Assistive Communication Surface
A DEVICE TO GAIN ATTENTION OF HEARING OR VISUALLY IMPAIRED CHILDREN

By
Kyle Smith and Amith Vijaykumar
For Assignment #1 in ARCH/ECE 868
Abstract

In this project we present a novel idea for a device which helps to gain the attention of a hearing or visually impaired student discretely in a classroom. The device contains a flexible surface and a few LEDs which can be activated by the teacher through a remote control. It is fitted flush with the desk and is inconspicuous. The device can be configured to respond in different ways depending on the need. It currently uses three different motions to gain the attention of the student and to convey a message. It is also capable of being removed from the desk if required and carried by the student to a different area. This device aims to help the student participate in the class as any other student would. It also helps the teacher communicate discretely with the student and avoid any unnecessary disturbances to other students.

Scenario

Madeleine is a sixth grade student who has a hearing impairment. Many times she won’t know when to look at the black board when she’s reading something on her desk. She will either have to constantly keep track of what the teacher is doing or miss something important. Currently the teacher has to ask Madeleine's neighbor to get her attention which makes Mary feel left out as well as cause disturbance to the other students. After the school received the assistive communication surface, Madeleine has been able to participate more fully in class activities and feels like a part of the class. The teacher also does not have to depend on anyone else to get Madeleine’s attention and can communicate discretely without disturbing others. Madeleine can concentrate on reading her material without keeping a track of what the teacher is doing because the whole surface of the device moves and gets her attention if needed.
Operation

The device consists of a silicone surface which is stretched over a plexiglass skeleton. The skeleton houses the arduino, two servo motors and two LEDs. The motors have arms attached to them which deform the silicone surface when they rotate. It is also fitted with an IR receiver to receive and decode the signals received from the remote control. This is done through a library created for this purpose.\(^1\) The motors and LEDs are programmed for three different operations as follows when the button indicated is pressed:

1. **Button 1** – To gain only the attention of the student.
   The two motors run in the same direction and deform the surface to one side and the red LED comes on.
2. **Button 2** – To gain the attention of the whole class
   The two motors run in opposite directions and twist the surface and the blue LED comes on.
3. **Button 3** – To indicate an emergency
   The LEDs start flashing alternatively and the surface keeps moving till it is stopped.

List of Electronics

1. Arduino Duemilanove board x 1
2. Futuba servo motors x 2
3. 10mm LEDs x 2
4. Connecting wires
5. Breadboard
6. TSOP4383 IR receiver x 1

List of other hardware

7. Silicone - *Dragon Skin® 10 Medium*
8. Assorted Screws
9. Aluminum bars x 2
10. #8 nuts and bolts
11. #8 threaded rods
12. #8, 1- inch plastic spacers
13. 3/32” Plexiglass

Construction

The frame of the device is made from laser cut plexiglass to keep it light so that it can be easily carried around if needed. The skeleton was put together from the individually cut parts as shown. The silicone surface was fabricated by pouring liquid silicone in a mold and curing it. It was then

stretched over the surface and held in place by a strip of plexiglass and screws. The arms were put together and fitted to the motor to give them more surface area to deform the silicone. The LEDs were fitted to the skeleton pointing upwards so that the silicon surface diffuses the light and is easily seen.

Fig 1. Plexiglass cut outs made from a laser cutter.

Fig 2. Silicone surface stretched over the top of the device.
Electrical Connections

The servo motors are connected to the PWM pins 9 and 10 of the Arduino and the LEDs are connected to digital pins 2 and 3. The IR receiver is connected to pin 11 as mentioned in the library. All the other standard connections for VCC and ground are made for the motors, the IR receiver and the LEDs.

Future work

The device could be modified to work with a Bluetooth receiver instead of IR as it is not limited to line of sight operation. It could then be controlled by any Bluetooth capable mobile phone. It could also be modified to be controlled through voice allowing for a true teacher-student interaction. For example, when the teacher says the word “class”, the device could be triggered. It could also listen to the schools PA system to know about emergencies.

Results

The device works as intended and responds to the different buttons pressed on the remote control from at least fifteen feet away. Some of the parts like the arms went through several iterations. A servo motor with a better torque rating could be used to provide smoother deformation of the surface as well as being able to work with heavier loads.

http://www.youtube.com/watch?v=aD3WQCtiur8
Code

```cpp
#include <IRremote.h>
#include <Servo.h>

Servo servo1; // create servo object to control a servo
Servo servo2;

int restPos1=83; // Initial position of the servo motors
int restPos2=83;

int highPos = 45; // Maximum angle which the servo should rotate
int pos1=25; // Intermediate angle
int pos2=0;

int time1=20; //Delay between each servo movement
int time2=5000; //Time for which the servos should keep their positions

int redLed = 3; // pin connected to Red LED
int blueLed = 2; // pin connected to Blue LED
int RECV_PIN = 11; // pin connected to IR receiver

//Codes for different buttons on the remote
int button1 = 1168;
int button2 = 3216;
int button3 = 144;
int button4 = 2192;

IRrecv irrecv(RECV_PIN);
decode_results results;

void setup()
{
    //Initialization of the different pins
    servo1.attach(9);
    servo2.attach(10);
    pinMode(redLed, OUTPUT);
    pinMode(blueLed, OUTPUT);

    Serial.begin(9600);

    irrecv.enableIRIn(); // Start the receiver
}

void loop()
{
    //Get the servos to initial position and turn off the LEDs
    servo1.write(restPos1);
    servo2.write(restPos2);
    digitalWrite(redLed, LOW);
    digitalWrite(blueLed, LOW);

    //Monitor the IR receiver and execute corresponding function
}
```

if (irrecv.decode(&results)) {
    Serial.println(results.value);

    if (results.value == button1) {
        individualAtt();
    } else if (results.value == button2) {
        emergency();
    } else if (results.value == button3) {
        classAtt();
    }

    else {
        servo1.write(restPos1);
        servo2.write(restPos2);
        digitalWrite(redLed, LOW);
        digitalWrite(blueLed, LOW);
    }

    irrecv.resume(); // Receive the next value
}
}

void individualAtt() {
    digitalWrite(redLed, HIGH);
    digitalWrite(blueLed, LOW);

    for (int j=0; j<3; j++) {
        for (int i=0; i<=highPos; i++) {
            servo1.write(restPos1+i);
            servo2.write(restPos2-i);
            delay(time1);
        }

        for (int i=highPos; i>=0; i--) {
            servo1.write(restPos1+i);
            servo2.write(restPos2-i);
            delay(time1);
        }
    }
}
for(int i=0;i<=pos1;i++)
{
    servo1.write(restPos1+i);
    servo2.write(restPos2-i);
    delay(time1);
}

delay(time2);

for(int i=pos1;i>=0;i--)
{
    servo1.write(restPos1+i);
    servo2.write(restPos2-i);
    delay(time1);
}

void emergency()
{
    for(int i=0;i<20;i++)
    {
        digitalWrite(redLed, LOW);
        digitalWrite(blueLed, HIGH);
        delay(100);
        digitalWrite(redLed, HIGH);
        digitalWrite(blueLed, LOW);
    }

digitalWrite(redLed, HIGH);
    digitalWrite(blueLed, HIGH);

    for(int j=0;j<5;j++)
    {
        for(int i=0;i<=highPos;i++)
        {
            servo1.write(restPos1+i);
            servo2.write(restPos2-i);
            delay(time1);
        }
        for(int i=highPos;i>=0;i--)
        {
            servo1.write(restPos1+i);
            servo2.write(restPos2-i);
            delay(time1);
        }
        for(int i=0;i<=highPos;i++)
        {
            servo1.write(restPos1-i);
            servo2.write(restPos2+i);
            delay(time1);
        }
    }
void classAtt()
{
    digitalWrite(blueLed, HIGH);
    digitalWrite(redLed, LOW);

    for(int j=0; j<3; j++)
    {
        for(int i=0; i<=highPos; i++)
        {
            servo1.write(restPos1+i);
            servo2.write(restPos2+i);
            delay(time1);
        }
        for(int i=highPos; i>=0; i--)
        {
            servo1.write(restPos1+i);
            servo2.write(restPos2+i);
            delay(time1);
        }
    }

    for(int i=0; i<=pos1; i++)
    {
        servo1.write(restPos1+i);
        servo2.write(restPos2+i);
        delay(time1);
    }
    delay(time2);

    for(int i=pos1; i>=0; i--)
    {
        servo1.write(restPos1+i);
        servo2.write(restPos2+i);
        delay(time1);
    }
}