www.ijarp.com ISSN 2456-9992 Page: 01-07

Research Article

International Journal Advanced Research Publications

Volume: 02
Issue: 11

IOT-BASED MACHINE HEALTH MONITORING SYSTEM

Y. Abhishek[1], M. Bhaskara Rao[1], G. Jaswanth,[1] [2*]Dr. Prashant Kumar Choudhary

[1]Department of Mechanical Engineering, GMR Institute of Technology, Rajam, Andhra Pradesh, India.

[2]Assistant Professor, Department of Mechanical Engineering, GMR Institute of Technology, Rajam, Andhra Pradesh, India.

Article Received: 02 October 2026, Article Revised: 21 October 2026, Published on: 11 November 2026

***Corresponding Author: Dr. Prashant Kumar Choudhary**

Assistant Professor, Department of Mechanical Engineering, GMR Institute of Technology, Rajam, Andhra Pradesh, India.

Doi: https://doi-doi.org/101555/ijarp.4301

ABSTRACT

Industrial machines are required for uninterrupted manufacturing activities; however, any kind of malfunction may lead to stoppages, high maintenance costs and shortened lives of machines. The current paper focuses on IOT-enabled machine health monitoring system that will allow for temperature and vibration monitoring in real time. In the process, the system is equipped with DS18B20 temperature sensor and SW-420 vibration sensor that will provide machine-related information. An ESP32 microcontroller processes sensor signals and compares them against pre-defined threshold values in order to categorize machine conditions into Healthy, Warning and Fault categories. The status of machines is displayed in real time by means of an OLED display; at the same time, LEDs and buzzer produce alarm signals in case of abnormal conditions. Leveraging the Wi-Fi capability embedded in ESP32 microcontroller, the system allows sending the monitoring information to the IOT cloud service – Thing Speak. The proposed system allows for efficient condition monitoring and helps recognize abnormal machine operation conditions.

KEYWORDS: Internet of Things (IoT), Machine Health Monitoring, ESP32, DS18B20, SW-420, Vibration Sensor, Temperature Sensor, ThingSpeak, Condition Monitoring, Predictive Maintenance.

1. INTRODUCTION

Industrial machines operate under varying mechanical and thermal conditions. As a result, while in operation, there occur vibrations and heating of the machine parts. Any changes in the above parameters can signify that there is something wrong with the machine such as too much friction, misalignment, imbalance, or overloading of the machine. Failure to detect the above parameters during the early stages will lead to machine failure, break down, production downtime and higher maintenance cost. Conventional maintenance relies on periodic inspections or corrective measures once there is a fault in the machine. Periodic inspection may fail to diagnose the problem occurring between the inspections while corrective measures may cause high repair costs and unplanned shut down of the machine. There is Internet of Things technology that connects sensors, embedded controllers, and cloud platform to enable continuous monitoring and data acquisition. In the designed system, the DS18B20 sensor is used to measure temperature while the SW-420 sensor is used to detect vibrations. The ESP32 will serve as the central controller, analyze sensor data, and transfer monitoring data via Wi-Fi to the cloud platform Thing Speak. Local monitoring is done using the OLED screen, LED, and buzzer. The major aim of this study is to design a cost-effective monitoring system that is capable of detecting any unusual operating condition in the machine.

2. OBJECTIVES OF THE PROJECT

1. To monitor machine temperature and vibration in real time.
2. To identify abnormal machine conditions at an early stage.
3. To use an ESP32 microcontroller for sensor-data collection and processing.
4. To classify machine health into Healthy, Warning and Fault conditions.
5. To provide immediate local alerts using an OLED display, LED and buzzer.
6. To transmit machine-monitoring data to an IoT cloud platform through Wi-Fi.
7. To enable remote monitoring of machine condition using ThingSpeak.
8. To support early fault identification, reduce unexpected downtime and improve maintenance planning.

3. LITERATURE REVIEW

The project presentation identifies existing work related to IoT-based machine monitoring and intelligent fault diagnosis. The following summary is based on the literature details supplied in the presentation.

S. No.	Title / Publication	Methodology / Parameters	Inference / Limitation
1	IoT-Based Real-Time Vibration and Temperature Monitoring System for Industrial Machinery Using ESP32 and MQTT, International Journal of Engineering Continuity (2026).	ESP32, vibration sensor, temperature sensor, MQTT, Wi-Fi communication, cloud monitoring and real-time data visualization.	Supports real-time monitoring and remote access. The presentation notes that it focuses mainly on condition monitoring and does not include predictive fault analysis.
2	"ESP32-Based Machine Condition Monitoring Using ThingSpeak," IJARSCT (2024).	ESP32, vibration sensor, temperature sensor, Wi-Fi and ThingSpeak for data logging and visualization.	Provides an economical cloud-based monitoring approach, but the presentation notes limited intelligent fault prediction and advanced analytics.
3	"AI-Powered Fault Diagnosis and Estimation of Remaining Useful Life Using IoT Framework," IEEE Access (2026).	IoT sensors, artificial intelligence / machine learning, cloud computing, fault diagnosis and remaining useful life estimation.	Can support fault prediction and remaining-useful-life estimation, but requires larger datasets, computational resources and more complex models.

From the reviewed work, IoT-based monitoring can make machine-condition information available remotely. The proposed project focuses on a comparatively simple arrangement using temperature and vibration inputs, an ESP32 controller, local alerts and ThingSpeak-based monitoring. The supplied presentation does not provide full author names, volume numbers, page ranges or DOI details for the listed papers; these bibliographic details should be verified before formal publication.

4. METHODOLOGY

The proposed methodology follows a sensor–controller–display–cloud arrangement. Temperature and vibration are selected as the main machine-condition parameters. The sensors are installed at suitable locations on the machine and connected to the ESP32. The controller acquires the readings, evaluates them against predefined threshold conditions and assigns a machine-health status. The status is indicated locally and the available data are transmitted to the cloud through Wi-Fi.

4.1 System Components and Working Principles

Component	Function and working principle
Machine under monitoring	Acts as the source of the monitored parameters. Mechanical components generate heat and vibration during operation; changes may indicate abnormal operating conditions.
DS18B20 temperature sensor	A digital temperature sensor using the 1-Wire communication protocol. It measures temperature at the mounting location and transmits digital data to the ESP32 through its data line.
SW-420 vibration sensor	A low-cost vibration and shock detection module. Its vibration-sensitive element changes electrical state in response to vibration or shock, and the module comparator produces a digital output.
ESP32 microcontroller	Acts as the central processing unit. It receives temperature and vibration inputs, compares them with predefined thresholds and determines the machine condition as Healthy, Warning or Fault.
OLED display	Provides a local visual indication of machine readings and operating condition. It communicates with the ESP32 through a display communication interface.
LED and buzzer	Provide visual and audible alerts when abnormal temperature or vibration conditions are detected.
Wi-Fi communication	The ESP32's integrated Wi-Fi capability is used to send the processed monitoring information to the IoT cloud platform.
ThingSpeak IoT cloud	Receives, stores, displays and supports analysis of the transmitted sensor data. Time-based graphs can be used to observe changes in machine parameters.

4.2 Operating Procedure

1. System initialization: The ESP32 starts and initializes the connected sensors, display and communication interfaces.

2. Sensor data acquisition: The DS18B20 measures machine temperature and the SW-420 provides a vibration-status signal.

3. Data processing: The ESP32 receives the sensor inputs and evaluates them using the programmed logic.

4. Threshold comparison: The acquired values or sensor states are compared with predefined threshold conditions.

5. Condition classification: Based on the comparison, the machine status is categorized as Healthy, Warning or Fault.

6. Local indication: The OLED displays the available readings and status. The LED and buzzer provide alerts when abnormal conditions are identified.

7. IoT data transmission: The ESP32 sends the monitoring information to ThingSpeak through Wi-Fi.

8. Remote monitoring: The user views the available temperature, vibration and machine-status information through the cloud platform.

4.3 IoT Communication and Remote Monitoring

The ESP32 includes an integrated Wi-Fi facility that enables wireless communication with an IoT cloud platform. After acquiring and processing sensor information, the controller can transmit parameters such as temperature, vibration status and machine condition to ThingSpeak. The platform can store and visualize the incoming data using time-based graphs. This arrangement allows a user to monitor machine condition from a mobile phone or computer without being physically present beside the machine. The supplied presentation identifies ThingSpeak as the cloud platform but does not specify the exact data-transfer protocol, update interval or cloud-channel configuration.

5. MODELING AND SYSTEM DESIGN

The system is designed around an ESP32-based embedded control unit. The DS18B20 and SW-420 act as input devices. The ESP32 processes the incoming information and controls the local output devices, namely the OLED display, LED and buzzer. The Wi-Fi interface connects the controller to ThingSpeak for remote monitoring.

The overall functional sequence is: machine operation → temperature and vibration sensing → ESP32 data collection → data processing and threshold comparison → Healthy / Warning / Fault classification → OLED, LED and buzzer indication → Wi-Fi communication → ThingSpeak cloud → remote monitoring.

5.1 System Flow

1. Start and initialize the ESP32.
2. Read temperature and vibration sensor inputs.
3. Process the sensor data.
4. Compare the inputs with predefined threshold conditions.
5. Determine whether the machine is Healthy, in Warning condition or in Fault condition.
6. Update the OLED display and operate the LED / buzzer according to the detected condition.
7. Transmit the monitoring data to ThingSpeak through Wi-Fi.
8. Continue monitoring and repeat the process.

6. RESULTS AND DISCUSSION

The project presentation defines a system intended to monitor machine temperature and vibration, process the sensor information using an ESP32, classify machine condition and provide local and remote indications. The intended status categories are Healthy, Warning and Fault. The OLED display provides a local indication, while the LED and buzzer alert the operator to abnormal conditions. Wi-Fi communication and ThingSpeak are included to support remote viewing of the machine data.

The supplied presentation does not provide measured temperature readings, vibration test data, threshold values, detection accuracy, cloud update interval, alert response time, power consumption or field-test results. Therefore, numerical performance claims cannot be made from the available project material. These values should be measured using a working prototype under repeatable test conditions.

Parameter	Suggested validation method	
Temperature measurement	Compare the DS18B20 reading with a calibrated reference thermometer at selected operating conditions.	
Vibration response	Apply repeatable vibration conditions and record the SW-420 output / response.	
Condition classification	Test the programmed Healthy, Warning and Fault logic using controlled input conditions.	
Local alert operation	Check OLED, LED and buzzer response for each defined machine condition.	
Cloud data transmission	Record whether sensor data are received and displayed correctly on ThingSpeak.	
Update interval and delay	Measure the time between sensor acquisition, cloud update and local alert.	
Power consumption	Measure current consumption during normal monitoring and alert conditions.	

The validation tests above are recommended measurements, not results already demonstrated in the supplied presentation. The final experimental paper should replace “To be measured” with actual observations obtained from the prototype.

7. CONCLUSION

An IoT-based machine health monitoring system using an ESP32 microcontroller has been proposed to monitor machine temperature and vibration in real time. The DS18B20 temperature sensor and SW-420 vibration sensor provide the input information, which is processed by the ESP32 using predefined threshold conditions. The machine condition is intended to be classified as Healthy, Warning or Fault. An OLED display, LED and buzzer

provide local indications, while Wi-Fi communication enables data transmission to the ThingSpeak cloud platform for remote monitoring. The proposed arrangement offers a simple approach to observing machine operating conditions and supporting early identification of abnormal behaviour. Further work should implement and test the complete prototype, finalize threshold values and report measured results for sensor response, classification, cloud communication, alert delay and power consumption.

8. REFERENCES

1. "IoT-Based Real-Time Vibration and Temperature Monitoring System for Industrial Machinery Using ESP32 and MQTT," International Journal of Engineering Continuity, 2026. (Bibliographic details to be verified.)
2. "ESP32-Based Machine Condition Monitoring Using ThingSpeak," International Journal of Advanced Research in Science, Communication and Technology (IJARSCT), 2024. (Authors and publication details to be verified.)
3. "AI-Powered Fault Diagnosis and Estimation of Remaining Useful Life Using IoT Framework," IEEE Access, 2026. (Authors, volume, pages and DOI to be verified.)