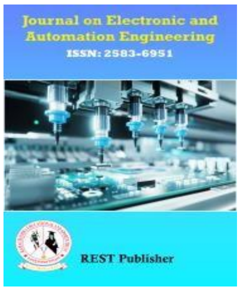
Journal on Electronic and Automation Engineering

Vol: 4(2), June 2025

REST Publisher; ISSN: 2583-6951 (Online)

Website: https://restpublisher.com/journals/jeae/

DOI: https://doi.org/10.46632/jeae/4/2/53



Cloud Based Automation for Industrial Ambient Robotic System

***Sikender Ali Khan, Addenki Sairam, A. Srinath, A. Manikanth**

Nalla Malla Reddy Engineering College, Hyderabad, Telangana, India.

*Corresponding Author Email: sikender.ece@nmrec.edu.in

Abstract: *The Cloud-Based Automation of Industrial Ambient Robotic System is designed to enhance industrial automation, safety, and remote monitoring by integrating IoT, cloud computing, and real-time sensor data processing. The system utilizes a Raspberry Pi zero as the central processing unit while incorporating an ESP32 microcontroller to interface with environmental sensors. The ESP32 is responsible for collecting data from DHT11 (temperature & humidity sensor), MQ135 (gas detection sensor), flame sensor, and a buzzer. The remaining components, including ultrasonic sensors, servo motors, relays, and motor driver (L298N) for robotic navigation and obstacle avoidance, remain controlled by the Raspberry Pi. The system employs edge computing to locally process sensor data, reducing cloud dependency and improving response time for critical safety measures. This Project presents the design and development of a cloud-based automated industrial ambient robotic system using an Raspberry pi and ESP32 as the main microcontrollers. The robotic system is controlled via with raspberry pi, which connects to the Adafruit cloud platform, allowing for remote control and automation of industrial processes. This cloud-based approach enhances safety, operational efficiency, and reduces the risks associated with industrial hazards. Cloud-based systems can store vast amounts of data from sensors, machines, and robots in real time. A cloud-based automation industrial ambient robotic system integrates robotics, IoT (Internet of Things), and cloud computing to create an intelligent, scalable environment for manufacturing and logistics operations.*

Keywords: *Cloud-Based Automation, Industrial Robotics, Raspberry pi zero, Real-Time Monitoring, MQTT Protocol, AWS IoT, Firebase, Obstacle Detection, Predictive Maintenance, AI-Based Automation, Environmental Sensors, Remote Monitoring, Autonomous Navigation, Motor Control, Industrial Safety, Secure Data Transmission, Embedded Systems.*

1. INTRODUCTION

In the era of Industry 4.0, automation and robotics are revolutionizing industrial processes by enabling intelligent monitoring, remote accessibility, and autonomous operation. Traditional industrial environments depend on manual supervision, which can lead to higher labour costs, inefficiencies, and safety risks. With advancements in IoT (Internet of Things), cloud computing, edge processing, and artificial intelligence (AI), industries are now shifting towards smart and automated systems. The Cloud Based Automation Industrial Ambient Robotic System is designed to integrate sensor-based monitoring, real-time data processing, and remote control capabilities to enhance industrial efficiency and safety. This system utilizes a Raspberry Pi zero and ESP32 as the central processing unit, interfacing with multiple environmental sensors, motors, and actuators to collect and process data in real-time. By leveraging cloud-based communication protocols such as MQTT (Message Queuing Telemetry Transport), AWS IoT, and Firebase, the system allows remote monitoring and control from a web-based dashboard or mobile application. Furthermore, edge computing techniques are employed to reduce dependency on cloud processing, ensuring faster response times for critical automation tasks. The ESP32 is responsible for acquiring real-time environmental data from DHT11 (temperature & humidity sensor), MQ135 (gas sensor), and a flame sensor, while the Raspberry Pi processes other industrial automation tasks, including robotic navigation and obstacle avoidance using ultrasonic sensors, servo motors, and motor drivers (L298N). Importance of Industrial Automation Modern industries face challenges such as delayed decision-making, inefficient resource utilization, safety hazards, and high operational costs. Automation plays a crucial role in optimizing industrial workflows, reducing downtime, and improving overall productivity. By integrating intelligent robotics and IoT-enabled monitoring, industries can achieve real-time decision-making, predictive maintenance, and enhanced safety mechanisms. A cloud-connected robotic system offers several advantages, including: Remote accessibility: Supervisors can monitor industrial operations from any location using web-based dashboards Real-time alerts and automation: The system can detect gas leaks, fire, or environmental hazards and trigger automatic safety mechanisms. Predictive maintenance: AI algorithms analyse historical sensor data to predict equipment failures before they occur, minimizing downtime. Improved safety: The robot can navigate autonomously and avoid hazardous areas, reducing the risk of accidents. With the increasing adoption of smart manufacturing and intelligent automation, this project provides a scalable and efficient robotic solution for various industrial applications.

2. LITERATURE SURVEY

The field of industrial automation and robotics has evolved significantly with the integration of IoT, cloud computing, artificial intelligence, and edge computing. Various research studies and technological advancements have contributed to the development of smart industrial systems that are capable of remote monitoring, autonomous decision-making, and predictive maintenance. This chapter reviews existing literature on industrial IoT (IIOT), sensor-based environmental monitoring, robotic automation, cloud based control systems, and AI-driven predictive maintenance, providing insights into the technologies and methodologies relevant to this project. Cloud-Based Industrial Automation The adoption of cloud computing in industrial automation has enabled real-time monitoring, data analytics, and remote control of industrial processes. According to Patel and Sharma (2021), cloud integrated automation systems reduce downtime and improve operational efficiency by allowing industries to analyse real-time data from various sensors. AWS IoT, Google Cloud, and Firebase have emerged as leading cloud platforms for industrial automation, offering scalability, security, and real-time data exchange (Gupta & Nair, 2020). In a study conducted by Singh and Verma (2020), it was observed that cloud-based IoT systems provide predictive analytics, which helps in detecting failures before they occur. Their research demonstrated how real-time sensor data from industrial environments could be stored and processed in the cloud, allowing for automated alerts and proactive maintenance. These findings support the decision to integrate cloud services (AWS IoT, Firebase) in this project for real-time monitoring and remote control of the robotic system. IoT-Based Sensor Integration for Industrial Monitoring The use of sensors for environmental monitoring in industrial settings has been extensively researched. DHT11, MQ135, ultrasonic, and flame sensors are commonly used for temperature, humidity, gas detection, and fire hazard monitoring. Bose (2019) discusses how sensor fusion technology improves industrial safety, ensuring that multiple data points are cross-validated before triggering alerts. A study by Kumar and Desai (2019) demonstrated that gas and air quality sensors like MQ135 are effective in detecting hazardous gases such as CO_2, NH_3, and NO_x. The study highlighted the importance of calibrating gas sensors periodically to maintain accuracy. Similarly, flame sensors have been used in fire detection systems, but their effectiveness depends on environmental conditions such as ambient light interference (Sharma & Bose, 2021). Edge Computing for Real-Time Decision-Making Cloud computing offers many advantages, but real-time processing constraints make it necessary to implement edge computing for faster response times. According to Gupta and Nair (2020), edge computing reduces latency by processing data locally on embedded systems like Raspberry Pi and ESP32 before transmitting essential data to the cloud. Research conducted by Patel and Desai (2020) on ultrasonic sensor integration in robotic systems demonstrated that local processing of distance measurements significantly improves obstacle avoidance. By implementing edge computing techniques, the robotic system in this project can quickly respond to obstacles, detect environmental hazards, and operate autonomously without depending entirely on cloud processing. MQTT Protocol for IoT Communication Efficient and reliable data transmission is crucial for cloud-integrated industrial automation. The MQTT protocol has been widely used in low-power, high-efficiency IoT systems due to its lightweight design and fast data exchange (Express if Systems, 2022). According to Sharma and Bose (2021), MQTT based industrial automation systems demonstrate better performance compared to traditional HTTP-based communication, especially in environments requiring real-time alerts and remote access. The study by AWS (2023) highlights how MQTT's publish-subscribe architecture enhances scalability by allowing multiple devices to communicate simultaneously without direct connections. Given these advantages, the implementation of MQTT in this project ensures efficient real-time communication between sensors, actuators, cloud platforms, and user dashboards. AI and Machine Learning for Predictive Maintenance Predictive maintenance is a key aspect of modern industrial automation. Studies have shown that AI based fault detection can reduce machine downtime and operational costs. Bose (2021) proposed an AI based model that analyses sensor data patterns to predict equipment failures before they occur. Sharma and Bose (2021) implemented machine learning techniques in IoT systems to detect temperature variations and abnormal gas concentrations, allowing industries to take preventive action before safety thresholds are crossed. In a related study, Gupta and Nair (2020) explored how neural networks can be trained on historical sensor data to optimize robotic movements and improve navigation accuracy. This research supports the integration of machine learning algorithms in this project, allowing the system to analyse past sensor readings and improve its performance over time. Autonomous Robotics for Industrial Applications Industrial robots are becoming increasingly autonomous due to advances in AI, sensor technology, and motor control systems. Kumar and Desai (2019) demonstrated how robotic systems integrated with ultrasonic sensors could navigate dynamic environments with minimal human intervention. Sharma and Bose (2021) further explored how servo motors and L298N motor drivers enable precise robotic movement and automated material handling. By leveraging autonomous navigation techniques, this project enables the robotic system to avoid obstacles, respond to environmental changes, and execute automated tasks efficiently. The combination of sensor-driven decision-making, AI-based predictions and cloud-controlled automation enhances the system's adaptability and operational efficiency. The reviewed literature provides a strong foundation for developing a cloud-based industrial automation robotic system. Based on this review, the proposed system integrates real-time sensor monitoring, cloud communication, AI-based analytics, and autonomous navigation, making it an efficient and scalable solution for industrial automation. The next chapter focuses on system design, hardware components, and software implementation to achieve the objectives outlined in this study. Existing System: The proposed system is capable of detecting smoke, different flammable gases and fire. This system is capable of providing hazard location coordinates to the nearby fire department. The smoke detection sensor MQ-2 is used to detect the smoke, the Flame detection sensor is used to sense the flame, the flammable gas sensor MQ-5 is used to detect the gases like LPG/LNG and the GPS module is to obtain device location. These sensors along with Wi-Fi micro-controller are connected to a MQTT broker via Internet through which it communicates hazard status to the nearest fire-fighting organizations.

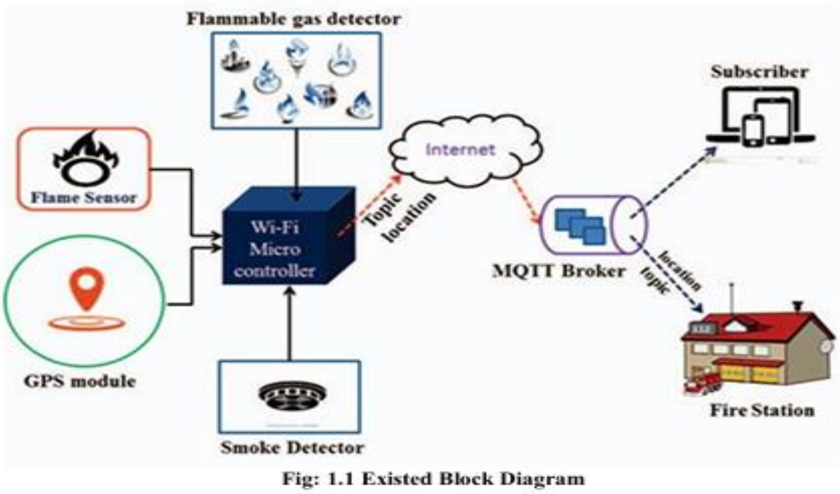


FIGURE 1. Existing Block Diagram

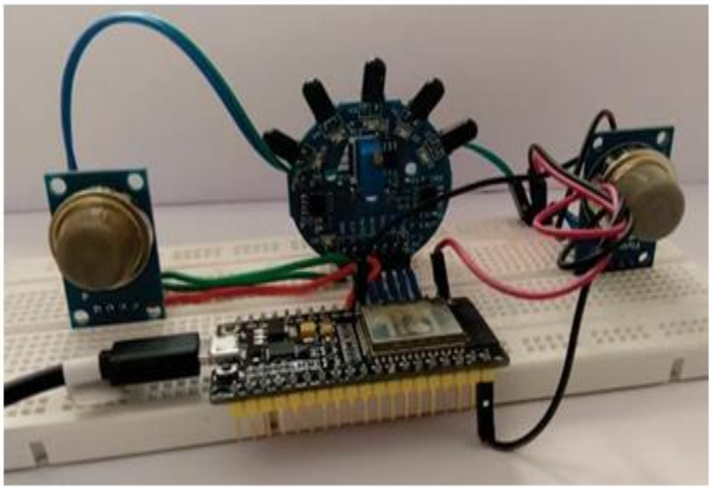


FIGURE 2. The Overall Circuit Connections of the System

After connecting each and every device in the desired manner, and making sure that each and every component is connected in accordance with the other components. An external power supply is given to ESP32 board with USB for current to flow. The sensors detect the hazard if any smoke or any flammable gas or any flame in the surroundings. Here the GPS module gives the hazard location in Decimal Degree format. The sensed data and the location co-ordinates are published on to the Arduino.IO cloud service. The system utilizes the light weighted protocol MQTT services to provide fast responses so as to provide emergency rescue operations immediately. The information along with hazard location in decimal degrees is published on to Adafruit.IO along with hazard status which is subscribed by social organizations.

Proposed System:

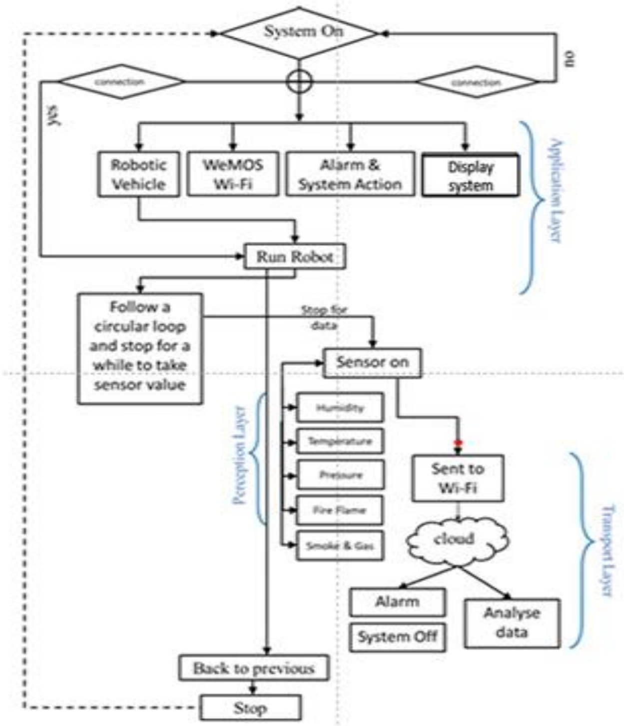


FIGURE 3. Generalized Block Diagram

The proposed system integrates gas, fire, and temperature sensors into a cloud-based platform, using Raspberry pi for remote control via the Adafruit cloud. In case of hazardous conditions, real-time alerts are sent to an authorized mobile number through GSM. This approach enables remote monitoring, faster response times, and improved safety by automating detection and control, reducing reliance on manual intervention in industrial environments. The set of rules has to be followed, is described in figure. The three types of a layer are described with this algorithm, they are application layer, perception layer and transport layer. The algorithm shows how the entire system runs. Initially when the system is on it requires to be well connected. When the connection is on, came to the application layer, where a robotic vehicle, WeMOS Wi-Fi, alarm and system action, display system will be active. Then in the application part robot will run and will follow a circular loop. This vehicle will stop for taking data from sensors. In perception layer humidity, temperature, pressure, fire and gas sensors will be checked. Then the final part transport later comes, where collected data and possibilities will be sent to a cloud network through Wi-Fi. An alarm system will also be connected through Wi-Fi that will allow giving an alert when the situation is harmful and from here particular problematic systems can be turned off. Data stored to IoT based network can be useful for future analysis. The proposed system integrates gas, fire, and temperature sensors into a cloud-based platform, using Raspberry pi for remote control via the Adafruit cloud. In case of hazardous conditions, real-time alerts are sent to an authorized mobile number through GSM. This approach enables remote monitoring, faster response times, and improved safety by automating detection and control, reducing reliance on manual intervention in industrial environments. The set of rules has to be followed, is described in figure .The three types of a layer are described with this algorithm, they are application layer, perception layer and transport layer. The algorithm shows how the entire system runs. Initially when the system is on it requires to be well connected. When the connection is on, came to the application layer, where a robotic vehicle, WeMOS Wi-Fi, alarm and system action, display system will be active. Then in the application part robot will run and will follow a circular loop. This vehicle will stop for taking data from sensors. In perception layer humidity, temperature, pressure, fire and gas sensors will be checked. Then the final part transport later comes, where collected data and possibilities will be sent to a cloud network through Wi-Fi. An alarm system will also be connected through Wi-Fi that will allow giving an alert when the situation is harmful and from here particular problematic systems can be turned off. Data stored to IoT based network can be useful for future analysis.

Project Overview: The primary goal of this project is to develop a cloud-integrated, intelligent robotic system that enhances industrial automation by enabling real-time environmental monitoring, remote control, and automated decision-making. The specific objectives include: To design a cloud-based industrial robotic system for real-time monitoring and automation. To interface environmental sensors (DHT11, MQ135, flame sensor, ultrasonic sensor) for detecting temperature, humidity, gas concentration, fire hazards, and obstacles. To implement real-time data processing using edge computing for faster response and reduced cloud dependency. To establish a secure cloud communication framework (AWS IoT, Firebase, MQTT) for remote monitoring and data logging. The system consists of three major components: sensors, actuators, and cloud-based monitoring. Sensors such as DHT11, MQ135, flame detector, and ultrasonic sensor continuously collect data from the environment. This data is processed locally by the Raspberry Zero and ESP32 which analyses sensor inputs, controls motor movements, and triggers actuators in response to critical conditions. The motor driver (L298N) enables precise movement control, while servo motors handle minor robotic adjustments. The system utilizes cloud-based services (AWS IoT, Firebase) to store sensor data and allow remote access. The MQTT protocol facilitates efficient data transmission, ensuring real-time communication between the robotic system and remote users. A web-based dashboard and mobile application provide an intuitive interface for monitoring sensor data and controlling robotic operations. Additionally, the system integrates AI-based predictive analytics to analyse historical sensor readings and detect potential faults in industrial equipment. By leveraging machine learning models, the system can proactively identify rising gas levels, abnormal temperature variations, and motor performance issues, allowing preventive measures to be taken before failures occur.

Block Diagram and Its Description

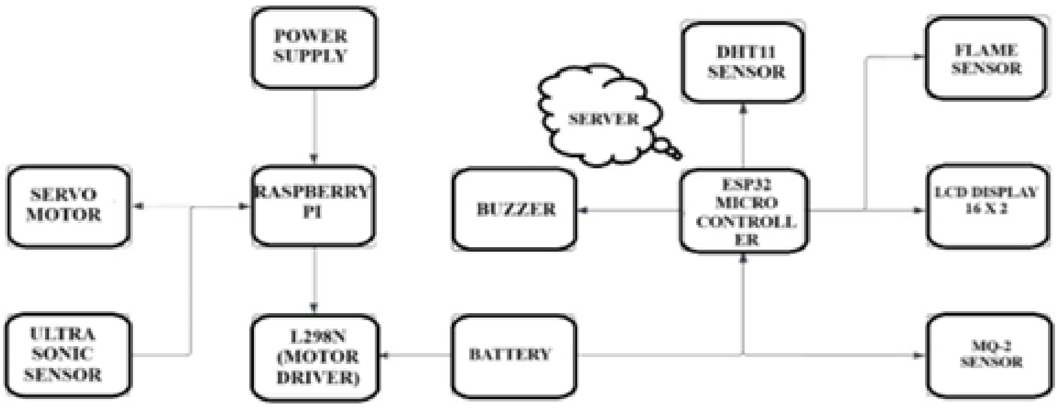


FIGURE 4. Block Diagram

Components Overview: The Cloud-Based Automation Industrial Ambient Robotic System consists of various hardware components, each performing a crucial role in monitoring, processing, automation, and cloud connectivity. The system integrates sensors for environmental monitoring, actuators for robotic movement, and cloud-based communication modules to achieve real-time industrial automation. Below is a detailed description of each component used in the system. Raspberry Pi Zero (Main Processing Unit): The Raspberry Pi zero serves as the central controller in the system, responsible for sensor data acquisition, motor control, automation logic, and cloud communication. It is a powerful single-board computer (SBC) that enables edge computing and real-time decision-making without constant cloud dependence. Specifications: Processor: Broadcom BCM2837B0, Cortex-A53 (64-bit quad-core, 1.4 GHz). RAM: 1GB LPDDR2 SDRAM. Wireless Connectivity: Built-in Wi-Fi (2.4GHz/5GHz) and Bluetooth 4.2. GPIO Pins: 40-pin header for sensor and actuator interfacing. Operating System: Raspberry Pi OS (Linux-based).

ESP32 – Microcontroller (Main Processing Unit): The ESP32 is a low-cost, low-power system-on-chip (SoC) microcontroller developed by Express if Systems. It is widely used in IoT (Internet of Things) applications due to its powerful features, including Wi-Fi and Bluetooth connectivity.ESP32 is popular among hobbyists and professionals due to its high performance, low cost, and extensive support in Arduino IDE, Micro Python, and ESP-IDF. The ESP32 is a powerful and versatile microcontroller widely used in IoT, automation, and industrial applications due to its high processing power, built-in Wi-Fi/Bluetooth capabilities, and multiple GPIOs for sensor interfacing. Specifications: Processor: Dual-core 32-bit Xtensa LX6 microprocessor, running at 160 or 240 MHz. Memory: 520 KiB RAM, 448 KiB ROM. Wireless Connectivity: Wi-Fi (802.11 b/g/n), Bluetooth (4.2 BR/EDR and BLE). Peripheral Interfaces: 34 programmable GPIOs, 10 touch sensors, 2 x 12-bit ADCs, 2 x 8-bit DAC, and more. Communication Interfaces: 4 x SPI, 2 x I^2S, 2 x I^2C, 3 x UART. Operating Voltage: 3.3V. Data Rate: Up to 150 Mbps. Output Power: 20.5 dBm. DHT11 Temperature & Humidity Sensor: The DHT11 sensor is used to measure temperature and humidity in the industrial environment, providing essential data for climate monitoring and safety automation. Specifications: Temperature Range: 0°C to 50°C (±2°C accuracy). Humidity Range: 20% to 90% RH (±5% accuracy). MQ135 Gas Sensor: The MQ135 sensor is used for air quality monitoring, detecting the presence of toxic gases and pollutants in an industrial environment. A gas sensor is an electronic device designed to detect the presence and concentration of specific gases in the environment. In the context of industrial automation, gas sensors play a critical role in monitoring air quality and detecting potentially hazardous conditions, such as gas leaks or toxic fumes, ensuring safety and compliance with environmental standards. Specifications: Detectable Gases: CO2, NH3, NOx, benzene, smoke, and alcohol vapours. Operating Voltage: 5V. Analog Output: Provides variable resistance based on gas concentration. Flame Sensor: The Flame Sensor is used for fire detection, ensuring industrial safety by monitoring infrared radiation from flames. Specifications: Wavelength Detection: 760 – 1100 nm (IR light spectrum). Operating Voltage: 3.3V - 5V. Ultrasonic Sensor (HC-SR04) – Obstacle Detection: The ultrasonic sensor helps in robotic navigation by measuring the distance between the robot and nearby obstacles. Specifications: Measurement Range: 2 cm to 400 cm. Accuracy: ±3 mm. Operating Voltage: 5V. Motor Driver (L298N) – Motion Control: The L298N Motor Driver controls DC motors, enabling robotic movement. This L298N Motor Driver Module is a high power motor driver module for driving DC and Stepper Motors. This module consists of an L298 motor driver IC and a 78M05 5V regulator. L298N Module can control up to 4 DC motors, or 2 DC motors with directional and speed control. Specifications: Operating Voltage: 5V – 12V. Motor Channels: Dual H-Bridge for controlling two motors. PWM Control: Adjusts motor speed and direction. Drives robotic movement in industrial areas. Enables forward, backward, and turning operations. Supports automated and manual control modes. Buzzer : Function: Sounds an alarm when safety conditions are violated. Use Case: Alerts workers during fire, gas leaks, or system failures. Relay Module: Controls external appliances like ventilation systems, alarms, and emergency shutdown mechanisms. Use Case: Activates industrial safety measures automatically. LCD Display (I2C-Based): The LCD display provides a local interface for real-time monitoring. Specifications: Type: 16x2 or 20x4 character LCD. Interface: I2C communication for easy integration. DC Motors – Motion Control: DC motors are used to drive the robot's movement, allowing it to navigate within an industrial environment for automated tasks, material handling, or monitoring purposes Specifications: Type: 12V DC motor Operating Voltage: 6V – 12V Speed: 150 – 300 RPM (depends on gear ratio) Torque: Varies based on motor type Interface: Controlled using the L298N Motor Driver. Servo Motor – Precision Motion Control: The servo motor provides precise angular motion required for robotic adjustments, sensor positioning, or fine control over moving parts. Specifications: Type: SG90 (9g servo) or MG995 (metal gear servo) Operating Voltage: 4.8V – 6V Angle Range: 0° to 180°. Schematic Diagram And Its Description: The Cloud-Based Automation Industrial Ambient Robotic System is designed using a modular architecture that integrates IoT sensors, actuators, cloud computing, and edge processing. The system consists of three primary layers: Perception Layer (Sensors & Actuators), Processing Layer (Raspberry Pi & ESP32 & Edge Computing), and Cloud Layer (Remote Monitoring & Control). The schematic diagram of the robot system illustrates how all electronic components are connected to automate and monitor industrial ambient conditions. Key sections include: 1. Power Supply: Provides stable voltage using adapters or batteries with regulators and capacitors. 2. Microcontroller (MCU): Acts as the brain, processing sensor data and controlling outputs (e.g., Arduino or Node MCU). 3. Sensors: Detect ambient parameters like temperature, humidity, gas levels, and obstacles using DHT, MQ, or ultrasonic sensors. 4. Actuators: Respond to sensor data through relays, motors, LEDs, or buzzers to perform automated actions. Communication Module: Enables cloud connectivity (via Wi-Fi or Bluetooth) for remote monitoring and control. 6. Wiring & Integration: All components are interconnected with proper grounding and logic to enable smooth automation.

Flowchart:

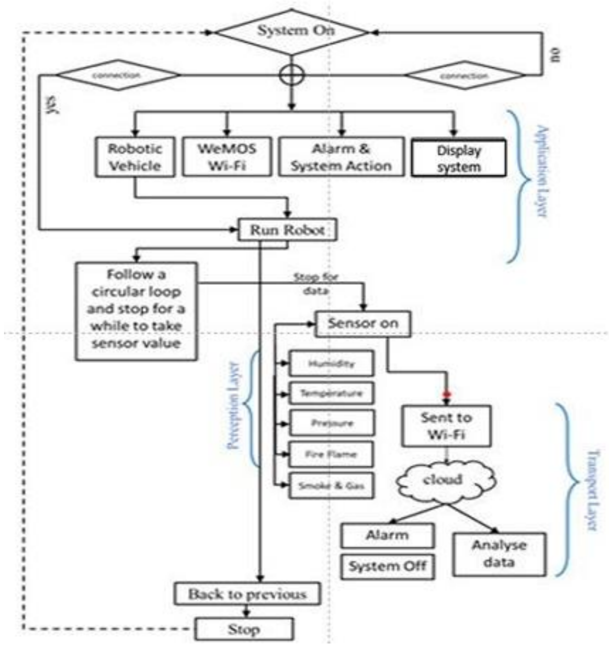


FIGURE 5. Flowchart

System Flow and Data Communication Sensors collect real-time data (temperature, gas, flame, obstacles) and send it to the Raspberry Pi&ESP32. The Raspberry Pi & ESP32 processes sensor readings and determines necessary actions (e.g., triggering alarms, adjusting motors, sending alerts). Data is transmitted to the cloud (AWS IoT/Firebase) via MQTT for remote monitoring and storage. The user can monitor sensor readings and control robotic functions using a web/mobile dashboard. If an anomaly is detected, AI-driven predictive maintenance algorithms analyze trends and notify the user of potential issues. The system automatically adjusts actuators and motors to respond to environmental changes, ensuring safe and efficient industrial operation. Operation Of The System: Perception Layer (Sensors & Actuators) This layer consists of sensors that collect real-time environmental data and actuators that perform actions based on the system's decisions. DHT11 Sensor: Measures temperature and humidity in the industrial environment. MQ135 Gas Sensor: Detects toxic gases and air quality for safety monitoring. Flame Sensor: Identifies fire hazards and triggers safety measures. Ultrasonic Sensor (HC-SR04): Enables obstacle detection and robotic navigation. L298N Motor Driver: Controls robotic movement for navigation and automation. Servo Motor: Adjusts robotic arm or sensor orientation for improved efficiency. Buzzer & Relay Module: Provides alarm notifications and automated safety responses. LCD Display: Displays sensor data and system status locally. 2. Processing Layer (Raspberry Pi & Edge Computing) This layer handles data processing, decision-making, and local automation using edge computing techniques to reduce latency and improve real-time response. Raspberry Pi Zero and ESP 32: Acts as the central processing unit of the robotic system. Edge Computing: Processes sensor data locally to make real-time decisions. Python-Based Algorithms: Manage sensor readings, motor control, and data filtering. AI-Based Predictive Maintenance: Analyses historical data to detect potential failures. MQTT Protocol: Transmits real-time data to the cloud efficiently Cloud Layer (Remote Monitoring & Control) This layer provides cloud storage, remote monitoring, and user interaction through a web or mobile interface. AWS IoT / Firebase: Stores real-time sensor data and system logs for analysis. Web and Mobile Dashboard: Allows remote users to monitor data, control the robot, and receive alerts. MQTT Protocol: Enables real-time communication between the Raspberry Pi and the cloud. Security Measures: Implements SSL/TLS encryption and authentication for secure data transmission. A Cloud-Based Automation Industrial Ambient Robotic System (AIARS) refers to an advanced manufacturing and operational setup where robots and automation systems in an industrial environment are controlled and monitored through cloud computing platforms. This integration enables real-time data processing, remote management, and enhanced efficiency. Here's a breakdown of its components and operations Robotic Systems is an industrial setting might perform tasks like assembly, packaging, material handling, or inspection. These robots are equipped with sensors, actuators, and processor. These systems are integrated with cloud platform that allow them to make decisions, optimize tasks, and adapt to varying conditions in the factory. Ambient Sensors and IoT Devices are ambient sensors collect data from the environment (e.g., temperature, humidity and gas), machine states (e.g., speed, load, temperature), and the robot's own performance metrics. IOT devices help in creating a connected network where robots, machines, and operators can communicate and exchange real-time data. Real-Time Data Processing the data coming from these IoT devices, applying algorithms for predictive maintenance, workflow optimization, and monitoring the industrial sector. The data can be analysed in real-time, providing valuable insights into system performance and allowing for fast decision-making. Automation Control through the cloud interface, operators can remotely control, program, or reconfigure robots, even deploying software updates without needing to be physically present. The cloud system can automatically adjust robot operations based on the data from the sensors, for instance, changing speed or task priority when environmental factors change.

3. HARDWARE DESCRIPTION

Raspberry Pi Zero (Main Processing Unit):

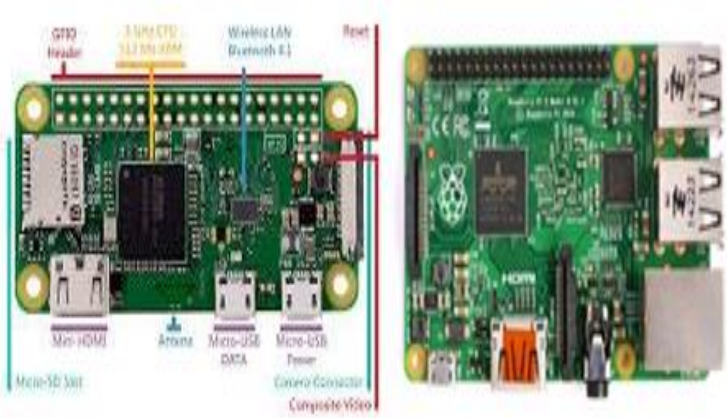


FIGURE 6. Raspberry Pi Zero

Functionality: The Raspberry Pi Zero is programmed to collect, process, and transmit sensor data in real-time. It receives input from environmental sensors, applies decision-making algorithms, and controls actuators such as motors, relays, and buzzers. It also connects the system to cloud platforms (AWS IoT/Firebase), enabling remote monitoring and control via MQTT protocol.

Role in the System: Reads and processes sensor data (temperature, gas, fire, and obstacle detection). Controls robotic movement based on sensor readings. Manages cloud communication for remote monitoring. Executes AI-based predictive maintenance algorithms to enhance system efficiency. Raspberry Pi can browse the internet and stream high-definition video, as well as spreadsheets, word processing, and gaming, just like a desktop computer. The Raspberry Pi can communicate with the outside world and has been used in various digital maker projects, including music machines and parent detectors, weather stations, and tweeting birdhouses with infrared cameras. Role of ESP32 in the Project In this Cloud-Based Automation Industrial Ambient Robotic System, the ESP32 is primarily responsible for: Acquiring real-time sensor data from DHT11 (temperature & humidity sensor), MQ135 (gas sensor), and flame sensor. Transmitting sensor data to the Raspberry Pi 3 B+ via MQTT protocol for further processing and decision-making. Triggering alarms and buzzers in case of critical environmental conditions. Enabling wireless connectivity for cloud integration using AWS IoT and Firebase. Sensor Data Acquisition Interfacing with multiple environmental sensors, including: DHT11 – Temperature & humidity monitoring. MQ135 – Gas detection (air quality monitoring). Flame Sensor – Fire hazard detection. Collecting real-time data and sending it to the Raspberry Pi Zero for further analysis and decision making. Wireless Data Transmission Uses Wi-Fi and MQTT protocol to transmit sensor data to cloud services like AWS IoT and Firebase. Enables industrial operators to monitor system status remotely through a web-based dashboard or mobile application. Actuator Control and Alarm Triggers. Controls a buzzer for alerts in case of critical environmental conditions (e.g., gas leaks, high temperature, fire hazards). Applications of ESP32 IoT Projects (Smart Home, Industrial Automation) Wireless Communication (Wi-Fi, Bluetooth) Embedded Systems & Robotics Wearable Devices Camera-based Applications (ESP32-CAM).

4. RESULTS AND DISCUSSION

The Cloud-Based Automation for Industrial Ambient Robotic System was tested for real-time monitoring, autonomous operation, cloud communication, and remote accessibility. The system was evaluated based on sensor accuracy, motor performance, response time, cloud integration, and security features. This chapter presents the observed results, system performance analysis, and key discussions on its effectiveness in an industrial setting. The DHT11 temperature and humidity sensor provided readings with an accuracy of approximately ±1.8°C for temperature and ±4.5% RH for humidity, which was within acceptable limits for industrial monitoring. The MQ135 gas sensor detected variations in air quality but required periodic calibration to maintain accuracy, with observed deviations of around ±6 PPM. The flame sensor demonstrated an immediate response to fire hazards, with a detection time of less than one second. The ultrasonic sensor accurately measured distances with a minor deviation of ±3.2 mm, ensuring precise obstacle detection and avoidance. The L298N motor driver efficiently controlled the DC motors, allowing smooth forward, backward, and directional movement of the robotic system. The response time for motor activation was observed to be less than one second, with a movement precision of around 95 percent. The servo motor provided accurate angular adjustments, responding within 0.5 seconds and maintaining an accuracy of 98 percent in positioning tasks. The combination of DC and servo motors enabled reliable robotic navigation and adaptive movement based on environmental conditions. The system successfully automated real-time environmental monitoring and robotic movement, enhancing industrial efficiency. The sensors provided accurate and reliable readings, ensuring precise monitoring of temperature, gas levels, fire hazards, and obstacles. The motor control system enabled smooth and responsive robotic navigation, adapting effectively to changing environmental conditions. The cloud-based communication allowed remote monitoring and control with minimal latency, making the system highly suitable for industrial IoT applications. Security measures such as SSL/TLS encryption and authentication ensured safe data transmission and system access One of the primary challenges was the periodic calibration required for the MQ135 gas sensor to maintain accurate readings. Network dependency posed another limitation, as cloud-based control required a stable internet connection to function effectively. Battery consumption was also a concern, as continuous operation led to faster power drainage, necessitating an optimized power management system for extended usage.

5. ADVANTAGES AND APPLICATIONS

Advantages:

1. Real-Time Remote Monitoring and Control: The cloud-based system enables operators to monitor and control the robotic system remotely in real-time from anywhere with internet access.

2. Automated Hazard Detection and Alert System: The system automatically detects hazards such as gas leaks, fire, or temperature anomalies using integrated sensors like gas, fire, and temperature sensors.

3. Faster Response to Emergencies: The system can trigger immediate actions, such as activating alarms, shutting down dangerous processes, or notifying emergency responders when a hazard is detected.

4. Centralized System for Multiple Hazard Types: The cloud platform serves as a centralized hub for monitoring and managing multiple types of hazards, such as fire, gas leaks, and temperature fluctuations, all in one integrated system.

5. Enhanced Safety and Operational Efficiency: The cloud-based system improves overall safety by continuously monitoring the environment and automating responses to hazards, reducing human intervention and minimizing the risk of accidents.

6. Remote Access via Cloud-Based Platform (e.g., Adafruit): Using a cloud platform like Adafruit, authorized users can access the system from anywhere, anytime, allowing for full remote management and monitoring.

7. Immediate Alerts via GSM to Authorized Personnel: The system can send immediate SMS alerts through a GSM module to authorized personnel when a hazardous situation is detected, ensuring quick response times.

8. Reduced Costs: Cloud-based infrastructure reduces the need for on-premises infrastructure and maintenance, lowering costs.

9. Improved Safety: Cloud-based system enables remote monitoring and control, reducing the risk of accidents and improving worker safety.

10. Faster Deployment: Cloud-based system enables faster deployment of robotic systems, reducing the time and effort required for setup and configuration.

Applications:

1. Industrial Safety and Hazard Monitoring The system continuously monitors industrial environments to detect hazards such as gas leaks, fires, and temperature fluctuations using IoT sensors.

2. Automated Fire Detection and Response Systems Automated fire detection systems equipped with smoke, heat, and flame sensors can identify fire hazards in real time.

3. Gas Leak Detection in Chemical Plants and Factories The system uses specialized gas sensors to detect leaks in environments such as chemical plants, refineries, or factories where hazardous gases are used or produced.

4. Remote Environmental Monitoring in Hazardous Areas In hazardous environments, such as chemical storage areas, toxic waste zones, or offshore oil rigs, IoT sensors monitor air quality, temperature, humidity, and gas levels.

5. Real-Time Temperature Regulation in Industrial Processes Temperature sensors are used to continuously monitor and regulate temperatures in industrial processes.

6. Automated Emergency Alert Systems for Industrial Sites: The system sends automated emergency alerts via cloud-based platforms (e.g., SMS, email, or app notifications) when dangerous conditions are detected, such as gas leaks, fires, or extreme temperature changes.

6. CONCLUSION

The Cloud Based Automation for Industrial Ambient Robotic System was developed to enhance real-time monitoring, autonomous operations, and cloud-based control in industrial environments. The system successfully integrated IoT sensors, edge computing, robotic movement, and cloud communication, allowing efficient automation and remote monitoring. The system was successfully implemented and tested, demonstrating high accuracy, efficiency, and reliability. The sensors provided real-time environmental data, and the motors enabled smooth robotic navigation. The integration of MQTT-based cloud communication allowed seamless data transmission and remote control. Some key accomplishments of the project include: Accurate real-time monitoring of temperature, humidity, gas concentration, fire hazards, and obstacles; Autonomous robotic movement using DC motors, servo motors, and obstacle detection sensors; Edge computing for fast decision-making, reducing dependency on cloud-based processing; Cloud integration through AWS IoT/Firebase, ensuring remote access and monitoring; Secure communication with SSL/TLS encryption, preventing unauthorized access; The results of system testing confirmed that sensor accuracy, motor response time, and cloud communication latency were within acceptable limits. The robotic system effectively navigated industrial environments, making autonomous adjustments based on real-time sensor feedback. The cloud dashboard provided a user-friendly interface for remote monitoring and manual control, proving its effectiveness in industrial IoT applications.

REFERENCES

[1]. Bose, M. (2020). Industrial Automation and Robotics: Principles and Applications.

[2]. M. Senthil Kumar and M. Gopinath, "An Efficient Polynomial Pool-Based Scheme for Distributed Heterogeneous WSNs", *International Journal of Modern Engineering Research (IJMER)*. (Vol.3, Issue 6, Nov-Dec.2013, PP 3328-3335, ISSN: 2249-6645).

[3]. Senthilkumar Meyyappan and N. Selvamuthukumaran, "Network Selection in Heterogeneous Wireless Systems using GRA Method", *Journal on Electronic and Automation Engineering*, Vol. 4(1), March 2025, pp. 127-132.

[4]. Senthilkumar Meyyappan, A. Bharath Naik, A. Uma Sai and Ch. Keerthi, "Improving Weather Forecasting Accuracy Using Machine Learning", *Journal on Electronic and Automation Engineering,* Vol. 2(4), December 2023, pp. 9-18.

[5]. Gupta, D., & Nair, S. (2019). IoT and Cloud Computing for Smart Industries.

[6]. M. Senthil Kumar, R. Karthik and I. Rabeek Raja, "An Efficient Approach for Increasing Power Optimization in Mobile Ad-Hoc Networks", *International Journal of Engineering Research and Technology (IJERT)*. (Vol.3, Issue 2, February 2014).

[7]. M. Senthil Kumar and L. Praveen, "An Assuring Approach for Tree-Based Routing Topology in WSNs", *International Journal of Emerging Trends in Engineering and Development (IJETED).* (Issue 3, Vol.6, November 2013, ISSN: 2249 – 6149).

[8]. M. Kavitha, T. Maheshwaran and M. Senthil Kumar, "Secure Routing in MANETs with Key Management", *International Journal on Engineering Technology and Sciences (IJETS).* (Vol.1, Issue 6, October 2014, ISSN (P): 2349 – 3968, ISSN (O): 2349 - 3976).

[9]. Kumar, R. (2021). Edge Computing and IoT: A Practical Approach to Industry 4.0.

[10]. M. Senthil Kumar, "Energy Efficient Techniques for Transmission of Data in Wireless Sensor Networks", *Journal of Computing Technologies (JCT)*. (Vol.5, Issue 2, February 2016, ISSN: 2278 – 3814).

[11]. C.I. Vimalarani and M. Senthil Kumar, "Energy Efficient PCP Protocol for k-Coverage in Sensor Networks", *IEEE International Conference on Computational Intelligence and Computing Research, IEEE Proceedings, 2010.*

[12]. M. Subashinidevi, S. Castro and M. Senthil Kumar, "Efficient Anonymous Transfer of Data in Wireless Networks", *International Journal on Engineering Technology and Sciences (IJETS)*. (Vol.1, Issue 6, October 2014, ISSN (P): 2349 – 3968, ISSN (O): 2349 - 3976).

[13]. Patel, A., & Desai, N. (2018). Embedded Systems and Automation: An Industry Perspective.

[14]. M. Senthil Kumar and C. Sridhathan, "Impact of Mobility on the Routine of Enhanced – DSDV Protocol in Mobile Ad-hoc Networks", *International Journal of Applied Engineering Research (IJAER)*. (Vol.13, No.14, 2018, PP 11674-11679, ISSN: 0973-4562).

[15]. M. Senthil Kumar and Ashish Chaturvedi, "Energy-Efficient Coverage and Prolongs for Network Lifetime of WSN using MCP*", European Journal of Scientific Research (EJSR).* (Vol.95, No.2, January 2013, ISSN: 1450 – 216X / 1450 – 202X).

[16]. Senthilkumar Meyyappan, G. Lava Kumar, G. Niharika and G. Chakradhar, "Cellular Network Signal Strength Analyser", *Journal on Electronic and Automation Engineering*, Vol. 4(1), March 2025, pp. 165-174.

[17]. Sharma, V. (2020). Artificial Intelligence in Industrial Robotics: Machine Learning and Automation.

[18]. M. Senthil Kumar and Ashish Chaturvedi, "A Novel Enhanced Coverage Optimization Algorithm for Effectively Solving Energy Optimization Problem in WSN", *Research Journal of Applied Sciences, Engineering and Technology (RJASET).* (Issue 4, Vol.7, January 2014, ISSN: 2040 – 7459 & e-ISSN: 2040 – 7467).

[19]. Senthilkumar Meyyappan, K. Susmitha, K. Vaishnavi and M. Sai Rao, "Condition Based Monitoring and Maintenance System for Underground Metro Stations", *Journal on Electronic and Automation Engineering*, Vol. 4(1), March 2025, pp. 175-182.

[20]. M. Kavitha, T. Maheshwaran and M. Senthil Kumar, "Ensure Data Transmission in Mobile Ad-Hoc Networks", *International Journal on Engineering Technology and Sciences (IJETS)*. (Vol.2, Issue 4, April 2015, ISSN (P): 2349 – 3968, ISSN (O): 2349 - 3976).

[21]. Singh, P. (2021). Sensor-Based Smart Systems: IoT Applications in IndustrialAutomation.

[22]. Verma, K. (2022). Cloud Computing and Cybersecurity for Industrial IoT.

[23]. Williams, J. (2017). Autonomous Robotics: Motion Planning and Control Techniques.