Final Design Report

MAE 162E – Mechanical Engineering Design 1

Group Number 7

Ahan Agarwal,
Joseph Cisneros,
Siddhant Singh,
Allan Ta,
Jacob Ursua

Instructor: Tsu-Chin Tsao

TAs: Will Shih, Toby Chen

Contents

1.0 Introduction

The need to transport water has never gone away, as is seen in all kinds of work, such as restaurants, possibly during meetings, and even transporting buckets of water/grain up hills. Additionally, the need for a machine such as this can also be applied to the home to aid the elderly and disabled. It could even help out the pets by filling up their water/food bowls.

1.1 Design Requirements

For the integrity of the electronics, the robot must have: an easily rechargeable or replaceable battery, can control its speed to prevent spillage of the carried content, and plastic encasing and/or water-resistant casing in the event of a spill. Secondly, the arm is what will do the heavy lifting so it must be able to grab three different sizes of cups filled up to 100% maximum (transporting 100 mL, 50 mL, and 25 mL of content), moving at a top speed of about 1 m/s and vertical translation via a linear actuator (with a sensor attached to see possible cups), along with the grabbing mechanism having rotational translation to pour the water. This arm will require sensors to detect the distance between it and the cups and to ensure the claw has firmly grasped the cups. The finality of the robot is to pick up a container, relocate it, pour the contents into another container, place the cup, and repeat. See the state block flow diagram below for motions and operations during performance.

1.2 Conceptual Design

Our design will consist of three motions: rotation, vertical translation, and gripping. For pouring rotation, JX Servo PDI-6221MG - 6V 20KG 180° 360° Large Torque Digital Aluminium Metal Gear 25T Arm for RC Car Model Toy Crawler Boat Airplane ($21). We are considering the Ultrasonic Sparkfun Sensor for the arm for $10 to aid in pouring. For gripping, we will include the MG996R which is roughly $30, and TETRIX MAX Gripper Kit ($14).

For vertical translation, KAIBRITE 200mm GGP Ball Screw Linear with Nema 23 Stepper Motor, Linear Guide Slide Table Ball Screw Motion Rail CNC Linear Guide Stage Actuator is used ($59) along with EASON stepper motor driver TB6600 ($11).

2.0 Product Requirements and System Behavioral Analysis

2.1 System Overview

The proposed system consists of a rover equipped with a robotic manipulator designed to autonomously or semi-autonomously detect, pick up, transport, and pour water. The rover base is a mobile platform with an integrated drive system, onboard power, and sensing hardware for navigation. Mounted on this base is a robotic manipulator, a three-degree-of-freedom (DoF) arm equipped with a gripper and a pouring mechanism. The arm is securely attached to the rover chassis, positioned near the front to provide an unobstructed workspace.

The robotic manipulator is designed with three degrees of freedom to accommodate various container sizes and pouring orientations. These include a linear actuator to raise and lower the end-effector, gripper actuation which opens and closes to grasp cups, and a rotational joint to control the pouring process. The end-effector features a force-feedback gripper to accommodate multiple container diameters (50 mL, 100 mL, and 200 mL), ensuring a gentle yet secure grip. Integrated distance sensors or limit switches assist in detecting cupcontact.

The system operates through four primary functions: detection, pick-up, transport, and pouring. Detection is facilitated by ultrasonic, infrared, or LiDAR-based sensors that scan the environment to distinguish target containers from other objects. Once a container is detected, the manipulator aligns the gripper and uses force feedback to close around it, ensuring secure handling without excessive pressure. The rover then navigates autonomously or semi-autonomously to a designated location, such as a basin or table, while stabilizing the container to minimize spillage. Finally, the manipulator rotates the container to a specified angle, pouring the water into the target receptacle efficiently and with minimal waste or splashing.

2.2 Key Design Requirements

2.2.1 Power and Electronics Protection

The system is powered by a rechargeable or replaceable battery, specifically a lithium-ion or NiMH pack, which provides sufficient capacity to operate the drive motors, actuator servos, and control electronics for extended periods. The battery mounting system is designed to allow quick swaps or easy recharging without requiring major disassembly. To prevent spillage during transport, motor controllers enable smooth acceleration and deceleration, reducing abrupt movements that could destabilize containers. Feedback from wheel encoders allows for closed-loop speed regulation, ensuring controlled motion. Additionally, the system includes a water-resistant covering for electronics.

2.2.2 Robotic Arm Capabilities

The robotic arm is specifically designed to handle three standard container sizes: 50 mL, 100 mL, and 200 mL. The adaptive gripper features finger-like jaws that adjust to grip various container diameters while incorporating force sensors to maintain a firm yet non-damaging hold. Vertical translation is achieved through a linear actuator mechanism, which raises and lowers the gripper assembly to accommodate different surface heights. Position sensing is provided through rotary encoders or linear potentiometers, ensuring precise and repeatable vertical alignment.

For enhanced handling, the gripping mechanism integrates distance sensing. Distance sensors, such as ultrasonic modules, measure the gap between the end-effector and the container, assisting in precise final alignment. A linear actuator, equipped with an encoder for closed-loop control, provides smooth vertical translation, allowing the end-effector to accommodate varying container heights. The rotational pouring mechanism is powered by a dedicated servo motor or a motor-gear assembly that tilts the container from an upright position to the desired pouring angle. Angular feedback from the motor ensures consistent and controlled pouring, with predefined speed limits to prevent rapid tilts that could cause spills.

2.2.3 Mobility and Motion Control

The rover operates autonomously, leveraging navigation algorithms for movement and obstacle avoidance. Autonomous navigation relies on predefined waypoints and real-time path planning based on sensor feedback.

For navigation and obstacle detection, the rover is equipped with front-facing range sensors, including ultrasonic and LiDAR modules mounted on the front C-beam. These sensors feed data into the avoidance and path-planning algorithms. The system employs sensor fusion, combining multiple data sources (ultrasonic, infrared, and LiDAR) to improve accuracy and reliability across varying lighting and environmental conditions.

The rover's mobility is defined by three degrees of freedom: forward/reverse translation, pivot/steering, and lateral slip. Forward and reverse motion is controlled by the rear wheels, enabling straightforward movement. Pivoting and steering are managed through differential control or a separate steering mechanism on the rear wheels, allowing rotation in place. The integrated omni-wheels provide the rover with an additional degree of lateral slip, improving maneuverability in tight or cluttered spaces.

3.1 Narrative Analysis

The rover will interact with the environment using a feedback loop to arrive at its destination and to record its initial and previous positions. The rover will take in its surroundings with the provided sensors and adjust accordingly. For instance, if there is an obstacle, the robot will idle, rotate its sensor until it finds an open path, and begin to move. If moving towards a previously marked path, the rover will note that and not move any further, backtracking to the point where the obstacle was found, and then search for another path.

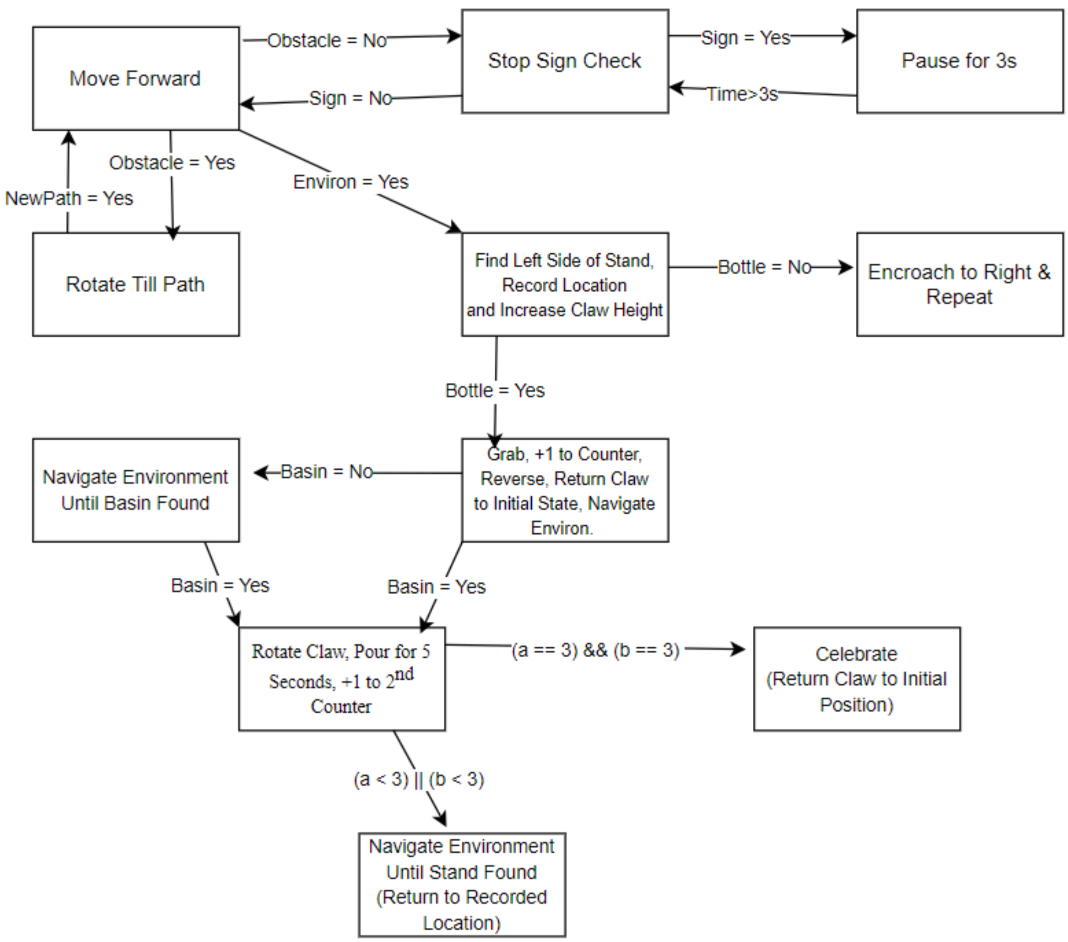
Assuming the rover has made it to the environment, it will once again have to navigate, but this time, without obstacles. It will mark the location right after the stop sign and then note to itself that it cannot pass 6 feet from that location in any direction in front of it, sweeping 180°. With these limitations, the robot will start traversing the environment until it finds a wall/obstacle. It will approach this wall and find the left end of the tower, recording the position of the tower so that it may return to it later. From there, the manipulator will raise vertically 2 inches. If the sensor on the claw detects an object, the bot will drive 2 inches forward and clamp down on the object (and add one to counter a). The bot will then reverse 2 inches and lower the claw back to its initial position. From there, the rover will navigate the environment once more, until it locates a basin. Once found, the bot will bring itself two inches from the basin, rotate the claw 135°, and hold for 7.5 seconds. It will then return the claw to 0°, and place the cup to the right of the basin, adding +1 to the basin counter. Next, the cup will return to the recorded position of the wall holding the cups. It will then move to the right 3 inches. It will then repeat the previous process until both counters are equal to three.

Next is error handling (e.g., container not found, insufficient grip, spillage). If the container is not found, the rover will move forward until it detects an obstacle. If found, the rover will rotate until there is no longer an obstacle. It will stay within the defined parameters of the environment until it can find a long continuous wall, then repeat the aforementioned process to find the cup atop the wall.

If there is insufficient grip, the bot will not stop and re-grip the cup. Instead, the bot will need to let itself know that the force exerted for gripping is changing beyond the parameters allowed and will need to increase such force.

In the case of spillage, the bot will carry out the basin task as if there was none. Additionally, if the cup fell out of the claw's grip, the dropped cup will be treated as a new obstacle. Counters a & b will still be increased by one by the end of the circuit.

3.2 State Diagram



-

Figure 1: State flow diagram of rover navigation and manipulator

3.0 Mechanical Design and Analysis

3.1 Narrative Design Analysis

The Rover's design was chosen to have two linear driving wheels and two omniscient wheels, whilst being rear wheel drive to aid in steering. The rover consists of an electronics plate in the middle of the rover for ease of access if the electronics need relocating. For manipulation, there are three motors: one to translate the linear actuator vertically, another to operate the gear claw, causing clamping, and the last to rotate the claw, along with the cup held. For feedback, the rover will consist of 3 sensors: one for navigation around the track, a second to locate and avoid obstacles, and the final to locate the cups/basins.

Due to the relative ease of access, aluminum and 3D-print are used as the chassis material and supporting components. The two ultrasonic sensors are used to avoid obstacles, along with locating, grabbing, transporting, and releasing cups. The third Lidar sensor is used for navigation through the course.

Figures 2 and 3 show the complete rover assembly with the track. The cups and the glasses will be placed at different heights using 3D-printed steps. The cups and glasses themselves too will be 3D printed for consistency. The rover will first navigate to the cups using the LIDAR and the ultrasonic sensors. Once the cup and its height are detected, the rover will rotate 180 degrees. Next, it will grab the cup and lift it off the steps using the linear actuator and motors in the claw.

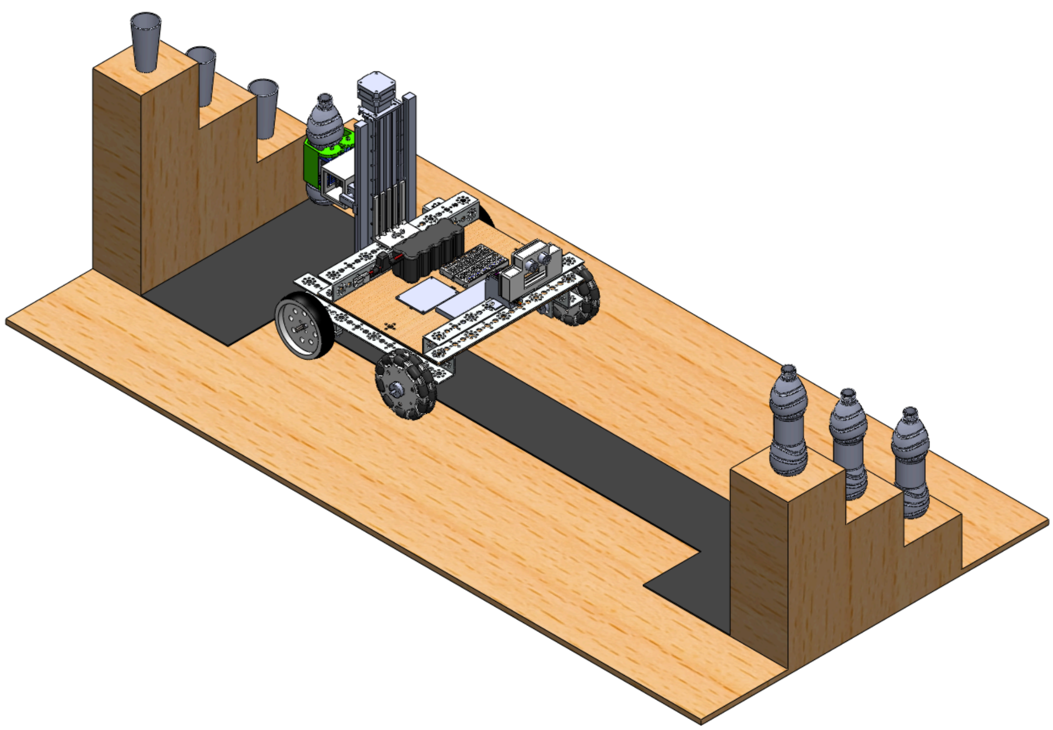


Figure 2: Back Isometric View: Environment with cups and cups

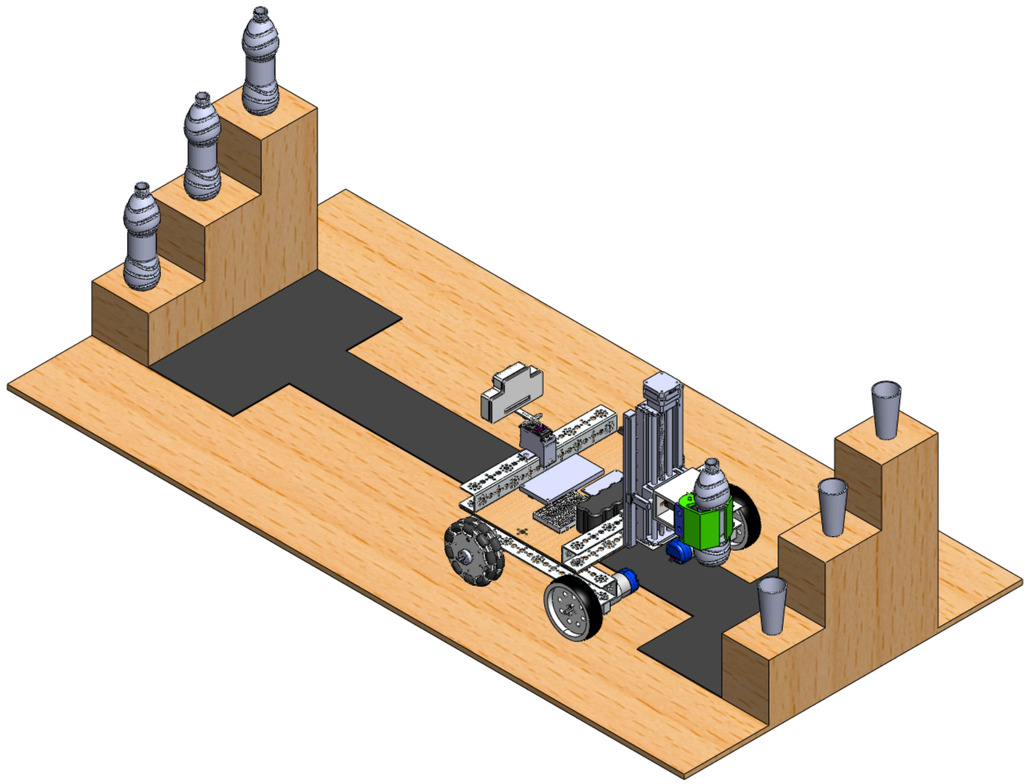


Figure 3: Front Isometric View: Environment with cups and cups

Figure 4 shows the rover navigating through to arrive at the target location- the glasses. Once the glass to transport the water has been chosen, the rover will navigate to the glass using the same ultrasonic and LIDAR sensors. After taking a 180-degree turn, the rover will tilt the cup using the servo motor in the gray box of the manipulator. The water will therefore be poured into the glass thereby completing the task.

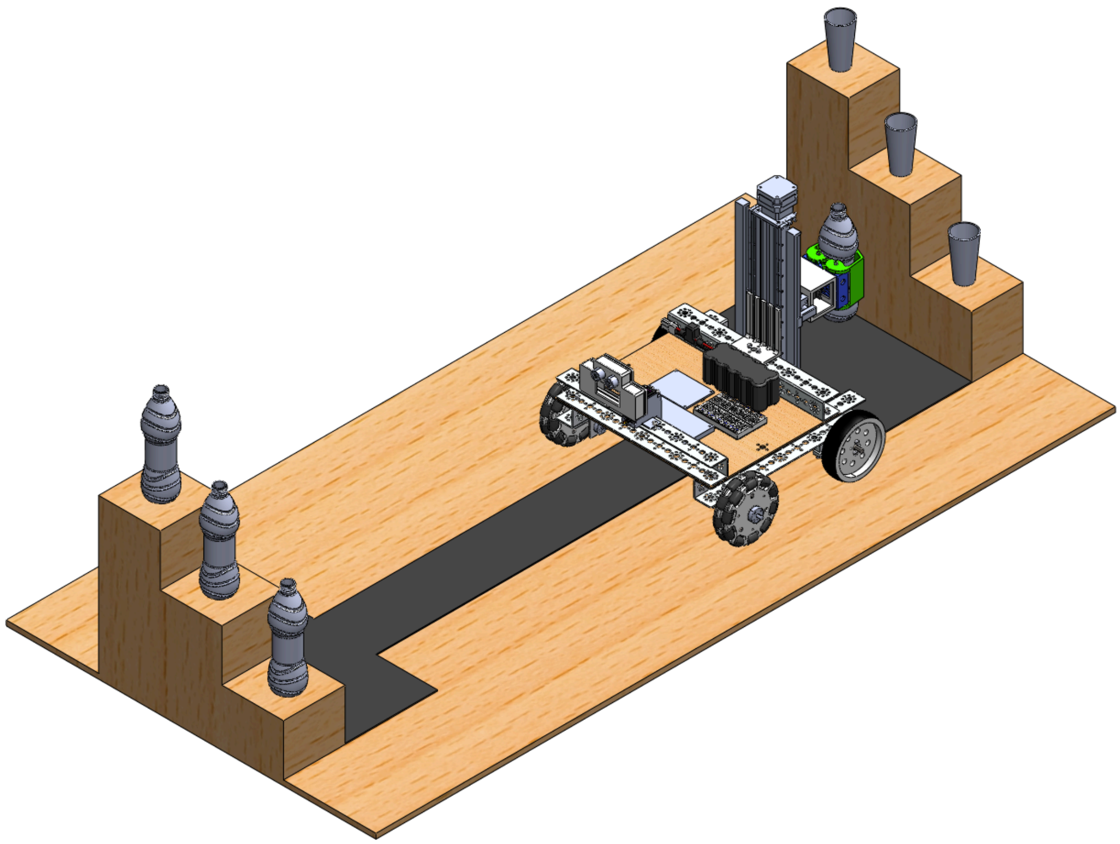


Figure 4: Rover carrying a cup on the claw to deliver it to the cups

3.2 CAD Drawings and System Assemblies

Overall System Assembly (Rover + Manipulator)

Figure 5 shows the complete rover assembly with the manipulator attached. The figure shows the front of the robot with the navigating sensors including the ultrasonic and lidar sensors. Figure 6 shows the back of the rover with the manipulator. The manipulator consists of 2 servo motors and 1 linear actuator stepper motor. The linear actuator is connected to the chassis using a 3D-printed part that screws into the c-channels as well as the nuts in the back of the linear actuator to ensure a firm connection. This is important since this will bear the moment of the entire manipulator including the cup with the water.

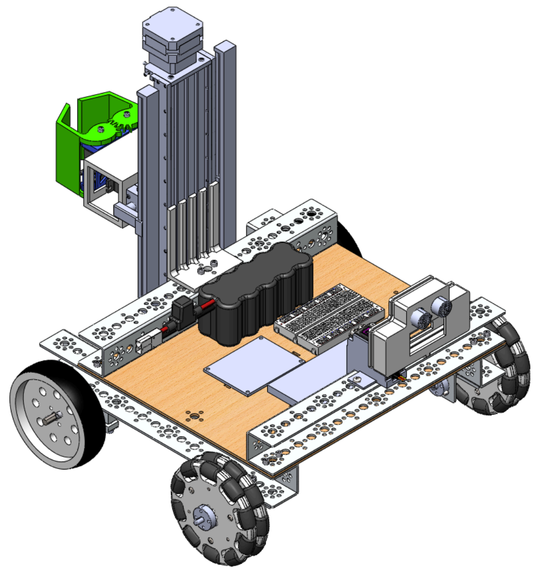


Figure 5: Full Assembly- Front of the rover with the manipulator

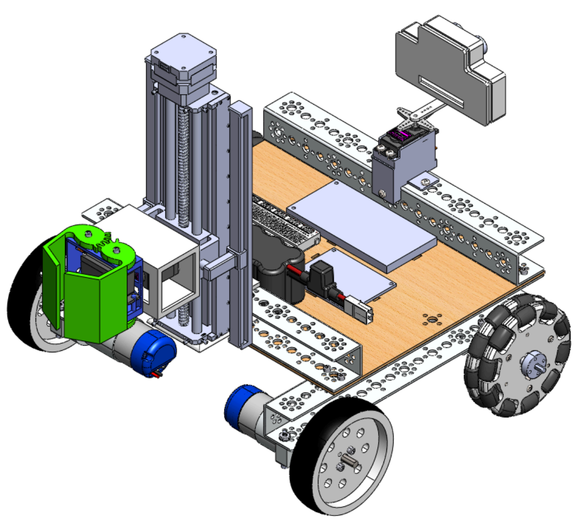


Figure 6: Full Assembly- Back of the rover with the manipulator

Rover Subsystem Assembly (Rover + Electronics)

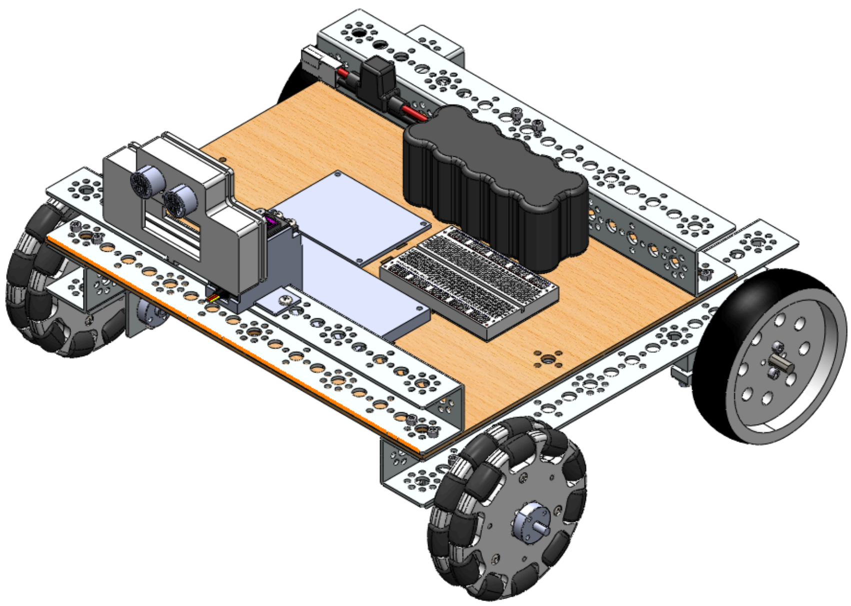


Figure 7: Assembly of rover and electronics

The Rover's design was chosen to have two linear driving wheels and two omniscient wheels, whilst being rear wheel drive to aid in steering. The rover consists of an electronics plate in the middle of the rover for ease of access if the electronics need to be rearranged. A navigation and control sensor for avoiding obstacles is located on the frontmost C beam on a motor to help the rover look around. The electronic plate was the first laser-cut acrylic equipped with holes for the electronic components. However, later the plate was replaced with laser-cut birch wood to make sure that the plate does not crack due to the stress on the screw holes.

Manipulator Subsystem Assembly (Mechanisms, Actuators, Sensors)

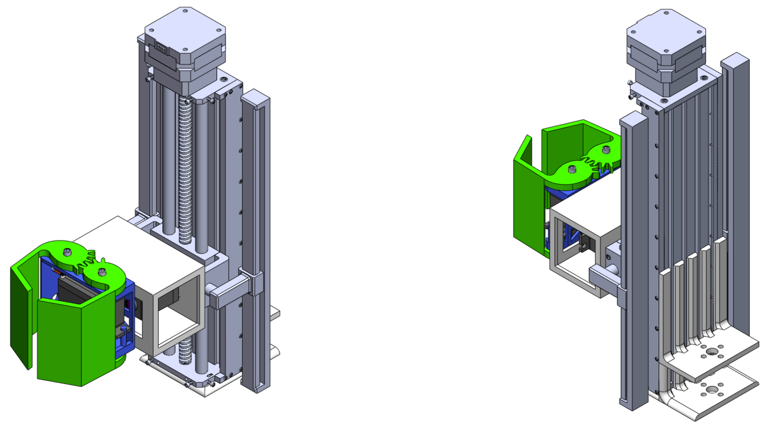
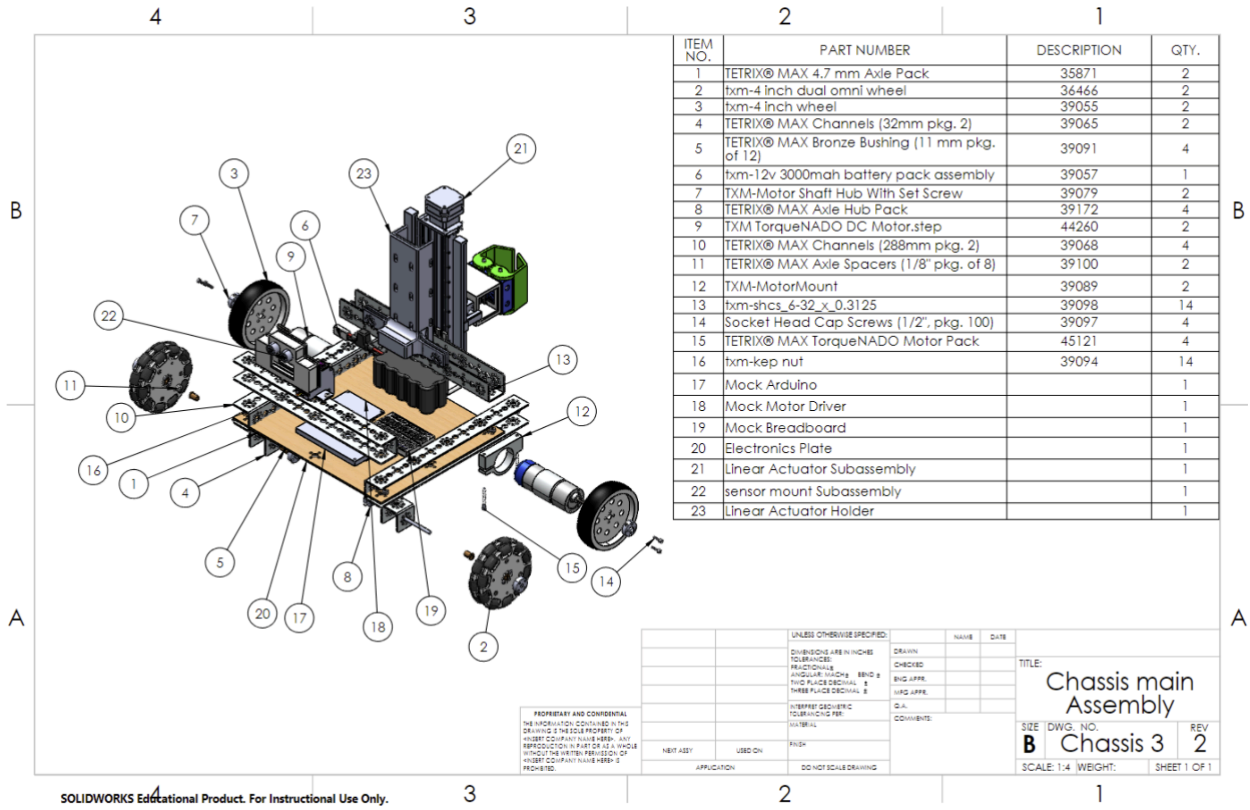


Figure 8: Assembly of manipulator

The manipulator consists of the main linear actuator, a cage to hold the pouring motor, and the gear claw to grab cups. The vertical translation system uses a linear actuator to move the claw up and down. A sensor will be attached to the top of the motor cage to help locate and position to grab a cup or pour a cup's contents into a cup. If the sensor reaches a cup with none in hand, the rover will move towards the cup and the claw will grab it. If the sensor sees a cup and the claw has a cup in hand, the manipulator will move to pour using the pouring motor.

Chassis BOM



ITEM NO.	PART NUMBER	DESCRIPTION	QTY.
1	TETRIX® MAX 4.7 mm Axle Pack	35871	2
2	txm-4 inch dual omni wheel	36466	2
3	txm-4 inch wheel	39055	2
4	TETRIX® MAX Channels (32mm pkg. 2)	39065	2
5	TETRIX® MAX Bronze Bushing (11 mm pkg. of 12)	39091	4
6	txm-12v 3000mah battery pack assembly	39057	1
7	TXM-Motor Shaft Hub With Set Screw	39079	2
8	TETRIX® MAX Axle Hub Pack	39172	4
9	TXM TorqueNADO DC Motor.step	44260	2
10	TETRIX® MAX Channels (288mm pkg. 2)	39068	4
11	TETRIX® MAX Axle Spacers (1/8" pkg. of 8)	39100	2
12	TXM-MotorMount	39089	2
13	txm-shcs_6-32_x_0.3125	39098	14
14	Socket Head Cap Screws (1/2", pkg. 100)	39097	4
15	TETRIX® MAX TorqueNADO Motor Pack	45121	4
16	txm-kep nut	39094	14
17	Mock Arduino		1
18	Mock Motor Driver		1
19	Mock Breadboard		1
20	Electronics Plate		1
21	Linear Actuator Subassembly		1
22	sensor mount Subassembly		1
23	Linear Actuator Holder		1

Figure 9: Chassis 3

Chassis 3 BOM lists all the parts used. The drawing allows assemblers to see where every part goes and what part numbers to use.

Sensor + Motor Mount Subassembly

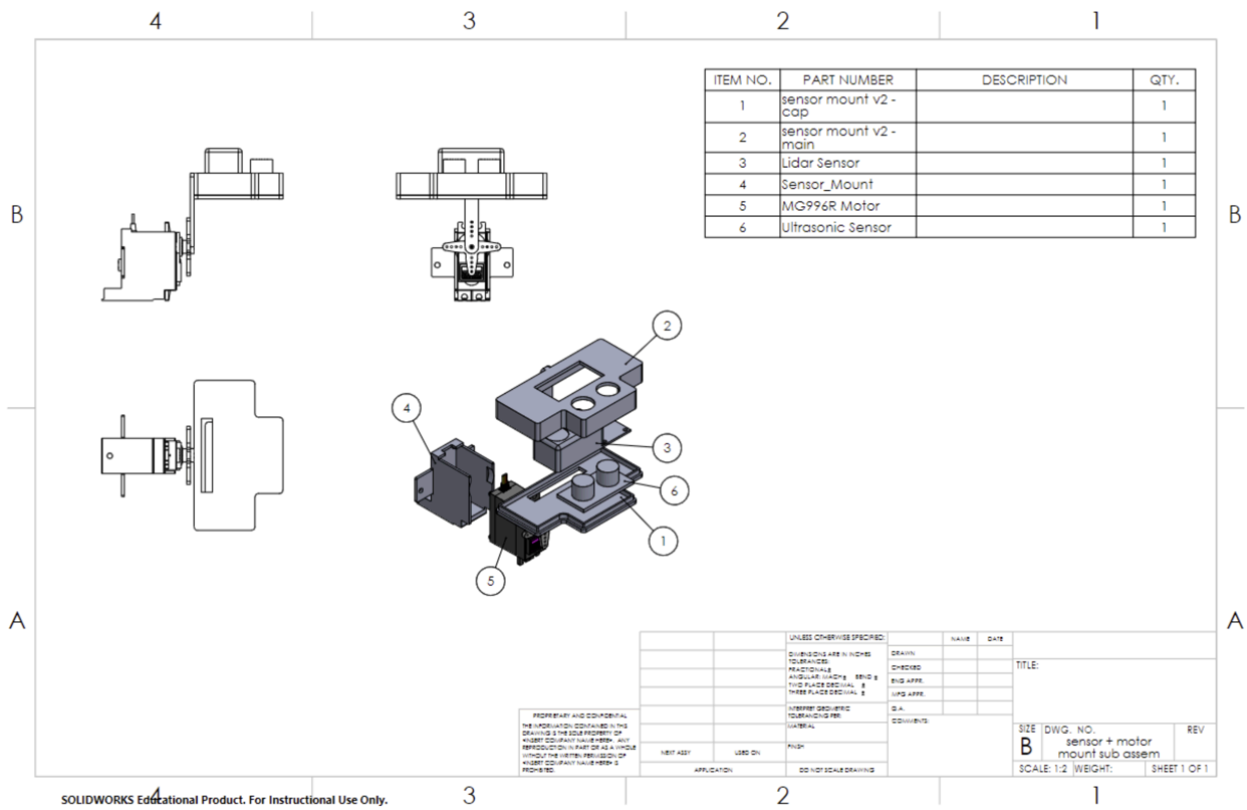


Figure 10: Sensor + motor mount sub assem

The Sensor + Motor Mount Subassembly houses the ultrasonic sensor and lidar used to help navigate the course. It is mounted on the front center of the robot. With the motor, the sensors can sweep 180 degrees for obstacle avoidance.

Sensor Mount Cap

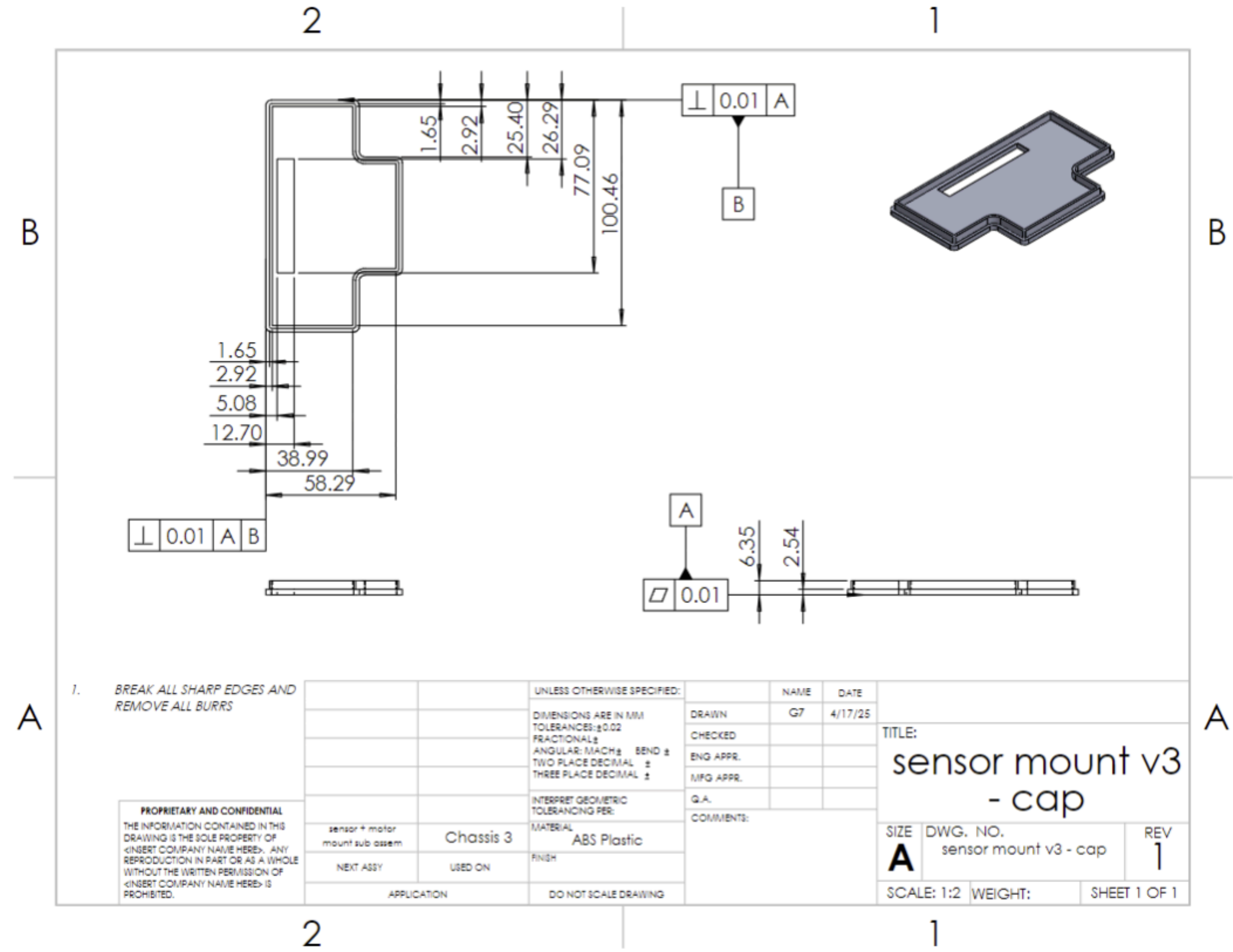


Figure 11: Sensor Mount Cap

Form: The sensor motor cap boasts a height of 58.29 mm, length of 100.46 mm and depth of 6.35 mm. It features a box opening so that the wires can connect to the ultrasonic sensor and/or lidar, with the lidar being below and the sensor above.

Fit: It mounts easily to Sensor Mount Main where it is slid on the back.

Function: The sensor mount cap primary purpose is to ensure that the ultrasonic sensor and lidar remain secure and void of possible FOD.

MG996R Base

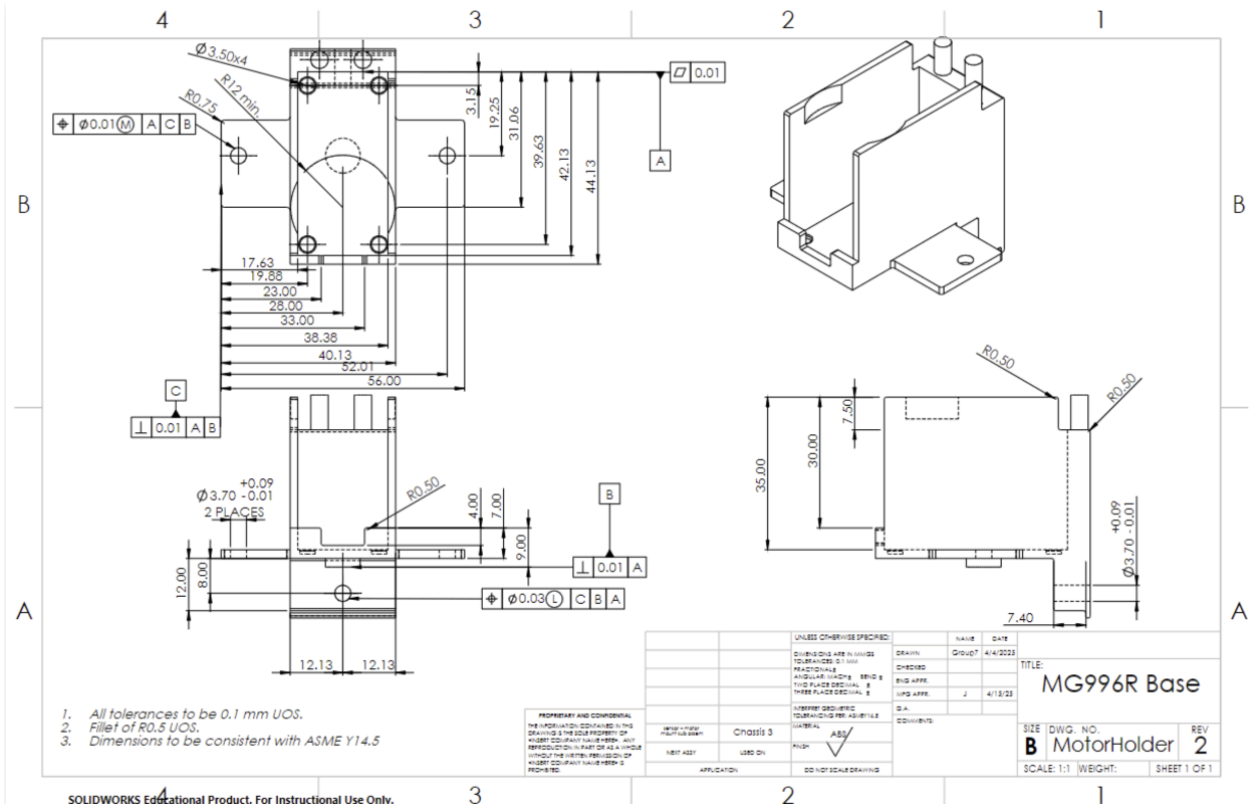


Figure 12: Mg996R base

Form: The Mg996R Base has a height from the base of 35 mm, width of 56 mm, and depth of 44.13 mm.

Fit: The has two wings with holes made to align with the whole of the C-Beam, allowing for screws and nuts to be placed in and secure it. Additionally, it has an inside design to perfectly fit the MG996R servo motor, along with two prongs in the back that go through the servo motor's openings for extra security. The circle at the bottom is made to let the assembler know where on the C-Beam it should be centered and prevents movement along the x-axis.

Function: The MG996R Base holds the servo motor, along with prongs that holds the servo motor securely in place during the robot's movements, providing a solid foundation for torque transmission. Due to the securement, the servo's rotation will remain constant, allowing for consistent rotational sweeps for the ultrasonic sensor and lidar.

This item was removed due to preventing low object readings.

Servo Mount (MG996R Base)

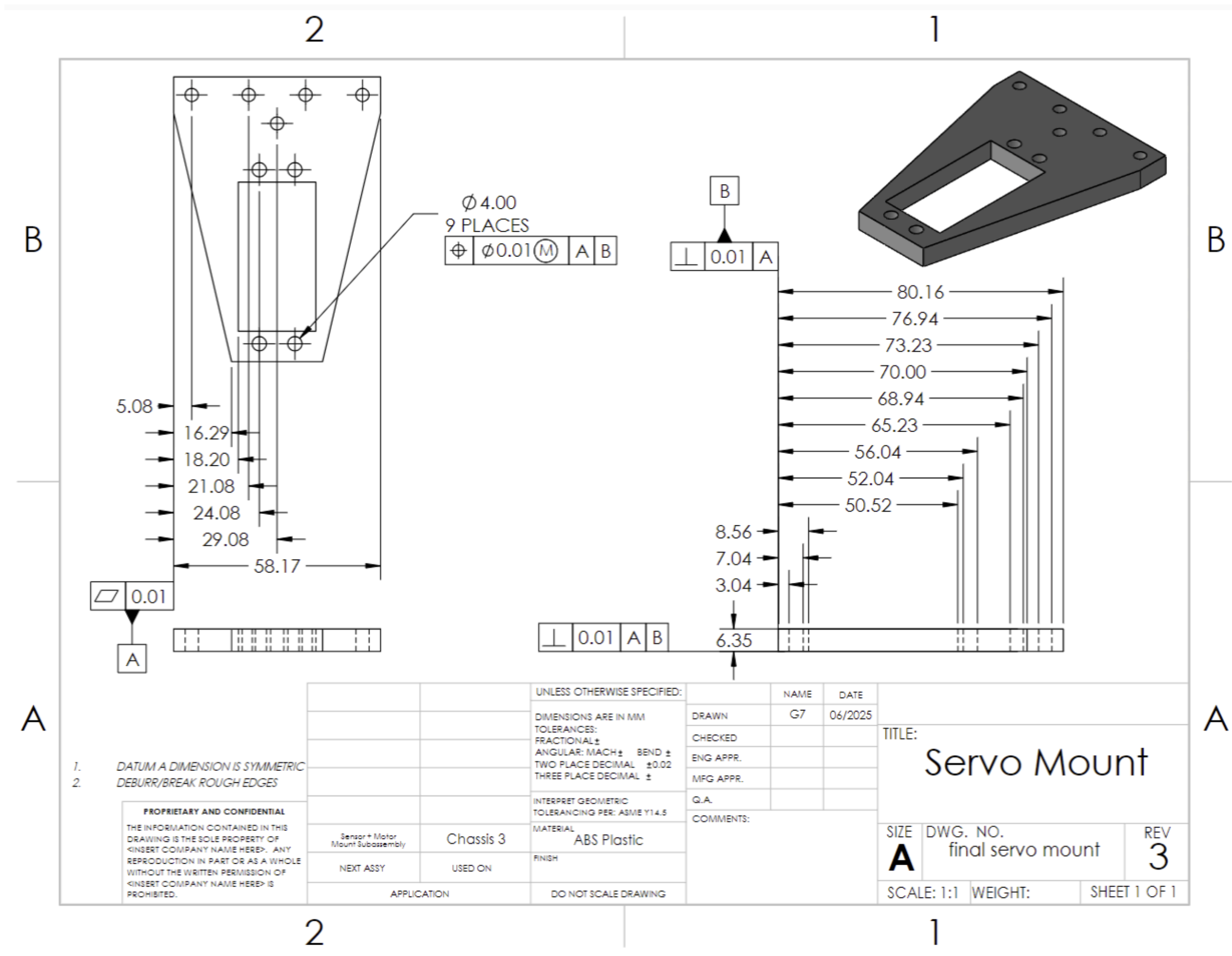


Figure 13: Servo Mount

Form: The third iteration of the servo mount has a width of 58.17 mm, a length of 80.16 mm and a depth of 6.35 mm.

Fit: It sits on top of the C-Beam with a rectangular cutout to slot the mg996r, along with holes matching the diameter of the C-Beam's holes for screws.

Function: The third iteration was designed to detect items closer to ground level. In our case, the previous design had the servo motor propped up, atop the C-Beam, and so it would detect only a small portion of the cone as detected. The top of the servo motor now being level with the top of the C-Beam remedied the titanic like behavior of only viewing the tip of the iceberg.

LIDAR Navigator Mount

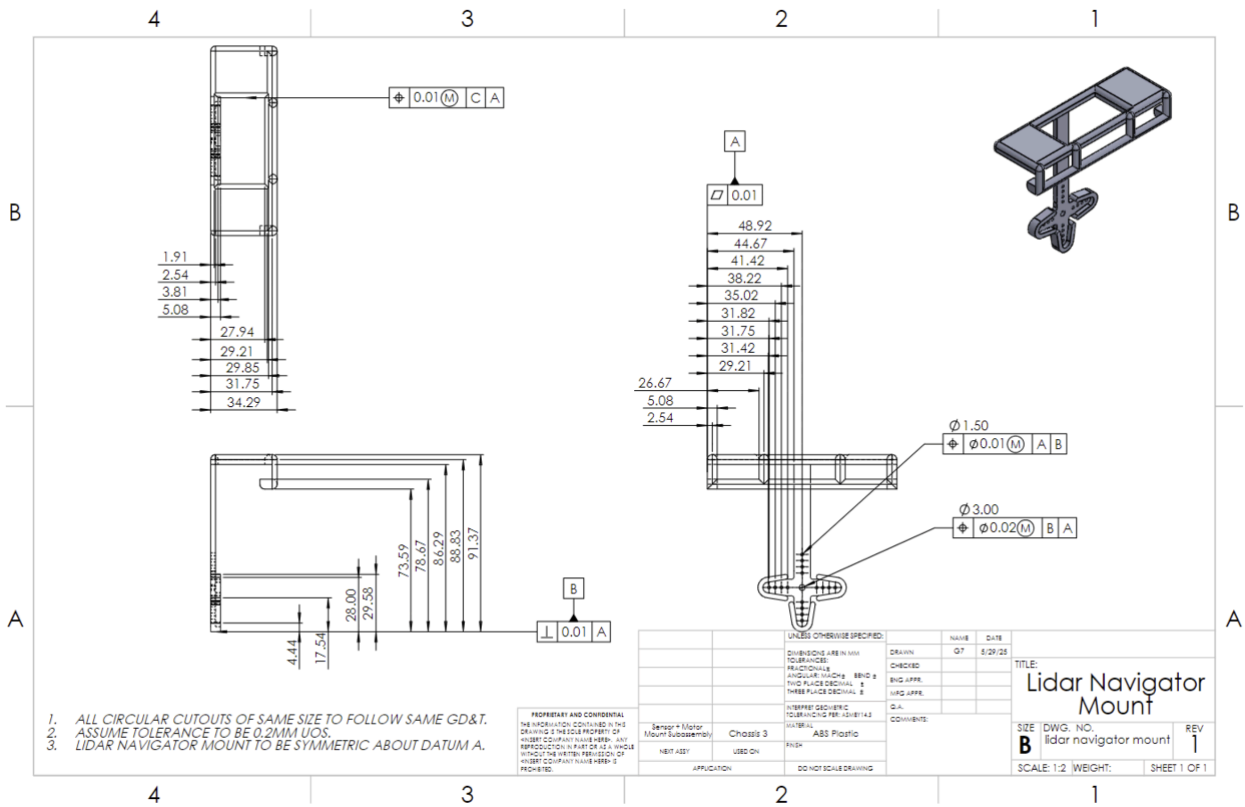


Figure 14: LIDAR Navigation Mount

Form: The LIDAR Navigation Mount boasts overall dimensions of 91.4 x 48.9 by 34.3 mm.

Fit: It is capable of wrapping around the MG996r Arm pad, and allows for connection to the sensor mount cap. Additionally, it houses the LIDAR.

Function: The LIDAR Navigation Mount extends further out so that it does not interfere with the C-Beam, allowing for a smooth sweep, while having a snug fit around the sensor. It also points down so that it can detect any possible obstacles and avoid them appropriately.

Sensor Mount Main

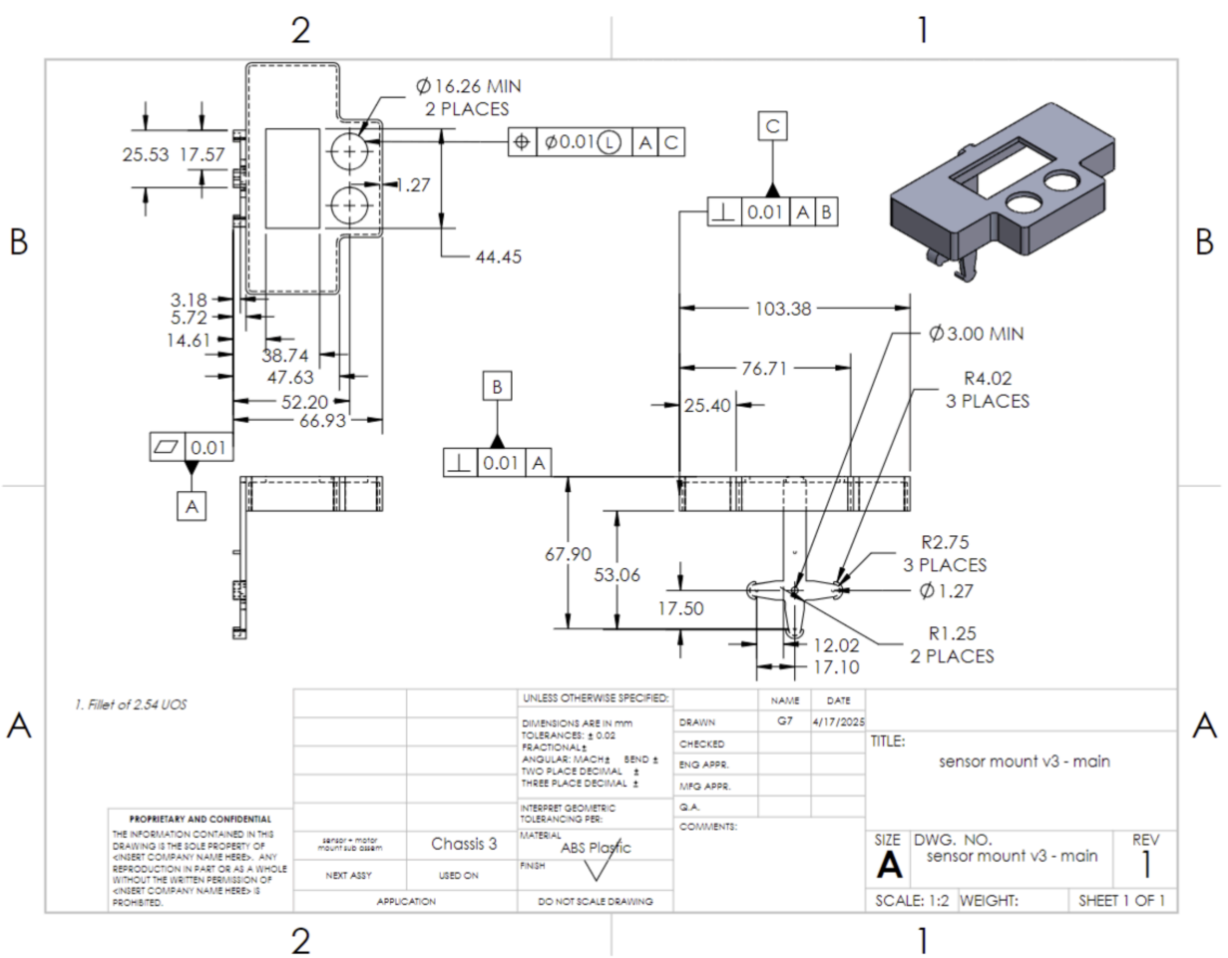


Figure 15: Sensor Mount Main

Form: The sensor mount has a height of 67 mm, width of 103 mm, and depth of 68 mm.

Fit: It can wrap around the MG996r Arm pad, and allows for connection to the sensor mount cap. Additionally, it houses the ultrasonic sensor and lidar.

Function: the sensor mount main extends further out so that it does not interfere with the C-Beam or any other parts, allowing for a smooth sweep, while having a tight fit around the sensors to “see” from.

This design was not used in the final product, as we moved the ultrasonic sensor for cup detection.

Arm 4 Pad

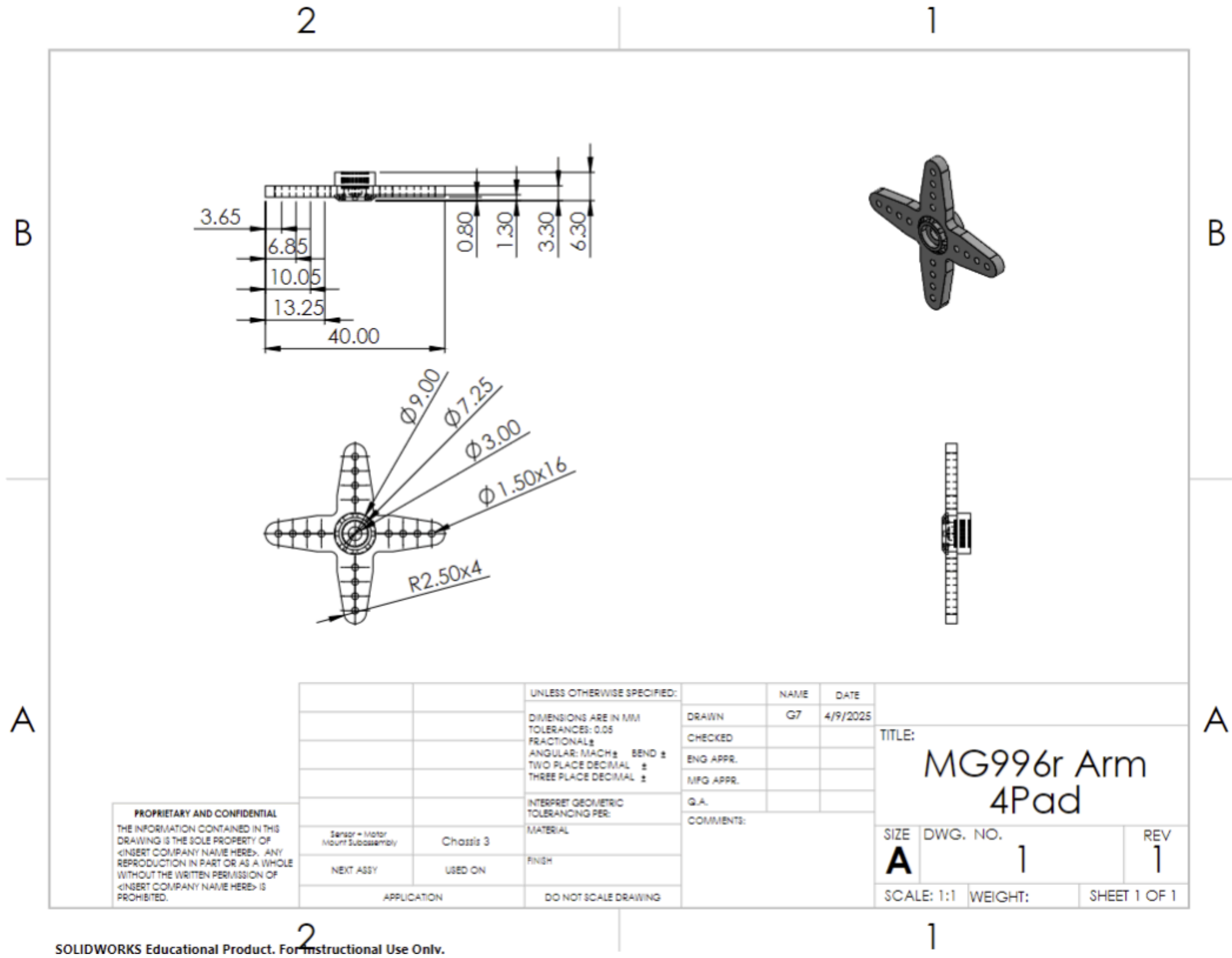


Figure 16: Mg996r Arm Pad

The MG996r Arm 4Pad is a cross shape with a 40 mm wingspan. The center piece is made to fit on the servo motor and the circular cutouts surrounding it are to allow for an item to be attached. It should be purchased and come with the mg996r servo motor.

Manipulator Subassembly

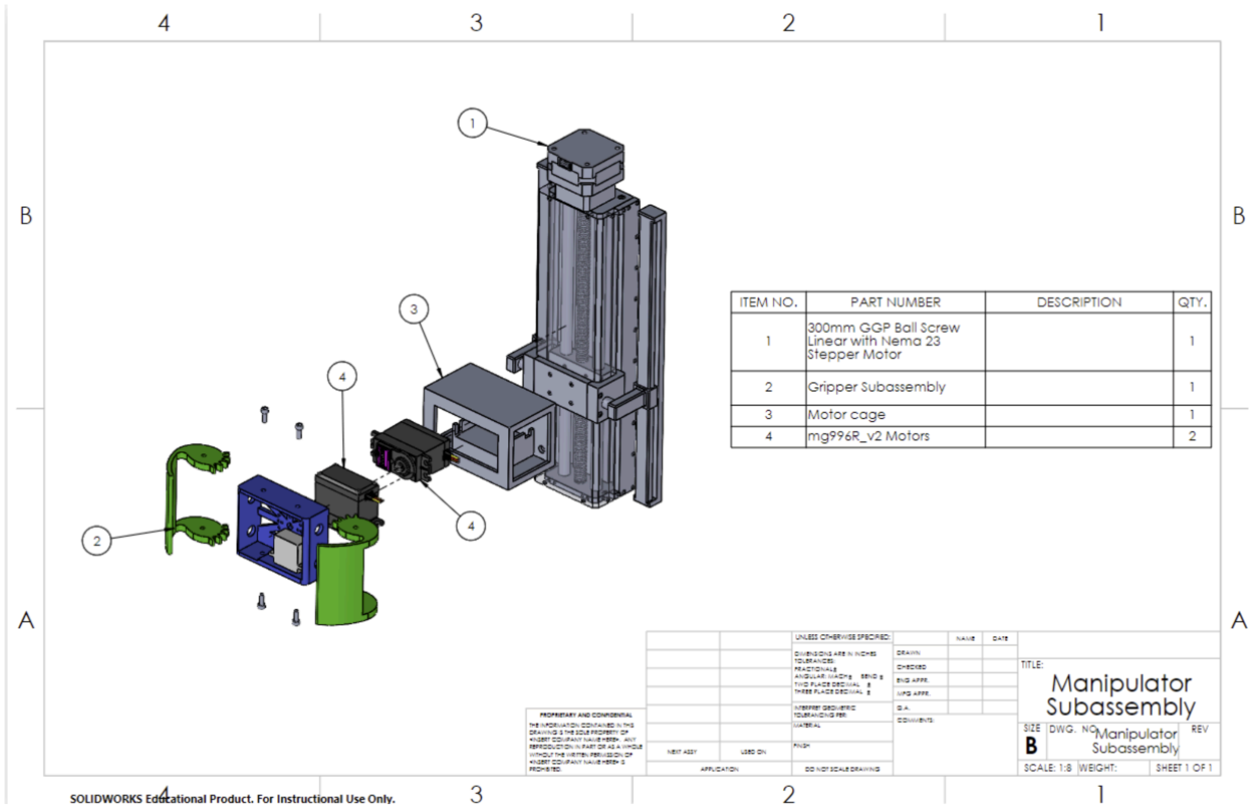


Figure 17: Manipulator Subassembly

The manipulator portrays to the assembler how the linear actuator, gripper subassembly, and two motors are connected.

Linear Actuator

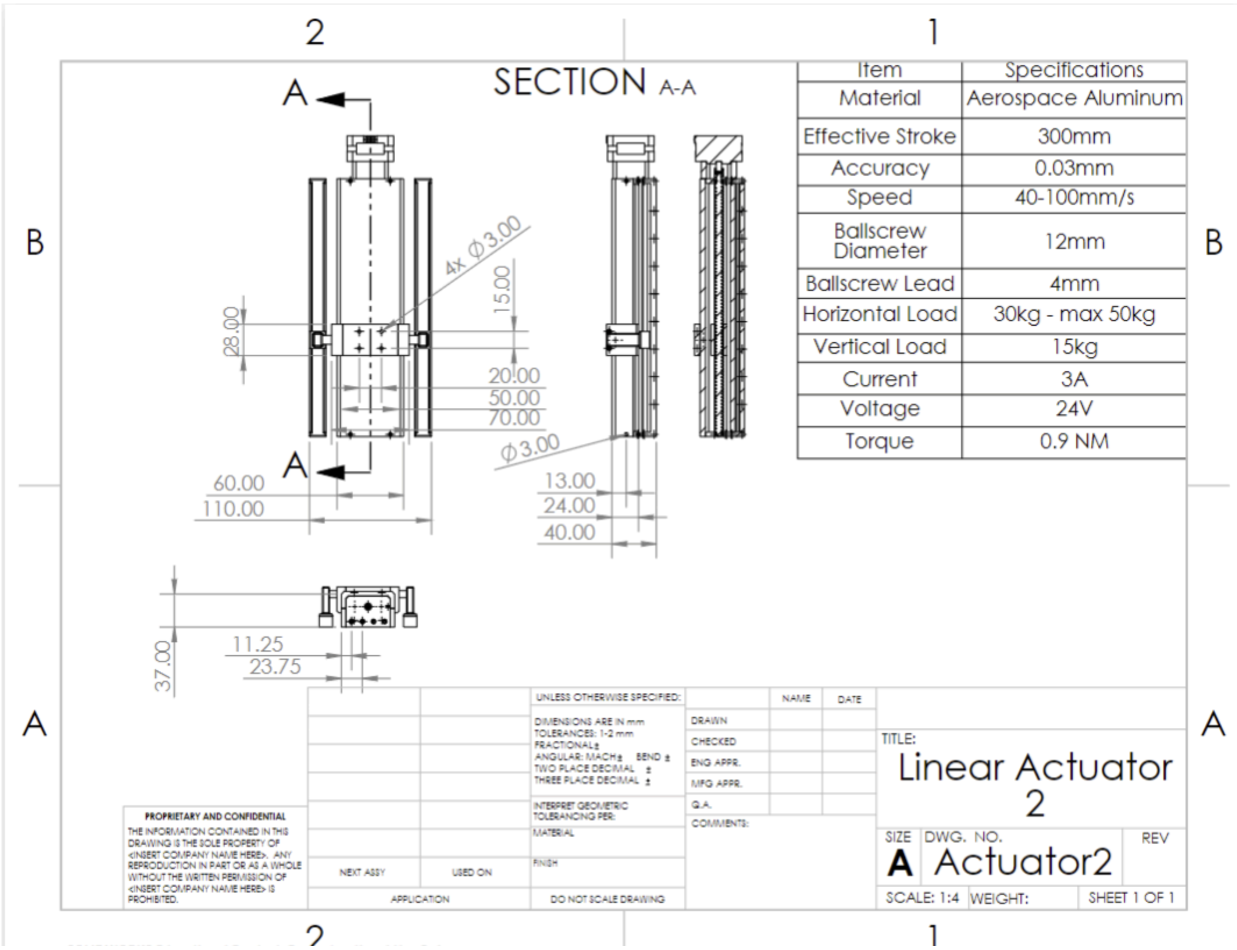


Figure 18: Linear Actuator 2

The linear actuator is to be purchased and it is made of Aerospace Aluminum. It is capable of vertical translation and will serve to raise and lower the claw so that it may complete its task at differing heights.

Linear Actuator Holder

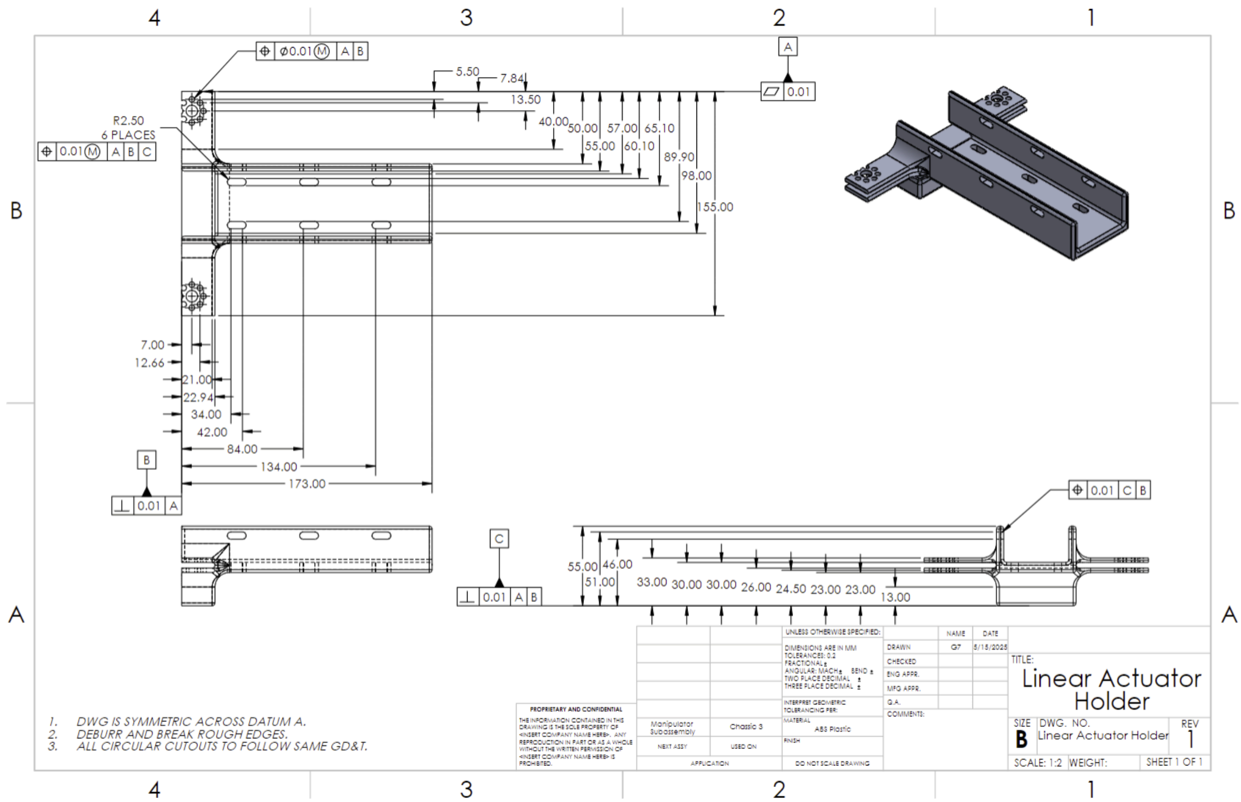


Figure 19: Linear Actuator Holder

Form: The Linear Actuator Holder has overarching dimensions, when oriented on the robot, a height of 173 mm, a width of 155 mm and a depth of 55m.

Fit: The design is made to align with the C-Beam. More specifically the suns and thickness of the beam, wrapping around it. It also has oblong circles to fit the screws needed to connect to the actuator.

Function: The Linear Actuator Holder serves as a support for the linear actuator to improve stability and help alleviate tension on the wooden plate. It snugly wraps around the thickness of the C-Beam and separates the nuts surrounding the inside of the linear actuator to evenly distribute the weight.

Gripper Subassembly

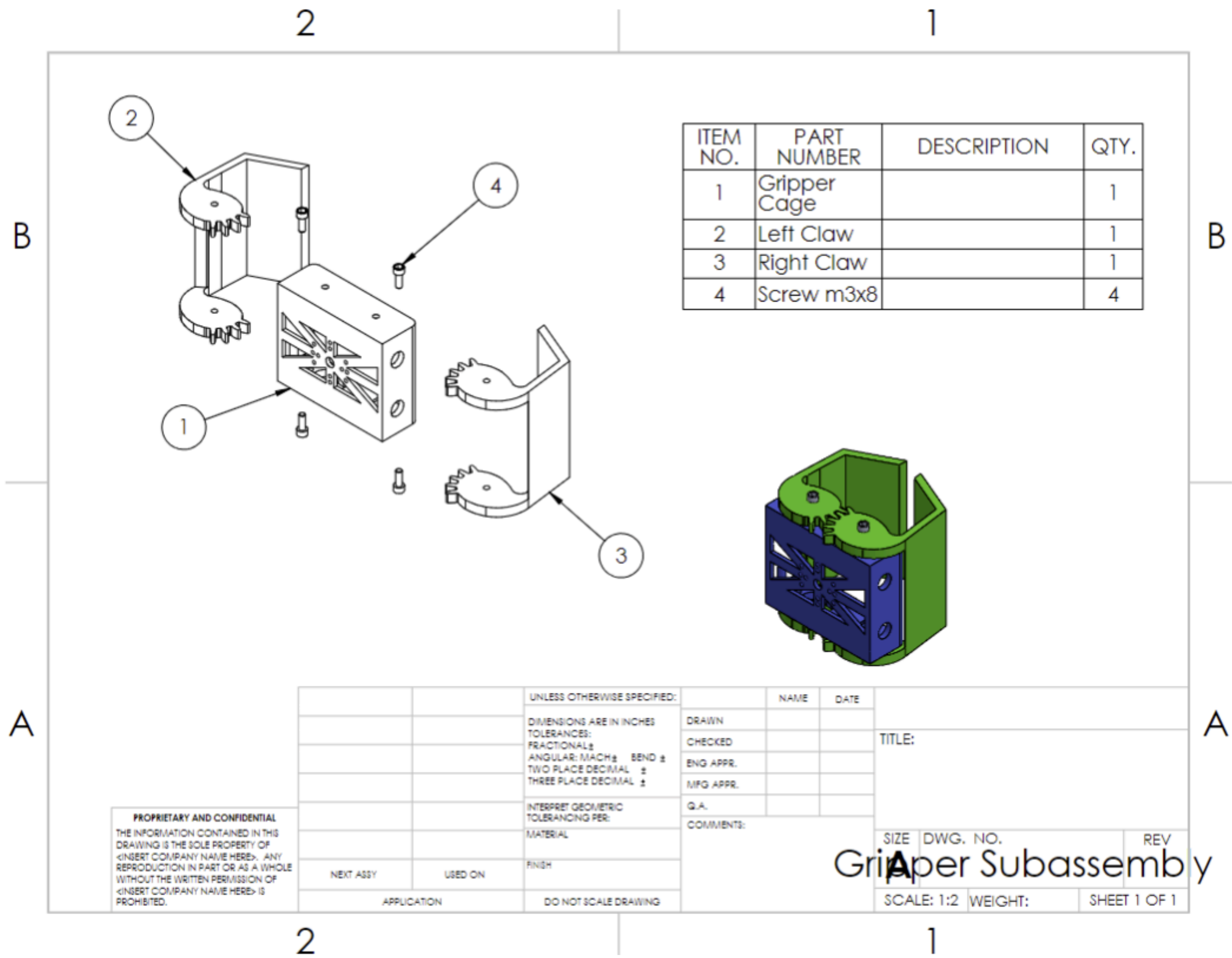


Figure 20: Gripper Subassembly

The Gripper Subassembly provides a more in depth description on how the claw is assembled, along with the parts needed to do so.

Gripper Motor Box

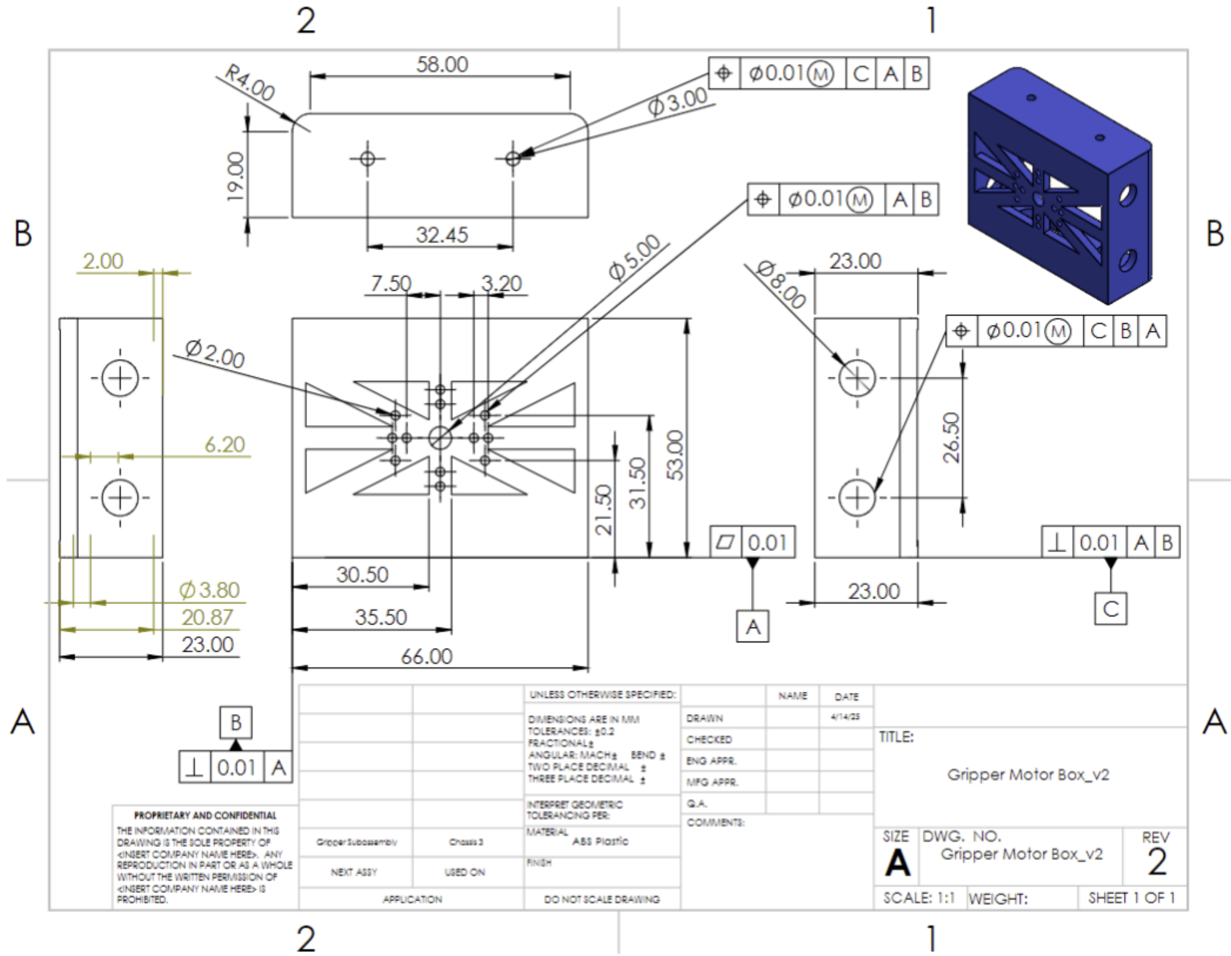


Figure 21: Gripper Motor Box

Form: The Gripper Motor Box has overall dimensions of 66 x 53 by 23 mm.

Fit: It has holes so that the screw m3x8 can fit into the box and connect to the claw.

Function: The motor box's primary function is to serve as a connecting piece between the motors and the claws.

Actuator to Motor Box

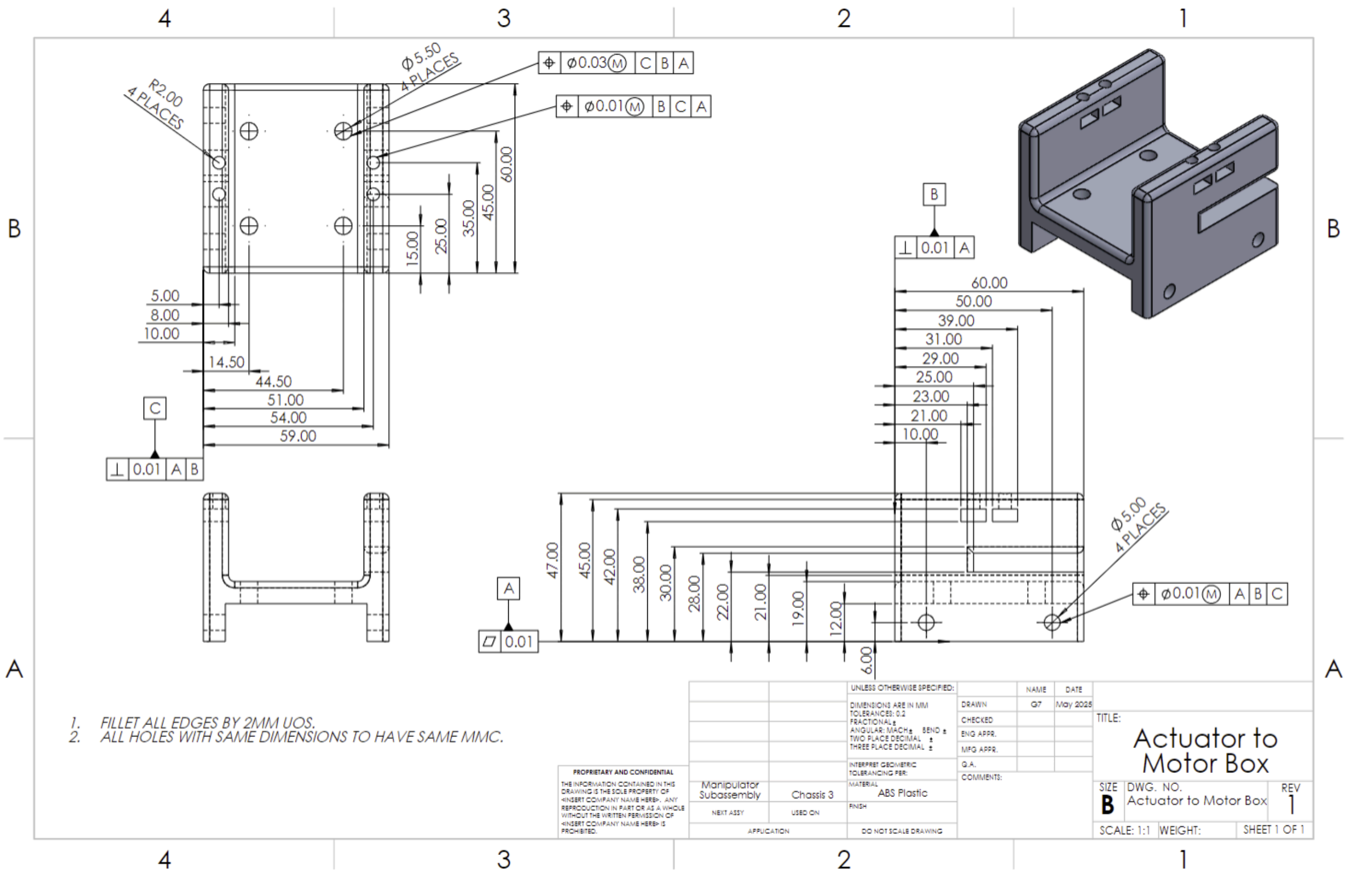


Figure 22: Actuator to Motor Box

Form: The motor cage has overall dimensions of 47 mm height by 59 mm width by 60 mm depth. It is to be 3D printed using ABS plastic.

Fit: The item can first be found on the manipulator subassembly where it can be seen between a servo motor and the linear actuator. It also holds the second ultrasonic sensor used for the claw.

Function: The motor cage houses the first servo motor, while also connecting the gripper subassembly to the linear actuator, thus completing the manipulator subassembly. It is designed to prevent unnecessary motion from the claw, so that it remains steady and firmly in place. It also firmly keeps the ultrasonic sensor in place, above the claw.

Claw

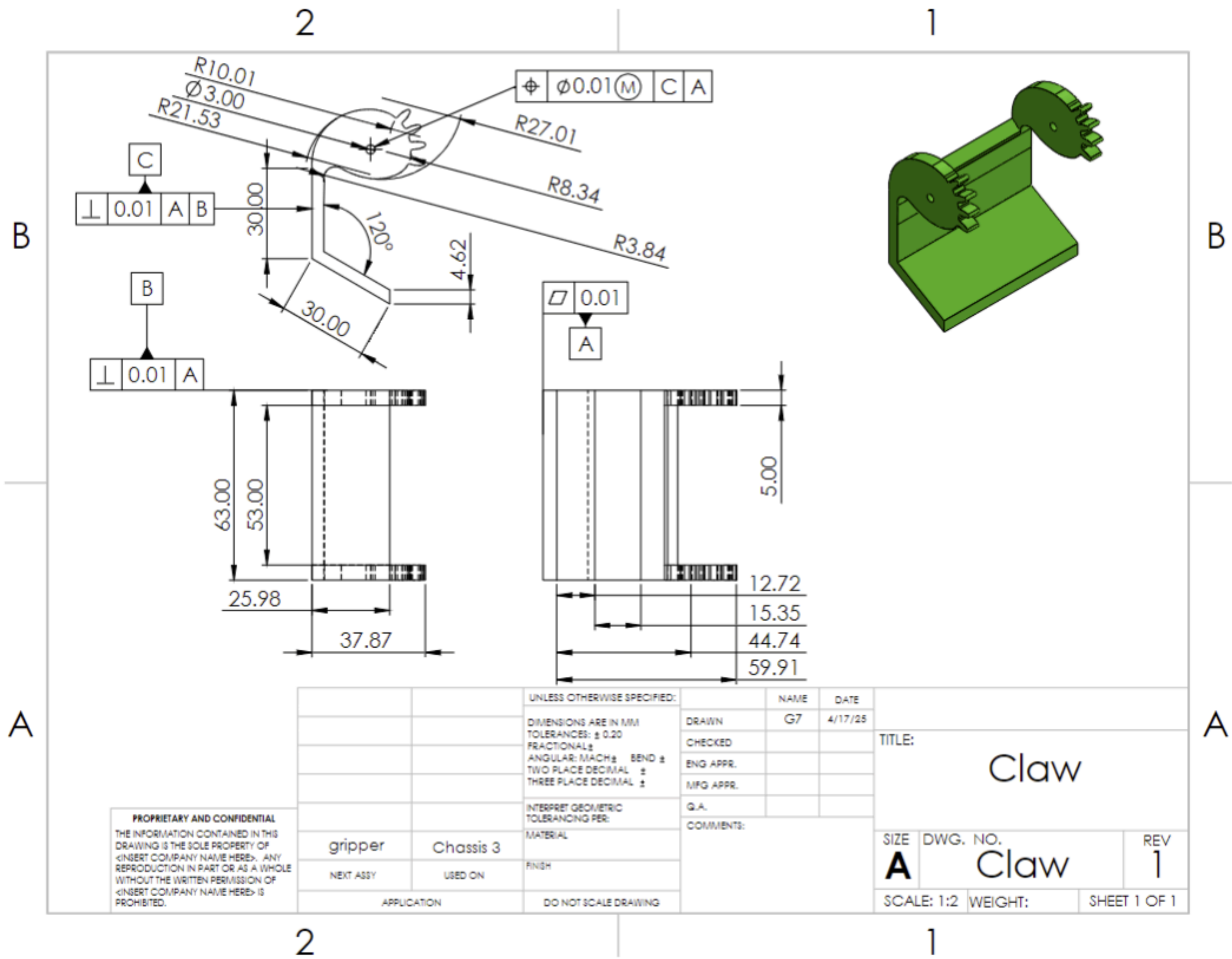


Figure 23: Claw

The Claw has overall dimensions of approximately 63 mm in height, 38 mm in width, and 60 mm in depth and key radii of 27 and 22 mm. It also consists of gear teeth for mechanical rotation with the opposite claw. It serves to firmly grasp and deliver the cups to their desired location.

Electronics Plate

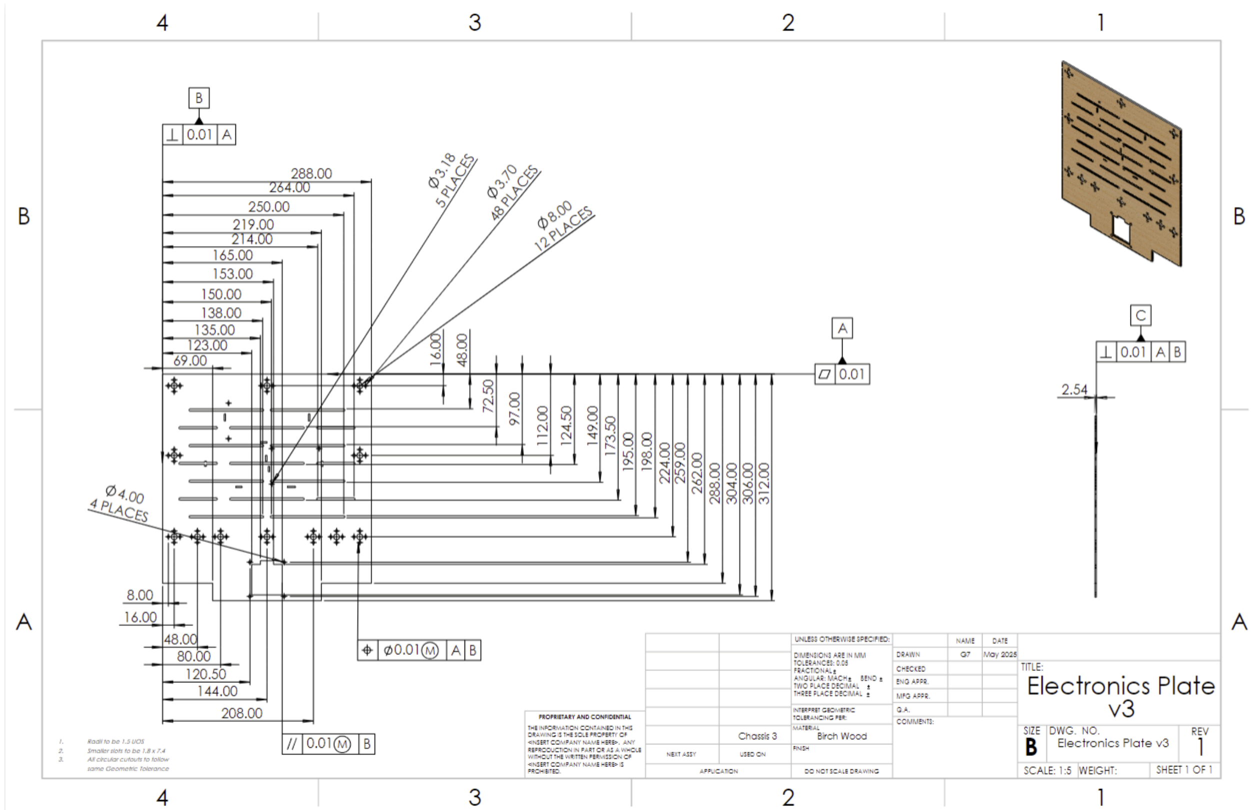


Figure 24: Electronics Plate

Form: The electronics plate has overall dimensions of 312 mm by 288 mm. To manufacture, it must be laser cut from birch wood preferably being 3/16 inches long, but can be as this as the dimension in the drawing.

Fit: The rectangular cutout directly integrates with the linear actuator, with the four surrounding circular cuts allowing for the linear actuator to be screwed in from below. The oblong cuts within the plate allows for zip ties to pass through seamlessly and the 8 mm diameter holes or for connections with the C-Beams.

Function: The third iteration of the electronics plate is now capable of securely holding the linear actuator and C-Beams. Additionally, it now has enough space to house all the electronics, along with allowing zip ties to be able to securely wrap around the electronics. With the new design, the electronics should not slip off the plate and remain stable throughout the robot's driving/manipulation tasks.

Cup (Environment)

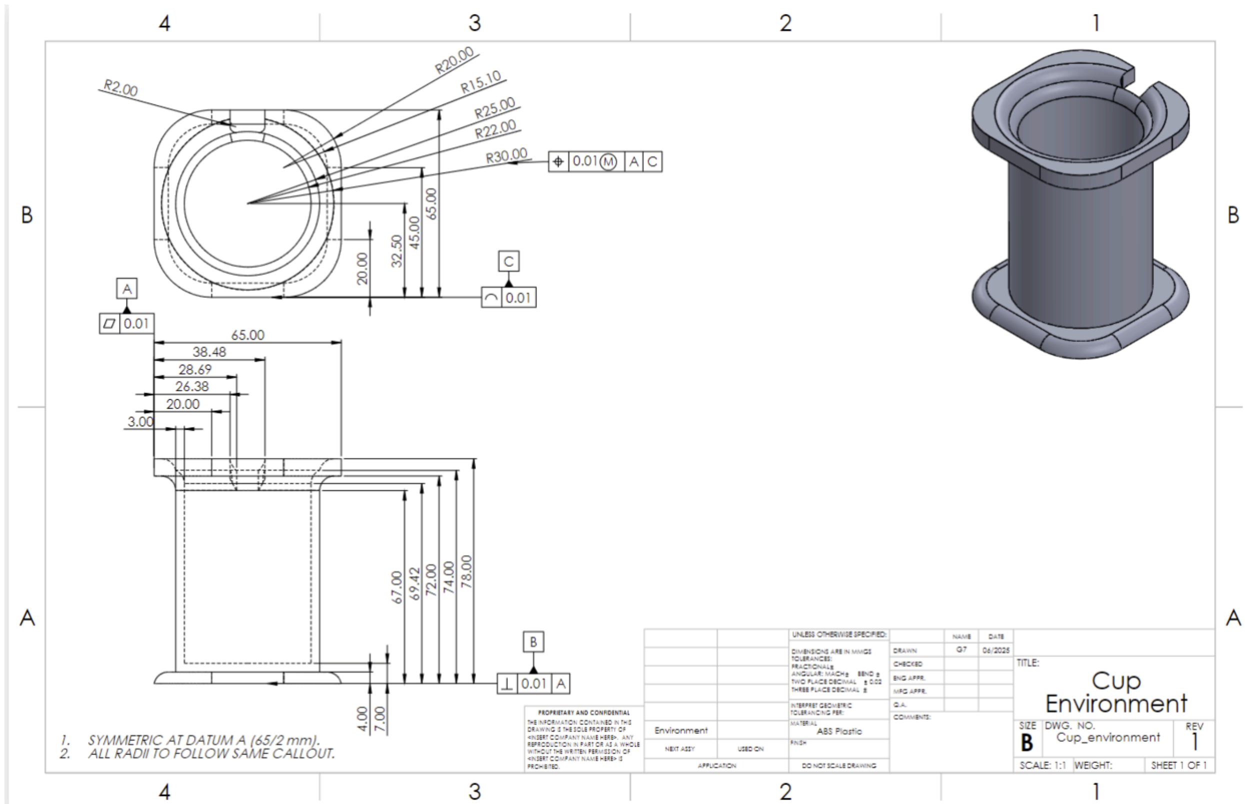


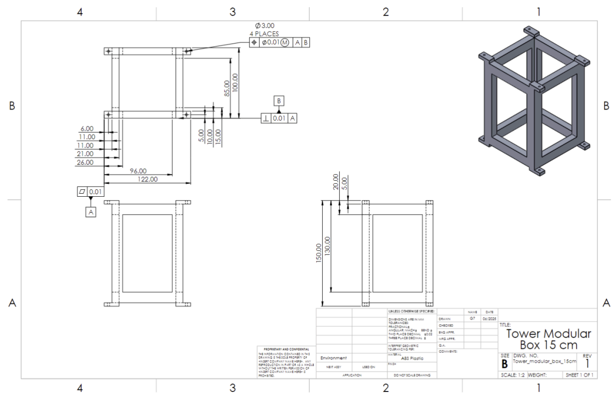
Figure 25: Cup Environment

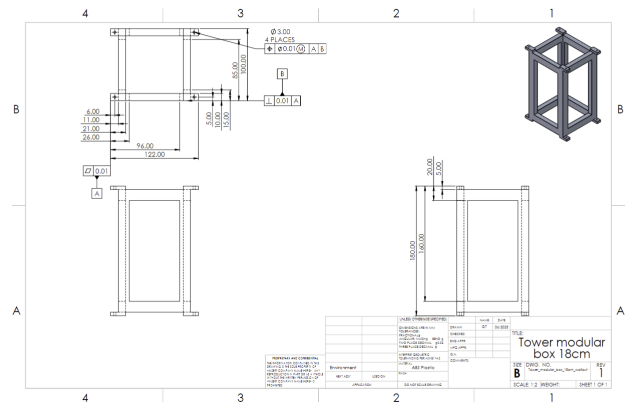
Form: The cup has a height of 78 mm and width & depth of 65 mm.

Fit: The design is to be able to securely fit within the claw's hand, with the protrusions on the top and bottom to prevent slipping through the claw's hands. Additionally, it has a cut out on the side at the top to allow ease of flow. When filled close to the brim, it is expected to be able to hold roughly 100 mL. ($V = \pi(2.2cm)^2 * 7.1cm = 107.95cm^3 \approx 100mL$)

Function: The cup is to be able to hold the 25, 50, and 100 mL of water.

Tower Modular Boxes (15&18cm)





Figures 26&27:Tower Modular Boxes (15cm, left and 18cm, right)

Form: Both modular boxes have width and depth of 122 mm and 100 mm respectively. They have varying heights of 15 cm and 18 cm.

Fit: They can fit into other modular boxes of varying heights.

Function: The modular boxes are stacked interchangeably to determine the varying heights of the cups.

Tower2 wallcut

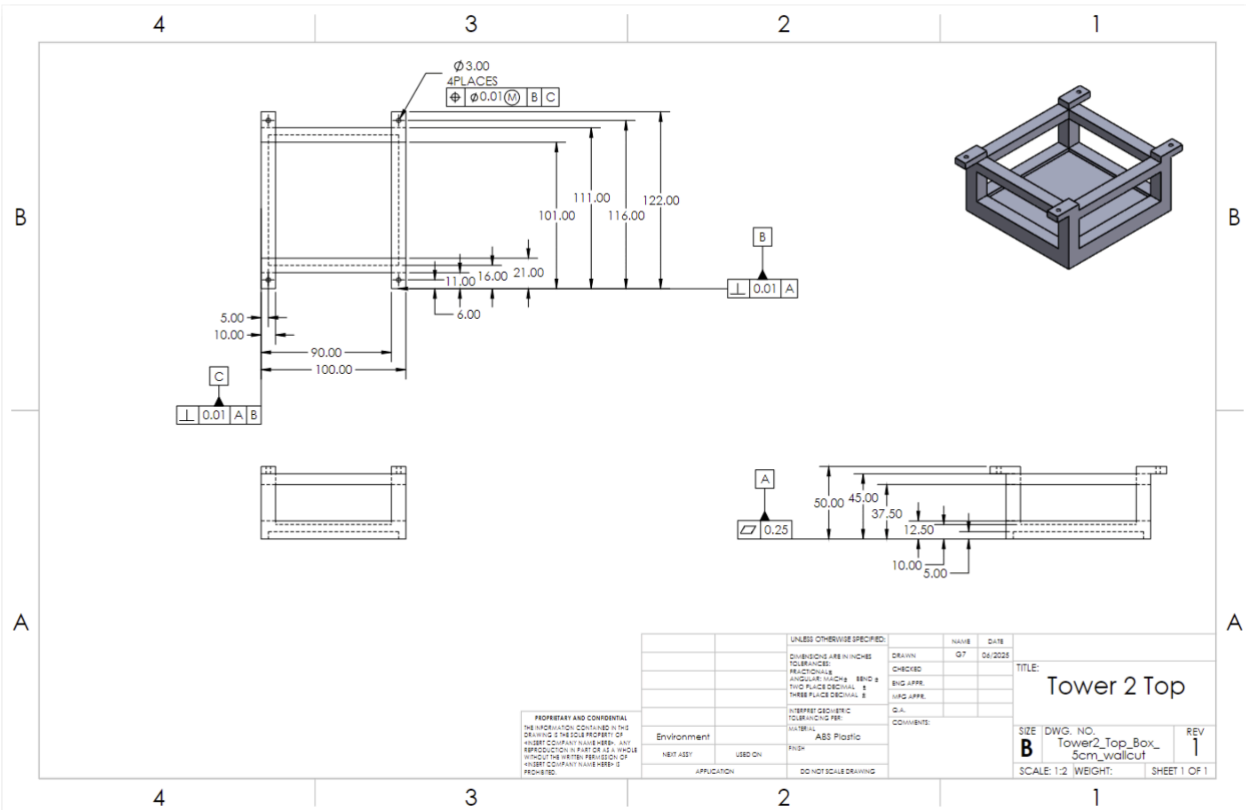


Figure 28: Tower2 wallcut

Form: The tower 2 wallcut has dimensions of 100 mm by 122 mm by 50 mm.

Fit: It is designed to fit into the tower modular boxes and fit the cup.

Function: It serves as a stand for the cup.

4.0 Mechatronics System Design and Analysis

Component	Type	Function	Location
HC-SR04 Ultrasonic Sensor	Distance sensor	Detect object proximity (cups/cups)	Front of robot & claw cage
RPLIDAR A1	LIDAR	Map environment and locate obstacles	Front C-beam (rotating mount)
MG996R Servo	Servo motor	Open/close gripper claws	Gripper cage

JX PDI-6221MG Servo	High-torque servo	Tilt cup to pour	Behind gripper
KAIBRITE Linear Actuator	Stepper + screw rail	Raise/lower claw assembly	Rear of manipulator arm
TB6600 Driver	Stepper motor driver	Drives linear actuator	Mounted on electronics plate

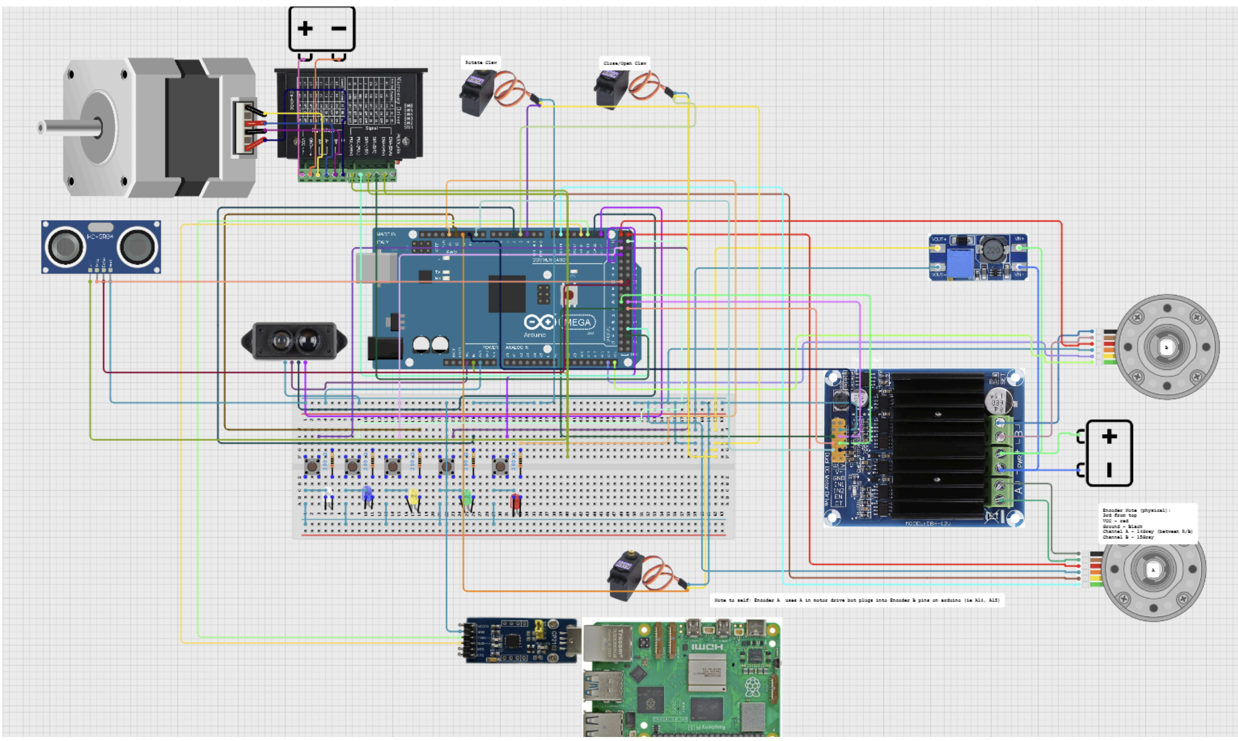


Figure 29: The circuit diagram including all the connections used in the rover is seen above with the following link in case zooming in is needed: https://app.cirkitdesigner.com/project/18ff5d42-748c-4939-98cb-f4acb01b5751.

4.1 Track Navigation Mechatronics

LIDAR-Based Environmental Mapping

Ultrasonic Sensor for Local Obstacle Avoidance

For local obstacle detection, the LIDAR ultrasonic sensor was mounted on the front of the chassis of the robot. A picture placement is shown in figure 30. It operates by sending out an infrared beam. This beam is reflected on any nearby surfaces and records the time to get back, then using this information to calculate distance from contact. This sensor is connected to the Arduino Mega via communication pins 20 and 21, requires 5V, but uses 3v3 logic.

Figure 30: LIDAR

Chassis Electronics Integration and Control Logic

For chassis-level electronics integration, all sensors and actuators are connected to an Arduino Mega mounted on the central electronics plate. A picture of the integrated electronics layout is shown in figure 29. Power is supplied by a 14.8V Li-ion battery, regulated down to 5V using a buck converter for sensors and servo motors. The Arduino handles signal routing, issuing PWM outputs to servos and step signals to the stepper motor driver. Sensor inputs, including LIDAR serial data and ultrasonic echo signals, are processed in real-time to determine navigation behavior. All wiring is managed using JST connectors and cable ties to reduce clutter and allow for easy debugging. The system follows a state-machine logic architecture, enabling mode switching between scanning, navigating, and manipulating based on sensor feedback.

4.2 Manipulation Task Mechatronics

Linear Actuator and Stepper Motor Driver

For the manipulation task, a 200mm length travel linear stage actuator with NEMA17 Stepper Motor was used. A picture of the linear actuator is shown in figure 31. This is powered by a 12 volt battery supply through a motor driver TB6600. By selecting the pulse per second on the stepper motor driver, which controls the speed with which the linear stage moves on the rail, it is made sure that the drink is not spilled on vertical translation.

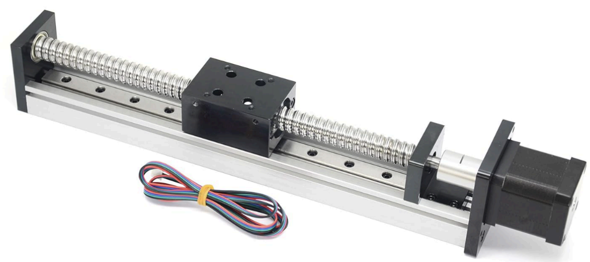


Figure 31: Linear Actuator with Stepper Motor

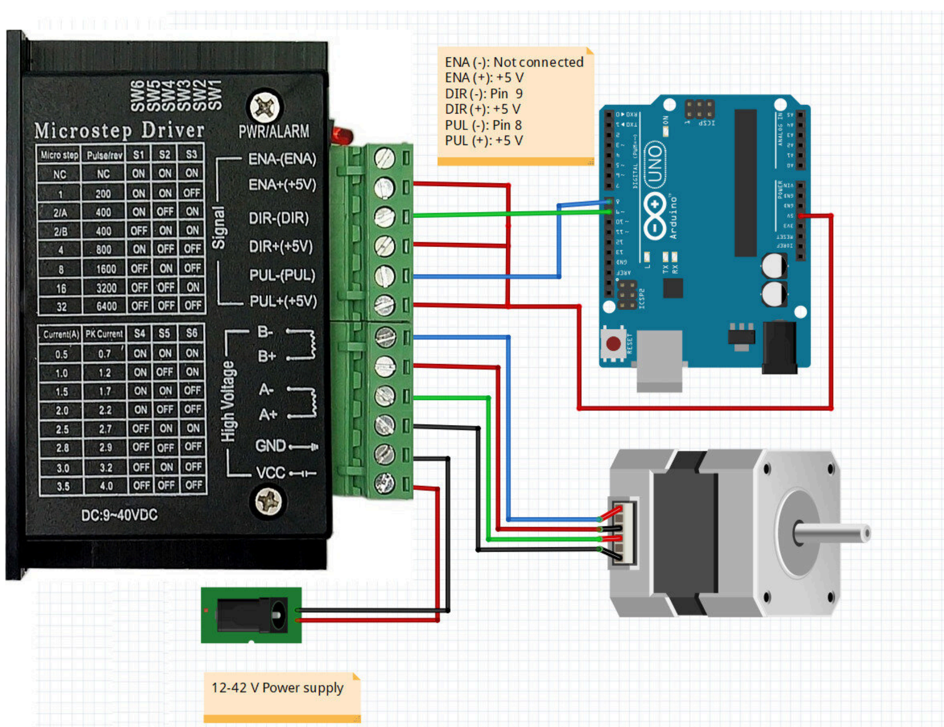


Figure 32: General Wiring for Stepper Motor Driver

Servo Motors for Claw and cup Tilting

Two mg996r servo motors are used in the gripping and tilting portion of the claw, each providing one degree of freedom. The first servo motor controls the claw’s ability to open and close, allowing the dexterity needed to securely grasp the cups. The second servo motor allows for the accurate rotation of the cup. These motors power the execution of the manipulation task, gripping, holding, and pouring, while minimizing the risk of spillage.

Figure 33: Mg996R Servo Motor

Ultrasonic Sensor for cup and glass detection

The HC-SR04 Ultrasonic Sensor is beneficial in a self-driving robot due to its ability to measure distance using soundwaves. This method makes it ideal for non-contact object detection, allowing the manipulator to prevent unnecessary spillage. Additionally, it will not have the same issues as optical sensors that may have difficulties with reflective items such as clear cups. The sensor allows for the manipulator to detect the cup and guide the claw to the cup’s location with precision, as well as confirm grip. It is then used in navigation aid across the environment to the basins, where the sensor will complete its task.

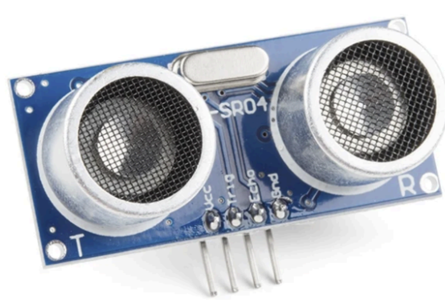


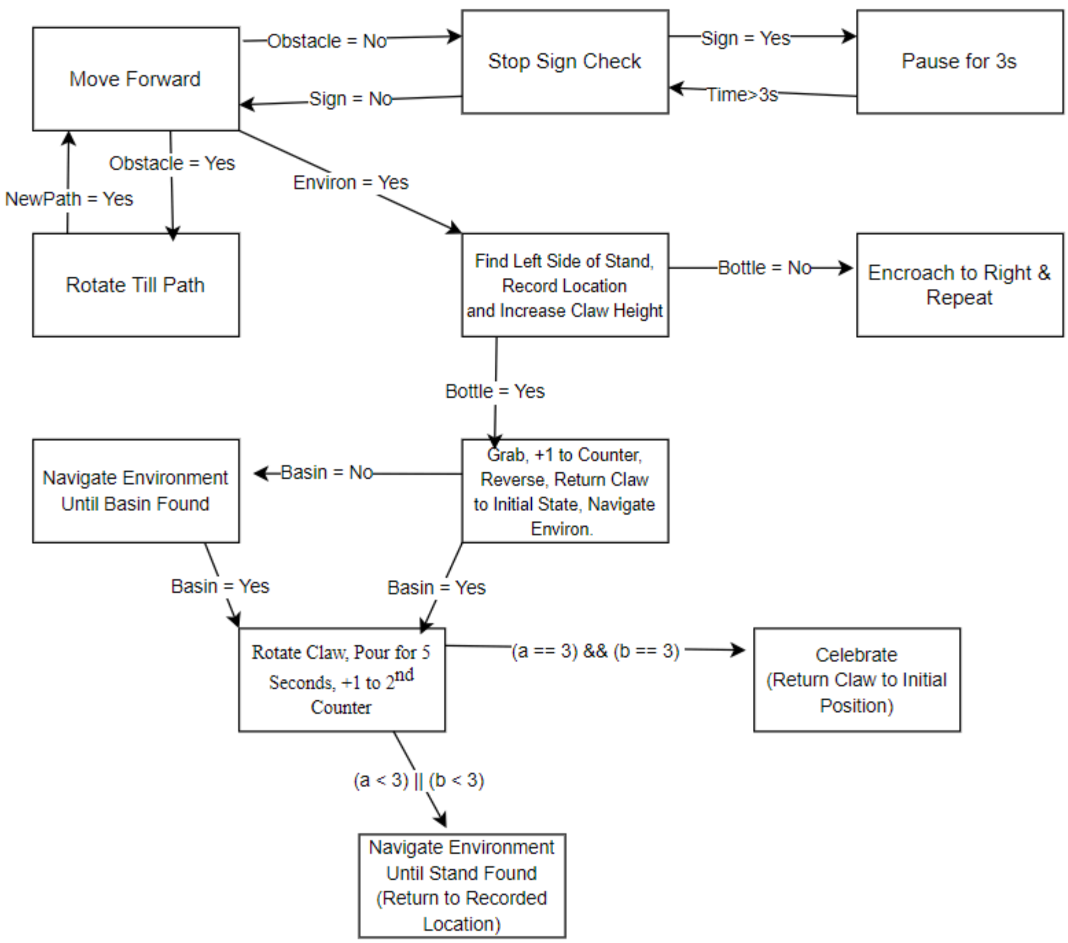
Figure 34: Physical Image of HC-SR04 Ultrasonic Sensor

5.0 Control System Design and Analysis

hhh

6.0 Software Design and Analysis

6.3.1 Stateflow Diagrams



-

Figure **XXX**: Stateflow Diagram used for coding and Manipulation of Grain.

6.3.2 Embedded Real Time Software Implementation

In order to maintain course, pure pursuit was coded into the Arduino with MATLAB.

To ensure the robot's ability to detect obstacles such as stop signs and lights, the Raspberry Pi is utilized. The coding language used is Python and it connects to the Arduino

Mega as it sits on the front left of the chassis's C-Beam. Whenever the Raspberry Pi camera detects a stop sign, it encodes a signal to send to the Arduino Mega's Simulink code, telling the robot to pause for 5 seconds. After 5 seconds, the robot will continue moving and ignore subsequent signals. If the camera instead detects a stoplight, it will first check to determine the stoplight's status. If the stoplight is red, then a signal will be sent to the Arduino Mega's Simulink code to stay stopped continuously until the light turns green. If the light is any other colour, no signal will be sent.

```
#define dirpin 51
#define steppin 49
int speed = 400;
#define stepsPerRevolution 10

#include <Servo.h>

Servo myServo1;
Servo mrServo2;
unsigned long MOVING_TIME = 3000; // moving time is 3 seconds
unsigned long moveStartTime;
int startAngle = 0; // 0°
int stopAngle  = 90; // 90°

void setup() {
 // put your setup code here, to run once:
 pinMode(A0, INPUT);
 pinMode(steppin, OUTPUT);
 pinMode(dirpin, OUTPUT);

 Serial.begin(9600);

 myServo1.attach(6);
 myServo1.attach(7);
 moveStartTime = millis(); // start moving

 // TODO: other code

}
```

```
void loop() {
 digitalWrite(dirpin, HIGH); //HIGH is counter clockwise Low is Clockwise
 int height =1000;
 for (int i=1;i<=height;i++)
 {
   step();
 }
 unsigned long progress = millis() - moveStartTime;

 if (progress <= MOVING_TIME) {
   long angle = map(progress, 0, MOVING_TIME, startAngle, stopAngle);
   myServo1.write(angle);
 }
 digitalWrite(dirpin, LOW); //HIGH is counter clockwise Low is Clockwise
 height =1000;
 for (int i=1;i<=height;i++)
 {
   step();
 }

 unsigned long progress = millis() - moveStartTime;

 if (progress <= MOVING_TIME) {
   long angle = map(progress, 0, MOVING_TIME, startAngle, stopAngle);
   myServo2.write(angle);
 }
}
void step() {

 for (int i = 0; i < .5 * stepsPerRevolution; i++) {
   digitalWrite(steppin, HIGH);
   delayMicroseconds(speed);
   digitalWrite(steppin, LOW);
   delayMicroseconds(speed);

 }
}
```

7.0 Implementation Results and Discussion

7.1 Project Planning

Throughout Team 7's time in this course, weekly meetings on Wednesday morning were put in place to develop, assign, and start new tasks. The first quarter was predominantly spent on basic assembly, coding, wiring, and CAD. More specifically, the first chassis was made, the sensors were coded to detect objects along with the wheels being able to move, the provided circuitry was connected, and the ultrasonic/LIDAR sensor mounts were constructed. Quarter 2 is where the bulk of the project takes place.

Around week 6, the team decided to add the manipulator's wiring connections, but a cosmetic issue arose: the wiring looked too cluttered. With little rebuke from the members initially, we went ahead and swapped out the small provided breadboard with one that was twice the size. The new breadboard had a little division in the middle making it the perfect choice to connect all the previous connections to the left and all the new manipulators to the right. Someone along the rewiring, the decision to connect the 12 DC Motor Drive was attached to the 5V of the Arduino, halting all progress for the day. The arduino proved to be cantankerous up until Tuesday of week 7, where Will pointed out that fatal mistake, which short circuited the arduino, rendering it useless. Will also found another issue concurrently; an item plugged into the "AREF" of the arduino. Luckily, Will graciously provided us with a second arduino, after subtracting the appropriate amount of money from our budget. These two flaws combined can be attributed to the arduino overheating and the spent time of lab 6.

The manipulator was not the only wiring moved to the right side. We also moved the ground of encoder B, back right of the rover, to this side. Throughout week 7, the gain of the tires were profusely changed to get encoder A to function. The thought process was that if encoder A is moving at such a slow speed, then the encoder not following the itinerary, is encoder A. The following step included testing the reaction to the stop sign, demonstrating encoder A to jitter and encoder B to continue rolling. The conclusion derived was that the jittering was a sign of encoder A being faulty. At the end of the night, Professor Tsao came up, for a second or two, and mentioned the encoder could be fried. Putting our minds at ease knowing the arduino was in top shape, we redirected our attention to encoder A, thus wrapping up the time spent in lab 7.

In the week following, the wiring was repeatedly checked to the point where it was indelibly ingrained. The encoder being fried, despite not being the connecting piece like the arduino, but a constituent, proved to be a tantamount issue. At the first available chance, we requested Toby to take a look, who recommended reading the outputs of the encoder while manually moving them. What we discovered was that encoder B, not encoder A, was the one being uncooperative. Relaying this info back to Toby, he quickly determined that the ground of encoder A was not properly connected to the breadboard. Toby then managed to procure us a new TXM Torque Nado (encoder), and we were back on track. What we failed to realize, is that during the rewiring in week 6, the division of the breadboard was a hard cutoff, so it needed an extra wire to connect the grounds. Luckily, the manipulator was hardly tested at this point, so the encoder was the only thing to have short circuited.

Afterwards, redesigning of the Servo Mount, and tweaks to the course navigation and manipulator were made.

7.2 Team's Communications/Contributions

As mentioned previously, the group met weekly on Google Meet on Wednesdays at 10:00 am to 11:00 am to discuss plans, progress, and current situations verbally. In order to communicate outside of Google Meet and class, we used GroupMe.

Ahan contributed:

Sid contributed:

Jacob contributed:

Joseph contributed:

Allan contributed:

(See excel)

7.3 New knowledge

Overall, the ob

Weeks 6, 7, 8

Figure **XX**: The Gantt Chart Displaying The Project Timeline, Portraying into what the Tasks Were and When They Happened.

**https://1drv.ms/x/c/d7cfc09a978dde28/EfsaaFUcz5JAtSoapVgqCZEBrhxgwl2QVPzH1Ji-WHAKeA?e=jKEmcQ - ** link to Gantt Chart

8.0 Conclusion

8.1 Summary of Current Design Progress

Overall, the objective is to navigate an obstacle course and detect cups at different heights filled with a grain. Using a combination of sensors the cup and its height are detected. Next, we use a robotic arm to grab the cup at the height detected. After moving to a target location, we tilt the cup into a glass to deliver the drink.

8.2 Future Work

The construction of the claw/rover will be done according to the schematics of the SolidWorks drawings. All testing will be done physically to determine if the code needs to be changed. All testing will be done with cups filled to 25mL, 50mL, and 100mL and consist of three steps. Step 1 will have rice in place of water due to easy cleanability and low risk to the electronics and wooden plate. After 5 repeated successful tests, the moved object will change to sand as its fluidity is not too dissimilar from water. The sand trials must be completed 10 consecutive times before moving to water, if failed, it will move back to test step 1. After 10 repeated successful tests of sand, the desired item, water, will be put into play.

Software development will follow a similar structure to state flow diagram. Using feedback control and the knowledge gained in lecture and labs, the state flow diagram will be extrapolated to Simulink and the control parameters will be determined. Further refinement will be made based on testing results (grip strength, spill control, etc.).

9.0 Acknowledgements

Team 7 would like to thank Tsu-Chin Tsao for teaching us the theory behind the project. We would also like to thank Toby Chen and Will Shih for assisting with trouble shooting throughout the quarters.

10.0 References

S. Rai, J. Simonelli, K. Chu, H. Chang, C. Kang, C. Lim, R. Shaefer, and T.-C. Tsao, "Mechatronics Pedagogy in Mechanical Engineering Capstone Design," in Proc. American Control Conference (ACC), Seattle, WA, USA, 2017, pp. 5343-5348.

C. Baine, *Engineering Mobile Robotics Teacher Guide*. Pittsburg, KS, USA: Pitsco, Inc., 2018, pp. 1-61.

11.0 Appendix

Parts List:

Component	Description	Price	Purchase Link
JX Servo PDI-6221MG	High-torque digital servo with metal gears, suitable for RC models requiring strong and precise movements.	$21	Amazon Link
Ultrasonic SparkFun Sensor	Distance measuring sensor using ultrasonic waves, ideal for obstacle detection and measuring proximity.	$10	SparkFun Link
MG996R Servo	Standard servo motor with metal gears, commonly used for various robotics and RC applications.	$30	Amazon Link

LewanSoul Mechanical Robot Arm Claw/Gripper	Mechanical gripper kit compatible with mg966r servo motor	$17	Amazon Link
KAIBRITE 200mm GGP Ball Screw Linear with Nema 23 Stepper Motor	Linear motion guide with a ball screw mechanism and Nema 23 stepper motor, suitable for precise linear movements.	$59	Amazon Link
EASON TB6600 Stepper Motor Driver	Stepper motor driver capable of handling high current, compatible with Nema 23 stepper motors.	$11	Amazon Link
14.8V 2600mAh Rechargeable Battery	Additional batteries to power Raspi and Linear Actuator	$19.99	Amazon Link