RESEARCH ARTICLE OPEN ACCESS

AI–ASSISTED MOTOR FAULT PREDICTION AND PROTECTION USING ESP32

Nandini R. Patil*, Prerna S. Wakhare, Sanskruti V. Deore, Uma Maheshwarrao, Dr. Prajakta Shirke
(School of Computer Science and Engineering, Sandip University, Trimbak Road, Nashik, Maharashtra, 422213, India)
Email: nandinirpatil21@gmail.com
*Corresponding author: Nandini R. Patil

****

Abstract

Motor monitoring and protection are essential in industrial and agricultural environments where continuous operation is critical. Motor failures caused by overheating, overcurrent, voltage fluctuations, and excessive vibration can result in significant downtime and financial losses. Conventional protection systems are typically limited to basic relay-based mechanisms or rely on costly PLC-based configurations. This paper proposes a low-cost IoT-based smart motor monitoring and protection system using the ESP32 microcontroller. The system continuously measures voltage, current, temperature, and vibration parameters of a DC 775 PMDC motor. The ESP32 serves as both a data acquisition unit and a standalone web server, hosting a real-time dashboard accessible via WiFi. A threshold-based protection algorithm identifies abnormal operating conditions and automatically disconnects the motor through a relay module. The system operates independently of cloud infrastructure, ensuring reduced latency and improved reliability. Experimental evaluation demonstrates accurate sensor readings, rapid response time, and effective fault detection. The proposed prototype provides an economical and scalable solution for real-time motor protection and preventive maintenance applications.

Keywords — **IoT, Motor Monitoring System, ESP32, Fault Detection, Real-Time Monitoring, Embedded Systems, Motor Protection, Smart Maintenance, Standalone Web Server**.

****

I. INTRODUCTION

Electric motors are widely used in industrial, agricultural, and domestic applications due to their efficiency and reliability. They are essential in equipment such as pumps, compressors, conveyor systems, and automated machinery. However, continuous operation under varying load conditions makes them vulnerable to faults like overheating, overcurrent, voltage imbalance, and excessive vibration, which can reduce efficiency, shorten lifespan, and cause unexpected downtime..Traditional motor protection systems mainly rely on electromechanical relays and manual inspections. While these methods provide basic safety, they do not support real-time monitoring or remote access. Advanced PLC-based systems offer better risks and challenges. Digital features such as Twitter control and monitoring features, but they are costly and require complex setup and maintenance, making them less practical for small-scale industries and agricultural users. To overcome these limitations, this research proposes a standalone IoT-based smart motor monitoring and protection system using the ESP32 microcontroller. The system continuously monitors voltage, current, temperature, and vibration parameters and displays real-time data through a web dashboard. A threshold-based protection algorithm automatically disconnects the motor during abnormal conditions, providing a cost-effective and reliable solution for preventive maintenance.

II. LITERATURE REVIEW

Motor monitoring and protection have been widely studied using various technologies and methods. In the past, protection systems were mainly based on electromechanical relays that helped prevent overcurrent and short-circuit issues. These systems are simple and dependable, but they cannot monitor motor health in real time or allow remote supervision. Many researchers later suggested PLC-based systems to improve reliability and control. These systems can track parameters such as current, voltage, and temperature and initiate protective actions when faults occur. However, PLC-based solutions are costly, require skilled personnel for setup and maintenance, and are not suitable for low-budget or small-scale applications.

With advancements in embedded systems, microcontroller-based motor protection systems have become more common. These systems use sensors to monitor electrical and mechanical conditions and provide local protection at a lower cost. However, they often offer limited user interaction and lack real-time data visualization features. More recently, IoT-based motor monitoring systems have gained attention, enabling wireless data transmission, real-time dashboards, and remote access through web or mobile applications. Many of these systems rely on cloud platforms for data storage and predictive maintenance, but cloud dependency introduces latency, recurring costs, security concerns, and reliance on stable internet connectivity.

Most existing motor monitoring systems are either expensive due to PLC implementation or dependent on cloud infrastructure for processing and visualization. Therefore, there is a need for a low-cost, standalone IoT-based system that provides real-time data visualization, accurate fault detection, and local control without relying on external cloud services.

III. SYSTEM ARCHITECTURE

A. Hardware Design

The hardware components used in the proposed system are as follows:

- **ESP32 Microcontroller** – Acts as the main processing unit of the system. It reads sensor data, processes the information, hosts the web server, and controls the relay module. It also provides built-in WiFi connectivity for wireless communication.

- **ACS712 Current Sensor** – Measures the current flowing through the motor and provides a proportional analog voltage output for detecting overcurrent conditions.

- **Voltage Divider Circuit** – Reduces the motor voltage to a safe level suitable for the ESP32 ADC input, enabling under-voltage and over-voltage monitoring.

- **DHT11 Temperature Sensor** – Measures the temperature near the motor to detect overheating conditions.

- **SW-420 Vibration Sensor** – Detects abnormal vibrations that may indicate mechanical faults such as imbalance or misalignment.

- **Relay Module** – Controls the motor power supply by switching the motor ON or OFF when a fault condition is detected.

- **DC 775 (12V) PMDC Motor** – Serves as the test motor for monitoring and fault simulation in the proposed system.

B. Software Design

The software for the proposed system is developed using Python in the Arduino IDE environment. The ESP32 performs both monitoring operations and web server functionalities.

- **Data Acquisition:** Sensor data are collected at regular intervals through a combination of analog and digital inputs of the ESP32.

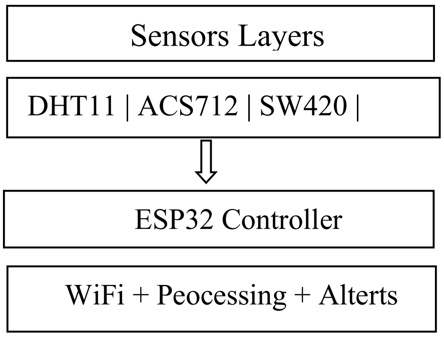
- **Fault Detection Algorithm:** The acquired sensor values are continuously compared with predefined threshold limits. If any parameter exceeds the safe operating range, the system identifies it as a fault condition.

- **Standalone Web Server:** The ESP32 operates as a standalone web server by creating its own WiFi network and hosting a web page. This enables users to access real-time motor data and system status.

• **Real-Time Communication:** JavaScript is implemented on the web interface to update sensor readings dynamically without requiring page refresh, ensuring smooth real-time monitoring.

• **Motor Control Logic:** The motor is controlled through the relay module based on detected fault conditions or user commands from the web interface.

1.6 System Architecture



IV. METHODOLOGY

The proposed methodology begins with continuous data acquisition from sensors, followed by signal processing and fault identification using programmed logic. The processed information is then displayed on a web interface to enable real-time monitoring and control actions.

A. Sensor Data Acquisition

The ESP32 continuously collects data from the connected sensors at predefined time intervals.

• The ACS712 current sensor provides an analog voltage proportional to the motor current.
• The voltage divider circuit scales down the motor voltage to a safe level for ESP32 measurement.
• The DHT11 sensor provides digital temperature region.
• The SW-420 sensor detects abnormal vibration levels.
• Analog signals are read through the ESP32's Analog-to-Digital Converter (ADC), while digital signals are processed directly through digital input pins.

B. ADC Conversion and Calibration

The ESP32 converts analog sensor signals into digital values using its built-in ADC. Calibration is performed to ensure measurement accuracy and reliability.

The motor current is calculated using the following formula:

I = |V_measured − V_offset| / Sensitivity

Where:
• V_measured = Voltage measured from the current sensor
• V_offset = Baseline voltage of the sensor
• Sensitivity = Sensor-specific constant

Voltage values are calculated using the standard voltage divider equation to determine the actual motor voltage.

C. Fault Detection Algorithm

The system continuously compares sensor readings with predefined safety thresholds.

• If Voltage < V_min → Under-voltage fault
• If Voltage > V_max → Over-voltage fault
• If Current > I_max → Overcurrent fault

• If Temperature > T_max → Overheating fault
• If Vibration exceeds threshold → Mechanical fault

Upon detection of any fault condition, the system generates an alert and activates the relay mechanism to disconnect the motor.

D. Relay Control Logic

When a fault condition is identified, the ESP32 deactivates the relay to disconnect the motor from the power supply. The system also allows manual motor control through the web interface for operational flexibility.

E. Real-Time Web Communication

The ESP32 operates as a standalone web server by creating a WiFi hotspot. The web interface is developed using HTML and CSS, while JavaScript enables dynamic data updates without page refresh. This ensures smooth real-time monitoring and remote motor control without dependence on cloud services.

V. RESULT AND DISCUSSION

The proposed IoT-based motor monitoring and protection system was evaluated under various normal and fault operating conditions to assess its performance, reliability, and response time. Fault scenarios such as overcurrent, overheating, voltage fluctuations, and excessive vibration were intentionally simulated to verify the effectiveness of the protection mechanism.

A. Normal Operating Condition

Under normal operating conditions, all sensor parameters remained within the predefined safe limits. The web dashboard displayed real-time voltage, current, temperature, and vibration readings without generating any fault alerts. The relay module remained in the ON state, allowing uninterrupted motor operation. The system demonstrated stable performance and accurate real-time data visualization during normal conditions.

B. Fault Condition Testing

1. Overcurrent Condition

The motor shaft was blocked slightly to mimic an overload. The current went above the set limit (Imax). The system found this problem and turned off the relay automatically. The dashboard showed an “Overcurrent Alert” message.

2. Overheating Condition

Heat was applied near the motor case. When the temperature went past the maximum limit (Tmax), the system gave an overheating alert and disconnected the motor.

3. High Vibration Condition

Some mechanical disturbance was added to simulate an imbalance. The vibration sensor found the unusual movement, and the dashboard showed a vibration alert.

B. Response Time Analysis

The system responded quickly, taking about 1–2 seconds from detecting a fault to turning off the relay. This fast response helps protect the motor and prevents damage.

C. Performance Evaluation

The system was able to detect all the simulated fault conditions accurately. The dashboard updated smoothly and stably when connected to a WiFi hotspot. Because the system is standalone, it doesn’t need cloud services, which saves time and money. Overall, the test results show that the system is reliable, monitors in real time, and provides good protection for small industrial and agricultural settings.

VI. CONCLUSION

This paper describes the design and implementation of a low-cost IoT-based smart motor monitoring and protection system using the ESP32 microcontroller. The system continuously monitors key motor parameters such as voltage, current, temperature, and vibration to ensure safe and reliable operation. A threshold-based protection algorithm is employed to detect abnormal operating conditions and automatically disconnect the motor using a relay module.

Parameter	**Threshold Limit**	**Measured value**	**Status**
Voltage	10- 14 V	12.1 V	Normal
Current	≤ 1.5A	0.82A	Normal
Temperature	≤ 50°C	32°C	Normal
Vibration	Low	Low	Normal

The system functions as a standalone web server and does not require external cloud services. It provides real-time dashboard updates through WiFi connectivity, enabling continuous monitoring and control. Experimental results demonstrate accurate fault detection, fast response time, and reliable performance under various simulated fault conditions.

The proposed solution is cost-effective, scalable, and suitable for small-scale industrial and agricultural applications. With future enhancements such as cloud integration and AI-based predictive analysis, the system can evolve into a more advanced smart maintenance platform.

ACKNOWLEDGMENT

The authors wish to express their sincere gratitude to Prof. Dr. Prajakta Shirke, School of Computer Science and Engineering, Sandip University, Nashik, India for his invaluable guidance and support throughout this research work.

The authors also thank Sandip University, Nashik, India for providing the academic environment and resources essential to the completion of this study.

REFERENCES

[1] J. Feukeu and M. Mbuyu, "Design and Implementation of an IoT-Based Induction Motor Protection and Control System Using Arduino and ESP32," *Journal of Electrical Systems*, vol. 21, no. 1, 2025.

[2] A. Adams, R. Putra, and D. Santoso, "Design and Build an IoT-Based 3-Phase Electric Motor Protection System," *Journal of Electrical Engineering, Electronics, Control, and Computer Science (JEEMECS)*, 2025.

[3] M. Rizqi ad S. Hartoyo, "Design and Development of an IoT-Based Electrica Disturbance Protection and Load Monitoring System Using ESP32," *Journal of Advanced Electrical Engineering and Informatics (JAEEI)*, 2025.

[4] N. Rostan, M. Ismail, and A. Rahman, "Motor Vibration and Temperature Detector Using IoT," *International Journal of Synergy in Engineering and Technology*, vol. 4, no. 2, 2024.

[5] M. Musyafiq, A. Prasetyo, and H. Wijaya, "IoT-Based Control, Monitoring, and Protection System for 3-Phase Induction Motors," *Journal of Power, Energy, and Control*, 2024.

[6] K. Mamatha and H. Thejaswi, "Induction Motor Condition Monitoring and Controlling Based on IoT," *International Journal of Research in Engineering, Science and Management (IJRESM)*, vol. 4, no. 6, pp. 120–124, 2021.