IOT BASED SMART GAS MANAGEMENT SYSTEM

A Major Project Report

Submitted in the partial fulfillment of the requirements for

the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

by

Sony Shrestha (150031122)

R. Chaitanya (150031090)

A. Akhila (150030024)

Under the Esteemed Guidance of:

A.V. Praveen Krishna

M. Tech [CSE], (Ph.D.)

(DEEMED TO BE UNIVERSITY)

Department of Computer Science and Engineering

K L Deemed to be University

Green Fields, Vaddeswaram- 522502, Guntur, Andhra Pradesh, India.

November, 2019

KL Deemed to be University

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

DECLARATION

We hereby declare that the project entitled "IOT based Smart Gas Management System" submitted for the B. Tech. (CSE) degree is my original work and the project has not formed the basis for the award of any other degree, diploma, fellowship or any other similar titles.

Signature of the Student

Place: Vaddeswaram, Guntur, Andhra Pradesh

Date: 18/11/2018

KL Deemed to be University

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project report entitled "IOT based Smart Gas Management System" is the bonafide work carried out by Sony Shrestha (150031122), R. Chaitanya (150031090) and A. Akhila (150030024), students of B Tech (CSE) of Koneru Lakshmaiah Education Foundation (Deemed to be University), Guntur District, Andhra Pradesh, India, during the academic year 2018-19, in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology (Computer Science and Engineering) and that the project has not formed the basis for the award previously of any other degree, diploma, fellowship or any other similar title.

Signature of the Guide

Place: Vaddeswaram, Guntur, Andhra Pradesh

Date: 18/11/2018

Acknowledgement

It is great pleasure for me to express my gratitude to our honorable President **Sri. Koneru Satyanarayana**, for giving the opportunity and platform with facilities in accomplishing the major project.

I express the sincere gratitude to our principal **Dr. A. Anand Kumar** for his administration towards our academic growth.

I record it as my privilege to deeply thank our pioneer **Dr. Hari Kiran Vege**, HOD, Department of CSE for providing us the efficient faculty and facilities to make our ideas into reality.

I express my sincere thanks to our project supervisor and guide A.V. Praveen Krishna for his novel association of ideas, encouragement, appreciation and intellectual zeal which motivated us to venture this project successfully.

Finally, it is pleased to acknowledge the indebtedness to all those who devoted themselves directly or indirectly to make this project report success.

Abstract

The problem of gas leakage and gas wastage is often encountered in our day-to-day life. LPG, Liquefied Petroleum Gas, is highly flammable gas used as fuel in heating appliances. Leakage of this gas raises the risk of building fire, suffocation or an explosion. The mentioned problem can be solved with the development of reliable techniques to detect gas leakage. As soon as gas leakage will be detected, user will be notified via SMS so that he/she can turn off gas valve from anywhere in his work place. The issue of gas wastage can be monitored with the help of infrared sensor. Fire can be detected by using fire sensor. The buzzer starts beeping whenever no vessel is detected over the gas stove beyond a certain amount of time period. In addition to these, it is often found that a person forgets to book gas cylinder due to his/her busy schedule. The main aim of our project is to design an IOT based Smart Gas Management System that will be able to detect gas leakage and fire. With the help of load sensor, automatic booking of a gas cylinder is also facilitated. Notification is sent to the booking agency to book a gas cylinder whenever load cell detects that the weight of gas in cylinder has reached below a particular threshold value.

Keywords

IOT, LPG, Gas Sensor, Infrared Sensor, Load Cell

Table of Contents

List of Abbreviations

IOT	Internet of Things
LPG	Liquefied petroleum gas
WIFI	Wireless Fidelity
GSM	Global System for Mobile
SMS	Short Message Service
LCD	Liquid Crystal Display
LED	Light Emitting Diode
IR	Infrared
UV	Ultra Violet
IDE	Integrated Development Environment
USB	Universal Serial Bus
DC	Direct Current
V	Volt
Hz	Hertz
MHz	Megahertz
mA	Milli ampere
KB	Kilo byte
TWI	Two Wire Interface
SPI	Serial Peripheral Interface
RAM	Random Access Memory
ROM	Read Only Memory

PROM	Programmable Read Only Memory
EPROM	Erasable Programmable Read Only Memory
EEPROM	Electrically Erasable Programmable Read Only Memory
M2M	Machine-to-machine
I/O	Input/ Output
ADC	Analog to Digital Converter
DAC	Digital to Analog Converter
RISC	Reduced Instruction Set Computer
CISC	Computer Instruction Set Computer
ISA	Instruction Set Architecture
CPU	Central Processing Unit
CO	Carbon Oxide
USB	Universal Serial Bus
Tx	Transmit
Rx	Receive

List of figures

List of tables

1. Introduction

LPG (Liquefied Petroleum Gas), simply called as propane or butane, is highly flammable gas that is often used as fuel for cooking purposes. This flammable mixtures of hydrocarbons gases are used as source of fuel in heating appliances, cooking equipment and many vehicles. Due to the flammable nature of LPG, its leakage can cause damage to life and property. Study shows that many people are losing their life due to leakage of gas in their kitchen. It is very important to be sure that gas leakage has not occurred in our kitchen. LPG, being heavier than air, does not disperse easily and leads to suffocation when inhaled. The ignition of leaked gas leads to explosion. To reduce the mortality rate due to gas leakage, it is important to implement Smart Gas Management System which will be able to detect the leakage of gas. As soon as leakage of gas is detected, user will be notified about it via an SMS so that he/she can take further action.

Fire and flames in kitchen can also damage lots of life and property. It is important to control such fire in case it occurs before much more damages have been made.

When a consumer forgets to turn off the gas burner, major amount of gas gets wasted. Gas wastage is another issue that must be addressed to prevent unnecessary wastage of gas in kitchen. Similar to gas leakage detection, it is important to implement a System which will be able to detect the wastage of gas. As soon as wastage of gas is detected, user will be notified about it via an SMS so that he/she can turn of gas burner.

It is often found that a person forgets to book gas cylinder due to his/her busy schedule. If booking of gas cylinder is not done on time, user might face a lots of problem. To overcome this issue, a System can be implement which will be able to monitor the level of gas in cylinder continuously. on detection of gas weight below some threshold value, automatic booking of gas cylinder can be facilitated.

Our proposed topic deals with these issues and provides an efficient solution to overcome them. This Smart Gas Management System will detect the leakage of gas with the help of gas sensor. The user will be notified about the leakage via SMS so that he/she can turn off gas valve from anywhere in his work place. Infrared sensor detects the wastage of gas and notifies the user when no vessel is detected above the burner after a particular amount of time period. Fire sensor detects the fire and flame in the area where it has been housed. Load cell will continuously measure the weight of gas cylinder. A notification will then be sent to the booking agency to book a gas cylinder whenever the load cell detects that the level of gas cylinder has reached below a particular threshold value.

In this regard, IOT based Smart Gas Management System deals with three major issues which is often encountered in our kitchen, i.e. Detection of Gas Leakage, Detection of fire and Automatic booking of gas cylinder. Gas Sensor (MQ2 Sensor), having high sensitivity to LPG, detects gas leakage as soon as it occurs. Flames and fire is detected by using Fire Sensor. Load Cell is used for continuous monitoring of weight of gas in cylinder. A threshold value is set which indicates that gas in cylinder is going to be completed soon and new gas should be booked soon before completion of previous one. The amount of gas being leaked and wasted and weight of gas available in cylinder can be displayed in LCD Display. Any wrong goings are indicated by beeping buzzer such as when gas is being leaked and when gas wastage is taking place.

1.1. Problem Statement

Many people today are too busy with their work and do not have enough time to concentrate on minor tasks like checking if the gas valve has been turned off properly or not, if gas burner is left open while there is no vessel over it or not and if gas in cylinder is about to get completed or not. All these tasks seem to be minor one but have severe impacts. Gas Leakage and fire can take life of thousands of people. Turning on gas burner while there is no vessel over it causes unnecessary wastage of gas. People often face problem if they forget to book gas cylinder on time, especially when there is scarcity of gas cylinder.

Gas Leakage might occur due to some negligence of user in kitchen i.e. leaving gas valve open. Huge damage to life and property can take place if such leakages are not detected on time. LPG, also called as Liquefied Petroleum Gas, is a group of flammable hydrocarbon gases liquefied through pressurization and comes from natural gas processing and petroleum refining. This gas is often used as fuel in heating, cooking, hot water and vehicles. It is flammable in nature and heavier than air. Due to this, this gas does not get dispersed easily and leads to suffocation when inhaled.

Gas Wastage is encountered when gas burner is left open even though it is not in use. Leaving gas burner open when it is not in use causes unnecessary wastage of gas. Gas is said to be wasted when the gas burner is left open and there is no vessel over it beyond particular time period. It is not a good practice to waste gas unnecessarily. Similarly, when an area gets caught with fire, huge amount of life and property gets destroyed.

It is often found that a person forgets to book gas cylinder due to his/her busy schedule. If booking of gas cylinder is not done on time, user might face a lots of problem. Problem faced is significant in case there is shortage of gas cylinder in market.

The proposed IOT based Smart Gas Management System provides solution to all these problems by using Gas Sensor, Fire Sensor and Load Cell to detect gas leakage, fire and weight of gas in cylinder respectively. Gas Sensor detects the leakage of gas and notifies it to user through buzzer and an SMS. Similarly, Fire Sensor detects the fire and notifies it to user through buzzer and an SMS. Load cell keeps track of weight of gas in cylinder and when the weight reaches below some threshold vale, automatic booking of gas cylinder is facilitated.

1.2. Motivation and Goals

The motivation for conducting the research in the area of Smart Gas Management System in IOT is the lack of sustainable and flexible solutions to address issues often encountered in our kitchen like gas leakage, gas wastage and booking of gas cylinder. Some of the early solutions provided are now being implemented but they still need improvement and lack some standardization. Several systems are proposed which solve these issues separately. i.e. There is Gas Leakage detector which only detects the leakage of gas. It can provide no information on wastage of gas and booking of gas cylinder. Similarly, there is Gas Wastage Detector which only detects the wastage of gas and provides no information on leakage of gas and booking of gas cylinder. Fire detector only detects flame and fire. The full potential of IOT means to go beyond the enterprise centric systems and move towards a user inclusive IOT. The System designed should be capable to solve multiple issues encountered on hand. The main aim of this project is to design a system which provides solution to all the problems that might occur in a kitchen.

1.3. Objectives

The various objectives of this research are:

- Design a Smart Gas Management System that is able to detect leakage of gas and notify it to user via an SMS and a buzzer
- Design a system that is able to detect fire and notify it to user via an SMS and a buzzer
- Design a system that is able to keep track of weight of gas inside cylinder and facilitate automatic booking of gas cylinder when weight of gas reaches below some threshold value
- Propose a novel system for solving issues often encountered in our kitchen
- Analyze the pros and cons of proposed system
- Discuss system's weaknesses, if present
- Find out various areas in which the system can be used
- Analyze behavior of various components used in the system

1.4. Delimitation

The following aspects are not investigated and not done in this project.

- The detailed internal architecture of components used in the system
- The detailed process for the software implementation

1.5. Methodology

The main focus of this project falls on designing the system that is capable to detect the leakage of gas, monitor gas wastage and facilitate automatic booking of a gas cylinder. The pros and cons of system is analyzed. the weaknesses of proposed system are discussed. The behavior of various components used in the system is analyzed.

In the system, Gas Sensor, Load Cell and Fire Sensor are interfaced to the microcontroller. These sensors act as input to the system. The interfaced Wi-Fi module is capable of giving microcontroller the access to available Wi-Fi Network. Gas Sensor detects the leakage of gas in area around it. As soon as leakage of gas is detected, user is notified about it and he/she can turn off the gas valve from anywhere in his/her work place. Load cell continuously monitors the weight of gas in cylinder and on detection of its weight below some critical value, a notification will be sent to booking agency to book a gas cylinder. Fire sensor detects fire in the area where it is placed. If fire is detected, buzzer starts beeping and necessary actions can be taken to control it. The response of all these sensors can be viewed in LCD-Display.

1.6. Novelty and Contribution

The main aim of this thesis is to research various solutions available to design Smart Gas Management System. the novel contribution made is designing a system that is capable of detecting gas leakage, gas wastage and booking of gas cylinder at same time. Several systems were proposed which solved these issues separately. i.e. There is Gas Leakage detector which only detects the leakage of gas. It can provide no information on wastage of gas and booking of gas cylinder. Similarly, there is Gas Wastage Detector which only detects the wastage of gas and provides no information on leakage of gas and booking of gas cylinder. Fire sensor only detects fire and sends necessary notification alert to user. The System designed should be capable to solve multiple issues encountered on hand. The proposed system provides solution to all the problems that might occur in a kitchen.

1.7. Outline of the thesis

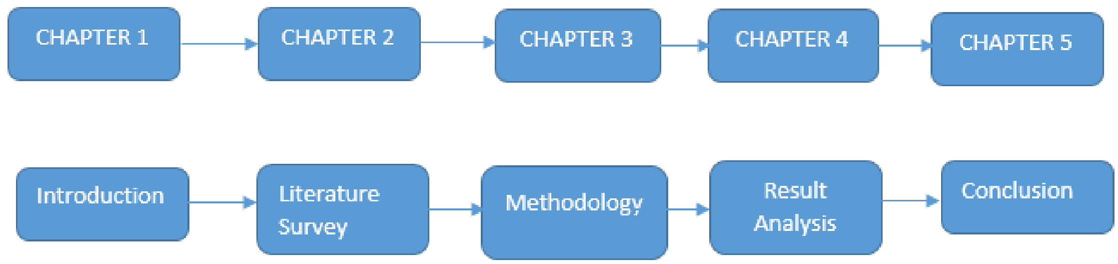

Fig 1.1 Chapter content overview of the thesis

This section provides an overview of the structure of this thesis. The thesis consists of six chapters.

Chapter 1 consists of brief introduction of the project topic, along with problem statement, motivation and goals, objectives, delimitation, methodology, novelty and contribution and outline of the project.

Chapter 2 is the literature review (literature survey) chapter. In this chapter, all the research works that have already been completed regarding this topic area are identified. All the current information relevant to the topic is analyzed and mentioned in this chapter.

Chapter 3 provides detailed explanation of the methodology that has been proposed to solve the problem. It is the core part of thesis which talks about how the methodology has been executed. It also provides detailed description of the components of the methodology. Clear explanation on why this method has been used for research is described in this chapter.

Chapter 4 consists of final results of the research and analysis of these results. The screenshots of different output are included.

Chapter 5 concludes the thesis. An overview of research process is presented. It provides overall summary of work carried out during the entire project.

2. Literature Survey

The proposed topic "IOT based Smart Gas Management System" deals with three major issues, gas leakage detection, fire detection and automatic booking of gas cylinder. This system makes use of gas sensor, fire sensor and load cell that will detect the leakage of gas cylinder, detect fire and continuously monitor the amount of gas in cylinder respectively. It has a provision of closing the valve automatically on detection of gas leakage. Buzzer starts beeping on detection of fire and gas leakage. Automatic booking of gas cylinder is facilitated by notifying booking agency to book a cylinder whenever load cell detects that the level of gas cylinder has reached below a particular threshold value.

2.1. IOT

2.1.1. Introduction to IOT

IOT, also called as Internet of Things, is the ability of various devices to be connected to each other through Internet. It is the network of physical devices, home appliances, vehicles and other items which are embedded with sensors, electronics, software, actuators, enabling these things to connect, collect and exchange data. It deals with expanding Internet connectivity beyond standard devices like laptops, mobile phone, tablets and desktops, to any traditionally dumb physical devices that are non-internet enabled. this technology enables physical devices to communicate and interact over the Internet, each of which can be monitored and controlled remotely.

Fig 2.1 Basic IOT diagram

2.1.2. Applications of IOT

The various applications of IOT include Smart home, Wearables, Smart City, Smart grids, connected car, Connected health, Industrial internet, Smart Retail, Smart Supply Chain, Smart Farming, Smart kitchen and so on.

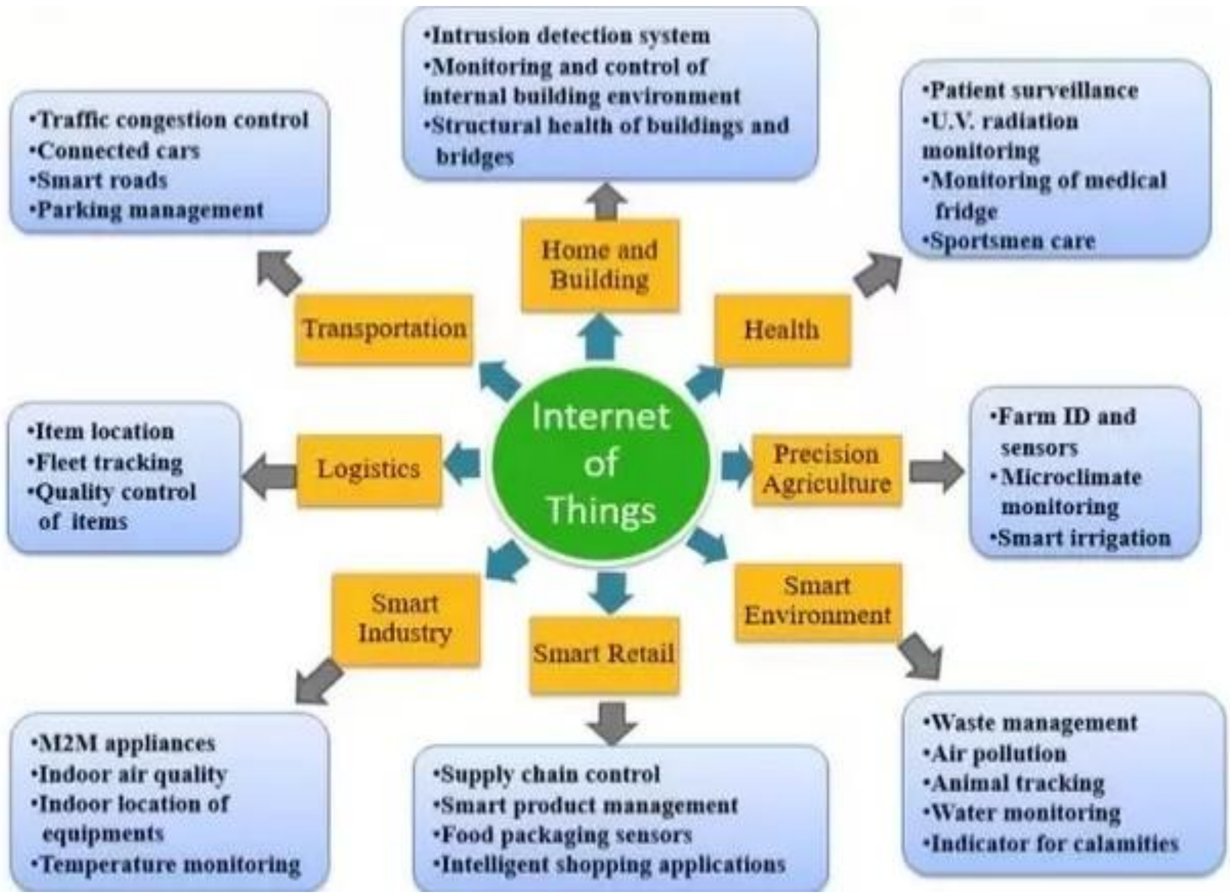

Fig 2.2 Applications of IOT in different domains

These applications can be categorized into different domains such as:

a) Home and Building

The concept of IOT is used in home and building for intrusion detection system, monitoring and controlling of internal building environment, analyzing structural health of buildings and bridges and so on. It can be used in home automation and building automation system to monitor and control the electrical, mechanical and electronic systems used in buildings. IOT driven "Smart Buildings" can be constructed by the integration of the Internet with building energy management systems.

b) Health

The application of IOT in the field of healthcare ranges from remote monitoring to integration of sensors and medical equipment. This technology is often used for patient surveillance, UV radiation monitoring, monitoring of medical fridge, sportsman care and so on. It has potential to keep patient safe and healthy. the medical condition of patient can continuously be monitored based on which physicians recommend treatment to be made.

c) Transportation

The concept of IOT that is used in the field of Transportation helps in traffic congestion control, identification of traffic violence, building connected cars, smart roads, parking management, etc. This technology can be used to monitor the traffic in city making identification of area with low traffic easier and enabling ambulance driver to take same route.

d) Logistics

IOT is used in the field of logistics for location tracking, fleet management, quality control of items, environment sensing and so on. It is used for tracking deliveries from the vendor to the manufacturing facility, tracking deliveries and materials around and inside manufacturing facility, monitoring sensitive goods to avoid damages and loss and monitoring assets to mitigate risk.

e) Smart Industry

IOT is used in Smart Industry which provides applications such as M2M appliances, indoor air quality, indoor location of equipment, temperature monitoring, manufacturing and so on. it facilitates increasing operational efficiency, optimizing production of goods and increasing safety of workers in industry.

f) Smart Retail

The applications of IOT under Smart Retail domain include Supply chain control, Smart product management, food packaging sensors, intelligent shopping applications. It automatically checks the inventory and real-time sale status and monitor information through the applications by mobile devices.

g) Smart environment

The applications of IOT under Smart environment include Waste management, Monitoring air pollution, Animal tracking, water monitoring, indication of calamities. Quality of air and water, atmospheric or soil conditions can also be monitored. it can even include the areas like monitoring the movements of wildlife in their habitats. This technology is also able to build the system that warns about various natural calamities prior to its occurrence.

h) Precision Agriculture

IOT helps in Smart Irrigation, Microclimate monitoring, collecting data on temperature, rainfall, humidity, pest infestation and soil content. The collected data can be used in automating farming techniques, taking appropriate decision to improve quality and quantity of crops, minimizing risks and reducing efforts required to manage crops. The rate of production of crop is found to be comparatively more by using Smart IOT techniques as compared to without using them.

2.2. Gas Sensors

2.2.1. Introduction to Gas Sensors

Gas Sensors are the devices that can detect the presence of different gases in a particular area. These devices are used in many IOT projects as an input which is interfaced by into the microcontroller through input pin. They can detect toxic, flammable, flammable gases along with oxygen depletion. Detection is helpful to prevent harmful effects caused by various hazardous gases. It is used as an important component of a safety system. In this project, gas sensor is used to detect the leakage of LPG so that explosion caused by it can be minimized. This ultimately saves life and property of people. These sensors have wide application in the field of environmental monitoring and protection. The properties of gas sensors such as surface area, porosity, agglomeration affect their sensing characteristics like stability, time to response, selectivity, sensitivity, durability, reproducibility and reversibility. The gas sensor having high specific surface is highly sensitive. Therefore, many researchers are concentrating on employing the techniques that can increase the specific surface of sensing device.

Gas leakage detection is achieved by using gas sensors. They are often integrated with a buzzer that starts beeping to alert people about leakage of harmful gases. Gas sensors can be categorized into two main types: portable devices and fixed gas detector. Portable detectors, usually battery operated, are either hand-held or worn on clothing to monitor the atmosphere around some personnel. The position of fixed type gas detectors is fixed in an area to be protected. The various gas sensors available in the market include MQ-2, MQ-3, MQ-4, MQ- 5, MQ- 6, MQ-7, MQ-8, MQ-9, MQ-131, MQ-135, MQ136, MQ137 and so on.

Sensor	Detects	Heater Voltage
MQ-2	Methane, Butane, LPG, Smoke	5V
MQ-3	Alcohol, Ethanol, Smoke	5V
MQ-4	Methane, CNG Gas	5V
MQ-5	Natural Gas, LPG	5V
MQ-6	LPG, Butane Gas	5V
MQ-7	Carbon Monoxide	Alternating 5V and 1.4V
MQ-8	Hydrogen Gas	5V
MQ-9	Carbon Monoxide, Flammable Gases	Alternating 5V and 1.4V
MQ-131	Ozone	6V
MQ-135	Air Quality (Benzene, Alcohol, Smoke)	5V
MQ-135	Hydrogen Sulfide Gas	5V
MQ-137	Ammonia	5V
MQ-138	Benzene, Toluene, Alcohol, Acetone, Propane, Formaldehyde gas, Hydrogen	5V
MQ-214	Methane, Natural Gas	6V

Table 2.1 Comparison of different Gas Sensors

2.2.2. Working principle of gas sensor

The interaction of gas with gas sensor causes ionization of gas molecule into its constituents which is then absorbed by the sensing element. As a result of this absorption, potential difference is created on the element and conveyed to the processor unit in the form of current through the path of output pins. The following diagram illustrates how an MQ-2 Gas Sensor looks like.

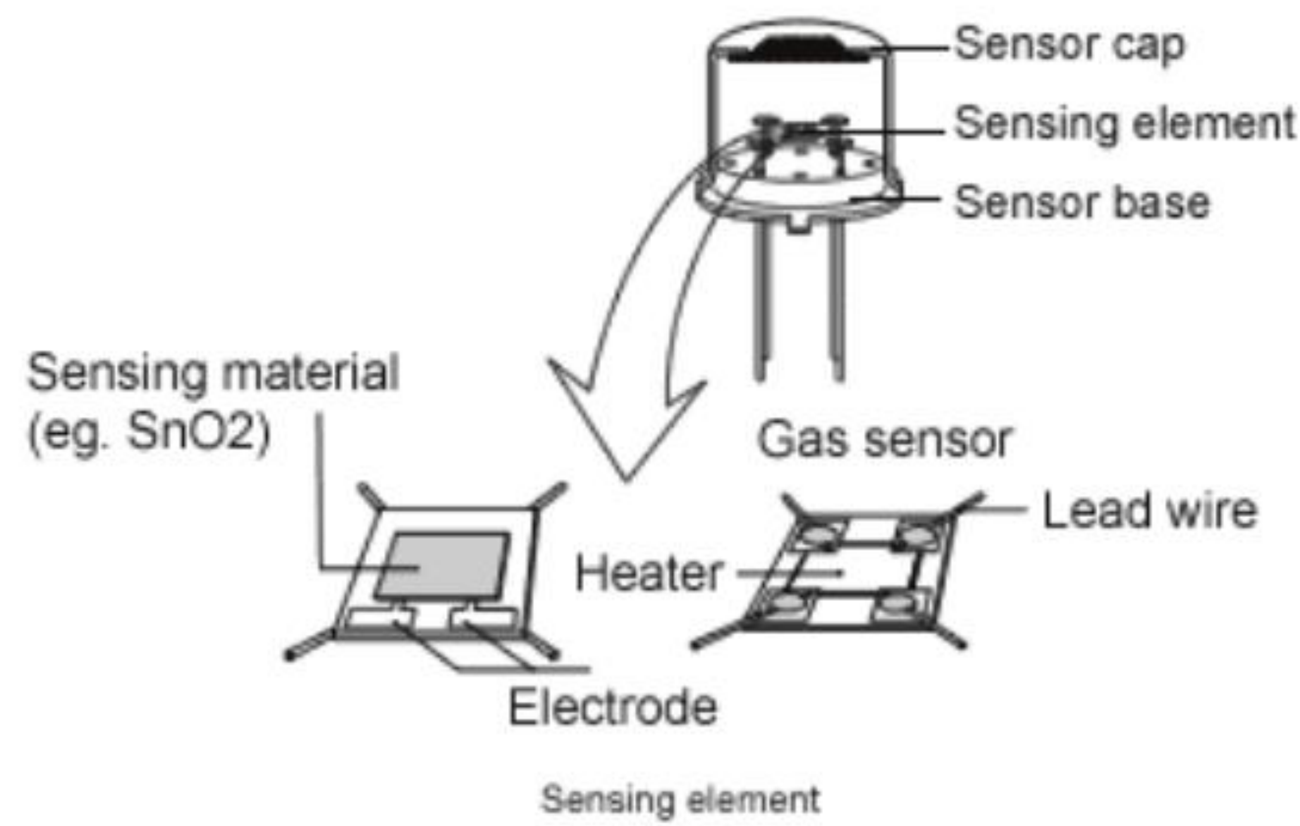

Fig 2.3 Working principle of Gas Sensor

2.3. Fire Sensor

2.3.1. Introduction to Fire sensor

Fire sensor is the device used to detect presence of flame or fire in the area where it is placed. When the flame has been detected, user can cause several actions to happen based on their requirement by interfacing it to several other components. A buzzer can be made to beep on detection of fire. Similarly, a fuel line can be made to deactivate and fire suppression system can also be activated. Flame detectors can be divided into three different categories: optical detectors, UV detectors, IR detectors. Optical detectors make use of optical sensors to detect flames. UV detectors can detect flame more quickly as compared to other detectors. UV radiations emitted at the time of ignition are used to detect flames and fires. IR detectors detects flame and fire by monitoring head radiations generated.

The specifications of fire sensor are discussed in the table below.

Parameter	Value
Operating voltage	+5V
Range	2m
Fire Detection	Active high output

Table 2.2 Specification of Fire Sensor

Pin No.	Pin Name	Details
1	Out	Active high output
2	+5V	Power supply input
3	GND	Power supply ground

Table 2.3 Pin configuration of Fire Sensor

2.3.2. Working principle of fire sensor

Fire sensor is most commonly used in many IOT projects as an input interfaced to microcontroller through input port. It can detect presence of fire in the area where it is housed. The module of fire sensor consists of IR sensor, comparator and LED. It has 3 pins namely gnd, vcc and out.

When IR sensor detects fire, LED glows and out pin is set high. The out pin of fire sensor acts as input to the microcontroller. Glowing LED indicates that the area has been caught with fire. The range of the sensor is 2 meter. It is possible to connect buzzer via microcontroller to provide audio signal on detection of fire.

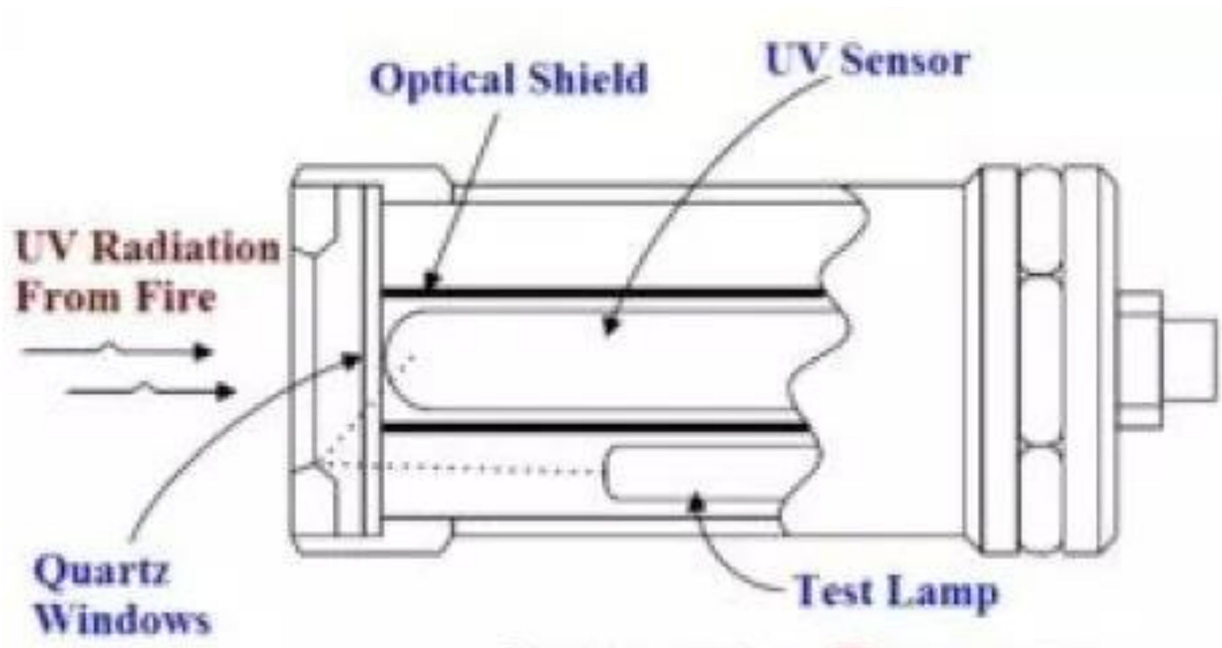

Fig 2.4 UV type fire detector

2.4. Load Cell

2.4.1 Introduction to Load Cell

Fig 2.5 Load cell

Load cell is a type of transducer that converts mechanical force into a measurable electrical output. Anything that needs to be weighted probably uses it for that purpose. Among the various varieties of load cells, strain gauge-based load cells are mostly used. It works on the principle that the strain gauge deforms when the material of the load cells deforms appropriately. It is available in different shapes and sizes. Appropriate one among them can be selected and added to different machinery and weighing equipment. The major features of load cell include its stiffness, good resonance values and tending to have long life cycles in application.

2.5. Microcontroller

2.5.1. Introduction to Microcontroller

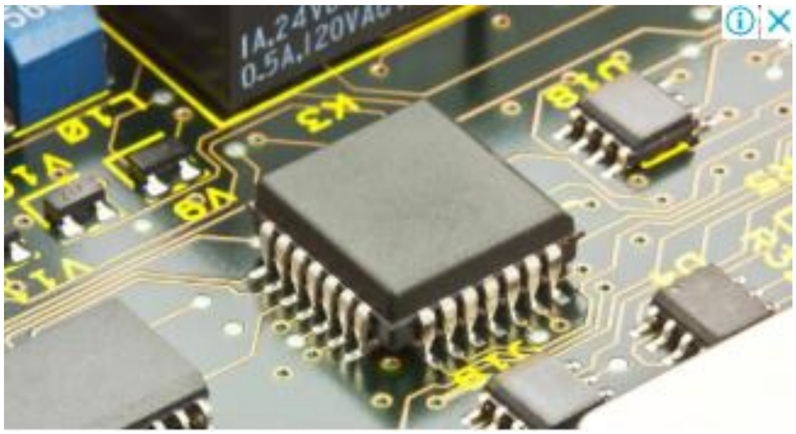

Fig 2.6 Microcontroller

Microcontroller is a IC chip that executes programs for controlling other connected devices. It comprises processor, memory, I/O pins for input and output. Microcontroller is called so because of its controlling actions. It is a small computer on a single integrated circuit. It is similar but less sophisticated than SoC (System on a chip).

The first microprocessor, 4-bit Intel 4004 was invented in 1971. It is the integral part of embedded systems which comprises a processor, small memory and programmable input-output peripherals. It is mainly used in products which are automatically controlled and performs predefined and preprogrammed tasks. In a specific system, number of input and output devices are interfaced into the microcontroller and is preprogrammed in such a way that system operates as according to the instructions provided in the program.

For example, let us consider a system to clarify the concept of microcontroller. In the IOT based Smart Gas Monitoring System, Gas Sensor (MQ2 Sensor), Load Cell and Infrared Sensor (GP2D120 Sensor) are interfaced to the microcontroller ATMEGA328. These sensors act as input to the system. The interfaced Wi-Fi module is capable of giving microcontroller the access to available Wi-Fi Network. Gas Sensor detects the leakage of gas in area around it. As soon as leakage of gas is detected, user is notified about it and he/she can turn off the gas valve from anywhere in his/her work place. Load cell continuously monitors the weight of gas in cylinder and on detection of its weight below some critical value, a notification will be sent to booking agency to book a gas cylinder. Object detector sensor detects the presence of vessel over gas burner. If absence of vessel detected over the gas burner beyond a particular time period, buzzer starts beeping and gas wastage is monitored. The response of all these sensors can be viewed in LCD-Display.

2.5.2. Areas where microcontrollers are used

Microcontrollers are used in embedded systems such as vending machines, washing machine, digital cameras, medical equipment, automobiles, smart phones, robots, smart watches and various home appliances.

2.5.3. Uses of Microcontroller

The use of microcontroller is visualized in the field of employing automation in embedded applications. The use of microprocessor in any system reduces the size and cost of the system, compared to one that makes use of microprocessor, memory and input/ output devices separately. The various features of microcontrollers such in-built microprocessor, RAM, ROM, Parallel Interfaces, Serial Interfaces, ADC (Analog to Digital Converter), DAC (Digital to Analog Converter) facilitates building required applications easier.

2.5.4. Types of Microcontroller

There are various types of microcontrollers available in the market each of which has different specifications like programmable memory, flash size, supply voltage, input/out pins, speed, etc. based on parameters, they can be classified into different subcategories.

a) According to data bus (bit size)

Based on the bit size of microcontroller, it can be classified as 8-bit to 32-bit microcontroller. The data bus of 8-bit microcontroller consists of 8 data lines. Similarly, the data bus of 16-bit microcontroller consists of 16 data lines and so on for all other microcontrollers.

b) According to memory

Microcontroller requires memory such as RAM (Random Access Memory), ROM (Read Only Memory), EPROM (Erasable Programmable Read Only Memory), EEPROM (Electrically Erasable Programmable Read Only Memory), flash memory for storing data and programs. It can be categorized based on this memory required in the microcontroller. Some microcontrollers available in the market have inbuilt memory chips while others require an external memory to be connected. They are respectively known as embedded memory microcontrollers and external memory microcontrollers.

c) According to number of Input/ Output pins

The number of input and output pins in microcontroller varies from one another. Based on the application to be designed, suitable one with appropriate number of input and output pin needs to be selected.

d) According to Instruction Set

Microcontrollers can make use of either RISC (Reduced Instruction Set Computer) or CISC (Computer Instruction Set Computer). The instruction set architecture (ISA) of RISC allows it to have fewer cycles per instruction than a CISC. RISC can operate at a higher speed. CISC allows application of one instruction as an alternative to many other instructions.

e) According to Memory Architecture

Based on memory architecture of microcontroller, it can be classified into Harvard memory architecture microcontrollers and Princeton memory architecture microcontrollers.

2.5.5. Features of Microcontrollers

Microcontroller is comprised of processor, I/O ports, timers, serial ports, DAC, ADC and Interrupt Control.

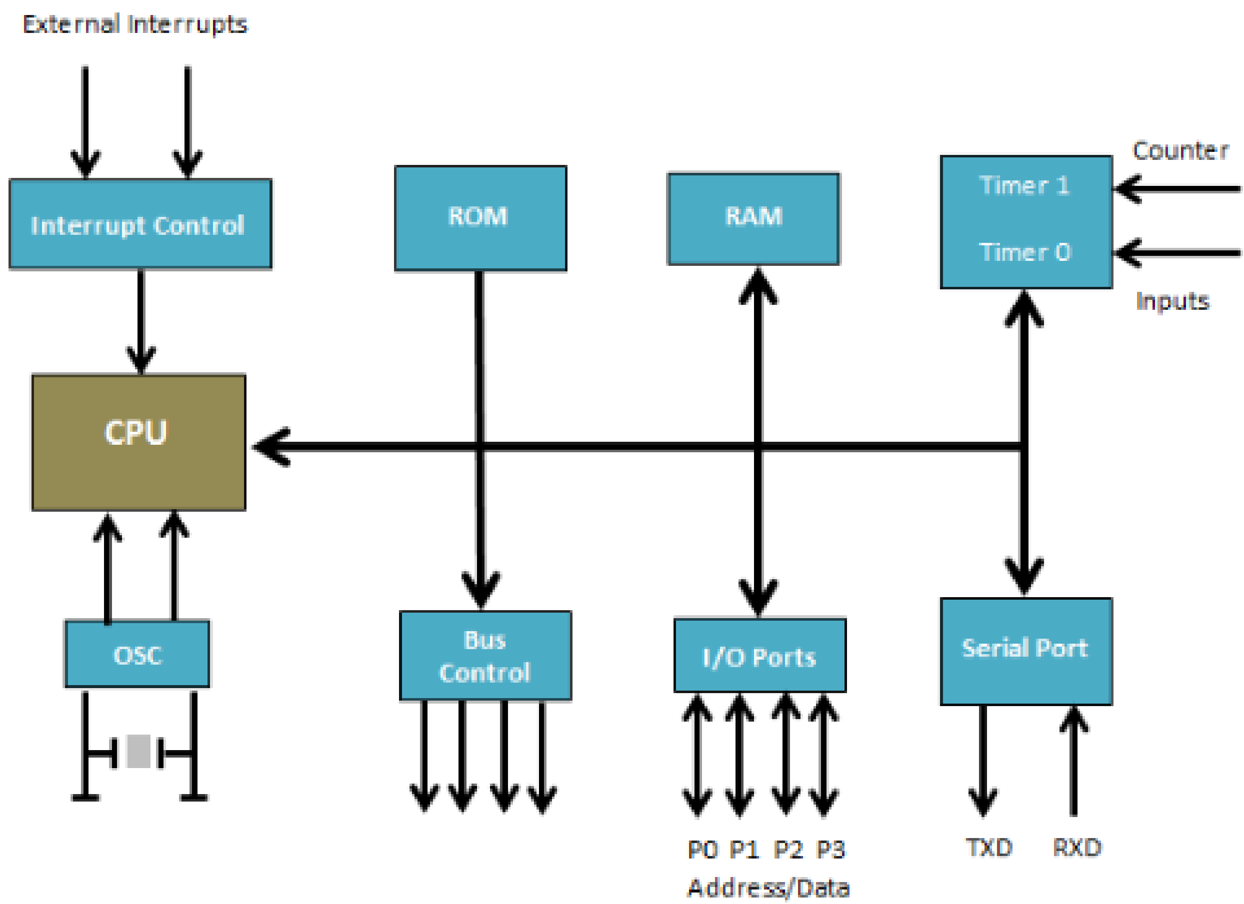

Fig 2.7 Features of a microcontroller

a) CPU/ Processor

The brain of microcontroller is processor. When we provide input through input pins and instructions through programs, processor in microcontroller processes data accordingly which is provided at the output pins.

b) Memory

In order to store all the programs and data, memory chips need to be integrated in a microcontroller. Different types of memory such as RAM (Random Access Memory), ROM (Read Only Memory), EPROM (Erasable Programmable Read Only Memory), EEPROM (Electrically Erasable Programmable Read Only Memory), Flash Memory can be integrated based on the requirement of application.

c) Input/ Output Ports

Input and Outer devices (like sensor, display units, etc) are interfaced with a microcontroller with the help of input and output ports. The number of input output pins may vary depending on the type of microcontroller.

d) Serial Ports

Serial port, a serial communication interface, facilitates microcontrollers serial interface with other peripherals. Through this part information gets transferred in and out one by one bit at a time.

e) Timer

Timers in microcontroller are required for various operations such as count external pulses, pulse generation, oscillation, modulation, etc.

f) ADC and DAC

Most of the microcontrollers are incorporated with inbuilt ADC (Analog to Digital Converter) and DAC (Digital to Analog Converter) to convert data from digital to analog and vice versa.

g) Interrupt Control

Interrupt Control, a kind of notification, interrupts the ongoing process and sends commands to perform the other task that has been defined by interrupt control.

2.6. Existing System

Mahalingam, R. T. Naayagi, N. E. Mastorakis [2] proposed a system where gas leakage is detected by the sensors and fed to the microcontroller so that it can process and produce an audio-video alarm. LEDs and buzzers are interfaced with microcontroller to form alerting mechanism. MQ5 sensor is used to detect leakage of various gases and PIC18F1320 microcontroller is to detect the gas leakage and beep the buzzer when specified exposure limit is exceeded.

HinaRuqsar, Chandana R, Nandini et al. [3] have proposed a system which monitors leakage of constantly with the help of gas sensors. The concentration of different gases in the atmosphere is made available through the internet. Xively IOT platform is used to provide real time sensor data over the Internet. The sensor data will be fed to an account (Twitter or Facebook) through Xively. This proposed system is advantegenous as real time data is made available through real time feed over Internet in addiction to detection of gas leakage.

The paper written by AshishShrivastava, RatneshPrabhaker, etc [4] proposed designing a system that can automatically detect and stop gas leakage in the premises where it has been suspected. GSM module is used to send SMS alert in case gas leakage has been detected. In addition to gas leakage detection, the system alerts user and turns off main power and gas supplies.

AsmitaVarma, Prabhakar S, Kayalvizhi Jayavel [5] have proposed Gas leakage detector system that uses concept of IOT technology for implementing smart alerting techniques like calling user, sending SMS and email to concerned authority. In this system, MQ-2 Sensor is used to detect presence of excess amount of harmful gases such as H2, LPG, CH4, CO, Alcohol, Smoke and Propane. As soon as gas leakage is detected, user will be notified about it through alarms and they will receive call and alert message in their mobile phone and email along with the detail of the area in which gas leakage has taken place. The main power supply of the building is cut off using relays when the concentration of gas is about to reach a lower explosion limit.

Chaitali Bagwe, Vidya Ghadi, Vinayshri Naik, Neha Kunte [7] proposed a system to detect and continuously monitor the leakage of gas and use various alert mechanism to notify the users and concerned authority about the mishap. Leakage is analyzed and data is made available to users via Internet. DHT22 Sensor sends temperature and humidity values. Since the behavior of the gases varies according to temperature and humidity of surrounding area, these vaues will help in making precise system by reducing the chances of false alarms.

AbidKhan, Neju K. Prince, Shailendra Kumar Dewangan , etc al [6] have proposed a system that measures amount of gas that has been consumed with respect to time. If the weight of gas in cylinder reaches a particular threshold level, message is sent by GSM to the booking agency to book a gas cylinder. In this system, MQ5 sensor is used to prevent damage or explosion caused by leakage of LPG. The alerting mechanisms included here are a buzzer, an LED and a SMS sent to the booking agency with the help of GSM. All the connected devices are controlled by a micro controller named 89C51RD2, which is cost efficient and flexible.

Anandhakrishnan S, Deepesh Nair, Rakesh K, Sampath K,Gayathri S Nair [1] have proposed a smart gas monitoring syste. In the proposed system, Gas Sensor (MQ2 Sensor), Load Cell and Infrared Sensor (GP2D120 Sensor) are interfaced to the microcontroller

ATMEGA328. These sensors act as input to the system. The interfaced Wi-Fi module is capable of giving microcontroller the access to available Wi-Fi Network. Gas Sensor detects the leakage of gas in area around it. As soon as leakage of gas is detected, user is notified about it and he/she can turn off the gas valve from anywhere in his/her work place. Load cell continuously monitors the weight of gas in cylinder and on detection of its weight below some critical value, a notification will be sent to booking agency to book a gas cylinder. Object detector sensor detects the presence of vessel over gas burner. If absence of vessel detected over the gas burner beyond a particular time period, buzzer starts beeping and gas wastage is monitored. The response of all these sensors can be viewed in LCD-Display.

2.7 Conceptual Diagram

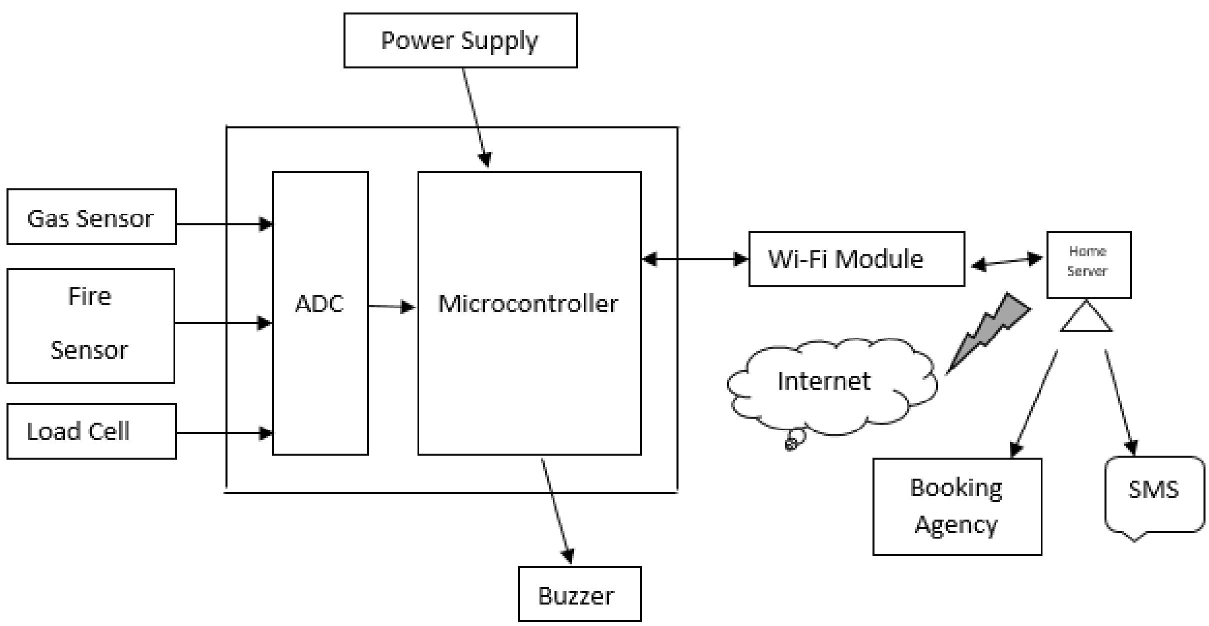

Fig 2.8 Conceptual Diagram of the System

The microcontroller is interfaced with various input and output devices. The input devices interfaced include Gas Sensor, Fire Sensor and Load Cell. Gas Sensor detects gas leakage. Fire Sensor detects fire. Load Cell monitors the weight of gas present in cylinder continuously. The interfaced output devices include LCD Display and buzzer. All the reading of input devices can be displayed in LCD Display. Buzzer starts beeping when fire and gas leakage has been detected. When the weight of gas in cylinder reaches below some threshold value, a notification is sent to booking agency to book a gas cylinder. The entire module is interfaced with WIFI module to provide access to Internet.

3. Methodology

3.1. Proposed Methodology

In the proposed system, Gas Sensor (MQ2 Sensor), Load Cell and Fire Sensor are interfaced to the microcontroller Arduino UNO. These sensors act as input to the system. The interfaced Wi-Fi module is capable of giving microcontroller the access to available Wi-Fi Network. Gas Sensor detects the leakage of gas in area around it. As soon as leakage of gas is detected, user is notified about it and he/she can turn off the gas valve from anywhere in his/her work place. Load cell continuously monitors the weight of gas in cylinder and on detection of its weight below some critical value, a notification will be sent to booking agency to book a gas cylinder. Fire sensor detects fires and flames in the area where it is housed. Buzzer starts beeping on detection of fire and damages due to fire can be minimized. The response of all these sensors can be viewed in LCD-Display.

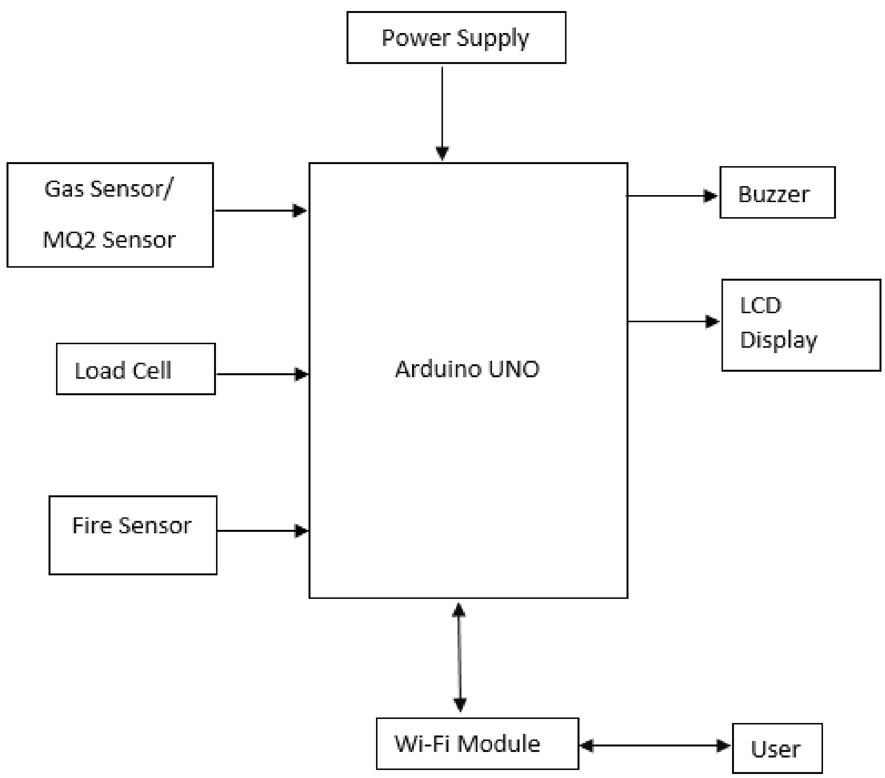

Fig 3.1 Block diagram of the system

3.2. Components Description

a) MQ-2 Sensor

Fig 3.2 MQ-2 Sensor

MQ-2 Gas Sensor is a gas sensor having high sensitivity to LPG, Propane and hydrogen. That's why, it is most widely used for the purpose of gas leakage detection. Potentiometer can be used to adjust its sensitivity. The main features of MQ-2 Sensor are that having high sensitivity and fast response time, measurements can be taken as soon as possible.

The following table summarizes various pins in MQ-2 sensor.

Pin Number	Pin Name
1	Vcc (+5V)
2	Ground
3	Digital out
4	Analog out

Table 3.1 Pin configuration of MQ-2 Sensor

b) Fire Sensor

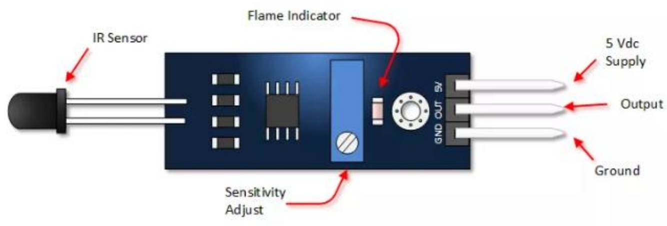

Fig 3.3 Fire Sensor

Fire sensor is the device used to detect presence of flame or fire in the area where it is placed. When the flame has been detected, user can cause several actions to happen based on their requirement by interfacing it to several other components. A buzzer can be made to beep on detection of fire. Similarly, a fuel line can be made to deactivate and fire suppression system can also be activated.

Flame detectors can be divided into three different categories: optical detectors, UV detectors, IR detectors. Optical detectors make use of optical sensors to detect flames. UV detectors can detect flame more quickly as compared to other detectors. UV radiations emitted at the time of ignition are used to detect flames and fires. IR detectors detects flame and fire by monitoring head radiations generated.

The module of fire sensor consists of IR sensor, comparator and LED. It has 3 pins namely gnd, vcc and out. When IR sensor detects fire, LED glows and out pin is set high. The out pin of fire sensor acts as input to the microcontroller. Glowing LED indicates that the area has been caught with fire. The range of the sensor is 2 meter.

The pin configuration of fire sensor is presented in table below.

Pin No.	Pin Name	Details
1	Out	Active high output
2	+5V	Power supply input
3	GND	Power supply ground

Table 3.2 Pin configuration of Fire Sensor

c) Load cell

Fig 3.4 Load cell

Load cell is a type of transducer that converts mechanical force into a measurable electrical output. Anything that needs to be weighted probably uses it for that purpose. Among the various varieties of load cells, strain gauge-based load cells are mostly used. It works on the principle that the strain gauge deforms when the material of the load cells deforms appropriately. It is available in different shapes and sizes. Appropriate one among them can be selected and added to different machinery and weighing equipment. The major features of load cell include its stiffness, good resonance values and tending to have long life cycles in application. It is used in many IOT application as an input device interfaced to microcontroller through Input port.

d) Arduino UNO

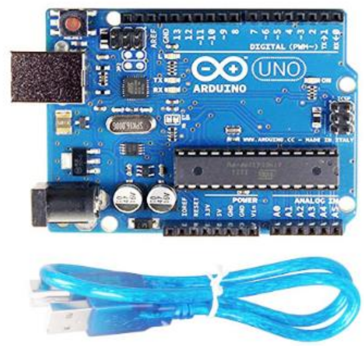

Fig 3.5 Arduino UNO

The Arduino UNO is a microcontroller board which is based on the Microchip ATmega328P microprocessor. This board consists of 14 digital pins and 4 analog pins. It can be programmed with the help of Arduino IDE via a type B USB cable. It can be operated at the voltage of 12V. The technical specifications of Arduino UNO can be summarized in the table given below.

Microcontroller	ATmega328P-8 bit AVR family microcontroller
Operating Voltage, Input Voltage	5 V, 7 to 20 V
Digital I/O Pins	14 (6 of them provide PVM output)
Analog Input Pins	6 (A0-A5)
DC Current per I/O Pin	20 Ma
DC Current for 3.3V Pin	50 Ma
Clock Speed/ Frequency	16 MHz
Weight	25 g
Length, Width	68.6 mm, 53.4 mm
Flash Memory	32 KB (0.5 KB is used for Bootloader)
SRAM	2 KB
EEPROM	1 KB

Table 3.3 Technical Specification of Arduino UNO

The pin configuration of ARDUINO UNO is presented in table below.

Pin Name	Pin Category	Purpose
Vin, 3.3V, 5V, GND	Power	Vin: Input voltage to Arduino 5V: Regulated power supply used to power microcontroller and other connected components 3.3V: 3.3V supply generated by on-board voltage GND: Ground pins
Reset	Reset	Resets the microcontroller
A0-A5	Analog pins	Provide analog input in the range of 0-5 V
Digital pins 0-13	Input/output Pins	Used as input or output pins
0 (Rx), 1(Tx)	Serial	Used to receive and transmit TTL serial data
2,3	External Interrupts	To trigger an interrupt
PWM	3, 5, 6, 9, 11	Provides 8-bit PWM output
SPI	10 (SS), 11(MOSI), 12(MISO), 13 (SCK)	SPI communication
Inbuilt LED	13	To turn on the inbuilt LED
TWI	A4 (SDA), A5 (SCA)	TWI Communication
AREF	AREF	Provide reference voltage for input voltage

Table 3.4 Pin configuration of Arduino UNO

e) **WIFI Module-ESP8266**

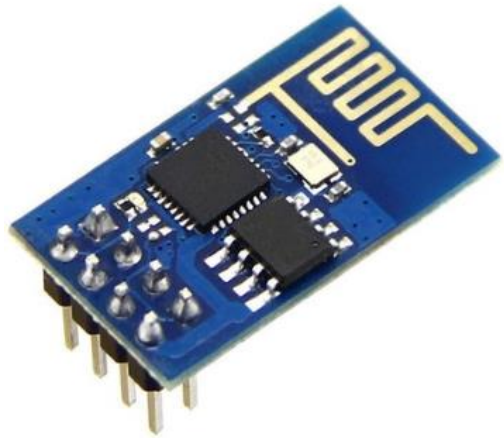

Fig 3.6 WIFI Module-ESP8266

The ESP8266, a low-cost Wi-Fi microchip pre-programmed with an AT command set firmware, is capable of giving any microcontroller access to available WIFI network and making simple TCP/IP connections using Hayes-style commands. It is a self-contained SOC with integrated TCP/IP protocol stack capable of either hosting an application or offloading all Wi-Fi networking functions from another application processor.

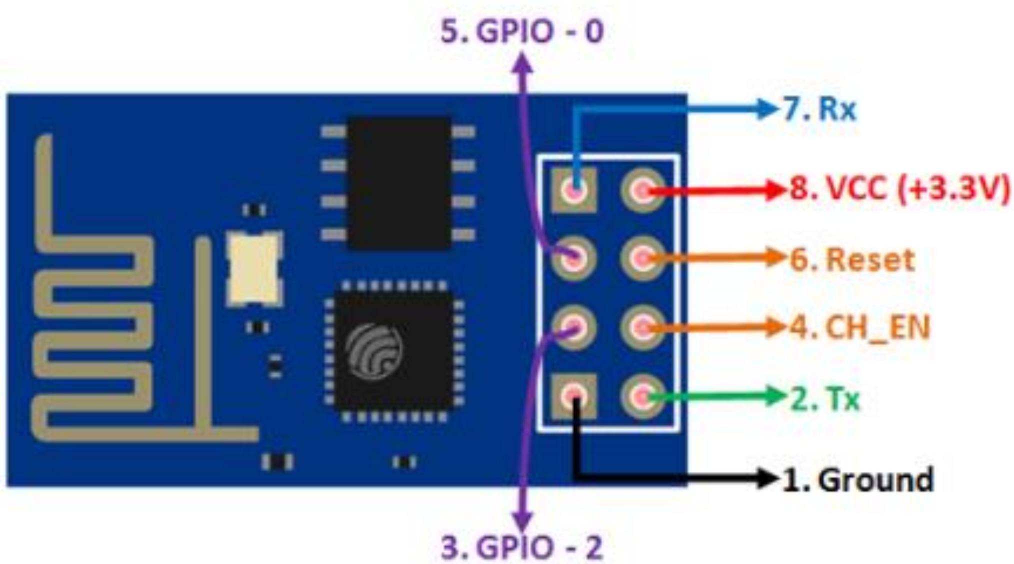

Fig 3.7 Pin configuration of WIFI Module-ESP8266

The pin configuration of WIFI Module-ESP8266 is presented in table below.

Pin Number	Pin Name	Alternative name	Purpose
1	Ground	-	Connected to the ground of the circuit
2	Tx	GPIO-1	Connected to Rx pin of programmer
3	GPIO-2	-	General purpose Input/output pin
4	CH_EN	-	Chip Enable – Active high
5	GPIO-0	Flash	General purpose Input/output pin
6	Reset	-	Resets the module
7	RX	GPIO-3	General purpose Input/output pin
8	Vcc	-	Connect to +3.3V

Table 3.5 Pin configuration of WIFI Module-ESP8266

f) Buzzer

Fig 3.8 Buzzer

Buzzer, often called as beeper, is an audio signaling device that may be either mechanical, electromechanical, or piezoelectric. The most popular applications of buzzer include alarm devices, timers, and confirmation of user input (for e.g. mouse click or keystroke).

It starts beeping when some action has been performed. (Ex: In our proposed topic, the buzzer starts beeping when no vessel is detected over gas burner even after specified time period).

The various features of buzzer include:

- Operating Voltage: 4-8V DC
- Rated Voltage: 6V DC
- Rated current: <30Ma
- Sound Type: Continuous Beep
- Resonant Frequency: ~2300 Hz
- Small and neat sealed package
- Breadboard friendly

The pin configuration of buzzer is presented in table below.

Pin Number	Pin Name	Description
1	Positive	Identified by longer terminal lead and can be powered by 6V DC
2	Negative	Identified by short terminal lead and connected to the ground of the circuit

Table 3.6 Pin configuration of Buzzer

g) LCD Display

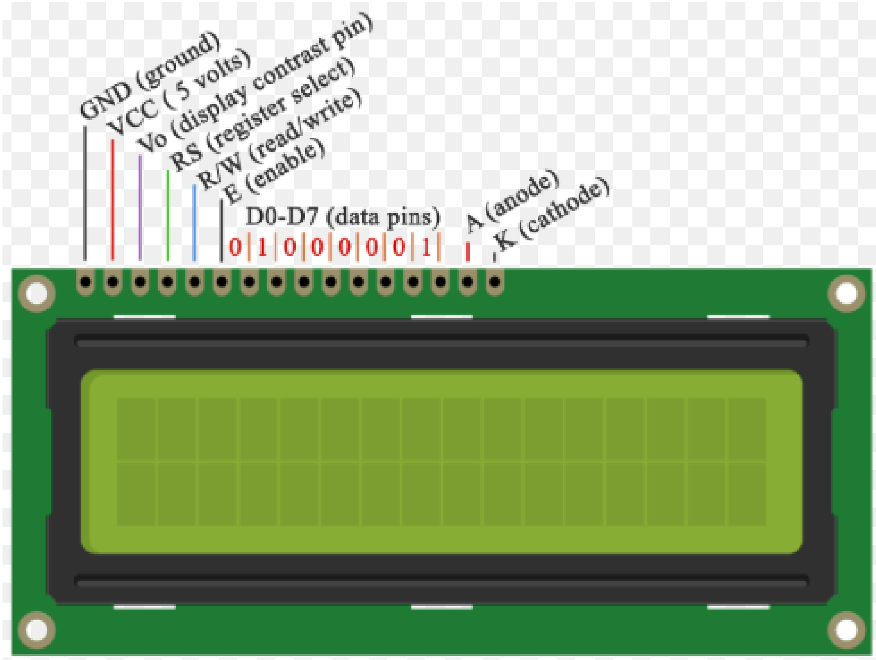

Fig 3.9 LCD Display

LCD Display is used to display the response of different sensors like gas sensor (MQ-2 sensor), infrared sensor and load cell.

The pin configuration of LCD Display is presented in table below.

Pin Number	Pin Name	Purpose
1	Ground	Ground (0V)
2	Vcc	Supply voltage (5V)
3	Vo/VEE	Contrast adjustment
4	RS (Register Select)	Command Register is selected when low and data register is selected when high
5	Read/ Write	Low to write to the register

		High to read from register
6	Enable	Sends data to data pins when a high to low pulse is given
7	DB0	8- bit data pins
8	DB1	8- bit data pins
9	DB2	8- bit data pins
10	DB3	8- bit data pins
11	DB4	8- bit data pins
12	DB5	8- bit data pins
13	DB6	8- bit data pins
14	DB7	8- bit data pins
15	Led+	Backlight VCC (5 V)
16	Led-	Backlight Ground (0V)

Table 3.7 Pin configuration of LCD Display

h) Strain Gauge

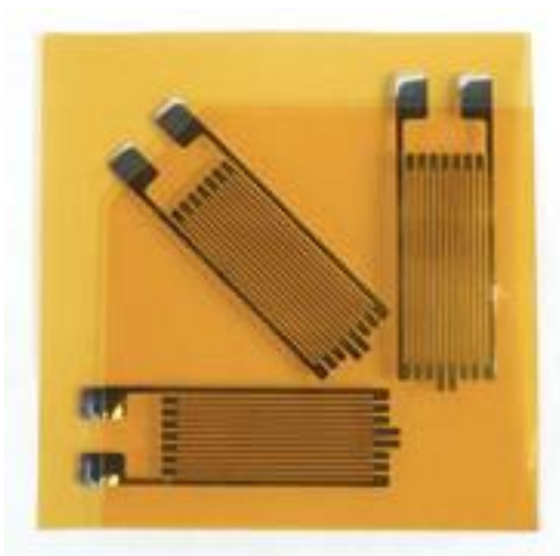

Fig 3.10 Strain Gauge

Strain Gauge, the device that measures strain on an object, is a sensor whose resistance varies with applied force. It has ability to convert force, pressure, tension, weight, etc into a change in electrical resistance which can then be measured.

The shape of the object is deformed due to the external force applied on an object. This deformation in shape which can be compressive or tensile is known as strain and can be measured by the strain gauge. The deformation of an object within the limit of elasticity makes the object either narrower and longer or shorter and broader. This results in change in end-to-end resistance.

The strain gauge is sensitive to small changes in the geometry of an object. The change in resistance of an objects facilitates the calculation of the amount of induced stress. In order to sense small change in resistance, it has a long thin metallic strip arranged in a zigzag pattern on a non-conducting material called the carrier.

When force is applied to any metallic wire, the length increases due to strain. The more is the applied force, the more will be the strain and more will be the increase in length of the wire. As the length of the stretched wire increases, its diameter decreases. The resistance of conductor is the inverse function of the length. So, increase in length of the conductor leads to decrease in its resistance. The change in resistance of the conductor can be measured and calibrated against the applied force. Thus, by using this principle, strain gauge can be used to measure force and related parameters such as displacement and stress.

i) Motor and motor drive

Fig 3.11 Motor

The electric motor is an electro-mechanical device that converts electrical energy into mechanical energy. The devices producing rotational force is known as the motor. It can broadly be classified into two types: AC motor and DC motor. The input for AC motor is alternating current and the input for DC motor is direct current. It is available in different sizes and is operated on main power or batteries. There are many electric motors available in our home such as in fridges, freezers, garage door openers, vacuum cleaners and so on. They are also available in factories for performing various jobs like lifting, pumping, pressing, sucking and drying.

4. Result Analysis

The experimental set up for IOT based Smart Gas Management System consists of input devices such as Gas Sensor, Fire Sensor and Load Cell and output devices like Buzzer and LCD Display interfaced into the microcontroller Arduino UNO. WIFI module is used to provide internet access. Buzzer starts beeping when gas leakage or fire has been detected. SMS is sent to user to notify them about mishap going on in their home. When load cell senses the weight of gas in cylinder is below some threshold value, SMS is sent to booking agency to notify them to book a gas cylinder.

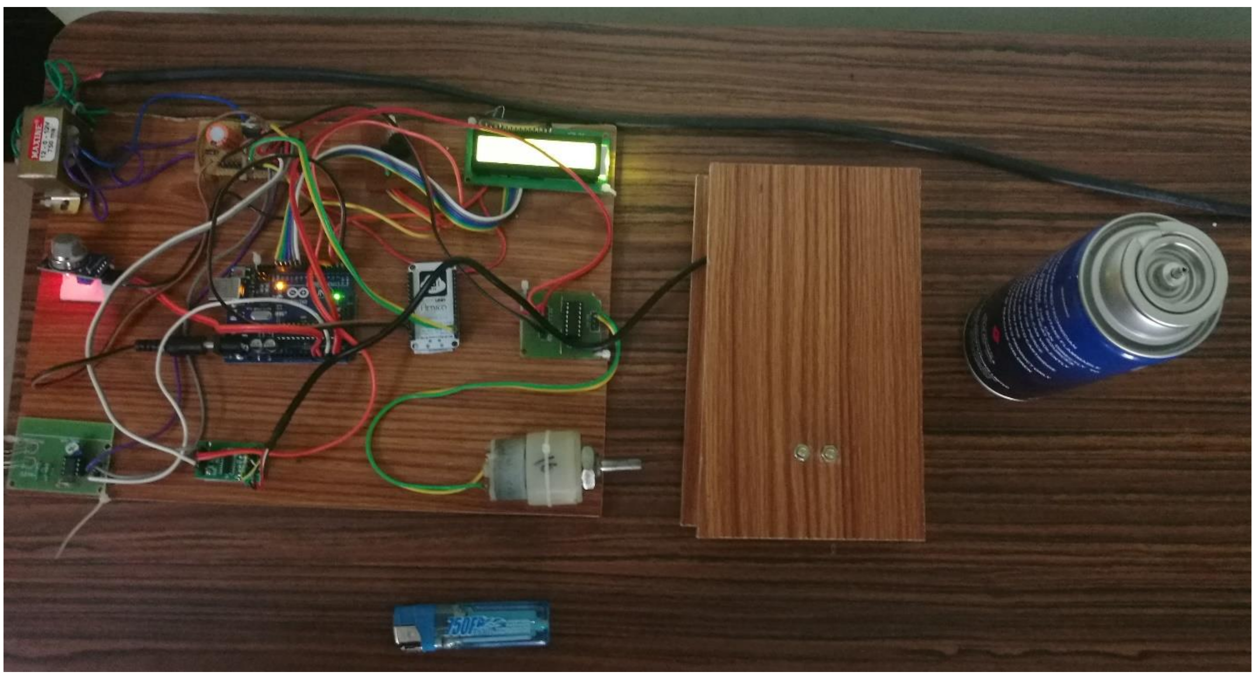

Fig 4.1 Experimental Setup for IOT based Smart Gas Management System

When the experiment is initiated, the LCD Display shows initial reading of LPG, weight and fire respectively.

Fig 4.2 Initial reading in LCD Display

The microcontroller requires 12 V of power supply to operate and remaining components require 5V to operate.

On detection of gas leakage, fire and low weight of gas cylinder, SMS is sent to user stating the mishap taking place in their building.

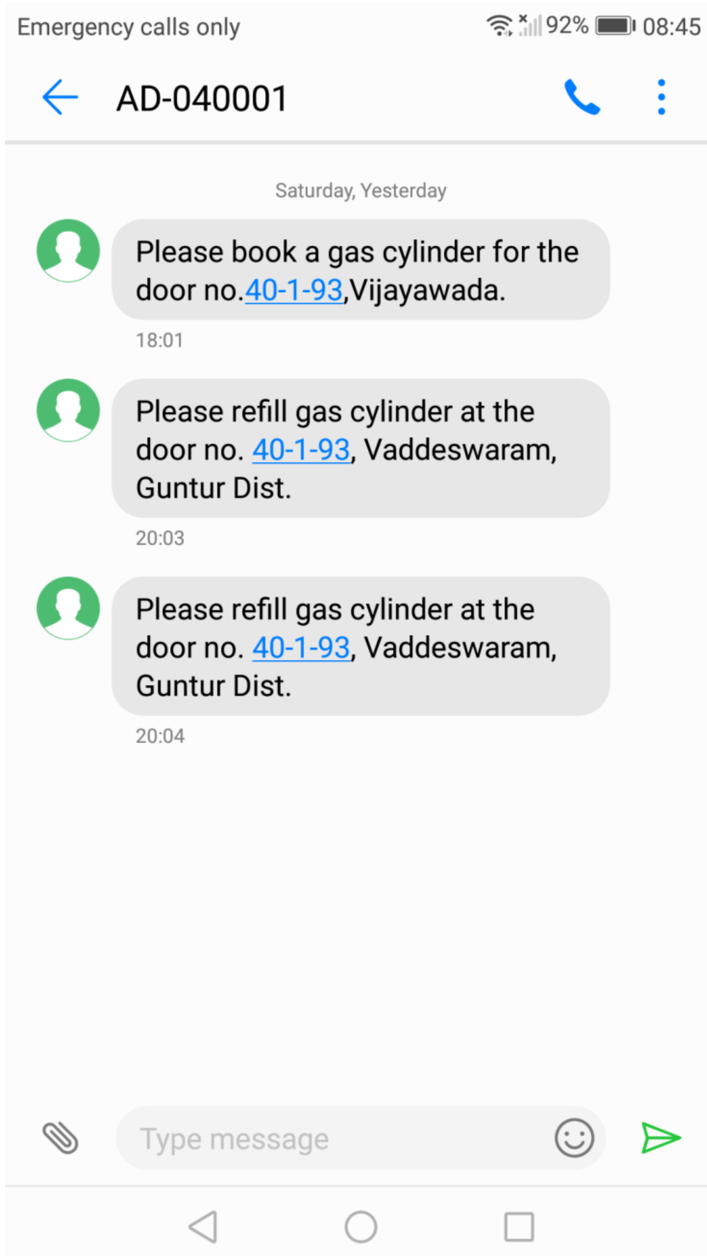

Fig 4.3 SMS alert for booking of gas cylinder

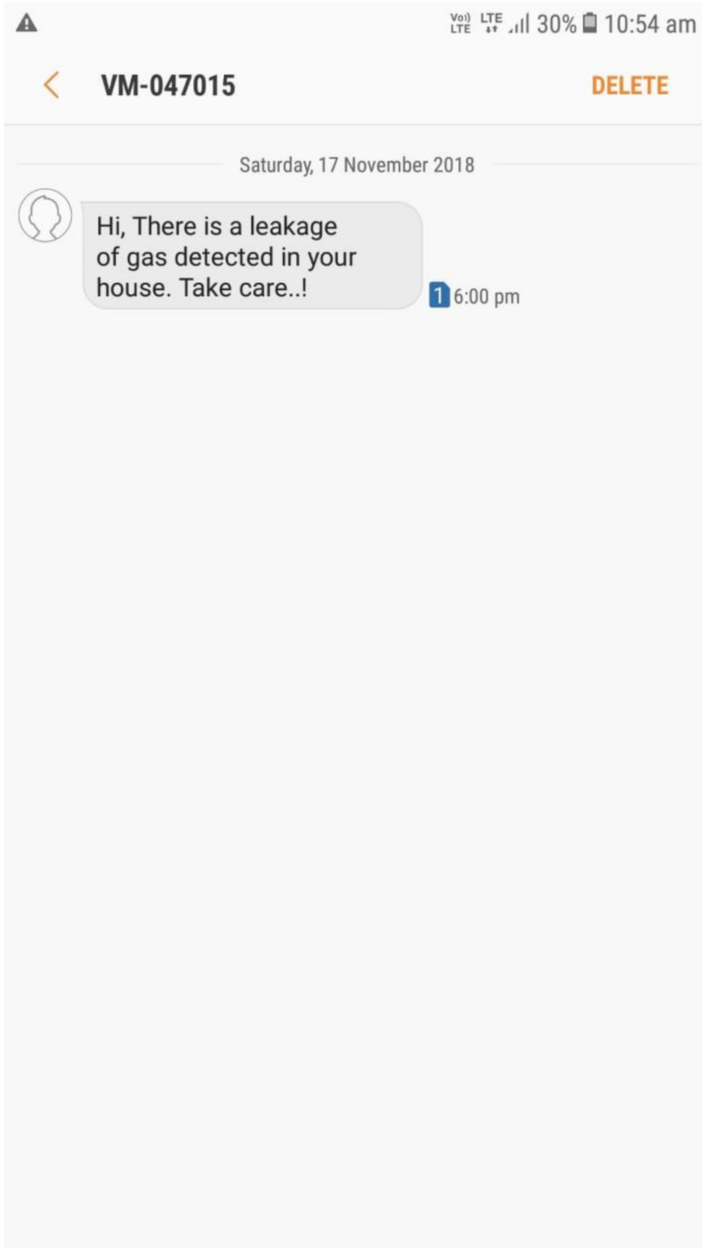

Fig 4.4 SMS alert for detection of fire

The responses of different sensors can be viewed in LCD Display. The graph below shows weight of gas in cylinder.

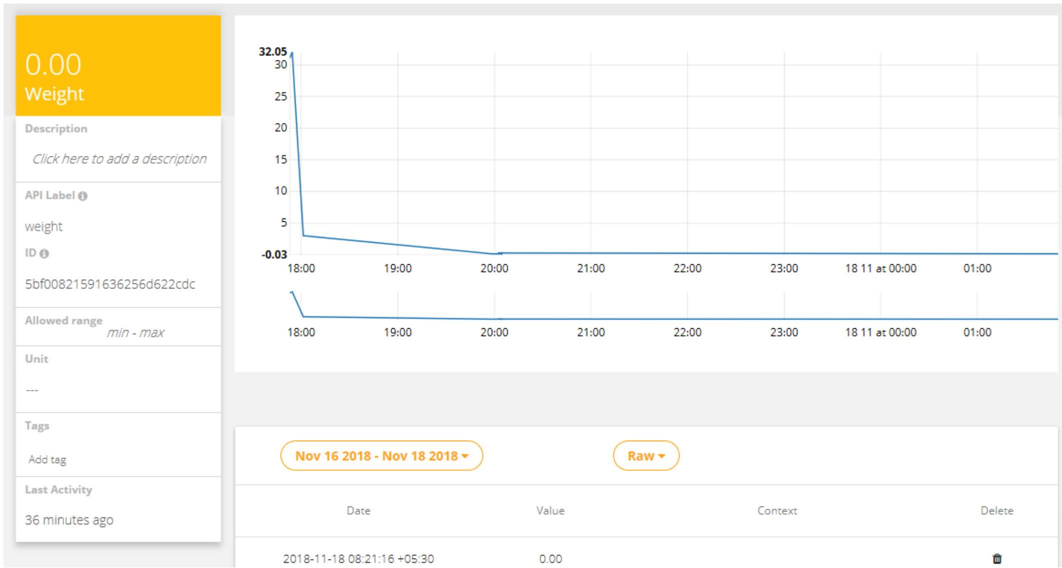

Fig 4.5 Graphical Representation of reading of weight of gas in cylinder

The system also detects the leakage of gas. The graph below shows concentration of LPG Gas in its surrounding.

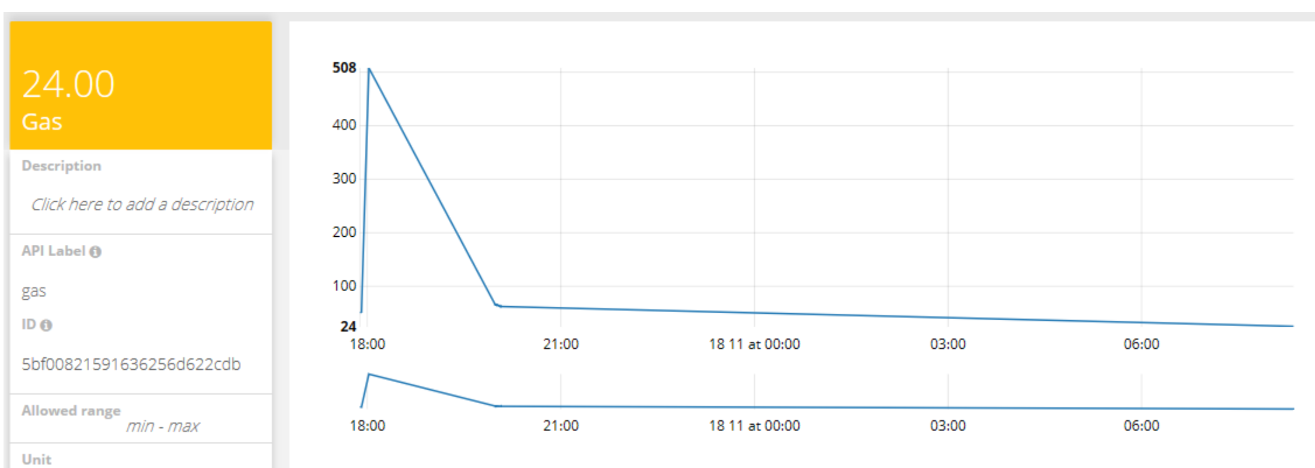

Fig 4.6 Graphical Representation of reading for fire detection

Corresponding readings at different instance of time can be obtained in our Desktop.

Nov 16 2018 - Nov 18 2018 ▾ Raw ▾

Date	Value	Context	Delete
2018-11-18 08:21:16 +05:30	0.00		
2018-11-18 08:21:10 +05:30	0.00		
2018-11-17 20:05:29 +05:30	0.27		
2018-11-17 20:05:25 +05:30	0.27		
2018-11-17 20:05:22 +05:30	0.27		
2018-11-17 20:05:19 +05:30	0.27		
2018-11-17 20:05:15 +05:30	0.27		
2018-11-17 20:05:12 +05:30	0.27		
2018-11-17 20:05:09 +05:30	0.27		
2018-11-17 20:05:05 +05:30	0.27		
2018-11-17 20:05:02 +05:30	0.27		

Fig 4.7 Reading of weight of gas in cylinder

Nov 16 2018 - Nov 18 2018 ▾ Raw ▾

Date	Value	Context	Delete
2018-11-18 08:21:16 +05:30	24.00		
2018-11-18 08:21:10 +05:30	25.00		
2018-11-17 20:05:29 +05:30	62.00		
2018-11-17 20:05:25 +05:30	62.00		
2018-11-17 20:05:22 +05:30	62.00		
2018-11-17 20:05:19 +05:30	62.00		
2018-11-17 20:05:15 +05:30	62.00		
2018-11-17 20:05:12 +05:30	62.00		
2018-11-17 20:05:09 +05:30	62.00		
2018-11-17 20:05:05 +05:30	62.00		

Fig 4.8 Reading for fire detection

We can add events such as sending SMS, email, telegram to notify user about gas leakage, fire and status of gas cylinder by filling parameters mentioned in following form.

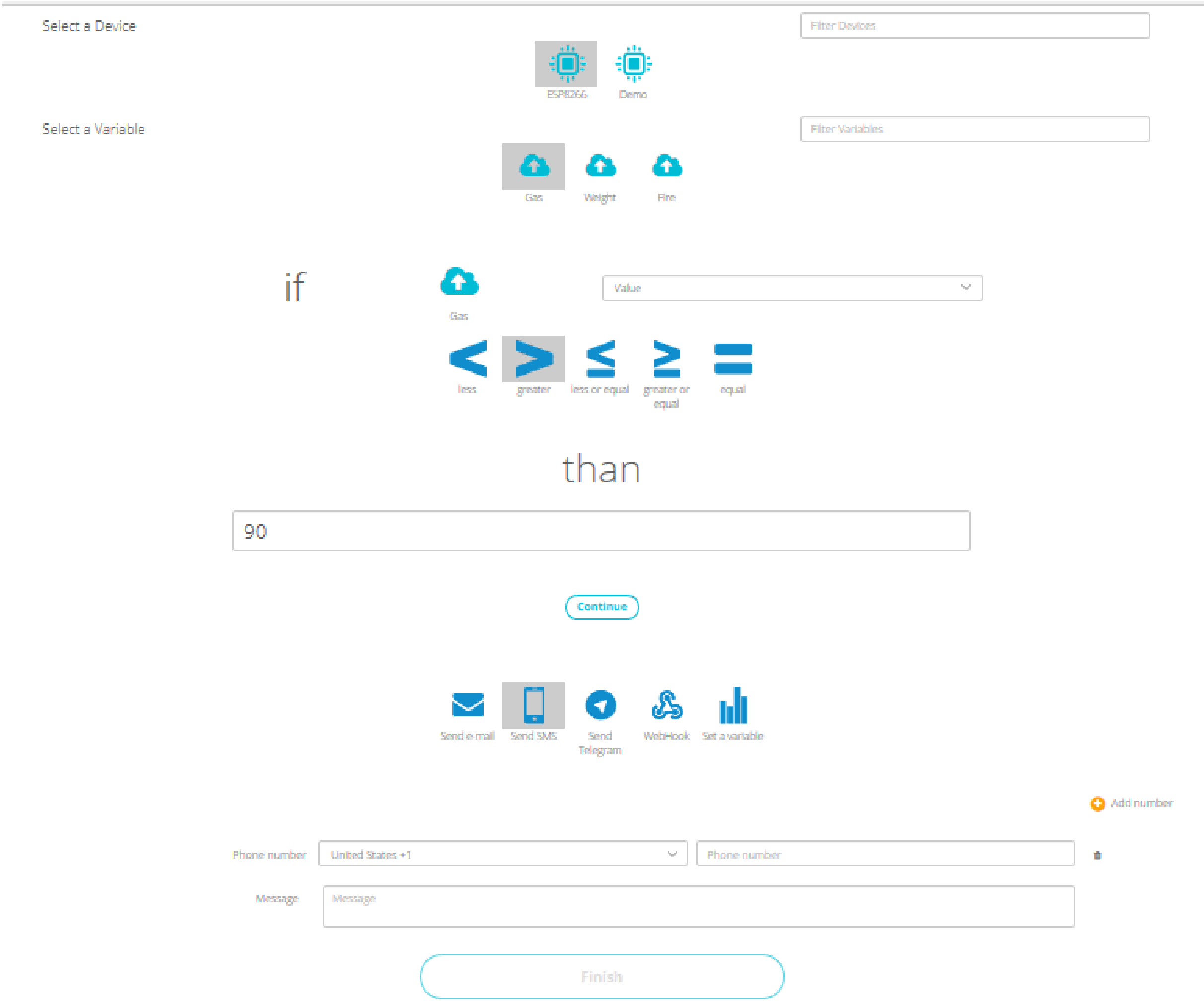

Fig 4.9 Adding events for gas leakage, fire detection and gas weight

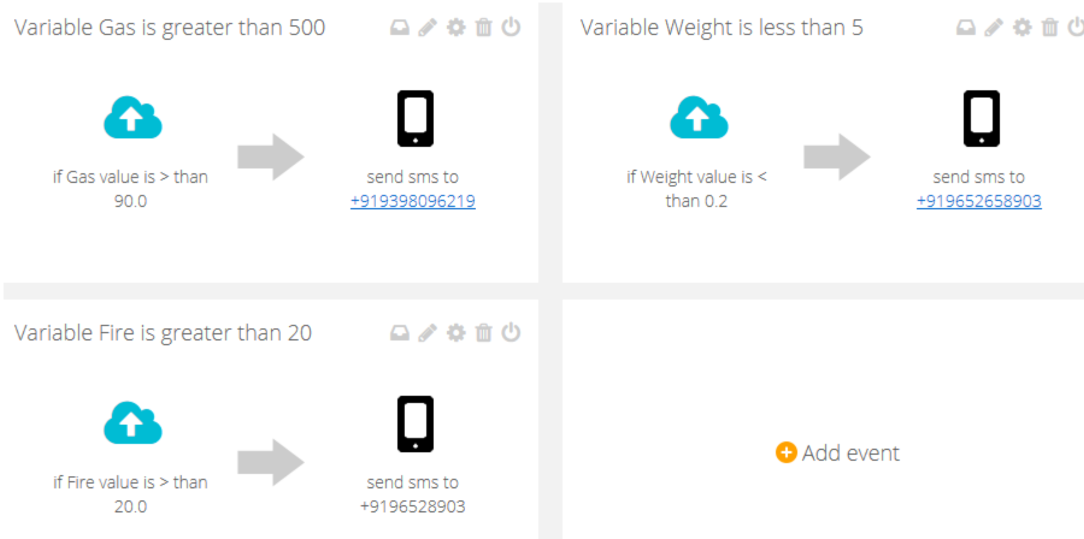

Fig 4.10 Events added for gas leakage, fire detection and gas weight

Conclusion

As the use of LPG gas is increasing day-by-day, the risk of its leakage and damages caused by this leakage is also increasing in same ratio. IOT based Smart Gas Management system monitors leakage and wastage of gas. In either cases, alert message will be sent to the house owner so that he/she can turn off gas valve on time before much damages are caused by leakage. It also detects fire in the area where it is housed and alerts people about it in case it occurs.

This system sends notification to gas cylinder booking agency in case it is found that the level of gas in cylinder reaches below some threshold value. The result of every module can be viewed in LCD display. Buzzer starts beeping if leakage or wastage of gas has been detected. Thus, the damages caused due to gas leakage and can be minimized by use of this system.

In this regard, IOT based Gas Management System has three different modules. First of all, it makes use of Gas Sensor to detect the leakage of gas. Secondly, Fire Sensor is used to detect flames and fire. Thirdly, Load Cell is used to monitor the level of gas in cylinder continuously. All these are considered to be input devices which are interfaced to microcontroller through input pins. Buzzer starts beeping if gas leakage or fire has been detected. In either case, SMS is sent to the user to alert about wrong happening in their building. When the weight of gas in cylinder reaches below some threshold value, message is sent to booking agency to confirm booking of gas cylinder in mentioned area. In this way, IOT based Gas Management System helps in achieving objective of Smart Kitchen by using the concept of IOT.

References

[1] Anandhakrishnan S, Deepesh Nair, Rakesh K, Sampath K, Gayathri S Nair-- “IOT Based Smart Gas Monitoring System”, IOSR Journal of Electrical and Electronics Engineering (IOSR-JEEE), e-ISSN: 2278-1676, p-ISSN: 2320-3331, PP 82-87, www.iosrjournals.org

[2] A.Mahalingam, R.T. Naayagi,N.E. Mastoralis “Design and Implementation of an Economic Gas Leakage Detector” ACA'12 Proceedings of the 11th international conference on Applications of Electrical and Computer Engineering

[3] HinaRuqsar, Chandana R, Nandini R, Dr. T P Surekha “Internet of Things (IOT) based Real Time Gas Leakage Monitoring and Controlling” Proceedings of the 2nd International Conference on Current Trends in Engineering and Management ICCTEM -2014 17 – 19, July 2014, Mysore, Karnataka, India.

[4] AshishShrivastava, RatneshPrabhaker, Rajeev Kumar and Rahul Verma “GSM based Gas leakage detection System” International Journal of Technical Research and Applications e-ISSN: 2320-8163, www.ijtra.com Volume 1, Issue 2 (may-june 2013)

[5] AsmitaVarma, Prabhakar S, KayalvizhiJayavel “Gas Leakage Detection and Smart Alerting and Prediction Using IoT” 2017 Second International Conference On Computing and Communications Technologies(ICCCT’17).

[6] AbidKhan, Neju K. Prince, Shailendra Kumar Dewangan, Praveen Singh Rathore “GSM Based Automatic LPG Ordering System with Leakage Alert” International Journal of Research in Engineering and Technology(IJRET).

[7] Chaitali Bagwe, Vidya Ghadi, Vinayshri Naik, Neha Kunte4- “IOT Based Gas Leakage Detection System with Database Logging, Prediction and Smart Alerting”, IOSR Journal of Engineering (IOSRJEN) ISSN (e): 2250-3021, ISSN (p): 2278-8719 Volume 1, PP 25-28, www.iosrjen.org

[8] Kumar Keshamoni, Sabbani Hemanth-- “Smart Gas Level Monitoring, Booking & Gas Leakage Detector over IoT”, 2017 IEEE 7th International Advance Computing Conference

[9] Mr. Mahesh S. Kholgade, Ms. Puja S. Dukare, Ms. Vaishanavi R. Deshmukh, Ms. Sneha V. Pathak, Ms. Minal P. Mahajan-- “LPG Leakage Detection and Control System by Using Microcontroller”, International Journal of Research in Advent Technology (IJRAT), E-ISSN: 2321-9637, Special Issue National Conference “CONVERGENCE 2017”, 09th April 2017

[10] Asmita Varma, Prabhakar S, Kayalvizhi Jayavel-- “Gas Leakage Detection and Smart Alerting and Prediction Using IoT”, 2017 IEEE

[11] Rohan Chandra Pandey, Manish Verma, Lumesh Kumar Sahu-- “Internet of Things (IOT) Based Gas Leakage Monitoring and Alerting System with MQ-2 Sensor”, 2017 IJEDR, Volume 5, Issue 2, ISSN: 2321-9939

[12] O. Mohana Chandrika & B. Alekya Hima Bindu-- “Automatic Gas Alerting System”, Imperial Journal of Interdisciplinary Research (IJIR) Vol-2, Issue-6, 2016 ISSN: 2454-1362, http://www.onlinejournal.in

[13]. D. Surie, O. Laguionie, T. Pederson, ―” Wireless sensor networking of everyday objects in a smart home environment‖, Proceedings of the International Conference on Intelligent Sensors”, Sensor Networks and Information Processing- ISSNIP- 2008, pp. 189 – 194.

[14] J. Tsado, O. Imoru, S.O. Olayemi, ― “Design and construction of a GSM based gas leak Alert system” ‖, IEEE Transaction, IRJEEE Vol. 1(1), pp. 002-006, September, 2014.

[15] M. Eisenhauer, P. Rosengren, P. Antolin, ― “A Development Platform for Integrating Wireless Devices and Sensors into Ambient Intelligence Systems”, pp.1-3.

[16] http://www.nodemcu.com/index_en.html

[17] https://www.engineersgarage.com/articles/gsm-gprs-modules

[18] http://whatis.techtarget.com/definition/light-emitting-diode-LED

[16] http://dataaspirant.com/2017/02/06/naive-bayes-classifier-machine-learning/

Appendix

Appendix 1: Source Code for Microcontroller (Arduino UNO)

```
#include<LiquidCrystal.h>

float f1;

int buz=5,gas=A1,fire=A2,m1=A4,m2=A5;

//RS=8,RW=Gnd,EN=9,D4=0,D5=1,D6=2,D7=3;

LiquidCrystal lcd(13,12,11,10,9,8);

//-----------Device-1-----------------//

byte PD_SCK;  // Power Down and Serial Clock Input Pin

byte DOUT;    // Serial Data Output Pin

byte GAIN;    // amplification factor

long OFFSET = 0;  // used for tare weight

float SCALE = 1;  // used to return weight in grams, kg, ounces, whatever

void setup()

{

  lcd.begin(16, 2);

  Serial.begin(9600);

  pinMode(buz,OUTPUT);

  pinMode(m1,OUTPUT);

  pinMode(m2,OUTPUT);

  //pinMode(gas,INPUT);

  pinMode(fire,INPUT);
```

```
  delay(500);

  lcd.setCursor(0,0);  //lcd.setCursor(column,line) starts with 0

  lcd.print(" IOT BASED GAS  ");

  lcd.setCursor(0,1);

  lcd.print(" MANAGEMENT SYS ");

  delay(500);

  //------------------ HX711 ----------------

  scale1_begin(2,3,128);  //(Dt,Sck,Gain)

  Serial.println("Before setting up the scale:");

  Serial.print("read: \t\t");

  Serial.println(scale1_read());     // print a raw reading from the ADC

  Serial.print("read average: \t\t");

  Serial.println(scale1_read_average(20));   // print the average of 20 readings from the
ADC

  Serial.print("get value: \t\t");

  Serial.println(scale1_get_value(5));   // print the average of 5 readings from the ADC
minus the tare weight (not set yet)

  Serial.print("get units: \t\t");

  Serial.println(scale1_get_units(5), 1);  // print the average of 5 readings from the ADC
minus tare weight (not set) divided

                                            // by the SCALE parameter (not set yet)

  scale1_set_scale(-110010);    // this value is obtained by calibrating the scale with
known weights; see the README for details

  scale1_tare(5);               // reset the scale to 0
```

```
  Serial.println("After setting up the scale:");

  Serial.print("read: \t\t");

  Serial.println(scale1_read());                // print a raw reading from the ADC

  Serial.print("read average: \t\t");

  Serial.println(scale1_read_average(20));      // print the average of 20 readings from the
ADC

  Serial.print("get value: \t\t");

  Serial.println(scale1_get_value(5));   // print the average of 5 readings from the ADC
minus the tare weight, set with tare()

  Serial.print("get units: \t\t");

  Serial.println(scale1_get_units(5), 1);       // print the average of 5 readings from the
ADC minus tare weight, divided

                                                // by the SCALE parameter set with set_scale

  lcd.setCursor(0,0);  //lcd.setCursor(column,line) starts with 0

  lcd.print("LPG : WEIG: FIRE");

  lcd.setCursor(0,1);

  lcd.print("    :     :     ");

  delay(1000);

}

void loop()

{

  //----- Read weights -----

  int f = digitalRead(fire);
```

```
float g = analogRead(gas);

f1=scale1_get_units(5);

lcd.setCursor(0,1);

lcd.print("    :     :     ");

lcd.setCursor(0,1);

lcd.print(g);

lcd.setCursor(5,1);

lcd.print(f1);

lcd.setCursor(12,1);

lcd.print(f);

Serial.println("*"+String(g)+"/"+String(f)+"/"+String(f1)+"#");

if(g>500 || f==1)

{

  digitalWrite(buz,HIGH);

}

else

{

  digitalWrite(buz,LOW);

}

scale1_power_down();              // put the ADC in sleep mode

delay(1000);

scale1_power_up();

delay(2000);
```

```
}
//*****************************************************************
//                Device-1 Functions
//*****************************************************************
void scale1_begin(byte dout, byte pd_sck, byte gain)
{
  PD_SCK = pd_sck;
  DOUT = dout;

  pinMode(PD_SCK, OUTPUT);
  pinMode(DOUT, INPUT);
  scale1_set_gain(gain);
}
bool scale1_is_ready()
{
  return digitalRead(DOUT) == LOW;
}
void scale1_set_gain(byte gain)
{
  switch (gain) {
    case 128:   // channel A, gain factor 128
      GAIN = 1;
      break;
```

```
    case 64:    // channel A, gain factor 64
      GAIN = 3;
      break;
    case 32:    // channel B, gain factor 32
      GAIN = 2;
      break;
  }
  digitalWrite(PD_SCK, LOW);
  scale1_read();
}
void scale1_yield(void) {};
long scale1_read()
{
  // wait for the chip to become ready
  while (!scale1_is_ready()) {
    // Will do nothing on Arduino but prevent resets of ESP8266 (Watchdog Issue)
    scale1_yield();
  }

  unsigned long value = 0;
  uint8_t data[3] = { 0 };
  uint8_t filler = 0x00;
```

```
  // pulse the clock pin 24 times to read the data
  data[2] = shiftIn(DOUT, PD_SCK, MSBFIRST);
  data[1] = shiftIn(DOUT, PD_SCK, MSBFIRST);
  data[0] = shiftIn(DOUT, PD_SCK, MSBFIRST);
  // set the channel and the gain factor for the next reading using the clock pin
  for (unsigned int i = 0; i < GAIN; i++) {
    digitalWrite(PD_SCK, HIGH);
    digitalWrite(PD_SCK, LOW);
  }
  // Replicate the most significant bit to pad out a 32-bit signed integer
  if (data[2] & 0x80) {
    filler = 0xFF;
  } else {
    filler = 0x00;
  }
  // Construct a 32-bit signed integer
  value = ( static_cast<unsigned long>(filler) << 24
      | static_cast<unsigned long>(data[2]) << 16
      | static_cast<unsigned long>(data[1]) << 8
      | static_cast<unsigned long>(data[0]) );
  return static_cast<long>(value);
}
long scale1_read_average(byte times)
```

```
{
  long sum = 0;
  for (byte i = 0; i < times; i++) {
    sum += scale1_read();
    scale1_yield();
  }
  return sum / times;
}
double scale1_get_value(byte times)
{
  return scale1_read_average(times) - OFFSET;
}
float scale1_get_units(byte times)
{
  return scale1_get_value(times) / SCALE;
}
void scale1_tare(byte times)
{
  double sum = scale1_read_average(times);
  scale1_set_offset(sum);
}
void scale1_set_scale(float scale)
{
```

```
  SCALE = scale;
}
float scale1_get_scale()
{
  return SCALE;
}
void scale1_set_offset(long offset)
{
  OFFSET = offset;
}
long scale1_get_offset()
{
  return OFFSET;
}
void scale1_power_down()
{
  digitalWrite(PD_SCK, LOW);
  digitalWrite(PD_SCK, HIGH);
}
void scale1_power_up()
{
  digitalWrite(PD_SCK, LOW);
```

Appendix 2: Source Code for WIFI Module-ESP8266

```
#include<LiquidCrystal.h>

float f1;

int buz=5,gas=A1,fire=A2,m1=A4,m2=A5;

//RS=8,RW=Gnd,EN=9,D4=0,D5=1,D6=2,D7=3;

LiquidCrystal lcd(13,12,11,10,9,8);

//-----------Device-1-----------------//

byte PD_SCK;  // Power Down and Serial Clock Input Pin

byte DOUT;    // Serial Data Output Pin

byte GAIN;    // amplification factor

long OFFSET = 0;  // used for tare weight

float SCALE = 1;  // used to return weight in grams, kg, ounces, whatever

void setup()

{

  lcd.begin(16, 2);

  Serial.begin(9600);

  pinMode(buz,OUTPUT);

  pinMode(m1,OUTPUT);

  pinMode(m2,OUTPUT);

  //pinMode(gas,INPUT);

  pinMode(fire,INPUT);

  delay(500);
```

```
  lcd.setCursor(0,0);  //lcd.setCursor(column,line) starts with 0

  lcd.print(" IOT BASED GAS  ");

  lcd.setCursor(0,1);

  lcd.print(" MANAGEMENT SYS ");

  delay(500);

  //------------------- HX711 -----------------

  scale1_begin(2,3,128);  //(Dt,Sck,Gain)

  Serial.println("Before setting up the scale:");

  Serial.print("read: \t\t");

  Serial.println(scale1_read());     // print a raw reading from the ADC

  Serial.print("read average: \t\t");

  Serial.println(scale1_read_average(20));   // print the average of 20 readings from the
ADC

  Serial.print("get value: \t\t");

  Serial.println(scale1_get_value(5));   // print the average of 5 readings from the ADC
minus the tare weight (not set yet)

  Serial.print("get units: \t\t");

  Serial.println(scale1_get_units(5), 1);  // print the average of 5 readings from the ADC
minus tare weight (not set) divided

                                            // by the SCALE parameter (not set yet)

  scale1_set_scale(-110010);    // this value is obtained by calibrating the scale with
known weights; see the README for details

  scale1_tare(5);               // reset the scale to 0

  Serial.println("After setting up the scale:");
```

```
  Serial.print("read: \t\t");
  Serial.println(scale1_read());                 // print a raw reading from the ADC
  Serial.print("read average: \t\t");
  Serial.println(scale1_read_average(20));       // print the average of 20 readings from the ADC
  Serial.print("get value: \t\t");
  Serial.println(scale1_get_value(5));   // print the average of 5 readings from the ADC minus the tare weight, set with tare()
  Serial.print("get units: \t\t");
  Serial.println(scale1_get_units(5), 1);        // print the average of 5 readings from the ADC minus tare weight, divided
                                                 // by the SCALE parameter set with set_scale
  lcd.setCursor(0,0);  //lcd.setCursor(column,line) starts with 0
  lcd.print("LPG : WEIG: FIRE");
  lcd.setCursor(0,1);
  lcd.print("    :     :     ");
  delay(1000);
}
void loop()
{
  //----- Read weights -----
  int f = digitalRead(fire);
  float g = analogRead(gas);
```

```
  f1=scale1_get_units(5);
  lcd.setCursor(0,1);
  lcd.print("    :     :     ");
  lcd.setCursor(0,1);
  lcd.print(g);
  lcd.setCursor(5,1);
  lcd.print(f1);
  lcd.setCursor(12,1);
  lcd.print(f);
  Serial.println("*"+String(g)+"/"+String(f)+"/"+String(f1)+"#");
  if(g>500 || f==1)
  {
    digitalWrite(buz,HIGH);
  }
  else
  {
    digitalWrite(buz,LOW);
  }
  scale1_power_down();              // put the ADC in sleep mode
  delay(1000);
  scale1_power_up();
  delay(2000);
}
```

```
//*********************************************************

//              Device-1 Functions

//*********************************************************

void scale1_begin(byte dout, byte pd_sck, byte gain)

{

  PD_SCK = pd_sck;

  DOUT = dout;



  pinMode(PD_SCK, OUTPUT);

  pinMode(DOUT, INPUT);



  scale1_set_gain(gain);

}

bool scale1_is_ready()

{

  return digitalRead(DOUT) == LOW;

}

void scale1_set_gain(byte gain)

{

  switch (gain) {

    case 128:   // channel A, gain factor 128

      GAIN = 1;

      break;
```

```
    case 64:    // channel A, gain factor 64
      GAIN = 3;
      break;
    case 32:    // channel B, gain factor 32
      GAIN = 2;
      break;
  }

  digitalWrite(PD_SCK, LOW);
  scale1_read();
}
void scale1_yield(void) {};
long scale1_read()
{
  // wait for the chip to become ready
  while (!scale1_is_ready()) {
    // Will do nothing on Arduino but prevent resets of ESP8266 (Watchdog Issue)
    scale1_yield();
  }

  unsigned long value = 0;
  uint8_t data[3] = { 0 };
  uint8_t filler = 0x00;
```

```
// pulse the clock pin 24 times to read the data
data[2] = shiftIn(DOUT, PD_SCK, MSBFIRST);
data[1] = shiftIn(DOUT, PD_SCK, MSBFIRST);
data[0] = shiftIn(DOUT, PD_SCK, MSBFIRST);

// set the channel and the gain factor for the next reading using the clock pin
for (unsigned int i = 0; i < GAIN; i++) {
  digitalWrite(PD_SCK, HIGH);
  digitalWrite(PD_SCK, LOW);
}

// Replicate the most significant bit to pad out a 32-bit signed integer
if (data[2] & 0x80) {
  filler = 0xFF;
} else {
  filler = 0x00;
}

// Construct a 32-bit signed integer
value = ( static_cast<unsigned long>(filler) << 24
    | static_cast<unsigned long>(data[2]) << 16
    | static_cast<unsigned long>(data[1]) << 8
```

```
      | static_cast<unsigned long>(data[0]) );


  return static_cast<long>(value);
}
long scale1_read_average(byte times)
{
  long sum = 0;
  for (byte i = 0; i < times; i++) {
    sum += scale1_read();
    scale1_yield();
  }
  return sum / times;
}
double scale1_get_value(byte times)
{
  return scale1_read_average(times) - OFFSET;
}
float scale1_get_units(byte times)
{
  return scale1_get_value(times) / SCALE;
}
void scale1_tare(byte times)
{
```

```
  double sum = scale1_read_average(times);
  scale1_set_offset(sum);
}
void scale1_set_scale(float scale)
{
  SCALE = scale;
}
float scale1_get_scale()
{
  return SCALE;
}
void scale1_set_offset(long offset)
{
  OFFSET = offset;
}
long scale1_get_offset()
{
  return OFFSET;
}
void scale1_power_down()
{
  digitalWrite(PD_SCK, LOW);
  digitalWrite(PD_SCK, HIGH);
```

```
}

void scale1_power_up()

{

  digitalWrite(PD_SCK, LOW);

}
```