Raspberry Pi Based Hand Gesture Recognition and Voice Generation Using Python

Priyanka Dash
Assistant Professor
St. Martin's Engineering College, Hyderabad.
Priyankadash2018@gmail.com

Mruthyunjayam Allakonda
Assistant Professor
St. Martin's Engineering College, Hyderabad
Jay.snist.06@gmail.com

***Abstract*— Today the main problem faced by the deaf and dumbfolksisthecommunicationshare,todelivertheirthought withotherdeafanddumbpeopleandwithothernormalpeople. Normal people can engross new information and knowledge through the daily noises, conversations and language that is spoken around them. Deaf and hard-of hearing people do not have that luxury. This system will assistance those folks by providing an intermediate to communicate. It is implemented using devices like flex sensors, and Arduino microcontroller. When the dumb people will use gesture to communicate with other people the voice will be generated so the normal people can understand what he or she wants to talk. The raspberry pi 3B model is used and to play the audio file usingpython.**

***Keywords*— Flex, Raspberry Pi, API, Python, ADC.**

I. INTRODUCTION

Gesturingisamethodofcommunicationtocurrentaspecific meaning. In India nearly 31% of people are either deaf or mute. So, the sign language explanation using sensors is being explored as an auxiliary tool for deaf and mute people to blend into society deprived of barriers. This wearable device will have five flex-sensor. Sensed data are collected and to classify the alphabets. Then, the recognized alphabet is further conveyed into voice. According to census 2011, In India there are 1,640,868 citizens who can't speak as well as 1,261,722 citizens who can't listen. More than 70% of Deaf population of India is working in Government as well as Private sectors. For communication they are dependent on Sign Languages. In India most popularly Indian Sign Language is used. This sign language is also used in many other countries in South Asian Region[3].

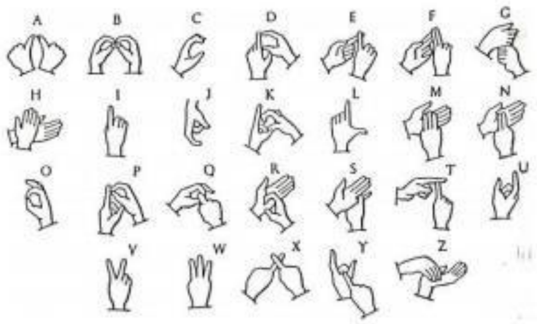

Figure 1 Indian Sign Language with a total of 26 letters using both left and right hands

InrestofthepopulationofIndiaveryfewcitizensareableto use Indian Sign Language, as they really don't need to learn sign language. This causes a communication barrierbetween DeafDumbandNormalperson.Thisleadstodisqualification of hearing impaired from main stream of the society. To overcome this problem a communication assistant is required,toconvertSignLanguagetoauditoryspeech.Some systemswerepreviouslydevelopedtoachievesameoutcome but they had disadvantages such as being non-portable, practically not implantable or expensive. All previous systems were focused on one waycommunication.

In this paper, thus, we uses the Arduino microcontroller and Raspberry Pi 3B model to complete this need. Proposed system is completely portable and focuses on two way communication.Maingoalsofthesystemaretoconverthand gesturestoauditoryspeechforcommunicationbetweenmute and normal people and to convert speech to readable text for communication between deaf to normal people. System includes two modules. Firstly code is in Arduino and after doinggesturebydumbfolkstherespectiveaudiowillbeplay usingpythontechnology.ThesystemisbasedonIndianSign Languageandcoversallthewordsrequiredindaytodaylife. Special customized section for personal information is also included. Due to different structure of grammar in sign language, a language processing algorithm is designed to arrange the words in grammatically proper English sentence [2].

II. LITERATUREREVIEW

The first Hand Talk glove was designed by Ryan Patterson within the year 2001[1]. He began his assignment together with his signing. Signing Interpreter consists of two separate mechanisms,aglovethathassomeflexiblesensorssewninto itwhichmonitorthepositionofthefingersbycalculatingthe electric resistance shaped by the fingers as they bend [1]. A little microcontroller on the rear of the hand converts the change within the electrical current into digital signals and conveys them wireless to a computer. The pc then reads the arithmetical values and converts them into the letters which appear on the screen. The most disadvantage with thismodel was that a computer or a laptop was always required for its functioning which made it less portable [1].

A. Embedded Based Hand Talk Assisting System for Deaf and Dumb. This technique was developed in March 2014. This technique uses an easy method by storing and running audio using keypad. This technique features a drawback that it doesn't usesigning.

B. Sigh language to speech converter. This technique was developed in May 2014. This technique converts thegesture to audio with the assistance of MATLAB. But the main drawback about the system is that it always requires a computer for conversion and its non-portable.

C. Interactive Glove. This technique was developed in November 2015. This technique only converts alphabetfrom signingtoaudio.So,thistechniquecreateslimitationbyonly allowing alphabets and no words are converted ofsigning.

D. Talk Aloud Gloves. This technique was developed April 2016.This system senses gestures with the flex sensors and audio is produced with the assistance of computer. The main drawback of this technique is that it requires computer and there's no way for other person tospeak.

III. SYSTEMDESIGN

Arduino:

The Arduino UNO may be an extensively used open-source microcontroller board foreseen on the ATmega328P microcontroller and developed by Arduino.cc. The board is provided with sets of digital and analog input/output (I/O) pins which will be interfaced to vary expansion boards (shields)andothercircuits.Theboardfeatures14Digitalpins and 6 Analog pins. It's programmable with the Arduino IDE (Integrated Development Environment) via a kind B USB cable

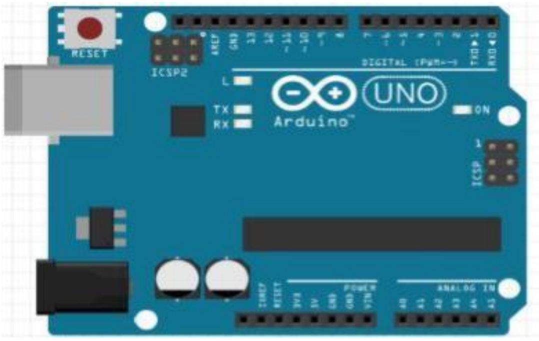

Figure 2 Arduino UNO

Technical specifications:

Microcontroller: ATmega, **Operating Voltage:** 5v, **Input Voltage:** 7-20v, Digital I/O **Pins:** 14 (of which 6 provide PWM output), **Analog Input Pins:** 6, DC Current per I/O Pin: 20 mA, **DC Current for 3.3V Pin:** 50 mA, **Flash Memory:** 32 KB of which 0.5 KB used by**, SRAM:** 2 KB, **EEPROM:**1KB,**ClockSpeed:**16MHz,**Length:**68.6mm, **Width:** 53.4 mm, **Weight:** 25g.

Flex sensor:

Flex sensors are resistive carbon elements. When bent, the sensor produces a resistance output correlated to the bend radius [9]. The variation in resistance is approximately 10 to30 K Ohm„s. An unflexed sensor has 10Kohm resistance andwhenbenttheresistanceincreasesto30Kohmat90o[3]. The sensor is about ¼ inch wide, 4-1/2 incheslong.

Figure 3 Flex Sensor

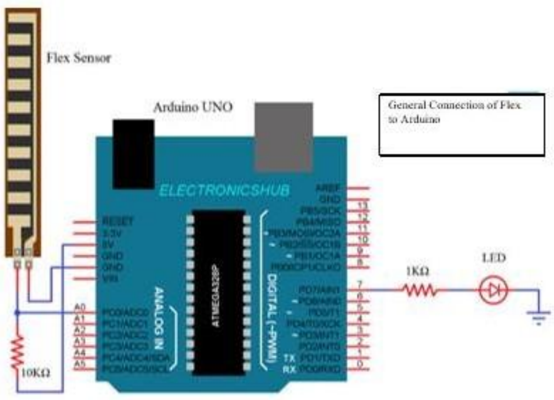

Figure 4 General Diagram

The sensor is incorporated in device using a voltage divider network. Voltage divider is used to determine the output voltage across two resistances connected in series i.e. basicallyresistancetovoltageconverter.Theresistorandflex forms a voltage divider which divides the input voltage by a ratio determined by the variable and fixedresistors.

Table 1 Flex technical specification

Flex length(mm)	2.2" (56)
Total Length (cm)	7
Life Cycle	>1 million
Height	0.43mm (0.017")
Flat Resistance	10K Ohms ±30%
Bend Resistance	minimum 20K Ohms ±30% (@ 180° pinch bend)
Power Rating	0.5 Watts continuous; 1 Watt Peak.
Shipment Weight	0.085 kg
Shipment Dimensions	3 × 2 × 1 cm

Raspberry Pi 3B Model:

Raspberry Pi presents the Raspberry Pi 3, a great device for learning, coding, and creating projects. The Raspberry Pi 3 has included integrated 802.11 b/g/n wireless LAN and Bluetooth classic and low energy (BLE). It also includes a muchfasterquad-coreCortex®A53processorrunningat1.2 GHz. The Raspberry Pi 3 has complete compatibility with Raspberry Pi 2, which means almost all theprevious

Raspberry Pi 2 accessories are compatible with Raspberry Pi 3 Model.

Figure 5 Raspberry Pi 3B Model

Table 2 Features of Raspberry pi 3B Model

Features	
A 1.2 GHz 64-bit quad-core ARMv8 CPU 802.11n wireless LAN Bluetooth 4.0 BLE Four USB ports 40 GPIO pins Full HDMI port	Ethernet port Camera interface (CSI) Display interface (DSI) Combined 3.5 mm audio jack and composite video Micro-SD card slot (now push-pull rather than push-push) Video Core ® IV 3D graphics core

IV. PROPOSEDSYSTEM

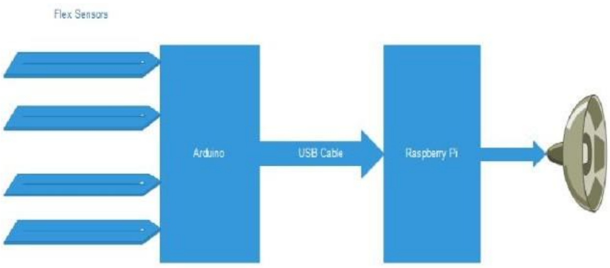

Figure 6: Proposed System

As shown in above figure 2, Arduino Uno R3, Raspberry Pi, Flex Sensors, and Speaker are used while building system. Flex Sensor are connected to Arduino through analog pinsof Arduino. While writing program each flex sensor has one unique function as output depends. Arduino connects to raspberry pi for sharing output values of flex sensor. Raspberry pi has installed audio drivers and to playsound.

V. Implementation andResults

The Arduino play very important role while implementing this concept, the flex sensor is connect to Arduino microcontroller. The five flex sensors used here the sensors is connected to analog pins of microcontroller A0, A1, A2, A3, A4 the pins are analog so the data will get range from 0 to1023soasperthatvaluestherestprocesswillbeanalyzed.

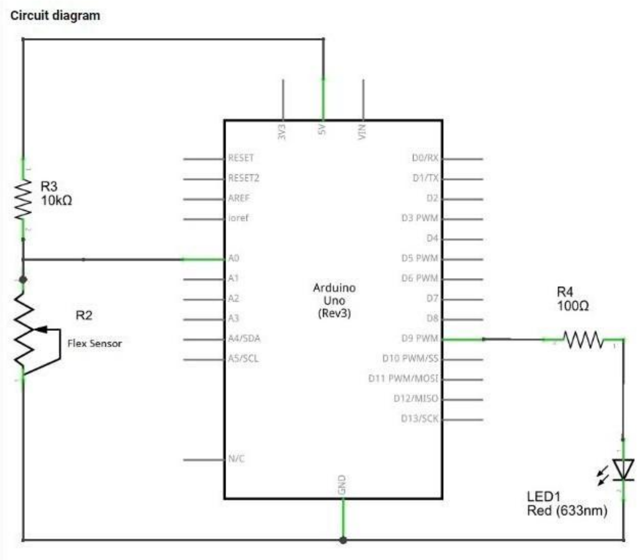

Figure 7 Flex to Arduino Connection

Figure 7 shows the single flex connection with Arduino.

File Edit Sketch Tools Help

finalDone

```
const int FLEX_PIN0 = A0;
const int FLEX_PIN1 = A1;
const int FLEX_PIN2 = A2;
const int FLEX_PIN3 = A3;
const int FLEX_PIN4 = A4;

// Pin connected to voltage divider output

// Measure the voltage at 5V and the actual resistance of your

// 47k resistor, and enter them below:
const float VCC = 4.7;

// Measured voltage of Ardunio 5V line
const float R_DIV = 47500.0;

// Measured resistance of 3.3k resistor

// Upload the code, then try to adjust these values to more

// accurately calculate bend degree.
const float STRAIGHT_RESISTANCE = 37300.0;

// resistance when straight
const float BEND_RESISTANCE = 90000.0;

// resistance at 90 deg
```

Figure 8 Code Part-I

The figure 8 shows the coding part here above snippet is for declaring the variable which is used while processing.

```
void setup()

{ Serial.begin(9600);

pinMode(FLEX_PIN0, INPUT);
pinMode(FLEX_PIN1, INPUT);
pinMode(FLEX_PIN2, INPUT);
pinMode(FLEX_PIN3, INPUT);
pinMode(FLEX_PIN4, INPUT);

pinMode(6,OUTPUT);
pinMode(7,OUTPUT);
}

void loop()

{ // Read the ADC, and calculate voltage and resistance from it

int flexADC0 = analogRead(FLEX_PIN0);
int flexADC1 = analogRead(FLEX_PIN1);
int flexADC2 = analogRead(FLEX_PIN2);
int flexADC3 = analogRead(FLEX_PIN3);
int flexADC4 = analogRead(FLEX_PIN4);

float flexV0 = flexADC0 * VCC / 1023.0;
float flexV1 = flexADC1 * VCC / 1023.0;
float flexV2 = flexADC2 * VCC / 1023.0;
float flexV3 = flexADC3 * VCC / 1023.0;
```

Figure 9 Code Part-II

The figure 9 shows the coding part here declared all flex analog pins where it actually connected and after getting values from it. The values can be multiplied by VCC the value of VCC is 4.7. This is voltage of Arduino microcontrollerandagaindividedby1023togetexactvalue. Shows in equationno.1.

$$flexV0 = \left(\frac{flexADC0 * 4.7}{1023.0}\right) - - - - - - - - (1)$$

```
float angle0 = map(flexR0, STRAIGHT_RESISTANCE, BEND_RESISTANCE, 0, 90.0);
float angle1 = map(flexR1, STRAIGHT_RESISTANCE, BEND_RESISTANCE, 0, 90.0);
float angle2 = map(flexR2, STRAIGHT_RESISTANCE, BEND_RESISTANCE, 0, 90.0);
float angle3 = map(flexR3, STRAIGHT_RESISTANCE, BEND_RESISTANCE, 0, 90.0);
float angle4 = map(flexR4, STRAIGHT_RESISTANCE, BEND_RESISTANCE, 0, 90.0);

//Serial.println("Bend0: " + String(angle0) + " degrees");
//Serial.println("Bend1: " + String(angle1) + " degrees");
//Serial.println("Bend2: " + String(angle2) + " degrees");
//Serial.println("Bend3: " + String(angle3) + " degrees");
//Serial.println("Bend: " + String(angle4) + " degrees");
int a0=String(angle0).toInt();
int a1=String(angle1).toInt();
int a2=String(angle2).toInt();
int a3=String(angle3).toInt();
int a4=String(angle4).toInt();

int a00=-(a0);
int a11=-(a1);
int a22=-(a2);
int a33=-(a3);
int a44=-(a4);
```

Figure 10 Code Part-III

The figure 10 shows the coding part here, the actual bending of flex can be analyzed and the value stored in variables.

```
else if (a00<60 && a11<58 && a22<60)
{
// Serial.println(a0);
// Serial.println(a1);
  Serial.println("I want It");
   digitalWrite(6,LOW);
  digitalWrite(7,HIGH);
}

else if(a22<60 && a33<60 && a44<60)
{
  Serial.println("Be Strong");
   digitalWrite(6,LOW);
  digitalWrite(7,HIGH);
}



else if(a22<58 && a44<58)
{
  Serial.println("We Rock");
   digitalWrite(6,LOW);
  digitalWrite(7,HIGH);
}
```

Figure 11 Code Part-IV

The figure 11 shows the coding part here, after bending flex as per gesture then it will print the message on serial port. This message is very important this message can be taken by further in python for processing.

```
else if(a11<58)
{
  //Serial.println(a1);
  Serial.println("Best of Luck");
   digitalWrite(6,LOW);
  digitalWrite(7,HIGH);
}
else if(a33<60 )
{
  Serial.println("I need it");
   digitalWrite(6,LOW);
  digitalWrite(7,HIGH);
}



else
{
   Serial.println("Try Again");
   digitalWrite(6,HIGH);
   digitalWrite(7,LOW);
   delay(10);
}

Serial.println();

delay(500); }
```

Figure 12 Code Part-V

Figure 12 shows the same as above, total 6 audio will play as per the gesture.

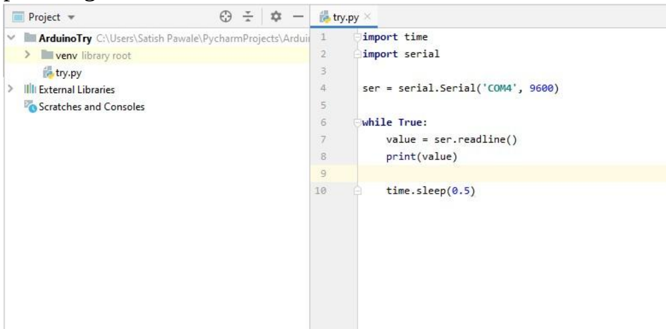

Figure 13 Python Code to access Arduino on Windows Platform

The figure 13 shows the interfacing of python with Arduino on Windows platform.

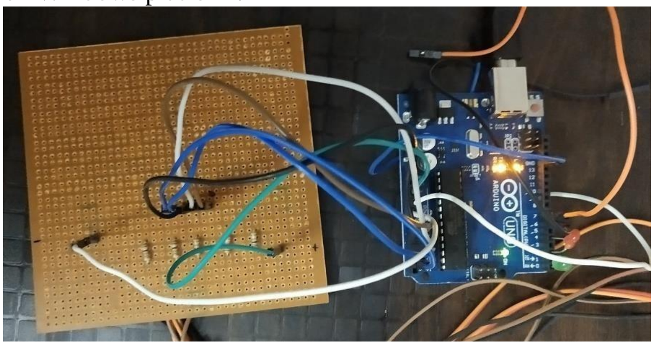

Figure 14 Arduino and Flex

The figure 14 shows the connection of Arduino with flex sensors on zero PCB to make connection easy.

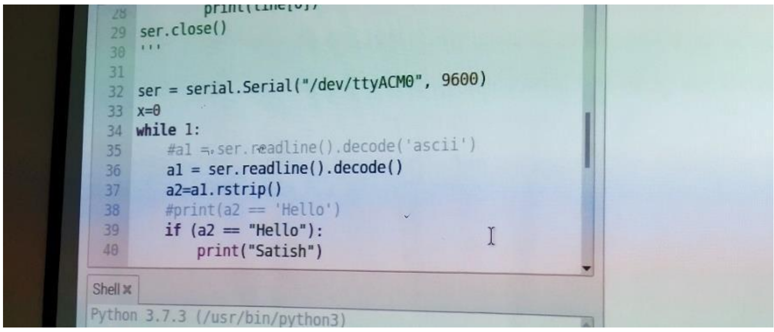

Figure 14 Arduino and Python Interfacing

Figure 15 shows the Arduino Python interfacing using Raspberry Pi model.

$$ser = serial.Seri(/dev/ttyACM0, 9600)$$

This shows the "ttyACMO" has connected Arduino microcontroller through USB port. The 9600 is baud rate to sync with Arduino and system.

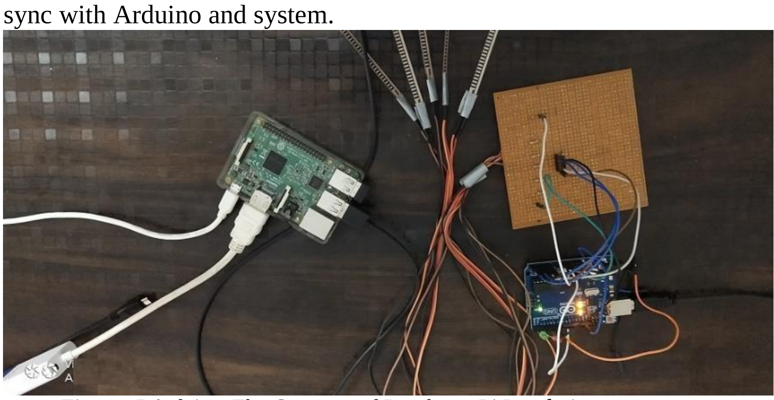

Figure 15 Arduino, Flex Sensors and Raspberry Pi Interfacing

Figure 16 shows the overall connection of Arduino, flex and Raspberry Pi model. The values getting from Arduino which will process in Raspberry Pi and the audio will play as per gesture using the Python code. The raspberry Pi have the audio port with 3.5mm audio jack so can be attached the speaker directly to Raspberry Pi.

VI. ADVANTAGES OFSYSTEM

Requires fewer components so its cost is low.
It is economical.
It is small in size, due to the small size we can place its hardware on our hand easily.
The whole apparatus carries less weight. Hence they are portable and flexible to users.

CONCLUSION

In this paper we reviewed different approaches proposed for handgesturerecognition.Theseapproachesvarylanguageto language.Basicimplementationofsensorgloveisdoneusing flex sensors. There are basically two parts in all approaches, oneisrecognizinghandposturewithoutmotionandonewith motion. Our approach is to perform these complex computation and operations using raspberry pi and Arduino and generate the audio with respect to handmotion.

FUTURE WORK

In this system, more sensors can be embedded to recognize full sign language with more perfection and accuracy.
The system can also be designed such that it can translate words from one language to another

ACKNOWLEDGMENT

I would like to thank Dr. Ashok Kumar Nanda for guidance and support. I willforeverremaingratefulforconstantsupportandguidance extended by him, for the completion of paper. I am also thankful to Dr. Ashok Kumar Nanda for his help during the literature search process.

REFERENCES

[1]SyedFaizAhmed,SyedMuhammadBaberAli,Sh.SaqibandMunawwarQureshi, Electronic Speaking Glove for Speechless Patients: A Tongue to a Dumb, Proceedings of the 2010, IEEE Conference onSustainableUtilization and Development in Engineering and Technology University TunkuAbdul Rahman, 20 & 21 November 2010, Faculty of Engineering, Kuala Lumpur, Malaysia.

[2] MandarTawde, HariomSingh, ShoebShaikh, “Glove for Gesture Recognition using Flex Sensor”, International Journal of Recent Trends in Engineering & Research (IJRTER) Volume 03, Issue 03; March - 2017 [ISSN: 2455-1457], DOI :10.23883/IJRTER.2017.3041.QVZQM.

[3] Golda Jeyasheeli P and Annapoorani K, “IoTBased Sign Language Interpretation System”, International Conference on Physics and Photonics Processes in Nano Sciences , Journal of Physics: Conference Series , 1362 (2019) 012034 IOP Publishing,doi:10.1088/1742-6596/1362/1/012034