SIGN LANGUAGE TO SPEECH CONVERSION

Submitted in partial fulfillment of the requirements for the award of Bachelor of

Engineering Degree in Electrical and Electronics Engineering

by

DHINESH KUMAR B (Register No. 40140702)

HARISANKAR S (Register No. 40140015)

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

SCHOOL OF ELECTRICAL AND ELECTRONICS ENGINEERING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A++” by NAAC

JEPPIAAR NAGAR,RAJIV GANDHI SALAI,

CHENNAI - 600119,TAMILNADU

APRIL-2024

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A++" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **HARISANKAR S** (Reg. No. 40140015) and **DHINESH KUMAR B** (Reg. No. 40140702) who carried out the project entitled "**SIGN LANGUAGE TO SPEECH CONVERSION**" under my supervision from November 2023 to April 2024.

DR.GT SUNDAR RAJAN

Internal Guide

DR. S. RADHIKA

Head of the Department

Submitted for Viva voce Examination held on ____________________

Name:

Signature:

Internal Examiner **External Examiner**

DECLARATION

We, **HARISANKAR S** (Reg.No.40140015) and **DHINESH KUMAR B** (Reg.No.40140702) hereby declare that the Project Report entitled "**SIGN LANGUAGE TO SPEECH CONVERSION**" done by us under the guidance of **DR.GT SUNDAR RAJAN** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electrical and Electronics Engineering.

1. Dinesh

2. Hari

SIGNATURE OF THE CANDIDATE

DATE:

PLACE: CHENNAI

ACKNOWLEDGEMENT

We are pleased to acknowledge our sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

We convey my thanks to **DR. N.M. NANDHITHA, M.E., Ph.D., Dean**, **School of Electrical and Electronics Engineering**, and **DR. S. RADHIKA , Head of the Department, Dept. of Electrical and Electronics Engineering** for providing us necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to our Project Guide **DR.GT SUNDAR RAJAN**,for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non-teaching staff members of the Department of Electrical and Electronics who were helpful in many ways for the completion of the project.

ABSTRACT

Speech is the easiest way for communication in the world. It becomes difficult for speech impaired people to communicate with normal people as they use sign language for communication. When a speech-impaired person communicates with normal person, the bridge gap between speech impaired and normal masses is too much to fill. The gesture recognition can be done in two ways, Image processing based and sensor-based. The Objective of the project is to design a smart glove for sign language translation that helps an easy way of communication for speech impaired or hearing-impaired people.In this project, glove need to be equipped with sensors such as Flex sensor, Accelerometer, Touch sensor which sense different sign language gestures. Flex sensors are placed on fingers which measure the bending of fingers according to a gesture made. An accelerometer is placed on the palm which measures the location of the hand in X, Y, Z axes. Touch sensors are placed in between the fingers and measures if there is any contact between the fingers. The sensed data from sensors is sent to Arduino UNO board for further processing and transfer data to an android phone via bluetooth module. The data we get will be in the form of text. This text data is then converted into speech through Google text -speech converter.

TABLE OF CONTENT

CHAPTER NO	TITLE			PAGE NO
	ABSTRACT			v
	TABLE OF CONTENT			vi
	LIST OF FIGURES			viii
	LIST OF TABLES			x
1	**INTRODUCTION**			1
	1.1	Project Objective		2
	1.2	Project Outline		3
2	**PROJECT DESCRIPTION**			4
	2.1	Different Sign Language		4
	2.2	Standard Signs		9
	2.3	Block Diagram for sign language		10
	2.4	Flow Chart for sign language		11
	2.5	Advantages		12
3	**SENSORS AND COMPONENTS**			13
	3.1	Flex Sensor		13
		3.1.1	Flex sensor overview	14
		3.1.2	Flex Sensor Working	14
		3.1.3	Reading a Flex Sensor	15
		3.1.4	Basic Flex Circuit and Characteristic	15
	3.2	HC-05 Bluetooth Module		16
		3.2.1	HC-05 Bluetooth Module Pinout	18
		3.2.2	HC-05 Technical Specfication	18

	3.2.3	HC-05 Default Settings	19
	3.2.4	Where to use HC-05 Bluetooth Module	19
	3.2.5	How to Use the HC-05 Bluetooth Module	19
	3.3	Arduino NANO	20
	3.3.1	History	21
	3.3.2	Arduino NANO Pinout	21
	3.3.3	Technical Specifications	22
	3.3.4	Communications	22
	3.3.5	Applications of Arduino NANO	23
4	**SIMULATION THEORY**		25
	4.1	Proteus Description	25
	4.2	Simulation	26
5	**SIMULATION CIRCUIT AND RESULT**		28
	5.1	Simulation Circuit of sign language to speech conversion	28
	5.2	Simulation Result	30
	5.3	Hardware Result	31
6	**CONCLUSION & FUTURE SCOPE**		33
	6.1	Conclusion	33
	6.2	Future Scope	33
	REFERENCE		34

LIST OF FIGURES

Figure No	Name of the Figure	Page No
2.1	American Sign Language	5
2.2	Mexican Sign Language	5
2.3	Chinese Sign Language	6
2.4	French Sign Language	6
2.5	Japanese Sign Language	7
2.6	Arabic Sign Language	7
2.7	Spanish Sign Language	8
2.8	Mexican Sign Language	8
2.9	Ukrainian Sign Language	9
2.10	Standard Signs	9
2.11	Block Diagram for Sign Language	10
2.12	Flow Chart for Sign Language	11
3.1	Flex Sensor	14
3.2	Flex Sensor Working	15
3.3	Basic Flex Circuit	16
3.4	Characteristics of flex sensor	16
3.5	HC-05 Bluetooth Module	18
3.6	ARDUINO NANO PINOUT	21

5.1	Simulation Circuit Diagram of Sign language to speech conversion	29
5.2	Simulation Result	30
5.3	Photocopy of Hardware	31
5.4	Display output	32

LIST OF TABLES

Table No	Name of the table	Page No
3.1	*HC-05 pin description*	17

CHAPTER - 1

INTRODUCTION

In the present world it is very complicated for the deaf & dumb people to talk with the ordinary people as impaired people lacks the amenities which a normal person should own. It actually becomes the same problem of two persons which knows two different language, no one of them knows any common language so its becomes a problem to talk with each other and so they requires a translator physically which may not be always convenient to arrange and this same kind of problem occurs in between the Normal Person and the Deaf person or the Normal Person and the Dumb person.Although technology has been evolving rapidly in this information age, deaf/mute people still use sign language as their only way of communication. Using sign language as a communication tool can be beneficial among those who are familiar with this language, but the problem remains when communicating with the wider community. Sign Language Translator is the appropriate solution that enables deaf/mutepeople to communication fluently through technology in different languages. As sign language is a formal language employing a system of hand gesture for communication (by the deaf). Many projects used glove-based systems for automatic understanding of gestural languages used by the deaf community [1]. The systems developed in these projects differed in characteristics such as number of classifiable signs, which could range from a few dozen to several thousand, types of signs, which could be either static or dynamic, and percentage of signs correctly classified. The simplest systems were limited to understanding of finger spelling or manual alphabets (a series of hand and finger static configurations that indicate letters). Takashi and Kishino [2] and Murakami and Taguchi [3] used a Data Glove for recognition of the Japanese alphabets.For recognition of the American alphabet, Hernadez-Herbollar used an Accele Glove [4]. The more complex systems aimed at understanding sign languages, a series of dynamic hand and finger configurations that indicate words and grammatical structures. For instance, Kim and colleagues used a Data Glove for recognition of the Korean language [5], Kadous a Power Glove for the Australian language [6], Vamplew a CyberGlove for the Australian language [7], Gao and colleagues a Cyber Glove for the Chinese language [8], [9], and Liang and Ouyoung a Data Glove for the Taiwanese language [10]– [12]. Some systems embedded interfaces for translating sign languages into text or vocal outputs [13]– [15]. For instance, the Talking Glove used a Cyber Glove and recorded, recognized, and translated American sign language into text or spoken English [16].Hand movement data acquisition is used in many engineering applications ranging from the analysis of gestures to the biomedical sciences. Glove-based systems represent one of the most important efforts aimed at acquiring hand movement data. While they have been around for over three decades, they keep attracting the interest of researchers from increasingly diverse fields. The development of the most popular devices for hand movement acquisition, glove-based systems, started about 30 years ago and continues to engage a growing number of researchers. We choose to study the glove systems for sign language understanding.

1.1 PROJECT OBJECTIVE

The Aim of the project is to develop a hand glove equipped with sensors such as Flex sensor, Accelerometer, Touch sensor which sense different sign language gestures. Flex sensors are placed on fingers which measure the bending of fingers according to a gesture made. An accelerometer is placed on the palm which measures the location of the hand in X, Y, Z axes.Touch sensors are placed in between the fingers and measures if there is any contact between the fingers. Firstly sensors were simulated to extract the sensed data. Secondly the sensed data from sensors is sent to Arduino UNO board for further processing and transfer data to an android phone via bluetooth module. The data will be in the form of text. This text data is then converted into speech through Google text -speech converter.

1.2 PROJECT OUTLINE

This project report is presented over the four remaining chapters. Chapter 2 describes the sign languages. Chapter 3 presents the principle of operation of sensors and various components used in the project. Chapter 4 explains the concepts of Arduino programming. Chapter 5 presents the simulation results of the various signs simulated using the Arduino simulator. Finally, conclusions are drawn in chapter 6.

CHAPTER- 2

PROJECT DESCRIPTIONS

Sign languages are visual languages that use hand, facial and body movements as means of communication. There are over 135 different sign languages all around THE world including American Sign Language (ASL), Australian Sign Language (Auslan) and British Sign Language (BSL). There are also signed representations of oral languages such as Signed Exact English (SEE) and mixes such as Pidgin Signed English (PSE). Sign language is commonly used as the main form of communication for people who are Deaf or hard of hearing, but sign languages also have a lot to offer for everyone. Sign languages are an extremely important communication tool for many deaf and hard-of-hearing people. Sign languages are the native languages of the Deaf community and provide full access to communication. Although sign languages are used primarily by people who are deaf, they are also used by others, such as people who can hear but can't speak. People who know a sign language are often much better listeners. When using a sign language, a person must engage in constant eye contact with the person who is speaking. Unlike spoken language, with sign languages a person cannot look away from the person speaking and continue to listen. This can be an extremely beneficial habit to have for spoken language as well as sign language. By maintaining eye contact in spoken language, it shows that a person is genuinely interested in what the other is saying.

2.1 Different Sign Languages:

American Sign Language (ASL):Although ASL has the same alphabet as English, ASL is not a subset of the English language. American Sign Language was created independently and it has its own linguistic structure. (It is, in fact, descended from Old French Sign Language.) Signs are also not expressed in the same order as words are in English. This is due to the unique grammar and visual nature of the sign language. ASL is used by roughly half a million people in the world and it is shown in Fig.2.1

Fig 2.1 American Sign Language

British Australian and New Zealand Sign Language (BANZSL):Sharing a sign language alphabet is British Sign Language, Australian Sign Language (Auslan) and New Zealand Sign Language. Unlike ASL, these alphabets use two hands, instead of one and it is shown in fig.2.2

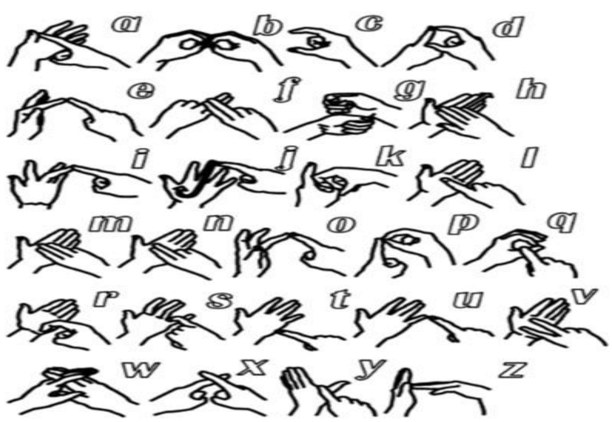

Fig 2.2 Mexican Sign Language

Chinese Sign Language (CSL):CSL's signs are visual representations of written Chinese characters, they use a one handed alphabet as shown in fig.2.3. There are many CSL dialects but the Shanghai dialect is the most common. The language has been developing since the late 1950' s and The Chinese National Association of the

Deaf, is working hard to raise awareness and promote use of the language throughout the country.

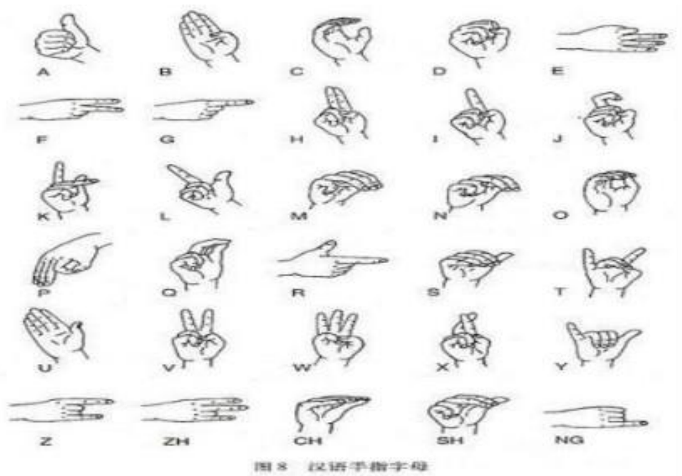

Fig 2.3 Chinese Sign Language

French Sign Language (LSF):French Sign Language is similar to ASL – since it is in fact the origin of ASL – but there are minor differences throughout. LSF also has a pretty fascinating history. LSF is shown in fig.2.4

Fig 2.4 French Sign Language

Japanese Sign Language (JSL) Syllabary:The Japanese Sign Language (JSL) Syllabary is based on the Japanese alphabet, which is made up of phonetic syllables. JSL is known as Nihon Shuwa in Japan and as shown in fig.2.5

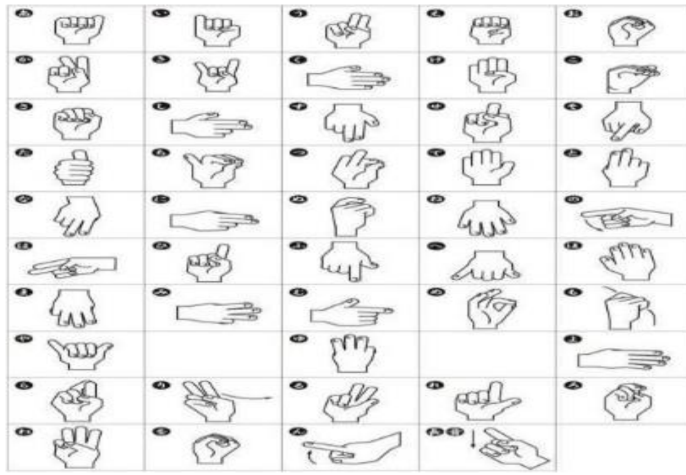

Fig 2.5 Japanese Sign Language

Arabic Sign Language:The Arab sign-language family is a family of sign languages across the Arab Mideast. Data on these languages is somewhat scarce, but a few languages have been distinguished, including Levantine Arabic Sign Language. Arabic sign language is shown in fig.2.6

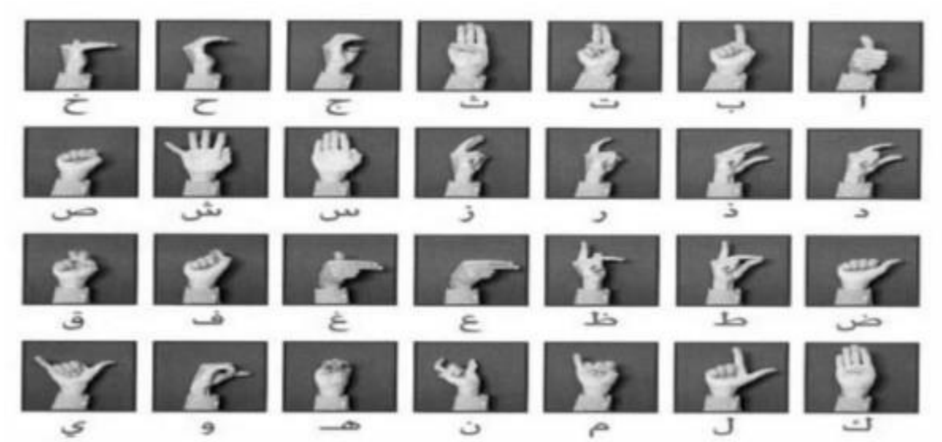

Fig 2.6 Arabic Sign Language

Spanish Sign Language (LSE):Spanish Sign Language is officially recognized by the Spanish Government. It is native to Spain, except Catalonia and Valencia. Many countries that speak Spanish do not use Spanish Sign Language! (See Mexican Sign Language below, for example.) SSL is mainly used in Spain and there is an estimated 100,000 signers of SSL. SSL is completely different from ASL, in the same way that English is different from Spanish. SSL is used across all of Spain as shown in fig.2.7

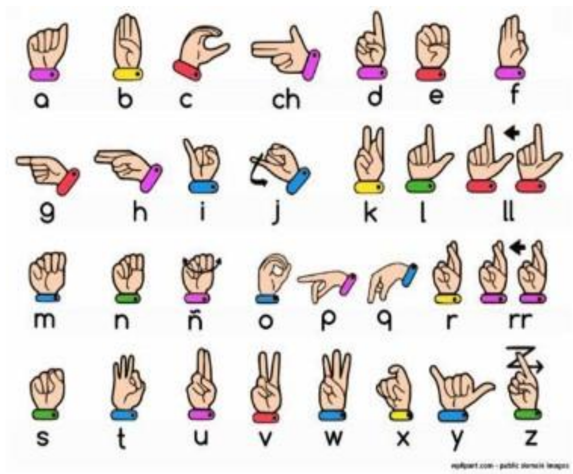

Fig 2.7 Spanish Sign Language

Mexican Sign Language (LSM):Mexican Sign Language (' lengua de señas mexicana' orLSM) is different from Spanish, using different verbs and word order. The majority of people who use Mexican Sign Language reside in Mexico City, Guadalajara and Monterrey. Variation in this language is high between age groups and religious backgrounds. LSM is shown in figure 2.8

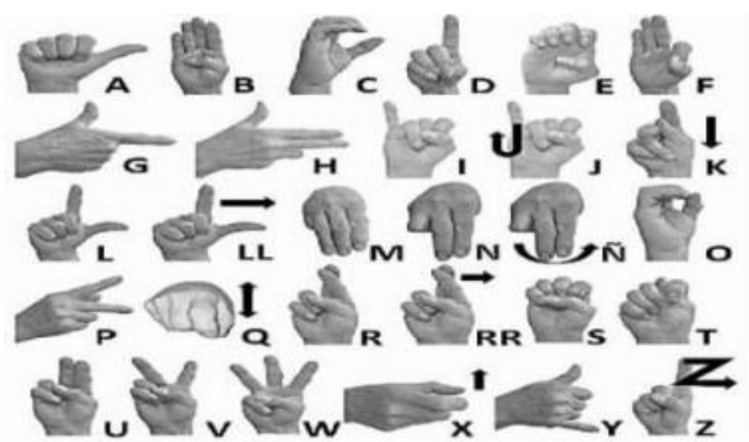

Fig 2.8 Mexican Sign Language

Ukrainian Sign Language (USL):Ukrainian Sign Language is derived from the broad family of French Sign Languages. It uses a one-handed manual alphabet of 33 signs, which make use of the 23 hand shapes of USL and is shown in fig.2.9

Fig 2.9 Ukrainian Sign Language

2.2 STANDARD SIGNS

There is no universal sign language. Different sign languages are used in different countries or regions. For example, British Sign Language (BSL) is a different language from ASL,and Americans who know ASL may not understand BSL. Some countries adopt features of ASL in their sign languages .

Fig 2.10 Standard Signs

It's not always practical to spell out words for everyday interactions. That's where these expressions come in handy! We can use common expressions to meet people, show your appreciation, and communicate with friends. It becomes easy for impaired persons to communicate with normal persons are as shown in fig.2.10.

2.3 BLOCK DIAGRAM FOR SIGN LANGUAGE:

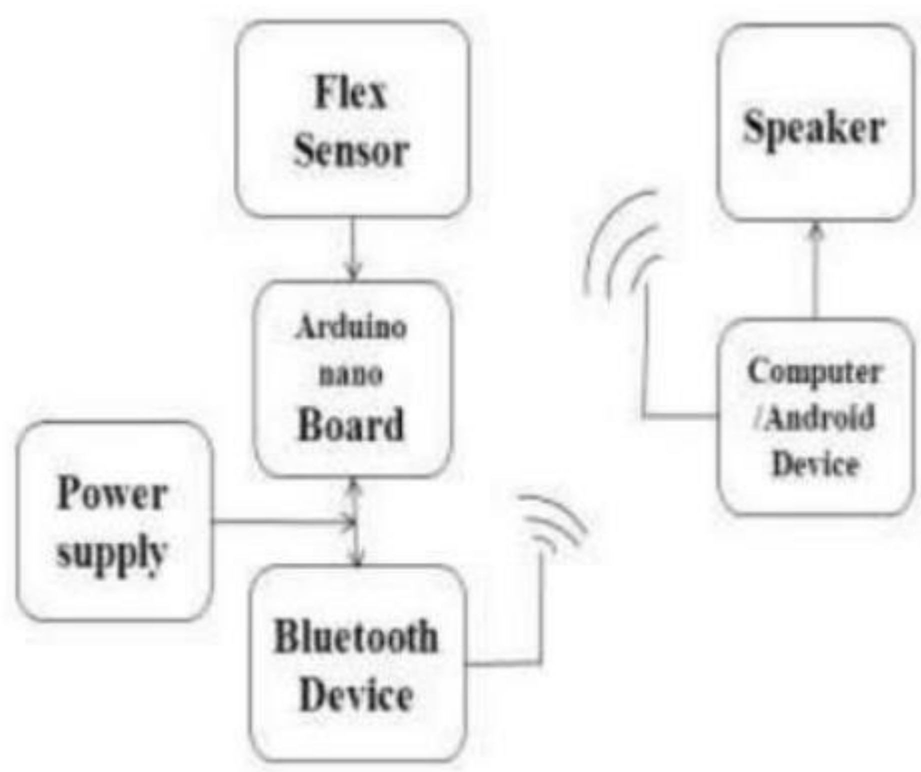

Fig 2.11 Block Diagram for Sign Language

Sign Language Phrases: Embedded phrases expressed in sign language block diagram is shown in above fig 2.11

Sign Language Recognition: The system recognizes and recognizes gestures, hand shapes, and movements in sign language input.

Lexical representation: Translates known symbols into written form, which represents spoken words.

Text-to-speech conversion: Converts the text representation of a sentence into spoken words using a text-to-speech synthesis engine.

Speech Achievement - The outcome of the program, which consists of oral sign language sentences.

2.4 FLOW CHART FOR SIGN LANGUAGE:

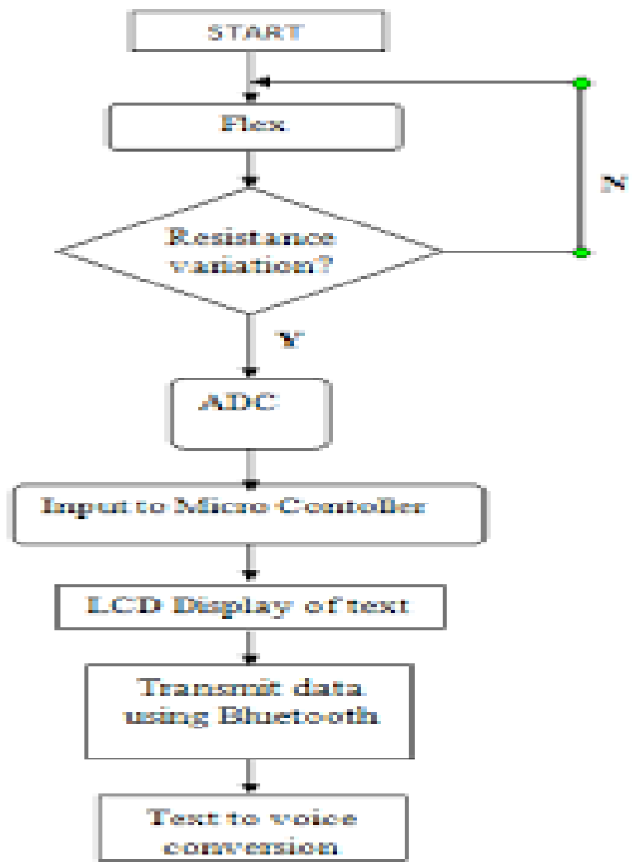

Fig 2.12 Flow Chart for Sign Language

The Flow Chart for Sign Language is shown in the above fig 2.12

Start - Initialization.

Enter Sign Language Language: The user enters a sign language phrase.

Recognize Symbols: The system recognizes symbols and interprets them from the input.

Symbol text translation: Convert known symbols into text representations.

Text-to-Speech Conversion: Converts a text representation to spoken words.

Output Spoken Sentence: Displays a spoken representation of a sign language sentence.

Conclusion: The end of the process.

2.5 ADVANTAGES

- It reduces frustration.
- It increases self esteem.
- It enhances languages and listening skills.
- It enriches relationships.
- It provides a window into your child's world.
- It increase their IQ.

CHAPTER- 3

SENSORS AND COMPONENTS

The proposed system consists of primarily two sections: 1. Transmitter Section 2. Receiver Section.

The devices contained in the transmitter section are:

Flex sensors

Accelerometer Sensor (MPU6050)

Touch Sensor

HC-05 Bluetooth Module

Arduino Uno Microcontroller.

The gloves contain flex sensors which are the main sensors for this product. They are devices which can show variable resistance based on various bend angles. The sensors are connected in a voltage divider circuit such that the resultant analog voltage is sent to one analog port of the micro-controller. The glove is mounted with 4 flex sensors, each on one finger of the glove except the thumb finger.

3.1 FLEX SENSORS

A **flex sensor** or **bend sensor** is a low-cost and easy-to-use sensor specifically designed to measure the amount of deflection or bending. It became popular in the 90s due to its use in the Nintendo Power Glove as a gaming interface. Since then people have been using it as a goniometer to determine joint movement, a door sensor, a bumper switch for wall detection or a pressure sensor on robotic grippers.

3.1.1 Flex Sensor overview

A flex sensor is basically a variable resistor that varies in resistance upon bending. Since the resistance is directly proportional to the amount of bending, it is often called a **Flexible Potentiometer**. Flex sensors are generally available in two sizes: one is 2.2″(5.588cm)long And anotheris 4.5″ (11.43cm) long. A flex sensor consists of a phenolic resin substrate with conductive ink deposited..A segmented conductor is placed on top to form a flexible potentiometer in which resistance changes upon deflection. Flex sensors are designed to flex in only one direction – away from ink. Bending the sensor in another direction may damage it. Also take care not to bend the sensor close to the base, because bottom of the sensor (where the pins are crimped on) is very fragile and can break when bent over is shown in fig 3.1

Fig 3.1 Flex Sensor

3.1.2 Flex Sensor Working

The conductive ink printed on the sensor acts as a resistor. When the sensor is straight,

resistance is about 25k as shown in fig.3.2

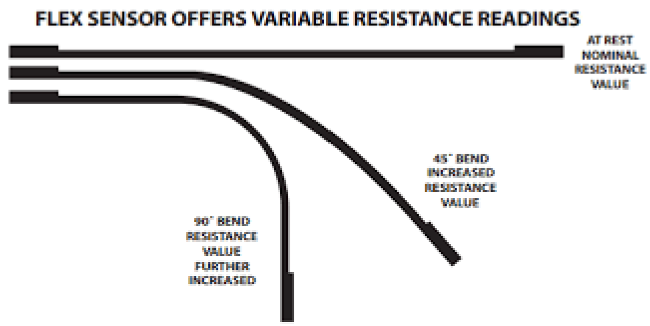

Fig 3.2 Flex Sensor working

When the sensor is bent, conductive layer is stretched, resulting in reduced cross section (imagine stretching a rubber band). This reduced cross section results in an increased resistance. At 90° angle, this resistance is about 100KΩ. When the sensor is straightened again, the resistance returns to its original value. By measuring the resistance, you can determine how much the sensor is bent.

3.1.3 Reading a Flex Sensor

The easiest way to read the flex sensor is to connect it with a fixed value resistor (usually 47kΩ) to create a voltage divider. To do this you connect one end of the sensor to Power and the other to a pull-down resistor as shown below. Then the point between the fixed value pull-down resistor and the flex sensor is connected to the ADC input of an Arduino. This Way you can create a variable voltage output, which can be read by Arduino's A DC input.

Note that the output voltage you measure is the voltage drop across the pull-down resistor, not across the flex sensor. In the shown configuration, the output voltage decreases with increasing bend radius.

3.1.4 Basic Flex circuit and Characteristics

Figure 3.3 shows circuit of basic flex sensor which consist of two or three sensors which are connected. The outputs from the flex sensors are given as inputs to op-amp and used a non inverted style setup to amplify their voltage. The greater the degree of bending the lower the output voltage

where R1 is the other input resistor to the non-inverting terminal and the characteristics are shown in fig.3.4

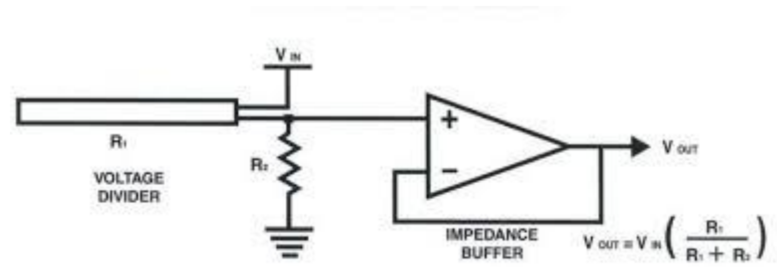

Fig 3.3 Basic Flex Circuit

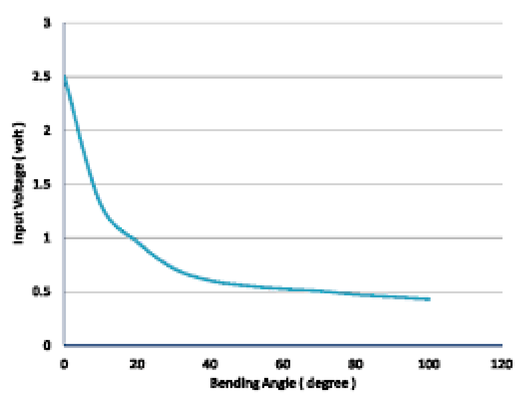

Fig 3.4 Characteristics of flex sensor

3.2 HC-05 BLUETOOTH MODULE

HC-05 Bluetooth Module is an easy to use Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. Its communication is via serial communication which makes an easy way to interface with controller or PC.HC-05 Bluetooth module provides switching mode between master and slave mode which means it able to use neither receiving nor transmitting data are shown in the table 3.1

Table 3.1 HC-05 pin description

1	*Key*	This pin is used to toggle between Data Mode (set low) and AT command mode (set high). By default it is in Data mode.
2	*Vcc*	Powers the module. Connect to +5V Supply voltage
3	*Ground*	Ground pin of module, connect to system ground.
4	*TXD-* transmitter	Transmits Serial Data. Everything received via Bluetooth will be given out by this in as serial data.
5	*RXD-Receiver*	Receive Serial Data. Every serial data given to this pin will be broadcasted via bluetooth
6	*State*	The state pin is connected to on board LED, it can be used as a feedback to check in Bluetooth is working properly.
7	*LED*	Indicates the status of Module Blink once in 2 sec: Module has entered Command Mode Repeated Blinking: Waiting for connection in Data Mode Blink twice in 1 sec: Connection successful in Data Mode

8	*button*	Used to control the Key/Enable pin to toggle between Data and command Mode

3.2.1 HC–05 BLUETOOTH MODULE PINOUT

The HC-05 Bluetooth module pinout is shown in below fig 3.5

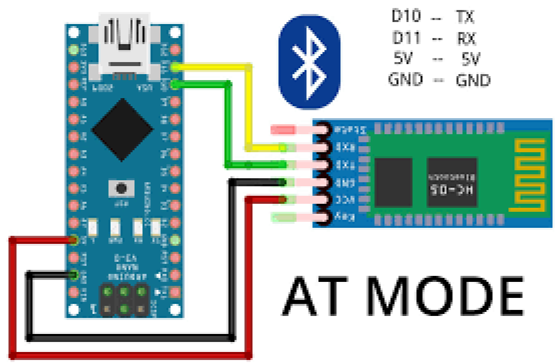

Fig 3.5 HC-05 BLUETOOTH MODULE

3.2.2 HC–05Technical Specifications

- Serial Bluetooth module for Arduino and other microcontrollers
- Operating Voltage: 4V to 6V (Typically +5V)
- Operating Current: 30mA
- Range: <100m
- Works with Serial communication (USART) and TTL compatible
- Follows IEEE 802.15.1 standardized protocol
- Uses Frequency-Hopping Spread spectrum (FHSS)
- Can operate in Master, Slave or Master/Slave mode
- Can be easily interfaced with Laptop or Mobile phones with Bluetooth
- Supported baud rate: 9600,19200,38400,57600,115200,230400,460800.

3.2.3 HC–05 Default Settings

- Default Bluetooth Name: "HC-05"
- Default Password: 1234 or 0000 Default Communication: Slave Default Mode: Data Mode
- Data Mode Baud Rate: 9600, 8, N, 1
- Command Mode Baud Rate: 38400, 8, N, 1 Default firmware: LINVO

3.2.4 Where to use HC-05 Bluetooth module

The **HC-05** is a very cool module which can add two-way(full-duplex)wireless functionality to your projects.You can use this module to communicate between two microcontrollers like Arduino or communicate with any device with Bluetooth functionality like a Phone or Laptop. There are many android applications that are already available which makes this process a lot easier. The module communicates with the help of USART at 9600 baud rates hence it is easy to interface with any microcontroller that supports USART. We can also configure the default values of the module by using the command mode. So, if you looking for a Wireless module that could transfer data from your computer or mobile phone to microcontroller or vice versa then this module might be the right choice for you. However, do not expect this module to transfer multimedia like photos or songs; you might have to look into the CSR8645 module for that.

3.2.5 How to Use the HC – 05 Bluetooth module

The **HC-05** has two operating modes, one is the Data mode in which it can send and receive data from other Bluetooth devices and the other is the AT Command mode where the default device settings can be changed. We can operate the device in either of these two modes by using the key pin as explained in the pin description.

It is very easy to pair the HC-05 module with microcontrollers because it operates using the Serial Port Protocol (SPP). Simply power the module with +5V and connect the Rx pin of the module to the Tx of MCU and Tx pin of module to Rx of MCU as shown in the Figure below

During power up the key pin can be grounded to enter into Command mode, if left free it will by default enter into the data mode. As soon as the module is powered you should be able to discover the Bluetooth device as“HC-05” then connect with it using the defaultpassword 1234 and start communicating with it.

3.3 Arduino NANO

The **Arduino Nano** is an open-source breadboard-friendly microcontroller board based on the Microchip ATmega328P microcontroller (MCU) and developed by Arduino.cc and initially released in 2008. It offers the same connectivity and specs of the Arduino Uno board in a smaller form factor.

The Arduino Nano is equipped with 30 male I/O headers, in a DIP-30-like configuration, which can be programmed using the Arduino Software integrated development environment (IDE), which is common to all Arduino boards and running both online and offline. The board can be powered through a type-B mini-USB cable or from a 9 V battery.

The Arduino Nano is a compact and versatile microcontroller board based on the ATmega328P microcontroller chip. It is designed to provide an easy and affordable platform for electronics enthusiasts, hobbyists, and professionals to create interactive projects.

Arduino Nano is one type of microcontroller board, and it is designed by Arduino.cc. It can be built with a microcontroller like Atmega328. This microcontroller is also used in Arduino UNO. It is a small size board and also flexible with a wide variety of applications. Other Arduino boards mainly include Arduino Mega, Arduino Pro Mini, Arduino UNO, Arduino YUN, Arduino Lilypad, Arduino Leonardo, and Arduino Due. And other development boards are AVR Development Board, PIC Development Board, Raspberry Pi, Intel Edison, MSP430 Launchpad, and ESP32 board.

This board has many functions and features like an Arduino Duemilanove board. However, this Nano board is different in packaging. It doesn’t have any DC jack so that the power supply can be given using a small USB port otherwise straightly connected to the pins like VCC & GND. This board can be supplied with 6 to 20volts using a mini USB port on the board.

3.3.1 HISTORY

In 2008, the Arduino Nano was released. In 2019, Arduino released the **Arduino Nano Every**, a pin-equivalent evolution of the Nano. It features a ATmega4809 microcontroller (MCU) with three times the RAM.The Arduino Nano is a compact microcontroller board based on the ATmega328 or ATmega168 microcontroller chip. It was developed to provide a smaller and more cost-effective alternative to the Arduino Uno, while maintaining compatibility with the Arduino software and hardware ecosystem.

3.3.2 ARDUINO NANO PINOUT

The Arduino Nano Pinout is shown in below fig 3.6

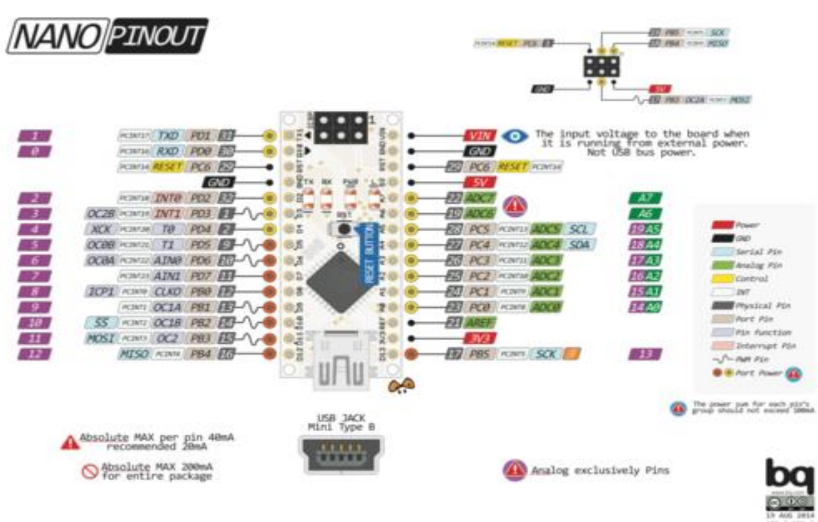

Fig 3.6 ARDUINO NANO PINOUT

3.3.3 Technical specifications

- Microcontroller: Microchip ATmega328P
- Operating voltage: 5 volts
- Input voltage: 5 to 20 volts
- Digital I/O pins: 14 (6 optional PWM outputs)
- Analog input pins: 8
- DC per I/O pin: 40 mA
- DC for 3.3 V pin: 50 mA
- Flash memory: 32 KB, of which 2 KB is used by bootloader
- SRAM: 2 KB
- EEPROM: 1 KB
- Clock speed: 16 MHz
- Length: 45 mm
- Width: 18 mm
- Mass: 7 g
- USB: Mini-USB Type-B
- ICSP Header: Yes
- DC Power Jack: No

3.3.4 Communication

The Arduino Nano has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provides UART TTL serial (5V) communication, which is available on digital pins 0 (RX) and 1 (TX).

An FTDI FT232RL on the board channels this serial communication over USB and the FTDI drivers (included with the Arduino firmware) provide a virtual com port to software on the computer. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board flash when data is being transmitted via the FTDI chip and the USB connection to the computer (but not for serial communication on pins 0 and 1). A SoftwareSerial library allows for serial communication on any of the Nano's digital pins. The

ATmega328 also supports I2C and SPI communication. The Arduino software includes the Wire library to simplify use of the I2C bus.

Automatic (software) reset

Rather than requiring a physical press of the reset button before an upload, the Arduino Nano is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the FT232RL is connected to the reset line of the ATmega328 via a 100 nanofarad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip.

This setup has other implications. When the Nano is connected to a computer running Mac OS X or Linux, it resets each time a connection is made to it from software (via USB). For the following half-second or so, the bootloader is running on the Nano. While it is programmed to ignore malformed data (i.e. anything besides an upload of new code), it will intercept the first few bytes of data sent to the board after a connection is opened.

3.3.5 Applications of Arduino Nano

These boards are used to build Arduino Nano projects by reading inputs of a sensor, a button, or a finger and gives an output by turning motor or LED ON, or and some of the applications are listed below.

Samples of electronic systems & products

- Automation
- Several DIY projects
- Control Systems
- Embedded Systems
- Robotics
- Instrumentation

Thus, this is all about an overview of Arduino nano datasheet. From the above information finally, we can conclude that for the beginners who are new to electronics, this Nano board is extremely suggested to go for this board due to its features like low cost and very simple to use in different applications. This board can simply connect to any computer throughout its mini USB port.

CHAPTER- 4

SIMULATION THEORY

4.1 PROTEUS DESCRIPTION

Proteus is a famous electronic layout automation (EDA) software program that offers a platform for designing, simulating, and analyzing electronic circuits. It is utilized by engineers, hobbyists, and students to create schematic diagrams, PCB layouts, and three-D fashions of digital circuits.

Proteus gives a wide range of capabilities and tools for electronic layout, such as:

Schematic Capture: Proteus lets in customers to create and edit schematic diagrams of digital circuits the usage of a drag-and-drop interface. It helps numerous additives, along with resistors, capacitors, transistors, ICs, and microcontrollers.

Simulation: Proteus provides a simulation environment that permits users to test and analyze the behavior of digital circuits before constructing them. It supports diverse simulation models, which includes SPICE, VHDL, and Verilog.

PCB Layout: Proteus lets in users to create PCB layouts of digital circuits the use of a graphical interface. It supports numerous PCB layout rules, which includes trace width, clearance, and thru length.

3-d Modeling: Proteus offers a 3-D modeling environment that allows users to visualize and examine the bodily properties of digital circuits. It helps various 3-D fashions, inclusive of enclosures, connectors, and cables.

Library: Proteus consists of a huge library of electronic additives, such as resistors, capacitors, transistors, ICs, and microcontrollers, that may be utilized in electronic designs.

Proteus is widely used in various industries, inclusive of car, aerospace, scientific, and purchaser electronics, for designing and simulating electronic circuits. It is to be had in diverse editions, together with Proteus 8 Professional, Proteus 8 Lite, and Proteus eight SCH, that cater to exclusive design wishes and budgets.

Simulation is the process of creating a digital version of a device or a system and jogging experiments on it to analyze its conduct and overall performance. In the context of electronic circuits, simulation refers back to the procedure of making a digital model of an digital circuit and going for walks experiments on it to analyze its electric conduct and performance.

4.2 SIMULATION

Simulation is an vital tool in digital design and evaluation, as it allows designers to check and optimize digital circuits before building them. Simulation can assist designers to pick out and fix layout mistakes, compare the overall performance of the circuit beneath various situations, and optimize the circuit for value, energy consumption, and different factors.

There are diverse simulation tools and software available for electronic circuits, such as Proteus, LTspice, PSpice, and Multisim. These gear offer a graphical interface for growing and modifying digital schematics, in addition to a simulation engine for analyzing the electric behavior of the circuit.

Simulation may be used to research numerous electric parameters of the circuit, including voltage, modern, strength, frequency, and impedance. It also can be used to investigate the transient response of the circuit, which includes the start-up conduct, the reaction to enter indicators, and the steadiness of the circuit.

Simulation can be accomplished at numerous degrees of abstraction, together with circuit-level simulation, system-level simulation, and behavioral simulation. Circuit-stage simulation includes modeling the individual additives of the circuit, including resistors, capacitors, and transistors, and analyzing their electric conduct. System-level simulation includes modeling the complete gadget, such as a digital or analog circuit, and studying its behavior and performance.

Behavioral simulation entails modeling the excessive-level behavior of the circuit, which includes its functionality and overall performance, without modeling the man or woman components.

Simulation is a powerful device in electronic design and analysis, as it permits designers to test and optimize electronic circuits before constructing them, reducing the time, cost, and risk related to physical prototyping and checking out.

CHAPTER 5

SIMULATION CIRCUIT AND RESULTS

5.1 SIMULATION CIRCUIT OF SIGN LANGUAGE TO SPEECH CONVERSION:

Sign language to speech conversion in Proteus, a software tool mainly used for electronic circuit design and simulation, can be indirect because it is not specifically designed for natural language processing tasks such as speech recognition or integration We can simulate . . . .

Input Interface: Create an input interface where the user can input sign language signals. This can be a simple button representing any gesture or switch emulation.

Gesture Recognition Module: Develop a sign language gesture recognition module. You can use logic circuits or microcontroller simulation to process the input signals and recognize known gestures.

Text Representation Module: Once the gesture is detected, simulate a module that converts the detected gesture into the text representation. This can be done by mapping each gesture to the corresponding text representation using look-up tables or simple logic circuits.

Text-to-Speech Module: Imitate a module that converts synthesized text into speech. Although Proteus does not directly support natural language processing, you can easily create part of this module using pre-recorded speech samples inspired by generated text

Output Interface: Creates an output interface where the compiled speech can be played. This can be a speaker sample connected to the output of the text-to-speech module.

Simulation: Integrate all modules in the Proteus environment to simulate the entire system. You can use virtual switches or buttons to simulate user input gestures and watch the system convert them into synthesized speech. Simulation Circuit Diagram of Sign language to speech conversion are shown in fig. 5.1 .

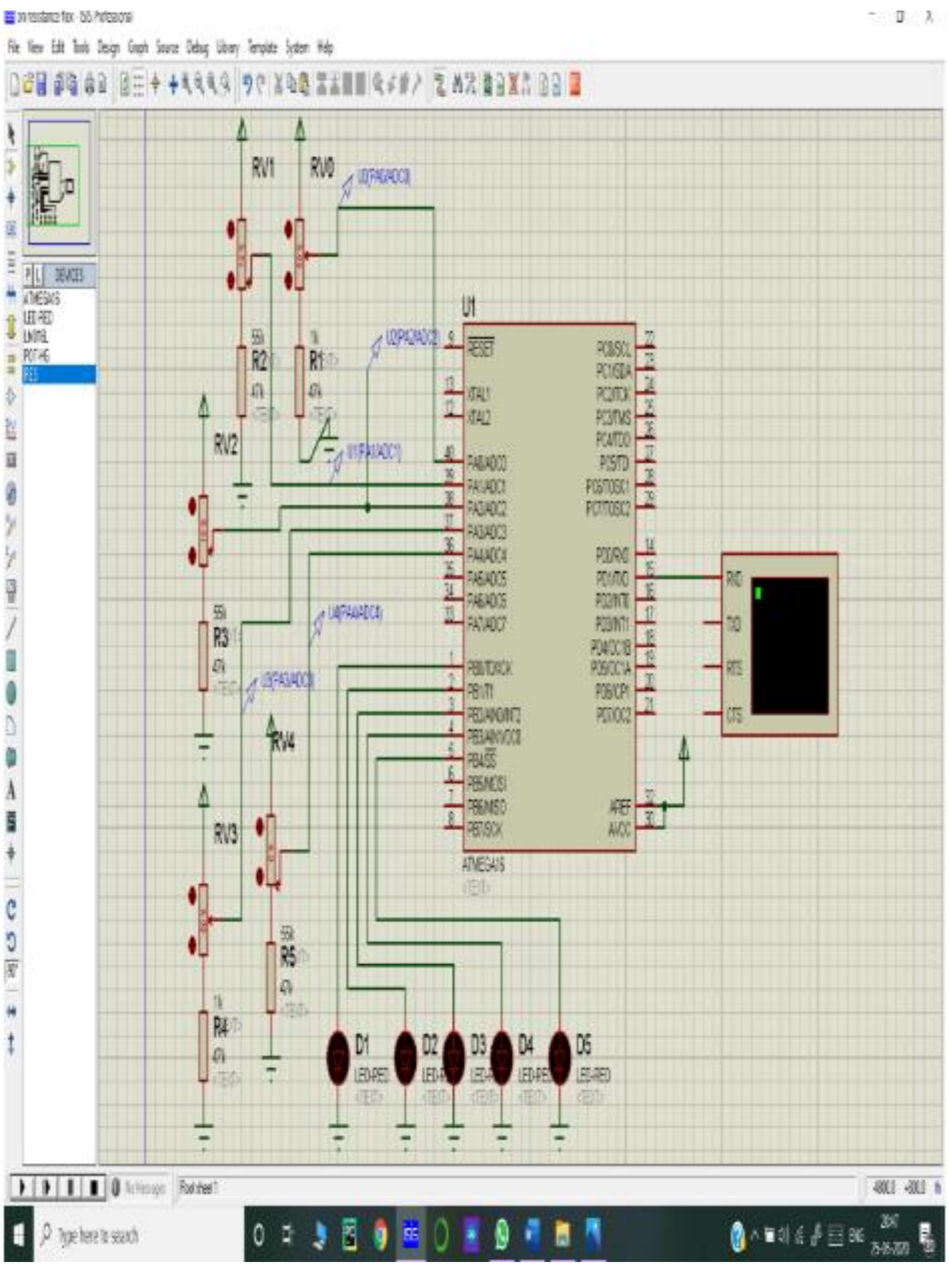

Fig. 5.1 Simulation Circuit Diagram of Sign language to speech conversion

5.2 SIMULATION RESULTS

Output of Proteus simulation for sign language to speech conversion.The simulation Result are shown in fig.5.2.

Fig 5.2 Simulation Result

1. "The simulation begins by capturing input from virtual sensors representing sign language gestures."

2. "The captured gestures are then processed by simulated recognition algorithms to identify the intended signs."

3. "Once the signs are recognized, they are translated into text representations using predefined mappings."

4. "The text representations are then passed to a virtual text-to-speech module, which synthesizes the spoken words."

5. "Finally, the synthesized speech is output through virtual speakers, allowing the user to hear the spoken representation of the sign language input."

5.3 HARDWARE RESULTS

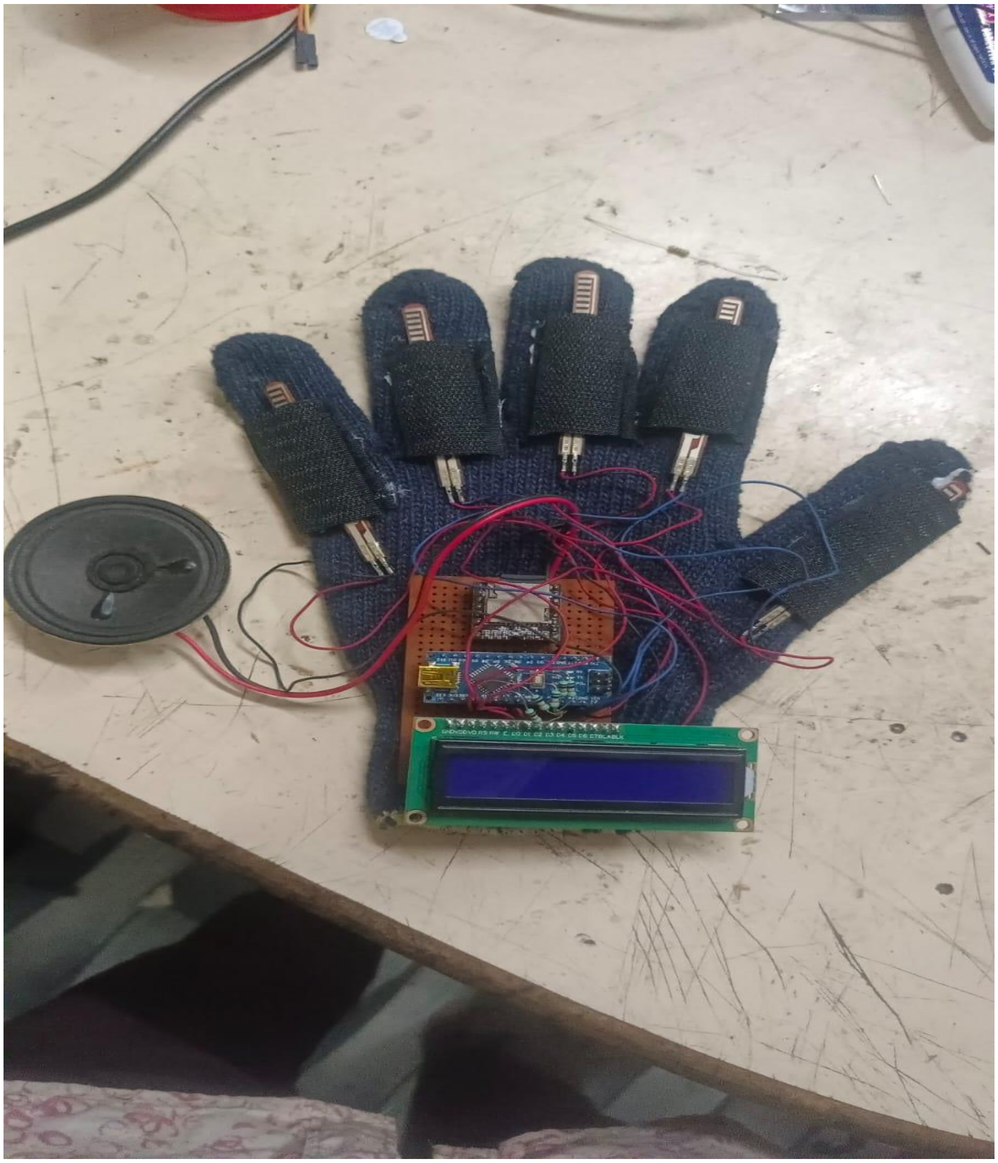

Fig. 5.3 Photocopy of Hardware

The photocopy of Hardware and Display output are shown in fig. 5.3 and fig. 5.4

1. "Hello, how are you?"
2. "What time is it?"

3. "Where is the nearest bus stop?"
4. "Can you show me the way to the library?"
5. "I would like to order a coffee, please."
6. "Do you know where I can find a pharmacy?"
7. "Excuse me, could you repeat that?"
8. "Thank you for your help."
9. "I need assistance with this."
10. "Goodbye, have a great day!"

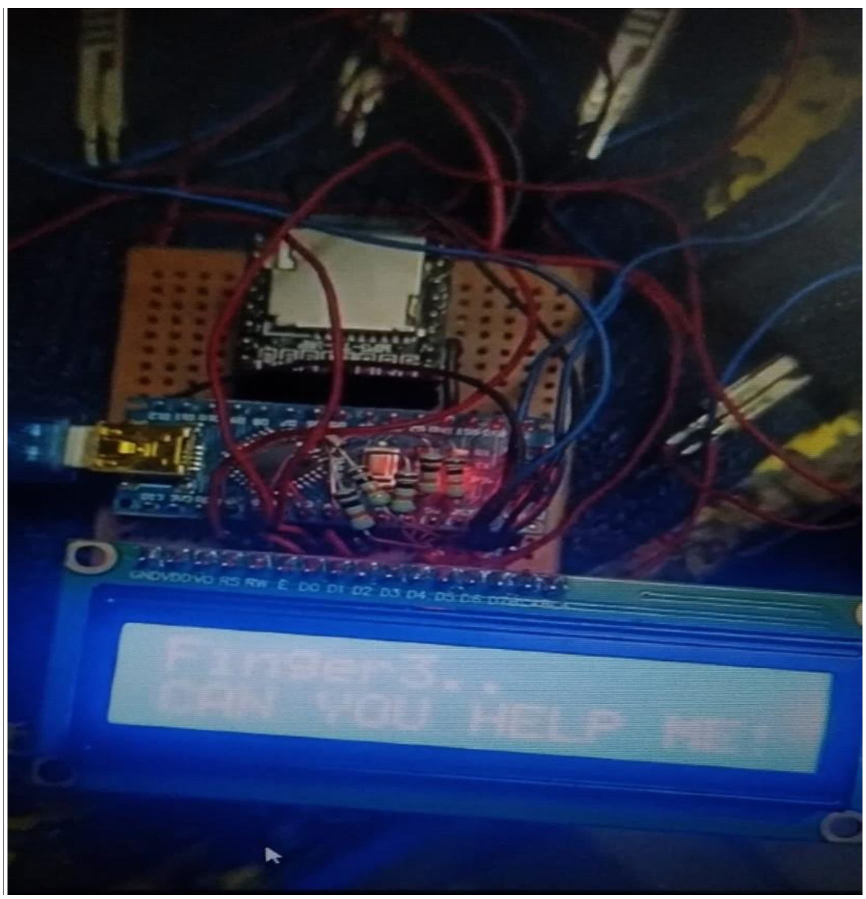

Fig 5.4 Display output

CHAPTER 6

CONCLUSION & FUTURE SCOPE

6.1 CONCLUSION

In this project, we constructed a Smart-Glove for supporting blind and deaf-blind people in communicating with normal people that are not familiar with braille. The Smart-Glove is able to connect to Android mobile and facilitate exchange of messages. Whereas the android application is able to send and receive text messages from and to the Smart-Glove and the Smart-Glove able to send and receive braille messages from and to the application. The Smart-Glove is light, cheap, easy to use and no risk. We believe that the project is an effective and very useful for deaf-blind people if they are taught braille where they can communicate with their families and people around them.As future scope of the project, the system may be extended to support other languages, and the system can use several way to communicate, it can use Wi-Fi connection, which enables a faster connection and better range from the base station or GSM module (Global System for Mobile communication) GSM is the most widespread and it's a cellular technology used for transmitting mobile data services, the most obvious advantage of it is widespread use throughout the world.

6.2 FUTURE SCOPE

Future advancements aim to enhance accuracy and efficiency in bridging communication gaps between sign language users and non-signers.Wearable devices equipped with advanced sensors are poised to revolutionize sign language communication, enabling seamless interaction across diverse environments.

Ethical considerations will play a pivotal role in ensuring the development of inclusive and culturally sensitive sign language technology.Collaborative research initiatives will drive innovation, addressing evolving needs and fostering greater accessibility for sign language users worldwide.

REFERENCES

[1] Gupta, Dhiraj-Design and development of a low cost Electronic Hand Glove for deaf and blind, 2nd International conference on computing for sustainable Global Development(INDIA.Com), pp501-505, 11-13 March 2015.

[2] Nishihara H. K. „Recognition of American Sign Language using Image Processing and Machine Learning‟, International Journal of Computer Science and Mobile Computing, Vol. 8,Issue 3,March 2019,pp:352-357.

[3] Padmanabhan, V. and Sornalatha, M., 2014. Hand gesture recognition and voice conversion and dumb people. International Journal of Scientific & Engineering Research, 5((5), p.427.

[4] Potdar, P.R. and Yadav, D.D., 2014. Innovative Approach for Gesture to Voice Connversion. International journal of innovative research and development, 3(6), pp.459-462.

[5] Rajaganapathy, S., Aravind, B., Keethana, B. and Sivagami, M.,2015. Conversation of Sign Language to Speech with Human Gestures. Procedia Computer Science, 50,pp.10-15.

[6] Shweta S.Shinde, Rajesh M. Autee and Vitthal K. Bhosale, „Sign Language Recognition for Deaf & Dumb‟, International Journal of Advanced Research in Computer Science and Software Engineering 3(9), September - 2013, pp. 103-106.

[7] Sidek, O Hadi, M.A., --Wireless gesture recognition system using MEMS accelerometer, International Symposium on Technology Management and Emerging Technologies (ISTMET), pp 444-447, 2014.