ABSTRACT

A gas sensor is a device that detects the presence of gases in an area, often as part of a safety system. This type of equipment is used to detect a gas leak or other emissions. A gas detector can sound an alarm to operators in the area where the leak is occurring, giving them the opportunity to leave. This type of device is important because there are many gases that can be harmful to organic life, such as humans or animals. Gas detectors can be used to detect combustible, flammable and toxic gases, and oxygen depletion.

Liquefied Petroleum Gas (LPG) is a main source of fuel, especially in urban areas because it is clean compared to firewood and charcoal. Gas leakage is a major problem in the industrial sector, residential premises, etc. Nowadays, home security has become a major issue because of increasing gas leakage. Gas leakage is a source of great anxiety with ateliers, residential areas and vehicles like Compressed Natural Gas (CNG), buses, and cars which are run on gas power. One of the preventive methods to stop accidents associated with the gas leakage is to install a gas leakage detection kit at vulnerable places.

The presence of hazardous LPG gas leakage in a domestic, work place, also stored gases container gas which exhibits ideal characteristics. For that sake, Gas leakage detection system is used. Here, an alarm unit is used to vibrate an alarm which is buzzer. Buzzer gives an audible sign of the presence of LPG volume. The sensors are widely used to detect essence of propane, Iso-butane, LPG and even smoke. The sensor has an advantage to combine a sensitivity response time. If the Gas sensor senses gas leak from work place or home, sensor output goes to active low (logic-0) condition. Arduino NANO is used in the project; low signals are overlooked by the Arduino and gas leakage is been noticed by the Arduino. The Arduino NANO turns on the LCD and buzzer. It even turns on the GSM modem after that, it continues to send messages SMS to mobile number specifically mentioned in the program of the source code for alerting danger to the people.

CHAPTER 1

INTRODUCTION

1.1 General Introduction

The Internet of Things (IoT) is a technology where the things are embedded which is connected over the internet like software and the internet enabled sensors which is collecting the data in the cloud. The data can be retrieved when it is required. The data is retrieved for the requested data only where, as other data will be stored in the cloud which is hidden.

Seamless connection between people, processes, and things is now feasible because to the capacity to link common goods such as kitchen appliances, automobiles, thermostats, and baby monitors to the internet via embedded devices. Physical objects may trade and capture each other. The characteristic of the IoT are connectivity which is internet connectivity over the hardware to the system controls; things which is any object which is connected over the internet; data which is collected from the sensors or any electronic devices; communication which the device is communicating with the data and analyse the data generated; intelligence which is based on the analysis of the data and the responding capability for the generated data for further process, action which linked to the intelligence where the manual actions may or may not be required and lastly, ecosystem which is important characteristics where the system or application should be designed based on the effective results in environment where there is harm from the application.

The gas detection Operators in the area where the leak is occurring can be alerted by a gas detector, giving them the chance to evacuate. Because many gases may be hazardous to biological life, such as people or animals, this sort of equipment is essential. Increased levels of these gases in the atmosphere will be highly hazardous. These gases may be combustible at particular temperatures and humidity levels, toxic once they exceed the required limits, or a contributing element in an area's air quality, creating issues like smog and impaired visibility, which can lead to significant injuries and even harm to people's health. Many civilizations have built-in fire- fighting systems.

1.2 Goal of the Project

The aim of this project is to propose and discuss a design of a gas leakage detection system that can automatically detect, alert and control gas leakage. This proposed system also includes an alerting system for the users. The system is based on a sensor that easily detects a gas leakage.

In open or closed situations, gas leakage may be dangerous and lethal. While traditional gas detection systems are quite accurate, they are unaware of a few key aspects in the area of warning people of a leak. As a result, we've built the application for both industry and the society which will detect the leakage of gas and also monitor the gas availability. Alerting techniques that include sending text messages to the relevant authorities as well as the ability to analyse sensor reading data. Nowadays, gas leakage and detection are major concerns in our daily lives. Carbon pollution is a major issue that must be addressed. LPG gas is very flammable, posing a risk to both people and property. To avoid such catastrophes, a significant amount of effort has gone into developing reliable systems for detecting gas leaks. Some leak detection techniques were created to cause the leak to be discovered since it is not always essential to know about the presence of a leak in order to take corrective action. Our major objective is to recommend a gas detection equipment that includes gas leakage detecting hardware to households in the area. This can monitor and warn about harmful chemicals in the air at workplaces such as factories, and it may also be used in households by alerting through an LCD display and sending a message to a registered phone number.

Keywords: LPG-Liquid Petroleum Gas, LCD-Liquid Crystal Display, GSM- Global System for Mobile Communications, MQ 2 sensor-Arduino, IoT-Internet of Things

CHAPTER 2

LITERATURE SURVEY

2.1 Study of similar work

The device which is used to detect the gas is already present the market which is widely used in many places like industries where there is plenty of chances of the explosion which may lead to massive destruction and the loss of man power; in homes, where the LPG gas used most widely in our daily necessity where it can detect the leakage of LPG gas; in cars, where most of the vehicles contains the cylinder and many more places. Dr.Walter Snelling was the first to introduce LPG gas in 1910. It's a blend of butane and commercial propane. It is very combustible, and numerous accidents occur as a result of LPG leaks. As a result, it is necessary to identify and prevent gas leakage.

Gas Detectors can be classified in a variety of ways. They're divided into categories based on the type of gas they detect, the technologies that power the sensor's output, and the components that impact the sensor's operation (semiconductors, oxidation, catalytic, photoionization, infrared, etc.). In our daily lives, we utilize a variety of gadgets for various purposes, and the majority of them have the ability to emit any type of gas or chemical when in operation in the air. In many scenarios, it is difficult for human to keep an eye on the levels of the concentration of the leaked gas or to detect whether there is leakage of gas or not. If there is any leakage in gas when there is no one around, it may cause explosion when there is even a spark or the surrounding will have the harmful gas which may lead to suffocation and will lead to have health issues in breathing. There are many applications for detection and monitoring of the leakage of gas, but still the researchers will make the efforts in making the advanced application where the cost of the application will be lesser.

2.1.1 Existing System

In existing system, with early detection of leaks on gas cylinders system using arduino based MQ-2 sensors. In the MQ-2 sensor-based LPG gas leak detection system using Arduino it has been successfully running according to an algorithm that has been designed and installed before in the test results. It only beeps the alarm once, if it detects gas leakage.

This system can detect gas faster in a closed room, on the contrary if in an open room the sensor can detect longer because the level of contaminated gas will be immediately wasted into the air. The most important thing in gas detection is having to inform the owner where the gas is placed by an alarm. The system is essentially useless if the owner is not there, since it is not safe.

2.1.2 Drawbacks of Existing System

This application system is only a prototype and has not yet been tested for accidental leakage of LPG gas cylinders in relevant agencies which can cause explosions and fires. In designing and making a prototype system, there are various kinds of weaknesses, both from the system planning and the equipment that has been made, it needs development including adding buzzer output that beeps continuously till the presence of leaked gas can alert the surrounding environment more efficiently.

CHAPTER 3

OVERALL DESCRIPTION

3.1 Proposed System

In the proposed system of gas detection system, the application contains both the monitoring and detection of the gases which are very harmful for the surrounding. Liquefied Petroleum Gas (LPG) is a main source of fuel, especially in urban areas because it is clean compared to firewood and charcoal. Gas leakage is a major problem in the industrial sector, residential premises, etc. Nowadays, home security has become a major issue because of increasing gas leakage. Gas leakage is a source of great anxiety with ateliers, residential areas and vehicles like Compressed Natural Gas (CNG), buses, and cars which are run on gas power. One of the preventive methods to stop accidents associated with the gas leakage is to install a gas leakage detection kit at vulnerable places.

In the detection of the gas, the sensor which is used to sense many gases is MQ 2 sensor. After the detection of leakage in the gas, the sensor sends the signal to the Arduino NANO for the further process where other hardware components are connected to each other. Through Arduino NANO, it sends the signal to the LCD display for displaying the alert message as LPG Detected, accordingly, the buzzer be on so that the surrounding people will the alerted. Even the Owner of the application will receive the message through GSM module.

3.2 Features of Proposed System

The presence of hazardous LPG gas leakage in a domestic, work place, also stored gases container gas which exhibits ideal characteristics. For that sake, an alarm unit is used to vibrate an alarm which is buzzer. Buzzer gives an audible sign of the presence of LPG volume. The sensors are widely used to detect essence of propane, iso-butane, LPG and even smoke. The sensor has an advantage to combine a sensitivity response time. If the LPG sensor senses gas leak from work place or home, sensor output goes to active low (logic-0) condition.

Arduino NANO is used in the project; low signals are overlooked by the Arduino and gas leakage is been noticed by the Arduino. The Arduino NANO turns on the LCD and buzzer. The buzzer beeps continuously till the presence of leaked gas went out. So, people didn’t miss the buzzer sound just after a single beep. It even turns on the GSM modem after that, it continues to send messages SMS to mobile number specifically mentioned in the program of the source code for alerting danger to the people.

- Circuit function is very simple and self-explanatory
- Arduino is forum for open-source electronics.
- It is used by both the hardware and the software.
- It is very cheap and affordable.
- It is efficient and portable that means we can place this circuit at any place easily.

3.3 Functions of Proposed System

In the monitoring system, the LPG cylinder is considered then the signal is sent to Arduino Nano which then proceeds the signal to the GSM modem which will send the SMS to the given contact number. It is easy to detect when we are out of that area, which will be easier to use.

The system which is proposed has the features like;

Gas leakage is sensed using MQ-2 sensor, when it is more than threshold value given which is interfaced by Arduino NANO.

The buzzer will be on to notify the surrounding people.

There will a display on the LCD for detection of the alert message like LPG gas detected!

Message will be sent to the given number in the code as alert message through GSM modem.

In the proposed system, there are two functions that is leakage of LPG gas detection —and the monitoring of the gas. The gas detection is done using MQ2 sensor. because it can detect hazardous gases including LPG gas. The proposed system will the used both for the industry as well as the household purpose.

3.4 Requirements Specification

System Analysis is the process of studying a procedure or business in order to identify its goals and purposes and create systems and procedures that will achieve them in an efficient way. System analysis relates closely to requirements analysis. Requirement specification simply means figuring out what to make before you make it. It determines what people need before you start developing a product for them. Requirement definition is the activity of translating the information gathered into a document that defines a set of requirements. These should

accurately reflect what the customer wants. It is an abstract description of the services that the system should provide and the constraints under the system must operate.

The requirements of specification of the proposed system are as follows:

1. Simple and effective
2. User friendly
3. Cost effective

Power:

The Arduino Nano can be powered via the Mini-B USB connection, 6-20V unregulated external power supply (pin 30), or 5V regulated external power supply (pin 27). The power source is automatically selected to the highest voltage source.

Memory:

The ATmega328 has 32 KB, (also with 2 KB used for the bootloader. The ATmega328 has 2 KB of SRAM and 1 KB of EEPROM.

Communication:

The Arduino Nano has a number of facilities for communicating with a computer, another Arduino, or other microcontrollers. The ATmega328 provide UART TTL (5V) serial communication, which is available on digital pins 0 (RX) and 1 (TX). An FTDI FT232RL on the board channels this serial communication over USB and the FTDI drivers (included with the Arduino software) provide a virtual com port to software on the computer. The Arduino software includes a serial monitor which allows simple textual data to be sent to and from the Arduino board. The RX and TX LEDs on the board will flash when data is being transmitted via the FTDI chip and USB connection to the computer (but not for serial communication on pins 0 and 1). A Software Serial library allows for serial communication on any of the Nano's digital pins. The ATmega328 also support I2C (TWI) and SPI communication. The Arduino software includes a Wire library to simplify use of the I2C bus. To use the SPI communication, please see ATmega328 datasheet.

Programming:

The Arduino Nano can be programmed with the Arduino software. Select "Arduino Duemilanove or Nano w/ ATmega328" from the Tools > Board menu (according to the microcontroller on your board). The ATmega328 on the Arduino Nano comes pre-burned with a bootloader that allows you to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol. You can also bypass the bootloader and program the microcontroller through the ICSP (In-Circuit Serial Programming) header using Arduino ISP or similar.

Automatic (Software) Reset:

Rather than requiring a physical press of the reset button before an upload, the Arduino Nano is designed in a way that allows it to be reset by software running on a connected computer. One of the hardware flow control lines (DTR) of the FT232RL is connected to the reset line of the ATmega328 via a 100 nano farad capacitor. When this line is asserted (taken low), the reset line drops long enough to reset the chip. The Arduino software uses this capability to allow you to upload code by simply pressing the upload button in the Arduino environment. When the Nano is connected to a computer, it resets each time a connection is made to it from software (via USB). The bootloader is running on the Nano. While it is programmed to ignore malformed data, it will intercept the first few bytes of data sent to the board after a connection is opened. If a sketch running on the board receives one-time configuration or other data when it first starts, make sure that the software with which it communicates waits a second after opening the connection and before sending this data.

3.5 Feasibility Analysis

An initial investigation culminates in a proposal that determine whether an ultimate system is feasible. When a proposed system is made and approved it initiates a feasibility study. The purpose of the feasibility study is to identify various candidate systems and evaluates whether they are feasible by considering technical, economical and operational feasibility and to recommend to best candidate system. The feasibility of such a program is listed in a simulated environment. Once all features are working property in a simulated environment, we can implement in a real platform. During product engineering, we consider following types of feasibility:

3.5.1 Technical Feasibility

Technical feasibility identifies whether the proposed system can be developed with the existing technologies and available hardware and software resources. As a part of the technical feasibility of the system , the following points are to be emphasized. Technical feasibility is frequently the most difficult area to assess at the stage of the product engineering process. It is essential that the process of analysis and definition be conducted in parallel with an assessment of technical feasibility. The considerations that are normally associated with technical feasibility are development risk, resource availability and technology.

3.5.2 Operational Feasibility

Proposed projects are beneficial only if they can be turned into information systems that will meet the operating requirements of the organization. This test of feasibility asks if the system will work when it is developed and installed. This project satisfies all the operational conditions. The project is found to work well on installation, all types of users can operate the system without any difficulty. User interfaces are designed in such a way that even ordinary users without having much knowledge in computer technology can easily operate the system. The access time of data is considerably low and the operation is less time consuming .

3.5.3 Economical Feasibility

An evaluation of development cost weighted against the ultimate income or the benefit derived from the developed system or product. Economic feasibility of a system means that the cost incurred in developing and implementing a system should not be higher than the financial benefits obtained by the users. During the economic feasibility study the following points were investigated.

- The cost to conduct a full system investigation
- The cost of hardware and software for the application being developed.
- The benefits derived by the users in terms of time , effort, accuracy of information, better decision making. Etc. are quantified and compared.

3.5.4 Behavioral Feasibility

Behavioral Feasibility evaluates and estimates the user attitude or behavior towards the development of new system. It helps in determining if the system requires special effort to educate, retrain, transfer, and changes in employee's job status on new ways of conducting business.

CHAPTER 4

OPERATING ENVIRONMENT

4.1 Hardware Requirements

Development

- Processor: Intel Core i5
- Ram:8 GB
- Hard Disk: 80GB Minimum
- Keyboard: Standard 101/102 keyboard
- Monitor: 15inch monitor

Deployment

- Processor: Intel Core i3
- Ram:4 GB
- Hard Disk: 80GB Minimum
- Keyboard: Standard 101/102 keyboard
- Monitor: 15inch monitor

4.2 Software Requirements

Development

- Operating System: Windows 10
- Environment: Arduino IDE

Deployment

- Operating System: Windows 7 or above
- Environment: Arduino IDE

4.3 Tools and Platforms

Tools:

- MQ-2 gas sensors
- Arduino Nano
- LCD Display (16*2)
- I2C
- DOT PCB

- GSM module
- Connecting wires – Jumper wire, Ribbon wire, female and male jack
- USB Cable
- Laptop or any PC w/ installed Arduino IDE
- Red LED
- Buzzer
- 12v Adapter

MQ 2 Sensor:

MQ2 is amongst the most common gas sensors throughout the MQ sensor series. The detection method performed on resilience shifts in sensing element whenever the gas comes into touch only with substance. It also is referenced as nothing more than a Gas Sensor of metals metal oxides style. Gas concentration can indeed be detected via a network of simple voltage splitters.

GSM 800C sim module:

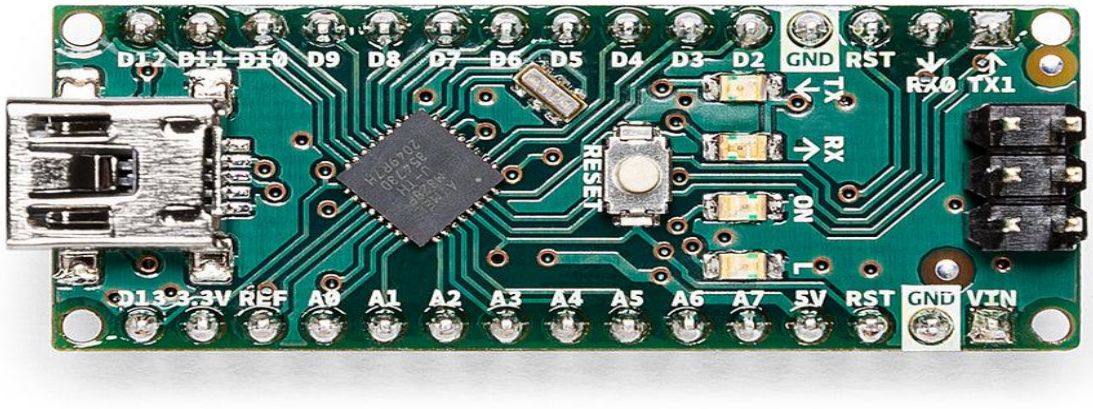
The main use of a GSM module, including the SIM 900, is to connect to a desktop with several refers to many types including such TTL Output (for Arduino, 8051 as well as other microcontrollers) and RS232 Output that directly interacts with a desktop.

LCD Display:

The LCD is a flat-screen display with the main operative mode of using liquid crystals. For businesses and customers the usage of LEDs in desktops, tv, computer monitors as well as computer monitors is broad and different. The use applications are extensively employed.

Arduino NANO:

The Arduino Nano is a small, complete, and breadboard-friendly board based on the ATmega328 (Arduino Nano 3.x). It has more or less the same functionality of the Arduino Duemilanove, but in a different package. It lacks only a DC power jack, and works with a Mini-B USB cable instead of a standard one.

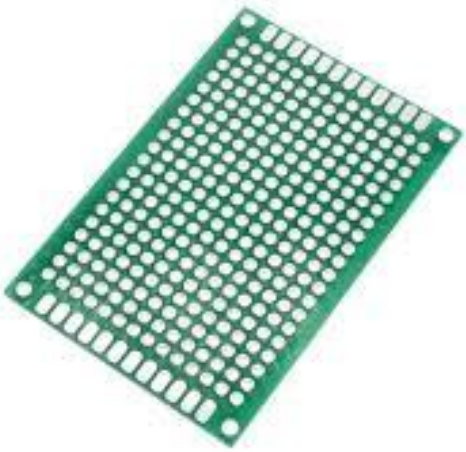


Buzzer:

Maybe a piezo buzzer is known as the Arduino buzzing. Essentially, you can link a small speaker straight to an Arduino. Clients can make this sound like a noise at the frequencies they specify. This buzzer causes a piezoelectric reversal effect depending on the noise.

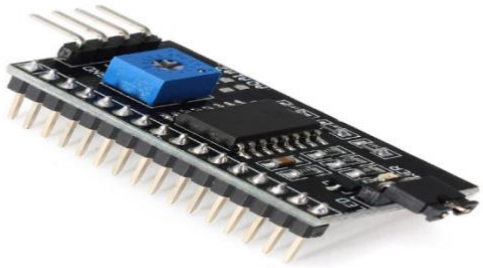
DOT PCB:

Perfboard is **a material for prototyping electronic circuits** (also called DOT PCB). It is a thin, rigid sheet with holes pre-drilled at standard intervals across a grid, usually a square grid of 0.1 inches (2.54 mm) spacing. These holes are ringed by round or square copper pads, though bare boards are also available.



I2C:

I2C Module has a inbuilt PCF8574 I2C chip that converts I2C serial data to parallel data for the LCD display. These modules are currently supplied with a default I2C address of either 0x27 or 0x3F. To determine which version you have check the black I2C adaptor board on the underside of the module. If there are 3 sets of pads labelled A0, A1, & A2 then the default address will be 0x3F. If there are no pads the default address will be 0x27. The module has a contrast adjustment pot on the underside of the display. This may require adjusting for the screen to display text correctly.



Platforms:

Embedded C

Embedded C is an extension of C language and it is used to develop micro-controller based applications. The extensions in the Embedded C language from normal C Programming. Language is the I/O Hardware Addressing, fixed-point arithmetic operations, accessing address spaces, etc. Embedded C Program has five layers of Basic Structures. They are:

- Comment: These are simple readable text, written in code to make it more understandable to the user. Usually comments are written in // or /* */.
- Pre-processor directives: The Pre-Processor directives tell the compiler which files to look in to find the symbols that are not present in the program.
- Global Declaration: The part of the code where global variables are defined.
- Local Declaration: The part of the code where local variables are defined.
- **Main function:** Every C program has a main function which drives the whole code. It basically has two parts the declaration part and the execution part. Where, the declaration part is where all the variables are declared, and the execution part defines the whole structure of execution in the program.

Arduino IDE

- **Arduino IDE** is an open-source software, designed by Arduino.cc and mainly used for writing, compiling & uploading code to almost all Arduino Modules.
- It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process. It is available for all operating systems i.e. MAC, Windows, Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role in debugging, editing and compiling the code.
- A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro and many more.
- Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code.
- The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board.
- The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the code into the given Arduino Module.
- This environment supports both C and C++ languages.

Writing Sketches:

Programs written using Arduino Software (IDE) are called **sketches**. These sketches are written in the text editor and are saved with the file extension (.ino). The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

NB: Versions of the Arduino Software (IDE) prior to 1.0 saved sketches with the extension .pde. It is possible to open these files with version 1.0, you will be prompted to save the sketch with the .ino extension on save.

Verify Checks your code for errors compiling it.

Upload Compiles your code and uploads it to the configured board. See uploading below for details.

Note: If you are using an external programmer with your board, you can hold down the "shift" key on your computer when using this icon. The text will change to "Upload using Programmer"

New Creates a new sketch.

Open Presents a menu of all the sketches in your sketchbook. Clicking one will open it within the current window overwriting its content.

Note: due to a bug in Java, this menu doesn't scroll; if you need to open a sketch late in the list, use the File | Sketchbook menu instead.

Save Saves your sketch.

Serial Monitor Opens the serial monitor.

Additional commands are found within the five menus:

- File
- Edit
- Sketch
- Tools
- Help

The menus are context sensitive, which means only those items relevant to the work currently being carried out are available.

CHAPTER 5

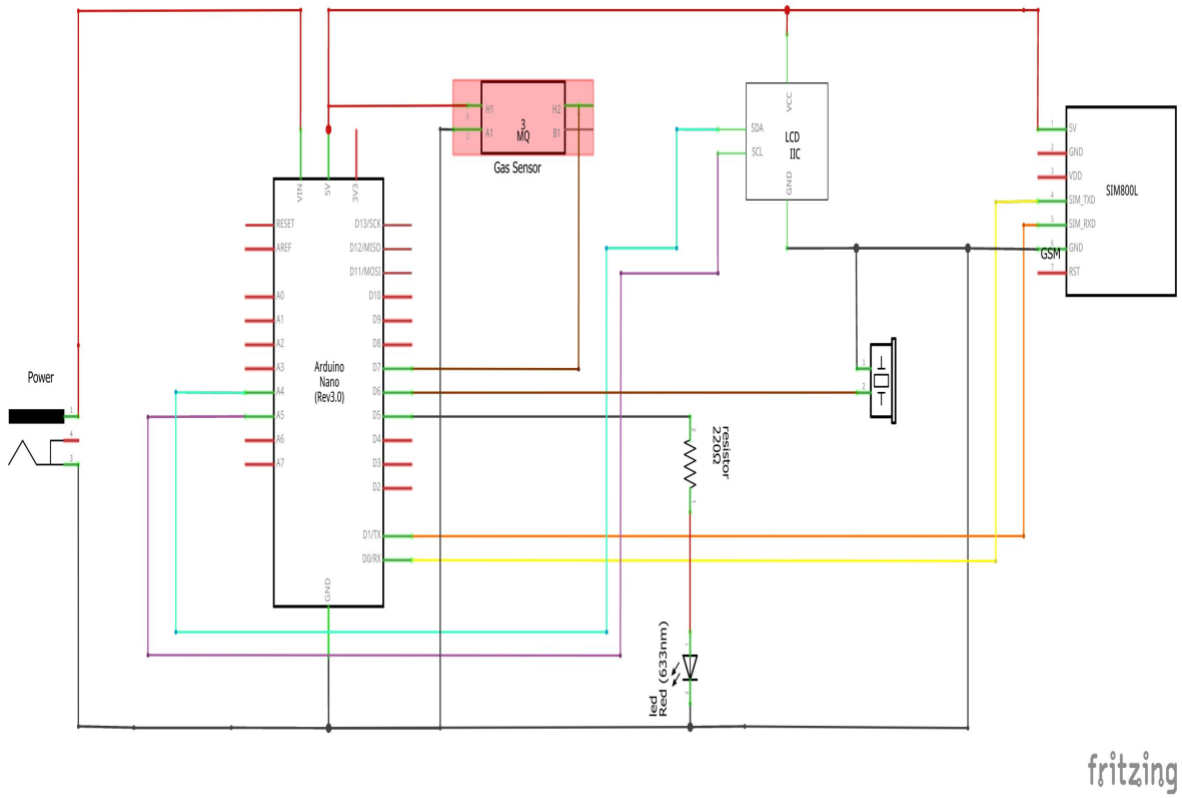
DESIGN

5.1 System Design

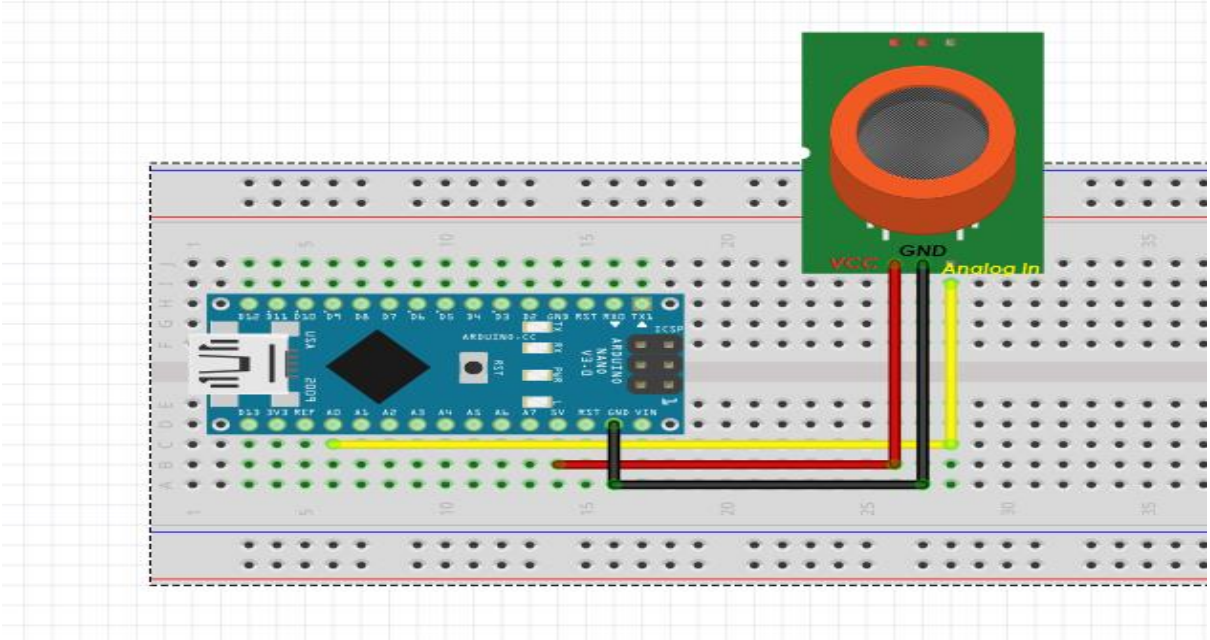
System design is the process of defining the architecture for a system to satisfy specified requirements. The purpose of design phase is to plan a solution for problem specified by the requirements. The goal of the design process is to produce a model for or representation of a system can be used later to build that system. The produced model is called design of the system.

5.2 Circuit Diagram

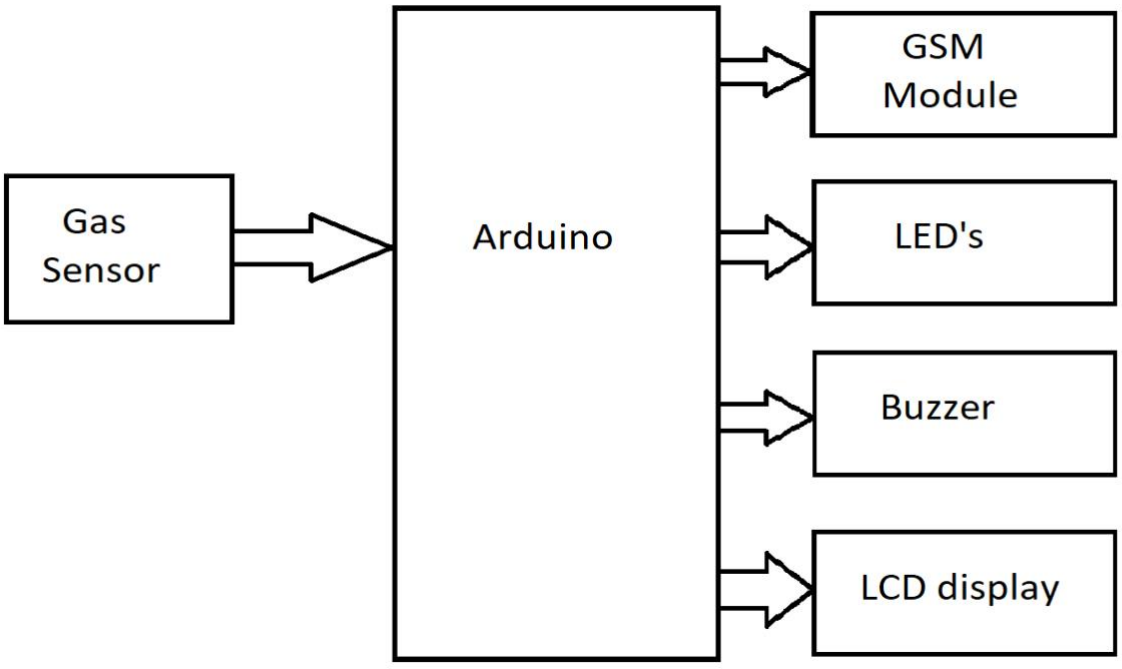
A circuit diagram is a visual display of an electrical circuit using either basic images of parts or industry standard symbols. Symbol usage depends on the audience viewing the diagram. These two different types of circuit diagrams are called pictorial (using basic images) or schematic style (using industry standard symbols). A schematic style circuit diagram is used to give a visual representation of an electrical circuit to an electrician. The pictorial style circuit diagram would be used for a broader, less technical audience.



Physical Connection of MQ-2 sensors to Arduino Nano



Block Diagram



5.2 Flow Chart

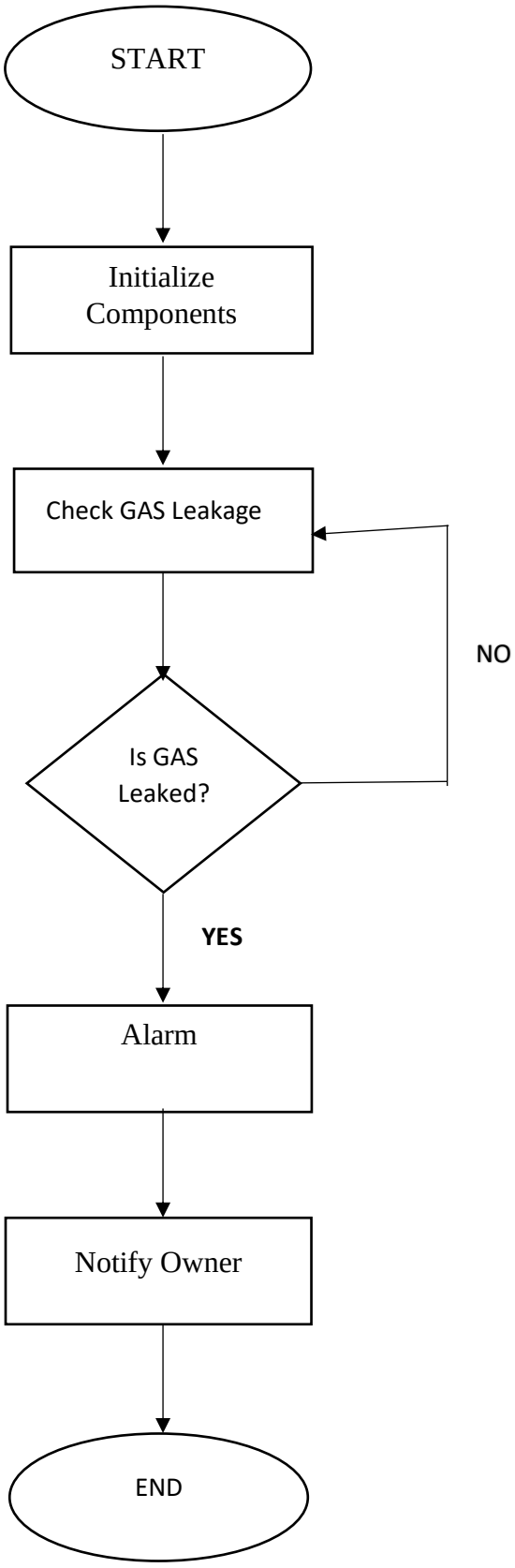
A flowchart is a type of diagram that represents a workflow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows.

Flow chart shows the whole range of operations to be carried out by the proposed gas leak detecting system.

Diagram Notations

Symbol	Name	Function
	Start/end	An oval represents a start or end point
	Arrows	A line is a connector that shows relationships between the representative shapes
	Input/Output	A parallelogram represents input or output
	Process	A rectangle represents a process
	Decision	A diamond indicates a decision

Flowchart Design



5.5 Input and Output

Each of the 14 digital pins on the Nano can be used as an input or output, using pinMode(), digitalWrite(), and digitalRead() functions. They operate at 5 volts. Each pin can provide or receive a maximum of 40 mA and has an internal pull-up resistor (disconnected by default) of 20-50 kOhms. In addition, some pins have specialized functions:

- Serial: 0 (RX) and 1 (TX). Used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip.
- External Interrupts: 2 and 3. These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attachInterrupt() function for details.
- PWM: 3, 5, 6, 9, 10, and 11. Provide 8-bit PWM output with the analogWrite() function.
- SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK). These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- LED: 13. There is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

The Nano has 8 analog inputs, each of which provide 10 bits of resolution (i.e. 1024 different values). By default they measure from ground to 5 volts, though is it possible to change the upper end of their range using the analogReference() function. Analog pins 6 and 7 cannot be used as digital pins. Additionally, some pins have specialized functionality:

- I2C: A4 (SDA) and A5 (SCL). Support I2C (TWI) communication using the Wire library (documentation on the Wiring website).

There are a couple of other pins on the board:

- AREF. Reference voltage for the analog inputs. Used with analogReference().
- Reset. Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

CHAPTER 6

PROGRAM

Program Algrorithm:

Step 1: START

Step 2: IF GAS DETECTED

Step 3: LED BLINKS, ALARM ACTIVATED, GSM ACTIVATED

Step 4: SEND SMS TO USER

Step 5: DISPLAY “Gas Leakage!!! EMERGENCY” ON PHONE

Step 6: ELSE

Step 7: CHECK PRESENCE OF GAS AGAIN

Step 8: STOP

Program Code:

```
#include <SoftwareSerial.h>
SoftwareSerial mySerial(9, 10);
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
LiquidCrystal_I2C lcd(0x27,16,2);

void setup()
{
  mySerial.begin(9600);   // Setting the baud rate of GSM Module
  Serial.begin(9600);    // Setting the baud rate of Serial Monitor (Arduino)
  delay(100);
  pinMode(3, OUTPUT);
  pinMode(5, INPUT);
  pinMode(7, OUTPUT);
  lcd.init();
}

void loop()
{
 int a=0;
```

```
a=digitalRead(5);
  if(a==0)
   {
     lcd.backlight();
     lcd.setCursor(1,0);
     lcd.print(" GAS LEAKAGE!");
     lcd.setCursor(0,1);
     lcd.print("   EMERGENCY!  ");

     digitalWrite(7,HIGH);
     digitalWrite(3,HIGH);
     delay(500);
     digitalWrite(3,LOW);
     delay(200);
     digitalWrite(3,HIGH);
     delay(500);
     digitalWrite(3,LOW);
     delay(200);

     SendMessage();
    }
  if(a==1)
  {
    lcd.backlight();
    lcd.setCursor(1,0);
    lcd.print("    NORMAL     ");
    lcd.setCursor(0,1);
    lcd.print("    WORKING!     ");
    digitalWrite(7,LOW);
    digitalWrite(3,LOW);
  }
```

```
 if (mySerial.available()>0)
   Serial.write(mySerial.read());
}

 void SendMessage()
{
  mySerial.println("AT+CMGF=1");    //Sets the GSM Module in Text Mode
  delay(1000);  // Delay of 1000 milli seconds or 1 second
  mySerial.println("AT+CMGS=\"+918304856647\"\r"); // Replace x with mobile number
  delay(1000);
  mySerial.println("GAS Leakage!!! EMERGENCY.");// The SMS text you want to send
  delay(100);
   mySerial.println((char)26);// ASCII code of CTRL+Z
  delay(1000);
}
 void RecieveMessage()
{
  mySerial.println("AT+CNMI=2,2,0,0,0"); // AT Command to receive a live SMS
  delay(10000);
 }
```

CHAPTER 7

FUNCTIONAL AND NON-FUNCTIONAL REQUIREMENTS

7.1 FUNCTIONAL REQUIREMENTS

There are two systems requirements analyses: the analysis of functional requirements and the analysis of non-functional requirements.

Functional Requirements:

Analysis of functional requirements is stages or processes which will be provided by the system overall. The stages are as follows:

• A short message will be sent automatically if the system has recognized a potential fire.

• The system will send a short message to the house owner and the fire department containing the house fire location.

Non-Functional Requirements:

Non-functional requirements are used in the form of hardware and software requirements, as shown below. The hardware and software specified in the following are required to design this system:

Non-functional requirements.

No	Requirements	Information
1	Hardware	Intel Core i3-7020U Processor, 2.3 GHz 1TB Hard Drive RAM 4 GB 1366 × 768 pixel HD Resolution - Monitor Screen Keyboard and mouse
2	Software	Windows 10 Operating System Arduino IDE

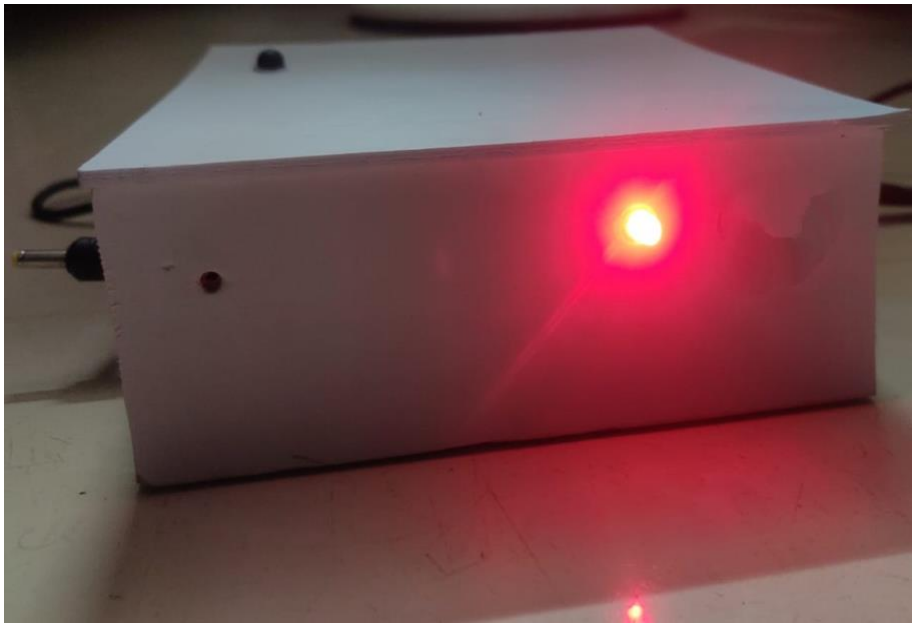
CHAPTER 8

RESULTS

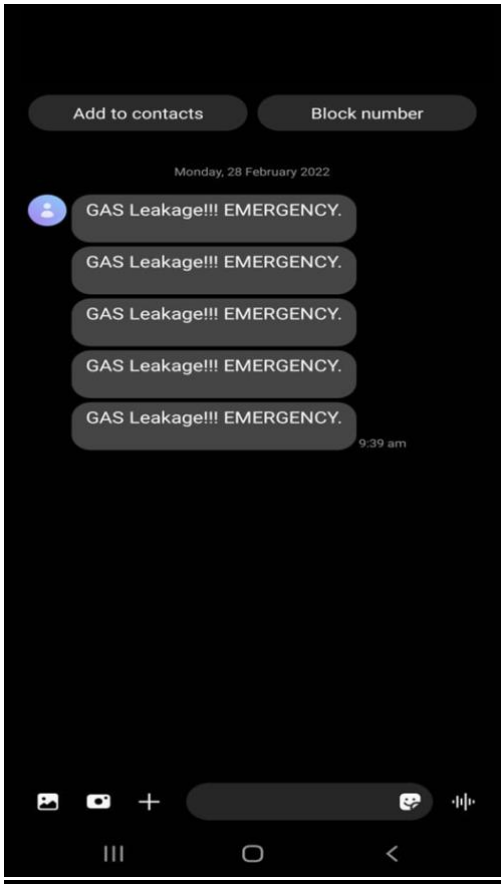
Screenshots:

Before connecting

After connected



SMS received the user to notify the gas leakage



CHAPTER 9

CONCLUSION

9.1 Conclusion

There are several gas leakage control systems in existence today and each address or focused on a particular way of providing solution to the common problem. The three monitoring and one control device incorporated enable it to address the delay in response time by automatic quick air control before the intervention of a personal despite being alerted using SMS. This idea reduces heavily the reliance on the action of a personal and increase the window of oppurtunity to avert disaster. In many similar designs like ours; Despite being cost effective; the system is not design to reduce the concentration of the gas by using air control system and that is what our system successfully incorporated with good response time of the ventilator. This project is built on GSM based monitoring system that utilizes the effective gas sensor (MQ-2), it has the capacity to sense various natural gases not only butane and propane using tin dioxide (SnO_2). Mainly this device is design to alert with a sound, SMS and provide a display on a screen. It requires no digital input, which makes the design cost effective, efficient and also environmentally friendly. In our work we focused on time of gas detection with respect to distance and the delay in response of the control system, at the end our design shows an acceptable efficiency. In our future work, we aim to calibrate the gas sensor for a specific gas to be sensed by training instead of the numerous gases it senses.

9.2 Future Enhancement

- Changes in software engineering technology are indeed rapid. By the time a decision is made to adopt a new method, conduct the training necessary to understand its applications and introduce the technology into the software development culture. Something new and better has come along and the process begins.
- Sense and automatically block the gas leakage without the help of anyone. Its very safe always without any guidance.
- Upgrade 4G sim network to 5G and above.
- Sends message through Whatsapp, telegram and e-mail.

CHAPTER 10

BIBLIOGRAPHY

10.1 Website:

- https://steemit.com/utopian-io/@jbeguna04/how-to-setup-mq-2-sensors-for-gas-detection-arduino-nano-simulation
- https://www.theengineeringprojects.com/2018/10/introduction-to-arduino-ide.html
- https://create.arduino.cc/projecthub/395902/gas-leakage-detector-system-bb0877
- https://www.smartdraw.com/circuit-diagram/#:~:text=A%20circuit%20diagram%20is%20a,the%20audience%20viewing%20the%20diagram.
- https://www.geeksforgeeks.org/what-are-the-differences-between-c-and-embedded-c/
- https://www.ijert.org/gas-detection-system-using-arduino

10.2 Publications:

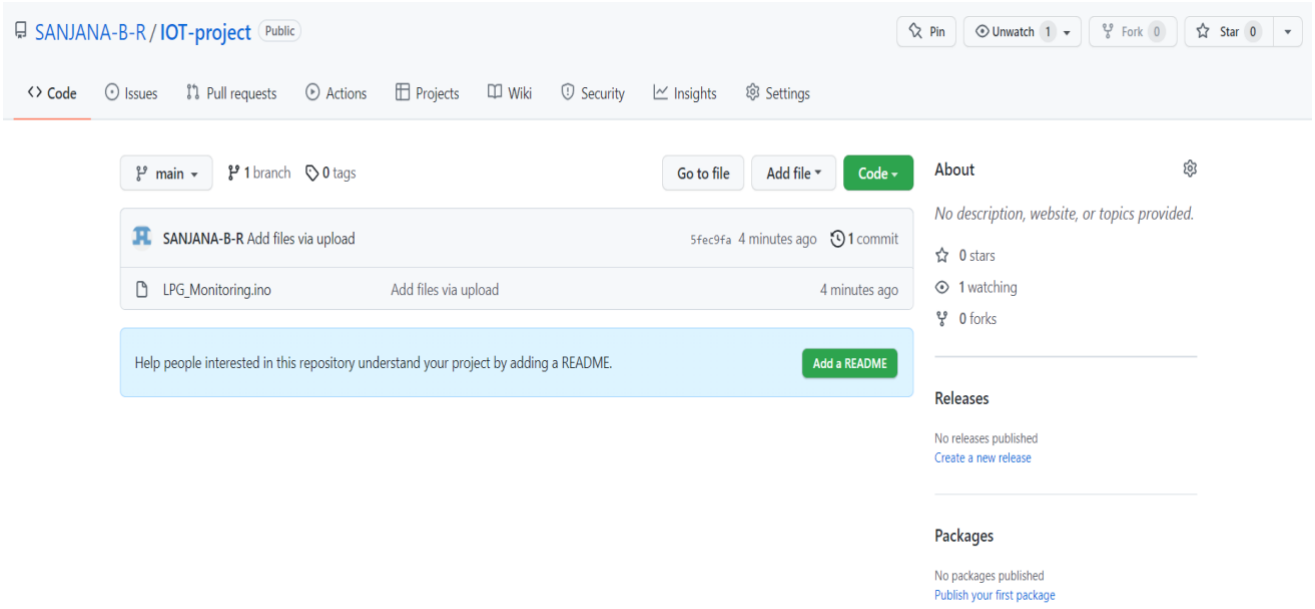
- https://www.researchgate.net/publication/342201570_Arduino_based_gas_leakage_control_and_temperature_monitoring_system
- https://www.ijert.org/research/gas-detection-system-using-arduino-IJERTCONV9IS12062.pdf

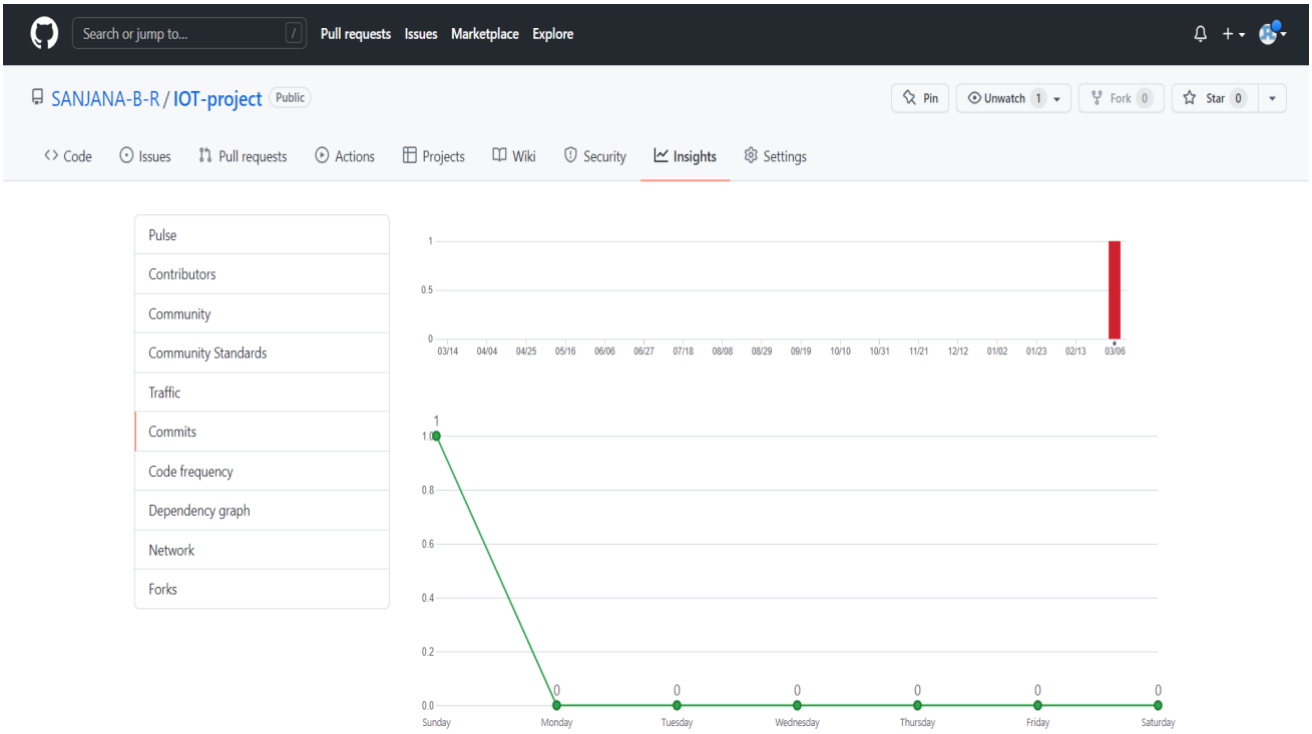
Appendices:

Abbreviations and Notations

- LPG - Liquid Petroleum Gas,
- LCD - Liquid Crystal Display,
- GSM - Global System for Mobile Communications,
- MQ 2 sensor - Arduino gas sensor,
- IoT - Internet of Things

GIT History:


SANJANA-B-R / IOT-project
Public
Pin
Unwatch 1
Fork 0
Star 0
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
main
1 branch
0 tags
Go to file
Add file
Code
SANJANA-B-R Add files via upload
5fec9fa 4 minutes ago
1 commit
LPG_Monitoring.ino
Add files via upload
4 minutes ago
Help people interested in this repository understand your project by adding a README.
Add a README
About
No description, website, or topics provided.
0 stars
1 watching
0 forks
Releases
No releases published
Create a new release
Packages
No packages published
Publish your first package


Search or jump to...
Pull requests
Issues
Marketplace
Explore
SANJANA-B-R / IOT-project
Public
Pin
Unwatch 1
Fork 0
Star 0
Code
Issues
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
Pulse
Contributors
Community
Community Standards
Traffic
Commits
Code frequency
Dependency graph
Network
Forks
03/14
04/04
04/25
05/16
06/06
06/27
07/18
08/08
08/29
09/19
10/10
10/31
11/21
12/12
01/02
01/23
02/13
03/06
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday