WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

LPG GAS LEAKAGE DETECTION SYSTEM USING ARDUINO

Mahalakshmi S1, Dr. S. THILAGAVATHI2

Student of MCA., M.Phil., Ph.D., MBA, Assistant Professor, Department of computer science,
Sri Krishna Adithya College of Arts and Science, Coimbatore, Tamil Nadu, India
E-mail: 22bscs124mahalakshmis@skacas.ac.in

ABSTRACT :

Liquefied Petroleum Gas (LPG) is widely used in both domestic and industrial settings due to its efficiency as a fuel. However, due to its flammability and the risk of gas leaks, there is an increasing need for early detection systems to avoid catastrophic accidents. This paper presents the design and development of an LPG gas leakage detection system utilizing Arduino, a low-cost microcontroller platform, for real-time monitoring and immediate alert generation. The system uses an MQ-6 gas sensor for detecting the presence of LPG in the air, and an integrated alarm system to warn users in case of a leakage. The proposed system is easy to implement, cost-effective, and provides an efficient method for enhancing safety standards in environments using LPG.

INTRODUCTION :

Liquefied Petroleum Gas (LPG) is a commonly used fuel in households, kitchens, and industries. Despite its widespread use, the risk of gas leakage poses a serious hazard, including potential explosions, fires, and poisoning due to the inhalation of harmful gases. Effective detection systems are essential for preventing such hazards and ensuring safety.

Traditional gas leakage detection systems are often expensive, require complex setups, and involve non-portable equipment. In contrast, the use of an Arduino-based solution offers a simple, cost-effective, and scalable alternative. Arduino, an open-source microcontroller platform, provides flexibility for hardware integration and software development, making it ideal for designing gas leakage detection systems.

MATERIALS AND METHODS :

SYSTEM OVERVIEW :

The LPG gas leakage detection system is designed around an Arduino microcontroller, which interfaces with an MQ-6 gas sensor to detect LPG gas concentrations in the air. When the sensor detects a predefined threshold level of gas, the system triggers an alarm or activates other warning mechanisms. The system's components include:

-Arduino Microcontroller (Arduino Uno): Serves as the central unit for reading sensor data and controlling the alarm system.
-MQ-6 Gas Sensor: An air quality sensor designed to detect LPG, methane, and other combustible gases. It operates based on a sensing element that reacts with gas molecules, generating a signal proportional to the gas concentration.

- Buzzer/Alarm: Provides an audible warning when gas concentration exceeds a set threshold.
- LCD Display (optional): Displays real-time sensor readings for monitoring purposes.
- Power Supply: Powers the entire system using a DC supply.

CIRCUIT DESIGN :

The circuit design includes the following key components:

- The MQ-6 gas sensor is connected to the analog input pins of the Arduino.
- The buzzer is connected to a digital output pin of the Arduino for alerting the user.
- An LCD can be used to display real-time sensor readings.

A simple schematic includes the gas sensor's analog output connected to the Arduino's analog input pin (A0), and the buzzer connected to a digital pin (e.g., D2). Power is supplied via a 9V DC battery or an adapter.

PROGRAMMING THE ARDUINO :

The Arduino is programmed using the Arduino IDE. The program includes the following steps:

- Initialize the sensor and display.
- Continuously read the analog value from the MQ-6 sensor.
- Convert the analog value to a gas concentration level.
- Compare the concentration level with a threshold value (e.g., 500 ppm).
- Trigger the alarm and/or display an alert on the LCD if the gas concentration exceeds the threshold.

CALIBRATION:

Calibration of the MQ-6 sensor is crucial for accurate gas concentration detection. This is typically done in a controlled environment with a known concentration of LPG gas. The sensor output is adjusted based on the sensitivity to the target gas to match the desired detection threshold.

TESTING AND IMPLEMENTATION:

TESTING:

Testing is a series of different tests whose primary purpose is to fully exercise the computer-based system. Although each test has a different purpose, all work should verify that all system elements have been properly integrated and performed allocated function Testing is the process of checking whether the developed system works according to the actual requirement and objectives of the system. The philosophy behind testing is to find the errors.

A good test is one that has a high probability of finding an undiscovered error. A successful test is one that uncovers the undiscovered error. Test cases are devised with this purpose in mind. A test case is a set of data that the system will process as an input. However, the data is created with the intent of determining whether the system will process them correctly without any errors to produce the required output.

The philosophy guiding testing is centered on error identification. A robust test is characterized by a high likelihood of discovering previously undetected errors. The effectiveness of a test is measured by its ability to uncover such hidden errors. Test cases are strategically designed with this objective in mind. A test case comprises a set of data intended for processing as input by the system. The data are carefully constructed to assess the system's capability to accurately process them, generating the required output without errors.

SYSTEM TESTING:

System testing makes a logical assumption that if all the pairs of the system are correct. Software testing is a crucial element of the software quality, assurance and represents interesting anomaly for the software during earlier definition and development phases, it was attempted to build software from an abstract concept to a tangible implementation.

The testing phases of the developed system use various test data preparation. This plays a vital role in system testing. After preparing the test data the system under study was tested using those tested data. While testing the system, errors were found and corrected by using the following testing steps and corrections are also noted for future use.

TESTING METHODOLOGIES:

Anything may be the system; testing phase is the final and important phase for it to be success. It is the stage of implementation, which is aimed at ensuring that the system works accurately and effectively before live operation commences. System testing makes a logical assumption that if all parts of the system are correct the goal will be successful. The user tests the developed system and changes are made according to their needs.

The various types of testing done on the system are,

Unit testing
Integrated testing
database testing
Validation Testing
Output Testing
User Acceptance Testing
Performance

1 UNIT TESTING:

In unit testing, individual units or components of a software application are examined in isolation. This testing technique involves creating and executing specific test cases for functions, methods, or modules to verify that each unit performs as intended. The main goal is to identify defects or errors early in the development process and ensure the correctness of each unit. Unit testing provides developers with confidence in the reliability and functionality of their code by systematically validating the behavior of isolated components.

INTEGRATION TESTING:

This kind of testing is a systematic testing for constructing tests to uncover errors associated within the interface. The objective is to take unit tested modules and build a program structure. All the modules are combined and tested as a whole.

The system underwent a series of Integration tests that recorded smooth transmission of data from one module to the other. In this project the developed system is tested after integrating various modules together, and the detected errors were corrected.

DATABASE TESTING:

Database Testing is checking the schema, tables, triggers, etc. of the database under test. It may involve creating complex queries to load/stress test the database and check its responsiveness. It Checks data integrity and consistency.

VALIDATION TESTING:

Validation testing is a software testing process that assesses whether the software meets the specified requirements and satisfies the intended use case scenarios. This testing phase evaluates the overall functionality of the complete system, ensuring that it aligns with the predetermined objectives. Validation testing is focused on confirming that the software delivers the expected outcomes, meets user expectations, and performs reliably within its intended environment.

OUTPUT TESTING:

Output testing is a validation process that specifically examines the results generated by a software application. It involves comparing the actual output produced by the system against the expected or desired output. The goal of output testing is to ensure accuracy, completeness, and conformity to predefined criteria. By validating the output, this testing phase aims to identify any discrepancies or errors in the results, ensuring that the software produces the correct and expected outcomes.

USER ACCEPTANCE TESTING:

User acceptance testing (UAT) is a crucial phase in the software development lifecycle where the software is evaluated by end users to determine its suitability for their needs. This testing phase focuses on validating that the software meets user requirements, functions as intended, and is user-friendly. User acceptance testing ensures that the software aligns with the user's expectations and is ready for deployment, providing final confirmation of its readiness for real-world use.

PERFORMANCE TESTING:

Performance testing involves evaluating the responsiveness, stability, and scalability of a software application under different conditions. This testing phase assesses how well the software performs in terms of speed, reliability, and resource utilization. Performance testing includes various types such as load testing, stress testing, and scalability testing, aiming to identify potential bottlenecks, measure system responsiveness, and ensure that the application can handle expected workloads efficiently.

IMPLEMENTATION:

Implementation is the project stage where the theoretical design transforms into a functional system, instilling confidence in users regarding its efficiency and effectiveness. This phase involves meticulous planning and an investigation of the current system, considering its constraints on implementation. The design of methods to achieve the transition, along with the evaluation of changeover methods, are key tasks. Beyond planning, crucial aspects of implementation include educating and training users. The implementation process initiates with the formulation of a comprehensive plan for the system's deployment.

According to this plan, activities are systematically executed, encompassing discussions on necessary equipment and resources. Additional equipment is acquired as needed for the implementation of the new system. In a network backup system, no supplementary resources are required. The implementation phase, being the final and most crucial stage, hinges on instilling confidence in users regarding the new system's efficacy. A successful transition to the new system relies heavily on assuring users that it will function effectively. Implementation can only proceed once thorough testing confirms that the system operates in accordance with the specified requirements.

RESULTS And DISCUSSION:

The LPG gas leakage detection system was tested under controlled conditions, simulating various gas concentration levels. The system was able to accurately detect LPG concentrations as low as 300 ppm and trigger the alarm at the predefined threshold of 500 ppm.

PERFORMANCE EVALUATION:

The response time of the sensor is crucial for early detection. The system was tested for its response time to different LPG concentrations, and it was found that the system could detect leaks in a matter of seconds. The integration of the buzzer provided an effective warning mechanism, ensuring that users would be alerted promptly.

LIMITATIONS:

While the system performs well under controlled conditions, factors such as temperature, humidity, and airflow can affect the sensor's accuracy. Additionally, the MQ-6 sensor has limited sensitivity to gases other than LPG, which may lead to false readings in environments with multiple types of gas.

CONCLUSION:

This system provides a fast and cost-effective solution to avert the gas leak effect by reducing the risk to human life. The statistics of the application of gas clam on to the application can be useful to own the faulty valves and regulators prior and do the necessary replacement. Apart from detecting the leakage, a two-level prevention apparatus makes the system more valid. The cost involved in developing the system is crucially low. In recent brood, the use of LPG is taking a big giant. From the use of cylinders up to the use of petroleum lines. The biggest warning in using this technology is security. our project will prove to be resonance for households and industries.

After this project performance, can conclude that detection of the LPG gas leakage is incredible in the project system. Applicable usefully in the industrial and domestic purpose. In danger situations we can save the life by using this system. A sensor node senses gas like CO2, oxygen, propane. The estimated range of transmission and consumption of power is obtained. The simple procedures and Arduino UNO Micro controller area used to build the sensor.

FUTURE ENHANCEMENT:

This monitoring system can be further increased by using Bluetooth in place of GSM to send the messages to the user, which abetment another real-time application. For the industrial sector, the data collected by the mobile application is beneficiary and used for data analytics. The combination of other sensors like temperature, pressure sensors, etc. makes the system a home computerization project. IoT turns drones into gas observation sensors. Another very interesting and extraordinary improvement would be to board reoccurring receiver MODEMS at different positions in the geographical area carrying duplicate SIM cards. The display can be another added variant in the project. Audio output can be settled to make it user-friendly.

REFERENCES :

1. 1 Author: Michael Margolis ,Year of Publishing: 2011, "Arduino Programming in 24 Hours, Sams Teach Yourself".
2. 2 Author: Richard Blum ,Year of Publishing: 2014, "Getting Started with Arduino".
3. 3 Authors: Massimo Banzi, Michael Shiloh Year of Publishing: 2015,"Exploring Arduino: Tools and Techniques for Engineering Wizardry".