Gas Leakage Detection System using IoT

Final Senior Project Report Submitted to

The Department of Network Engineering and Security

Faculty of Computer and Information Technology

Jordan University of Science and Technology

In Partial Fulfillment of the Requirements for the Degree of Bachelors of Science in

Network Engineering and Security

By

Omar Mahmoud Al-Souqi, 132963
Daniil Mazen Al-Zoubi, 130783
Mohammad Khalaf Anagreh , 130897
Mohammad Hatem Al-Qar'an, 131961
Mohammad Khaled Al-Rashed, 131383

Supervisor: Dr. Mohammad M. Al-Shurman

Spring 2023

DEDICATIONS

We would like to dedicate this report on “Gas Leakage Detection System based on IoT” to all people who have been instrumental in our academic journey and the successful completion of this project.

First of all, we would like to express our deepest gratitude to our supervisor ‘Dr. Mohammad M. Al-Shurman’, for their invaluable guidance, continuous support, and encouragement throughout the duration of this project. Their expertise and insightful feedback have played a critical role in shaping this work.

We are also indebted to faculty members of ‘Network Engineering & Security’ at Jordan University of Science & Technology, whose knowledge and passion for their respective fields have nurtured our intellectual growth. Their dedication to teaching and research has been a constant source of inspiration for us.

To my friends and classmates, thank you for being there during the highs and lows of this journey. Your discussion, and shared experiences have made this project all the more meaningful and enjoyable.

We would like to extend our heartfelt appreciation to our families for their unconditional love, unwavering support, and belief in our abilities, and we are forever grateful for their presence in our life.

This project report is a culmination of the collective efforts and support of all those who have played a part, directly or indirectly, in its completion. It is with deep appreciation and heartfelt thanks that we dedicate this work to all of them.

<u>نموذج حقوق الملكية الفكرية لمشاريع التخرج فى قسم هندسة و أمن شبكات الحاسوب</u>

يتم قراءة وتوقيع هذا النموذج من قبل الطلاب المسجلين لمشاريع التخرج في قسم هندسة و أمن شبكات الحاسوب

تعود حقوق الملكية الفكرية لمشاريع التخرج ونتائجها (مثل براءات الاختراع أو أي منتج قابل للتسويق) إلى جامعة العلوم والتكنولوجيا الأردنية، وتخضع هذه الحقوق إلى قوانين وأنظمة و تعليمات الجامعة المتعلقة بالملكية الفكرية وبراءات الاختراع.

بناءا على ما سبق أوافق على ما يلي:

1) أن أحفظ كافة حقوق الملكية الفكرية لجامعة العلوم والتكنولوجيا الأردنية في مشروع التخرج.
2) أن ألتزم بوضع اسم جامعة العلوم والتكنولوجيا الأردنية و أسماء جميع الباحثين المشاركين في المشروع على أي نشرة علمية للمشروع كاملا أو لنتائجه. و يشمل ذلك النشر في المجلات و المؤتمرات العلمية عامة أو النشر على المواقع الإلكترونية أو براءات الاختراع أو المسابقات العلمية.
3) أن ألتزم بأسس حقوق التأليف المعتمدة في جامعة العلوم والتكنولوجيا الأردنية.
4) أن أقوم بإعلام الجهة المختصة في الجامعة عن أي اختراع أو اكتشاف قد ينتج عن هذا المشروع و أن ألتزم السرية التامة في ذلك و أن أعمل من خلال الجامعة على الحصول على براءة الاختراع التي قد تنتج عن هذا المشروع.
5) أن تكون جامعة العلوم والتكنولوجيا الأردنية هي المالك لأي براءة اختراع قد تنتج عن هذا المشروع و تشمل هذه الملكية حق الجامعة في إعطاء التراخيص و التسويق و البيع كمؤسسة راعية و داعمة لكافة الأنشطة البحثية. و يكون حق للطالب شمول اسمه على براءة الاختراع كأحد المخترعين، و في حال تم إعطاء تراخيص أو تسويق و بيع لأي من منتجات المشروع يمنح المخترعون بما فيهم الطالب نسبة من الإيرادات حسب تعليمات البحث العلمي في جامعة العلوم والتكنولوجيا الأردنية.

اسم الطالب	دانيال مازن الزعبي	التوقيع
إسم الطالب	محمد خلف عناقرة	*التوقيع*
إسم الطالب	محمد خالد الراشد	*التوقيع*
إسم الطالب	محمد حاتم القرعان	*التوقيع*
إسم الطالب	عمر محمود السوقي	*التوقيع*
إسم المشرف	**د.محمد الشرمان**	***التوقيع***

تاريخ ....9/6/2023.........

ACKNOWLEDGMENTS

First, **Alhamdulillah** for everything, and as it’s said in the Holy Qur’an -all praises be to Allah-

“Then remember Me; I will remember you. Be grateful to Me, and do not reject Me.”

(Surah al-Baqarah 2:152)

We would like to thank our supervisor Dr. Mohammad M. Al-Shurman for his amazing effort, patience, kindness and taking care of us along this important journey in our lives who believed in our ideas gave us valuable comments and suggestion that helped us in making our project.

Without forgetting to thank our colleagues and friends for their endless support and pushing us all the time.

ABSTRACT

Every year we hear about dramatic accidents that ends people live from gas suffocation, especially in winter when people use gas heaters, and as we say 'every problem born with a solution', our researches led us to work on a solution that aims to detect gas leakage and alert people especially people with special needs and expel the gas from the area as soon as possible. This project proposes a gas leakage detection system using an MQ-2[1] sensor, ESP32 micro controller [2], smf27 buzzer [3], SIM800L [4] module, and LED lights [5].

Gas Sensor MQ2, is known for its high sensitivity to multiple combustible gases, serves as the primary gas detection component in the system. It continuously measures the concentration of gases like H2, LPG, CH4, CO, Alcohol, Smoke or Propane. The sensitivity of the sensor can be adjusted by potentiometer. [7]

ESP32 is a micro controller that acts as the central processing unit and a highly-integrated solution for Wi-Fi and Bluetooth IoT applications that will receive the input data from the MQ2 sensor and analyze it in real-time and then take actions based on them.

Upon detecting the presence of hazardous gases above a predefined threshold, the system triggers appropriate responses to alert the user. We used different methods to alert the user, starting from buzzer that is employed to emit a loud audible alarm, drawing immediate attention to the potential danger. Simultaneously, LED lights are activated to provide visual indications, enhancing the visibility of the gas leak alarm. As an action to be done we added fans that's role is to kick out the polluted air away.

To enable remote monitoring and notification, a SIM800L module is incorporated into the system. It facilitates the communication between the gas leakage detection system and the user's mobile device. When a gas leak is detected, the system sends an alert message to the user's registered phone number, providing real-time information about the gas leakage incident. For monitoring we used Blynk Application [6] that has a lot of features user can use to control the system.

Through the implementation of this gas leakage detection system, the safety of industrial and domestic environments can be significantly improved. Early

detection and prompt response to gas leaks contribute to a safer and more secure living and working environment. Future enhancements could involve the integration of additional sensors that can connect a lot of rooms areas together.

TABLE OF CONTENTS

PAGE

LIST OF TABLES

TABLE **PAGE**

LIST OF FIGURES

FIGURE **PAGE**

CHAPTER 1

INTRODUCTION

Internet of things (IoT) has become one of the most connected technologies with billions of instances around the world.
The ability to monitor, detect and alert gas leakage remotely is becoming increasingly important as more and more people rely on technology for work and daily events management.

1.1.Background and Motivation

1.1.1..Gas Leakage issues

1.1.1.1. Introduction to Gas Leakage Problems

- **Gas leakage issues hold significant importance in various industries and residential areas due to the potential risks they pose. Understanding the significance of gas leakage problems is crucial in developing effective detection and prevention systems.**

- **There a lot of potential risks associated with gas leakage such as fire hazards (In 2010[8], a natural gas pipeline explosion in San Bruno, California, resulted in a massive fire that destroyed dozens of homes, caused multiple fatalities, and injured numerous people. The explosion was triggered by a gas leak, emphasizing the destructive potential of such incidents.) and health hazard Gas leaks can pose severe health hazards to individuals exposed to toxic or asphyxiating gases like (Carbon monoxide (CO) leaks are a significant concern, as this odorless and colorless gas can be fatal).**

1.1.1.2.Consequences of Gas Leaks

- **Gas Leakage cause a lot of damage in many aspects such as property damage, environmental pollution and human causalities. That's what people are always afraid of because one mistake can cause huge damage.**

- **In September 2019[9], a gas leakage incident occurred in Zarqa, Jordan, the incident took place in a densely populated neighborhood, where a gas leak from a residential building resulted in a major explosion. The explosion caused significant damage to nearby buildings, including the collapse of several structures, and led to multiple casualties.**

1.1.2 .Existing Solutions and Limitations

1.1.2.1 Overview of Gas Detection Methods

- **In the past people was detecting gas leakage manually by having a schedule by when the last time they checked any gas source and it was leading a lot of mistakes such as forgetting to do a check the gas source periodically.**

1.1.2.2 Limitation of Gas Leakage Detection Systems

- **-Traditional Gas Leakage Detection Systems has single type of alerting and a single type of notifying, our system has two types of alerting methods to alert people without forgetting that in the modern world, people with special needs are living alone so we added this feature for them specifically.**

- **-Disabled people are part of our society so adding an automatic action to kick out the polluted air was essential.**

1.2. Objectives and Scope

1.2.1 Project Objectives

1.2.1.1 Purpose of Project

- **The main objective of developing a gas leakage detection system is to have a safe environment always by fast detecting of gas leakage and take automatic actions and send alerts and notifications to the user.**

- **By detecting gas leakage, we are saving people's life by the well of Allah and take automatic actions instead of human's actions to reduce the time needed to refresh the air and in cases there are children or people with special needs.**

1.2.1.2 Key Goals and Deliverables

- **One of the key goals is to achieve real-time gas leak detection capabilities. The system should be able to continuously monitor the environment for gas leaks and provide immediate alerts or notifications when a leak is detected. This ensures prompt action can be taken to mitigate risks and prevent accidents.**

- **Adding more than one method of alerting and notifying give us ability to let the project be suitable for all people and integrate people with special needs with the rest of society.**

1.2.2 Scope of the project

1.2.2.1 Targeted Gas Types and Environment

- **Our system using MQ2 sensor can detect different types of gases such as hydrogen, LPG (liquid petroleum gas), methane, carbon monoxide, smoke, and propane in.**

- **Having the system in closed environment such as building, rooms and inserting it nearby any gas sensor will give the user the best results.**

1.2.2.2 System Components and Integration

- **Having a system that aims to detect, notify and alert contains a MQ2 sensor to detect gas leakage, LED lights and sound buzzer to alert, SIM800L and Blynk Application to notify and the last thing is fans to kick out the polluted air.**

- **MQ2 is a high sensitivity sensor that can detect gas leakage so fast, so inserting it near any gas source give the best results and It can be handled in the wall or inserted above which means can be done as the user needs.**

CHAPTER 2

REVIEW OF RELATED LITERATURE

The Internet of Things (IoT) has made it possible to develop creative and effective solutions across a range of industries, including safety and environmental monitoring. Gas leak monitoring is one such area of interest where IoT-based technologies offer considerable advantages over conventional approaches. Fortunately for us, there are many scientific research articles, theses, dissertations, and other scholarly works available that we can review to understand, study the approaches used, and identify problems in this quickly developing sector. When initiating this project, extensive research and thorough preparations are essential. Teammates need to conduct an extensive search to gather knowledge and understand the existing landscape of IoT-based projects detection systems. By examining previous related work, the team can gain insights into the existing methodologies, technologies, challenges, and advancements in gas leakage detection. This research helps in identifying gaps in the literature and provides a foundation to build upon existing knowledge, ensuring that the project takes a comprehensive and innovative approach to address the specific requirements and challenges in gas leakage detection based on IoT.

2.1. IoT in general

2.1.1. A Review on Internet of Things (IoT) paper

During our search we found a lot of search paper that explains IoT in general and we relied a lot on this search paper that talks about: The idea of the Internet of Things (IoT), which describes a network where every device and things are connected to the internet, is covered in the extract. The Internet of Things (IoT) is a technology that permits communication between intelligent machines and has the potential to alter the structure of the current internet. IoT's brief history is discussed, beginning in 1982 with a modified Coca-Cola machine that was connected to the internet. The IoT architecture and vision are then presented in the paper. The architecture is made up of six layers: coding, perception, network, middleware, application, and business layers. The vision involves a network where everything is connected and can interact. IoT development technologies such as RFID, wireless sensor networks, cloud computing, networking, nanotechnologies, MEMS technologies, and optical technologies are also covered. IoT applications for smart homes, smart transportation, and smart industries are mentioned in the excerpt's conclusion [10].

2.1.2. History of IoT paper

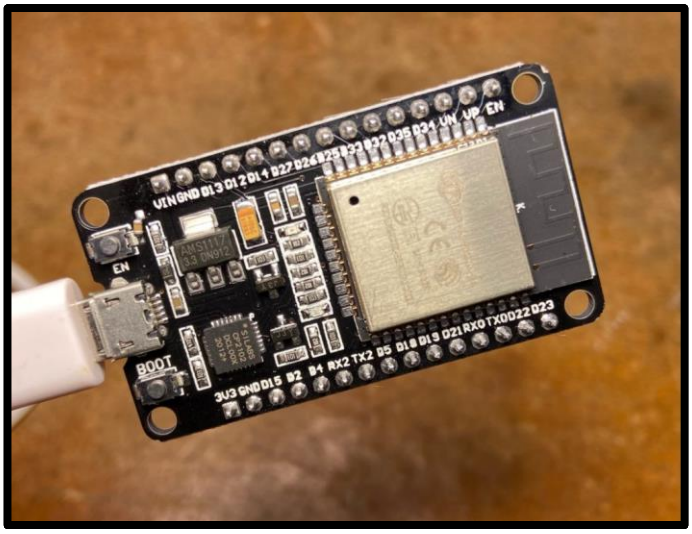
Any subject's history must be studied in depth if one wants to fully comprehend it. Whether it be a scientific field, a cultural phenomenon, or a technical advancement, exploring its past offers insightful information and context that enhances our learning process. By delving into a subject's past, we may better understand its growth, spot significant turning points, draw lessons from past achievements and mistakes, and acquire a greater understanding of its present situation and potential future developments. And this article was perfect because it talked about the history of the Internet of Things (IoT) dates to the invention of early communication methods like the telegraph and radio. Although the word "IoT" was first used in 1999, instances of connected gadgets, including a Coca-Cola machine, were already in use at that time. RFID-based inventory tracking by businesses like Walmart helped the Internet of Things (IoT) gain pace in the early 2000s. Smart homes, smart cities, and smartphones have all been added to the Internet of Things throughout time, enabling targeted marketing and healthcare monitoring. With connected cars and the development of autonomous vehicles, the automotive sector has also embraced the Internet of Things. The Internet of Things (IoT) is now present in neighborhoods, where smart gadgets and sensors improve security and energy usage. The Internet of Things (IoT) is still reshaping sectors and changing how people use technology, increasing productivity and connectedness [11].

2.2. Parts and software we used

2.2.1. ESP32 article

This article is one of many articles we read about the ESP32, we summarized it as follows: Espressif created the low-cost, low-power ESP32 microcontroller. It is the ESP8266's replacement and comes with a dual-core CPU, Wi-Fi, and Bluetooth capabilities. The ESP32 is well-liked because it is reasonably priced, uses little power, supports Wi-Fi and Bluetooth, has a dual-core design, and has a robust peripheral interface. Both the MicroPython firmware and the Arduino programming language are compatible with it. The USB-to-UART interface, pin arrangement, antenna connection, battery connector, and any extra hardware features should all be taken into account when selecting an ESP32 development board. the ESP32 DEVKIT DOIT board is advised. The Arduino IDE may be used to program the ESP32, and the ESP32 DEVKIT DOIT board is frequently used as a reference [12].

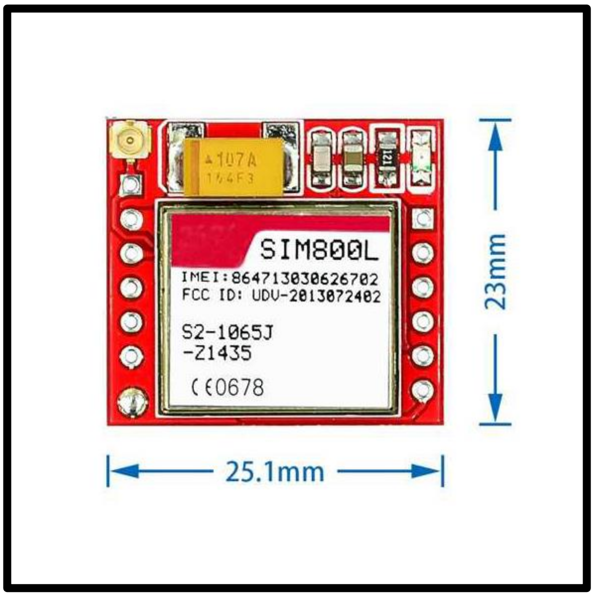
Figure 2.1 ESP32



2.2.2. SIM800L article

We relied a lot on this article to understand the functionalities of SIM800L. We summarized it as follows: A compact GSM modem that may be utilized in numerous Internet of Things projects is the SIM800L GSM/GPRS module. It can operate internationally since quad-band GSM/GPRS networks are supported. The module has the ability to send SMS messages, make phone calls, establish a GPRS internet connection, and receive FM radio broadcasts. It may be powered by a Li-Po battery or a DC-DC buck converter and requires an additional antenna for network access. LED indications for network status are included in the hardware of the module. Software serial communication may be used to link it to an Arduino. The Arduino code included in the overview explains how to send an SMS, verify network connectivity, and interface with the SIM800L module using AT commands [13].

Figure 2.2 SIM800L

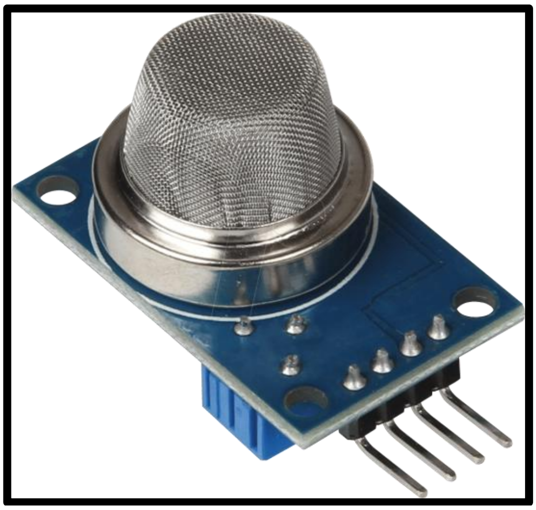


2.2.3 MQ2 article

Since this sensor relies on semiconducting Prenosils it needs a lot of reading to fully understand how it works. We relied on this article and summarized as follow:

A flexible sensor, the MQ2 gas/smoke sensor can identify a wide range of gases in the atmosphere, including smoke, alcohol, propane, hydrogen, methane, and carbon monoxide. It uses 800mW of power while operating at 5V DC. The sensor makes use of metal oxide semiconductor technology, and it detects gases by adjusting the resistance of the sensing material. The interior construction of the MQ2 sensor is made up of a tin dioxide-coated aluminum oxide ceramic detecting element. Tin dioxide is surface adsorbed with oxygen when the sensor is heated, potentially forming a barrier. The potential barrier becomes lessened in the presence of reducing gases, enabling current to pass through the sensor. Digital and analog outputs are also available from the MQ2 sensor module. While the digital output shows the presence of flammable gases, the analog output voltage fluctuates proportionally to gas concentration. The module has a potentiometer for altering the digital output's sensitivity. Additionally, it contains LEDs that show the gas concentration and power status. The sensor must be calibrated, especially if it has been sitting about for a while. For best accuracy, the sensor must be fully warmed up for 24 to 48 hours. The sensor readings may start out high during the warm-up phase before eventually stabilizing. By reading the analog input pin on the Arduino, you may use the analog output to measure the gas concentration. The measurements may be utilized to establish a threshold value that, when exceeded by the gas concentration, will cause an action to be taken [14].

Figure 2.3 MQ2 Sensor



2.2.4. Blynk application article

We found a lot of research and articles about Blynk app, but we relied mostly on documentation of the Blynk official website, and we came up with the following. Users of the iOS and Android mobile apps Blynk may monitor data from linked devices and remotely operate them. Developer Mode and End-user Mode are the two operating modes for the app. Users may build a mobile dashboard using modular user interface components known as widgets in developer mode. Widgets can do a variety of tasks including buttons, sliders, and charts. Based on how each Widget functions, each one has unique settings. Both makers and end users utilize end-user mode, which is focused on controlling devices, automations, and notifications. Depending on the settings, the home screen has up to three tabs. Users may add new devices using the menu, and the Devices tab includes all previously created devices along with associated tiles. Users may build triggers based on time, sunset/sunrise, device triggers, manual execution (scenes), or any combination of these using the Automation tab, which is available when at least one DataStream is made available for automation. A list of alerts from all devices is shown on the alerts tab [15].

CHAPTER 3

ANALYSIS AND DESIGN

In this chapter, we will delve into the detailed analysis and design aspects of our system, aiming to develop a robust and efficient solution for gas leakage detection and prevention.

The primary objective of this chapter is to analyze the requirements, constraints, and specifications of our gas leakage detection system. We will conduct a comprehensive analysis of the system's functional and non-functional requirements, considering factors such as accuracy, reliability, scalability, and real-time monitoring capabilities. This analysis will serve as a foundation for designing an effective and tailored solution that meets the specific needs of our project.

Following the analysis phase, this chapter will focus on the design of our gas leakage detection system using IoT. We will discuss the overall system architecture, sensor placement strategies, communication protocols, and data processing methodologies. By considering the insights gained from the literature review and previous works, we will develop a well-designed and optimized solution that integrates seamlessly with existing IoT infrastructure.

Throughout this chapter, we will emphasize the importance of user-centered design principles and usability considerations. By understanding the end-users' needs and ensuring an intuitive and user-friendly interface, we aim to enhance the system's usability and ease of operation. Additionally, we will address security and privacy concerns, implementing appropriate measures to protect sensitive data and ensure the system's integrity.

By undertaking a rigorous analysis and design process, we aspire to develop a gas leakage detection system that excels in its functionality, reliability, and adaptability. This chapter will serve as a comprehensive guide, outlining the key analysis and design considerations, methodologies, and decisions that contribute to the successful implementation of our gas leakage detection system using IoT.

3.1. Design Requirements

To achieve the goal of developing an efficient and reliable gas leakage detection system that aims to do 3 main actions detect, alert and take actions, the project needs to fulfill the following design requirements:

3.1.1 Accuracy and Reliability:

- The system should accurately detect and identify gas leaks, ensuring a minimal false alarm rate to avoid unnecessary disruptions and alarms.
- It should provide reliable and consistent performance, minimizing instances of missed detections or false negatives.

3.1.2 Real-time Monitoring:

- The system using Blynk Application will continuously monitor gas levels in real-time, providing immediate feedback on any detected anomalies or deviations from normal gas concentrations.
- It should offer real-time visualization of gas levels and alerts to enable proactive response measures.

3.1.3 Communication and Data Integration:

- The system should establish a reliable and secure communication network between the gas sensors, central processing unit –ESP32- , and user interface using Blynk application.

3.2. Engineering Standards

Our gas leakage detection system using IoT will adhere to various engineering standards to ensure interoperability, compatibility, security, and reliability. The following are the engineering standards that our project will follow:

3.2.1 Internet Protocols and Standards:

- Internet Protocol Suite (TCP/IP): Our system will utilize the TCP/IP protocol suite for communication between the gas sensors, central processing unit –ESP32- , and user interface –Blynk-. This ensures compatibility with existing network infrastructure and enables seamless integration with the internet.

3.2.2 Wireless Protocols:

- Wi-Fi (IEEE 802.11): The system may utilize Wi-Fi technology for wireless communication between the gas sensors and the ESP32. This standard provides reliable data transmission and widespread compatibility.
- Bluetooth (IEEE 802.15.1): Bluetooth wireless technology will be employed for short-range communication between the gas sensors and the central processing unit. Bluetooth offers low-power

consumption, ease of connectivity, and compatibility with a wide range of devices.

<u>These two protocols are options that can be used in ESP32</u>

3.3. Realistic Constrains

Designing a gas leakage detection system using IoT involves considering various realistic constraints that will impact the overall design. These constraints can be categorized into economic, environmental, manufacturability and sustainability, and other factors. Quantitative expressions of these constraints are crucial to guide the design process effectively.

3.3.1 Economic Constraints:

- Bandwidth Consumption: The system should optimize the bandwidth consumption to minimize the impact on network infrastructure and ensure efficient data transmission.
- Memory Footprint: The software application developed for the system should have a limited memory footprint to optimize resource utilization and enable smooth operation on constrained devices.
- Cost Considerations: The overall design should be cost-effective, taking into account the affordability of components, sensors, and communication modules.

3.3.2 Environmental Constraints:

- Battery and Power Consumption: The system should minimize power consumption to extend the battery life of wireless sensors and reduce the environmental impact of frequent battery replacements.
- Energy Efficiency: The design should aim for energy-efficient operation, utilizing power-saving techniques and optimizing the use of resources.

3.3.3 Manufacturability and Sustainability Constraints:

- Final Product Dimensions and Weight: The design should consider the physical dimensions and weight of the gas leakage detection system to ensure ease of installation, portability, and integration into different environments.
- Materials and Components: The selection of materials and components should prioritize sustainability, considering factors such as recyclability, environmental impact, and compliance with regulations.

3.4 Alternative/Different Design Approaches/Choices:

- In this section, various design options will be explored. Each option will be presented, highlighting its advantages and disadvantages.

Table 3.1 : Comparison between micro controllers

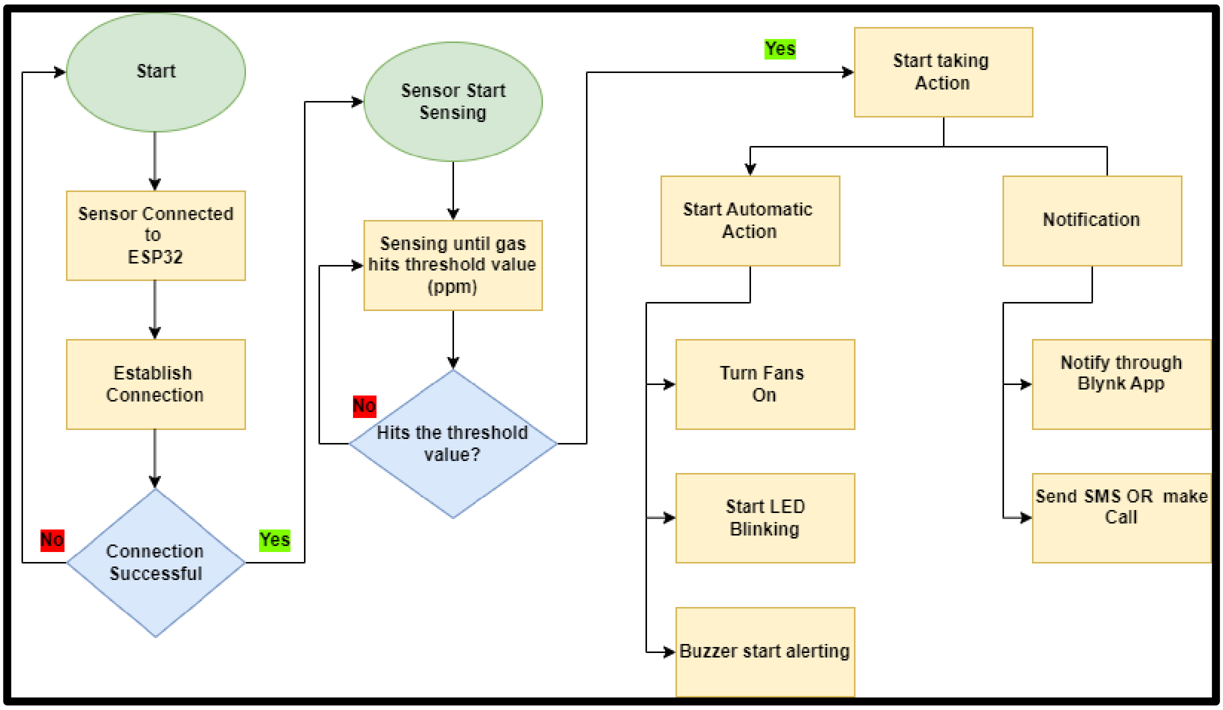
-	ESP32	Arduino Uno
Cost	Less Expensive	Expensive
Using / Programming	Harder than Arduino Uno	Suitable for beginners
Pins I/O	18 ADC , 48 GPIO	6 ADC, 14 Digital Pins
Advantages	Smaller / light weight	Bigger and heavier than ESP32

3.5 Developed Design:

- This section will present the design developed for the gas leakage detection system. The thought process behind the design will be explained, considering the requirements, constraints, and engineering standards. The developed design will be illustrated through flow charts, schematics, pseudo codes, system level diagrams, architecture diagrams, system/transistor diagrams, and any other appropriate visual representations. This will provide a comprehensive understanding of the design's structure and functionality.

- By addressing these realistic constraints and exploring alternative design approaches, our gas leakage detection system using IoT will be developed in a manner that optimizes resources, considers environmental impact, meets economic considerations, and ensures the manufacturability and sustainability of the final product.

Figure 3.1: Flow Chart



PSEUDO CODE:

```
Green LED turn on;

While(gas_level > gas_threshold)

 {

  Green LED turn off;
  Red LED turn on;

  CPU Fans turn on;

  Smf buzzer turn on;

  Notify by Blynk;

  Send SMS by SIM800L; }
```

CHAPTER 4

IMPLEMENTATION

To successfully wrap up our project, we used multiple hardware components together and connected them to the ESP32 dev kit. The result of these hardware connections was a manly to warn the user if any gas leakage is detected in an easy and simple way.

4.1 The input:

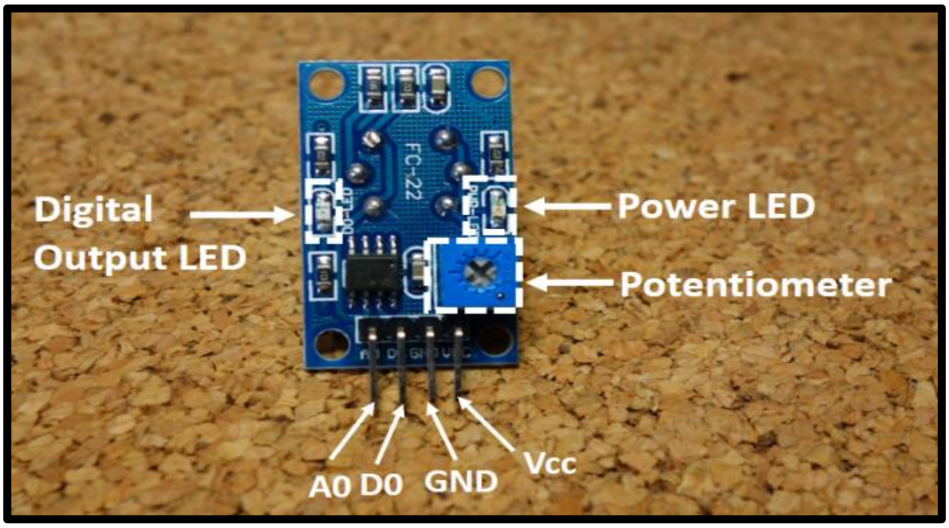
4.1.1 Setting the appropriate threshold for Gas sensor (MQ2):

For precise and consistent gas level detection, it is essential to set the proper trigger threshold for the gas sensor. In this procedure, the precise gas concentration at which the sensor should activate an alarm. It requires careful consideration of a number of variables, including the kind of gas being detected, the setting in which the sensor is being used, and the sensor's sensitivity. We can optimize the sensor's performance, reduce false alerts, and effectively reduce possible dangers related to gas leaks or hazardous circumstances by precisely calibrating the threshold.

4.1.2 trial-and-error approach:

Trial-and-error threshold setting for the MQ2 gas sensor requires a methodical process of experimentation and adjusting. The sensor is initially calibrated in a clean environment to a baseline level. The sensor's response is then monitored and recorded after being exposed to increasing amounts of the target gas. The threshold is gradually changed throughout the trial phase while the sensor's response is assessed for each concentration level. The objective is to determine the threshold at which the target gas may be consistently detected while reducing false alarms by the sensor. The ideal threshold value can be found by carefully observing and analyzing the sensor's performance at various threshold levels. This number makes sure that sensitivity and specificity are balanced, allowing for accurate target gas detection while lowering the possibility of false positives or false negatives. In order to keep the threshold setting's effectiveness and dependability over time, regular calibration and validation are required [16].

Figure 4. 1 Rear Side of MQ2 Sensor



The senor can be adjusted using the potentiometer as shown above.

4.1.3 Why this approach:

Due to its intrinsic effectiveness and adaptability, the trial-and-error approach is a useful problem-solving technique applied in a variety of industries. It enables people to investigate many choices, test out various solutions, and pick up lessons from their failures along the way. One can learn important things about the issue at hand by methodically attempting several solutions and evaluating the results. This strategy enables people to iterate and improve their tactics depending on in-the-moment feedback, which builds creativity, flexibility, and resilience. Additionally, the trial-and-error approach encourages teammates to find creative solutions and overcome obstacles through a process of ongoing experimentation.

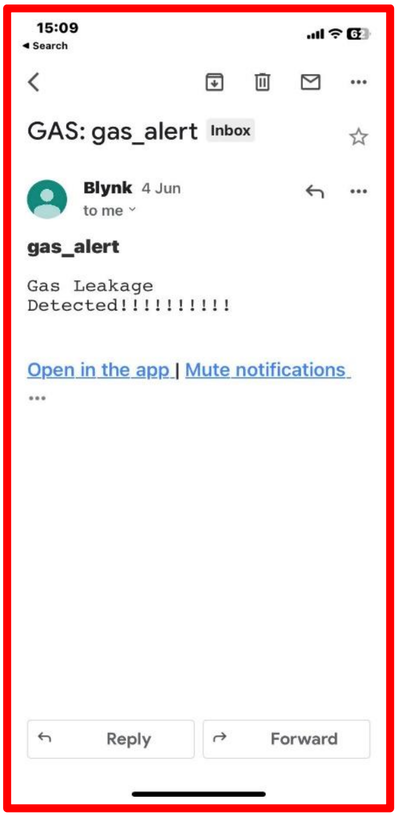
4.2 The outputs:

4.2.1 mobile phone using Blynk app:

A practical and effective option is to use the Blynk app to identify gas leaks in a project that makes use of the MQ2 sensor. Real-time gas detection is simply accomplished by integrating the MQ2 sensor with the Blynk platform. Through the Blynk app, consumers may remotely check gas levels using their smartphones or other devices thanks to its user-friendly UI. The MQ2 sensor provides data to the Blynk app, which displays the gas levels in a simple and understandable way, when it detects the presence of gas. In order to ensure that the gas leak can be fixed in a timely manner, the Blynk app can also be configured to send messages or warnings to users when gas levels surpass a predefined threshold. The MQ2 sensor's integration with the Blynk app not only improves safety measures but also gives users peace of mind by enabling continuous monitoring and quick response to instances where there may be gas leaks.

Figure 4. 2 Blynk Notifying

Figure 4. 3 Blynk Notifying



4.2.2 mobile phone using SIM800L:

The SIM800L module is used to permit phone calls, providing an efficient and quick means of communication. By incorporating the SIM800L module into the project, gas leak detection by the MQ2 sensor can cause the SIM800L module to call a predetermined user's when it detects the leak. In the event of a gas leak, this enables real-time warning and immediate action. The GSM features of the SIM800L module allow for communication across cellular networks, offering dependable connectivity and greater coverage. The project can improve safety measures by using this integration to directly inform customers via phone calls, allowing them to take prompt action to deal with the gas leakage scenario.

Figure 4. 4 SIM800L Notify

4.2.3 Buzzer:

This kind of output is mainly used for people that don't use mobile phones, for example children. We have implemented an SFM buzzer as a safety measure to detect gas leakages. When a gas leak occurs, the SFM (Sound Frequency Modulation) buzzer acts as an auditory alarm mechanism to notify people around. The SFM buzzer is turned on when the gas concentration exceeds a predetermined threshold as determined by a gas sensor. The loud, recognized sound it makes ensures that people are instantly made aware of the potential risk. This implementation is essential for adding an extra degree of security since it makes sure that even in crowded or noisy areas, people may quickly become aware of the gas leak and take the required safeguards.

4.2.4 Extra level of safety (LED lights):

We've added an extra level of safety feature that uses LED lights to turn on when gas leaks. We used green and red LED lights, green are turned on when the gas level is less than threshold value but when the gas leakage is detected, red lights are turned on and the green lights are turned off. The LED lights immediately turn on when the sensor detects a gas leakage and sends a signal to them to do the same. This implementation's goal is to give people nearby a visual cue that will warn them of a potential threat and push them to take immediate action. In addition to improving safety, this sophisticated technology enables quick reactions to gas leaks, reducing the dangers connected with such occurrences.

CHAPTER 5

TESTING AND EVALUATION

In this section, every part that needs to be programmed and tested will be listed here separately, as we programmed our code in this way and then combined all coding segments.

5.1 MQ2 sensor with alerting parts

5.1.1 LED light

The code segment below reads the ppm value and stores it Gas_Value, then if the value exceeds the predefined threshold the LED turns on.

```
void loop()
{
  Gas_Value = analogRead(PIN_No_MQ2);
  if (Gas_Value > Threshold)
  {
    digitalWrite(PIN_No_LED, HIGH);
  }
  else
  {
    digitalWrite(PIN_No_LED, LOW);
  }
  delay(1000);
}
```

5.1.2 SFM buzzer

The code segment below reads the ppm value and stores it Gas_Value, then if the value exceeds the predefined threshold the SFM buzzer starts making sound.

```
void loop()
{
  Gas_Value = analogRead(PIN_No_MQ2);
  if (Gas_Value > Threshold)
  {
    ledcWrite(buzzerChannel, buzzerResolutionMax / 2)
  }
  else
  {
    ledcWrite(buzzerChannel, 0);
  }
  delay(1000); }
```

5.2 MQ2 sensor with taking action part

5.2.1 external Fan

The code segment below reads the ppm value and stores it Gas_Value, then if the value exceeds the predefined threshold the fan starts flushing out gas.

```
void loop()
{
  Gas_Value = analogRead(PIN_No_MQ2);
  if (Gas_Value > Threshold)
  {
    digitalWrite(PIN_No_Fan, HIGH);
    delay(FAN_CONTROL_DURATION);
  }
  else
  {
    digitalWrite(PIN_No_Fan, LOW);
    delay(FAN_CONTROL_DURATION);
  }
  delay(1000);
}
```

5.3 MQ2 sensor with notifying parts

5.3.1 MQ2 sensor with SIM800L

The code segment below reads the ppm value and stores it Gas_Value, then if the value exceeds the predefined threshold the SIM card inside the SIM800L will send an SMS message.

```
void loop() {
  int Gas_Value = analogRead(PIN_No_MQ2);
  if (Gas_Value > Threshold)
  {
    if (sendSMS(phone, "Gas leack detected"))
    {
      Serial.println("SMS sent successfully");
    }
    else
    {
      Serial.println("Failed to send SMS");
    }
  }
  delay(1000);
}

bool sendCommand(const char* command, const char*
expectedResponse)
{
  sim800lSerial.print(command);
  delay(1000);
```

```
    String response = "";
    while (sim800lSerial.available())
    {
      response += sim800lSerial.read();
    }
    return response.indexOf(expectedResponse) != -1;
  }

  bool sendSMS(const char* phoneNumber, const char*
  message)
  {
    sim800lSerial.print("AT+CMGS=\"");
    sim800lSerial.print(phoneNumber);
    sim800lSerial.println("\"");
    delay(1000);

    sim800lSerial.print(message);
    delay(1000);

    sim800lSerial.write(26);
    delay(1000);

    return sendCommand("", "OK");
  }
```

5.3.2 MQ2 sensor with BLYNK application

The code segment below is for the BLYNK app to alert when gas leaks.

```
char auth[] = "";
char ssid[] = "";
char pass[] = "";
char ssid[] = "";
char pass[] = "";
void setup()
{
  Serial.begin(9600);
  pinMode(32,INPUT);
  pinMode(33,OUTPUT);
  pinMode(12,OUTPUT);
  WiFi.begin(ssid,pass);
  Blynk.begin(auth,ssid,pass);
}
void loop()
{
  gas = map(analogRead(32),0,4095,0,1000);
  blynk_send_reading();
  Serial.println(gas);
  if(gas>500)
  {
     Blynk.logEvent("gas_alert") ;
  }
```

```
}
void blynk_send_reading(){
  Blynk.virtualWrite(V0,gas); }
```

<u>Here is a QR code that presents our testing part of our project</u>

Figure 5. 1

The link is in references page[17]

CHAPTER 6

CONCLUSION

<u>'Alhamdulillah for all blessings and for his guidance all the time.'</u>

This chapter is a conclusion of our work on graduation project 'Gas Leakage Detection System using IoT' that its main goal is to integrate IoT in our everyday life activities to let it be an essential part of our lives, to sum everything, gas leakage can occur everywhere and anytime so we worked to detect it, alert and take actions using technology and IoT to take actions automatically instead of manual detecting and repairing by humans. It highlighted the need of continuous sensing of the air, using different types of alerting and taking actions. Our goal is to save people's life and be a reason in making them safe.

As any project it has strengths and weaknesses and it's normal because perfectness can't be achieved.

6.1 Lessons Learned and Challenges:

6.1.1 Lessons Learnt: In this subsection, we will reflect on the lessons we have learnt throughout the development and implementation of the gas leakage detection system. We will discuss managing project timelines, and addressing unexpected issues. We will highlight the valuable knowledge and experiences that can guide future projects in this domain.

6.1.1.1 We learnt that first phase is the most important phase in the project that contains researching, reading a lot of blogs and opinions about ideas, searching for what people really need in having solution for their problems, planning for the next step and managing the work.

6.1.1.2 We learnt how the code should be strict and cover all the test cases that user need and has the best results after testing all the cases that can happen.

6.1.1.3 We learnt how to manage our work between dividing the work between group members helping each other and managing the unexpected issues such as code errors and hardware issues.

6.1.2 Challenges Faced: In this subsection we will address the challenges encountered during the project's lifecycle. We will identify and discuss the technical, and operational obstacles that we encountered and the strategies employed to overcome them. By acknowledging these challenges, we can provide valuable insights for future researchers and developers who may undertake similar projects.

6.1.2.1 Hardware and availability: We faced some problems in finding the suitable hardware for our project that fulfil our needs because there aren't a lot of hardware shops in Jordan that have a good collection of hardware and normal prices and we were forced to change MQ5 sensor into MQ2 and the SIM800L chip takes the big SIM card which was difficult to get the old version of SIMs and we didn't find any air suction machine that can used to kick out the polluted air so we were forced to used computer CPU fans that let us use another electrical chips like transistors and resistors .

6.1.2.2 Choosing the best threshold value of gas level that is not very low so it will be considered as false positives and not very high so it will be prominent to users was hard it needed endless times of testing.

6.1.2.3 We couldn't make a box to contain all chips in one plastic box because to make a customized one we need 3D printing and it's hard to make it because there isn't a lot of choices in our city.

6.2 Conclusion and Future Outlook:

6.2.1 Summary of Achievements: In our project we achieved our main goal is detecting gas leakage first and then alert people using more than one method and finally take automatic actions such as notifying all these action are automatic actions that can occur in less than 10 seconds.

6.2.2 Future Outlook: Our future outlook is to let the 'Gas Leakage Detection System be installed in every room that contains a gas source and connect them together in the same user interface and in industrial cases to connect them with civil defense directly.

REFERENCES

Follow this style of referencing using single spacing for each entry and using normal line spacing between entries.

[1] MQ2 Sensor: https://www.pololu.com/file/0J309/MQ2.pdf

[2] ESP32 devkit v1: https://roboeq.ir/files/id/4034/name/ESP32%20MODULE.pdf/

[3] smf-27 buzzer: https://www.kwtss.com/wpcontent/uploads/2020/05/SFM27-buzzer-Datasheet.pdf

[4] SIM800L: https://www.makerhero.com/img/files/download/Datasheet_SIM800L.pdf

[5] LED lights: https://www.farnell.com/datasheets/1498852.pdf

[6] Blynk Application : https://docs.blynk.io/en/

[7] MQ2 Paragraph: https://wiki.seeedstudio.com/Grove-Gas_Sensor-MQ2/

[8] San Bruno Incident: https://en.wikipedia.org/wiki/San_Bruno_pipeline_explosion

[9] Zarqa-Jordan 2019 Incident: https://en.royanews.tv/news/18538/2019-08-29

[10] **A Review on Internet of Things (IoT) paper:** https://www.researchgate.net/profile/Nirmala-Svsg/post/What_are_the_latest_developments_in_IOT_architectures/attachment/59d6387d79197b8077995b28/AS%3A397885586853892%401471874724584/download/56.pdf

[11] **History of IoT paper:** https://www.dataversity.net/brief-history-internet-things/

[12] **ESP32 article:** https://randomnerdtutorials.com/getting-started-with-esp32/

[13] **SIM800L article:** https://lastminuteengineers.com/sim800l-gsm-module-arduino-tutorial/

[14] **MQ2 article:** https://lastminuteengineers.com/mq2-gas-senser-arduino-tutorial/

[15] **Blynk application article:**
https://docs.blynk.io/en/blynk.apps/overview

[16] **trial-and-error approach:**
https://en.wikipedia.org/wiki/Trial_and_error

[17] Link of the our project demonstration

https://drive.google.com/drive/folders/14Zxkd3-7kNmE7QlZ7eBIeFKhRljL0aly?fbclid=IwAR2cHf5JWYDIDRxRIt24Swei4lj4SOMSfc-PSysi4tYPlkdDvvfEkECcq_4

APPENDICES

APPENDIX A

Gas Detecting

```
// Include required libraries
#include <WiFi.h>
#include <WiFiClient.h>
#include <BlynkSimpleEsp32.h>
#include <SoftwareSerial.h>

// Blynk credentials
char auth[] = "Ps2nflFRN--bS-30ezgKo0OwmKfWP97X";

// WiFi credentials
char ssid[] = "Anagreh-2.4G";
char pass[] = "123456789";

// Pin definitions
#define MQ2_PIN 34
#define GREEN_LED_PIN 35
#define RED_LED_PIN 32
#define BUZZER_PIN 33

// Gas threshold level
const int gasThreshold = 500;

// Current gas value
int gasValue = 0;

// SoftwareSerial for SIM800L module
SoftwareSerial sim800lSerial(22, 23); // TX & DX 22, 23
```

```
// Blynk function to send gas reading to virtual pin V0
void blynk_send_reading() {
Blynk.virtualWrite(V0, gasValue);
}

// Function to send an SMS message
void sendSMS(const String& phoneNumber, const String& message) {
sim800lSerial.println("AT+CMGS=\"+962799176197\"");
delay(100);

sim800lSerial.println(message);
delay(100);

sim800lSerial.write(26);
delay(100);
}

void setup() {
// Initialize serial communication
Serial.begin(9600);

// Configure pin modes
pinMode(MQ2_PIN, INPUT);
pinMode(GREEN_LED_PIN, OUTPUT);
pinMode(RED_LED_PIN, OUTPUT);
pinMode(BUZZER_PIN, OUTPUT);

// Connect to WiFi and Blynk
WiFi.begin(ssid, pass);
Blynk.begin(auth, ssid, pass);

// Initialize SIM800L module
sim800lSerial.begin(9600);
delay(1000);
```

```
sim800lSerial.println("AT+CMGF=1");
delay(100);

sim800lSerial.println("AT+CNMI=2,2,0,0,0");
delay(100);
}

void loop() {
// Read the gas sensor value
gasValue = map(analogRead(32), 0, 4095, 0, 1000); //Mapping the 4095 to 1000 on gauge

// Send gas reading to Blynk
blynk_send_reading();

// Print gas value to serial monitor
Serial.println(gasValue);

// Check if gas value exceeds threshold
if (gasValue > gasThreshold) {
Serial.println("Gas exceeded the threshold!");

// Turn off green LED, turn on red LED, and start buzzing
digitalWrite(GREEN_LED_PIN, LOW);
digitalWrite(RED_LED_PIN, HIGH);
digitalWrite(BUZZER_PIN, HIGH);

// Send blynk notification
Blynk.logEvent("gas_alert");

// Send SMS message
sendSMS("+962799176197", "Gas Leakage Detected!");
}
else {
Serial.println("No gas leakage.");

// Turn on green LED, turn off red LED, and stop buzzing
```

```
digitalWrite(GREEN_LED_PIN, HIGH);
digitalWrite(RED_LED_PIN, LOW);
digitalWrite(BUZZER_PIN, LOW);
}
// Run Blynk tasks
Blynk.run();
}
```