A project report on

IoT BASED SMART WHEELCHAIR FOR FALL DETECTION AND HEALTH MONITORING SYSTEM

Submitted in partial fulfillment of the requirements for the degree of B.Tech
In
Electronics and Computer Science Engineering By

Abhishek Raj Kashyap	**2130004**
Arundhati Lal	**2130013**
Harsh Prakash	**2130177**
Saumya Raj	**2130178**

Under the guidance of
Prof. Sukanta Kumar Sabut

School of Electronics Engineering **KALINGA INSTITUTE OF INDUSTRIAL TECHNOLOGY**
(Deemed to be University) BHUBANESWAR

APRIL 2024

CERTIFICATE

This is to certify that the project report entitled "**IoT based Smart Wheelchair for Fall Detection and Health Monitoring System**" submitted by

Abhishek Raj Kashyap	2130004
Arundhati Lal	2130013
Harsh Prakash	2130177
Saumya Raj	2130178

In partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Electronics and Computer Science Engineering is a bona fide record of the work carried out under our guidance and supervision at School of Electronics Engineering, KIIT (Deemed to be University).

Signature of Supervisor
Prof Sukanta Kumar Sabut
School of Electronics Engineering
KIIT (Deemed to be University)

ACKNOWLEDGEMENT

We are incredibly grateful to our supervisor, Prof. Sukanta Kumar Sabut, whose outstanding leadership has been crucial to the success of our project. His commitment, generosity, and painstaking attention to detail have been an ongoing source of motivation. We are truly grateful to him for his patience and unwavering support, particularly for his thorough evaluation of our work.

Along with our project coordinators, we would also like to express our gratitude to Dr. (Mrs.) Sarita Nanda, Associate Dean and Associate Professor, and Dr. (Mrs.) Suprava Patnaik, Dean and Professor, School of Electronics Engineering, for their invaluable assistance and wise recommendations during the sixth semester project for our undergraduate course.

Roll Number	Name	Signature
2130004	**ABHISHEK RAJ KASHYAP**	
2130013	**ARUNDHATI LAL**	
2130177	**HARSH PRAKASH**	
2130178	**SAUMYA RAJ**	

Date: 06/11/2024

ABSTRACT

One of the main causes of medical issues that older adults and those with impaired mobility deal with is falling. When these people are by themselves, they frequently hurt themselves via falls. In the event of a falling incident, immediate medical assistance is required to lower the risk of serious injuries from falling that could prove fatal. A number of methods have been created that hide their conditioning with webcams. However, the installation and operational costs are expensive and limited to interior spaces. A concern shared by some drug users is sequestration. A wheelchair-person fall detection system using the Internet of Things has been suggested by this investigation. It is dependable and reasonably priced, and it can describe falls and notify users to ask for aid. For fall detection, the suggested system combines a Raspberry Pi Pico W microprocessor, an MPU6050 model, and a Python programming. The device also makes it easier to monitor medical parameters such as oxygen levels and blood pressure. MAX30102 and DS18B20 detectors are among the detectors we use for health monitoring. When a fall occurs, the MPU6050 detects abnormal body motions, which triggers the Pico W to alert the designated person while incident data is kept in a database. All the information will be sent to the Blynk mobile operation in the event of a fall. If there is a fall occurrence and assistance is required, the IoT system will send dispatch alerts to the person on the list to alert them. Additionally, this method offers quick response and a cheaper cost of perpetration. It is installable in wheelchairs that are sold. However, this system's drawback is that it needs a strong WiFi connection. Better GUI design and GPS position enforcement are two ways to improve the system's unborn recommendation; more discovery methods can be incorporated to further enhance the system's delicacy.

TABLE OF CONTENTS

Acknowledgement

Abstract

1. **Introduction**... **7**
   1.1 Motivation...7
   1.2 Literature Review.. 8
   1.3 Objective..9

2. **Methodology**...**10**
   2.1 Tools Used...10
   2.2 Proposed Model... 14
   2.3 Working.. 15

3. **Challenges and Trade-off**.. **17**
   3.1 Challenges and Remedy.. 17
   3.2 Alternative Trade-off.. 17

4. **Design Approach and Algorithm**....................................... **18**
   4.1 Design Flowchart... 18
   4.2 Individual Code.. 20

5. **Result Observation**..**27**
   5.1 Result Analysis.. 27
   5.2 Discussion...28

6. Conclusive Remarks... **29**
6.1 Conclusion...29
6.2 Future Scope.. 29

References...**30**
Appendix A: Gantt Chart... **32**
Appendix B: Project Summary... **33**

CHAPTER 1

Introduction

Countries worldwide are focused on advancing their nations, with a key aspect being the growth of their populations. As populations increase, particularly with a rise in older individuals and those facing health issues, there is a corresponding surge in the demand for healthcare services. The elderly and individuals with health issues often lack the physical strength to walk independently, relying on wheelchairs for mobility. However, those using wheelchairs while living independently face an elevated risk of falls, which can result in emotional and physical harm, potentially leading to acute injuries and also death without prompt medical attention.

To mitigate the risks associated with falls, immediate medical attention is essential. Therefore, a dependable fall detection system becomes crucial for identifying incidents in elderly and physically challenged individuals, prompting swift assistance. Presently, there are two main types of wheelchairs: commercial and smart or powered wheelchairs. Commercial wheelchairs, widely used globally, lack integrated technology. On the other hand, smart or powered wheelchairs feature a controller unit, enabling users to provide input through methods such as joystick control or voice commands, allowing automated navigation. However, these smart wheelchairs are less prevalent in everyday households or medical facilities due to their higher costs. Consequently, there is a need for a cost-effective fall detection system designed for implementation in the commonly used commercial wheelchairs.

Individuals who are elderly or living with disabilities heavily rely on wheelchairs for daily mobility. They face a heightened risk of falls, and the consequences can be severe. Quick response and rescue times are imperative in situations where falls occur, particularly if the affected individuals are living alone or their families are not nearby.

Motivation-

Our primary motivation is to enhance the quality of life for individuals with mobility challenges. Creating a smart wheelchair with fall detection capabilities can provide users with increased independence and confidence in their daily activities. Falls among elderly or physically impaired individuals can lead to serious injuries and complications. By developing a wheelchair with fall detection and monitoring features, we are directly addressing a critical issue in healthcare and contributing to the well-being of the target user group. Developing an IoT-based solution involves the integration of various technologies such as sensors, connectivity, and data analytics. The project will provide an opportunity to explore and apply cutting-edge technologies in a meaningful and impactful way.

Literature Review-

[1] The smart wheelchair has various features, including automatic navigation and object detection. However, it lacks an integrated fall detection system. Existing commercial fall detection systems typically rely on wearables like pendants or wristbands, requiring user intervention to signal a fall event. Unfortunately, these systems do not possess the capability to autonomously seek assistance. Hence, there is a demand for a cost-effective fall detection device or system that can easily integrate with current wheelchairs, particularly commercial models, and autonomously initiate help without the need for manual button presses or voice commands.[2] This project aims to enhance the functionality of a conventional wheelchair by integrating a health monitoring device based on a microcontroller. This device utilizes pulse detection to identify any cardiovascular abnormalities and sends alerts via a wireless network to predefined cell phones or email addresses. The targeted sensors are located within the armrests, serving as the interface for interaction. The objective was to develop an affordable and user-friendly wheelchair safety system that enables individuals with disabilities to stay connected with necessary contacts through SMS notifications. [3] presented a system design featuring a touch screen interface for human control of a smart wheelchair. This technology was chosen for its user-friendly nature, allowing operation through touch screen interactions. The system incorporates two distinct modes: hand-operated and automated. In the hand-operated mode, the touch screen facilitates direct interaction with the wheelchair. Meanwhile, the automated mode empowers patients to navigate indoor spaces along predefined paths programmed into the system. Additionally, an infrared sensor has been integrated for barrier sensing. The wheelchair is equipped with wireless communication capabilities to address emergency situations. [4] proposed a concept for a voice-operated smart wheelchair designed to cater to individuals with

physical disabilities, particularly those affecting the hands. The wheelchair's control relies on a motor control and drive system, incorporating DC Motors. When the system identifies and matches a voice command with pre-stored memory, a generated digital signal is transmitted to the processor, directing the wheelchair's movements accordingly. The research aims to elevate the capabilities of powered wheelchairs through the integration of a voice recognition system. Numerous sensors will be employed to perceive the surroundings, complemented by a speech interface to interpret and execute commands. [5] Here, the patient's position is identified through a tilt sensor using rolling ball technology, seamlessly interfaced with the CC3200 microcontroller featuring an integrated Wi-Fi on-chip module using the 802.11 b/g/n network protocol. The GeoTagging technique facilitates tracking of the patient's location. When the patient tilts on the wheelchair or adjustable bed, the tilt sensor detects the motion, and the resulting data is processed by the microcontroller. Subsequently, the data is uploaded to the ThingSpeak cloud, serving as a real-time values storage platform with graphical representation.To enhance alert capabilities, a SIM 900A GSM module is incorporated, enabling the system to send alerts in the event of detected abnormalities in the patient's position. Additionally, the GeoTagging feature enables easy access to the patient's location information. [6] The proposed system offers a simple and effective solution for disabled individuals, featuring automatic wheelchair functions including umbrella, foot and head mats, and obstacle detection. Weather conditions are monitored via humidity sensor, while obstacle detection is aided by GPS location tracking. Prototype testing confirms the system's efficiency.

Objective-

Our project aims to create a smart fall detection system based on IoT that can be designed using an accelerometer and gyroscope sensor to detect person movements. Also a health monitoring system is integrated with the wheelchair. It also aims on providing the exact location of the person if any mishap has occurred.

CHAPTER 2

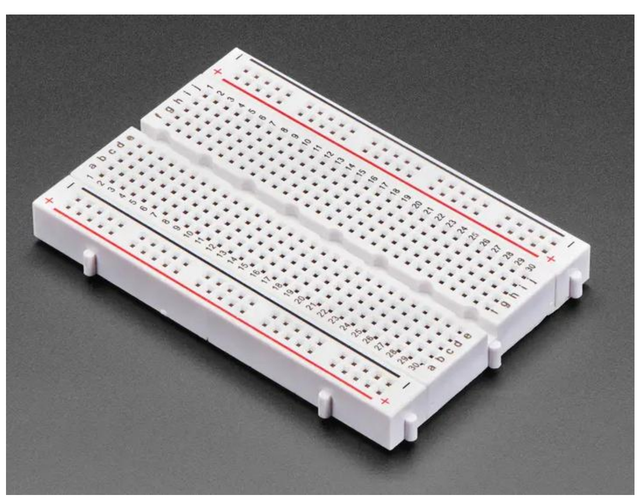
Methodology

Here, for the fall detection we use the MPU6050 module which consists of an accelerometer, gyroscope and a motion sensor. The gyroscope is used to detect the angular motion of the body. And the accelerometer measures the vibration, or in other words acceleration of the body. All the components are connected to the Raspberry Pi PicoW. It will send an alert to the respective person on the hour of need. It allows the microprocessor to connect to a 2.4GHz Wi-Fi connection using IEEE 802.11 bgn. When a fall is detected, an abnormal movement is detected by the gyroscope and the accelerometer and then the buzzer is triggered. From here, the PicoW receives an interrupt and sends an alert to the registered person's mobile number along with the coordinates of the fallen person. The system will also contain a health monitoring system which will regularly record the Oxygen level, beats per minute and temperature. It will help the registered user know whether the fall which has occurred is dangerous so that the person can take action as per the damage.

Tools used-

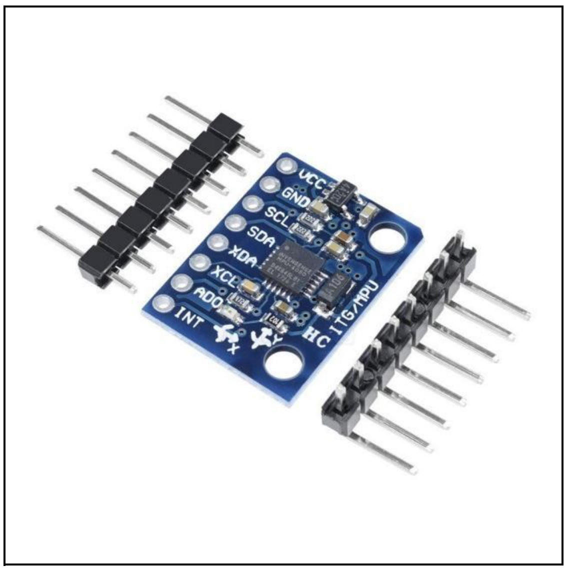
1. VeroBoard:

A breadboard is a building basis for creating semi-permanent prototypes of electrical circuits. Breadboards, unlike perfboards and stripboards, do not require soldering or track destruction, making them reusable. It's a rectangular plastic board with a lot of small holes in it. These holes allow you to quickly insert electrical components to prototype (create and test an early version of) an electronic circuit, such as this one with a battery, switch, resistor, and an LED (light-emitting diode), as well as wires, resistors, capacitors and coils.



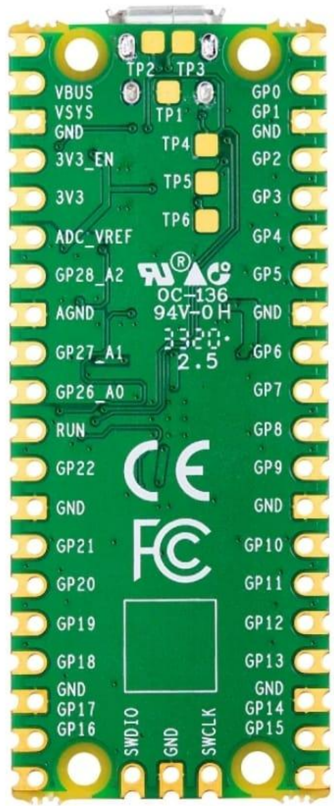
2. MPU6050 module-

The MPU6050 sensor module combines a three-axis accelerometer, three-axis gyroscope, and DMP into a small package to function as a complete six-axis motion tracking device. It communicates with microcontrollers in an easy-to-use manner by using the I2C protocol bus interface. Its versatility in applications is further enhanced by the inclusion of an Auxiliary I2C bus for communication with other sensor devices, like a three-axis magneto or pressure sensor.



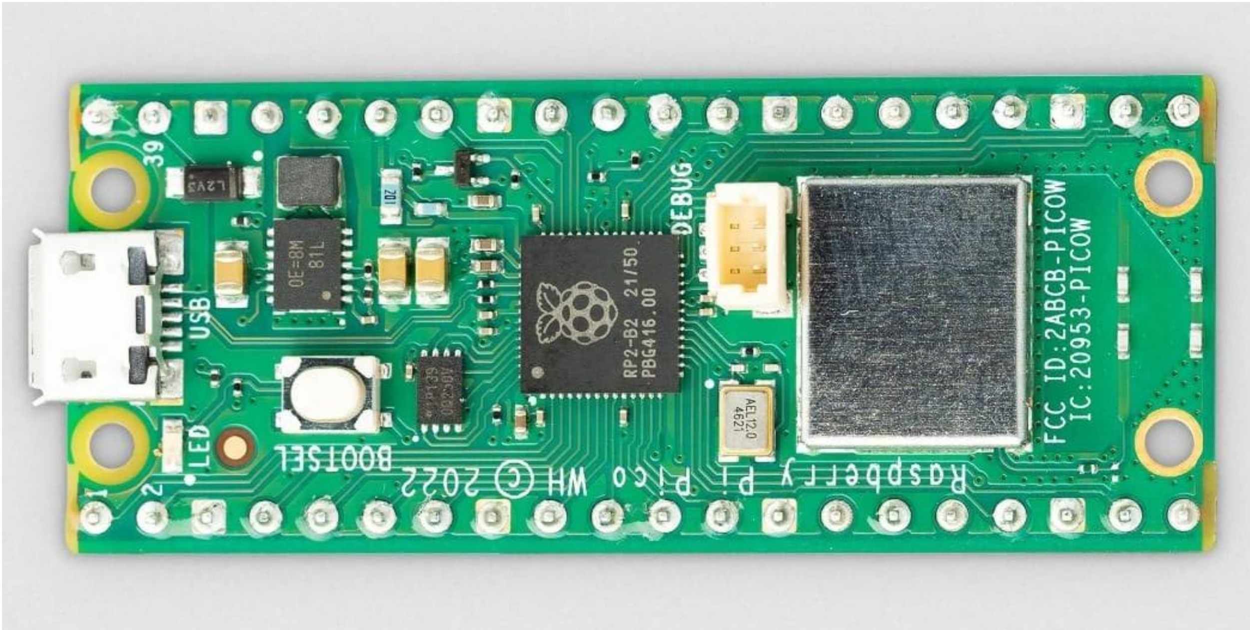
3. Raspberry Pi Pico:

The Raspberry Pi Pico is a microcontroller that may be used for a wide range of projects, including ones that require WiFi. It is a low-cost, user-friendly device that supports a number of programming languages, including C, C++, and MicroPython. Unlike the Raspberry Pi 4, the Raspberry Pi Pico does not use a full desktop operating system. Instead, it runs code with no desktop interface. It is breadboard compatible, and when connected via a USB cable, the Raspberry Pi Pico serves as a flash drive, allowing user to drag and drop software files onto it.



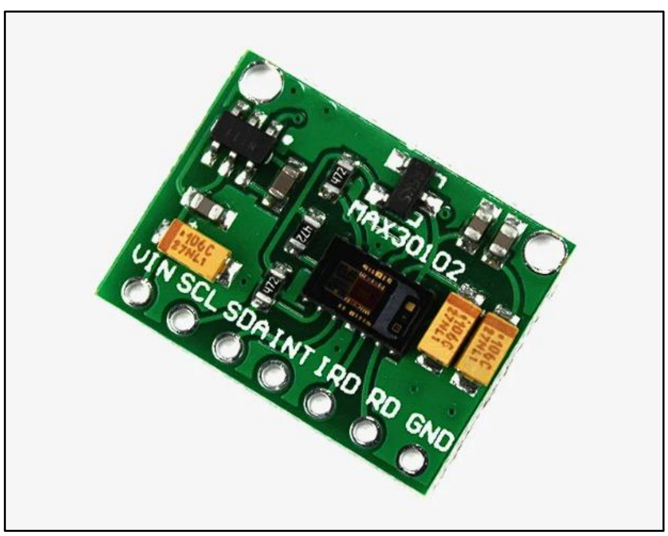
4. Raspberry Pi PicoW:

The Raspberry Pi Pico W is a low-cost, wireless microcontroller board developed for physical computing and Internet of Things applications. It has a 2.4 GHz wireless interface, Bluetooth 5.2, and a built-in antenna. It can be powered by micro USB, external power sources, or batteries. The Pico W supports drag-and-drop programming over USB. The Pico W's wireless capabilities allow your creations to talk to each other and to the internet. Essentially, the Pico W is a Pico with added Wi-Fi support, expanding its use cases for internet-connected applications.



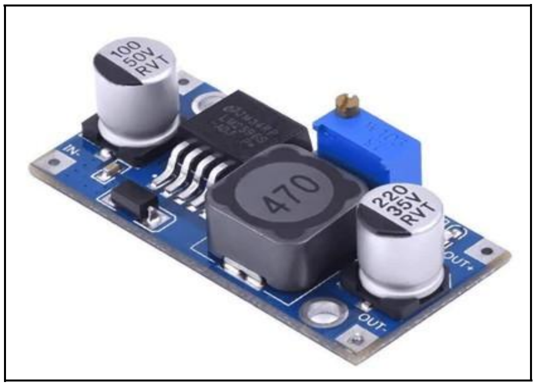
5. MAX30102:

The MAX30102 is a biosensor module that monitors heart rate and blood oxygen levels (SpO2). It's a compact, integrated system that includes LEDs, photodetectors, optical elements, and low-noise electronics.



6. Buck Converter:

A buck converter is a kind of DC-DC power converter that increases current proportionately while stepping down voltage from a higher level to a lower level. By regulating the duty cycle of the switching signal, it effectively controls the output voltage using a switching element, usually a transistor.

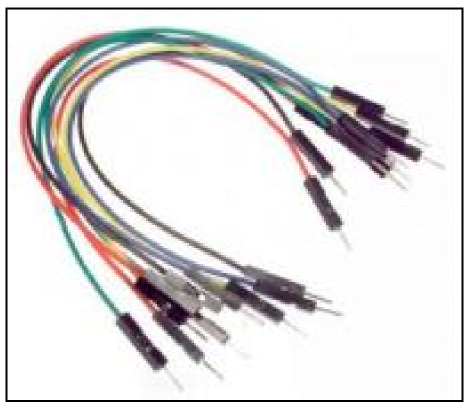


7. Sunboard:

Sunboard is a material that may be quickly and easily adapted to a variety of art, craft, and display purposes. Made of two layers of rigid, smooth paper or plastic, with a polystyrene foam core pressed between, Sunboard provides an easily cuttable, long-lasting surface that can be used for hanging photos, making signs, building architectural models, and building displays. Sunboard is a well-liked option for designers, artists, and enthusiasts due to its accessibility, user-friendliness, and support for multiple media.

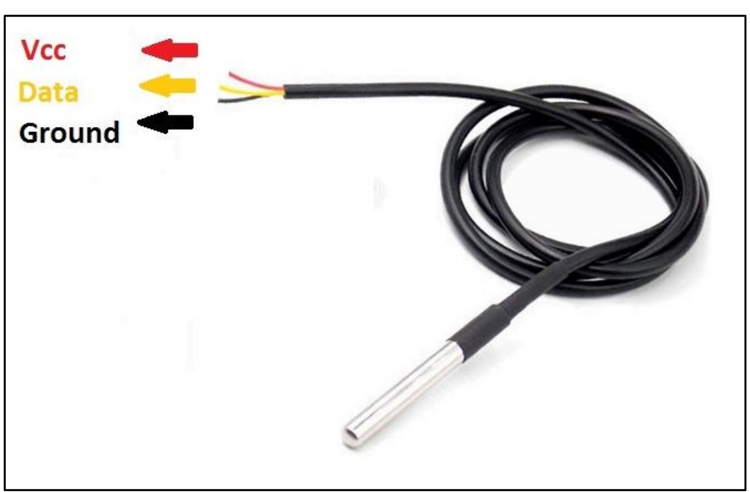
8. Connecting wires:

They are required for making the necessary connections.



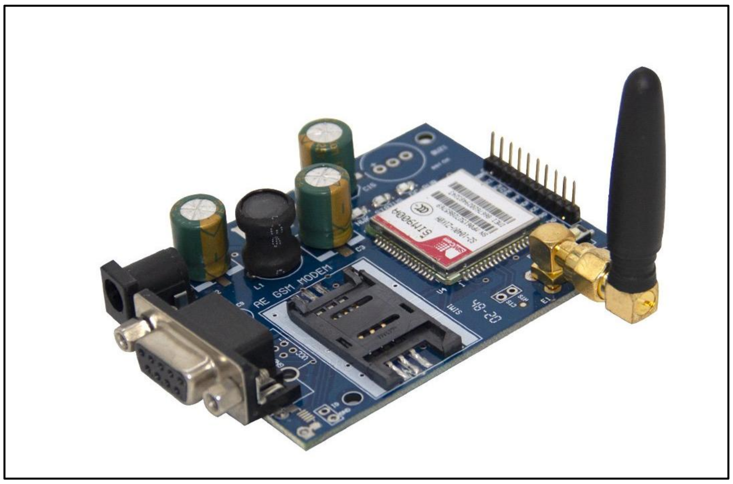
9. DS18B20:

The DS18B20 is a miniature temperature sensor with an integrated 12-bit ADC. It conveniently connects to an Arduino digital input. The sensor communicates via a single cable and requires very few extra components. The sensors have a claimed accuracy of +/-0.5 degrees Celsius for the temperature range of -10 to +85 degrees Celsius.



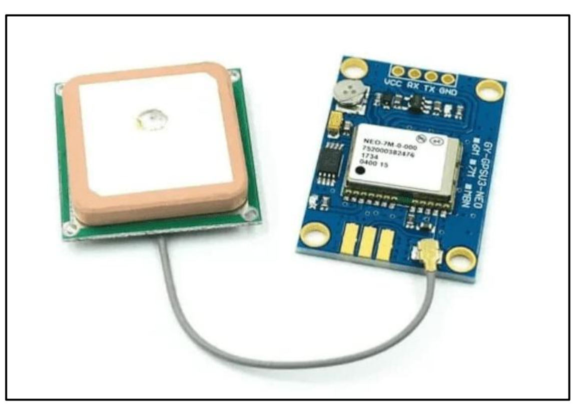
10. SIM900A GSM Module:

The SIM900A GSM module is a tiny, easy to use GSM (Global System for Mobile Communications) module that links microcontrollers and embedded systems to wireless networks to enable telephone service, messaging via SMS, and basic GPRS data handling. It performs at 2G (or at times 3G) frequencies (generally 900/1800 MHz) and is often utilized for remote monitoring, SMS warnings, and IoT applications that call for communication via cell phones. The SIM900A module interacts to a microcontroller and microprocessor via serial communication (UART) and requires an external source of power, typically 5V.



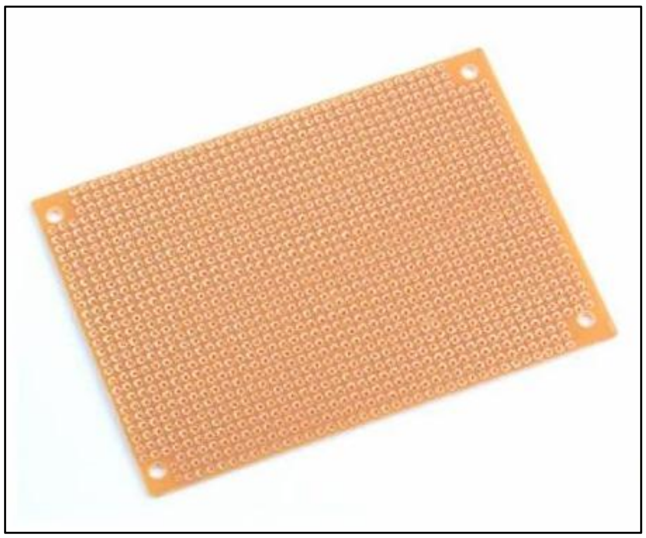
11. NEO7M GPS Module:

Ublox created the Neo7M GPS module, a high-performing GPS receiver that is frequently used for accurate position tracking in robots, drones, and do-it-yourself electronics. It provides quick and accurate location improvements by combining a GPS receiver with an antenna. Dependability is increased by the module's support for many satellite systems, such as GPS, SBAS, and QZSS. The Neo7M is ideal for applications looking for consistent navigational statistics because of its small form factor, low power consumption, and integrated EEPROM for recording settings.

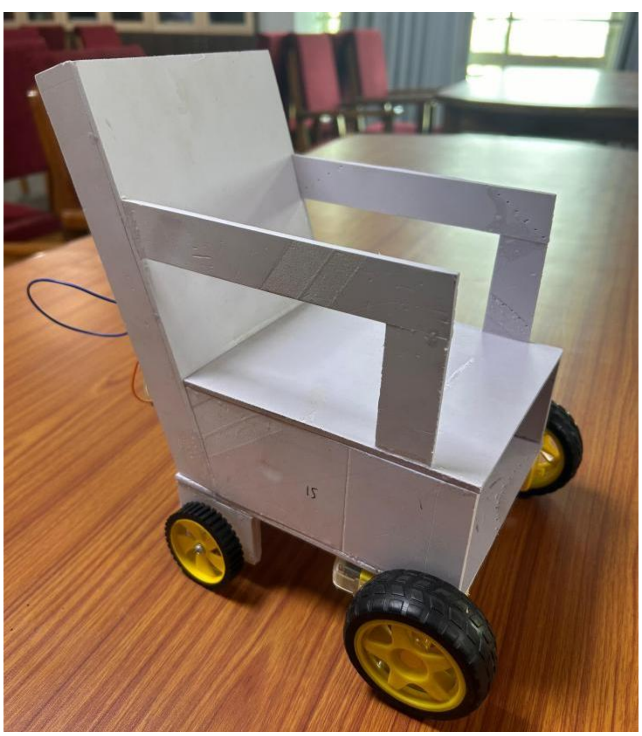


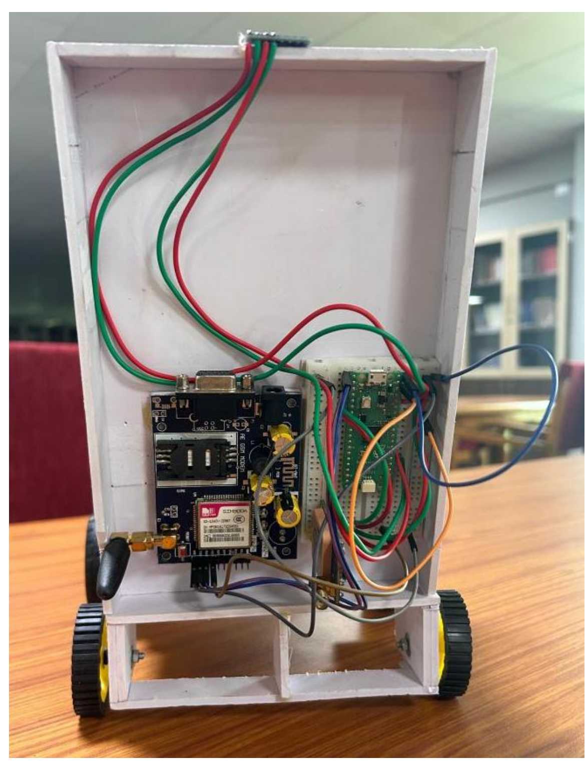
12. Veroboard:

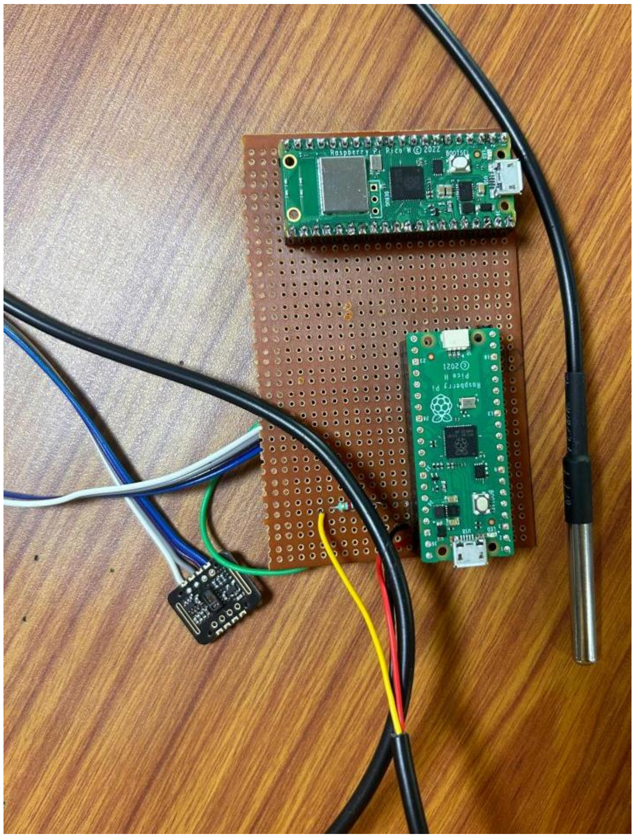
Veroboard, a type of stripboard, is a pre-made circuit board consisting of copper strips adhered to an insulating paper board. It is commonly used in electronic circuitry, where electrical connections are established through copper strips on the underside of the board. Veroboard features a printed circuit design, with rows of copper tracks and drilled holes for soldering electronic components. Its distinctive 0.1-inch grid of holes and broad parallel copper strips running along one side facilitate circuit construction.

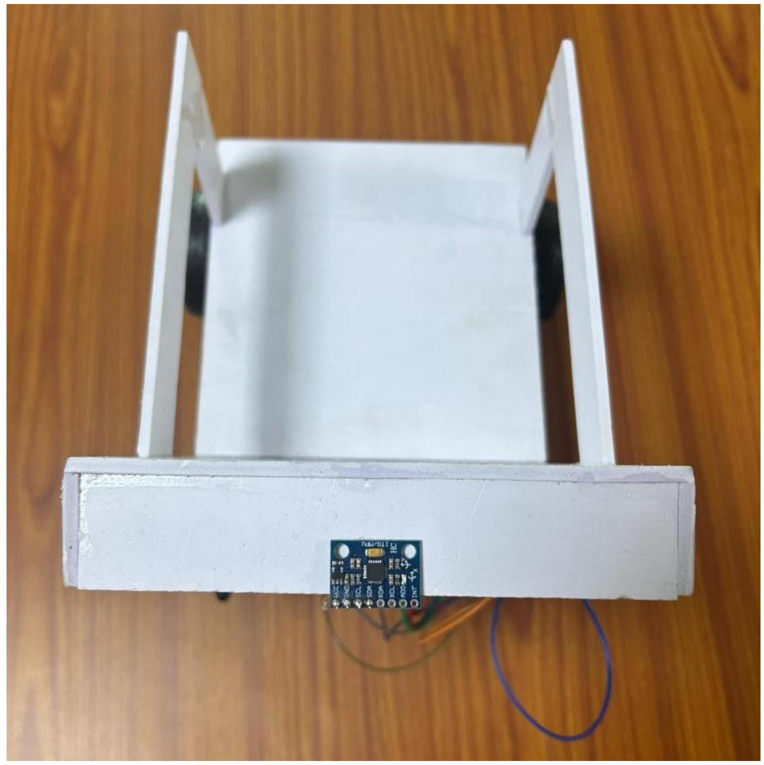


Proposed Model-









Working-

- Firstly, all the sensors are activated when the wheelchair is in the initial position.
- The health monitoring sensors, i.e, the heart beat sensor starts monitoring the vitals of the person.
- When the wheelchair is at normal speed and no changes or danger is there, no notification is sent to the registered user.
- The close contacts of the patient have to register themselves in the Blynk IoT application for emergency purposes. Whenever a mishap will occur the respective contacts will be notified.
- We are using the accelerometer and gyroscope to detect falls. Here, we're using the MPU6050 model, which has both sensors integrated. As previously indicated, the accelerometer detects a change in the wheelchair's speed and alerts the registered contact to potential danger if it crosses the cut-off point. Notifications within the app will also be sent.
- Now we also have to keep a check at the patient's BP and SpO2 level. With continuous contact with the fingertips of the patient, the registered user is able to keep a check on the same via the app

- In case the patient falls, then the contact of the patient with the heartbeat sensor is cut. As a result of which the value of becomes zero.
- As soon as it becomes zero, a mail is sent to the registered user to check the orientation of the body of the patient whether the fall is serious or not.
- Here, we are using the MAX30102 sensor. The MAX30102 is an integrated pulse oximetry and heart-rate monitor module.
- If the registered user is not so far away, he/she can come at once and help the patient if any mishap has occurred.
- If he/she is unable to come at that moment, they may call for the neighbors to check upon or can contact the nearby hospital if the damage is serious.

CHAPTER 3

Challenges and Alternative Trade-off

Challenges and Remedy-

The challenge regarding the exact location of the patient has been dealt with. We have implemented a gps/gsm module which will share the exact location of the system with the intended person. The remaining challenge that prevails is that of protection. We are still in the process of implementing low-cost protection mechanism. Air bags are the recommended tool for the purpose. We are currently in progress to implement the same

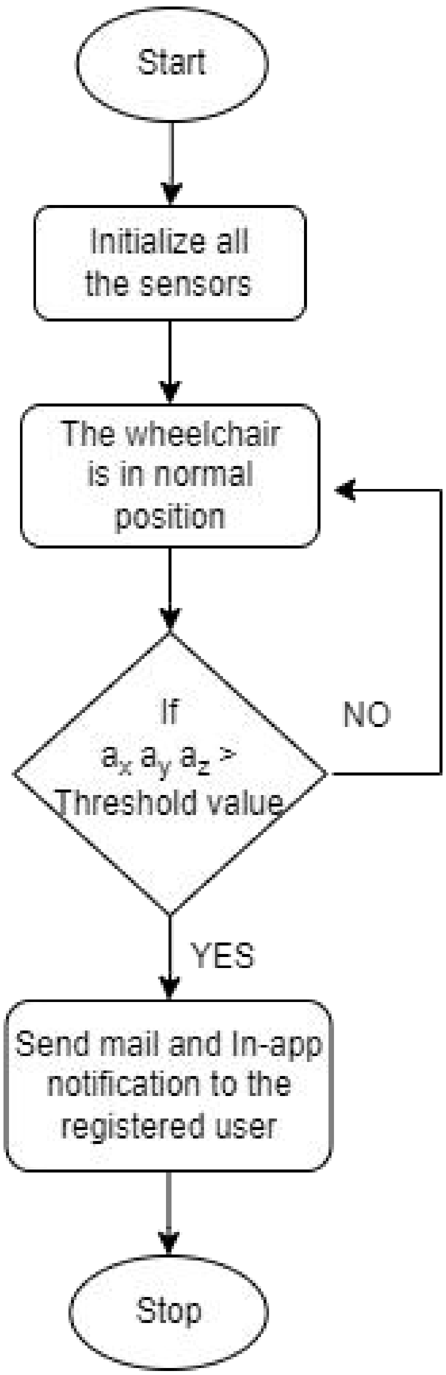
Alternative and Trade-offs-

The only tradeoff we are facing is regarding the delay. In real time, we will be facing the delay as the data will take time to reach the respective user. But we have tried to decrease the delay as much as possible. We are implementing the system in a commercialized wheelchair. In short, using IoT we are making the simple thing a smart one. And besides being smart, we have also tried to make it as low-cost as possible. The alternative to this can be an in-built smart wheelchair with all the amenities present beforehand.

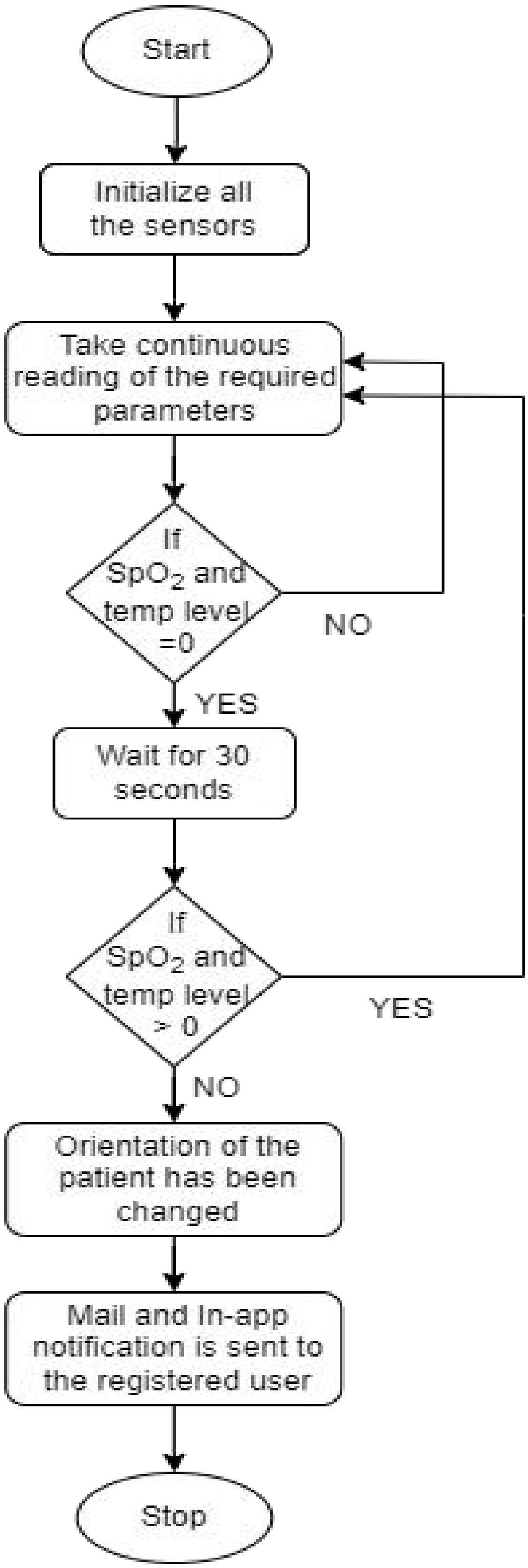
CHAPTER 4

Design Approach and Algorithm

The working of our model has been designed in the form of flowchart:



This describes the fall detection process



This describes the health monitoring system

Individual Code-

The individual code for the calibration of MPU6050 is given below:

```
from machine import Pin, I2C
import utime

PWR_MGMT_1 = 0x6B
SMPLRT_DIV = 0x19
CONFIG = 0x1A
GYRO_CONFIG = 0x1B
ACCEL_CONFIG = 0x1C
TEMP_OUT_H = 0x41
ACCEL_XOUT_H = 0x3B
GYRO_XOUT_H = 0x43

def init_mpu6050(i2c, address=0x68):
    i2c.writeto_mem(address, PWR_MGMT_1, b'\x00')
    utime.sleep_ms(100)
    i2c.writeto_mem(address, SMPLRT_DIV, b'\x07')
    i2c.writeto_mem(address, CONFIG, b'\x00')
    i2c.writeto_mem(address, GYRO_CONFIG, b'\x00')
    i2c.writeto_mem(address, ACCEL_CONFIG, b'\x00')

def read_raw_data(i2c, addr, address=0x68):
    high = i2c.readfrom_mem(address, addr, 1)[0]
    low = i2c.readfrom_mem(address, addr + 1, 1)[0]
    value = high << 8 | low
    if value > 32768:
```

```
        value = value - 65536
    return value


def get_mpu6050_data(i2c):
    temp = read_raw_data(i2c, TEMP_OUT_H) / 340.0 + 36.53
    accel_x = read_raw_data(i2c, ACCEL_XOUT_H) / 16384.0
    accel_y = read_raw_data(i2c, ACCEL_XOUT_H + 2) / 16384.0
    accel_z = read_raw_data(i2c, ACCEL_XOUT_H + 4) / 16384.0
    gyro_x = read_raw_data(i2c, GYRO_XOUT_H) / 131.0
    gyro_y = read_raw_data(i2c, GYRO_XOUT_H + 2) / 131.0
    gyro_z = read_raw_data(i2c, GYRO_XOUT_H + 4) / 131.0

    return {
        'temp': temp,
        'accel': {
            'x': accel_x,
            'y': accel_y,
            'z': accel_z,
        },
        'gyro': {
            'x': gyro_x,
            'y': gyro_y,
            'z': gyro_z,
        }
    }
```

This code sets up and reads data from an MPU6050 sensor, which is often used to record acceleration, gyroscope data, and temperature. The init_mpu6050 function initializes the sensor by defining registers over an I2C interface to provide sample rates, gyro, and accelerometer parameters. The read_raw_data function extracts raw 16-bit data from specified sensor registers and transforms it to signed values while allowing for two's complement representation. The get_mpu6050_data function collects temperature, acceleration, and gyroscope measurements by converting raw data to human-readable units (degrees Celsius for temperature and scaled values for acceleration and rotation), then returns a dictionary containing the parsed sensor data. This code enables easy interaction with the MPU6050 to detect movement and orientation.

The individual code for calibration of GPS/GSM module is given below:

```
import time
from machine import UART, Pin
from micropyGPS import MicropyGPS
from machine import I2C
import utime
import math
from mpu6050 import init_mpu6050, get_mpu6050_data

i2c = I2C(0, scl=Pin(21), sda=Pin(20), freq=400000)
init_mpu6050(i2c)

led = Pin(25, Pin.OUT)
buzz = Pin(10, Pin.OUT)

# Initialize UART for the GPS module (TX=Pin 4, RX=Pin 5)
uart_gps = UART(1, baudrate=9600, tx=Pin(4), rx=Pin(5))

# Initialize UART for the SIM900A module (TX=Pin 16, RX=Pin 17)
uart_sim = UART(0, baudrate=9600, tx=Pin(0), rx=Pin(1))

# Initialize MicropyGPS object with UTC offset (0 for UTC, adjust if needed)
gps = MicropyGPS(0)

# Set up a switch (button) on Pin 15
button_pin = Pin(15, Pin.IN, Pin.PULL_UP)

# Function to read and update GPS data
```

```python
def update_gps():
    while uart_gps.any():
        data = uart_gps.read(1)
        if data:
            try:
                # Attempt to decode the data
                data = data.decode('utf-8')
                gps.update(data)
            except Exception as e:
                print("Error decoding GPS data:", e)

def send_sms(message):
    def uart_read_with_timeout(timeout=5):
        start_time = time.time()
        while True:
            if uart_sim.any():
                return uart_sim.read()
            if time.time() - start_time > timeout:
                return b""  # Return empty byte if timed out

    def check_response(expected_response, timeout=5):
        response = uart_read_with_timeout(timeout)
        print(response)  # Print the response for debugging
        return expected_response in response

    # Initialize SIM900A module
    uart_sim.write('AT\r')
    time.sleep(1)
    if not check_response(b"OK"):
        print("SIM900A failed to respond to AT command.")
        return

    # Set SMS mode to text
    uart_sim.write('AT+CMGF=1\r')
    time.sleep(1)
    if not check_response(b"OK"):
        print("Failed to set SMS mode to text.")
        return

    # Start sending the message
```

```
    uart_sim.write('AT+CMGS="+916371568550"\r')  # Replace with your phone number
    time.sleep(1)
    if not check_response(b">"):  # Wait for the '>' prompt
        print("No '>' prompt received.")
        return

    # Send the actual message followed by Ctrl+Z separately
    uart_sim.write(message)
    time.sleep(1)
    uart_sim.write(b"\x1A")  # Send Ctrl+Z to complete the SMS command
    print("Sent Ctrl+Z to finalize SMS")

    # Wait for confirmation after sending Ctrl+Z
    response = uart_read_with_timeout(timeout=10)  # Longer timeout for final confirmation
    print(response)
    if b"+CMGS" in response:
        print("SMS sent successfully.")
    else:
        print("Failed to send SMS or no confirmation received.")

def calculate_tilt_angles(accel_data):
    x, y, z = accel_data['x'], accel_data['y'], accel_data['z']

    tilt_x = math.atan2(y, math.sqrt(x * x + z * z)) * 180 / math.pi
    tilt_y = math.atan2(-x, math.sqrt(y * y + z * z)) * 180 / math.pi
    tilt_z = math.atan2(z, math.sqrt(x * x + y * y)) * 180 / math.pi

    return tilt_x, tilt_y, tilt_z

def complementary_filter(pitch, roll, gyro_data, dt, alpha=0.98):
    pitch += gyro_data['x'] * dt
    roll -= gyro_data['y'] * dt

    pitch = alpha * pitch + (1 - alpha) * math.atan2(gyro_data['y'], math.sqrt(gyro_data['x'] *
gyro_data['x'] + gyro_data['z'] * gyro_data['z'])) * 180 / math.pi
    roll = alpha * roll + (1 - alpha) * math.atan2(-gyro_data['x'], math.sqrt(gyro_data['y'] *
gyro_data['y'] + gyro_data['z'] * gyro_data['z'])) * 180 / math.pi

    return pitch, roll

pitch = 0
```

```
roll = 0
prev_time = utime.ticks_ms()


# Main loop
while True:
    # Update GPS data
    update_gps()
    data = get_mpu6050_data(i2c)
    curr_time = utime.ticks_ms()
    dt = (curr_time - prev_time) / 1000

    tilt_x, tilt_y, tilt_z = calculate_tilt_angles(data['accel'])
    pitch, roll = complementary_filter(pitch, roll, data['gyro'], dt)

    prev_time = curr_time

    print("Tilt angles: X: {:.2f}, Y: {:.2f} degrees".format(tilt_x, tilt_y))

    # Check if GPS has a valid fix
    if gps.fix_stat >= 1:  # A fix is valid if the status is 2 or more
        #print("Abhi aur Chalega")
        print("Time (UTC):", "{:02}:{:02}:{:02}".format(gps.timestamp[0], gps.timestamp[1],
gps.timestamp[2]))
        print("Latitude:", gps.latitude[0], gps.latitude[1])
        print("Longitude:", gps.longitude[0], gps.longitude[1])
        print("Altitude (m):", gps.altitude)
        print("Speed (km/h):", gps.speed[2])
        print("Satellites in use:", gps.satellites_in_use)
        print("Date (DD/MM/YYYY):", "{:02}/{:02}/{}".format(gps.date[0], gps.date[1],
gps.date[2] + 2000))
        print("----------------------------")

        if tilt_x > 20 or tilt_x < -20 or tilt_y > 10 or tilt_y < -20 :
            latitude = f"{gps.latitude[0]} {gps.latitude[1]}"
            longitude = f"{gps.longitude[0]} {gps.longitude[1]}"
            message = f"Help me! My location is Latitude: {latitude}, Longitude: {longitude}"
            send_sms(message)
            print("SMS sentz:", message)
            led.value(1)
            buzz.value(1)
```

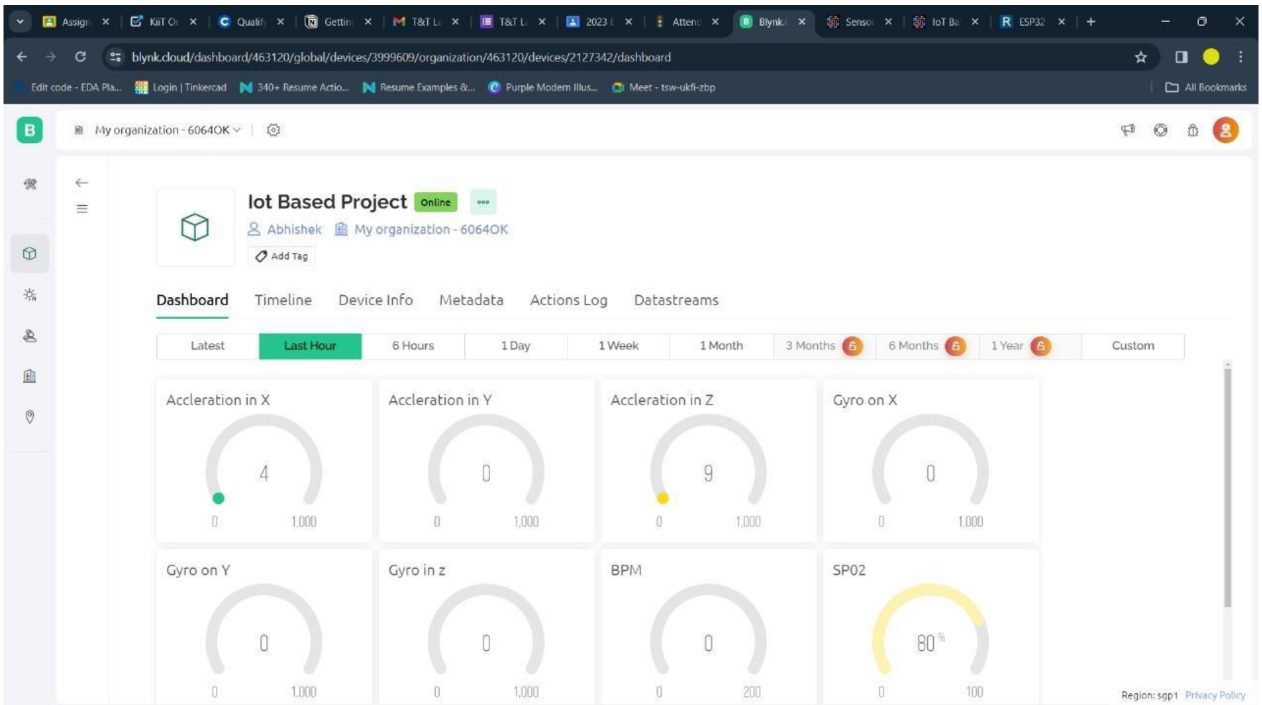
```
        else:
            led.value(0)
            buzz.value(0)

    else:
        print("Waiting for valid GPS data...")

    # Pause briefly before the next update
    time.sleep(1)

print("Loop Khatam")
```

Based on particular tilt circumstances, this code sends emergency SMS notifications and utilizes a microcontroller to track orientation and GPS position. It sets up an MPU6050 sensor to monitor orientation and tilt using gyroscope and accelerometer data, which are then processed by a complementary filter to determine exact tilt angles. In order to ascertain latitude, longitude, and other information, it also reads GPS data from a GPS module. An LED and buzzer are activated as alerts by the software, which uses a SIM900A module to send an SMS with the current GPS position if the tilt above predetermined thresholds. Continually looping over orientation, GPS, and status data, the computer waits for a legitimate GPS fix before assessing circumstances and issuing alarms.

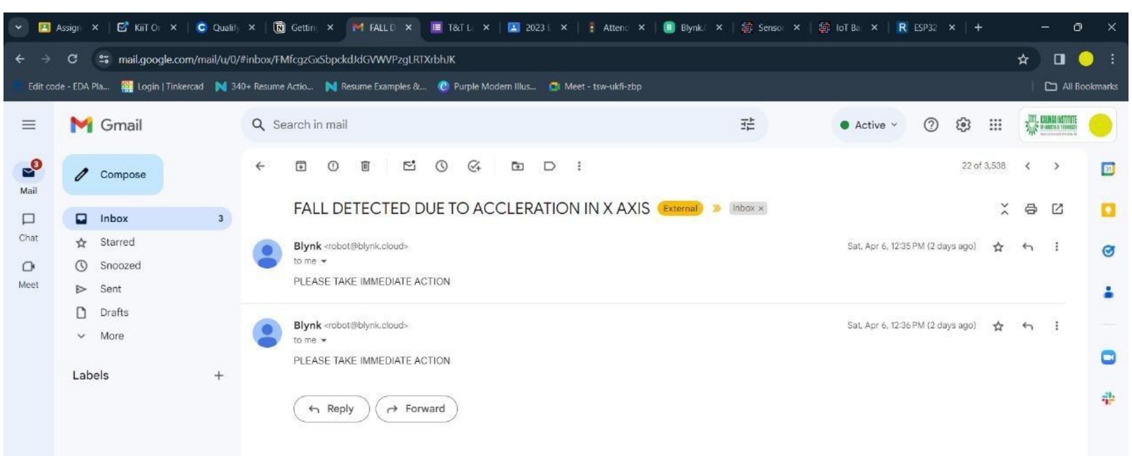
CHAPTER 5

Result Observation

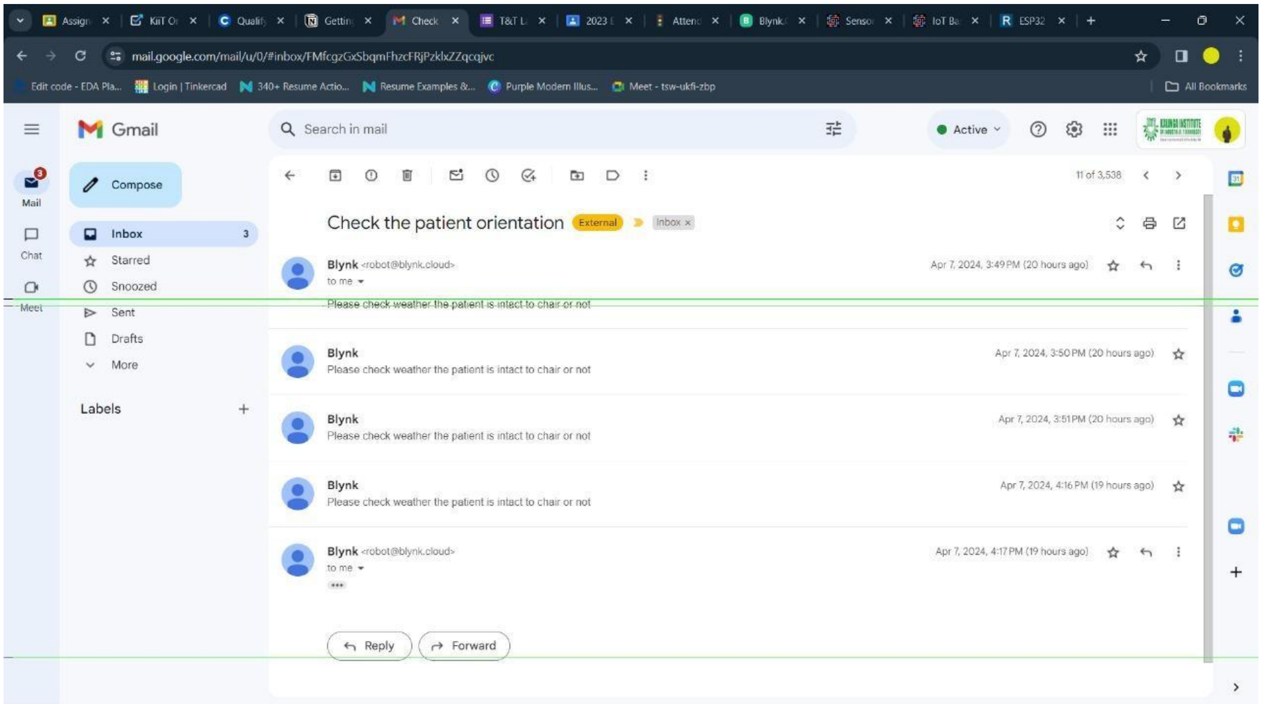
Result Analysis:



The Blynk IoT interface where all the parameters are shown



The mail received at the user side when threshold value of accelerometer is crossed



The mail interface when the contact with patient is cut with sensor as a result of falling down

Discussion-

The data shown in the interface can be helpful for the registered user if he/she is staying away from the patient for most of the time and usually the patient works independently. The data can be analyzed and processed to see if the consequences of falling are happening mostly and proper actions can be taken.

CHAPTER 6

Conclusive Remarks

Conclusion-

To sum up, we can say that our model consists of the following characteristics:

(i) Detection of the fall of the patient by observing the change of the orientation of the body of the patient and also by the acceleration of the wheelchair

(ii) Keeping a health monitoring system which includes the heart beat, Oxygen level and temperature of the patient.

(iii) It will be able to give the precise location of the patient with the help of GSM/GPS module

We have tried to implement these features in as low cost as possible so that all the patients who need them can have access to them.

Future Scope-

We are trying to implement automatic locomotion of the wheelchair so that the person sitting on wheelchair may not need manual support for locomotion. Also we will try to implement protection of the patient if a fall occurs.

REFERENCES

[1] Vishwakarma,S.K., 2017. “ Development of smart wheelchairs ” Dehradun DIT University.

[2] SayaliM. Kamble, Pratiksha R. Kawle, Suwarna S. Mohile, MeghaR. Meshram,Prof. Gayatri Padole Mam, Neetu Gyanchandani Mam, 2022. “ IoT Grounded Person/ Wheelchair Fall Discovery System ”, IJRASET,pp. 2321- 9653

[3] Posugade,V.G., Shedge,K.K. & Tikhe,C.S., 2012. Touch- Screen Grounded Wheelchair System. International Journal of Engineering Research and Application(IJERA), 2(2),pp. 1245-1248.

[4] Ms.S.D.Suryawanshi,Mr.J.S.Chitode &Ms.S.S.Pethakar, 2013. Voice Operated Intelligent Wheelchair. International Journal of Advanced Research in Computer Science and Software Engineering, 3(5),pp. 487- 490.

[5] K. Evangili Supriya andK. Raghavendra Rao, 2019. “ perpetration of Cloud IoT- Grounded Case Fall Discovery System with GeoTagging using cock Detectors and Texas Instruments ’ CC3200 ”, Indian Journal of Science and Technology(IJST),pp. 0974- 5645

[6] Deepak Kumara, Reetu Malhotrab,S.R. Sharma, Chitkara University Institute of Engineering and Technology, Chitkara University, Punjab, India, 2020, “ Design and Construction of a Smart Wheelchair ”, Elsevier B.V,pp. 302- 307

[7] Nsengumuremyi,L., Karki,A.J. & CR,M., 2018. Smart Wheelchair Using Medical Iot. International Journal for Research in Applied Science & Engineering Technology, 6(5),pp. 387-393.

[8] Narendra Babu Alur, Kurapati Poorna Durga, Boddu Ganesh, Manda Devakaruna, Lakkimsetti Nandini,A. Praneetha,T. Satyanarayana,K. Girija Sravani, 2023. “ Design of Smart Wheelchairs with Health Monitoring System ”.

[9] Divya Jennifer DSouza, Sanket Srivastava, Ruth Prithika, Sahana Rai A N, NMAM Institute of Technology, Nitte, Karnataka, India, 2019, “ IoT grounded Smart seeing Wheelchair to help in Healthcare ” IRJET, p- ISSN 2395- 0072

[10] Bao J, Li W, Li J, “ Sitting Posture Recognition grounded on data emulsion on pressure bumper, ” TELKOMNIKA Indonesian Journal of Electrical Engineering, 2013, 11(4),pp. 1769-1775.

[11] Seneviratne,P., 2018. Hand- On the Internet of effects with Blynk. InP. Bisht, ed. figure on the power of Blynk to configure smart bias and make instigative IoT systems. Birmingham Packt Publishing,p. 246.

[12] L. Hou,J. Latif,P. Mehryar,A. Zulfiqur,S. Withers andA. Plastropoulos," IoT Based Smart Wheelchair for Elderly Healthcare Monitoring," 2021 IEEE 6th International Conference on Computer and Communication Systems(ICCCS), Chengdu, China, 2021,pp. 917- 921, doi 10.1109/ ICCCS52626.2021.9449273. keywords{ Temperature detectors; pall computing; Wheelchairs; elderly citizens; Prototypes; Medical services; Real- time systems; smart wheelchair; internet of effects; real- time monitoring; handicap avoidance},

Appendix A: Gantt Chart

	July	Aug	Sep/Oct	Nov
Literature Survey				
Problem identification				
Research on the Project Objective				
Hardware/Software/Tool Selection				
Formation of Codes/Experiment Design				
Trial and Testing				
Challenges and Remedy				
Assembling of the Prototype/Model				
Project Demonstrations				
Formation of the Project Report				
Finalizing of Project Presentation				

Appendix B: Project Summary

Project Title	IoT BASED SMART WHEELCHAIR FOR FALL DETECTION AND HEALTH MONITORING SYSTEM
Team Members	Abhishek Raj Kashyap , Arundhati Lal , Harsh Prakash and Saumya Raj
Supervisor	Prof. Sukanta Kumar Sabut
Semester/Year	7th semester/4th year
Project Abstract	Falls pose significant risks to the elderly and those with mobility impairments, often resulting in serious injuries or fatalities, especially when they occur in isolation. While existing webcam-based monitoring technologies are effective, they are costly, limited to indoor settings, and raise privacy concerns. To address these issues, this research proposes a cost-effective IoT-based fall detection system for wheelchairs. Utilizing an MPU6050 sensor and Pi PicoW MP, the system detects abnormal movements indicative of a fall and alerts designated contacts via a Blynk mobile app and email notification. Health monitoring features, including blood pressure and oxygen levels, are integrated using MAX30102 and DS18B20 sensors. Data, including wheelchair location, is stored in a database for analysis. While the system offers quick response and affordability, it relies on a stable WiFi connection and could benefit from improved GUI design and GPS integration. Future enhancements could include additional detection systems for enhanced accuracy.
Code and standard	MPU6050(includes gyroscope and accelerometer), MAX30102(oxygen level and heart beat sensor) interfaced with Raspberry Pi PicoW module. Necessary codes are written in Python Data collected in the Blynk IoT application
Overall Cost	4500-5000
Culminating Knowledge and lifelong learning experience	For this project we took knowledge from the following subjects, EC 3003 Microprocessors and Microcontrollers EC 3093 MMI Lab EC 4003 Wireless and Mobile Communication EC 6128 Wireless Sensor and Networks

A project report on IoT BASED SMART WHEELCHAIR FOR FALL DETECTION AND HEALTH MONITORING SYSTEM

ORIGINALITY REPORT

15%	15%	3%	12%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	how2electronics.com Internet Source	5%
2	iotprojectsideas.com Internet Source	4%
3	pdfcoffee.com Internet Source	1%
4	www.isindexing.com Internet Source	1%
5	Submitted to University of Hertfordshire Student Paper	1%
6	www.researchgate.net Internet Source	1%
7	www.coursehero.com Internet Source	1%
8	www.cademix.org Internet Source	<1%
9	Submitted to Durban University of Technology	<1%

Student Paper

10	fdocuments.in Internet Source	<1 %
11	Submitted to Higher Education Commission Pakistan Student Paper	<1 %
12	Submitted to Lebanese International University Student Paper	<1 %
13	Ton Duc Thang University Publication	<1 %
14	123dok.com Internet Source	<1 %
15	suspace.su.edu.bd Internet Source	<1 %
16	ece.anits.edu.in Internet Source	<1 %

Exclude quotes	On	Exclude matches	< 10 words
Exclude bibliography	On		