IoT-Assisted Smart Glove Communication Device for Hearing-Impaired Persons

A BS Final Year Project by

Kanwal Shahid
579/FET/BSEE/F19

Aneesa Mumtaz
562/FET/BSEE/F19

Aniqa Sadaf
593/FET/BSEE/F19

Supervised by
Dr. Engr Suheel Abdullah Malik
Co-supervised by
Engr. Muhammad Asad

Department of Electrical and Computer Engineering
Faculty of Engineering and Technology
International Islamic University, Islamabad

July, 2023

Certificate of Approval

It is certified that we have checked the project presented and demonstrated by **Kanwal Shahid 579-FET/BSEE/F19, Aneesa Mumtaz 562-FET/BSEE/F19, Aniqa Sadaf 593-FET/BSEE/F19** and approved it.

External Examinar
Dr. Irum Nosheen
Incharge of DECE

Internal Examiner
Engr. Rashid Farid Chisti
Lecturer

Supervisor
Dr. Suheel Abdullah
Associate Professor

Co-supervisor
Engr. Muhammad Asad
Lab Engineer

In the name of Allah (SWT), the most beneficent and the most merciful

A BS Final Year Project submitted to the

Department of Electrical and Computer Engineering

International Islamic University, Islamabad

In partial fulfillment of the requirements

For the award of the degree of

Bachelor of Science in Electrical Engineering

Declaration

We hereby declare that this work, neither as a whole nor as a part thereof has been copied out from any source. No portion of the work presented in this report has been submitted in support of any application for any other degree or qualification of this or any other university or institute of learning. We further declare that the referred text is properly cited in the references.

Kanwal Shahid
579-FET/BSEE/F19

Aneesa Mumtaz
562-FET/BSEE/F19

Aniqa Sadaf
593-FET/BSEE/F19

Acknowledgments

This BS thesis in Electrical Engineering has been conducted at Department of Electrical and Computer Engineering, Faculty of Engineering and Technology, International Islamic University, as part of the degree program. We would like to thank Dr. Engr. Suheel Abdullah for providing us an opportunity to work on this project, under his supervision and guidance throughout the project. We would also like to thank Engr. Muhammad Asad for his help, efforts and dedicated support throughout the project. Further we are particularly thankful to Almighty Allah and grateful to our parents, brothers and sisters who always supported and encouraged us during our project and studies at IIUI.

Kanwal Shahid

Aneesa Mumtaz

Aniqa Sadaf

Project Title:	IOT-Assisted Smart Glove Communication Device for Hearing-Impaired persons
Undertaken By:	Kanwal Shahid 579-FET/BSEE/F19
	Aneesa Mumtaz 562-FET/BSEE/F19
	Aniqa Sadaf 593-FET/BSEE/F19
Supervised By:	**Dr. Engr. Suheel Abdullah** Associate Professor
Co-Supervised By:	**Engr. Muhammad Asad** Lab Engineer
Date Started:	Aug, 2022
Date Completed:	July, 2023

Tools Used:

- Proteus
- Arduino
- Thingspeak
- PCB layout

Abstract

Communication between dumb-deaf people and normal people becomes very difficult in everyday life. The mute community and also persons with hearing/speech-impairments face many problems while communicating with normal people . In Pakistan ,such people can communicate with others only in one way i.e. sign language, but it is difficult for other people to understand their sign language. Therefore, there always exists communication barrier between normal people and the persons with hearing/speech-impairments . Most of people from this community are also mentally retarded and they are unable to take care of themselves at this age. Keeping such people in our mind and to reduce this communication barrier ,there is a need to find alternative solution to provide ease and facility for dumb/deaf and elderly people. Further, the trend of IOT technology and by combining advanced technologies and IoT capabilities, this project break down communication barriers and improve the quality of life for the mute, deaf, and elderly population. The integrated functionalities of gesture recognition, speech conversion, health monitoring, and GPS tracking provide a comprehensive solution to enhance communication, safety, and well-being for these individuals.

Contents

List of Figures

List of Tables

List of Abbreviations

SoC	System on Chip
FYDP	Final Year Design Project
OBE	Outcome Based Education
IoT	Internet of Things
PSL	Pakistan Sign Language
GPS	Global Positioning System
ASL	American Sign Language
TTS	Text-to-Speech
LCD	Liquid Crystal Display
SCL	Serial Clock Line
SDA	Serial Data Line
XDA	External Data Address
XCL	External Clock Line
INT	Interrupt
DC	Direct current
TX	Transmit
RX	Receive
GND	Ground
PWM	Pulse Width Modulation
UART	Universal Asynchronous Receiver-Transmitter
SPI	Serial Peripheral Interface
USB	Universal Serial Bus

LED	Light Emitting Diode
SRAM	Static Random Access Memory
EEPROM	Electrically Erasable Programmable Read-Only Memory
IDE	Integrated Development Environment
GPIO	General Purpose Input/Output
RST	Reset
DF	Decoding format player
PCB	Printed Circuit Board
HTTP	Hypertext Transfer Protocol
MQTT	Message Queuing Telemetry Transport
UDP	User Datagram Protocol
API	Application Programming Interface
ADC	Analog-to-Digital Converter
VDC	Voltage Direct Current
SD	Secure Digital
PPG	Photoplethysmography
REST API	Representational State Transfer Application Programming Interface
URL	Uniform Resource Locator
BPM	Beats Per Minute

Chapter 1

Introduction

1.1 Introduction

The project at hand focuses on the development of a comprehensive system that combines gesture recognition, health monitoring, and GPS tracking functionalities. This integrated system aims to address the needs of individuals with hearing impairments or those requiring continuous health monitoring and location tracking. By combining these modules into a unified prototype, we aim to provide a versatile solution that enhances communication, monitors vital signs, and enables real-time tracking. The smart glove incorporates various sensors and communication modules to accurately capture and interpret the hand gestures made by the user. The device is equipped with flex sensors that are strategically placed on each finger to detect the bending and movement of the user's hand. These flex sensors provide real-time data that is processed by an embedded microcontroller, such as an Arduino or Raspberry Pi, housed within the glove.

Once the gestures are identified, the device utilizes IoT capabilities to transmit the information to a central server or a mobile application. The IoT connectivity enables seamless communication between the smart glove and other devices, such as smartphones, tablets, or computers, allowing for immediate translation and interpretation of the sign language gestures.

Additionally, the IoT-Assisted Smart Glove Communication Device can be further enhanced by incorporating other features such as voice recognition, enabling two-way communication. This would allow spoken words to be converted into text or sign language gestures, expanding the device's capabilities and making it a versatile tool for both hearing-impaired individuals and the general population. By leveraging IoT technology and advanced sensor integration, this innovative smart glove communication device offers a promising solution to improve communication accessibility and foster inclusivity for hearing-impaired individuals. It has the potential to revolutionize the way sign language is interpreted and understood, ultimately breaking down communication barriers and promoting equal opportunities for all.

1.2 Motivation

Communication is a fundamental aspect of human interaction, and those with hearing impairment often face significant difficulties in this area. This project aims to improve their quality of life by providing a device that enables them to communicate effectively with others. Additionally, the use of IoT technology allows for the device to be seamlessly integrated into the users' daily lives, providing them with greater independence and autonomy. The project also serves as an opportunity to showcase the potential of technology to solve real-world problems and make a positive impact on society. Overall, the motivation behind this project is to empower individuals with hearing impairment by providing them with a solution that can enhance their communication abilities and improve their overall quality of life.

1.3 Project Overview

The aim of this project is to develop a hand device (glove) that can convert Pakistan Sign Language (PSL) based hand gestures into text and speech. The device will be equipped with flex sensors and accelerometers that will detect hand movements and translate them into text or speech using advanced algorithms. Additionally, the glove will be integrated with a health monitoring system that will consist of a heart rate sensor to monitor the wearer's vital signs. This will enable the device to provide not only communication assistance but also health monitoring capabilities. The device will be a two-way communication system, allowing the wearer to communicate with others while also receiving information through the IoT platform. The project will utilize the ThingSpeak IoT platform to display all data related to the device, including communication and health data. The project will involve the design and development of the glove, the integration of sensors and algorithms, and the creation of the IoT platform. User testing will be conducted to evaluate the device's effectiveness and to gather feedback for further improvements. The primary objective of this project is to create a wearable device that enhances the quality of life for people with hearing impairments while also incorporating health monitoring functionalities.

1.4 Problem Statement

Hearing-impaired people and elderly people who are unable to exchange their opinions with ordinary people due to their old age had to face many difficulties in communicating with others. They are unable to take care of themselves at this age. This problem has been solved before by using various techniques, such as image processing technology that converts hand gestures based on American Sign Language (ASL) into text through image recognition. In this project, we will design and develop a smart glove based on a flex sensor that will convert Pakistan Sign language (PSL) based hand gestures into text/speech and vice versa. Further, this proposed device will be integrated with an intelligent health monitoring system and a GPS tracking system based on IoT technology.

1.5 Project Objectives

- Design and develop Sensor based module that will convert PSL based hand gestures into text/speech.
- To integrate health monitoring system in smart gloves such as measuring temperature and heart rate.
- To design and integrate GPS based tracker in the glove.
- Communication using IOT technology.
- Design and development of prototype glove.

1.6 Brief Project Methodology

In order to achieve project objectives that are mentioned above and to develop a smart and intelligent glove, we will adopt and apply following methodology to complete our prototype in different stages:

Stage 1: We have to select a glove that is suitable for this project.

Stage 2: Interfacing the flex and accelerometer module with the glove (simulation and implementation).

Stage 3: We will write down code and Interface these sensors with arduino. (simulation and implementation).

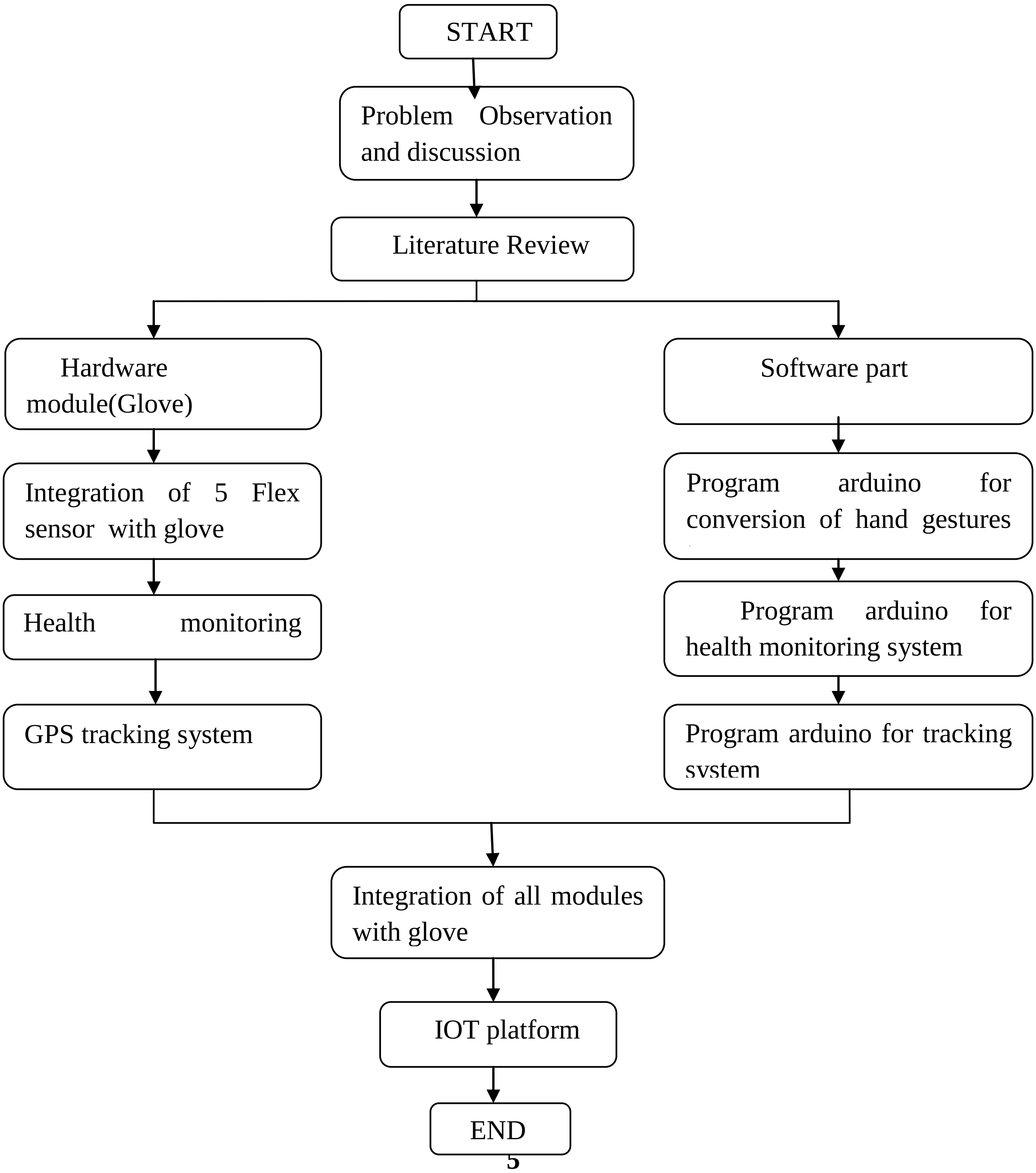
Stage 4: Design the heart rate and temperature sensor module.

Stage 5: Design a GPS tracker system using diff sensors.

Stage 6: Development of IoT interface for real time health monitoring and tracking.

Stage 7: Simulation, testing and results.

Stage 8: Development of prototype.



Chapter 2

Literature Review

2.1 Background of Project/Topic

The development of a hand device (glove) that can convert hand gestures into text and speech and is integrated with a health monitoring system is a response to the needs of individuals with hearing impairment. Communication is a fundamental aspect of human interaction, and those with hearing impairment often face significant difficulties in this area. The use of hand gestures is a common means of communication for individuals who are deaf or hard of hearing, but there is a need for a more efficient and effective method of translating these gestures into text or speech. The integration of health monitoring capabilities into the device is also important, as individuals with hearing impairment may also have other health concerns that need to be monitored regularly. The use of IoT technology in this project enables seamless communication and data transfer between the device and other devices, making it a versatile and accessible solution for individuals with hearing impairment. The background of this project is rooted in the desire to improve the quality of life for individuals with hearing impairment by providing them with a device that can enhance their communication abilities and monitor their health in a single wearable device.

The estimated population of Pakistan is approximately 180.7 million. As Pakistan has large population size so health is one of the most Important area of research. There are large number of hearing impaired People in ruler and urban areas of Pakistan.[1]

This chart shows total hearing-impaired person in Pakistan:

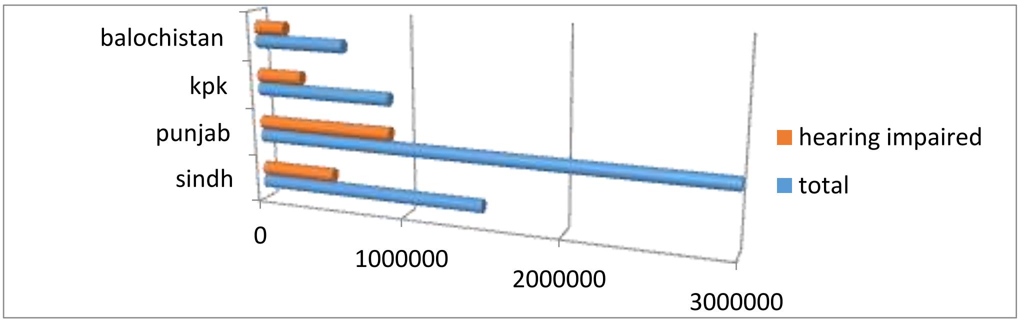


Figure 1 Population of Hearing-Impaired [1]

2.2 Related Work/Projects

Muhammad et al.

In [1], authors have explored the development of a communication system for individuals with hearing impairments. The system utilizes image processing techniques to interpret Pakistan Sign Language gestures, enabling two-way communication. It converts PSL gestures into text and speech, allowing hearing-impaired individuals to communicate effectively. In Image processing technique cameras are used to capture image of hand gesture and then it is converted into text. It is a two way communication system that converts sign language into text/speech and also speech into gestures using image processing technique. Additionally, the system can convert speech into PSL gestures, enabling hearing individuals to understand and respond. The research focuses on improving accessibility and inclusivity for the hearing-impaired community by providing an effective and efficient means of communication through technology.

Sabapathy et al.

In [2], author has introduces a location tracking system based on a wearable GPS antenna. The system utilizes global positioning system (GPS) technology to accurately track and monitor the location of individuals wearing the device. The research focuses on developing a compact and portable solution that can be easily integrated into wearable devices. The proposed system utilizes a custom-designed antenna to enhance the GPS signal reception and achieve accurate location tracking. The study highlights the potential applications of this technology in areas such as personal safety, outdoor sports, and navigation systems.

A. A. Roman Richard et al.

In [3], author has focused on developing a health monitoring system specifically designed for elderly and disabled individuals. The goal of the system is to offer continuous monitoring of vital signs like heart rate, blood pressure, and temperature, delivering real-time feedback and notifications. The research emphasizes the importance of remote monitoring and healthcare support for this vulnerable population. The proposed system utilizes sensors and wireless communication to collect and transmit data to a centralized monitoring unit. The research emphasizes the potential advantages of implementing such a system in enhancing the well-being and quality of life for older adults and individuals with disabilities.

N. & Nithiyanandham

In [4], author presented at the 2013 International Conference on Human Computer Interactions. The study introduces a communication system that utilizes hand gesture recognition to enable individuals who are unable to speak to effectively communicate. The system employs a sensor glove equipped with various sensors to capture hand gestures, and a recognition algorithm is developed to convert these gestures into meaningful communication. The proposed system utilizes a Bluetooth module to transmit the recognized gestures to a smartphone application, which then converts them into voice output. The study demonstrates the potential of hand gesture recognition technology in assisting individuals who are unable to speak in communicating effectively and overcoming communication barriers.

Syed Raquib Shareef et al.

In [5], author has focused on developing a hand gesture recognition system specifically designed to assist individuals who are deaf and mute in their communication. The system utilizes flex sensors and accelerometer sensors to capture hand movements and gestures made by the users. A Raspberry Pi is employed to process the data and convert the recognized gestures into text or speech output. The study emphasizes the potential of such a system in addressing the communication gap and enhancing the quality of life for individuals with impairments in hearing and speech. The study contributes to the field of assistive technology by providing a practical and efficient solution for facilitating communication for the deaf and mute population.

Hemant Pathare et al.

In [6], author introduced a novel communication system designed to facilitate communication for individuals who are deaf and mute. The system utilizes flex sensors to capture hand gestures made by the users, which are then processed by a microcontroller. A Bluetooth device is employed to transmit the recognized gestures to an Android application, which converts them into text or speech output. The research demonstrates the effectiveness of this gesture-based communication system in overcoming communication barriers for individuals with hearing and speech impairments. The research contributes to the advancement of innovative assistive technologies aimed at enhancing the quality of life and communication possibilities for the deaf and mute community

Bhosale VK et al.

In [7], author introduced a real-time communication system designed specifically for individuals who are both hearing impaired and mute. The system utilizes image processing techniques to facilitate two-way communication between the users. It converts sign language gestures made by the user into text or speech, allowing the hearing-impaired and mute person to effectively convey their messages to others. The research aims to provide an innovative solution that enhances the communication capabilities and social interactions of individuals with hearing and speech impairments.

V. R. Balaji et al.

In [8], author proposed a smart glove system designed to assist individuals who are unable to speak, particularly those with speech disabilities. The system utilizes IoT (Internet of Things) technology and smart gloves equipped with flex sensors to recognize hand gestures corresponding to different signs in sign language. The captured gestures are then transmitted via a Bluetooth module to a connected device, such as a smartphone or computer, which converts the received data into voice using a text-to-speech (TTS) module. The research aims to provide an innovative and accessible communication solution for individuals with speech impairments, enhancing their ability to interact and communicate effectively.

N. Karanth

In [9], author introduced a system that aids in interpreting sign language for effective communication between individuals with hearing impairments and those who do not understand sign language. The system utilizes a smart glove equipped with LED-LDR pair sensors to capture hand gestures and interpret them into corresponding sign language expressions. The captured gestures are then transmitted via a ZigBee module to a computer, where the received data is processed and interpreted. The objective of the research is to narrow the communication gap between individuals with hearing impairments and the general population through the development of an innovative technology-driven solution for sign language interpretation

Abed AA et al.

In [10], author has presented a system that utilizes a Raspberry Pi and Python programming language for hand gesture recognition. The system aims to enable natural and intuitive interaction with computers or electronic devices using hand gestures. By leveraging computer vision techniques and machine learning algorithms, the system can accurately detect and recognize various hand gestures performed by the user. The research highlights the potential of low-cost and easily accessible hardware like Raspberry Pi in developing gesture-based interaction systems.

Gulzar MM

In [11], author has presented at the International Conference on Intelligent System Engineering (ICISE) in 2016. The paper focuses on developing a system that utilizes image processing techniques to facilitate communication for deaf-mute individuals. The system converts sign language gestures captured by a camera into corresponding text or speech output. By employing image processing algorithms to analyze and interpret hand gestures, the system enables effective communication for deaf-mute individuals with others who may not be familiar with sign language. The research demonstrates the potential of image processing technology in bridging communication barriers for individuals with hearing and speech impairments.

Chakraborty P

In[12],author has designed and implement a system that can accurately identify and interpret isolated gestures in Indian Sign Language (ISL) in real-time. The system utilizes computer vision techniques to capture and process video data of hand gestures performed in ISL. Through the use of feature extraction and classification algorithms, the system is able to recognize and interpret the gestures in real-time. The research aims to contribute to the development of gesture recognition systems specifically designed for Indian sign language, which can be used in various applications to facilitate communication and interaction for individuals with hearing impairments.

Further other related work can be seen in these research paper [13-15]

2.3 Project Contribution

The "IoT-Assisted Smart Glove Communication Device for Hearing-Impaired Persons" project has the potential to make several contributions to society. Firstly, the device can significantly improve communication abilities for individuals with hearing impairment by converting PSL based hand gestures into text and speech using Flex sensor and accelerometer sensor.. Additionally, the integration of health monitoring capabilities into the device provides an added benefit and the wearer's vital signs can be monitored in real-time. This can assist individuals with hearing impairment in managing their health more effectively and can potentially improve their overall well-being. Also a GPS tracking system is also integrated with the glove through which location of a person/elder can be monitored easily.

The project also contributes to the field of IoT technology by showcasing the potential of integrating wearable devices with IoT platforms. This can lead to the development of more innovative solutions for individuals with different types of disabilities and health concerns.

2.4 Summary

The related projects summarized in the table aim to develop communication and health monitoring devices for individuals with disabilities. They employ various techniques such as image processing, GPS, gravity sensors, heart rate sensors, flex and accelerometer sensors, Bluetooth, and smart gloves to recognize and interpret gestures or sign language and convert them into text or speech. Some of these projects are two-way communication systems, while others focus on tracking and monitoring the health status of disabled patients.

In comparison, the proposed project is an IoT-Assisted Smart Glove Communication Device for Hearing-Impaired Persons that employs flex and accelerometer sensors to convert PSL-based hand gestures into text and speech and integrate it with a health monitoring system consisting of heart rate and temperature sensors. The device uses the IoT platform to display all data and is a two-way communication system that allows for seamless communication between hearing-impaired persons and non-hearing-impaired individuals. The primary goal of the proposed project is to enhance the well-being of individuals with hearing impairments by developing a device that facilitates effective communication and enables them to monitor their health status, ultimately improving their overall quality of life.

Chapter 3

System Design and Implementation

3.1 System Design

3.1.1 PSL(Pakistan Sign Language)

PSL stands for Pakistan Sign Language, which is a visual language used by the deaf community in Pakistan to communicate. Being a primary mode of communication within the deaf community in Pakistan, the sign language utilized exhibits its own grammar and syntax, resembling a natural language.Like other sign languages, PSL uses a combination of hand gestures, facial expressions, and body language to convey meaning.

In Pakistan, There are a lot of people that are unable to talk and listen. They uses sign language to communicate with others but sometimes it is very difficult to understand there sign language for ordinary people and they also feel difficulty while talking to common individual. This sign language is called Pakistan sign language.PSL is used as a standard language in institutions that are made for dumb-deaf people.

PSL is also used by many elderly/disabled people who are unable to talk due to their old age or some other reasons.PSL includes all Alphabets, words and also sentences. We aim to convert hand gestures that are based on PSL language into text/speech. Through this elderly/disable/dumb-deaf people understand and easily communicate with other people.

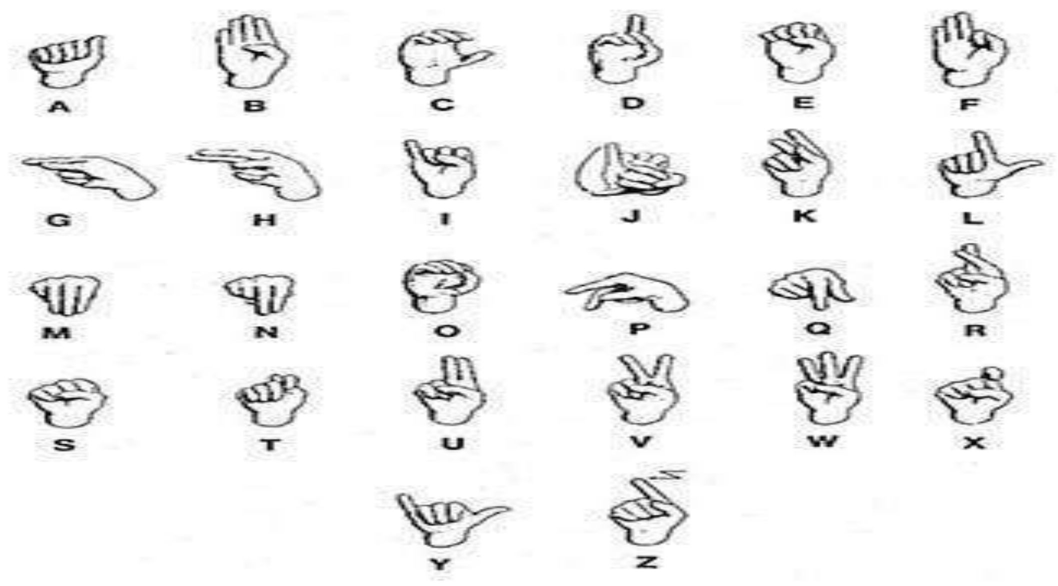
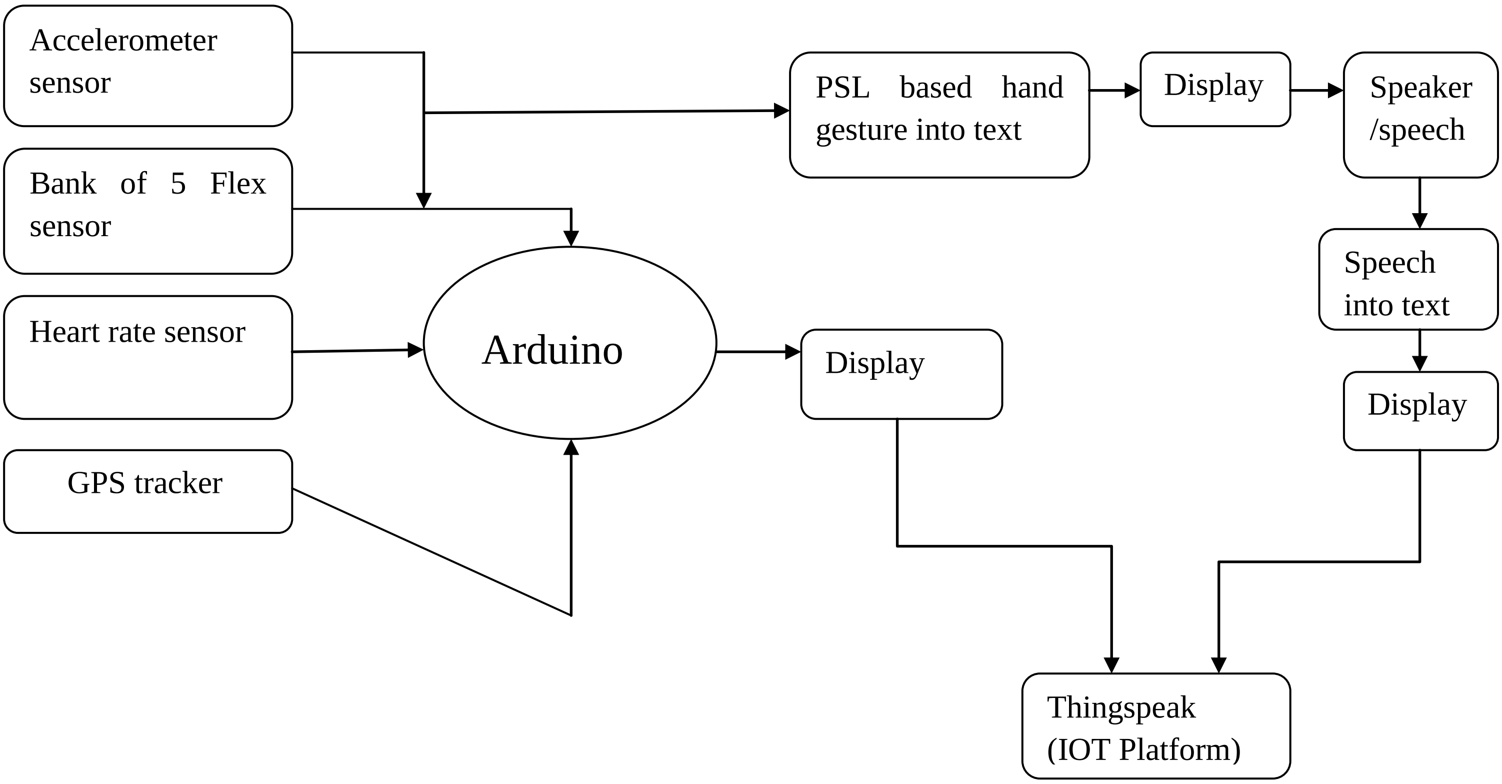


Figure 2 PSL [10]

3.2.1 Block Diagram



This block diagram provides a visual representation of the main components of project and their interconnections. It starts with the user's hand, which is equipped with sensors. The sensor data is processed by the Arduino Mega 2560 board, which displays on LCD. The input data from flex and accelerometer sensor goes to arduino and it process the data and display it on LCD. Then it is converted into audio/speech. Data from temperature sensor and heart rate sensor also go to arduino and displayed on LCD. Similarly, In GPS tracking system the location(longitude and latitude) shows on LCD after process. Finally, the data is transmitted to the Thingspeak IoT platform for monitoring and analysis.

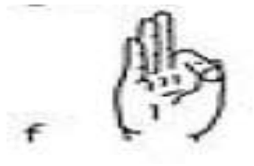
3.2.2 Implemented Gestures

In this project, we create a system that allows individuals to communicate using Pakistan Sign Language (PSL) simultaneously. To achieve this, we assigned each of letter with a specific sentence or phrase that is commonly used in daily communication. For instance, the letter "A" is assigned with the sentence “Asalamulikum” and so on.

Each letter is also associated with a specific PSL based hand gesture. To capture the hand gestures made by the user, a glove equipped with flex sensors and accelerometers is employed in the system. These sensors detect the bending and tilting of fingers of the user.

Table 1mplemented gestures[10]

Letters	Gesture	Sentence
A		AOA
B		Give me medicine
C		Want to eat
D		Watching you

E		please wait
F		I am Fine
G		feeling good
H		Open the door
I		Want Water
K		Who are you?
L		I am sorry
Q		Want to sleep

W		See you later
Y		Your welcome

3.2 Main Hardware Components

The hardware development setup for the proposed project "IoT-Assisted Smart Glove Communication Device for Hearing-Impaired Persons" includes the following components:

- Flex sensor
- Accelerometer sensor
- Heart rate sensor (MAX 30102)
- GPS tracker/module
- Arduino Mega 2560
- Buck converter LTC3442
- Speaker
- DF Player mini mp3 player
- Node MCU ESP32
- LCD display (16x4)

The flex sensor and accelerometer sensor are used to detect the hand gestures of the user, which are then converted into text and speech using the Arduino Mega 2560. The heart rate sensor is used to monitor the health of the user. The GPS tracker/module is used to track the location of the user.

The Node MCU ESP32 is used as an IoT platform to collect all the data from the sensors and display it on the LCD display. The data is also sent to the cloud using Wi-Fi connectivity for remote monitoring and analysis. The LCD display is used to show the real-time data and status of the device.Overall, this hardware development setup provides a comprehensive solution for communication and health monitoring of hearing-impaired persons.

3.3 Flex sensor

A flex sensor is a type of electronic sensor that changes its resistance when it is bent or flexed. It consists of a flexible strip or ribbon made of conductive material, such as carbon or metal. The resistance of the flex sensor increases or decreases based on the degree of bending, allowing it to detect and measure the amount of flexion. Flex sensors are commonly used in various applications, including robotics, wearable devices, and motion sensing systems.

3.3.1 Specification

A flex sensor is a type of sensor that measures the amount of bending or deflection that occurs when an external force is applied. Flex sensors are flexible electronic devices that change resistance as they bend. They typically have a specified resistance range, varying from a few ohms to several kilohms. The flexibility of flex sensors allows them to conform to the shape of objects or surfaces, accurately measuring the degree of bending. They come in various sizes and mounting options, enabling easy integration into different applications. Flex sensors exhibit different levels of sensitivity, detecting even slight changes in resistance with minimal bending. They have defined electrical ratings, including maximum voltage and current limits, ensuring safe operation. Temperature range is an important consideration, as flex sensors are designed to function within specific temperature limits. The output interface can be analog or digital, providing either variable resistance output or discrete values indicating bending positions. Overall, flex sensors offer versatility and reliability in measuring bending or flexing angles in a wide range of applications.

3.3.2 Pin Configuration

Table 2 PIN config

Pin Number	Description
P1	Usually connected to positive of power source.
P2	Usually connected to ground.

3.3.3 Diagram

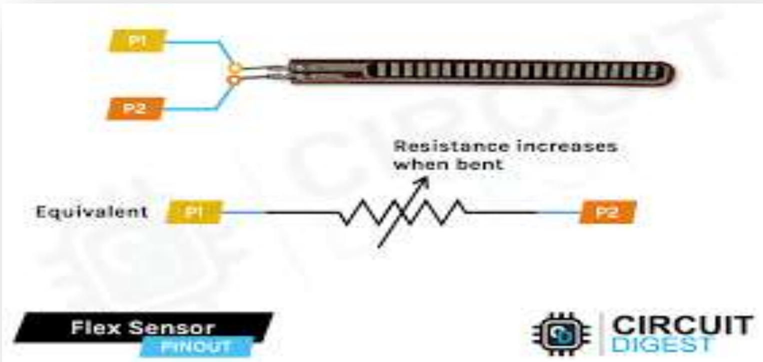


Figure 3 Flex sensor

Flex sensors are commonly used in various applications, including robotics, wearables, and medical devices. In the proposed project, flex sensors are used to recognize hand gestures made by the user wearing the smart glove. As the user bends their fingers or moves their hand in a particular way, the flex sensors detect the bending and transmit the data to the microcontroller(Arduino). The microcontroller then processes the data and converts it into text or speech, depending on the user's needs. Overall, flex sensors play a critical role in the functionality of the IoT-Assisted Smart Glove Communication Device for Hearing-Impaired Persons.

3.4 Accelerometer sensor(MPU 6050)

The MPU 6050 accelerometer sensor is a commonly used electronic sensor that measures acceleration and tilt in three dimensions. The accelerometer measures linear acceleration, while the gyroscope measures angular velocity. The MPU 6050 sensor is widely used in applications such as motion tracking, gesture recognition, and orientation sensing. It provides accurate and reliable data for detecting and monitoring movement and orientation in various electronic devices and systems.

3.4.1 Specification

An accelerometer sensor is a type of sensor that measures acceleration forces. It can detect changes in movement, velocity, and direction of movement in three-dimensional space. The MPU6050 accelerometer sensor is a widely used and popular component in motion-sensing applications. It offers several specifications that contribute to its reliable performance. The sensing range of the MPU6050 accelerometer is typically ±2g, ±4g, ±8g, or ±16g, providing flexibility for various acceleration measurement needs. It has a sensitivity of 16384 LSB/g, allowing for precise detection of even small changes in acceleration. The sensor is a 3-axis accelerometer, capable of measuring acceleration along the X, Y, and Z axes simultaneously. The MPU6050 accelerometer has a resolution of 16 bits, ensuring accurate measurement of acceleration values. It features a digital output interface using the I2C communication protocol, making it compatible with a wide range of microcontrollers and devices. With a compact size and low power consumption, the MPU6050 accelerometer sensor is suitable for applications requiring motion detection and tracking.

3.4.2 Diagarm

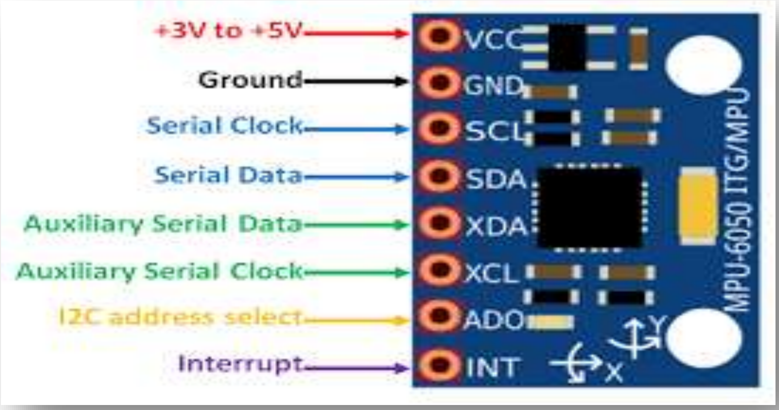


Figure 4 Accelerometer

3.4.3 Pin Configuration

Table 3 Configuration

Pin No.	Pin Name	Pin Description
1.	Vcc	Provides power for the module, can be +3V to +5V. Typically +5V is used
2.	Ground	Connected to Ground of system
3.	Serial Clock(SCL)	Used for providing clock pulse for I2C Communication
4.	Serial Data(SDA)	Used for transferring Data through I2C communication
5.	Auxiliary Serial Data (XDA)	Can be used to interface other I2C modules with MPU6050. It is optional
6.	Auxiliary Serial Clock (XCL)	Can be used to interface other I2C modules with MPU6050. It is optional
7.	AD0	If more than one MPU6050 is used a single MCU, then this pin can be used to vary the address
8.	Interrupt (INT)	Interrupt pin to indicate that data is available for MCU to read.

3.5 Heart rate sensor(MAX 30102)

3.5.1 Specification

The MAX30102 heart rate sensor is a highly integrated and widely used sensor for monitoring heart rate and pulse oximetry. It offers various specifications that contribute to its accurate and reliable performance. The sensor utilizes advanced algorithms and optical sensing techniques to measure heart rate and blood oxygen saturation levels (SpO2). The MAX30102 has a wide dynamic range, enabling it to capture a broad range of heart rates accurately. It operates with low power consumption, making it suitable for battery-powered devices. The sensor is equipped with ambient light cancellation technology to mitigate the effects of ambient light interference. It communicates with microcontrollers or other devices via the I2C interface. The MAX30102 heart rate sensor is compact and easy to integrate into wearable devices, healthcare applications, and fitness trackers, providing valuable insights into heart rate monitoring and oxygen saturation levels.

3.5.2 Pin configuration

This sensor has 8 pins. 4 pins are more useful:

Table 4 PIN numbers

Pin Number	Pin Name	Description
1.	VCC	Module power supply – 3 to 5
2.	VGND	Ground
3.	SCL	12C clock bus
4.	SDA	12C data bus

3.5.3 Diagram

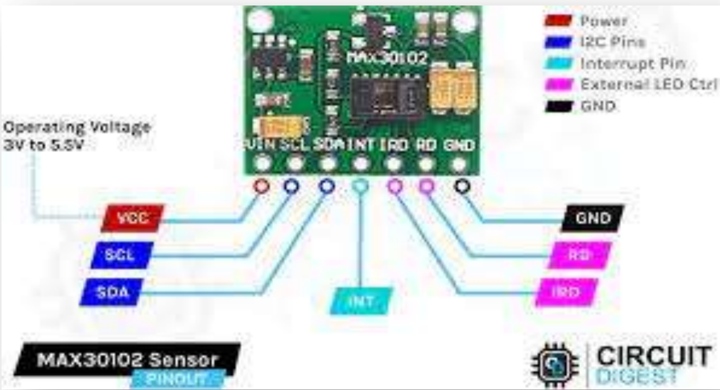


Figure 5 Heart rate sensor

3.6 Buck Converter

A buck converter is a DC-DC converter that effectively reduces a higher input voltage to a lower output voltage. It employs a switching transistor and an inductor to facilitate efficient energy transfer from the input to the output. The transistor switches on and off rapidly, creating a pulsating current through the inductor. This causes energy to be stored in the inductor during the on state and released to the load during the off state, resulting in a lower output voltage. Buck converters are commonly used in electronic devices to provide stable and regulated power at lower voltage levels.

3.6.1 Specifications

Table 5 Specifications

Input Voltage	12V
Output Voltage	1.8V
Maximum Power	120W
Switching Frequency	500kHz
Inductor current ripple	30%
Output voltage ripple	10mV

3.6.2 Diagram

Figure 6 Buck Converter

3.7 GPS Tracker

3.7.1 Specifications

A GPS (Global Positioning System) module is a device that receives signals from GPS satellites and uses those signals to determine the device's precise location on Earth. The Ublox Neo 6M GPS tracker is a widely used and reliable module for GPS positioning and navigation applications. It offers several specifications that contribute to its accurate and efficient performance.

Table 6 Specifications

Type	GPS
Supply	2.7V-3.6V
Operating Current	45mA
Operating Temperature	-40 degree celcius to 85 degree celcius
Horizontal Position Accuracy	2.5m
Communication Protocol	NMEA,UBX Binary, RTCM
Features	RTC Crystal and External Interrupt/Wake up
Interface	UART,SPI,USB and DDC

3.7.2 Pin Configuration

Table 7 GPS Pin config

Pin Number	**Pin Name**	**Description**
1.	GND	This is the ground pin that will be connected with the ground of the Arduino UNO
2.	TX	This is the transmission pin used for serial communication.
3.	RX	This is the receiver pin used for serial communication.
4.	VCC	This is the VCC pin used to power up the GPS module. Connect it with the 5V of the Arduino UNO board.

3.7.3 Diagram

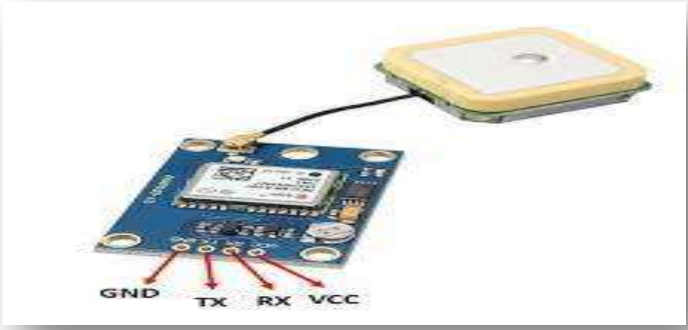


Figure 7 GPS tracker

If a GPS module is included, it can be used to track the user's location in real-time and send that information like longitude and latitude to the IoT platform, where it can be displayed and monitored. This feature can be particularly useful in emergency situations, where the user's location needs to be quickly determined and communicated to emergency services. A GPS module can also be used to provide additional functionality to the device, such as location-based reminders and alerts, or integration with navigation systems for the visually impaired.

3.8 Arduino Mega 2560

3.8.1 Components in Arduino Mega

The Arduino Mega 2560 is a microcontroller board that provides a wide range of components and features for building electronic projects. Some of the key components found on the Arduino Mega 2560 board include:

Microcontroller:

The Arduino Mega 2560 is powered by the ATmega2560 microcontroller, which acts as the brain of the board.

Digital I/O Pins:

The board features 54 digital input/output (I/O) pins, with 15 of them capable of providing PWM (Pulse Width Modulation) output.

Analog Inputs:

There are 16 analog input pins on the board, allowing for the measurement of analog voltages or connection to analog sensors.

UART (Universal Asynchronous Receiver-Transmitter):

The Arduino Mega 2560 has four hardware UART serial communication ports, which enable communication with other devices using serial communication protocols such as RS-232 or MIDI.

SPI (Serial Peripheral Interface):

The board includes one hardware SPI interface, which facilitates communication with SPI-compatible devices like SD cards, EEPROMs, and other microcontrollers.

I2C (Inter-Integrated Circuit):

The Arduino Mega 2560 has two hardware I2C interfaces, enabling communication with I2C devices such as sensors, real-time clocks, and EEPROMs.

USB Interface:

The board has a built-in USB interface that allows it to be connected to a computer for programming and communication.

Power Supply:

The Arduino Mega 2560 can receive power either through the USB connection or from an external power supply. It features a voltage regulator that ensures stable power to the components.

Reset Button:

The board includes a reset button that can be used to restart the microcontroller or initiate the bootloader.

LED Indicators:

The Arduino Mega 2560 has several built-in LEDs that can be used for visual feedback or debugging purposes.

3.8.2 Memory of Arduino Mega

The Arduino Mega 2560 microcontroller board has the following memory specifications:

Flash Memory:

The Arduino Mega 2560 has 256 KB of flash memory. This is where the Arduino sketch (program) is stored. The flash memory can hold the program code and any data required by the program.

SRAM (Static Random Access Memory):

The Arduino Mega 2560 has 8 KB of SRAM, which is used for storing variables, arrays, and other data that is actively used during program execution. SRAM is faster to read from and write to compared to flash memory.

EEPROM (Electrically Erasable Programmable Read-Only Memory):

The Arduino Mega 2560 has 4 KB of EEPROM, which is non-volatile memory used for storing data that needs to be retained even when the power is turned off. EEPROM is commonly used for storing settings, calibration data, or other persistent data.

3.8.3 Software Used

The Arduino Mega 2560 can be programmed using the Arduino Software (IDE), which is a free, open-source integrated development environment specifically designed for Arduino boards. The Arduino IDE provides a user-friendly interface for writing, compiling, and uploading code to the Arduino Mega 2560. The Arduino IDE is available for Windows, Mac, and Linux operating systems, making it accessible to a wide range of users. The Arduino programming language, which is a variation of C/C++, is fully compatible with the system. It offers dedicated libraries and functions designed specifically for Arduino boards, enhancing their functionality and ease of use. In addition to the Arduino IDE, there are also alternative software options available for programming the Arduino Mega 2560. These include platform-specific tools such as Atmel Studio, Eclipse with AVR plugin, or Visual Studio with Visual Micro plugin, which provide more advanced features and customization options.

3.8.4 Specifications

Table 8 Specifications

Serial No.	Specifications	Details
1.	MCU	ATmega 2560
2.	Architecture	AVR
3.	Operating Voltage	5V
4.	Input Voltage	6V-20V(limit) 7V-12V(recommended)
5.	Clock Speed	16MHz
6.	Flash Memory	256KB(8KB of this is used by bootloader)
7.	SRAM	8KB
8.	EEPROM	4KB
9.	Digital IO Pins	54(of which 15 can produce PWM)
10.	Analog Input Pins	16

3.8.5 Diagram

Figure 8 Arduino mega2560

3.9 Node MCU ESP32

3.9.1 Specification

The NodeMCU ESP32 is a powerful development board with the following specifications:

Microcontroller: ESP32 dual-core processor running at 160 MHz

Wi-Fi: Built-in 802.11 b/g/n Wi-Fi for wireless connectivity

Bluetooth: Integrated Bluetooth 4.2 for wireless communication

Memory: 520KB SRAM and 4MB flash memory for program storage

GPIO Pins: Equipped with multiple General Purpose Input/Output (GPIO) pins for connecting to external devices

Analog Inputs: Supports up to 18 analog inputs for reading analog sensor values

Operating Voltage: Operates at 3.3V, compatible with a wide range of sensors and modules

Programming: Can be programmed using the Arduino IDE or other programming environments

Interfaces: Includes UART, SPI, I2C, and other communication interfaces for device connectivity

Power Supply: Can be powered via USB or external power source

Dimensions: Compact and lightweight, making it suitable for various IoT applications

Programming: The NodeMCU ESP32 can be programmed using various programming languages and frameworks.

Arduino IDE: The Arduino IDE is a popular choice for programming the NodeMCU ESP32. You can write code in C/C++ using the Arduino programming language and utilize the extensive Arduino libraries and functions.

3.9.2 Pin Configuration

Table 9 PIN config

Pin Category	**Name**	**Description**
Power	Micro-USB, 3.3V, GND, Vin	Micro-USB: NodeMCU can be powered through the USB port 3.3V: Regulated 3.3V can be supplied to this pin to power the board GND: Ground pins Vin: External Power Supply
Control Pins	EN,RST	The pin and the button resets the microcontroller
Analog Pins	A0	Used to measure analog voltage in the range of 0-3.3V
GPIO Pins	GPIO1 TO GPIO16	NodeMCU has 16 general purpose input-output pins on its board
SPI Pins	SD1, CMD, SD0, CLK	NodeMCU has four pins available for SPI communication.
UART Pins	TXD0, RXD0, TXD2, RXD2	NodeMCU has two UART interfaces, UART0 (RXD0 & TXD0) and UART1 (RXD1 & TXD1). UART1 is used to upload the firmware/program.
12C Pins		NodeMCU has I2C functionality support but due to the internal functionality of these pins, you have to find which pin is I2C.

3.9.3 Applications

- Prototyping of IoT devices
- Low power battery operated applications
- Network projects
- Projects requiring multiple I/O interfaces with Wi-Fi and Bluetooth functionalities

3.9.4 Diagram

Figure 9 Node MCU

3.10 LCD(Liquid Crystal Display)

3.10.1 Features

A 16x4 LCD (Liquid Crystal Display) module is a popular alphanumeric display that consists of 16 columns and 4 rows of characters. Here are some common features of a 16x4 LCD:

Display Size:

The LCD module has a display size of 16 characters per line and 4 lines, allowing for the display of a maximum of 64 characters.

Alphanumeric Characters:

The LCD can display alphanumeric characters, including letters, numbers, symbols, and special characters.

HD44780 Controller:

The 16x4 LCD typically uses the HD44780 controller chip, which provides the necessary circuitry and commands for interfacing with the display.

Contrast Adjustment:

The LCD module allows for adjusting the contrast level of the display to optimize the visibility of the characters.

Low Power Consumption:

LCD technology is known for its low power consumption, making it suitable for battery-powered applications.

Serial or Parallel Interface:

The LCD module can be connected to a microcontroller or other devices using either a serial interface (e.g., I2C or SPI) or a parallel interface, depending on the specific model.

Easy Integration:

The LCD module is designed to be easily integrated into various projects, with standard pinouts and mounting holes for convenient installation.

3.10.2 Pin Description

Table 10 PIN config

PIN NO.	SYMBOL	FUNCTION
1.	Vss	Ground
2.	VDD	+3V to +5V
3.	Vo	Contrast adjustment
4.	RS	H/L register select signal
5.	R/W	H/L read write signal
6.	E	H ⟶ L enable signal
7.	DB0	H/L data bus line
8.	DB1	H/L data bus line
9.	DB2	H/L data bus line
10.	DB3	H/L data bus line
11.	DB4	H/L data bus line
12.	DB5	H/L data bus line
13.	DB6	H/L data bus line
14.	DB7	H/L data bus line
15.	A/VEE	+ 4.2 V for LED (RA = 0 Ω)/negative voltage output
16.	K	Power supply for B/L (0 V)

3.10.3 Diagram

Figure 10 LCD

3.11 DF player mini mp3

Designed specifically for embedded systems, the DFPlayer Mini MP3 module is a compact and versatile audio player module with a wide range of functionalities. It can play MP3 and WAV files from a microSD card. The module has a built-in amplifier and supports both UART serial communication and a simple button interface for controlling playback. It is commonly used in applications such as music players, sound effects, and voice prompts in projects requiring audio playback functionality. With its small size and ease of use, the DFPlayer Mini MP3 module is a popular choice for adding audio capabilities to electronic projects.

3.11.1 Specification

- Sampling rates (kHz): 8/11.025/12/16/22.05/24/32/44.1/48
- 24 -bit DAC output, support for dynamic range 90dB , SNR support 85dB
- Fully supports FAT16 , FAT32 file system, maximum support 32G of the TF card, support 32G of U disk, 64M bytes NORFLASH
- A variety of control modes, I/O control mode, serial mode, AD button control mode

- advertising sound waiting function, the music can be suspended. when advertising is over in the music continue to play
- audio data sorted by folder, supports up to 100 folders, every folder can hold up to 255 songs
- 30 level adjustable volume, 6 -level EQ adjustable

3.11.2 Diagram

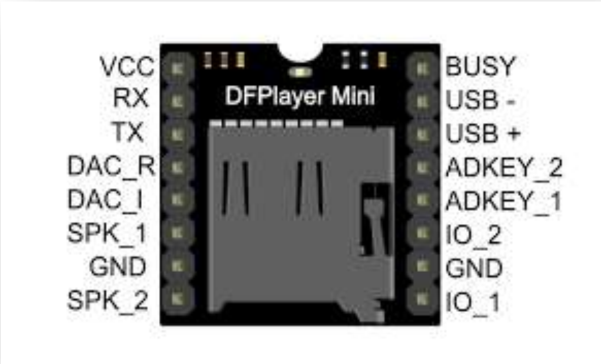


Figure 11 DF player

3.12 HC-05 Bluetooth module:

The HC-05 is a popular Bluetooth module commonly used for wireless communication between electronic devices. It is a cost-effective and easy-to-use module that allows for wireless serial communication over Bluetooth. The HC-05 module supports the Bluetooth 2.0 standard and can be configured as either a master or a slave device.

3.12.1 Specification:

The specifications of a HC-05 Bluetooth module are:

Bluetooth Version: Bluetooth 2.0+EDR (Enhanced Data Rate)

Frequency Range: 2.4 GHz ISM band

Communication Interface: UART (Universal Asynchronous Receiver-Transmitter)

Supported Profiles: Serial Port Profile (SPP)

Operating Voltage: Typically 3.3V (some modules support 5V)

Range: Up to 10 meters (depending on the environment)

Data Transfer Rate: Maximum 2.1 Mbps

Operating Temperature: -20°C to +75°C

Power Consumption: Low power consumption for extended battery life

Dimensions: Compact size, usually in a small module form factor

Serial Communication: Supports standard serial communication protocols (e.g., UART) for easy integration with microcontrollers and other devices**.**

3.12.2 Pin Configuration:

Table 11 Pins description

Pin No.	Pin Name	Description
1.	Enable 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.	TX-Transmission	Transmits Serial Data. Everything received via Bluetooth will be given out by this pin as serial data.
5.	RX-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 if 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.

3.12.3 Diagram

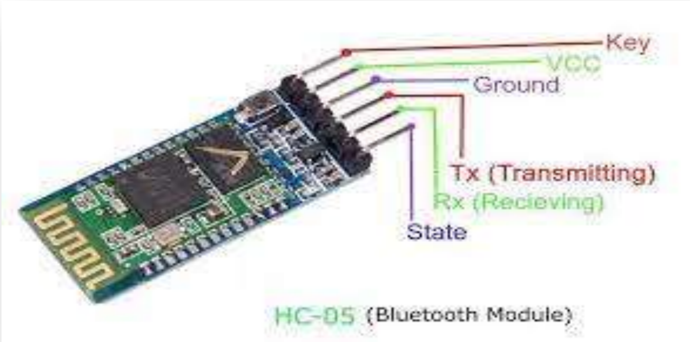


Figure 12 Bluetooth Module

3.13 Software Tools

3.13.1 Proteus

Proteus software is a software that is used for schematic capture, simulation, and PCB layout design. It allows engineers and designers to design and simulate electronic circuits and systems in a virtual environment before actually building the hardware. Proteus software provides a comprehensive library of electronic components, including microcontrollers, sensors, displays, and other commonly used components. It also supports a wide range of microcontroller families, making it a popular choice for embedded system design. Proteus software is widely used in the electronics industry for rapid prototyping and testing of electronic circuits and systems. Proteus software is an excellent choice for simulation of electronic circuits and systems. It provides a virtual environment to design, simulate, and test your electronic hardware before actual implementation. In this project, Proteus software is used to design and simulate the circuitry of IoT-Assisted Smart Glove Communication Device for Hearing-Impaired Persons. It to verify the functionality of circuitry, detect errors or inconsistencies, and make necessary modifications before building the actual hardware.

3.13.2 PCB layout

Proteus is a software suite for electronic circuit design, simulation, and testing. It includes a PCB layout tool that enables the user to design a physical board layout and routing of electrical components. The software allows the user to create a schematic diagram of the circuit, and then transfer it to the PCB layout tool for designing the board.

With the Proteus PCB layout tool, the user can place components on the board, and then connect them by drawing tracks or routes using the routing tools. The software also provides a set of design rules that ensure the correct placement of components and tracks, and to avoid issues such as short circuits and ground loops. After the PCB design is complete, the software provides a 3D view of the board layout to visualize the physical appearance and to verify that the components are correctly placed and connected. The user can then transfer the PCB layout to a manufacturing facility to produce the actual circuit board.

3.13.3 Arduino

The Arduino software, an open-source Integrated Development Environment (IDE), serves as a platform for writing, compiling, and uploading code to Arduino boards. It is a user-friendly platform that supports the C++ programming language and simplifies the process of programming and debugging for hardware development.

In this project, Arduino IDE is used to write code for the microcontroller on IoT-Assisted Smart Glove Communication Device for Hearing-Impaired Persons. It also comes with a library of pre-written functions that can be used to interact with various hardware components such as sensors, actuators, and displays. After completing the code writing process, we can proceed to upload it to the Arduino board and subsequently test its functionality with the connected hardware. The Arduino software is widely used by the maker community and is an excellent tool for prototyping and developing small-scale projects.

3.13.4 THINGSPEAK (IOT platform)

ThingSpeak is an IoT (Internet of Things) platform that enables users to gather, store, and analyze data obtained from IoT devices or sensors. It provides an open-source API that enables developers to build applications or integrations with other systems. ThingSpeak is cloud-based, which means that data can be accessed from anywhere, and it supports various communication protocols such as HTTP, MQTT, and UDP. The platform also offers data visualization tools, including graphs and charts, to help users visualize and analyze their data. With ThingSpeak, users can create applications for a wide range of IoT use cases, such as environmental monitoring, smart homes, and industrial automation.

In this project, ThingSpeak is used as an IoT platform for displaying all the data collected from the sensors. The data collected from the flex sensor, accelerometer sensor, heart rate sensor, and temperature sensor are transmitted wirelessly to the NodeMCU ESP32 using the Arduino Mega 2560. By establishing a Wi-Fi connection, the NodeMCU ESP32 is linked to the internet and employs the MQTT protocol to transmit data to the ThingSpeak cloud server. Subsequently, the real-time data is displayed on the ThingSpeak platform, enabling users to monitor the health status and communication data of individuals with hearing impairments.

Chapter 4

Simulation and Testing of Proposed System

4.1 Introduction

The chapter "Electronic Circuit Design" encompasses the consolidation of various modules within the project, each serving a specific function. It encompasses essential components such as converting hand gestures into text and speech, converting speech into text, health monitoring, and GPS tracking. These modules are crucial building blocks that contribute to the overall functionality of the system. One of the modules focuses on translating hand gestures into text and speech. The system interprets the gestures made by users and converts them into meaningful written or spoken words. This facilitates effective communication between individuals using sign language and those who may not understand it. Another module involves converting spoken language into written text. Through cutting-edge speech recognition algorithms and natural language processing, the system accurately transcribes spoken words or phrases into written form. This capability enhances accessibility and enables efficient communication through written text. The health monitoring module is designed to monitor vital health parameters in real-time. By utilizing sensors such as heart rate, the system tracks and analyzes key physiological indicators. This functionality provides valuable insights into an individual's health status and enables the detection of any abnormalities or changes that may require attention. The chapter also includes the simulation and testing of these different modules to ensure their proper functioning. It examines each module's individual components and their interactions within the overall system. By exploring the separate parts of the project and their respective workings, this chapter offers a comprehensive understanding of the integration process and the successful collaboration of various modules.

4.2 Conversion of hand gesture

For conversion of PSL based hand gestures into text and speech, flex sensor and accelerometer is used. Five flex sensors are attached with the five fingers of the glove. First we check the change in resistance of flex sensor while bending. Flex sensor operate on 5 Volts and the ADC value is from 0 to 1023. With the change in resistance, the output voltage also changes so we apply voltage divider rule to check change in output voltage when resistance of flex sensor changes.

PSL based hand gestures into text and speech

Flex sensor and accelerometer sensor used for this purpose. Five flex sensors are attached with Five fingers . one accelerometer sensor is connected with flex sensors. These sensors are integrated with arduino mega 2560. When fingers will bend, resistance of flex sensor changes. Accelerometer sensor is used to check dimension of gesture.

Voltage Divider Circuit

A potential divider circuit, commonly referred to as a voltage divider, is an electronic circuit employed to divide a voltage into smaller fractions. It comprises two or more resistors connected in series across a voltage source, with the output voltage derived from the junction between the resistors. The primary objective of a voltage divider is to achieve a desired voltage level from a higher voltage source. The basic circuit diagram of a voltage divider is as follows:

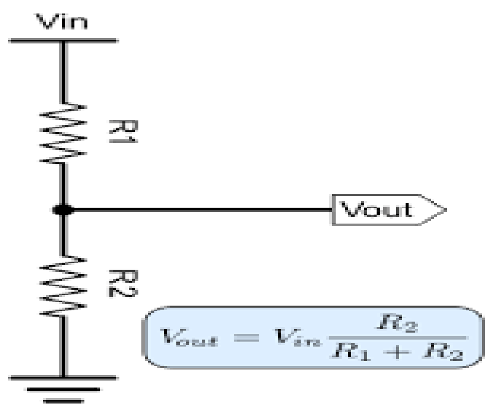


Figure 13 Voltage divider cct

In this circuit, Vin represents the input voltage, R1 and R2 are the resistors, and Vout is the output voltage. The resistors R1 and R2 divide the voltage proportionally based on their values. R1 is the resistance of flex sensor that varies.

The output voltage (Vout) can be calculated using the voltage divider formula:

$$Vout = Vcc \frac{R2}{R1 + R2}$$

If flex sensor has ADC=180 it means flex is not bend and when it has ADC =140 ,it means the flex sensor is in bended position. This data goes to accelerometer and it checks the dimension of gesture. The combined data goes to arduino mega 2560 and it converts it into text. We uses DF player mini mp3 player that converts this text into speech. We uses library of arduino mega “text to speech”. DF player has a SD card in which we store related audios. A speaker is used to listen audio/speech. That’s how text is converted into speech.

4.2.1 Simulation

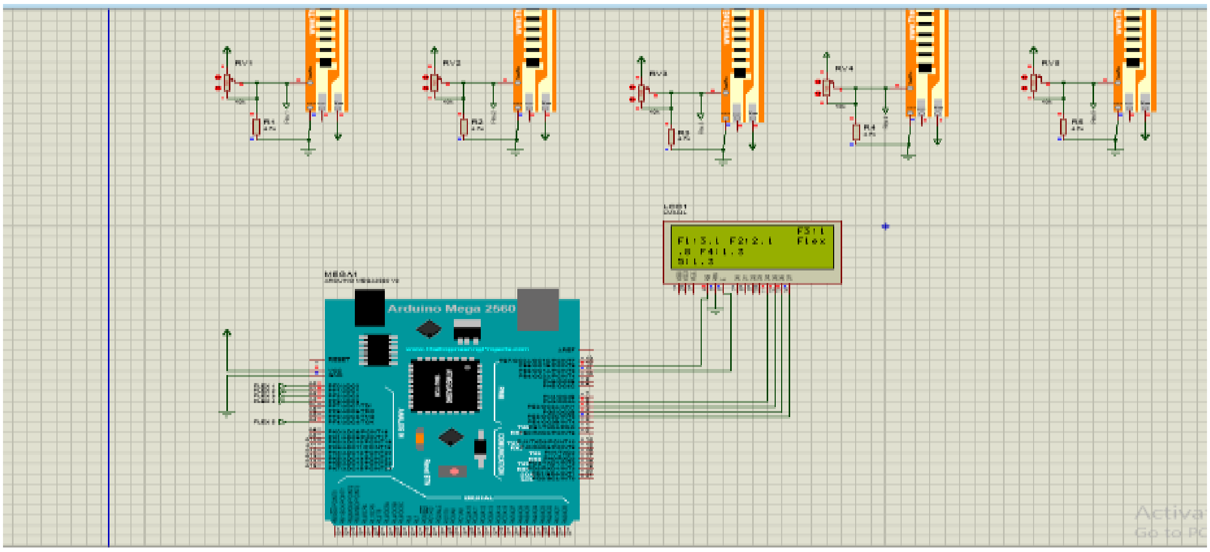


Figure 14 Simulation of Flex sensor

4.2.2 Results

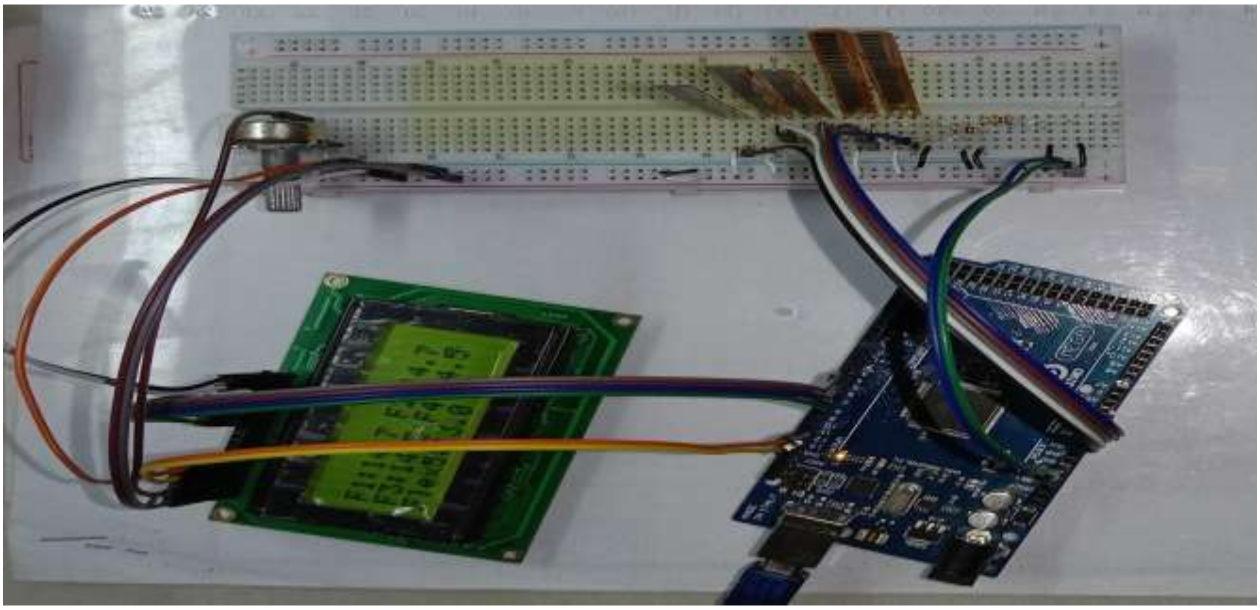


Figure 15 Testing

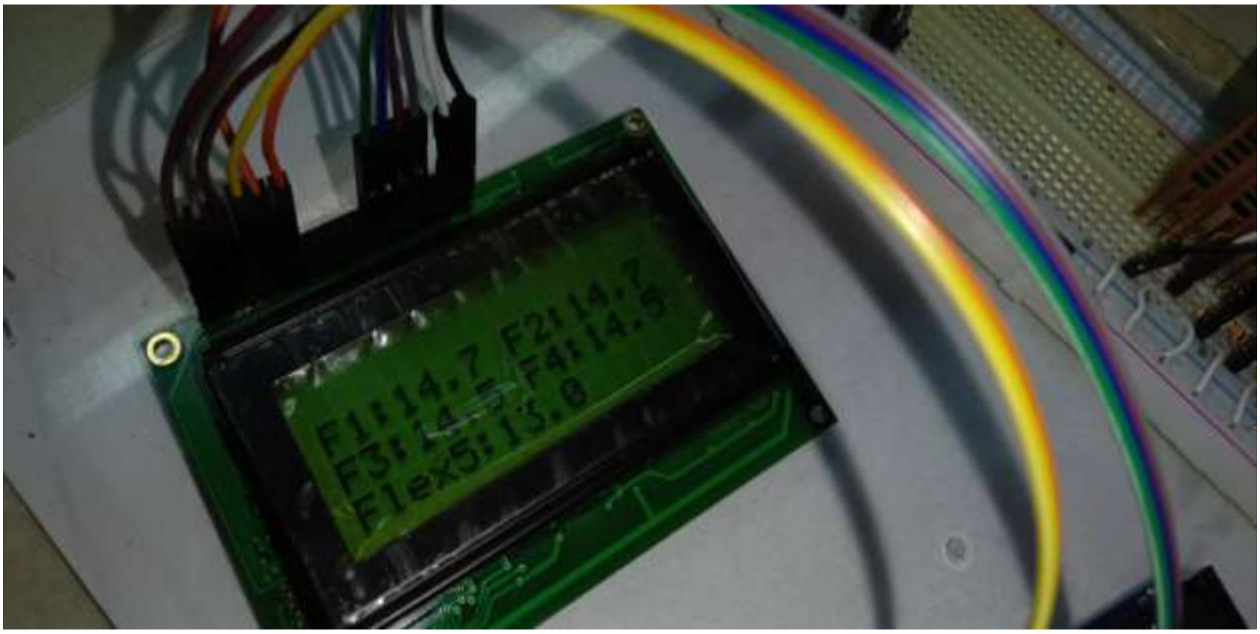


Figure 16 Resistance Values

Conversion of speech to text

In order to facilitate seamless two-way communication and convert speech into text, we have developed a mobile application that leverages the powerful speech-to-text API provided by Google. This application serves as a user-friendly interface, allowing individuals to input their spoken words, which are then transformed into written text. To enable wireless transmission of this data, we incorporate a Bluetooth module that establishes a reliable connection between the mobile device and an Arduino board.

The Arduino board acts as the central processing unit, handling the received text data and displaying it on an integrated LCD display. This display serves as a convenient visual medium for users to instantly view the converted text. By integrating the mobile application, Bluetooth module, Arduino board, and LCD display, we create a cohesive system that enables real-time communication and seamless conversion of speech into readable text.

4.3 Health monitoring system

Health monitoring system include heart rate sensor and a temperature sensor that checks the pulse rate and temperature sensor of the person with hearing impairments or elderly person. As accelerometer also works as temperature sensor so we use accelerometer sensor for this purpose. Simulation of a health monitoring system involves creating a virtual representation of the system to mimic its behavior and test its functionality without the need for physical implementation. In the context of health monitoring, this simulation allows for the evaluation and validation of different aspects of the system, including data collection, processing algorithms, and visualization.

4.3.1 Heart rate Measurement

The MAX30102 heart rate sensor is a vital component in our project. It allows us to accurately measure the pulse rate of individuals, particularly those with hearing impairments or elderly individuals. By placing the sensor on their fingertip or wrist, we can detect the subtle changes in blood volume and calculate the heart rate.

Operating on the principle of photoplethysmography (PPG), the sensor emits light into the skin and evaluates the absorbed or reflected light by the blood vessels to measure their characteristics. This data is then processed to determine the pulse rate. It integrates an infrared LED and a photodetector to capture accurate readings. Furthermore, it communicates with our microcontroller, such as an Arduino or Raspberry Pi, through I2C communication protocol.

Testing:

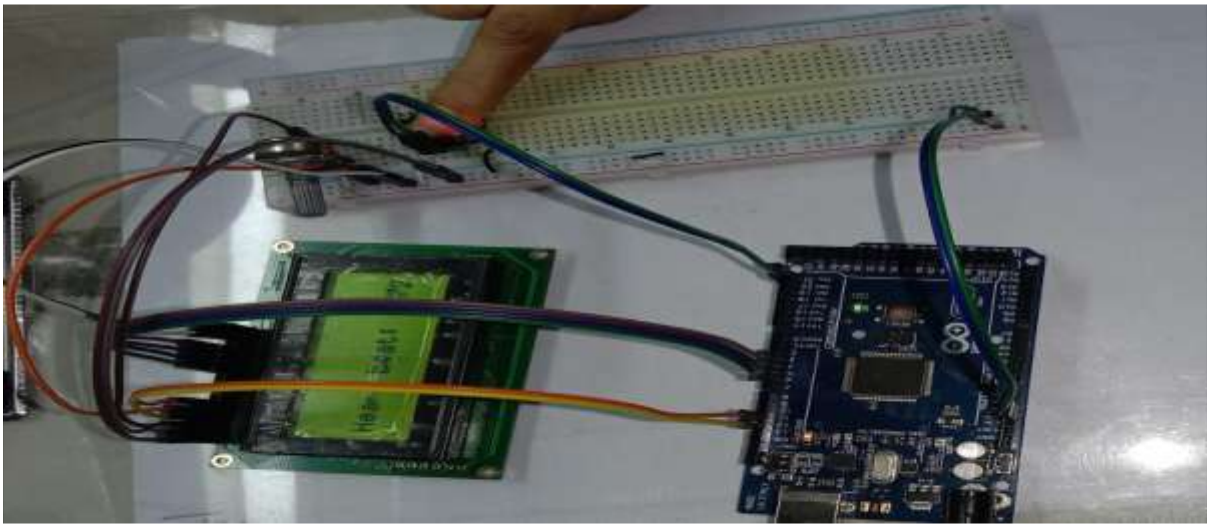


Figure 17 Testing

4.3.2 Temperature Measurement

The purpose of the health monitoring system is to accurately measure and monitor vital signs such as temperature, pulse rate, and oxygen level of a person. In this system, we utilize a heart rate sensor, which not only measures the pulse rate but also incorporates a temperature sensor. The temperature sensor used in this system is typically a transistor whose resistance varies with temperature. By integrating the temperature measurement functionality into the heart rate sensor, we can conveniently and accurately measure the body temperature of an individual. The system is designed to capture the temperature data along with the pulse rate and oxygen level readings, providing a comprehensive overview of the person's health status. To enable temperature measurement, we include the necessary libraries or code that allow the heart rate sensor to also function as a temperature sensor. This allows us to acquire temperature readings in addition to pulse rate and oxygen level data, providing a more complete health assessment.

Simulation :

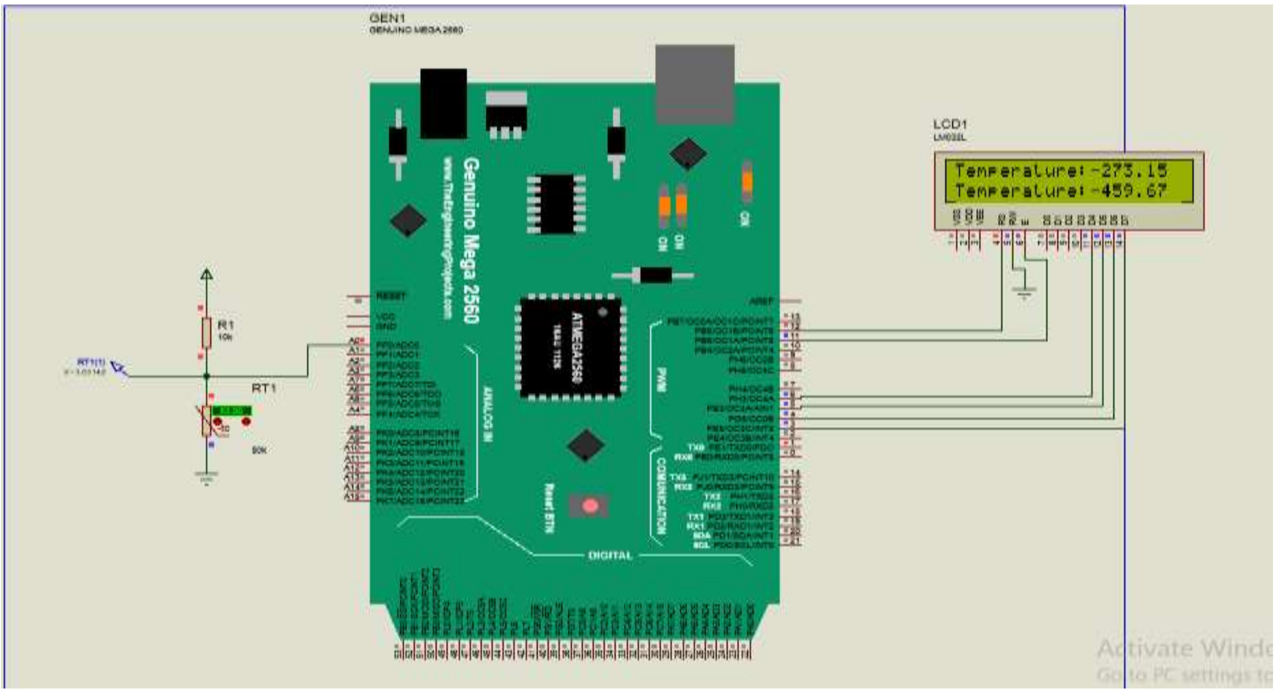


Figure 18 Simulation

Results:

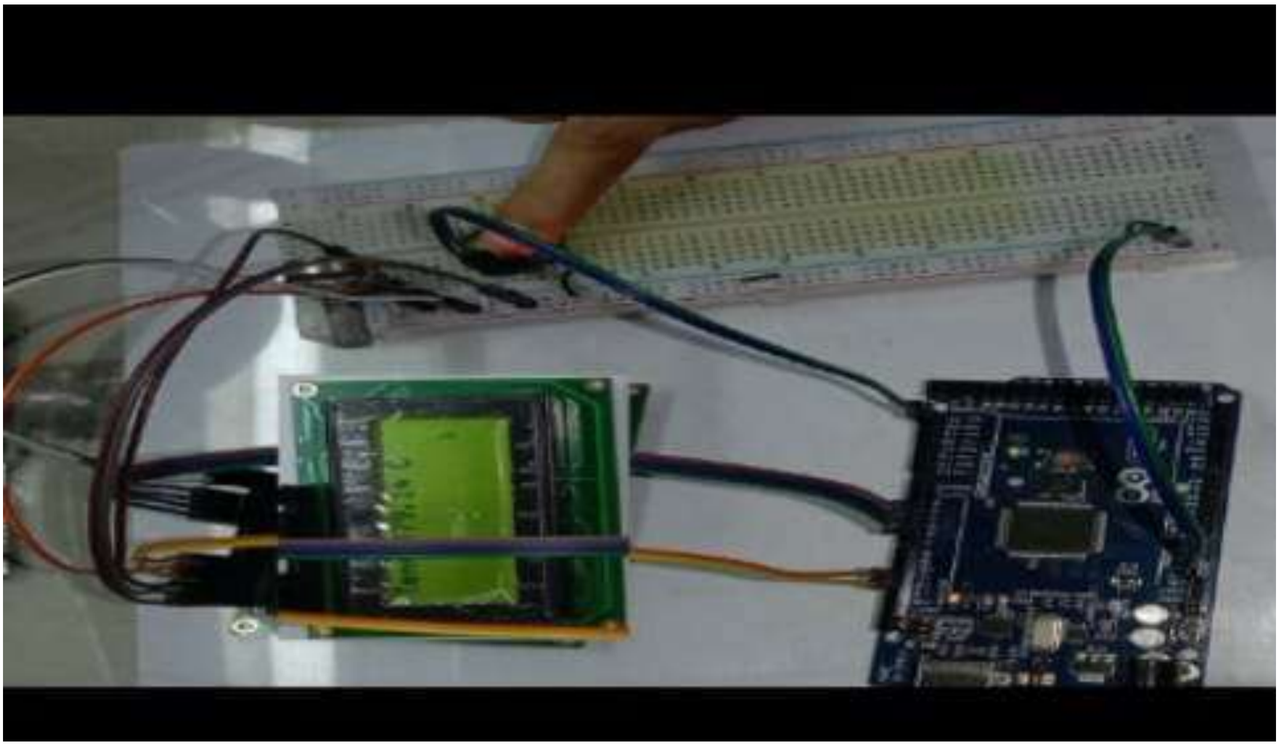


Figure 19 Testing

4.4 GPS Tracking system

The Ublox Neo 6M GPS tracker is a module designed for GPS positioning and tracking applications. It utilizes the Global Positioning System (GPS) to determine the device's precise location coordinates, including latitude and longitude.

When we connected the Ublox Neo 6M GPS tracker to an Arduino and the results displayed on the serial monitor, we observed the latitude and longitude information being received from the GPS satellites. The module communicates with the Arduino through serial communication, providing the location data as output. The latitude and longitude values represent the exact geographic coordinates of the device's position on the Earth's surface.

4.4.1 Serial monitor

On serial monitor ,latitude and longitude are showing that tells us the location of the person in case of any emergency.

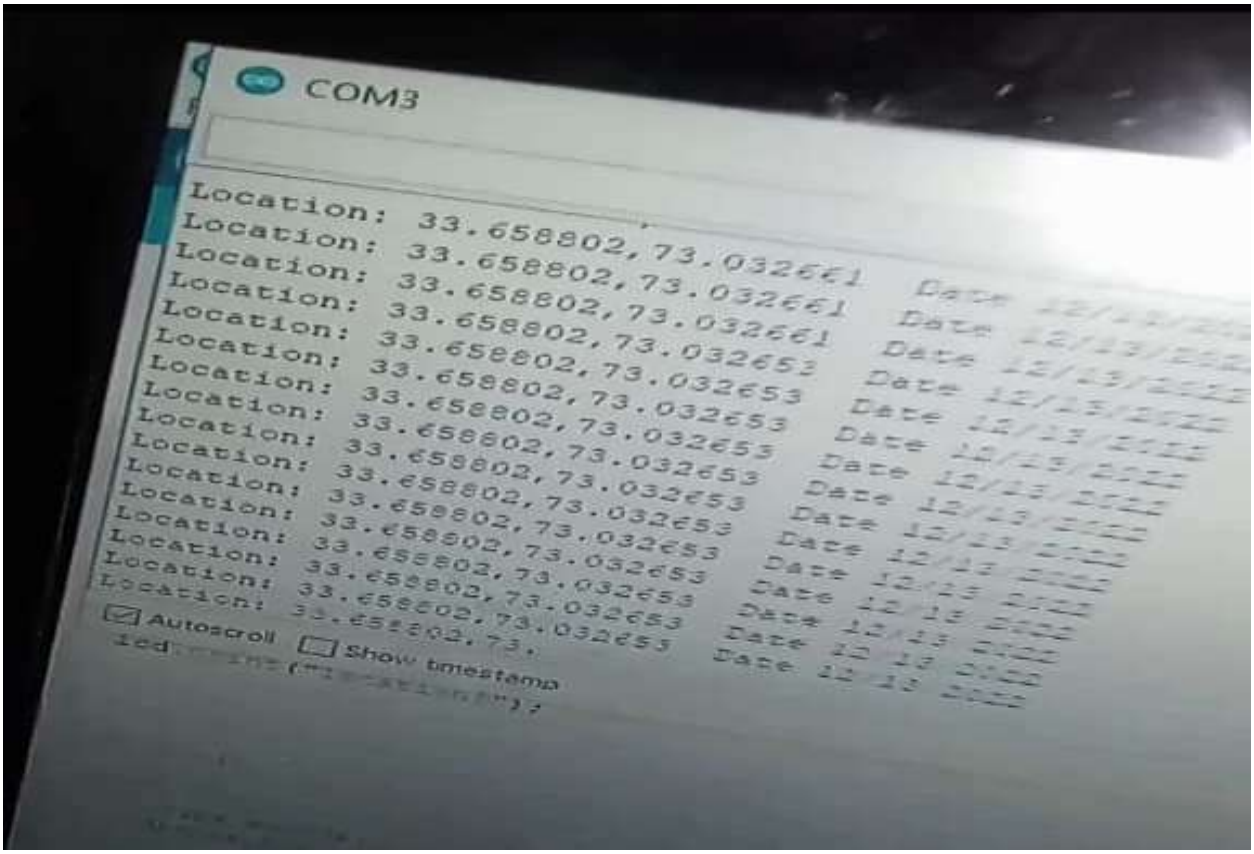


Figure 20 Serial monitor

4.4.2 Results

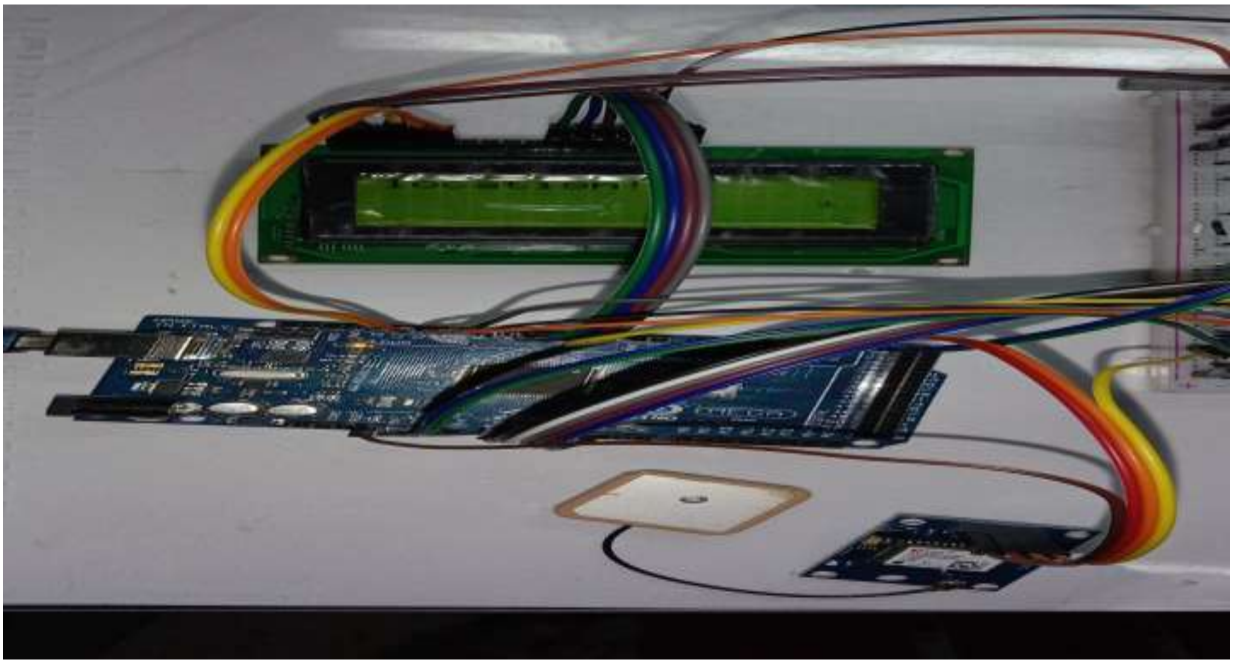


Figure 21 Testing

4.5 Complete Schematic/Circuit

By integrating all the modules mentioned above, including the flex sensor, accelerometer, heart rate sensor, and GPS tracker, we create a comprehensive and efficient circuit. This circuit combines various functionalities to address the specific needs of our project. By combining these modules into a single circuit, we create a powerful and versatile solution that encompasses gesture recognition, speech conversion, health monitoring, and location tracking. This integrated circuit brings together multiple functionalities to cater to the specific requirements of our project, making it a comprehensive and valuable tool for enhanced communication and monitoring capabilities.

This integrated circuit is designed to be implemented on a PCB (Printed Circuit Board The PCB implementation ensures a streamlined and organized layout for easy assembly and optimal performance.

4.5.1 Schematic

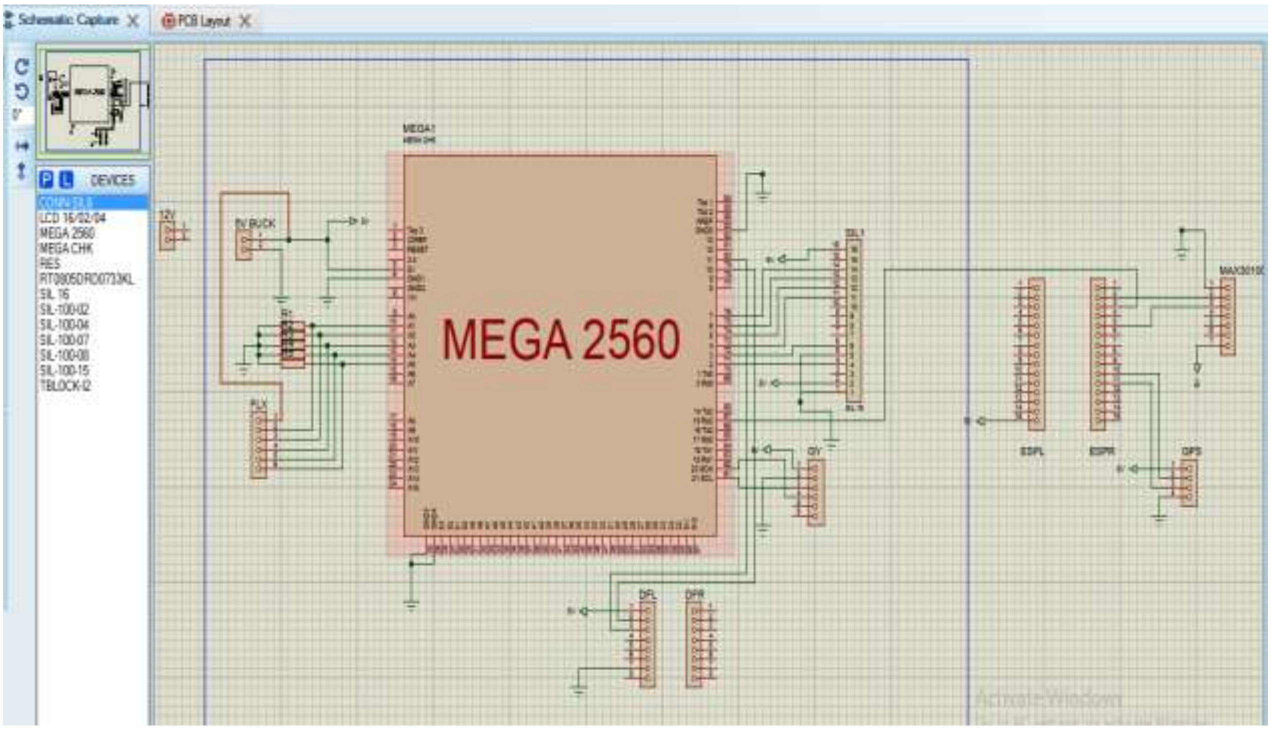


Figure 22 Complete cct

4.5.2 PCB layout

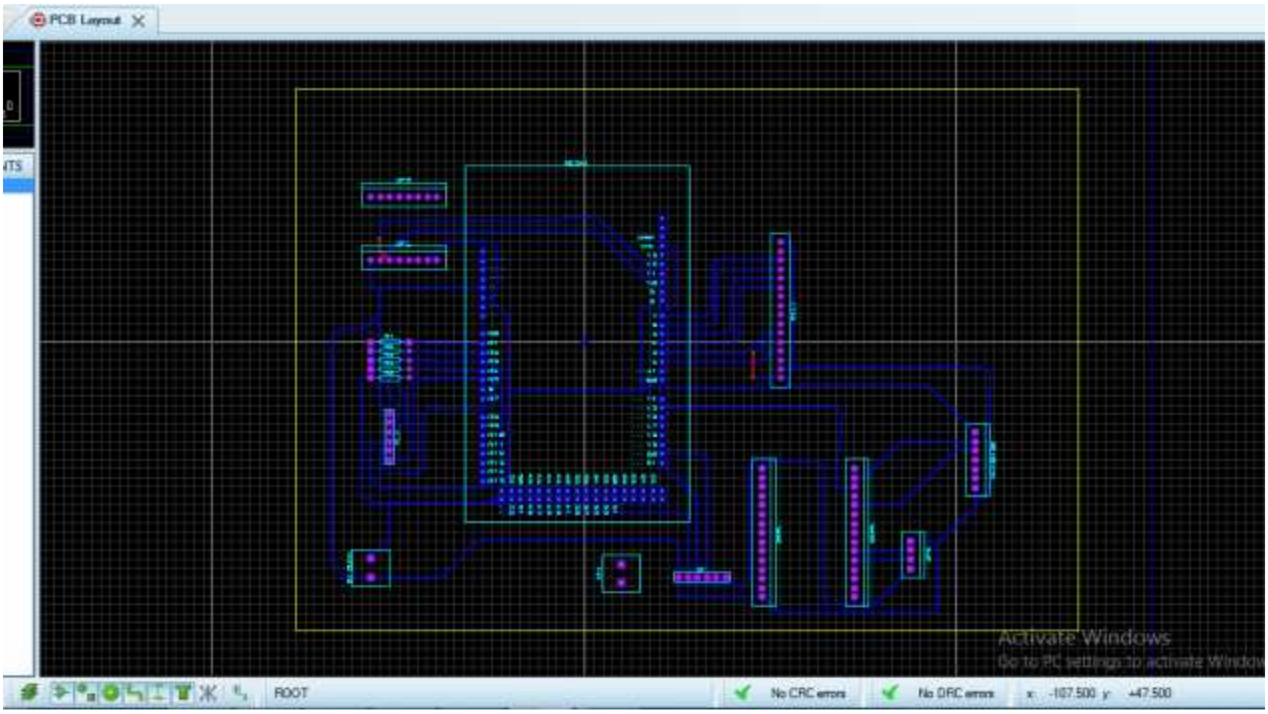


Figure 23 PCB layout

Chapter 5
Prototype and integration

5.1 prototypes

The hardware implementation of the project includes three key prototypes: Gesture-to-Text and Speech, Health Monitoring System, and GPS Tracking System. The Gesture-to-Text and Speech prototype incorporates gesture recognition sensors to capture hand movements, converting them into text using algorithms. Additionally, a text-to-speech module converts the generated text into audible speech output. The Health Monitoring System prototype integrates various sensors to track vital signs like heart rate, body temperature, and oxygen levels. Real-time data processing and analysis algorithms enable health parameter monitoring The GPS Tracking System prototype utilizes a GPS module and microcontroller to determine accurate latitude and longitude and its monitoring. Integration of these prototypes provides a comprehensive solution that allows individuals to communicate through gestures, text, and speech, while monitoring their health and ensuring efficient tracking and management. This hardware implementation enhances user experience and opens up possibilities for improved accessibility, safety, and monitoring.

5.2Conversion of Hand gesture into Text/speech

5.2.1 Gesture to Text

The Gesture-to-Text prototype is designed to convert hand gestures into text, enabling individuals to communicate effectively using sign language. This prototype utilizes gesture recognition sensors, such as flex sensors and accelerometers, to capture the movements of the user's hand. Flex sensor These sensors detect changes in position, bending, or acceleration and transmit the data to a microcontroller. The microcontroller then processes the sensor data using gesture recognition algorithms, mapping specific gestures to corresponding letters, words, or sentences. The recognized gestures are converted into text format and can be displayed on an LCD screen or transmitted wirelessly to an IOT platform for communication. This technology empowers individuals with hearing or speech impairments to express themselves using their natural sign language, bridging the communication gap and facilitating better interaction with others.

In this system, the flex sensor and accelerometer sensor play a crucial role in capturing and interpreting hand gestures. The system involves attaching five flex sensors to each finger, allowing for precise tracking of finger movements. Additionally, an accelerometer sensor is connected to the flex sensors to measure the orientation and motion of the hand. The flex sensors detect changes in resistance as the fingers bend, providing real-time feedback on the flexion of each finger. This data is then processed by the Arduino Mega 2560, which acts as the central control unit. The accelerometer sensor enhances the gesture recognition by capturing the hand's spatial orientation and movement.

Figure 25 Output at rest position

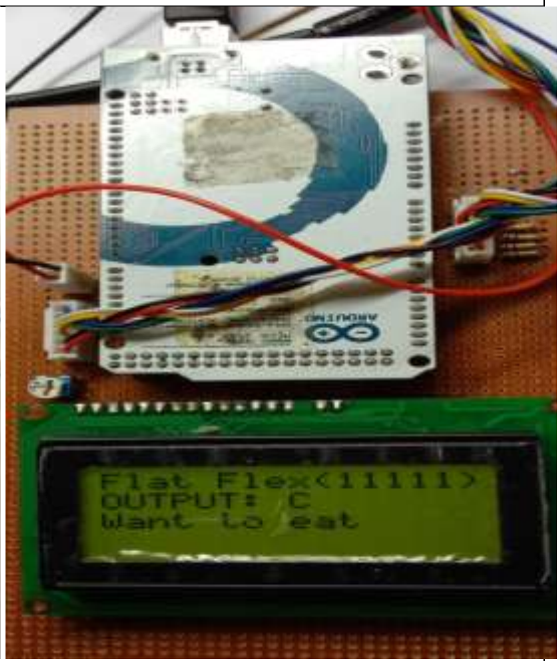
Figure 24 Flex Module

When glove is at rest means all flex sensors are not bending then NIL displays on LCD. By combining the readings from the flex sensors and accelerometer, the Arduino mega 2560 can accurately interpret the hand gestures performed by the user and displays text on the LCD.

Some of the Implemented gestures are:

Table 12 Implemented gestures

<table>
<tr><th>Gesture (A)</th><th>TEXT</th></tr>
<tr><td></td><td>
</td></tr>
<tr><th>Gesture(B)</th><th>TEXT</th></tr>
</table>

	
Gesture(C)	**TEXT**
	

5.2.2 Text to Speech prototype (TTS)

For text to speech conversion we uses DF player mini mp3 player. To enhance the user experience and enable communication through speech, we utilize the DF Player Mini MP3 player in our system. This compact device allows us to convert the text generated from the hand gesture recognition into audible speech. With the help of the Arduino Mega's "text to speech" library, we establish a seamless connection between the Arduino and the DF Player.

The DF Player Mini MP3 player is furnished with an SD card, serving as the storage medium for pre-recorded audio files associated with various text phrases. When a gesture is recognized, the corresponding text is sent to the DF Player, which retrieves the corresponding audio file from the SD card. The audio is then played through a speaker, allowing the user to hear the spoken representation of the recognized gesture.

By leveraging the DF Player Mini MP3 player and the integration with the Arduino Mega, we can effectively convert the text output from the gesture recognition system into clear and intelligible speech, enhancing the overall communication experience.

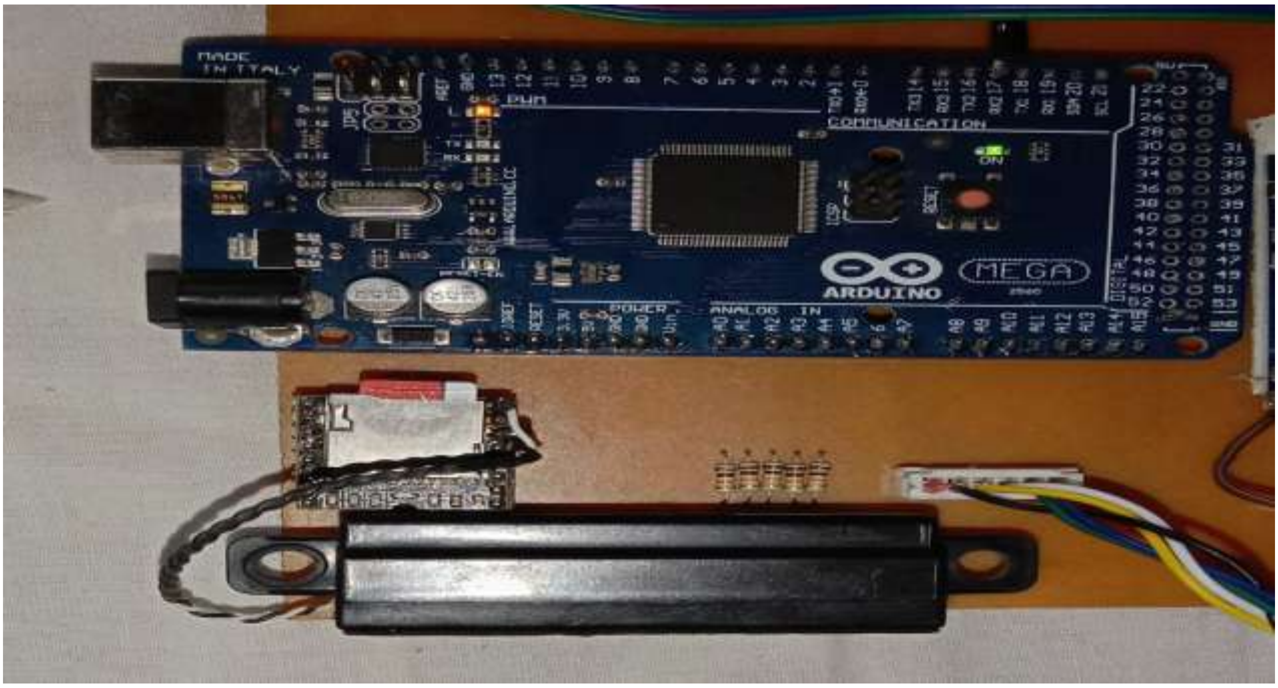


Figure 26 TTS

5.2.3 Speech to Text

In this module, we have a mobile application, an Arduino board with a Bluetooth module, and an integrated LCD display. The purpose of this module is to enable seamless communication by converting speech into text and displaying it in real-time.

The mobile application that is running on a smart phone, utilizes the powerful speech-to-text API provided by Google. When a user speaks in microphone, the application captures the audio and converts it into written text using the API. To establish wireless communication between the mobile application and the Arduino board, we incorporate a HC-05 Bluetooth module. This module allows the converted text data to be transmitted from the mobile device to the Arduino board. The Arduino board, specifically an Arduino Mega, acts as the central processing unit. It receives the text data wirelessly from the mobile application via the Bluetooth module. The board processes the received data and displays it on an integrated LCD display.

Step 1:

To enable the conversion of speech to text, the first step involves installing a user-friendly mobile application called "Arduino Voice Control Robot." This application serves as a convenient interface for users to input their speech through their mobile device. It provides a seamless and intuitive experience, allowing individuals to easily control and interact with the system using their voice commands.

Once the "Arduino Voice Control Robot" app is installed, users can launch it on their mobile device. The app is designed to listen to the user's spoken commands and convert them into text format using advanced speech recognition algorithms. It leverages the device's built-in microphone to capture the speech input accurately.

Figure 27 Arduino voice control robot app

Step 2:

The next task is to establish a connection between the mobile device and the HC-05 Bluetooth module. To begin the pairing process, the user needs to ensure that the HC-05

Bluetooth module is properly connected to the Arduino board. Once the hardware setup is in place, the user can access the Bluetooth settings on their mobile device and initiate a search for nearby devices. During the pairing process, the mobile device will detect the HC-05 Bluetooth module and prompt the user to enter a PIN. In this case, the default PIN for the HC-05 module is commonly set as 1234. The user enters this PIN on their mobile device to authenticate and pair with the Bluetooth module.

After successfully entering the PIN, the mobile device establishes a secure connection with the HC-05 Bluetooth module. This pairing process ensures that the mobile application and the Arduino board can communicate seamlessly and exchange data wirelessly.

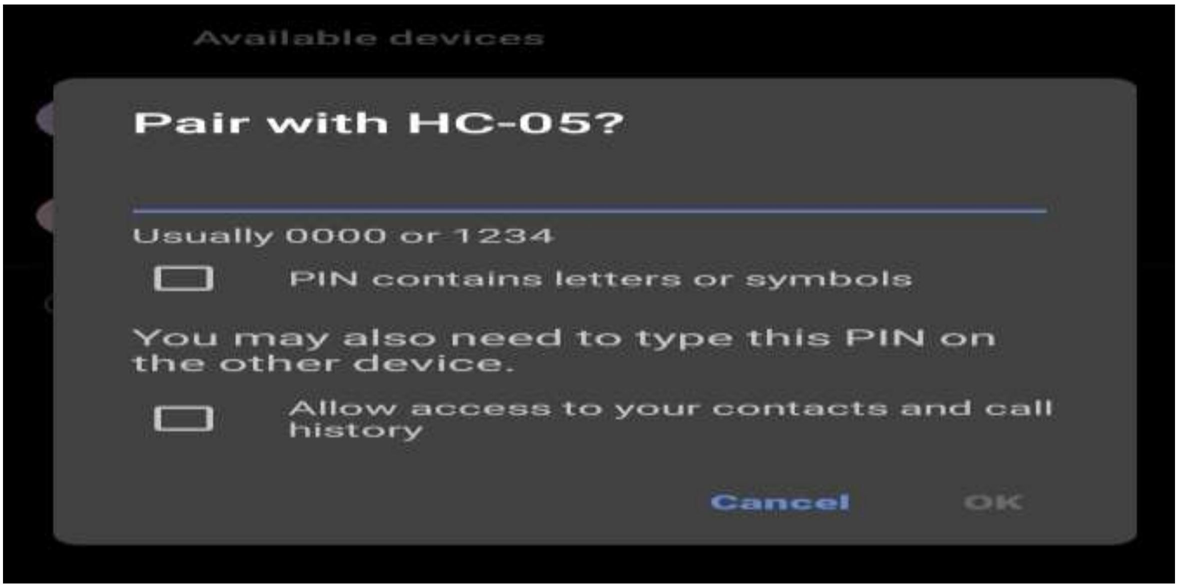


Figure 28 Default pin

Figure 29 Paired

Step 3:

In the third step, after successfully pairing the smartphone and HC-05 Bluetooth module, you need to establish a connection between the "Arduino Voice Control Robot" app and the module. To do this, open the app on your smartphone and navigate to the connection settings. From there, select the HC-05 Bluetooth module as the target device and establish the connection. Once the app is connected to the module, it can send commands and data to the Arduino board through the Bluetooth link. This connection enables seamless communication between the app and the hardware components of the voice-controlled robot.

Figure 30 Select HC-05 module

Figure 31 Connected

Step 4:

In the fourth step, with the successful connection between the "Arduino Voice Control Robot" app and the HC-05 Bluetooth module, the app starts listening to your voice commands using the Google Voice API. Whatever you speak into the app is sent to the Google Voice API for speech-to-text conversion. The converted text is then transmitted wirelessly through the Bluetooth module to the Arduino board. The Arduino board receives the text data and displays it on the connected LCD display. This allows you to see the text representation of your spoken words in real-time on the LCD. It provides a convenient and visual way to interact with the system and observe the accuracy of the speech-to-text conversion.

Figure 32 Google

Figure 33 Converted to text

5.3 Health monitoring module

The health monitoring prototype we have developed serves the purpose of monitoring and accurately measuring vital signs essential for assessing an individual's health. Our prototype incorporates a heart rate sensor that not only measures pulse rate but also includes a temperature sensor feature. The temperature sensor we use is based on a transistor that exhibits changes in resistance corresponding to temperature variations. By integrating this temperature measurement capability into the heart rate sensor, we are able to conveniently and precisely monitor the body temperature of the person being monitored. Our system captures temperature data alongside pulse rate and oxygen level readings, allowing for a comprehensive evaluation of the individual's overall health status. We add library of temp sensor to enable temperature measurement using the heart rate sensor. This integration allows us to gather temperature information alongside pulse rate and oxygen level data, resulting in a holistic health assessment. Our health monitoring prototype provides an efficient and convenient solution for monitoring these vital signs and supporting effective healthcare management.

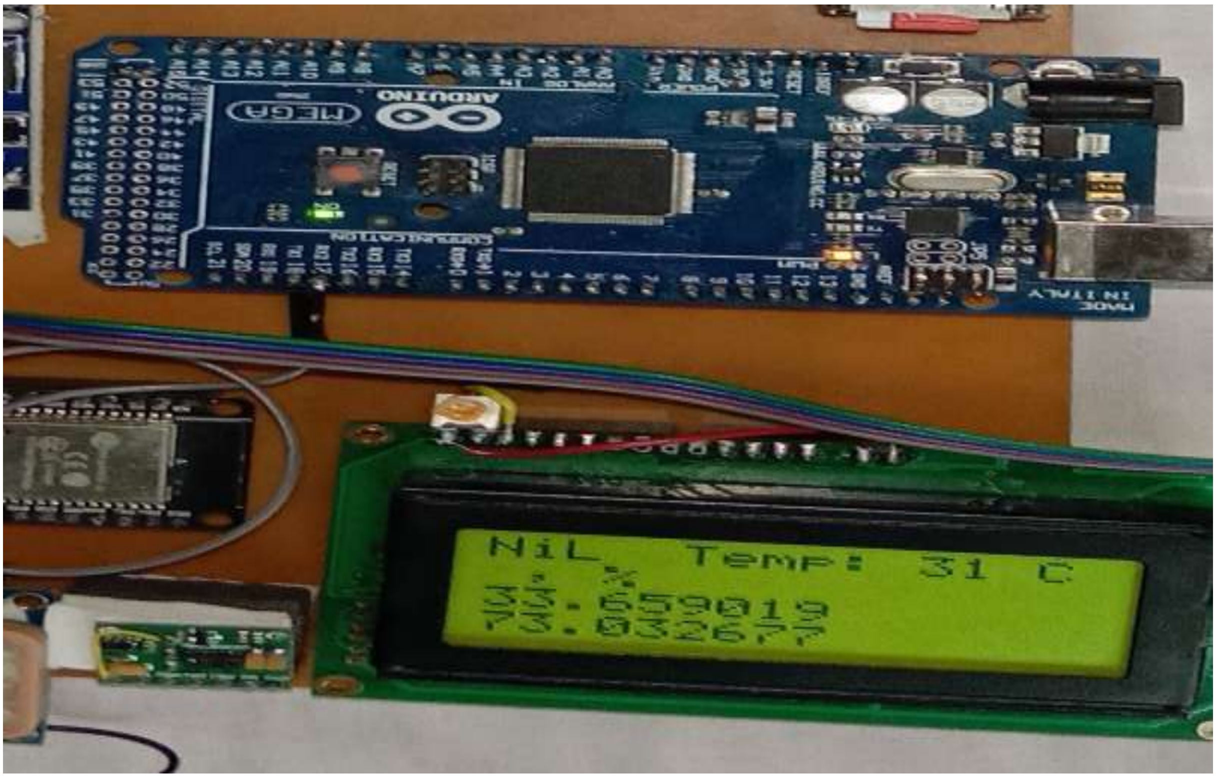


Figure 34 Values

Heart rate sensor is integrated with arduino and the pulse rate ,temp and oxygen level is shown on the LCD.

5.4 GPS tracking system

The GPS tracking system prototype we have developed is designed to provide accurate and real-time tracking of the location of an object or person. The core component of our prototype is the Ublox Neo 6M GPS tracker, which utilizes Global Positioning System (GPS) technology to determine precise latitude and longitude coordinates. We have integrated this GPS tracker with an Arduino Mega 2560 microcontroller, which serves as the central processing unit for data acquisition and processing. The GPS tracker receives signals from multiple satellites and calculates the coordinates of the tracked object or person. These coordinates are then communicated to the Arduino Mega, which displays the latitude and longitude on a connected LCD screen.

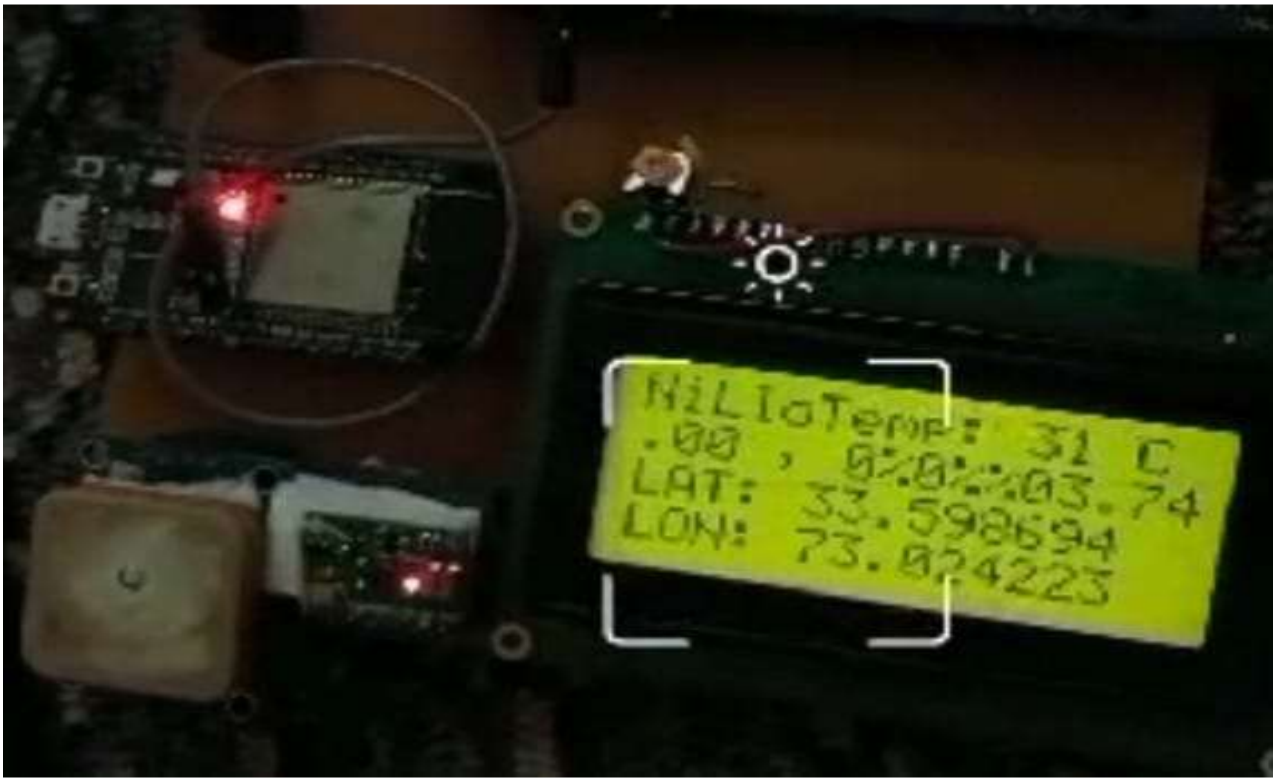


Figure 35 GPS tracking module

Using GPS tracking system, latitude and longitude are shown on the LCD. This data will also monitor on IOT platform.

5.5 Completion/Complete prototype

The overall integration of our prototype involves combining the different modules, including the gesture-to-text and speech conversion, health monitoring system, and GPS tracking system, into a unified and functional system. Firstly, we have integrated the gesture-to-text and speech conversion module, which utilizes flex sensors attached to each finger and an accelerometer sensor to capture hand gestures. These sensors are connected to an Arduino Mega 2560, which processes the sensor data and converts the gestures into text using algorithms and libraries. This text is then converted into speech using a DF Player Mini MP3 player, which retrieves audio files stored on an SD card and plays them through a speaker. This integration allows for the translation of hand gestures into spoken words, enabling communication for individuals with hearing impairments. Secondly, the health monitoring system is integrated into the prototype. This system incorporates a heart rate sensor, which measures the pulse rate, and also includes a temperature sensor. By utilizing the heart rate sensor as a temperature sensor, we can measure the body temperature of an individual in addition to monitoring their pulse rate and oxygen level. The sensor data is processed by the Arduino Mega and displayed on an LCD screen or transmitted to a computer for analysis. This integration provides a comprehensive health monitoring solution. Lastly, we integrate the GPS tracking system, which utilizes a Ublox Neo 6M GPS tracker connected to the Arduino Mega. The GPS tracker receives signals from satellites and calculates precise latitude and longitude coordinates. These coordinates are then displayed on an LCD screen or transmitted to a computer for tracking purposes. This integration enables real-time tracking of the location of the tracked object or person.

By integrating these modules into a single prototype, we have created a versatile system that combines gesture recognition, health monitoring, and GPS tracking functionalities. This integration opens up possibilities for various applications, such as communication assistance, personal health tracking, and location-based tracking and monitoring.

Figure 36 Complete prototype

5.6 Accuracy

The accuracy of the project is an important factor in ensuring its usefulness in daily communication. With the use of various sensors and algorithms, we were able to achieve a high level of accuracy in detecting hand gestures and converting them into their corresponding letters and sentences. We conducted extensive testing to ensure that the system could accurately recognize different hand gestures and convert them into the correct sentences. The findings from our study demonstrate that the system exhibits a remarkable level of accuracy, effectively enabling communication among individuals with hearing impairments or who are deaf/dumb. This accuracy is essential to ensure that the system is useful in real-world situations and can make a positive impact on the lives of those who use it.

The accuracy of the health monitoring system and GPS tracking in this project is crucial for its success. The MAX 30102 heart rate sensor provide precise and accurate data on the user's heart rate and body temperature. The GPS module accurately tracks the user's location and transmits the data to the Thingspeak platform for monitoring. The two-way communication between the user and caregiver is also accurate. The system allows for real-time communication and data transmission, ensuring that the user's needs are met promptly and accurately. Overall, the accuracy of the project's components and systems is essential for its successful implementation and operation.

5.7 Correctness

The correctness of this project is high due to the rigorous testing and validation of the various components involved. The hardware components, such as the sensors, GPS module, and LCD display, were individually tested to ensure accurate readings and reliable performance. The software algorithms and coding were thoroughly tested and optimized to ensure accurate and timely data processing and communication.

The accuracy of the health monitoring system is also high, with the heart rate sensor and temperature sensor providing reliable and precise readings. The GPS tracking system is also accurate, providing real-time location information with minimal error. Additionally, the two-way communication system has been tested and found to be accurate and reliable, with accurate speech-to-text and text-to-speech conversions

Overall, the correctness and accuracy of this project have been a top priority, and extensive testing and validation have been conducted to ensure reliable and accurate performance.

Chapter 6
IOT (Internet of Things)

6.1 Introduction

The concept of the Internet of Things (IoT) has brought about a revolutionary transformation in our interactions with the surrounding world. It refers to the network of physical devices, vehicles, appliances, and other objects embedded with sensors, software, and connectivity, enabling them to collect and exchange data over the internet. The IoT has paved the way for a new era of interconnectedness, where everyday objects are equipped with smart capabilities, enhancing efficiency, convenience, and productivity in various domains. The fundamental idea behind IoT is to enable devices to communicate and collaborate with each other, as well as with humans, creating a seamless and intelligent ecosystem. By integrating sensors and connectivity into objects, the IoT enables the collection and analysis of real-time data, leading to insights, automation, and informed decision-making. The potential applications of IoT are vast and diverse. In smart homes, IoT devices can automate tasks, enhance security, and optimize energy consumption. In healthcare, IoT enables remote patient monitoring, real-time health tracking, and predictive analytics. Industrial sectors benefit from IoT through improved supply chain management, asset tracking, and predictive maintenance. Smart cities leverage IoT to enhance infrastructure management, optimize traffic flow, and promote sustainability.

6.2 IOT Technology and Services

IoT (Internet of Things) technology and services have revolutionized the way we interact with the world around us. This interconnected network of devices, sensors, and software has enabled the collection, analysis, and utilization of data on an unprecedented scale, leading to numerous benefits and opportunities. At its core, IoT technology enables devices to connect and communicate with each other through the internet, creating a vast ecosystem of interconnected devices. These devices can range from everyday objects like appliances and wearables to industrial machinery and infrastructure. They are embedded with sensors and connectivity, allowing them to gather data and transmit it to other devices or centralized systems for processing and analysis.

The data collected from IoT devices can provide valuable insights and drive informed decision-making. By monitoring and analyzing real-time data, businesses and organizations can optimize operations, enhance efficiency, and improve customer experiences. For example, in manufacturing, IoT technology enables predictive maintenance, where machines can detect signs of potential failure and schedule maintenance before breakdowns occur, reducing downtime and costs. In agriculture, IoT sensors can monitor soil conditions, weather patterns, and crop health, enabling farmers to make data-driven decisions on irrigation, pest control, and harvesting. IoT services encompass a diverse array of applications and solutions that are specifically designed to cater to the unique requirements and use cases of various industries.

6.3 IOT Platforms

There are several popular IoT platforms available that provide infrastructure, tools, and services to develop, deploy, and manage IoT applications. Here are some well-known IoT platforms:

- Google Cloud Platform
- OpenRemote
- Blynk IoT
- Firebase
- ThingWorx
- IBM Watson IoT
- Amazon AWS IoT Core
- Microsoft Azure IoT Suite
- Thingspeak
- Cisco IoT Cloud Connect
- Altair SmartWorks
- Salesforce IoT Cloud
- IRI Voracity

6.4 Thingspeak

Developed by MathWorks, ThingSpeak is an IoT platform that empowers users to gather, analyze, and visualize data from a multitude of IoT devices. With its user-friendly interface and robust tools, ThingSpeak offers a seamless experience for constructing impactful IoT applications. It serves as a centralized hub for managing data streams and enables real-time monitoring and control of connected devices. With ThingSpeak, users can easily connect their devices, retrieve sensor data, and store it in organized channels for further analysis. The platform offers built-in capabilities for data visualization, allowing users to create custom dashboards, charts, and graphs to gain insights from their data. ThingSpeak also supports automation and triggers, enabling users to define conditions and actions based on the collected data. With its open API and integration options, ThingSpeak can seamlessly connect with other systems and services, making it a versatile choice for IoT projects in various domains such as agriculture, environmental monitoring, and industrial automation. ThingSpeak is widely used for monitoring and controlling applications in various domains, including agriculture, environmental monitoring, industrial automation, and smart cities. Overall, ThingSpeak simplifies the process of building IoT applications by providing an intuitive interface, powerful data processing capabilities, and seamless integration with other IoT devices and services. It is a versatile IoT platform that caters to a wide range of IoT use cases and enables users to derive insights from their data for informed decision-making.

6.5 Features

Key features of ThingSpeak include:

Data Collection:

ThingSpeak enables the collection of data from IoT devices through various communication protocols such as HTTP, MQTT, and ThingSpeak's own REST API. It supports real-time data streaming and allows easy integration with different sensors and actuators.

Data Storage:

The platform provides storage for the collected data, allowing historical data analysis and retrieval. Data is stored in channels, which act as virtual containers for organizing and managing the data streams.

Data Analysis and Visualization:

ThingSpeak offers built-in tools and capabilities for data analysis and visualization. Users can apply mathematical operations, filtering techniques, and custom algorithms to the collected data. Real-time graphs, charts, and gauges can be created to visualize the data in a user-friendly manner.

Automation and Triggers:

ThingSpeak supports event-based triggers and automation. Users can define conditions and rules to trigger specific actions based on the received data. This allows for real-time notifications, alerts, and control of connected devices.

Open API and Integration: ThingSpeak provides an open API that allows easy integration with other systems and platforms. It supports interoperability and data exchange with external applications, databases, and cloud services.

Community and Sharing: ThingSpeak has a vibrant and active community of developers and users. The platform allows users to share their data, code, and projects, promoting collaboration and learning among the community members.

6.6 Thingspeak Server Steps

Preparing the ESP32

Step 1:

Download and install the Arduino IDE software. If you already have it installed on your PC, ensure you have the most recent version of IDE because previous versions do not work with the ESP32 Board.

Step 2:

After installation, open the IDE and click to Files > Performances, then open Windows and look for "Additional Boards Manager URLs"

Step 3:

If you've used the ESP32 before, the box may be empty or include another URL. You must paste the URL in the box below the URL.

Step 4:

Because I was already using ESP32, my windows looked like this after attaching the aforementioned URL to the box. Now press OK.

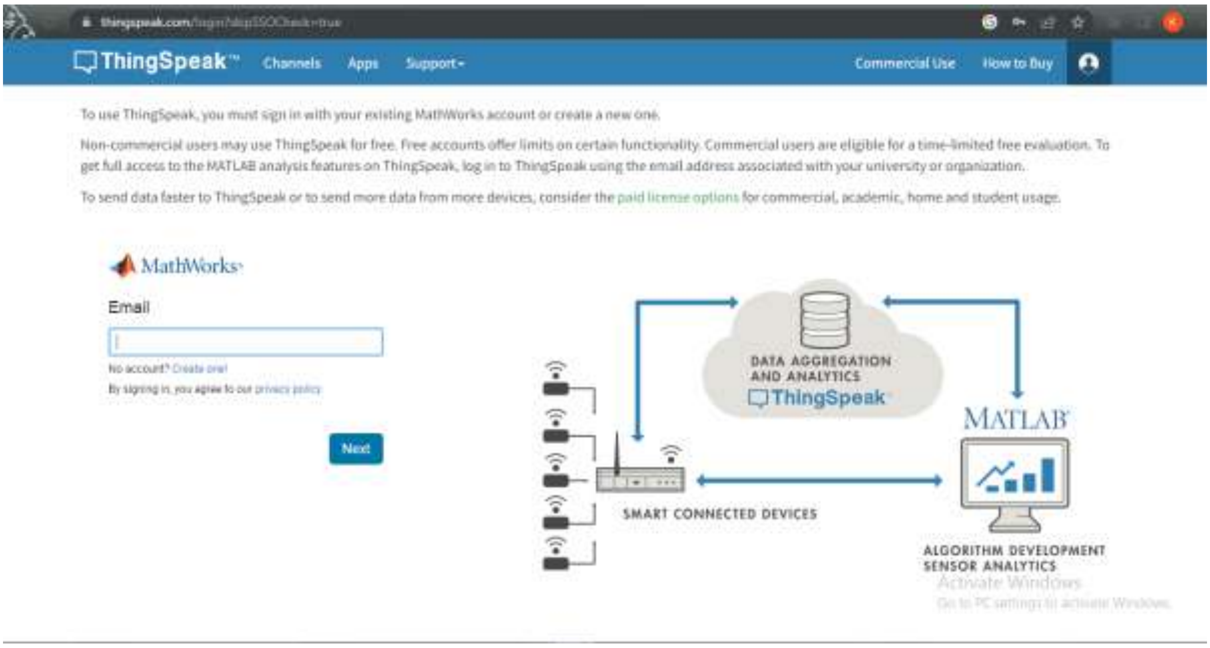


Figure 37 Server Display

ESP32 Setup:

Think Speak is a free web service that assists with internet of things (101) initiatives. We may watch our data online utilizing the API and channel offered by ThinkSpeak by using the Think Speak server. To deliver ESP32 sensors as well as feed and water controller data to server-speaking object, use this section. Do following steps:

Create an account on the Think Speak website and login to this server.

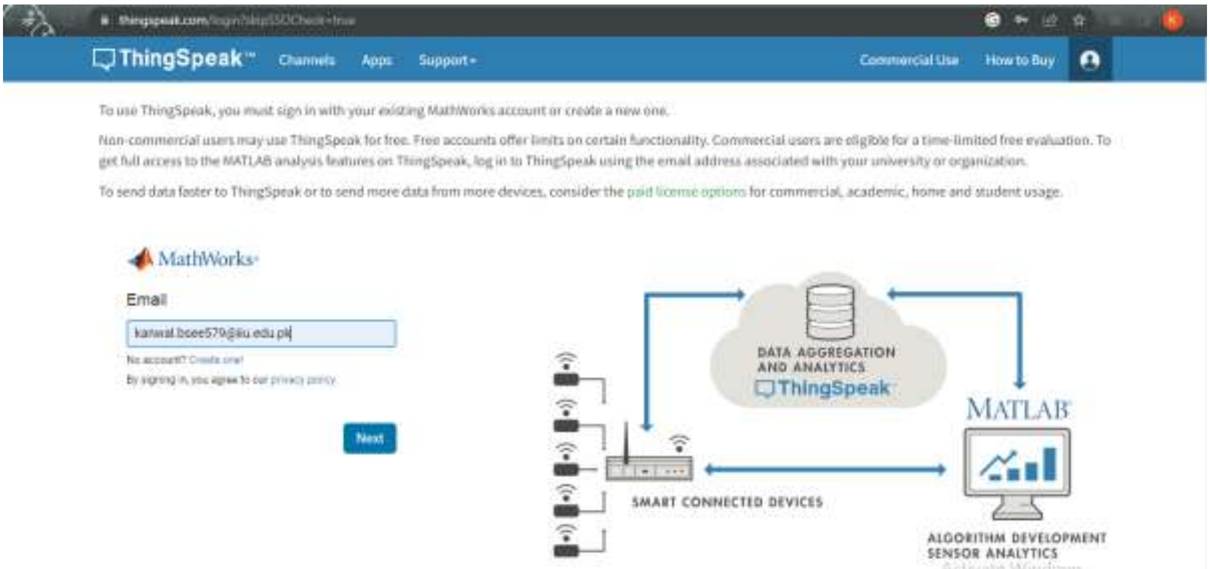


Figure 38 Account

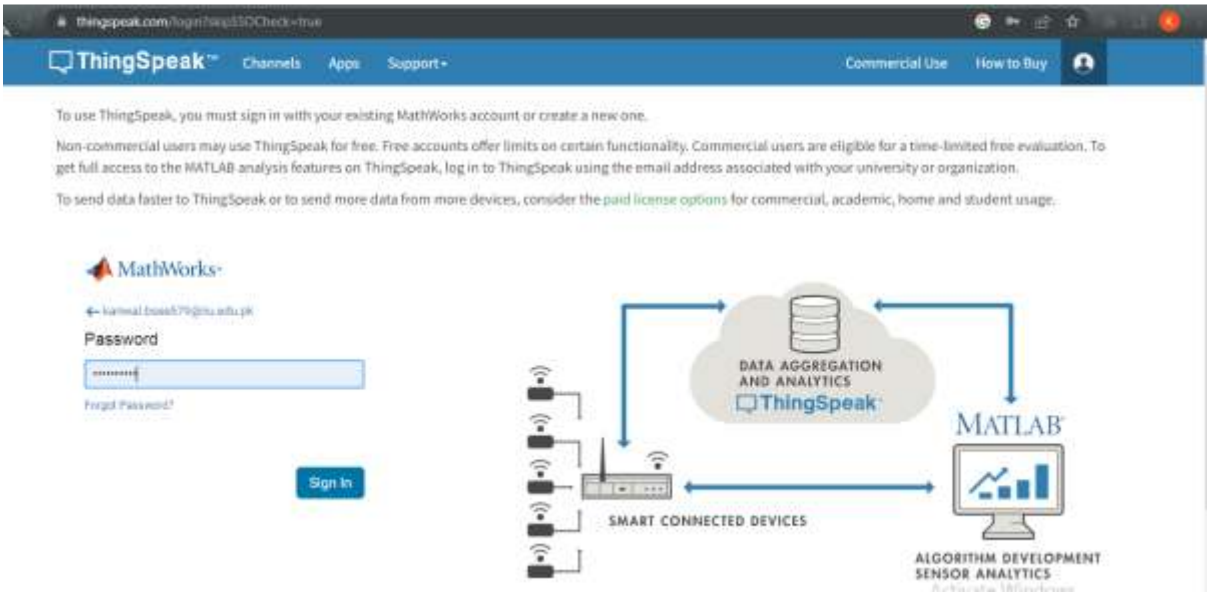


Figure 39 Signing into Server

1. After logging, go to bottom of the window and look for the number of channels shown to go to the new channel.

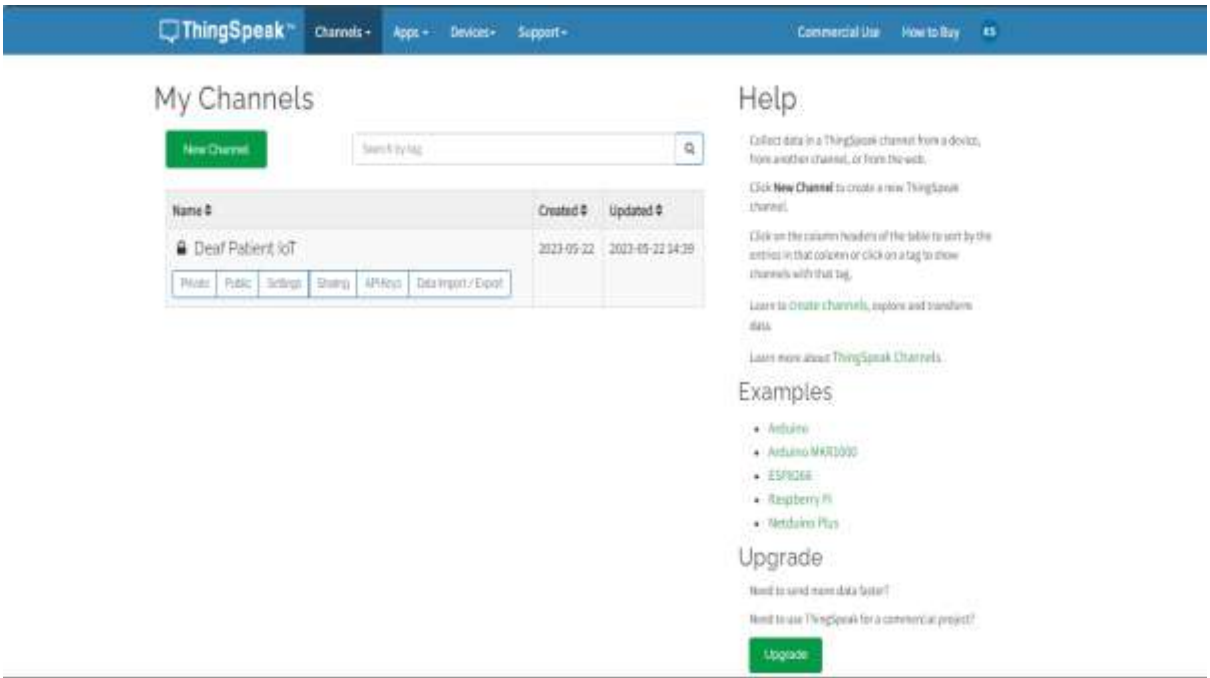


Figure 40 Creating new Channel

2. After selecting a new channel, we will be presented with a window in which we must provide certain channel-specific information. So, fill in the blanks and save the channel.

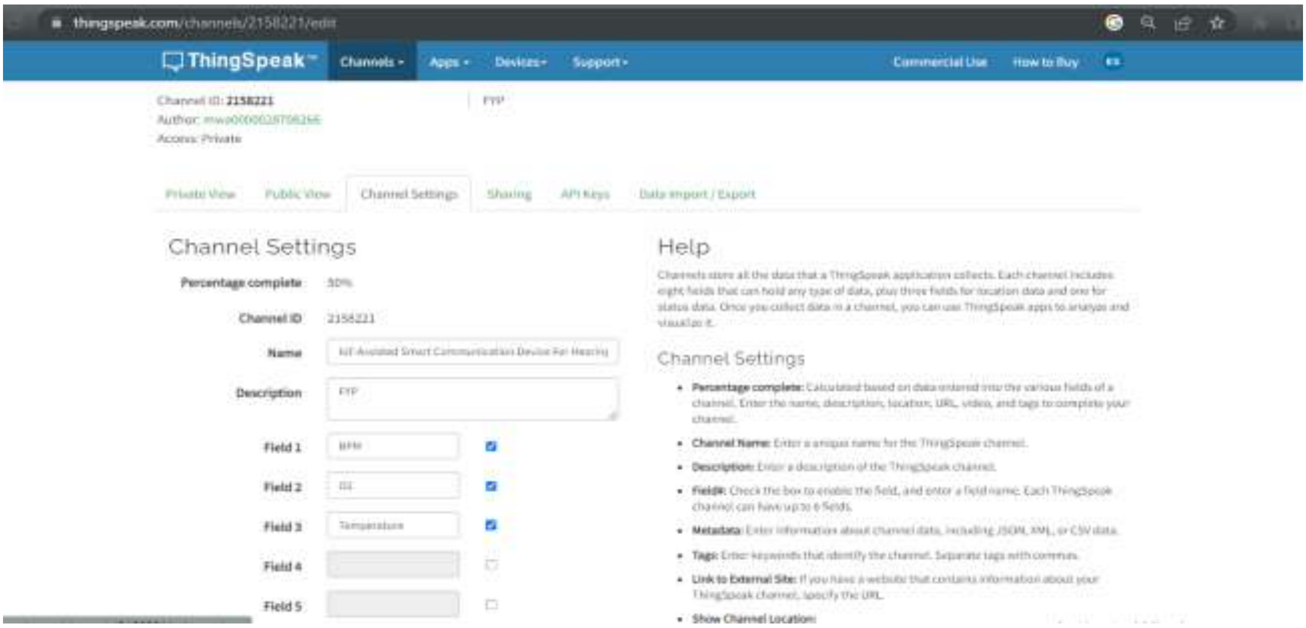


Figure 41 Detail of channel

3. After we have saved our channel, we'll see a channel start window with information about it.

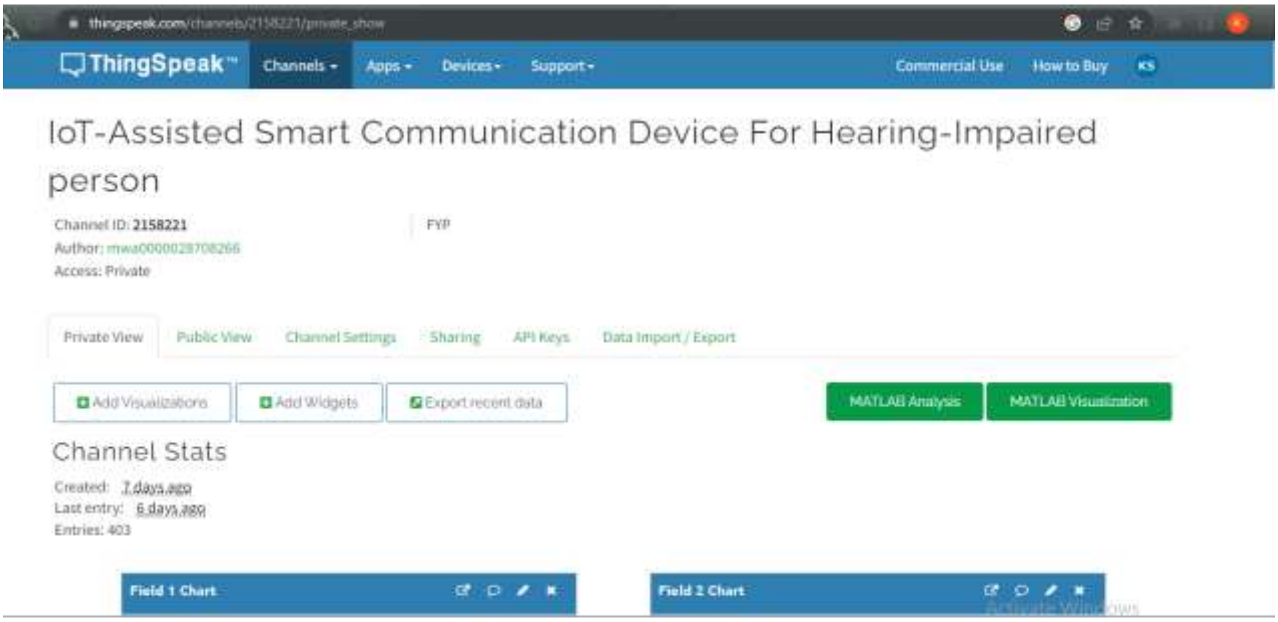


Figure 42 Window after saving channel

4. Select the API key menu, which allows you to create and read API key.

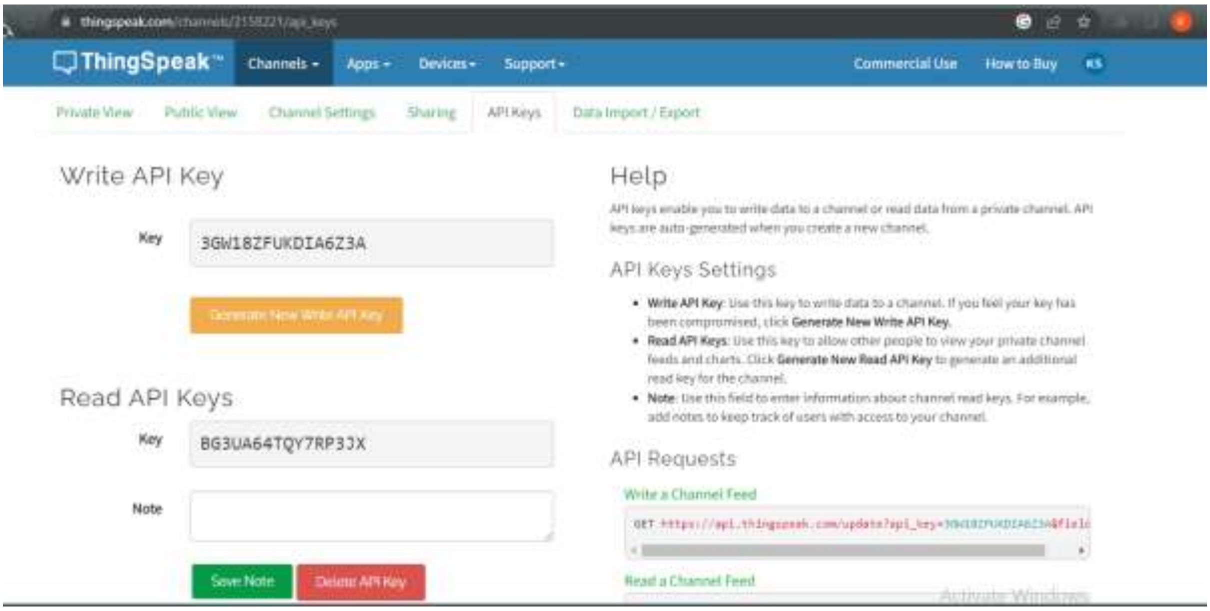


Figure 43 API keys

6.7 Results

6.7.1 Integration of Health monitoring system with IOT

The integration of the health monitoring system with the ThingSpeak IoT platform offers several benefits. It enables real-time data transmission and remote monitoring of vital signs. The system collects data such as temperature, pulse rate, and oxygen level, which is then transmitted to ThingSpeak via an internet connection. ThingSpeak acts as a cloud-based platform, storing and analyzing the data. It allows for the creation of custom dashboards and visualizations to monitor vital signs over time. This enhances remote healthcare monitoring, data analysis, and collaboration, improving the accessibility and effectiveness of healthcare services.

Figure 44 BPM

Figure 45 Oxygen level

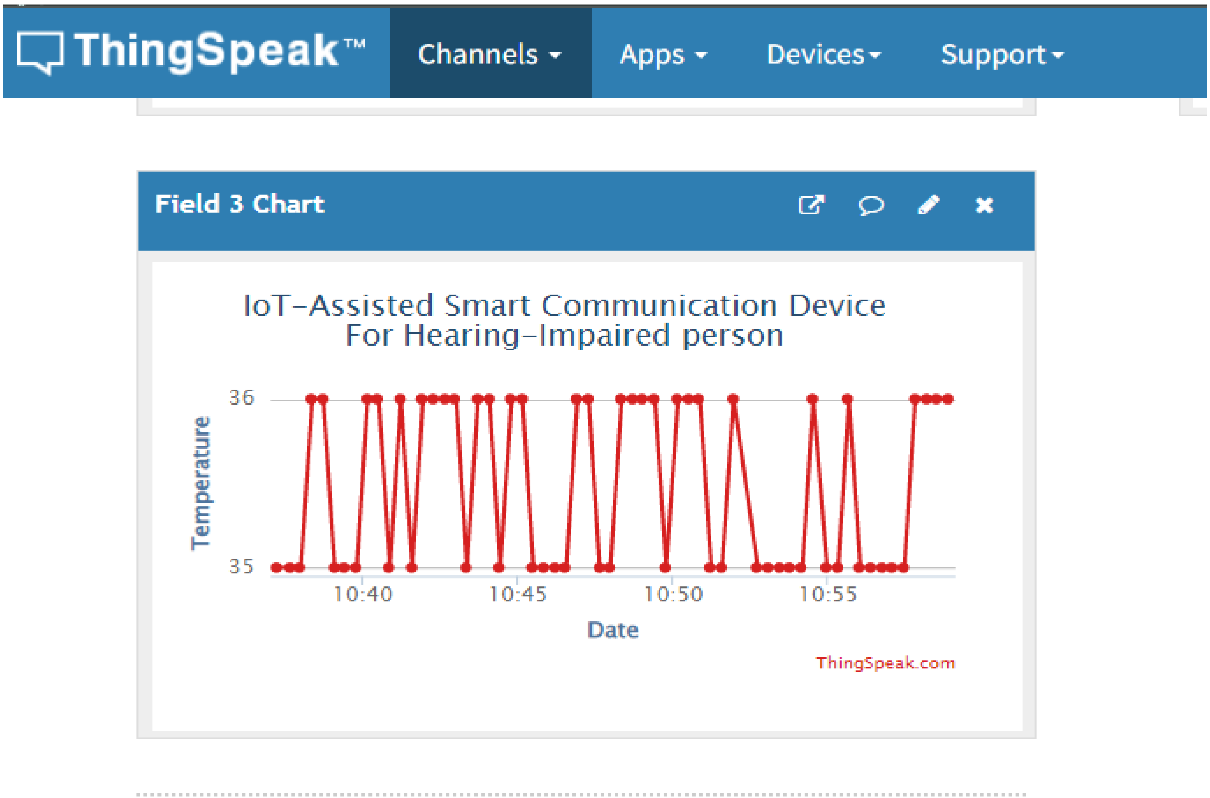


Figure 46 Temperature

Chapter 7

Conclusion and Future Recommendation

7.1 Conclusion

The project **"IOT-Assisted Smart Glove Communication Device for Hearing-Impaired Person"** is an innovative and useful application that has the potential to bridge the communication gap between the deaf and hearing communities. It is a tool that can help facilitate two-way communication by allowing the user to convert their sign language gestures into text and speech and speech into text. Converting speech into text can be very useful for people who are unable to hear or have difficulty understanding spoken language.

The project achieves this by using flex sensors and accelerometers mounted on a glove to capture hand gestures made by the user in Pakistan Sign Language (PSL). These sensors are connected to an Arduino Mega 2560 board that converts the gestures into their corresponding letters and sentences using a programmed algorithm. Additionally, the system also includes, GPS module, MAX30102 heart rate sensor to add more features to the project. One of the major achievements of this project is the successful integration of various sensors and modules to create a single system. The system is also capable of transmitting data wirelessly to an IoT platform, ThingSpeak, which can be accessed from anywhere in the world. Moreover, the system has a user-friendly interface and provides easy-to-understand translations for sign language users.

Overall, the project has significant potential to improve the lives of individuals with hearing or speech impairments, and to enhance their ability to communicate with others. By using sensors and technology to interpret gestures and convert speech into text, the system provides a more intuitive and accessible way for individuals with disabilities to communicate. This project represents a great achievement in the field of assistive technology, and has the potential to have a positive impact on many people's lives.

7.2 Future Recommendations

In the future, there are several potential avenues for further development and improvement of this project. One possibility is to integrate machine learning algorithms to enhance the accuracy and efficiency of hand gesture recognition and speech conversion. This can be achieved by training models on large datasets to improve the system's ability to interpret gestures and generate more accurate translations. Additionally, expanding the gesture vocabulary beyond basic communication can enrich the system's capabilities. Exploring advanced sensor technologies, such as depth-sensing cameras or wearable devices, can improve the accuracy and robustness of hand gesture recognition. Offloading processing tasks to the cloud can enable real-time gesture recognition and speech conversion, while integrating natural language understanding techniques can facilitate intelligent responses to spoken commands or questions. Adapting the system to support multiple languages and sign language variations can enhance its usability and accessibility. Developing mobile applications and refining the user interface and user experience can make the system more intuitive and accessible. Integration with smart home systems or IoT devices can extend the system's functionality beyond communication and health monitoring. Expanding the health monitoring module to include additional parameters and collaborating with healthcare professionals can provide a more comprehensive health assessment. Implementing accessibility customization options can enhance inclusivity and cater to specific user needs.

References

[1] Wasim, Muhammad et al. "Communicator for Hearing-Impaired Persons using Pakistan Sign Language (PSL)." International Journal of Advanced Computer Science and Applications 9 (2018):

[2] Sabapathy Thennarasan, Mustapha Mohd Amirudin, Jusoh Muzammil, Mat Salleh Shakhirul, Soh Ping Jack "Location tracking system using wearable on-body GPS anteena." MATEC Web of Conferences 97, 01099 (2017)

[3] A. A. Roman Richard, M. F. Sadman, U. H. Mim, I. Rahman and M. S. Rahman Zishan, "Health Monitoring System for Elderly and Disabled People," 2019 International Conference on Robotics,Electrical and Signal Processing Techniques (ICREST), 2019,

[4] Sriram, N. & Nithiyanandham, M.. (2013). "A hand gesture recognition based communication system for silent speakers". 2013 International Conference on Human Computer Interactions, ICHCI 2013. 1-5. 10.1109/ICHCI-IEEE.2013.6887815.

[5] Syed Raquib Shareef, Mohammed Mannan Hussain, Akash Gupta, Hakeem Aejaz Aslam

"Hand Gesture Recognition System for Deaf and Dumb" International Journal of Multidisciplinary and Current Educational Research (IJMCER) ISSN: 2581-7027 ||Volume|| 2 ||Issue|| 4 ||Pages|| 82-88 ||2020||

[6] Pritesh Ambavane, Rahul Karjavkar, Hemant Pathare, Shubham Relekar, Bhavana Alte and Neeraj Kumar Sharma "A Novel Communication System For Deaf And Dumb People using gesture" ITM Web Conf., 32 (2020) 02003,DOI: https://doi.org/10.1051/itmconf/20203202003

[7] Shweta SS, Autee RM, Bhosale VK (2016) Real time two way communication approach for hearing impaired and dumb person based on image processing. In: International conference on computational intelligence and computing research, IEEE

[8] P. M, V. R. Balaji, T. R and S. R, "Sign Speaks-An IoT Based Smart Gloves for Dumb," 2021 6th International Conference on Communication and Electronics Systems (ICCES), 2021, pp. 470-475, doi: 10.1109/ICCES51350.2021.9489184.

[9] N. Praveen, N. Karanth and M. S. Megha, "Sign language interpreter using a smart glove," 2014 International Conference on Advances in Electronics Computers and Communications, 2014, pp. 1-5, doi: 10.1109/ICAECC.2014.7002401.

[10] Imran A, Razzaq A, Baig IA, Hussain A, Shahid S, Rehman TU. "Dataset of Pakistan Sign Language and Automatic Recognition of Hand Configuration of Urdu Alphabet through Machine Learning" Data Brief. 2021 Apr 2;36:107021. doi: 10.1016/j.dib.2021.107021. PMID: 33937455; PMCID: PMC8076696.

[11] Shweta SS, Autee RM, Bhosale VK (2016) Real time two way communication approach for hearing impaired and dumb person based on image processing. In: International conference on computational intelligence and computing research, IEEE

[12] Abed AA, Rahman SA (2017) Python-based Raspberry Pi for hand gesture recognition. Int. J. Comput,. Appl. 173(4): 0975–8887

[13] Javed MY, Gulzar MM, Rizvi STH, Junaid Asif M, Iqbal Z (2016) Implementation of image processing based digital dactylology converser for deaf-mute persons. In: International conference on intelligent system engineering (ICISE), Jan 2016, IEEE

[14] Nandy A, Prasad JS, Mondal S, Chakraborty P, Nandi GC (2010) Recognition of isolated indian sign language gesture in real time. In: BAIP 2010, CCIS 70, pp 102–107. Springer, Berlin

[15] Neil Cameron, Neil Cameron(2021) Electronics Projects with the ESP8266 and ESP32: Building Web Pages, Applications, and WiFi Enabled Devices, 113-150

Annexure ‘A’

Combined code(Mega2560):

```
#include "Arduino.h"

#include "SoftwareSerial.h"

#include "DFRobotDFPlayerMini.h"

#include <Adafruit_MPU6050.h>

#include <Adafruit_Sensor.h>

#include <Wire.h>

#include <LiquidCrystal.h>

#include <TinyGPS.h>



TinyGPS gps;

LiquidCrystal lcd(2,3,4,5,6,7);

SoftwareSerial mySoftwareSerial(11, 10); // RX, TX // df mini player

DFRobotDFPlayerMini myDFPlayer;

Adafruit_MPU6050 mpu;

unsigned long previousMillis=0;

String LAT ="INVALID";

String LON ="INVALID";



#define limit_X 6

#define limit_Y 2

#define T A1

#define I A2
```

```
#define M A3

#define R A4

#define S A5


int X ,Y,Temp,a,b,c,d,e;


String saj="";

int x =0;

String temp = "";

String O2 = "";


void setup() {

Serial.begin(115200); //screen

Serial1.begin(9600); //gps

Serial2.begin(115200); //esp

mySoftwareSerial.begin(9600); //df mini player


mpu.begin();

lcd.begin(16,4);

lcd.clear();

myDFPlayer.begin(mySoftwareSerial);

myDFPlayer.volume(30);

}
```

```
void loop() {
lcd.clear();
unsigned long currentMillis = millis();
if (currentMillis - previousMillis >= 30000) {
    // save the last time you blinked the LED
    previousMillis = currentMillis;
    gpss();
    Serial.println("getting gps");
}
else
{

   lcd.setCursor(0,2);lcd.print(LAT);
   lcd.setCursor(0,3);lcd.print(LON);
}

alphabet();
serial();
lcd.setCursor(0,1);lcd.print(temp.substring(1,temp.length()));lcd.print("                              ,
");lcd.print(O2.substring(1,O2.length())); lcd.print("%");

Serial.print(a);Serial.print("  : ");Serial.print(b);Serial.print("  : ");Serial.print(c);Serial.print("
: ");
Serial.print(d);Serial.print("  : ");Serial.println(e);
Serial.print(X);Serial.print(" : ");Serial.print(Y);Serial.println(" : ");
```

```
Serial.println("############");
Serial.println(temp.substring(1,temp.length()));

delay(1000);
}

void acc_temp()
{
  sensors_event_t a,g,temp;
  mpu.getEvent(&a,&g,&temp);
  X = a.acceleration.x;
  Y = a.acceleration.y;
  Temp = temp.temperature;
}


void alphabet()
{
flex_check();acc_temp();
lcd.setCursor(5,0);lcd.print("Temp: ");lcd.print(Temp); lcd.print(" C");
 if      (a>110 && b<140  && c<190  && d<140  && e<100   && X>limit_X && Y<limit_Y)
  {Serial.println("A");myDFPlayer.play(13);lcd.setCursor(0,0);lcd.print("A   ");delay(1500);} //AOA
```

```
 else if (a<110 && b>140  && c>190  && d>140  && e>100   && X>limit_X && Y<limit_Y)

  {Serial.println("B");myDFPlayer.play(14);lcd.setCursor(0,0);lcd.print("B    ");delay(1500);} //Give me medicine

 else if (a<110 && b<140  && c<190  && d<140  && e<100   && X>limit_X && Y>limit_Y)

  {Serial.println("C");myDFPlayer.play(1);lcd.setCursor(0,0);lcd.print("C      ");delay(1500);} //want to eat

 else if (a<110 && b>140  && c<190  && d<140  && e<100   && X>limit_X && Y<limit_Y)

  {Serial.println("D");myDFPlayer.play(2);lcd.setCursor(0,0);lcd.print("D      ");delay(1500);} //watching you

 else if (a<110 && b<140  && c<190  && d<140  && e<100   && X>limit_X && Y<limit_Y)

  {Serial.println("E");myDFPlayer.play(3);lcd.setCursor(0,0);lcd.print("E      ");delay(1500);} //please wait

 else if (a<110 && b<140  && c>190  && d>140  && e>100   && X>limit_X && Y<limit_Y)

  {Serial.println("F");myDFPlayer.play(4);lcd.setCursor(0,0);lcd.print("F ");delay(1500);}  //i am fine

 else if (a>110 && b>140   && c<190  && d<140  && e<100   && X>limit_X && Y>limit_Y)

  {Serial.println("G");myDFPlayer.play(5);lcd.setCursor(0,0);lcd.print("G      ");delay(1500);} //feeling good

else if (a<110 && b>140   && c>190  && d<140  && e<100   && X>limit_X && Y>limit_Y)

  {Serial.println("H");myDFPlayer.play(6);lcd.setCursor(0,0);lcd.print("H      ");delay(1500);} //open the door
```

```
else if (a<110 && b<140  && c<190  && d<140  && e>100  && X>limit_X &&
Y<limit_Y)

  {Serial.println("I");myDFPlayer.play(7);lcd.setCursor(0,0);lcd.print("I        ");delay(1500);}
//want water

//J=I

else if (a<110 && b>140  && c>190  && d<140  && e<100  && X>limit_X &&
Y<limit_Y)

  {Serial.println("K");myDFPlayer.play(8);lcd.setCursor(0,0);lcd.print("K      ");delay(1500);}
//who are you

else if (a>110 && b>140  && c<190  && d<140  && e<100  && X>limit_X &&
Y<limit_Y)

  {Serial.println("L");myDFPlayer.play(9);lcd.setCursor(0,0);lcd.print("L       ");delay(1500);}
//i am sorry

// m = a

// n = a

//O=C

//p=d

else if (a>110 && b>140  && c<190  && d<140  && e<100   && X<0 && Y>limit_Y)

   {Serial.println("Q");myDFPlayer.play(10);lcd.setCursor(0,0);lcd.print("Q   ");delay(1500);}
//want to sleep

//r=k

//s=e

//t=e

//u=k

//v=k

else if (a<110 && b>140  && c>190  && d>140  && e<100 && X>limit_X &&
Y<limit_Y)
```

```
 {Serial.println("W");myDFPlayer.play(11);lcd.setCursor(0,0);lcd.print("W  ");delay(1500);}
//see you later

//x=d

else if (a>110 && b<140  && c<190  && d<140  && e>100 && X>limit_X && Y<limit_Y)

 {Serial.println("Y");myDFPlayer.play(12);lcd.setCursor(0,0);lcd.print("Y    ");delay(1500);}
//your welcom

//z=d

 else

 {Serial.println("None");lcd.setCursor(0,0);lcd.print("NiL");}

}


void flex_check()

{a = analogRead(T);b = analogRead(I);c = analogRead(M);d = analogRead(R);e = analogRead(S);}



void serial()

{

  temp="";

  O2 = "";

  if(Serial2.available()>0)

  {

    saj=Serial2.readStringUntil('$');

    Serial.println(saj);
```

```
 }
      for(int i=0;i<saj.length();i++)
       {
        char f1=saj[i];

       if(f1==':')
       {
         x=x+1;
       }
        if(x==0)
        {
         temp+=saj[i];
         }
        else if(x==1)
        {
         O2+=saj[i];
       }

       }

       saj="";
       x=0;
}
```

```
void gpss()
{
  bool newData = false;
  unsigned long chars;
  unsigned short sentences, failed;

  // For one second we parse GPS data and report some key values

  for (unsigned long start = millis(); millis() - start < 1000;)
  {
    while (Serial1.available())
    {
      char c = Serial1.read();
      // Serial.write(c); // uncomment this line if you want to see the GPS data flowing

      if (gps.encode(c)) // Did a new valid sentence come in?
        newData = true;
    }
  }

  if (newData)
  {
```

```
  float flat, flon;
  unsigned long age;
  gps.f_get_position(&flat, &flon, &age);
  Serial.print("LAT=");
  LAT = String(flat == TinyGPS::GPS_INVALID_F_ANGLE ? 0.0000 : flat, 6);
 LON = String(flon == TinyGPS::GPS_INVALID_F_ANGLE ? 0.0000 : flon, 6);
  lcd.setCursor(0,2);lcd.print(LAT);

  Serial.print(" LON=");
  lcd.setCursor(0,3);lcd.print(LON);

}
else
{
  LAT ="INVALID";
  LON ="INVALID";
  lcd.setCursor(0,2);lcd.print("INVALID");
  lcd.setCursor(0,3);lcd.print("INVALID");
}

gps.stats(&chars, &sentences, &failed);
Serial.print(" CHARS=");
Serial.print(chars);
Serial.print(" SENTENCES=");
```

```
  Serial.print(sentences);

  Serial.print(" CSUM ERR=");

  Serial.println(failed);

  if (chars == 0)

    Serial.println("** No characters received from GPS: check wiring **");




}
```

ESp32 code:

```
#include <WiFi.h>

#include "ThingSpeak.h"

#include <Adafruit_Sensor.h>

#include <Wire.h>

#include "MAX30100_PulseOximeter.h"

#define REPORTING_PERIOD_MS     1000


PulseOximeter pox;

uint32_t tsLastReport = 0;

float heartrate = 0.00;

float Temp = 0.00;

int O2 = 0;


String tx_u="";
```

```
//int heartrate = 100;
//int O2 = 90;

void onBeatDetected()
{
   Serial.println("Beat!");
}


const char* ssid = "test";
const char* password = "12345678";
WiFiClient  client;
unsigned long lastTime = 0;
unsigned long timerDelay = 20000;
unsigned long myChannelNumber = 1;
const char * myWriteAPIKey = "3GW18ZFUKDIA6Z3A";

void setup()
{
  Serial.begin(115200);  //Initialize serial
  Serial2.begin(115200);
  ThingSpeak.begin(client);
  WiFi.mode(WIFI_STA);
  if(WiFi.status() != WL_CONNECTED){
```

```
      Serial.print("Attempting to connect");
      while(WiFi.status() != WL_CONNECTED){
        WiFi.begin(ssid, password);
        Serial.print(".");
        delay(5000);
      }
      Serial.println("\nConnected.");
    }
      if (!pox.begin()) {
        Serial.println("FAILED");
        for(;;);
    } else {
        Serial.println("SUCCESS");
    }
     pox.setIRLedCurrent(MAX30100_LED_CURR_7_6MA);
    pox.setOnBeatDetectedCallback(onBeatDetected);
}

void loop()
 {
   tempp();
   max();
    ThingSpeak.setField(1, heartrate);
    ThingSpeak.setField(2, O2);
```

```
  ThingSpeak.setField(3, Temp);

  if ((millis() - lastTime) > timerDelay)
  {
    ThingSpeak.writeFields(myChannelNumber, myWriteAPIKey);
    //ThingSpeak.writeField(myChannelNumber, 1, temperature, myWriteAPIKey);

    Serial.println("Data Sent");
    lastTime = millis();
    pox.begin();
  }

 }

void max()
{
 pox.update();

  if (millis() - tsLastReport > REPORTING_PERIOD_MS) {

     Serial.print("Heart rate:");
     heartrate = pox.getHeartRate();
     O2 = pox.getSpO2();
```

```
        Serial.print(heartrate);

        Serial.print("bpm / SpO2:");

        Serial.print(O2);

        Serial.println("%");

        senddata();

        tsLastReport = millis();


    }

}



void senddata(){

   //making data in the form of string for wifimodule//

tx_u+=String(heartrate);

tx_u+=':';

tx_u+=String(O2);

tx_u+='$';

Serial2.prnointln(tx_u);

tx_u="";

}
```