TRUERAIN – THE SMART SPRINKLER

Final Report

Machine Design (ENGG*3280)

Instructor: Hari Simha, Ph. D, P. Eng.

Group 9, Section 205

Chase Ambeau - 0896304

Brennan Jay - 0887114

Jun Lu (Aaron) - 0885914

Orion Miller - 0939412

Andrew Roberts - 0945811

Due Date: 01 / 12 / 2017

Executive Summary

Millions of small-time gardeners and greenhouse owners around the globe experience the issue of watering their plants following a strict schedule. When watering, the average plant needs a certain amount of water in order to sustain life. If it is not well kept, the greenery will soon wither away and die. This particular point of issue can be resolved by the TrueRain, a smart sprinkler mechanism that can water a garden following a strict schedule and locate the specific positions of each plant with the pre-programmed knowledge of how much water will be required for the sprout to sustain life.

The simple, ergonomic design created grants the user the ability to simply mount the mechanism along a fence or on the walls of a greenhouse in order to roll along a track whilst watering the plants. The machine comes in a standard length of four feet, with all of the mechanical and electrical parts included. The system can further be designed such that a customizable model can be implemented, adding more tracks and racks to create a full circuit for the mechanism. With the use of a simple controller, the TrueRain system was programmed to locate the position of a plant or multiple plants the as it travelled along the track, also knowing the duration of time each plant required attention for.

In an ideal environment, the TrueRain sprinkler system would roll along the roller rack using the force exerted by the pinion on the DC motor, taking it across the track to the programmed location of each plant. Once at its position of action, the stepper motor would rotate itself to its designated angle to ensure the water pumped through a hose reaches its target as it spews out of its custom designed nozzle. The main problem with the machine once completed was the pinions ability to flow flawlessly across the rack to reach its programmed position. Accuracy with the pinion along the rack proved difficult to design a precise program that allowed the mechanism to line itself up exactly.

Some additional modifications that could be made to ensure a better, more efficient design would include the sprinkler being more level with the ground at the base of the plant to guarantee that plants were not being pelted by water, losing water from splashing off leaves, and leaving the leaves sunburnt from the sun rays.

CONTENTS

LIST OF TABLES

List Of Figures

Introduction

Most current sprinkler systems are designed to water monocultures. They either spray in sweeping arcs, or cover a static area without specific attention to the locations of plants, because each plant requires roughly the same amount of water. Distributed irrigation systems with multiple smaller sprinklers can water with greater control, but require piping systems to transport water to the plants.

Personal gardens tend to be smaller, with multiple species packed close together, requiring different watering regimes. Where an automatic watering system is desired (in a large nursery or a garden that must go unattended for extended periods), the ability to water individual plants at different times, rather than watering the entire garden or a section of it at intervals, offers advantages in the health and growth of some species.

A watering system with these features can be accomplished with a mobile directed sprinkler, controlled and scheduled with an arduino microcontroller. Such a system would afford growers greater control over the distribution of their water, while reducing installation time and complexity compared to current distributed irrigation systems.

The proposed system uses a DC motor to pull a sprinkler assembly on a sled along tracks. The tracks are secured at height and would run the length of a room or a garden. Servos adjust the nozzle angle and flow rate and an arduino unit controls the motor and servos and remembers the watering schedule and plant locations.

BACKGROUND

In order to design a mechanism that would provide a sufficient amount of water to a specific location, a series of criteria and constraints were followed to produce the most practical, modern machine. Ensuring this, the following criteria and constraints were made effective immediately after the design stage ensued.

CRITERIA

Minimal Losses: Waters plants accurately without wasting water to areas where no plants resided. This ensures that the consumer is not losing money for water that they are not using as well as helping the environment conserve its water resources.

Expandability: Able to be customized to fit the lengths of the consumers garden/greenhouse. If a longer track is needed, one can purchase said amount of track and rack to extend its distance travelled.

Autonomous: Carries out a watering plan based on a pre-programmed cycle made for the specific regimes of each plant. One would simply need to initially set up the plan for the machine to work autonomously.

Aesthetics: The sprinkler should be pleasing to the eye. It should be small and able to be hidden away so that it is not in the way of people.

CONSTRAINTS

Cost Efficient: The cost for one unit will be approximately $250 or less, which includes the materials used and the assembly of the machine.

Safety: The factor of safety for the product must be greater than 1.5 in order to ensure no one is hurt by the machine if failure occurs. Also, the design must be waterproofed in order to prevent electrical damage to the mechanism.

Weight: The weight of the sprinkler system must be less than 25lbs to ensure the consumer can easily take the machine of the track if needed as well as ease of transportation/shipping.

Size: The size that the machine can work efficiently on is a 2ft by 4ft gardening space to ensure accuracy is met.

Covered: In order for the machine to remain in an environment such as a greenhouse where the air is full of humidity, it must be covered and waterproofed to prevent corrosion and fouling of the materials.

Gentle: Must be able to gently water plants so that the plants are not battered by jet streams as they are watered. If this were to occur, the plant would surely die from the aggression to the leaves.

Detailed Design And Analysis

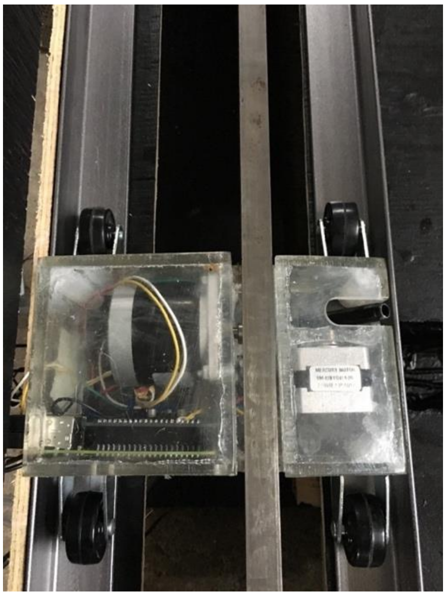
The TrueRain mechanism was made up of various components; The main housing, driving mechanism/motors, motor for angling of nozzle, the pump, the demo enclosure and the electrical components.

The demo enclosure was designed to be relatively portable (able to move from demonstration to demonstration with minimal deconstruction). It was made out of plywood and 2x4's. The enclosure was also designed to act as a reservoir for the water that is used by the pump. The slot in the middle of the enclosure acting as the reservoir can be seen in Figure 3. The reservoir was lined with plastic and sprayed heavily with a water proof sealant in order to try to prevent leaking. The box is 4 feet long and 2 feet wide. The box was designed to be 4 feet long in order to show the full capabilities of the linear movement of the mechanism. The ledge that the mechanism sits on has a slot on the bottom, and also at the back of it. The slot on the bottom was added in order for the hose to be able to move with the mechanism. The slot on the back was cut for the wires on the back of the Pi to be able to move along with it as well.

The driving mechanism was designed to use a rack and pinion system. The pinion gear sat on a 12V motor that sits inside the housing. The pinion gear mates with the rack which was supported by a 4ft steel bar and 2 brackets on either end that are screwed into the demo enclosure. The steel bar was added due to the small size of the rack. Since the rack was so long (4ft), it bowed in the middle which effected the strength of it creating problems when the pinion mated in the middle. The rack was also very slim, so the added steel bar adds much needed support.

Another motor located inside the housing was used for the angling of the nozzle. This was necessary in order to be able to reach plants at varying distances. The smaller 12V stepper motor's shaft was glued to the nozzle. Using the controller programmed to the Pi, the stepper motor makes it possible to change the angles to reach exactly where the plant is located. The stepper motor was used because of the ability to change the angle precisely, in this case by 1.8 degrees. Other motors don't have that option.

Figure 1: Motors in housing



A pump is used with the demo enclosure in order to display the capabilities of the mechanism. In a final production version however, the system will have the option to be connected to the customer's regular garden hose, or pump from a reservoir such as a rain collecting drum.

The pump is connected to the nozzle in the housing via a clear tube. The pump gets its water from the reservoir located in the middle of the demo enclosure as mentioned above. The pump is able to reuse the water due to the design of the enclosure; all of the used water flows back into the reservoir. This makes it possible to keep the demonstration going without stoppage.

Figure 2: Pump used in assembly

The housing was the heart of the TrueRain mechanism. It supported both motors, the pi, and the nozzle. It was made out of an acrylic sheet and cut with a band saw. It was glued using Locktite, an extremely strong adhesive intended for gluing aquarium tanks. The holes used to bolt both motors and the slots cut out to make the hose and cords accessible were cut using the mill. The dimensions of the box were 177mm long by 107mm wide, and 67mm at its tallest.

Because of concerns about the imperfections associated with cutting by hand on a band saw, it was originally planned to create the plates on a CNC router at the Formula SAE shop the group had access to. A CAD file of the proper layout for machining the plates was created, and imported into MasterCAM to program all the required toolpaths. However, this router had only been acquired recently and had never been used, and difficulties were encountered when trying to run the router with control software not built specifically for the model. Because of this problem, the group was unable to CNC machine the plates, so the housing was created with the bandsaw, which was ultimately accurate enough to function as intended.

The electrical aspect of the mechanism is the most complex component. The motors and pump are powered by an external AC/DC inverter supplying 12V DC that is secured to the back of the enclosure. The Pi controls the external pump with one of its GPIO connector pins (GPIO 17 in this case) which supplies a 3.3V signal to a relay circuit. When the signal to the circuit is high, the 12V DC supply powers the pump, and when it is low the pump is off.

The operation of the system is quite easy. Using the remote controller provided, use the right face buttons to move the sled side to side, and to change the angle of the nozzle. The pump is activated using the R1 button on the top of the controller. To turn the pump off, simply press the button again. The programming code used for the operation of the controller can be found in the appendix.

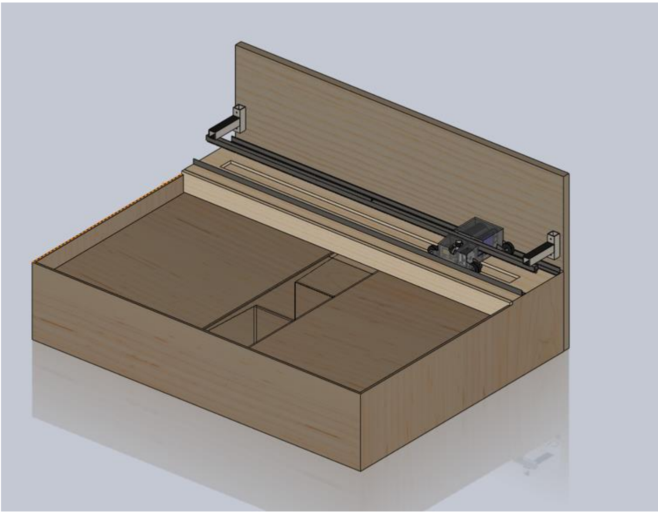


Figure 3: Completed assembly

Analysis And Optimization

$$F_{rack} = m_{rack} * g \qquad +\downarrow \Sigma F_y = 0 = F_{rack} - F_R \qquad \Sigma M_z = F * l$$

$$F_{rack} = (0.168kg)\left(9.81\frac{m}{s^2}\right) \qquad F_R = 1.648N \qquad \Sigma M_z = (1.648N)(0.135m)$$

$$F_{rack} = 1.648N \qquad \Sigma M_z = 0.222N \cdot m$$

$$\hookrightarrow 0.111N \cdot m \ \ for\ single\ support$$

Figure 4: Determining the bending moment about the upper support beam

$$F_{board} = m_{board} \cdot g \qquad F_{sled} = m_{sled} \cdot g \qquad F_{L-Rail} = m_{L-Rail} \cdot g$$

$$F_{board} = (0.550kg)\left(9.81\frac{m}{s^2}\right) \qquad F_{sled} = (3.5kg)\left(9.81\frac{m}{s^2}\right) \qquad F_{L-Rail} = (0.497kg)\left(9.81\frac{m}{s^2}\right)$$

$$F_{board} = 5.396N \qquad F_{sled} = 34.335N \qquad F_{L-Rail} = 4.876N$$

$$+\uparrow F_y = 0 = F_{board} + F_{sled} + 2F_{L-Rail} - F_R \qquad M_A = l\,(F_{board} + F_{sled} + 2F_{L-Rail})$$

$$+\uparrow F_y = 49.483N \qquad M_A = (0.135m)(5.396N + 34.335N + 4.876N)$$

$$M_A = 6.680N \cdot m$$

Figure 5: Determining the bending moment about the base support beam

By determining the bending moment about the top and base support beams, we are able to ensure the force acting on the board and the rack are low enough for our base and beams to support.

$$r = \frac{D_p}{2} = 0.00762m \qquad T_{max} = 84oz \cdot in = 0.593N \cdot m$$

$$F_t = \frac{T}{r} \qquad V_m = \frac{\pi D_p n}{1000} \qquad F_d = \frac{600+V_m}{600}F_t \qquad F_s = \frac{S_n Y b}{P_d}$$

$$F_t = \frac{0.593N\cdot m}{0.00762m} \qquad V_m = \frac{\pi(0.0152m)(15)}{1000} \qquad F_d = \frac{600+(1.19\times10^{-5})}{600}(77.82) \qquad F_s = \frac{(1270)(0.245)(0.00635)}{0.0159}$$

$$F_t = 77.821N \qquad V_m = 7.16\times10^{-4}\frac{m}{min} \qquad F_d = 77.82N \qquad F_s = 111.54N$$

$$V_m = 1.19\times10^{-5}\frac{m}{s}$$

$$N_{sf} \geq \frac{F_s}{F_d} \rightarrow N_{sf} \geq \frac{111.54N}{77.82N} \rightarrow N_{sf} \geq 1.433$$

Figure 6: Determining the factor of safety of the pinion gear

Determining the factor of safety of the stepper motor shaft is to make sure the pinion gear would not break after millions of rotations and the factor of safety is more than 1.4 which meets the factor of safety standard.

$F_{pin} = m_{pin} \cdot g$	$M_z = F_{pin} \cdot l$	$\sigma_x = \frac{M_z c}{I_z}$
$F_{pin} = (0.0136kg)\left(9.81\frac{m}{s^2}\right)$	$M_z = (0.133N)(0.02m)$	$\sigma_x = \frac{(0.00267)\left(\frac{0.00635m}{2}\right)}{\frac{\pi}{64}(0.00635m)^2}$
$F_{pin} = 0.133N$	$M_z = 0.00267N \cdot m$	$\sigma_x = 4.293MPa$
$\tau_{xy} = \frac{Tc}{J}$	$\sigma_a{}' = \sqrt{\sigma_x{}^2 + 3\tau_{xy}{}^2}$	$N_{sf} = \frac{S_y}{\sigma_a{}'}$
$\tau_{xy} = \frac{(8.16\times10^{-3})\left(\frac{0.00635m}{2}\right)}{\frac{\pi}{32}(0.00635m)^2}$	$\sigma_a{}' = 12.122MPa$	$N_{sf} = \frac{275MPa}{12.122MPa}$
$\tau_{xy} = 6.545MPa$		$N_{sf} = 22.686$

Figure 7: Determining the factor of safety of the stepper motor shaft

Determining the factor of safety of the stepper motor shaft is to make sure the motor would not break after millions of rotations and the factor of safety is more than 22 which is more than enough for us.

$E_{steel} \gg E_{wood}$	$E_{steel} = 209GPa$	$\therefore L - Rail\ resists\ deflection\ more\ than\ track$
$I_{L-Rail} = \frac{bh^3}{12} + \frac{hb^3}{12}$		$\delta_{max} = \frac{Fl^3}{48EI}$
$I_{L-Rail} = \frac{(0.0015)(0.0184)^3}{12} + \frac{(0.0184)(0.0015)^3}{12}$		$\delta_{max} = \frac{(34.3N)(1.219m)^3}{48(209\times10^9)(7.84\times10^{-10})}$
$I_{L-Rail} = 7.84 \times 10^{-10} m^4$		$\delta_{max} = 0.0079mm$
$I_{wood} = \frac{bh^3}{12}$	$\delta_{max} = \frac{Fl^3}{48EI}$	
$I_{wood} = \frac{(0.125)(0.01)^3}{12}$	$\delta_{max} = \frac{(34.3N)(1.3)^3}{48(11\times10^{-9})(1\times10^{-8})}$	
$I_{wood} = 1 \times 10^{-8} m^4$	$\delta_{max} = 0.0137mm$	

Figure 8: Determining the deflections of wood and steel

A maximum deflection of 0.0079 mm and 0.0137 are also found. To determine the deflections of wood and steel L rail, we can ensure the material we choose to support the system is strong enough to resist any break of the system.

$$\dot{Q} = 0.2388\frac{L}{s} * \frac{0.001m^3}{L} \qquad v = \frac{4\dot{Q}}{\pi d^2} \qquad A = \pi r^2$$

$$\dot{Q} = 1.388 \times 10^{-4}\frac{m^3}{s} \qquad v = \frac{4(1.388\times10^{-4})}{\pi(0.009525)^2} \qquad A = \pi(0.009525)^2$$

$$v = 1.948\frac{m}{s} \qquad A = 2.850 \times 10^{-4}m^2$$

$$\overrightarrow{P_o} = \rho V v \qquad \vec{F} = \frac{\overrightarrow{dp}}{dt}$$

$$\overrightarrow{P_o} = \left(10^3\frac{kg}{m^3}\right)(3.048m)(2.85\times10^{-4}m^2)\left(1.948\frac{m}{s}\right) \qquad \vec{F} = \frac{1.692\frac{kg\cdot m}{s}}{\left(\frac{3.048m}{1.948m/s}\right)}$$

$$\overrightarrow{P_o} = 1.692\frac{kg\cdot m}{s} \qquad \vec{F} = 1.081N$$

Figure 9: Pump analysis

By determining the pressure of the water coming out from the pump, we are able to get the desired amount of water to use for the plants.

$$F_w = (0.0133kg)\left(9.81\frac{m}{s^2}\right) \qquad +\uparrow F_y = 0 = F_N - F_w$$

$$F_w = 0.130N \qquad F_N = 0.130N$$

$$\hookrightarrow \frac{F_N}{4} = 0.0326 \rightarrow For\ single\ wheel$$

$$\Sigma M = 0 = M_o - (F_N * l)$$

$$M_o = (0.130N)(0.024m)$$

$$M_o = 0.00312N\cdot m \rightarrow 0.00039N\cdot m\ \ for\ bending\ moments\ of\ each\ wheel\ bracket$$

Figure 10: Determining the forces on wheels and bending moment on wheel brackets

By determining the bending moment analysis, we are able to ensure the bending moment of each wheel bracket 0.00039 Nm is low enough to keep the sled stable no matter the housing system is located in anywhere of the sled.

FEA analysis was also performed on a Solidworks assembly of the rack, support bar, and mounting brackets. These components deserved extra attention because it was critical that the gear forces on the rack did not push the rack and pinion apart from each other anywhere along the motion. In the simulation, the gear forces were placed in the middle, to create the worst-case scenario where the bar's deflection would be highest. The static deflection to gravity was also used. A maximum stress of 8.9 MPa was found; this is well within acceptable limits as the yield strength for low carbon steels is 200 GPa. A maximum deflection of 0.28 mm is also found; this amount should be low enough to keep the gears meshing because the overall height of the gear teeth is around 2.2 mm.

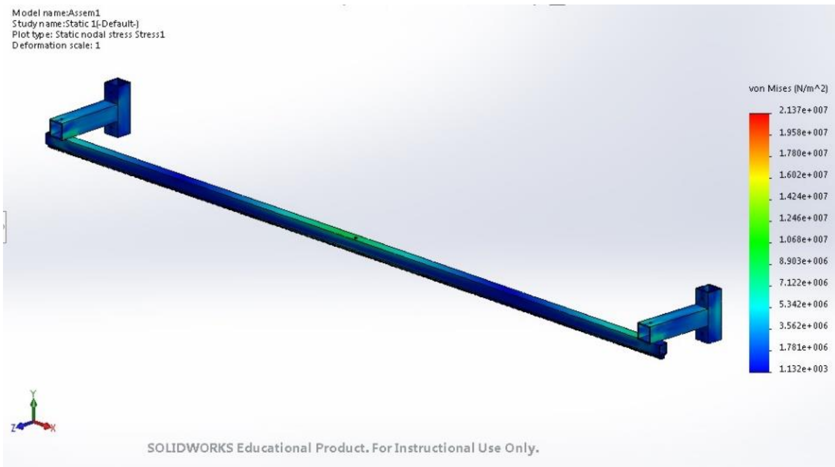


Figure 11: Stress Analysis of Rack Assembly (Max Stress = 8.9 MPa)

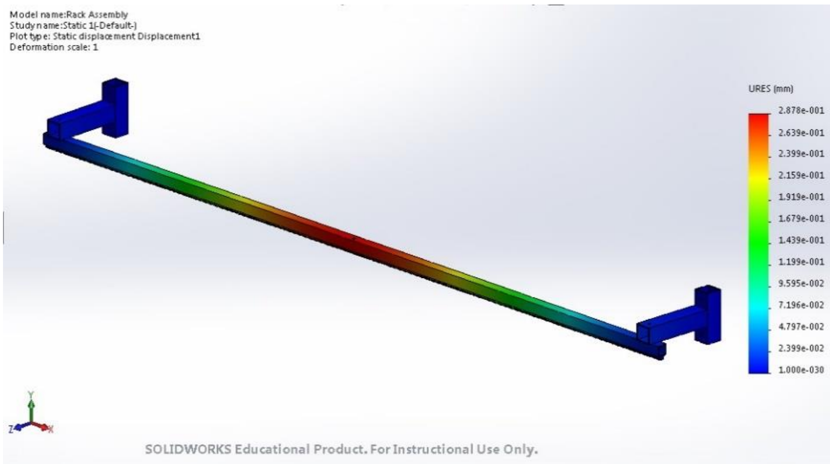


Figure 12: Displacement Analysis of Rack Assembly (Max Displacement = 0.28 mm)

Testing And Evaluation

Testing was conducted throughout the whole procedure of designing and manufacturing this mechanism in order to address problems as early as possible. To start, an appropriate material was to be chosen for the main demo enclosure. Wood was the first choice and with an estimate of the overall weight of the main housing and its components, a force far exceeding that of the mass was applied to the wood to determine if it was a suitable material. The wood passed the test.

The main method of movement, arguably the most important aspect of the design, was the rack and pinion system. The pinion gear fit well with the selected motor, however the purchased rack was very slim and could easily be bent by hand due to the length of it being 4ft. This was a major problem considering the design required the rack to be supported in the air by just two brackets at either end of it. Gravity alone would make the rack bow in the middle and the force of the pinion moving along the rack would cause it to bend even more. The solution to this problem was to mount the rack on a steel bar which added more than enough support to keep the rack straight and sturdy.

The material for the main housing was chosen to be clear acrylic due to its light weight and strength. The first prototype was cut using a band saw and glued together using a strong adhesive. The adhesive held true, however, the rough cuts with the band saw weren't precise enough. While testing the mechanism, it was found that the wheels on the bottom plate weren't able to be screwed in due to misalignment of the housing. A whole new housing in order for the mechanism to function correctly. The band saw was used for the new bottom plate as well, but with an increased attention to the detail.

In the final design, the mechanism would be hooked up to a house via a hose. However, for the purpose of demonstration, a pump was needed in order to display the function of the machine. After the first test of the hose, it was found that it produced a stream way too powerful to water flowers. The solution to this problem was to print out a nozzle to better direct the stream, as well as clamp the hose to reduce the flow and velocity of the water when it reached the nozzle.

When testing this nozzle with the enclosure, another problem arose. The enclosure we made for the demonstration purposes was not sealed water tight. The pocket that was meant to act as a reservoir for the used water, leaked quite badly. To fix this problem, the pump was fully submerged in a leak proof bin, and the reservoir pocket was insulated using a garbage bag and tape. This method wasn't ideal, however due to the time constraints it was the only quick fix the group could think of.

The most extensive testing was done on the electrical components of the mechanism. Programming the Raspberry Pi to operate and control both stepper motors and the pump took many trials. Finding the correct timing and speed of the motors to go the correct distances proved to be quite challenging. Also, the wires connected to the Raspberry Pi would come loose which would cause problems. These problems just took time and patience to figure out the correct algorithm to execute the tasks the sprinkler was designed for. As with every early design, unforeseen problems arise. In the case of this mechanism however, many of the problems that came up were generally expected. From the start, the programming aspect was a point of concern, as it proved to be.

COST ANALYSIS

The estimated overall cost of the mechanism calculated in the preliminary stages of the design was around $406 CAD. In the end, the total cost came out to be $435.40, exceeding our original estimate. This final cost is excluding labour and any material supplied by the University of Guelph.

Below, Table 1 displays the purchased parts. This cost could have been reduced however. A large purchase of this mechanism was the acrylic to construct the main housing that held the motors and Raspberry Pi. The acrylic sheet purchased could have made at least 3 enclosures. Purchasing a smaller sheet would have brought the total cost down closer to the original estimate. Also, parts like the solenoid valve and the moisture sensors were not used in the final model. There were also some things necessary to buy for the prototype that would not be required on a production model, such as the black paint bought to paint the demo box. Along with most prototypes, with better planning, many excessive purchases could have been avoided. Better planning would result in less wasted material which would greatly reduce the cost. With mass production, leftover materials would be reused in the construction of another unit. This means that the end price of $435.40, could be reduced to an approximate $362.3 per unit which would be below the original estimate.

Table 1: Bill of Materials

Bill of Materials			
Item	Store	Quantity	Price ($)
Water solenoid Valve	Elmwood Electronics	1	$9.99
Raspberry Pi Stacking Header		1	3.99
NEMA-17 Stepper Motor	Canada Robotix	1	18.99
NEMA-23 Stepper Motor	Robotshop	1	45.18
Stepper Motor Hat for Raspberry Pi		1	27.69
4ft 1/4" Rack Gear	McMaster Carr	1	39.23
15 Tooth Pinion Gear		1	43.47
12V Brushless DC Water Pump	Amazon	1	18.99
Fel-Pro Gasket Material		1	11.09
USB Controller		1	19.99
Soil Moisture Sensor		1	8.98
Hose Clamps	Home Depot	3	4.14
CIL Black Paint		1	19.97
1 1/2" Wood Screws		1	12.98
M4 Bolts + Nuts		4	2.04
1" 4-40 Machine Screws		1	8.94
MC Bolts		1	2.97
Adhesive Repair Xtreme	Rona	1	9.34
Acrylic Clear Sheet		1	76.49
Vinyl Tubing		1	0.85
Total w/ Tax and Shipping			$435.40

Design Evaluation

The first prototype of the TrueRain system performed the tasks it was designed to do and the design was more or less well done. However, after the evaluation, of the performance at the trade show and reflection of the initial constraints, it was apparent that there were some aspects of the design that could be improved.

The wooden demonstration enclosure was well built in terms of structural integrity. During the whole testing phase, moving it from various locations and the trade show performance it stayed strong. There was a major problem with it however. The slot cut out for the reservoir was not water tight. The water would leak through the bottom which was a major problem. For future demonstrations a plastic container will have to be used as a reservoir for the pump.

The systems means of movement worked very well. The rack and pinon binded well and the chosen motor had no problems moving the weight of the sled even with all of the cords and hose attached to it. The only problem was that there is a noise associated with the motor while its moving along the rack. Oil was added to the rack in order to reduce the amount of friction and in turn reducing the amount of noise between the pinion gear and rack but that had very little effect. This is a very small problem and there isn't an easy solution apart from selecting a quieter motor. The other motor that controlled the angle of the nozzle worked flawlessly.

The pump used to propel the water was slightly overpowered for the purposes of the demonstration but it ended up working out after kinking the hose slightly to reduce the flow. When the pump was activated, the water started spraying from the nozzle quickly and pretty accurately. However, when the pump turned off, the water didn't stop flowing immediately. The stream slowly retreated and water ended up dripping inside of the enclosure. This was a major safety concern seeing as water and electricity can be very dangerous. Luckily everything was water tight, but this was a flaw in the design and will have to be addressed in the final product.

The acrylic housing which held all of the motors and the Raspberry Pi held together perfectly. There were no signs of the adhesive loosening up and the gasket material bolted to the bottom plate proved useful in keeping the excess water from spreading to the different compartments holding electrical parts and wires.

All of the electrical aspects of the mechanism functioned as well as could be expected. The remote controller had a very quick response time and had no problems operating throughout the whole presentation. The only problem was the accuracy of the system. Since the system was being operated by a human, it was only as accurate as the eye of the operator. It was difficult to hit the exact spots where the "plants" were located. With further programming, this problem could be addressed and the system could hit the exact locations.

Overall, the TrueRain system performed all of the tasks it was supposed to do. The end performance was not perfect but it did perform very well and the group is happy with the end result.

CONCLUSION

Based on the outcome of this project, TrueRain was a successful feat. The sprinkler encountered numerous obstacles, however, all were overcome in the end to produce an amazing mechanism that wowed many judges and onlookers. The machine was able to comply with many of the criteria and constraints made initially, such as having a good factor of safety and being able to hit within a 2ft by 4ft enclosure. The smart sprinkler was also able to be controlled to a great level of accuracy along the rack and adjust itself without fault to aim for the targeted location. The use of the clear acrylic housing proved a welcoming sight to judges as they were impressed by the simplicity of the interior mechanical components of the machine. Some weaknesses observed of the TrueRain was the over budgeted cost of the prototype and the fact that the sprinkler could not produce a continuous jet while rolling along the rack. The cost of the mechanism ended up being approximately $200 over the allotted budget. Further, the motors used in the prototype were not meant to be running for the three hours that was the Trade Show. From this, overexertion of the two motors was encountered, heating up the plexi-glass enclosure to a somewhat alarming level. Another weakness was that some of the screw holes were not drilled in the correct location, making it difficult to ensure that pieces within the enclosure were stabilized and not able to be bounced around.

In the future, a more advanced prototype of TrueRain will be produced. This new prototype will allow the mechanism to be pre-programmed with a program allowing its user to be able to set the exact locations of the plants as well as their specific watering regimes without having to remotely operate the sprinkler when in use. Further, this robot will be set to water the plants at the base, so as to ensure the leaves are not damaged by the jet stream as it rains down from above.

The TrueRain smart sprinkler was intended for use in small hobby gardens or in greenhouses. It should be noted that the electrical aspects of the sprinkler system should not get water on them as it would result in the machine not functioning properly. It is also recommended that the TrueRain sprinkler not be run along the track whilst the nozzle also be adjusted as it will result in failure of the mechanism.

APPENDIX

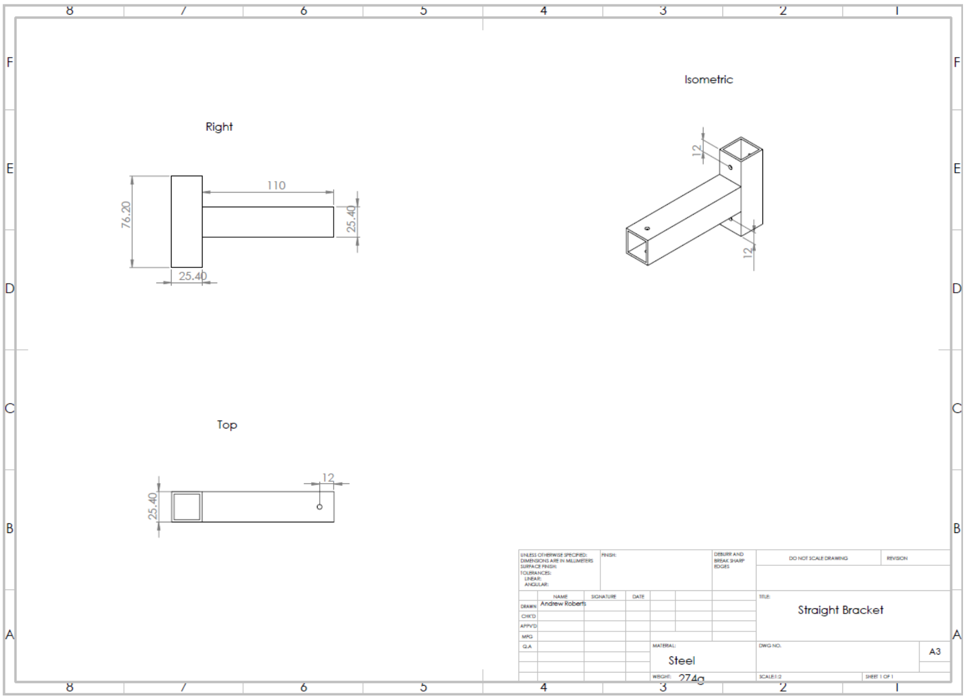


Figure 13: Straight Bracket CAD

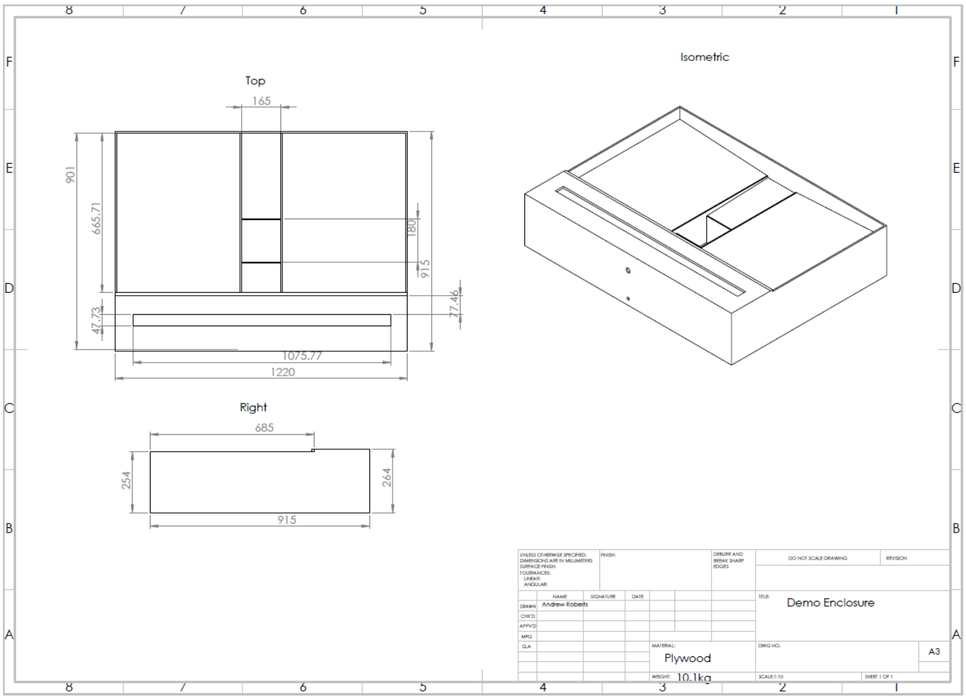


Figure 14: Demo Enclosure CAD

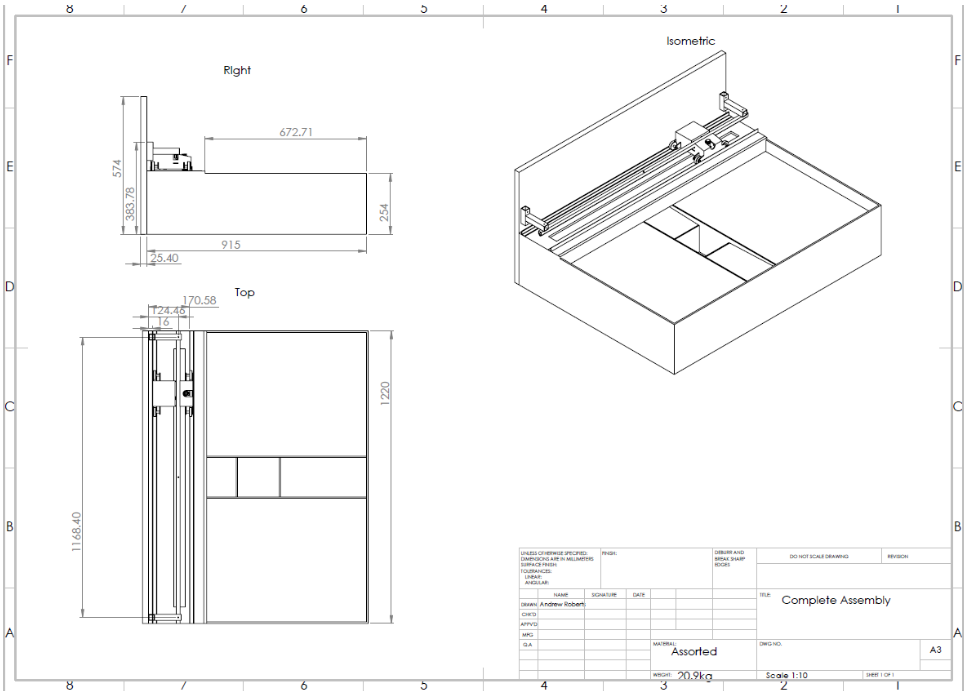


Figure 15: Complete Assembly CAD

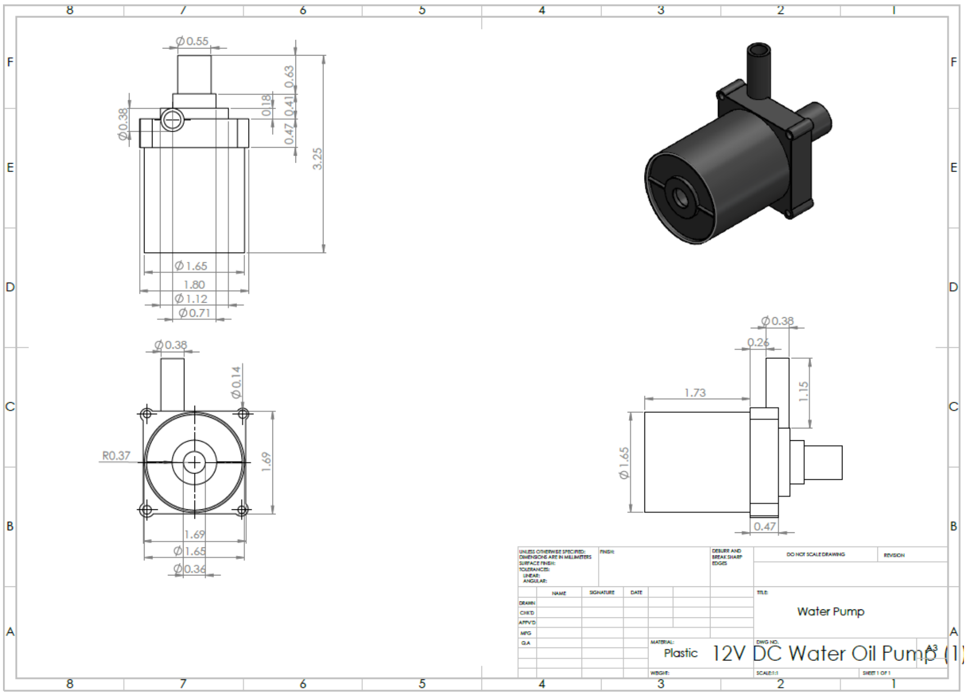


Figure 16: Pump CAD

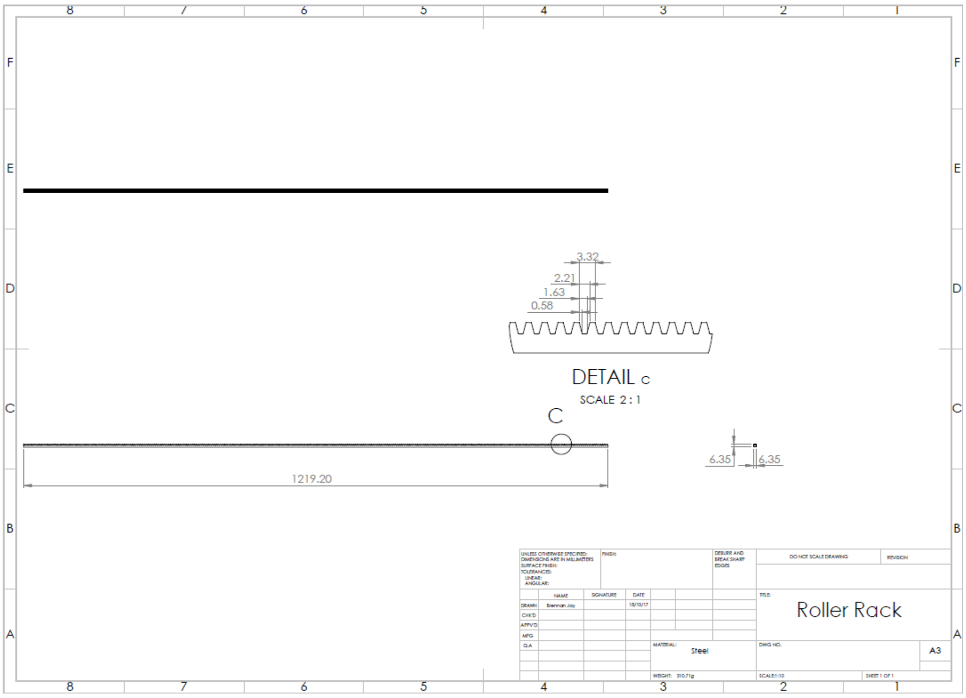


Figure 17: Rack CAD

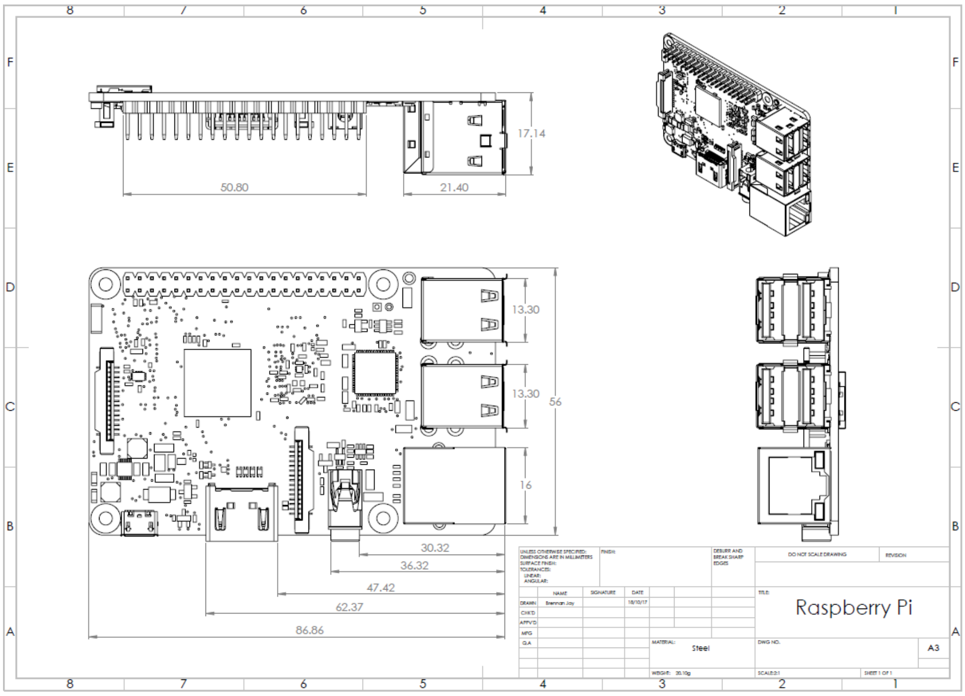


Figure 18: Raspberry Pi CAD

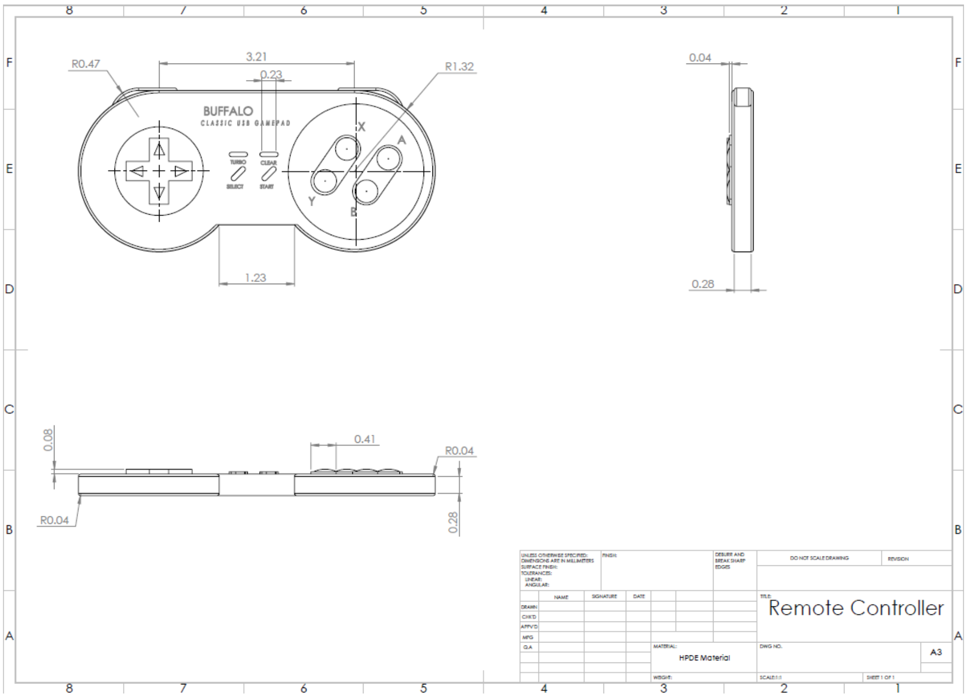


Figure 19: Remote Controller CAD

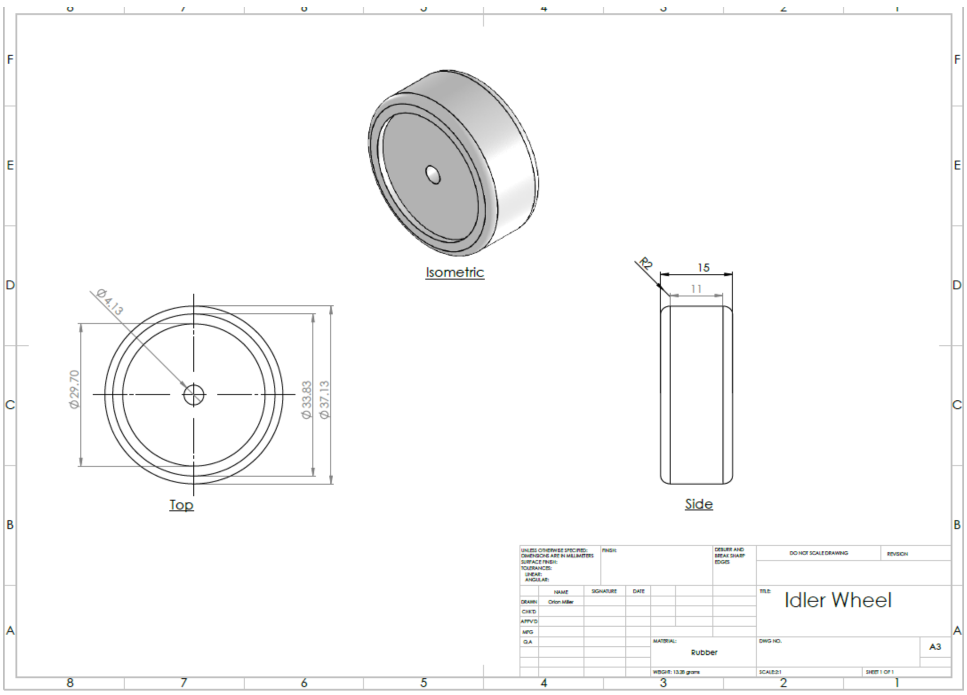


Figure 20: Wheel CAD

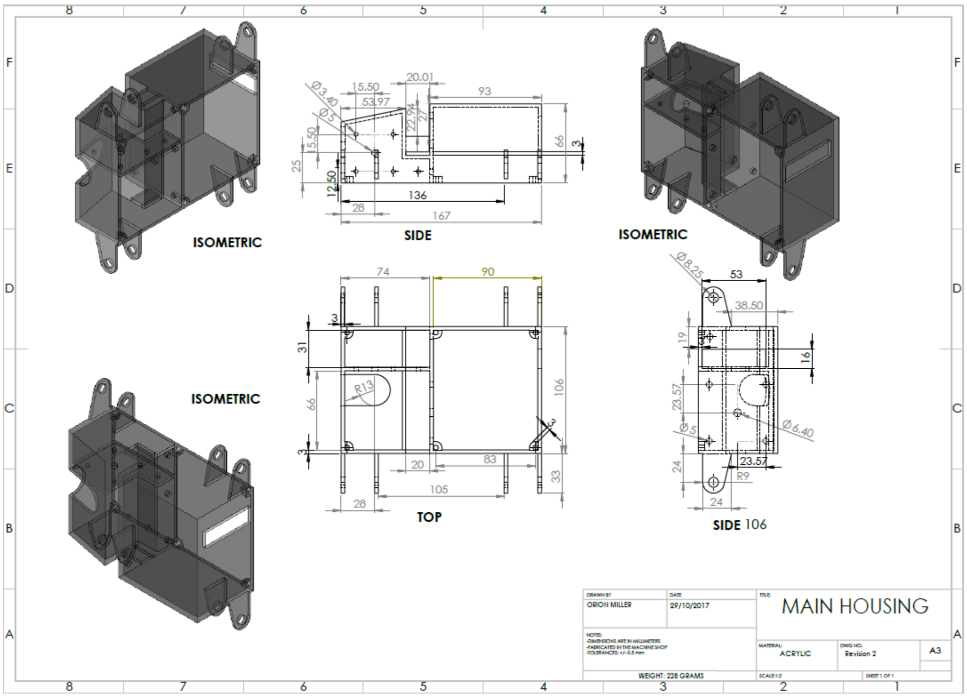


Figure 21: Main Housing CAD

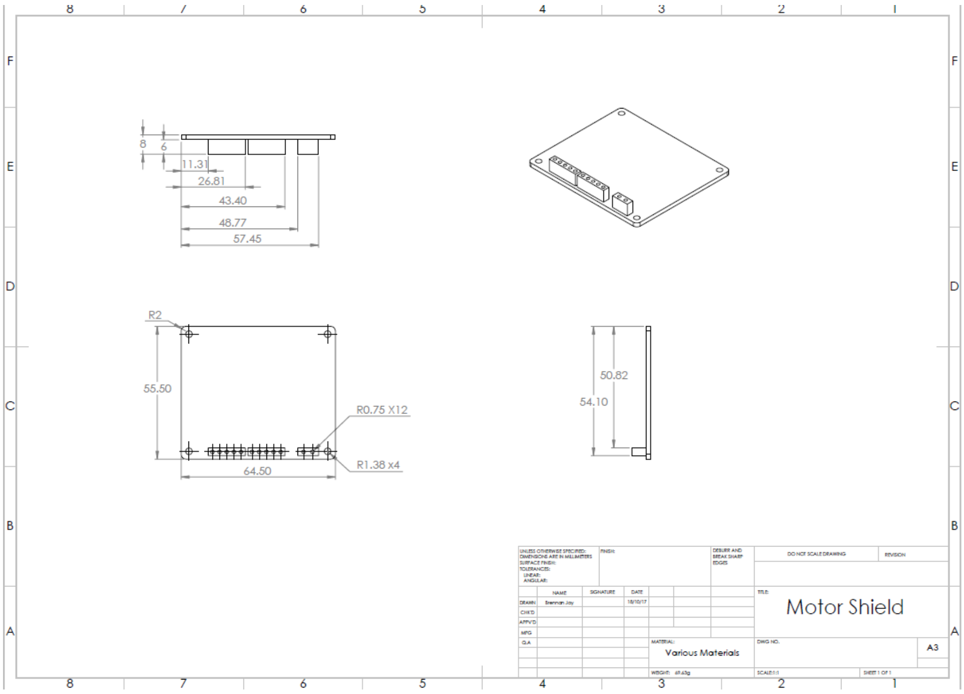


Figure 22: Motor Shield CAD

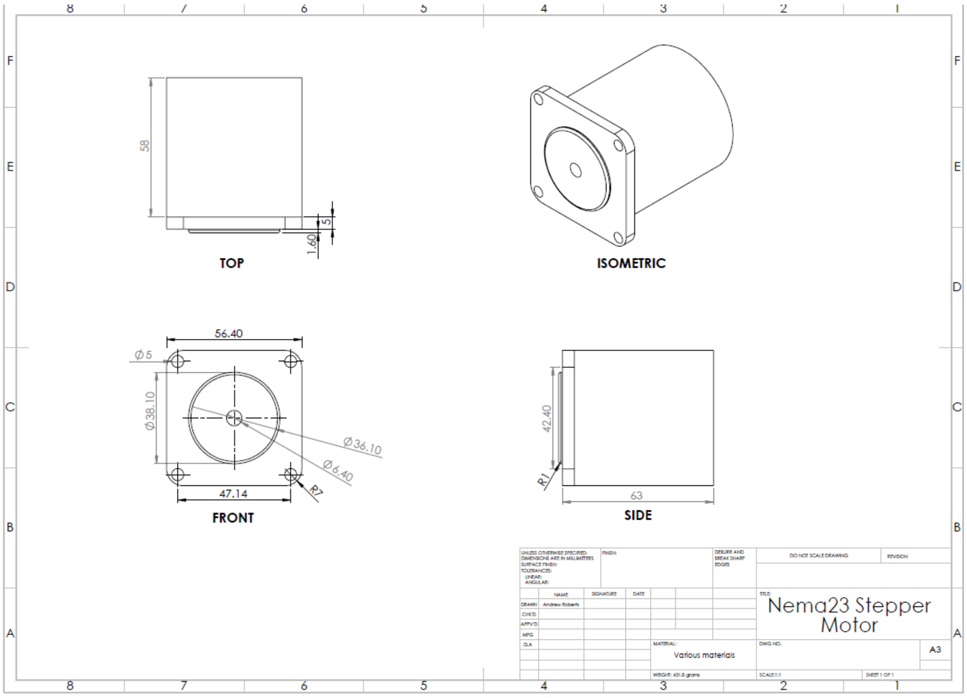


Figure 23: NEMA 23 Stepper Motor CAD

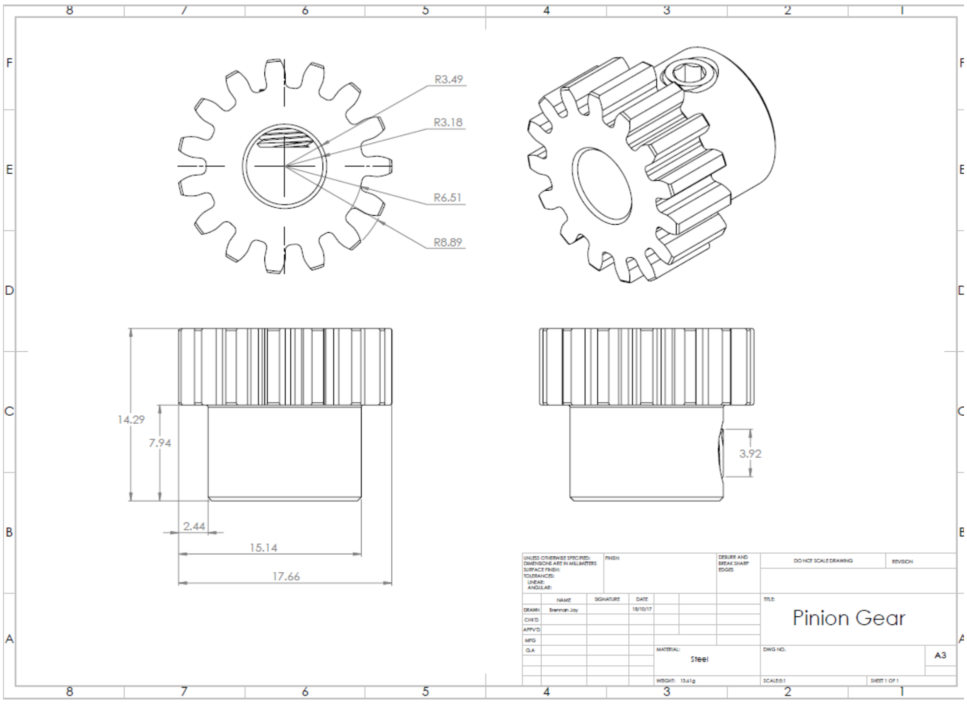


Figure 24: Pinion Gear CAD

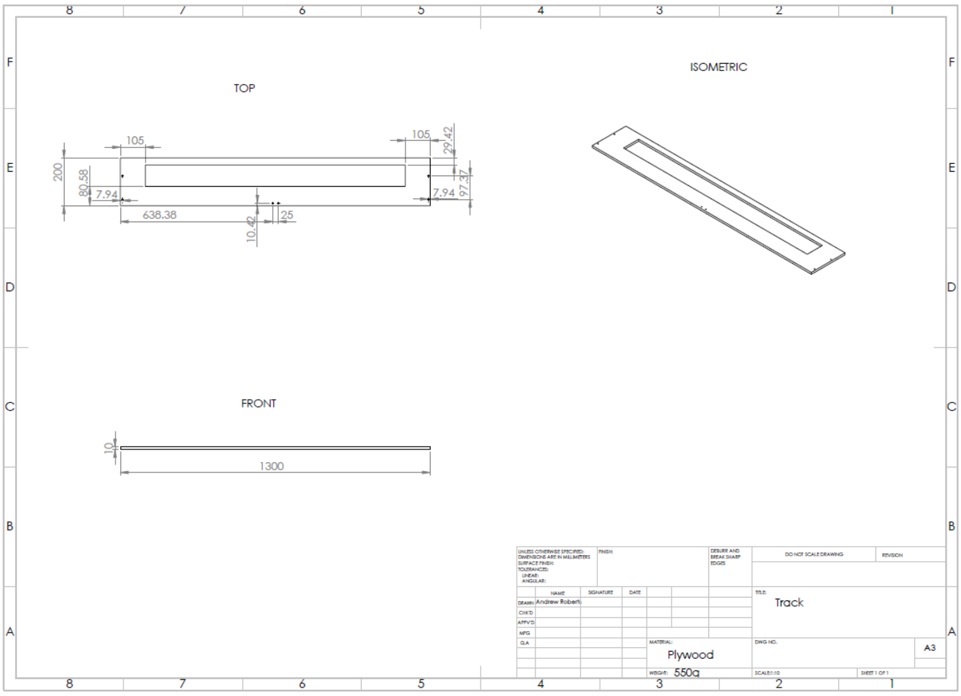


Figure 25: Enclosure Track CAD

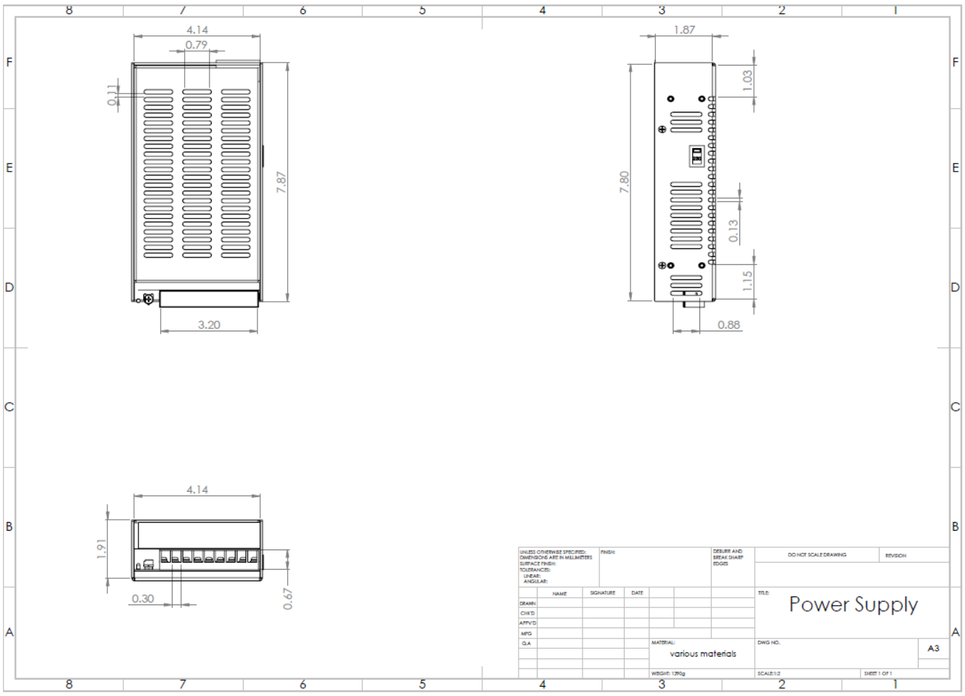


Figure 26: Power Supply CAD

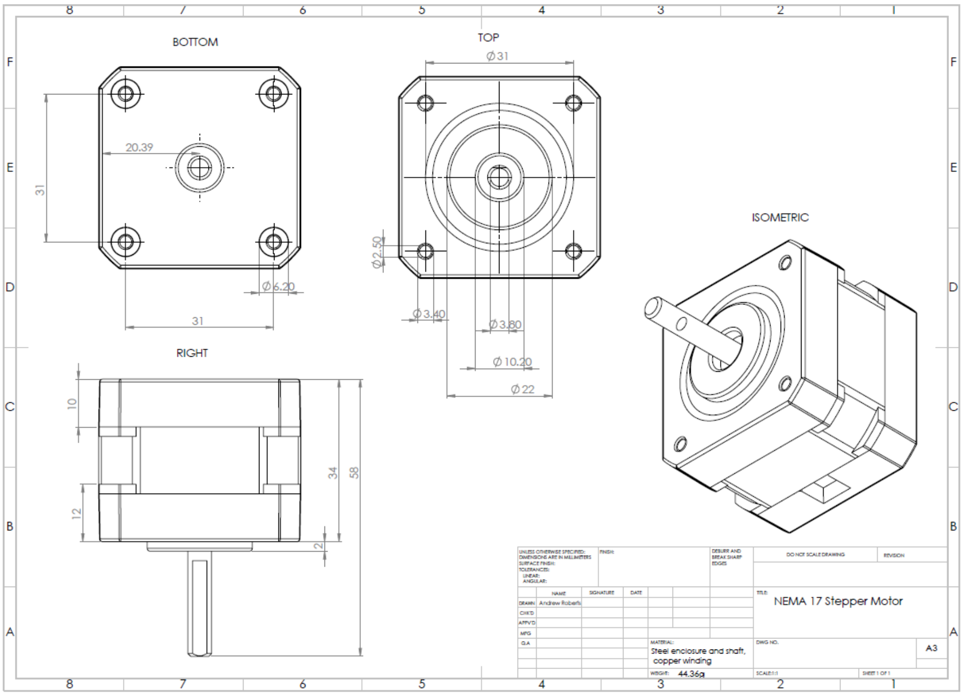
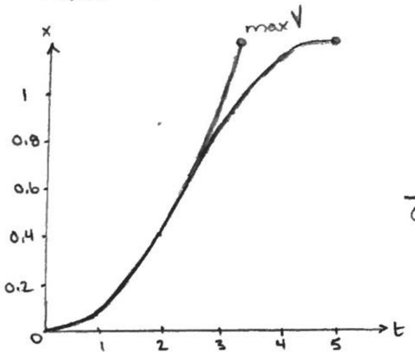


Figure 27: NEMA 17 Stepper Motor CAD

DETERMINE TORQUE IN STEPPER MOTOR

- Travel 1.2m in 5s.



Achieve max vel. @ 2.5s

$$\frac{\Delta x}{\Delta t} = \frac{1.2}{1.5} = 0.8\,m/s$$

$$\vec{a} = \frac{0.8\,m/s}{2.5\,s} = 0.32\,m/s^2$$

$m = 3.4kg$

$\hookrightarrow F = ma$

$= (3.4kg)(0.32\,m/s^2)$

$= 1.088N$

Assume l is 7.5mm

$\hookrightarrow T = (0.0075)(1.088)$

$= 8.16 \times 10^{-3}\ N\cdot m$

Figure 28: Calculations to determine the torque required from the stepper motor

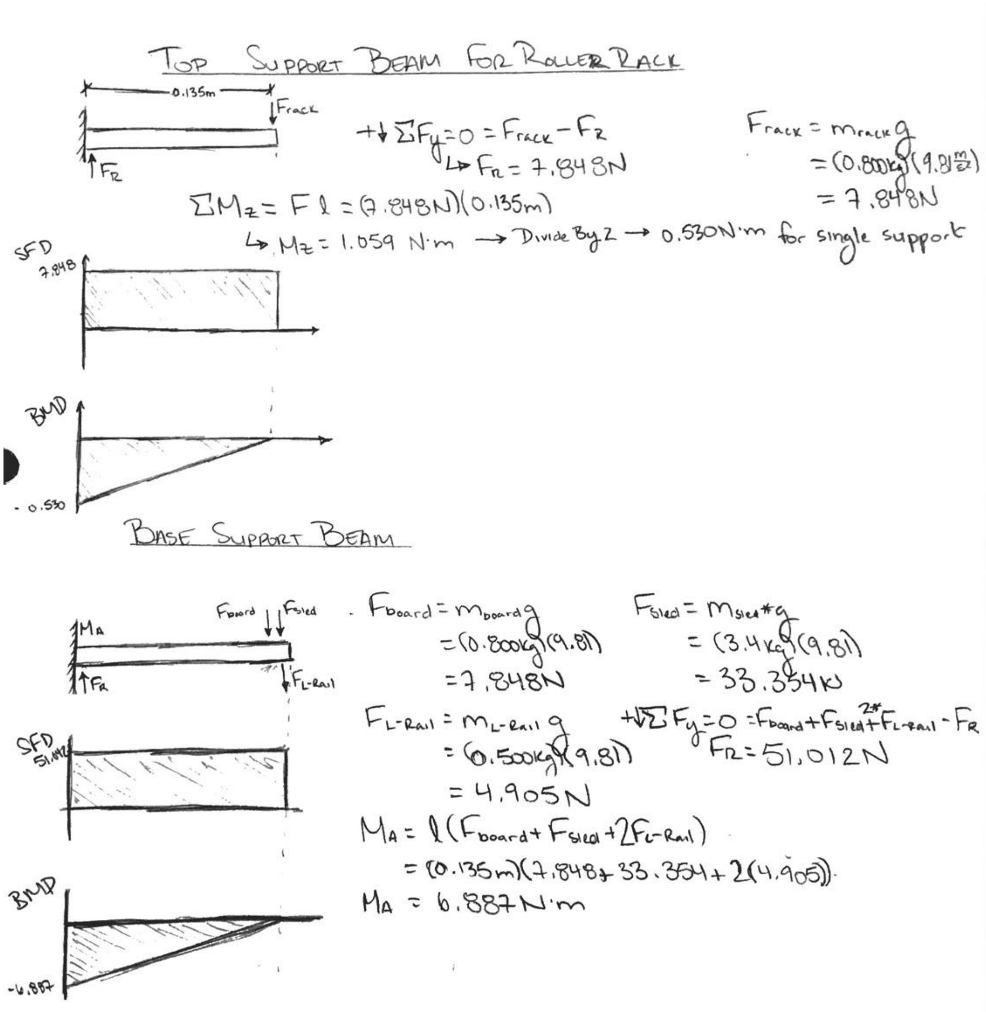


Figure 29: Calculations for support beams

PINION GEAR

$\frac{D_P}{2} = 0.00762\,m = r$ Torque of Motor $= 125\,oz\cdot in = 0.883\,N\cdot m$

Transmitted Force: $T = F_t r \rightarrow F_t = \frac{T}{r} = \frac{0.883\,N\cdot m}{0.00762\,m} = 115.879\,N$

Surface Speed: $V_m = \frac{\pi D_p n}{1000} = \frac{\pi(0.0152\,m)(15)}{1000} = 7.163\times10^{-4}\,m/min$
$= 1.19\times10^{-5}\,m/s$

Dynamic Force for Commercial Gears:
$F_d = \frac{600+V_m}{600} F_t = \frac{600+(1.19\times10^{-5})}{600} * (115.879) = 115.879\,N$

Allowable Bending Force: $S_n = 1270\,MPa$, $Y = 0.245$, $b = 0.00635\,m$, $P_d = 0.0159\,m$
$F_s = \frac{S_n Y b}{P_d} = \frac{(1270)(0.245)(0.00635)}{0.0159} = 111.54\,N$

Factor of Safety: $\frac{F_d}{N_{sf}} \geq F_s \longrightarrow N_{sf} \geq \frac{F_d}{F_s} \longrightarrow N_{sf} \geq \frac{115.879}{111.54}$
$\hookrightarrow N_{sf} \geq 1.039$

Figure 30: Equations to find factor of safety of pinion gear

Pump Analysis

Flow Rate: $\dot{Q} = 0.1388\ L/s * \frac{0.001\,m^3}{L} = 1.388\times10^{-4}\ m^3/s$

$$\dot{Q} = 1.388\times10^{-4}\ m^3/s$$

Velocity: $V = \frac{4\cdot\dot{Q}}{\pi(d^2)} = \frac{4(1.388\times10^{-4})}{\pi(0.009525)^2} = 1.948\ m/s$

$$V = 1.948\ m/s$$

Area of Hose: $A = \pi r^2 = \pi(0.009525\,m)^2 = 2.850\times10^{-4}\ m^2$

$$A = 2.850\times10^{-4}\ m^2$$

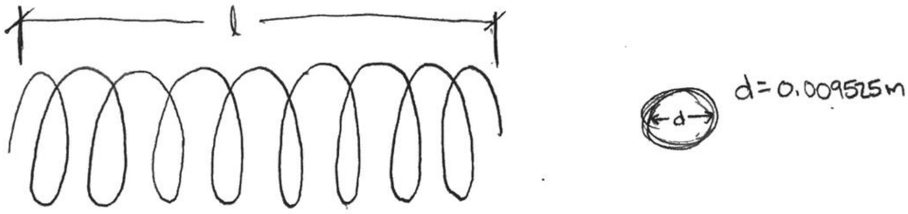
Forces: $\vec{P}_o = mV = \rho\forall v = (10^3\ \frac{kg}{m^3})(lAv) = (10^3)(3.048m)(2.850\times10^{-4})(1.948)$

$= 1.692\ \frac{kg\cdot m}{s}$

$$\vec{P}_o = 1.692\ \frac{kg\cdot m}{s}$$

$\vec{F} = \frac{d\vec{p}}{dt} = \frac{1.692\ kg\,m/s}{(3.048m/1.948m/s)} = 1.081\ N$

$$F = 1.081N$$



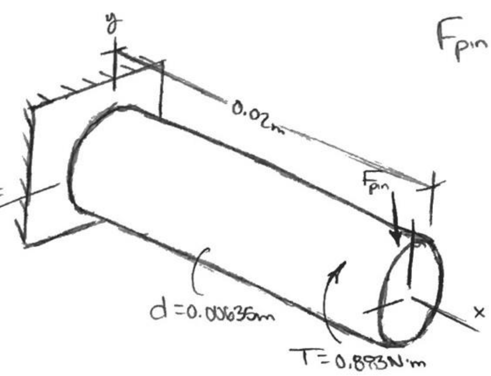
$l = 10ft = 3.048m$

Figure 31: Calculations for pump analysis

Stepper Motor Shaft

$\varnothing = 0.00635,\ l = 0.02m,\ T_{max} = 0.883N\cdot m,\ T_{req'd} = 8.16\times10^{-3}\ N\cdot m$

$m_{shaft} = 0.02492kg,\ m_{pinion} = 0.0136kg,\ S_y = 275$



$F_{pin} = m_{pin}g$
$= (0.0136kg)(9.81)$
$= 0.133N$

$M_z = F_{pin} * l$
$= (0.133)(0.02)$
$= 2.67\times10^{-3}$

$\sigma_x = \frac{M_z c}{I_z} = \frac{(2.67\times10^{-3})(\frac{0.00635}{2})}{\frac{\pi}{64}(0.00635)^2} = 4.283MPa$

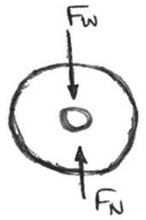
$\tau_{xy} = \frac{Tc}{J} = \frac{(8.16\times10^{-3})(\frac{0.00635}{2})}{\pi/32\,(0.00635)^2} = 6.545MPa$

$\sigma_a' = \sqrt{\sigma_x^2 + 3\tau_{xy}^2} = 12.118\ MPa$

$n_{sf} = \frac{S_y}{\sigma_a'} = \frac{275}{12.118} = 22.69$

Figure 32: Calculations for shaft of stepper motor

Forces in Wheels



$+\uparrow \Sigma F_y = 0 = F_N - F_W$

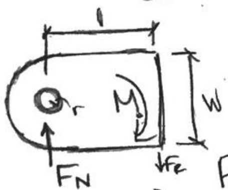
$\hookrightarrow F_N = 33.354$

↑ Total Force Acting on all 4 Wheels

$F_W = (3.4\,Kg)(9.81\,\frac{m}{s^2})$
$= 33.354\,Kg$

$\frac{F_N}{4} = 8.338\,N$ ← For single wheel Reaction Force

Force on Support Bracket of Wheels



$l = 0.024\,m,\ r = 0.004125\,m,\ h = 0.003\,m,\ W = 0.01775\,m$

For Bending

For $\frac{d}{h} \geq 0.25$, $K_t = A_e^{(b(d/W))}$

$\frac{d}{h} = \frac{8.25\times10^{-3}}{0.003} = \underline{2.75}$ $\therefore A = 1.80820$, $b = -0.166702$

$\frac{d}{W} = \frac{8.25\times10^{-3}}{0.01775} = 0.465 \longrightarrow K_t = A_e^{(b(d/W))}$
$= (1.80820)^{(-0.166702(0.465))}$

Geometric Stress Concentration Factor $= 0.832$

$\curvearrowright + \Sigma M = 0 = M_o - F_N * l$
$M_o = (33.354\,N)(0.024\,m)$
$= 0.800\,N{\cdot}m \longrightarrow$ Total Bending Moment
$= 0.100\,N{\cdot}m \longrightarrow$ For each of 8 Support Brackets

Figure 33: Calculations for forces acting on wheels and bending moment on brackets

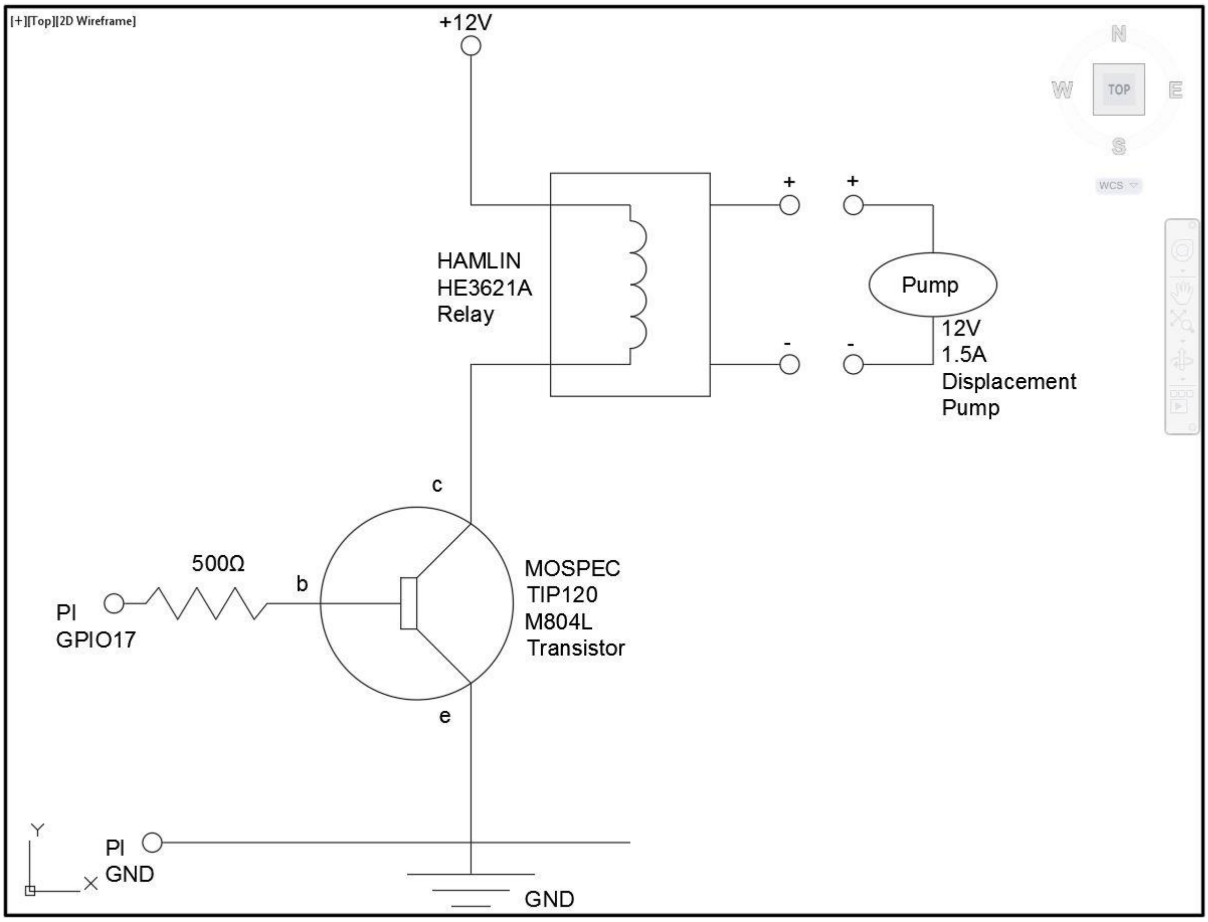


Figure 34: AutoCAD electrical diagram of circuit used to implement programming

Figure 35: Raspberry Pi Code

The following program "pad2.py" controls the machine's motion through a usb controller input (this is '/dev/input/event1' in the program):

```
Import RPI.GPIO as GPIO
Import atexit
from Adafruit_MotorHAT import Adafruit_MotorHAT, Adafruit_StepperMotor
From evdev import InputDevice, categorize, ecodes

GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup (17, GPIO.OUT) # GPIO pin that will control pump
Pump = 0 # pump is initially off

Gamepad = InputDevice('/dev/input/event1')

Button4 = 291
Button3 = 290
Button2 = 289
Button1 = 288
Rtrig1 = 293

Mh = adafruit_MotorHAT()

Def turnOffMotors():
      Mh.getMotor(1).run(Adafruit_MotorHAT.RELEASE)
      Mh.getMotor(2).run(Adafruit_MotorHAT.RELEASE)
      Mh.getMotor(3).run(Adafruit_MotorHAT.RELEASE)
      Mh.getMotor(4).run(Adafruit_MotorHAT.RELEASE)

Def turnOffGPIO():
      GPIO.output(17, GPIO.LOW)

Atexit.register(turOffGPIO)
Atexit.register(turnOffMotors)

MyStepper1 = mh.getStepper(200,1)
MyStepper1.setSpeed(500)
MyStepper2 = mh.getStepper(200,2)
MyStepper2.setSpeed(500)

XAxis = 0
Theta = 87

For event in gamepade.read_loop():
      If event.value == 1:
            If event.code == button4:
                  If xAxis > 0:
                        MyStepper1.step(320,
                  Adafruit_MotorHAT.BACKWARD,
                  Adafruit_MotorHAT.DOUBLE)
                        xAxis= xAxis - 3.5 # change x location by 3.5
                  inches
```

```
elif event.code == button3:
      If theta > 45:
            MyStepper2.step(1, Adafruit_MotorHAT.BACKWARD,
      Adafruit_MotorHAT.DOUBLE)
            Theta = theta - 3.6 # change theta by  - 2
      degrees

elif event.code == button2:
      If xAxis < 42:
            MyStepper1.step(320,
      Adafruit_MotorHAT.FORWARD, Adafruit_MotorHAT.DOUBLE)
            xAxis= xAxis - 3.5 # change x location by 3.5
      inches

elif event.code == button1:
      If theta < 87:
            MyStepper2.step(1, Adafruit_MotorHAT.FORWARD,
      Adafruit_MotorHAT.DOUBLE)
            theta = theta + 3.6 # change theta location by
      + 2 degrees

elif event.code == rtrig1:
      If pump == 0:
            GPIO.output(17,GPIO.HIGH)
            Pump = 1
      Elif pump == 1:
            GPIO.output(17,GPIO.LOW)
            Pump = 0
```