# 1 Railway Track Fault Detection System

## 1.1 Overview:

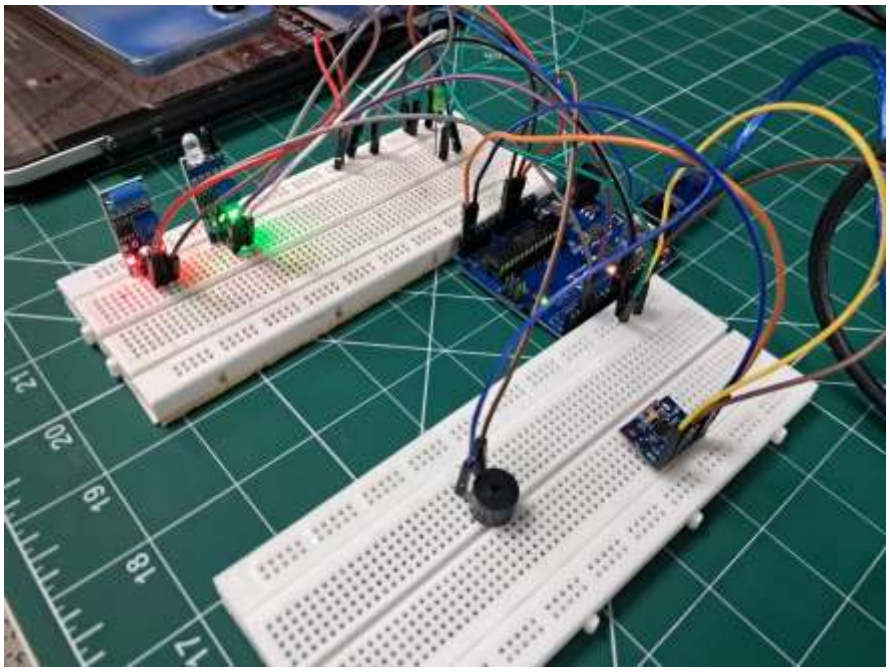
This report explains the Arduino-based Railway Track Fault Detection System that uses three types of sensors:

1. ADXL345 Accelerometer – to detect tilt or displacement of the track.

2. SW-420 Vibration Sensor – to detect abnormal vibrations or shaking.

3. IR Sensor – to detect gaps or obstacles on the railway track.

The system is designed to improve safety by identifying potential track faults in real time and providing alerts through a buzzer and LED.



The system continuously reads sensor data and decides if there is any fault based on the following conditions:

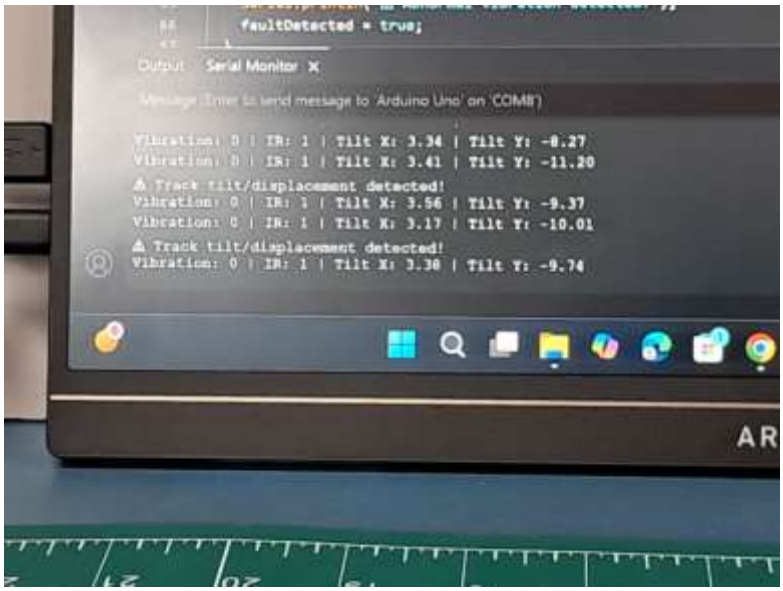
- Abnormal vibration detected by SW-420.
- Excessive tilt detected by ADXL345.
- Obstacle or gap detected by IR sensor

## 1.2 COMPONENTS USED:

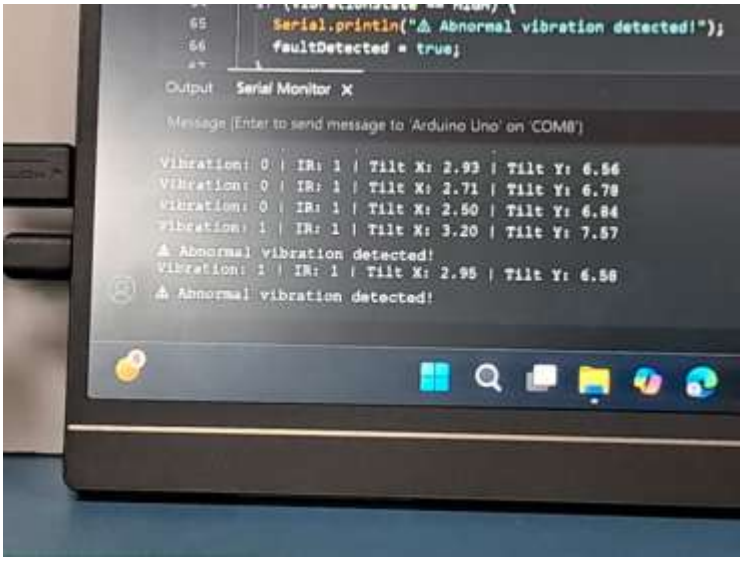
I. Arduino Uno
II. ADXL345 Accelerometer Sensor
III. SW-420 Vibration Sensor
IV. IR Sensor
V. Active Buzzer
VI. LED

## 1.3 SENSORS WORKING:

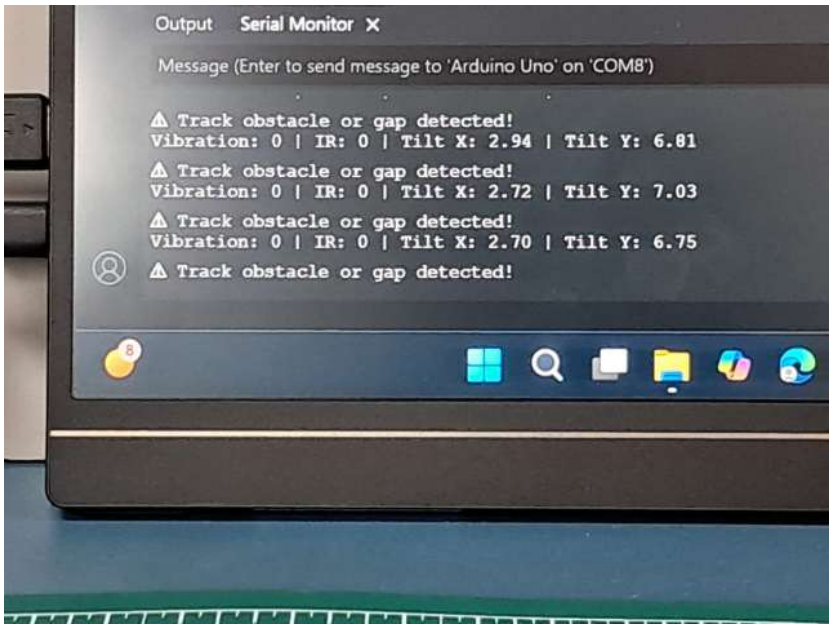
I. **ADXL345 Accelerometer Sensors:** The ADXL345 is a sensor that measures movement and tilt in three directions (X, Y, and Z axes). It uses tiny moving parts inside to detect changes in acceleration, which are then sent to the Board.
**For Railway Application:** When the track moves or tilts, the sensor detects these changes and sends the information to the Board, which then calculates the tilted angles.




II. **SW-420 Vibration Sensor:** This sensor detects sudden or abnormal vibrations, like those caused by a loose rail or a train approaching at high speed.
**For Railway Applications:** Inside the sensor, a spring and metal contact work together to detect vibrations. When the sensor shakes, the spring touches the contact, sending a signal to the system. If the vibrations are too strong, it might mean there's a problem with the rail, like a crack or loose part.




III. **IR Sensor**: This sensor checks for gaps or obstacles on the track using infrared light.
**For Railway Applications:** The sensor sends out infrared light, which bounces back if it hits something. If the light gets blocked (by an obstacle), the sensor sends a signal to the system. This helps detect breaks or objects on the track.




## 1.4 SOURCE CODE:

```
#include <Wire.h>
```

```
#define ADXL345_ADDR 0x53  // I2C address for ADXL345

#define vibrationPin 2
#define irPin 3
#define buzzer 8
#define led 9

// Threshold values (tune as per setup)
const int vibrationThreshold = 200;   // for SW-420 (digital or analog)
const float tiltThreshold = 10.0;    // degrees
int irState = 0;

void setup() {
  Serial.begin(9600);
  Wire.begin();

  // --- Initialize ADXL345 ---
  Wire.beginTransmission(ADXL345_ADDR);
  Wire.write(0x2D);  // Power Control register
  Wire.write(0x08);  // Set the Measure bit to begin measurement
  Wire.endTransmission();

  pinMode(vibrationPin, INPUT);
  pinMode(irPin, INPUT);
```

```
  pinMode(buzzer, OUTPUT);
  pinMode(led, OUTPUT);

  Serial.println("🚆 Railway Track Fault Detection System Initialized (ADXL345 Version)");
  delay(1000);
}

void loop() {
  // --- 1. Read Vibration Sensor ---
  int vibrationState = digitalRead(vibrationPin);

  // --- 2. Read IR Sensor ---
  irState = digitalRead(irPin);

  // --- 3. Read Accelerometer from ADXL345 ---
  int16_t ax, ay, az;
  readAccelerometer(ax, ay, az);

  // Convert to tilt angles (in degrees)
  float angleX = 0, angleY = 0;
  float denomX = sqrt(pow(ay, 2) + pow(az, 2));
  float denomY = sqrt(pow(ax, 2) + pow(az, 2));

  // Avoid division by zero
```

```
if (denomX != 0 && denomY != 0) {
  angleX = atan(ax / denomX) * 57.2958;
  angleY = atan(ay / denomY) * 57.2958;
}

Serial.print("Vibration: "); Serial.print(vibrationState);
Serial.print(" | IR: "); Serial.print(irState);
Serial.print(" | Tilt X: "); Serial.print(angleX, 2);
Serial.print(" | Tilt Y: "); Serial.println(angleY, 2);

bool faultDetected = false;

// --- Fault Conditions ---
if (vibrationState == HIGH) {
  Serial.println("⚠ Abnormal vibration detected!");
  faultDetected = true;
}

if (abs(angleX) > tiltThreshold || abs(angleY) > tiltThreshold) {
  Serial.println("⚠ Track tilt/displacement detected!");
  faultDetected = true;
}

if (irState == LOW) {
  Serial.println("⚠ Track obstacle or gap detected!");
```

```
    faultDetected = true;
  }

  // --- Alarm Section ---
  if (faultDetected) {
    digitalWrite(buzzer, HIGH);
    digitalWrite(led, HIGH);
    delay(1000);
  } else {
    digitalWrite(buzzer, LOW);
    digitalWrite(led, LOW);
  }

  delay(500);
}

// --- Function to Read Data from ADXL345 ---
void readAccelerometer(int16_t &x, int16_t &y, int16_t &z) {
  Wire.beginTransmission(ADXL345_ADDR);
  Wire.write(0x32);  // Data registers starting from 0x32
  Wire.endTransmission(false);
  Wire.requestFrom(ADXL345_ADDR, 6, true);

  if (Wire.available() == 6) {
    x = (Wire.read() | (Wire.read() << 8));
```

```
    y = (Wire.read() | (Wire.read() << 8));
    z = (Wire.read() | (Wire.read() << 8));
  } else {
    x = y = z = 0;  // Prevent NaN by defaulting to 0 if read fails
  }
}
```

### 1.5 CONCLUSION:

This project provides a simple and efficient way to detect potential railway track faults using low-cost sensors. It can help prevent accidents by providing early warnings of abnormalities such as track displacement, vibrations, or obstacles. Future improvements can include GPS modules to send alerts with location data, and SD card logging for maintenance analysis. By using them together, the system can effectively identify potential hazards and alert in real-time, reducing risks and improving maintenance efficiency.