International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Diagnostic Self-Monitoring System Using ESP32

Mugdha Kamble[1], Siddhi Ghone[2], Siddhi Vaidya[3], Rugved Golande[4] ,Trupti Dudhat [5]

Vidyalankar

ABSTRACT:

The goal is to create a diagnostic system for the ESP32 that can monitor, detect, and prevent hardware issues in real time. This system will track critical parameters such as voltage, current, temperature, communication dropouts, component failures, and signal integrity, providing proactive alerts to ensure smooth operation. It will autonomously log faults and report any issues, preventing potential system failures. By leveraging predictive algorithms, the system can forecast potential hardware issues based on historical data, allowing for timely maintenance. Additionally, the diagnostic solution will feature remote access via a cloud API, enabling real-time monitoring and remote troubleshooting capabilities. The outcome is a reliable, self-diagnostic system that extends the lifespan of hardware and minimizes downtime, ensuring optimal performance over time

Keywords: Diagnostic Monitoring, IoT, ESP32, Real-Time Analysis, Preventive Maintenance

Introduction:

The objective is to design an advanced diagnostic system for the ESP32 that monitors hardware health, performs predictive maintenance, and enables remote diagnostics. The system will continuously track critical components such as power supply, sensor status, communication modules, and memory to detect issues like voltage fluctuations, component failures, and communication dropouts. It will feature automatic fault identification, logging, and reporting capabilities to proactively address hardware issues before they lead to system failures. A remote diagnostics interface will be integrated, allowing users to access diagnostics via a cloud-based API, enabling them to run tests, retrieve logs, and troubleshoot hardware remotely without physical access to the ESP32. The technical goals are to develop a comprehensive health monitoring suite, integrate predictive maintenance algorithms, and implement a secure cloud-accessible API for real-time diagnostics and remote control.

Component Used:

1.ESP32Microcontroller
Acts as the brain of the system, handling communication and data processing.

2. MPU6050 Module
Monitors acceleration and gyroscopic data for motion tracking and anomaly detection.

3. ACS712 Current Senso
Tracks power consumption and current flow, ensuring electrical safety.

4. Voltage Sensor
Measures voltage levels, allowing for early detection of faults.

5. Temperature Senso
Monitors temperature variations to avoid overheating or failures.

6. MFRC522 Module
Enables RFID-based identification for secure device access or tracking.

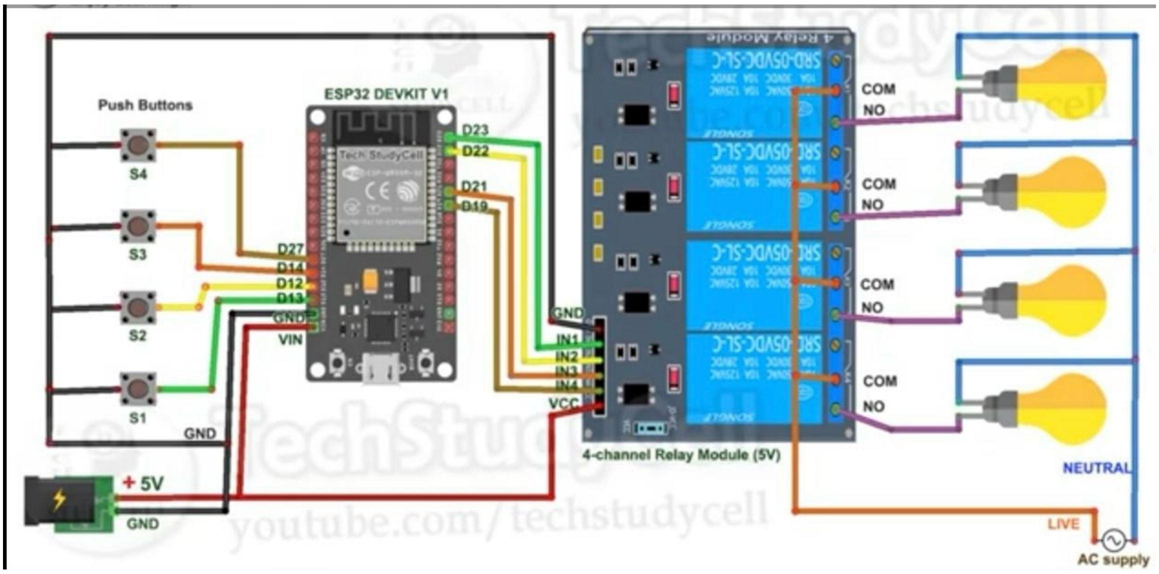
7. Micro SD Card Adapter
Provides local storage for diagnostic data backups.

8.L298N Motor Driver

Manages power delivery to connected actuators like a fan for temperature control.

9. 12V Fa

Ensures efficient cooling during operations.



Software & Tools :

1. Programming Environment

- Arduino IDE: For ESP32 coding and integration with sensors.

2. Web/Mobile Interface

- Designed for real-time monitoring, using HTML, CSS, and JavaScript.

3. Database Integration

- Data logging and visualization via cloud services like Firebase or a local database using XAMPP.

Methodology :

1. Data Collection

- Sensors connected to the ESP32 gather real-time data, such as motion, temperature, voltage, and current levels.

2. Processing

- The ESP32 processes the collected data, identifying trends or anomalies through pre-defined thresholds.

3. Storage

- Data is logged locally on an SD card and optionally synced to a cloud database for remote access.

4. User Interaction

- Users access the system via a responsive web or mobile app interface for real-time data visualization and diagnostics.

5. Notification System

- Alerts are sent to the user in case of anomalies detected, through email or app notifications.

Additional Features:

1. Google Assistant Integration

- Allows users to interact with the system using voice commands for seamless and hands-free operation.
- Example commands include turning on devices, querying real-time data, or setting up alerts.

2. Sinric Pro Integration

- Facilitates smart home connectivity and remote control of devices.
- Ensures secure communication and synchronization with IoT devices like fans, sensors, and more.

Applications:

1. Home Automation

- The system enhances smart home functionality by enabling automated control of appliances such as lights, fans, and temperature-regulating devices.
- Integrating Google Assistant and Sinric Pro provides hands-free control and personalized settings.

2. Digital Twin Integration (Planned)*

- A digital twin of the system will offer a virtual representation, enabling advanced simulations and real-time monitoring.
- Users can visualize data and predict potential failures, enhancing the system's diagnostic capabilities.

End-User Benefits:

Real-time Monitoring: Track critical parameters of the ESP32 and the entire deployed application in real-time, ensuring optimal performance.
Early Fault Detection & Anomaly Prediction: Detect faults and predict anomalies early, reducing the risk of system failures and minimizing downtime.
Remote Debugging: Troubleshoot and resolve issues remotely, eliminating the need for physical access to devices and saving time and resources.
Improved Reliability & Longevity: Proactively address issues before they escalate, enhancing the system's reliability and extending its operational lifespan.

Results:

The system ensures accurate diagnostics, identifying faults and generating actionable insights. It proves especially effective in:

- Monitoring device health.
- Preventing overheating and power inefficiencies.
- Logging long-term performance data for predictive maintenance.

Conclusion:

This research introduces a comprehensive diagnostic system that integrates real-time monitoring of ESP32 alongside a digital twin framework representing the entire deployed application. By extending digital twin capabilities beyond the microcontroller to encompass the full system, we enable holistic real-time monitoring, predictive diagnostics, and enhanced operational insights for embedded and IoT applications.

The proposed system mirrors the physical deployment in a virtual environment, ensuring real-time synchronization of all critical parameters, including GPIO states, sensor data, network interactions, and system-wide performance metrics. This enables early fault detection, anomaly prediction, and remote debugging, significantly improving reliability and maintainability. The integration of cloud-based analytics further enhances the system's scalability, enabling multi-device monitoring and adaptive fault response mechanisms.

Our approach demonstrates the potential of digital twins in embedded system diagnostics, offering a paradigm shift from reactive troubleshooting to proactive, AI-assisted system maintenance. Future advancements may include deep learning-driven anomaly detection, automated corrective actions, and integration with edge computing frameworks for real-time decision-making. This research sets a strong foundation for the next generation of intelligent, self-diagnosing embedded systems, revolutionizing real-time monitoring and predictive maintenance in industrial, automotive and IoT domains.