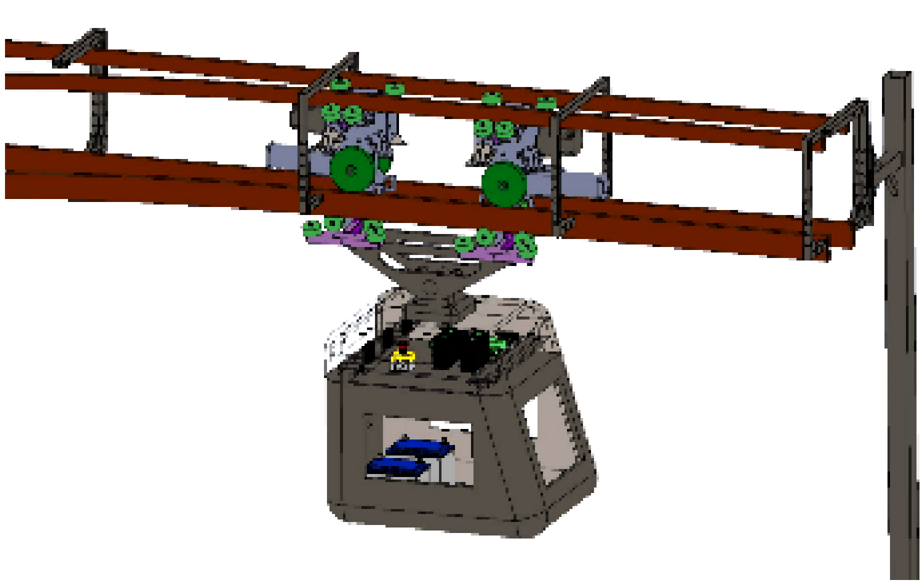
SPARTAN SUPERWAY

Solar **P**owered **A**utomated **R**apid **T**ransit **A**scendant **N**etwork:
Half-Scale Team

Gregory White
Brandon Keith Scully
Keanu Heggem
Pinqian Lin

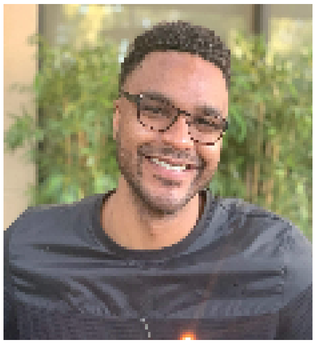
May 15, 2020

San Jose State University



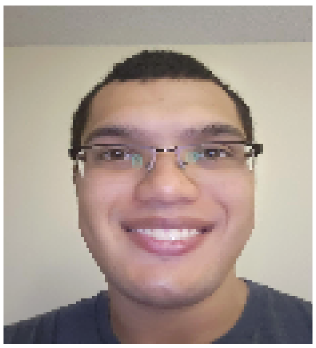
Advisor: Dr. Burford Furman

Team Member Roles



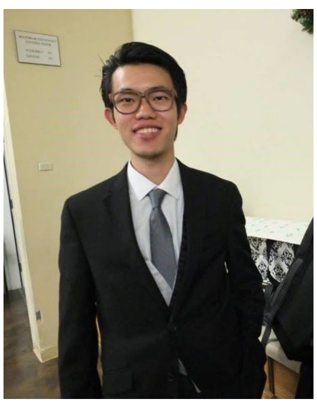
Gregory White **(Project Lead)**
gwhite51191@gmail.com

- Bogie Redesign
- Switch Mechanism
- Mechanical Repair
- Mechanics/Electronics Team



Keanu Heggem
kotainokage24@gmail.com

- Switch Mechanism
- Mechatronics Coding
- Controller Redesign
- Electronics Team



Pinqian (Edward) Lin
hellolynn124@gmail.com

- Control Box Repair/Redesign
- Half-Scale Track Assembly
- Mechanics Team

Brandon Scully
bkscully@yahoo.com

- Half-Scale Track Redesign
- CAD Modeling/Documentation
- Mechanics Team

San José State University
SPARTAN SUPERWAY

Abstract

The 2019-2020 half-scale team started with the single goal of redesigning and perfecting the switch arm. Throughout the year, and due to multiple setbacks and anticipated issues; several secondary goals were established. The bogie was refurbished and upgraded, and a new switch arm along with a control box was designed. CAD models for an improved track and test code programs for trial runs of the prototype switch arm with wireless controls were developed. Due to the COVID-19 pandemic shelter-in-place order, much of the progress of the project exists in CAD and FEA analysis; abandoning intentions to manufacture prime designs.

The final designs of this project were developed through repeated trial testing and analysis of results. The switching arm was designed to counteract the problem of the bogie twisting as it attempted to cross the Y-section of the track, and the redesign of the track was done to account for the extra clearance that the wheels would need to cross under the support brackets that held the track in place. The wireless controller was first conceptualized through flowcharts and computer diagrams then were built and coded. The code for the trial run eliminated the unnecessary bits from the master code inherited from last year, only focusing on the movement functions and the limit switch. The code was also designed to move the stepper motor in the smallest possible movements, to ensure that it stopped precisely. The control box was redesigned both to carry the system controls underneath the bogie and to prevent a recurrence of the damaging short that happened in December 2019.

The prototype switch arm was a resounding success. The bogie traversed the curved branch of the Y-section without issue moving forwards and backward. Additionally, a very fleshed out CAD of the final switch arm design was created. The track redesign produced a detailed CAD model, and the trial run code operated both the limit switches and the switch arm flawlessly. The wireless controller, while unsuccessful, produced promise in the use of nRF24L01 wireless communications modules in the system controls and left a blueprint for future years to follow in creating the controller. The control box design produced a defined CAD model that could be used to manufacture the box. Future years would be recommended to manufacture all the CAD models left behind by this year’s team and to finish the prototype codes and wireless controller to make an effective model bogie.

Acknowledgments

We would like to sincerely thank Dr. Burford Furman for supervising us throughout this school year for the 2019-2020 Spartan Superway half-scale project. Specifically, Dr. Furman has shown continued support and encouragement for our project by scheduling weekly meetings, enquiring about updates on our tasks, and most importantly, providing invaluable advice when a design problem arose in our project. Additionally, we would like to thank Mr. Eric Hagstrom for attending our presentations over the past year, providing feedback on our design solutions for the control box and the switch arm as well as demonstrating how to document and organize our work in the Spartan Superway archive on Google Drive. Furthermore, we would like to thank Oscar Balvaneda and Hector Gomez for their advice on documenting our CAD files and input in carrying out our goals for the 2019-2020 Spartan Superway half-scale Team. Lastly, we would like to thank Mr. Ronald Swenson for providing the facility and materials to carry out our project, for allowing us to present our solutions to guest speakers/conventions and for emphasizing the importance of the implementation of the Spartan Superway in our transportation system today.

Our list of sponsors for this project are stated as follows**:**

1. Swenson Builder
2. I.N.I.S.T.

Table of Contents

Table of Figures

Table of tables

Executive Summary

Introduction

The Spartan Superway was developed by Dr. Furman and Engineer Ron Swenson, it was designed to be a suspended transportation network that will transfer passengers between one location to the next without interruption or delay. Most motor vehicles rely on limited pollutant fossil fuel sources, the Spartan Superway is sustainable and competitive in the sense of using only renewable energy sources. Also, most methods of transportation remain competitive for the same space, creating public congestion and safety hazards, endangering road users, bike riders, and so on. The Spartan Superway will set the roadways free for people to live and prosper in. Public transportation is unreliable and does not provide sufficient accessibility for low socioeconomic and disabled people. The main objective of the 2019-2020 half-scale bogie design and controls team was to refine the design of mechanical and mechatronic systems in such a way that the bogie will consistently traverse the fork of the track during trial testing. The trial testing and redesign of the switching arm along with other key components will bring the Spartan Superway much closer to public implementation.

Objectives

- Perform dynamic analysis on the bogie mechanism adopted from last year’s team
- Improve the design and performance of the bogie
- Prototype a new switch arm which clamps onto the track firmly
- Create a trial test code for switch arm prototypes to find the prime design
- Redesign a suspended control system housing that attaches to the bogie
- Diagnose the hardware and wiring connection errors in the control system
- Build and program a wireless controller to enable mounting of the control box while maintaining manual failsafe controls of the system

Procedure and Results

This project is a progression of the previous year's half-scale team. The inherited half-scale bogie and control box had several issues, so our team first had to evaluate the project state and assess what needs to be addressed. The mechanical model and mechatronics of the bogie were not functional and needed to be diagnosed and improved. The steering mechanisms were modified and trial-tested with wood prototypes, and the metal shafts that control the drive and steering components consisting of chains and sprockets were realigned with pilot holes and set screws. Various gearbox and bearing components were grounded with a rotary tool. The track also needed modification, it was cut in some portions that interfered with the bogie steering mechanism. The diligent work of the 2019-2020 half-scale team resulted in the completion of all objectives outlined in the Fall 2019 semester.

Owing to its simplicity and flexibility an Arduino microcontroller board was used for the mechatronics of the control box inherited from the efforts of last year's team. The code was constructed through stages of development. First, a separate code was written specifically for trial testing, utilizing limit switches while also adding a micro-stepping feature for the motor driver. These codes were then integrated into an alternative code that mapped the functions of the wireless controls to specified pins. Once evaluated, the interchangeable codes were uploaded according to what needed to be tested. The control box, unfortunately, short-circuited and needed to be diagnosed and refurbished to continue any testing that needed to be done. The system was rewired and a soldered board was installed that manages and regulates power; preventing further irreversible damage to control box components. Many features were removed to reduce clutter and make the control box less likely to have the same issues. Following the implementation of these modifications, it was possible for trial testing of the bogie switch arm to resume. As the trial testing code continued to develop based on the needs of the half-scale mechanics sub-team, the progress of the trial testing did so as well. After many prototypes and modifications to the mechatronics and mechanics of the bogie, trial testing was successful and repeatable, the prime design of the switch arm and control box housing was conceptualized.

Conclusion and Recommendations

The progress of the 2019-2020 half-scale bogie and controls team was the result of hard work and skill, this is demonstrated by the refined functionality of the bogie while traversing the switch. The bogie reliably switching on the track was a goal attempted by many teams previously, but was only accomplished by the 2019-2020 half-scale team. The bogie worked as per requirement. It was able to reliably navigate the track due to: modifications to the code, redesigned steering arms, and shaft alignment procedures. The main issues were the short-circuited motor driver that was irreparable and the bogie alignment, these problems were the most time consuming to correct. Changes in the structural configuration of the switch arms improved bogie performance in many ways. Future half-scale teams must be focused on analysis and manufacturing the final switch arm design as well as fastening the drive shaft. The second course of action would be the introduction of wireless communication with a DC power source in addition to repeating the alignment procedure on the second bogie.

Introduction and Project Description

Population growth has led to considerable congestion in urban areas. Cities are growing at a rate that existing infrastructure cannot handle, and as such these populated areas are becoming overwhelmed with vehicles. This is not a new trend. In 1984 urban freeway congestion was estimated to cause 1.2 billion hours of delay annually (Lindley, 1987). This is making commutes take longer and use more gasoline, resulting in more fuel consumption in everyday life. Furthermore, these same vehicles that are congesting the roadways in cities are also emitting toxic gases. It is well known that petroleum-based vehicular transport is responsible for a large number of greenhouse gas emissions (Kenworthy, 2003). The amount of automobile use on the streets results in a massive flux of harmful vapors into the atmosphere. This issue is exasperated by the aforementioned growth in traffic congestion in urban areas. This combination causes a steep rise in greenhouse gas emissions in these places.

Urban areas often have public transportation systems to both ease congestion in cities and provide people with a cheap means to move about everyday life without a personal vehicle. However, these transit systems rarely can stand up to the task. Constant population growth means that the systems are outdated within a few decades, and can be expensive to expand further. The primary of these systems, public buses, can get caught up in the congestion as well. Their more expensive and more private counterparts in taxis suffer from the same issue. While some cities manage to make these systems dependent on energy sources that are less emissive than petroleum, the time that they can spend bound in traffic is still an unsolved issue. Other transit systems such as the BART and the New York subway system can avoid the surface level traffic issue but exist underground and are therefore expensive to build and maintain. (Reilly, 2000). They are also more limited by geology than aboveground systems would be. This often leads to cost overruns and significant delays. Furthermore, tunneling can be very difficult in urban areas due to the density of pre-existing facilities (Reilly, 2000). This points towards an elevated solution to the congestion problem.

An issue that most public transit systems encounter is convenience. Public transit is typically bound to assigned roadways, and this means that passengers must take specific paths and detours to get to destinations. As the average income within urban areas rises, the people living within them find ownership of a car to be increasingly attractive due to greater independence and the lack of rigid schedules that public transit must adhere to (Poudenx, 2008). This has led to less dependence on public transit and a continued rise in greenhouse emissions through personal vehicular transport. Other factors such as economic development can also be to blame, as they have resulted in heavily populated nations such as China pushing their populations towards vehicle ownership (Poudenx, 2008). A reliable and flexible transit system that can adhere to the individual rider's needs and schedule is needed to turn urban residents back towards public transit . This is where ATNs have arisen as potential solutions.

Automated Transit Networks, or ATNs, are personal vehicles designed to run on dedicated guideways. They are typically known as Personal Rapid Transit or PRT vehicles, or more colloquially as podcars (Furman et al, 2014). These vehicles are different from other public transit solutions in that they transport individual parties. This is similar to taxis, but the advantage of ATNs is that they do not share their guideways with any other modes of transportation. This system would provide the convenience that so many other transit systems lack and could therefore significantly reduce the number of petroleum-dependent vehicles on the road.

The Spartan Superway aims to take this concept a step further. The Superway is a plan to create an ATN that would run above existing roadways and is entirely dependent upon renewable energy resources. The ATN would consist of a fleet of podcars that automatically respond when called by a user at a station. This system would be unimpeded by the congestion of the roads below and thus would be more effective than current bus transit. The system's planned reliance on renewable energy resources would drastically cut transportation-related greenhouse emissions by reducing the number of cars on the road.

A transportation system like the Superway does not exist in a large scale capacity. A system of this type would have heavy ramifications within the global society. If applied in a city it would spark a race to catch up to or surpass such a system and would be an attractive citywide landmark to promote tourism and modernization. Building a transportation system depending entirely on renewable energy which extends throughout urban areas would be a monumental feat, and would revolutionize inner-city transportation.

The transportation system would be especially helpful to disadvantaged groups such as the poor and disabled. The poor depend more on public transit than other groups do, as evidenced by most gathering in areas where public transit is a viable option. Pathak and his contemporaries (2017) outline the poverty distribution within the city of Atlanta within the state of Georgia in the United States and find that the percentage of the population in areas with public transit access is significantly higher than those without such access. This implies that poor and impoverished families cannot survive without public transit options, and many cities simply don't have the public transit systems in place to aid them. Similarly, many are limited to public transit by disability and therefore are burdened by inefficient public transit options. Rapidly growing cities such as Jeddah in Saudi Arabia have grown so much that their streets have become a convoluted mess that makes public transit a near impossibility, further burdening these groups. In 2012 96% of all transportation in Jeddah was done through private vehicles (Aljoufie, 2014). Similar situations exist in the capital city of Riyadh and various cities throughout the world. Cities such as these would benefit immensely from a modern public transit system. The Superway is a promising option for those in populous areas and underprivileged groups as a whole.

Project Objectives

The primary objective of the 2019-2020 half-scale team was to improve the design and performance of the switching mechanism by prototyping a new switch arm that clamps onto the track firmly and redesigning a functioning prototype code for the switching mechanism to switch successfully at the Y-junctions of the half-scale track. Additionally, we aimed to redesign the housing for the system controls, making it more compact, organized, and easily accessible for replacements and repair. Furthermore, we aimed to diagnose the hardware and connection errors in the systems controls while also performing dynamic analysis on the bogie mechanism following the progress of last year's half-scale team. Lastly, we anticipated building and programming a wireless controller to allow for suspended mounting of the control box while retaining the fail-safes of the manual control system.

As a result of the shelter in place order, in addition to our original primary objective, the 2019-2020 half-scale team aimed to design a conceptual simulation of the modified bogie mechanism paired with the prime design of the switch arm on the half-scale track using SolidWorks. The team planned to also redesign the half-scale track in Solidworks to assess the interference issues with the switch arm and the "C shaped beams" while traversing the current design of the half-scale track.

Design Requirements and Specifications

The 2019-2020 half-scale team continued to improve the mechanical design of the switch arm and control system of the bogie initiated and passed on to the current team from the previous

year's half-scale teams. The 2019-2020 half-scale team initially intended to achieve certain design requirements and specifications which changed as the project was moving forward and due to the Covid-19 pandemic. Because of the Covid-19 pandemic and shelter in place order, the planned prototype testing of the switch arm mechanism had been postponed and the plan of manufacturing the final design of the switch arm and control system box had been changed to the redesign phase.

At the start of the 2019-2020 school years, the team spent time reviewing the progress of the project and proposed ideas for workable solutions for the obstacles and challenges that the project is facing. A complete driving bogie and a control system were built, but a lot of problems started showing up as more trial runs were carried out. From the report of the previous team, the bogie was able to run on the track with a speed of half-foot per second and it theoretically should be able to carry a 600-pound weight. From the trial run at the beginning of the 2019-2020 school year, the team found out that the switch arms only turn one direction, the limit switch was not responding when the arm reached the maximum angle, the switch arm system does not apply enough force to keep the bogie on the track at the turning point, the gear chain started to slip as the bogie was traveling through the Y-intersection and as a result, the bogie fell off the track. To push this project forward, the switch arm must be strong enough to hold the bogie in position while turning at the Y-intersection. The switch arm has to apply enough force to the track to prevent the bogie from tilting with the designed load of 600-pounds. The teeth of the gear or the length of the gear chain have to be modified to the correct dimensions to prevent slipping. The switch arm should be able to switch direction while the bogie is moving forward or backward without stopping.

The half-scale team also laid out major design requirements to test the switch arm mechanism. The mechatronics of the bogie had to be improved. The broken wire on the limit switch has to be soldered. A new code that implements the limit switch has been written to prevent the switch arm from overturning. However, this code is only at the testing stages and must be improved. A new code that turns the switch arm to the left or right position while the bogie runs back and forth has to be written to make sure the switch arm switches properly. The Arduino panel is supposed to be changed to a new one because some of the pins have connection issues. Due to the Covid-19 pandemic, the panel could not be changed. The sensors and stepper motors have instead been assigned to the functional pins on the Arduino. The number of teeth of the chain gear needs to be increased to prevent slack chain slipping from the gear. Cross-drilled holes for pinned shafts need to be added to prevent the wheels from twisting. The number and size of the wheels of the switch arm need to be increased to have more contact area with the track when the bogie is turning. The power of the system is changed from the power supply to the battery supply. The control housing is redesigned to hang under the bogie and is supplied by two 12V batteries. Because of the Covid-19 pandemic, the manufacturing of the switch arm and control system housing has to be postponed to the next school year for next year's team to fabricate.

State-of-the-Art Literature

Self-driving cars have grown in popularity in this day and age and have been a major subject of discussion. It is highly probable that the automobile industry will follow this methodology of producing autonomous vehicles as production costs will be greatly reduced (Guerra, 2016, p. 210). Currently, there are transit companies that use autonomous train cars. For instance, the KLIA Aerotrain in Malaysia acts as a "complimentary shuttle" between two terminals and a fixed path (KLIA Aerotrain, n.d.).

Research that aided the progression of this project is primarily emphasized on elevated transport systems. Specifically, the SPARTAN Superway project aims to build a high-level integrated transport networking infrastructure that powers transportation pod vehicles through renewable energy. Companies such as JPods, FuTran, and SkyTran are now making this dream a possibility (Personal Rapid Transit (PRT), Personal Automatic Transport (PAT), PodCar, IPM and ATN Quicklinks, n.d.). One service, ModuTram, has a very similar approach to how we develop and automate our Personal Rapid Transit (PRT), Personal Automatic Transport (PAT), PodCar, IPM, and ATN Quicklinks, n.e. Its architecture involves the use of small podcars as well as a control structure that brings passengers to a designated area and provides a flexible timetable to satisfy customer requirements (Personal Rapid Transit (PRT), Personal Automatic Transport (PAT), PodCar, IPM and ATN Quicklinks, n.e.) as seen in **Figure 1A**.

Figure 1A: Podcar for ModuTram

SkyTran planned their rail networks on the premise that they needed to be cost-effective and increase the speed of trains for an improved travel experience for passengers (SkyTran, n.d.). The only differentiated feature between SkyTran and the Spartan Superway, though, is that they rely on a magnetic component for their suspension system contrary to ours, where we insist on a mechanically-operated system. The rendering of the rail network is pictured in **Figure 1B**.

Figure 1B: Podcar for SkyTran

FuTran has emerged as the top rival of ATNs with a suspended commuter rails system (Futran, 2018). Some of the only locations FuTran has implemented its system is in South Africa. FuTran 's elevated guideway is planned to reduce congestion created by road transport. It is in line with the concept of the Spartan Superway to build a suspended track/guideway that reduces the use of personal transport to improve the quality of life and mobility for pedestrians to cycle on the streets.

So far as costs are concerned, commuter rail travel around the country and the Bay Area has been shown to have a high cost of usage per mile. For starters, the Boston Green Line Extension ended up costing $490 million per mile of track (Bosselman, 2018). BART (Bay Area Rapid Transit) costs nearly $780 million a mile to fly to San

Jose (Levy, 2018). However, a transport network similar to the Superway's, JPods, has a projected cost of around $10 million per mile of track (Competitive Advantages, n.d.). From these results, we may infer that, relative to standard transit networking networks, the future of ATN will provide an affordable, efficient, innovative, and, most importantly, safe means of transport.

To sum up the importance of what the previous half-scale teams had done over the years, the 2016-2017 half-scale team managed to complete the manufacturing and assembly of the bogie. Our work was focused on the design of a track and switch arm using data from past half-scale teams and the analysis of the dynamics of the bogie during trial testing. Two bogie systems have been used to support a standard 600-pound load. The 2018-2019 half-scale team continued the progress made in previous years by developing the mechatronics portion of the project which gave the bogie streamlined drivetrain and steering control. On the mechanical design side, they improved the design of the upper and lower control arms along with the shafts to decrease the twisting of the bogie at the Y-section; but ultimately were unsuccessful with correcting major errors in the control box and driveshaft. The 2019-2020 half-scale team developed the bogie design and functionality using state of the art literature of prior half-scale teams by troubleshooting and optimizing the switching process. We designed a control box that is integrated into the carriage to contain the electrical components. Subsequently, the mechanical design and control sub-teams diagnosed and repaired many components of the bogie. The work of the 2019-2020 half-scale team resulted in an improved design of the steering control arms, the control box, the track, and many mechanical parts.

Description of Designs

Code

The code that the team received at the beginning of the academic year was designed purely for testing and not with practical functionality in mind. The code was designed to only allow the bogie to move forwards and backward and to move the switching arm left and right in large arcs, all following the inputs on a 5 button LCD panel attached to the top of the master Arduino. Additionally, limit switches were set at each end of the switching arm's path. These switches were intended to stop the rotation of the arm to prevent overturning and damage to the gearbox. Functions for reading hall effect sensors and ultrasonic sensors were included in the code. A pair of each of these sensors was to be included on the bogie. The code also contained functions for an autonomous mode of running. This mode, activated by a switch on the bogie's control box, would make the Arduino act upon inputs received from the sensors attached to the bogie when it was not receiving inputs from the 5 button panel.

The team attempted to modify the code to make the bogie behave as it would in a practical application. To this end, the team decided to test whether the code could be modified to move the bogie in a direction while engaging the switch arm. The logic behind this move was that the Superway is intended to function as a quick and effective transport solution, and stopping at every track switch to rotate the switch mechanism would significantly slow the system and create congestion in the network. The team proved that the bogie could move while exercising the switch mechanism, but it was necessary to modify the directional functions on the Arduino's code to do so. The 5 buttons on the LCD panel were on a single circuit, and therefore multiple buttons could not be pressed simultaneously to achieve the desired input. This discovery forced the team to review its options and create plans for a wireless controller, which will be detailed later.

The team also focused on making the limit switches properly functional. It soon became apparent that the code had multiple issues that prevented the Arduino from reacting properly to the readings from the limit switches. The code was designed so that the stepper motors would turn in large arcs with the button presses on the 5 button panel. This resulted in the stepper motor overturning even after hitting the limit switch, despite the presence of an ISR (Interrupt Service Routine) function that was intended to stop it. Additionally, when the limit switch would trigger, the Arduino would stop functioning altogether. These issues were resolved by modifying the stepper directional functions to move the switching arms in small steps rather than large ones and by removing the snippets of code that were discovered to cause errors when the switches were triggered. The ISR was also removed, as the team determined that it was unnecessary. These changes made the limit switches properly functional, but this made the movement of the switching arm slow and unideal for practical use. This was the code used for the trail testing of the switch arm and is found in **Appendix C.**

To fix this another test code was created. This code uses the small step method to home the stepper motor to one side. The stepper then moves between orientations in a single, large movement. This code is based around the observation that the switching arm never needs to be anywhere except on either end of its path, and so never needs to move unless the bogie needs to traverse a track switch in the opposite direction of the switching arm's orientation at that time. The number of steps that this movement needs to traverse has not been determined and will have to be left to a future year to find. This code is in **Appendix C.**

Wireless Controller

Starting the academic year, the bogie was controlled by a 5 button LED panel attached to the master Arduino. The 5 buttons were on a single circuit, which prevented users from pressing them simultaneously to achieve the desired input. The team wanted to preserve the 4 movement functions already present in the code, which would move the bogie forwards and backward without moving the switch mechanism and could move the switching arm left and right without moving the bogie. These would be complemented by 4 more functions that would move the bogie forwards and backward while moving the switching arm left and right. This would require 8 functions in total, more than the 5 button panel had space for. To this end, the team decided to use a joystick to control the bogie. This joystick would be divided into 8 directional categories, with the 4 functions to move the biggie while engaging the switching arm becoming diagonal inputs and the 4 original inputs becoming the cardinal direction inputs. This combined with the plan to hang the control box underneath the bogie created a problematic setup. The team realized that attempting to control the bogie using a joystick that was moving with the bogie would be cumbersome, and introduced the wireless controller to make inputting commands easier.

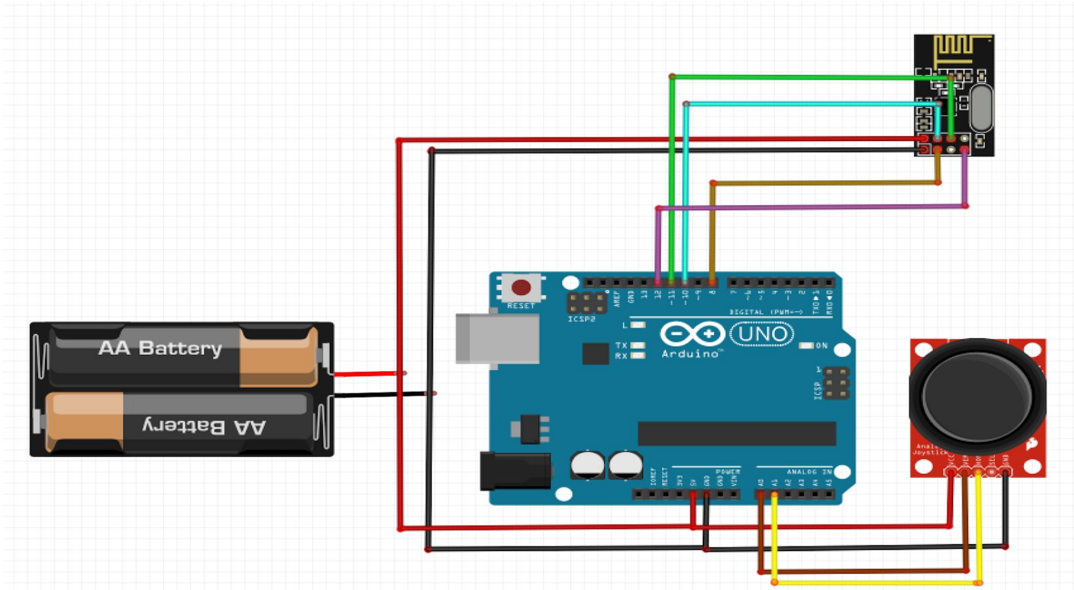


Figure 2: Schematic of Wireless Controller with Joystick

The wireless controller is intended to be a handheld device that remotely controls the bogie. This controller would utilize an nRF24L01 wireless communication module to communicate with the master Arduino. Originally there was a third Arduino that was to act as a go-between for the 2 systems and take sensor inputs as well. This configuration is shown in **Figure 29** in **Appendix B**. This setup was later deemed unnecessary as the master Arduino had the available pins to accommodate the nRF24 module and due to concerns that the presence of the third Arduino created more room for problems to arise within the circuit. Later design changes combined the wireless controller idea with an idea set forth by the previous team that utilizes an Arduino on the bogie that handles all sensor inputs and relays them to the master Arduino using the nRF24 modules. The combination of these two ideas would yield a setup that allows the master Arduino to switch between reading inputs from the controller and inputs from the Arduino on the bogie depending on the mode the Arduino is in. The bogie Arduino and the controller Arduino would transmit on different frequencies, and the Master Arduino would change the frequency it receives when a switch is flipped or a button is pressed. A partial schematic of the master Arduino setup corresponding to the controller schematic can be located in **Appendix B** as **Figure 30**.

System Controls

The control system the team received at the beginning of the academic year was comprehensive, but some of the hardware was not applicable at the current phase of the half-scale project. The main functions of the control box that the previous team created were to run the bogie forward or backward and turn the switch arm left or right. At the beginning of the academic year, the code from last year was only able to move the bogie and the switch arm. Some of the build-in hardware was not coded and did not have any meaningful functionality at the current phase when the switch arm was not tested. The previous team was forethoughtful to include future implementations of the control box, **Figure 31** shows the control box from the previous year can be found in **Appendix B**. The hardware for the autonomous mode of the bogie is built in the control system. Because the switch arm has not been tested for its functionality yet, having extra components in the control system had made the circuit analysis much harder when problems were showing up. Since the team had set the goal of this team to be making sure the project had a functional switch arm mechanism, the team removed the hardware that is currently not being used.

To ensure that the bogie can be tested at the Y-intersection with a functional switch arm, a 24 volt DC motor is used to drive the bogie and a 12-volt stepper motor is used to switch the arm. The 24 volt DC motor will be controlled by a four-channel relay and the 12-volt stepper motor will be controlled by the stepper motor driver. The control system will be commended by an Arduino. The control system will be run under three different voltages, 5 volts, 12 volts, and 24 volts. The 12-volt power source is powering the Arduino and a cooling fan, the 24-volt power source is powering the 24 volt DC motor and the 5-volt power source is provided by the Arduino and is powering the stepper motor and all the sensors in the control system. **Figure 3** shows the modification of the control box from this year.

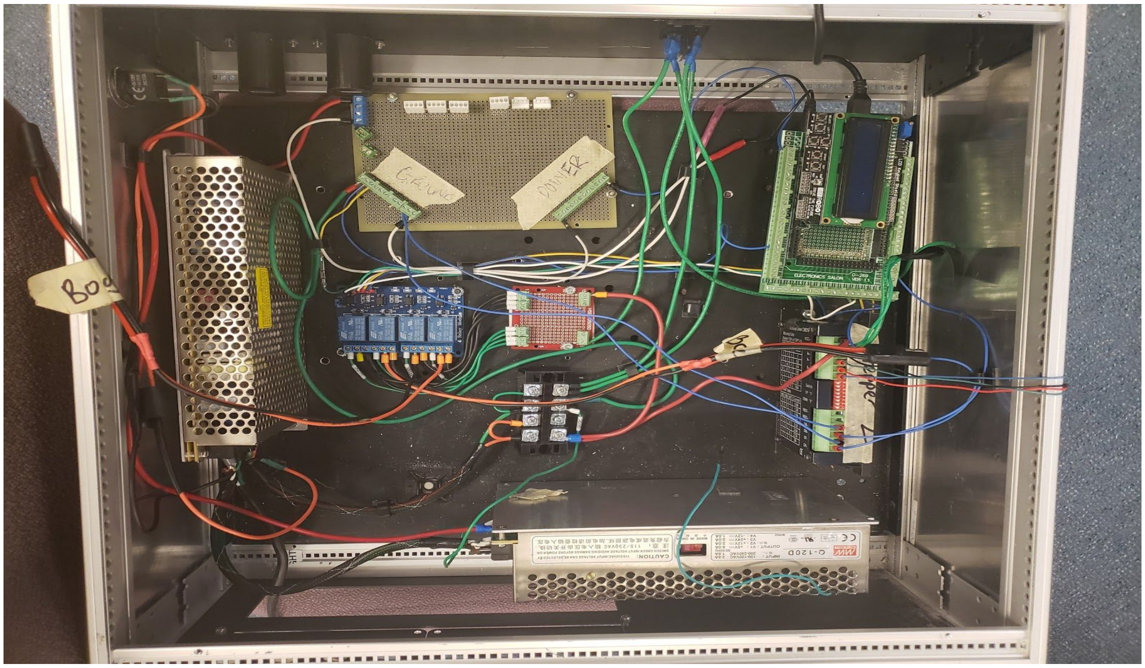


Figure 3: The modification of the control box this year

Besides the component mentioned in this paragraph hall effect sensor, ultrasonic sensor, and limit switch are used for executing the autonomous mode, the limit switch is used to stop the switch arm from overturning. The hall effect sensor and ultrasonic sensor are used to monitor the speed of the wheels and the distance between bogies, and it is removed in the current phase to keep the control system simple. Some parts in the control system are soldered for connections, and it was hard to replace when things were not functioning properly. The team replaced the soldered circuit board with ramps connectors so that components will be easily disassembled or replaced.

Control box

The control box the team received from the previous year is a modified computer case to fit all the components, **Figure 31** shows the control box from the previous year can be found in **Appendix B**. The computer case has drilled holes and cut out sections to secure the components and to fit all the plugs that are needed. Since all the components are screwed into the sideboard of the computer case, the team had a hard time working around the case to take individual parts out. The prototype of the new control box was designed to be easily assembled. The prototype of the first redesigned control box used PVC as the main materials because it is easy to assemble. **Figure 32** shows the PVC prototype of the control box can be found in **Appendix B**.After the prototype for the control box was made, the team decided that the box was too big and too heavy. While the team had moved on to redesign the prototype, the Covid-19 pandemic had ruined the plan of manufacturing of any prototype and the redesign had shifted to design-focused instead of manufacturing.

Since the design limitation had been freed from manufacturing, the final design of the control box has the shape of the carrying compartment of the Spartan Superway project. **Figure 4** shows the final design of the control box. The control box had four main parts. The main body will be simulated as the cabin of the compartment to carry weights. The two 12-volt batteries will be placed in the main body since the batteries are the heaviest among all the components. The side cover can be taken off to remove the control system from the box. The top cover is designed to be made from clear polycarbonate plastic because the plastic needs to be strong enough to hold 600 pounds of desired weight. A guideway is designed so that the top cover can be slid in and aligned with the rest of the parts. **Figure 33** shows the guild way of the top cover from the final design of the control box can be found in **Appendix B**.

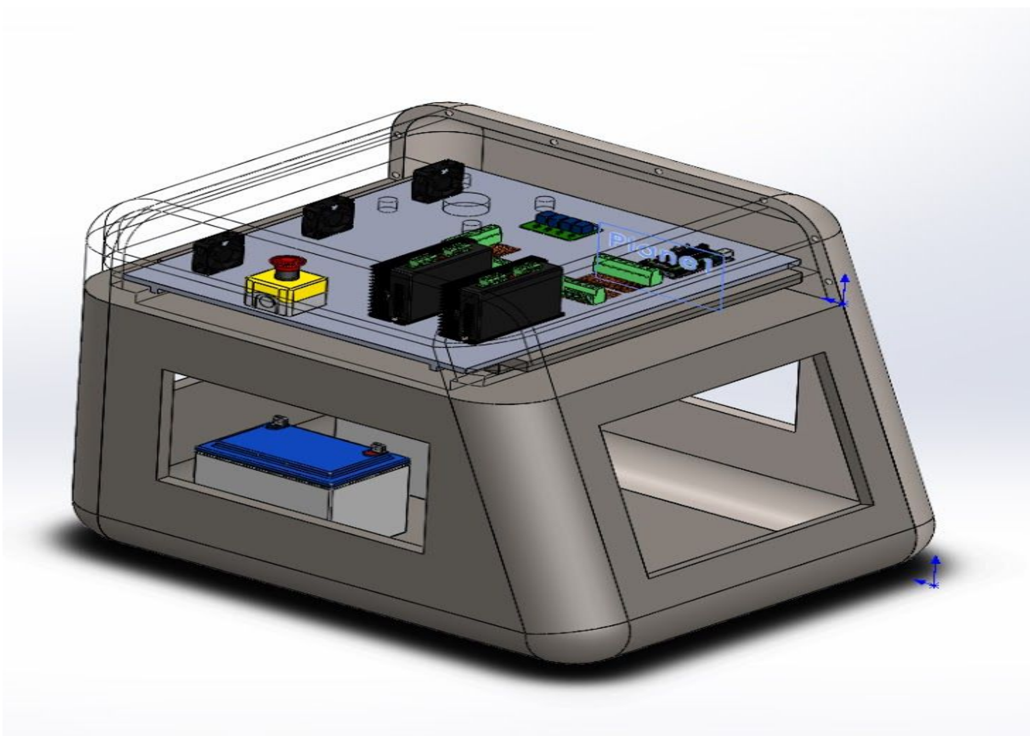


Figure 4: Prime design of the control box

A component board with two legs is designed to slide into the top cover and to be held in place. **Figure 33** in **Appendix B** also shows the legs design of the component board from the final design of the control box.The component board will hold all the control system components. **Figure 5** shows the exploded view of the final design of the control box.

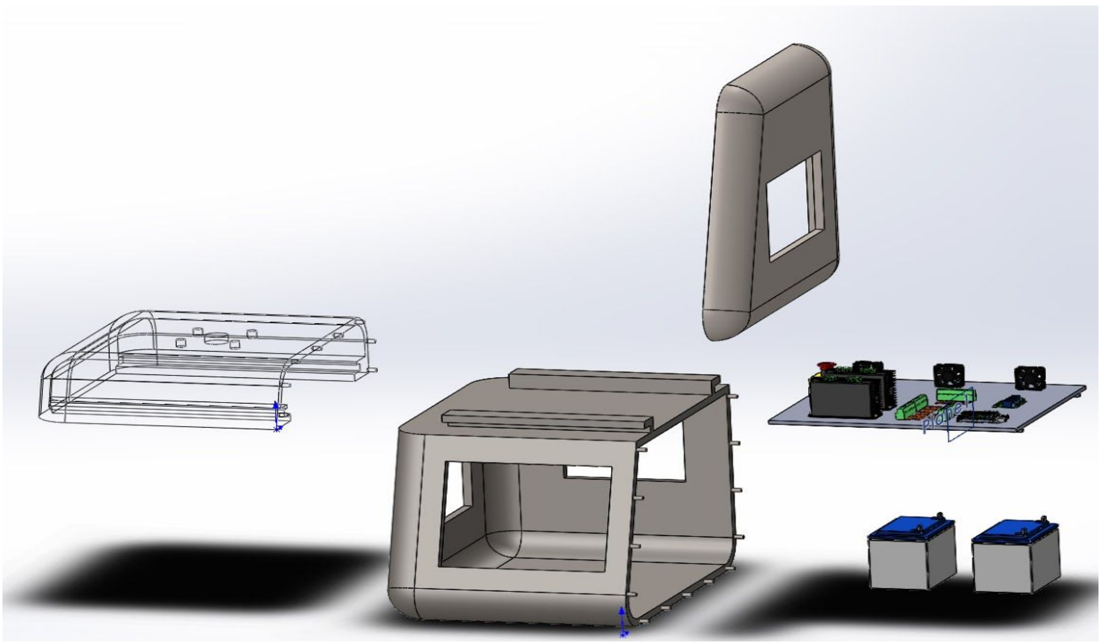


Figure 5: Control box (exploded)

Connection Arm

The previous team had designed a stretchable shelf to hold the control box and connect the compartment to the bogie. **Figure 34** shows the stretchable shelf design for the control box from the previous team can be found in **Appendix B**. Due to the change of facilities last year, the connection shelf can not be found in the Spartan Superway Design Center. Since the control box is designed to be the compartment and to hold the control system, a new connection arm is designed to connect the control box to the bogie. **Figure 6** shows the connection arm design connecting the control box to two bogies. A two degree of freedom rotational connection is designed to keep the compartment horizontal when the bogie is going through slope or tilting. The connection arm is made of three main parts, the base, the cross-connection, and the connection bar. The base will be screwed into the top cover of the control box. Two opposite arms from the cross-connection will be connected to the base and the other two will connect to the connection bar. The connection bar is designed as a triangle shape because two bogies are used to carry one compartment. **Figure 7** shows the full design of the connection arm.

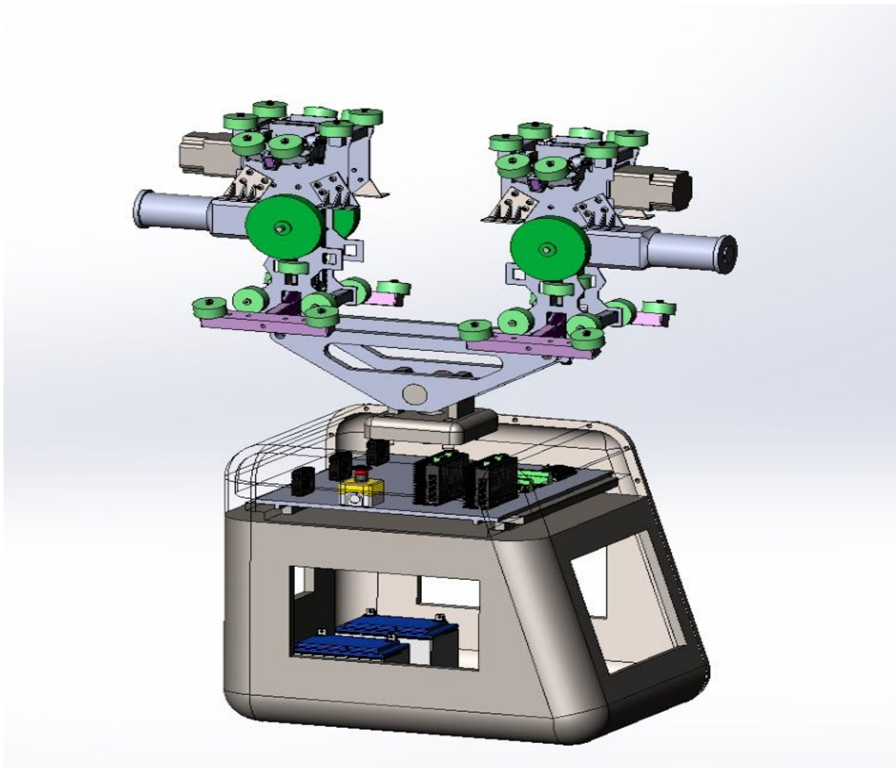


Figure 6: Control box with connection arm and bogies

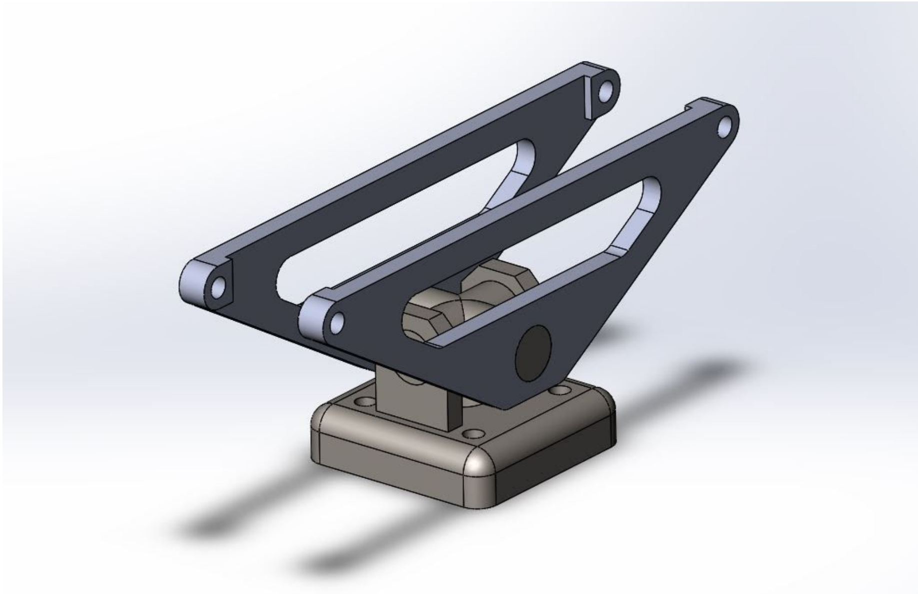


Figure 7: Prime design of the connection arm

Switch Arm Assembly

The main function of the switch arm mechanism is to clamp/grip onto the tracks to hold the bogie in place while traversing. After inheriting last year's design of the switch arm, we decided to address two main issues with that design. Firstly, the wheels of the switch arm did not properly grip onto the tracks, which caused the bogie to fall off. Secondly, the geometry of the switch arm was such that as it was tilted to one side to clamp onto the track, it was found through multiple trials that the wheels were not consistent in maintaining contact with the track. To address this, our prime design of the switch arm is functioned to ensure that the wheels of the arm reliably grip onto the track and the switch arm was made to be adjustable so that the wheels maintain contact with the track at all times. As can be seen from **Figure 8**, we designed an "E-shaped" bracket on each side of the switch arm.

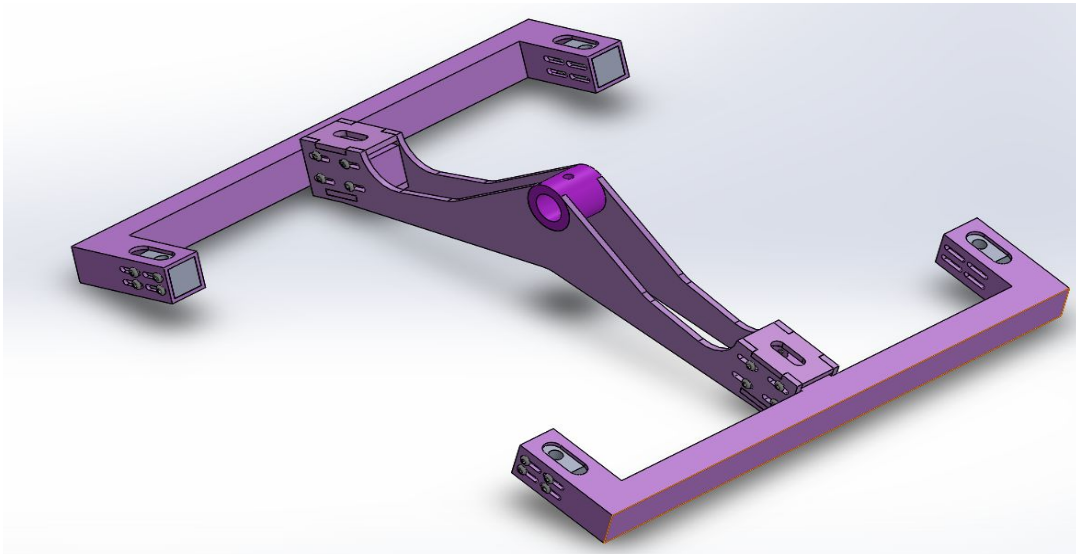


Figure 8: Prime design of switch arm (assembled)

At each end of the E-shaped bracket, a wheel is placed. After multiple iterations, we concluded that using two wheels instead of only one wheel provides the best form of traction. One of the other unique features of this switch arm beside the E shaped brackets is the use of slots. With two slots placed on each end of the switch arm, the wheels are attached on top of these slots, allowing them to be adjustable to maintain contact with the side surface of the track. Additionally, one slot on each side is placed in the middle of the E shaped bracket. This is to ensure that the overall E shaped bracket can be adjustable if it is too close/distant from the track. In the exploded view in **Figure 9**, it is shown that our design is easy to assemble as well as manufacture (Please refer to **Table 2** in **Appendix B** for the specified components). For the assembly, we used the same control arm bracket as the previous team (denoted by component G in Fig. 9), the slots are represented by component D. Additionally, components A, D, F, and E from the exploded view were replicated on each side of the switch arm, showing the ease of assembly and not worrying that each of these components has different dimensions.

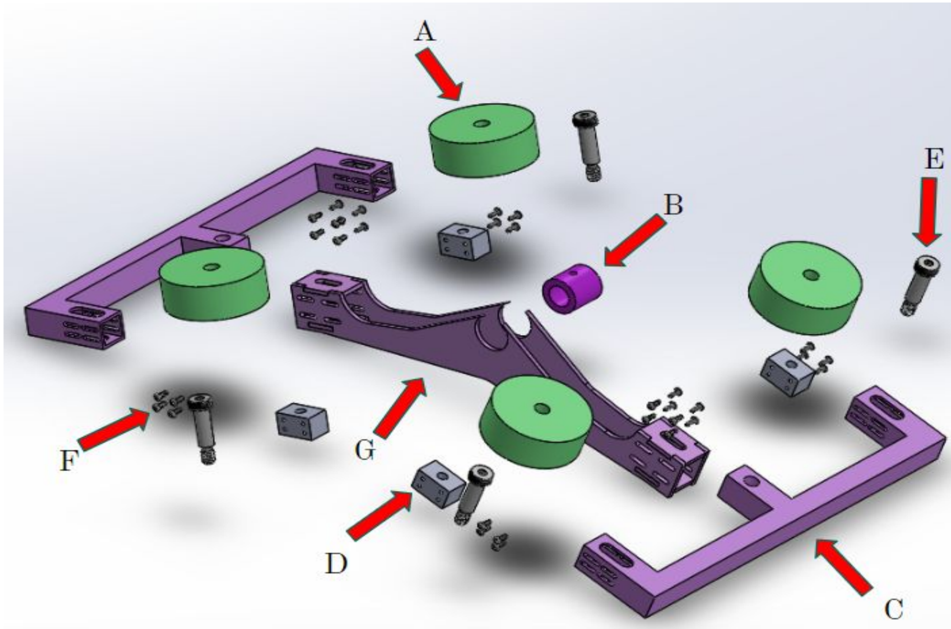


Figure 9: Prime design of switch arm (exploded)

To ensure that our switch arm is conservative in terms of space and efficiency, the dimensions of the E-shaped bracket are 11.00 inches x 2.30 inches x 0.80 inches with symmetrical tolerances of 0.02 inches, shown in **Figure 10**.

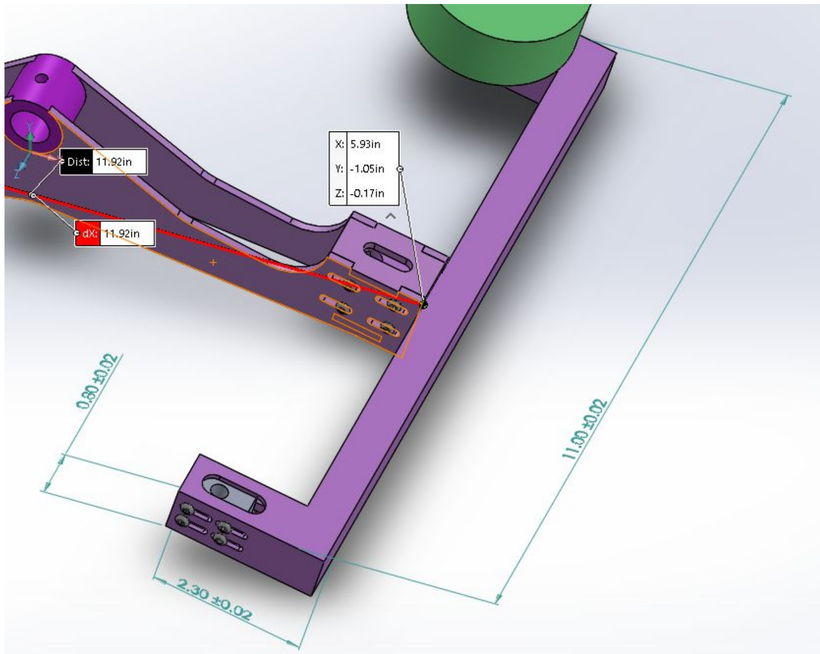


Figure 10: Dimensions of the e-shaped bracket

As a blueprint for next year's team to replicate, the prime switch arm design is placed on the modified bogie to show the full assembly of our bogie mechanism as pictured in **Figure 11**. For the upper control arms of the bogie, we adopted the “two-wheel” method which has been the prime concept for our switch arm prototype designs. With the use of a T-shaped bracket, a wheel is placed at either end, resulting in a stronger grip on the upper portion of the track.

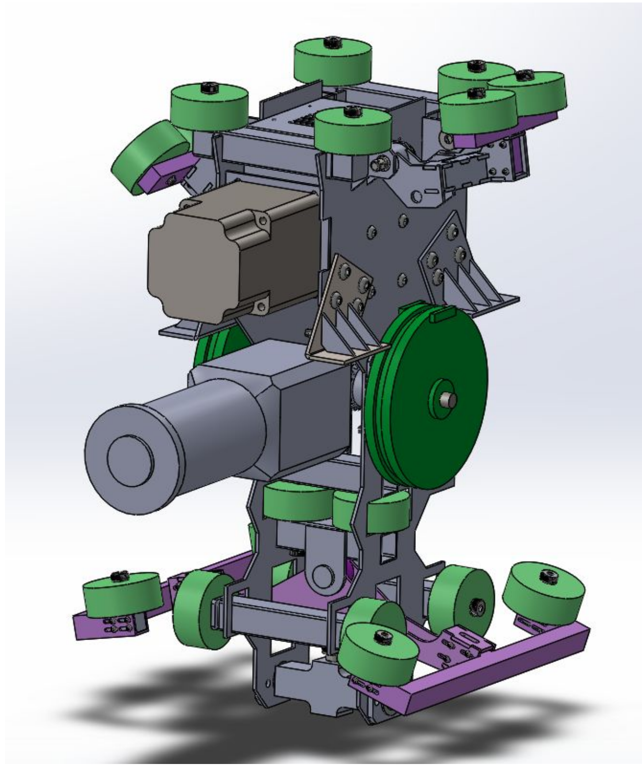


Figure 11: Updated main bogie assembly

Half-Scale Track

To successfully test our bogie, the half-scaled portion of the track from previous years was used. We replicated a portion of the track assembly in the Half-Scale room in SolidWorks (as can be seen in **Figure 12**) and modified the dimensions of the track to accompany our prime switch arm design as well as the bogie. In our simulation, the total length of the track from one end to the other is 202 inches (approximately 5 meters).

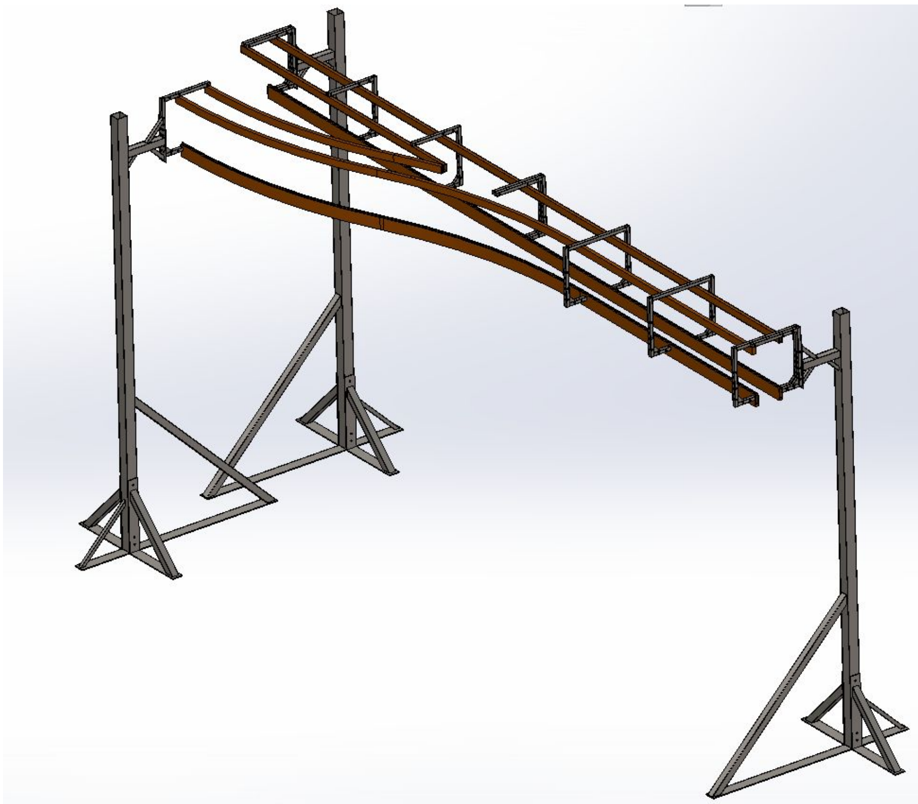


Figure 12: Half scaled track iteration

As the left side of the track splits and curves, the thickness of the left side (1.30 inches) is more than the right side (1.15 inches). The use of the inverted C-shaped brackets which is distributed along the track firmly holds the rails in place while being supported by an elevated structural steel beam at each end. We also increased measurements of the track by .35 inches for a total height of 3.35 inches, as seen in **Figure 13.**

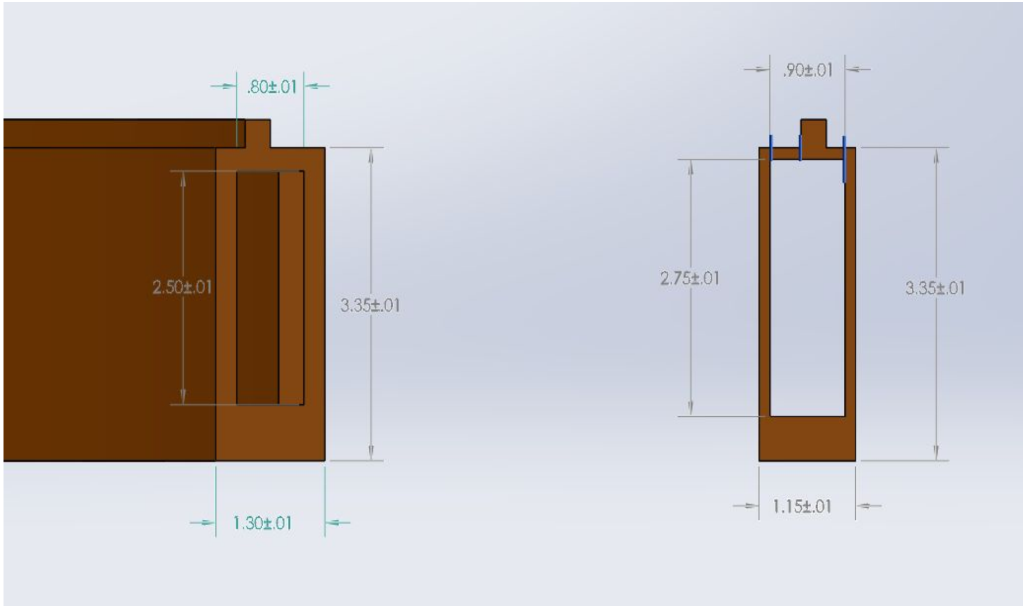


Figure 13: Half scaled track dimensioning

Since the Half-Scale track has already been manufactured, another design solution would be to add a solid steel bar and weld it to the bottom of the track **Figure 14**. The dimensions for that steel bar would be 0.35 inches x 1.30 inches for the left side of the track and 0.35 inches x 1.15 inches for the right side of the track. This steel bar would be extended to the whole length of the track (202 inches).

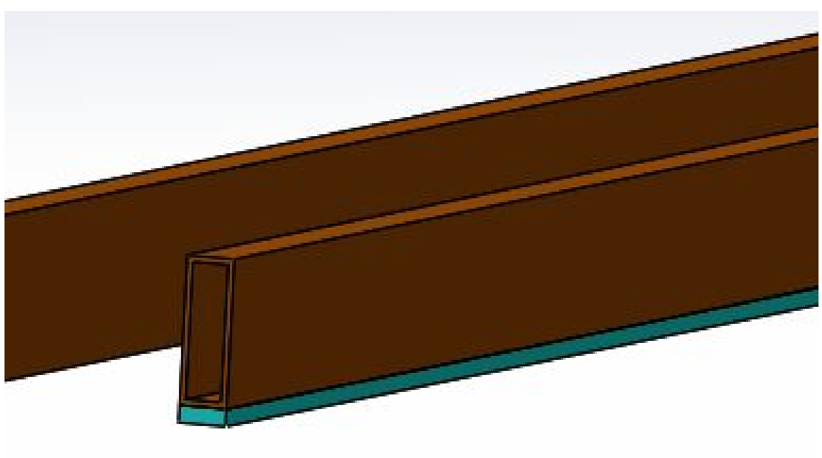


Figure 14: Alternative design for the track assembly

Analysis: Validation & Testing

Code and Wireless Controller

The iterations of the code were tested on the bogie. Through this, the team discovered that improving the code so that the system could achieve a more realistic functionality would require changes to its design. To serve this end several variations of the code were created. Included in these variations were the prototype codes for the wireless controller. Testing of whether the bogie could move while engaging the switch mechanism was rather simple, as the code to do each function already existed. The functions were simply combined under a single function name. After the discovery that the bogie could carry out this new function, the focus shifted to creating a control setup that could hold all 8 desired functions. This led to the wireless controller, which was first tested through attempting to send signals between two Arduinos using the nRF24 module. Following the successful test, one of the Arduinos was connected to the master Arduino in an attempt to complete the controller.

Variants of this design were tested in attempts to make the controller functional. After the redesign of the controller, it was determined that the master Arduino could function as the receiver for the signals sent out by the controller, and that discovery eliminated the Arduino that was connected to it. When the controller still could not work research led to code changes and to other solutions that were never realized. The controller was set aside due to time constraints and the mechatronics team switched its focus towards creating a viable test code for the prototype switching arm.

The creation of test code for the redesigned switch arm required the limit switches to become functional. Through iterative testing, it was determined that the code for the stepper motor and for the limit switch functions needed to be changed. That same iterative testing led to the conclusion that the ISR function that was written to trigger when the limit switches were activated was not effective in this situation. Further research into applications of this function confirmed this and the code was changed to remove unnecessary lines and to shorten the movements of the stepper for more accurate positioning of the stepper motor. This was necessary for trial testing of the switching arm.

Control System and Housing

The goal of the team this year is to design and test a functional switch arm. To achieve the goal this year the control system has to be fully functional with the code. The control box is designed to run two bogies at the same time, but only one bogie was tested at the end of last year. The two stepper drivers and the four-channel relay are used to control two separate sets of 24 volt DC motor and 12-volt stepper motor. One of the sets was programmed to run in autonomous mode and was not able to be controlled by buttons on the control panel, and one of the stepper drivers was not responding well to the commands. Since the control system did not respond to the commands as expected, the team suspected that the hardware of the control system might have issues.

The team used a multimeter to test the continuity of each wire and components of the system and found out that one of the pins in one of the stepper drivers does not have current flow. Because only one of the stepper drivers is functional and the shelter in place order had extended the time for ordering and receiving new parts and the autonomous mode is not ready to be used, the team decided to only test the switch arm in one bogie. The team had removed the sensors that are used for autonomous mode and replace the soldered wire with ramps connectors. The team also finds out that one of the Arduino pins that are in use has connection problems and is not outputting any signals to the stepper driver. After replacing the wire and connectors and changing the pins in the Arduino, the control operates and responds the way the team expected. The control box is redesigned to hold 600 pounds of weight and to contain all the hardware components that are needed. Because of the shelter in place order from the Covid-19 pandemic, the control box is not manufactured. FEA analysis is applied to the control box to ensure the control box will be able to hold the desired weight. **Figure 15** shows the FEA analysis result for von Mises stress and **Figure 16** shows result for factor of safety. Most parts of the control box are designed to be steels and the top clear cover is designed to be polycarbonate. A simulation of 600 pounds of desired weight is applied at the bottom of the control box. The minimum factor of safety of the control box is about 7 which means the material and the design will hold the desired weight.

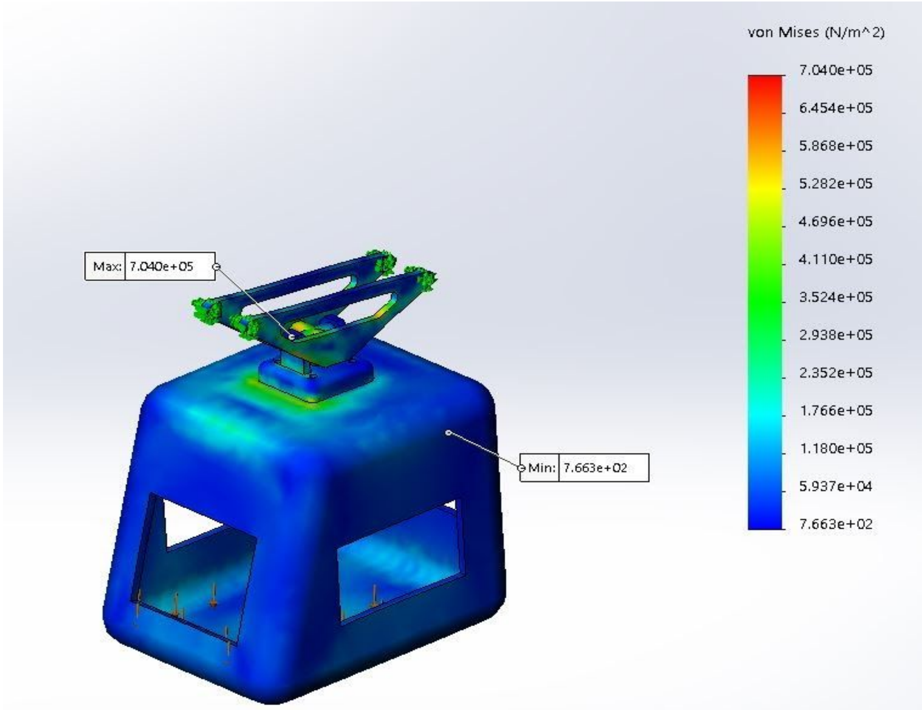


Figure 15: FEA analysis for von Mises stress

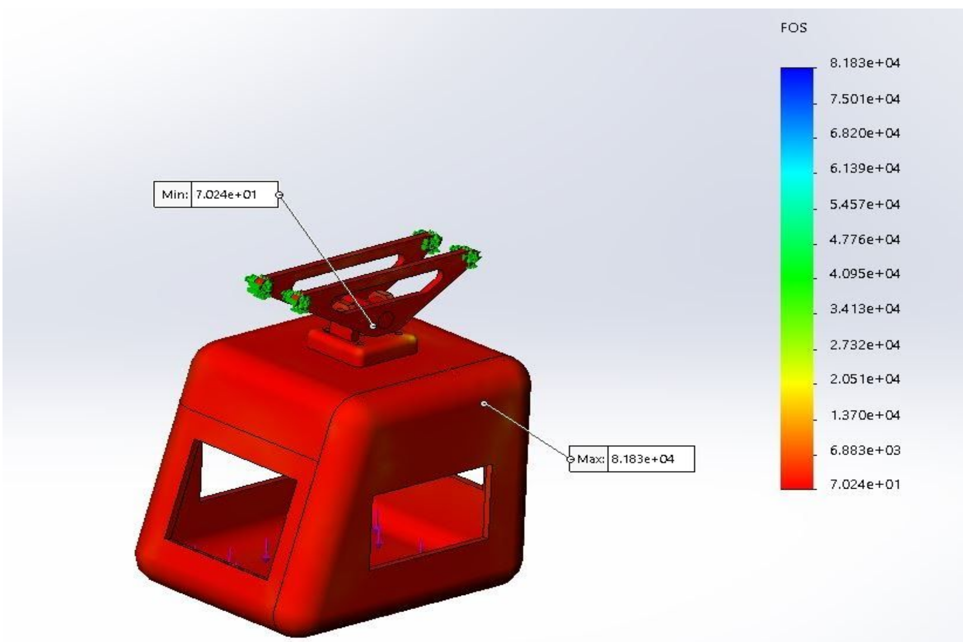


Figure 16: FEA analysis for factor of safety

Switch Arm Prototype Trial Testing

The primary design concern for the existing bogie is that it has been unable to successfully traverse the intersection at the specified speed. The objective of the project calls for the bogie to cantilever on one of the drive wheels while being kept upright by the suspension wheels. This design has a high rate of failure at speeds surpassing 5 MPH. Considering this issue, modifications to the switching mechanism and additions to the track were necessary to allow the bogie to stay grounded while navigating through intersections. The design specifications of each prototype were influenced by trial testing performance aspects.

Trial Testing: Segment One

The prototype pictured in **Figure 17** was a suitable solution to the bogie shifting during linear to fork transition testing, although the design was a useful base to the prime design, the model would be extremely difficult and costly to manufacture.

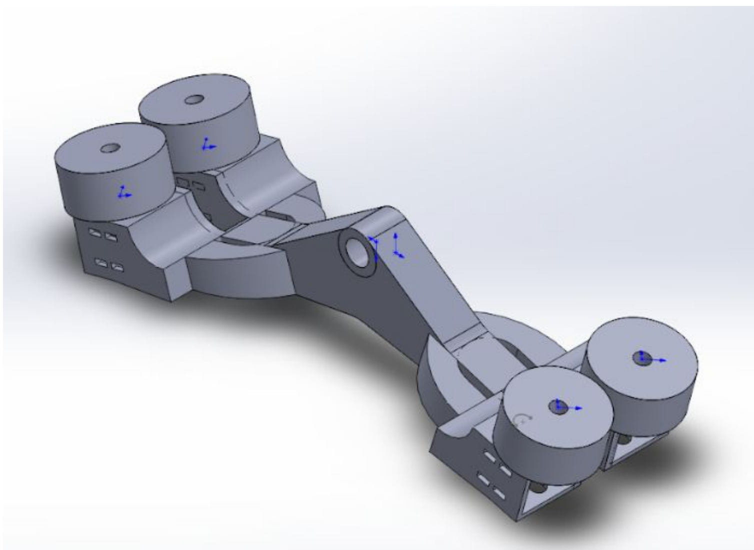


Figure 17: Prototype 1

Trial Testing: Segment Two

Prototype 2 was more of a broad approach to solving issues of ideal distance between the wheels to appropriately carry the load of the bogie as it cantilevers the track. The wood model was the most useful for real world testing and analysis. Prototype 2 is pictured in **Figure 18**.

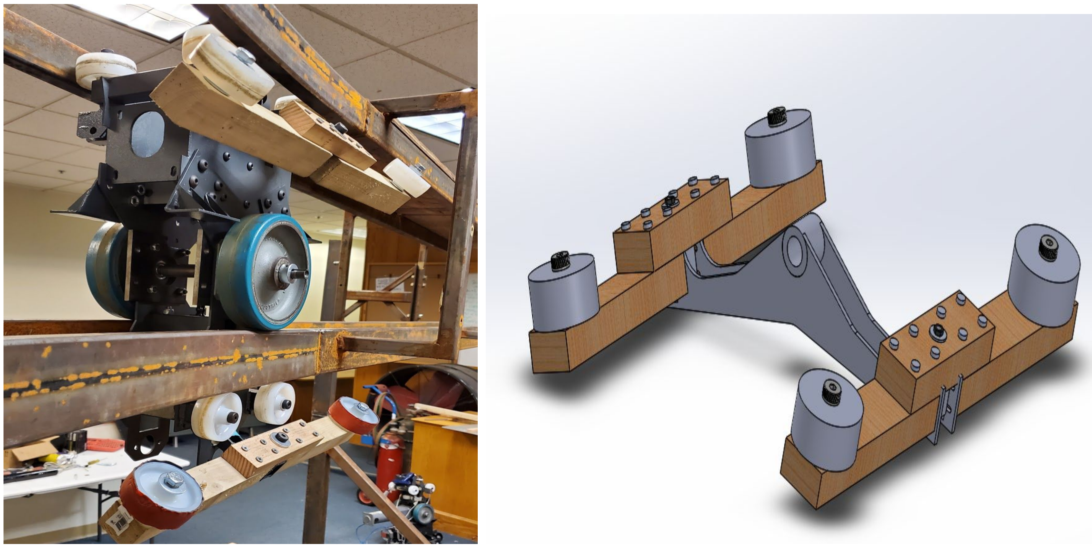


Figure 18: Prototype 2 (wood model)

Trial Testing: <u>Segment Three</u>

This model inspired the design of the rapid wood prototype. The prototype was given several adjustment points to modify length, height, and width on demand. The design of Prototype 3 can be seen in **Figures 19.**

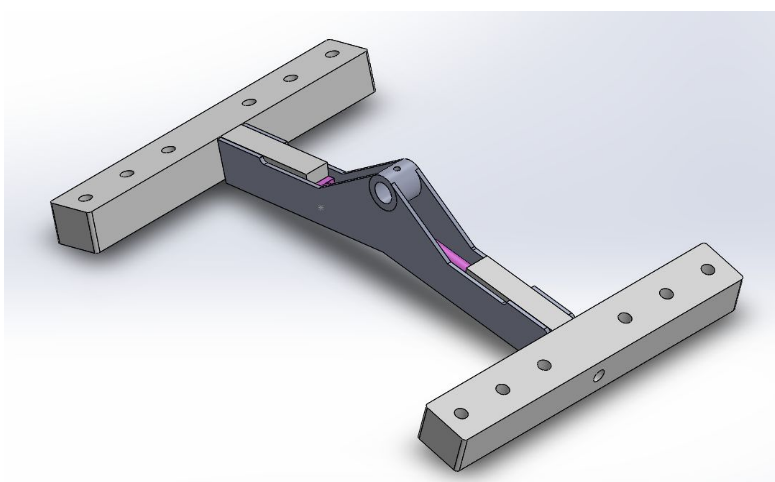


Figure 19: Prototype 3 (specialized model)

This prototype was also promising, but failed under the standard load of the switch arm gears. The summation of these stresses and the displacement analysis for the secondary prototype can be seen in **Figures 20-21.**

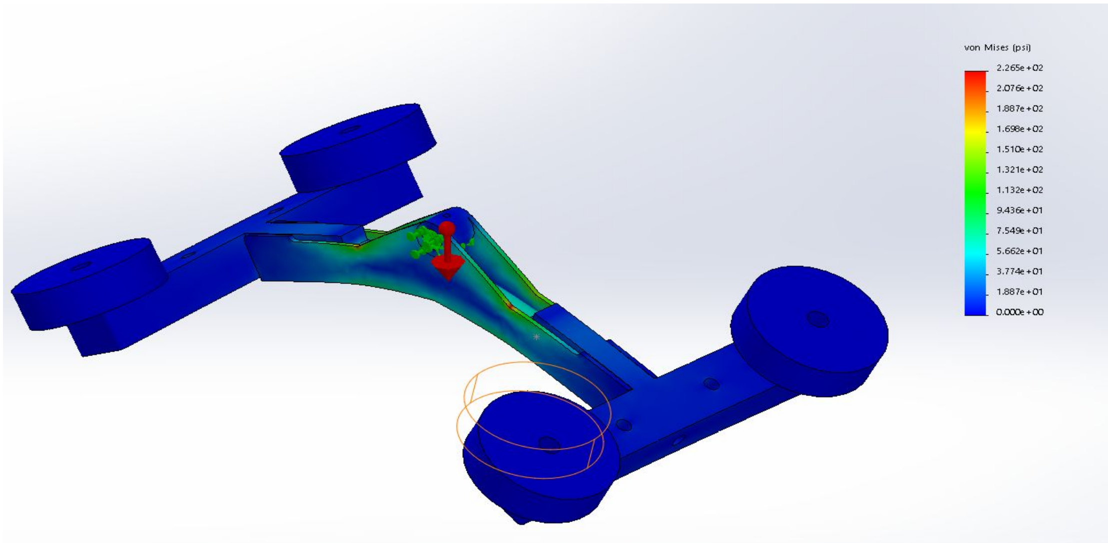


Figure 20: Prototype 2 Von Mises Stress

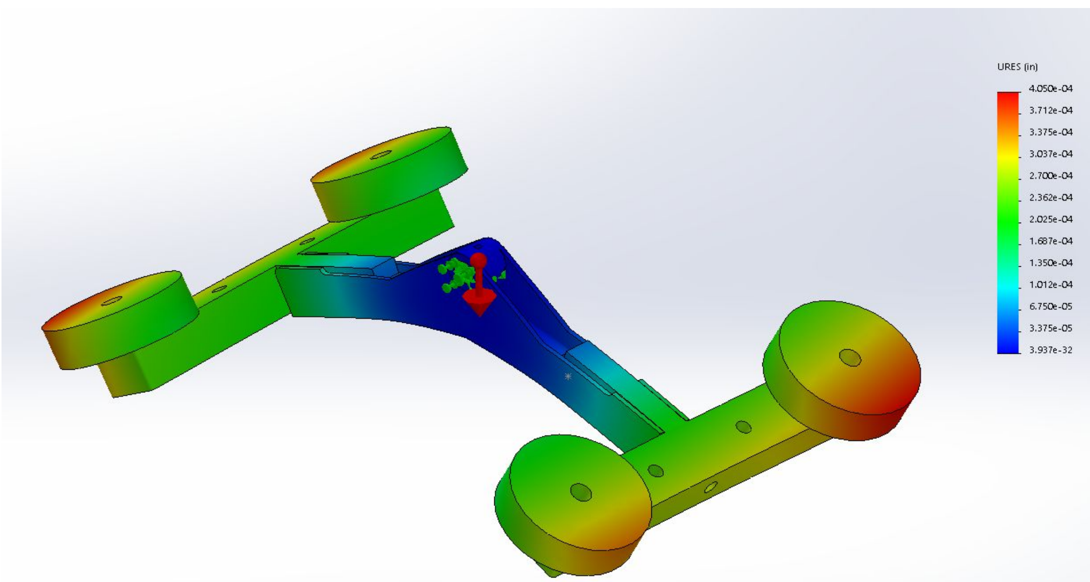


Figure 21: Prototype 2 displacement

Trial testing: <u>Segment Four</u>

The prime design allows for real-time adjustments to height, length, and phi angles. This design incorporates the aesthetics of the current bogie switch arm setup and is tolenced in such a way that the “e brackets,” were a direct fit to the previous models square insert hole which is installed on the bogie at the moment. The final design of the switch arm in **Figure 22** was based on our final trial testing segment.

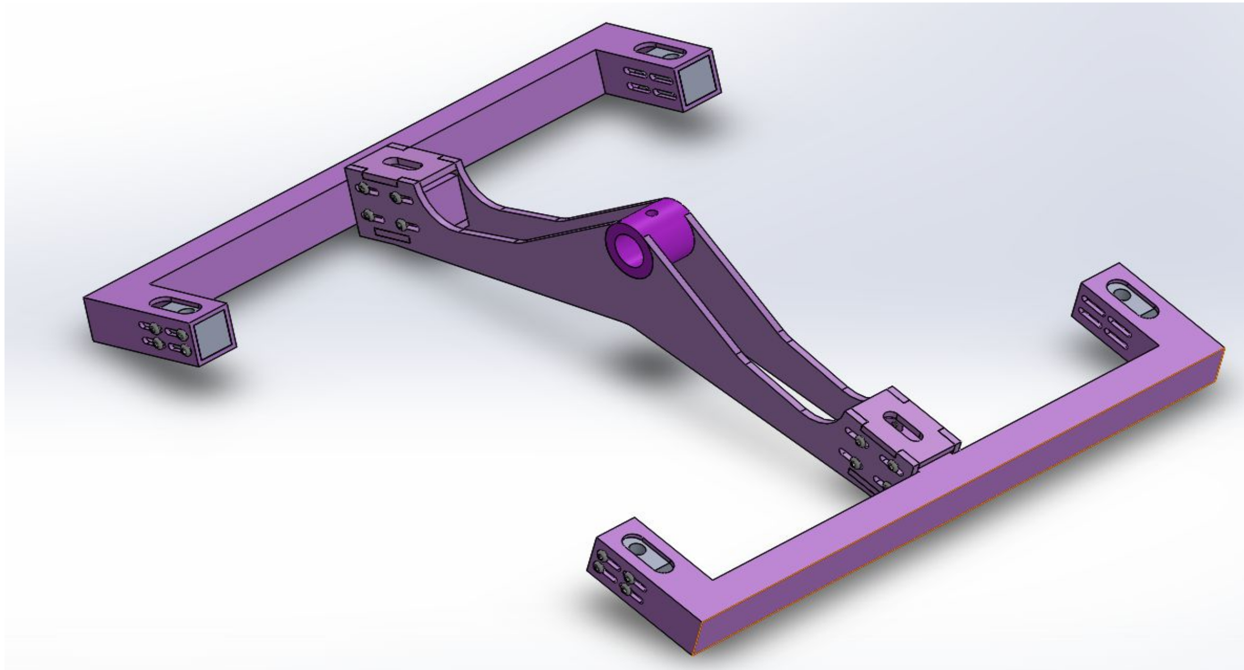


Figure 22: Switch arm prime design

The bogie is able to traverse the track reliably using the adjustment screws and slots. The right side CAD in **Figure 23** simulates how the adjustable slots allow the wheels of the bogie to securely clamp onto the track.

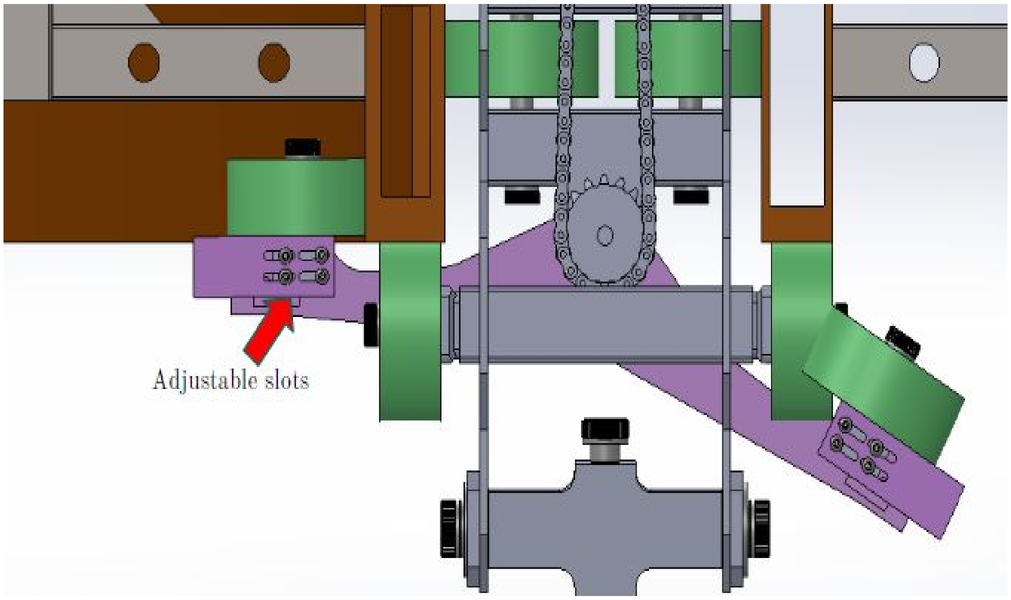


Figure 23: Function of using adjustable slots on the switch arm

Since fabrication has been curtailed this semester due to the shelter in place order, we were not able to manufacture the prime design of the switch arm. However, through the use of simulations and FEA analysis in SolidWorks, we are able to analytically represent our model and show its efficiency in carrying out its desired functions before manufacturing.

For the prime design of the switch arm, we conducted an FEA simulation test for its resultant displacement as seen in **Figure 26**. The von Mises stress distribution plot, seen in **Figure 25**, and the overall factor of safety (FOS) plot, shown in **Figure 24**, were then generated. We simulated the switch arm in a setting where it is in its neutral position (not tilted to either side) and a fixed constraint was placed in the center hole. **Figures 24-26** show the resultant plots of FOS, von Mises stress, and displacement. We then applied a gravitational force load on the arm, basing this simulation on how much the arm will deform based on its weight. Finally, we refined the mesh to acquire more accurate data within our plots.

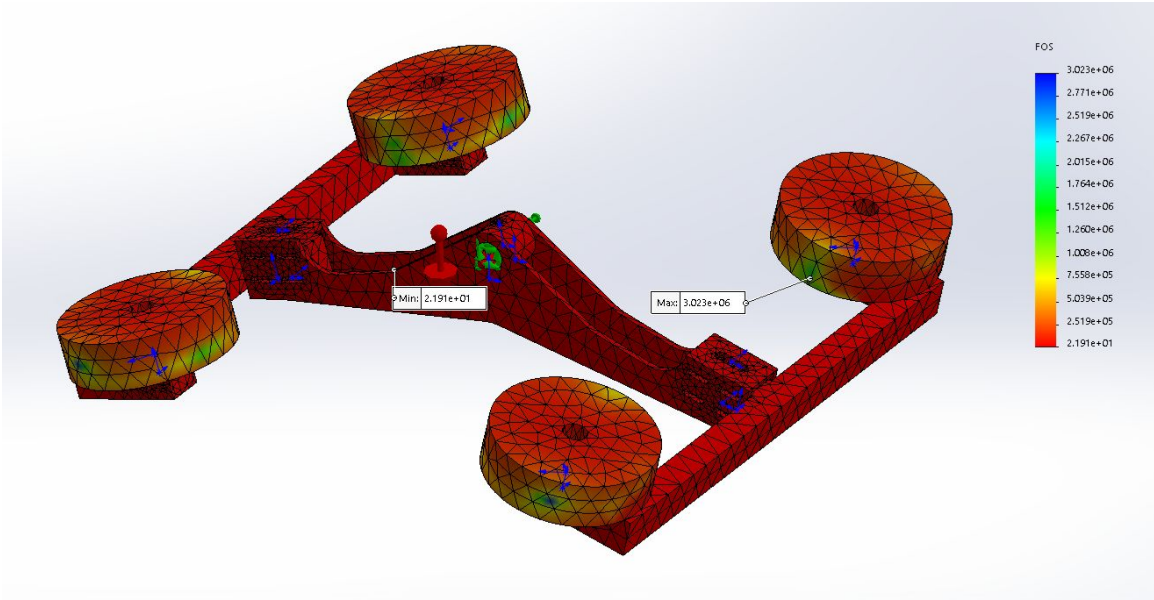


Figure 24: Factor of Safety of prime design

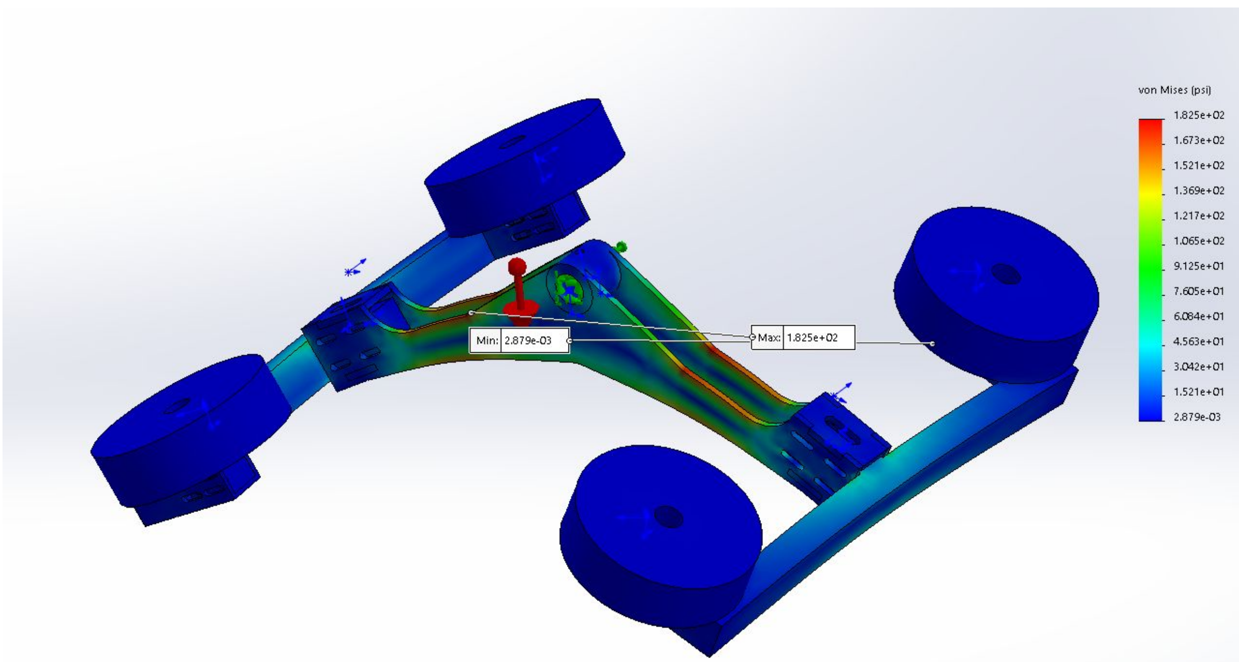


Figure 25: Von Mises Stress of prime design

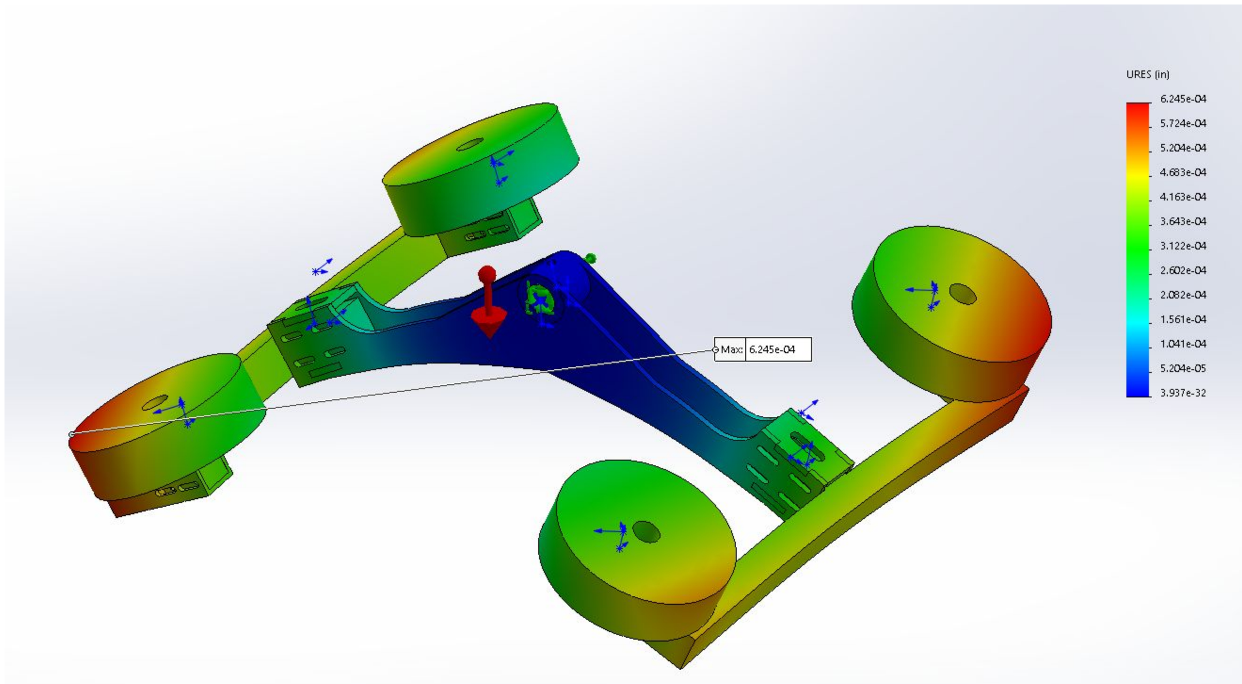


Figure 26: Deformation of prime design

Once the materials and means of fabrication become available again, we would send the CAD drawings of the prime design of the switch arm to a manufacturer. Since we used the control arm frame and the cubic slot connections from the previous team, the only parts that need to be manufactured are the E-shaped brackets with the hollow slots.

Half-Scale Track Design

The main issue of the current Half-Scale Track is that there is not enough clearance for the switch arm to traverse on the track without colliding with the C-shaped beams (**Figure 27**). As a design solution, we have decided to add 0.35 inches to the width of the track and extend it. To validate our changes to increase the width and thickness of the track to accommodate our switch arm design, we simulated our theoretical model in SolidWorks.

Figure 27: Interference between the switch arm and the c-shaped beam

Before mating our bogie onto the track, we ensured that the dimensions of the track fully accommodate the dimensions of the bogie and that each of the wheels on the bogie touches the track. After making the desired dimensional changes to the rails, we obtained a clearance of 0.51 inches from the top of the stainless steel shoulder screw to the bottom filleted edge of one of the C-shaped beams on the track shown in **Figure 28**.

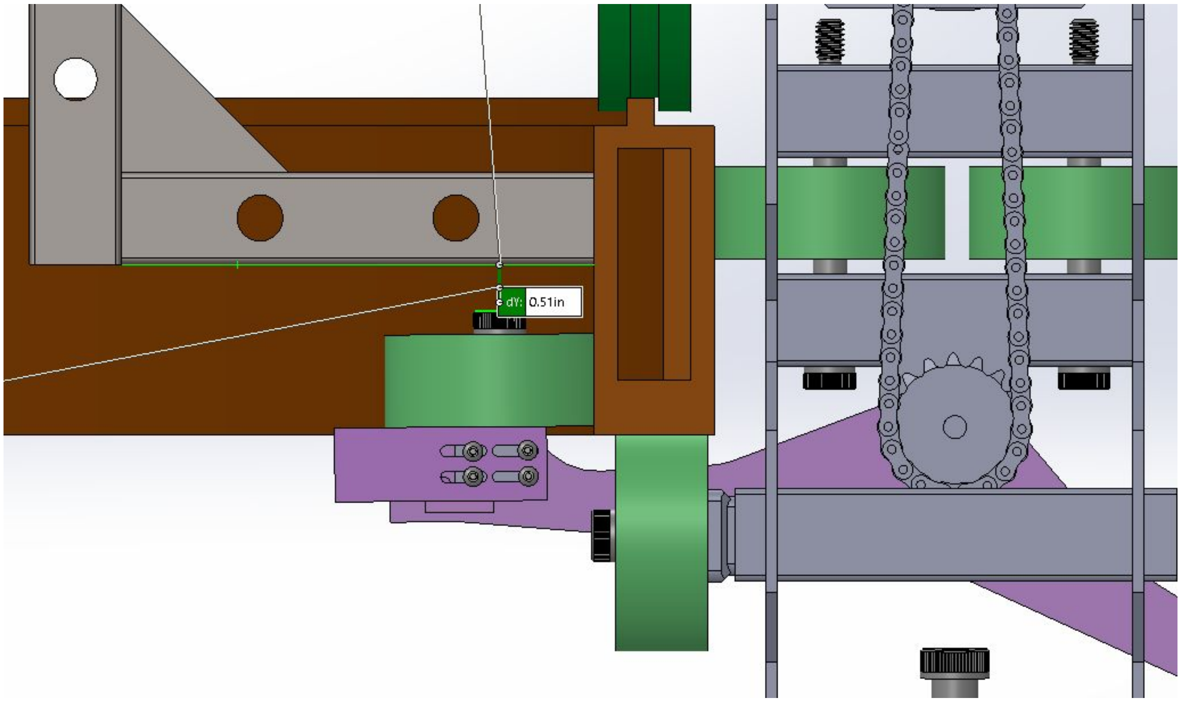


Figure 28: Half scaled track with bogie clearance

Results and Discussion

Code and Wireless Controller

The testing of the operational code for the controller verified that the 5 button LED panel would need to be replaced to allow the bogie to run in a practical application. The reason for this is that the 5-button panel is only useful for determining whether individual functions of the system work as intended. It is not a good option for controlling the bogie in a practical application. The testing also revealed that integrating any wireless functionality would be difficult but necessary. The end goal for the system is to accomplish autonomous functionality while carrying the control system underneath it in a safe and secure method. This would stipulate having as few wires running between the bogie and controller as possible and would also imply the need for a closed control box. Both of these would require wireless communication, as it lessens the number of wires running between the bogie and the control system and allows the system to be controlled externally without the need for an opening to reach into. To finish the control box, either the wireless controller or the idea from the previous year's team of an Arduino atop the bogie sending sensor inputs to the master Arduino would need to be completed.

The change of the stepper motor functions to move the switching arm in small increments had 2 major benefits. Firstly, it made the stepper motor stop upon triggering the limit switch, which made testing of the switch arm possible. Secondly, it allowed for the creation of the homing code. Homing of the switch arm is a smoother and quicker way to turn the switch mechanism to desired positions without the worry of overturning. Unfortunately, the code was not able to be tested due to the emphasis on the trial of the switching arm, the completion of which had been this team's primary goal from the start of the academic year. Due to the code's positive reception when first presented, a future team would be recommended to test and polish this code, and to integrate the aforementioned wireless functions to it so that the bogie could be controlled

Control system and Control Box

The control system had no issues operating after the diagnosis and refurbishment of some of the components. Following the development of the code and control system, the switch arm prototype was able to be tested on the Y-intersection of the track. The team successfully utilized the control system to test the switch arm functionality. During this academic year, the team removed some components used for autonomous mode. After diagnosing the control system, not all broken parts were replaced or installed. The Arduino pins have connection issues that weren't a priority to be replaced because of the shelter-in-place order and shipping limitations. The simulation for the control box offered a factor of safety of 7.00 under a load of 600 pounds. The result of the simulation shows that the control box is capable of supporting more weight than what the team had anticipated. When the control box can be manufactured as designed, it will be able to support the components that the future teams want to implement.

Switch Arm Assembly

After carrying out multiple design iterations, we concluded with a prime switch arm design that incorporates the important aspects of each prior prototype. As we were unable to fabricate our ideal switch arm design, we based our support for the prime switch arm design on Prototype #2 (wood-based design). After trial testing this prototype, it was found that the use of 2 wheels on the switch arm placed on a cantilever bracket fulfilled our design goal of ensuring that the wheels provide enough traction to grip onto the tracks to prevent the bogie from falling. Additionally, with the incorporation of slots at the areas where the wheels are placed, flexible adjustments can be made to ensure that the wheels are always in contact with the track.

To solidify the fabrication of the prime design of the switch arm, we conducted an FEA static simulation analysis. For the results, we obtained a maximum displacement of 0.0006245 inches, a maximum Von Mises stress value of 182.5 psi, and a minimum safety factor of 21.91. To elaborate on these results, we expected the area of high stress to be placed on the edges of the control arm bracket. This is because, in these areas, the “elongated” section of the control arm bracket is withstanding the weight of the E-shaped bracket. Additionally, we expected a low displacement similar to the switch arm design used by the previous Half-Scale team. However, we obtained an over-designed safety factor. We are confident that the design is an ideal solution but as a suggestion to future teams, along with further simulation analysis for this design, we propose to use a material other than 1060 aluminum alloy. We used this material as it is strong, moderately lightweight, and is reliable. It is recommended to use a different material that exhibits the same qualities as 1060 aluminum alloy but is lighter and does not output an over-defined safety factor value.

Half-Scale Track Design

With the proof of concept shown in SolidWorks of our prototype design of the half-scale track, the next plan of action would be fabrication. With the shelter in place orders, we were unable to carry out this process. However, if placed in an ideal situation where we are presented with the opportunity to replicate our results, instead of manufacturing a new track assembly, we would instead prototype the second alternative design solution for the Half Scale Track, which is welding a solid steel bar at the bottom of the track with the specified dimensions, as previously mentioned (Figure 14**)**. The steel bar solution would be cost-effective and useful during the first phase in prototype testing of the track.

Conclusion and Future Work

The successful trial of the prototype switching arm indicates that the 2019-2020 half-scale team accomplished its original and primary design goal. The wooden prototype performed better than expected, seamlessly traversing the curved Y-section moving forwards and backward multiple times. Furthermore, the team has created detailed CAD models for the switching arm and has performed FEA analysis upon it. Along with that, the team was able to provide a trial code that perfectly positioned the switching arm better than the previous code had, and left a prototype code based around a homing function to bring the half-scale model closer to practical implementation. A redesign for the control box has been completed and has undergone an FEA analysis.

Throughout this project, the team learned how easily plans can be sent off the rails by events occurring both within and outside of the project. The short that burned a stepper driver in early December 2019 forced the team to go into the break without taking time to verify what damage had befallen the control box. The subsequent failure to fix this damage set the team back weeks and prompted a complete rewiring of the control box after the damage was repaired to ensure that such an event could not happen again. Furthermore, the project was severely impeded by the COVID-19 pandemic, which forced a shelter-in-place order beginning in the middle of March 2020. This order continued past the end of the academic year and shut down hopes of manufacturing the switch arm. It also ruined the team’s meeting schedule and forced the team to meet in groups no larger than 2 people. This was a massive setback and prevented the team from testing the switch arm until early May 2020. During this time the team could only depend on CAD models and FEA simulations to make progress. Along with this lesson came one about the value of perseverance, as despite these impediments the team accomplished its primary design goals and laid the groundwork for the completion of every secondary goal.

The team has left a foundation for future years to build on. The first and foremost goal of next year’s team should be the manufacturing of the prototype switch arm as outlined by the CAD models provided by this year’s team. Secondary goals include the redesign of the drive shaft, testing and completion of the homing test code, the completion of the wireless controller, and the manufacturing of the redesigned control box. Also, the system is designed to run with 2 bogies supporting the control box. Therefore a future team will need to make mechanical updates to the second model bogie and replace the burnt second stepper driver. The code is written to only run one bogie, which means that a future team will have to program it to run a second one simultaneously. When doing this it will be important to keep in mind that the bogies are designed to run facing in opposite directions.

Further into the future, outside of the scope of work for next year's team would be the completion of autonomous mode. This requires multiple sensors to be operational. The limit switches already function, but the ultrasonic sensors have not been tested and the Hall effect sensors require an electrical current running along the track to function. A future team would be advised to enlist the help of a Wayside Power team to install charged rails along the track once all the previous changes are complete. Doing so would both provide the bogie with a reliable power source and would give the Hall effect sensors a current to track. The completion of all these steps would prepare the half-scale model bogie for practical implementation.

References

Aljoufie, M. (2014). Spatial analysis of the potential demand for public transport in the city of Jeddah, Saudi Arabia. In C.A. Brebbia's *Urban Transport XX* [e-book version]. Retrieved from https://books.google.com/books?hl=en&lr=&id=LGKbBAAAQBAJ&oi=fnd&pg=PA113&dq=public+transit+in+saudi+arabia&ots=-uRGT1wNWh&sig=CKXh94I5JU7j3F1BEw8O9nD2d-M#v=onepage&q=public%20transit%20in%20saudi%20arabia&f=false

Arduino Board Leonardo. (n.d.). Retrieved December 14, 2019, from https://www.arduino.cc/en/Main/Arduino_BoardLeonardo.

Bosselman, A. (2018, June 18). Bay Area subway and rail costs: Why are they among the highest in the world? Retrieved December 14, 2019, from https://sf.curbed.com/2018/6/18/17464616/bay-area-subway-train-rail-costs-price-bart-muni.

Competitive Advantages. (n.d.). Retrieved December 14, 2019, from https://www.jpods.com/aboutjpods/advantages.html

Futran (2018) Retrieved from https://futrangroup.com/

Guerra, E. (2016). Planning for cars that drive themselves: Metropolitan planning organizations, regional transportation plans, and autonomous vehicles. *Journal of Planning Education and Research*, *36*(2), 210-224.

Kenworthy, J.R. (2003). Transport energy use and greenhouse gases in urban passenger transport systems: A study of 84 global cities. In: International Sustainability Conference, 17 - 19 September, Fremantle, Western Australia

KLIA Aerotrain. (n.d.). Retrieved from http://klia.info/arrivals/aerotrain.htm.

Levy, A. (2018, December 19). Why Does Urban Rail Cost So Much in the U.S.? Retrieved from https://www.citylab.com/transportation/2018/01/why-its-so-expensive-to-build-urban-rail-in-the-us/551408/.

Lindley, J., (1987). Urban Freeway Congestion: Quantification of the Problem and Effectiveness of Potential Solutions. *ITE Journal*, *1*(1), 27-32

Pathak, R., Wyczalkowski, C., Huang, X (2014). Public Transit Access and the Changing Spatial Distribution of Poverty, *Regional Science and Urban Economics, 66*(1), 198-212 Retrieved May 15, 2020 from https://www.sciencedirect.com/science/article/pii/S0166046217302405

Personal Rapid Transit (PRT), Personal Automated Transport (PAT), PodCar, IPM and ATN Quicklinks. (n.d.). Retrieved December 14, 2019, from http://faculty.washington.edu/jbs/itrans/prtquick.htm.
SkyTran. (n.d.). Retrieved December 14, 2019 from https://www.skytran.com/

Poudenx, P., (2008). The effect of transportation policies on energy consumption and greenhouse gas emission from urban passenger transportation, *Transportation Research Part A: Policy and Practice, 42*(6), 901-909

Reilly, J.J. (2000). The management process for complex underground and tunneling projects. *Tunneling and Underground Space Technology, 15*(1), 31-44

Appendix A

Product Name	Description	Quantity	Cost Per Unit ($)	Links
PCB breadboard	Electrical	1	7.49	N/A
Hall effect sensor	Hall effect sensor - US5881LUA	1	0.96	https://www.digikey.com/product-detail/en/melexis-technologies-nv/US5881LUA-AAA-000-BU/US5881LUA-AAA-000-BU-ND/431876
Communication modules	NRF24L01 wireless module	2	2.43	https://www.mouser.com/ProductDetail/SparkFun/WRL-13678?qs=WyAARYrbSnZdmwzIRTs1Tw%3D%3D&gclid=Cj0KCQjwhtT1BRCiARIsAGlY51I3SIHEvlqIQDeQR9qF1LFUwHhXrtAXqSNBneMFn_Y-PARe9dYK6JgaAkVbEALw_wcB
Joystick Module	JoyStick Module Shield 2.54 mm 5 pin Biaxial Buttons Rocker	1	1.99	https://www.bananarobotics.com/shop/Joystick-Module?gclid=Cj0KCQjwhtT1BRCiARIsAGlY51KoS1mC2KmAkKwhah8V3j9vAOMZ9pWtcsjtJSm_jMq6XjS_m9KndZkaAkReEALw_wcB
Limit Switches	URBEST 10Pcs SPDT 3 Terminals Snap Action Micro Momentary Limit Switch	1	6.99	https://www.toyboxtech.com/urbest-micro-switch-momentary-spdt-3-terminals-snap-action-limit-switch-10pcs/
PVC Pipes	10 ft. PVC pipes	5	18.4	https://www.homedepot.com/p/1-in-x-10-ft-PVC-Schedule-40-Plain-End-Pipe-531194/202280936
PVC junctions	T shaped PVC pipe junctions	20	19.8	https://www.homedepot.com/p/Charlotte-Pipe-1-in-PVC-Sch-40-S-x-S-x-S-Tee-PVC024001000HD/203812199

	Crossed PVC pipe junction	1	2.98	https://www.homedepot.com/p/Dura-Corp-1-in-Sch-40-PVC-Cross-C420-010/100348109
	Corner PVC pipe junctions	8	21.36	https://www.homedepot.com/p/1-in-x-10-ft-PVC-Schedule-40-Plain-End-Pipe-531194/202280936
soldering kit	Electrical	1	10	https://www.northerntool.com/shop/tools/product_200639083_200639083
multimeter	electrical	1	FREE	N/A
tape measure	hardware	1	FREE	N/A
breadboards	electrical	2	FREE	N/A
Electrical wire	50 feet electrical wire	1	5	https://www.northerntool.com/shop/tools/product_200639083_200639083
3D print filament	Sent 3D printing for switch arm prototype from outside vendor, PLA filament, ~ 2 kg worth	1	20	Outside Vendor
Wire connectors	EL-SKY 800 pcs Assortment Ferrule Wire Copper Crimp Connectors	1	14	https://www.ebay.com/i/193381165366?chn=ps
Wood	1 meter long	1	FREE	
Dremmel	7300-N/5 Rotary Tool	1	32	https://www.walmart.com/ip/Dremel-7300-N-5-4-8V-Mini-Mite-Cordless-Rotary-Tool/15173820?wmlspartner=wlpa&selectedSellerId=0
	TOTAL		163.4	

Table 1: Bill of Materials for the 2019/2020 Half-Scale Team

ME195-B HALF-SCALE TEAM GANTT CHART

Project Lead: Greg White
Members: Brandon Scully, Keanu Heggem, Pinqian Lin

Task	Start Date	Days to Complete	Assigned Role
Inventory Check	29/01/2020	10	All
Bill of Materials	29/01/2020	5	Brandon
Test/Replace Steering Sprocket	02/02/2020	10	Greg
Weld/smoothe sides of track	09/11/2019	7	Greg and Brandon
Wiring schematics	22/01/2020	10	Edward
Wirieless controller	12/02/2020	20	Keanu
Finish limit switch code	26/02/2020	20	Keanu
Code for smooth transition without stop	26/02/2020	20	Keanu and Edward
Wood prototype design (switch arm)	05/02/2020	7	Greg and Brandon
Test Wood Prototype	04/03/2020	7	Greg and Brandon
Design Control Box (CAD)	19/02/2020	21	Edward
Switch Arm Final Design (CAD)	25/01/2020	55	Brandon
FEA analysis (Control Box and Arm)	12/03/2020	10	Brandon and Greg
Simulation of working model: CAD	16/04/2020	19	Brandon
Assemble and Fix Control Box	21/03/2020	20	Edward and Keanu
Trial Testing and Re-Modification Period	13/04/2020	25	All
Finalize Bill of Materials	22/04/2020	14	Brandon
Final Presentation	08/05/2020	1	All
Shop Clean Up	10/05/2020	1	All

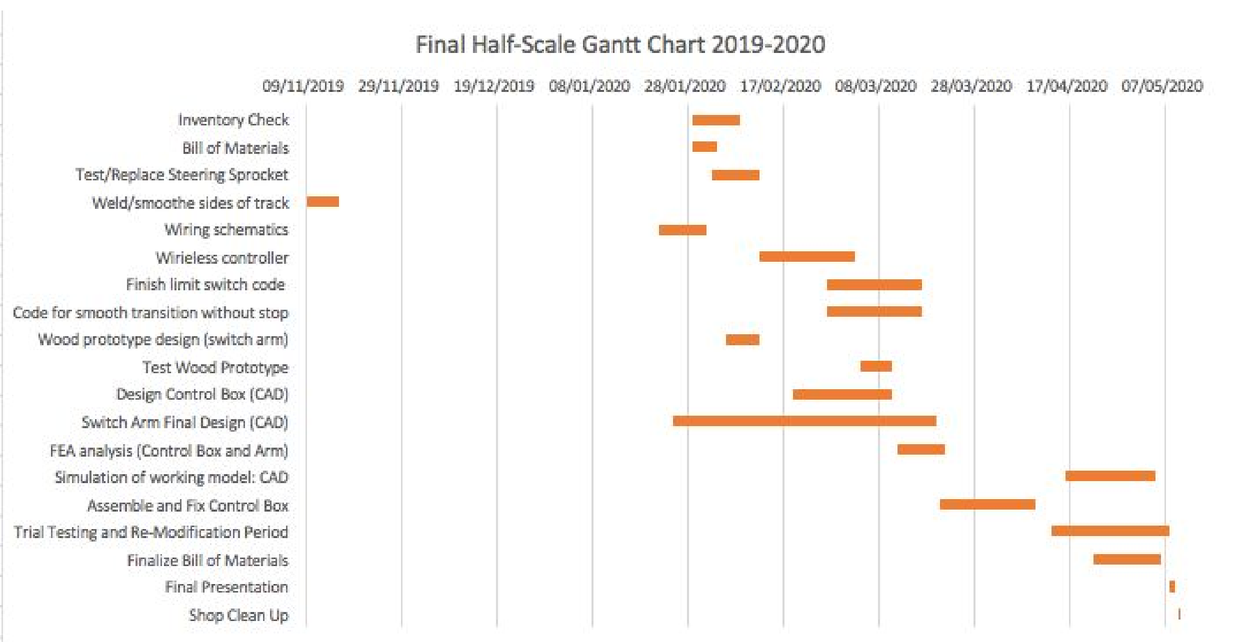


Figure 35: Modified Gantt Chart and Team Tasks for the Spring 2020 Half Scale Team

Appendix B

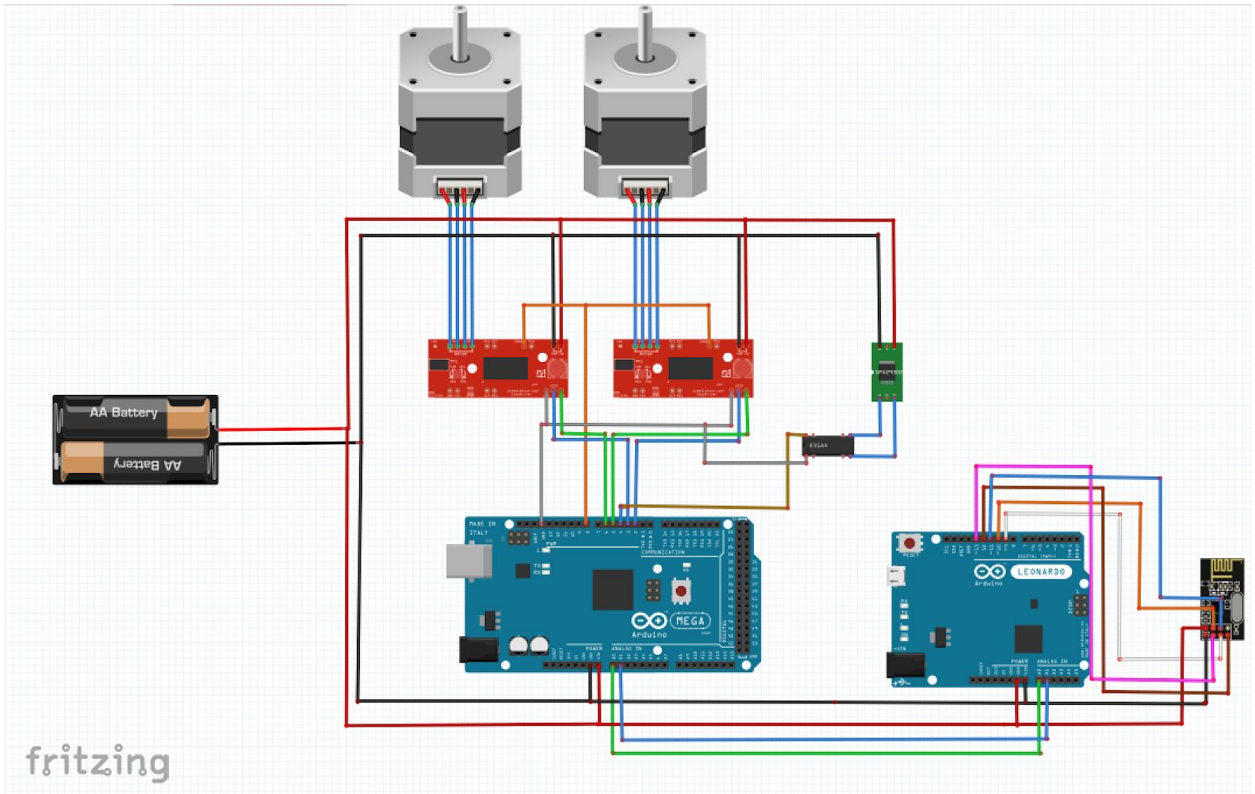


Figure 29: Schematic of Master Arduino with Power Supply, Stepper Motors, Server Arduino and NRF24L01 Wireless Communications Module

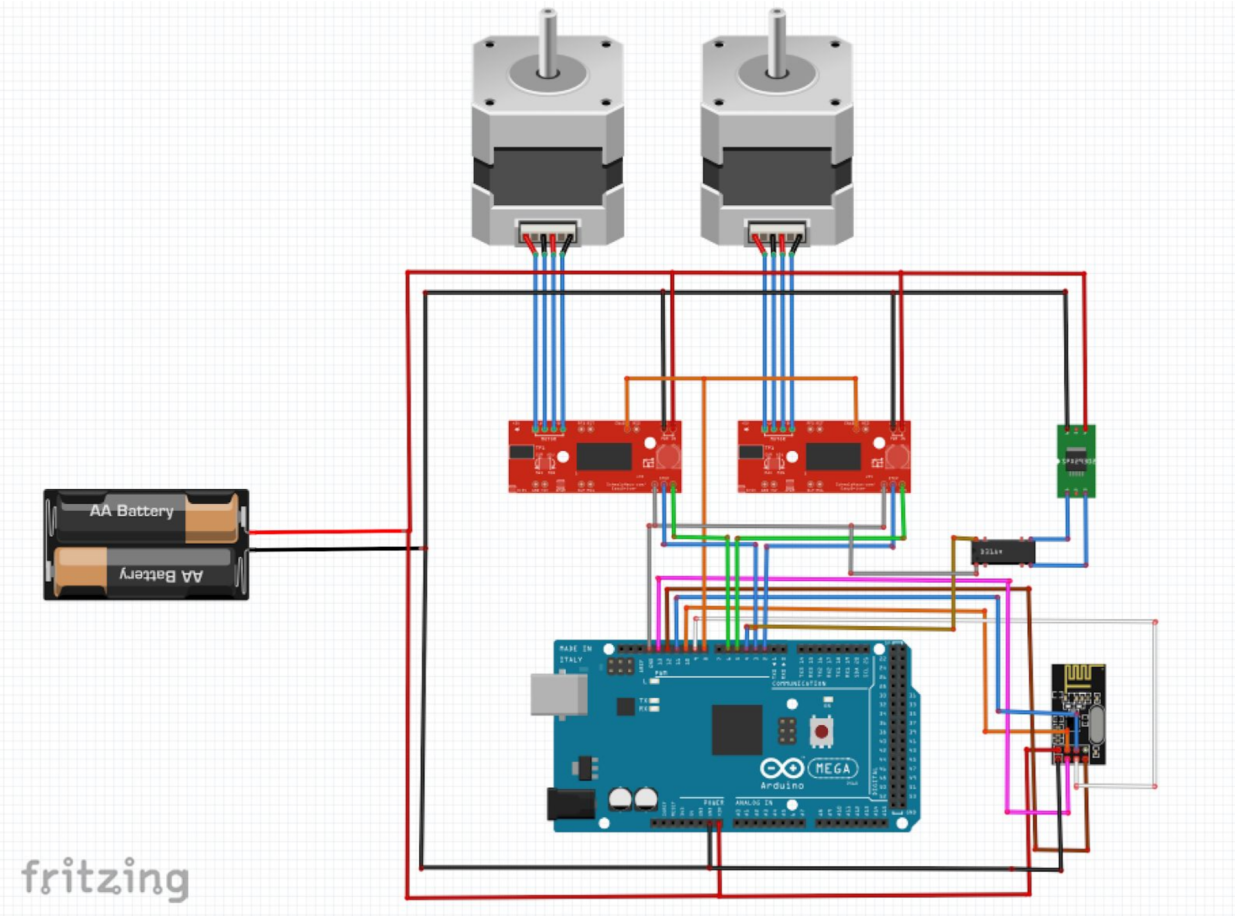


Figure 30: Master Arduino with Power Source, Stepper Motors, and NRF24L01 Wireless Communications Module

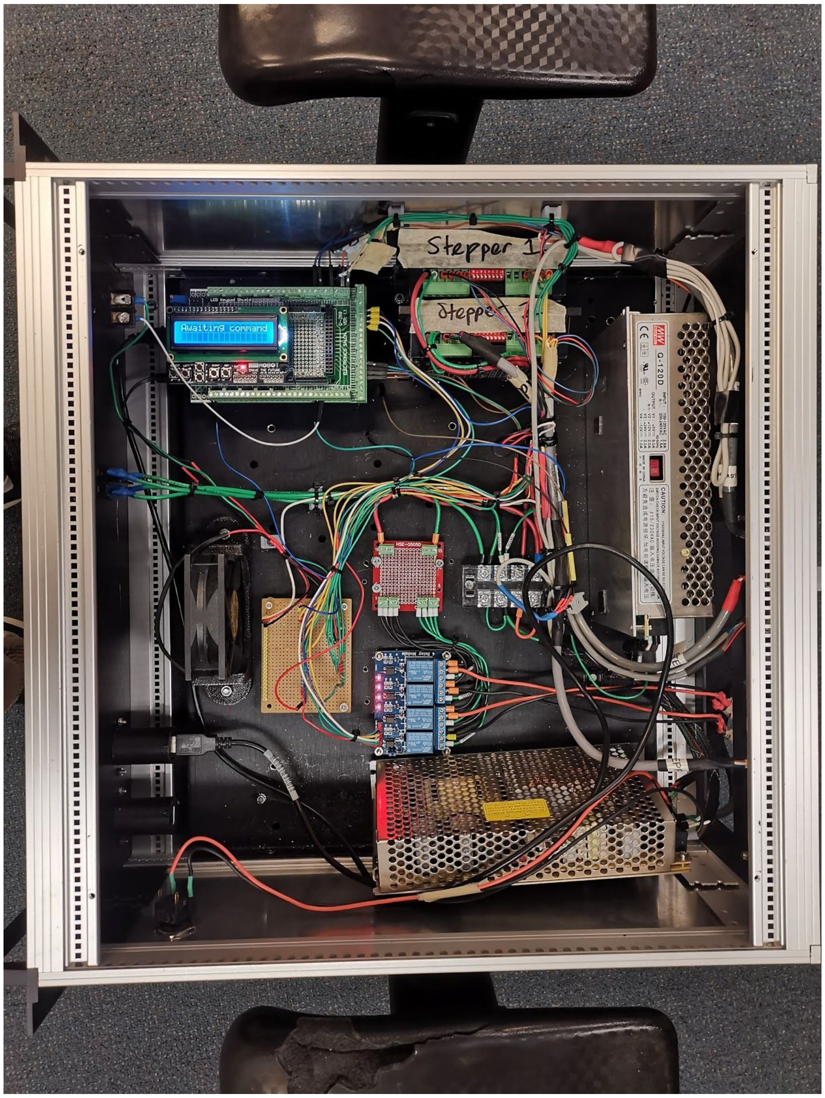


Figure 31: Control box designed from the previous year

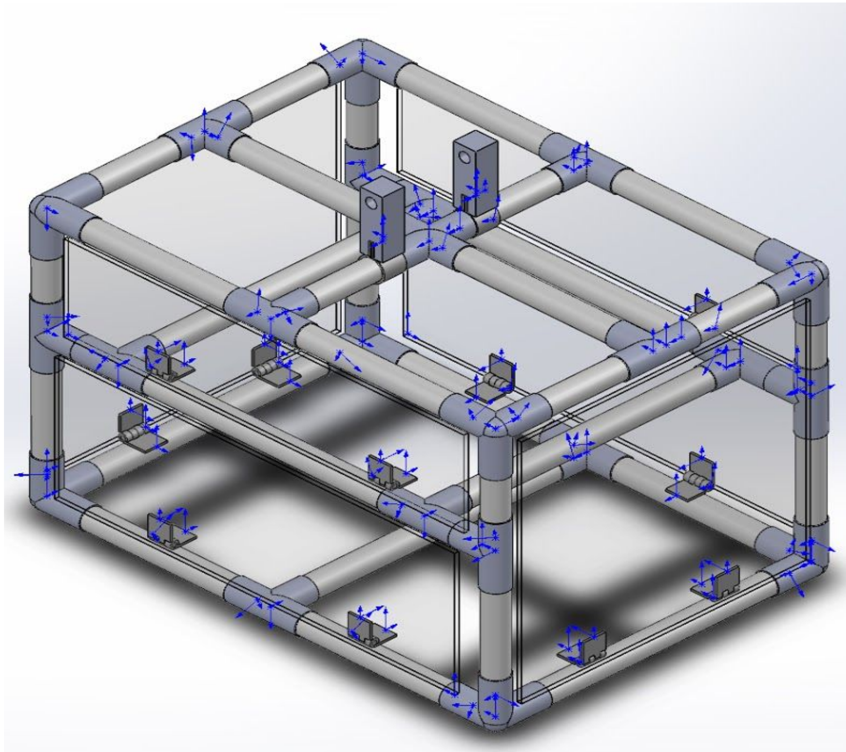


Figure 32: The PVC prototype of the control box

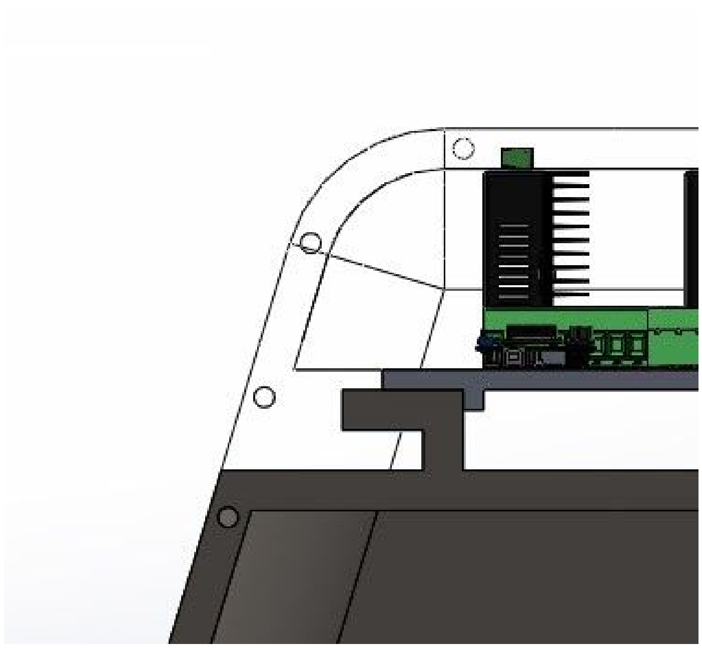


Figure 33: The guild way design of the top cover and the legs design of the component board

Figure 34: The stretchable shelf design for the control box from the previous team

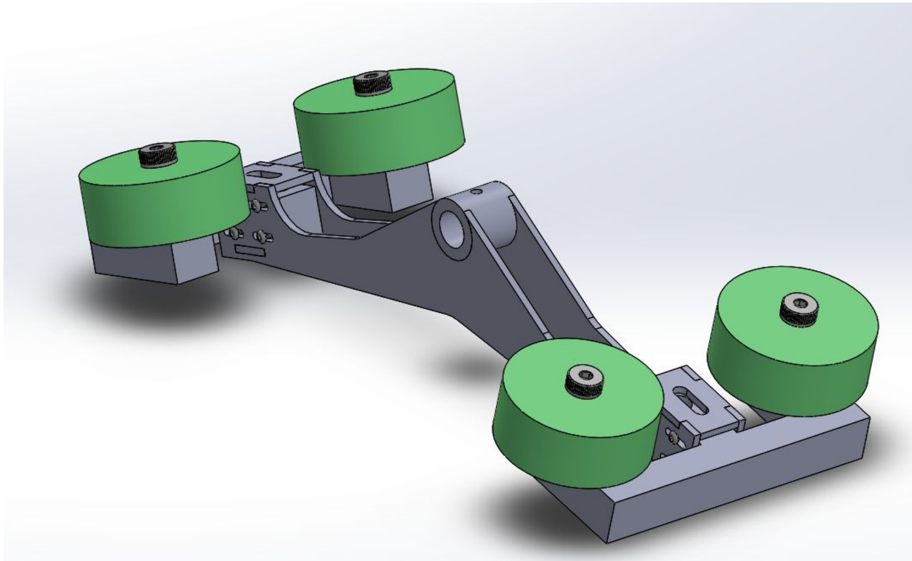


Figure 36: Isometric close-up view of the Prime Switch Arm Design

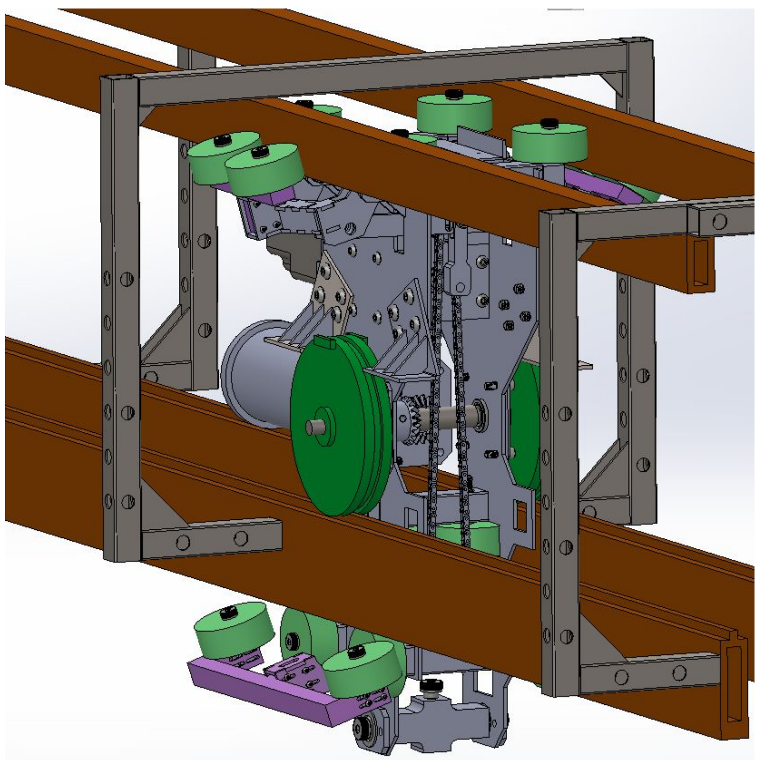


Figure 37: Visual representation of our modified bogie on the ideal Half-Scale track design

Part Letter	Part Name
A	BW002
B	LCA003
C	LCA300
D	LCAI001
E	97345A226 316 Stainless Steel Shoulder Screw
F	91255A107 Alloy Steel Button-Head Socket Screw
G	ControlArm Version 2 frame

Table 2: Part designations for the exploded view of the Switch Arm design as represented in the SolidWorks files of previous teams

San José State UNIVERSITY iNiST SPARTAN SUPERWAY

Appendix C

Code for Homing Stepper and Single Fluid Rotation:

```
#include <Wire.h>

#include <LiquidCrystal.h>

#include <SoftwareSerial.h>

#include <Stepper.h>


// select the pins used on the LCD panel

LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

// define some values used by the panel, buttons, and stepper

int lcd_key     = 0;

int adc_key_in  = 0;

int sensorvalue = 0;


#define btnRIGHT  0

#define btnUP     1

#define btnDOWN   2

#define btnLEFT   3

#define btnSELECT 4

#define btnNONE   5
```

```
//Limit Switches for oversteer protection
//Bogie 1
const byte limitL1 = 20;      //attachInterrupt pins can only be 2, 3, 18, 19, 20, 21
const byte limitR1 = 21;
//Bogie 2
const byte limitL2 = 19;
const byte limitR2 = 18;


#define pulse1 26             //Color brown on breadboard - stepper 1 on bogie 1
#define stprDIR1 23                   //Color orange on breadboard - stepper 1 on bogie 1
#define pulse2 24             //Stepper 2 on bogie 2
#define stprDIR2 25                   //Stepper 2 on bogie 2
#define autoSwitch 53         //Autonomous mode switch
#define in1 30        //Bogie motor 1 YELLOW
#define in2 31        //ORANGE
#define in3 32        //GREEN
#define in4 33        //Bogie motor 2 BLUE
#define dirB1 34              //PURPLE
#define dirB2 35              //GREY
int test = 300;       //The two values for the stepping size
int test2 = -300;
```

```
int fast = 20;            //The speed the stepper will move at

int onerev = 7200;            //The steps for one revolution

int STOP = 0;

int StepperPosition = 0;      //Variable signifying position of stepper

int StepperMax = 1200;      //Number of steps required to move to right side (note

variable "onerev" a few lines above)

int StepperReturn = -1200;          //Number of steps required to move to left side (equal

to the negative value of StepperMax)

Stepper stpr1(onerev, pulse1, stprDIR1);          //Define number of staps for a full rev

Stepper stpr2(onerev, pulse2, stprDIR2);

//Other

int read_LCD_buttons()

{

  adc_key_in = analogRead(A0);   // read the value from the sensor

  if (adc_key_in > 1000) return btnNONE;

  if (adc_key_in < 850 && adc_key_in > 600)  return btnSELECT;

  if (adc_key_in < 600 && adc_key_in > 408)  return btnLEFT;

  if (adc_key_in >= 0 && adc_key_in < 50)         return btnRIGHT;

  if (adc_key_in < 320 && adc_key_in > 253)  return btnDOWN;

  if (adc_key_in < 150 && adc_key_in > 100)  return btnUP;

  return btnNONE;  // when all others fail, return this...
```

```
}

//------------------------------------------------------------------------------------------------------------
void setup()
{
  //LCD
  lcd.begin(16, 2);
  lcd.setCursor(0, 0);
  lcd.print("Spartan Superway");
  lcd.setCursor(3, 1);
  lcd.print("2019");
  delay(1000);
  lcd.clear();
  delay(100);
  //Limit Switches for bogie 1
  pinMode(limitL1, INPUT_PULLUP);
  pinMode(limitR1, INPUT_PULLUP);
  //Limit Switches for bogie 2
  pinMode(limitL2, INPUT_PULLUP);
  pinMode(limitR2, INPUT_PULLUP);
```

```
//Steering
pinMode(stprDIR1, OUTPUT);  // CW- pin - direction
pinMode(pulse1, OUTPUT);  // CP- pin - pulse1/step
pinMode(stprDIR2, OUTPUT);  // CW- pin - direction
pinMode(pulse2, OUTPUT);  // CP- pin - pulse1/step
//Steering sensors
//Serial
Serial.begin(115200); // Open serial monitor at 115200 baud to see ping results.
//Powertrain
pinMode(in1, OUTPUT);   //motor 1 - HIGH = FORWARD  |   LOW = REVERSE
pinMode(in2, OUTPUT);   //motor 1 - LOW = FORWARD   |   HIGH = REVERSE
pinMode(in3, OUTPUT);   //motor 2 - HIGH = FORWARD  |   LOW = REVERSE
pinMode(in4, OUTPUT);   //motor 2 - LOW = FORWARD   |   HIGH = REVERSE
//Autonomous Mode
pinMode(autoSwitch, INPUT_PULLUP);
while (limitL1==HIGH){     // Home stepper and initialize position variable
        stpr1.setSpeed(fast);
        stpr1.step(test);
        delay(5);
}
StepperPosition = 0;
```

```
}
//------------------------------------------------------------------------------------------------------------
----------------------------------------------------------
void loop() {
  // put your main code here, to run repeatedly:
  lcd.setCursor(0, 0);
  //delay(1);// read the value from the sensor
  lcd_key = read_LCD_buttons();  // read the buttons
  //Serial.println("Initialized");
 switch (lcd_key)      //Execute a function based on user input
 {
  case btnUP:
  {
          lcd.clear();
          Serial.println ("Going Forward");
          lcd.print ("Going Forward");
          forward();
          delay(10);
          break;
  }
  case btnDOWN:
```

```
{
        lcd.clear();
        Serial.println ("Going Backwards");
        lcd.print ("Going Backwards");
        backwards();
        delay(10);
        break;
}
case btnRIGHT:
{
        lcd.clear();
        Serial.println ("Turning Right");
        lcd.print ("Turning Right");
        right();
        delay(10);
        break;
}
case btnLEFT:
{
        lcd.clear();
        Serial.println ("Turning Left");
```

```
        lcd.print ("Turning Left");
        left();
        delay(10);
        break;
  }
  case btnNONE:
  {
        lcd.clear();
        Serial.println ("Bogie Stopped");
        stopBogie();
        delay(10);
  }
 }

}
//Directional Functions
void forward() {
  delay(5);
  digitalWrite(in1, HIGH );
  digitalWrite(in2, LOW);
  digitalWrite(in3, HIGH);
```

```
  digitalWrite(in4, LOW);
  Serial.println("\t \t \t Moving Forward");
}
void backwards() {
  delay(5);
  digitalWrite(in1, LOW );
  digitalWrite(in2, HIGH);
  digitalWrite(in3, LOW);
  digitalWrite(in4, HIGH);
  Serial.println("\t \t \t Moving Backwards");
}
void right() {
  if (StepperPosition = 0){   // If switching arm is on left end of arc
        stpr1.setSpeed(fast);         // Sets stepper speed
        stpr1.step(StepperMax);     // Moves switching arm by number of steps required
to reach right side
        StepperPosition = StepperMax;   // Changes stepper position to right side
   } else {
        Serial.println("Right Limit Reached");
  }
}
```

```
void left() {
  if (StepperPosition = StepperMax){  // If switching arm is on right end of arc
        stpr1.setSpeed(fast);         // Sets stepper speed
        stpr1.setSpeed(StepperReturn);   // Moves switching arm by number of steps
required to reach left side
        StepperPosition = 0;          // Changes stepper position to right side
  } else {
        Serial.println("Left Limit Reached");
  }
}
void stopBogie() {
  digitalWrite(in1, LOW);
  digitalWrite(in2, LOW);
  digitalWrite(in3, LOW);
  digitalWrite(in4, LOW);
  delay(5);
  Serial.print("\t \t \t Bogie Stopped");
}
```

Trial Code:

```
/*******************************************************

  Half Scale Bogie

  2020

  Test Code for Revised Switching Arm

  Written by Keanu Heggem

********************************************************/

#include <Wire.h>

#include <LiquidCrystal.h>

#include <SoftwareSerial.h>

#include <Stepper.h>

#include <NewPing.h>


// select the pins used on the LCD panel

LiquidCrystal lcd(8, 9, 4, 5, 6, 7);

// define some values used by the panel, buttons, and stepper

int lcd_key     = 0;

int adc_key_in  = 0;

int sensorvalue = 0;

#define TRIGGER_PIN1  40  // Arduino pin tied to trigger pin on the ultrasonic sensor.

#define ECHO_PIN1        42  // Arduino pin tied to echo pin on the ultrasonic sensor.
```

```
#define TRIGGER_PIN2  41  // Arduino pin tied to trigger pin on the ultrasonic sensor.

#define ECHO_PIN2         43  // Arduino pin tied to echo pin on the ultrasonic sensor.

#define MAX_DISTANCE 200 // Maximum distance we want to ping for (in centimeters).

Maximum sensor distance is rated at 400-500cm.

#define btnRIGHT  0

#define btnUP        1

#define btnDOWN   2

#define btnLEFT   3

#define btnSELECT 4

#define btnNONE   5

//Limit Switches for oversteer protection

//Bogie 1

const byte limitL1 = 20;      //attachInterrupt pins can only be 2, 3, 18, 19, 20, 21

const byte limitR1 = 21;

//Bogie 2

const byte limitL2 = 19;

const byte limitR2 = 18;


#define pulse1 26             //Color brown on breadboard - stepper 1 on bogie 1

#define stprDIR1 23                   //Color orange on breadboard - stepper 1 on bogie 1

#define pulse2 24             //Stepper 2 on bogie 2
```

```
#define stprDIR2 25                   //Stepper 2 on bogie 2
#define autoSwitch 53       //Autonomous mode switch
#define in1 30          //Bogie motor 1 YELLOW
#define in2 31          //ORANGE
#define in3 32          //GREEN
#define in4 33          //Bogie motor 2 BLUE
#define dirB1 34                //PURPLE
#define dirB2 35                //GREY
#define hallF 44
#define hallR 45
int test = 300;         //The two values for the stepping size
int test2 = -300;
int fast = 20;          //The speed the stepper will move at
int onerev = 7200;              //The steps for one revolution
int STOP = 0;

int hallturn = 6500;            // The only one being used right now is hallturn when going
forward.
int hallflip1 = 6500;           // It is overwritten to be negative after being triggered going
forward
int hallflip2 = 6500;
```

```
Stepper stpr1(onerev, pulse1, stprDIR1);          //Define number of staps for a full rev
Stepper stpr2(onerev, pulse2, stprDIR2);
//Other
int read_LCD_buttons()
{
  adc_key_in = analogRead(A0);   // read the value from the sensor
  if (adc_key_in > 1000) return btnNONE;
  if (adc_key_in < 850 && adc_key_in > 600)  return btnSELECT;
  if (adc_key_in < 600 && adc_key_in > 408)  return btnLEFT;
  if (adc_key_in >= 0 && adc_key_in < 50)        return btnRIGHT;
  if (adc_key_in < 320 && adc_key_in > 253)  return btnDOWN;
  if (adc_key_in < 150 && adc_key_in > 100)  return btnUP;
  return btnNONE;  // when all others fail, return this...
}


//-----------------------------------------------------------------------------------------------------------
void setup()
{
  //LCD
  lcd.begin(16, 2);
  lcd.setCursor(0, 0);
```

```
lcd.print("Spartan Superway");
lcd.setCursor(3, 1);
lcd.print("2019");
delay(1000);
lcd.clear();
delay(100);
//Limit Switches for bogie 1
//attachInterrupt(digitalPinToInterrupt(limitL1), isr, FALLING);
//attachInterrupt(digitalPinToInterrupt(limitR1), isr, FALLING);
pinMode(limitL1, INPUT_PULLUP);
pinMode(limitR1, INPUT_PULLUP);
//Limit Switches for bogie 2
//attachInterrupt(digitalPinToInterrupt(limitL2), isr, FALLING);
//attachInterrupt(digitalPinToInterrupt(limitR2), isr, FALLING);
pinMode(limitL2, INPUT_PULLUP);
pinMode(limitR2, INPUT_PULLUP);

//Steering
pinMode(stprDIR1, OUTPUT);  // CW- pin - direction
pinMode(pulse1, OUTPUT);  // CP- pin - pulse1/step
pinMode(stprDIR2, OUTPUT);  // CW- pin - direction
```

```
  pinMode(pulse2, OUTPUT);  // CP- pin - pulse2/step
  //Steering sensors
  pinMode(hallF, INPUT);
  pinMode(hallR, INPUT);


  //Serial
  Serial.begin(115200); // Open serial monitor at 115200 baud to see ping results.
  //Powertrain
  pinMode(in1, OUTPUT);   //motor 1 - HIGH = FORWARD  |   LOW = REVERSE
  pinMode(in2, OUTPUT);   //motor 1 - LOW = FORWARD   |   HIGH = REVERSE
  pinMode(in3, OUTPUT);   //motor 2 - HIGH = FORWARD  |   LOW = REVERSE
  pinMode(in4, OUTPUT);   //motor 2 - LOW = FORWARD   |   HIGH = REVERSE
  //Autonomous Mode
  pinMode(autoSwitch, INPUT_PULLUP);
}


//------------------------------------------------------------------------------------------------------- -
void loop() {
  //Serial.println(adc_key_in);
  lcd.setCursor(0, 0);
  //delay(1);// read the value from the sensor
```

```
  lcd_key = read_LCD_buttons();  // read the buttons
  //Serial.println("Initialized");
  switch (lcd_key)               // depending on which button was pushed, we perform an
action
  {
          case btnLEFT:
          {
          Serial.println(adc_key_in);
          lcd.clear();
          leftlimit();
          delay(1);
          break;
          }
          case btnRIGHT:
          {
          Serial.println(adc_key_in);
          lcd.clear();
          lcd.print("Turning Right");
          rightlimit();
          delay(100);
          break;
```

```
}
case btnUP:
{
Serial.println(adc_key_in);
Serial.println(combinevar);
lcd.clear();
lcd.print("Going Forward");
forward();
break;
}
case btnDOWN:
{
Serial.println(adc_key_in);
Serial.println(combinevar);
lcd.clear();
lcd.print("Going Backwards");
backwards();
break;
}
case btnNONE:
{
```

```
        stopBogie();
        while (int x = digitalRead(autoSwitch)  == LOW) {
        lcd.clear();
        lcd.print("Autonomous Mode");
        Serial.println("Autonomous Mode");
        //autonomousMode();
        break;
        }
        //Serial.println("Awaiting Command");
        lcd.print("Awaiting command");
        }
  }


void leftlimit() {
  while (digitalRead(limitL1) == LOW) {
        //delay(1);
        //stpr1.setSpeed(1);
        //stpr1.step(-1);
        Serial.println("Steering left limit triggered, stop turning");
        break;
```

```
  }
  while (digitalRead(limitL1)  == HIGH) {
          //delay(1);
          Serial.println ("Turning Left");
          lcd.print ("Turning Left");
          stpr1.step(test);
          // this is to alternate directions for autonomous mode test = ~test;
          break;
  }
}
void rightlimit() {
  while (digitalRead(limitR1) == LOW) {
          //delay(1);
          //stpr1.setSpeed(1);
          //stpr1.step(-1);
          Serial.print("Steering right limit triggered, stop turning");
          break;
  }
  while (digitalRead(limitR1) == HIGH) {
          //delay(1);
          Serial.println ("Turning Right");
```

```
        lcd.print ("Turning Right");
        stpr1.setSpeed(fast);
        stpr1.step(test2);
        break;
  }
}
void forward() {
  delay(5);
  digitalWrite(in1, HIGH );
  digitalWrite(in2, LOW);
  digitalWrite(in3, HIGH);
  digitalWrite(in4, LOW);
  Serial.println("\t \t \t Moving Forward");
}
void backwards() {
  delay(5);
  digitalWrite(in1, LOW );
  digitalWrite(in2, HIGH);
  digitalWrite(in3, LOW);
  digitalWrite(in4, HIGH);
  Serial.println("\t \t \t Moving Backwards");
```

```
}

void stopBogie() {

  digitalWrite(in1, LOW);

  digitalWrite(in2, LOW);

  digitalWrite(in3, LOW);

  digitalWrite(in4, LOW);

  delay(5);

  Serial.print("\t \t \t Bogie Stopped");

}
```