ADROID Conference Series: Engineering and Technology
Volume 2, Issue 2, pp.68-74, 2026

Cloud-Connected IoT Framework for Instantaneous Air Quality and Acoustic Pollution Tracking with Intelligent Alerts

P. Rahul Das, P. Lakshmi Sruthi, M. Keerthi

Geethanjali College of Engineering and Technology, Hyderabad, Telangana, India
Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad, Telangana, India

rahuldaspathlavath@gmail.com, plsruthi@gmail.com, mkeerthi.cse@gcet.edu.in

Abstract. An economical solution, the IoT-based monitoring system, is proposed to curb rising air and noise pollution in cities. The sensors monitoring gases (MQ7 for CO, MQ2 for LPG and smoke) and a sound sensor have been employed to collect real-time data. The data was processed by an Arduino, displayed locally on an LCD, and then sent to ThingSpeak for analysis and remote access. This system has completely changed the manual monitoring systems that were slow and outdated into a fast and cloud-based system, which makes the environmental monitoring process more efficient, accessible, and responsive. Moreover, the system includes sensor calibration, real-time alerting via a buzzer, and secure data transmission. It enables people and authorities to respond in time, thereby enhancing environmental awareness and public health.

Keywords: IoT, Air Pollution, Sound Pollution, Arduino, MQ7, MQ2, Sound sensor, ThingSpeak, Environmental Monitoring

1. INTRODUCTION

Air and noise pollution have become the primary threats to the environment of the present day, with urbanization, industrialization, and increased vehicle emissions being the main factors that cause this menace to grow. Conventional methods of monitoring are slow and cumbersome but the integration of the IoT technology presents a revolutionary solution. The proposed system makes use of an Arduino microcontroller along with gas sensors (such as MQ135/MQ7) and a sound sensor for the purpose of real-time environmental data collection. It uploads the collected data to the ThingSpeak cloud for offline storage, visualization, and remote access. The system notifies instantly whenever the pollution levels exceed the safe limits and it also enables the analysis of historical data in terms of trends. It contributes to the realization of smart cities, compliance of environmental laws and regulation, and raising public health consciousness. Also, this IoT solution provides a cost-effective, scalable and green approach to tackle current pollution issues and paves way for AI/ML-based predictive diagnostics in future, creating smarter and cleaner cities and healthier communities. By delivering users with live and location based pollution data, the system helps to increase environmental responsibility and awareness of citizens.

II. LITERATURE REVIEW

A survey on recent work related to the IoT for monitoring air and noise pollution shows rapid advancements and open issues. Several studies highlight the ability of IoT to provide continuous environmental sensing, remote data accessibility, and predictive analysis as its major contributions. One of the IEEE paper "An IoT Based Air Pollution Monitoring System for Smart Cities" addresses the case of CO, smoke, PM2.5 and PM10 pollutants along with temperature and humidity of the surrounding being monitored in real time and the user can access the information through a global android application. A different research paper from ResearchGate, titled "Pollution Control Using Internet of Things (IoT)," not only mentions but also proves that the scope of such applications can be extended to sensors for UV, water pH, turbidity, and temperature, thus demonstrating the versatility of IoT among different environmental aspects. In addition, it is worth mentioning that the study has found ways to reduce IoT hardware costs by as much as 90%, thus making it feasible to deploy the technology over large urban areas.

III. EXISTING SYSTEM

Traditional pollution monitoring approaches are massive, government-regulated systems that depend on stationary sites with very advanced sensors for their operation. The stations perform the controls of pollutants in the air including carbon monoxide, nitrogen dioxide, sulfur dioxide, fine particles (up to 2.5 micrometers in diameter), and noise as a major concern for environmental research, legal compliance, and policy formation.

The information is collected, sent to the mainframe, and then analyzed in the lab which results in a lag in real-time reaction. Such institutional provisions are important not just for the enforcement of clean air standards but for the entire policy process. Moreover, some stations are equipped with acoustic sensors that detect noise from vehicles and industries, thus facilitating the formulation of plans for noise control. Although the accuracy level of these systems is very high, they are still costly, immobile, and lacking in scalability.

IV. PROPOSED SYSTEM

The system that is suggested is an IoT-based solution for pollution monitoring which can track both airs as well as sound pollution with the help of an Arduino microcontroller together with sensors such as MQ2, MQ7, and a sound sensor. It offers a monitoring way for harmful air pollutants like carbon monoxide (CO) and methane (CH_4) plus noise levels that is low-cost, scalable, and integrated. The data is shown in real-time on an LCD screen and sent to the ThingSpeak cloud platform thus enabling remote access through web or mobile applications. The system includes automated alerts in situations where the pollution levels go above the preset thresholds and keeps a record of the data for analysis of the trends. It is tiny, mobile, and adaptable to various places, among which are urban, industrial, and residential areas. Having very low power consumption, it guarantees uninterrupted running even in areas with no electricity, while security of data is assured through use of encryption protocols.

V. METHODOLOGY

a. Hardware and Software Requirements

1. Hardware Requirements

Hardware components of the system are necessary for the real-time process of collecting, processing, and transmitting environmental data. The next list shows the most important parts and their functions:

1. Arduino: The platform Arduino, a microcontroller that is open-source, is the brain of many electronical projects. It has the advantage of being programmed and working with sensors, actuators, and different devices at the same time thus resulting in the processing and controlling of data in real-time.

2. MQ2: The MQ2 gas sensor mainly recognizes the presence of carbon monoxide (CO), methane (CH_4), LPG, and smoke. This sensor proves to be crucial for safety systems as it constantly monitors air quality and alerts the people if there exist spots with poisonous gases.

3. MQ7: The MQ7 is a sensor intended for detecting carbon monoxide (CO) and is designed to indicate the quantity of CO in the air that is safe to breathe (non-toxic). The principle underlying the measurement of this sensor is the change in resistance of the sensor's heating element when exposed to CO.

4. Sound Sensor: The sound sensor works by identifying the sound level or acoustic signal present in the environment. The sound waves are converted into electrical signals by the sensor and then processed by the microcontroller, like Arduino.

5. LCD: A liquid Crystal Display (LCD) is a display screen that shows the data coming from sensors or gives the current status through messages.

6. Buzzer: A buzzer is a basic output device that can sound an alert or alarm as a control effect. Its function of being a warning system is realized by the buzzer in the environmental monitoring system where it alerts the user immediately if the pollution level goes above the safe limit.

7. WiFi Chip: A WiFi chip, like the ESP8266 or ESP32, is a microchip that enables wireless internet connectivity for microcontroller-based projects. IoT applications depend on these chips, that let devices either send their data to the cloud or talk to distant servers for real-time supervision and control.

8. Jumper Wires: Jumper wires, referred to as flexible electrical wires, make it possible to directly connect components on a breadboard or to a microcontroller via wires. These wires are the main means of prototyping and quick and easy circuit wiring without soldering to say the least.

9. Breadboard: The breadboard is a tool for making prototypes and for testing electronic circuits without having to solder them. This gives the possibility to connect components easily through the insertion of leads into the board's holes.

2. Software Requirements

For the Cloud-Connected IoT Framework to Track Air Quality and Acoustic Pollution proposed here that several essential software components will be used for smooth operation:

1. Arduino IDE: A platform-independent development environment for making, debugging, and deploying code to microcontrollers based on Arduino. It comes with OTA programming, syntax highlighting, and library management which makes sensor and module integration very easy.

2. Firmware: It is the main part of the IoT system, deciding communication protocols, gathering sensor data, and controlling the power supply. It oversees the working of sensors such as MQ7 and MQ2 while also doing error recovery and system diagnostics to maintain fault tolerance.

3. Libraries:

• SoftwareSerial.h: Allows serial communication with devices like the ESP8266 Wi-Fi module on any digital pins so that the default serial pins can be used for debugging.

• ThingSpeak.h: Allows sensor data transfer from Arduino to the ThingSpeak cloud platform for real-time storage and analysis.

4. ThingSpeak API: It is a cloud-based platform where data are stored, visualized, and analyzed. It allows multi-field data collection and provides event-based triggers to send automated alerts when pollution levels go beyond the set limits.

b. Block Diagram

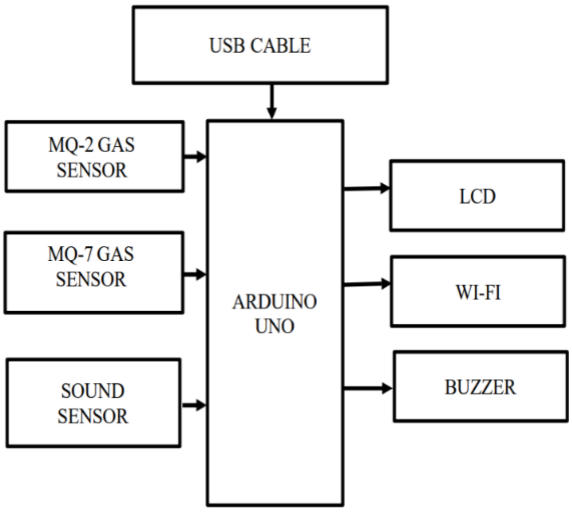

Fig. 1. Module Diagram

c. Detail Design

The suggested system for monitoring atmospheric and acoustic pollution through IoT involves the use of an Arduino Uno microcontroller that detects pollution levels in real-time. The system has incorporated various sensors such as an MQ7 sensor for carbon monoxide, an MQ2 sensor for smoke and flammable gases, and a sound sensor for measuring the flow of noise. The system first gathers the data from these sensors, processes it, and then sends it to a cloud-based platform, which is ThingSpeak, for further analysis as well as for remote monitoring. This system is perfect for the three areas urban, industrial, and residential, supplying very important information for the safety of the environment and the health of the public. Moreover, the ESP8266 Wi-Fi module is part of the hardware, which makes it possible to send data wirelessly to ThingSpeak so it can be accessed from afar. The scalability of the system is one of its major strengths because it can be used at any place with internet access, thus being a versatile solution for the pollution monitoring problem. After a period of time, the current setup can be expanded with the addition of more sensors, including PM2.5 and PM10 sensors, calculating the fine particulate matter.

TABLE 1. CIRCUIT CONNECTIONS

Component	Arduino Pin
MQ2 Sensor	A0
MQ7 Sensor	A1
Sound Sensor	A2
Buzzer D8	D8
ESP8266 (via SoftwareSerial)	RX → D9, TX → D10
LCD	RS → D2 EN → D3 D4 → D4 D5 → D5 D6 → D6

	D7 → D7

d. Algorithm

So to monitor and control the environmental pollutants, IoT based pollution monitoring system works in a manner like it operates layer by layer.

1. System Initialization: Sensors (MQ7 for carbon monoxide, MQ2 for smoke and the sound sensor for noise) attached to arduino have to be initially powered on and at the time of power on initialized, so that they are ready for data collection.
2. Connect to Wi-Fi: Now this is where we tell the ESP8266 to connect to your Wi-Fi network.
3. Sensor Data Acquisition: The Arduino reads sensors and stores raw readings in regular intervals in real time for further processing.
4. Processing and Filtering: Based on the calibration functions the raw data is transformed into pollution levels and further processing includes noise filtering to reduce environmental variations.
5. Decision Making and Threshold Checking: Processed data are compared against the preset pollution thresholds.

e. Flowchart

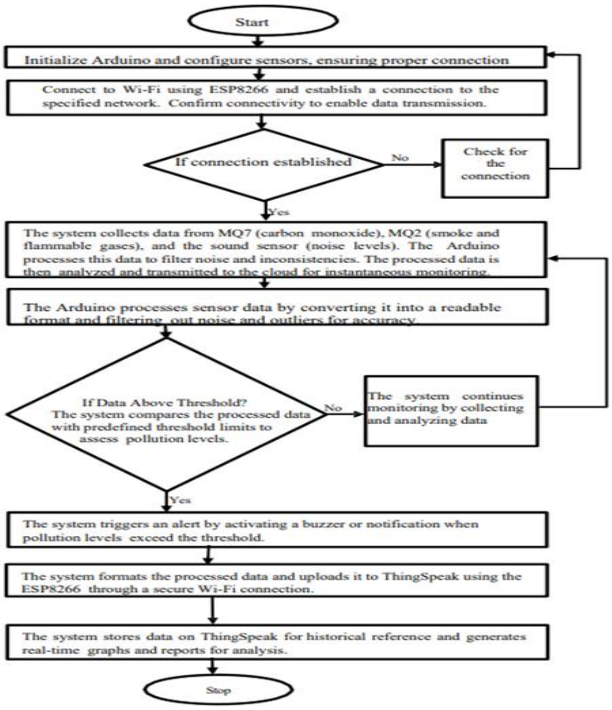

Fig. 2. Flowchart

f. Implementation

The process of IoT based Air and Sound Pollution Monitoring System from sensor configuration to real-time data transmission and collection include several vital stages. Here is a list of the major processes to kick off the system:

1. Setup the Arduino and the sensors (MQ7, MQ2 and sound sensor) and write the code for data aquisition.
2. Join the module ESP8266 to the designated Wi-Fi network, wait for connection if failed till success.
3. In the hardware system, real-time data are collected persistently from MQ7, MQ2 and sound sensor, and the raw data are stored temporarily for processing.
4. Use calibration curves to translate raw sensor output into real pollution levels under eliminating noises and environmental variations.
5. Process the data: against established pollution threshold values, inform alert mechanism (buzzer) if the data is more than threshold value, else continue monitoring in the system.
6. After process the data, format to cloud then upload to thingspeak from esp8266 module use wifinetwork for cloud realtime monitoring.

VI. DISCUSSIONS

a. Website View

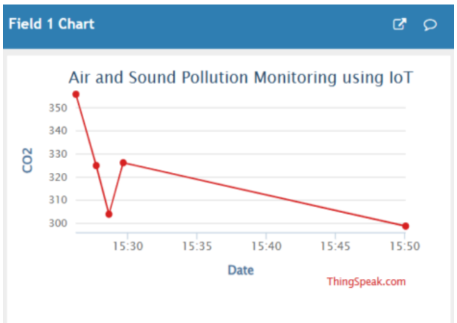

Fig. 3. CO2 Level Monitoring Chart

The depiction shows a real time CO2 level reading from the monitoring system, which at first showed the extremely high concentration of about 360 ppm, then a sudden drop followed by some mild fluctuations. This could be an indication of changes in environment, for example better ventilation or less emission. It is of great importance that the system achieving extremely accurate detection of such fluctuation, especially for application of monitoring air quality in non-circulating air environment like classrooms or manufacturing plants.

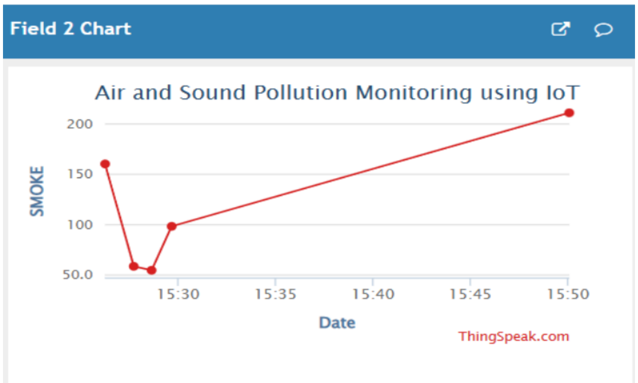

Fig. 4. Smoke Level Monitoring Chart

The concentration changes of smoke without interruption as monitored by the system based on IoT technology for real-time monitoring, is shown in the figure, with an initial very high value of about 155 units, then a steep decline, and finally, a gradual increase at the later stage. The changes thus implied can be interpreted as the variations in the environmental conditions or the sources of pollution. The rapid response and accuracy of the system are crucial for the detection of air quality changes and thus it helps in monitoring the trends, identifying health risks, and supporting prompt actions for environmental and public health management.

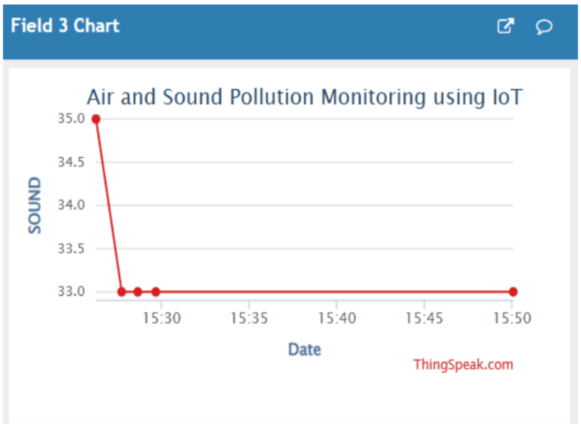

Fig. 5. Sound Level Monitoring Chart

The chart presents the sound levels that were recorded through the IoT-based system and they started at about 35 dB and with a slight dip to 33 dB, then continued to be constant—showing a quiet and stable acoustic environment. It can be inferred that there were either very few people or machines in the area. Such noise levels are very conducive for sensitive zones such as residential areas, hospitals, or schools.

a. Mobile View

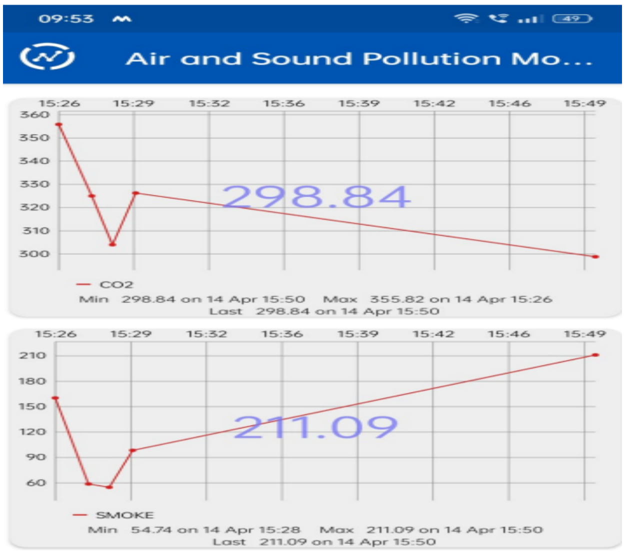

Fig. 6. Mobile view of CO2 and Smoke

Fig.7. Mobile view of Sound

The mobile view of the graphs before any UI enhancements were applied is depicted in the above images (Figure 6 and Figure 7). The mobile view is very important because it allows users to access and interact with data visualizations without any difficulties on smaller screens every day and it is stated that it is a necessity in the mobile-first digital landscape. Users are enabled to quickly and easily spot trends, patterns, and insights even during their movement, without being dependent on larger devices. These screenshots are the starting point for the evaluation of the impact of the subsequent design upgrades.

VII. CONCLUSION AND FUTURESCOPE

The IoT-based Air and Sound Pollution Monitoring System is an advanced solution for the real-time monitoring of air and noise pollution with high precision. The system uses an Arduino microcontroller along with gas sensors (MQ135, MQ7, MQ2), a noise sensor, and an LCD display for feedback and a buzzer for alarming when the pollution exceeds safe levels. The transmitted data through the ESP8266 module to ThingSpeak is visualized, stored, and analyzed. The modular design of the system facilitates easy sensor upgrades making it usable in the cities, industries, and research areas. Moreover, the possible area of expansion for the system is enormous. Future advancements could be the application of artificial intelligence in predictive analytics, a user-engaging mobile application, and automatic environmental reactions through the integration with smart city infrastructure.

Conflict of Interest

The authors declare no conflicts of interest regarding the current research.

References

[1] Arnab Kumar Saha, Sachet Sircar, and Priyasha Chatterjee, “A Raspberry Pi Controlled Cloud-Based Air and Sound Pollution Monitoring System with Temperature and Humidity Sensing,” *IEEE*, 2018, doi: 10.1109/ICCMC.2018.8487704.

[2] Harsh Gupta, Dhananjay Bhardwaj, and Himanshu Agrawal, “An IoT-Based Air Pollution Monitoring System for Smart Cities,” *IEEE*, 2019, doi: 10.1109/ICIRCA.2019.8897970.

[3] Himadri Nath Saha, Supratim Auddy, and Avimita Chatterjee, “Pollution Control Using Internet of Things,” *IEEE*, 2017, doi: 10.1109/ICACCI.2017.8126113.

[4] K. Cornelius, N. Komal Kumar, and Sagar Pradhan, “An Efficient Tracking System for Air and Sound Pollution using IoT,” *IEEE*, 2020, doi: 10.1109/ICACCS48705.2020.9074210.

[5] K.S.E. Phala, A. Kumar, and Gerhard P. Hancke, “Air Quality Monitoring System Based on ISO/IEC/IEEE 21451 Standards,” *IEEE Transactions on Industrial Electronics*, 2015.

[6] Frances Moore, “Climate Change and Air Pollution: Exploring the Synergies and Potential for Mitigation in Industrializing Countries,” *Sustainability*, vol. 1, no. 1, pp. 43–54, 2009.

[7] R.D. Brook, F.B. Cascio, W. Hong, et al., “Air Pollution and Cardiovascular Disease: A Statement for Healthcare Professionals from the Expert Panel on Population and Prevention Science of the American Heart Association,” *Circulation*, vol. 109, no. 21, pp. 2655–2671, 2004.

[8] H. Ali, J.K. Soe, and S.R. Weller, “A Real-Time Ambient Air Quality Monitoring Wireless Sensor Network for Schools in Smart Cities,” in *Proc. IEEE Int. Smart Cities Conf. (ISC2)*, Oct. 2015.

[9] K. Sarath and R. Guttikunda, “Health Impacts of Particulate Pollution in a Megacity—Delhi, India,” *Environmental Development*, vol. 6, pp. 8–20, Apr. 2013.

[10] J. Dutta et al., “AirSense: Opportunistic Crowd-Sensing Based Air Quality Monitoring System for Smart City,” in *Proc. IEEE SENSORS*, Oct.–Nov. 2016.

[11] Kwok Wai Tham et al., “A Wireless Sensor-Actuator Network for Enhancing IEQ,” in *Proc. 15th Conf. Int. Society of Indoor Air Quality & Climate (ISIAQ)*, 2018.

[12] W. Wang, Y. Yuan, and Z. Ling, “The Research and Implement of Air Quality Monitoring System Based on ZigBee,” in *Proc. 7th Int. Conf. Wireless Communications Networking and Mobile Computing (WiCom)*, Sept. 2011.

[13] Arun Kumar et al., “Implementation of Smart LED Lighting and Efficient Data Management System for Buildings,” *Energy Procedia*, vol. 143, pp. 173–178, Dec. 2017.

[14] Marcello A. Gómez Maureira and D.O. Livia Teernstra, “ThingSpeak — an API and Web Service for the Internet of Things,” *2018*.

[15] Ming Zhao et al., “Machine-to-Machine Communication and Research Challenges: A Survey,” *Wireless Personal Communications*, vol. 97, no. 3, pp. 3569–3585, 2017.

[16] Sulayman K. Sowe, Takashi Kimata, Mianxiong Dong, and Koji Zettsu, “Managing Heterogeneous Sensor Data on a Big Data Platform: IoT Services for Data-Intensive Science,” *2014*.

[17] Chen Xiaojun, Liu Xianpeng, and Xu Peng, “IoT-Based Air Pollution Monitoring and Forecasting System,” *2015*.

[18] Chang-Se Oh, Min-Seok Seo, Jung-Hyuck Lee, Sang-Hyun Kim, YoungDon Kim, and Hyun-Ju Park, “Indoor Air Quality Monitoring Systems in the IoT Environment,” *2015*