Milestone 3: Design Blueprint

Team 2: Data Collection

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

---

Chassis Design Brief

The main goal of the chassis is to provide the structural support for the mission systems while maintaining a form that is lightweight and maneuverable. The main requirements are as follows.

- Position the claw subassembly at a height where it can grab near the top of the silo
- Have ample space on top of the chassis for the interfacing plate.
- Provide a location to mount the aruco marker such that it is visible at all times, and ideally so that it is centered with respect to at least one axis
- Provide ample space to house and arduino, a battery, 2 motor mounts for locomotion, and a handful of small electronics including the kill switch, wifi module, and navigation sensors.

The main challenge we faced with the design was balancing the requirements (particularly spatial requirements) that our mission system has versus keeping the chassis maneuverable.

We decided to do a 2.5 layer chassis in order to make the most of our specific design requirements. Because we needed a relatively tall chassis to be able to position the claw at the right height, we decided to add additional layers so that we could reclaim the area taken up by our interface plate and avoid expanding the chassis outwards in order to mount our electronics.

The bottom “deck” has the mounts for support wheels on our locomotion system. The design has an “electronics bay” between the bottom and middle deck” which houses our arduino, motor controller, line follower, mission systems limit switch, and drive motors. The top surface of the middle deck is mostly taken up by the mission system, composed of a claw subassembly which is bolted to the deck, and the interface plate, which sits in a hole in the deck and is secured with bolts. The space at the back of the deck houses the battery, kill switch, and limit switches for navigation. Finally, a shorter top deck is mounted above the battery, on which the Aruco marker is mounted.

The decks of the chassis are made of cast sheets of recycled PLA plastic which have been laser-cut with the necessary hole patterns. They are separated by 3D printed spacers which double as support brackets for the locomotion system. The spacers will be located using pins and have their connection reinforced with glue. All other components will be bolted to their respective deck using M3 hex screws and M3 nuts.

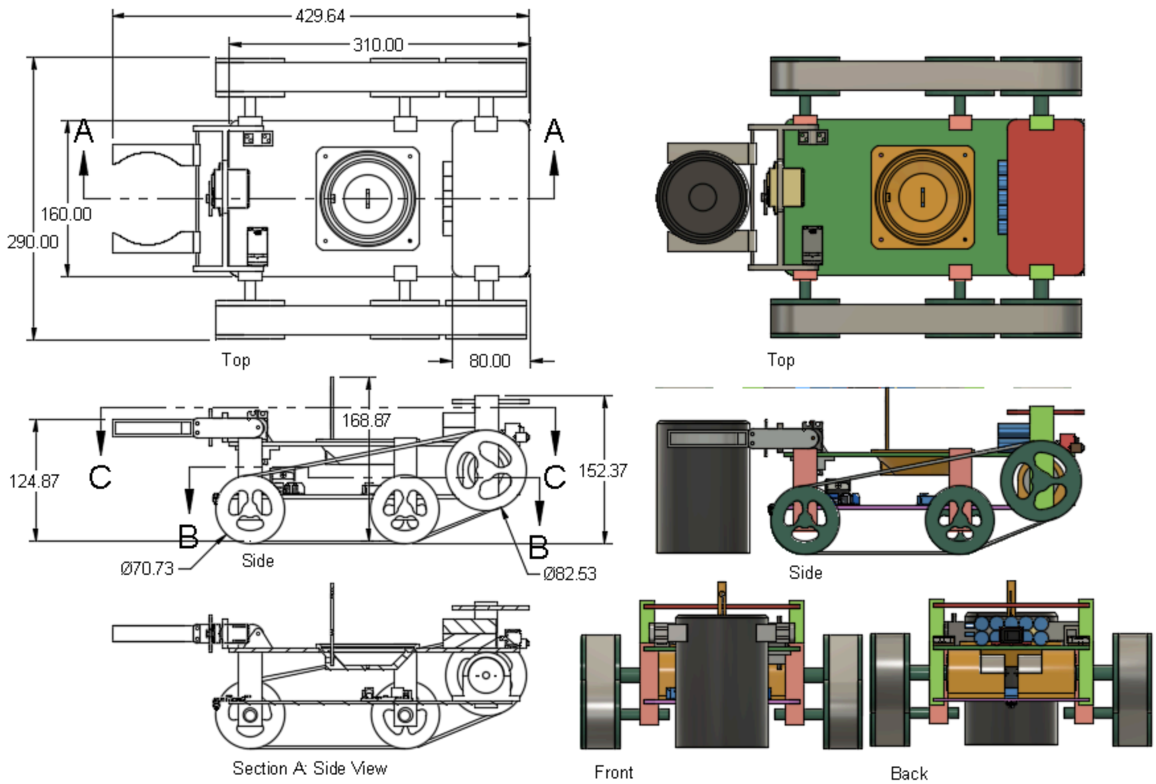


Figure 1. Chassis Overall Dimensions, Color CAD Images (with pylon mockup for reference)

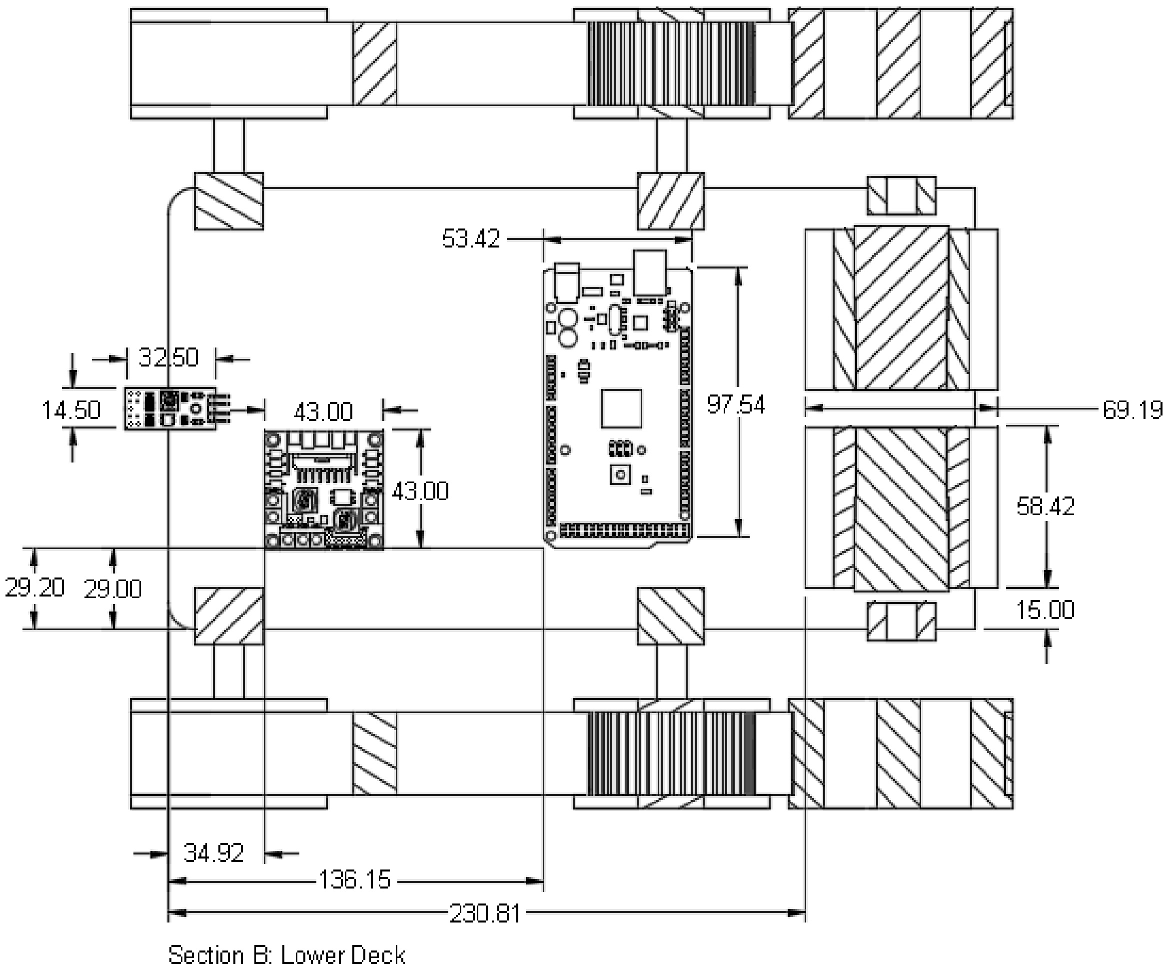


Figure 2. Bottom Deck Detailed View

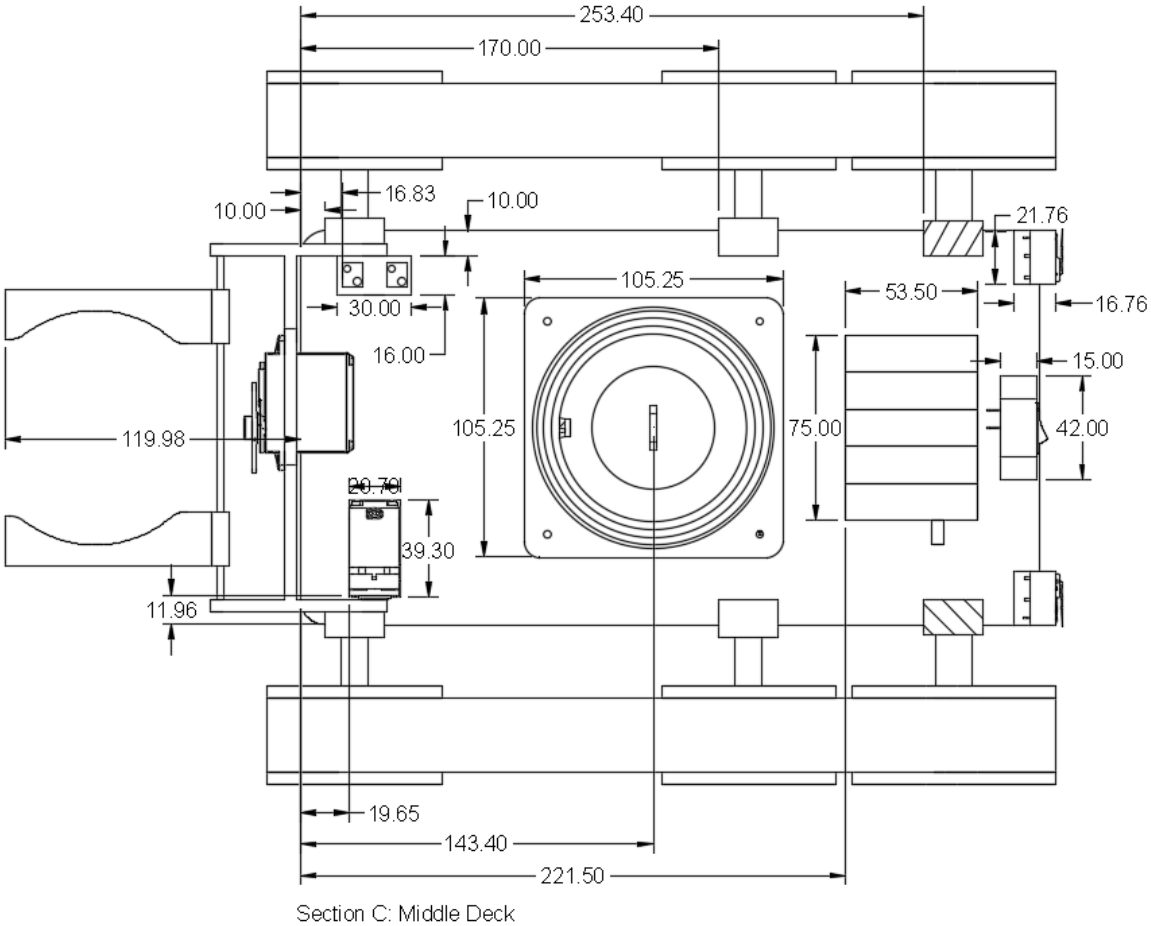


Figure 3. Middle Deck Detailed View

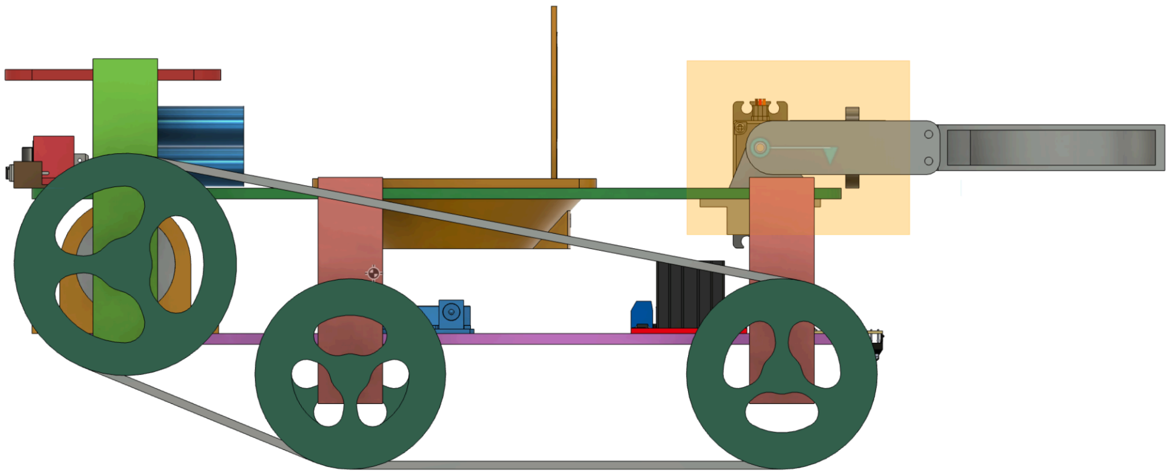


Figure 4: Colored View (with center of gravity) from the Y-Z Axis

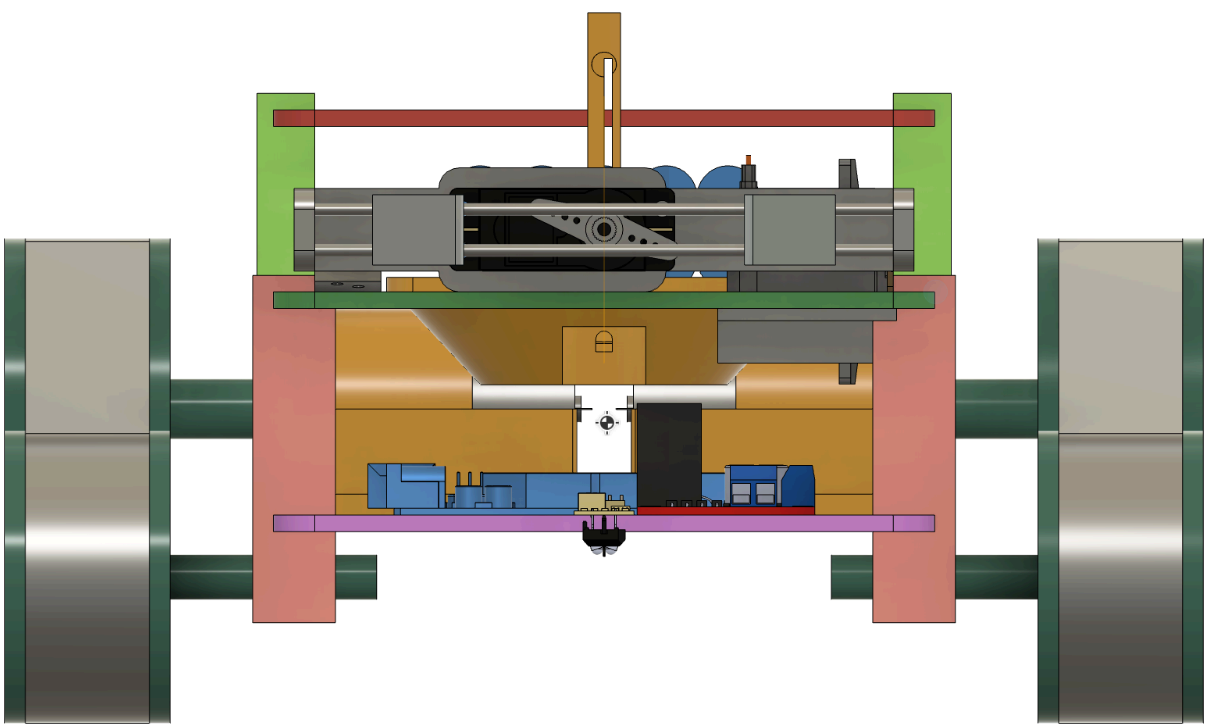


Figure 5: Colored View (with center of gravity) from the X-Z Axis

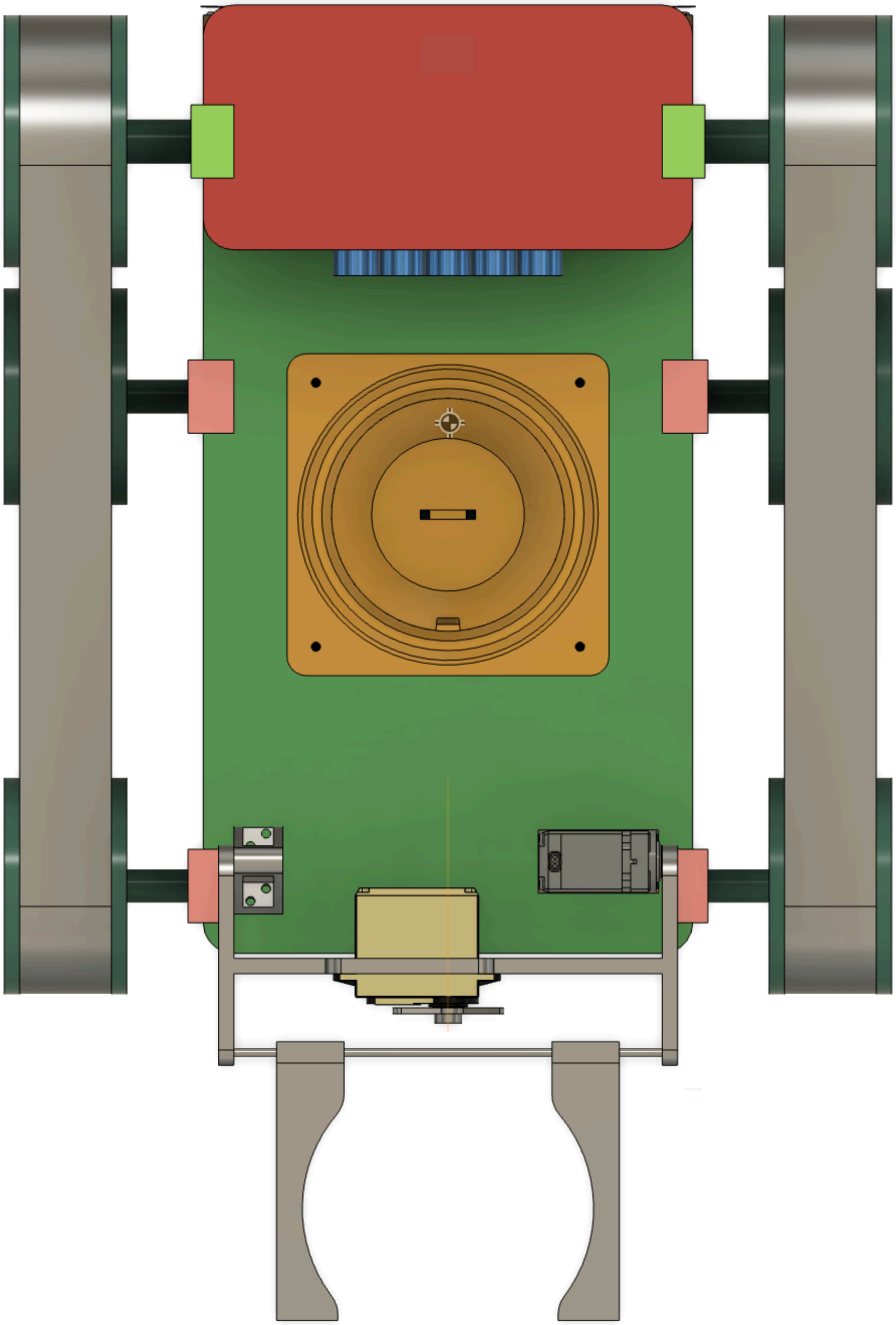


Figure 6: Colored View (with center of gravity) from the X-Y Axis

As seen in these images, Fusion 360 calculates that the center of gravity of the OTV assembly is aligned with the center of the x-axis, between the middle and bottom plates, slightly towards the battery end of the OTV. Since the center of gravity is close to the geometric center of our OTV, this will make it easy for the OTV to navigate, specifically for 90 degree turns because the OTV will rotate about a point close to its geometric center. Plus, it will make it simple for the Arduino subteam to adjust for the shift in center of gravity. Lastly, when going over the log, since the center of gravity is near the side that will approach the log, the OTV will be able to go over the log smoother once the center of gravity is over the log.

Mission Design Brief

Our OTV must be able to precisely locate the data pylon, extract the puck from the data pylon, determine whether or not the puck is magnetic, read the square wave signal from the two electrodes inside the data pylon, and transmit the data collected (square wave and magnetic). The main challenge we foresee is being able to precisely locate the data pylon. This is a difficult task to have the OTV perform, as the OTV does not know precisely where the pylon is in the arena, and it does not have the same intuitive sensing abilities that a human operator would have. We will have to use sensors and control logic to locate the pylon with enough precision to perform the other parts of the task. The second challenge is performing the tasks required inside the pylon, as they require precise movement of the OTV's mission systems and the shape of the pylon limits space for movement.

We chose to use our "garbage truck" mechanism in order to circumvent the requirements for precise movement in confined space. This design grabs the pylon with a claw and rotates it 180 degrees onto an interfacing plate with embedded sensors. The only moving parts on the mechanism are the closing claw and the rotating claw arm, both of which interact with the data pylon externally, and thus are not confined by the limited space within the pylon. Provided proper movement of the claw, all data collection can be performed by non-moving parts on the stationary interface plate.

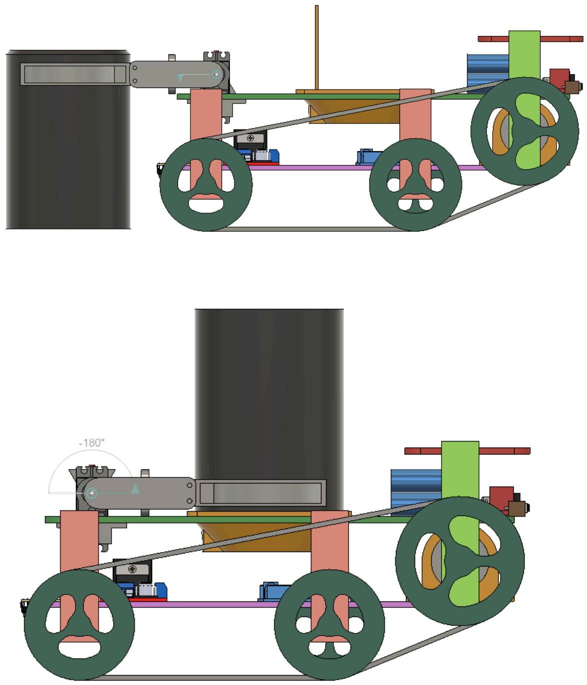


Figure 7. Claw rotation demonstration

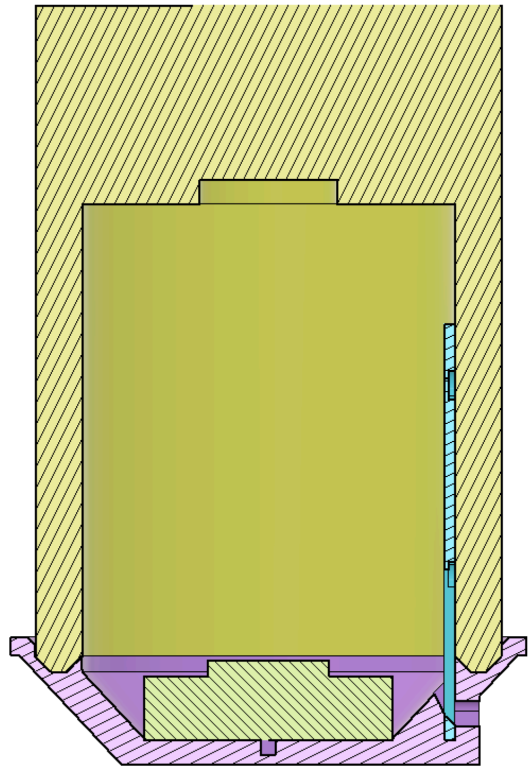


Figure 8. Cross-section showing mating of the electrode antenna (blue), interface Plate (pink), and pylon/puck mockup (yellow).

For the mission, we designed an interface plate, electrode antenna, and claw subassembly for our mission system.

The interface plate will be 3D printed and attached to the middle deck of the chassis using four m3 screws. The opening of the pylon should fit flush with the opening on the interface plate, creating a contained environment for our sensors to operate in. The interface plate features a depression at the bottom with sloped walls, guiding the puck under the influence of gravity such that it falls onto a reed switch (passive magnetic field sensor) mounted with glue the rectangular indent visible in the center of the plate. The interface plate also contains a slot for our electrode antenna to fit into under a tolerance fit and holes in the bottom for the wires from the reed switch and antenna to feed into the electronics bay.

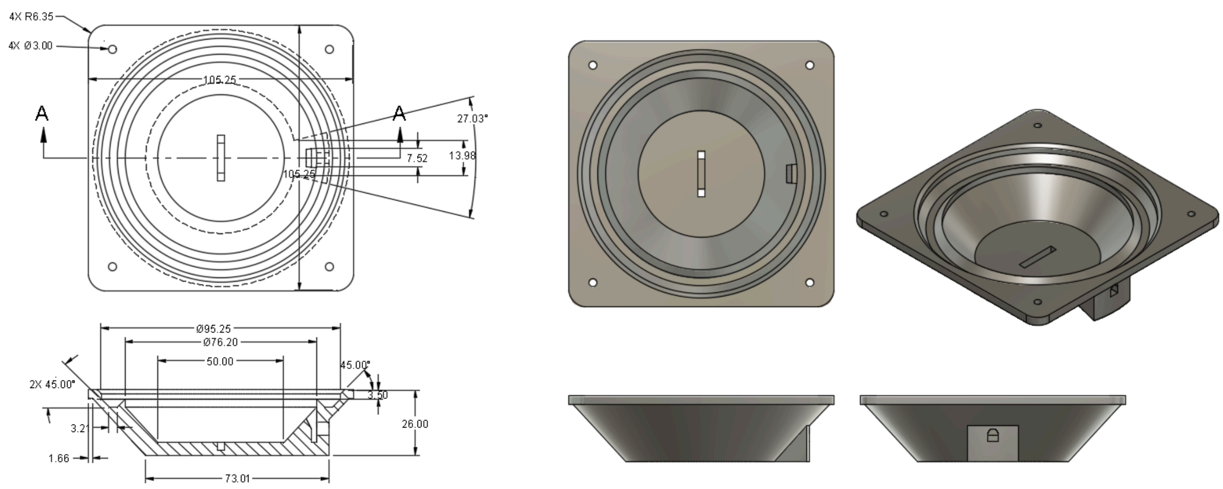


Figure 9. Interface Plate

The electrode antenna is a flexible piece of 3D printed PLA dimensions 85mm x 8mm x 2.3mm with two circular depressions of 6mm diameter and 1mm depth positioned 40mm apart. These depressions are meant to house electrode ends created by flowing solder over the stripped ends of two 2mm wires placed with their ends in said depression. The antenna also features a full-depth slot which is 4mm wide up to the first electrode and 2mm wide between the first and second electrode, accommodating both 2mm wires from the electrodes. The wires will be secured in the antenna using a small amount of liquid rubber or similar material to increase durability while maintaining flexibility.

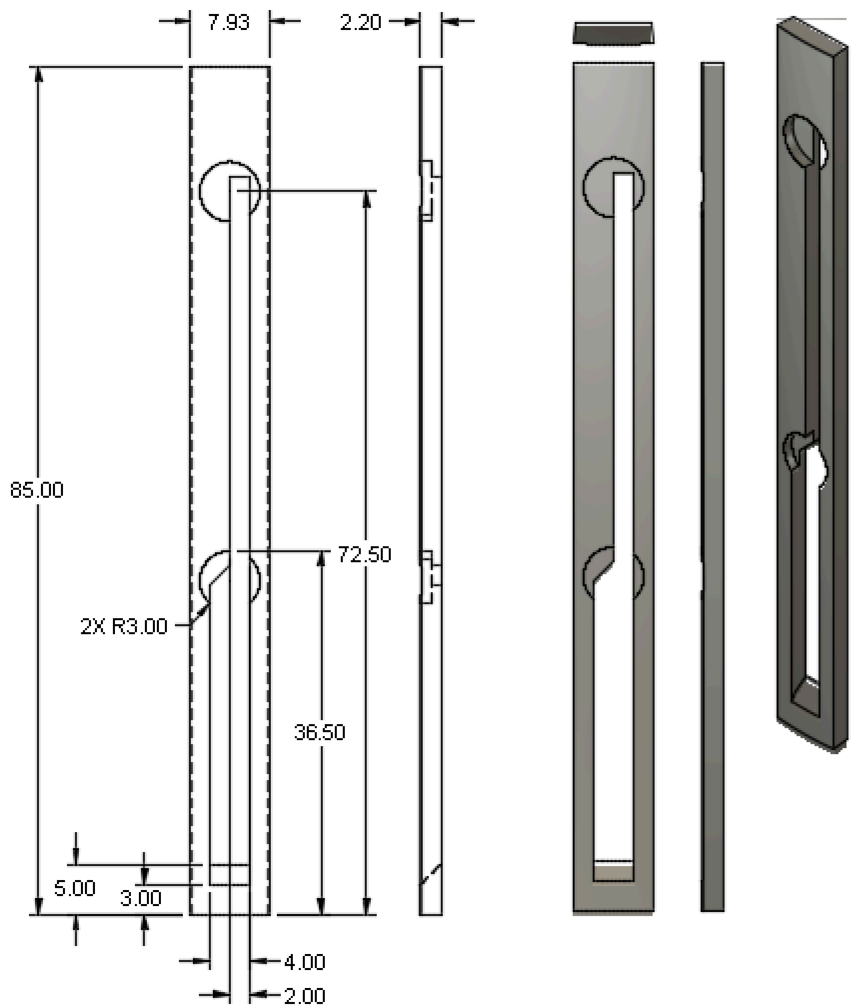


Figure 10. Electrode Antenna

The claw subassembly is going to be primarily 3D printed (with dense gyroid infill), using also a pair of 3mm carbon fiber tubes as guide rods for the sliding jaws, steel wire as linkages, two servo motors, and M3 bolts for assembly.

The claw is composed of a number of parts:

- The claw arm, made of three 3D printed plates fit together with a glue-reinforced tolerance fit.
- The arm drives servo, which rotates the claw arm to create the “flipping” motion. It is attached to the claw arm by screwing into a heat-set insert. The servo is mounted to the chassis with a tolerance fit in a slot in the middle deck, and is supported by a bracket.
- The servo mount, which is 3D printed and attaches to the underside of the deck using four m3 screws with nuts.
- The joint mount, which provides a second point of stability for the rotating claw mechanism. Its axis of free rotation is collinear with the axis of the arm drive servo. It is 3D printed, attached to the claw arm via a short piece of 3mm carbon fiber tubing and secured to the deck via four M3 screws with nuts.
- The claw servo, which is used to close the claw. It is tolerance-fit into a slot in the center of the claw arm and secured with four m3 screws with nuts.
- The claw servo arm, which is a stock part that comes with the claw servo and is secured with a tolerance fit and provided screw.
- The guide rods, which are cut from 3mm carbon fiber tubing, and are secured with a tolerance fit into pockets on either side of the claw arm. Once the arm is glued together, the guide rods cannot slip out.
- The claw jaws, which are 3D printed with a rubberized inner surface and ride on the guide rods.
- The claw linkages (not shown in drawing), which are constructed of 2mm steel wire and are 54mm long (not counting the length used for securing to the pins on the jaws and servo arm).
- The linkage pins, which are a pair of anchors that connect the linkage wires to the servo arm.

FBD and torque calculations for claw:

- When calculating for torque, we used 8.0 N for the pulling force of velcro as a safety factor against error.

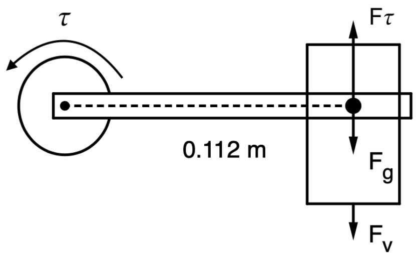


Figure 11. Free body diagram for claw

F_v = pulling force of velcro
m = mass of pylon
r = length of arm
F_g = force of gravity

$F_V = 8.0$ N
m = 0.2 kg
r = 0.112 m

$F_g = (0.2 \text{ kg})(9.8)$
$F_g = 1.96$ N

$\tau = (r)(F_V + F_g)$
$\tau = (0.112 \text{ m})(8.0 \text{ N} + 1.96 \text{ N})$
$\tau = 1.11 \text{ N m} = 11.3 \text{ kg cm}$

When choosing the servos for the claws, we had to consider the torque required to lift the pylon off the ground and flip it. For our specific mission, the pylon is always located in the center of the mission zone, velcroed to the ground. Thus, we must also consider the force of the velcro as well as the force of gravity of the pylon when choosing servo motors. For the force of velcro, we used 8N as per force values given in the source under “Velcro Peel Strength” in the sources section on page 60. After the above calculations, we found that the torque required for the arm drive servo must be above 11.3 kg-cm. Additionally, we assumed that the claw servo should have a similar torque in order to properly hold the pylon. The servos we chose are the MG966R servo motor for the claw servo which has 9.4 kg-cm torque at 5V and the FS5115M servo motor for the arm drive serve which has 14 kg-cm torque at 5V, giving us enough torque to lift and flip the pylon. If we do need more torque, these servos can increase torque at larger voltages. We would need to purchase a voltage regulator for 6V, as seen in the budget, Figure 46, this is feasible.

The claw assembly will work in tandem with two sensors. The first is a line follower secured to the bottom deck with M3 bolts. This will be used, in conjunction with the vision system and WiFi module, to navigate to the mission site. The second sensor is a limit switch mounted to the middle deck on a sliding mount, so that the exact point of detection can be adjusted during testing. When the limit switch detects the pylon is centered between the claw jaws, the jaws will be closed via the servo mounted in the claw arm, then the claw arm will be rotated 180 degrees by the servo fixed to the chassis, placing the pylon upside down against the interface plate.

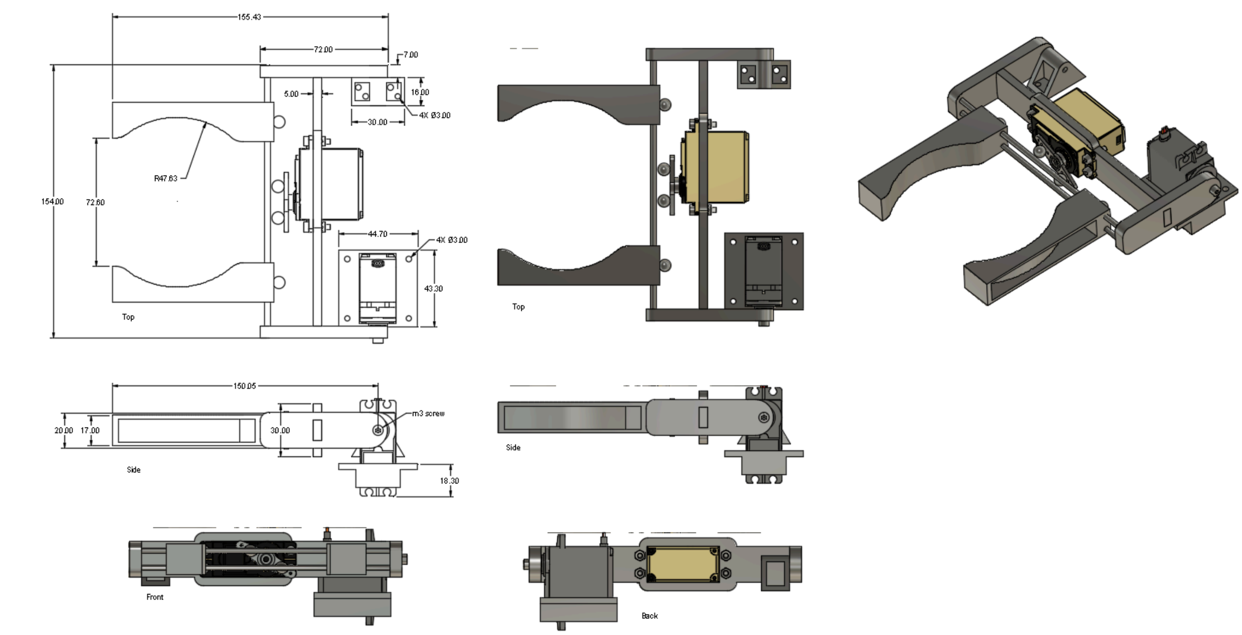


Figure 12. Claw Assembly

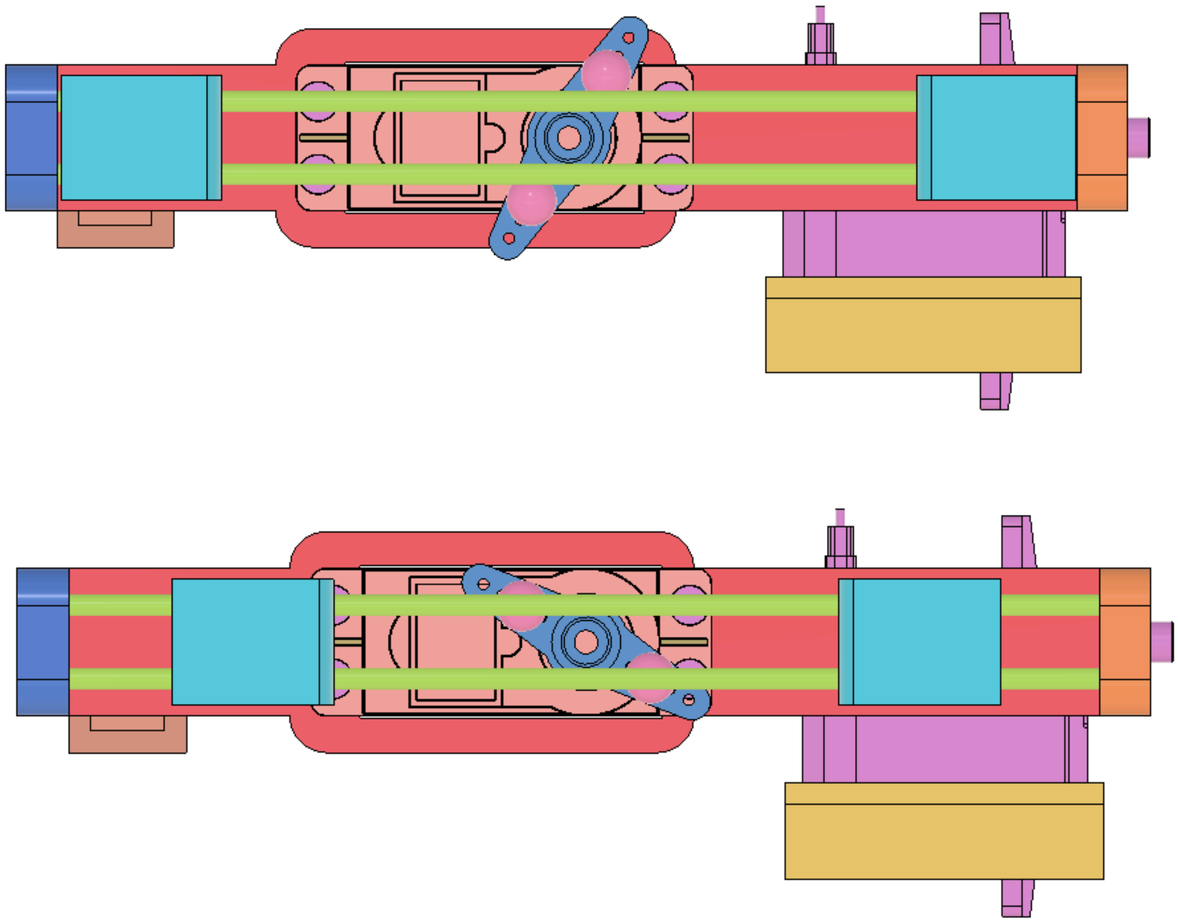


Figure 13. Closing Mechanism Detail

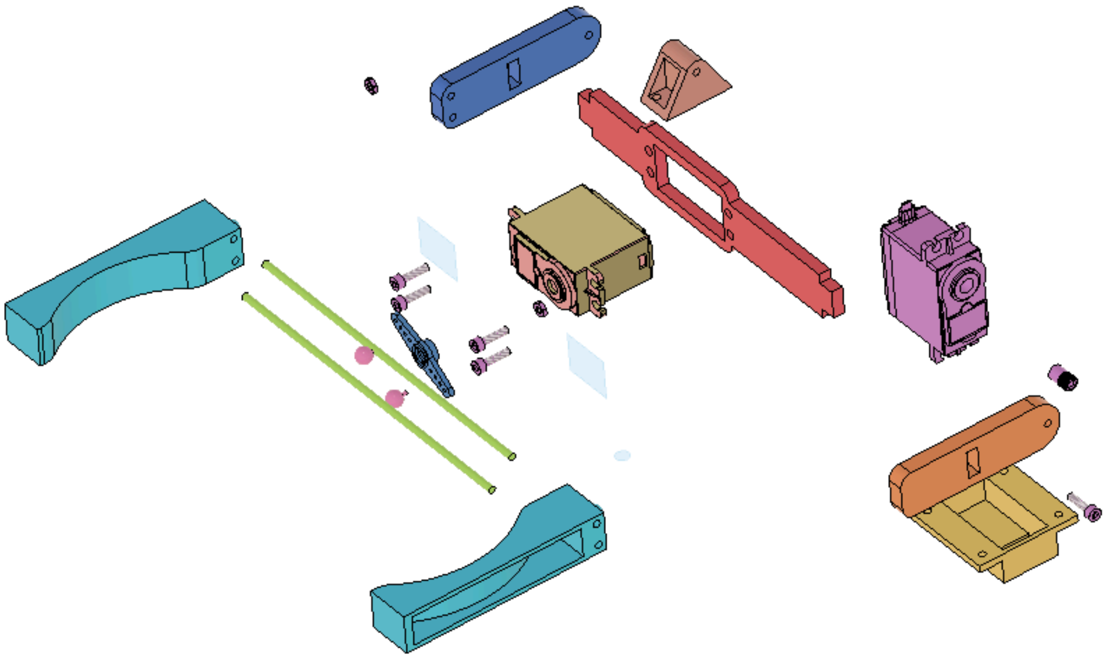


Figure 14. Exploded view of claw

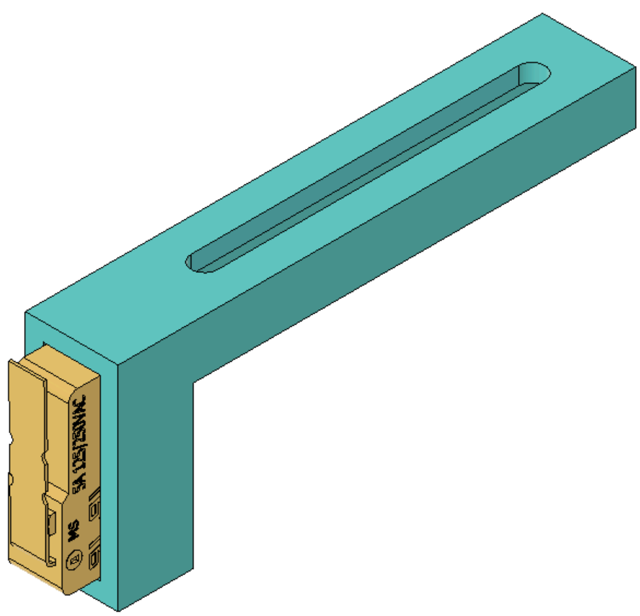


Figure 15. Adjustable limit switch

If during testing we find that the line follower and vision system are insufficient to navigate to the pylon, we will add a pair of “sensor whiskers”, which are essentially button sensors mounted in casings with long rotating arms. The whiskers will be mounted such that the arms run coincident with the edges of the claw opening and diverge to either side, providing a wedge-shaped sensing boundary that we can use to adjust our angle of approach. This additional step is represented in the flow control chart, but may be skipped if found to not be needed. The casings and arms will be 3D printed.

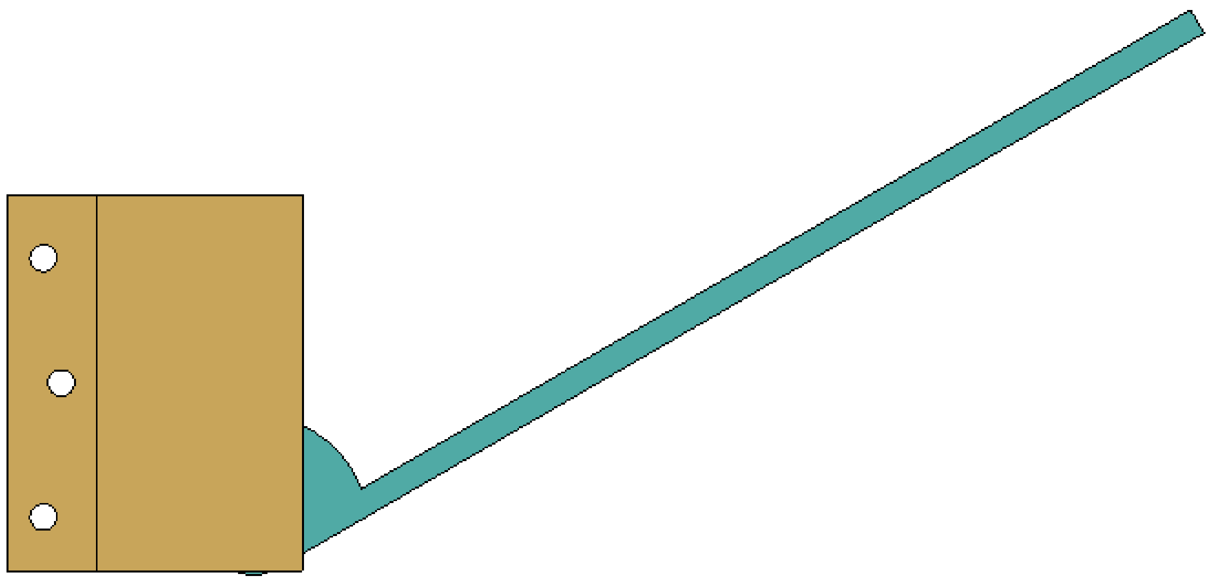


Figure 16. Top view of sensor whisker

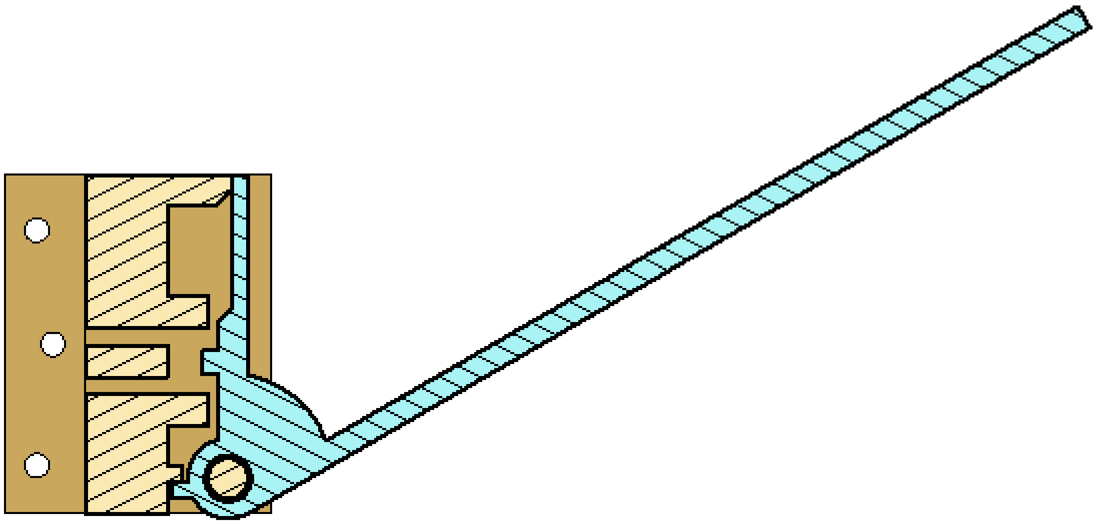


Figure 17. Section view of sensor whisker

All sensors for navigation on the OTV are haptic switches, so the sensor ranges depend on where the sensor is placed on the OTV and the resolution depends on how far the switches need to be pressed to activate. Currently the sensors are placed such that they extend off of the OTV so obstacles can be detected and the switches do not need to be pressed hard in order to be activated. For mission sensors, resolution depends only on the consistency of activation, since the reed switch is a switch and the electrodes will be able to pick up any signal as long as the reed switch is near a magnet and the electrodes are in contact with the pylon. All sensors do not need high amounts of current to operate so they will all be connected to our microcontroller's 5V output. For more details, see the Electrical Design Brief beginning on page 25.

The overall procedure is to first, self-orient using the vision system, determining position and rotating to the correct heading based on which start zone we've detected we are in. Next, proceed to the mission site, using the vision system and line follower in tandem. If it is determined that the whisker sensors are needed, reduce speed when within 5 cm of estimated pylon location and use whisker sensors to adjust heading on approach to pylon. When the adjustable limit switch is triggered, rotate the claw servo to 119 degrees, closing claw. Then rotate the arm drive servo to 180 degrees, mating the pylon with the interface plate. Take sensor readings from the reed switch (switch will output 0v if no magnets present, >0v if magnets are present) and the electrode antenna. Transmit data via WiFi module. The control loop then switches to obstacle navigation.

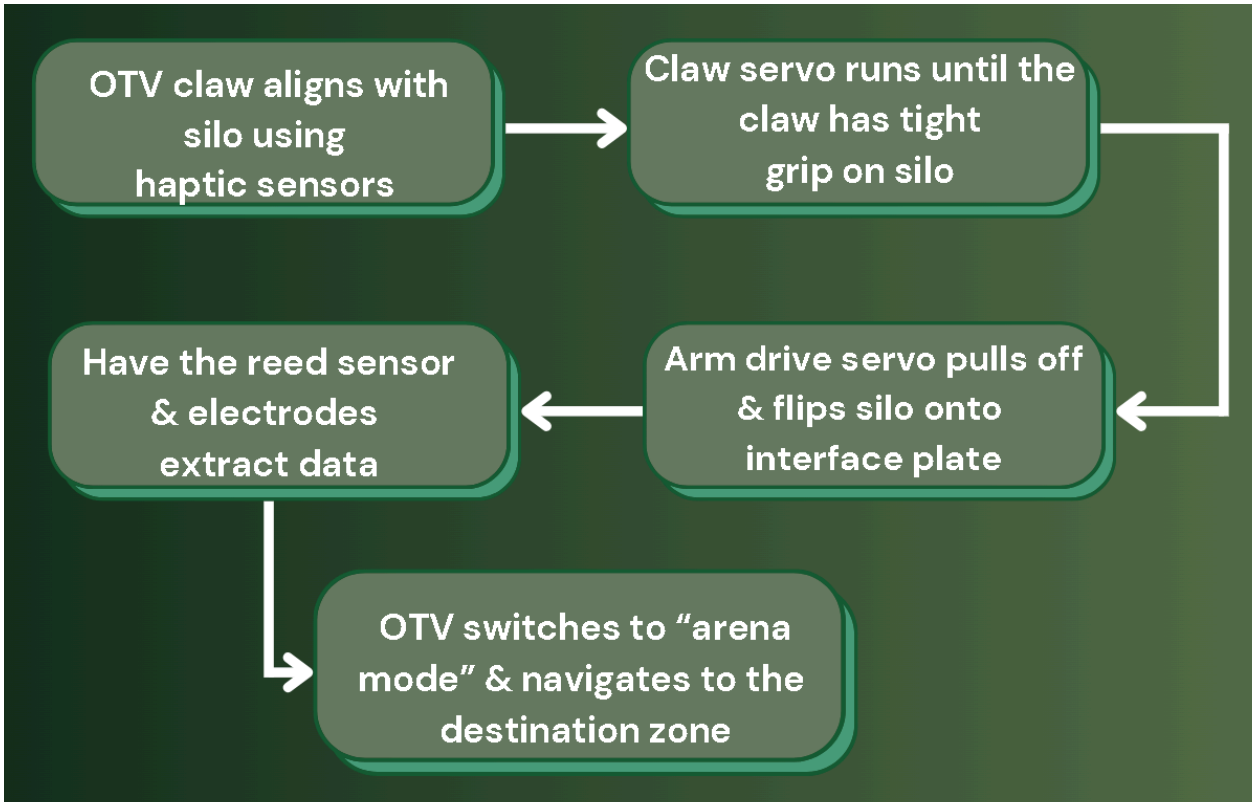


Figure 18. Mission specific control loop

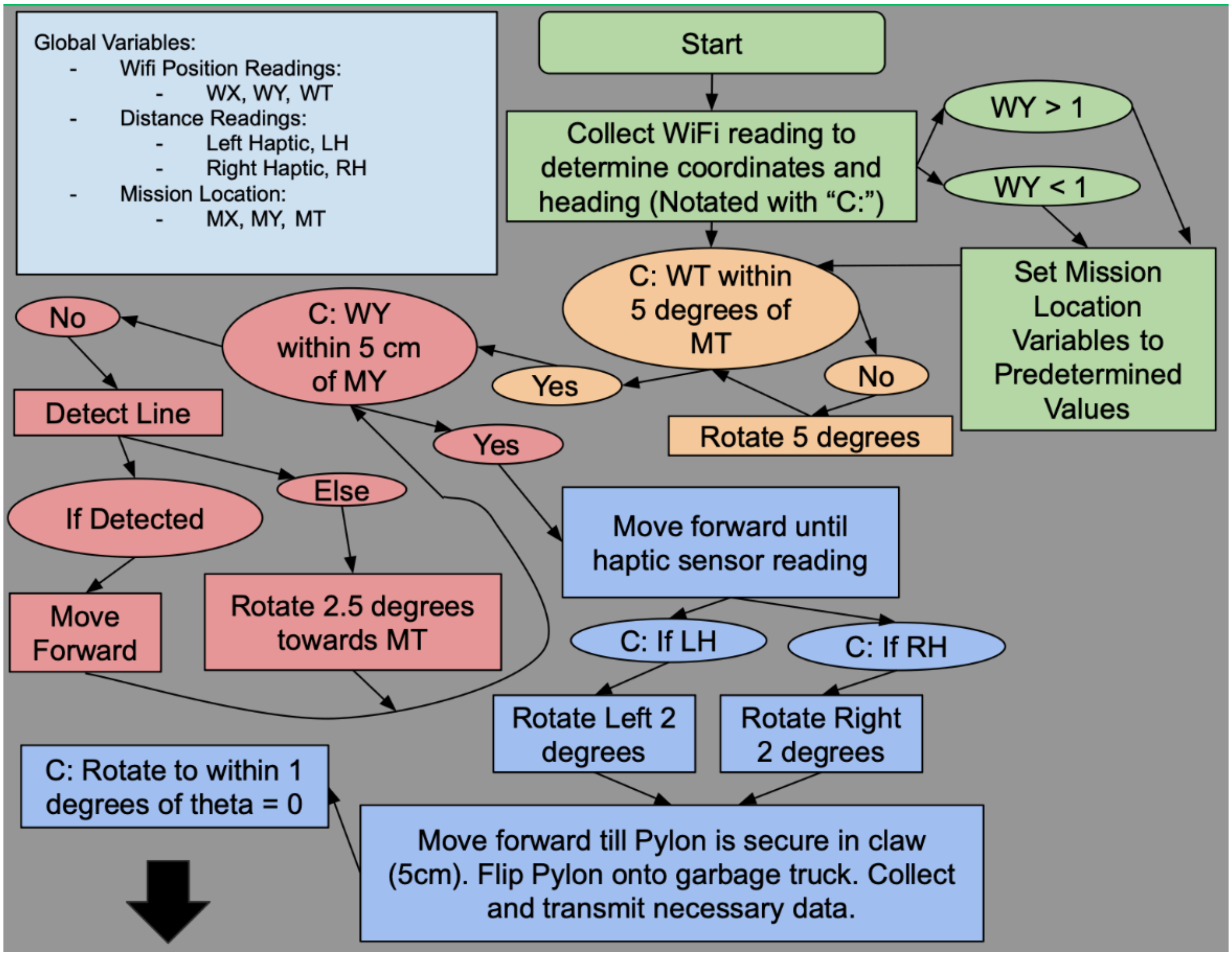


Figure 19. Full Control Loop

Propulsion Design Brief

Our OTV must be able to drive to the mission site, then navigate through the arena and over the log. Our challenges include finding a motor that can handle a minimum of 3kg of weight and a 0.7 coefficient of rolling resistance, and finding treads with a proper coefficient of friction against the arena floor.

We chose to build an OTV with the Tank Drive method of locomotion. Tank Drive will enable the OTV to rotate around its center of mass for more efficient navigation. In addition, the OTV has a tank design by having belted wheels. We chose to use belted wheels for increased distribution of torque onto non-drive wheels to improve speed and for increasing distribution of traction and power of our OTV when climbing over the log. When maneuvering around the obstacle course in the arena, the OTV will be driving backwards, the front being where the mission system components are located. This is a result of attempting to reduce the risk of damaging the mission system components. For the remainder of the propulsion section, the back where the battery is will be considered the front of the OTV.

The OTV will have three wheels on each side. The front pair of wheels will be raised and controlled by two motors. In this location, the front wheels will have greater leverage when going over the log and be directly powered by the motors. The other four wheels are non-drive with rotation being caused by the belts. Should the OTV not make it over the log completely, the belt will ensure that the OTV can continue to move using the front wheels to power the rotation of the back wheels.

The drive wheels have a radius of 40 mm with indents of 0.75 mm every 2 mm. The supporting wheels have a radius of 30 mm with the same indents of 0.75 mm every 2 mm. The choice in radius of the wheels allows the OTV to require less torque for one rotation of the wheels while the indents are based on the tread size of the belts. The wheels have an additional raised wall on the sides of the wheels to prevent the belts from slipping off the wheels. 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 plan to cut and glue multiple pieces of the tread together. The tread size provides traction against the arena floor while preventing slippage between the belt and the wheels during movements that require high torque.

The Greartisan DC 12V 30RPM Gear Motor on Amazon was chosen for the low cost and great adaptability. The provider supplies a variety of motors with different speeds and torque for the same low price. This allowed us to change between motors as our design for our OTV changed during the initial phases and later select a motor based on the final requirements needed.

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 cut outs and nuts. Directly connecting the motor to the front wheels ensures that there is no possible slipping in the connection between the two parts.

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

Motor Details (with supporting calculations) of:

- Torque required by design (FBD, equilibrium equations)

Torque Required: 0.55Nm

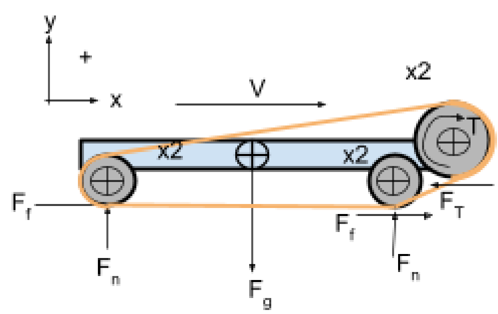


Figure 20. Free body diagram for required torque.

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:

$\tau = F_T \cdot r$

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

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

- Goal for linear speed of OTV:

Our goal for the OTV's average speed is 0.04 m/s. With a minimum speed of 0.027m/s, 0.04m/s as a goal for linear speed will ensure that our OTV completes the mission within the time limit while allowing us to navigate safely.

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

- Motor characteristic graph with operating point (explicitly state the operating torque and operating angular speed)

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 have 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 is 9.55rpm. Combining these two requirements, we selected the Greartisan DC 12V 30RPM Gear Motor.

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

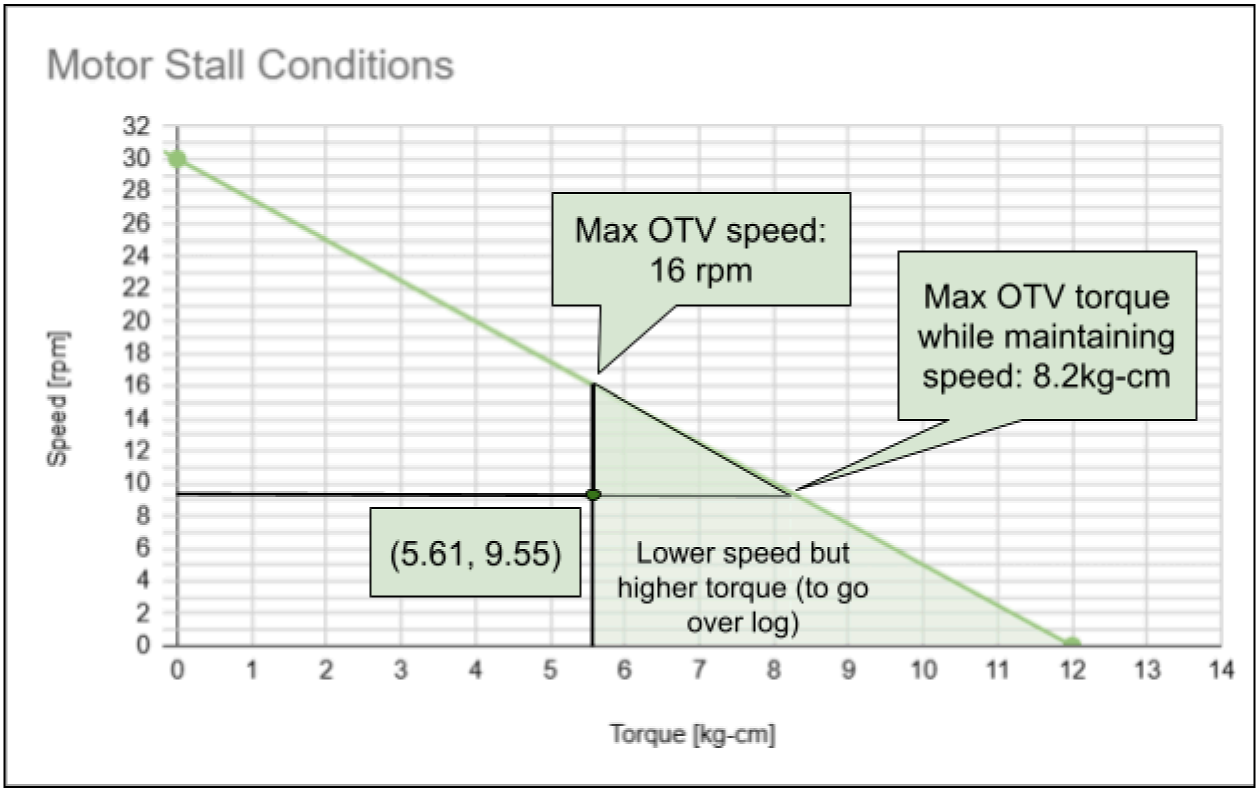


Figure 21. Motor stall conditions graph

- Operating linear speed: 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.

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

$$v = r\omega$$

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

- Operating current: 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.

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

$$x = Required\ current$$

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

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

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

Electronics Design Brief

Our OTV must be able to run autonomously for ten minutes without recharging and have an easily accessible kill switch that will turn the OTV on and off. The battery must be able to power all the components and must not be made of Lithium or Lead Acid. When determining the battery, we have to consider all of the power, current draw, and voltage requirements of all components. Also, we must consider the amount of electrical components we are using and whether they can all be connected to the Arduino microcontroller we plan to use.

We are using a Tenergy 12V NiMH (Nickel Metal Hydride) battery. The battery has a 2000mAh capacity, 2A of max continuous discharge, and a C-rate of 1C. The weight of the battery is 0.255kg. An NiMH battery was chosen because lithium and lead-acid batteries are prohibited. We chose a battery that has the same voltage as the motors and H-bridge to ensure sufficient voltage. NiMH batteries have a relatively stable discharge curve, so their voltage remains close to 12V for most of its cycle before dropping when approximately 90% capacity has been discharged.

We are using an Elegoo Mega R3 microcontroller as the OTV's arduino based microcontroller. The Elegoo Mega operates at 5V, so it will receive power from a regulated step-down from the 12V battery from its onboard voltage regulator. This allows the actuators and sensors to safely operate at 5V without overvoltage risks. We are using the L298N motor driver as our H-bridge. This H-bridge motor driver is rated at the same voltage as the motors to efficiently control the motor direction and speed. We are using one MG996R servo and one FS5115M servo which can both operate with 5V and have stall currents of 1.2A and 3A respectively. The kill switch must cut power to critical systems, mainly the motors and H-bridges that operate at 12V. Placing the kill switch at the main 12V input allows for a complete shutdown of propulsion in case of emergency. Additionally, the kill switch will be located on the side of the OTV near the battery, making it easily accessible.

For current draw calculations, we will assume the worst case scenario for high current draw components, including the drive motors and servos. Since we are using the L298N motor driver, the max current the motors can receive is 2A before the driver begins to break down. As stated before, the servos have stall currents of 1.2A and 3A respectively which will be used since we are able to change the voltage source of the servos if needed. However, since the Elegoo can only output a certain amount of current, testing will be done in the future to determine if the Elegoo Mega can sufficiently power the servos or if we need to switch voltage sources. The current draw for sensors is assumed to be a non-factor since most of them are switches and we are already assuming perpetual stall conditions. The current draw for the rest of the components were also estimated using their rated or operational current values.

Components	Voltage [V]	Current [A]	Power [W]
Drive Motors (x2)	12	4	48
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 Switch	5	0	0
Haptic Switches (x4)	5	0	0
Line Sensor	5	0	0
Total	12	9.3	78.7

Figure 22. 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 that requirement.

Our strategy for handling power is to use pulse width modulation. By using pulse width modulation, we will 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 will utilize the L298N H-Bridge motor drivers to simplify our electrical wiring, improve power amplification, and ease the coding of motors. Furthermore, all other electrical components will run through the Elegoo Mega to regulate power to those components.

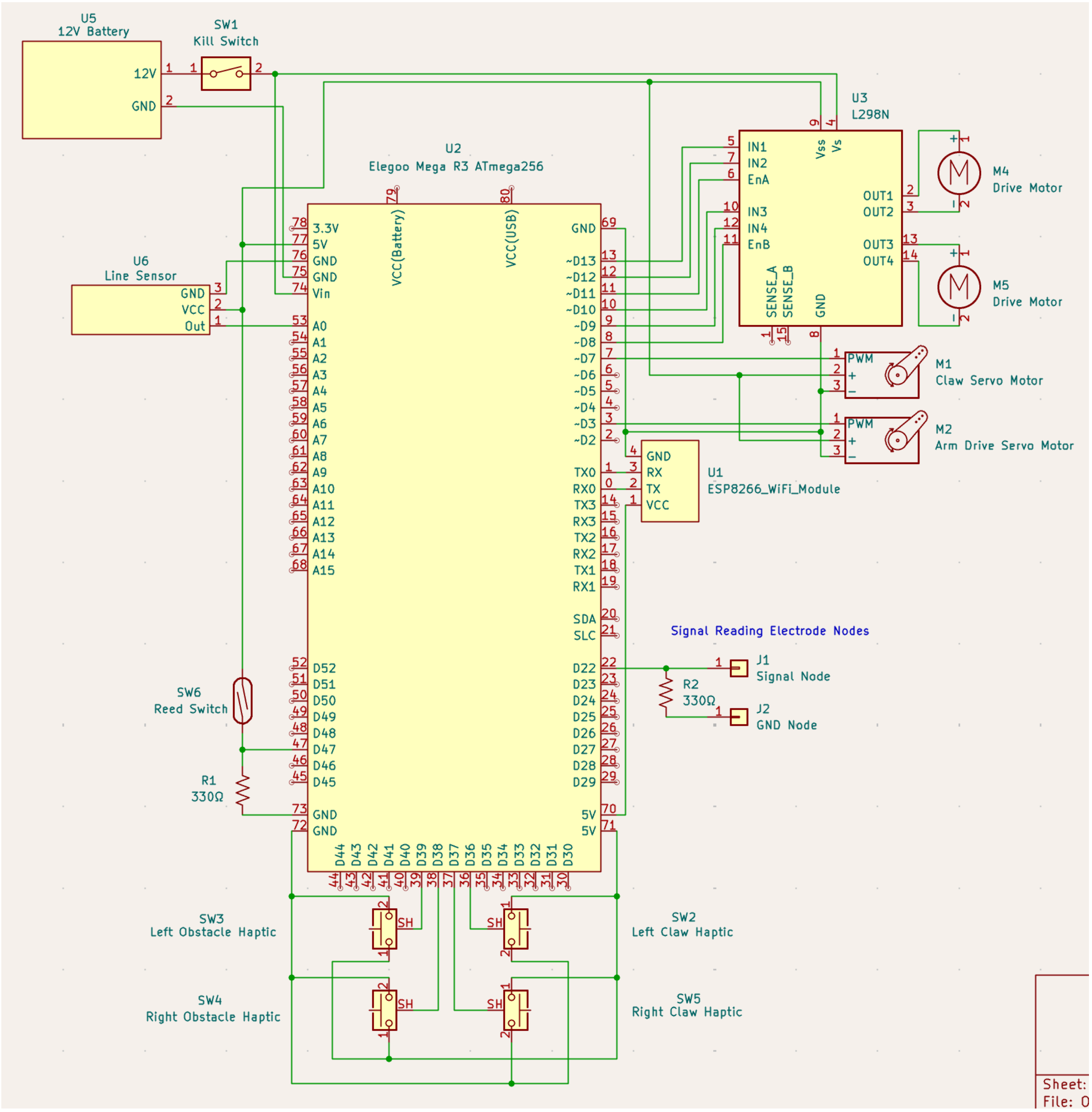


Figure 23. Circuit schematic diagram

Component	Type	Arduino Pin(s)
L298 Motor Driver (Drive)	Motor Controller	~D8, ~D11, D9, D10, D12, D13
Claw Servo Motor	Actuator	~D7
Arm Drive Servo Motor	Actuator	~D3
ESP8266 WiFi Module	Communication	TX0, RX0
Line Sensor	Sensor	A0
Reed Switch	Sensor	D47
Signal Reading Antenna	Sensor	D22
Left Obstacle Haptic Switch	Sensor	D39
Right Obstacle Haptic Switch	Sensor	D38
Left Claw Haptic Switch	Sensor	D36
Right Claw Haptic Switch	Sensor	D37
Power Pins	Power	Vin, 5V, GND

Figure 24. Arduino pin assignment chart

As seen in both the circuit diagram and the arduino pin chart, the Elegoo Mega has sufficient pins to support the current sensors attached to the OTV along with plenty of extra pins for potential future use. There is an argument that we do not need to use a mega sized microcontroller, however, we believe that this upgrade is appropriate. The Elegoo Mega is priced at $22.99 while the Elegoo Uno is priced at $14.99, so we will only be paying about $8 more. According to the budget, shown as Figure 46 , we have plenty of room for this extra spending. Also, we have heard from various sources that using a mega sized microcontroller is much easier to work with than an uno sized microcontroller due to the flexibility, space, and volume of pins on the microcontroller. Plus, the Elegoo Mega is recommended for 12V while the Elegoo Uno is recommended for under 9V, and if we are using a 12V battery, it makes sense to go with the Elegoo Mega so our microcontroller will work properly. If by the end of production we need to downgrade due to cost and we are not using enough pins to justify the mega sized microcontroller, we will consider downgrading to Elegoo Uno, however, we see no other reason to go with anything other than a mega sized microcontroller.

In the circuit diagram, all sensors and servos are connected to the 5V power supply from the Elegoo Mega while the L298 and arduino are directly connected to the battery. Also, the kill switch is connected directly to the battery's positive output terminal to ensure a complete disconnection of power from the battery upon activation. The arduino pin chart outlines what pins each component will use, with specific use cases for each depending on the component. Specifically, digital pins in the chart with a "~" will specifically be used for the pulse width modulation output functionality while the others will just be used for digital I/O functionality, even if they are capable of pulse width modulation output functionality. Specific pin assignments are subject to change depending on the production of the OTV, but functionalities must stay the same. For specific component details, see the budget, Figure 46.

Compiled Set of Engineering Drawings

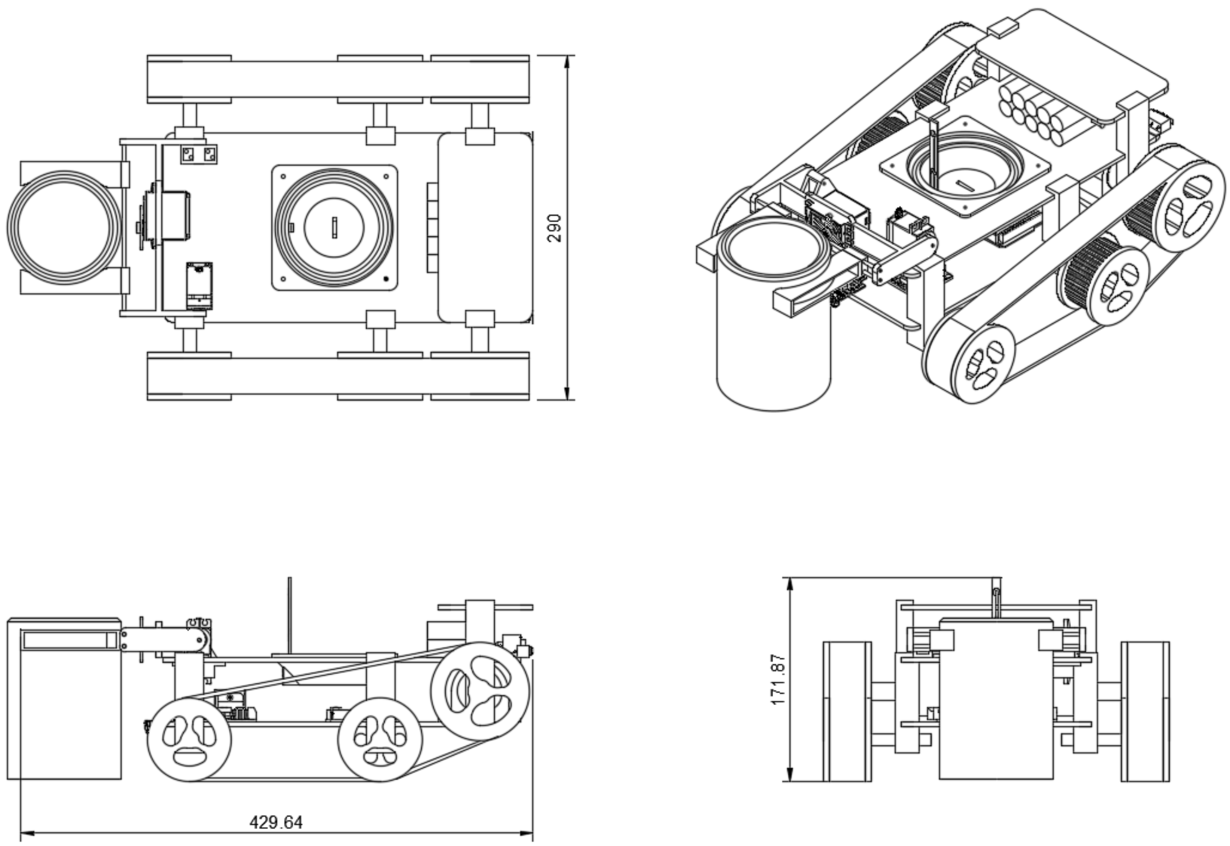
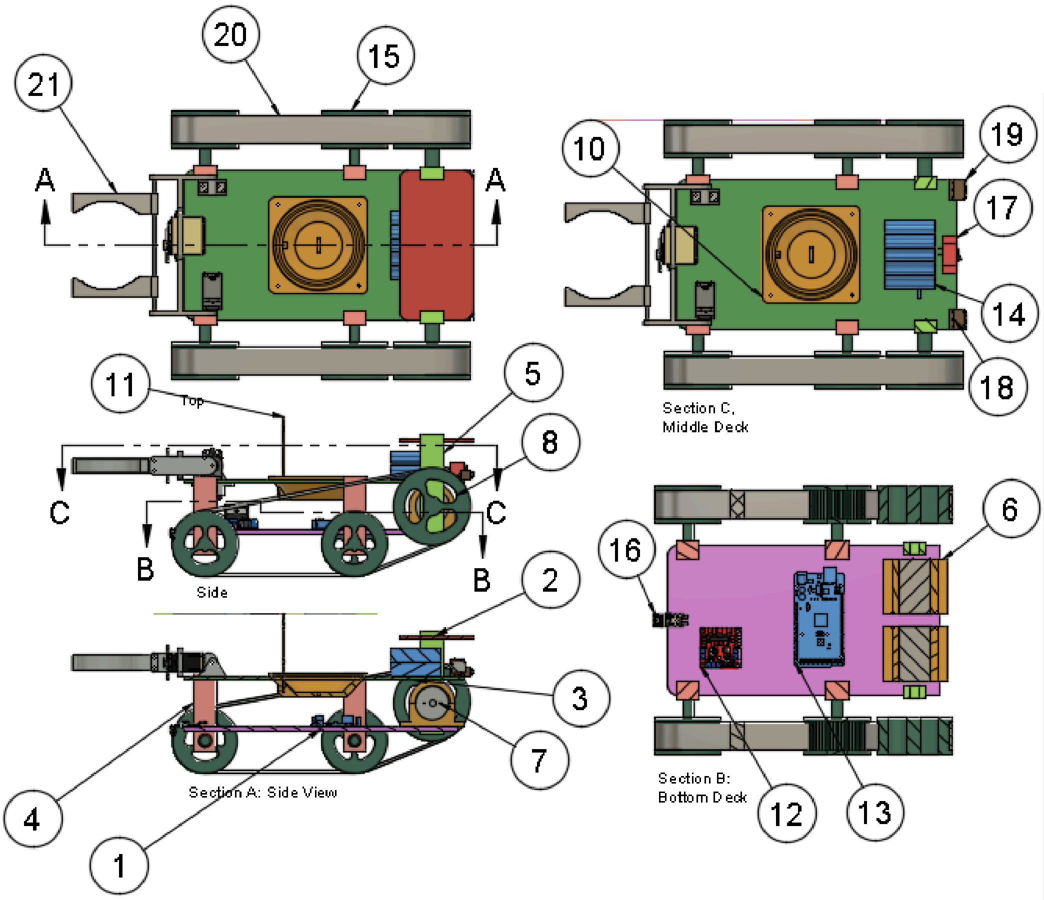


Figure 25. OTV assembly with silo in claw (unit: mm)



ITEM	QTY	PART NUMBER	MATERIAL
21	1	2SERVOCLAW	
20	2	BELT	STEEL
19	1	P:\E-SWITCH SECURE3D_MODELS_STP_FORMAT\MS085XX02FXXXP1A.STP	STEEL
18	1	P:\E-SWITCH SECURE3D_MODELS_STP_FORMAT\MS085XX02FXXXP1A.STP	STEEL
17	1	RA1113112R	STEEL
16	1	TCRT5000L MODULE	
15	4	SUPPORT WHEEL	STEEL
14	1	BATTERY	STEEL
13	1	ARDUINOMEGA_STEP_AP203	STEEL
12	1	L298N DRIVER DC MOTOR, STEPPER MOTOR	STEEL
11	1	CARGOINTERFACINGPLATE_ANTENNA	STEEL
10	1	CARGOINTERFACINGPLATE_PLATE	STEEL
8	2	DRIVE WHEEL	STEEL
7	2	FINALIZED MOTOR	STEEL
6	2	MOTOR_MOUNT	STEEL
5	2	DOUBLEHEIGHTSPACER	STEEL
4	4	MS2SPACER	STEEL
3	1	MIDDLEDECK	STEEL
2	1	TOPDECK	STEEL
1	1	BOTTOMDECK	STEEL

PARTS LIST

Figure 26. OTV assembly with component labels (unit mm)

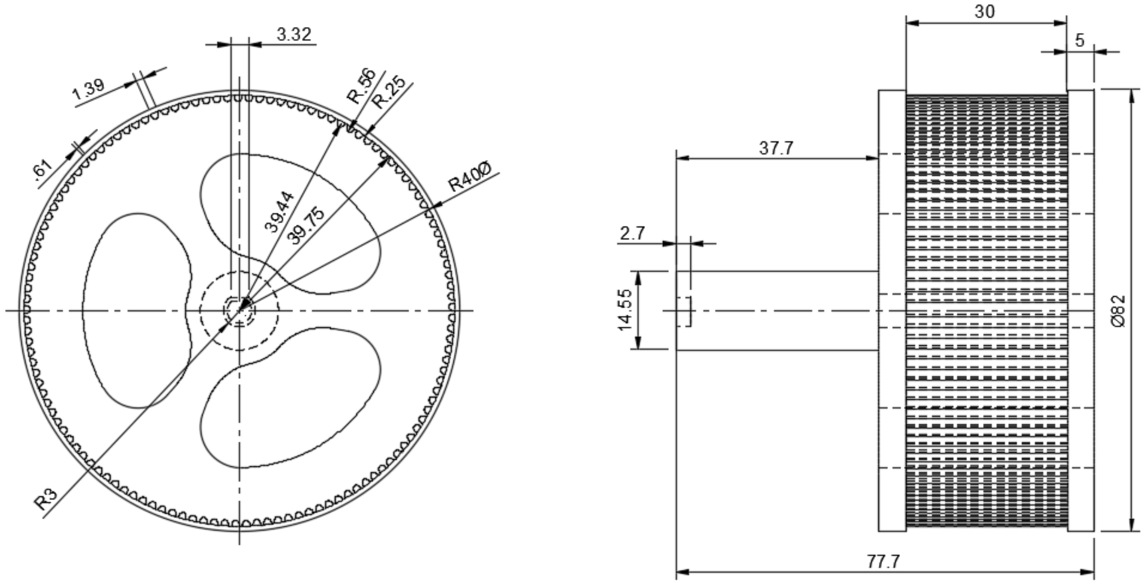


Figure 27. Drive wheel dimensions (unit mm)

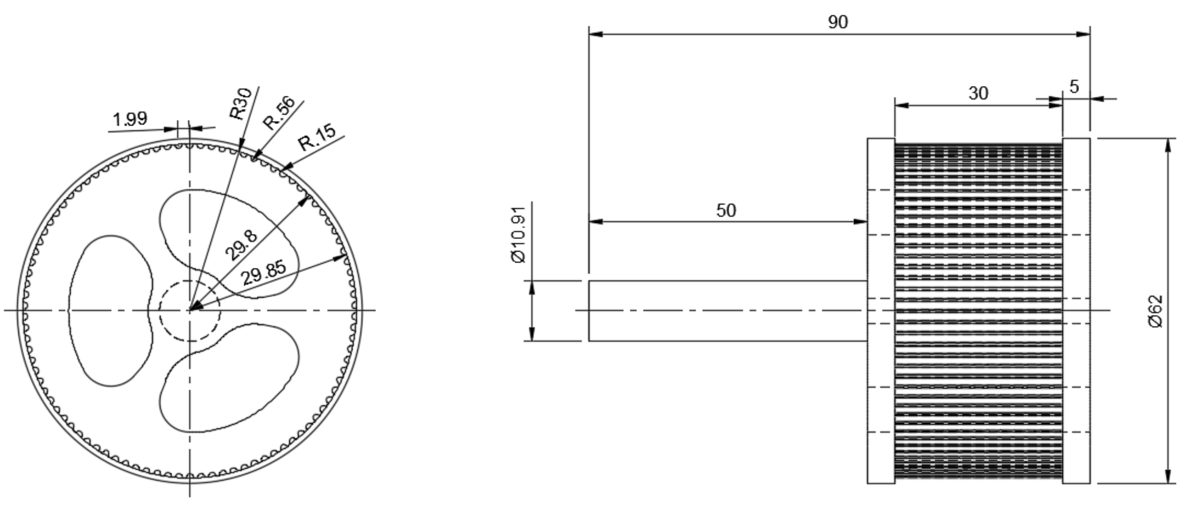


Figure 28. Support wheel dimensions (unit mm)

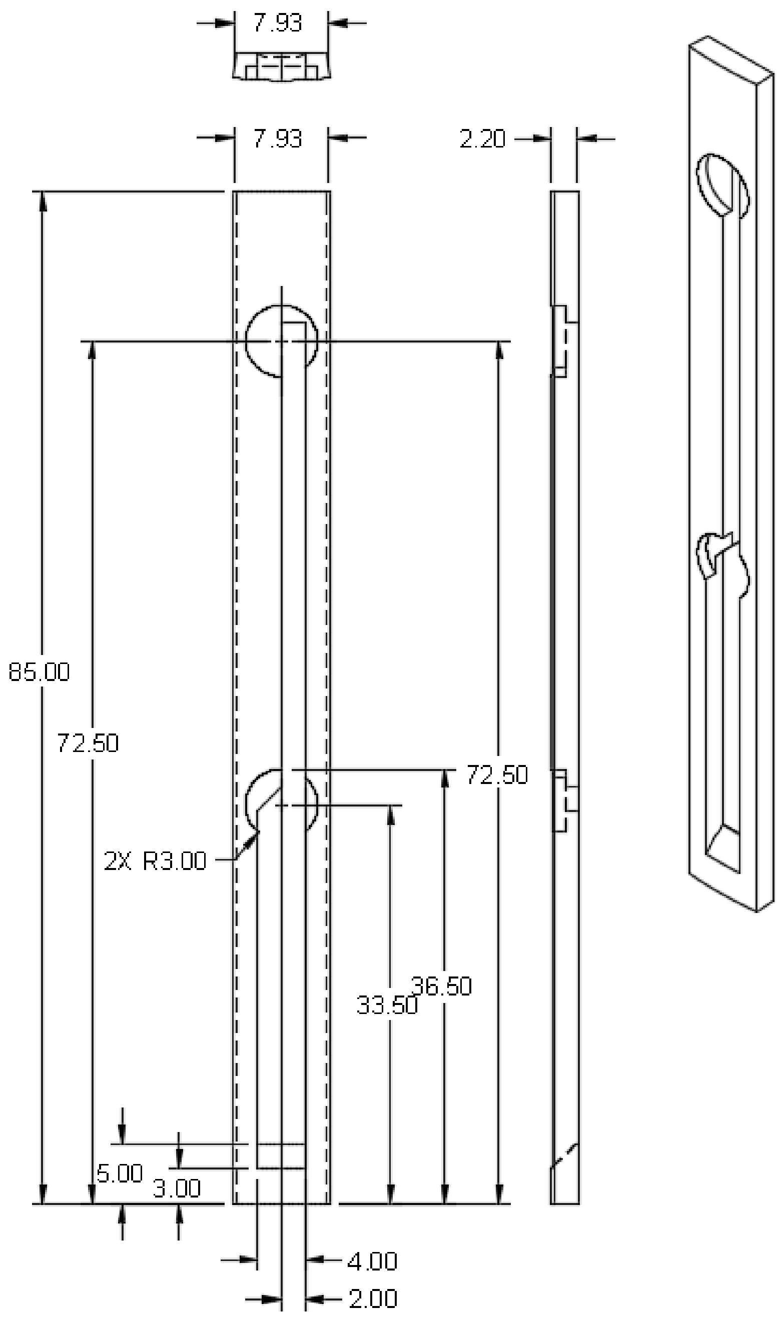


Figure 29. Cargo interfacing plate antenna (unit mm)

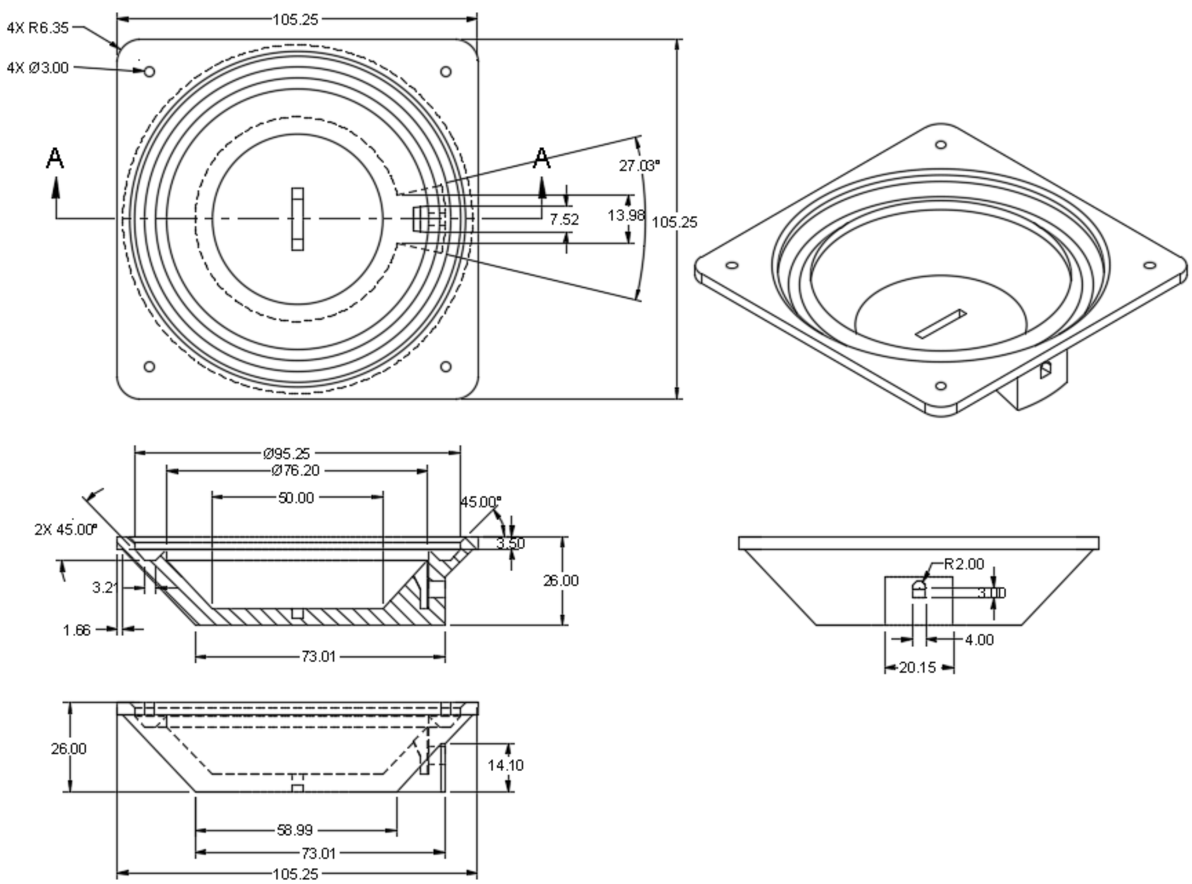


Figure 30. Cargo interfacing plate (unit mm)

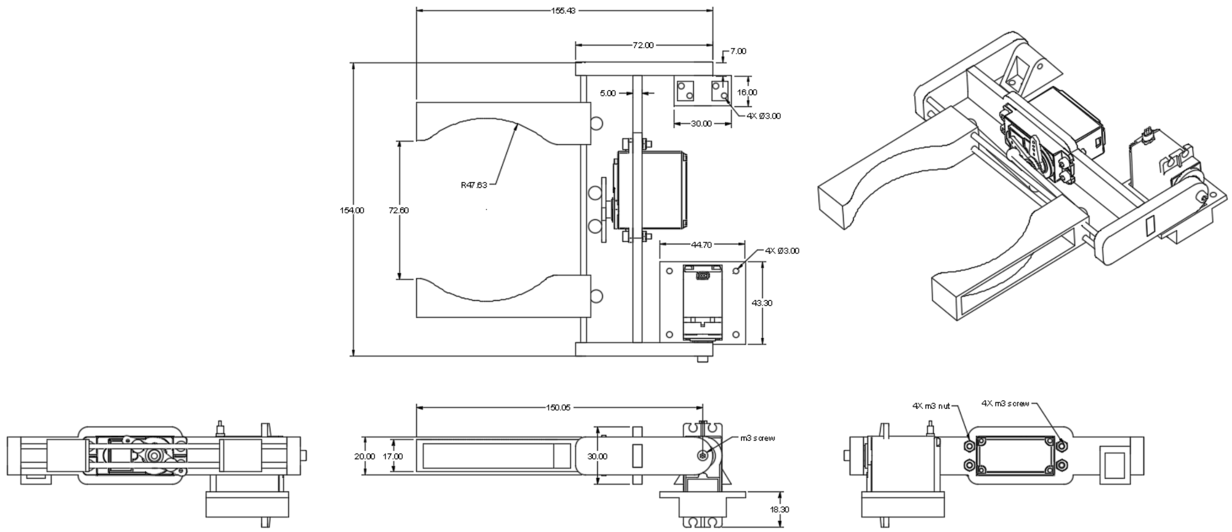
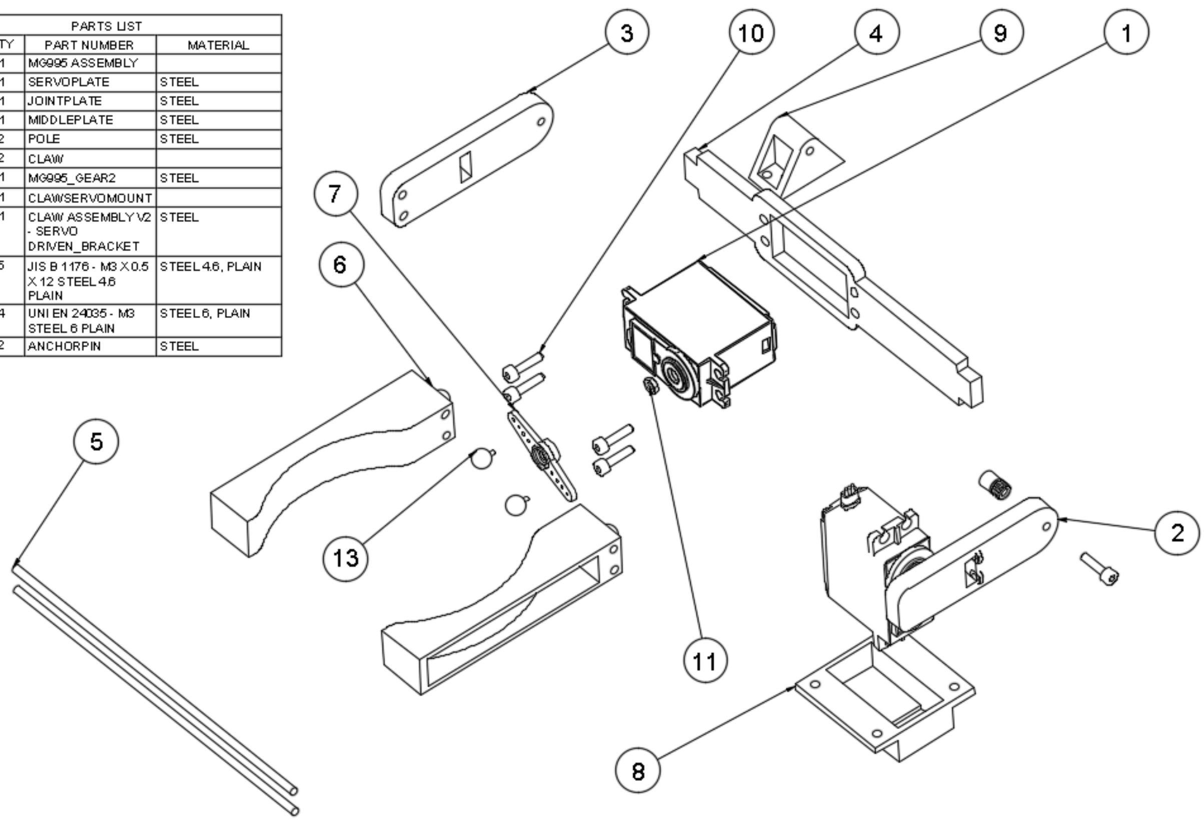


Figure 31. Claw assembly with servo motors (unit mm)



PARTS LIST			
ITEM	QTY	PART NUMBER	MATERIAL
1	1	MG995 ASSEMBLY	
2	1	SERVOPLATE	STEEL
3	1	JOINTPLATE	STEEL
4	1	MIDDLEPLATE	STEEL
5	2	POLE	STEEL
6	2	CLAW	
7	1	MG995_GEAR2	STEEL
8	1	CLAWSERVOMOUNT	
9	1	CLAW ASSEMBLY V2 - SERVO DRIVEN_BRACKET	STEEL
10	5	JIS B 1176 - M3 X 0.5 X 12 STEEL 4.6 PLAIN	STEEL 4.6, PLAIN
11	4	UNI EN 24035 - M3 STEEL 6 PLAIN	STEEL 6, PLAIN
13	2	ANCHORPIN	STEEL

Figure 32. Exploded claw assembly (unit mm)

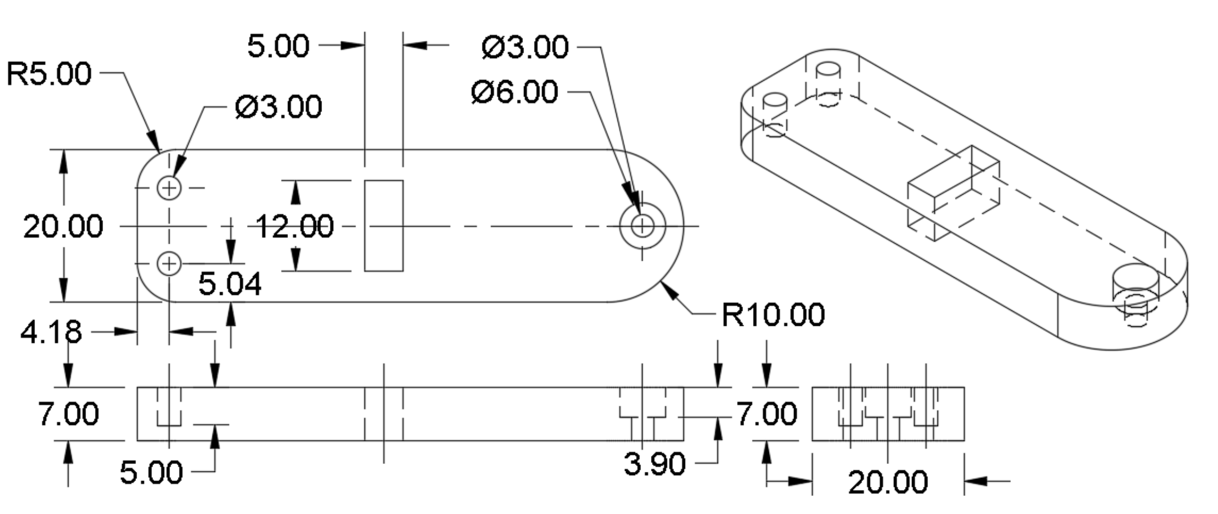


Figure 33. Servo Plate (unit mm)

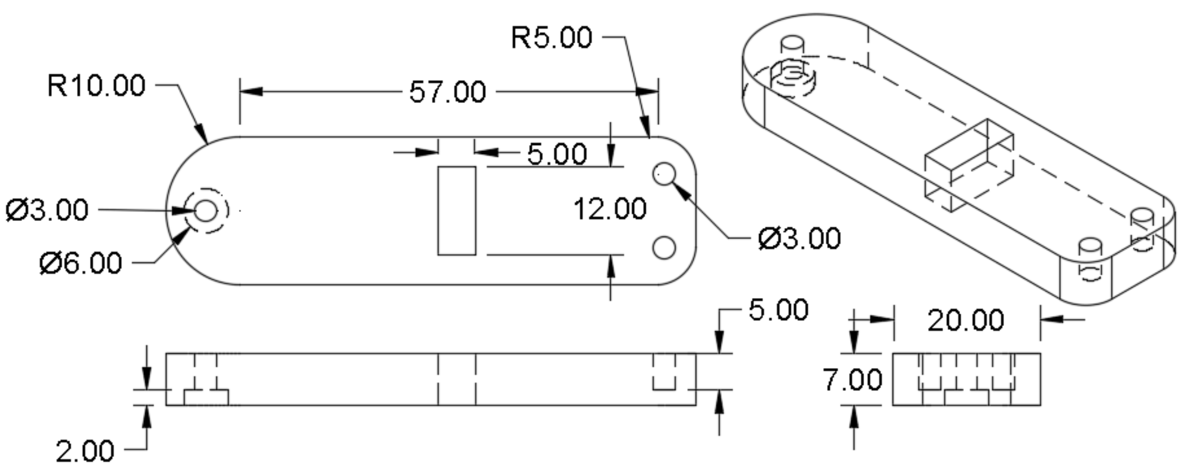


Figure 34. Joint Plate (unit mm)

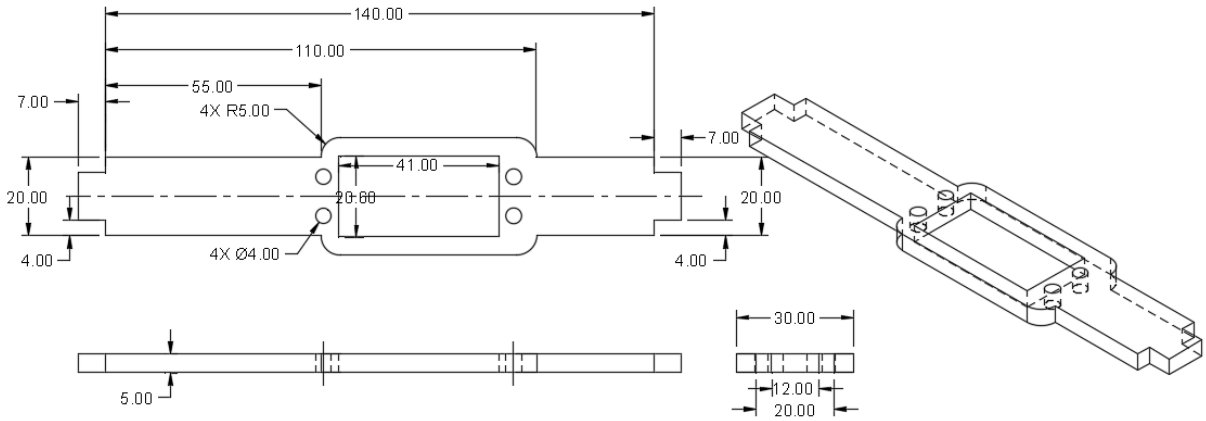


Figure 35. Middle Plate (unit mm)

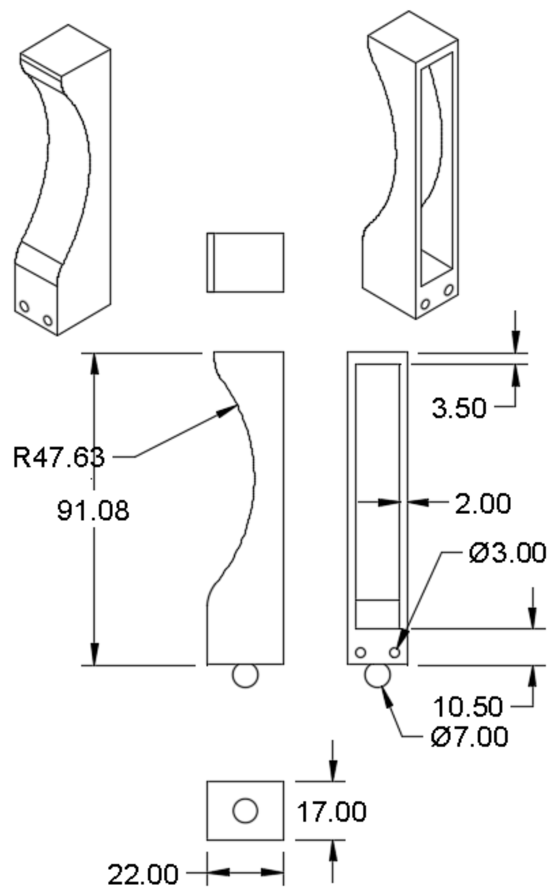


Figure 36. Jaw portion of claw (unit mm)

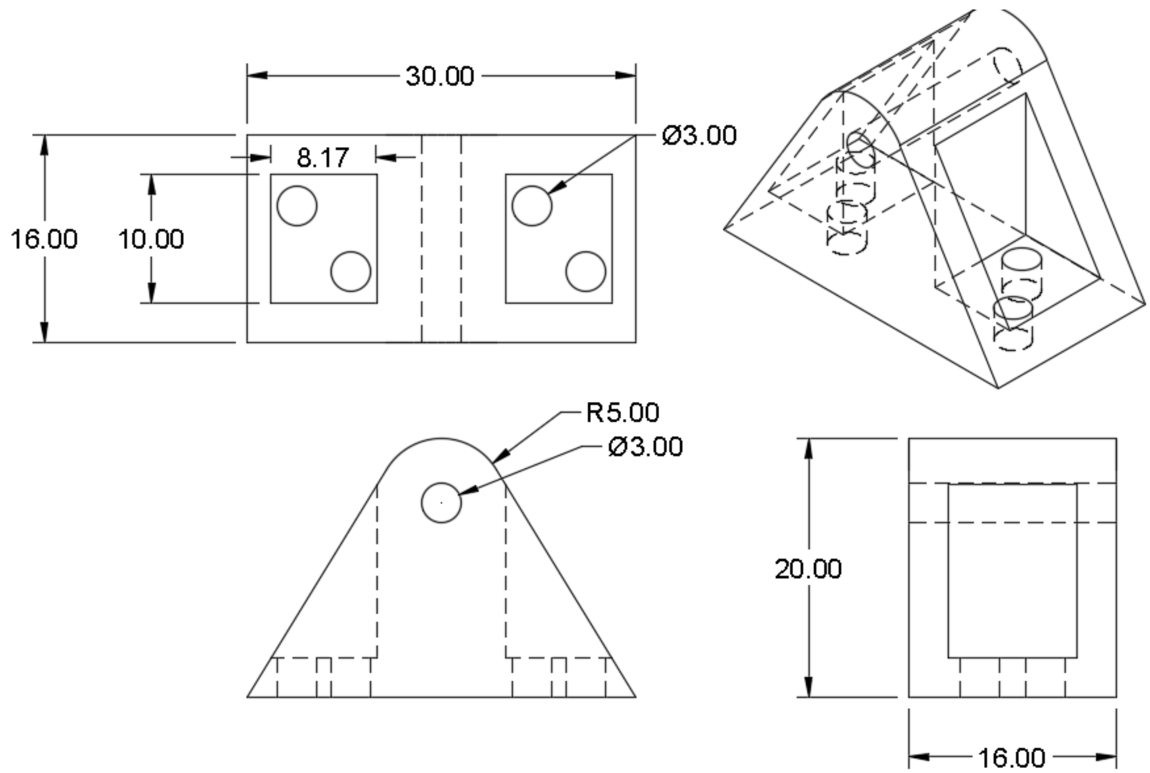


Figure 37. Joint Bracket (unit mm)

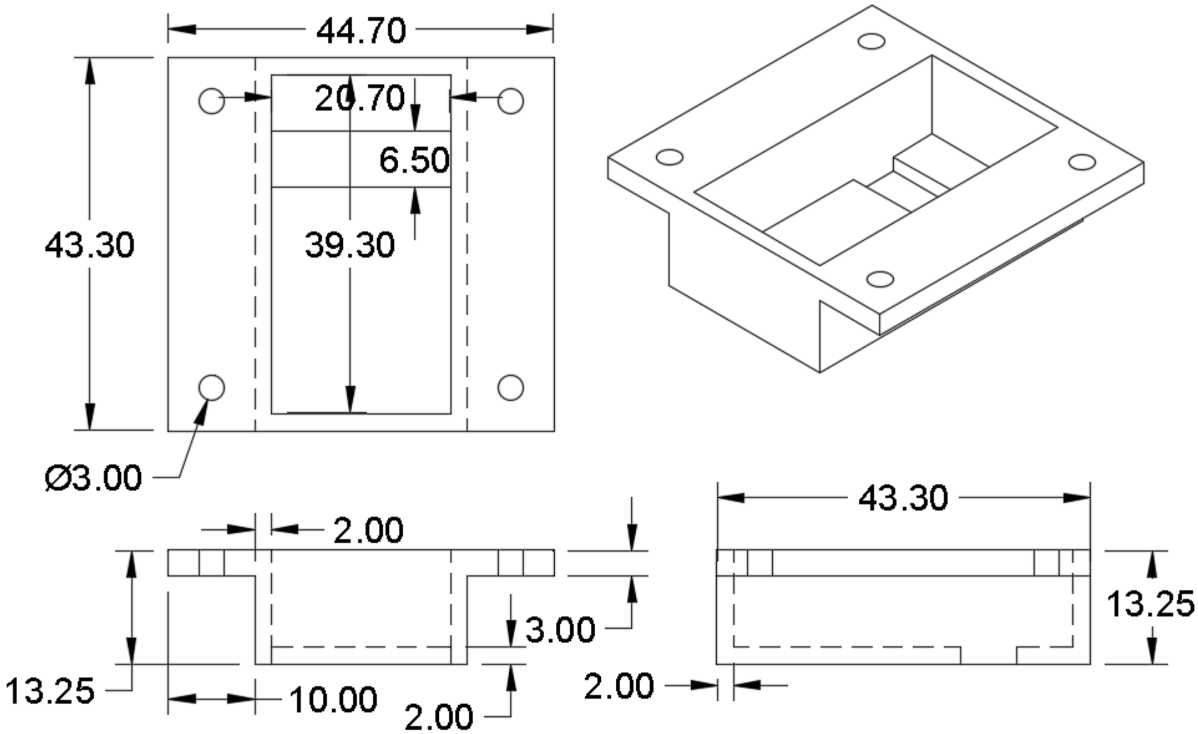


Figure 38. Servo Mount (unit mm)

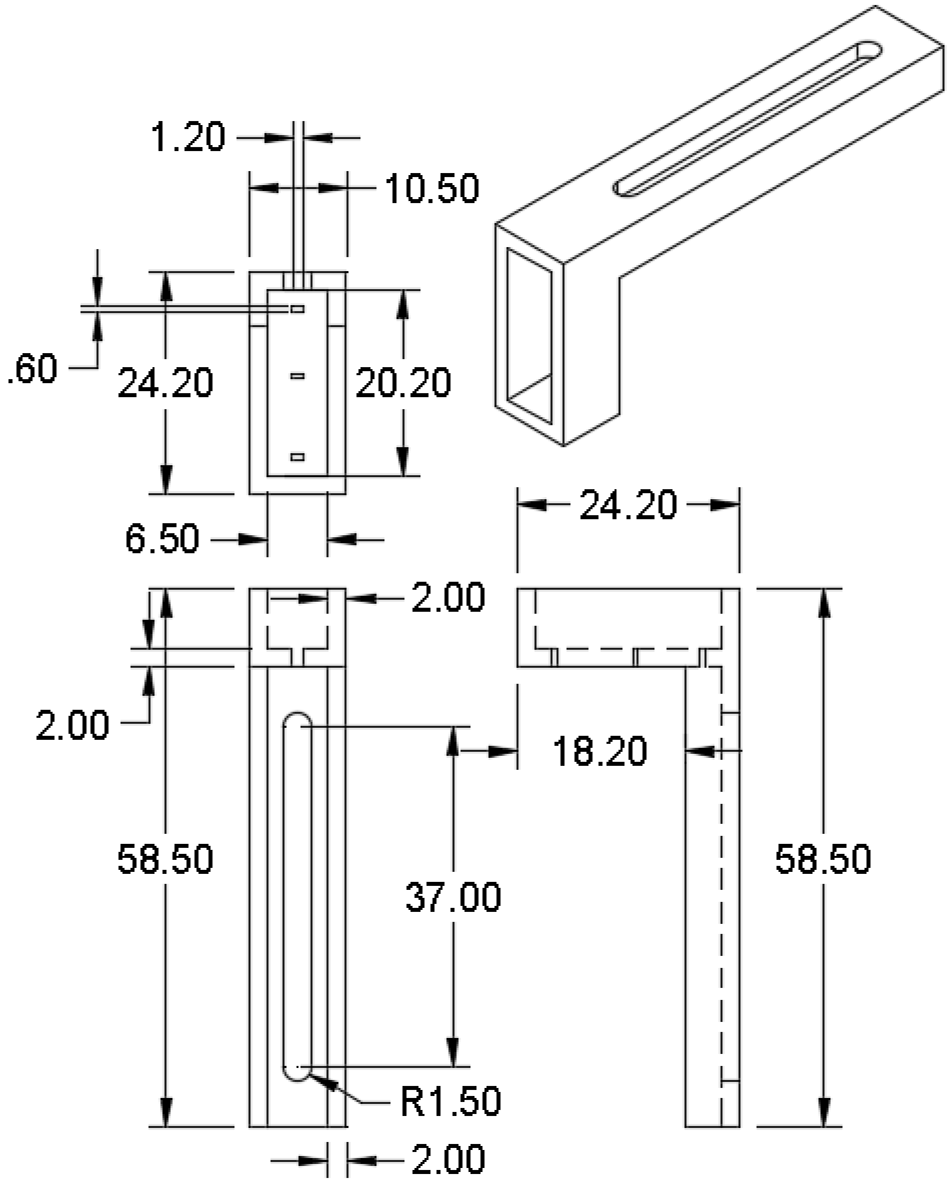


Figure 39. Adjustable light switch mount (unit mm)

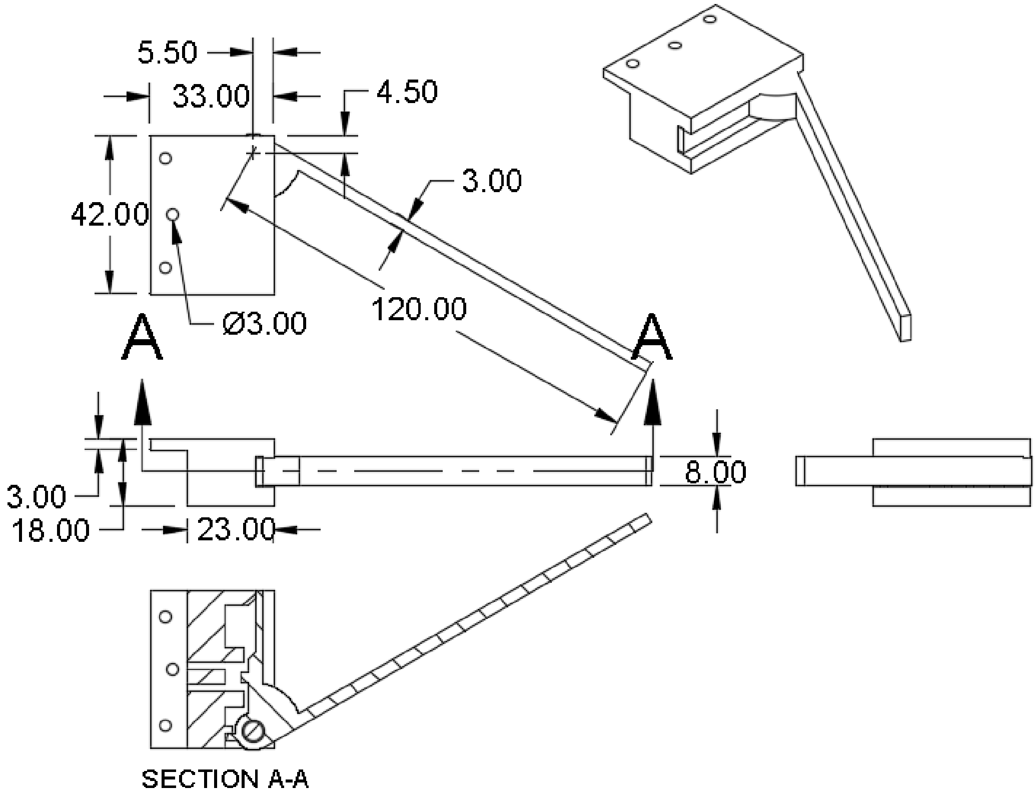


Figure 40. Whisker Sensor (unit mm)

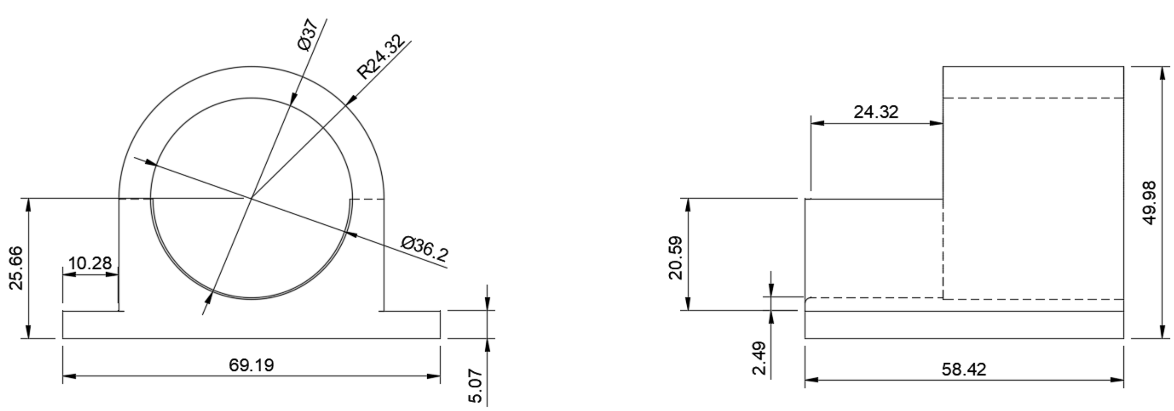


Figure 41. Motor Mount (unit mm)

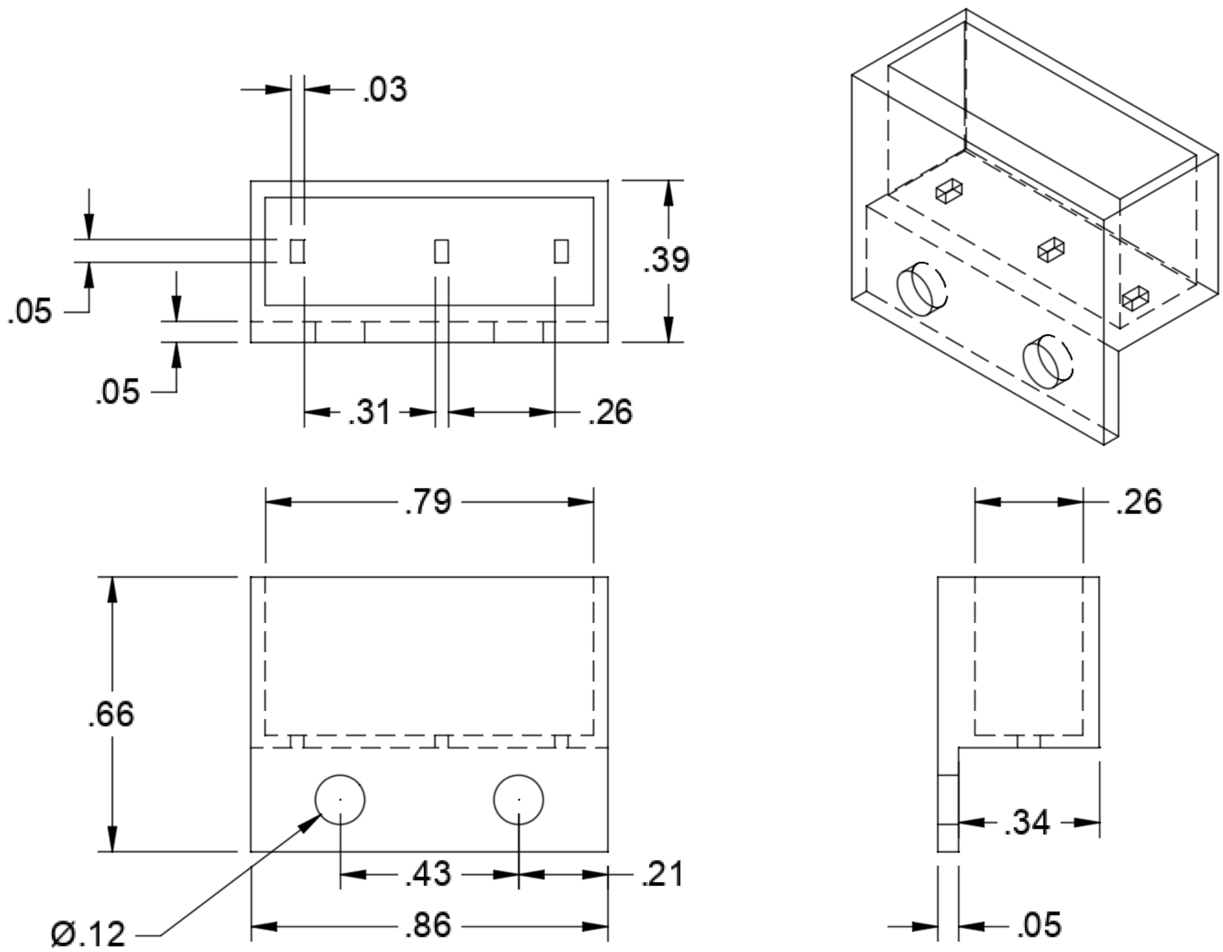


Figure 42. Left Snap-Action Switch Mount (unit in)

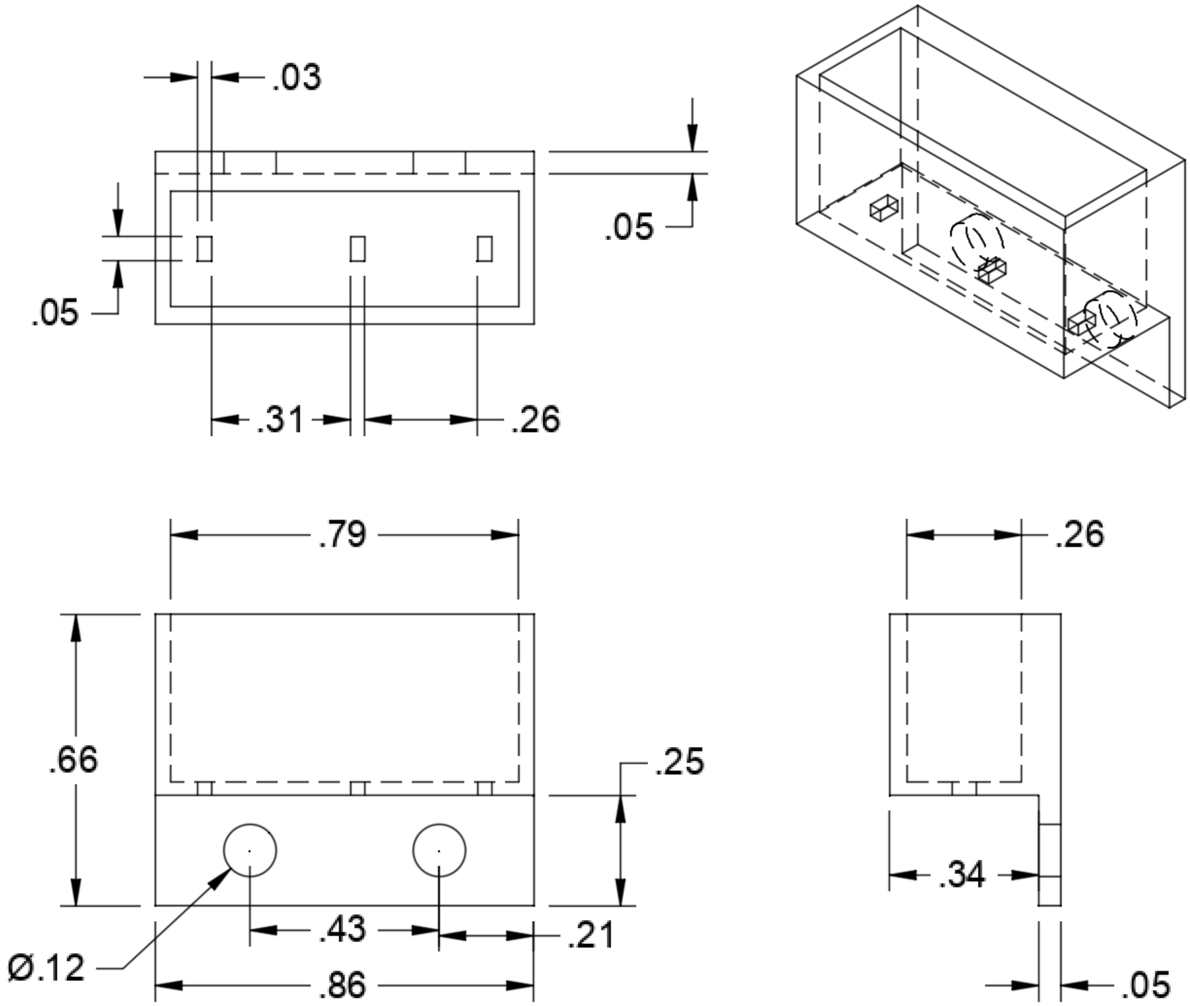


Figure 43. Right Snap-Action Switch Mount (unit in)

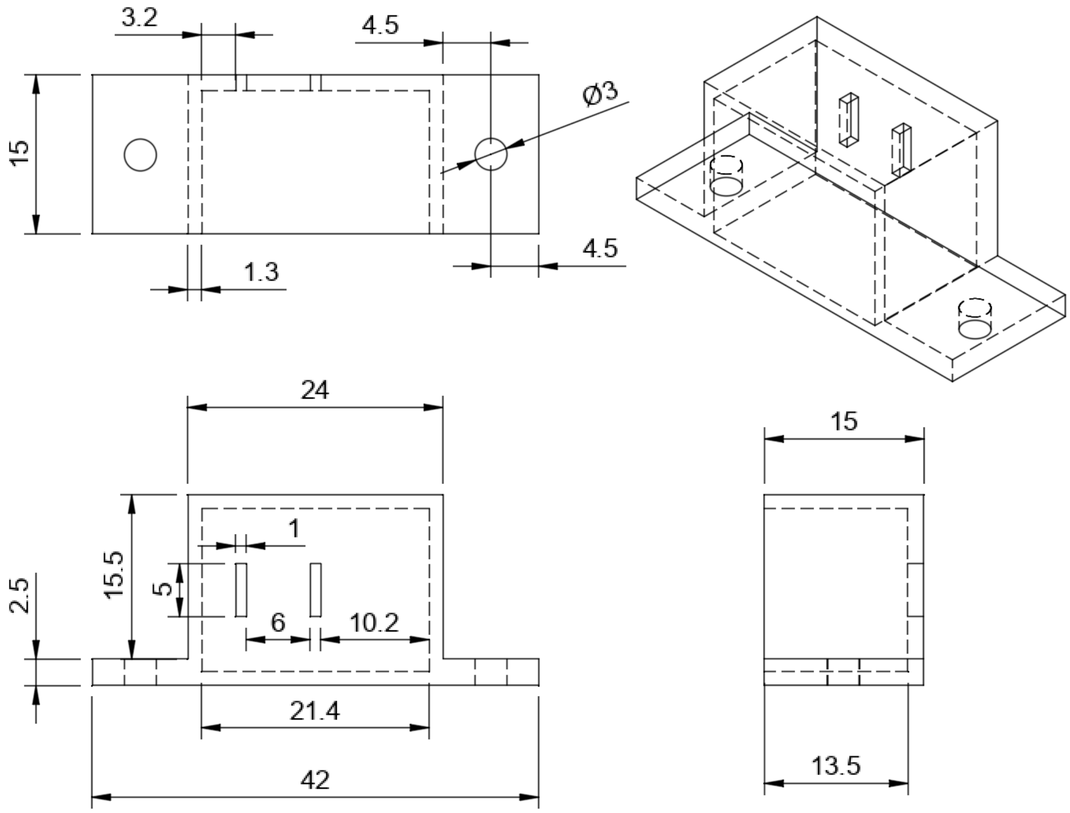


Figure 44. Kill Switch Mount (unit mm)

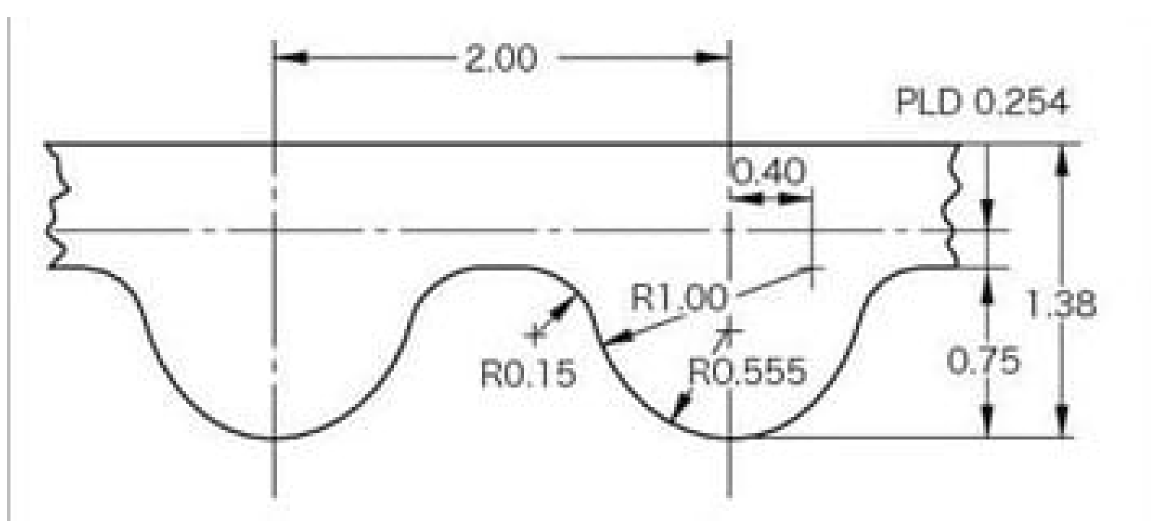


Figure 45. Belt Tread (unit mm)

Testing Plans

Since our OTV design is layered, our overall testing plan is to build the OTV from bottom to top, testing what we can during the assembly process. Specifically for MS5, our goal will be to complete all of the tasks for locomotion before attempting the mission tasks with the OTV. Most components for navigation are located on the bottom layer, so when production is complete on the bottom layer, the electrical and arduino subteams will be able to test the OTV on locomotion and navigation. Additionally, throughout mission system production, the mission subteam will conduct tests to determine if specific functionalities operate effectively.

This method allows us to test our OTV early and often throughout the production process and gives every subteam tasks throughout the build phase such that one or more subteams do not have a high dependence on another subteam to do their work. This can happen specifically with arduino and electrical subteams waiting on mechanical subteams for a built OTV to test, which we are trying to avoid.

For each test outlined in MS5 and beyond, a specific program will be made using functions created by the arduino subteam that will be used throughout tests and eventually in the final program. Ideally, once OTV production is complete, we will know most tests will run for propulsion and navigation, but will be tested again for confirmation and credit on MS5. Additionally, we will have a better understanding of what variables change in our code as production progresses, making it easier to debug any code we create. Below are specific testing plans for propulsion, navigation, electrical, and mission systems.

Propulsion and Navigation Systems Testing:

Certain tests for MS5 that are required for propulsion can be completed before production, but propulsion and navigation testing with the OTV will begin once the bottom layer of the OTV has been developed, which includes the chassis and connected components. Components on other layers necessary for navigation, mainly the Aruco marker, will be temporarily placed on the bottom layer of the OTV as close to the (x,y) coordinates where they will be placed on the final OTV to ensure accuracy. The following list includes tests from MS5 and group defined tests that will need to be completed in order for the OTV to properly move and navigate throughout the course. Each test is listed in the order we plan to complete each task with the physical OTV, however, we will have to repeat each test throughout the production process. Additionally, we may complete these tasks out of order depending on whether the test requires that we need the OTV, but again, we will complete these tests with the OTV once we have sufficient production complete. Descriptions of pass/fail requirements along with testing procedures will be described below each test. Each test is a prerequisite test for the following tests and test cases are defined in each definition of the failure of a test.

1. Your team should demonstrate that your Arduino is capable of receiving coordinate and heading information of your vehicle using wireless transmissions.
    - We will connect our ESP8266 WiFi Module to our arduino, place our Aruco marker in the obstacle course, and attempt to receive information from the overhead vision system. As production progresses, we will attach the electrical components and Aruco marker to our OTV and test this functionality again.
    - Pass: We are able to see heading information in the serial monitor within the Arduino IDE.
    - Fail: We are not able to see heading information in the serial monitor within the Arduino IDE.
2. Your team should demonstrate that your Arduino can send mission messages via wireless communication using appropriate code/functions from enes100.umd.edu and view these mission messages at the main computer of the overhead vision system.
    - We will connect our ESP8266 WiFi Module to our arduino and attempt to wirelessly transmit information to the overhead vision system. At first, we will send meaningless information, for example the heading information we receive or a prewritten message, but once our OTV is complete, we will test sending mission task information to the overhead vision system.
    - Pass: We are able to view our messages at the main computer of the overhead vision system.
    - Fail: We are not able to view our messages at the main computer of the overhead vision system.

3. From within the landing zone, your craft must travel directly to a point right before the limbo bar and log (3.4 m away) in under 3 minutes. Please note:
    a. You may remove all obstacles.
    b. You may "catch" your OTV on the other end of the arena (no programming is necessary for this sub-task).
        - We will start this test at the start of propulsion testing described on page 46. Ideally, we do create code that performs this task and potentially stops before the limbo/log, but if we feel it necessary, we will connect the battery to the motors directly so they move forward.
        - Pass: The OTV is able to move forward and reach a point right before the limbo/log from the landing zone in under 3 minutes.
        - Fail: The OTV is not able to either move forward, reach a point right before the limbo/log from the landing zone, or do this process in under 3 minutes.
4. From the open zone, the OTV must be able to pass either the limbo or log into the goal zone
    - We will use the code from the previous test, but edited so the OTV will only go the required distance it needs to go, or we will connect the battery to the motors directly so they move forward and catch the OTV in the goal zone.
    - Pass: The OTV is able to navigate past the limbo or log into the goal zone without breaking, which is defined as not being able to operate after the test.
    - Fail: The OTV either cannot navigate past the limbo or log into the goal zone or breaks.
5. Within the arena, your craft must perform a 90 degree turn. This must be demonstrated 3 times in a row. Please note:
    a. You may choose to adjust the wiring from that of the prior sub-task (forward locomotion) to make your OTV turn, or you may use programming to have your OTV complete this sub-task.

  - We will complete this task with code to ensure rotation in both directions. Furthermore, we will add an additional requirement in consecutive tests that the OTV must rotate about its geometric center.
  - Pass: The OTV is able to complete 3 consecutive 90 degree turns where 2 of those turns are in one direction and the last turn is in the opposite direction of the previous 2 turns.
  - Fail: The OTV is either not able to turn or cannot turn 3 consecutive times.

6. Your team must demonstrate that the sensors used for obstacle avoidance have been calibrated.
   - Since we are using snap action switches, the only calibration needed is to ensure the arduino receives a signal from the switches. So, for this test, we will have two tests, one for basic functionality, and one as a goal for obstacle avoidance.
   - Basic Functionality Test: Connect the switches to the arduino, activate the switches, and determine their activation with the serial monitor
     - Pass: Activation of switches is reflected on the serial monitor
     - Fail: Activation of switches is not reflected on the serial monitor
   - Obstacle Avoidance: The OTV will be placed behind (on the side of the landing and mission zone) a row of 1 or 2 obstacles such that the obstacle avoidance switches are facing the obstacles. If we choose one obstacle for the test, it will be placed behind the middle of the obstacle and the obstacle will be placed on the bottom end of the arena. If we choose two obstacles for the test, the OTV and first obstacle will be placed in the orientation described before and the second obstacle will be placed immediately above the first obstacle. The arduino on the OTV will have code that will move the OTV towards the first obstacle, detect when it hits an obstacle, then make the OTV move backwards, turn left, move forward to the middle of the obstacle course, turn right, and move forward. If there are 2 obstacles, the code will repeat this process from where it detects the obstacle.
     - Pass: The OTV is able to detect the obstacles it encounters and navigate around them.

- Fail: The OTV is either not able to detect the obstacles or not able to navigate around them.

7. Navigation code - no obstacles
    a. Navigation code must demonstrate navigation from a random starting position and orientation in the landing zone to the mission site and then to a point right before the limbo and log.
    b. Navigation code may be demonstrated using the simulator (provided that you talk through the navigation and code with your instructor) or your team's functional OTV.
        - This test will first be done through the simulator then conducted with the OTV
        - Pass: The OTV upon multiple randomized test runs orients itself towards the mission site, navigates to the mission site, and navigates to a point before the limbo/log from the mission site
        - Fail: The OTV cannot consistently either orient itself towards the mission site, navigate to the mission site, or navigate to a point before the limbo/log from the mission site.
8. Navigation code - obstacle avoidance
    a. Navigation code must demonstrate that under randomized conditions, the OTV could successfully encounter and navigate past one obstacle.
    b. Navigation code may be demonstrated using the simulator (provided that you talk through the navigation and code with your instructor) or your team's functional OTV.
        - This test will first be done through the simulator then conducted with the OTV. Ideally, we plan to complete this test by repeating test 6 but with randomized obstacles, but if we cannot do this, the test will repeat test 6 but only require the OTV can pass at least one obstacle from the mission site rather than navigating to the limbo/log.
        - Pass: The OTV upon multiple randomized test runs orients itself towards the mission site, navigates to the mission site, and either navigate to a

point before the limbo/log from the mission site or navigate around at least one obstacle

- Fail: The OTV cannot consistently either orient itself towards the mission site, navigate to the mission site, or navigate to a point before the limbo/log from the mission site.

When evaluating these tests, we will first perform the tests. If the test is passed, we will note that we have passed the test under the conditions we performed the test, including the state of the OTV and the specific code used. If a test is not passed, we will first evaluate whether the hardware and software is set up correctly. For electrical errors, we can run previously passed tests or debug using the serial monitor to determine if electrical components are being recognized by the arduino. For software errors, we can run previously passed tests, run the code on the ENES100 website, or debug using the serial monitor to determine logical errors. For mechanical errors, we can compare CAD connections with the OTV to ensure the physical OTV is set up the way we planned. If those errors are fixed and we still do not pass a test, we will evaluate our approach to the test, brainstorm ways to improve the OTV, and go from there.

Electrical Systems Testing:

There are two main tests that must be confirmed for our OTV to qualify that relate to the electrical system. The first test is that the kill switch is able to turn on and off the OTV whenever it is activated. The second test is that the OTV is able to run for 10 minutes at full power without recharging the battery.

These tests will be conducted whenever a set of electrical components are added to the OTV. The kill switch test should always be conducted directly after the set of electrical components are added to the OTV. The runtime test should be conducted anytime after the electrical components have been added. Depending on the situation, this test can be conducted multiple days after the electrical components have been added. Passing these tests requires the completion of the description of the test and failing means the incompleteness of the description of these tests.

If a test is passed, there is no need to test it again until more electrical components are added to the OTV. If a test is not passed, we will check whether the electrical components are connected in accordance with the circuit diagram, Figure 23. If the test is still not passed after connections are confirmed, we will reevaluate the connections and calculations. Additionally, we will notify a professor or TA for their assistance.

Mission Systems Testing:

Throughout the production of the second layer of the OTV, mission systems tests will be conducted to ensure that the mission systems perform appropriately and complete the mission. Unlike propulsion systems testing, these tests must be performed in the order they are outlined in this document, but when they are performed in terms of the production timeline is flexible. Furthermore, we do not need to pass all of these tests before MS5, however production of each layer needs to be completed by MS5. Below are the tests we will conduct during production of the mission systems. Descriptions of pass/fail requirements along with testing procedures and production requirements will be described below each test. Each test is a prerequisite test for the following tests and test cases are defined in each definition of the failure of a test.

1. Tolerances of Claw Subassembly and Interface Plate Testing:
    - Tolerances will be tested in-house using a pilon mockup to ensure tight tolerances. We desire sub-mm tolerance on these parts to ensure reliability during the mission. Testing in house will provide a quick turn-around for any modifications, such as sizing adjustments or changing the angle of the claw servo to account for tolerance issues.
    - Pass: Tolerances are appropriate for functionality of claw and interface plate, including rotation of the claw and antenna contact with the inside of the pylon.
    - Fail: Tolerances are not appropriate for the functionality of either the claw or interface plate such that the claw can rotate and the antenna makes contact with the inside of the pylon.
2. Sensor Testing:

- Reed switch, electrodes, and wires will be added to the interface plate but not permanently adhered. The reed switch will be added to the bottom of the interface plate and the electrodes and wires will be added to the antenna. Arduino code will be used to output sensor readings to the serial monitor on the Arduino IDE. To test the reed switch, the magnetic puck will be placed in the interface plate in multiple orientations and the output from the serial monitor will be confirmed. To test the antenna, the pylon will be put on the interface plate and the serial monitor will output the duty cycle of the pylon.
- Pass: The serial monitor displays both a confirmation that the reed switch has been activated and the duty cycle the pylon outputs
- Fail: Either the serial monitor does not confirm the reed switch has been activated or the duty cycle the pylon outputs is not displayed on the serial monitor.

3. Claw and Pylon Rotation Testing:
    - The pylon or mockup will be placed in the claw and the rotation mechanism will activate to ensure that the claw arm rotation does, in fact, grab the pylon and mate the pylon with the interface plate. Additionally, the puck inside the pylon will land in the interface plate after rotation. Once this has been verified, the adjustable limit switch can be adjusted so that it triggers when the silo is centered in the claw.
    - Pass: The pylon is rotated by the claw from an upright position to an upside down position onto the interface plate such that the antenna makes contact with the inside of the pylon and the puck goes into the plate.
    - Fail: Either the claw cannot pick up the pylon, the claw cannot rotate the pylon, the pylon does not mate with the interface plate such that the antenna connects with the inside of the pylon, or the puck does not fall into the interface plate.
4. Full Mission Run Test
    - This test will be used to verify that the systems work as expected in the field, and used to determine whether the whisker sensors will be needed in the mission. In this test, the OTV will approach and align itself with the pylon, the claw will grab the pylon and rotate the pylon onto the OTV, and the sensors will read and

transmit the data collected from the pylon and puck. For each test, we will determine beforehand whether the puck is magnetic and the duty cycle of the pylon and compare those results with the results transmitted by the OTV.

- Pass: The OTV is able to perform the mission task such that the data transmitted to the overhead vision system is the same as what we have predetermined before the test.
- Fail: Either the OTV cannot align itself with the pylon, the OTV cannot grab or rotate the pylon onto the OTV, the pylon does not mate with the interface plate, the sensors do not collect the correct data, or the OTV does not transmit the correct data to the overhead vision system.

When evaluating each test, we will first perform the test. If the test is passed, we will repeat the test to make sure it passes every time. If the test is not passed, we will first evaluate for error using evaluation methods described in the last paragraph for propulsion and navigation system testing on page 51. If the test is still not passed, we will reevaluate the production of the mission system parts, and if needed, reproduce the parts with adjustments. If that is deemed not an issue, then we will evaluate the approach to the test, brainstorm additional ideas, and go from there.

If we are able to pass all of these tests with our OTV, we should be able to complete a full run of the mission and obstacle course without failure consistently after necessary combinations of code.

Teamwork

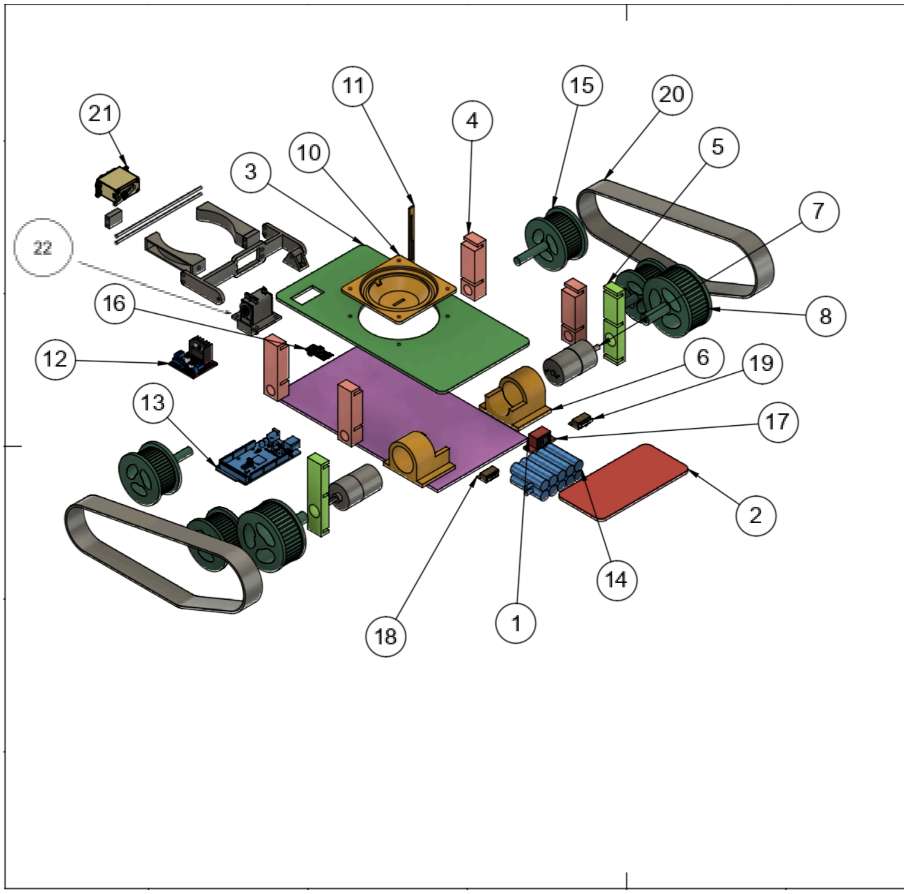
Our design reflects the team’s goals and interests because we decided the overall design through a bracket format, taking which idea had more votes to move on until we unanimously decided on an Oscar the Grouch’ Garbage Truck themed OTV that would function like a garbage truck. This theme and method of voting demonstrates our team's ability to come together and share ideas and combine our interests.

When dividing into subgroups, we each assigned ourselves to subteams we were most interested and/or experienced in. Dividing this way ensured each individual would perform most effectively to the team and get the most out of the class for themselves. Despite some of our experience, the assignments in this class stretch our skills and knowledge and push each of us to research and try new things while having other people to lean on and ask for help, helping us advance and grow closer as a team.

Currently, our team is doing well with our design habits. We have been able to effectively create and use models to our advantage to both communicate and store ideas. Also, we have kept up with updating our models when there are changes to the design of our OTV. We plan to continue what we have been doing in terms of updating our models throughout production. As we go into the build phase, we will reevaluate our Project Management Plan to create clear deadlines and outline subteam dependencies. This will give us a clear vision on our OTV production and when we will be conducting certain tests. Furthermore, we will begin documenting through discord and google drive our testing progress and use GitHub to share testing code.

As a team we are struggling with communication, whether that would be asking for help or when we feel there is an issue. This is something we need to work on and should actively reflect on as we need to work together to be successful in this course. This plays a huge role in equitable teamwork, improving communication can help us ask for help if we feel like we are doing too much or too little and distribute work accordingly. When we meet, we are extremely productive and are able to get a lot done. This is where most of our productive discourse occurs. We meet at least once a week, with subteam meetings between the main meetings. Something we can include into our weekly meetings is an open discussion of what we can improve on as a team. Also, we can improve our communication by making larger individual efforts to communicate in discord and actively speak up about our individual availability and specific plans for the OTV throughout the week so that work can be distributed equally and subgroups are not left in the dark. We have not experienced any major failures, but we have been able to successfully work through miscalculations and other challenges together, lifting each other up along the way.

Bill of Materials



Item	Qty	Part Number	Material
22	1	FS5115M Servo	
21	1	2ServoClaw	
20	2	Belt	Steel
19	1	P:\E-Switch Secure\3D_MODELS_STP_FORMAT\MS085xx02FxxxP1A.stp	Steel
18	1	P:\E-Switch Secure\3D_MODELS_STP_FORMAT\MS085xx02FxxxP1A.stp	Steel
17	1	RA1113112R	Steel
16	1	TCRT5000L Module	
15	4	Support Wheel	Steel
14	1	Battery	Steel
13	1	ArduinoMega_STEP_AP203	Steel
12	1	L298N Driver DC Motor, Stepper Motor	Steel
11	1	CargoInterfacingPlate_Antenna	Steel
10	1	CargoInterfacingPlate_Plate	Steel
9	1	SiloMockup	Steel
8	2	Drive Wheel	Steel
7	2	Finalized motor	Steel
6	2	Motor_mount	Steel
5	2	DoubleHeightSpacer	Steel
4	4	MS2Spacer	Steel
3	1	MiddleDeck	Steel
2	1	TopDeck	Steel
1	1	BottomDeck	Steel

Parts List

Figure 45. Exploded Assembly Drawing of the OTV

Diagram Number	Product	Amount	Manufacturer	Vendor	Model Number	Cost [$]	Mass [kg]
13	Elegoo MEGA R3 ATmega 2560	1	Elegoo Inc.	Amazon	EL-CB-003	$22.99	0.064
12	BOJACK L298N Motor DC Dual H-Bridge Motor Driver	2	BOJACK	Amazon	BJ-L298N	$13.98	0.080
14	Tenergy NiMH Battery Pack 12V 2000mAh High Capacity Rechargeable Battery	1	Tenergy	Amazon	N/A	$21.99	0.255
18	MS0850502F030 P1A Snap Action Switch	2	E-Switch	DigiKey	EG5141-ND	$2.70	0.002
17	RA1113112R Rocker Switch	1	E-Switch	Mouser Electronics	612-RA1113112R	$0.71	0.003
16	TCRT5000 Line Follower Sensor	1	AYASOSO	Amazon	B0D3TC3DTH	$1.75	0.040
N/A	Reed Switch	1	N/A	Amazon	N/A	$0.40	0.001
21	MG996R Servo Motor	1	TowerPro	Amazon	N/A	$4.50	0.055
22	FS5115M Servo Motor	1	FEETECH	Pololu	N/A	$19.95	0.062
7	Greartisan DC 12V 30RPM Motors	2	Greartisan	Amazon	B071GTTX4D	$30.00	0.444
N/A	Other Electrical Components (ex. Wires, Resistors, WiFi)	N/A	N/A	Amazon	N/A	$10.00	0.050
4	3D Print Spacers	8	In-House (UMD or Personal Printer)	N/A	N/A	$0.50	0.100
1,2,3	Recycled PLA	N/A	In-House (UMD or Personal Printer)	N/A	N/A	$1.00	0.560

N/A	3D Print Mission Task Components	N/A	In-House (UMD or Personal Printer)	N/A	N/A	$0.50	0.100
8,15	3D Print Wheels	6	In-House (UMD or Personal Printer)	N/A	N/A	$0.50	0.120
6	3D Print Mounts	2	In-House (UMD or Personal Printer)	N/A	N/A	$0.50	0.040
20	GT2 Timing Belt Closed Loop Rubber Belt	1	Zeelo	Amazon	ZR-200527001	$8.00	0.020
Total						$139.97	1.996

Figure 46. Bill of Materials

Our pending approval date for all of the products listed above is 03/24/25. Some model numbers could not be found due to vendors not listing them. 3D printed components will be printed with the combination of ENES100 3D printers, Terrapin Works printers, and Steven Vilcheck's personal 3D printer which will reduce manufacturing costs.

Sources

- Greartisan Motor: https://www.amazon.com/Greartisan-Electric-Reduction-Centric-Diameter/dp/B071GTTX4D?th=1
- L298N Motor Driver: https://www.amazon.com/BOJACK-H-Bridge-Controller-Intelligent-Mega2560/dp/B0C5JCF5RS
- Tenergy 12V 2000mAh Battery: https://www.amazon.com/Tenergy-Capacity-Rechargeable-Replacement-Equipments/dp/B077Y9HNTF
- Elegoo Mega R3: https://www.amazon.com/ELEGOO-Compatible-Arduino-Projects-Compliant/dp/B01H4ZLZLQ?source=ps-sl-shoppingads-lpcontext&ref_=fplfs&smid=A2WWHQ25ENKVJ1&gQT=2&th=1
- Kill Switch: https://www.mouser.com/ProductDetail/E-Switch/RA1113112R?qs=QtyuwXswaQh2BdIH9uXjxQ%3D%3D&srsltid=AfmBOopBEbEc44a1-6RePW94ggrCRb7zkmhJ6UIrceyEO6N4JO47oK9G
- Snap-Action Switch: https://www.digikey.com/en/products/detail/e-switch/MS0850502F030P1A
- Velcro Peel Strength:
- https://www.hookandloop.com/brands/velcro/peel-and-stick
- Line Follower: https://www.amazon.com/TCRT5000-Infrared-Follower-Obstacle-Avoidanc/dp/B0D3TC3DTH?gQT=1
- Tank Treads: https://www.amazon.com/Zeelo-Timing-Pulley-Meters-Printer/dp/B0897CJKS1
- Arm Drive Servo Motor: https://www.pololu.com/product/3426

- Claw Servo Motor: https://www.amazon.com/4-Pack-MG996R-Torque-Digital-Helicopter/dp/B07MFK266B?th=1