

MOTION AND POSITION TRACKING SYSTEM USING MPU6050 SENSOR

TEAM MEMBERS:

Joseph Manuel Thomas R

Ram Prasanth S

Manoj N

Revanth Kumar M G

Aim:

To design and implement a fault detection system using the MPU6050 sensor, capable of accurately detecting and analysing real-time orientation, acceleration, and angular velocity for applications in robotics, gesture control, and navigation systems.

Tools / Hardware Required:

- ESP32 Dev board
- MPU6050 Module
- Jumper Wires
- Bread Board
- Power Source
- OLED Display (SSD1306)
- Buzzer

Theory:

Motion and Position tracking system is a crucial system in applications such as Automotive Industry, Industrial Automation, Consumer Electronics, Medical Devices and Aerospace and Defense. This project includes:

1. ESP32:
 - ESP32 has Wi-Fi + Bluetooth, runs on 3.3V, and has a fast processor for real-time tasks.
 - It offers 48 GPIO pins, but most dev boards give access to 25–34 usable pins.
 - These pins connect sensors, displays, and other hardware for IoT or embedded projects.
2. MPU6050 Module:
 - The MPU-6050 is an integrated 6-axis motion sensor combining a 3-axis gyroscope and 3-axis accelerometer, featuring an on-chip Digital Motion Processor (DMP) and 16-bit ADCs for accurate data
3. OLED Display (SSD1306):
 - An SSD1306 OLED is a compact, monochrome display with a 128x64 resolution, using an SSD1306 driver IC for self-lit pixels and featuring an I2C or SPI interface for easy connection to microcontrollers.

Pin Connections:

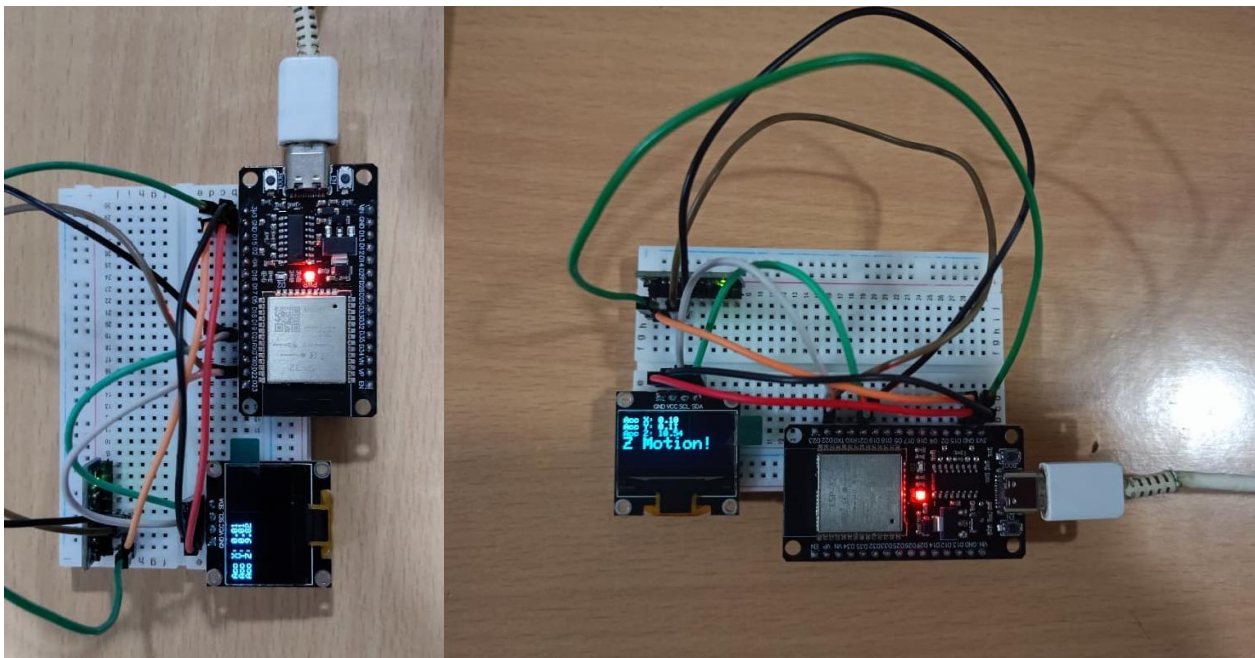
ESP32	MPU6050 Pin	SSD1306 Display
3.3V	VCC	VCC
GND	GND	GND
GPIO21	SDA	SDA
GPIO22	SCL	SCL

ESP32	TMB12A05 BUZZER
GPIO25	+
GND	-

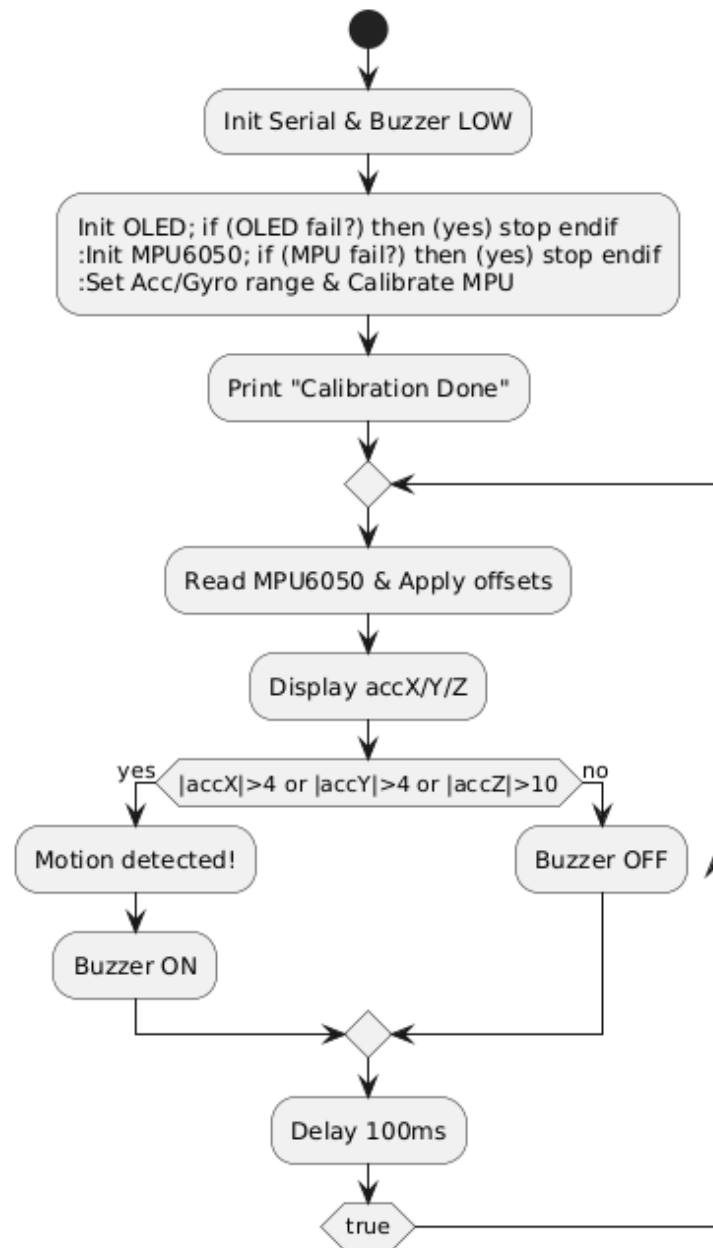
Procedure:

- Connect the ESP32 with the MPU6050 sensor using I²C pins (SDA, SCL, VCC, GND).
- Connect the OLED display to the same I²C pins of the ESP32.
- Connect the TMB12A05 buzzer to a GPIO pin of the ESP32 and GND.
- Install Arduino IDE and add ESP32 board support.
- Install the required libraries for MPU6050 and OLED display.
- Write or upload the program to the ESP32 using Arduino IDE.
- Power the ESP32 and observe the sensor values on the OLED display.
- Move the MPU6050 sensor to test motion detection.
- Check that the buzzer turns ON when sudden motion is detected.

Circuit:



Flowchart:



Program:

```
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#include <Adafruit_MPU6050.h>
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 64
#define BUZZER 25
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, -1);
Adafruit_MPU6050 mpu;
float offX=0, offY=0, offZ=0;
```

```

void calibrateMPU(int n=500){
  Serial.println("Calibrating... Keep MPU6050 still!");
  for(int i=0;i<n;i++){
    sensors_event_t a,g,t; mpu.getEvent(&a,&g,&t);
    offX+=a.acceleration.x; offY+=a.acceleration.y; offZ+=a.acceleration.z-9.81;
    delay(5);
  }
  offX/=n; offY/=n; offZ/=n;
  Serial.printf("Offsets: %.2f, %.2f, %.2f\n", offX, offY, offZ);
}

void setup(){
  Serial.begin(115200);
  pinMode(BUZZER,OUTPUT); digitalWrite(BUZZER,LOW);
  if(!display.begin(SSD1306_SWITCHCAPVCC,0x3C)){Serial.println("SSD1306 fail"); for(;;);}
  display.clearDisplay(); display.setTextSize(1); display.setTextColor(SSD1306_WHITE);
  display.setCursor(0,0); display.println("MPU6050 Init..."); display.display();
  if(!mpu.begin()){Serial.println("MPU6050 fail"); while(1) delay(10);}
  mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
  mpu.setGyroRange(MPU6050_RANGE_500_DEG);
  mpu.setFilterBandwidth(MPU6050_BAND_21_HZ);
  delay(1000); calibrateMPU();
  display.clearDisplay(); display.setCursor(0,0); display.println("Calibration Done!");
  display.display(); delay(1000);
}

void loop(){
  sensors_event_t a,g,t; mpu.getEvent(&a,&g,&t);
  float x=a.acceleration.x-offX, y=a.acceleration.y-offY, z=a.acceleration.z-offZ;
  display.clearDisplay(); display.setCursor(0,0); display.setTextSize(1);
  display.printf("Acc X: %.2f\nAcc Y: %.2f\nAcc Z: %.2f\n", x, y, z);
  bool motion=false;
  if(fabs(x)>4){display.setTextSize(2); display.println("X Motion!"); Serial.println("⚠ Motion on X axis!"); motion=true;}
  if(fabs(y)>4){display.setTextSize(2); display.println("Y Motion!"); Serial.println("⚠ Motion on Y axis!"); motion=true;}
  if(fabs(z)>10){display.setTextSize(2); display.println("Z Motion!"); Serial.println("⚠ Motion on Z axis!"); motion=true;}
  display.display();
  digitalWrite(BUZZER,motion?HIGH:LOW);
  delay(100);
}

```

Result:

Thus the fault detection system was successfully implemented using the ESP32, MPU6050 sensor, OLED display, and TMB12A05 buzzer. The MPU6050 accurately measured acceleration and angular velocity values, which were displayed in real time on the OLED screen. When sudden motion was detected, the buzzer was activated, providing an audible alert. Thus, the system worked effectively by offering both visual and audio feedback for motion detection.