MS9 Design Brief - Data Extraction

Members: Ella Bierly, Jacquelyn Eng, Vihaan Le, Steven Vilcheck, Phillip Chen, Thomas Kimberlin, Ariana Butterworth, Ryan Tran

---

Design Brief 1 - Constructing a Prototype

Our OTV is a two tiered, tank driven robot. We used two motors, located at the back of the OTV with three wheels on each side, connected by a belt. When trying to figure out the design of the OTV, we used the idea of a garbage truck, where it picks up the can and dumps it in the truck. We decided to use a claw which picked up the pylon and "dumped" the puck into the interface plate where there were reed switches to determine if the puck was magnetic. The two tiered system allowed for our electronic components to be hidden and not interfere with the claw mechanism.

We started construction of the OTV from the bottom up. Once cutting the PLA in the shape we needed for the bottom layer of the chassis, we added our newly 3D printed parts, such as the motor mount, wheels, and supports. The supports held the wheels properly against the chassis and connected the two layers of chassis together. Once those components were attached, either by hot glue or screws, we started to add electrical components. As seen in figure 1, We have the Arduino Mega wired to the two motors through an H-bridge and our battery. The belts were purchased off of amazon but cut and glued by our team to match the length of our OTV perfectly.

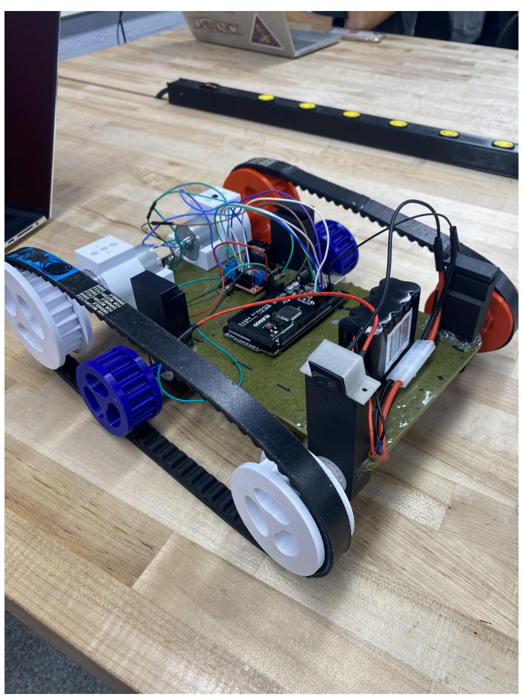


Figure 1. First layer of the OTV with initial electrical components

While the locomotion and electrical sub teams were working on the bottom part of the Chassis, the mission subteam was working on the upper part of the Chassis. The mission subteam 3D printed the claw and utilized two servos, one for claw and one for the flipping maneuver. Once the claw was tested and found to be in working order, it was attached to the top of the upper chassis. Additionally, the interface plate which held the reed switches (used to determine magnetism) and the electrode (used to determine duty cycle) was attached to the upper level of the chassis.

Figure 2. Our 3D printed claw before attached to the chassis.

Once the upper and lower level of the chassis were completed, we needed to attach them together. This took time as we needed to wire the electrical components from the upper chassis to the lower chassis, which could be difficult at times as we were dealing with extremely short wires and a limited view of our Arduino. Once the chassis were attached using the 3D printed wheel supports, taking them apart risked messing up the wiring. Once the two layers were attached, the kill switch was moved to the upper layer and the limit switches (used for navigation) were attached.

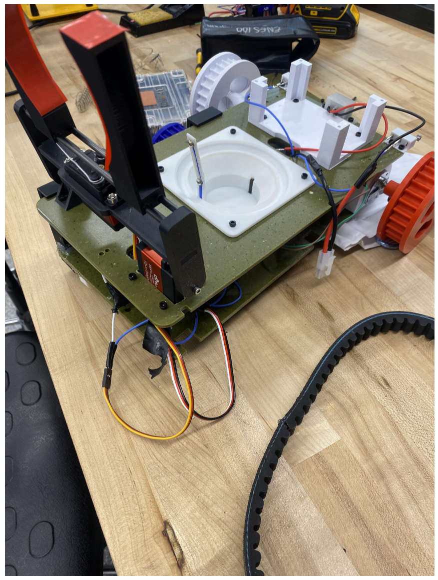


Figure 3. Chassis being attached together.

After some trial and error, we decided to extend the limit switches further away from the body of the OTV, this allowed us to add a larger ArUco marker for better navigation. This also gave us room in case an obstacle got too close to the OTV and hit another component. We also moved the battery from the designed cage to the front of the bottom chassis around underneath the claw. This allowed for a better balanced OTV. Additionally, we had trouble with our wheels staying straight and not bending due to the tension of the belt, so we added carbon fiber supports to each wheel. All of these additions helped our OTV be more successful.

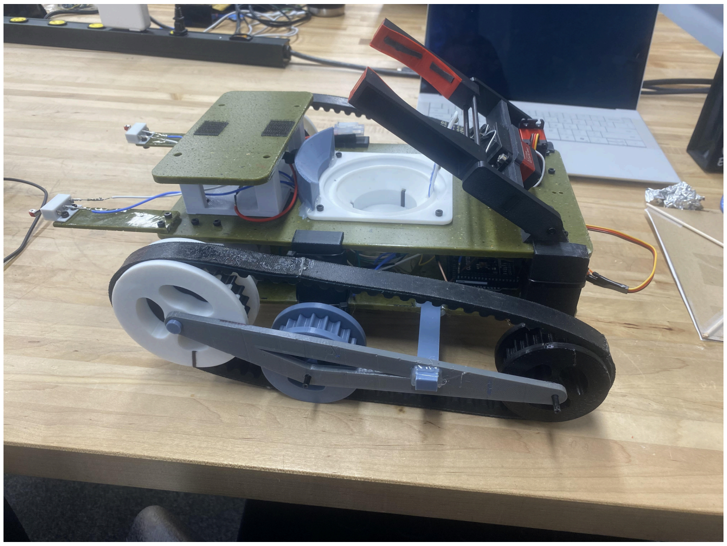


Figure 4. Completed OTV, without ArUco marker

Design Brief 2 - Mission

The hardware for mission systems is broadly divided into two subsystems, the claw and the interface plate.

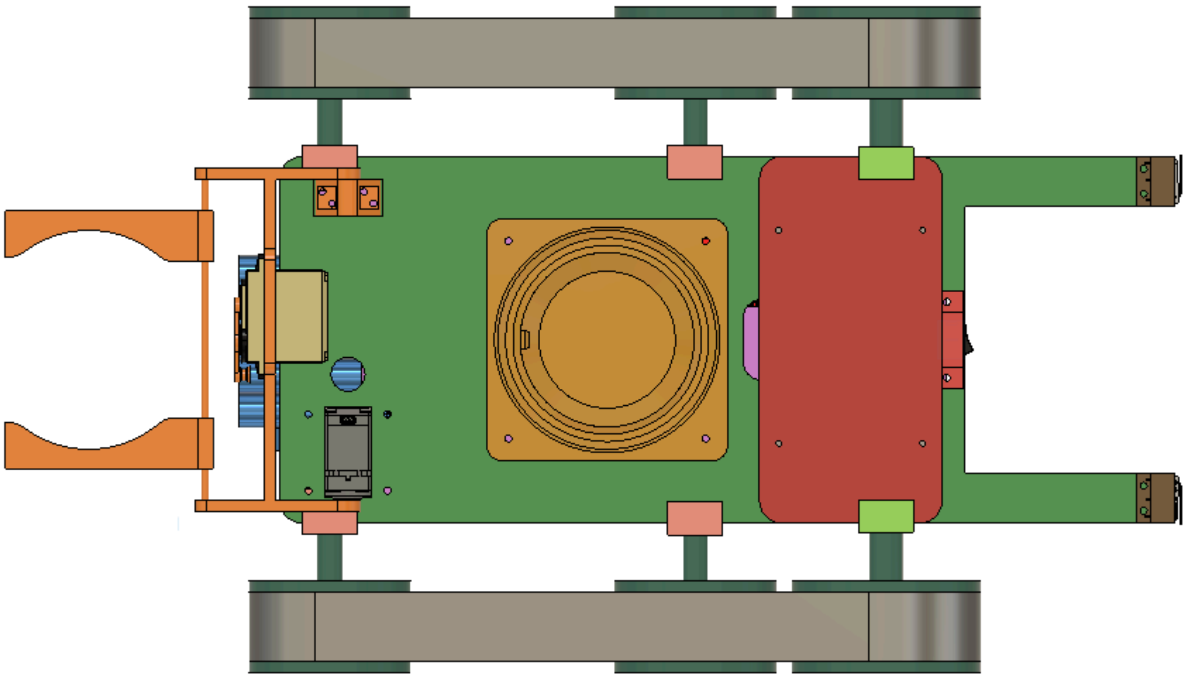


Figure 5. Top down view of OTV. The claw is pictured in safety orange and the interface plate in gold.

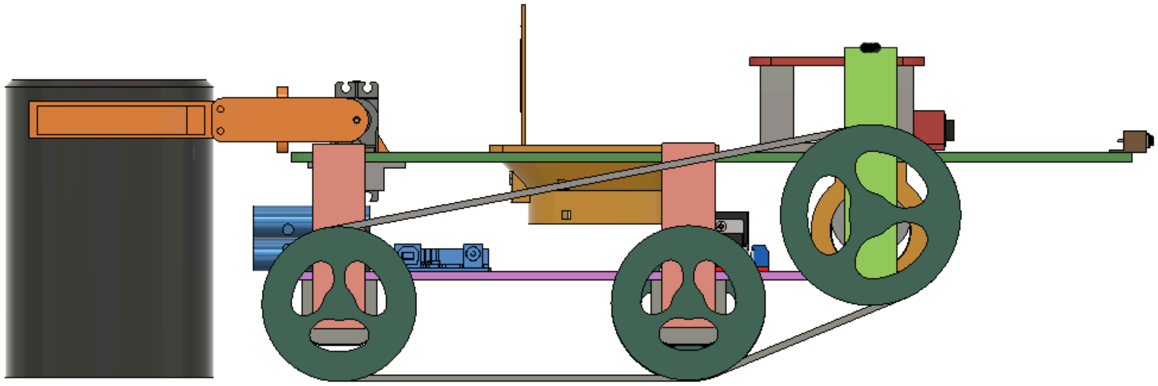


Figure 6. Side view of OTV holding the pylon.

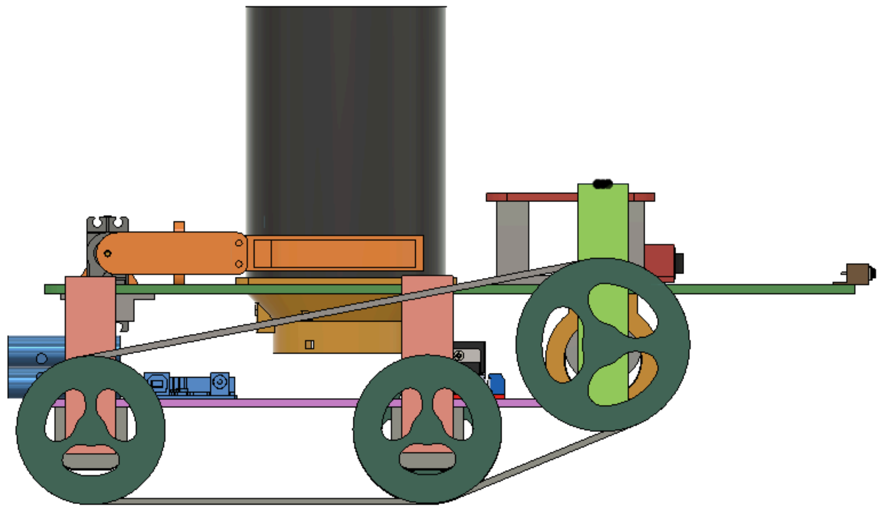


Figure 7. Mission system in action

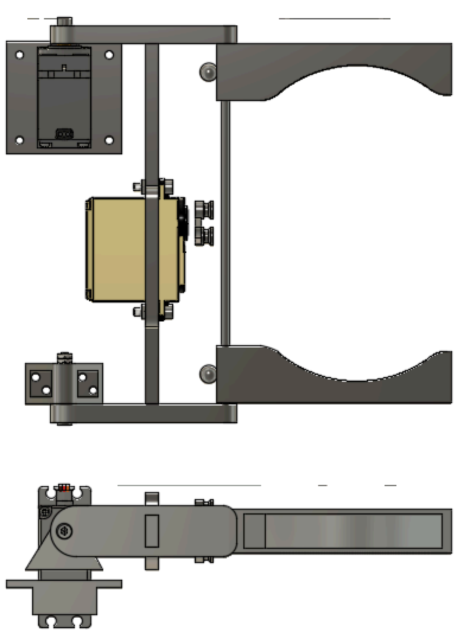
The claw arm is located at the "mission front" of the OTV. This is the side that the OTV will initially align to face towards the mission site. The purpose of the claw arm is to grab the data pylon, rotate it 180 degrees about a pivot point on the otv, and place the pylon on the interface plate with its opening facing down.

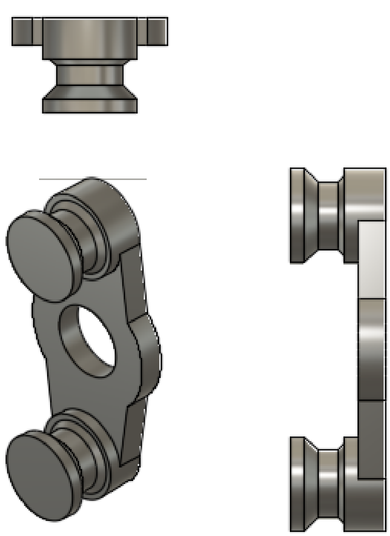
The central structure of the claw is the claw arm, an "H" shaped structure assembled from three parts that are 3D printed flat and joined with a tolerance fit. The claw arm mounts to the chassis at one end via a servo mounted to the chassis and a hinge built from a 3D printed bracket and an m3 bolt. The servo for closing the claw sits in a cutout in the center of the claw arm, and the end of the arm opposite where it mounts to the chassis holds a pair of carbon fiber rods along which the claw jaws slide. The jaws are connected to the thorn of the jaw servo with steel wires.

Ground and power for both servos are supplied from the "common ground line" and "common 5v line" that run throughout the entire OTV. The control channel for each servo is wired directly to a PWM pin on the arduino.

3D printing was chosen to allow for rapid prototyping and to get precise geometry for the inner surface of the claw, which is meant to exactly match the curvature of the outer surface of the data pylon. Prefabricated carbon fiber rods were used for their increased strength and lower friction compared to PLA, making them ideal for the jaws to slide along. Steel wire was chosen to connect the servo thorn to the jaws because it is very strong in tension compared to PLA and the length of the wire can be easily and cheaply adjusted to change the characteristics of the claw. The wires are wrapped around mounting points on the back of the jaws, and on our custom servo thorn. The thorn provides the correct length lever arm for the claw to fully close with maximum mechanical advantage. It is 3D printed with a round, slightly undersized hole which is plastically deformed using a spare servo output shaft and a vice. This gives us a tight enough fit to prevent slipping.   The use of servos over motors was chosen because of the ability of the servos to accurately turn to a specific angle, ensuring repeatability even in cases where the opposing force from the velcro might differ (due to the pylon being placed with more or less force). We chose high-torque servos for this application.

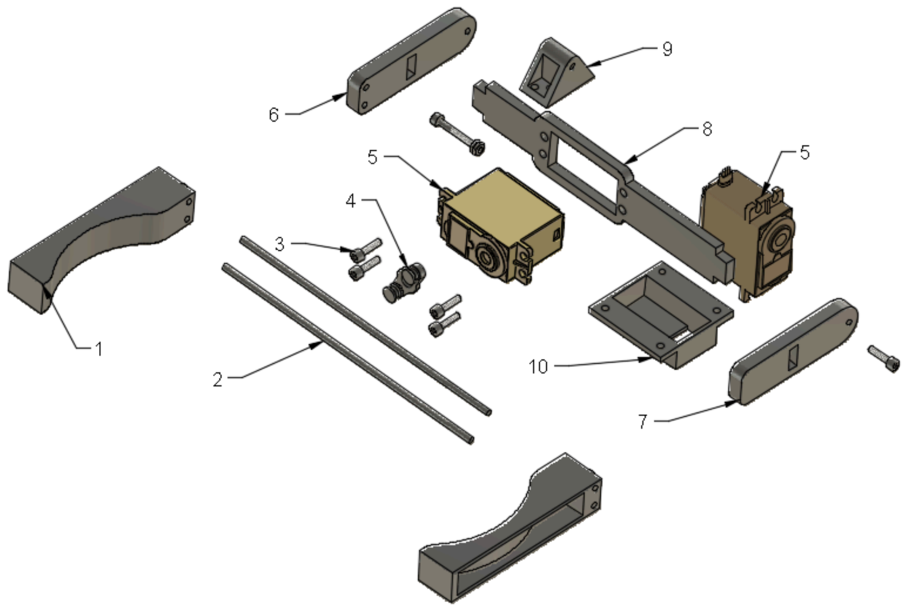
By the final run, a pair of wedges (printed in blue-gray) were added to the claw. They attach via a tolerance fit and were added to provide a greater acceptable margin of error to get the data pylon in the claw.





(Left) Figure 8. Claw Assembly CAD image

(Right) Figure 9. Detail image of custom servo thorn



	Features
1	Jaw
2	Carbon Fiber Rod
3	Bolt
4	Servo Thorn
5	Servo
6	Arm Plate 1
7	Arm Plate 2
8	Middle Arm Plate
9	Hinge Mount
10	Servo Mount

Figure 10. Claw exploded view

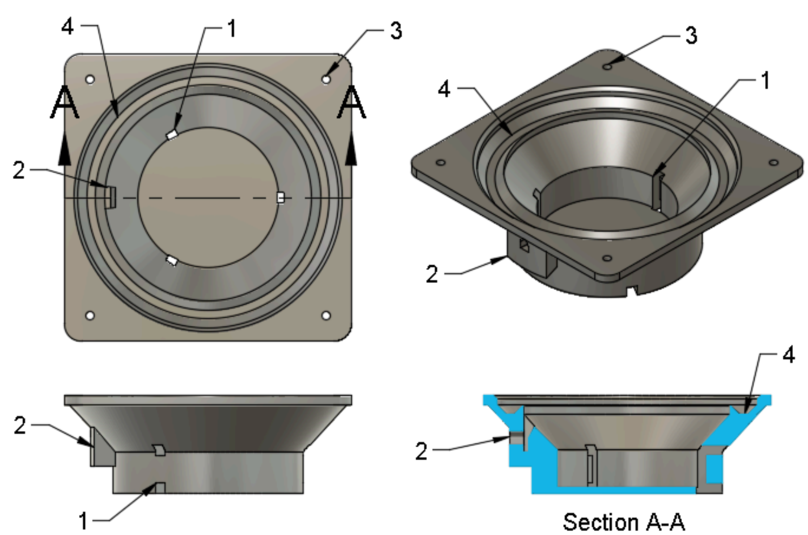
The interface plate is located in the center of the OTV on its top surface. It sits inside a circular cutout in the top deck of the OTV and is secured and aligned by four m3 bolts running through the OTV deck and the flange on the interface plate, which houses all the sensors for mission requirements.

The main body of the plate is 3D printed out of PLA as a single part. The top surface contains a ring-shaped depression for the rim of the data pylon to slot into. Concentric with this mating geometry are a conical surface and a cylindrical hole, meant to guide the puck from the data pylon into the correct orientation for the reed switches to read whether or not the puck is magnetic. The reed switches sit vertically in cutouts on the sides of the cylindrical hole. The interface plate also contains a slot for a replaceable “electrode antenna”. This antenna is a flexible piece of PLA with a pair of wires mounted in it. The antenna is seated at such an angle that the tips of the wires press against the electrodes inside the data pylon when the pylon is seated in the interface plate.

The electrode wires pass directly to a ground and digital pin on the arduino. This was done to facilitate easy replacement of the antenna (compared to connecting it to the common ground wire) as it was anticipated that the antenna might wear out or break. One lead from each reed switch is connected to the common ground wire, while the other lead is connected to an individual digital pin, allowing each sensor reading to be differentiated for testing/troubleshooting.

3D printing was selected for manufacturing to allow for the custom geometry necessary to mate with both the data pylon and the puck. The mating features were integral for ensuring proper alignment of the sensors with both the puck and the electrodes inside the data pylon. Reed switches were chosen due to their lack of passive power draw compared to other magnetic sensors. The vertical alignment of the reed switches was chosen based on testing. It was found that this configuration for the sensors produced much more reliable readings compared to the original interface plate design, which had one sensor sitting flush in the bottom of the plate.

By the final run, a "backboard" was added (printed in blue-gray) to address the issue of the puck occasionally bouncing out of the interface plate. It indexes against the edge of the flange on the interface plate and is held in place by hot glue.



	Feature
1	Reed Switch Location
2	Antenna Slot
3	Bolt Hole (Chassis)
4	Mating Geometry (Data Pylon)

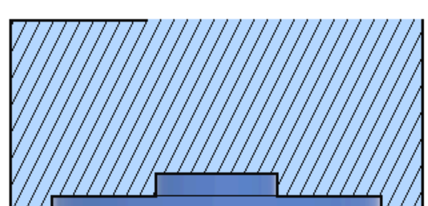
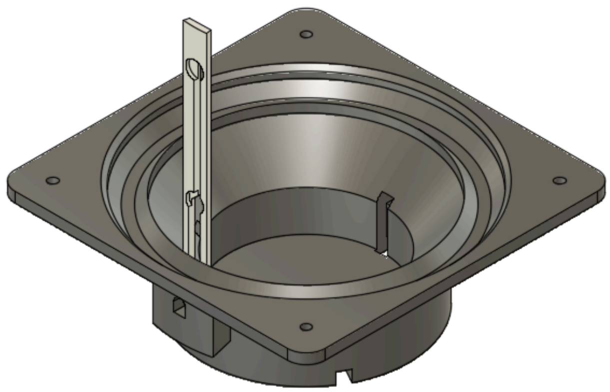


Figure 11. Interface Plate Detail



(Left) Figure 12. Interface Plate with electrode antenna

(Right)Figure 13. Section view showing mating of interface plate (purple), electrode antenna (orange), data pylon (blue), and Puck (green)

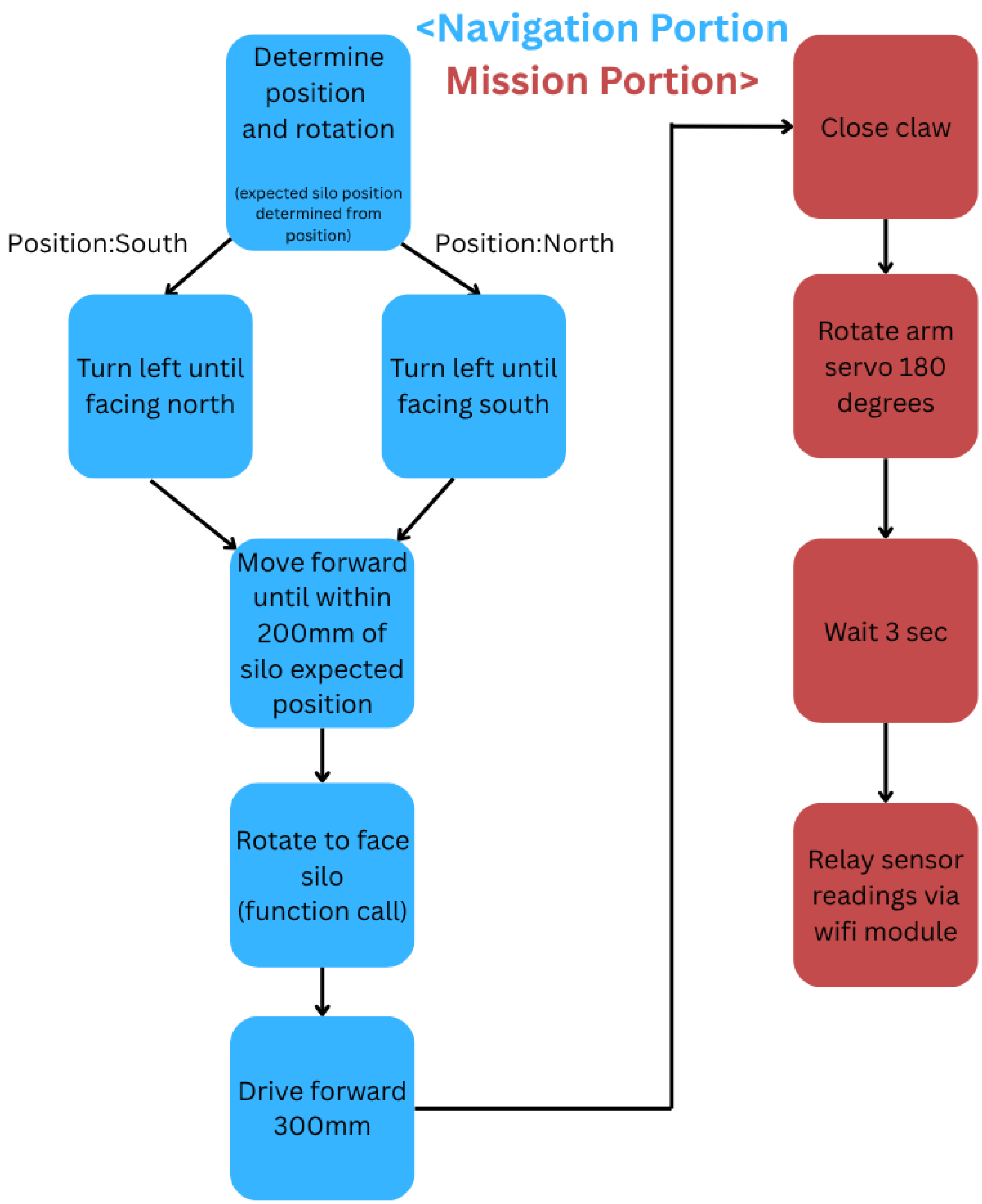


Figure 14. Mission Systems Control Flowchart

Design Brief 3 - Propulsion Modeling

The OTV has three wheels on each side. The front pair of wheels are raised and controlled by two motors. The other four wheels are non-drive with rotation being caused by the belts. The drive wheels have an outer radius of 45 mm and inner radius of 40 mm with indents of 0.75 mm every 2 mm. The supporting wheels have an outer radius of 35 and inner radius of 30 mm with the same indents of 0.75 mm every 2 mm. The belts that go over the wheels have a height of 1.38 mm and a tread depth of 0.75 mm. The teeth of the belts are spaced every 2mm. We chose the Greartisan DC 12V 30RPM Gear Motor on Amazon.

The motors are fitted inside motor mounts that are bolted onto the chassis. These motors are directly connected to the raised front wheels. The wheels fit directly over top of the motors' rods and are friction fitted together along with any additional glue needed. The back four wheels are connected to the chassis with spacers and bearings.

We chose a tank drive system where one side rotates forwards while the other side rotates backwards while turning. This enabled the OTV to rotate around its center of mass for better steering. When moving forward, both sides will rotate in the same direction. We calibrated the motor speeds to accommodate for drift while turning.

Predicted Behaviors:

The predicted torque required for our motors is 0.55Nm.

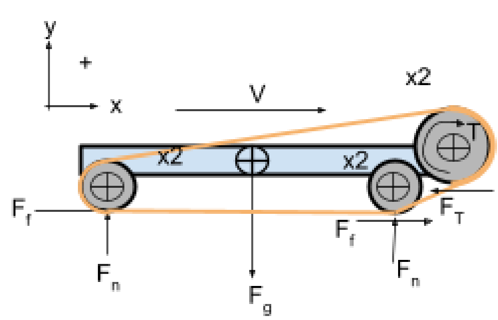


Figure 15. Free body diagram for required torque.

Equilibrium in the y-direction:

When calculating the required torque, we used 4kg for the mass of the OTV as a buffer against error.

$\Sigma F_y = ma_y = 0$

$\Sigma F_y = 4F_n - F_g$

$F_n = \frac{1}{4}F_g = \frac{mg}{4}$

$F_n = 9.81N\ with\ 4kg$

Equilibrium in the x-direction:

$F_{RR} = F_n \cdot C_{RR}$

$C_{RR} = 0.7$

$\Sigma F_x = ma_x = 0$

$\Sigma F_x = 4F_{RR} - 2F_T$

$F_T = 2F_{RR} = (2)(0.7)(9.81N)$

$F_T = 13.73N$

Torque Calculation:

$\tau = F_T \cdot r$

$\tau = (13.73N)(0.04m)$

$\tau = 0.55Nm = 5.61kg\ cm$

Our goal for the OTV's average speed was 0.04 m/s. With a minimum speed of 0.027m/s, 0.04m/s as a goal for linear speed would ensure that our OTV completed the mission within the time limit and navigated safely.

$v_{min} = \frac{8m}{300s} = 0.027m/s$

Considering that the kinetic coefficient of friction is always equal to or less than the static coefficient of friction, the operating torque is equal to or less than the required torque (0.55Nm or 5.61 kg-cm) to move the OTV on the flat arena. Though the required torque to go over the log will be higher, we accounted for this increase by finding a motor which can provide more than two times the amount of torque required to move on a flat surface. Furthermore, with a goal of 0.04m/s linear speed, the goal for angular speed was 9.55rpm. Combining these two requirements, we selected the Greartisan DC 12V 30RPM Gear Motor.

$$\omega = \frac{v}{r} = \frac{0.04m/s}{0.04m} = 1\frac{rad}{s} = (1\frac{rad}{s})(\frac{60s}{1min})(\frac{1rev}{2\pi}) = 9.55rpm$$

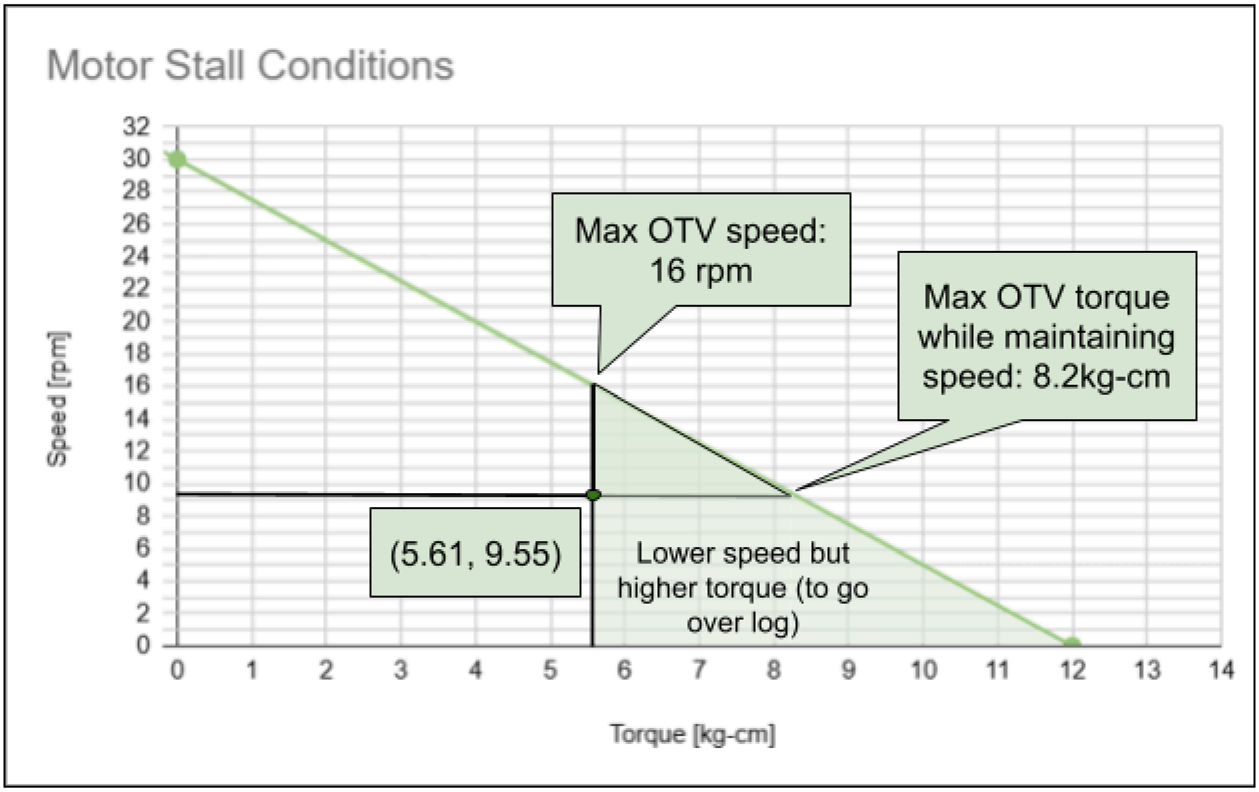


Figure 16. Motor stall conditions graph

The predicted operating linear speed was 0.067 m/s. We used the max OTV speed possible with the amount of torque required for the OTV to move to calculate the highest operating linear speed.

$$\omega = 16\,rpm = (16\frac{rev}{min})(\frac{1\,min}{60s})(\frac{2\pi}{1rev}) = 1.68\frac{rad}{s}$$

$$v = r\omega$$

$$v = (0.04m)(1.68\frac{rad}{sec}) = 0.067\frac{m}{s}$$

The predicted operating current was 420.75 mA. Considering that the ratio between the rated current over rated torque should be equal to the actual torque and current, we used the required torque to calculate the required operating current.

$$\frac{Rated\ Current}{Rated\ Torque} = \frac{x}{Required\ Torque}$$

$$x = Required\ current$$

$$Rated\ Current = 0.92A = 900mA$$

$$x = (Required\ Torque)\frac{Rated\ Current}{Rated\ Torque}$$

$$x = (5.61kg\ cm)\frac{900mA}{12kg\ cm} = 420.75mA$$

Comparison to Results:

There was a slight drift caused by a difference in rotational velocity of the motors when provided the same energy.

Trials:

	Speed (m/s)
1 meter, test 1	0.0917
1 meter, test 2	0.0915
1.57 meters	0.0942
2.36 meters, test 1	0.0923
2.36 meters, test 2	0.0883
Average	0.0916

Figure 17. Motor Test

Considering this, the OTV's actual operating linear speed is 0.0916 m/s. Accounting for the extra radius from the tread, the actual rotational speed is 0.0213 rad/s.

$$\omega = \frac{v}{r} = \frac{0.0916}{0.045} = 2.036\ rad/s = 19.438\ rpm$$

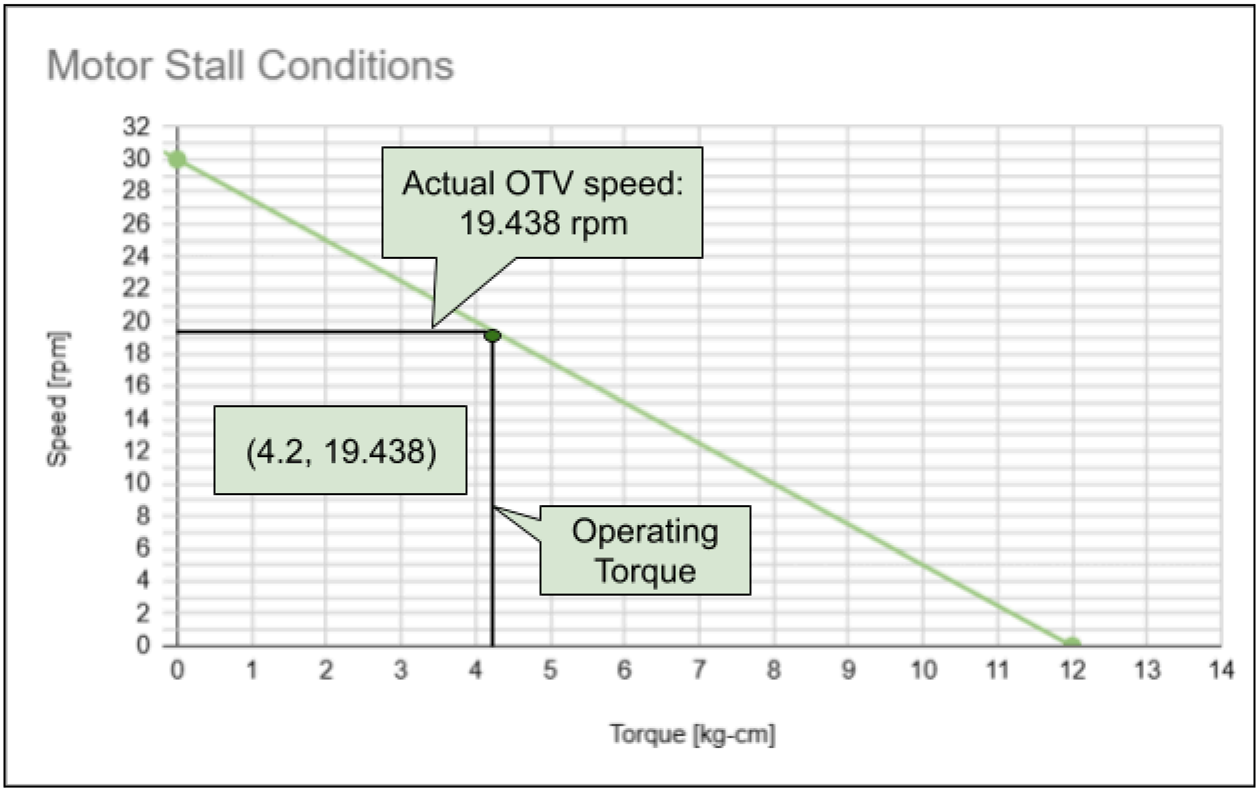


Figure 18. Motor Stall Conditions graph.

The operating torque is approximately 4.2kg-cm.

The operating torque for the OTV when moving is lower than the predicted 5.61 kg-cm. The reason behind the higher predicted torque is because we made the predicted weight to be higher than the maximum to provide a buffer against any testing errors.

Design Brief 4 - Electronics

We used a Tenergy 12V NiMH (Nickel Metal Hydride) battery. The battery has a 2000mAh capacity, a maximum continuous discharge of 2A, and a C-rate of 1C. It weighs 0.255kg. An NiMH battery was chosen because lithium and lead-acid batteries were prohibited. We selected a battery with the same voltage as the motors and H-bridge to ensure sufficient power. NiMH batteries have a relatively stable discharge curve, maintaining a voltage close to 12V for most of the cycle before dropping sharply after approximately 90% of the capacity was discharged.

We used an Elegoo Mega R3 as the OTV's Arduino-based microcontroller. The Elegoo Mega operates at 5V, and it received power from a regulated step-down from the 12V battery through its onboard voltage regulator. This allowed the actuators and sensors to safely operate at 5V without overvoltage risks. We used the L298N motor driver as our H-bridge. This motor driver was rated at the same voltage as the motors, enabling efficient control of motor direction and speed.

We used one MG996R servo and one FS5115M servo, both capable of operating at 5V. Their stall currents were 1.2A and 3A, respectively. The kill switch was required to cut power to critical systems, mainly the motors and H-bridges that operated at 12V. We placed the kill switch at the main 12V input to allow for a complete shutdown of propulsion in case of emergency. Additionally, the kill switch was located on the top layer of the chassis for easy access.

For current draw calculations, we assumed the worst-case scenario for high-current-draw components, including the drive motors and servos. Since we used the L298N motor driver, the maximum current the motor drivers could handle was 2A before potential breakdown. As previously stated, the servos had stall currents of 1.2A and 3A, which we used in our calculations, knowing we could change the servos' voltage source if needed. The current draw from sensors was considered negligible, as most were switches and we were already assuming perpetual stall conditions. The current draw for the remaining components was estimated using their rated or typical operational current values.

Components	Voltage [V]	Current [A]	Power [W]
Drive Motors (x2)	12	4	48

Components	Voltage [V]	Current [A]	Power [W]
L298 Motor Driver	12	0.1	1.2
Elegoo Mega	12	0.5	6
Arm Drive Servo Motor	5	3	15
Claw Servo Motor	5	1.2	6
WiFi Module	5	0.5	2.5
Kill Switch	12	0	0
Reed Switches (x3)	5	0	0
Haptic Switches (x2)	5	0	0
Signal Reading Antenna	5	0	0
Total	12	9.3	78.7

Figure 19. Chart of components

$$Energy = Power * Time = 78.7W * 0.167h = 13.15Wh$$

$$Required \quad Capacity = \frac{Energy}{Battery \quad Voltage} = \frac{13.15Wh}{12V} = 1095mAh$$

$$Expected \quad Run - Time = \frac{Battery \quad Capacity}{Total \quad Current \quad Draw} = \frac{2Ah}{9.3A} \times \frac{60min}{1h} = 12.9 \quad minutes$$

As seen in our calculations, our OTV has a total current draw of 9.3A, and can run for 12.9 minutes in the worst case scenario. This means that if everything went wrong, our OTV will still be able to run for 10 minutes, meeting its duration requirement. Thankfully, we never faced the worse case scenario and our battery lasted multiple runs without needing to be recharged.

Our strategy for handling power was to use pulse width modulation. By using pulse width modulation, we could manipulate and control the electronic components by cycling voltage width to get desired current output on components and reduce risk of dangerous current draws from certain components. We utilized the L298N H-Bridge motor driver to simplify our electrical wiring, improve power amplification, and ease the coding of motors. Furthermore, all other electrical components ran through the Elegoo Mega to regulate power to those components.

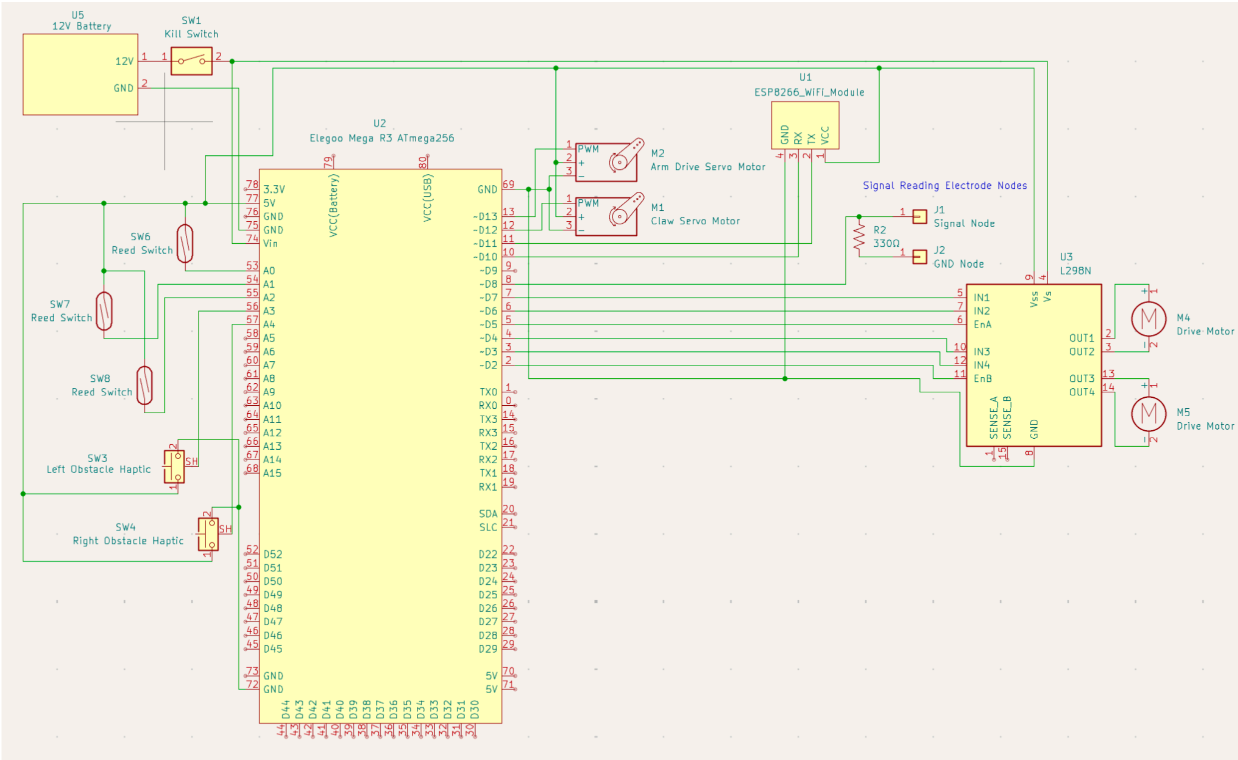


Figure 20. Circuit schematic diagram

Component	Type	Arduino Pin(s)
L298 Motor Driver (Drive)	Motor Controller	~D2, ~D5, D3, D4, D6, D6
Claw Servo Motor	Actuator	~D12
Arm Drive Servo Motor	Actuator	~D13
ESP8266 WiFi Module	Communication	D10, D11
Reed Switches	Sensor	A0, A1, A2
Signal Reading Antenna	Sensor	D8
Left Obstacle Haptic Switch	Sensor	A3
Right Obstacle Haptic Switch	Sensor	A4
Power Pins	Power	Vin, 5V, GND

Figure 21. Arduino pin assignment chart

As seen in both the circuit diagram and the Arduino pin chart, the Elegoo Mega had a sufficient number of pins to support all sensors and actuators. With all electrical components wired to the Arduino, the number of pins in use justified our choice of the Elegoo Mega R3 microcontroller. Another reason we chose the Elegoo Mega was because it was recommended for use with 12V, while the Elegoo Uno was recommended for under 9V. Since we were using a 12V battery, it made sense to select the Elegoo Mega to ensure proper functionality of the microcontroller.

In the circuit diagram, all sensors and servos were connected to the 5V power supply from the Elegoo Mega, while the L298N motor driver and the Arduino were connected directly to the battery. The kill switch was also connected directly to the battery's positive output terminal to ensure a complete power disconnection when activated.

The Arduino pin chart outlines which pins were assigned to each component, with specific functions depending on the component. Digital pins marked with a "~" in the chart were designated for pulse width modulation (PWM) output, while other digital pins were used for general I/O functionality—even if they were also capable of PWM. While specific pin assignments were subject to change during OTV production, the intended functionalities of each pin were required to remain consistent.

The Wi-Fi module enabled communication between the OTV and the computer screen. It transmitted key data such as puck magnetism, OTV orientation, and other useful information by linking the computer to the ArUco marker on the OTV.

Design Brief 5 - Compiled Set of Engineering Drawings

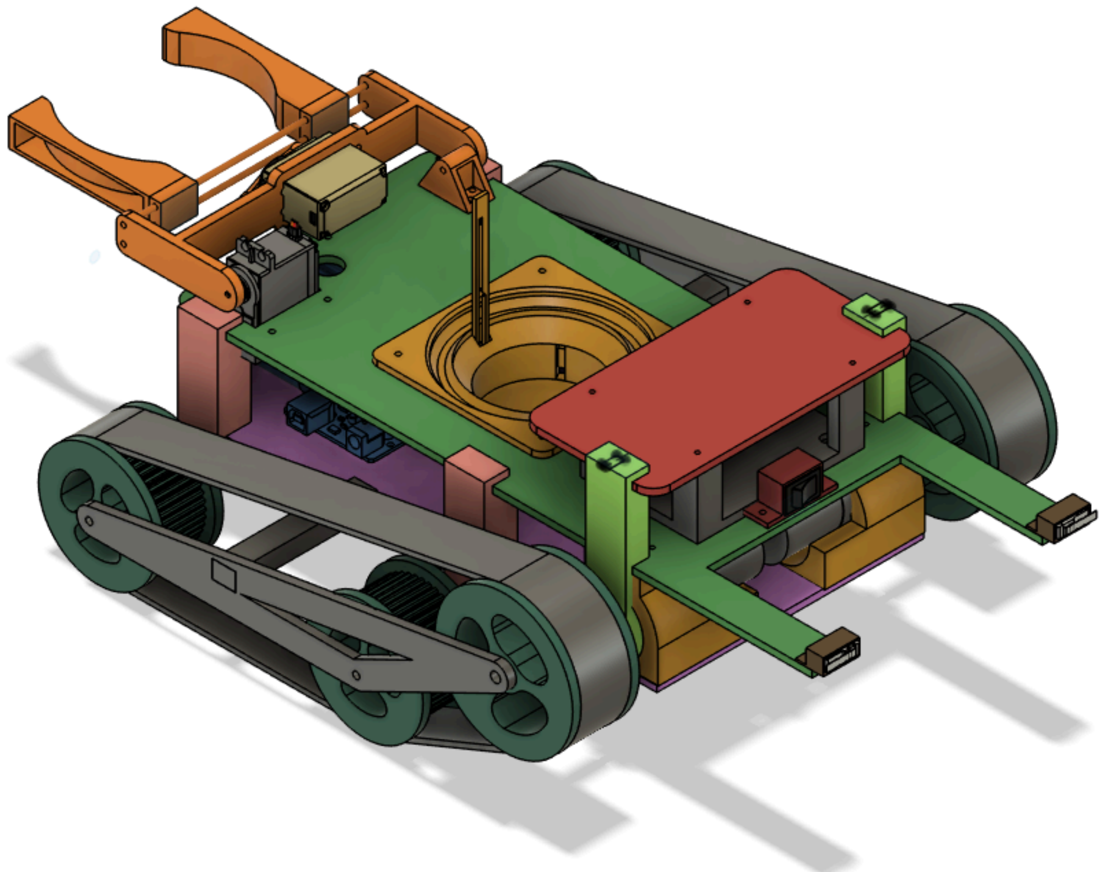
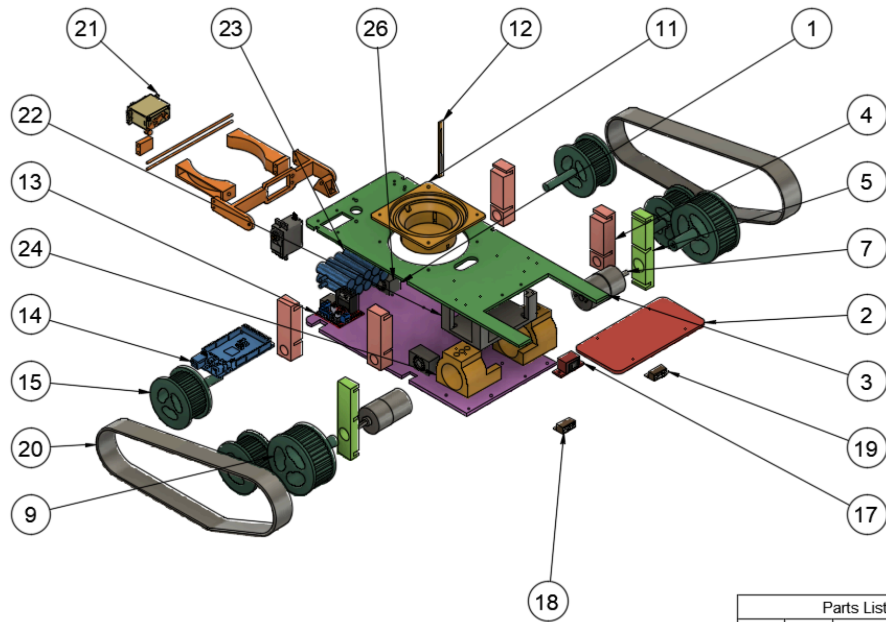


Figure 22. OTV Assembly



Parts List

Item	Qty	Part Number
1	1	BottomDeck
2	1	TopDeck
3	1	MiddleDeck
4	4	MS2Spacer
5	2	DoubleHeightSpacer
6	1	Motor_mount v2
7	2	Finalized motor
8	1	Motor_mount v2 (1)
9	2	Drive Wheel
10	1	SiloMockup
11	1	CargoInterfacingPlate_Plate
12	1	CargoInterfacingPlate_Antenna
13	1	L298N Driver DC Motor, Stepper Motor
14	1	ArduinoMega_STEP_AP203
15	4	Support Wheel
16	1	TCRT5000L Module
17	1	RA1113112R
18	1	P:\E-Switch Secure\3D_MODELS_STP_FORMAT\MS085xx02FxxxP1A.stp
19	1	P:\E-Switch Secure\3D_MODELS_STP_FORMAT\MS085xx02FxxxP1A.stp
20	2	Belt
21	1	2ServoClaw
22	1	Battery Cage
23	1	Battery
24	2	Motor_mount
25	1	Wheel Outer Support
26	4	Bottom Chassi Support

Figure 23. OTV Exploded View

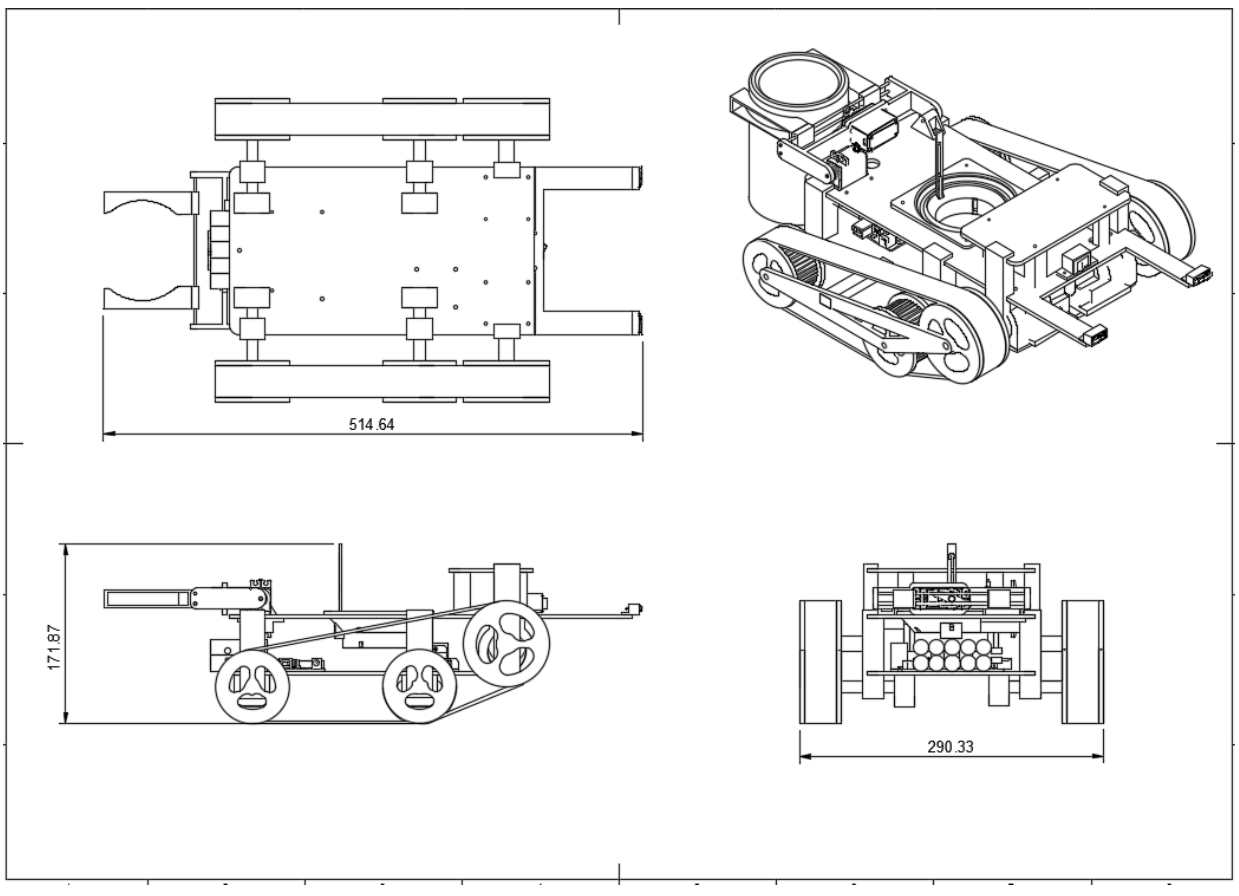


Figure 24. OTV Assembly Dimensions

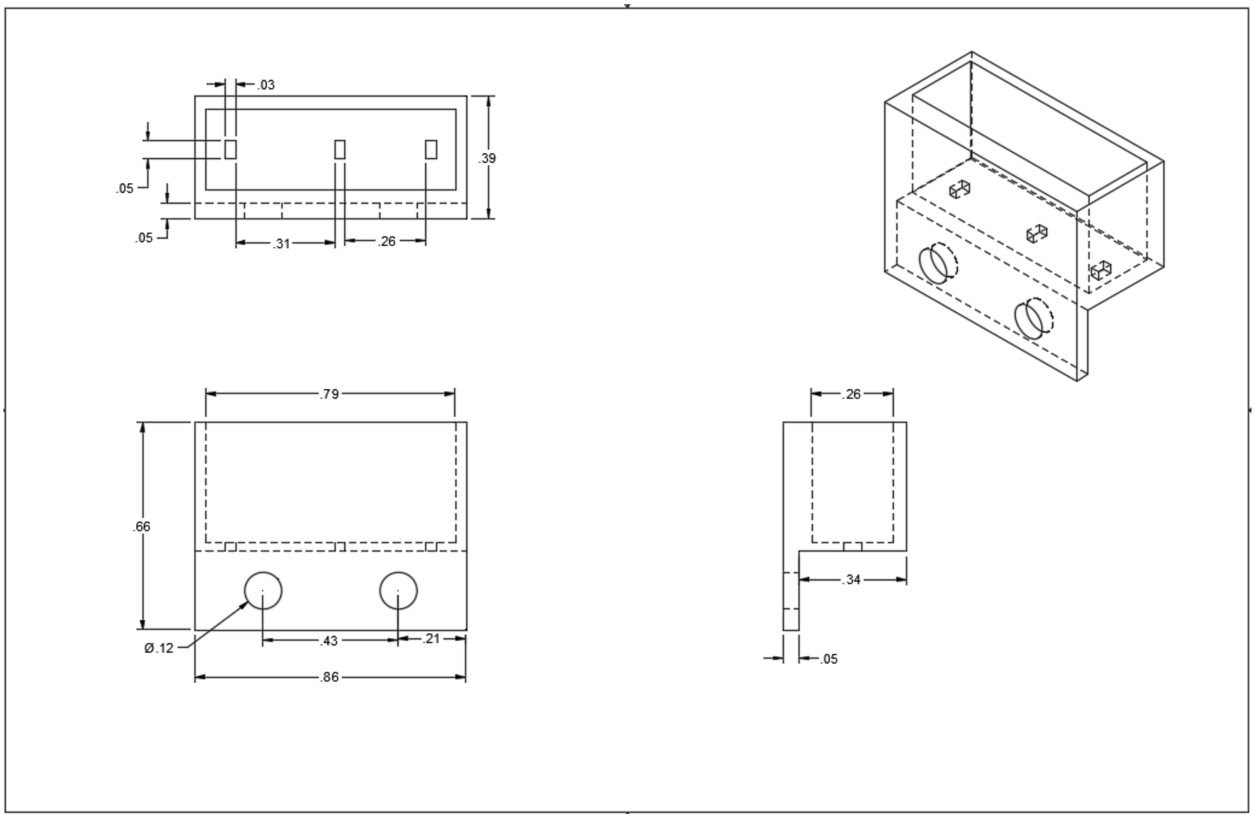


Figure 25. Left Limit Switch Cage

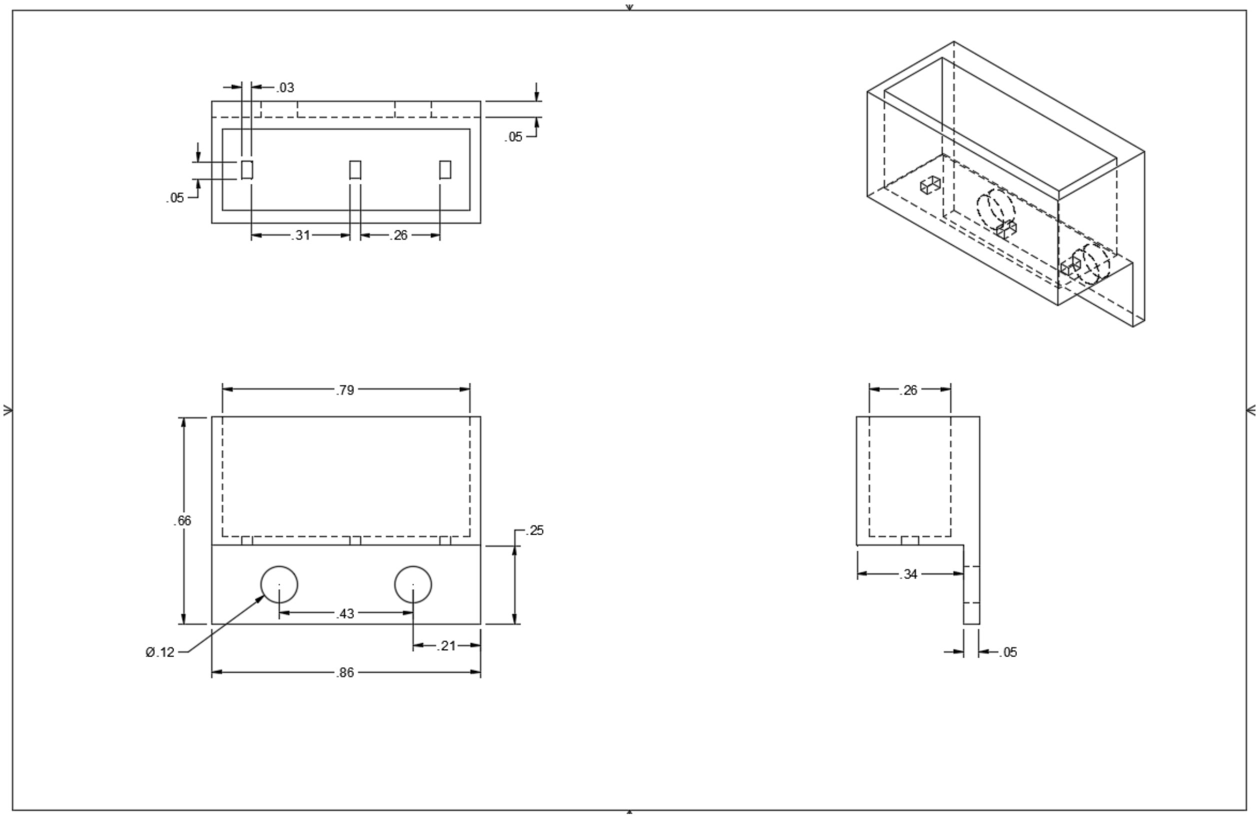


Figure 26. Right Limit Switch Cage

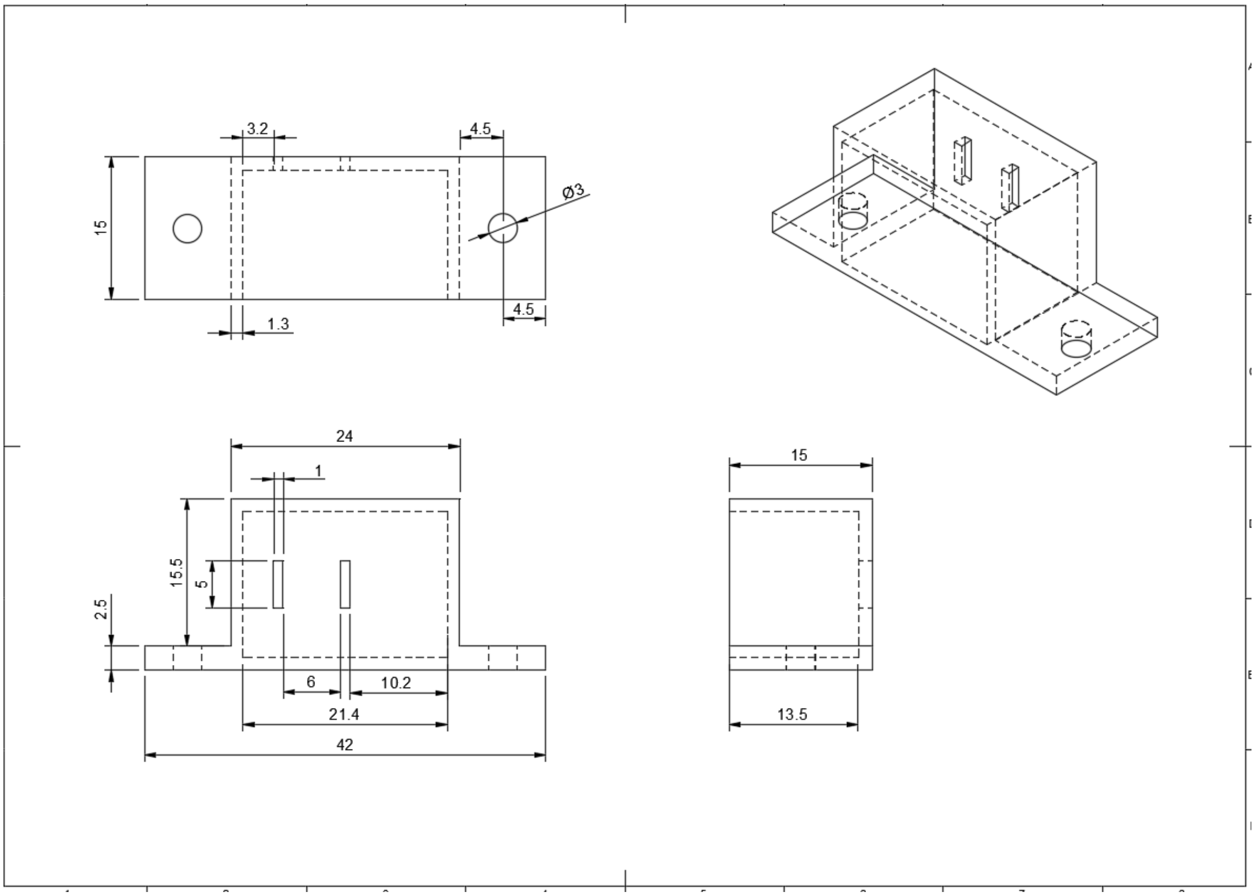


Figure 27. Kill Switch Cage

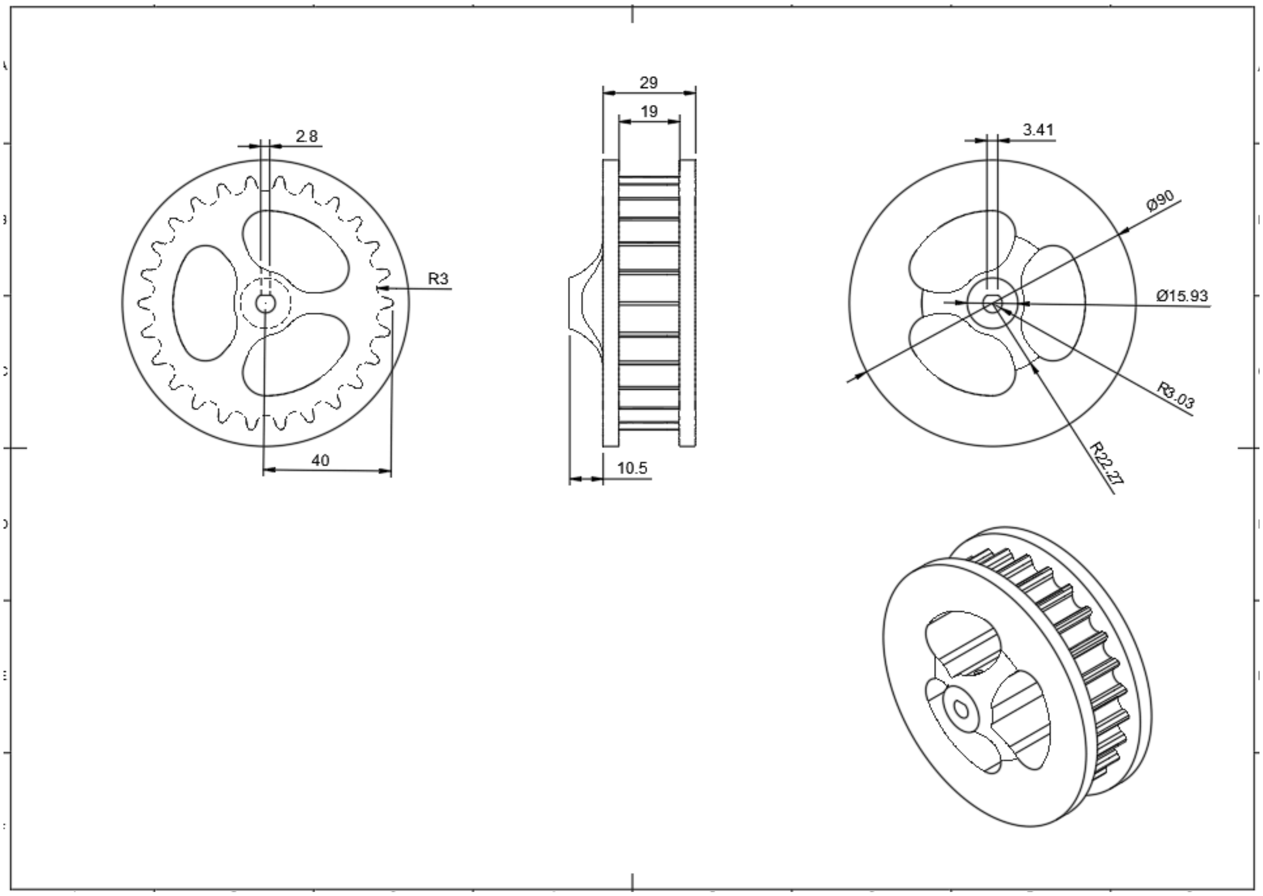


Figure 28. Drive Wheels

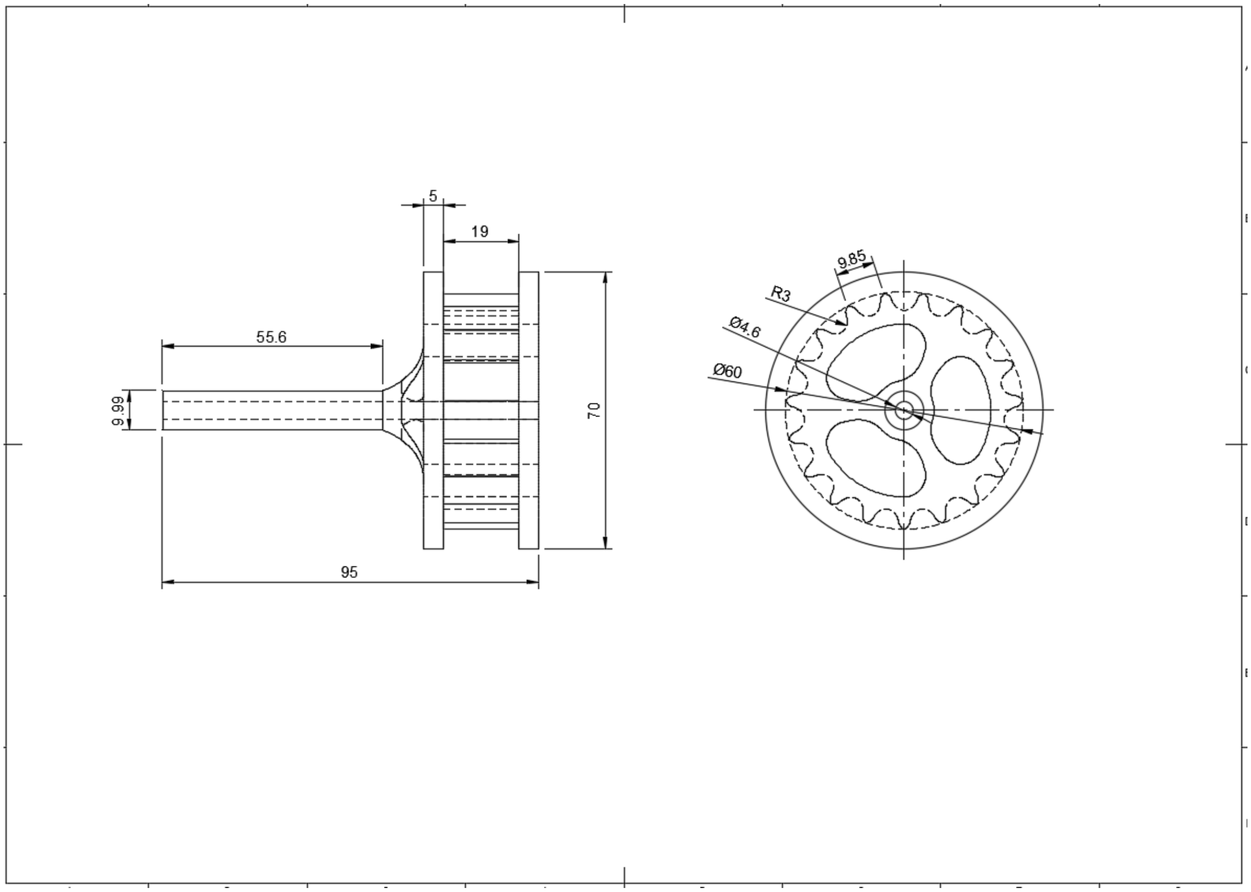


Figure 29. Support Wheels

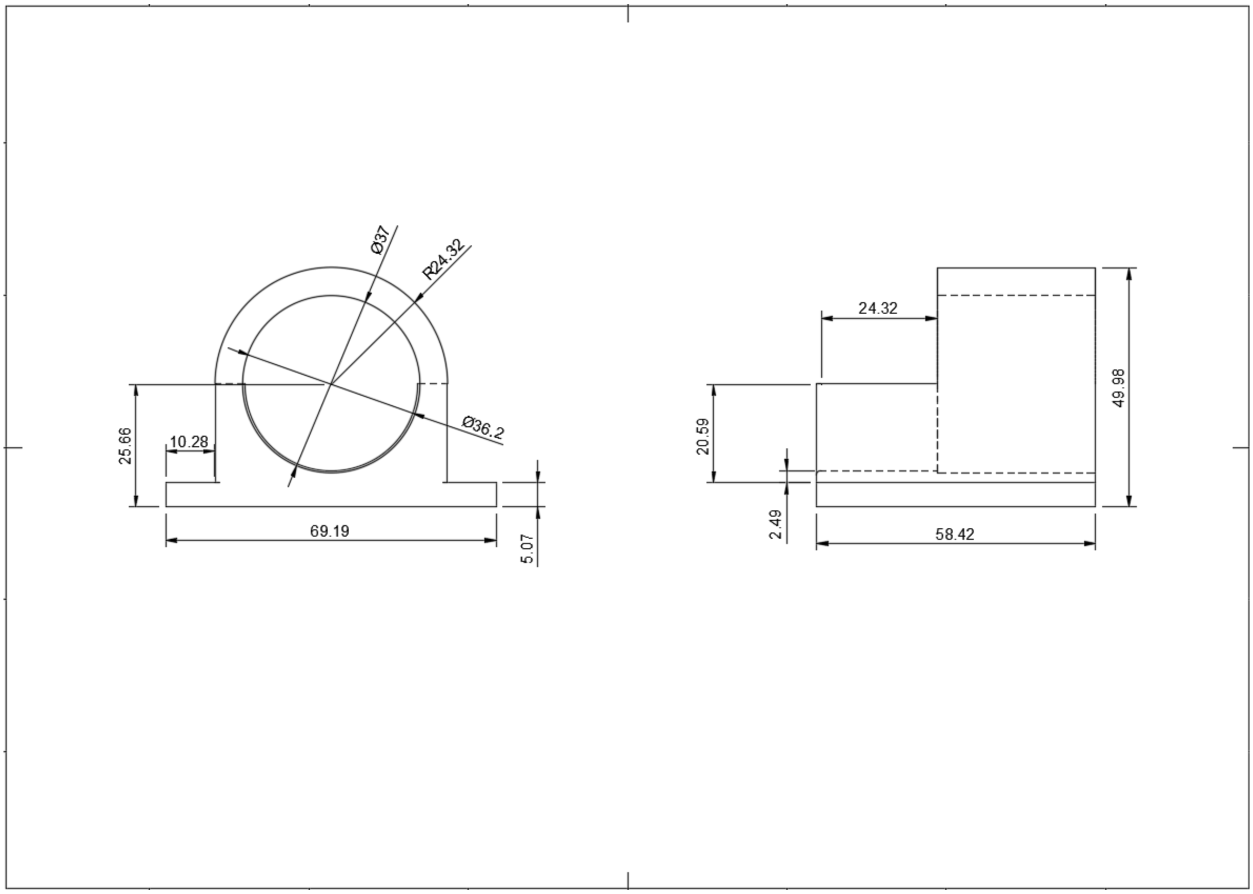


Figure 30. Motor Mount

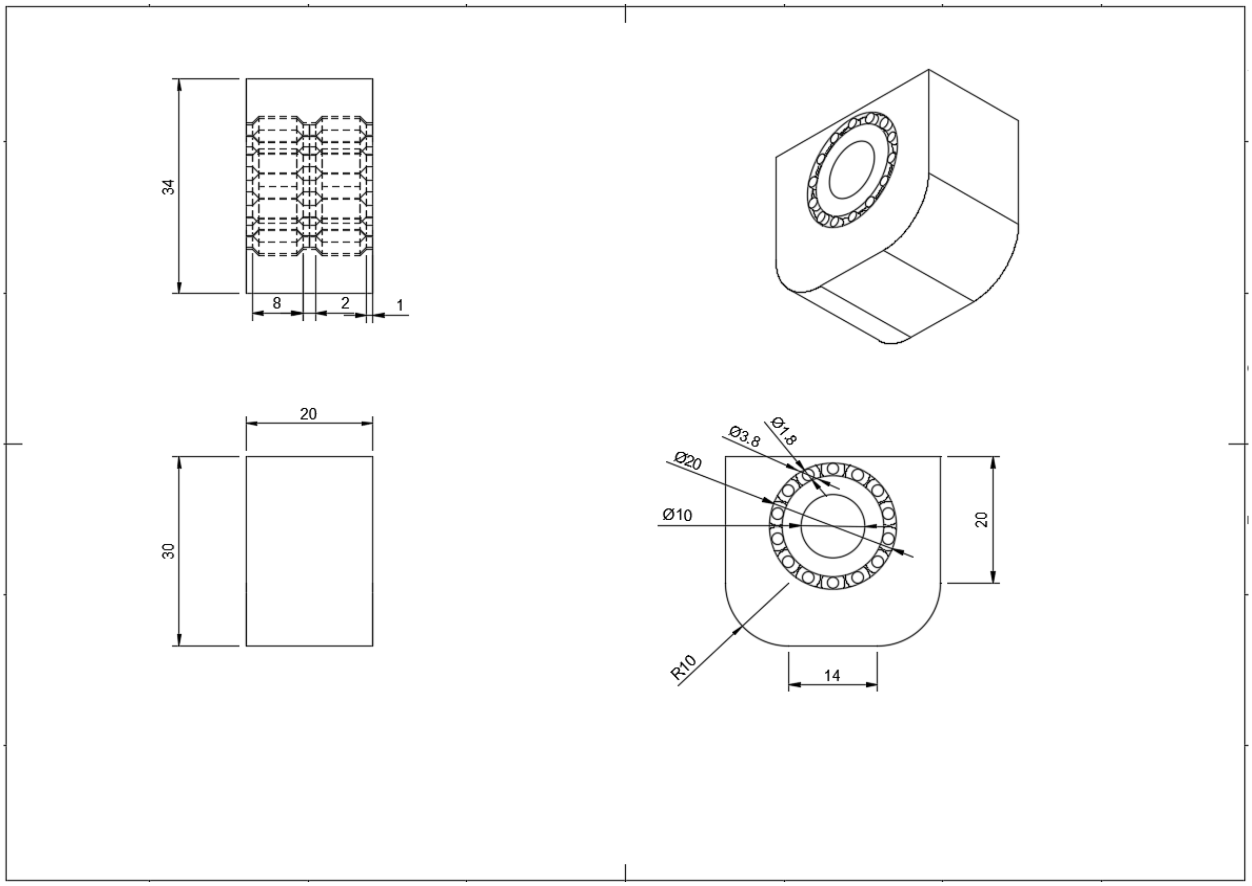


Figure 31. Bottom Chassis Support with Bearings

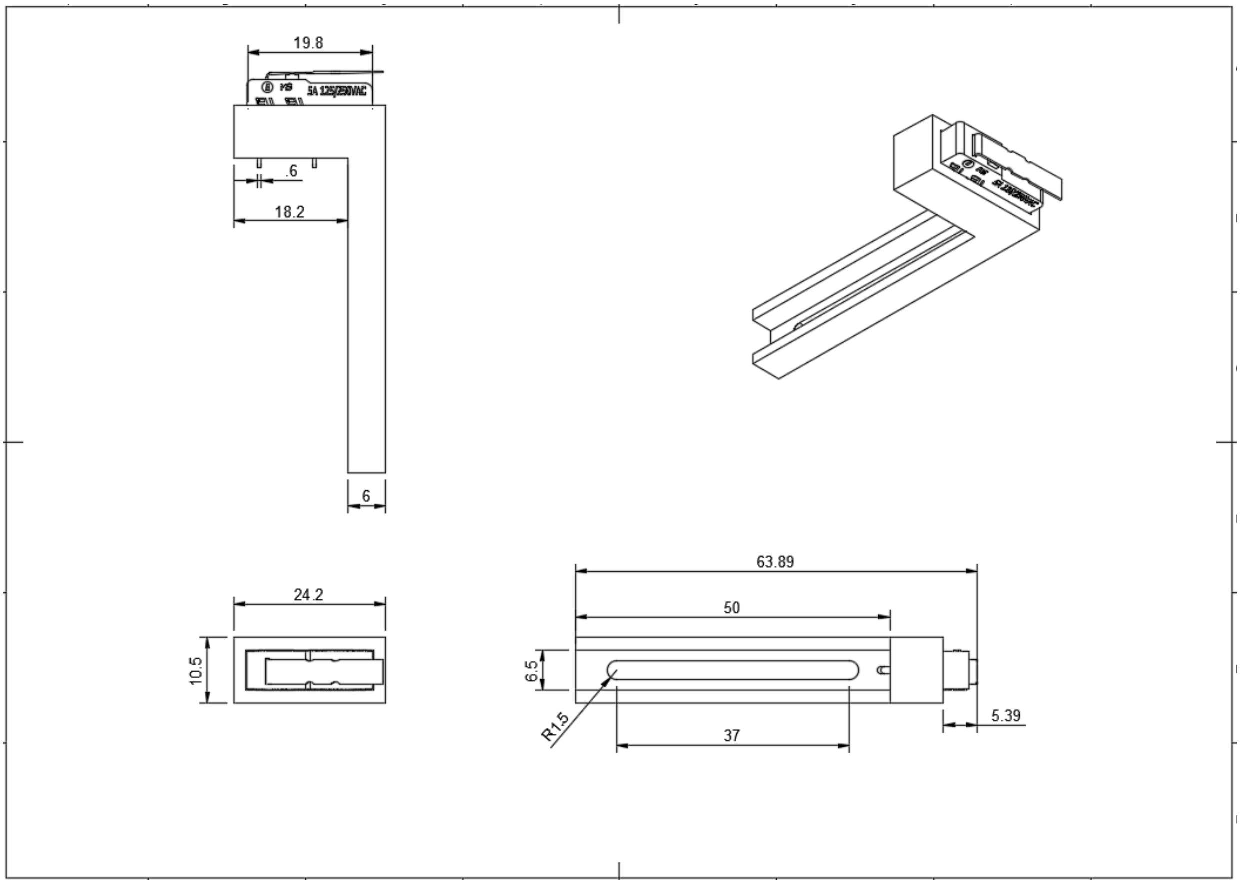


Figure 32. Sensor Mount

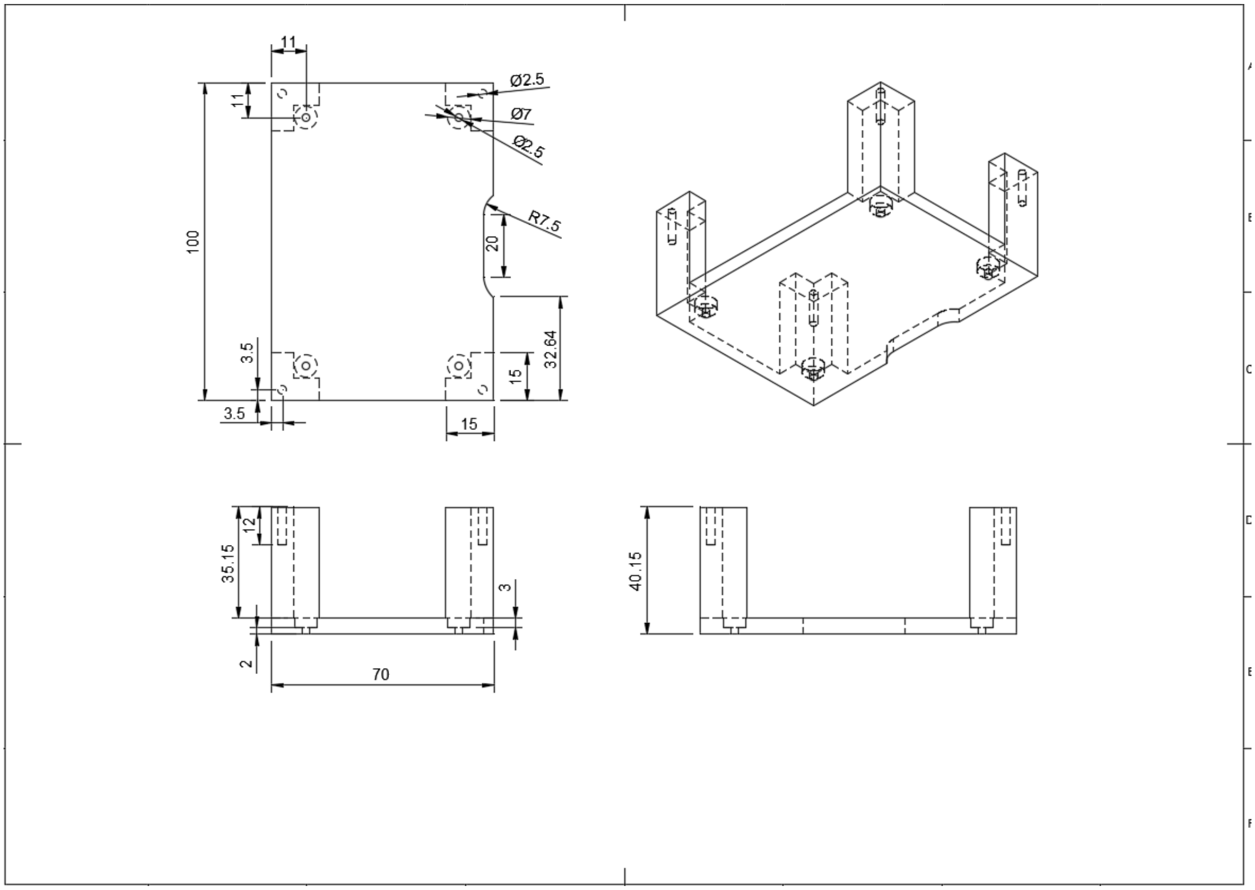


Figure 33. Battery Cage

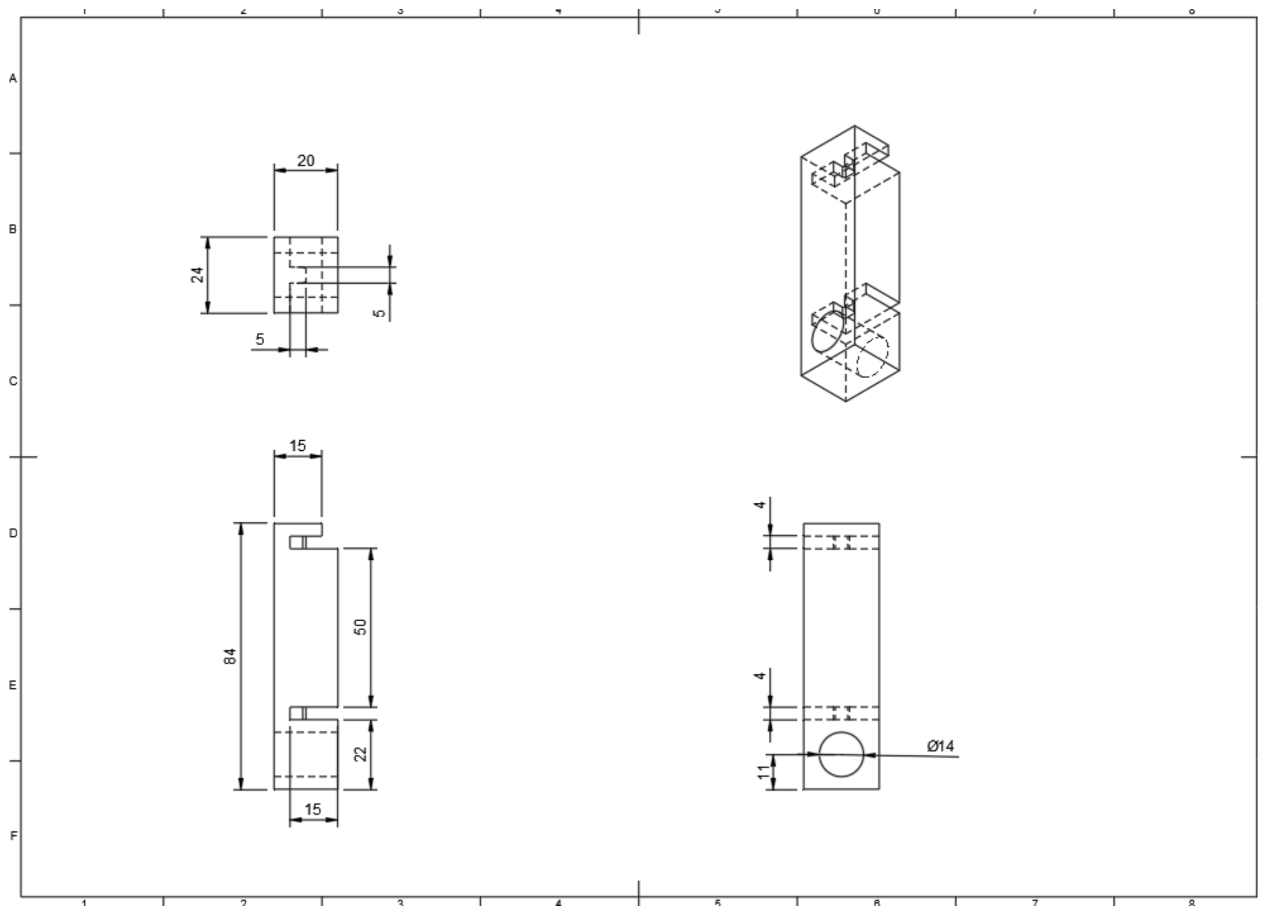


Figure 34. Spacer Without Bearing

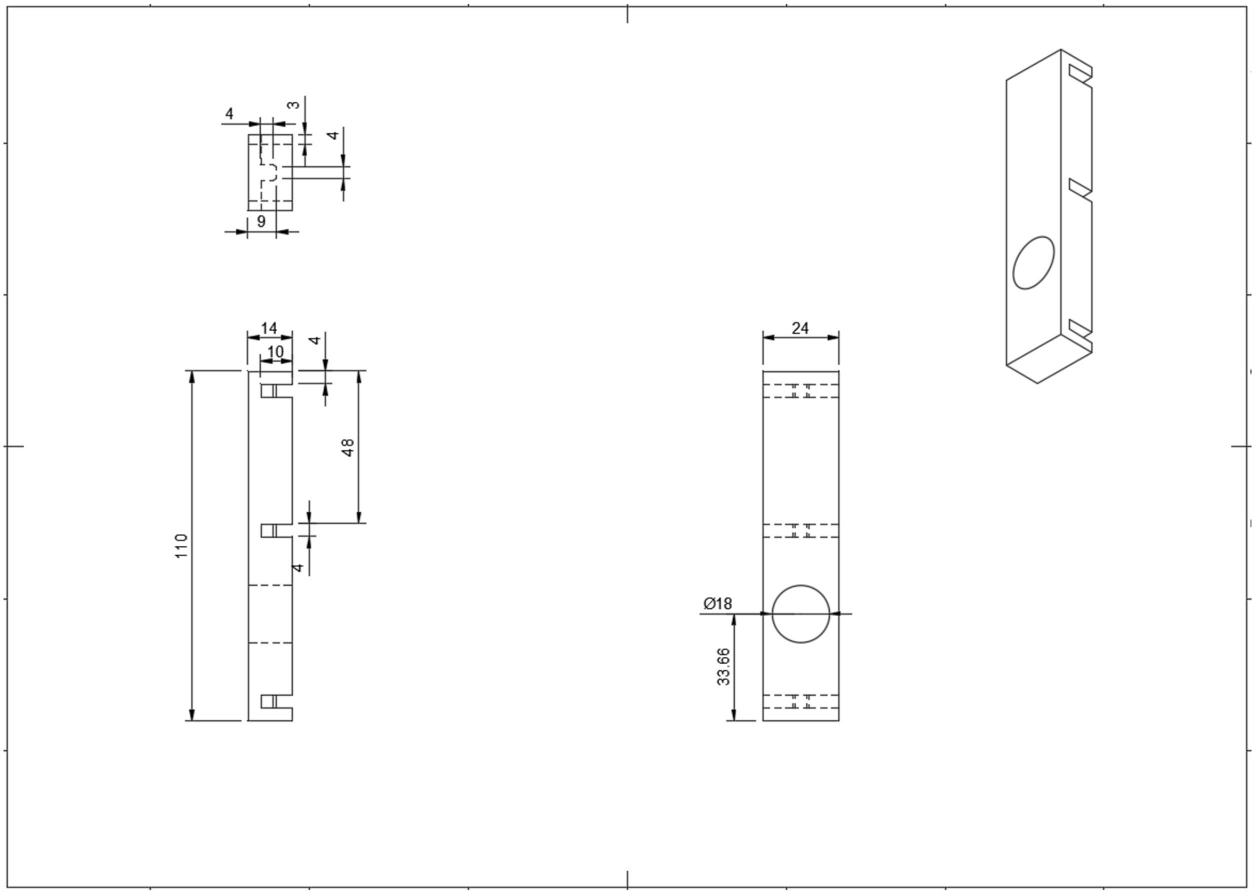


Figure 35. Double Height Spacer Without Bearing

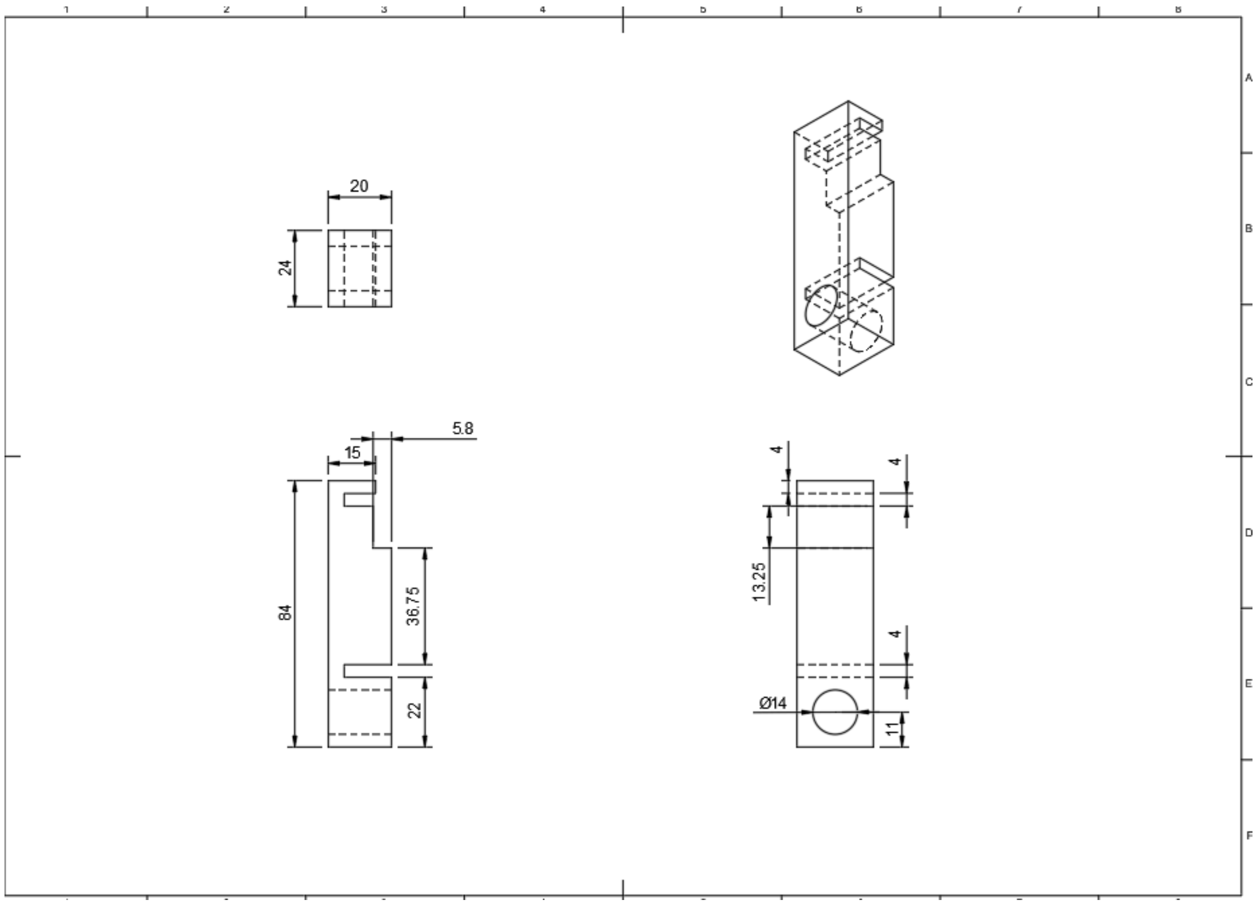


Figure 36. Spacer With Servo Cut Out Without Bearing

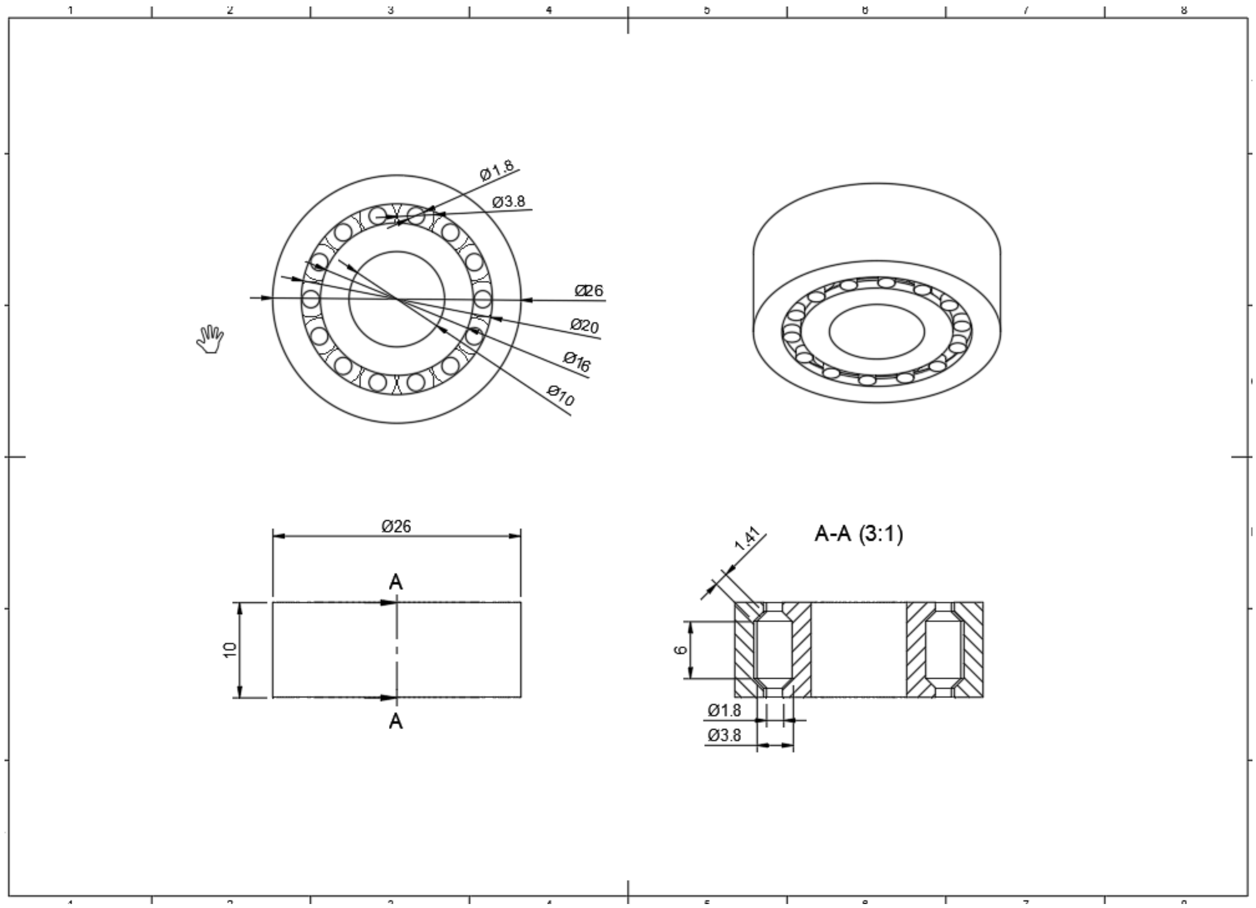


Figure 37. Ball Bearing

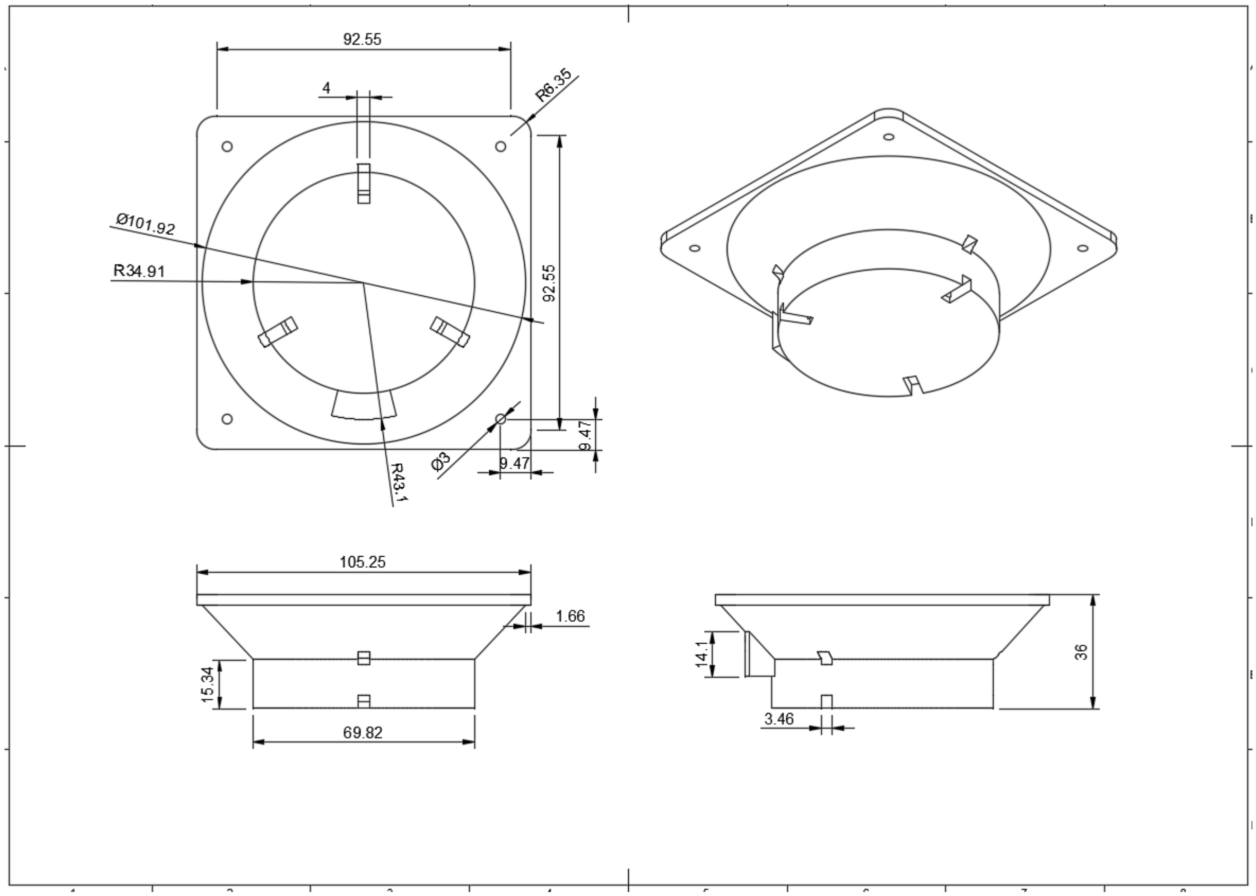


Figure 38. Cargo Interfacing Plate

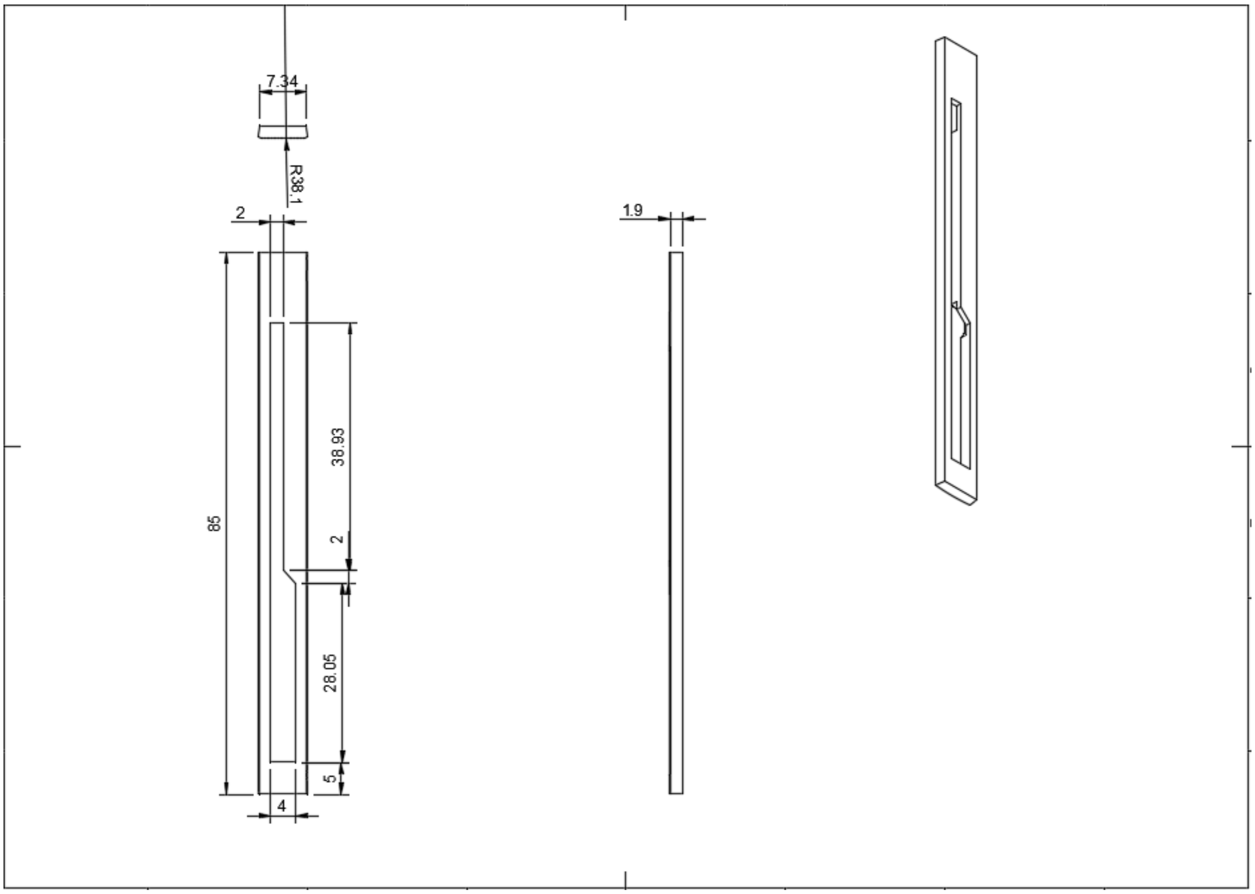


Figure 39. Cargo Interfacing Plate Antenna

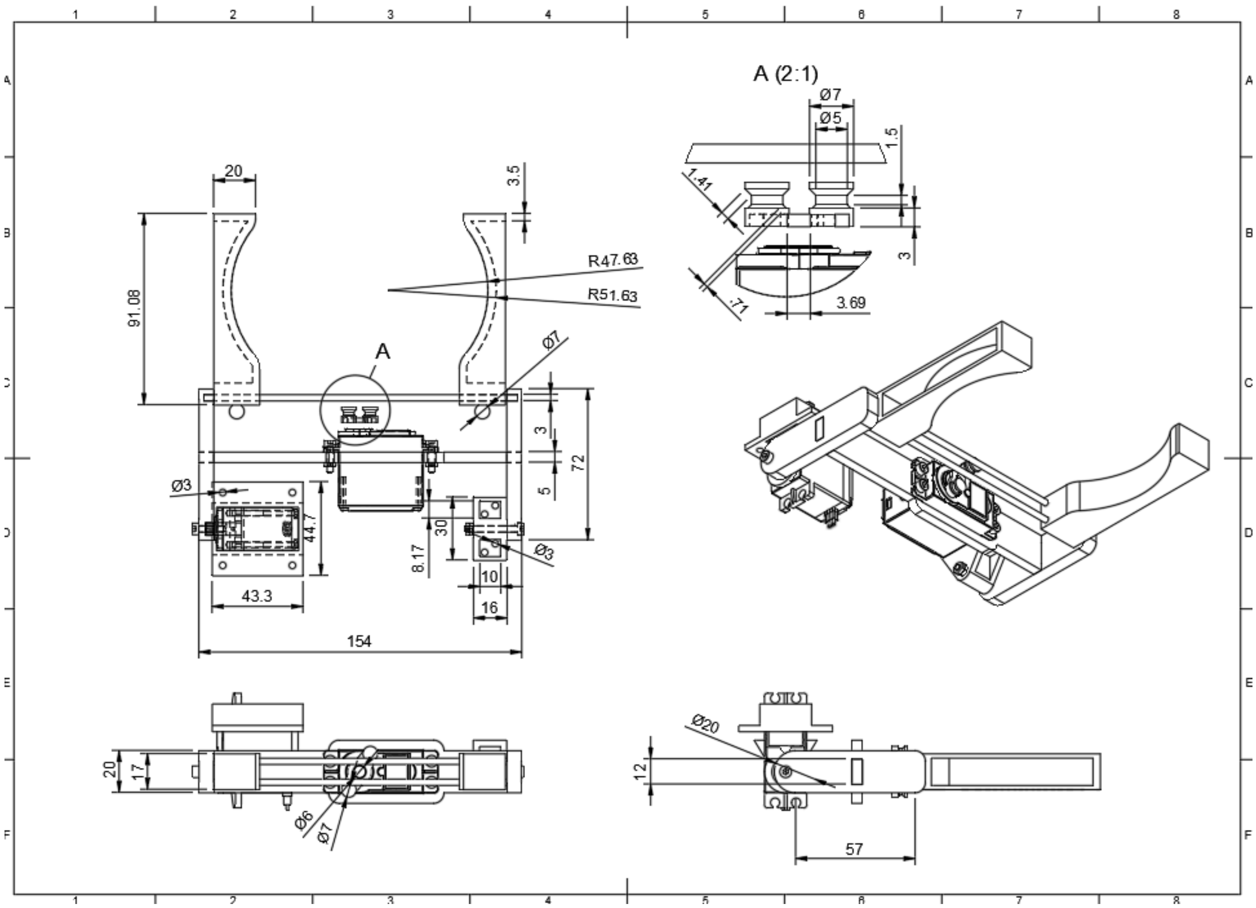


Figure 40. Claw Assembly

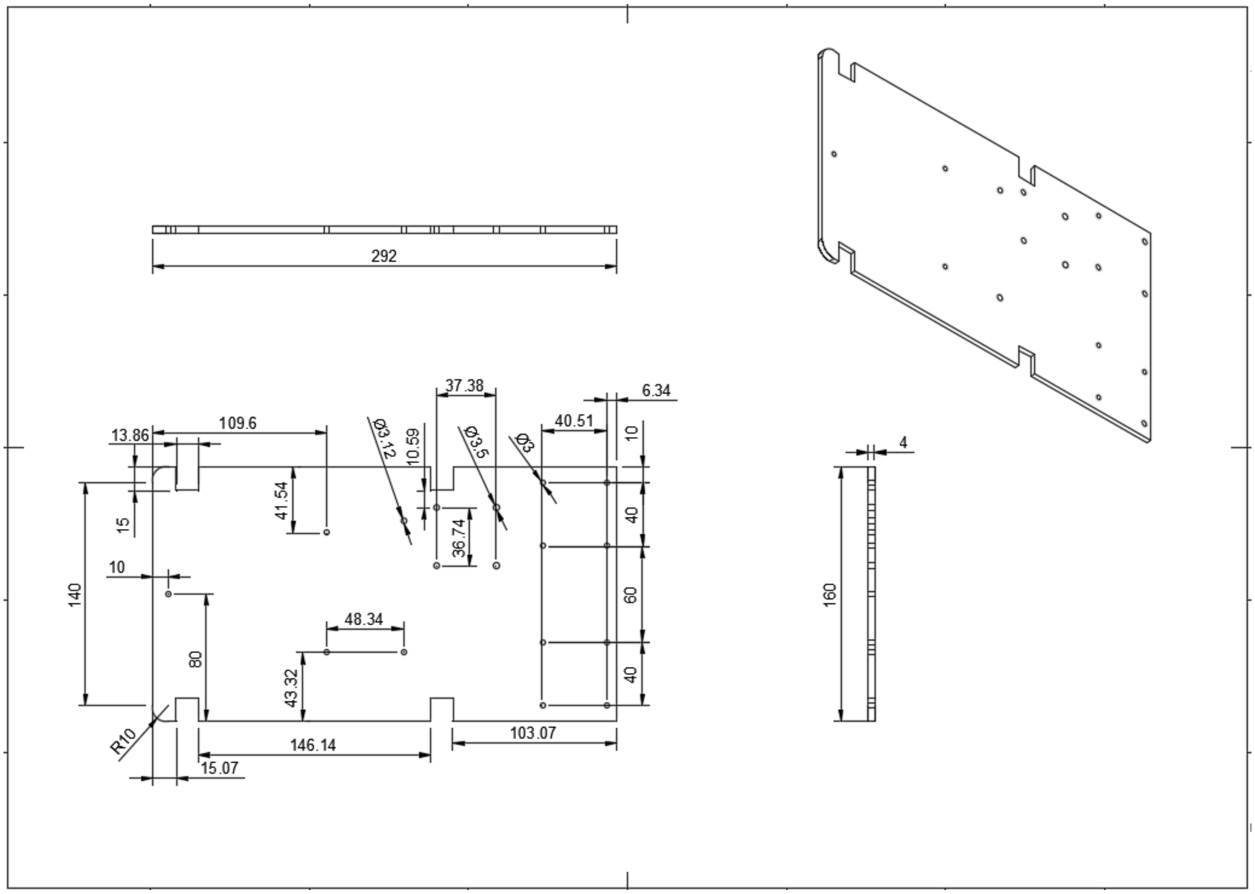


Figure 41. Bottom Deck

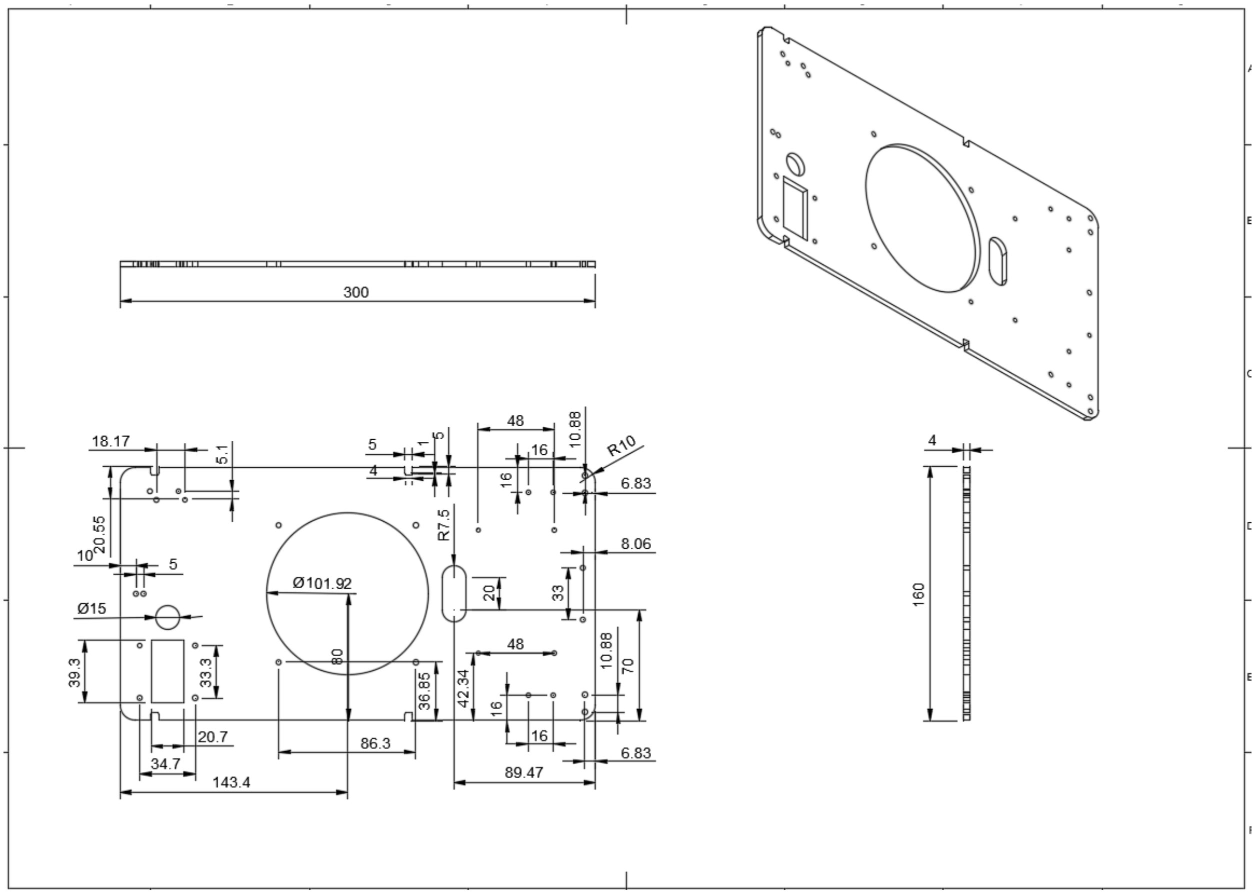


Figure 42. Middle Deck

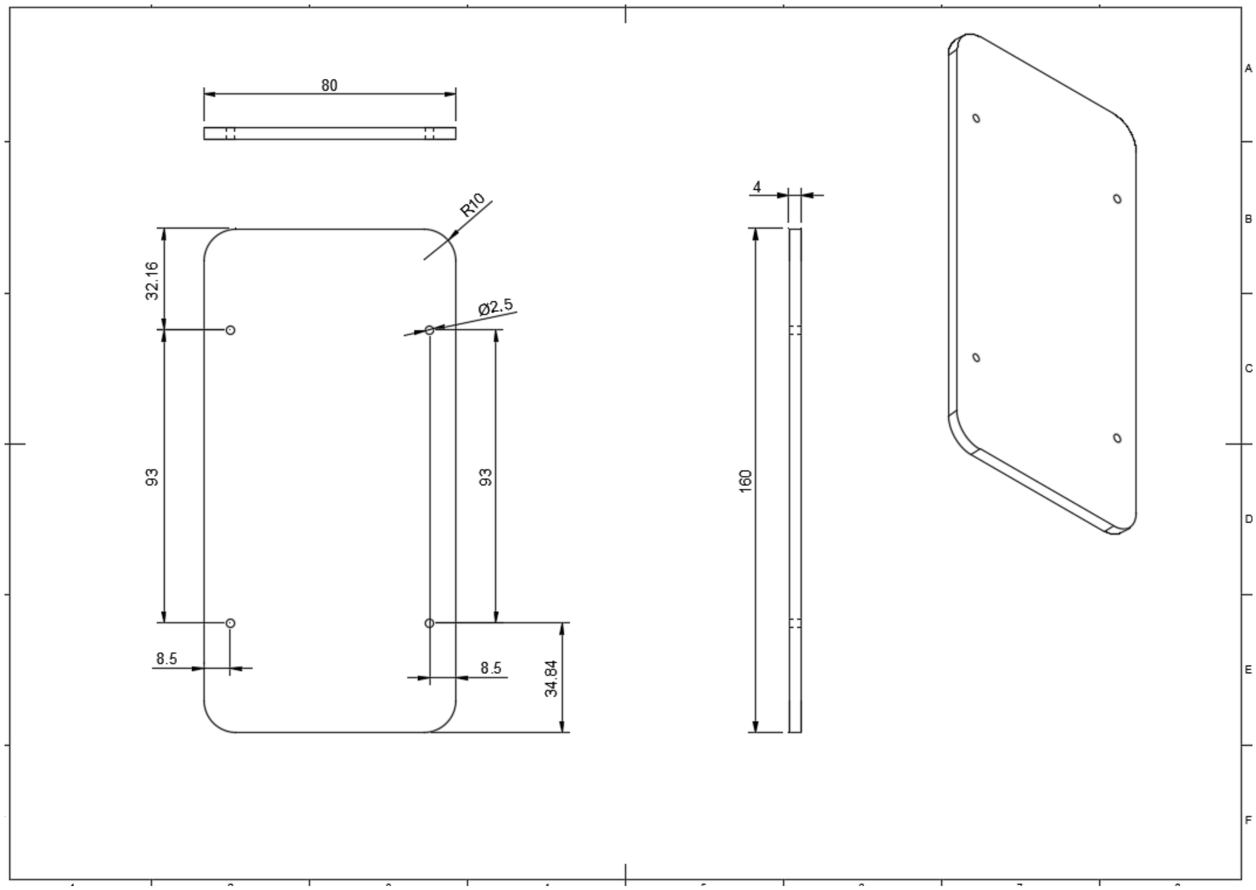


Figure 43. Top Deck

Design Brief 6 - Troubleshooting and Iteration

Share a specific example of how your team creatively approached problem solving. What tools, resources, techniques, and people were used in this creative approach?

A specific example of how our team creatively approached problem solving came during the development and testing of the mission sub-assembly of our OTV. Initially, our team believed that simple press-fit mounts for the servo motors and a single reed switch for magnetic detection would be sufficient to meet our functional goals. However, once testing began, we encountered numerous challenges that required quick thinking, collaboration, and creative use of available tools and materials.

As we began testing each electronic component needed for the MS5 milestone, Ryan encountered a significant issue involving two different servo motors—one positional and one continuous rotational. Since the rotational servo didn't return to a home position, precise control was challenging. To compensate, we conducted repeated volume testing within the pylon silo to determine the timing required for consistent operation, factoring in load and resistance. This ensured predictable positioning, which was critical for the successful transmission of data and the mechanical reliability of the system.

Next, we moved on to testing the puck detection system. Our initial plan involved using a single reed switch. Ryan spent over an hour testing its orientation in various configurations but found inconsistent readings. He consulted Steven, Vihaan, and Thomas, and together they observed that the reed switch performed most reliably when perpendicular to the puck's magnetic radius. From this insight, the team collaboratively designed a new interface plate that held three vertically-aligned reed switches. This ensured at least one switch would always detect the puck, regardless of its orientation. After the new plate was 3D printed and tested, we achieved a 100% detection rate—assuming the puck landed within the plate's boundaries.

Once component-level testing was complete, the mission sub-team assembled the entire upper PLA mission layer. That's when a new wave of problems began. Pin allocation and cable management became immediate issues, which we quickly addressed and reflected in our updated circuit schematic.

A critical mechanical issue arose when the claw struggled to close reliably. We had to consider two factors: the risk of stalling or burning out the servo from over-rotation, and the tensile stress on the thin wire connecting the servo to the claw arms. To reduce the strain on both components, we resourcefully applied 3M tape to the inner edges of the claw. This gave it additional grip on the pylon and allowed it to function without excessive torque or mechanical tension.

While assembling the servo responsible for flipping the pylon onto the interface plate, we noticed it was loose and giving inconsistent results. Ryan improvised by using gum wrappers as shims, wedging them between the PLA housing and the servo to secure it tightly. This low-tech fix stabilized the servo and drastically improved its consistency. After this adjustment, we achieved an 85% success rate in puck extraction (23 out of 27 attempts), a major improvement over earlier tests.

Integration with the lower propulsion layer introduced another challenge. Testing in the arena revealed that changes to the tracks and sprockets were altering the OTV's center of mass, which in turn affected how the claw sat relative to the ground. Once the final drivetrain configuration was set, the claw was angled upward and struggled to grip objects properly. To solve this, we first tried redistributing weight—specifically, moving the battery to the front—but this was not sufficient. Ryan and Thomas, during lab hours, proposed and carefully executed the solution of shaving down the PLA spacers supporting the claw. This adjustment leveled the claw and restored consistent gripping performance.

Throughout this process, our most effective technique was collaborative brainstorming. Every major hurdle was addressed through open discussion among teammates. While our solutions were not always high-tech, they were resourceful and made full use of the tools and materials around us—including lab supplies and even trash like gum wrappers. We leveraged CAD tools for redesigns, the 3D printer for rapid prototyping, and constantly validated changes through iterative testing.

Ultimately, the mission sub-team succeeded not because we had the most advanced tools, but because we creatively applied what we had—both in terms of materials and minds. The combination of team communication, hands-on experimentation, and clever repurposing of resources allowed us to overcome obstacles and deliver a highly functional OTV.

Design Brief 7 - Teamwork and Project Management

How did your team plan for integration of subsystems? What would your team do differently to make integration go more smoothly?

To plan for effective subsystem integration, our team adopted a layered design approach for our OTV. We physically divided the OTV into two main layers, allowing for parallel development and testing of individual subsystems. This structure enabled different sub-teams to focus on their respective areas without dependency delays and gave us the ability to identify and address integration issues early in the process.

The first layer was dedicated to the propulsion and electrical subsystems. This layer included the motors, Arduino, H-bridges, and other electrical components required for mobility and power. It also contained the mechanical elements needed to implement a tank drive, such as the wheels, bearings, and treads. By isolating these components on their own platform, the propulsion and electrical teams were able to build and test their systems independently, without having to wait for the mission subsystem to be completed. This independence proved especially valuable for hitting early project milestones, which required a functioning drive system before the full OTV was ready.

This approach allowed for faster iteration and hands-on learning. For example, while testing just the first layer, we discovered that specific weight distributions were necessary to consistently climb over a log on the course. This insight allowed us to strategically place heavier components on the second layer during planning. In addition, we learned how to properly secure wheels and spacers to the chassis, which informed our mounting strategy when integrating the second layer. We also addressed wiring challenges early, allowing us to organize sensor and actuator wiring in a way that would accommodate both layers. Since the propulsion layer was optimized for wiring, we were able to plan how second-layer components would route their wiring to avoid interference and ensure clean integration.

The second layer was dedicated to the mission subsystem. It housed critical components such as the claw, limit switches, reed switches, and electrodes used to detect and complete mission objectives. These elements required high precision and reliable operation in order to grab and flip the pylon onto the OTV, detect the puck’s magnetic field, and read the duty cycle from the pylon's signal. As a result, this layer took longer to build than the first. However, the separation of layers ensured that the additional time spent on mission development didn't delay the propulsion and electrical subsystems. To support this process, we divided the coding sub-team between layers, allowing one coder to focus entirely on navigation using the first layer, while another worked on the mission system's code independently on the second layer.

Thanks to this layered architecture, final integration of all subsystems was relatively smooth and straightforward. Code integration was also seamless, as it had been structured from the beginning into distinct navigation and mission stages. However, one challenge we faced after full integration was the limited accessibility to the electronics housed in the first layer. Once the second layer was mounted, troubleshooting electrical issues became difficult due to the tight space. In cases where we suspected a hardware issue, we often had to remove the second layer entirely, fix the issue, and then reinstall and rewire the second layer, which took significant time and delayed testing.

In hindsight, to make integration go more smoothly, we would have planned for more accessible placement of key electronic components. Locating these components in areas that could be reached without disassembly would have reduced troubleshooting time and allowed for more efficient testing.

In conclusion, our layered approach allowed each subsystem to be developed and refined independently while minimizing bottlenecks. It led to more effective integration, better subsystem performance, and a clearer path to project milestones. For future iterations, improving accessibility and planning for serviceability will be a top priority.

Design Brief 8 - Sustainability

See MS9_Sustainability_Report.pdf in the current directory.