```
// ================================
// PARALYTIC RISK MONITORING SYSTEM
// ESP32 + MAX30102 + MPU6050
// UI-Friendly Version + Fall + Tremor
// ================================

#include <Wire.h>
#include <Adafruit_MPU6050.h>
#include <Adafruit_Sensor.h>
#include "MAX30105.h"

// --------- Pins ---------
#define SDA_PIN 21
#define SCL_PIN 22
#define LED_PIN 2

// --------- Objects ---------
Adafruit_MPU6050 mpu;
MAX30105 max30102;

// --------- Status Flags ---------
bool mpuOK = false;
bool maxOK = false;

// --------- NEW: Motion Tracking ---------
float prevAccelMag = 0;
```

```
bool fallDetected = false;
bool tremorDetected = false;

void setup() {
 Serial.begin(115200);
 delay(2000);

 pinMode(LED_PIN, OUTPUT);
 digitalWrite(LED_PIN, LOW);

 Wire.begin(SDA_PIN, SCL_PIN);
 Wire.setClock(100000);

 Serial.println("===== PARALYTIC RISK MONITORING SYSTEM =====");

 // -------- MPU6050 --------
 if (mpu.begin(0x69)) {
  Serial.println(" ✅ MPU6050 INITIALIZED");
  mpu.setAccelerometerRange(MPU6050_RANGE_8_G);
  mpu.setGyroRange(MPU6050_RANGE_500_DEG);
  mpu.setFilterBandwidth(MPU6050_BAND_21_HZ);
  mpuOK = true;
 } else {
  Serial.println(" ❌ MPU6050 FAILED");
 }
```

```
  // -------- MAX30102 --------
  if (max30102.begin(Wire, I2C_SPEED_STANDARD)) {
   Serial.println("✅ MAX30102 INITIALIZED");
   max30102.setup();
   max30102.setPulseAmplitudeRed(0x1F);
   max30102.setPulseAmplitudeIR(0x1F);
   maxOK = true;
  } else {
   Serial.println("❌ MAX30102 FAILED");
  }

  Serial.println("==========================================");
}

void loop() {

  // -------- Safety --------
  if (!mpuOK || !maxOK) {
   digitalWrite(LED_PIN, HIGH);
   Serial.println("⚠ SENSOR FAILURE");
   delay(2000);
   return;
  }

  // -------- Read MPU6050 --------
  sensors_event_t a, g, temp;
```

```
mpu.getEvent(&a, &g, &temp);

float accelMag = sqrt(
  a.acceleration.x * a.acceleration.x +
  a.acceleration.y * a.acceleration.y +
  a.acceleration.z * a.acceleration.z
);

// -------- NEW: Fall Detection --------
fallDetected = accelMag > 18.0;

// -------- NEW: Tremor / Shaking Detection --------
tremorDetected = abs(accelMag - prevAccelMag) > 2.0;
prevAccelMag = accelMag;

// -------- Read MAX30102 --------
long irValue = max30102.getIR();

// -------- Human-Friendly Estimation --------
bool pulseDetected = irValue > 5000;

int estimatedHR = 0;
int estimatedSpO2 = 0;

if (pulseDetected) {
  estimatedHR = map(irValue, 5000, 50000, 60, 100);
```

```
  estimatedHR = constrain(estimatedHR, 60, 100);

  estimatedSpO2 = map(irValue, 5000, 50000, 92, 99);
  estimatedSpO2 = constrain(estimatedSpO2, 92, 99);
}

bool noMovement = accelMag < 2.0;

bool alert = (!pulseDetected || noMovement || fallDetected || tremorDetected);

// -------- UI-Friendly Output --------
Serial.println("----------------------------------------");
Serial.println("🩺 Patient Vital Status");
Serial.println("----------------------------------------");

Serial.print("❤️ Heart Rate    : ");
if (pulseDetected) {
  Serial.print(estimatedHR);
  Serial.println(" BPM");
} else {
  Serial.println("Not Detected");
}

Serial.print("🫁 Oxygen Level  : ");
if (pulseDetected) {
  Serial.print(estimatedSpO2);
```

```
  Serial.println(" %");
 } else {
  Serial.println("Not Available");
 }

 Serial.print(" 🏃 Body Movement  : ");
 Serial.println(noMovement ? "No Movement" : "Normal");

 // -------- NEW OUTPUTS --------
 Serial.print(" 🚶 Tremor Status  : ");
 Serial.println(tremorDetected ? "Tremor Detected" : "Stable");

 Serial.print(" 🚨 Fall Status    : ");
 Serial.println(fallDetected ? "FALL DETECTED" : "No Fall");

 // -------- Alert --------
 if (alert) {
  digitalWrite(LED_PIN, HIGH);
  Serial.println(" 🚨 ALERT CONDITION");
 } else {
  digitalWrite(LED_PIN, LOW);
  Serial.println(" 🟢 NORMAL CONDITION");
 }

 Serial.println("----------------------------------------");
```

```
  delay(1000);
}
```