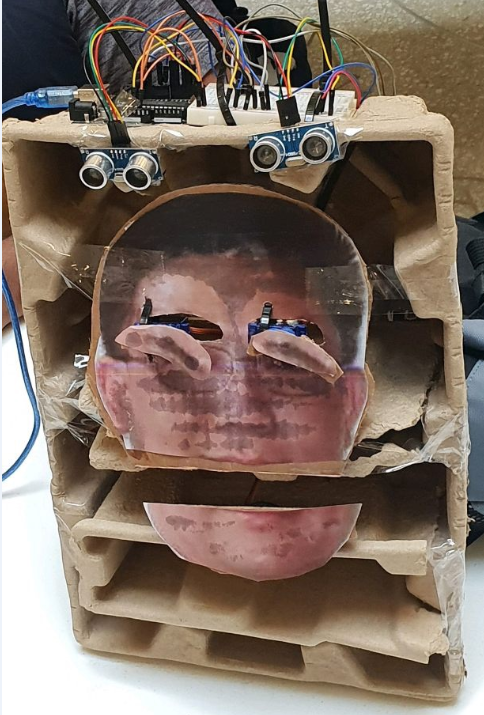
ANIMATRONICS FACE: PROF Feng Yusheng



Maximilian Pittman

Dept. of Mechanical Engineering

San Antonio, TX, USA 78249

[maxpittman1031@gmail.com](mailto:maxpittman1031@gmail.com)

Abdullah AI-Adwani

Dept. of Mechanical Engineering

San Antonio, TX, USA 78249

[abdullahaladwani@yahoo.com](mailto:abdullahaladwaniq8@gmail.com)

Varghese George

Dept. of Mechanical Engineering

San Antonio, TX, USA 78249

[Joemoelarry@gmail.com](mailto:Joemoelarry@gmail.com)

Submitted as part of project for ME4543 Mechatronics, Fall 2018

**TABLE OF CONTENTS**

|  |  |
| --- | --- |
|  |  |
| Abstract | 3 |
| Literature Review | 4 |
| Brainstorming (initial planning) | 6 |
| Supporting Structure | 6 |
| Joints and Motors | 7 |
| Sensors | 7 |
| Programming | 8 |
| Lesson learnt and suggestions | 8 |
| Personnel and Bill of Materials | 9 |
| References | 10 |
| Appendix A: Code snippets | 11 |

**ABSTRACT**

The animatronics project our team was tasked with was pretty challenging. The project was constrained to have, at the very minimum, two different sensors that serve as the stimulus, as well as two moving parts on the animatronic face. The two sensors that the team has chosen are ultrasonic sensors and a push-button. The ultrasonic sensors use sound to detect objects. It works when sound is ejected from an emitter, bounces off the object, and comes back to a receiver. The time of flight of sound is used to gauge how far an object is located. The push-button sensor gets activated by manual touch. In order to make this project a success, the team decided to go with two ultrasonic sensors, where each sensor serves as the stimulus to independently move micro-servo motors that control the eyebrows, and one push-button sensor that, when manually pushed, serves as the stimulus to move the mouth through the same process. The team has satisfactorily completed all tasks and have managed to meet all endpoints indicated in the project guideline.

**Section 1: Literature review**

The animatronic face I researched was for a summer camp to teach kids on how to design and build a working face and costume. The project consisted on moving eyes and a moving mouth that was mounted on a frame that a user could wear. The builder used a 3D printer that was able to print parts for the eyeballs and eye mechanism out of plastic. The rest of the frame and other moving mechanisms consisted of hand built parts out of card board and wood. Moving mechanism only consisted of the two eyes and the mouth. Those used three PDI-HV6214MG servos that were configured in such a way that could move the eyes in 3 degrees of motion while the mouth could only move up and down. A nodemcu microcontroller was used to interface all of the moving parts. All of those components worked together to make the face seen in figure 1.



Figure 1: First animatronics face researched

The robot face project I researched utilized an animatronic face that had the capability to pan the neck left and right, and to tilt the neck up and down. The face was made out of two cardboard pieces with several cutouts for displaying other features. The first cardboard piece fashioned the eyebrows, eyes, and nose, while the other cardboard piece was that of the mandible (lower jaw). The first cardboard was stationary, while the second cardboard was mobile (with up and down movement) to help emulate human speech. The eyes were made out of ping-pong balls that employed a 2-axis gimbal system along with other structural components like thin cross-sections of PVC pipes. The whole face was placed on top of a servo that allowed panning of the neck from left to right, with another really strong servo anchored to a point of the face that allowed the up and down tilting of the neck. There was a total of seven servos, with each having a specific function. The first servo controlled the left and right panning of the neck, as mentioned above. The second servo controlled the up and down tilting of the neck. The third and fourth servos controlled the left and right eyebrows, respectively. The fifth and sixth servos controlled the left and right eyes, respectively. The seventh servo controlled the mandible of the face. One ultrasonic sensor (HC-SR04) was utilized to detect a stimulus and to help start a cascade of other responses. The first and second servos utilized were HS-485HB and HS-645MG, respectively. All the other servos used were HS-55 Sub-micro servos. A 7805 +5V power supply regulator was used to prevent the voltage from going beyond 5V. A PIC 18F452 and PICkit 2 was used to control the coordinated actuation of all servos and sensors. This can be seen in figure 2.

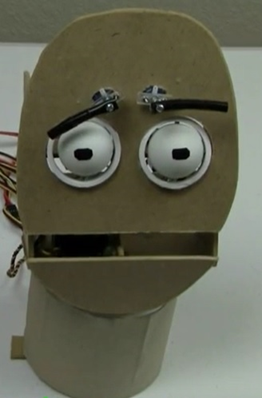


Figure 2: Second animatronics face researched

The animatronic project I researched about was named “The Yorick Project”, which was a 3-axis talking skull using Amazon' Alexa as its voice via AlexaPi software and Raspberry Pi. The main hardware components used to construct this project was Raspberry Pi 3 Model B, Pololu Maestro Servo Controller, Lindbergh 3-Axis Skull, USB 2.0 Mini Microphone, and Audio Servo Driver Board. As for the software apps and online services, they used AlexaPi, Amazon Alexa Alexa Voice Service, and Pololu Maestro Controller Software. These software programs are used for turning the Pi into an Alexa client device, while the Audio Servo Controller is used to turn the output command into the skull's jaw. Other servos are used to control the skull's nod, turn, tilt, and eye movements. Installed LEDs will light up when an output is provided from AlexaPi software to 2 GPIO pins and as a result, the skull will respond via the Amazon Alexa Service.The Pololu controller software is used to develop the servo sequences, which in turn commands the skull to perform certain movements such as tilting, nodding, and eye movement. This can be seen in figure 3



Figure 3: Third and final face researched animatronics face

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

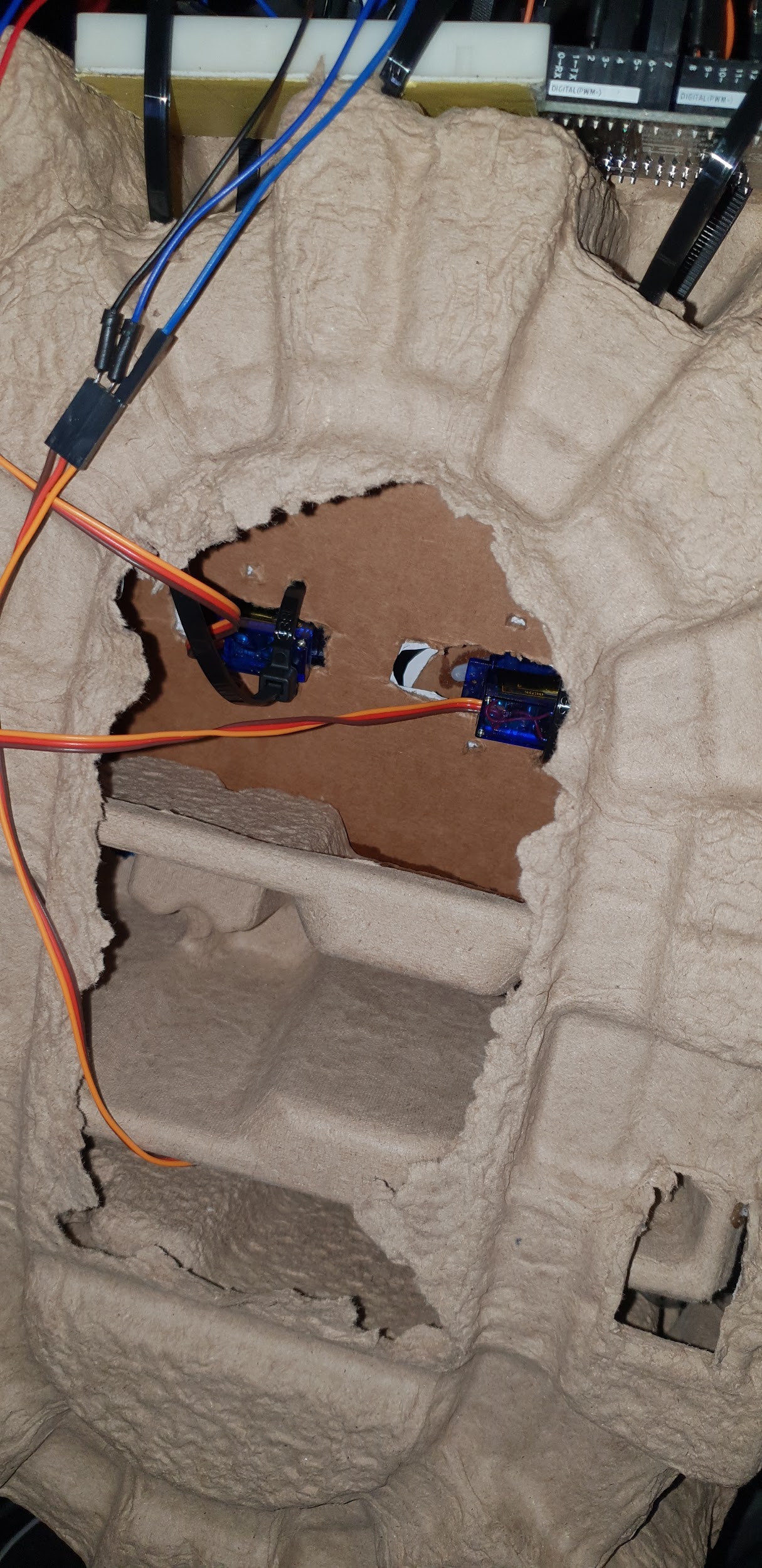
Our assigned professor for this project is Professor Feng. The structure of the head will be made out of cardboard. A picture of Dr. Feng will be pasted on the cardboard structure with other separate parts that will represent the eyebrows and nose. The mouth, both eyebrows, and the nose will move with a soundtrack of our selected professor. This will be done using micro-servos. A motion sensor, possibly an ultrasonic sensor, will be used as the stimulus. Sound will be projected with the help of a mp3 shield and will be synchronized with other movements. The coordination of all parts will be done with the help of an Arduino Mega2560 that will be powered by a laptop. The team plans to have the professor say phrases like “You will fail heat transfer”, and “You shall not pass heat transfer”.

**Section 3: Supporting structure**

The supporting structure was very simple. It consisted of nothing more than cardboard that was once used for lining within a computer box. This was chosen because of how inexpensive it was. We also cut two pieces from the same card board for stabilizing each sensor, motor, and structure for the face. Zip-ties were also used to help anchor the micro-servos to the main chassis. The face was glued onto a different piece of cardboard that was fashioned onto the the two cut out pieces.

**Section 4: Joints and motors**

As a group, we decided to create three joints; one for the mouth to make it speak and two others for the eyebrows to give the illusion of different emotions. For the mouth, we glued the servo motor arm to the cardboard mouth and modified the code in which the servo arm would only rotate 60 degrees on each direction. For the eyebrows, we glued the micro-servos to make-shift eyebrows and modified the code in which the servo arm would only rotate 180 degrees in each direction. These micro-servos were anchored to the main chassis by utilizing zip-ties.



**Section 5: Sensors**

The two sensors that were used were two ultrasonic sensors and one button. The two ultrasonic sensors were used to control the eyebrows while the button was used to control the mouth movement. Each sensor was decided upon for a couple of reasons. Both ultrasonic were picked because of how easy they are for coding. Ultrasonic sensors send out a high frequency wave out one opening while the second opening receives the echo. Figure 4 shows a ultrasonic sensor with all four pins.



Figure 4: Ultrasonic Sensor

So when something that is within a predesigned distances is sensed then the ultrasonic sensor does the function it was programmed to do. The function is to have someone be able to walk up to the animatronic face and cause it to react. The button was picked for simplicity. We had a lab that dealt with how they work and how to code them. The button works in a very understandable and simple fashion. When the button is not pressed there is no connection. On the contrary, when the button is pressed, connection is restored. This activates the programmed function. Figure 5 shows a button.



Figure 5: close up of a button that was used.

**Section 6: Programming for interaction**

At the very beginning, we faced some complications with the programing aspect of this project, as none of us have had any sufficient background in coding. It took us a while after extensive research and watching youtube tutorials to grasp enough knowledge to start coding. We modified the code in such a way that the servo arm intended for the mouth part would only rotate 60 degrees in each direction, while keeping the two servos for the eyebrows at 180 degrees. Here is a youtube video of our mechatronic face in action.

<http://www.youtube.com/watch?v=59Nr_JY9t10&feature=youtu.be>

**Section 7: Lessons learnt and suggestions**

We have learned a lot as we worked on this project. For starters none of us had great knowledge in programming so it was a challenge for us to write the code and figure out how to modify it in such a way that it meets our requirements. Another thing we learned was the vast list of sensors that can be used in arduino and the correct way of wiring those sensors on to arduino. We faced some complications with wiring all the servos and trying to connect them with their intended sensors, but we managed to do it and now have a better understanding on this issue which will help us greatly if we ever faced it again in the future. Watching youtube tutorial videos relating to arduino definitely helped us out in creating this project, as it was much easier to understand it by watching it rather than read about it. A suggestion that could be used to improve would have been to allocate more time to the project itself. It was hard to find the necessary time to work on this project. Reason being all group members had to spend time on other project we were working on such as senior design. This fact made it hard for the group as a whole to meet up. Another issue that we faced was trying to meet professor Feng to acquire the picture of the face and the voice that we needed. Something that could have been done to improve this was to meet with him earlier. He might have been available earlier in the year rather than the time that we tried to meet with him.

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

**(a) Personnel**

|  |  |  |
| --- | --- | --- |
| Task | Main Personnel | Secondary personnel |
| Concept | All |  |
| Chassis | Abdullah | Max, Varghese |
| Button | Abdullah | Max, Varghese |
| Ultrasonic sensor | Max | Abdullah, Varghese |
| MP3 voice integration | Varghese | Abdullah, Max |
| Integrating sensors | All |  |
| Overall programming and integration | All |  |

**(b) Bill of materials**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No. | Description | Website/comment | Qty. | Unit $ | Total $ |
| 1 | Arduino MEGA 2560 | Provided | 1 |  |  |
| 2 | Small stepper motor | <https://www.sparkfun.com/products/10551> Provided | 3 |  |  |
| 3 | cardboard | Scrap |  |  |  |
| 4 | Mini Metal Speaker |  |  |  |  |
| 5 | Micro SD card | Amazon | 1 | 5.99 | 5.99 |
| 6 | Tape | UTSA bookstore | 1 | 2.95 | 2.95 |

**References:**

# [1] Figure one

# <https://hackaday.io/project/16193-open-animatronics#menu-description>

# [2] Figure two

<http://www.pyroelectro.com/tutorials/animatronic_neck/>

[3] Figure three

<https://www.hackster.io/mike-mcgurrin/the-yorick-project-f9bac7>

[4] Video

<http://www.youtube.com/watch?v=59Nr_JY9t10&feature=youtu.be>

**Appendix A: Code**

#include <Servo.h>

#define trigPin 7

#define echoPin 6

#define trigPin1 4

#define echoPin1 3

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

Servo myservo1;

Servo servo;

int angle =90; // initial angle for servo

int angleStep =10;

int sound = 250;

void setup() {

// Servo button demo by Robojax.com

Serial.begin(9600); // setup serial

myservo.attach(9);

myservo1.attach(12);

pinMode(2,INPUT\_PULLUP);

Serial.println("Robojax Servo Button ");

Serial.begin (9600);

pinMode(trigPin, OUTPUT);

pinMode(echoPin, INPUT);

pinMode(trigPin1, OUTPUT);

pinMode(echoPin1,INPUT);

servo.attach(8);

}

void loop() {

while(digitalRead(2) == LOW){

// change the angle for next time through the loop:

angle = angle + angleStep;

// reverse the direction of the moving at the ends of the angle:

if (angle <= 60 || angle >= 120 ) {

angleStep = -angleStep;

}

myservo.write(angle); // move the servo to desired angle

Serial.print("Moved to: ");

Serial.print(angle); // print the angle

Serial.println(" degree");

delay(50); // waits for the servo to get there

}// while

long duration, distance;

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

duration = pulseIn(echoPin, HIGH);

distance = (duration/2) / 29.1;

if (distance < 5) {

Serial.println("the distance is less than 5");

servo.write(90);

}

else {

servo.write(0);

}

if (distance > 60 || distance <= 0){

Serial.println("The distance is more than 60");

}

else {

Serial.print(distance);

Serial.println(" cm");

}

delay(50);

digitalWrite(trigPin1, LOW);

delayMicroseconds(2);

digitalWrite(trigPin1, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin1, LOW);

duration = pulseIn(echoPin1, HIGH);

distance = (duration/2) / 29.1;

if (distance < 5) {

Serial.println("the distance is less than 5");

myservo1.write(90);

}

else {

myservo1.write(0);

}

if (distance > 60 || distance <= 0){

Serial.println("The distance is more than 60");

}

else {

Serial.print(distance);

Serial.println(" cm");

}

delay(50);

}