ELDERLY FALL DETECTION AND ALERT SYSTEM

A Mini Project Report

Submitted to the APJ Abdul Kalam Technological University

in partial fulfillment of requirements for the award of degree

Bachelor of Technology

in

Electronics and Communication Engineering

by

Athulya S (NSS20EC032)

Deeraj (NSS20EC034)

Gopika G Krishnan (NSS20EC040)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

NSS COLLEGE OF ENGINEERING PALAKKAD

KERALA

August 2023

DEPT. OF ELECTRONICS & COMMUNICATION ENGINEERING

NSS COLLEGE OF ENGINEERING PALAKAKD

2022-23

CERTIFICATE

This is to certify that the report entitled **Elderly Fall Detection and Alert System** submitted by **Athulya S** (NSS20EC032), **Deeraj** (NSS20EC034) & **Gopika G Krishnan** (NSS20EC040) to the APJ Abdul Kalam Technological University in partial fulfillment of the B.Tech. degree in Electronics and Communication Engineering is a bonafide record of the miniproject work carried out by them under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Asst. Prof Saranya R
(Project Guide)
Dept.of ECE
NSS College of Engineering
Palakkad

Prof Vinod G
Head of the Department
Dept.of ECE
NSS College of Engineering
Palakkad

DECLARATION

We hereby declare that the project report **Elderly Fall Detection and Alert System**, submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology of the APJ Abdul Kalam Technological University, Kerala is a bonafide work done by us the under supervision of Asst Prof Saranya R

This submission represents our ideas in our own words and where ideas or words of others have been included, we have adequately and accurately cited and referenced the original sources.

We also declare that we have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. We understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree, diploma or similar title of any other University.

Athulya S

Deeraj

Gopika G Krishnan

Palakkad

04-08-2023

Contents

Abstract

Sudden falls, especially in elders is a recurring problem. According to various health surveys, most of the people die or enter into a permanent vegetative state due to lack of treatment on injuries due to fall. Most injuries in the elderly like bone fractures and dislocation have mainly occured due to unexpected fall in elders.

Presently most of the fall detection systems are non-wearable[3], employing a camera based technology to detect falls. The readings from camera need not be accurate and may deliver erroneous fall detection alerts. Another class of fall detection systems comprise of employing accelerometer only, which only detect plane changes in the spatial coordinates. But the information solely from this sensor can give erroneous results.

Our problem statement focusses on the issues adhering to falls in elders and we have tried to present an efficient detection cum alert system using multiple sensors and GSM, GPS module. By this solution, our objective is to accurately determine falls of a person using a sensor-based system and timely alert close relatives and the medical assistance unit. One of the sensors that we use is a basic 3-Axis accelerometer and the add-on that we have employed for fall detection is a myoware sensor.

Acknowledgement

We take this opportunity to express my deepest sense of gratitude and sincere thanks to everyone who helped us to complete this work successfully. We express our sincere thanks to Prof Vinod G, Head of the Department, Department of Electronics and Communication Engineering, NSS College of Engineering, Palakkad for providing us with all the necessary facilities and support.

We would like to express our sincere gratitude to Prof Saranya R, Assistant Professor, Department of Electronics and Communication Engineering, NSS College of Engineering, Palakkad for the constant support, mentorship and co-operation.

Last, but not the least, we take pleasant privilege in expressing our heartful thanks to our teachers, friends and all well-wishers who were of precious help in completing the miniproject.

Athulya S

Deeraj

Gopika G Krishnan

List of Figures

List of Abbreviations

1. EMG. . . . . Electro-myogram
2. GSM. . . . . Global System for Mobile Communication
3. GPS. . . . . Global Positioning System
4. ADC. . . . . Analog to Digital Converter
5. DC. . . . . Direct Current
6. ML. . . . . Machine Learning
7. IC. . . . . Integrated Circuit

Chapter 1

Introduction

The miniproject mainly has 2 parts, a *fall detection system* and *an alert system*. The fall detection system uses an accelerometer and an EMG sensor to detect the fall. Additionally, we have added a simple obstacle detector using Ultrasonic Proximity Sensor for assistance during mobility of a person. An alert system is also included, which consists of a GSM and GPS module. The GSM module uses *GSM* standard, which is widely used in digital cellular communication. We have added a GPS module to track the location of the person falling and integrated it to the GSM module, so that an alert message can be sent along with the user's location.
At first, when a fall occurs, there will be an obvious change in the 3- dimensional plane of the person. While falling, a person tends to be hypertensive, which reflects on the person's body as muscle contractions. The accelerometer by a capacitive measurement detects the spatial orientation of the person whereas the EMG sensor detects the muscle contractions using a combination of 3 lead wires attached to a convenient spot in the person's body. This is sent to a microcontoller which processes the parameters from the sensors and gives an alert if required through the GSM-GPS module. Elders have poor eyesight, so they tend to tremble over objects infront of them or in their vicinity. Hence by using a proximity sesnor, we have built a simple obstacle detector which triggers a buzzer whenever an obstacle infornt is detected. The buzzer is controllable by the user, since an obstacle detected may be a human or any useful stuff. Pressing a button mounted on the system can stop the sound from the buzzer.

The approach that we are following to implement the project are as follows:-

- Literature Survey: Review existing research and literature on elderly fall detection[4]
- Data collection: Collect fall-related data using sensors and other sources
- Data Analysis: Analyse useful data from the sensors to develop an ideal algorithm for the proposed idea
- Fall Detection Algorithm Development: Develop an algorithm to detect falls and simultaneously to generate alert
- Validation and Testing: Validate the fall detection algorithm using models
- Deployment: Integrate the fall detection system into a wearable device
- Evaluate the effectiveness of the system and refine algorithm and sensors if needed

Chapter 2

Literature Review

For the smooth completion of the miniproject, we have referred to various technical articles, research papers[1], [3], [4] and blogs for a thorough understanding of the concerning issue and its related solutions.
According to a research conducted by Harold et al.[2], the mortality rate due to aging and falling is increasing at an exponential rate. The number of falls is increasing twice per year and is expected to increase in the coming years. Also, it is estimated that about 60% of the deaths in elderly in the upcoming years will be due to sudden, unexpected falls and long-term injuries due to these falls.

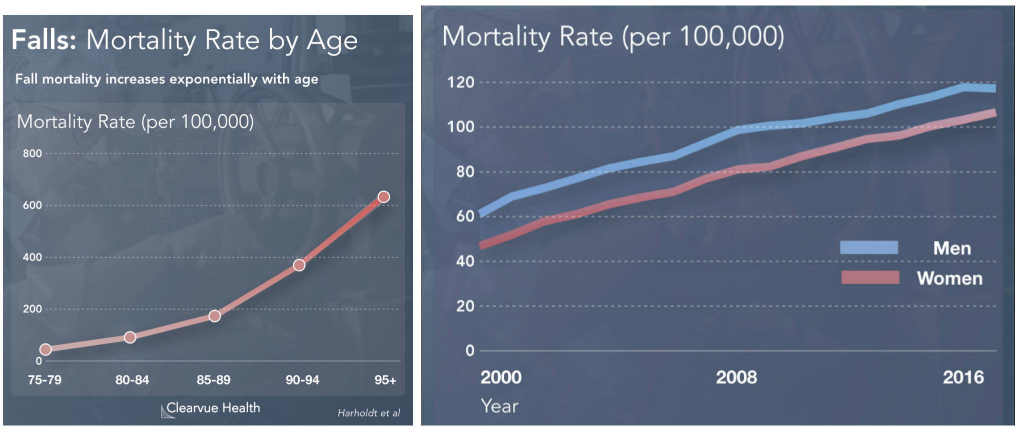

Figure 2.1: Fall Mortality Rate by age and year, Source: *Clearvue Health*

With the help of the following research papers, a brief idea of the existing solutions implemented for the foresaid problem statement are as follows:-
1).*Fall Detection System using IoT and Big Data - Diana Yacchirema et al.*
The paper presents a non-wearable system using Internet of Things integrated with

Big Data Analysis. It helped to formulate the fall detection algorithm to be employed for our solution.

2).*Fall Detection for Elderly People using Machine Learning - Sejal Badgujar et al.*

Presents a non-wearable solution for elderly falls using machine learning technologies. This paper helped us to streamline our algorithm further.

3).*Review of Fall Detection Techniques: A Data Availability Perspective - Shehroz S Khan et al.*

The paper helped us in providing information about the different aspects of fall measurements and what all sensor-based fall detection techniques that we can rely on.

For the implementation of messaging system using GPS and GSM module, we have referred to the following blog, *Accident alert system by ahmadlogs*

Chapter 3

System Development

3.1 Fall Detection Algorithm

The 3 Axis accelerometer works on the principle of capacitance changes and send out the corresponding acceleration values in a 3-Dimensional plane. The square root of sum of squares of the acceleration values are taken into consideration for the initial part of fall detection, given by the formula:-

$$Accelerometer\ Threshold = \sqrt{a_x{}^2 + a_y{}^2 + a_z{}^2}$$

This formula serves as the basis for one part of the fall detection. A threshold value of acceleration, corresponding to a strenous fall is taken into account. The other threshold is determined by the EMG Sensor, which gives a voltage value of approximately 0.5V or below for relaxations and above 0.5V for muscle contractions. Whenever these two thresholds are crossed, the fall detection buzzer beeps as well as a prompt is sent to the GSM module for timely passage of messages as required. A conditional statement has been employed to activate the buzzer and alert system.

3.2 Obstacle Detection Algorithm

After a lot of trials, we concluded with a distance of 20cm approximately to detect obstacles infront of the user. This is to be detected by an Ultrasonic sensor, since it has a wide range for detecting objects infront of the sensor.

3.3 Block Diagram

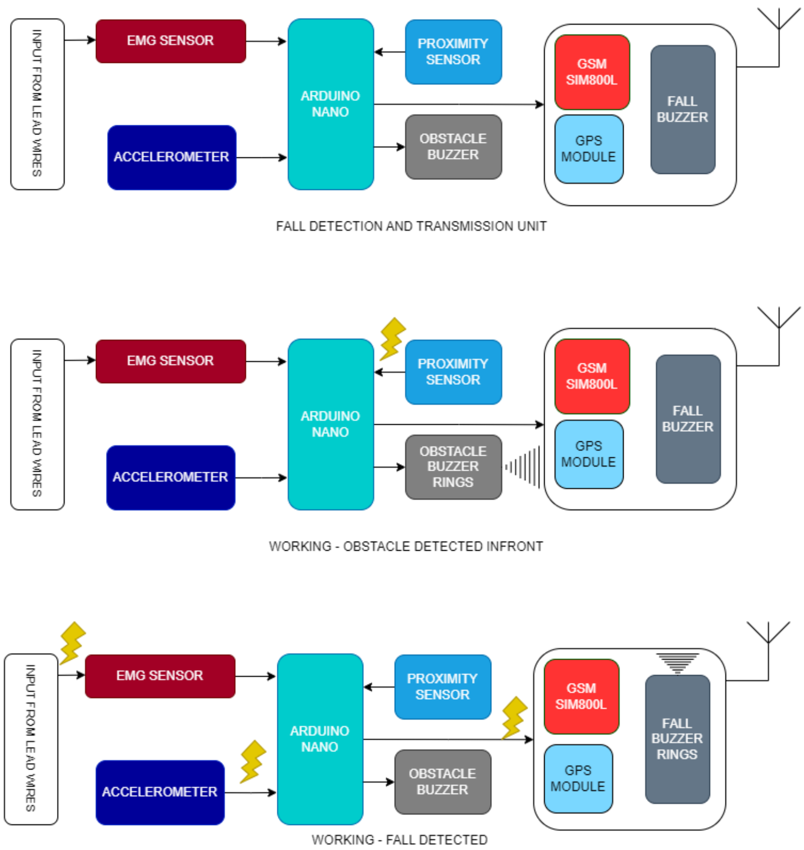

Figure 3.1: Block Diagram Of the Proposed Idea

When an object is in front, few elders may not know its presence and suddenly fall. This can be prevented by using a proximity sensor.

Steps for obstacle detection:-

1).The sensor detects an object in front

2).The sensor input is fed to the microcontroller.

3).The input is processed in the microcontoller and gives output correspondingly

4). The output is connected to a buzzer system, which beeps whenever an object is in front

When a person suddenly falls, to know the extremity of the fall, we use two sensors; an accelerometer and an EMG muscle sensor. The alert is given using GSM Module

Steps for fall detection and alert:-

1).The accelerometer detects the orientation change of the person during a fall and the EMG sensor indicates muscle contraction of the person

2).The sensor inputs are fed to the microcontroller

3).According to the algorithm developed a fall is assessed and indication is sent to the digital pin of the microcontroller

4). Close ones near the elder is notified by the beep of buzzer, an alert message with the location

3.4 Circuit Diagram

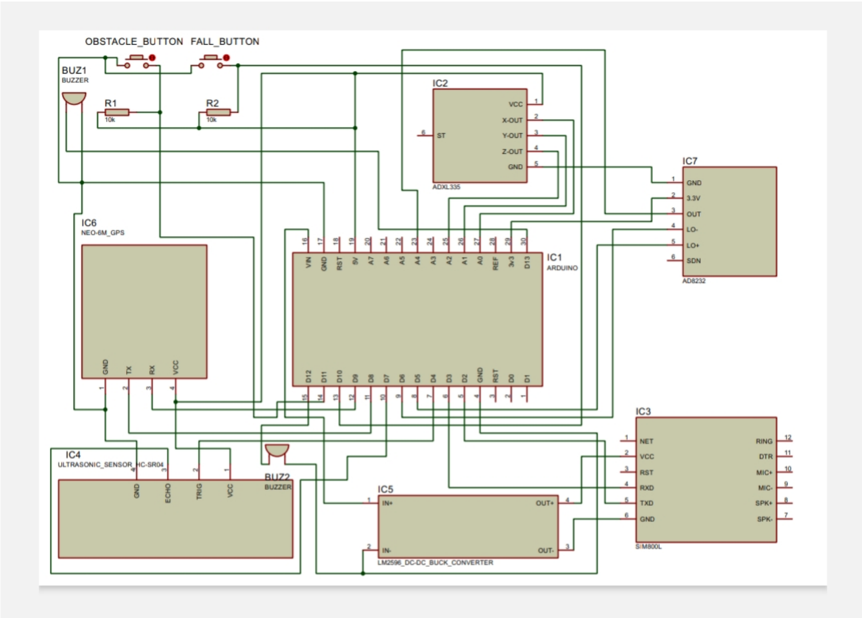

Figure 3.2: Schematic of the Circuit

Arduino Nano controls the whole system. The whole system gets a power supply through and adapter while the GSM Module gets the power supply from 12V adapter through a buck boost converter. This is to ensure the working of GSM Module. GSM Module is connected to the GPS Module and the fall buzzer and the button. The ultrasonic sensor and the obstacle buzzer, button is connected with it, which gets the supply from Arduino's 5V supply pin. The accelerometer and the EMG sensor is connected via the analog pins of Nano board.

Chapter 4

Components Used

4.1 ADXL335 Accelerometer

The ADXL335 is a three-axis accelerometer sensor that is widely used for measuring acceleration in different applications. It is known for its small size, low power consumption, and analog output signals corresponding to acceleration along the X, Y, and Z axes. Manufactured by Analog Devices, this sensor utilizes micro-electromechanical system (MEMS) technology, which integrates mechanical and electrical components on a single chip.

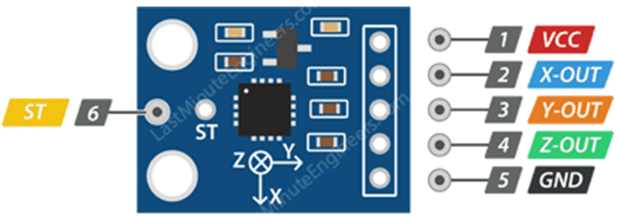

Figure 4.1: Pin Diagram of ADXL335

At its core, the ADXL335 incorporates a tiny, flexible cantilever beam with a mass at one end and capacitive plates on either side. When subjected to acceleration, the mass causes the beam to bend, resulting in a change in capacitance between the plates. This alteration is then converted into electrical signals, providing relevant data to external circuitry for interpretation and processing.

Due to its compactness and energy efficiency, the ADXL335 is well-suited for various applications that require three-dimensional movement and orientation

monitoring. It is commonly used in robotics, motion detection, gesture recognition, tilt sensing, vibration monitoring, and other areas where accurate acceleration measurement is essential.
To make use of the ADXL335 data, it is typically interfaced with external components such as microcontrollers or analog-to-digital converters (ADCs). These external devices read the analog signals from the accelerometer, enabling further processing and analysis for a wide range of practical applications.

4.2 AD8232-Instrumentational Amplifier Module

The AD8232 stands out as a highly integrated signal conditioning chip that has been expertly designed to cater specifically to biopotential measurements. With Analog Devices at the helm of its manufacturing, this chip offers a user-friendly and efficient solution for capturing and amplifying electrical signals generated by different muscles.
Biopotential measurements, particularly electromyography (EMG), play a critical role in medical and research fields. They provide valuable insights into muscle activity, helping healthcare professionals diagnose neuromuscular disorders, evaluate rehabilitation progress, and enable researchers to delve deeper into human movement and physiology.

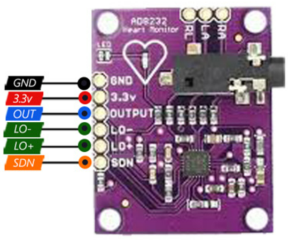

Figure 4.2: Pin Diagram of AD8232

The AD8232 chip incorporates three key functions that are instrumental in optimizing the biopotential measurement process:
1. **Instrumentation Amplification:** At the core of the AD8232, an instrumentation amplifier provides the crucial task of amplifying weak biopotential signals. These signals are inherently small, and proper amplification is essential to ensure they are

robust enough for further processing and analysis. The high-quality amplification offered by this chip contributes to obtaining reliable and accurate measurements.
2. **Filtering:** Biopotential signals are often accompanied by unwanted noise and interference from various sources. The AD8232 addresses this issue with its built-in filtering capabilities. By employing carefully designed filters, the chip effectively removes noise, artifacts, and other unwanted signals, resulting in a cleaner and more accurate representation of the actual biopotential data.
3. **Right-Leg Drive Circuit (RLD):** The RLD circuit is a distinctive feature of the AD8232 that plays a vital role in reducing common-mode noise. Common-mode noise arises from environmental factors and electrical interference that can distort the biopotential measurements. With the RLD circuit's active cancellation technique, the chip significantly mitigates such noise, leading to more reliable and precise biopotential signals.

Thanks to its compact form factor and low-power consumption, the AD8232 can be seamlessly integrated into various medical devices, wearable health monitors, and research equipment. Its versatility allows for non-invasive monitoring of muscle activity, empowering patients and researchers alike.

Hence, the AD8232 serves as an invaluable tool in the field of biopotential measurements, offering an all-in-one solution that simplifies the acquisition of accurate and meaningful data. By harnessing the capabilities of this integrated chip, medical professionals can make better-informed decisions, and researchers can push the boundaries of their understanding of human physiology and movement.

4.3 Ultrasonic Sensor - HCSR04

The HC-SR04 is an ultrasonic distance measuring sensor module commonly used in electronics projects and robotics. It utilizes ultrasonic sound waves to calculate the distance between the sensor and nearby objects. The module consists of an ultrasonic transmitter and receiver pair, which work in tandem to enable distance measurement based on the time it takes for the sound waves to travel to the object and back. It is a popular choice for hobbyists, students, and makers due to its simplicity, affordability, and ease of use.

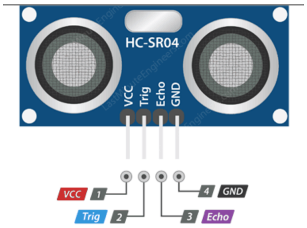

Figure 4.3: Pin Diagram of HC-SR04

To operate the HC-SR04, a trigger signal is sent to the transmitter, causing it to emit a burst of ultrasonic pulses. These pulses travel through the air until they encounter an object, at which point they get reflected back towards the receiver. The receiver then captures the echoed sound waves, and by measuring the time taken for the round-trip, the sensor can calculate the distance using the time-of-flight principle. The HC-SR04 provides distance measurements with relatively high accuracy and can be interfaced with microcontrollers and other electronic devices to integrate distance sensing capabilities into various projects.

One of the main advantages of the HC-SR04 is its versatility, making it suitable for a wide range of applications. It is commonly used in obstacle detection and avoidance systems for robots and autonomous vehicles, water level monitoring, presence detection in home automation projects, and even as a simple rangefinder in DIY electronics. The sensor's ability to function in both indoor and outdoor environments and its compatibility with various microcontrollers, such as Arduino and Raspberry Pi, make it a go-to choice for makers and hobbyists looking to incorporate distance sensing into their projects without the need for complex setups.

4.4 Arduino Nano

The Arduino Nano is a compact and versatile microcontroller board based on the ATmega328P microcontroller chip. It is a member of the Arduino family and shares many features with its larger sibling, the Arduino Uno. However, the Nano is significantly smaller in size, making it an excellent choice for projects with limited space requirements. Despite its small form factor, the Nano packs a punch with its 32KB flash memory, 2KB RAM, and 1KB EEPROM, providing sufficient resources for a wide range of applications.

The Arduino Nano board includes a variety of digital and analog input/output pins, enabling users to connect and control various sensors, actuators, and other electronic components. Additionally, it features a mini-USB port for easy programming and power supply, as well as a barrel jack for external power sources. This flexibility makes it suitable for both standalone projects and those requiring communication with a computer or other devices.

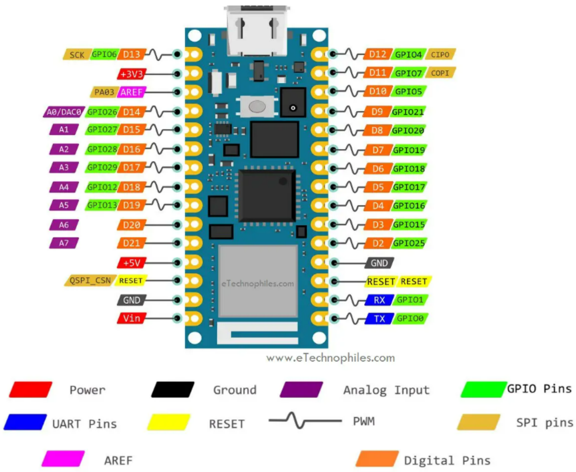

Figure 4.4: Pin Diagram of Arduino Nano

The Nano's popularity is due in part to its compatibility with the Arduino development environment, which provides an easy-to-use and beginner-friendly platform for programming and prototyping. Its small size, ample capabilities, and cost-effectiveness make it a favorite among hobbyists, students, and electronics enthusiasts, facilitating the rapid development of innovative and interactive projects. Whether it's used for home automation, robotics, wearable devices, or even educational purposes, the Arduino Nano continues to be a go-to choice for a wide range of electronic projects.

4.5 Buck Converter

The LM2596 is a popular and widely used step-down voltage regulator integrated circuit (IC) that provides efficient and reliable DC-DC voltage conversion. Manufactured by Texas Instruments, the LM2596 is a versatile device capable of converting higher input voltages to lower output voltages with excellent efficiency. It is commonly employed in various electronics projects and power supply applications where a stable and regulated voltage is required.

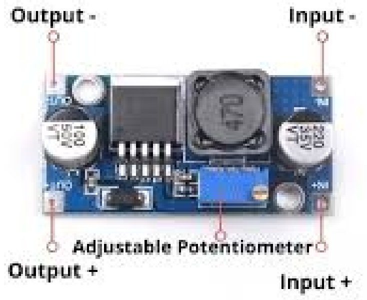

Figure 4.5: Pin Diagram of DC-DC Buck Converter

The LM2596 offers a wide input voltage range and is available in different versions to support various output voltage options. Its architecture incorporates a feedback control system that allows users to adjust the output voltage precisely, making it adaptable to diverse voltage requirements. The IC is equipped with built-in protection features, including thermal shutdown and current limiting, ensuring safe operation and protection against overloading and overheating. With its ease of use, reliability, and cost-effectiveness, the LM2596 has become a go-to choice for designers and hobbyists seeking an efficient and compact DC-DC voltage regulator solution.

4.6 SIM800L GSM Module

SIM800L is a compact and versatile GSM/GPRS module designed for wireless communication in various applications. Manufactured by SIMCom, this module is part of the SIM800 series and is widely used for Internet of Things (IoT) projects, remote monitoring systems, and other applications that require cellular connectivity. The SIM800L module supports quad-band GSM and GPRS communication, making it compatible with a wide range of cellular networks worldwide.
One of the key features of the SIM800L module is its small size, which makes it ideal for projects with limited space requirements. Despite its compact form

Figure 4.6: Pin Diagram of GSM Module

factor, the module offers a rich set of features, including SMS and data transmission capabilities, audio functionality for voice calls, and support for various Internet protocols like HTTP, FTP, and TCP/UDP. The SIM800L can be interfaced with microcontrollers, such as Arduino and Raspberry Pi, allowing developers to easily integrate cellular connectivity into their projects. With its reliable performance, low power consumption, and cost-effectiveness, the SIM800L remains a popular choice for IoT and communication-based projects that require seamless cellular connectivity.

4.7 NEO-6M GPS Module

The NEO-6M GPS module is a compact and high-performance Global Positioning System (GPS) receiver designed for accurate positioning and navigation. Developed by u-blox, the NEO-6M module is equipped with advanced GPS technology that allows it to acquire satellite signals quickly and provide accurate location data.
It operates on a low power consumption, making it suitable for a wide range of applications, including vehicle tracking, geocaching, asset monitoring, and outdoor sports.
The NEO-6M GPS module features multiple communication interfaces, such as UART, I2C, and SPI, enabling seamless integration with various microcontrollers and communication protocols. The module comes with a built-in patch antenna, but it also has an option for an external antenna, allowing users to optimize the GPS performance based on their specific needs. Its compact size and ease of

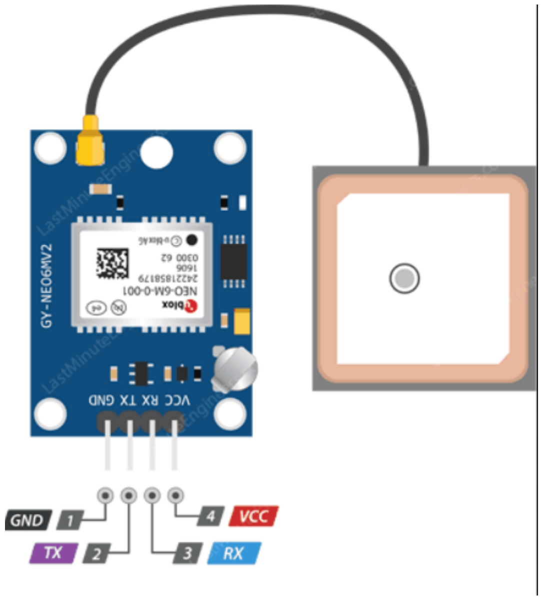

Figure 4.7: Pin Diagram of GPS Module

use make it a popular choice among hobbyists, engineers, and IoT enthusiasts looking to incorporate precise positioning capabilities into their projects without the complexities associated with traditional GPS systems. Whether used in drones, smartwatches, or navigational devices, the NEO-6M GPS module is a reliable and efficient solution for obtaining accurate and real-time location information.

Chapter 5

Inferences

5.1 Hardware Implementation

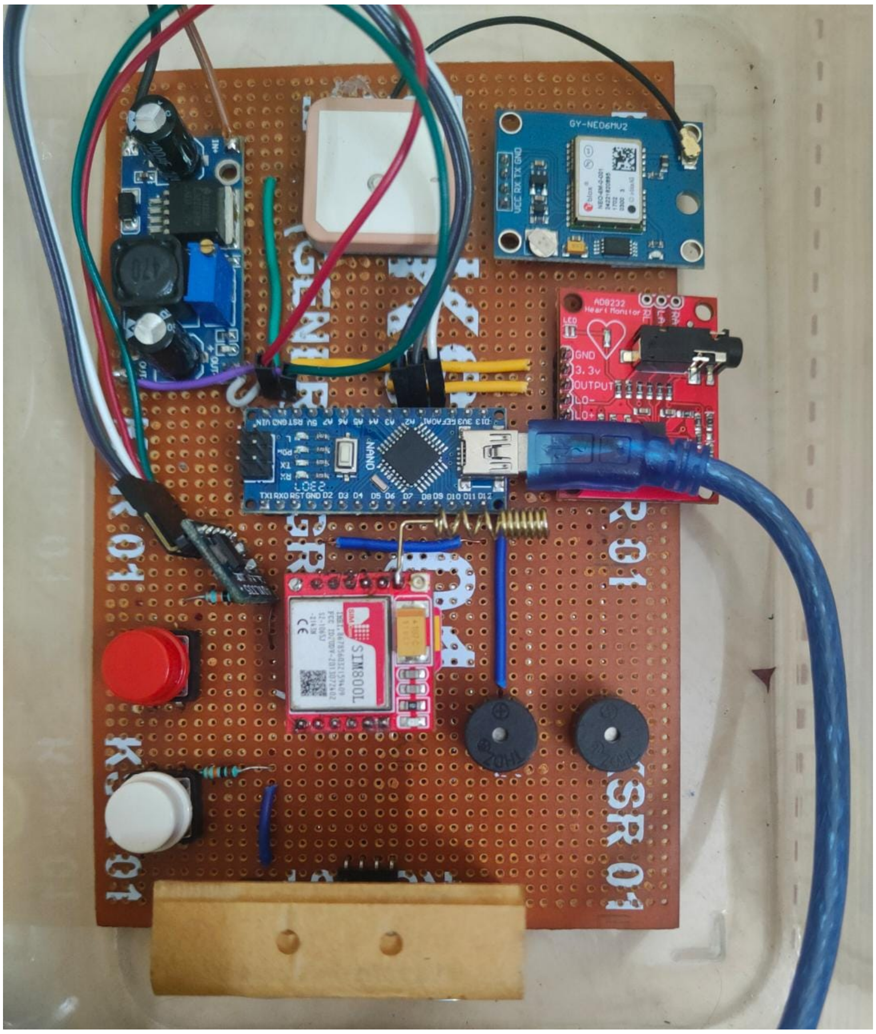

Figure 5.1: Hardware implemented on General Purpose PCB

5.2 Working of the Circuit

The working of the circuit are as follows:-

When a person suddenly falls, to know the extremity of the fall, we use two sensors; an accelerometer and an EMG muscle sensor. The accelerometer works on the principle of capacitance changes and send out the corresponding acceleration values in a 3-Dimensional plane. The square root of sum of squares of the acceleration values are taken into consideration for the initial part of fall detection. By numerous trial and error methods, we concluded with a threshold value(approximately 4.5 Arduino ADC output) and put it up in the code. Then the EMG module value is taken and for contractions the ADC value was found out to be greater than 60 where the relaxation of muscles give a value of 55 or less.

The alert is given using a GSM Module. Initially, the GSM module is set up to find a network and it automatically connects after sdetecting a viable network. The steps involved in fall detection and simultaneous alert is as follows:-

1). The accelerometer detects the orientation change of the person during a fall and the EMG sensor indicates muscle contraction of the person.

2). The EMG sensor inputs are fed to the microcontroller.

3) .According to the algorithm developed a fall is assessed, based on the thresholds and an indication is sent through digital Arduino pin to a buzzer and LED. Along with that, a prompt is sent to the GSM module to send the alert message. The GSM module sends the message along with the location tracked using GPS module. If the detected is a daily activity and not a strenous fall, the user can stop the alert and the beep of the buzzer by pressing the respective button on the system within a time period of 30s, if exceeded it is concluded as a fall.

4). Whenever an obstacle infront is detected, another piezo buzzer beeps. It can also be controlled by a button.

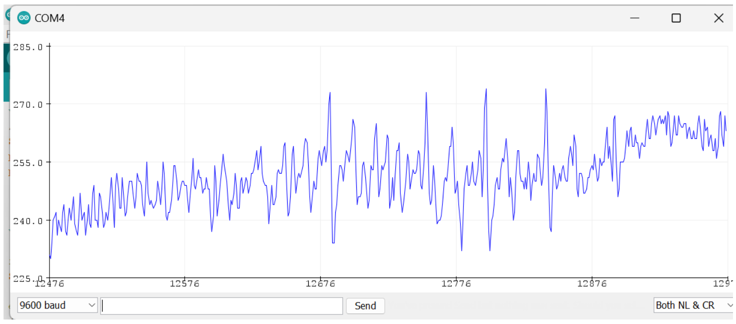

Figure 5.2: EMG Sensor Readings and plots, Spike corresponds to contraction of the muscles

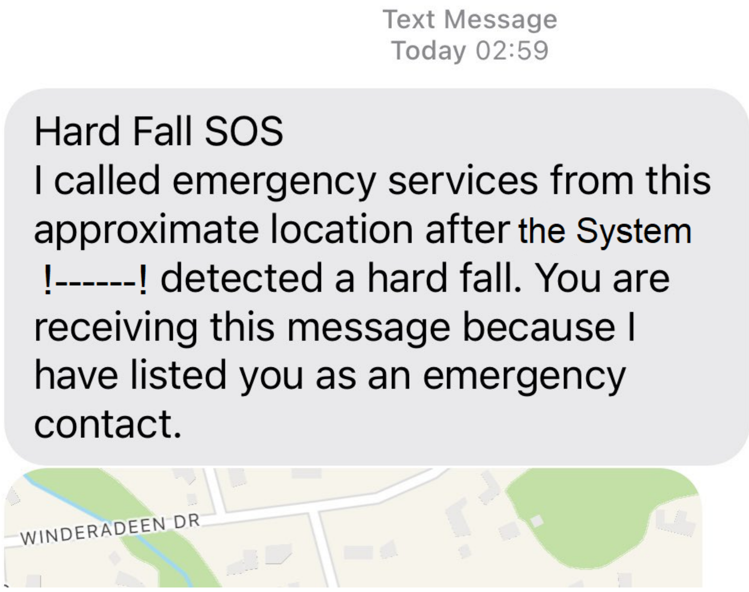

Figure 5.3: Message Sent using GSM alongwith GPS Module

Chapter 6

Conclusion

The fall detection system and alert system that we have implemented focusses on minimizing sudden falls in elders, which inturn gives a productive community. Prolonged effects of sudden falls in elders may lead to serious injuries or in the worst case leads to death. It can also equip the old-aged to be independent and improve their quality of life. The system we presented is hence socially relevant and also economical. It is roughly estimated that the chances of falling will increase in the further years and so, implementing such a system for elders will lead to a fruitful life for the elders. It will also help to improve coordination and increase their strength and become more lively in their daily life. Through this project, we believe that we were able to device a system that conveys the motto ‘Healthy Aging’.

6.1 Future Works

We plan to revolutionize the fall detection system by introducing cutting-edge advancements and innovative features. Our vision is to implement technologies such as Convolutional Neural Networks (CNN) and regression techniques[1], to enable the system to differentate regular daily activities and falls with accuracy. By training the model and continuously updating it with new data, the fall detection system will perform real-time analysis of sensor data, correctly identifying potential falls while minimizing false alarms.

Bibliography

[1] Sejal Badgujar and Anju S Pillai. “Fall detection for elderly people using machine learning”. In: *2020 11th international conference on computing, communication and networking technologies (ICCCNT)*. IEEE. 2020, pp. 1–4.

[2] Klaas A Hartholt et al. “Mortality from falls among US adults aged 75 years or older, 2000-2016”. In: *Jama* 321.21 (2019), pp. 2131–2133.

[3] Shehroz S Khan and Jesse Hoey. “Review of fall detection techniques: A data availability perspective”. In: *Medical engineering & physics* 39 (2017), pp. 12–22.

[4] Diana Yacchirema et al. “Fall detection system for elderly people using IoT and big data”. In: *Procedia computer science* 130 (2018), pp. 603–610.

Appendix A

A.1 Program

Listing A.1: Arduino Code for the proposed idea

```
#include <AltSoftSerial.h>
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
#include <math.h>
#include<Wire.h>

//--------------------------------------------------------------
const String EMERGENCY_PHONE = "+91**********";
//--------------------------------------------------------------
#define rxPin 2
#define txPin 3
SoftwareSerial sim800(rxPin,txPin);
//--------------------------------------------------------------
AltSoftSerial neogps;
TinyGPSPlus gps;
//--------------------------------------------------------------
String sms_status,sender_number,received_date,msg;
String latitude, longitude;
//--------------------------------------------------------------
#define FALL_BUZZER 13
```

```
#define FALL_BUTTON 10
#define OBSTACLE_BUZZER 12
#define OBSTACLE_BUTTON 11
//-------------------------------------------------------------
#define x_out A0
#define y_out A1
#define z_out A2
#define emg_out A4
#define L_plus 5
#define L_minus 6
#define trigPin 4
#define echoPin 7
//-------------------------------------------------------------
byte updateflag;
long duration;
float distance;
int xaxis = 0, yaxis = 0, zaxis = 0;
int deltx = 0, delty = 0, deltz = 0;
int vibration = 2, devibrate = 75;
int magnitude = 0;
int sensitivity = 20;
double angle;
boolean impact_detected = false;
boolean obstacle_detected = false;
const int DISTANCE_THRESHOLD = 50;
unsigned long time1;
unsigned long impact_time;
unsigned long alert_delay = 30000; //30 seconds
//-------------------------------------------------------------
void setup()
{
```

```
//------------------------------------------------------------
Serial.begin(9600);
//------------------------------------------------------------
sim800.begin(9600);
//------------------------------------------------------------
neogps.begin(9600);
//------------------------------------------------------------
pinMode(FALL_BUZZER, OUTPUT);
pinMode(FALL_BUTTON, INPUT_PULLUP);
pinMode(OBSTACLE_BUZZER, OUTPUT);
pinMode(OBSTACLE_BUTTON, INPUT_PULLUP);
pinMode(trigPin, OUTPUT
pinMode(echoPin, INPUT);
pinMode(L_plus, INPUT);
pinMode(L_minus, INPUT);
pinMode(emg_out, OUTPUT);
//------------------------------------------------------------
sms_status = "";
sender_number="";
received_date="";
msg="";
//------------------------------------------------------------
sim800.println("AT");
delay(1000);
sim800.println("ATE1");
delay(1000);
sim800.println("AT+CPIN?");
delay(1000);
sim800.println("AT+CMGF=1");
delay(1000);
sim800.println("AT+CNMI=1,1,0,0,0");
```

```
  delay(1000);
  //------------------------------------------------------------
  time1 = micros();
  //------------------------------------------------------------
  xaxis = analogRead(x_out);
  yaxis = analogRead(y_out);
  zaxis = analogRead(z_out);
  //------------------------------------------------------------
}
void loop()
{
  //------------------------------------------------------------
  if (micros() - time1 > 1999) Impact();
  //------------------------------------------------------------
  if(updateflag > 0)
  {
    updateflag=0;
    Serial.println("Impact detected!!");
    Serial.print("Magnitude:");
    Serial.println(magnitude);

    getGps();
    digitalWrite(FALL_BUZZER, HIGH);
    impact_detected = true;
    impact_time = millis();
  }
  //------------------------------------------------------------
  if(impact_detected == true)
  {
    if(millis() - impact_time >= alert_delay) {
      digitalWrite(FALL_BUZZER, LOW);
```

```
      makeCall();
      delay(1000);
      sendAlert();
      impact_detected = false;
      impact_time = 0;
    }
  }

  if(digitalRead(FALL_BUTTON) == LOW){
    delay(200);
    digitalWrite(FALL_BUZZER, LOW);
    impact_detected = false;
    impact_time = 0;
  }
  //----------------------------------------------------------------
  while(sim800.available()){
    parseData(sim800.readString());
  }
  //----------------------------------------------------------------
  while(Serial.available())  {
    sim800.println(Serial.readString());
  }
  //----------------------------------------------------------------

  //------------Obstacle Detection----------------------------------
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Sets the trigPin on HIGH state for 10 micro seconds
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
```

```
  // Reads the echoPin, returns the sound wave travel time in
microseconds
  duration = pulseIn(echoPin, HIGH);
  // Calculating the distance
  distance = duration * 0.034 / 2;

  if(distance < DISTANCE_THRESHOLD){
    obstacle_detected = true;
  }
  if (obstacle_detected == true){
    digitalWrite(OBSTACLE_BUZZER, HIGH);
    delay(2000);
    obstacle_detected = false;
  }
  else
    digitalWrite(OBSTACLE_BUZZER, LOW);

  if(digitalRead(OBSTACLE_BUTTON) == LOW){
    delay(200);
    digitalWrite(OBSTACLE_BUZZER, LOW);
    obstacle_detected = false;
  }
  Serial.print("Obstacle Distance: ");
  Serial.println(distance);
  Serial.print("\n\n");
  delay(1000);

}
void Impact()
{
  //-----------------------------------------------------------
```

```
time1 = micros();
//-----------------------------------------------------------
int oldx = xaxis;
int oldy = yaxis;
int oldz = zaxis;

xaxis = analogRead(x_out);
yaxis = analogRead(y_out);
zaxis = analogRead(z_out);

//-----------------------------------------------------------
vibration--;
Serial.println(vibration);
if(vibration < 0) vibration = 0;
//Serial.println("Vibration Reset!");

if(vibration > 0) return;
//-----------------------------------------------------------
deltx = xaxis - oldx;
delty = yaxis - oldy;
deltz = zaxis - oldz;
if (magnitude >= sensitivity) //impact detected
{
  updateflag=1;
  // reset anti-vibration counter
  vibration = devibrate;
}
else
{
  //if (magnitude > 15)
    //Serial.println(magnitude);
```

```
      //reset magnitude of impact to 0
      magnitude=0;
    }
  }

  void parseData(String buff){
    Serial.println(buff);

    unsigned int len, index;
    //----------------------------------------------------------------
    index = buff.indexOf("\r");
    buff.remove(0, index+2);
    buff.trim();

    if(buff != "OK"){
      //----------------------------------------------------------------
      index = buff.indexOf(":");
      String cmd = buff.substring(0, index);
      cmd.trim();

      buff.remove(0, index+2);
      //----------------------------------------------------------------
      if(cmd == "+CMTI"){
        index = buff.indexOf(",");
        String temp = buff.substring(index+1, buff.length());
        temp = "AT+CMGR=" + temp + "\r";
        sim800.println(temp);
      }
      //----------------------------------------------------------------
      else if(cmd == "+CMGR"){
        if(buff.indexOf(EMERGENCY_PHONE) > 1){
```

```
          buff.toLowerCase();
          if(buff.indexOf("get gps") > 1){
            getGps();
            String sms_data;
            sms_data = "GPS Location Data\r";
            sms_data += "http://maps.google.com/maps?q=loc:";
            sms_data += latitude + "," + longitude;
            sendSms(sms_data);
          }
        }
      }
      //------------------------------------------------------------
    }
    else{
    //The result of AT Command is "OK"
    }
  }
  void getGps()
  {
    boolean newData = false;
    for (unsigned long start = millis(); millis() - start < 2000;){
      while (neogps.available()){
        if (gps.encode(neogps.read())){
          newData = true;
          break;
        }
      }
    }

    if (newData)
    {
```

```
    latitude = String(gps.location.lat(), 6);
    longitude = String(gps.location.lng(), 6);
    newData = false;
  }
  else {
    Serial.println("No GPS data is available");
    latitude = "";
    longitude = "";
  }

  Serial.print("Latitude= "); Serial.println(latitude);
  Serial.print("Lngitude= "); Serial.println(longitude);
}
void sendAlert()
{
  String sms_data;
  sms_data = "Accident Alert!!\r";
  sms_data += "http://maps.google.com/maps?q=loc:";
  sms_data += latitude + "," + longitude;

  sendSms(sms_data);
}
 void sendSms(String text)
{
  //return;
  sim800.print("AT+CMGF=1\r");
  delay(1000);
  sim800.print("AT+CMGS=\""+EMERGENCY_PHONE+"\"\r");
  delay(1000);
  sim800.print(text);
  delay(100);
```

```
  sim800.write(0x1A);
  delay(1000);
  Serial.println("SMS Sent Successfully.");
}


boolean SendAT(String at_command, String expected_answer,
unsigned int timeout){

    uint8_t x=0;
    boolean answer=0;
    String response;
    unsigned long previous;

    while( sim800.available() > 0) sim800.read();

    sim800.println(at_command);

    x = 0;
    previous = millis();

    do{
        if(sim800.available() != 0){
            response += sim800.read();
            x++;
            if(response.indexOf(expected_answer) > 0){
                answer = 1;
                break;
            }
        }
    }while((answer == 0) && ((millis() - previous) < timeout));
```

```
  Serial.println(response);
  return answer;
}
```