IoT BASED GAS LEAKAGE DETECTION

A

Minor Project

Report submitted in partial fulfilment of the requirement for the

Award of Degree

of

Bachelor of Technology

In

Electronics and Communication Engineering

Submitted to

Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, M.P.

Prepared and Submitted by

Sahil Soni (0187EC211040)

Arpit Shrivastava (0187EC211006)

Riya Shrivastava (0187EC211036)

Gourav Sahita (0187EC211014)

Under the Supervision of

(Dr. Sachin Kalraiya)

DEPARTMENT OF ELECTRONICS & COMMUNICATION

Sagar Institute of Science and Technology, Bhopal

Session 2023-24

SAGAR INSTITUTE OF SCIENCE & TECHNOLOGY, BHOPAL (M.P.)

Department of Electronics & Communication Engineering

CERTIFICATE

This is to certify that the work embodies in this minor project entitled **'IoT BASED GAS LEAKAGE DETECTION'** being submitted by Arpit Shrivastava (0187EC211006), Riya Shrivastava (0187EC211036), Gourav Sahita (0187EC211014), and Sahil Soni (0187EC211040), 'Bachelor of Engineering in Electronics and Communication Engineering' discipline to Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal (M.P.) during the academic year 2021-25 is a record of a bona fide piece of work.

Approved and Supervised by

Prof. (Pradeep Singh Yadav)

Prof. (Sachin Kalraiya)

Assistant Professor, EC

Forwarded by

Dr. Ravi Shankar Mishra
HoD, ECE Dept.
SISTec, Bhopal

Dr. Dinesh Kumar Rajoriya
Principal,
SISTec, Bhopal

SAGAR INSTITUTE OF SCIENCE & TECHNOLOGY, BHOPAL (M.P.)

Department of Electronics & Communication Engineering

DECLARATION

We Arpit Shrivastava (0187EC211006), Riya Shrivastava (0187EC211036), Gourav Sahita (0187EC211014), and Sahil Soni (0187EC211040), students of 'Bachelor of Technology in Electronics & Communication Engineering', Session: 2023-24, Sagar Institute of Science & Technology, Gandhi Nagar, Bhopal (M.P.), at this moment declare that the work presented in this project report entitled **'IoT BASED GAS LEAKAGE DETECTION'** is the outcome of our bona fide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics. It contains no material previously published or written by another person nor material that has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Arpit Shrivastava (0187EC211006)
Riya Shrivastava (0187EC211036)
Gourav Sahita (0187EC211014)
Sahil Soni (0187EC211040)

SAGAR INSTITUTE OF SCIENCE & TECHNOLOGY, BHOPAL (M.P.)

Department of Electronics & Communication Engineering

CERTIFICATE OF APPROVAL

The project report entitled **'IoT BASED GAS LEAKAGE DETECTION'** being submitted by Arpit Shrivastava (0187EC211006), Riya Shrivastava (0187EC211036), Gourav Sahita (0187EC211014), and Sahil Soni (0187EC211040), has been examined by us and is now approved for the award of 'Bachelor of Technology in Electronics & Communication Engineering' for which it has been submitted. It is understood that by this approval the undersigned do not necessarily endorse or approve any statement made, opinion expressed or conclusion drawn therein, but approve the project report only for the purpose for which it has been submitted.

(Internal Examiner)
Name:
Designation:
Address: ECE Dept.
SISTec, Bhopal

(External Examiner)
Name:
Designation:
Address:

ACKNOWLEDGEMENT

We owe an enormous debt of gratitude to my project supervisors, **Prof. (Sachin Kalraiya)**, for guiding and inspiring me from the beginning through the end of this thesis with his intellectual pieces of advice and insightful suggestions. We truly appreciate and value his consistent feedback on my progress, which was always constructive and encouraging, and ultimately drove us in the right direction. We also owe a lot of thanks to several people who have helped and motivated us throughout our project work as well as throughout our graduation at SISTec, Bhopal, in particular**, Dr. Dinesh Kumar Rajoriya (Principal, SISTec, Bhopal), Dr. Ravi Shankar Mishra, (Prof. & HOD of ECE Dept., SISTec, Bhopal),** all teaching and non-teaching staff of ECE department for their valuable assistance they offered us generously during the past four years. We wish to thank all our family members and our friends for their unwavering faith and belief in us throughout our life.

Arpit Shrivastava (0187EC211006)
Riya Shrivastava (0187EC211036)
Gourav Sahita (0187EC211014)
Sahil Soni (0187EC211040)

ABSTRACT

The increasing concern for environmental safety and human health has led to a growing demand for efficient gas leakage detection systems. In this context, our project focuses on the development of an Internet of Things (IoT) based Gas Leakage Detection System utilizing the ESP8266 Wi-Fi module and the MQ-6 gas sensor.

The core objective of our project is to create a reliable and responsive system capable of detecting potentially hazardous gas leaks in real-time, thereby mitigating the risks associated with gas-related incidents. The ESP8266 module facilitates seamless communication between the gas sensor and the cloud, enabling remote monitoring and alert mechanisms.

The MQ-6 gas sensor, which is sensitive to various combustible gases such as LPG, propane, and methane, serves as the primary sensing component. When a gas leak is detected, the sensor triggers the ESP8266 module to send immediate alerts to predefined users through the cloud platform. This ensures timely response and intervention, reducing the probability of accidents and damage.

The proposed system not only enhances safety but also provides an efficient means of monitoring gas levels in different environments. The IoT architecture allows users to access real-time data through a user-friendly interface, providing insights into gas concentrations over time. Additionally, the system can be integrated with home automation systems for automatic shutdown of gas supply in case of a detected leak.

By combining IoT technology with gas sensing capabilities, our project aims to contribute to a safer living and working environment, emphasizing the importance of proactive gas leak detection and prevention. The low-cost and energy-efficient nature of ESP8266, coupled with the sensitivity and specificity of the MQ-6 sensor, make this system a practical and scalable solution for a wide range of applications, from residential settings to industrial facilities.

LIST OF FIGURES

TABLE OF CONTENTS

CONTENTS **Page Number**

CHAPTER 1- INTRODUCTION

1. HARDWARE AND SOFTWARE

1.1 HARDWARE USE

1.1.1 ESP8266

Description

The ESP8266 is a popular and versatile Wi-Fi module and microcontroller, widely used in the field of Internet of Things (IoT) and embedded systems. Developed by Espressif Systems, the ESP8266 has gained immense popularity due to its low cost, small form factor, and robust performance. It is particularly known for its capability to provide Wi-Fi connectivity to microcontroller-based projects, making it an ideal choice for IoT applications.

Features

- **Integrated Wi-Fi**: The ESP8266 comes with built-in Wi-Fi functionality, enabling seamless connectivity to wireless networks. This feature is crucial for IoT projects requiring remote communication.

- **Microcontroller Functionality**: Despite its primary role as a Wi-Fi module, the ESP8266 also houses a microcontroller unit (MCU) with GPIO (General Purpose Input / Output) pins. This allows it to execute standalone applications without the need for an external microcontroller.

- **Low Cost**: One of the most significant advantages of the ESP8266 is its affordability. This has contributed to its widespread adoption in both hobbyist and professional projects.

- **Low Power Consumption**: The ESP8266 is designed to operate with low power consumption, making it suitable for battery-powered and energy-efficient applications.

- **Community Support**: The ESP8266 has a large and active community of developers and enthusiasts. This community support provides a wealth of resources, including tutorials, forums, and open-source projects, making it easier for users to implement and troubleshoot their projects.

- **Firmware Over-the-Air (OTA) Updates**: The ESP8266 supports firmware updates over Wi-Fi, allowing users to update their devices remotely without physical access.

- **Multiple Operating Modes**: It can operate in various modes, such as Station mode (connecting to an existing Wi-Fi network), Access Point mode (creating its own Wi-Fi network), and both modes simultaneously.

Specifications

- Processor: Tensilica L106 32-bit microcontroller
- Clock Speed: 80 MHz
- Memory: 32-bit RISC CPU: 80 MHz, 64 KB instruction RAM, 96 KB data RAM
- External QSPI flash: up to 16 MB
- Wi-Fi: IEEE 802.11 b/g/n protocol, Station, SoftAP, and SoftAP+ Station modes
- GPIO: 17 GPIO pins available for various digital and analog functions GPIOs support PWM, I2C, SPI, and UART
- Interfaces: UART, I2C, I2S, SPI, GPIO, PWM
- Operating Voltage: 3.0V - 3.6V
- Power Consumption: Varies based on usage, but optimized for low power applications
- Dimensions: Compact form factor, typically in the range of 14mm x 24mm
- Operating Temperature Range: -40°C to 125°C.

PIN Specifications

- TXD (GPIO1): Transmit Data pin for UART communication.
- RXD (GPIO3): Receive Data pin for UART communication.
- GPIO0: General Purpose Input /Output pin, used for mode selection during boot (flash mode or normal mode).
- GPIO2: General Purpose Input /Output pin.
- EN (CH_PD): Enable pin, keeps the module enabled when pulled high.
- RESET (RST): Reset pin, used to restart the module.
- 3.3V: Power supply pin (3.3V).
- GND: Ground.

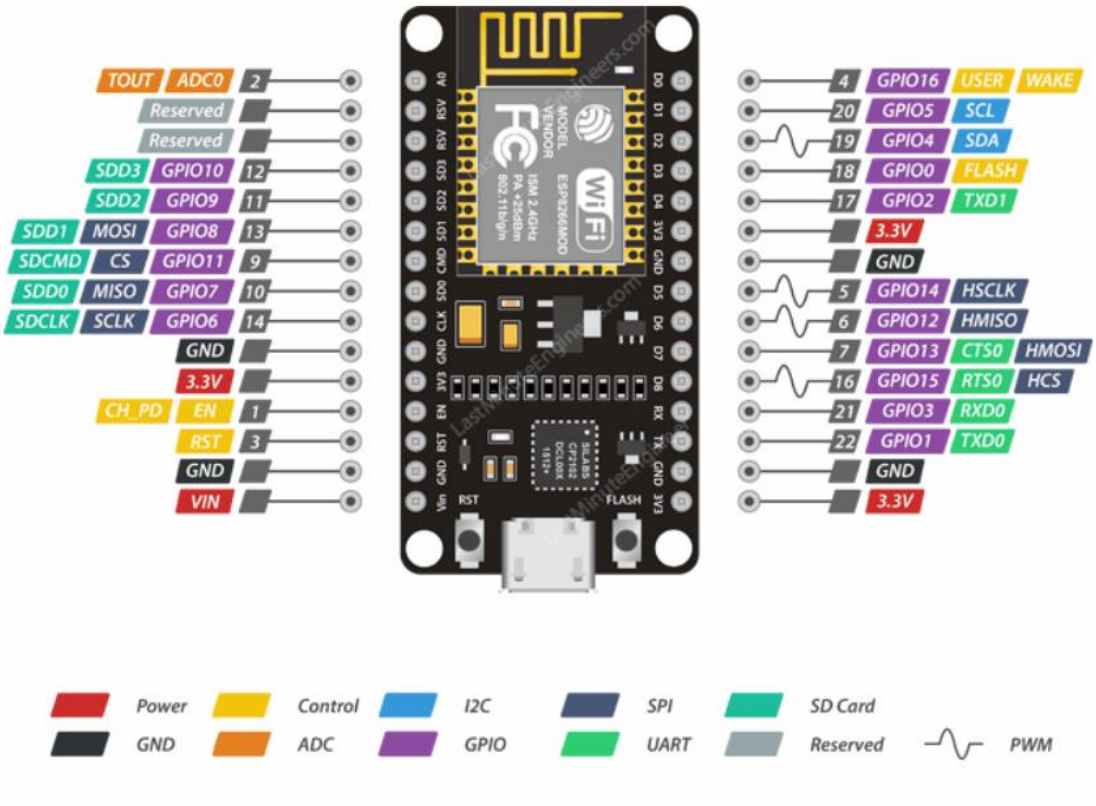

Figure 1.1.1

ESP8266

1.1.2 MQ6 Gas Sensor

Description

The MQ6 sensor operates on the principle of resistance changes in the presence of specific gases. It consists of a sensitive semiconductor element that undergoes a change in resistance when exposed to the target gas. The resistance change is proportional to the gas concentration in the air, allowing for the detection and measurement of various combustible gases.

Features

- **Wide Detection Range**: The MQ6 sensor is capable of detecting a broad range of combustible gases, including LPG (liquefied petroleum gas), propane, methane, butane, and other hydrocarbons.
- **High Sensitivity**: It exhibits high sensitivity to low concentrations of combustible gases, making it suitable for early detection of potential gas leaks.
- **Quick Response Time**: The sensor provides a rapid response to changes in gas concentration, enabling real-time monitoring and timely alerts.
- **Simple Interface**: The MQ6 sensor is relatively easy to interface with microcontrollers or microprocessors, making it a popular choice for IoT projects and gas detection systems.
- **Low Power Consumption**: It is designed with low power consumption, allowing for extended operation without draining significant amounts of power.
- **Stability and Longevity**: The sensor demonstrates stable performance over time, and when used within specified conditions, it can have a relatively long operational life.
- **Compact Design**: The MQ6 sensor is compact and lightweight, facilitating easy integration into various electronic systems and devices.

Specifications

- **Operating Voltage:** Typically operates at a voltage between 4.0V and 6.0V DC.
- **Heating Element Voltage:** The sensor includes an internal heating element that requires a specific voltage (e.g., 5V) for proper operation.
- **Preheat Time:** It requires a certain preheat time to stabilize and achieve accurate readings. This time is usually in the range of a few minutes.
- **Detection Range:** The MQ6 sensor can detect concentrations of combustible gases in the air within a specified range, often given in parts per million (ppm) or percentage of the lower explosive limit (LEL).
- **Output Signal:** The output of the MQ6 sensor is an analog voltage signal that varies with changes in gas concentration.
- **Operating Temperature Range:** The sensor is designed to operate within a specific temperature range, and it may not perform optimally outside of this range.
- **Dimensions:** Physical dimensions of the MQ6 sensor can vary, but it is generally a compact module suitable for integration into electronic systems.

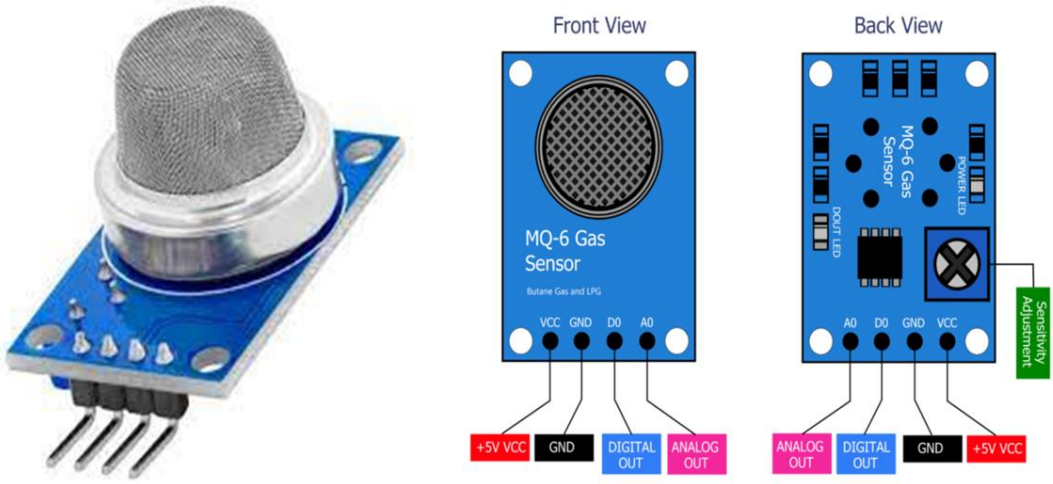

Figure 1.1.2

MQ6 Sensor

1.1.3 BZ-Buzzer

Description:

A buzzer is an electromechanical component designed to produce audible sound or tones when an electric current is applied. It is commonly used in electronic projects, alarms, and other applications where an audible alert is required.

Features:

1. **Audible Alert:** Buzzer produces a loud and distinctive sound, serving as an effective audible alert.
2. **Compact Size:** Buzzer components are typically compact, making them suitable for various electronic devices and applications.
3. **Simple Design:** **Buzzer design is straightforward, usually consisting of a coil, diaphragm, and contacts, allowing for easy integration into circuits.**
4. **Low Power Consumption:** **Buzzer components generally have low power requirements, making them energy-efficient for battery-operated devices.**
5. **Versatility:** **Available in various types, including piezoelectric and electromagnetic buzzers, offering versatility in design and functionality.**
6. **Easy to Interface:** **Buzzer components can be easily interfaced with microcontrollers and other electronic circuits using simple wiring.**

Specifications:

1. **Operating Voltage:** **Typically operates within a specified voltage range, such as 3V to 12V, depending on the type and model.**
2. **Sound Output:** **Measured in decibels (dB), indicating the loudness of the sound produced. Common buzzers have sound outputs ranging from 70dB to 110dB.**
3. **Frequency:** **Specifies the frequency at which the buzzer produces sound, often ranging from a few hundred hertz to several kilohertz.**
4. **Type:** **Different types include piezoelectric buzzers and electromagnetic buzzers, each with its own characteristics and applications.**

5. **Mounting:** Available in surface-mount (SMD) and through-hole mounting options, allowing for flexibility in circuit design.

Pin Specifications:

1. **Positive (+) or VCC:**
    - Connects to the positive terminal of the power supply, providing the necessary voltage for the buzzer to operate.
2. **Negative (-) or GND:**
    - Connects to the ground (0V) terminal of the power supply, completing the circuit.
3. **Signal (S) or Control:**
    - Connects to the signal or control pin of the microcontroller or circuit, enabling the buzzer to be activated or controlled.

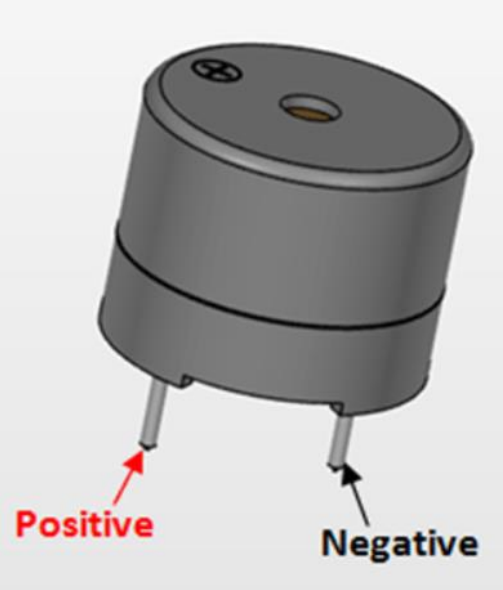

Figure 1.1.3

BZ-Buzzer

1.2 SOFTWARE USE

1.2.1 Arduino IDE

Description:

The Arduino IDE is a software development environment that is specifically designed for writing, compiling, and uploading code to Arduino microcontrollers. It provides a user-friendly interface, making it accessible for both beginners and experienced developers. The IDE simplifies the process of programming Arduino boards, enabling users to create a wide range of projects, from simple LED blinkers to complex IoT applications.

Features:

- **Cross-Platform Compatibility**: Arduino IDE is compatible with Windows, macOS, and Linux operating systems, allowing users to develop code on their preferred platform.
- **Code Editor**: The IDE features a code editor with syntax highlighting, auto-completion, and error checking, making it easier for users to write and debug their Arduino code.
- **Library Manager**: Arduino IDE includes a library manager that simplifies the process of adding, managing, and updating libraries, enabling users to easily incorporate pre-written code into their projects.
- **Serial Monitor**: The built-in Serial Monitor allows users to communicate with their Arduino boards in real-time, facilitating debugging and the monitoring of sensor data.
- **Board Manager:** Arduino IDE supports a variety of Arduino boards and compatible microcontrollers. The Board Manager simplifies the process of adding new board profiles and ensuring compatibility.
- **Integrated Examples:** The IDE comes with a variety of example sketches that serve as starting points for different types of projects, helping users understand and implement various functionalities.
- **Easy Firmware Updates:** Users can easily update the firmware on their Arduino boards using the IDE, ensuring access to the latest features and improvements.
- **Open-Source**: Arduino IDE is open-source software, allowing the community to contribute to its development and customize it according to their needs.
- **Extensibility:** The IDE supports the use of third-party plugins and extensions, expanding its capabilities and allowing developers to tailor the environment to their specific requirements.

- **Project Sketches and Libraries Management**: Arduino IDE helps manage project sketches and libraries, providing an organized workspace for developers to work on multiple projects seamlessly.

Specifications:

- **Programming Language**: Arduino IDE uses a simplified version of C++ for programming Arduino boards.
- **File Formats**: Arduino IDE saves projects in the form of sketches, which consist of a primary '.ino' file along with other supporting files and folders.
- **Compiler:** The IDE uses the AVR-GCC compiler to convert the written code into machine-readable instructions for the Arduino microcontroller.
- **Upload Protocol**: Arduino IDE uses a boot-loader and a specific upload protocol to transfer compiled code from the computer to the Arduino board via a USB connection.
- **Version Control**: Arduino IDE supports basic version control features, allowing users to save and revert to previous versions of their sketches.

User Interface:

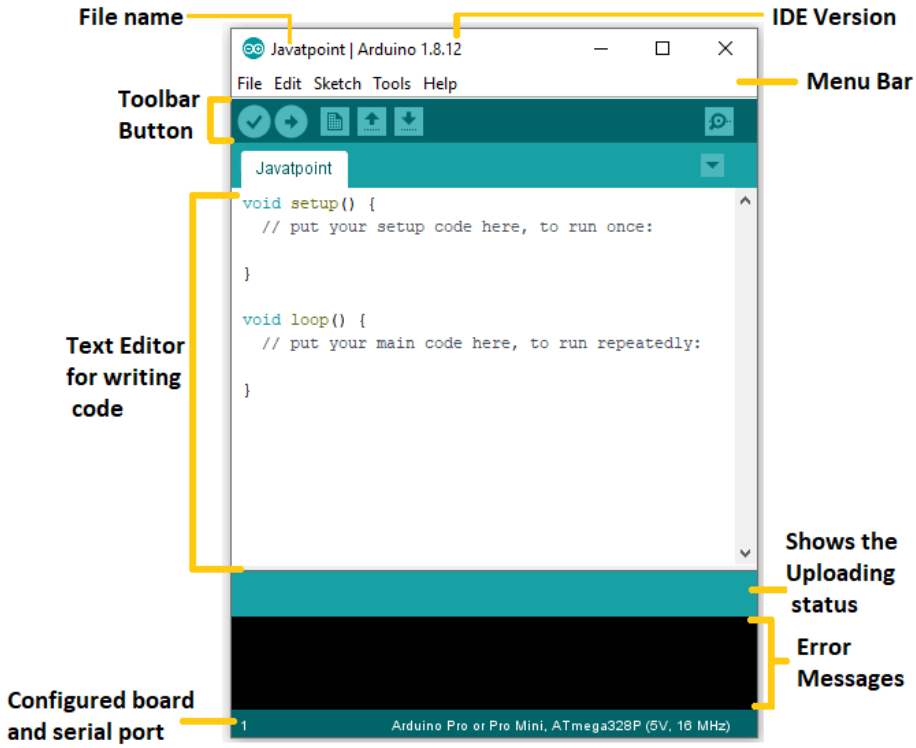

Figure 1.1.4

Arduino IDE UI

1.2.2 IFTTT

Description:

IFTTT, which stands for "If This Then That," is a powerful web-based service that allows users to create automated connections between different applications, devices, and online services. The platform operates on the principle of conditional statements, where a trigger event in one service results in an action in another. Users can create simple automation rules, known as applets, to streamline their digital workflows and connect various online services seamlessly.

Features:

- **Applets**: IFTTT operates based on applets, which are simple conditional statements that link a trigger (If This) to an action (Then That).
  Users can create custom applets or choose from a library of pre-existing ones tailored for popular services and devices.
- **Wide Range of Integrations:** IFTTT supports a vast array of third-party services, apps, and devices, ranging from social media platforms and smart home devices to productivity tools and IoT devices.
- **User-Friendly Interface:** The platform provides an intuitive and user-friendly interface, allowing individuals with varying technical expertise to create and manage applets effortlessly.
- **Cross-Platform Compatibility**: IFTTT is compatible with major platforms, including iOS, Android, web browsers, and various IoT ecosystems, making it versatile for users across different devices.
- **Automation Triggers:** IFTTT supports a diverse set of triggers, such as receiving an email, a change in weather conditions, a new post on social media, or an event in a calendar, providing flexibility in automation possibilities.
- **Multi-Step Applets**: Users can create complex automations involving multiple steps or actions, allowing for more sophisticated and personalized workflows.
- **IoT Device Integration**: IFTTT plays a significant role in the Internet of Things (IoT) ecosystem, enabling users to connect and automate smart home devices, wearables, and other IoT-enabled gadgets.
- **Conditional Logic:** Users can implement conditional logic in their applets, introducing more complexity by specifying additional criteria for triggering actions.

Specifications:

- **Channels:** IFTTT connects to different services and devices through channels. Each channel represents a specific platform or application that users can integrate into their applets.
- **Triggers and Actions:** Triggers are events that initiate an applet, and actions are the resulting tasks performed when the trigger conditions are met.
- **Platform Compatibility:** IFTTT is accessible through web browsers and has dedicated applications for both iOS and Android platforms, ensuring compatibility across various devices.
- **Customization:** Users can customize their applets to suit their specific needs, tailoring the automation process according to personal preferences and requirements.
- **Community Support:** IFTTT boasts an active and engaged community, with users often sharing and contributing new applets, expanding the available automation options.

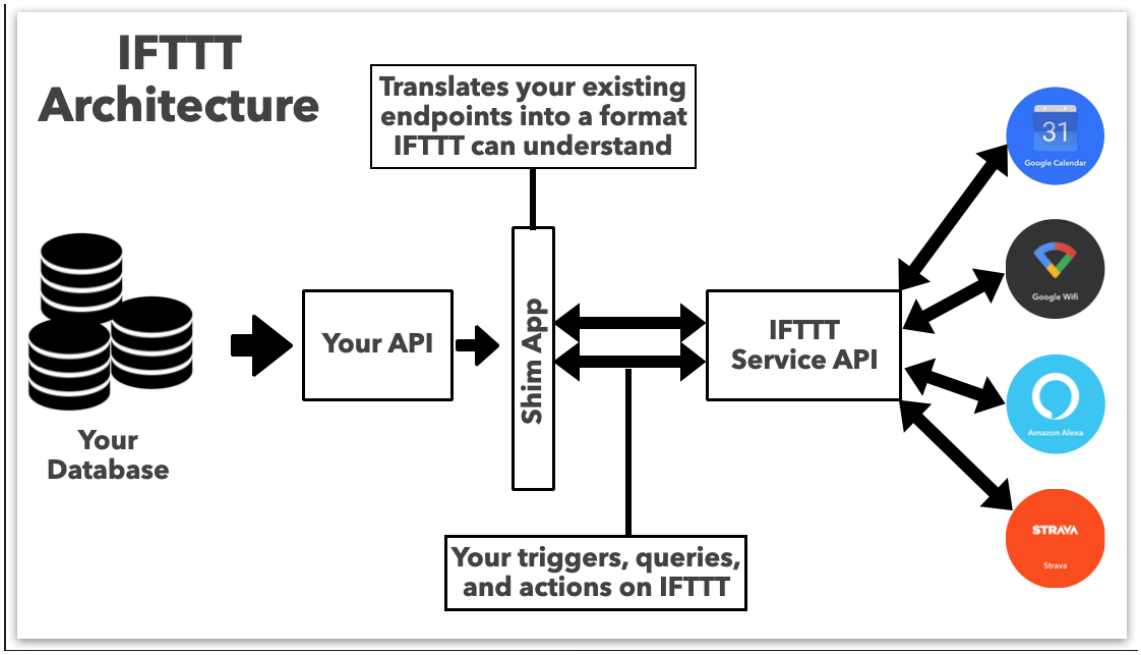

Figure 1.1.5

IFTTT Architecture

1.3 PCB DESIGN:

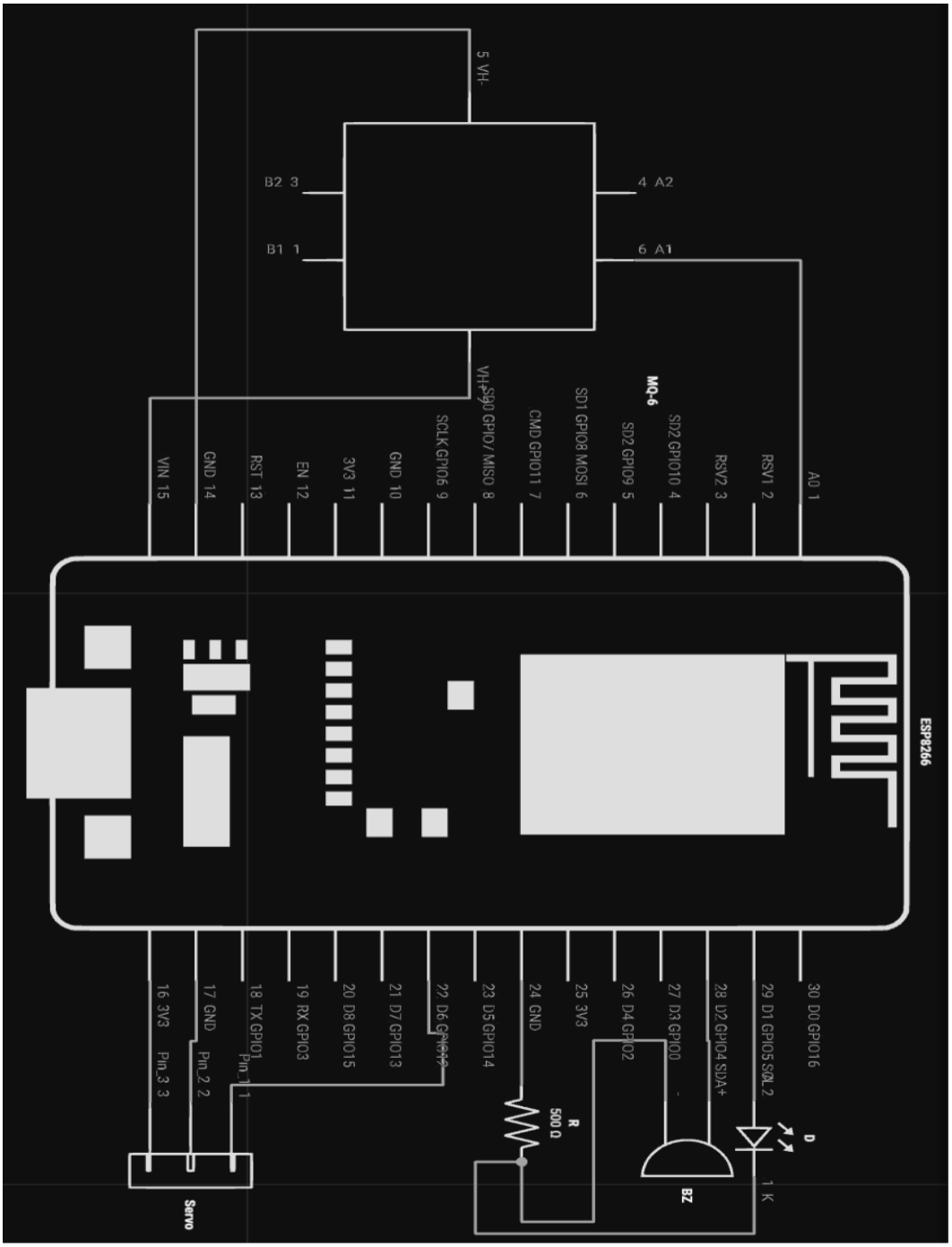

Figure 1.1.6

PCB Scheme

1.4 FINAL CODE:

```
#include <ESP8266WiFi.h>

const char* ssid = "JioFiber-AQEm9";  //Enter the Wi-Fi ID
const char* password = "Adarsh@123"; //Enter the Password
const char* host = "maker.ifttt.com";

const int gasSensorPin = A0;  // Analog pin for MQ-6 gas sensor
const int buzzerPin = D2;
const int ledpin = D1;

void setup() {
  Serial.begin(115200);
//   LED
  pinMode(D1, OUTPUT);
//   ...
//Bugger
  pinMode(buzzerPin, OUTPUT);
//.......

  Serial.println("Email from Node Mcu");
  delay(1000);

  Serial.print("Connecting to ");
  Serial.println(ssid);
```

```
  WiFi.begin(ssid, password);

  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }

  Serial.println("");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}

void loop() {
  int gasValue = analogRead(gasSensorPin);
  int gasThreshold = 300;
  Serial.println(gasValue);

  if (gasValue > gasThreshold) {
    digitalWrite(D1, HIGH);
    digitalWrite(buzzerPin, HIGH);
    sendGasDetectedEmail();
    delay(300000);  // Delay for 5 minutes to avoid sending multiple emails in a short period
  }
}

void sendGasDetectedEmail() {
  WiFiClient client;
```

```
  const int httpPort = 80;

  if (!client.connect(host, httpPort)) {
    Serial.println("Connection failed");
    return;
  }

  String url = "https://maker.ifttt.com/trigger/coco/with/key/bpSUyI-fz2MRKu9n4_A5i2";

  Serial.print("Requesting URL: ");
  Serial.println(url);

  client.print(String("GET ") + url + " HTTP/1.1\r\n" +
               "Host: " + host + "\r\n" +
               "Connection: close\r\n\r\n");

  Serial.println("Gas detected! Email notification sent.");
  delay(5000);
  digitalWrite(D1, LOW);
  digitalWrite(buzzerPin, LOW);
  delay(1000);
}
```

2. OVERVIEW

In recent years, there has been a growing concern about gas leaks in various environments, including homes, industries, and commercial spaces. Gas leaks can lead to severe consequences such as fire, explosions, and adverse health effects. To address this issue, the integration of Internet of Things (IoT) technology with gas sensors presents a promising solution. This project focuses on developing a gas leakage detection system using the ESP8266 microcontroller and the MQ6 gas sensor.

3. OBJECTIVES

- **Early Detection:** Design a system that can detect gas leaks at an early stage, preventing potential hazards.
- **Real-time Monitoring:** Implement real-time monitoring capabilities to enable quick response and intervention in case of a gas leak.
- **Remote Notification:** Integrate IoT capabilities to send instant notifications to users or authorities through email, SMS, or a mobile app.
- **Data Logging:** Create a system that logs gas concentration data over time, allowing for analysis and historical tracking of gas levels.
- **Cost-Effective Solution:** Develop a system that is affordable and can be easily implemented in both residential and industrial settings.

4. TECHNOLOGY MILESTONES:

1. **ESP8266 Microcontroller:** Utilize the ESP8266, a low-cost Wi-Fi module with built-in TCP/IP stack, to enable wireless communication and connectivity to the Internet.
2. **MQ6 Gas Sensor:** Incorporate the MQ6 gas sensor, known for its high sensitivity to various gases, including LPG, propane, methane, and butane.
3. **IoT Connectivity:** Implement MQTT (Message Queuing Telemetry Transport) protocol for efficient communication between the gas detection system and an IoT platform.
4. **Cloud Integration:** Use cloud platforms like ThingSpeak, IFTTT or Blynk to store and visualize the gas concentration data in a user-friendly interface.
5. **Mobile Application:** Develop a mobile application to provide users with a convenient interface for monitoring gas levels and receiving real-time alerts.
6. **Power Management:** Design power-efficient strategies to ensure prolonged operation of the gas detection system, possibly incorporating sleep modes for the ESP8266.

CHAPTER-2 LITERATURE SURVEY

Gas Leakage Detection System Using IoT and Cloud Technology: A Review

June 2023, E3S Web of Conferences 391

V Praveen Sharma (Affiliation: Department of Computer Science and Engineering, Chaitanya Bharathi Institute of Technology, Hyderabad-500075)
Dr. Raman Dugyala (Affiliation: Department of Computer Science and Engineering, Chaitanya Bharathi Institute of Technology, Hyderabad-500075)
Dr. V Padmavathi (Affiliation: Department of Computer Science and Engineering, Chaitanya Bharathi Institute of Technology, Hyderabad-500075)
Vijendar Reddy Gurram (Affiliation: Department of Information and Technology, Gokaraju Rangaraju Institute of Engineering and Technology, Hyderabad-500090

The suggested model is made up of sensor devices that detect environmental values like Voltage, as well as the current characteristics of the many household appliances are used to calculate power usage. The IoT platform Xively gives channel utility to deploy the prototype into an integrated solution and provides the framework for connecting the smart sensor to the internet [9]. This project aimed to prevent industrial mishaps, monitor hazardous chemicals, and communicate alert messages to the industry's safety control board. The central board is an Ardunio Uno R3 board. a microcontroller that is linked to a sensor Including sensors that can continuously monitor their respective environmental conditions, such temperature, gas sensors, and alcohol sensors. Data from the sensor is stored online so that it can be processed further and evaluated to increase safety.

CHAPTER-3 METHODOLOGY

- **Hardware Setup**: Connect the MQ6 gas sensor to the ESP8266 microcontroller and calibrate the sensor for accurate gas concentration measurements.
- **Programming**: Code the ESP8266 to read data from the MQ6 sensor, establish a Wi-Fi connection, and transmit data to the chosen IoT platform.
- **IoT Cloud Configuration**: Set up an account on the selected cloud platform, configure the IoT device, and design the data visualization interface.
- **Website Development**: Develop a web application that can communicate with the gas detection system, displaying real-time data and sending alerts.
- **Testing and Calibration:** Conduct extensive testing to ensure the system's reliability, accuracy, and responsiveness to gas leaks. Calibrate the sensor as needed.
- **Documentation:** Create comprehensive documentation, including circuit diagrams, source code, and a user manual for future reference and replication.

CHAPTER-4 CONCLUSION

The proposed IoT-based gas leakage detection system aims to enhance safety by providing an early warning system and real-time monitoring capabilities. By combining the power of the ESP8266 microcontroller, the MQ6 gas sensor, and IoT technology, this project seeks to create a cost-effective and efficient solution for gas detection in various environments.

In conclusion, the IoT-based gas leakage detection project represents a significant advancement in ensuring the safety and security of both residential and industrial environments. The integration of Internet of Things (IoT) technology with gas detection systems has proven to be a proactive and efficient approach in monitoring and preventing potential hazards associated with gas leaks.

Through the deployment of smart sensors and a robust communication infrastructure, the system enables real-time monitoring of gas levels and immediate detection of any anomalies. This not only enhances the overall safety measures but also provides a swift response mechanism to mitigate the risks associated with gas leaks.

The project's success lies in its ability to offer a seamless and interconnected solution that transcends traditional gas detection methods. With the power of IoT, users can access and control the system remotely, receive instant alerts, and gather comprehensive data for analysis. This not only empowers individuals and organizations to take timely actions but also allows for predictive maintenance, minimizing downtime and potential damages.

Furthermore, the scalability and adaptability of the IoT-based gas leakage detection system make it suitable for various applications, ranging from homes and commercial establishments to industrial facilities. The integration of machine learning algorithms and data analytics further enhances the system's capabilities, enabling it to learn from historical data and continuously improve its performance.

In essence, the IoT-based gas leakage detection project is a testament to the transformative potential of technology in safeguarding lives and property. As we move towards a more connected and automated future, this project exemplifies the positive impact that innovative solutions can have on addressing critical safety concerns. With ongoing advancements in IoT and related technologies, the possibilities for improving gas detection systems are boundless, promising a safer and more secure environment for individuals and communities.

CHAPTER-5 REFERENCES

1. lOT Based Industrial Plant Safety Gas Leakage Detection System

By ieee.org

Ravi Kishore Kodali

Department of ECE, National Institute of Technology, Warangal, WARANGAL, INDIA

R.N.V. Greeshma

Department of ECE, National Institute of Technology, Warangal, WARANGAL, INDIA

Kusuma Priya Nimmanapalli

Department of ECE, National Institute of Technology, Warangal, WARANGAL, INDIA

Yatish Krishna Yogi Borra

Department of ECE, National Institute of Technology, TADEPALLEGUDEM,

Andhra Pradesh, INDIA

2. IOT Based Smart Gas Leakage Detection and Alert System

Proceedings of the 4th International Conference on Advances in Science & Technology (IAST2021)

4 Pages Posted: 15 Jun 2021

Sejal Shah

K J Somaiya Institute of Engineering and Information Technology, Mumbai,

Aakash Parashar

K J Somaiya Institute of Engineering and Information Technology, Mumbai,

Chanchal Rai

K J Somaiya Institute of Engineering and Information Technology, Mumbai,

Sagar Pokhariyal

K J Somaiya Institute of Engineering and Information Technology, Mumbai

3. Gas Leakage Detection and Alert System using IoT

Sayali Joshi,
Shital Munjal,
Prof. Uma B. Karanje

Computer Engineering, Marathwada Mitra Mandal Institute of Technology, Lohgaon, Pune, Maharashtra, India

4. International Research Journal of Engineering and Technology (IRJET)

e-ISSN: 2395-0056

A GAS LEAKAGE DETECTOR USING IOT

Dr. C K Gomathy[1], Mr. S. Megha Shyam Raju[2], Mr. K. Sheshank Reddy[3]