
 - $L = 0.0388 \text{ m}$
 - $t = 0.0050 \text{ m}$
 - $w = 0.0425 \text{ m}$
 - $b = 0.0125 \text{ m}$
 - $h = 0.0112 \text{ m}$
 - $d_o = 0.0225 \text{ m}$
 - $d_i = 0.0175 \text{ m}$

Unknowns: Bending stress (σ)

The moment of inertia was found by combining the inertias from each section (1, 2, 3):

$$I = \sum(I + A\bar{x}^2)_i = I_1 + A_1\bar{x}_1^2 + I_2 + A_2\bar{x}_2^2 + A_3\bar{x}_3^2$$

$$\rightarrow I = \frac{\pi}{64}(d_o^4 - d_i^4) + \frac{\pi}{4}(d_o^2 - d_i^2)\left(\frac{w}{2}\right)^2 + \frac{bh^3}{12} + hb\left(\frac{w}{2}\right)^2 + \frac{wt^3}{12} + tw\left(\frac{w}{2}\right)^2 = 2.4e - 7m^4$$

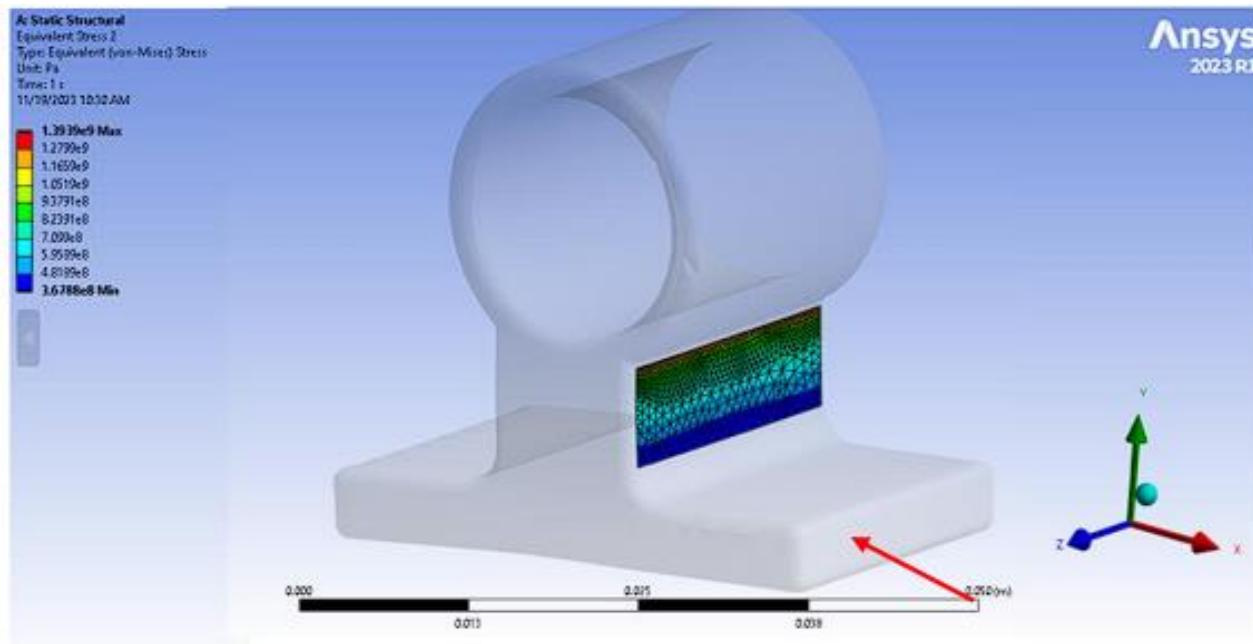
This was also confirmed using the SolidWorks models created for this component. The center of gravity about the y -axis was found for the composite shape:

$$\bar{y} = \frac{\sum A_i y_i}{\sum A_i} = \frac{\frac{\pi}{4}(d_o^2 - d_i^2)\left(\frac{d_o^2}{2} + t + h\right) + hb\left(t + \frac{h}{2}\right) + \frac{t^2 w}{2}}{\frac{\pi}{4}(d_o^2 - d_i^2) + hb + tw} = 0.0125m$$

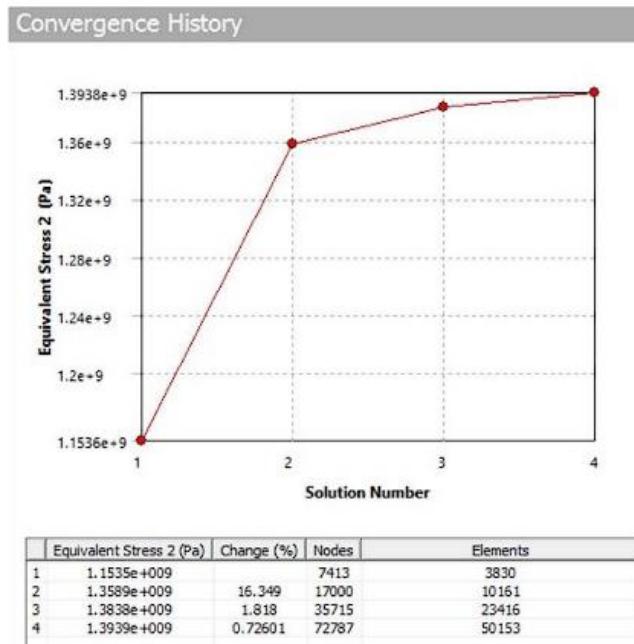
The bending stress was then found and compared to the maximum stress given by the manufacturer:

$$\sigma = \frac{M\bar{y}}{I} = \frac{FL\bar{y}}{I} = \frac{(53,635) \text{ N} (0.0388) \text{ m} (0.0125) \text{ m}}{(2.4e - 7) \text{ m}^4} = 107.4 \text{ MPa}$$

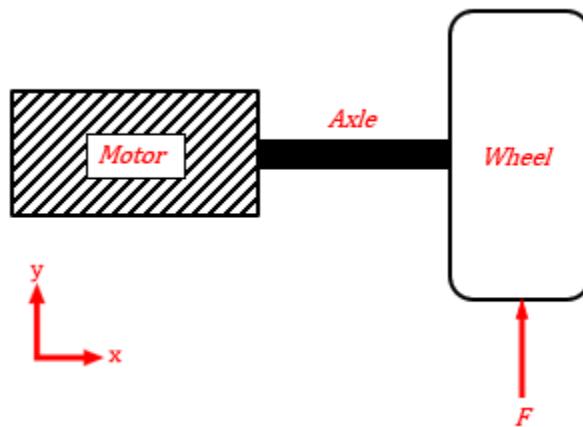
This gives a factor of safety of 4.15 when considering the maximum allowable stress of 445.6 MPa from the manufacturer. These results were also confirmed with the following ANSYS model:



This was also fixed about the circular section, and the force was taken as the maximum reaction force plus the impact force. The convergence confirming the above results are shown below. There is approximately a 20% difference, which is thought to be a result from the sharp edges from the ANSYS model being taken into account.



Free-Body Diagram:



Assumptions:

- The axle is fixed at the motor housing
- The force is equivalent to the maximum wheel reaction force

Known:

- Yield Stress: $\sigma_{aluminum} = 215 \text{ MPa}$
- Young's Modulus: $E_{aluminum} = 196.5 \text{ GPa}$
- Geometry:
 - Outer Diameter: 0.02226 m
 - Inner Diameter: 0.01987 m
 - Length: 0.1524 m
- Force: $F_{max} = 232.3 \text{ N}$

Unknown:

- Maximum bending stress
- Axle deflection

Bending Moment:

$$M = FL = (232.3) \text{ N} (0.1524) \text{ m} = 35.4 \text{ N} \cdot \text{m}$$

Moment of Inertia:

$$I = \frac{\pi}{64} (d_o^4 - d_i^4) = \frac{\pi}{64} [(0.02226)^4 - (0.01987)^4] \text{ m}^4 = 5.15e-9 \text{ m}^4$$

Bending Stress:

$$\sigma = \frac{Mc}{I} = \frac{M \left(\frac{d_o}{2} \right)}{I} = \frac{(35.4) \text{ N} \cdot \text{m} \left(\frac{0.02226}{2} \right) \text{ m}}{(5.15e-9) \text{ m}^4} = 76,505,242 \text{ Pa} = 76.5 \text{ MPa}$$

Considering the yield stress of aluminum, this gives a factor of safety of 2.8 for this component.

Deflection:

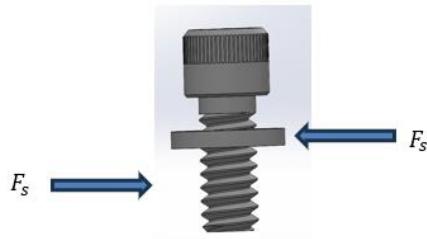
$$\delta = \frac{ML^2}{2EI} = \frac{(35.4) N \cdot m (0.1524)^2 m^2}{(2)(196.5e9) Pa (5.15e-9) m^4} = 4.06e-4 m = 0.4 mm$$

This deflection was used to ensure the motor extension, which runs through the center of the axle, does not come into contact while operating.

XIII.D.2.c. Eng. Analysis and Materials Selection Details for SS2-P5 – Motor Attachment Bolts

Michael Cannon (ME)

A bolt shear failure analysis was conducted on the motor attachment bolts. This is not a lengthy calculation but these bolts form a critical connection and warrant investigation.



FBD of bolt

The shear force on the bolt can be found by the following equation:

$$\tau = \frac{F_s}{A}$$

The shear force was found for a worst-case scenario where the entire weight of the fully loaded rover was applied to just one bolt. This case is exceedingly unlikely. In this scenario, the shear force applied to the bolt is 979 N, and the shear area can be found as

$A = (\pi * D^4)/4 = (3.14 * 0.005^4)/4 = 0.000196 \text{ m}^2$, where D is the diameter of the bolt. The stress is then found as:

$$\tau = \frac{F_s}{A} = \frac{979}{0.000196} = 4.99 \text{ MPa}$$

Using a yield strength of 220 MPa, the maximum shear strength of the bolt is found as:

$$S_b = 0.6 * S_y = 0.6 * 220 = 132 \text{ MPa}$$

$$SF = \frac{S_b}{\tau} = \frac{132}{4.99} = 26.4$$

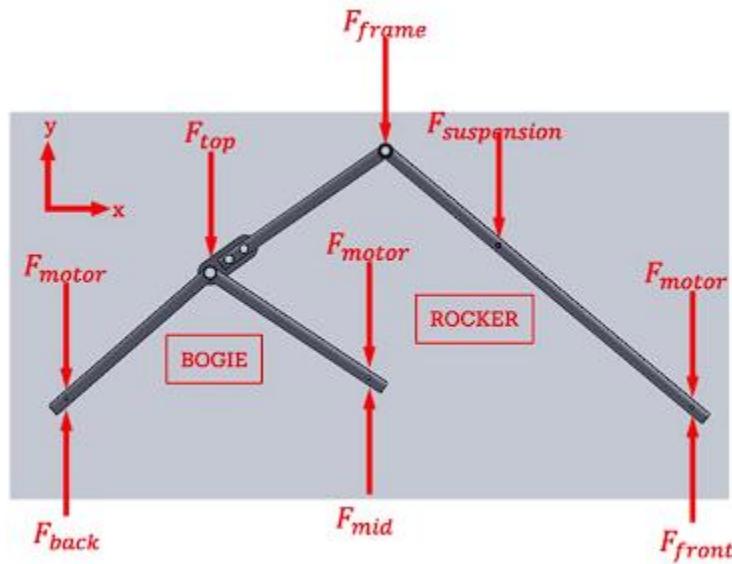
This indicates the part will not fail under the assumed worst-case scenario.

XIII.D.3. Eng. Analysis Details for SS3 - Legs

XIII.D.3.a. Eng. Analysis and Materials Selection Details for SS3-P1/P2 – Rocker-Bogie

Kyle Pitre (ME)

Free-Body Diagram:



Assumptions:

- Force of frame and all of its components are concentrated on one half of the rover
- The rover is on level ground in order to determine the angles of the legs
- The material is aluminum

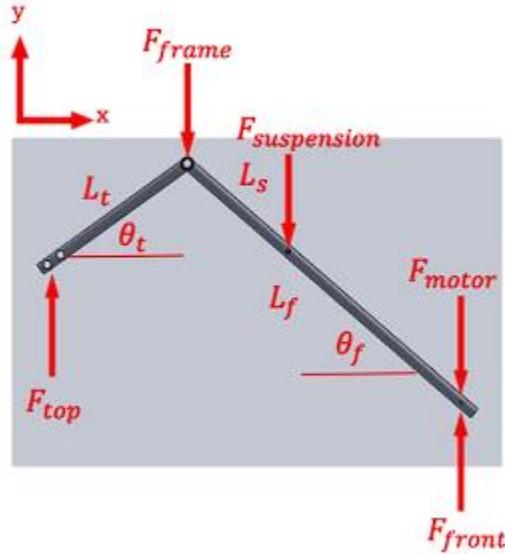
Knowns:

- Frame weight: $F_{frame} = 375 \text{ N}$
- Suspension Weight: $F_{suspension} = 22 \text{ N}$
- Motor Weight: $F_{motor} = 30 \text{ N}$

Unknowns:

- Reaction forces: $F_{back}, F_{top}, F_{mid}, F_{front}$

Considering the rocker separately:



The moment acting about the middle joint (where the axle connects) was first analyzed:

$$\begin{aligned}\sum M = 0 \rightarrow F_{top}L_t \sin(\theta_t) + F_{motor}L_f \sin(\theta_f) + F_{suspension}L_s \sin(\theta_f) &= F_{front}L_f \sin(\theta_f) \\ \rightarrow F_{front} &= \frac{F_{top}L_t \sin(\theta_t) + F_{motor}L_f \sin(\theta_f) + F_{suspension}L_s \sin(\theta_f)}{L_f \sin(\theta_f)} \\ &= F_{top} \left[\frac{L_t \sin(\theta_t)}{L_f \sin(\theta_f)} \right] + F_{motor} + \frac{F_{suspension}L_s}{L_f}\end{aligned}$$

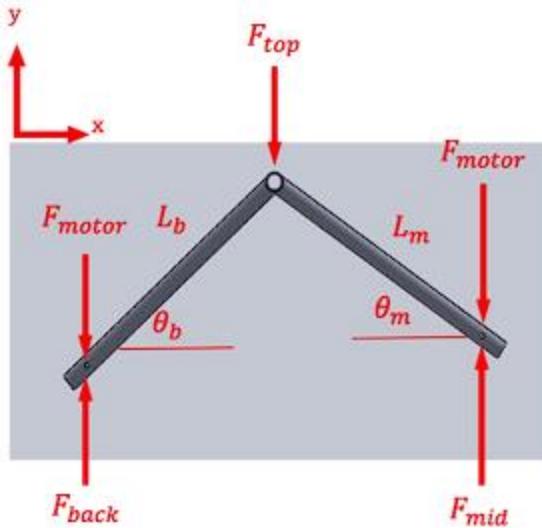
The summation of the forces in the y-direction was set to equal 0:

$$\sum F_y = 0 \rightarrow F_{frame} + F_{suspension} + F_{motor} = F_{top} + F_{front}$$

Combining the two equations:

$$\begin{aligned}\rightarrow F_{top} &= \left[F_{frame} + F_{suspension} \left(1 - \frac{L_s}{L_f} \right) \right] \left[\frac{L_t \sin(\theta_t)}{L_f \sin(\theta_f)} + 1 \right]^{-1} \\ &= \left[375 + 22 \left(1 - \frac{0.205}{0.577} \right) \right] N \left[\frac{0.295 \sin(53.2)}{0.577 \sin(24.2)} + 1 \right]^{-1} = 194.7 N \\ \rightarrow F_{front} &= F_{frame} + F_{suspension} + F_{motor} - F_{top} \\ &= (375 + 22 + 30 - 197.7) N = 232.3 N\end{aligned}$$

The bogie was analyzed next:



XIII.D.3.b. Eng. Analysis and Materials Selection Details for SS3-P3—Leg Bracket FEA

Michael Cannon

Material Selection

The leg bracket will be manufactured out of A36 carbon steel. The primary reason for this is that the differential bar will be water jetted from a plate of A36 steel and there will be leftover material. Using this scrap to manufacture these brackets will save cost by not requiring the purchase of additional materials.

Analysis, FEA

The bracket was run through FEA in Ansys to determine the maximum stresses present.

Assumptions

All of the force passing into the leg is supported by one of the brackets.

The two bolt holes are fixed.

The force is applied to the bearing hole in a direction in line with the leg.

Results

The maximum recorded Von-Mises stress was found to be 5.3 MPa. The yield stress of A36 steel is 250 MPa. This results in a factor of safety of 47.

$$SF = \frac{\sigma_y}{\sigma_{VM}} = \frac{250}{5.3} = 47$$

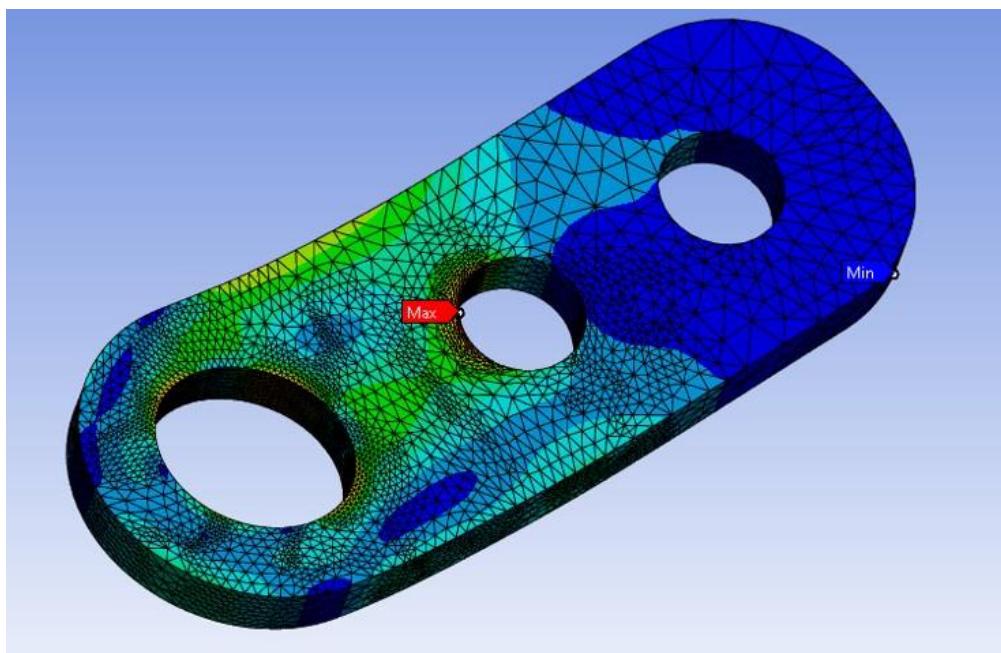
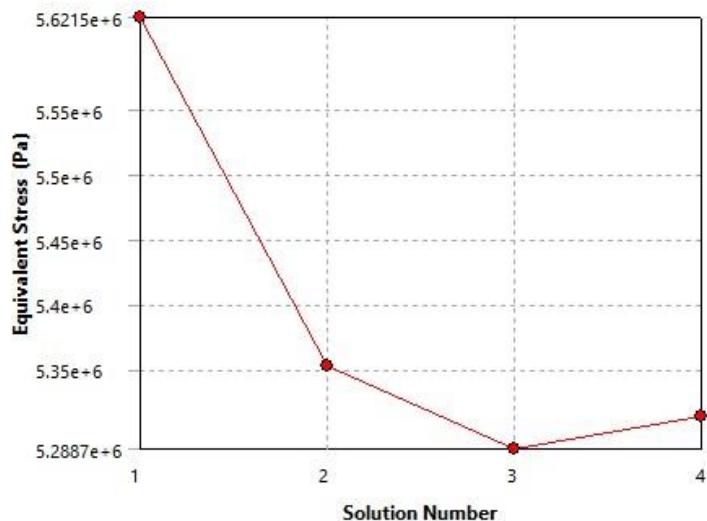


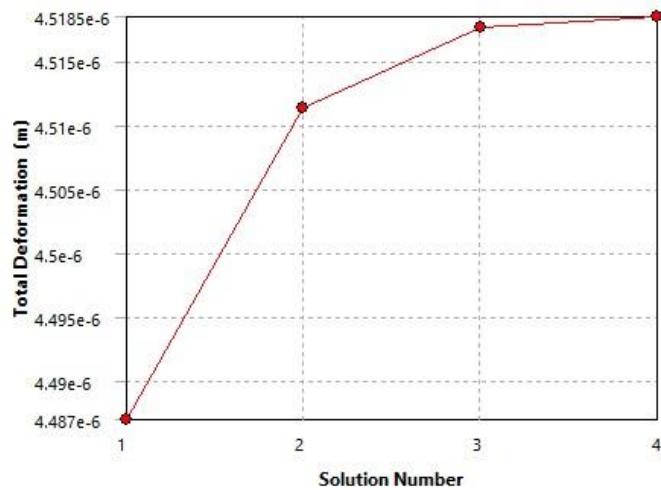
Figure XII.D.3-13: Distribution of Von-Mises stress in bracket under loading. (Ansys)

In order to ensure that the solution was mesh independent, a convergence analysis was conducted in Ansys. The results show both the stress and the deformation solutions converging with increasing mesh size.



	Equivalent Stress (Pa)	Change (%)	Nodes	Elements
1	5.6215e+006		1482	216
2	5.3525e+006	-4.9021	9086	5198
3	5.2887e+006	-1.2003	42497	27567
4	5.3144e+006	0.48493	174000	120033

Figure XII.D.1-14: Convergence plot of stress with increasing mesh (Ansys)



	Total Deformation (m)	Change (%)	Nodes	Elements
1	4.487e-006		1482	216
2	4.5114e-006	0.54249	9086	5198
3	4.5177e-006	0.14125	42497	27567
4	4.5185e-006	1.6999e-002	174000	120033

Figure XII.D.1-15: Convergence of deformation solution with increasing mesh size (Ansys)

XIII.D.4. Eng. Analysis Details for SS - 4 – Differential

XIII.D.4.a. Eng. Analysis and Materials Selection Details for SS4-P1 – Differential Bar Supports: stress and deflection

Michael Cannon (ME)

Material selection

This part needs to be welded to the frame. Therefore, steel is required for good weld compatibility. The grades of steel available for this part were general purpose 1018 carbon steel, low carbon 1020 carbon steel, and structural A36 carbon steel. All had acceptable engineering and manufacturing characteristics. A36 steel was selected on the basis of lower cost.

FEA simulation

Due to the somewhat complex geometry of this part, FEA was chosen as the best approach for this analysis. The software used was Ansys.

The following assumptions were used:

- 1) The entirety of the force is directed down. Since the frame axle is primarily responsible for supporting loads in the horizontal direction, this is a reasonable assumption.
- 2) The force applied to the support was 1114 N. This is the highest force experienced by the differential supports under design loads. This comes from Appendix XII.D.1.a and is part of the worst-case scenario assumptions for reaction forces.

- 3) The force was carried by only one support. Normally this force would be shared by two supports. This is another worst-case scenario assumption.
- 4) The cutout at the top of the support remains fixed. This is the section that will be welded to the frame.

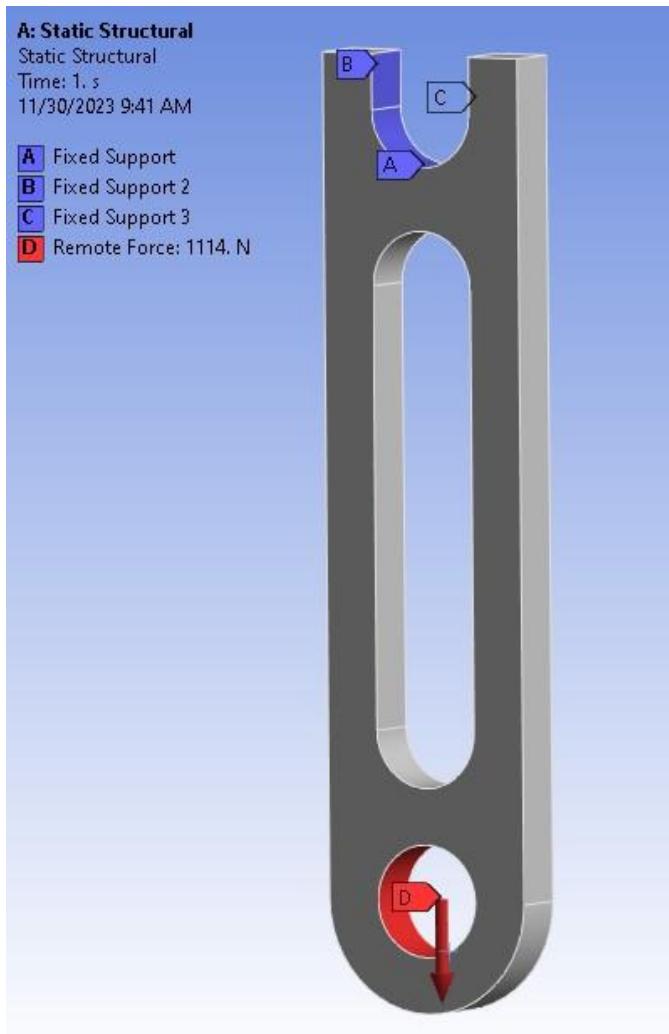


Figure XII.D.4-1: Boundary conditions set in Ansys. (Ansys)
Results

The maximum Von-Mises stress found in the part was 19.4 MPa (2814 psi). The yield strength of A36 steel is 250 MPa (36260 psi). This results in a factor of safety of 12.9. Note that this safety is conservative due to the assumptions. The maximum deflection was found to be 6.3E-6 m (0.00153 in). This level of deflection is negligible.

$$SF = \frac{\sigma_y}{\sigma_{max,vm}} = \frac{250}{19.4} = 12.9$$

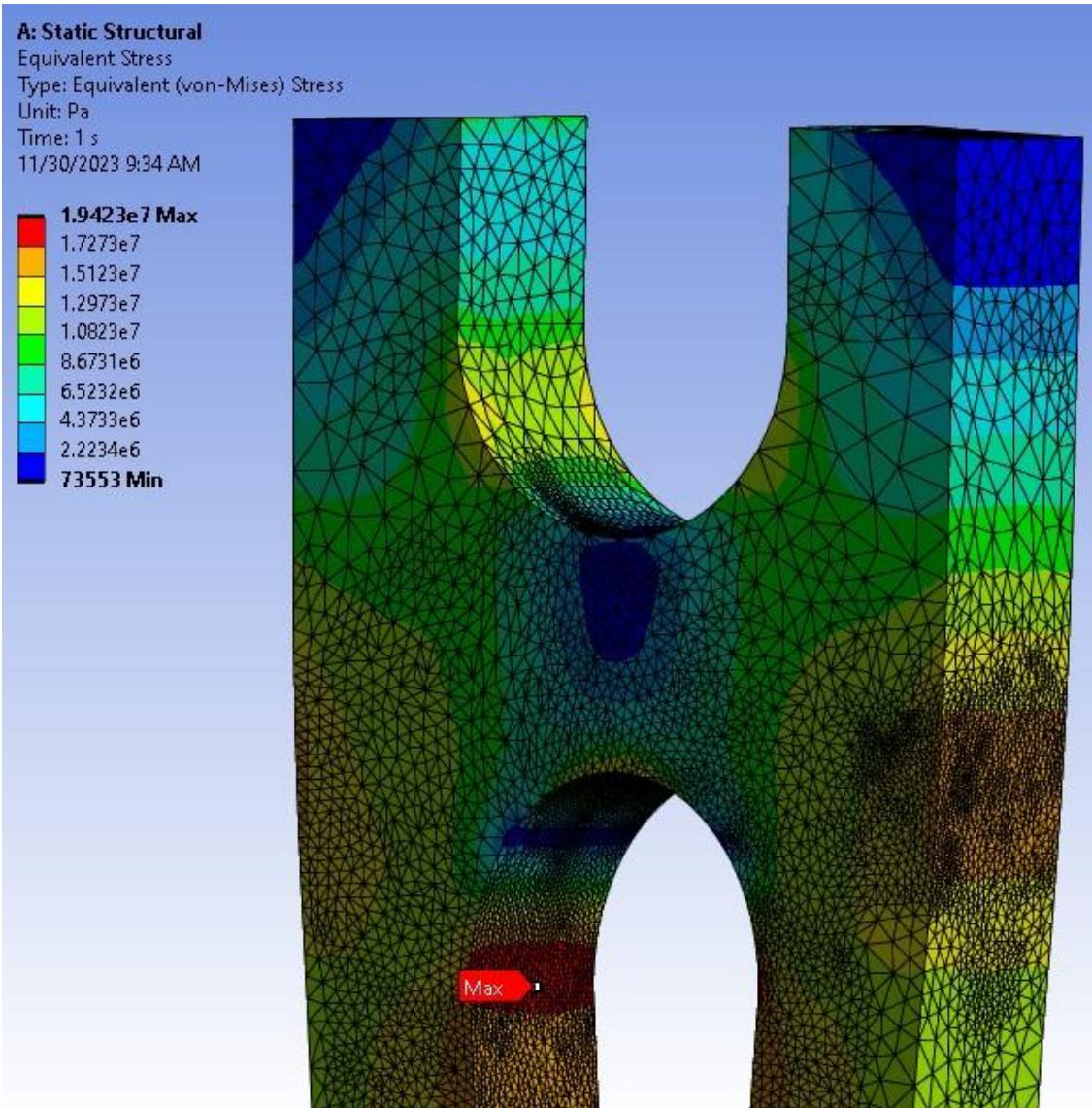


Figure XII.D.4-2: FEA showing maximum stress of differential bar strut. (Ansys)

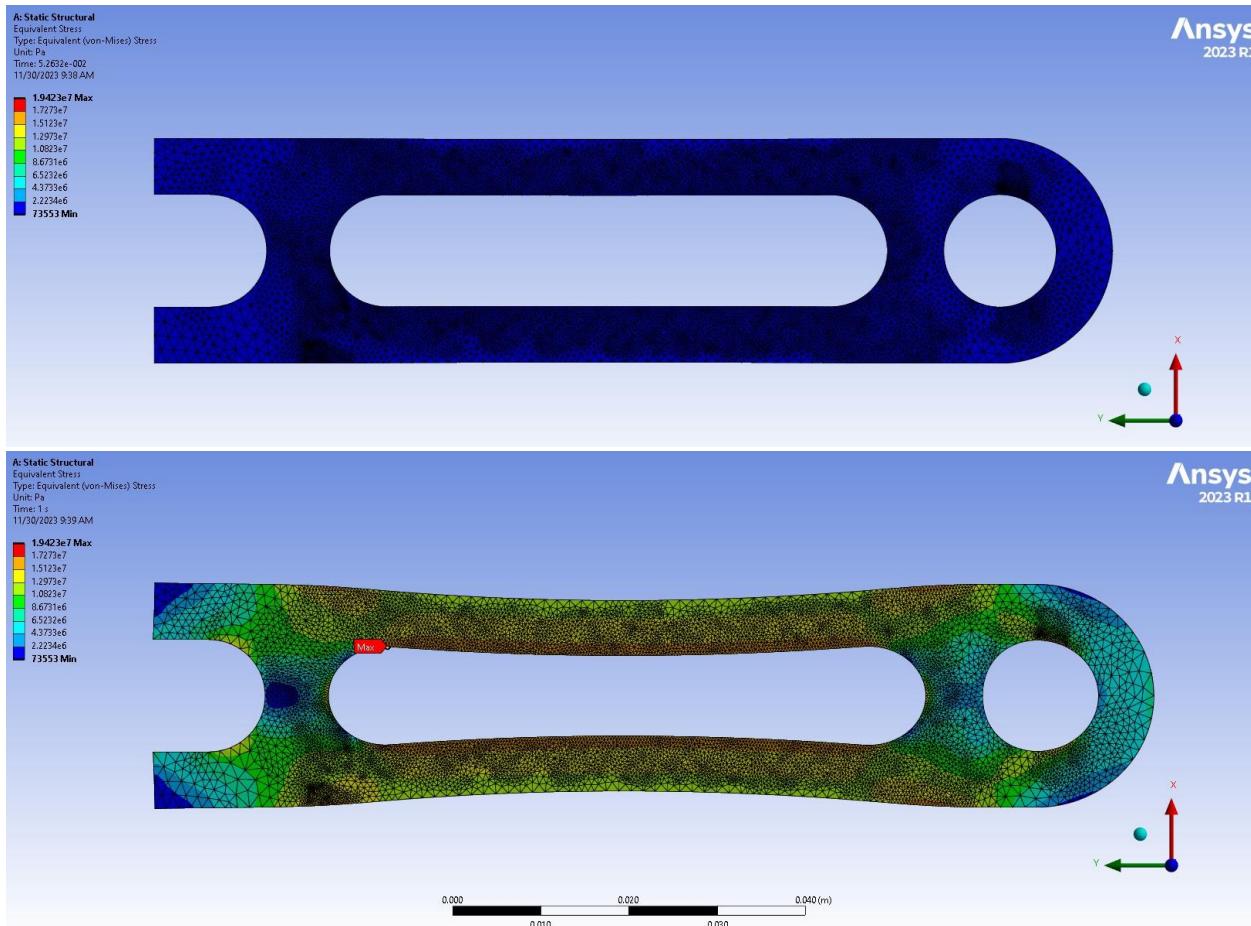
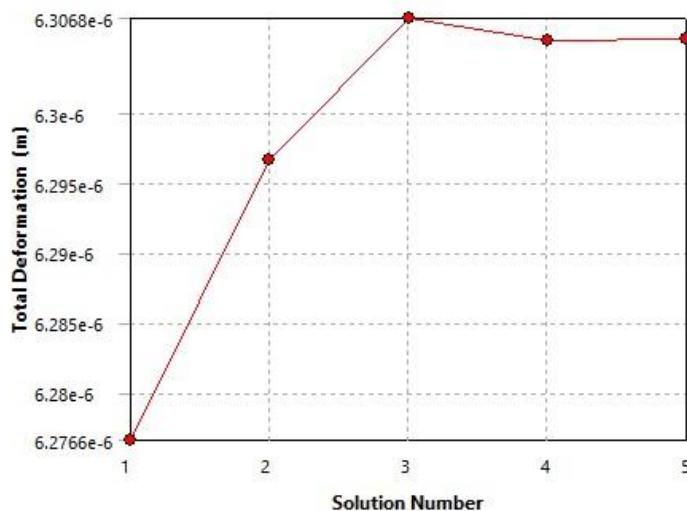


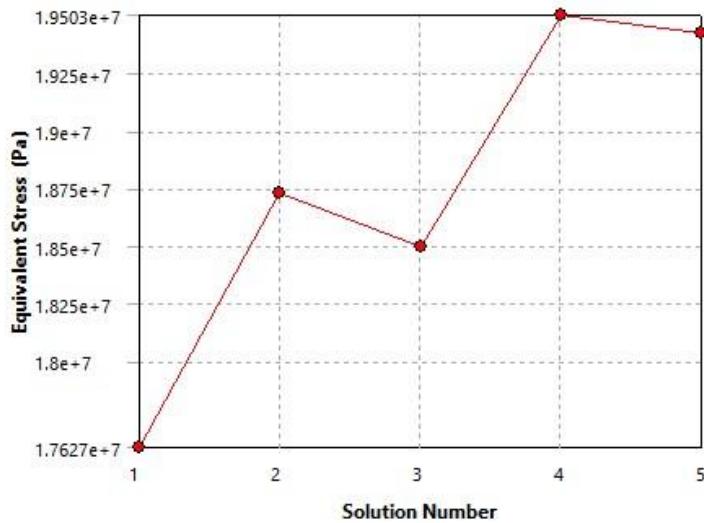
Figure XII.D.4-3 Unloaded and loaded view of differential bar support (Ansys) Validation

To ensure that the solution was mesh independent, a convergence analysis was run in Ansys on both the stress and deflection solutions. The results, shown in the figures below, both show the solution converging with increasing mesh size.



	Total Deformation (m)	Change (%)	Nodes	Elements
1	6.2766e-006		1040	126
2	6.2967e-006	0.32046	6960	3340
3	6.3068e-006	0.16103	42571	25143
4	6.3052e-006	-2.6268e-002	82316	40776
5	6.3053e-006	1.7008e-003	264855	130915

Figure XI.D.4-4 Convergence plot of total deformation of the differential bar support (Ansys)



	Equivalent Stress (Pa)	Change (%)	Nodes	Elements
1	1.7627e+007		1040	126
2	1.8731e+007	6.0758	6960	3340
3	1.8497e+007	-1.2576	42571	25143
4	1.9503e+007	5.2915	82316	40776
5	1.9423e+007	-0.41101	264855	130915

Figure XII.D.4-5: Convergence plot of Von-Mises stress for the differential bar support (Ansys)

Material selection

The part will be manufactured from one of the extra struts that will be cut off the frame. This sets the material for the part at 1010 steel. This saves on cost since purchasing additional materials is not needed. Since the properties of 1010 steel work well for this part, there is no reason to use a different material.

Analysis

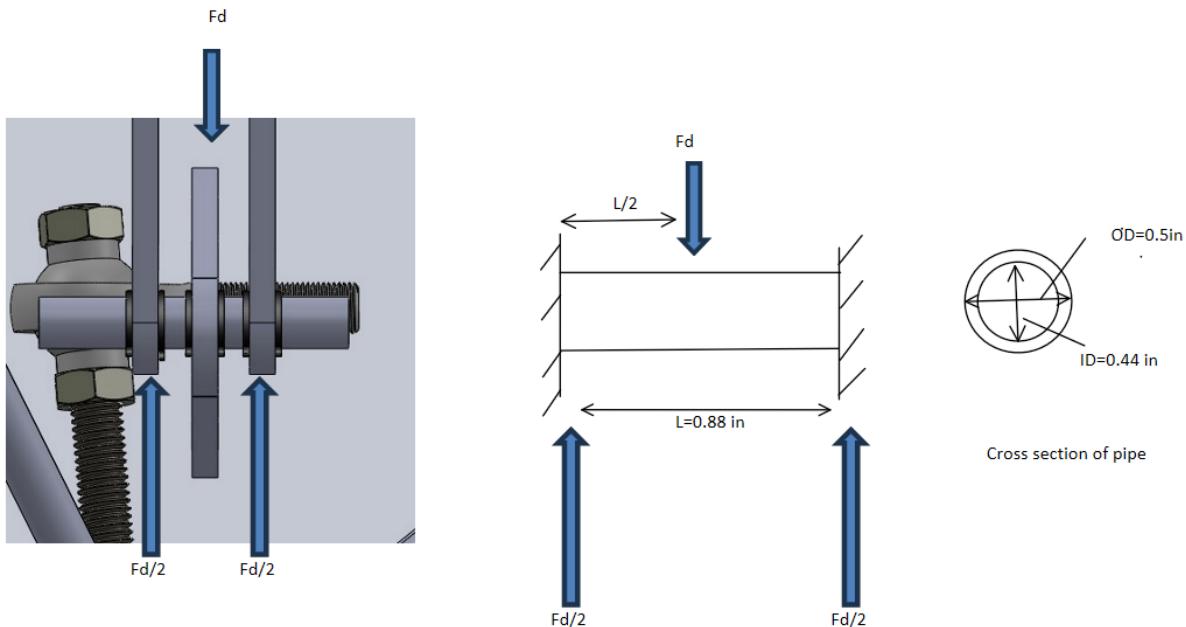


Figure XII.D.4-6: FBD of differential axle. Actual system on the right. Modeled system in the middle. Cross section on the right.

- Assumptions
 - 1) Only vertical forces. Same reasons as in XII.D.4.a.
 - 2) Force applied is considered maximum experienced when operating under design conditions. Reasons stated in Appendix XII.D.41a.
 - 3) Force is applied at the center of the axle.
- Equations used

$$\sigma_b = \frac{M_b * y_{max}}{I}$$

$$I = \frac{\pi * (OD^4 - ID^4)}{64} = \frac{\pi * (0.0127^4 - 0.011176^4)}{64} = 5.112 * 10^{-10} [m^4]$$

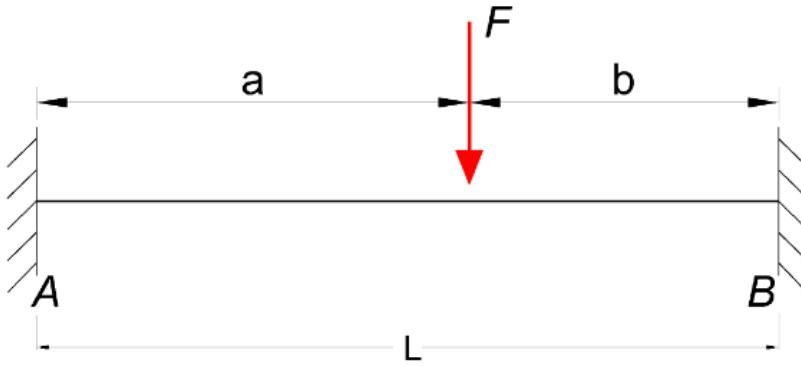


Figure XII.D.4-7: Beam with fixed supports used to simulate axle. [X (engineering toolbox)]

$$M_b = F * a * \frac{b^2}{L^2} = F * \left(\frac{L}{2}\right) * \frac{\left(\frac{L}{2}\right)^2}{L^2} = \frac{FL}{8} = 1114 * \frac{0.022352}{8} = 3.1 [N * m]$$

$$\gamma_{max} = \frac{OD}{2} = 0.00635 [m]$$

$$\sigma_b = 3.1 * \frac{0.00635}{5.112 * 10^{-10}} = 38.7 [MPa]$$

$$SF = \frac{\sigma_y}{\sigma_b} = \frac{305}{38.7} = 7.9$$

Note: Under this loading, Von-Mises failure theory reduces to $\sigma_y = \sigma_b$ for failure. (See appendix XII.D.1.b)

XIII.D.4.c. Eng. Analysis and Materials Selection Details for SS4-P3 – Differential Bar: Stress and deflection FEA

Michael Cannon

Material Selection

In order to save cost and reduce scrap, it was decided that the differential bar should be cut from the same base part as the differential bar supports. This sets the material for the differential bar as A36 structural steel, the same as the support. The manufacturing and material properties of A36 meet the requirements of the part, and by not choosing a different material, a round \$30-\$80 can be saved by not needing to buy a separate base part.

Analysis

The complex geometry of this part required the stress analysis to be done using FEA. This was carried out in Ansys. The following assumptions were used:

- 1) The bar was fixed and supported at the center hole.
- 2) A force of 556 N was applied to each of the bolt holes. (see Appendix XII.D.1.a)
- 3) The forces at the bolt holes are directly downward.

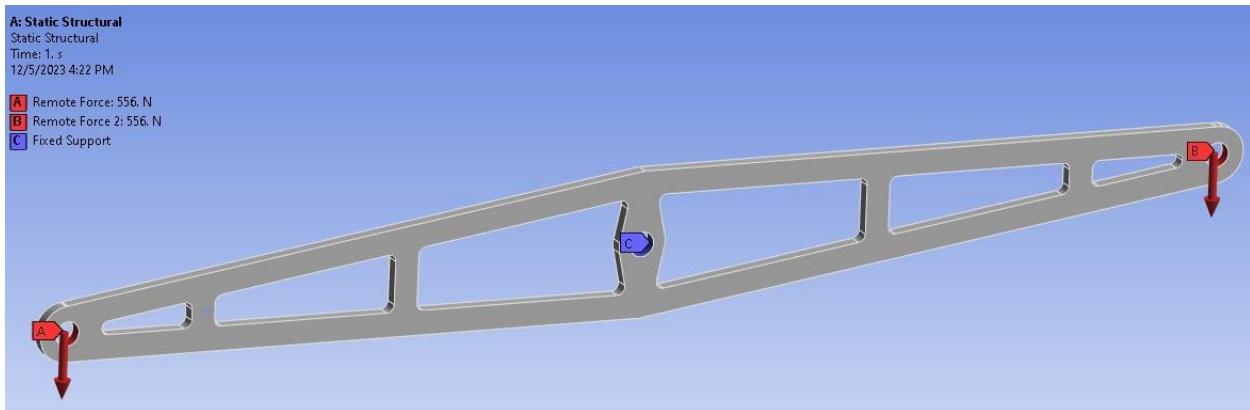


Figure XII.D.4-8: Boundary conditions for differential bar FEA (Ansys)
Results

The maximum Von-Mises stress experienced by the part was found to be 92.4 MPa. This resulted in a safety factor of 2.7

$$SF = \frac{\sigma_y}{\sigma_{max}} = \frac{250}{92.4} = 2.7$$

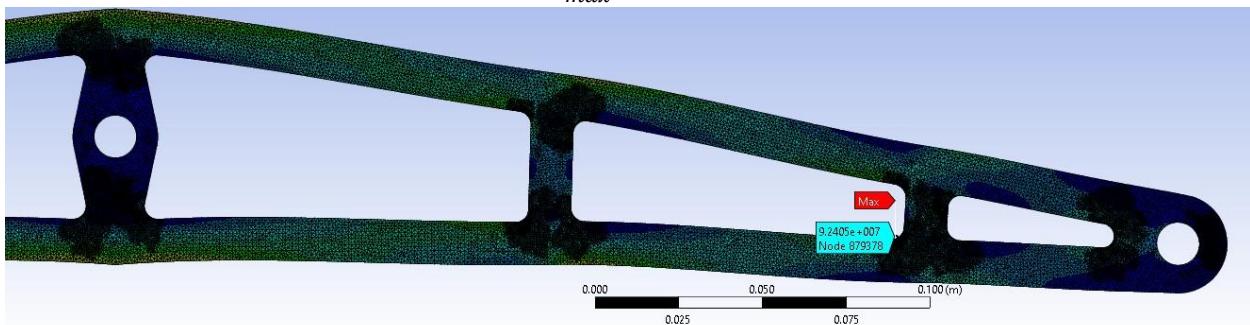


Figure XII.D.4-9: Location of maximum stress on differential bar (Ansys)

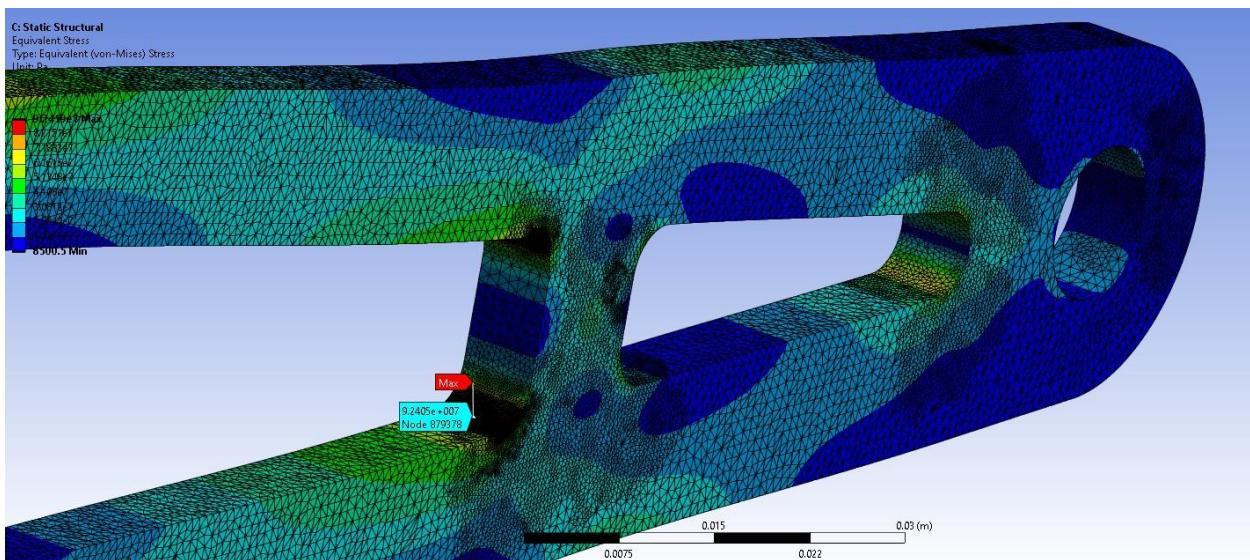


Figure XII.D.4-10: Location of maximum stress on differential bar (Ansys)

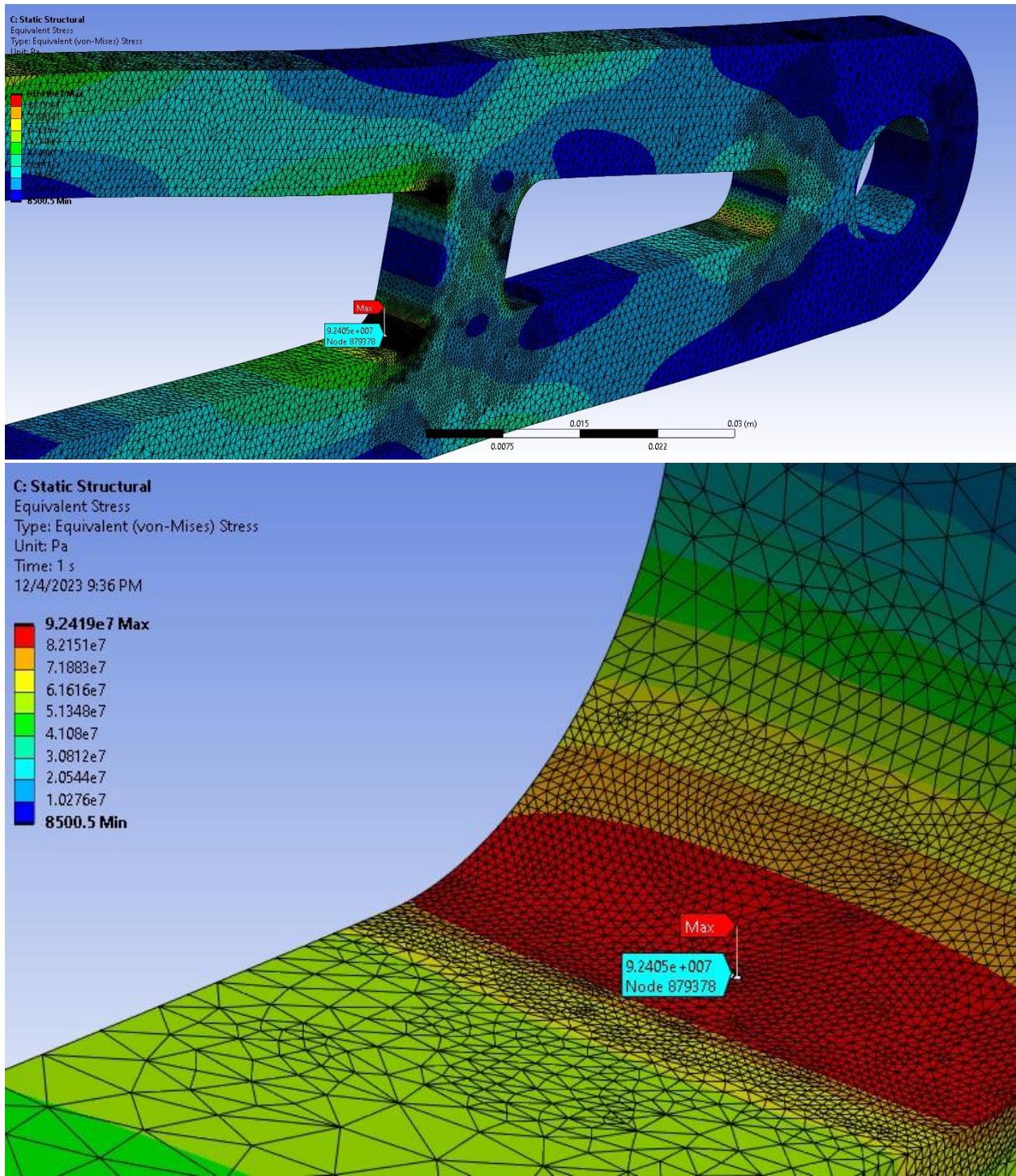


Figure XII.D.4-11: Location of maximum stress on differential bar (Ansys)

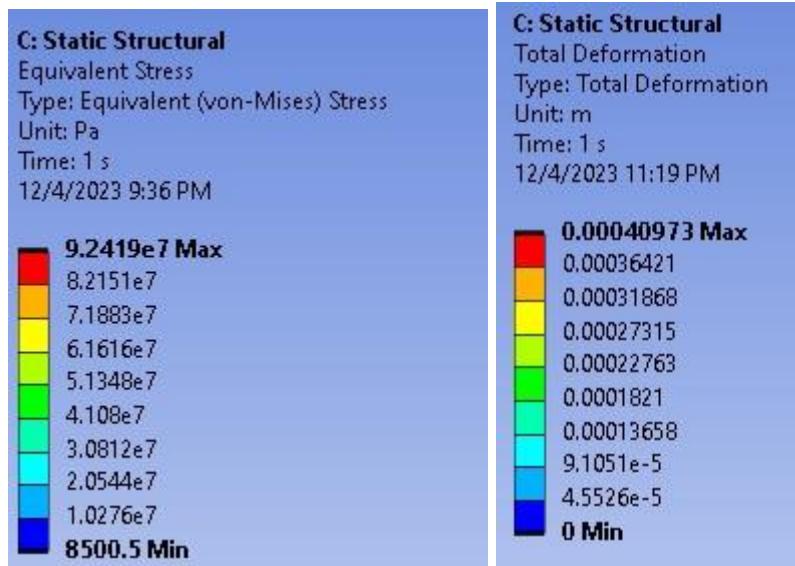


Figure XII.D.4-12 Range of Equivalent Stress and Total Deformation (Ansys)

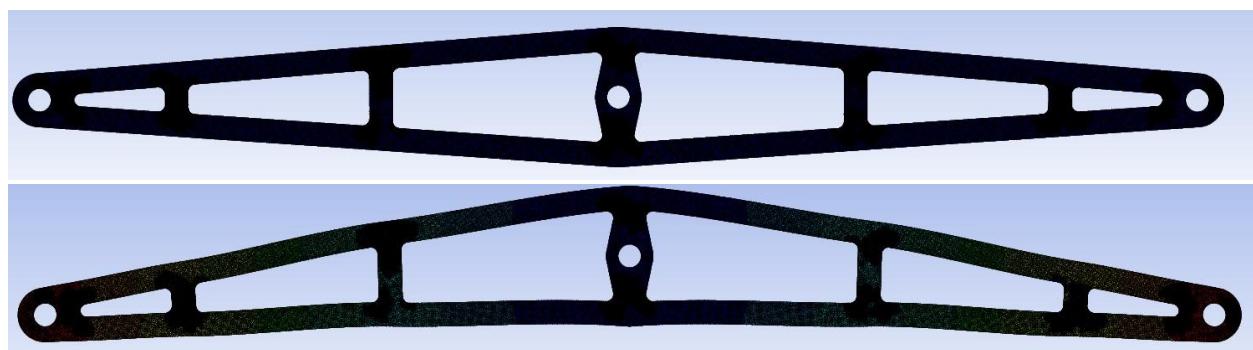


Figure XII.D.4-13: Before and after view of exaggerated deflection from applied load (Ansys)
Validation

To ensure that the solution was mesh independent, a convergence analysis was run in Ansys on both the stress and deflection solutions. The results, shown in the figures below, both show the solution converging with increasing mesh size.

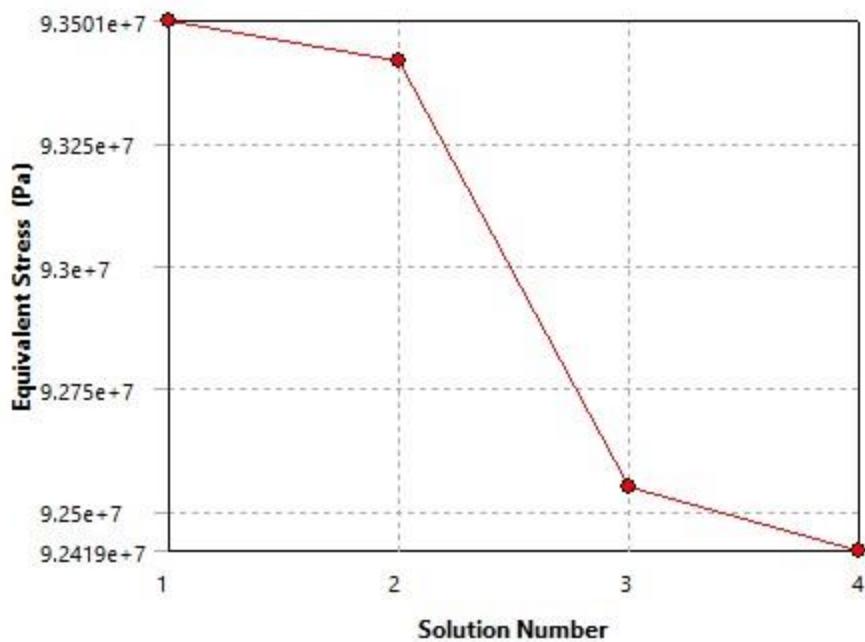
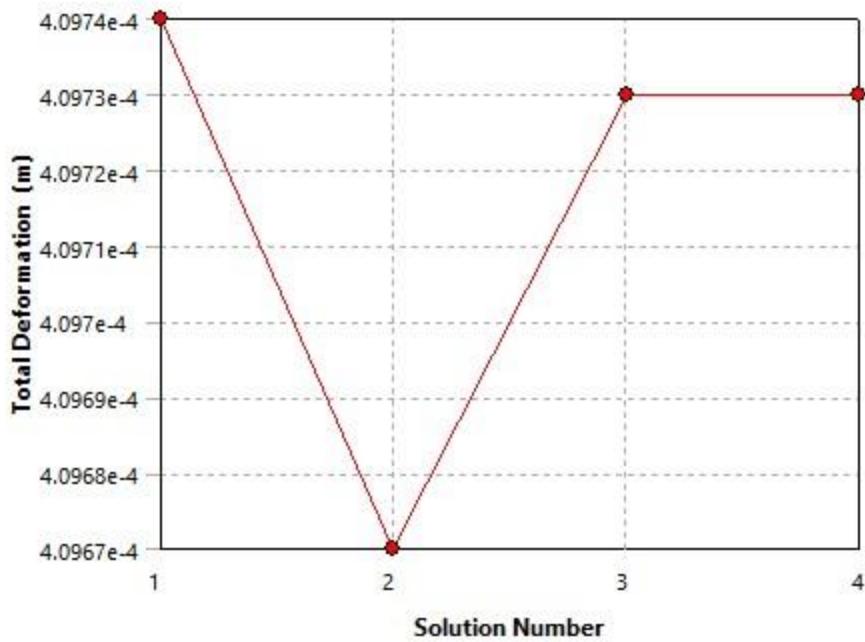


Figure XII.D.4-14: Convergence plot of Von-Mises stress for the differential bar (Ansys)



	Total Deformation (m)	Change (%)	Nodes	Elements
1	4.0974e-004		645949	138600
2	4.0967e-004	-1.7638e-002	418212	209820
3	4.0973e-004	1.4115e-002	878314	514959
4	4.0973e-004	1.3709e-003	1727925	1104112

Figure XII.D.4-15: Convergence plot of total deformation of the differential bar (Ansys)

XIII.D.4.d. *Eng. Analysis and Materials Selection Details for SS4-P4 – Differential Strut: Stress and Buckling*

Michael Cannon

Material Selection

The material chosen for this part was 18-8 stainless steel. Alternate materials were steel grades 2, 8, B7, and 316 as this was what was available for this part from vendors.

	Column2	Column3	Column4	Column5	Column6	Column7
Part	19NP05	10W668	10P636	10P577	19NM92	464W09
Material	Stainless Steel	Stainless Steel	Steel	Steel	Steel	Steel
Grade	316	18-8	2	2 B7		8
Finish	Plain	Plain	Black Oxide	Zinc Plated	Zinc Plated	Black Oxide
Yield strength	30000	45000	45000	45000	105000	-----
Price	11.02	5.96	2.33	3.33	5.46	88.36

Figure XII.D.4-16: Comparison data for selected part. (Grainger)

18-8 steel was chosen due the stainless steel being resistant to oxidation and the price being low. The higher yield strength than 316 stainless steel drove the selection between those two grades.

Analysis

The first part of the analysis was for tensile failure. Since the strut is held by ball joints, there will be no moment produced at the ends of the beam under normal operation. Under this condition, the entirety of the applied force will be directed along the axle. Under this loading, the normal stress in the strut is equal to the principal shear stress and the Von-Mises failure stress is equal to the yield stress.

$$\sigma_n = \sigma_1 \quad \sigma_{VM} = \sigma_y$$

The stress area was found using the minor diameter of the $\frac{1}{2}$ -13 2A threads.

$$A = \pi \left(\frac{d_m}{2} \right)^2 = \pi * \left(\frac{0.4041}{2} \right)^2 = 0.1283 \text{ in}^2$$

External Thread Screw Threads Size Chart

Screw Size	Class Thread	Major Diameter			Pitch Diameter			Minor Diameter
		Basic ^a	Max.	Min.	Max.	Min.	Max.	
1/2-13	2A	0.5000	0.4985	0.4876	0.4485	0.4435	0.4041	

Figure XII.D.4-17: External Thread Screw Size Chart

The maximum force supported by the differential system under design loads is 250 lbf (1114 N). In order to maintain static equilibrium, that force must be split between the reactions at the ends of the differential bar with each side taking half of the load, 125 lbf (556 N).

At its most extreme angle the strut sits at 32.5° . Under this condition, the strut must supply 232 lbf (1032 N) to produce the needed 125 lbf (556 N) needed in the vertical direction. This results in an internal stress of 1814 psi (12.5 MPa) and a factor of safety of 24.8.

$$F = F_s \sin(\gamma_{max}) = 125 * \sin(32.2) = 232 \text{ lbf}$$

$$\sigma_n = \frac{F}{A} = \frac{232}{0.1283} = 1814 \text{ psi}$$

$$SF = \frac{\sigma_y}{\sigma_n} = \frac{45000}{1814} = 24.8$$

A buckling analysis was also conducted to examine the case where the strut is put under compressive stress. The strut was modeled as a beam with pin-pin style supports to emulate the ball joints.

Case	1	2	3	4	5
Constraints					
k	4	1	.25	2.046	1

Figure XII.D.4-18: Different modes of buckling. Case 5 used for this analysis

The following equations were used to determine the critical buckling force. Length was found in SolidWorks.

$$I = \frac{\pi r^4}{4} = \pi * \frac{0.2021^4}{4} = 0.0013 \text{ in}^4, F_{cr} = \frac{\pi^2 EI}{KL^2} = \frac{\pi^2 * 29000000 * 0.0013}{1 * 4} = 23400 \text{ lbf (104 kN)}$$

$$SF = \frac{F_{cr}}{F} = \frac{23400}{232} = 101$$

Original MATLAB code

```
% Rover Differential Strut Strength
% Michael Cannon
% Nov 22, 2023 (start date)
% Dec 4, 2023 (last updated)
```

```
clear; close all, clc
```

```
Sy= 45000; % yield stress (psi)
r=0.5/2; % (in)
r=0.4041/2; % thread minor diameter.
[https://www.engineersedge.com/screw_threads_chart.htm]
A=3.14159*r^2; % (in^2)
P=125; %lbf
gamma_max=32.5; % deg
Fr=P/sind(gamma_max);
```

Sa=Fr/A; % (psi) Pure axial tension loading, all force on one strut

SF=Sy/Sa;

STR=['Axial stress = ', num2str(Sa), ' psi'];

disp(STR)

disp('')

STR2=['Yield stress = ', num2str(Sy), ' psi'];

disp(STR2)

disp('')

STR3=['SF = ', num2str(SF)];

```

disp(STR3)
disp('')

% Buckling

K=1; % Pin-pin ends
E=29000*1000; %psi
https://asm.matweb.com/search/SpecificMaterial.asp?bassnum=mq304a
I=3.14159*(r^4)/4; % in^4
L=4; % in
Fcr=3.14159^2*E*I/(K*L^2);

STR4=['Critical force for buckling = ', num2str(Fcr), 'lbf'];
disp(STR4)

SF_buckling=Fcr/232;

disp('')
STR5=['SF = ',num2str(SF_buckling)];
disp(STR5)

```

XIII.D.4.e. Eng. Analysis and Materials Selection Details for SS4-P1-P5 – Differential System: Kinematics

Michael Cannon

A kinematic analysis was conducted on the differential system to ensure that the system could complete its full range of desired motion without surpassing the mechanical limits of individual components. This process was carried out iteratively in MATLAB. The desired system output was a 10 in (0.254 m) difference in front wheel height. The limits of the system were $\pm 13^\circ$ for the differential bar angle and $\pm 32.5^\circ$ for the swivel angle on the rod-end joints. Iterative adjustments to the geometry driven by this analysis led to a configuration that conformed to these standards.

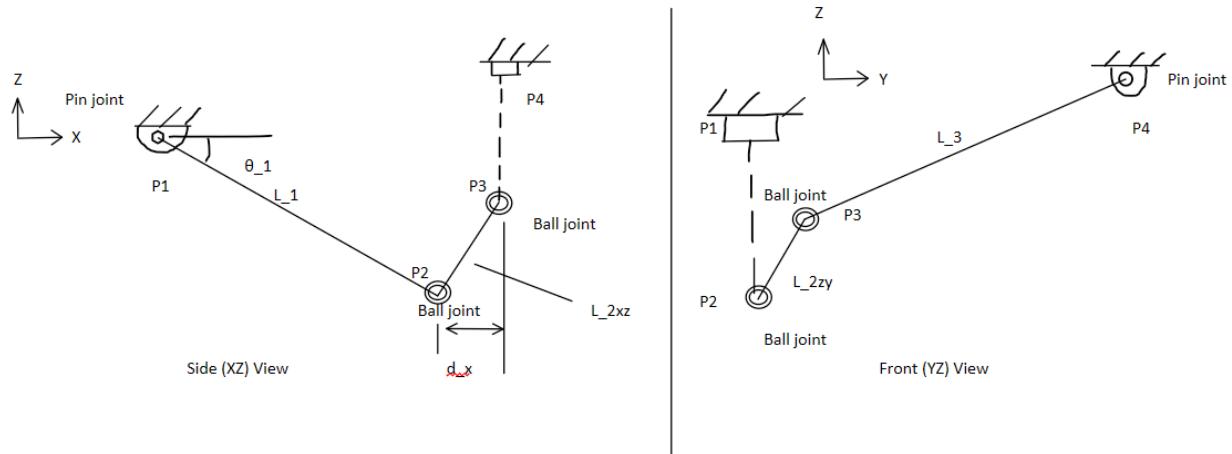


Figure XII.D.4-19: Simplified diagram of links in the differential system.

Known

θ_1 (input): Positive angle of front leg from x-axis

L_1 : Distance along front leg from axle to ball-joint

L_2 : Length along strut between ball-joints

L_3 : Length along differential bar from ball-joint to center.

$P_1(x,y,z)$: Axle location

$P_4(x,y,z)$: Center of differential location

Unknown

$P_2(x,y,z)$: Location of lower ball-joint

$P_3(x,y,z)$: Location of upper ball-joint

θ_3 : Angle between differential bar and y-axis (max allowable $\pm 13^\circ$)

ϕ : Ball swivel angle (max allowable: $\pm 32.5^\circ$)

The following points can easily be determined by geometry.

$$P_{2y} = P_{1y}, \quad P_{3x} = P_{4x}, P_{2y} = L_1 \cos(\theta_1), P_{2z} = -L_1 \sin(\theta_1)$$

The tricky thing about this analysis is that the mechanism is moving in three dimensions. Even though the system is a 1 DOF system, the ball joints have 2 DOF. Another thing to note is that the system is symmetrical about the center of the rover. This means that only one half of the system needs to be solved. Once one half is known the other will behave in an exactly opposite manner. Another important note is that the diagram in figure XII.D.4-19 shows only the *projected* lengths of L_2 . While the length L_2 is constant, the projected lengths on the different planes change over time.

The *projected* distance, d_x , between points P_2 and P_3 on the x-axis can be found as:

$$d_x = P_{4x} - P_{2x}$$

The *projected* distance of L_2 on the XY plane, L_{2xy} can be found as:

$$L_{2zy} = \sqrt{L_2^2 - d_x^2}$$

The next step is to draw an imaginary line between points P4 and P2 projected on a ZY plane centered at point 4.

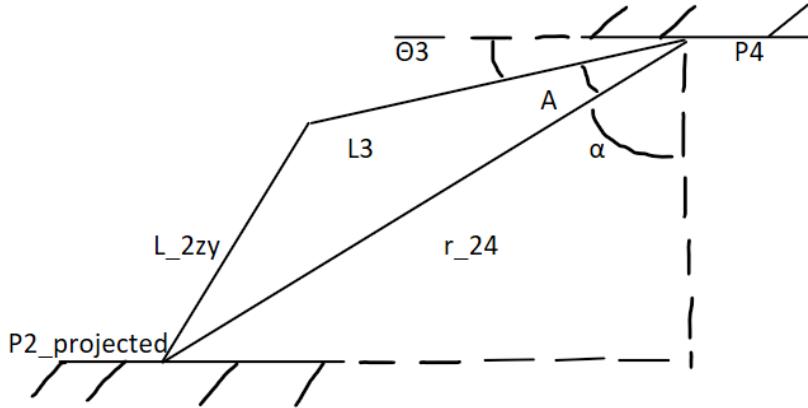


Figure XII.D.4-20: Visual aid for understanding this confusing analysis.

This line can be found as:

$$r_{24} = \sqrt{(P_{2y} - P_{4y})^2 + (P_{2z} - P_{4z})^2}$$

Inserting the known lengths into the general equation for an SSS triangle gives:

$$A = \arccos\left(\frac{r_{24}^2 + L_3^2 - L_{2zy}^2}{2r_{24}L_3}\right)$$

Using equations for a right triangle:

$$\alpha = \arctan\left(\frac{P_{4y} - P_{2y}}{P_{4z} - P_{2z}}\right)$$

The angle of the differential arm can now be found as:

$$\theta_3 = 90^\circ - \alpha - A$$

The angle made between the strut and the XY plane is:

$$\gamma = \arctan\left(\frac{P_{3z} - P_{2z}}{r_{xy}}\right)$$

Wence,

$$r_{xy} = \sqrt{(P_{3x} - P_{2x})^2 + (P_{3y} - P_{2y})^2}$$

The ball swivel angle is therefore:

$$\phi = 90^\circ - \gamma$$

And using distance from SolidWorks, the vertical distance between front wheels is:

$$\Delta h_{fw} = 19.54 \sin(\theta_1) - 8.78$$

This information was entered into MATLAB where the input θ_1 was set to range from 17 to 55 degrees. Outputs of Δh_{fw} , ϕ , and θ_3 were plotted against θ_1 . The geometry was then adjusted in SolidWorks and new lengths entered into the code. This was done until the output conformed to

the required standards. These results were validated by articulating the system in the SolidWorks model and measuring the angles.

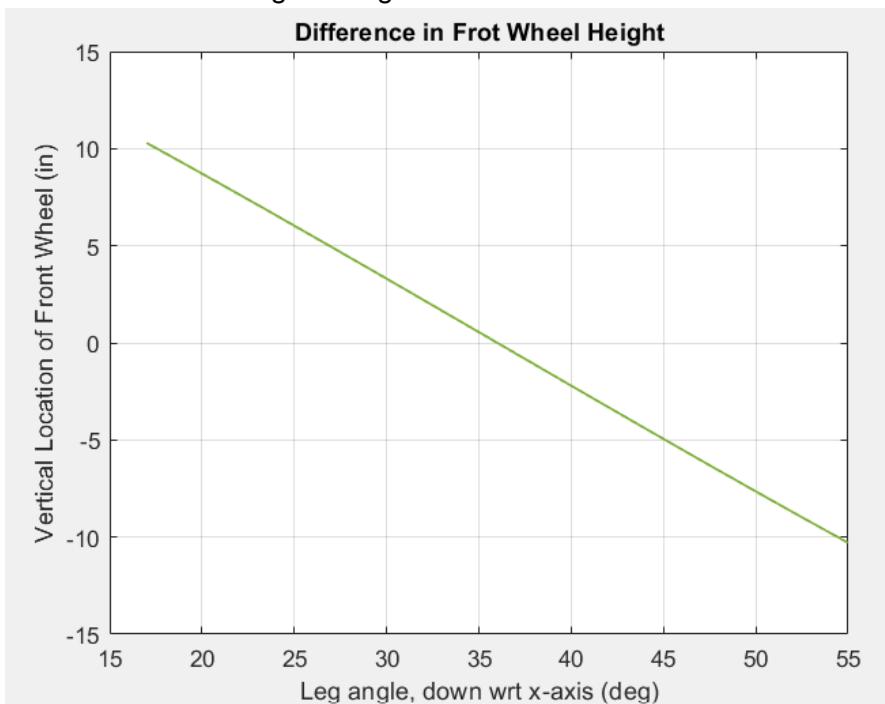


Figure XII.D.4-21: Difference in Front Wheel Height (MATLAB)

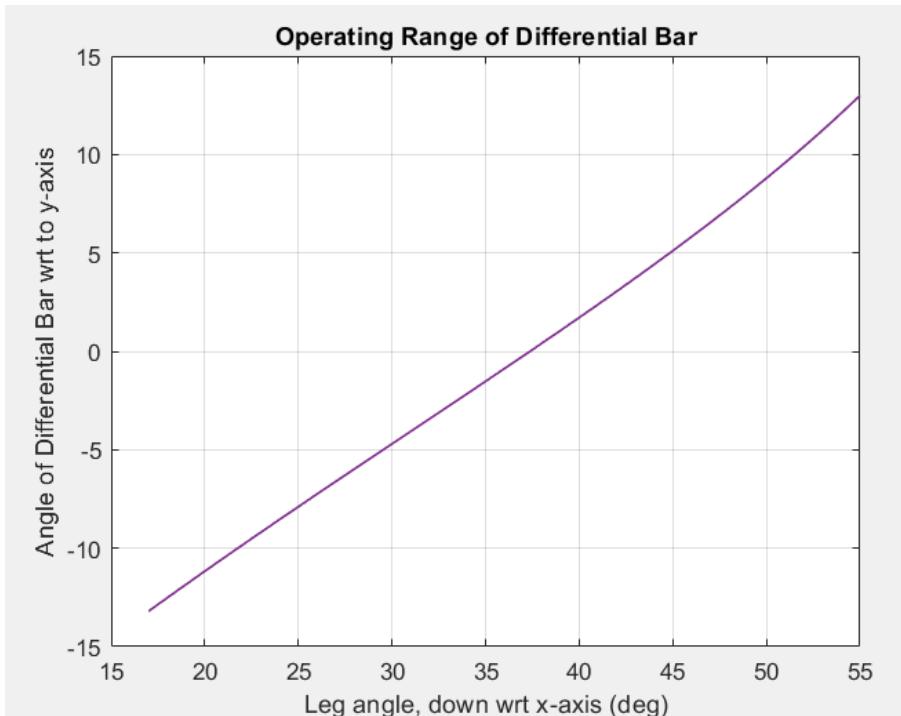


Figure XII.D.4-22: Operating Range of Differential Bar

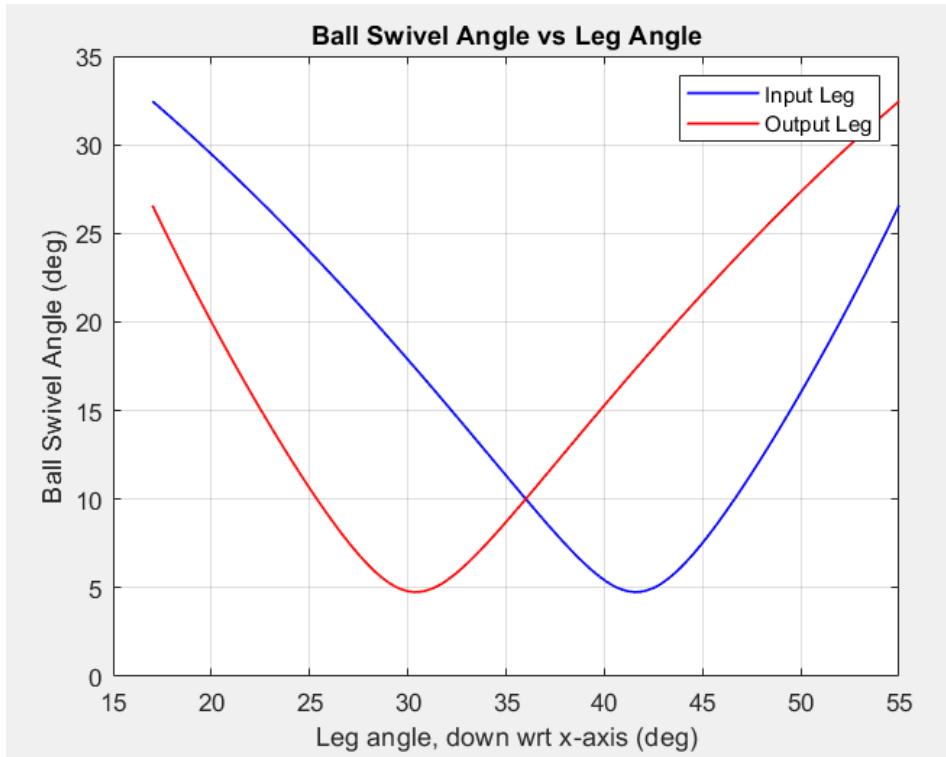


Figure XII.D.4-23: Ball Swivel Angle vs Leg Angle

Original MATLAB code

```
% Rover Differential Kinematics
% Michael Cannon
% Nov 17, 2023 (start date)
% Dec 5, 2023 (last updated)
```

```
clear; close all; clc
```

```
theta_1i=17; % Min leg angle (down wrt to top of frame (x-axis)) (deg)
theta_1f=55; % Max leg angle (down wrt to top of frame (x-axis)) (deg)
```

```
theta_1=[17:0.1:55]; % Operating range of legs
%theta_1=37; % Test theta
```

```
%L1=9.23; % Length of leg to pin (in) V1
%L1=9.23+1.29; %V2
L1=11.06; %V3
%L2=3.41; % Length between pins (in)
%L2=3.41+0.8; %V2
L2=4.52; %V3
L3=12.35; % Half length of differential bar (in)
P1x=0; % Origin at axel
P1y=0;
P1z=0;
```

```

%P4x=7.305; % Location of differential bar axel (in) V1
%P4x=7.305+0.39+0.39; % V2
P4x=8.25; %V3
P4y=12.71; % (in)
P4z=-2.24; % (in)
% location of P2
P2y=P1y;
P2x=L1*cosd(theta_1);
P2z=-L1*sind(theta_1);

P3x=P4x;

dx=P4x-P2x;
L2zy=sqrt(L2.^2-dx.^2);
r24=sqrt((P2y-P4y).^2+(P2z-P4z).^2);
A_rad=acos((r24.^2+L3.^2-L2zy.^2)./(2*r24*L3));
A=A_rad.*180/3.14159;
alpha_rad=atan2(P4y-P2y,P4z-P2z);
alpha=alpha_rad.*180/3.14159;
theta_3=90-alpha-A;

P3y=P4y-L3.*cosd(theta_3);
P3z=P4z-L3.*sind(theta_3);

rxy=sqrt((P3x-P2x).^2+(P3y-P2y).^2);
gamma_rad=atan2(P3z-P2z,rxy);
gamma=gamma_rad.*180/3.14159;
Xi=90-gamma;

Pwz=-19.54*sind(theta_1)-8.78;
Pwz2=-19.54*sind(flip(theta_1))-8.78;
Dh=Pwz-Pwz2;

figure(1)
plot(theta_1,Xi,'b','LineWidth',1)
hold on
plot(theta_1,flip(Xi),'r','LineWidth',1)
grid on
title('Ball Swivel Angle vs Leg Angle')
legend('Input Leg','Output Leg')
xlabel('Leg angle, down wrt x-axis (deg)')
ylabel('Ball Swivel Angle (deg)')

figure(2)

```

```

plot(theta_1,Dh,'color','#77AC30','LineWidth',1)
grid on
title('Difference in Frot Wheel Height')
%legend('Input Leg','Output Leg')
xlabel('Leg angle, down wrt x-axis (deg)')
ylabel('Vertical Location of Front Wheel (in)')

figure(3)
plot(theta_1,theta_3,'color','#7E2F8E','LineWidth',1)
grid on
title('Operating Range of Differential Bar')
%legend('Input Leg','Output Leg')
xlabel('Leg angle, down wrt x-axis (deg)')
ylabel('Angle of Differential Bar wrt to y-axis')

```

XIII.D.5. Eng. Analysis Details for SS5 – Electric Propulsion

XIII.D.5.a. Eng. Analysis and Materials Selection Details for SS5-P1 - Motor

Patrick Maloney (EE)

Given: $V_{max} = 1.25 \frac{m}{s}$ and $r_w = 6.5 \text{ in} = 0.1651 \text{ m}$ where V_{max} is the maximum speed of the rover and r_w is the radius of the wheel, the maximum angular speed of the wheel, ω_L , can be calculated:

$$\omega_L = \frac{V_{max}}{r_w} = \frac{1.25}{0.1651} = 7.571 \frac{\text{rad}}{\text{s}} = 72.30 \text{ rpm}$$

Given the selected gear ratio of $r = \frac{1}{32}$, the maximum speed of the motor can be calculated:

$$\omega_m = \frac{\omega_L}{r} = 32(7.571) = 242.28 \frac{\text{rad}}{\text{s}} = 2313.6 \text{ rpm}$$

We want the rover to accelerate in $t = 3 \text{ s}$ which gives linear and angular acceleration of the rover and wheel respectively of:

$$a = \frac{V_{max}}{t} = \frac{1.25}{3} = 0.417 \frac{\text{m}}{\text{s}} \quad \text{and} \quad \alpha = \frac{a}{r_w} = \frac{0.417}{0.1651} = 2.52 \frac{\text{rad}}{\text{s}^2}$$

The torque calculations for the “All Terrain Wheels” in SS2 Drivetrain give a torque that the motor has to produce with the gear ratio of 32:

$$T_{em} = 0.477 \text{ Nm}$$

Based on the theoretical maximum torque and the maximum speed of the motor, a motor was selected with a gear ratio of 1:32.

In order to get an idea of the voltage and current that the motor will be operating at under different loads, a theoretical duty cycle was created and simulated using a Simulink model. This model is shown below:

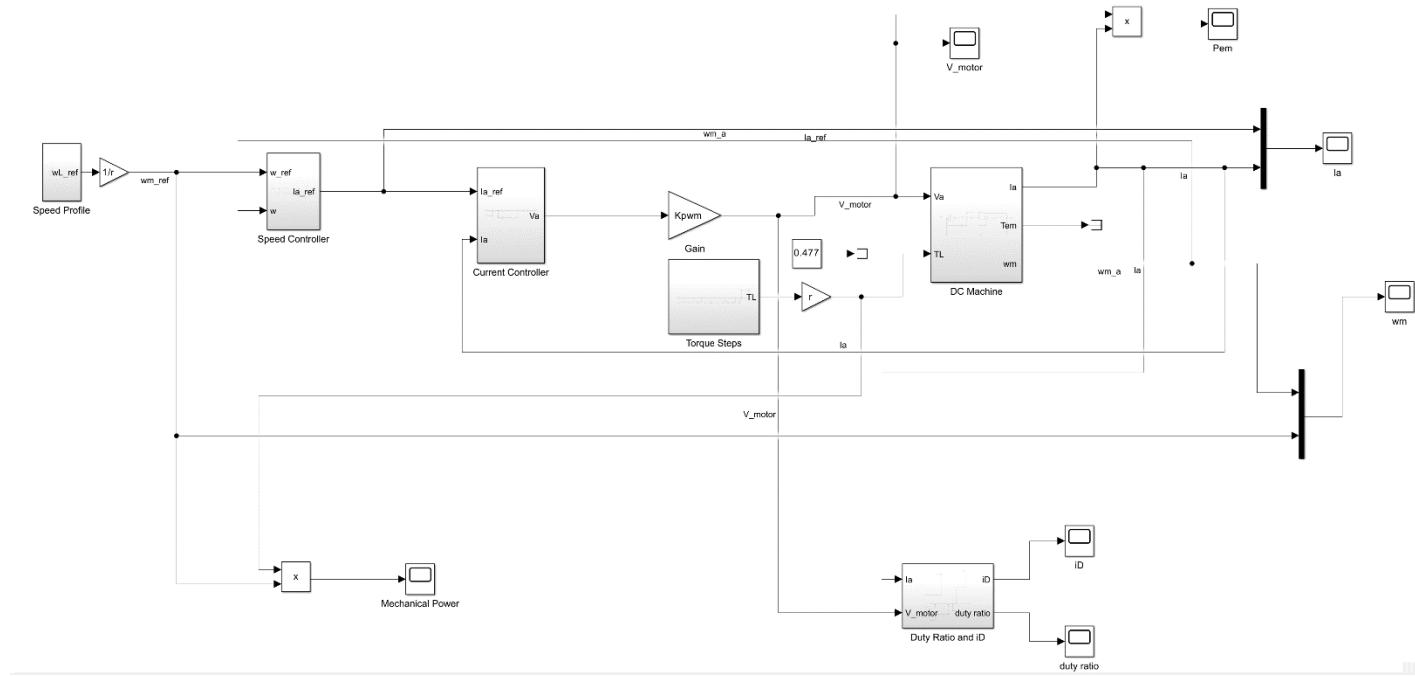


Figure XII.D.5.a.1

Each subsystem of this model will be explained below, with equations and results.

In order to calculate the current drawn from the battery, the following equations for duty ratios must be used:

$$d_a = \frac{1}{2} + \frac{1}{2} \left(\frac{V_a}{V_d} \right)$$

$$d_b = \frac{1}{2} - \frac{1}{2} \left(\frac{V_a}{V_d} \right)$$

$$i_d = (d_a - d_b) I_a$$

where I_a is the armature current of the motor, V_d is the voltage input to the motor controller from the battery, d_a and d_b are the duty ratios, and i_d is the current drawn from the battery.

A subsystem block was created in Simulink to do these **duty ratio calculations** for the various scenarios in Table XII.D.5.a.1 and output the currents seen in the second column of that table. That subsystem block is shown below:

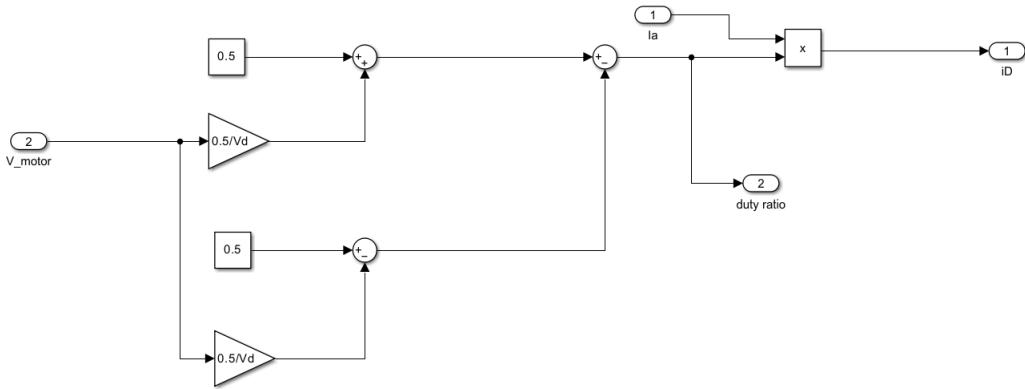


Figure XII.D.5.a.2

In order to get accurate numbers for the current drawn by the motor, the speed and current must be controlled in the model using closed loop feedback. The block diagram for the **current controller** is shown below:

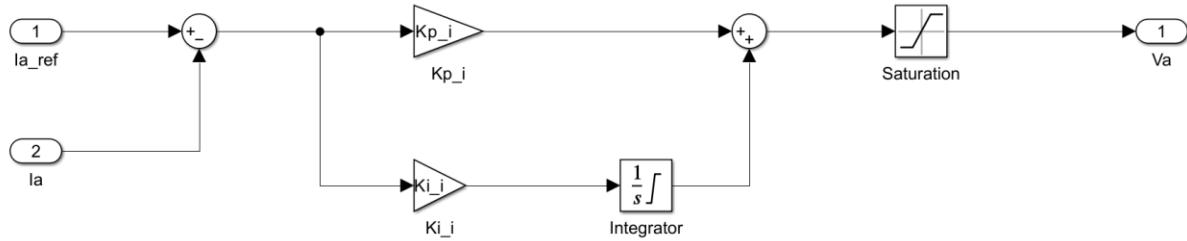


Figure XII.D.5.a.3

K_p_i and K_i_i are the proportional and integral gains for the PI controller and were calculated using the following equations:

$$k_{i,I} = \frac{\omega_{c,I} R_a}{k_{pwm}} = \frac{(3600\pi)(0.19)}{24} = 89.54$$

$$k_{p,I} = k_{i,I}\tau_e = 89.54(0.00842) = 0.754$$

where $\omega_{c,I}$ is the crossover frequency, R_a is the armature resistance, k_{pwm} is the power processing unit gain, and $\tau_e = \frac{L_a}{R_a}$ is the time constant where $L_a = 1.6 \text{ mH}$ was chosen to be an arbitrarily small, non-zero number.

The output of the integrator as well as the whole controller is limited to +1 V and -1 V.

A **speed controller** was also designed for the simulation and the block diagram is shown below:

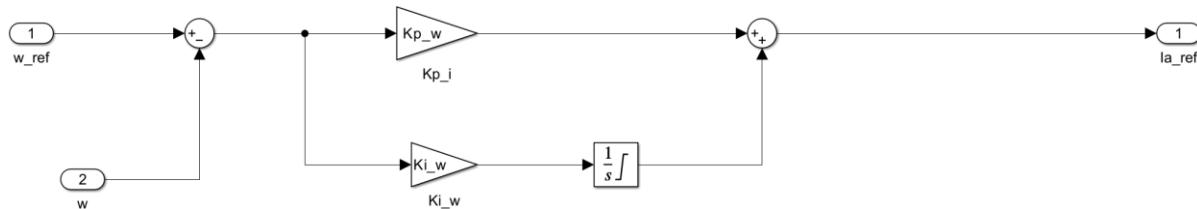


Figure XII.D.5.a.4

K_p and K_i are the proportional and integral gains for the PI controller and were calculated using the following equations:

$$k_{i,\Omega} = -\frac{\cos(90^\circ) J \omega_{c,\Omega}^2}{k_t} = -\frac{\cos(-120^\circ)(0.00028)(360\pi)^2}{0.04025} = 4466.52$$

$$k_{p,\Omega} = -\frac{\sin(90^\circ) J \omega_{c,\Omega}}{k_t} = -\frac{\sin(-120^\circ)(0.00028)(360\pi)}{0.04025} = 6.84$$

where $\omega_{c,\Omega}$ is the crossover frequency for the speed controller, PM is the phase margin, J is the total moment of inertia of the load and rotor, and k_t is the torque constant of the motor. The output of the integrator is limited to the rated current of the motor (+17 A and -17 A).

The DC motor itself also needs to be modeled according to the following equations in order to output correct current and voltage:

$$V_a = E_a + I_a R_a + \frac{di}{dt} L_a$$

$$T_{em} = \frac{r_1}{r_2} T_L + T_f + \left(J_m + \left(\frac{r_1}{r_2} \right)^2 J_L \right) \frac{d\omega_m}{dt}$$

where V_a is the armature voltage, $E_a = \omega_m k_e$ is the back emf voltage, $\frac{r_1}{r_2} = \frac{1}{32}$ is the gear ratio, T_L is the load torque, J_m is the moment of inertia for the rotor, J_L is the moment of inertia for the load, and $T_f = B\omega_m$ is the torque due to friction in the motor.

These equations are realized in the model in the **DC machine** subsystem:

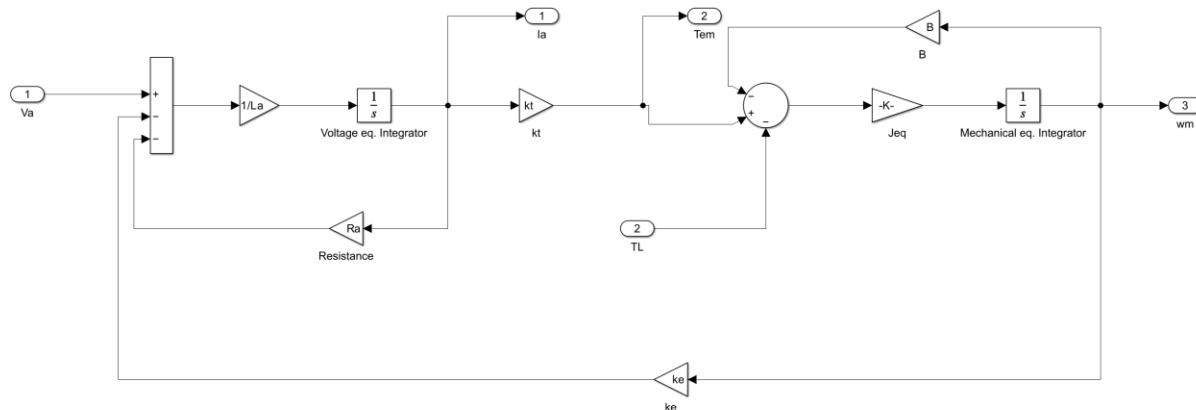


Figure XII.D.5.a.5

Load #	Current (A)	tstart (min)	Tduration (min)	Description	Load torque (Nm)
1	13.1	0	5	2 degree slope, firm grass (C=0.055)	5.55
2	14.9	5	5	2 degree slope, soft grass (C=0.075)	6.38

3	21.66	10	2	5 degree slope, medium mud (C=0.09)	9.19
4	37.34	12	1	10 degree slope, soft mud (C=0.15)	15.25
5	13.1	13	10	2 degree slope, firm grass (C=0.055)	5.55
6	21.66	23	5	5 degree slope, medium mud (C=0.09)	9.19
7	14.9	28	10	2 degree slope, soft grass (C=0.075)	6.38
8	22.12	38	5	7 degree slope, firm sand (C=0.06)	9.38
9	36.46	43	1	15 degree slope, firm grass (C=0.055)	14.85
10	13.1	44	10	2 degree slope, firm grass (C=0.055)	5.55
11	18.19	54	5.5	4 degree slope, soft grass (C=0.075)	7.84
12	36.88	59.5	0.5	17 degree slope, smooth dirt (C=0.025)	15.03

Table XII.D.5.a.1

Above are the results for the simulation. The current is the continuous current drawn from the battery by all six motors at different load torques for different scenarios.

For this theoretical duty cycle, the maximum continuous armature current for one motor is:

$$I_{max} = 12.45 \text{ A}$$

The peak armature current for one motor during a transient is:

$$i_{peak} = 14.27 \text{ A}$$

If we were to accelerate at our desired rate on the roughest terrain we expect to operate on with $T_{em} = 0.477 \text{ Nm}$, then we can expect the peak armature current for one motor to be:

$$i_{peak} = 17.52 \text{ A}$$

For this theoretical duty cycle, the maximum continuous current drawn from the battery is shown in the above table to be 37.34 A.

If we were to accelerate at our desired rate on the roughest terrain we expect to operate on with $T_{em} = 0.477 \text{ Nm}$, then we can expect the peak current drawn from the battery by the motors to be:

$$i_{d,peak} = 41.65 \text{ A}$$

Below are several graphs generated by the simulation for different desired variables:

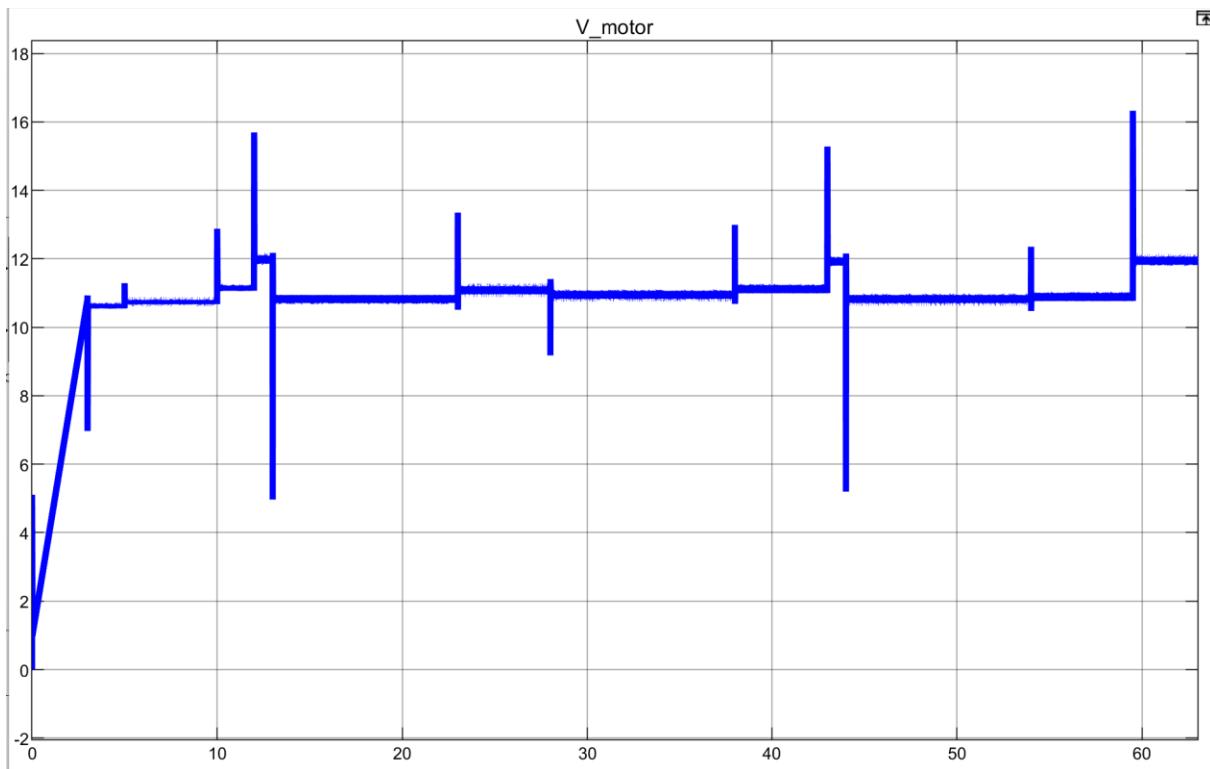


Figure XII.D.5.a.6

This is the armature voltage graph. It is clearly seen that the normal, steady-state, operating voltage for the motors will be between 10 V and 12 V.

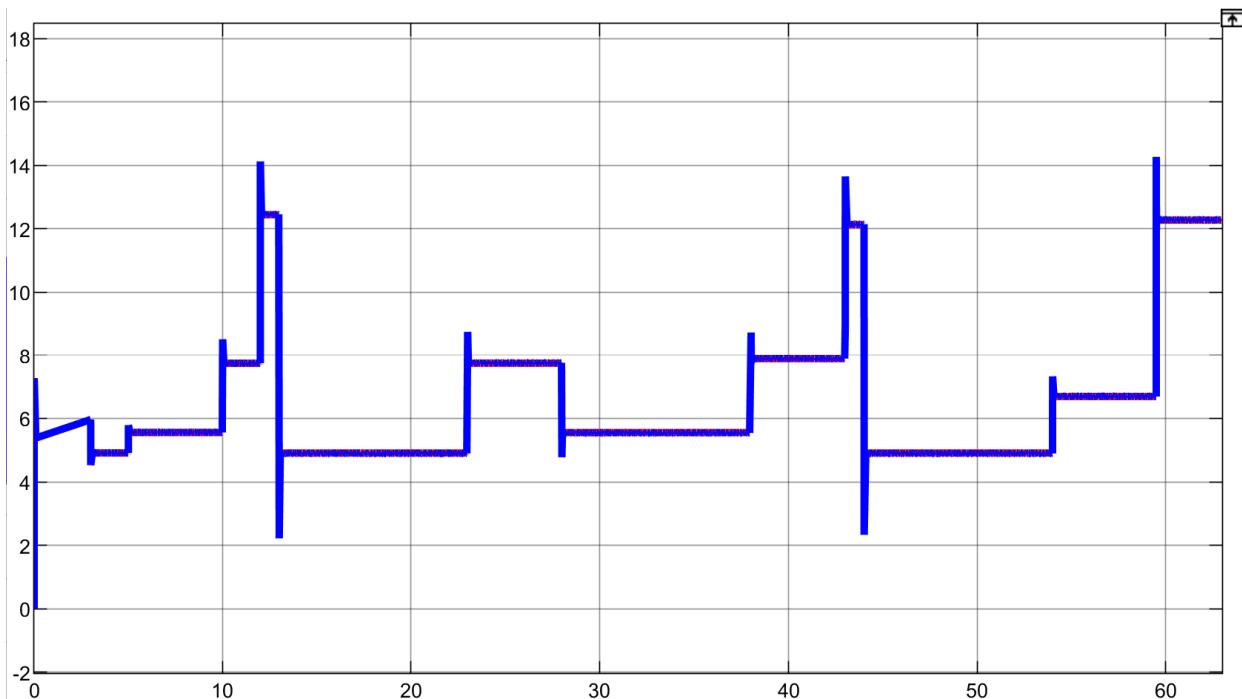


Figure XII.D.5.a.7

This is the armature current graph. The maximum continuous current and maximum peak current were both mentioned earlier in this section.

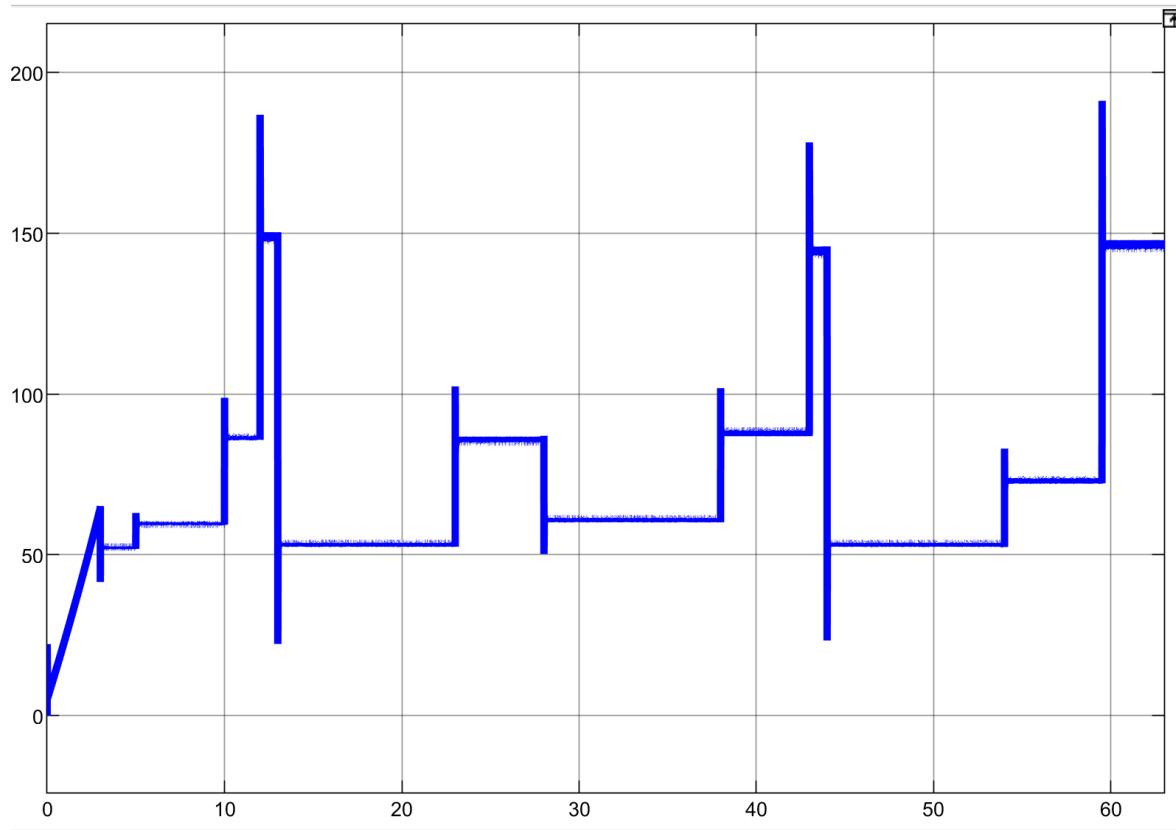


Figure XII.D.5.a.8

Above is the graph for the electrical power of the motor. This is calculated by the simple equation: $V_a I_a = P_{em}$. At our maximum load we expect the motor to have an electrical power of 150 W which is well under the rated power of the motor that we selected.

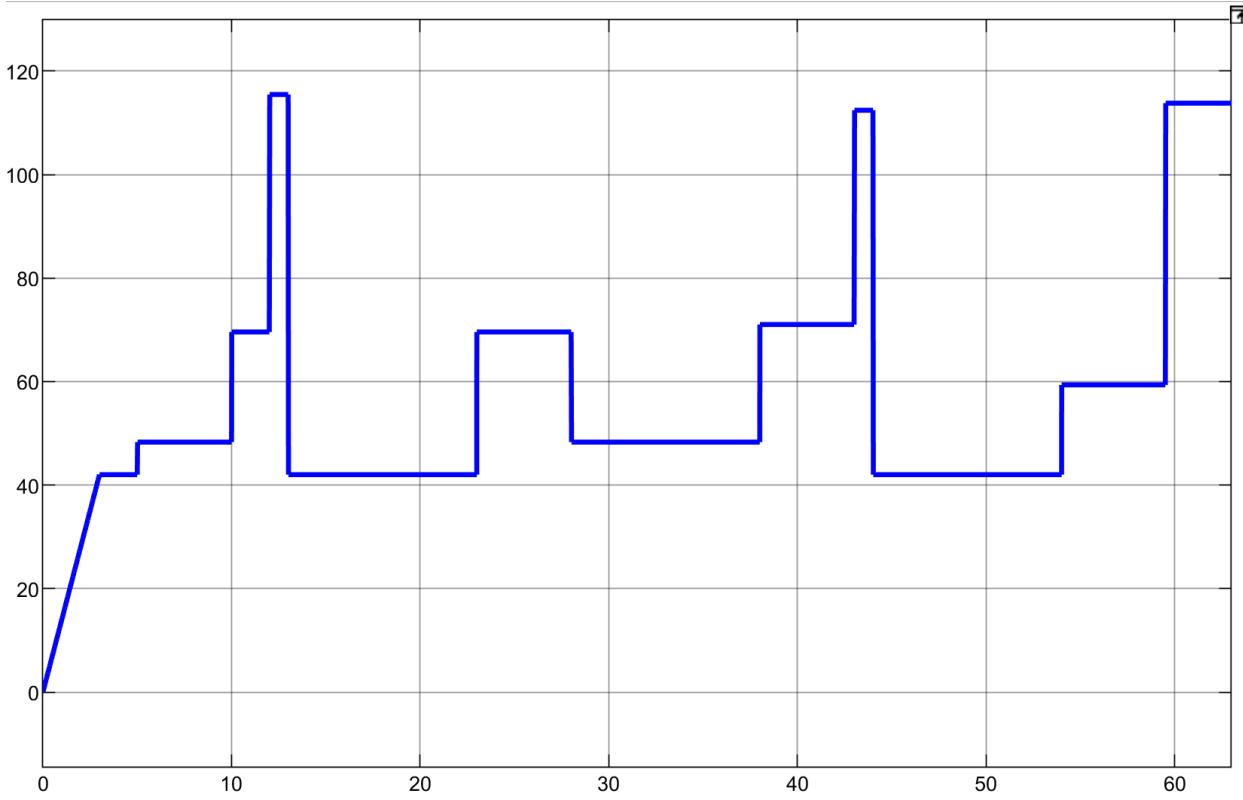


Figure XII.D.5.a.9

Above is the graph for mechanical power which was calculated using the following equation: $T_{em}\omega_m = P_{mech}$. This assumes that there is no loss of mechanical power through the gearbox which is the ideal case and not going to be exact when actually manufactured.

This does however allow for the efficiency of the motor to be verified using the following equation and comparing to the rated efficiency of the motor that was selected.

$$\eta = \frac{P_{mech}}{P_{em}} = \frac{115.5}{150.2} = 76.9\%$$

This efficiency is approximately the same as the rated efficiency of the selected motor so the model should be tuned properly.

XIII.D.5.b. Eng. Analysis and Materials Selection Details for SS5-P2 – Motor Controller

Patrick Maloney

The following is the flowchart for the code of the motor controller. Some of the sections in the flowchart are done automatically by the ESC itself, such as generating the carrier signal and getting the input from the battery. The main parts that are important are determining the direction of motion and then setting the duty cycle.

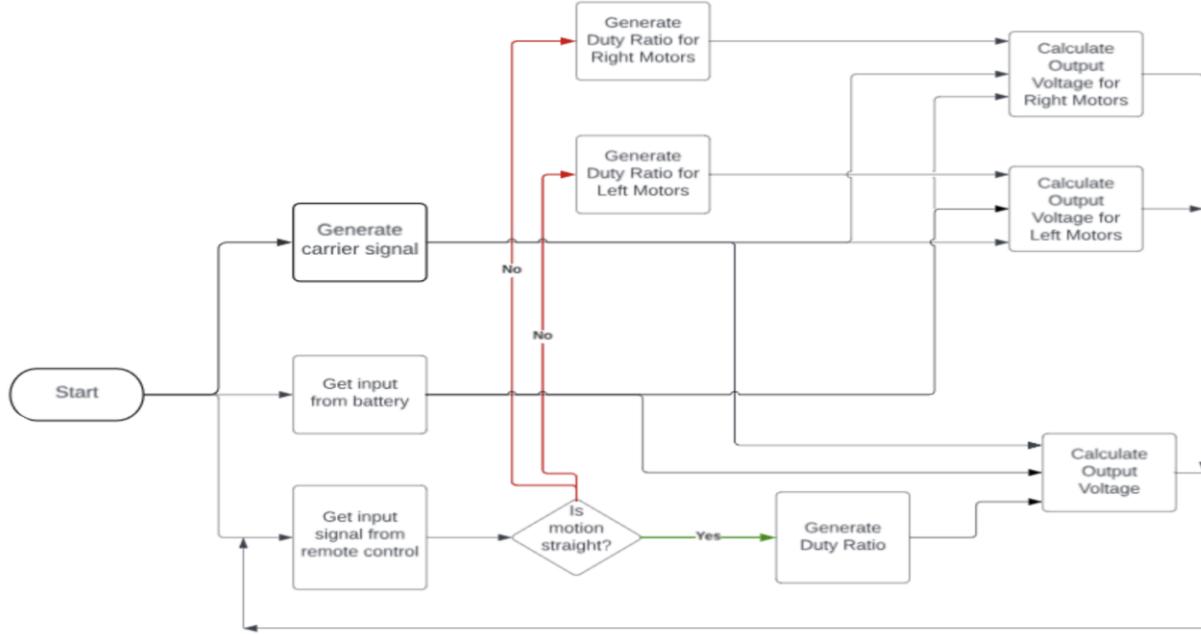


Figure XII.D.5.b.1

The most important thing to note about this flowchart is that there is a difference between straight motion and turning motion. In order to turn clockwise, for example, the motors on the right side of the rover need to turn slower than the motors on the left side of the rover.

Calculations were done and a simulation was run to get the voltage difference for the following realistic scenario: turning clockwise at full speed ($1.25 \frac{m}{s}$) with an angular velocity of $\omega = 0.524 \frac{rad}{s}$ and a turn radius of $r_c = 0.5 m$.

$$V_L - V_R = \omega r_c = 0.262 \frac{m}{s}$$

To maintain the constant linear speed of $1.25 \frac{m}{s}$, V_L must increase by the same amount that V_R decreases.

$$\begin{aligned} V_L &= 1.25 + \frac{0.262}{2} = 1.381 \frac{m}{s} \\ V_R &= 1.25 - \frac{0.262}{2} = 1.119 \frac{m}{s} \\ \omega_{m,L} &= \frac{V_L}{r_w} r = \left(\frac{1.381}{0.1651}\right) 32 = 267.67 \frac{rad}{s} \\ \omega_{m,R} &= \frac{V_R}{r_w} r = \left(\frac{1.119}{0.1651}\right) 32 = 216.89 \frac{rad}{s} \end{aligned}$$

Using the exact same Simulink model that is detailed in the analysis for the motor, it can be found that the voltage for the left motors is about $13.40 V$ and the voltage for the right motors is about $11.02 V$. This results in a voltage difference of approximately $2.38 V$.

Below is the code that we have saved as a backup that we know for sure works so that we can continue adding on and experimenting with new code:

```

1 #include <Servo.h>
2 #include <Wire.h>
3 #include <Adafruit_Sensor.h>
4 #include <Adafruit_BNO055.h>
5 #include <utility/imumaths.h>
6
7 Adafruit_BNO055 bno = Adafruit_BNO055(55);
8
9 #define IN1 3 // Arduino pin 3 is connected to MDD30 pin IN1.
10 #define AN1 5 // Arduino pin 5 is connected to MDD30 pin AN1.
11 #define AN2 6 // Arduino pin 6 is connected to MDD30 pin AN2.
12 #define IN2 4 // Arduino pin 4 is connected to MDD30 pin IN2.
13
14 #define IN1_1 42 // Arduino pin 42 is connected to MDD30 pin IN1_1.
15 #define AN1_1 7 // Arduino pin 7 is connected to MDD30 pin AN1_1.
16 #define AN2_1 8 // Arduino pin 8 is connected to MDD30 pin AN2_1.
17 #define IN2_1 43 // Arduino pin 43 is connected to MDD30 pin IN2_1.
18
19 #define IN1_2 47 // Arduino pin 47 is connected to MDD30 pin IN1_2.
20 #define AN1_2 9 // Arduino pin 9 is connected to MDD30 pin AN1_2.
21 #define AN2_2 10 // Arduino pin 10 is connected to MDD30 pin AN2_2.
22 #define IN2_2 46 // Arduino pin 46 is connected to MDD30 pin IN2_2.
23
24
25 char inChar;
26 char inServo;
27 Servo roverServo;
28 int angle = 60;
29 signed int speedLeft = 0, speedRight = 0;
30 signed int speedLeftActual = 0; signed int speedRightActual = 0;
31
32
33 void setup() {
34   pinMode(IN1, OUTPUT);
35   pinMode(AN1, OUTPUT);
36   pinMode(AN2, OUTPUT);
37   pinMode(IN2, OUTPUT);
38   pinMode(IN1_1, OUTPUT);
39   pinMode(AN1_1, OUTPUT);
40   pinMode(AN2_1, OUTPUT);
41   pinMode(IN2_1, OUTPUT);
42   pinMode(IN1_2, OUTPUT);
43   pinMode(AN1_2, OUTPUT);
44   pinMode(AN2_2, OUTPUT);
45   pinMode(IN2_2, OUTPUT);
46   pinMode(2, OUTPUT);
47
48   roverServo.attach(2);
49   roverServo.write(60);
50
51   Serial.begin(9600);
52   //SerialUSB.begin(9600); // For communication with Jetson Nano
53
54   motorControl(0, 0);
55
56   /* Initialise the sensor */
57   /*(if(!bno.begin())
58   {
59     /* There was a problem detecting the BNO055 ... check your connections */
60     /*Serial.print("Ooops, no BNO055 detected ... Check your wiring or I2C ADDR!");*/
61     while(1);
62   }
63
64   delay(1000);
65
66   bno.setExtCrystalUse(true);*/
67 }
68
69 void motorControl(signed int sL, signed int sR){
70   if(sL > 0 && sR > 0){
71     analogWrite(AN2, sL);
72     analogWrite(AN2_1, sL);
73     analogWrite(AN2_2, sL);
74     digitalWrite(IN2, HIGH);
75     digitalWrite(IN2_1, HIGH);
76     digitalWrite(IN2_2, HIGH);
77     analogWrite(AN1, sR);
78     analogWrite(AN1_1, sR);
79     analogWrite(AN1_2, sR);
80     digitalWrite(IN1, HIGH);
81     digitalWrite(IN1_1, HIGH);
82     digitalWrite(IN1_2, HIGH);
83   }
84
85   else if(sL > 0 && sR < 0){[
86     analogWrite(AN2, sL);
87     analogWrite(AN2_1, sL);
88     analogWrite(AN2_2, sL);
89     digitalWrite(IN2, HIGH);
90     digitalWrite(IN2_1, HIGH);]

```

```

90     digitalWrite(IN2_1, HIGH);
91     digitalWrite(IN2_2, HIGH);
92     sR = sR * -1;
93     analogWrite(AN1, sR);
94     analogWrite(AN1_1, sR);
95     analogWrite(AN1_2, sR);
96     digitalWrite(IN1, LOW);
97     digitalWrite(IN1_1, LOW);
98     digitalWrite(IN1_2, LOW);
99 }
100
101 else if(sL < 0 && sR > 0){
102     sL = sL * -1;
103     analogWrite(AN2, sL);
104     analogWrite(AN2_1, sL);
105     analogWrite(AN2_2, sL);
106     digitalWrite(IN2, LOW);
107     digitalWrite(IN2_1, LOW);
108     digitalWrite(IN2_2, LOW);
109     analogWrite(AN1, sR);
110     analogWrite(AN1_1, sR);
111     analogWrite(AN1_2, sR);
112     digitalWrite(IN1, HIGH);
113     digitalWrite(IN1_1, HIGH);
114     digitalWrite(IN1_2, HIGH);
115 }
116
117 else if(sL < 0 && sR < 0){
118     sL = sL * -1;
119     analogWrite(AN2, sL);
120     analogWrite(AN2_1, sL);
121     analogWrite(AN2_2, sL);
122     digitalWrite(IN2, LOW);
123     digitalWrite(IN2_1, LOW);
124     digitalWrite(IN2_2, LOW);
125     sR = sR * -1;
126     analogWrite(AN1, sR);
127     analogWrite(AN1_1, sR);
128     analogWrite(AN1_2, sR);
129     digitalWrite(IN1, LOW);
130     digitalWrite(IN1_1, LOW);
131     digitalWrite(IN1_2, LOW);
132 }
133
134 else if(sL == 0 && sR == 0){
135     analogWrite(AN2, sL);
136     analogWrite(AN2_1, sL);
137     analogWrite(AN2_2, sL);
138     digitalWrite(IN2, LOW);
139     digitalWrite(IN2_1, LOW);
140     digitalWrite(IN2_2, LOW);
141     analogWrite(AN1, sR);
142     analogWrite(AN1_1, sR);
143     analogWrite(AN1_2, sR);
144     digitalWrite(IN1, LOW);
145     digitalWrite(IN1_1, LOW);
146     digitalWrite(IN1_2, LOW);
147 }
148 }
149
150 void loop() {
151     /* Get a new sensor event */
152     /*sensors_event_t event;
153     bno.getEvent(&event);
154     imu::Vector<3> accel = bno.getVector(Adafruit_BNO055::VECTOR_ACCELEROMETER);
155     float temp = bno.getTemp();*/
156
157     String sensorData = "Temp = " + String(temp) + ", Slope = " + String(event.orientation.y) + " degrees";
158     SerialUSB.println(sensorData);/*
159
160     if (Serial.available() > 0) {
161         char inChar = Serial.read(); //read incoming byte
162
163         /*
164         IN1 and AN1 refer to "motor left" which is actually the motors on the right side of the rover
165         IN2 and AN2 refer to "motor right" which is actually the motors on the left side of the rover
166         according to github code, "motor right" dir pin is HIGH when motors are forward and LOW when motors are reversed
167         according to github code, "motor left" dir pin is LOW when motors are forward and HIGH when motors are reversed
168         in practice I found that, because of an error in the ESC, both dir pins need to be HIGH when forward and LOW when reversed
169         */
170
171         //reverse left
172         if (inChar == '1') {
173             while(speedLeft > -24 && speedRight > -56){[
174                 speedLeft = speedLeft - 1; //this one is - 1 and Right is - 2 because the final value of Right is double Left
175                 speedRight = speedRight - 2;
176                 speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
177                 speedRightActual = speedRight*1.275;
178                 motorControl(speedLeftActual, speedRightActual);

```

```

179     |     delay(10);
180     |
181     speedLeft = -24;
182     speedRight = -56;
183     speedLeftActual = speedLeft*1.275;
184     speedRightActual = speedRight*1.275;
185     motorControl(speedLeftActual, speedRightActual);
186 }
187
188 //reverse straight
189 else if (inChar == '2') {
190     while(speedLeft > -40 && speedRight > -40){
191         speedLeft = speedLeft - 1;
192         speedRight = speedRight - 1;
193         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
194         speedRightActual = speedRight*1.275;
195         motorControl(speedLeftActual, speedRightActual);
196         delay(10);
197     }
198     speedLeft = -40;
199     speedRight = -40;
200     speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
201     speedRightActual = speedRight*1.275;
202     motorControl(speedLeftActual, speedRightActual);
203 }
204
205 //reverse right
206 else if (inChar == '3') {
207     while(speedLeft > -56 && speedRight > -24){
208         speedLeft = speedLeft - 2;
209         speedRight = speedRight - 1;
210         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
211         speedRightActual = speedRight*1.275;
212         motorControl(speedLeftActual, speedRightActual);
213         delay(10);
214     }
215     speedLeft = -56;
216     speedRight = -24;
217     speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
218     speedRightActual = speedRight*1.275;
219     motorControl(speedLeftActual, speedRightActual);
220 }
221
222 //in place turn right
223 else if (inChar == '4') {
224     while(speedLeft < 40 && speedRight > -40){
225         speedLeft = speedLeft + 1;
226         speedRight = speedRight - 1;
227         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
228         speedRightActual = speedRight*1.275;
229         motorControl(speedLeftActual, speedRightActual);
230         delay(10);
231     }
232     speedLeft = 40;
233     speedRight = -40;
234     speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
235     speedRightActual = speedRight*1.275;
236     motorControl(speedLeftActual, speedRightActual);
237 }
238
239 //forward right
240 else if (inChar == '5') {
241     while(speedLeft < 56 && speedRight < 24){
242         speedLeft = speedLeft + 2;
243         speedRight = speedRight + 1;
244         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
245         speedRightActual = speedRight*1.275;
246         motorControl(speedLeftActual, speedRightActual);
247         delay(10);
248     }
249     speedLeft = 56;
250     speedRight = 24;
251     speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
252     speedRightActual = speedRight*1.275;
253     motorControl(speedLeftActual, speedRightActual);
254 }
255
256 //forward straight
257 else if (inChar == '6') [
258     while(speedLeft < 40 && speedRight < 40){
259         speedLeft = speedLeft + 1;
260         speedRight = speedRight + 1;
261         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
262         speedRightActual = speedRight*1.275;
263         motorControl(speedLeftActual, speedRightActual);
264         delay(10);
265     }
266     speedLeft = 40;
267     speedRight = 40;

```

```

268     speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
269     speedRightActual = speedRight*1.275;
270     motorControl(speedLeftActual, speedRightActual);
271 }
272
273 //forward left turn
274 else if (inChar == '7') {
275     while(speedLeft < 24 && speedRight < 56){
276         speedLeft = speedLeft + 1;
277         speedRight = speedRight + 2;
278         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
279         speedRightActual = speedRight*1.275;
280         motorControl(speedLeftActual, speedRightActual);
281         delay(10);
282     }
283     speedLeft = 24;
284     speedRight = 56;
285     speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
286     speedRightActual = speedRight*1.275;
287     motorControl(speedLeftActual, speedRightActual);
288 }
289
290 //in place turn left
291 else if (inChar == '8') {
292     while(speedLeft > -40 && speedRight < 40){
293         speedLeft = speedLeft - 1;
294         speedRight = speedRight + 1;
295         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
296         speedRightActual = speedRight*1.275;
297         motorControl(speedLeftActual, speedRightActual);
298     }
299     speedLeft = -40;
300     speedRight = 40;
301     speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
302     speedRightActual = speedRight*1.275;
303     motorControl(speedLeftActual, speedRightActual);
304 }
305
306 //emergency stop
307 else if (inChar == '9') {
308     //slowed stop for 1 L = -10 and R = -24
309     if(speedLeft == -24 && speedRight == -56){
310         while(speedLeft < 0 && speedRight < 0){
311             speedLeft = speedLeft + 1;
312             speedRight = speedRight + 2;
313             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
314             speedRightActual = speedRight*1.275;
315             motorControl(speedLeftActual, speedRightActual);
316             delay(10);
317         }
318     }
319
320     //slowed stop for 2 L = -24 and R = -30
321     else if(speedLeft == -40 && speedRight == -40){
322         while(speedLeft < 0 && speedRight < 0){
323             speedLeft = speedLeft + 1;
324             speedRight = speedRight + 1;
325             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
326             speedRightActual = speedRight*1.275;
327             motorControl(speedLeftActual, speedRightActual);
328             delay(10);
329         }
330     }
331
332     //stop for 3 L = -30 and R = -10
333     else if(speedLeft == -56 && speedRight == -24){
334         while(speedLeft < 0 && speedRight < 0){
335             speedLeft = speedLeft + 2;
336             speedRight = speedRight + 1;
337             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
338             speedRightActual = speedRight*1.275;
339             motorControl(speedLeftActual, speedRightActual);
340             delay(10);
341         }
342     }
343
344     //stop for 4 L = 30 and R = -30
345     else if(speedLeft == 40 && speedRight == -40){
346         while(speedLeft > 0 && speedRight < 0){
347             speedLeft = speedLeft - 1;
348             speedRight = speedRight + 1;
349             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
350             speedRightActual = speedRight*1.275;
351             motorControl(speedLeftActual, speedRightActual);
352             delay(10);
353         }
354     }
355
356     //stop for 5 L = 30 and R = 10

```

```

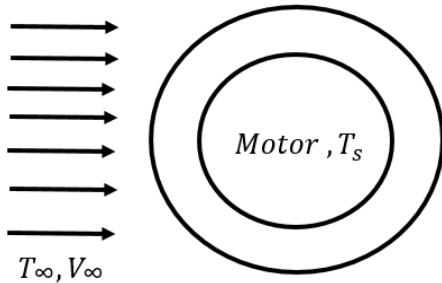
357
358     else if(speedLeft == 56 && speedRight == 24){
359         while(speedLeft > 0 && speedRight > 0){
360             speedLeft = speedLeft - 2;
361             speedRight = speedRight - 1;
362             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
363             speedRightActual = speedRight*1.275;
364             motorControl(speedLeftActual, speedRightActual);
365             delay(10);
366         }
367     }
368
369     //stop for 6 L = 30 and R = 30
370     else if(speedLeft == 40 && speedRight == 40){
371         while(speedLeft > 0 && speedRight > 0){
372             speedLeft = speedLeft - 1;
373             speedRight = speedRight - 1;
374             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
375             speedRightActual = speedRight*1.275;
376             motorControl(speedLeftActual, speedRightActual);
377             delay(10);
378         }
379     }
380
381     //stop for 7 L = 10 and R = 30
382     else if(speedLeft == 24 && speedRight == 56){
383         while(speedLeft > 0 && speedRight > 0){
384             speedLeft = speedLeft - 1;
385             speedRight = speedRight - 2;
386             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
387             speedRightActual = speedRight*1.275;
388             motorControl(speedLeftActual, speedRightActual);
389             delay(10);
390         }
391     }
392
393     //stop for 8 L = -30 R = 30
394     else if(speedLeft == -40 && speedRight == 40){
395         while(speedLeft < 0 && speedRight > 0){
396             speedLeft = speedLeft + 1;
397             speedRight = speedRight - 1;
398             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
399             speedRightActual = speedRight*1.275;
400             motorControl(speedLeftActual, speedRightActual);
401             delay(10);
402         }
403
404         speedLeftActual = 0;
405         speedRightActual = 0;
406         motorControl(speedLeftActual, speedRightActual);
407     }
408
409     //servo left
410     else if(inChar == 'L'){
411         while(angle < 120){
412             if(Serial.available() > 0){
413                 inChar = Serial.read();
414                 if(inChar == '0'){
415                     Serial.println("Break Left");
416                     Serial.println(inChar);
417                     break;
418                 }
419             }
420             roverServo.write(angle);
421             angle++;
422             delay(30);
423             Serial.println("GO LEFT");
424         }
425         roverServo.write(angle);
426     }
427
428     //servo right
429     else if(inChar == 'R'){
430         while(angle > 0){
431             if(Serial.available() > 0){
432                 inChar = Serial.read();
433                 if(inChar == '0'){
434                     Serial.println("Break Right");
435                     break;
436                 }
437             }
438             roverServo.write(angle);
439             angle--;
440             delay(30);
441             Serial.println("GO RIGHT");
442         }
443         roverServo.write(angle);
444     }
445 }

```

The comments talking about the speed do not line up directly with the actual speeds because the speeds are changed based on the needs of the rover.

XIII.D.5.c. Eng. Analysis and Materials Selection Details for SS5-P3 – Heat Transfer Analysis

Tristan Hughes



Assumptions: Steady flow, aluminum housing

Given: $V_{in} = 1.25 \frac{m}{s}$, $q = 2.8 \text{ W}$, $r_o = .06985 \text{ m}$, $r_i = .0624 \text{ m}$, $T_{in} = 26.667^\circ\text{C}$, $L = .103124 \text{ m}$, $h = 26.2 \frac{mW}{m^2 \cdot K}$, $k = 190 \frac{W}{m \cdot K}$

Unknown: T_s

Two types of heat transfer are being considered, convection of the air onto the motor housing and conduction of the motor onto the motor housing. Utilize resistor method to get

$$q = \frac{T_s - T_{in}}{R_{tot}}$$

$$R_{tot} = \frac{\ln\left(\frac{r_o}{r_i}\right)}{2\pi * k_{in} * L} + \frac{1}{h * 2\pi * r_o * L} = .84347$$

$$T_s = 29.0317^\circ\text{C}$$

Final convergence temperature is based off the T_{in} which is naturally subject to change. Code has been written to be able to quickly redo this analysis if needed. At the time this analysis was performed, it was assumed that there would be an aluminum housing that would cover the motor, however this has since been changed to instead only fit around the gearbox of the motor. For this reason, it's likely that this analysis is overly conservative. The heat generation, q , value was taken from the Ampflow company website for the expected torque generated at max capacity.

XIII.D.6. Eng. Analysis Details for SS6 – Power Supply

XIII.D.6.a. Eng. Analysis and Materials Selection Details for SS6-P1 - Battery

Ethernan Smith (EE)

Before being able to select the lithium-ion battery for use as the power supply, and before conducting any analysis on it, we had to construct a duty cycle to determine the needed capacity of our battery. This was only possible after assessing the current draw of the motor under various scenarios, including different terrains and slopes. Patrick conducted the motor analysis to

determine these current values. After Patrick identified the current values during his motor analysis, we constructed a realistic hypothetical hour of operation. This hour involved the rover traversing over various terrains at differing slopes. The hypothetical hour of operation, along with the current values of the motors as the rover traverses different terrains, can be found Figure XII.D.6.a.1. Using the scenario we devised, we obtained the corresponding duty cycle shown in Figure XII.D.6.a.5. This duty cycle considers all electrical components that will be drawing current during our hour of operation. The duty cycle was modeled in Simulink utilizing two switch blocks in series to function as ‘if-then’ blocks. For example, if load 5 added 13.1A during seconds 780-1380, it would be implemented using switch blocks as follows. The first switch block has an initial input of 0, a secondary input of 13.1, and has a time threshold of 780 (seconds). Then it would pass through a second switch block which would utilize the output of the first switch block as its initial input, have a time threshold of 1380 (seconds), and a secondary input of 0. An example of this method can be seen in Figure XII.D.6.a.3. This method ensures that the load adds current only during the time it is meant to contribute, and 0 otherwise. After all loads were accounted for in this manner, a summer was used to add them together, resulting in the waveform shown in Figure XII.D.6.a.5 in appendix (same as one mentioned above). Based on the duty cycle amp hour rating and an understanding of the system’s voltage demands, we chose a 36V 50Ah lithium-ion battery from a company called Enduro as our power supply. Before confirming the use of this battery in the rover, we needed to conduct an analysis to ensure that it could adequately power the rover for our intended full hour of operation. Utilizing Simulink’s software, a model was constructed to perform the necessary analysis. This model implemented multiple output scopes connected to a battery block, with its terminals connected to a controlled current source. The controlled current source’s input was the duty cycle model that was discussed in V.A.6.a. Parameters for the battery block were set using values from the datasheet of the battery being considered for the rover. The waveforms for the state of charge and voltage over the hour of operation were then obtained utilizing the model and can be found in Figure XII.D.6.a.9 and Figure XII.D.6.a.10 in appendix, respectively. The battery is equipped with a built-in battery management system (BMS) that which the datasheet states prevents the battery from operating when it drops below 30V, while also recommending that the battery be disconnected when the voltage gets to 33.6V. With this knowledge, the voltage waveform in Figure XII.D.6.a.10 can be used to verify that the voltage will not get low enough to halt operations during our hour of operation. Additionally, the state of charge will not come even remotely close to dropping below 20%, which is the practical limit that it is advised to avoid discharging below. In conclusion, the 36V 50Ah battery from Enduro can adequately power our rover as intended. Along with the 36V battery, we will also be purchasing a 36V 18A battery charger from Enduro as it is compatible with the battery selected. The datasheet for the battery charger can be found in Figure XII-18 in appendix, with additional facts presented in figure 19. Below are all figures relating to the battery analysis, labeled accordingly. Note that the time the analysis is run is 3600 seconds, which is equivalent to an hour.

Load #	Current (A)	tstart (min)	Tduration (min)	Description	Load torque (Nm)	Motor Torque (Nm)
1	13.1	0	5	2 degree slope, firm grass (C=0.055)	5.55	0.17
2	14.9	5	5	2 degree slope, soft grass (C=0.075)	6.38	0.20
3	21.66	10	2	5 degree slope, medium mud (C=0.09)	9.19	0.29
4	37.34	12	1	10 degree slope, soft mud (C=0.15)	15.25	0.48
5	13.1	13	10	2 degree slope, firm grass (C=0.055)	5.55	0.17
6	21.66	23	5	5 degree slope, medium mud (C=0.09)	9.19	0.29
7	14.9	28	10	2 degree slope, soft grass (C=0.075)	6.38	0.20
8	22.12	38	5	7 degree slope, firm sand (C=0.06)	9.38	0.29
9	36.46	43	1	15 degree slope, firm grass (C=0.055)	14.85	0.46
10	13.1	44	10	2 degree slope, firm grass (C=0.055)	5.55	0.17
11	18.19	54	5.5	4 degree slope, soft grass (C=0.075)	7.84	0.25
12	36.88	59.5	0.5	17 degree slope, smooth dirt (C=0.025)	15.03	0.47
				BNO055 - Adafruit 9-DOF Absolute Orientation IMU Fusion Breakout		
13	0.0123	0	60	DS18B20 - Programmable Resolution 1-Wire Digital Thermometer		
14	0.0015	0	60	DS18B20 - Programmable Resolution 1-Wire Digital Thermometer		
15	0.0015	0	60	DS18B20 - Programmable Resolution 1-Wire Digital Thermometer		
16	0.0015	0	60	DS18B20 - Programmable Resolution 1-Wire Digital Thermometer		
17	0.0015	0	60	DS18B20 - Programmable Resolution 1-Wire Digital Thermometer		
18	0.0022	0	60	RCWL-1601 - Ultrasonic Distance Sensor		
19	0.0022	0	60	RCWL-1601 - Ultrasonic Distance Sensor		
20	0.034	0	60	GPS Breakout		
21	0.5	0	60	Webcam		
22	0.8	0	60	Arduino Due		
23	2.5	0	60	Jetson Nano		

Figure XII.D.6.a.1 - Hypothetical hour of operation with descriptions

The above scenario was used as our duty cycle. For the model in Simulink, these times were converted from minutes to seconds.

Load #	A*min	Ah
1	65.5	1.0916667
2	74.5	1.2416667
3	43.32	0.722
4	37.34	0.6223333
5	131	2.1833333
6	149	2.4833333
7	36.46	0.6076667
8	131	2.1833333
9	36.46	0.6076667
10	131	2.1833333
11	100.045	1.6674167
12	18.44	0.3073333
13-21	33.402	0.5567
22	48	0.8
23	150	2.5
Total		19.757783

Figure XII.D.6.a.2 – Duty Cycle Values

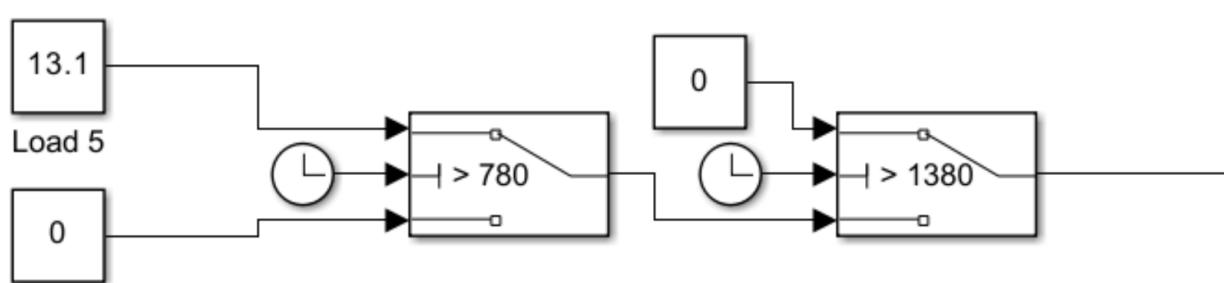


Figure XII.D.6.a.3– Example of Switches as If-then block

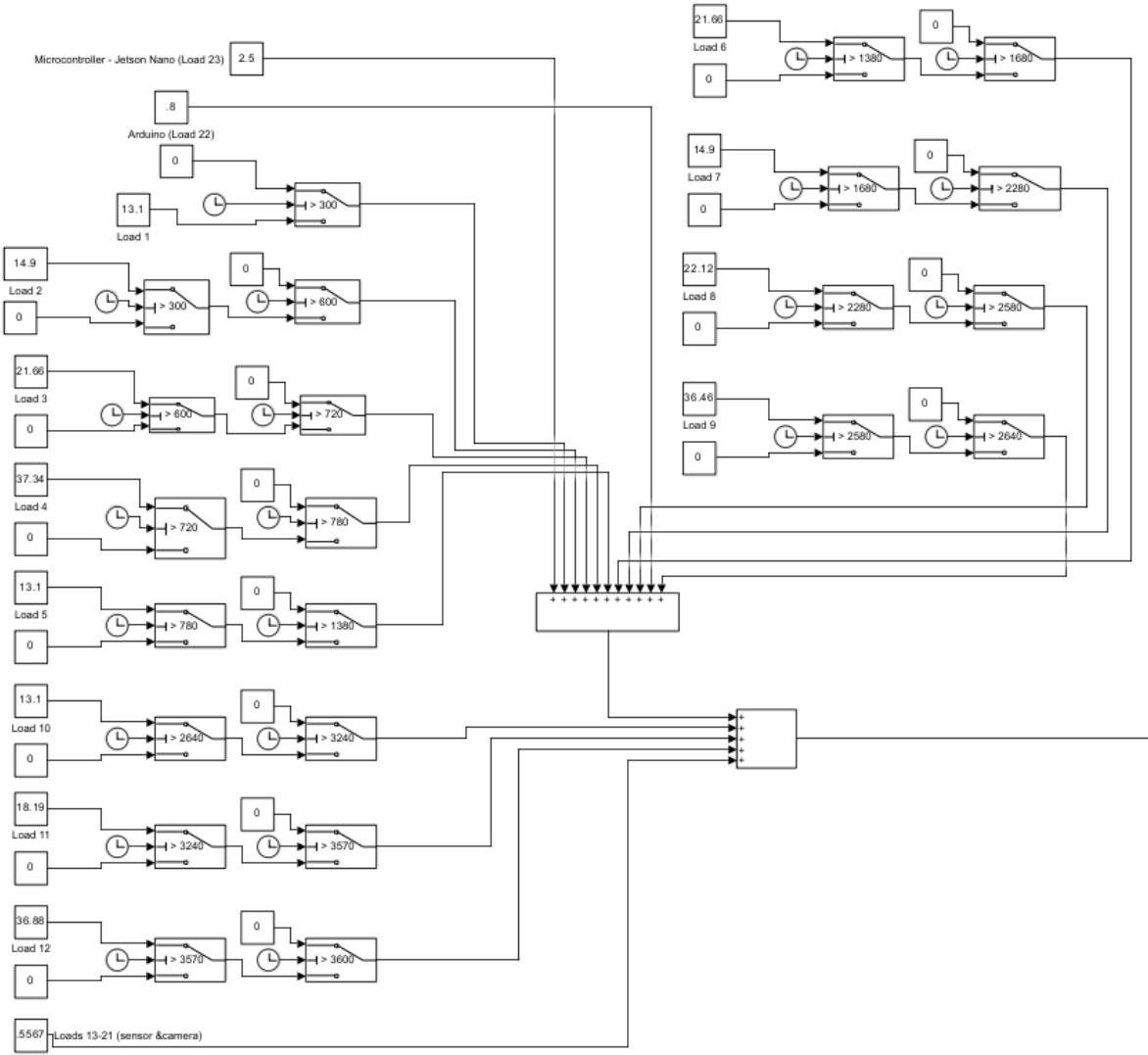


Figure XII.D.6.a.4– Duty Cycle modeled in Simulink

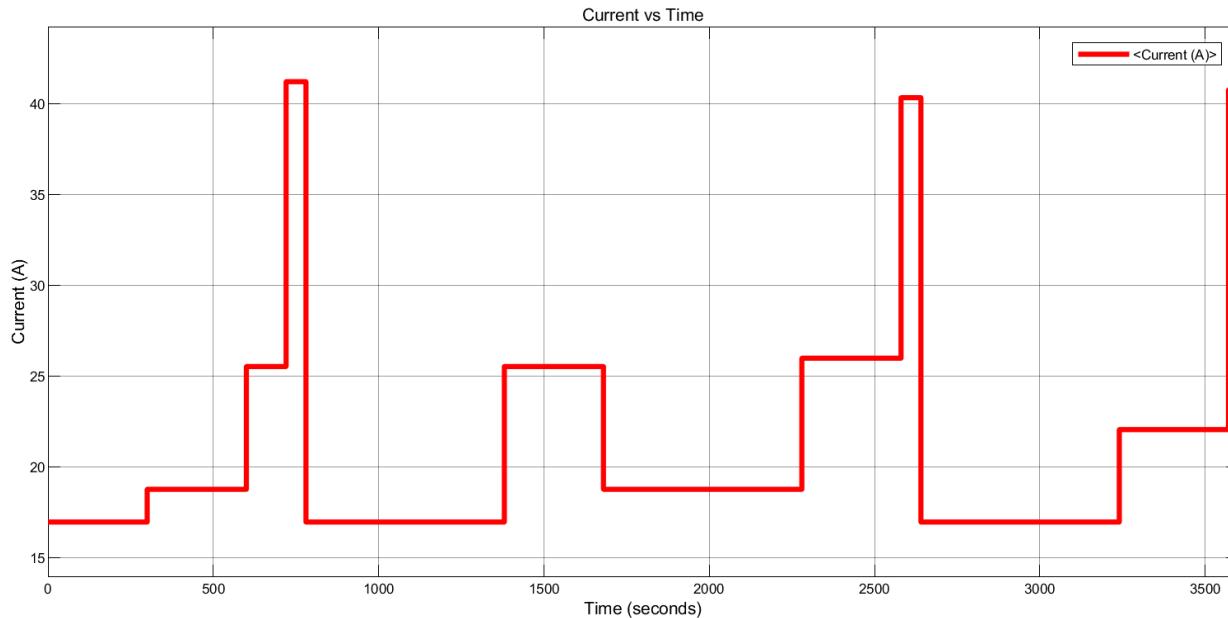


Figure XII.D.6.a.5 – Duty Cycle

Criterion	Weights	Battery				
		PowerTex 36V 50Ah	Enduro 36V 50Ah	Enjoybot 36V 100AH	Dakota 36V 60Ah	Cao MM 36V 20 Ah
Cost	44.09%	0	0	-3	-3	1
Durability	31.47%	0	0	0	0	0
Mass	12.60%	0	1	0	-1	0
Maintenance	8.12%	0	0	0	0	-3
Dimension	3.73%	0	0	-1	0	1
Weighted Total	100.00%	0	0.1260	-1.3600	-1.4487	0.2346
Additional Notes		No data sheet		Larger than we need		Would need at least 3 in parallel - thus BMS would be less straightforward

Figure XII.D.6.a.6 – Decision Matrix for Battery Selection

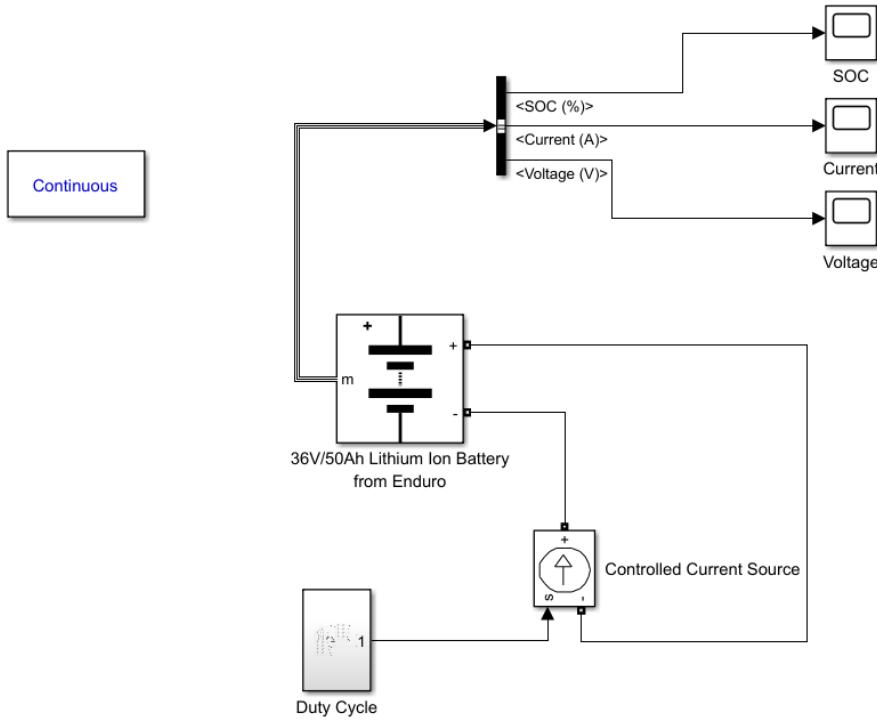


Figure XII.D.6.a.7 – Battery Model in Simulink

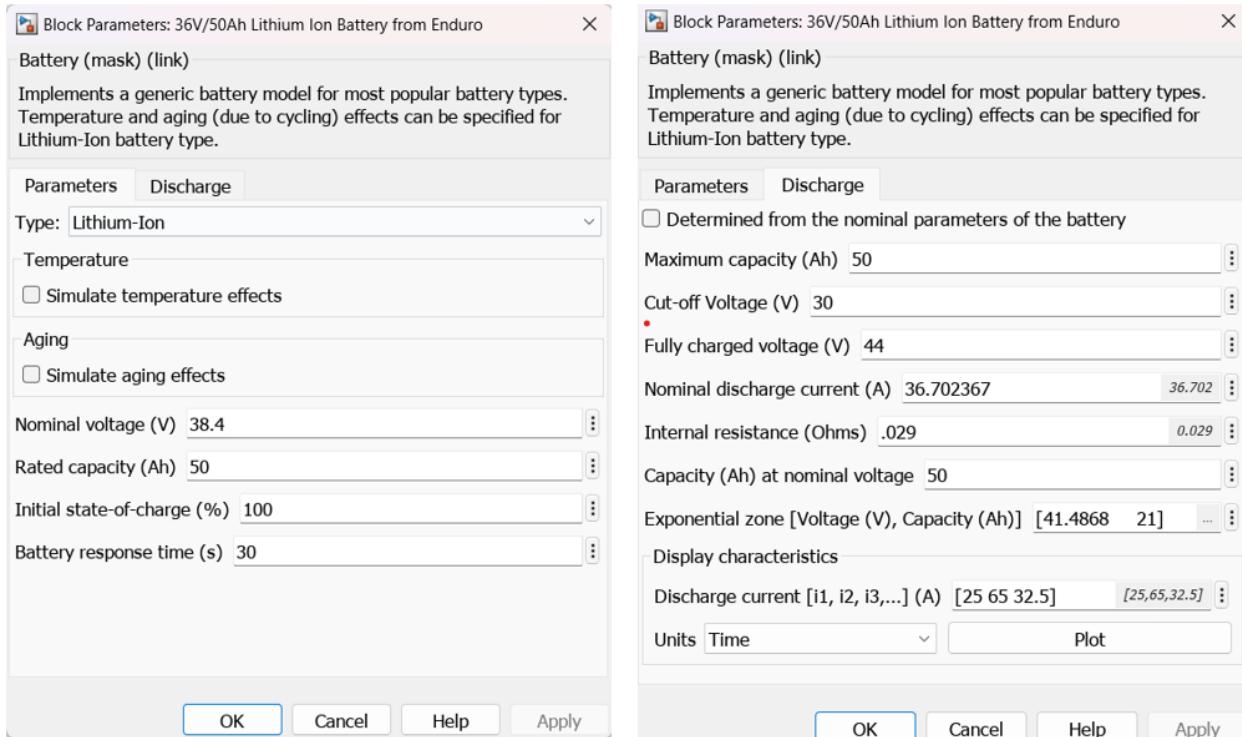


Figure XII.D.6.a.8 – Battery Parameters for battery block, based on datasheet for our battery

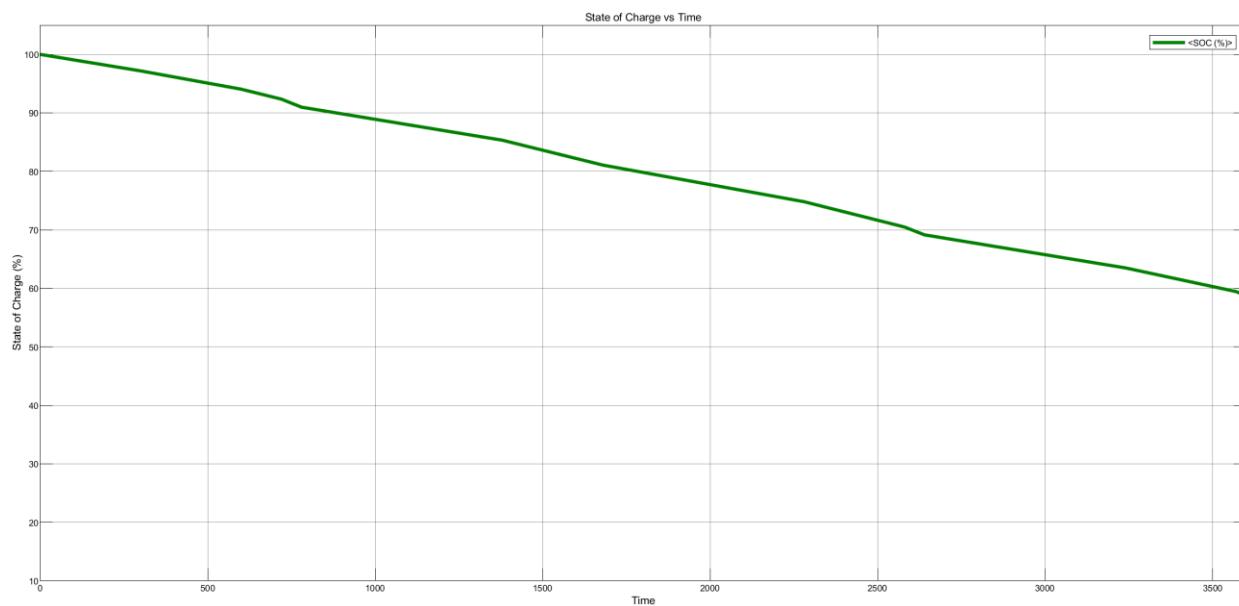


Figure XII.D.6.a.9 – State of Charge vs Time

The state of charge does not drop below 55%, staying well above the 20% practical limit.

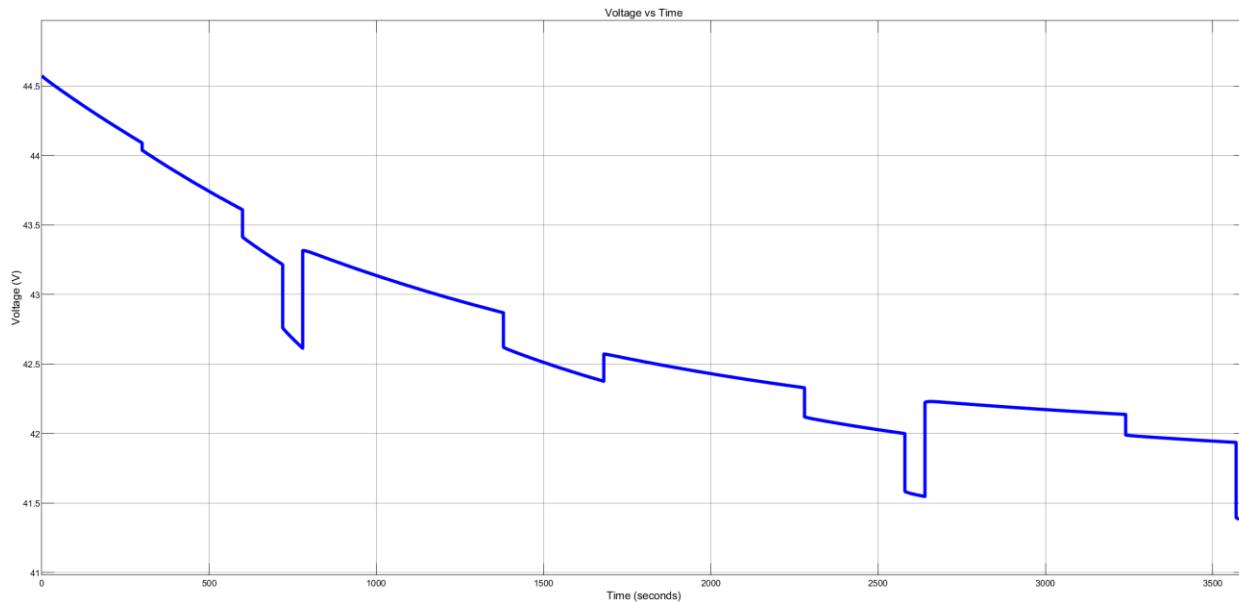


Figure XII.D.6.a.10 – Voltage vs Time

The voltage does not drop below the recommended cutoff voltage listed on the datasheet, thus the BMS will not step in and cut off the battery to halt operations.

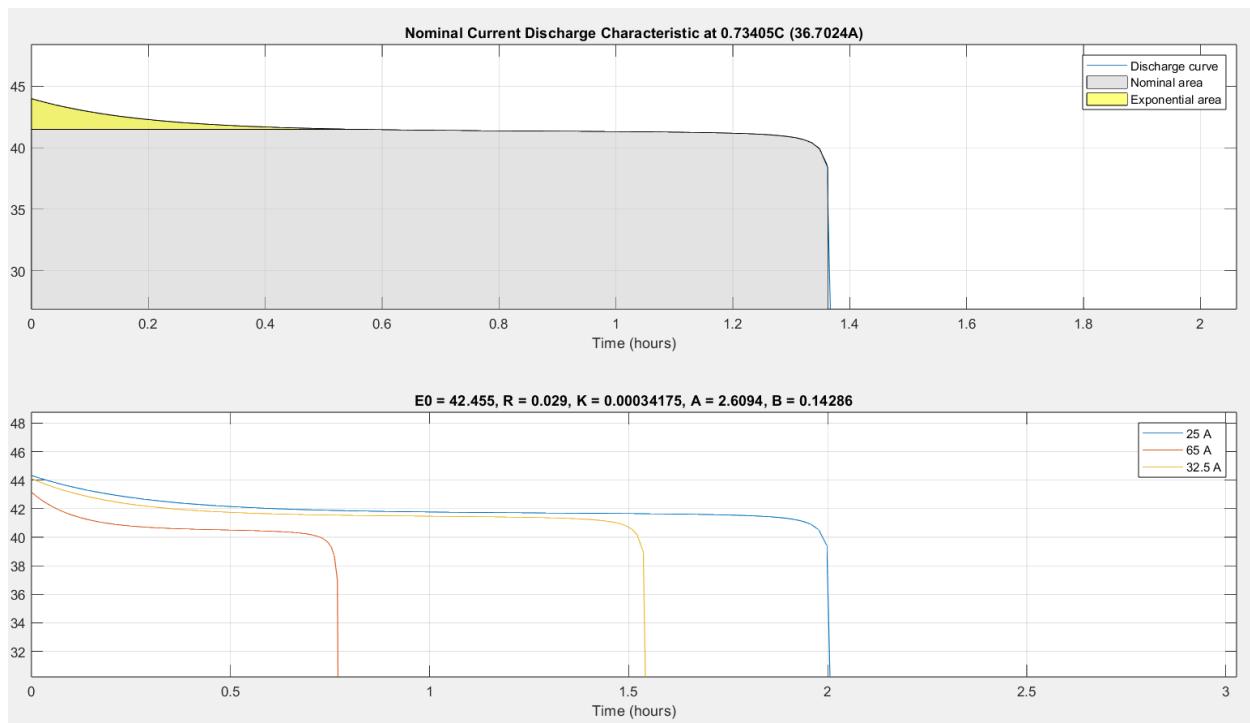


Figure XII.D.6.a.11 – Battery Discharge Characteristic

For the lithium-ion battery type, the model uses these equations.

- Discharge Model ($i^* > 0$)

$$f_1(it, i^*, i) = E_0 - K \cdot \frac{Q}{Q - it} \cdot i^* - K \cdot \frac{Q}{Q - it} \cdot it + A \cdot \exp(-B \cdot it)$$

- Charge Model ($i^* < 0$)

$$f_2(it, i^*, i) = E_0 - K \cdot \frac{Q}{it + 0.1 \cdot Q} \cdot i^* - K \cdot \frac{Q}{Q - it} \cdot it + A \cdot \exp(-B \cdot it)$$

For the nickel-cadmium and nickel-metal-hydride battery types, the model uses these equations.

- Discharge Model ($i^* > 0$)

$$f_1(it, i^*, i, Exp) = E_0 - K \cdot \frac{Q}{Q - it} \cdot i^* - K \cdot \frac{Q}{Q - it} \cdot it + \text{Laplace}^{-1}\left(\frac{\text{Exp}(s)}{\text{Sel}(s)} \cdot 0\right)$$

- Charge Model ($i^* < 0$)

$$f_2(it, i^*, i, Exp) = E_0 - K \cdot \frac{Q}{|it| + 0.1 \cdot Q} \cdot i^* - K \cdot \frac{Q}{Q - it} \cdot it + \text{Laplace}^{-1}\left(\frac{\text{Exp}(s)}{\text{Sel}(s)} \cdot \frac{1}{s}\right).$$

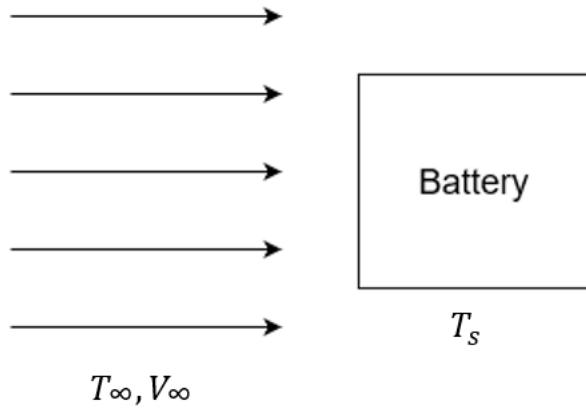
In the equations:

- E_{Batt} is the nonlinear voltage, in V.
- E_0 is the constant voltage, in V.
- $\text{Exp}(s)$ is the exponential zone dynamics, in V.
- $\text{Sel}(s)$ represents the battery mode. $\text{Sel}(s) = 0$ during battery discharge, $\text{Sel}(s) = 1$ during battery charging.
- K is the polarization constant, in V/Ah, or polarization resistance, in Ohms.
- i^* is the low-frequency current dynamics, in A.
- i is the battery current, in A.
- it is the extracted capacity, in Ah.
- Q is the maximum battery capacity, in Ah.
- A is the exponential voltage, in V.
- B is the exponential capacity, in Ah^{-1} .

Figure XII.D.6.a.12 – Equations used in Simulink Model

XIII.D.6.b. Eng. Analysis and Materials Selection Details for SS6-P2 – Thermal Management

Tristan Hughes



Assumptions: Steady flow over the battery.

Known: $V_{in} = 1.25 \text{ m/s}$, $T_s = 26 \text{ C}$, $T_{in} = 21 \text{ C}$, $L = .3302 \text{ m}$, $W = .2159 \text{ m}$

Properties of air at 21 C: $\nu = 30.84 * 10^{-6} \text{ m}^2/\text{s}$, $k = 36.4 * 10^{-3} \text{ W/m}$, $\text{Pr} = .687$

Find: q of the battery

Solution:

$$Re = V_{in} * L / \nu = 13348,$$

$$Nu_L = .664 * Re^{.5} Pr^{.33} = 67.7893, h = \frac{Nu_L * k}{L} = 7.28 ,$$

$$q' = h * L(T_{in} - T_s) = -13.7071 \text{ W/m},$$

$$q = q' * W = -2.9594 \text{ W}$$

This works under the assumption that the flow over the battery will be due to the rover moving at its speed of 1.25 m/s, however in real scenarios wind effects will likely cause the flow to be moving at a higher velocity, so this calculation is likely an underestimate.

```

% The following code is written for convection HT on the battery
% This is written for Capstone Team 57 for Fall 2023-Spring 2024
% by Tristan Hughes

%%%%% Variables

% Input here
Tin_F = 70
Ts_F = 80

Tin_C = .5555 * (Tin_F - 32)% in C, this is temperature of the air
Ts_C = .5555 * (Ts_F - 32) % in C, surface temperature of the battery
Uin = 1.25 % in m/s, rate that air is hitting object
rho = 101.25 % atmospheric pressure in kpa
L = .3302 % in m, length of battery
W = .2159 % in m, width of battery

%%%% Air Properties
v = 30.84 * 10^-6 % m^2/s
k = 36.4 * 10^-3 % W/m
Pr = .687 % dimensionless

%%%%% Equations

% q_prime = h*L*(Tin-Ts)
% q_prime is heat tranfer per length, in W/m

Re_L = (Uin*L)/v

Nu_L = .664 * (Re_L^.5) * (Pr^.333)

h = (Nu_L*k)/L

q_prime = h * L * (Tin_C - Ts_C)

q = q_prime * W % Gives watts

% so you remove 3 joules/s of heat from the battery

```

Tin_F = 70	Ts_F = 80
Tin_C = 21.1090	
Ts_C = 26.6640	
Uin = 1.2500	
rho = 101.2500	
L = 0.3302	
W = 0.2159	
v = 3.0840e-05	
k = 0.0364	
Pr = 0.6870	
Re_L = 1.3384e+04	
Nu_L = 67.7893	
h = 7.4728	
q_prime = -13.7071	
q = -2.9594	

XIII.D.7. Eng. Analysis Details for SS7 – Power Distribution

- Voltage Converters
- Wire Sizing
- Bend Radius of Wire
- Voltage Drop

XIII.D.7.a. Eng. Analysis and Materials Selection Details for SS7-P7 – Voltage Converters

Ethernan Smith (EE)

The ratings of the electrical components were used to construct the following flowchart, with voltage converters being inserted as necessary to ensure that components were receiving power at the proper ratings.

Componet	Max Current Draw	Voltage
BNO055 - Adafruit 9-DOF Absolute Orientation IMU Fusion Breakout	12.3 mA	1.7 to 3.6
DS18B20 - Programmable Resolution 1-Wire Digital Thermometer	1.5 mA	3 or 5 V
RCWL-1601 - Ultrasonic Distance Sensor	2.2 mA	3 or 5 V
GPS Breakout	34 mA	3 or 5 V
Webcam	N/A	N/A
Arduino Due	130 mA	7-12 V
Jetson Nano	2.5 A	5 V
Servo Motor for Camera	>1A	4.8-6V

Figure XII.D.7.a.1 – Ratings of electrical Components

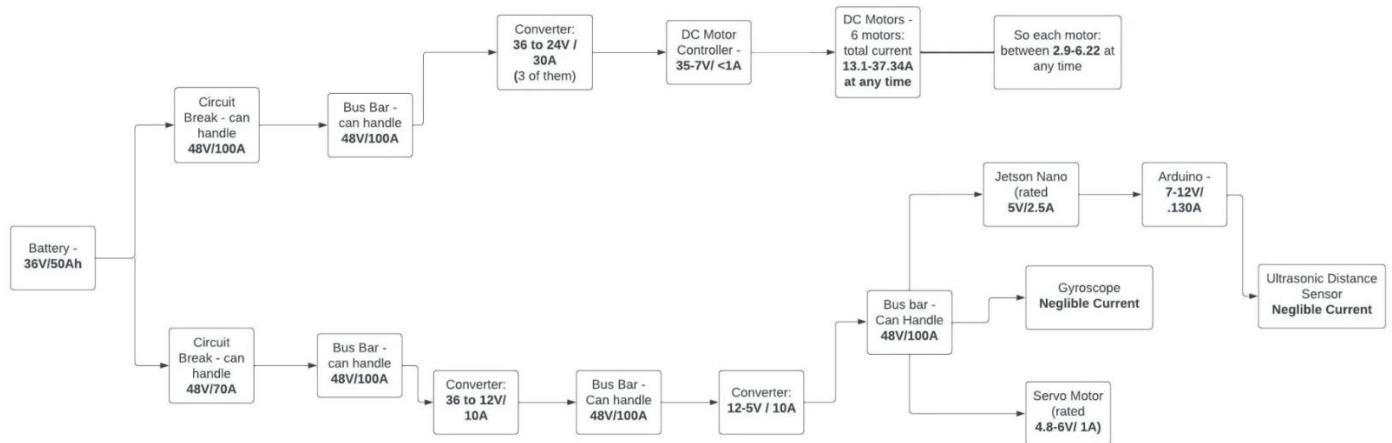


Figure XII.D.7.b.1 Voltage Converter Diagram with Ratings

XIII.D.7.b. Eng. Analysis and Materials Selection Details for SS7-P2 - Wires

Patrick Maloney and Ethernan Smith

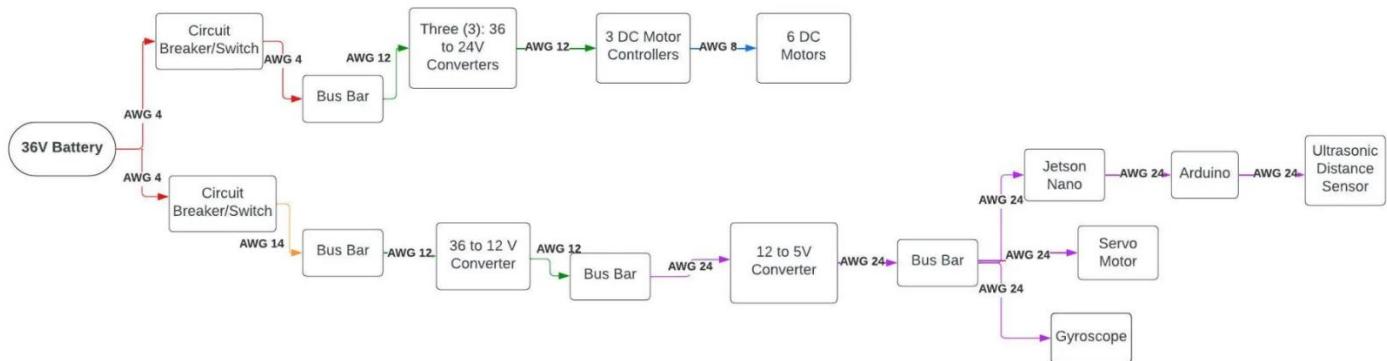


Figure XII.D.7.b.1 – Flowchart of Electrical components with wire sizes

Because we were planning on running the wires to the motors through the legs, we needed to make sure that the physical wire is able to bend through the legs without breaking and shorting to the frame. This is still useful if future teams want to run the wires right alongside the legs and the frame. Using the diameter of 8 gauge wire from the following AWG chart and the fact that the inner diameter of the leg is 1 inch = 25.4 mm we can determine if running the wires through or right alongside the legs is even possible.

AWG gauge	Conductor Diameter Inches	Conductor Diameter mm	Conductor cross section in mm ²	Ohms per 1000 ft.	Ohms per km	Maximum amps for chassis wiring	Maximum amps for power transmission	Maximum frequency for 100% skin depth for solid conductor copper	Breaking force Soft Annealed Cu 37000 PSI
0000	0.46	11.684	107	0.049	0.16072	380	302	125 Hz	6120 lbs
000	0.4096	10.40384	84.9	0.0618	0.202704	328	239	160 Hz	4860 lbs
00	0.3648	9.26592	67.4	0.0779	0.255512	283	190	200 Hz	3860 lbs
0	0.3249	8.25246	53.5	0.0983	0.322424	245	150	250 Hz	3060 lbs
1	0.2893	7.34822	42.4	0.1239	0.406392	211	119	325 Hz	2430 lbs
2	0.2576	6.54304	33.6	0.1563	0.512664	181	94	410 Hz	1930 lbs
3	0.2294	5.82676	26.7	0.197	0.64616	158	75	500 Hz	1530 lbs
4	0.2043	5.18922	21.1	0.2485	0.81508	135	60	650 Hz	1210 lbs
5	0.1819	4.62026	16.8	0.3133	1.027624	118	47	810 Hz	960 lbs
6	0.162	4.1148	13.3	0.3951	1.295928	101	37	1100 Hz	760 lbs
7	0.1443	3.66522	10.6	0.4982	1.634096	89	30	1300 Hz	605 lbs
8	0.1285	3.2639	8.37	0.6282	2.060496	73	24	1650 Hz	480 lbs
9	0.1144	2.90576	6.63	0.7921	2.598088	64	19	2050 Hz	380 lbs
10	0.1019	2.58826	5.26	0.9989	3.276392	55	15	2600 Hz	314 lbs
11	0.0907	2.30378	4.17	1.26	4.1328	47	12	3200 Hz	249 lbs
12	0.0808	2.05232	3.31	1.588	5.20864	41	9.3	4150 Hz	197 lbs
13	0.072	1.8288	2.63	2.003	6.56984	35	7.4	5300 Hz	150 lbs
14	0.0641	1.62814	2.08	2.525	8.282	32	5.9	6700 Hz	119 lbs
15	0.0571	1.45034	1.65	3.184	10.44352	28	4.7	8250 Hz	94 lbs
16	0.0508	1.29032	1.31	4.016	13.17248	22	3.7	11 kHz	75 lbs
17	0.0453	1.15062	1.04	5.064	16.60992	19	2.9	13 kHz	59 lbs
18	0.0403	1.02362	0.823	6.385	20.9428	16	2.3	17 kHz	47 lbs
19	0.0359	0.91186	0.653	8.051	26.40728	14	1.8	21 kHz	37 lbs
20	0.032	0.8128	0.519	10.15	33.292	11	1.5	27 kHz	29 lbs
21	0.0285	0.7239	0.412	12.8	41.984	9	1.2	33 kHz	23 lbs
22	0.0253	0.64516	0.327	16.14	52.9392	7	0.92	42 kHz	18 lbs
23	0.0226	0.57404	0.259	20.36	66.7808	4.7	0.729	53 kHz	14.5 lbs
24	0.0201	0.51054	0.205	25.67	84.1976	3.5	0.577	68 kHz	11.5 lbs

As seen above, the wire diameter of 8 gauge copper wire is 3.2639 mm.

The bending radius of wires much larger than this have a bending radius of 8 times the diameter according to 2014 NEC 300.34 and 2014 NEC 300.24, so it was decided to use 5 times the diameter for the bending radius of 9 gauge copper wire.

This gives a bending radius of $r = 3.2639(5) = 16.32 \text{ mm}$

The maximum bend radius through the legs can be calculated by taking the inner diameter of the legs and subtracting the diameter of the copper wire.

$$r_{max} = 25.4 - 3.2639 = 22.14 \text{ mm}$$

Clearly it is going to be possible to bend the wires through the legs, however, since a future team would be running, at times, six wires through the same section of the frame, they would need to verify with testing that this is possible.

Voltage Drop

It is desirable to know the voltage drop over the longest run of each wire size to ensure that we are going to be providing enough voltage to our components.

This calculation can be done using Ohm's Law $V = IR$. Where V is the voltage drop, I is the current through the wire, and R is the resistance of the wires based on the length and the ohms per km table from the AWG Table above.

Below are the longest lengths of wire we expect for each wire size as well as justification for the lengths:

Battery to Bus Bar (4 AWG): 0.4 meters at maximum (a quarter of the length of the rover is 0.31 m plus a little extra)

DC-DC Converter to motor controllers (12 AWG): 0.3048 meters (1 foot) (each strut is one foot long so this is if the converter and the controller were on opposite sides of the electronics housing area)

Motor controllers to motors (8 AWG): 1.143 meters (45 inches) (leg length is 23 inches, frame is 12 inches tall, add an extra 10 inches for slack and also for going from the end of the leg to the back of the motor)

Electronic Components (24 AWG): 1.0772 (camera and servo motor) (length from half of the rover to the end and then up to the top of the frame is 0.62 meters plus 12 inches and then add an extra 6 inches to be safe.)

These are all overly conservative estimates compared to the actual lengths.

Battery to Bus Bar:

$$V = IR = 118A \left(0.81508 \frac{\text{ohms}}{\text{km}} \right) \left(\frac{1 \text{ km}}{1000 \text{ m}} \right) (0.4 \text{ m}) = 0.0038 \text{ V}$$

This is a 0.11% drop in voltage when compared to the nominal voltage of the battery (36V). The current used is assuming all six motors are running at rated speed and current.

DC-DC Converter to Motor Controllers:

$$V = IR = 34 A \left(5.20864 \frac{\text{ohms}}{\text{km}} \right) \left(\frac{1 \text{ km}}{1000 \text{ m}} \right) (0.3048 \text{ m}) = 0.054 \text{ V}$$

This is 0.22% drop in voltage when compared to the voltage input to the motor controllers (24V). The current used is assuming two motors are running at rated speed and current.

Motor Controllers to Motors:

$$V = IR = 17 A \left(2.060496 \frac{\text{ohms}}{\text{km}} \right) \left(\frac{1 \text{ km}}{1000 \text{ m}} \right) (1.143 \text{ m}) = 0.040 \text{ V}$$

This is a 0.33% drop in voltage when compared to the average voltage input to the motors (12 V). The current used is assuming one motor at rated current.

Electronic Components:

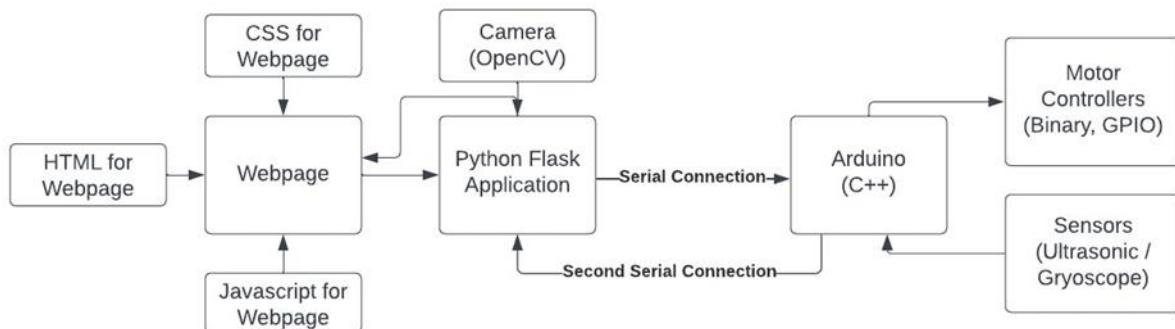
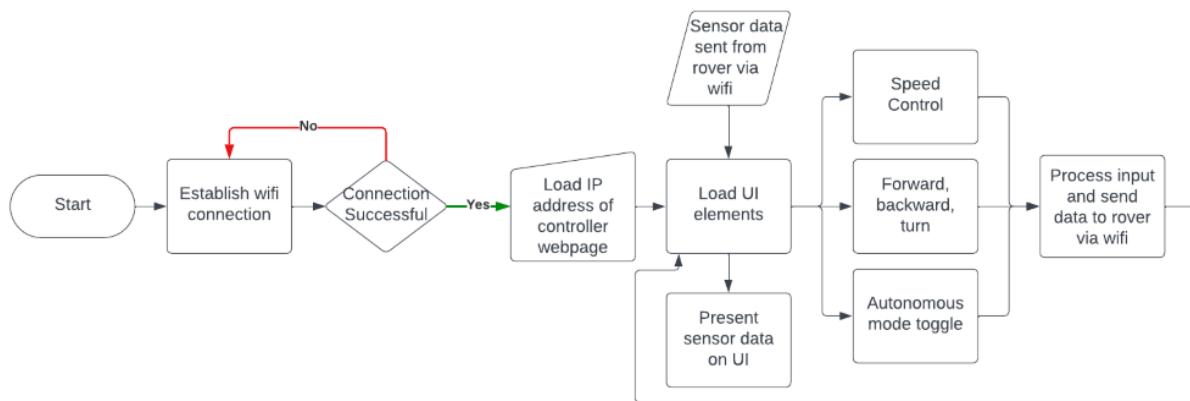
$$V = IR = 2.31 A \left(84.1976 \frac{\text{ohms}}{\text{km}} \right) \left(\frac{1 \text{ km}}{1000 \text{ m}} \right) (1.0772 \text{ m}) = 0.210 \text{ V}$$

This is a 4.36% voltage drop if using 4.8 V for servo motor. This current is using the worst-case stall current which would likely never occur as the only load is the camera.

XIII.D.8. Eng. Analysis Details for SS8 - Controller

XIII.D.8.a. Eng. Analysis and Materials Selection Details for SS8-P1 - Controller

Creighton Cathey (EEC)



XIII.D.9. Eng. Analysis Details for SS9 – Object Detection and Reaction

XIII.D.9.a. Eng. Analysis and Materials Selection Details for SS9-P1 - Servomotor

Patrick Maloney (EE)

To determine if the servomotor selected could move the camera selected, the moment of inertia of a rectangular prism approximating the camera was found. If we take a rectangular prism that is rotating about the axis parallel with its height, then the moment of inertia can be found using the following equation:

$$J_h = \frac{1}{12}m(w^2 + d^2)$$

Where J_h is the moment of inertia, m is the mass of the camera, w is the width of the camera, and d is the length of the camera. The specs of the camera that we chose indicate that the mass is 4.3 oz, the width is 2 inches, and the height is 3 inches. These values can be converted into SI units as seen below

$$\begin{aligned} m &= 4.3 \text{ oz} = 0.122 \text{ kg} \\ w &= 2 \text{ in} = 0.0508 \text{ m} \\ d &= 3 \text{ in} = 0.0762 \text{ m} \end{aligned}$$

The moment of inertia can then be found:

$$J_h = \frac{1}{12}(0.122)(0.0508^2 + 0.0762^2) = 0.0000853 \text{ kg} \cdot \text{m}^2$$

The top speed of the servo that we selected can be derived from the fact that its no-load speed is 0.18 s to go 60 degrees ($\frac{\pi}{3} \text{ rad}$) $\rightarrow 5.82 \text{ rad/s}$. If we wanted to accelerate to full speed in 0.01 s, we would get an acceleration of $581.8 \frac{\text{rad}}{\text{s}^2}$. Therefore, the torque needed to be overcome by the servo is shown below:

$$T_{em} = J_h \frac{d\omega}{dt} = 0.0000853(581.8) = 0.0496 \text{ Nm}$$

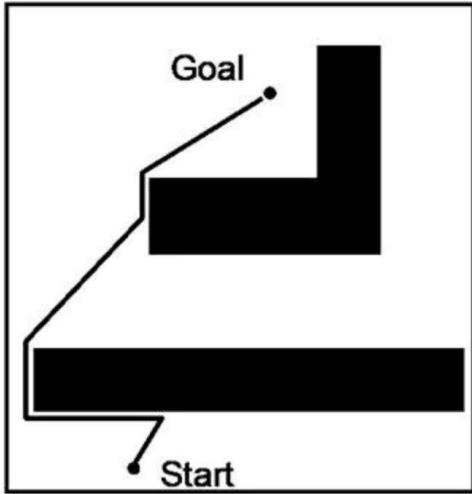
The camera itself provides no opposing torque during steady-state operation of the servomotor so the torque shown above only needs to be overcome for a short amount of time. This is assuming the moment of inertia of the servo to be negligible.

The stall torque of the selected servomotor is 2.01 Nm so the servo will be able to accelerate the camera.

XIII.D.9.b.

Eng. Analysis and Materials Selection Details for SS9 - Algorithm

Creighton Cathey



```

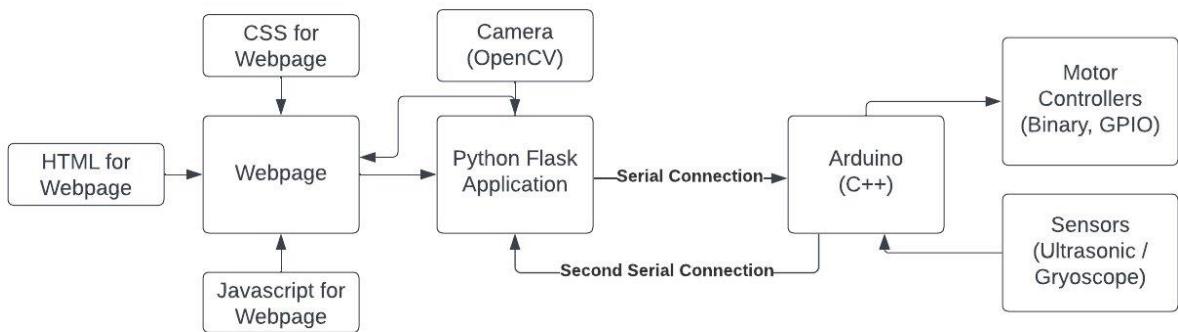
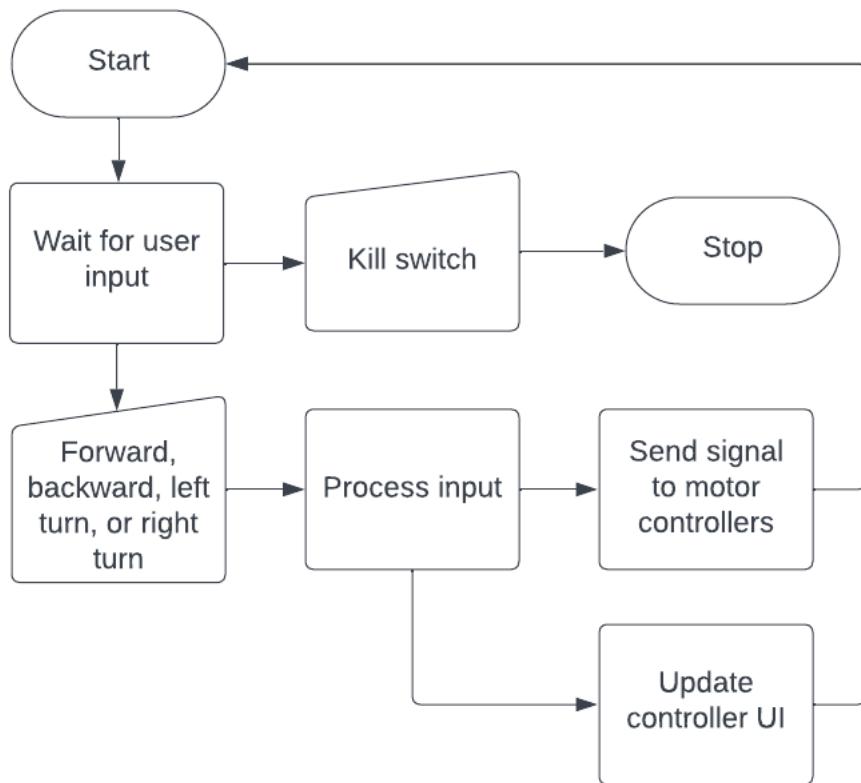
192 void handleAutonomousMode() {
193     float distanceLeft = measureDistance(TRIG_PIN1, ECHO_PIN1); // Left sensor
194     float distanceRight = measureDistance(TRIG_PIN2, ECHO_PIN2); // Right sensor
195     int autonomousSpeed = 51; // Set a 20% speed for autonomous movement (20% of max 255)
196
197     // Conditions for obstacle avoidance
198     if (distanceLeft < 30.0 && distanceRight < 30.0) {
199         motorControl(-autonomousSpeed, -autonomousSpeed); // Reverse a bit
200         delay(1000); // Reverse for 1 second
201         if (distanceLeft > distanceRight) {
202             motorControl(autonomousSpeed, -autonomousSpeed); // Turn left
203         } else {
204             motorControl(-autonomousSpeed, autonomousSpeed); // Turn right
205         }
206     } else if (distanceLeft < 30.0) {
207         motorControl(autonomousSpeed / 2, autonomousSpeed); // Turn slightly right
208     } else if (distanceRight < 30.0) {
209         motorControl(autonomousSpeed, autonomousSpeed / 2); // Turn slightly left
210     } else {
211         motorControl(autonomousSpeed, autonomousSpeed); // Move forward
212     }
213 }
214

```

XIII.D.10. Eng. Analysis Details for SS10 - Communication

XIII.D.10.a. Eng. Analysis and Materials Selection Details for SS10

Creighton Cathey (EEC)



Analysis – Link Budget

5 GHz Band

Tx Power – 23 dBm (datasheet)
Antenna Gain – 9 dBi (Wi-fi antenna gain)
Cable Loss – 0.8 dB (2 feet of LMR100 wire)
Path Loss – 12 dB (calculation below)

Path loss equation (free space)

$$20\log_{10}\left(4\pi \left(\frac{(0.1km)}{(5GHz)}\right)\right) = 11.995db$$

Link budget equation

$$23dbm + 9dbi - 0.8db - 12db = 19.2db$$

Link budget without fade margin = 19.2 dB
Link budget with fade margin = 9.2 dB
Link budget = 9.2 - 19.2 dB

2.4 GHz Band

Tx Power – 20 dBm (datasheet)
Antenna Gain – 9 dBi (Wi-fi antenna gain)
Cable Loss – 0.8 dB (2 feet of LMR100 wire)
Path Loss – 5.6 dB (calculation below)

Path loss equation (free space)

$$20\log_{10}\left(4\pi \left(\frac{(0.1km)}{(2.4GHz)}\right)\right) = 5.62db$$

Link budget equation

$$20dbm + 9dbi - 0.8db - 5.6db = 22.6db$$

Link budget without fade margin = 22.6 dB
Link budget with fade margin = 12.6 dB
Link budget = 12.6 - 22.6 dB

XIII.E.Manufacturing & Assembly Supplement

XIII.E.1. Manufacturing Processes

Team 57 made use of a variety of manufacturing techniques during the spring semester, including but not limited to: Mill, Lathe, Waterjet, MIG Welding (steel & aluminum), 3D printing, and soldering.

Table XIII-11: Manufacturing Processes for Parts of Sub-System SS1

Part #	Manufacturing Process
SS1-P1	Cut to size with reciprocating saw and angle grinder. Painted.
SS1-P2	3D printed.
SS1-P3	Cut to size with bandsaw. Drilled circular cutouts with end mill. Smoothed corners and edges on belt grinder.
SS1-P4	Cut to size with bandsaw. Lathe used to adjust OD and cut grooves for retaining rings.
SS1-P5	3D printed.
SS1-P6	3D printed.
SS1-P7	3D printed.

Table XIII-12: Manufacturing Processes for Parts of Sub-System SS2

Part #	Manufacturing Process
SS2-P1	Cut to size with bandsaw. Drilled bolt holes with end mill. Threaded by hand with threading tool. Rough edges grinded on belt grinder.
SS2-P2	Purchased.
SS2-P3	Cut to size with bandsaw.
SS2-P4	Purchased.
SS2-P5	Purchased.
SS2-P6	Cut with waterjet. Rough edges grinded on belt grinder.
SS2-P7	Purchased.
SS2-P8	Purchased.
SS2-P9	Lathe used to reduce OD and drill hole for motor. Hex head cut on end mill. Hitch pin hole drilled on end mill. Key slot drilled and milled with end mill.
SS2-P10	Purchased.
SS2-P11	Purchased.
SS2-P12	Purchased.
SS2-P13	Purchased. Exposed ends interfering with wheel rotation cut off with hack saw and bolt cutter.

SS2-P14	Purchased.
SS2-P15	Purchased. Disassembled. Inner knobs cut off with hacksaw. Holes drilled with power drill.
SS2-P16	Purchased.
SS2-P17	Cut with waterjet. Rough edges grinded on belt grinder.
SS2-P18	Purchased.

Table XIII-13: Manufacturing Processes for Parts of Sub-System SS3

Part #	Manufacturing Process
SS3-P1	Struts and bearing cut to size. Circular cutouts cut on end of struts with end mill. Bolt hole drilled in struts with end mill. Struts MIG welded to bearing. Ends grinded down on belt grinder for better fit.
SS3-P2	Struts and bearing cut to size. Circular cutouts cut on end of struts with end mill. Struts MIG welded to bearing.
SS3-P3	Cut on waterjet. Rough edges grinded on belt grinder.
SS3-P4	Purchased.
SS3-P5	Purchased.
SS3-P6	Cut to size with bandsaw. OD and ID adjusted on lathe. Hole for set screw drilled with end mill.
SS3-P7	Cut to size with bandsaw. Holes drilled on end mill.
SS3-P8	Purchased.
SS3-P9	Purchased.

Table XIII-14: Manufacturing Processes for Parts of Sub-System SS4

Part #	Manufacturing Process
SS4-P1	Cut on waterjet. Rough edges grinded on belt grinder.
SS4-P2	Cut with bandsaw.
SS4-P3	Cut on waterjet. Rough edges grinded on belt grinder.
SS4-P4	Purchased. Length cut down with hack saw.
SS4-P5	Purchased.
SS4-P6	Purchased.

Table XIII-15: Manufacturing Processes for Parts of Sub-System SS5

Part #	Manufacturing Process
SS5-P1	Purchased
SS5-P2	Purchased

Table XIII-16: Manufacturing Processes for Parts of Sub-System SS6

Part #	Manufacturing Process
SS6-P1	Purchased

Table XIII-17: Manufacturing Processes for Parts of Sub-System SS7

Part #	Manufacturing Process
SS7-P1	Purchased
SS7-P2	Purchased

Table XIII-18: Manufacturing Processes for Parts of Sub-System SS8

Part #	Manufacturing Process
SS8-P1	Code
SS8-P2	Code

Table XIII-19: Manufacturing Processes for Parts of Sub-System SS9

Part #	Manufacturing Process
SS9-P1	Purchased
SS9-P2	Purchased
SS9-P3	Code
SS9-P4	Purchased
SS9-P5	Purchased
SS9-P6	Purchased
SS9-P7	Purchased

SS9-P8	Purchased
--------	-----------

Table XIII-20: Manufacturing Processes for Parts of Sub-System SS10

Part #	Manufacturing Process
SS10-P1	Purchased
SS10-P2	Purchased

XIII.E.2. Assembly Processes

To facilitate easier transportation, repair, and redesign, a majority of the rover's components are removable. This led to an extensive assembly process for all major subsystems of the rover.

XIII.E.2.a. *Assembly Process for Sub-System SS1 - Frame*

The frame was recycled from a previous years rover project, however it was cut to size and added onto substantially. Team 57 used angle grinders to cut a 4 ft X 2 ft section of the frame first. After, once the waterjet was complete, brackets for the central axle that supported the legs and the axle that supported the differential bar were clamped on, and then welded via the MIG.

Before being welded, channels were cut into the axle for the suspension so that retaining rings may be added. Similarly, the axle for the legs was lathed down to a proper diameter on either to fit the legs, and channels cut for retaining rings.

The axle brackets were placed around the frame axle and then welded to the middle of the frame. This tight fit ensured the axle would be secure but prevents it from being removed. The Electronics platforms are 3D printed and connect to the frame through a removable snap fit. Other electronic systems are attached in a similar way.

After full assembly of the rover, the team also painted the frame white. This accomplishes two things, one: it assists in heat management of the rover by reflecting sunlight, and two: assist with the qualitative constraint of keeping the rover aesthetically pleasing.

ADD PICTURES HERE

XIII.E.2.b. *Assembly Process for Sub-System SS2 - Drivetrain*

First, the parts for the axle and body of the motor housing were cut to size. The caps for the motor housings were complete via a waterjet. Once the three parts were complete, they were welded together, with the axle and cap being welded first, and the cap after. Holes were then drilled and threaded into the main body of the motor housing.

The drive shafts were custom parts that were lathed. To attach them to the motors, the key must first be taken out, then the drive shaft slotted in, and then key hammered back in.

The motors slot into the housing with the screws already in. Due to an oversight, there is no way to hold a screwdriver to the screw when tightening, so the thread must instead be held while the nut is tightened. This requires that the screws be sawed off every time the subsystem is disassembled. The solution to this is to mill slots in the motor housing to allow access for a hex wrench.

The wheel slides onto the axle and keys into the hubcap. The hubcap is bolted to the wheel with three half-inch bolts and nuts. A retaining ring is placed on the axle to prevent the wheel sliding too close to the motor housing. A hitch pin prevents the wheels from sliding off the axles away from the rover.

Lastly, the motor housing simply screws onto the leg attachment.

XIII.E.2.c. Assembly Process for Sub-System SS3 - Legs

The manufacturing of the leg subsystem started with cutting the hollow aluminum bars to size. Afterwards, a C shape was cut on the mill using a hole-saw bit that was the same diameter was the piece the leg was being welded to.

Next, the legs were welded together utilizing the MIG aluminum welding machine in the shop. Practice was conducted beforehand with a test piece to ensure that the proper angle and technique could be achieved.

After the welds had been completed, bushings were created via a lathe as a way to step up the diameter of the legs to the motor housing attachments. This was due to an error in ordering. Holes were drilled in the bushings so that the screw of the motor housing attachment may press into it, keeping the bushing tight on the legs.

The set screw tightens the motor housing attachments to the leg struts. To attach the legs to the frame, the rocker bearing is slid onto the frame and secured in place by retaining rings.

To attach the rocker to the bogie, one of the RB brackets and RB spacers are placed against the end of the rocker and lined up with the holes with the spacer on the inside. Two bolts are then passed through all three parts. The bogie is then added by placing the axle in the large hole on the RB bracket. The second spacer and bracket are then added and the nuts tightened to the bolts, securing all components in place.
rings.

XIII.E.2.d. Assembly Process for Sub-System SS4 – Differential Suspension

The differential suspension subsystem made ample use of the waterjet to cut the main components, including the differential bar itself, as well as the two brackets that were welded to the frame. As mentioned previously in the frame section, the axle was put on the lathe and channels cut into it for retaining rings.

To assemble the system, the differential bar is held between the two supports and the axle passed through all three holes. Retaining rings are then added to keep the components from sliding along the length of the axle. Two rod-end joints are bolted to the outer holes of the differential bar. The other two are bolted to their corresponding holes on the legs. The two differential struts are then passed through the rod-end joints. Nuts are used to secure the struts to the top and bottom of the rod-end joints. The system is now stable and ready to support the rover.

XIII.E.2.e. Assembly Process for Sub-System SS5 - Electric Propulsion

We connected wires from the appropriate voltage converter to the motor controllers and then from the controllers to the motors. The motors are screwed inside the motor housing. The code for the control of the motors was written and modified several times over the development of the project. The version seen in the appendix is the version that was used at the presentation and the showcase and the stable backup version. There is also an updated version with different speed options.

XIII.E.2.f. Assembly Process for Sub-System SS6 – Power Supply

Two struts were cut out of the frame to allow for the battery to fit in the rover. The frame holds the battery securely. The battery strap is passed over the top of the frame and secured for extra stability. If even more security is desired, an extra ratchet strap can be easily attached.

XIII.E.2.g. Assembly Process for Sub-System SS7 – Power Distribution

The wires coming from the battery were stripped and had ring terminals crimping onto them for connection with the battery terminals. The positive terminal has two wires coming from it, one goes to a breaker that switches on the motors and the second goes to a breaker that switches on the control and communication systems. At each point of connection between components, wires were stripped and either soldered together (for devices with existing wires), or screwing in to a terminal (for devices with screw terminals). For the motor controllers specifically, loops needed to be made and soldered at the end of all connected wires so that the wire could fit around the screw and be locked in to place. Heat shrink, electrical tape, or both were used at all soldering points and device connection points. For connections to the Arduino, stranded 24 AWG wire was crimped into a solid connection to make plugging the wire into the Arduino screw terminals, as well as the breadboard, easier. Control signal for the motor controllers was sent from the Arduino to the controllers using a 4 pin grove connector at the controller end and free wires plugged in to the Arduino at the other end.

XIII.E.2.h. Assembly Process for Sub-System SS8 – Controller

The digital controller was created using a combination of Python, HTML, CSS, and JavaScript. A Flask application written in Python renders a webpage on the Jetson Nano's webserver hotspot, which is started by a bash script and the Python service on boot. The Flask application serves the webpage of the controller that is written in HTML, CSS, and JavaScript. The HTML acts as the foundation of the buttons and structure of the webpage, the CSS helps achieve the style of the controller and positioning of the buttons on the controller, and the JavaScript handles the commands for each button pressed on the controller. These commands are then sent from the JavaScript back to the Flask application, and the Flask application determines what to do with those commands.

XIII.E.2.i. Assembly Process for Sub-System SS9 – Object Detection and Reaction

Ultrasonic sensors are wired into the Arduino and the 5-volt bus bar for power. For the ultrasonic sensors, a function was added in C++ to the Arduino that will read the distance from the sensors every second and then write those distances to the Arduino's NativeUSB serial connection, which

sends the information to the Jetson Nano. The camera was set up on the Jetson Nano using OpenCV. This framework was chosen because of its utilization in computer vision applications. We configured the camera using OpenCV to set the foundation of object detection via computer vision for future applications. A function was also written to implement the Bug-0 algorithm in C++.

XIII.E.2.j. Assembly Process for Sub-System SS10 – Communication

A M.2 wi-fi NIC module was added to the Jetson Nano to enable the use of wi-fi for our communication system. We had to configure the wi-fi module by downloading the correct drivers for our module. We then wrote bash scripts and a python service to host a wi-fi hotspot and Flask server on startup. This allows us to connect our controller to the wi-fi module and remotely control the rover.

]

XIII.F. Design Phase Testing Supplement

[N/A]

[N/A]

XIII.F.1. Design Phase Testing Methods and Details

[N/A]

XIII.F.2. Design Phase Testing Results

[N/A]

XIII.G. Testing & Validation Supplement

[]

XIII.G.1. Testing & Validation of Function F1 - [Drive on Various Terrains]

Michael Cannon (ME), Tristan Hughes (ME), Kyle Pitre (ME)

[]

XIII.G.1.a. *Test Protocol Description - F1*

Suspension Test

Objective: The rocker-bogie leg mechanism and differential suspension system are designed to allow the rover to drive over uneven terrain and obstacles while maintaining six points of contact with the ground. The goal of the test is to find the tallest low-radius obstacle that the rover can transverse. Low-radius obstacles refer to obstacles that have radius to height ratios close to one. These obstacles have extremely high slopes and induce the more extreme angles in the rover's legs. Examples include tree roots, curbs, and parking blocks.

Test description: Rover driven over low-radius obstacles of increasing height until suspension limit was reached. Tests were filmed so that any visual data could be extracted and recorded.

Note: Due to limited testing windows, tests had to be conducted on and around campus, and utilize whatever test obstacles could be found. Given adequate time, a dedicated testing course designed to test the rover to its limits should be constructed.

Turning Test

Objective: Measure turning rates and radii for the two turning methods on several terrains. The turn-in-place method is only suited for sand, smooth concrete, and tile. The moving turn can be conducted on any surface.

Test description: Rover was directed to execute the desired type of turn on the chosen terrain. Tests were filmed so that turn rates and radii could be calculated at a later time. Some turning tests were conducted with an additional 80 lbf of load attached. These are considered a part of the carrying capacity and can be found in Section X.P.

Terrain Test

Objective: Drive over selected terrain (concrete, grass, gravel, sand) and verify acceptable rover performance. Acceptable performance is defined as: ability of rover to maintain six points

of contact with the ground, ability of the rover to make turns, ability to maintain at least 0.5 m/s with motors at 25% of maximum speed, minimal wheel slipping.

Test description: Rover was driven and maneuvered on concrete (outside AMMF and PFT), grass (outside PFT), gravel (outside PFT), and sand (UREC volleyball courts). Tests were filmed so that speed could be determined by tracking wheel movement and other test requirements observed.

One thing to note about all of these tests is the rover speed and turn rate can be altered by changing the code that controls the motors. The code currently uploaded to the rover is a newer version than that used in most of these tests. The code may also be altered by a future team, depending on what outputs they require.]

XIII.G.1.b. *Equipment and Instrumentation - F1*

- Measuring tape
- Stopwatch
- Camera
- Video editing software (iMovie)

XIII.G.1.c. *Data Acquisition & Analysis – F1*

Michael Cannon (ME)

Suspension Test

Obstacles measured with measuring tape. Test footage reviewed to track strut angles and general rover performance.

Turning Test

Turn rates measured by reviewing test footage. Video editing software was used to find the turn rate in the following way. A video of the rover turning would be uploaded. It would then be cut so that the video would start and end on the exact frame of a known turn angle (usually 90 degrees). The distance rotated would then be divided by the length of the video to get the turn rate.

The turn radius, when not zero, was measured with a measuring tape.

Terrain Test

Editing software was also used to get average speed data from test footage. By tracking the rotation of the wheel bolts, a video could be edited frame by frame to a known number of wheel rotations (usually 3 or 4). This could be used to find distance traveled.

$$D = d * \pi * N$$

Where D=distance, d=wheel diameter of 13 inches, and N=number of rotations. The time of travel, now equal to the time of the edited video, could now be factored in to back out average speed.]

XIII.G.1.d. *Results Details – F1*

Table XIII-21: Suspension test results

Test #	Obstacle Description	Obstacle Height (in)	Pass/Fail
Test 1	Traffic cone on its side	3	Pass
Test 2	Curb	4	Pass

Test 3	Root	5.5	Pass
Test 4	Parking Block	7	Pass (Bogie tilt angle maxed out)

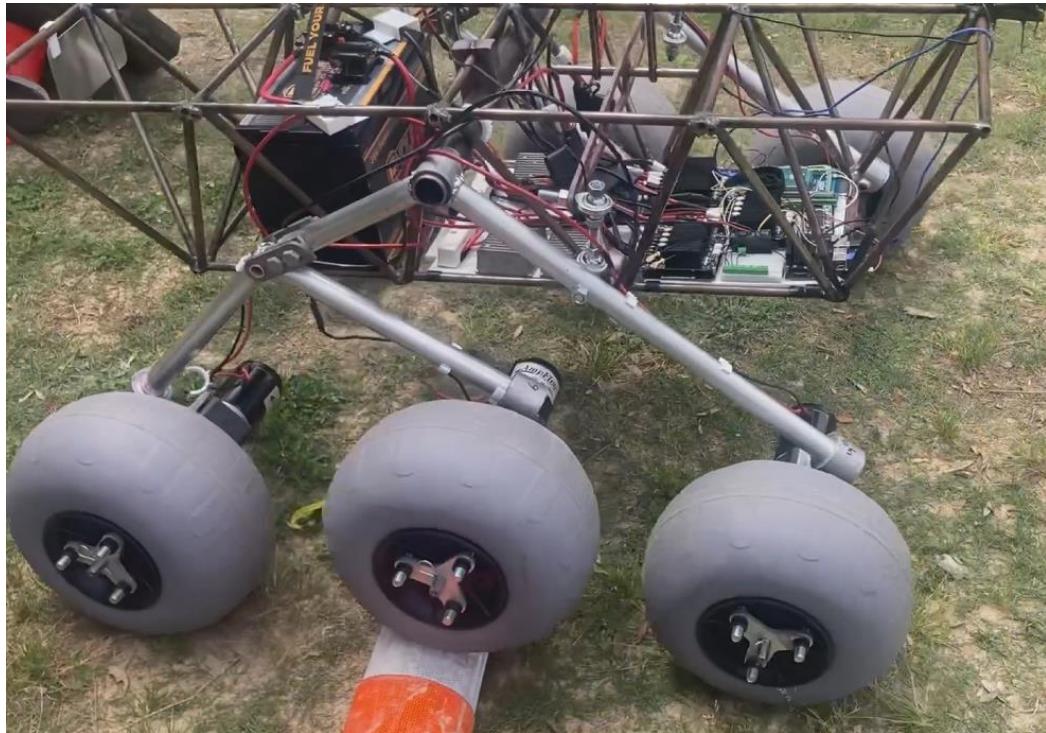


Figure XIII-11: Rover driving over cone in Suspension Test 1



Figure XIII-12: Rover driving over curb in Suspension Test 2



Figure XIII-13: Rover driving over large root in Suspension Test 3



Figure XIII-14: Maximum bogie tilt angle reached while overcoming 7-inch parking block.

Table XIII-3: Terrain test results

Terrain	Average speed (m/s)	All 6 wheels maintain contact?	Noticeable wheel slipping during straight drive?	Turning types possible.
Rough Concrete	0.56	Yes	No	Moving Turn
Smooth Concrete		Yes	No	Turn-in -place, Moving Turn
Gravel	0.56	Yes	No	Moving Turn
Grass		Yes	No	Moving Turn
Sand		Yes	No	Turn-in -place, Moving Turn

Table XIII-2: Turning test results

Terrain	Turn Type	Turn Radius (ft)	Turn Rate (deg/s)	Direction
---------	-----------	------------------	-------------------	-----------

Smooth Concrete	Turn-in-place	0	23.4	CW
Smooth Concrete	Turn-in-place	0	25.7	CCW
Sand	Turn-in-place	0	15	CW
Sand	Turn-in-place	0	10.8	CCW
Grass/Concrete	Moving Turn	7	11.3	CW
Sand	Moving Turn	3.5	15	CW

XIII.G.2. Testing & Validation of Function F2 - Facilitate Expansion

Michael Cannon (ME), Tristan Hughes (ME), Kyle Pitre (ME)

[]

XIII.G.2.a. Test Protocol Description - F2

While the rover was undergoing test drives to validate the “Drive on Various Terrains” function, an extra load of 36 Kg was added to simulate the addition of extra components. The rover was then driven over concrete, grass and sand. The tests were filmed and the footage reviewed to evaluate the rover’s general performance with the additional load. The load was centered roughly 1 ft from the front of the rover as this was the easiest location for attachment.

[]

XIII.G.2.b. Equipment and Instrumentation – F2

- 36 Kg weight.
- Ratchet straps
- Measuring tape
- Camera
- Video editing software (iMovie)

[]

XIII.G.2.c. Data Acquisition & Analysis – F2

Average drive speeds were determined through the same methods as described in the previous section.

[]

XIII.G.2.d. Results Details – F2

Table XIII-4: Extra load test results

Terrain	Able to drive and navigate?	Speed (m/s)	Unloaded Speed (m/s)
Concrete	Yes	0.62*	0.55
Grass	Yes	0.50	0.56
Sand	Yes	0.50	0.56

The rover exhibited a slight but acceptable decrease in speed. All other systems continued to operate as designed. This confirms that the rover can still operate with additional equipment installed.

XIII.G.3. Testing & Validation of Function F3 – Distribute Power to Electronics

Ethernan Smith (EE) and Patrick Maloney (EE)

XIII.G.3.a. *Test Protocol Description – F3*

Various tests were performed to ensure proper distribution of power to the electronics.

Voltage Converter (Ethernan Smith (EE)):

Each voltage converter underwent testing to validate its ability to adequately convert voltages. For the 36/48V to 24V converters, this involved applying a voltage from the DC power supply to the high end of the converter and then measuring the output voltage using a multimeter on the low side. This procedure was repeated for each converter, with each one tested five times to ensure accurate results. The applied voltages were not chosen randomly; instead, they were selected based on the anticipated voltage levels for each converter.

Bus bar & Electrical Subsystems:

To test the bus bars, we once again utilized the DC power supply. Clamps connected to the DC power supply were used to apply a voltage of 40V to the bus bar(s) being tested. Appropriately sized wires were then fed from the bus bars to the previously tested voltage converters, with temporary connections established between the wires from the bus bars and those on the voltage converters. Using a multimeter, the output voltages were measured on the low side of the voltage converter. Since we had already verified that the voltage converters function properly, if the voltage on the low side matched the rating of the voltage converter, we concluded that the subsystem, and therefore the bus bar, functioned as expected

Torque Test: (Patrick Maloney EE and Michael Cannon ME)

The motor used for testing was placed on an elevated surface along with all the components required for it to run as desired (battery, step down converter, ESC). The driveshaft connected to the motor was connected at the other end to the rim of an extra wheel. This rim acted as a pulley that supported the testing weight. During each test the motor was run to lift the weight from the ground to a height above the ground. This test was videoed with a tape measure next to the weight to measure how high the weight was lifted. This was done in order to calculate the speed the motor was rotating at. The values of the predicted current come from the model described in section XIII.D.5 Electric Propulsion.

XIII.G.3.b. *Equipment and Instrumentation – F3*

- DC Power Supply
- Multimeter
- Temporary Wire Connectors

Torque Test:

- Phone for use of Endure Power app for current drawn from battery measurement.
- Phone for video of test for speed calculation
- Tape measure
- Spare wheel rim
- Cable used as pulley

- 10 and 5 pound weights meant for barbells
- Laptop for control of the motor
- Battery
- 36/24V converter
- ESC

XIII.G.3.c. Data Acquisition & Analysis – F3

Each test was conducted 5 times, to ensure accurate measurements. The tests were pass/fail, with a test being successful so long as the output voltage is within 1V of the expected value.

The 15 pound test was repeated three times and the 20 and 25 pound tests were repeated five times. The current was measured for each trial and averages were taken for each set of tests. The standard deviation and standard uncertainty in the mean was calculated for each set of tests using the following equations:

$$\sigma = \sqrt{\frac{\sum (x_i - \mu)^2}{N}}$$

$$\Delta\mu = \sigma \sqrt{N}$$

where N is the number of trials,

$$\bar{x} = \frac{\sum x_i}{N}$$

is the value of current for each test,

$$\mu = \bar{x}$$

is the mean,

$$\sigma = \sqrt{\frac{\sum (x_i - \mu)^2}{N}}$$

is the standard deviation, and

$$\Delta\mu = \sigma \sqrt{N}$$

is the standard uncertainty in the mean. The full results and data are shown in the next sections.

XIII.G.3.d. Results Details – F3

Results for Voltage Converter Tests:

Table XIII.5: 12V to 5V 10A Converter Test Result Data Table:

Test #	Vin (from DC power supply)	Expected output	Vout (initial)	Vout (after 60s)	Pass	Fail	N/A	Comments
1	12V	5V	4.955V	4.956V	✓			
2	12.1V	5V	4.956V	4.956V	✓			
3	11.9V	5V	4.955V	4.956V	✓			
4	11.6V	5V	4.955V	4.955V	✓			
5	12.4V	5V	4.956V	4.955V	✓			

We have three 48/36V to 24V 30A Converters, and we will be testing each:

Table XIII6: 48/36V to 24V 30A Converter Test (1) Result Data Table:

Test #	Vin (from DC power supply)	Expected output	Vout (initial)	Vout (after 60s)	Pass	Fail	N/A	Comments
1	40.4V	24V	24.26V	24.26V	✓			
2	45.3V	24V	24.26V	24.26V	✓			
3	36.2V	24V	24.26V	24.26V	✓			
4	35V	24V	24.26V	24.26V	✓			
5	42V	24V	24.26V	24.26V	✓			

Table XIII7: 48/36V to 24V 30A Converter Test (2) Result Data Table:

Test #	Vin (from DC power supply)	Expected output	Vout (initial)	Vout (after 60s)	Pass	Fail	N/A	Comments
1	40.4V	24V	24.25	24.24V	✓			
2	45.3V	24V	24.24V	24.24V	✓			
3	36.2	24V	24.26V	24.25V	✓			
4	35V	24V	24.25V	24.25V	✓			
5	42V	24V	24.25V	24.25V	✓			

Table XIII8: 48/36V to 24V 30A Converter Test (3) Result Data Table:

Test #	Vin (from DC power supply)	Expected output	Vout (initial)	Vout (after 60s)	Pass	Fail	N/A	Comments
1	40.5V	24V	24.25V	24.25V	✓			
2	45.3V	24V	24.25V	24.25V	✓			
3	36.2V	24V	24.25V	24.25V	✓			
4	35.1V	24V	24.25V	24.25V	✓			
5	42V	24V	24.25V	24.25V	✓			

Table XIII8: 48/36V to 12V 30A Converter Test (3) Result Data Table:

Test #	Vin (from DC power supply)	Expected output	Vout (initial)	Vout (after 60s)	Pass	Fail	N/A	Comments
1	40.5V	12V	12.12V	12.12V	✓			
2	45.3V	12V	12.12V	12.12V	✓			
3	36.2V	12V	12.12V	12.12V	✓			
4	35.1V	12V	12.12V	12.12V	✓			
5	42V	12V	12.12V	12.12V	✓			

	Torque Test	Steady-State Current (A)	Standard Deviation	Standard Uncertainty of the Mean
1	4.68 Nm	Test 1	3.3	
2		Test 2	2.1	
3		Test 3	2.4	
4		Average	2.6	0.509901951 0.294392029 2.6 ± 0.3
5		Test 1	2.8	
6	6.24 Nm	Test 2	3.5	
7		Test 3	2.8	
8		Test 4	3	
9		Test 5	4.6	
10		Average	3.3	0.304105245 3.3 ± 0.3
11	7.80 Nm	Test 1	3.5	
12		Test 2	5.1	
13		Test 3	3.1	
14		Test 4	3.3	
15		Test 5	4.6	
16		Average	3.9	0.785875308 0.351454122 3.9 ± 0.4
17				
18				

Figure XIII-15: Test Results

Conclusion:

All voltage converters are in working order as the voltages actual output voltages were extremely close to the expected outputs, and different by the same amount every time the test was run.

Bus Bar and Electrical Subsystem Testing:

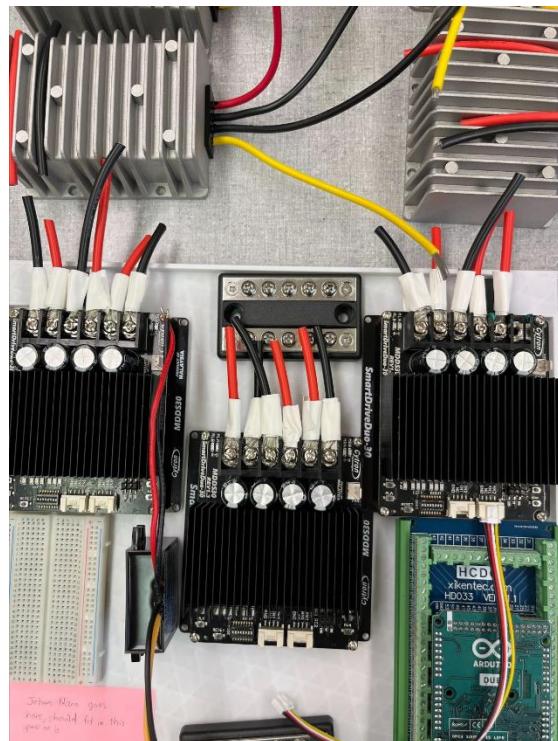
Because the voltage converters were all verified as working properly, the testing for the bus bars and subsystems did not need to be repeated to ensure accurate results. Instead, when subsystems were constructed, when the output voltages were verified to be what we expected to be, we could move forward with confidence that that system would operate properly.

Each bus bar was tested as described in XII.G.3.a. The first subsystem tested was from one bus bar down to the output of one of the 36/48V to 24V converters. Once we verified that the output voltage was ~24V (24.26V), the second bus bar was tested in a subsystem that included the 36/48V to 12V converter, as that is how that bus bar would be used when implemented in the rover. The output voltage for that test verified that the subsystem was operating properly, as the measured output voltage on the low side of the voltage converter was found to be 12.45V. After completing testing for the bus bar subsystems, motor controller connections were made by soldering wire loops and then tested. These connections along with the motors connected with these connections were tested at the same time. We connected the previously successfully tested subsystem that included the 36/48V to 24V converter to the battery, with the newly made motor connections attached to the low side of the voltage converter. Patrick then implemented and tested his motor controller code to verify that the motors were being properly powered and could turn the driveshafts as desired. This was repeated with all motors, and at the conclusion of these tests all motors were all verified as operating correctly.

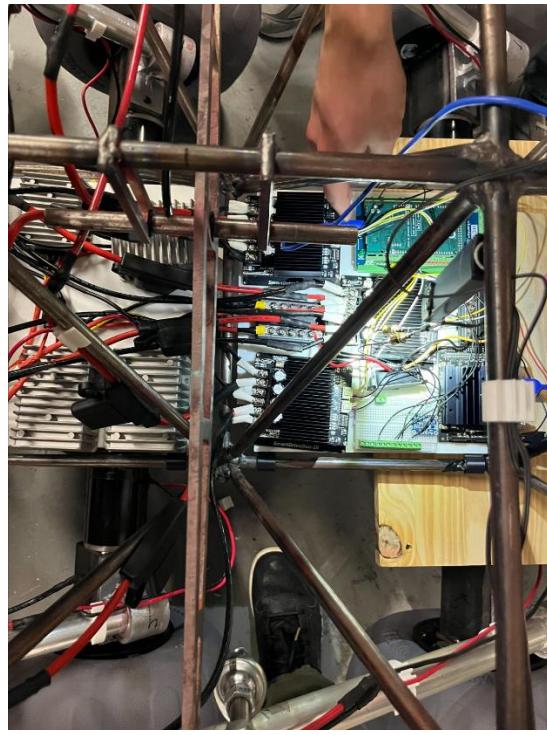
Circuit Breaker/Full Rover Testing:

After all subsystems had been constructed and implemented onto the rover following their successful testing as described above, the entire circuit was to be tested with the circuit breaker acting as an on/off switch. The circuit was implemented in its entirety as planned, shown in the flowchart in the figure. Once the construction of the circuit on-board the rover was completed, the circuit breaker was closed in, and the electrical components were inspected to verify that they were all properly receiving power. They were, which can be seen in the figure. A picture of

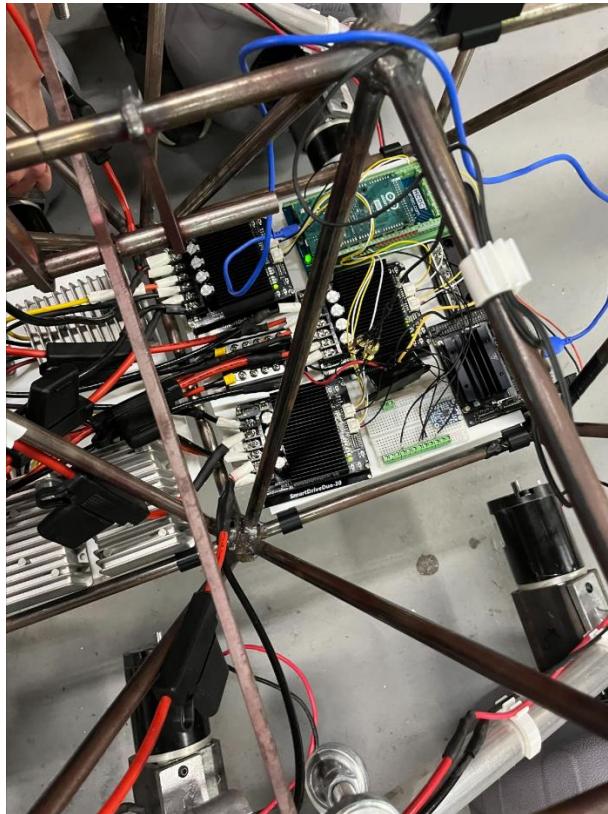
the circuit as a whole is seen in the figure. At the conclusion of this test the circuit breaker's manual 'open' button was pushed, cutting power from the circuit. Again, the electrical components were inspected, and upon inspection it was verified that they were no longer receiving power from the battery. So, the entire circuit had been implemented successfully as intended. However, the circuit had to be adjusted as described in section the, and this test was repeated when the adjustment was made, and that test was a success as well.



All motor controller loop connections



Final Setup of all Electrical Components



Final Setup of all Electrical Components with circuit breaker closed in, notice all LED lights are on, signaling that the electrical systems are being powered as desired

Torque Test:

The current values are slightly low but still follow the experimental values and are within the uncertainty. All torques were also provided by the motors at above the minimum speed required. This test validates the motor and battery selection. Expected rover speed was calculated to be higher than required by our constraints.

Table XIII5: Torque test results and calculations, measured values

Torque Test		Rise Time (s)	h1 (inches)	h2 (inches)	Δh (inches) (± 2)
15 lbf	Test 1	1.838	5	51	46
	Test 2	2.016	5	57	52
	Test 3	2.294	5	60	55
	Average				
20 lbf	Test 1	2.334	5	58	53
	Test 2	2.384	5	57	52
	Test 3	2.426	5	58	53
	Test 4	2.374	5	57	52
	Test 5	2.222	5	55	50
	Average				
25 lbf	Test 1	2.782	5	67	62
	Test 2	2.262	4	53	49
	Test 3	2.234	4	51	47
	Test 4	2.254	5	56	51
	Test 5	2.376	4	57	53
	Average				

Table XIII5: Torque test results and calculations, calculated values

Torque Test		Average Velo (ft/s)	Average Velo (m/s)	Angular Velo	Resulting Drive Velo (ft/s)	Resulting Drive Velo (m/s)
15 lbf	Test 1	2.08560029	0.635690968	9.067827349	6.800870511	2.072905332
	Test 2	2.149470899	0.65515873	9.34552565	7.009144237	2.136387164
	Test 3	1.997965708	0.608979948	8.686807425	6.515105568	1.985804177
	Average	2.077678966	0.633276549	9.033386808	6.775040106	2.065032224
20 lbf	Test 1	1.892316481	0.576778063	8.227462961	6.170597221	1.880798033
	Test 2	1.817673378	0.554026846	7.902927731	5.927195798	1.806609279
	Test 3	1.820555098	0.554905194	7.915456946	5.936592709	1.809473458
	Test 4	1.825329963	0.556360573	7.936217233	5.952162924	1.814219259
	Test 5	1.875187519	0.571557156	8.152989212	6.114741909	1.863773334
	Average	1.846212488	0.562725566	8.027010816	6.020258112	1.834974673
25 lbf	Test 1	1.857177091	0.566067577	8.074683004	6.056012253	1.845872535
	Test 2	1.80518715	0.550221043	7.848639783	5.886479837	1.794199054
	Test 3	1.753207998	0.534377798	7.622643468	5.716982601	1.742536297
	Test 4	1.885536823	0.574711624	8.197986189	6.148489642	1.874059643
	Test 5	1.858866442	0.566582492	8.08202801	6.061521007	1.847551603
	Average	1.831995101	0.558392107	7.96519609	5.973897068	1.820843826

Table XIII5: Torque test results and calculations, known values

Torque Test	Radius (inches)	Torque (lbf*ft)	Torque (N*m)
15 lbf	Test 1	2.76	3.45
	Test 2	2.76	3.45
	Test 3	2.76	3.45
	Average		
20 lbf	Test 1	2.76	4.6
	Test 2	2.76	4.6
	Test 3	2.76	4.6
	Test 4	2.76	4.6
	Test 5	2.76	4.6
	Average		
25 lbf	Test 1	2.76	5.75
	Test 2	2.76	5.75
	Test 3	2.76	5.75
	Test 4	2.76	5.75
	Test 5	2.76	5.75
	Average		

Resulting Drive Velo Vs Load on Wheel

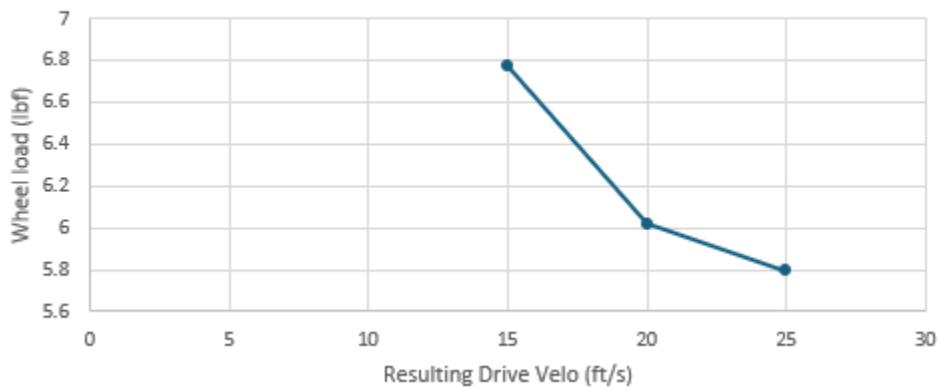


Figure XIII5: Expected drive velocity vs applied load:

Table XIII5: 12V to 5V 10A Converter Test Result Data Table:

	Stopwatch	reaction delay (theoretical average)	Adjusted time	Average
15 lbf test 1	2.33	0.4	1.93	
	2.4	0.4	2	
	2.2	0.4	1.8	1.838
	2.08	0.4	1.68	
	2.18	0.4	1.78	
15 lbf test 2	2.49	0.4	2.09	
	2.3	0.4	1.9	
	2.34	0.4	1.94	2.016
	2.49	0.4	2.09	
	2.46	0.4	2.06	
15 lbf test 3	2.62	0.4	2.22	
	2.73	0.4	2.33	
	2.67	0.4	2.27	2.294
	2.74	0.4	2.34	
	2.71	0.4	2.31	
20 lbf test 1	2.77	0.4	2.37	
	2.66	0.4	2.26	
	2.8	0.4	2.4	2.334
	2.71	0.4	2.31	
	2.73	0.4	2.33	
20 lbf test 2	2.84	0.4	2.44	
	2.74	0.4	2.34	
	2.83	0.4	2.43	2.384
	2.85	0.4	2.45	
	2.66	0.4	2.26	
20 lbf test 3	2.69	0.4	2.29	
	2.67	0.4	2.27	
	2.95	0.4	2.55	2.426
	2.92	0.4	2.52	
	2.9	0.4	2.5	
20 lbf test 4	2.8	0.4	2.4	
	2.74	0.4	2.34	
	2.82	0.4	2.42	2.374
	2.78	0.4	2.38	
	2.73	0.4	2.33	

20 lbf test 5	2.52	0.4	2.12	2.222
	2.59	0.4	2.19	
	2.54	0.4	2.14	
	2.81	0.4	2.41	
	2.65	0.4	2.25	
25 lbf test 1	3.24	0.4	2.84	2.782
	3.2	0.4	2.8	
	3.25	0.4	2.85	
	3.09	0.4	2.69	
	3.13	0.4	2.73	
25 lbf test 2	2.7	0.4	2.3	2.262
	2.59	0.4	2.19	
	2.63	0.4	2.23	
	2.72	0.4	2.32	
	2.67	0.4	2.27	
25 lbf test 3	2.65	0.4	2.25	2.234
	2.68	0.4	2.28	
	2.53	0.4	2.13	
	2.62	0.4	2.22	
	2.69	0.4	2.29	
25 lbf test 4	2.61	0.4	2.21	2.254
	2.74	0.4	2.34	
	2.62	0.4	2.22	
	2.63	0.4	2.23	
	2.67	0.4	2.27	
25 lbf test 5	2.72	0.4	2.32	2.376
	2.8	0.4	2.4	
	2.75	0.4	2.35	
	2.78	0.4	2.38	
	2.83	0.4	2.43	

XIII.G.4. Testing & Validation of Function F4 – Operate Safely

Patrick Maloney (EE) and Ethernan Smith (EE)

XIII.G.4.a. Test Protocol Description – F4

For the test of the digital kill-switch, the rover malfunctioned while in otherwise normal operation and the digital kill-switch was pressed. For the test of the physical kill-switch the rover was manually switched off while moving using the switch on top of the battery.

XIII.G.4.b. Equipment and Instrumentation – F4

N/A

XIII.G.4.c. Data Acquisition & Analysis – F4

N/A

XIII.G.4.d. Results Details – F4

In both tests, the rover came to a stop immediately, verifying that this function has been realized in the final product.



Figure XIII5: Activating the physical kill-switch

XIII.G.5. Testing & Validation of Function F5 – Transmit Data

Creighton Cathey (EEC)

XIII.G.5.a. Test Protocol Description – F5

Several tests were conducted to evaluate the rover's Wi-Fi data transmission capability, including Wi-Fi ping tests and serial monitor latency tests. The Wi-Fi ping test assessed the latency of the Wi-Fi connection from our web server to the digital controller using a MacBook Air as the controller to send pings to the Jetson Nano's IP address. Serial monitor latency tests evaluated the round-trip communication time between the Jetson Nano and an Arduino, accounting for command transmission and processing delays on both devices. These tests were crucial to verify the Wi-Fi module's performance under operational conditions.

XIII.G.5.b. Equipment and Instrumentation – F5

The equipment used for the transmission tests included a MacBook Air (M2, 2022), serving as the controller, and the Jetson Nano configured as a Wi-Fi access point. The instrumentation involved in the tests included a network interface card for the Jetson Nano. Tools

such as terminal applications for ping tests and development environments for scripting serial communication were utilized to accurately simulate and measure the transmission metrics.

XIII.G.5.c. Data Acquisition & Analysis – F5

Data from the Wi-Fi and serial transmission tests were recorded and analyzed. The Wi-Fi ping tests provided latency measurements, while the serial monitor tests offered insights into round-trip communication times, which included Arduino processing times. The data collected helped assess the overall efficiency of the data transmission system. The analysis focused on average latencies, standard deviations, and reliability under different test conditions. The results can be seen in the section below.

XIII.G.5.d. Results Details – F5

The results from the Wi-Fi ping tests showed an average latency of 4.96 ms, demonstrating excellent response times within the desired parameters for operational efficiency. Serial communication tests indicated an average round-trip latency of 16.08 ms, which was considered acceptable given the processing requirements on the Arduino. These results affirm the chosen Wi-Fi module's efficacy in meeting the rover's communication needs, with detailed results and comparative analyses documented here for reference.

Table XIII5: Testing results for F5

Network Latency	Average Network Latency	Webserver Latency	Average Webserver Latency	Serial Latency	Average Serial Latency	Average Latency
4.904	4.96	1.26	1.23	16.76	16.08	22.27
4.429		1.11		14.24		
4.141		1.34		16.8		
4.275		1.21		16.48		
3.356		1.29		16.45		
5.87		1.14		16.81		
3.385		1.18		17.28		
3.733		1.23		16.04		
3.54		1.16		16.11		
3.982		1.45		16.56		
6.176		1.24		16.05		
4.66		1.22		16.15		
5.369		1.15		15.96		
5.016		1.33		16.15		
5.468		1.16		16.23		
4.452		1.25		16.91		
5.204		1.33		15.95		
5.088		1.24		16.9		
4.877		1.23		16.55		
5.755		1.28		16.25		
6.116		1.18		15.76		
10.011		1.2		15.14		
7.058		1.32		16.57		
5.241		1.22		15.43		
6.647		1.31		16.86		
12.788		1.24		15.02		
7.528		1.26		16.4		
5.325		1.31		15.96		
6.092		1.14		17.1		
5.394		1.11		16.57		
3.272		1.18		17.3		
3.265		1.15		13.35		
4.04		1.26		12.24		
5.333		1.3		16.37		
3.399		1.16		16.04		
5.533		1.15		13.06		
4.203		1.21		16.28		
3.262		1.24		16.54		
7.723		1.16		17.08		
4.502		1.15		16.86		
4.395		1.24		16.25		
4.414		1.31		15.65		
4.537		1.15		16.44		
3.999		1.22		16.58		
4.282		1.47		16.06		
3.78		1.16		16.19		
4.224		1.16		16.16		
4.086		1.27		16.28		
2.741		1.39		16.25		

XIII.G.6. Testing & Validation of Function F6 – Operate Remotely

Creighton Cathey (EEC)

XIII.G.6.a. *Test Protocol Description – F6*

To ensure the functionality and reliability of the Digital Controller subsystem for the remote operation of the rover, a comprehensive testing protocol was implemented. This included an operating distance test, where the rover was controlled remotely across the entire length of a 250-foot testing field to assess signal integrity and control responsiveness. Additionally, usability tests were conducted with various digital devices (laptop, tablet, and phone) using a web application designed for rover control. These tests gauged user interface effectiveness and the ease of maneuvering the rover from different devices, ensuring the system was intuitive and robust across various platforms.

XIII.G.6.b. *Equipment and Instrumentation – F6*

The equipment used for the remote operation tests consisted of several digital devices, including a laptop, a tablet, and a smartphone, each serving as the controller at different times. The rover was equipped with a Wi-Fi access point to maintain a stable connection with the controllers. The instrumentation included software tools for monitoring network connectivity and response times and cameras to provide real-time visual feedback from the rover to the controller, ensuring the operator could effectively manage the rover's actions from a distance.

XIII.G.6.c. *Data Acquisition & Analysis – F6*

During the remote operation tests, data were collected on key performance metrics, including signal strength and user feedback on control ease and responsiveness. The operating distance test provided quantitative data on signal continuity and latency across the length of the testing field. User testing sessions were analyzed to gather qualitative insights into the system's usability and the effectiveness of the user interface on different devices.

XIII.G.6.d. *Results Details – F6*

The results from the operating distance test confirmed that the rover could be controlled effectively over a distance of 250 feet with no noticeable increase in latency or loss of Wi-Fi signal strength. The camera feed maintained a high frame rate throughout the test, indicating robust video streaming capabilities essential for remote operation. Feedback from the team members on the digital controller's usability was overwhelmingly positive, leading to minor iterations in the user interface to enhance user experience further. These outcomes demonstrate the Digital Controller subsystem's capability to support reliable and efficient remote operation of the rover across various digital platforms.

XIII.G.7. Testing & Validation of Function F7 – Monitor System Status

Creighton Cathey (EEC) and Patrick Maloney (EE)

XIII.G.7.a. *Test Protocol Description – F7*

Gyroscopic Sensor/Temperature Sensor Test:

To test if the slope of the gyroscope and temperature are accurate, the sensor was wired to the Arduino and a small amount of code was written to output the slope on the correct axis and the temperature to the serial monitor of a laptop. The breadboard that the sensor was on was then placed on a phone with a level app open and the slope on the serial monitor was compared to the slope on the phone. The room's temperature was also recorded on the serial monitor via the sensor and compared to the approximate known temperature based on the thermostat setting.

XIII.G.7.b. *Equipment and Instrumentation – F7*

- Arduino for power and signal
- Laptop for output
- Breadboard for sensor
- Phone for slope comparison

XIII.G.7.c. *Data Acquisition & Analysis – F7*

During the slope test, the slope of the phone was held constant for 10 seconds while the output slope from the gyroscope was observed on the output of the serial monitor. The output was recorded at four different slopes as shown in the next section. For the temperature test, the output temperature was observed for 30 seconds.

XIII.G.7.d. *Results Details – F7*

Recorded Temperature: 21.5 degrees Celsius

Approximate Actual Temperature: 21.1 degrees Celsius

Level App Slope (degrees)	Gyroscope Slope (degrees)
5	4.98
10	10.06
12	12.10
15	14.89

From the results of the two tests above, it can be concluded that both aspects of the sensor are accurate for our purposes and can be used to display correct information to the user via the GUI.

XIII.G.8. Testing & Validation of Function F8 – Contain Autonomous Features

Creighton Cathey (EEC)

XIII.G.8.a. *Test Protocol Description – F8*

The test protocol for the autonomous features of the rover was meticulously structured into two phases to assess the Object Detection and Reaction subsystem. The initial phase concentrated on the ultrasonic sensors, where objects were strategically placed at precise distances from the sensors to verify their detection accuracy and effective range. This phase was essential for setting a performance baseline for the sensors under controlled conditions. The subsequent phase involved testing the Bug-0 algorithm by placing the rover on blocks, which elevated the wheels off the ground to simulate movement scenarios. This setup enabled controlled testing of the algorithm's responsiveness to objects placed at varying distances, assessing its capability to process sensor inputs and make appropriate navigational decisions.

XIII.G.8.b. *Equipment and Instrumentation – F8*

The test equipment included the rover equipped with its ultrasonic sensors and the Jetson Nano Dev Kit for handling the computational tasks of the subsystem. The Arduino Due microcontroller played a crucial role in interfacing with the ultrasonic sensors and facilitating data processing. A breadboard setup was utilized during the ultrasonic sensor testing phase for flexible sensor connections and adjustments. The setup excluded using the webcam, focusing solely on the ultrasonic sensors and computational elements. Data logging and real-time software tools were crucial for capturing and evaluating sensor outputs and the algorithm's decision-making efficacy.

XIII.G.8.c. *Data Acquisition & Analysis – F8*

Data collection was executed in stages, starting with the ultrasonic sensors. Initial tests revealed that reading the sensors at a rate of once per second led to errors and inconsistent data. The protocol was adjusted to read the sensors ten times per second to address this issue. This high-frequency data collection allowed for the implementation of a filtering algorithm that removed the highest and lowest readings, calculating an average from the remaining data. This method significantly enhanced the accuracy and reliability of the sensor readings. Following the sensor optimization, data from the Bug-0 algorithm testing phase were collected, focusing on how well the algorithm responded to the refined sensor inputs and navigated around simulated obstacles.

The screenshot displays two instances of the Arduino IDE running on a Mac OS X system. Both windows are titled 'Arduino Due (Native ...)' and show the same sketch named 'sketch_apr23b.ino'. The sketch is a simple ultrasonic distance measurement program.

Sketch Content:

```
void loop() {
    float distance1 = measureDistance(TRIG1_PIN1, ECHO_PIN1);
    float distance2 = measureDistance(TRIG2_PIN2, ECHO_PIN2);
    Serial.println("Distance Sensor 1: ");
    Serial.print(distance1, 1); // Print distance with 1 decimal.
    Serial.println(" cm, Distance Sensor 2: ");
    Serial.print(distance2, 1); // Print distance with 1 decimal.
    Serial.println(" cm");
    delay(500); // Delay between measurements
}

float measureDistance(int trigPin, int echoPin) {
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);

    long duration = pulseIn(echoPin, HIGH);
    float distance = duration * 0.034 / 2; // Calculate the distance
    return distance;
}
```

Serial Monitor Output:

Both serial monitors show identical output, indicating distance measurements from two sensors. The data is printed in centimeters with one decimal place.

Output 1 (Left):

```
Distance Sensor 1: 451.0 cm, Distance Sensor 2: 104.2 cm
Distance Sensor 1: 25.1 cm, Distance Sensor 2: 107.8 cm
Distance Sensor 1: 35.2 cm, Distance Sensor 2: 107.8 cm
Distance Sensor 1: 28.1 cm, Distance Sensor 2: 107.8 cm
Distance Sensor 1: 38.1 cm, Distance Sensor 2: 33.6 cm
Distance Sensor 1: 32.3 cm, Distance Sensor 2: 33.6 cm
Distance Sensor 1: 30.1 cm, Distance Sensor 2: 33.6 cm
Distance Sensor 1: 36.0 cm, Distance Sensor 2: 30.7 cm
Distance Sensor 1: 34.8 cm, Distance Sensor 2: 30.7 cm
Distance Sensor 1: 31.4 cm, Distance Sensor 2: 31.4 cm
Distance Sensor 1: 118.2 cm, Distance Sensor 2: 451.0 cm
Distance Sensor 1: 451.0 cm, Distance Sensor 2: 104.2 cm
Distance Sensor 1: 16.1 cm, Distance Sensor 2: 35.8 cm
Distance Sensor 1: 32.4 cm, Distance Sensor 2: 29.6 cm
```

Output 2 (Right):

```
Distance Sensor 1: 451.0 cm, Distance Sensor 2: 104.2 cm
Distance Sensor 1: 25.1 cm, Distance Sensor 2: 107.8 cm
Distance Sensor 1: 35.2 cm, Distance Sensor 2: 107.8 cm
Distance Sensor 1: 28.1 cm, Distance Sensor 2: 107.8 cm
Distance Sensor 1: 38.1 cm, Distance Sensor 2: 33.6 cm
Distance Sensor 1: 32.3 cm, Distance Sensor 2: 33.6 cm
Distance Sensor 1: 30.1 cm, Distance Sensor 2: 33.6 cm
Distance Sensor 1: 36.0 cm, Distance Sensor 2: 30.7 cm
Distance Sensor 1: 34.8 cm, Distance Sensor 2: 30.7 cm
Distance Sensor 1: 31.4 cm, Distance Sensor 2: 31.4 cm
Distance Sensor 1: 118.2 cm, Distance Sensor 2: 451.0 cm
Distance Sensor 1: 451.0 cm, Distance Sensor 2: 104.2 cm
Distance Sensor 1: 16.1 cm, Distance Sensor 2: 35.8 cm
Distance Sensor 1: 32.4 cm, Distance Sensor 2: 29.6 cm
```

XIII.G.8.d. Results Details – F8

The results from the ultrasonic sensor tests initially showed discrepancies in object detection, which were mitigated by increasing the sampling rate and applying outlier filtering. With these adjustments, the sensors demonstrated high reliability in detecting objects within their optimal range of up to 2.5 meters, meeting the subsystem's operational requirements. The subsequent testing of the rover on blocks, using the Bug-0 algorithm, showed that the algorithm processed these sensor inputs to make navigational decisions. These tests provided valuable insights, significantly improving sensor performance and processing. The iterative enhancements ensured a robust framework for the subsystem's functionality, setting a solid foundation for further development and future educational applications.

Arduino Code (C++):

```
1 #include <Servo.h>
2 #include <Ultrasonic.h>
3
4
5 #define IN1 3 // Arduino pin 3 is connected to MDD530 pin IN1.
6 #define AN1 5 // Arduino pin 5 is connected to MDD530 pin AN1.
7 #define AN2 6 // Arduino pin 6 is connected to MDD530 pin AN2.
8 #define IN2 4 // Arduino pin 4 is connected to MDD530 pin IN2.
9
10 #define IN1_1 42 // Arduino pin 42 is connected to MDD530 pin IN1_1.
11 #define AN1_1 7 // Arduino pin 7 is connected to MDD530 pin AN1_1.
12 #define AN2_1 8 // Arduino pin 8 is connected to MDD530 pin AN2_1.
13 #define IN2_1 43 // Arduino pin 43 is connected to MDD530 pin IN2_1.
14
15 #define IN1_2 47 // Arduino pin 47 is connected to MDD530 pin IN1_2.
16 #define AN1_2 9 // Arduino pin 9 is connected to MDD530 pin AN1_2.
17 #define AN2_2 10 // Arduino pin 10 is connected to MDD530 pin AN2_2.
18 #define IN2_2 46 // Arduino pin 46 is connected to MDD530 pin IN2_2.
19
20 #define TRIG_PIN1 11
21 #define ECHO_PIN1 12
22 #define TRIG_PIN2 13
23 #define ECHO_PIN2 14
24
25 Ultrasonic ultrasonic1(TRIG_PIN1, ECHO_PIN1); // Initialize sensor 1
26 Ultrasonic ultrasonic2(TRIG_PIN2, ECHO_PIN2); // Initialize sensor 2
27
28
```

```

--  

29  char inChar;  

30  char inServo;  

31  Servo roverServo;  

32  int angle = 60;  

33  signed int speedLeft = 0, speedRight = 0;  

34  signed int speedLeftActual = 0; signed int speedRightActual = 0;  

35  signed int setSpeed = 35;  

36  signed int setSlowSide = 0;  

37  signed int setFastSide = 0;  

38  unsigned long currentMillis = 0;  

39  long distance1 = 0; // Declare distance variables globally  

40  long distance2 = 0;  

41  unsigned long lastPingTime = 0;  

42  const long pingInterval = 1000;  

43  

44 void setup() {  

45   pinMode(IN1, OUTPUT);  

46   pinMode(AN1, OUTPUT);  

47   pinMode(AN2, OUTPUT);  

48   pinMode(IN2, OUTPUT);  

49   pinMode(IN1_1, OUTPUT);  

50   pinMode(AN1_1, OUTPUT);  

51   pinMode(AN2_1, OUTPUT);  

52   pinMode(IN2_1, OUTPUT);  

53   pinMode(IN1_2, OUTPUT);  

54   pinMode(AN1_2, OUTPUT);  

55   pinMode(AN2_2, OUTPUT);  

56   pinMode(IN2_2, OUTPUT);  

57   pinMode(2, OUTPUT);  

58  

59   roverServo.attach(2);  

60   roverServo.write(60);  

61  

62   Serial.begin(9600);  

63   SerialUSB.begin(9600);  

64  

65   motorControl(0, 0);  

66 }

67  

68 void motorControl(signed int sL, signed int sR){  

69   if(sL > 0 && sR > 0){  

70     analogWrite(AN2, sL);  

71     analogWrite(AN2_1, sL);  

72     analogWrite(AN2_2, sL);  

73     digitalWrite(IN2, HIGH);  

74     digitalWrite(IN2_1, HIGH);  

75     digitalWrite(IN2_2, HIGH);  

76     analogWrite(AN1, sR);  

77     analogWrite(AN1_1, sR);  

78     analogWrite(AN1_2, sR);  

79     digitalWrite(IN1, HIGH);  

80     digitalWrite(IN1_1, HIGH);  

81     digitalWrite(IN1_2, HIGH);  

82   }  

83  

84   else if(sL > 0 && sR < 0){  

85     analogWrite(AN2, sL);  

86     analogWrite(AN2_1, sL);  

87     analogWrite(AN2_2, sL);  

88     digitalWrite(IN2, HIGH);  

89     digitalWrite(IN2_1, HIGH);  

90     digitalWrite(IN2_2, HIGH);  

91     sR = sR * -1;  

92     analogWrite(AN1, sR);  

93     analogWrite(AN1_1, sR);  

94     analogWrite(AN1_2, sR);  

95     digitalWrite(IN1, LOW);  

96     digitalWrite(IN1_1, LOW);  

97     digitalWrite(IN1_2, LOW);  

98   }  

99 }

```

```

100    else if(sL < 0 && sR > 0){
101        sL = sL * -1;
102        analogWrite(AN2, sL);
103        analogWrite(AN2_1, sL);
104        analogWrite(AN2_2, sL);
105        digitalWrite(IN2, LOW);
106        digitalWrite(IN2_1, LOW);
107        digitalWrite(IN2_2, LOW);
108        analogWrite(AN1, sR);
109        analogWrite(AN1_1, sR);
110        analogWrite(AN1_2, sR);
111        digitalWrite(IN1, HIGH);
112        digitalWrite(IN1_1, HIGH);
113        digitalWrite(IN1_2, HIGH);
114    }
115
116    else if(sL < 0 && sR < 0){
117        sL = sL * -1;
118        analogWrite(AN2, sL);
119        analogWrite(AN2_1, sL);
120        analogWrite(AN2_2, sL);
121        digitalWrite(IN2, LOW);
122        digitalWrite(IN2_1, LOW);
123        digitalWrite(IN2_2, LOW);
124        sR = sR * -1;
125        analogWrite(AN1, sR);
126        analogWrite(AN1_1, sR);
127        analogWrite(AN1_2, sR);
128        digitalWrite(IN1, LOW);
129        digitalWrite(IN1_1, LOW);
130        digitalWrite(IN1_2, LOW);
131    }
132
133    else if(sL == 0 && sR == 0){
134        analogWrite(AN2, sL);
135        analogWrite(AN2_1, sL);
136        analogWrite(AN2_2, sL);
137        digitalWrite(IN2, LOW);
138        digitalWrite(IN2_1, LOW);
139        digitalWrite(IN2_2, LOW);
140        analogWrite(AN1, sR);
141        analogWrite(AN1_1, sR);
142        analogWrite(AN1_2, sR);
143        digitalWrite(IN1, LOW);
144        digitalWrite(IN1_1, LOW);
145        digitalWrite(IN1_2, LOW);
146    }
147 }
148
149 void loop() {
150
151     currentMillis = millis();
152
153     if (currentMillis - lastPingTime >= pingInterval) {
154         distance1 = ultrasonic1.distanceRead();
155         distance2 = ultrasonic2.distanceRead();
156         SerialUSB.print("Sensor 1: ");
157         SerialUSB.print(distance1);
158         SerialUSB.print(" cm, Sensor 2: ");
159         SerialUSB.print(distance2);
160         SerialUSB.println(" cm");
161
162         lastPingTime = currentMillis; // Save the last ping time
163     }
164

```

```

165 //maximum speed for test right now will be 50 out of 100
166 if (Serial.available() > 0) {
167     char inChar = Serial.read(); //read incoming byte
168
169     /*
170      INL and AN1 refer to "motor left" which is actually the motors on the right side of the rover
171      IN2 and AN2 refer to "motor right" which is actually the motors on the left side of the rover
172      according to github code, "motor right" dir pin is HIGH when motors are forward and LOW when motors are reversed
173      according to github code, "motor left" dir pin is LOW when motors are forward and HIGH when motors are reversed
174      in practice I found that, because of an error in the ESC, both dir pins need to be HIGH when forward and LOW when reversed
175     */
176
177     if (inChar == 's'){
178         setSpeed = 40;
179         setSlowSide = setSpeed*0.6;
180         setFastSide = setSpeed*1.4;
181     }
182     else if (inChar == 'm'){
183         setSpeed = 70;
184         setSlowSide = setSpeed*0.6;
185         setFastSide = setSpeed*1.4;
186     }
187     else if (inChar == 'f'){
188         setSpeed = 100;
189         setSlowSide = setSpeed*0.6;
190         setFastSide = setSpeed*1.4;
191     }
192 }
193
194 /**
195  * reverse left
196  * if (inChar == '1') {
197  *     while(speedLeft > -setSlowSide && speedRight > -setFastSide){
198  *         speedLeft = speedLeft - 1; //this one is - 1 and Right is - 2 because the final value of Right is double Left
199  *         speedRight = speedRight - 2;
200  *         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
201  *         speedRightActual = speedRight*1.275;
202  *         motorControl(speedLeftActual, speedRightActual);
203  *         delay(10);
204  *         Serial.print(speedLeft);
205  *     }
206  *     speedLeft = setSlowSide;
207  *     speedRight = setFastSide;
208  *     speedLeftActual = speedLeft*1.275;
209  *     speedRightActual = speedRight*1.275;
210  *     motorControl(speedLeftActual, speedRightActual);
211  *
212  *     //reverse straight
213  *     else if (inChar == '2') {
214  *         while(speedLeft > -setSpeed && speedRight > -setSpeed){
215  *             speedLeft = speedLeft - 1;
216  *             speedRight = speedRight - 1;
217  *             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
218  *             speedRightActual = speedRight*1.275;
219  *             motorControl(speedLeftActual, speedRightActual);
220  *             delay(10);
221  *         }
222  *         speedLeft = -setSpeed;
223  *         speedRight = -setSpeed;
224  *         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
225  *         speedRightActual = speedRight*1.275;
226  *         motorControl(speedLeftActual, speedRightActual);
227  *
228  *     //reverse right
229  *     else if (inChar == '3') {
230  *         while(speedLeft > -setFastSide && speedRight > -setSlowSide){
231  *             speedLeft = speedLeft - 2;
232  *             speedRight = speedRight - 1;
233  *             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
234  *             speedRightActual = speedRight*1.275;
235  *             motorControl(speedLeftActual, speedRightActual);
236  *             delay(10);
237  *         }
238  *         speedLeft = -setFastSide;
239  *         speedRight = -setSlowSide;
240  *         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
241  *         speedRightActual = speedRight*1.275;
242  *         motorControl(speedLeftActual, speedRightActual);
243  *
244  *     //in place turn right
245  *     else if (inChar == '4') {
246  *         while(speedLeft < setSpeed && speedRight > -setSpeed){
247  *             speedLeft = speedLeft + 1;
248  *             speedRight = speedRight - 1;
249  *             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
250  *             speedRightActual = speedRight*1.275;
251  *             motorControl(speedLeftActual, speedRightActual);
252  *             delay(10);
253  *         }
254  *         speedLeft = setSpeed;
255  *         speedRight = -setSpeed;
256  *         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
257  *         speedRightActual = speedRight*1.275;
258  *         motorControl(speedLeftActual, speedRightActual);
259  *     }
260  *
261  */

```

```

262 | //forward right
263 | else if (inChar == '5') {
264 |     while(speedLeft < setFastSide && speedRight < setSlowSide){
265 |         speedLeft = speedLeft + 2;
266 |         speedRight = speedRight + 1;
267 |         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
268 |         speedRightActual = speedRight*1.275;
269 |         motorControl(speedLeftActual, speedRightActual);
270 |         delay(10);
271 |     }
272 |     speedLeft = setFastSide;
273 |     speedRight = setSlowSide;
274 |     speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
275 |     speedRightActual = speedRight*1.275;
276 |     motorControl(speedLeftActual, speedRightActual);
277 | }
278 |
279 //forward straight
280 else if (inChar == '6') {
281     while(speedLeft < setSpeed && speedRight < setSpeed){
282         speedLeft = speedLeft + 1;
283         speedRight = speedRight + 1;
284         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
285         speedRightActual = speedRight*1.275;
286         motorControl(speedLeftActual, speedRightActual);
287         delay(10);
288     }
289     speedLeft = setSpeed;
290     speedRight = setSpeed;
291     speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
292     speedRightActual = speedRight*1.275;
293     motorControl(speedLeftActual, speedRightActual);
294 }
295 |
296 //forward left turn
297 else if (inChar == '7') {
298     while(speedLeft < setSlowSide && speedRight < setFastSide){
299         speedLeft = speedLeft + 1;
300         speedRight = speedRight + 2;
301         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
302         speedRightActual = speedRight*1.275;
303         motorControl(speedLeftActual, speedRightActual);
304         delay(10);
305     }
306     speedLeft = setSlowSide;
307     speedRight = setSpeed;
308     speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
309     speedRightActual = speedRight*1.275;
310     motorControl(speedLeftActual, speedRightActual);
311 }
312 |
313 //in place turn left
314 else if (inChar == '8') {
315     while(speedLeft > -setSpeed && speedRight < setSpeed){
316         speedLeft = speedLeft - 1;
317         speedRight = speedRight + 1;
318         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
319         speedRightActual = speedRight*1.275;
320         motorControl(speedLeftActual, speedRightActual);
321     }
322     speedLeft = -setSpeed;
323     speedRight = setSpeed;
324     speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
325     speedRightActual = speedRight*1.275;
326     motorControl(speedLeftActual, speedRightActual);
327 }
328

```

```

---  

329 //emergency stop  

330 else if (inChar == '9') {  

331     //slowed stop for 1 L = -10 and R = -30  

332     if(speedLeft == -setSlowSide && speedRight == -setFastSide){  

333         while(speedLeft < 0 && speedRight < 0){  

334             speedLeft = speedLeft + 1;  

335             speedRight = speedRight + 2;  

336             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed  

337             speedRightActual = speedRight*1.275;  

338             motorControl(speedLeftActual, speedRightActual);  

339             delay(10);  

340         }  

341     }  

342  

343     //slowed stop for 2 L = -30 and R = -30  

344     else if(speedLeft == -setSpeed && speedRight == -setSpeed){  

345         while(speedLeft < 0 && speedRight < 0){  

346             speedLeft = speedLeft + 1;  

347             speedRight = speedRight + 1;  

348             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed  

349             speedRightActual = speedRight*1.275;  

350             motorControl(speedLeftActual, speedRightActual);  

351             delay(10);  

352         }  

353     }  

354  

355     //stop for 3 L = -30 and R = -10  

356     else if(speedLeft == -setFastSide && speedRight == -setSlowSide){  

357         while(speedLeft < 0 && speedRight < 0){  

358             speedLeft = speedLeft + 2;  

359             speedRight = speedRight + 1;  

360             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed  

361             speedRightActual = speedRight*1.275;  

362             motorControl(speedLeftActual, speedRightActual);  

363             delay(10);  

364         }  

365     }  

366  

367     //stop for 4 L = 30 and R = -30  

368     else if(speedLeft == setSpeed && speedRight == -setSpeed){  

369         while(speedLeft > 0 && speedRight < 0){  

370             speedLeft = speedLeft - 1;  

371             speedRight = speedRight + 1;  

372             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed  

373             speedRightActual = speedRight*1.275;  

374             motorControl(speedLeftActual, speedRightActual);  

375             delay(10);  

376         }  

377     }  

378  

379     //stop for 5 L = 30 and R = 10  

380     else if(speedLeft == setFastSide && speedRight == setSlowSide){  

381         while(speedLeft > 0 && speedRight > 0){  

382             speedLeft = speedLeft - 2;  

383             speedRight = speedRight - 1;  

384             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed  

385             speedRightActual = speedRight*1.275;  

386             motorControl(speedLeftActual, speedRightActual);  

387             delay(10);  

388         }  

389     }  

390  

391     //stop for 6 L = 30 and R = 30  

392     else if(speedLeft == setSpeed && speedRight == setSpeed){  

393         while(speedLeft > 0 && speedRight > 0){  

394             speedLeft = speedLeft - 1;  

395             speedRight = speedRight - 1;  

396             speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed  

397             speedRightActual = speedRight*1.275;  

398             motorControl(speedLeftActual, speedRightActual);  

399             delay(10);  

400         }  

401     }  

402 }

```

```

402 //stop for 7 L = 10 and R = 30
403 else if(speedLeft == setSlowSide && speedRight == setFastSide){
404     while(speedLeft > 0 && speedRight > 0){
405         speedLeft = speedLeft - 1;
406         speedRight = speedRight - 2;
407         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
408         speedRightActual = speedRight*1.275;
409         motorControl(speedLeftActual, speedRightActual);
410         delay(10);
411     }
412 }
413
414 //stop for 8 L = -30 R = 30
415 else if(speedLeft == -setSpeed && speedRight == setSpeed){
416     while(speedLeft < 0 && speedRight > 0){
417         speedLeft = speedLeft + 1;
418         speedRight = speedRight - 1;
419         speedLeftActual = speedLeft*1.275; //to convert the speed to a 0-255 scale, can't use map() because -100 speed would be mapped to 0 speed
420         speedRightActual = speedRight*1.275;
421         motorControl(speedLeftActual, speedRightActual);
422         delay(10);
423     }
424 }
425
426 speedLeftActual = 0;
427 speedRightActual = 0;
428 motorControl(speedLeftActual, speedRightActual);
429 }
430
431
432 //servo left
433 else if(inChar == 'L'){
434     while(angle < 120){
435         if(Serial.available() > 0){
436             inChar = Serial.read();
437             if(inChar == '9'){
438                 Serial.println("Break Left");
439                 Serial.println(inChar);
440                 break;
441             }
442         }
443         roverServo.write(angle);
444         angle++;
445         delay(30);
446         Serial.println("GO LEFT");
447     }
448     roverServo.write(angle);
449 }
450
451 //servo right
452 else if(inChar == 'R'){
453     while(angle > 0){
454         if(Serial.available() > 0){
455             inChar = Serial.read();
456             if(inChar == '9'){
457                 Serial.println("Break Right");
458                 break;
459             }
460         }
461         roverServo.write(angle);
462         angle--;
463         delay(30);
464         Serial.println("GO RIGHT");
465     }
466     roverServo.write(angle);
467 }
468 }
469 }
470

```

Flask Application Code (Python):

```

1  from flask import Flask, render_template, request, jsonify, Response
2  import serial
3  import time
4  import cv2
5  import logging
6
7  # Configure logging to file
8  logging.basicConfig(filename='processing_times.log', level=logging.INFO)
9
10 app = Flask(__name__)
11
12 # Setup the serial connection to the Arduino
13 # Replace '/dev/ttyACM0' with your Arduino's serial port
14 arduino = serial.Serial('/dev/ttyACM0', 9600, timeout=1)
15 time.sleep(1) # Allow time for the connection to establish
16
17 @app.route('/')
18 def home():
19     # Render a basic control panel page (HTML needs to be created in the templates folder)
20     return render_template('control_panel.html')
21
22 @app.route('/command', methods=['POST'])
23 def command():
24     received_time = time.time() # Timestamp when the request is received
25
26     data = request.get_json()
27     direction = data['direction']
28
29     # Send the command to the Arduino
30     arduino.write(direction.encode())
31
32     processed_time = time.time() # Timestamp after processing the command
33     processing_time = processed_time - received_time # Calculate the processing time
34
35     # Log the processing time
36     logging.info(f"{{time.strftime('%Y-%m-%d %H:%M:%S')}} - Processing Time: {processing_time:.6f} seconds - Command: {direction}")
37
38     # Return a success message to the client including the processing time
39     return jsonify({"message": "Command sent to the rover", "direction": direction, "processing_time": processing_time})
40
41 def generate_frames():
42     # Initialize the camera
43     cap = cv2.VideoCapture(-1) # Change '-1' to '0' or other depending on your camera setup
44
45     while True:
46         success, frame = cap.read()
47         if not success:
48             break
49         else:
50             # Convert the captured frame to JPEG
51             ret, buffer = cv2.imencode('.jpg', frame)
52             frame = buffer.tobytes()
53             # Yield the frame in multipart/x-mixed-replace format
54             yield (b'--frame\r\n'
55                   b'Content-Type: image/jpeg\r\n\r\n' + frame + b'\r\n')
56
57 @app.route('/video')
58 def video():
59     # Return the video stream response
60     return Response(generate_frames(), mimetype='multipart/x-mixed-replace; boundary=frame')
61
62 if __name__ == '__main__':
63     # Run the Flask application
64     app.run(host='0.0.0.0', port=5000, debug=True, threaded=False, processes=3)
65

```

Webserver Code (HTML):

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <title>Rover Control Panel</title>
6      <link rel="stylesheet" href="{{ url_for('static', filename='css/style.css') }}> <!-- Link to the CSS file -->
7      <script src="{{ url_for('static', filename='js/script.js') }}></script> <!-- Link to the JavaScript file -->
8  </head>
9  <body>
10     <div id="overlay-controls">
11         <!-- Mode Switch Container -->
12         <div class="mode-switch-container">
13             <div class="mode-switch" id="modeManual">Manual</div>
14             <div class="mode-switch" id="modeAuto">Autonomous</div>
15         </div>
16
17         <div id="direction-controls" class="grid grid-cols-3 gap-1">
18             <button class="dir-btn" data-direction="slightleft"></button>
19             <button class="dir-btn" data-direction="forward">Fwd</button>
20             <button class="dir-btn" data-direction="slightright"></button>
21             <button class="dir-btn" data-direction="left">L</button>
22             <button id="stop-btn" class="dir-btn" data-direction="stop">STOP</button>
23             <button class="dir-btn" data-direction="right">R</button>
24             <button class="dir-btn" data-direction="backleft"></button>
25             <button class="dir-btn" data-direction="backward">Rev</button>
26             <button class="dir-btn" data-direction="backright"></button>
27         </div>
28
29         <div id="servo-controls">
30             <div class="controls-label">Camera Controls</div>
31             <div class="servo-button-container">
32                 <button id="servo-left" class="servo-button" data-direction="s1">Left</button>
33                 <button id="servo-right" class="servo-button" data-direction="s2">Right</button>
34             </div>
35         </div>
36
37         <div id="video-container">
38             
39         </div>
40
41         <!-- Speed Control Slider -->
42         <div class="slider-container">
43             <input type="range" min="1" max="3" step="1" value="2" class="speed-slider" id="speedSlider">
44             <div class="labels">
45                 <span>Slow</span>
46                 <span>Medium</span>
47                 <span>Fast</span>
48             </div>
49         </div>
50     </div>
51 </body>
52 </html>
```

Webserver Code (CSS):

```
1  body, html {
2      margin: 0;
3      padding: 0;
4      width: 100%;
5      height: 100%;
6      overflow: hidden;
7      position: relative;
8      -webkit-user-select: none;
9      -moz-user-select: none;
10     -ms-user-select: none;
11     user-select: none;
12 }
13
14 #video-container {
15     position: absolute;
16     top: 0;
17     left: 0;
18     width: 100%;
19     height: 100%;
20     z-index: 1; /* Ensure it's below the controls but covers the whole background */
21 }
22
23 #camera-feed {
24     width: 100%;
25     height: 100%;
26     object-fit: cover; /* Ensure the video covers the full area without being stretched */
27     position: absolute;
28     top: 0;
29     left: 0;
30 }
31
32 #overlay-controls {
33     position: absolute;
34     top: 0;
35     left: 0;
36     width: 100%;
37     height: 100%;
38     display: flex;
39     flex-direction: column;
40     justify-content: flex-end; /* Aligns children (controls) at the bottom */
41     align-items: flex-end; /* Aligns children to the right */
42     z-index: 3;
43 }
44
```

```
45 #direction-controls {
46   display: grid;
47   grid-template-columns: repeat(3, 60px);
48   grid-gap: 10px;
49   padding: 20px; /* Adds some padding around the controls */
50   margin: 10px; /* Adds margin to avoid touching the screen edges */
51   border-radius: 10px; /* Optional: rounds the corners of the control panel */
52   z-index: 3;
53 }
54
55 .dir-btn {
56   width: 60px;
57   height: 60px;
58   background-color: #fff;
59   border: 2px solid #333;
60   transition: transform 0.1s;
61 }
62
63 .dir-btn:active {
64   transform: translateY(2px);
65   background-color: #ccc;
66 }
67
68 #stop-btn {
69   grid-column: 2;
70   grid-row: 2;
71   background-color: red;
72   color: white;
73 }
74
75 /* Servo button styling to match driving buttons */
76 #servo-controls {
77   position: absolute;
78   left: 30px; /* Adjust based on your layout preferences */
79   bottom: 30px; /* Adjust based on your layout preferences */
80   z-index: 3;
81   display: flex;
82   flex-direction: column; /* Stack vertically the label and the button container */
83   align-items: center; /* Center-align the contents */
84 }
85
```

```
86 .controls-label {
87   margin-bottom: 10px; /* Space between label and buttons */
88   color: ■#fff; /* Ensure label is visible against any background */
89   font-size: 16px; /* Adjust size as needed */
90   text-align: center; /* Ensures text is centered above buttons */
91 }
92
93 .servo-button-container {
94   display: flex;
95   justify-content: center; /* Center-align the buttons next to each other */
96 }
97
98 .servo-button {
99   width: 90px; /* Adjust width as needed */
100  height: 50px; /* Adjust height as needed */
101  margin: 0 5px; /* Space between buttons */
102  background-color: ■#fff;
103  border: 2px solid □#333;
104  transition: transform 0.1s;
105 }
106
107 .servo-button:active {
108   transform: translateY(2px);
109   background-color: ■#ccc;
110 }
111
112
113
114 /* Slider styling at the bottom middle of the screen */
115 .slider-container {
116   position: absolute;
117   width: 300px;
118   bottom: 20px;
119   left: 50%;
120   transform: translateX(-50%);
121   z-index: 3;
122 }
123
```

```
124 .speed-slider {  
125     -webkit-appearance: none;  
126     width: 100%;  
127     height: 15px;  
128     border-radius: 5px;  
129     background: #ddd;  
130     outline: none;  
131     opacity: 0.7;  
132     transition: opacity .2s;  
133 }  
134  
135 .speed-slider:hover {  
136     opacity: 1;  
137 }  
138  
139 .speed-slider::-webkit-slider-thumb {  
140     -webkit-appearance: none;  
141     appearance: none;  
142     width: 25px;  
143     height: 25px;  
144     border-radius: 50%;  
145     background: #461D7C;  
146     cursor: pointer;  
147 }  
148  
149 .speed-slider::-moz-range-thumb {  
150     width: 25px;  
151     height: 25px;  
152     border-radius: 50%;  
153     background: #461D7C;  
154     cursor: pointer;  
155 }  
156  
157 .labels {  
158     display: flex;  
159     justify-content: space-between;  
160     margin-top: 5px;  
161     font-size: 16px;  
162     color: #333;  
163 }  
164
```

```
165 .mode-switch-container {
166   position: absolute;
167   top: 20px;
168   right: 20px;
169   display: flex;
170   align-items: center;
171   z-index: 3;
172   font-size: 16px;
173 }
174
175 .mode-switch {
176   padding: 10px 20px;
177   cursor: pointer;
178   color: □#333; /* Dark color for visibility */
179   border: 2px solid □#461D7C;
180   margin-left: 5px;
181   transition: background-color 0.3s, color 0.3s;
182   user-select: none;
183   -webkit-user-select: none;
184   -moz-user-select: none;
185   -ms-user-select: none;
186 }
187
188 .mode-switch:first-child {
189   margin-left: 0;
190   border-right: 2px solid □#461D7C; /* Remove border between two buttons to appear as one unit */
191 }
192
193 .mode-switch.active {
194   background-color: □#461D7C;
195   color: ■#fff; /* White color for the active button text */
196 }
197
198 .mode-switch:not(.active) {
199   background-color: ■#fff; /* White background for inactive mode */
200   color: □#333; /* Dark text for readability */
201 }
202
```

Webserver Code (JavaScript):

```
1  document.addEventListener('DOMContentLoaded', () => [
2    // Handling directional, servo, and mode controls
3    const directionButtons = document.querySelectorAll('#direction-controls .dir-btn');
4    const servoButtons = document.querySelectorAll('.servo-button');
5    const speedSlider = document.getElementById('speedSlider');
6    const modeButtons = document.querySelectorAll('.mode-switch');
7
8    // Event listener for the mode switches
9    modeButtons.forEach(button => {
10      button.addEventListener('click', (event) => {
11        const mode = button.textContent.trim();
12        setMode(mode.toLowerCase());
13        const commandCode = mode === 'Manual' ? 'M' : 'A';
14        sendDirectionToServer(commandCode);
15      });
16    });
17
18    // Event listener for the speed slider
19    speedSlider.addEventListener('input', (event) => {
20      const speed = event.target.value;
21      const speedCommand = { '1': 's', '2': 'm', '3': 'f' }[speed];
22      sendDirectionToServer(speedCommand);
23    });
24
25    // Add event listeners for directional buttons
26    directionButtons.forEach(button => {
27      button.addEventListener('mousedown', (event) => {
28        const direction = button.getAttribute('data-direction') || 'stop';
29        const commandCode = translateDirectionToCommand(direction);
30        sendDirectionToServer(commandCode);
31      });
32
33      button.addEventListener('mouseup', (event) => {
34        sendDirectionToServer(translateDirectionToCommand('stop'));
35      });
36
37      button.addEventListener('touchstart', (event) => {
38        event.preventDefault(); // Prevent the touch event from triggering the mousedown event
39        const direction = button.getAttribute('data-direction') || 'stop';
40        const commandCode = translateDirectionToCommand(direction);
41        sendDirectionToServer(commandCode);
42      });
43    });
44  ])
```

```

43
44     button.addEventListener('touchend', (event) => {
45         sendDirectionToServer(translateDirectionToCommand('stop'));
46     });
47 };
48
49 // Add event listeners for servo buttons
50 servoButtons.forEach(button => {
51     button.addEventListener('mousedown', event => handleServoControl(button));
52     button.addEventListener('touchstart', event => {
53         event.preventDefault(); // Prevent the touch event from triggering the mousedown event
54         handleServoControl(button);
55     });
56
57     button.addEventListener('mouseup', event => sendDirectionToServer('9')); // Send stop command when button is released
58     button.addEventListener('touchend', event => sendDirectionToServer('9'));
59 });
60
61 function handleServoControl(button) {
62     const direction = button.getAttribute('data-direction');
63     const commandCode = translateDirectionToCommand(direction);
64     sendDirectionToServer(commandCode);
65 }
66
67 function translateDirectionToCommand(direction) {
68     switch (direction) {
69         case 'forward': return '6';
70         case 'backward': return '2';
71         case 'slightleft': return '7';
72         case 'slightright': return '5';
73         case 'backleft': return '1';
74         case 'backright': return '3';
75         case 'left': return '8';
76         case 'right': return '4';
77         case 's1': return 'L';
78         case 's2': return 'R';
79         case 'stop': default: return '9'; // Command to stop the movement
80     }
81 }
82
83 function setMode(mode) {
84     modeButtons.forEach(button => {
85         if (button.textContent.trim().toLowerCase() === mode) {
86             button.classList.add('active');
87         } else {
88             button.classList.remove('active');
89         }
90     });
91 }
92
93 function sendDirectionToServer(commandCode) {
94     fetch('/command', {
95         method: 'POST',
96         headers: { 'Content-Type': 'application/json' },
97         body: JSON.stringify({ direction: commandCode }),
98     })
99     .then(response => response.json())
100    .then(data => console.log('Success:', data))
101    .catch(error => console.error('Error:', error));
102 }
103
104

```

XIII.G.9. Testing & Validation of Qualitative Constraint Q1 - Portability

[All]

[]

XIII.G.9.a. Test Protocol Description - Q1

At several points of the semester when the rover was unable to drive and/or driving to the location was not convenient, team members manually moved the rover. This was done by lifting for short distances and by pickup truck for long distances.

[]

XIII.G.9.b. Equipment and Instrumentation - Q1

N/A

[]

XIII.G.9.c. Data Acquisition & Analysis – Q1

N/A

[]

XIII.G.9.d. Results Details – Q1

The rover was found to be possible to be moved by two people for short distances, and easy for four people. The frame has several points to grab when carrying conveniently. The rover was also found to be easy to move in a pickup truck as the frame has several points where ratchet straps attach easily. In cases of moving through narrow spaces (IE doors) the legs are detachable.



Figure XIII-16: Rover being carried by two people. Figure XIII-17: Rover secured in pickup truck.

XIII.G.10. Testing & Validation of Qualitative Constraint Q2 - Durability

Kyle Pitre (ME)

XIII.G.10.a. Test Protocol Description - Q#

See the strain gauge testing section for the strain gauge setup.

The impact analysis was performed by conducting two sets of drop tests. We printed extra platforms to perform drop tests to compare to the results to the impact analysis. One of the platforms had an extra motor controller from last year's team velcroed to it in order to test the point where these would detach, and the second had a 2.5 pound weight representing the weight of all the electronics on a single platform attached to it to determine when it would break through the platform. We dropped the velcroed platform vertically as in the analysis as this would be the orientation in which the velcro would detach first. The second platform was dropped with the platform horizontal to since this would force the weight to fall through the middle. Both platforms were secured to a spare frame section to accurately resemble how the platforms would be fit onto the rover. We dropped the platforms in their mock up frames at increasing heights until the point of failure

After the conclusion of testing, the rover was inspected for damage and wear.

XIII.G.10.b. Equipment and Instrumentation - Q#

Impact test:

Measuring tape, phone for videoing, timer

Inspection

Visual inspection. Pictures taken when possible.

XIII.G.10.c. Data Acquisition & Analysis – Q#

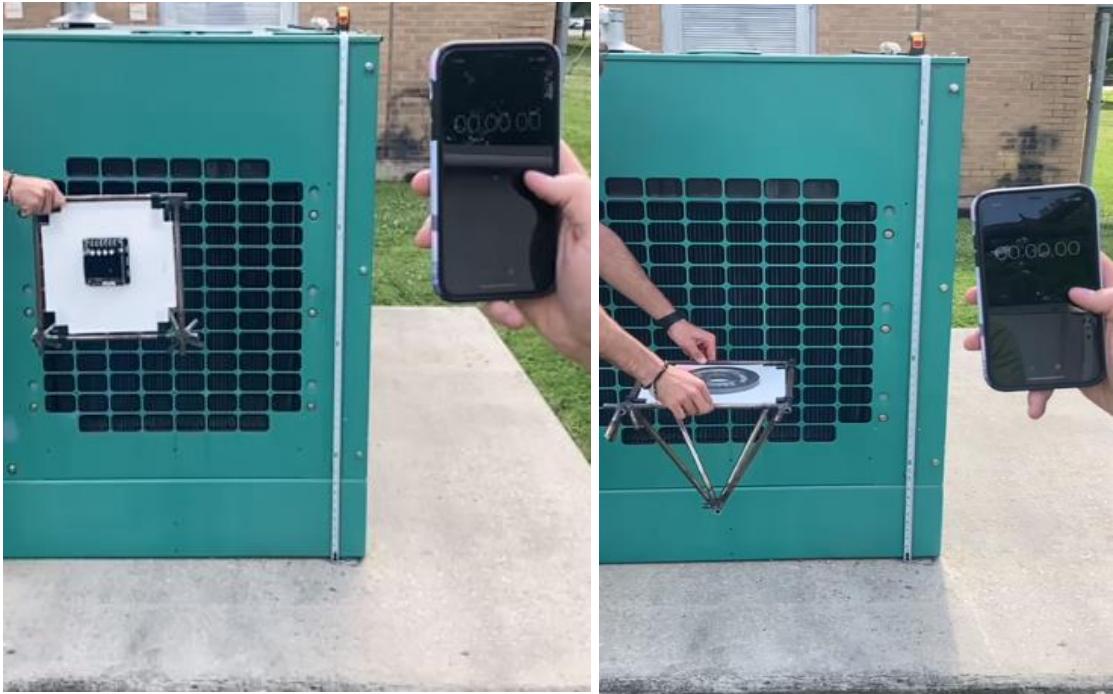
Impact test:

The drops were videoed to have a reference of the drop height for failure. This was also used to calculate the velocity at impact.

XIII.G.10.d. Results Details – Q#

Impact test:

For both sets of tests, similar results were found with a maximum drop height of 1.5 feet. This is a low distance, but this is again in the worst possible orientations that the rover is not likely to experience due to the rocker-bogie suspensions system ensuring there is always contact with the ground to reduce the chances of free falling. The drops were videoed to be analyzed and find the maximum impact speed to be 8 ft/s, which is above the maximum speed of the rover so running into objects should not bring upon issues.



Inspection

Overall, the rover is in good shape with all components still functioning as desired. However, there are a few things to note. There is some wear on the inside of the wheel where it contacts the axle. There is no indication that this has affected rover performance. The center electronics platform is warping slightly down, although it is still stable. This could be easily replaced with a stiffer platform if it becomes an issue. A notable amount of wear and abrasion was found on the driveshaft that was inspected (shown below). This is likely due to some contact between the axle and the driveshaft. Most of the wear is on the key which appears to have not been hammered down far enough during assembly. This should not continue to be an issue once the part has been worn past the point of contact. The driveshaft remained secure to the motor and did not exhibit significant warping or looseness. It was also noted that there could be a potential fatigue issue. Since not all of the axles are welded on perfectly straight, some of the driveshafts get pulled off center by the wheel. This introduces more force than was expected onto the back of the driveshaft. Since this force cycles every time the wheel turns, it could introduce loading fatigue to the part. To be safe, this component should be inspected regularly to see if this develops any further.



Figure XIII-18: Steel dust from grinding between keyway/driveshaft and wheel axle.



Figure XIII-19: Steel dust from grinding between keyway/driveshaft and wheel axle.

XIII.G.11. Testing & Validation of Qualitative Constraint Q3 – Ease of Operation

Creighton Cocathey (EEC), Tristan Hughes (ME)

XIII.G.11.a. Test Protocol Description – Q3

An anonymous poll was conducted.

XIII.G.11.b. Equipment and Instrumentation – Q3

N/A

XIII.G.11.c. Data Acquisition & Analysis – Q3

Google forms.

XIII.G.11.d. Results Details – Q3

The poll had an average of 9.1.

How intuitive are the controls for the rover?

24 responses

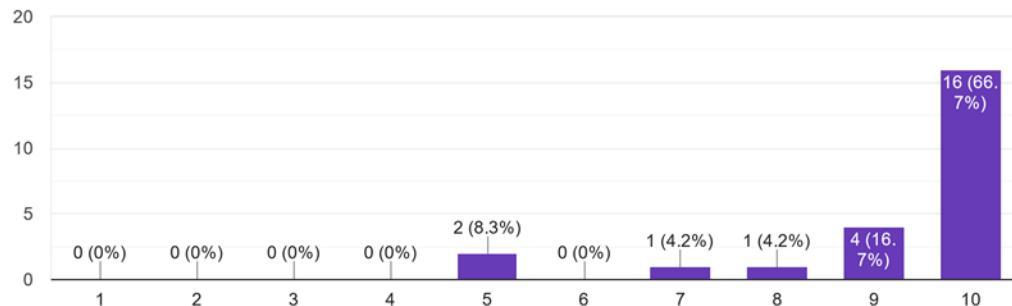


Figure XIII-20: Test Results

XIII.G.12. Testing & Validation of Qualitative Constraint Q4 – Aesthetically Pleasing

Tristan Hughes (ME)

XIII.G.12.a. Test Protocol Description – Q4

Anonymous poll.

XIII.G.12.b. Equipment and Instrumentation – Q4

N/A

XIII.G.12.c. Data Acquisition & Analysis – Q4

Google Forms.

XIII.G.12.d. Results Details – Q4

How aesthetically pleasing would you say the rover is?

24 responses

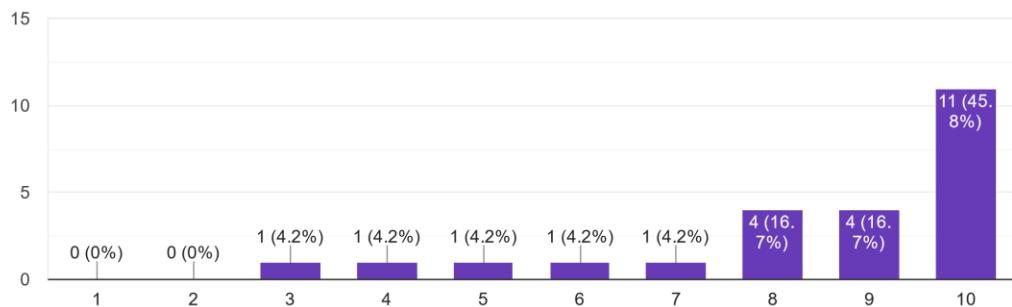


Figure XIII-10: Test Results

XIII.G.13. Testing & Validation of Quantitative Constraint M1 - Weight

[Tristan Hughes (ME).]

XIII.G.13.a. Test Protocol Description - M1

Weighing the rover.

XIII.G.13.b. Equipment and Instrumentation - M1

XIII.G.13.c. Data Acquisition & Analysis – M1

N/A

XIII.G.13.d. *Results Details – M1*

The rover was found to be 77.5 kg once fully assembled.

|

XIII.G.14. Testing & Validation of Quantitative Constraint M2 - Latency

Creighton Cocathey (EEC)

[]

XIII.G.14.a. *Test Protocol Description - M2*

The latency testing protocol was designed to measure the total communication system latency, which is critical for ensuring the responsiveness of the rover's control system. The test involved assessing individual components of the system to quantify their respective latencies, which were then summed to determine the overall system latency. Key components tested included network latency via Wi-Fi, web server processing time, and serial communication latency between the Jetson Nano and the Arduino. Each element was tested under consistent conditions to ensure accuracy, with multiple runs conducted to obtain average values. This approach allowed for a detailed breakdown of latency sources within the system, aiding in optimization and ensuring that the system stayed within the desired performance threshold.

|

XIII.G.14.b. *Equipment and Instrumentation - M2*

The equipment used for the latency tests included a MacBook Air (M2, 2022) to establish a network connection, the web server configured on a suitable platform for processing commands, and the Jetson Nano and Arduino set up for serial communication. The testing utilized software tools capable of precise timestamping to measure the time each component of the system responds. Network latency was measured using standard ping tests from the connected device to the Jetson Nano. The web server processing time was recorded by timestamping command receipts and completions. For serial communication, specialized scripts were used to send and receive test signals between the Jetson Nano and Arduino, with timestamps marking each transmission and receipt.

|

XIII.G.14.c. *Data Acquisition & Analysis – M2*

Data collection involved recording the latency times for each component during multiple test runs to ensure consistency and reliability of results. Network latency was measured using the average response time from numerous Wi-Fi ping tests. Web server processing latency was determined by averaging the time differences between receiving a command and completing its processing. Serial communication latency data were collected by measuring the roundtrip times for signals sent from the Jetson Nano to the Arduino and back. All data points were logged and analyzed statistically to compute average latencies for each component. These values were then summed to provide the total system latency. The analysis focused on identifying any outliers or variations in latency times and determining their causes to optimize system performance. The

final reported average latency of 22.27 ms, compared to the upper limit of 50 ms, demonstrated the system's efficiency and responsiveness within the set constraints. Each test was run 50 times.

```
22 @app.route('/command', methods=['POST'])
23 def command():
24     received_time = time.time() # Timestamp when the request is received
25
26     data = request.get_json()
27     direction = data['direction']
28
29     # Send the command to the Arduino
30     arduino.write(direction.encode())
31
32     processed_time = time.time() # Timestamp after processing the command
33     processing_time = processed_time - received_time # Calculate the processing time
34
35     # Log the processing time
36     logging.info(f"{time.strftime('%Y-%m-%d %H:%M:%S')} - Processing Time: {processing_time:.6f} seconds - Command: {direction}")
37
38     # Return a success message to the client including the processing time
39     return jsonify({"message": "Command sent to the rover", "direction": direction, "processing_time": processing_time})
40
```

XIII.G.14.d. Results Details – M2

The comprehensive analysis of the latency tests provided detailed insights into the performance of each component within the communication system. The network latency was found to be exceptionally low at an average of 4.96 milliseconds, indicating a highly efficient network setup with minimal traffic and strong signal quality, which is crucial for maintaining quick communication links. The web server processing time was also notably quick, averaging only 1.23 milliseconds. This rapid processing speed underscores the effectiveness of our server configuration and the optimized command processing scripts, ensuring that commands are executed with minimal delay. The most significant latency came from the serial communication between the Jetson Nano and Arduino, which recorded an average roundtrip time of 16.08 milliseconds. This latency is higher due to the inherent processing demands and the physical limitations of serial data transmission. Despite this, the sum of these individual latency measurements yielded a total system latency of 22.27 milliseconds, which is well under our upper limit of 50 milliseconds. This result confirms that our communication system operates efficiently and meets the project's stringent responsiveness criteria, ensuring that the rover can perform its tasks reliably under operational conditions. |

Network Latency	Average Network Latency	Webserver Latency	Average Webserver Latency	Serial Latency	Average Serial Latency	Average Latency
4.904	4.96	1.26	1.23	16.76	16.08	22.27
4.429		1.11		14.24		
4.141		1.34		16.8		
4.275		1.21		16.48		
3.356		1.29		16.45		
5.87		1.14		16.81		
3.385		1.18		17.28		
3.733		1.23		16.04		
3.54		1.16		16.11		
3.982		1.45		16.56		
6.176		1.24		16.05		
4.66		1.22		16.15		
5.369		1.15		15.96		
5.016		1.33		16.15		
5.468		1.16		16.23		
4.452		1.25		16.61		
5.204		1.33		15.95		
5.088		1.24		16.9		
4.677		1.23		16.55		
3.125		1.28		16.25		
6.116		1.18		15.76		
10.011		1.2		15.14		
7.058		1.32		16.57		
6.241		1.22		15.43		
6.647		1.31		16.86		
12.788		1.24		15.02		
7.528		1.26		16.4		
5.325		1.31		15.96		
6.092		1.14		17.1		
5.394		1.11		16.57		
3.272		1.18		17.3		
3.265		1.15		13.35		
4.04		1.26		12.24		
5.333		1.3		16.37		
3.399		1.16		16.04		
3.533		1.15		13.06		
4.203		1.21		16.28		
3.262		1.24		16.54		
7.723		1.16		17.08		
4.502		1.15		16.86		
4.395		1.24		16.25		
4.414		1.31		15.65		
4.537		1.15		16.44		
3.999		1.22		16.58		
4.282		1.47		16.06		
3.78		1.16		16.19		
4.224		1.16		16.16		
4.086		1.27		16.28		
2.741		1.39		16.25		

XIII.G.14.e.

XIII.G.15. Testing & Validation of Quantitative Constraint M3 - Budget

Kyle Pitre (ME)

XIII.G.15.a. Test Protocol Description – M3

The budget for this project was set at \$6,000. The purchases for all components, manufacturing, and testing were tracked to ensure this maximum cost was not exceeded.

XIII.G.15.b. Equipment and Instrumentation – M3

N/A

XIII.G.15.c. Data Acquisition & Analysis – M3

All purchases were tracked throughout the duration of the project in an Excel sheet.

XIII.G.15.d. Results Details – M3

Table XIII-22: Bill of Materials

Subsystem	Components	Quantity	Cost/Item	Total Cost	Manufacturer

Frame	Rust Paint	1	19.99	19.99	Amazon
	Velcro Squares	2	3.12	6.24	Amazon
	Electrical Tape	1	1.23	1.23	Amazon
Legs	Rocker	1	49.8	49.8	Grainger
	Bogie	2	30.25	60.5	Grainger
	Rocker Axle	1	73.7	73.7	Grainger
	Bogie Axle	1	23.72	23.72	Grainger
	Leg Bushing	1	44.16	44.16	Grainger
	Leg Bolts	1	13.49	13.49	Amazon
Drivetrain	Wheels	3	60.6	181.8	Amazon
	Wheel Axle	1	52.24	52.24	Grainger
	Wheel Hub	1	20.72	20.72	Grainger
	Motor Extension	1	24.59	24.59	Grainger
	Motor Attachment	6	15.3	91.8	Grainger
	Wheel Bolts	1	49.99	49.99	Amazon
	Wheel Nuts	1	9.96	9.96	Amazon
	Motor Extension Set Screw (1)	1	10.4	10.4	Grainger
	Set Screw	1	10.4	10.4	Grainger
	Wheel Retaining Rings	1	5.61	5.61	McMaster
	Motor Extension Set Screw (2)	1	10.99	10.99	Amazon
	Motor Housing Bolts	1	8.99	8.99	Amazon
Suspension	Axle Bearing	1	20.53	20.53	Grainger
	Axle Brackets	1	34.96	34.96	Grainger
	Differential Bar	1	54.48	54.48	Grainger
	Differential Strut	2	5.96	11.92	Grainger
	1.5" Retaining Rings	1	11.38	11.38	McMaster
	0.5" Retaining Rings	1	8.41	8.41	McMaster
	Rod End Joint	2	21.55	43.1	McMaster
	Nuts & Bolts	1	9.99	9.99	Amazon
Electric Propulsion	Motor	6	246	1476	Powerhouse Engineering
	Motor Housing	1	188.32	188.32	Grainger
	Bolts	1	15.35	15.35	Grainger
Power Supply	Battery	1	999	999	Enduro Power
	Battery Charger	1	249	249	Enduro Power
Power Distribution	12 to 3V Converter	1	5.09	5.09	AliExpress
	36/48V to 24V 30A Voltage Converter	3	49.99	149.97	Amazon
	36/48V to 12V Voltage Converter	1	18.99	18.99	Amazon

	12 to 5V Voltage Converter	1	14.99	14.99	Super Bright
	Power Distribution Bar	3	27.11	81.33	McMaster
	ELEGOO Multicolored Dupont Wire	1	6.98	6.98	Amazon
	12 AWG Wire	1	10.98	10.98	Amazon
	24 AWG Wire	1	6.28	6.28	Amazon
	4 AWG Wire (black)	10	1.32	13.2	Wire & Cable
	4 AWG Wire (red)	10	1.32	13.2	Wire & Cable
	8 AWG Wire (black)	45	0.56	25.2	Wire & Cable
	8 AWG Wire (red)	45	0.56	25.2	Wire & Cable
	4 Pin Buckled Grove 20 cm Cable	1	12.99	12.99	Amazon
	Inline Fuse Holder	3	9.99	29.97	Amazon
	25A Maxi Blade Fuse	6	2.96	17.76	McMaster
	Circuit Breaker	1	23.99	23.99	Amazon
Controller	Micro-Controller	1	52	52	Amazon
	Motor Controller	1	79	79	Maker Motors
	Rail Mount	1	35	35	Amazon
	Jetson Nano	1	149	149	Amazon
	Miuzei 20 kg Servo Motor	1	13.59	13.59	Amazon
	Servo Motor	1	7.99	7.99	Amazon
	#8-32 1/4" Machine Screw	1	5.9	5.9	Amazon
Object Detection/Reaction	Ultrasonic Distance Sensor	1	3.95	3.95	Adafruit
	Camera	1	57.99	57.99	Amazon
Communication	Temperature Sensors	4	9.95	39.8	Adafruit
	Gyroscope/Accelerometer	1	34.95	34.95	Adafruit
	Wi-Fi Module	1	23.99	23.99	Amazon
Manufacturing & Testing	Waterjet Manufacturing (1)	1	97	97	AMMF
	Weld Testing	1	41	41	Amazon
	Waterjet Manufacturing (2)	1	39.5	39.5	AMMF
	Strain Gauge Testing	1	80	80	MMTEF
Total (with shipping)		\$5606.76			

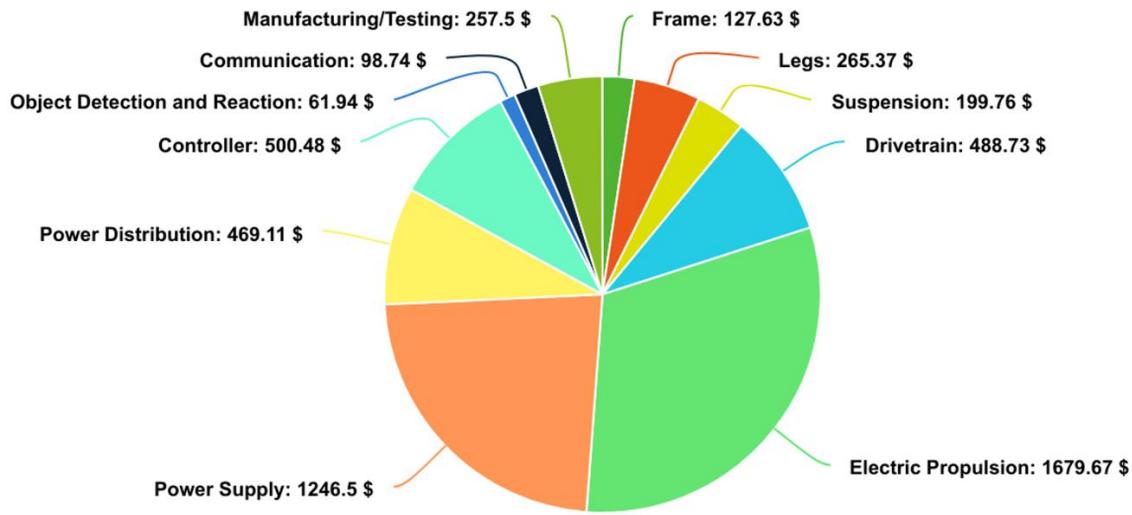


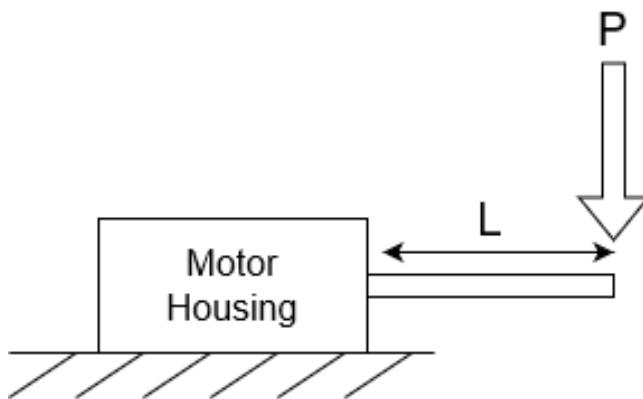
Figure XIII-21: Test Results

XIII.G.16. Testing & Validation of Quantitative Constraint M4 – Carrying Capacity

Kyle Pitre (ME)

XIII.G.16.a. Test Protocol Description – M4

Strain gauge tests were performed on the wheel axles prior to assembly to ensure these components can withstand the expected loads. To do this, the motor housing was secured to a table with the wheel axle welded to it to resemble how this component is assembled to the rover. Strain gauges were placed in the center of the rod where the maximum stresses and strains are expected. Weights were placed on the free end of the rod in increments of 5 lbs. up to the maximum expected load of 50 lbs. in order to obtain a sufficient sample size. This weight includes the weight of the rover as well as the additional weight expected to be added by future Capstone teams. A picture of the setup is shown below.



XIII.G.16.b. Equipment and Instrumentation – M4

- Clamps

- Strain Gauges
- P3 Strain Indicator and Recorder
- Weight
- Wheel Axle

XIII.G.16.c. Data Acquisition & Analysis – M4

The P3 strain indicator and recorder measures the strains in units of micro-strain. This was used to calculate the stress for each data set using Hooke's law:

$$\sigma = E\varepsilon$$

Here, the modulus of elasticity (E) was 30 million psi as this part is made from steel. The deflection of the rod was then found using the relation:

$$\delta = \frac{\sigma L^2}{3Ec}$$

XIII.G.16.d. Results Details – M4

Table XIII-23: Test Result Data Table

Weight	Strain	Stress (psi)	Deflection (in)
0	0.0000	280	0.001
5	0.0001	1959	0.005
10	0.0001	2799	0.008
15	0.0001	3799	0.010
20	0.0002	6438	0.018
30	0.0003	7837	0.021
35	0.0003	8677	0.024
40	0.0004	9797	0.027
50	0.0005	12596	0.034

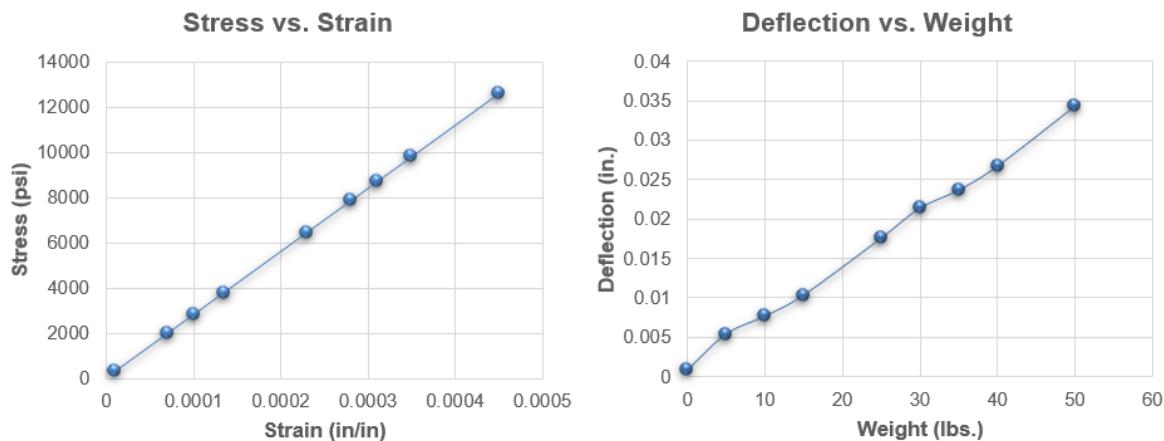


Figure XIII-22: Test Results

XIII.G.17. Testing & Validation of Quantitative Constraint M5 - Speed

Michael Cannon (ME), Patrick Maloney (EE)

XIII.G.17.a. *Test Protocol Description – M5*

For test 1 and 2, the rover was taken to an open hallway in the AMMF. Marks were set at 10 ft intervals with tape. The terrain for this test was smooth concrete. The rover was placed several feet behind the line so that it would have time to accelerate to full speed. It was then directed to drive forward. These tests were done with motor speed set to 50% of the maximum rated speed. Test 1 covered a distance of 20 ft, while test 2 covered a distance of 30 ft.

XIII.G.17.b. *Equipment and Instrumentation – M5*

- Camera.
- Measuring tape
- Video editing software (iMovie)

XIII.G.17.c. *Data Acquisition & Analysis – M5*

The tests were filmed by a team member. The footage was uploaded to a video editing software. For tests 1 and 2, the videos were edited down to start and end on the exact frame where the rover crossed the start and finish lines. This allowed for accurate drive times to be combined with the known drive distances to output rover speed.

Tests 3 and 4 were conducted on rough concrete and grass respectively. For these tests the speed was calculated in a similar manner to the terrain tests by using the editing software to track wheel rotations and find drive time. (See XII.G.1.c)

XIII.G.17.d. *Results Details – M5*

These results show that the rover is capable of surpassing the 1.25 m/s speed target. Since the motors were only set to 50%, it is possible to get even greater speed. However, these higher speeds are difficult for the operator to work with and are therefore unpractical.

Table X-13: Speed test results

Trial #	Terrain	Average Speed (m/s)
1	Smooth Concrete	1.39
2	Smooth Concrete	1.33
3	Rough Concrete	1.35
4	Grass	1.35
Average	-	1.355



Figure XIII-23: Rover during speed test 2.

XIII.G.18. Testing & Validation of Quantitative Constraint M6 - Dimensions

Tristan Hughes (ME)

XIII.G.18.a. Test Protocol Description – M6

Measure final dimensions of rover.

XIII.G.18.b. Equipment and Instrumentation – M6

Measuring tape

XIII.G.18.c. Data Acquisition & Analysis – M6

N/A

XIII.G.18.d. Results Details – M6

Final dimensions were: $1.25 \times 1.25 \times 0.625\text{m}^3$

XIII.G.19. Testing & Validation of Quantitative Constraint M7 – Operation Time

Ethernan Smith (EE)

XIII.G.19.a. *Test Protocol Description – M7*

Testing of the battery involved running the full rover and all its electronics for as long as possible on various terrains. Throughout these tests, the battery system was monitored using the Enduro Power app via Bluetooth. The application's interface is depicted in Figure XIII12, providing users with real-time information such as current drawn from the battery, battery voltage, battery temperature, and remaining state of charge. These values were recorded during testing and later analyzed in comparison to predictions from the Simulink models, as described in the Engineering Analysis section for SS6 – Power Supply.

XIII.G.19.b. *Equipment and Instrumentation – M7*

- ‘Enduro Power’ App

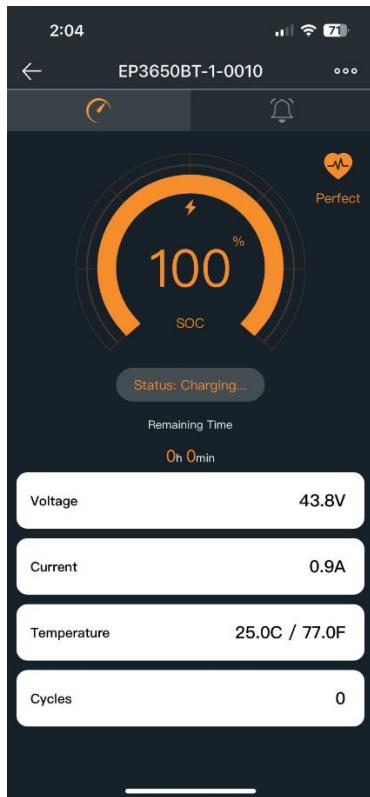


Figure XIII12: ‘Enduro Power’ Application Interface

XIII.G.19.c. *Data Acquisition & Analysis – M7*

The data was recorded manually using pen and paper. The data recorder documented various parameters including terrain type, slope, battery voltage, operating current range, duration of time on each terrain, initial state of charge upon beginning traversal, and final state of charge upon completion. Additionally, specific movements such as turning in place or turning right were recorded along with the corresponding battery currents. Testing encompassed a variety of terrains as per the sponsor's request for versatility. For analysis, the recorded current ranges average currents were inputted into our Simulink simulation to compare the battery's performance with the expected behavior of a lithium-ion battery, ensuring that the state of charge depletion aligns with

expectations. The final state of charge should ideally fall within a 3% margin of the anticipated value to validate the battery selection.

XIII.G.19.d. Results Details – M7

Three separate extended tests were conducted, each terminated prematurely for various reasons. Although none of the tests individually lasted a full hour as initially planned, the battery was only briefly charged once between the second and third tests. Consequently, these tests were combined into one large test to provide insight into the battery's behavior over the total duration. The combined and individual test results are presented below:

Combined Testing Results:

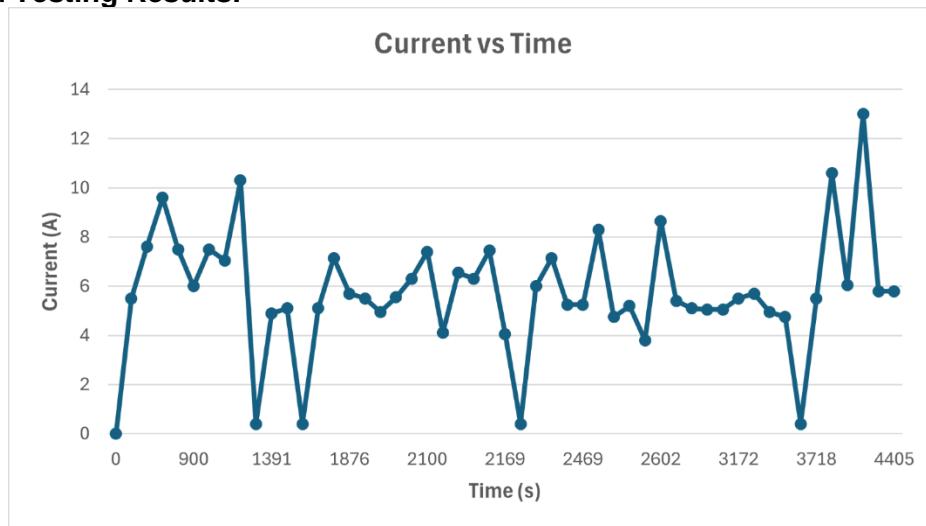


Figure XIII12: Combined Extended Tests Current Throughout

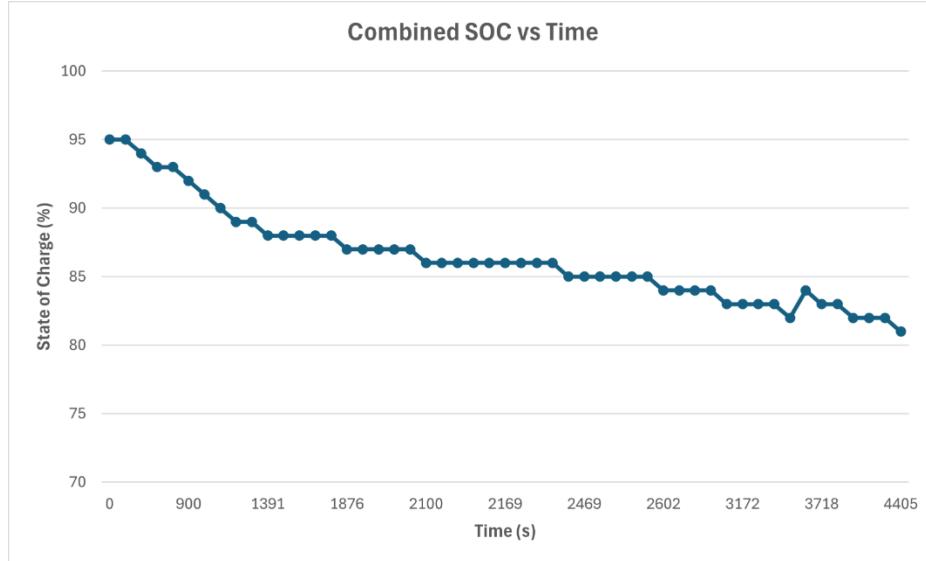


Figure XIII13: Combined Extended Tests State of Charge

The battery was only charged once – briefly between the second and third test, hence the increase in SOC around 3400s

Result of Combined Testing: After 4405s of total operation (73 minutes) the SOC dropped from 95 to 81

Individual Testing Results:

Extended Test One:

SOC (I)	SOC (F)	Time (start, s)	Time (end, s)	Time (s)	Current (average range, A)	Average Current	Voltage (V)	Terrain Description	Slope (degree)	Notes
89	89	0	0	0	0	0	39.9			(filler line for graph)
89	89	0	30	30	0.4	0.4	39.9			Idling
89	88	31	131	101	4.7-5.1	4.9	39.6	Concrete	0	Navigating shop
88	88	132	311	180	4.8-5.4	5.1	39.5	Outside Concrete		
88	88	312	568	257	0.4	0.4	39.7	Concrete	0	Idling
88	88	569	578	10	4.9-5.3	5.1	39.6	Road		
88	88	579	586	8	6-8.3	7.15	39.7	Concrete	5	
88	87	587	616	30	5.7	5.7	39.6	Grass	0	
87	87	617	762	146	5.1-5.9	5.5	39.7	Grass	0	
87	87	763	812	50	4.5-5.4	4.95	39.7	Grass	0	
87	87	813	822	10	5.2-5.9	5.55	39.8	Grass	5	Uphill
87	87	823	832	10	6.3	6.3	39.6	Sand	3	Uphill
87	86	833	840	8	7.0-7.8	7.4	39.7	Grass	10	Uphill
86	86	841	860	20	3.6-4.6	4.1	39.8	Grass	10	Downhill
86	86	861	875	15	6.0-7.1	6.55	39.5	Mulch	0	
86	86	876	895	20	6-6.6	6.3	39.5	Grass	7	Uphill
86	86	896	899	4	6.6-8.3	7.45	39.7	Grass	2	Driving over cones
86	86	900	909	10	3.9-4.2	4.05	39.7	Grass	4	Downhill in reverse
86	86	910	1029	120	0.4	0.4	39.8	Grass		Idling
86	86	1030	1149	120	5.4-6.6	6	39.7	Grass	3	Nonstop driving uphill

Figure XIII14: Extended Test One - Description (1)

86	86	1150	1159	10	7-7.3	7.15	39.7	Grass	3	Driving downhill
86	85	1160	1189	30	5.1-5.4	5.25	39.7	Grass	3	Downhill Reverse
85	85	1190	1209	20	4.8-5.7	5.25	39.8	Grass	3	Downhill Reverse
85	85	1210	1224	15	7.9-8.7	8.3	39.7	Grass	0	Right reverse turn
85	85	1225	1284	60	4.5-5	4.75	39.7	Grass	2	Downhill
85	85	1285	1314	30	5-5.4	5.2	39.6	Gravel		
85	85	1315	1334	20	3.7-3.9	3.8	39.7	Concrete	4	Downhill
85	84	1335	1342	8	8.3-9	8.65	39.8	Concrete	0	Turning
84	84	1343	1462	120	4.9-5.9	5.4	39.7	Concrete	0	
84	84	1463	1612	150	5-5.2	5.1	39.8	Concrete	0	
84	84	1613	1642	30	4.6-5.5	5.05	39.8	Concrete	0	
84	83	1643	1762	120	4.7-5.4	5.05	39.7	Concrete	0	
83	83	1763	1912	150	5.2-5.8	5.5	39.7	Concrete	0	
83	83	1913	1932	20	5.4-6	5.7	39.8	Grass	0	
83	83	1933	2112	180	4.5-5.4	4.95	39.8	Concrete	0	
83	82	2113	2232	120	4.2-5.3	4.75	39.7	Concrete	0	Navigating shop
Final Results										
SOC I	SOC F			Tot Time(s)	Tot Time(m)					
89	82			2232	37.2					

Figure XIII15: Extended Test One - Description (2)

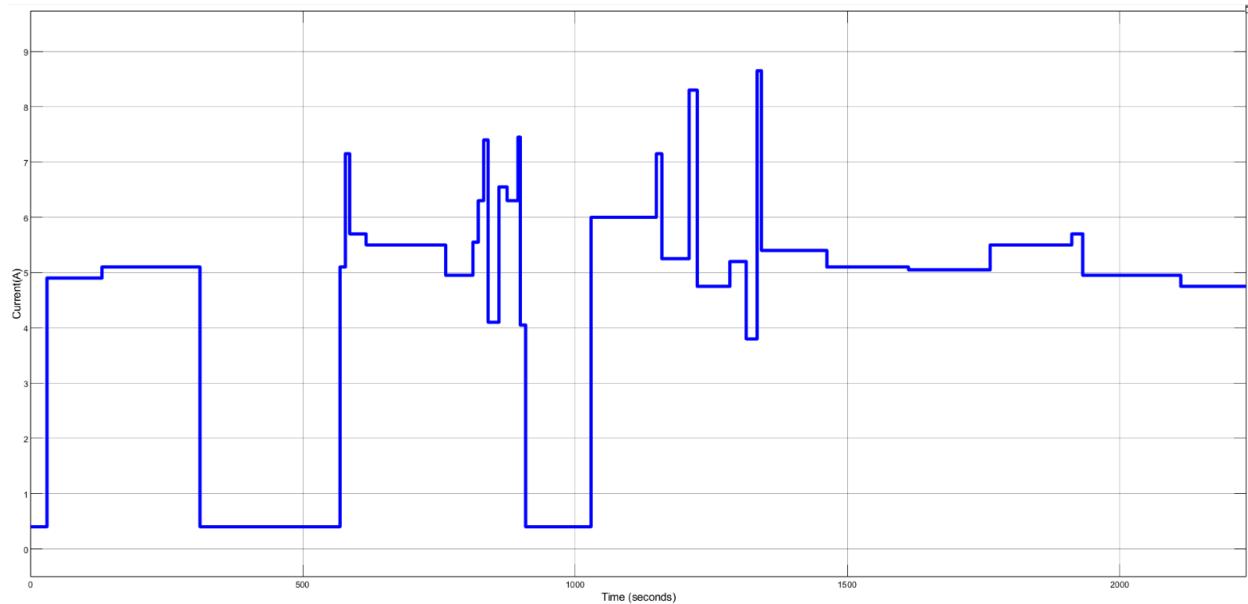


Figure XIII16: Extended Test One - Current Drawn from Battery

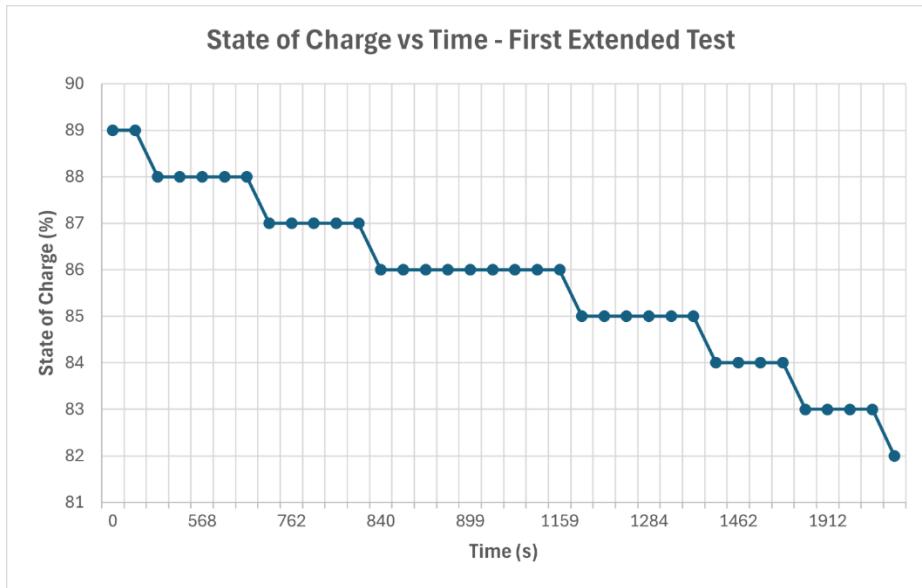


Figure XIII17: Extended Test One - State of Charge of Battery

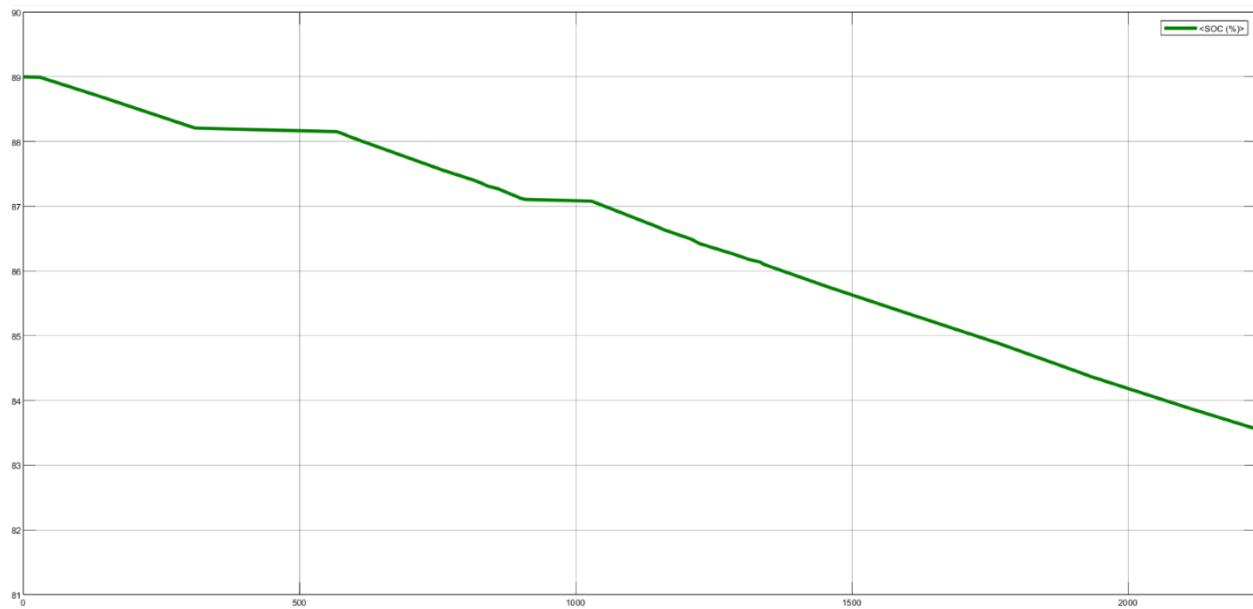


Figure XIII18: Extended Test One – Simulink Expected State of Charge of Battery

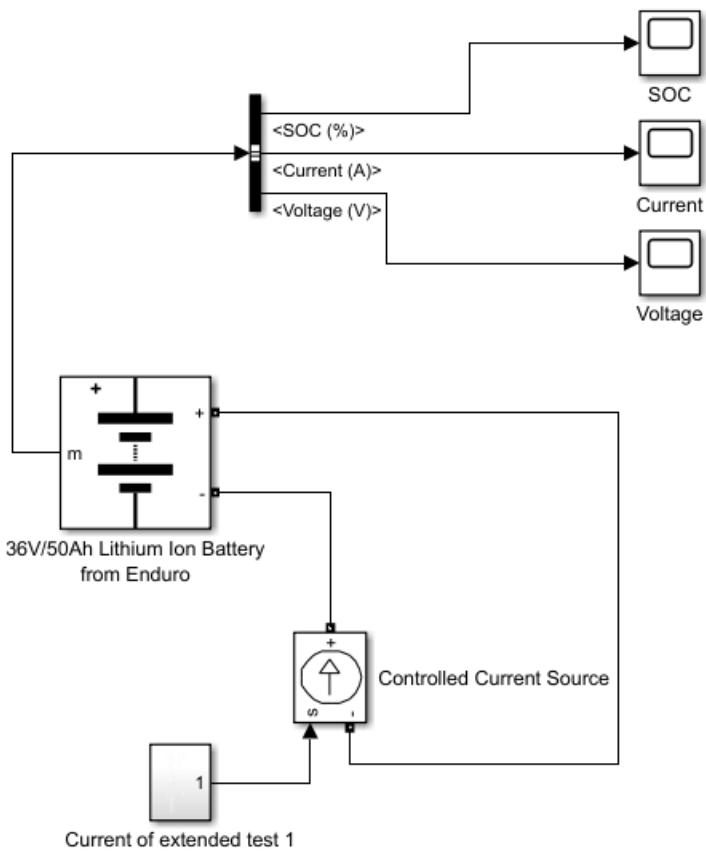


Figure XIII19: Simulink Battery Model Template

The above template was adjusted according to the test, the controlled current source's input was the current experienced while testing, and the initial state of charge was adjusted to match what we experienced when testing.

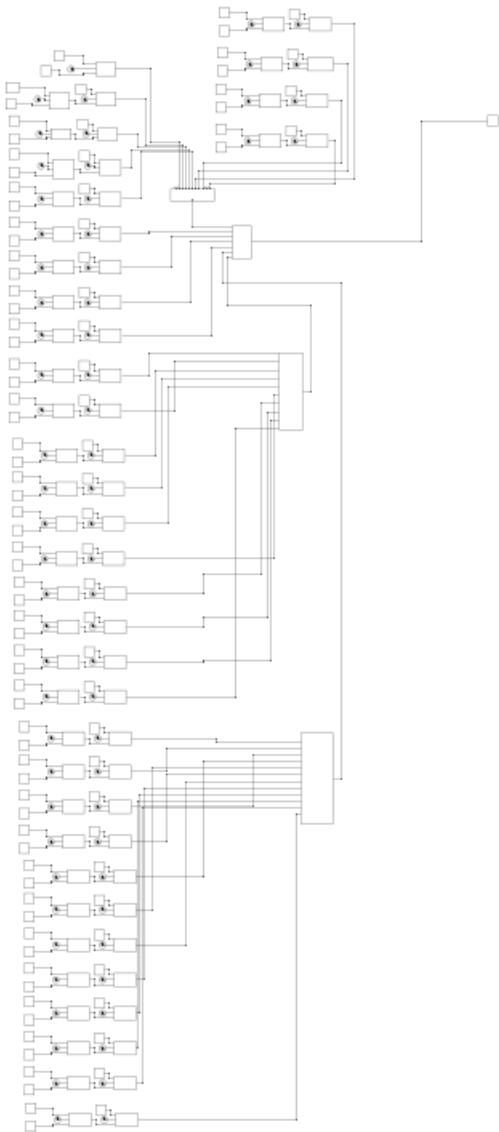


Figure XIII20: Simulink Battery Model Current for Extended Test One

This was done using the same method as described in the engineering analysis for the power supply section.

Conclusion – Extended Test One:

The final state of charge during the test was found to be 82%, whereas the Simulink model predicted a state of charge of approximately 83% if the battery started with an initial charge of 89%, and the average of the current ranges experienced during testing were used. This resulted in a difference of +1% from our actual test, which falls within our acceptable range.

Extended Test Two:

SOC (I)	SOC (F)	Time (start, s)	Time (end, s)	Time (s)	Current (average range, A)	Average Current	Voltage (V)	Terrain Description	Slope (degree)	Notes
95	95	0	0	0	0		40			(filler line for graph)
95	95	0	270	270	5.3-5.7	5.5	40	Sand - no additional weight	0	Urec beach volleyball courts
95	94	271	525	255	6.7-8.5	7.6	40	Sand - with additional weight (+80lbs)	0	Urec beach volleyball courts
94	93	526	540	15	9.2-10	9.6	39.9	Sand - with additional weight (+80lbs)	14	Urec beach volleyball courts
93	93	541	780	240	7-8	7.5	39.8	Concrete - with additional weight (+80lbs)	0	Navigating urec
93	92	781	900	120	5.5-6.5	6	39.9	Grass - with additional weight (+80lbs)	0	Navigating urec
92	91	901	910	10	5.4-9.6	7.5	39.9	Grass - No additional weight	20	At the levee
91	90	911	1090	180	5.5-8.6	7.05	39.8	Grass - No additional weight	0	At the levee
90	89	1091	1260	170	7.4-13.2	10.3	39.8	Grass - with additional weight (+80lbs)	20	At the levee

Figure XIII14: Extended Test Two - Description (1)

Final Results		Total Time (s)	Total Time (m)	Notes
SOC I	SOC F			
95	89	1260	21	Some percentage was lost during transportation

Figure XIII14: Extended Test Two – Description (2)

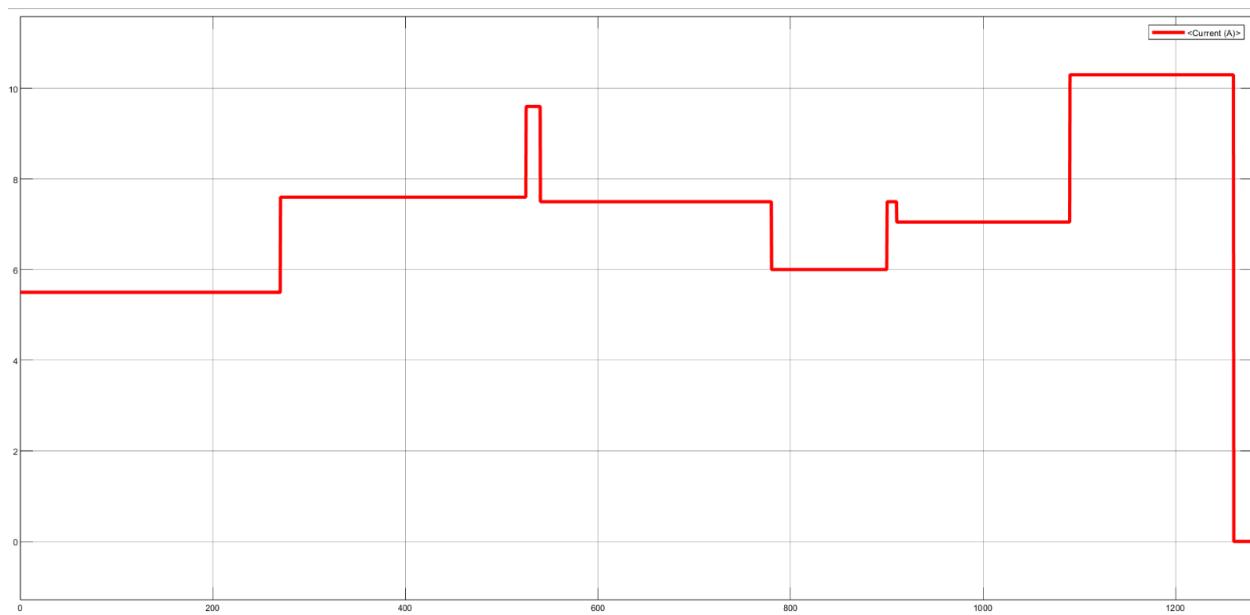


Figure XIII16: Extended Test Two- Current Drawn from Battery

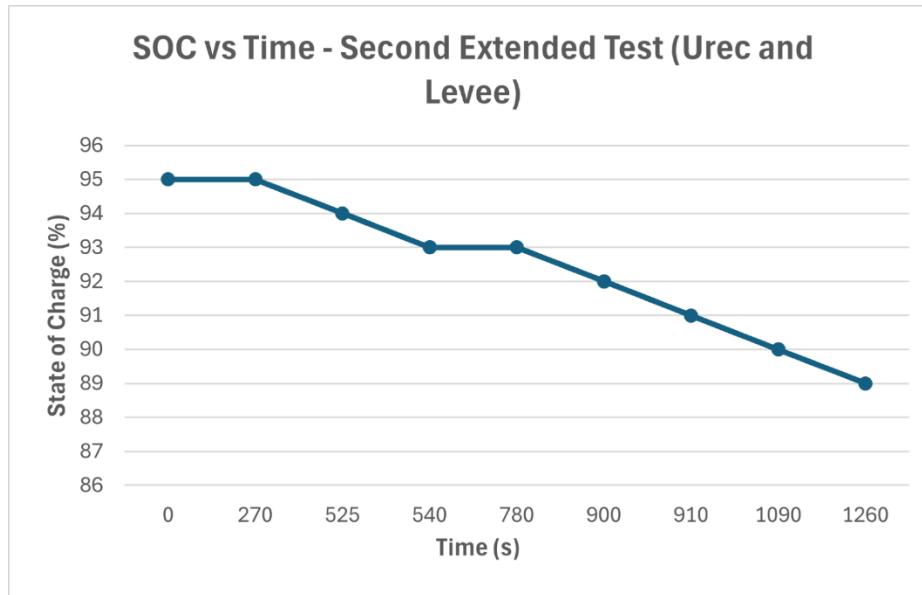


Figure XIII17: Extended Test Two- State of Charge of Battery

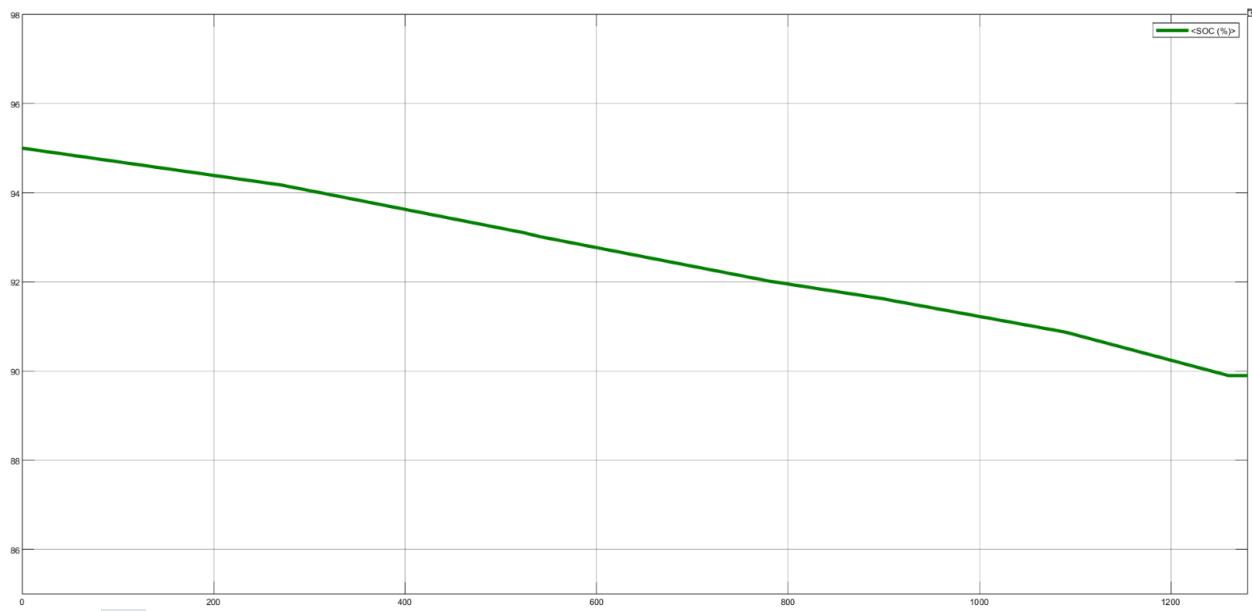


Figure XIII18: Extended Test Two – Simulink Expected State of Charge of Battery

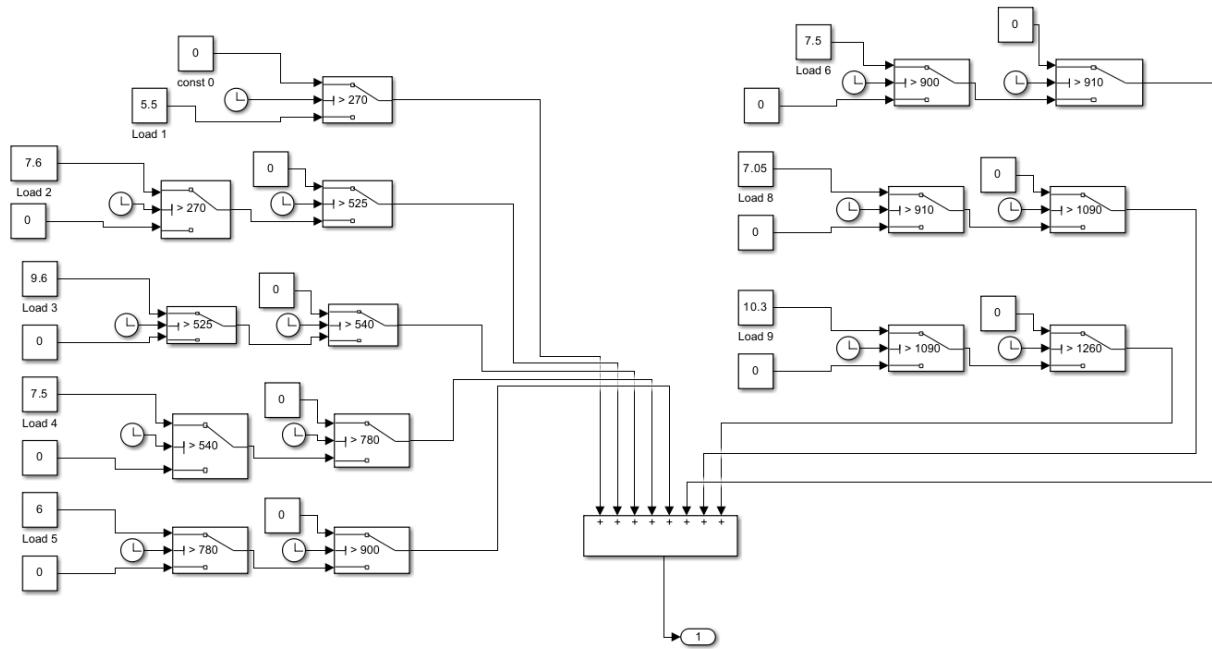


Figure XIII20: Simulink Battery Model Current for Extended Test Two

This was done using the same method as described in the engineering analysis for the power supply section.

Conclusion – Extended Test Two:

The final state of charge during the test was found to be 89%, whereas the Simulink model predicted a state of charge of approximately 90% if the battery started with an initial charge of

95%, and the average of the current ranges experienced during testing were used. This resulted in a difference of +1% from our actual test, which falls within our acceptable range.

Extended Test Three:

SOC(I)	SOC(F)	Time (start, s)	Time (end, s)	Time (s)	Current (average range, A)	Average Current	Voltage (V)	Terrain Description	Slope (degree)		Notes
84	84	0	0	0	0		40	Filler line for graph			Entire Test was done with 80 pounds of additional weight
84	84	0	60	60	0.4	0.4	40	Idling (code adjustment)			Turning Currents were the same as previously measured
84	83	61	226	165	5-6	5.5	39.6	Concrete outside the shop	0		Battery Remained at a temperature of around 28 Degrees Celcius, not exceeding 28
83	83	227	232	5	10.6	10.6	39.7	Going over bump	0		
83	82	233	496	263	5.6-6.5	6.05	39.9	Grass	0		
82	82	497	512	15	12.5-13.5	13	39.8	Mulch	12 (Uphill)		
82	82	513	588	75	5.4-6.2	5.8	39.6	Grass	0		
82	81	589	913	324	5.4-6.2	5.8	39.6	Concrete	0		
<hr/>											
<hr/>											Test ended early, the wheel shaft failed
SOC I	SOC F		Tot Time(s)	Tot Time(m)							
84	81		907	15.1166667							

Figure XIII14: Extended Test Three- Description

Note that this test was terminated early because one of the wheel shafts failed.

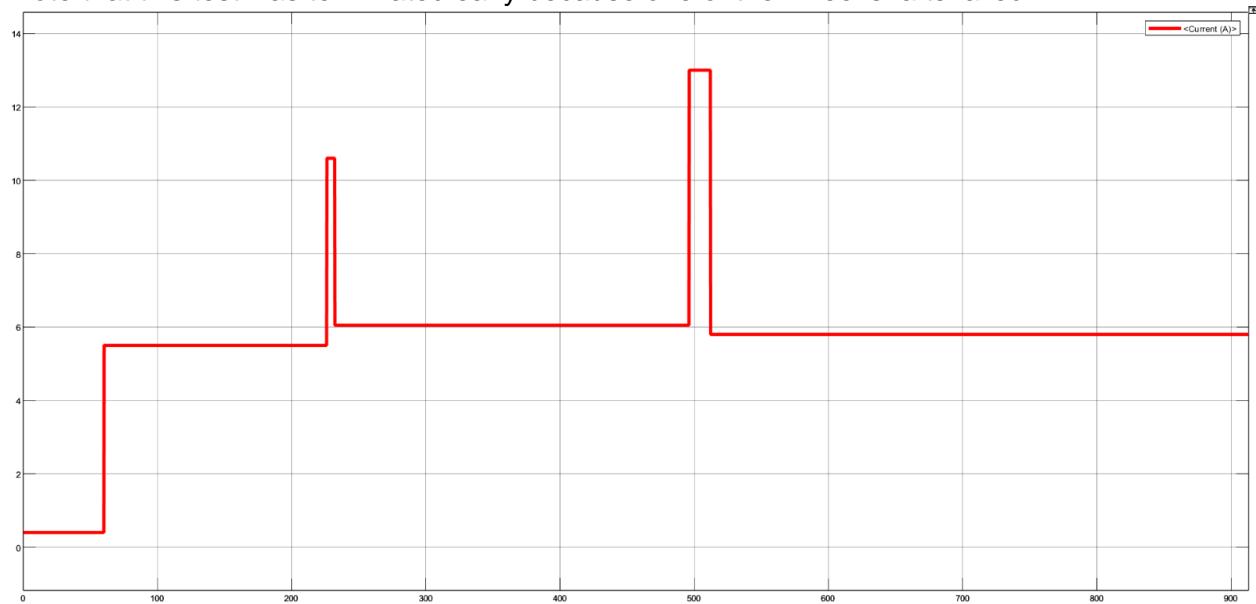


Figure XIII16: Extended Test Three- Current Drawn from Battery

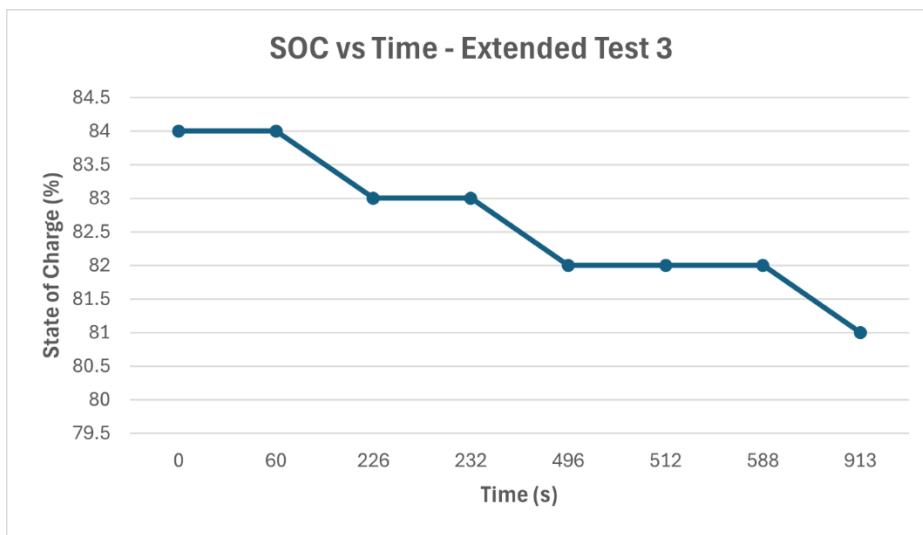


Figure XIII17: Extended Test Three- State of Charge of Battery

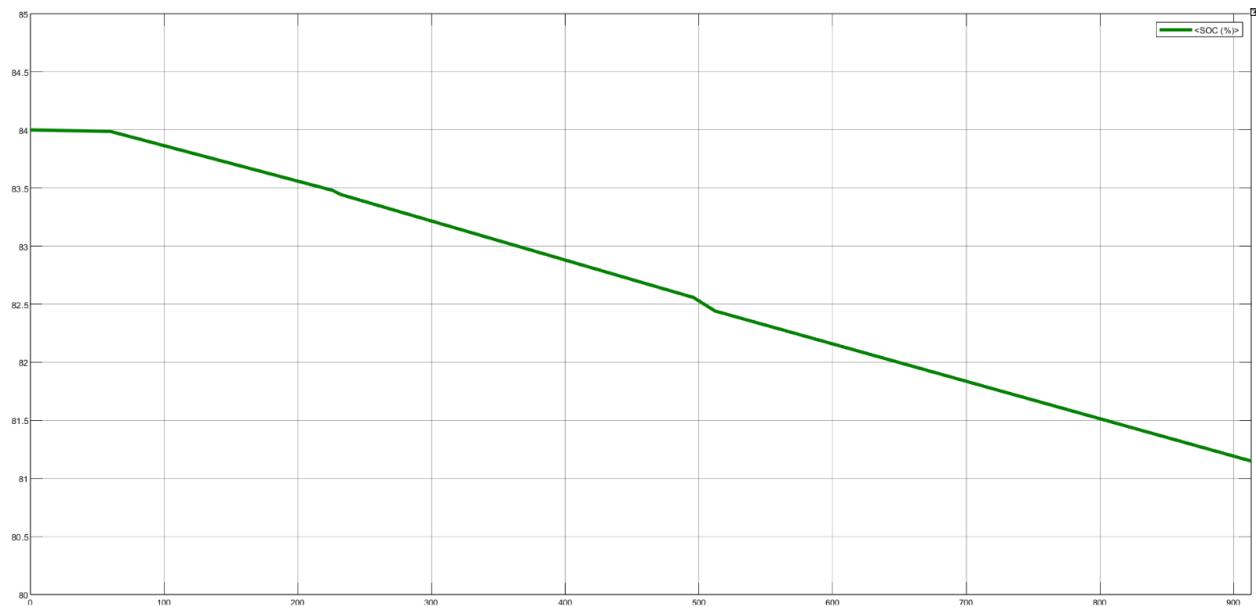


Figure XIII18: Extended Test Three– Simulink Expected State of Charge of Battery

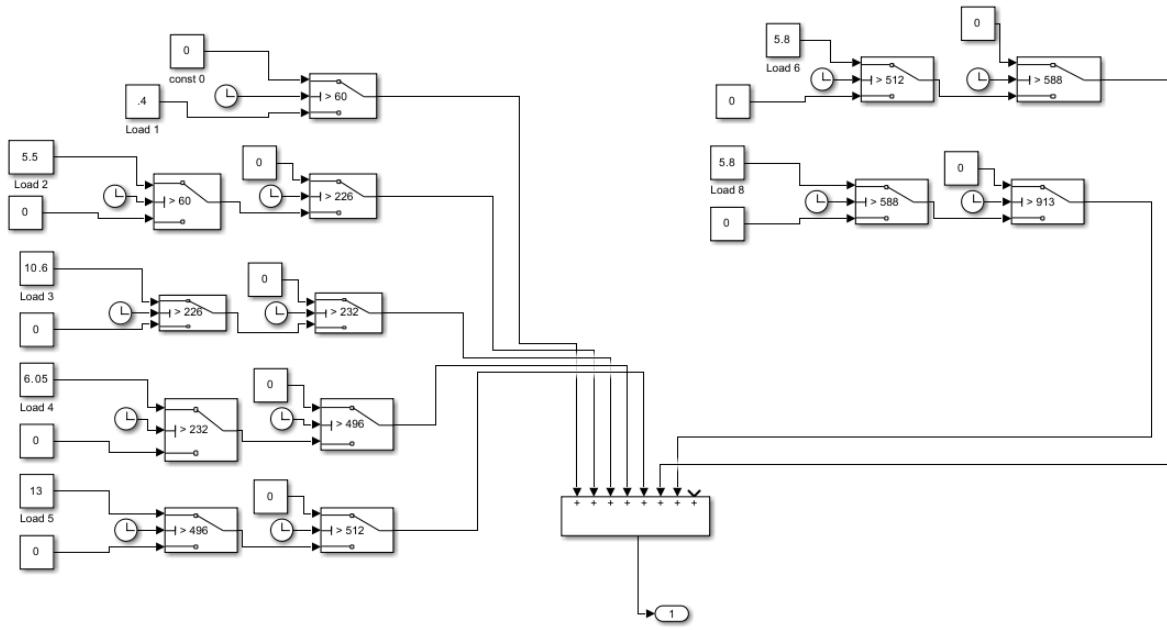


Figure XIII20: Simulink Battery Model Current for Extended Test Three

This was done using the same method as described in the engineering analysis for the power supply section.

Conclusion – Extended Test Three:

The final state of charge during the test was found to be 81%, whereas the Simulink model predicted a state of charge of approximately 81% if the battery started with an initial charge of 84%, and the average of the current ranges experienced during testing were used. This resulted in a difference of 0% from our actual test, which falls within our acceptable range.

Relevant Currents:

- The current when the rover is idling is 0.4A
- Turning Currents on Grass and Concrete (with additional load of 80 lbs):
 - Front right turn: 6.5-7.8A
 - Front left turn: 4.7-6.2A
 - Back left turn: 7.7 - 8.5A
 - Back right turn: 7.7 - 8.7A
- Turning Currents on Sand with no additional load:
 - Straight left: 5.4-5.9A
 - Straight right: 8.1-9.4A
 - Back left: 9.3-10.3A
 - Back right: 9.6-11.2A
- With Additional load on Sand:
 - Straight left: 7.4-8.2A

Back right: 10.2-11.3A
Back left: 11.8-12.4A
Straight right: 9.8-11.1A
Turn in place: 12.8-15.4A

XIII.G.20. Testing & Validation of Quantitative Constraint M8 - Slope

Michael Cannon (ME)

XIII.G.20.a. Test Protocol Description – M8

The objective of this test was to test the rover's ability to climb slopes. A minimum required slope of 5 degrees was requested by the sponsor. An ambitious test slope of 20 degrees selected by team. The testing location was selected was the Mississippi River levee which has a slope ranging from 15 to 20 degrees. The rover was driven directly up the side on a path of uneven dirt and short grass. This test was also conducted with the extra 35 kg load attached and the motors set to 25 percent of maximum speed. Due to a limited testing window, this test could only be completed once.

XIII.G.20.b. Equipment and Instrumentation – M8

- Camera
- Measuring tape
- Video editing software (iMovie)

XIII.G.20.c. Data Acquisition & Analysis – M8

The average speed was found by taking the measured drive distance and dividing it by the time of travel as recorded by the footage. This was done using video editing software as noted in earlier testing sections.

XIII.G.20.d. Results Details – M8

The rover was successfully able to climb to the top of the levee, exceeding our target by 300%.

Table XIII-24: Slope test results

Slope (degrees)	Terrain	Load (kg)	% of Max motor speed	Average rover speed (m/s)
15-20	Uneven dirt and short grass	35	25	0.33



Figure XIII-24: Rover climbing the levee during the slope test.

XIII.G.21. Testing & Validation of Quantitative Constraint M9 – Detection Distance

Creighton Cocathey (EEC)

XIII.G.21.a. Test Protocol Description – M9

The testing protocol for the detection distance was designed to assess the performance of ultrasonic sensors in accurately detecting objects at various distances. The primary objective was to determine the optimal range within which the sensors maintain high accuracy, fulfilling the project's requirements. Tests involved connecting two ultrasonic sensors to an Arduino using a breadboard setup, allowing for flexibility in adjusting and configuring the setup. Objects were systematically placed at incremental distances from the sensors, starting close and extending to the sensors' maximum range. This methodical approach enabled a detailed evaluation of the sensor's performance across its entire operational range.

XIII.G.21.b. Equipment and Instrumentation – M9

The equipment setup for the detection distance tests included two ultrasonic sensors connected to an Arduino microcontroller via a breadboard. This configuration facilitated easy modifications and real-time testing adjustments. The setup also involved various objects used as targets for the sensors, positioned at defined distances to test detection accuracy. Measurement

tools to precisely place and verify the distance of objects from the sensors were employed to ensure test accuracy. Data logging equipment was used to record sensor outputs at each distance, allowing for detailed analysis of sensor performance.

XIII.G.21.c. Data Acquisition & Analysis – M9

Data acquisition focused on capturing sensor readings at multiple distances to analyze the range and accuracy of detection. Sensors were programmed using C++ in the Arduino IDE to take multiple readings per second to mitigate the effect of any transient or spurious readings. A filtering algorithm was applied to these readings, where the highest and lowest values were discarded, and the remaining data were averaged to determine the most reliable reading at each distance. This approach was critical in identifying and correcting outliers, thus refining the detection capabilities of the ultrasonic sensors. The analysis aimed to establish a clear profile of sensor performance across varying distances, particularly focusing on the maximum reliable detection range.

XIII.G.21.d. Results Details – M9

The results from the ultrasonic sensor tests revealed that the sensors could reliably detect objects up to a distance of 2.5 meters, well within the project's minimum requirement of 2 meters. This optimal detection range was established after iterative testing and refining of data collection methods, notably the high-frequency sampling and outlier filtering technique. These findings confirmed the sensors' effectiveness in meeting the project specifications for detection distance. The detailed analysis of sensor performance across different ranges provided valuable insights into their operational limits and capabilities, ensuring that the system design is based on robust and accurate sensor data. The successful completion of these tests marks a significant milestone in validating the ultrasonic sensors' functionality and reliability for future project applications.]

XIII.H. Project Management Supplement

XIII.H.1. Schedule and Milestones Details

[]

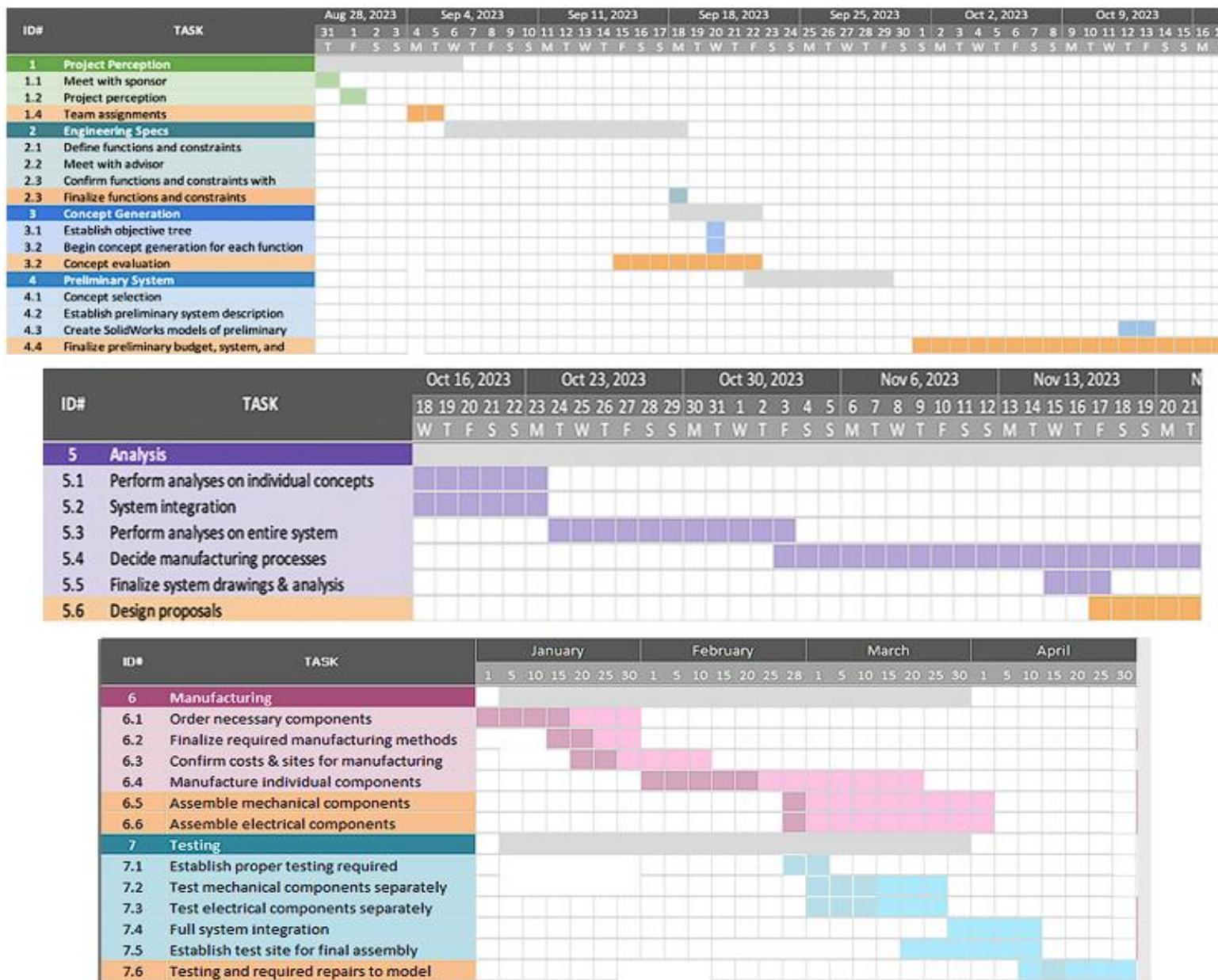


Figure XIII-25: Gantt Chart with Detailed Project Time-Line

XIII.H.2. Budget Details

The total spending for the project was \$5,606.76. This includes all components ordered, the costs of manufacturing, and testing costs. This leaves the team with an extra \$393.24 that will be recycled into the budget for next year's team as they expand upon this project. A full breakdown of the costs for each subsystem can be seen in Table XIII-10 below. |

Table XIII-25: Bill of Materials/Parts |

Subsystem	Components	Quantity	Cost/Item	Total Cost	Manufacturer
Frame	Rust Paint	1	19.99	19.99	Amazon
	Velcro Squares	2	3.12	6.24	Amazon
	Electrical Tape	1	1.23	1.23	Amazon
Legs	Rocker	1	49.8	49.8	Grainger
	Bogie	2	30.25	60.5	Grainger
	Rocker Axle	1	73.7	73.7	Grainger
	Bogie Axle	1	23.72	23.72	Grainger
	Leg Bushing	1	44.16	44.16	Grainger
	Leg Bolts	1	13.49	13.49	Amazon
Drivetrain	Wheels	3	60.6	181.8	Amazon
	Wheel Axle	1	52.24	52.24	Grainger
	Wheel Hub	1	20.72	20.72	Grainger
	Motor Extension	1	24.59	24.59	Grainger
	Motor Attachment	6	15.3	91.8	Grainger
	Wheel Bolts	1	49.99	49.99	Amazon
	Wheel Nuts	1	9.96	9.96	Amazon
	Motor Extension Set Screw (1)	1	10.4	10.4	Grainger
	Set Screw	1	10.4	10.4	Grainger
	Wheel Retaining Rings	1	5.61	5.61	McMaster
	Motor Extension Set Screw (2)	1	10.99	10.99	Amazon
	Motor Housing Bolts	1	8.99	8.99	Amazon
Suspension	Axle Bearing	1	20.53	20.53	Grainger
	Axle Brackets	1	34.96	34.96	Grainger
	Differential Bar	1	54.48	54.48	Grainger
	Differential Strut	2	5.96	11.92	Grainger
	1.5" Retaining Rings	1	11.38	11.38	McMaster
	0.5" Retaining Rings	1	8.41	8.41	McMaster
	Rod End Joint	2	21.55	43.1	McMaster

	Nuts & Bolts	1	9.99	9.99	Amazon
Electric Propulsion	Motor	6	246	1476	Powerhouse Engineering
	Motor Housing	1	188.32	188.32	Grainger
	Bolts	1	15.35	15.35	Grainger
Power Supply	Battery	1	999	999	Enduro Power
	Battery Charger	1	249	249	Enduro Power
Power Distribution	12 to 3V Converter	1	5.09	5.09	AliExpress
	36/48V to 24V 30A Voltage Converter	3	49.99	149.97	Amazon
	36/48V to 12V Voltage Converter	1	18.99	18.99	Amazon
	12 to 5V Voltage Converter	1	14.99	14.99	Super Bright
	Power Distribution Bar	3	27.11	81.33	McMaster
	ELEGOO Multicolored Dupont Wire	1	6.98	6.98	Amazon
	12 AWG Wire	1	10.98	10.98	Amazon
	24 AWG Wire	1	6.28	6.28	Amazon
	4 AWG Wire (black)	10	1.32	13.2	Wire & Cable
	4 AWG Wire (red)	10	1.32	13.2	Wire & Cable
	8 AWG Wire (black)	45	0.56	25.2	Wire & Cable
	8 AWG Wire (red)	45	0.56	25.2	Wire & Cable
	4 Pin Buckled Grove 20 cm Cable	1	12.99	12.99	Amazon
	Inline Fuse Holder	3	9.99	29.97	Amazon
	25A Maxi Blade Fuse	6	2.96	17.76	McMaster
Controller	Circuit Breaker	1	23.99	23.99	Amazon
	Micro-Controller	1	52	52	Amazon
	Motor Controller	1	79	79	Maker Motors
	Rail Mount	1	35	35	Amazon
	Jetson Nano	1	149	149	Amazon
	Miuzei 20 kg Servo Motor	1	13.59	13.59	Amazon
	Servo Motor	1	7.99	7.99	Amazon

	#8-32 1/4" Machine Screw	1	5.9	5.9	Amazon
Object Detection/Reaction	Ultrasonic Distance Sensor	1	3.95	3.95	Adafruit
	Camera	1	57.99	57.99	Amazon
Communication	Temperature Sensors	4	9.95	39.8	Adafruit
	Gyroscope/Accelerometer	1	34.95	34.95	Adafruit
	Wi-Fi Module	1	23.99	23.99	Amazon
Manufacturing & Testing	Waterjet Manufacturing (1)	1	97	97	AMMF
	Weld Testing	1	41	41	Amazon
	Waterjet Manufacturing (2)	1	39.5	39.5	AMMF
	Strain Gauge Testing	1	80	80	MMTEF
Total (with shipping)		\$5606.76			

XIII.I. Reference Materials

XIII.I.1. Codes and Standards

1. AWS Committees and Standards Program | Shaping the Future of Welding Standards. (n.d.).
Www.aws.org. Retrieved December 6, 2023, from <https://www.aws.org/Standards-and-Publications/Codes-andStandards/#numberOfResults=5>
2. 2014 NEC 330.34
3. 2014 NEC 330.24

XIII.I.2. References and Bibliography

1. Mars.Nasa.Gov. (n.d.-a). Learn about the rover - NASA.
<https://mars.nasa.gov/mars2020/spacecraft/rover/>
2. Reinventing the wheel. (n.d.). Reinventing the Wheel.
<https://www.nasa.gov/specials/wheels/>
3. McMaster-Carr. (n.d.). <https://www.mcmaster.com/products/motors/compact-dcmotors-7/>
4. EaglePicher Batteries are Returning to Mars. (n.d.)
<https://www.eaglepicher.com/resources/news-and-events/eaglepicher-batteries-are-returning-to-mars/#:~:text=The%20rover's%20main%20power%20system,volt%20battery%20weighing%2059%20pound>
5. Mars Perseverance Rover Power Source. (n.d.)
<https://mars.nasa.gov/mars2020/spacecraft/rover/electrical-power/>
6. Meriam, J. L., Kraige, L. G., & Bolton, J. N. (2018). Engineering mechanics. Wiley.
7. Webb Machinery. (2018, February 12). Champ CNC Knee Mills - Webb Machinery.
Webb Machinery - Providing High Quality Machines for Over 40 Years!
<https://www.webbmachinery.com/champ-cnc-knee-mills/>
8. The Welder, July 19, 2018, Andrew Pfaller,
<https://www.thefabricator.com/thewelder/article/arcwelding/tig-welding-aluminumwhat-you-need-to-know>
9. EML2322L -MAE Design and Manufacturing Laboratory Drive Wheel Motor Torque Calculations Reference Citation: White Hydraulics Drive Products. (n.d.).
<https://mae.ufl.edu/designlab/motors/EML2322L%20Drive%20Wheel%20Motor%20Torque%20Calculations.pdf>
10. Mohan, Ned. (2003) Electric Drives: An Integrative Approach. MNPERE
11. <https://www.thefabricator.com/thefabricator/article/waterjetcutting/preparing-a-fabricators-game-plan-for-jumping-into-waterjet-cutting>
12. <https://www.normanmachinetool.com/product/acer-precision-geared-head-engine-lathes-17-x-40/>
13. <https://machinehub.com/listings/343280-24-quot-wilton-strand-model-24514-drillpress-stock-115522>