Hive: The Grounded Swarm

University of Central Florida
College of Electrical Engineering and Computer Science

Group 21

Dr. Sun
Dr. Borowczak

Senior Design 1 - Final Draft

Isaac Finley - Electrical Engineering
Cooper Fitzgerald - Electrical Engineering
John McClain - Mechanical Engineering
Cameron Nichols - Computer Engineering, Electrical Engineering, Computer Science
Benjamin Palladino - Electrical Engineering

Table of Contents

1.0 Executive Summary

Robotic automation is becoming an increasingly crucial investment for businesses spanning numerous industries. As technology develops, these robots have become progressively less reliant on human intervention. Most commonly, autonomous mobile robots, or AMRs, are mainly used to perform non-value added tasks that can greatly improve the efficiency of a production process. AMRs are able to move through their environment independently and make thorough decisions without the need for a supervisor; a massive benefit when compared to their successors, automated guided vehicles, who require a predefined path and fail to smoothly mesh with their human counterparts in the workplace. AMR capabilities generally include computer vision, artificial intelligence, path planning and autonomous navigation, motion control, safety features, sensing, and some degree of a human-machine interface. In order to achieve such capabilities, an AMR must be equipped with a means of locomotion, an array of comprehensive sensors, powerful computation hardware, and an instrument to perform their designated responsibility. Such robots most prominently take on the role of moving items from one place to another and often work concurrently amongst several other units.

This project seeks to design and construct two autonomous mobile robots that can perform tasks simultaneously. Each unit will be controlled by server-hosted software that maps the environment via camera data and wirelessly sends commands to each robot. The robots will navigate their surroundings and manipulate their workplace based on these commands. This means the primary workload of the system is performed remotely, reducing the computational hardware required on the unit. The robots will feature a port where various tools can be easily swapped out in order to make the unit capable of performing a multitude of tasks. To narrow the scope of this project, the specific attachment developed in this project is a motorized latching mechanism that tows a payload on a wheeled pallet from one point to another.

The team is interdisciplinary, composed of four computer science students, four electrical engineering students, and one mechanical engineering student. This report covers the electrical and mechanical engineering portion of the project. The computer science students are primarily responsible for developing software that uses computer vision to plan and route the movement of robots. Using the camera data, the software will locate the robots and payload, directing them from pickup location to destination. The electrical engineering students are tasked with developing the hardware; namely a mobile robot capable of receiving commands via radio frequency transmission, and towing a payload across some pre-defined space. In addition to the swappable attachment port for flexibility of interaction with its environment, the AMR will feature on-board sensors to add a layer of reliability, ensuring it can safely navigate its workplace. The mechanical engineering student is responsible for designing a structurally sound robot frame, as well as a robust drive train capable of safely pulling payloads; taking into consideration needs for torque, strafing ability, and mechanical compatibility of swappable attachments.

2.0 Project Description

The goal of this project is to develop two autonomous mobile robotic agents tuned for a warehouse setting. The two agents will be responsible for driving to a payload, latching onto the payload, and towing the payload to a given destination in a pre-defined space. The payload will consist of a milk crate on wheels containing items of various weight. The frame of each robot is composed of T-slot aluminum channels assembled in a cube for ease of component attachment. Each robot features four-wheel drive using mecanum wheels, which are capable of movement in all directions. The system is powered by an onboard battery. The battery will be complemented by supercapacitors, a sponsor requirement, which will aid the system in peak power delivery during the initiation of the towing operation. The robots will feature a modular attachment port, capable of operating various tools for adaptable interactment with the work environment. For the selected warehouse scenario, the attachment is a motorized latching mechanism that will hook onto a small metal bar spanning the front of each payload, and release once the payload has reached its destination.

A central camera located above the area will observe the operating environment, and provide data to an algorithm that dictates each robot's behavior. The data will be used to locate the position of each robot, as well as the payload and its destination. The software will create an optimized path for each unit, ensuring that both systems can simultaneously operate without interference. This computer vision software is developed by the computer science portion of the team. This team is also responsible for creating a router to communicate with and control the robots. The agents will wirelessly receive commands from this router, which is connected to the server hosting the software, through radio frequency transmission. The robots will decode the transmission data and use it to control the motors, traveling to the payload and then maneuvering the path to its destination as defined by the data. The overarching aim is to focus on adaptability; meaning the system can be tuned for various situations rather than companies needing to purchase an entirely new system. AMRs have and will continue to automate businesses, so it becomes increasingly necessary to develop adaptable robots that can integrate into processes without needing to redesign the whole facility.

2.1 Project Motivation

The concept of designing a robotic system arose from the computer science half of our team. Cameron, a member of the Robotics Club of Central Florida (RCCF), had found a team of computer science students looking to design and develop a robotic fleet deployment and management platform. Seeking to innovate, the team decided to offset the robots computation to a cloud server, rather than perform the operations on the physical hardware. In doing so, the team sought to make it cheaper to scale such robots as well as easier to implement more advanced autonomy. Given that robots traditionally have onboard computer systems dictating their every move, offsetting the computation to a server offers a unique trait of adaptability not seen on most robots. Of course, no deliverable can be brought to fruition without a physical robot; which is where the electrical and mechanical team came into play.

After investigating the need for robot systems we found that 88% of businesses worldwide plan to adopt robotic automation into their future infrastructure (McKinsey & Company). As competition amongst international business leaders continues to become increasingly fierce, the push to industry-wide robotic automation is inevitable. Robotic solutions are put in place in order to increase production rates, eliminate human-need for monotonous or dangerous tasks, and increase the accuracy and consistency of industrial processes. Such solutions are seen in numerous scenarios from nearly every industry. The overwhelming desire of businesses to introduce robotic solutions arises from the undeniable benefits and efficiency robots bring to industrial environments; however, this does not imply a lack of disadvantages that can hinder how such applications are performed.

In the modern robotics industry, most robotic solutions are single-application oriented. In other words, complex robotic systems are uniquely designed for one static application. By nature, this is highly inefficient for several reasons. Firstly, businesses will need to sacrifice a massive upfront investment for each individual solution they wish to apply. The problems faced by such a company may span numerous processes, requiring numerous unique solutions to be created in order to solve their problems. Aside from their cost, the individualized solutions each require a team of designers who must become familiar with the nuances of each application; taking up valuable time and exhausting efforts. Additionally, the static solution does not account for business scenarios to change in the future, as the robotic system is designed for a singular use case. This hinders the adaptability of the company, preventing its systems from adhering to market changes. As a whole, requiring numerous unique robotic applications increases the difficulty of maintaining production, as different systems will require distinct care and have particular needs.

In order to align our goals with the computer science team, we decided to focus on the adaptable nature of the robot. This project seeks to develop an autonomous mobile robot that expands upon the adaptability and modularity of traditional robotic systems in order to provide businesses with a solution that can better grow alongside their goals. The system aims to bridge the existing commonalities amongst typical robotic applications with flexibility in mind. This allows for use of existing elements that have proven their reliability whilst leaving room to modify secondary components of the system for a particular application. By featuring the same core elements that characterize traditional systems, there is a basis applicable to numerous scenarios but with added innovations for flexibility.

Due to the nature of the software team's goals, it was pre-determined that the unit would be an autonomous mobile robot as opposed to a robotic arm, or an automated guided vehicle; so our brainstorming was centered around the innovative features of our AMR. After looking through the most common applications, settings, and objectives of AMRs, we decided on a warehouse situation. In warehouses, AMRs are traditionally used for transporting, lifting, and packing items.

2.2 Objectives and Goals

The goal of this project is to design two autonomous mobile robots that will simultaneously tow a payload from one point to another in a confined area. Table 2.1 highlights the disciplines involved in this project, their individual goal, and objective used to achieve said goal. This table considers the goals in the scope of the entire project, however some disciplines are the responsibilities of the computer science team and will not be detailed in this report.

Table 2.1: Goals and Objectives

Discipline	Goal	Objective
Navigation & Control Software	Environment mapping, navigation, and control.	Develop a program to interpret camera data to create a path for robots, making adjustments from robot feedback.
Router	Robot command transmission and feedback data reception.	Program and equip a microcontroller with a transceiver to route commands to robots and intercept feedback.
Embedded Software	Interpretation of radio frequency and sensor data, drive control.	Program a processor to read sensor data, control locomotive and servo motors, and process radio frequency data.
Onboard Hardware	Data transmission and reception, locomotion, sensing, payload manipulation, and user interfacing.	Integrate a transceiver, sensors, motors, and user control with a processor.
Mechanical	Robot structure, drive mechanism, and payload hitching.	Develop a supporting frame, drivetrain system of sufficient torque, and a payload latching mechanism.

2.2.1 Navigation and Control Software (Software Team)

Overall, the software team is developing a fleet management and deployment platform that allows users to control a robotic fleet that uses a ROS-based software package. The users will be able to control whatever hardware system or simulation they would like to implement with commands available through the web-app. For this project, the team is focused on supporting moving specific payloads, navigating to specific points, or sending velocity commands to specific robots.

To do that in this specific use case, they will be working on creating a ROS package that can take a 2D overhead view of a bounded area, identify specific areas or objects in the image through the use of ArUco tags, and construct a multi-layered occupancy grid that labels the individual objects in the area. For example, the occupancy grid will have a layer consisting of just the robots, a layer consisting of the payloads, and another layer consisting of any other objects that don't fall into any of those categories, which will be treated as obstacles. Once this occupancy grid is created, we can use a pathfinding algorithm for navigation and obstacle avoidance. This occupancy grid will track orientation and location. This system will be implemented as a series of python scripts that perform the computer vision, and a series of costmap2D layer plugins written in C++.

2.2.2 Router (Software Team)

A microcontroller will be used to take data processed on the server, and send it to the robot via the WebSocket protocol. The system will also receive data from the robot and send it back to the server in order to make position adjustments based on feedback. The software team takes on this responsibility because the system will directly interface with their developed mapping algorithm.

2.2.3 Embedded System

The embedded software is the onboard program primarily used to read robot sensor data, drive locomotive and servo motors, process incoming data, and prepare data for transmission. The program must be able decipher transmission data into motor control commands, which are used for both mobility and latching mechanism operation. This system is developed by the electrical team, and is distinct from the navigation and control software which is used to map the robot’s environment and determine their locomotion.

2.2.4 Hardware

The hardware system revolves around the use of a processor which upholds a wireless communication protocol for interaction with the server, four driving motors for locomotion, a servo motor for the attachment mechanism, and a user interface for status indication and emergency shutdown capabilities. The robot also features sensors for collision avoidance and backup payload detection.

2.2.5 Mechanical

A structurally sound frame must be constructed that is capable of withstanding the maximum payload weight, supporting onboard hardware, and surviving collisions. A drivetrain must be developed that can support precise control via software, and provide enough torque to tow the maximum payload.

2.3 Requirement Specifications

The purpose of requirement specifications are to address the technical needs of a design. After numerous brainstorming sessions, the team decided upon the specifications of the robots that would best align with our goals. If these requirements are met, our product will uphold technology readiness level 6. This section distinguishes between functional requirements of the robot, listed below, and unit-measured specifications of system properties, listed in Table 2.2.

- The AMR will wirelessly receive commands from a router.
- The AMR will translate commands into motor locomotion.
  - The agent will navigate to the payload.
- The AMR will translate commands into latching attachment operations.
  - The agent will latch on to the payload.
- The agent will tow the payload to the given destination.
- The agent will release the payload.
- The agent will have collision avoidance sensors as a backup to the software control.
- The agent will have LED status indication lights to show when in operation and when in idle.
- The agent will have an emergency stop button.
- The agent will have four-wheel drive and be capable of strafing.
- The agent will have a swappable attachment port.
  - Attachment port is mechanically compatible with various tools.
  - Attachment port offers various electrical connections.

Table 2.2: Requirement Specifications

Specification	Description	Unit
Operation Lifetime	Runtime of the agent in operation.	30 mins
AMR volume	Maximum volume of the agent including all onboard components.	≤ 2 ft^3

Specification	Description	Unit
AMR weight	Maximum weight of each agent.	≤ 30 lbs
Range	Minimum transmission distance between onboard transceiver and router.	≥ 10 ft
Drivetrain Torque	Necessary torque to tow maximum prescribed payload.	6 kg-cm
Speed	Minimum speed of the agent.	≥ 0.25 ft/sec
Camera FOV	Field of vision for the overhead observation camera.	130°
Modular Connections	Minimum number of GPIO pins available on processor.	≥ 12
Processor Speed	Minimum speed of the agent's processor.	≥ 8 MHz
Processor Compatibility	Development platform of the processor.	Arduino IDE
Locomotion Area	Area of terrain the agent is capable of traversing.	10x10 ft
Power	Maximum peak power consumption of the agent.	< 40 W
Payload Weight	Maximum towing capability of the agent.	25 lbs
Cost	Maximum cost of a single agent.	≤ $200

2.4 Main Electrical Block Diagram

This section illustrates the main block diagram that will be used in this project. It includes the path of power flow and data along with color coding to section the work loads between the team.

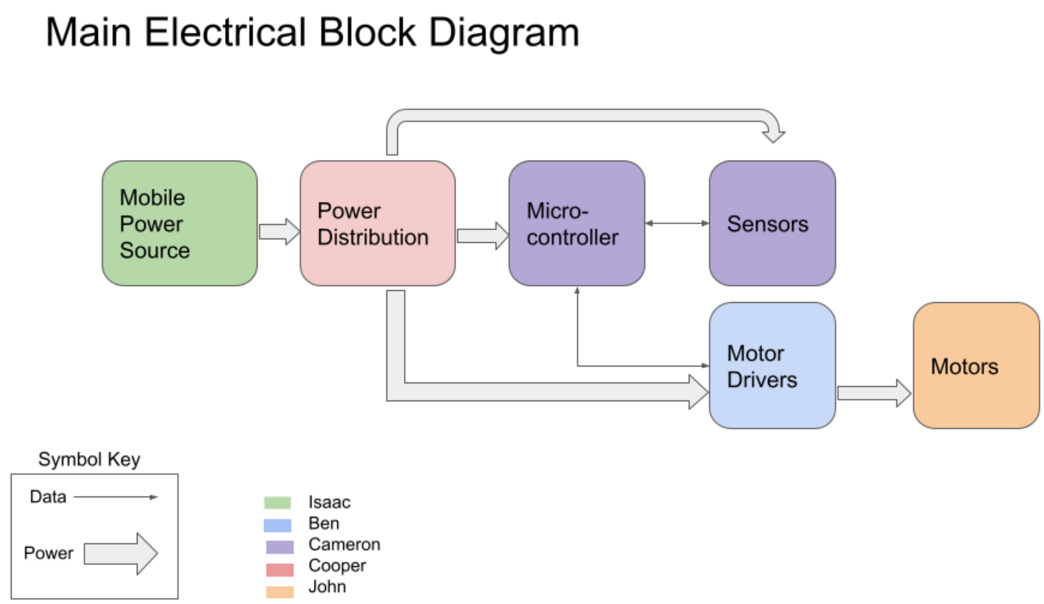

Figure 2.1: Main Electrical Block Diagram

2.5 House of Quality

The house of quality chart is a tool used in the quality management process of an engineering project. It is a matrix that helps to identify the relationship between customer requirements and the technical specifications of the product. This helps to ensure that the final product meets the expectations of both the and technical requirements.

The most important engineering requirements of the project are minimizing the cost and power consumption, while maximizing the hardware interfacing. Due to the budget constraints and limited energy storage of a small onboard battery, these aspects of the design must be complemented with the overarching goal of having an adaptable and modular autonomous mobile robot.

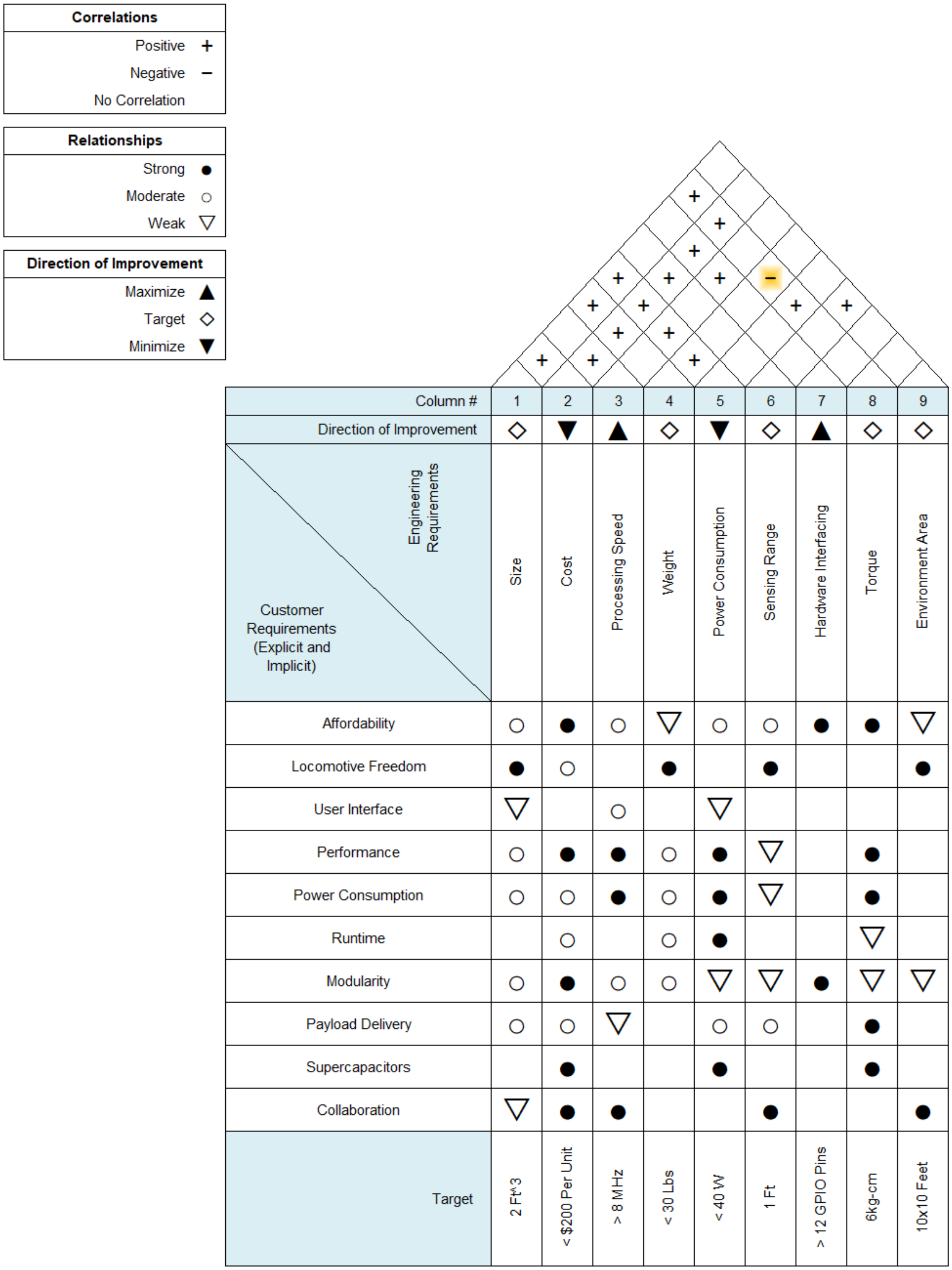

Figure 2.2: House of Quality Diagram

3.0 Technology Investigation

This section will discuss the technology surrounding the Hive team project, similar technologies, as well as technologies that will possibly be implemented for the agent. Each section focuses on a field of technology that is relevant to some sub-system of the design.

3.1 Existing Similar Projects and Products

This following text discusses similar projects that are in some way, shape, or form associated with the Hive team's plans and goals for the agent. While not necessarily focused on the same technological implementation this team is focused on, these projects involve their own spin on the modularity aspect of the robot, a main goal of our project.

3.1.1 Polybot

One such example of a project similar in ideology to the agent is known as "Polybot" which was done by the Xerox Palo Alto Research Center in California. The goal was to create a robot that was capable of adapting in the field, meaning that the robot would be able to recognize and respond to stimuli by changing its shape. (Yim et al.) These types of adaptable robots are broadly classified into three main categories; chain, lattice, or mobile reconfiguration.

Chain type robots are interconnected where each part is connected to one or more other parts, which are also connected to one or more other parts and so on, with the capability to connect to other points and disconnect as needed to be able to adapt itself into a necessary shape. Essentially, imagine a plane of points, all connected to the points surrounding it. Then take a point in that plane and connect it to another point that it is not already connected to. This would change the geometry of the robot and make it more suited for the adaptability purpose. Changing the geometry of the robot allows for many things, primarily of which is allowing the robot to better adapt to the environment around it as well as adapting its gripper or other equivalent actuator for manipulating the environment around it in accordance with the task it is performing. This is the methodology the Xerox Palo Alto Research Center in California primarily focused on as they found it to be the most suited for the purposes of the "Polybot." (Yim et al.)

Lattice type robots change shape by shifting into different areas while still connected. It is akin to moving a pawn on a chessboard, given that the chessboard is three dimensional. (Yim et al.) Essentially, each piece can only move to an adjacent space which helps simplify the process of changing shape as compared to the chain structure, which can be moved to any arbitrary point so long as it is still connected to at least one other piece.

The last type of configuration is the mobile reconfiguration type, which can have parts detach from the main body and move independently to attach to each other and create configurations that way. (Yim et al.) While this has amazing uses in principle, executing it practically is too difficult to achieve properly. None of these fit what the agent will be, as when adaptability is said for the agent it is meant that there will be a base that a user can then modify or attach modules to easily that will allow for the agent to be adaptable. This is not adaptability while in use, but adaptability as a result of the rigorous planning that engineering requires. This will allow for the agent to be capable of handling many different situations so long as they are planned for, which allows for engineers to truly shine and perform tasks more efficiently.

The primary difference between the agent and the “Polybot” is that the “Polybot” is capable of manipulating its geometry to perform a task or otherwise adapt itself to its surroundings to perform better while the agent will not be able to, but the agent will be able to have modules put on or installed prior to a task, or while the execution of a task is paused, to be able to better perform said task or in its current environment.

3.1.2 Rizon

Another way adaptability in robots is commonly achieved is through the use of AI, which will be explored by taking a look at the Rizon robot which was created by Flexiv Ltd., founded from Stanford. Rizon utilizes direct force control along with AI to have the capability to have a good tolerance in terms of position variance as well as high disturbance rejection and the ability for quick redeployment between different projects or tasks. (Gundawar) These three things help differentiate the Rizon robot from other robots in the industrial field that strive to achieve similar tasks. These traits allow the Rizon robot to work particularly well when the environment is uncertain and it needs to be capable of performing tasks regardless of the environment and particularly when the environment is unknown prior to the tasks or the environment is prone to changing while the task is ongoing and the robot cannot be reconfigured and fixed. This is a common situation that can arise, as it is not often that while performing a task that a robot’s execution can be easily stopped to solve a problem that arises unless it is absolutely necessary to do so, i.e. the task is unable to be performed for whatever reasons. Allowing the robot to be able to adapt to this type of situation without having to stop and reset the robot and delay the tasks completion even more would allow for increased productivity from the robot and therefore more efficient usage of time for everyone as the task being performed will be able to be done continuously.

The Rizon robot also uses AI to be able to learn simple tasks by using computer vision to sense and recognize objects in its field of vision and using the information it gathers to generate motion as well as task plans, and also combine the simple tasks it knows and create more complex tasks. This use of AI allows for further use cases of the Rizon robot as having many simple actions it can take and then combine into more complex tasks allows for a more varied set of tasks to be done, and therefore expands the use cases of the Rizon robot.

The relevance of the AI approach to adaptability is that using something that can learn from its experiences and use those experiences to create new options for it to take and tasks to perform truly allows for adaptability through learning, rather than the adaptability through modularity that the Hive team will be executing. Both are valid, though different, strategies to adaptability, as adaptability through modularity allows for pre-planning to truly shine and allow for the knowledge of a task to be performed to translate better into adaptability as the right modules may be attached to the robot to give it the capabilities of performing the task at hand. However, adaptability through learning, such as the AI discussed previously, allows for more adaptability in the field and while performing tasks as it does not need to be removed from the field or have its execution stopped to be able to adapt to unexpected phenomena happening, and instead can learn from the unexpected phenomena and be prepared should that situation arise again, which allows for more flexibility in the design and programming process as fewer unexpected situations need to be accounted for. (Gundawar)

3.1.3 Self-Reconfigurable Robots

Self-reconfigurable robots are robots that have the capability to automatically change their shape and adapt to the environment around them to better perform their tasks or be capable of performing new tasks. While this is a phenomenal type of robot, the proper classification is something covered in a paper by Ning Tan, et al. called “A Framework for Taxonomy and Evaluation of Self-Reconfigurable Robotic Systems.” (Tan et al.) The division they make for classification purposes is inter-reconfigurability, intra-reconfigurability, and nested reconfigurability.

Intra-reconfigurability is used to define a robot that is a collection of components that act as single entities while being able to change their internal configuration without any external disassembly or assembly. They use an index to keep track of how many reconfigurations the robot has, with the more possible reconfigurations corresponding to a higher index number. Typically, intra-reconfigurability is used for modules pertaining to functionality, such as locomotion and mobility systems, grippers, and sensing capabilities. The majority of the intra-reconfigurable mechanisms contribute to enhancing locomotion capabilities of the mobility systems by transforming the robot’s mobility systems into a specific configuration that are more suited towards a type of locomotion and allows for greater adaptability to the surrounding environment and traversing through it. The typical way for achieving an intra-reconfigurable robot is through creating relative motion through the robot’s linkages, which act as joints and are the primary methodology through which intra-reconfigurability is achieved. (Tan et al.) An example of an intra-reconfigurable robot would be the aforementioned “Polybot” which was developed by the Xerox Palo Alto Research Center in

California. (Yim et al.) Inter-reconfigurability is defined as when multiple robots and complex structures can assemble together to form new configurations, which therefore means that the composite robots that make up the inter-reconfigurable robot are modular in nature, which is what allows them the inter-reconfigurability.

Inter-reconfigurable robots can be homogeneous or heterogeneous, where homogeneous refers to modular robots that have numerous of the same or very similar modules which allows for easy scalability, while heterogeneous refers to having different design and components between the modules which allows for greater versatility in the design as more unique and specific parts can be added to the robot easily. (Tan et al.) Both types have their advantages and disadvantages for usability and accessibility, where which one is chosen to be used will greatly depend on the tasks required to be performed for the given robot.

Nested reconfigurability refers to combining the advantages of both intra-reconfigurability and inter-reconfigurability by being a set of modular robots with individual reconfigurations that combine with other homogeneous or heterogeneous robot modules. Heterogeneous inter-reconfiguration can achieve the same configuration as nested reconfiguration although the reconfigurability of individual robot modules is different. The major benefit to nested reconfiguration is that nested reconfigurable robots are capable of generating more complex configurations for performing specific tasks that are too complex or difficult for a single unit to perform as well as being better for responding to programmable assembly requirements. A few examples of nested reconfigurable robots include the soft growing robot, the self-folding robot, and the Robogami. (Tan et al.) The robot that the Hive team is developing would best be defined under the inter-reconfigurable robot category. This is because, as stated previously, inter-reconfigurability is defined as essentially adding more onto the robot, which is exactly what the Hive team plans on for the modularity aspect of the agent, which will increase the adaptability of the agent and its use cases as a whole. This was chosen as it was determined to best fit the goal of the Hive team, which in this case is to allow for greater usage of the agent.

The initialization of the concept of the reconfigurable robot was done in a paper called “Approach to the Dynamically Reconfigurable Robotic System” by Toshio Fukuda and Seiya Nakagawa in 1988. (Fukuda and Nakagawa) Drawing from the idea of cells in living organisms, it was decided to propose a methodology of implementing a cell-like structure to robotic systems to allow each cell to detach and combine autonomously based on the task it is performing. This would function with the limited resources available to the robot as it is performing its task, and allow the robot to reconfigure itself for the purposes of achieving its task or extending its use time. The proposed system would follow three primary tenants, which are: consisting of several cells in a structure, each cell having some measure of intelligence, and each cell being able to combine and detach from each other automatically. Cells are defined in this paper as having three levels which scale in complexity, where the first level are basic cells that are useful in almost every configuration, such as joint cells, and the third level are the cells required for specific tasks that need some level of complexity or specificity in the robot design and cannot be done through the use of more basic cells. (Fukuda and Nakagawa)

The idea proposed for communication between the cells was that there would be a main control unit that could communicate with the cells and the cells would be able to communicate with each other as well as the main control unit. Optimal configuration is defined as being determined through the determination of the necessary and sufficient degrees of freedom, the joint combinations, and arm length between joints. Some mathematical properties are then defined as being given to each point, which are the required position and orientation for the manipulator end, the required force for the manipulator end, the required positioning accuracy for the manipulator end, a set position and orientation for the manipulator base, the type of an end-effector, and the physical constraints present in the working space. These are important to understand as the robot needs to be able to understand each of these to be able to perform its requisite task as well as be operational as a whole. The mathematical nature of the properties is irrelevant to the understanding of what they are and will therefore not be discussed here.

The coordinate systems relevant to this type of system are a coordinate system from the base of the manipulator and a coordinate system from the end of the manipulator. These are necessary coordinate systems as referencing from the base of the manipulator allows for a relative knowledge of where parts are in relation to the base, and a coordinate system referencing from the end of the manipulator allows for and understanding of where outside objects are in relation to the manipulator that will be manipulating them. The way to obtain the achievement of a line in one of the coordinate systems can be achieved by dividing the length of the line into the number of cells that will be used to execute said line and performing some basic calculations to be able to execute the production of the line. The joints do have constraints on how they can move, notable in the number of degrees of freedom they have, and any configuration that violates the constraints on how the joints can move must not be considered when taking into account how many possible configurations can be achieved as they would not be physically possible to execute. This is important to consider as if it is physically not possible for a robot to do something, then the robot should not be assigned that task that it cannot perform. This type of robot defined in this paper would be classified as intra-reconfigurable.

The relevance this paper has to the Hive team and the agent is that this paper could be considered the origin of modular robots, and it is always important to understand the history behind the same type of concept as what is currently being worked on. Understanding how modular robots were originally viewed versus how they are viewed today and the differences in how they are executed shows the trend of improvement and opens up further avenues for more improvement. For example, the agent will not be consistent with many cells that reconfigure themselves into different configurations to perform a task or sustain itself, but will rather be capable of having modules added and removed from it easily to be able to perform the task at hand.

A paper by Ning Tan et al. titled “Nested Reconfigurable Robots: Theory, Design, and Realization” covers nested reconfigurable robots, which is using a classification system from another paper authored by Ning Tan that was mentioned earlier and will thus not be the topic of discussion for this paragraph. The studied robot utilizes polyominoes, which are plane geometric figures that are formed by connecting equal cells edge to edge. Specifically, a type of polyominoes called tetrominoes are used for a robot named the Hinged-Tetro, which has a useful geometry related to nested reconfiguration as it can change its structure as well as combining with other Hinged-Tetros to form more complex configurations, which is within the definition of a nested reconfigurable robot. (Tan et al.)

While the agent for the Hive team does not fall under the definition of nested reconfigurability, it does fall under inter-reconfigurability of which nested reconfigurability draws a portion of its definition from, which makes this important to understand for the Hive team as it is indeed relevant. Hinged-Tetro uses omni-wheels, which the Hive team discussed using for the agent but ultimately decided against, to have sufficient mobility regardless of the shape it may currently hold. The energy consumption from the joint motors for each of the possible configurations was measured and recorded to be 1.48 W to hold a configuration and 1.96 to 3.7 W to perform a reconfiguration. The greater the number of joint rotations corresponds to the higher energy consumption. The Hinged-Tetro robot is 2D, which is more something to note than detracting from it being classified as nested-reconfigurable.

Notably, the Hinged-Tetro robot described and used for the paper is described as a prototype and had issues with the docking mechanism and the autonomous inter-reconfiguration between two or more modules, which is something being worked on. (Tan et al.) This impacts the Hive team as the agent will be collaborating with other agents that are the same and will have modules that can be attached to them to adapt them to the task being performed. While not exactly the same as the inter-reconfigurability discussed as part of the Hinged-Tetro, it is similar enough to draw some comparisons that can be used to help assist the Hive team with the production of the agent.

As the Hinged-Tetro robot moves around its cells to interact and merge with other cells, the agent can use this type of technique to draw on for collaborative purposes when working with other agents. It is of critical importance that an understanding of what has been done before be understood, as drawing from previous experiments and data is how advancement is made in any scientific field. Advancement and progress do not exist in a vacuum, and using the works of others far smarter than oneself to assist in the generation of ideas for solutions to problems that one is encountering is always valuable and in fact what many geniuses have done in order to be able to finish their product and create wonders.

3.2 Relevant Technologies and Concepts

These sections discuss the technologies relevant to the Hive team and the design and construction of the agent.

3.2.1 Mobile Power

In designing any electronic device, especially a mobile robotic solution, one of the most important components of the system is the power source. A power source is a device that supplies electrical energy to a load. As robots continue to become more sophisticated and capable of carrying out increasingly complex tasks, the need for reliable and efficient power sources is crucial to the success of this project. In this section we will look into the relevant technologies of mobile power.

3.2.1.1 Tethered Power Source

To deliver power to the robot our options include using a mobile power storage or a tether (wired connection).

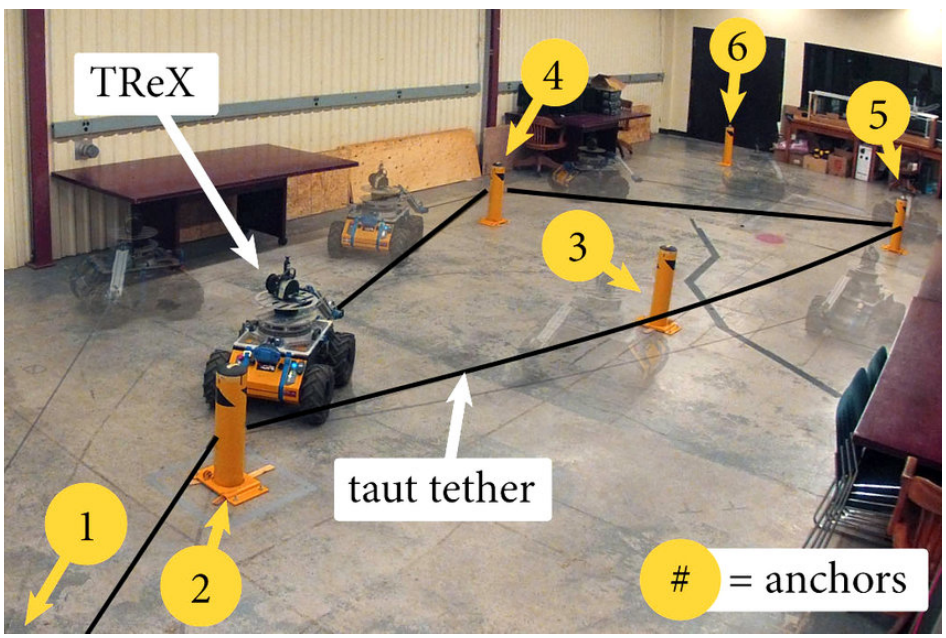

Figure 3.1: TSLAM: Tethered simultaneous localization and mapping for mobile robots (McGarey et al. 2)

A tether is a length of cable stretching from the robot to a stationary power source a distance away. Tethers are usually used to save on the volume and mass of a battery, keep a robot light, reduce volume, and to give communication resilience through a wired connection. But this has its constraints. Using a tether can become quite restricting and complicated, especially in mobile robotic solutions. Systems which implement tethers require a complex cable management solution. This is to prevent the tether from being tangled, damaged, or caught by protrusions and to help with the organization of the tether as it extends and retracts from the power source. Using a cable management solution, one's robot is forced to stay in a well developed area, preventing robots from being used outdoors as the tether can be stuck on limbs, rocks, trees, bushes, etc. Lastly, a tether solution is not suitable for mobile robotic solutions.

3.2.1.2 Batteries

The next option is to use an energy storage device, a battery. The structure of a battery consists of two electrodes, the negative anode and the positive cathode which are separated by an electrolyte. An electrolyte is a conductive medium that uses ions to transfer electricity but not electrons. When the battery is connected to a closed circuit, a chemical reaction occurs at the electrodes that produces an electrical current. The specific chemical reaction that results in this current depends on the battery type, and distinct types of batteries have individual advantages and disadvantages. These batteries also range in size, power density, and energy density, to power things ranging from small electronics to major grid power generation systems.

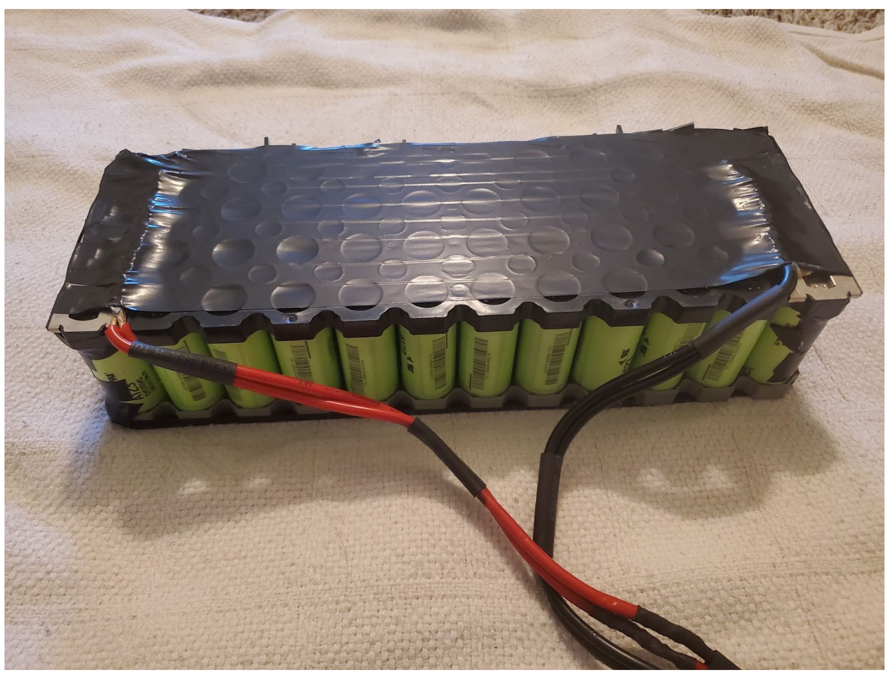

Figure 3.2: Mobile Power Source for an Electric Skateboard. Photo taken by Isaac Finley

A battery allows much flexibility in a robot environment and allows multiple robots to work together without the issues of cables tangling. A mobile power source is also more suitable in extending the operational distance of a robot. When originally the distance a robot could travel was dictated by the physical length of the tether cable to provide power, it is now dictated by the capacity of the onboard battery and wireless communication range. And with today's modern age, lithium-ion (Li-ion) batteries, which are very energy dense, allow for a lot of energy in a very small package and common wireless communications can reach up to 125 feet with more complex solutions reaching cell towers.

But these benefits in flexibility do come with their challenges. The first challenge to defeat is with the battery. Batteries are not free sources of energy and can be delicate to use. The most well known specificity of using batteries are their voltage requirements. Batteries normally have a safe voltage range at which they operate at their best and safest. For Li-ion batteries, their voltage range is between 3.0 and 4.2 V. If a battery is used in their under-voltage range, the battery may damage and permanently lose performance and if charged to above their rated voltage, some batteries have been known to catch fire and explode. As such, it is imperative that when using batteries, they are to be kept at a suitable voltage range depending on the battery used.

The next concern with using batteries in a robotic environment is that batteries are limited power storage devices. If a battery is not constantly connected to a power source when in use, at some point it will run out of energy. To be able to use the battery again it needs to be recharged, which takes time. Here lies the next issue with mobile power storages. There will need to be time taken out of the day to recharge the battery preventing continuous operation. If recharging is slow and/or required often throughout the day due to low capacities of the battery, the recharging downtime will be detrimental to the speed of accomplishing tasks. While recharge time can be decreased by raising the charging power, this comes at a cost of cycle-life, which is the next concern for batteries.

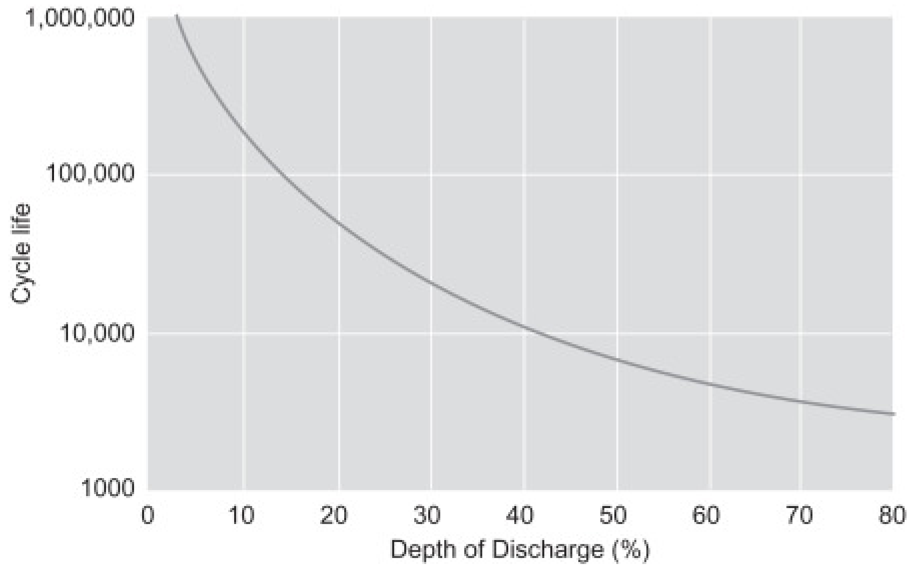

Figure 3.3: "The cycle life of batteries is the number of charge and discharge cycles that a battery can complete before losing performance" (Kalogirou)

For batteries, a charge and discharge is called a cycle, and as batteries go through hundreds of cycles their energy retaining ability gets reduced. The number of cycles a battery can go through before losing performance is called the "Cycle-Life". And this phenomenon of degrading chemical properties and physical integrity gets worse as a battery gets used more intensely. For example, a battery being charged at 2 A and discharged at 4 A will degrade much faster than a battery being charged at 0.5 A and discharged at 1 A.

And lastly with batteries, there is a game of optimizing between capacity. weight, energy density, and power density. If a battery does not have enough energy stored to be able to move the agent, then we cannot use a battery. A common solution is to add several batteries to work together as a team. But this can become complicated due to varying voltages between each battery cell and the weight of a team of batteries can become a large strain on the mobile agent. Thus, this becomes an optimization game of how much capacity can we house on our agents to increase operation time, and how much weight is too much weight for the agent to hold.

The next game is energy density and power density. Normally, an energy storage device compromises in one of those density fields. Some examples of this are batteries and supercapacitors. Batteries are energy dense devices, used best in situations of low power draw over a long duration of time. While supercapacitors are power dense devices, used best in situations where short bursts of energy are needed. A battery cannot be used in a power dense environment due to risks of damage, and a supercapacitor is not suitable for an energy dense situation because a supercapacitor cannot hold much energy without using an immensely large number of supercapacitors.

This is where our sponsor, Capacitech Energy, has made their novel invention, to put the two devices together to get the best of both worlds, to be able to use the large amount of stored energy of a battery and while getting the peak power capability of a supercapacitor for surges. The implementation of this system is discussed in further detail in following sections.

Mobile energy storage is not the only area where issues may arise when converting to a wireless solution. All communications need to become wireless as well. While wireless communications thrive in allowing flexibility of environment and are well suited for mobile solutions like cell phones and computers. It is obvious to most people that an unreliable connection can put a fast halt on operations. These delicate signals sent via radio can be intercepted by many obstacles and can be problematic at long distances for most common communication methods. Additionally, conflicting radio signals of similar frequencies can also alter data being communicated wirelessly. And as signals become weaker, data transmission speed also suffers. Without a strong connection and a fast transmission speed, agents may receive incorrect commands to do the wrong task, or fail to receive any stop commands and cause damage to the environment or to itself. And lastly, wireless communication is difficult to design into PCBs and often require long prototyping times. With many nuances including impedance matching and intricate trace placements to avoid harmful electrostatic frequencies from effecting data lines. Wireless communication comes with difficulties of application as well.

Despite the seemingly endless nuances of converting to a wireless solution our team decided to go with the use of a mobile power source. There are many factors that played into this decision. Firstly, because our goal of Hive is to create a modular robotic solution. We value the flexibility of mobile power source over a tethered solution despite the challenges it may bring during implementation. Secondly, there are many inexpensive and easily obtainable Battery Management Solutions (BMS) which include undervoltage and overvoltage protections. Thirdly, there have been many recent breakthroughs of energy dense lithium ion and lithium polymer batteries and power dense supercapacitors. A wide selection of capacities, life-cycles, discharge rates, and prices have been created which will aid in our design of a mobile power storage solution. Fourthly, using wireless communication would allow much easier adaptability within many workplaces as Wi-Fi is already installed in most buildings. Fifthly, wireless communication is becoming more user-friendly, with new technologies allowing easier use of wireless communications, including many pre-assembled packages to minimize design times. And the last factor that played into this decision is the fact that there are countless developers working with and having success with wireless communications and want to share their findings through online forums. These forums are communities of people who are willing to assist in development with one another if issues were to arise.

3.2.1.2.1 Exploration of battery options

As stated before, there are many different battery types and their most distinguishing features is their chemical compositions, or electrolytes. In the following sections we will go over some relevant technologies of battery chemistry types.

3.2.1.2.1.1 Lithium-Ion Batteries

By their name, lithium-ion batteries use lithium ions as the primary functionality of their chemical behavior. These batteries are lightweight and have a high energy density, which makes them ideal for our use case of an autonomous mobile robot. They can also be recharged quickly, which is important for our agents that have a minimum runtime of 30 minutes. Lithium-ion batteries are easily combined in series and parallel, making it easy to customize a battery with an exact fit for the application.

3.2.1.2.1.2 Lead-Acid Batteries

Lead-acid batteries are a type of rechargeable battery that uses lead and lead oxide as electrodes and sulfuric acid as the electrolyte. These are much heavier than lithium-ion batteries but are less expensive and more durable. Unlike lithium-ion batteries, which generally come in cells and can be combined, lead-acid batteries are often found in a box form-factor and not able to be combined in a space saving manner.

3.2.1.2.1.3 Lithium Iron Phosphate

Lithium Iron Phosphate batteries (LiFePO4) are a type of lithium ion rechargeable battery but with different ratings. LiFePO4 batteries charge and discharge in a similar way as other lithium ion batteries, but different strengths and weaknesses. Including increased power output, faster charging, reduced weight and a longer lifetime compared to other lithium ion batteries but come at a higher cost, a slightly lower operating voltage, and less energy density.

3.2.1.2.1.4 Nickel Metal Hydride

Nickel metal hydride batteries (NiMH) uses electrodes made out of nickel oxide hydroxide and a negative electrode using hydrogen-absorbing alloy. Their charging and discharging process occurs through a reversible hydrogen adsorption and desorption reaction. NiMH batteries have the positives of a decent cycle-life and durability but there are batteries in the lithium ion category which are cheaper and better performing in energy density, power density, and efficiency. So, NiMH batteries are not often used in today’s modern age.

3.2.1.2.2 Battery Comparison Table

The most important factors that were weighed in making a decision for the battery included energy density, power density, cycle life, and cost.

Energy density refers to the amount of energy that can be stored in a battery per unit volume or weight, measured in Wh/kg. A higher energy density means that a battery can store more energy in a smaller or lighter package. This is one of the most critical factors in selecting a battery, as we want the device to meet a minimum runtime requirement while not imposing a large load on the chassis and motors. Having a battery capable of storing the necessary energy for an agent is a non-negotiable specification.

Power density, on the other hand, relates to the amount of power that can be delivered by a battery per unit of volume or weight, measured in W/kg. We also want to maximize this, considering that our motors will consume a decent portion of power and will be towing a hefty payload.

Cycle life is the number of charge and discharge cycles that a battery can handle before its capacity degrades below a certain threshold. This is a parameter that will be crucial to consider, especially when this project is scaled up. We do not want the consumer to have to repurchase or repair batteries frequently, if at all. This is a robot that theoretically would perform nonstop duties in a warehouse over an extended period of time, and the battery must uphold this performance.

Table 3.1: Battery Comparison

Battery Type	Energy Density	Power Density	Cycle Life	Cost
Lithium-Ion	100-265 Wh/kg	250-340 W/kg	~1000 Cycles	$152/kWh
Lead-Acid	30-50 Wh/kg	180-260 W/kg	~ 300-500 Cycles	$549/kWh
Lithium Iron Phosphate	90-160 Wh/kg	2000-4500 W/kg	~ 2,700-10,000 Cycles	$131/kWh
Nickel Metal Hydride	70 Wh/kg	200 W/kg	~ 500 - 700 Cycles	$83/kWh

3.2.1.3 Supercapacitors

Supercapacitors are energy storage devices that store and release electrical energy quickly and efficiently. Like their lower performing counterparts, capacitors, they store energy in an electric field between two conductive plates separated by a dielectric. Compared to capacitors, supercapacitors have much higher capacitance in a much smaller form factor. For example, a fingernail sized capacitor may have a capacitance of 10 μF, while the same sized supercapacitor could have a capacity of 10 F, one million times greater. This difference being due to their different electrolyte and factoring into their costs. Our sponsor, Capacitech Energy, is involved with applications using supercapacitors, so as a funding requirement we will be using supercapacitors to complement the onboard battery.

Supercapacitors have several advantages over traditional batteries. They can charge and discharge much faster than batteries, making them ideal for applications that require high power bursts. They also have a longer lifespan than batteries, with up to 10 times more charge/discharge cycles. With that being said, supercapacitors cannot hold enough energy to operate the agents for the necessary runtime. For this reason, supercapacitors are added in parallel with the battery to complement it during peak power surges. Essentially, when the motors require a large amount of current, likely during the initial tow startup, the supercapacitors will aid the battery with a burst of energy that could otherwise damage the battery. Including the supercapacitors allows us to keep the battery healthy, while increasing peak power capability of the agents.

3.2.1.4 Solar Panels

Another consideration for mobile energy is to generate it on site. Solar panels, or photovoltaic (PV) panels, are components that convert sunlight into electricity. They are made up of multiple solar cells that are connected together and housed in a protective casing. When sunlight hits the solar cells, it excites electrons, which creates an electric current that can be used to power electrical devices or stored in batteries. To power the robot alone, there would need to be very large and efficient solar panels, so this isn’t reliable for our application. Rather, they would be complemented by batteries so that the batteries recharge via sunlight while the robot is in use. This would extend the lifetime of the agents in an environmentally friendly manner.

3.2.1.5 Voltage Regulators

Voltage regulators are electronic devices that are used to uphold a constant voltage level in an electrical circuit, generally taking an input voltage that is not entirely constant. Using regulators ensures that the voltage does not fluctuate too much and doesn’t cause damage to sensitive electronics, such as our motor drivers, motors, sensors, and microcontroller. There are a variety of voltage needs amongst the components in our project, including 3.3 V, 5 V, and 12 V, all with varying power requirements. In addition, the modularity aspect of our project requires that we have ample voltage buses in order to support a variety of attachments. We will have an individual regulator for each of these levels.

There are two main types of voltage regulators: linear and switching. Linear regulators work by using a variable resistor to control the output voltage, while switching regulators use a switching element like a MOSFET or diode, in order to regulate the output voltage. There are two main types of linear regulators: series and shunt. Series regulators use a series pass transistor to regulate the output voltage, while shunt regulators use a shunt transistor to regulate the output voltage. Linear regulators are simpler and less expensive than switching regulators, but they are less efficient and generate more heat. Switching regulators use a switching element, generally a transistor or diode, to regulate the output voltage. There are several different types of switching regulators, including buck converters, boost converters, buck-boost converters, and flyback converters. Switching regulators are more complex and expensive, but they are more efficient and generate less heat.

A DC buck converter is a specific type of switching regulator that is used to step down the input voltage to a lower output voltage. Due to the fact that we will be using a battery pack to power each unit, there will be a need to step down its voltage to various levels. Buck converters work by using a switch that turns on and off at a high frequency. When the switch is on, current flows through an inductor and charges up a capacitor. When the switch is off, the inductor releases its stored energy into the load. This phenomenon is pictured below in Figure 3.4. There are two main types of DC buck converters, including synchronous and non-synchronous converters. Synchronous buck converters use a rectifier, such as a MOSFET, to improve efficiency, while non-synchronous buck converters use a diode. This efficiency, however, comes with the tradeoff of being more expensive.

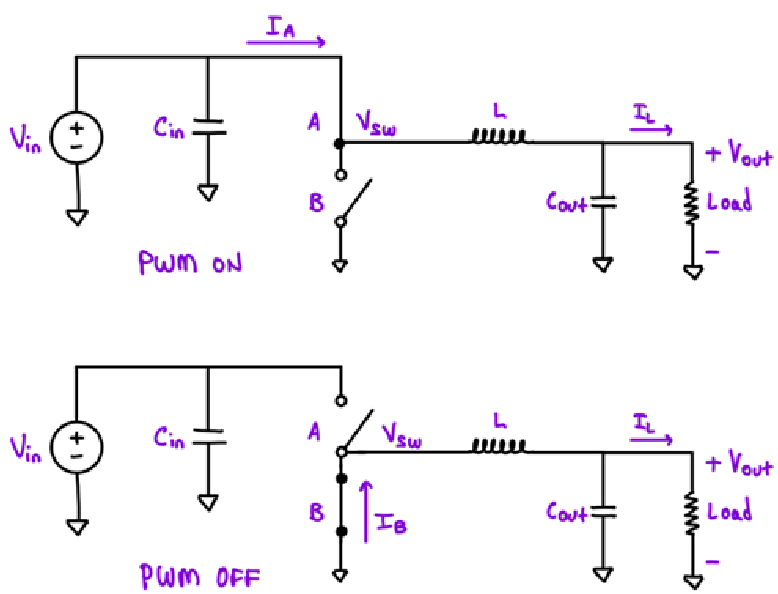

Figure 3.4 - Switching states of a DC buck converter. Designed by Benjamin Palladino

3.2.2 Motors

By nature, motors are vital to an autonomous robot. Not only are they needed for locomotion, but they are also used for the AMR to perform its designated work; in our case, towing. We will use motors to both drive the AMR, and operate the payload hitching mechanism. In searching for a suitable motor, we consider cost, size, reliability, power consumption, torque, weight, and precision.

3.2.2.1 Brushed DC Motors

Brushed DC motors are the most commonly used motors in autonomous mobile robots due to their ability for continuous rotation. They operate using the principle of Lorentz force, essentially using current to spin a rotor as the current-induced magnetic field interacts with that of the stator. They are generally considered the simplest of motors and are easy to configure for many applications. They operate on direct current and only require connection through a positive and negative power terminal. The rotation speed of a DC motor can be controlled either by varying its voltage or the strength of its magnetic field. These motors can be designed to provide either high torque at low or high speeds at low torque. In our application, torque and speed will be balanced so that the robot can perform its task efficiently, while still maintaining the power needed to tow heavier payloads. The downside to DC motors is that they comparatively have limited speed control, can introduce electromagnetic interference, and need to be regularly maintained.

3.2.2.2 Brushless DC Motors

Brushless DC motors are similar to brushed DC motors but do not have brushes, which reduces wear and increases efficiency. Brushed DC motors have carbon brushes that contact a commutator on the rotor, switching the direction of the current in the coil as it rotates. Brushless DC motors, on the other hand, use electronic controllers to switch the direction of current in the coil. Because they lack physical contact of brushes there is less internal friction, meaning they are more efficient, have greater reliability, and have lower maintenance requirements. They also offer greater speed control than brushed DC motors.

3.2.2.3 Servo Motors

Servo motors are a type of DC motor that can rotate to a specified angle defined by the user. These motors differ from DC motors in that they contain a feedback mechanism that allows them to maintain a specific position or speed even as the load changes. The control circuit communicates with a microcontroller that can dictate the desired position or speed of the motor. Based on this, the control circuit adjusts the power supplied to the motor to achieve the desired result. Servo motors are generally more expensive than standard DC motors due to their greater precision and control. For this reason, servo motors would be applied to the attachment mechanism rather than the drivetrain.

3.2.2.4 Stepping Motors

Stepping motors are another type of motor that is commonly used in mobile robots. As evident from their name, they operate by moving in small steps rather than continuous rotation, which makes them ideal for applications where precise positioning is required, like in 3D printers. These, like servo motors, have potential use in the attachment mechanism. Stepping motors are also capable of generating high torque at low speeds.

3.2.2.5 AC Motors

An AC motor is a type of electric motor that operates on an alternating current. An AC motor mainly consists of a stator and a rotor. The stator is the stationary part of the motor that contains the coils of wire that produce a rotating magnetic field when an alternating current is passed through them. The rotor is the rotating part of the motor that contains the conductive bars that interact with the magnetic field and produce torque. AC motors are commonly used in industrial applications due to their high efficiency and reliability but are still applicable to larger scale AMRs. They are preferred over DC motors in many cases because they require less maintenance and have a longer lifespan. Unlike DC motors, AC motors do not require brushes to transfer power from the power source to the rotor, which reduces wear and tear on the motor.

Compared to DC motors, AC motors have several advantages. For one, they are more efficient at converting electrical energy into mechanical energy. This is because they can operate at higher speeds without overheating, which allows them to generate more power per unit of time. Additionally, AC motors are more reliable than DC motors because of the previously mentioned lack of brushes. In an autonomous mobile robot (AMR), AC motors can be used to drive the wheels or in the attachments that manipulate objects. Despite these benefits, having an AC power source introduces a new layer of complexity to the agent, as all the other components will require DC power.

3.2.2.6 Motor Comparison Table

The following table ranks motors using the criteria of torque, efficiency, durability, controllability, and cost. Because the parameters for each motor can widely vary for products of higher cost/caliber, the table considers the most general and common specifications of each motor type.

Torque is a critical parameter in selecting a motor for the agent. Given that we are focusing our application on towing a payload through a warehouse setting, the motor must be able to not only provide enough torque for the maximum payload weight, but also support propel a robust chassis and overcome any friction from the environment.

Efficiency is another highly important factor in motor selection. Because the agent is powered by an onboard battery with a limited storage of energy, it is crucial that the most efficient components are selected. Having motors with a higher efficiency will increase our runtime which must be 30 minutes at a minimum.

Durability refers to the ability of the motors to continuously perform their duties without significant degradation. The overarching goal of the agent is to be as modular as possible, and having motors that can repeatedly execute commands over extended time periods is important to fulfilling this. We do not want a company to have to frequently service or replace the motors.

Controllability involves the ability to regulate and change the speed, torque, and direction of the motor's rotation. This is generally achieved through adjusting the voltage, current, or frequency of the input signal. Some motors are more responsive to these changes, and will therefore have a higher controllability.

Table 3.2: Motor Comparisons

Motor Type	Torque	Efficiency	Durability	Controllability	Cost
Brushed DC	High	High	Low	High	Lowest
Brushless DC	High	High	High	High	High
Servo	Low	Low	High	Highest	High
Stepping	Low	Lowest	High	High	High
AC Motor	Highest	Highest	Highest	Lowest	Highest

3.2.2.7 Motor Drivers

Motor drivers are electronic devices that control the speed, direction, and torque of an electric motor. Motors, on their own, cannot simply regulate these parameters. Drivers are necessary because electric motors require a more precise control of their power supply to effectively operate. Motor drivers work by converting an input from a microcontroller or other control circuit into a high-power signal that can drive the motor. Each driver is designed for a specific type (or several types) of motor and has its own unique set of features; this means that DC motors, stepper motors, and servo motors, for example, may each require separate drivers. In addition to control, motor drivers also provide protection against overvoltage, overcurrent, and other electrical faults that can damage the motors.

3.2.3 Sensors

Sensors convert a physical phenomenon into an electric signal that the microprocessor can interpret. Sensors can either be analog, meaning they generate a continuous output signal directly varying with the change of the measurement, or digital, meaning they generate a discrete and quantized output signal. There are several types of sensors that can be used for collision avoidance on the agents. These sensors include ultrasonic sensors, infrared sensors, laser range finders, and cameras. There are also limit switches, which can be used on the payload hooking mechanism to ensure that it has correctly been latched. Each sensor has its own set of advantages and disadvantages that must be considered when selecting the appropriate sensor for a particular application.

3.2.3.1 Ultrasonic Sensors

Ultrasonic sensors use sound waves to detect and measure distances of objects. They emit high-frequency sound waves, which bounce off objects and return to the sensor. The time it takes for the sound waves to return is then measured and used to calculate the distance between the sensor and the object, since the speed of sound is a known constant. These sensors are generally quite inexpensive and can detect objects at close range with fair accuracy. Their accuracy decreases as detection range increases, but our application will not be an issue for this. Their downside is that they can be affected by fluctuations in temperature and humidity, and may not work well in environments with a lot of acoustic interference since they rely on the measurement of reflected sound waves.

3.2.3.2 Infrared Sensors

Infrared sensors work by detecting the amount of infrared radiation emitted by an object. Infrared waves are a form of electromagnetic radiation whose wavelength is longer than visible light. Depending on its temperature, all objects emit infrared radiation in various amounts. This stream of radiation bounces off objects in the path of the sensor and is reflected back to it. The sensor then measures the time it takes for the radiation to return, and using the constant speed of light, it calculates the distance to the object. These sensors are relatively inexpensive and can detect objects at short to medium range. However, they can be affected by ambient light and could pose problems in environments with significant infrared interference.

3.2.3.3 Laser Range Finders

Laser range finders work by emitting a laser beam and measuring the time it takes for the beam to bounce back after hitting an object. The speed of the laser beam is the speed of visible light which is known and constant; so in conjunction with the time it takes for the laser to reach the sensor, it can easily calculate the distance to the object that it is reflected off of. These sensors are more expensive than ultrasonic or infrared sensors but are more accurate and can detect objects at longer ranges. Despite this, they require more power and are generally more expensive.

3.2.3.4 Cameras

Cameras capture images of the environment and require computer vision algorithms to detect objects in these images. They are relatively inexpensive but would require significant computational power to process images. Because a primary goal of the project is to offload major computation to a cloud-based server, this would be less than ideal for onboard collision avoidance. The processor used for driving motors, data transmission and reception, and sensor data processing would need to be strengthened in order to support a camera-based collision avoidance system. This would also require the development of a complicated algorithm, which is already being performed by the software team. Alongside this, cameras can also be affected by changes in lighting conditions in the environment.

3.2.3.5 Limit Switches

Limit switches are a type of electromechanical sensor used to detect the presence (or lack of presence) of an object (dependent on configuration) or to determine the position of a moving component. Limit switches contain a mechanical actuator that is triggered when it comes into contact with an object or when a moving component reaches a certain position. The state of the internal switch is then altered, indicating that the object or part has been detected or that a certain position has been reached. Because the latching mechanism is too small and precise for the overhead camera to accurately detect, limit switches will need to be used to ensure that the latching of the payload has occurred before towing it to its destination. Limit switches can be of three varieties, all of which have possible applications for our agents. Lever-type limit switches use a lever arm that is actuated by the object or moving part. When the lever arm is pressed, it activates the switch and sends a signal to the control system. Plunger-type limit switches use a plunger that is pushed in or pulled out by the object or moving part. Pushing or pulling the plunger would activate the signal. Rotary-type limit switches use a rotary actuator that rotates when the object or moving part reaches a certain position. If the actuator rotates, the switch is activated.

3.2.3.6 Sensor Comparison Table

Table 3.3 compares the sensors in terms of range, accuracy, complexity, cost, and field of view. There are some specialized versions of these sensors with profound range and accuracy capabilities, but this table only considers the commonly available devices on a lower end budget.

Range relates to the maximum distance at which a sensor can detect an object in its field of view. Having a greater range decreases the chance of collision, but there is also potential to simplify the design by having a binary collision detector, like a limit switch. The downside to this is that the robot will have to come into contact with an object for a moment in order to trigger the sensor.

Accuracy of a sensor is critical, as the sensor should be able to detect as many items in the way as possible. A false positive would result in the agent unnecessarily stopping, wasting time that could’ve been used to perform its duties. Missing the detection of an item, on the other hand, could result in the robot being damaged or someone being injured.

We also want to consider complexity, as adding unnecessary difficulty in processing sensor data could prove detrimental to the performance of the agent.

FOV was the last consideration, as it is important to consider multiple angles when discussing collision avoidance.

Table 3.3: Sensor Comparison

Sensor Type	Range	Accuracy	Complexity	Cost	FOV
Ultrasonic	<10 m	98-99%	Low	$5-50	15-60°
Infrared	<5 m	90-95%	High	$10-100	10-45°
Laser	<1 km	98-99%	High	$100-1000	0.25-5°
Camera	<5 m	Dependent on Algorithm	Highest	$20-5000	<180°
Limit	<1 cm	100%	Lowest	$5-50	N/A

3.2.3.7 Inertial Measurement Unit

An inertial measurement unit, or IMU, is an electronic device that measures and reports force, angular rate, and orientation of a body, using a combination of accelerometers, gyroscopes, and magnetometers. An IMU works based on the principles of Newton's laws of motion and the Coriolis effect. The IMU contains several sensors that each measure different physical quantities. The accelerometer measures linear acceleration in three axes and provides information about the movement of the robot in space.Knowing the acceleration due to gravity allows it to make accurate observations about these parameters, as shown in the left of Figure 3.5. The gyroscope measures angular velocity around three axes and provides information about the rotation of the robot. Using the known angular velocity and mass allows for calculations to be made regarding the object's rotational force, as shown on the right of Figure 3.5. The magnetometer measures the magnetic field around the robot and provides information about its orientation. As a failsafe, we will be including the IMU to correct the unit if it were to stray off a path, and the overhead camera alone could not detect it. The IMU will provide real-time information about the location and position of the robot, providing feedback to the unit itself, not the server.

The IMU combines all of these measurements to calculate the position, velocity, and orientation of the robot in real-time. The IMU uses complex algorithms such as Kalman filters to integrate these measurements and provide accurate position information. The Coriolis effect comes into play when a rotating object experiences a force perpendicular to its direction of motion. This effect is used by gyroscopes to measure angular velocity accurately. This device is not measured in the collision avoidance sensor table due to the fact that it is a standalone unit that will be used for certain, unlike the sensors of which only one will be selected. These aspects of the IMU are highlighted below in Figure 3.5.

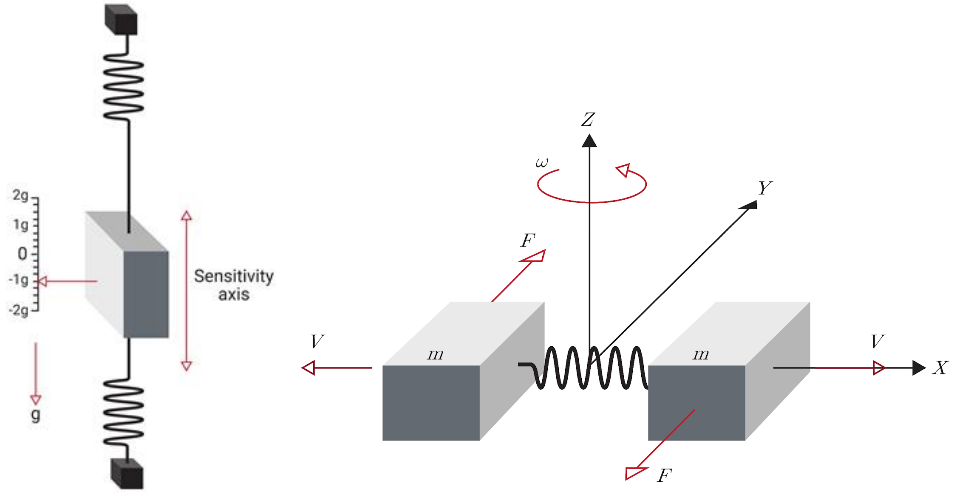

Figure 3.5 - Inertial measurement unit functional principles. (VectorNav).

3.2.4 Microcontrollers

Microcontrollers are small computers on an integrated circuit (IC) that are designed to control devices and processes. In this project, the microcontroller is the brains of the system; responsible for transmitting and receiving data, translating received data into motor actuation, attachment operation, providing the user interface, and processing sensor data. Microcontrollers generally consist of a central processing unit (CPU), memory, input/output ports, and various peripherals like analog-to-digital converters, and communication interfaces.

3.2.4.1 Microcontroller Architecture

The instruction set architecture, or ISA, of a microcontroller is the fundamental set of instructions that the processor can execute. The ISA defines the hierarchy of what the processor is capable of, how it uses registers, and how it addresses those registers. Software is built for a specific ISA, and cannot be run on just any ISA. The ISA also defines the types of data that can be processed, such as floating point and integer types. There are two main types of ISAs, each with their own benefits. CISC (Complex Instruction Set Computing) processors have a large instruction set with more complex instructions that can perform multiple operations in a single instruction. RISC (Reduced Instruction Set Computing) processors, on the other hand, have a smaller instruction set with simpler instructions that perform one operation at a time. RISC instructions are more primitive, but require more of them to perform a task that may have only taken one CISC instruction. There are many competing microcontroller architectures, both of RISC and CISC type, compared in the table below.

Table 3.4: Microcontroller Architectures

Architecture	Word-Width	Type	Company	Pros/Cons
AVR	8-Bit	RISC	Atmel Corporation	Pros: Low power consumption, high code density, ease of use. Cons: Limited memory, instruction set, and community support.
ARM	32-Bit	RISC	ARM Holdings	Pros: Energy efficient, high performance, scalability. Cons: Limited compatibility and support for legacy systems.
PIC	8-Bit	RISC	Microchip Technology	Pros: Low cost, ease of use, wide availability, integrated peripherals. Cons: Limited processing power, memory, and development tools.
MSP430	16-Bit	RISC	Texas Instruments	Pros: Low power consumption, cost-effective, wide range of peripherals. Cons: Limited memory, peripheral support, and development tools.
8051	8-Bit	CISC	Intel Corporation	Pros: Low cost, ease of use, versatile, large user community. Cons: Limited memory and peripheral support, lower speeds.

Architecture	Word-Width	Type	Company	Pros/Cons
Xtensa	32-Bit	RISC	Cadence Design Systems	Pros: Flexibility, scalability, low power consumption. Cons: Complexity, limited software support, higher cost.

3.2.4.2 Input/Output Interfaces

Microcontrollers have various input/output interfaces such as analog-to-digital converters, digital-to-analog converters, and pulse-width modulation outputs. Each of these interfaces take on distinct roles in processing data and controlling hardware. These interfaces will be necessary for both controlling motors and interpreting sensor data. No comparison table is provided, as all three of these technologies will be utilized in the project.

3.2.4.2.1 ADC/DAC

DAC and ADC are two important components that are used in robotics, primarily for sensor data interpretation. DAC stands for digital-to-analog converter, and ADC stands for analog-to-digital converter. They are essentially the complement of each other. A DAC is a device that converts digital signals into analog signals. The output of a DAC is a continuous signal that can be varied over a range of values. An ADC does the opposite. It takes an analog signal and converts it into a digital signal that can be processed by a microcontroller. This is arguably more important for our application because most sensors produce analog signals. By converting these signals into digital form, they can be more easily processed and analyzed by the robot's control system. In an autonomous mobile robot, both DACs and ADCs are needed to control the various motors and sensors that are used to navigate and interact with the environment. ADCs and DACs can be realized through various methods, but typical applications feature the use of operational amplifiers used as comparators. The usage of these two components will be dependent on the type of sensors incorporated.

3.2.4.2.2 PWM

PWM, or pulse-width modulation, is used to control the amount of power delivered to a device by rapidly switching it on and off. In robotics, PWM is commonly used to control the speed of motors and the brightness of LEDs because it allows for control of motor speed by varying the duty cycle, which is the percentage of time the motor is on compared to off. Without PWM, the motor would only be able to operate at a single speed, which would limit the functionality of the robot.

3.2.4.3 Serial Communication Interfaces

Serial communication interfaces are used for device-to-device communication over some channel, where the data is transmitted sequentially. This is in opposition to parallel communication, where multiple bits can be sent over multiple lines simultaneously. Such communication is necessary for the microcontroller to be programmed, as well as to communicate with sensors and other hardware.

3.2.4.3.1 UART

UART stands for universal asynchronous receiver/transmitter. Asynchronous communication means there is no clock signal shared between the two devices. Data is sent byte by byte, each beginning with a start bit, followed by the actual data bits, and then a stop bit. The transmitter sends each bit in the byte sequentially, starting with the least significant bit (LSB) and ending with the most significant bit (MSB). When receiving data, the UART device waits for the start bit to arrive and then starts reading each subsequent bit until it reaches the stop bit. Occasionally a framing error occurs when the receiver does not detect a valid stop bit at the expected position in the data frame. This bit scheme is shown below in Figure 3.6. UART is commonly used in many electronic devices such as microcontrollers, computers, and communication systems. It is a simple and reliable method of transmitting data over short distances.

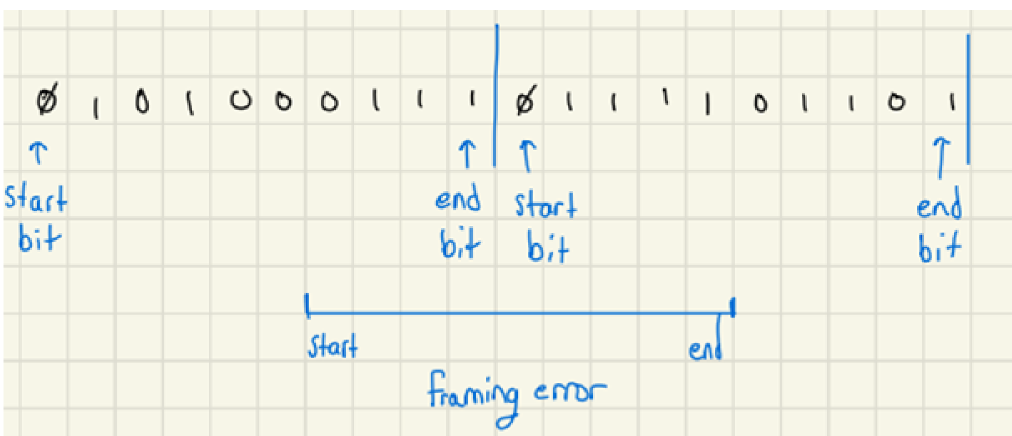

Figure 3.6 UART transmission data scheme. Designed by Benjamin Palladino

3.2.4.3.2 SPI

SPI stands for serial peripheral interface. SPI is what is known as a full-duplex interface, which means that data can be transmitted and received at the same time. The SPI interface consists of four signals. The first signal is the SCLK line, or the serial clock. This line is essentially a clock signal, which is used to synchronize the data transfer between the master and slave devices. The MOSI line (master out slave in) is used by the master device to send data to the slave device. The MISO line (master in slave out) is used by the slave device to send data back to the master device. Lastly, the SS line (slave select) is used by the master device to select which slave device it wants to communicate with. The SPI protocol sends data in a series of 8-bit frames. The master device first initiates communication using the SS line to select a slave device. The clock signal on the SCLK line then begins, and both devices can transmit and receive data simultaneously on the MOSI and MISO lines. This is implemented in its most primitive form below, in Figure 3.7. The data transfer rate of SPI is dependent on the frequency of the SCLK signal.

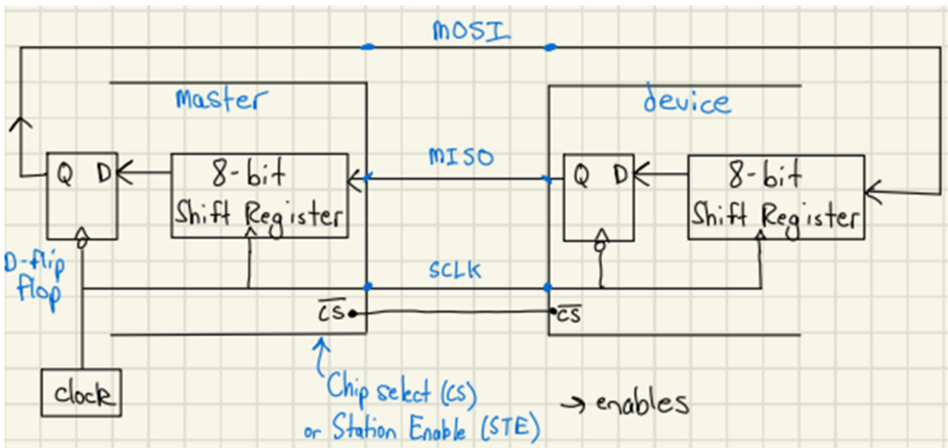

Figure 3.7 - SPI protocol hardware diagram. Designed by Benjamin Palladino

3.2.4.3.3 I2C

I2C, or inter-integrated circuit, is a two-wire bus that allows multiple devices to communicate with each. The I2C bus consists of a serial data line (SDA) and a serial clock line (SCL). This hardware layout is shown below in Figure 3.8. The SDA line is bidirectional, meaning data can travel both to and from connected devices, and is used to transmit data between said devices. The SCL line synchronizes the data transmission between devices. Both lines are pulled high by pull-up resistors and are driven low by the devices on the bus. This means that the lines are normally connected to a digital high voltage, and pulled to ground when needed for operation. I2C is distinctive in that each device on the I2C bus has a unique address that is used to identify it. Depending on the device, the address can be either 7 or 10 bits. The I2C interface has various behaviors, including read, write, and combined read/write. In a write transaction, the master device sends data to a slave device. In a read transaction, the master device receives data from a slave device. In a combined read/write transaction, the master device sends data to a slave device and then receives data from the same slave device. Firstly, the master device sends a start condition by pulling the SDA line low while allowing the SCL line to remain high. The master then sends the address of the particular slave device it wants to communicate with. The slave can either acknowledge the reception of its address and communication will proceed; or, an error occurs. After addressing a slave device, the master can send or receive data by toggling the SCL line. Data is transmitted from MSB to LSB. The receiver acknowledges the reception of each byte by pulling the SDA line low during the ninth clock pulse. To end a transmission, the master sends something called a stop condition by allowing both lines to be high.

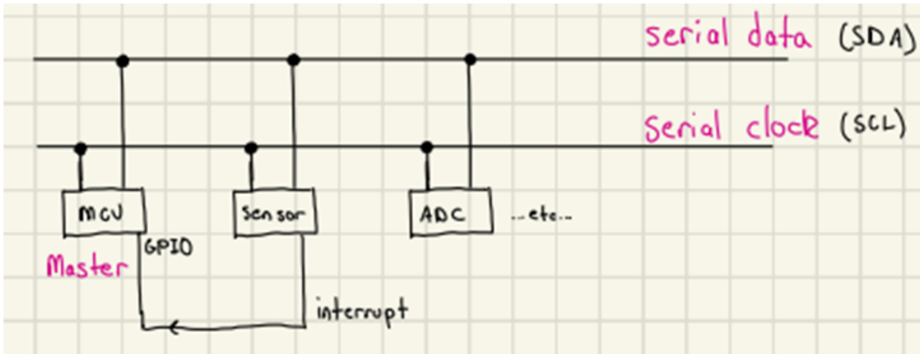

Figure 3.8 - I2C hardware example schematic. Designed by Benjamin Palladino

3.2.4.3.4 Two Wire Automotive Interface

Two Wire Automotive Interface, or TWAI, is a protocol used to enable communication between various electronic control units (ECUs), most often within automobiles. TWAI is designed to provide reliable and deterministic communication between ECUs. It works by dividing time into fixed intervals, known as time slots. Each ECU is assigned one or more time slots during which it can transmit data. This ensures that all ECUs have equal access to the network and that there are no collisions or conflicts between messages. The time slots are determined by a master clock that synchronizes all the ECUs on the network.

CAN bus, or Controller Area Network bus, is one example of a communication protocol that can be used with TWAI. In a CAN bus system, each ECU has a unique identifier that determines its priority on the network. When an ECU wants to send a message, it checks the bus to see if any other ECUs are currently transmitting. If the bus is idle, the ECU can start transmitting its message. If another ECU is already transmitting, the ECU waits until the transmission is complete before attempting to transmit its own message. This topology is highlighted in Figure 3.9. CAN bus also uses error detection and correction mechanisms to ensure that data is transmitted reliably. Each message includes a checksum that allows the receiving ECU to detect any errors in transmission. If an error is detected, the receiving ECU requests that the message be retransmitted.

TWAI and CAN buses are technologies that could enable reliable and safe communication for the agents. By using fixed time intervals and prioritization schemes, these protocols ensure that critical messages are transmitted without delay or interference.

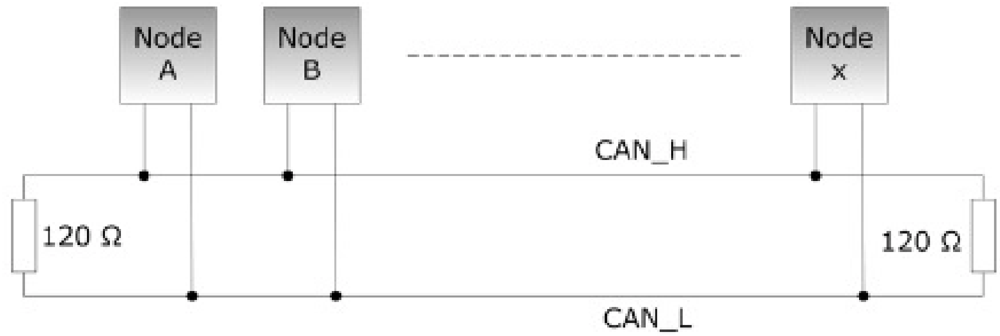

Figure 3.9 CAN bus topology. (Voss)

3.2.4.3.5 Serial Communication Interface Comparison

Four key factors that are critical for serial communication interfaces in our agents are data transfer rate, latency, and compatibility. We neglected the cost of these protocols because they are generally very similar to apply to the scale of our project.

Data transfer rate is one of the most important factors for the agents. This refers to the speed at which data can be transmitted between different components of the robot. A high data transfer rate is essential for real-time control and feedback, which is crucial for applications such as navigation, and obstacle avoidance. The interface must be able to transmit data quickly and reliably to ensure that the robot can operate effectively in dynamic environments.

Latency is another important factor to consider, which refers to the delay between sending a command or receiving data and the corresponding action being taken by the robot. Low latency is essential for real-time control applications where delays can cause errors or even accidents. The interface must be designed to minimize latency and ensure that commands are executed quickly and accurately.

Compatibility is also an important consideration, especially given the modularity needs of the robot. The interface must be compatible with all the components of the robot, including sensors and controllers. Compatibility issues can cause delays or errors in data transmission, which can affect the overall performance of the robot.

Table 3.5: Serial Communication Interface Comparison

Protocol	Data Transfer Rate	Latency	Compatibility
UART	<3 Mbps	ms-μs	Highest
SPI	< 10 Mbps	μs-ns	High

Protocol	Data Transfer Rate	Latency	Compatibility
I2C	<3.4 Mbps	μs	High
CAN	~1 Mbps	ms-μs	High

3.2.4.4 Wireless Communication Protocols

Wireless communication is a form of telecommunication that uses electromagnetic waves, such as radio waves, microwaves, infrared radiation, and visible light to transmit information over a distance. Largely due to the scope and cost of our project, we will be focused on radio frequency technology, as these are widely available and suitable for our small scale application. Radio frequency refers to electromagnetic waves in the range of about 3 kHz to 300 GHz.

Wireless communication as a whole works by converting data into electromagnetic waves that can be transmitted through the air. The process involves three main components: a transmitter, a receiver, and a medium. The transmitter converts the data into an electromagnetic signal and sends it through the medium, which could be air or space. The receiver then captures the signal and converts it back into data.

While wired serial communications can be used onboard, it would not be feasible to have a robot tethered to the main router which dictates its behavior. For this reason, our microcontroller unit will communicate wirelessly using one of several radio frequency communication protocols. Each protocol comes with its own advantages and disadvantages. Because our agents are commanded by a cloud based server, wireless communication is imperative in navigating the agents around their environment. The following table summarizes the operation of major wireless communication protocols, as well as their pros and cons.

3.2.4.4.1 Wi-Fi

Wi-Fi operates on frequencies ranging from 2.4 GHz to 5 GHz, which are unlicensed frequencies available for public use. The Wi-Fi network consists of two main components: the access point and the client device. The access point acts as a central hub that connects multiple client devices to the network. The client device can be a laptop, smartphone, tablet, or any other device with Wi-Fi capabilities. When a client device wants to connect to a Wi-Fi network, it sends a request to the access point. The access point responds by sending a signal that contains the network name and password. Once the client device receives this information, it uses it to connect to the network. Once connected, the client device can send and receive data over the network. The data is transmitted in packets, which are small units of data that are sent and received by the devices. The packets are received by the antennas of the client device or access point.

3.2.4.4.2 Bluetooth

Bluetooth uses the same 2.4 GHz frequency band as Wi-Fi, but it operates on a different channel. It uses a technique called frequency hopping spread spectrum (FHSS) to avoid interference with other devices that use the same frequency band. FHSS involves rapidly switching between different frequencies within the band, which makes it difficult for other devices to interfere with the signal. When two bluetooth-enabled devices want to communicate with each other, they first need to establish a connection. This involves a process called pairing, where the two devices exchange information and create a secure link between them. Once the connection is established, data can be transmitted between the devices.

One of the key advantages of Bluetooth is its low power consumption. This makes it ideal for use in small, battery-powered devices. Bluetooth also has a relatively short range compared to Wi-Fi, which makes it more secure since it is less likely that someone outside of the immediate area can intercept the signal.

3.2.4.4.3 Zigbee

Zigbee is a wireless communication protocol designed for low-power, low-data rate, and short-range communication. It operates on the IEEE 802.15.4 standard and uses radio frequency communication to establish connections between devices. Zigbee uses a mesh network topology, where each device can act as a router to extend the range of the network. This allows for greater coverage and reliability compared to other wireless protocols. Zigbee also uses a unique addressing scheme that allows for up to 65,536 (2^{16}) nodes on a single network.

In terms of RF communication, Zigbee operates in the 2.4 GHz frequency band and uses direct sequence spread spectrum (DSSS) modulation. DSSS spreads the signal over a wider frequency range, making it more resistant to interference and allowing multiple devices to communicate simultaneously without interfering with each other.

3.2.4.4.4 Wireless Communication Protocol Comparison

When selecting a wireless communication protocol for the agent, several factors needed to be considered to ensure optimal performance and efficiency. The most important of these factors include range, bandwidth, power consumption, and cost.

Range is a crucial factor in selecting a wireless communication protocol for the agents. It refers to the maximum distance between the robot and the base station or control unit that the wireless signal can travel without significant degradation. While the scenario for our project may be centered in a constrained environment, the application is destined for a warehouse which could span several hundred feet and have many obstacles between the transmitter and receiver. A longer range allows the robot to operate over a larger area, making it more versatile and useful. However, a longer range usually comes at the expense of other factors such as power consumption and cost.

Bandwidth is another important factor. It refers to the amount of data that can be transmitted over the wireless link per unit time. A higher bandwidth allows for faster data transfer, which is crucial for real-time applications. However, higher bandwidths also require more power and can increase the cost of the system.

Power consumption is also a critical factor, given that the agents will be battery operated with a high operating time requirement. The wireless communication module must consume minimal power to extend battery life and reduce downtime.

Cost is another essential factor to consider when selecting a wireless communication protocol for an autonomous mobile robot. The cost of the wireless communication module must be reasonable and within budget while providing optimal performance. A high-cost module may provide better performance but may not be feasible for some applications.

Table 3.6: Wireless Communication Protocols

Protocol	Range	Bandwidth	Power Consumption	Cost
Wi-Fi	100 - 150 ft	600 Mbps - 1.3 Gbps	Low	Low
Bluetooth	~200 ft	2 Mbps	Lowest	Medium
Zigbee	~230 ft	250 Kbps	Low	Medium

3.2.5 Newton's First Law

Newton's First Law states a body in motion tends to stay in motion and a body at rest tends to stay at rest, unless otherwise acted on. An example of this is friction, whichis a force that is created when two bodies move alongside each other. Friction is always in the direction opposite of motion and tends to slow bodies until they come to a complete stop. This concept of friction is used in drivetrain design, and in the wheel to ground interface. A wheel with an applied torque only drives forward thanks to the friction, as without this the vehicle would not move forward. Using these concepts and constraining some physical parameters of the wheels planned to be used for this project, the required torque of the motor can be estimated, aiding in the proper selection of motor and gearbox pairs.

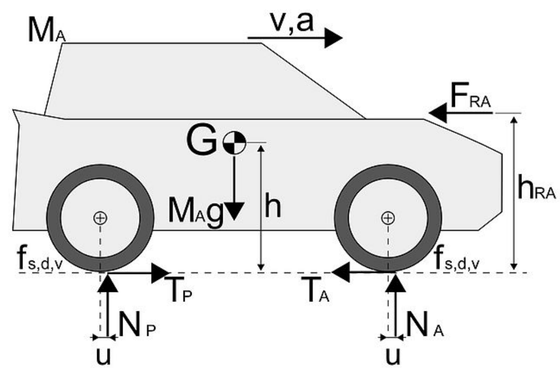

Figure 3.10 Example of Newton's First law and how it applies to wheels on a vehicle. (Nappi)

3.2.6 3D Printing and Manufacturing

With a major goal of our project being modularity, we wanted to ensure that we could quickly produce parts that could change the dynamic of our robot. With a group member already having a 3D printer it seemed obvious to utilize such a tool. This idea quickly evolved into making assemblies that use little to no hardware. With this in mind, it was necessary to further explore all the methods of 3D printing.

In recent history the concept of 3D printing has quickly evolved into an enormous industry. With the first 3D printer originating in 1981, which was only used in industry for quick prototyping and not available to the public; the designs of printers have been iterated, leading to many different types of printers in today's market. Starting around 2005, the 3D printer began to gain attraction and quickly moved to what it is today. At this time anyone with the means and motivation can get their hands on a printer and begin creating. With these concepts becoming more mainstream and the increasing ability to print with different materials, the possibilities are endless. The three main types of printers on today's market are Stereolithography (SLA), Selective Laser Sintering (SLS), and Fused Deposition Modeling (FDM). Each of these types of printing methods have their pros and cons with certain ideal applications that make them shine.

3.2.6.1 Stereolithography (SLA)

SLA was the method used in the world's first 3D printer in 1981. Today it is still one of the most used technologies in industry. This type of printing revolves around the concept of photopolymerization, "photopolymerization is a process in which smaller monomers are linked to become a chainlike polymer through a photochemical reaction that is aided by a catalyst", or in other words, polymerization that is induced by light. Ultimately a solid model is created shining ultraviolet light through a liquid resin that induces polymerization.

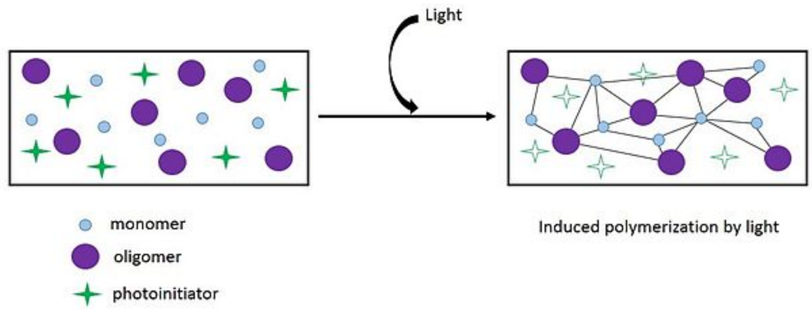

Figure 3.11 Example of how UV light induces polymerization and the polymer structure begins to form. (Iriidium)

With the understanding of how the physical material is created, the next step of fully understanding how 3D printers work is understanding the software side of this manufacturing process. Across all types of 3D printing G-Code is created based on a 3D model file. In this instance a laser at the bottom of the transparent body holding the resin references the G-Code to coordinate the exact position to permanently cure the resin one layer at a time using Digital Light Processing (DLP). As layers are created a stepper motor moves the base of the part up as it continues to be printed and there is also tilt actuation that allows for another axis of manufacturability. This type of printing offers higher resolution than SLA,but is dependent on the intended pixel size of the monomer. With this, the highest resolution is contained to smaller build areas restricted to the X and Y axes. The maintenance on these machines are very similar to your typical printer, aside from the light source. These printers are very similarly constructed. Considering all the characteristics of these printers, it is now time to look at the pros and cons of printing in such a manner. Starting with the pros: resin 3D printing allows for rapid functional prototyping, tight tolerances, smooth surfaces, mold creation, and a wide range of materials. This type of printing is used in several industries, including manufacturing engineering, dentistry, and the jewelry industry. While this may seem too good to be true, there are some drawbacks that come with this type of printing such as: high cost, it is environmentally unfriendly, it requires skilled labor to assemble, and the finished printers are weaker than their counterparts. Pictured below is an example of the complex shapes that can be created with such techniques with no support material needed.

Figure 3.12 Example of how SLA printing can produce complex structures without support material needed. (Gustavo Rocha SI)

3.2.6.2 Selective Laser Sintering (SLS)

Selective Laser Sintering 3D printers use a high-power laser to sinter small particles of polymer powder, creating a complete structure. The unfused polymer powder supports the part in the process eliminating the need for support material. The main benefits of using this type of 3D printing are having the capability to create complex geometries with negative space features. The mechanical characteristics of the final part are much more robust than parts created with other types of printing, and SLS allows for custom manufacturing. These machines operate off of their own type of G-Code and often use a nylon-based powder to print, which produces the most lightweight, cost effective, and strongest part. To better understand how this type of 3D printing works a Figure 3.13 has been placed below to help illustrate this process.

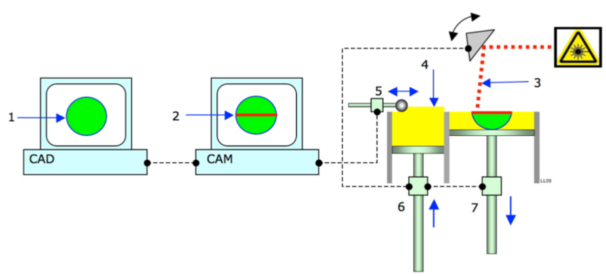

Figure 3.13 Example above shows all of the moving parts in SLS printing. (Lieshout)

This diagram shows the main components and moving parts of these printers. To manufacture these parts, the machine sinters layer by layer until the appropriate shape of material has been created. To do this, nylon powder is placed on a heated bed. Once the laser has sintered it a roller moves across the bed to disperse the next layer of powder for the next layer. Once the part is completed, it is removed from the bed and bead blasted to ensure that the part is clean.

The greatest downside of this type of 3D printing is the cost. Out of the three types of 3D printing, it is the most expensive with SLA coming in second and FDM coming third. This is important when considering that these printing methods are capable of quickly producing functional prototypes and end-use parts.

3.2.6.3 Fused Deposition Modeling (FDM)

Fused Deposition Modeling (FDM) also known as fused filament fabrication (FFF) can produce basic proof of concept models and simple prototyping by extruding thermoplastic filaments through a heated nozzle, completed layer by layer. This type of printing uses the cartesian coordinate system to move a nozzle and heated bed in the XY and Z directions. Out of the three listed methods of 3D printing FDM has the lowest resolution and accuracy, making it difficult to achieve high-quality finishes. This can, however, be improved with post-processing methods. To accomplish this, spools of different types of thermoplastics pulled into filament are extruded layer by layer. The two main types of filaments used in this are Polylactic Acid (PLA) and Acrylonitrile Butadiene Styrene (ABS), which are some of the most common thermoplastics used for this application due to their lower cost and accessibility. Keep in mind when looking at these filaments, most of them are umbrella terms for a wide range of thermoplastics. So, depending on your exact application it may be worth going with a variation of the material. It is important to note that following the same pattern as cheaper materials, this type of printing is the most affordable and therefore used heavily in hobbyists' DIY projects. The scope of what is printable with this method is much slimmer due to the constraints of the cartesian system making it to where support material is needed for more complex shapes. In a more industrial setting these printers have industrial counter parts that utilize a thermoplastic called Polyvinyl alcohol (PVA) to help fix the issue of support. PVA is a water-soluble synthetic polymer that is utilized to help mitigate the issues caused from support material in 3D printed parts. To effectively use this thermoplastic, a dual extruder FDM printer is used with the primary material loaded into one extruder (typically PLA or ABS), while the second extruder contains PVA which is used solely for the required support material of the printed part. When applying this technique, the finished part is made of two materials; rather than cutting away the support like you would with just one material, the PVA will just need to sit in water for around 4-6 hours and the support will dissolve into water, leaving only the finished product behind. These types of printers are advancing more and more and becoming increasingly accessible to the public and have their place in industrial settings. Pictured below in Figure 3.14 is an example of an FDM printer in use.

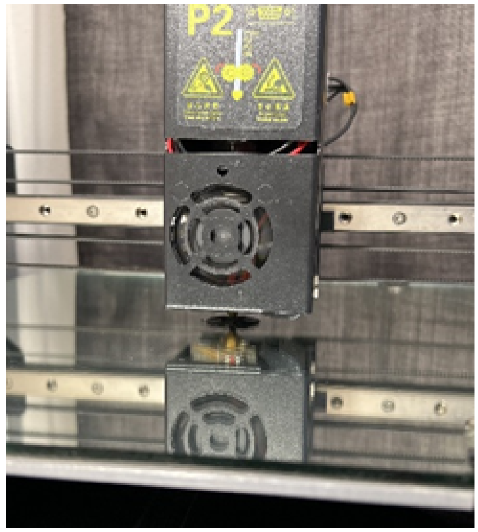

Figure 3.14 Example of FDM printing in action taken by Cameron Nichols.

3.2.6.4 Manufacturing silicon

During our project, we anticipate that we will need to design and manufacture rollers for a mecanum wheel, which need to have a high coefficient of friction. Silicon is a hybrid between a synthetic rubber and synthetic plastic polymer that is very tacky in nature, coming in with a coefficient of friction of around 1.0 in most cases. This seems to be the best candidate for roller material choice, with rubber following up close in second. With silicon technology advancing in recent years, there are more and more options out there for producing quality parts using silicon. While still being a relatively new technology, 3D printing silicon is possible but has some unique issues from a material standpoint and cost. Due to its high viscosity when it is melted, it makes 3D printing precise parts very difficult, as it cannot be heated and extruded using FDM printing or cured with UV light using SLA printing. However, one company by the name of Spectroplast, has created a proprietary process that can make them light-sensitive, allowing for SLA processes to be used. Rather than using the pricey solutions of directly 3D printing silicon, a more budget friendly option is 3D printing a mold that can then be used in techniques such as, injection molding, thermoforming, compression molding, and silicone casting, which all allow for custom end-use silicon parts.

3.2.6.5 Injection Molding

Casting has been a method of manufacturing for years, where a molten material (usually a type of steel) is poured into a mold allowing it to take on a new shape, after cooling. The key difference between casting and injection molding is the way the molten material is placed into the mold. In casting the acting force on the liquid material is gravity, whereas, in injection molding the material is introduced into the mold at a high pressure. Both processes have their pros and cons. Casting can produce higher precision parts at a cost of time and ultimately cost. While injection molding does have some rather large startup costs, the entire process is much quicker - lasting anywhere from 2 seconds to 2 minutes in many cases. For prototyping, injection molding is the better option, especially when many parts from this process are adequate for end use, considering that this type of molding can still produce rather tight tolerances. It is at this point that I would like to mention that this could be one of the best options for creating our rollers, assuming cost and availability fall in line. Bringing this back into the scope of manufacturing silicone, complete silicone parts can be produced using injection molding with Liquid Silicone Rubber (LSR) acting as the working material. Pictured below is Figure 3.15 showing all of the moving parts in the injection molding manufacturing process.

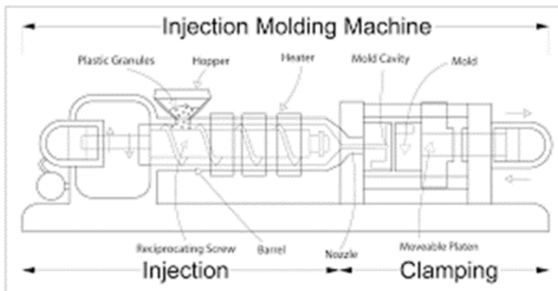

Figure 3.15: Diagram of the moving parts in the injection molding process. (Rockey)

3.2.6.6 Thermoforming

Thermoforming is another type of manufacturing using molds that is like injection molding but is used for certain parts where the geometry requires the negative space of the mold to be formed. So rather than a liquid material filling a cavity, the heated sheet of material is pressed by the mold and forms the final part in this manner. This manufacturing process can be better than others depending on the final shape of the required geometry. It is important to note that with the plans of creating our rollers out of silicon we will more than likely not use this process, however, it is worth noting as another method of potential prototyping.

3.3 Strategic Components and Part Selection

In this section, we will present the decision process that we used to decide on electrical and mechanical components for our ground agents. These components include the embedded system, motors, attachment system, voltage regulators, batteries, and sensors.

3.3.1 Microcontroller

This section goes over how the decision was made on what microcontroller would be used. When choosing the microcontroller, our team primarily focused on ensuring there would be a sufficient number of pins available for our application, including pins dedicated to controlling the four driving motors, pins dedicated to reading sensor data, pins dedicated to interface with attachments, etc. We also wanted to make sure that the microcontroller was either capable of wireless communication or capable of interfacing with a transceiver to achieve wireless communication, that way we could effectively communicate with the robot router. We looked primarily at two microcontrollers to choose from: the ATmega2560 microcontroller from Atmel, and the ESP32-WROOM-32 microcontroller unit module from Espressif Systems. Both of these microcontrollers are relatively easy to work with and are compatible with the Arduino IDE, which many members of our team are familiar with. Due to the time constraints of our project, ease of use and familiarity were heavily weighted and is what led to us considering these two microcontrollers.

3.3.1.1 Atmel ATmega2560 Microcontroller

The ATmega2560 is an 8-bit microcontroller that uses the AVR Reduced Instruction Set Computer, or AVR RISC, architecture. It can achieve clock speeds of up to 16 MHz at an operating voltage range of 4.5 to 5.5 V. It has 256 KB of in-system self-programmable flash memory and 8 KB of SRAM. The microcontroller also includes 32 8-bit general purpose working registers with a total of 86 programmable I/O lines. Six of the 32 registers can be used as three 16-bit indirect address register pointers for Data Space addressing, which enables efficient address calculations. It also has four 8-bit PWM channels, twelve PWM channels with programmable resolution from 2 to 16 bits, and a 16-channel, 10-bit analog to digital converter. The microcontroller also features four programmable serial USART, a master/slave SPI, and a byte oriented 2-wire serial interface. It also offers two 8-bit Timer/Counters with separate prescaler and compare modes as well as four 16-bit Timer/Counters with separate prescaler, Compare- and Capture modes. The timer modules with the PWM feature can be used to vary LED brightness, motor speed, or other devices that utilize a variable frequency, which is of utmost importance since we will be driving our agents with motors. We will now discuss the different timer modules present within the ATmega2560.

The first timer module the microcontroller offers is Timer/Counter0. This is a general purpose 8-bit Timer/Counter module with two independent Output Compare Units, two double buffered Output Compare Registers, and PWM support. This module also features three independent interrupt sources. Timer/Counter0 has four modes of operation: normal mode, clear timer on compare match mode, fast PWM mode, and phase correct PWM mode. The fast PWM mode provides a high frequency PWM waveform generation option that can have a frequency twice as high as the phase correct PWM mode. This high frequency makes this mode well suited for rectification, power regulation, and digital to analog conversion, while also allowing physically small sized external components, such as capacitors, therefore reducing the total system cost. The phase correct PWM mode provides a high-resolution phase correct PWM waveform generation option that is preferred for motor control applications.

Another timer module that the microcontroller offers is Timer/Counter2. This is a general purpose, single channel, 8-bit Timer/Counter module with PWM and asynchronous operation. Like Timer/Counter0, this module also has two double buffered Output Compare Registers and the same four modes of operation. For this module, the Timer/Counter can be clocked by an internal synchronous or an external asynchronous clock source. The main features of this module that have not already been mentioned are: frequency generator, 10-bit clock prescaler, overflow and compare match interrupt sources, and allows clocking from an external 32 kHz watch crystal independent of the I/O clock.

The remaining four timer modules are Timer/Counter1, 3, 4, and 5. These modules are 16-bit, which allows for a higher resolution and better precision for timing events. The main features of these modules are: three independent Output Compare Units, double buffered Output Compare Registers, one input capture unit, input capture noise canceler, clear timer on compare match, glitch-free phase correct PWM, variable PWM period, frequency generator, external event counter, and twenty independent interrupt sources. (“ATmega640/1280/1281/2560/2561 Datasheet”).

The ATmega2560 can be found on the Arduino Mega 2560, which operates at 5 volts with a 16 MHz clock speed. This board features 16 analog input pins, four hardware serial pins, an in-circuit serial programming header, and 54 digital I/O pins, 15 of which provide PWM output. The board also features an on-board USB type-B connector, which makes it possible to be connected to a computer and then programmed via the Arduino IDE. The ATmega2560 can be seen on the Arduino Mega 2560 Rev 3 board in Figure 3.16.

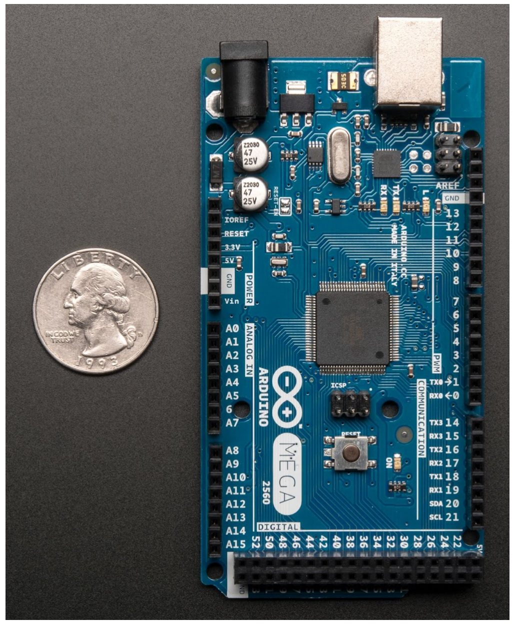

Figure 3.16: Atmega2560 microcontroller on an Arduino Mega 2560 Rev3 ("Arduino Mega 2560 R3 (Atmega2560 - assembled) - Mega!")

The Arduino Integrated Development Environment, or the Arduino IDE, is a cross-platform application that is available for many of the most popular operating systems, such as Windows, Linux, and macOS. An advantage of this IDE is that it uses a variant of the C++ programming language, which many of us are familiar with. Uploading code to a supported board is as simple as selecting the correct board and port from the menu, selecting the compile button, and then selecting the upload button. The Arduino IDE also allows sketches to be managed with more than one file. These files can be normal Arduino code files (.ino extension), C files, C++ files, or header files. It also features many useful tools, such as the serial monitor. The serial monitor displays serial messages sent from the supported board over USB or another serial connector, and also allows data to be sent to the board, which is a great way to receive data for debugging purposes. The Arduino IDE also comes with a number of libraries and supports the ability to download or create custom libraries. Some of the official Arduino libraries include a library for controlling servo motors, libraries for using the SPI, I2C, and UART protocols, libraries for memory management and data storage, libraries for audio sampling and playback, and many more.

The ease of use and the functionality of the Arduino Mega 2560 with the Arduino IDE makes this a great choice for testing before our PCB is finished. Plus, if we were to use the ATmega2560, then everything we developed for testing with the Arduino Mega 2560 would work just as easily with our finalized PCB.

3.3.1.2 Espressif ESP32-WROOM-32 MCU

The ESP32-WROOM-32 is a powerful, generic Wi-Fi + Bluetooth + Bluetooth LE, dual-core 32-bit MCU module that can have 4, 8, or 16 MB of flash memory with 520 KB of SRAM. It can achieve clock speeds of up to 240 MHz at an operation voltage range of 3.0 to 3.6 V. This module also features 26 GPIOs and supports a variety of peripherals, including SD card, Ethernet, UART, SPI, SDIO, I2C, LED PWM, Motor PWM, I2S, IR, pulse counter, GPIO, capacitive touch sensor, ADC, DAC, and TWAI. At the core of this module is the ESP32-D0WD-V3 chip, which is designed to be scalable and adaptive with two CPU cores that can be individually controlled. The chip also has a low-power coprocessor that can be used instead of the CPU to save power while performing tasks that do not require a lot of computing power, such as monitoring peripherals. The module also supports a data rate of up to 150 Mbps and 20 dBm output power at the antenna. (“ESP32-WROOM-32E ESP32-WROOM-32UE Datasheet”)

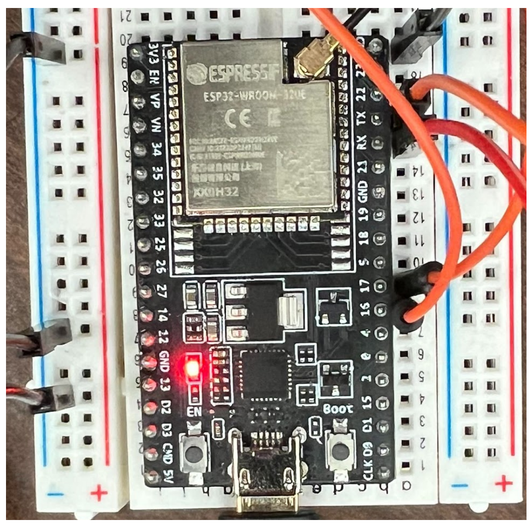

Figure 3.17: ESP32-WROOM-32UE MCU on an ESP32-DevKitC-32UE. Photo taken by Cameron Nichols

3.3.2 Motors for Drivetrain Implementation

In this section, several potential options will be studied to determine which is best for this application. Considering that we are planning on creating two agents with four motors a piece, the cost of each motor is a great deciding factor in this. With the test environment being smaller, we will be able to use slower motors that produce more torque.

3.3.2.1 Greartisan 12V DC Motor

This is a sturdy and durable 12 V DC brushed motor that uses precision winding technology that can produce power on command 30% faster than their competitor. This motor is also rated for a current of 0.6 A. This company has a selection of motors each with different integrated gearboxes. From some basic calculations using an assumed weight experienced per motor and an anticipated coefficient of friction for the vehicle to ground interface and finally choosing an anticipated angular velocity, it was determined that the required torque of each motor would need to be at least 5 kg-cm. The selected motor has a max speed of 30 rpm with a reduction ratio of 1:172 and produces 6 kg-cm of torque. This will provide enough torque to both move the weight of the agent with and without the payload attached to it. With the use of a holonomic mecanum drivetrain each wheel will need to be driven, for four motors at $15 each the total investment in motors comes out to $60 per agent. This comes out to 30% of the budget for the first prototype and 6% of the final allotted budget. Once making the decision to go with a holonomic drivetrain using mecanum drive, the next challenge faced was the significantly greater cost faced by needing two more motors than a nonholonomic drivetrain. These motors do a great job of filling the required design requirements while also being budget friendly.

Figure 3.18: Greartisan 12V DC Motor Photo Taken by John McClain

3.3.2.2 Vexta SMK014A-A

The VEXTS SMK014A-A is a low-speed synchronous AC motor. It can operate at both 50 and 60 Hz with operating currents of 80 and 90 mA, respectively. A disadvantage of this motor is the need for a single-phase 100/115 VAC power source, if this motor is selected this could affect the other components in the control system, we would need to verify that the other components would compile with this power source. This motor having a rated torque of 8.16 kg-cm would produce enough to move the agent and payload together. This is accomplished with a 6:1 speed reducing gear ratio, the rated speeds are 10 and 12 rpm at 50 and 60 Hz respectively. These motors maintain a small form factor with a frame size of 60 mm. While this motor would be effective in accomplishing our design specifications, the price point of these motors comes in at $191 each, which is much more expensive than its competitors.

Figure 3.19: VEXTA SMK216A AC Motor (VEXTA)

3.3.2.3 ElectroCraft RPX52 Brushless Motor

DC Brushless motors can produce torque at a wide range of speeds. The ElectroCraft RPX52 is no exception as it is a “highly dynamic and controllable small frame motor”. The RPX 52 can run at 12, 24, and 48 V and comes with a hall effect sensor. With a stall torque of 7.14 kg-cm it can produce adequate torque while operating at speeds between 0 - 5000 rpm. A potential disadvantage of this is the peak current it draws coming in at 36.8 A, in the scope of this project this would require some major design changes to safely and effectively use this motor. With the frame size coming in at 52 mm this motor fits the size criteria of the project. Finally, possibly the biggest con of this motor is the price tag it comes with, coming in at $562.00, however, this is a versatile motor and highly energy efficient.

Figure 3.20: ElectroCraft Brushless DC Motor (ElectroCraft)

3.3.3 Motor for Payload Interaction Latch

Knowing that one of the design requirements for our project is to move a payload, the team needed to solidify the idea of a latching mechanism. It was decided that a motor was going to be needed to accomplish actuation. However, the sole purpose of this motor would not be to move the latch up and down; With plans to implement something similar to a standard gate latch. This motor purpose was to serve as a means to lock the mechanism in place after a payload had been captured. With this in mind, this motor would need to weigh very little and have a small form factor. This being said DC and AC motors were cut from the list, leaving us with just stepper motors and servo motors. Stepper motors would serve as a great option that would be able to precisely articulate the locking mechanism on demand. Ultimately, it was decided that this kind of precision was not needed especially at the cost that they would come at. Finally, a servo motor was selected for this application; not only would a servo fill the light weight and small form factor, it would also move the mechanism at the required level of precision on command as needed. It was all these factors that made the servo motor the most cost-effective option for manipulating the locking mechanism of our payload interaction unit. The table below shows the analysis of potential motors.

3.3.3.1 SV1260MG-Mini Digital High Voltage Servo

This servo produced by Savox can be run at 6 or 7.4 V with a stall torque of 8.0 kg-cm and 12.0 kg-cm respectively at a stall current of 4000 mA. This servo has a bounded range of travel of 100 degrees, this range of motion will effectively allow the locking mechanism to move as it should. Most importantly its ability to move at higher speeds makes it a great option for our locking mechanism. It has the capability to move at 0.065 sec/60 degrees at 6 V and 0.055 sec/60 degrees at 7.4 V. Finally, the servo's weight is 40.0 g and costs $88.99 each, the cost being its greatest downfall.

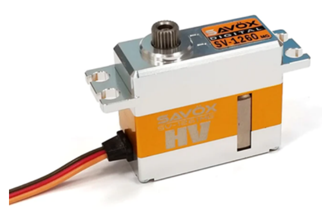

Figure 3.21: Savox SV1260MG- Digital Servo (Savox)

3.3.3.2 LD-20MG Digital Servo

The LD-20MG is another potential option for this application. Unlike its counterparts, it is a high torque, low speed servo, however, this is not a deal breaker. Operating at 7.4 V it produces a stall torque of 20 kg-cm requiring 1 A of current at full load. This servo is much slower than its counterparts with the capability of rotating at 0.20 sec/60 degrees. This is a con as it will increase the interaction time between the agent and the payload. However, this can be justified by its cost at $16.

Figure 3.22: LD-20MG Digital Servo (Robot Shop)

3.3.3.3 JX PS-1171MG Servo

The JX PS-1171MG servo is a cheap and easily accessible servo that is advertised to be utilized in small robotics and RC vehicle applications. At an operating voltage of 4.8 V it has a stall torque of 3.0 kg-cm, it can also be run at 6 V producing a stall torque of 3.5 kg-cm. With the same two operating voltages, its operating speeds are 0.13 sec/60 degrees and 0.11 sec/60 degrees, respectively. Its total range of motion is limited at 120 degrees which will need to be considered when designing our mechanism to transfer the energy from the servo to the pin of the gate latch. This servo uses the standard three wires to operate with the red being positive, the brown being ground, and the orange being the data input.his will be a sound candidate for the proposed modular attachment system. Finally, with the dimensions of the servo coming in at 29.5 x 11.6 x 30.2 cm and a mass of 17.5 g this is a solid option for the proposed applications.

3.3.4 Gate Latch

In this section the selection of the model of the gate latch planned to be used will be explained and compared against other potential candidates.

3.3.4.1 National Hardware N184-861 BPB21

The National Hardware N184-861 BPB21 is a gravity latch also know as an automatic latch it carries a smaller footprint coming in at 305mm x 203mm x 140mm, this latch may be a bit bigger than necessary for the planned prototype, however it will be the perfect size then, moving towards the more finalize form of the agent. It is made from steel that is painted black; this paint will help with fighting corrosion. It is important to understand that many gate latches intended for environments are outside, therefore, in terms of durability and reliability they will almost be over designed in this sense. Usually this comes with a tradeoff with cost, luckily in this case, this latch comes in at an affordable price of $6.00. Making it a potential option for a payload interaction subsystem. While the pin side of the latch carries a small footprint, the style of the latch side brings a much larger footprint. While this latch is advertised as one of the strongest on the market.

Figure 3.23: National Hardware N184-861 BPB21 (National Hardware)

3.3.4.2 Sankins Gravity Gate Latch

The Sankins gravity gate latch is another potential option for a payload interaction subsystem. While the pin side of the latch carries a small footprint, the style of the latch side brings a much larger footprint. While this latch is advertised as one of the strongest on the market, it is important to remember that our project will require much less than that of its intended application, along with the cost of the latch coming in at $20, we decided to continue the search for more latches. Ultimately, its greatest con is the issues that are foreseen when implementing this into our final design.

3.3.5 Motor Controller

Motor controllers make controlling a motor a much simpler task. With the use of H-bridge motor controllers, the polarity of the voltage applied to the motor can be easily switched, meaning that with DC motors, the direction that the motor spins can be changed. As seen in Figure 3.24, an H-bridge typically consists of four switches. For an H-bridge to work properly, switches S1 and S2 should never be closed at the same time, as this would cause a short circuit over the voltage supply which can cause damage. The same also applies to switches S3 and S4. This results in motor controllers typically being designed such that one input will open S1 and close S2 for one state, and do the opposite for the other state, while a second input does the same, but for S3 and S4. Switches that are diagonal to each other are closed to apply a certain polarity across the motor terminals. For instance, if switches S1 and S4 are closed, while the other switches are open, then one voltage is applied across the motor terminals. Likewise, if switches S2 and S3 are closed, while the other switches are open, then the voltage polarity is inverted across the motor terminals.

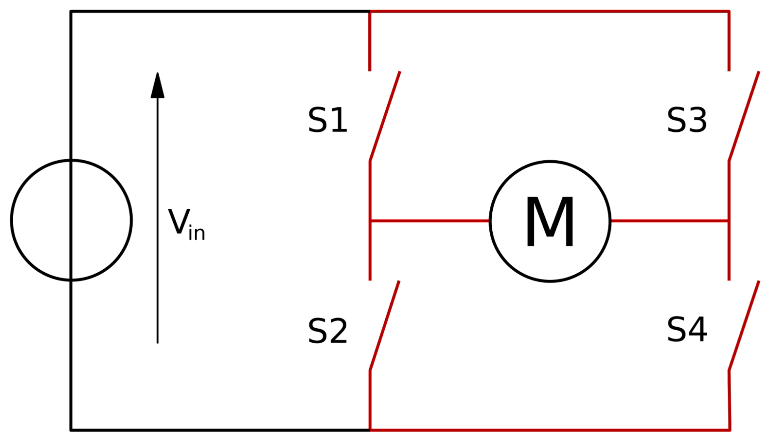

Figure 3.24: Schematic of an H-bridge, where the voltage Vin is supplied to the bridge consisting of switches S1 - S4 and motor M.
Cyril BUTTAY, CC BY-SA 3.0 <http://creativecommons.org/licenses/by-sa/3.0/>, via Wikimedia Commons

Although H-bridges are typically used to change the direction that the DC motor spins, they can also be used to brake the motor or let the motor coast. Braking of the motor is caused by applying a short over the terminals of the motor. This can be accomplished by either closing the upper switches (S1 and S3), or closing the lower switches (S2 and S4), but not both. For the motor to coast, there must be a high impedance between the motor terminals. This can be accomplished by having no more than one switch be closed at a time. Next, we will discuss how to connect a typical motor driver to GPIO and PWM pins to have it operate in three different modes: drive/coast mode, sign-magnitude mode, and lock anti-phase mode (Paradis). More specific details will be provided for each motor driver that we will be discussing later.

The drive/coast mode is the most common connection method for basic usage of H-bridge motor drivers and is what most tutorials will cover. For this mode to work properly, typically one GPIO pin will be used to control the state of one pair of switches (e.g., S1 and S2), another GPIO pin will be used to control the state of the other pair of switches (e.g., S3 and S4), while a third pin capable of supplying a PWM signal will be used to allow/inhibit the two GPIO pins. Inhibiting the two GPIO pins will result in all four switches being open. Do note that while one GPIO pin will control the state of one half of the switches, these switches will always be in opposite states. To drive the motor, the PWM pin must be set to a high state while the two GPIO pins must be set to opposite states. Changing the states of the GPIO pins will change the direction that the motor spins. For the motor to coast, the PWM pin must be set to a low state, which will inhibit the two GPIO pins, in which case the states of the GPIO pins do not matter. For the motor to brake, the PWM pin must be set to a high state while the two GPIO pins are set to the same state. This will cause the motor terminals to be shorted across either the upper switches or across the lower switches, depending on the state of the GPIO pins. It is worth noting that driving the motor using the aforementioned method will drive the motor at a constant speed that is dictated by the supply voltage. However, it is possible to vary the speed at which the motor spins. This works by alternating between driving the motor and coasting, which can be accomplished by varying the duty cycle of the PWM pin. A 50% duty cycle, meaning that the voltage is set to high 50% of the time and set to low the other 50%, results in the average voltage that is being supplied to the motor being equal to 50% of the maximum voltage. However, this does not necessarily equate to 50% speed, since that is dependent on many other factors, one of which is the load connected to the motor.

The sign-magnitude mode, of which the drive/coast mode is a variation of, typically only uses one GPIO pin and one PWM pin per bridge, which is advantageous over the drive/coast mode since one less pin is required. Typically, the GPIO pin is used to control the state of one pair of switches (e.g., S1 and S2), while the PWM pin is used to control the state of the other pair of switches (e.g., S3 and S4). In this mode, to achieve a varying speed, the bridge is alternating between driving the motor in one direction and braking. For a given direction, half of the switches remain in a constant state while the other half is switching states at a modulated rate, where the two switches in either half are always set to opposite states. Specifically, one of these halves is composed of switches S1 and S2, while the other half is composed of switches S3 and S4. To make things clear, in one direction, switches S1 and S2 remain at constant yet opposite states, whereas switches S3 and S4 continually switch states while always remaining in opposite states of each other. For example, in one cycle, S1 is closed while S2 is open, and S3 is closed while S4 is open, which is braking the motor. In the next cycle, S1 is still closed while S2 is still open, but now S3 is open while S4 is closed, which is driving the motor. In this example, the GPIO pin is used to control the direction in which the motor will spin, while the PWM pin will modulate the speed. It is worth noting that coasting is not possible with this configuration since at any point in time, half of the switches are closed while the other half are open, which means the motor is either always being driven or braked. To allow for coasting, an additional GPIO pin would be required to inhibit the other two pins and thus cause all four switches to be open. It is also worth noting that the PWM signal will need to be inverted to achieve the same average voltage in different directions. For instance, if the PWM duty cycle is set to 25% while the GPIO pin is set to a low state, this means that the PWM pin will be set to a high state and thus differ from the GPIO pin 25% of the time, which results in the average voltage being 25% of the supply. On the contrary, if the PWM duty cycle is set to 25% while the GPIO pin is set to high, the states of the two pins now differ 75% of the time, resulting in the average voltage being 75% of the supply. To invert the PWM signal, in this case the 25% duty cycle will need to be changed to 75%, since it is going from being set to high 25% of the time to being set to low 25% of the time.

The lock anti-phase mode typically uses two PWM signals, one of which will control the states of switches S1 and S2, while the other will control the states of S3 and S4. These PWM signals are out of phase with each other. With the use of an inverter, either external or internal, depending on if the controller contains one, then only one PWM pin is needed. This is because the PWM pin will be used to control one half of the switches (e.g., S1 and S2), while also being connected to the input of the inverter. The output of the inverter will then be used to control the other half of the switches (e.g., S3 and S4). Without the use of an inverter, two PWM pins will be needed to supply the two PWM signals. In either configuration, this mode does not require a GPIO pin, which is less than the other two modes. However, if the configuration is used in which two PWM pins are needed, then that is more PWM pins than the other two modes. In an ideal world, a PWM duty cycle of 50% would result in an average voltage of 0 V being supplied to the motor, meaning it is not being driven. This is because since the two PWM signals are opposite, the motor is being driven in one direction for half the time and being driven in the other direction for the other half of the time. The change in direction is happening extremely quickly, which does not give the motor enough time to actually spin. Moving from a PWM duty cycle of 50% towards a duty cycle of 0% results in the motor increasing speed in one direction, whereas going from a duty cycle of 50% towards a duty cycle of 100% results in the motor increasing speed in the opposite direction. That is, a duty cycle of 0% and a duty cycle of 100% are both maximum speed, but in opposite directions. This is because with a duty cycle other than 50%, the motor is being driven in one direction longer than it is being driven in the other direction, which makes the average voltage nonzero. It is worth noting that since the two PWM signals never have the same value, the motor never brakes, at least not by the motor terminals shorting (short brake) like with the coast/drive and sign-magnitude modes. However, we are not in an ideal world and so there might be some variation to what has been stated so far. For instance, even though it is not possible to short brake, dynamic braking can occur.

Dynamic braking occurs when the motor becomes a generator, and can either be rheostatic or regenerative. Rheostatic braking occurs when the generated electrical power is dissipated by heat in resistors (which are typically intentionally used), whereas regenerative braking occurs when the generated electrical power is returned back to the supply. If dynamic braking is not handled appropriately by the system as a whole, damage can occur which could be substantial. Dynamic braking can occur in all the previously mentioned driving modes.

We want our system to be modular and as robust as possible, and thus we would prefer to have more options for driving. For drive/coast and sign-magnitude modes, it is possible to have the motor drive, coast, and brake with the proper configuration. For lock anti-phase mode however, it is not possible to brake the motor without adding a switch that bypasses the motor controller and shorts the motor terminals. For the sign-magnitude mode, since a 50% PWM duty cycle results in 0 V being supplied to the motor while 0% or 100% results in maximum voltage being supplied, the resolution of the applied voltage is essentially cut in half, resulting in less precision. For these reasons, we will be using the drive/coast mode. We will now discuss different options of motor controllers as well as how to control these motor controllers in the three aforementioned modes.

3.3.5.1 STMicroelectronics L298N

The L289N is a dual full-bridge motor driver that is manufactured by STMicroelectronics and is designed to accept standard transistor-transistor logic levels and drive inductive loads, such as DC or stepping motors. This driver has a total of 15 pins, where each bridge has two input pins, two output pins, an enable pin, and a current sensing pin. The remaining three pins are logic supply voltage, supply voltage (for the motors), and ground. The L289N has a maximum constant current of 2 A per channel with a peak current of 3 A per channel. It can also have a maximum voltage of 50 V for the power supply and 7 V for the input, enable, and logic supply. Each of the two bridges contains four NPN bipolar junction transistors. Two of the transistors, which will be referred to as the “upper” transistors, have their collectors tied to the supply voltage, their emitters each tied to a different one of the motor terminals, and each of their bases tied to the output of two different logical AND gates. Both of these logical AND gates have one input connected to the enable pin and the other input connected to one of the input pins. The other two transistors, which will be referred to as the “lower” transistors, have their collectors tied to the emitters of the upper transistors, which are also tied to the motor terminals. The emitters of the lower transistors are tied together and are both connected to a single current sensing pin, while their bases are each tied to the output of two different logical AND gates. These logical AND gates have one input connected to the enable pin and the other input connected to the inverse of one of the input pins. This configuration makes the input pins set the bridge state when the enable pin is high, whereas a low state of the enable pin inhibits the bridge. This means that each bridge can independently control one DC motor, where it can directly control the direction that the motor spins and can indirectly control the speed at which the motor spins via the use of Pulse Width Modulation (PWM). To relate this to Figure 3.24, one of the upper transistors acts as switch S1 while the other upper transistor acts as switch S3. The lower transistor that acts as S2 has its collector connected to the emitter of the transistor that acts as S1. The other lower transistor acts as S4. (“Dual Full-Bridge Driver”)

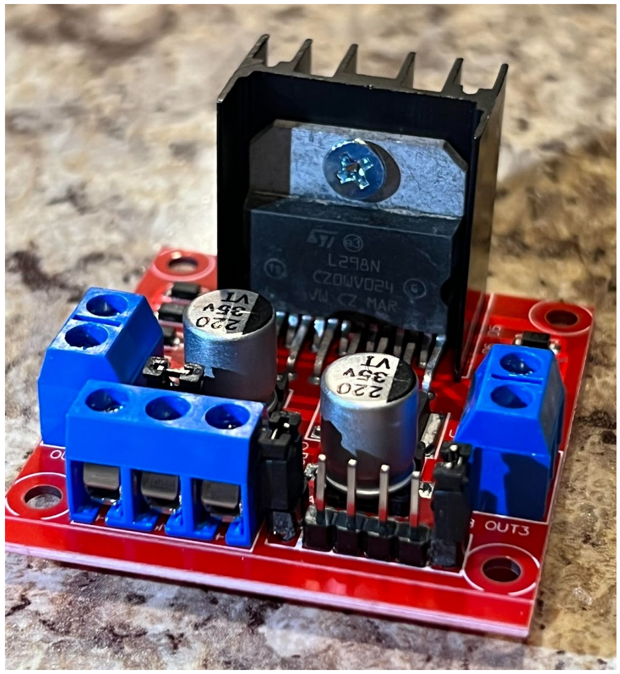

Figure 3.25: STMicroelectronics L298N Motor Driver on a Qunqi Motor Drive Controller Board Module. Photo taken by Cameron Nichols

Next, we will discuss how to connect the L298N motor driver to operate in three modes: drive/coast mode, sign-magnitude mode, and lock anti-phase mode. While discussing these different modes, we will be considering a single bridge as it would be redundant to discuss both bridges since it can be duplicated to also work on the second bridge.

In drive/coast mode, the L298N requires three control pins per bridge to function properly: two GPIO pins and one PWM pin. Each GPIO pin is connected to an input pin and the PWM pin is connected to the enable pin. When the enable pin is set to a low state, the motor will coast. When the enable pin is set to a high state, control of the motor is determined based on the input pins. When the input pins have opposite states, the motor will be driven in a direction that is dictated by specific states of the input pins. When the input pins are set to the same state, the motor will brake.

In sign-magnitude mode, the L298N requires two control pins per bridge to function properly: one GPIO pin and one PWM pin. The GPIO pin is connected to one of the input pins while the PWM pin is connected to the other. The enable pin is tied to a high state, meaning that control of the motor is always based on the input pins. The input pin that is connected to the GPIO pin determines which direction the motor will spin, while the input pin connected to the PWM pin determines the frequency at which the motor switches between driving and braking. To allow for coasting, a third control pin will be required to control the enable pin, which would make this mode and the drive/coast mode practically indifferent.

In lock anti-phase mode, since the L289N does not have an internal inverter, either two PWM pins, or one PWM pin and an external inverter are required. Regardless of which configuration is used, both PWM signals are used to vary the frequency at which the motor switches directions.

3.3.5.2 Toshiba TB6612FNG

The TB6612FNG is a dual full-bridge motor driver that is manufactured by Toshiba and is designed to accept standard transistor-transistor logic levels and drive inductive loads, such as DC or stepping motors, much like the L289N. This driver has a total of 24 pins, which includes the typical inputs and outputs for each channel, supply voltages for motors and logic, PWM inputs for each channel, and a dedicated standby pin. The TB6612FNG has a maximum constant current of 1.2 A per channel, with a peak current of up to 3.2 A per channel. It can also have a maximum voltage of 15 V for the power supply for the motors, and 7 V for the inputs and logic supply. The TB6612FNG utilizes laterally-diffused metal-oxide semiconductors (LDMOS) transistors, which have a noticeably lower voltage drop when compared to bipolar junction transistors. When the output current is 1 A, then the typical voltage drop experienced is 0.5 V with a maximum voltage drop of 0.7 V. When the output current is 0.3 A, then the typical voltage drop is 0.15 V with a maximum voltage drop of 0.21 V. When in standby mode, there is a maximum supply current of up to 2.2 mA and a maximum input current of only 25 µA. Each channel can control one DC motor, where it can directly control the direction that the motor spins and the speed at which the motor spins via the use of PWM. Table 3.7 shows the relationship between the output and input of each channel, where CW/CCW is in reference to the direction in which the motor would spin and stands for clockwise/counterclockwise, and 1/0 is in reference to the logic level of the pin and represents on/off, X means that the logic level does not matter, and H/L is used to represent if the voltage is high or low. (“TB6612FNG”)

Table 3.7: Control Function of TB6612FNG

Input				Output		
IN1	**IN2**	**PWM**	**Standby**	**OUT1**	**OUT2**	**Mode**
1	1	X	1	L	L	Short Brake
0	1	1	1	L	H	CCW
		0	1	L	L	Short Brake
1	0	1	1	H	L	CW
		0	1	L	L	Short Brake

Input				Output		
IN1	**IN2**	**PWM**	**Standby**	**OUT1**	**OUT2**	**Mode**
0	0	1	1	OFF (High Impedance)		Stop
X	X	X	0	OFF (High Impedance)		Standby

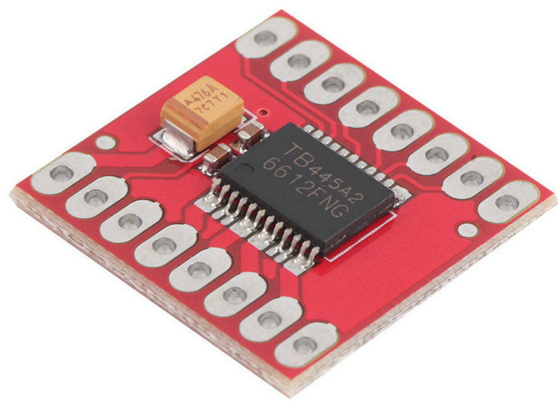

Figure 3.26: Toshiba TB6612FNG Motor Driver on a Development Board (Bullas)

Next, we will discuss how to connect the TB6612FNG motor driver to operate in three modes: drive/coast mode, sign-magnitude mode, and lock anti-phase mode. While discussing these different modes, we will be considering a single channel or bridge, as it would be redundant to discuss both channels since it can be duplicated to also work on the second one. There is more than one way to connect this motor driver to operate in each of the different modes, but we will be describing connection methods that result in the least number of used pins. The PWM pin present on the TB6612FNG will be referred to as the "PWM port" to avoid confusion with the PWM pin from the microcontroller.

In drive/coast mode, the TB6612FNG requires a minimum of three control pins per bridge to function properly: two GPIO pins and one PWM pin. In this minimum configuration, each GPIO pin is connected to an input pin, the PWM pin is connected to the standby pin, while the PWM port is tied to a high state. When the standby pin is set to a low state, the motor experiences high impedance and will thus coast. When the standby pin is set to a high state, control of the motor is determined based on the input pins. When the input pins have opposite states, the motor will be driven in a direction that is dictated by specific states of the input pins. When the input pins are both set to a low state, the motor will coast. Both input pins can be set to a high state to allow for braking.

In sign-magnitude mode, the TB6612FNG requires a minimum of two control pins to function properly: two PWM pins. In this minimum configuration, each PWM pin is connected to a different input pin. The PWM port and standby pin are both tied to a high state. Each input pin will control either the direction that the motor spins or the frequency at which the motor switches between driving and braking. This control scheme will be swapped between the two pins for different directions. For example, to spin the motor clockwise, input pin 1 should be set to a high state, while input pin 2 will control the speed by switching states. To spin the motor counterclockwise, the opposite is true: input pin 2 should be set to a high state, while input pin 1 will control the speed by switching states. This is because in this minimum configuration, the only option for braking is achieved when both input pins are set to a high state, while driving the motor is achieved by the input pins having opposite states. This results in one input pin remaining at a high state for one direction while the other input pin switches states. Coasting can also be achieved in this minimum configuration by setting both inputs to a low state.

In lock anti-phase mode, the TB6612FNG requires a minimum of two control pins to function properly: two PWM pins. In this minimum configuration, the connections are the same as it is for the minimum configuration of the sign-magnitude mode. However, there is still a difference in how the two modes work. In this mode, the two PWM signals are always opposite and are used to vary the frequency at which the motor switches directions. It is also possible to set both inputs to a high state or both to a low state to allow for braking or coasting, respectively.

3.3.5.3 Texas Instruments DRV8876

The DRV8876 is an H-bridge motor driver that is manufactured by Texas Instruments and is designed to accept standard transistor-transistor logic levels and drive inductive loads, such as DC motors. This driver has a total of 16 pins, which includes the typical enable, input, and output pins, as well as supply voltages for motors, current sensing, sleep mode, and more. The input pins have multiple purposes, where input 1 can function as the enable pin, while input two can function as the phase pin, depending on the selected mode. The DRV8876 has a maximum peak current of 3.5 A per output, with a maximum voltage of 40 V for the power supply for the motors, and 5.75 V for the logic pins. The DRV8876 utilizes N-channel metal-oxide-semiconductor field-effect transistors (MOSFETs) for the switching capabilities, which have a voltage drop of around 0.5 V for our application, which is a noticeably lower voltage drop when compared to bipolar junction transistors. When in sleep mode, there is a maximum supply current draw of 1 μA, and while not in sleep mode, there is a maximum input logic current of 75 μA. This driver can control one DC motor, where it can control the direction that the motor spins and the speed at which the motor spins via the use of PWM. This driver provides three control modes to support different control schemes: PH/EN mode, PWM mode, and independent half-bridge mode ("DRV8876 H-Bridge Motor Driver with Integrated Current Sense and Regulation"). The control mode is selected through the PMODE pin, which is described in Table 3.8.

Table 3.8: PMODE Functions

PMODE State	Control Mode
Logic Low	PH/EN
Logic High	PWM
High Impedance	Independent Half-Bridge

When the PMODE pin is set to a low state on power up, the driver is latched into the PH/EN mode. This mode allows for the H-bridge to be controlled with a speed and direction type of interface, which can be seen in Table 3.9.

Table 3.9: PH/EN Control Mode

nSLEEP	EN	PH	OUT1	OUT2	Description
0	X	X	High Z	High Z	Sleep
1	0	X	L	L	Brake (Low-Side Slow Decay)
1	1	0	L	H	Reverse
1	1	1	H	L	Forward

When the PMODE pin is set to a high state on power up, the device is latched into the PWM mode. This mode allows for the H-bridge to enter the high impedance (or high Z) state without setting the sleep pin to a low state. Table 3.10 shows the truth table for this mode.

Table 3.10: PWM Control Mode

nSLEEP	IN1	IN2	OUT1	OUT2	Description
0	X	X	High Z	High Z	Sleep
1	0	0	High Z	High Z	Coast
1	0	1	L	H	Reverse
1	1	0	H	L	Forward
1	1	1	L	L	Brake (Low-Side Slow Decay)

When the PMODE pin has high impedance on power up, the device is latched into the independent half-bridge control mode. This mode allows for each half-bridge to be directly controlled to support high-side slow decay or driving two independent loads. Table 3.11 shows the truth table for this mode.

Table 3.11: Independent Half-Bridge Control Mode

nSLEEP	INx	OUTx	Description
0	X	High Z	Sleep
1	0	L	OUTx Low-Side On
1	1	H	OUTx High-Side On

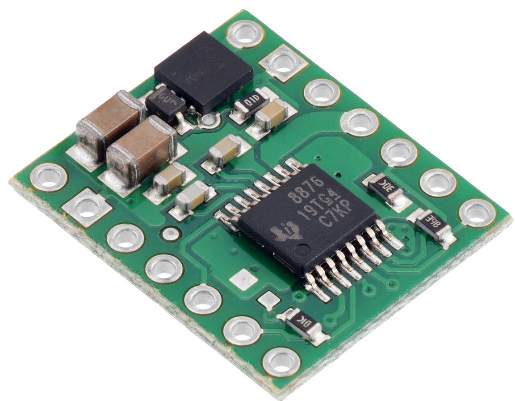

Figure 3.27: Texas Instruments DRV8876 Motor Driver on a Breakout Board ("DRV8876 Single Brushed DC Motor Driver Carrier")

Next, we will discuss how to connect the DRV8876 motor driver to operate in three modes: drive/coast mode, sign-magnitude mode, and lock anti-phase mode. Due to the multiple control modes, there will be multiple ways to connect this motor driver to operate in each of the different drive modes, but we will be describing connection methods that result in the least number of used pins.

In drive/coast mode, the DRV8876 requires a minimum of two control pins to function properly: two PWM pins. In this minimum configuration, one PWM pin is connected to the EN/IN1 pin, the other is connected to the PH/IN2 pin, and both the sleep pin and the PMODE pin are tied to a high state, which has the device operate in the PWM control mode. To alternate between driving and coasting, one input pin should be set to a low state while the other utilizes a PWM signal to control the speed. The input pin that remains at a low state will determine what direction the motor will spin. It is also possible to brake in this mode by setting both inputs to a high state. An additional pin could be connected to the sleep pin to allow the device to enter sleep mode.

In sign-magnitude mode, the DRV8876 requires a minimum of two control pins to function properly: one GPIO pin and one PWM pin. In this minimum configuration, the PWM pin is connected to the EN/IN1 pin, the GPIO pin is connected to the PH/IN2 pin, the sleep pin is tied to a high state, and the PMODE pin is tied to a low state, which has the device operate in the PH/EN control mode. The PH/IN2 pin determines the direction that the motor will spin, while the EN/IN1 pin sets the frequency at which the motor alternates between driving and braking. An additional pin could be connected to the sleep pin to allow the device to enter sleep mode.

In lock anti-phase mode, the DRV8876 requires a minimum of two control pins to function correctly: two PWM pins. In this minimum configuration, one PWM pin is connected to the EN/IN1 pin, the other is connected to the PH/IN2 pin, and the sleep pin is tied to a high state. The device can either operate in the PWM control mode or the independent half-bridge control mode, which means the PMODE pin can either be tied to a high state or be disconnected, respectively. In either control mode, driving the motor requires the two input pins to remain in opposite states, while switching states at a frequency to control how often the motor switches directions. In the PWM control mode, setting both input pins to a low state results in coasting, while setting both input pins to a high state results in braking. In the independent half-bridge control mode, setting both input pins to a low state results in coasting. In either control mode, an additional pin could be connected to the sleep pin to allow the device to enter sleep mode.

3.3.6 Inertial Measurement Unit

This section goes over how the decision was made on what inertial measurement unit, or IMU, would be used. When choosing the IMU, our team primarily focused on ensuring versatility of available readings, and so we decided to use an IMU that encompasses an accelerometer, a gyroscope, and a magnetometer. Utilizing three axes for each of these devices results in the IMU having 9 degrees of freedom. The accelerometer is used for measuring acceleration, but when this acceleration is integrated velocity can be measured, and when integrated again displacement can be measured. The gyroscope is used to measure angular velocity, which when integrated, provides angular displacement. The magnetometer is used to measure a magnetic field, which can be utilized to act as a compass to provide a heading reference. It is worth noting, and can have a significant impact, that these different devices can have errors. For the accelerometer and gyroscope, they suffer from accumulated error, which can have an exponential error growth in some cases. For instance, with the accelerometer, a constant error in acceleration results in a linear error growth in velocity and a quadratic error growth in displacement. For the magnetometer, the magnetic field of the Earth is not the only field that is being measured. Nearby permanent magnets will introduce a magnetic field that can affect readings, and even certain materials can deflect or warp magnetic fields, which can affect readings if they are in a close enough proximity to the magnetometer. Due to how big of an impact these errors can have on the accuracy of the IMU, a low error rate is a heavily weighted criterion in our decision of the IMU that we will use. We looked primarily at two IMUs to choose from: the MPU-9255 MotionTracking device from InvenSense, and the MinIMU-9 v5 from Pololu. Through the combined experience of Hive: The Grounded Swarm members from both the software and hardware teams, we have firsthand experience with using both of these IMUs.

3.3.6.1 InvenSense MPU-9255

The InvenSense MPU-9255 is a multi-chip module consisting of two dies, one of which contains the 3-axis accelerometer and the 3-axis gyroscope, while the other contains the 3-axis magnetometer. Thus, the MPU-9255 is a 9-axis IMU. This IMU features I2C and SPI serial interfaces, has an operation voltage range of 2.4 to 3.6 V, a separate digital IO supply range from 1.71 V up to the operating voltage, and has a typical supply current draw of 3.7 mA when all 9 axes are enabled. The MPU-9255 is also designed to interface with additional sensors on its auxiliary I2C bus. Some relative features of the integrated gyroscope, accelerometer, and magnetometer can be seen below in Table 3.12.

Table 3.12: Relevant Features of Integrated Devices of MPU-9255 at 25°C

	Accelerometer	Gyroscope	Magnetometer
Axes	3	3	3
ADC	3 16-bit	3 16-bit	3 16-bit
Range	±2, ±4, ±8, or ±16 g	±250, ±500, ±1000, or ±2000°/sec	±49 G
Normal Operating Current	450 μA	3.2 mA	280 μA
Self-Test?	Yes	Yes	Yes
Noise Density	300 μg/$\sqrt{Hz}$	0.01°/sec/$\sqrt{Hz}$	480 nT/$\sqrt{Hz}$
Initial Tolerance	Zero-g X, Y: ±60 mg Z: ±80 mg	ZRO ±5°/sec	Zero-G ±3 G

Apart from the three integrated devices listed above, the MPU-9255 also includes some additional features. This IMU has a 512 byte first in, first out (FIFO) buffer, an embedded temperature sensor, a precision clock with 1% drift from -40 to 85°C, programmable interrupts, and is 10,000 g shock tolerant.

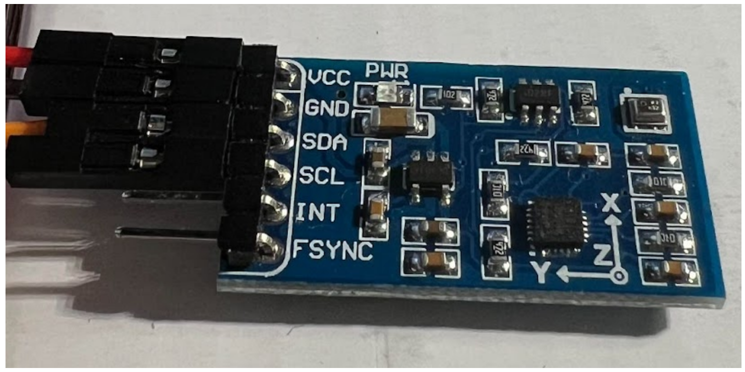

Figure 3.28: InvenSense MPU-9255 on a 10 DOF IMU Sensor. Photo taken by Cameron Nichols

3.3.6.2 Pololu MinIMU-9 v5

The Pololu MinIMU-9 v5 is a board that combines STMicroelectronics' LSM6DS33 3-axis gyroscope and 3-axis accelerometer with STMicroelectronics' LIS3MDL 3-axis magnetometer to form a 9-axis IMU. Some relative features of the integrated gyroscope, accelerometer, and magnetometer can be seen below in Table 3.13.

Table 3.13: Relevant Features of Integrated Devices of Pololu MinIMU-9 v5 at 25°C

	Accelerometer	Gyroscope	Magnetometer
Axes	3	3	3
ADC	3 16-bit	3 16-bit	3 16-bit
Range	±2, ±4, ±8, or ±16 g	±125, ±245, ±500, ±1000, or ±2000°/sec	±4, ±8, ±12, or ±16 G
Normal Operating Current	70 µA	0.9 mA	270 µA
Self-Test?	Yes	Yes	Yes
Noise Density	90 µg/$\sqrt{Hz}$	0.007°/sec/$\sqrt{Hz}$	400 nT/$\sqrt{Hz}$
Initial Tolerance	Zero-g ±40 mg	ZRO ±10°/sec	Zero-G ±1 G

The carrier board includes a voltage regulator that allows the module to be powered by voltages from 2.5 to 5.5 V, and the regulator output can supply almost 150 mA to external devices. Each of the three sensors have their clock and data lines tied together and can be interfaced over the I2C bus that the carrier board offers.

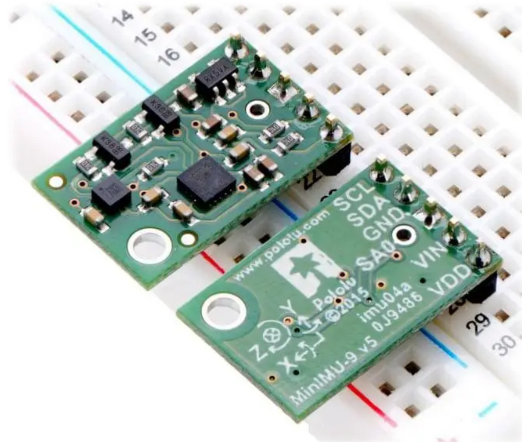

Figure 3.29: Two Pololu MinIMU-9 v5 modules in a breadboard. ("MinIMU-9 V5 - IMU Digital Combo Board - 9 Degrees of Freedom")

3.3.7 Battery Module

The battery module design has a strong effect on our agent's system's overall performance. The ability of our agents to operate efficiently and autonomously for extended periods of time is almost directly based upon the energy and power capacity of the battery module. But the power source for Hive is not just about finding a power source that meets the simple requirements of voltage and capacity. It also involves considering important factors such as size, weight, durability, user convenience, power delivery and ease of recharging. The ability of our agents to operate efficiently and autonomously for extended periods of time is based upon the energy and power capacity of the battery module. As such the mobile power design is a crucial part of our robotic project, as it provides the necessary power to enable the mobility and functionality of the robot.

In this section, we will delve into the design considerations, thought-processes, difficulties, and solutions that we came across as we work to develop a mobile power system that will provide the necessary power to our robot while remaining flexible enough to meet the needs of a wide range of potential users.

3.3.7.1 Requirements for Battery Module Design

Requirements: The design of the battery module for our modular robotic project was funneled by a number of high-level requirements that were determined during the preliminary planning stages of the project. These requirements were carefully considered and made to ensure that the battery solution would be able to effectively and efficiently power the robot during peak power and energy usage. The following requirements were identified.

*Power Output**:*** The battery module must be able to provide sufficient power to drive the various motors and actuators of the robot, to be able to perform the Agent’s intended tasks without any power restrictions. The power output of the battery module was determined based on the continuous power requirements of the individual components of the robot, any power spike requirements during operation, as well as any predicted loads from any future modules which may be added onto the agent for additional functionality. An additional consideration is to confirm that the battery will not be damaged by the expected constant power consumption.

Capacity: The battery module must have sufficient capacity to provide the desired operational runtime of 30 minutes of constant usage. The operational runtime was determined based on the specific tasks and anticipated usage scenarios of the robot, as well as many safety margins to ensure that the robot can complete its tasks without prematurely running out of power and being able to reach the recharging station.

Voltage: The battery module must provide the appropriate voltage levels to meet the requirements of the robot's electrical system. This includes providing the correct voltage levels for the motors, sensors, and other electronic components of the robot including future modular additions. It also includes being compatible with the voltage regulation or conversion necessary within the power distribution to ensure stable and reliable operation.

Safety: The battery module must include many safety features to protect against overcharging, over-discharging, short-circuiting, and other potential hazards. This includes using high-quality battery cells and using well tested and documented protection circuits. This also includes implementing any additional safety measures such as voltage monitoring, current limiting, and some user interface to prevent any safety risks associated with the battery. These safety regulations may change depending on the type of battery chemistry used.

Size and Weight: The size and weight of the battery module must be carefully considered to ensure that it can not only fit within the volumetric space permitted on the agent but also without compromising its mobility or performance. The battery module should be as compact and lightweight as permissible, while still providing the necessary power and capacity for the robot's operation without damaging the battery's cycle life.

Rechargeability: The battery module must be rechargeable to allow for easy replenishment of power during operation. The rechargeability of the battery was considered in terms of convenience and performance. Including ease of recharging, speed of recharging, and the ability to withstand multiple charging cycles without significant degradation in performance.

Conclusion: In conclusion, the battery module design is directed by the requirements of power output, capacity, voltage, safety, size and weight, durability, and rechargeability. These requirements were carefully considered to ensure that the following designed battery solution would be able to effectively and reliably power our agents and enable their optimal performance. In the next section, we will dive into the specific details to further explain the requirements for the development of the battery module.

3.3.7.2 Detailed Power Requirements

In Section 3.3.7.1 we detailed the high-level requirements for our Power Delivery for the battery module. In this section we will continue that process by developing the detailed power requirements for our battery module. First, power delivery is specifically detailed by peak power delivery. This information needs to be carefully considered because if our battery module is unable to deliver the spikes of current necessary during peak load then our agents may abruptly fail and shut down because of the lack of power. Over demanding of peak power from any battery also has the effects of damaging the battery cycle life because of sped up chemical deterioration and wasting a substantial amount of energy through heat within the battery. As such, we will now go into the details of what is the peak power that we need to deliver to our robots.

To start this analysis it is important to understand what is being powered. The most ‘power hungry’ device on our agents are the driving motors. The four driving motors will be responsible for not only driving the agents but also pulling the payload. This will involve a lot of energy draw and during initial movement when the agent needs to overcome static friction, the peak power draw will be at its highest. The peak power draw of the motors is the stall current, which is 0.69 A per motor. As such, there will be a total of 2.76 A of peak power draw going to the motors. But this current is not equivalent to the power that the battery needs to deliver. To convert this motor power to battery power we need to convert to watts.

$$Watts = Volts \times Amps$$
$$12\,V \times 2.76\,A = 33.12\,W_{MotorPeak}$$

The remaining power consumption devices are the servo motors and the microcontroller. We plan on using a single servo motor, the LD-20MG. The servo motor when used at full power is expected to pull a maximum of 3 A. This equivalates to 15 W when run at 5 V. Lastly is the microcontroller, which is used to run the brains and logic control of the agent and also to power the communication device. The microcontroller is expected to demand a peak power draw of 0.5 A at 9 V. Which equates to 4.5 W.

To find the total peak power that the battery module needs to deliver we must consider the condition of when all devices are pulling their maximum peak power draw at the same time to ensure that our battery can handle the worst condition. As such, adding the peak power draw values together we get the following equation:

$$33.12\ W_{Motors} + 15\ W_{Servo} + 4.5\ W_{MC} = 52.67\ _{Peak}$$

52.67 W of peak power is one of the detailed requirements of which we will use to filter our battery cell options through.

3.3.7.3 Detailed Capacity Requirements

In Section 2.2 we identified that the high level requirement for our battery module that our agents will house needs to provide a runtime of at least 30 minutes. The following section will describe the mathematical process of converting this high-level requirement to the detailed requirements to ensure that the batteries that we select will fulfill this high-level requirement.

To deduce what battery module capacity we need, we first need to understand what our load profile is going to be throughout the 30 minute runtime. The major contributor to the agent's power demand is the motors. Next would be the servo motors and the microcontroller. So first we will deduce the expected load profile of the motors, then the servos, and the microcontroller last. The two possible scenarios are shown below in Figure 3.30.

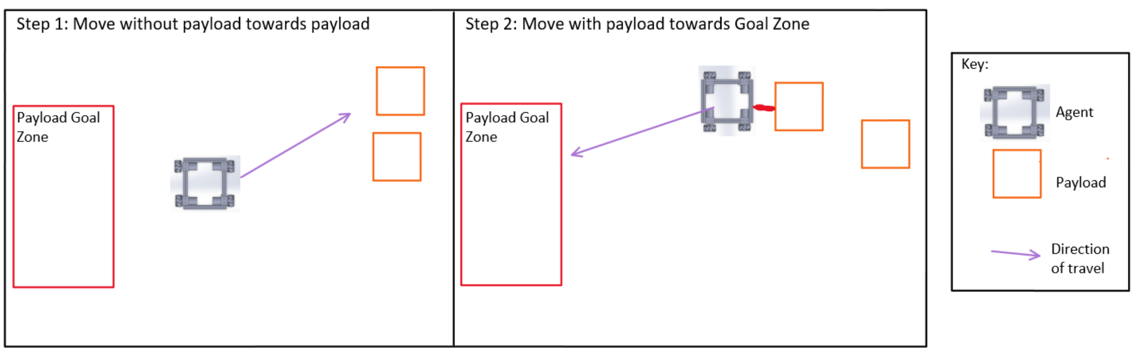

Figure 3.30: Visual demonstration of the two load states. Designed by Isaac Finley

3.3.7.3.1 Motor Energy Demand

The first load profile to consider is the motors. As shown in Figure 3.30, we expect the agent to be driving towards the individual payloads 50% of the time and driving with the payload to their destination the other 50% of the time. Following this load profile we can produce some further deductions as to the necessary energy draw. For the first 50%, because the agent will be driving without the payload, the motors will not be driving at their maximum power draw. We expect the motors to be working at 30% capacity, but to have healthy margins for future module attachments, we expect the motors to be driving at 50% with no payload attached. For the second half of the load profile, we expect the motors to be driving at 80% capacity to pull the payload, but to add safety margins we will expect the motors to be working at 100% capacity when pulling the payload. Following this mindset, each individual motor when at 100% load draws 0.69 A and 12 V:

Watts per motor:

$0.69\ A\ \times 12\ V\ =\ 8.28\ W_{per\ motor}$

So for 15 minutes we will expect 33.12 W of power draw from the motors. For the second half, if we expect the motors to be running at 50%, then for 15 minutes the device will be running at 16.56 W for 15 minutes. Converting these values to a energy over time:

50% motor load over a 15 minute period:

$16.56\ W_{50\%\ motor\ load} \times 15\ minutes \times (\frac{1\ Hour}{60\ minutes})\ =\ 4.14\ W\ *\ Hrs_{15\ min\ -\ motor\ load}$

100% motor load over a 15 minute period:

$33.12\ W_{100\%\ motor\ load} \times 15\ minutes \times (\frac{1\ Hour}{60\ minutes})\ =\ 8.28\ W\ *\ Hrs_{15\ min\ -\ motor\ load}$

Total motor energy demand over 30 minutes:

$4.14\ W\ *\ Hrs_{15\ min\ -\ motor\ load} + 8.28\ W\ *\ Hrs_{15\ min\ -\ motor\ load}\ =\ 12.4\ W\ *\ Hrs_{Motor\ Load}$

So, to have a runtime of at least 30 minutes, the motors will require a battery capable of delivering 12.4 W/Hrs.

3.3.7.3.2 Servo Energy Demand

Next we will find the load profile of the servo motor. The servo motor is expected to be used twice per payload. If we expect to deliver 5 payloads maximum within 30 minutes, then the servo will need to be used 10 times total. For every use, the servo motor will be on for 5 seconds, giving us a total of 50 seconds of servo use. If the maximum power load from the servo is 3 Amps at 5 Volts, then we can expect the servo motor to use 15 W over a 50 second period.

Servo energy demand:

$15\ W_{Servo} \times (50\ seconds)\ \times (\frac{1\ minute}{60\ seconds}) \times (\frac{1\ hour}{60\ minutes})\ =\ 0.208\ W\ *\ Hrs_{Servo}$

We can safely deduce that the servo motor will require 0.208 W/Hrs from the battery module.

3.3.7.3.3 Microcontroller Energy Demand

Lastly is the microcontroller. The load profile of the microcontroller is expected to be used through the entire runtime operation of the agent. As such, if the microcontroller at maximum will consume 0.5 A at 9 V:

Microcontroller energy demand:

$(0.5\ A\ \times 9\ V)\ \times (30\ minutes) \times (\frac{1\ hour}{60\ minutes})\ = 2.25W\ *\ Hrs_{MC}$

As shown the microcontroller will consume 2.25 W/Hrs of energy from the battery module.

3.3.7.3.4 Total Energy Consumption

To find the total energy consumption that is necessary to have a runtime of at least 30 minutes all energy consumptions need to be added together.

Total Energy Consumption:

$$12.4\,W * Hrs_{Motor\ Load} + 0.208\,W * Hrs_{Servo} + 2.25W * Hrs_{MC} = 14.86\,W * Hrs$$

Our battery needs to store at least **14.86 W/Hrs** of energy to be able to have a runtime of 30 minutes. This is the next detailed requirement that our battery cell selection will be filtered by.

3.3.7.4 Detailed Voltage Requirements

In this section we will go over the detailed voltage requirements of the battery module. This includes the preferred voltage levels that our battery module needs to provide for easy usage within the Agent. Similar to previous sections, to start this analysis we must first look at the loads that the battery is supplying energy to. This includes the motors, servos, microcontrollers, and sensors. These devices will require, in order, 12 V, 5 - 7 V, 5 V, and 3.3 V. While these voltages will not be supplied directly from the battery but rather through a set of voltage converters, the battery must be designed to conveniently mesh with the designed voltage converters.

The first decision to make is whether the voltage converters will be buck converters or boost converters. Boost voltage converters are simply stated as voltage boosters, those devices raise the voltage of the input through various means, Including: switched capacitor, voltage multiplier, multistage, magnetic coupling, and switched inductor (also known as voltage lifting). Buck Voltage converters will lower the voltage through methods such as linear regulators and switched-mode power supply.

The main thing to consider when choosing a voltage converter is that their wattage will not change from input to output. There will be no additional energy being supplied out of the voltage converter, just changed voltages. For example, if we supply 6 V at 1 A to the input, when the output is boosted to 12 V, using $Watts = Volts \times Current$: $6\,V \times 1\,A = 6\,W$, the output can only output 6 W. Which is $6\,W\,/\,12\,V = 0.5\,A$. As shown, the output is now only 0.5 A at 12 V. And the opposite is true when the voltage is bucked to a lower voltage, increasing current delivery. For our design, we opted for the easier to design and more abundant selection of buck converters.

As such we need to make sure that our battery module has a function voltage usable by our buck converters. For buck converters our criteria is to drop the voltage to the highest voltage used in our system. If our motors use a 12 V supply, then our battery module at its lowest charge needs to be able to power a voltage buck converter to 12 V. This means that our specific voltage requirement is to have our battery module at its lowest charge, be above 12 V. This will dictate not only which battery to use, given that different battery cell chemistries will supply different nominal voltage levels, but will also dictate how many cells will be connected in series to increase the battery module voltage. Once again, our specific voltage requirement is for the battery module to be **above 12 V.**

3.3.7.5 Detailed Size and Weight Requirements

The robot size is 10 in wide by 10 in long by approximately 15 in tall. and we have designated an 8.5 in by 8.5 in by 8.5 in space within the agent to house the battery. As such the battery needs to be within those volumetric requirements to fit onto the agent.

For the weight requirements, the wheel strength is the crutch for this system. The wheels are expected to be able to handle 25 lbs of weight per wheel. If the agent's electronics and structural frame is approximately 15 lbs of weight, then according to the structural integrity of the wheels:

Wheel Structural integrity limit:

$$4_{Wheels} \times 25\ lbs_{per\ wheel} = 100\ lbs_{Agent\ weight\ limit}$$

Battery weight constraint:

$$100\ lbs_{Agent\ weight\ limit} - 15\ lbs_{weight\ before\ battery} - 20\ lbs_{payload\ weight} = 65\ lbs_{Battery\ Constraint}$$

As shown above the battery must be lighter than 65 lbs according to the structural integrity of the wheels. This weight constraint is substantially more than what we expect the battery module to be. This is because the average car battery is 30 to 60 lbs, which is viewed as not energy dense. And we plan on using newer technology which has great energy density and will be much lighter than a car battery.

After observing these expected limitations, the battery needs to fulfill the volumetric and weight requirements which we found to be an **8.5 in cubed box** and a weight constraint of **less than 65 lbs**.

3.3.7.6 Battery Module Requirement Scoring

In this section we will discuss the scoring ratios that go towards each detailed requirement that was discussed above. For review, the detailed requirements that were discussed are listed below in short form.

Table 3.14: Battery Module Selection Requirements

Battery Module Selection Requirements					
52.67 W of peak power	14.86 W*Hours of capacity	At least 12 V when put in series at 10% capacity	Fit within an 8.5 in cubed box	Weight less than 65 lbs	Cost

Each of these requirements are vital to the performance of the agents. Out of the 5 ratings, the peak power rating and low voltage rating are of the least importance. This is because if we are unable to provide a peak power wattage of 52.67 W, then we will employ techniques from Capacitech Energy to assist in this peak power rating. As such it is not as important for our battery module to provide this much peak power. As for the low voltage rating, if a singular battery cell is unable to meet the 10% 12 V rating, then we will employ a common technique of connecting cells in series to increase the battery module's voltage. These two requirements will be rated out of a 10 point scale each.

The last four requirements of Capacity, Volume, Weight, and cost are of utmost importance. This is because if a single rating is not met, then our project will not come to fruition. If we do not design for 14.86 W/Hrs of capacity, then our agent cannot have a 30 minute operational time. If the battery module does not fit within the volumetric constraint, then our agent cannot house the battery within its capabilities. If our battery is too heavy, then we may have catastrophic failure in wheel integrity and be unable to move our 20 lb payload goal. And lastly, as we are very budget minded in this project, if we can afford the battery module can be the determining factor on which module we purchase. As such, these last four topics of capacity, volume, weight, and cost are each equally important and are each rated to 20 points. The following battery selection will be filtered through and compared to one another based upon these requirements.

Table 3.15: Battery Module Selection Requirements Scaling

Battery Module Selection Requirements					
52.67 W of peak power	14.86 W*Hours of capacity	At least 12 V when put in series at 10% capacity	Fit within an 8.5 in cubed box	Weight less than 65 lbs	Cost
10 points	20 points	10 points	20 points	20 points	20 points

3.3.7.7 Battery Selection

In this section we will discuss possible battery cell selections and their respective battery Module design counterparts. Then once designed into a battery module, these designs will be scored based upon the requirements they must fulfill and a best option will be selected at the end of this section based upon this scoring.

ANR26650M1-B

This battery cell is of the LiFePO4 chemistry and boasts of having 2.4 Ah of capacity at a 0.5C with a maximum discharge of 120 A pulse. Because this cell is a LiFePO4, at 10% capacity this cell is at 3 V. As such the battery module design using this battery to get to 12 V will need 4 cells in series. If we use 4 cells in series, using their nominal voltage of 3.3 V, the battery module voltage is 13.2 V and their capacity is 2.4 Ah, we can get a total of 31.68 Wh, per series string of this module. Additionally, if we have a 120 Pulse A rating, we get a resulting 2,610 W of peak power. Each cell is shaped like a cylindrical and has a diameter of 26 mm by a height of 65 mm. As such the battery module with 4 of these cells occupies a space of 2.04 In wide, 2.04 in deep, and 2.56 in tall. Next is weight, the cell weighs 76g, so the battery module will weigh 0.67 lbs. Fitting within all requirements. The only downfall being their price per cell of $6.53 per cell, with a minimum quantity purchase of 5 cells equalling $32.65 per battery module.

Samsung 30Q 18650

This battery cell is of the Lithium Ion chemistry and has the ratings of 3 Ah, 10 A continuous, 15 A pulse, 3.6 V nominal, 3.0 V cutoff, 48g, and 18.33 mm diameter by 64.85 mm in height. Using these ratings, the battery module will need to be 4 series cells, for a 12 V at 10% battery module, also giving a nominal module voltage of 14.4V. The peak power of the module would be 216 W and the module capacity is 43.2 W*Hrs. The Module volume in inches cubed is 1.44 in wide, 1.44 in deep, and 2.51 in height. The biggest downfall is the requirement of purchasing a minimum quantity of 10 cells at $5.50 per cell.

Samsung 25R

This battery cell is also of the Lithium Ion chemistry and has the ratings of 2.5 Ah, 10 A continuous, 20 A pulse, 3.6 V nominal, 3.0 V cutoff, 43.04 g, 18.33 mm diameter by 64.85 mm in height. Using these ratings we get similar results to the previous batteries with a battery module requiring 4 cells in series and having a module pack nominal voltage of 14.4 V. The peak power of the module would be 288 W and the module capacity is 36 W*Hrs. The module volume in inches cubes is the same as the Samsung 30Q with 1.44 in wide, 1.44 in deep, and 2.51 in height. The biggest downfall once again being in price with a minimum purchase requirement of 10 cells at $3.85 per cell. The battery module price would be $38.50.

Table 3.16: Battery Selection

Battery Name	Module Peak Power (Watts)	Module Capacity (Watt*Hrs)	Module Voltage at 10% capacity	Module Volume (inches cubed)	Module Weight (Lbs)	Battery Module Cost	Total Point Score
ANR26650M1-B	2,610 W	31.68	12 V	10.65	0.67	$32.65	90
Samsung 30Q	216 W	43.2	12 V	5.204	0.42	$55.00	80
Samsung 25R	288 W	36	12 V	5.204	0.38	$38.50	85

Each of these cells fit within the requirements of our project with their determining factor being their price. We decided to use the ANR26650 because one of our team members previously owned 4 of these cells and is donating to the project free of charge. As such, despite the lower module capacity and higher weight, the cost of the module was the determining factor, with a total point score of 90 points.

3.3.7.8 Supercapacitors Design

In this section we will go over the design process and reasoning why we decided upon using Capacitech Energy's Hybrid battery and supercapacitor design.

"Supercapacitors are another type of energy storage device. Unlike batteries, which store energy through chemical reactions, supercapacitors store the majority of their energy electrostatically. As a result, they can charge and discharge energy much faster than batteries, with power densities typically 10 times greater.

The biggest advantage of supercapacitors is their high power density, allowing them to supply large amounts of power in short bursts. This makes them ideal for applications that require quick bursts of energy such as starting a motor or pump. Supercapacitors also have a much longer lifespan than batteries since there are no chemical reactions taking place, reaching cycle lives of up to one million…

…However, supercapacitors have a much lower energy density than batteries and are not suitable for applications that require long-term storage of energy. Supercapacitors also have a high self-discharge rate, meaning they will lose a good chunk of their charge when not in use, and are typically more expensive than batteries."

So, while supercapacitors have their downsides, consisting of high self-discharge rates, much lower energy density, and their high price, supercapacitors still have their benefits for Hive. As Capacitech Energy stated, supercapacitors are best used in bursts of energy, for example when starting a motor. Our project will consist of many motor stops and starts when pulling heavy payloads and capacitors will help in these areas in overcoming static friction.

3.3.7.8.1 Supercapacitor Requirements

The high level requirements for our supercapacitor modules are to assist in burst power applications to overcome static friction when pulling our payloads. Capacitech will be supplying the Hive project with a cable based supercapacitor system to assist in our burst power delivery capabilities when developing Prototype 2.

3.3.8 DC Voltage Regulators

Product research will be divided into 2 sections: Voltage Regulator Device Research and Connector Research.

The Voltage Regulator Device Research will contain 3 sections, one for each voltage type we plan to include in our Agent. Within these sections devices will be put through the filter of requirements and showcased below.

Table 3.17: 3.3 V Regulator

Device Name	**Simple Implementation *(Less than 10 components)***	**Max Input Voltage *(above 16.8 V)***	**Output Current *(500 mA Output)***	**Cost *(Less than $1.50)***	**Stock *(Above 2000 Units)***
ZLDO1117QG33TA	2 components	18 V	1 A	$0.67	28,610 Units
NCV1117DT33T5G	2 components	20 V	1 A	$0.78	2,567 Units
MC33269DR2-3.3G	3 components	20 V	800 mA	$0.83	28,221 Units
LDI1117-3.3U	2 components	20 V	1 A	$0.94	2,404 Units

We will not be using the NCV1117DT33T5G, MC33269DR2-3.3G, or the LDI 1117-3.3U voltage regulators. The primary reason we will not be using the NCV1117DT33T5G voltage regulator is because there is not a lot of stock currently in for it, which means that it is less likely for us to be able to obtain the amount we need in a timely manner. It is also slightly more expensive than another option, which could potentially add up. The reasoning behind why we will not be using the MC33269DR2-3.3G voltage regulator is because it is relatively expensive, which is antithetical to our goal of constructing the agent inexpensively while maintaining operational viability, of which this part provides no additional utility over another for the price it costs. We will not be using the LDI 1117-3.3U voltage regulator as it is the most expensive of the voltage regulators compared above, which is antithetical to our goal of constructing the agent inexpensively while maintaining operational viability. It is critical to preserve the budget when possible as we are limited in budget and therefore need to pick what parts we use smartly to ensure we do not exceed the budget we are afforded. It also has the least in stock of the voltage regulators compared above, which means that it is less likely for us to be able to obtain the amount we need in a timely manner. As clearly seen on the premise of the reasoning behind the voltage regulators not being used, amount in stock is the most important consideration with price being a close secondary consideration as we do have a limited budget but being able to actually obtain the voltage regulators that we need is more important.

We will use the ZLDO1117QG33TA 3.3 volt regulator. The reason we will be using the ZLDO1117QG33TA 3.3 V regulator is because it is inexpensive and has quite the expansive amount currently in stock. The ZLDO1117QG33TA 3.3 V regulator being inexpensive is useful as saving money and preserving the budget is one of our main goals for the agent, as well as constructing multiple agents for coordination purposes as that will effectively double the cost as there will be two agents constructed instead of one. It is also critical to preserve the budget when possible as we are limited in budget and therefore need to pick what parts we use smartly to ensure we do not exceed the budget we are afforded. The ZLDO1117QG33TA 3.3 V regulator having an expansive stock is good for us as it ensures that we will be able to obtain one, which is good as it will be required for the operation of the agents and will allow us a measure of security against encountering supply chain issues.

The 3.3 V regulator will be used for the ESP32-WROOM-32. The limit switches can use 3.3 V or 5 V. The ESP32-WROOM-32 is critical to maintaining the normal operation of the agent, so it is critical that there is a way to supply power to the ESP32-WROOM-32 because otherwise the agent would not operate properly. The limit switches are also vital to the normal operation of the agent as they function as the methodology through which the agent is capable of detecting collisions which means it is vital power is supplied to the limit switches so that the agent can determine its location with more efficiency and accuracy.

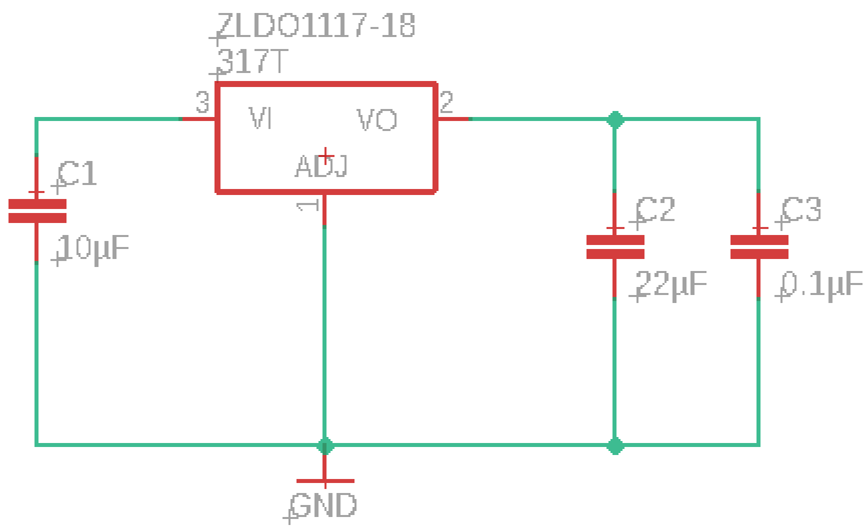

Figure 3.31: Basic 3.3 V regulator. Designed by Cooper Fitzgerald.

The above circuit design is of a 3.3 V regulator using three capacitors and the selected component to take an input voltage and produce an output of 3.3 V, which is what voltage regulators do. The capacitors selected have a package size of 1206 as we want to reduce the size of the voltage regulator to have more space and generally keep the size of the agent down.

Table 3.18: 5.0 Volt Regulator

Device Name	**Simple Implementation *(Less than 10 components)***	**Max Input Voltage *(above 16.8 V)***	**Output Current *(1 A Output)***	**Cost *(Less than $1.50)***	**Stock *(Above 2000 Units)***
NCP1117DT50G	2 components	20 V	1 A	$0.76	1,957 Units
BD50FC0FP-E2	2 components	26.5 V	1 A	$0.95	16,264 Units
NCV1117DT50RKG	2 components	20 V	1 A	$0.79	35,019 Units
LDI1117-05H	2 components	20 V	1 A	$0.98	7,277 Units

We will not be using the NCP1117DT50G, BD50FC0FP-E2, or LDI1117-05H voltage regulators. Our reasoning for avoiding the use of the NCP1117DT50G voltage regulator is because it has an extremely limited stock and would therefore be difficult to obtain in a timely manner which would be counterproductive as it would halt the construction of the agent and its prototypes. We are avoiding using the BD50FC0FP-E2 voltage regulator because it is fairly expensive and would not provide enough additional utility to the agent to justify buying it. While it can accept a greater input voltage than the other voltage regulators detailed above, that is unnecessary for the agent and the usage of the 5 V regulator, so there is no need to opt for the more expensive option. The LDI1117-05H voltage regulator will not be used because it is the most expensive option for no benefit as compared to the other options listed above. It also has a lower amount in stock, which means that it is less likely for us to be able to obtain the amount we need in a timely manner. As clearly seen on the premise of the reasoning behind the voltage regulators not being used, amount in stock is the most important consideration with price being a close secondary consideration as we do have a limited budget but being able to actually obtain the voltage regulators that we need is more important.

We will be using the NCV1117DT50RKG 5 V regulator. The reason we will be using the NCV1117DT50RKG 5 V regulator is because it is inexpensive and has a very significant amount currently in stock. The NCV1117DT50RKG 5 volt regulator being inexpensive is useful as saving money and preserving the budget is one of our main goals for the agent, as well as constructing multiple agents for coordination purposes as that will effectively double the cost as there will be two agents constructed instead of one. There being a significant amount of the NCV1117DT50RKG 5 V regulator currently in stock is a huge boon as it practically guarantees that we will be able to obtain what we need and will be able to produce and construct a fully operational pair of agents that will function as required. Another benefit to the NCV1117DT50RKG 5 volt regulator is that it can handle up to 20 V of input, which is approximately what the maximum voltage capable of being provided by the battery is, so the NCV1117DT50RKG 5 V regulator will be able to handle the cases of the battery surging to near maximum and will therefore allow for the agent to remain operational in even more situations which is good as it provides us with additional security for the proper operation of the agent.

The 5 V regulator will be used for our PCB. The PCB is obviously a critical part to the functionality of the agent as it will be assisting the offboard decision maker in sending commands to individual parts of the agent and ensuring that all commands received are properly executed.

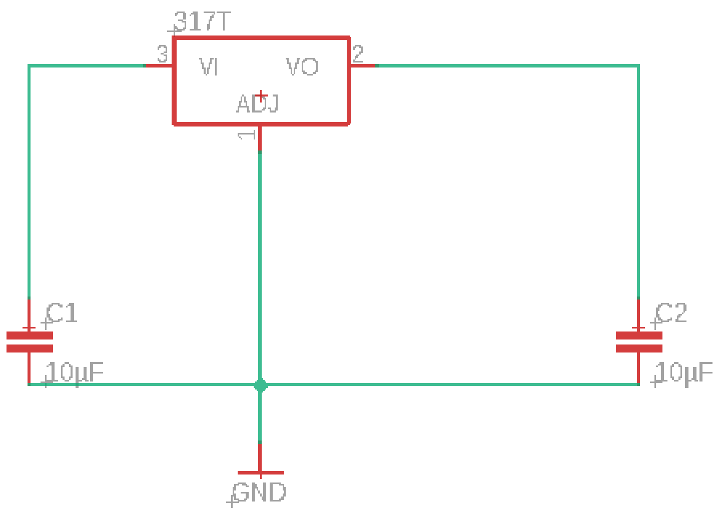

Figure 3.32: Basic 5 V regulator. Designed by Cooper Fitzgerald.

Above is a basic 5 V regulator circuit, using two capacitors along with the component selected above to convert an input voltage to a 5 V output, which is what will need to be done. The capacitors selected have a package size of 1206 as we want to reduce the size of the voltage regulator to have more space and generally keep the size of the agent down.

Table 3.19: 12.0 Volt Regulator

Device Name	**Simple Implementation *(Less than 10 components)***	**Max Input Voltage *(above 16.8 V)***	**Output Current *(3 A Output)***	**Cost *(Less than $1.50)***	**Stock *(Above 2000 Units)***
LM1085IT-12 /NOPB	4 components	18 V	3 A	$2.46	2,688 Units
MIC29300-12 WU-TR	2 components	26 V	3 A	$4.72	4,372 Units
ACT4533BY H-T	19 Components	38 V	3.5 A	$1.43	7,581 Units
LMR51430X FDDCR	6 components	36 V	3 A	$2.03	4,995 Units

We will not be using the ACT4533BYH-T, MIC29300-12WU-TR, or LMR51430XFDDCR voltage regulators. The ACT4533BYH-T voltage regulator will not be used because it has 19 components and that is too complex for the agent. We will not be using the MIC29300-12WU-TR voltage regulator because of how expensive it is, which is almost twice as expensive as the next most expensive voltage regulator at $4.73 per unit. This would be a drag on our budget if we were to go for it as we are going to be constructing multiple agents, and requiring multiple of these would limit our capability to buy other more important parts. Also the MIC29300-12WU-TR voltage regulator isn't better than others on the above list in terms of maximum voltage input and current output. We will not be using the LMR51430XFDDCR voltage regulator because it is slightly more expensive than the option we are going with and we would like to preserve the budget as much as we reasonably can. As clearly seen on the premise of the reasoning behind the voltage regulators not being used, amount in stock is the most important consideration with price being a close secondary consideration as we do have a limited budget but being able to actually obtain the voltage regulators that we need is more important.

We will use the LM1085IT-12/NOPB 12 volt regulator. The reason we will be using the LM1085IT-12/NOPB 12 V regulator is because there is a significant amount in stock as well as being inexpensive and having a low amount of components. There being a significant amount in stock is important because supply chain issues are currently running rampant, and being assured, or at least as assured as reasonably expected, of being able to acquire the 12 volt regulator is of critical importance as it is necessary for the agent to operate properly. The LM1085IT-12/NOPB 12 volt regulator being inexpensive is good as it allows for us to maintain our budget for other higher priced items and also for the duplicity of creating two agents to be able to demonstrate the cohesion and teamwork that is desired. It will also not negatively affect the agent to use this lower priced 12 volt regulator, so saving money is overall a net positive as it will not negatively impact the agent's operations. The LM1085IT-12/NOPB 12 V regulator also has an output current of 3 A, which will be required for the cases they will be used for. The LM1085IT-12/NOPB also has a lower number of components which is good as it keeps the complexity of the voltage regulator down.

The 12 V regulator will be used for the motors. The motors we are using will draw 12 V, which means that 12 V regulators will be required for the proper operation of the motors and therefore the agent as the agent will have no methodology of locomotion without the motors running. Locomotion is most certainly required for proper operation of the agent as the demonstration will have the agent moving a payload, which will inherently require locomotion, which therefore means the motors are going to require power, which inherently requires the 12 V regulator to accomplish.

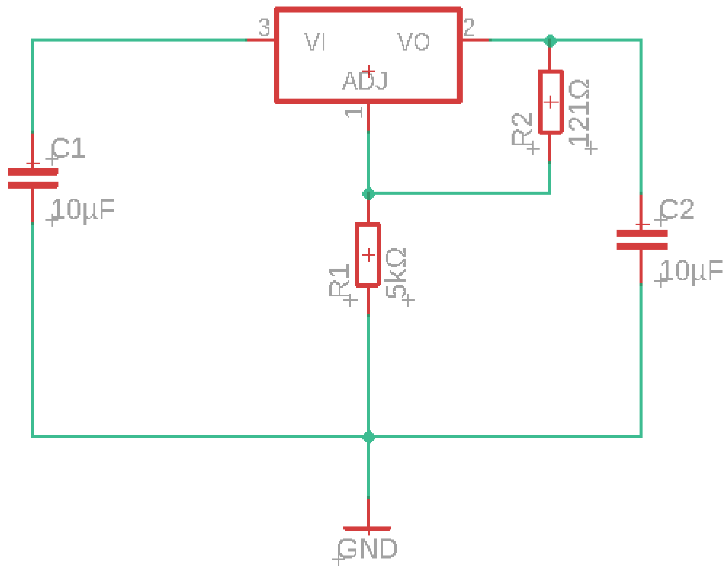

Figure 3.33: Basic 12 V Regulator. Designed by Cooper Fitzgerald.

Above is a basic 12 V regulator circuit, using two capacitors and two resistors along with the component selected above to convert an input voltage to a 12 V output, which is what will need to be done. The capacitors and resistors selected have a package size of 1206 as we want to reduce the size of the voltage regulator to have more space and generally keep the size of the agent down.

3.4 Possible Architectures for Proposed Subsystems

When it comes to the mechanical side of this project, there are two main subsystems, those being the surface to ground interface and the payload manipulation. Due to the planned environment for the agent, any unmanned aerial options were quickly cut from the list. An example of this would be a drone. While this does have potential to work, the team determined that it would be unrealistic in most warehouses. This would also be not plausible due to the hardware and coding constraints from the proposed communication plan. This section compiles the considerations taken when making the final decisions in concept design.

3.4.1 Hovercraft

The hovercraft architecture is a surface to ground interface in which the main body of the vehicle has little to no contact with the surface it travels over. The hovercraft instead relies on blowers to produce a large volume of air below the body creating an air cushion for the vehicle to travel over. On initial thought this could be applicable to our situation, however there are many issues with this concept for our design. Firstly, our agent is required to be able to accurately pull or push a payload to a specific spot. While hovercrafts can accomplish this, the main issue is control. This type of design does not have the capability to safely decelerate quickly, which becomes increasingly problematic with larger loads . Finally, this design would require several AC motors to create the necessary air cushion. Not only are they less efficient and consume more power, but they are also typically more expensive. Ultimately, this design does not seem to have what it takes for this application, especially when scaled to the actual warehouse environment. It is important to note another major benefit of the hovercraft design is the ability to traverse over both land and water, which is not necessary in this situation.

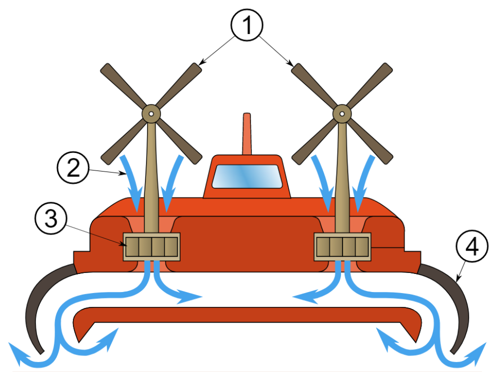

Figure 3.34: Diagram of how a hovercraft hovers. The blue arrow shows the direction of fluid flow in which it creates a pocket of air to travel over. (Woland)

After these considerations we decided to go with a wheel to ground interface that would provide our agent with the most speed and force, while still being capable of accurately and safely controlling itself with and without a payload.

3.4.2 Rail System

Another type of agent to ground interface considered was the implementation of a track system. This would consist of first taking our defined warehouse environment and then designing a track system for the agents to travel across. This system has many benefits in warehouse application. The first being, its modularity and use of vertical integration, with the use of this design sources have reported that this architecture provides the fastest fulfillment per square foot, maximized uptime, and is highly energy efficient (Jenkins). This checks many of the boxes for the required tasks the agent must perform in our demo, however, the goal of the project is to create an autonomous robot that is modular and adaptable in nature. This architecture shines in the warehouse setting, but it is important to understand that our agent will be performing tasks in a warehouse environment for the demo, this will not be the sole purpose of the agent in the long term, therefore, this architecture does not check all the boxes necessary to be implemented in the design of the robot.

3.4.3 Drivetrain

This section covers the decision-making process that we used to select a drivetrain system. For mobile ground robots, the drivetrain is of utmost importance, as without it, the robot would be unable to move. There are many different types of drivetrains, each with their own sets of advantages and disadvantages.

For our project, we wanted to design our robots to be as modular and as robust as possible. Holonomic drive systems offer much more freedom than nonholonomic drive systems, as these systems allow the robot to drive in any direction, including completely sideways. We considered a few different holonomic drive systems, including Mecanum Drive, X-Drive, H-Drive, and Crab Drive. We will be discussing the advantages and disadvantages of each of these systems and why we chose the system that we did.

3.4.3.1 Mecanum Drive

Mecanum Drive is a method of driving using specially designed wheels that allow for holonomic maneuverability. These wheels are arranged similarly to a standard Tank Drive system, but utilize smaller rollers that have an axis of rotation at 45 degrees to the wheel plane and at 45 degrees to the axle line. To allow for holonomic maneuverability, the rollers at the top of the wheels must have their axles be parallel to the diagonal of the vehicle frame. The use of these rollers means that the wheels will exert a force in the diagonal direction, which when combined with varying the rotational speed and direction of each wheel, then the summation of the force vectors can create both linear and rotational movements, allowing full strafing capabilities. To fully utilize the potential of Mecanum Drive, each of the four wheels must be independently driven, meaning a minimum of four motors is required. Mecanum Drive allows for a holonomic system, but due to roller friction, the traction and available force in the sideways direction is reduced, meaning it is not as efficient to strafe as it is to drive forward or backward. Mecanum wheels are relatively expensive, so to use this drivetrain system, we would need to manufacture our own wheels to save on costs.

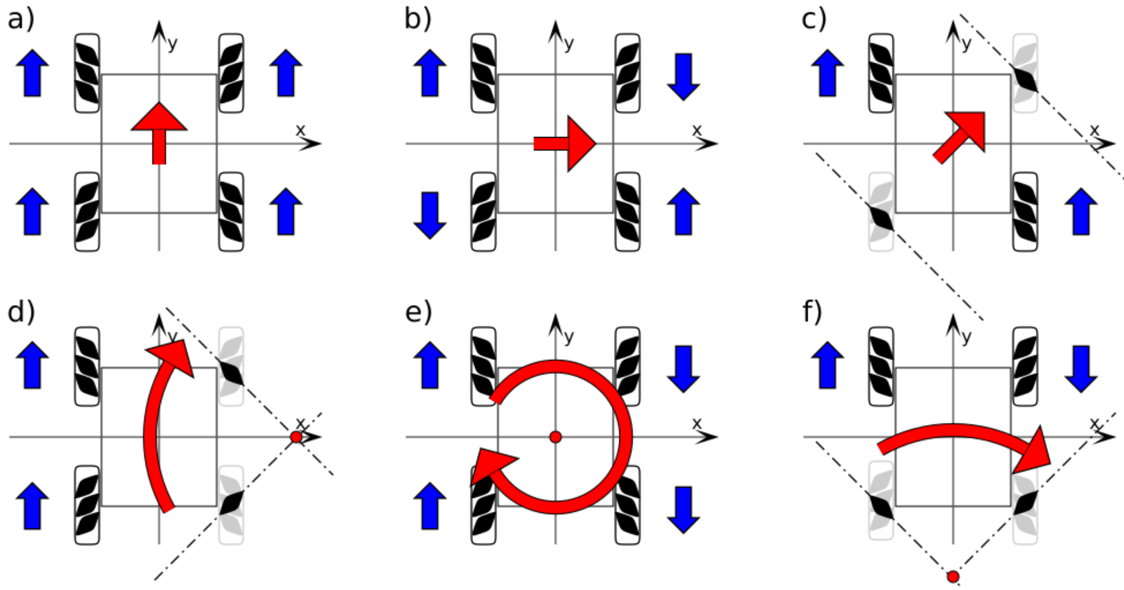

Figure 3.35: Mecanum Drive wheel control principle. Blue arrows show drive direction of the wheels and red shows the moving directions of the vehicle. (a) Moving straight ahead. (b) Moving sideways. (c) Moving diagonally. (d) Moving around a bend. (e) Rotation. (f) Rotation around the central point of one axle.
Mrmw, CC0, via Wikimedia Commons

3.4.3.2 X-Drive

X-Drive is another method of driving utilizing specially designed wheels, specifically omni wheels, to allow for holonomic maneuverability. These omni wheels also utilize rollers, but these rollers are arranged perpendicular to the turning direction of the overall wheel. To employ this drive system, each of the four wheels need to be rotated 45 degrees such that the main axles are parallel to the diagonal of the vehicle frame, which will have the main axles make an “X” shape, hence the name “X-Drive”. The orientation of these wheels makes the drivetrain take up more space, which can be an issue if there is a small size requirement. Like the Mecanum Drive, this drive system will also require a minimum of four motors, and the wheels will exert a force in the diagonal direction, which allows for full strafing capabilities. Unlike the Mecanum Drive, the sideways speed and the forward speed is equivalent in X-Drive, meaning it is much more maneuverable. However, these wheels in this orientation make it much easier for the robot to be pushed around, and thus can be more difficult to quickly come to a complete stop.

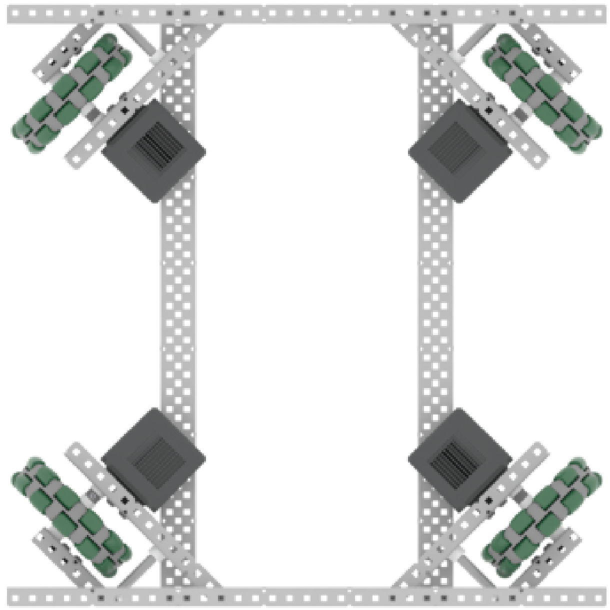

Figure 3.36: Bottom-view of an X-Drive system. (“VEX Drivetrains”)

3.4.3.3 H-Drive

H-Drive is another method of holonomic driving also utilizing omni wheels. This drive system uses a minimum of five omni wheels and three motors. In this minimum configuration, four of the wheels are positioned the same as in a Tank Drive system, while the fifth wheel is placed in the center of the drivetrain and is perpendicular to the other four wheels. One motor can be used to drive the left two wheels, another to drive the right two wheels, and a third motor to drive the center wheel. The top-down view of this drive system gives it the name of “H-Drive”, where the center wheel makes up the horizontal line in the “H”, while the other wheels make up the vertical lines. While this system does allow for holonomic maneuverability, its strafing is generally much slower and/or weaker than its forward motion. To ensure the center wheel has adequate contact with the ground, especially in uneven terrain, it is generally required to have a tension system that pushes the center wheel toward the ground. This system is less maneuverable than both the Mecanum Drive and the X-Drive, but it also requires less motors.

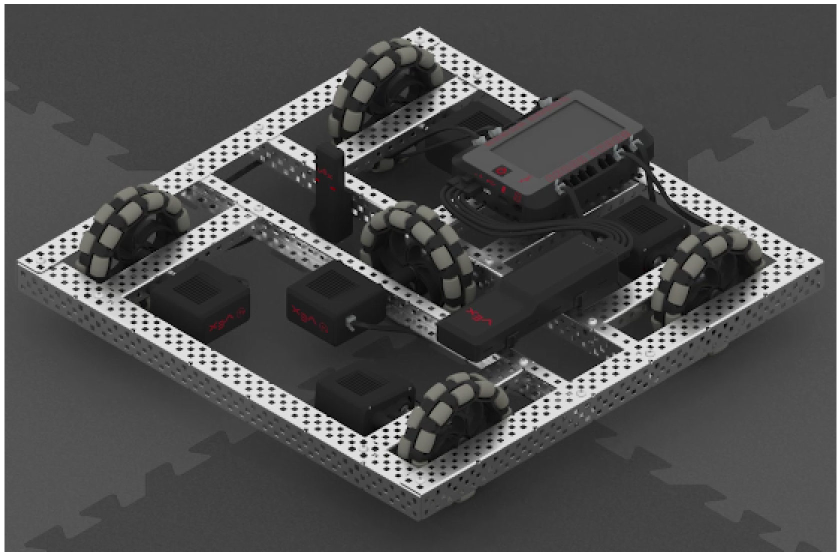

Figure 3.37: An H-Drive system. ("VEX Drivetrains")

3.4.3.4 Crab Drive

Crab Drive is a method of holonomic driving that is quite different from the other systems that were previously mentioned. This system does not have to use omni wheels, rather, standard pneumatic wheels can be used. This system works by allowing all wheels to be rotated along the Z-axis. This rotation can be achieved by the use of a single motor that drives a chain or belt which then can rotate all of the motors at the same time. This system typically uses four wheels, and the method of driving them can vary depending on the application. Technically, two motors could be used to drive this system and allow the robot to traverse in all directions, however the robot would not be able to effectively rotate. To achieve rotation while minimizing the number of motors that will be used, one motor can be used to rotate all the wheels (as previously mentioned), another motor can be used to drive the left two wheels, and a third motor can be used to drive the right two wheels. This drivetrain can be quite complicated to implement, especially with only three motors. It would be much easier to independently drive each wheel, which requires four motors, while having a fifth motor used for the rotation of the wheels along the Z-axis. However, the use of five motors will make the overall cost of the robot more expensive. While this system does allow for holonomic driving, the strafing is not instantaneous since it takes time to turn the wheels to the desired angle.

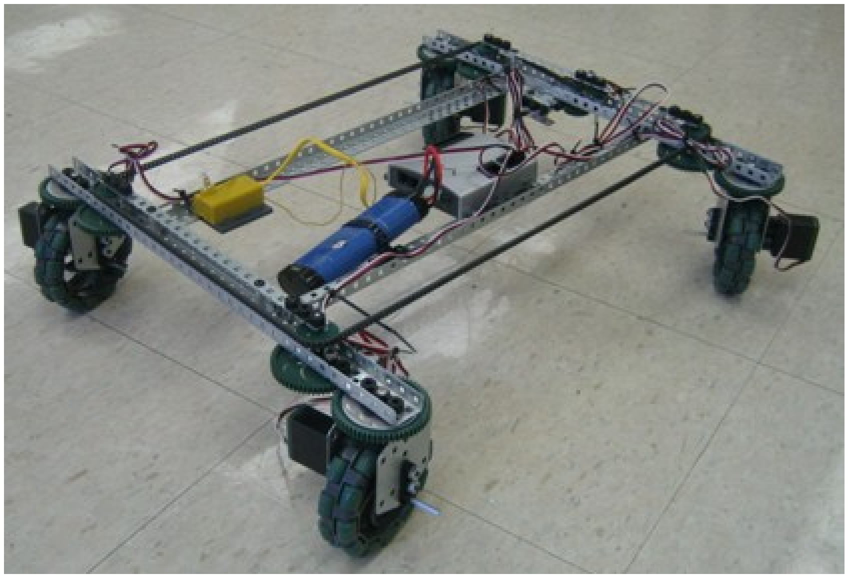

Figure 3.38: A Crab Drive system. (Mathews)

3.4.3.5 Selected Drivetrain

We decided to use the Mecanum Drive for our ground agents. When compared to the other drive systems that were mentioned, Mecanum Drive offers great maneuverability that is second only to X-Drive, while offering more traction and friction than X-Drive. While the strafing speed and strafing force of the Mecanum Drive is slightly less than that of its forward speed and force, it is perfectly acceptable for our application. When looking at the cost, H-Drive is going to be the cheapest option, as the omni wheels are cheaper than mecanum wheels, and H-Drive requires only three motors, whereas Mecanum Drive requires four. We decided to manufacture our own mecanum wheels via 3D printing, which makes the overall cost of the wheels themselves negligible. While H-Drive is still going to be cheaper than Mecanum Drive due to fewer motors, H-Drive has significantly reduced strafing speed and force compared to Mecanum Drive.

After these considerations we decided to go with a wheel to ground interface that would provide our agent with the most speed and force, while still being capable of accurately and safely controlling itself with and without a payload.

3.4.4 Payload Manipulation

The main design requirement for this project aside from moving is: The agent must be able to manipulate a payload. So, firstly we must define a payload. Initially this was just a simple box, with the scale of an average amazon box (about 1 cubic foot). When thinking about this, the first consensus was to move the box by lifting it and moving it to the specified location. This would work well as it would allow for symbols to be placed on the boxes and the overhead camera could then be able to understand where the targets are at and move the agents accordingly. After looking into methods of lifting, the two best results were either pneumatics or an electric motor with a type of power transmission. Afterward, we quickly found out that the energy to lift these boxes just 5 inches off the ground would be too strenuous for a battery powered agent. This required the definition of our payload to be redefined. It was then decided that the payload would be a cart on wheels that we designed to carry boxes instead. By doing this we would be able to move more boxes at once for a much cheaper energy cost.

3.4.5 Latching Mechanism

With the newly defined payload, the lifting mechanism was replaced with a latching mechanism. The main purpose of this device was to be capable of attaching and detaching itself from the payloads. This design would be a more passive approach and is planned to only need a single servo to run. Our inspiration came from your typical gate latch. When it comes to gate latches there are three main types: Gravity, spring loaded, and bolt secured. For our application, the gravity type would be best, by choosing this design a single simple servo would be the only motor needed for articulation of the device. Picture below is a standard gravity gate latch.

Figure 3.39: A Gravity gate Latch (Sarah)

This will be a very robust design in many aspects. The main one being that this design will take all the stress off the motor and put it all on the structure of the agent and the mechanism itself. Doing this will allow for cheaper cost, since a weaker motor can be implemented. This will also greatly benefit the coding team as it will allow for a greater margin of error when attempting to latch to the targeted payload. It is important to note that speed and latching is more important than latching at a very specific spot on the payload, so to allow for the quickest latch, two of the proposed mechanisms will be placed on the perimeter of the agent. The main goal of this is to minimize the time it takes for the agent to acquire payloads that it may not be completely square with. Between the holonomic drive and this, the time required to articulate a payload should be minimum. Doing this will lead to quicker articulation of the payloads in the environment leading to an overall increase in production and efficiency in our system.

Since we may not use an off the shelf gravity latch it is important to understand the conceptual design of it. The main two parts of any gravity latch is the frame housing the lever and the pin that slides into its position locking the payload in this case into place. These levers are designed with curves in them to easily accept the pin. With this design in place the only way to keep the pin in place is to lock the lever in place. This is where we will need to adapt this concept to fit our design, essentially our proposed single servo will be used to articulate an locking mechanism to restrict the movement of the payload.

3.4.6 Linear Slide Articulation

When considering possible payload manipulation architectures another type that was considered was a linear slide system that would be mounted on the side of the agent. The main strategy for this design would have the agent pull up close to the payload and then extend the slide to the length of the payload and somewhere a finger would extend capturing the payload, while this architecture does seem plausible, it has several cons that steered the team away. The first being that with this design the agent would only be able to grab from one side of its perimeter, whereas we would like to reserve the ability to attach payloads on all sides of the agent at once. The second issue with this would be the limitations of the size of payload that we could manipulate, with the slides being mounted on the perimeter of the agent, it can only be as large as the bot’s length. With something like a latch, we would be able to attach to much larger payloads and attempt to move them. Finally, the team decided that this method would not be as energy efficient as other options due to the potential need for more actuators and mechanical losses that can happen with longer translational slider mechanisms. An example of one of these mechanisms is pictured below.

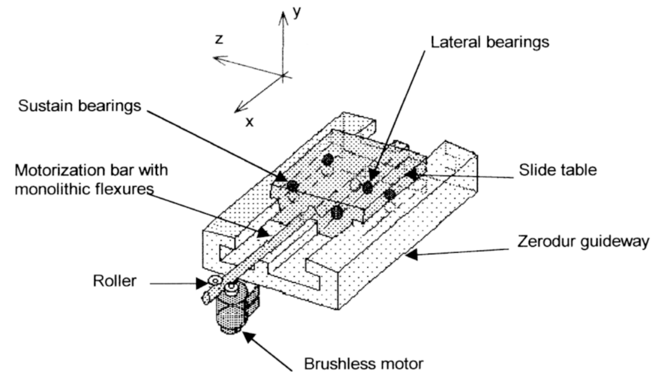

Figure 3.40: An example of these slides are the traditional drawer slides (Mekid)

3.4.7 Lifting Mechanism

Another proposed concept for this subsystem was a lifting mechanism. This would consist of a passive or active grabber actuator and an actuator that would be used to lift the payload off the ground. Immediately, there are some advantages to this. The first seemingly obvious one is the fact that the payload would be able to be elevated to a higher level. This could most definitely be used in a warehouse application after conducting research. It was found that many robots in this environment act as help to employees to help increase productivity. From here it was understood that lifting a payload to heights greater than a foot would not make sense in the applicable environment to our agent. This did not however did not eliminate this as a potential architecture for the subsystem, it was decided that lifting the payload could still be utilized but in this case the height it would be elevated to would be much lower than say a shelf. The main disadvantage of this mechanism is the number of resources it would take to lift a payload, especially as the weight of the payload increases. With an expected payload of up to 25 pounds, it was deemed ineffective for our application, not to mention with hopes of moving more weight than planned this strategy quickly dropped to the bottom of the list for potential payload manipulation devices, however, to fully develop this concept follow up research was conducted. The two main plausible methods found for this concept were either the application of pneumatic or electric actuators. Pneumatics is commonly implicated in industrial settings, however over the years, they have made their way into robotic designs time and time again. A great benefit of this type of system is the input to output cost for power to torque. Through the application of pneumatic circuitry and two 2-way actuators, this could be utilized to grab and lift a payload to move. The downfalls of this come with the great size and weight of the equipment needed to employ these systems, this is usually not an issue in an industrial environment. In the scope of a small mobile autonomous robot however, power and size resources are much more limited. These systems seem to shine in large scale stationary projects. When looking into electric actuation, similar issues arise, even with a passive gripper, the amount of energy required to lift heavier loads would be too high, making it to where this would not be feasible as we continue to increase the weight of the payload. Pictured below is an example of a pneumatic schematic of a 2-way actuator.

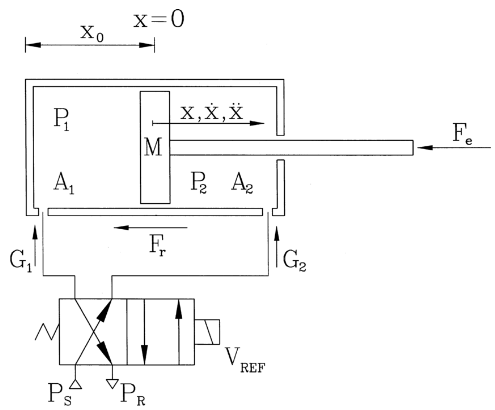

Figure 3.41: Shows the circuit of a two way pneumatic actuator. (Gastaldi)

3.4.8 Locking Mechanism

When it comes to a gravity latch, it is important to understand that without the pin in place the latch will not hold anything in place. In most applications of this latch, it relies on human interaction. In this instance someone would pull the gate too and place a pin in the designated hole to restrict the motion of the lever. In our case we need to automate this process, to accomplish this, some sort of mechanical system will need to be introduced that will actuate a pin into its correct position completely limiting the motion of the mechanism. There are several ways to actuate such a system to create the motion necessary. One of the most common methods of accomplishing this is the use of kinematic chains. Kinematic chains refer to the assembly of joints and links that provide a controlled output motion in response to a supplied input motion. A link is a rigid body that possesses at least two nodes that serve as a point of attachment for rigid bodies, whereas a joint is a kinematic pair that allows motion between the connected links. Finally, a mechanism consists of several kinematic chains that are used to transmit force and motion. With all of this, it is important to understand that there are many different types of commonly implemented mechanisms and each of them are characterized in different ways, however, they are all characterized by four main things. Them being degrees of freedom (DOF), overall geometry, joint variables, and end effector coordinates. These are quantified values that explain what is going on in the mechanism and explain what the system is capable of. This all leads into the discussion of power transmission. The most basic definition of power transmission is a combination of actuator and mechanism used to transfer motion/force from where it is created to where it is needed. While there are several different types of power transmission, for the proposed locking mechanism, this discussion will mainly focus on the types used to convert rotational motion to translational motion. Narrowing the search once more, this mechanism does not need to be capable of cyclical motion, for example, something like a gearbox in a drivetrain. The main point of this mechanism will be to move a pin in and out of the gate latch frame. A very solid option for this would be a slider crank option. This type of power transmission is made up of three main components: a crank, connecting link, and slider. The crank is the link in the system that is connected to the actuator and will serve as the main source of torque and moves in purely rotational motion. The slider is the link in the system that consists of purely translational motion and is considered the output of the system. The final piece of this mechanism is called the connecting link, this piece completes the mechanism as it moves with translational and rotational motion. With the geometry and joint variables known, the relation between input and output for both displacement and velocity can be found and expressed as:

$$x = r\cos\theta + l\left[1 - \left(\frac{r}{l}\sin\theta\right)^2\right]^{1/2}$$

Figure 3.42a: The equation above shows the parameters that define displacement of a classic crank-slider mechanism. (Alciatore)

$$\dot{x} = -r\dot{\theta}\left[\sin\theta + \frac{r}{2l}\frac{\sin(2\theta)}{\left[1 - \left(\frac{r}{l}\sin\theta\right)^2\right]^{1/2}}\right]$$

Figure 3.42b: The equation above is the derivative of the first therefore expressing the velocity of a classic slider-crank mechanism. (Alciatore)

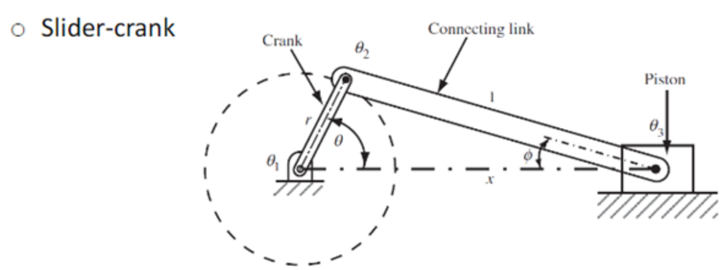

Figure 3.42c: The diagram above shows the basic geometry of a slider-crank mechanism that directly ties to the equations presented. (Alciatore)

If we used a servo at the origin of the crank, we will be able to utilize this mechanism to move a pin in place to lock the latch, with this we could also mirror this setup and control two pins at once, allowing for two latches to be used at once.

Another potential mechanism that could be used for this application is a rack and pinion. This mechanical system uses a pinion gear mated to a rack to transfer rotational motion to translational motion. With this system the same physics applies to them just as every other gear train, through varying the diameter and the number of teeth of each gear the gear ratio changes, which in turn alters the input to output relation. The greatest downfall of this system would be the fact that the axis of the rotational motion and translational motion are not collinear, this leads to an increase in space needed for this system to operate as intended. Through varying the geometries of each mechanism, the same amounts of torque can be produced with both, therefore, it comes down to the size and cost constraints that will play the largest part in making this decision. Realistically, it would cost more to manufacture a rack and pinion than it would for a slider crank mechanism. Pictured below is an example of a rack and pinion.

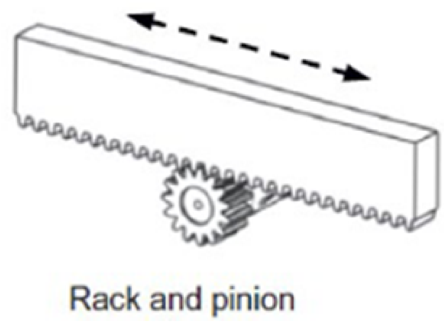

Figure 3.43: The diagram above shows rotational motion of the pinion gear being changed to translational motion of the rack. (Alciatore)

3.4.7 Possible Payload Architectures

With plans to move packages with a payload system, this introduces the need for cart design. After the payload was defined, this allowed for possible architectures to be researched and discussed with the team. Many of the similar robots in this industry rely on a more traditional non holonomic tank drive to drive the bot around the environment. With our agents utilizing a holonomic mecanum drive the payload designs that are used currently will not work. The current designs coming from Prime utilize a static shelf design where the bot maneuvers under and then extends a lifting mechanism to raise an entire shelf of goods. This design allows for several goods to be moved at once, however, they require a greater amount of torque and friction produced to function properly. While this design relies on vertical integration our proposed design relies on horizontal integration. While this may not be the absolute best design for warehouse implementation it does, however, promote modularity, which is one of the main goals for our project. To properly utilize our agents for payload manipulation we will implement latching mechanisms that can be placed on any side of the bot allowing the agent to interact with payloads from all faces. This plays off the benefits a holonomic drivetrain provides, this being the fact that our agents will not have a true front and back, this will allow for the most efficient path planning while maintaining modularity.

While Prime utilizes an autonomous shelf moving design, another company, Locus, has implemented different designs in which their bots act more as personal assistants for warehouse workers. Their design aims to reduce aisle roaming time of the workers and overtime wages, contributing to an overall more efficient warehouse environment.

Our agents will mainly serve as a payload manipulator rather than as a worker's assistant. Our payload will utilize horizontal integration and needs to have the ability to follow the strafing movement of the agent. So, this led the research to a simple square payload with the same footprint of the agent with four caster wheels mounted on each of the corners. This architecture will allow for holonomic movement with little friction, and it will be capable of carrying the anticipated load the payload will experience. This seems to be the best architecture for the payload subsystem and will allow our agents to not only efficiently manipulate a warehouse environment, but also show the modularity capabilities of the agents.

3.5 Parts Selection Summary

In this section, we will go over the selected components for the agents in Hive: The Grounded Swarm, as well as the decision process for choosing these components.

3.5.1 Microcontroller

For the selection of the microcontroller, we decided to go with the Espressif ESP32-WROOM-32 MCU. Although both this component and the ATmega2560 can be easily programmed via the easy-to-use Arduino IDE, the ESP32-WROOM-32 has many more advantages. Not only does it natively support Wi-Fi, Bluetooth, and Bluetooth LE; but when compared to the ATmega2560, it has a wider data bus, additional flash memory, additional SRAM, a much higher clock speed, a lower operating voltage, and supports a larger variety of peripherals. Plus, the addition of the low-power coprocessor means that less power can be consumed for less intensive tasks, such as peripheral monitoring. Table 3.20 below shows a comparison of the most relevant specifications between the two microcontrollers, the ATmega2560 and the ESP32-WROOM-32.

Table 3.20: Comparison of Atmega2560 and ESP32-WROOM-32

	ATmega2560	ESP32-WROOM-32
Architecture	AVR	Xtensa
Data Rate (Max)	16 MHz	240 MHz

	ATmega2560	ESP32-WROOM-32
Wireless	None	Wi-Fi & Bluetooth
Operating Voltage	4.5 - 5.5 V	3.0 - 3.6 V
Flash Memory	256 KB	4, 8, or 16 MB
SRAM	8 KB	520 KB

3.5.2 Motors for Drivetrain Implementation

Based on our engineering design specifications, we decided that the goal was to carry more weight at a slower speed rather than less weight at a faster speed. With these motors' main purpose being to continuously rotate a wheel to traverse an environment; both stepper and servo motor types were excluded from the search. Between DC and AC motors, DC motors were the best option as they would produce the needed torque at much lower cost. Our search for motors quickly moved towards DC motors with speed reducer gearbox ratios. During the search for these motors, many companies came up, therefore, it took some time to find the company that provided products that best matched our constraints and required specifications. This search led us to the company Greartisan, as the motors they provide match the price point required and have high reviews, while also clearing our 1st order dynamic calculations. This led us to selecting them for the application in our drivetrain. Seen below is the weighted evaluation method used for drivetrain motor selection.

Table 3.21: Weighted Rating Evaluation Method for Motor Selection

	Motor	Greartisan 12V DC		Vexta SMK 216A AC		ElectroCraft Brushless DC	
Criteria	**Importance Weight**	**Rating**	**Weighted Rating**	**Rating**	**Weighted Rating**	**Rating**	**Weighted Rating**
Torque	60%	3	1.8	5	3.0	4	2.4
High Efficiency	10%	2	0.2	1	0.1	4	0.4
Low Cost	30%	5	1.5	1	0.3	1	0.3
Totals			3.5		3.4		3.1

3.5.3 Motor for Payload Interaction Latch

It is important to note that the proposed design for the locking mechanism will put the full load of the payload on itself rather than the servo. The servo torque requirements will be greatly decreased; however, the servo would be required to run at a much higher speed. This is necessary as the agent needs to be able to quickly latch onto and lock a payload into place once it is acquired. A high torque/ low speed servo is not ideal in this case. Below you will see our weighted evaluation method used to select the correct servo for our application.

Table 3.22: Weighted Criteria for Servo Selection

	Servo	Savox SV 1260MG		LD-20MG		JX PS-1171MG	
Criteria	**Importance Weight**	**Rating**	**Weighted Rating**	**Rating**	**Weighted Rating**	**Rating**	**Weighted Rating**
Torque	20%	4	0.8	3	0.6	3	0.6
Angular Velocity	30%	4	1.2	3	0.9	4	1.2
Low Cost	50%	1	0.5	5	2.5	5	2.5
Totals			2.5		4		4.3

3.5.4 Gate Latch

In this section of the report, we will use the weighted rating evaluation method to choose which gate latch best fits our application.

Table 3.23: Weighted Criteria for Gate Latch Selection

	Latch	National Hardware N184-861 BPB21		Sankins Gravity Gate Latch	
Criteria	**Importance Weight**	**Rating**	**Weighted Rating**	**Rating**	**Weighted Rating**
Foot Print	30%	5	1.5	2	0.6
Weight	30%	4	1.2	3	0.9
Low Cost	40%	4	1.6	3	1.6
Totals			4.3		3.1

3.5.5 Motor Controller

For the selection of the motor controller, we decided to go with the Texas Instruments DRV8876. When comparing the three motor drivers, the use of bipolar junction transistors for the L298N results in a much higher voltage drop when compared to the other two motor controllers which use MOSFETs, and thus the decision was quickly narrowed down to be between the TB6612FNG and the DRV8876. When comparing these two motor controllers, they both can supply our selected 12 V motors rated at 0.6 A with no problem, but if we wanted to later scale up to larger motors that require a higher voltage, then the DRV8876 is the clear choice. Besides this, the DRV8876 also contains much more information in the datasheet on controlling the driver, and provides the most flexibility in control, which is quite beneficial to our project since we want our agents to be versatile. Table 3.24 below shows a comparison of the most relevant specifications between the three motor controllers, the L298N, the TB6612FNG, and the DRV8876.

Table 3.24: Comparison of Motor Drivers L298N, TB6612FNG, and DRV8876

	L298N	TB6612FNG	DRV8876
Supply Voltage	4.8 - 46 V	2.5 - 13.5 V	4.5 - 37 V
Logic Supply Voltage	4.5 - 7 V	2.7 - 5.5 V	N/A
Continuous Current per Channel (Max)	2 A	1.2 A	?
Architecture	Dual Full-Bridge	Dual Full-Bridge	Single H-Bridge
Switch Type	NPN Bipolar Junction Transistors	LDMOS Transistors	N-Channel MOSFETs
Voltage Drop	1.8 - 4.9 V	0.15 - 0.7 V	~0.5 V

3.5.6 Inertial Measurement Unit

For the selection of the inertial measurement unit, we decided to go with the Pololu MinIMU-9 v5. This IMU has an additional option when choosing the sensitivity range for the gyroscope, and has an additional three options when choosing the sensitivity range for the magnetometer. The MinIMU-9 v5 also has a lower normal operating current for all three integrated sensors. While this IMU does have a larger zero-rate output than the MPU-9255, it does have a lower zero-g offset, a lower zero-gauss level, and a lower noise density for the accelerometer and gyroscope, all of which are important in minimizing the accumulated error. Table 3.25 below shows a comparison of the most relevant specifications between the two inertial measurement units, the MPU-9255 and the MinIMU-9 v5.

Table 3.25: Comparison of MPU-9255 and MinIMU-9 v5

	MPU-9255			MinIMU-9 v5		
	Accel.	**Gyro.**	**Magn.**	**Accel.**	**Gyro.**	**Magn.**
Range	±2, ±4, ±8, or ±16 g	±250, ±500, ±1000, or ±2000°/sec	±49 G	±2, ±4, ±8, or ±16 g	±125, ±245, ±500, ±1000, or ±2000°/sec	±4, ±8, ±12, or ±16 G
Normal Operating Current	450 μA	3.2 mA	280 μA	70 μA	0.9 mA	270 μA
Noise Density	300 μg/$\sqrt{Hz}$	0.01 °/sec/$\sqrt{Hz}$	480 nT/$\sqrt{Hz}$	90 μg/$\sqrt{Hz}$	0.007 °/sec/$\sqrt{Hz}$	400 nT/$\sqrt{Hz}$
Initial Tolerance	Zero-g X, Y: ±60 mg Z: ±80 mg	ZRO ±5°/sec	Zero-G ±3 G	Zero-g ±40 mg	ZRO ±10°/sec	Zero-G ±1 G

4.0 Standards and Design Constraints

This section deals with any standards related to the technology and systems used in the design and manufacturing of the agent. It will also discuss the impact of the realistic design constraints and how they may affect the design, manufacturing, and testing of the agent.

4.1 Standards

There are many applicable standards for the construction and usage of the agent that will be discussed, which will be taken into consideration when the construction of the agent is done.

4.1.1 ISO Applicable Standards

ISO 8373 refers to robots and robotic devices in both industrial and non-industrial applications. The way a robot is defined under this standard is as a programmed actuator with some degree of autonomy, which is the ability to perform tasks without the need for human intervention. An industrial robot is defined as being a reprogrammable multipurpose manipulator that can move in three or more axes that is used in industrial environments. The industrial robot includes the manipulator, the controller, and any hardware and software that allows for programming or instructing the robot. Modularity is defined as a set of characteristics that allow a system to be separated into modules and recombined. Modules are components with a defined way to remove and re-add to a system, which can be hardware, software, or a combination of the two. As the agent will be modular in nature, it is critical to understand the definition of modularity, especially relating to robotics. The agent will be integrating a methodology of modularity for the additions to the robot, so multiple different modules will be able to be implemented to the agent. A robot actuator is a mechanism that converts energy to affect motion of the robot, which in the agent will most notably be motors to provide locomotion to the robot, however there will be other actuators, such as the methodology through which the robot maneuvers the payload. Included are also definitions for robotic arms, robotic wrists, and robotic legs, however none of these are projected to be needed for the agent so they will not be stated here. The definition of joint used in ISO 8373 is a mechanical part that connects two rigid bodies and enables constrained relative motion between them. This is important as for the actuator that affects the payload, it will likely need to extend in some way and will therefore need a joint, or multiple joints, to effectively actuate the payload. The end-effector is a device specifically designed for the robot to be able to perform its task, which in the case of the agent, will be a gripper. The agent will be a rectangular, or Cartesian, robot, which means that it will have three axes which form a Cartesian coordinate system as this was determined to be optimal for the construction of the agent. The agent also will be a mobile robot, defined as a robot capable of moving under its own control, specifically a wheeled robot as the agent will use wheels as its primary methodology of locomotion. The agent will need to be able to utilize kinematics to determine how to move joints, as well as knowing its pose, defined as a combination of position and orientation. The agent will be functioning in autonomous mode, which is defined as the robot functioning in a mode where it can accomplish its task without direct human intervention being necessary. A user-interface will also likely be necessary as it allows for information exchange between a human user and the robot. The agent will have a protective stop implemented for safeguarding purposes, in the manner of an emergency stop button. A reduced speed safety function can also be implemented, as that requires a mode of operation where the speed of the robot is limited to a maximum of 250 mm/s, which is approximately 0.55 miles per hour, which is defined in this standard. The normal operating conditions of the agent will be defined in another section as this standard does not detail any specifics on what those should be, but the agent will have a clearly defined set of normal operating conditions that will be considered a baseline when considering operations of the agent.

ISO 9283 refers to pose and pathing for robots. This is relevant to the agent as it is an autonomous robot which means it will need an element of path planning, and in order to path plan efficiently the pose is necessary. Additional relevant definitions included in this standard are the word cluster, meaning a set of measured points used to calculate the accuracy and repeatability, and measuring time, which is the time elapsed when measurements are recorded. These are important for the agent as it is imperative that the robot be able to repeat tasks as being able to do a task once would prove nothing and the robot needs to be consistent. Measuring time matters as knowing how long it will take for the agent to move from one place to another and do a task will allow for proper planning on the human side of things to be able to optimize efficiency.

ISO 10218-1 refers to the safety measures that should be taken for usage of industrial robots. An important factor is a limiting device, which restricts the maximum space by stopping the robot, which will be relevant for the agent as for the demonstration of the agent, it will be in a constrained space so having a limiting device to ensure it does not go out of bounds will be a necessity. A protective stop will also be a necessity for safety, which will be implemented as an emergency stop button in an easily accessible location on the agent. The term safety-rated is defined as having a safety function with a specified safety-related performance. Safety-rated monitored speed is a function that causes a protective stop when the speed of the robot along one or more axes exceeds its maximum value, which will be used for the agent as a basic safety measure. Safety-rated reduced speed was defined earlier, and can easily be implemented into the agent as mentioned prior. An important note is that the distinction between safety-rated monitored speed and safety-rated reduced speed is that safety-rated monitored speed limit can be set to speeds greater than 0.25 m/s. A single point of control is the ability to operate a robot such that only one source of control can be used to control the robot and even the initiation of another control source will not override the control of the robot and take over it. This is relevant for the agent as it will be important to ensure that only one set of instructions is transmitted to the agent at any given time. Simultaneous motion is defined in this standard as the motion of two or more robots at the same time under the same control and which may be coordinated or synchronous. This is extremely relevant to the agent as the demonstration will be showing how the agent can collaborate and work with other agents in the area, which means understanding the principle of simultaneous motion and determining a methodology to implement it is an important factor to consider for the agent.

ISO 10218-2 refers to the safety requirements for the integration of industrial robots, which includes the design, manufacturing, installation, operation, maintenance and decommissioning, the necessary information for the aforementioned, and component devices. This is relevant to the agent as it is intended to be an industrial robot, or at the very least have some industrial applications. The agent is not intended to be a collaborative robot under the definition given in ISO 10218-2, although it could be adapted to be a collaborative robot. The definition of collaborative robot is a robot designed for direct interaction with a human within a specified space, which the agent is not being designed with that specifically in mind. A distance guard refers to a guard that does not completely enclose a zone but instead prevents or reduces access by virtue of its dimensions and distance from the danger zone. This is relevant to the agent because it details a way to help ensure the robot does not leave the zone it is operational in. A safe state is the condition of a machine or piece of equipment where it does not present an impending hazard. This definition is important to note as having a safe state for the agent will be imperative to keep it as safe as possible. The operating space of a robot is defined as the section of the restricted space in which the robot performs its tasks. The restricted space is the maximum space the robot is allowed to be in, the absolute limit on where the robot can go. This is important for the agent as the robot needs to know what the zone its aloud in is, and what the hard limits for where it can go are. A safeguarded space is defined as a space defined by the perimeter safeguarding, which is important for the agent as it will need to understand what space it has access to and what space it needs to work around to accomplish its goals. The way this can be defined for the purposes of the Hive team is by marking the area the agent is allowed in and ensuring the agent does not leave that area under its own power for any reason.

ISO 14539 refers to grasp-type grippers specifically and object handling more generally. It discusses how to classify different grippers as well as important factors to consider when designing a gripper, such as the degrees of freedom, the states of the object being effected, the amount of force applied by the gripper, the coordinate system the gripper utilizes, and the degrees of mobility. The primary way ISO 14539 classifies grippers is by breaking them into how they grasp the object, which is broken up into form closure, force closure, external, and internal grasps. A form closure grasp has the property of having zero or fewer degrees of freedom on the object when friction forces at the contact points are not being considered. A force closure grasp is very similar to a form closure grasp, with the difference being that when friction forces at the contact points are not being considered there are one or more degrees of freedom, but when the friction forces at the contact points are being considered the degrees of freedom are zero or less. An external grasp is a grasp that affects the external surface of the robot while an internal grasp is a grasp that affects the internal surface of the robot. The degrees of freedom on a grasped object are the available movements of the object when it is being grasped. The states of an object in ISO 14539 are defined as gripped, when the object is affected by the gripper but not the environment, semi-gripped, when the object is affected by the gripper and the environment, laid, when the object is affected by the environment and not the gripper, and free, when the object is not affected by the environment or the gripper. The three major types of forces for grippers are defined as contact force, which is the force at a contact point, manipulating force, which is the total sum of the vector contact forces affecting the object, and gripping force, which is the force applied when two contact forces are applied and other forces result in a net zero. The coordinate system of the robot can be with reference to many different things, such as the object being effected, the world, the gripper, the mechanical interface, or a camera. The degrees of mobility refer to how many axes a gripper has that it can move in. This is relevant to the agent as it will be moving a payload around, and understanding the different definitions related to grippers is important as moving the payload will involve using points of contact from the robot to the payload, which can technically classify as a gripper under the above definitions. This means understanding what standards are relevant to the grasping and moving of the payload are imperative to ensuring the agent is well designed. It also will allow for the agent to have what is determined by the Hive team to be the best type of gripper for the task at hand, which will help improve efficiency and overall effectiveness of the agent.

4.1.2 OSHA Applicable Standards

OSHA 1910.212 refers to machine guarding to protect people in the area of operation from nip-points, rotating parts, sparks, and any hazards caused by the point of operation. Guarding can be done in many ways, for example for electrical problems a way of guarding is simply using electrical safety devices. The methodology for guarding the agent will be to have an emergency off switch so the robot can be disabled while in operation if it needs to be able to. The attaching mechanism will also be made in a way that it does not cause unnecessary safety concerns and will be safe to be around while the robot is in operation. All nip-points and rotating parts will be guarded to ensure safety for those around or operating on the agent. This can be done by literal blocking of the nip-points and rotating parts to ensure there are no issues with safety for humans in the area of the agent. None of the requirements listed in 1910.212(a)(3)(iv) will be involved in the operation of the agent.

OSHA 1910.94 refers to ventilation. 1910.94(a)(4) referring to exhaust ventilation systems is important to keep in mind, most especially 1910.94(a)(4)(i)(a) which simply says to repair dust leaks as they are noticed. This will be important to consider for testing and demonstration purposes as if it is done in an enclosed area airflow will be important to keep in mind for the safety of all humans in the room. The agent may have a need for an exhaust system, which means this is of some importance to keep in mind. Also needed to be noted is 1910.94(a)(4)(i)(b), that static pressure drops at exhaust ducts leading from the equipment need to be checked which is important if the agent ends up having need to exhaust ducts. 1910.94(a)(5), referring to personal protection equipment will most certainly need to be referenced for the construction of the robot as the construction will utilize an abrasive wheel which will cause particles to enter the air and therefore require equipment to counteract those particles and the operator remain safe. Safety is not just a concern during the operation of the agent as there are plenty of potentially hazardous situations that can occur during the construction of the agent.

OSHA 1910.95 refers to noise safety and dictates how long a person can be hearing loud noises without personal protection equipment in the span of one day. This standard is unnecessary as the agent will not be running long enough during testing or the demonstration, nor at a volume loud enough, for personal protection equipment to be warranted or required. During the construction there will be sounds that would be classified as loud noises, but nothing for a significant enough amount of time as to require hearing protection, although it would still be advisable to use.

OSHA 1910.132 refers to general requirements for personal protection equipment. No personal protection equipment will be needed for the operation of the robot as it will be designed with safety in mind, however there will exist the need for personal protection equipment during the construction phase of the robot as many of the tools projected to be needed will be potentially hazardous and will benefit from the usage of personal protection equipment. It is important to consider safety during the construction phase of the agent as injury most certainly could occur during that time and would be extremely disappointing if it could have been avoided by using personal protection gear.

OSHA 1910.147 refers to service and maintenance of machines of which the unexpected startup or release of stored energy could be hazardous to those operating on it. This is applicable to the agent as if the robot turns on while it is being worked on there is the chance for injury simply from not expecting it. The means to fix this according to 1910.147(c)(1) is to ensure that the machine be isolated from the energy source and disabled, rendering the ability for it to have sudden energy surges a moot point. There also must be strict procedures on how to disable the machinery and separate it from the power source.There also must be lockout devices and tagout devices to ensure that the machines are properly isolated from power sources. These lockout and tagout devices should be standardized in some way across a given facility. Power will be disconnected during any operations on the agent and it will be ensured that there will be no way for power to accidentally be enabled while the agent is being worked on. The Hive team will develop rigorous procedures to ensure that everyone knows and is capable of performing the procedure of locking out the robot to properly ensure the safety of all team members or any other people operating on the agent. An easy way to ensure power is disconnected from the agent and it cannot suddenly start up is by disconnecting the battery and allowing the capacitors to drain, which would ensure the agent could not suddenly start without warning.

OSHA 1910.176 refers to the general handling of materials, which is what the agent will be doing for the demonstration, which proves the relevance of this section. Safe clearance must be allowed for any turns or passages, with these being unobstructed and it being clear on what can and cannot go through the passages. This means that physical clearances must be posted to inform people of the clearance limits as well as marking any permanent passages so it is known and they can be kept cleared at all times. 1910.176 also refers to the storage of materials, which would apply to storing the robot. When storing stackable materials it is imperative to ensure that they are safely stacked, meaning not stacked too high and interlocked so as to not fall and cause harm to anyone attempting to remove the materials from storage. The storage area should also be endeavored to be kept clean of potentially hazardous conditions, i.e. any tripping, fire, pest, environmental, or explosion hazards. This is relevant to the Hive team and the agent as if the agent is stored improperly, it could be hazardous as well as the Hive team needing to be aware of the hazards inherent in storing the agent and the assorted materials and tools used to design and construct it. Much care will be taken to ensure the safety of the Hive team as well as any sharing or accessing the storage areas for everything involved in the creation of the agent. Safety is not just a concern during construction or operation of the agent, but rather a persisting concern that always must be considered for the safety of humans in the work area.

4.1.3 ANSI Applicable Standards

ANSI/ATIS 0600003-2007 (R2012) refers to battery enclosure and rooms/areas. This includes adequate ventilation of battery generated gasses, the dissipation of battery generated heads, the control of room and enclosure temperature, the management of battery electrolyte spills, and in general the control of any contaminants within the battery room or enclosure. Ventilation of battery generated gasses is important to keep in mind when designing the agent, although it is unlikely to be a concern, as we do not wish for any gaseous buildup in the agent that may cause non optimal conditions in normal operation. Control over environment temperature will not be feasible as the intended environment for the agent is variable. Management of battery electrolyte spills will be important to keep in mind as any battery electrolyte spills will prove hazardous for both the Hive team members, as well as anyone else in the area, as well as the environment so it is critical to have procedures in place in the event that there is a battery electrolyte spill. This is relevant to the Hive team as a whole and the agent in particular as battery safety is extremely important to human health as well as the continued operation of the agent as a whole, which therefore means that we must endeavor to ensure that the battery safety is thought of as important while designing and constructing the agent.

ANSI/NECA 411-2014 refers to the installation and maintenance of uninterruptible power supplies, specifically three phase uninterruptible power supplies rated 30 kVA or more and rated 600 V or less. The definition used in the standard for uninterruptible power supplies is any solid-state power system that provides continuous regulated AC power at the output terminals, while operating from either an AC power source or from a battery system. Given that the agent is operating on a battery system as its source of power, this standard is relevant to the Hive team and is something that we can draw from to ensure the continued operations of the agent and maintaining the battery system of the agent without failure, as well as the installation. This is critical to consider as the battery system will be the only source of power for the agent, which therefore means that the continued proper operation of the battery system is pivotal to the continued proper operation of the agent itself.

ANSI C18.4-2015 refers to portable cells and batteries specifically in relation to the environment. As the agent will be using a battery to operate during its runtime, the standard is relevant and furthermore, important to know and understand for the Hive team as we deeply care about ensuring the continued functionality of the environment so we must remain diligent to ensure the agent meets the criteria specified for preserving the environment detailed in this standard. This standard covers what the relationship between battery standards and the environment is, as well as how certain battery standards can have adverse impacts on the environment. The Hive team will endeavor to ensure that we are constantly aware of the potential environmental effect the agent, and therefore its battery system, will have and will therefore endeavor to ensure that we do not follow the standards that are potentially hazardous to the environment if our research shows a large chance of following that standard leading to a hazardous environmental reaction.

ANSI/BHMA A156.2-2011 refers to bored and preassembled locks and latches, which is what our latching mechanism for the agent would fall under. This means that this standard is important to understand as the latching mechanism of the agent is critically important to the normal operation of the agent as without it, it is incapable of properly manipulating the payload which is what needs to be done for demonstration purposes. Ensuring the proper operation of the latching mechanism is critical to ensuring the proper operation of the agent as a whole.

ANSI/EIA 198 refers to ceramic dielectric capacitors, specifically classifications and testing. This is relevant to the Hive team because we will be using capacitors, and while they are minor components, they can cause failure so it will be important to choose the right capacitors based on our requirements as well as testing them to ensure they will not fail given a range of conditions that can be reasonably expected to happen.

ANSI/EIA 364-68A-2008 (R2015) refers to actuating mechanism test procedure for electrical connectors, which establishes a test method for determining the strength of an actuating mechanism of a connector release mechanism, which will be important for the Hive team because the latching mechanism will be actuated and ensuring that the actuation of it succeeds is vital to the proper operation of the agent. Testing the actuating mechanism for the latching mechanism will also assist us in understanding exactly how long and how far away the agent needs to be from the payload to determine if it will successfully manipulate the payload or not.

4.1.4 IEEE Applicable Standards

IEEE 1184-2006 refers to batteries for uninterruptible power supply systems. The agent will not exactly be an uninterruptible power supply system, but while it is in operation it will be disadvantageous to have any complications arise from the battery. An uninterruptible power supply system is one that runs on batteries or capacitors to allow a main power source to start back up, typically only lasting a few minutes in duration. As the agent will run solely on a battery system, there is relevancies in this standard, which allows for us to draw from this standard and therefore the knowledge accumulated by past generations and allow the agent to operate efficiently and not have to worry about surges or anything that may interrupt the battery system from operating as intended.

4.1.5 UL Applicable Standards

UL 2595-2015 refers to general requirements for battery-powered appliances. The agent is not an appliance, obviously, but it is battery powered which means there is some overlap that is applicable to our group and therefore the agent. Having an understanding of the standards surrounding specifically battery powered appliances can help us further understand the complexities and different viewpoints behind general battery powered mechanisms which will allow us to be able to draw from those different viewpoints and construct the best battery system we can for the agent and ensure that it operates properly and under proper operating conditions at all times.

UL 2595-2015(a) refers to safety for general requirements for battery-powered appliances. As mentioned before, the agent is not an appliance but is battery powered so there are some common factors that can be used to assist in developing safety mechanisms for the battery of the agent and effectively developing the battery system of the agent. Safety is a key factor in our goals, so understanding a standard for safety of battery-powered appliances will assist us in understanding safety for our agent, which will be battery powered and therefore requires some safety measures. There are other standards from other standardizations that we are also referring back to to ensure that we design and construct the agent such that it operates safely under all reasonable conditions it will experience.

4.2 Constraints

Hive: The Grounded Swarm has various constraints that must be addressed for the system to be fully operational and function as intended.

4.2.1 Economic Constraints

The cost of designing and manufacturing the robot is an important factor to consider for this project, as there is a budget allocated by the sponsor for five hundred dollars. Given that multiple agents are being produced to use in tandem, it gives a limit on how much each agent can cost along with limiting how much prototyping can be done to ensure the design is valid and will work as intended. The sponsor for this project is Capacitech Energy, and one requirement is that supercapacitors produced by Capacitech Energy must be used in the project and shown to improve the efficiency of the battery. The bill of materials also must be approved by the sponsor prior to any parts being purchased, which means much research is needed to ensure that the best components are selected and any prototyping that is wanted will need to be meticulously thought out in order to keep costs down as much as reasonably possible.

Another economic constraint is that the manufacturing cost per robot should be kept down as much as possible for marketing potential. The lower the manufacturing cost, the lower the agent can be sold per unit for the same profit, or the more profit can be made from selling it at a higher price per unit. As the agents are meant to coordinate with each other, keeping the manufacturing cost down also allows for more agents to be coordinating on the same job for the same price. For this purpose, a balance between the price of components and the quality of the components must be struck, where the price is relatively low while not overly sacrificing the quality of the components and therefore, the functionality of the robot. The goal is to keep the cost of construction per agent lower than $200 as that will allow for the construction of two agents, which is the goal to be able to show coordination between the agents.

4.2.2 Time Constraints

The time allocated to completing the project and building the fully functional robot can be considered as the most important constraint for this project, as it has to be done by the end of Senior Design 2 and there is no flexibility with regards to completing it later than that. This means that there is approximately two semesters, or nine months, until the final deliverable is due. Many things that were considered for the robot are now secondary goals, for example, the primary goal of modularity will not be able to be fully explored as designing and constructing a functional robot that can perform a demonstration adequately is the most important thing to do with the given time, so while performing the demonstration will include some modularity as the robot will need modules to be able to do the demonstration, many other potential modules will unfortunately have to be left to the wayside. For example, different actuators for different types of payloads will not be designed and constructed as ensuring that the actuator for the payload being used in the demonstration is more important as it will actually be used for the demonstrable product. The agent needs to above all be capable of performing testing and then properly performing the demonstration under normal operating conditions by the deadline we have.

Another important factor to consider is the length of time it may take to get the PCBs in, as may popular components and suppliers have supply chain issues currently, so ensuring that the components used in the PCB are in stock and available for consumer purchase is going to be important especially as throughout the prototyping process there will most likely be many interactions of the PCB so there will be multiple orders interspersed throughout the time, which may cause issues. There also exists the issue of print time for any 3D printed parts, as the majority of parts that can be 3D printed have to be assembled after being printed, which greatly slows the rate at which pieces can be printed. Another issue with 3D printing is that if a print fails, or the printer filament needs to be changed, or the piece is incompatible with the rest of the agent and needs to be modified; the printer needs to be recalibrated and the print therefore must be restarted. These time related issues do not account for the innate time to print required for the components being printed as the time to print scales with size and number of parts to print. Also to be noted for 3D printing parts is that typically 3D printing a single copy of the piece to be printed along with the other components for the mechanism it makes up to produce one full unit of whatever the mechanism is, to ensure that everything fits together and works properly before duplicating the parts if multiple of the mechanisms are required for the proper operation of the agent.

4.2.3 Environmental Constraints

The environment in which the robot is located will matter a significant amount as the agent needs to be able to navigate its environment and manipulate the payload, which means where testing and usage of the agent is done will matter. Ideally there will be no change in terrain or any rough terrain for initial testing. In terms of the environment as a whole, the battery that will be used is not good as it has harmful chemicals, but that is only an issue if the battery leaks. Aside from catastrophic failure of the agent, the agent will not have major effects on the environment as a whole. There are the components used to make the agent, and the tools that will be used to construct the agent and their respective impacts on the environment. The tools used in the construction process of the agent will have minimal impacts on the environment aside from contributing to entropy and the byproducts of how we as humans inherently generate power in the form of electricity. The production and manufacturing of the parts we will use for the agent have negative consequences for the environment, but we as the Hive team will endeavor to keep our use of said parts as safe for the environment as possible and also we will endeavor to choose parts that are produced and manufactured by environmentally friendly companies whenever it is feasible.

4.2.4 Social Constraints

The goal of the agent is to ultimately create a modular robot that can be adapted for many different robotics uses. For this to be accomplished appropriately, it needs to be capable of being adapted into many different uses, and it must be done easily. The primary reason the method of adapting the robot should be capable of being done easily is twofold. Firstly, easier implementation of adjusting the purpose and function of the robot by replacing modules allows for more adaptability in its modularity as more people will be able to use and design modules around this basis. Secondly, it being easier to use allows for a simpler time of implementing the methodology for adaptability through modularity by allowing the robot to be simplistic to make and equally simplistic to use. The adaptability through modularity being easy to use will also open up access to more people to be able to use it which will allow it to have a broader impact, as many people are enamored by convenience and will do whatever is the most convenient. As the adaptability through modularity will be easy to use as well as easy to understand, it will be convenient to use and create more modules and therefore more accessible.

4.2.5 Political Constraints

There are no political constraints that affect the design or production of the agent as it does not fall under any definitions or requirements that are currently bound by any laws or other political standings in either the United States of America, which is where the entirety of the Hive team resides, or internationally.

4.2.6 Ethical Constraints

The Hive team will endeavor to ensure that the agent does not have any base malicious uses, however given the aspect of modularity that is desired, there is the chance the agent could be used maliciously. For example, the aspect of payload manipulation that will be demonstrated could easily be used to transport nefarious goods, such as any type of contraband, or even move bombs into undesirable locations. Obviously that is not an intended or even recommended use of the agent, however the existence of any payload moving robot comes with those same risks. The modularity also poses some ethical dilemmas as a robot that can be adapted to anything is great for having that capability and being able to do many good things but it equally has the opportunity to be adapted to do many bad things as well. After all, technology is neither good nor bad, rather it all depends on the user. There will be safety features implemented to ensure that the agent does not harm any humans in the area around it during its operations as the safety of humans is a goal of critical importance to the Hive team as a whole.

There will be no skimming out on the production of the agent, the primary goals that were set will be met, without a desire for money saving. Obviously when money can be saved without compromising the functionality of the agent it will, but if it comes down to saving money or having a fully functional robot, the agent team will choose to have the robot be functional. The agent will not endanger any humans during its operations and will also have a status light to show when it is in operation and when it is not. The agent will also follow the three laws of robotics, which are: A robot may not injure a human being or through inaction allow a human being to come to harm, a robot must obey orders given to it by human beings except where such orders would conflict with the First Law, and a robot must protect its own existence as long as such protection does not conflict with the First or Second Law. These are Isaac Asimo's Three Laws, which were originally made for science fiction purposes but impact thought on the ethics of artificial intelligence, so they do deserve some mention as the agent will be following them. Modern day ethics for robots and AI are highly debated, so the Hive team will be leaning on the side of caution to ensure we maintain a good level of ethical responsibility amongst ourselves.

4.2.7 Health and Safety Constraints

The agent will endeavor to avoid obstacles as it carries out its duties, which obviously includes humans. This will ideally avoid the situation in which the robot hits someone in the legs or runs a person's foot over, which is the most of an issue the agent will have in terms of safety in general. If this situation does occur, the agent will have a low weight so it will not cause any major injuries, as well as the fact that the wheels will not be able to go over any close-toed shoes, which adds to the safety of the agent. The agent will also have an emergency off switch that can be flicked if need be to shut down the robot posthaste. Safety is a big concern for the agent, which is why there will be the emergency stop button as the safety of people around the agent is extremely important. There will be no materials that are hazardous to one's health used in the production of this robot aside from batteries, but those are safely contained in the batteries. No harmful chemicals are needed, and all electrical components will be out of the way so there will be a massively reduced risk of someone getting accidentally electrocuted or causing an electrical fire, as both would greatly endanger the safety of everyone around along with causing a catastrophic failure of the agent, which would mean the task the robot was performing would be stopped, and here that might not be as big of of a deal as if it was performing more delicate tasks. There will be safety features implemented into the agent to ensure the safety of all humans in the operational area while the agent is under normal operating conditions. The Hive team deeply cares for human safety, which is why this is an important constraint to keep in mind during the design and construction of the agent.

4.2.8 Manufacturability Constraints

The primary manufacturability constraint is availability and timing. In order to ensure that the robot works through the various stages of prototyping, the PCBs will need to be ordered well in advance along with ensuring all components on said PCBs are in stock and available for consumer purchase. The time it will take for the PCBs to arrive is also an important note as the robot will be unable to function without a PCB. The sensors required will also be important to get as without the sensors, the robot will not be able to function properly and gather the information it needs to properly make decisions, so the sensors required will need to be in stock or other sensors will have to be considered. The motors are the other important part of ensuring the robot runs properly as without them the robot will not have any means of locomotion and will thus be unable to move. Therefore, the motors will need to be selected such that they are both in stock and available for consumer purchase as well as capable of providing the robot with enough torque to be able to provide the robot and payload combination with locomotion, as the robot does need to be able to move the payload. This is a physical constraint for the robot, as the robot must be able to physically move to be able to perform the tasks required. Another consideration is that all parts must work together for the end result of the agent functioning properly because all of the parts will need to be used together, working simultaneously to ensure the agent can operate properly and produce the desired end goal.

4.2.9 Sustainability Constraints

The sustainability goal of the agent is to ensure half an hour of continuous use at base so that the demonstration can be performed without issue. This includes ensuring that the robot does not suffer from random power surges that might kill the battery as well as ensuring that the battery can last the full half hour of continuous use. Another goal is to ensure that the agent can be used multiple times, as in it does not tear itself apart while moving or manipulating the payload. This is a constraint on how much torque at a maximum can be used as well as on the body of the agent as it needs to be able to withstand a fair amount of torque, especially with the goal of adaptability through modularity as the torque required to manipulate this specific payload is not going to be the same as the torque required for other certain use cases the robot may have. Given how the goal is adaptability through modularity, sustainability is a very important goal as ideally the agent will be mass produced and will be capable of being adapted into many different forms of expression and will therefore need to be able to have a production value capable of matching the potential demand and will need to be capable of lasting through any usage of the agent that may arise.

4.2.10 Testing/Presentation Constraints

While testing, the agent will need to be in an area that it can see, which means that if the area testing is done in changes, or is moved, then it will need to be recalibrated. Given the relative size of the robot and its wheels, the ground testing is done on, will need to be fairly smooth so that the robot does not get stuck, so the ground will need to be relatively level as well as smooth. Similarly, the ground will need to be free of environmental hazards such as mud that might cause the robot’s wheels to get stuck and the robot will be left unable to move. It is likely that these will be met by the nature of the planned testing, which will be indoors, however this planned nature has additional constraints that must be addressed. While indoors, all walls must be accounted for along with any obstacles that may litter the floor. Also to note is that indoor use is what the agent will be intended for, which means that ensuring the agent is operational in said conditions and is tested in said conditions is something imperative to ensuring the agent's success.

The presentation will need to be done somewhere that has already been calibrated for, which means that prior to presenting calibration will need to be done to ensure the agent can see the full field and be able to path plan with the given information. A full scale test presumably will be reasonable for presentation as it will take place in a ten foot by ten foot area, which will include the robot and the payload to be moved. The space will need to be defined in terms of a restricted space, which details the exact bounds of the field the robot can move in with the robot not being able to move out of this restricted space, as well as an operational space within the restricted space that details where the robot will be performing its tasks. The agent will need to be capable of running for the entire duration of the presentation, which is assumed to be approximately 30 minutes, so there will need to be enough power and therefore a large enough power source to provide power for that long.

5.0 Project Hardware and Software Design

This section discusses the design of the agent. The section initially delves into generalizations for each sub system, then focusing into the decisions made for the first prototype. Following this, we discuss the plans for the final agent.

5.1 Subsystems

In this section, we will present the application design of the modular robotic system with detailed descriptions of many necessary subsystems and their requirements. To begin finding the requirements for our project we must first look back onto the goals of our project. Our first main goal is to create a robotic system which can be used and adapted for many situations to close the gap on the need to create a unique robotic solution for every situation. To create a robotic solution that is best suited for every situation is to allow for great flexibility in the workplace. To do this we first analyzed the commonalities that are most frequently occurring between recently made robotic solutions on the market. Through our research we found that most solutions use a power source, locomotion, manipulation, and a processor. This is where we began our development.

Everything within a robot branches from its power source. The power source is the heart beat of the robot and without power a robot cannot function. Thus, most robot designs begin by deciding on a power source most suited for their required robotic solution. Some devices rely on a tethered solution which employs a stationary power source with cables running the distance to the robot, while others rely on a rechargeable battery as their source of energy. Either solution has their positives and negatives and for our first prototype we decided upon a tethered solution and for our second prototype a battery solution.

Following the power source, we found it is necessary to have a power distribution system to convert the unorganized energy from the power source to complementary configurations to run the various devices housed within the robot. This is where our modularity requirement begins taking effect as there are many different common voltages which are used for various tasks. Low voltages are used for lower power consumption devices while higher voltages are for more power intensive tasks. Voltage configurations included the 3.3 V, 5 V, and 12 V configurations. Most microcontroller devices and sensors range in either the 3.3V or 5 V range, while high power devices (Motors, electromagnets, etc) commonly run in the 12 V range and higher.

After deciding upon the power source and finding a way to distribute the power to all devices in the system, it is necessary to figure out what will run the logic of the robot. When choosing a processor there are many considerations to make including, programming language, processing capabilities, RAM, number of IO ports, etc. With the processor being the brain of the system, there are many requirements and specifications that need to be explored to choose and create a successful processor solution.

Next is a method of locomotion. Locomotion is how the robot will move to transport goods or to be present at multiple locations to perform tasks. Our team went to great lengths to find a movement method most suitable for our modular robotic solution. Including 2 wheel drive, rear wheel drive, skidding, drifting, angular driving, etc. We decided to go with a design centered around using mecanum wheels.

Lastly is manipulation. Manipulation is the method of interacting with the environment. Manipulation depends on the objective of the task but can include methods of gripping, pushing, pulling, lifting, or etc. Our method of manipulation went through many iterations. Starting from lifting payloads, encompassing the payload (more vividly understood as “hugging” the payload), our last and current design is utilizing a system to latch onto a protruding bar along the outer edge of the payload.

5.1.1 Power Distribution

The job of the power distribution block is to take the fluctuating input power of the battery and convert it into easily accessible and stable sources of energy. Methods can include voltage regulators, buck converters, boost converters, and popular connectors to be able to access this energy.

5.1.1.1 Power Distribution Requirements

As we did in previous sections, to understand what is necessary in our power distribution design, we must first understand the goals of our project. As stated in the past, the most prevalent motivation is modularity and adaptability. How can we create a modular energy distribution system system that allows for easy adoption into robotic solutions and allows the ease of designing new modules. Our second motivation is cost. We want our agents to be as inexpensive as possible for our team to develop as we are limited on funding. These motivations bring up many different requirements to discuss, starting from voltages configurations.

Voltage Configurations: The easiest way to allow modularity is to give options, the most common voltage configurations are the 3.3 V, 5 V and 12 V options. The 3.3 volt configuration is most commonly suited to run low power solutions like microcontrollers and sensors. Next is the 5 V solution which is used to power higher wattage controllers like the Arduino Uno, high power sensors like ultrasonic sensors, and many other options. Lastly, the 12 V configuration is best suited for high power applications, like driving motors, electromagnets, and more.

Connectors: Another way to add modularity into our system is to use common connectors and have them be easily accessible. Using common connectors allows for the most acceptance by future module designers, aiding in the motivation of adaptability and accessibility.. Another benefit with using common connectors is that it allows us to cut down on development time because the process of being accepted as a common connector confirms the integrity and durability of the connector. If there were to be issues with the connector, it would have been pointed out and fixed well before it became a common connector. Also aiding in accessibility would be to have the connectors on the outside of the device. Having a port on the outside of the robot rather than needing to access the internals will help with fast development as assembly and disassembly is minimized for our team and future module designers.

Easy Development: The easier to implement and develop the better. As with this project we are limited on development time and funding, so the less mistakes made and the fewer prototypes that are required for a finished product, the better.

Voltage Stability: A well made distribution system adds stability to power rails. Voltage instability is detrimental to most devices' integrity. For example, if a voltage spike were to occur on any power rail, devices are likely to be damaged due to the overvoltage. If the opposite occurs and a dip in voltage occurs, then devices can turn off and cease to function causing a failure in the system. As such, the less voltage fluctuations there are in a system, the more stable the components that draw power from the distribution system will be.

5.1.2 Chassis

The chassis is the main body of the agent, it ties all the subsystems together to create a functional agent. The chassis will define the overall footprint of the final product. Aside from space the next most important aspect of the chassis is its stiffness, a chassis with great amounts of deformation either within the main structure or even just relative to each other piece could lead to catastrophic failure of the project. Ultimately, the chassis needs to be able to withstand all normal loadings as well as impact loading while also accommodating the other major components of the project. One final note, the more rigid the chassis is, will allow the best performance of the drivetrain.

5.1.3 Drivetrain

The drivetrain is the next major subsystem aside from the chassis as it will dictate how the agent will move through our environment. The drivetrain includes, the 4 DC motors, 4 mecanum wheels, and the chassis to motor interface. With one of the major requirements of our project needing to be able to move a payload, it is crucial that the mecanum wheels can produce enough friction under loading and they must add the ability of strafing to the robot. Tuning the overall diameter of this wheel will provide different amounts of speed and torque. On the other hand, different motors could be tested at a much greater cost.

5.1.4 Microprocessor

Firstly, the microcontroller is responsible for receiving and processing data from the server. This includes instructions on where to move, when to operate the attachment mechanism, and any other relevant information necessary for the robot to operate effectively. The microcontroller must be able to interpret this data accurately and quickly in order to make decisions. The microcontroller is also responsible for managing motor drivers on the robot. These drivers control the speed and direction of the motors that power the robot's wheels. The microcontroller will adjust these drivers based on the input it receives from the server or other sensors on the robot. The microcontroller is likewise responsible for managing modular attachment ports on the robot. These ports allow for additional sensors or tools to be added to the robot as needed. The microcontroller must be able to detect when a new attachment has been added or removed and adjust its functionality accordingly. The microcontroller is also the means of interpreting sensor data, namely from the attachments. These sensors are used to detect when certain parameters have been reached or exceeded, such as the limit switches that will detect when the latching mechanism has connected to the payload. The microcontroller must be able to detect when these conditions have been reached and adjust its behavior accordingly. Lastly, the microcontroller is responsible for managing an IMU on the robot. This sensor provides information about the robot's orientation and movement in space. The microcontroller must be able to interpret this data accurately in order to control the robot's movements effectively. The feedback loop for this will be local, meaning the IMU data will be used by the MCU to correct itself rather than sending data back to the server for the algorithm.

5.1.5 Sensors

A limit switch will be placed on the front of the attachment mechanism to detect once the payload has been secured. The limit switch will sit on the edge of the latch, pushing against a metal bar fixed to the payload once attached. This is a simple binary means of ensuring that the payload has actually been connected to the latching mechanism.

5.1.6 Attachment Ports

In order to support the modularity of the robot, the agent will feature attachment ports on each wall of the chassis. As described earlier, attachments can include sensors, grippers, cameras, and other devices that enable the robot to perform specific tasks. The ports need a variety of voltage buses to have the ability to provide power to a diverse range of attachments, each of which may require a different input. Without these, each attachment would require its own power source, adding weight and complexity to the robot. Extra GPIO pins are also necessary for modular attachments. These pins allow the microcontroller unit to communicate with the attachment and control its functions. For example, the latching attachment will require a PWM signal in order to operate the servo. To create the most adaptable solution without sacrificing space and complexity, each side of the robot will be equipped with a 3.3 V, 5 V, 12 V, and a ground port, as well as 3 GPIO ports, directly connected to the microcontroller unit. As shown below in Figure 5.1, each port will take lines directly from the voltage regulators on the main PCB. The cabling will be soldered directly into holes placed on the main PCB, and will lead to pin headers on each port so that attachments can easily be plugged in. The cabling for each port will be routed through the grooves of the aluminum extrusions. This modularity allows for greater flexibility in the robot's capabilities without requiring significant changes to the overall design.for each port will be routed through the grooves of the aluminum extrusions.

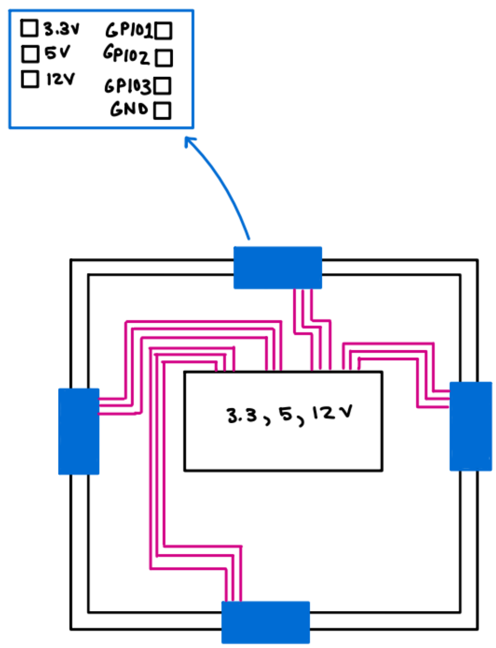

Figure 5.1: Attachment port layout. Designed by Benjamin Palladino.

5.1.6.1 Latching

In order for the agent to be capable of manipulating the payload it will latch to the payload with the gate latch selected in **Section 3.5.4.** In order to ensure that the latch does not open once the agent begins to move, the locking mechanism described in **Section 3.4.8** will be used to ensure that the latch stays closed during operation.

5.2 Prototype 1

This section covers the design and construction of our first prototype. Prototype 1 was designed quickly to confirm the possibility of our designs and to quickly find any faults in judgment that cannot be found when only designing. Other reasons include time-effectiveness of purchasing components early to negate the possibility of shipping delays and to do in-house testing when details may not be included on datasheets, but prototype 1 mainly serves as a proof-of-concept for our drivetrain with our custom mecanum wheels. This prototype can be seen below in Figure 5.2.

Figure 5.2: Prototype 1 Using a PC PSU, Arduino Mega 2560, a Lynxmotion PS2 V4 Controller, Greartisan 12V DC Motors, and Custom 3D Printed Mecanum Wheels
Image taken by Cameron Nichols

5.2.1 Mechanical

After determining that it would not be cost effective to buy manufactured wheels, our only other choice was to design and manufacture them ourselves. With a 3D printer being easily accessible to the team this was the obvious choice for manufacturing. This ease of access is not the only reason for selecting this method of manufacturing however. Using the 3D printer, the team was capable of producing verifiable prototypes that could then be iterated later in the design process.

Considering that we will be moving a payload effectively by pushing or pulling it with the body of the agent, the first major design challenge was figuring out how to produce enough friction to drive both the agent and payload around the environment. Due to the capabilities of our manufacturing operation, the two main materials that could be printed were PLA and TPU. While this may seem daunting it was not a major problem to overcome. PLA is a hard plastic material that when printed properly creates very smooth surfaces. TPU is quite the opposite in terms of hardness, it is closer to rubber in terms of malleability, however the friction coefficient is much lower than rubber. With these two materials in mind, the next step in the process was to develop an understanding of how a mecanum wheel achieves its movement.

5.2.1.1 Mecanum Wheels

As previously stated, a mecanum wheel has rollers around the perimeter of the wheel rather than your traditional wheel. Our wheel came in at 60mm tall using 8 rollers, These rollers are tilted at a 45 degree angle changing the direction of the force provided from the friction of the roller. Out of the two possible materials, TPU was chosen as it would provide more friction. Its high resistance to abrasion would allow the material to deform as loads are added to the agent, this would result in a greater contact patch with the surface. To increase the friction of the coefficient, the team is currently testing several options. The first being simply coating the rollers to see how the material may change, another being simply adding a wrapping around them (such as grip tape). It is important to note that this could be done by adding something similar to our surface, however, this would be a last resort option. Finally, if we were able to print with TPE (a subset of TPU) it is known to have a higher friction coefficient and acts more like a rubber-like material. Pictured below is the initial design of the mecanum wheel.

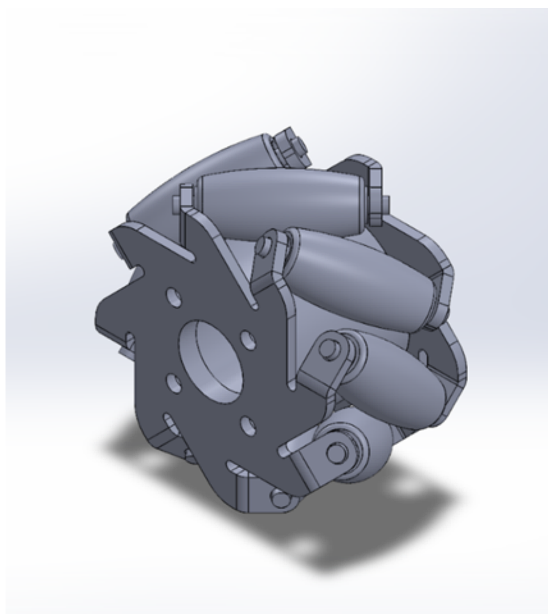

Figure 5.3: CAD Model of Mecanum Wheel by John McClain

While focused on the design of the mecanum wheel, a very important part of this process was manufacturing and material selection. As stated above, the wheels are made from PLA and TPU, two rather common thermoplastics used in 3D printing. Considering that each of the materials selected has different properties, they both have different allowable tolerances that must be considered. Throughout the concept development of these wheels, each part of the mecanum wheel assembly was first conceptualized in the scope of the overall assembly and then TPU and PLA were considered. As mentioned before the main part that utilizes TPU is the roller that will produce the friction we need to drive forward. This then slides onto a PLA bushing that would allow the roller to spin about its axis freely. For this PLA to TPU interaction, we found that a tolerance between 0 mm-0.25 mm was acceptable and in some cases, a negative tolerance was used to ensure that TPU had a snug enough fit to remain static relative to the PLA part. As for PLA to PLA interaction, a tolerance of 0.5 mm was used all throughout the manufacturing process.

5.2.1.2 Motors

After properly assembling the mecanum wheels the next order of business was to attach them to the Greartisan 12V DC motor. This motor comes equipped with a 3 mm D-shaped shaft, making it to where the wheel cannot rotate in the radial direction. In this case, the wheel can still slide off of the shaft in the axial direction. To eliminate this a hub was designed with a threaded hole, this hole allows for a set screw to be threaded into the shaft locking it in place allowing for no axial motion to occur.

5.2.1.3 Frame Assembly

Stepping away from wheel design, the next step of the prototype was building the frame that would pull everything together. To create the frame, we opted to use four 203.2mm (8 inch) pieces of 20mmx20mm 6063-T5 aluminum extrusion that would then be held together with the appropriate 20mmx20mm gussets. With a yield stress of 145MPa and an ultimate tensile stress of 186MPa, this will serve us greatly. The main concern is potential collision with other agents and payloads. With a proposed movement speed of under one mile per hour this frame will be able to withstand even the worst predicted crash, especially after bumpers and sensors are added to help aid in collision avoidance. This design works best as it will be quick to create and is lightweight and will provide more than enough strength to carry our testing electronics and test loads. With the plans of implementing attachments to the frame the agents, this works well, as extrusion is very easy to work with simple hardware such as bolts and t nuts, the design possibilities are endless. Doing this will allow for cheap sound testing of our wheels and the strafing concept. I would also like to note that if this initial design works it will be very easy to build off; this is especially important since the final project will require multiple agents to be assembled for full proof of concept.

Finally, aside from mecanum wheel validation, this prototype will also allow our counterparts on the coding team to begin testing their algorithms for the path planning capabilities the agent will need to navigate the test environment. Pictured below is the CAD model of the first prototype. On a final note, a piece of plastic cardboard was cut and added to the robot for all of the prototype’s electronics.

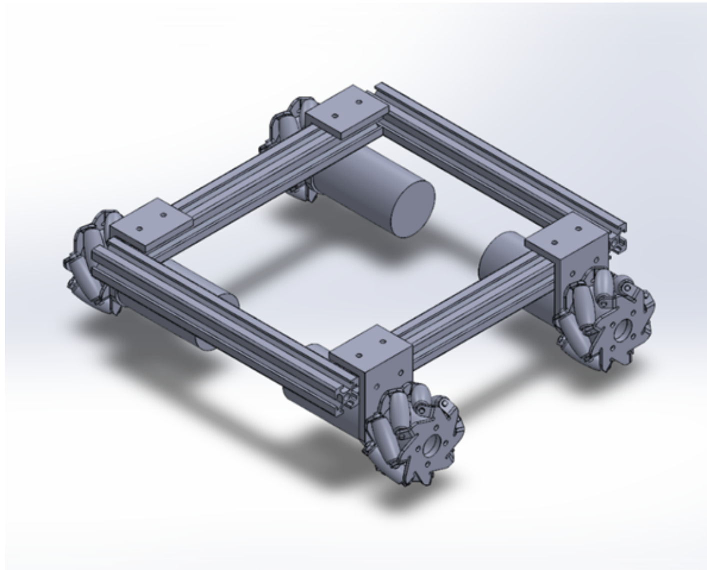

Figure 5.4: CAD model of first prototype drivetrain by John McClain

One final note, with the proposed drivetrain and frame layout, the center of mass will be very low from the start. This will allow for a good base to build off of. This is important, as an agent tipping over is quite suboptimal.

5.2.1.4 Prototype 1 Troubleshooting

Once the initial prototype was constructed and wired, we began to attempt to drive the agent around. Everything was working as designed except for a very vital component of the mecanum wheels. The first issue was the rotation of the extrusion about the gusset. The gusset alone has some tolerance in the tab, making it to where the entire beam would rotate upon the loading of the wheels. This is a problem, for best performance we need the contact patch of the wheel to be normal to the ground, this is important as it will ensure that the friction force generated by the wheel is directed in the desired direction, leading to the best driving characteristics of the drivetrain. To fix this we changed the design of our motor mount to a two-piece design, by doing this we added more rigidity, not allowing the motor to rotate relative to the chassis. Finally, more pieces were added alongside the gussets to ensure that the chassis does not flex under normal loading conditions. Doing this did improve the driving of the agent, however, the main issue was still present, the agent was not strafing as intended. After some trouble shooting, we believe that the issue comes from clearance issues, ultimately, the frames along with the caps were hitting the ground drastically affecting the amount of friction that the rollers could produce. This was caused by lack of proper clearance; in the CAD model the roller was confirmed to hit first as the wheel rotated. The issue arises with implementing manufacturing tolerances, once the final model was constructed, under loading, the tolerances caused the roller to be pushed up past the frames. In doing so, the required contact patch was not made leading to drivability issues.

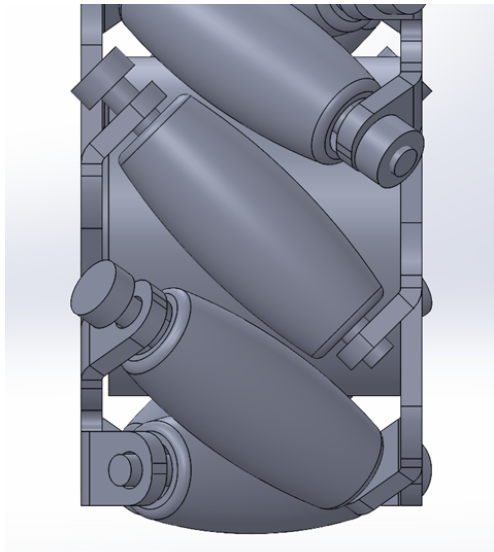

Figure 5.5: CAD Model of Mecanum Wheel Interference Inspection by John McClain

After analyzing the initial prototype of the wheels, a new design was iterated, and the following changes were made. First, the bushing tolerances were tightened to alleviate the sagging issue, next the caps on the end of the rod and the friction fit caps were printed at a smaller diameter to eliminate interference with the ground. The final design change was the implementation of chassis stiffeners. In initial testing it was found that even though the motor was not moving relative to the extrusion, the extrusion itself would move causing the wheels to have unwanted camber, these pieces were implemented to eliminate this and ensure that the contact patch is exactly normal to the ground.

5.2.2 Electrical

This section will go over the electrical system for our first prototype. The goal for our first prototype is to quickly find faults in our system's design and to quickly iterate to reach Hive's goals. This first prototype will be a very rough iteration of our design as using products that are on hand for the sake of speed of implementation.

5.2.2.1 Microcontroller

The microcontroller board chosen for our first prototype is the Arduino Mega 2560 Rev3, which is based on the ATmega2560 microcontroller. This board was chosen because it is simple to implement, and we already had a few in our possession. Although this microcontroller differs from the one that we will be using on our final product, it is sufficient for our first prototype since we will not require Wi-Fi and are primarily testing our drivetrain system, which just requires motor control and the ability to receive input in order to drive.

5.2.2.2 Motor Controller

The motor controller chosen for our first prototype is the Texas Instruments L293D. This motor controller was chosen because we already had at least two in our possession, and while this differs from the motor controllers we will have chosen for our final product, they will suffice for testing of our drivetrain. The L293D is a quadruple half-H driver which can drive inductive loads such as relays, solenoids, DC and bipolar stepping motors. We focused on the DC motor control application, and found that this driver can control the speed and direction of two DC motors with an output current of up to 600 mA per channel. Since we will be independently driving four motors, we used two of these drivers. We connected these drivers to be controlled using the sign-magnitude mode, which is described within **Section 3.3.5**. Supplying these drivers with a voltage of around 12.3 V from the PSU resulted in the motors being supplied with around 10.5 V, which while having a high voltage drop of about 16%, is sufficient for testing purposes.

5.2.2.3 Inertial Measurement Unit

The inertial measurement unit (IMU) chosen for our first prototype is the InvenSense MPU-9255. This IMU was chosen because, much like the other components, was already in our possession. While this differs from the IMU that we have chosen for our final product, the accuracy and noise levels are quite acceptable for prototyping purposes. Using this IMU to perform angle tracking showed some improvements to the strafing ability of the prototype, though the strafing is still far from perfect.

5.2.2.4 Power Supply

Due to the fact that we did not yet acquire the voltage regulators and batteries for the agent, we needed a power supply capable of providing several different voltages and capable of delivering a maximum of 2.4 A to the motors. We made use of a recycled PC power supply, which featured 3.3 V, 5 V, and 12 V leads, capable of powering all our onboard hardware. The major limitation of taking this shortcut is the fact the power supply unit requires a 120 VAC connection, meaning it needs to be tethered to a wall outlet. This severely limits the range of the robot, but because the primary goal of prototype one was to test the drivetrain functionality and mecanum wheels, it will not significantly affect its performance. This is only a temporary measure, and the second prototype involves a finalized power supply consisting of a battery, voltage regulators, supercapacitors, and other filtering and safety mechanisms.

5.2.3 Software

This section will go over the software that we developed and used for our first prototype. Since the main purpose of this prototype is to test our drivetrain, the software is relatively simple and is broken into five different files: AgentControl.ino, Motor.h, Motor.cpp, Mecanum.h, and Mecanum.cpp.

5.2.3.1 Main Control

The main file, AgentControl.ino, is an Arduino sketch file and utilizes the other files to control the agent prototype. Apart from the files that were developed by a member on our team, Cameron Nichols, we also utilized a library for interfacing a PS2 controller for Arduino control (madsci1016). A library is also included for interfacing with the MPU-9255. The ArduinoControl.ino file begins by including references to the aforementioned libraries and our Mecanum.h file, and then defines the pins that we are using to control the speed and direction of each of our four motors as well as the pins used to interface with the PS2 controller breakout board. Next, the four motors we will be using are initialized by calling the constructor from the Motor.h file, and the mecanum drivetrain is initialized by calling the constructor from the Mecanum.h file. The code then enters the *setup()* function, where serial communication is established for telemetry and debugging purposes, the PS2 controller is configured, and the function *calculateIMUError()* is called.

The *calculateIMUError()* function begins by printing some messages that state that calibration is about to start, and that the IMU should be stationary. After this, the code enters a loop and reads all the gyroscope and accelerometer values every 4 milliseconds for a total of 1,250 iterations, or 5 seconds. These values are continuously added up, and are used to determine the error of the sensors. For the acceleration on the Z-axis, 9.81 meters per second squared is subtracted from the error each iteration, as it is expected to experience acceleration due to Earth's gravitational pull. After this, all the errors are divided by the total number of iterations, 1,250, which will give the average. These values are used later. After this function executes from the call within the *setup()* function, control of the code goes to the *loop()* function.

The *loop()* function will execute and will remain running while the Arduino Mega 2560 remains powered. This function begins by calling *updateAngles()* function, which will be discussed later, and then the *read_gamepad()* function from the PS2 controller library, which updates the values for all the buttons and analog joysticks on the controller. We decided to use the left joystick on the PS2 controller to control the strafing and forward motion of the agent prototype using the X- and Y-axes, respectively, and decided to use the X-axis of the right joystick to control turning. We also decided to use the X- and Y-axis of the right joystick to control angle tracking based on the IMU, but only if the right bumper on the controller is pressed. The PS2 controller library will return a value of 0 to 255 for each of the axes, where a value of 127 is the midpoint and means that the joystick is centered and in its default position. After the *read_gamepad()* function is called, the values from the four aforementioned axes are scaled to a range of -1.0 to 1.0. We decided that it would be useful for testing purposes to include a way to instruct the agent to only go forward, backward, or strafe, without the possibility of combining these different directions, which can happen when using the analog joysticks. Thus, we decided to utilize the D-Pad on the controller to do just that. After the joystick values are scaled, the code then enters an area where the D-Pad values are read and two values are updated based on the status of the D-Pad. The variable *pad_y* will be set to 1.0 or -1.0 if the D-Pad up button or down button is pressed, respectively. The variable *pad_x* will be set to 1.0 or -1.0 if the D-Pad right button or left button is pressed, respectively. Next is a conditional statement, which if the select button on the controller is pressed, then the gyroscope Z angle is reset to 0 degrees.

Next in the *loop()* function, there is another conditional statement, which says that if the right bumper on the PS2 controller is not pressed, then the variable *turn* should be equal to the X-axis of the right joystick. Otherwise, if the right bumper is pressed and one of the right joystick's axes have a value greater than 0.2, then the *turn* variable is set to the return value of the *getTurnCmd()* function, which is passed an argument of the *getAngleFromJoystick()* function, which takes in the two axes from the right joystick as arguments. The *getAngleFromJoystick()* function performs some basic trigonometry to get the angle that the joystick is at, where straight up is 0 degrees, and straight down is +/- 180 degrees. The *getTurnCmd()* function returns the error between the current angle of the IMU and the desired angle from the *getAngleFromJoystick()* function. The *turn* variable is then scaled and constrained so that an error greater than or equal to 15 degrees will result in 100% turning speed. After this is a conditional statement where if either *pad_x* or *pad_y* is nonzero, then the *Mecanum.drive()* function is called by passing in the variables *pad_y*, *pad_x*, and the *turn* variable. If both *pad_x* and *pad_y* are zero, then the *Mecanum.drive()* function is called by passing in the scaled values of the left joystick's two axes and the *turn* variable. It is worth noting that the Y-axis value is the first argument to the function while the X-axis is the second argument, and while this might seem counterintuitive, there is a reason and this will be discussed with the Mecanum.h and Mecanum.cpp files.

The *updateAngles()* function that was mentioned earlier begins by calling the *updateAccelAngles()* and *updateGyroAngles()* functions. Both of these functions update the angles that each sensor reads, based off the linear acceleration and angular velocities of the IMU. The accelerometer is able to calculate the pitch and roll angles, based off the acceleration, but not yaw. The gyroscope is able to read the angular velocity in all three axes, and integrates these values to produce the angular displacement. The values produced by these functions account for the errors that were calculated at the beginning. After control of the code goes back to the *updateAngles()* function, a complementary filter is used to combine the angles from the gyroscope and the accelerometer. While the pitch and roll are not used for this prototype, we included the ability for it to be determined so that the data is available in case we, or any other users, wish to implement them.

5.2.3.2 Motor Control

The two files for motor control, Motor.h and Motor.cpp, are a header file and a C++ file, respectively. A header file essentially declares what the class will do, while the C++ file will essentially define how the class will achieve these features. Thus, the header file, Motor.h, just declares the variables and functions that are a part of the Motor class and will be implemented within the C++ file, Motor.cpp. It is worth noting that the header file includes a reference to the Arduino.h file, which allows the Motor class to use functions that are specific to Arduino sketches. Motor.cpp begins with the constructor which has parameters for the speed and direction pin that will be used to control the motor as well as a third parameter for if the motor should be set up for inverted control. Typically, motors that face the same direction should have the same inverse value. That is, if the left motors are not inverted, the right motors should be. This constructor saves the values of the parameters and then calls the *Motor.init()* function, which is used for initialization. Within this initialization function, the pin modes of the speed and direction pins are set to be outputs, which is accomplished through the Arduino specific function *pinMode()*. The variables *Motor.speed* and *Motor.direction* are also initialized to 0 and false, respectively. Next is the *Motor.drive()* function, which has a single parameter for speed, which has a range of -255 to 255. This function simply sets the variables *Motor.speed* and *Motor.direction* based on the speed parameter and the motor inversion status from the constructor. If the motor is not inverted, then *Motor.direction* is set to the Boolean result of if the speed parameter is less than or equal to zero, while *Motor.speed* is set to the speed parameter if it is positive, otherwise it is incremented by 255. If the motor is inverted, then *Motor.direction* is set to the Boolean result of if the speed parameter is positive, while *Motor.speed* is set to the negative of the speed parameter if it is less than or equal to zero, otherwise it is set to 255 minus the speed parameter. The reasoning behind why the values are determined this way is because we are using the sign-magnitude mode, which is described in **Section 3.3.4**, and because it is designed so that a value of 255 used to drive all of the motors will make the agent prototype move forward at full speed, while -255 will make the agent prototype move backward at full speed. Finally, this function ends by writing the *Motor.direction* value to the motor direction pin, and writing the *Motor.speed* value to the motor speed pin. The Motor class also contains a few functions used to just return the values of some of the variables for debugging purposes.

5.2.3.3 Mecanum Control

The two files for Mecanum Drive control, Mecanum.h and Mecanum.cpp, are set up similarly as the files for the Motor class. The Mecanum header file includes a reference to the Motor.h file, which means that since a reference to Mecanum.h is included in the ArduinoControl.ino file, a reference to Motor.h does not need to be included within the Arduino sketch file. Mecanum.cpp begins with the constructor which has four parameters, each of which are a reference to a Motor object. These references are saved and will be used as the front left, front right, back left, and back right motors, respectively. This constructor then calls the *Mecanum.init()* function, which is used for initialization. Within this initialization function, the variables *Mecanum.x*, *Mecanum.y*, and *Mecanum.z* are all set to zero, and will later be used to determine forward motion, strafing, and turning, respectively. The uses of these three axes is why the X- and Y-axes needed to be swapped in the Arduino sketch file, since we wanted the X-axis of the left joystick to be used for strafing while the Y-axis was to be used for forward motion. Next is the *Mecanum.drive()* function, which takes in three parameters for the x, y, and z values and are saved in the corresponding variables mentioned in the initialization function. After saving these values, the velocities of each motor are calculated as follows:

- Motor 1: x + y + z
- Motor 2: x - y - z
- Motor 3: x - y + z
- Motor 4: x + y - z

We determined these formulas by using the Mecanum Drive wheel control principle as seen in Figure 3.35. Since the x, y, and z values all have a range of -1.0 to 1.0, that means that each of the above velocities would have a range of -3.0 to 3.0. To make these velocities have a range of -255 to 255, each of these velocities are then passed to the *Mecanum.scaleMotors()* function. This function will store the maximum value out of each of the velocities, or 1 if all of them are less than 1. Next, each velocity is multiplied by 255 and divided by the maximum value mentioned above. This is so that if any of the velocities have a magnitude greater than 1, then all the velocities should be scaled down so that the maximum velocity is 1. However, if all the velocities already have a magnitude less than 1, then that means the agent prototype is meant to drive at a slower speed and should not be scaled up. After scaling the velocities by 255 divided by the previously mentioned maximum value, control of the code is returned back to the *Mecanum.drive()* function. This function ends by calling *Motor.drive()* for each of the four motors and passing the newly scaled velocities. The Mecanum class also contains three functions that are used to return the x, y, and z values that are currently saved within the class.

5.3 Prototype 2

At this stage of the project, prototype 2 is considered to be the finalized design based on considerations from the first prototype. Rather than calling it the final design, we expect that even after fulfilling our current design plans, there will be aspects of the robot that we will want to change after its full development. For this reason, we decided to keep the name of this section prototype 2. It is worth noting that development of this prototype has not yet begun and is still in the planning phase.

5.3.1 Mechanical

After considering the observations seen from prototype 1, we decided to redesign the frames of the mecanum wheels. With the previous two-piece design, there were points in time when there were no rollers touching the ground, this was deemed an issue caused by the frame design. With the FDM printed mecanum wheels we were able to provide proof of concept, allowing us to move forward in iterating the design. Going to a one-piece design will also allow for larger rollers creating a larger contact, therefore, creating more friction. The team is also considering increasing the overall diameter of the wheels, due to the rather low slow top speed of the agent. While considering this change, it is important to recognize the loss of force at the ground that will be experienced when increasing the radius arm that the motor will experience. This tradeoff will be justified as the initial assumptions made when calculating the required motor’s torque were based on much higher weights than the final agent's design. Along with the geometry change of the frames, we would like to manufacture them differently and with another material. Using SLS printing methods we would be able to end use frames that would meet and exceed the needs of the wheels, we understand that this type of manufacturing is not cheap and easily accessible. As a close second, we will print using FDM methods with ABS as the printing material. As for the TPU rollers, while they do provide plenty of friction there is more performance that can be achieved. Ideally, we would like to use SLA printing methods to print a mold for the roller. Using this mold, we would like to use silicon as the working material in the mold. Doing so would produce a roller with a much higher coefficient of friction. Much like SLS, SLA is a method of 3D printing that is not extremely accessible to the public, as a second option we plan on applying post processing methods such as coating them in a higher coefficient of friction material. Finally, for prototype 2 we will implement the first iteration of the latching design along with the locking mechanism. Much like the mecanum wheel prototype, this first iteration of the locking mechanism will be printed with PLA through FDM printing to provide a full proof of concept of the design. Once the design is validated, we will make the necessary design changes and manufacture the finalized components out of a metal alloy, at this moment we are looking at the viability of aluminum.

5.3.2 Electrical

This section contains the development of the electrical system in prototype 2. This will include using previous implementations from prototype 1 that worked well, fixing issues, and changing outdated systems to fit better with the final goal of Hive. Some things that will be changed include the power source and battery implementation, power distribution, and creating a PCB application.

5.3.2.1 Microcontroller

The microcontroller unit chosen for the final agent is the Espressif ESP32-WROOM-32. The final PCB will feature pin headers in order for the

ESP32-WROOM-32 module to be plugged directly into the board. The GPIO pins for each motor driver will be connected through traces on the board, but all unused GPIO pins will be left untapped in order to carry over to the attachment ports.

5.3.2.1.1 Microcontroller Application Considerations

This section will detail the considerations and design aspects that will accompany the microcontroller to create an extremely successful and reliable microcontroller system.

Stable power source: As stated before, our microcontroller will need a very stable power source to prevent issues. As such we are adding additional filter capacitors right at the entrance of the microcontroller chip because having power rails run by multiple devices each pulling different amounts of energy at any given time will cause power fluctuations. Additionally the distance that the power rail will need to travel to reach not only the microcontroller but also the many other devices that will tap into the power rail gives more opportunity to capacitance and inductance issues. Adding to the importance of an additional power filter capacitor onto the input of the microcontroller.

Prevent Interference: wireless communication and computer processing is very delicate and often needs protection from outside disturbances. As we are combining both of these devices together in a single location we must be extremely considerate of possible interferences. One of these disturbances can be from power traces on the board. It is well known that wires will create a magnetic field when passing large amounts of current and this is to be true on PCB traces as well. Another consideration in confirming that the microcontroller does not receive interference is to avoid placing power traces nearby the device.

Convenience: Another consideration is about convenience of use. Given the goal of modularity and because we are creating prototype 2 instead of a finished product, we believe that some modifications will be made and accessibility of the ports on the microcontroller will be necessary. Another convenience consideration is for programming. As more modules are designed for the device changes to the code on the microcontroller will become necessary, at least during the prototyping stage. New technology has been developed to allow for wireless programming but for our prototypes this technology will not be applied to our design.

5.3.2.2 Motor Controller

The motor controller chosen for our second prototype is the Texas Instruments DRV8876. While the Texas Instruments L293D worked for the first prototype, the DRV8876 is a better option. This motor controller can handle more voltage and current than the L293D, which makes it a nice option for scalability in the future if larger motors are desired. The DRV8876 also has a lower voltage drop and less power dissipation, which makes it much more suitable for battery-powered applications such as ours. This lower voltage drop also means that the motors can receive their rated voltages without needing to increase our power supply by two or more volts.

5.3.2.3 Inertial Measurement Unit

The inertial measurement unit (IMU) chosen for our second prototype is the Pololu MinIMU-9 v5. While the InvenSense MPU-9255 worked for the first prototype, the MinIMU-9 v5 is an overall better option. This IMU has a lower power consumption than the MPU-9255, which is more suited for battery-powered applications, where a lower power consumption is more heavily weighted. The MinIMU-9 v5 also has lower noise levels than the MPU-9255, which can result in more accurate measurements.

5.3.2.4 Full Schematic

The full schematic to be developed on the PCB consists of a 3.3 V, 5 V, and 12 V DC buck converter, as well as four motor controllers for our DC brushless motors. The buck converters are directly fed from the battery, whose leads will be brought via 18 AWG wire directly to a fortified PCB trace. The 3.3 V output will be brought directly to the electrical ports on each side of the PCB, since none of the main onboard components will need 3.3 V other than the attachments. The 5 V output will be connected to the microcontroller input power pin as well as each of the electrical ports. The 12 V traces will be brought directly to the motor drivers, as well as each of the electrical ports. Not pictured in the schematic are the IMU, microcontroller unit, limit switch, and emergency stop button.

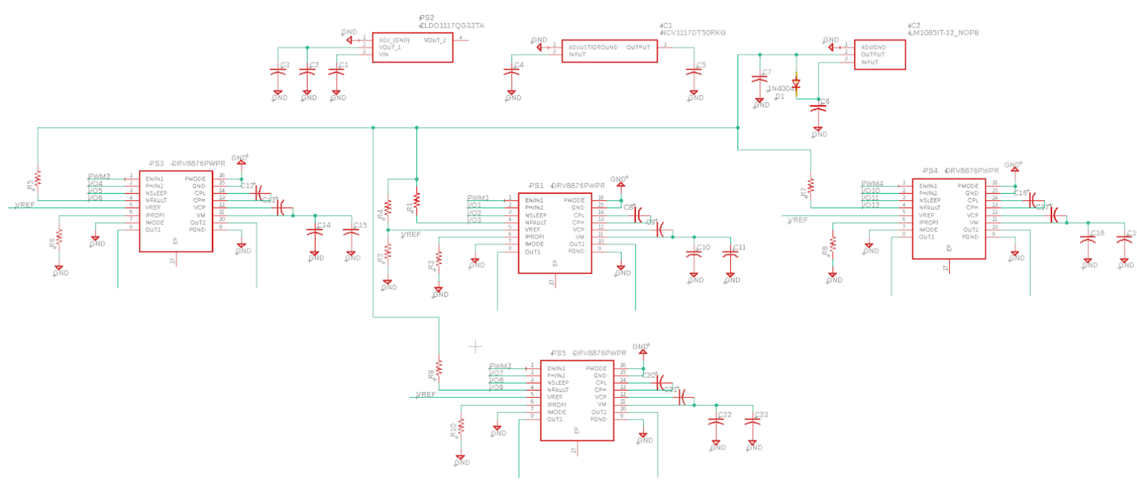

Figure 5.6: Full project schematic with voltage regulators and motor drivers. Designed by Benjamin Palladino.

5.3.2.5 PCB Design

We will be embedding a compatible pin header into our PCB so that the ESP32 module can be plugged directly in. Some of the pins will be traced directly to inputs of the onboard voltage regulators and motor drivers, while some pins will be left untraced in order to account for any external connections like the limit switch and IMU.

5.3.2.6 Battery Implementation

This section will be an extension of **Section 3.3.7** Battery Module with a focus on the application of the design. Starting by reviewing the battery selected, then going over the requirements of the battery module, and application of features.

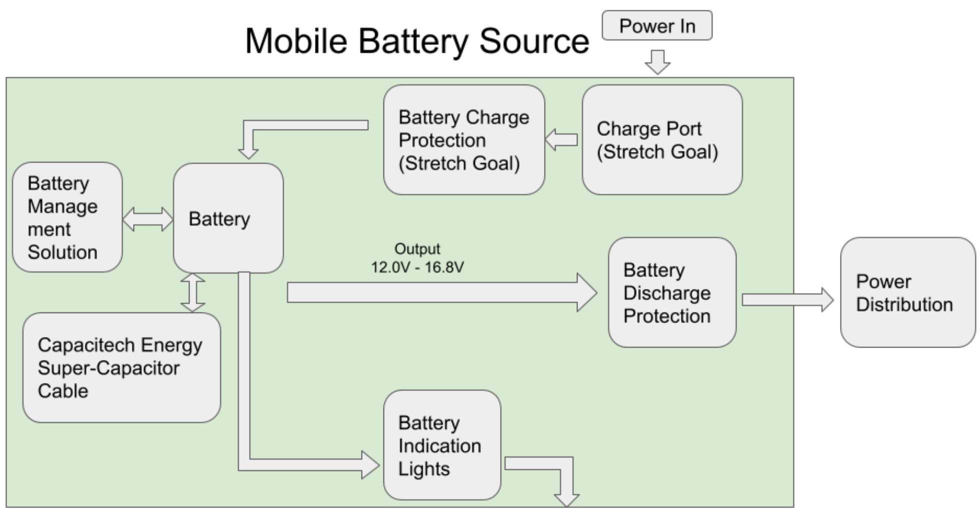

Figure 5.7: Mobile Battery Source Application Block Diagram. Designed by Isaac Finley

5.3.2.6.1 First Analysis of Requirements

In this section we will go over the requirements for the application of the battery module. Requirements help to prevent the team from getting distracted by features which may not be vital to the development of the project by creating a clear guideline of what can be considered in our design. The following requirements were created with our project's goals of a robotic solution which is adaptable for many situations.

Safety Mechanisms: Safety mechanisms are necessary to prevent damage to the battery and danger to the environment and people. Some safety mechanisms for the battery module we are to consider include safe charging, over discharge protection, surge protection, and an emergency switch.

Charging Lithium Ion batteries has certain rules that must be upheld to keep the battery in top condition and the user safe. One of these rules is to prevent lithium ion cells from getting overcharged to above 4.2 V per cell. If overcharged the chances for catastrophic failure increases dramatically and the cell's capacity will quickly degrade. This is why a charging circuit is necessary to keep the battery cells from being overcharged. But this issue continues to be true when we put cells in series to increase a battery module's voltage. When multiple cells are put in series a new phenomenon of balancing is required. An unbalanced battery module refers to a module with multiple cells connected in series and the voltages of the individual cells do not match one another. This occurs because battery modules are commonly depleted and used by the extreme terminal ends, so that we can utilize the higher voltage. But by using these terminals energy is not pulled from each cell evenly. In an ideal world all batteries will be made identically but in reality each cell will have slightly varying internal resistances which will discharge and charge at different rates very slightly. Over time as the battery module is cycled repeatedly, these cells will become out of balance.

The issue with unbalanced cells is best described through an example: if we consider a battery module with 2 batteries in series, with a 4.2 V maximum voltage rating on each cell, we get a maximum battery module voltage of 8.4 V. If we were to charge the battery module using the extreme terminals up to 8.4 V, initially the voltages would match as the variation between the batteries is very small. But as the module is cycled, the cells will become out of balance and create a phenomenon where the first cell becomes 4.0 V because it was slow at charging and the second cell will be charged to be 4.4 V. This is where the overcharge concern will resurface when we are using a battery module with multiple cells in series. With a cell being charged to the overvoltage condition it can cause fires, explosions, chemical burns, battery puffing, etc.

Common solutions to safe charging and the balancing issue is to implement either or a combination of an Active Balancer and Battery Management Solution (BMS). Hive is going to use a BMS because of the increased functionality of a charger, balancer, discharge protection and surge protection included within a single package.

Other issues to consider for safety is the protection of bare wires. In a warehouse there are many chances of items becoming lodged or snatching onto wires that may be exposed pulling out connections and rendering the battery inoperable. Additionally, leaving any open contacts is of bad practice and allows for the elements to weather away at the battery and cause short circuits. As such our battery will have all wires neatly packed away to prevent snagging and all batteries will be covered with a popular battery protection tape called fish tape. Fish tape is a tape made out of seaweed which gives the tape the properties of toughness and being not conductive. This tape will protect the batteries from any outside influences.

Emergency Switch: An emergency switch is the power cut-off safety switch. It needs to be easily visible and accessible to cut off power for the possibility of something going wrong.

User interface: The importance of user interface is to communicate with the user the status of the battery module. This is to help facilitate charging when getting close to low capacity within the battery module, but to also inform when the battery is fully charged to inform the user that the agent can now be used.

5.3.2.6.2 Application of Battery Module Design

In Section 3.3.7 a battery cell was selected and a battery module was designed to fit the requirements for Hive. This battery was the ANR26650M1-B with four cells connected in series to increase the voltage of the battery module.

For the design of the battery module, Hive is going to use 4 cells in series to create a higher usable system voltage Then a common BMS will be connected to each individual battery cell to allow for the BMS to balance the individual cells. An XT-60 port will be used as the output connector from the BMS towards the power distribution design to allow for easy connectability. The XT-60 was chosen because of its popularity in small electronic vehicles, easy connectability, and high current capability of 60 A. Another XT-60 will be connected to the charge port of the BMS and clearly labeled to prevent any issues when implementing the charging procedure. The connectors and wires will be connected through soldering.

5.3.2.4 Power Distribution Design

Once we have finalized our design for our power source, next is using this energy. But this energy is not ready to be used straight from the battery. As the battery module is used their energy becomes depleted and their voltages fluctuate greatly during their charging and discharging cycles. Since their voltages are unstable and can occasionally become unpredictable with instant voltage changes with large loads most devices cannot use this energy as is; especially microcontrollers which require a very stable power source. Stabilizing this volatile voltage and converting this “raw” power to a stable and predictable power source is the job of the power distribution section.

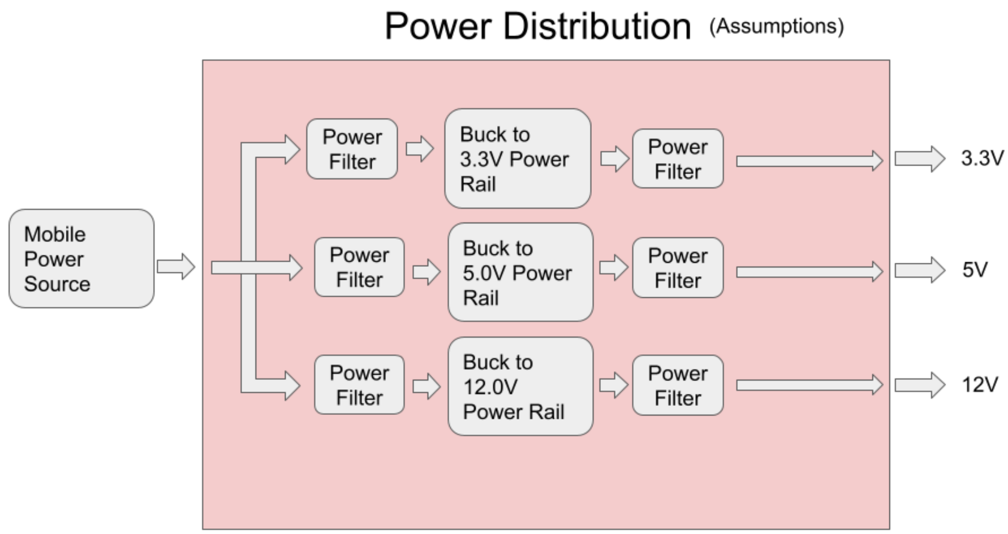

Figure 5.8: Power Distribution Block Diagram. Designed by Isaac Finley

Power distribution supplies power to the entire system, as such it needs to be able to supply all of the voltage levels and power levels required for each device. In the power distribution we include a 3.3 V regulator which only requires 3 parts, the regulator, then 2 capacitors to aid in power filtering. The devices that require the 3.3 V power rail are low power devices and as such the regulator can be lower power as well.

The next voltage configuration is the 5 V power rail. This device is designed in a very similar way to the 3.3 V regulator, only requiring two additional power filter capacitors for stability. The devices that pull energy from this power rail require more energy but are still relatively low power devices compared to the motors. As such this 5 V regulator will be capable of delivering more power than the 3.3 V regulator.

The last voltage configuration is the 12 V power rail. The high power devices draw power from this power rail and as such a large amount of current is necessary for this device. As the drive train requires a total of 2.76 A and being prepared for the modules that will be added to the agents, the 12 V power rail will be capable of handling 5 A or more.

5.3.3 Software

This section will go over the software that we will develop and use for the second prototype. Most of the code that was used in the first prototype will be reused for this prototype, but with some newer additions to introduce more of the functionality that we desire. For one, the software will be updated to support the newer hardware that is mentioned in the previous electrical section. The software will also be updated so that the microcontroller unit, the ESP32-WROOM-32, will act as a WebSocket client and will communicate with the Robot Router using this method. Then, the commands that are received from the Robot Router will need to be correctly parsed and interpreted so that the robot reacts accordingly. The accelerometer data from the MinIMU-9 v5 will also be derived so that the linear velocity data will be available so that the agent will be able to tell if it is driving correctly. For instance, if the agent is instructed to strafe, but it starts moving forward or backward, then this will be reflected in the velocity data and the agent will be able to correct itself.

6.0 Project Prototype Testing Plan

This section of the documentation is dedicated to explaining the testing environment and what was considered while making our decisions. This section will also detail the testing environments of all of the subsystems and the components that make them up. The use of these testing environments serves as a method of validation of the prototypes and their components, they are meant to serve as a test for the agent as a whole and for each of the components used. By doing this, each component is proven to work for this application. Using this technique of validation allows for successive implementation of subsystems until the final state of the project is complete.

6.1 Hardware Test Environment

With our project serving as a proof of concept for this type of robotic system, there are many ways to implement and test our project. In this project, the team has decided to use a warehouse environment as the testing area. For the robotic system to be successful, it must be able to operate and effectively maneuver around said environment. From research, the team discovered that most warehouses typically have a smooth flat surface on the ground level. The most common example found was polished concrete. With a surface like this, there are many industrial machines that thrive in this environment. Some examples include forklifts, carts, dollies, etc. The most common trait of all these machines are their chassis to surface interface. All of these listed rely on a standard traction wheel consisting of both pneumatic and non pneumatic options. Now considering that strafing is an engineering specification for our project, this narrows our wheel selection down to only holonomic wheels. The most common trait of all of these is their inferior traction capability. Finally, if the final project can operate on smooth level concrete at room temperature, it will succeed in the intended environment. To be counted as successful in this environment our agent will fulfill two requirements.

1. Freely move around a concrete surface without in all directions including: left, right, forward, backward, and at 45 degrees.
2. The ability to pull a payload to a designated location on the warehouse floor without becoming immobile because of a lack of friction.

To achieve this validation, our goal is to utilize the Robotics Club of Central Florida’s lab in Partnership II, which has a large enough space and adequate testing surface.

6.1.1 Motor Testing Environment

In the design of each agent, the two main types of motors used are DC motors and servo motors. These motors have different characteristics and therefore will be subjected to different tests for each. However, both types of motors can be tested before implementation into the final project. To test these motors, they were mounted to a solid testing bench and tested for the following: stall torque, maximum speed, rated current/voltage, and weight. This is critical to our design as it allowed for us to ensure that each component was capable of what was marketed and therefore capable of performing well in the prototype and final product.

6.1.2 Wheel Testing Environment

After the wheels are manufactured and assembled, they will be put through some simple loading and spinning tests to ensure that they spin freely and can be tested further. Once this is established, they will be implemented into the prototype.

6.1.3 Sensor Testing Environment

To successfully navigate the environment, each agent will be equipped with several types of sensors, allowing them to interpret the environment. Each sensor is subjected to its own environment and series of tests, for each sensor this is for the most part identical, however; some sensors were subjected to their own unique tests. The common tests among the bunch include sensitivity, reaction speed, update time, and power usage. These will serve as the basic benchmarks for each of the sensors selected for the project. The components included are our limit switches, ultrasonic sensors, and an above-head camera.

6.1.3.1 Limit Switches

The tests that we will perform for the limit switches are sensitivity, reaction speed, update time, and power usage. These tests will allow us to ensure the limit switches operate properly and will work for the purposes of our agent’s operations.

Sensitivity: For sensitivity we will need to ensure that the limit switch is only triggered when desired to be. This is important because this limit switch is meant to only be activated when coming into contact with the environment. If any false triggers were sent to the system because of too high a sensitivity, the agent will stop unnecessarily. Additionally, if a limit switch is not sensitive enough, then the agent may become damaged by hitting obstacles. These possible weaknesses need to be tested and their limitations need to be designed around.

Reaction Speed: Reaction speed can be tested by triggering the limit switch and timing how long it takes for the agent to react. This test is important as when the limit switch is triggered, the agent should halt normal operations immediately. So, it will be critical to ensure we make the reaction speed as fast as possible or put into place a methodology to ensure the reaction speed does not cause any failures.

Update Time: For the update time testing, we will measure how long it takes for a signal to be received and recognized from when the limit switch is activated, which will tell us the update time. This is of critical importance because knowing the time it takes for the agent to receive the information plays into understanding the reaction speed. As the agent physically cannot react to the stimulus from the limit switch getting triggered if the agent has yet to receive the information about the limit switch being triggered, so the agent's total time to respond to the limit switch being triggered would be the time it takes to realize the limit switch was triggered plus the time it takes to actuate the response to the limit switch triggering.

Power Usage: Lastly, to test the power usage we can use a multimeter to determine the amount of power drawn by the limit switches. This is important to understanding how much power is necessary to drive the device and will play into whether the battery can handle the load required and it will also give a good estimation of how long the agent will be able to sustain operations for as that will be a concern for demonstration purposes that we will need to ensure the agent can operate for the full duration of the demonstration.

6.1.3.2 Ultrasonic Sensors

The tests that will be performed for the ultrasonic sensors are sensitivity, reaction speed, update time, and power usage. These tests will allow us to ensure the ultrasonic sensors operate properly and will work for the purposes of our agent's operations.

Sensitivity: Sensitivity covers the range of the ultrasonic sensor, meaning the field of view and what distance classifies as triggering the sensor to inform the agent it is too close to an obstacle. The field of view can be seen in the data sheet for the selected ultrasonic sensor and can easily also be tested by manually moving an object in and out of frame. For the distance to trigger the ultrasonic sensor, we will need to see how the ultrasonic sensor reacts at different distances and then using that information we can design and program a system around this distance to ensure that the ultrasonic sensor triggers at an appropriate distance.

Reaction Speed: Reaction speed is how fast the agent can react to the stimulus provided by the ultrasonic sensor, which can be tested by triggering the ultrasonic sensor and timing how long it takes for the agent to react to said stimulus. This is important because if the agent does not react quickly enough, it may become damaged and could therefore suffer in its operations, which is something that we would like to avoid.

Update Time: Update time is how long it takes to be able to read the data from the ultrasonic sensor. This can be measured by timing how quickly the data updates in the software. This is important because knowing how long the ultrasonic sensor's readings are spaced out and how long it takes to sense something is important to the speed and efficiency of the agent and the overall effectiveness of operation for the agent.

Power Usage: Power usage can be tested by using a multimeter to observe the power usage and is important to be able to plan around how much power the agent will be drawing versus how much power is available to be drawn to determine an approximate runtime for operation of the agent.

6.1.3.3 Over-Head Camera

The tests that we will perform for the camera are sensitivity, reaction speed, and update time. These tests will allow us to ensure the above-head camera operates properly and will work for the purposes of our agent's operations.

Sensitivity: For sensitivity in this case, it refers to the pixel density of the camera, also known as resolution.The resolution of the camera will be given by data sheets. This information is important in recognizing objects in the camera's feed. If an object is not detailed enough the software team will not be able to recognize the difference between the agent and other objects. How well things can be detected that are inside the camera's range is something that will be purely done with the software as setting the limits on what counts as important is not something the camera can do on its own.

Reaction Speed: Reaction speed will be how quickly the agent can react to stimulus provided by the camera, which will primarily be applicable when dealing with dynamic environments. If the environment and objects in the environment are not moving there will not need to be a high reaction speed as the environment will be changing slowly. Reaction speed will be determined by not only the speed of sending the camera feed to the robot server but also the speed of processing this data and sending a command to the agent. This will be tested by running the agent and the code giving instructions on the basis of the camera and ensuring the agent does not run into any obstacles.

Update Time: Update time is how long it takes for the image to go from the camera and be received by the processor that is processing it, which can be measured by moving something in the frame of the camera and timing how long it takes for it to register as moving. This update time will play a part in the reaction speed of the agent.

6.1.3.4 Sensor Considerations

It is also important to ensure that the sensors all work together and do not interfere with each other's usage and information gathering capabilities. If the devices interfere with each other it will negatively affect the operation of the agent and will cause the agent to not have the information needed to operate properly. The primary test that will be done to determine if there are collaborative issues in using all of the sensors will be an electromagnetic interference (EMI) and electromagnetic compatibility (EMC) test. An EMI/EMC test can be performed by measuring the electromagnetic interference (EMI) and electromagnetic compatibility (EMC) of the regulator under various conditions to check if the regulator meets the specified requirements. As stated previously, this is important to ensure that the agent functions properly with all of the sensors working together as the agent will need the information provided by the sensors to properly operate, so it is important to ensure through the EMI/EMC test that the agent will not have any difficulties arising from the electrical and magnetic components being used simultaneously as there will be a significant number of electrical and magnetic components being used during normal operations and functionality of the agent.

We expect the sensors will all work with each other without interfering with each other during normal operations of the agent. The reaction time for the limit switches should be lower than 10 ms, and the camera should be in the same range for the agent to properly be able to orient itself. The maximum power draw of the sensors should be significantly less than the maximum power output of the battery.

6.1.4 Communication Modules Testing Environment

After acquiring the components required for communication, some simple tests will be conducted to ensure their functionality. A simple sensor reading will be sent from the transceiver on the agent to another receiver acting as a placeholder for the off board server which is planned to be used in the final product. Ultimately this will prove that communication between the agent and server is achievable.

The first test we will perform is a signal strength and range test to ensure the agent can communicate with the decision maker and vice versa. This is of critical importance because we need to ensure the agent can communicate with the decision maker and vice versa because the agent will not be able to operate properly if it does not have the ability to properly communicate at range. A latency test will also need to be performed to better understand how much reaction speed the agent as a whole has and how quickly it can respond to commands from the decision maker. An environmental test will also need to be performed to ensure that no environmental effects affect how the communication module acts. Once the signal strength, latency, and environmental tests are complete, we will compile the data and analyze the results. Based on our findings, we may need to make adjustments to the agent's communication system or take other corrective actions to ensure that it performs optimally in various conditions. Ultimately, our goal is to create an agent that can communicate effectively and efficiently with the decision maker, while also being able to adapt to changing circumstances and perform its duties effectively.

We expect the signal strength and range test to inform us that the communication module is capable of covering the entire ten foot by ten foot area that will be used for the demonstration as that is the bare minimum of what will be needed for communication purposes. The robot should have a latency of 100 ms or less for it to be able to respond to commands from the decision maker in an adequate amount of time. We expect there to be minimal environmental effects on how the communication module works as if there were, the agent would not always be able to receive commands from the decision maker, which would be counterproductive to the continued normal operation of the agent and its receiving of commands properly as the agent will not be able to function without receiving commands from the decision maker.

6.1.5 Battery Testing Environment

To ensure that the selected battery will be usable by every device within the agent, a testing environment is necessary. We will be testing the different capabilities of the battery including: voltage output, power delivery, energy delivery, and normal environment operating conditions.

The most basic test that will need to be performed on the battery, and therefore the first test that will be performed is testing to ensure the battery is outputting the correct voltage, which can simply be done by measuring the output voltage of the battery after a charge cycle and as long as it is the correct voltage, then we can ensure that the battery will be outputting the correct voltage for the rest of the power system to successfully operate from. Another vital test to perform for the battery is the discharge time test, which measures how long the battery can provide power before needing to be recharged, which is important as this will determine how long the demonstration can be and it is vital to ensure the agent will be operational for the full duration of the demonstration, and this can be tested by running the battery until it runs out and timing how long that takes, which will give a good idea of if it will be capable of operation throughout the full demonstration. Similarly, the charge time of the battery should also be tested so that it can be noted down how long it takes to fully charge the battery so that things can be done more efficiently. Also the cycle life should not be an issue with the number of times the battery is projected to be discharged and charged, so minimal if any testing will be done on that subject matter as the battery will only be used for the testing that needs to be done and the demonstration that will be performed.

To test if the battery module is able to handle a high enough power delivery rating to produce enough power for long enough, a test bench will be utilized to operate several DC motors and servos at the same time. The time, voltage, amperage everything is powered will be recorded. The goal is to prove that not only proving that the battery will have the capability needed but also give an idea of the true capabilities of the device.how long the agents will be able to operate.

The battery will need to be able to function under numerous circumstances that will need to be rigorously tested for. Firstly, temperature and its effects on the battery will need to be tested to ensure that neither high nor low temperatures will negatively impact the battery, which will be done by observing the power output of the battery at temperatures other than room temperature to ensure that the battery stays within an acceptable power range. Also to be tested is the battery's resilience against impacts and vibrations as the battery will be a part of the agent, and if the agent suffers and impacts or vibrations the battery will as well, given that they will be securely fastened together. The best way to test these will be to be measuring the power output of the battery with neither affecting it, then each individually affecting the battery, and finally both affecting the battery at the same time, to indicate a scenario that is not unlikely to occur during the demonstration, which will ensure the agent is fully operational at all points in time. Also notable in terms of ensuring the demonstration runs smoothly is ensuring that the battery's performance does not degrade over time, which can be tested by checking the battery's performance after each use case to ensure that performance does not drop, and if need be a separate battery can be used for the demonstration to fix any issues that may arise from aging of the initial battery. Testing the safety features of the battery is also something that will be done as the Hive team values safety and the battery could be potentially hazardous without safety measures. The testing procedures for ensuring all safety features work properly will be to subject the battery to normal and extreme conditions that it might encounter while a part of the agent and checking the battery to ensure there were no failures and that it is still safe to use. Finally, an EMI test will be performed to ensure the battery's electromagnetic interference (EMI) does not interfere with other components on the agent, which will be extremely important as there will be many other components on the agent that need to be functional so ensuring the battery does not interfere with them is critical to the proper operation of the agent. This test can be performed by having the battery in close proximity to the other components comprising the agent and running them simultaneously, making sure all of the components function as intended for proper operation of the agent.

We expect the battery to output in a range of approximately 12 V to 16.8 V and not degrade over the course of testing and demonstrating the agent. The battery's safety features should remain operational under all conditions that we test at to ensure the safety of the agent and the people near it. The safety measures will also need to pass the tests performed on them as safety is of critical importance to us and it would be less than ideal if the safety features failed while testing or demonstrating the agent. It is also expected that the battery will not suffer from problems arising from impacts and vibrations as that is likely to occur and therefore it is important that the battery maintains normal operations under those conditions. The battery should also work with all other components being used in the agent as the battery will in fact be the main power supply of the agent and therefore it would be disadvantageous to say the least if the battery was incapable of cooperating with the other components the agent consists of.

6.2 Hardware Specific Testing

After the hardware is verified to work properly from the manufacturer, our team will begin to perform specialized tests to ensure that the equipment will work at a level of at least tech readiness level 6. As tests are completed the required changes will be implemented into the prototype leading up to the final product; these tests will also serve as validation as to why we picked these components over the others proposed.

6.2.1 Wheel and Motor Testing

With friction being a great concern and design challenge for the project, it was imperative to test the wheels manufactured. To test these wheels, the first initial prototype was created. This included a simple frame, four wheels and motors and all the necessary electronics to power and control the prototype. For testing purposes, a large sheet of eighth inch plywood will sit on top of the agent spanning its width and length. This platform is meant to serve as a payload loading area. Then a spring scale will be attached to both the agent and a solid upright surface. Starting with the spring scale taut, we will command the agent to attempt to forward and observe the wheel to ground interaction. After recording the results, weight will be added in increments until wheel slip is achieved. The readout of the spring scale just before wheel slippage occurs will act as our maximum weight bearing load. If necessary, the coefficient of friction may need to be increased to carry the desired load. After this specific test the following should be answered: will the wheels produce enough friction and will the motor produce enough torque to carry the planned load.

6.2.2 Chassis Testing

The chassis of our agent will sit on top of the holonomic drivetrain and will serve as the main platform for placing our electronics. With the planned environment to be a more industrial setting, durability and reliability are both very important characteristics that need to be tested to ensure the most optimized performance of the device. The most common method of damage will more than likely come from a collision. Whether it be from another agent or the environment both could lead to catastrophic failure. Therefore, stress analysis and impact testing will be conducted to test the integrity of the structure at key points of stress concentration.

6.3 Electrical Testing

Several different tests will need to be performed for the electrical components composing the agent, which will be detailed here. Devices include voltage regulators of various configurations, microcontrollers, and motor drivers.

6.3.1 Voltage Regulator Testing

The first methodology of testing the voltage regulators will be a multimeter test. This is the most common and easiest way to test a voltage regulator, which is done by using a multimeter to check the output voltage. The multimeter should be set to the DC voltage measurement mode, and the red probe should be connected to the output terminal of the regulator while the black probe is connected to the ground terminal. This will ensure that the voltage regulator is outputting the correct voltage and not an incorrect voltage that would signify the voltage regulator not working. Another way to test the voltage regulators is by applying a load to them and measuring the output voltage. A load can be a light bulb or any other device that draws a significant amount of current, which will likely be a motor for the 12-volt regulator. If the voltage drops significantly when a load is applied, the regulator may be faulty, which is not good for the operation of the agent. The Hive team will also perform a test to ensure the voltage regulators remain operational at higher temperatures as a voltage regulator should be able to handle high temperatures without failure. This heat test will be performed by heating the regulator with a heat gun or a hair dryer and checking for any changes in the output voltage. This is important to do as the agent might need to be operational at higher temperatures, and the voltage regulators are an important factor in ensuring the continued operation of the agent. Along with ensuring the voltage regulators remain operational at higher temperatures, a temperature coefficient test should also be done. A voltage regulator should have a predictable change in output voltage with temperature changes, which is what a temperature coefficient test will check. A temperature coefficient test can be performed by gradually increasing or decreasing the temperature of the regulator while monitoring the output voltage to check if the voltage changes according to the specified temperature coefficient. This test can also be used to learn the temperature coefficient if it is not known, assuming the voltage regulator works as intended.

An electromagnetic interference (EMI) and electromagnetic compatibility (EMC) test will also be necessary to perform for the voltage regulators on the agent, as a voltage regulator should not interfere with other circuits or be susceptible to interference from other circuits, which is what an EMI/EMC test tests for. An EMI/EMC test can be performed by measuring the electromagnetic interference (EMI) and electromagnetic compatibility (EMC) of the regulator under various conditions to check if the regulator meets the specified requirements. This will be critical as the agent will have many other electrical and magnetic components on it running simultaneously with the voltage regulators, so it is critical to ensure that they do not interfere with each other. The voltage regulators will also need to be able to run through sustained operation of the agent, so ensuring power dissipation and stability of the output will be important for proper operation of the agent. Power dissipation can be tested by running the regulator at maximum output current and monitoring the regulator temperature to check if it stays within the specified limit. This is important because the heat needs to be dissipated otherwise the voltage regulator will not be outputting the proper voltage and will therefore not be operating under the proper conditions after a set period of time. To ensure that the voltage regulator will operate at proper conditions after extended use, an aging test can be performed by running the regulator continuously for an extended period and checking if the output voltage remains within the specified range. This is important to ensure that the voltage regulators can maintain a stable output voltage over a long period of time, mostly which matters to ensure that the demonstration and testing processes proceed smoothly and without issue. The final important test that will be performed for the voltage regulators is ensuring that the voltage regulators are capable of handling a variety of inputs, as they are designed with that purpose in mind and therefore it needs to be tested to ensure the design works properly. An input voltage range test can be performed by gradually increasing or decreasing the input voltage while monitoring the output voltage to check if the regulator can maintain a stable output, which will allow for the Hive team to know if the voltage regulators can handle the variety of inputs that may be subjected to them and therefore if the design for the voltage regulators do what is intended to be done.

It is expected that the voltage regulators output at the specified voltage and current as the design and construction of the agent will be based around those values, so it is extremely important that we ensure the rated power is correct. It is also critical that the voltage regulators can work with each other and with the other electrical and magnetic components used in the agent, so we expect to see them to be capable of working through the testing we perform. The input voltage range should be the rated input voltage, or at least close enough that the redundancies in the input voltage range make up for the result. Variances in output voltage and current due to temperature should be minimal as it will be critical to ensure the proper voltage and current is being delivered at all times.

6.3.2 ESP32-WROOM-32 Testing

The ESP32-WROOM-32 will need to be capable of communicating with the decision maker that is off the agent in a timely manner as well as communicating with the other components consisting of the agent. To ensure this, a number of tests will be performed. Testing the speed of communication will be an important test to perform as that will be a primary driving factor behind the agent's reactionary speed. This can be tested by sending a command to the ESP32-WROOM-32 and timing how long it takes for the agent to execute the command that it received. This will give us an idea of the reactionary speed of the agent in the field, which will assist us in determining how much leeway is needed to ensure proper operation of the agent. Another test that will be performed is testing the range of the ESP32-WROOM-32 to ensure that it can receive commands in the entirety of the demonstration zone, which is projected to be a ten foot by ten foot area. This can simply be done by positioning the ESP32-WROOM-32 at least fifteen feet away and seeing if it can receive commands as that would be the longest distance assuming the command is sent from one corner to the other. Another test that will be performed is how much power the ESP32-WROOM-32 draws, which will be important to know so we can ensure the agent can operate properly for the entirety of the demonstration, which will be approximately 30 minutes. This can be measured by using a multimeter to observe how much power is drawn and calculating power drawn from that. An electromagnetic interference (EMI) and electromagnetic compatibility (EMC) test will also be performed. An EMI/EMC test can be performed by measuring the electromagnetic interference (EMI) and electromagnetic compatibility (EMC) of the regulator under various conditions to check if the regulator meets the specified requirements. This needs to be done to ensure the ESP32-WROOM-32 can function properly with all the other components being used in the agent as all components will need to be able to function together to ensure proper operation of the agent.

We expect for the ESP32-WROOM-32 to be capable of communicating in less than 100 ms to ensure the agent is capable of reacting to new stimuli in its field of view within a reasonable amount of time and to hopefully avoid crashing into any obstacles. The range of communication should be at a minimum 15 feet, to ensure that communication can happen from one corner of the demonstration area to the other corner of the demonstration area. This will ensure that the ESP32-WROOM-32 is always in range for communication for our purposes, and any extra range of communication is just a nice extra. The ESP32-WROOM-32 will draw a power low enough that all the components in the agent along with the ESP32-WROOM-32 can run continuously for 30 minutes so that the demonstration can be completed without issue. It is expected that the ESP32-WROOM-32 will be able to function properly with all other components on the agent and not interfere with any other components as this is critical to proper operation of the agent during all operations it will be performing.

6.3.3 Motor Driver Testing

This section will detail the testing environment and requirements of the motor drivers. The motor drivers will need to be able to control and power the DC motors of the agent without issue. The motor drivers will also need to be able to deliver enough current during operation of the DC motors to move the agent and a load. The testing environment to confirm these conditions are to appropriately wire the motor driver to a power source, DC motors, and a microcontroller for data input. The microcontroller will send information to the motor driver and the DC motors need to move smoothly. Then while recording the power going into the motor drivers, we will put the motors under load by attaching a wheel to the shaft and manually putting the wheel on a surface and adding pressure. We expect the motor drivers to be able to drive the DC motors with a heavy load without issue.

7.0 Administrative Content

In this section, the information regarding team organization and planning will be covered. This section will include the discussion and creation of the main timeline for the project. Along with the timeline, each team member rolls, and the overall budget will be outlined. This section is purely administrative and will not cover any information pertaining to the design or implementation of the final project.

7.1 Milestone Discussion

Below is the milestone timeline we've created, acting as a content map for senior design I and senior design II. These milestones will break down what needs to be accomplished each week in order to complete the research and design challenges of the project; while also the required documentation needed by the end of the senior design I. By the end of senior design, I anticipate having a fully developed CAD model of the final design and a PCB and battery balancing system. Finally, we would like to have a prototype started.

By the start of senior design II, we plan to have acquired all components and have them tested to ensure that they will satisfy our needs for the project. During the semester of senior design II, we will build our final prototype that will serve as our main point of presentation. Our goal is to have a prototype that will operate at a minimum of tech readiness level 6. Once the device is fully functional and our paper reaches its final draft, we will begin to prepare our final presentation. It is at this point that the team shall have a very clear understanding of the project and each of its subsystems.

In the table below you will see our main timeline for the project, as we transition from senior design I to II you should see the move from a more researched based approach to an approach focused on physical implementation and testing of the final prototype. As testing is completed, we will iterate the design of the device appropriately.

Table 7.1: Main Timeline

Task #	Description	Date Completed
1	Initial project idea brainstorming	1/13/2023
2	Project idea solidification and organization of group member roles	1/20/2023
3	Initial divide and conquer documentation project specifications created along with subsystem box diagrams	2/3/2023
4	Benchmarking of other products in the space	2/10/2023

Task #	Description	Date Completed
5	Initial PCB Design Brainstorm	2/17/2023
6	Order prototype 1 parts	3/03/2023
7	Finished first draft of report	3/20/2023
8	Initial Drivetrain and frame design	4/06/2023
9	Initial payload manipulation mechanism design	4/6/2023
10	Mobile power source design	4/6/2023
11	Prototype and testing of drivetrain and frame	4/07/2023
12	Final edits and proofread of report	4/20/2023
13	Finish attachment ports	4/25/2023
14	Payload manipulation test	4/27/2023
15	Finalize PCB design	4/31/2023
16	Order PCB and remaining parts	5/2/2023
17	END OF SENIOR DESIGN I	5/2/2023
18	Build final full prototype	9/15/2023
19	Testing and validation of design specifications	9/29/2023
20	Redesign based on results	11/13/2023
21	Finalize the project	11/20/2023
22	Finalize the report	11/17/2023
23	Prepare for senior design showcase and presentation	12/1/2023

Aside from this table each group member's main role and contribution in the project will be provided.

John McClain acts as the mechanical lead. He is in charge of researching and testing the final structure and drivetrain of the agent. Apart from this, he is also in charge of the development of the manipulation system. Both systems are of high importance to the project as they will serve as the setup for the electrical team to work on and test the electronics of each system. These two main responsibilities are broken up into smaller tasks such as: motor selection, frame material selection, wheel design, and manufacturing of the final agent and its supporting parts.

Cameron Nichols acts as the embedded software lead. His main task is to ensure that the agent can interact with the environment and maneuver to the payload and move them accordingly. He also maintains contact with the software team associated with this project as well. This is very important as he is our best chance of ensuring that each team understands the needs and concerns of the others. He will also be involved within the design of the PCB. Finally, also contributed to the manufacturing of 3D printed parts.

Issac Finley acts as the mobile power supply lead. He will be in charge of designing and manufacturing a battery that will be able to run the agents for a thirty minute demostration. He will also assist in the design of the PCB layout.

Ben Palladino acts as the project manager. He is in charge of keeping everyone on track with the decided upon timeline for the project. He will also contribute to the PCB design.

Cooper Fitzgerald's main role is standard compliance, with the thought that this could be implemented into a workplace setting, it is important to ensure that safety standards are met to ensure workplace safety for all people and operations. He is also the one to handle the voltage regulators.

Table 7.2: Research and Design

Task #	Description	Lead	Date Completed
1	Prototype 1 Frame design	John	3/03/2023
2	Prototype 1 Motors and Wheels	John Cameron	3/31/2023
3	Power supply and balancing system along with super capacitors	Issac Ben	3/24/2023
4	Payload manipulation	John	3/24/2023
5	Necessary safety standards	Cooper	3/24/2023
6	Payload Design	John	4/7/2023
7	Sensors	Cameron	4/10/2023
8	Microcontroller	Ben Cameron Issac Cooper	4/14/2023
9	Modular mechanical and electrical connections	Everyone	4/25/2023

Task #	Description	Lead	Date Completed
10	Components	Ben Cameron Issac Cooper	4/25/2023
11	PCB layout	Ben Cameron Issac Cooper	4/31/2023
12	Communication methods between agent and sever	Cameron	5/2/2023

7.2 Budget and Finance Discussion

The main source of budget for this project will be provided by Capacitech Energy. This is contingent upon the team successfully implementing supercapacitors into the project. These super capacitors are a patented component owned by Capacitech Energy, upon successful implementation, Capacitech Energy has agreed to sponsor the project, the agreed upon amount of capital is $500. Each component varies in price depending on the vendor it is procured from, at the same the prices of these components also vary as supply and demand change. The price point of each agent will be no more than $300 each and a minimum of two agents are planned to be manufactured from the showcase presentation. The remaining budget will be spent on the external hardware needed for agent to server communication. One of the main goals of the project is to design an autonomous robotic ground-based system that can adapt to several situations using modular attachments, ultimately this would lead to a lower cost for the end customer. Considering the scope of this project has been narrowed to payload manipulation in a warehouse setting, the prices of each component in the below table are relevant to this market and thus the chosen target price for the system has been based on such.

Table 7.3: Project Budget per Agent

Part	Price	Quantity
Motor and gearbox combo	$15 per unit	4
Extrusion	$2.4 per foot	1,625 mm
IEEE 802.15.4 Transceiver	$17.5 per unit	1
Servo	$7.5 per unit	2
Ultrasonic sensor	$4 per unit	4 - 16

Part	Price	Quantity
Wiring and Hardware	Acquired	N/A
Status light	Acquired	1
PCB	$10 per unit	3
16 MHz Crystal	$0.34 per unit	5
5 V Regulator	$1	5

Aside from the main electrical components listed, the main frame and chassis can be made of two different materials. Those being aluminum and wood. Both have similar hardware and manufacturing requirements, however, with fluctuation in the supply prices can vary. So, to take advantage of the best deal at the time of manufacture, both materials will be considered for construction.

Appendix A: References

"AK8963 Datasheet by AKM Semiconductor Inc. | Digi-Key Electronics." *Digikey*, October 2013, https://www.digikey.com/htmldatasheets/production/1474913/0/0/1/ak8963.html.

"Arduino Mega 2560 R3 (Atmega2560 - assembled) - Mega!" *Flickr*, 22 August 2013, https://www.flickr.com/photos/adafruit/9572390896.

"ATmega640/1280/1281/2560/2561 Datasheet." *Microchip Technology*, 2 March 2017, http://ww1.microchip.com/downloads/en/DeviceDoc/Atmel-2549-8-bit-AVR-Microcontroller-ATmega640-1280-1281-2560-2561_datasheet.pdf.

Bullas, John C. "speedometer stepper motor power 6 Dual DC Stepper Motor Drive Controller Board Module TB6612FNG Replace L298N RS." *Flickr*, 3 May 2016, https://www.flickr.com/photos/johnbullas/26527801560.

Capacitech Energy. "Batteries or Supercapacitors? Why Not Both? — Capacitech Energy." *Capacitech Energy*, 7 April 2023, https://www.capacitechenergy.com/blog/batteries-or-supercapacitors-why-not-both. Accessed 24 April 2023.

"DRV8876 H-Bridge Motor Driver with Integrated Current Sense and Regulation." *Texas Instruments*, August 2019, https://www.ti.com/lit/ds/symlink/drv8876.pdf?ts=1682356379073&ref_url=https%253A%252F%252Fwww.google.com%252F.

"DRV8876 Single Brushed DC Motor Driver Carrier." *Pololu*, https://www.pololu.com/product/4036.

"Dual Full-Bridge Driver." *STMicroelectronics*, January 2000, https://www.sparkfun.com/datasheets/Robotics/L298_H_Bridge.pdf.

ElectroCraft. "RPX52 RapidPower™ Xtreme Brushless DC Motor." *ElectroCraft*, https://www.electrocraft.com/products/bldc/RPX52/.

"ESP32-WROOM-32E ESP32-WROOM-32UE Datasheet." *Espressif Systems*, 18 May 2020, https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32e_esp32-wroom-32ue_datasheet_en.pdf.

Fukuda, Toshio, and Seiya Nakagawa. "Approach to the Dynamically Reconfigurable Robotic System." *SpringerLink*, Journal of Intelligent and Robotic Systems, 1 September 1987, https://link.springer.com/article/10.1007/BF00437320. Accessed 24 April 2023.

Gastaldi, Laura. "Double acting pneumatic actuator. | Download Scientific Diagram." *ResearchGate*, December 1999, https://www.researchgate.net/figure/Double-acting-pneumatic-actuator_fig1_220675233.

Gundawar, Pratik. "Adaptive Robots: A New Era Has Begun." *United States Cybersecurity Magazine*, American Publishing, 2023, https://www.uscybersecurity.net/adaptive-robots/. Accessed 24 April 2023.

Gustavo Rocha SI. “File:Imagem da próxima geração em estereolitografia SLA.jpg.” *Wikimedia Commons*, 14 November 2019, https://commons.wikimedia.org/wiki/File:Imagem_da_pr%C3%B3xima_gera%C3%A7%C3%A3o_em_estereolitografia_SLA.jpg.

Hymel, Shawn. “How to Use a Calibrated Magnetometer as a Compass.” *Digikey*, https://www.digikey.com/en/maker/projects/how-to-calibrate-a-magnetometer/50f6bc8f36454a03b664dca30cf33a8b.

Iriidium. “Photopolymer Scheme.” *Wikimedia Commons*, 13 March 2014, https://commons.wikimedia.org/w/index.php?search=photopolymerization&title=Special:MediaSearch&go=Go&type=image.

Jenkins, Abbey. “Warehouse Automation Explained: Types, Benefits & Best Practices.” *NetSuite*, 9 December 2020, https://www.netsuite.com/portal/resource/articles/inventory-management/warehouse-automation.shtml. Accessed 24 April 2023.

Kalogirou, Soteris, editor. *McEvoy's Handbook of Photovoltaics: Fundamentals and Applications*. Elsevier Science, 2017. Accessed 24 April 2023.

Lieshout, Laurensvan. “File:Selective laser sintering principle.png.” *Wikimedia Commons*, 5 September 2009, https://commons.wikimedia.org/wiki/File:Selective_laser_sintering_principle.png.

madsci1016. “Arduino-PS2X.” *GitHub*, 26 September 2011, https://github.com/madsci1016/Arduino-PS2X.

Mathews, Loren. “VEX Drive Trains.” *SlidePlayer*, https://slideplayer.com/slide/9319247/.

McGarey, Patrick, et al. “TSLAM: Tethered simultaneous localization and mapping for Mobile Robots.” *The International Journal of Robotics Research*, vol. 36, no. 12, 2017, p. 23.

McKinsey & Company. *Industrial Robotics*. July 2019, https://www.mckinsey.com/~/media/mckinsey/industries/advanced%20electronics/our%20insights/growth%20dynamics%20in%20industrial%20robotics/industrial-robotics-insights-into-the-sectors-future-growth-dynamics.ashx.

Mekid, Samir. “Schematic view of the linear slide. | Download Scientific Diagram.” *ResearchGate*, May 2000, https://www.researchgate.net/figure/Schematic-view-of-the-linear-slide_fig2_223018205.

“MinIMU-9 v5 Gyro, Accelerometer, and Compass (LSM6DS33 and LIS3MDL Carrier).” *Pololu*, https://www.pololu.com/product/2738.

“MinIMU-9 V5 - IMU Digital Combo Board - 9 Degrees of Freedom.” *Solarbotics Ltd.*, https://www.solarbotics.com/product/23226/.

“MPU-9255 Product Specification.” *InvenSense*, 14 September 2014, https://stanford.edu/class/ee267/misc/MPU-9255-Datasheet.pdf.

Nantel, Eric. “PS2 Versions & Connections.” *Lynxmotion*, 8 February 2023, https://wiki.lynxmotion.com/info/wiki/lynxmotion/view/servo-erector-set-system/ses-electronics/wireless/ps2-versions-connections/.

Nappi, Mattia L. “External forces on a simple car model.” *Wikimedia Commons*, 22 September 2012, https://commons.wikimedia.org/wiki/File:External_forces_on_a_simple_car_model.jpg.

National Hardware. “Automatic Gate Latch - Black N184-861.” *National Hardware*, https://www.national-hardware.com/p/21-automatic-gate-latches.

Paradis, Doug. “L298N Motor Driver Board – Drive Modes.” *dprg.org*, 10 January 2021, https://www.dprg.org/l298n-motor-driver-board-drive-modes/.

Porter, Bill. “PlayStation 2 Controller Arduino Library v1.0.” *The Mind of Bill Porter*, 5 June 2010, http://www.billporter.info/2010/06/05/playstation-2-controller-arduino-library-v1-0/.

Robot Shop. “Home.” *YouTube*, https://www.robotshop.com/products/hiwonder-ld-20mg-full-metal-gear-digital-servo-w-20kg-high-torque-aluminium-case?gclid=CjwKCAjw0ZiiBhBKEiwA4PT9z5356dvBhhJ30Lb1K9TJHxHY8iywShFBF283cTIOSEK4MzbRuVR1iBoCJ1AQAvD_BwE.

Rockey, Brendan. “File:Injection moulding.png.” *Wikimedia Commons*, 26 February 2009, https://commons.wikimedia.org/wiki/File:Injection_moulding.png.

Sarah. “How to Install a Self-Adjusting Gate Latch.” *Ugly Duckling House*, https://www.google.com/url?q=https://www.uglyducklinghouse.com/disclai mer/&sa=D&source=docs&ust=1682381637788658&usg=AOvVaw3N3MiI5HAPVQEJrcUkJHM6, 26 April 2016, https://www.uglyducklinghouse.com/self-adjusting-gate-latch/.

Savox. “MINI DIGITAL HIGH VOLTAGE SERVO 0.055/167@ 7.4V ALUMINUM.” *Savox USA*, https://www.savoxusa.com/products/savsv1260mg-mini-digital-high-voltage.

Tan, Ning, et al. “A Framework for Taxonomy and Evaluation of Self-Reconfigurable Robotic Systems.” *IEEE Access*, 09 Janurary 2020, https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8954702.

Tan, Ning, et al. “Nested Reconfigurable Robots: Theory, Design, and Realization.” *CORE*, International Journal of Advanced Robotic Systems, 9 February 2015, https://core.ac.uk/download/pdf/42578242.pdf.

“TB6612FNG.” *Toshiba*, 01 October 2014, https://toshiba.semicon-storage.com/info/datasheet_en_20141001.pdf?did=10660.

VectorNav. “What is an Inertial Measurement Unit? · VectorNav - IMU.” *VectorNav*, 2023, https://www.vectornav.com/resources/inertial-navigation-articles/what-is-an-inertial-measurement-unit-imu.

“VEX Drivetrains.” *BLRS Wiki*, https://wiki.purduesigbots.com/hardware/vex-drivetrains. Accessed 10 March 2023.

VEXTA. “SMK216A-GN / 2GN3SA 2.22 in. (56.4 mm) Low-Speed Synchronous Motor.” *AC Motors*, https://www.google.com/url?q=https://www.orientalmotor.com/company/terms-of-access.html&sa=D&source=docs&ust=1682385352946711&usg=AOvVaw2tnJmUX5J-cxrOvW-LyQzR, https://catalog.orientalmotor.com/item/single-phase-low-speed-synchronous-motors/60mm-smk-low-speed-synchronous-motors/smk216a-gn-2gn3sa

Voss, Wilfried. “Controller Area Network (CAN Bus) - Physical Layer And Bus Topology - Copperhill.” *Copperhill Technologies*, 10 June 2019, https://copperhilltech.com/blog/controller-area-network-can-bus-physical-layer-and-bus-topology/. Accessed 24 April 2023.

Woland, Messer. “Hovercraft scheme.” *Wikimedia Commons*, 7 April 2007, https://commons.wikimedia.org/w/index.php?search=hovercraft+diagram&title=Special:MediaSearch&go=Go&type=image. Accessed 24 April 2023.

Yim, Mark, et al. “Modular Robots.” *IEEE Xplore*, February 2002, https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=981854. Accessed 24 April 2023.

Appendix B: Mechanical Engineering Design Competence Evaluation

Project Title: Hive: The Grounded Swarm   Semester: Spring 2023

ME Design Areas	Critical/Main contributor	Strong contributor	Necessary but not a primary contributor	Necessary but only a minor contributor	Only a passing reference	Not Included in this Design Project
Thermal-Fluid Energy systems					X	
Machines & Mechanical Systems		X				
Controls & Mechatronics		X				
Materials Selection			X			
Modeling & Measurement Systems				X		
Manufacturing		X				

Mechanical Topics Utilized in this Senior Design Project:

Topic	Criticality to Project	Section and Page(s)	Comments
Thermal-Fluid Systems	Only in passing reference	IV.1 Pg. 109	Possibility for batteries to expel harmful gasses that may need to be ventilated.
Machines & Mechanical Systems	Strong contributor	III.3 Pg. 52-56, III.4 Pg. 87-99	We will be constructing a minimum of two Agents that will manipulate a controlled environment.
Controls & Mechatronics	Strong contributor	II.2 Pg. 5-6, III.2 Pg. 28-40, III.3 Pg. 57-70, V.2 Pg. 128-132	The main control system will be cloud based (Worked on by our software counterparts) ,however, embedded code will be needed to take commands from the cloud and in turn move the agents accordingly.

Topic	Criticality to Project	Section and Page(s)	Comments
Materials Selection	Necessary but not a primary contributor	V.2 Pg. 123-124	When manufacturing mecanum wheels we had to evaluate different materials for the rollers that would best fit our needs.
Modeling & Measurement Systems	Necessary but only a minor contributor	VI Pg. 141-151	Measurement systems will be used to validate our physical designs.
Manufacturing	Strong contributor	III.2 Pg. 42-47, V.2 Pg. 126-127, V.3 Pg. 133	Along with proper material selection, the correct manufacturing method must be picked when designing the rollers for the mecanum wheels