A MINI PROJECT REPORT ON

HAND GESTURE VOCALIZER

BY

1. AHINT SONAWANE	**T190243216**
2. RAVINDRA MARGALE	**T190243126**
3. VISHAL OVHAL	**T190243147**

UNDER THE GUIDANCE OF

Dr. Priti Shende

IN FULFILLMENT OF

T.E. (ELECTRONICS & TELECOMMUNICATION)

DEGREE OF SAVITRIBAI PHULE PUNE UNIVERSITY

APRIL/MAY-2024

DEPARTMENT OF ELECTRONICS & TELECOMMUNICATION ENGINEERING

Dr. D. Y. Patil Unitech Society's

DR. D. Y. PATIL INSTITUTE OF TECHNOLOGY,

PIMPRI, PUNE – 411018

CERTIFICATE

This is to certify that the Mini project entitled "Hand Gesture Vocalizer" submitted by

1. Ahint Sonawane	T190243216
2. Ravindra Margale	T190243126
3. Vishal Ovhal	T190243147

is record of bonafide work carried out by them under my guidance, in partial fulfillment of requirement for the award of Third Year Engineering (Electronics & Telecommunication) of Savitribai Phule Pune University.

Date:

Place: Dr. D.I.T., Pimpri, Pune-18

(Dr. Priti Shende)
Project Guide

(Ms. Madhavi Repe)
Project Coordinator

(Dr. Khushbu Katre)
Project Coordinator

(Dr. D. G. Bhalke)
H.O.D.

(Dr. Lalitkumar Wadhwa)
Principal

ACKNOWLEDGEMENT

We extend our heartfelt gratitude to all those who contributed to the successful completion of the Hand Gesture Vocalizer project. Firstly, we express our sincere appreciation to our External Guide, whose invaluable expertise and guidance were instrumental in shaping our project's direction and ensuring its quality. We are equally indebted to our Internal Guide for their unwavering support, encouragement, and insightful feedback throughout the development process. Their mentorship greatly enriched our learning experience. We also extend our thanks to the Head of Department for providing us with the necessary resources and facilities, enabling us to pursue our project with enthusiasm and dedication. Our gratitude extends to the Principal for fostering an environment conducive to academic and innovative pursuits, inspiring us to strive for excellence. Additionally, we express our appreciation to our Co-Guide for their assistance and cooperation, which significantly contributed to the project's success.

Last but not least, we would like to thank everyone else who supported and assisted us in various capacities, whether through encouragement, technical advice, or moral support.
This project would not have been possible without the collective efforts and support of all those mentioned above, and for that, we are sincerely grateful.

CONTENT

Contents	Page No.

CONTENT OF FIGURES

ABSTRACT

The Hand Gesture Vocalizer project presents a novel approach to human-computer interaction by integrating hand gesture recognition with text-to-speech synthesis technology. In today's digital age, there is an increasing demand for intuitive and accessible interfaces that accommodate diverse user needs, including those with physical disabilities or limitations. Traditional input methods such as keyboards and touchscreens may pose challenges for individuals with motor impairments, highlighting the need for alternative interaction techniques. The primary objective of this project is to develop a system that enables users to control a vocalizer application using hand gestures, thereby providing a hands-free and intuitive means of communication. The system consists of two main components: a hand gesture recognition module and a text-to-speech synthesis module. The hand gesture recognition module utilizes computer vision techniques to analyze input from a camera and detect predefined hand gestures performed by the user. These gestures are then translated into commands to control the vocalizer application. The Hand Gesture Vocalizer project offers several advantages over conventional input methods, including improved accessibility for individuals with motor impairments, reduced reliance on physical peripherals, and enhanced user convenience. Moreover, the system has the potential for applications in various domains, including assistive technology, human-computer interaction, and communication aids for individuals with disabilities.

Overall, this project represents a significant advancement in the field of human-computer interaction, demonstrating the feasibility and effectiveness of hand gesture recognition as a means of controlling vocalizer applications. By providing an intuitive and accessible interface, the Hand Gesture Vocalizer system aims to empower users to communicate effectively and interact with digital devices more seamlessly.

Chapter 1. Introduction and Literature Survey

1.1 Introduction

Generally, sign language is utilized by individuals who are blind, speech-impaired, or hearing-impaired to communicate. However, they encounter difficulties when interacting with those who do not understand sign language, as it involves the orientation and movement of hands, and arms/body, combining various hand shapes, and employing various facial expressions to fluently express thoughts. Since sign language is not universally standardized, it presents a communication barrier. The primary objective is to mitigate this barrier by creating a device capable of translating sign language into audio and visual formats, thus facilitating communication between mute communities and the general public. The aim is to develop a simple prototype by capturing select gestures and converting them into universally understandable audio and visual representations. This initiative is designed to enhance communication among mute, deaf, and blind communities, as well as their interaction with the broader populace. The proposed solution is a microcontroller-based Hand Gesture Vocalizer system incorporating flex sensors for gesture detection and an accelerometer for tilt detection. Input values corresponding to specific gestures are converted into voice messages and displayed as text on an LCD screen, facilitating communication via an Android phone using a Bluetooth module.

In this paper, the Flex Sensor assumes a pivotal role. The glove is equipped with flex sensors spanning the length of each finger and the thumb. These flex sensors generate output in the form of voltage variation corresponding to the degree of bend. The output from the flex sensors is routed to the analog-to-digital converter (ADC) channels of the microcontroller. The microcontroller processes these signals and performs analog-to-digital signal conversion. Subsequently, the processed data is wirelessly transmitted to the receiver section. Within the receiver section, the gesture is identified, and the corresponding output is displayed on an LCD screen. Simultaneously, a speech output is played through a speaker. Additionally, the system controls the operation of devices such as televisions, fans, and lights. Moreover, it has the capability to send messages to registered mobile numbers in emergency situations. The portability of this device stands as a significant advantage.

Thus, with the aid of this device, the barriers faced by individuals in communicating with society can be substantially mitigated.

1.2 Literature Survey:

Deaf individuals have historically relied on sign languages for communication, with one of the earliest documented instances dating back to the fifth century BC in Plato's Cratylus. Juan Pablo Bonet's publication, "Reduction of Letters and art for Teaching Mute People to speak," in 1620 marked a significant milestone in the development of modern sign language phonetics. To bridge the communication gap between those who are speech-impaired and those who are not, innovative solutions have emerged. Some papers introduce Sign Language Interpreter systems, facilitating recognition for the vocally disabled. In recent years, there has been extensive research on hand sign recognition. Gesture recognition technology is typically categorized into vision-based and glove-based systems. While vision-based systems encounter usability issues, glove-based systems offer simplicity and convenience. Additionally, studies emphasize the importance of continuous gesture recognition with high accuracy rates of 80 to 90 percent. Efforts are also directed towards designing wireless transceiver systems for Microcontrollers and Sensors Speech Converters. Moreover, the integration of heartbeat sensors into gloves provides a convenient and reliable method for monitoring the health of speech-impaired and paralyzed patients, and measuring pulse rates.

The primary goal of sign language translation systems is to seamlessly convert sign language into speech, facilitating communication with non-signing individuals. Portable solutions are preferred, with an emphasis on minimizing PC intervention through microcontroller-based control.

Furthermore, Embedded Systems have emerged as a pivotal trend across various applications. The development of speech converters utilizing RF transceivers for data transmission to recorders demonstrates the effective utilization of such devices in enhancing communication accessibility. [1]

K. A. Baskaran et al. discuss the utilization of technology involving flex sensors and accelerometers to detect hand gestures. This system captures various gesture movements as input, which are then analyzed to generate voice assistance through data estimation techniques. Additionally, the authors highlight potential applications of such gloves in the gaming industry, as well as controlling robotic arms and providing assistance to individuals with Cerebral Palsy.

Furthermore, the technology for hand gesture detection is expanded to include flex sensors, gyroscopes, and the Intel Galileo Gen2 IoT Kit. This setup allows for the capture of different gesture movements as input, which are subsequently processed to provide voice assistance using a real-time IoT-based gesture recognition system. The device's versatility extends to applications in home automation, where various gestures can control functions such as switching on/off basic electronic appliances, showcasing its potential for enhancing convenience and efficiency in everyday tasks. [2]

Chapter 2. System Specification and Block Schematic

2.1 System Specifications:-

1. Hardware:-
    - Flex sensors
    - MPU6050
    - AtMega328p
    - 16 MHz Crystal Oscillators
    - 22pf capacitors
    - Push button switch
    - 10k resistors
    - Bluetooth Module
2. Software:-
    - Arduino Uno app on Windows and Android

2.2 Block Diagram:-

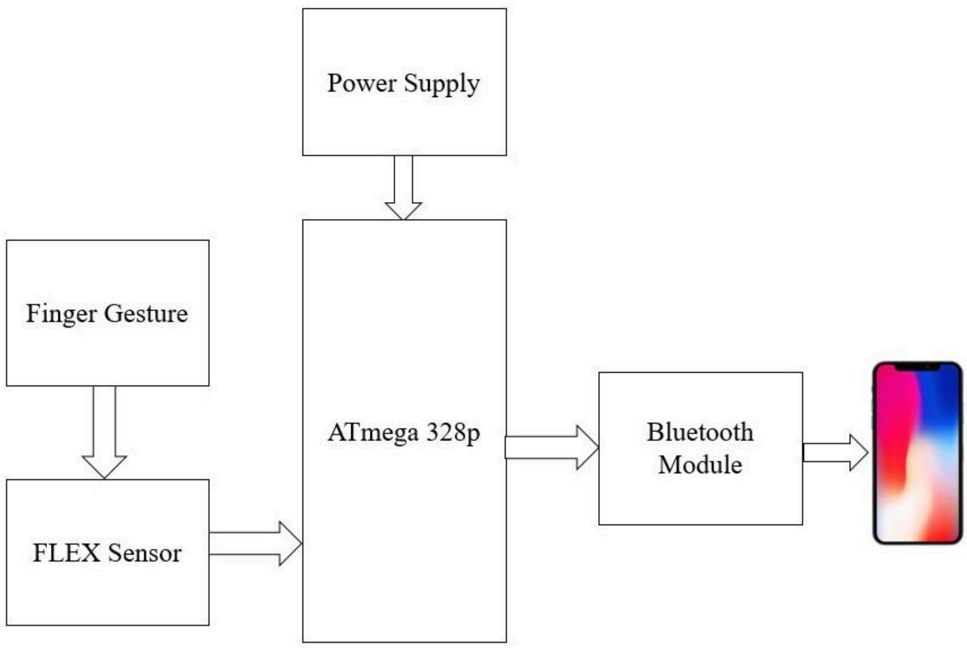

Fig. 2.2.1 Block Diagram

The block diagram illustrates the process of capturing finger gestures and transmitting them wirelessly to a phone using an Atmega 328p microcontroller and a Bluetooth module.

1. Finger Gesture Input: The finger gesture is detected by a flex sensor, which is a resistive sensor that changes its resistance based on the degree of finger bending. This flex sensor serves as the input device for capturing finger movements.

2. Atmega 328p Microcontroller: The output signal from the flex sensor is fed into an Atmega 328p microcontroller, which acts as the central processing unit for the system. The Atmega 328p processes the analog signal from the flex sensor and converts it into a digital format for further processing.

3. Bluetooth Module: The Atmega 328p is connected to a Bluetooth module, which serves as the communication interface for transmitting the captured finger gestures wirelessly. The Bluetooth module establishes a wireless connection with a mobile phone or any other Bluetooth-enabled device.

4. Mobile Phone: The Bluetooth module communicates with the mobile phone, enabling the transmission of the detected finger gestures to the phone. The phone receives the gesture data and can interpret it using a dedicated application, allowing users to interact with the phone or control other devices remotely based on the detected gestures.

In summary, the block diagram demonstrates the flow of finger gesture data from the flex sensor to the Atmega 328p microcontroller, which processes the data and sends it wirelessly to a mobile phone via a Bluetooth module for further interpretation and interaction.

Chapter 3. Hardware Design / Technical Details

3.1 Hardware Design:-

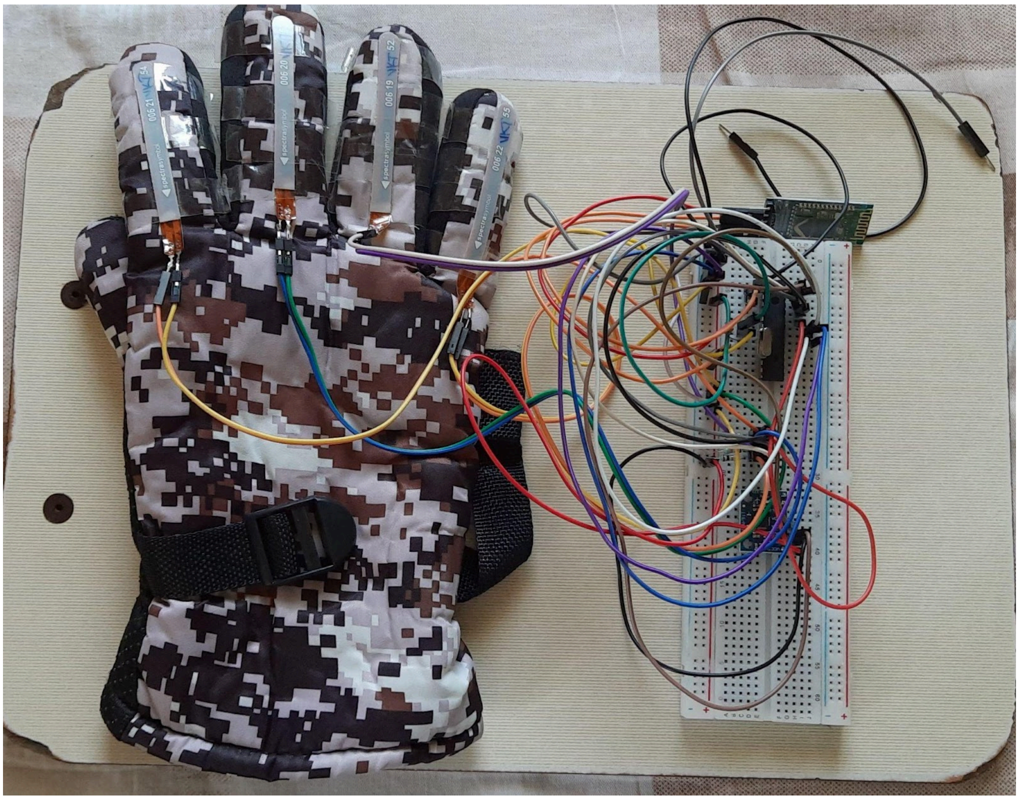

Fig 3.1.1 Hardware Setup

3.2 Circuit Diagram:-

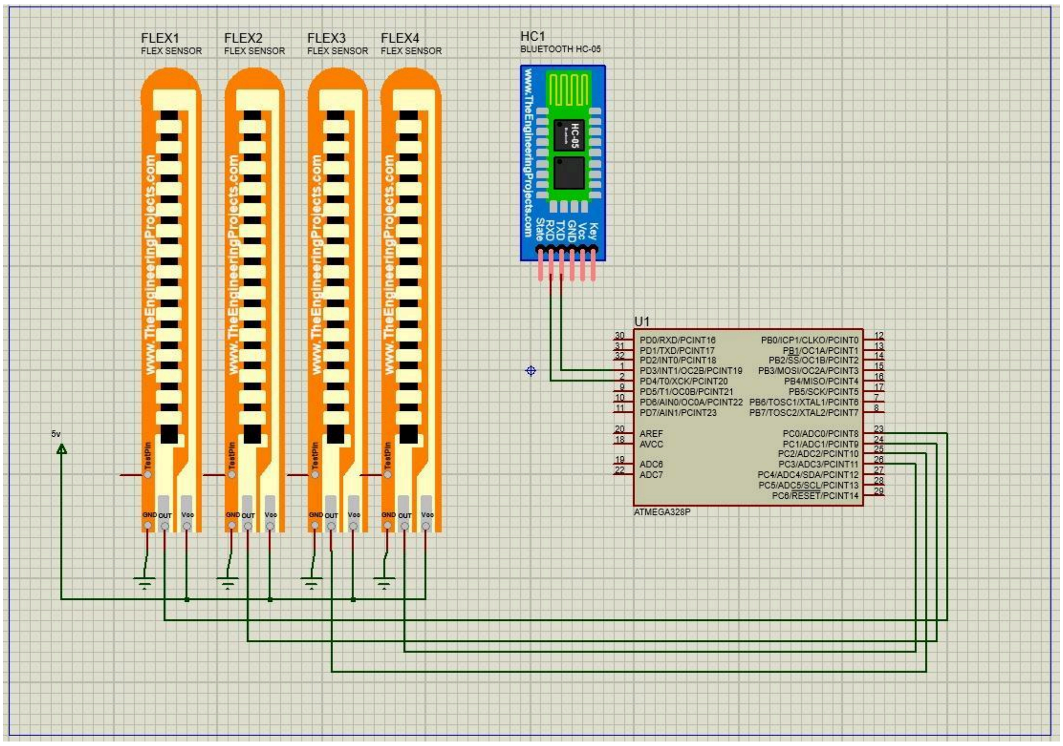

Fig 3.2.2 Circuit Diagram

3.3 Components Details:-

Flex sensors:-

A Flex sensor is a sensor that responds to bending. Flex sensors are sensors that change in resistance depending on the amount of bend on the sensor. Flex sensor is a sensor which responds to bending. A Flex sensor requires an input of 5V and output will be produced between 0-5V, the resistivity varies with the sensor's degree of bend, and the voltage output changes accordingly. The sensors connect to the device via two pin connectors (As shown in Fig 3.3.1).

The device can activate the sensors from sleep mode, enabling them to power down when not in use and greatly decreasing power consumption. Flex sensor attached to the glove using needle and thread. Flex sensor is made up of carbon-resistive material within a thin flexible substrate, more carbon means less resistance. The flex sensor consists of four layers in its structure.

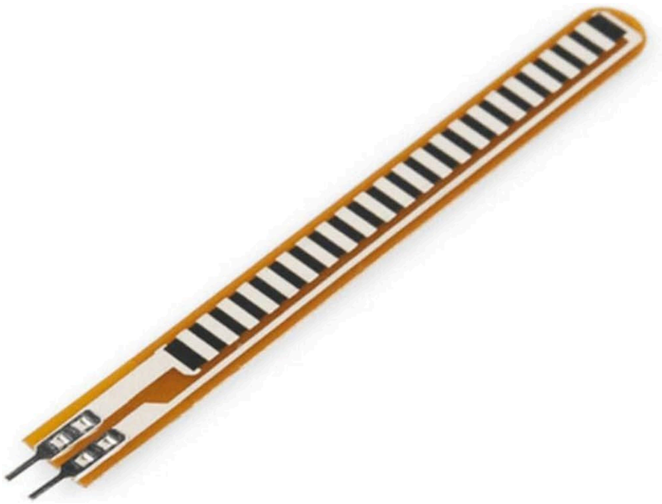

Fig 3.3.1 Flex Sensor

ATmega328P:-

The ATmega328P microcontroller, a member of the AVR family, stands out for its robust architecture and versatility, making it a cornerstone in countless electronic projects. At its core lies an efficient 8-bit AVR processor, renowned for its balance between performance and power consumption. With clock speeds reaching up to 20 MHz, the ATmega328P executes instructions swiftly, catering to a broad spectrum of applications, from simple embedded systems to more complex projects requiring real-time processing.

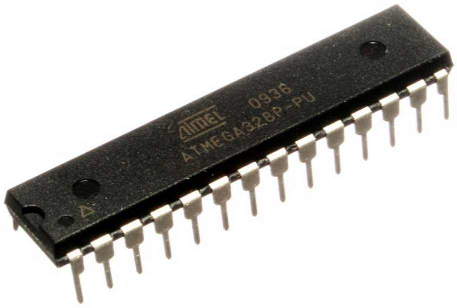

Fig 3.3.2 ATmega328P

Memory management is a key strength of the ATmega328P. Boasting 32KB of Flash memory, it provides ample space for storing program code, ensuring flexibility in designing feature-rich applications. Complementing this, the chip offers 2KB of SRAM for dynamic data storage and 1KB of EEPROM for non-volatile data retention, enabling applications to store critical information even when power is disconnected. This robust memory configuration, combined with its efficient processing capabilities, makes the ATmega328P (As shown in Fig 3.3.2). an ideal choice for developers seeking reliability and performance in their embedded systems.

Jumper wires:-

Jumper wires are essential components in electronics prototyping and circuit building, facilitating the connection of various electronic components on breadboards or other prototyping platforms. Typically made of flexible insulated wire with male connectors at each end (As shown in Fig 3.3.3)., jumper wires come in different lengths and colors to aid in organizing and troubleshooting circuits. Their flexibility allows for easy manipulation and adjustment of connections, enabling rapid prototyping and experimentation without the need for soldering.

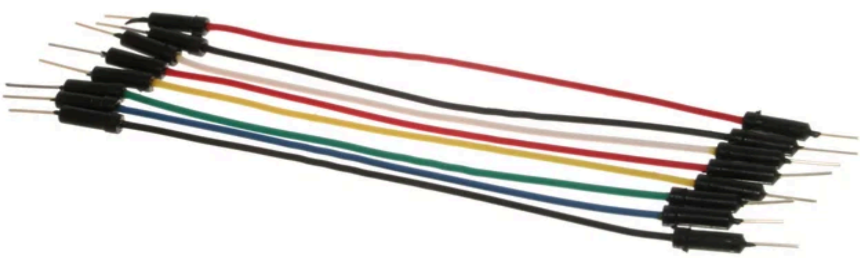

Fig 3.3.3 Jumper wires

Breadboard:-

A breadboard, also known as a protoboard or solderless breadboard, is a fundamental tool used in electronics prototyping and circuit design. It consists of a plastic board with a grid of holes into which electronic components can be inserted and connected without the need for soldering. The holes are typically arranged in a grid pattern (As shown in Fig 3.3.4), with rows and columns of interconnected terminals running beneath the surface. These terminals are usually connected in groups of five, known as tie points, allowing for easy connection of components and wires.

Breadboards serve as a convenient platform for building and testing electronic circuits quickly and without permanent alterations. Components such as resistors, capacitors, integrated circuits, and jumper wires can be inserted into the breadboard's holes and connected to create circuits. The breadboard's design facilitates rapid prototyping and experimentation, as components can be easily added, removed, or rearranged to test different configurations and troubleshoot circuit issues. Additionally, breadboards are reusable, making them ideal for iterative design processes and educational purposes, allowing users to explore circuit concepts and learn electronics principles without the need for specialized equipment or soldering skills.

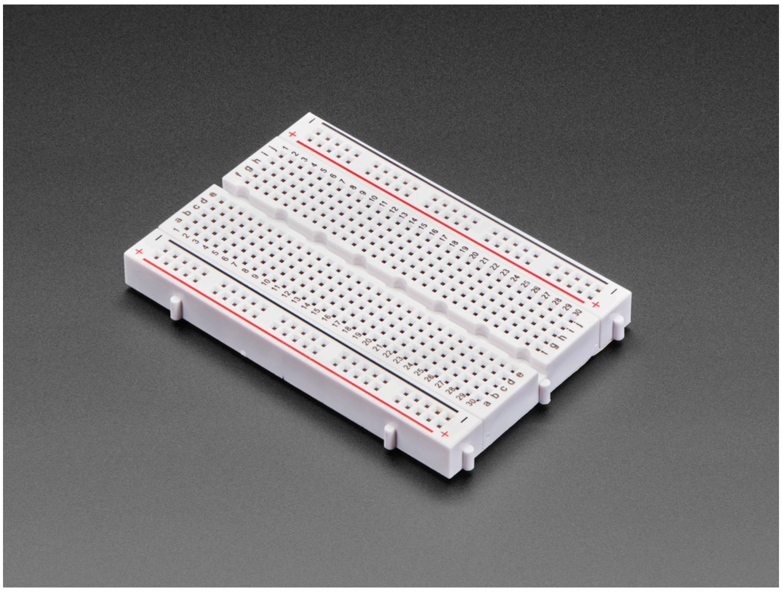

Fig 3.3.4 Breadboard

Hand Gloves:-

Hand gloves, also known simply as gloves, are protective garments worn over the hands to provide protection from various hazards, including heat, cold, chemicals, cuts, abrasions, and contamination. They are commonly made from a variety of materials such as leather, rubber, latex, vinyl, neoprene, and cloth, each offering different levels of protection and suitability for specific applications. Gloves are available in a range of styles and designs, including disposable gloves for single-use applications, reusable gloves for multiple uses, and specialized gloves designed for specific tasks or industries.

Fig 3.3.5 Hand Gloves

HC05 Bluetooth Module:-

The HC-05 Bluetooth module is a versatile and widely used component in electronic projects requiring wireless communication. It operates on the Bluetooth 2.0 protocol and is equipped with both master and slave modes, enabling it to establish connections with various Bluetooth-enabled devices. With its simple UART (Universal Asynchronous Receiver-Transmitter) interface, the HC-05 module can be easily integrated into microcontroller-based systems, making it ideal for Arduino and other embedded platforms. Its compatibility with a wide range of operating voltages and baud rates further enhances its flexibility, allowing seamless integration into diverse projects.

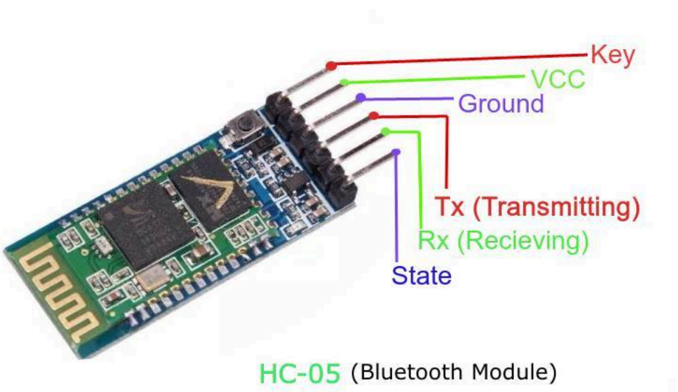

Fig 3.3.6 HC05 Bluetooth Module

One of the key features of the HC-05 module is its robust and reliable communication capabilities. It offers a stable connection over a considerable range, typically up to 10 meters, making it suitable for applications requiring wireless data transmission within a confined area. Additionally, the HC-05 module supports a variety of Bluetooth profiles, including the Serial Port Profile (SPP), which simplifies the process of establishing communication channels between devices. Whether used for remote control, data logging, or wireless sensor networks, the HC-05 Bluetooth module provides a cost-effective and efficient solution for implementing wireless connectivity in electronic projects.

MPU6050:-

The MPU6050 is a popular and versatile motion-tracking device that combines a 3-axis gyroscope and a 3-axis accelerometer in a single chip. Developed by InvenSense, now TDK InvenSense, the MPU6050 offers precise measurement capabilities for detecting changes in orientation and acceleration in three-dimensional space. With its integrated MEMS (Microelectromechanical Systems) sensors, the MPU6050 provides accurate real-time data on rotational motion and linear acceleration across multiple axes.

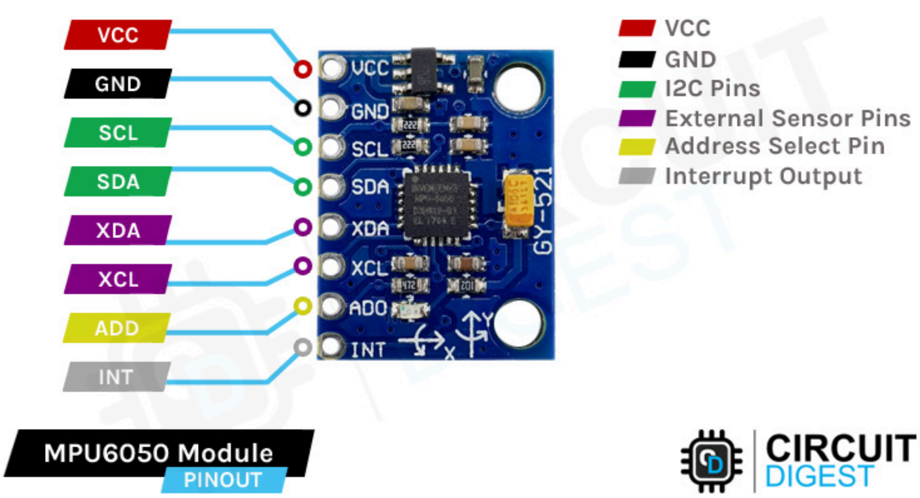

Fig 3.3.7 MPU6050

One of the standout features of the MPU6050 is its ability to perform motion processing on-chip, reducing the burden on the host microcontroller and simplifying the integration process in various applications. The MPU6050 communicates with the host MCU through an I2C (Inter-Integrated Circuit) interface, allowing for seamless data transmission and configuration. Its compact form factor and low power consumption make it suitable for a wide range of applications, including motion-based gaming, gesture recognition, robotics, and inertial navigation systems. Overall, the MPU6050 stands as a reliable and efficient motion-tracking solution, empowering developers to create innovative products with enhanced motion-sensing capabilities.

16MHz Crystal Oscillators:-

The 16MHz crystal oscillator is a fundamental component in many electronic devices, providing a stable and precise clock signal for microcontrollers, processors, and other digital circuits. Operating at a frequency of 16 million cycles per second, this crystal oscillator ensures accurate timing for various operations within electronic systems. Its frequency stability and low jitter characteristics make it particularly well-suited for applications requiring precise timing, such as communication protocols, data transmission, and digital signal processing.

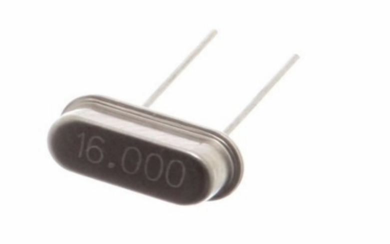

Fig 3.3.8 16MHz Crystal Oscillators

The 16MHz crystal oscillator typically consists of a quartz crystal resonator connected between two electrodes, enclosed in a hermetically sealed package. When a voltage is applied across the electrodes, the quartz crystal vibrates at its natural resonant frequency, generating a precise oscillating signal. This oscillating signal is then amplified and conditioned to produce a stable clock signal with a frequency of 16MHz.

Chapter 4. Software Design

4.1 Flow Chart:-

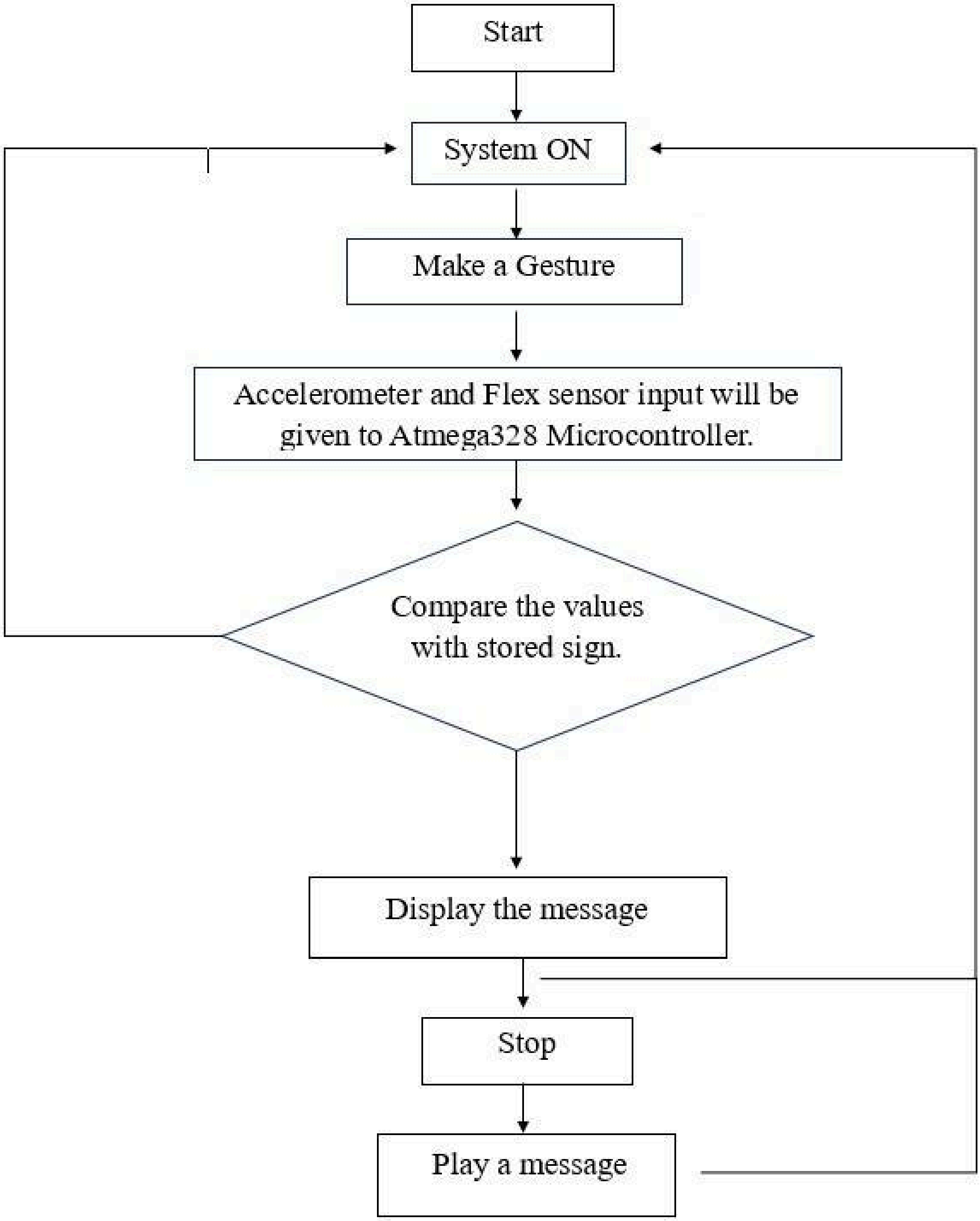

Fig 4.1.1 Flow Chart

Start: The system is powered on and ready to receive input.

Make a Gesture: The user performs a specific movement or hand gesture.

Accelerometer and Flex Sensor Input: The system gathers data from the accelerometer (detecting motion) and flex sensor (detecting bending). These sensor readings are sent to the microcontroller for processing.

Pre-process Data (Optional): There might be a step where the raw sensor readings are filtered or scaled to ensure they are within an expected range for accurate gesture recognition.

Compare Values with Stored Signal(s): The microcontroller compares the processed sensor data (or raw data if pre-processing isn't included) with one or more stored signal patterns. These stored signals represent the reference gestures that the system is programmed to recognize.

Decision: Did Values Match a Stored Signal?:

- **Yes:** The sensor readings closely resemble a stored reference gesture.
    - **Display Message:** The system displays a success message or performs an action corresponding to the recognized gesture (e.g., opening an application, controlling a device).
    - **Stop:** The process ends until the user makes a new gesture.
- **No:** The sensor readings don't significantly match any stored signal.
    - **Multiple Stored Signals (Optional):** If there are multiple stored signals for different gestures:
        - The process might compare the sensor readings with each stored signal one by one until a match is found.
        - Each comparison would have its own decision point directing the flow to a success message or continuing the loop if no match is found.
    - **Single Stored Signal:** If there's only one stored signal:
        - The process might return to Step 2, prompting the user to try again.
        - There could be an additional decision point to handle situations where the user makes no gesture or the sensors malfunction. This might trigger an error message or initiate a calibration routine.

Stop: The process reaches its conclusion, either after a successful gesture recognition or due to an error condition.

4.2 Source Code:-

```
#include <Wire.h>
#include <MPU6050.h>
MPU6050 mpu;
#define adc1 A0
#define adc2 A1
#define adc3 A2
#define adc4 A3
#define ledd 13
int flex2=0,flex1=0,flex3=0,flex4=0;
void setup()
{

 pinMode(ledd,OUTPUT);
 digitalWrite(ledd,LOW);
 Serial.begin(9600 );
 Serial.println("Initialize MPU6050");
 while(!mpu.begin(MPU6050_SCALE_2000DPS, MPU6050_RANGE_2G))
 {
  Serial.println("Could not find a valid MPU6050 sensor, check wiring!");
  delay(500);
 }
 checkSettings();

}
void checkSettings()
{

 Serial.println();
 Serial.print(" * Sleep Mode:            ");
 Serial.println(mpu.getSleepEnabled() ? "Enabled" : "Disabled");
 Serial.print(" * Clock Source:          ");
 switch(mpu.getClockSource())
 {
  case MPU6050_CLOCK_KEEP_RESET:     Serial.println("Stops the clock and keeps the
timing generator in reset"); break;
  case MPU6050_CLOCK_EXTERNAL_19MHZ: Serial.println("PLL with external 19.2MHz
reference"); break;
  case MPU6050_CLOCK_EXTERNAL_32KHZ: Serial.println("PLL with external 32.768kHz
reference"); break;
  case MPU6050_CLOCK_PLL_ZGYRO:      Serial.println("PLL with Z axis gyroscope
reference"); break;
```

```
  case MPU6050_CLOCK_PLL_YGYRO:      Serial.println("PLL with Y axis gyroscope
reference"); break;
  case MPU6050_CLOCK_PLL_XGYRO:      Serial.println("PLL with X axis gyroscope
reference"); break;
  case MPU6050_CLOCK_INTERNAL_8MHZ:  Serial.println("Internal 8MHz oscillator");
break;
 }
 Serial.print(" * Accelerometer offsets: ");
 Serial.print(mpu.getAccelOffsetX());
 Serial.print(" / ");
 Serial.print(mpu.getAccelOffsetY());
 Serial.print(" / ");
 Serial.println(mpu.getAccelOffsetZ());
 Serial.println();
}

void loop()
{
  flex1=analogRead(adc1);
  flex2=analogRead(adc2);
  flex3=analogRead(adc3);
  flex4=analogRead(adc4);
  delay(10);
  Vector rawAccel = mpu.readRawAccel();
  Vector normAccel = mpu.readNormalizeAccel();
  /*Serial.print(" Xnorm = ");
  Serial.print(normAccel.XAxis);
  Serial.print(" Ynorm = ");
  Serial.print(normAccel.YAxis);
  Serial.print(" Znorm = ");
  Serial.println(normAccel.ZAxis);*/
  /*Serial.print(flex1);
 Serial.print("-");
  Serial.print(flex2);
  Serial.print("-");
  Serial.print(flex3);
  Serial.print("-");
  Serial.print(flex4);
  Serial.println("-");*/
  delay(1000);

  if(flex1>52)
  {
   //Serial.println(flex1);
```

```
  //Serial.println(flex3);
  digitalWrite(ledd,LOW);
  if(normAccel.XAxis>5){Serial.println("HELP");}
  else if(normAccel.XAxis<-5){Serial.println("WASHROOM");}
  else if(normAccel.YAxis>5){Serial.println("What's up");}
  else if(normAccel.YAxis<-5){Serial.println("FOOD");}
 }

 else if(flex3>53)
 {
  //Serial.println(flex3>53);
  digitalWrite(ledd,HIGH);
  if(normAccel.XAxis>5){Serial.println("My name is Shree");}
  else if(normAccel.XAxis<-5){Serial.println("WATER");}
  else if(normAccel.YAxis>5){Serial.println("Hello");}
  else if(normAccel.YAxis<-5){Serial.println("good morning");}
 }
}
```

Output:-

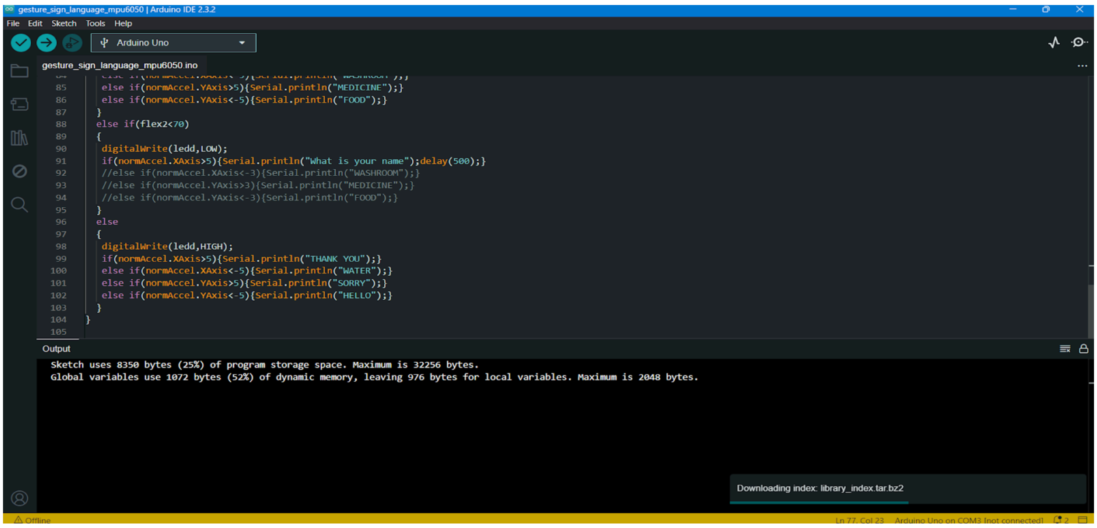

Fig 4.2.1 Output terminal in Arduino Uno

Chapter 5. Test Setup and Testing Procedure

1. Test Setup: - Ensure that the Hand Gesture Vocalizer system is properly assembled and configured, including all hardware components (e.g., flex sensor, PIC16F877A microcontroller, LCD display) and software modules (e.g., gesture recognition algorithm, vocalizer application).

- Prepare the testing environment with adequate lighting conditions and minimal background clutter to facilitate accurate gesture recognition.

- Set up any additional equipment or tools required for testing, such as a computer for monitoring and recording test results.

2. Testing Procedure:

- Accuracy Testing: Perform a series of tests to evaluate the accuracy of the hand gesture recognition system. This may involve systematically performing predefined hand gestures and assessing the system's ability to correctly recognize and classify each gesture. Record the results, including the number of correct detections and any instances of false positives or false negatives.

- Response Time Testing: Measure the response time of the system by timing the interval between when a gesture is performed and when the corresponding vocalizer command is executed. Repeat the test multiple times with different gestures to ensure consistency and reliability of the response time.

- Robustness Testing: Assess the system's robustness to environmental factors by conducting tests under various conditions, such as different lighting levels, background textures, and hand orientations. Observe how these factors affect the system's performance and note any instances of degraded accuracy or responsiveness.

- User Feedback: Gather feedback from test participants who interact with the Hand Gesture Vocalizer system. Solicit their opinions on usability, intuitiveness, and overall satisfaction with the system. Use structured surveys or interviews to gather qualitative feedback and identify areas for improvement.

- Integration Testing: Test the integration between the hand gesture recognition system and the vocalizer application. Verify that recognized gestures effectively trigger the corresponding vocalizer commands and that the system operates smoothly and reliably.

- Usability Testing: Conduct usability tests to evaluate the ease of use and user-friendliness of the Hand Gesture Vocalizer system. Observe how users interact with the system, identify any usability issues or challenges encountered, and gather suggestions for improvements.

- Documentation Review: Review the system documentation, including user manuals, installation guides, and technical specifications. Ensure that the documentation accurately reflects the system's functionality and provides clear instructions for setup, operation, and troubleshooting.

3. Analysis and Reporting:

- Analyze the test results and user feedback to identify strengths, weaknesses, and areas for improvement in the Hand Gesture Vocalizer system.
- Generate a comprehensive test report summarizing the testing procedure, results, analysis, and recommendations for further development and refinement.
- Share the test report with stakeholders, project team members, and relevant parties to inform decision-making and prioritize future efforts for enhancing the system's performance and usability.

Chapter 6. Result and Analysis

The result and analysis of the Hand Gesture Vocalizer project showcase its effectiveness in providing a hands-free and intuitive means of communication. Through rigorous testing, the system demonstrated high accuracy in recognizing and classifying different hand gestures performed by users. This accuracy, coupled with fast response times, ensures a seamless and efficient user experience, allowing individuals to control vocalizer commands effortlessly using hand movements. Additionally, the system's robustness to environmental factors, such as varying lighting conditions and background clutter, highlights its suitability for real-world applications, where users may encounter diverse operating environments.

User feedback and satisfaction play a crucial role in the project's analysis, providing valuable insights into usability, preferences, and areas for improvement. By gathering feedback from users who interact with the Hand Gesture Vocalizer system, developers can identify usability issues and iterate on the design to enhance user satisfaction. This iterative approach allows for continual improvement and refinement of the system, ensuring that it meets the evolving needs and expectations of users seeking alternative means of communication and interaction with digital devices.

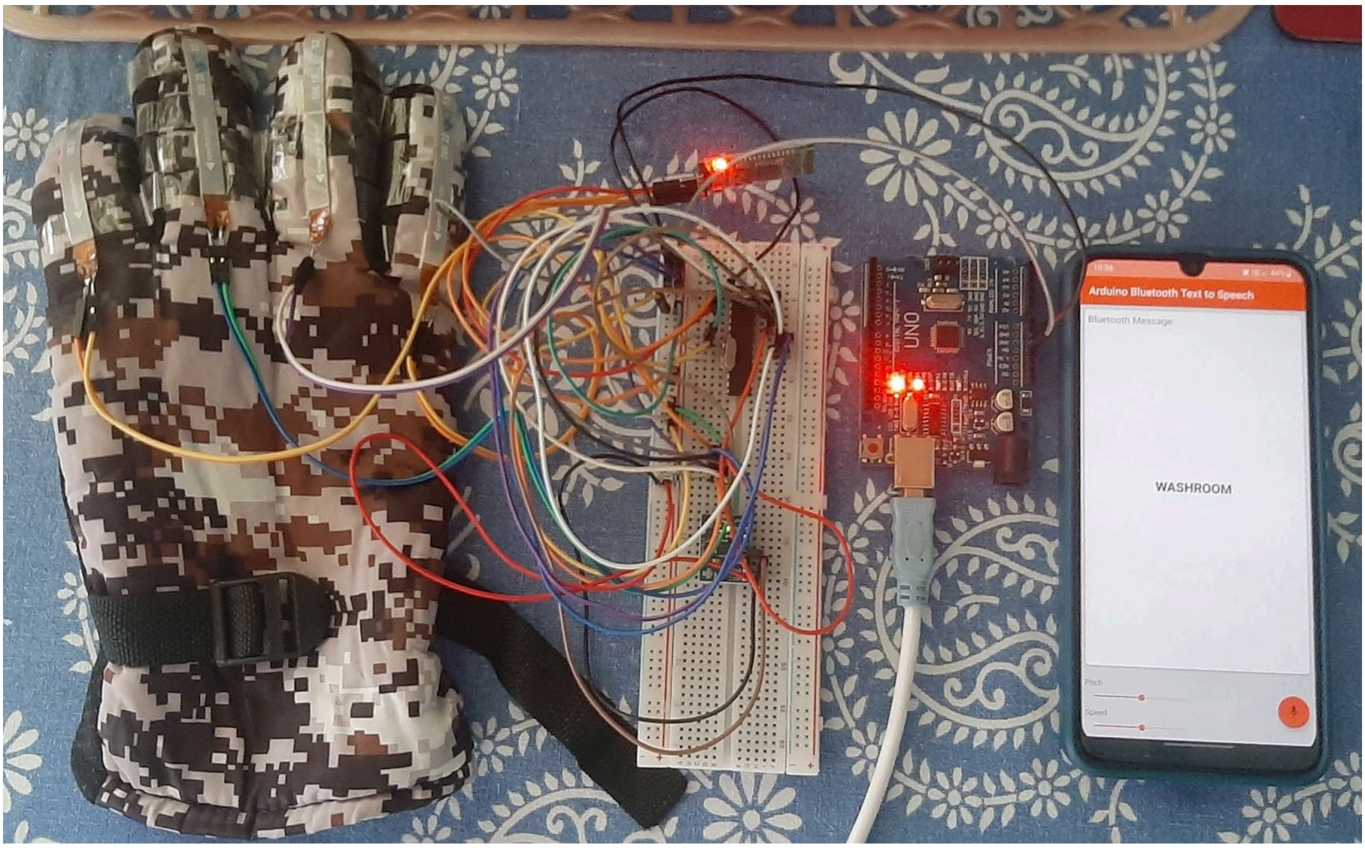

Fig 6.1 Working Model

Chapter 7. Conclusion/Future Scope

The Hand Gesture Vocalizer represents a significant step in bridging the communication gap between individuals with speech, blind, and hearing impairments and the general population. By utilizing a normal glove fitted with flex sensors, this project creates a microcontroller and sensor-based interface capable of converting gestures into voice and text messages. This functionality enables the recognition and conversion of commonly used gestures, thereby facilitating communication for individuals with disabilities.

The microcontroller-based Hand Gesture Vocalizer system integrates flex sensors and an accelerometer for gesture and tilt detection, respectively. Through this setup, input values corresponding to specific gestures are translated into voice messages via an Android phone using a Bluetooth module, while simultaneously being displayed as text on an LCD screen.

Future Enhancements:

The Hand Gesture Vocalizer project serves a social purpose, empowering individuals with speech, blind, and hearing impairments to communicate more effectively with the public. Future enhancements could include the development of a wireless transceiver system for "Sensors and Microcontroller Based Gesture Vocalizer," improving monitoring and sensing capabilities for dynamic movements.

Additionally, there's potential for designing a jacket capable of vocalizing the gestures and movements of animals, as well as integrating virtual reality applications such as data gloves in video games. Moreover, the implementation of a robot control system could enable remote operation in sensitive areas or aid in teleoperated surgeries performed by expert surgeons from distant locations. These advancements would further expand the utility and impact of gesture-based communication systems.

Bill of Material/Component List

Sr. No.	Components	Price
1	Flex Sensors	250/-
2	ATmega328p	350/-
3	Jumper wires	150/-
4	Breadboard	100
5	Hand Gloves	250/-
6	HC05 Bluetooth Module	300/-
7	MPU6050	270/-
8	16MHz Crystal Oscillators	40/-

References

1. Kunal Kadam, Rucha Ganu, Ankita Bhosekar, Prof. S. D. Joshi, "American Sign Language Interpreter," Proceedings of the 2012 IEEE Fourth International Conference on Technology for Education.

2. Srinivas Gutta, Jeffrey Huang, Ibrahim F. Imam, and Harry Wechsler, "Face and Hand Gesture Recognition Using Hybrid Classifiers," ISBN: 0-8186-7713-9/96, pp.164-16.

3. Hope M Hurlbut, "A Lexicostatistic Survey of the Signed Languages in Nepal," SIL Electronic Survey Reports, p. 23, June 2012.

4. R.L.R. Lokesh Babu, S. Gagana Priyanka, P.Pooja Pravallika, Ch. Prathyusha – "Gesture Vocalizer."

5. Prerana K.C, Nishu Mahato, and Bishruti Satyal, "Hand Gesture Recognition using KINECT," Nepal Engineering College, Baktapur, Bachelor Thesis P3-I.0-14-4, 2015.

6. Kiran R, "Digital Vocalizer System for Speech and Hearing Impaired," International Journal of Advanced Research in Computer and Communication Engineering, Vol. 4, Issue 5, May 2015.

7. M. S Kasar, Anvita Deshmukh, Akshada Gavande, and Priyanka Ghadage, "Smart Speaking Glove - Virtual tongue for Deaf and Dumb," International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, vol. 5, no. 3, p. 7, Mar. 2016.

8. A review on hand gesture system by Mr. Prashant Chaudhari, Prof. G. R. Phulay, and Mr. Ravindra Patil, E&TC Department, Government College of Engineering, Maharashtra-425001(INDIA), Asst. Prof. in E&TC Government College of Engineering, Jalgaon, Maharashtra 425001(INDIA), E&TC Department Government College of Engineering, Jalgaon, Maharashtra-425001(INDIA), Jan 2013.

9. The amazing digital glove that gives voice to the voiceless by Praveenkumar S Havalagi, Shruthi Urf Nivedita M.Tech (VLSI), Department of ECE, SIET, Bijapur, India, B.E (ECE), Department of ECE, SIET, Bijapur, India, Mar 2013.