**VIBRATION-BASED MOTOR HEALTH MONITORING SYSTEM USING ESP32 AND ADXL345**

**Mr. S. R. Pandit1, Mr. Kshitij Patil2, Mr. Vedant Vaidya3**

1Assistant Professor, Department of Instrumentation and Control, MVPS’s KBTCOE, Nashik, Maharashtra, India

2Student, Department of Instrumentation and Control, MVPS’s KBTCOE, Nashik, Maharashtra, India

3 Student, Department of Instrumentation and Control, MVPS’s KBTCOE, Nashik, Maharashtra, India

**ABSTRACT**

This paper presents a low-cost wireless vibration monitoring system utilizing an ESP32 microcontroller and an ADXL345 accelerometer. The system is designed to continuously monitor the vibration of a motor and calculate the vibration magnitude using real-time sensor data. When the measured vibration exceeds a predefined threshold, the ESP32 microcontroller sends an alert notification directly to the user's mobile phone via the Telegram messaging platform. This functionality serves as a simple yet effective predictive maintenance tool, enabling early detection of mechanical issues such as imbalance or wear. The system is easy to install, cost-effective, and scalable, making it well-suited for use in industrial settings, workshops, and academic environments.

**Keywords:** Accelerometer, ESP32, ADXL345, Telegram notification, IOT, Preventive maintenance

**INTRODUCTION**

Motor vibration analysis is a critical aspect of predictive maintenance in industrial and automation settings. Excessive vibration can be an early indicator of mechanical faults such as imbalance, misalignment, or bearing failure. Traditional vibration analysis systems are often expensive and difficult to implement in smaller setups. This paper proposes a compact, wireless vibration monitoring system using ESP32 and ADXL345 to simplify fault detection. The system uses real-time data acquisition, threshold-based analysis, and wireless alerts via Telegram to ensure timely fault reporting.

1. **METHODOLOGY**

The monitoring system is centered around the ADXL345 accelerometer, which is capable of measuring acceleration in three axes (X, Y, and Z) with high sensitivity. The sensor communicates with the ESP32 via the I²C protocol, with SDA and SCL lines connected to GPIO21 and GPIO27 respectively. The CS pin of the ADXL345 is grounded to enable I²C mode, and the SDO pin is also grounded to select the default device address.The ESP32 reads acceleration data from the ADXL345 at fixed intervals. The individual axis values are used to calculate the total vibration magnitude using the Euclidean norm formula:

**V = √(x² + y² + z²)**

This formula combines the three-axis data into a single vibration magnitude value, expressed in meters per second squared (m/s²). A user-defined threshold value, experimentally determined to be 11.0 m/s², is used as the decision point. If the calculated vibration magnitude exceeds this threshold, the ESP32 sends a notification through Telegram using the UniversalTelegramBot library. This wireless alerting mechanism ensures that users are promptly informed of potential motor issues.

**2.1 Hardware Connection**

|  |  |  |
| --- | --- | --- |
| **ADXL345 Pin** | **ESP32 GPIO Pin** | **ESP32 Pin Name** |
| **VCC** | 3.3V | 3V3 |
| **GND** | GND | GND |
| **SDA** | GPIO21 | SDA |
| **SCL** | GPIO27 | SCL |
| **CS** | GND | - |
| **SDO** | GND | - |

**Table 1:** Connections

**2.2 Software Implementation**

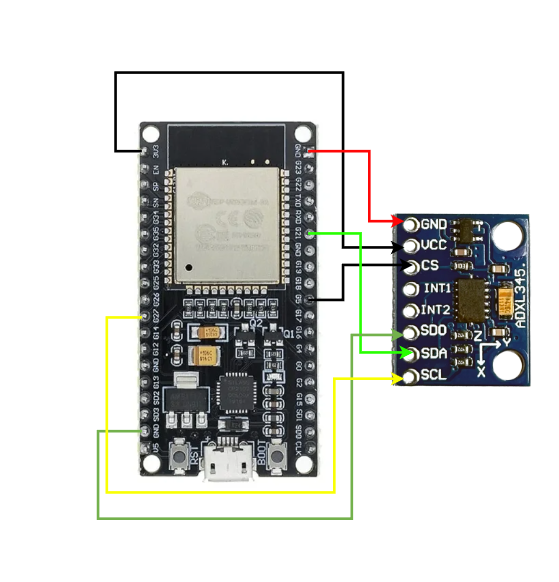
The ESP32 was programmed using Arduino IDE. Libraries used include WiFi.h, UniversalTelegramBot.h, Adafruit\_ADXL345\_U.h, and ArduinoJson.h. The code initializes the Wi-Fi connection, sets up the ADXL345 sensor, reads acceleration values continuously, and calculates the total vibration. If the vibration exceeds the safe limit, an alert is sent using the Telegram Bot API.

1. **MODELING AND ANALYSIS**

**Figure 2:**ADXL345 accelerometer

**Figure 1:** ESP32

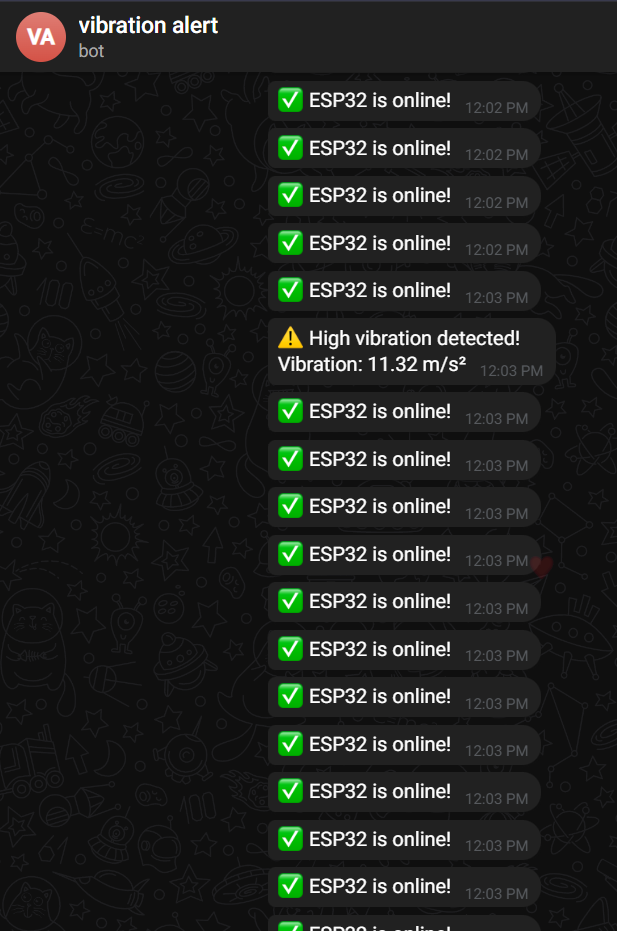
.

****

**Figure 3:** Circuit diagram

**Figure 3:**Circuit Diagram

**3.RESULTS AND DISCUSSION**

****The vibration monitoring system was successfully deployed and tested. It provided accurate measurements and consistent notifications. A 1-second loop interval was ideal for real-time monitoring without spamming the Telegram API. The project demonstrated that low-cost components could effectively replicate more expensive predictive maintenance tools.

**Figure 4:** Telegram app notifications

1. **CONCLUSION**

This study demonstrates the feasibility of a low-cost, wireless vibration monitoring system using ESP32 and ADXL345. It enables early fault detection, reduces unplanned downtimes, and improves motor reliability. The use of Telegram for notifications provides a simple and user-friendly interface for real-time alerts. Future work can include logging data to the cloud and integrating machine learning for smarter fault classification.

1. **REFERENCES**
2. Adafruit Industries. "ADXL345 Datasheet." https://cdn-shop.adafruit.com/datasheets/ADXL345.pdf
3. T. Mohana Priya, Dr. M. Punithavalli & Dr. R. Rajesh Kanna, "Machine Learning Algorithm for Development of Enhanced Support Vector Machine Technique to Predict Stress," Global Journal of Computer Science and Technology: C Software & Data Engineering, Volume 20, Issue 2, 2020, pp 12-20.
4. Espressif Systems. "ESP32 Technical ReferenceManual." [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32\_technical\_reference\_manual\_en.pdf