Twister Dorothy Inspired Candy Machine

February 2018

Twister Dorothy Inspired Candy Machine Construction

1 DISCLAIMER

This document and the attached drawings are not to be used for construction.

You are hereby authorized to view, copy, print, and distribute the materials from this document or the attached drawings subject to the following conditions:

- The materials are for your personal informational use only and not for commercial purposes.
- No liability in respect of any of the contents of such publications or use thereof is accepted and no warranties expressed or implied are made in relation thereto.

Disclaimer

ANY INFORMATION CONTAINED IN THIS DOCUMENT OR THE ATTACHED DRAWINGS ARE PROVIDED [AS IS] FOR YOUR INFORMATIONAL PURPOSES ONLY, WITHOUT WARRANTY OF ANY KIND, INCLUDING ANY IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. THE AUTHOR OF THIS DOCUMENT AND ATTACHED DRAWINGS WILL IN NO EVENT BE LIABLE FOR ANY DIRECT, INDIRECT OR PUNITIVE DAMAGE OF ANY KIND THAT RESULTS FROM THE USE OF OR INABILITY TO USE THIS DOCUMENT OR THE ATTACHED DRAWINGS, INCLUDING WITHOUT LIMITATION, LOSS OF PROFITS, GOODWILL, OR BUSINESS INTERRUPTION. THE AUTHOR OFFERS NO ASSURANCES, WARRANTIES OR GUARANTEES AS TO THE VALIDITY OR COMPLETENESS OF THIS INFORMATION. INFORMATION WILL BE CHANGED, UPDATED AND DELETED WITHOUT NOTICE.

1 Overview

The Twister Dorothy inspired candy machine will dispense M&M's or a similar sized candy of your choice. The candy machine was designed to replicate the Dorothy device from the movie Twister. The candy machine uses flashing weather sensors, 3D printed auger for dispensing candy and audio playback.

You will need a 3D printer to create the auger, auger mounting brackets and the sensor discs. The candy machine is Arduino controlled and the electronics utilized for control are fairly simple. I used an off the shelf relay board that interfaces the Arduino to the candy machine components.

2 Components

The candy machine is controlled by an Arduino microcontroller. The Arduino handles the interface to the auger servo motor, JBtek 4 channel DC 5V relay module for Arduino and the dispense pushbutton.

2.1 Arduino Uno R3 Controller

The candy machine utilizes an Arduino Uno R3 controller. I used the Arduino Mega since I had a few spares available. The Arduino Uno R3 is shown below

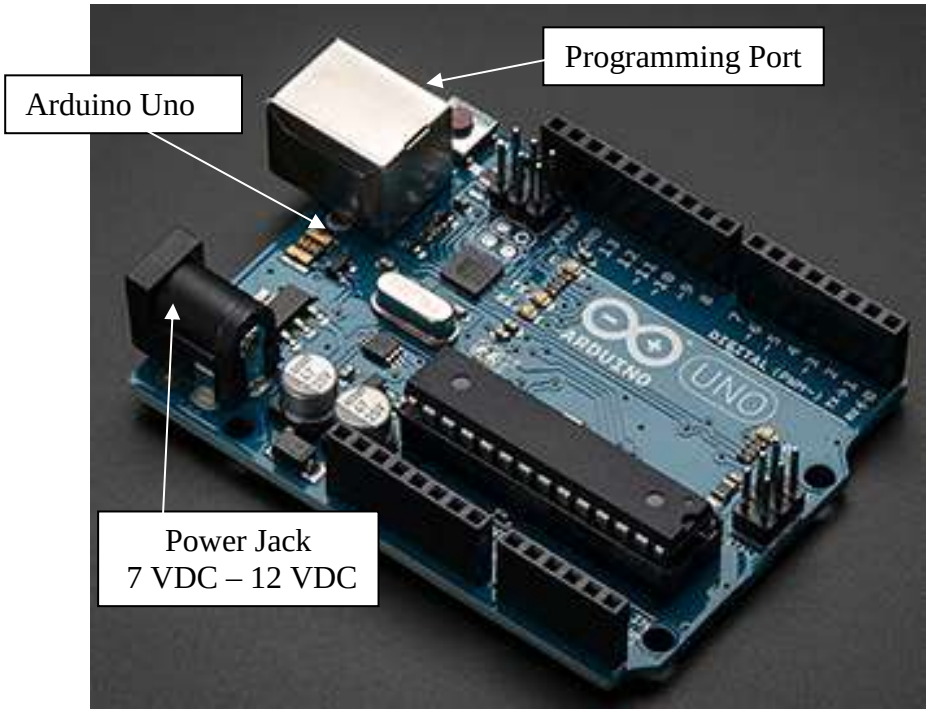


Figure 2.11 – Arduino Uno R3

2.2 Proto-Screwshield R3 Kit for Arduino

I used a Proto-Screwshield for the Arduino controller. The Proto-Screwshield kit was purchased from Adafruit.com and provides screw connections for the Arduino pins. This is handy while developing and provides a rugged connection for I/O pins. The Proto-Screwshield is shown below.

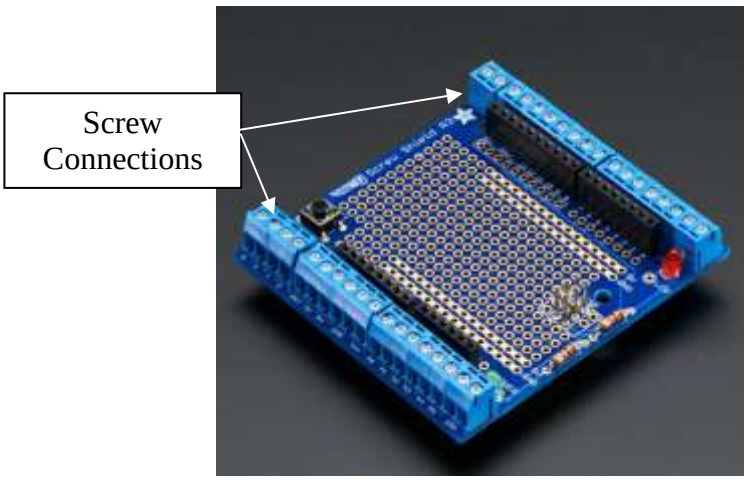


Figure 2.21 – Arduino Proto-Screwshield

2.3 4 Channel Relay Module for Arduino

I used a 4 channel relay module for controlling the anemometer motors, dry contact closure to the MP3 playback board and a means to apply servo power to the candy dispenser servo motor when commanded to dispense. I remove power to the motor when not in use since the life span of the servo motor is increased significantly. The relay module was purchased from amazon.com.

Figure 2.31 – JBtel 4 Channel DC 5V Relay Module

2.4 MP3 Player Board

The candy dispenser utilizes an audio playback board for producing twister themed audio once the dispense pushbutton is pressed. The audio board utilized for the Dorothy audio is a triggered MP3 player board with 10 watt amplifier and terminal block connections. Audio file(s) are stored on a micro SD card.

The MP3 player board was purchased from electronics123.com.
http://www.electronics123.com/shop/product/4-buttons-triggered-mp3-player-board-with-10w-amplifier-and-terminal-blocks-8267

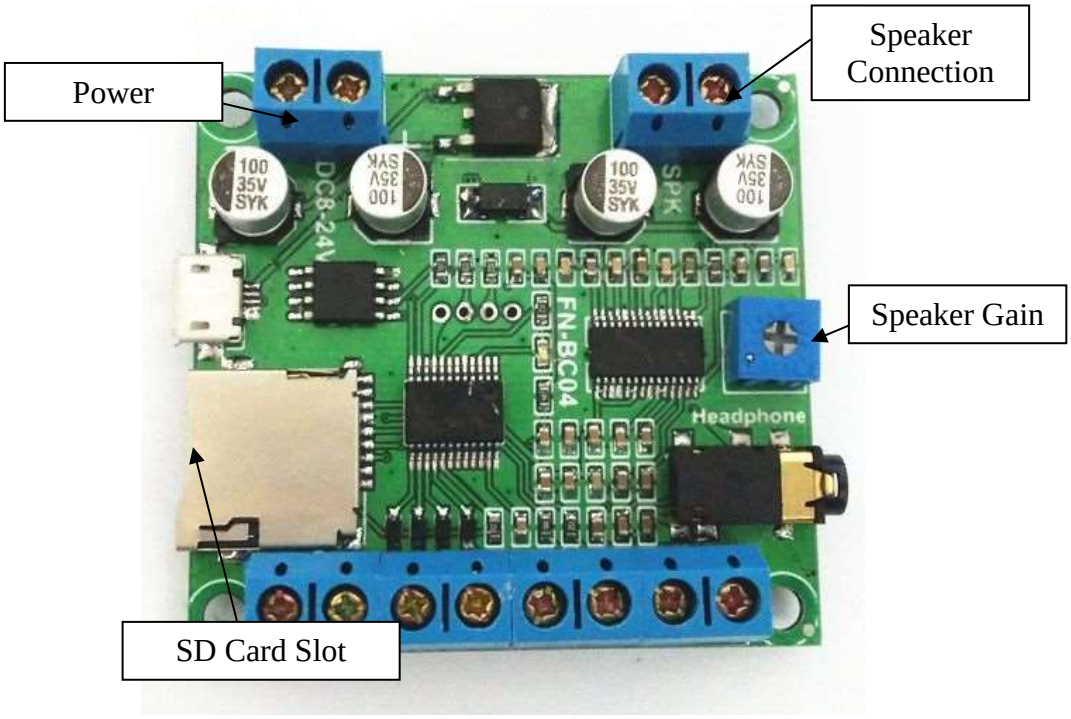


Figure 2.41 – Electrocnics123.com MP3 Player Board

There is 1 audio file stored on the SD card. A small rotary potentiometer is located on the audio playback board to control the level of the audio signal sent to the speaker terminals. In the picture above, this is defined as speaker gain.

Audio Track
I used Audacity to create an audio file that contains Twister themed audio that is played when the dispense pushbutton is pressed.

2.5 Audio Speakers

The candy dispenser utilizes two 5 watt speakers driven by the MP3 player board.

The speaker utilized are Gikfun 2" 8 Ohm 5W Full Range Audio Speaker Stereo Woofer Loudspeaker for Arduino (Pack of 2pcs) EK1688 as shown below.

The speakers were purchased from amazon.com.
https://www.amazon.com/gp/product/B014GM8BP8/ref=oh_aui_detailpage_o05_s00?ie=UTF8&psc=1

Figure 2.51 – Candy Dispenser Speaker

2.6 Candy Dispenser Auger Servo Motor

The candy dispenser utilizes a single HSR-2645CR servo that is a fully proportional high torque digital servo. The HSR-2645CR servo rotates at a specific speed based on a signal from the Arduino microcontroller. The servo speed is fully proportional meaning that the further the signal deviates from the center (1500usec), the faster the servo will rotate. The servo can rotate both clockwise and counterclockwise. The voltage range of the servo is between 4.8 volts to 7.4 volts. The servo is powered via a 12 volt to 6 volt converter. A 24T spline servo block was to increase the servo load –bearing capabilities by helping to isolate the lateral load from the servo spline and case. The servo block also provided a method for mounting the auger to the servo. The servo and servo block was purchased from servocity.com.
https://www.servocity.com/hsr-2645cr-servo
https://www.servocity.com/637118

Figure 2.61 – HSR-2645CR Servo

Figure 2.62 – Standard Hub Shaft ServoBlock™ (24T Spline)

2.7 Anemometer Rotation Motors

The candy dispenser utilizes two 90 RPM micro ear motors to spin the anemometers when the candy is dispensed.

The micro gear motors are controlled by a single channel of the 4 channel relay module. The relay is fed via the 12 VDC power that is fed to the candy dispenser. When the relay energizes, the motors are provided with 12 VDC.

The motors were purchased from servocity.com.
https://www.servocity.com/90-rpm-micro-gear-motor

Figure 2.71 – Anemometer Micro Gear Motor

2.8 Anemometer Motor Hub

Each anemometer utilizes a 3mm bore hub attached to the motor shaft and the anemometer center section so that it can rotate.

The hubs were purchased from servocity.com.

https://www.servocity.com/lightweight-set-screw-hub-3mm-bore

Figure 2.81 – Lightweight Set Screw Hub (3mm Bore)

2.9 Pushbutton

The candy dispenser utilizes one 16mm momentary illuminated SPST pushbutton that initiates the candy dispense process. The pushbutton and quick-connect wires were purchased from adafruit.com and are shown below.

https://www.adafruit.com/products/1439
https://www.adafruit.com/products/1152

Figure 2.91 – 16 mm Pushbutton

Figure 2.92 – Pushbutton Arcade/Button Quick-Connect Wire Pair

2.10 Sensors

The Dorothy sensors utilize Christmas tree clear plastic ornament balls purchased on eBay. The ornament balls are the perfect size for the Dorothy sensors. The entire candy dispenser will require 42 sensor balls.

The clear ornament balls were purchased from eBay.com.
http://www.ebay.com/itm/141806428734?_trksid=p2060353.m2749.l2649&var=440908640692&ssPageName=STRK%3AMEBIDX%3AIT

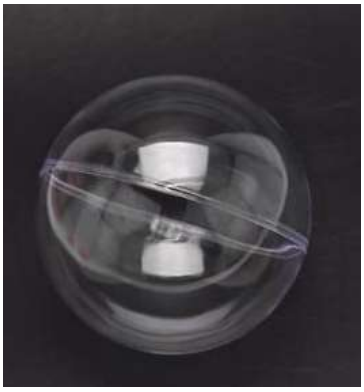


Figure 2.101 – Christmas tree clear plastic ornament balls

2.11 Sensor LEDs

The Dorothy sensors utilize flashing red and green LEDs. The following items were purchased from amazon.com.

50pcs Red Self Flash Flashing Blink Water Clear Bright LED Emitting Diode Light
http://www.amazon.com/gp/product/B01275BWKM?psc=1&redirect=true&ref_=oh_aui_detailpage_o01_s00

100pcs Green Self Flash Flashing Blink Water Clear Bright LED Emitting Diode
http://www.amazon.com/gp/product/B012730TBM?psc=1&redirect=true&ref_=oh_aui_detailpage_o01_s00

Figure 2.111 – Christmas tree clear plastic ornament balls

2.12 12 VDC to 6 VDC Converter

The Dorothy candy dispenser has a single 12 VDC power feed. The servo requires a 6VDC power source. The relay board is also powered from the 6 VDC converter. The power converter used is a DROK® Waterproof High Efficiency Car Power Converter DC12V to 6V/3A Step Down Voltage Module 9-22V to 6V Power Supply.

The power converter was purchased from amazon.com.

http://www.amazon.com/gp/product/B00CGQRIFG?psc=1&redirect=true&ref_=oh_aui_detailpage_o08_s00

Figure 2.121 – Power Converter

2.13 12 VDC Power Supply Adapter

The Dorothy candy dispenser is powered via the 12 VDC power supply adapter shown below.

The power converter was purchased from amazon.com.

http://www.amazon.com/gp/product/B015FKKGBC?psc=1&redirect=true&ref_=oh_aui_detailpage_o08_s00

Figure 2.131 – 12 VDC Power Supply Adapter

2.14 Terminal Block

A terminal block was used to make wire connections. The terminal block shown below is used.

The terminal block was purchased from amazon.com and are packaged as two pieces.

https://www.amazon.com/gp/product/B0050MODRE/ref=oh_aui_search_detailpage?ie=UTF8&psc=1

Figure 2.141 – Terminal Block

2.15 Wire

The wire I used was purchased from servocity.com. The wire is a twisted pair and is purchased in 10' lengths. I use this wire a lot and keep several feet on hand.

https://www.servocity.com/22-awg-2-conductor-twisted-wire

Figure 2.151 – Wire

2.16 Servo Extension

The candy dispenser augur servo motor is connected to a servo extension wire. The extension can be purchased in various lengths.

https://www.servocity.com/hitec-super-duty-extensions

Figure 2.161 – Servo Extension

2.17 Liquid Tape

The liquid electrical tape I used is shown below. This can be purchased at Lowes.

Figure 2.171 – Liquid Electrical Tape

2.18 Dorothy Five Gallon Bucket

The Dorothy candy dispenser was developed using a five gallon bucket as the starting point for the design. The bucket has an ideal shape and size as the platform for the candy dispenser.

The bucket used was purchased from amazon.com and is a United Solutions PN0020 White Five Gallon Plastic Industrial Pail - 5 Gallon Plastic Bucket for Industrial in White.

http://www.amazon.com/gp/product/B005SB1ORY?psc=1&redirect=true&ref_=oh_aui_detailpage_o04_s00

Figure 2.181 – Five Gallon Bucket

2.19 Dorothy Candy Feed Container

The Dorothy candy dispenser utilizes the food grade container with lid shown below to hold the candy that will be dispensed. The size of the container allows the feed mechanism to be attached and fit within Dorothy.

The Clear Food Grade PET Plastic Square Grip Storage Jar w/ Cap - 32 Fluid Ounces (3-4 Cup Storage Capacity) by Pride of India was purchased from amazon.com.

http://www.amazon.com/gp/product/B01A01TH8E?psc=1&redirect=true&ref_=oh_aui_detailpage_o08_s00

Figure 2.191 – Clear Food Grade PET Plastic Square Grip Storage Jar

2.20 Dorothy Sensor Plate

The Dorothy sensor plate sits 3 inches below the top of the bucket and holds all of the Dorothy sensors.

A 10 ½ inch diameter clear cake circle was used and purchased from tapplastics.com.

http://www.tapplastics.com/product/plastics/cake_circles/cake_circle_acrylic/602

Figure 2.201 – 10 ½ inch diameter clear cake circle

2.21 Dorothy Lid

The Dorothy lid covers the sensors and sits on the top of the Dorothy bucket.

A 12 inch diameter .118 in thick acrylic circle was used and purchased from tapplastics.com.

http://www.tapplastics.com/product/plastics/cake_circles/clear_acrylic_circles/140

Figure 2.211 – 12 inch diameter acrylic circle

2.22 Dispenser 1-1/2" Clear Schedule 40 PVC Tee

The Dispenser utilizes a clear PVC Tee. A normal Tee can be substituted at a reduced cost. I chose the clear Tee since I was designing as I went and wanted to visually see how things would interact. I purchased the Tee from usplastic.com.

http://www.usplastic.com/catalog/item.aspx?itemid=32554&catid=592

Figure 2.221 –-1/2" Clear Schedule 40 PVC Tee

2.23 Filament

The clock utilizes HIPS black filament for all of the 3D printed parts except the sensor discs were printed with Peak Green filament.

The Black and Peak Green filament was purchased at Lulzbot.com https://www.lulzbot.com/products/hips-3mm-filament-1kg-reel-esun .

The pictures below show the filament colors.

Figure 2.231 – Black

Figure 2.232 – Peak Green

3 Dorothy Sensor Construction

Once the 3D parts are printed and the components have been gathered, it is time to start the build process. The following steps outline the assembly process.

3.1 Dorothy Sensor 3D Printing

There are 42sensors that are used for the Dorothy candy dispenser. The first thing to do is print the sensor discs as shown below in the Peak Green color. You will need 42 discs.

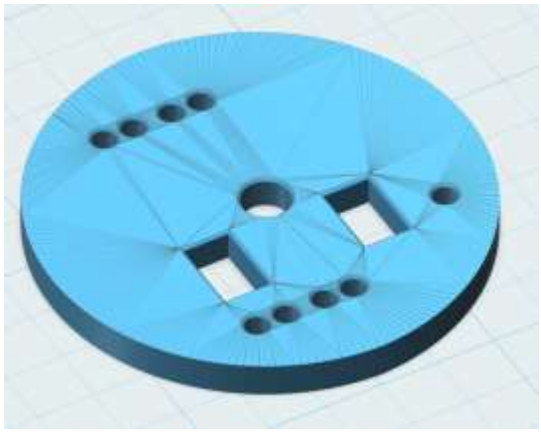


Figure 3.11 – Sensor Disc 3D Rendering

Figure 3.12 – Printed Sensor Disc

The next thing to do is to print the battery holders and battery in the Black color. You will need 84 battery holders and 42 batteries.

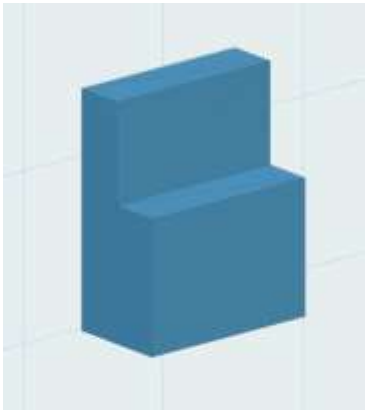


Figure 3.13 – Battery Holder Rendering

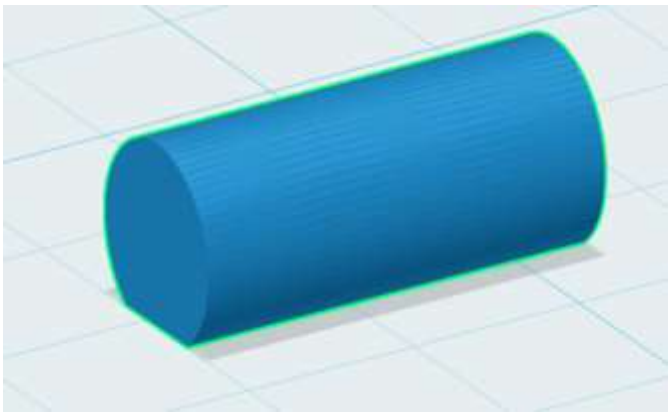


Figure 3.14 – Battery Rendering

3.2 Dorothy 3D Printed Sensor Assembly

The sensor disc, battery holders and battery are glued together using SCIGRIP Weld-On 3 cement. The cement is a high strength water thin acrylic cement that dries quickly. I used the SCIGRIP Weld-On 3 cement from Tapplastics.com
http://www.tapplastics.com/product/repair_products/plastic_adhesives/weld_on_3_cement/131

Figure 3.21 – Acrylic Cement

The cement is applied with a Syringe Hypodermic Applicator as shown below. I used the syringe from Tapplastics.com (SY20-65)
http://www.tapplastics.com/product/repair_products/plastic_adhesives/hypo_type_solvent_cement_applicator/409

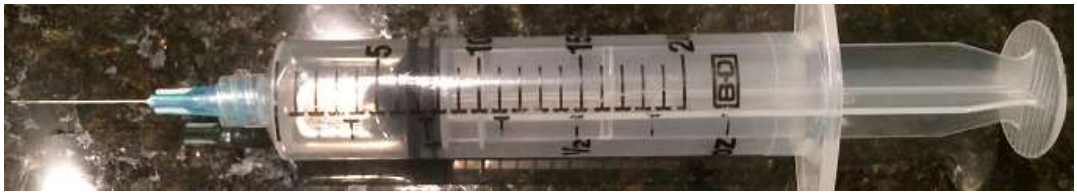


Figure 3.22 – Cement Applicator Syringe

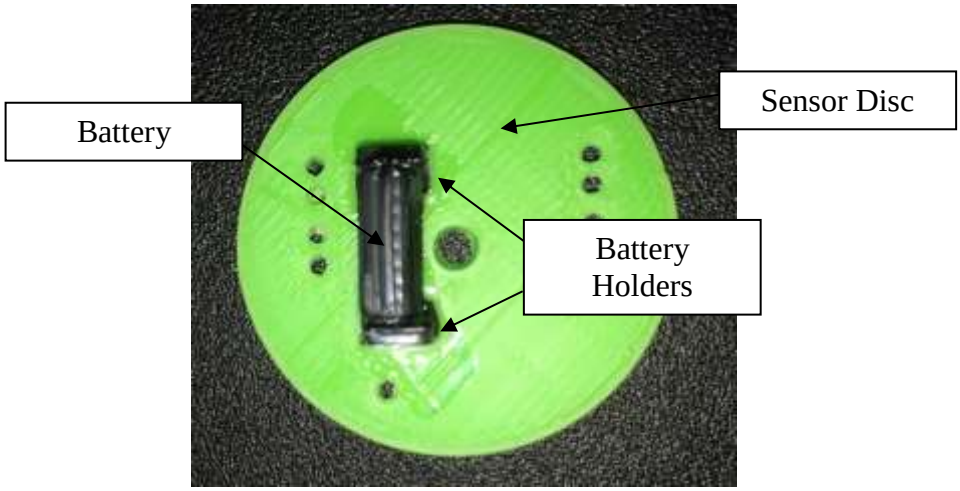


Figure 3.23 – Sensor Assembly

3.3 Dorothy 3D Printed Sensor Hole Cleanup

The sensor disc holes should be drilled out to clean up any excess from the 3D printing process. Use a 1/8”drill bit to drill the holes as shown below.

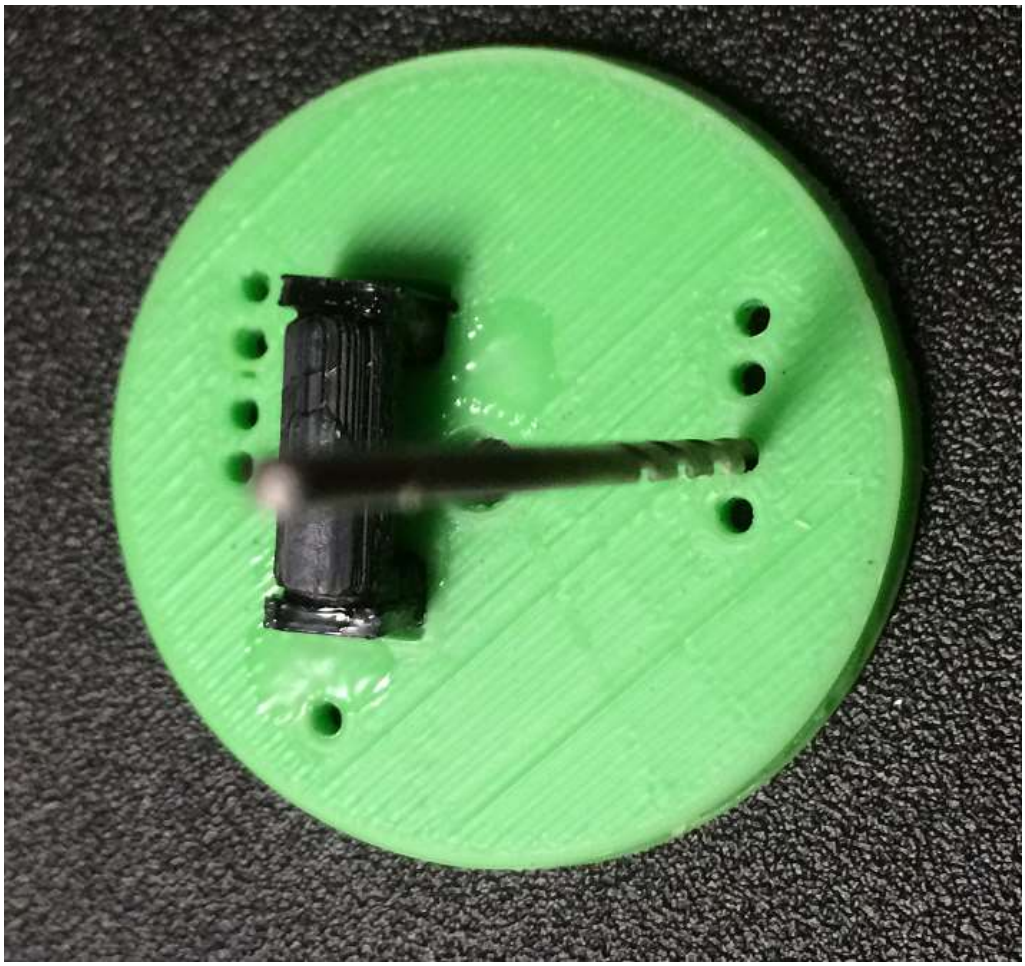


Figure 3.31 – Sensor Disc Hole Cleanup

3.4 Print Sensor Decals

Each sensor utilizes two decals. One decal is the Flufftronics texts and the other decal covers the printed battery to resemble an actual battery. The attached PDF contains the artwork for the sensor decals. You can just print this normally on 8.5" x 11" decal paper.

Sensor Decals

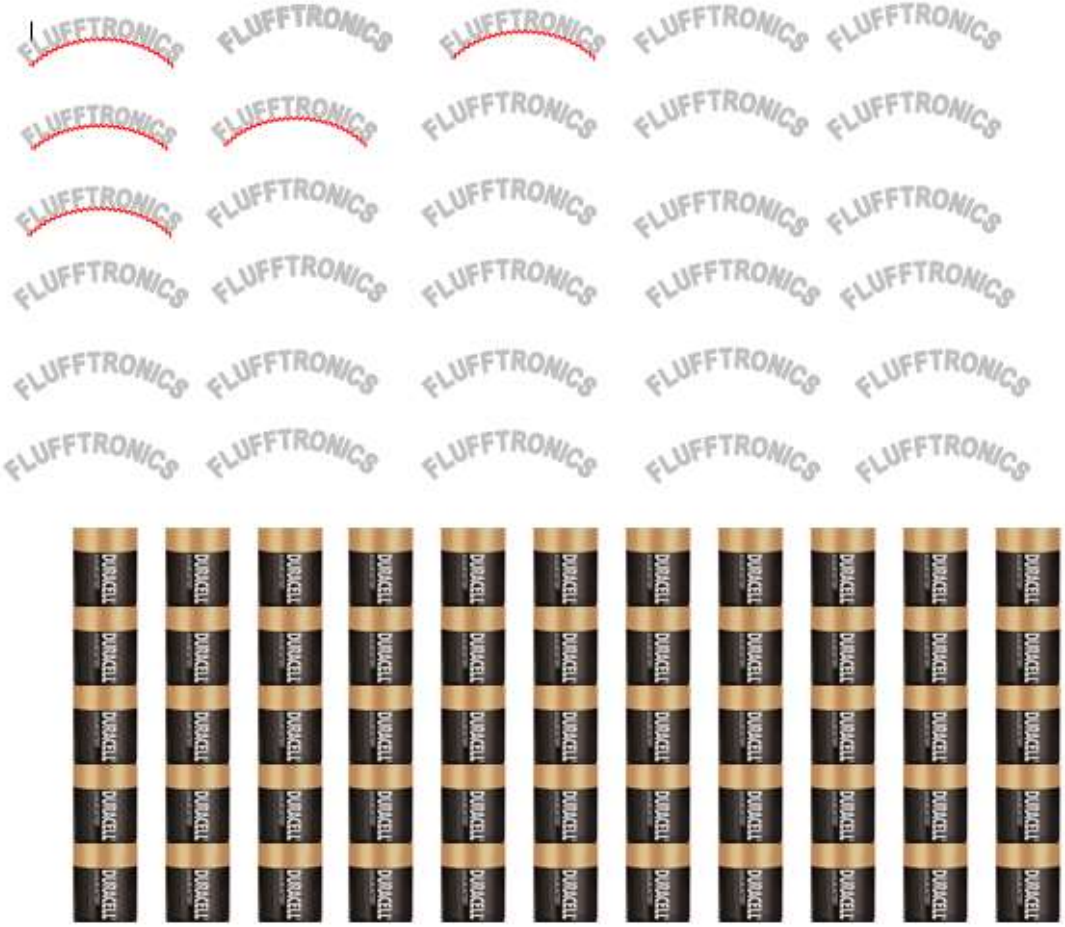


Figure 3.41 – Sensor Decals

Cut the decals out with scissors and place the decal into water and after 45 seconds remove the decal and slide the clear portion of the decal sensor disc or sensor battery. Adjust as required and used a damp paper towel to press the decal firmly on the sensor component.

Once the sensor has dried for a few hours. I sprayed lacquer on the front and side of each decal. The lacquer can be purchased at Lowes or Home Depot. The resultant sensor should appear as follows:

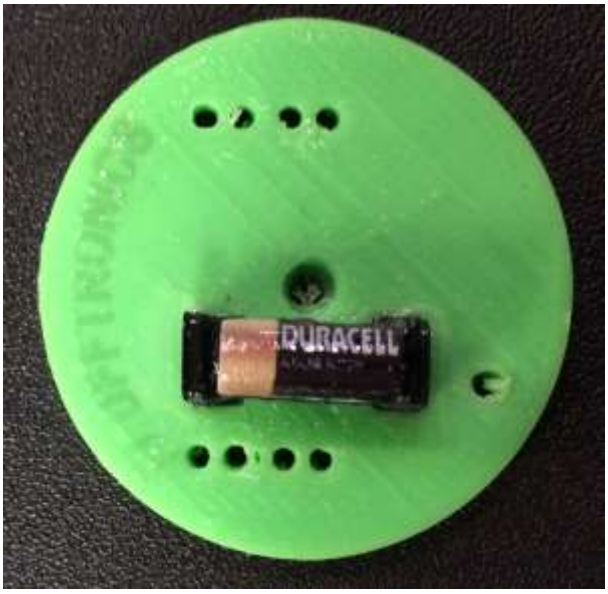


Figure 3.42 – Sensor with Decals

Figure 3.43 – Spray Lacquer

Lightly spray the decal covered sensor using multiple coats to avoid having the decal get ruined. Once the lacquer has dried you may need to run the drill bot through the holes one more time to clear any excess lacquer.

3.5 Attach Sensor Electronics

Each sensor utilizes two 1000 ohm resistors, a red flashing LED and a green flashing LED. Insert the components into the sensor disc as shown below. As shown the longer leg of the LED should be oriented to the left (closest to the resistor).

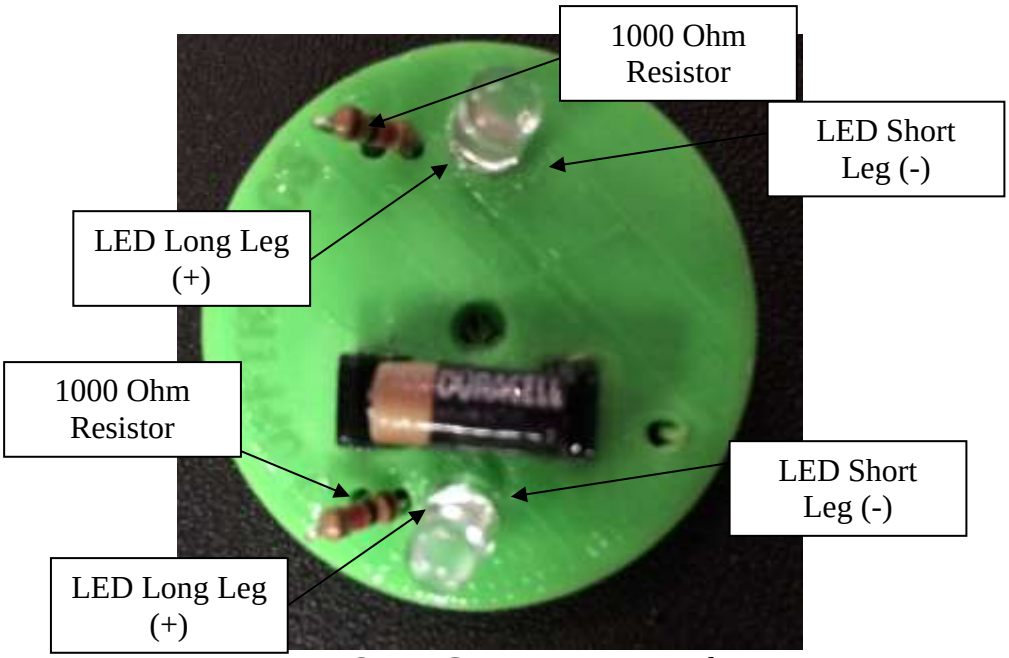


Figure 3.51 – Components Inserted

Flip the sensor over and solder all connections as shown below.

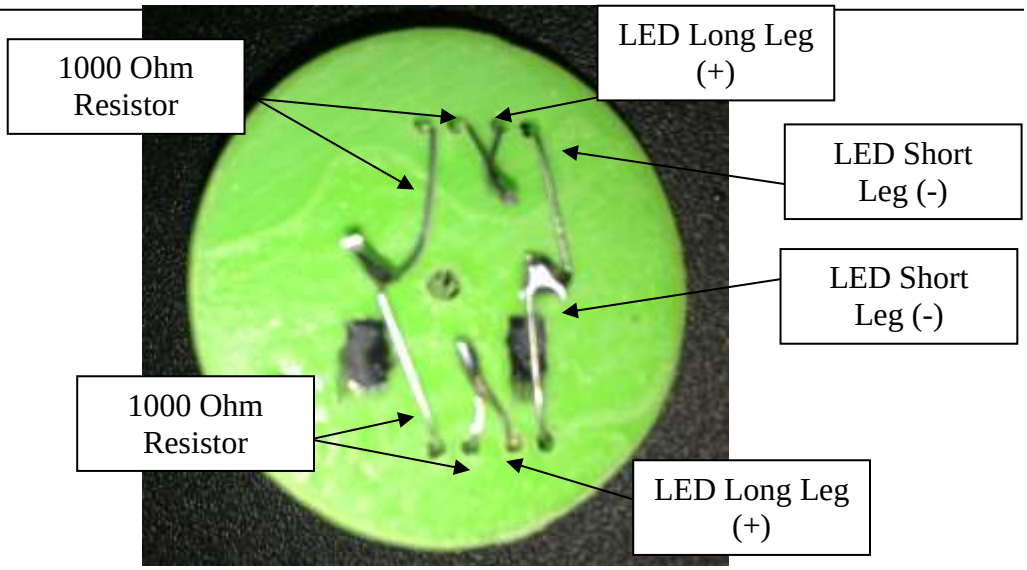


Figure 3.52 – Solder Connections

Once the sensor wires are soldered you can check if it operates by applying +12 VDC to the locations shown below. As a test, you can just use a 9 volt battery.

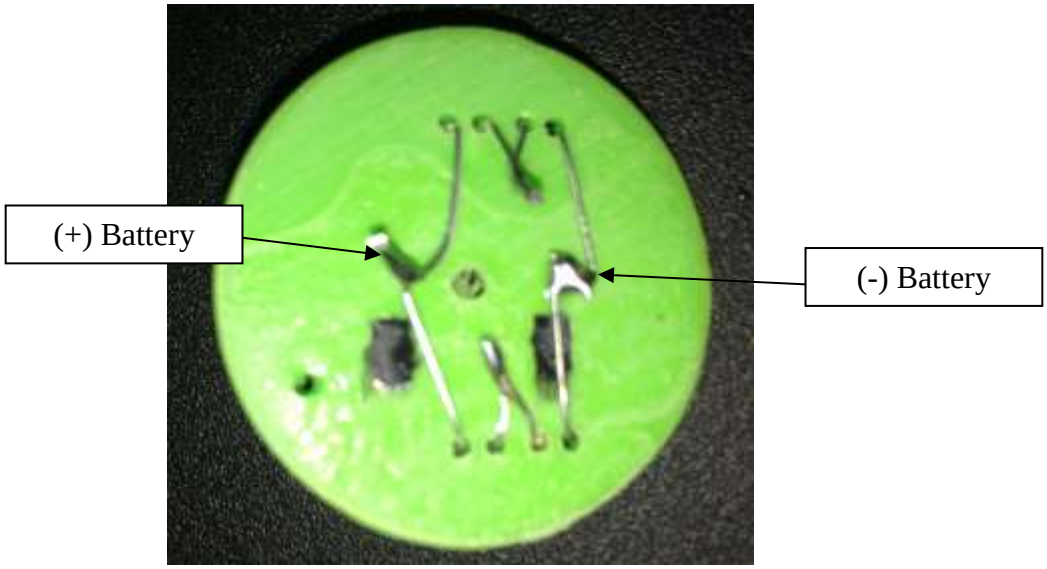


Figure 3.53 – Test Sensor

3.6 Prepare Sensor Globes

As you can see from the picture below, the top portion of the sensor dome will need two holes drilled into it. The smaller hole is for the antenna wire (side) and the larger hole is for the mounting screw (top).

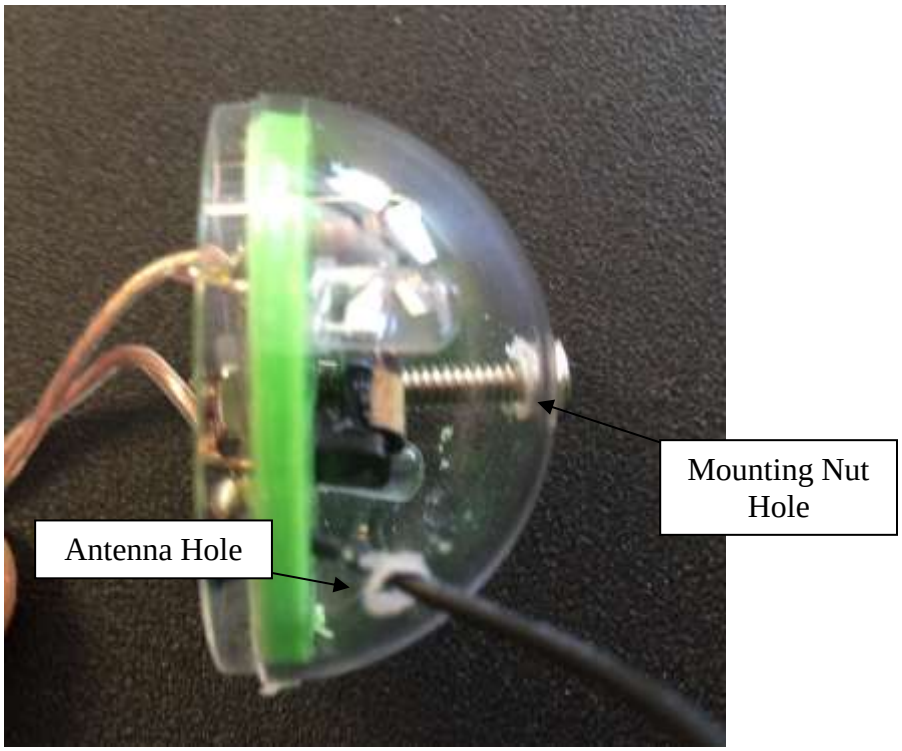


Figure 3.61 – Top Sensor Mounting Holes

The mounting hardware that was used to attach the sensor disc to the top mounting globe is shown below. The hardware was purchased at Lowes. Drill a hole in the top that will accommodate the 6/32" inch screw and a 1/8" hole for the antenna on the side of the sensor globe as shown above.

Figure 3.62 – Sensor Mounting Hardware

The bottom globe requires a hole to allow the sensor power wiring to pass through. This can be a ¼" hole that is drill in the center of the bottom globe. Once the hole is drilled, paint the bottom globe with chrome spray paint.

Figure 3.63 – Paint

Figure 3.63 – Painted Lower Globes

Let the globes sit for about a week before handling them after you paint. The chrome paint takes a long time to cure

3.7 Complete Sensor Assembly

Cut a 6" piece of black wire and run it through the hole shown below. Tie a knot on the underside of the sensor disc where the solder connections are made.

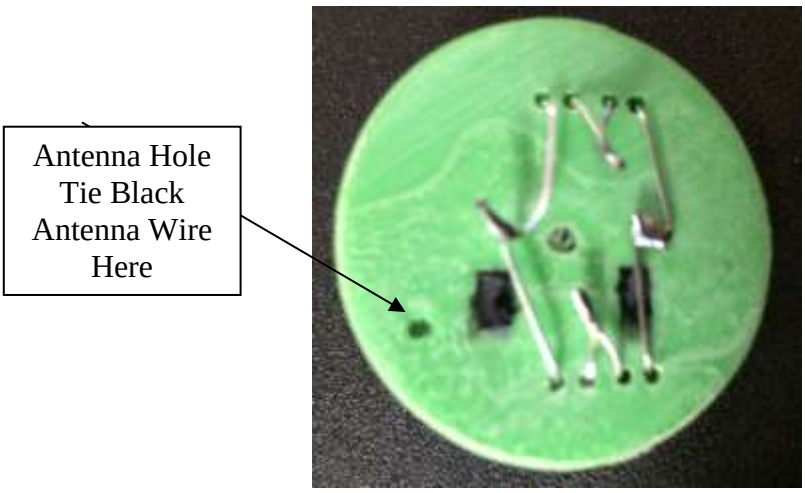


Figure 3.71 – Antenna Location

Once the antenna wire is tied, run the antenna wire through the small hole that was drilled on the top globe and run a 6/32" screw from the top of the globe through the center of the sensor disc. Attach with a 6/32 nut.

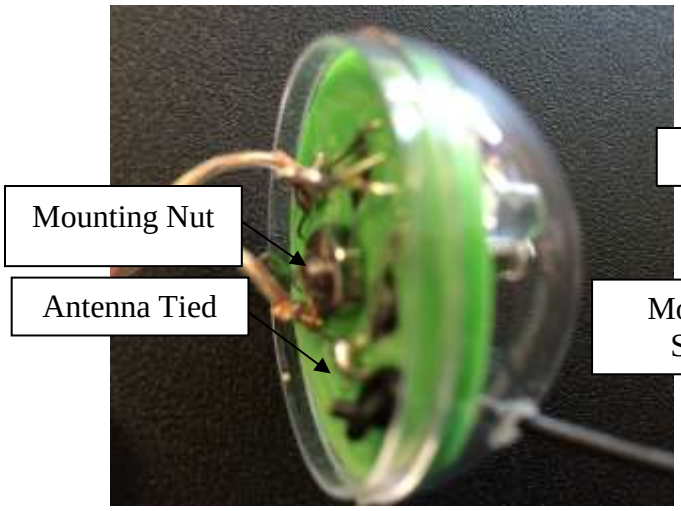


Figure 3.72 – Antenna

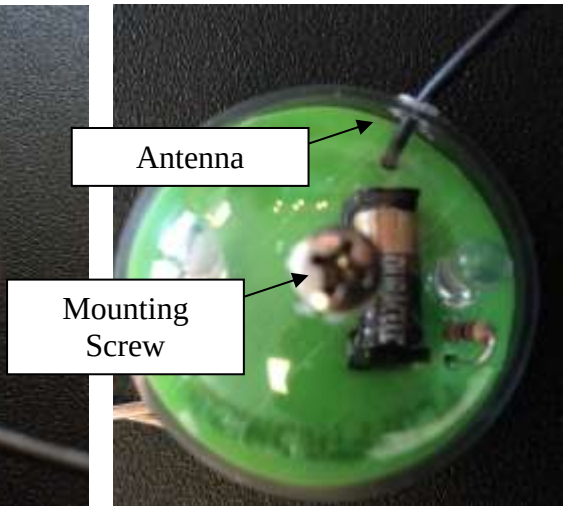


Figure 3.73 – Mounting Screw

Solder a 24" wire pair to the sensor areas shown below. I used speaker wire for this connection but pretty much anything would work. The wires for each sensor will be cut to length once the sensors are installed so longer is OK.

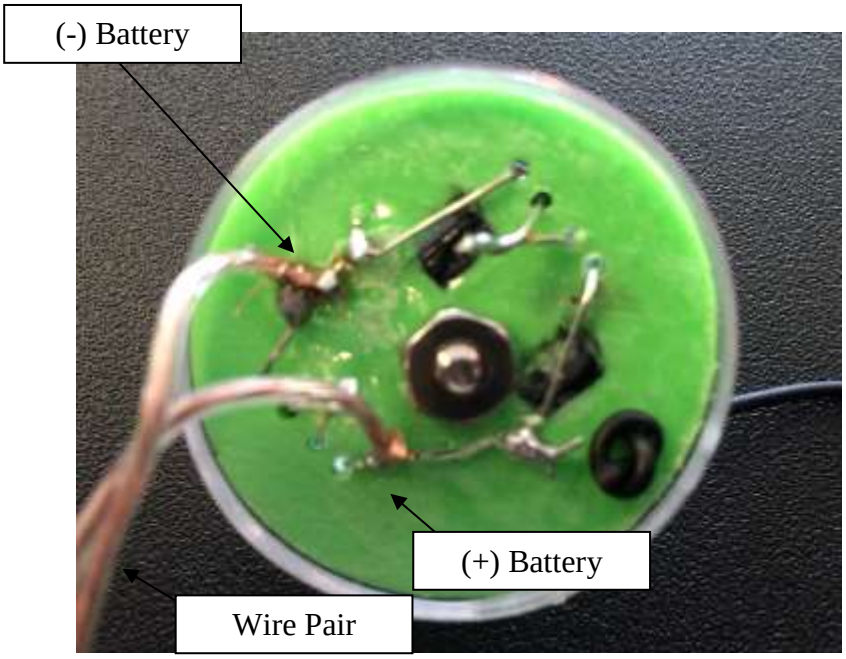


Figure 3.74 – Sensor Power Wires

To ensure everything stays in place and is electrically insulated/isolated, I apply liquid electrical tape to the back side of the sensor disc as shown below. Let this sit overnight and it will be dry by the morning.

Figure 3.75 – Liquid Electrical Tape Coating

Run the power wire from the upper sensor through the hole on the lower painted globe and press the two halves together.

Figure 3.77 – Lower Sensor

Figure 3.78 – Upper Sensor

You have just completed one Dorothy sensor, only 41 to go!

4 Dorothy Sensor Circular Base Construction

Once the sensors are assembled, we need to create a sensor plate for holding the sensors. The following steps outline the assembly process.

4.1 Prepare Sensor Circular Base

All 42 sensors will rest upon the 10.5' cake circle. Drill several holes as shown below that will allow the sensor power leads to pass through the cake circle. I have a large semicircle drilled into the bottom of the cake circle shown below. I am use his as a method for lifting the sensors out of the way so that the candy container can be refilled.

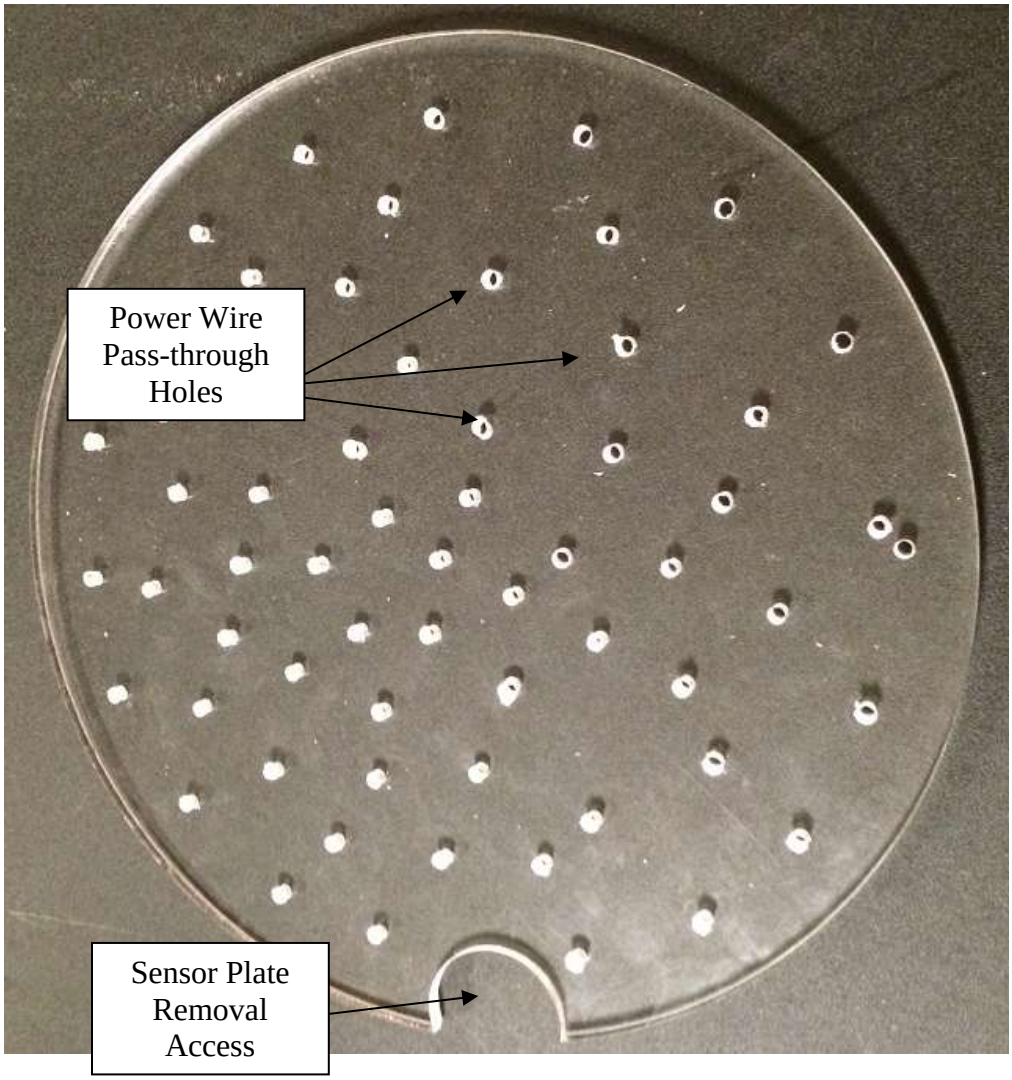


Figure 4.11 – Sensor Mounting Plate

Once the holes are drilled I mounted (4) 3 / 4" PVC plugs to the cake circle using 8/32" screws and lock nuts. 3 / 4" PVC couplers are slipped over the caps as shown below. Later we will cut (4) 10" sections of 3 / 4" PVC pipe that will be placed into the couplers. This will act as the stand for the sensor plate. The PVC connectors were purchased at Home Depot

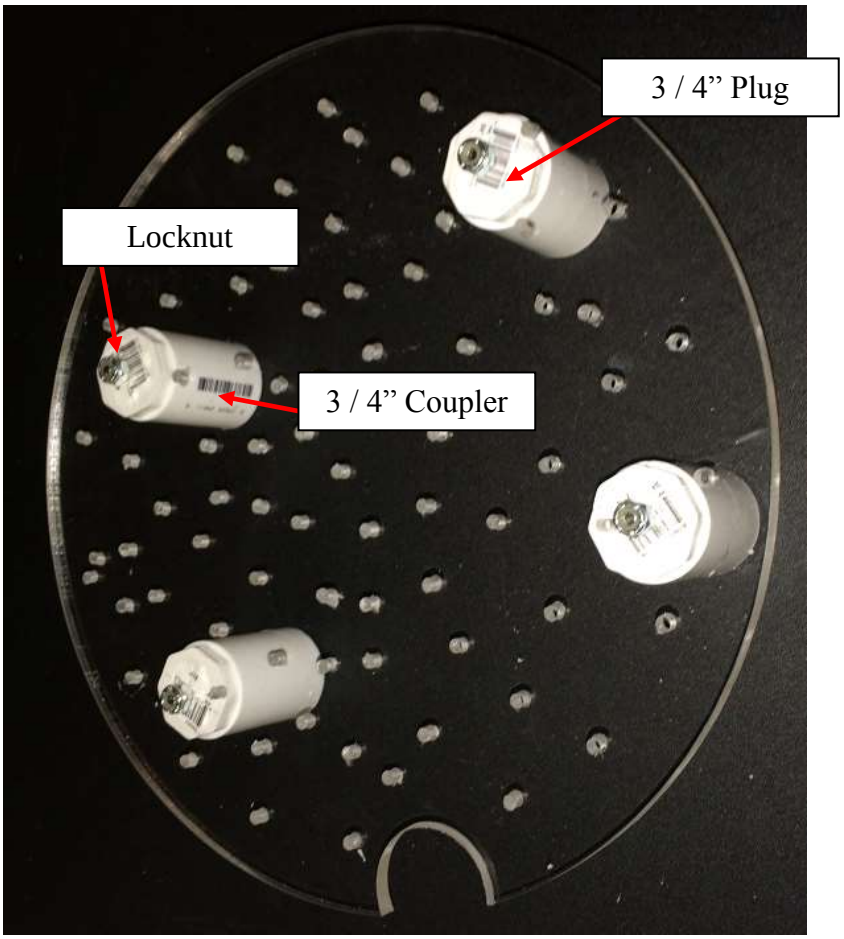


Figure 4.12 – Sensor Mounting Plate PVC Stand Mounts

Figure 4.13 – 3 / 4 "Plug

Figure 4.14 – 3 / 4 "Coupler

4.2 Paint Sensor Plate

Once the mounting feet were installed, I sprayed both sides of the sensor plate with Rust oleum hammered silver paint.

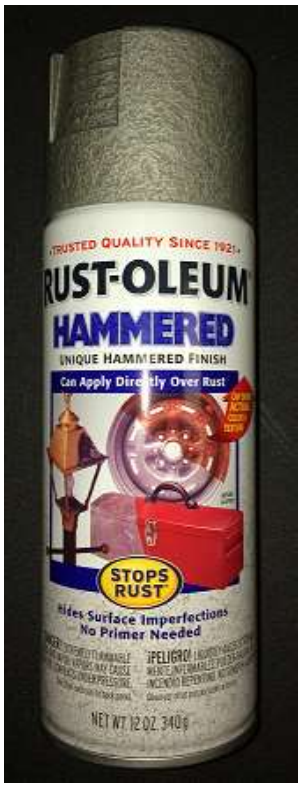


Figure 4.21 – Paint

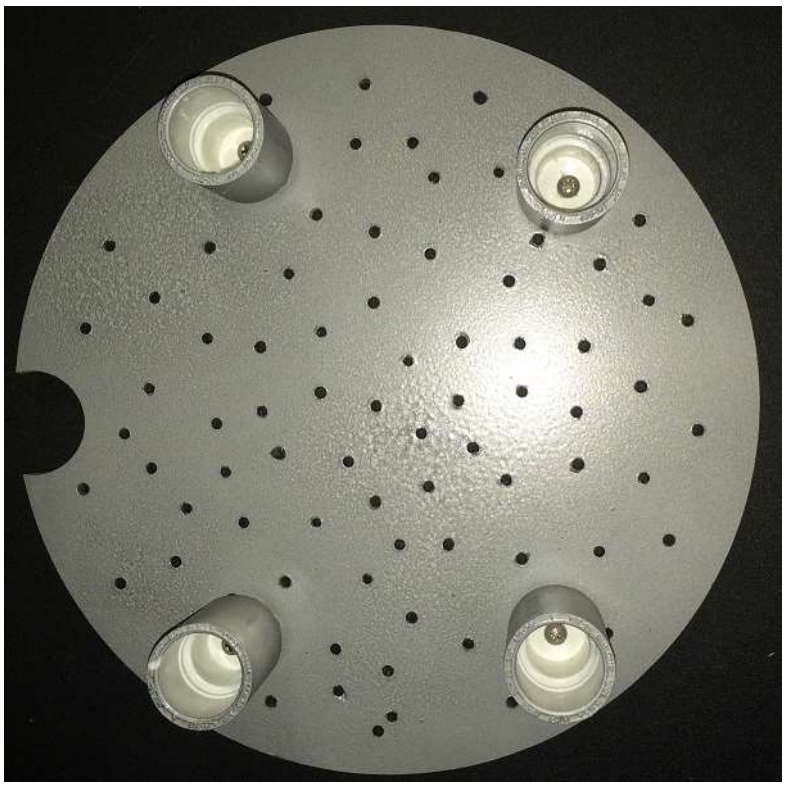


Figure 4.22 – Painted Sensor Plate

4.3 Attach Sensors

Each sensor power wire was fed through the mounting plate holes. I used some wire ties to temporarily keep the wires bundled together. The goal is to try to keep the center area as shallow as possible (no wires bunched up) since the plate will sit over the candy feeder and clearance is minimal.

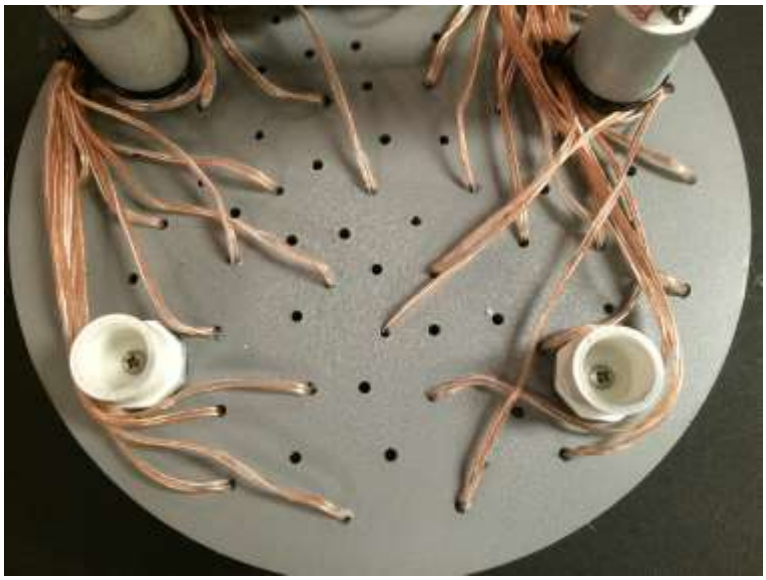


Figure 4.31 – Wire Routing

A 3: screw was added near the access semi-circle on the sensor plate to aid in the removal of the sensor plate. A lock nut is located on both sides of the sensor plate to hold the screw securely in place.

Figure 4.32 – Sensor Plate Removal Screw

With the sensor plate in the upright position, arrange the sensors on the plate so the power wires feed through the holes in the desired positon. Once you have it how you want it, glue the sensors to the plate using GE Silicone Iron Grip Adhesive (Purchase at Home Depot). The adhesive does not dry totally clear so avoid getting adhesive on the upper portion of the sensors where the adhesive would be visible.

The sensor plate utilizes 42 sensors. Since they do not all fit on the sensor plate, I opted for a tiered effect and have 6 of the sensors resting on top of the gaps between the sensors located on the plate. Refer to the pictures below. I just added adhesive to the bottom of the 6 sensors so they attach to the top portion of the other sensors.

Figure 4.33 – Adhesive Figure 4.34 – Sensor Plate Removal Screw

4.4 Make Sensor Plate Legs

Make the (4) legs for the sensor plate so that it sits above the candy dispenser.

The 3 / 4 “ PVC coupler is shown connected to a section of 3 / 4 “ PVC pipe below. The overall length from coupler to PVC pipe is 11”. Therefore the PVS pipe length is about 10 1 / 2”. The coupler will simply slide over the PVC plugs mounted on the sensor plate.

Figure 4.41 – PVC Coupler and PVC Pipe

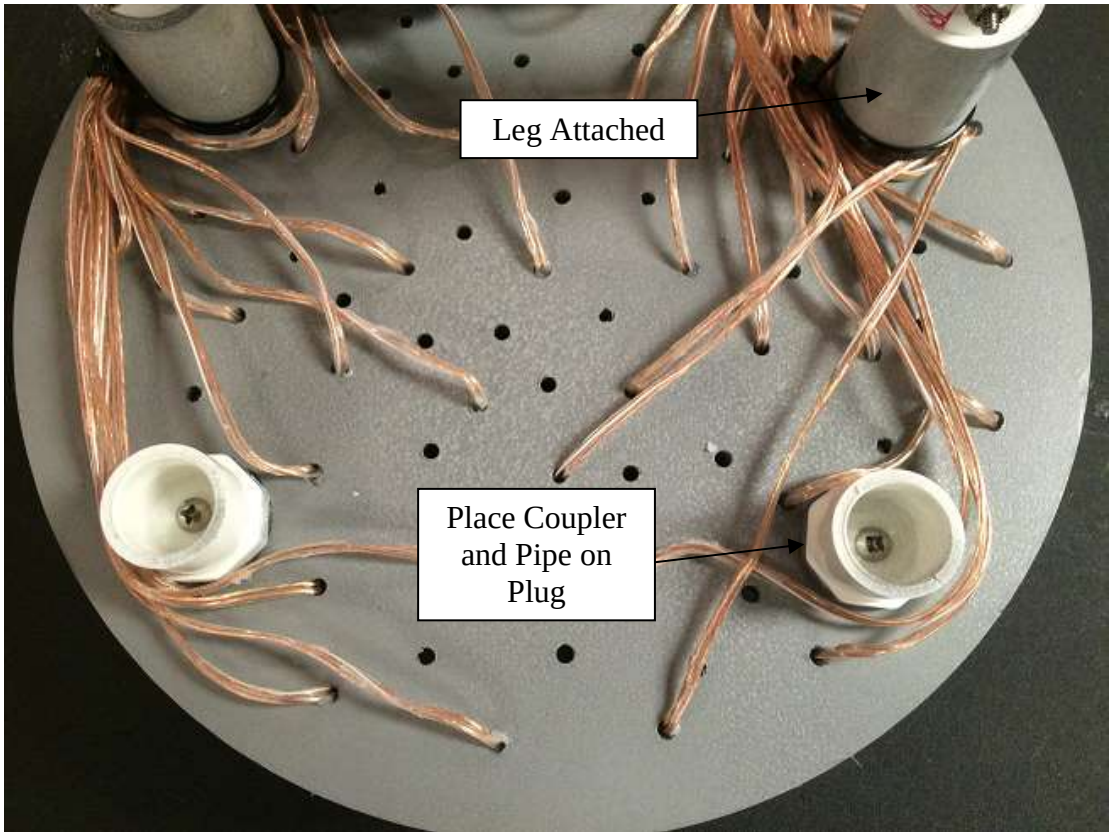


Figure 4.42 – Leg Mounting

5 Dorothy Candy Dispenser Construction

5.1 Candy Dispenser Assembly Overview

The candy dispenser is comprised of the following components.

- Candy Container
- Candy Container Feed Plate
- Clear PVC T
- Continuous Rotation Servo
- Servo Block
- Servo Mounting Plate
- Auger
- Candy Outfeed Tube

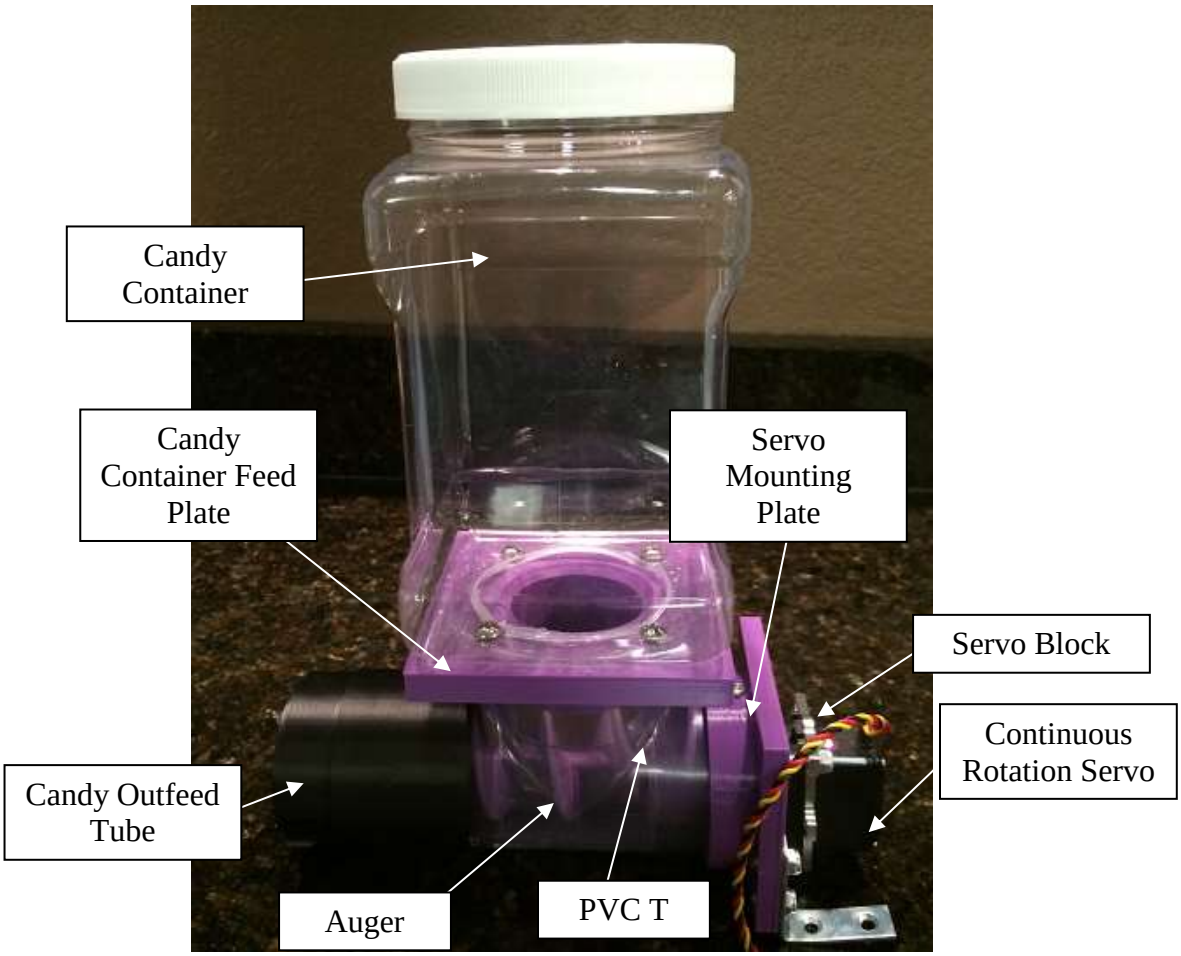


Figure 5.11 – Candy Dispenser Assembly

5.2 Candy Container / Feed Plate Construction

The candy container and feed plate are joined together using #10-24 x 3 / 4" screws and locknuts.

5.2.1 Feed Plate Printing

The feed plate is a printed piece that will be joined to the candy container. Print the feed plate as shown below in the color of your choice. The candy dispenser is not visible so utilize whatever you have available.

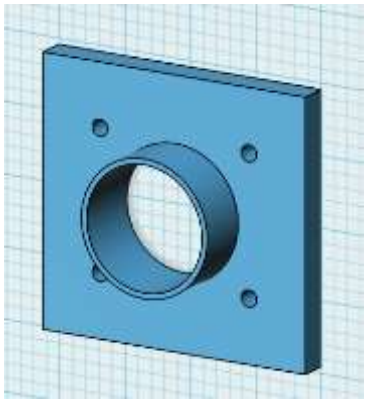


Figure 5.211 – Feed Plate 3D Rendering

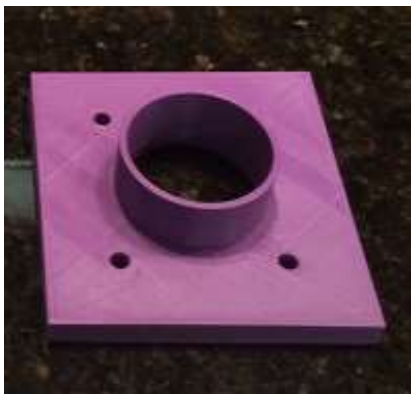


Figure 5.212 – Printed Feed Plate

5.2.2 Feed Plate / Candy Container Preparation

The feed container is mounted to the feed plate as shown below.

Figure 5.221 – Feed Plate / Candy Container Assembly

Place the feed plate on the bottom of the candy container as shown below. Use a marker to locate each of the four holes for the mounting the two pieces together and the outline for the candy to exit the container.

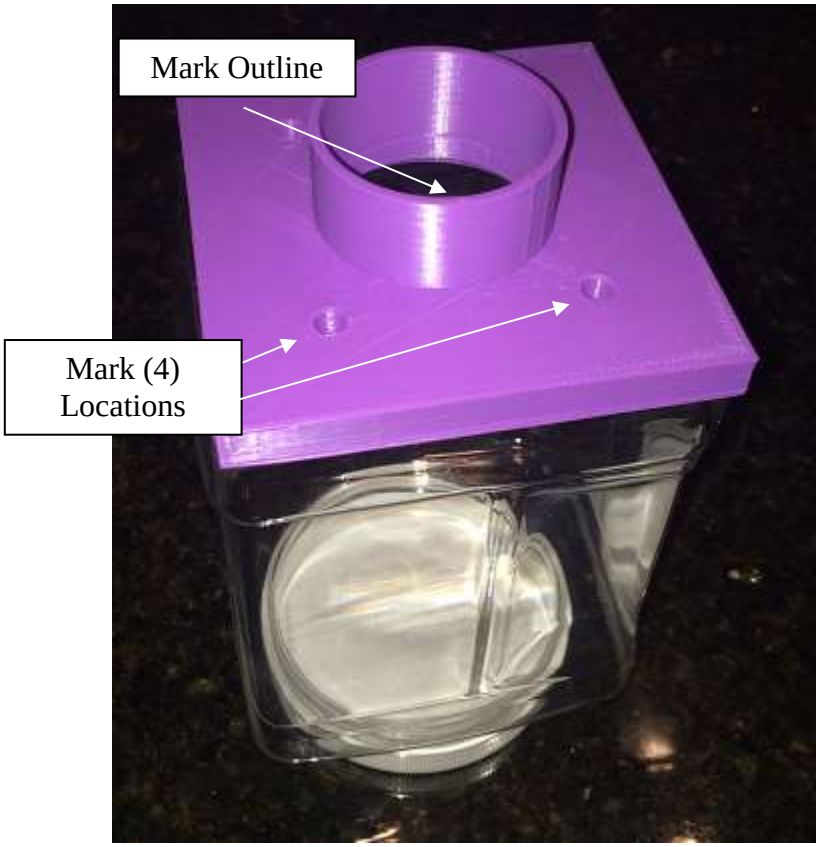


Figure 5.222 – Candy Container Alignment Marks

Drill the (4) small mounting locations so that #10-24 x 3 / 4" screws and be placed as shown below. Drill (1) 1-7 / 8" hole in the center bottom of the candy container.

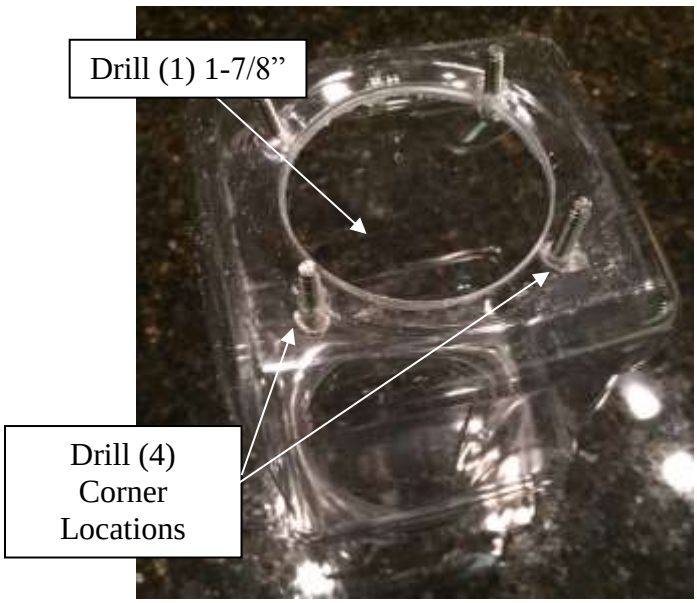


Figure 5.222 – Candy Container Drilling Results

5.2.3 Feed Plate / Candy Container Assembly

Join the candy container to the feed plate using (4) #10-24 x 3 / 4" screws and locknuts as shown below.

Figure 5.231 – Feed Plate / Candy Container Assembly

5.3 Auger Construction

The candy feed mechanism accomplished by utilizing an auger to dispense the candy. The auger is printed in two halves and glues together using the acrylic cement I use throughout the project.

5.3.1 Auger Printing / Assembly

Print both halves of the auger and glue together with the acrylic cement. Once both halves are joined, I sanded the augur so that the augur blades did not touch the inside of the clear T used to house the auger. Once the sanding was complete, I drilled out the mounting holes to clean them up and sprayed shellac on the augur since what I have read, it is a food safe finish.

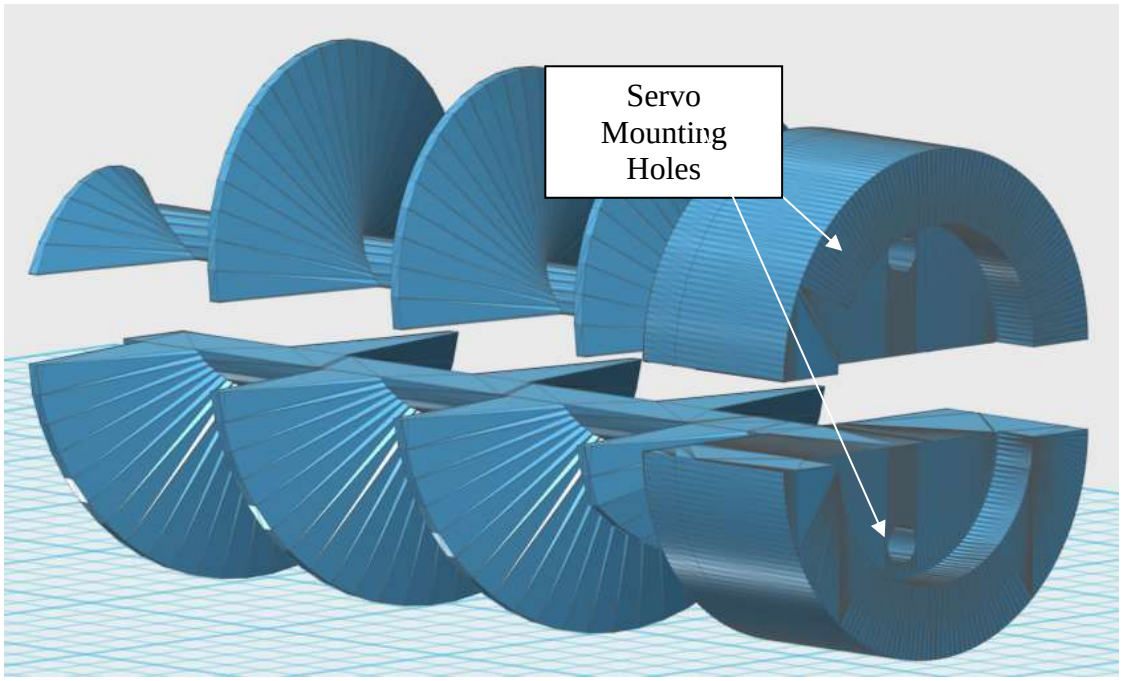


Figure 5.311 – Auger Printing / Assemble

5.4 Servo Mounting Plate Construction

The servo mounting plate attaches the servo / auger to the clear T. The servo mounting plate is printed as a single piece.

5.4.1 Servo Mounting Plate Printing

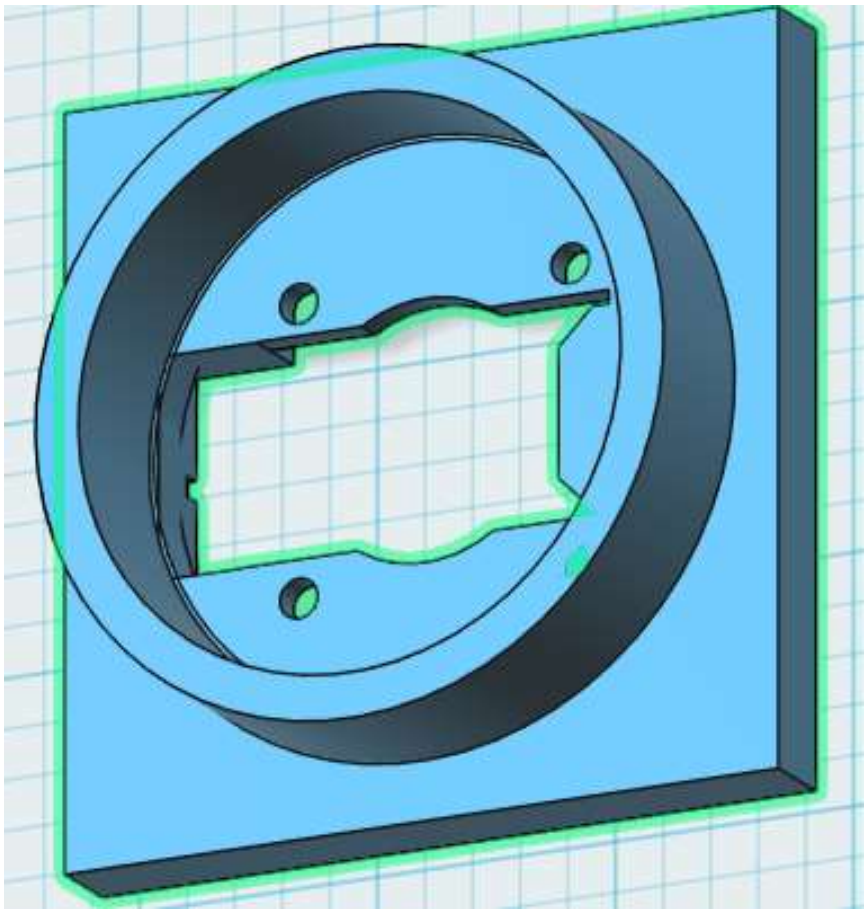


Figure 5.411 – Servo Mounting Plate

5.4.2 Prepare Servo Hub

The servo block parts will contain a servo hub. The servo hub is attached to the servo and the auger. This combination will provide the rotary motion required for the auger. The hub is shown below.

Figure 5.421 – Servo Mounting Hub

The servo block hub has tapped holes. I used 6/32 screws and locknuts to hold the hub to the auger to I drill two holes large enough to allow a 6/32" screw to pass through the hole easilt.

Figure 5.422 – Servo Mounting Drilling

5.4.3 Assembling Servo / Servo Mounting Plate / Auger

This step requires patience and is somewhat like a complicated puzzle to figure out the first time. I will list the steps required to fit everything together and provide a bunch of pictures so you can see how it is assembled.

1) Place X shaped servo block bearing is placed over the servo collar.
2) The servo X shaped bearing piece and servo is placed in the recess area on the back of the servo mounting plate/

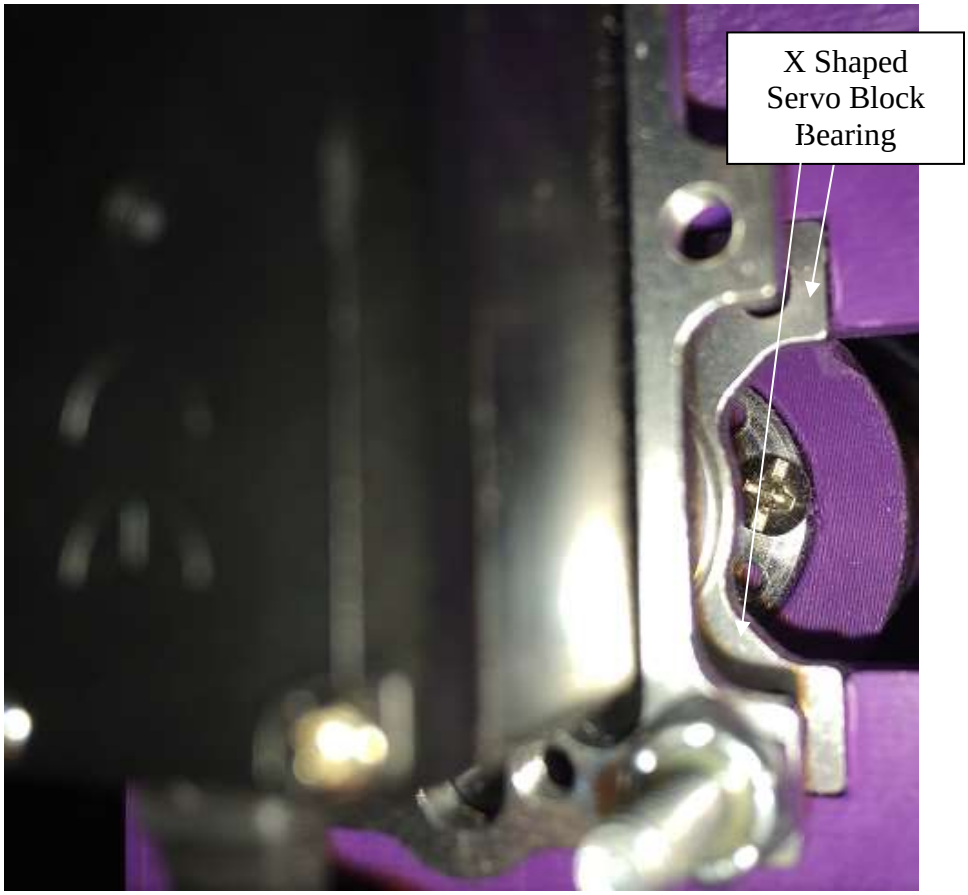


Figure 5.431 – Servo Block X Shaped Bearing

3) The servo hub is attached to the servo shaft via the small screw that goes in the center of the hub.

4) Slide the servo block rectangular mounting piece over the back of the servo.

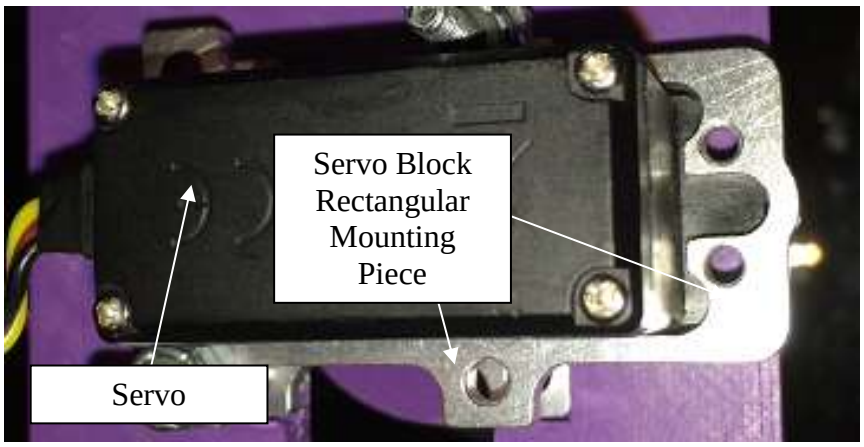


Figure 5.432 – Servo Block Rectangular Mounting Piece

5) Attach the mounting plate to the servo rectangular piece using two 6/32" screws and lock nuts.

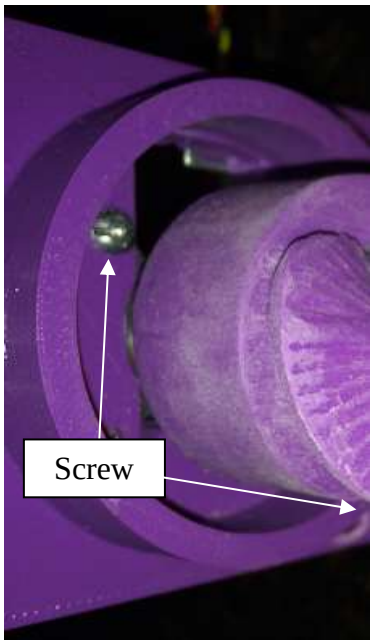


Figure 5.433 – Auger Side Screws

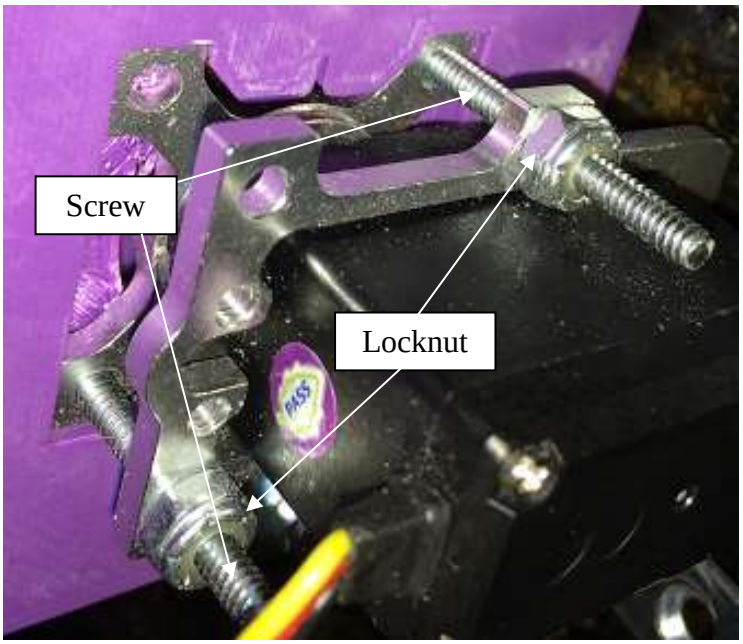


Figure 5.434 – Servo Side Screws

6) Attach the auger to the servo hub. This is accomplished by inserting one 6 / 32: screw through the back opening of the servo mounting plate. You will need to manually rotate the servo to get the servo hub hole to align with the hole in the back of the mounting plate so you can get a Philips screw drive in there. One you do this, attach a locknut to the screw. Rotate the auger/servo to insert the seconds screw and attach.

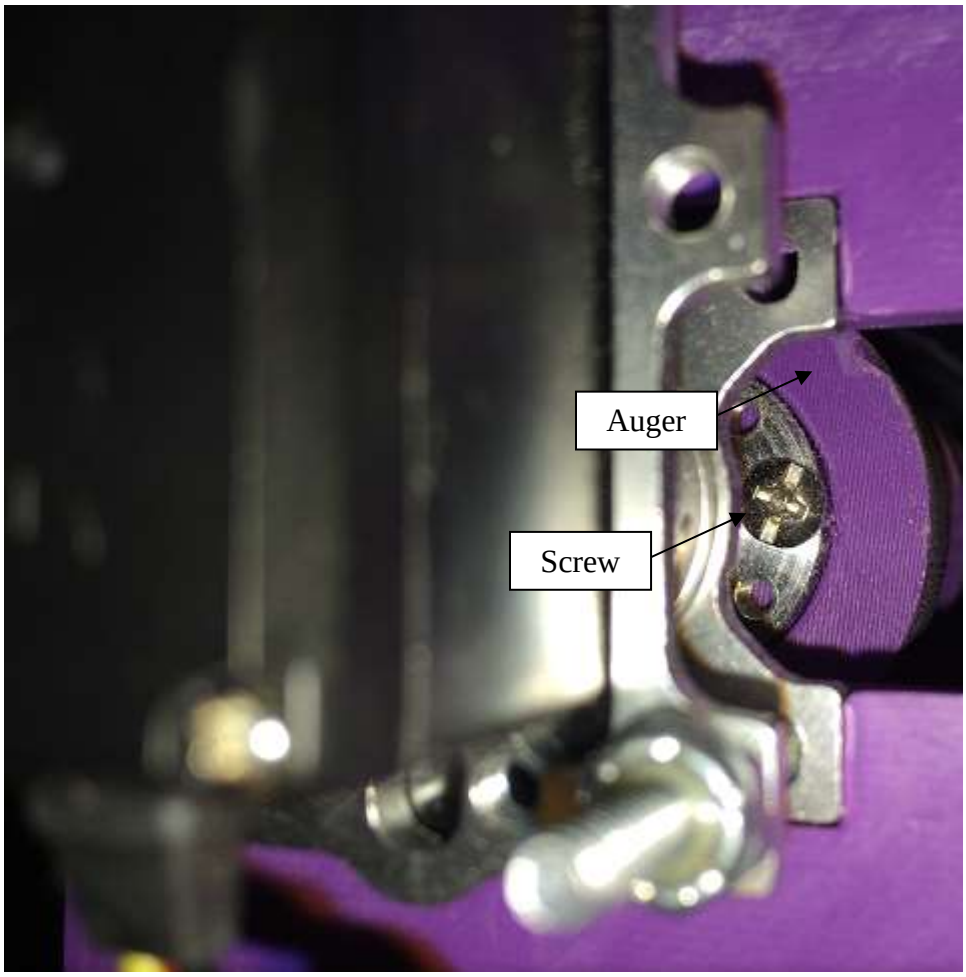


Figure 5.435 – Servo Mounting Hub / Auger Mounting

7) Attach mounting feet to servo mounting plate. The servo mounting feet are just L brackets that I purchased from Home Depot to allow the candy dispenser to be secured to Dorothy. Just drill four holes that align with the brackets as shown below and attached to 6/32" screws and locknut.

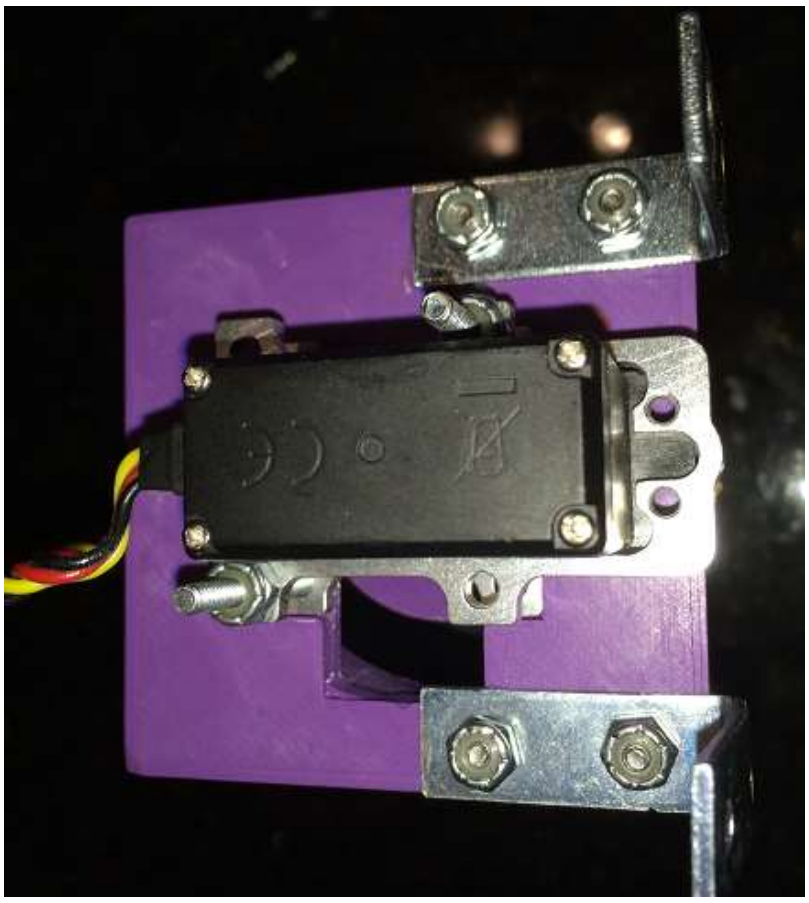


Figure 5.435 – Servo Mounting Hub / Auger Mounting

When you are all done you should have an assembly that appears as follows.

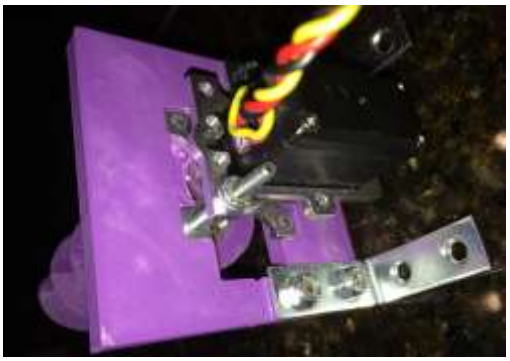


Figure 5.436

Figure 5.437

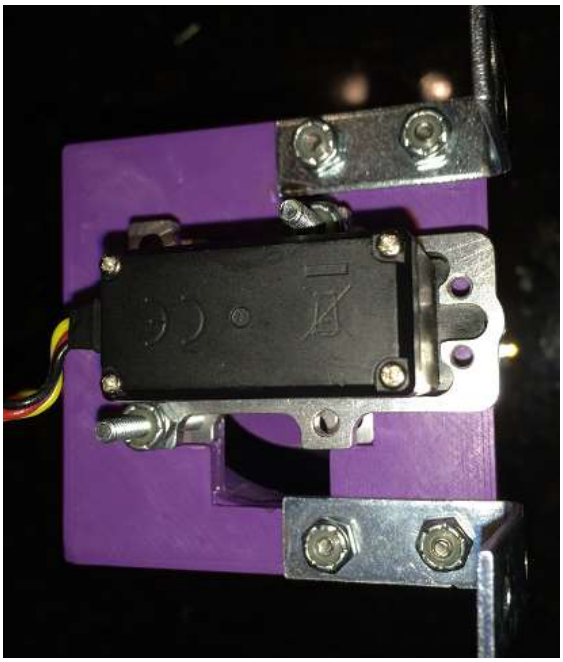


Figure 5.438

Figure 5.439

5.5 Candy Outfeed Tube

The Clear T has a candy outfeed tube attached to it that exits through Dorothy that the candy is dispensed through.

5.5.1 Candy Outfeed Tube Printing

Print the candy outfeed tube with black filament.

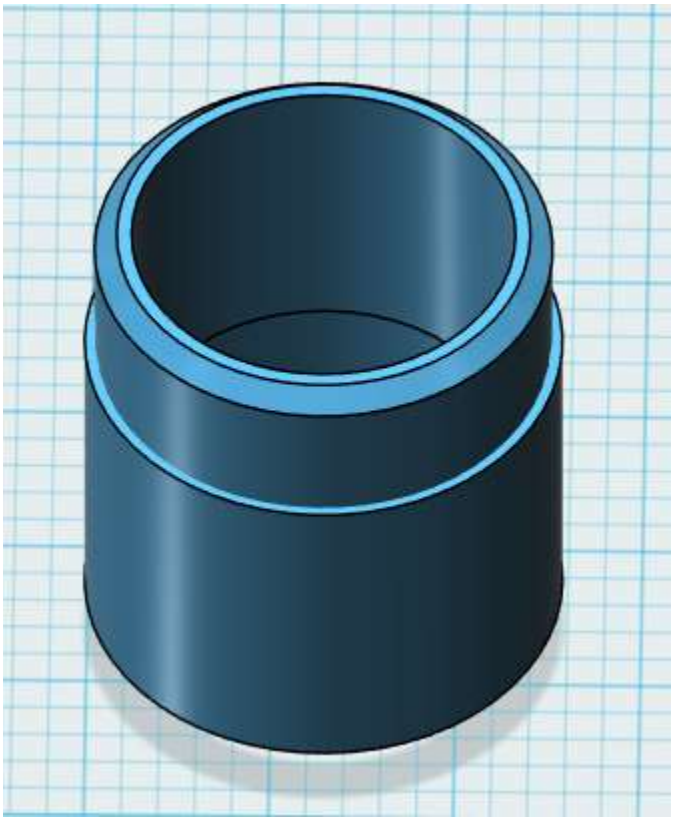


Figure 5.511 – Candy Outfeed Tube

5.6 Candy Dispenser Assembly

Assemble all of the components as shown below. The candy outfeed tube will slide over the clear T. The auger will slide into the clear T and the servo mounting plate can be pushed over the clear T. The candy container / feed plate can be pushed over the top of the clear T. I attached the servo mount and feed plate with a machine screw so the assembly would not move.

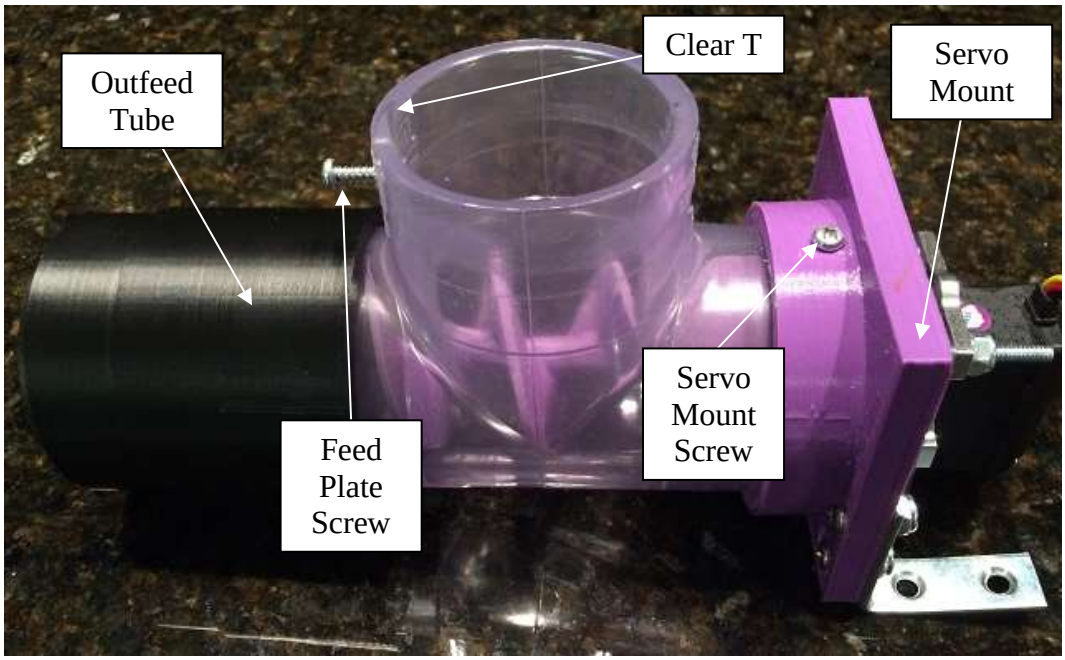


Figure 5.611 – Dispenser without Candy Container / Feed Plate

The entire dispenser assembly is shown below.

Figure 5.612 – Dispenser Assembly

6 Dorothy Dry Assembly

The Dorothy candy dispenser should be assembled prior to sanding and painting. Assembling prior to paint allows all of the components to be properly mounted so that corrective measures can be implemented so that the final assembly process is a simple operation with good results.

6.1 Candy Dispenser Component Designation

Once the initial assembly process was completed and all components were fitted to the candy dispenser, I designated four quadrants for the dispenser disassembly. The parts for each quadrant were placed in containers marked A, B, C and D. Using this method of grouping the parts by quadrant, I could easily reassemble the dispenser once the parts were painted. A top down view with the quadrant designation is shown below. Quadrant 'A' is the candy dispenser side. Dry fit the candy dispenser on the inside bottom of the bucket with the outfeed facing Quadrant A. Drill a hole the diameter of the dispenser opening and slide the dispenser through the opening. Drill holes on the bottom of the bucket to mount the dispenser.

Figure 6.11 – Dispenser Opening

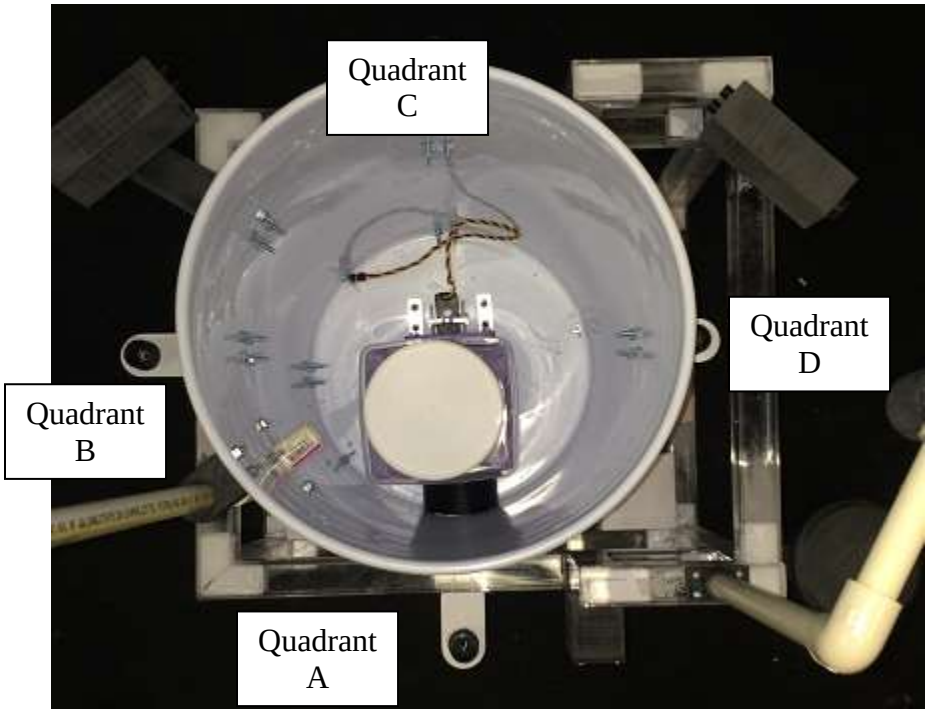


Figure 6.12 – Dispenser Quadrants

6.2 Quadrant 'A' Components

The quadrant 'A' components are shown below. These components were removed and placed in a quadrant 'A' container for painting.

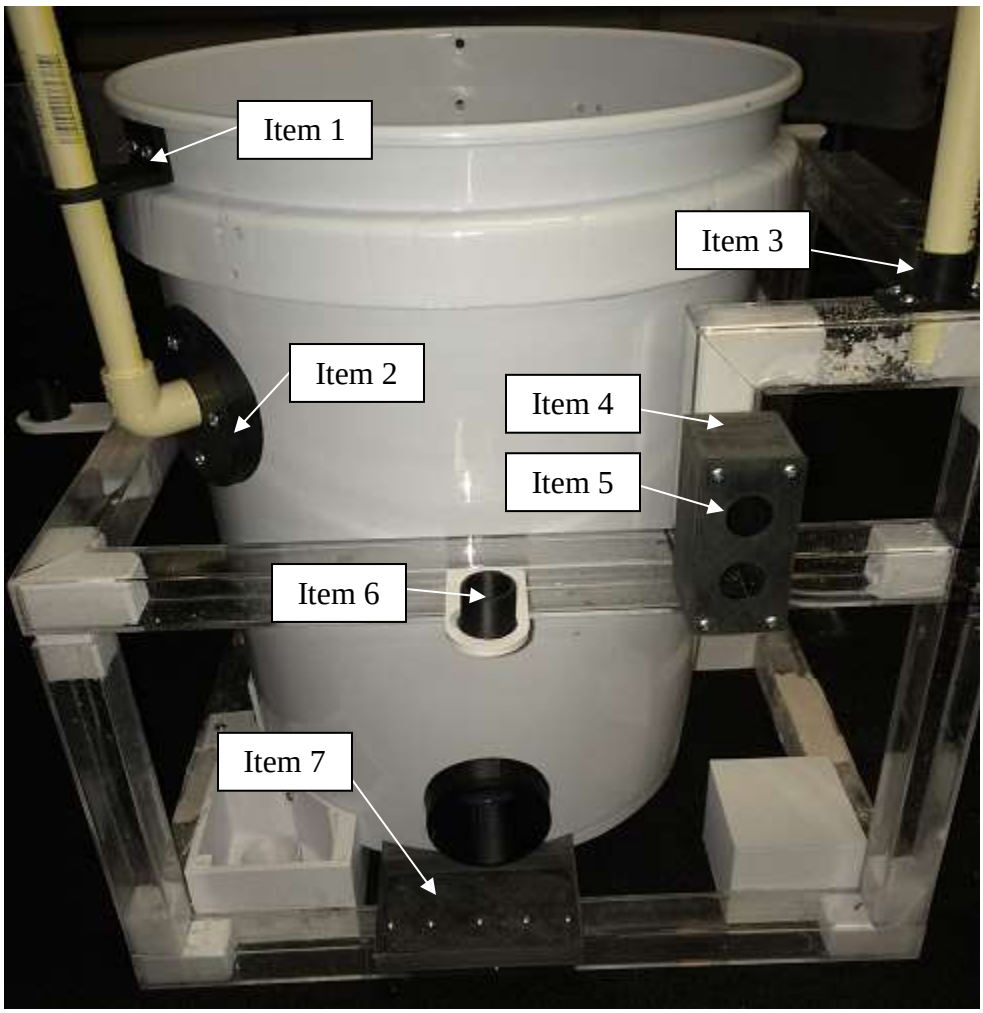


Figure 6.21 – Dispenser Quadrant 'A'

Quadrant 'A' 3D Printed Components (File Name)

Item 1 – Anemometer Bracket Upper.stl
Item 2 – Anemometer Bracket Lower.stl
Item 3 – Anemometer Holder.stl
Item 4 – Pushbutton Box Housing.stl
Item 5 – Pushbutton Box Cover.stl
Item 6 – Frame Beacon Bracket.stl & Beacon Base.stl
Item 7 – Candy Dish.stl

6.3 Quadrant 'B' Components

The quadrant 'B' components are shown below. These components were removed and placed in a quadrant 'B' container for painting.

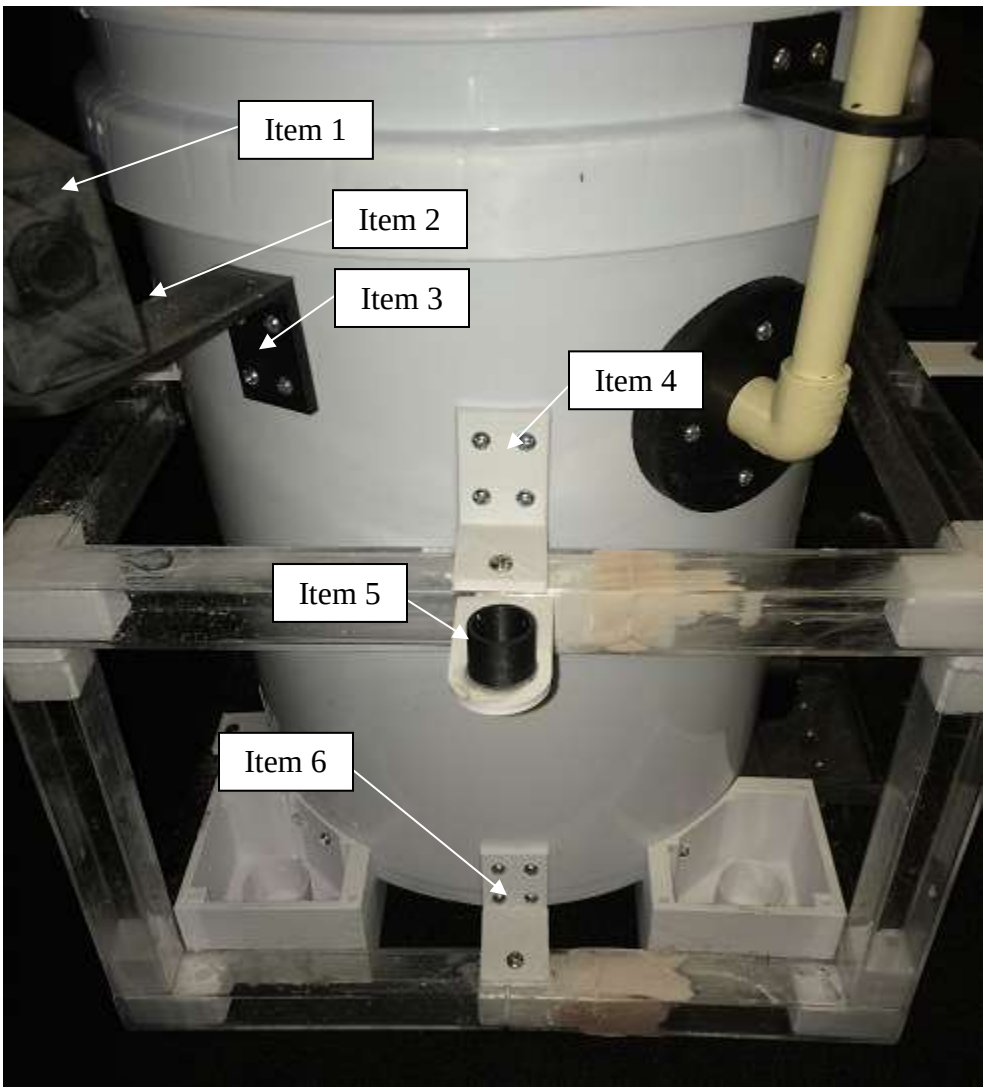


Figure 6.31 – Dispenser Quadrant 'B'

Quadrant 'B' 3D Printed Components (File Name)

Item 1 – Camera Body.stl
Item 2 – Camera Parts.stl
Item 3 – Camera Bracket.stl
Item 4 – Bracket Upper.stl
Item 6 – Frame Beacon Bracket.stl & Beacon Base.stl
Item 7 – Bracket Lower.stl

6.4 Quadrant 'C' Components

The quadrant 'C' components are shown below. These components were removed and placed in a quadrant 'C' container for painting.

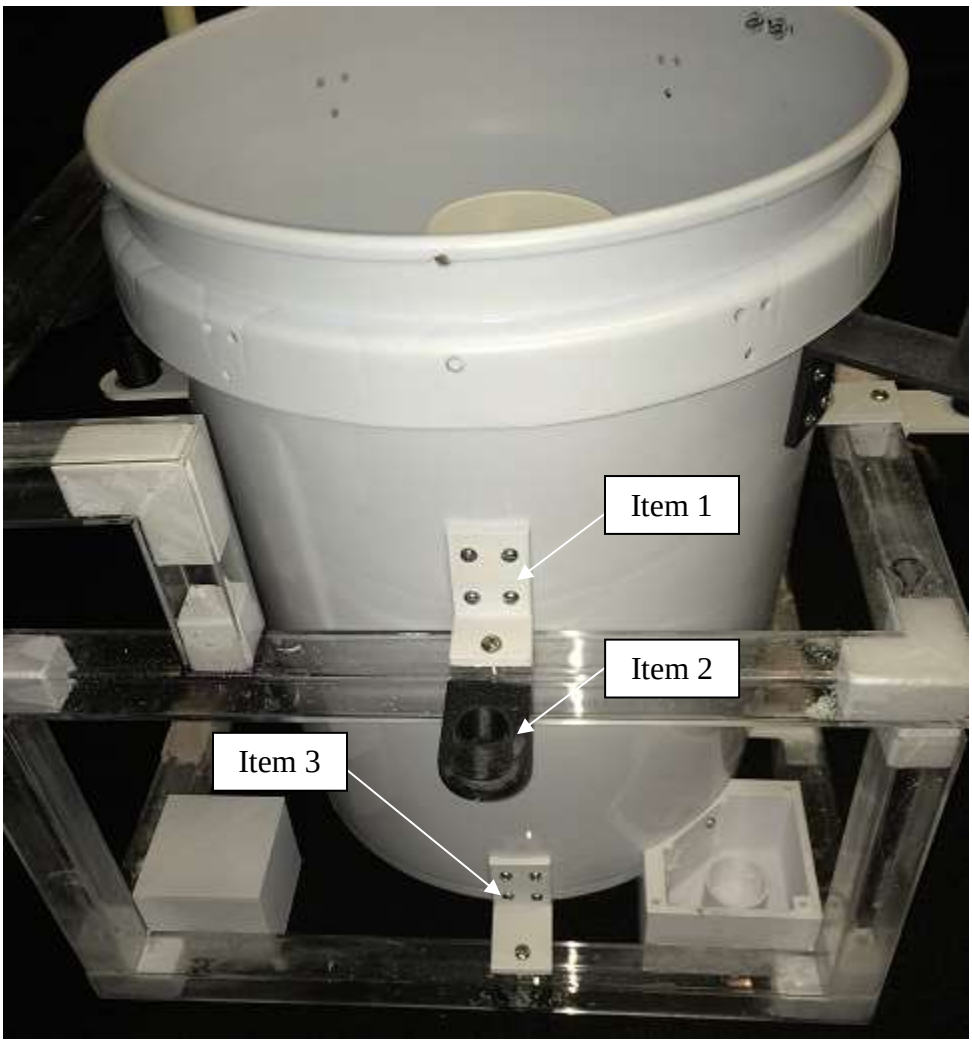


Figure 6.41 – Dispenser Quadrant 'C'

Quadrant 'C' 3D Printed Components (File Name)

Item 1 – Bracket Upper.stl
Item 2 – Frame Beacon Bracket.stl & Beacon Base.stl
Item 3 – Bracket Lower.stl

6.5 Quadrant 'D' Components

The quadrant 'D' components are shown below. These components were removed and placed in a quadrant 'D' container for painting.

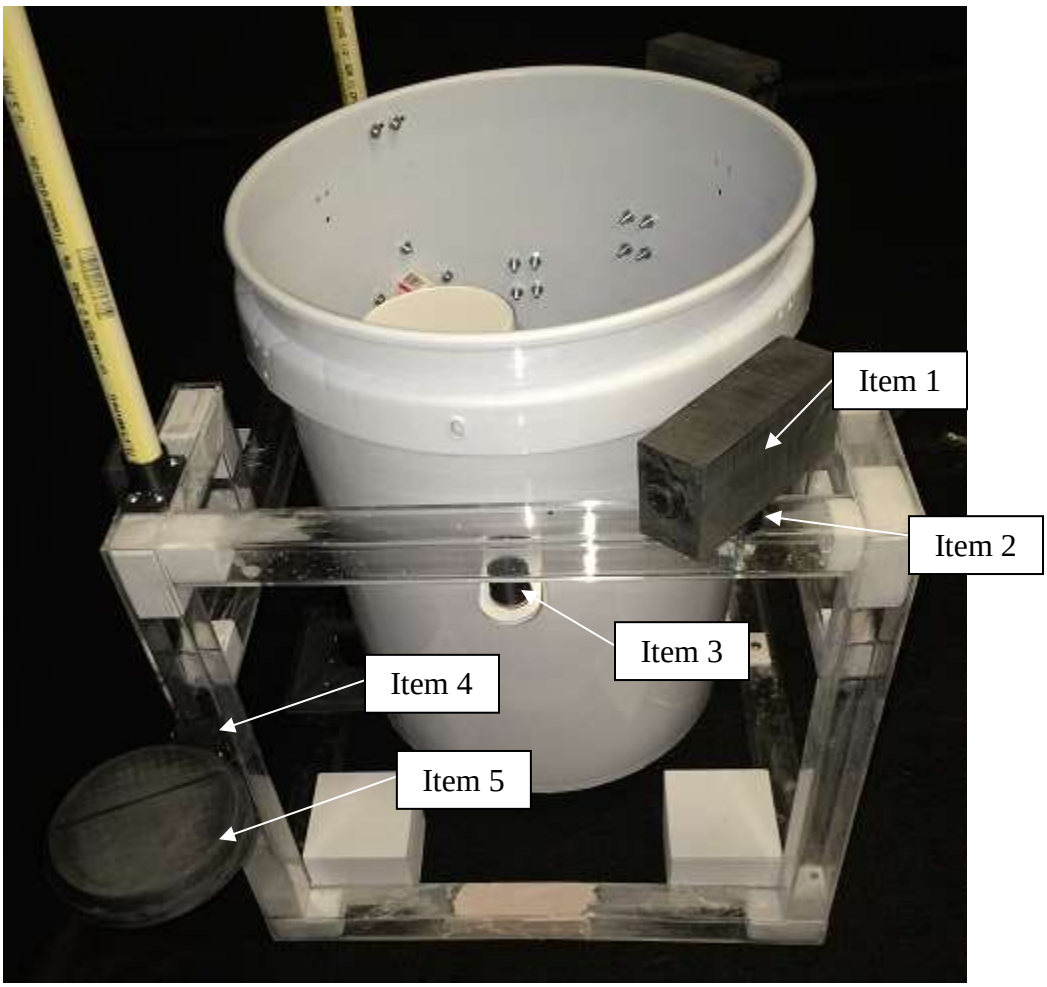


Figure 6.51 – Dispenser Quadrant 'D'

Quadrant 'D' 3D Printed Components (File Name)

Item 1 – Camera Body.stl
Item 2 – Camera Parts.stl
Item 3 – Frame Beacon Bracket.stl & Beacon Base.stl
Item 4 – Disc Sensor Bracket.stl
Item 5 – Disc Sensor Top.stl & Disc Sensor Bottom.stl

6.6 Top Down View

The Top Down view is shown below. The anemometer tubing is shown with one motor bracket attached.

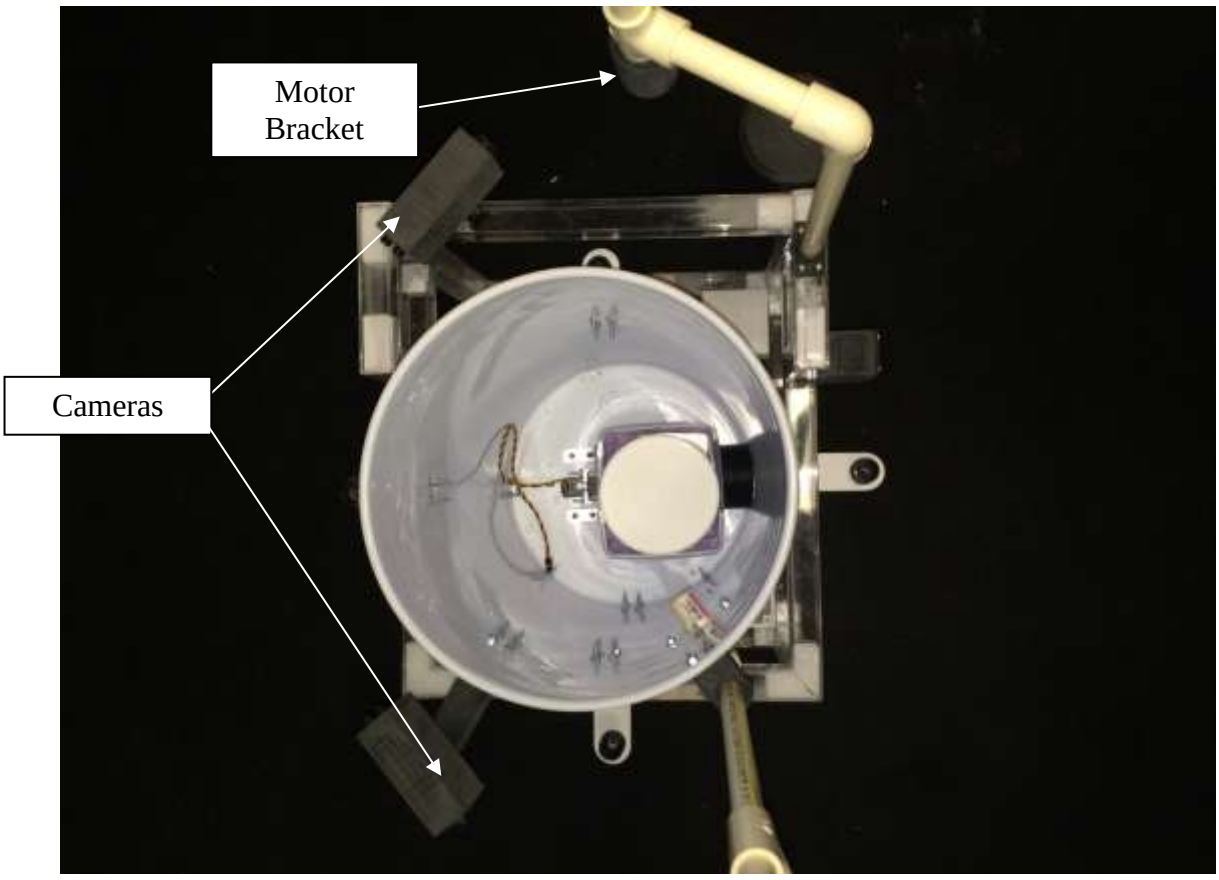


Figure 6.51 – Top Down View

6.7 Common Square Framing

The square framing is comprised of 1 inch outer diameter tuning that is cut to fit around the "bucket". The clear framing is cut with a miter saw and joined with 3D printed corner and joiner sections. The stanchion piece is used to feed the wires from the pushbutton panel trough the framing.

Figure 6.71 – Stanchion

Figure 6.72 – Frame Assembly

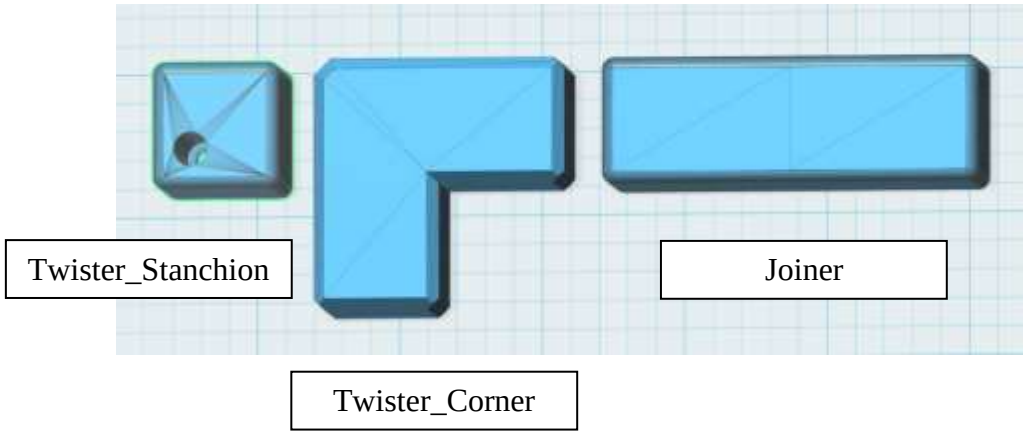


Figure 6.73 – Frame Joining Components

Quadrant Common Square Tubing Connectors
Item 1 – Joiner.stl
Item 2 – Twister_Corner.stl
Item 3 – Twister_Stanchion.stl

6.8 Anemometer Components

The anemometer components are shown below. Three anemometer cups are inserted into the Anemometer center piece. The round motor hub is mounted to the anemometer center section.

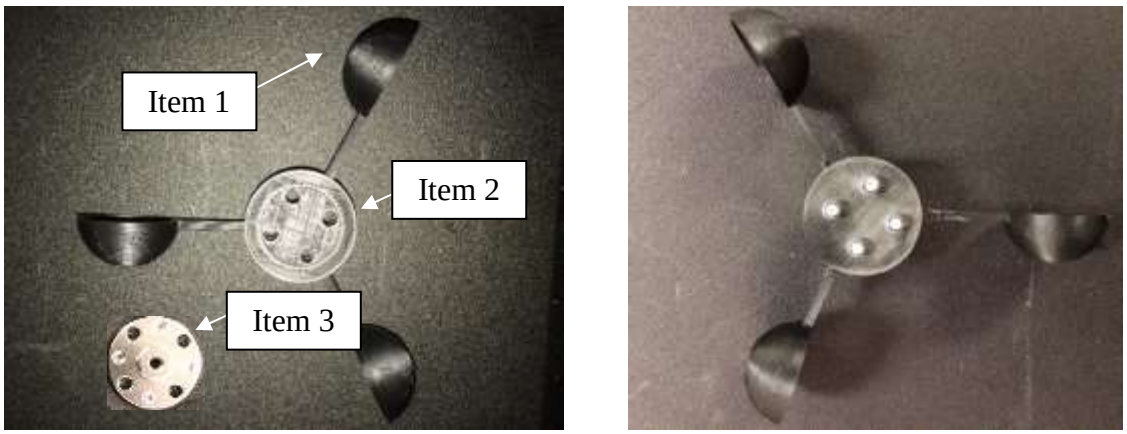


Figure 6.81 – Anemometer Components

Quadrant 'D' 3D Printed Components (File Name)

Item 1 – Anemometer Cups.stl
Item 2 – Anemometer Center.stl
Item 3 – Frame Beacon Bracket.stl & Beacon Base.stl
Item 4 – Disc Sensor Bracket.stl
Item 5 – Disc Sensor Top.stl & Disc Sensor Bottom.stl

The anemometers are mounted to the PVC tubing with the motor bracket shown below. The motor shaft protrudes through the square hole and is attached to the hub via a hub set screw. The round portion of the bracket shown below slides over the PVC tubing.

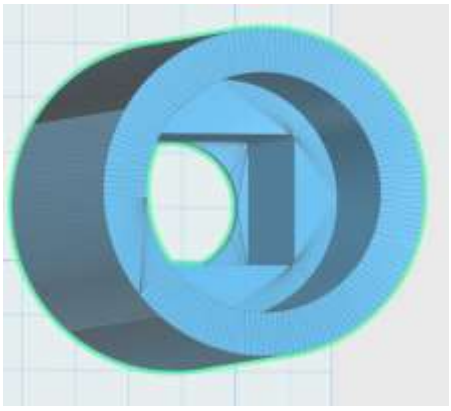


Figure 6.82 – Motor Bracket / Holder

The weather vanes are attached to the ¾" PVC tubing as shown below. PVC tubing, caps, elbows and tees are used for mounting the weather vanes and anemometers.

Figure 6.83 – Weather Vanes

6.9 Speakers

The speakers are placed in the two corner boxes as shown below. The speaker boxes are located on the front left hand side and rear left hand side of Dorothy.

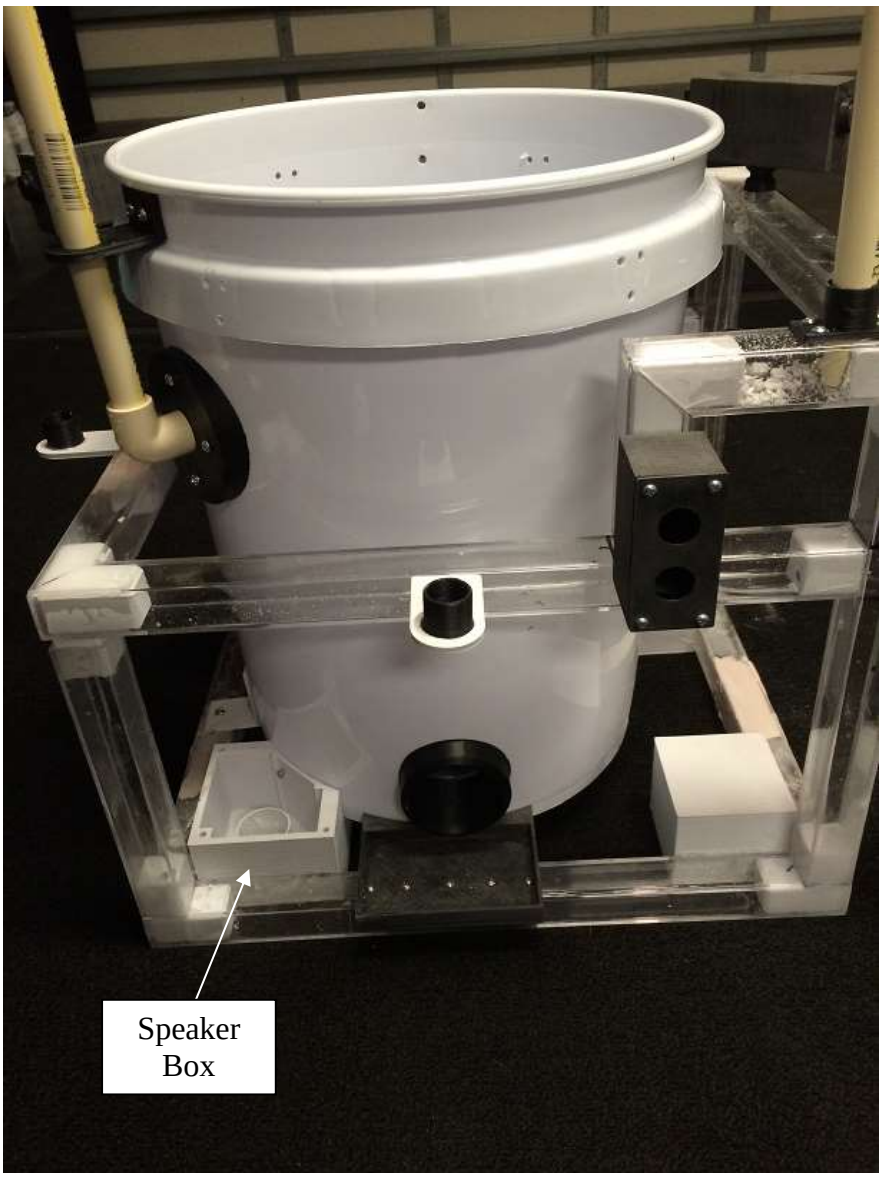


Figure 6.91 – Speaker Box Location

Figure 6.92 – Speaker Box with Speaker Installed

Figure 6.92 – Speaker Box with Lid Installed

6.10 Candy Dispenser Control Panel

Candy is dispensed by pressing the red pushbutton located on the candy dispenser shown below. The yellow pushbutton will run a show only and rotate the anemometers and play audio. The red pushbutton will rotate the anemometers, play audio and dispense the candy by rotating the auger motor.

Figure 6.101 – Candy Dispenser Control Panel

The candy dispenser control panel is printed in two parts (Pushbutton Box Housing and the Pushbutton Box Cover).

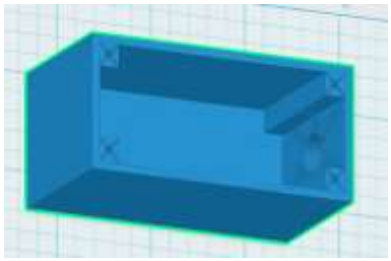


Figure 6.102 – Pushbutton Box Housing.stl

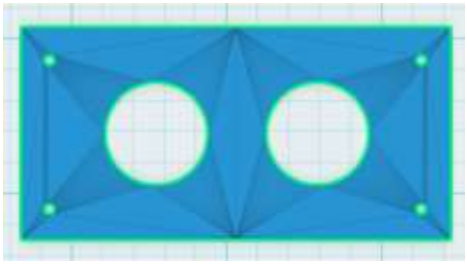


Figure 6.103 – Pushbutton Box Cover.stl

6.11 Candy Dispenser Lid Retainers

The Candy dispenser uses four lid retainers for allowing the clear Dorothy sensor lid to sit inside the retainers as shown below.

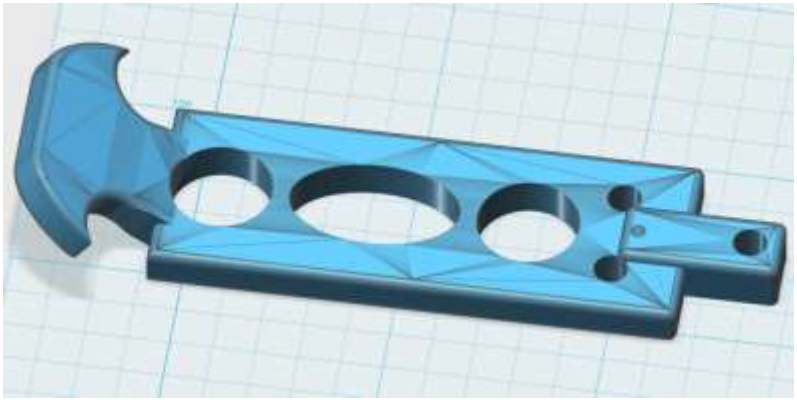
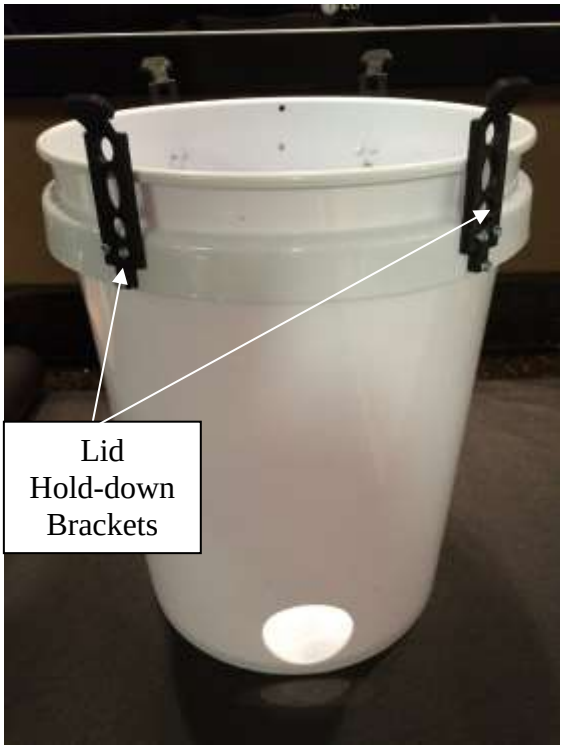


Figure 6.111 – Lid Holddown Bracket.stl



7 Painting

Sand all parts as required and using Bondo glazing putty, fill the seams of the frame assembly and all areas that require filling/blending. Once the glazing dries (1 -2 hours), sand using 220 grit sandpaper. Repeat as required.

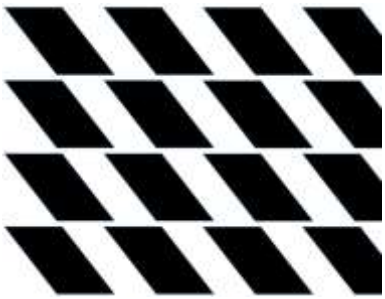
Figure 7.1 – Bondo Glazing Putty

Paint the Dorothy bucket, control panel pushbutton assembly, camera bodies, candy dish and tubing for the anemometer / weather vanes using RUST-OLEUM hammered silver paint. The frame is painted with RUST-OLEUM yellow paint. I sprayed the candy dish with shellac since the dish will be in contact with candy (food items).

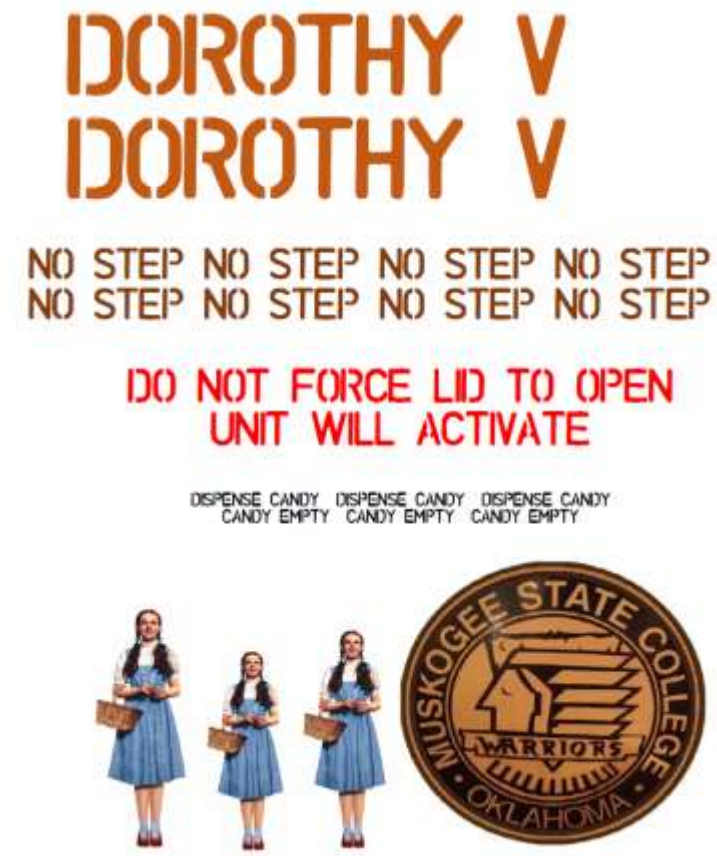
8 Decals

The frame for the Dorothy candy dispenser uses decals for the frame assembly that mimics the caution painting on the frame as shown below. The remaining decal graphics are for the located on Dorothy. The attached PDF contains the artwork for the Caution and Dorothy decals. The decals can be printed normally on 8.5" x 11" waterslide decal paper.

Caution Decals



Dorothy Decals



8.1 Decal Placement Front

8.2 Decal Placement Left

8.3 Decal Placement Rear

8.4 Decal Placement Rear

9 Arduino Code

```
//John Guarnero
//February 2018
//Dorothy Inspired Candy Dispenser

#include <Servo.h>

const int Red_PB_Pin_Light = 0;     // the number of the pin for the dispense pushbutton indicator
const int Yellow_PB_Pin_Light = 1;  // the number of the pin for the show only pushbutton indicator
const int Red_PB_Pin = 3;           // the number of the pin for the dispense pushbutton
const int Yellow_PB_Pin = 4;        // the number of the pin for the show only pushbutton

const int Strobe_1_Pin = 5;         // the number of the strobe 1 pin
const int Strobe_2_Pin = 6;         // the number of the strobe 2 pin
const int Strobe_3_Pin = 7;         // the number of the strobe 3 pin
const int Strobe_4_Pin = 8;         // the number of the strobe 4 pin
const int Servo_Power_Pin = 9;      // the number of the Servo Power pin
const int Anemometer_Pin = 11;      // the number of the anemomoter pin
const int Audio_Play_Pin = 12;      // the number of the Audio Trigger pin

int Dispense_PB = 0;                // variable for reading the dispense pushbutton status
int Dispensing = 0;                 // variable for reading the dispensing mode status
int Show_PB = 0;                    // variable for reading the show pushbutton status
int Show_Only = 0;                  // variable for reading the show pushbutton status
int Dispense_PB_Not_Pressed = 1;
int Show_PB_Not_Pressed = 1;

unsigned long previousMillis = 0;         // will store last time strobe LEDs were updated
unsigned long previousMillisPB = 0;       // will store last time for the pushbutton indicators were updated
unsigned long previousMillisDispense = 0; // will store last time the dispense mode was updated
unsigned long previousMillisShow = 0; // will store last time the dispense mode was updated

const long interval = 250;          // interval at which to blink (milliseconds)

int StrobeState = LOW;              // Strobe State
int PBLightState = LOW;             // Pushbuton Light State

Servo CandyServo;                   // create servo object to control a servo

void setup()
{
 CandyServo.attach(2);              // attaches the servo on pin 2 to the servo object
 pinMode(Red_PB_Pin_Light, OUTPUT);
 pinMode(Yellow_PB_Pin_Light, OUTPUT);
 pinMode(Red_PB_Pin, INPUT_PULLUP);
 pinMode(Yellow_PB_Pin, INPUT_PULLUP);
 pinMode(Strobe_1_Pin, OUTPUT);
 pinMode(Strobe_2_Pin, OUTPUT);
 pinMode(Strobe_3_Pin, OUTPUT);
 pinMode(Strobe_4_Pin, OUTPUT);
```

```
 pinMode(Anemometer_Pin, OUTPUT);
 pinMode(Audio_Play_Pin, OUTPUT);
 pinMode(Servo_Power_Pin, OUTPUT);

 digitalWrite(Red_PB_Pin_Light, LOW);
 digitalWrite(Yellow_PB_Pin_Light, LOW);
 digitalWrite(Strobe_1_Pin, HIGH);
 digitalWrite(Strobe_2_Pin, HIGH);
 digitalWrite(Strobe_3_Pin, HIGH);
 digitalWrite(Strobe_4_Pin, HIGH);
 digitalWrite(Anemometer_Pin, HIGH);
 digitalWrite(Audio_Play_Pin, HIGH);
 digitalWrite(Servo_Power_Pin, HIGH);
}

void loop()
{
unsigned long currentMillis = millis();

   Dispense_PB = digitalRead(Red_PB_Pin);   //Dispense Candy PB - Input HIGH is not pressed
   Show_PB = digitalRead(Yellow_PB_Pin);   //Dispense Candy PB - Input HIGH is not pressed

 if (Dispense_PB == HIGH && Show_PB == HIGH) //Inputs High so the pushbuttons are not pressed
 {

 }

 if ((Dispense_PB == LOW && Dispense_PB_Not_Pressed == 1 && Show_Only == 0)) //Input High is not
pressed and output HIGH is Relay Off
 {
    Dispensing = 1;
    previousMillisDispense = currentMillis;
    Dispense_PB_Not_Pressed = 0;
  }

if ((Show_PB == LOW && Show_PB_Not_Pressed == 1  && Dispensing == 0)) //Input High is not pressed
and output HIGH is Relay Off
 {
    Show_Only = 1;
    previousMillisShow = currentMillis;
    Show_PB_Not_Pressed = 0;
  }

 if (Dispensing == 0 && Show_Only == 0) //Stop Dispensing Candy
 {
    digitalWrite(Anemometer_Pin, HIGH);
    delay (10);
    digitalWrite(Audio_Play_Pin, HIGH);
    delay (10);
    digitalWrite(Servo_Power_Pin, HIGH);
    delay (10);
 }
```

```
if (Show_Only == 1) //Dispense Candy
{

   digitalWrite(Anemometer_Pin, LOW);        //run the anemometers - High = Off
   digitalWrite(Servo_Power_Pin, HIGH);       //power to the servo - High = Off
   digitalWrite(Audio_Play_Pin, LOW);        //play theme music - High = Off
   digitalWrite(Red_PB_Pin_Light, HIGH);     //make red pushbutton indicator on solid
   digitalWrite(Yellow_PB_Pin_Light, HIGH);  //make yellow pushbutton indicator on solid
}


 if (Dispensing == 1) //Dispense Candy
{
   digitalWrite(Anemometer_Pin, LOW);        //run the anemometers - High = Off
   digitalWrite(Servo_Power_Pin, LOW);       //power to tne servo - High = Off
   digitalWrite(Audio_Play_Pin, LOW);        //play theme music - High = Off
   digitalWrite(Red_PB_Pin_Light, HIGH);     //make red pushbutton indicator on solid
   digitalWrite(Yellow_PB_Pin_Light, HIGH);  //make yellow pushbutton indicator on solid
}


// Start rotating auger for dispensing candy

//Dispense amount 1
  if (currentMillis - previousMillisDispense >= 5001  && currentMillis - previousMillisDispense <= 5800)
{
     CandyServo.write(140);                // rotate dispense
 }

 if (currentMillis - previousMillisDispense >= 6001  && currentMillis - previousMillisDispense <= 6200)
{
     CandyServo.write(93);                 // stop
  //    digitalWrite(Audio_Play_Pin, HIGH);
 }

 if (currentMillis - previousMillisDispense >= 6201  && currentMillis - previousMillisDispense <= 6400)
{
     CandyServo.write(50);                 // rotate clear candy
 }

 if (currentMillis - previousMillisDispense >= 6601  && currentMillis - previousMillisDispense <= 6800)
{
     CandyServo.write(93);                 // stop
 }


//Dispense amount 2
  if (currentMillis - previousMillisDispense >= 7001  && currentMillis - previousMillisDispense <= 7800)
{
     CandyServo.write(140);                // rotate dispense
 }

 if (currentMillis - previousMillisDispense >= 8001  && currentMillis - previousMillisDispense <= 8200)
{
```

```
    CandyServo.write(93);              // stop
 }

 if (currentMillis - previousMillisDispense >= 8201  && currentMillis - previousMillisDispense <= 8400)
{
    CandyServo.write(50);              // rotate clear candy
 }

 if (currentMillis - previousMillisDispense >= 8601  && currentMillis - previousMillisDispense <= 8800)
{
    CandyServo.write(93);              // stop
 }


//Dispense amount 3
  if (currentMillis - previousMillisDispense >= 9001  && currentMillis - previousMillisDispense <= 9800)
{
    CandyServo.write(140);              // rotate dispense
 }

 if (currentMillis - previousMillisDispense >= 10501  && currentMillis - previousMillisDispense <= 10700)
{
    CandyServo.write(93);              // stop
 }

 if (currentMillis - previousMillisDispense >= 10701  && currentMillis - previousMillisDispense <= 10900)
{
    CandyServo.write(50);              // rotate clear candy
 }

 if (currentMillis - previousMillisDispense >= 10901  && currentMillis - previousMillisDispense <= 11100)
{
    CandyServo.write(93);              // stop
 }


//Dispense amount 4
  if (currentMillis - previousMillisDispense >= 11101  && currentMillis - previousMillisDispense <= 11800)
{
    CandyServo.write(140);              // rotate dispense
 }

 if (currentMillis - previousMillisDispense >= 12201  && currentMillis - previousMillisDispense <= 12400)
{
    CandyServo.write(93);              // stop
 }

 if (currentMillis - previousMillisDispense >= 12401  && currentMillis - previousMillisDispense <= 12600)
{
    CandyServo.write(50);              // rotate clear candy
 }

 if (currentMillis - previousMillisDispense >= 12601  && currentMillis - previousMillisDispense <= 12800)
{
```

```
    CandyServo.write(93);              // stop
 }


// Done Dispensing so turn everything off
  if ((currentMillis - previousMillisDispense >= 17000) && Dispensing == 1)
{
    digitalWrite(Anemometer_Pin, HIGH);
    digitalWrite(Audio_Play_Pin, HIGH);
    digitalWrite(Servo_Power_Pin, HIGH);
    Dispensing = 0;
    Show_Only = 0;
    Dispense_PB_Not_Pressed = 1;
    Show_PB_Not_Pressed = 1;
}

  if ((currentMillis - previousMillisShow >= 17000) && Show_Only == 1)
{
    digitalWrite(Anemometer_Pin, HIGH);
    digitalWrite(Audio_Play_Pin, HIGH);
    digitalWrite(Servo_Power_Pin, HIGH);
    Dispensing = 0;
    Show_Only = 0;
    Dispense_PB_Not_Pressed = 1;
    Show_PB_Not_Pressed = 1;
}
// }


//Blink the strobe beacons
 if (currentMillis - previousMillis >= interval)
 {
    // save the last time you blinked a strobe
    previousMillis = currentMillis;

    // if the strobe is off turn it on and vice-versa:
    if (StrobeState == LOW)
    {
      StrobeState = HIGH;
       digitalWrite(Strobe_1_Pin, HIGH);
       digitalWrite(Strobe_2_Pin, HIGH);
       digitalWrite(Strobe_3_Pin, LOW);
       digitalWrite(Strobe_4_Pin, LOW);
      }
      else
    {
      StrobeState = LOW;
       digitalWrite(Strobe_1_Pin, LOW);
       digitalWrite(Strobe_2_Pin, LOW);
       digitalWrite(Strobe_3_Pin, HIGH);
       digitalWrite(Strobe_4_Pin, HIGH);
    }
 }
```

```
//Blink the red and yellow pushbutton LEDs
 unsigned long currentMillisPB = millis();

 if (currentMillisPB - previousMillisPB >= 200)
 {
    // save the last time you blinked the Pushbutton LEDs
    previousMillisPB = currentMillisPB;

    // if the strobe is off turn it on and vice-versa:
    if (PBLightState == LOW)
    {
      PBLightState = HIGH;

 digitalWrite(Red_PB_Pin_Light, HIGH);
 digitalWrite(Yellow_PB_Pin_Light, LOW);

    } else
    {
      PBLightState = LOW;
 digitalWrite(Red_PB_Pin_Light, LOW);
 digitalWrite(Yellow_PB_Pin_Light, HIGH);
    }
   }
}
```