TABLE OF CONTENTS

TABLE OF CONTENTS

ABSTRACT

ABSTRACT

Vision is the most beautiful and vital gift from nature to all his creatures especially for human beings. But unfortunately there are some people who lack this beauty and are unable to capture the beauty of this world from their own eyes. Third eye for blind is a development which involves various areas such as, IoT, hardware and software integration and automation that enables visually impaired persons to see and explore the world confidently and independently by recognizing the nearby objects by using ultrasonic waves and inform the person with a beep ring or vibration. As per WHO (world health organization) 2.2 billion people suffer from vision impairment. They are facing troubles in their lifestyle. This device can act as an innovation for the visually impaired people. Due to their vision, they find it difficult to navigate in public. They always require aid of different person for navigating in dense population. They also sometimes get into danger due to their vision impairment.

INTRODUCTION

CHAPTER 1

INTRODUCTION

As per the definition of blindness, we mean the person without sense of sight. A blind person has no ability to see anything. While struggling for the different levels of comforts of the general population, we have reached to a point where we have started to completely ignore the people who are living a miserable life due to lack of vision. They face enormous challenges in their daily lives and hence end up living a dependent life. They experience a completely different life from the normal people and experience detached and uninterested conduct towards them for being physically disabled. They need other individuals for their movement from one place to another.

Sight is the basic sense of life and therefore a person's movement from place to place in this condition is a major challenge for the visually impaired. The target of this task , This project for the blind or visually impaired person will provide a gadget that is helpful to them as well as the persons who depend on any individual due to lack of sight. Third eye for blind task can be an innovation for the sightless individuals, it will help them to move from here and there and among different places with confidence by knowing the nearest obstructions while wearing the band which leaves the ultrasonic waves which inform the person with beep-sound or vibration. It can let the person who is not able to move and distinguish even snags due to lack of vision. They just have to put on the gadget as a spectacle which is part of their accessories. As per WHO (world health organization) 2.2 billion people suffer from vision impairment. They experience a lot of troubles in the daily lives. This device can be an innovation for the physically disabled or blind individuals.

SYSTEM STUDY

CHAPTER 2

SYSTEM STUDY

2.1 EXISTING SYSTEM:

There are three major existing system which are widely used by blind. They are:

- White Cane.
- Vision (a torch for blind).
- Vibration gloves.

2.1.1 WHITE CANE

White Cane is a long stick used by blind to scan their environment. It is also difficult to handle in busy traffic. White Cane also causes huge inconvenience to general public.

DISADVANTAGES

- May crack or break easily.
- The stick may get stuck at the foot path cracks.
- Difficult to handle at crowded place.
- High chances for getting missed.

2.1.2 VISION (A TORCH FOR BLIND)

Vision is a smart device which consists of ultrasonic sensor which resides inside a box. When an object is detected, buzzer sound is produced. The blind has to carry the box manually. The whole system is controlled by a micro controller such as Arduino, Arduino Nano, and ESP 32, Etc.

DISADVANTAGES

- Difficult to carry.
- High chances of getting dropped.
- Inconvenient to use.

2.1.3 VIBRATION GLOVES

This system consists of a glove attached with ultrasonic sensor. When an object is detected vibrator motor produces vibration that notifies the blind. The entire system is controlled by a micro controller such as Arduino, Arduino Nano, ESP 32, etc.

DISADVANTAGES

- Difficult to handle.
- Difficult to perform other works wearing the gloves.
- May cause strain in the client hand.

2.2 PROPOSED SYSTEM

We are going to build a system that enables blind to detect object with ease. It consists of two ultrasonic sensors along with buzzer and vibration motor to notify the user.

ADVANTAGES

- Light weight.
- Portable.
- Requires very low skill to handle.
- Convenient to use.

SYSTEM REQUIREMENTS

CHAPTER 3

SYSTEM REQUIREMENTS

System requirements refer to the detailed descriptions of what a system must accomplish or possess to meet the needs and expectations of its users or stakeholders.

3.1 HARDWARE REQUIREMENTS

- ESP 32 : Micro controller (controls the system).
- Ultrasonic sensor : To detect the nearby object.
- Vibration motor : To notify the user.
- Buzzer : To notify the user.
- Wires : To complete the circuit.
- GPS module : To fetch user current location.

3.2 SOFTWARE REQUIREMENTS

- Language Used : C++
- Platform : Arduino IDE.

PROJECT DESCRIPTION

CHAPTER 4

PROJECT DESCRIPTION

The goal of the project is to develop a system which is used by blind to navigate to places easily. The system uses a pair of ultrasonic sensor to detect the object. Once any object is detected, it notifies the user with buzzer sound and vibration pulses. In case of any emergency, the user can expose their current location to the target by pressing a push button. The location is fetched using GPS module.

4.1 MODULE DESCRIPTION

The "IoT BASED THIRD EYE FOR VISUALLY CHALLENGED PEOPLE" system will be developed using ESP 32 as microcontroller as the microcontroller and Arduino as programming platform. The application will have three modules, each with a specific function are.

- Object Detection.
- Buzzer and Vibrant notification.
- Sending SOS message.

4.1.1 OBJECT DETECTION

- In object detection phase the nearby objects are detected by the ultrasonic sensor.
- Once an object is detected the sensors transmits signal to the micro controller.

4.1.2 BUZZER AND VIBRANT NOTIFICATION

- In detection phase the microcontroller makes the buzzer and vibration motor to produce their corresponding output.
- These outputs will notify the user about the incoming objects in his path.

4.1.3 SENDING SOS MESSAGE

- In case of emergency, the user can send SOS message to the target what's app number along with his current location.
- The user current location is fetched using GPS module.
- The location is sent to the target what's app number using Twilio API.
- Hence the location of the blind can be found with ease.

SYSTEM DESIGN

CHAPTER 5

SYSTEM DESIGN

System design refers to the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. It involves translating requirements into a blueprint that outlines the system functions.

5.1 CIRCUIT DIAGRAM

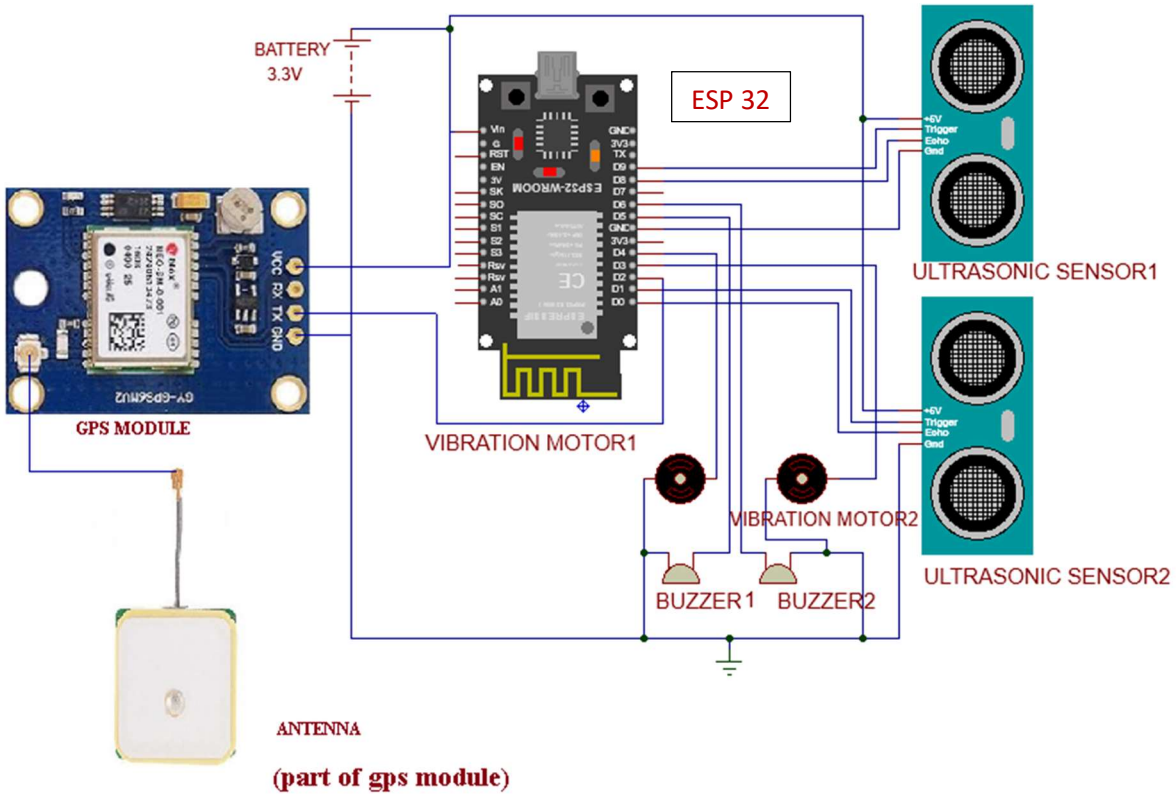

FIGURE 5.1 CIRCUIT DIAGRAM

SOFTWARE DESCRIPTION

CHAPTER 6

SOFTWARE DESCRIPTION

The various software used in the existing system are:

- Arduino IDE.
- Twilio API.

6.1 ARDUINO IDE

Arduino IDE is an open-source software, designed by Arduino.cc and mainly used for writing, compiling & uploading code to almost all Arduino modules. It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process.

6.2 TWILIO API

Twilio API power its platform for communications. Behind these APIs is a software layer connecting and optimizing communications network around the world to allow users to call and message anyone, globally.

It uses REST API (Representational State Transfer) is an architecture pattern that describes how distributed system can expose a consistent interface. In are general ‘REST API’ means an API accessed using the HTTP protocol at a predefined set of URLs.

6.3 MANAGED CODE

Arduino IDE provides a user-friendly interface for managing and developing code for Arduino microcontroller boards. The IDE includes several features and tools to help users write, compile, and upload code to the Arduino board.

To manage code in the Arduino IDE, users typically follow the following steps:

- Open the Arduino IDE and create a new sketch. The sketch is the file that contains the code for the Arduino board.
- Write the code for the Arduino board using the built-in text editor. The IDE includes features such as syntax highlighting and auto completion to make coding easier.
- Verify and compile the code by clicking on the "Verify" button. This checks the code for errors and ensures that it is compatible with the Arduino board.
- Upload the code to the Arduino board by clicking on the "Upload" button. This transfers the code to the board and allows it to be executed.
- Monitor the output of the code by viewing the message console. This displays any debug information or output from the Arduino board.
- Test and refine the code as necessary. The IDE allows users to modify and refine the code as needed, and then repeat the process of verifying, compiling, and uploading the code.

6.4 MANAGED DATA

Arduino IDE provides several ways to manage data when programming and developing code for Arduino microcontroller boards. Here are some ways to manage data in the Arduino IDE:

- Variables: Arduino allows users to declare and use variables, which are used to store data such as numbers, text, or Boolean values. Variables can be declared as global or local, depending on their scope.
- Arrays: Arrays are used to store multiple values of the same type.
- Constants: Constants are values that cannot be changed during runtime. They are typically used to store data such as pin numbers or other configuration settings.
- Serial communication: Arduino supports serial communication, which allows users to send and receive data between the microcontroller board and a computer or other device. This is often used for data logging or to interface with external sensors.
- Libraries: Arduino includes a library of pre-written code that can be used to manage data, such as the Wire library for I2C communication or the SD library for reading and writing data to an SD card.

6.5 COMMON LANGUAGE SPECIFICATION

The Common Language Specification (CLS) is a set of rules and guidelines for programming languages, including the Arduino programming language. It is designed to ensure that code written in different programming languages can be easily used and interchanged.

6.6 ARCHITECTURE OF ARDUINO IDE

The architecture of the Arduino IDE (Integrated Development Environment) can be divided into two main components: the front-end and the back-end.

6.6.1 FRONT-END

The front-end of the Arduino IDE is responsible for providing the user interface and editing capabilities. It is built using the Java programming language and relies on the Swing toolkit for its graphical user interface (GUI). The front-end also manages the compilation and upload process, as well as the communication with the Arduino board.

6.6.2 BACK-END

The back-end of the Arduino IDE is responsible for compiling the user's code into machine-readable instructions that can be uploaded to the Arduino board. It consists of a number of components, including the Arduino core libraries, the AVR toolchain (which includes the compiler, linker, and other tools), and a set of protocols for communicating with the board.

SYSTEM TESTING

CHAPTER 7

SYSTEM TESTING

The purpose of testing is to discover errors in testing and discovering conceivable fault or weakness in a working product if provides a way to check the functionality of components, subassemblies or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of testing each test type addresses a specific testing requirement.

7.1 TYPES OF TESTING

- Unit Testing
- Integration Testing
- Performance Testing
- Security Testing

7.1.1 UNIT TESTING

Unit testing is an important part of ensuring that the individual learning system for malware detection correctly and producing the expected outputs. By testing each component in isolation. Developers can identify and address issues early on, before them larger problems in the overall systems.

7.1.2 INTERGRATION TESTING

Integration testing is an important part of ensuring that the different components of the machine learning systems for malware detection are working together as expected.

7.1.3 PERFORMANCE TESTING

- Testing the system's processing speed.
- Testing the system's accuracy.
- Testing the system's scalability.
- Testing the system's resource usage.

7.1.4 SECURITY TESTING

Security testing is an important part of ensuring that the machine learning system for malware detection is secure and that it is protecting user data and system resources from different types of attacks. By testing the system's resilience to different types of attacks and its data protection mechanisms. Developers can identify and address any security vulnerabilities early on, before they become larger problems in the overall system.

PROJECT IMPLEMENTATION

CHAPTER 8

PROJECT IMPLEMENTATION

Project implementation involves putting plans into action to achieve the desired goals and objectives outlines in the project plan. The following are the steps implemented in our project.

- Circuit Design and Assembly
- Programming the Microcontroller
- Integration with Twilio API
- Deployment and Monitoring

8.1 CIRCUIT DESIGN AND ASSEMBLY

Design the circuit layout based on the selected components. Connect the microcontroller to the ultrasonic sensor. Ensure proper wiring and connections to avoid any short circuits or malfunctions.

8.2 PROGRAMMING THE MICROCONTROLLER

Write the firmware code for the microcontroller. Implement algorithms to read data from the ultrasonic sensor. Program the microcontroller to control the buzzer, vibration motor and GPS module.

8.3 INTEGRATION WITH TWILIO API

Once the circuit is designed, integrate the GPS module with the Twilio API. So that the last module "**Sending SOS message in case of emergency**" can be implemented.

8.4 DEPLOYMENT AND MONITORING

Deployment refers to the process of making the hardware and software available for use. It involves transferring the application from a development environment to a production environment where the end user can access it. Monitoring on the other hand involves continuously observing the performance, availability and behavior of an application.

SYSTEM MAINTENANCE

CHAPTER 9

SYSTEM MAINTENANCE

System maintenance involves the regular upkeep, updates, and optimization of hardware, software and other components within the system. It includes tasks such as applying patches, upgrading software and upgrading hardware within the system.

9.1 HARDWARE CHECK

Inspect the I/O devices such as sensors, cameras, or any wearable tech to ensure they are functioning correctly.

9.2 SOFTWARE CHECK

Check for any software updates for the IoT devices and the central system. Install updates to ensure security and performance enhancements.

9.3 DATA MANAGEMENT

Review the data collected by the system to ensure it’s accurate and relevant. Clean up any unnecessary data to optimize storage and processing.

9.4 BATTERY MANAGEMENT

The whole system is powered by using a 3.7V lithium ion battery. It should be charged regularly and completely changed when required.

CONCLUSION & FUTURE ENHANCEMENT

CHAPTER 10

CONCLUSION & FUTURE ENHANCEMENT

10.1 CONCLUSION

Our project is successfully removing the problem of existing navigation techniques like carry stick with us while walking, use of another person while moving one place to another and many more issue was successfully resolved by this project. The project has the feature to detect the distance of objects that are major issue for blind people after detecting the object distance they also know about the direction where the object was.

10.2 FUTURE ENHANCEMENT

The project can be improved as follow in the future:

- 180 degrees coverage.
- Sending SMS to the target phone number.
- Different tone for buzzer and different vibration intensity for vibration motor for detecting object at different direction.

10.2.1 180 DEGREE COVERAGE

By using servo motor by we can cover 360 degrees. A servo motor is a DC motor can rotate about 180 degrees.

10.2.2 SENDING SMS TO THE TARGET PHONE NUMBER

By enhancing the functionality of Twilio API, we can send SMS along with the WhatsApp app message. This functionality ensures that the message is received by the target without any delay.

10.2.3 DIFFERENT INTENSITY SIGNAL FOR BUZZER AND VIBRATION MOTOR.

By using active buzzers, different frequency sounds can be emitted. Also vibration module can be vibrated in different pulse. Different sounds and pulse can be used by the blind to detect the objects in different direction.

APPENDIX

CHAPTER 11

APPENDIX

11.1 SOURCE CODE

```
#define trigPin1 5
#define echoPin16
#define motor17
#define buzzer18
#define trigPin2 9
#define echoPin2 10
#define motor2 11
#define buzzer2 12
#include <TinyGPS++.h>
#define SERVO_PIN 14
long duration, distance, UltraSensor1, UltraSensor2;
char data;
String SerialData="";
void setup()
{
Serial.begin (9600);
pinMode(trigPin1, OUTPUT);
pinMode(echoPin1, INPUT);
pinMode(motor1, OUTPUT);
```

```
pinMode(buzzer1, OUTPUT);
pinMode(trigPin2, OUTPUT);
pinMode(echoPin2, INPUT);
pinMode(motor2, OUTPUT);
pinMode(buzzer2, OUTPUT);
digitalWrite(motor1,LOW);
digitalWrite(motor2,LOW);
digitalWrite(buzzer1,LOW);
digitalWrite(buzzer2,LOW);
}
void loop()
{
SonarSensor(trigPin1, echoPin1);
UltraSensor1 = distance;
SonarSensor(trigPin2,echoPin2);
UltraSensor2 = distance;
while(Serial.available())
{
 delay(10);
 data=Serial.read();
 SerialData+=data;
}
```

```
  if(SerialData=="display distance")
  {
  Serial.print("distance measured by the first sensor: ");
  Serial.print(UltraSensor1);
  Serial.println(" cm");
  Serial.print("distance measured by the second sensor: ");
  Serial.print(UltraSensor2);
  Serial.println(" cm");
  Serial.println("--------------------------------------------------------------------------
----------------------------");
  }
  SerialData="";
  if(UltraSensor1 <=70)
  {
   digitalWrite(buzzer1,HIGH);
  digitalWrite(motor1,HIGH);
  }
  else
  {
   digitalWrite(buzzer1,LOW);
   digitalWrite(motor1,LOW);
  }
```

```
else
{
 digitalWrite(buzzer2,LOW);
 digitalWrite(motor2,LOW);
}
}
void SonarSensor(int trigPinSensor,int echoPinSensor)
{
digitalWrite(trigPinSensor, LOW);
delayMicroseconds(2);
digitalWrite(trigPinSensor, HIGH);
delayMicroseconds(10);
digitalWrite(trigPinSensor, LOW);
duration = pulseIn(echoPinSensor, HIGH);
distance= (duration/2) / 29.1;
 }
Servo servo;
Adafruit_ADXL345_Unified accel = Adafruit_ADXL345_Unified(12345);
TinyGPSPlus gps;

#define WIFI_SSID "VK014"
#define WIFI_PASSWORD "vkengg014"
```

```
#define BOT_TOKEN
#define CHAT_ID "2040191695"
#define RELAY1_PIN 12
#define RELAY2_PIN 13
const unsigned long BOT_MTBS = 10;
X509List cert(TELEGRAM_CERTIFICATE_ROOT);
unsigned long bot_lasttime;
String googleMapsLink;
void  handleNewMessages(int numNewMessages)
{
  Serial.println("handleNewMessages");
  Serial.println(String(numNewMessages));
  for (int i = 0; i < numNewMessages; i++)
  {
    String chat_id = bot.messages[i].chat_id;
    String text = bot.messages[i].text;
    String from_name = bot.messages[i].from_name;
    if (from_name == "")
      from_name = "Guest";

    if (text == "/open")
    {
```

```
      bot.sendChatAction(chat_id, "typing");
      delay(4000);
      bot.sendMessage(chat_id, "Emergency System Activated:/off");
      digitalWrite(RELAY1_PIN, HIGH);
      servo.write(180);
      digitalWrite(RELAY2_PIN, HIGH);
    }
    if (text == "/off")
    {
      digitalWrite(RELAY1_PIN, LOW);
      digitalWrite(RELAY2_PIN, LOW);
      servo.write(0);
      bot.sendMessage(chat_id, "off");
    }
  }
}
void setup() {
  Serial.begin(9600);
  while (!Serial) {
    delay(10);
  }
  Serial.println("ADXL345 test!");
```

```
if (!accel.begin()) {
  Serial.println("Could not find a valid ADXL345 sensor, check wiring!");
  while (1);
}
Serial.println();
Serial.print("Connecting to Wifi SSID ");
Serial.print(WIFI_SSID);
WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
secured_client.setTrustAnchors(&cert);
while (WiFi.status() != WL_CONNECTED)
{
  Serial.print(".");
  delay(500);
}
Serial.print("\nWiFi connected. IP address: ");
Serial.println(WiFi.localIP());
Serial.print("Retrieving time: ");
configTime(0, 0, "pool.ntp.org");
time_t now = time(nullptr);
while (now < 24 * 3600)
{
  Serial.print(".");
```

```
    delay(100);
    now = time(nullptr);
  }
  Serial.println(now);
  bot.sendMessage(CHAT_ID, "Bot started up");
  pinMode(RELAY1_PIN, OUTPUT);
  pinMode(RELAY2_PIN, OUTPUT);
  digitalWrite(RELAY1_PIN, LOW);
  digitalWrite(RELAY2_PIN, LOW);
  servo.attach(SERVO_PIN);
  servo.write(0);
}
void loop() {
  location(); // Call the location function in the loop
  accident();
  if (millis() - bot_lasttime > BOT_MTBS)
  {
    int numNewMessages = bot.getUpdates(bot.last_message_received + 1);
    while (numNewMessages)
    {
      Serial.println("got response");
      handleNewMessages(numNewMessages);
```

```
      numNewMessages = bot.getUpdates(bot.last_message_received + 1);
    }
    bot_lasttime = millis();
  }
}
```

11.2 OUTPUT SCREENSHOTS

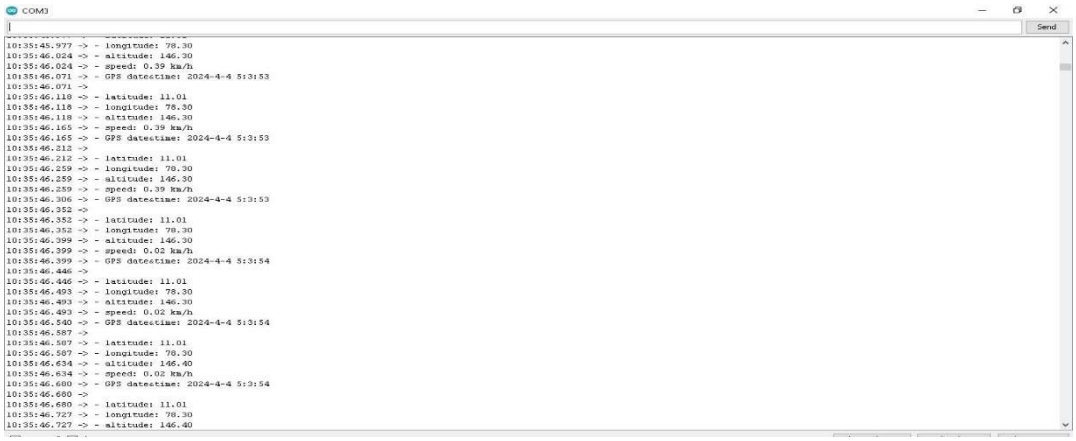

Figure 11.1 Output of GPS module

Figure 11.2 Object detection

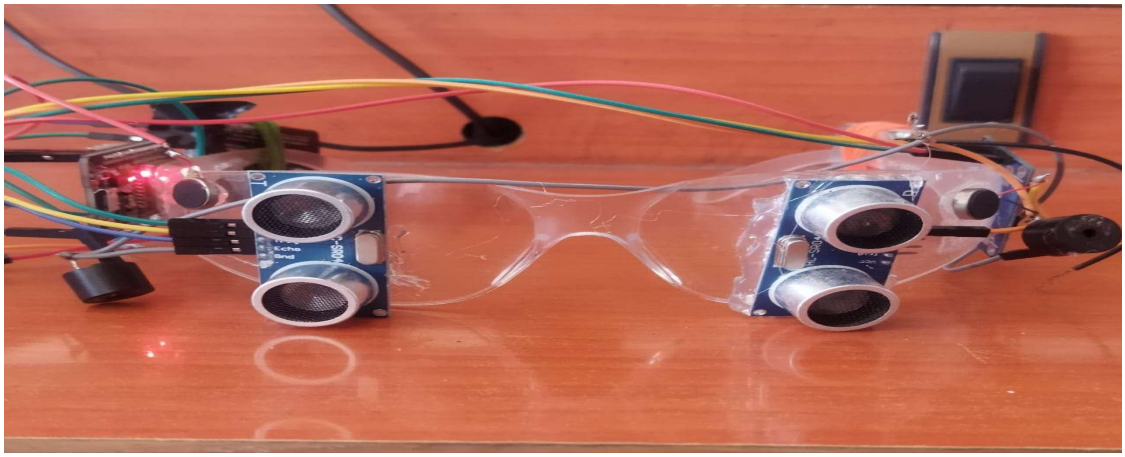

Figure 11.3 Glass with ultrasonic sensor

REFERENCES

CHAPTER 12

REFERENCES

- "**OBJECT RECOGNITION WITH ULTRASONIC SENSOR**" BY M Ihsan, Paolo Gaudiano, 1999.This project describes about detecting about at varying distance.
- "**MULTISENSOR-BASED OBJECT DETECTION IN INDOOR FOR VISUALLY IMPAIRED PEOPLE**" by Charmi T patel, Vaidehi, Laxmi S desai, 2018.This project describes about using multiple sensors within indoor for detection objects.
- "**OBJECT DETECTION USING ULTRASONIC SENSOR**" by G Arun Francis, M Arulselvan, P Elangkumaran, S Keerthivarman, J Vijaya kumar, 2020. This project describes about radar that works using ultrasonic sensor.
- "**SMART ASSISTIVE SYSTEM FOR VISUALLY IMPAIRED PEOPLE**" by Usman Masud, Tareq Saeed, Hunida M, 2022.This project describes about object detection and scenes classification.
- "**THIRD EYE FOR VISUALLY CHALLENGED USING ECHOLOCATION TECHNOLOGY**" by Arvitha M.V.S, Ambresh G. Biradar, Chandana M. This project describes about obstacles detection and water detection.
- "**THIRD EYE BLIND PEOPLE USING ULTRASONIC VIBRATING GLOVES WITH IMAGE PROCESSING**" by Suprabha Putphod, Sneha Kumphar, Prashant Mhargude, Parvia Knikar.
- "**SMART GLOVES FOR BLIND**" by Arsh.A.Ghate, Vishal G.Chavan
- "**EFFECTIVE FAST RESPONSE SMART STICK FOR BLIND PEOPLE**" by Ayat Nada, Samia ashelly, Mahmoud A.Fakhr, Ahmed F.Seddik.

- "**A NOVEL APPROACH OF ASSISTING THE VISUALLY IMPACT NAVIGATE PATH AND AVOIDING OBSTACLE – COLLISIONS"** by Suman Deb, S.Thirupathi Reddy, Ujjwal Baidya.
- "**NAVIGATION SYSTEM FOR BLIND – THIRD EYE"** by Samartha Koharwal, Samer Rani Awad, Aparna Vyakaranam.
- "**THIRD EYE 360º OBJECT DETECTION AND ASSISTANCE FOR VISUALLY IMPAIRED PEOPLE**" by Rohan Jain Yukta, Bhattat Tejal, Patil, Khalid.