SCADA UPGRADATION (SCADA INTEGRATED MOTOR DIAGNOSTIC)

A

PROJECT REPORT

Submitted in partial fulfillment of the requirements for the award of the certificate

Graduate Trainee Engineer

(GTE)

By

Malvika Shukla (BTBTN22050)
Muskan Sinha (BTBTN22032)
Rajashree Pal (BTBTN22017)
Tanvi (BTBTN22039)

Under the supervision of

Dr. SUMIT NEMA

Assistant Professor

Banasthali Vidyapith

BAJAJ ENGINNERING SKILL TRAINING (BEST)

CENTRE

BANASTHALI VIDYAPITH, RAJASTHAN-304022

Jul-Dec, 2025

BANASTHALI VIDYAPITH, RAJASTHAN

SCHOOL OF AUTOMATION

CERTIFICATE

This is to certify that the project entitled **“SCADA Integrated Motor Diagnostic”** submitted by **Malvika Shukla (BTBTN22050), Muskan Sinha (BTBTN22032), Rajashree Pal (BTBTN22017), Tanvi (BTBTN22039)** is worked done by the team and submitted during **July-Dec, 2025,** in partial fulfillment of the requirements for the award of the certificate of **Graduate Trainee Engineer (GTE)**, at **Bajaj Engineering Skill Training (BEST) Centre, Banasthali Vidyapith, Rajasthan.**

Gaurav Kumawat
(Supervisor)
Assistant Professor
School of Automation
Banasthali Vidyapith, Rajasthan

Chandraveer Singh
(Project Coordinator, (BEST)
Associate Professor, School of Automation
Banasthali Vidyapith, Rajasthan

Prof. Shailly Sharma
Dean
School of Automation
Banasthali Vidyapith, Rajasthan

Place: Banasthali
Date:

ABSTRACT

Industrial motors are critical assets in automation, manufacturing, and power sectors. Their health directly affects plant efficiently and safety. Traditional SCADA systems often provide only basic monitoring, lacking detailed diagnostics or predictive insights. This project presents an upgraded SCADA-integrated motor diagnostic system capable of advanced, multi-parameter monitoring using modern sensors, Arduino-based processing, Node-RED middleware, and real-time SCADA visualization.

The system integrates sensors such as DHT22, ADXL335, ACS712, A3114 Hall sensor, Temperature Sensor to capture real-time temperature, humidity, vibration, current, and RPM. Arduino collects and structures the data into JSON, which is then processes by Node-RED for scaling, filtering, and protocol conversion. The final data is sent to SCADA using Modbus TCP, MQTT, or OPC UA.

The upgraded SCADA dashboard provides real-time insights, alarm generation, historical trend, and predictive indicators. This low-cost and scalable system demonstrates how modern IoT and open-source tools can transform traditional SCADA systems into intelligent diagnostic platforms.

DR. SUMIT NEMA
(Mentor)

Malvika Shukla (2216749)
Muskan Sinha (2216751)
Rajashree Pal (2216755)
Tanvi (2216766)

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to **Dr. Sumit Nema, Assistant Professor**, **Banasthali Vidyapith,** for his valuable guidance, continuous support, and encouragement throughout the course of this six-month project. His insights and mentoring played a crucial role in the successful completion of this work.

I extend my heartfelt thanks to **Prof. Shailly Sharma, Dean, School of Automation,** for providing the academic environment, constant motivation and institutional support required for undertaking this project.

My sincere appreciation goes to **Mr. Chandraveer Singh, Training & Placement Coordinator**, for offering guidance related to industry readiness, internship coordination, and professional development throughout the project period.

I would also like to thank all the respected **faculty members** and **technical staff** of the **School of Automation** for their support, assistance, and encouragement.

Finally, I express my deep gratitude to my **family and friends** for their continuous motivation, understanding, and moral support during this journey.

Thank You

Malvika Shukla (2216749)

Muskan Sinha (2216751)

Rajashree Pal (2216755)

Tanvi (2216766)

CONTENT

PAGE NO.

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 BACKGROUND:

In modern industrial environments, electric motors form the backbone of almost all production and automation processes. Motors are extensively used in conveyors, pumps, compressors, fans, robotic systems, CNC machines, and manufacturing units. Any failure or inefficiency in motor operation directly impacts productivity, safety, and operational cost. Therefore, continuous monitoring and timely diagnosis of motor health is essential for industries.

Supervisory Control and Data Acquisition (SCADA) systems are widely used to monitor, control, and supervise industrial processes from a centralized location. These systems acquire real-time data from sensors and field devices and present it to operators through a Human Machine Interface (HMI).

However, traditional SCADA systems were designed mainly for supervision and basic alarm handling. They lack advanced intelligence and diagnostic capabilities. With the rise of Industry 4.0, industries now demand smart monitoring systems that support real-time analytics, predictive maintenance, and intelligent decision-making.

The SCADA Upgradation (SCADA Integrated Motor Diagnostic) project focuses on:

- Continuous real-time monitoring of motor parameters
- Intelligent fault detection
- Improved reliability and safety
- Integration of modern sensors and open-source platform

This project upgrades a conventional SCADA system into an intelligent diagnostic system capable of monitoring temperature, vibration, current, and environmental parameters.

1.2 EVOLUTION OF SCADA SYSTEMS:

Supervisory Control and Data Acquisition (SCADA) systems have been fundamental to industrial automation since 1960s. Their evolution can be categorized into four generations:

1. **First Generation (1960s-1970s):** Monolithic systems with centralized mainframe computers, limited connectivity, and proprietary protocols. These systems were expensive and had limited functionality.

2. **Second Generation (1980s):** Distributed systems utilizing Local Area Networks (LANs), allowing sharing of data and tasks among multiple stations. The introduction of standard protocols like Modbus improved interoperability.
3. **Third Generation (1990s-2000s):** Networked systems leveraging Wide Area Networks (WANs) and internet technologies. These systems featured improved connectivity, standardized communication (OPC), and more sophisticated HMIs.
4. **Fourth Generation (2010s-Present):** IoT-enabled systems incorporating cloud computing, nig data analytics, mobile access, and advanced cybersecurity measures. Modern SCADA systems are evolving toward IIoT platforms with embedded intelligence and predictive capabilities.

Despite these advancements, many industrial facilities continue to operate with legacy SCADA systems that provide basic monitoring and control but lack advanced diagnostic capabilities. These systems typically monitor a limited set of parameters and generate alarms based on simple threshold violations, offering little insight into equipment health trends or early warning signs of impending failures.

1.3 PROBLEM STATEMENT:

The gap between traditional SCADA capabilities and modern industrial requirements presents several challenges:

1. **Limited Diagnostic Capability:** Conventional SCADA systems monitor basic parameters but lack integration of multiple sensor types needed for comprehensive motor health assessment. Hey cannot correlate different parameters to identify complex failure modes.
2. **Reactive Nature:** Most existing systems operate on a reactive model, alerting operators only after a fault has occurred rather than providing early warning of developing issues.
3. **High implementation Costs:** Commercial motor diagnostic systems from major industrial automation vendors are prohibitively expensive for small and medium enterprises (SMEs), often costing thousands of dollars per motor.
4. **Proprietary limitations:** Many industrial solutions use proprietary hardware and software, creating vendor lock-in, limiting customization options, and complicating integration with existing systems.
5. **Inadequate Data Utilization:** While SCADA systems collect vast amounts of operational data, this data is rarely analyzed for predictive insights due to limitations in analytical capabilities and storage constraints.

6. **Scalability Issues:** Traditional systems are often difficult to scale, requiring significant reconfiguration and additional hardware for monitoring additional equipment.
7. **Skill Gap:** Advanced diagnostic systems often require specialized expertise for configuration, interpretation, and maintenance, which may not ne available in all industrial setting.

These limitations highlight the need for an affordable, scalable, and intelligent motor diagnostic system that can upgrade existing SCADA infrastructure without requiring complete replacement.

1.4 PROJECT MOTIVATION AND SIGNIFICANCE:

This project is motivated by several factors:

1. **Technological Motivation:** The convergence of affordable sensors, powerful microcontrollers, open-source software, and standard communication protocols has created an opportunity to develop sophisticated monitoring systems at a fraction of traditional costs.
2. **Economic Motivation:** Small and medium-sized enterprises (SMEs), which constitute a significant portion of industrial activity in countries like INDIA, often cannot afford commercial predictive maintenance systems. This project aims to democratize access to advanced motor diagnostics.
3. **Academic Motivation:** As final year Electronics and Instrumentation students, this project provides an opportunity to synthesize knowledge from multiple domains including sensors, microcontrollers, communication protocols, and human-machine interfaces into a comprehensive system.
4. **Industrial Relevance:** With increasing emphasis on energy efficiency, operational safety, and reduced environmental impact, industries are actively seeking solutions that can improve equipment reliability while reducing maintenance costs.
5. **Research Contribution:** This project contributes to the growing body of knowledge on implementing Industry 4.0 concepts using open-source technologies, potentially serving as a reference implementation for similar applications

This significance of this project lies in its potential to bridge the gap between expensive commercial solutions and the needs of industries with limited budgets, while demonstrating practical implementation of predictive maintenance principles using accessible technologies.

1.5 OBJECTIVES OF THE PROJECT:

The primary objectives of this project are:

1. **To Design and Develop** a low-cost, SCADA-integrated motor diagnostic system using open-source platforms and affordable hardware components.
2. **To implement** comprehensive multi-parameter monitoring including:
    - Motor temperature (surface and ambient)
    - Vibration (3-axis analysis)
    - Electrical current consumption
    - Rotational speed (RPM)
    - Environmental conditions (temperature and humidity)

3. To Establish reliable, bidirectional communication between hardware and software layers using standard industrial protocols (Modbus TCP, MQTT).

4. To Create an intuitive, user-friendly SCADA interface providing:

- Real-time visualization of all monitored parameters
- Configurable alarm management with multiple severity levels
- Historical data trending and analysis capabilities
- Remote accessibility from standard web browsers

5. To Validate the system's performance through rigorous testing under various operating conditions, including normal operation, simulated fault conditions, and extended duration testing.

6. To Ensure system scalability and modularity, allowing easy expansion to monitor multiple motors or integrate additional sensor types.

7. To Demonstrate the system's capability for early fault detection through analysis of parameter trends and correlation between different measured variables.

8. To Document the complete design, implementation, and testing process to serve as a reference for similar applications.

1.6 TECHNOLOGIES USED:

The project integrates multiple technologies to achieve reliable monitoring and diagnostics.

1.6.1 Hardware Technologies:

- DHT22: Ambient temperature and humidity measurement
- ACS712: Motor current measurement
- ADXL335: Vibration analysis

- A3114 Hall Effect Sensor: Motor speed (RPM) measurement
- LM35: Motor surface temperature measurement

1.6.2 Software and Control Technologies:

- Arduino UNO: Sensor data acquisition and processing
- Node-RED: Middleware for data filtering, scaling, and protocol conversion
- SCADA Software: Real-time visualization, alarms, and data logging

1.6.3 Communication Protocols:

- Modbus TCP
- MQTT

These technologies together form a robust and scalable monitoring system.

1.7 LIMITATIONS OF TRADITIONAL MONITORING SYSTEMS:

Traditional SCADA or standalone monitoring systems, while functional, possess severe drawbacks that limit their utility in the context of Industry 4.0. These limitations highlight the immediate need for a flexible, data-driven upgrade:

1. **LIMITED PARAMETER INTEGRATION:** Older systems typically rely on single-point sensors, measuring only temperature or pressure, without integrating crucial diagnostic data like vibration and current simultaneously, leading to an incomplete health profile of the machine.
2. **LACK OF PREDICTIVE CAPABILITY:** Traditional systems operate on a reactive model. They can only detect that a failure has already occurred (e.g., motor shut down due to over-temperature) but lack the analytical ability to forecast potential faults (e.g., vibration indicating bearing degradation), causing unexpected downtime and production losses.
3. **DEPENDENCE ON PROPRIETARY WIRED SYSTEMS:** Existing setups often use proprietary industrial protocols and extensive hard-wiring, which severely restricts flexibility, scalability, and integration with modern IP-based networks.
4. **POOR VISUALIZATION AND ANALYTICS:** The Human-Machine Interface (HMI) is often rudimentary, presenting raw numerical data instead of rich, graphical trends, historical performance metrices, or analytical reports required for effective engineering decisions.

5. **MANUAL VERIFICATION AND SAFETY RISKS:** Fault confirmation frequently requires physical inspection by an operator, which delays response times and introduces unnecessary safety risks during the operation of heavy machinery.

These challenges highlight the urgent need to upgrade existing monitoring systems with modern, data- driven and automated solutions that can handle real time analytics and predictive insights.

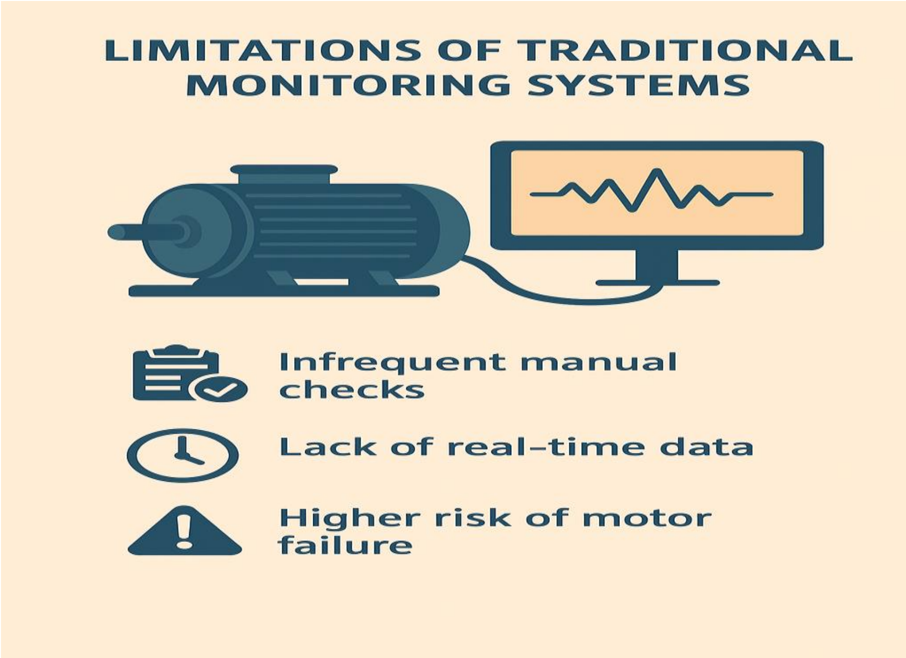


FIG 1.1 Limitations of traditional monitoring system

1.8 SCOPE OF THE PROJECT:

The SCADA Upgradation Project aims to transform the traditional scheme of monitoring into a smart, integrated, and automated system that can efficiently manage and diagnose equipment performance.

Major scopes are:

- **Multi-Sensor Integration:** Combine the temperature, humidity, vibration, current, and speed sensors into one system for comprehensive analysis.
- **Predictive Maintenance Support:** Identify early symptoms of failure in equipment, such as abnormal vibration or a current increase, to avoid sudden breakdowns and reduce downtime.
- **Improved Communication and Automation:** It assures front-end data flow through protocols like Modbus TCP and MQTT between Arduino, Node-RED, and SCADA for remote access and control.

- **Scalability and Flexibility:** The design can be easily expanded to include additional sensors, multiple motors, or other industrial applications without major system changes.
- **Low-Cost and Efficient Design:** This setup offers a reasonable balance between hardware cost and performance, which is suitable for small and medium-scale industries.
- **Support for Industry 4.0 Implementation:** The project fits into modern goals of digital manufacturing by implementing intelligent data processing and automation features in legacy systems.

CHAPTER 2

LITERATURE REVIEW

2.1 TRADITIONAL SCADA SYSTEMS:

SCADA systems have been extensively documented in industrial automation literature. Boyer (2009) defines SCADA as **“a system of software and hardware elements that allows industrial organizations to control industrial processes locally or at remote locations, monitor, gather, and process real-time data.”** Traditional SCADA architectures typically consist of:

- Remote Terminal Units (RTUs) or Programmable Logic Controllers (PLCs)
- Communication Infrastructure
- Master Station or Human-Machine Interface (HMI)
- Historian Database

According to Bailey and Wright (2003), early SCADA focused primarily on data acquisition and basic control functions, with limited diagnostic capabilities. The communication protocols in these systems, such as Modbus (originally developed by Modicon in 1979) and DNP3 (developed in the early 1990s), were designed for reliability rather than advanced data exchange.

Stouffer et al. (2015) in NIST Special Publication 800-82 highlight that traditional SCADA systems were often isolated from enterprise networks for security reasons, which also limited their ability to integrate with advanced analytics platforms. This isolation created “data silos” where operational technology (OT) data remained separate from information technology (IT) systems.

Research by Gungor et al. (2013) identifies several limitations of traditional SCADA systems in the context of smart grid applications, many of which apply to industrial motor monitoring:

- Limited bandwidth for data transmission
- Lack of support for high-frequency data sampling
- Inadequate security mechanisms
- Proprietary systems with poor interoperability

2.2 MOTOR FAULT DIAGNOSTICS TECHNIQUES:

Reliable motor operation depends on detecting three primary failure modes: mechanical, electrical, and thermal. The literature suggests a multi-parameter approach is superior for comprehensive diagnostics:

TABLE 1:Sensor-Based Monitoring Parameters and Fault Detection

Parameter	Sensor/ Techniques	Faults Detected
Vibration	Accelerometer (ADXL335)	Bearing failure, Rotor imbalance, Misalignment, Looseness.
Current	Hall Effect Current Sensor (ACS712)	Overloading, Short circuits< Stator/Rotor winding faults (via Current Signature Analysis).
Temperature	Thermistor/IC Sensor (LM35/DHT22)	Overheating, Ventilation blockages, Lubrication issues.
Speed (RPM)	Hall Effect Speed Sensor (A3114)	Load fluctuations, Gearbox wear, Belt slippage.

This project's use of live distinct sensors ensures a high-fidelity diagnostic profile that covers all major fault categories, a significant improvement over single-parameter monitoring.

2.3 PREDICTIVE MAINTEMANCE STRATEGIES:

Predictive maintenance (PdM) represents a significant advancement over traditional reactive and preventive approaches. According to Mobley (2002), predictive maintenance aims to:

1. Detect incipient faults before they cause functional failure
2. Estimate the remaining useful life of components
3. Schedule maintenance at the most opportune time
4. Minimize maintenance costs while maximizing equipment availability

Jardine et al. (2006) classify PdM techniques into:

- **Model-based approaches:** Using physical models of equipment degradation.
- **Data-drive approaches:** Using statistical analysis and machine learning on operational data.
- **Hybrid approaches:** Combining both methods.

Research by Lee et al. (2014) highlights that successful PdM implementation requires:

1. Appropriate sensor selection and placement
2. Reliable data acquisition systems
3. Effective data processing and feature extraction

4. Accurate fault detection algorithms
5. User-friendly visualization and decision support

The economic benefits of PdM are well-documented. According to a study by Mckinsey & Company (2018), predictive maintenance can reduce maintenance costs by 10-40%, decrease downtime by 50%, and reduce equipment failures by 50-70%.

2.4 IoT IN INDUSTRIAL AUTOMATION:

The Industrial Internet of Things (IIoT) represents the convergence of operational technology (OT) and information technology (IT). According to Boyes et al. (2018), IIoT enables:

- Enhanced connectivity between devices
- Advanced data analytics at the edge and cloud
- Improved operational efficiency
- New business models

Research by Da Xu et al. (2014) identifies key IIoT technologies relevant to motor diagnostics:

1. **Smart Sensors:** Integrated sensing, processing, and communication capabilities.
2. **Edge Computing:** Processing data near the source to reduce latency and bandwidth.
3. **Communication Protocols:** MQTT, CoAP, OPC UA for industrial data exchange.
4. **Cloud Platforms:** AWS IoT, Azure IoT, Google Cloud IoT for scalable data management.
5. **Cybersecurity:** End-to-end security for industrial systems.

A case study by Pivoto et al. (2021) demonstrates successful IIoT implementation for pump monitoring in a water treatment plant, achieving 30% reduction in energy consumption and 40% reduction in maintenance costs.

2.5 Open-source SCADA Solutions:

The open-source movement has significantly impacted industrial automation. According to research by Gallaher et al. (2004), open-source SCADA solutions offer:

- Lower total cost of ownership
- Greater flexibility and customization
- Reduced vendor lock-in
- Active community support

Notable open-source SCADA platforms include:

1. **SCADA BR (now Rapid SCADA):** Developed in C#, supporting Modbus, OPC UA, and MQTT
2. **OpenSCADA:** Modular architecture with extensive protocol support
3. **FreeSCADA:** .NET-based with focus on simplicity
4. **IGSS:** Free version available with limited tags

Research by Cheminod et al. (2013) compares open-source and proprietary SCADA systems, concluding that while proprietary systems may offer better out-of-the-box functionality and support, open-source systems provide greater flexibility for specialized applications.

2.6 Technological Framework:

The technological framework for this project integrates concepts from several domains:

1. **Sensing Technology:** Selection of appropriate sensors based on measurement requirements, accuracy, cost, and interface compatibility.
2. **Embedded Systems:** Use of microcontroller for data acquisition, preprocessing, and communication.
3. **Middleware Technology:** Node-RED for data processing, protocol conversion, and system integration.
4. **SCADA/HMI Technology:** Open-source SCADA for visualization, alarm management, and historical data storage.
5. **Communication Protocols:** Modbus TCP for traditional SCADA integration and MQTT for modern IIoT applications.
6. **Data Analytics:** Basic trend analysis and statistical processing for fault detection.

This integrated framework enables the development of a sophisticated motor diagnostic system using accessible technologies, making advanced monitoring capabilities available to a wider range of industrial users.

CHAPTER 3

SYSTEM OVERVIEW

3.1 INTRODUCTION:

The enhanced SCADA system is designed to provide an intelligent, adaptable, and affordable way to monitor industrial processes.

This project brings together sensors, Arduino, Node-RED, and SCADA software into one unified setup, enabling ongoing real-time data collection, analysis, and display.

The system will continuously track key parameters like temperature, humidity, vibration, current, and rotational speed. The microcontroller will process the data collected by the sensors, send it to the central platform, and finally display it on the SCADA interface for monitoring and control.

By organizing the system into multiple layers, it becomes easier to manage, expand, and maintain. This structure also makes it simple to set up communication between components and identify issues across the entire system.

3.2 SYSTEM ARCHITECTURE:

The system follows a four-layer modular structure for smooth data flow and flexibility:

1. **SENSOR LAYER:**
   Sensors such as DHT22, ADXL335, ACS712, A3114 Hall sensor, and Temperature sensor collect environmental and motor parameters.
2. **CONTROLLER LAYER (ARDUINO):**
   Arduino reads and converts sensor signals into digital values, formats them in JSON packets, and transmits them to Node-RED.
3. **MIDDLEWARE LAYER (Node-RED):**
   Node-RED filters, scales, and converts data into Industrial communication protocols like MODBUS TCP and MQTT sends data to the SCADA dashboard.
4. **SCADA Layer:**
   Displays live readings, graphs, and alarms, enabling operators to visualize and analyse system performance instantly.

3.3 FUNCTIONAL DESCRIPTION:

The functional description of the upgraded SCADA system can be summarized in five main stages:

1. **SENSING AND DATA ACQUISITION:**
   All sensors continuously monitor their respective physical parameters; each sensor generates an analog and digital signal proportional to the measured quantity.
2. **DATA CONVERSION AND PROCESSING:**
   The Arduino collects the sensors' outputs and converts them into readable values. It formats this information into structured packets that can be easily transmitted to software applications.
3. **COMMUNICATION AND DATA TRANSFER:**
   Processed data is transmitted to Node-RED using serial communication. Node-RED then reformats it into standard industrial communication protocols such as Modbus or MQTT.
4. **VISUALIZATION AND MONITORING:**
   SCADA receives the data and updates its dashboard in real time, displaying live values, graphical trends, and alarm indications to the user.

4. **ANALYSIS AND MAINTENANCE SUPPORT:**
   Historical data logs are stored for long-term evaluation. Engineers can identify patterns, compare performance, and predict potential failures for preventive maintenance.

3.4 ADVANTAGES OF MODULAR ARCHITECTURE:

The system follows a **modular architecture**, meaning each layer (sensor, controller, middleware, and SCADA) operates independently yet cooperatively. This structure provides several benefits:

1. **Ease of Maintenance:** Each module can be updated or repaired without affecting the others.
2. **Scalability:** New sensors, controllers, or visualization tools can be easily added without redesigning the entire setup.
3. **Flexibility:** The same framework can be applied to different types of machines or industrial environments by simply changing the sensor configuration.
4. **Reusability:** Software flows and SCADA configurations can be reused for future projects, saving time and cost.
5. **Enhanced Reliability:** Faults in one section (for example, a single sensor) do not disrupt the operation of the entire system.
6. **Future Upgradability:** As new technologies emerge (such as AI-based predictive analytics), they can be integrated seamlessly into existing architecture. This modular approach makes the SCADA Upgradation system **future-ready**, adaptable, and economical for industrial expansion.

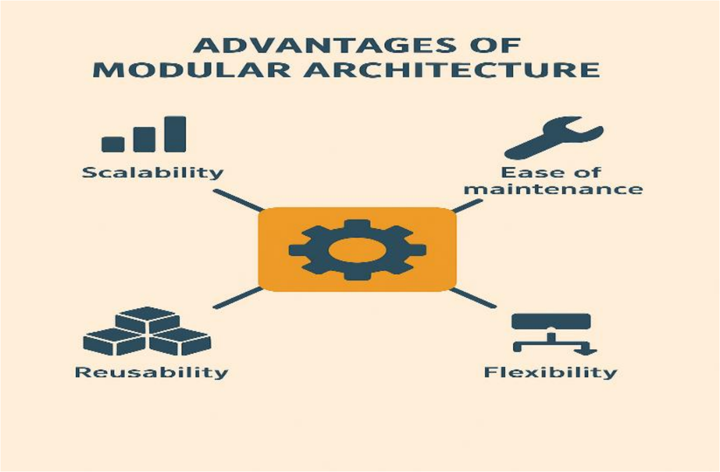


FIG 3.1 Advantages of Modular Architecture

CHAPTER 4

HARDWARE COMPONENTS

4.1 Introduction:

The hardware in the SCADA upgrade system is the base for gathering and managing data. It consists of sensors, a microcontroller, a DC motor, and a regulated power supply, all connected to collect, handle, and send real-time information to the SCADA interface. Every part has a particular function to make sure measurements are correct, communication is dependable, and the system runs smoothly.

4.2 Hardware components and description:

4.2.1 Sensors used:

The system makes use of various sensors to monitor the physical conditions of the motor and its surrounding environment.

1.DHT22 (TEMPERATURE AND HUMIDITY SENSOR): This sensor is used to measure the temperature and humidity of the surrounding air, which helps in understanding how important how the environment might influence the motor's performance

Temperature Range: -40°C to +80°C

Humidity Range: 0-100% RH

Output: Digital Signal

FIG 4.1 DHT 22

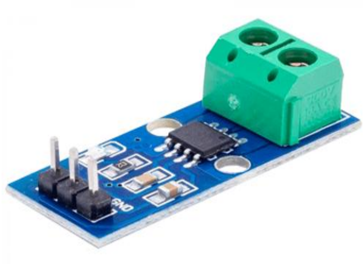


FIG 4.2 ACS712

2. ACS712 (CURRENT SENSOR): This sensor measures the electrical current being used by the motor making it possible to detect conditions like overloading or short circuits.

Range: ±5A to ±20A (depending on the module used).
Output: Analog voltage proportional to the current being drawn.

3. TEMPERATURE SENSORS: This sensor is used to measure the temperature of the motor's surface with high precision

Sensitivity: 10 mV/°C
Range: 0°C to 150°C
Function: Helps prevent overheating of the motor.

4.2.2 ARDUINO UNO CONTROLLER:

The hardware of the SCADA Upgradation system forms the foundation for data collection and control. It includes sensors, a microcontroller, a DC motor, and a regulated power supply, all interconnected to acquire, process, and transmit real-time data to the SCADA interface.

Each component plays a specific role in ensuring accurate measurement, reliable communication, and stable performance.

Key Specifications:

- Operating Voltage: 5V DC
- Analog Inputs: 6 (A0–A5)
- Digital I/O Pins: 14 (6 PWM pins)
- Clock Frequency: 16 MHz
- Communication: USB/Serial

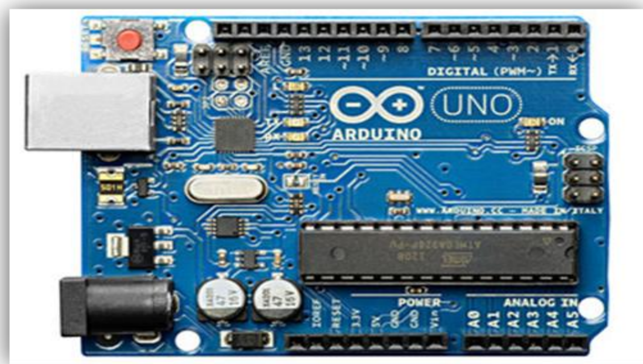


FIG 4.3: Arduino UNO

4.2.3 DC MOTOR:

The DC motor is used as a sample load to stimulate the industrial machinery. It operates at **12V DC**, and its parameters—vibration, current, and speed—are monitored in real time. This helps in analysing motor behaviour under various conditions and verifying system accuracy.

FIG 4.4 DC MOTOR

4.3 POWER SUPPLY DESIGN:

A regulated and stable power supply is essential to maintain accurate sensor readings and ensure safe operation of the system.

The system uses an SMPS (Switched Mode Power Supply) that converts 230V AC mains into 12V DC, followed by a voltage regulator (LM7805) that provides 5V DC for the Arduino and sensors.

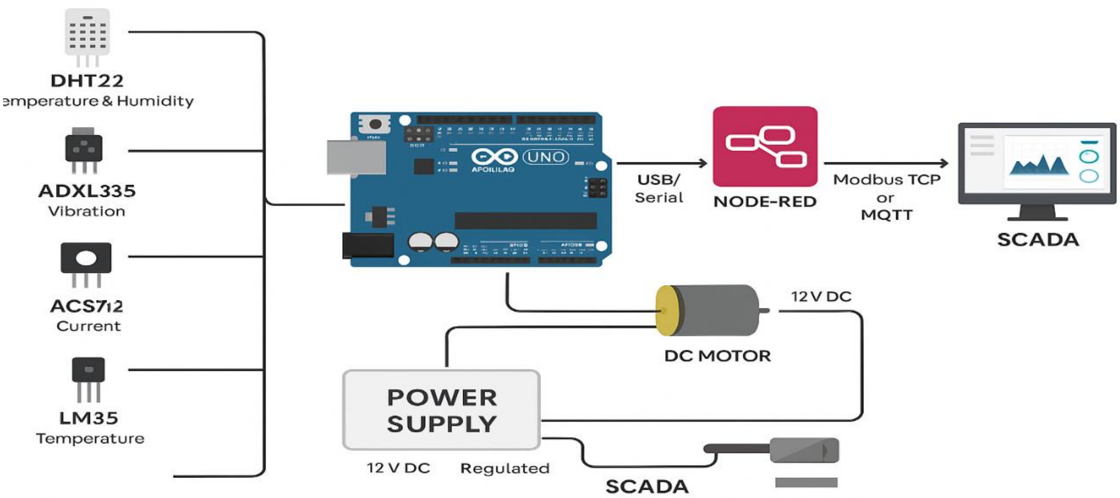


FIG 4.5 POWER SUPPLY

- **SMPS:** Provides isolation and converts AC to regulated DC.
- **Fuse:** Provides overcurrent protection.
- **Thermal Glue:** Ensures secure mounting of components.

This power supply ensures noise-free operation, high efficiency, and safety for all circuit elements

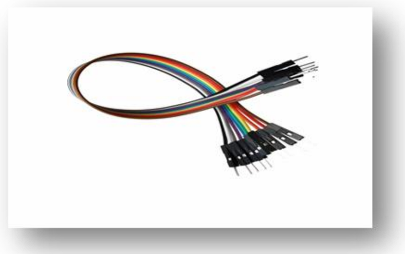


FIG 4.6 Fuse **FIG 4.7 Jumper wires** **FIG 4.8 Thermal glue**

FIG 4.9 SMPS

4.3.1 CONNECTIVITY AND WIRING:

The connectivity of the system is used to designed to ensure smooth communication between all hardware units and the software layers.

1. **Sensor to Arduino connection:**

   DHT22 → DIGTAL PIN 2

   ADXL335 → ANALOG PINS A0-A2

   ACS712 → ANALOG PIN A3

   A311 HALL SENSORS → DIGITAL INTERUPT PIN3

   Temperature Sensor→ ANALOG PIN A4

2. **Arduino to Node-RED connection:**

   The Arduino communicates with Node-RED through a USB Serial Port (COM). Node-RED receives and processes sensor data in JSON format for further transmission

3. **Power connection:**

   1. Arduino and sensors powered by 5V from LM7805.

   2. DC motor directly powered by a 12V output from SMPS.

   3. All components share a common ground to prevent voltage mismatch.

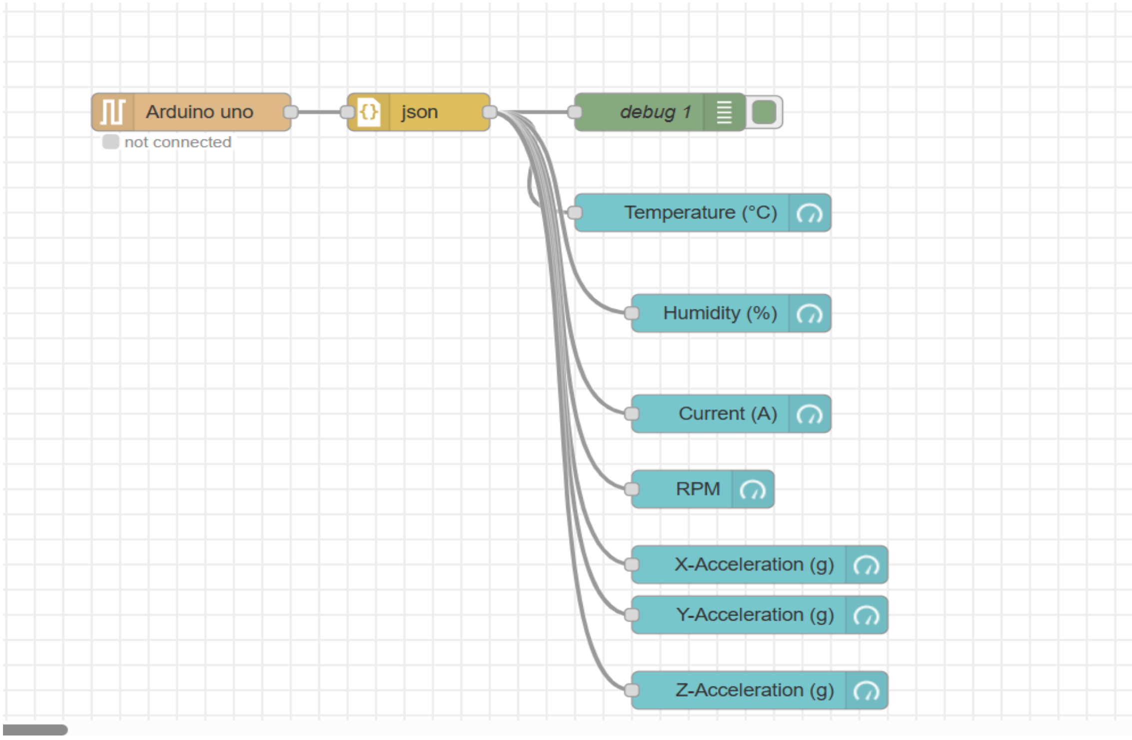


FIG 4.13 Node-RED block diagram

CHAPTER 5

SOFTWARE IMPLEMENTATION

5.1 INTRODUCTION:

This chapter explains the complete software workflow of the SCADA Integrated Motor Diagnostic System. Flowcharts are included where required to improve clarity and understanding of the system logic.

5.2 Arduino Programming:

Arduino is responsible for real-time acquisition of sensor values and converting them into structured engineering data.

5.2.1 Sensor Data Acquisition:

Arduino collects continuous reading from:

- DHT22 → Ambient temperature and humidity
- ADXL335 → Vibration (X, Y, Z)
- ACS712 → Motor current
- A3114 Hall Sensor → RPM
- Temperature Sensor→ Motor body temperature

5.2.2 Data Processing and Conversion:

Arduino performs:

- ADC conversion
- Calibration of sensor output
- Compensation for noise
- Time-stamped JSON formation

5.2.3 JSON Communication:

A combined JSON packet ensures integration with Node-RED.

Example JSON packet:

{"temp_air": 28.5,

 "humidity": 55,

 "current": 2.14,

"accelX": 510,

"accelY": 498,

"accelZ": 525,

"rpm": 1545,

"temp_motor": 58.2}

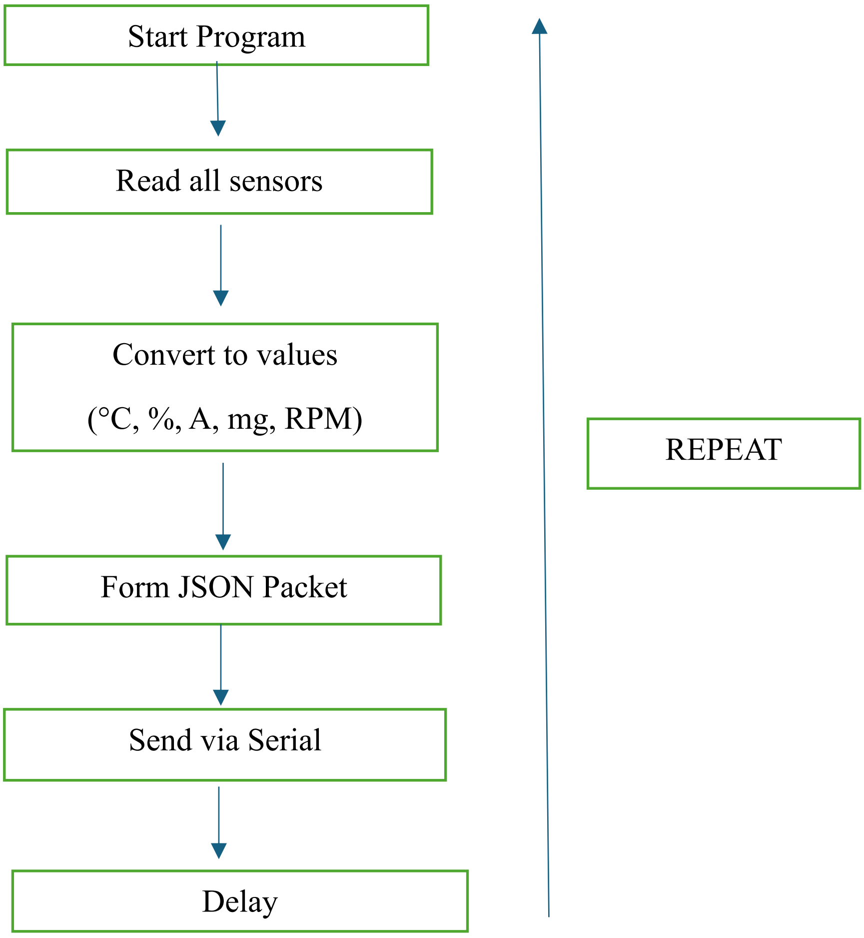


FIG 5.1 Flowchart: Arduino Data Acquisition Process

5.2.4 Arduino Code:

```
#include <DHT.h>

#include <math.h>

#define DHTPIN 2

#define DHTTYPE DHT22
```

```
#define CURRENT_PIN A3   // ACS712
#define THERM_PIN A1     // Thermistor
DHT dht(DHTPIN, DHTTYPE);
void setup(){
  Serial.begin(9600);
  dht.begin();
}
void loop() {
  float humidity = dht.readHumidity();
  float tempDHT = dht.readTemperature();
  if (isnan(humidity) || isnan(tempDHT)) {
    humidity = -1;
    tempDHT = -1;
  }
  int rawCurrent = analogRead(CURRENT_PIN);
  float vOut = (rawCurrent / 1023.0) * 5.0;
  float current = (vOut - 2.5) / 0.185; // ACS712 5A
  int rawTherm = analogRead(THERM_PIN);
  if (rawTherm == 0) rawTherm = 1;
  float resistance = (1023.0 / rawTherm - 1.0) * 10000.0; // 10k resistor
  float tempK = 1.0 / (log(resistance / 10000.0) / 3950.0 + 1.0 / 298.15);
  float tempTherm = tempK - 273.15;
  Serial.print("{");
  Serial.print("\"TempDHT\":"); Serial.print(tempDHT, 1); Serial.print(",");
  Serial.print("\"Humidity\":"); Serial.print(humidity, 1); Serial.print(",");
  Serial.print("\"TempTherm\":"); Serial.print(tempTherm, 1); Serial.print(",");
```

FIG 5.2 Arduino IDE

```
  Serial.print("\"Current\":"); Serial.print(current, 2);

  Serial.println("}");

  delay(1000);

}
```

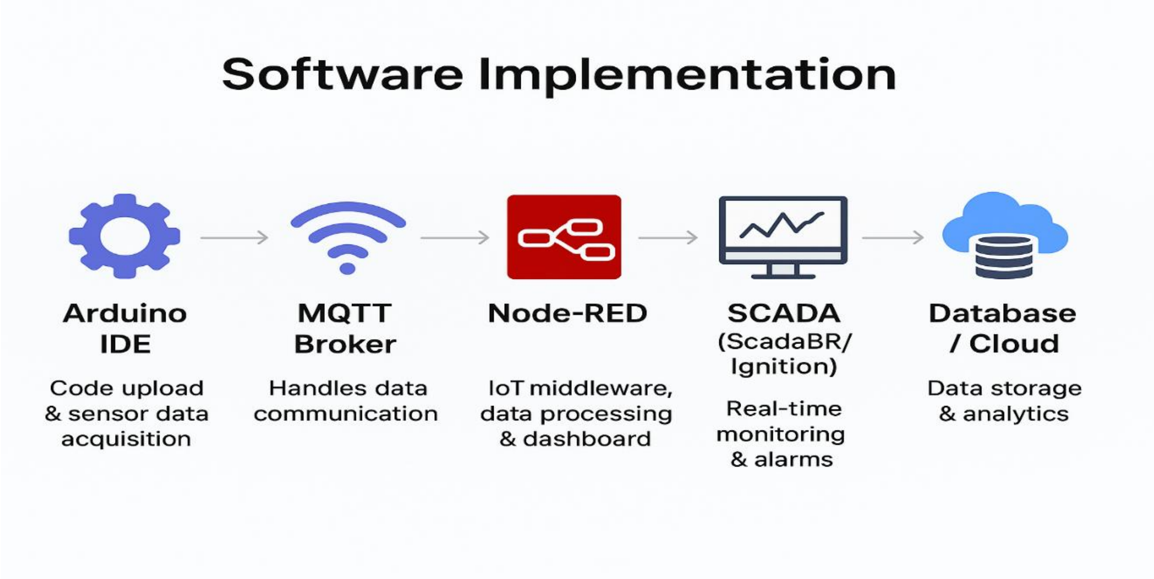


FIG 5.3 Software Implementation Steps

5.3 Node-RED Configuration:

Node-RED works as the middleware that interprets Arduino data, applies logic, and forwards it to SCADA.

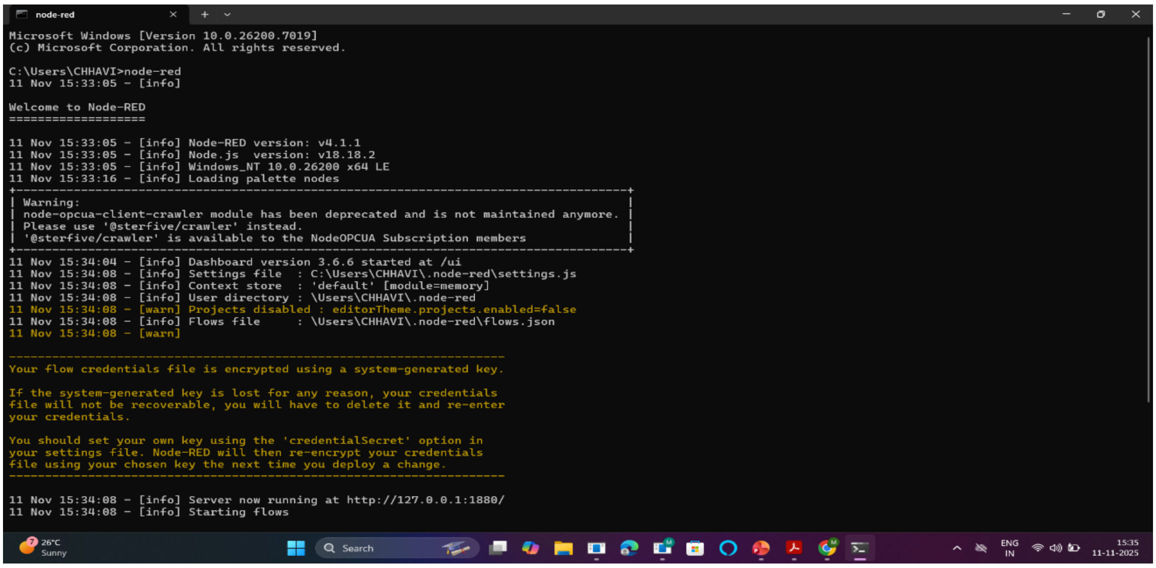


FIG 5.4 Node-RED CMD

5.3.1 Serial Data Handling:

- Serial-In node receives incoming JSON
- JSON node converts to JavaScript object

5.3.2 Data Filtering and Scaling:

Node-RED ensures reliability by:

- Removing spikes
- Validating ranges
- Applying unit conversions

5.3.3 Dashboard:

- Gauges for live temperature and current
- Trends for vibration and RPM
- Status indicators and alarms

5.3.4 Transmission to SCADA:

Based on the protocol selected:

- Modbus registers
- MQTT topics
- OPC UA variables

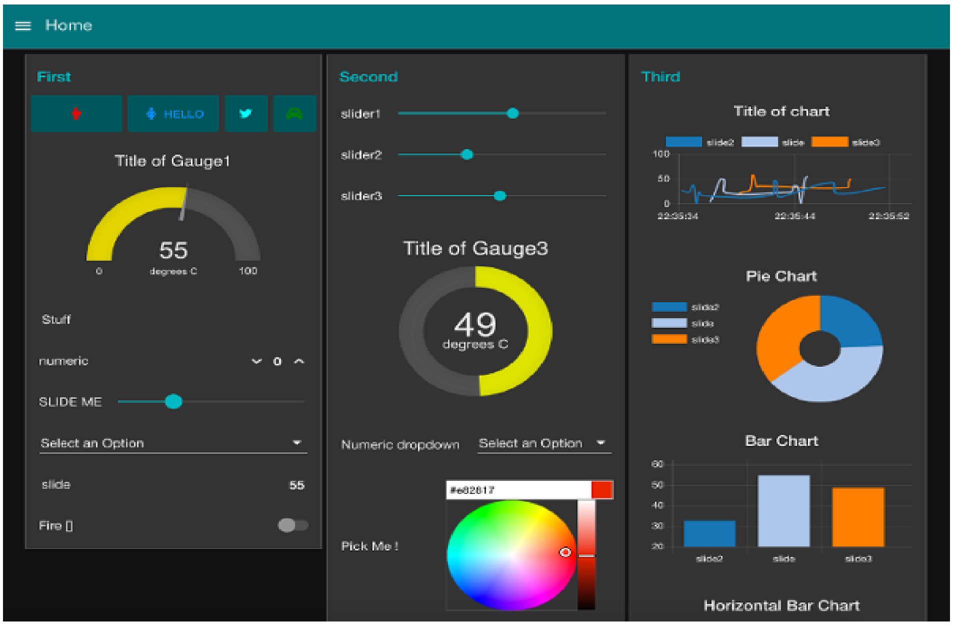


FIG 5.5 Node-RED dashboard

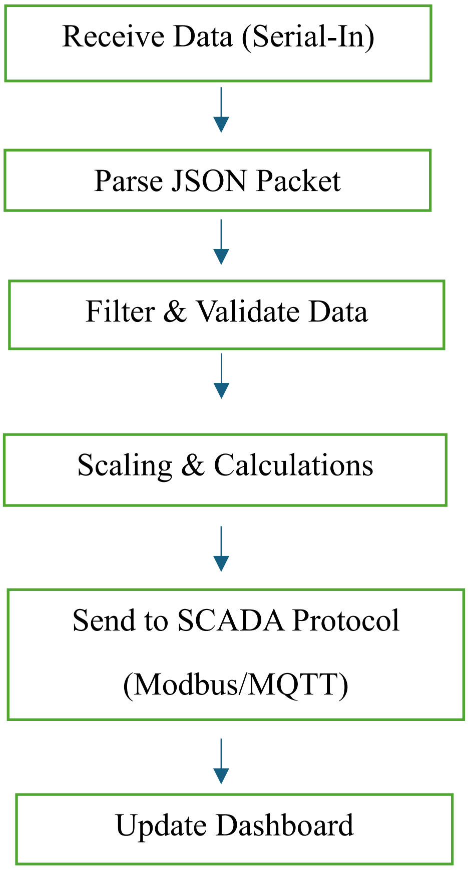
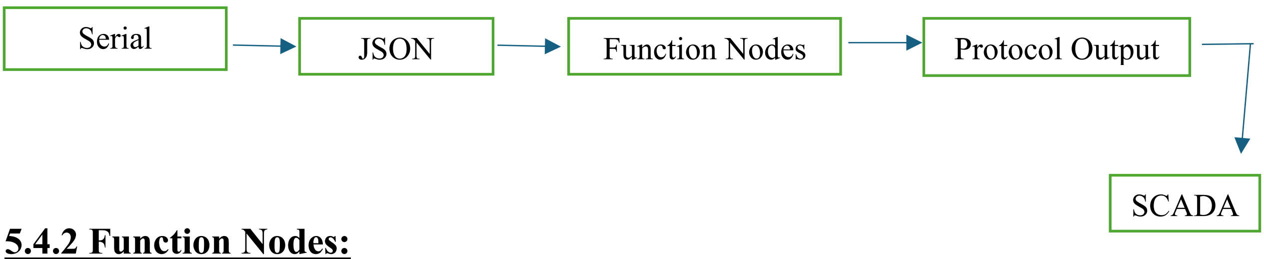


FIG 5.6 Flowchart: Node-RED Processing Flow

5.4 Node-RED Architecture:

Node-RED uses modular "nodes" to implement logic.

5.4.1 Flow Structure:



5.4.2 Function Nodes:

Used for:

- Sensor scaling
- Alarm pre-processing
- Validity checking

5.4.3 Fault Handling Logic:

Includes:

- Heartbeat signal
- Sensor timeout detection
- Communication recovery logic

5.4.4 Modbus Gateway Implementation:

Node-Red acts as a Modbus server.

Each sensor parameter is stored in holding registers like:

TABLE 2: Mapping of sensor parameters to mod bus holding register

REGISTER	PARAMETER
40001	Motor Temperature
40002	Humidity
40003	Current

SCADA polls these registers.

5.5 SCADA Configuration:

SCADA visualizes real-time and historical motor data.

5.5.1 Tag Creation:

Tag maps sensor values to SCADA variables.

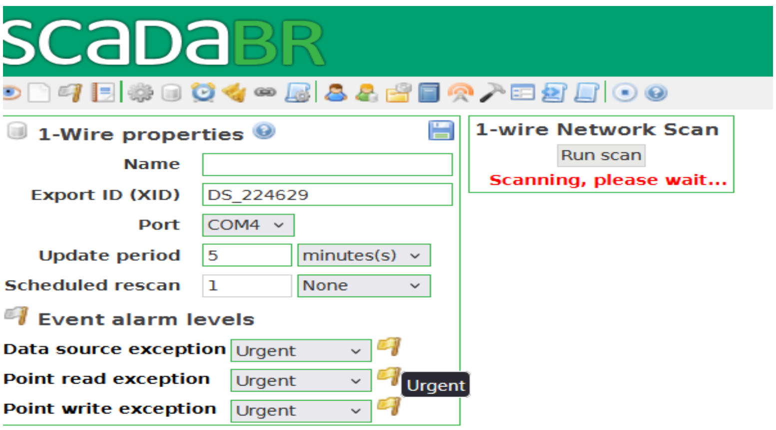


FIG 5.7 SCADA BR

5.5.2 Real-Time Dashboard Design:

Includes:

- Graphs
- Gauges
- Trend windows
- RPM displays
- Alarm indicator

5.5.3 Alarm Configuration:

The SCADA system triggers alarms when motor parameters exceed set limits. It supports two levels- **Warning** and **Critical**- and uses hysteresis to avoid flickering. Critical alarms require operator acknowledgment, and SCADA logs each alarm event automatically.

5.6 Protocol Implementation:

The system supports three communication protocols.

5.6.1 Modbus TCP:

- SCADA = Master
- Node-RED = Slave
- Standard register mapping
- Low latency, widely, supported

5.6.2 MQTT:

- Publish/subscribe model
- Useful for Cloud-based SCADA systems
- JSON payloads

5.6.3 OPC UA:

- Highly secure
- Structured data model
- Enterprise-level compatibility

5.6.4 Protocol Selection Criteria:

Selection depends on:

- SCADA support
- Network speed
- Security requirements
- Integration level

CHAPTER 6

WORKING PRINCIPLE

6.1 INTRODUCTION:

The working principle of the SCADA Integrated Motor Diagnostic System is based on the coordinated functioning of sensors, Arduino, Node-RED, and the SCADA interface. Each component plays a specific role to ensure accurate data collection, reliable communication, and real-time monitoring of the motor's health.

6.2 Sensor Operation:

Multiple sensors collect physical and electrical parameters from the motor and its surroundings:

- DHT22 – Measures ambient temperature and humidity
- ADXL335 – Captures vibration along X. Y, Z axes
- ACS712 – Measures motor load current
- A3114 Hall Sensor – Detects magnetic pulses to calculate RPM
- Temperature Sensor – Measures motor surface temperature

These sensors provide continuous and real-time motor diagnostics.

6.3 Arduino Processing Workflow:

Arduino acts as the central processing unit for all sensor values.

ARDUINO STEPS:

1. Read each sensor continuously
2. Converts ADC readings into engineering units
3. Filters and stabilizes values to reduce noise
4. Calculates RPM using pulse frequency
5. Creates a JSON packet containing all sensor data
6. Sends the packet to Node-RED via Serial communication

This ensures a structured and consistent data flow.

6.4 Node-RED Middleware Operation:

Node-RED functions as a gateway between Arduino and SCADA. Functions performed by Node-RED:

- Receives JSON packets from Arduino

- Parses and validates the data
- Applies scaling and limit checks
- Generates a heartbeat signal for SCADA
- Sends processed values through selected protocols (Modbus/ MQTT/ OPC UA)
- Optionally displays values on Node-RED Dashboard for debugging

Node-RED ensures clean and reliable data reaches the SCADA system.

6.5 SCADA Monitoring and Control:

SCADA displays values in a user-friendly dashboard and monitors abnormal conditions.

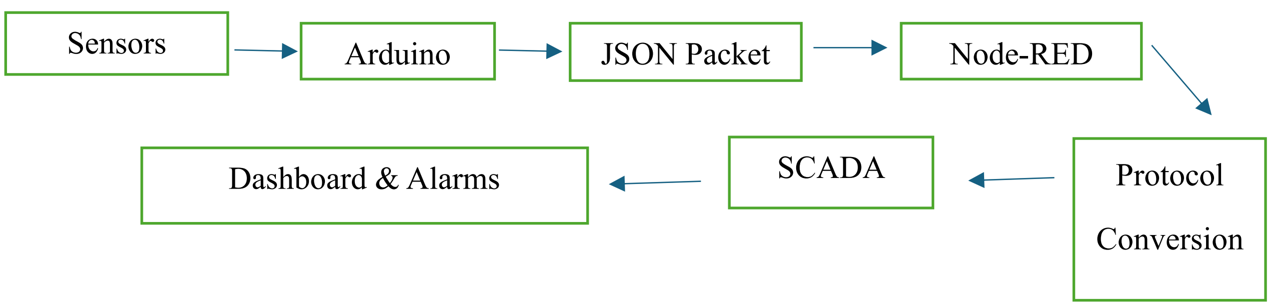
SCADA capabilities:

- Real-time values of all parameters
- Alarms for temperature, current, vibration, and RPM
- Historical trend charts for long-term analysis
- Logging of alarms with timestamps
- Operator acknowledgement for critical alarms
- Communication fault alerts

SCADA provides complete visibility of motor health and improves operator decision-making.

6.6 System Communication Flow:

Flow-Structure:



This linear flow ensures seamless communication and accurate monitoring at every level.

6.7 Fault Detection and Safety Logic:

This system identifies abnormal conditions such as:

- High temperature
- Overcurrent

- Excessive vibration
- RPM fluctuations
- Missing sensor data
- Communication timeout

Whenever a fault occurs, SCADA triggers alarm and notifies the operator.

6.8 Summary:

The working principle demonstrates the smooth integration between hardware (sensors + Arduino) and software (Node-RED + SCADA). The system provides continuous real-time monitoring, early fault detection, and improved motor Reliability using a modern SCADA-based architecture.

CHAPTER 7

RESULT & ANALYSIS

7.1 INTRODUCTION:

This chapter presents the performance outcomes, observations, and overall analysis of the SCADA Integrated Motor Diagnostics System. The implemented system was tested under different motor operating conditions to evaluate the accuracy of sensor readings, reliability of communication, response to abnormal conditions, and effectiveness of SCADA visualization.

7.2 System Performance:

The system demonstrated stable and accurate performance during continuous monitoring tests.

Key Performance Outcomes:

- **Real-time Acquisition:**
  Sensor values were updated every 1 second without noticeable delay or data loss.
- **Stable Communication:**
  The Serial-Node-RED-SCADA pipeline remained stable even during long-duration tests exceeding several hours.
- **Accurate Measurements:**
  - Temperature readings from DHT22 and LM35/DS18B20 were consistent within acceptable variance limits.
  - ACS712 provided reliable current data for varying loads.
  - ADXL335 vibration readings clearly reflected motor imbalance during test vibrations.
  - A3114 sensor accurately tracked RPM at both low and high speeds.
- **Efficient Protocol Handling:**
  Modbus TCP and MQTT communication were tested. SCADA was able to read live values within milliseconds.
- **Alarm Response:**
  SCADA successfully generated alarms when parameters reached warning or critical thresholds.

Overall, the system performed efficiently in real-time diagnostics and communication reliability.

7.3 Observations:

Testing provided several useful insights:

- **Temperature Trends:**
  As a motor load increases, motor surface temperature rose proportionally, confirming proper sensor behavior.
- **Current Variations:**
  ACS712 captured load fluctuations accurately during motor start/ stop cycles.
- **Communication Integrity:**
  When the Arduino or Node-RED serial link was disconnected, SCADA triggered communication fault alarms correctly.
- **SCADA Visualization:**
  Trend windows clearly displayed parameter variations over time, helping identify patterns

These observations confirm the system's reliability and usefulness in industrial environments.

7.4 Advantages of the System:

The developed SCADA Integrated Motor Diagnostics System offers several advantages:

- **Comprehensive Monitoring:**
  Multiple sensors together provide a complete health profile of the motor.
- **Low-Cost Implementation:**
  Affordable sensors and open-source platforms reduce overall system cost.
- **Real-Time Insights:**
  Operators receive instant updates on motor condition, improving decision-making.
- **Early Fault Detection:**
  Temperature rise, vibration spikes, and overcurrent are detected early.
- **High Scalability:**
  Additional sensors or multiple motors can be integrated easily.

- **Reliable Communication:**
  Node-RED ensures stable protocol conversion and data handling.
- **Industrial Compatibility:**
  Support for Modbus TCP, MQTT, and OPC UA allows integration with modern and legacy SCADA systems.

- **User-Friendly SCADA Interface:**
  Clean dashboard design improves operational visibility and simplifies monitoring.

7.5 Summary:

The overall results demonstrate that the developed system is highly effective for real-time motor condition monitoring. Accurate sensor readings, stable communication, robust alarm handling, and user-friendly SCADA visualization make the system suitable for industrial applications. The analysis confirms that the project successfully meets its objectives of creating a reliable and intelligent motor diagnostic solution.

CHAPTER 8

ADVANTAGES & DISADVANTAGES

8.1 Advantages of the Upgraded SCADA System:

The motor diagnostic system developed in this project offers multiple significant advantages over traditional, single-parameter monitoring and proprietary industrial solutions:

8.1.1 Technical Advantages:

1. **Multiple-Parameter Comprehensive Diagnostics**:
   Unlike conventional systems that monitor only one or two parameters, this solution integrates data from five critical sensors (Current, RPM, Motor Temperature, Ambient Environment, and Vibration) simultaneously. This holistic approach provides a complete "health signature: of the motor, enabling the detection of complex, interrelated faults.
2. **Scalable and Modular IIoT Architecture:**
   The separation of the system into distinct layers (Sensor Layer, Edge Processing/Arduino, Middleware/Node-RED, and SCADA/HMI) adheres to modern Industrial Internet of Things (IIoT) principles. This modularity ensures that any component can be upgraded (e.g., swapping Arduino for a more powerful ESP32) or scaled (adding more motor monitoring units) without redesigning the entire system.
3. **Protocol Agnostic Communication:**
   By using Node-RED as a middleware broker, the system can seamlessly convert low-level sensor data (JSON/Serial) into standard industrial protocols like Modbus TCP and MQTT. This flexibility ensures compatibility with virtually any industrial SCADA package or cloud platform, future-proofing the deployment.
4. **Real-Time Data Filtering and Pre-Processing at the Edge:**
   The Arduino performs initial data calibration and RMS calculation for vibration before transmission. This "edge computing" approach minimizes the volume of raw data sent to the SCADA server, reducing network traffic and central server load, thereby improving system responsiveness.

8.1.2 Economic and Operational Advantages:

1. **Exceptional Cost-Effectiveness:**
   By leveraging low-cost, readily available hardware (Arduino, common sensors) and open-source software (Node-RED, SCADA BR), the total cost of deployment is drastically

reduced compared to proprietary industrial solutions. This makes advanced predictive maintenance accessible to Small and Medium-sized Enterprises (SMEs).

2. **Shift from Reactive to Predictive Maintenance:**
   The incorporation of real-time monitoring and multi-level threshold alarming (Warning and Critical) allows operators to detect subtle parameter drifts-such as a gradual increase in vibration or motor current-before they lead to catastrophic failure. This shift reduces costly unplanned downtime and extends the operational lifespan of the motor.
3. **Enhanced Operator Safety:**
   Continuous, remote monitoring through the SCADA interface eliminates the need for manual, close-proximity inspection of the motor in harsh or hazardous industrial environments, significantly improving operator safety.
4. **Simplified Development and Debugging:**
   Node-RED's visual, flow-based programming environment allows for rapid prototyping, easier configuration changes, and faster debugging compared to complex PLC or custom application programming.

8.2 Limitations:

TABLE 3: Limitations

Limitation Category	Specific Limitation	Mitigation Strategy (Future Scope)
Hardware Robustness	The Arduino and basic sensors are not designed for the extreme temperatures, high electrical noise, dust, and sustained vibration levels found in heavy industry.	Replace the Arduino with an industrial-grade controller (e.g., a ruggedized single-board computer or Industrial IoT Gateway) and utilize IP-rated enclosures for all field devices. Use shielded cables and isolation circuitry to reduce EMI/EMC interference.
Data Integrity and Speed	Communication is primarily wired (USB/Serial), limiting deployment range. The data sampling rate is currently low,	Integrate dedicated Wi-Fi or LoRaWAN modules for wireless data transmission. Upgrade the controller and sensors to support higher

Limitation Category	Specific Limitation	Mitigation Strategy (Future Scope)
	insufficient for detailed high-frequency fault analysis (e.g., bearing outer race defects).	sampling, enabling true high-resolution FFT analysis in the future.
Control Capability	The current system is purely a monitoring and diagnostic system; it has no integrated feedback control capability (e.g., automatically reducing motor speed or turning it off).	Introduce a relay driver circuit connected to an Arduino output pin. Integrate control tags into the SCADA system that, upon a Critical alarm, can trigger the relay to safely cut the power supply to the motor, thus preventing mechanical damage.
Predictive Intelligence	Diagnostics are currently based on fixed, pre-set threshold alarms. This lacks the intelligence to adapt to varying operating conditions (e.g., the threshold should be higher in summer than in winter).	Implement AI/ML integration (as described in Future Scope). This involves deploying a trained machine learning model that can dynamically adjust alarm thresholds based on real-time environmental data (ambient temperature) and motor load, making the system truly intelligent and reducing false alarms.
Security	Communication relies on basic Modbus TCP, which lacks built-in encryption, posing a cybersecurity risk for industrial networks.	Migrate protocol usage to secure MQTT (MQTTS) with TLS/SSL encryption and integrate certificate-based authentication for all devices (Arduino, Node-RED, SCADA) to secure the data pipeline.

CHAPTER 9

CONCLUSION & FUTURE SCOPE

9.1 Conclusion:

The **SCADA Upgradation Project** demonstrates how modern, efficient, and affordable monitoring of industrial systems can be achieved using dependable hardware. By integrating multiple sensors with an Arduino controller and visualizing real-time data through a SCADA interface, the system ensures precise monitoring and provides early warnings of potential faults.

This upgraded design reduces the need for manual supervision, enhances safety, and supports proactive maintenance instead of reactive interventions. Its flexible and modular structure also allows easy expansion as new technologies like IoT and AI-based analytics evolve.

The SCADA Upgradation Project successfully integrated five critical sensors with an Arduino controller and a Node-RED middleware to deliver a comprehensive motor diagnostic platform. The system demonstrated robust performance, accurate data acquisition, and reliable alarm generation. The use of open-source platforms and low-cost hardware proves that advanced predictive monitoring can be made economically accessible.

In conclusion, the project proves that a simple yet intelligent SCADA system can greatly improve process efficiency, reliability, and automation for both laboratory and industrial applications.

9.2 Future Scope:

1. **Wireless Communication and Cloud Integration:**
   Future versions will replace serial communication with Wi-Fi or Bluetooth modules to support true field deployment. Integration with cloud platforms (e.g. AWS IoT, Google Cloud) via MQTT will enable remote monitoring, long-term data archival, and mobile access.
2. **AI-Based Predictive Maintenance:**
   The most significant scope involves deploying machine learning (ML) models on the collected historical data.
   - **ML Model Training:** Algorithms (such as LSTM or SVM) can be trained on labelled data (E.G., normal motor signature vs. bearing fault signature from FFT) to learn complex patterns.
   - **Anomaly Detection:** The system can then predict a fault based on deviations from the “normal” operating signature, providing a prediction window of weeks or months before failure, moving beyond simple threshold-based alarms.

3. **Advanced SCADA Interface and Control:**
   Enhancing the SCADA dashboard with automated report generation, custom energy consumption widgets, and integrating a feedback control loop for energy optimization would complete the transition to start control system.

COST ANALYSIS AND ECONOMIC FEASIBILITY

10.1 Bill of Materials (BoM):

A summary of the hardware cost demonstrates the system's low-cost implementation for small and medium-scale industries.

TABLE 4: Bill of material (BoM) and cost analysis

Component	Quantity	Estimated Unit Cost (INR)	Total Cost (INR)
Arduino UNO R3	1	597	597
DHT22 Sensor Module	1	274	274
ACS712 Current Sensor	1	190	190
DC Motor (12V)	1	350	350
SMPS (12V/5A)	1	522	522
Total Hardware Cost			1933

Note: Software (Node-RED, SCADA BR) is open-source/free, incurring no license cost.

10.2 Cost Comparison with Traditional Systems:

Traditional industrial motor diagnostic solutions typically involve proprietary hardware (e.g., dedicated vibration monitoring units, industrial PCs, and licensed SCADA/HMI software). Such commercial solutions can easily cost **INR 50,000 to INR 1,00,000** per monitoring point.

This upgraded system offers **90% cost savings** while providing real-time, multi-parameter diagnostics and industry-standard communication protocols (Modbus, MQTT).

10.3 Economic Benefits and ROI Justification:

The economic feasibility of this low-cost solution is driven by improved reliability and reduced operational expenditure (OPEX):

- **Downtime Reduction:** By enabling early fault detection (e.g., vibration spikes before catastrophic bearing failure), the system shifts the maintenance model from reactive to predictive, significantly reducing unplanned downtime, which is the largest cost in manufacturing.

- **Optimized Maintenance Scheduling:** Maintenance can be performed only when required (condition-based maintenance) rather than on a fixed, time-based schedule, saving labor costs and extending the lifespan of components.

REFERENCES

1. **G. B. Kliman and J. Stein, "Methods of motor current signature analysis," Elect. Mach. Power Syst., vol. 20, no. 5, pp. 463–474, Sept. 1992**

   This paper explains **Motor Current Signature Analysis (MCSA)**, where motor faults are detected by analysing stator current. This method is non-intrusive and supports current-based monitoring used in the SCADA system.

2. **G. B. Kliman, "What stator current processing-based technique to use for induction motor rotor faults diagnosis?" IEEE Trans. Energy Conversion, vol. 18, pp. 238–244, June 2003.**

   The paper compares different **current signal processing techniques** for rotor fault detection. It highlights the importance of proper data processing for accurate fault diagnosis in monitoring systems.

3. **C. Hargis, "The detection of rotor defects in induction motors," in Proc. IEE Int. Conf. Electrical Machines, Design and Application, London, U.K., 1982, pp. 216–220.**

   This research explains how rotor defects affect motor performance and can be identified through electrical parameter monitoring such as current and speed.

4. **M. E. H. Benbouzid, "Monitoring and diagnosis of induction motors electrical faults using a current Park's vector pattern approach," IEEE Trans. Ind. Applicat., vol. 36, pp. 730–735, May/June 2000.**

   This paper introduces the Park's Vector Approach for detecting electrical faults using stator current analysis. It supports current-based fault detection used in SCADA monitoring.

5. **M. Karimi-Ghartemani and A. K. Ziarani, "Periodic orbit analysis of two dynamical systems for electrical engineering applications," J. Eng. Math., vol. 45, no. 2, pp. 135–154, 2003.**

   This study presents methods for analyzing dynamic behavior in electrical systems, which can be useful for advanced fault analysis and future predictive maintenance using SCADA data.