DESIGNING AND FABRICATING JARVIS – FOUR WHEELED SUMOBOT

ME 353 - MECHATRONICS

Authors – Evan Lutchmidat, Khushant Khurana, Scott Losirisup

MAY 12TH, 2023

Contents

INTRODUCTION

From the original sumo bot tournament founded in 1987 by Hiroshi Nozawa, the robot sumo competition has not only resided in Japan but also expanded all over the world. The competition first moved to the United States for the first time in 1998 in the “Robogames” held in San Francisco, California. In 2019, more than 30 countries have participated in the tournament since 1987. All in all, the competition is well known around the world and is popular enough to be held in the Cooper Union “End of Year Show”.

With this said, Sumo-bots are high power and wheeled robots designed for competing in the robot-sumo contest. The goal of the competition is to push the opponent’s robot outside a 3ft-by-3ft ring made of neoprene rubber. The team is tasked with designing and fabricating a 10 inch by 10 inch by 6-inch robot with a weight limit of no more than 5 lbs. The exact rule set is given in the appendix.

The team is also given a cost limit of $200, and the use of liquids, compressed gasses, gels, and hazardous substances is disallowed for design. Moreover, no electrical – lasers- or mechanical – sharp edges or rotating blades- weapons are permitted on the battlefield. To ensure the safety of electrical components powered by a battery, specifications are provided for the maximum amperage for motors – 4.5 amps - and maximum power supply of the battery – 14 V. Finally, the team is asked to design for a kill switch in times of emergency. To plan the design process, the team breaks down the timeline into sub parts including – brainstorming, preliminary sketches, final designs, finding mechanical parts, making computer-aided designs, fabricating, finding electrical components, and finalizing the assembly.

STRATEGY/DESIGN OVERVIEW

The entire purpose of the Sumo Bot competition can be classified under the two categories of edge detection and pushing the other robot out of the ring. While the first solely relies on the placement and tuning of IR sensors used, the latter depends on many factors such as: torque generated by the motors, use of wedge, friction between the tires and the mat and body design. In fact, these design specs can be aligned with the two overarching strategies of the robot as described below.

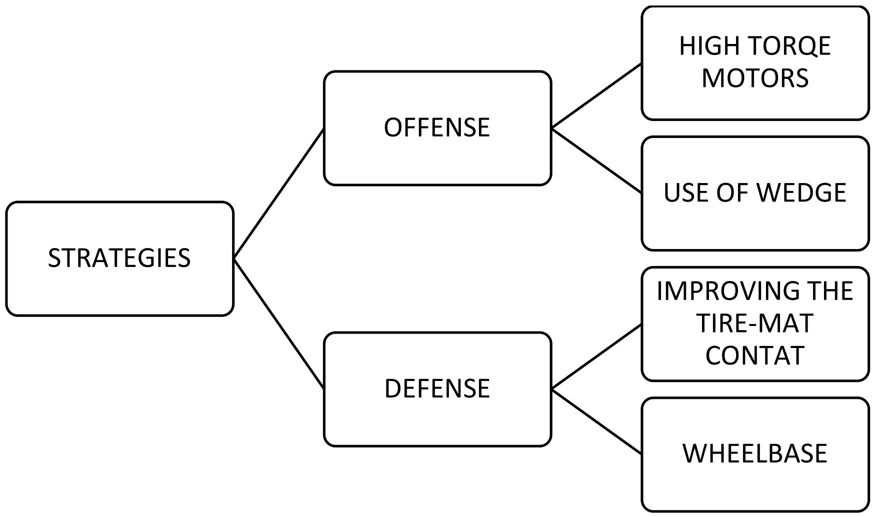


Figure 1. Flow tree diagram of strategy

Starting with the motors, the team decides to follow the strategy of high torque and low rotational speed since winning the battle heavily relies on the bot's sole power of pushing and not moving fast on the mat. Accordingly, the team initiates its research by looking for motors that match the ruleset and the team's priorities. As explained in the appendix, the ruleset requires to use motors with stall current of no more than 4.5 amps.

Despite playing a big role in generating power to push other bots, the motors fail to accomplish this if the wheels lose contact with the ground. Accordingly, the team decides to use this strategy to ensure its success. Moreover, the team designs JARVIS to have wedges in all four sides. Since no ultrasonic sensors or other distance sensors are used in JARVIS, it becomes for the bot's front to stay in line with the opponent and hence having a wedge only in the front might backfire if the other bot attacks from other sides. Accordingly, by having wedge in all four corners, the team ensures that JARVIS is not only protected but it can also attack from other sides if the need arises.

The robot has been designed with a higher gearing ratio to provide low RPM and high torque, which is advantageous in the arena. With this design, our robot can hit other robots and out push them with brute force. Additionally, the robot is protected on all four sides by essentially wedged walls, providing no blind spots to be attacked from. Even if hit from the back, our robot is still able to gain leverage on opposing robots. Furthermore, our robot is capable of reversing direction such that the back of the robot can become the front and vice versa, providing a tactical advantage in the arena. To further enhance its capabilities, the robot has been equipped with Teflon mini wedges that can ride the floor and get underneath every robot. The robot's chassis has been designed to be strong to withstand any impact in the arena Thus, in battle, structural integrity should not be a question and losing or winning due to breaking should not be a probable outcome.

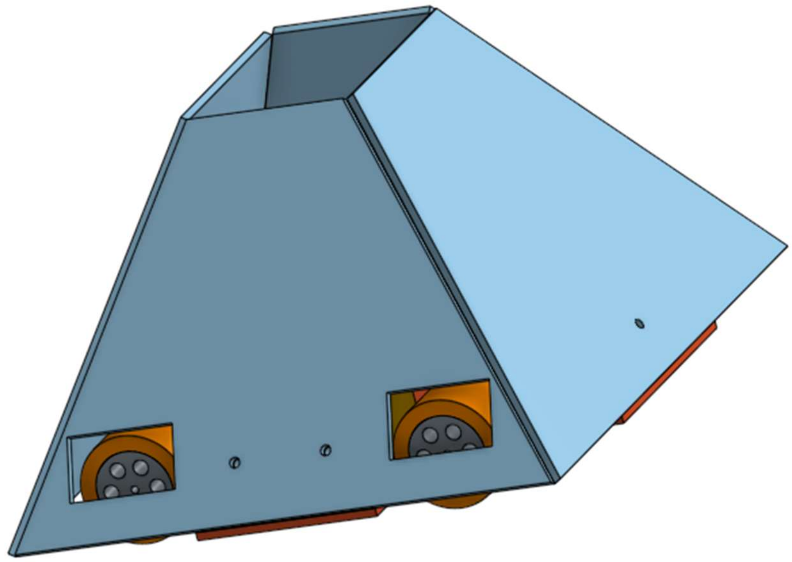


Figure 2. View of fully assembled JARVIS without wedges

Initially, the offensive plan was to design the robot to be self-aware of the arena and to understand its position on the ring. The plan was to dimension the arena then break it into a grid, with many squares in which would be a coordinate, essentially creating a local positioning system which could be interpreted in code as a Matrix with the unique matrix element positions being a coordinate or position to move to. This is where the original Matrix name derives from. However, it was unfortunately revealed that this concept may not be within the scope of this project for the course with the lack of basic knowledge for digital electronics and design and scratch C coding. With this strategy, the idea was to be agile and quick, to maneuver around the opponent and blindside them from the side such that the opposing robots' wheels lose traction getting stuck on the wedge of the team's robot and being pushed off of the mat. Essentially, the robot would run to an edge to determine where it truly is on the grid since it is a rectangle, then relocate to the middle of the grid. Once in the middle, the robot would continuously spin in place until the opposing robot was detected and then chase after the opponent and ram into them. An alternative concept with the matrix grid of the arena would be to bait the opposing robot to the edge and follow up by then attacking them after sensing them using a sensor such as the light detection and ranging sensor (LIDAR) to detect the location of them and sneak up on the opponent while they were closer to the edge. This idea would be more beneficial for this design since it would've been designed for speed and minimal torque thus minimal pushing power.

After researching JARVIS's offense with the constraint of the team's knowledge and capabilities, the team decides to shift its focus on defense. Since it is very likely that opponent bot's motors have higher torque as JARVIS's, the team ensures its success by focusing on the contact between the robot and mat – wheels. Having sticky wheels can help the bot generate a lot of traction which prevents the wheels from slipping. The force of traction comes directly from the friction factor between the rubber and the mat. After researching through various candidates, the team finds silicon to generate to the highest traction due to its friction coefficient. Besides the rubber, the wheel size also factors a lot in how much traction a bot can generate. Bigger wheels allow for more contact area and hence the bot can grip to the mat better. Accordingly, the team decides to use large wheels. The larger the wheels, also imply higher top speed although this is limited by the motor speed as well.

Although, traction can help the bot a lot. It can also backfire if the opponent bot can get under JARVIS. In that scenario, the high traction wheels can act as a pivot point and make the bot tip over. To prevent this from happening, the team decides to use a large wheelbase which can prevent high torques generated laterally and accordingly prevent the robot from tipping over.

With the newly designed approach, there still must be an offensive plan. For JARVIS, the plan is to now play with strength and torque, which is rather opposite of what the original plan was. Strength is rather important considering the name of the entire project is sumo robots, and sumo wrestlers are all about strength. Instead of the usage of LIDAR sensors now, the team decides ultrasonic sensors are in place of them and used only in conjunction with the infrared photo reflectors. As for gameplan, this refined version of offense more or less includes just staying within the ring and winning any encounters the robot comes across through being stronger than others. For this goal, the initial motor speed that was considered was 485 revolutions per minute which is in the middle of torque and speed however the team would commit to the torque and take the next step up to 170 revolutions per minute to attempt to out muscle all competitors. To add onto the concept of being stronger, the team decided that having more than two motors would be ideal and to be specific, having four motors may be the answer to their desire for strength. Each being of 170 revolutions per minute implies the robot is already strong as is. As the amount of force, the robot can put down depends on the friction the wheel has with the ground, it also depends on the amount of torque the robot can get on the ground. Having four sources of torque versus two is significantly better and provides more pushing power. The downside to choosing four motors with no sort of axle or steering system in this robot is the turning and turning radius. Just like a rear wheel drive car versus an all-wheel drive car, the rear wheel drive car will have superior mobility since the front wheels are free however just like the all-wheel drive car there is more applied force in contact with the ground. For a lower center of gravity, all electronics were to be placed on the top layer since light and easily accessible and the battery is intended to be placed in the middle of the bottom layer.

MECHANICAL DESIGN

There are two versions of the robot. Version one aka, “Jarvis_1” and the final version two, “JARVIS” With this said, Jarvis provided a solid foundation as it had multiple faults. These problems were fixed in its later and final iteration, “JARVIS”.

Iteration 1 of Jarvis:

For the first time attempting to fabricate all parts and assemble, it was rather smooth. All the CAD had been finished and everything lined up near perfectly. In this assembly the team had:

- Side Supports 1 V1 (with 2 holes on side), 3D printed.
- Side Supports 2 V1 (with 1 hole on side), 3D printed.
- Wheel Mold V1, 3D printed.
- Top layer of chassis V1 (6x6.5), 3D printed.
- Bottom layer of chassis V1 (8x8), laser cut wood.
- Motor Mounts V1 (30 mm total height), 3D printed.
- 6-32 hardware for assembly; M3 for wheel mold

Assembling it, the team used 2” 6-32 screws in place of the standoffs the team intended on using for two reasons, one being that the team didn’t have the standoffs available now and two being the team didn’t account for the outer diameter thickness of the standoff in the support such that the standoff would not be able to fit through. Now let’s address each component and its design and issues they posed.

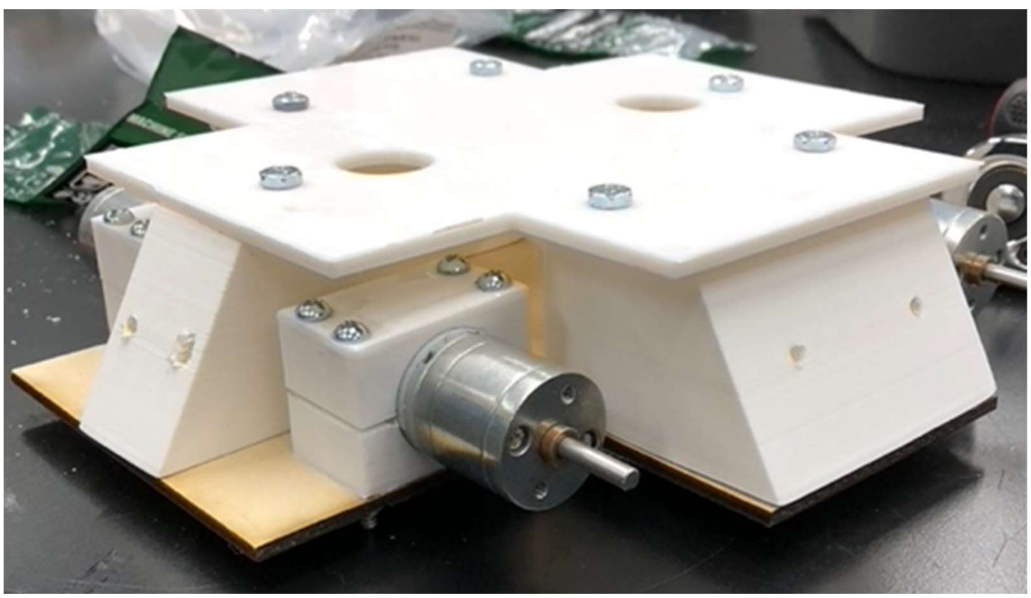


Figure 3. Isometric view of fully assembled first iteration of Jarvis without wheels or electronics

Side Supports 1 (V1)

This support was near perfect for its purpose the team had designated for it. It was an extruded right triangle, dimensioned 1.5 inches high, 2 inches long and 0.75 inch wide. This piece had 4 total holes, 2 on the top which cut fully through the entire piece from top to bottom such that the standoffs can go through it, and the other two on the slant face and went about 5 mm deep. This hole was a blind hole as it was intended to be tapped and serve as threads to hold the side wall. The first issue that arose with this version of the side support was the wall thickness of the holes for the 3D print were too thin. They were printed to be about 1 mm thick, which could not withstand being tapped. The next issue was the hole size for the standoffs. All holes on this design were made to be M3 compatible as that was the initial plan for hardware however the team ended up upsizing to 6-32 since it was more conveniently available. Even then however, the team did not account for the outside diameter of the standoffs which were about 7 mm thick. This version was additionally printed with PLA and low infill with a concern of the entire robot surpassing 5 lbs. The infill percent on this robot was 20%.

Side Support 2 (V1)

This support was flawed in many aspects. It was designed to be smaller than the other supports as it would be on the axial side of the motor. In short, the motors needed that extra space on the chassis so they did not poke out very much. This support was similarly designed in concept to support 1, whereas there was 1 hole on top that cut fully through to the bottom and 1 blind hole on the slant face that was to be tapped to hold the side wall. This support suffered the same issues as support 1 however the team additionally forgot to extrude the hole fully through from top to bottom and in turn failed to be able to have a standoff go through the support for this reason. This support was also printed with PLA and 20% infill.

Wheel Mold (V1)

Wheels were an interesting development for the team due to the inaccessibility to sumo robot wheels from a third party such as JSumo. The wheels that came pre-made offered convenience for hubs and time, however they would not be as fitting as wheels designed to the CAD file. Additionally, self-made wheels offered many benefits of material and softness choice, size, and even thickness of silicon/rubber. Thus, making the wheels by hand was this team's decision. While it had the right concept, wheel mold V1 was a failure. It had a hollow cylindrical design for the motor to be able to slide into and 4 M3 holes on the face to mount to the motor. This mold failed as the mounting plan for the motor was wrong and the silicone would not be able to properly attach to the proposed hub for the wheels.

Top Layer of Chassis (V1)

This layer was made to hold the breadboard and organize electronics. It also served as a second way to stabilize the support. It was sized 6" by 6.5" and 1/8" thick. Initially the idea was to also have support on this level as well to hold the acrylic walls however that fell through with the understanding that the bottom layer supports were more than enough. The first issue that arose with this version was the sizing. The breadboard the team are using is exactly 6.5" long, meaning it just barely fit on the top. Additionally, when fully assembled, the nuts holding the standoff screws were interfering with the placement of the breadboard and thus it could not rest flat. It also did not offer a way to mount the breadboard. The second issue were the sides. Since the length was the same length as the breadboard, when it came to lay the acrylic down, the wall would interfere with the breadboard and the edge of the layer and the only way around this would be to cut the breadboard, which wasn't feasible. The team was also under the conclusion that the team did not want to make any extraneous cuts into the walls. This layer also had little circular holes for wires to be able to pass through the top layer to the bottom.

Bottom Layer of Chassis (V1)

This layer was designed to be 8” by 8” however ended up becoming 8” by 8.5” to compensate for the elongated top layer. With this setup between top and bottom layer, the robot would be able to have the walls be placed at an angle of 57 degrees with respect to the ground, which was perfect since the team’s goal was to be around 60 degrees. The purpose in choosing 60 degrees was so the team could have an angle still capable of getting underneath the robot while also reducing the normal force a robot could get down while on top of the team’s robot if they were to climb up. The bottom layer had 18 holes in total, 12 for motor mounts and 6 for side supports. This version in specific was built to be a mockup so it was a laser cut piece of wood. The final would be made out of acrylic. This layer also had all M3 sized holes.

Motor Mounts (V1)

These motor mounts were simple in concept. It was two bridge shaped components in which you tighten one side then use the other to clamp the motor in between. It consisted of 4 total holes for connection and mounting to the bottom layer of the chassis itself. For this version, the team ended up tapping the holes in the mounts 6-32. The first problem posed with this version was the fact that the team did not account for tolerancing in the 3D print and thus the motor mounts did not actually clamp down the motors but instead left free space for the motors to slide in and out. To resolve this, the team filed down one side of each mount which helped it to achieve its function. These mounts were determined to be too bulky in the end and too tall. Additionally, another lesson was learned with this piece: no threads on threads.

Assembly

In regards to the assembly of this version of Jarvis, the team was able to successfully get everything bolted up and tightened. The standoff fasteners were inserted from bottom up such that the head of the screw was on the bottom of the chassis (as depicted below in figure 1.1). These were flat head screws, 6-32 and 2” long. They went through the chassis completely, from the bottom layer, through the support, then through the top layer in which it was held in place by a nut. One concern about this was that there were not enough threads for the nut to properly hold on and thus with enough vibration it could fall loose. Next, the motor mounts were held in place by 6-32 Phillips head screws that were 1.5” long. They were inserted from top down and threaded through each motor mount. In doing so, the team realized four fasteners for the motor mounts was overkill and it could be reduced to just two per motor mount. On the bottom, a nut was additionally used to ensure it would remain in place. The back of each motor was lined up with the back of the motor mounts when held in place. That completes version one assembly of Jarvis.

Iteration 2: JARVIS

When assembling the final version of the robot, “JARVIS”, the team identified all errors with Jarvis and applied the changes to this final version. For this version of the robot, the team made:

- Side Supports 1 V2 (with 2 holes on side), 3D printed.
- Side Supports 2 V2 (with no hole on side), 3D printed.
- Wheel Mold V2, 3D printed.
- Wheel Hubs, machined out of steel using 10-32 hardware.
- Top layer of chassis V2 (6.63x7.77), 3D printed.

- Bottom layer of chassis V2 (8.5x9.6), laser cut wood
- Motor Mounts V2 (30 mm total height), 3D printed
- 6-32 hardware for assembly; M3 for wheel mold

*note: all pictures and visuals of components as well as supporting captions can be found in Appendix II

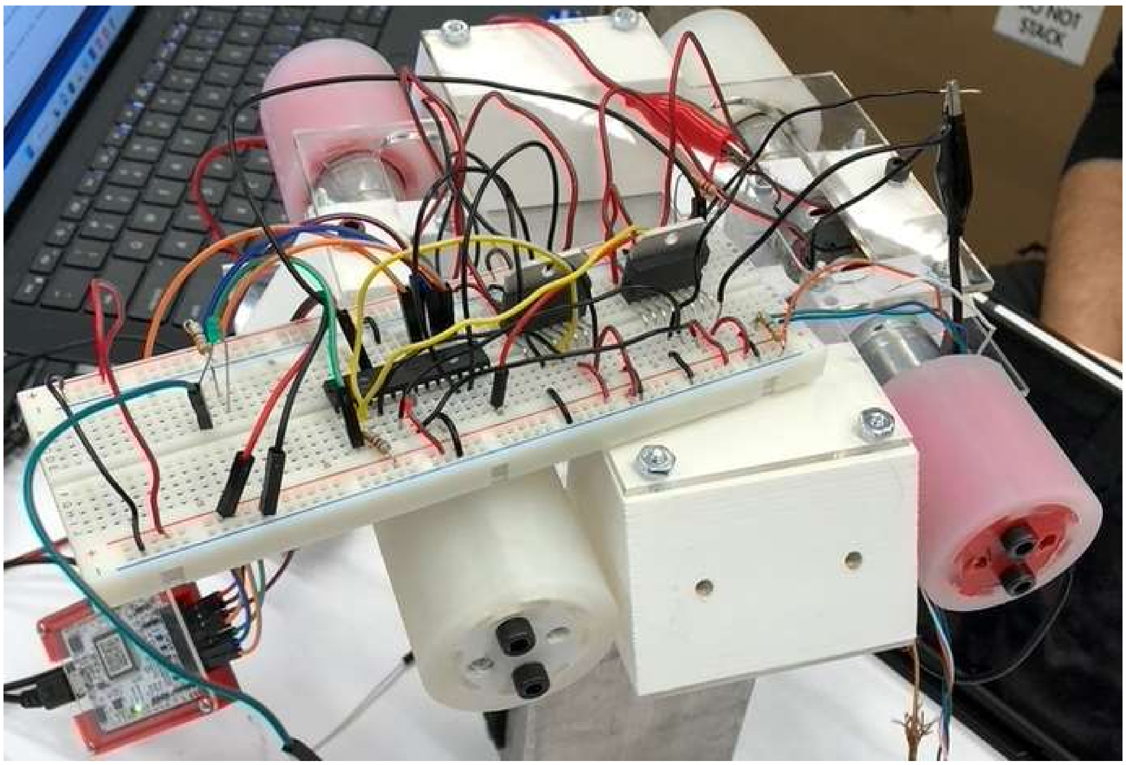


Figure 4. Full assembled iteration two of JARVIS without walls and wedges

Side Supports 1 (V2)

For the second iteration for the first side support, the team redesigned the shape so it would match the new lower and upper chassis that will later be discussed. The new design is wider than the last iteration with its dimensions now being 1.57" high, 2.83" long and 2.36" wide. The part still consists of 4 holes with two to connect the acrylic wall and the other two to mount the upper and lower chassis. The two holes on the slide are 5 millimeters deep each and every hole on the part was tapped to be able to constrain parts after screwing them in. With iteration one of this part being too thin, the team decided to make the print thicker so the part can be properly tapped for screws. With the robot needing more weight, instead of the infill being 20% like last iteration, the team decided to make it 40% this time. The team opted to not use standoffs and to just use screws as this side support did not move when force was being applied to it.

Side Supports 2 (V2)

Many changes were made into this support than the first one as the first one had many flaws. With the previous iteration, the support would not fit once the two motors were installed onto the bottom chassis. To fix this, a circular hole was carved into the non-slanted portion of the support to allow for the support to wrap around the motors. With the circular hole cut, there was no way a support screw could be designed to go through the entire support; therefore, two different holes were made to connect the upper and lower chassis to this version of the side support. One hole on the top of the support and another hole on the bottom. Previously for the supports, to connect the lower and upper chassis to the support, one long screw would be used to connect the three together; now with the second iteration of the support, two screws are now used to connect the three components together. Moreover, instead of the infill being 20%, it was changed by the team to be 40%.

Wheel Mold (V2)

After the failure of the first wheel mold, the team wanted to tackle all the issues that prevented its success. Since directly placing the motor into the 3-D printed mold will rip it once it applies a torque, the team machined stainless steel motor hubs to connect the mold and motor together to prevent any damage to the mold. The mold also was redesigned to have four holes that would be used to connect the hubs to the mold. In addition, a hole was also made on the hubs to allow a set screw to hold the motor in a desired state. The wheels are 50 mm long, 30 mm in diameter and the silicon is 4 mm thick.

Wheel Creation Process

For creating the custom wheel's, a silicon mixing kit was used. Thanks to the help of staff and other peers, the process was simply explained, and the team was able to successfully execute it on their first try. Smooth-SIL 940 mixing kit was used for the silicon. The first one begins by pouring 50 grams of solution A and 30 grams of solution B, and using this proportion one could scale accordingly. Mix thoroughly for about 3-5 minutes. After this, put cups with solutions into a vacuum chamber to remove all air and air bubbles from within each solution. Assist process by popping air bubbles afterwards. Once the majority of air bubbles are removed, pour into mold and allow the silicon to set for 4-5 hours. For removal, removes all screws from mold. Then pry mold apart from the wheel. Use isopropyl alcohol as necessary to make process easier and help silicon separate from the mold. Once removed, cut off all excess flashing and silicon and wheels are prepared for battle.

Wheel Hubs

With the creation of the wheels and their base, it became quickly apparent to the team that there was no means of connecting the wheels to the motors successfully. Thus, hubs were required in order to attach wheels to the motors. Machining the wheel hubs was a tedious task. This is due to each hub's need for hole accuracy. The hubs contained four screw holes to hold the wheels in place and a set screw hole to allow the motors to rotate the wheels. Starting off, the team placed a stainless-steel rod in the drive center of the Lathe. Afterwards, the team calibrated the machinery by setting the end of the rod to zero. The rotating live center was also centered onto the stainless-steel rod and marked as zero. The tool post was rotated to its unlocked position and a sharp brass cartridge was locked onto the lathe. The drive center was actuated allowing the steel rod to spin, and the tool post slide handle was rotated to slowly chip off the stainless steel to match the initial radius of the sketched wheel hub. Following the last step, the team rotated the cartridge hand wheel to maneuver the brass to cut a desired length.

With the smaller part of the hub done, both hand wheels were reset, and the larger part of the hub was cut the same way the smaller part was. With the hub model done, the next part was to cut off the stainless-steel rod. To do that, the team stopped the rotation of the lathe and placed a new thicker brass insert. The lathe was turned on again and the brass insert was maneuvered all the way through the stainless steels insert to cut off the hub. The rod was then replaced with the newly cut hub and the hub was rotated again. A file was pressed against its sharply cut edges while they rotated to make them smoother. Lastly with the hub in the lathe still, the team attached a drill bit to the tail stock and while the hub was rotating maneuvered the drill bit into the hub to make the motor hole. 4 more holes and the set screw hole were made this way positioning the hub differently every time.

Top Layer of Chassis (V2)

The purpose of this layer is still to hold the breadboard, organize the team's wires, and mount the electronics. The layer has been scaled larger to better fit the breadboard with dimensions: 6.63" by 7.77" and 1/8" thick. The top chassis sits on top of the four side supports screwed into all four of them. With the new sizing, the breadboard fit well on the top chassis and gave enough room for the chassis mounting screws to not disrupt the mounting of the breadboard. With the new design, the team also integrated two next to the breadboard to allow for the wires to be easily managed and cleanly wired.

Bottom Layer of Chassis (V2)

The purpose of this layer is to hold the motor mounts, motors, and give a base for the rest of the robot to sit on. Due to previous issues with the side wall panels and the wheels touching each other, the lower chassis has been redesigned by the team to fit everything perfectly. The entire lower chassis has been made larger to account for the side panels being in contact with the wheels. The new dimensions of the lower chassis are: 8.5" by 9.6" and 1/8" thick.

Motor Mounts (V2)

With the wheels in the previous iteration of the robot being too low to the ground, the team realized the reasoning for this was the wheels being too high. In other words, the motor mounts were propping the motors up too high, allowing the distance from the ground to the lower chassis being too small. Due to this issue, the team decided to lower the height of the motor mounts. This led to a lower rise of the motors, allowing the wheels to contact the ground without the lower chassis scraping the ground. The screw holes and design for them stayed the same besides the height.

Assembly

Assessing the final version of the robot "JARVIS," the team successfully overcame every obstacle created from the first version "Jarvis." The two supports by the motors were modified to allow for the mounts and motors to live simultaneously next to each other. Moving on, the other two larger supports were modified to account for the change in sizing of the newly developed upper and lower chassis. Moreover, the upper chassis was newly designed for wire management and easy breadboard placement. In addition, the lower chassis was redesigned to allow for the wheels to not contact the side panels. The wheels were also redesigned to allow for the insertion of wheel hubs to prevent the destruction of the 3-D printed wheels. Finally, the motor mounts were redesigned to increase the height between the lower chassis and the ground. To assemble the robots, the same hardware was used for the same parts previously. With the same screws, lower motor mounts were first connected to the lower chassis. The motors were then inserted into the mounts and the mounts were tightened to fasten the motors. The motor hubs slid onto the motors and set screws were used to fasten each hub to their respective motors. With each motor hub connected to their respective motors, wheels were attached to each of the hubs with two screws each. After the completion of the lower chassis, the four supports were placed onto their respective locations and the top chassis was placed on top of the four supports. After laying the upper chassis on the four supports, screws were used to connect all four supports to the upper and lower chassis. Next, washers were used to fasten all the parts to connect them to one another. The breadboard was then connected to the upper chassis and the IR sensors were placed in their respective locations. Finally, the four side panels were screwed onto their respective supports.

ELECTRONICS

After designing the mechanical aspect of JARVIS, the team initiates the bot's electronic architecture. A list of all the electronic components is given below in the table and the circuit diagram of the all the components is shown in figure 5.

COMPONENT	QUANTITY
ATMEGA8-16	1
LED's	2
10k potentiometer	1
10k resistor	2
330 ohms resistor	
Flip switch	1
L298N motor driver	2
L7805CV voltage regulator	1
IR Sensor OPB743WZ	2

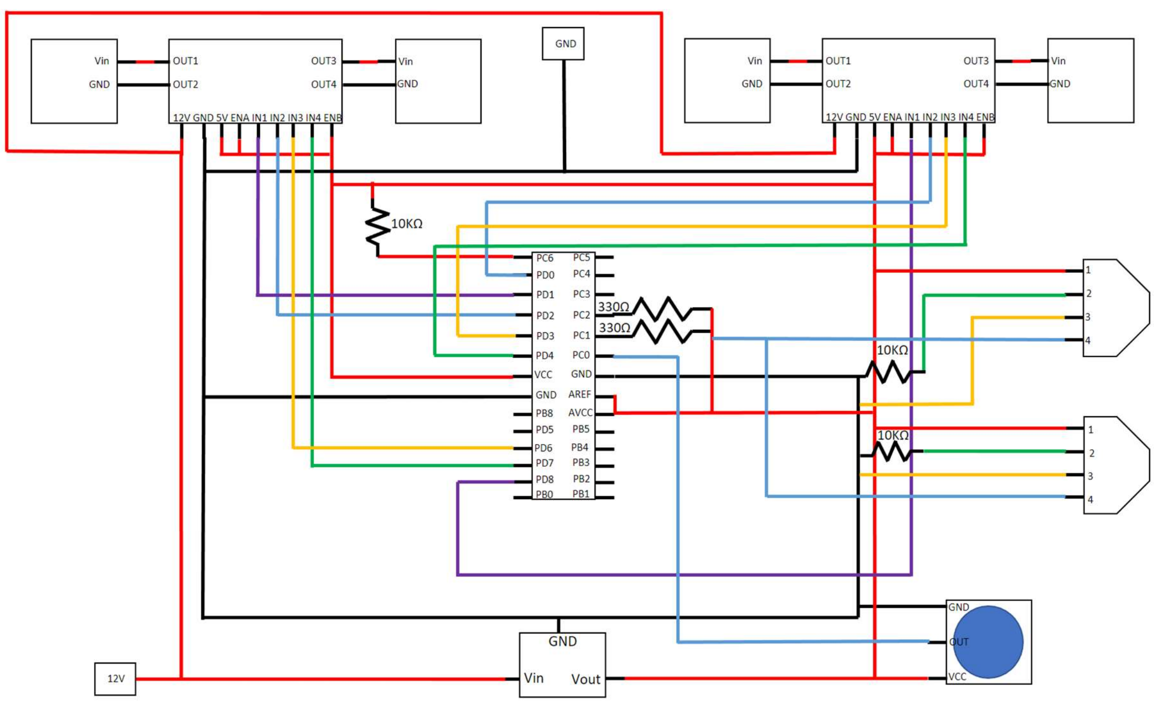


Figure 5. Layout of breadboard layout with all electronics used included

JARVIS uses atmega8-16 microchip as the processing unit which generates precision output signals, count external events, and measure parameters of input signals. As shown in the schematic, JARVIS uses PORT D to access the output of the motor drivers and increase the functionality of motors by spinning in both directions. PORT C on the other hand serves as the medium to collect inputs from the potentiometer and the two IR sensors which are connected to

PC0, PC1, and PC2 respectively. Since all inputs require a reference signal, the AREF and AVCC ports are connected to 5 V.

However, atmega8-16 and other electrical components require 5V and JARVIS's motors are designed to operate at 12 V. Accordingly, the team decided to use a L7805CV voltage regulator which allowed the conversion of 12V signal to 5V. Therefore, as seen in the picture, the source voltage for the motor drivers comes from 12 V supplied by the battery while the power rails are 5 V due to the voltage regulator.

ALGORITHM AND PROGRAMMING

As explained above JARVIS uses two IR sensors which force the bot to perform separate set of moves. A flow chart of the programming is given below.

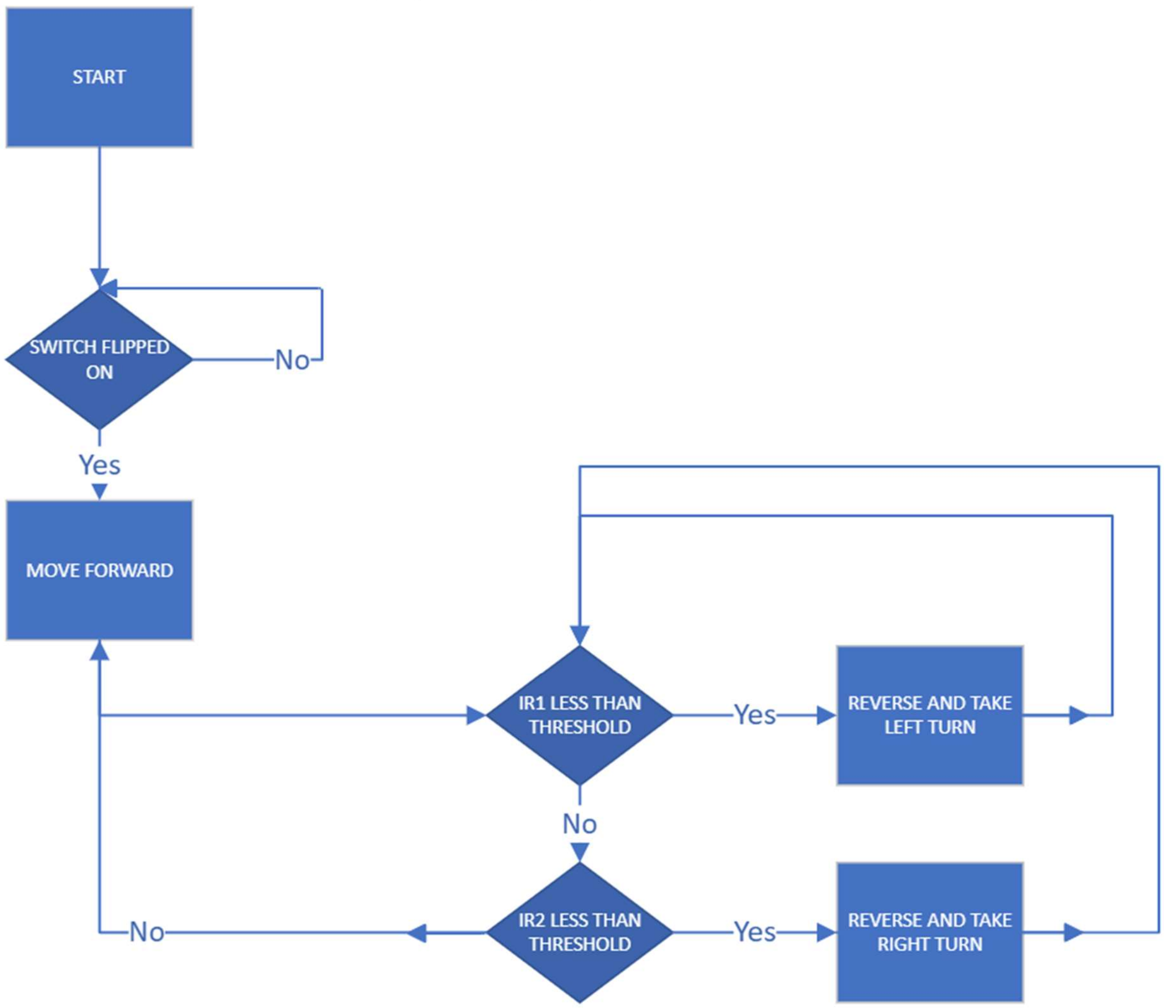


Figure 6. Flow diagram of robot logic and processing from code input

As shown in the illustration, the front switch controls the power to all the electronic components. Once the switch is flipped to its on position, JARVIS starts moving forward. As it does so, it gets inputs from both the IR sensors. If the first IR sensor's input is less than the threshold – meaning it sees white – JARVIS is programmed to reverse for 1500 ms and take a left turn for 3000 ms. This is due to the placement of IR1 sensor on the front right of the chassis. The reverse function sets the wheels to go back, hence forcing JARVIS to come back in the center of the ring and then taking a left turn. Similarly, the team decides to place the IR2 sensor on front left and hence forcing JARVIS to reverse and take a right turn. If both IR sensors sense a value higher than the set threshold, JARVIS keeps moving forward. A graphical representation of the algorithm is shown in figure

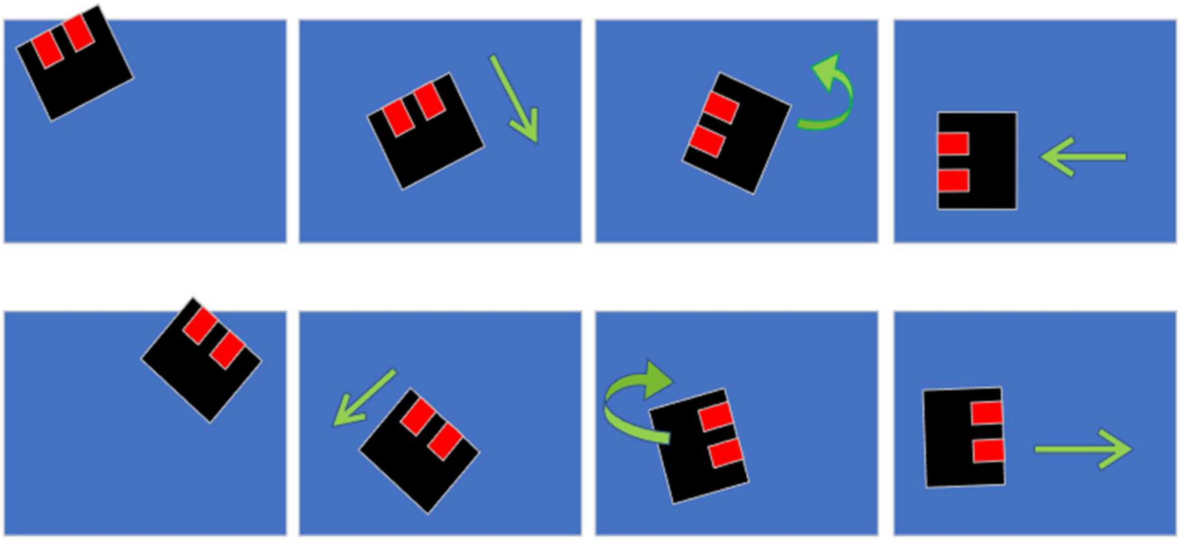


Figure 7. Visual of how robot should behave and operate in order to compete

The sought after code can be found in appendix A.

DISCUSSION

Initially, there was only access to 5 volts since a voltage regulator was not available. In turn, the motors in which are 12V for peak performance were restricted regarding speed and torque. Even at 5V, however, the robot moved at an acceptable speed for the team.

The process of building the breadboard and circuitry was tricky. There were many instances of shorting the circuit and the robot not doing anything. Additionally, there were various instances of the robot only executing half of its functionality. The motor driver was the most troublesome component. The orientation of it was initially flipped, but even once appropriately adjusted, it still gave problems with all pins struggling to connect to the breadboard. At this point, the team decided to steer toward the direction of a L298N driver board, a prebuilt motor driver controller. This gave for more convenient use of the motor driver and ease of access for all pins in the motor driver. This additionally cleared up the breadboard such that it wasn't as crowded or clustered anymore. There was never an issue of overloading the microcontroller or the breadboard with more voltage than the chip can handle.

The infrared photo reflector sensor came very long, which would be convenient for larger robots however for its purpose on JARVIS, each wire did not need to be longer than around 8 inches. To address this dilemma, each wire was cut to around 4 inches, then put in conjunction with a jumper cable. Stripping each of each wire, they were both stranded which made it convenient for infusing both wires together. To make the connection better between each wire, independent flux was applied to the joint of wire. Then using a solder with lead as the material, the connection was finalized. For safety and aesthetic purposes, DR-25 heat shrink was applied over the live wires. While putting each together, there were times where heat shrink was not pre-slid onto the wires before soldering and this caused a small inconvenience since the right size heat shrink could not be applied once the soldering was complete. Thus, a larger, upsized heat shrink was applied. This did not have any impact on the performance or safety as all live wires were still as effective. With one of the infrared photo reflectors, the connection was intermittent and would provide very inconsistent results with the edge detection. Through redoing the process of connecting the jumper cables to the infrared photo reflector wires, this specific infrared photo reflector began to cooperate and function as expected. On the robot, two were placed in the front of the robot, one towards the front left and one towards the front right. The front right one was the one that failed to detect consistently. Aforementioned however, through resoldering that infrared photo reflector, the robot processed the edge inputs flawlessly and acted as expected.

Regarding batteries, the standout requirement was that the batteries must be made out of either nickel-cadmium or nickel-metal hydride. This narrowed down the search since the first, most convenient battery material to be chosen is lithium ion. After thorough research into battery choices available, the team determined the appropriate battery specifications needed to help the robot function at peak performance is a voltage of 12 volts and discharge current of 9 amps. The battery the team came to order was a 12 volt, 1800 mAh, 10 cell rechargeable battery pack. One factor that was considered but underestimated was the amount of charge four motors together would require. A battery charger was additionally ordered to fill the battery with charge. Although the battery did allow the robot to function and move, the charge the battery held wasn't very significant. This in turn led to the battery dying within 20-35 seconds of continuous use and thus not feasible for the requirement of this project. For temporary use and testing, an oscilloscope was used as the voltage source. For competition, multiple of these battery packs will be used, providing us with 40 total cells, each which will power their own motor. This should be enough charge to compete and last throughout the competition.

CONCLUSIONS

Overall, after designing the robot for the sumo competition it becomes apparent that strategies across the board are widely similar. While some groups preferred speed over strength, the two categories essentially developed into speed or strength with the same method of attacking. With basic level of knowledge in regard to digital electronics and logical design, JARVIS was able to remain in the ring and even out muscle a variety of objects as well as robots. On the flip side, JARVIS has yet to reach its full potential with LPS and utilizing a LIDAR sensor. The ideology for defense was successful as JARVIS was blinded by no robot and was always prepared for any angle. The speed of the motors was ideal, providing the quickness to get around and the strength to push other robots around. For the next iteration, possibly lowering the revolutions per minute of the motor to give access for even more torque could be ideal and better suit the strategy JARVIS was built for. Additionally, a larger battery pack would play to JARVIS's advantage and possibly bring JARVIS even lower to the ground since the mat wasn't as bumpy and hilly as anticipated. This competition came down to a battle of strength and wedge height in the end. For JARVIS, these two were its strengths and JARVIS is ready to opponent willing to challenge.

APPENDIX I: Rules

No.	Rule
1	The “ring” will be approximately 3ft by 3ft square and made from neoprene rubber (McMaster # 9455K62 - ASTM D2000 BC).
2	The opponents will be placed at one of three starting positions (straight on, 45 degrees, and 90 degrees from opponent) within the ring.
3	When signaled, the teams will activate their robots simultaneously.
4	The robot will attempt to locate the other robot and push it out of the ring.
5	The robot is considered “out of the ring” when more than half of the robot is no longer within the main area of the ring.
6	If neither robot has won after 1 minute (or when both teams agree on a draw), the round is considered a draw.
7	Bumps and imperfections will be less than 1/8” at any single spot and no more than 1/2” over the entire area.

APPENDIX II: Code

```
// Team JARVIS - Evan Lutchmidat, Khushant Khurana, Scott Losirisup
#include <xc.h>
#include <avr/io.h>
#include <util/delay.h>

//Define all the functions.
unsigned int get_Adc(unsigned char);
void sbi(volatile uint8_t *, unsigned char);
void cbi(volatile uint8_t *, unsigned char);
void forward();
void left_turn();
void right_turn();
void reverse();
void stop();

void right_turn();
void left_turn();

//Defining pins for the motor logic!
//Motors A and C are in front. Motors B and D are in back. For now we are using one
driver to control all motors. So 2 motors would work in conjunction.
//Motors A and B are outside for right turn. Motors C and D are inside for right turn.

#define IN1_PIN PD1 // Control Pin 1 for Motor A
#define IN2_PIN PD2 // Control Pin 2 for Motor A

#define IN3_PIN PD3 // Control Pin 1 for Motor B
#define IN4_PIN PD4 // Control Pin 2 for Motor B

#define IN5_PIN PD5 // Control Pin 1 for Motor C
#define IN6_PIN PD6 // Control Pin 2 for Motor C

#define IN7_PIN PD7 // Control Pin 1 for Motor D
#define IN8_PIN PD0 // Control Pin 2 for Motor D

int main(void)
{
       unsigned int threshold, ir1, ir2;
       //threshold - controlled by a potentiometer. - PC0
       //ir1 - the front left ir sensor. - PC1
       //ir2 - the front right ir sensor. - PC2
       DDRD = 0xFF; // set all pins to output
       while(1){
                     threshold = get_Adc(0);
                     ir1 = get_Adc(1);
                     ir2 = get_Adc(2);

                     if (ir1 <= threshold){
                            reverse();
                            _delay_ms(1500);
                            left_turn();
                            _delay_ms(3000);
                     }
                     if (ir2 <= threshold){
                            cbi(&PORTB, PB1);
```

```
                                    reverse();
                                    _delay_ms(1500);
                                    right_turn();
                                    _delay_ms(3000);
                            }
                            else{
                                    forward();
                            }
                            _delay_ms(10);
        }
        return 0;
}

//Conversion of inputs. The channels are pins PC0-PC2.

unsigned int get_Adc(unsigned char channel){
        ADCSRA = (1<<ADEN) | (1<<ADPS2) | (1<<ADPS1) | (1<<ADPS0);
        ADMUX = (1<<REFS0);
        ADMUX = (ADMUX & 0xF8) | (channel);
        sbi(&ADCSRA, ADSC);
        while(ADCSRA & (1<<ADSC));
        return ADC;
}

// Setting the LED connected to IR sensor on.
void sbi(volatile uint8_t * SFR, unsigned char bitval){
        *SFR |= (1<<bitval);
}

// Setting the LED connected to IR sensor off.
void cbi(volatile uint8_t * SFR, unsigned char bitval){
        *SFR &= ~(1<<bitval);
}

void forward()
{
        PORTD |= (1 << IN1_PIN)  | (1<<IN5_PIN) | (1<<IN3_PIN) | (1<<IN7_PIN) ; // (1 <<
IN3_PIN) Set bits to 1
        PORTD &= ~((1 << IN2_PIN) |  (1<<IN6_PIN) | (1<<IN4_PIN) | (1<<IN8_PIN)); // (1 <<
IN4_PIN) |Set bits to 0
}

void stop()
{
        PORTD |= ((1 << IN1_PIN) | (1 << IN2_PIN) | (1 << IN3_PIN) | (1 << IN4_PIN) |
(1<<IN5_PIN) | (1<<IN6_PIN) | (1<<IN7_PIN) | (1<<IN8_PIN)); // Set all bits to 1. So stop
all motors.
}

void reverse()
{
        PORTD |= (1 << IN2_PIN) | (1 << IN6_PIN) | (1<<IN4_PIN) | (1<<IN8_PIN); // Set
pins to 1.
        PORTD &= ~((1 << IN1_PIN) | (1 << IN5_PIN) | (1 << IN3_PIN) | (1<<IN7_PIN)); //|
Step pins to 0.
}

void right_turn(){
```

```
	PORTD |= (1 << IN1_PIN) | (1 << IN3_PIN) | (1<<IN6_PIN) | (1<<IN8_PIN); // Set
pins to 1.
	PORTD &= ~((1 << IN2_PIN) | (1 << IN4_PIN) | (1<<IN5_PIN) | (1<<IN7_PIN)); // Set
pins to 0.
}

void left_turn(){
	PORTD |= (1 << IN2_PIN) | (1 << IN4_PIN) | (1<<IN5_PIN) | (1<<IN7_PIN); // Set
pins to 1.
	PORTD &= ~((1 << IN1_PIN) | (1 << IN3_PIN) | (1<<IN6_PIN) | (1<<IN8_PIN)); // Set
pins to 0.
}
```

Appendix III: SDS Sheets and Other Specification Sheets

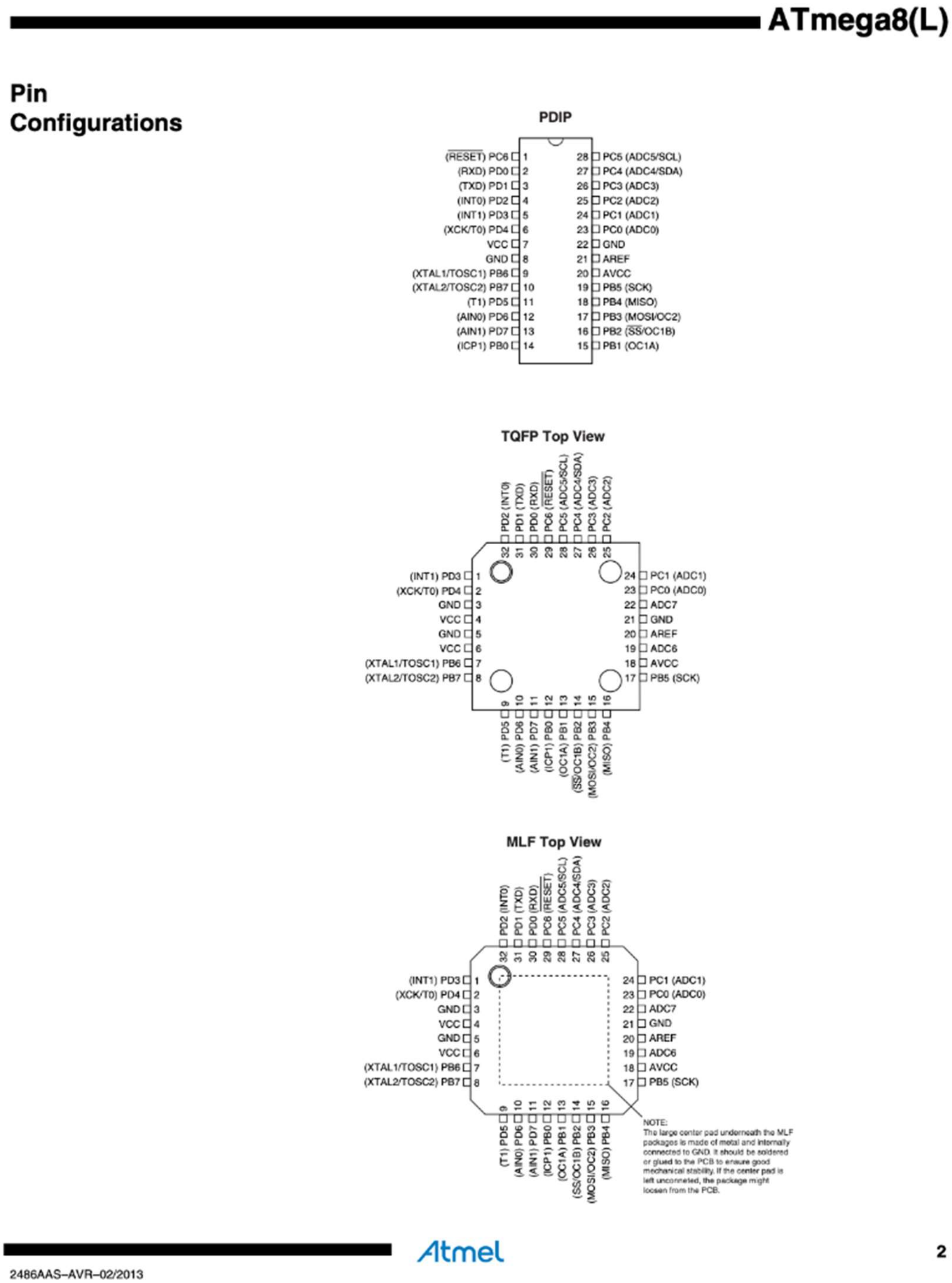


Figure 8. Specifications of Programmable Microchip ATmega8

Product Folder | Order Now | Technical Documents | Tools & Software | Support & Community

TEXAS INSTRUMENTS

LM340, LM340A, LM7805, LM7812, LM7815

SNOSBT0L – FEBRUARY 2000 – REVISED SEPTEMBER 2016

LM340, LM340A and LM7805 Family Wide V_{IN} 1.5-A Fixed Voltage Regulators

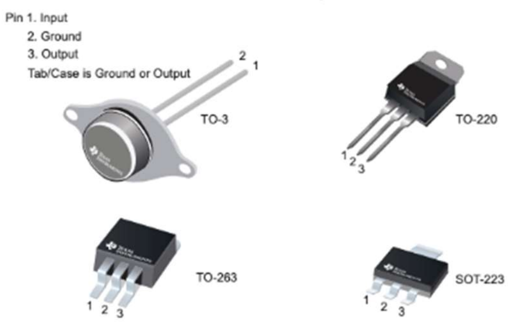
1 Features

- Output Current up to 1.5 A
- Available in Fixed 5-V, 12-V, and 15-V Options
- Output Voltage Tolerances of ±2% at T_J = 25°C (LM340A)
- Line Regulation of 0.01% / V of at 1-A Load (LM340A)
- Load Regulation of 0.3% / A (LM340A)
- Internal Thermal Overload, Short-Circuit and SOA Protection
- Available in Space-Saving SOT-223 Package
- Output Capacitance Not Required for Stability

2 Applications

- Industrial Power Supplies
- SMPS Post Regulation
- HVAC Systems
- AC Inventors
- Test and Measurement Equipment
- Brushed and Brushless DC Motor Drivers
- Solar Energy String Invertors

Available Packages



3 Description

The LM340 and LM7805 Family monolithic 3-terminal positive voltage regulators employ internal current-limiting, thermal shutdown and safe-area compensation, making them essentially indestructible. If adequate heat sinking is provided, they can deliver over 1.5-A output current. They are intended as fixed voltage regulators in a wide range of applications including local (on-card) regulation for elimination of noise and distribution problems associated with single-point regulation. In addition to use as fixed voltage regulators, these devices can be used with external components to obtain adjustable output voltages and currents.

Considerable effort was expended to make the entire series of regulators easy to use and minimize the number of external components. It is not necessary to bypass the output, although this does improve transient response. Input bypassing is needed only if the regulator is located far from the filter capacitor of the power supply.

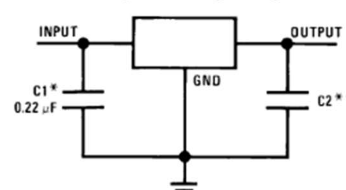
LM7805 is also available in a higher accuracy and better performance version (LM340A). Refer to LM340A specifications in the *LM340A Electrical Characterisitcs* table.

Device Information(1)

PART NUMBER	PACKAGE	BODY SIZE (NOM)
LM340x LM7805 Family	DDPAK/TO-263 (3)	10.18 mm × 8.41 mm
	SOT-223 (4)	6.50 mm × 3.50 mm
	TO-220 (3)	14.986 mm × 10.16 mm
	TO-3 (2)	38.94 mm x 25.40 mm

(1) For all available packages, see the orderable addendum at the end of the data sheet.

Fixed Output Voltage Regulator



*Required if the regulator is located far from the power supply filter.
**Although no output capacitor is needed for stability, it does help transient response. (If needed, use 0.1-µF, ceramic disc).

Figure 9. Specifications of Voltage Regulator LM340

L298

DUAL FULL-BRIDGE DRIVER

- OPERATING SUPPLY VOLTAGE UP TO 46 V
- TOTAL DC CURRENT UP TO 4 A
- LOW SATURATION VOLTAGE
- OVERTEMPERATURE PROTECTION
- LOGICAL "0" INPUT VOLTAGE UP TO 1.5 V (HIGH NOISE IMMUNITY)

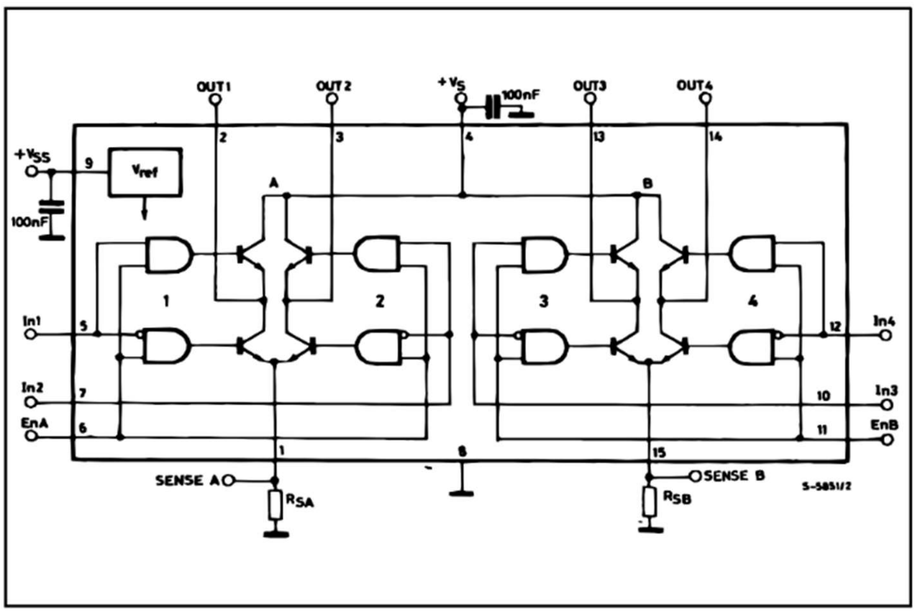
Multiwatt15 PowerSO20

ORDERING NUMBERS : L298N (Multiwatt Vert.)
L298HN (Multiwatt Horiz.)
L298P (PowerSO20)

DESCRIPTION

The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors. Two enable inputs are provided to enable or disable the device independently of the input signals. The emitters of the lower transistors of each bridge are connected together and the corresponding external terminal can be used for the connection of an external sensing resistor. An additional supply input is provided so that the logic works at a lower voltage.

BLOCK DIAGRAM



Jenuary 2000 1/13

Figure 10. Specifications of Motor Driver L298

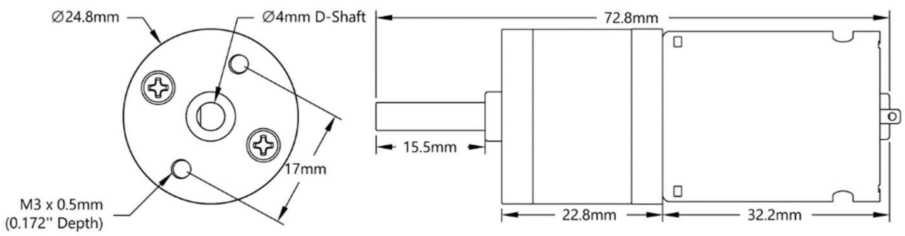


Figure 11. Blown-Up Layout and Drawing of Motor

SPECS

Output Shaft Style	D-shaft
Motor Type	Brushed DC
Output Shaft Support	Bushing
Gear Material	Metal
Weight	3.17 oz (87g)
Voltage (Nominal)	12V
Speed (No Load @ 12VDC)	170 rpm
Current (No Load @ 12VDC)	0.10A
Current (Stall @ 12VDC)	3.8A
Torque (Stall @ 12VDC)	306 oz-in (22.04 kgf-cm)
Gearbox Style	Straight Cut Spur
Connector Type	Male Spade Terminal
Gear Ratio	57:1

Figure 12. Specifications of Motor

Safety Data Sheet

OSHA 1910.1200
Revision Date: 06/23/2015
Print Date: 11/14/2016
Version: 2.3

EVONIK INDUSTRIES

1. Product and Company Identification

1.1. Product identifier

Trade name : ACRYLITE® - Sheets/Rods/Tubes

1.2. Recommended use of the chemical and restrictions on use

Recommended use(s): building glazing. light advertising. furniture. trade-fair booth design. displays. decoration. Industrial use

Non-recommended use(s): None known.

1.3. Details of the supplier of the safety data sheet

Evonik CYRO LLC
299 Jefferson Road
Parsippany, NJ 07054-0677
USA

973-929-8000
973-929-8040 (fax)

1-973-929-8060 (Product Information Number)
1-800-424-9300 (24 Hour Emergency Number, CHEMTREC)

2. Hazards identification

2.1. Classification of the substance or mixture

This article is not classified according to GHS

Classification according to Regulation 29CFR 1910.1200

This product is not considered to be a hazardous substance or mixture when classified in accordance with Regulation 29 CFR 1910.1200 (US GHS).

2.2. Label elements

This article is not classified according to GHS

2.3. Other hazards

None known

3. Composition/information on ingredients

3.1. Substances

3.2. Mixtures

Hazardous Ingredients

Component	CAS-No.	Content	Hazard class / Hazard category / Hazard statement

Figure 13. MSDS of Acrylic Sheets used

Appendix IV: Bill of Materials

ME 353 - Mechatronics			
Cost Report: Summary			
Team Matrix			

Items	Vender	Contact	Price
3	Canal street plastics	DOARIsANYqkD0RUFeVOeI807uJ	18
1	Servocity	http://www.servocity.com/	59.96
2	Home Depot	https://www.homedepot.com/	33.12
	Raw materials	AACE lab	47
1	Professor Cusack	Electronics Kit	25
		Total	183.08

Figure 14 – Bill of Materials

Appendix V: Pictures from Iterations of JARVIS

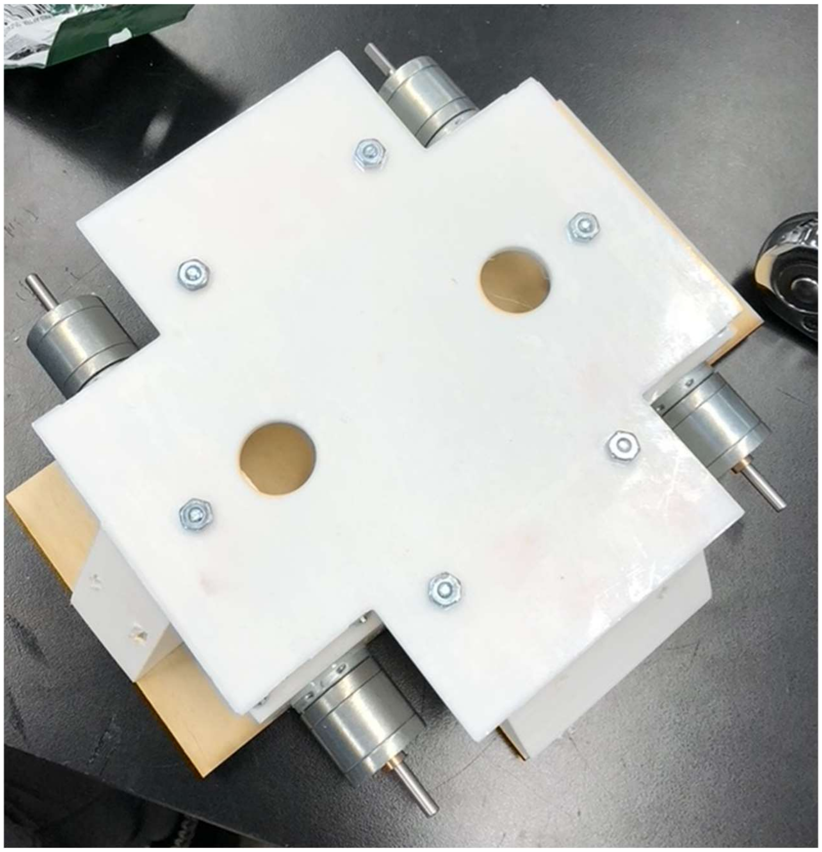


Figure 15. Top view of iteration one of Jarvis

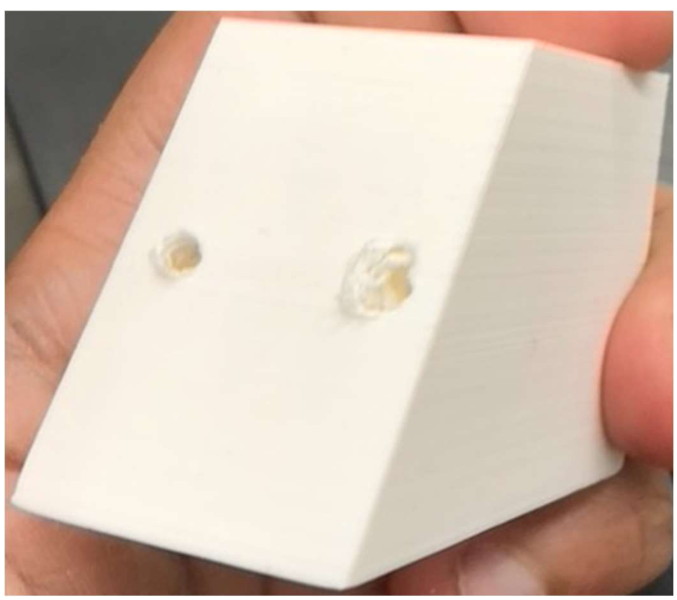


Figure 16. Side Support 2 version one with a stripped mounting hole

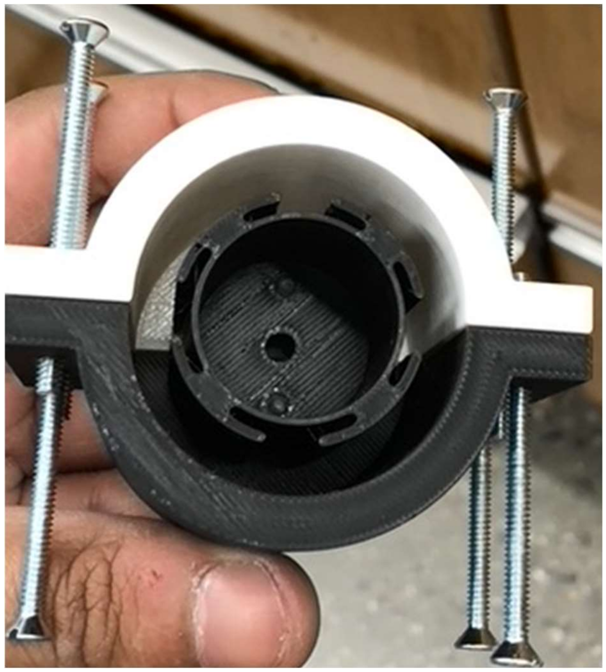


Figure 17. Wheel Mold version two fully assembled prior to filling with silicon

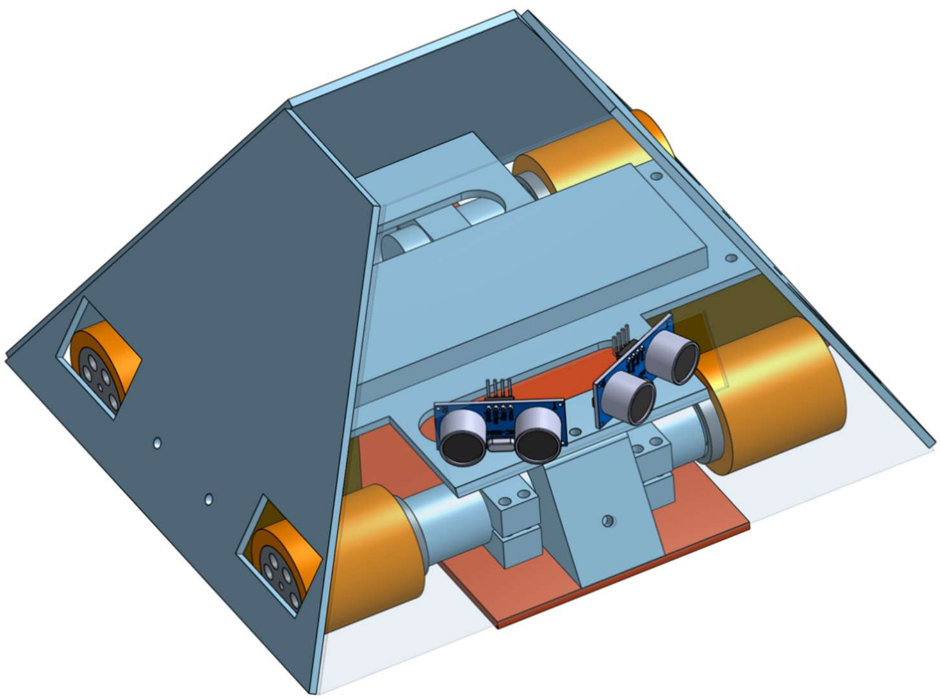


Figure 18. JARVIS final CAD with front wall transparent

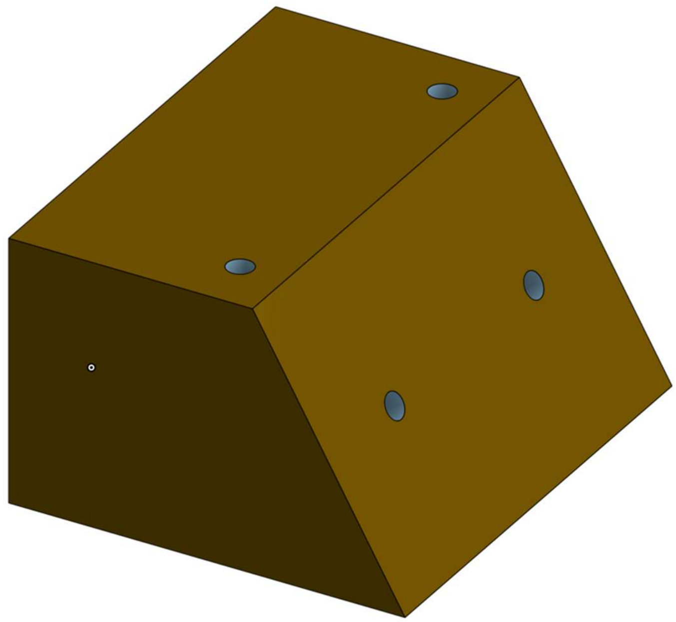


Figure 19. Side Support 1 version two final CAD

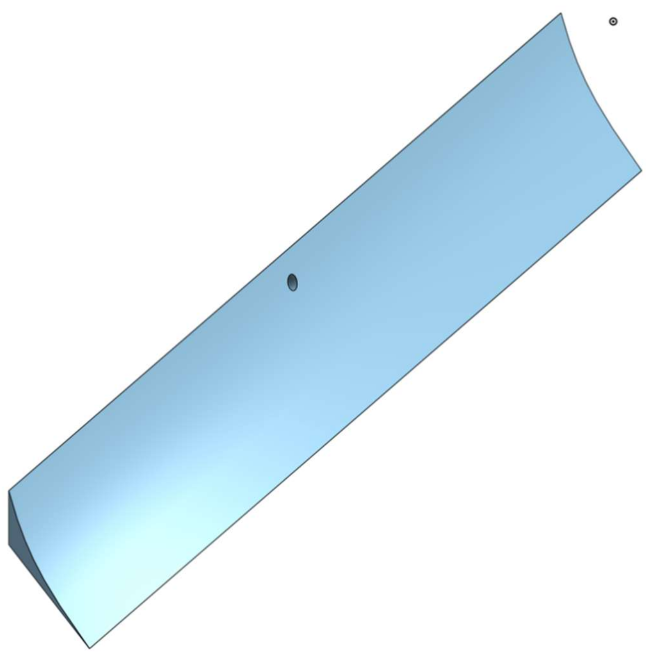


Figure 20. CAD Wedge to be attached to bottom of each wall; made of teflon

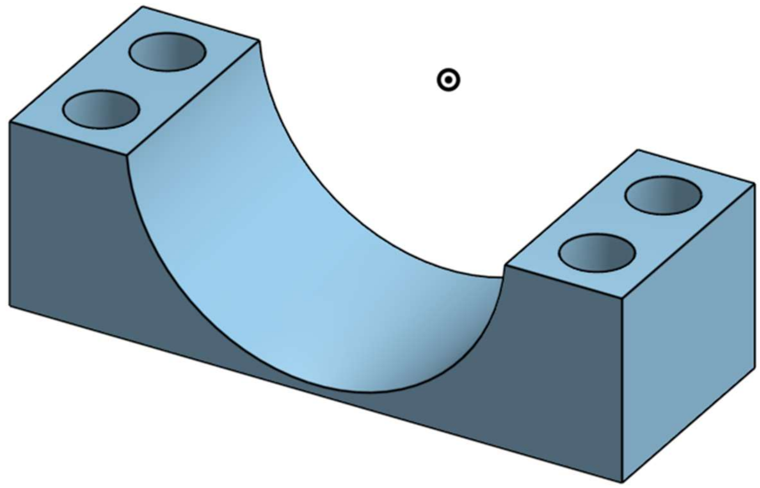


Figure 21. Final iteration of the motor mounts

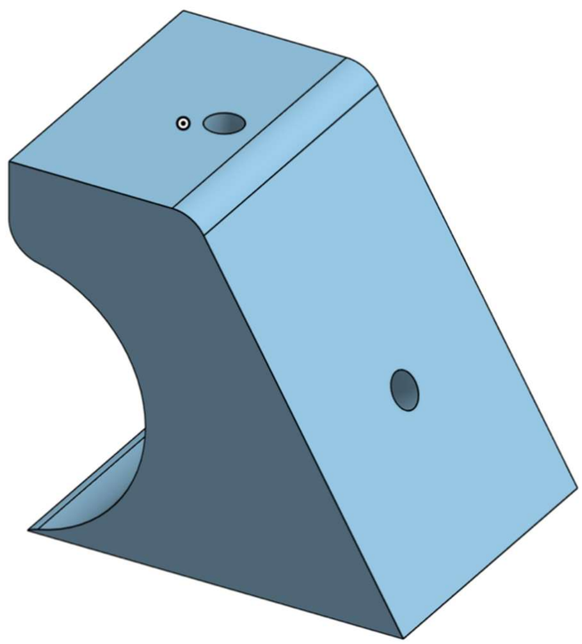


Figure 22. Side Support 2 version two with one hole for mounting and extruded semi-circle portion for motor placement

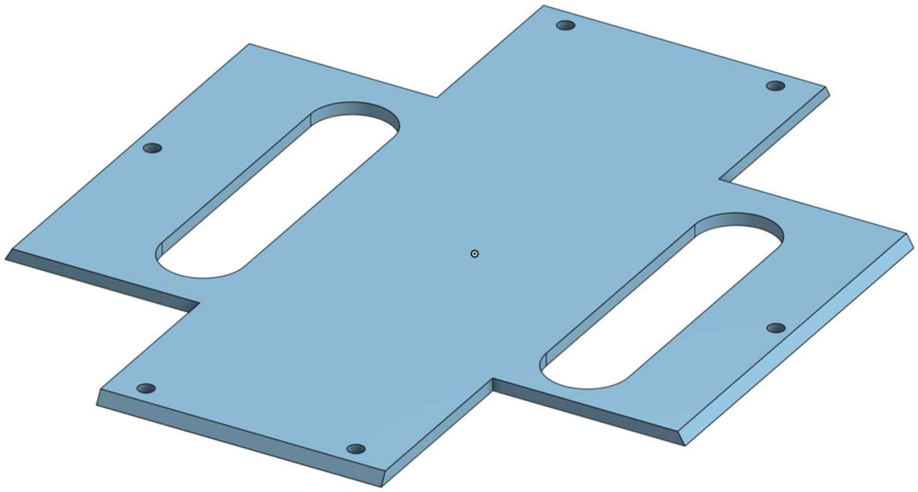


Figure 23. Final iteration of top layer of the chassis; dimensioned 6.63” x 7.77”

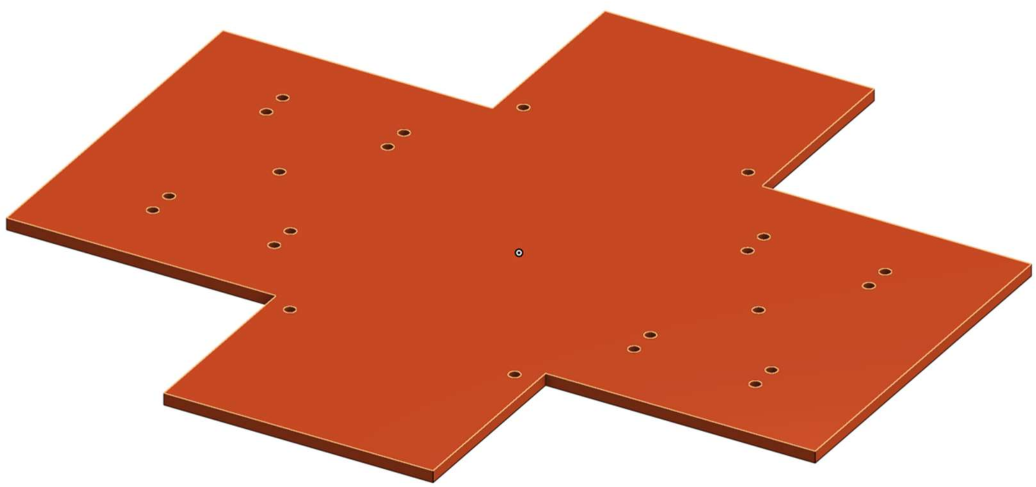


Figure 24. Final iteration of the bottom layer of the chassis; dimensioned 8.5” x 9.6”

BIBLIOGRAPHY

[1] https://www.servocity.com/170-rpm-econ-gear-motor/

[2] Alberto L., "How to make the best sumo Robot Wheels to increase the Grip and Torque" https://miscircuitos.com/how-to-cast-sumo-wheels-handmade/