University of Massachusetts Dartmouth

Department of Mechanical Engineering

MNE 498 – Mechanical Engineering Design Project

LUBRICANT EFFECTIVENESS CHARACTERIZER

Project Report

Submitted to *Nye Lubricants Inc.*

Project and Report Created By:

Peter Lunn

Nathen Arruda

Cam Whittle

Peter McGrory

Ryan Proulx

ABSTRACT

This report gathers all the technical and business aspects of the "Lubricant Effectiveness Characterizer," previously known as "Bearing Torque Tester" and summarizes the efforts to meet the stated design specifications. Concepts including 3D modeling, calculation, simulation and prototyping will be elaborated upon, as well as important thought processes that led to critical design decisions. Electrical design, circuitry and programming will also be discussed, providing an insight into the Graphical User Interface and the brain of the machine. Furthermore, a final design render will be presented, broken down and discussed, and a detailed operations guide will be issued. Mechanical drawings with proper Geometrical Design & Tolerancing for every machined component will be available in *Appendix A.* A separate spreadsheet of the bill of materials will be submitted along with this report. Additional materials, including code and .pdf files of all drawings will also be separately submitted with the report. To track project progress, an online service called *Trello Board* was used to assign tasks to team members. An image of this service will be available in *Appendix B.* Additionally, two separate detailed project journals were updated throughout the project, one for the mechanical side and one for the electrical side. A project conclusion will be included in this report, which will cover the challenges faced during the year, final budget spending, and the overall purpose of the project. Lastly, final placement of the project will be revealed, as ranked by the judges. Acknowledgements we be also be provided in this section to thank everyone who gave countless hours to make this project prosperous.

TABLE OF CONTENTS

LIST OF FIGURES AND TABLES

LIST OF SYMBOLS

F.S.	Factor of Safety
σ	Stress
σ_u	Ultimate Stress
σ_{actual}	Actual Stress
$\sigma_{bending}$	Bending Stress
P	Pressure/Load
$P_{critical}$	Critical Pressure/Load
A	Area
δ	Deformation
L	Length
D	Diameter
E	Young's Modulus
S	Slenderness
I	Area Moment of Inertia
K	Effective Length Factor
M	Bending Moment
y	Perpendicular Distance From Neutral Axis
τ	Shear Stress
V	Shear Force
x	Position along X-axis
Bi	Biot Number
h	Heat Transfer Coefficient
L_C	Characteristic Length
k	Thermal Conductivity
T_{end}	Temperature at End of Shaft
α	Thermal Diffusivity
$T_{initial}$	Initial Temperature
T_∞	Temperature of Heat Source
C	Coefficient for Transient One-Dimensional Conduction
ζ	Coefficient for Transient One-Dimensional Conduction
t	Time

1. INTRODUCTION

In August of 2019, Dr. Jason Galary, director of Research and Development at *Nye Lubricants Inc.,* decided to sponsor Research and Development intern, Peter Lunn, for a senior capstone project at the University of Massachusetts Dartmouth. The project was designated the title, "Bearing Torque Tester," for which desired outcomes and broad requirements were provided. The project was later renamed "Lubricant Effectiveness Characterizer." After a team was assembled, the final group consisted of *Nye* interns and mechanical engineering undergraduates, Peter Lunn (team lead), Nathen Arruda, and Ryan Proulx, as well as computer/electrical engineering undergraduates, Cameron Whittle and Peter McGrory. *Nye Lubricants* Research and Development engineer, Richard Raithel (shown at right) was appointed the team's overall project manager.

Figure 1. Project Overseer Richard Raithel

As suggested in the name, the "Bearing Torque Tester" machine will be designed to measure and collect torque data from a spinning ball bearing. The torque it measures is the friction produced between the inner race and the outer race of the bearing. This torque value is meant to provide data on the effectiveness of how specific lubricants wear over time and under different environmental and loading conditions. The renaming of the project to "Lubricant Effectiveness Characterizer" made for a more accurate representation of the function of the machine.

At the beginning of the fall academic semester, *Nye* narrowed their broad requirements of the project. Dr. Jason Galary provided the team with the specifics and numerical necessities, which will be discussed under Section 2.1 Design Criteria and Logistics. In principle, the team is to design a machine capable of spinning, heating, and cooling a bearing and gathering torque data from said bearing, all while axially shearing the internal balls. This will require the purchase of equipment, such as electric motors, torque transducers, and heat torches. Hence, *Nye* has provided the team with a tentative budget. From a mechanical standpoint, *Nye* requested that the team have a fully designed, justified, and machined rig, which fulfilled all requirements by April 29, 2020.

Given the system requirements provided by Nye, the system needed to be controlled by a central, easy to operate, and robust graphical user interface (GUI). The Engineers at Nye expressed the need to have a UI that allowed the user to manually change parameters at run time: The length of the run time, the temperature of the grease in the bearing, the applied load on the bearing, and the speed of the bearing during the test. These controls must be present in the final UI in the form of limited user inputs. Along with these needs, the customer wanted the torque and temperature readings to be displayed in real time in the form of a dynamic line graph. This would allow the users to track the grease's effectiveness during the test visually. Additionally, for security reasons, auxiliary implementations needed to be considered such as a physical and electrical emergency brake and a dynamic speed ramp up in the motor controls to keep the system from starting at a maximum speed.

For the overall electrical design details, it was left open ended to the team. Decisions such as the microprocessors used, language used, and interface designing environment were left up for discussion. As a result, Cameron and Peter spent a majority of the first few months comparing different options and whittling the choices down to a finalized high-level systems architecture and software design. Both of these will be explored deeply in their respective sections 3.1 System Architecture and 3.2 Software Design. Given that this design, in essence, is a system of systems, an effective and timely communication architecture was necessary to make sure all data is moved around in real time. Once the components began to arrive and physical assembly had begun, the first step for the electrical team was to make sure that all of the components could be effectively used and operated efficiently with the chosen 3rd party application. These functional prototypes are important to as they lead the way for the final system integration and testing. The functional prototypes will be explored in detail in section 3.5 Functional Prototypes and the final design implementations will be discussed in section 3.6 Final System Integration.

2. MECHANCIAL DESIGN

2.1 DESIGN CRITERIA AND LOGISTICS

To commence work on the project, the team first needed to know specific customer (*Nye*) requirements and desired outcomes. When the project was assigned, *Nye* provided the broad requirements that the rig be able to spin a bearing, heat, cool, and apply an axial load to the bearing. These were later focused down to numerical values. Dr. Jason Galary of *Nye* provided the following requirements and desired outcomes:

- Spin a bearing at a maximum of 3500 RPM
- Heat the bearing to maximum 200°C
- Cool the bearing to minimum 0°C
- Axially load the bearing with maximum of 100 lbf
- Measure and collect torque data from bearing

This project called for the purchase of various components, some being relatively expensive. For this reason, *Nye* assigned the team a tentative budget of $30,000. During the design process, the team looked for ways to reduce cost, simplify manufacturing, and eliminate unnecessary purchases.

2.2 DESIGN CONSIDERATIONS

When meeting with *Nye*, it was important to identify critical design considerations that would later play a larger role in the ultimate design of the Torque Tester. The first design consideration identified by *Nye* was that the tester must have the ability to acquire high resolution data capable of measurements in units of in-oz. This is because of the sensitive nature of the resistance inside of a bearing as is. Bearings are made so that customers no longer need to worry about this resistance. The next design consideration is that the design must be easy to use. The procedure that is formulated for operation must be easily executable. This is to ensure that testing can be performed seamlessly with all of *Nye*'s other testing methods. The final design consideration was one identified by the team. While the budget provided was a generous $30,000, the cost should be kept to a minimum, where possible. The consideration that was of least importance was to maintain an economic design.

2.3 DESIGN OPTIONS

For this design, each major component had multiple designs that were considered and reiterated on in order to come up with a final design that accomplished the goals that *Nye* had set forth. One of the biggest considerations was the overall layout of the test machine which was to either mount the motor parallel to the test bearing with a belt connecting them or directly couple the motor to the inside of the bearing. In order to spin the inner race of the bearing, two additional designs were created one of which would be a cone that was pressed into the inner race and the other being a clamping system to hold the bearing. As per *Nye*'s requirements the bearing had to be heated so both conductive heating with a coil and forced convection using a heat torched were both looked at as potential options. Another challenge faced was developing a method for cooling the bearing so both a liquid cooling system that would have been incorporated into the cup and a vortex tube was looked at. When heating the bearing this created the issue of keeping the heat away from torque sensor as it could not handle large fluctuations in temperature so adding a ceramic portion to the shaft supporting the housing was considered. Lastly, one of the biggest challenges faced was taking the axial load from the bearing so that the torque sensor did not receive the force while also not interfering with the torque measurement. To accomplish this, both an air bearing and face-mount crossed-roller bearing were incorporated into different designs. All these decisions came with many other small details that had to be carefully considered in order to understand any tradeoffs that may be accompanied with the choices made. The final decision will be elaborated on in-depth in the following sections.

2.4 NARROWING DECISIONS

The two main driving factors behind all the decisions made throughout the design process were accuracy and cost. For the overall design, the parallel setup was chosen as the lack of a direct coupling of the motor to the bearing would create less vibration by utilizing a belt system to

connect the two. This had the added benefit of not needing to put the axial load on the motor while also keeping the overall height low as the machine tipping over was a concern. This did make for a more complicated system as a belt drive system and tensioner would be needed that could still handle the 3500rpm requirement set by *Nye*, however the tradeoff in complexity and increased footprint was worth the gain in accuracy. To spin the bearing, the simplest method of a cone-shaped spindle was chosen as it had the advantage of self-leveling and made mounting the bearing in the machine far easier. Thanks to the simplicity of the spindle design it also helped to reduce the cost as it was a very simple part to machine. For heating the bearing a heat torch was chosen as it was the only non-contact method of heating the bearing to the 200°C requirement that could be found. As for cooling, a vortex tube was chosen as it was another method of non-contact cooling that could achieve the 0°C goal and potentially even cooler if coupled with a system for cooling the incoming stream of air. Both the heat torch and vortex tube required a source of compressed air, but that was not an issue as the laboratory already had a supply setup throughout. The heat torch and vortex tube also had the advantage of being extremely simple and easy to control while keeping cost down as well as not interfering with the torque readings. Machinable ceramic was looked at to add between the bearing housing and bearing shaft to keep heat from transferring up the shaft, however it was determined that for the sake of costs it should not be added. For taking the axial load away from the torque sensor an air bearing was initially specified as this would have no interference with the torque readings. The air bearing had the added advantage of keeping heat away from the torque sensor as well thanks to a steady stream of compressed air running over the shaft and adding additional length as well rigidity. Ultimately this was changed to the crossed-roller bearing as the manufacturing costs of the air bearing housing would have been too expensive due to very tight geometric tolerancing.

2.5 FINAL DESIGN

After months of hefty analysis and numerous redesigns, the team narrowed down a final design in early March. The design, as seen in *Figure 2* below, utilized the parallel components design. As a tradeoff, this required a significant amount of solid metal. From the figure, four custom-made aluminum plates were required to support the design. These plates are represented by the number 1 in the figure. The other main structural component in the design is the twelve aluminum legs, represented by the number 2. At the very top of the design sits the torque transducer (red component), with its housing. This housing was originally to be made of aluminum, however, the team figured that it would be very cost and time efficient to 3D print it, using a high temperature resin. Furthermore, since it was printed from a resin LCD printer, the component was able to meet its tolerancing specification. The transducer and housing component are represented by the number 3 from the figure. On the left side of the main plate is the servo motor, marked with the number 4. This was chosen by the electrical engineering majors after extensive research. Ultimately, a motor was chosen that could be fully integrated with the design and could hit the 3500 RPM mark. Represented by the number 5 is the pneumatic piston. This piston was chosen to apply the 100 lbf axial load to the inner race of the bearing. It has a stroke length of 1 inch, which works perfectly to minimize height. After deciding to use a pulley system to drive the shaft, two stainless steel, 2.5 inch-diameter v-belt pulleys were implemented to support the 3500 RPM mark. The pulley in the middle is an idler pulley, which acts to provide tension to the v-belt. This preserves the life of the belt and reduces overall vibration. These pulleys are represented by the number 6 in the figure. One of the most complicated components to design was the test bearing cup, represented by the number 7. This component had three primary redesigns, each with pros and cons. The initial idea was to create a clamping ring around the outer race of the bearing that would hold it in place. Because of heating/cooling concerns, the idea was shifted to a clamping cup with a lip for the outer race to sit on. This was the first primary redesign. After concerns of design complexity and the ability to attach it to the peek shaft, the clamping idea was scrapped. The next redesign brought the cup significantly closer to the final design. This version utilized a slide-in design, as seen in the figure, with two screws on the top to attach to the

peek shaft. The bottom of the cup required a ring to prevent the unloaded bearing from falling out. Concerns about stress points and machinability arose, which prompted the team to make a final adjustment. Instead of one solid piece, the piece was split in two; the retaining ring and the cup. This design can be seen in more detail in *Figures 22 & 23* in *Appendix A.* Number 8 on the figure represents the thrust bearing shaft, made of peek plastic. This component was also one of the most complex to design. As discussed in Section 2.4, the initial design for this shaft was to implement an air bearing. However, after budgetary concerns, *Nye* opted to find an alternative for the air bearing. This prompted the team to rapidly find a solution. After implementing a thrust bearing, the team quickly spotted a flaw added by the thrust bearing. Since the thrust bearing did not need nearly as much height as the air bearing and since the transducer can only handle 93°C, heating became an issue. As seen in *Figure 3* under Section 2.7, stainless was no longer an option since it heats at a relatively rapid rate. After days of calculations and considerations, peek turned out to be the best option. It could handle both compressive and torsional load and only reached about 30°C on the transducer side. The thrust bearing is marked with the number 9. This thrust bearing will add resistance to our shaft, which ideally should have zero added resistance. However, as long as the added resistance by the thrust bearing is consistent throughout every test, then it will be considered "normalized" into testing. The component labeled number 10 is the nozzle, which will deliver hot or cold air to the bearing. It has one inlet specifically for the heat torch and another specifically for the vortex tube. The second bearing cup in the design, marked number 11, is simply to provide the hardened stainless shaft with a place to spin. A simple design flaw was spotted in this cup in late February. Since two bearings are press-fitted into the cup it would have been nearly impossible to remove these bearings if the cup was one solid piece. Instead, the cup was separated into a top and bottom to easily allow the bearings to be pushed out. As simple as this catch sounds, it would have costed the team lots of time and money in the process. The wire-looking piece marked with the number 12 is the load cell. This will work to ensure the proper load is being applied to the bearing. This component was tricky to incorporate. Initially, the team tried to implement a button load cell where the support cup could sit on top of. After some concern about slipping or bending of the shaft, the team decided that a double

sided, threaded load cell would allow the load cell to screw into the piston plate and support cup. The shaft on the left side of the rig, labeled number 13 is the steel motor shaft. The shaft on the right side, also labeled number 13 is the hardened stainless bearing shaft. Like the support bearing cup, there was a last-minute flaw that was spotted in the shaft. The thrust bushings for which the shaft rests in requires the shaft to be hardened. If the shaft had not been hardened, the bushing balls would have ruined the shaft, consequentially costing the team valuable time and money. The fully finished rig stood at a height of 2'5", a width of 15", and a depth of 10". An exploded view of this model can be seen in *Figure 21* in Section 2.10.

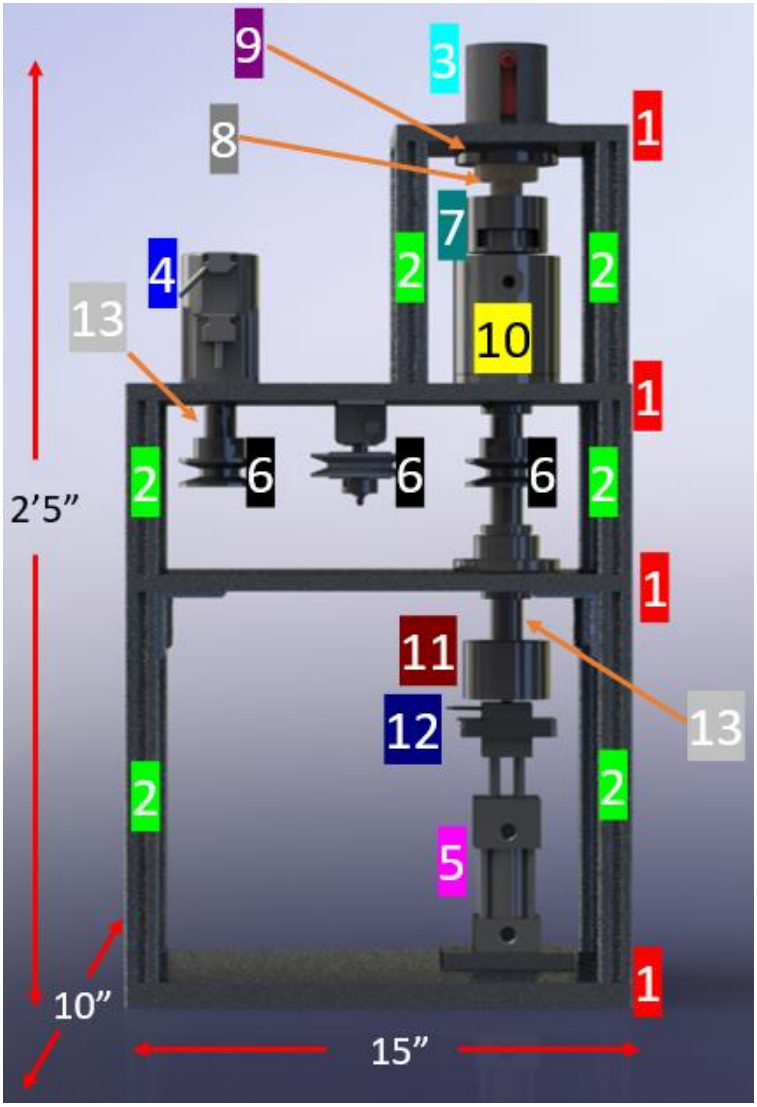

Figure 2. Final Design Component Callout

2.6 MECHANICAL DRAWINGS

After finalization of the design and model of the rig, the team then constructed mechanical drawings for each component. These were also made in SolidWorks for easy referencing. For about two months, the mechanical members worked with both Richard Raithel and *Nye* machinist, Mason Wood, to create and perfect the drawings. After countless reiterations and drawing tweaks, the team produced a whopping nineteen total drawings. Each drawing was thoroughly drawn with very specific views, correct dimensions, appropriate tolerancing, and attention to extremely fine detail. The drawings were thoroughly inspected by Mason to ensure that the machine shop would have no issues machining the parts. These painstaking measures were taken so that the team could have the machined parts returned as quickly as possible and without time and money consuming errors. The drawn components are as follows:

1. Bearing Cup Disc
2. Bearing Cup Drawing
3. Bottom Legs
4. Bottom Plate
5. Frame Bushing Ring
6. Main Plate
7. Mid Legs
8. Mid Plate
9. Motor Shaft
10. Piston Adapter
11. Piston Cup Adapter
12. Piston Cup Bottom
13. Piston Cup Top
14. Simplified Cylinder Nozzle
15. Test Bearing Shaft
16. Thrust Bearing Shaft
17. Top Bracket
18. Top Legs
19. Torque Sensor Housing

For images of all drawings, see *Appendix A*.

2.7 THEORETICAL CALCULATIONS

In order to justify the final design and obtain a general idea of how to the material will withstand the forces exhorted on the structure, quick and rough hand calculation can be done. These calculations will consist of the bending of the top plate as well as the bending if the bracket at the top of the assembly. Buckling of the shaft as well as the compressive stress induce. The tensile stress in all the 8020-extrusion. The heat transfer through the shaft and to the torque transducer. All calculations will be done using simplified geometry and are used to get an idea of the physics at play. Various assumptions will also be made for the same reason and will later be justified using simulations.

To begin, the tensile stress in the 8020 extrusion will be considered. All the known parameters such as cross-sectional area, and young's modulus can be seen in *Table* **1**. The maximum load exhorted by the pneumatic piston will be 100 lbs. The load will be distributed through the four lengths of the 8020 extrusion. Assuming the load is distributed evenly, each length will be subjected to a load of 25 lbs. Using the following equation, the stress can be found in each of the legs.

$$\sigma = \frac{P}{A}$$

Where P is the load applied to each leg and the A is the cross-sectional area of the 8020 extrusion. Having determined the stress in tension, the known ultimate tensile stress UTS can be compared and a factor of safety can be generated for the design.

$$F.S. = \frac{\sigma_u}{\sigma_{actual}}$$

The factor of safety calculated is 543 and is an extremely high. As rigidity is something that is every necessary to minimize vibration, the stabilization that this supply is very important and for that reason the factor of safety does not need to be reduced or optimized.

Furthermore, the deformation of the supports of length 10.75in, 5.125in, and 7.224in seen in *Figures* 24, 38, 39 can be computed to verify no tolerancing issues can arise from the loading using the equation for axial deformation below.

$$\delta = \frac{\sigma L}{E}$$

Where L is the length of the respective support and E is the young's modulus for the 8020-aluminum extrusion.

Next, it is important to analyses the shaft that is being loaded into the bearing. Assuming the bearing is a ridged support and assuming pure compression with no rotation in the shaft, the computation can be made simple and hand calculations can be performed. The analysis to check for buckling and yielding will be done.

A diagram of the shaft and loading can be seen in *Figure*

37 along with a list of known parameters such as the young's modulus of the shaft and various geometric parameters necessary for the calculations. The shaft is under the full load of the pneumatic piston which is 100lbs. The stress and deformation, in the circular cross-section can be determined similarly to that of the supports. To perform the buckling analysis, it is important to observe the slenderness ratio given by the following equation.

$$S = \frac{L}{D}$$

Where L is the length of the shaft and D is the diameter. When this ratio is below 40, the yielding of the member is the only consideration. If the S us between 40 and 120 then the both yielding and buckling must be considered and then finally when S is above 120 then the yielding must be considered because it will always buckle before it yields. As the design of this shaft was changing throughout the year it was important to observe buckling even if it wasn't happening just in case a change was made where buckling occurred.

When observing buckling scenarios, there are 3 conditions. Condition 1 is a situation where both supports are pins, condition 2 is a scenario where both the supports are fixed, the 3 scenario is where one end is fixed and the other is free and finally the 4th condition is when one end is fixed and the other is free.

When finally having the Buckling factor, the following equation can be used to calculate the critical load where the member will buckle.

$$P_{critical} = \frac{\pi^2 EI}{(KL)^2}$$

Where E is the young's modulus for the peek shaft, and I is the area moment of inertia. With the properties of the Peek shaft available the critical value can be determined. Knowing the loading that will be applied a factor of safety for buckling can be determined with the following equation.

$$F.S._{buckling} = \frac{P_{critical}}{P}$$

Where $P_{Critical}$ is the buckling loading and P is the normal 100lbs. loading that will be applied from the piston. If the buckling is of major concern then the critical stress can be calculated and if the yielding is of a concern then then stress can be calculated but either can be calculated using simple tools such as MATLAB so that an iterative approach can be taken during the design of the shaft.

The next features analyzed on the design were the two plates holding the assembly together. The top bracket where the torque transducer, and the bottom plate holding all of the major components such as the motor and the bearing holding apparatus will be experiencing bending due to the pneumatic piston. In order to perform the analysis, a 2D projection can be viewed from the side and the analysis can begin.

The top bracket will be the first to observe. All known values such as material length, width, height and other key properties cam be viewed in *Figure 38*. The shear and bending moment diagram can be derived.

Both ends where assumed to be pinned joints as an effort to minimize the complexity of the calculations. Utilizing the equation below the max shear stress and bending stress can be determined.

$$\sigma_{Bending} = \frac{My}{I}$$

$$\tau = \frac{V}{A}$$

Furthermore, the resulting maximum deflection can be determined using the following equation.

$$\delta = \frac{Px^3}{48EI}$$

Where P is the point load being applied, x is the position along the x direction of the beam, and I is the first moment of area. A factor of safety for the design can now be established and used to compare with other iterative design decisions within a software such as MATLAB.

Similarly, the bottom plate holding components such as the motor and bearing apparatus can be analyzed. The shear and bending moment diagram can be seen in *Figure 42 and 43*.

It is important that the torque transducer read accurate data throughout the length of the test. In order to ensure that this is the case, extra caution must be taken when ensuring that the length of the bearing shaft does not overheat. To make any crucial design decisions to the shaft a thermal analysis must be performed. The geometry of the shaft is determined but the material is not. Two materials, Peek and Stainless Steel were observed for their thermal conductivity and material strength.

To begin the analysis, it is important to determine the Biot number. If the Biot number is less than 0.1 then the model can be solved using the method of lumped capacitance and if the model is over 0.1 a more complex method utilizing, and error function must be considered. The equation to determine the Biot number is shown below.

$$Bi = \frac{hL_C}{k}$$

Where L_c is the characteristic length of the geometry being observed. Both methods can be used to determine the temperature at any point in time for the end of the shaft, or where the transducer will be placed. In the case where lumped capacitance can be used, the following equation can be used.

$$T_{end} = \exp\left(-\frac{\frac{hL_c\alpha t}{k}}{L_c^2}(T_{initial} - T_\infty) + T_\infty\right)$$

Where h is the convective coefficient of air in natural convection, alpha is the thermal diffusivity, k is the thermal conductivity of the material and the T_∞ is the temperature of the heat source. In the case where the Biot number is above 0.1 then an error function must be used. The following equation can be used for that case.

$$T_{end} = (T_{initial} - T_\infty) * C * \exp\left(-\frac{\zeta^2\alpha t}{L_c^2}\right) + T_\infty$$

With the following equations, a simple code can be written in MATLAB and a sweep of time can be plotted for both stainless steel and Peek and the two curves can be compared. The plot can be seen in *Figure 3* on the next page.

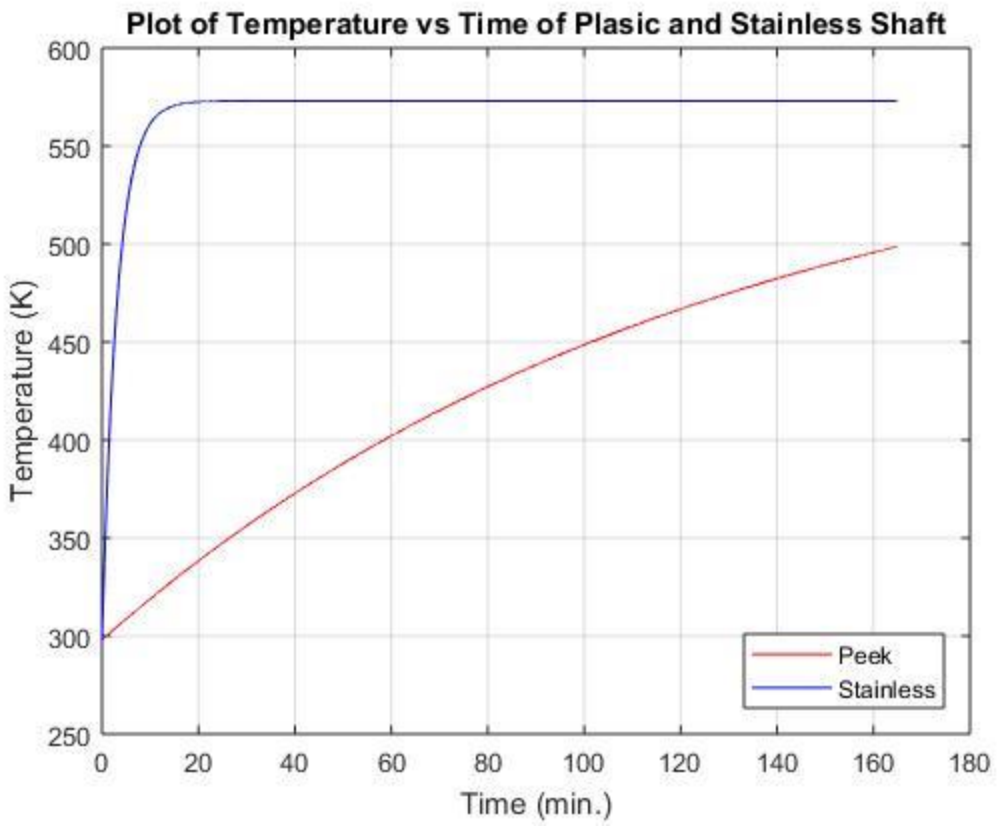

Figure 3. Peek vs Stainless Steel plot for error function equation.

2.8 SIMULATIONS

For the project simulations were done on the three load bearing plates which are the bearing bracket, top plate, and bottom plate which were all intended to be made from 6061 T6 Aluminum. The purpose for the simulations was to ensure that a reasonable factor of safety was being maintained and that there was minimal deformation of the plates as that could impact accuracy of the test. Additionally, the bearing cup and shaft were also run through a thermal simulation to ensure that the heat would not reach the torque sensor as this component was highly sensitive to increased temperatures.

For the aluminum plates *Table 1* shows the material specifications used to obtain the results in the three simulations that were done on them. Each of the plates was fixed around the outside bolt

holes where they would have been fixed in place by the legs in the assembly. The load of 100lbf was then placed around the inner bolt holes where the plates would have been loaded during the actual testing of the bearing.

Table 1: Properties used for aluminum plates in COMSOL

Material:	6061-T6 Aluminum
Young's Modulus:	68.9 GPa
Poisson's Ratio:	0.33
Density:	2700 $^{kg}/_{m^3}$
Yield Stress:	40000 psi
Force:	100 lbf
Mesh:	Normal

From the simulations that were done on the plates the maximum stress found was right around the bolt holes on the bearing bracket as this is one of the smallest areas where the load was applied to. The stress at this point was 4531.70 psi putting the factor of safety at 8.82 which was the lowest

found in the whole design; however this was a worst case with the actual design having the stress spread out more around the holes than what was done in the simulation.

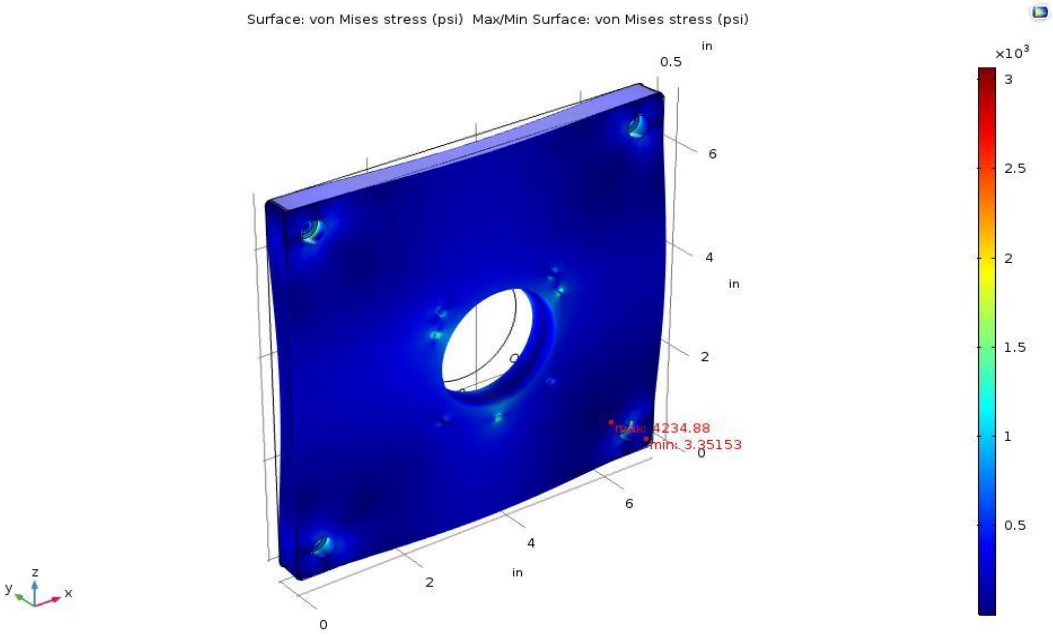

Figure 4. Von Mises stress in bearing bracket

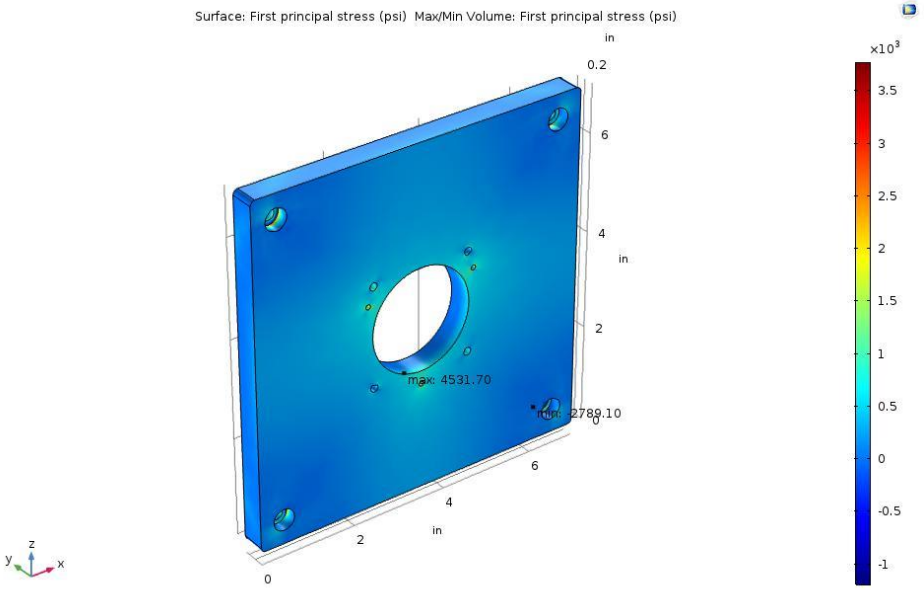

Figure 5. First principal stress in bearing bracket

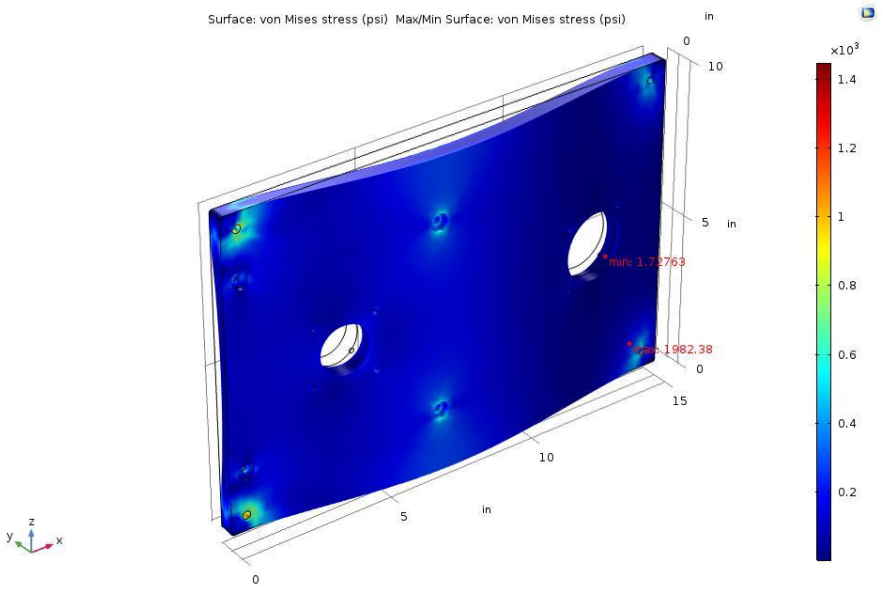

Figure 6. Von Mises stress in top plate

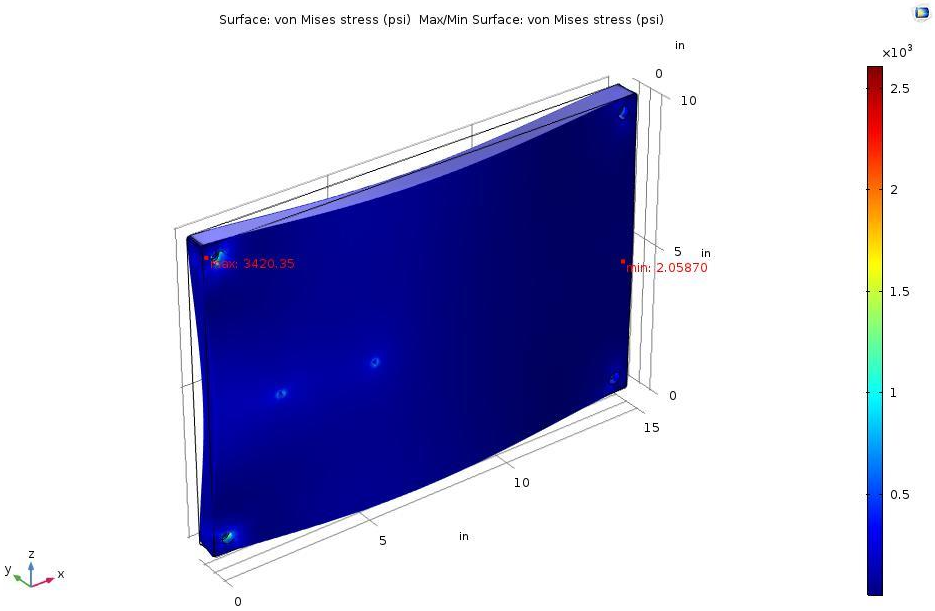

Figure 7. Von Mises Stress in bottom plate

In the larger plates the same simulation was done however the maximum stress found was much lower as the load was more evenly spread out. For these components, the factor of safety was substantially higher despite being simulated under less than ideal conditions to get an idea for what would happen in a worst-case scenario.

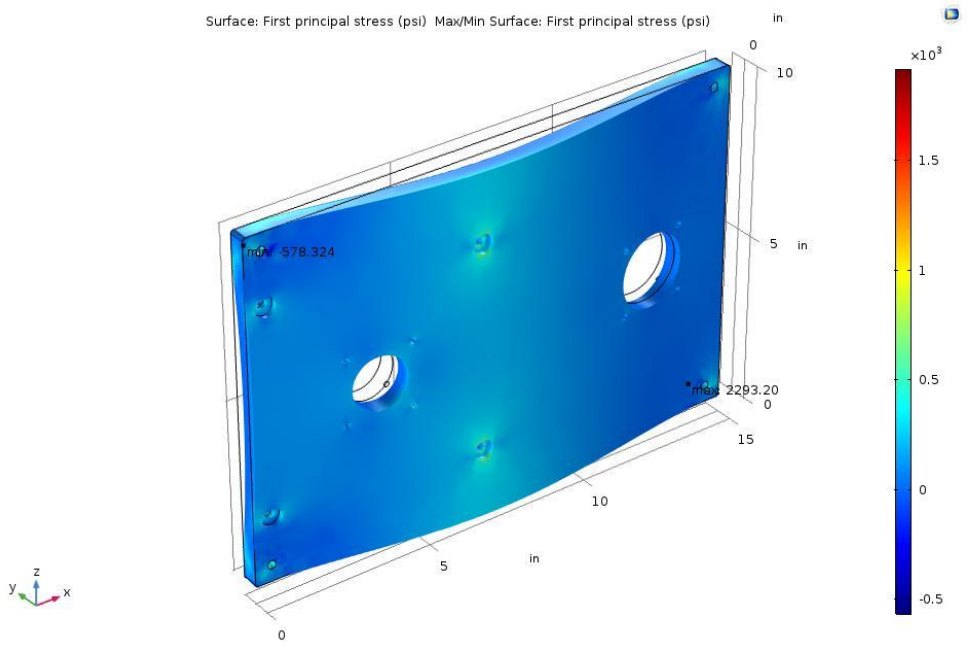

Figure 8. First principal stress in top plate

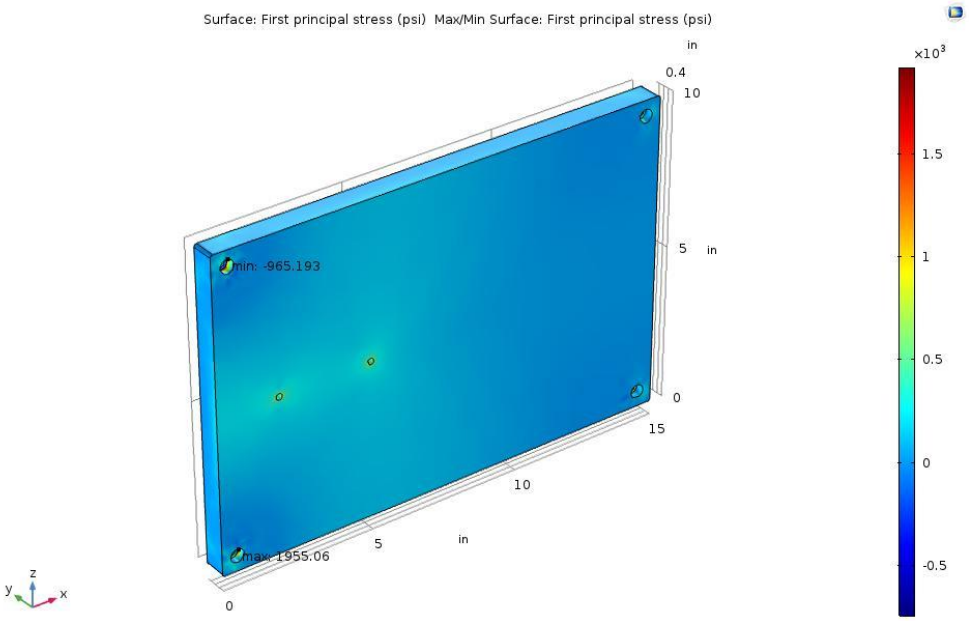

Figure 9. First principal stress in bottom plate

In addition to looking at stress in the plates there was also concern for the deflection that would occur under loading. If the deflection were too high that could cause the bearing supporting the shaft to bind or put additional stress on the torque sensor as it could not take being axially loaded. To check this in COMSOL a cut plane was added to the center of each plate and the deformation at that cut was plotted in order to see how much deflection there was. In the center line of each of the plates there are one or two holes which are the reasons for the blank areas in the graphs below. The peak deflection in all of them was right around 0.001 inches which was under our tightest tolerances for machining of the plates at 0.005 inches, so the deflection was a non-issue in the design.

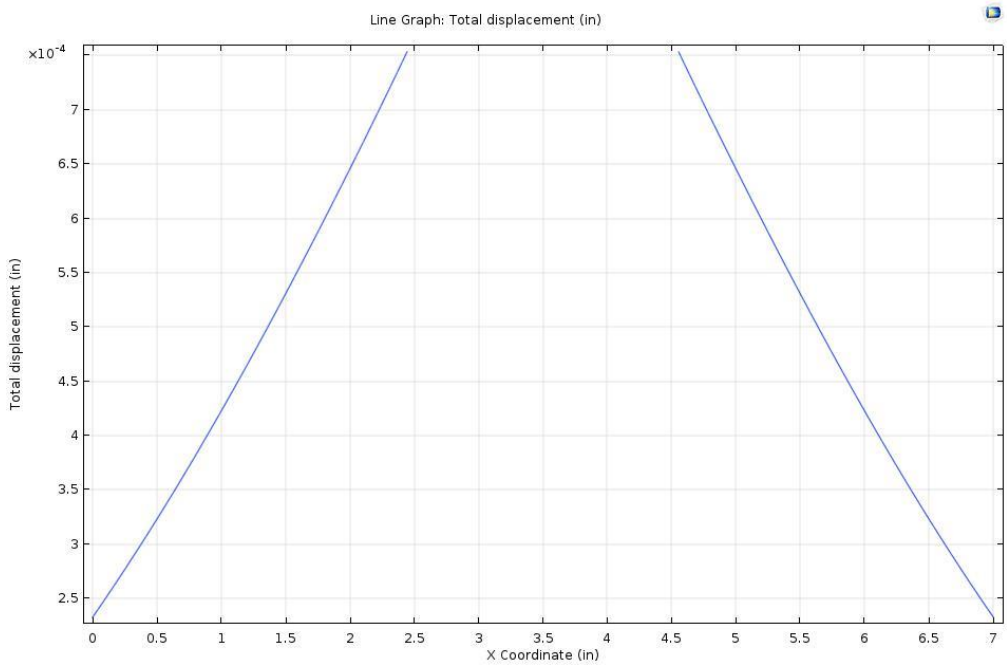

Figure 10. Bearing bracket displacement along center line

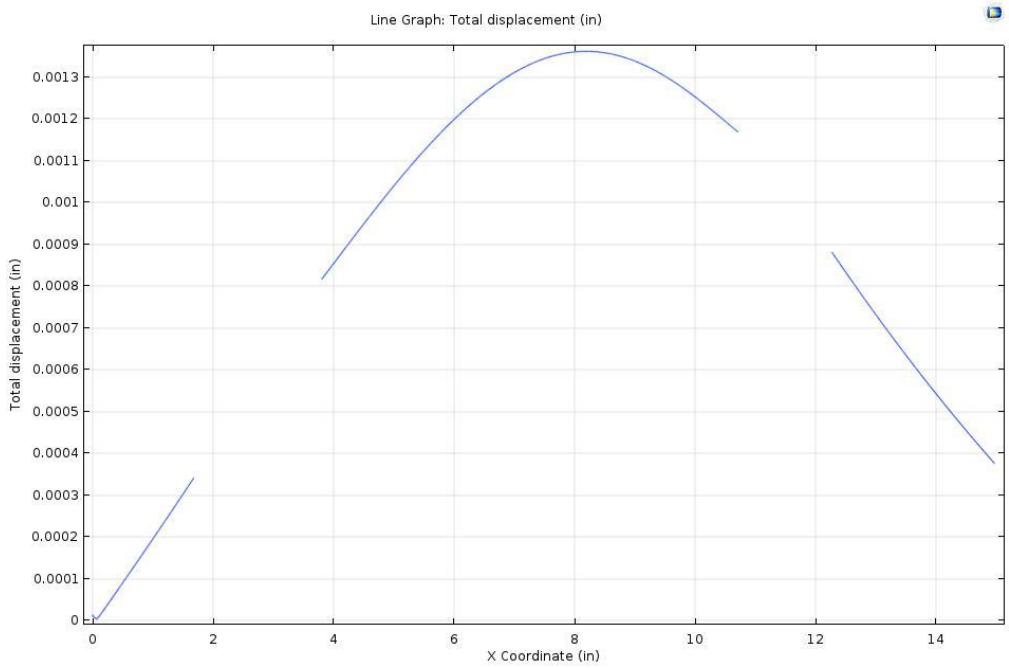

Figure 11. Top plate displacement along center line

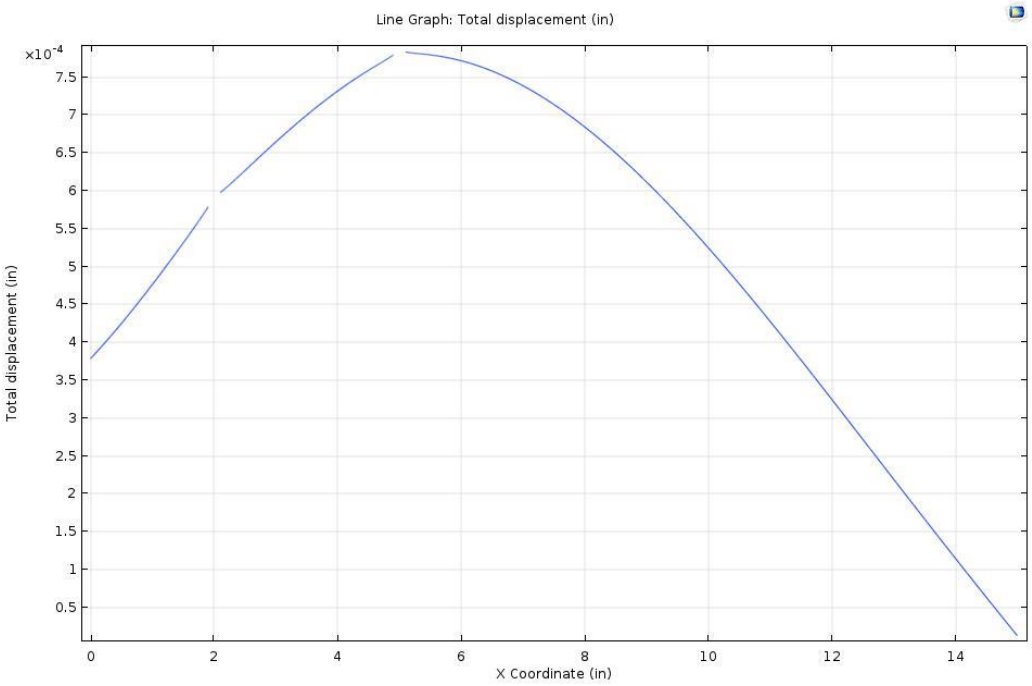

Figure 12. Bottom plate displacement along centerline

Lastly, one of the biggest concerns was keeping heat away from the torque sensor as it could not handle temperatures beyond 93℃ and even at that temperature the supplier advised that the accuracy would be impacted. Initially a section of machinable ceramic was going to be added in

to separate the bearing cup from the shaft that connects to the torque transducer. After design changes had been made the final decision was to use a shaft made of Polyether Ether Ketone or PEEK as this could be 3D printed for less than the cost to machine a metal and ceramic shaft and could handle the high temperatures. A steady-state simulation was performed in SolidWorks as COMSOL had difficulty handling the geometry of the original CAD files. The material properties used for the simulation are shown in Table 2.

Table 2: Material properties used for each component in thermal simulation

Component:	Thrust Bearing Shaft
Material:	Polyether Ether Ketone
Specific Heat:	$1850\ {}^{J}/_{kg*K}$
Thermal Conductivity:	$0.24\ {}^{W}/_{m*k}$
Density:	$1310\ {}^{kg}/_{m^3}$
Component:	Bearing Cup
Material:	6061-T6 Aluminum
Specific Heat:	$896\ {}^{J}/_{kg*K}$
Thermal Conductivity:	$166.9\ {}^{W}/_{m*k}$
Density:	$2700\ {}^{kg}/_{m^3}$
Component:	Bearing
Material:	Alloy Steel
Specific Heat:	$460\ {}^{J}/_{kg*K}$
Thermal Conductivity:	$50\ {}^{W}/_{m*k}$
Density:	$7700\ {}^{kg}/_{m^3}$
Mesh Quality:	High
Convection Coefficient:	$4.2\ {}^{W}/_{m^2*K}$
Ambient Temperature:	298.15 K

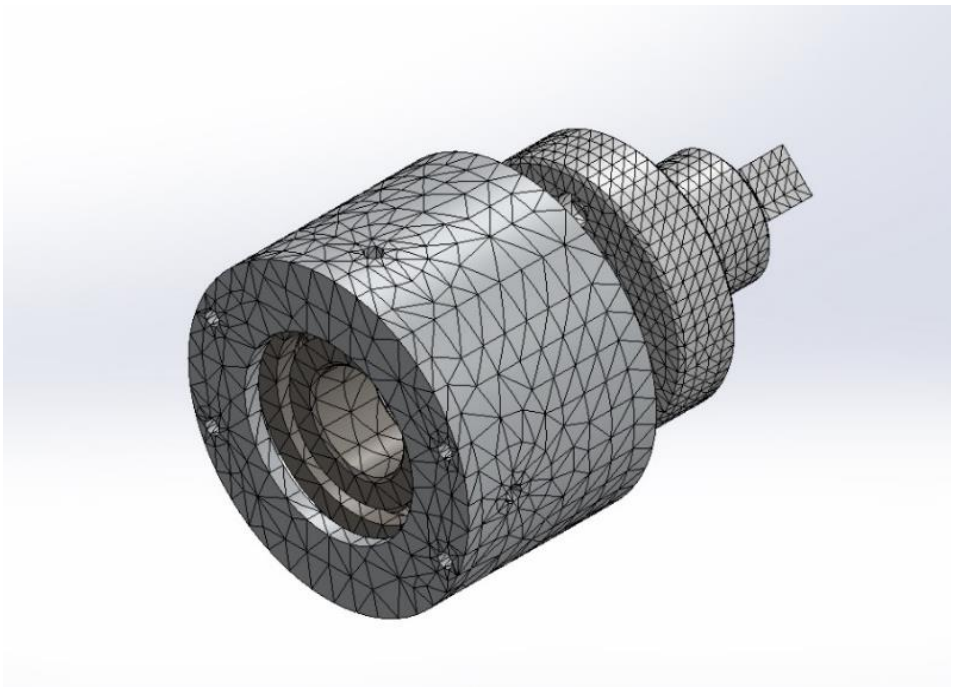

Figure 13. Assembly in SolidWorks showing mesh used

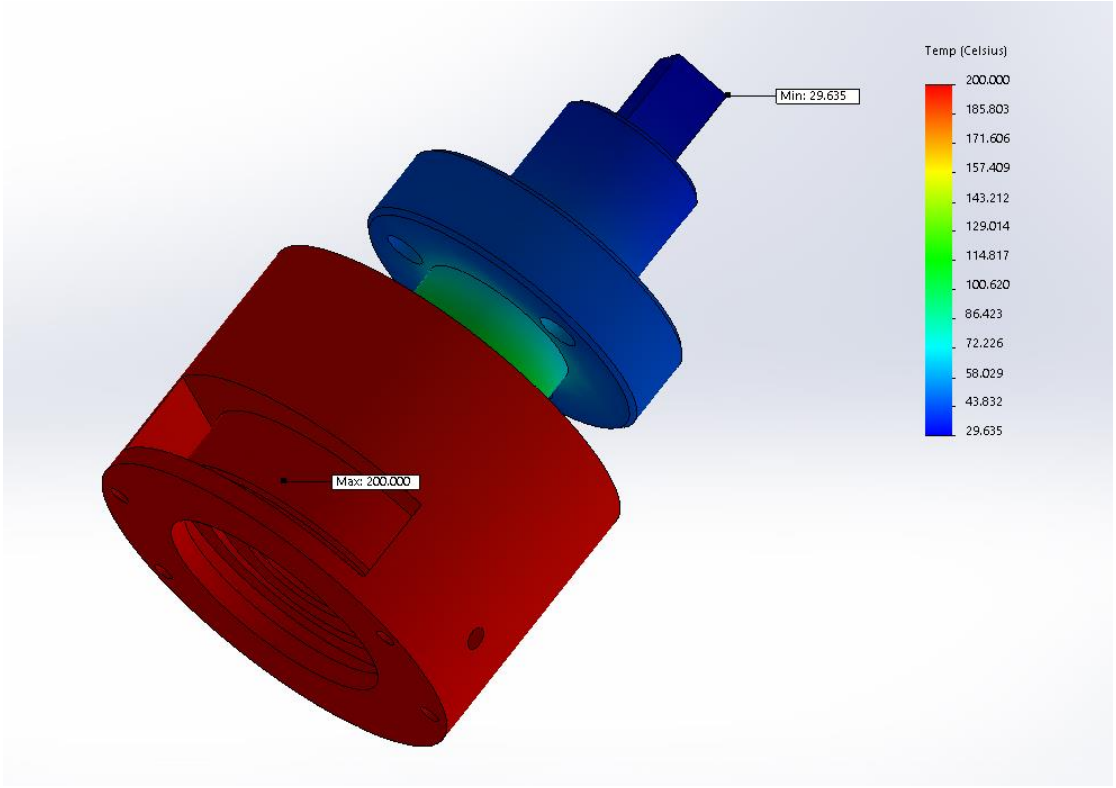

Figure 14. Temperature Distribution of assembly

The simulation was done on the complete assembly with a temperature being applied to the bottom surfaces and bearing. Convection was added around the rest of the surfaces to simulate the cooling effect from the air around it. From *Figure 14* the bearing cup perfectly distributes the heat evenly as was desired to completely heat the bearing. The PEEK shaft on the other hand does a wonderfully bad job at transferring heat with the temperature reduced to nearly ambient by the time it reaches the flange. At the end connecting to the torque senor the temperature was only 5 degrees higher than what the ambient was set at for the simulation which was more than safe for the sensor being utilized.

2.9 PLANS TO ASSEMBLE

An important consideration when designing something is to make sure that the assembly is easy. To emphasize this, the entire build uses a minimum variety of standard bolts and nuts. The entire assembly can be broken into four smaller sub-assemblies and then ultimately assembled to form the whole functioning test apparatus. The first subassembly seen in *Figure* 15, consists of the lower aluminum structural plate, four 8020 extrusion, the pneumatic piston, the piston adapter, bearing cup adapter, load cell, lower cut and upper cup for the bearing shaft. It is important to note the two bearings that must be pressed into the bearing cup prior to assembling onto the load cell. All components will be fastened together with the appropriate nuts referenced in the design schematics of each component. Once this sub-assembly is complete, it can be placed aside, and the middle sub-assembly can begin.

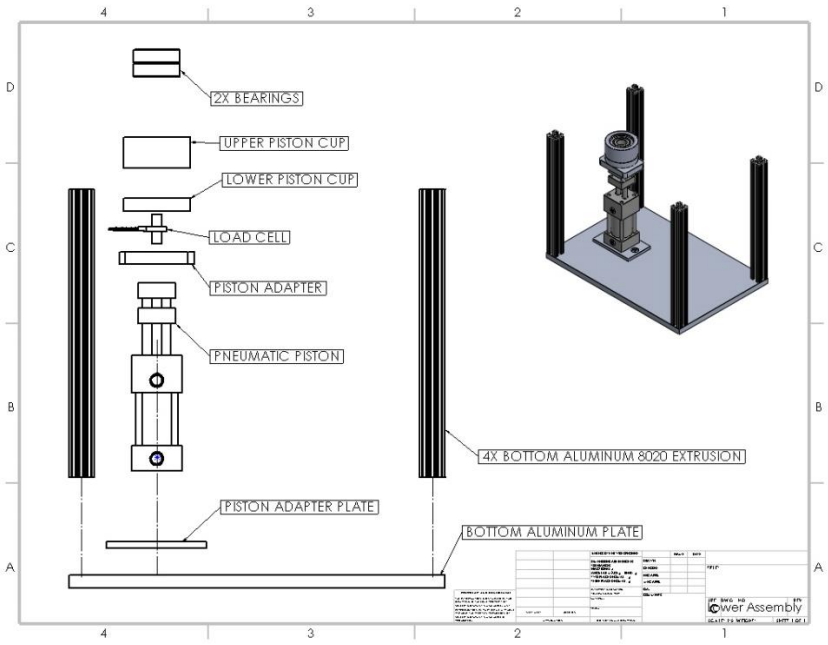

Figure 15. Lower sub-assembly.

The middle assembly can be seen in *Figure 16.* It consists of the middle aluminum structural plate, four smaller lengths of 8020 extrusion, the sliding bearing casing, and the eight leg brackets for the 8020 extrusion on the lower assembly. The sliding bearing is held in place by two snap rings on both sides of the sliding bearings and the casing held in by the nuts referenced on the drawings. The middle sub-assembly can be placed aside and then the upper assembly can pursue.

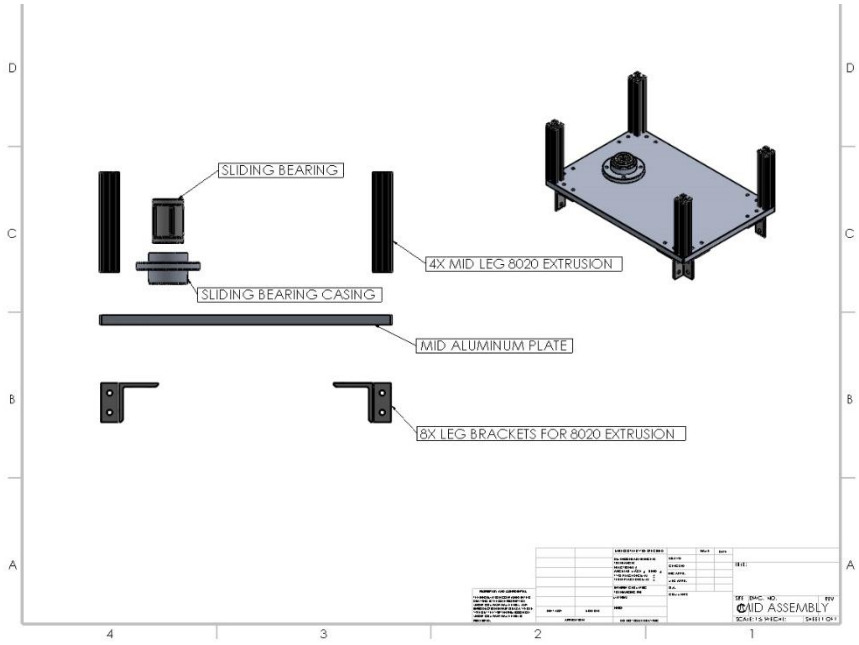

Figure 16. Middle sub-assembly.

The third assembly is the upper assembly and can be seen in *Figure 17*. This assembly consists of the upper aluminum structural plate, the piston shaft, the servo motor, the 4 middle lengths of 8020 extrusion, the second sliding bearings along with the sliding bearing casing and the nozzle to supply hot and cold air to the bearings and hear and cool the lubricant inside. The pully system along with the tensioner must be installed with the belt at this point. One all components are assembled; the shaft will be free to translate about the sliding bearings and should be fastened down in the meantime prior to assembling the three sub-assemblies.

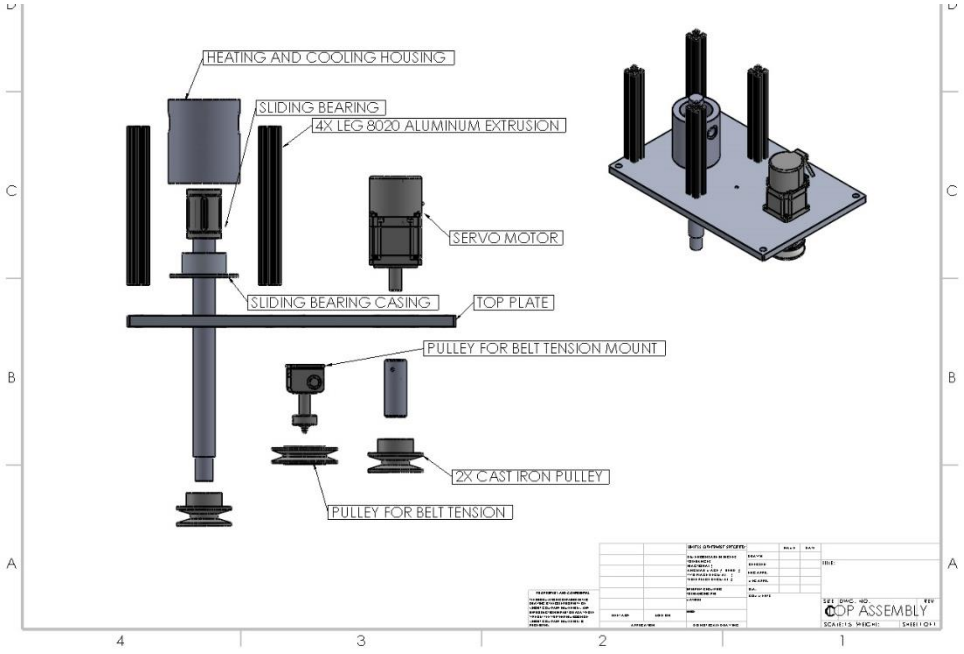

Figure 17. Top sub-assembly.

Before proceeding with the fourth assembly, the three sub-assemblies, i.e. the Bottom, Middle and Top sub-assemblies can be assembled to form the body of the apparatus. This can be seen in *Figure 18* below.

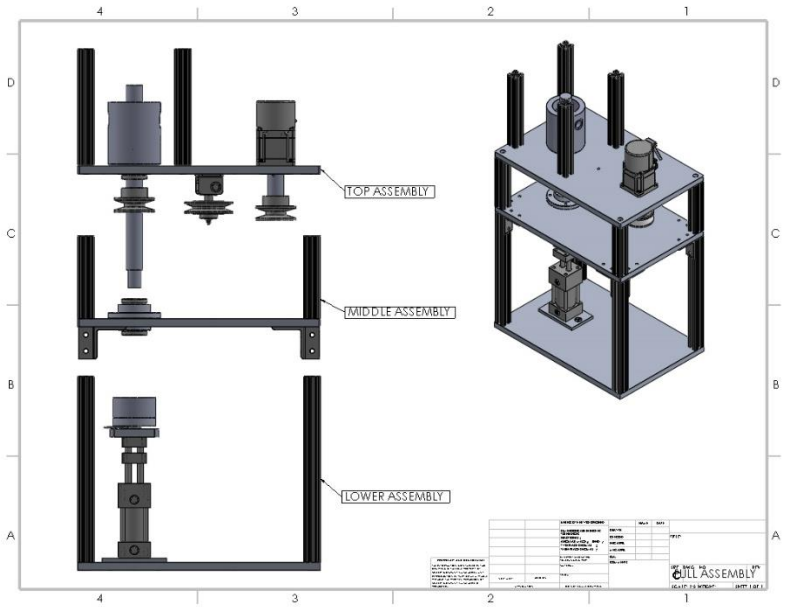

Figure 18. Assembly of Middle, Top and Bottom sub-assemblies.

Finally, the accessory assembly can be put together and be seen in *Figure 19* below. This consists of the torque transducer, the transducer casing, smaller top aluminum plate, thrust bearing, peek shaft, and the bearing cup holder. These components will be fastened together using the recommended bolts or nuts recommended in the schematics of each part.

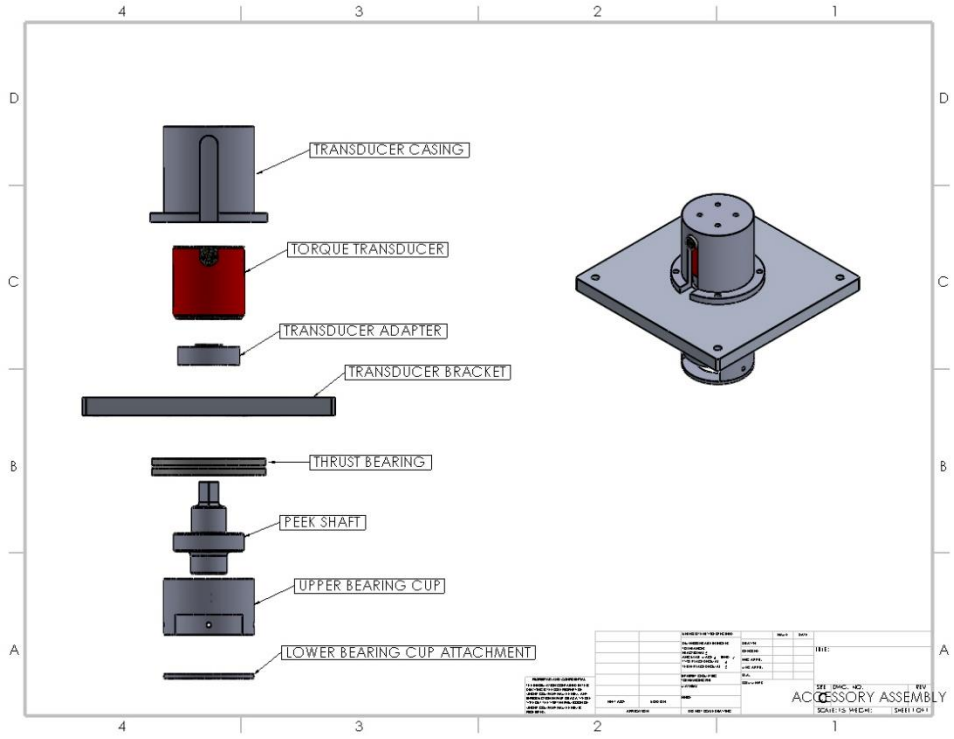

Figure 19. Accessory sub assembly.

The top accessory assembly can be connected to the body of the device and the full build be together. It's at this point where checking loose connections, tightening nuts and bolts can be performed to make sure nothing comes loose during the operation. A figure of the accessory assembly connecting to the body can be seen below in *Figure 19* noting that the square faces of the end of the peek shaft must align with the transducer in the correct orientation.

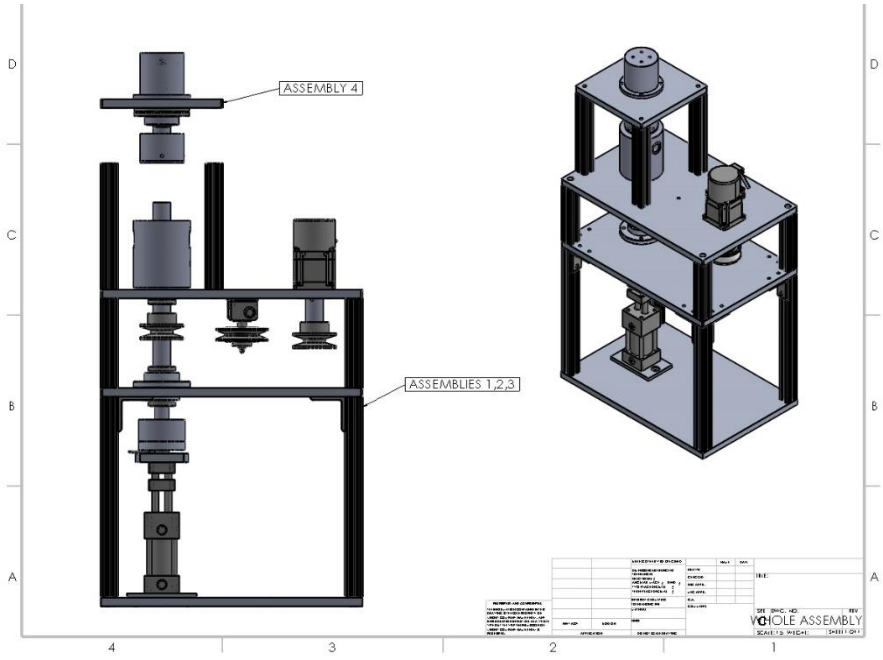

Figure 20. Assembly of accessory assembly to body.

2.10 OPERATION

After two semesters of extremely hard work on this project, the team has a sublime understanding of how the machine would ideally operate. Reference *Figure 2* in Section 2.5 for components. To begin the test, the operator will load the *Nye*-produced lubricant into the test bearing and slide the bearing into the opening of the bearing cup (7). With the v-belt properly secured into each pulley (6), the pneumatic piston (5) will extend and push the cone-shaped design at the top of the test bearing shaft (13) securely into the pocket of the bearing cup. Since the bearing cup has a lip for the outer race to sit on, this acts to shear the balls between the inner and outer race. Once the piston applies the desired load, the desired temperature will be delivered through the nozzle (10) by either the heat torch for hot, or the vortex tube for cold. Once the load cell (12) reads the correct load delivered from the piston, and the IR thermocouples read the correct temperature, the servo motor (4) will spin the motor shaft (13) at the desired RPM. This will in turn spin the test bearing shaft at the same RPM. In this whole process, the inner race of the test bearing will attempt to spin the outer race. However, since the outer race is secured by the bearing cup, and the bearing cup is secured by the torque transducer via the thrust bearing shaft, the transducer will measure the torque value

between the inner race and outer race. Because the torque transducer cannot handle much axial load, the team implemented a thrust bearing (9) to absorb the axial load from the piston. The outer race of the thrust bearing simply bolts to the top bracket, while the thrust bearing shaft threads into the inner race of the thrust bearing. This acts to take the load off the transducer. In *Figure 37,* in *Appendix A*, one may notice that the thrust bearing shaft has a square extrusion at one end. This extrusion fits into a square female aluminum adapter, which connects to the torque transducer. This can be better seen in the exploded view in *Figure 21* below. The twisting motion from the test bearing will be secured by this square piece, allowing the transducer to observe the torque value. While the test is running, the thermocouple will monitor the test bearing for any temperature changes and plot it on a graph in the GUI. The load cell will monitor the load delivered by the piston and will output it to an indicator on the GUI. Most importantly, the torque transducer will read the torque values, while the test is running, output them onto a graph on the GUI, and will write the data to a file, which can be exported to Excel. Once the test runs the desired time length, the motor will stop spinning, the heat torch or vortex tube will shut off, and the piston will detract into its initial position. To remove the test bearing, simply insert a pen into a hole on the opposite side of the cup opening and push.

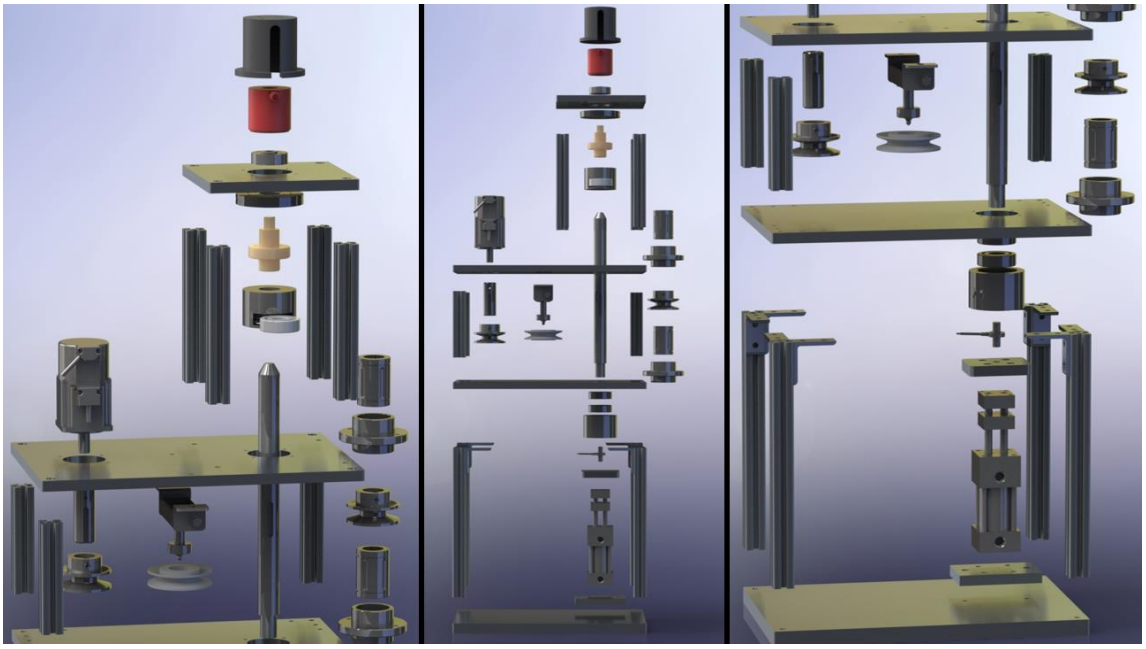

Figure 21. Exploded View

3. ELECTRICAL DESIGN

3.1 SYSTEM ARCHITECTURE

Shown below is *Figure 22*, is the overall high-level electrical system design. Shown in the gray box at the top is the central computer running the Windows 10 OS, that houses all the programs and devices. Running on this computer is the main system UI that was designed in Windows Forms, as well as the Arduino IDE. The UI, that was developed in Windows Forms with the C# object-oriented programming language, acts as they main interface between the user and the system. This is where all the controls for the system, as well as the outputted data is shown. The UI is described more in depth in section 3.3 Graphical User Interface.

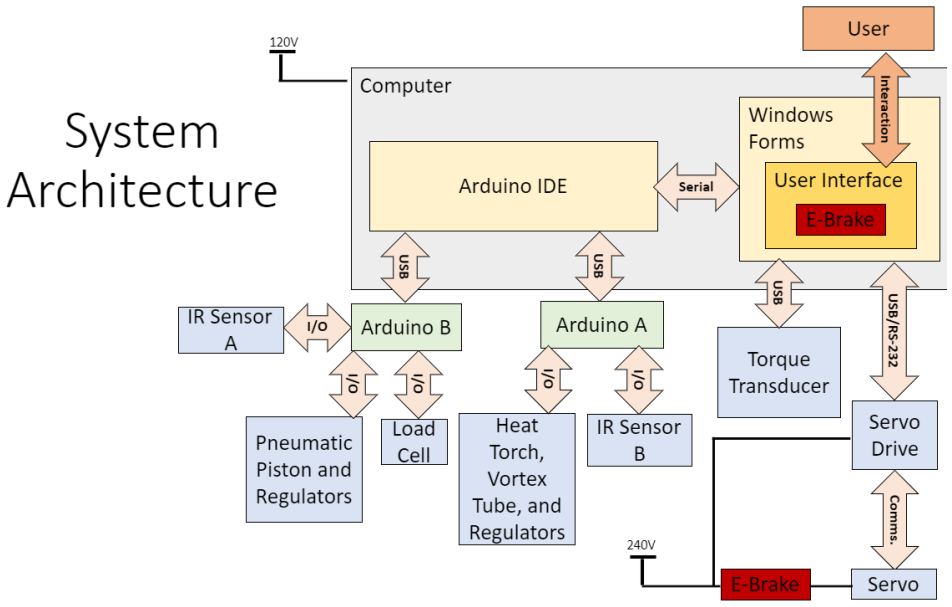

Figure 22: Electrical Systems Diagram

The blue boxes represent all of the sensors/actuators that are connected into the overall system. Some of the larger/main components connect directly to the UI while the smaller components connect through I/O ports into the Arduino. Specifically, the servo and SureServo drive connect to the system through RS-232 and communicate using the ModBus communication protocol. The UI handles this communication without the help of a 3rd party system, saving overall execution time and bypassing the reformatting of the servo drive. The torque transducer also comes in through USB and is programmed using software made available

by the supplier, Futek. The information is stored in a .txt file and is then read by the UI. This allows the data to be stored locally for later viewing and also lets the system access it once every instruction cycle.

The Arduino communicates to Windows Forms over serial communication. This is gone into depth in section 3.4 Communication Framework. On the IDE, a basic program is running that handles all local information from the sensors/actuators, as well as sending the information to the UI when needed. There are 2 separate Arduino boards connected, as 1 board was not enough to house all of the I/O and I2C ports needed. Both IR temperature sensors, as well as the heating and cooling apparatus, are connected through the Arduino. This way, the Arduino code handles the temperature regulation. Lastly, the pneumatic piston and regulators are also connected though I/O ports.

Additionally, both e-brakes are shown in the way that they are connected. The electrical e-brake is available during runtime on the UI. This button is mapped to the end of the program and halts all operation, resetting the system. In the case of the UI freezing or an immediate danger, a physical e-brake is also made available on the motor. This brake cuts power to the motor, while leaving the drive powered. This way, the motor will cease to spin, and the drive will receive an error code, waiting for the system to be reset to normal operation. This provides an added layer of safety to the overall system.

3.2 SOFTWARE DESIGN

As stated in Section 3.1 System Architecture, this system of systems centralized hub is located in a Window's Forms application. Here, all of the main data processing, controlling, and user interaction is handled. Windows forms in object-oriented C# was chosen as the team has experience designing complex UIs in this language. Along with the main C# program, there are also components coming in from the external Arduino IDE interface. More information on how these programs interact and the way they communicate can be found in section 3.4 Communication Framework.

Running on Visual Studio is a graphical user interface, created by the team. More on how this GUI is designed and laid out can be viewed in section 3.3 Graphical User Interface. Here, all user interaction is centralized, given various controls to the engineer using the machine. Shown below is a basic software flow

diagram for how this end-level program should function. Indicated by the three green boxes are the separate execution phases. The first is a grease work-in period, as the grease must be slowly distributed within the bearing prior to the test startup. The second is a temperature regulation setup, which ensures that the heat torch and vortex tube are holding the grease in the bearing at a constant set temperature, using the IR temperature sensor. Lastly is the actual Lubrication test itself, where the bearing begins to spin, and the readings are taken. In a normal program execution, all three of these processes would be run one after another in series. The reason they are split up into separate tasks is to allow the tester to rerun the grease work in/temperature setup as needed or omit them entirely if desired.

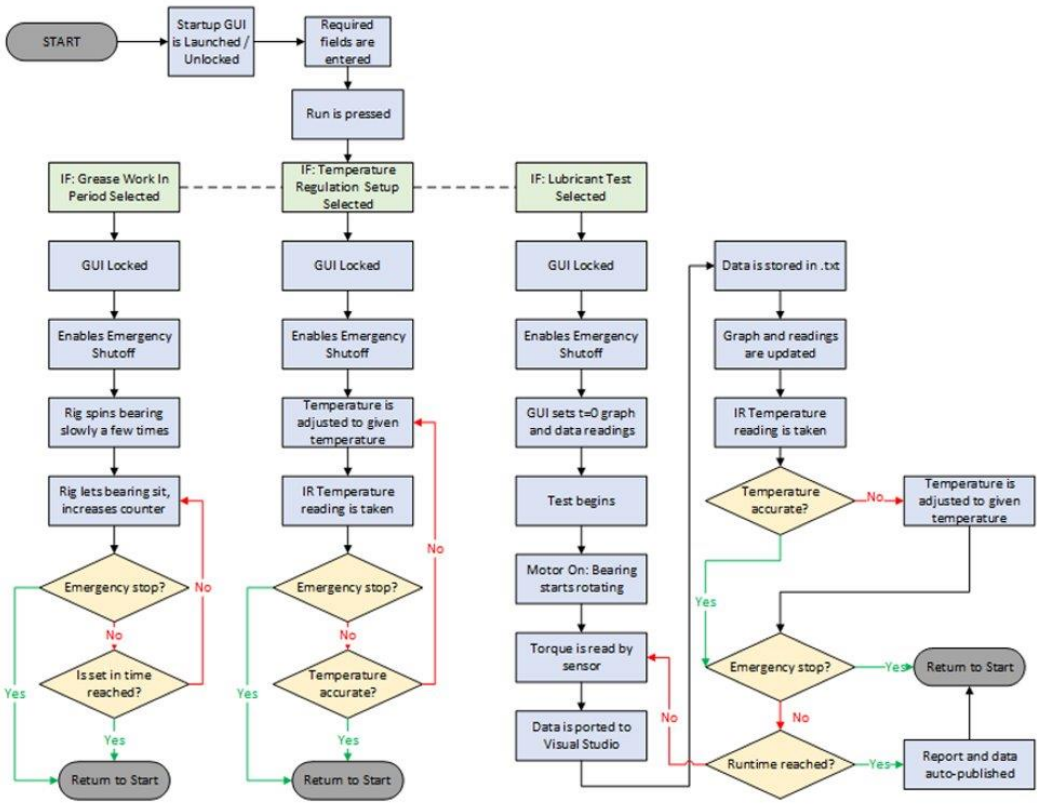

Figure 23: Software Execution Flow Diagram

The grease work in phase consists of the motor spinning slowly for a few rotations, giving the grease enough time to coat the entire inside of the bearing evenly. This is important, as it gives the all tests a consistent startup. This step, as for all steps, will enable both emergency brakes on startup, to give the user the highest

possible safety while running the tests. After the grease work in period has ended, the user has the option to rerun the test again or move onto the next phase.

The second phase is the temperature regulation setup. For this, the system will consider the desired run temperature set into the UI by the user. Once this value is stored, the system will begin to heat/cool the grease in the bearing as needed to get the temperature closer to its desired state. Using the IR temperature sensor mounted at a safe distance, the system can internally configure the two components to regulate until the temperature has reached an acceptable level. Every execution cycle, the Arduino mounted IR sensors will plot the current temp sensor onto the UI, so that the engineer on site can make sure the system is working as desired. At this point the UI will alert the user that the temperature is stable, and testing can begin.

Lastly is the main lubrication test. Once the test is ready to begin, the user will press the RUN button, enabling the system to startup. Like the other phases, the UI will lock, and the emergency brakes will enable to be made available to the user. The temperature and Torque graphs will both initialize, and readings will begin collection. Additionally, the pneumatic piston will set to the desired weight before the bearing begins to spin. Once the motor begins turning, it will slowly ramp up until it reaches the desired speed, stored in the UI by the user. During the duration of the test, the temperature regulation system will continue to work to make sure the temperature stays at the desired level. Once the set runtime is reached or the brakes have been pressed, the system will cut power to the motor causing it to slow down. While this is occurring, the temperature system will cease and allow the system to coast back to room temp. Lastly, the data collected and graphs will be placed into an auto-generated test report, along with the user inputted parameters, to give a consistent and user friendly overview of the conducted test. Once the system returns to its idle state, a new test is ready to begin.

3.3 GRAPHICAL USER INTERFACE

The user interface consists of three separate tabs, each containing their own graph and title corresponding to what is being monitored in real time (torque, temperature, or RPM). At the top is three textboxes for the

adjustable parameters depending on the test being done. There are three buttons to break apart the stages of a test; one to slowly work the grease into the bearing, one to stabilize the temperature to the given value, and one to perform the actual test for the duration of the entered runtime. A safety feature was added through an emergency stop button, which will clear out the graph and any parameter values, setting up for another test. The current torque values would be displayed to give the user an estimation at that moment of the running test, while still being able to monitor the system through the multiple graphs. A feature that was not added due to time constraints on the project was a way of collecting all incoming information to present to the user once a test is completed. This raw data would be helpful if trying to gather or interpret certain information from multiple tests and could be obtained simply by adding a few lines of code right after reading from an incoming port instructing a quick write to a .csv text file for logged storage.

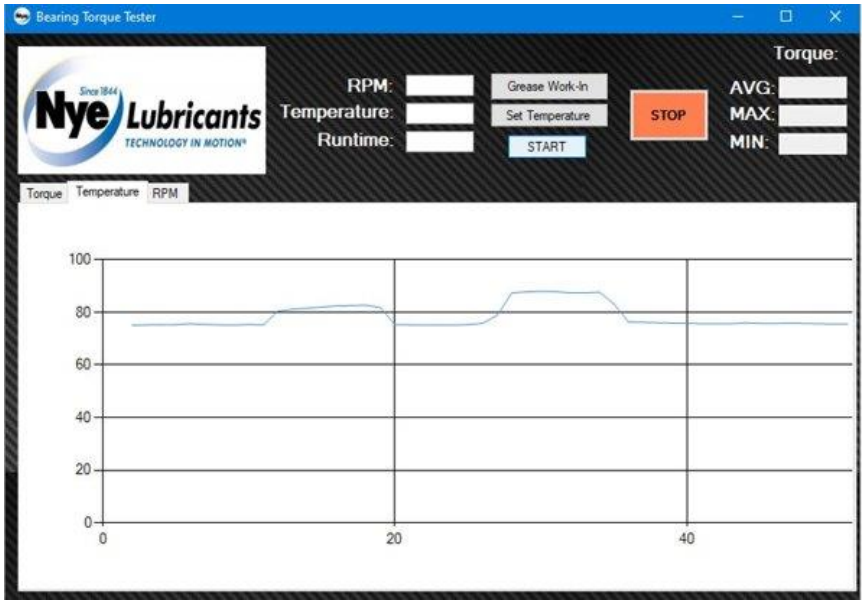

Figure 24: User Interface

3.4 COMMUNICATION FRAMEWORK

Given the complexity of the proposed final design, it was essential to come up with and then implement a hierarchal framework for how all the actuator/sensor subsystems would communicate with the user interface. As expressed in the system architecture review, the full design consists of multiple sensors and

actuators, connected to the main UI through two separate main channels: The USB interface and the Arduino interface. For simplicity, the team will discuss each section independently.

The main UI was designed and built in Visual Studio's Windows Forms. Since this central code served as the main hub for all the data and processing, all information had to come and go from this location. Larger and more complex components in the system connect directly to the PC from either USB or RS-232. These components include things such as the torque transducer, servo motor, and servo drive. A lot of these components come with their own software that make ease of use much better for certain applications. Since all data is centralized in Visual Studio, all these external sensors/actuators needed to be programmed with the team's 3rd party software in order to cut down on operating times. Particularly for the servo drive, the communication framework that is used is ModBus communication. For this certain application, the main system sends out messages to the drive in ModBus to change the speed of the motor as well as turn it on or off. This saves an exponential amount of time since the drive doesn't have to reformat like with the servo's original program. The registers can just be changed independently in a fraction of a second. Data coming into the system, like from the torque transducer, can be placed into a local .txt file and then read from using the program in each program execution cycle. This allows the information to be stored locally as well as accessed in real time.

The second main channel is through the Arduino IDE. All smaller I/O based components come into the system through an Arduino UNO board. The board is controlled by a small amount of code in the Arduino IDE. The IDE's main job is to collect data on the sensors and send data to the actuators. Since it acts as a pathway from the Arduino board to the Visual Studio main UI. There is a serial port between the two digitally, working as a pipeline, that allows for this data to be sent. For example, the IR temperature sensor collects data in real time and uses that to update the current temp variable inside the Arduino IDE. On each instruction cycle, the UI sends a "READY" instruction to the IDE asking for the current value, of which the IDE then sends back. This allows for the most current value to always be available each cycle, without having a backup in the pipeline. The Arduino side takes care of components such as the pneumatic piston,

actuators, IR sensors, heat torch, and vortex tube. Because there aren't enough pins on a single Arduino UNO to handle all these components, a second board can be used to handle the extra ones. The first could then be used as a master device, communicating to both the Arduino IDE and the second board, while the other board sends its data through a serial port to the master Arduino. Alternatively, both Arduinos can communicate directly with the IDE. In both instances, the system will be responsive, and all necessary pathways will be open.

3.5 FUNCTIONAL PROTOTYPES

Before starting with the actual components, our team wanted to test communications between Visual Studio and Arduino IDE to assure the plan would work. A simple write/read program was created in the two environments and were able to receive a fast-enough connection for a real time viewing display. This was important for the user interface aspect as we wanted to give an appealing graph to constantly understand what is currently happening within the system.

For prototyping the infrared sensor and laser diode, our group knew that this was going to be built through an Arduino board and began looking for an applicable sensor that would be able to function properly and accurately, while still falling in the requirements of an Arduino component (low voltage, small in size, etc.). After settling on an appropriate sensor, our group had to write a short segment of code that is rapidly reading the temperature from the IR sensor and writing this information out to a port in order to prepare for integration with the full system.

When the servo arrived, a drive had come with it to be the mediator between the servo itself and the user. This servo system requires a 240V input, which was only achievable at *Nye* because our team does not have access to the higher voltage outlets on campus. The first prototype was solely done with only the servo and its drive. Through this setup, time was spent learning the functionalities of the necessary registers to get the servo moving. After feeling confident using the drive, the brands SureServo Pro software was installed, which gives the user access to every register at once rather than scrolling through on the drives display. These two methods were a good start but required a hard reset after each test, losing any ease of use.

3.6 FINAL SYSTEMS INTEGRATION

To integrate the Arduino usage into the GUI, the same procedure done in an earlier prototype of communications was used between the Arduino IDE and Visual Studio. This approach utilizes a COM serial port connection through a USB interface to always be reading on the GUI's end. Meanwhile, on the same computer, the Arduino code is active and constantly writing the read temperature to the same port. To prevent any lockup within the main flow of code, this interaction is entirely handled outside by using a threaded process. This threaded process, when evoked, will run in tangent with any other currently active code; a method that would be used on other components utilizing an Arduino board.

To avoid using the given servo software or the actual drive controls, the communications were diverted through a Modbus RTU protocol implemented into the GUI, with an RS-232 cable connecting the main computer and the drive. This protocol sets up a master-slave system configuration allowing the GUI to send out requests and receive responses. Giving the user this control simplifies the process by hiding the messages going back and forth in the makeup of the interface, which removes any wait time for running a test. As a safety measure, a physical brake was added between the drive and the power source. This way the servo and drive have no way of performing any unwanted actions when the brake is triggered.

3.7 ECE WORK CONTINUATION

Due to the national health crisis, this project's full scope wasn't able to be fully executed during the senior capstone course. Because of this, For *Nye* to finish this system, there are still a few components to implement onto the electrical side. First, we will discuss the work that has been completed on the prototyping/implementation side of things. Primarily a lot of the underlying architecture was developed and designed. The communication framework from Arduino to Visual Studio was implemented using serial ports and used to fully implement the IR sensor for temperature feedback. Additionally, the servo motor and drive are completely implemented, bypassing the SureServo software and communicating directly to Visual Studio. A lot of work was also put into the user interface, as it was designed and tested. Quality of life security implementations have been added to help users such as error messages and minor bug fixes.

Looking at the big picture, currently the system has a fully designed interface that has full motor and temperature sensor implementation. The runtime timers are also fully implemented, and the real time graphs are implemented for temperature and available for torque. Lastly both e-brakes are implemented and have been tested accordingly.

The work that the team was able to complete was limited to the ECE labs at school, as well as *Nye*, closing their doors temporarily for the rest of the semester. All of the components that had been received up to that point have been completed, however some weren't received yet. The Torque transducer for one still needs to be prototyped. Once the information is in Visual Studio however, the data can be graphed similarly to the real time temperature graph. Additionally, the heating and cooling components need to be implemented, interacting with the implemented temperature reading system to fully finish the regulation setup phase of operation. Lastly, the pneumatic piston subsystem and small load cell need to be implemented through the Arduino to allow for an applied load and check on that load. As far as programming goes, the last thing that would need to be done is the grease work in phase, just making sure the motor jogs for a set amount of time. Once that is complete the system should be operational, given that thorough reliability testing is conducted. The team was really looking forward to finishing this project, but this was out of our hands.

CONCLUSION

After conclusion of the project, Team Lubricant is proud of the hard work accomplished. After months of brainstorming, hypothesis testing, designing, and experimenting, the team was able to narrow to a final, functional design. Every detail of the rig was thoroughly designed, tested, and redesigned as needed. Although the team was not able to physically manufacture the rig, due to the COVID-19 pandemic, Team Lubricant has confidence that the machine would perform almost exactly as expected, accounting for minor adjustments along the way.

Like every project, the "Lubricant Effectiveness Characterizer" had its fair share of challenges. The most prominent of these challenges was the COVID-19 Pandemic. Unfortunately, when UMass Dartmouth closed for the semester in early March, so did all *Nye* facilities. Nearly all physical work done on the project was done inside *Nye*. When the facility closed, the team had no way of accessing the physical components, forcing *Nye* to suspend all physical work of the project. This included manufacturing of the rig, as the 3rd party machine shop had also closed its doors. The other pressing issue the team faced throughout the duration of the project was budgetary concerns and restrictions. *Nye* had initially awarded the team $30,000 of initial budget, however, because of the hefty cost of internal *Nye* projects, the Lubricant Effectiveness Characterizer required budget cuts. Ultimately, this triggered several time-costly redesigns, including the use of a thrust bearing instead of an air bearing. The team was also not able to buy the heat torch required to heat the test bearing to 200°C, because of cost. With that restriction, the test bearing would only be able to heat and cool under the capabilities of the vortex tube. Lastly, because of the complex nature of the project, the team became extremely creative when designing components. Some components needed to withstand extreme temperature; some would be placed under substantial load. This made it difficult to use conventional materials and methods, consequentially challenging the team's true ability to adapt situations.

Ultimately, Team Lubricant was able to complete the mechanical and electrical design of the rig with $5,955 and could have a fully manufactured rig with $10,655. With an awarded budget of $30,000, the team came in about $20,000 under budget, assuming manufacturing would be completed. The team was able to save

substantial money and time through continuous design simplifications, material selection, and smart purchasing.

In two short semesters, the team was able to mechanically and electrically design a machine capable of testing the effectiveness of a lubricant inside a bearing spinning at high speed, under an axial load, and under extreme temperature conditions. It functions to essentially create a lubricant failure in a controlled environment, rather than in a potentially dire situation, such as in an airplane engine. For this reason, the Lubricant Effectiveness Characterizer will provide *Nye Lubricants* with a selling point over many other lubricant companies. It will provide their customers with peace of mind that a *Nye*-produced lubricant will perform exactly as expected, even under the harshest of conditions. This machine will uphold the world-renowned *Nye* reputation and will further solidify them as a world leader in the lubrication industry.

FINAL PLACEMENT AND ACKNOWLEDGMENTS

After a full year of hard work, Team Lubricant is honored to announce that the "Lubricant Effectiveness Characterizer" has placed 1st place in this year's final capstone event. Every team member worked around the clock to ensure that this project was victorious. The team would like to thank the *University of Massachusetts Dartmouth,* including Dr. Hamed Samandari, Team Lubricant's academic advisors, and all who helped make this project a success. Team Lubricant would also like to thank the sponsor, *Nye Lubricants Inc.*, with special regard to Richard Raithel, Mason Wood, and Jason Galary, for making this project possible and for providing undivided attention to help the team prosper.

APPENDICIES

APPENDIX A – MECHANICAL DRAWINGS

1. Bearing Cup Disc

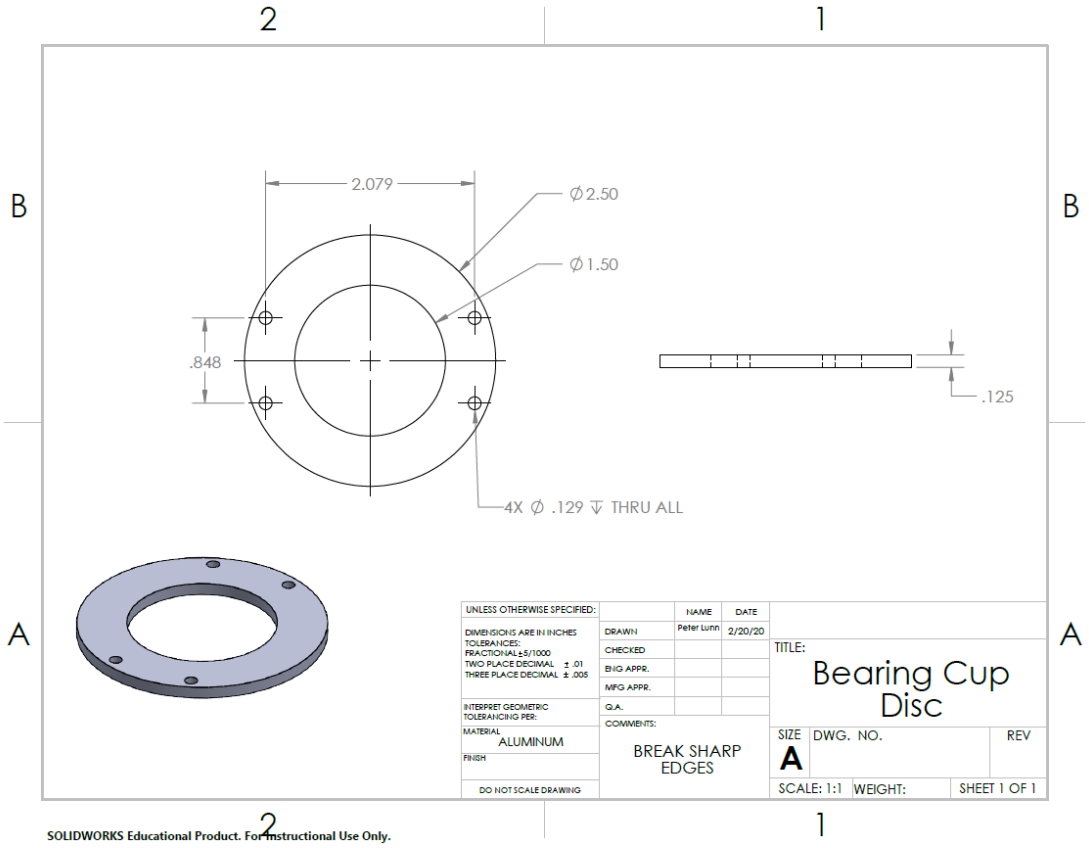

Figure 22. Bearing Cup Disc Drawing

Continue to Next Page to View Other Drawings

2. Bearing Cup Drawing

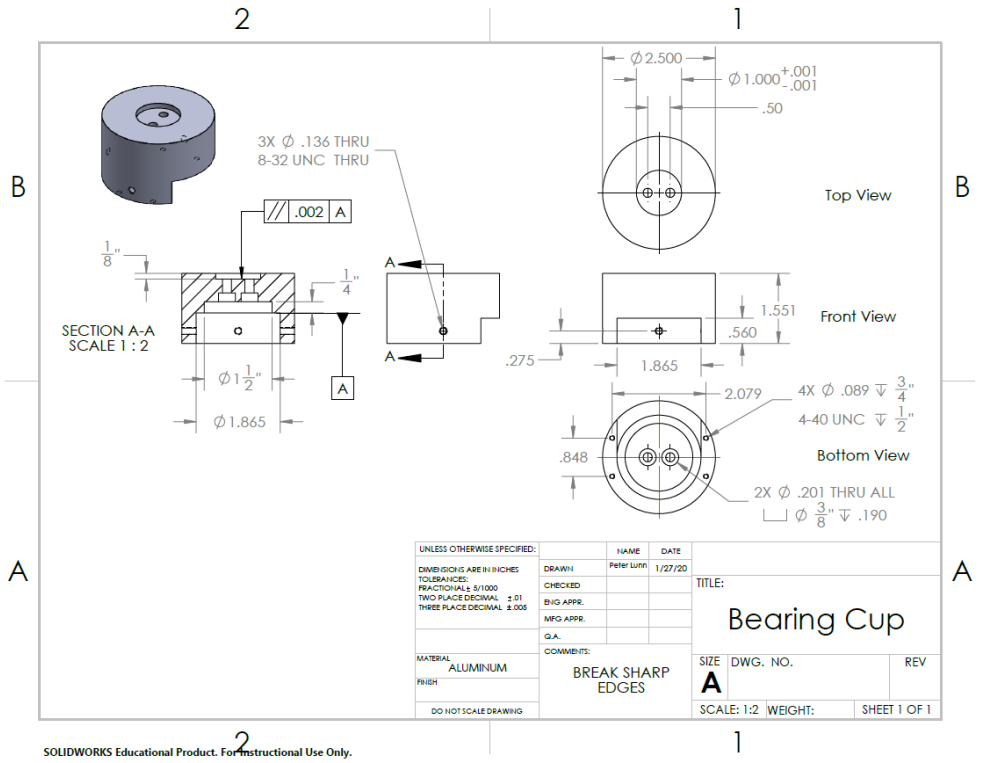

Figure 23. Bearing Cup Drawing

3. Bottom Legs

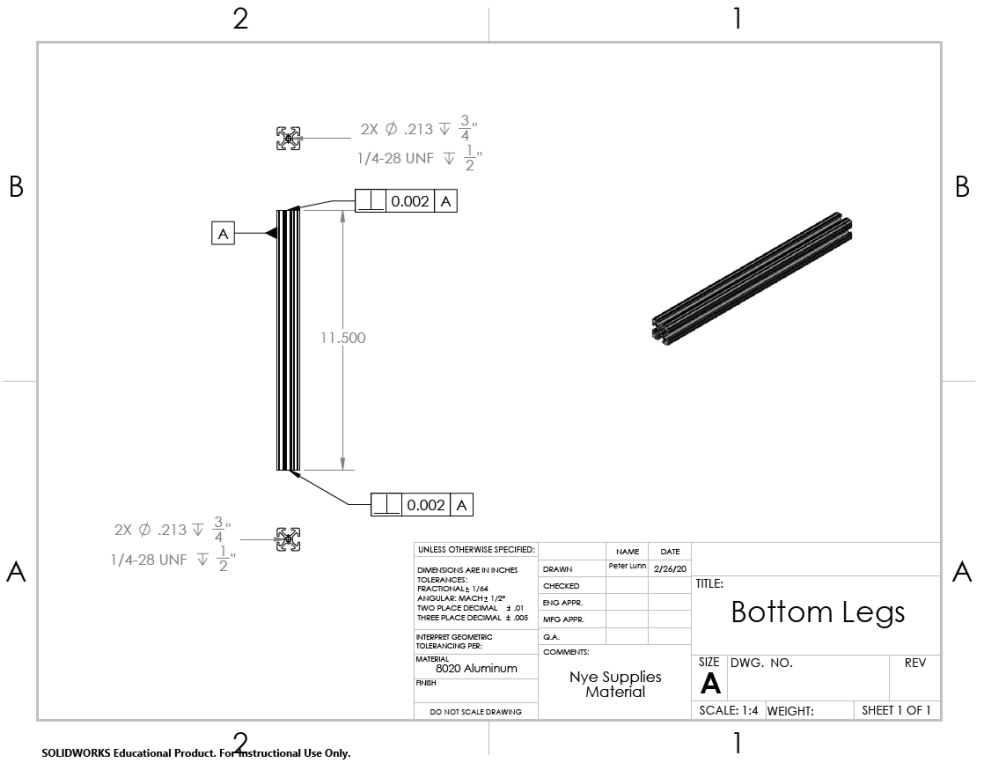

Figure 24. Bottom Legs Drawing

4. Bottom Plate

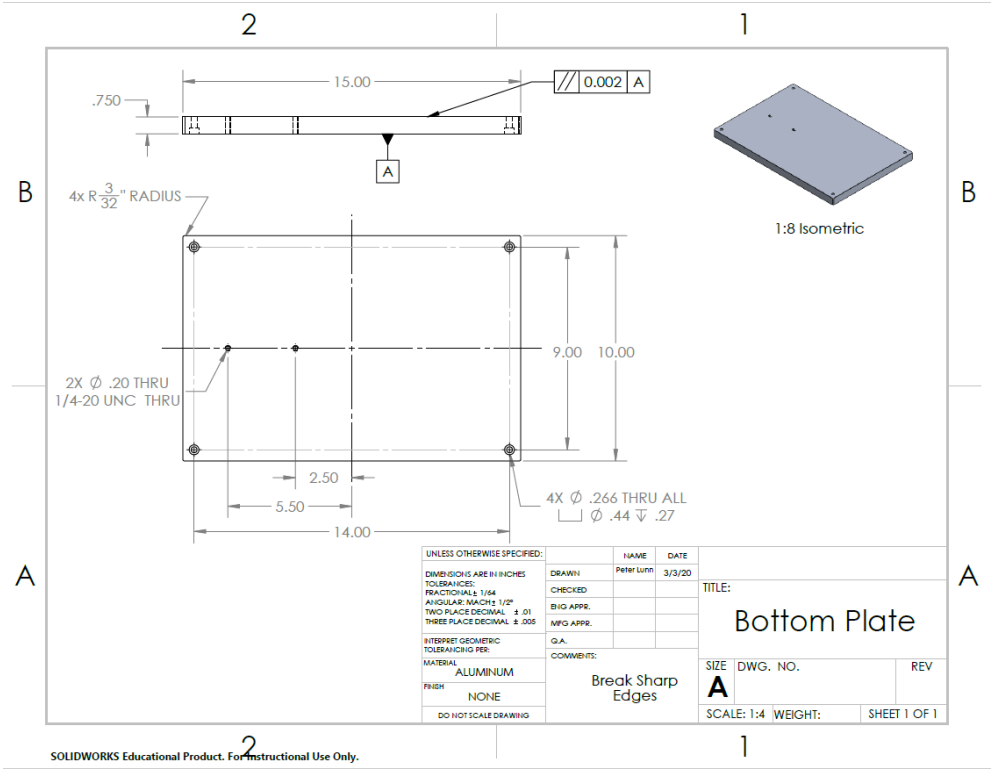

Figure 25. Bottom Plate Drawing

5. Frame Bushing Ring

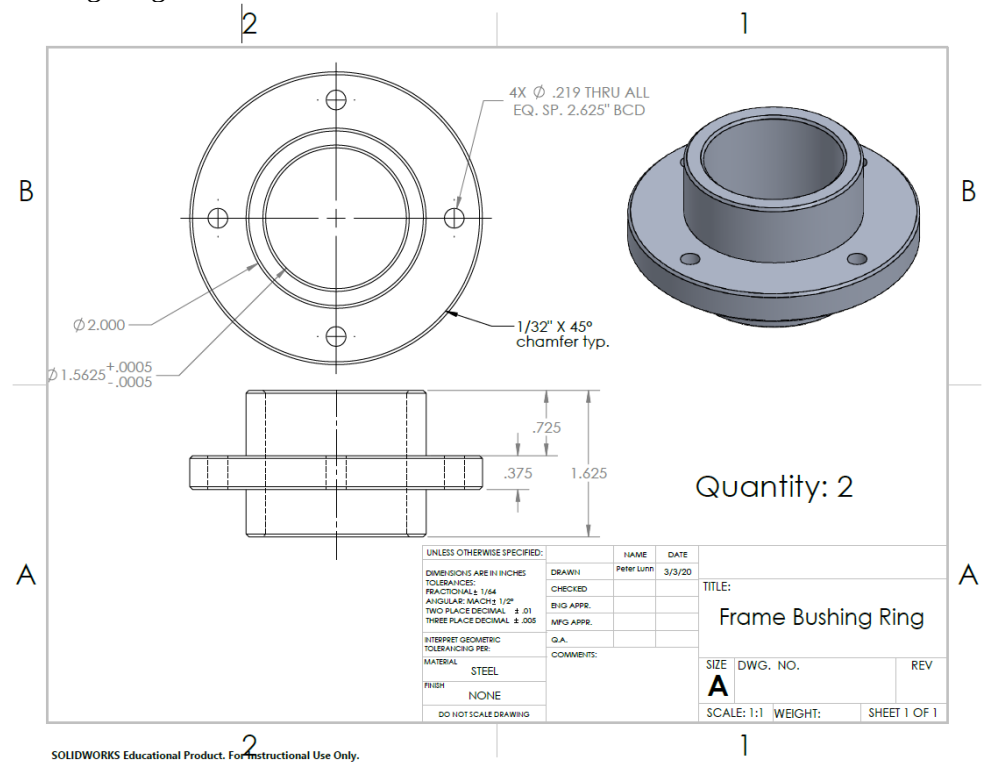

Figure 26. Frame Bushing Ring Drawing

6. Main Plate

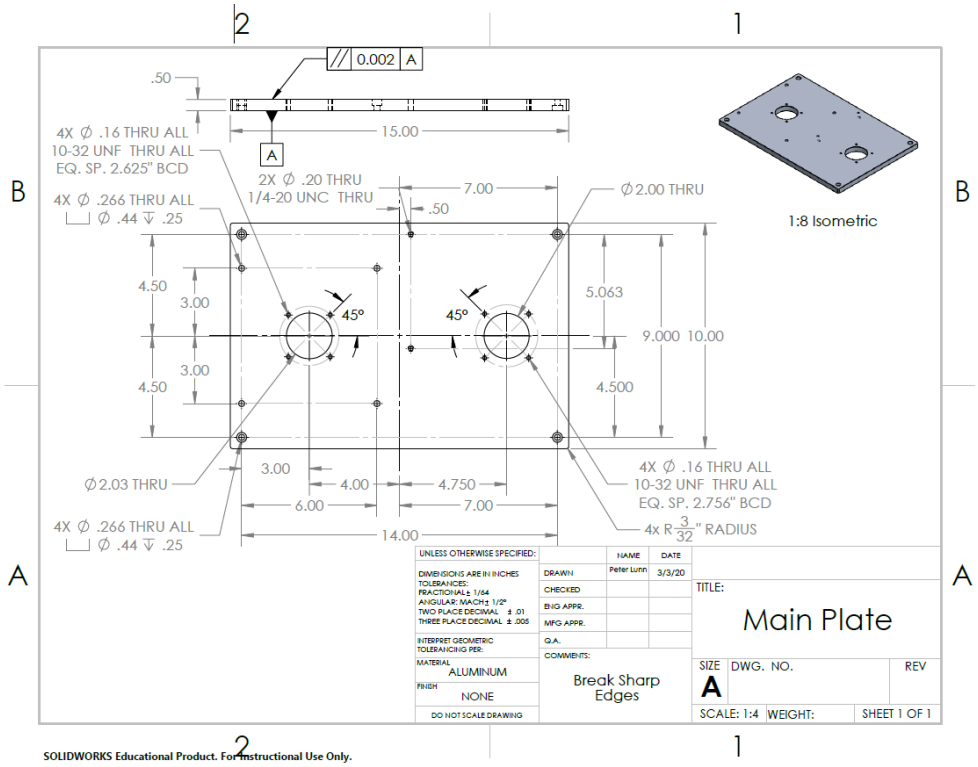

Figure 27. Main Plate Drawing

7. Mid Legs

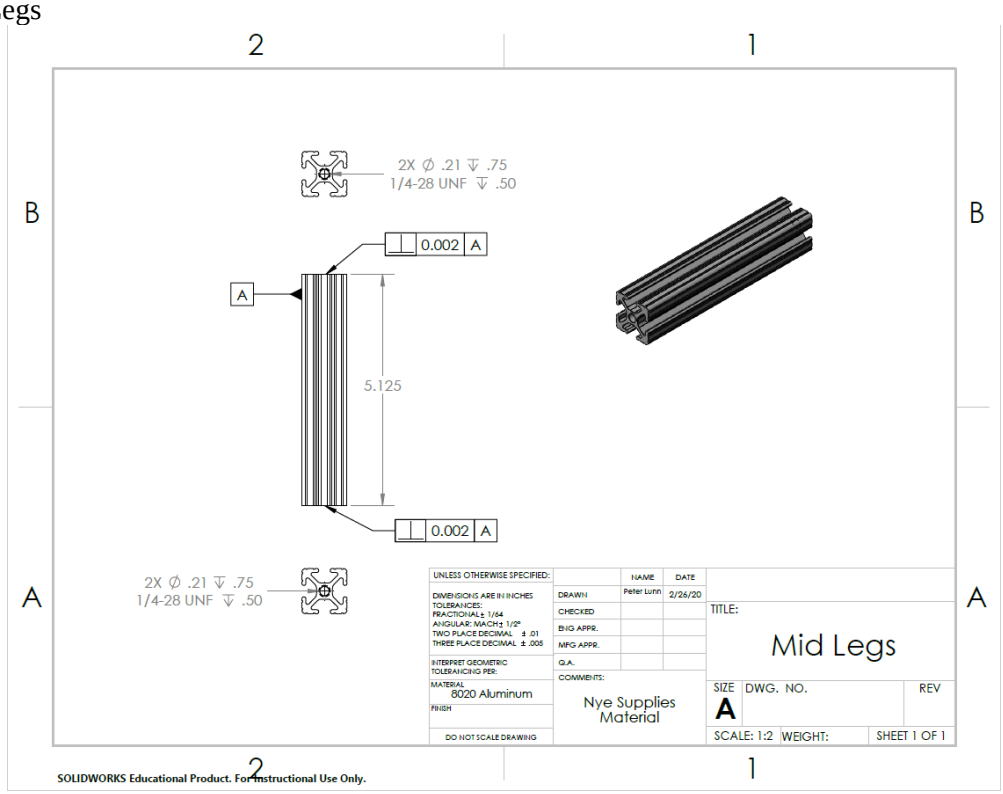

Figure 28. Mid Legs Drawing

8. Mid Plate

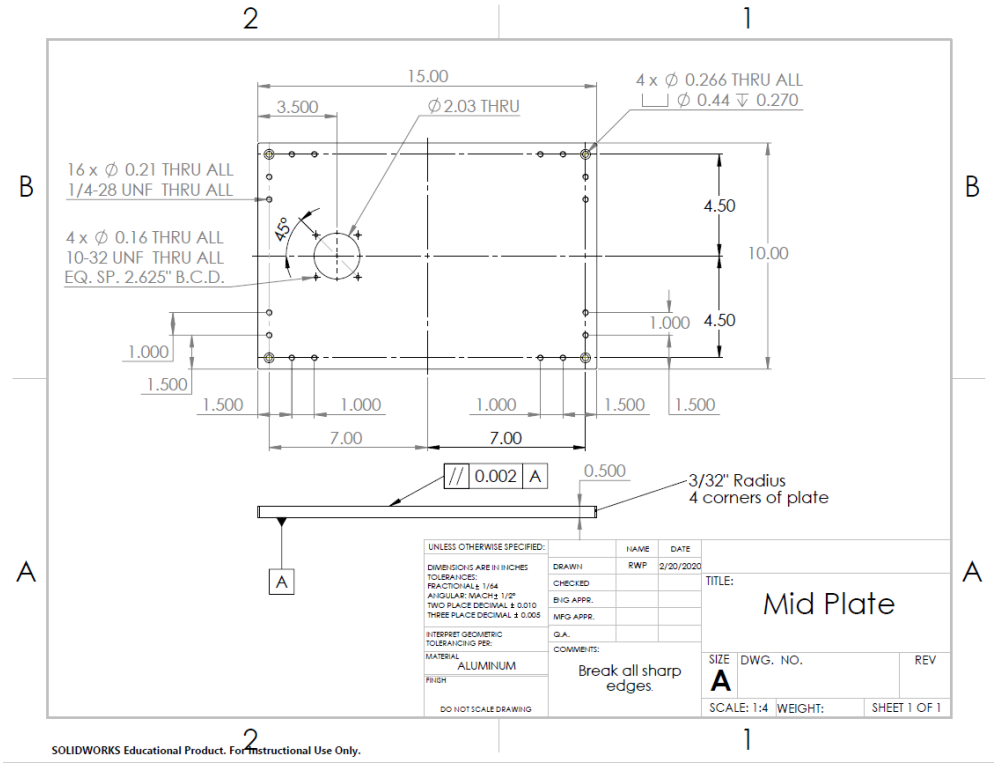

Figure 29. Mid Plate Drawing

9. Motor Shaft

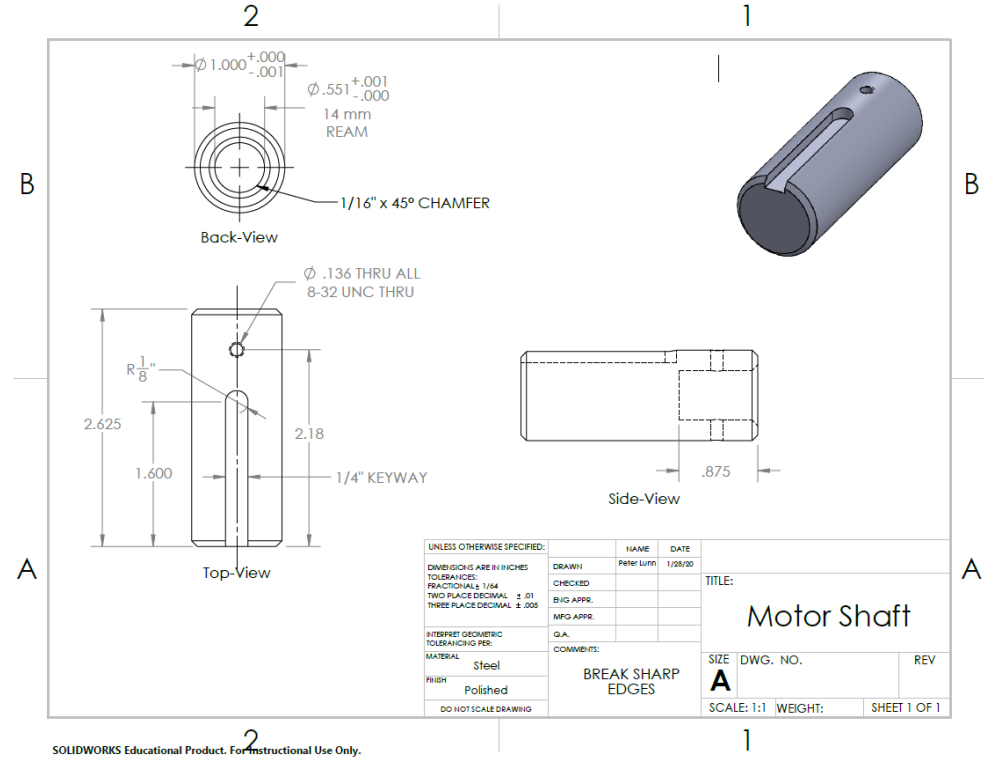

Figure 30. Motor Shaft Drawing

10. Piston Adapter

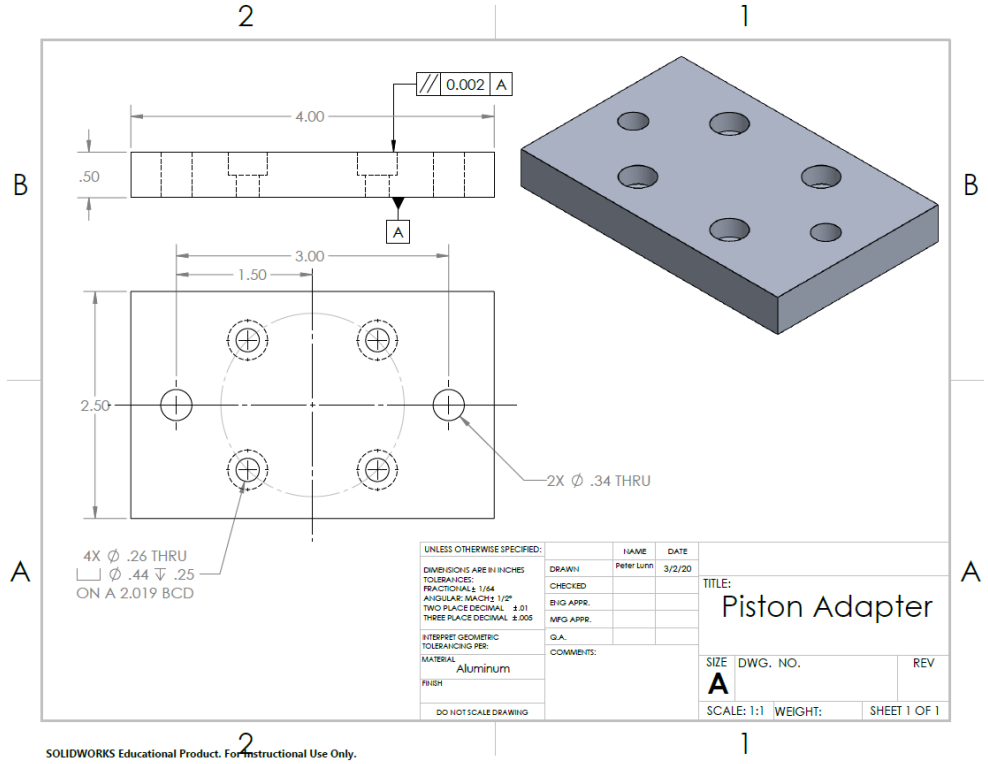

Figure 31. Piston Adapter Drawing

11. Piston Cup Adapter

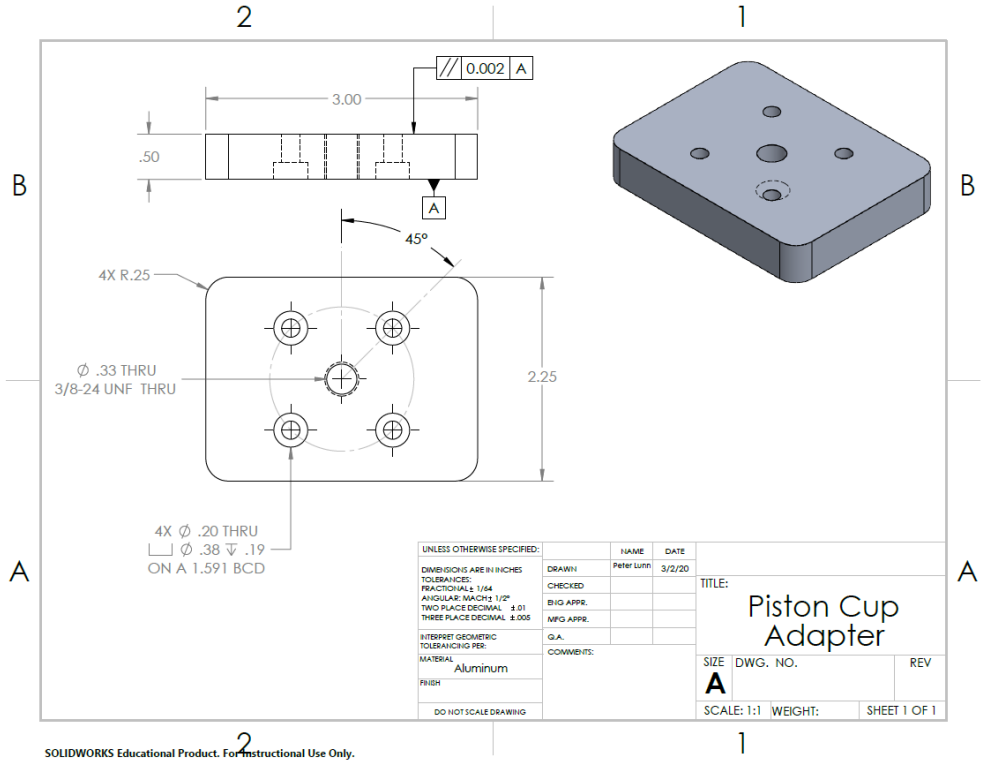

Figure 32. Piston Cup Adapter Drawing

12. Piston Cup Bottom

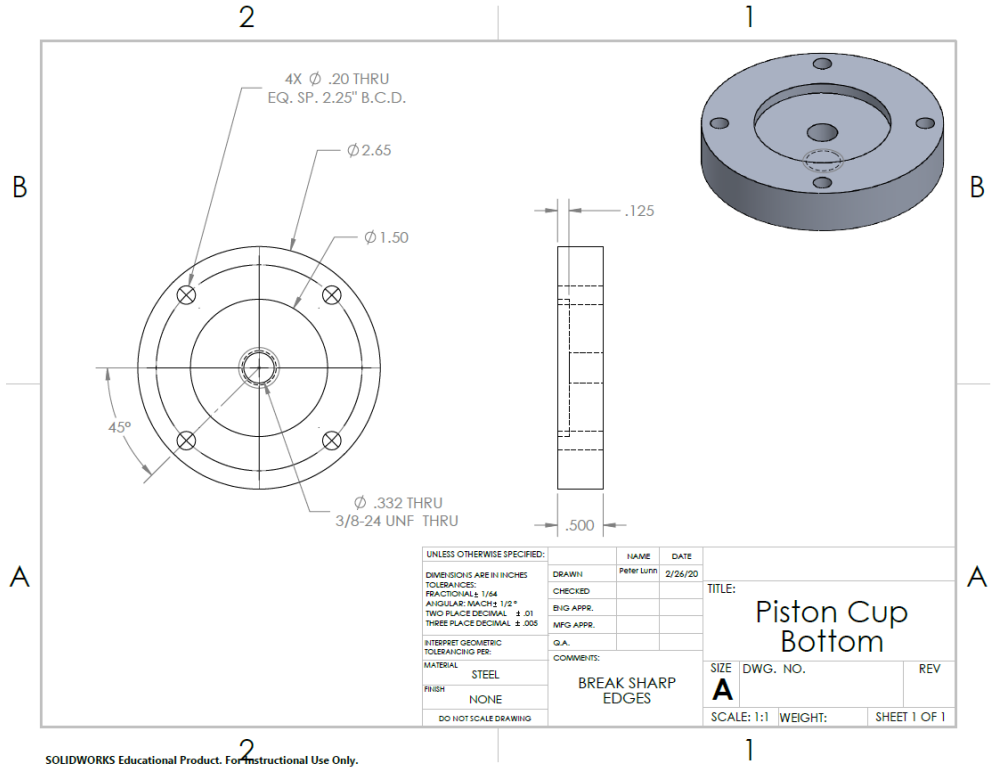

Figure 33. Piston Cup Bottom Drawing

13. Piston Cup Top

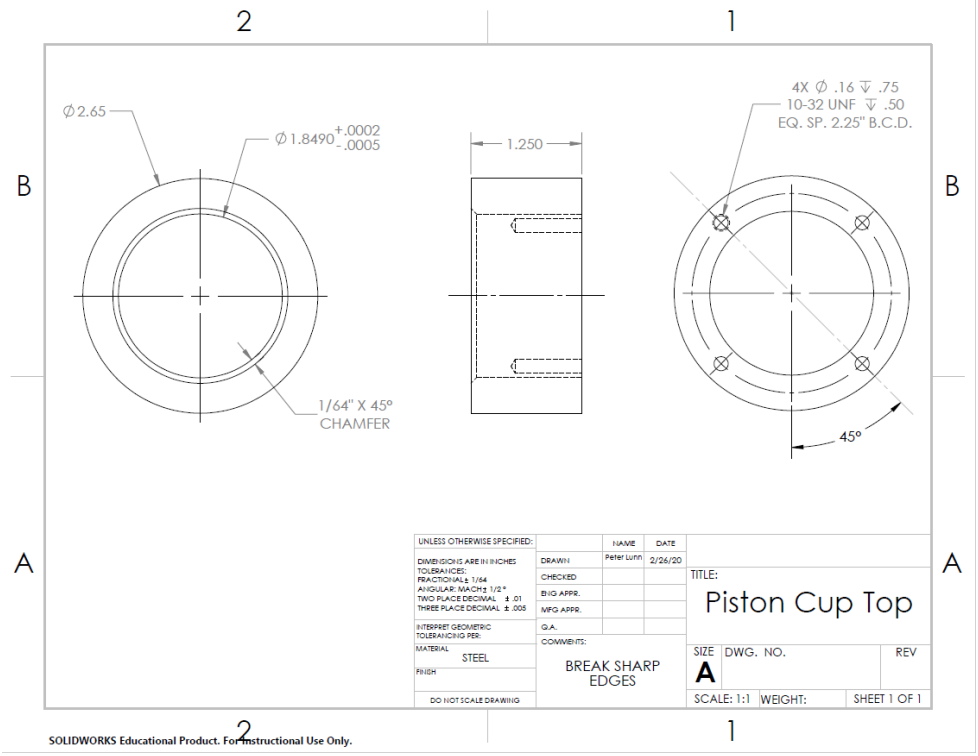

Figure 34. Piston Cup Top Drawing

14. Simplified Cylinder Nozzle

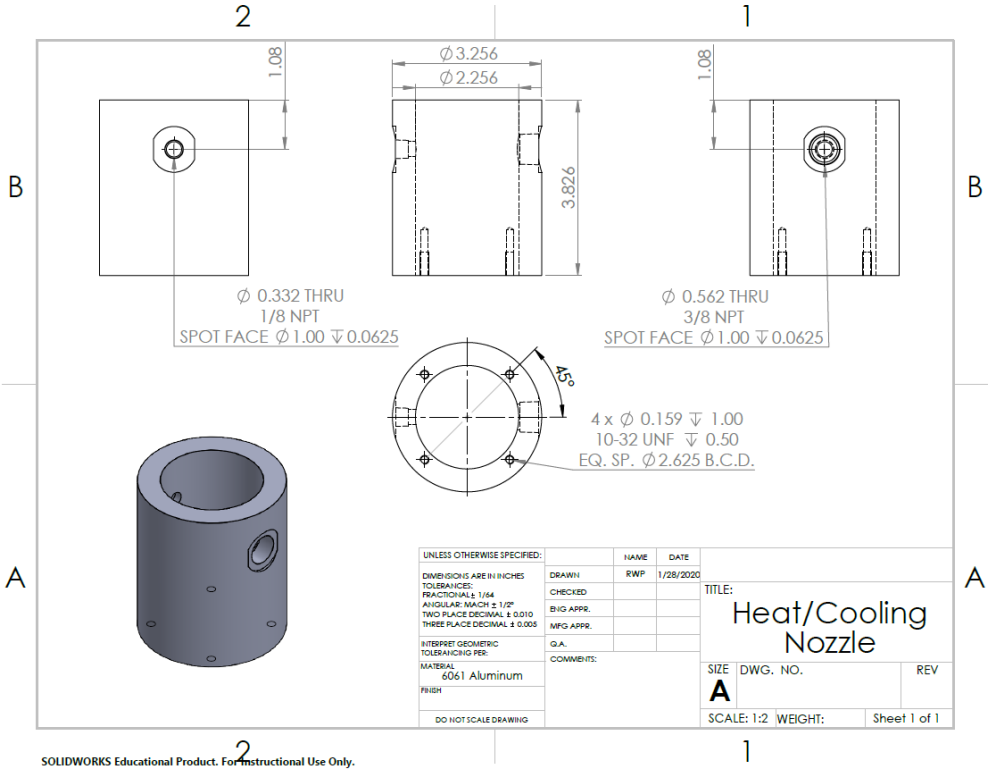

Figure 35. Simplified Cylinder Nozzle Drawing

15. Test Bearing Shaft

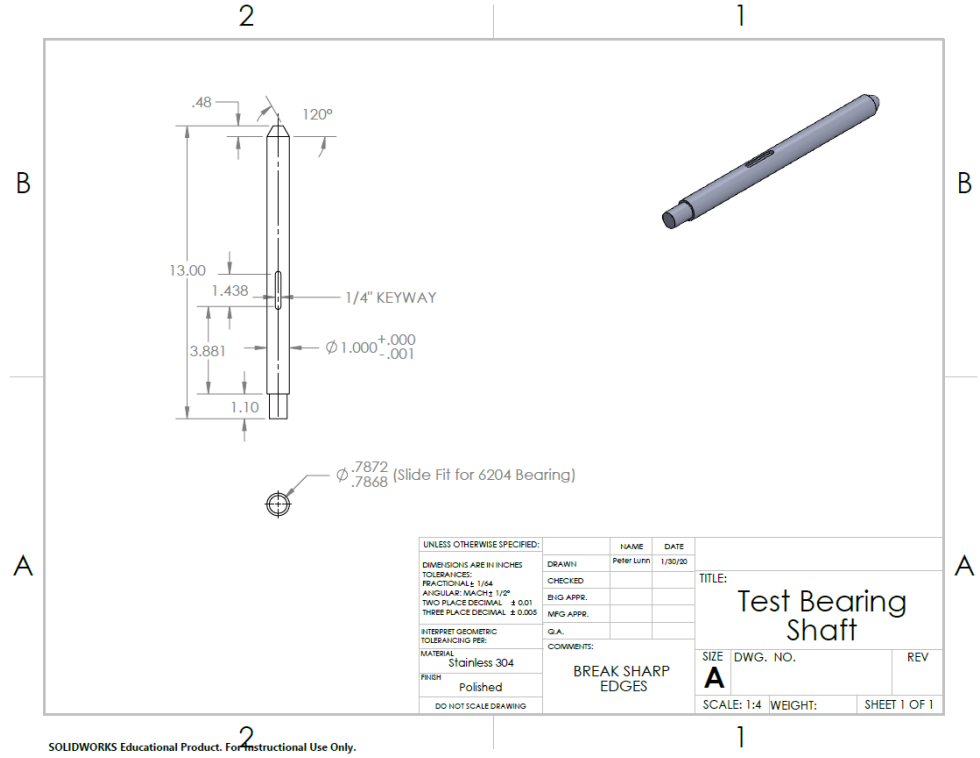

Figure 36. Test Bearing Shaft Drawing

16. Thrust Bearing Shaft

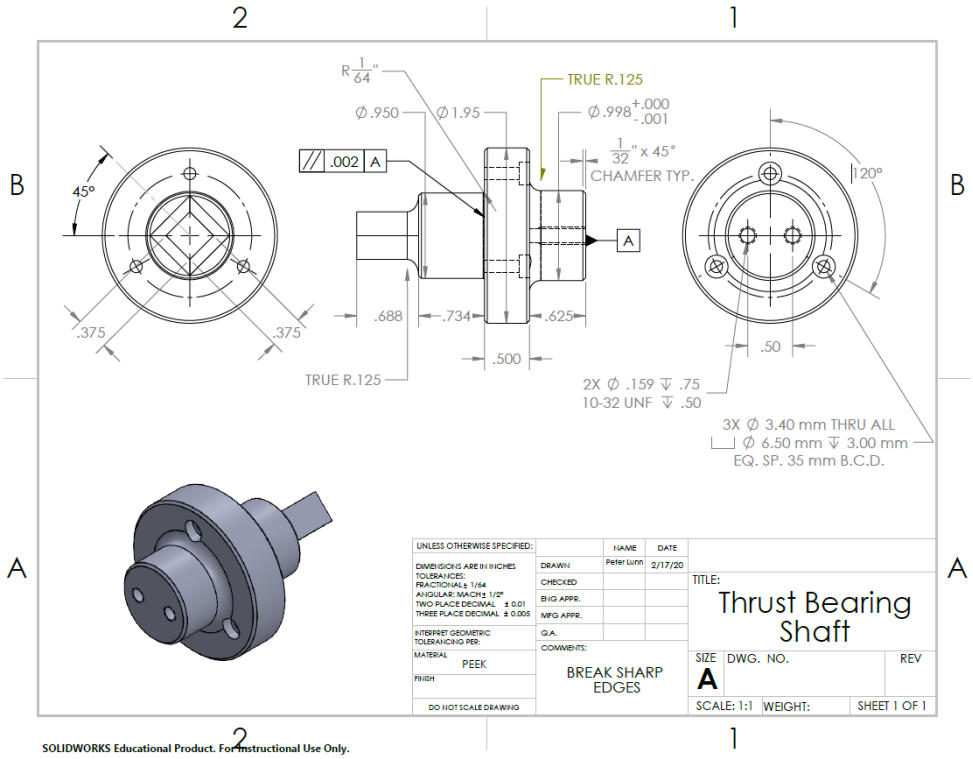

Figure 37. Thrust Bearing Shaft Drawing

17. Top Bracket

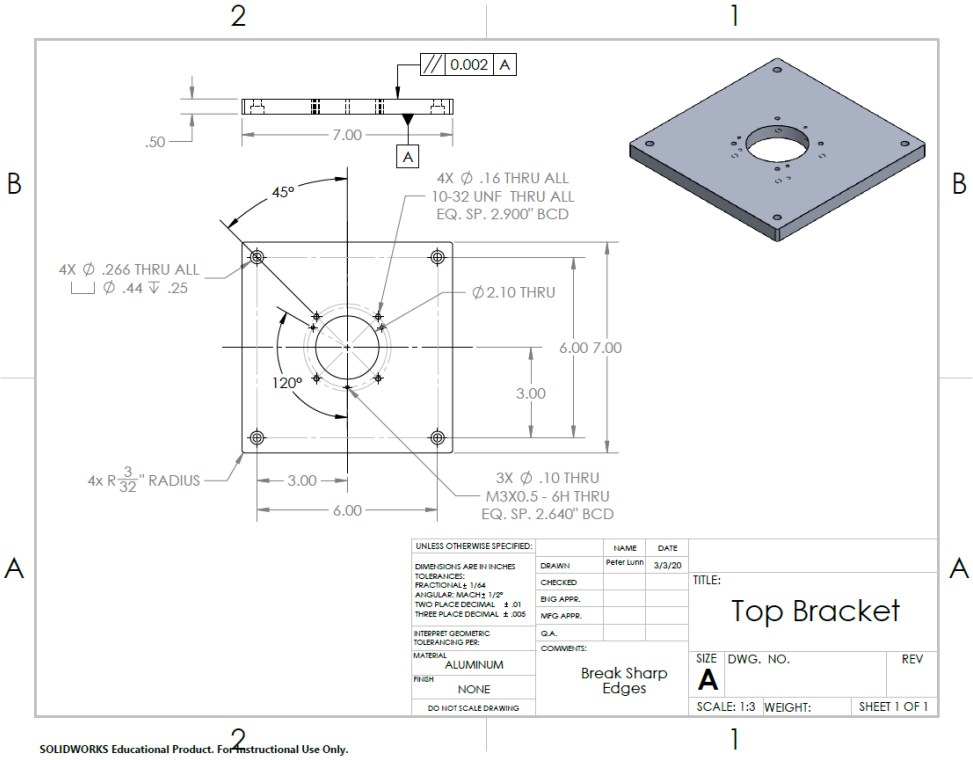

Figure 38. Top Bracket Drawing

18. Top Legs

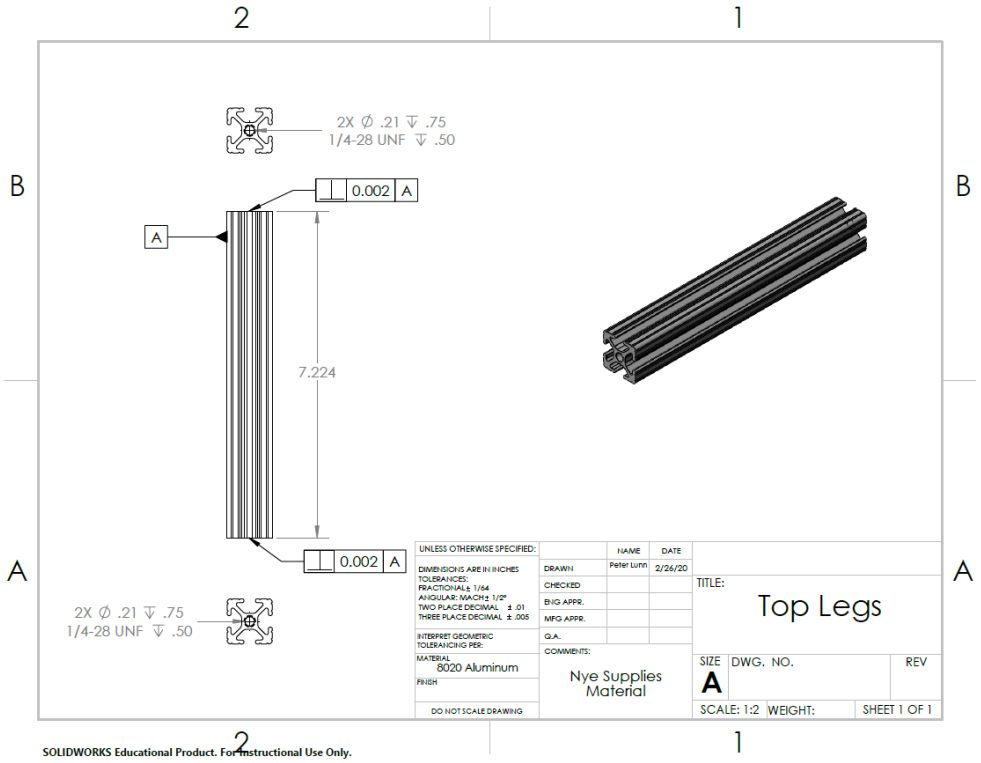

Figure 39. Top Legs Drawing

19. Torque Sensor Housing

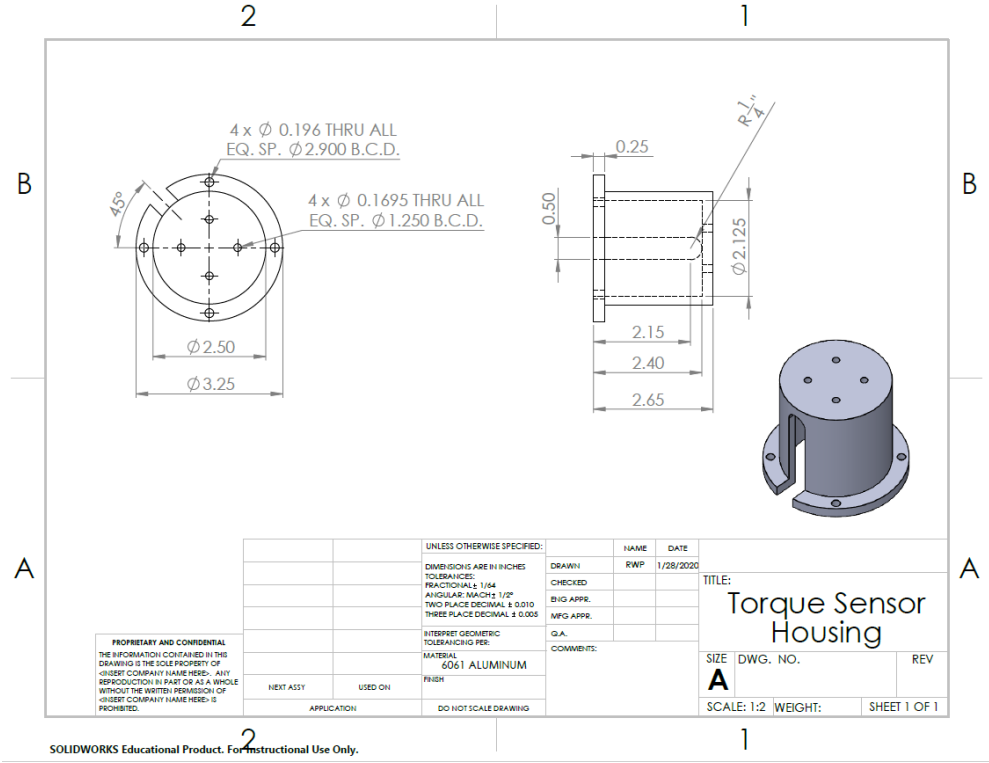

Figure 40. Torque Sensor Housing Drawing

APPENDIX B – TRELLO BOARD

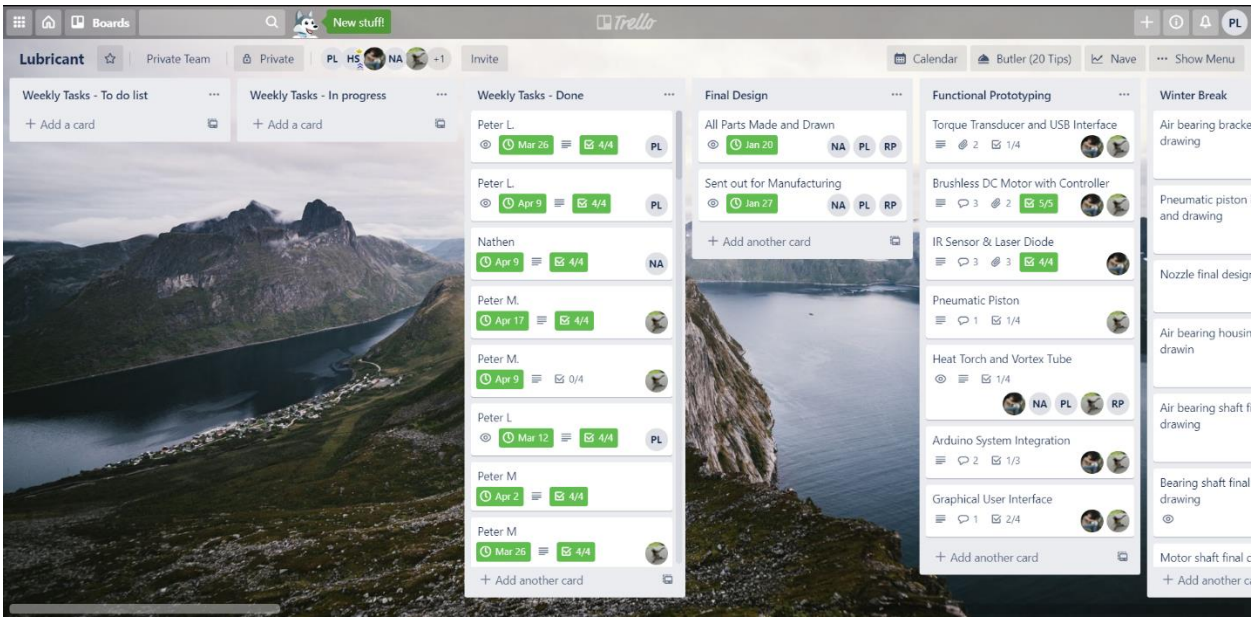

Figure 41. Trello Board Progress Tracker

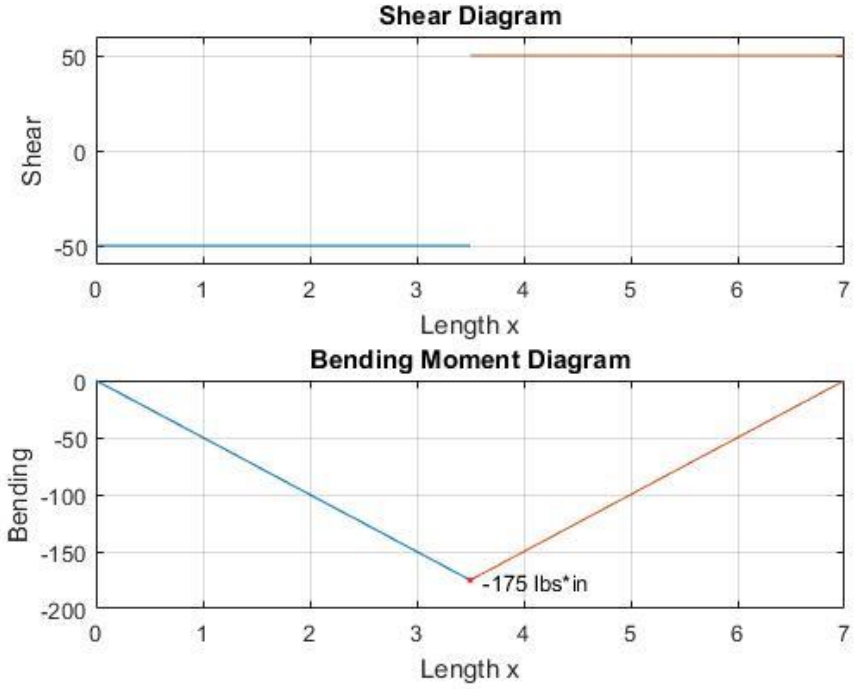

Figure 42: Shear and Bending moment diagram for large aluminum top plate.

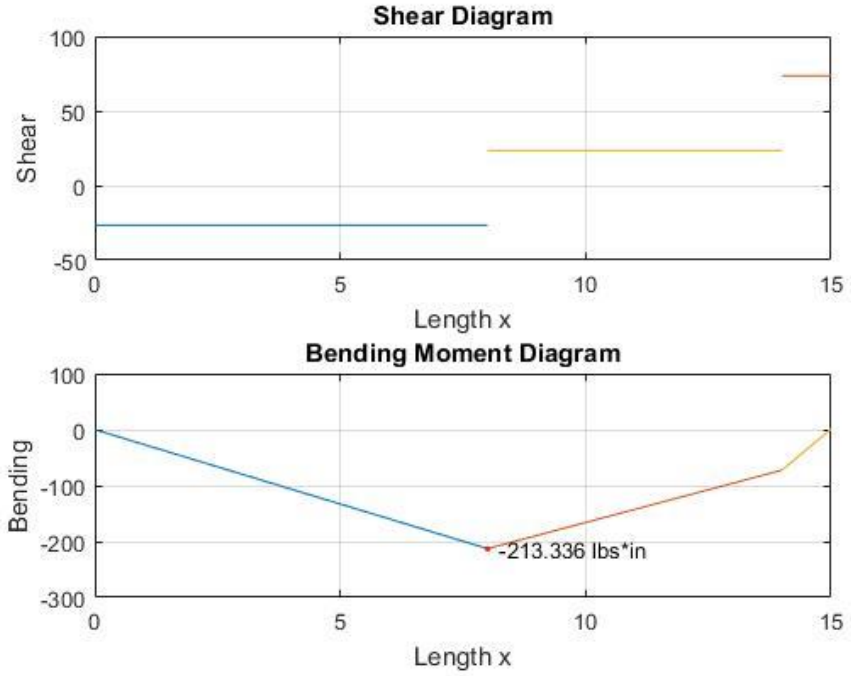

Figure 43: Shear and Bending moment diagram for top smaller top aluminum plate.