MEGN540 Mechatronics Project Final Report

Group 2: Aarushi Doctor, Dylan Cook, Nick Taylor

Link to video demonstration: https://youtu.be/Aypjt35x-sM?feature=shared

1. Problem Statement

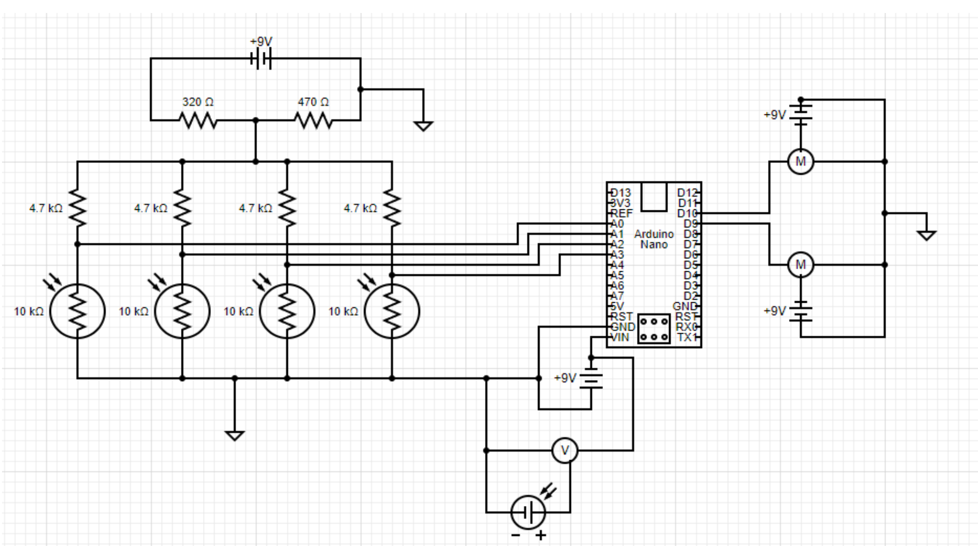
With growing concerns in the world over power consumption and production methods, renewable energy sources have seen a noticeable rise in popularity. Solar power is a great alternative to conventional power sources for many reasons. However, solar panel performance is sensitive to its orientation with respect to the location of the sun in the sky. And that relative orientation difference has a dynamic dependency on the time of day as well as the solar panel geographic location. One popular solution to this problem is to actuate the solar panel throughout time to minimize the angle of incidence between the normal vector of the solar panel surface and the direction of the sun's rays. Our project aimed to develop a small scale controller to minimize this incident angle throughout time, maximizing the solar panel energy output.

2. Design Concept

The final design is made up of three subsystem level components: an electrical subsystem consisting of a light dependent resistor (LDR) array, a mechanical system consisting of a dual-axis motor actuation system, and an arduino microcontroller used to interface between the electrical and mechanical subsystems. The term "photoresistor" and "LDR" are used interchangeably in this document.

a. Electrical Subsystem

The following figure shows the electrical schematic for the project.



i. Power Requirements

As shown in the figure above, the project is powered by four individual 9V batteries. The project regrettably does not do anything useful with the power generated by the solar panel. For reference, the solar panel is labeled as a photovoltaic cell in this electrical schematic above (shown in the bottom of the electrical schematic). The power generated by the solar panel is measured by an LCD display (shown as a voltmeter in the figure above), but this power is lost. The focus of the group was placed on the control system of the solar tracker, and efficient power management was not made a requirement for our project. Implementation of a battery charging system powered by the solar panel is a very strong consideration for future steps after completing this project.

ii. Resistor Value Decision Space

Within the electrical subsystem of this project, there are 3 locations within the holistic circuit where resistor value optimization was considered: the voltage divider leading into the photoresistor sensing array, the resistors within the photoresistor array, and then the photoresistors themselves. These resistor values are labeled in the electrical schematic above for reference.

The voltage divider resistors were determined by the voltage divider equation.

$$V_{out} = V_{in}(R_2/(R_1 + R_2))$$

To determine the resistor values, V_in was set to 9V and V_out was set to 5V. From the resistor kit we ordered at the beginning of the semester, we did our best to match the ratios governed by the voltage divider equations given the voltage requirements of the circuit, while trying to keep resistor values as small as possible to conserve power and not waste current by using unnecessarily large resistors. Resistor values that were too small did short one of our circuits while testing and fried one of our Arduinos, so we learned not to make the resistor values too small either.

The resistors within the photoresistor sensing array were chosen in conjunction with the resistor value of the photoresistors. We made this decision by setting up a breadboard circuit with one resistor in series with one photoresistor given a voltage of 5V, and then used a voltmeter to measure the voltage drop across the photoresistor. Using trial and error, we chose the resistor/photoresistor combination which provided the biggest change in voltage drop across the photoresistor, comparing when the photoresistor was exposed to significant light and when the photoresistor was exposed to very little light.

iii. Additional Electrical Subsystem Comments

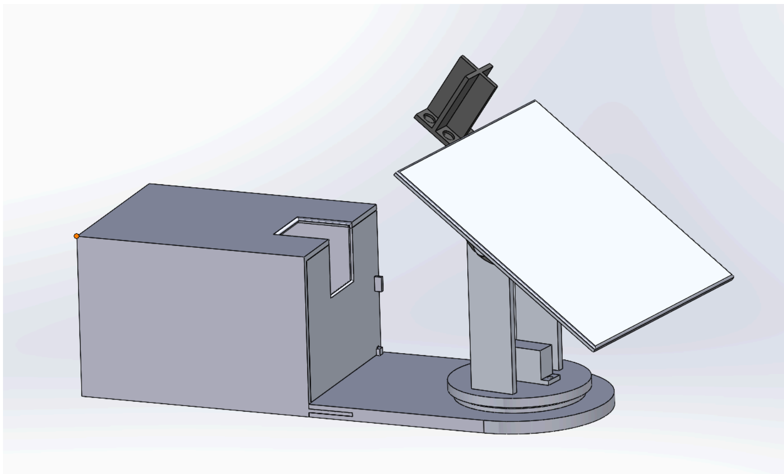
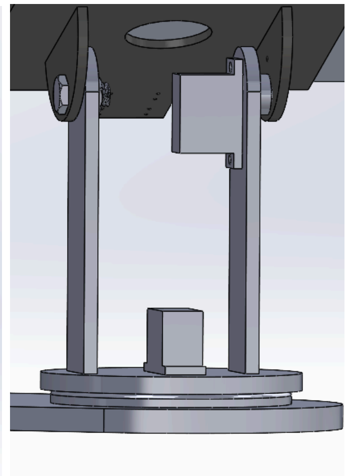
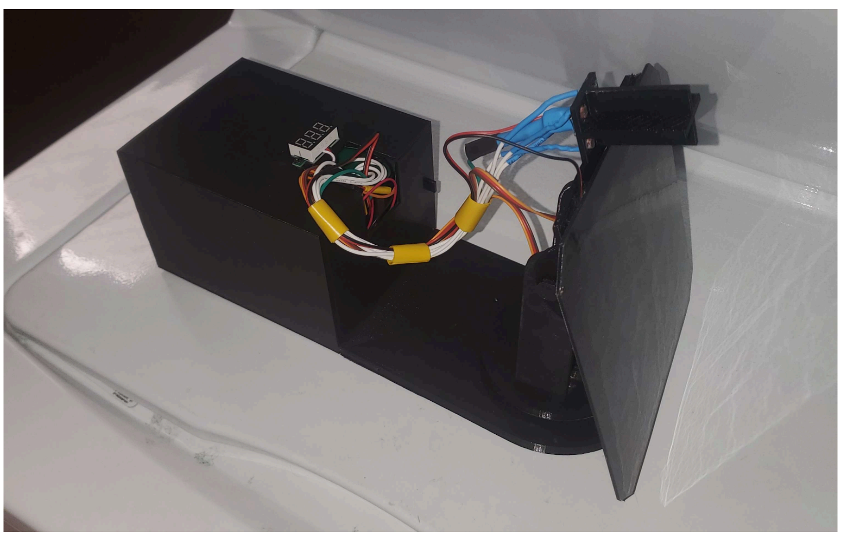
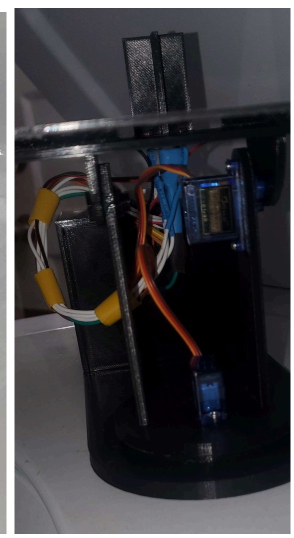
It's important to note that one individual universal ground was used for this project. The electrical schematic above shows multiple separate electrical grounds. But in actuality, these grounds were all connected to unify one universal ground for the entire project. This was done using a ground rail on a breadboard.

As previously stated, the solar panel is represented as a photovoltaic cell in this electrical schematic. The LCD display which measures the solar panel voltage output is shown as a voltmeter in this electrical schematic.

b. Mechanical Subsystem

The mechanical system is a completely 3D printed assembly designed in SolidWorks, composed of two main modules. The first module is a housing component, primarily designed to house the electrical components of the project in a remote location out of the way of the solar panel range of motion. Its location was chosen to reduce the chance of wire disconnections. This housing module also serves to help protect electrical components from environmental conditions. The main features are the sliding door, mounting holes, and wire path. The door allows for the electrical parts to be removed, evaluated or changed. Mounting holes on the bottom of the housing module allow for the PCB to mate with four three-inch standoffs which fix the PCB to the mechanical structure. This allows space for the arduino to sit directly underneath the PCB on a breadboard, reducing the required wire lengths for all arduino connections. Finally, the wire path is simply a hole in the sliding door which allows wire access in and out of the housing module.

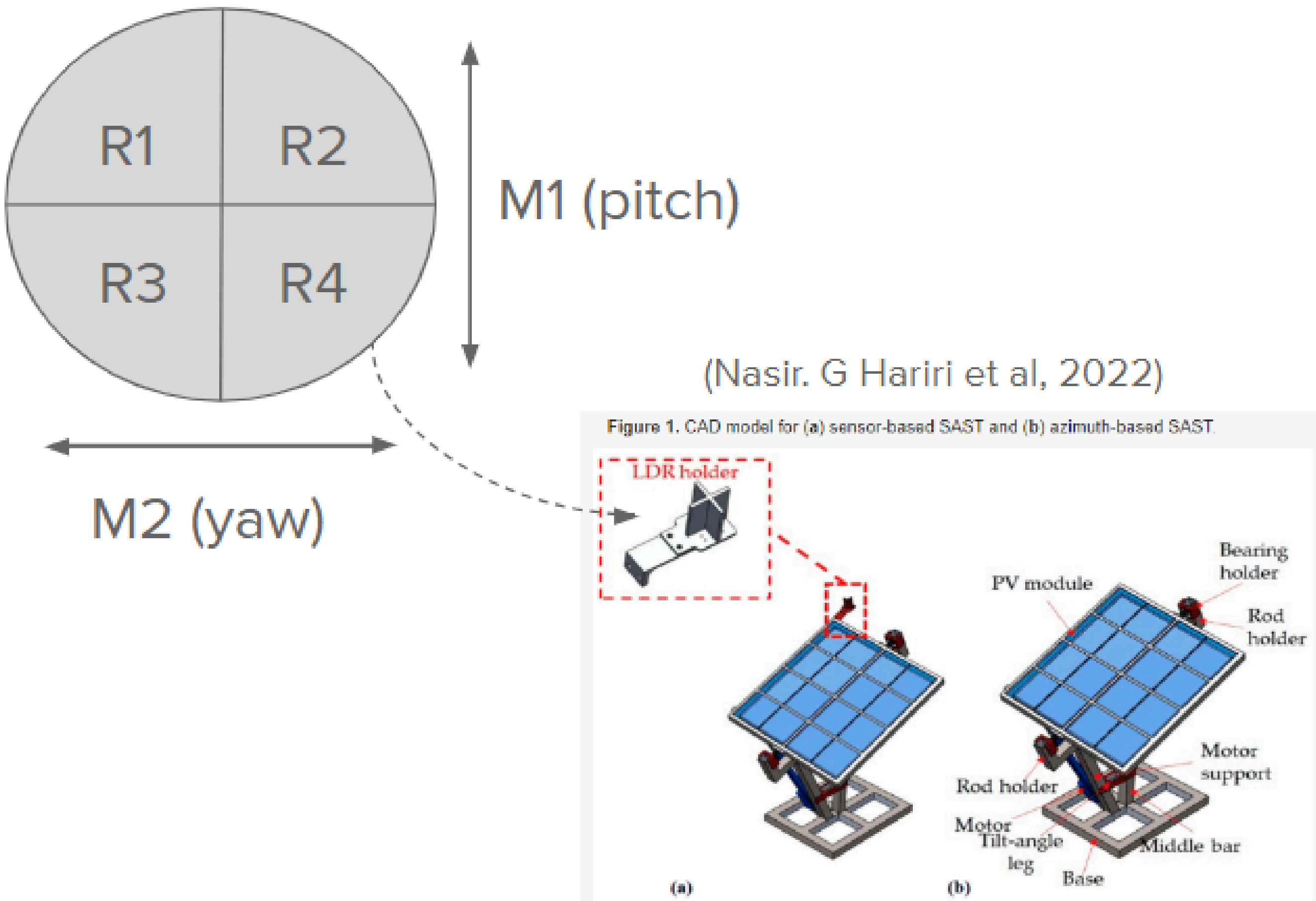
The second module of the mechanical subsystem is where the actuation takes place. This module is a dynamic structure which allows the solar panel to rotate in both pitch and yaw directions. The photoresistor array is mounted to the solar panel mounting plate, which consists of 4 photoresistor mounts situated in an array designed to cast synthetic shadows over specific photoresistors depending on the orientation of the sun within the body frame of the solar panel. The actuation is controlled by the two SG90 servo motors. The bottom motor, used for yaw control, was placed in a location to help keep the structure balanced and minimize off axis rotation. The bottom of the structure around the yaw motor sits on a ring with a concave mate to give the structure stability while reducing the amount of friction by minimizing the surface area where the two components mate. The pitch motor was mounted to a vertical column which the solar panel pivoted around.

c. Software Subsystem

The third and final subsystem of our design solution is the programming element of our project. This component serves as the mechatronics interface between the mechanical and electrical subsystems. Pseudo-code is provided at the end of this section.

The controller runs an infinite loop which starts by reading in the input signals and scaling them to represent the correct voltage drop across each photoresistor according to the reference voltage of the photoresistor sensing circuit power input. From these initial LDR voltage drop signals, the algorithm then determines what we call a voltage “cutoff”, which represents the threshold voltage signal that determines whether a given photoresistor is in the shade or not. If the given voltage drop is above the determined cutoff threshold, the algorithm assumes the photoresistor is in the shade. If the given voltage drop is below the cutoff, the algorithm assumes the given photoresistor is not in the shade and exposed to a direct light source. The controller program then runs a simple if-else conditional chain with 4 cases. We have 2 motors which can both actuate in 2 directions (positive and negative rotations, each), so 2*2 = 4. Hence, we have 4 cases: one for each possible actuation action. The condition to enter each case is determined by comparing specific photoresistor voltage drops to the cutoff voltage threshold.



From the given variable definition in the image above, each of the 4 actuation cases is determined by the voltage drops across specific photoresistors. For example, if the pitch motor needs to pitch down, that would mean that photoresistors 1 and 2 are in the dark, and that would be apparent in the voltage signals read into the arduino. We would expect the voltage drops across those photoresistors to be larger than the cutoff voltage, so the condition to enter that actuation case is written “if the voltage drop across photoresistors 1 and 2 are BOTH larger than the cutoff voltage, then…”

Each of the 4 actuation cases has an associated condition like the one just specified. Based on this condition definition, we know which motor needs to actuate according to which condition is entered. Continuing with the pitch down example from before, if the voltage drop across photoresistors 1 and 2 are BOTH larger than the cutoff voltage, then we tell the pitch motor to pitch down by a previously defined step size using commands from the Servo.h library provided by arduino. We found the system performed with the best fluidity using a step size of 2 degrees. Because the only input to the control algorithm is sensing information (and not information from the previous control loop), our project implements open loop control.

Boundary cases are implemented within this conditional chain to prevent the motors from operating outside of a proper angular position range. The pitch motor operates between 80 and 174 degrees, and the yaw motor operates between 6 and 174 degrees. These numbers were chosen for multiple reasons. First, the servo motors have a 180 degree range of motion, so we can’t operate outside of 0 and 180 degrees. We implemented a small margin to stay away from those boundaries to avoid any issues. A margin of 6 degrees was chosen because the step size we used is an even number, so keeping the margin as an even number allowed the motor positions to always be even for uniformity.

To summarize, each actuation direction has its own condition within our if-else condition chain. That condition is entered when the photoresistors opposite to that actuation direction on the photoresistor sensing array are shaded in the dark from the given light source. Once that condition is entered for a given actuation direction, the arduino tells the given motor to rotate in the appropriate direction by a step size of 2 degrees. We also implemented boundary cases which prevent each motor from operating outside of a specified range of angular position. This algorithm assumes the solar tracker system is only exposed to one individual light source, and this algorithm performs poorly when the photoresistor sensing array is exposed to multiple light sources.

The following pseudo-code is provided to demonstrate the information written in this section above.

I. Initialize all variables, servo motor structures and data communication pins
II. Start infinite loop
    A. Read in voltage drops across each photoresistor
    B. Build voltage cutoff which is dependent on all 4 photoresistor signals
    C. Enter if-else conditional chain
        1. if R1 > cutoff AND R3 > cutoff
            a) if yaw motor is not close to exiting proper operating range
                (1) Tell yaw motor to pitch LEFT (left in its own body frame)
        2. else if R2 > cutoff AND R4 > cutoff
            a) if yaw motor is not close to exiting proper operating range
                (1) Tell yaw motor to pitch RIGHT (in its own body frame)
        3. else if R3 > cutoff AND R4 > cutoff
            a) if pitch motor is not close to exiting proper operating range
                (1) Tell pitch motor to pitch UP
        4. else if R1 > cutoff AND R2 > cutoff
            a) if pitch motor is not close to exiting proper operating range
                (1) Tell pitch motor to pitch DOWN
    D. Run a delay command and repeat loop

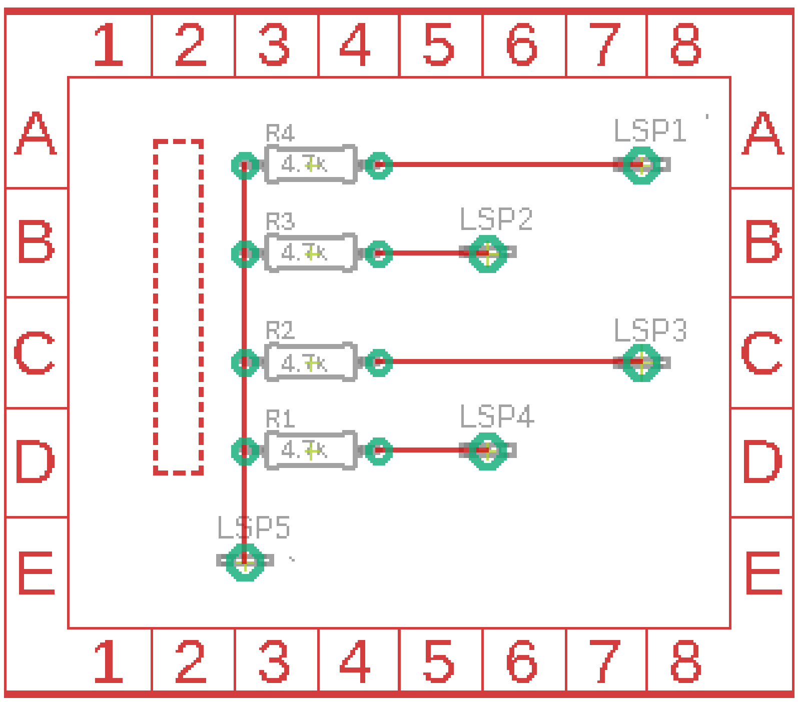
3. Project Requirements

The purpose of this section is to overview the project requirements as a course assignment and highlight how they fit into the report up until this section.

a. PCB Design

The PCB was designed to link the photoresistor array to a power source while including baseline resistors. If any photoresistors were exposed to a significant amount of light, their resistance would drop. If baseline resistors were not included in the photoresistor sensing circuit and the photoresistors resistance dropped very low after being exposed to significant amounts of light, the circuit could short. The purpose of these subsequent resistors is to prevent each photoresistor sub-circuit from shorting in the event that the photoresistor supplied resistance was too low.

The PCB consists of 4 resistors and 5 solder pads. The 4 resistors are connected in parallel between a power source and the solder pads leading to the rest of the photoresistor sensing circuit. The 4 solder pads following the baseline resistors were soldered to both a single photoresistor, and an input pin to the arduino. The voltage running through these solder pads represents the voltage drop across each individual photoresistor. The fifth solder pad was used to connect the circuit to the power source. In actuality, this solder pad was connected to a voltage divider to supply the circuit with a 5V power signal. The design of the PCB was implemented using Eagle software. Initially, the schematic was drafted, followed by the creation of the board layout. Below is the completed board layout:



b. Code Implementation

The photoresistor sensing array sits in an electrical circuit conveniently designed to provide the arduino nano microcontroller with the voltage drop across each photoresistor for every control loop. With knowledge of the voltage drop across each photoresistor, the arduino can distinguish which photoresistors are in the shade, and which photoresistors are sitting in direct light. Our project implemented open loop control to take whichever photoresistors are in the shade, out of the shade. This was accomplished by taking in voltage drops across 4 photoresistors, running that information through a coding algorithm to determine which photoresistors are in the shade, figuring out which motor needs to actuate in which direction to take those photoresistors out of the shade, and then sending PWM signals to the appropriate motor. For more explicit details, please see section 2.c above.

c. Sensing and Actuation

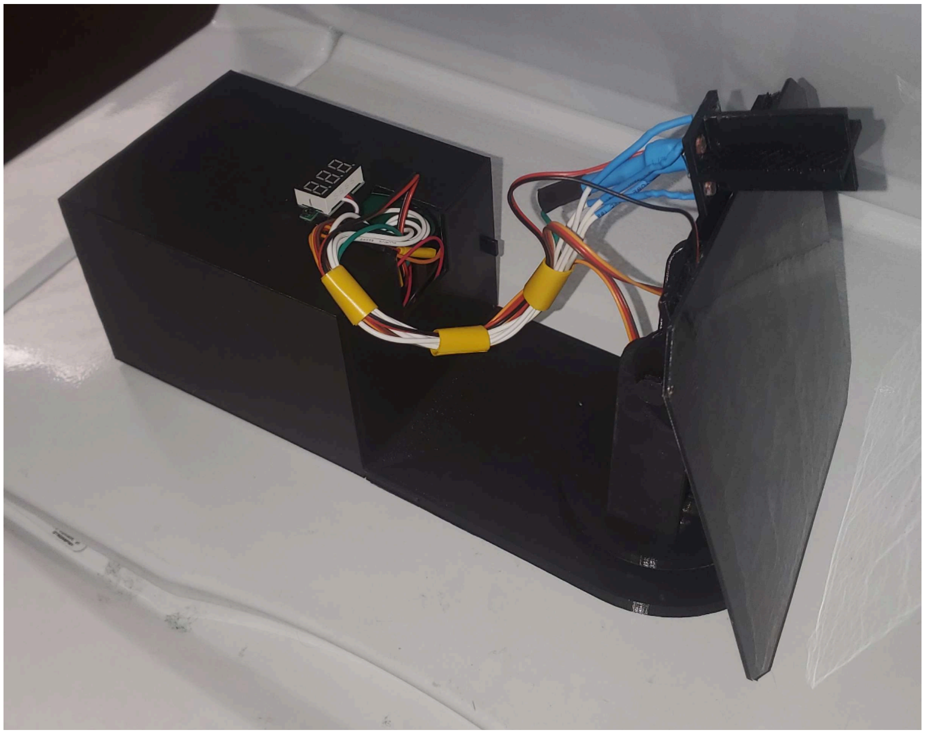
i. Sensing

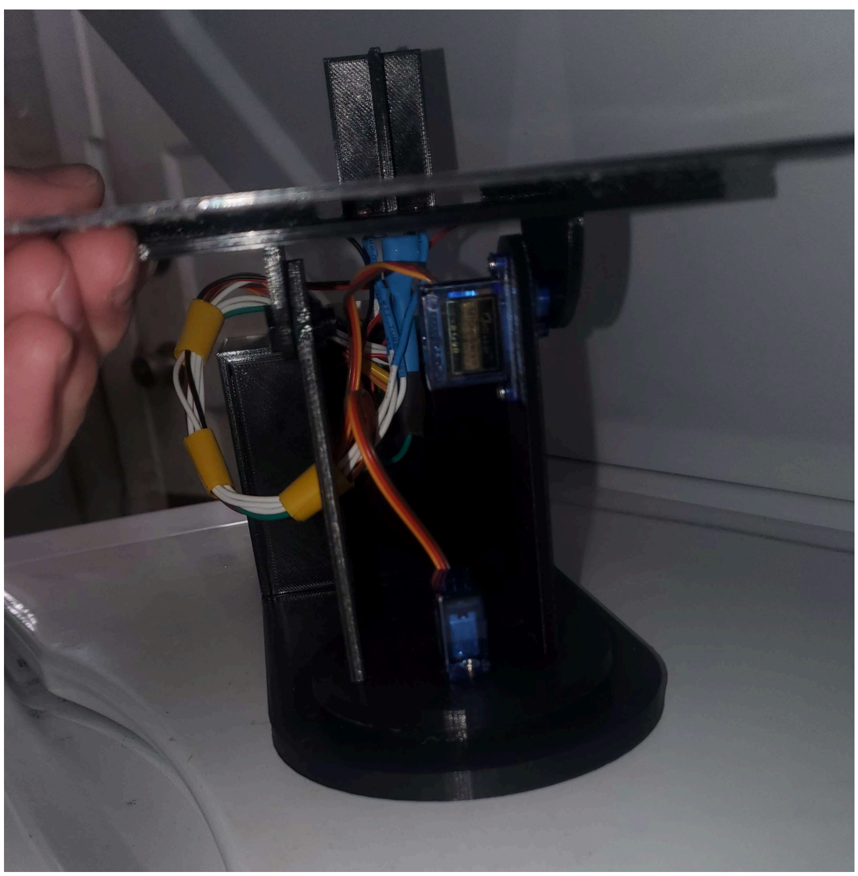
The photoresistor sensing array module makes up the sensing component of our design for this project. The photoresistors are mounted to a structure designed to cast synthetic shadows over different photoresistors depending on the orientation of the solar panel with respect to a given light source. As the amount of light each photoresistor is exposed to changes in time, the photoresistor also changes resistance. As a result of the circuit we designed, the voltage drop across each photoresistor varies as it changes resistance. By reading the voltage drop across each photoresistor, we are able to glean information about where the sun is located within the body axis of the solar panel. From that information, we are able to send control signals to the dual axis motor system to actuate the solar panel so that its normal vector aligns with the direction of the sun's rays, minimizing the incident angle of the solar panel and maximizing its energy output.

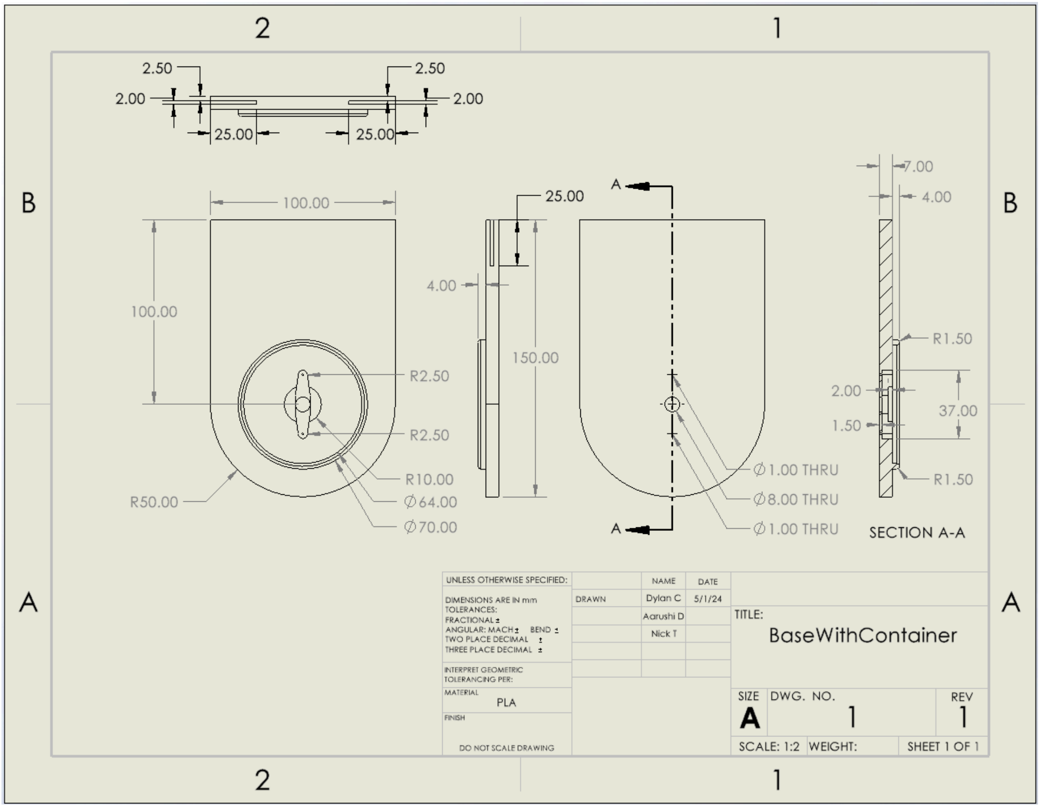
ii. Actuation

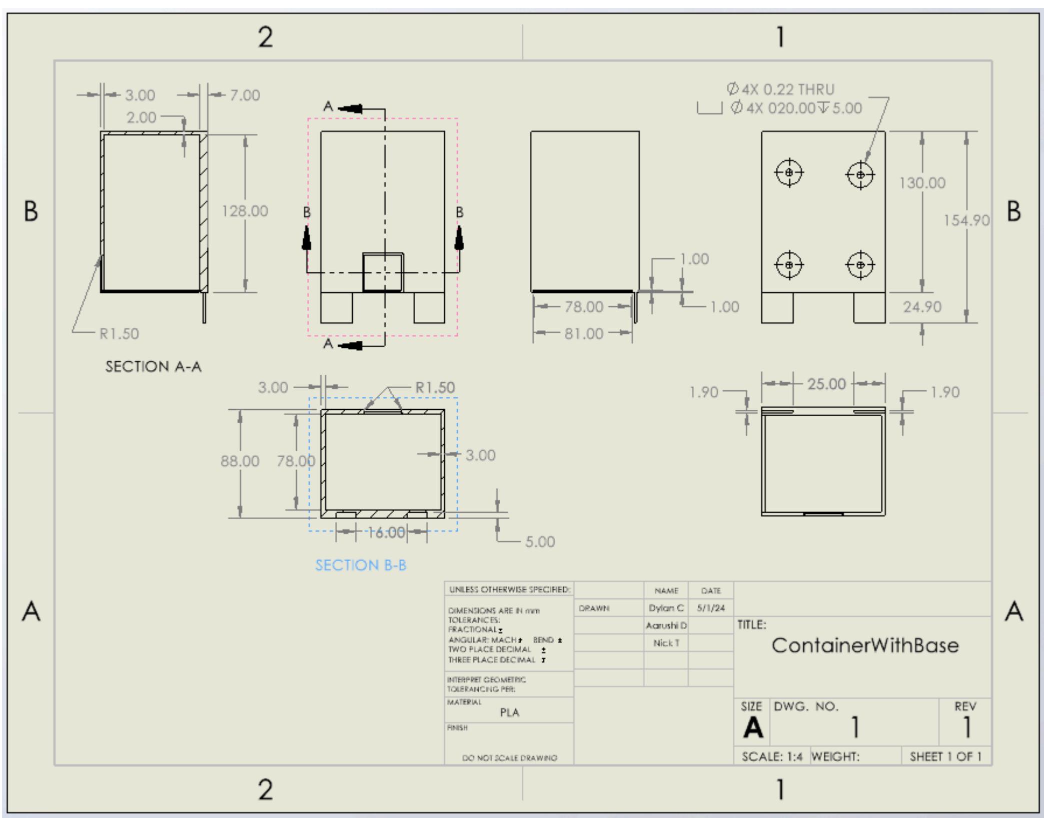
The dual axis motor system makes up our actuation system for this project, which consists of a pitch motor and a yaw motor. Each motor receives PWM signals from the arduino, changing the solar panel orientation 2 degrees every loop at an update rate on the order of 1 ms. Both motors are SG90 servo motors, which are small lightweight servo motors with 180^o of actuation. The motors also came with 3 different "horns" , which are different sized arms used for mounting. Despite being one of the smallest servos on the market and very inexpensive, they are surprisingly powerful. With a weight of 14.7 grams, they have the capability to output 2.5 kg-cm of torque, only requiring ~9V. Interestingly, we ran into an issue where the motors did not function correctly given an input of 5V. Even though the motors are rated for a power input range of [4.8V, 6V], the motors only functioned correctly for us when provided a power input above 8V.

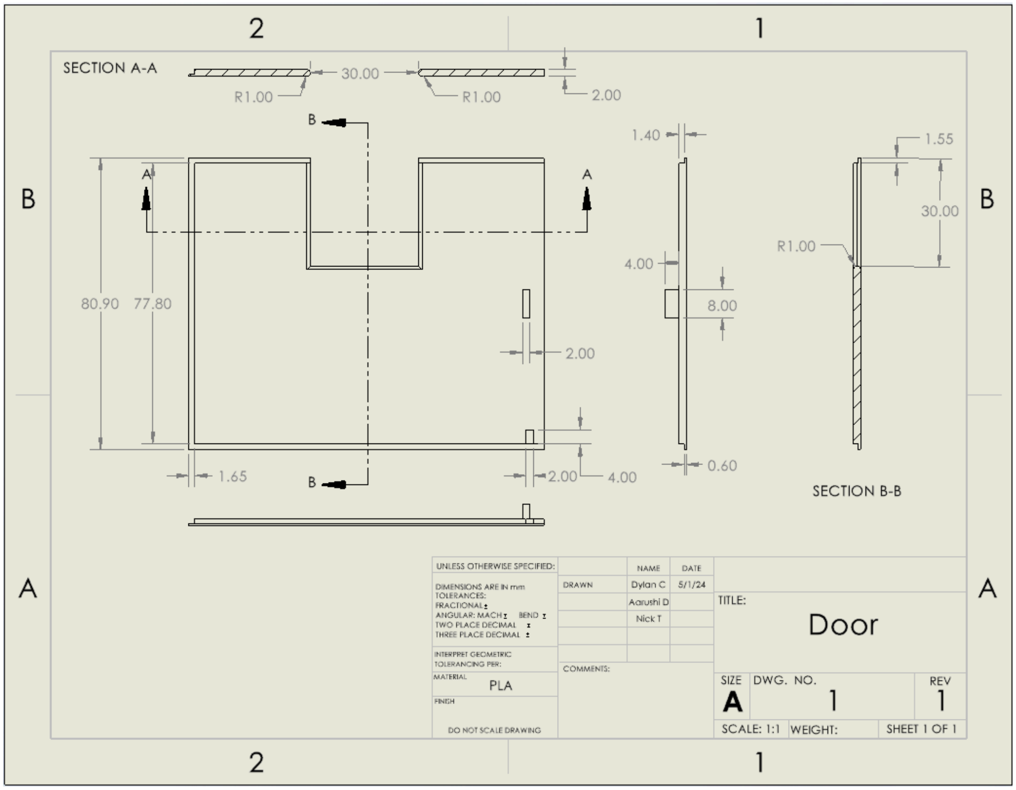
4. Image Appendix

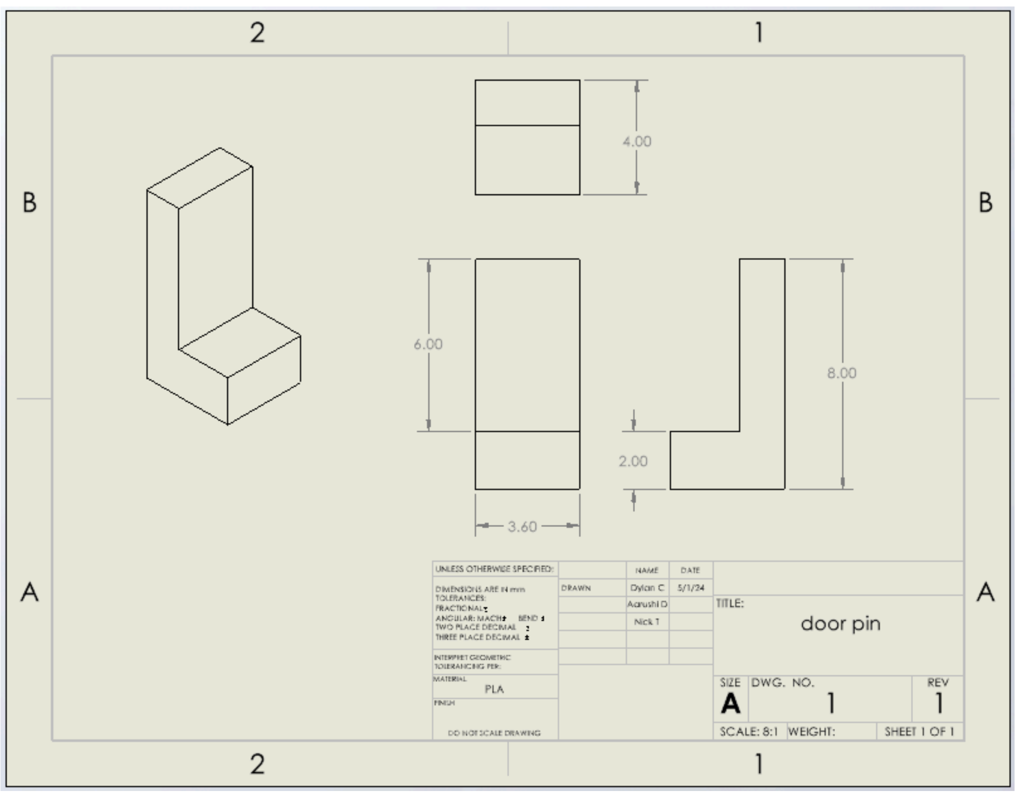


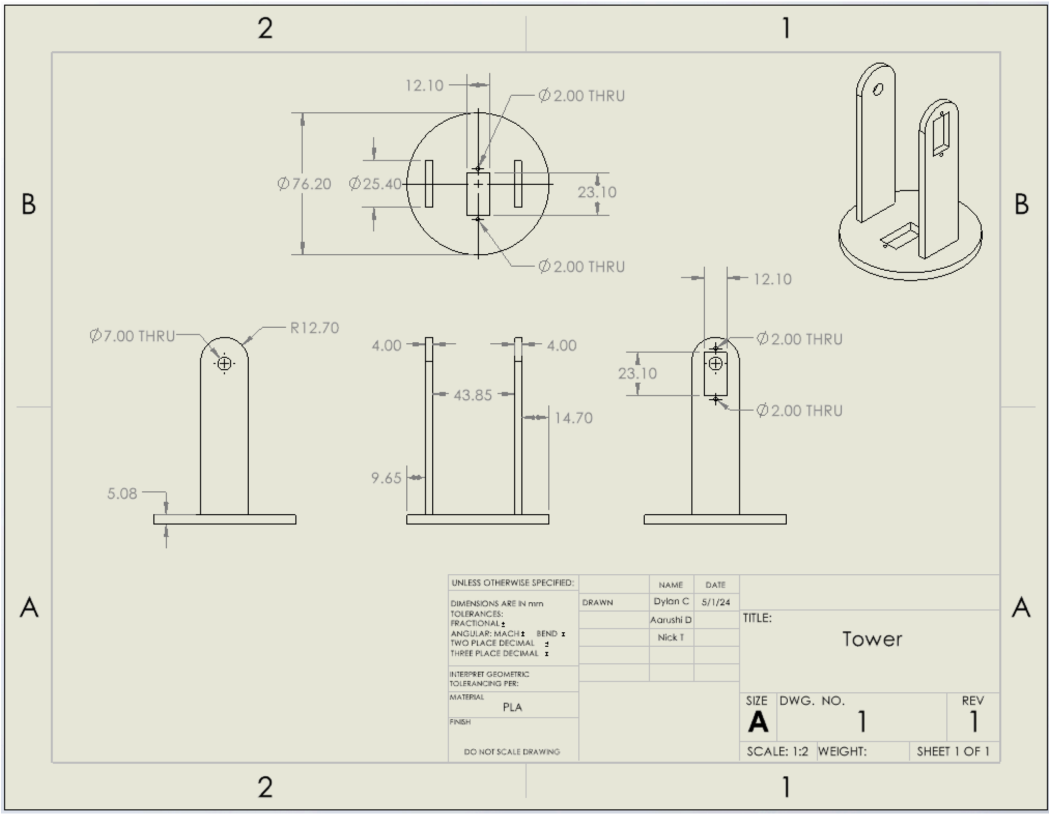


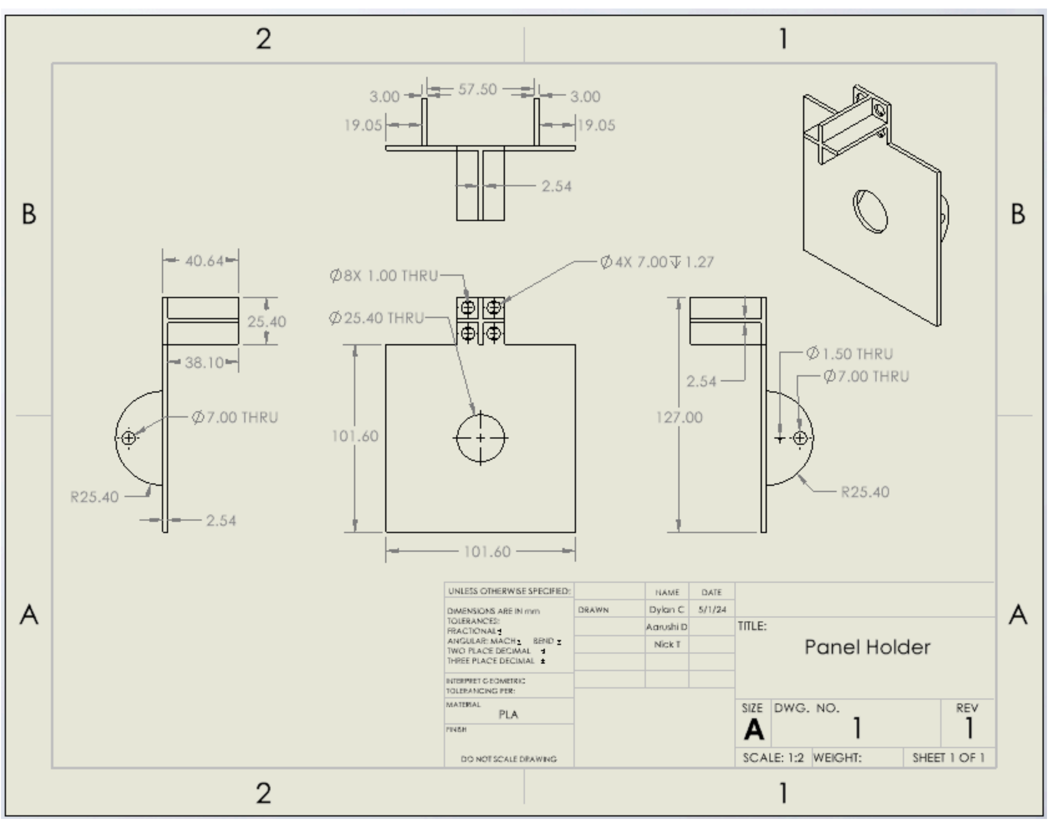

2.50
2.00
25.00
100.00
25.00
4.00
150.00
R2.50
R10.00
R50.00
Ø64.00
Ø70.00
Ø1.00 THRU
Ø8.00 THRU
7.00
R1.50
37.00
1.50
SECTION A-A
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN mm
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
PLA
FINISH
DO NOT SCALE DRAWING
NAME
DATE
DRAWN
Dylan C
5/1/24
Aarushi D
Nick T
TITLE:
BaseWithContainer
SIZE
A
DWG. NO.
1
REV
1
SCALE: 1:2
WEIGHT:
SHEET 1 OF 1

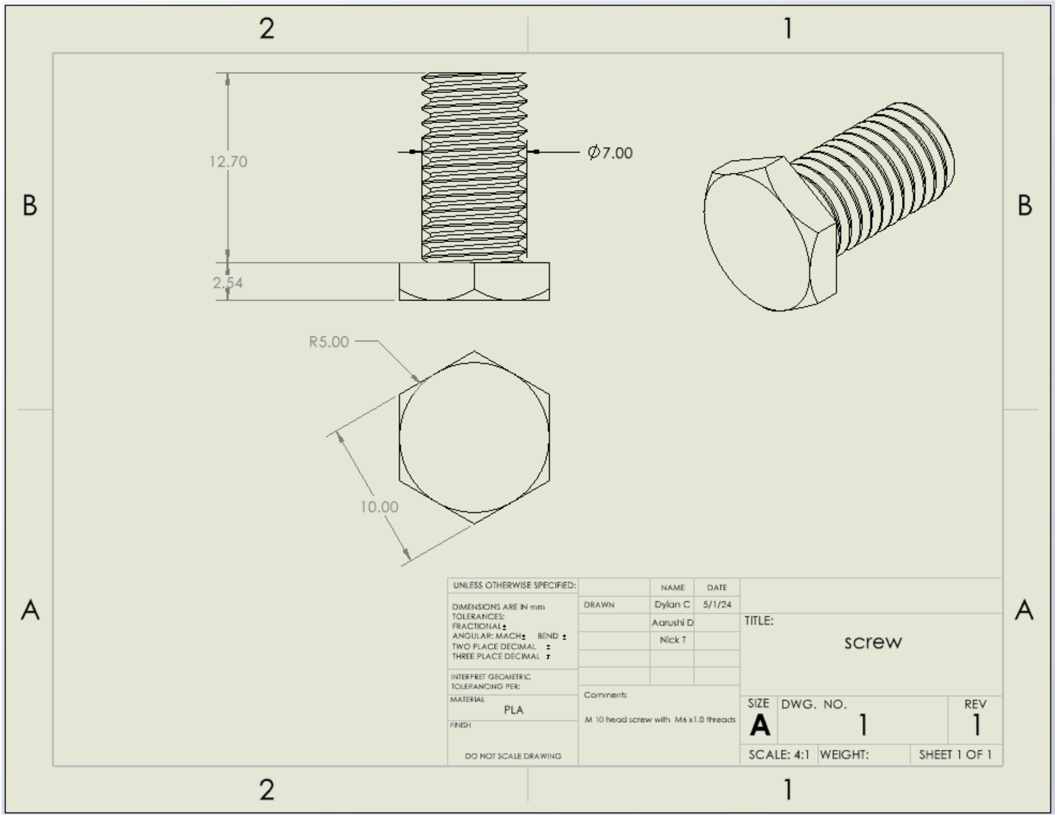

3.00
7.00
2.00
128.00
R1.50
SECTION A-A
1.00
78.00
81.00
Ø4X 0.22 THRU
Ø4X 020.00 ↧ 5.00
130.00
154.90
24.90
88.00
78.00
16.00
5.00
SECTION B-B
1.90
25.00
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN mm
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
PLA
FINISH
DO NOT SCALE DRAWING
NAME
DATE
DRAWN
Dylan C
5/1/24
Aarushi D
Nick T
TITLE:
ContainerWithBase
SIZE
A
DWG. NO.
1
REV
1
SCALE: 1:4
WEIGHT:
SHEET 1 OF 1

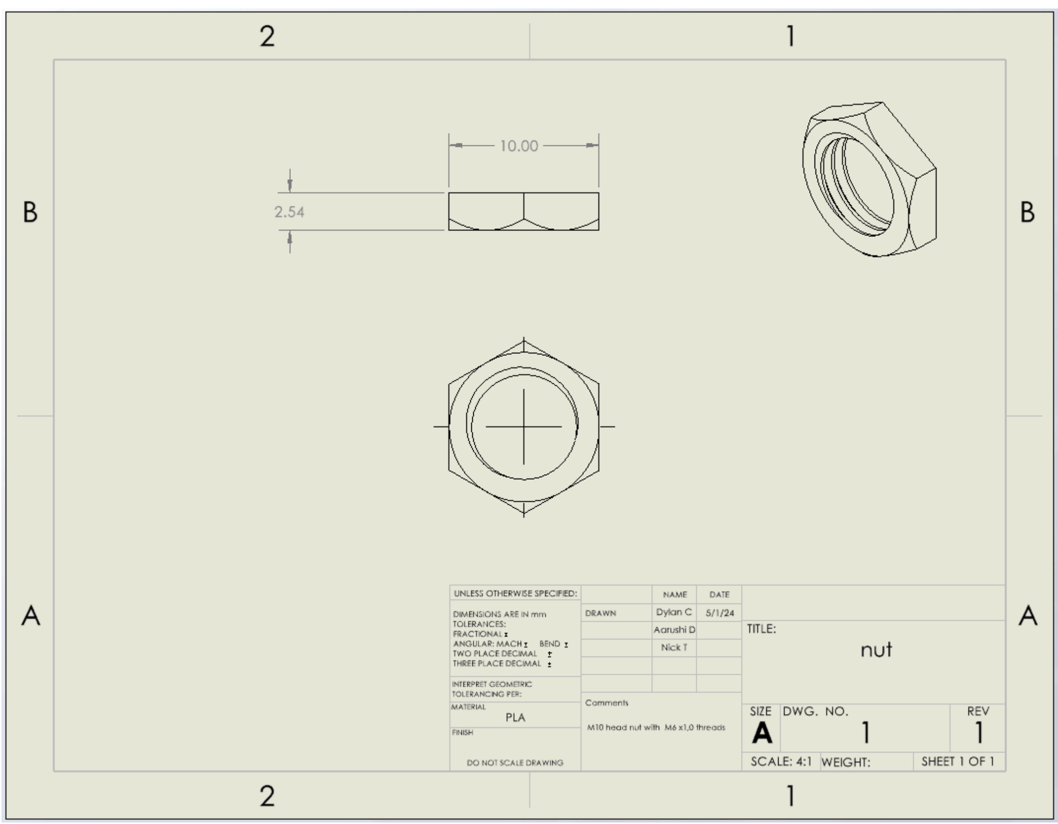

SECTION A-A
30.00
R1.00
R1.00
2.00
1.40
1.55
30.00
R1.00
4.00
8.00
2.00
80.90
77.80
1.65
2.00
4.00
0.60
SECTION B-B
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN mm
NAME
DATE
DRAWN
Dylan C
5/1/24
Aarushi D
Nick T
TITLE:
Door
MATERIAL
PLA
SIZE
A
DWG. NO.
1
REV
1
SCALE: 1:1
WEIGHT:
SHEET 1 OF 1
DO NOT SCALE DRAWING


4.00
6.00
8.00
2.00
3.60
UNLESS OTHERWISE SPECIFIED:
NAME
DATE
DRAWN
Dylan C
5/1/24
Aarushi D
Nick T
TITLE:
door pin
MATERIAL
PLA
SIZE
A
DWG. NO.
1
REV
1
SCALE: 8:1
WEIGHT:
SHEET 1 OF 1
DO NOT SCALE DRAWING


12.10
Ø2.00 THRU
Ø76.20
Ø25.40
23.10
Ø2.00 THRU
12.10
Ø7.00 THRU
R12.70
4.00
4.00
43.85
14.70
Ø2.00 THRU
23.10
Ø2.00 THRU
9.65
5.08
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN mm
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
PLA
FINISH
DO NOT SCALE DRAWING
NAME
DATE
DRAWN
Dylan C
5/1/24
Aarushi D
Nick T
TITLE:
Tower
SIZE
A
DWG. NO.
1
REV
1
SCALE: 1:2
WEIGHT:
SHEET 1 OF 1


3.00
57.50
3.00
19.05
19.05
2.54
40.64
25.40
38.10
Ø7.00 THRU
R25.40
2.54
Ø8X 1.00 THRU
Ø4X 7.00 ↧ 1.27
Ø25.40 THRU
101.60
101.60
2.54
127.00
Ø1.50 THRU
Ø7.00 THRU
R25.40
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN mm
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
PLA
FINISH
DO NOT SCALE DRAWING
NAME
DATE
DRAWN
Dylan C
5/1/24
Aarushi D
Nick T
TITLE:
Panel Holder
SIZE
A
DWG. NO.
1
REV
1
SCALE: 1:2
WEIGHT:
SHEET 1 OF 1


2
1
B
A
12.70
2.54
Ø7.00
R5.00
10.00
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN mm
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
PLA
FINISH
DO NOT SCALE DRAWING
NAME
DATE
DRAWN
Dylan C
5/1/24
Aarushi D
Nick T
Comments
M 10 head screw with M6 x1.0 threads
TITLE:
screw
SIZE
A
DWG. NO.
1
REV
1
SCALE: 4:1
WEIGHT:
SHEET 1 OF 1


2
1
B
A
10.00
2.54
UNLESS OTHERWISE SPECIFIED:
DIMENSIONS ARE IN mm
TOLERANCES:
FRACTIONAL ±
ANGULAR: MACH ± BEND ±
TWO PLACE DECIMAL ±
THREE PLACE DECIMAL ±
INTERPRET GEOMETRIC TOLERANCING PER:
MATERIAL
PLA
FINISH
DO NOT SCALE DRAWING
NAME
DATE
DRAWN
Dylan C
5/1/24
Aarushi D
Nick T
Comments
M10 head nut with M6 x1,0 threads
TITLE:
nut
SIZE
A
DWG. NO.
1
REV
1
SCALE: 4:1
WEIGHT:
SHEET 1 OF 1