

# **IoT Based Industry Protection System**

**Atharv S. Samgir, Krishna P. Bobade, Yash S. Bhosale, Avadhut D. Shinde**

Department of Electronics and Telecommunication

Jayawantrao Sawant Polytechnic, Pune, India

atharvsamgir04@gmail.com, yashbhosal265@gmail.com

bobadekrishna51@gmail.com, avadhutshinde977@gmail.com

**Abstract:** *The IOT industry protection system using NodeMCU controller is a system designed to protect industries from losses due to accidents using Internet of things. Gas leakages may lead to fires leading to huge industrial losses, also instant fire detection is needed in case of furnace blasts or other conditions. Also low lighting in industries may create improper work conditions increasing the probability of accidents. The system makes use of NodeMCU Controller to achieve this functionality. The system makes use of temperature sensing along with light and gas sensing to detect fire, gas leakage as well as low lighting to avoid any industrial accidents and prevent losses. The system consists of light, gas and temperature sensors interfaced with NodeMCU controller and LCD screen. The sensor data is constantly scanned to record values and check for fire, gas leakage or low light and then this data is transmitted online. The Wifi module is used to achieve internet functionality. The IOT server then displays this information online, to achieve the desired output. Keywords : NodeMcu, Temperature Sensor, Flame Sensor, Smoke Sensor, IoT, Relay*

*Problem: Traditional industrial safety systems are often reactive and lack real-time remote monitoring, leading to delayed responses during gas leaks, fires, or machine overheating.*

*Solution: This paper proposes an IoT-based framework that integrates various sensors with a microcontroller to monitor environmental and machine health parameters.*

*Method: Data is collected via sensors, processed locally, and transmitted to a cloud platform for real-time visualization and automated alerts.*

*Result: The system ensures a 24/7 surveillance mechanism, reducing human intervention and minimizing industrial downtime.*

**Keywords:** *NodeMCU controller*

## **I. INTRODUCTION**

As the society is growing with various developments, the outmoded forms of storing various food products in cold store rooms is failing to satisfy human needs. As the society is growing with various developments, the outmoded forms of storing various food products in cold store rooms fails to satisfy human need. Through the monitoring of the temperature and humidity inside cold storage rooms, the goodness of the products can be ensured for a longer time. Recent research has revealed that operations of wireless sensor systems are largely affected by their on-board temperature . We can implement sensors in wide area over the machines and instruments and control and monitor the circumstances by using concept of IoT . In this paper we have designed a gateway which will be the central part of this whole system. The function of the gateway is to gather data, process them, upload them and process user control information. If the network connection is not established then the data will be stored and upon reestablishment of the network it will be uploaded. The terms of "things" in the IoT vision is very broad and includes a variety of physical elements. The terms of things include portable personal items such as smart phones, tablets and digital cameras. With the rapid increase in the number of user of and digital cameras. Furthermore, IoT includes elements of our environments (be it home, car or office), and things equipped with RFID tags connected to a gateway device. From those mentioned so far, a huge number of devices and things will be connected to the Internet, each providing data and information and some even services.



## II. SYSTEM OVERVIEW

The IoT-Based Industry Protection System represents a sophisticated fusion of embedded hardware and cloud-based intelligence designed to revolutionize industrial safety. At its core, the system functions as a decentralized, 24/7 digital watchdog that monitors high-risk environmental factors such as gas leaks, fire outbreaks, and unauthorized human presence within hazardous zones. Unlike traditional safety systems that rely on loud, local alarms which can go unnoticed in empty facilities, this framework utilizes a multi-layered sensor array to "feel" the industrial environment in real-time. By leveraging a high-performance microcontroller at the edge, the system can differentiate between minor atmospheric fluctuations and genuine emergencies, ensuring that safety protocols are only triggered when necessary.

The true innovation of this project lies in its Dual-Response Architecture. When a sensor threshold is breached—such as a sudden spike in methane levels or an infrared fire signature—the system executes a localized, hardware-level interrupt. This means the system can physically disconnect machinery or activate ventilation fans in milliseconds, operating independently of the internet or human intervention to prevent an immediate catastrophe. Simultaneously, the system utilizes the MQTT protocol to transmit this critical event data to a cloud management platform. This cloud integration serves as a global notification hub, instantly alerting plant managers via encrypted mobile apps or SMS while logging the data into a "Digital Twin" dashboard for long-term forensic analysis and predictive maintenance.

Ultimately, this system transforms industrial protection from a reactive, manual process into a proactive, intelligent infrastructure. It bridges the gap between physical factory floors and digital oversight, providing a fail-safe environment that is both cost-effective and highly scalable. By reducing the response time between a disaster's onset and corrective action to nearly zero, this IoT framework offers a robust solution for the rigorous safety demands of Industry 4.0, ensuring that human lives and expensive industrial assets are protected through autonomous, data-driven decision-making.

## III. METHODOLOGY

The development of the Industrial Protection System follows a modular framework designed to bridge the gap between physical environmental monitoring and remote digital supervision. The architecture is strategically divided into four distinct layers: the perception layer, the processing layer, the communication layer, and the application layer.

### Perception and Data Acquisition

The foundational stage involves the deployment of a heterogeneous sensor array strategically placed throughout the industrial facility. To mitigate risks associated with common industrial hazards, the system incorporates an MQ-2 gas sensor for detecting combustible gases and smoke, a DHT11 or LM35 sensor for thermal monitoring, and a Light Dependent Resistor (LDR) to ensure adequate illumination for worker safety. These sensors act as the "eyes" of the system, converting physical environmental changes into measurable electrical signals. The sensitivity of these sensors is calibrated during the initialization phase to establish baseline environmental conditions, ensuring that even minor deviations from the norm are captured with high fidelity.

### Centralized Processing and Logic Execution

At the core of the hardware architecture is a microcontroller, typically an Arduino Uno or ESP32, which functions as the primary processing unit. The methodology employs a threshold-based decision logic programmed within the controller's firmware. As the analog signals are received from the perception layer, the controller performs an Analog-to-Digital Conversion (ADC) and compares the real-time values against predefined safety limits (e.g., a temperature limit of 45°C or a gas concentration threshold of 200 ppm). If the environment remains within safe parameters, the system continues its routine polling. However, if a threshold is breached, the microcontroller executes an immediate local response—triggering an audible buzzer and activating relays to engage exhaust fans or emergency lighting—to provide instant onsite mitigation before human intervention is possible.

### Connectivity and Cloud Integration

To facilitate remote monitoring, the system utilizes a communication bridge, often through an ESP8266 Wi-Fi module. The methodology emphasizes the use of the MQTT (Message Queuing Telemetry Transport) or HTTP protocol to transmit encapsulated data packets to a cloud-based IoT platform, such as ThingSpeak or Blynk. This stage is critical for



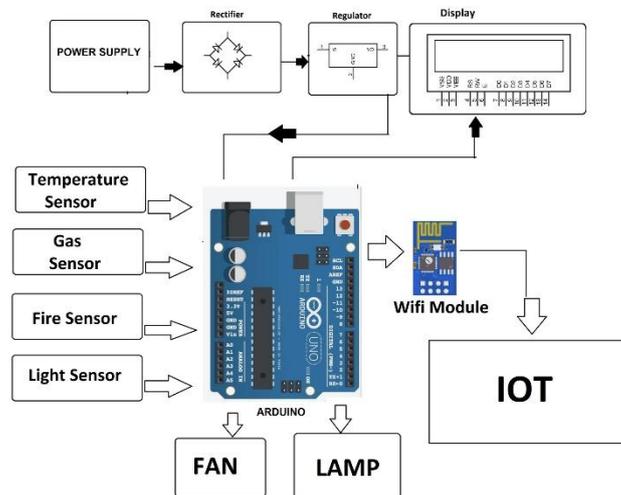
transforming local data into global information. By offloading the data to the cloud, the system overcomes the limitations of local storage, allowing for historical data logging and complex analytical processing that can predict potential equipment failures before they manifest as hazards.

### User Interface and Remote Visualization

The final layer of the methodology involves the development of a graphical user interface (GUI) that serves as the end-user dashboard. This interface visualizes the live data streams through interactive charts and indicators, providing safety officers with a real-time overview of the plant's status. Furthermore, the integration includes a notification service that pushes alerts via Email or SMS if a critical fault is detected. This ensures that stakeholders are informed of industrial anomalies regardless of their physical proximity to the site, completing the loop from physical sensing to actionable intelligence.

## IV. BLOCK DIAGRAM AND EXPLANATION

### 1. Power Supply Section



Power Supply → Rectifier → Regulator → Display & Arduino

Power Supply: Provides AC input.

Rectifier: Converts AC to DC using a bridge rectifier.

Regulator: Maintains a constant DC voltage (usually 5V) for Arduino and LCD.

Display (LCD): Shows temperature, gas level, fire detection status, etc.

### 2. Input Section (Sensors)

Four sensors are connected to the Arduino:

- Temperature Sensor

Measures temperature of the environment.

If temperature increases beyond limit → system can turn ON fan or give alert.

- Gas Sensor

Detects harmful gases (like LPG, methane, smoke).

If gas leakage is detected → alarm/fan activation + IoT alert.

- Fire Sensor

Detects flame or infrared radiation from fire.

If fire detected → emergency alert.



- Light Sensor (LDR)  
Detects light intensity.  
Can automatically control lamp based on brightness.

### 3. Processing Section (Arduino)

- Arduino UNO is the brain of the system.  
Reads data from all sensors.  
Compares values with preset threshold limits.  
Makes decisions.  
Sends output to:
- LCD display
  - Fan
  - Lamp
  - WiFi module

### 4. Output Section

- Fan  
Turns ON when:  
Temperature is high  
Gas detected
- Lamp  
Turns ON/OFF based on:  
Light intensity

### 5. Communication Section (WiFi Module → IoT)

- WiFi module (like ESP8266) connects Arduino to the internet.  
Sends real-time sensor data to IoT platform.  
User can monitor system remotely on mobile or computer.  
Alerts can be sent in case of fire or gas leakage.  
Working Principle (Step-by-Step)  
Sensors continuously monitor environment.  
Arduino reads sensor values.  
If values exceed safety limits:  
Fan or lamp activates.  
Alert message sent via WiFi.  
LCD displays current readings.  
Data is uploaded to IoT cloud for remote monitoring.

## V. ADVANTAGES

- Real-Time Monitoring  
Continuously monitors temperature, gas, fire, and light conditions without human supervision.  
Early Hazard Detection  
Detects gas leakage, fire, or overheating at an early stage, reducing risk of accidents.  
Remote Access  
Using WiFi and IoT, data can be monitored from anywhere through mobile or computer.  
Automatic Control



Fan and lamp operate automatically based on sensor readings. No manual operation required.

Increased Safety

Improves safety in industries, laboratories, and homes by providing instant alerts.

Cost Effective

Arduino and sensors are low-cost compared to industrial safety systems.

Low Power Consumption

Consumes less power due to efficient microcontroller and sensor operation.

Expandable System

More sensors (humidity, motion, pressure, etc.) can be added easily in future.

User-Friendly

LCD display shows real-time data clearly.

Reduces Human Error

Automation minimizes dependency on manual monitoring.

## **VI. HARDWARE COMPONENTS**

### **1. Arduino UNO**

Main controller of the system.

Reads sensor data

Processes signals

Controls fan, lamp, and WiFi module

### **2. Temperature Sensor (e.g., LM35 / DHT11)**

Measures surrounding temperature.

Provides analog or digital output

Used for overheating detection

### **3. Gas Sensor (e.g., MQ-2 )**

Detects LPG, methane, smoke, etc.

Gives analog output

Used for gas leakage detection

### **4. Fire Sensor (Flame Sensor Module)**

Detects flame using infrared light.

Digital output

Used for fire safety

### **5. Light Sensor (LDR – Light Dependent Resistor)**

Measures light intensity.

Controls automatic lamp operation

### **6. WiFi Module (ESP8266)**

Provides internet connectivity.

Sends sensor data to IoT platform

Enables remote monitoring



7. LCD Display (16×2)

Displays temperature, gas level, fire status, etc.

8. Fan

Used for ventilation when temperature or gas level is high.

9. Lamp

Used for automatic lighting control.

10. Power Supply Unit

Includes:

Transformer (if AC supply used)

Bridge Rectifier

Voltage Regulator (like 7805)

Provides regulated 5V DC supply.

11. Supporting Components

Resistors

Capacitors

Relay Module (for controlling fan/lamp)

Connecting wires

Breadboard or PCB

## VII. LIMITATIONS

### Internet Dependency

Remote monitoring depends on WiFi. If the internet fails, IoT updates and alerts will not work.

### Sensor Accuracy

Low-cost sensors (like MQ series or basic flame sensors) may give inaccurate readings or false alarms.

### Limited Range

WiFi module works within limited network coverage.

### Power Failure Issue

If power supply fails and no backup is available, the system stops working.

### Not Suitable for Heavy Industrial Use

Arduino-based systems are good for small-scale projects, but not ideal for large industrial automation without industrial-grade controllers.

### Maintenance Required

Sensors require periodic calibration and cleaning for accurate performance.

### Security Risk

IoT systems can be vulnerable to hacking if proper network security is not implemented.



#### Limited Processing Capability

Arduino has limited memory and processing speed compared to advanced industrial systems.

### IX. FUTURE SCOPE

#### Integration with AI

Artificial Intelligence can be added to predict fire, gas leakage, or overheating before they become serious hazards.

#### Cloud Data Analytics

Sensor data can be stored in the cloud for long-term analysis, helping industries understand patterns and improve safety planning.

#### Mobile Application Development

A dedicated Android app can be developed for real-time monitoring, alerts, and system control.

#### SMS and Email Alerts

System can be upgraded to send automatic SMS, email, or push notifications during emergency conditions.

#### Use of Industrial Controllers

For large industries, Arduino can be replaced with PLC or industrial IoT controllers for higher reliability.

#### Addition of More Sensors

Future expansion may include:

- Humidity sensor
- Smoke sensor
- Motion sensor
- Pressure sensor
- Water level sensor

#### Solar Power Integration

System can be powered using solar panels for energy-efficient and remote area applications.

#### Smart Automation Integration

Can be integrated with smart building systems for complete industrial automation.

### X. CONCLUSION

The IoT-based Industry Protection System provides an effective and low-cost solution for monitoring industrial safety parameters such as temperature, gas leakage, fire, and light intensity. The system continuously collects data through sensors, processes it using the Arduino controller, and takes automatic action when abnormal conditions are detected.

With the help of WiFi connectivity, real-time data can be monitored remotely, which improves response time during emergencies. The automatic control of devices like fan and lamp reduces manual effort and increases overall safety.

Although it has some limitations such as dependency on internet and basic sensor accuracy, it is highly suitable for small-scale industries, laboratories, and smart safety applications.

Overall, this system enhances industrial safety, reduces risks, and demonstrates how IoT technology can be used for smart and efficient protection solutions.



**REFERENCES**

- [1]. Banzi, M., & Shiloh, M., Getting Started with Arduino, Maker Media.
- [2]. Raj Kamal, Internet of Things: Architecture and Design Principles, McGraw Hill Education.
- [3]. K. V. K. K. Prasad, Embedded Systems and Applications, Dreamtech Press.
- [4]. Official Arduino Documentation – <https://www.arduino.cc> □
- [5]. ESP8266 WiFi Module Documentation – <https://www.espressif.com> □
- [6]. IoT Tutorials and Projects – <https://www.electronicwings.com> □
- [7]. Sensor Datasheets (MQ Gas Sensor, LM35, Flame Sensor) – Available on manufacturer websites.
- [8]. Research articles on “IoT Based Industrial Safety System” available on IEEE Xplore.
- [9]. Papers on “Smart Industry Monitoring using IoT” available on Google Scholar.

