Dylan Tobia and Gabriel Ruffo

ISTA 303

Dr. Jansen

3 December 2017

Skittle Sorter Write-Up

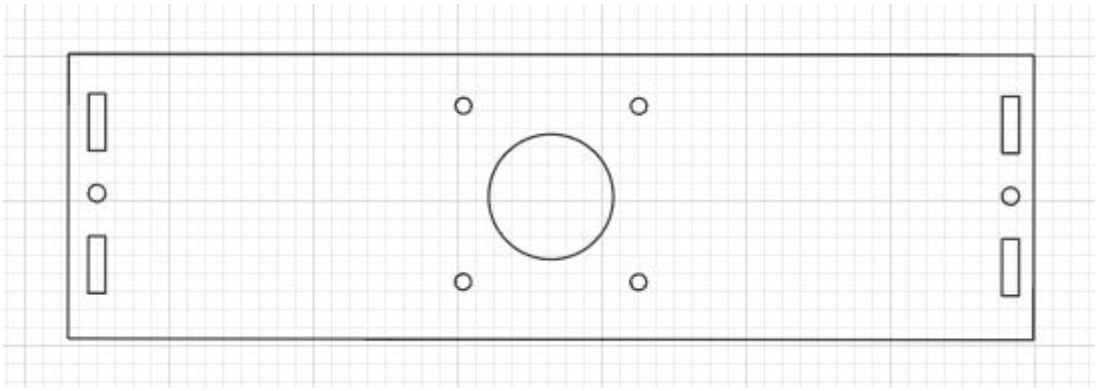
I. Introduction

With last semester's winning design in mind, we decided to model our Skittle sorter as a modification of the rotating hopper design. Instead of grabbing a Skittle and moving it to a fixed location for every Skittle, Dylan designed a bottom plate with a moving arm that has a hole and a covering arm plate. Each Skittle color has a preset location along the arm, with the colors that occur most often closest to the pickup location. When a Skittle is picked up, it is dragged over the color sensor, to which the arm on the bottom plate will move the drop hole to its specified location, with the covering arm to the right to prevent the Skittle from falling prematurely. Not only does this allow for an earlier drop for certain colors, but it also gets rid of the need for a separate Skittle dropping subsystem that guides the Skittles to their prospective buckets. With this alteration, we believe that on average, our Skittle sorter will sort a bag of Skittles at a slightly faster rate than the winning design last semester.

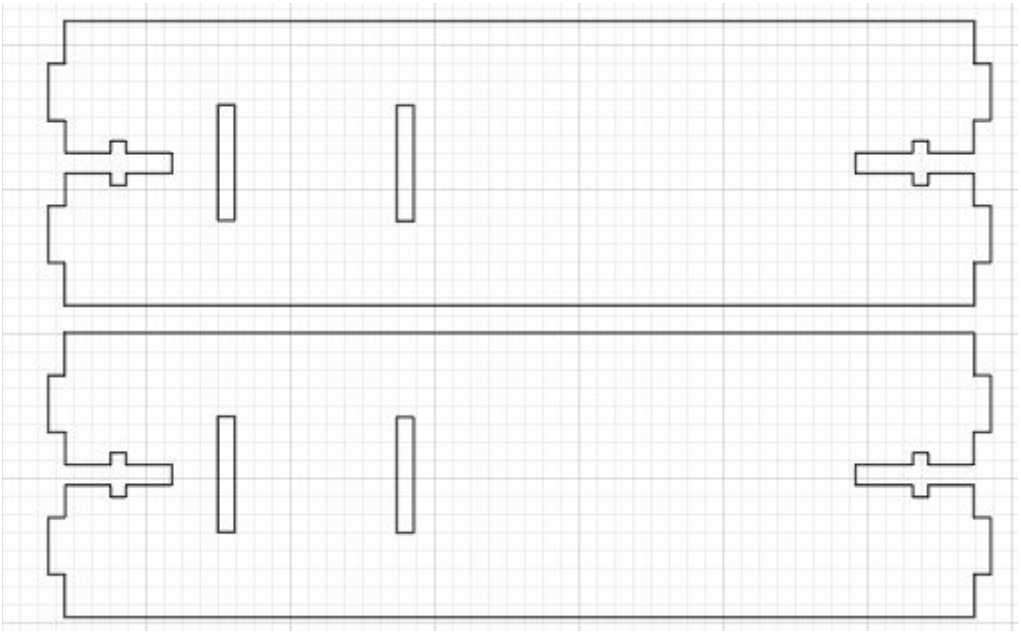
II. Design

(Subsystems with their designs - Feeder, Sensor, Sorter)

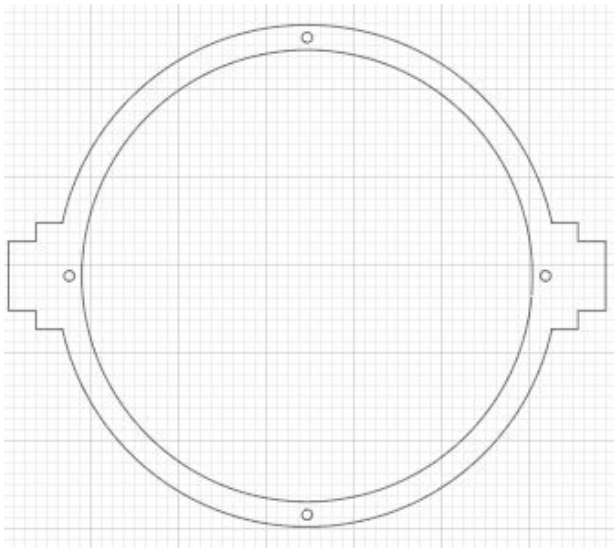
Our Skittle sorter works by a constantly spinning plate grabbing single Skittles at a time, passing them over a color sensor, and moving a bottom plate arm to drop the Skittles in their specified buckets. The top spinning plate is controlled by a stepper motor spinning at around 48 rpm, pausing every 50 steps to allow the Skittle’s color to be read. The stepper motor is held by two plates and two washers above the main compartment that holds the Skittles and rotating plates:

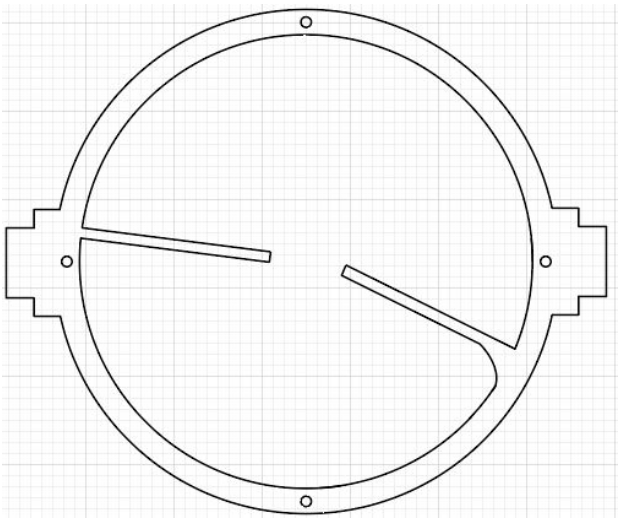


These two plates and the stepper are held up by two arms that also have slots to hold the main compartment as well:

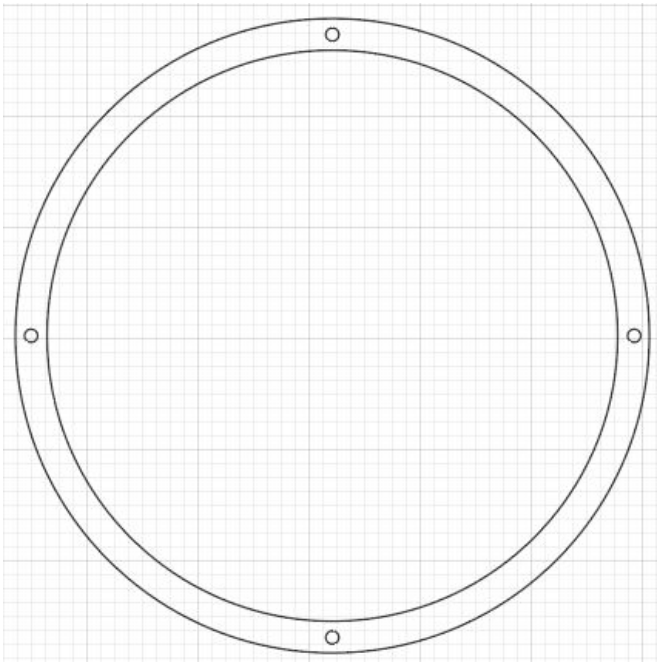


The main compartment is held up by two different kinds of rings: a normal ring with side slots, and an armed ring with side slots:

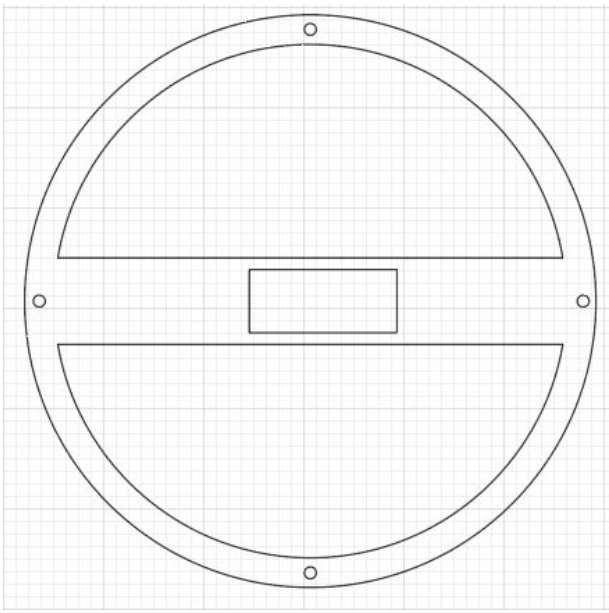




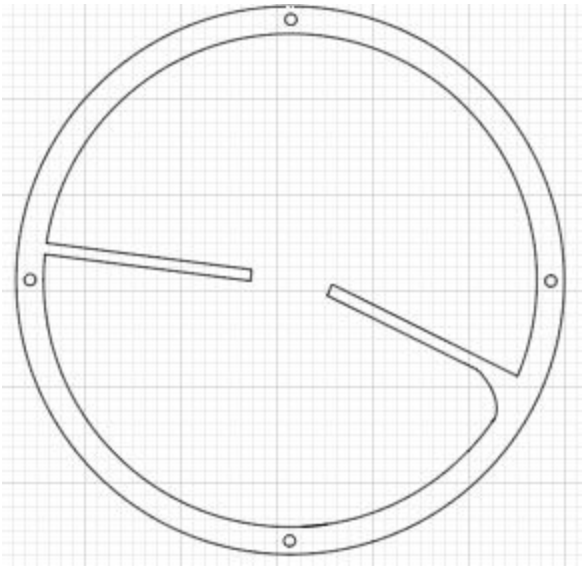
The first ring is placed on the bottom of the entire compartment, and the armed ring helps keep the Skittles in one area, and goes 10 rings above the bottom one. The majority of the rest of the rings are normal rings with no arms:



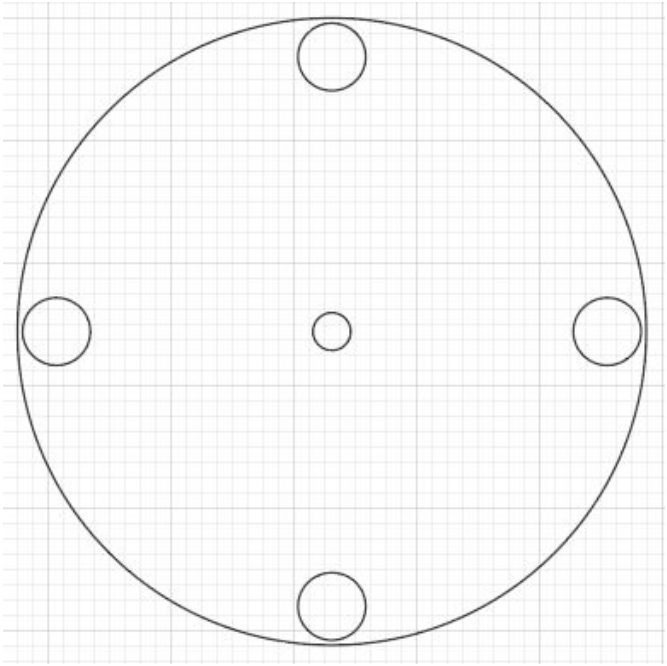
The ring that goes right above the bottom ring with side slots is the ring that holds the servo that controls the arm with the hole that allows the Skittles to drop into their buckets:



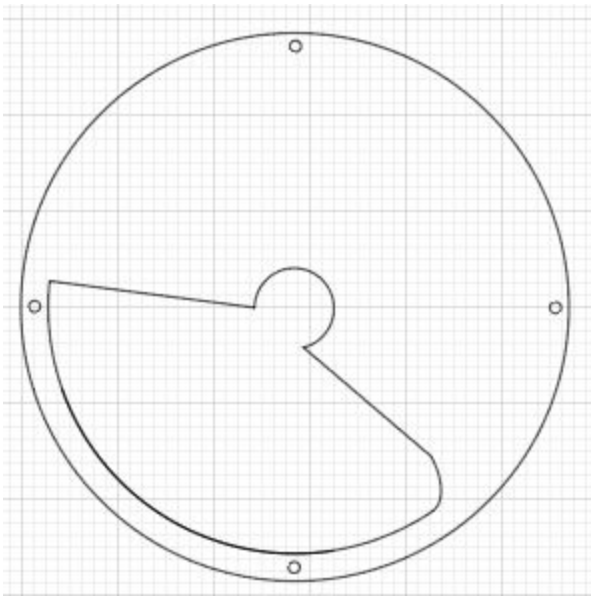
After a number of the normal rings, we placed one armed ring that would keep the Skittles in one area:



After which the main rotating plates which are connected to the stepper motor are inserted:

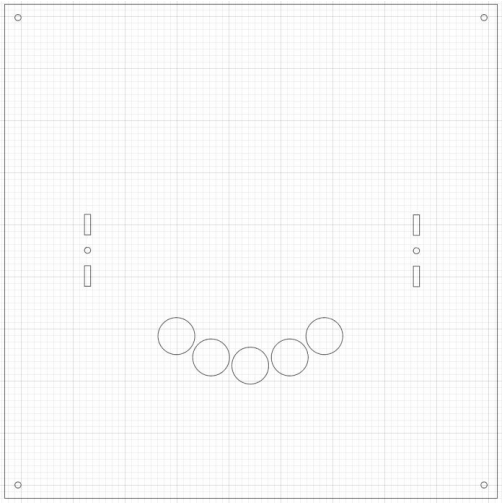


Three of these plates are kept together using the main bolt and a nut. After a few more armed rings, a cover plate ring was placed to keep the color sensor area dark:

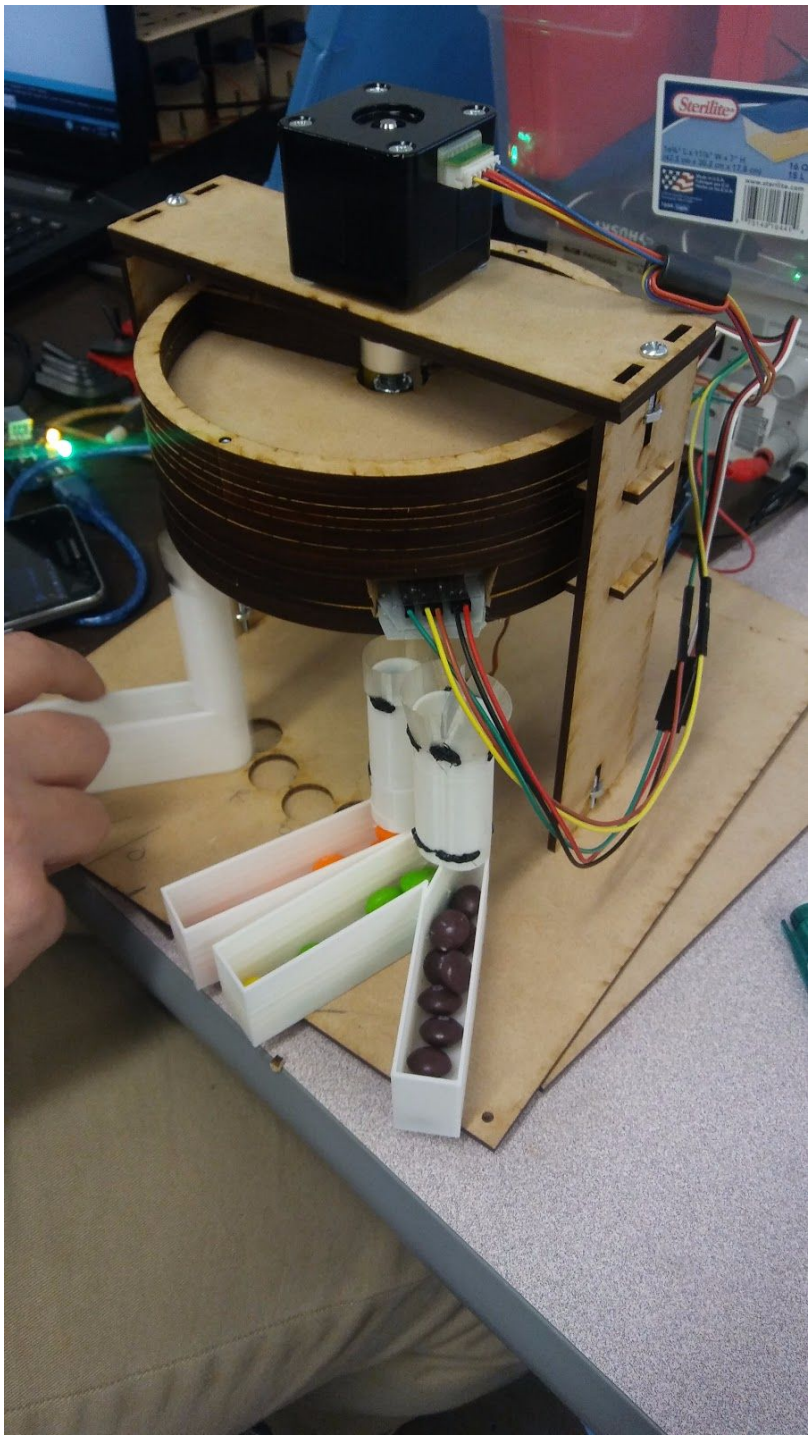


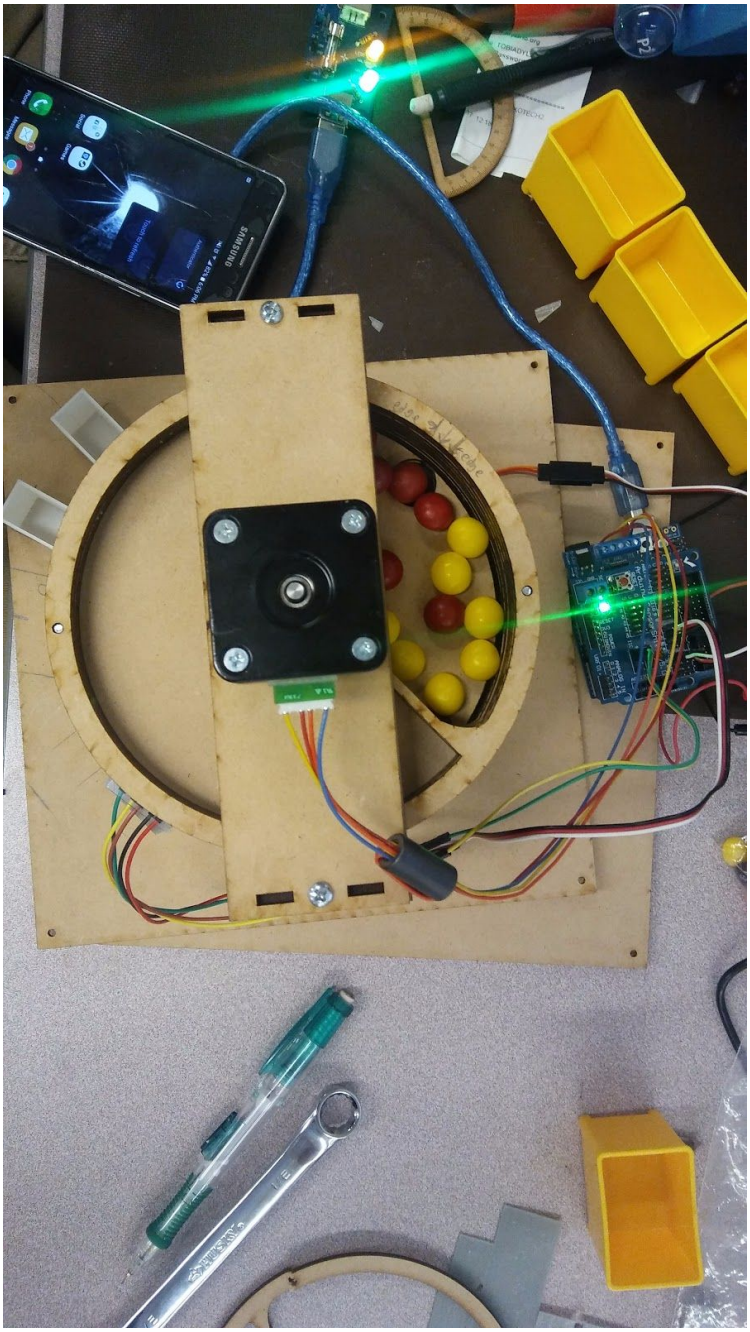
For the color sensor area, we cut out sections from normal rings to allow for easy press-fit insertion of the sensor. The main plate with the holed arm also has a very thin sheet of plastic that is placed over the color sensor area so that the Skittle is not directly on top of the sensor, something that caused a few problems during color detection.

The main compartment is held together by four 2 inch screws, and kept up using the two slotted arms that also hold up the stepper motor. The arms are then mounted to the bottom of a large plate that holds the 5 different buckets for the 5 different colored Skittles:



Final product:

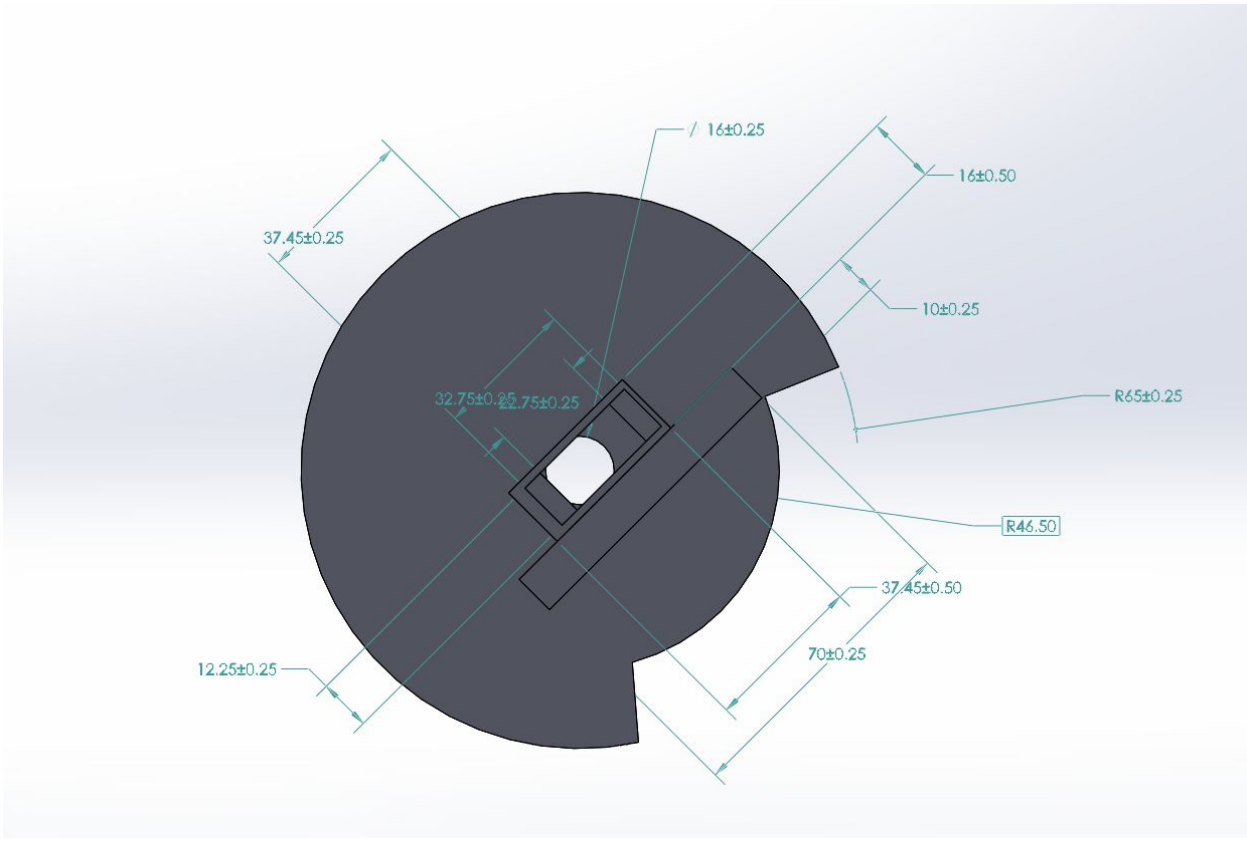


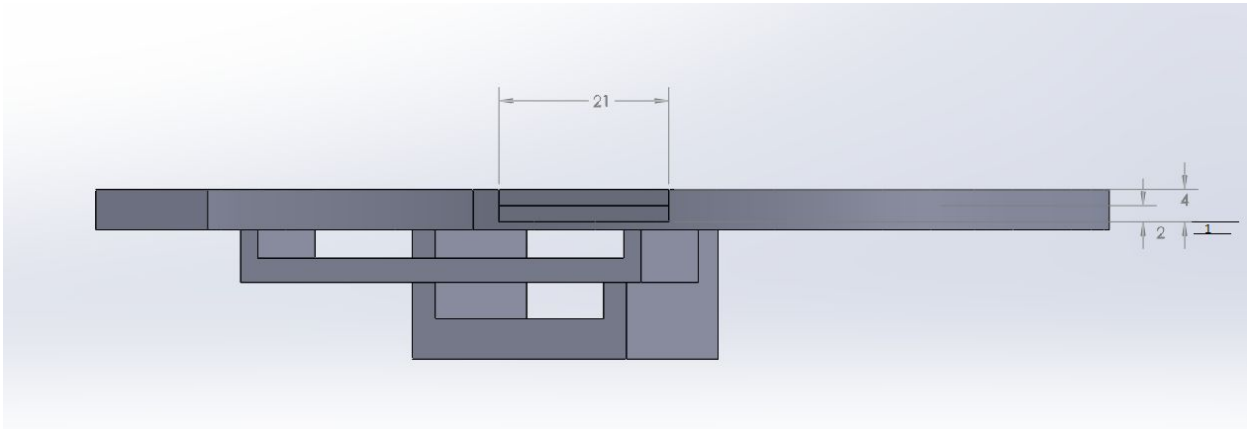


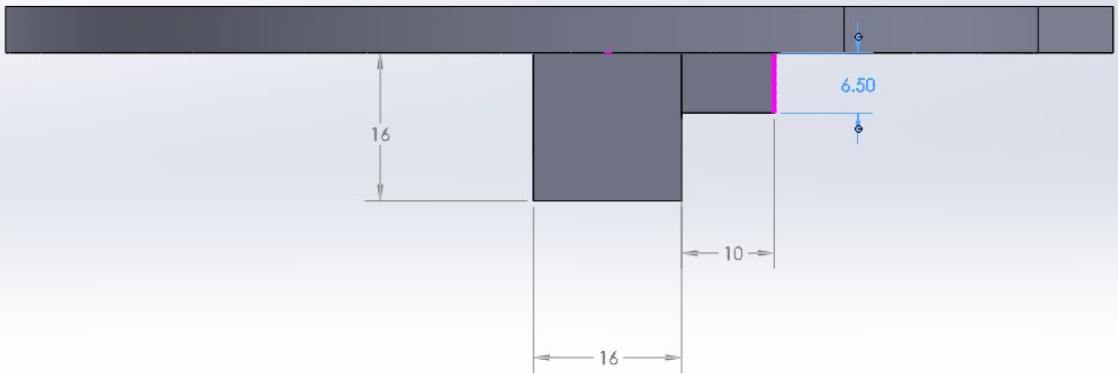
III. Schematics

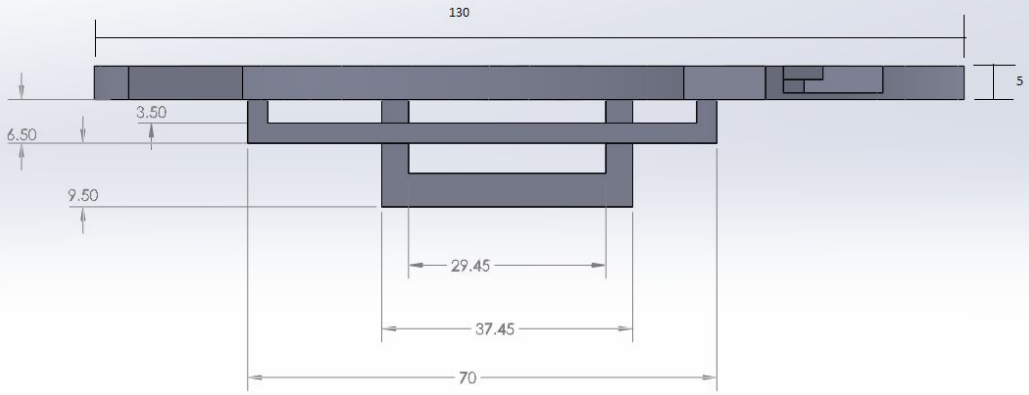
We 3d printed a couple parts: The bottom “plate”, the servo arm, and the buckets for which to catch the skittles. The bottom plate has a press fit mount for the color sensor, as well as the servo. The servo arm is held in place by the servo, and a little plastic shelf. This keeps the servo arm from drooping too much during the sort process. The buckets just catch the skittles.

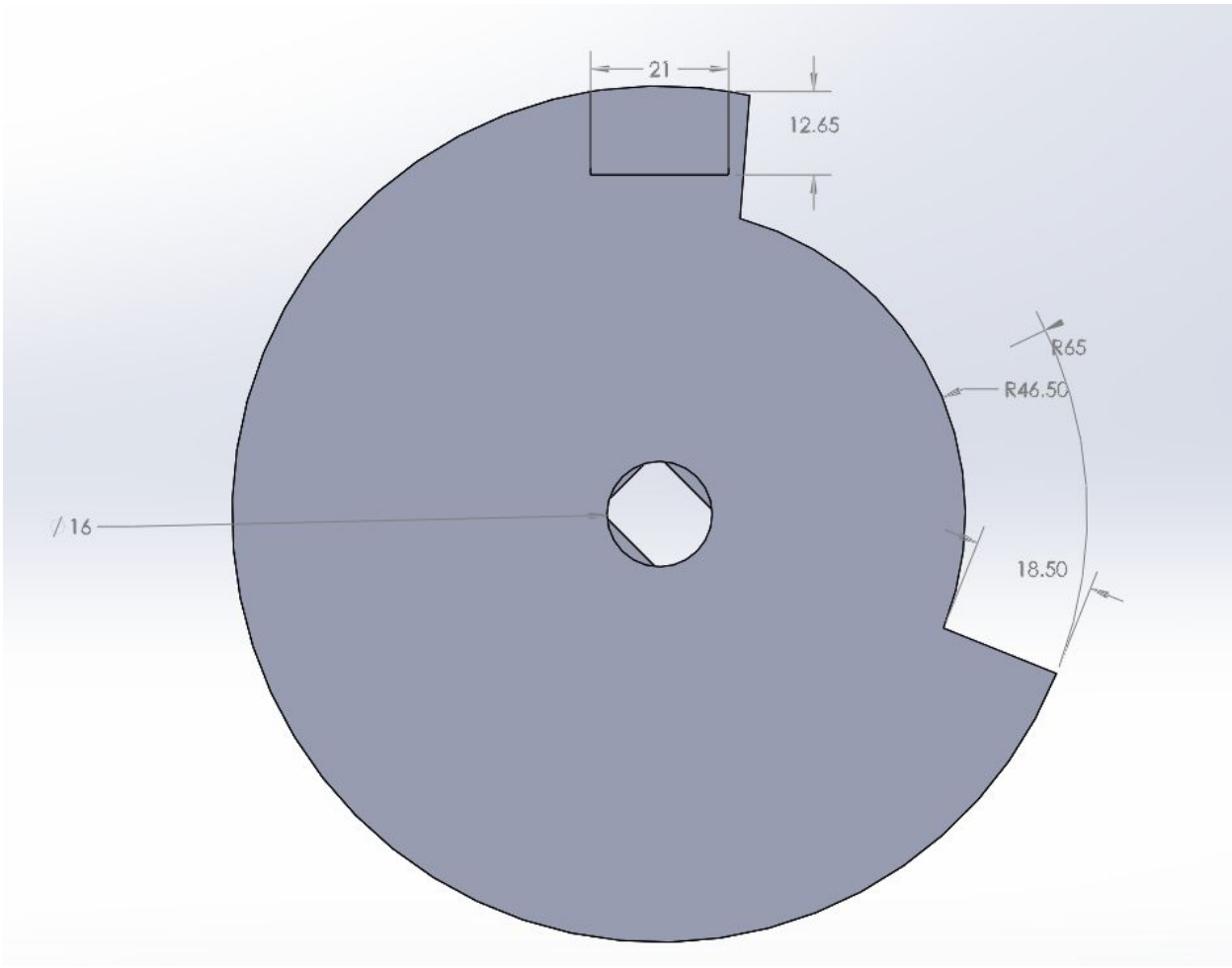
We have two different varieties of buckets, 3 single buckets, and 1 double bucket. The double bucket has two circle shaped holes, that have been combined and split, to handle the amount of distance the servo could move at the end.

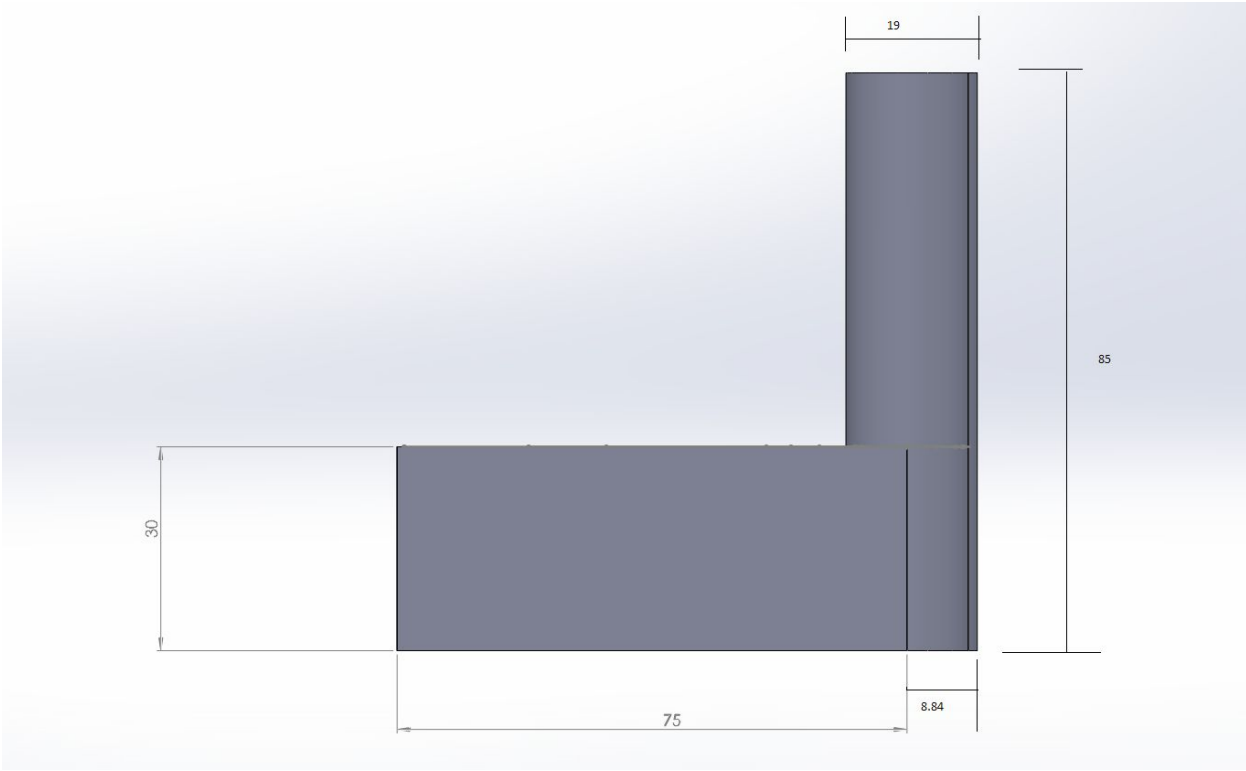
- 
16±0.25
16±0.50
37.45±0.25
10±0.25
32.75±0.25
22.75±0.25
R65±0.25
R46.50
37.45±0.50
70±0.25
12.25±0.25

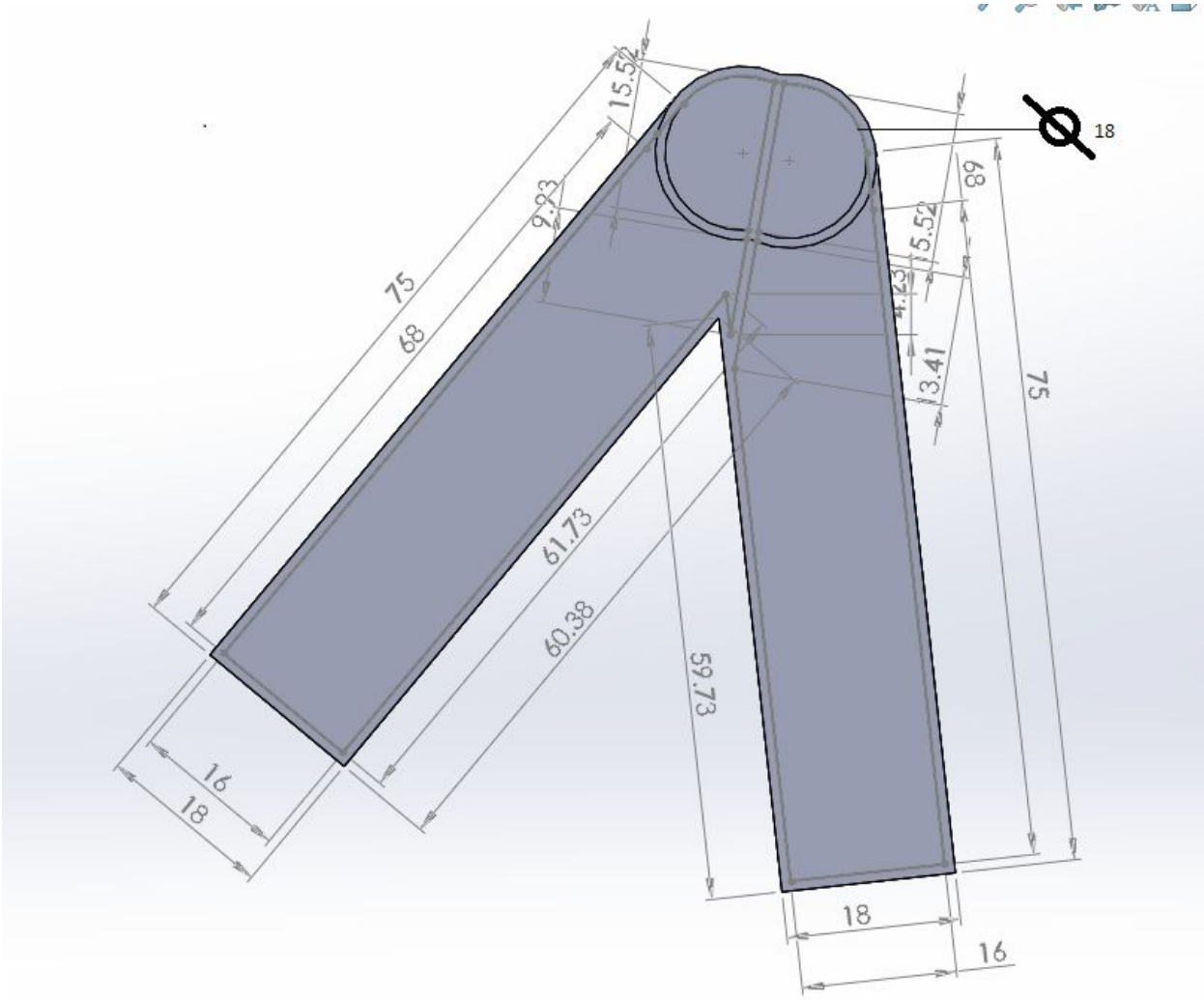

21
4
2
1

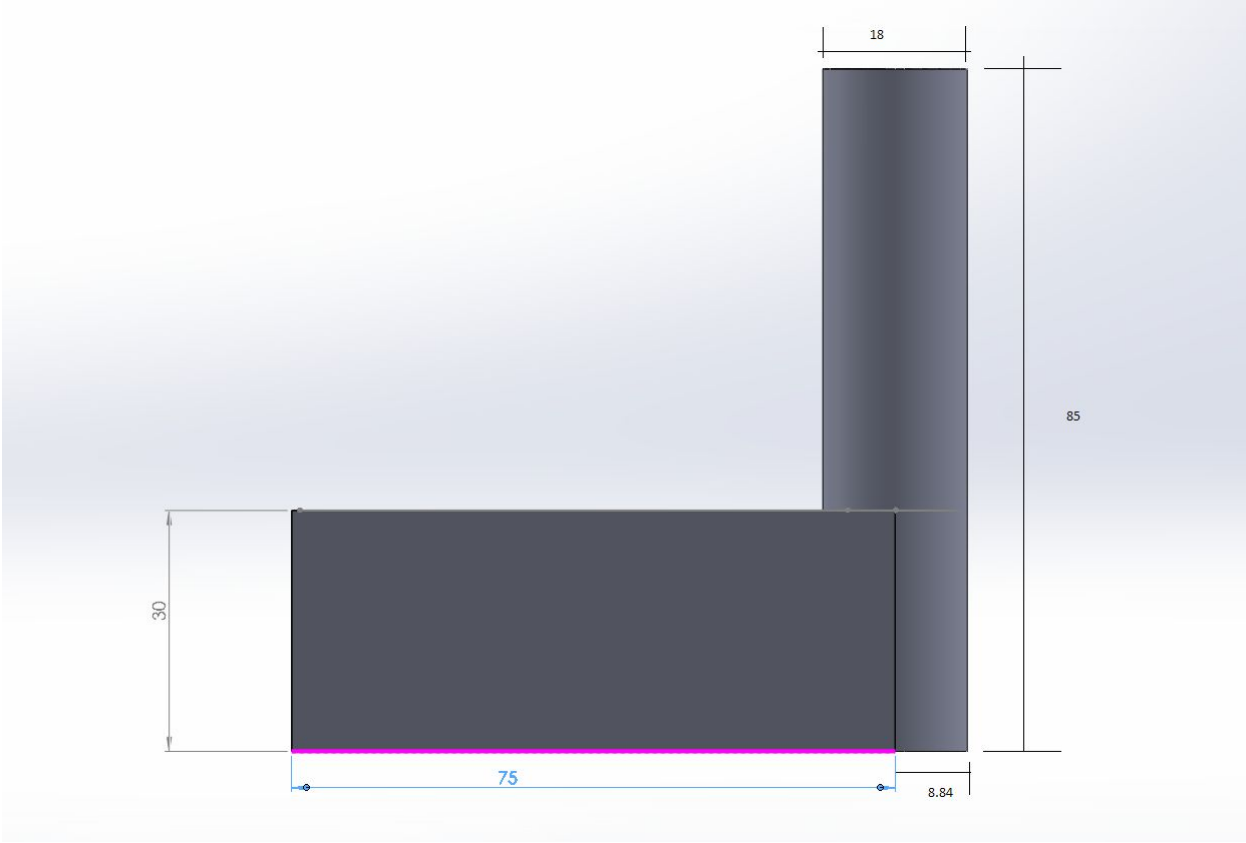

6.50
16
10
16


130
5
3.50
6.50
9.50
29.45
37.45
70

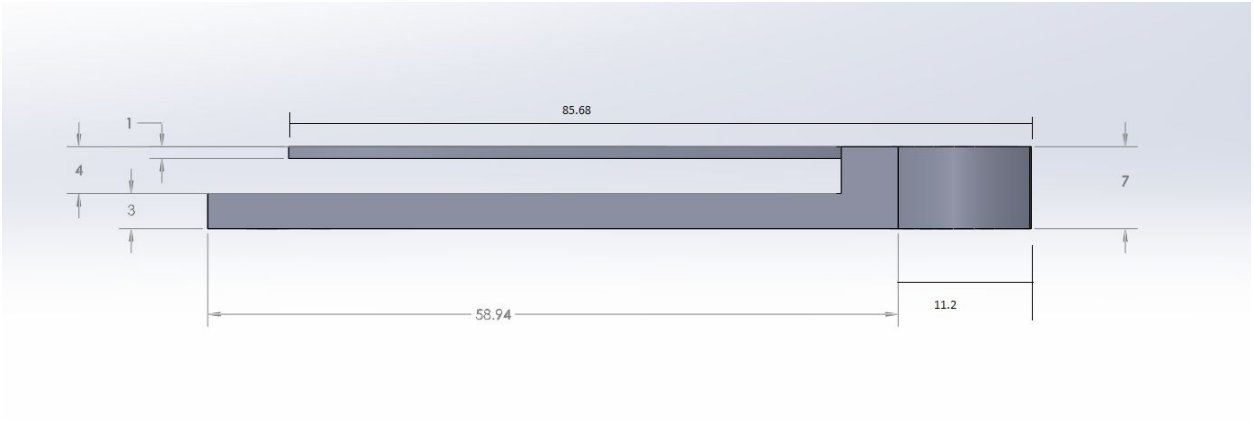

21
12.65
R65
R46.50
/ 16
18.50

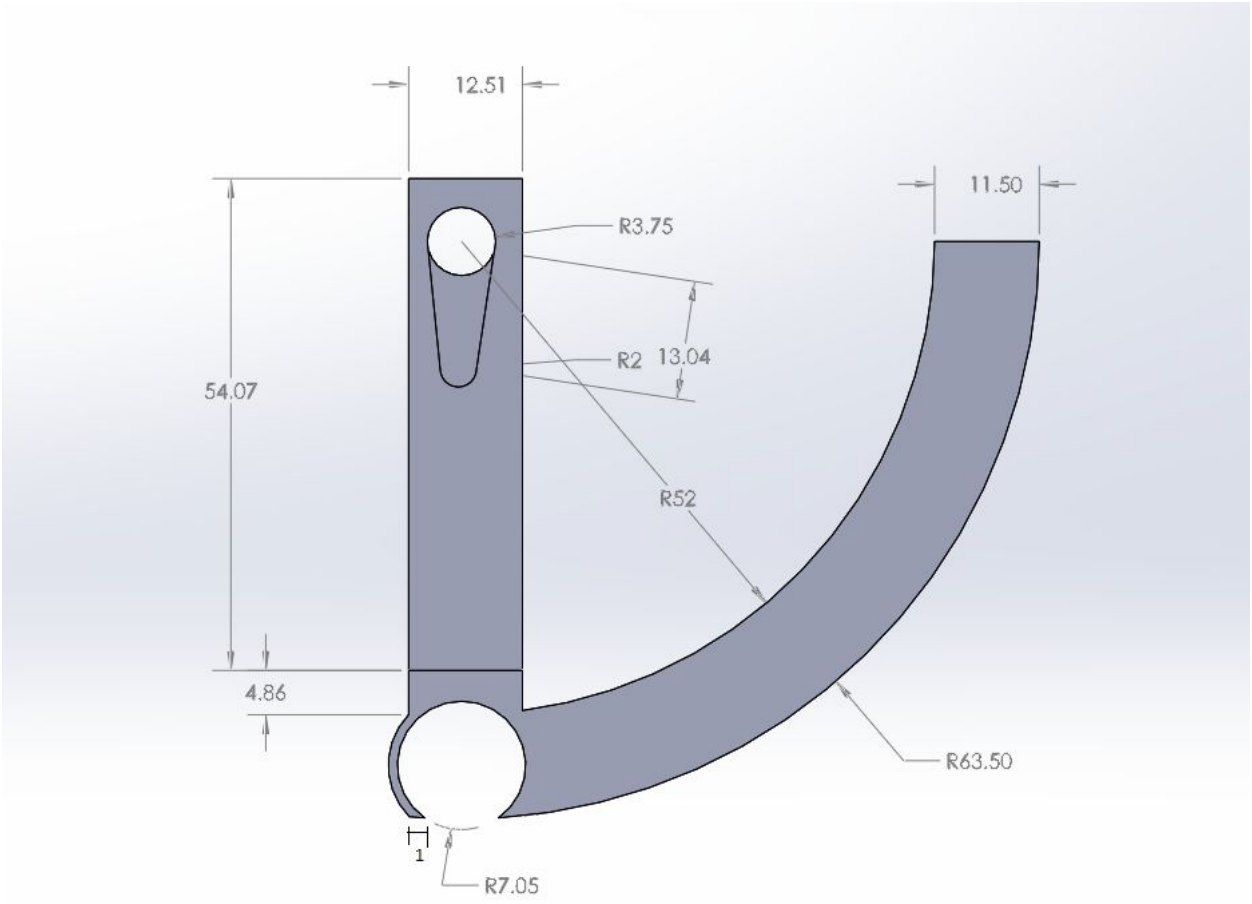

19
85
30
75
8.84


15.52
18
89
9.23
15.52
4.43
3.41
75
68
75
61.73
60.38
59.73
16
18
18
16



The Servo arm itself has a “Flimsy” slide area, so that when the servo moves closer and farther away from the color sensor, it can bend. We wanted to servo arms hole as close to the surface of the bottom plate as possible.





IV. The Code

The code itself is relatively simple. We took the color sensing code that you gave us, and tweaked it very slightly so that it read a skittle 3 times, instead of just once. We then took the

averages of the 3 rgb values given to use from the sensor, and used your original code to do the rest.

The loop function has an inner if statement, which stops the sorter from running until it has been calibrated, and the user types “go” into the serial viewer. After typing “go” the flow of the code is as follows:

1. Take reading of color
2. Move servo to correct position
3. Move stepper 50 steps

It repeats this over and over again until turned off. Very simple. The getColor() function returns an int depending on which color was read. This int is passed to the moveServo(color) function. The moveServo(color) function checks which value was passed to it, and then moves the servo to the angle related to that color.
The stepper is moved by a simple stepper->move(50) command.

At this point, we can accurately (with only a few discrepancies) sort a 60 skittles in 36 seconds. If given more time, we would offset our stepper motor and add gears to it and our wheels drive train, so that we could spin our wheel much faster.