Gesture Controlled Glove

by

Name	Roll No.	Registration No:
SANU BHATTACHARYA	**11700314081**	**141170110263** of 2014-2015
SHUBHAM KUMAR	**11700314094**	**141170110276** of 2014-2015
CHANDRIKA BANERJEE	**11700314033**	**141170110215** of 2014-2015
PRITHA MUKHERJEE	**11700314059**	**141170110241** of 2014-2015

A comprehensive project report has been submitted in partial fulfillment of the requirements for the degree of

Bachelor of Technology

in

ELECTRONICS & COMMUNICATION ENGINEERING

Under the supervision of

Mrs. Tiya Dey Malakar

Assistant Professor

Department of Electronics & Communication Engineering
RCC INSTITUTE OF INFORMATION TECHNOLOGY
Affiliated to Maulana Abul Kalam Azad University of Technology, WestBengal
CANAL SOUTH ROAD, BELIAGHATA, KOLKATA – 700015

May, 2018

CERTIFICATE OF APPROVAL

This is to certify that the project titled "**Gesture Controlled Glove**" carried out by

Name	Roll No.	Registration No:
SANU BHATTACHARYA	**11700314081**	**141170110263** of 2014-2015
SHUBHAM KUMAR	**11700314094**	**141170110276** of 2014-2015
CHANDRIKA BANERJEE	**11700314033**	**141170110215** of 2014-2015
PRITHA MUKHERJEE	**11700314059**	**141170110241** of 2014-2015

for the partial fulfillment of the requirements for B.Tech degree in **Electronics and Communication Engineering** from **Maulana Abul Kalam Azad University of Technology, West Bengal**is absolutely based on his own work under the supervision of Mrs**. Tiya Dey Malakar**. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Optional in case of External Supervisor

....

Dr./Mr./Ms./Mrs.
Designation and Department
Institute

....

Mrs. Tiya Dey Malakar
Professor , Dept. of ECE
RCC Institute of Information Technology

....

Dr. Abhishek Basu
Head of the Department (ECE),
RCC Institute of Information Technology

DECLARATION

"We Do hereby declare that this submission is our own work conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute and that, to the best of our knowledge and belief, it contains no material previously written by another neither person nor material (data, theoretical analysis, figures, and text) which has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgement has been made in the text."

....

SANU BHATTACHARYA
Registration No:141170110263 OF 2014-2015
Roll No: 11700314081

....

SHUBHAM KUMAR
Registration No: 141170110276 OF 2014-2015
Roll No: 11700314094

....

CHANDRIKA BANERJEE
Registration No: 141170110215 OF 2014-2015
Roll No: 11700314033

....

PRITHA MUKHERJEE
Registration No: 141170110241 OF 2014-2015
Roll No: 11700314059

Date:

Place:

CERTIFICATE of ACCEPTANCE

This is to certify that the project titled "**Gesture Controlled Glove**" carried out by

Name	Roll No.	Registration No:
SANU BHATTACHARYA	**11700314081**	**141170110263** of 2014-2015
SHUBHAM KUMAR	**11700314094**	**141170110276** of 2014-2015
CHANDRIKA BANERJEE	**11700314033**	**141170110215** of 2014-2015
PRITHA MUKHERJEE	**11700314059**	**141170110241** of 2014-2015

is hereby recommended to be accepted for the partial fulfillment of the requirements for B.Tech degree in **Electronics and Communication Engineering** from **Maulana Abul Kalam Azad University of Technology, West Bengal**

Name of the Examiner Signature with Date

1. ………………………………………………………………………

2.……………………………………..………………………………..

3.………………………………………………………………………

4. ………………………………………..……………………………

Contents

INTRODUCTION:-

Humans and machines do not interface well. In an attempt to bridge the gap between humans and the systems they interact with, a plethora of input methods have been devised: keyboards, mouse, joysticks, game controllers, and touch screens are just a few examples.

There have been growing attempts in finding natural ways for human–machine interaction (HMI) for multimedia entertainment . That is because traditional ways, e.g., by using a mouse or a keyboard, are inherently limited by speed and space, and lack of an immersed sense within the simulated environments . Use of other options than mouse or keyboard i.e. grasping virtual objects, hand, head or body gesture, eye fixation tracking are becoming popular with popularity of ubiquitous and ambient devices like digital TV, play stations. We will see more elderly people and fewer younger people as a process of huge demographic change. The older population will continue to grow significantly in the future. It is widely accepted that we need to address this issue through more research work.

In recent years, gesture recognition for HMI has become popular because it helps overcome the limitations of traditional ways and enriches personal experience during the interaction between human and machine for entertainment .

If instead of moving your whole arm to complete a motion gesture, you only need to move a finger?
A gradual development of motion control and gesture recognition over the years has shown us that in practical use, motion control gestures don't need to be large flailing motions. It's not necessary to be swinging your arms all over the place. After all, like all animals, we humans want to conserve our energy for when it's needed.

In addition, touch screens were a great innovation 15 years ago, but the lack of tactility made it difficult to touch type or control devices without looking directly at them. Our use of computing is increasingly mobile these days, so mobile devices needs to be controlled on the go.

New technologies such as augmented reality goggles meant a new input device was needed to replace keyboards, mice and touchscreens. And what better input device than the hands themselves? These developments have finally been realised in gesture sensing touch gloves and more recently — touch implants.

Gesture recognition is also important in automated surveillance and human monitoring applications , where they can yield valuable clues into human activities and intentions . Gestures are defined as human motion sequences with trajectories performed in a short interval of time. Human gestures are a natural means of interaction and communication among people. Gestures employ hand, limb and body motion to express ideas or exchange information non- verbally . Gestures can be divided into two groups: static and dynamic gestures . According to input device, the gesture recognition technique can be divided into three categories: glove-data based, vision- based and accelerometer-based .

EXECUTIVE SUMMARY:-

Gesture recognition is a topic in computer science and language technology with the goal of interpreting human gestures via mathematical algorithms. Gestures can originate from any bodily motion or state but commonly originate from the face or hand. Current focuses in the field include emotion recognition from face and hand gesture recognition. Users can use simple gestures to control or interact with devices without physically touching them. Many approaches have been made using cameras and computer vision algorithms to interpret sign language. However, the identification and recognition of posture, gait, polemics, and human behaviors is also the subject of gesture recognition techniques.Gesture recognition can be seen as a way for computers to begin to understand human body language, thus building a richer bridge between machines and humans than primitive text user interfaces or even GUIs (graphical user interfaces), which still limit the majority of input to keyboard and mouse.

Gesture recognition enables humans to communicate with the machine (HMI) and interact naturally without any mechanical devices. Using the concept of gesture recognition, it is possible to point a finger at the computer screen so that the cursor will move accordingly. This could make conventional input devices such as mouse, keyboards and even touch-screens redundant.

Gesture Recognition Features:

- More accurate
- High stability
- Time saving to unlock a device

The major application areas of gesture recognition in the current scenario are:

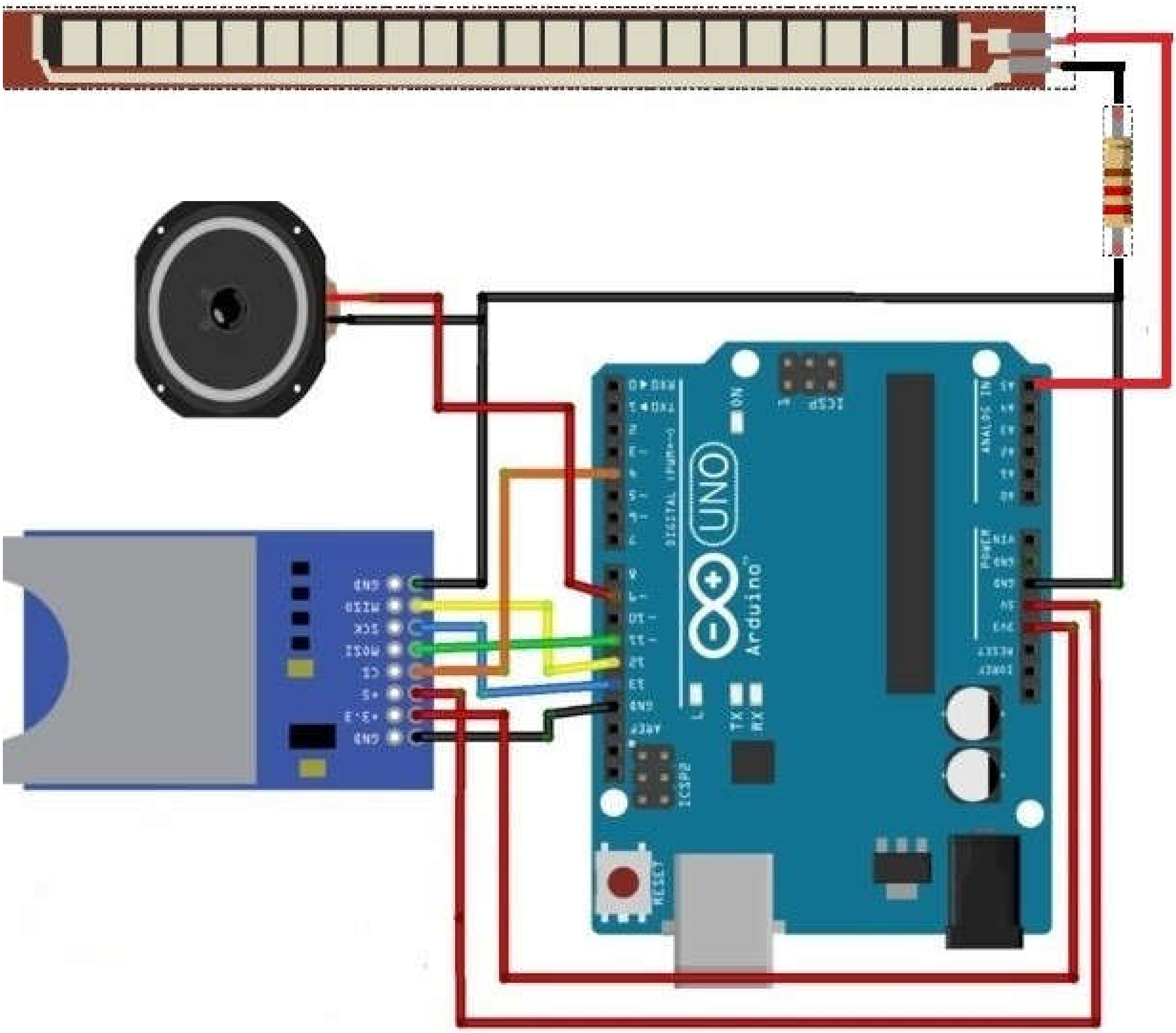
- Automotive sector
- Consumer electronics sector
- Transit sector
- Gaming sector
- To unlock smartphones
- Defence
- Home automation
- Sign language interpretation

Gesture recognition technology has been considered to be the highly successful technology as it saves time to unlock any device.

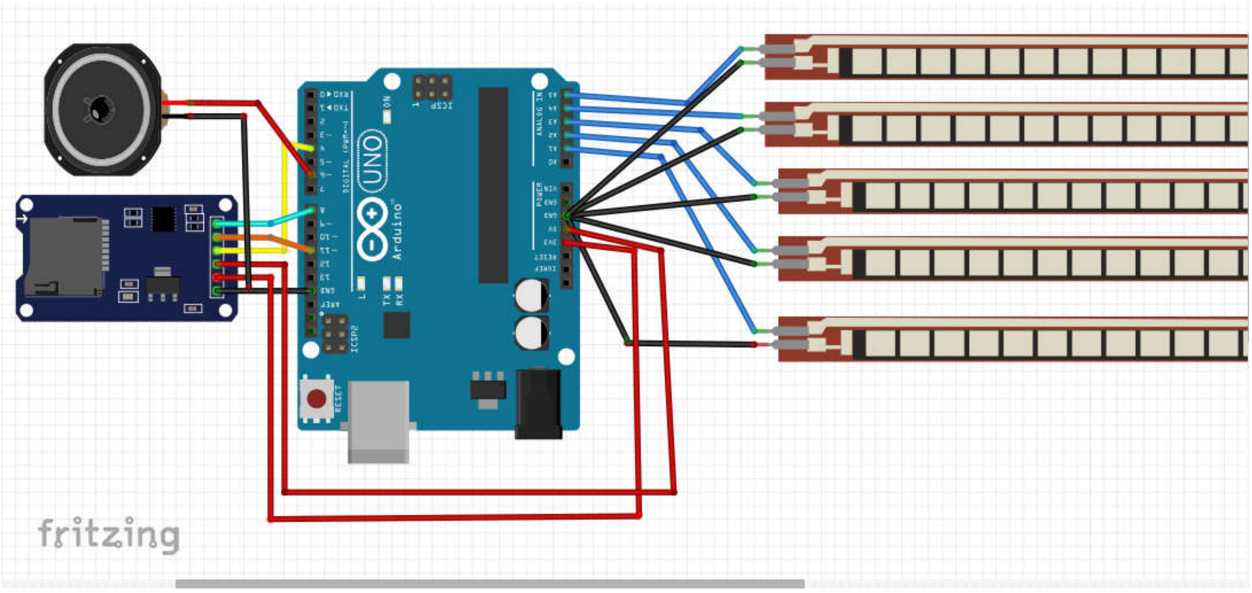
Gesture Controlled Glove aims to bridge the gap between the user and traditional physical hardware devices. Given the high learning curve in understanding how to use foreign technologies, we hope to break away from conventional control mechanisms and explore an intuitive way to control these devices. It provides a tangible interface that relies on hand gestures to wirelessly control any device or software. By removing the distance between the user and traditional hardware devices, our goal is to feel more like an extension of the body as opposed to an external machine.
As an investigation into this idea, the goal of this project is to capture simple hand gestures and use that input to obtain a desired output.

Humans naturally use gesture to communicate. It has been demonstrated that young children can readily learn to communicate with gesture before they learn to talk. A gesture is non-verbal communication made with a part of the body. We use gesture instead of or in combination with verbal communication. Using this process, human can interface with the machine without any mechanical devices. Human movements are typically analyzed by segmenting them into shorter and understandable format. The movements vary person to person. It can be used as a command to control different devices of daily activities, mobility etc. So our natural or intuitive body movements or gestures can be used as command or interface to operate machines, communicate with intelligent environments to control home appliances, smart home, telecare systems etc.

CIRCUIT DIAGRAM:-



Single flex sensor circuit



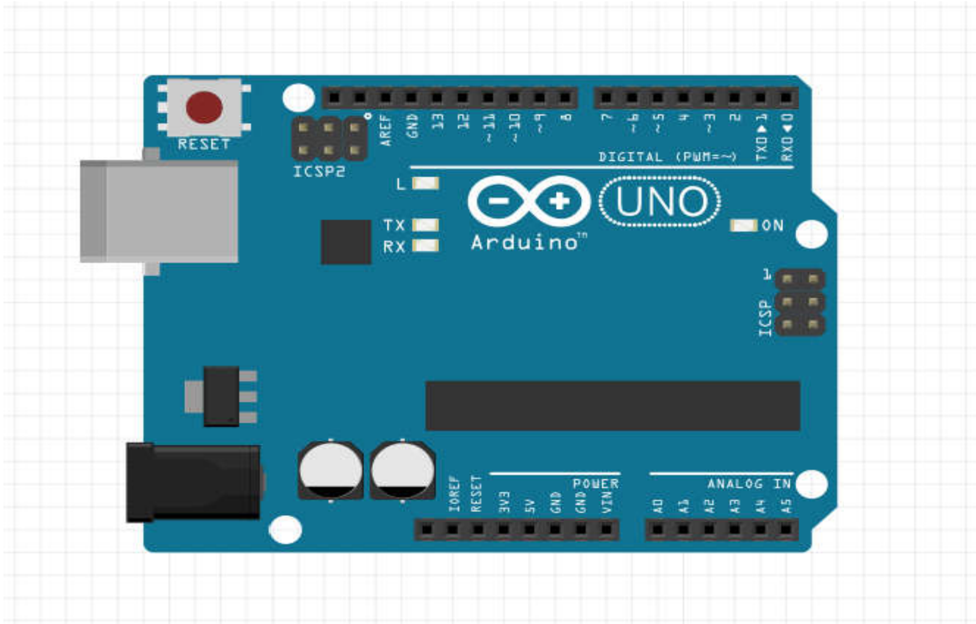
5 flex sensor circuit

The project uses an Arduino Uno as the microcontroller at the centre, to which all the other devices are connected. The Arduino processes the input taken from the flex sensors and provides an analog value as the output. Using this value for each flex sensor we can determine the particular gesture and use this input as a conditional logic to control other devices that are connected to the Uno.

We can use the same approach to output audio signals where each audio file is stores in a micro-sd card with *.wav* extension. The micro-sd card is placed inside the micro-sd card module and connected to the Arduino Uno. Each gesture is assigned a particular audio file that is present in the micro-sd card and using simple if-else-if logic we can program the micro controller to output a particular audio for a particular gesture.

ARDUINO UNO

The Arduino UNO is a widely used open-source microcontroller board predicted on the ATmega328P microcontroller and developed by Arduino.cc. The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The board features 14 Digital pins and 6 Analog pins. It is programmable with the Arduino IDE (Integrated Development Environment) via a type B USB cable. It can be powered by a USB cable or by an external 9 volt battery, though it accepts voltages between 7 and 20 volts. It is additionally homogeneous to the Arduino Nano and Leonardo. The hardware reference design is distributed under a Creative Commons Attribution Share-Alike 2.5 license and is available on the Arduino website. Layout and production files for some versions of the hardware are also available. "Uno" means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform. The ATmega328 on the Arduino Uno comes preprogrammed with a bootloader that allows to upload new code to it without the use of an external hardware programmer. It communicates utilizing the pristine STK500 protocol. The Uno also differs from all preceding boards in that it does not use the FTDI USB-to-serial driver chip. Instead, it features the Atmega16U2 (Atmega8U2 up to version R2) programmed as a USB-to-serial converter. The Arduino UNO is generally considered the most user-friendly and popular board, with boards being sold worldwide for less than 5$.



Arduino Uno

Technical specifications

- Microcontroller: ATmega328P
- 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 bootloader
- SRAM: 2 KB
- EEPROM: 1 KB
- Clock Speed: 16 MHz
- Length: 68.6 mm
- Width: 53.4 mm
- Weight: 25 g

General Pin functions

- **LED**: There is a built-in LED driven by digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.
- **VIN**: The input voltage to the Arduino/Genuino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin.
- **5V**: This pin outputs a regulated 5V from the regulator on the board. The board can be supplied with power either from the DC power jack (7 - 20V), the USB connector (5V), or the VIN pin of the board (7-20V). Supplying voltage via the 5V or 3.3V pins bypasses the regulator, and can damage the board.
- **3V3**: A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND**: Ground pins.
- **IOREF**: This pin on the Arduino/Genuino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage and select the appropriate power source or enable voltage translators on the outputs to work with the 5V or 3.3V.
- **Reset**: Typically used to add a reset button to shields which block the one on the board

Arduino IDE

The Arduino integrated development environment (IDE) is a cross-platform application (for Windows, macOS, Linux) that is written in the programming language Java. It originated from the IDE for the languages *Processing* and *Wiring*. It includes a code editor with features such as text cutting and pasting, searching and replacing text, automatic indenting, brace matching, and syntax highlighting, and provides simple *one-click* mechanisms to compile and upload programs to an Arduino board. It also contains a message area, a text console, a toolbar with buttons for common functions and a hierarchy of operation menus. The source code for the IDE is released under the GNU General Public License, version 2.

The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures. User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub *main()* into an executable cyclic executiveprogram with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program *avrdude* to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

A program written with the Arduino IDE is called a *sketch*. Sketches are saved on the development computer as text files with the file extension *.ino*. Arduino Software (IDE) pre-1.0 saved sketches with the extension *.pde*.

A minimal Arduino C/C++ program consist of only two functions:

- *setup()*: This function is called once when a sketch starts after power-up or reset. It is used to initialize variables, input and output pin modes, and other libraries needed in the sketch.
-
- *loop()*: After *setup()* has been called, function *loop()* is executed repeatedly in the main program. It controls the board until the board is powered off or is reset.

Light Emitting Diode(LED)

A **light-emitting diode** (**LED**) is a two-lead semiconductor light source. It is a p–n junction diode that emits light when activated. When a suitable current is applied to the leads, electrons are able to recombine with electron holes within the device, releasing energy in the form of photons. This effect is called electroluminescence, and the color of the light (corresponding to the energy of the photon) is determined by the energy band gap of the semiconductor. LEDs are typically small (less than

1 mm^2) and integrated optical components may be used to shape the radiation pattern.

Appearing as practical electronic components in 1962, the earliest LEDs emitted low-intensity infrared light. Infrared LEDs are still frequently used as transmitting elements in remote-control circuits, such as those in remote controls for a wide variety of consumer electronics. The first visible-light LEDs were of low intensity and limited to red. Modern LEDs are available across the visible, ultraviolet, and infrared wavelengths, with very high brightness.

Early LEDs were often used as indicator lamps for electronic devices, replacing small incandescent bulbs. They were soon packaged into numeric readouts in the form of seven-segment displays and were commonly seen in digital clocks. Recent developments have produced LEDs suitable for environmental and task lighting. LEDs have led to new displays and sensors, while their high switching rates are useful in advanced communications technology.

LEDs have many advantages over incandescent light sources, including lower energy consumption, longer lifetime, improved physical robustness, smaller size, and faster switching. Light-emitting diodes are used in applications as diverse as aviation lighting, automotive headlamps, advertising, general lighting, traffic signals, camera flashes, lighted wallpaper and medical devices. They are also significantly more energy efficient and, arguably, have fewer environmental concerns linked to their disposal.

Unlike a laser, the color of light emitted from an LED is neither coherent nor monochromatic, but the spectrum is narrow with respect to human vision, and for most purposes the light from a simple diode element can be regarded as functionally monochromatic.

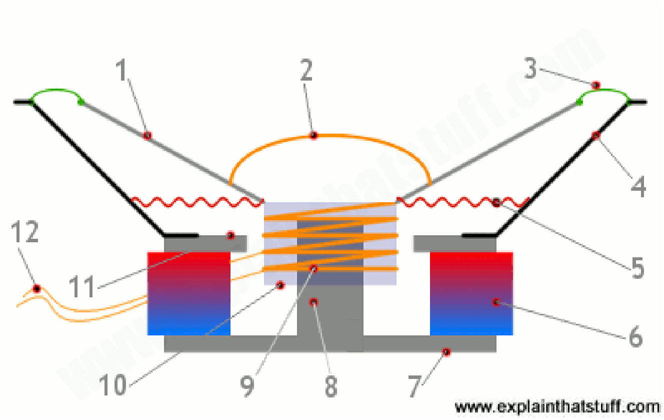
LED

<u>SPEAKER:-</u>

They receive audio input from the computer's sound card and produce audio output in the form of sound waves. Most computer **speakers** are active **speakers**, meaning they have an internal amplifier which allows you to increase the volume, or amplitude, of the sound.

In order to translate an electrical signal into an audible sound, speakers contain an electromagnet: a metal coil which creates a magnetic field when an electric current flows through it. This coil behaves much like a normal (permanent) magnet, with one particularly handy property: reversing the direction of the current in the coil flips the poles of the magnet.

What are the parts of speaker? And what do they do?



Diaphragm (cone) (1): Moves in and out to push air and make sound.

Dust cap (dome) (2): Protects the voice coil from dust and dirt.

Surround (3): A piece of elastic rubber that flexibly fastens the diaphragm to the basket (outer frame).

Basket (4): metal framework around which the speaker is built.

Spider (suspension)(5): A support that holds the voice coil in place, while allowing it to move freely.

Magnet (6): Typically made from ferrite or powerful neodymium.

Bottom plate (7): Made of soft iron.

Pole piece (8): Concentrates the magnetic field produced by the voice coil.

Voice coil (9): The coil that moves the diaphragm back and forth.

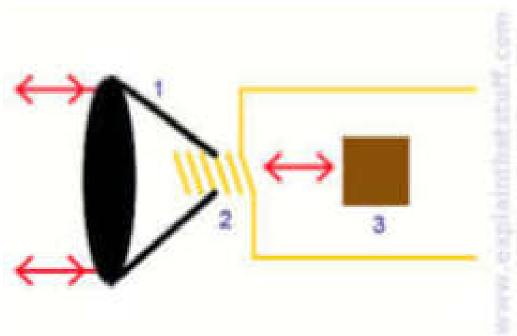
Former (10): A cylinder of cardboard onto which the coil is wound.

Top plate (11): Also made of soft iron.

Cables (12): Connect stereo amplifier unit to voice coil.

Inside a speaker, an electromagnet is placed in front of a permanent magnet. The permanent magnet is fixed firmly into position whereas the electromagnet is mobile. As pulses of electricity pass through the coil of the electromagnet, the direction of its magnetic field is rapidly changed. This means that it is in turn attracted to and repelled from the permanent magnet, vibrating back and forth.

The electromagnet is attached to a cone made of a flexible material such as paper or plastic which amplifies these vibrations, pumping sound waves into the surrounding air and towards your ears.



The frequency of the vibrations governs the pitch of the sound produced, and their amplitude affects the volume – turn your stereo up high enough and you might even be able to see the diaphragm covering the cone move.

To reproduce all the different frequencies of sound in a piece of music faithfully, top quality speakers typically use different sized cones dedicated to high, medium and low frequencies.

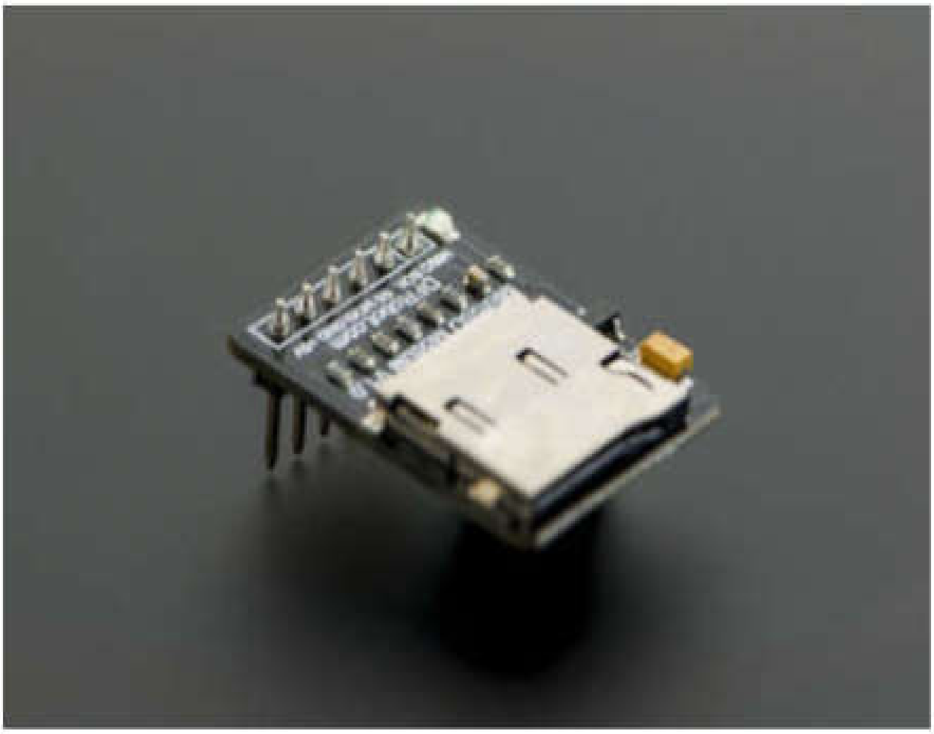
MICRO-SD CARD MODULE:-

Secure Digital (**SD**) is a non-volatile memory card format developed by the SD Card Association (SDA) for use in portable devices.

The standard was introduced in August 1999 by joint efforts between SanDisk, Panasonic (Matsushita Electric) and Toshiba as an improvement over Multimedia Cards (MMC), and has become the industry standard. The three companies formed SD-3C, LLC, and a company that licenses and enforces intellectual property rights associated with SD memory cards and SD host and ancillary products

Secure Digital includes four card families available in three different sizes. The four families are the original Standard-Capacity (SDSC), the High-Capacity (SDHC), the extended-Capacity (SDXC), and the SDIO, which combines input/output functions with data storage.

The three form factors are the original size, the mini size, and the micro size. Electrically passive adapters allow a smaller card to fit and function in a device built for a larger card. The SD card's small footprint is an ideal storage medium for smaller, thinner and more portable electronic devices.



A Micro SD(TF) module from DFRobot. It is compatible with TF SD card (commonly used in Mobile Phone) which is the most tiny card in the market. SD module has various applications such as data logger, audio, video, graphics. This module will greatly expand the capbility an Arduino can do with their poor limited memory. This module has SPI interface and 5V power supply which is compatible with Arduino UNO/Mega. The Pinout is fully compatiblw with DFRobot's IO Expansion Shield V5.

CODE:-

Our projects aims to use flex sensors attached to each finger of the glove, and connected to an Arduino microcontroller. Using the input from the flex sensor we can control external devices such as LEDs which are connected to the Arduino.

The project uses an Arduino Uno as the microcontroller at the centre, to which all the other devices are connected. The Arduino processes the input taken from the flex sensors and provides an analog value as the output. Using this value for each flex sensor we can determine the particular gesture and use this input as a conditional logic to control other devices that are connected to the Uno.

We can use the same approach to output audio signals where each audio file is stores in a micro-sd card with *.wav* extension. The micro-sd card is placed inside the micro-sd card module and connected to the Arduino Uno. Each gesture is assigned a particular audio file that is present in the micro-sd card and using simple if-else-if logic we can program the micro controller to output a particular audio for a particular gesture.

```
#include "SD.h"
#define SD_ChipSelectPin 10
#include "TMRpcm.h"
#include "SPI.h"

const int FLEX_PIN = A0;
const int FLEX_PIN1 = A5;
const int FLEX_PIN2 = A1;
int led=4;
int led1=2;
int led2=3;// 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 = 4.98; // Measured voltage of Ardunio 5V line
const float R_DIV = 47500.0; // Measured resistance of 3.3k resistor


TMRpcm tmrpcm;


const float STRAIGHT_RESISTANCE = 37300.0; // resistance when straight
const float BEND_RESISTANCE = 90000.0; // resistance at 90 deg


void setup()
{
  tmrpcm.speakerPin=9;
  Serial.begin(9600);
  pinMode(FLEX_PIN, INPUT);
    pinMode(FLEX_PIN1, INPUT);
  pinMode(FLEX_PIN2, INPUT);

  pinMode(led,OUTPUT);
   pinMode(led1,OUTPUT);
    pinMode(led2,OUTPUT);
tmrpcm.setVolume(4);
tmrpcm.play("test.wav");

}

void loop()
{
   digitalWrite(13,HIGH);
    // Read the ADC, and calculate voltage and resistance from it

  int flexADC = analogRead(FLEX_PIN);
  float flexV = flexADC * VCC / 1023.0;
  float flexR = R_DIV * (VCC / flexV - 1.0);

  float angle = map(flexR, STRAIGHT_RESISTANCE, BEND_RESISTANCE,
```

In this Arduino code we have declared the variables at the beginning outside the method. A0,A1,A5 are the pins to which flex sensors are connected 4,2,3 are the pins to which the leds are connected.

In the setup method we have declared which is the input pin and which is the output pin. We have declared flex pins to be input ports whereas led pins to be output ports. That is, we are taking the input from the flex sensors,reading the analog from the sensor and then we are getting the desired output in the form of led status. We have stored few audios in .wav extension. For a particular input and with the particular status of led we are getting an audio output.

```
  float flexV = flexADC * VCC / 1023.0;
  float flexR = R_DIV * (VCC / flexV - 1.0);

  float angle = map(flexR, STRAIGHT_RESISTANCE, BEND_RESISTANCE,
                   0, 90.0);
  Serial.print("Bend: " + String(angle) + " degrees     ");


int b = analogRead(FLEX_PIN2);
  float c = b * VCC / 1023.0;
  float d = R_DIV * (VCC / c - 1.0);

  float a = map(d, STRAIGHT_RESISTANCE, BEND_RESISTANCE,
                   0, 90.0);
  Serial.print("Bend: " + String(a) + " degrees     ");


int b1 = analogRead(FLEX_PIN1);
  float c1 = b1 * VCC / 1023.0;
  float d1 = R_DIV * (VCC / c1 - 1.0);

  float a1 = map(d1, STRAIGHT_RESISTANCE, BEND_RESISTANCE,
                   0, 90.0);
  Serial.println("Bend: " + String(a1) + " degrees");
  Serial.println();
if(angle>16)
{
  digitalWrite(led,HIGH);
}
else
digitalWrite(led,LOW);
if(a>-14)
{
  digitalWrite(led1,HIGH);
}
else
digitalWrite(led1,LOW);

if(a1>-2)
{
  digitalWrite(led2,HIGH);
}
else
digitalWrite(led2,LOW);


  delay(500);
 digitalWrite(13,LOW);
}
```

In the loop method we have used the function “analogRead” which reads the analog input from the flex sensor.The analog value which is thus taken as an input is nothing but a voltage input. This value is then subjected to calculation to get an intermediate value.

This value is then checked using conditional statements.If the value matches the required specification of the condition then the led goes to “HIGH” state else it remains in the “LOW” state.

This is in short the working principle of the Gesture Controlled Glove.

<u>WORKING</u>

The entire project can be divided into 3 parts:-

a) The input interface
b) The processing
c) The output

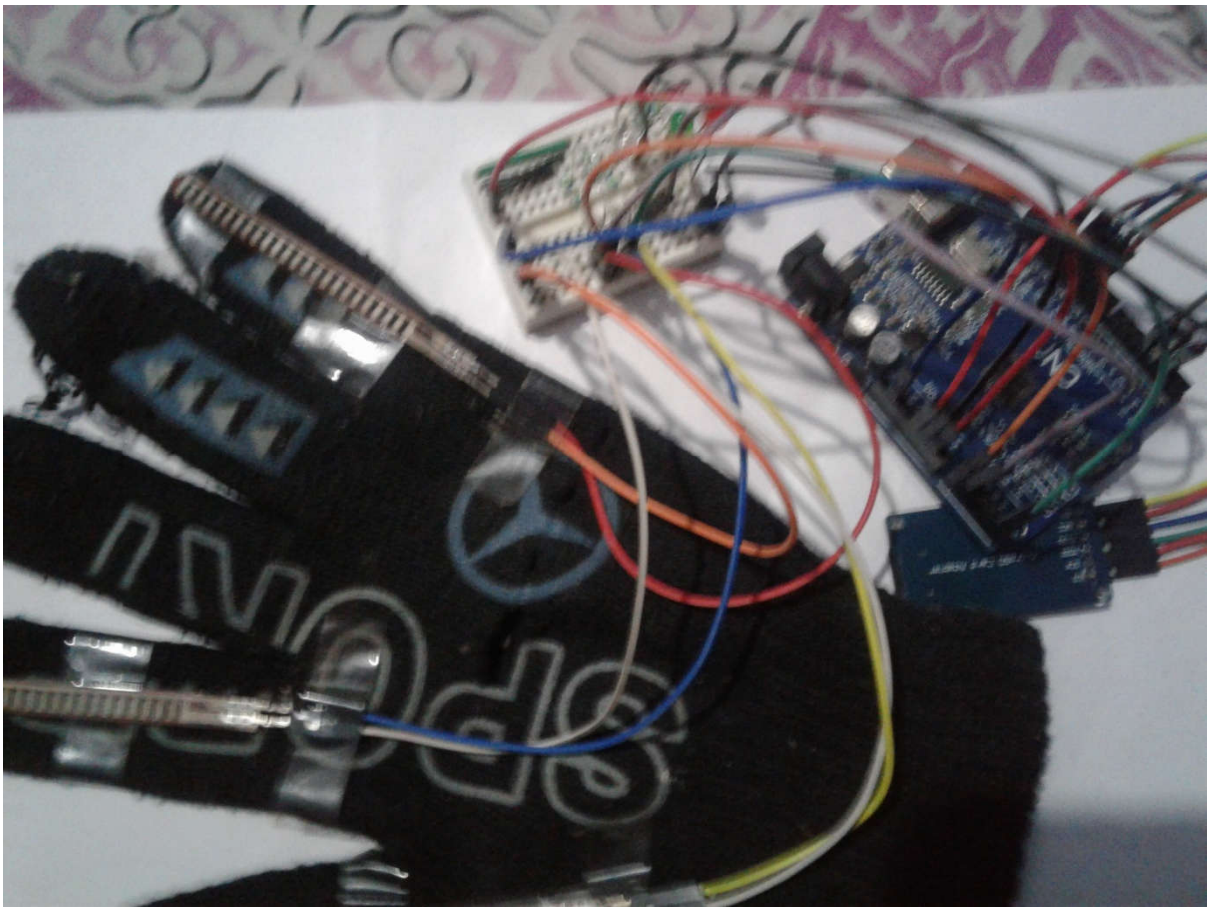
The input signal "senses" a real life input, in this case the bend of a finger. When a particular finger in bent, the flex sensor associated with that finger, has an abrupt change in its resistance value. The value of this resistance is sent to the Arduino Uno microcontroller which is the processing unit of the entire project. The program burned in the Uno, checks whether any flex sensor crosses a predetermined threshold value of the resistance.

The Arduino Uno continuously scans the flex sensor and records its value. If at any point the flex sensor is bent, then the corresponding change in the resistance value can be read. Using a conditional statement in the code of the microcontroller we can control other external devices that are connected to the board.

In the output stage we are using three Light Emitting Diodes which are connected to three different digital input/output pins of the arduino itself. These pins can be controlled using a simple function called *digitalWrite().* Using this function we can turn each light emitting diode ON or OFF. Any other devices connected to the digital pins can be controlled in a similar fashion.

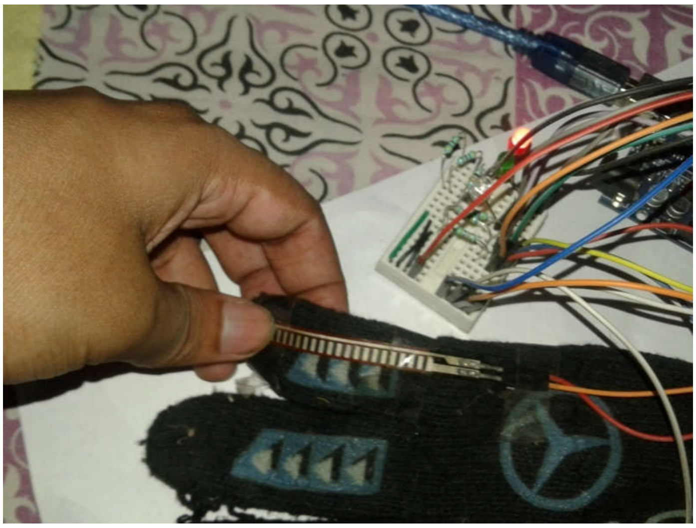
The program that is burned into the microcontroller has several conditional statements. At every five seconds the input of the flex sensor is read by the arduino. When any of these inputs crosses a specified threshold level, the conditional clause present in the code turns on an led. Each led corresponds to each flex sensor, so when a particular flex sensor in bent, the led corresponding to that sensor will light up. When the flex sensor is released, the led is turned off. This can be done for different combinations as well. As we can turn on an led with the help of one flex sensor, we can turn on two leds when two flex sensor is bent. In this state there are two fingers which are bent and 1 finger is relaxed. The output of this state would be "ON ON OFF". Using different combinations like these wee can implement several devices, audios, leds etc. Hence the application of the project is vast and could be used in variety of application where gesture control is required.

- When none of the flex sensors are bent the glove is in a “relaxed” state, this implies that the user does not wish to communicate any messages using the output. Hence all the leds in this state is turned OFF. This state is the most important of all the states as it indicates that no signal



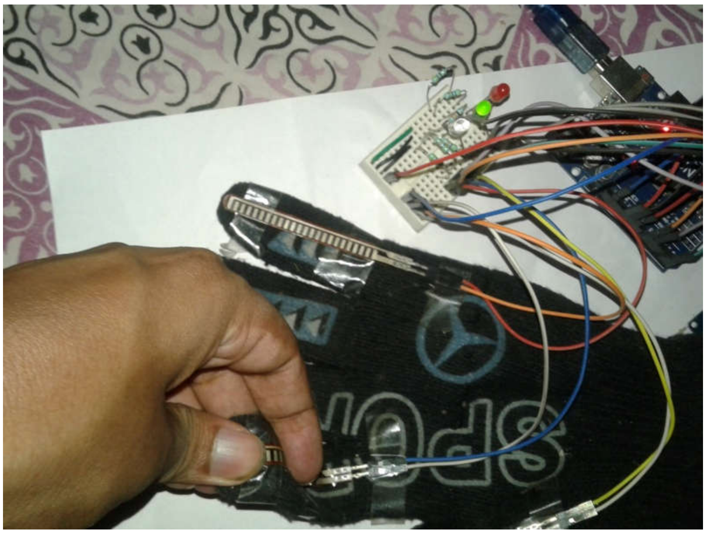
Relaxed state all leds OFF

- When a particular flex sensor is bent, the led corresponding to the flex sensor lights up. As given in the picture, when the led corresponding to the red led is bent, the red led turns ON. It is noteworthy that only a single led corresponding to that particular sensor is on and other leds are OFF.



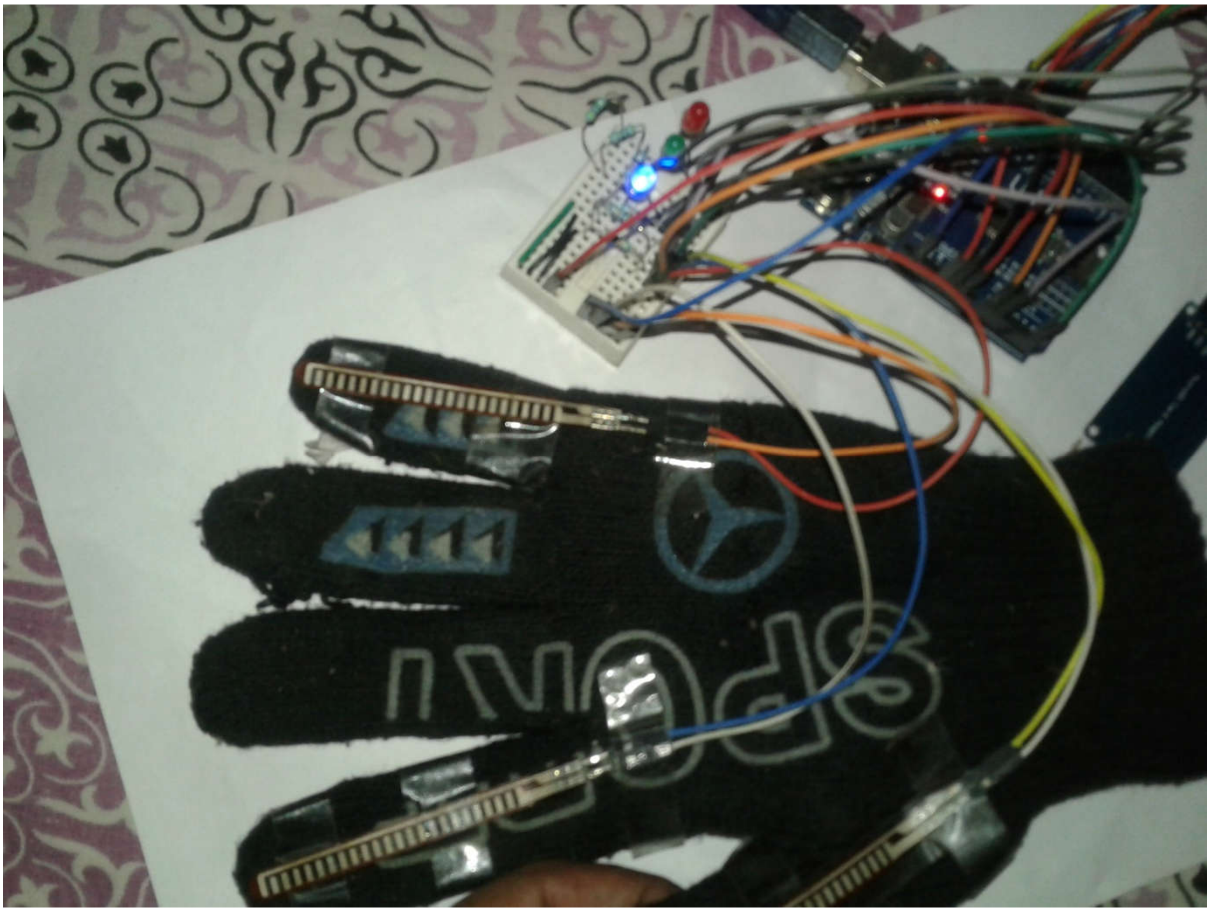
Only red led ON

- When the flex sensor corresponding to green led is bent, the green led lights up as well.



Only green led ON

- Similarly when the flex sensor corresponding to the blue led is bent, the blue led is turned ON.



Only blue led ON

PROBLEMS:-

- The first major problem associated with the project was the calibration of the flex sensors. As the flex sensors were used for longer period of time, the threshold value for bending kept changing, sometimes drastically, this in turn made the leds turn ON and OFF for incorrect gestures.

- Heating effect also de-calibrated the flex sensor as it made the the flex sensor have fluctuating resistance values on prolong uses. This in turn gave incorrect output of the leds.

- Since the maximum bit resolution that arduino can support is 8-bit so playing 8-bit audio using arduino turned out to be a difficult task. As the output from the 8-bit audio was almost in-audible and was difficult to implement. The loudness was not upto the mark either. Even after using an audio amplifier the loudness could not be amplified to a sufficient extent.

- The main problem with the project was the quality of the flex sensors used, as the sensors were the integral part of the project hence using a cost effective sensor may not provide the best results. And since flex sensors are not abundant in the market hence it is always difficult to find quality flex sensors.

APPLICATIONS:-

The applications of this device is only limited to our imagination. Some of the applications are listed below:

- It can be used commercially as a singular input interface for various devices such as computers, cell phones, T.V, etc.
- It can be a used as a game controller in games using multiple hot keys.
- Applications also include controlling robotic arms in medical surgeries and industrial work.

Although the primary objective of this device is to assist the speech impaired, yet further applications are limitless.

CONCLUSION:-

The most satisfying segment of this project has been the fact that the output of the whole procedure exactly resembles that of the expected outcome, resulting from the application of the particular theory which has been applied in this project. Moreover the extension of application that has been long associated with the following project,i.e adding audio effect to the fundamental aspect can serve as a great prospect in the future. However, in this experimental case the system cannot register, nor apprehend any audio file exceeding the size of 8bit which made the audio extension an improbable task.

Natural User Interface (NUI) let users quickly immerse in applications and master control with minimum learning. It is critically important for AR/VR applications and ambient intelligence systems. In burgeoning applications like autonomous drone control and in-car infotainment navigation, NUI can greatly increase the usability.

One key contributor to NUI is touch-less gesture control which allows manipulating virtual objects in a way similar to physical ones. It completely removes the dependency on any mechanical devices like a keyboard or mouse.

One advantage of a wired glove is that they require less computing power. They are useful in cases where haptic feedback is important, like industrial robot control. However, requiring the user to put on a glove is a barrier for mass market adoption.

This experimental project can definitely prove as an ace in the hole for the speech impaired people if produced on large scale with improvised technology. They will be able to get an audio to a particular hand gesture. The extended provision of this fundamental work can also be sensor controlled consumer hardware.

ACKNOWLEDGEMENT:-

Acknowledgements Without the guidance and motivation of our groups of people, this project would not have achieved as much as it did. These groups of people include our advisors, other faculty of the Electronics and Computer Engineering Department at RCCIIT, and other students. This project would not exist without the support our two advisor, Assistant Professor, Mrs Dey. We approached them separately with an idea for a project that was not quite in their research areas, but with enthusiasm they accepted advisor ship of the project. Throughout the project they kept us motivated, and whenever we had a problem, they always had plenty of suggestions about how to solve it. Other faculty of RCCIIT helped us with specific challenges.