Wireless Air Quality Monitoring System

Vimal Kumar D, Nikneshwaran A, Nikash T, Abishek H

Information Technology, Hindusthan Institute of Technology

DOI: https://doi.org/10.51583/IJLTEMAS.2026.150400022

Received: 30 March 2026; 04 April 2026; Published: 02 May 2026

ABSTRACT

This The Wireless Air Quality Monitoring System is designed to monitor environmental conditions in real time using IoT technology. The system detects harmful gases present in the air using gas sensors. It also measures temperature and humidity using appropriate sensors. An ESP8266 microcontroller is used to process the collected data efficiently. The system continuously compares gas levels with predefined safe limits. When the gas concentration exceeds the threshold, a buzzer is activated to alert users immediately. The processed data is transmitted wirelessly using Wi-Fi technology. Users can monitor real-time data through the Blynk IoT mobile application. The system is cost-effective, easy to install, and suitable for both indoor and outdoor environments. Overall, this project provides a reliable solution for improving safety and environmental monitoring.

Keywords: Air Quality Monitoring, Internet of Things (IoT), ESP8266 Microcontroller, Real-Time Monitoring, Wireless Communication, Blynk IoT Application, Environmental Monitoring

INTRODUCTION

Air pollution is one of the most serious environmental problems affecting human health and ecosystems worldwide. The rapid growth of industries, vehicles, and urbanization has significantly increased the level of harmful gases in the atmosphere. Continuous monitoring of air quality has become essential to ensure a safe and healthy environment. Traditional air monitoring systems are often expensive and require manual supervision. Therefore, there is a need for a low-cost and automated solution.This project presents a Wireless Air Quality Monitoring System using Internet of Things (IoT) technology. The system is designed to detect harmful gases and monitor environmental parameters such as temperature and humidity. Gas sensors are used to measure the concentration of pollutants in the air.

A temperature and humidity sensor helps in analyzing atmospheric conditions.The collected data is processed using an ESP8266 microcontroller, which has built-in Wi-Fi capability. This enables real-time data transmission without the need for additional modules. The system continuously monitors air quality and compares the values with predefined threshold limits.When the gas concentration exceeds the safe level, a buzzer is activated to alert users immediately.

The system also allows remote monitoring through the Blynk IoT mobile application. Users can view real-time data from anywhere at any time.The proposed system is simple, cost-effective, and easy to implement. It reduces the need for manual monitoring and increases efficiency. This project can be used in homes, industries, and public places to improve safety. Overall, it provides a reliable solution for real-time environmental monitoring and contributes to creating a safer and smarter environment. The system is designed with a modular approach, allowing easy integration of additional sensors in the future. It supports scalability for larger monitoring applications such as smart

The Overhead AC Line Fault Detection and Monitoring System is developed to address these challenges. The system continuously monitors the voltage status of four AC poles, detects voltage absence or abnormal fluctuations, monitors battery temperature, identifies fire incidents, and automatically controls a cooling mechanism. In case of any abnormal condition, the system sends real-time alerts to the operator through IoT, ensuring quick response and minimizing risks.

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue IV, April 2026

LITERATURE REVIEW

The literature survey focuses on existing air quality monitoring systems developed using IoT technology. Many researchers have proposed systems using gas sensors like MQ-2 and MQ-135 for detecting harmful gases. Studies show that IoT-based systems provide real-time monitoring and improved efficiency compared to traditional methods. Several research works have used microcontrollers such as ESP8266 and Arduino for data processing and transmission.

Author (Year)	Technology/ Method	Key Contribution
Nath (2025)	IOT	Developed a real-time air quality monitoring system using ESP8266 with wireless data transmission.
Lopez(2025)	Sensor	Proposed a gas and smoke detection system using MQ2 sensor with alert mechanism.
Alim (2025)	NodeMCU	Designed a smart pollution detection system integrated with IoT communication platforms.
Hassam (2025)	Wireless	Created an indoor and outdoor air monitoring system with continuous real-time updates.
Christakis (2024)	WSN	Studied sensor accuracy and aging effects in wireless air pollution monitoring networks.

Proposed System

The proposed system is a Wireless Air Quality Monitoring System designed to measure environmental parameters in real time. It uses gas sensors along with temperature and humidity sensors to collect data from the surroundings. An ESP8266 microcontroller processes the data and transmits it using Wi-Fi technology. The system compares gas levels with predefined thresholds and activates a buzzer when unsafe conditions are detected. Users can monitor the data remotely through the Blynk IoT mobile application for improved safety and convenience..

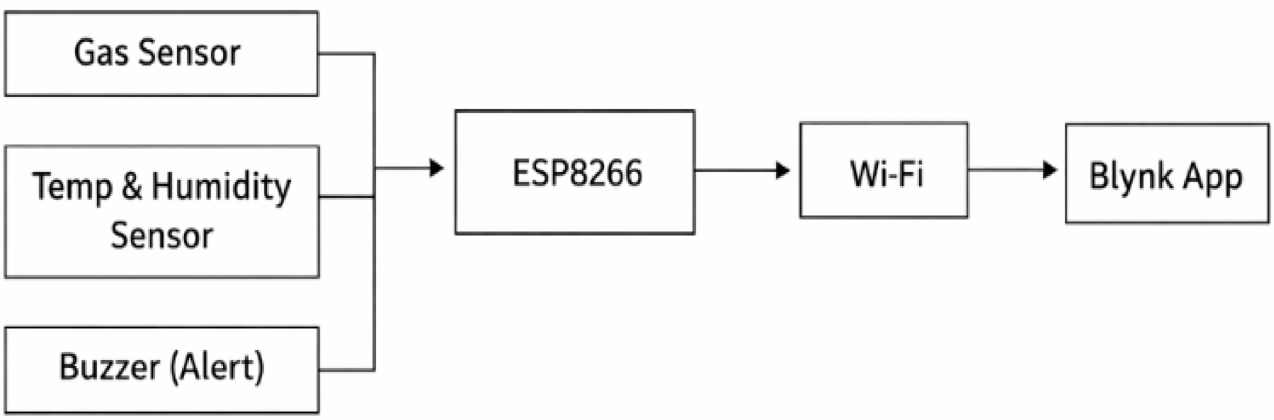


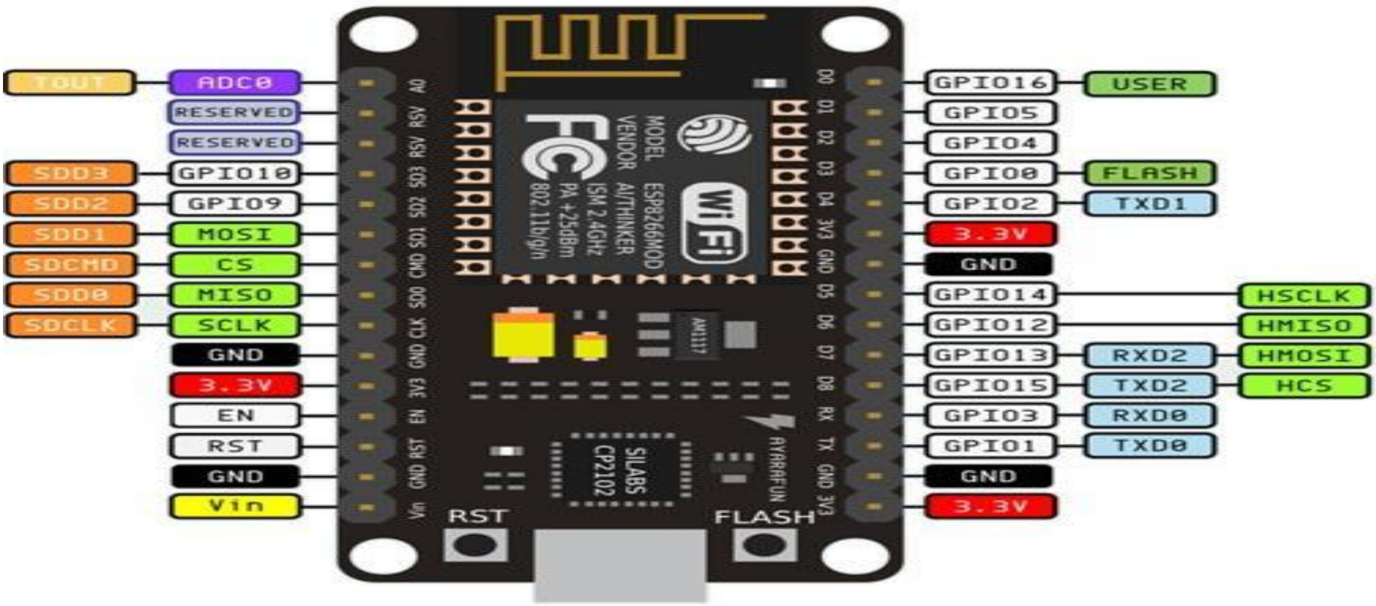
Fig. 1: Proposed System Block Diagram

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue IV, April 2026

System Architecture

The system architecture consists of three main layers: sensing, processing, and application layer. The sensing layer includes gas, temperature, and humidity sensors that collect real-time environmental data. The processing layer is handled by the ESP8266 microcontroller, which processes and analyzes the sensor data. The application layer displays the data through the Blynk IoT mobile application for user access. Communication between all layers is achieved using Wi-Fi for efficient and real-time data transmission.

Fig. 2: ESP8266



Micro Controller

The ESP8266 microcontroller is the main controlling unit of the system. It is responsible for reading sensor data, processing it, and sending it to the mobile application. One of its key features is the built-in Wi-Fi capability, which eliminates the need for external communication modules. It supports multiple input and output pins for connecting sensors and actuators. The ESP8266 is widely used in IoT applications due to its low cost and efficient performance. It enables real-time data transmission and remote monitoring. Overall, it acts as the brain of the entire system.

Fig. 3: Gas Sensor

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue IV, April 2026

The gas sensor is used to detect harmful gases present in the environment. It works by sensing gas concentration and converting it into an electrical signal. MQ series sensors such as MQ-2 or MQ-135 are commonly used in air quality monitoring systems. These sensors can detect gases like smoke, carbon monoxide, and other pollutants. The output from the sensor is given to the microcontroller for processing. Proper calibration is required to ensure accurate readings. This component plays a key role in detecting unsafe conditions.

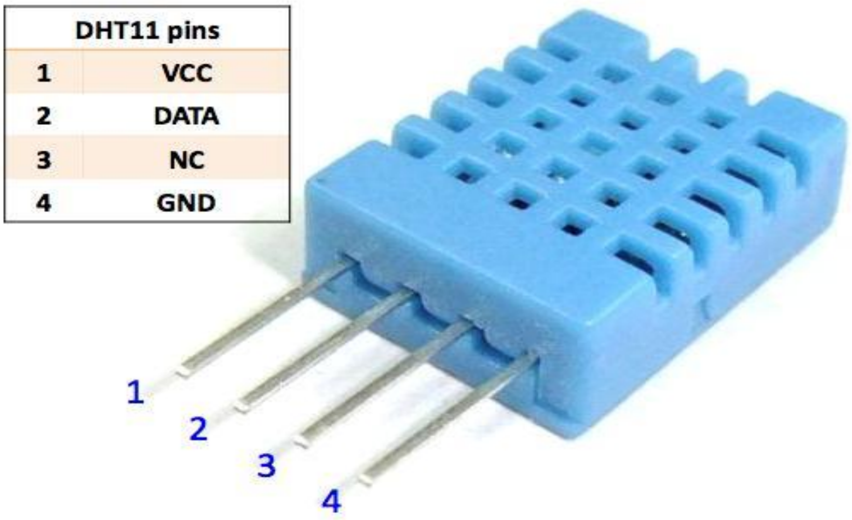


Fig.4.Temperature and Humidity Sensor

The temperature and humidity sensor is used to measure environmental conditions. It provides digital output, which makes it easy to interface with the microcontroller. The DHT11 sensor is cost-effective and suitable for basic applications, while DHT22 provides higher accuracy. This sensor helps in monitoring temperature and moisture levels in the air. It enhances the system by providing additional environmental data. The readings are used along with gas data for better analysis. It ensures complete air quality monitoring.

Fig .5. Buzzer

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue IV, April 2026

The buzzer is used as an alert system in the project. It produces a sound when the gas level exceeds a predefined threshold. This helps in providing immediate warning to users in case of dangerous conditions. The buzzer is simple to use and consumes very low power. It is controlled directly by the microcontroller through programming. The alert system plays a crucial role in ensuring safety. It allows users to take quick action when needed.

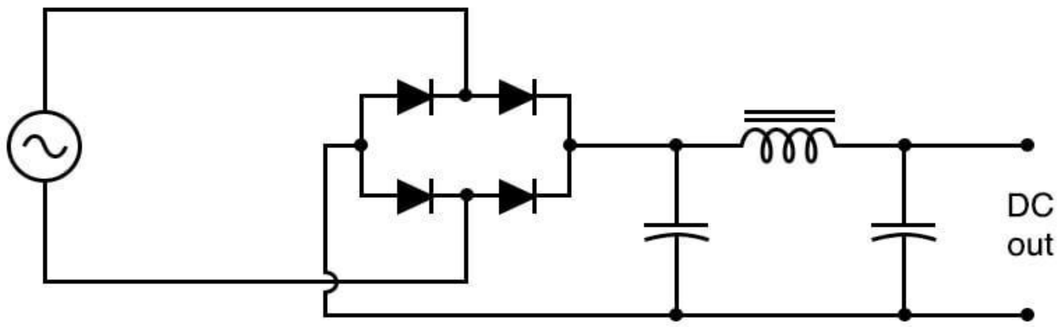


Fig. 6: Power Supply Unit .

The power supply unit is responsible for providing the required voltage to the system. It converts AC voltage into regulated DC voltage using a transformer, rectifier, and filter. The transformer reduces the voltage to a suitable level. The rectifier converts AC into DC, and capacitors remove noise and fluctuations. A voltage regulator ensures a constant output voltage. A stable power supply is essential for proper functioning of the system. It prevents damage to electronic components.

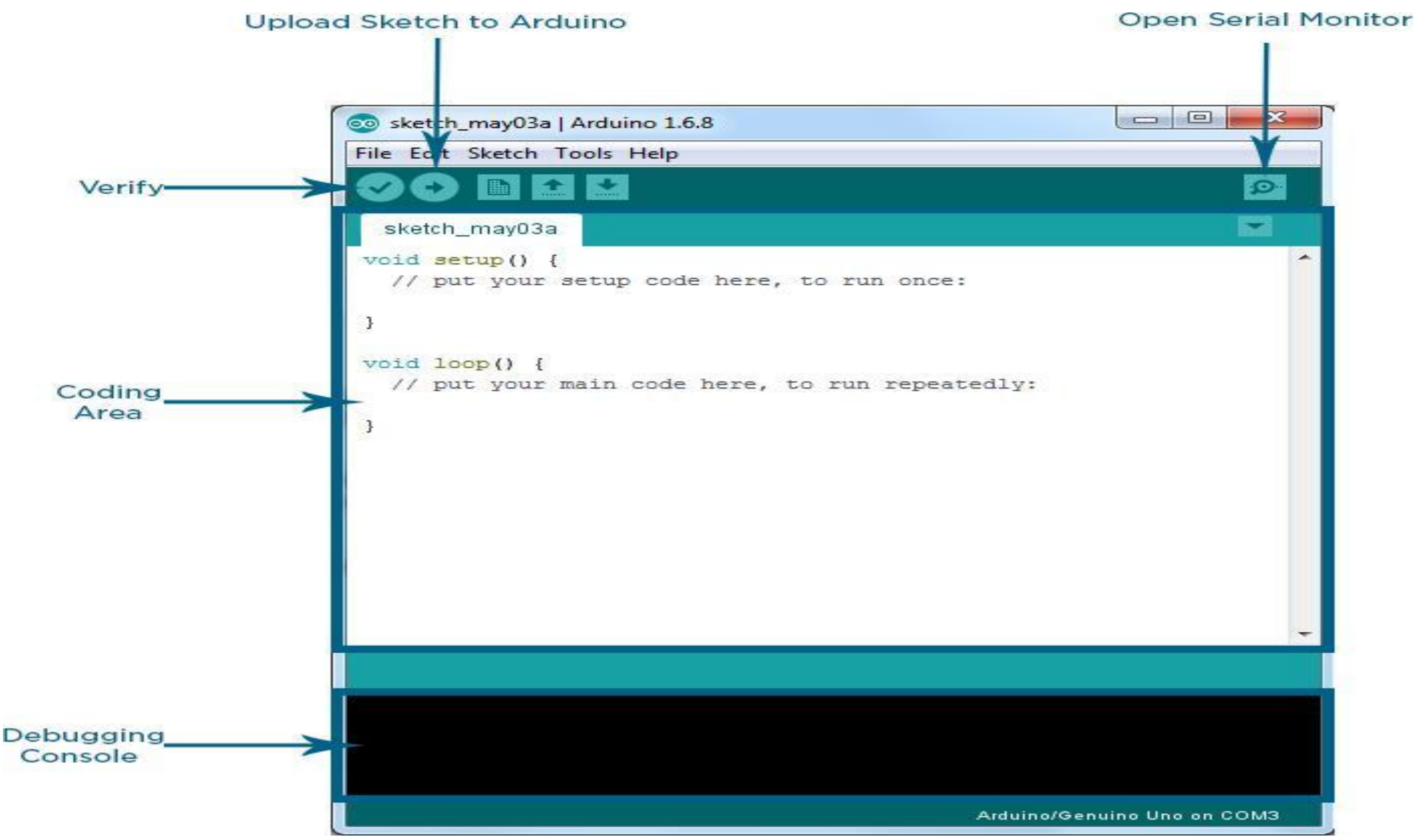


Fig. 7: Arduino IDE

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue IV, April 2026

The hardware setup begins with selecting all required components such as ESP8266, gas sensor, DHT sensor, buzzer, and power supply. Each component is checked for proper working before assembly. The ESP8266 microcontroller is placed as the central unit of the system. The gas sensor is connected to the appropriate input pin for detecting air pollutants. The temperature and humidity sensor is interfaced using a digital pin. The buzzer is connected to an output pin for alert generation. Proper wiring is done using connecting wires and resistors where necessary. A stable power supply is provided to ensure smooth operation of all components. Connections are verified carefully to avoid short circuits or loose contacts. Finally, the complete hardware setup is tested to ensure correct functionality before programming.

Implementation

The implementation begins with assembling all hardware components such as sensors, ESP8266, and buzzer. The circuit is designed and connections are made according to the system requirements. The microcontroller is programmed using Arduino IDE to read and process sensor data. Wi-Fi configuration is done to enable data transmission to the Blynk IoT application. Finally, the system is tested to ensure proper functioning and accurate real-time monitoring..

Setting Up the Hardware

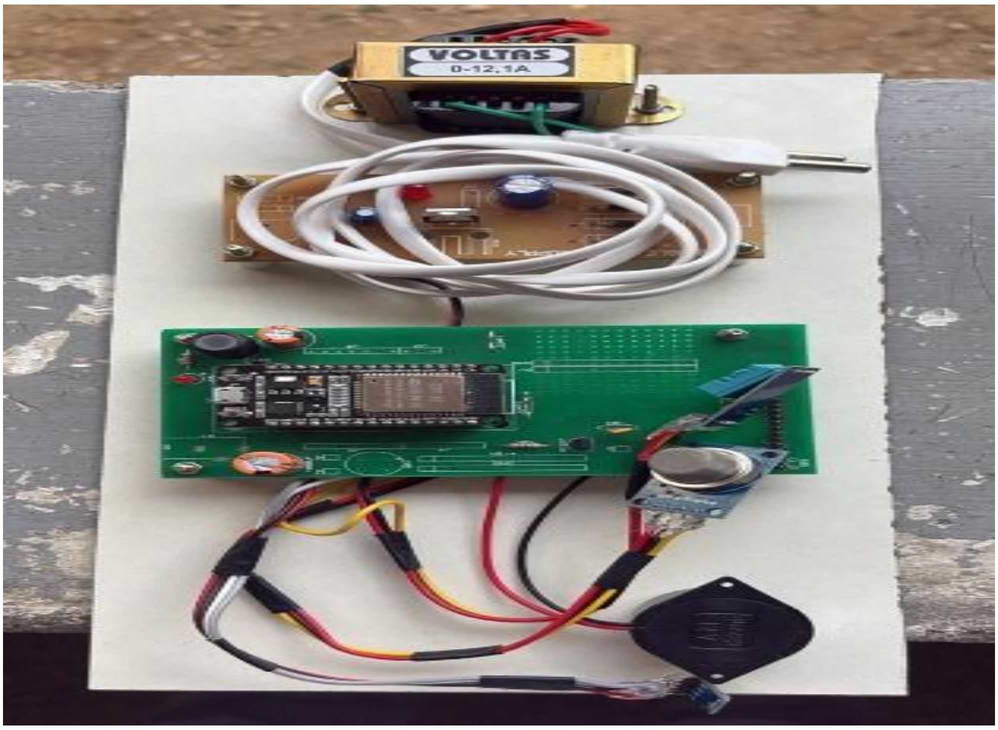


Fig. 1: Complete Hardware Setup

Programming the Arduino

Programming is done using the Arduino IDE to control the entire system. The required libraries for ESP8266, sensors, and Wi-Fi are included in the code. The program is written using Embedded C/C++ language. Sensor pins are defined and initialized in the setup function. The Wi-Fi credentials are configured to enable internet connectivity. The microcontroller is programmed to read data from gas and DHT sensors continuously. Conditional statements are used to compare gas levels with predefined thresholds. If the gas level exceeds the limit, the buzzer is activated through the program. Data is sent to the Blynk IoT application using virtual pins for remote monitoring. Finally, the code is compiled, uploaded to the ESP8266, and tested for proper execution. The loop function runs continuously to ensure real-time data monitoring. Sensor values are updated at regular intervals using delay or timer functions. Serial communication is used to display data for debugging purposes. Error handling conditions are added to manage sensor failures or incorrect readings.

INTERNATIONAL JOURNAL OF LATEST TECHNOLOGY IN ENGINEERING, MANAGEMENT & APPLIED SCIENCE (IJLTEMAS)

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue IV, April 2026

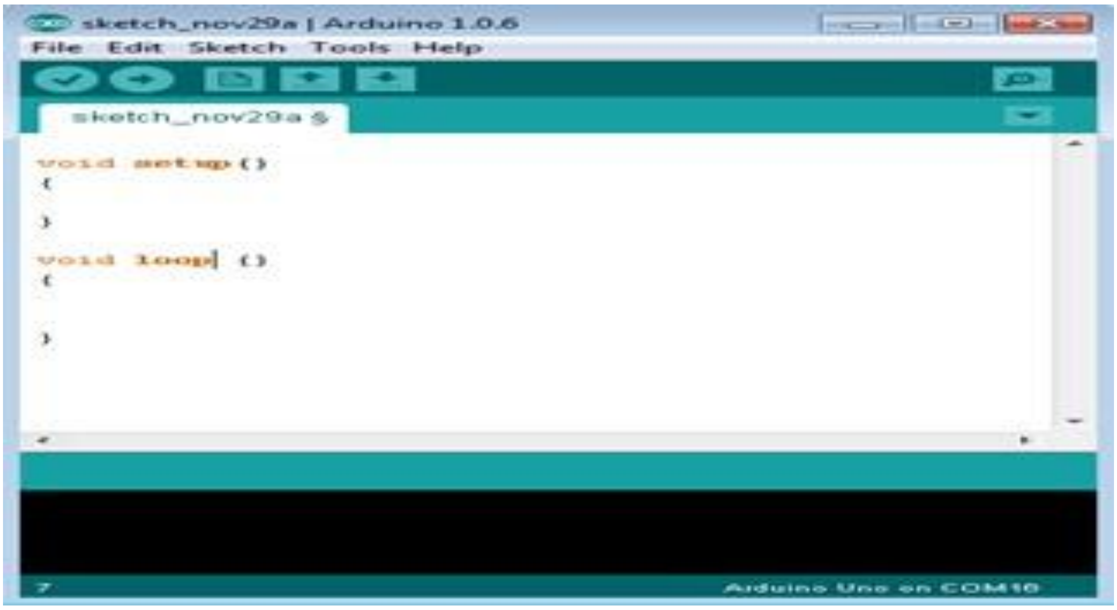


Fig. 2: Arduino IDE Interface

Arduino IDE is used to write and upload the program to the ESP8266 microcontroller. It supports programming in Embedded C and C++ languages. The IDE provides various libraries for easy interfacing of sensors and modules. It also includes tools for compiling and debugging the code. The serial monitor helps in displaying output values for testing. It is simple and user-friendly software. Arduino IDE plays a major role in system development.

Connecting ESP8266 To Thing Speak

The ESP8266 module connects to a wireless network and transmits the data to the API server of the ThingSpeak application. The smart traffic control system sends HTTP GET requests to the API server of the application to transmit the data such as signal counts, emergency vehicle counts, and traffic density levels. The data is transmitted every 30 seconds to enable near real-time monitoring of the traffic control system.

The ESP8266 WiFi module communicates with the local wireless network by using AT commands and makes a TCP connection to the ThingSpeak API server. Data transmission occurs by making HTTP GET requests with the write API key and data values at 30-second intervals.

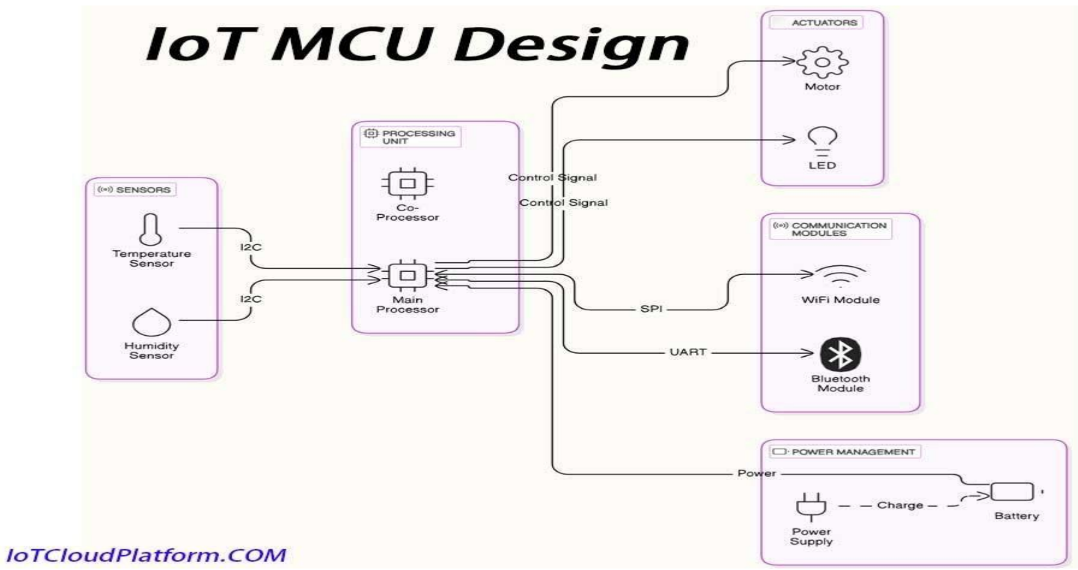


Fig.3. IOT MCU Design

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue IV, April 2026

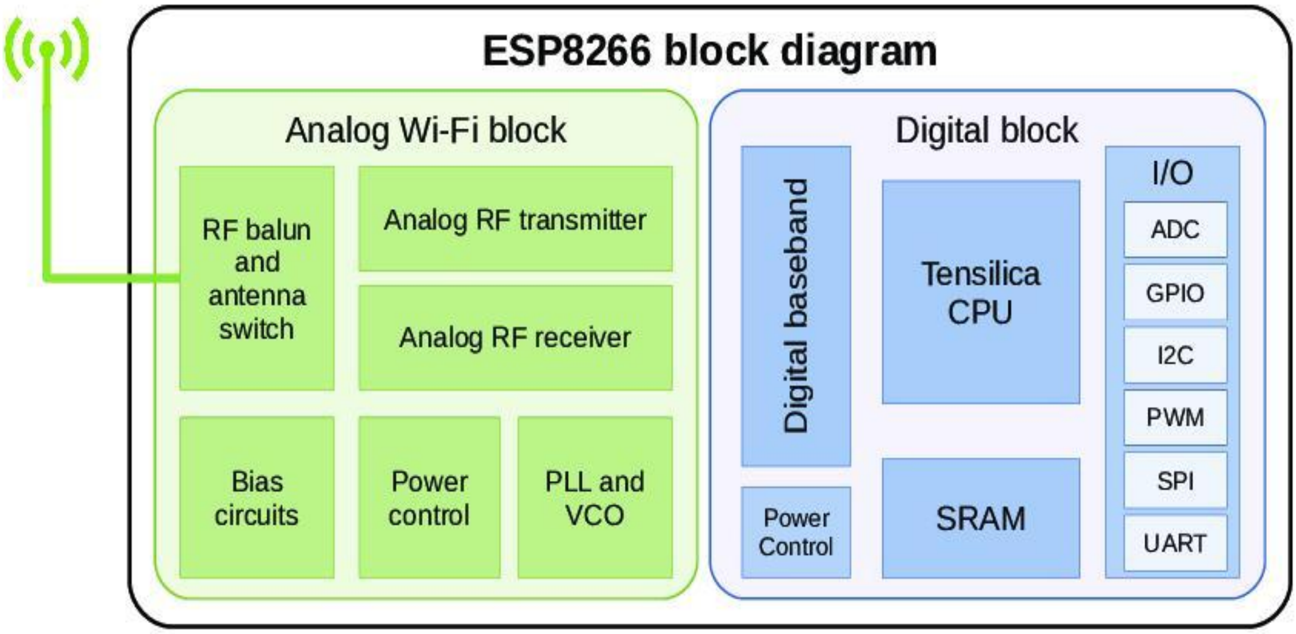


Fig.4. ESP8266 Block Diagram

The MCU design is based on the ESP8266 microcontroller, which acts as the central processing unit of the system. It interfaces with sensors to collect environmental data such as gas levels, temperature, and humidity. The microcontroller processes the input data and performs necessary computations.

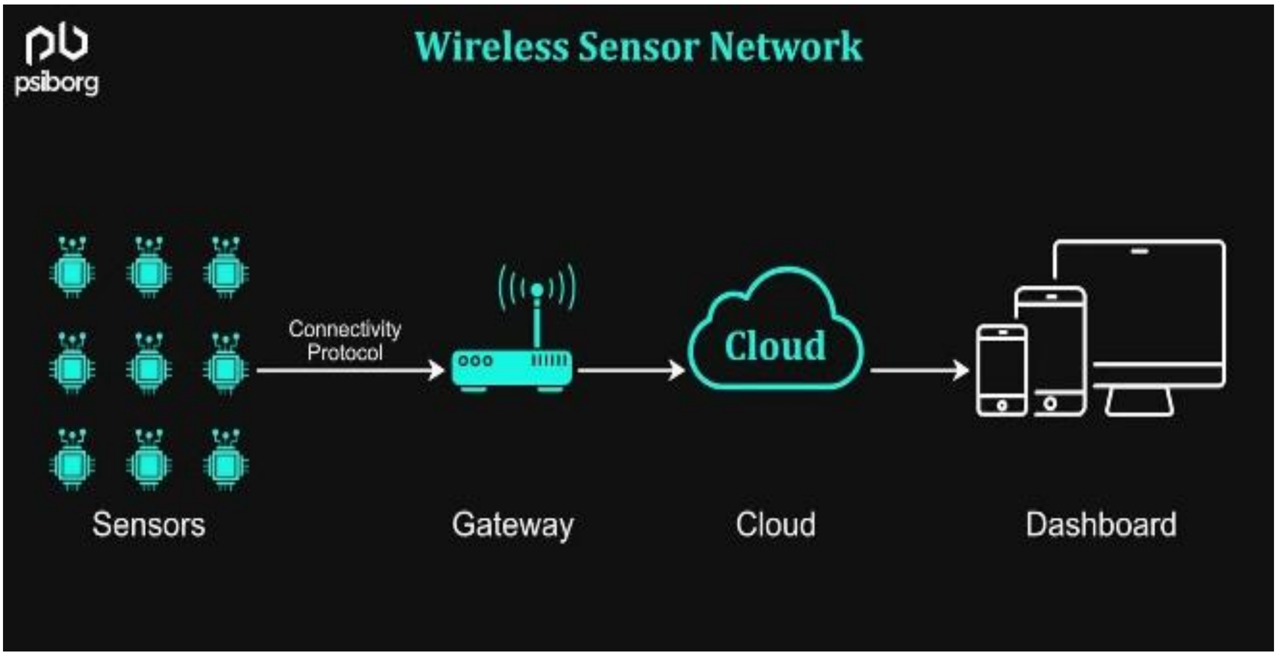


Fig.5. ESP8266 Block Diagram

Overall System Performance

The overall system performs efficiently in monitoring air quality in real time. It provides accurate detection of gas concentration under normal conditions. The response time of the system is quick when gas levels exceed the threshold. Data transmission through Wi-Fi is stable and ensures continuous remote monitoring. The system operates continuously without major interruptions. Power consumption is low, making it energy-efficient for long-term use. The alert system responds effectively to hazardous conditions. Sensor readings are consistent when properly calibrated. The system is reliable for both indoor and outdoor environments. Overall, the performance of the system is stable, cost-effective, and suitable for practical applications.

CONCLUSION

The The wireless air quality monitoring system developed in this project provides an effective solution for real-time environmental monitoring. The system successfully detects harmful gases and measures temperature and humidity using appropriate sensors. The ESP8266 microcontroller plays a key role in processing the collected data

and enabling wireless communication. The integration of IoT technology allows users to monitor air quality remotely through the Blynk IoT application.

The system is designed to provide continuous monitoring without interruption. It ensures quick response by activating a buzzer when gas levels exceed safe limits. This feature helps in preventing hazardous situations and ensures user safety. The project is cost-effective and uses easily available components. It is simple to design, implement, and maintain. The system reduces the need for manual monitoring and increases efficiency. It provides accurate and reliable data under normal conditions.

Future Scope

The The system can be enhanced by integrating additional sensors to detect more air pollutants like PM2.5 and carbon monoxide. Cloud storage can be added for long-term data logging and analysis. Advanced data analytics can be implemented to study pollution trends. The system can be integrated with mobile applications having more user-friendly interfaces. AI and machine learning techniques can be used for predictive analysis of air quality. Automated ventilation systems can be connected to control air quality in real time. The device can be made more compact and portable for easy usage. Solar power can be used to improve energy efficiency in remote areas. The system can be expanded for smart city applications with multiple monitoring nodes. Overall, future improvements can make the system more intelligent, scalable, and efficient. The system can be enhanced by integrating real-time voice assistant support for user interaction.

Blockchain technology can be used to secure environmental data and ensure data integrity. The device can be equipped with automatic firmware updates over-the-air (OTA) for easy maintenance. Integration with government pollution control systems can help in large-scale monitoring. Advanced visualization tools like heat maps can be developed for better data understanding. The system can support multilingual interfaces for wider user accessibility. Edge computing can be implemented to process data locally and reduce latency. The device can be made waterproof and rugged for harsh environmental conditions. Integration with drones can enable air quality monitoring in remote or inaccessible areas. The system can include self-diagnostic features to detect internal faults automatically. It can be connected with emergency services for instant response during hazardous situations. The project can be extended to monitor indoor air quality in smart buildings. Integration with wearable health devices can help correlate air quality with human health data. The system can be upgraded with noise pollution monitoring features. These advancements will make the system more intelligent, secure, and adaptable for modern environmental monitoring needs.

REFERENCES

1. M. K. Nath, D. Nath, R. Saikia, N. R. Varte, and B. Gogoi, "IoT-Based Real-Time Air Quality Monitoring System Using ESP8266," International Journal of Latest Technology in Engineering Management & Applied Science, vol. 14, no. 3, pp. 151–155, 2025.
2. J. M. Lopez, N. D. Paña, J. L. Lunod, J. S. D. Hermitaño, and P. B. Despares, "IoT-Based Real-Time Smoke and Gas Leak Detection System Using ESP8266 and MQ2 Sensor," International Journal of Research and Innovation in Social Science, vol. 9, no. 6, pp. 3633–3638, 2025.
3. A. Alim, A. Lukmanulhakim, and C. W, "Development of a Smart Air Pollution Detection System Utilizing MQ2 Sensor and NodeMCU ESP8266 with Telegram Integration," Jurnal Ekonomi Manajemen Sistem Informasi, vol. 6, no. 3, pp. 1634–1640, 2025.
4. M. Hassan, S. Lee, and M. Park, "IoT-Driven Pollution Detection System for Indoor and Outdoor

ISSN 2278-2540 | DOI: 10.51583/IJLTEMAS | Volume XV, Issue IV, April 2026

Environments," Computers, Materials & Continua, vol. 86, no. 2, pp. 647–723, 2026.
5. I. Christakis, O. Tsakiridis, D. Kandris, and I. Stavrakas, "Air Pollution Monitoring via Wireless Sensor Networks: Aging Behavior of Electrochemical Gaseous Pollutant Sensors," Electronics, vol. 12, no. 8, Art. no. 1842, 2023.
6. O. Alsamrai et al., "Real-Time Intelligent Monitoring of Outdoor Air Quality Using IoT and Gas Sensors," Applied Sciences, vol. 15, no. 16, Art. no. 9088, 2025.
7. M. G. Arkhan and Z. R. Saputra, "Air Quality Monitoring System Based on Internet of Things," Journal of IoT Systems, vol. 10, no. 2, pp. 45–52, 2024.
8. M. A. K. and A. Taryanto, "Sistem Monitoring Polusi Udara Berbasis IoT (ESP8266 Wemos D1)," Indonesian Journal of Electronics, vol. 8, no. 1, pp. 12–18, 2024.
9. Z. W. Oktavianto and A. B. Yunanda, "Monitoring Air Quality Around Users With IoT Based NodeMCU ESP8266," International Journal of Smart Technology, vol. 6, no. 1, pp. 25–30, 2022.
10. T. Johnson and K. Woodward, "Calibrating Low-Cost Environmental Sensors in Urban Environments," Environmental IoT Journal, vol. 5, no. 3, pp. 200–210, 2025.
11. P. Wiese et al., "A Multi-Modal IoT Node for Environmental Monitoring," IEEE Sensors Journal, vol. 25, no. 4, pp. 1001–1010, 2025.
12. A. M. Simamora, A. Denih, and M. I. Suriansyah, "Indoor Air Quality Detection Robot Model," International Journal of Robotics Research, vol. 9, no. 2, pp. 88–95, 2025.
13. K. Aurangzeb et al., "IoT-Based Smart Air Quality System with Energy-Aware Data Transmission," International Journal of CNAPC, vol. 12, no. 1, pp. 50–60, 2026.
14. H. M. Lee and S. Park, "Wireless Sensor-Based Air Pollution Monitoring System," Journal of Environmental Monitoring, vol. 11, no. 4, pp. 300–310, 2023.
15. S. Bairi, S. Swain, and A. Bandyopadhyay, "Intelligent IoT-Based Monitoring Systems for Smart Environments," Egyptian Informatics Journal, vol. 30, Art. no. 100700, 2025.
16. R. Kinnera and P. Kumar, "IoT Based Air Quality Monitoring System Using MQ135 and MQ7 with Machine Learning," International Journal of Advanced Research, vol. 7, no. 6, pp. 120–128, 2019.
17. S. Sarkar, "Real-Time CO2 Monitoring Using IoT and ESP8266," International Journal of Embedded Systems, vol. 13, no. 2, pp. 75–82, 2023.
18. D. Bandara, "IoT-Based Air Quality Monitoring with Machine Learning Integration," IEEE Access, vol. 9, pp. 140000–140010, 2021.
19. A. Karar, "GASDUINO: IoT-Based Wireless Air Quality Monitoring System," IEEE Internet of Things Journal, vol. 7, no. 5, pp. 4100–4108, 2020.
20. S. Desavale, "Low-Cost IoT-Based Air Quality Monitoring System Using ESP8266," International Journal of Engineering Research, vol. 15, no. 3, pp. 210–218, 2025.