Princeton University

MAE 322: Mechanical Design

Prepared for: Daniel Nosenchuck, Glenn Northey, and Al Gaillard

Search and Rescue Robot: Final Design Report

Jose Yanez, Serg Zhelezniak, Andrew Witmer, Kate Andre, Matthew Hetrick, Alex Hsia, Greg Luck, Maya Naphade

May 14, 2019

Contents

1 Executive Summary

Caesar is a Search and Rescue Robot that can deliver small medical kits through difficult terrain to victims in need. It is designed to navigate a course that consists of: retrieving a 4" x 4" x 4", 3.0 lb medical kit, navigating through an obstructed course, scaling a 12" high wall with a 6" step on one side, traveling through a 3' wide curved chute, and finally delivering the medical kit to a basket that has an incandescent light (representative of the victim). Medical kit retrieval and obstacle course navigation are performed using open-loop control, whereas traveling over the wall, traveling through the chute, and driving to the victim are performed using closed-loop control.

Caesar's design was an adaptation of a traditional battle tank in order to replicate effective battlefield navigation systems. To meet the specific requirements of this course, additional features such as arms attached to the four corners of the long side of the rectangular chassis (Caesar's frame) were added to increase Caesar's versatility in various terrains. Other mechanisms such as light sensors, proximity sensors, motors, and gears were added to the design to maintain the robot's robustness throughout the course, specifically as it traverses over the wall. On the arms are two pairs of treads, solely implemented for the traversal of the wall. Caesar traverses the wall by raising its front pair of arms up onto the 6" step and the treads on these arms propel the robot up onto the first step using a battery-powered motor. Subsequently, the arms rotate forwards to propel the robot over the wall, much like a four-legged animal would climb a flight of stairs. To complete the rest of the tasks, Caesar employs proximity and light sensors. The proximity sensors are programmed to autonomously navigate through the curved chute and to stop the robot sufficiently close to the victim (the basket), while the light sensors are programmed to autonomously navigate to the victim (the incandescent light). Time is also an important factor in the design of Caesar; the entirety of the robot was designed to minimize weight so that the course can be completed rapidly thus maximizing the probability of saving the victim.

Ultimately, Caesar is designed to be robust, controllable, and efficient, allowing it to easily traverse obstacles and efficiently deliver the medical kit to the victim.

2 Introduction

The objective of this project was to create a Search and Rescue Robot (SaRR) that was capable of autonomously retrieving and delivering a medical kit to a victim in need. The victim was represented by an incandescent light. The medical kit was a 4 inch x 4 inch cube with a 3 inch long cylindrical handle. The handle had a 1 inch wide cylindrical knob with a 1.5 inch diameter which the robot gripped. In order to get to the "victim," the robot needed to avoid obstacles in its path, which were represented by an obstacle course. The obstacle course consisted of an array of cones and markers that the SaRR had to navigate using open-loop remote control steering. After traversing the course, the robot had to scale a 12 inch high wall with a 6 inch step on one side, as presented later in the report.

After the step, the robot entered a 3 foot wide chute that had turns of 30 degrees, -60 degrees, and 30 degrees respectively. The robot then navigated autonomously to the "victim" and delivered the medical kit into a 4" deep and 11" wide open box.

The first task that the robot had to complete was to retrieve and store the medical kit. This task was completed using open-loop remote control. The robot completed this task using the extendable arm on the top surface of the body of the robot, which is depicted in the Design and Analysis section. After collecting the med kit, the robot had to navigate through the obstacle course with cones and other obstructions. This obstacle course was also navigated using open-loop remote control.

After completing the open-loop portion of the course, the robot was tasked with surmounting a wall using closed-loop control. The wall is one foot high and three feet wide, with a six-inch step on the front side and a one-inch diameter rigid horizontal bar three feet above the wall. The robot's arms (located on either side of the main body and covered in treads) were implemented to climb over the initial step and again over the wall itself. The arms were designed have a 360-degree range of motion to turn in a full circle and pull the body of the robot onto the step and then again over the wall. A schematic representation of the robot with two of its arms fully extended is shown in Figure 1.

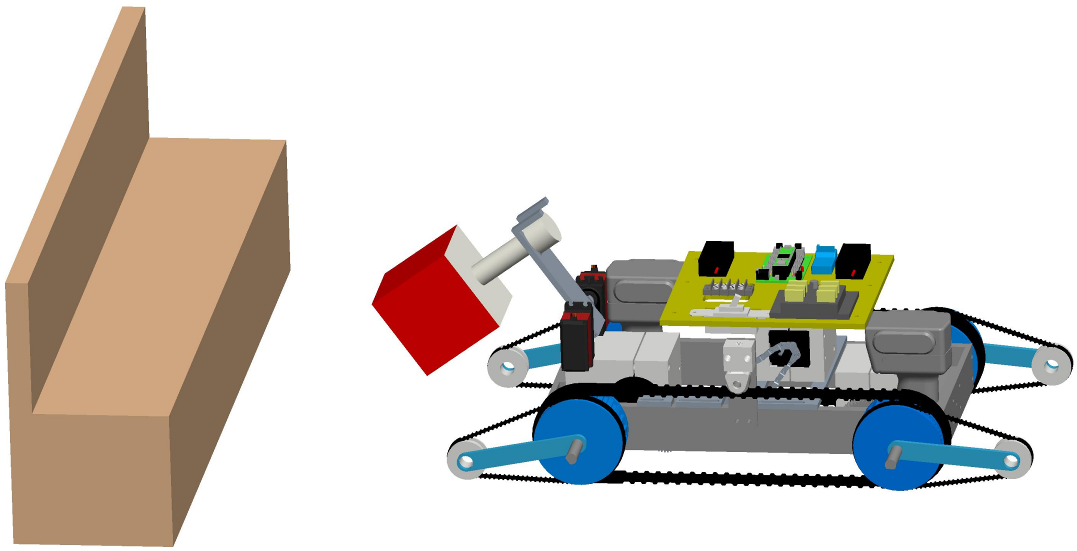


Figure 1: **Caesar with Medical Kit and Wall**

After getting over the wall, the robot had to navigate through the angular chute. The chute was 3 feet wide, with three turns between the wall and the final goal. The robot navigated through the chute autonomously, using proximity sensors to avoid hitting the walls on either side of the chute. Upon exiting the chute, the robot had to use a light beacon as a guide to deliver the medical kit to its final destination by way of light sensors. Both the light and proximity sensors were coded using an Arduino, and the sensitivity of these sensors was adjusted for the robot to stop far enough from the wall to avoid colliding with it. The medical kit was then "delivered" by being placed into an open basket which is 4 inches high and 11 inches wide.

During the design process, research was conducted into existing rescue robot designs and tank vehicles. In this research, it was found that many rescue robots use treaded tracks over their wheels and arms to help navigate potentially rugged terrain. These two features were also incorporated into our final design, as they are some of the features that would be the most practically useful for a rescue robot. The design of the treads was inspired by different designs from military tanks, which have a high degree of maneuverability and are capable of traversing difficult terrain.

The design philosophy of the robot attempted to establish a balance between elegance and utility. The primary design consideration during the development of this robot was making the robot as lightweight as possible while still being able to achieve all of the required tasks. Each component of the robot was constructed to effectively accomplish a particular aspect of the course. Furthermore, the design was created with the goal of having ease of motion; each component on the robot was placed intentionally to make the robot mobile and easily controllable. For example, the side arms were treaded along with the wheel treads in order to provide traction and reduce slip when the robot tried to pull itself up onto and over the steps. The treads on the arms added a utility that would not have otherwise been present and allowed for Caesar to pull itself onto the step without being overly heavy or cumbersome. The arms demonstrated one example of the balance between elegance and utility that motivated much of the robot's design.

3 Project Management

3.1 Key Milestones

Each given deadline corresponded with a major subsystem of the robot that needed to be completed. Incorporating these deadlines allowed for time to test and refine each subsystem. Furthermore, the sequential addition of individual subsystems to the overall robot revealed flaws that arose from the attempted integration of components designed by different sub-teams within our group. An overview of the timeline for the project is given below:

Due Date	**Objective**	**Subsystem**	**Projected Time**
01 March	Open-loop drive train control (lab groups)		24 Hours
08 March	Autonomous navigation to a light source	Sensor Integration	24 Hours
05 April	Speed trials with a ramp	Drive-train	28 Hours
12 April	Open-loop wall traverse	Wall traverse mechanism	50 Hours
19 April	Open-loop package retrieval and placement	Gripper arm mechanism	48 Hours
26 April	Closed-loop chute traverse and package placement	Autonomous control	32 Hours
13 May	**Complete Project**		
14 May	**Final Competition**		

Table 1: Key Deadlines

Note that the anticipated time for each objective included robot assembly, for which all members were responsible. However, it did not include setback troubleshooting or potential system optimization for a given task. In terms of setback troubleshooting, we implemented what is best described as an iteration approach. We identified the specific problems that caused a setback, analyzed potential solutions to those problems, and continuously implemented what we determined to be the solutions most likely to remedy the problems until the identified setback was resolved. In addition to our own team, we consulted both Glenn and Al whose tremendous experience in the machine shop proved invaluable during this process.

In terms of working hours, our team had 8 members. With each member spending a minimum of 3 hours per week in the lab (or working remotely), a minimum total of 24 hours of work time per week was easily achieved. Weeks with more intensive objectives resulted in more extensive time commitments. Preliminary assessments predicted that total completion time for the robot would be approximately 400 hours. However, in actuality, the total time necessitated for completion was approximately 1200 hours, nearly three times the initial assessment. This was due to multiple unforeseen design and manufacturing difficulties.

3.2 Division of Responsibilities

Each team member was assigned responsibility of a specific subsystem(s). This way, the team could focus on and specialize on specific aspects of the robot, making the overall design process more efficient. The subsystem teams were designed to be cross-functional so that each team member had thorough knowledge of more than one system of the robot, allowing for expertise to be dispersed.

The team was split up into four different groups: design, machining/assembly, electronics, and coding. All 8 team members participated in assembling the robot throughout the semester regardless of sub-team specialization. The design team also acted as co-Project Managers, helping each other to delineate responsibilities as well as helping the team stay focused and on track. Weekly team meetings also took place on Tuesday afternoons and a dedicated group chat was created in order to communicate throughout the week.

Sub-Task	**Team Member Assigned**
Design	Jose, Serg, and Andrew
Code	Andrew and Serg
Electronics	Andrew and Serg
Manufacturing and Assembly	Jose, Serg, Andrew, Greg, Maya, Kate, Alex, Matt

Table 2: Division of Responsibilities

3.2.1 Overview of Management Approach

The project managers were responsible for keeping everyone on track by sending weekly messages to the central GroupMe chat, along with more specific goals for the subteams each week. Certain teams needed more help to achieve a given weekly goal, at which point the project managers worked with members of other subteams to schedule helping hours based on availability. As previously stated, there were weekly debriefing meetings to ensure that the entire team was aware of all goals for a particular week and to identify solutions to any potential holdups inhibiting a certain goal, at which point the team discussed the optimal strategy to overcome the setback. For setbacks specific to a subteam, that subteam was in charge of implementing a collaboratively-planned solution, and in the event of a serious challenge, all members were asked to contribute to tackling the problem.

To resolve inter-team conflicts, discussions were held to remedy tense situations. If absolutely necessary, professor discussions would have been held to resolve the situation.

For team motivation, the feeling of testing successes drove a lot of our work. We also all wanted to make the robot work and have the sense of accomplishment of creating something so involved and challenging. Also, the project managers would send motivating messages to the group to keep morale high at the beginning of each week.

The overall management philosophy was that the project managers would outline the tasks for the week but the individual teams carried ultimate responsibility for completing the necessary tasks that would result in the achievement of a weekly milestone. Unfortunately, this system did not work perfectly and not all milestones were always met. In these situations, however, there was an effort made to keep in mind the importance of staying on track as each milestone, even if not completed on the date of the test, had to be completed in order to ensure that the robot was fully functional for the final deadline.

In terms of final reflections, we did not set enough SMART goals in the beginning of the course or plan out the entire semester and ensure the communication of such plans to everyone. We followed a one week at a time approach to meet each milestone rather than setting a clear plan at the beginning and use that to align all of our calendars. As a result, there was much miscommunication regarding when and who would come in to work on a particular part of the robot, thus causing a lot of unproductive time and redundant work that inhibited our ability to be efficient.

3.3 Cost

Our team was given an overall budget of $500, with which specialty objects could be purchased. All stock materials were sourced from the machine shop. A breakdown of our budget allocation was as follows:

Item	**Source**	#	**Total Cost**
Main Drive Treads	Royal Supply	2	$141.96
Sub Drive Treads	McMaster Carr	4	$85.00
6061 Aluminum Stock	McMaster Carr	1	$43.55
High Tolerance 6061 Aluminum Rod	McMaster Carr	1	$35.81
UHMW Tube Stock	McMaster Carr	2	$15.85
SAE 660 Sleeve Bearing	McMaster Carr	6	$50.50
Hitec D-845WP Servo Motor	Amazon	2	$205.64
Combined Cost			$476.15

Table 3: Budget Overview

4 Specifications

4.1 Proposed Specifications

The proposed specifications of Caesar are listed below. The most noteworthy aspect of Caesar was the decision to use multiple treads instead of wheels to successfully scale the wall. Although the robot's chassis was designed such that either end could have been designated as the front, one end was chosen and the gripper arm and auxiliary arms were placed on the chassis accordingly.

It was initially expected that the robot would weigh around 18 lbs. The robot's actual weight following completion was 25 lbs. This was determined by weighing each of Caesar's individual components to arrive at total weight amount. Additional dimensional specifications of the robot are provided below.

Aspect	**Length (in)**
Chassis Width	9
Chassis Length	17.5
Total Height	8
Wheel Radius	3.8

The large belts, one on each side of the robot, have a belt width of 1" and a pitch of 1/2". The smaller belts, a pair on the front and back, have a belt width of 1" and a pitch of 0.200".

It is expected that Caesar's maximum possible speed will be 4.8 mph. Calculations for this measure were obtained by conducting a speed trial of the robot. The robot was timed on its ability to drive a 50ft straight segment of road at maximum speed. The time to complete the 50ft segment was 7.1s, resulting in a speed of 7.04ft/s or equivalently 4.8mph. Given ample time for the scaling of the wall, the course can be completed in approximately 90-200 seconds. When solely cruising at full speed which corresponds to the minimum power draw, the robot can operate for 3.2 minutes with a maximum range of 0.256 miles. At maximum power draw, when all mechanisms are in use, it can operate for 2.1 minutes. This corresponds to a range of approximately 0.1085 miles.

4.2 Operational and Navigational Modes

Caesar has both an operational and a navigational mode. It has an open-loop RC drive mode in which the robot is controlled via a wireless RC controller that allows an operator to both control the robot's motion and its medical kit retrieval arm. The autonomous navigational mode uses multiple sensors in order to perform the wall, chute, and light-seeking tasks.

Caesar is controlled by a Teensy microcontroller that receives input from the wireless RC controller, two Cadmium Sulfide light sensors, and one proximity sensor. For the operational mode, the RC controller has two joystick inputs and two switch inputs. One joystick is used to control the motion of the robot such that the vertical channel of the joystick controls its forward and backward motion while the horizontal channel controls its turning. The other joystick is used to control the rotation of the auxiliary arms with the vertical channel and the movement of the mechanical retrieval arm with the horizontal channel. One of the switches of the RC controller is used to switch between the open-loop operational mode and the closed-loop autonomous mode. For the autonomous mode, the only input that is used from the RC controller is the switch that controls the mode of operation. Additionally, the robot receives inputs from the light and proximity sensors in order to perform the autonomous tasks.

When the robot is switched to autonomous mode, it performs three different tasks: wall traversal, chute navigation, and light navigation and medical kit delivery. The wall traversal is performed by the front two arms of the robot as well as the forward drive mechanism of the chassis. In the first step, the robot rotates the two arms forward lifting the chassis while the main treads move in the forward direction pushing the chassis over the first step and onto the second. The robot then repeats this sequence, rotating the front arms over the second step while the main treads are moving in the forward direction in order to climb the second step and traverse the wall. In the case that the robot does not fully traverse the wall and has the back end stuck on the wall's edge, the back arms rotate in the forward direction and push the robot over the second step. For the chute navigation task, two proximity sensors are used to determine the robot's horizontal position in the chute. The robot moves forward while using both proximity sensors to maintain its position in the middle of the chute. Once the robot fully exits the shoot, the light navigation task begins. This task is initiated once the robot senses it has exited the chute. Caesar uses both of the light sensors as well as the front proximity sensor for this final process. Initially, the robot rotates until it identifies the light source. Using the light sensors to keep the light source centered in front of the robot, the robot moves forward until it reaches a distance close to the target receptacle indicated by the proximity sensor. The robot then begins the med kit drop off phase in which the arm moves downward until it is above the receptacle. The med kit is then subsequently released.

4.3 Control System

The autonomous operation of the robot, particularly with regards to the control logic that allows for its obstacle avoidance and medical kit delivery, can be decomposed into three steps: wall ascent, chute traversal, and light-guided delivery.

4.3.1 Wall Ascent

Caesar's wall ascent is accomplished through the coordinated operation of the main belt drive and the two front-facing arms. Upon approaching the first step of the wall, the code positions the arms on the first step while activating the main treads. The action propels the robot onto the first step. The arms are subsequently rotated to touch the second step and the main belt treads activated to propel the front edge of the robot over the wall. The med kit is also lowered to move the center of mass of the robot behind the wall and allow for easier traversal onto the opposite side. Prior to falling downward, the arms are once again rotated in order to cushion the fall of the robot and prevent damage. A block diagram outlining the coding logic of the wall ascent is shown in Figure 2.

4.3.2 Chute Traversal

In order to traverse the chute swiftly and efficiently, the robot utilizes three proximity sensors, with one positioned on its left and right sides respectively and one positioned on the front of the chassis. The two side sensors allow for the robot's detection and determination of its proximity to the sides of the chute while the front sensor allows for its determination of whether it has exited the chute. Sensor placement is discussed in greater detail in Section 5.6. Navigation of the chute is accomplished by means of a proportional control law:

$$Lwheel = 1400 + K\Delta \tag{1}$$

$$Rwheel = 1600 + K\Delta \tag{2}$$

where K is proportional constant determined by trial and error and Δ determines the difference in proximity values between the left and right sensor of the robot. The 1400 and 1600 value for the Lwheel and Rwheel respectively

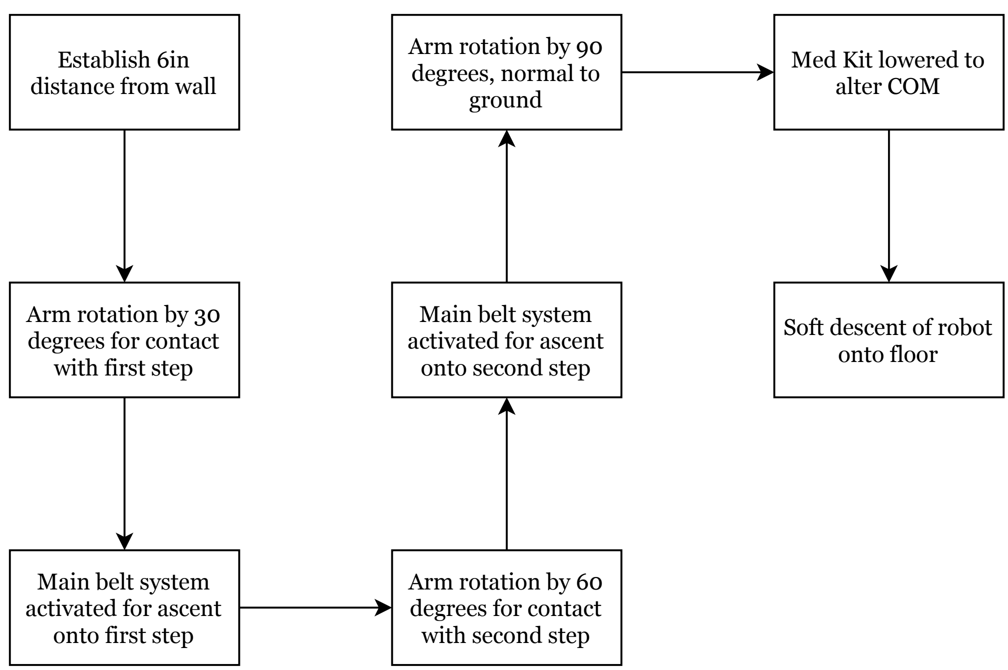


Figure 2: **Wall Ascent Logic**

allow for the robot to be constantly moving forward while making adjustments using the proportional feedback. Upon exiting the chute, the front sensor will note an increased signal strength thus causing the chute traversal code to terminate and the light-sensing code to initiate.

4.3.3 Light-Sensing and Med Kit Delivery

Two light sensors are placed on the front of the robot in order to detect and follow a source of light provided by the medical kit deposition box. In order to detect the source of light, the code establishes a minimum light value threshold and keeps rotating the robot until the position of the light is found. The robot moves in the direction of the detected light, making directional adjustments that maximize the light signal input. Upon reaching a distance of 5in from the source of light, the front facing proximity sensor prevents any additional motion of the robot. The arm code is subsequently activated, with the arm lowering the med kit into the deposition box. The block system schematic for this process is shown in Figure 3.

The complete block system diagram for the autonomous logic, inclusive of all sections of the course, is presented in Figure 4.

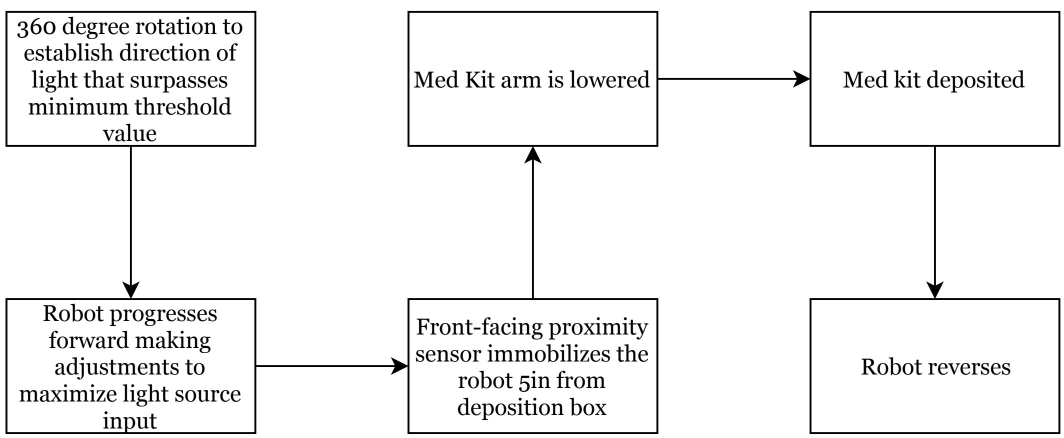


Figure 3: **Light Sensing Logic**

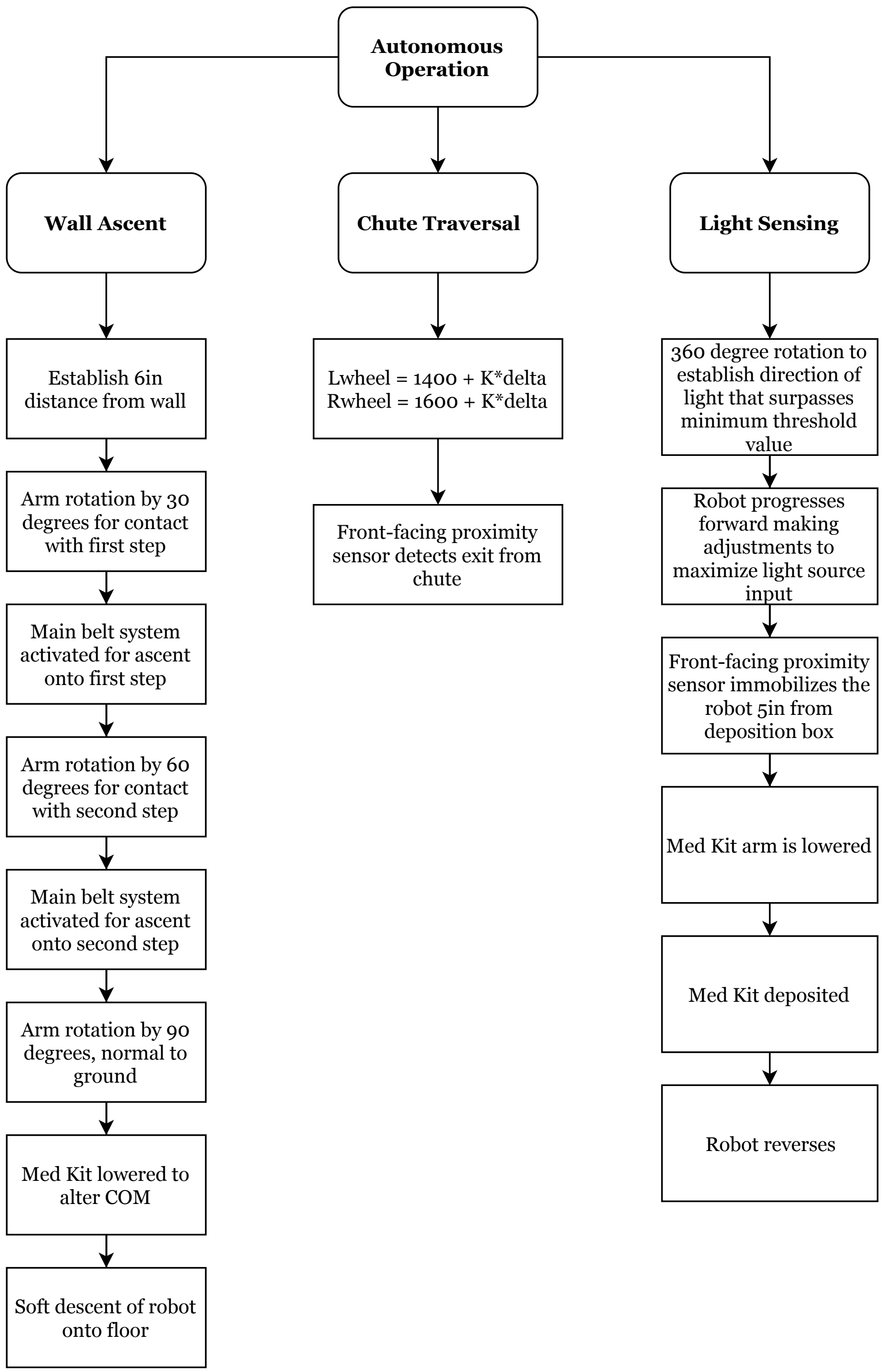


Figure 4: **Autonomous Code Block Schema**

5 Design and Analysis

5.1 Motion Design

Caesar's design centered on the premise of constructing an all-terrain system that would be able to retain a significant degree of contact with any flooring material. Preliminary research indicated that wheels, while simple, may have a tendency to slip, especially on adverse surfaces. A rubber tread system would be able to account for this particular disadvantage by means of maximizing the surface area of the robot that would be in contact with the floor. As a result, Caesar's main means of motion was decided to be provided by rubber belts with teeth on both sides, as shown in Figure 5.

Figure 5: **Main Robot Treads**

The treads used to drive the robot are the Gates TP400H100 series, with a belt pitch of 0.5in, a pitch length of 40in, and a pitch width of 1in. The teeth protrusions on both sides of the belt allow it to be driven by an adapted gear mechanism described in Section 5.2, as well as for providing significant grip during the robot's ascent over irregular terrains such as wooden steps. Due to the enhanced grip and increased surface area, these tank treads were determined and selected as the optimal motion mechanism for the robot, taking into account the obstacles and terrains that it would have to undergo in the extent of the course.

5.2 Gear Design

In order to facilitate Caesar's motion, two gear mechanisms that adequately mesh with the particular dimensions and characteristics of the TP treads were required. Design of the gear mechanisms was developed by adapting the CAD file of the TP400H100 belt and building the gear around the belt such that the dimensional characteristics of the two allowed for precise meshing. A visual representation of the preliminary driving wheel design is demonstrated in Figure 6. The driving wheel features a small driving gear composed of 20 teeth and a pitch angle of 20 degrees that would be driven by a gear of 50 teeth and a pitch angle of 20 degrees. The wheel also features a belt track used propel the TP tank tread and an arm belt track that is used for rotating the robot's arms. The arm mechanism is presented and discussed in Section 5.4 of the report. As the characteristics of such a wheel are complex in

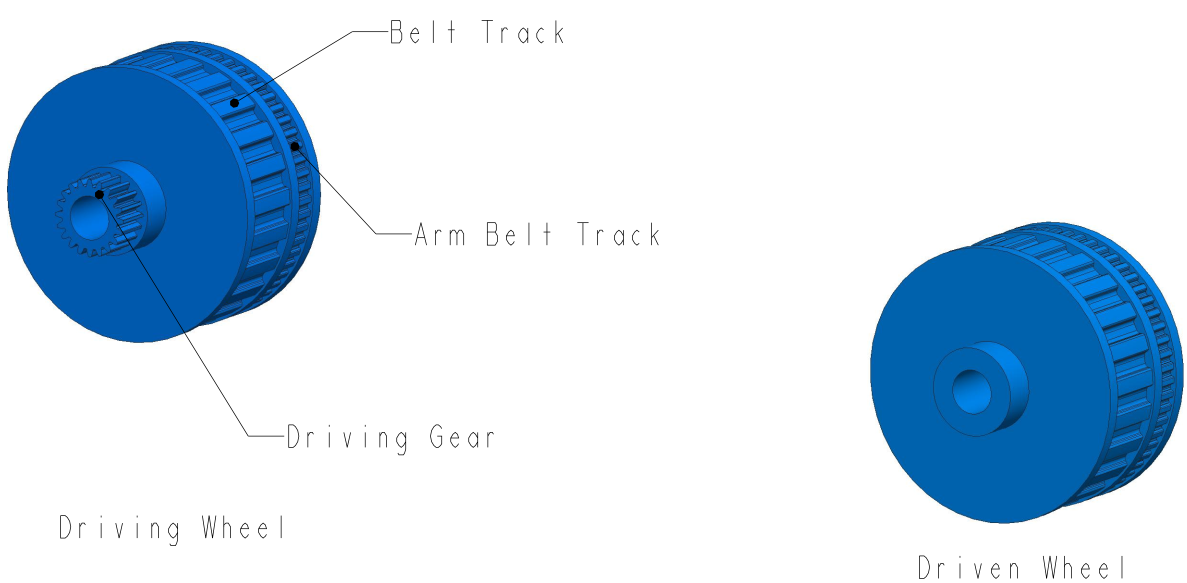


Figure 6: **Driving and Driven Wheel Design.**

nature and difficult to manufacture by manual machining, the wheels were 3-D printed. A similar design process was implemented for the construction of the driven wheel. The driven wheel is shown in Figure 6.

Initial testing of the plastic gears revealed an unanticipated flaw: material wear due to extreme torques. The small gear on the driving wheel experienced excessive amounts of torque produced by the plastic gear of the motor, resulting in the plastic teeth disintegrating on both the wheel and the motor gear and the robot being immobilized. To fix the issue, the small plastic gear on the driving wheel was removed and replaced by a metal gear with identical characteristics that featured the same quantity of teeth and the same pitch angle. Similarly, the plastic gear attached to the motor was replaced by a metal gear that also featured 50 teeth and a pitch angle of 20 degrees. Both metal gears are shown in Figure 7.

Both driving wheels were powered by two CIM 12.0V Vex Robotics Motors. The rotation of the motor would turn the aforementioned 50 tooth metal gear that would subsequently turn the small driven gear, thus propelling the robot forward. The process described is displayed in Figure 8.

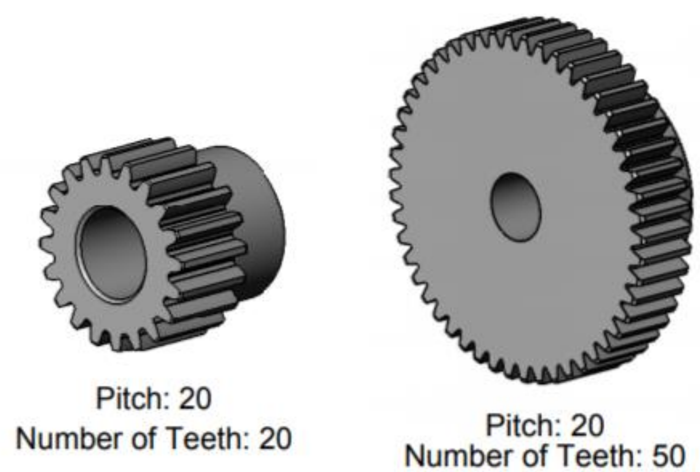


Figure 7: **Metal Gear Replacements**

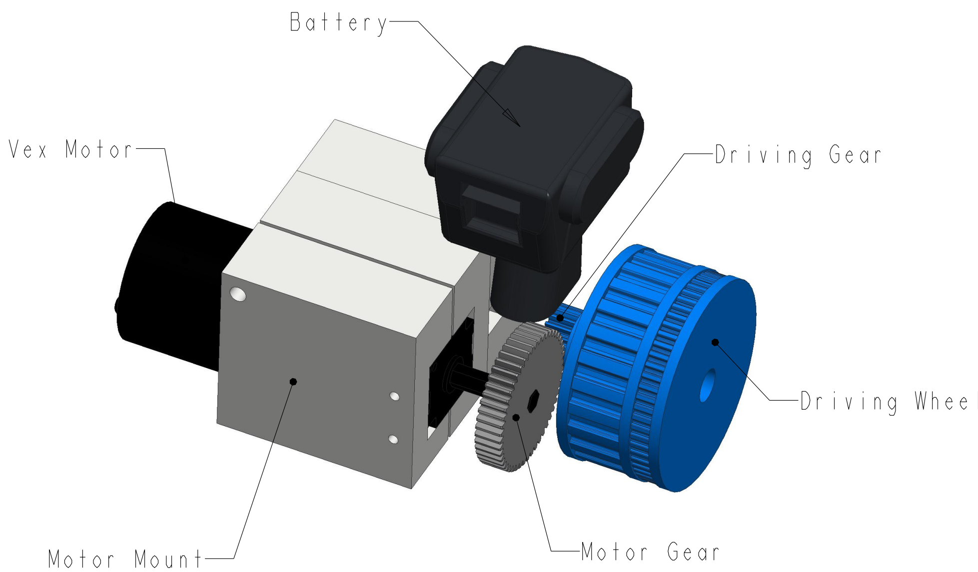


Figure 8: **Motor and Driving Wheel Interaction**

5.3 Chassis Design

Two sets of driving and driven wheels were manufactured to allow for Caesar's motion. The four wheels were subsequently attached to a rectangular, aluminum chassis as shown in Figure 9. An aluminum chassis was utilized in order to minimize the weight of the entire robot and allow for more efficient and swifter maneuverability. The driven gears were placed on opposite sides of the chassis in order to allow for efficient motor placement as well as enable Caesar to have turning capabilities. The opposite placement of the two wheels and motors allowed Caesar to make both left and right movements.

The motors were tightly secured to the aluminum frame to prevent miscellaneous disturbances and perturbations during the robot's movement and ascent of the steps. In order to produce sufficient tension in the main belt that propels the robot forward, two tensioners were also attached to the aluminum chassis. Adjustment of the tensioners increased the contact between the driving wheels and the TP400H100 belt, thus providing sufficient grip

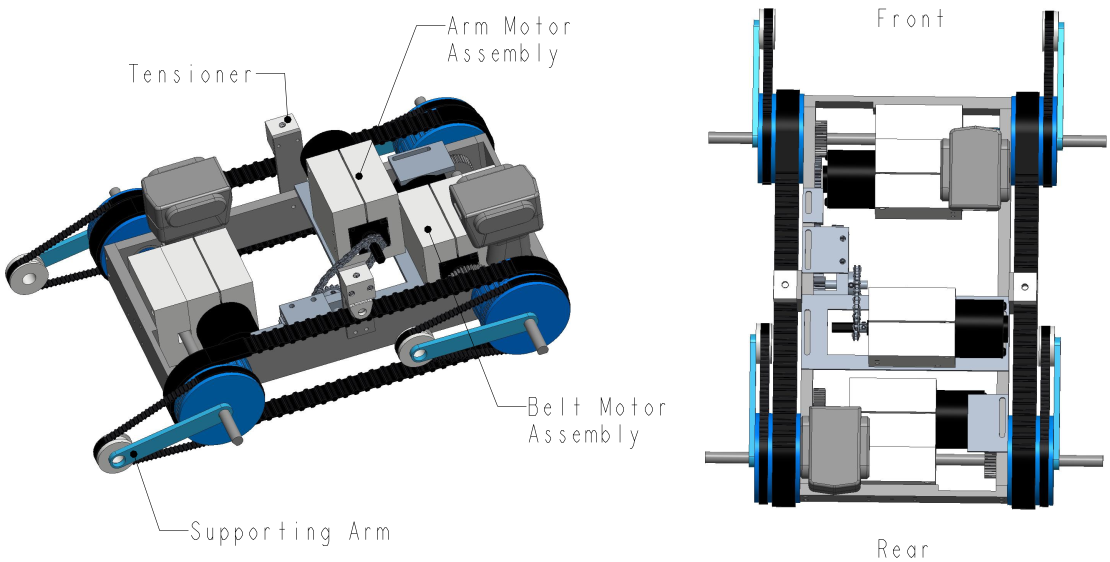


Figure 9: **Chassis Schematic**

and alleviating any issues with belt slippage.

5.4 Arm Design

In order to get over the 12-inch high wall, treaded arms were added to the front of the robot. These front-facing arms measure 7 inches in length and are driven by TP200xL037 belts. The treads on the arm are connected to the same 3-D printed gears that drive the main treads, so every time the main drive treads are spun up, the arm treads are spun as well. However, rotation of the arms themselves is driven my an independent motor. Therefore, there are two separate forces that can be transferred through the arm: a pulling force from the treads and a pushing/rotating force from the rotation of the arms.

During normal operation, these treaded arms remain in a vertical position to not interfere with normal operations. When the robot approaches the step and switches into autonomous mode however, the arms deploy and rotate until they make contact with the lip of the first step. Once the arms are

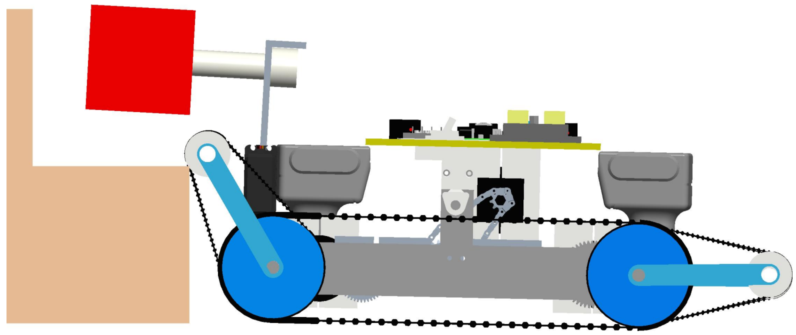


Figure 10: Initial Contact with the Wall

in contact with this elevated step, as seen in Figure 10, further rotating the arms allows the robot to lift itself up and driving the treads allows it to pull itself up onto the step. The robot continues pulling itself upwards until the main drive treads come into contact with the step. Once main drive contact is established, the main drive can provide the additional required traction to propel the robot over the step. This process is repeated again to get the robot over the second step of the wall.

Finally, once atop of the wall as seen in Figure 11, the arms can be rotated under the robot to push against the back side of the wall. This push provides enough force to push the robot's center-of-gravity past the peak of the wall, allowing the robot to land on the other side.

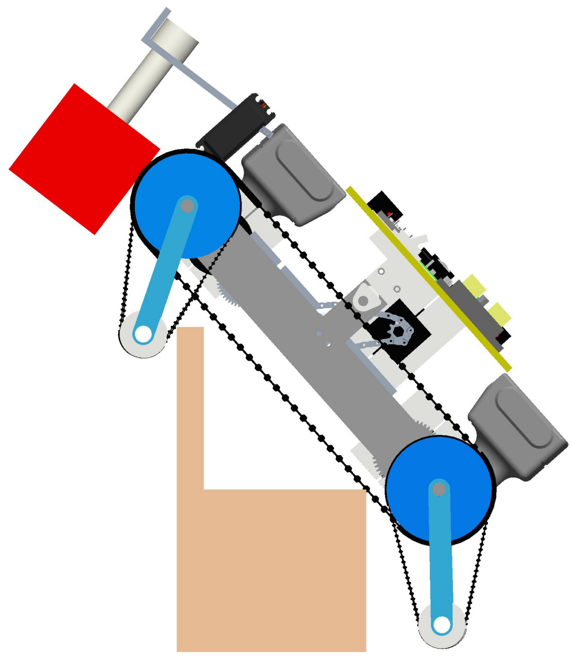


Figure 11: Traversing up the Wall

Analysis of the step ascent using these arms is outlined in Section 5.7.3.

As an additional note, a rear arm – with identical design to the front arm – was initially a planned configuration of the robot. These rear arms would have provided additional torque and robot pitch angle, but their addition was deemed too complicated and not rewarding enough. Currently, these arms are locked in place as to not be accidentally activated during the robot's operation. On the flip side, when these arms locked into place pointing towards the rear of the robot, their length provides additional tipping stability to the robot, making it less likely to tip over itself backwards while attempting to climb over the wall.

5.5 Kit Retrieval Arm Design

To retrieve the med kit, a simple 6-inch arm with a grappling hook design was used. The arm is attached to two Hitec 2845WP servo motors, each with a stall torque of 50 kg-cm. The gap in the grappling hook is designed to be slightly larger than the handle diameter of the medical kit. This way, by approaching the medical kit with the retrieval arms level, as seen in Figure 13, the robot can put the medical kit within the hook's reach. By lifting the retrieval arm, the medical kit becomes locked in place as seen in Figure 14. In this configuration, the medical kit remains secure in all reasonable robot orientations: the medical kit cannot move vertically out of the grappling hook due to its narrow receptacle diameter and cannot move laterally out of the grappling hook because of the two prongs at the front.

Discussion of the servo motor choice, as well as a torque analysis of the medical kit lifting procedure is outlined in Section 5.7.4.

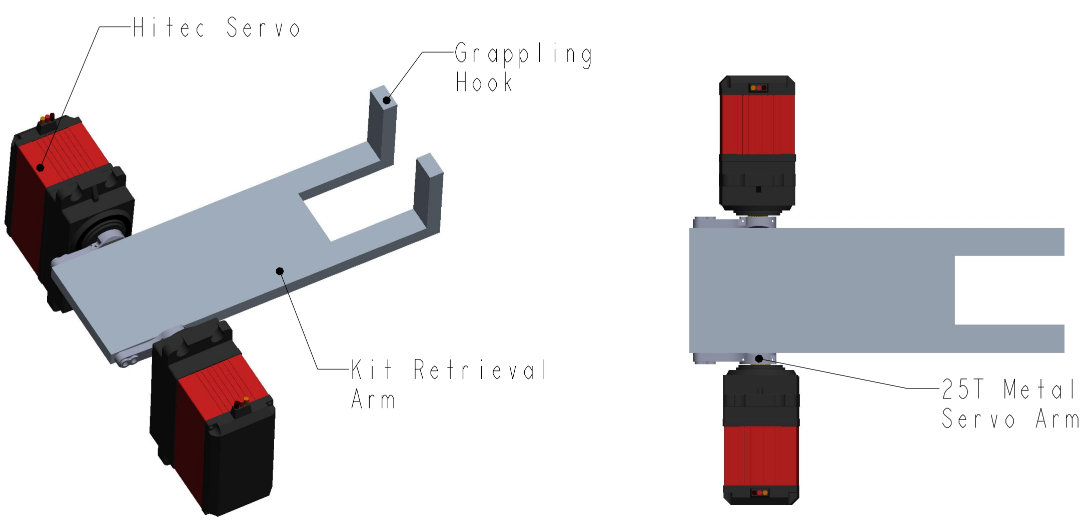


Figure 12: **Kit Retrieval Arm**

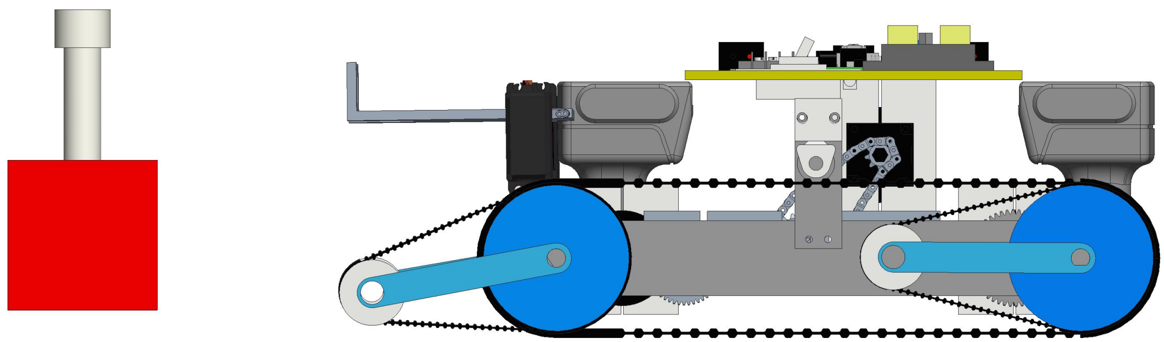


Figure 13: **Medical Kit Approach**

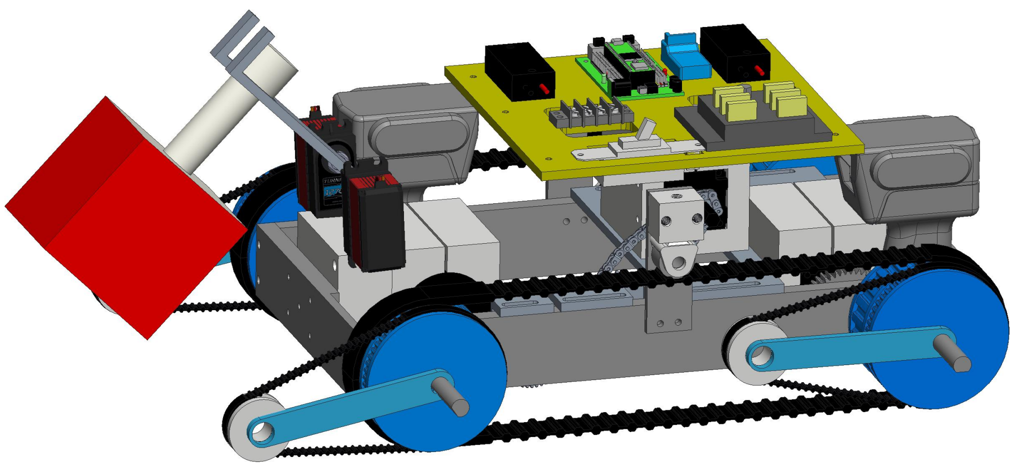


Figure 14: **Driving Configuration**

5.6 Sensor Placement

In order to autonomously navigate the chute, three proximity sensors were placed on the robot. The first two sensors are placed on the sides of the robot. The relative difference in readings between these two sensors tells the robot its relative orientation in chute. To adjust for any deviations between the robot's position and the center line of the chute, the robot can rotate itself away from the nearest wall and drive itself to achieve a better tracking of the chute center line.

The third sensor is placed on the front of the robot to detect when the robot successfully exits the chute. This sensing step is required to inform the robot when to switch to the next autonomous phase: light tracking. If light-sensitive control had been used for the entirety of the autonomous drive portion of the test run, the robot might have erroneous control inputs due to randomized light noise that the sensors pick up from around the room. Conversely, failure to switch into light-sensing at the right time prevents the robot from accurately and quickly moving to deposit the medical kit at the correct location.

5.7 Static and Dynamic Analysis

5.7.1 Weight and COM

The motion of the robot and its capacity to surmount the wall depends in part on its center of mass and the manner in which it is altered by the raising and lowering of the med kit via the kit retrieval arm. In order to analyze these aspects, the robot was deconstructed into the following sections: driving motor assembly, arm motor assembly, and med kit retrieval arm.

Driving Motor Assembly: The total mass of the driving motor assembly, as shown in Figure 15, was determined by means of a scale to be 4.9lbs. Weighing each individual component (i.e. plastic casing) separately and incorporating the values into CREO allowed for the determination of the center of mass of the assembly that is displayed by the position of the 'X' symbol in Figure 15.

Arm Motor Assembly: The total mass of the arm motor assembly, as shown in Figure 16, was determined by means of a scale to be 5.2lbs. Weighing of each individual component separately and incorporating the values into CREO allowed for the determination of the center of mass of the assembly that is displayed by the position of the 'X' symbol in Figure 16.

Med Kit Arm Assembly: The total mass of the med kit army assembly, as shown in Figure 17, was determined by means of a scale to be 3.5lbs. Weighing of each individual component separately and incorporating the resulting values into CREO allowed for the determination of the center of mass of the assembly that is displayed by the position of the 'X' symbol in Figure 17.

Entire Robot: Taking into consideration each of these components and assembling them into the chassis of the robot yielded a final mass of the robot of 25lbs. Inputting the material composition of each component of the robot into CREO allowed for the determination of the center of mass of the entire assembled robot, as displayed by the position of the 'X' symbol in Figure 18. The point denotes the COM in the default driving position of the robot with the position of the med kit as shown in the figure. The COM would change if the position of the med kit was changed by means of servo actuation. Analysis of the dynamics of robot, inclusive of the changes of the COM on it's ascent of the wall, is presented in Section 5.7.3.

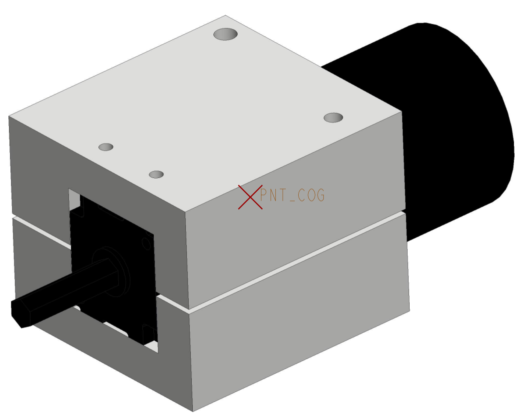


Figure 15: **Motor Assembly Center of Mass**

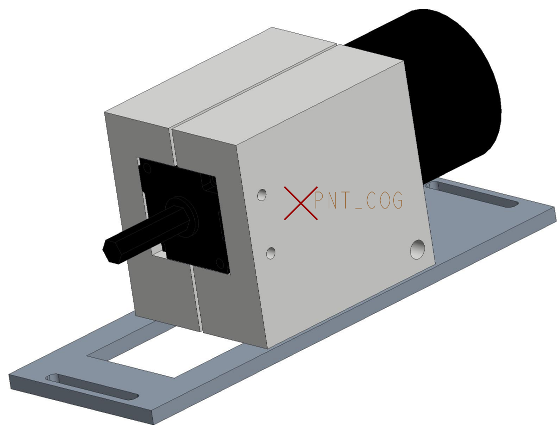


Figure 16: **Arm Motor Assembly Center of Mass**

5.7.2 Grip and Traction

One of the most crucial aspects of the robot's motion is the ability of the motors to generate enough torque that would be able to produce Caesar's stable, continuous movement and the capacity of the miniature arms to grab and propel the robot onto the step.

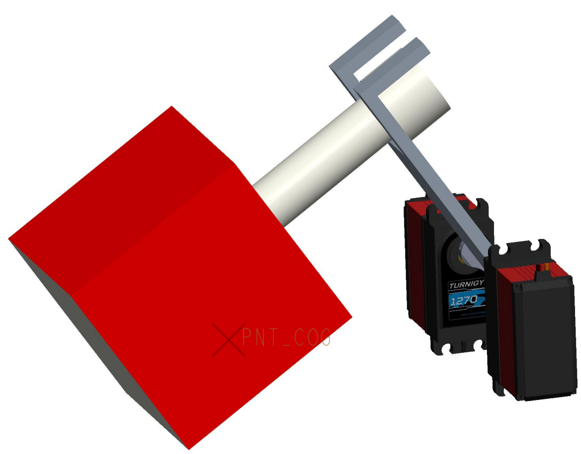


Figure 17: **Med Kit Arm Assembly Center of Mass**

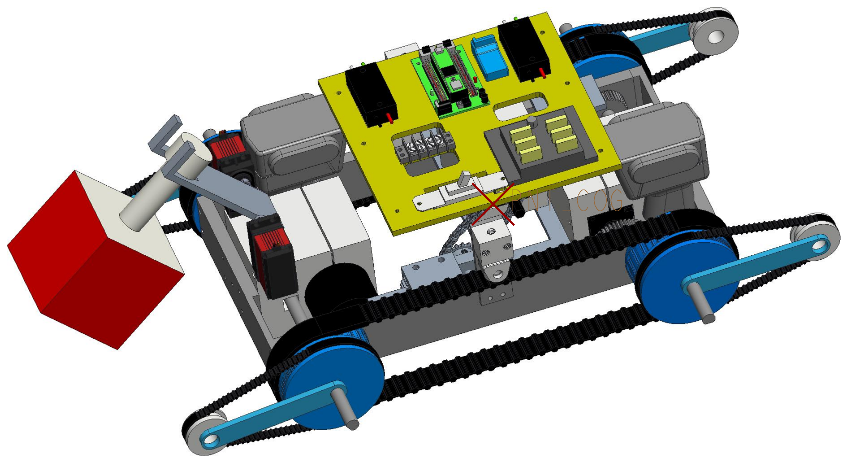


Figure 18: **Caesar Center of Mass**

In order to generate movement and establish traction, the treads must first exceed the force of static friction F_s given by

$$F_s = \mu_s \cdot F_N \tag{3}$$

where μ_s is the coefficient of static friction and F_N is the normal force exerted on the robot. The coefficient of static friction for dry wood is approximately 0.3 [1]. As noted in the Specifications sections, the mass of the entire robot will be approximately 25lbs. As the robot is nearly symmetric across its central axis, the load can be assumed to be distributed evenly across the two driving shafts and thus uniformly distributed across each of the four gears. A total of 6.25lbs, or equivalently, 2.83kg, will result in a normal force of approximately 27.8N upon each gear. This would correspond to a cumulative normal force on one side of the gear train of 55.5N. Substituting these figures into the above equation, $F_s = 0.3 * 55.5 = 16.65$N. As the CIM motor is able to output a total of 334 Nm/s, there is an excess of force that would be generated and transmitted through the rubber tread in order to exceed the static friction force, thus allowing the robot to be propelled forward. Preliminary testing presented no traction issues, as expected.

[1] Engineering ToolBox, (2004). Friction and Friction Coefficients. Accessed 03 May. 2019.

5.7.3 Arm and Wall Ascent Analysis

The first stage in the robot's ascent of the wall is contact with the first step, as seen in Figure 19 below.

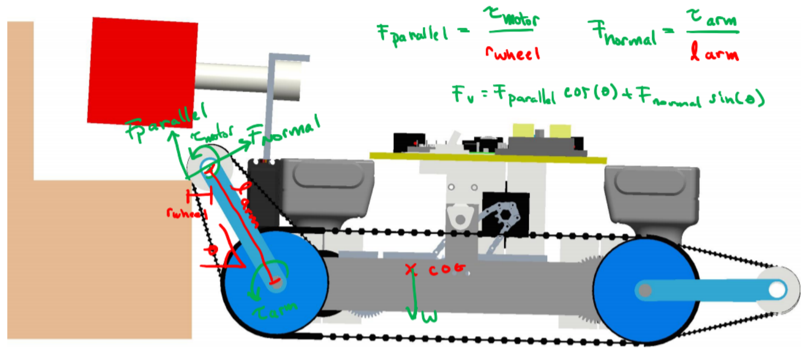


Figure 19: **Initial Ascent**

When the front arm makes contact with the lip of the first step, the arms must provide both enough torque to lift the robot and enough force through the treads to propel the robot forward so that it can place its main treads on the step. The torque requirement can be calculated by defining the vertical force component required to lift the robot, which is just the mass of the robot: 25 lb or 111.2 N. Because the main drive treads are lying flat on the floor at this stage, all lifting forces must therefore come from the arm. When looking at the contact point between the arm and the step, the reaction forces can be broken into a tangential and normal component. The tangential force comes from the driving of the treads, which are directly connected to the main drive train (whose torque is outlined in the section above). The normal force applied against the step can be calculated from the arm's length and motor torque. The motor used to rotate the arms is the same CIM motor used in the main drive train, but with a 320:1 gear reduction (which comes from 40:1 gear reduction from a planetary gearbox inline with the motor in-series-with a 8:1 gearbox). For an arm angled at θ when it makes contact with the wall, the force equation for this first stage can be summarized in the following equation:

$$\left(\frac{\tau_{arm} * gearratio}{l_{arm}} cos(\theta) + \frac{\tau_{drivetrain}}{r_{arm}} sin(\theta)\right) = F_v = W_{robot} \qquad (4)$$

Plugging in known values, approximating the contact angle $\theta = 60°$, and assuming the torque required from the main drive is half the stall torque yields

$$\Big(\frac{\tau_{arm} * 320}{0.1778m} cos(60°) + \frac{1.2Nm}{0.025m} sin(60°)\Big) = F_v = 111.2N \tag{5}$$

Therefore, the required torque for the arm motor is $0.0604Nm$. While this theoretical requirement is extremely low, losses in the system (e.g. inefficiencies in the gearbox, slipping treads, etc.) will mean a higher requirement. Regardless, if the same CIM motor that is used in the drive train is used to rotate the arms, there should be more than sufficient torque to lift the robot.

After this initial contact, as θ increases, the force requirement increasingly shifts upon the second term in Equation 4; that of the main drive train. This represents the robot using the arm treads to pull itself up the wall, rather than using the force from its arms' rotation to push itself up. This configuration can be seen in Figure 20. Here, the two points of contact are from the main drive train and the rear arms, which do not rotate. Because these two points are both driven by the same drive train, Equation 4 can be modified to the body-axis, becoming:

$$\frac{\tau_{drivetrain} * gearratio}{r_{maingears}} = W * sin(\theta) \tag{6}$$

Therefore, for a gear ratio of 2:1 and a main gear radius of 2 inches, the stall torque of the CIM motors is more than sufficient to pull the robot up the steps in this configuration using its main drive alone. This configuration can be seen in Figure 20.

Traversal of the second step is similar to the first step. Once the main drivetrain of the robot is sufficiently seated upon the first step, the front arms can rotate to make contact with the lip of the second step. The orientation of the robot in these stages can be seen in Figures 21 and 22. The dimensions of the robot make it such that the front and rear arms can simultaneously maintain contact with the upper step and floor, respectively. This allows the main drivetrain motors to exert force through the treads along a distributed contact area, improving the robot's ability to maintain traction during the climb.

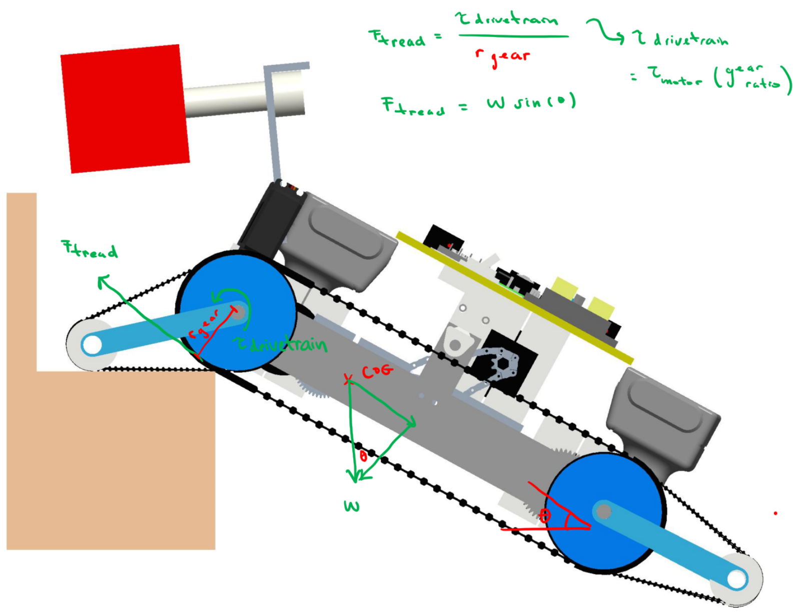


Figure 20: **Predominantly Main Drive Ascent**

During the final stage, as depicted in Figure 22, the front arm can loop over the top step and push against the back of the step. The robot pushes against the back of the step until its center of mass moves beyond the top of the step, allowing the robot to land on the other side of the wall. The center of mass can be shifted forward on the robot by manipulating the medical kit arm.

5.7.4 Wall Ascent Failure

The final design configuration of the robot was unable to scale the wall. This is due to a few flawed assumption in the analysis. Firstly, the friction between the treads and steps weren't modeled. While it was assumed that the drivetrain's torque could be fully transferred to pulling forces in the tread, the amount of force exerted by the treads is limited by their friction-based interaction with the contact surface. For example, in Figure 19, the parallel force isn't just $\frac{\tau_{motor}}{r_{wheel}}$, but rather the product of the normal force applied, F_{normal}, and the static coefficient of friction $\mu_{static} = 0.3$. Therefore, climbing forces are governed by the weight of the robot and the torque outputted

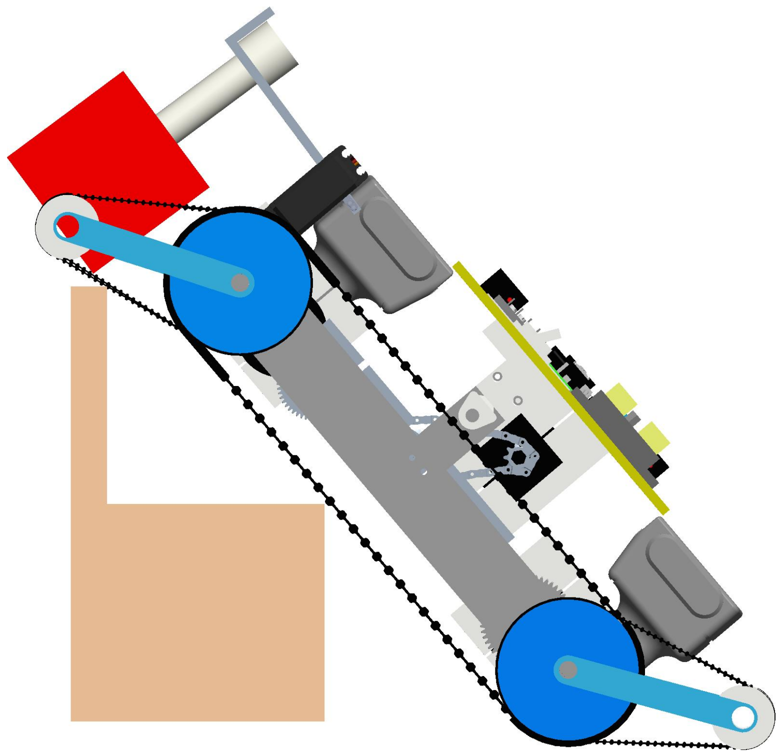


Figure 21: Initiation of Second Step Ascent

by the arm motor, rather than the torque of the drivetrain motor. With this limitation on the treads contribution to wall-climbing forces, the torque demanded from the arm motors should have been a higher priority and may have required further gear reductions to increase the overall generated torque.

Additionally, the motor's center of gravity was higher and further back than expected, due to the new position of the circuit board, battery packs, and arm motor. When the robot pitched up, this center of gravity pulled the robot backwards, away from the wall, making it difficult for the front arms and drivetrain to maintain proper contact. At this high pitch angle, the weight vector of the robot was increasingly parallel to the robot's orientation, and the robot's weight no longer helped maintain the normal force required for proper transmission of force through the robot's treads.

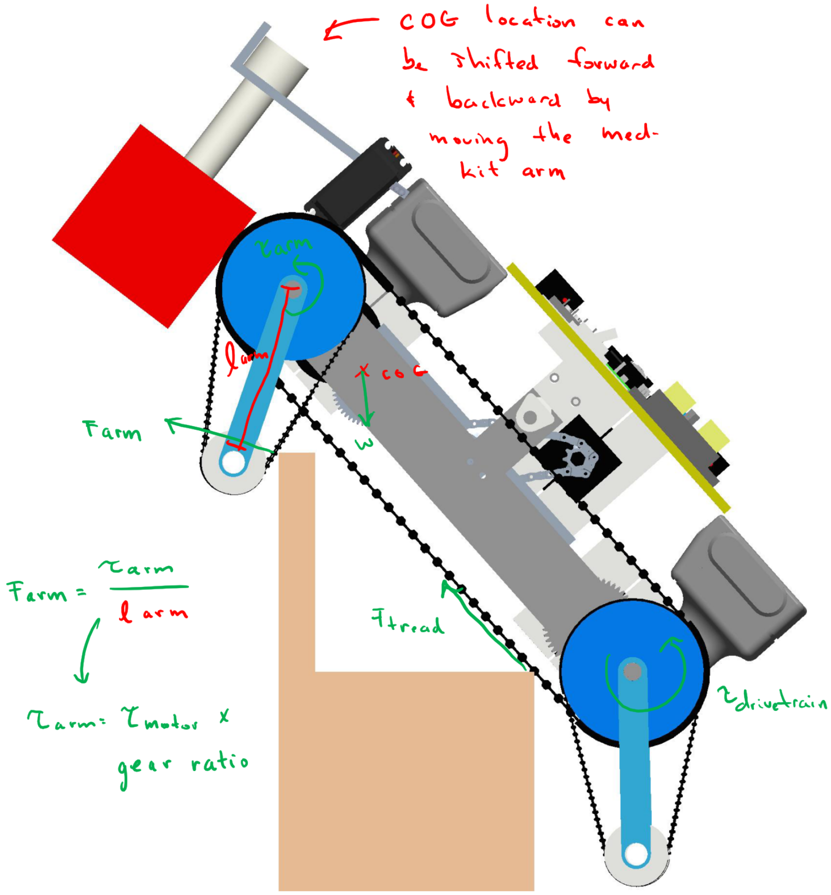


Figure 22: Culmination of Second Step Ascent

These problems could have been fixed given more time. By shifting the orientation of the robots internal components lower and more forward, the center-of-gravity could have been maintained in the favorable position it was designed to be in. Furthermore, control and actuation of the rear arms would have allowed for greater pitch control as the robot ascended the wall. Full contact of the rear arm with the ground, as depicted in Figure 22, would provide a more stable base, as well as a contact surface with high normal force for transmission of the drivetrain torque. However, the existing complexities and manufacturing difficulties with the existing design made it difficult to find enough time to test, identify problem areas, and redesign to accommodate for the challenges that we faced.

5.7.5 Med Kit Pickup Analysis

Analysis of the medical kit pickup involves a simple force and torque analysis, as shown in Figure 23. Analysis can be conducted when the arm lays flat: here, the moment exerted by the medical kit is the greatest so motor design around this point ensures that there is sufficient torque for all operating points of the arm. As mentioned in Section 5.5, two Hitec 2845WP servo motors were used in the robot, with a combined stall torque of 50 kg-cm. For a 1.36 kg medical kit and 15.24 cm arm, the maximum torque induced by the medical kit is 20.72 kg-cm. The weight of the arm itself is approximately 0.6 kg, so the torque provided by the two servo motors is more than sufficient to lift the medical kit.

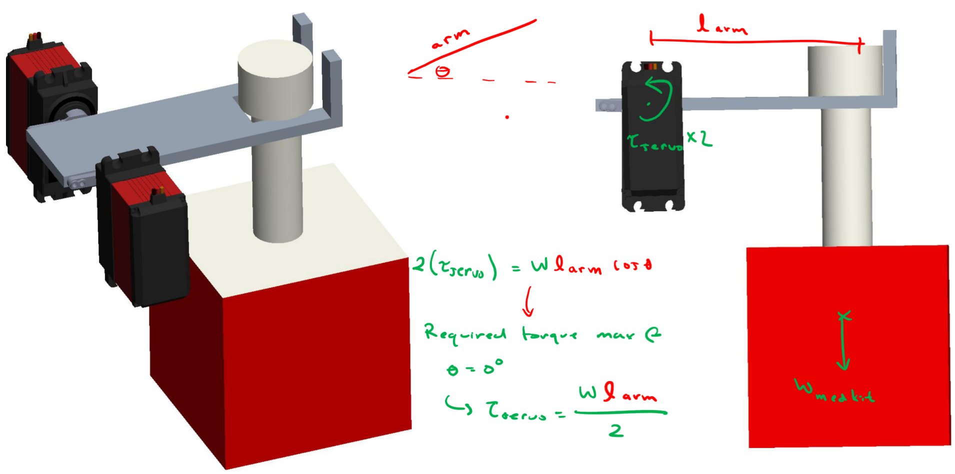


Figure 23: **Grappling Hook and Medical Kit**

5.7.6 Turning Radius

As Caesar can turn while in a fixed position by activating each motor independently depending on the desired direction, the approximate turning radius will be equal to the length of the entire chassis, which is 17.5in.

5.8 Loads and Drop Stress

The robot was designed and manufactured with the notion of symmetry to simplify the assembly process. Majority of the weight is concentrated near the front and rear of the robot, particularly over the driving axles. Battery packs and CIM motors are located at such extremities. The electronics and controls of the robot are housed at the center in order to have close proximity to both of the motors that are located at opposite ends from each other.

All of the aforementioned components have an approximate mass of 25lbs. Since there are four main gears, either driving or driven, across which this mass will be distributed evenly, each gear is expected to support approximately a quarter of the total mass or 6.25lbs.

One additional factor worthy of consideration is the amount of force exerted onto the robot in the drop test. The impact velocity from the specified 12in vertical distance can be calculated by

$$V_f = \sqrt{2g\Delta y} \tag{7}$$

where Δy is the change in height and g is the acceleration of gravity. Substitution yields a final impact velocity V_f of 2.45m/s. The impact force for a robot with a total weight of 25lbs, or equivalently 11.34kg, will be approximately 11.34*9.81 = 111.25N. This would be evenly distributed across each of the four gears, resulting in each gear experiencing approximately 27.81N of force. Due to the strength of the material used to manufacture the robot - aluminum and hard plastic - no damage of any sort is anticipated.

6 Test Results

Testing of Caesar was completed on the week of May 6th, 2019. The medical kit retrieval, speed, and wall-breaching tests were completed using open-loop control. Chute navigation and the medical kit deposition were completed using closed-loop control. To ensure that the tests were thorough, each test below was completed three times with the best, worst, and average times listed. The results that are listed below do not include the failed tests. Except traversing the wall, all other tests were completed successfully. The final course is composed of the five tests in the order presented below. The average combined time of the four successful tests (the wall-breaching test excluded because it was unsuccessful), was 65.01 seconds with a best time of 55.43 seconds and a worst time of 80.0 seconds. However, with the theoretical wall traversal taking 60 seconds, which should be plenty of time, the entire course could be completed in as little as 115 seconds.

6.1 Retrieval

Caesar was placed forwards-facing, 10 feet away from the medical kit. The timer was initiated as soon as Caesar proceeded towards the medical kit. Caesar subsequently lowered its arm, captured the handle of the medical kit, and suspended it vertically in the air, upon which point the timer was stopped. It took an average of 9.47 seconds to traverse the distance and successfully pick up the medical kit. The best time was 7.1 seconds, while the worst time was 13.5 seconds. In this case, the worst time appeared to be an outlier as the other two test trials were completed in 7.1 and 7.8 seconds respectively. The outlier occurred because on the first attempt, the medical kit was initially missed by the hook. However, this error could occur in the real-world application of Caesar because this part was completed using open-loop control.

6.2 Speed

Similar to the retrieval test, the robot was placed forwards-facing on a flat surface and a finish line was marked 50 feet away. The timer was initiated when Caesar began to move forward and was terminated when Caesar crossed the finish line. Caesar completed the trial on average in 7.1 seconds. The best time was 6.6 seconds, while the worst time was 7.6 seconds. The average time resulted in a speed of 4.7 miles per hour.

6.3 Wall-Breaching

Caesar was placed one foot away from the first step of the wall. The timer was initiated as the robot began to advance towards the wall and was terminated as soon as the robot was on the ground behind the wall. Unfortunately, Caesar was unable to surmount the obstacle successfully. The front and rear arms did not have the necessary traction to lift the robot up and over the wall, especially in the positions shown in Figure 21 and Figure 22. This occurred because the final iteration of our robot had a center of mass more towards the rear than anticipated, specifically due to the placement of the wiring board and the med-kit arm motor. A solution would be to move the center of mass towards the center and redesign the arms to have more traction. A more detailed description of this failure is discussed in the conclusion.

6.4 Chute Navigation

Similar to the speed trial, Caesar was placed forwards-facing fully inside the confines of the chute which was described earlier in the report. Unlike the previous tests, this test was completed using closed-loop control. The timer was initiated when the closed-loop control was started and was terminated as soon as the entire length of Caesar was outside the end of the chute. On average, Caesar completed this trial in 27.11 seconds, with a best time was 23.33 seconds and a worst time was 34.0 seconds. The autonomous control of Caesar was successful, however it was much slower than manual control. Caesar had to make many small adjustments including left and right turns. These movements dramatically increased the amount of time needed to traverse the chute.

6.5 Medical Kit Deposition

This test was unique because Caesar was placed away from the deposition basket. The timer began as soon as the closed-loop control was initiated and Caesar began to rotate to detect the light emitted from the basket. Once Caesar was close enough to the basket (approx. 5 inches away) the arm used to pick up and hold the medical kit dropped and deposited the medical kit into the basket. On average, this test was completed in 21.33 seconds, with a best time of 18.4 seconds and a worst time of 24.9 seconds. The autonomous control worked more smoothly in this function than in the chute portion.

6.6 In-Class Demonstration

Despite our consistent success in preliminary testing the day before the due date, we were unsuccessful in our attempts at kit retrieval, chute navigation, and medical kit deposition throughout the in-class demonstration. The first problem that we faced was with the kit retrieval, in which the robot arm did not move in its full range of motion. We were able to overcome this challenge by replacing the batteries of the motors and the controller. Due to our inability to breach the wall, we did not attempt the wall-breaching in the demonstration. Next, one of the proximity sensors on the robot did not detect the walls of the chute and thus the robot was unable to successfully navigate the chute portion of the course. Though it is unclear why this worked in testing and not the demonstration, we replaced the proximity sensor, which helped remedy the problem. We had to adjust the sensitivity values of the new proximity sensor accordingly in order for Caesar to be able to execute the chute traversal. Finally, we taped down one of the light sensors that was precariously secured to the robot, and once we did that Caesar was successful in navigating towards the light. Following these failures, we sat down to discuss our takeaways and what we can apply going forward as we look to our next endeavours.

7 Further Work and Conclusions

Caesar's design was inspired by a tank and was created with stability in mind. The robot was robust which is necessary for a rescue robot traversing a multitude of potentially hazardous terrains. The robot was able to navigate the course using a combination of open and closed-loop control to retrieve a medical kit and deposit it in a victim beacon.

In any mechanical design process, it is important to account for the error inherent in the construction process. Any machine or tool has a certain margin of error that must be accounted for in creating a design. As much of Caesar's construction involved making parts by hand, this margin of error was significantly higher than just the error margin of the tools that were being used due to the additional impact of human error. One of the biggest challenges faced when building the robot was the compounded effects of error in a design that did not allow for it. This meant that a lot of adjustments, like shaving down gears or adding plates to the chassis to hold the gearbox in place, had to be made to ensure that all of the components of the robot worked as intended. In the future, when designing a robot such as Caesar, it is important to make a robust and flexible design that accounts for the errors inherent in the human construction process so as to eliminate the necessity for constant adjustment of the design to compensate for minor issues.

Additionally, we need to address Caesar's failure to surmount the wall. After failing the wall-climbing milestone, we discussed two options: to add another motor or gearbox for controlling the back two arms or increasing the power of the motor for the front two arms. We decided to go with the motor power option to keep Caesar's structural integrity intact, so we created a gear box that used an optimal gear ratio to create more power. However, in the end this proved to be the wrong choice because Caesar's center of gravity just barely missed shifting over the wall, which could have been achieved if instead there was a small push from the rear legs. With the current status, we would have had to redesign the weight distribution of the robot, which was not plausible in order for it to successfully pass the rest of the tests. Given more time and less external commitments (like if this project was a full-time job), we would have been able to implement both solutions and work out any testing kinks. Because of prior commitments and other events that came up, specifically having to remake many of the gears and small pieces to make Caesar accomplish the other tasks, we were not able to implement both solutions and our incorrect choice ended up being our downfall in climbing the wall. Also, while many groups employed large wheels to essentially brute force their way over the wall, our use of treads exponentially increased the difficulty of implementing autonomous movement over the wall.

We need to consider what we have learned about project management and the many things we could have changed given a second go-round on this project. First and foremost, our design was too ambitious; we wanted to accomplish something that had never been done in this class. Every piece other than the motors was custom-made, whether through 3D printing or through machining. This exponentially increased the time required to build the robot as well as the potential for human error. In hindsight, this design was too complicated for the comparatively simple goals presented. Secondly, we should have implemented better testing procedures. We essentially worked up until a test and then hoped it worked in demonstration. There were no milestones where we were completely confident in our robot's success. Specifically, regarding the final demonstration, we should have run a quick check of the batteries before we began the first course attempt. We had batteries that were not working and this prevented us from being successful on our first try. We then had issues with our sensors once we fixed the batteries and had to adjust values until the autonomous navigation worked. If we had checked these little intricacies before we attempted the final demonstration, we would not have experienced the problems that occurred.

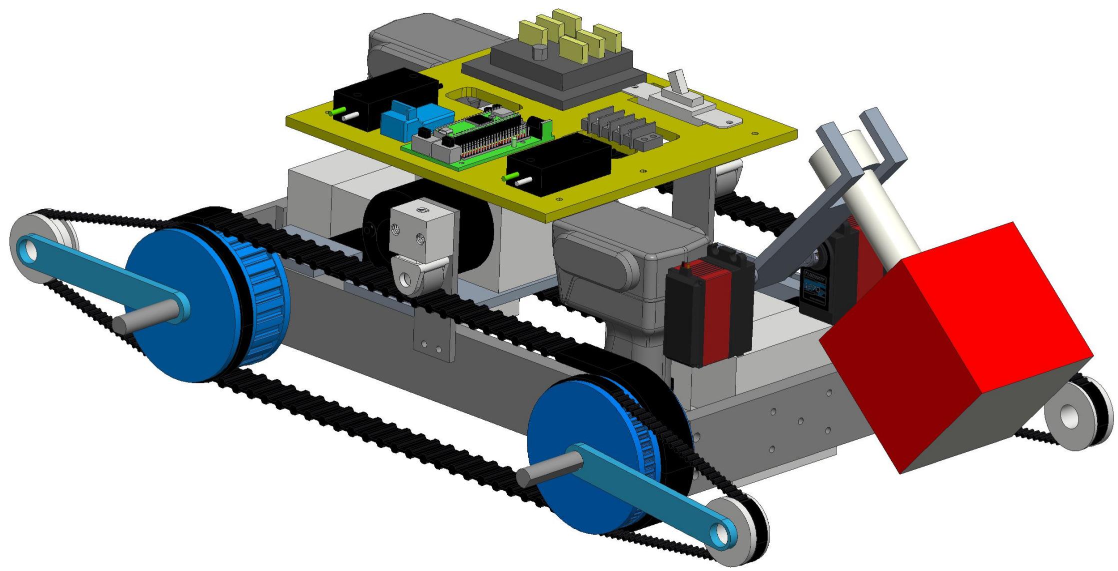
Because of these problems with our approach to constructing the robot, we learned extensively about how to approach a complicated and time-sensitive project. We now know that an extensive and organized timeline with small goals within milestones can make projects more manageable and help to divide work evenly, as well as keep the team honest and on track. Furthermore, we learned that before the presentation, we should thoroughly check our work to ensure that all of our parts are as well-aligned as they were in preliminary tests in order to make our results more consistent. We should also have plans in place in the event that there are problems with our project during testing, despite any success that we assume we will have.

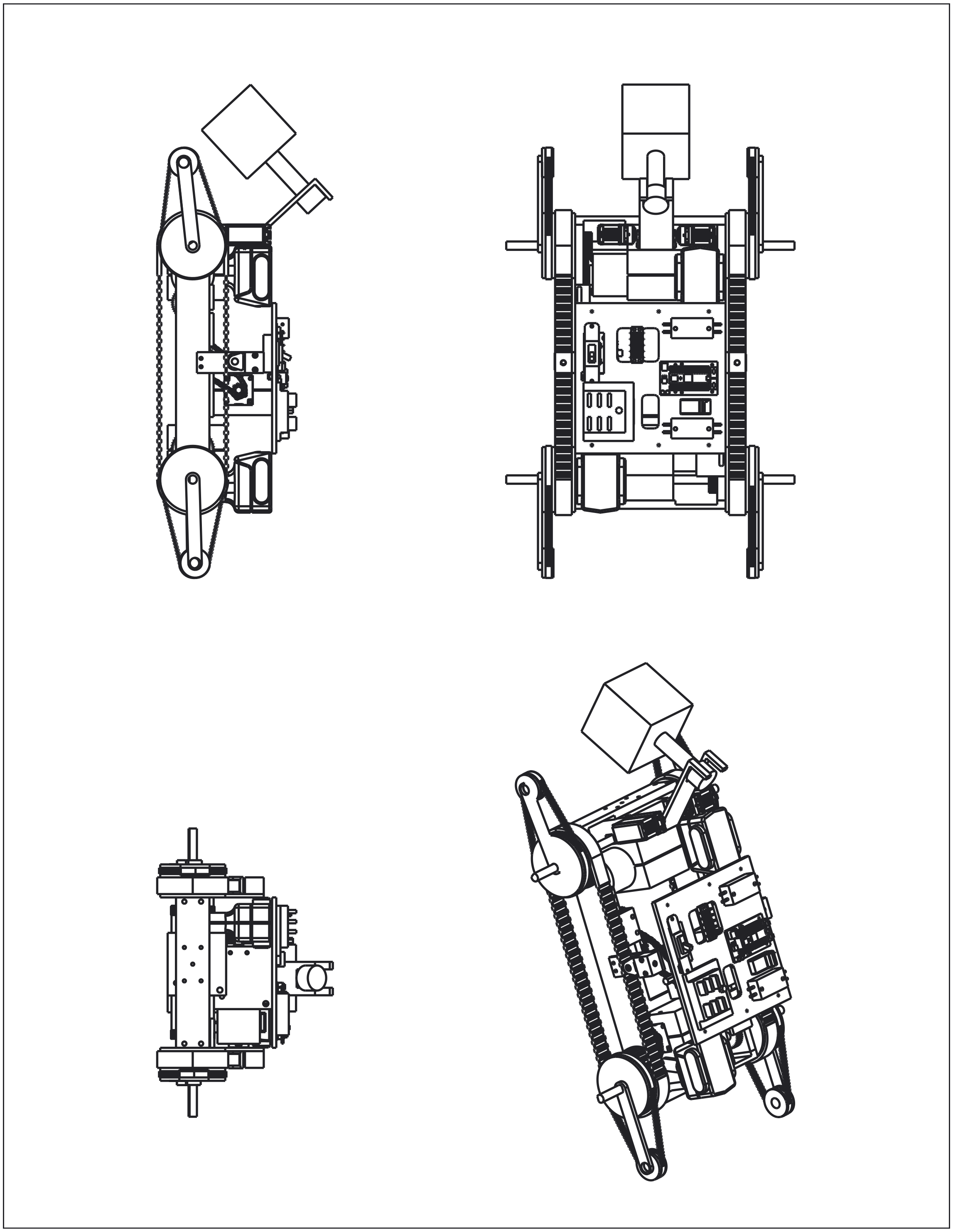
Caesar successfully retrieved the medical kit, autonomously navigated the curved chute, and used light sensing to drive to the victim and deposit the medical kit (on the day before final testing), but was unable to surmount the wall. Through the process of designing and constructing Caesar, we were able to understand many of the practical and aesthetic considerations required when creating a robot and this skill can be applied to any other design and construction projects that we may encounter in the future. We also learned various important elements of project management and how to work as a team to coalesce various tasks into one fully-functioning product, especially the importance of setting realistic SMART goals and focusing on planning at the onset of the project to avoid later conflict. Ultimately, designing and constructing Caesar taught the whole group the importance of both practical engineering and project management considerations.

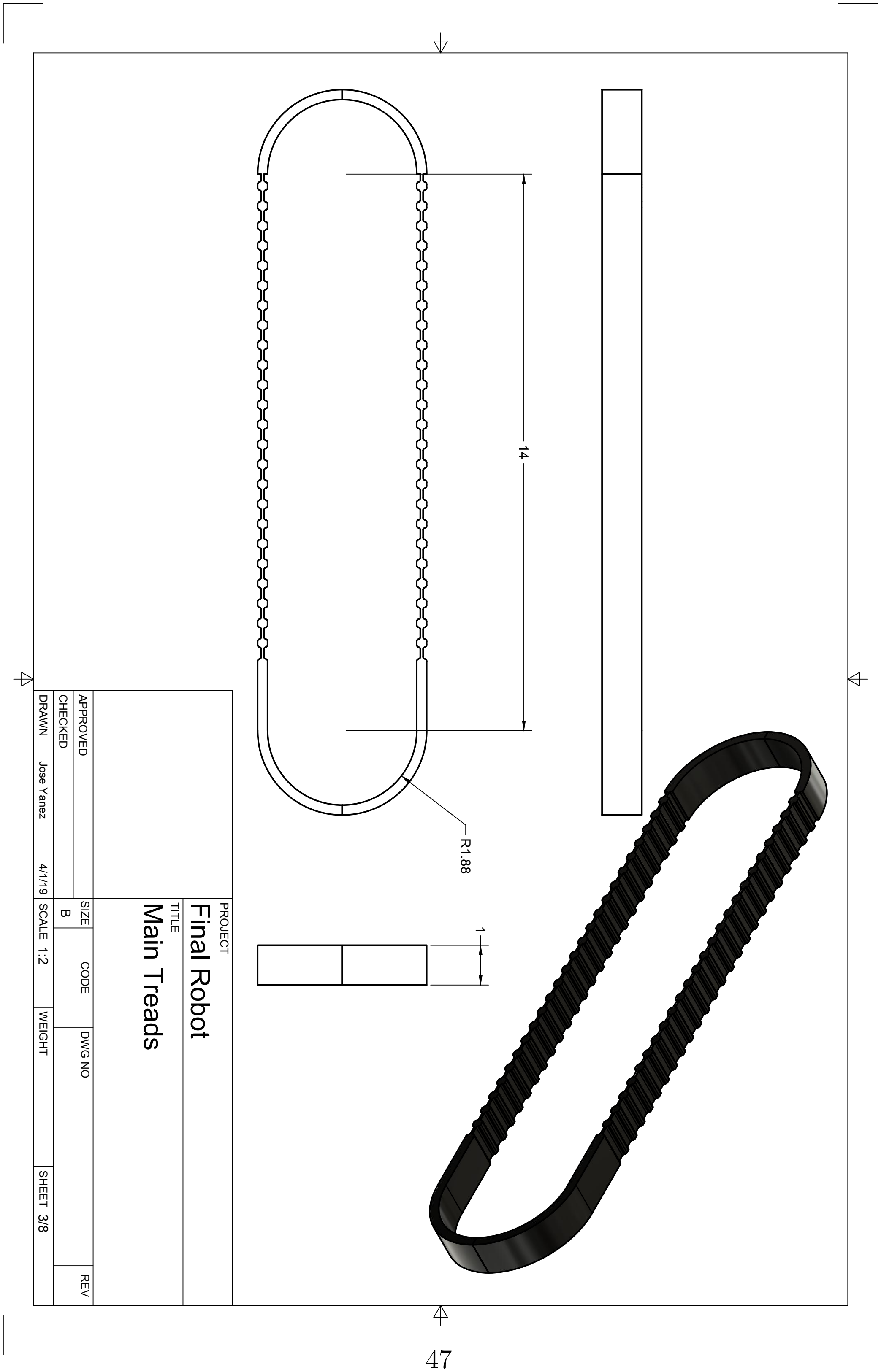
8 Design Drawings

The following pages present the critical components and sub-assemblies of the robot and their respective dimensional drawings.

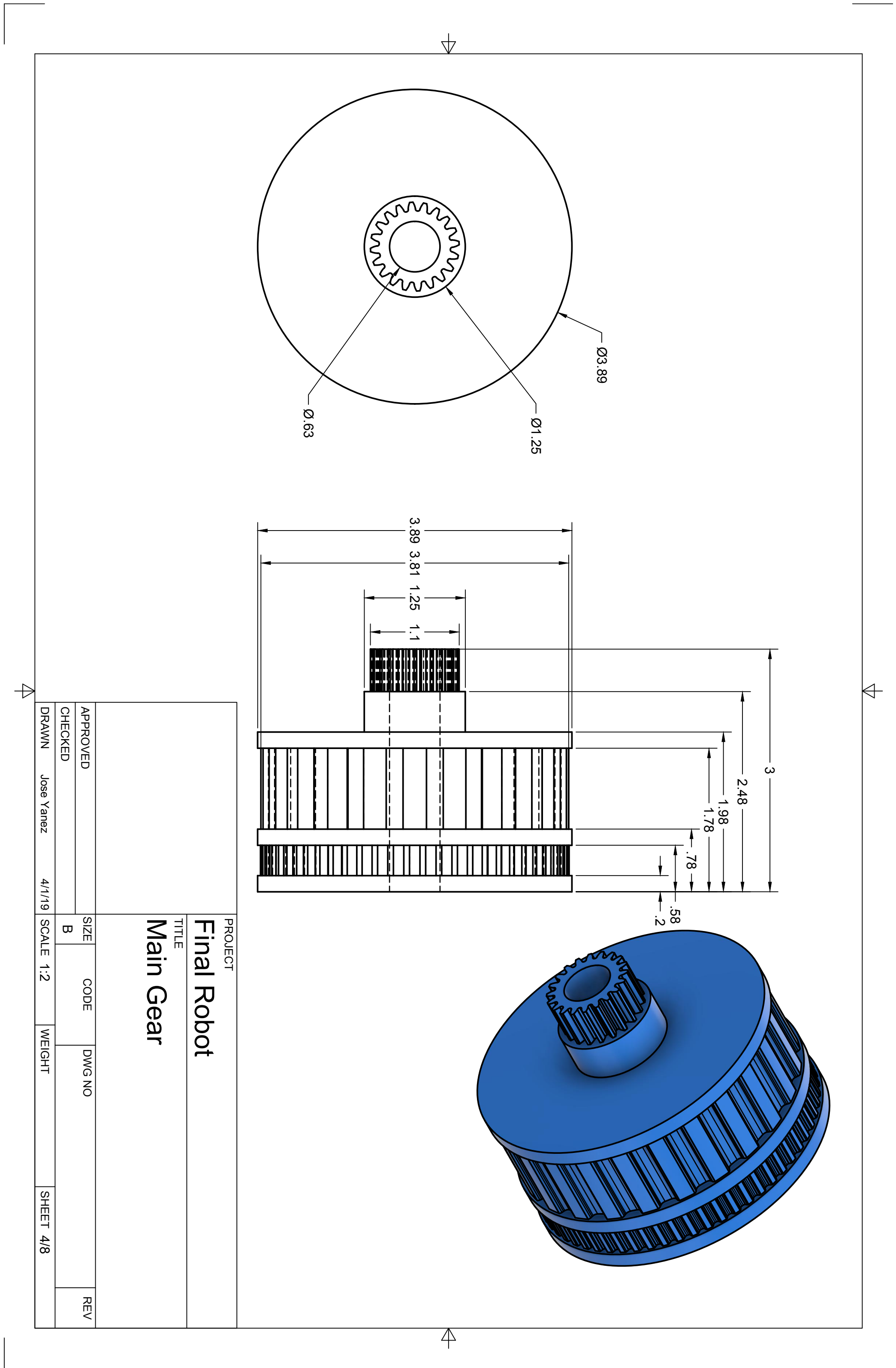
Caesar - Final Design

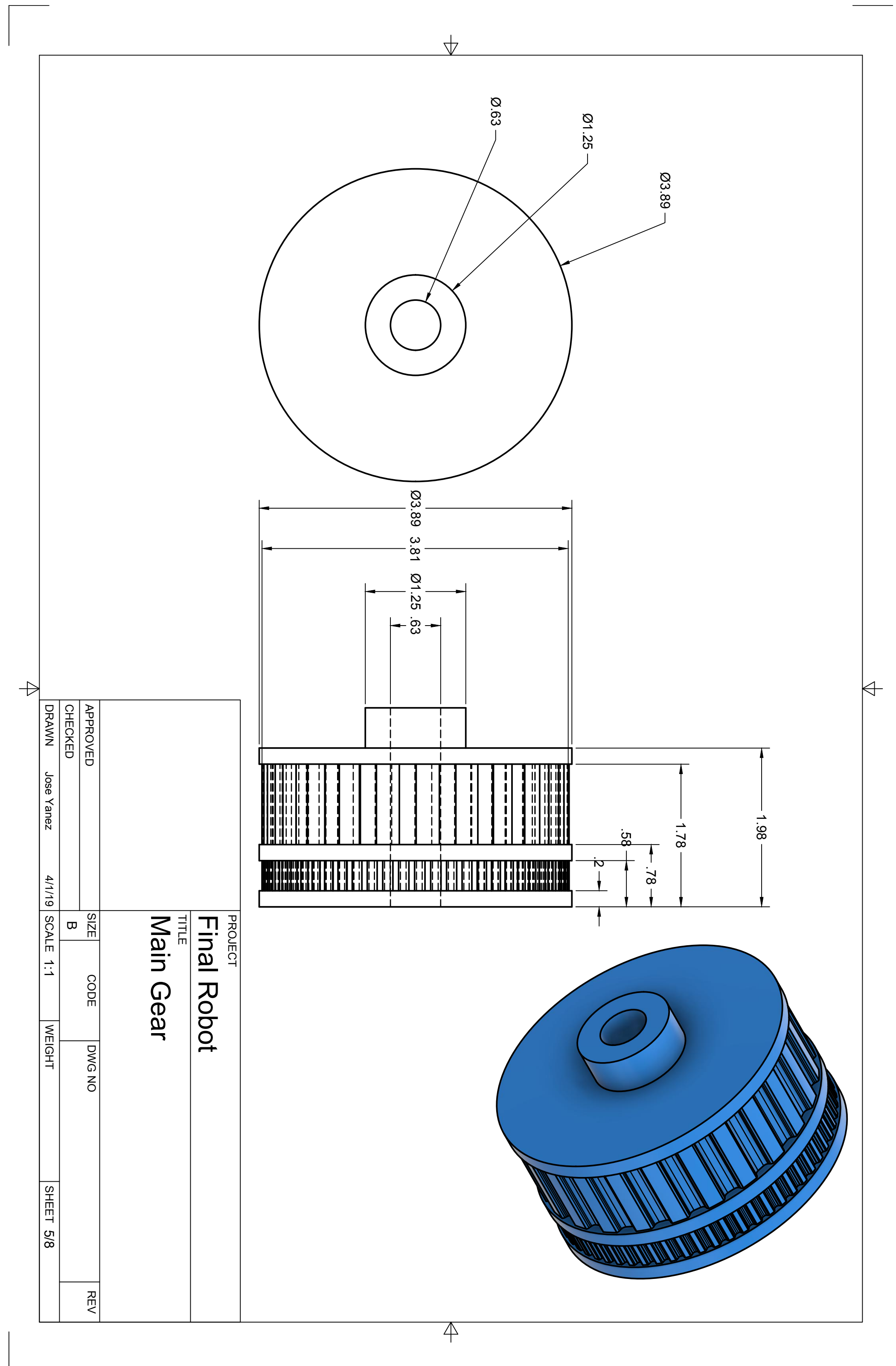


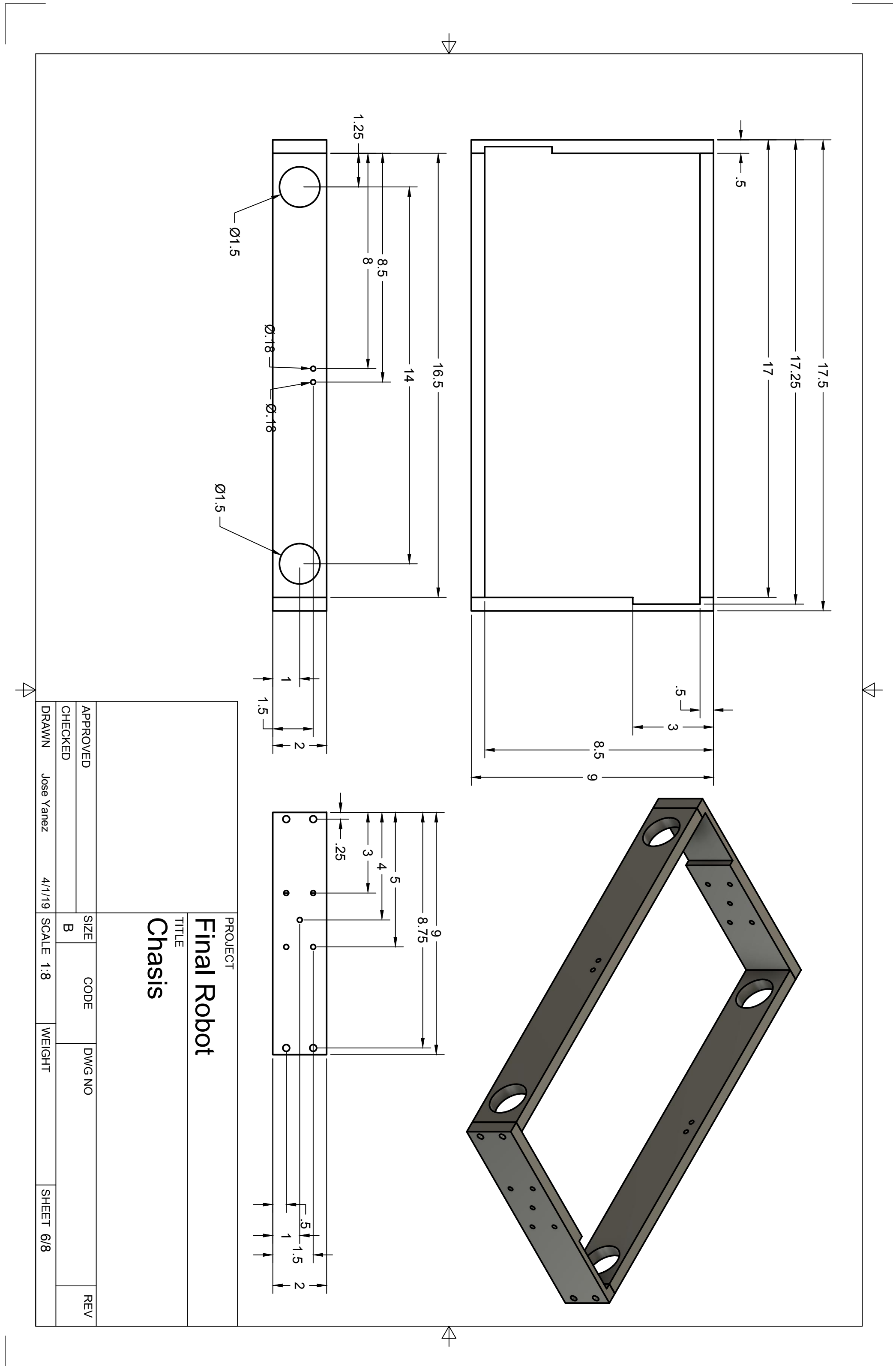


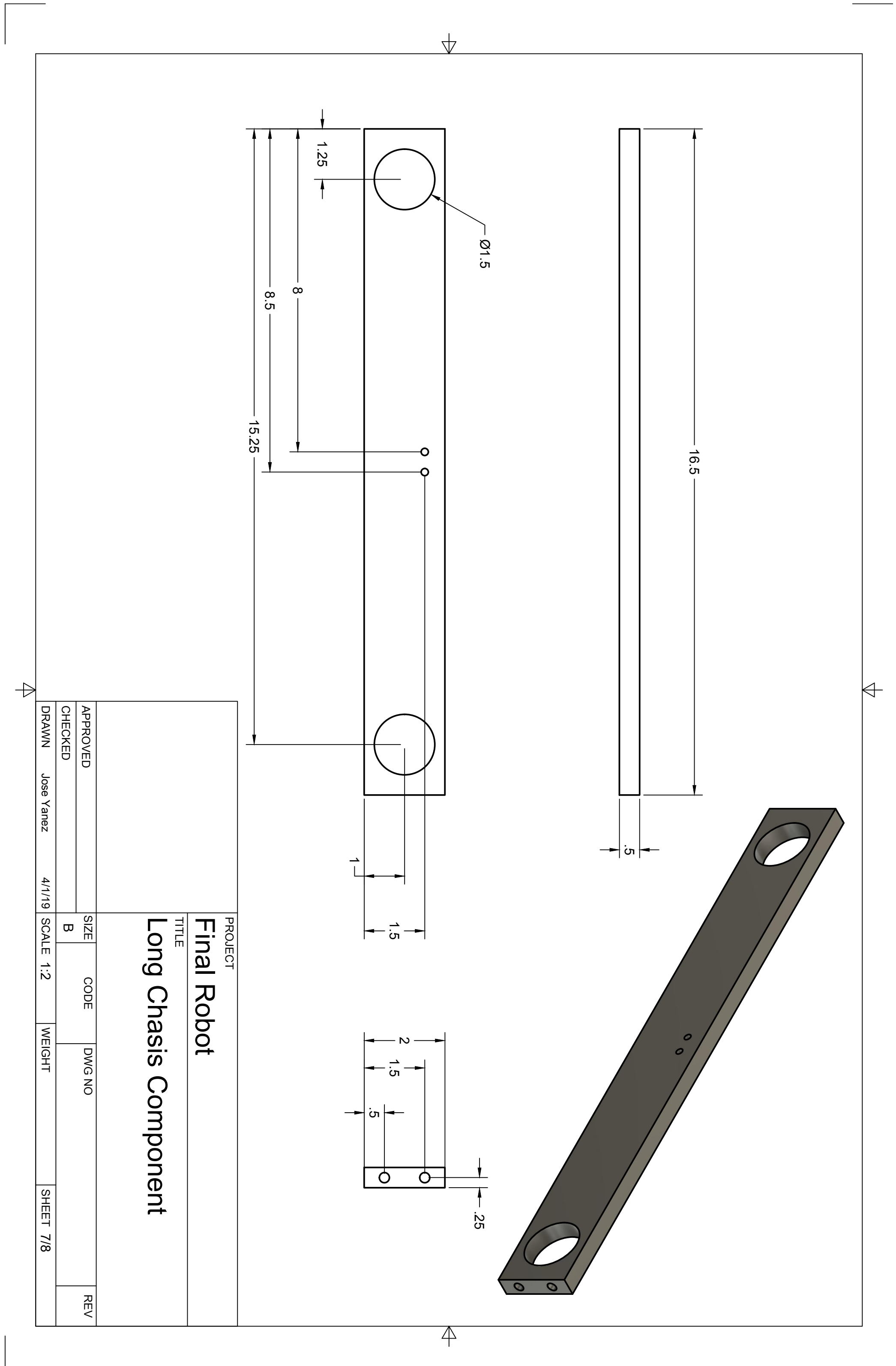


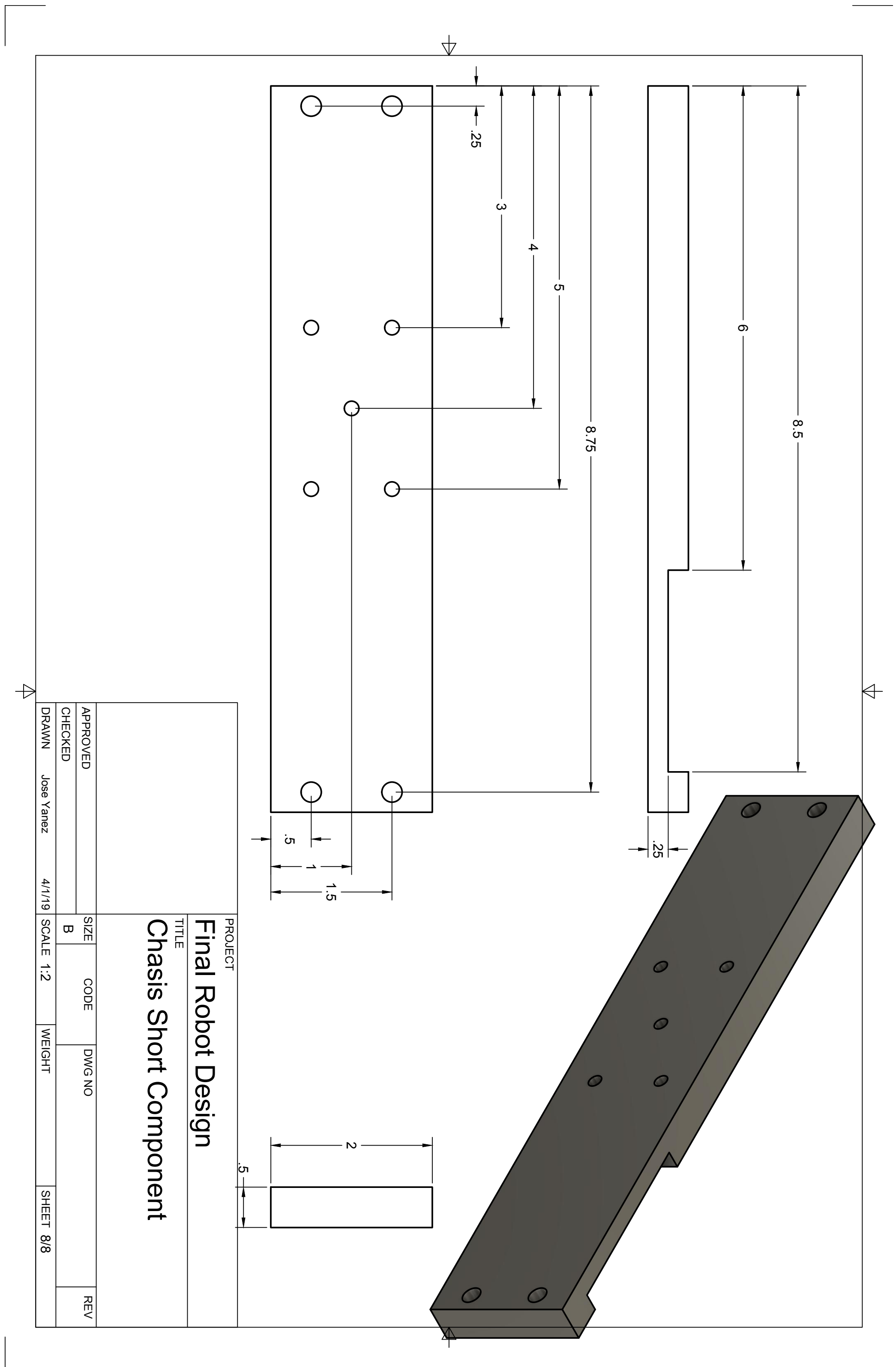
14
R1.88
1
PROJECT
Final Robot
TITLE
Main Treads
APPROVED
CHECKED
DRAWN Jose Yanez 4/1/19
SIZE B
CODE
DWG NO
REV
SCALE 1:2
WEIGHT
SHEET 3/8

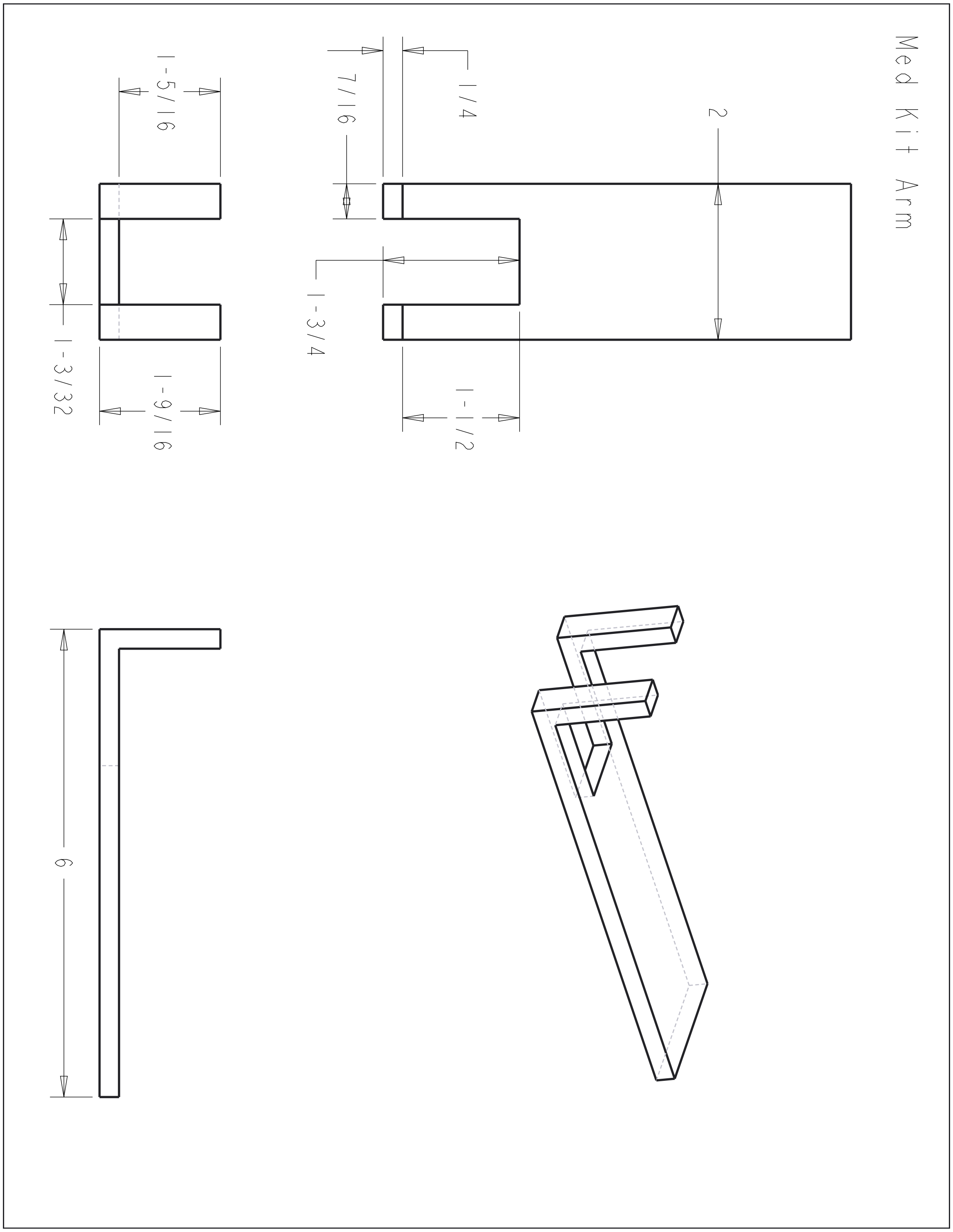

Ø3.89
Ø1.25
Ø.63
3.89
3.81
1.25
1.1
3
2.48
1.98
1.78
.78
.58
.2
PROJECT
Final Robot
TITLE
Main Gear
APPROVED
CHECKED
DRAWN Jose Yanez 4/1/19
SIZE B
CODE
DWG NO
REV
SCALE 1:2
WEIGHT
SHEET 4/8

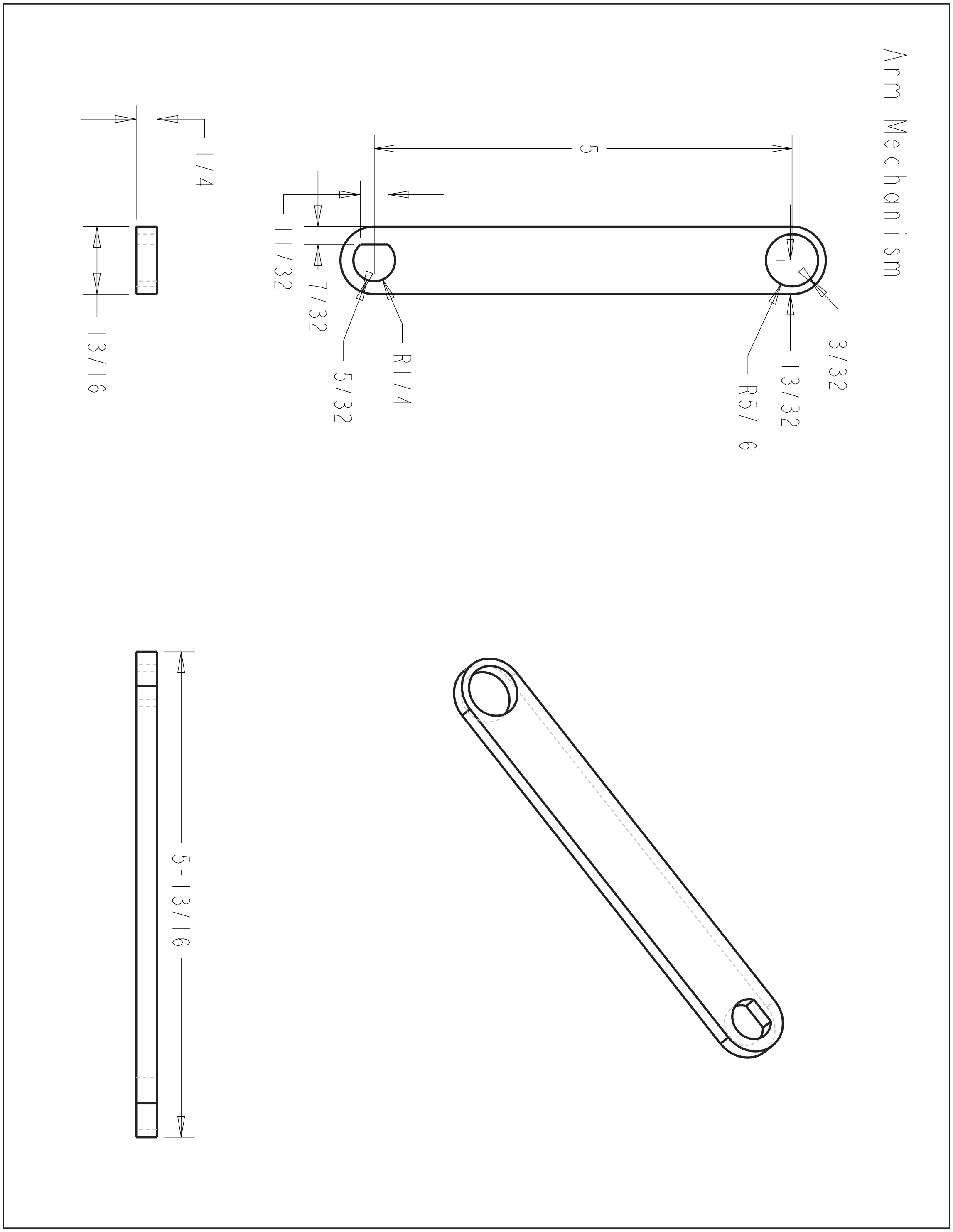

Ø.63
Ø1.25
Ø3.89
Ø3.89
3.81
Ø1.25
.63
1.98
1.78
.58
.78
.2
PROJECT
Final Robot
TITLE
Main Gear
APPROVED
CHECKED
DRAWN Jose Yanez 4/1/19
SIZE B
CODE
DWG NO
REV
SCALE 1:1
WEIGHT
SHEET 5/8

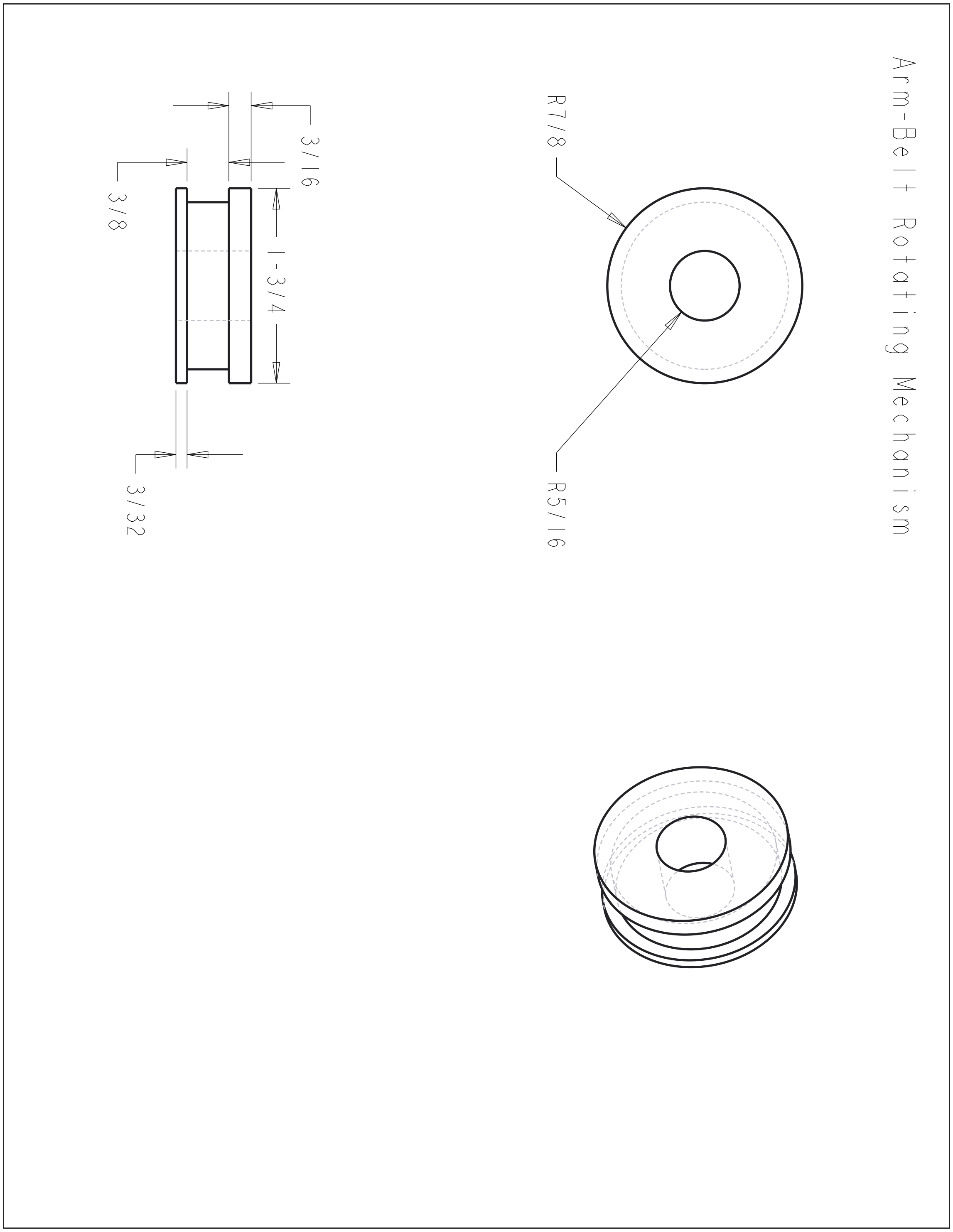

PROJECT
Final Robot
TITLE
Chasis
APPROVED
CHECKED
DRAWN
Jose Yanez
4/1/19
SIZE
B
CODE
DWG NO
REV
SCALE 1:8
WEIGHT
SHEET 6/8
Ø1.5
Ø.18
1.25
8
8.5
14
16.5
17
17.25
17.5
.5
3
8.5
9
1
1.5
2
.25
4
5
8.75

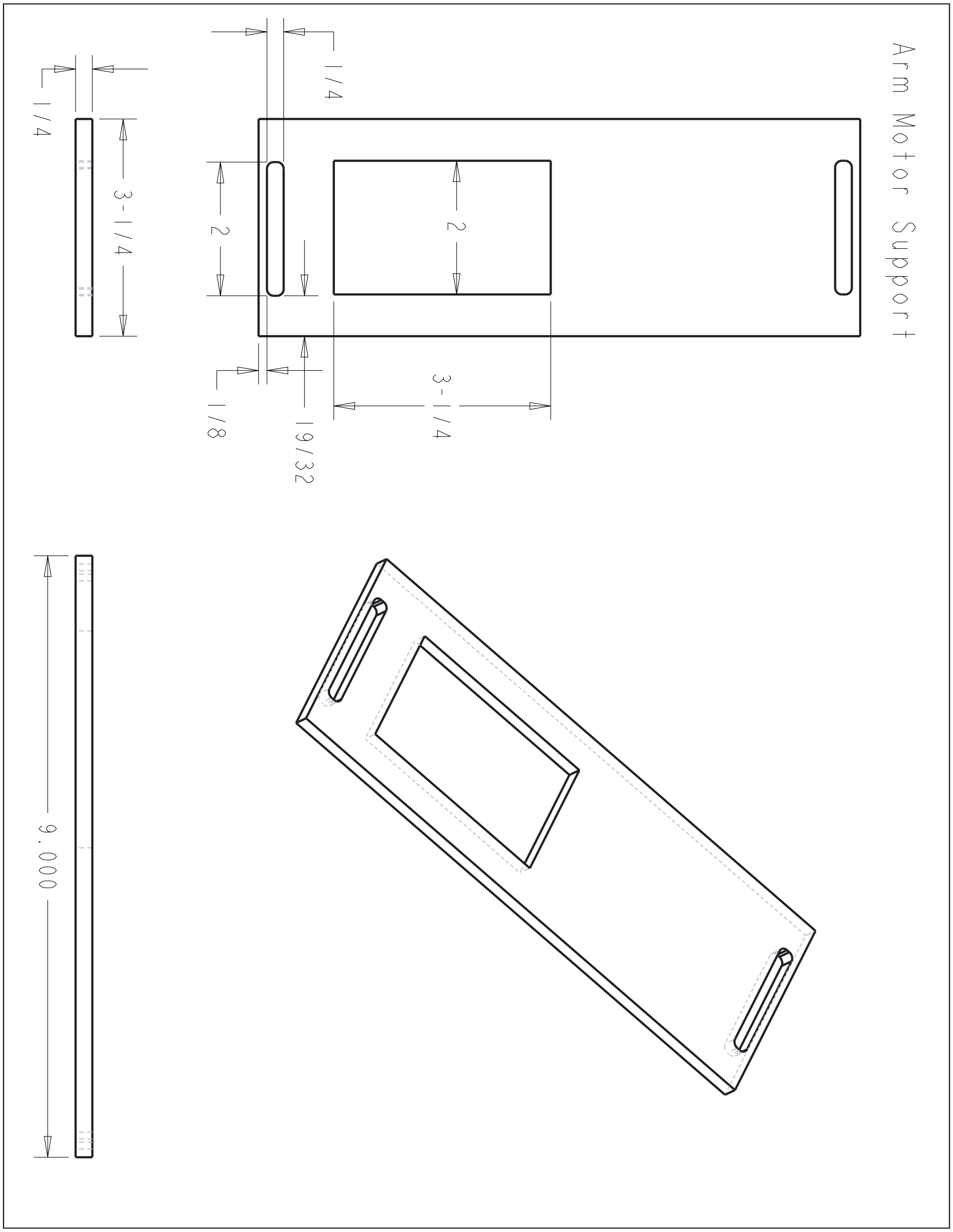

1.25
Ø1.5
8
8.5
15.25
16.5
1
1.5
.5
2
1.5
.5
.25
PROJECT
Final Robot
TITLE
Long Chasis Component
APPROVED
CHECKED
DRAWN Jose Yanez 4/1/19
SIZE B
CODE
DWG NO
REV
SCALE 1:2
WEIGHT
SHEET 7/8

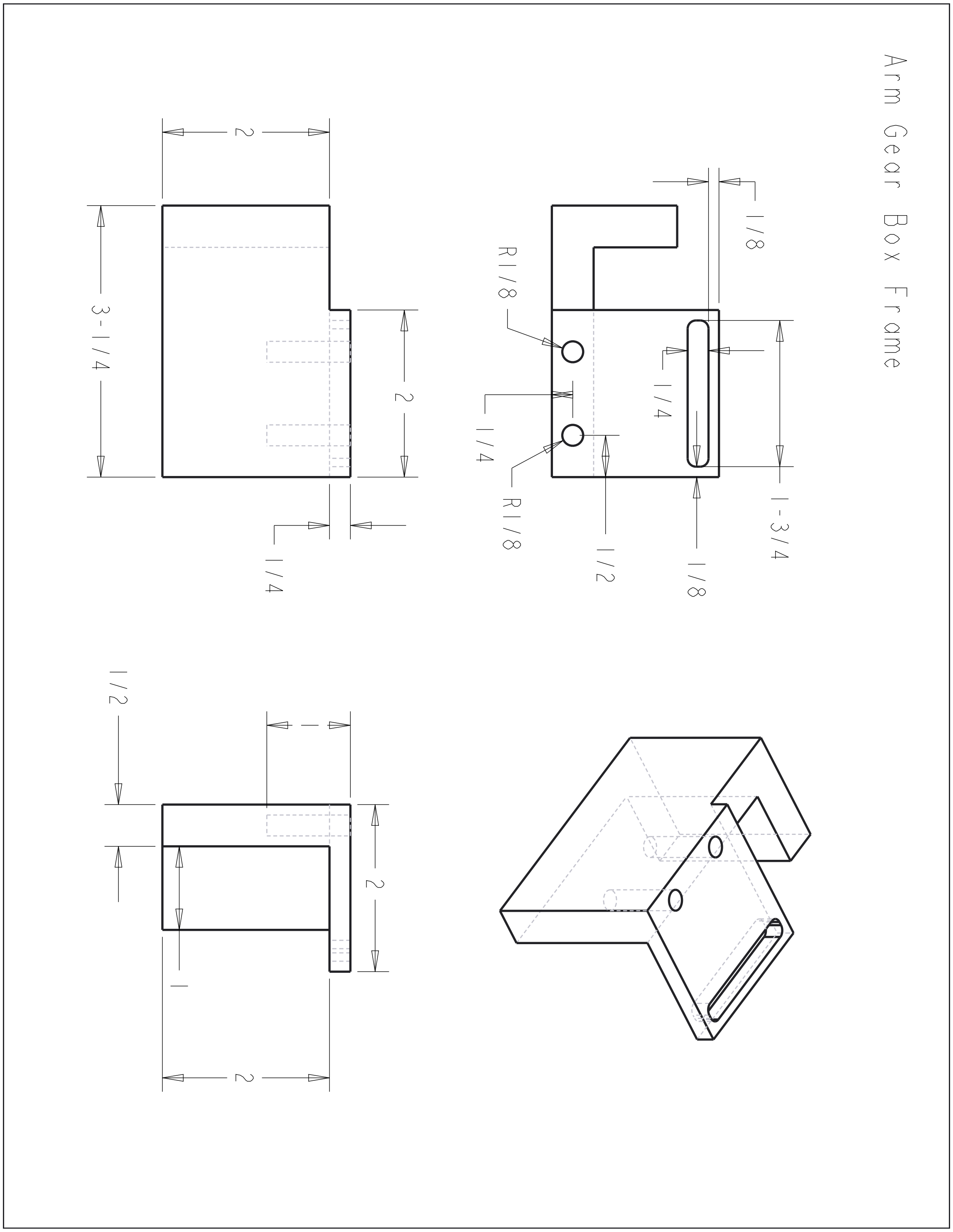

PROJECT
Final Robot Design
TITLE
Chasis Short Component
APPROVED
CHECKED
DRAWN
Jose Yanez
4/1/19
SIZE
B
CODE
DWG NO
REV
SCALE 1:2
WEIGHT
SHEET 8/8
.25
3
4
5
8.75
6
8.5
.5
1
1.5
.25
2
.5

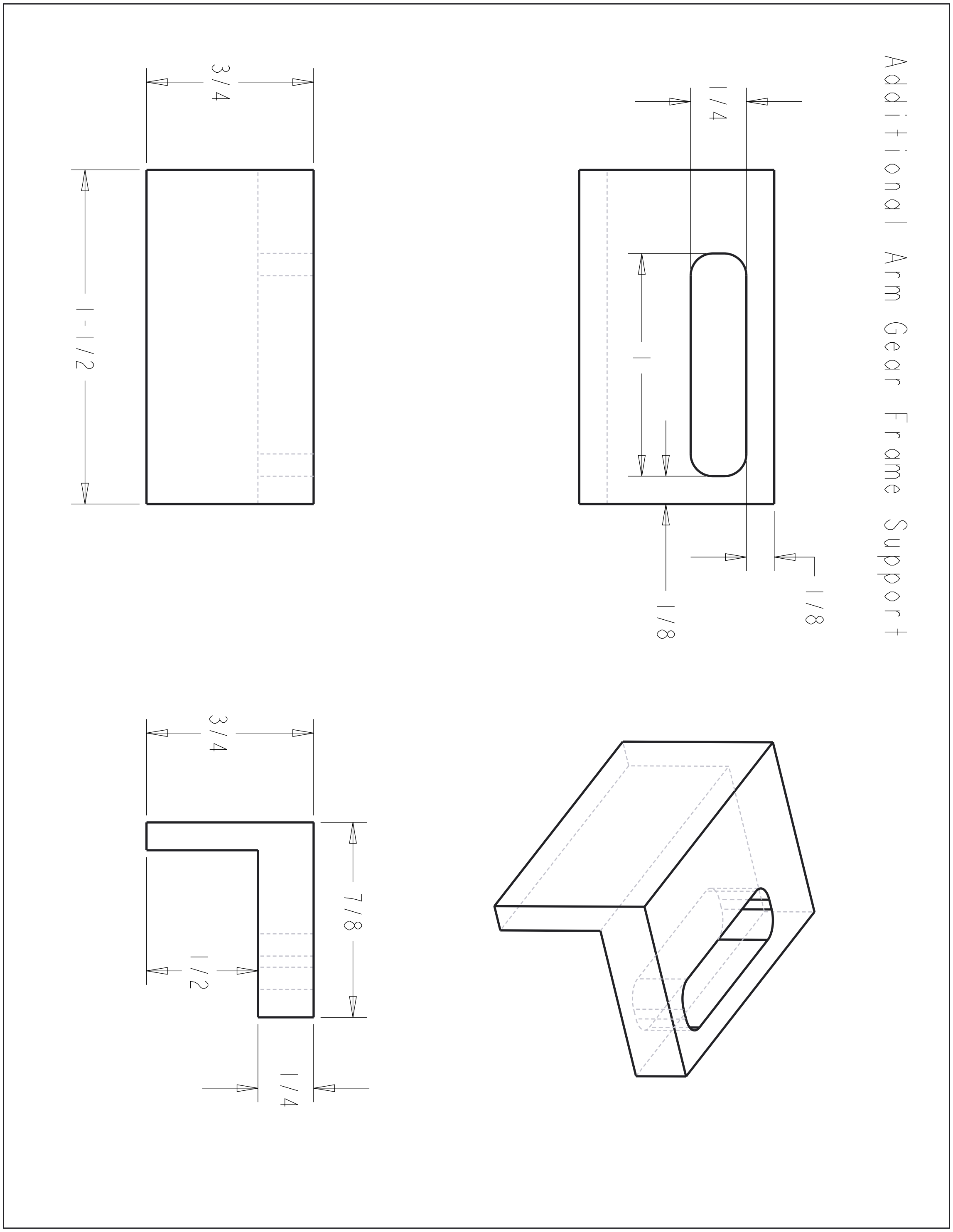

Med Kit Arm
2
1/4
7/16
1-5/16
1-3/4
1-1/2
1-3/32
1-9/16
6

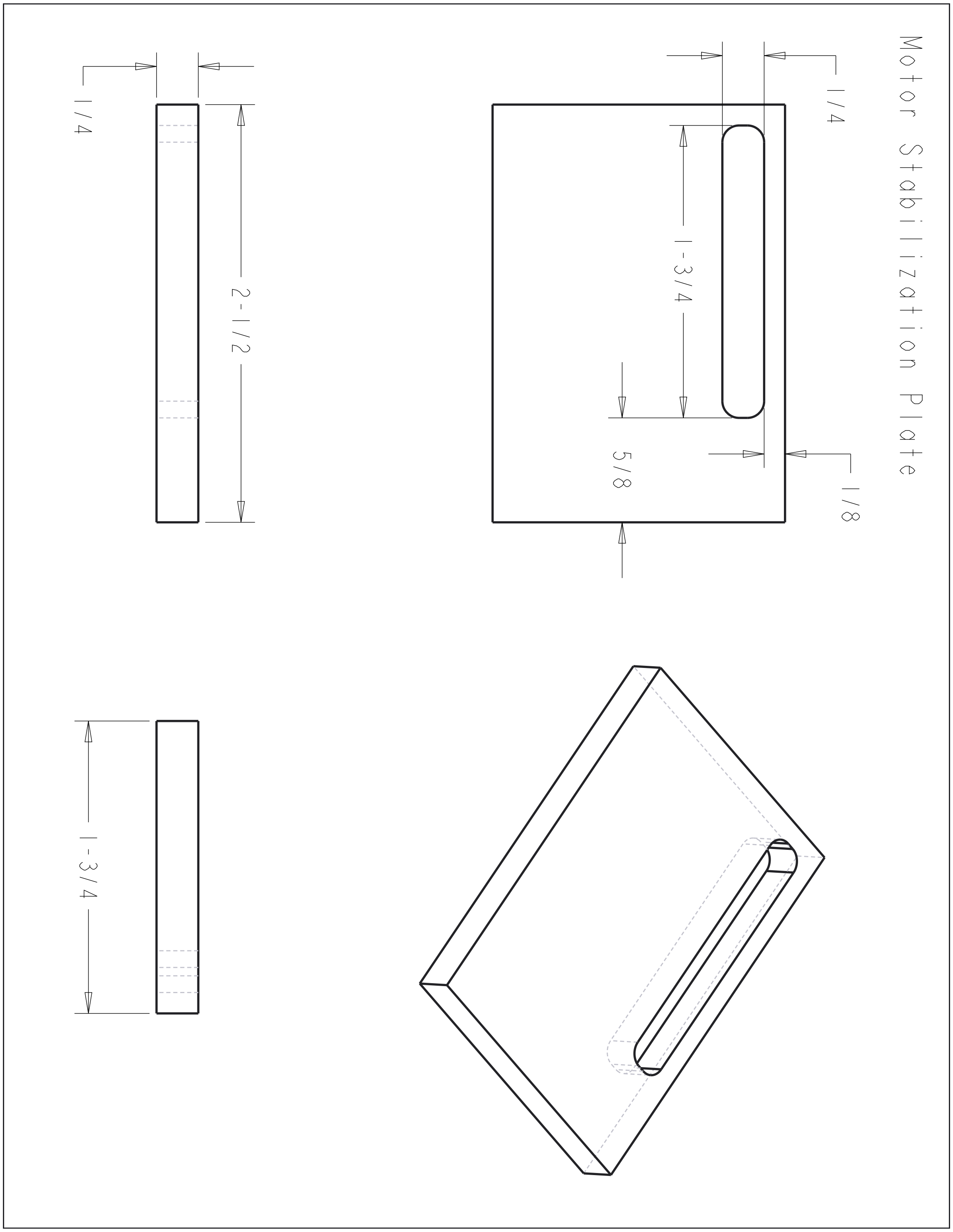

Arm Mechanism
5
1/4
11/32
7/32
5/32
R1/4
3/32
13/32
R5/16
13/16
5-13/16


Arm-Belt Rotating Mechanism
R7/8
R5/16
3/16
3/8
1-3/4
3/32


Arm Motor Support
1/4
1/4
3-1/4
2
2
1/8
19/32
3-1/4
9.000


Arm Gear Box Frame
2
3-1/4
2
1/4
1/8
R1/8
1/4
1/4
R1/8
1/2
1/8
1-3/4
1/2
1
2
1
2


Additional Arm Gear Frame Support
3/4
1-1/2
1/4
1
1/8
1/8
3/4
7/8
1/2
1/4


Motor Stabilization Plate
1/4
1-3/4
5/8
1/8
1/4
2-1/2
1-3/4