ANIMATRONICS FACE: PROF COMBS



Haitham Hussain

Dept. of Mechanical Engineering

San Antonio, TX, USA 78249

[igs407@my.utsa.edu](mailto:Rowdy.runner2@utsa.edu)

Meggan Wolanin

Dept. of Mechanical Engineering

San Antonio, TX, USA 78249

dfe695@my.utsa.edu

Brenton Clark

Dept. of Mechanical Engineering

San Antonio, TX, USA 78249

Brenton.mc121@gmail.com

Submitted as part of project for ME4543 Mechatronics, Fall 2018

TABLE OF CONTENTS

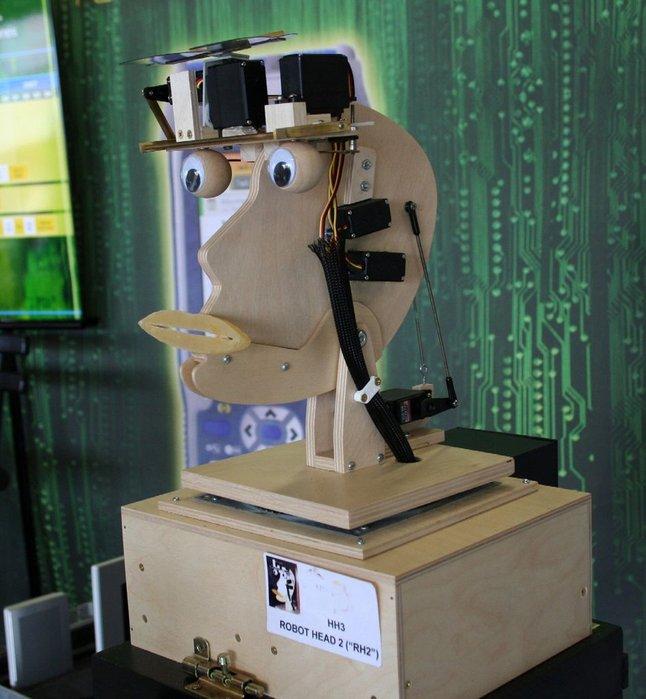
|  |  |
| --- | --- |
|  |  |
| Abstract | 3 |
| Literature Review | 4 |
| Supporting Structure | 12 |
| Joints and Motors | 13 |
| Sensors | 13 |
| Programming | 14 |
| Lesson learnt and suggestions | 14 |
| Personnel and Bill of Materials | 15 |
| Appendix A: Code snippets | 18 |
|  |  |

**ABSTRACT**

An animatronic face was created based on Professor Combs from UTSA’s Mechanical Engineering Program. A representation of the professor’s face was attached to a wooden frame containing servos to move the jaw and eyebrows. On the face was a passive infrared (PIR) sensors and a touch sensor that were connected to an arduino. These sensors would trigger the movement of the jaw and eyebrows as well as activate a speaker that played clips of the professor speaking.

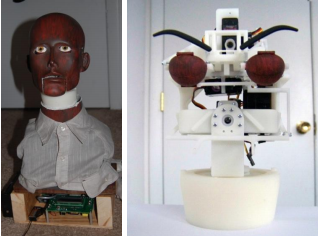
**Section 1: Literature review**

An animatronic face was created as shown in reference [1]. The instructions provided by the creator indicates that they used plywood for the base of the face, servos from model airplanes to control the facial expressions, a servo controller, a single-board computer to control the face and a compact flash card to store the program as well as a micro switch. The head had the capability to turn left and right as well as tilt up and down using two servos. Multiple servos were also used to allow the eyes to look up and down and to both sides. The single board computer was used to send inputs to the servo controller and to contain the audio used without having to have the face connected to a laptop or desktop. The creator programmed both the mouth movements and the facial expressions to coincide with the audio. The programming of the subtle movements that occur when talking such as the raising of eyebrows or the head tilting was found to be the more difficult part. The purpose of the animatronics was to talk about technology to younger kids. The animatronic was brought to multiple schools to show and talk to the children about technology.



[1] Animatronic Plywood Face

An animatronic face was created as shown in reference [3]. The instructions provided by the creators show that they used polystyrene mannequin head as the main face structure were several, different servo motors, motor controller, motor attachment rods, single board computer, power supplies, servo horn, actuators, atlanto-occipital joints, universal joints, mounting plates, springs, coupling rod and a cylindrical slot where used for the movement and control of the neck roll, neck pitch, neck yaw, eyebrows, eye pitch, eye yaw, eyelids and jaw. Where each feature in the animatronic face had different speeds, angles and torques. Various face-tracking softwares were used to make the face move like a human which were then converted into MATLAB then into Visual Show Automation for the final script.The purpose of the animatronics was to construct a humanoid animatronic head that had sufficient degrees of freedom to mimic human facial expression as well as human head movement and could be animated using face-tracking software to eliminate the amount of time spent on trial-and-error programming intrinsic in animatronics, and also have a human's voice.



[2] Animatronic Mannequin

An animatronic face was created as shown in reference [5]. The instructions provided by the creators show that they used a skull that was purchased from a halloween store. The skull has a hinged jaw so it would be easy to move the jaw up and down using an arduino. The material that was used to construct this project are an Arduino Duemilanove, Ladyada WaveShield Kit, two LED’s, speakers for sound, PVC and fittings for the movement of the jaw, and a PIR Sensor Module which allows you to sense motion, the maker used it to detect whether a human has moved in or out of the sensors range. Also a small servo was attached to the jaw of the skull to control the movement. To time the mouth movements with the audio the creator used the Arduino to read the voltage outputs. As the voltage increased the mouth opened and as the voltage dropped the mouth closed. The creator also connected the motion sensor and servos to a small project circuit board and the used CAT-5 network cable to connect this to the Arduino. This allowed for an adequate amount of wires to be available while still conserving space. The purpose of the animatronics was to create a talking skeleton decoration that could be used for halloween that could be programmed with phrases of the user’s choice.

[3]

The purpose of this project was to build an animatronic hand puppet robot [7] controlled with Raspberry Pi and hobby servos. An Alamode board interfaces with Raspberry Pi for synchronization of timing between the mouth and voice of the puppet. Due to the compatibility of the Alamode board with Arduino headers, a servo shield was used. This allowed arduino to be used to program mini speakers for a voice box. In terms of a chassis, a mechanical arm was used to operate the puppet. A problem arose when the original hobby servos lacked the torque required for the motion, so they were replaced with stronger models. An aluminum sheet with a thickness of 3 mm was cut to the desired geometry, acting as a bracket for the various servos. Next, acrylic plates were glued to the servos to act as the fingers driving a normal puppet. Finally, tape was wrapped extraneously to reinforce structure, and a cigar box was used for the puppets base.

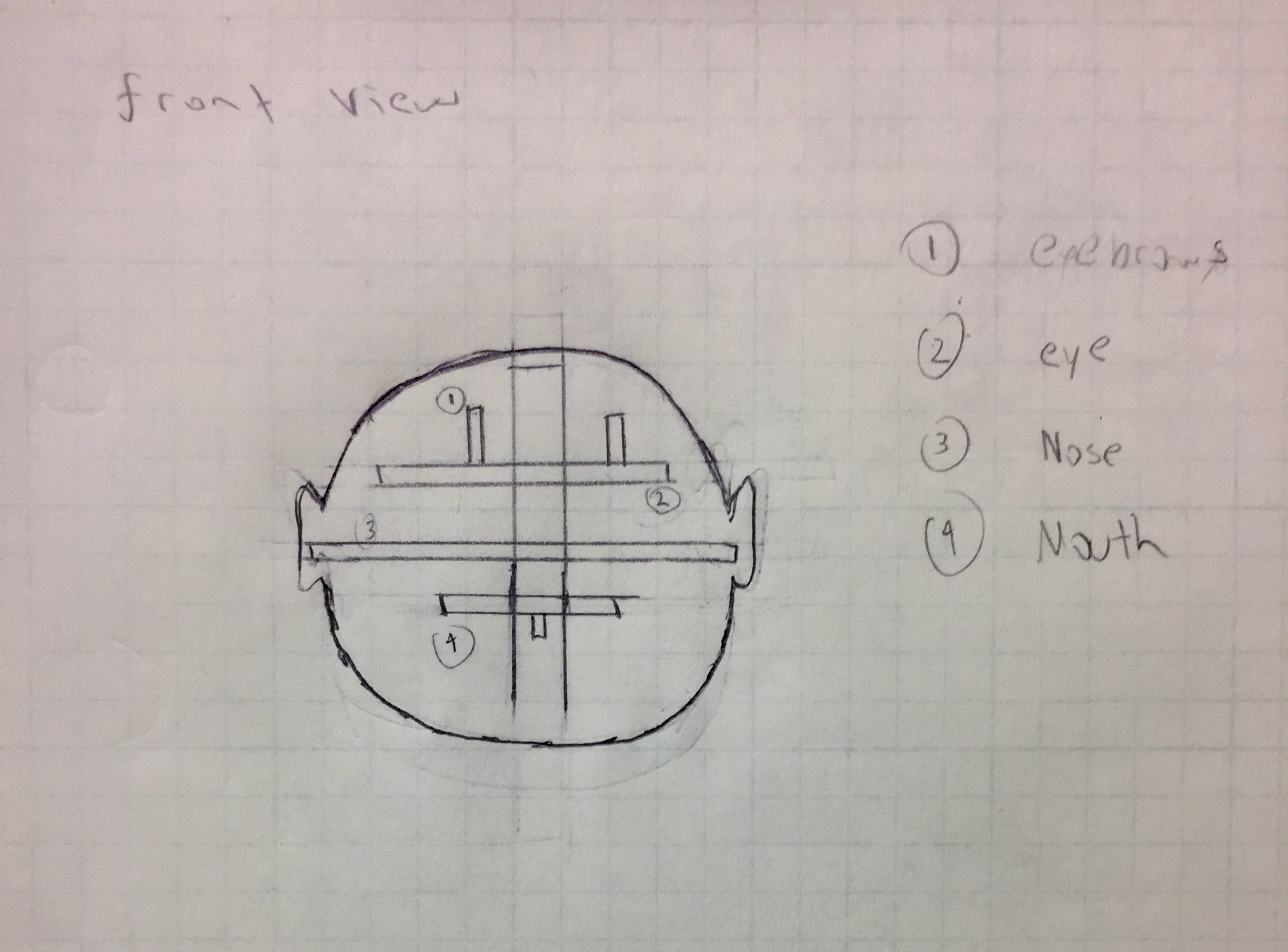
[4] Robopup

**Section 2: Brainstorming (initial planning)**

The purpose of the animatronic face was to represent the assigned professor with some added humor. The facial movements should match the phrases spoken by the animatronic.

For the overlay it was decided that a blank mask would be used. The mask would be cut apart to allow the parts to move. The jaw would be able to cut along the mouth line to allow the jaw to move. A high resolution picture of the professor would be printed and glued onto the mask. For the eyebrows a set of fake eyebrows which will move up and down which will add more character and dimension to the face. It was also decided to use LED to light up the animatronics eyes, we thought about making the eyes move but then decided against eye movement because of difficulty and lack of time. Other ideas for the facial features were considered but ultimately were decided against. Another idea that was considered was to 3D print the face but this idea was deemed to be too time intensive and expensive.

For the chassis two ideas were considered, an erector set or a cardboard base. The erector set has the advantage of being structurally sound. Since the pieces are made of metal they will be able to handle more weight and applied force than the cardboard, per inch thickness. There are also AC and DC powered motors that built specifically to work with the erector sets. However compared to the cardboard the erector set will cost more and there is not as much freedom in design since one must work with the provided lengths and shapes. The cardboard has the advantage of being cheap. It would be possible cut up old boxes to use for the base. This would also allow for more customizability in the pieces used. The pieces could be cut to whatever shape is needed. The disadvantages of the cardboard are that the cardboard is potentially less sturdy. It too much weight or force is applied the the cardboard could bend. There could also be issues with transportation if cardboard is used. Unlike the erector set which uses nuts and bolts the cardboard would be held together with glue or tape. These joints would be less sturdy and the process of moving the animatronic face could result in damage. The cardboard also would be more susceptible to water damage as it would damage the structural stability of the chassis. Ultimately it was decided that an erector set would be used. If shipping times permit it will be purchased online otherwise it will be purchased locally at either Interek or a local hobby shop.

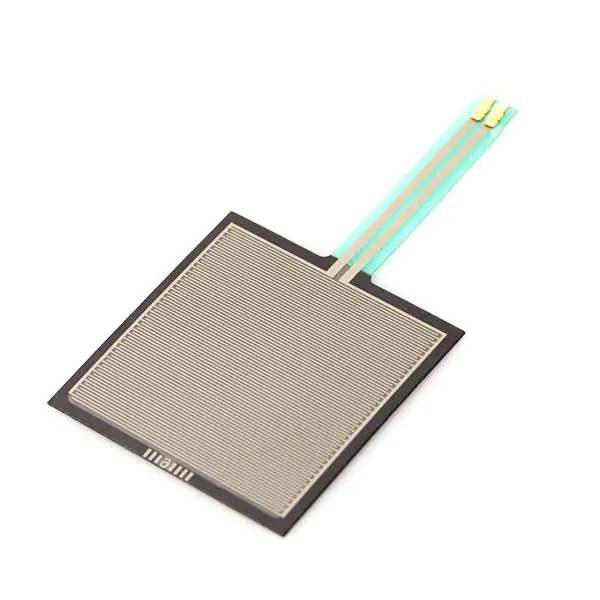
****

**Figure 1. Chassis Concept Sketch**

****

**Figure 2. Erector Set**

The next topic discussed was the sensors that are going to be used. The first idea for a sensor is a infrared motion detector sensor, the way this sensor works is it uses infrared a type of electromagnetic radiation to detect movement when anything passes in front of the animatronic face. Another idea is placing a pressure sensor or a touch sensor inside the nose, where it works by a person putting pressure on the nose which will then activate the face’s movements and audio. The sound is going to be emitted from a Adafruit Speaker 3 or mini speaker placed inside the mouth. The reason one of these speakers will be used is that they are small in size and have a very low weight and can be easily placed inside the mouth. The voice that is going to be emitted is recorded with different phones for different qualities and then an audio editing software is also going to be used to cut the phrases that are needed for the final soundtrack . Another idea was a sound sensor but we decided against it because any sound emitted by anything will activate it and it will not interact with the person when they want to.



**Figure 3. Infrared and Pressure Sensors**



**Figure 4. Adafruit Speaker 3**

Motor selection was the next topic discussed amongst the group. DC Motors are the most common type of electrical motors used. They work by sending a charge to various loops of a copper coils. The magnetic field of the coils reacts with the magnetic field of the the magnetics in the outer casing of the motor, creating a torque that rotated the coils. This coil rotation will generate the current that rotates the motors shaft, kept in a uniform direction by a commutator shaft. The DC brush motor is the oldest and most common motor, making it available and cheap. However, the brushes undergo wear and can over heat. The DC brushless motor is an improved form, however specialized control systems as well as a gearbox are necessary to regulate power control and precision. A stepper motor is another form of a DC Motor with a specialized control system. It is commonly found in advanced robotics projects. This motor works by providing an exact amount of charge to move the motor a set distance or “step”. The quantity of steps and speed at which they occur are used to fine tune control of the motor. The group decided against this motor due to its higher cost and the fact that the precision provided is not necessary for animatronics. A servo motor is renowned for its precise control and ease of use. A servo in itself is not just a motor, but a DC Motor with a few additional parts. It consists of a control system (positional feedback sensor and respective circuit) to provide the communication between the motor and shaft necessary for precision. Most servos are powered via the impulses sent from the Arduino microcontroller. These impulses sent are in the form of pulse width modulation, which is just a type of digital signal used by Arduino. This motor is commonly selected for a novice robotics projects due to its ease of programming. The biggest drawback of this motor is the power required for the necessary torque. The group considered this, and ultimately decided this would be easily provided by a few batteries, possibly connected in series and also keep the laptop connected to the arduino for more power. The Servos medium balance of cost, precision and ease of use made this the optimal choice for this project.

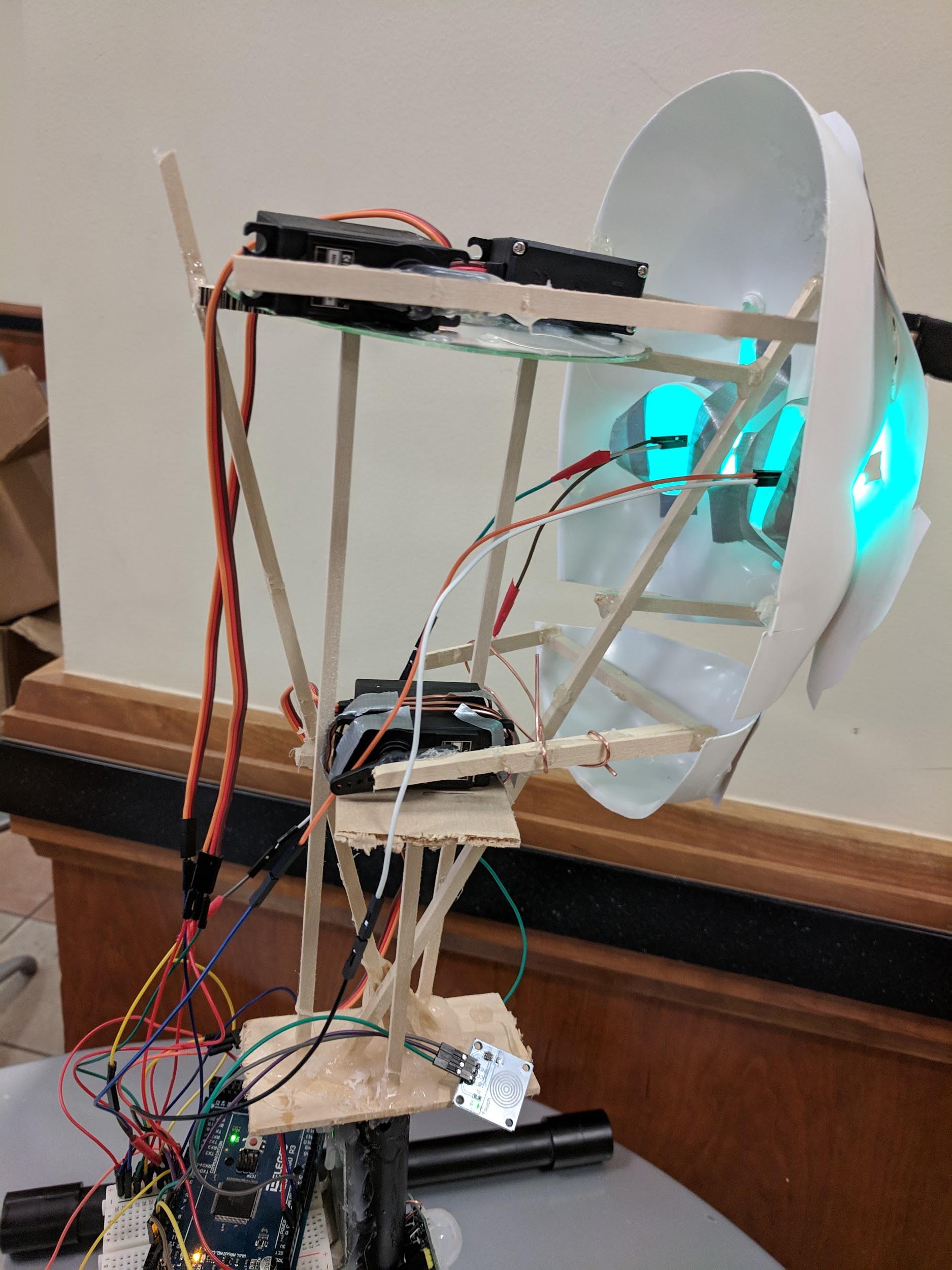
****

**Figure 5. Servo Motor**

**Section 3: Supporting structure**

The first attempt of the supporting structure was created out of plastic tile trim however the trim proved difficult to work with. Instead strips of unfinished wood from a craft store was used which was considerably cheaper per piece and easier to work with. A plastic mask was used to create the features of the face. The jaw was cut off that way it could be moved separate from the rest of the face. Slits were created above the eye holes to allow the movement of the eyebrows. Since the eyebrows were to be one of the moving parts of the face they needed to be created separately. They were created out of craft foam hot glued to thin balsa wood. The main form of the face was attached a wooden frame created out the unfinished wood strips. The strips were glued into the form of a rectangular prism and the mask was then attached to this prism. A second rectangular prism was then attached to the original in order to allow a place to attach the motor that would control the jaw movement. In both of the prisms a wider piece of unfinished wood was attached to hold the motors. The jaw itself just had a thin piece of wood horizontally across to give stability and a single piece of wood attached perpendicular to the center of the jaw which would then attach to the motor.

The entire face structure was then attached with hot glue to a pvc frame that will act as the shoulders and a location to hold the arduino.

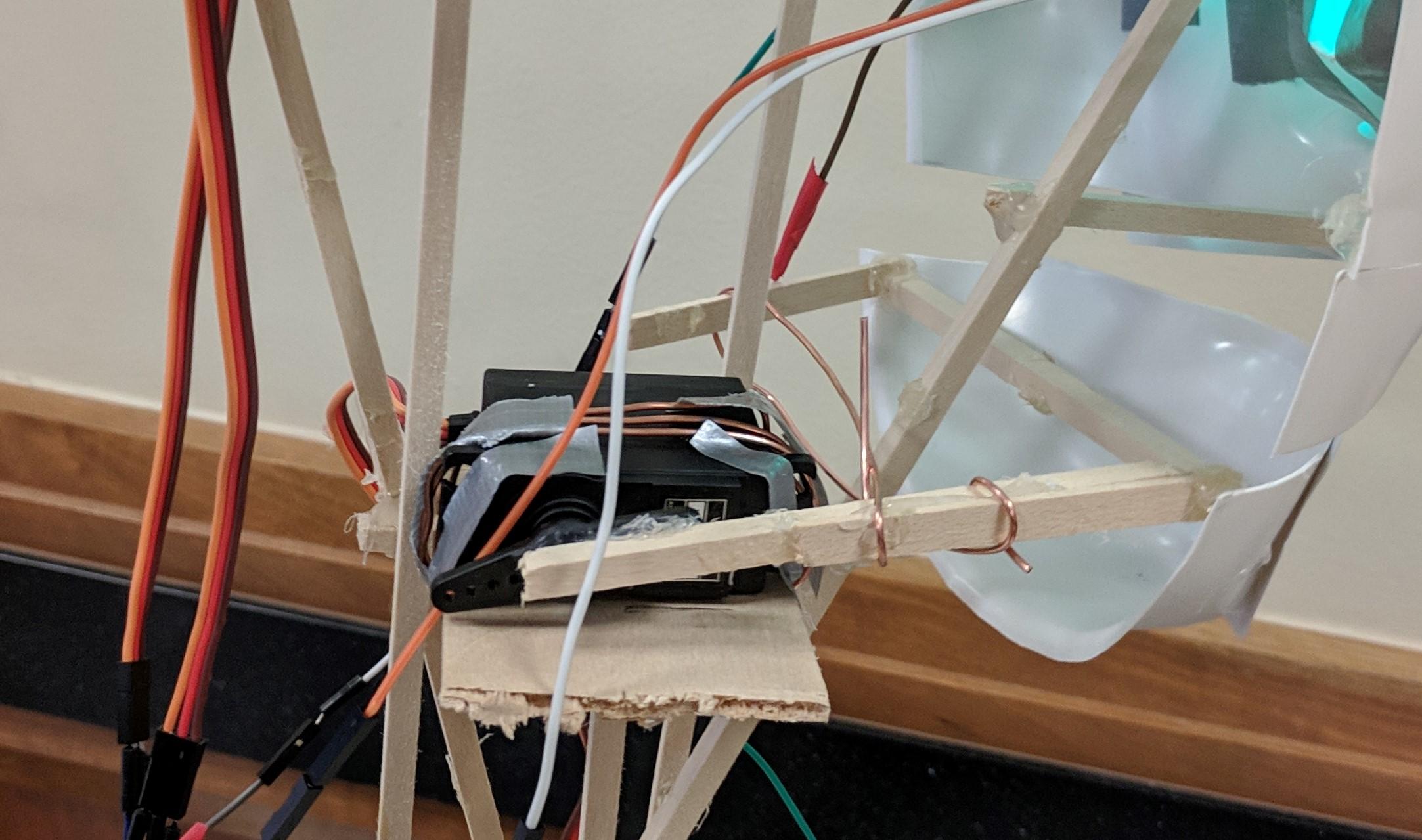


**Section 4: Joints and motors**

The eyebrows were one of the features chosen to be moving parts. The joints used were simple, created from the motors and the wood strips the eyebrows were attached to. Each eyebrow was glued to a wooded strip. This strip was then attached to a 180 degree rotation servo motor which sits on a wooden platform at the back of the supporting structure. As the servos rotate the eyebrows move up and down.

The jaw was also a created using a simple joint similar to the eyebrows. The center piece of wood from the jaw’s supporting structure was attached directly to the servo so that it would move with servo.

Both the eyebrows and the jaw use the same type of 180 degree rotation servo motor. The motors were advertised for use with RC cars, airplanes, and helicopters. Since the joints were to move through various degrees of rotation a servo that had a large angle of rotation was chosen. These servos also did not draw large amounts of power thus minimizing the amount of additional voltage that needed to be supplied. Parts such as the eyebrows could be powered solely through that arduino.



**Section 5: Sensors**

One of the sensors used a PIR ( passive infrared) sensor. The sensor uses infrared radiation to detect movement in front of the sensor. As a passive sensor it does not generate radiation to use for detection, instead it picks up infrared generated or reflected by objects in front of it. The sensors picks up changes in infrared radiation in front of it and converts this into a output voltage change thus triggering detection. The sensor consists of a thin layer of film of material, often gallium nitride (GaN), caesium nitrate (CsNO3), or polyvinyl fluorides. A translucent plastic covers the sensor. This plastic cover is transparent to infrared radiation and helps reduce interference from foreign objects such as dust which could obscure the sensors view or damage it. The PIR sensor was chosen because it would be easy to set off and we could control what was located in front of it, thus diminishing the risk of it accidentally triggering.

The other sensor used was a capacitive touch sensor/switch. The sensor uses the body capacitance, a property of the human body that allows it to act as a capacitor. The sensors charges and discharges its metal exterior. When a person touches the sensor they increase the capacitance which in turn triggers the switch. The sensor chosen stays at a low output until it is touched then switches to a high output. If it is not touch again within 12 seconds the sensor then switches back to low output. This sensor was chosen because it would have a low chance of interference from unwanted sources.

The sensors were placed under the face in the front so that they could be triggered by a person passing in front of the face.

**Section 6: Programming for interaction**

In the code for the face the sensors were named as inputs and the pins they were attached to defined. The servos and the LEDS were the named as outputs and their pins defined as well. The initial angles of the servos were also given. The code was written so that when the PIR sensor’s output was high and this detecting motion or touch the LEDs would turn on and the servos would begin moving. The code also defined the angles the servos would move through and how for how long they would remain moving and the LEDs would remain lit.

For the touch sensor the code servos would not move if the state was high. Once the sensor was touched and the input changed to low the LEDs would turn on and the servos would move through angles defined in the code.

In both the section of the code for the PIR sensor and the touch sensors there are delay that give allow the movement to continue for the same amount of time as the recorded audio.

[5] <https://youtu.be/1INbbDA13yo>

**Section 7: Lessons learnt and suggestions**

1. Easier To Work With Face/Structure Materials - While the mask gave a more realistic shape to the face the curves and angles created difficulties. Trying to get the frame to fit into the mask proved challenging at times because the areas where it would connect were curved. To work around this problem the mask was attached using hot glue. Any gaps that resulted from incorrect measurements or difficult fit could be filled in with the glue. Care had to be taken with the hot glue, since it was a high temperature variant it could melt the mask if one was not careful.
2. Difficulty Working With Materials- The first attempt at the frame was made using plastic tile trim and super glue. The trim proved difficult to cut down to the length needed. Scissors could not cut through it and any cuts that were made were jagged. The width of the trim also made it hard to work with. It ended up being too wide for certain parts of the frame. There was also an issue with joining the materials. The super glue would not join the tile trim pieces. Even when the pieces where held together for 10 minutes they would still fall apart once the pressure was removed. To solve this both the frame material and the type of glue was changed. Instead of tile trim thin unfinished wood pieces were used. These were made of a light wood that was easier to cut and was available in many different widths, lengths, and thicknesses. The wood could be cut with simple kitchen scissors making construction easier. Instead of super glue, hot glue was used to join the frame pieces. The hot glue proved to be sturdy and had no issues bonding the wooden frame pieces. The hot glue also could be used to fill in any gaps that occured from measurement errors thus allowing the group to minimise the amount of recutting needed. The use of hot glue also allowed for the frame to modified as needed since the glue could be either cut through or reheated until it softened.

Improvement Suggestions

1. Low Temperature Hot Glue and Hot Glue Gun - The hot glue used was high temperature variant. If care was not taken this would melt the mask and had the risk of burns to the person assembling the frame. For a future attempt a low temperature hot glue should be used. The low temperature hot glue melts at a temperature of 248F instead of the 400F of the high temperature hot glue. The lower temperature will result in less severe burns and and a lower chance of melting the construction materials
2. Flat Face- While the mask gave a more realistic 3D look to the face it was difficult to attach the frame to it since one had to take into account the curves. It was also more difficult to attach the picture of the professor’s face since the 2D pic had to be made to fit on a 3D object. The picture also had to be resized to match the dimension of the mask with proved challenged. For a future attempt at the face a flat representation of the professor’s face would be used. The picture of the teaches face could be placed on a piece of cardboard cut to match the face’s shape. This would also simplify the movement of the features since the curves and dimensions of the mask would not need to be taken into account when programming the features range of movement.
3. PIR Sensor Sensitivity- The sensor was too sensitive and would be set off by any movement even if it was far from the face. For a future version of the face the sensitivity would be adjusted so as to not be so easy to set off.

**Section 8: Personnel and bill of materials**

**(a) Personnel**

|  |  |  |
| --- | --- | --- |
| Task | Main Personnel | Secondary personnel |
| Structure/Chassis Design | Meggan | Brenton |
| Joints/Motor interfacing | Brenton | Meggan |
| Sensor integration/Interfacing sensors | Haitham | Brenton/Meggan |
| Programming and Integration | Haitham | Brenton |

**(b) Bill of materials**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Description | Website/comment | Qty. | Unit $ | Total $ |
| 1 | Arduino MEGA 2560 | Provided | 1 |  |  |
| 2 | 180 Degree Rotation Servo Motor (2 Pack) | http://a.co/d/hu7UeKk | 2 | 14.99 | 29.98 |
| 3 | Craft Foam 9”x12” | Joann Fabrics | 1 | 1.79 | 1.79 |
| 4 | Mask | Joann Fabrics | 1 | 5.99 | 5.99 |
| 5 | Unfinished Wood 3/16 x 3/16 x 24” | Joann Fabrics | 6 | 1.34 | 8.04 |
| 6 | Unfinished Wood 1/16 x ¼ x 24” | Joann Fabrics | 2 | 0.96 | 1.92 |
| 7 | Unfinished Wood ⅛ x 2 x 24” | Joann Fabrics | 1 | 2.84 | 2.84 |
| 8 | PIR Motion Sensor 5 pack | Amazon | 1 | 8.99 | 8.99 |
| 9 | Touch Sensor 5 pack | Amazon | 1 | 6.99 | 6.99 |
| 10 | Arduino Power Supply | Amazon | 1 | 6.99 | 6.99 |
| 11 | PVC | Already Owned | 2 |  |  |
| 12 | Hot Glue Gun/Glue | Already Owned | 1 |  |  |

The total price for this project excluding the Arduino MEGA was **$73.5.**

**References:**

[1] Wooden Animatronic Robot Head

<https://www.instructables.com/id/Talking-Animatronic-Robot-Head/>

[2] Animatronic Skeleton

<https://web.wpi.edu/Pubs/ETD/Available/etd-050112-072212/unrestricted/Fitzpatrick.pdf>

[3] Arduino Halloween Skull

<https://www.instructables.com/id/Talking-Arduino-Halloween-Skeleton/>

[4] Robopup

https://www.hackster.io/breakpointer/robopup-robotic-hand-puppet-b47b7

[5] Video of Project

<https://youtu.be/1INbbDA13yo>

**Appendix A: Code**

#define ctsPin 12

#include <Servo.h>

// defining servos

Servo servomouth;

Servo servorighteyebrow;

Servo servolefteyebrow;

// led pins attached

int led1 = 7; // led pins attached

int led2 = 8;

int PIR = 13; // motion sensor pin attached

int state = LOW;

int val = 0;

void setup() {

Serial.begin(9600); // initialize serial

pinMode(led1, OUTPUT); // initalize LED as an output

pinMode(led2, OUTPUT);

pinMode(PIR, INPUT); // initialize sensor as an input

pinMode(ctsPin, INPUT);

//attaches servos to pin

servomouth.attach(3);

servorighteyebrow.attach(6);

servolefteyebrow.attach(9);

//initial position of servos

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

}

void loop() {

val = digitalRead(PIR); // read sensor value

if (val == HIGH) { // check if the sensor is HIGH

digitalWrite(led1, HIGH); // turn LED ON

digitalWrite(led2, HIGH);

delay(100); // delay 100 milliseconds

// turns servo from 0 to angles required and back

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(250);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(250);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(250);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(250);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(250);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(250);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(250);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(250);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(250);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(250);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(250);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(250);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(250);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(250);

if (state == LOW) {

Serial.println("Motion detected!");

state = HIGH; // update variable state to HIGH

}

}

// Touch Sensor Starts

else {

digitalWrite(led1, LOW); // turn LED OFF

digitalWrite(led2, LOW);

delay(200); // delay 200 milliseconds

if (state == HIGH){

Serial.println("Motion stopped!");

state = LOW; // update variable state to LOW

delay(5000);

}

}

{

int ctsValue = digitalRead(ctsPin);

if (ctsValue == HIGH)

{

digitalWrite(led1, HIGH);

digitalWrite(led2, HIGH);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(500);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(500);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(500);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(500);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(500);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(500);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(500);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(500);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(500);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(500);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(500);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(500);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(500);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(500);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(500);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(500);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(500);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(500);

servomouth.write(22);

servorighteyebrow.write(165);

servolefteyebrow.write(25);

delay(500);

servomouth.write(32);

servorighteyebrow.write(180);

servolefteyebrow.write(10);

delay(500);