An Automated Fire Suppression Mechanism Controlled Using an Arduino

A Thesis Submitted to the Department of Electrical and Electronic Engineering of **BRAC University**

By

**Rafiz Ibne Rashid- 12321052**
**Syed Muntaha Rafid- 13121037**

**Supervised by:**

**Dr. A. K. M. Abdul Malek Azad**

Professor
Department of Electrical and Electronic Engineering
BRAC University, Dhaka.

In partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Electronic Engineering

Fall 2017
BRAC University, Dhaka.
DECLARATION

We do hereby declare that the thesis titled “An Automated Fire suppression mechanism controlled using an Arduino”, a thesis submitted to the Department of Electrical and Electronics Engineering of BRAC University in partial fulfillment of the Bachelors of Engineering in Electrical and Electronics Engineering. This is our original work and was not submitted elsewhere for the award of any other degree or any other publication.

Signature of Supervisor: ..............................................
Signature of Authors: ..............................................

Dr. A. K. M. Abdul Malek Azad
Professor
Dept. of Electrical and Electronic Engineering
BRAC University, Dhaka.

Rafiz Ibne Rashid

Syed Muntaha Rafid
ACKNOWLEDGEMENT

We are thankful to our thesis supervisor Dr. A.K.M. Abdul Malek Azad, Professor, Department of Electrical and Electronic Engineering (EEE), BRAC University, for his guidance for completion of the thesis. Our regards to Project Engineer, Ataur Rahman for his support throughout the whole thesis time span. We are thankful to EEE department, BRAC University for providing us the necessary equipment for the completion of this project and funding this project undertaken by Control and Application Research Centre (CARC).
ABSTRACT

The proposed fire suppression system is a real time monitoring system that detects the presence of flame of certain wavelength in the air due to fire and suppresses the fire via continuous water running through multiple sprinklers. The controlling unit used to develop the fire alarm system is an Arduino Uno. The key feature of this system is the ability to remotely send an alert when a fire is detected. Fire is an undesirable event that could cause a great loss of social wealth, human life and confidential amenities. To prevent these losses, various alarm systems have already been developed such as smoke detectors, temperature sensor based systems etc. These systems comes with a high chance of human error that can mislead the sensing and hence the whole system in general. Our fire suppression system on the other hand is extremely well thought out; it has no chance of human error as it is fully self-sufficient. The room or the area will have dedicated overhead sprinklers connected to the water supply through its dedicated solenoid valve which controls the flow of water in the pipe lines. The triggering devices of this system are the flame sensors, when they detect fire an alarm will go off. The coding set in the Arduino will come into work and provide the optimal voltage to the solenoid valve through a relay and hence water will rush out the sprinklers extinguishing fire in no time. The sensors we will be using are highly sensitive, thus it will detect fire even before it reaches to a point of concern. Water will be evenly spread out through the overhead sprinklers. Unlike other alarm systems, the components we used are readily available and reasonable in price. Our automated fire suppression system can be used both for commercial and residential purposes.
# TABLE OF CONTENTS

Acknowledgement.......................................................................................................................... ii
Abstract ........................................................................................................................................ iii
List of figures.................................................................................................................................. vii
List of tables............................................................................................................................... x
List of abbreviation................................................................................................................... xi

**CHAPTER 1: Introduction.** ........................................................................................................ 1
1.1 Motivation of the work and background.............................................................................. 1
1.2 Literature review and problem formulation....................................................................... 4
1.3 Project objective.................................................................................................................. 6
1.4 Contribution of this project.............................................................................................. 8
1.5 Thesis organization........................................................................................................... 8
1.6 Conclusion.......................................................................................................................... 9

**CHAPTER 2: Fire suppression mechanism.** .......................................................................... 10
2.1 Basic concept of fire suppression mechanism................................................................... 10
2.2 Existing fire extinguishing mechanisms.......................................................................... 10
2.3 Advantages of our automated system over existing technologies.................................. 13
2.4 Conclusion........................................................................................................................... 14

**CHAPTER 3: Fundamental concept of an embedded system and components**

*used in our embedded system.* .............................................................................................. 15
3.1 Introduction to embedded system..................................................................................... 15
3.2 Benefits of an embedded system..................................................................................... 16
3.3 Arduino Uno....................................................................................................................... 18
3.4 Four input NAND gate....................................................................................................... 22
CHAPTER 4: An automated fire suppression mechanism controlled using an Arduino

4.1 Overview of our whole experiment ..............................................37
  4.1.1 Introduction ........................................................................37
  4.1.2 Setup of our experiment .........................................................38
4.2 Electrical connections ..................................................................39
  4.2.1 Circuit diagram of the automated electronic system .................39
  4.2.2 Step by step description of the electrical connection ...............39
  4.2.3 Protection circuit against back electro motive force ..................42
    4.2.3.1 Introduction to back electro motive force .........................42
    4.2.3.2 Our protection circuit .......................................................44
4.3 Power management .....................................................................45
4.4 Design and calculation of the demonstration room .......................46
4.5 Sprinkler System and calculations regarding the location of the flame
  sensors .......................................................................................47
  4.5.1 Existing works of sprinkler heads .........................................47
  4.5.2 Step by step description of our sprinkler system and its calculations..49
  4.5.3 Placement and calculations regarding the location of the flame
  sensors .......................................................................................51
4.6 Control Algorithm .......................................................................54
  4.6.1 Arduino Code ......................................................................54
  4.6.2 Explanation of the code ..........................................................56
4.7 Conclusion ..................................................................................57

CHAPTER 5: Field tests and analysis ...................................................58
5.1 Wavelength of fire .......................................................................58
  5.1.1 Effect of sensor triggering wavelengths ....................................58
  5.1.2 Effect of non-sensor triggering wavelengths ..............................59
5.2 Distance of triggering the sensor……………………………………………………………60
5.3 Electrical measures of flame sensor…………………………………………………63
5.4 Conclusion…………………………………………………………………………68

CHAPTER 6: Conclusion and future work…………………………………………………69
6.1 Summary…………………………………………………………………………………69
6.2 Conclusion and Future work………………………………………………………….70

References………………………………………………………………………………..72

Appendix…………………………………………………………………………………..76
Appendix A: Electrical connection with all components running for a single sprinkler…………………………………………………………76
Appendix B Demonstration room construction………………………………………77
Appendix C Sprinkler system for single room………………………………………78
Appendix D Comparator circuit…………………………………………………………79
Appendix E Single sprinkler and flame sensor front view…………………………80
Appendix D Comparator circuit…………………………………………………………79
Appendix E Single sprinkler and flame sensor front view…………………………80
Appendix D Comparator circuit…………………………………………………………79
Appendix E Single sprinkler and flame sensor front view…………………………80
LIST OF FIGURES

Figure 3.1: The flow diagram of an embedded system.

Figure 3.2b: An example of an embedded system - Intel 4004.

Figure 3.3a: A typical example of a code in the Arduino software.

Figure 3.3b: A typical Arduino Uno R3.

Figure 3.4a: The logic diagram along with an expanded version of a typical 4 input NAND gate.

Figure 3.4b: The PIN assignment of a HD74HC20 4 input NAND gate.

Figure 3.5a: The flame sensor module used in our project.

Figure 3.6a: Output terminals of a relay.

Figure 3.6b: The SONGLE relay we used in the project.

Figure 3.7a: The parts of a typical solenoid valve.

Figure 3.7b: The solenoid valve we used for our project.

Figure 3.8a: Sprinkler head with glass bulb containing heat responsive liquid.

Figure 3.8b: Sprinkler head after removing glass bulb.

Figure 4.1.2: Setup of our experiment.

Figure 4.2.1: Circuit diagram of the automated system.

Figure 4.2.2a: Function of 4 input NAND Gate.

Figure 4.2.2b: Functionality of a relay in our project.

Figure 4.2.3.1a: When electromagnet is connected to power supply.
Figure 4.2.3.1b: When power supply is disconnected from electromagnet.

Figure 4.2.3.2a: Design of our protection circuit.

Figure 4.2.3.2b: Picture of our protection circuit.

Figure 4.3a: Two 12V, 20AH batteries.

Figure 4.3b: 24V to 5V DC step down voltage converter.

Figure 4.4: Demonstration room.

Figure 4.5.1: Radius of distribution of water (centimeter) with respect to pressure. (Pound per square inch) and mounting height (meter).

Figure 4.5.2a: Top view of sprinkler system with dimensions.

Figure 4.5.2b: Placement of a single sprinkler.

Figure 4.5.2c: Top view of sprinkler system.

Figure 4.5.3a: Radius of detection.

Figure 4.5.3b: Fire detection area of four flame sensors.

Figure 4.5.3c: Fire detected in the center of the demonstration room. Water is flowing through the solenoid and out the sprinkler heads.

Figure 4.5.3d: Fire suppressed successfully.

Figure 4.6.1a: Arduino code of an automated fire suppression mechanism controlled using an Arduino.

Figure 4.6.1b: Flowchart of the Arduino code of an automated fire suppression mechanism controlled using an Arduino.

Figure 5.1.1a: The spectral range of an electromagnetic spectrum.

Figure 5.1.1b: The spectral range in the infrared radiation at which the flame sensor triggers.
Figure 5.1.2: The ranges in which the flame sensor does not trigger or cannot detect any infrared.

Figure 5.2a: The flame sensor is ON; the switched OFF LED shows that fire is yet to be detected.

Figure 5.2b: The switched ON LED shows that the flame sensor detects fire at 20cm.

Figure 5.2c: The switched ON LED shows that the flame sensor detects fire at 40cm.

Figure 5.2d: The switched ON LED shows that the flame sensor detects fire at its maximum range of 100cm.

Figure 5.3a: The typical electro optical curves for YG1006 sensor.

Figure 5.3b: The schematic diagram of 1 LM 393 comparator chip.

Figure 6.2: System replaced with Thermo-graphic cameras and Raspberry Pi 3.

Figure Appendix A: Electrical connection with all components running for a single sprinkler.

Figure Appendix B: Diagram of an automated fire suppression mechanism controlled using an Arduino for a single room.

Figure Appendix C: Sprinkler system for single room.

Figure Appendix D: Comparator circuit with buzzer and LED.

Figure Appendix E: Single sprinkler and flame sensor front view.

Figure Appendix F: Controlling circuit along with the Flame sensors.

Figure Appendix G: Flowchart of the fire extinguishing process.
LIST OF TABLES

Table-1: Financial losses and casualties of countrywide fire incidents.

Table-2: Some related factory and slum incidents in and around Dhaka are mentioned

Table-3: The Truth table of a 4 input NAND gate.

Table 4: The maximum rating of a HD74HC20 4 input NAND gate.

Table 5: The recommended operating conditions for a HD74HC20 4 input NAND gate.

Table 6: The coil data chart of SONGLE relay at 20 degree Celsius.

Table 7: Truth table of 4 input NAND Gate (HD74HC20)

Table 8: Ratings of sprinkler head

Table 9: The electro-optical characteristics of YG 1006 sensor (25°C).

Table 10: The electro-optical characteristics include voltage, time, current and wavelength.

Table 11: shows the maximum ratings of a LM 393 comparator chip.
LIST OF ABBREVIATIONS

DSP- Digital signal processing.

AVR- Alf and Vegard's RISC processor.

IDE- Integrated development environment.

NFPA- National fire protection association.

NAND- means not AND.

VCC- refers to the voltage from the positive terminal of a power supply.

GND- indicates the ground terminal or the negative terminal of a power supply.

IC- Integrated circuit.

NO- Normally open terminal.

COM- is the common or return connection of the relay essentially when the relay is activated.

NO- Normally connected terminal.

CEMF- Counter electro motive force.

EMF- Electro motive force.

DC- Direct current.

PSI- pounds per square inch.
Chapter 1
Introduction

In this chapter we first introduce the topic of Fire Suppression System using Arduino Uno and describe the problem statement related to the simulation environment. Then the main research objectives are outlined followed by the research contributions. Finally, the main thesis overview is presented.

1.1 Motivation of the work and background

Fire has been a danger to assets and population through ages. In very recent times there have been multiple cases of fire throughout Dhaka city damaging properties worth millions and moreover taking away lives of innocent people. Locations like factories, slums and other open areas are most vulnerable to such disastrous events. Offices and other workplaces are also exposed to such events which may cause loss of important confidential documents and treasuries.

From January to February, incidents of 1,845 fires were recorded by Fire Service and Civil Defense all over Bangladesh.

Table-1[2]:

<table>
<thead>
<tr>
<th>Year</th>
<th>Countrywide fire</th>
<th>Financial loss</th>
<th>Casualty/Injury</th>
</tr>
</thead>
<tbody>
<tr>
<td>2016</td>
<td>16,858</td>
<td>240 crore 43 lakh</td>
<td>152/247</td>
</tr>
<tr>
<td>2017</td>
<td>1,845</td>
<td>35 crore 81 lakh</td>
<td>14/41</td>
</tr>
<tr>
<td>Year</td>
<td>Dhaka city fire</td>
<td>Financial loss</td>
<td>Casualty/Injury</td>
</tr>
<tr>
<td>2016</td>
<td>3,020</td>
<td>100 crore 63 lakh</td>
<td>15/81</td>
</tr>
</tbody>
</table>

The estimated financial loss from fires last year was Tk240.43 crores according to Fire Service and Civil Defense.
Some related factory and slum incidents in and around Dhaka are mentioned below in Table 2[1]:

<table>
<thead>
<tr>
<th>Factory name</th>
<th>Location</th>
<th>Incident date</th>
<th>Cause and outcome</th>
<th>Injuries</th>
<th>Deaths</th>
</tr>
</thead>
<tbody>
<tr>
<td>Next Collection</td>
<td>AshuliaSavar</td>
<td>3/1/2015</td>
<td>Thirty people have been injured in a stampede when the garment workers were stepping down from the factory following fire panic. The incident took place at Next Collection in Narshinghapur area of Ashulia around 11a.m. on Sunday. An electrical short circuit caused a &quot;sparkle of fire.&quot;</td>
<td>30</td>
<td>0</td>
</tr>
<tr>
<td>Unnamed shoe factory</td>
<td>Bangshal Dhaka</td>
<td>2/21/2014</td>
<td>Four workers have sustained burn injuries after a shoe factory in the capital's busy Bangshal area was burned down in a fire on Friday night. The fire broke out at a small shoe factory at the Kayettuli area around 10p.m. Two firefighting units doused the flames after 30 minutes of effort. Another victim of the shoe factory fire succumbed to his injuries last night at the Burn and Plastic Surgery Unit of Dhaka Medical College Hospital.</td>
<td>3</td>
<td>1</td>
</tr>
<tr>
<td>Amin dyeing</td>
<td>Fatullah, Narayanganj</td>
<td>1/5/2016</td>
<td>Some workers were smoking in the chemical go down of Amin Dyeing Factory around 1am. The explosion took place due to the leakage in the gas pipeline</td>
<td>5</td>
<td>0</td>
</tr>
<tr>
<td>Matrix Sweaters Ltd.</td>
<td>Choydana, National University</td>
<td>2/2/2016</td>
<td>Fire broke out at Matrix Sweater Factory at 7:30 am for the second time in last four days in the district. Joydebpur Fire Service station officer said fire broke out at the eighth floor of the factory. On information, eight unit of Tongi and Joydevpur fire service rushed to the spot and was trying to douse fire. Causes of fire and damages caused could not be known yet. On January 29 fire broke out on the same factory.</td>
<td>5</td>
<td>0</td>
</tr>
</tbody>
</table>
### Multifabs co.

**Location**: Gazipur, Dhaka  
**Date**: 7/4/2017 
**Event Description**: A boiler explosion at a Bangladeshi garment plant near the city of Dhaka killed 10 people and injured dozens. The blast occurred at a plant operated by local Bangladeshi firm Multifabs while maintenance work was going on, company and fire brigade officials said. The explosion at the boiler, located in a tin-roofed shed, partially damaged a nearby three-storey factory building. This was one of the latest industrial tragedy to hit one of the world’s biggest garment producers.

### Korail Slum

**Location**: Mohakhali, Dhaka  
**Date**: 3/16/2017 
**Event Description**: Many houses at one of the largest slums in Dhaka have been razed to the ground by a fire that broke out at the early hours of 16th March. Authorities are yet to report any casualties over the fire, the third in a year, at the Korail slum in the capital’s Mohakhali area. Fire service said that the incident was reported around 3am on Wednesday. It took the firefighters a little over five hours to put out the flames. Korail slum is one of the largest slums in the capital with thousands of inhabitants, hundreds of tin-shed houses at the slum were gutted. The slum, on a 150-acre land of the state run Bangladesh Telecommunication Company Ltd (BTCL), is home to several hundreds of thousands of people of the low-income group. Hundreds of houses of the slum were burnt in two incidents of fire on Dec 4 and Mar 14 last year.

### Banshbarhi Slum

**Location**: Mohammadpur, Dhaka  
**Date**: 2/16/2017 
**Event Description**: A fire has raged through a slum in Dhaka, leaving more than 100 shanties completely burnt. The fire service said no casualties were reported from the incident which took place early on Thursday morning at the ‘Banshbarhi Slum’ in the capital’s Mohammadpur area. Flames erupted at the slum around 3:30am, said Fire Service Control Room official Palash Modak. It took 13 fire-fighting units almost two and a half hours to put out the fire. The cause of the fire is yet to be known. Locals said more than 100 shanties were completely gutted in the fire.

These are just some of the reported events, other than these there have been news of multiple fire events in the last 3 or 4 years which were not reported. The very recent fire ravaged around 500 homes in Korail slum in Mohakhali. Fire disasters are increasing day by day and should not be overlooked time after time. These
unfortunate events gave rise to our project which suppresses fire in a much more
efficient way than the existing methods. Slums have narrow lanes and many
flammable materials in its premise which causes fire to spread out quickly and
makes work harder for Fire brigade to operate on the inside. Factories have many
electrical appliances which may be the cause for fire. We motivated ourselves to
build a technology that could reduce the fire hazard to a minimum at a minimum
cost.

1.2 Literature review and problem formulation.

Fire causes huge loss of lives and properties every year in Bangladesh. Analyzing
past fire incidents, facts are revealed. Some of the main causes are insufficient fire
defense materials, electric short circuit from faulty electrical wiring, presence of
inflammable materials, violation of fire safety and lack of adequate awareness etc.
Some factories and recent buildings have proper installation and fire safety
arrangements such as fire alarm, fire extinguishers, water supply system etc. But
the argument is these conventional fire extinguishing systems are not enough to
take prompt action during fire and save life. Traditional manual system does not
ensure 24/7 monitoring from fire protection. Moreover, existing fire protection
system could spread panic inside the whole building since it does not announce the
location of fire or intensity. It only raises alarm whenever fire is detected at any
place. Frightened people could start to run away haphazardly. As a result buildings
full of workers in the factories women, children could be smashed by the outgoing
pressure of the frightened crowd and injured severely. On the contrary, sometimes
people do not realize the intensity of the fire and not willing to evacuate fire
affected building quickly. It could lead a devastating result.

In this paper a Fire suppression system is presented that can reduce these hazards
to a great extent. The proposed system is fully automated and does not effectively
require any human effort to extinguish the unfortunate event. It is self-sufficient as
well. The fire alarm will set off once the fire breaks down which will open the
solenoid valves connected to water pipes by the help of a controlling unit (Arduino
Uno). Relays are connected in between the Arduino and the solenoid valve in order
to provide the required voltage to the valve. The sprinklers used in this system can cover a relatively large radius and can effectively suppress the fire within seconds.

Fire suppression systems are used to extinguish or prevent the spread of fire in a building or in an open space. An automatic fire suppression system control and extinguish fires without human intervention. When fires are extinguished in the early stages loss of life and property is minimal since 93% of all fire related deaths occur once the fire has progressed beyond the early stages.

Today there are numerous types of Automatic Fire Suppression Systems and standards for each one. Systems are as diverse as the many applications.

The existing fire alarm system in market nowadays, is too complex in term of its design and structure. Since the system is too complex, it needs regular preventive maintenance to be carried out to make sure the system operates well. Meanwhile, when the maintenance is been done to the existing system, it could raise the cost of using the system. Therefore, the proposed project is designed with a low cost and all level users can have one for a safety purpose.
1.3 Project objectives

Our projects objective is to design a Fire Alarm & Fire suppression System that would fulfill the following objectives:

i) Indicate the room in which fire erupted.

ii) Sound the alarm if fire occurs.

iii) Run the controlling unit (Arduino Uno) and control the fire by supplying water to the remote area by motor pump.

iv) False Alarm occurrence should be kept to minimum.

v) The system should also provide the flexibility to suppress the fire in multiple rooms.

vi) The sprinklers used in the system should suppress the fire over the correct radius and not waste any water unnecessarily.

vii) The system should never be in any ambiguous state. Under normal conditions the system should indicate the state of the room as safe to the normal people.

Furthermore there are other objectives to our research as well which includes:

i) To have a clear concept of an automated fire suppression system.

ii) To have a clear perception of the Arduino and its different characteristics involved in the project.

iii) To have a clear idea of the flame sensors those are used in the project by proper identification of their detection mechanism.
iv) Design and implement our proposed system at a proper place and safe environment.

v) Determine the faults in our system and finding the proper solution.

vi) Monitor the overall system over certain period of time.

In a way to achieved above objectives, this project needs to be implemented as below:

i) This fire alarm system can incorporate the solenoid valve and flame sensors.

ii) The Arduino is used as the heart of this fire alarm system that controls the entire operations involved.

iii) The fire alarm system is capable to locate and identify the place that is in fire while it is controlled by the Arduino.

iv) Capable to display the output in terms of extinguishing the fire completely in a short period of time.
1.4 Contributions of the research

i) Once properly executed, this project will save human lives and amenities from disaster.

ii) A self-sufficient fire suppression system is always required in both residential and industrial environment.

iii) An advanced controlling unit (Arduino Uno) is coded and designed in such a manner that can effectively and efficiently control this system under all circumstances.

iv) An ideal power back up has been proposed in case of emergency power supply and its performance has been justified.

v) An automated fire suppression system is created at such a low cost that can prove to be very beneficial for government purposes in the near future.

1.5 Thesis organization

The rest of the research is divided into the following chapters:

Chapter 2: Fire suppression mechanism

Chapter 3: Fundamental concept of an embedded system and components used in our embedded system.

Chapter 4: An automated fire suppression mechanism controlled using an Arduino.

Chapter 5: Field tests and analysis.

Chapter 6: Conclusion and future work.
1.6 Conclusion

In a nutshell Chapter 1 gives us an idea of the overall project in terms of its objectives, contributions and literature review. This chapter also talks about the tragedies caused by fire in Bangladesh which motivated us to come up with this self-sufficient system. It also states the problems we faced during the project and ways to formulate those problems. The thesis organization gives us hints about how the rest of the paper will be followed through.
Chapter 2

Fire Suppression Mechanism

2.1 Basic concept of Fire suppression mechanism.

Suppressing fire by throwing water onto it has been used since ancient times. To provide an automatic spray of water to control a fire, sprinkler systems were developed in the late 19th century. Since then, automatic sprinklers have become the most common fixed fire suppression system for providing fire safety in buildings. Sprinklers control fire development by wetting and cooling the fuel surface.

They are effective for fires involving solid materials (referred to as solid fuels) but are not effective for flammable liquids (called liquid fuels) such as gasoline, diesel and jet fuels. The old adage “oil and water do not mix” must be kept in mind. Fire suppression systems for liquid fuels typically use foam or dry chemicals, which cover the fuel surface, hence limiting thermal feedback to the liquid fuel surface and fuel vaporization.

2.2 Existing fire extinguishing mechanisms

The new fire suppression systems use technologies such as: inert and halocarbon gaseous agents, water mist systems and solid gas generator.

Gaseous Systems:
Two types of gaseous agents are available for use in total flooding systems-halocarbon agents and inert gases. A general requirement for such systems is that the enclosure must be capable of holding the gas and be able to withstand the high pressures produced during discharge.
Halocarbon Agents-
Halocarbon agents are chemicals similar to Halon except that their molecular structure has been modified to reduce the number or to eliminate completely, the chlorine and bromine atoms, which are responsible for ozone depletion. These agents extinguish fires primarily by cooling. Acceptance of a halocarbon agent by regulatory authorities hinges on the agent’s toxicity. Two toxicological aspects must be considered. One is the toxicity of the agent itself, and the other is the toxicity of combustion by-products of the agent produced under fire conditions. Results from both small-scale and full scale tests have shown that the halocarbon replacement systems extinguish fires well, though not as effectively as Halons. To provide the same level of fire protection as Halons, larger amounts of halocarbon agents are needed. This means larger and heavier cylinders are required, which may create weight and space problems. The test results also show that halocarbon agents produce five to ten times more toxic gases than Halon 1301 during fire suppression. These gases include hydrogen fluoride (HF) and carbonyl difluoride (COF2), with levels produced in test fires significantly exceeding all human exposure limits. The levels of HF and COF2 likely to be produced in actual applications will depend on many factors such as agent type and concentration, fire type and size, and discharge and extinguishment times.

Inert Gas Agents-
Inert gas agents are applied as total flooding agents. They extinguish fire by displacing the oxygen in the enclosed space and eventually reducing its concentration below the level required for combustion. Inert gases, such as nitrogen, argon and helium, are clean and naturally occurring, have zero ozone depletion potential and no global warming potential. They are not subject to thermal decomposition when used in extinguishing fires, and hence form no combustion by-products. One of the disadvantages of using inert gas systems is that a large volume of agent is required to extinguish a fire. As well, inert gases cannot be liquefied and must be stored in cylinders as high pressure gases, which has implications for space and weight. Inert gases also require a discharge system sufficiently robust to withstand the high pressures involved. The rapid displacement of oxygen, high noise levels and rapid cooling are also a concern if the agent is to be discharged into an occupied space. [4]
Water Mist:
The term "water mist" refers to fine water sprays in which 99% of the volume of the spray is in droplets smaller than 1000 microns in diameter. Fire suppression by water mist is mainly by physical mechanisms. No significant chemical effects are involved. Water mist does not behave like a “true” gaseous agent. The compartment tests showed that its effectiveness in fire suppression is substantially affected by the fire size, the degree of obstruction, ceiling height, and the ventilation conditions. Water mist characteristics, such as variety of drop sizes and spray momentum, have a direct influence on effectiveness. To effectively suppress a fire, a water mist system must generate and deliver optimum-sized droplets with an adequate concentration. The selection of the optimum size of droplets for the design of the system is dependent on the potential size of the fire, properties of the combustibles, and the degree of obstruction and ventilation in the compartment. There is no one drop size distribution to fit all fire scenarios. There are several water mist systems available commercially. Some employ high or intermediate pressures of water through small orifices in a nozzle to produce the mist, while others use twin fluid nozzles (water and air). Water mist systems have demonstrated a number of advantages, such as good fire suppression capability, no environmental impact and no toxicity. As a result, they have been considered for numerous applications. One potential application is shipboard machinery spaces. Water mist systems are able to extinguish a wide variety of fires when natural ventilation, such as open doors and hatches are allowed, whereas gaseous agents were not effective under such conditions. Water mist systems also rapidly reduced the compartment temperature and significantly improved visibility. These advantages allow accessibility to the compartment during fire suppression. [5]

Gas Generators:
Based on automotive airbag technology, gas generators have been developed for fire suppression applications. Gas generators can produce a large quantity of gases (mainly N2, CO2 and water vapor) by combustion of solid propellants. Solid propellants consist of oxidizers and fuel ingredients, and are able to burn without ambient air. Gas generators can be very compact and can provide very fast discharge (in a few milliseconds). Currently, there are two types of gas generators available: conventional and hybrid. Conventional gas generators contain a propellant and an electrical initiator. When a signal is received from a
detector/controller, the electrical initiator ignites a charge to start a combustion process in the propellant. Rapid combustion of the solid propellant generates large amounts of N2, CO2 and water vapor, which rapidly increases the internal pressure. A hermetic seal is ruptured and the gas products are discharged within milliseconds into the protected space. Suppression is by oxygen displacement and gas discharge dynamics (blowing effect).A hybrid gas generator consists of an electrical initiator, a solid propellant chamber and a suppression agent chamber. The heat and pressure generated by the combustion of the propellant are used to heat and expel the liquefied suppressant. Gas generators are limited for use in unoccupied spaces only, because of their high temperature and high momentum discharge.[6]

2.3 Advantages of our automated system over existing technologies.

The automated system that we designed has multiple advantages over other technologies:

i) We used solenoid valves which allow water to flow on one dedicated pipe; this enables us to localize one affected region at a time rather than spreading water over the entire area.

ii) The flame sensors we used are effective and responsive to fire over a relatively large distance.

iii) The time delay is negligible between the emergence of fire and detection procedure. As soon as the fire is detected, the buzzer turns ON and makes a loud noise alerting the people around the affected area to evacuate as the sprinklers project water.

iv) No external power supply is required since batteries are used to supply power to the solenoid valves. Two 12V batteries in series, which is 24V is enough to run a single system.

v) The Arduino Uno used in our project only requires a power supply of 5V which is readily accessible. The 24V power supplied from the battery can be converted to 5V through a step down voltage regulator.

vi) A protection circuit is designed using capacitor and diode in such a way that will reduce the flow of back EMF which can be an issue when the system is switched ON and OFF.
2.4 Conclusion

Chapter 2 portrays the basics of fire suppression mechanisms along with the existing systems of fire suppression mechanisms such as gaseous systems, water mist systems and compressed-air foam mechanisms.

Most importantly this chapter tells us the advantages of our systems over the existing mechanisms. These advantages include localizing the fire, highly responsive flame sensors, fast response between fire detection and extinguishing process, independence of the system from grid power, an intelligent and efficient controlling unit and the system’s own protection unit.
Chapter 3
Fundamental Concept of an Embedded System and Components Used In Our Embedded System

3.1 Introduction to an embedded system

The electronic system which integrates the hardware circuitry with the software programming techniques for providing project solutions is called as embedded systems. By using this embedded system technology the complexity of the circuits can be reduced to a great extent which further reduces the cost and size. Embedded system was primarily developed by Charles Stark for reducing the size and weight of the project circuitry.

An embedded system is basically an electronic system that can be programmed or non-programmed to operate, organize, and perform single or multiple tasks based on the application. In the real time embedded systems, all the assembled units work together based on the program or set of rules or code embedded into the microcontroller. But, by using this microcontroller programming techniques only a limited range of problems can be solved.

![Flow diagram of an embedded system](image)

**Figure 3.1:** Flow diagram of an embedded system.[8]
Modern embedded systems are often based on microcontrollers (i.e. CPU's with integrated memory or peripheral interfaces), but ordinary microprocessors (using external chips for memory and peripheral interface circuits) are also common, especially in more-complex systems. In either case, the processor(s) used may be types ranging from general purpose to those specialized in certain class of computations or even custom designed for the application at hand. A common standard class of dedicated processors is the digital signal processor (DSP).

One of the very first recognizably modern embedded systems was the Apollo Guidance Computer developed by Charles Stark Draper at the MIT Instrumentation Laboratory. [8]

### 3.2 Benefits of an embedded system

Many electrical and computer engineering projects involve some kind of embedded systems which makes it important to study these kind of systems. An embedded system is an electronic system that contains at least one controlling device, for instance, the brain. This brain is called the microcontroller that controls the peripherals as input and output devices. In fact, every embedded system consists of two parts. The first part is called the hardware and the other one is the software. The software part always needs a program that can be used to manipulate the hardware. There are several examples of devices containing embedded systems such as: automobiles, traffic lights, cameras, mp3 audio players, wireless network routers, and many other electronic devices.

Moreover, a lot of upgrades have been made on embedded systems in the last ten years that led to decrease the weight and cost of embedded systems. Embedded systems could benefit people in many fields such as, raises people’s standard of living; being used in different fields of aerospace engineering and industry; being widely used in military applications and mobile robots; however, Embedded Systems have some minor disadvantages. Embedded systems are used in a variety of applications. Here are some important applications: electronic products, household appliances, Medical equipment, automobiles, industrial applications, aerospace, and communications.
Embedded systems helped in developing automotive safety systems, for instance, electronic fuel injection systems, Global Positioning System (GPS), and anti-lock braking systems (ABS).

![Intel 4004](https://en.wikipedia.org/wiki/File:4004_Intel.jpg)

**Figure 3.2b**: An example of an embedded system—Intel 4004. [7]

Alternatively, embedded systems are being used in mobile robots and some military applications. In fact, there has been an enormous increase of interest in mobile robots as they are perfect tools for engineering education. In the past, mobile robots were controlled by large, expensive, and heavy computer systems that could not be carried and had to be linked via cables.

Furthermore, it is hard to maintain embedded systems because the software part of embedded systems is typically monolithic and these systems are hard to upgrade and customize. Also, it is hard to carry files from one machine to another because there are different programming languages that can used to manipulate the hardware of the embedded system, so carrying files from one machine to another is hard and may not work with the other hardware. In conclusion, it can be seen that there are many beneficial uses of embedded systems in our daily life which are raising people’s standard of living in many fields, being used effectively in different fields of industry and aerospace applications, being used in mobile robots and in some military applications. [7]
3.3 Arduino Uno

Arduino is an open source computer hardware and software company, project, and user community that designs and manufactures single-board microcontrollers and microcontroller kits for building digital devices and interactive objects that can sense and control objects in the physical world.

Arduino board designs use a variety of microprocessors and controllers. The boards are equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, which are also used for loading programs from personal computers. The microcontrollers are typically programmed using a dialect of features from the programming languages C and C++. A program written with the IDE for Arduino is called a sketch. Sketches are saved on the development computer as text files with the file extension .ino. Arduino Software (IDE) pre-1.0 saved sketches with the extension .pde.

Figure 3.3a: A typical example of a code in the Arduino software. [9]
The version that we will be using in our research is Arduino Uno R3. The **Arduino Uno R3** is a microcontroller board based on the ATmega328 (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller. We need to simply connect it to a computer with a USB cable or power it with an AC-to-DC adapter or battery to get started.

A summary of the configuration of the device:

<table>
<thead>
<tr>
<th>Microcontroller</th>
<th>ATmega328</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating Voltage</td>
<td>5V</td>
</tr>
<tr>
<td>Input Voltage (recommended)</td>
<td>7-12V</td>
</tr>
<tr>
<td>Input Voltage (limits)</td>
<td>6-20V</td>
</tr>
<tr>
<td>Digital I/O Pins</td>
<td>14 (of which 6 provide PWM output)</td>
</tr>
<tr>
<td>Analog Input Pins</td>
<td>6</td>
</tr>
<tr>
<td>DC Current per I/O Pin</td>
<td>40 mA</td>
</tr>
<tr>
<td>DC Current for 3.3V Pin</td>
<td>50 mA</td>
</tr>
<tr>
<td>Flash Memory</td>
<td>32 KB (ATmega328) of which 0.5 KB used by bootloader</td>
</tr>
<tr>
<td>SRAM</td>
<td>2 KB (ATmega328)</td>
</tr>
<tr>
<td>EEPROM</td>
<td>1 KB (ATmega328)</td>
</tr>
<tr>
<td>Clock Speed</td>
<td>16 MHz</td>
</tr>
</tbody>
</table>
The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

![Typical Arduino Uno R3](image)

**Figure 3.3b:** A typical Arduino Uno R3. [9]

The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts.

The power pins are as follows:

- **Vin:** The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source).
- **5V:** The regulated power supply used to power the microcontroller and other components on the board.
- **3.3V:** A 3.3 volt supply generated by the on-board regulator. Maximum current draw is 50 mA.
- **GND:** Ground pins.
Each of the 14 digital pins on the Uno 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 kiloohms. In addition, some pins have specialized functions:

- **Serial 0 (RX) and 1 (TX):** These are used to receive (RX) and transmit (TX) TTL serial data. These pins are connected to the corresponding pins of the ATmega8U2 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.
- **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 using the SPI library.
- **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 Uno has 6 analog inputs, labeled A0 through A5, each of which provide 10 bits of resolution (i.e. 1024 different values).

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.

The Arduino Uno can be programmed with the Arduino software. The maximum length and width of the Uno PCB are 2.7 and 2.1 inches respectively, with the USB connector and power jack extending beyond the former dimension. [9]
3.4 Four input NAND gate

The NAND gate has an output that is normally at logic level “1” and only goes “LOW” to logic level “0” when all of its inputs are at logic level “1”. The Logic NAND Gate is the reverse or complementary form of the AND gate. The truth table of a typical NAND gate is shown below:

<table>
<thead>
<tr>
<th>Input A</th>
<th>Input B</th>
<th>Input C</th>
<th>Input D</th>
<th>Output Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

**Table 3:** The truth table of a typical NAND gate. [12]

The 4 inputs of the NAND gate are connected to the flame sensors directly through prolonged wires. When either of the sensors detects a fire, it sends out a signal to the microcontroller which carries out the necessary functions in order to extinguish
the fire. As we can see from the truth table, if either one of the inputs connected to the sensors gives a logic “0” the NAND gate gives an output of logic “1”.

This means that when the fire ignites at any corner of the enclosed area, then either one of the sensors can detect the fire at once and start up the extinguishing procedure.

The 4 input NAND gate we used in our project has a model number of HD74HC20. The HD74HC20 is identical in pin-out to the LS20. The device inputs are compatible with standard CMOS outputs; with pull-up resistors, they are compatible with LSTTL outputs.

![NAND gate diagram](image1)

**Figure 3.4a:** The logic diagram along with an expanded version of a typical 4 input NAND gate. [12]

Like any other logic gates, a HD74HC20 4 input NAND gate also acquires certain features. The features of a 4 input NAND gate are as follows:

- **High Speed Operation:** $tpd = 9 \text{ ns typ (CL = 50 pF)}$
- **High Output Current:** Fanout of 10 LSTTL Loads
- **Wide Operating Voltage:** $VCC = 2$ to $6 \text{ V}$
- **Low Input Current:** 1 $\mu\text{A}$ max
- **Low Quiescent Supply Current:** $ICC (\text{static}) = 1 \mu\text{A max (Ta = 25°C)}$
Figure 3.4b The PIN assignment of a HD74HC20 4 input NAND gate. [12]

Table 4: The maximum rating of a HD74HC20 4 input NAND gate. [12]

<table>
<thead>
<tr>
<th>Item</th>
<th>Symbol</th>
<th>Ratings</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Supply voltage range</td>
<td>( V_{CC} )</td>
<td>-0.5 to 7.0</td>
<td>V</td>
</tr>
<tr>
<td>Input / Output voltage</td>
<td>( V_{IN}, V_{OUT} )</td>
<td>-0.5 to ( V_{CC} + 0.5 )</td>
<td>V</td>
</tr>
<tr>
<td>Input / Output diode current</td>
<td>( I_{OK}, I_{OK} )</td>
<td>±20</td>
<td>mA</td>
</tr>
<tr>
<td>Output current</td>
<td>( I_{O} )</td>
<td>±25</td>
<td>mA</td>
</tr>
<tr>
<td>( V_{CC}, GND ) current</td>
<td>( I_{CC} ) or ( I_{GND} )</td>
<td>±50</td>
<td>mA</td>
</tr>
<tr>
<td>Power dissipation</td>
<td>( P_{t} )</td>
<td>500</td>
<td>MW</td>
</tr>
<tr>
<td>Storage temperature</td>
<td>( T_{stg} )</td>
<td>-65 to +150</td>
<td>°C</td>
</tr>
</tbody>
</table>

HD74HC20 4 input NAND gate device contains a protection circuit that is used to guard against damage that can be caused due to high static voltages and electric fields. When using this device, precautions must be taken so that the voltage applied should not exceed the maximum rating. For proper operation, \( V_{IN} \) and \( V_{OUT} \) should always be maintained in such a way so that GND is less than or equal to \( V_{IN} \) or \( V_{OUT} \) which in turn should be less than or equal to \( V_{CC} \). Inputs that are unused should always be connected to the \( V_{CC} \) or GND logic voltage level. [12]
Table 5: The recommended operating conditions for a HD74HC20 4 input NAND gate. [12]

3.5 Flame sensor

The flame sensor module that we used is sensitive to the flame and radiation. It can also detect ordinary light source at the range of a wavelength from 760nm to 1100 nm. The detection distance is up to 100 cm (1 m). This Flame sensor has a digital signal as an output. The output is 0 (0V) when radiation in its range is detected, else the output is 1 (3.3 or 5V, depends on its Vcc). [10]

Figure 3.5a: The flame sensor module used in our project.
The descriptions of the pins are as follows:

- Vcc: (3.3-5V).
- GND - Ground.
- DO - board digital output interface (0 and 1).

Some of the common characteristics of this flame sensor module include [11]:

- It can detect a flame or a light source of a wavelength in the range of 760nm-1100 nm.
- It can detect flame at a distance ranging from 20cm (4.8V) ~ 100cm (1V).
- It can detect flame at an angle about 60 degrees and is sensitive to the flame spectrum.
- It is based on YG1006 sensor, a NPN phototransistor.
- Comparator chip LM393 allows the module readings to become more stable and easy to capture.
- The range of wavelength detection can be adjusted in accordance to our needs.
- The module operates at voltage ranging from 3.3V-5V.
- Digital Output: DO- digital switch outputs (0 and 1)
- The module contains a power indicator and digital switch output indicator.[10]
3.6 Relay

A relay is an electrically operated switch. Current flowing through the coil of the relay creates a magnetic field which attracts a lever and changes the switch contacts. The coil current can be on or off so relays have two switch positions and most have double throw (changeover) switch contacts as shown in the diagram:

![Circuit symbol for a relay](image)

**Figure 3.6a:** Output terminals of a relay. [13]

Relays allow one circuit to switch a second circuit which can be completely separate from the first. For example a low voltage battery circuit can use a relay to switch a 230V AC mains circuit. There is no electrical connection inside the relay between the two circuits; the link is magnetic and mechanical. The coil of a relay passes a relatively large current, typically 30mA for a 12V relay, but it can be as much as 100mA for relays designed to operate from lower voltages. Most ICs cannot provide this current and a transistor is usually used to amplify the small IC current to the larger value required for the relay coil. The maximum output current for the popular 555 timer IC is 200mA, enough to supply a relay coil directly. Most relays are designed for PCB mounting but you can solder wires directly to the pins providing you take care to avoid melting the plastic case of the relay. The relay's switch connections are usually labeled COM, NC and NO:

- **COM** = Common, always connect to this, it is the moving part of the switch.
- **NC** = Normally Closed, COM is connected to this when the relay coil is off.
- **NO** = Normally Open, COM is connected to this when the relay coil is on.
We need to connect to COM and NO if we want the switched circuit to be on when the relay coil is on. Also, we need to connect to COM and NC if we want the switched circuit to be on when the relay coil is off.

We need to consider several features when choosing a relay:

1. Physical size and pin arrangement: If we are choosing a relay for an existing PCB we will need to ensure that its dimensions and pin arrangement are suitable. We should find this information in the supplier's catalogue or on their website.

2. Coil voltage: The relay's coil voltage rating and resistance must suit the circuit powering the relay coil. Many relays have a coil rated for a 12V supply but 5V and 24V relays are also readily available. Some relays operate perfectly well with a supply voltage which is a little lower than their rated value.

3. Coil resistance: The circuit must be able to supply the current required by the relay coil.

4. Switch ratings (voltage and current): The relay's switch contacts must be suitable for the circuit they are to control. We will need to check the voltage and current ratings. Note that the voltage rating is usually higher for AC, for example: "5A at 24V DC or 125V AC".

5. Switch contact arrangement (SPDT, DPDT etc): Most relays are SPDT or DPDT which are often described as "single pole changeover" (SPCO) or "double pole changeover" (DPCO).

The relay we used was of SONGLE Company and acquires these special features:

- Has a switching capacity available by 10A in spite of small size design for high density P.C. board mounting technique.
- UL, CUL, TUV recognized.
- Has a selection of plastic material for high temperature and better chemical solution performance.
• Have several sealed types available.
• Contains a simple relay magnetic circuit to meet low cost of mass production.[13]

**Figure 3.6b:** The SONGLE relay we used in the project.

<table>
<thead>
<tr>
<th>Coil Sensitivity</th>
<th>Coil Voltage Code</th>
<th>Nominal Voltage (VDC)</th>
<th>Nominal Current (mA)</th>
<th>Coil Resistance (Ω) ±10%</th>
<th>Power Consumption (W)</th>
<th>Pull-In Voltage (VDC)</th>
<th>Drop-Out Voltage (VDC)</th>
<th>Max-Allowable Voltage (VDC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRD (High Sensitivity)</td>
<td>03</td>
<td>03</td>
<td>120</td>
<td>25</td>
<td>abt. 0.36W</td>
<td>75% Max.</td>
<td>10% Min.</td>
<td>120%</td>
</tr>
<tr>
<td></td>
<td>05</td>
<td>05</td>
<td>71.4</td>
<td>70</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>06</td>
<td>06</td>
<td>60</td>
<td>100</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>09</td>
<td>09</td>
<td>40</td>
<td>225</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>12</td>
<td>30</td>
<td>400</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>24</td>
<td>15</td>
<td>1600</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>48</td>
<td>48</td>
<td>7.5</td>
<td>6400</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRD (Standard)</td>
<td>03</td>
<td>03</td>
<td>150</td>
<td>20</td>
<td>abt. 0.45W</td>
<td>75% Max.</td>
<td>10% Min.</td>
<td>110%</td>
</tr>
<tr>
<td></td>
<td>05</td>
<td>05</td>
<td>89.3</td>
<td>55</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>06</td>
<td>06</td>
<td>75</td>
<td>80</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>09</td>
<td>09</td>
<td>50</td>
<td>180</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>12</td>
<td>12</td>
<td>37.5</td>
<td>320</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>24</td>
<td>18.7</td>
<td>1280</td>
<td>abt. 0.51W</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>48</td>
<td>48</td>
<td>10</td>
<td>4500</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Table 6:** The coil data chart of SONGLE relay at 20 degree Celsius. [13]
3.7 Solenoid Valve

A solenoid valve is an electromechanically operated valve. The valve is controlled by an electric current through a solenoid: in the case of a two-port valve the flow is switched on or off; in the case of a three-port valve, the outflow is switched between the two outlet ports. Its main function is to allow the fluid to pass through only when certain threshold voltage is applied to it.

Figure 3.7a: The parts of a typical solenoid valve. [12]

Here are the various parts of the solenoid valve and their working according to the figure above:

1) Valve body: This is the body of the valve to which the solenoid valve is connected. The valve is usually connected in the process flow pipeline to control the flow of certain fluid like liquid or air. Ordinarily the flow from the valve is controlled by the handle, but in case of the automatic valve the solenoid valve is connected to the valve.
2) Inlet port of the valve: This is the port through which the fluid enters inside the automatic valve and from here it can enter into the final process.

3) Outlet port: The fluid that is allowed to pass through the automatic valve leaves the valve through the outlet port. The solenoid valve controls the flow of the fluid from inlet port to the outlet port. The outlet port is eventually connected to the process where the fluid is required.

4) Coil/ Solenoid: This is body of the solenoid coil. The body of the solenoid coil is cylindrical in shape, and it is hollow from inside. The body is covered with steel covering and it has metallic finish. Inside the solenoid valve there is solenoid coil.

5) Coil windings: The solenoid consists of several turns of the enameled wire wound around the ferromagnetic material like steel or iron. The coil forms the shape of the hollow cylinder. Externally this coil is covered with the steel covering and inside the hollow part there is a plunger or the piston, whose motion inside the hollow space is controlled by the spring.

6) Lead wires: These are external connections of the solenoid valve that are connected to the electrical supply. The current is supplied to the solenoid valve from these wires. When the solenoid valve is energized, the current flows through these wires to the solenoid valve and when the solenoid valve is de-energized the flow of current stops.

7) Plunger or piston: This is the solid round metallic part cylindrical in shape and placed in the hollow portion the solenoid valve. When the electrical current is passed through the solenoid valve, the magnetic field is generated inside the hollow space. Due to this the plunger tends to move vertically in the hollow space. When the electrical current is stopped to the solenoid valve, the magnetic field is stopped and the plunger is remains the existing place due to the force of the spring.

8) Spring: The plunger moves inside the hollow space due to the action of the magnetic field against the action of the spring. The magnetic field generated inside the solenoid valve tends to move the plunger, but the spring tends to stop the motion of the plunger in which ever the position it is. This action of the spring against the magnetic field helps keeping the plunger in the position where the flow
of current to the solenoid valve is stopped. The spring performs very crucial action inside the hollow space. For one, the plunger is in the vertical position, so the spring helps keeping it at the desired position instead of allowing the plunger to fall to the bottom due to gravity when the current to the solenoid valve is stopped. Secondly, the spring also prevents the movement of the plunger due to force of the fluid flowing through the valve body. If the spring was not there the plunger would have moved up when the fluid is present and moved down when the fluid is not there. Thus the spring actually forces the plunger to carry out the control of the fluid. It allows the movement of the plunger only to the extent when the electric current is flowing through the solenoid valve.

9) Orifice: The orifice is an important part of the valve though which the fluid is flowing. It is the connection between the inlet and the outlet port. The flow of fluid from the inlet port to the outlet port takes place from this port. In the ordinary valves, this port is covered with the valve disc at the bottom of the stem of the valve to which the handle is connected. Thus in ordinary valves, the opening of the orifice are controlled by the handle, but in case of the solenoid valves, the opening of the orifice is controlled by the plunger. The movement of the plunger is in turn controlled by the spring and the current flowing through the solenoid valve. If the current passing through the solenoid valve is constant, the position of the plunger and hence opening of the orifice remains constant. If the sensor senses that more flow of the fluid is required, it allows the increase in current passing through the solenoid valve, which creates more magnetic field and more upwards motion of the plunger. This leads to further opening of the orifice and more flow of the fluid from the inlet port to the outlet. If the required flow of fluid is less, the sensor allows passage of the lesser current to the solenoid valve. When the sensor senses that the fluid is no more required in the process, it stops the flow of the current to the solenoid valve completely. Due to this the solenoid valve gets de-energized and the plunger reaches the bottom most position and closes the orifice completely thus stopping the flow of fluid from the inlet port to the outlet port.
Figure 3.7b: The solenoid valve we used for our project.

A brief working procedure of a solenoid valve is given below:

Initially the sensor senses the process towards the outlet side of the solenoid valve. When it senses that certain quantity of the flow of the fluid is required, it allows the current to pass through the solenoid valve. Due to this the valve gets energized and the magnetic field is generated which triggers the movement of the plunger against the action of the spring. Due to this the plunger moves in upwards direction, which allows the opening of the orifice. At this instant the flow of the fluid is allowed from the inlet port to the outlet port.

If the current passing through the solenoid valve is constant, the position of the plunger and hence opening of the orifice remains constant. If the sensor senses that more flow of the fluid is required, it allows the increase in current passing through the solenoid valve, which creates more magnetic field and more upwards motion of the plunger. This leads to further opening of the orifice and more flow of the fluid from the inlet port to the outlet. If the required flow of fluid is less, the sensor allows passage of the lesser current to the solenoid valve.

When the sensor senses that the fluid is no more required in the process, it stops the flow of the current to the solenoid valve completely. Due to this the solenoid valve gets de-energized and the plunger reaches the bottom most position and
closes the orifice completely thus stopping the flow of fluid from the inlet port to the outlet port.

In this way the solenoid coil operates the valve as if it is being operated by the human being. When the flow of certain quantity of fluid is required it opens the valve to required extent and when the flow is not required it shuts the valve entirely. [12]

3.8 Miscellaneous devices

Apart from the main components that are used in our project, we also used some sub components which is described in this chapter. Here is a list of the miscellaneous devices we used in our project:

- A capacitor and a diode are used for our protection circuit against back EMF.
- A buzzer is used in order to produce the alerting noise which will set off when fire ignites in the enclosed area.
- A LED light is used along with a resistor which only lights up when fire sets out in the enclosed area.
- A pair of 12V batteries is used to give the power supply to both the Solenoid valve and the Arduino.
- A voltage regulator is used to convert the 24V supply from the battery to 5V which is necessary for the Arduino.
- Multiple wires are prolonged and used for various purposes in our project such as connecting the flame sensors to the Arduino, connecting the Arduino to the relay and also connecting the relay to the protection circuit.
- Water pipes are used to supply water to the solenoid valves from the main water supply.
- Sprinkler heads were used to spread water over the affected area. The glass bulb containing the heat responsive liquid was removed from the pendent so that water can flow any time fire is detected.
Figure 3.8a: Sprinkler head with glass bulb containing heat responsive liquid.

Figure 3.8b: Sprinkler head after removing glass bulb.
3.9 Conclusion

Basic concept of an embedded system, its benefits and its importance in our project has been justified at the start of this chapter. The controlling unit used in our system, an Arduino, has been described part by part along with its control algorithm (code). Supporting devices such as the flame sensors, relay, solenoid valve and HD74HC20 IC’s basic functionality and their contribution in our project have been explained. Miscellaneous devices such as the buzzer and LED which make up the alarm system, the capacitor and the diode which make up the protection circuit and the batteries for our power management has been talked about. In the end it is shown with the aid of before and after pictures how the glass bulb containing the heat responsive liquid has been removed from the sprinkler heads before installing in the pipe lines.
Chapter 4

An Automated Fire Suppression Mechanism Controlled Using an Arduino

4.1 Overview of the whole system

4.1.1 Introduction

This chapter shall concentrate on our project; an automated fire suppression mechanism controlled using an Arduino. After understanding the basic concepts of fire suppression system from chapter 2 and embedded system from chapter 3 we are going to explain how we used all the components to implement our system.

Our design has six major parts, the electrical connection, sprinkler system, design of the demonstration room, calculations regarding location of flame sensors, the control algorithm and finally the power management. In electrical connections we will explain the circuitry of our automated system along with the protection circuit that protects it. We will show how we used our sprinkler system as effectively and efficiently as possible with the help of using data from existing sprinkler systems. Sensing the fire is the first and most important task, in order to do that we had to carefully place the flame sensors in correct location. The calculations regarding placement of sensors, design of demonstration room and explanations will be provided for better understanding. The automated system’s decision making process that is the control algorithm will be explained step by step.
4.1.2 Setup of our experiment

Figure 4.1.2: Setup of our experiment.

Here we can see the demonstration room fitted with four over head sprinklers that comes through one solenoid valve. The water supply pipe line that comes from the reservoir is controlled by this solenoid valve. Flame sensors are placed just above the sprinklers. There is no chance of electrocution as the flame sensors are heavily insulated with silicone. The wire from the flame sensors run through the insulated tubing place on the back of the wall to the comparator circuit. The Arduino, comparator circuit, alarm, relay, protection circuit and the batteries are place at a distance from the demonstration room as shown in Figure 4.1.2.
4.2 Electrical connections

4.2.1 Circuit diagram of the automated electronic system

![Diagram of the automated electronic system](image)

Figure 4.2.1: Circuit diagram of the automated system.

4.2.2 Step by step description of the electrical connection

The controlling unit of the system is the Arduino. Arduino has Digital and Analog pins, we decided to use the digital pins since both the flame sensors and the relay operates using digital signal. Pin 2 to pin 7 of Arduino are digital input pins and pin 8 to pin 13 are digital output pins. For receiving signals from flame sensors we used pin 7 and for output we used pin 13 to turn relay ON or OFF.

As shown in figure 4.2.1, the four flame sensors will be constantly checking for traces of flame. If it detects no flame its output signal is 1 and if it detects flame its output signal is 0. The output signals of the sensors are feed to the 4 input NAND
gate (HD74HC20). We selected input pin 1, 2, 4 and 5 of the NAND gate. One may question why a NAND gate was used. To answer that question we need to look at the function and truth table of a 4 input NAND gate. Let us assume that the inputs are A, B, C, D and the output is Y.

Therefore the function will be:

\[ Y = A \cdot B \cdot C \cdot D \]

![Figure 4.2.2a: Function of 4 input NAND Gate.](image)

And the truth table:

<table>
<thead>
<tr>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>Y</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

**Table 7:** Truth table of 4 input NAND Gate (HD74HC20).
From Table 7 we can see if any of the independent variables A, B, C and D are 0 the output is 1. The only time the output is 0 is when A, B, C and D all are 1. In other words we can say if one, two, three or four of the flame sensors detect flame the output of the NAND gate will be high. If none of them detect flame only then the output is low.

Rather than soldering the HD74HC20 IC directly into the board socket headers were used, this was done so that the IC can be replaced if it is damaged in an event. Multiple male-male headers were soldered to the GND and Vcc pin of the IC; we took Vcc and GND connections of Flame sensors from these headers and only Vcc for the relay. Output pin 6 of NAND gate is connected to digital input pin 7 of the Arduino. Digital output pin 13 of Arduino is connected to relay through a buzzer and a LED. When the relay will switch the solenoid ON the buzzer will go off and the LED will glow showing fire is detected. If output of NAND gate is low (0), input pin of relay is made low (0) and if output of NAND gate is high (1), input pin of relay is made high (1).

We must understand how a relay module works before we further discuss the connections. A relay turns ON when it receives 0 and turns OFF when it receives 1. Its output pins are NO, COM and NC. We will be operating our relay in the manner shown in figure 4.2.2b below:

**Figure 4.2.2b:** Functionality of a relay in our project.
If the input signal to the relay is low, batteries stays disconnected from load, but if the signal is high then batteries are connected to the load. On pin 13 a buzzer is connected through a 1 kilo-ohm resistor and a BC547 transistor. This buzzer will stay ON as long as the load is connected to the batteries. The COM terminal of the relay is connected to the negative terminal of the batteries and the NC terminal is connected to the protection circuit. The other terminal of the protection circuit is connected to the positive terminal of the batteries. Here the load consists of a solenoid valve and an emergency light connected in parallel. The load is then again in parallel connected to the protection circuit.

Our Arduino is programmed in such a way so that once the batteries are connected they remain connected for 30 seconds. This 30 seconds delay has a very important purpose. As we know fire flickers as it burns due to wind and random distribution of fuel. Due to this reason the output signal of the Flame sensors change constantly. If the delay was not introduced the load would also constantly connect and disconnect. After several trials we found 30 seconds was enough time to suppress the fire for our system. This time delay can be increased or decreased depending upon the nature of fire.

4.2.3 Protection circuit against back electro motive force

4.2.3.1 Introduction to back electro motive force

Any electronic project may require driving an electromagnetic locking device. It can be a powerful electromagnet or a solenoid. All these devices are driven by same basic laws of physics. The electromagnet will store energy when powered and will produce a back electro motive force or counter electro motive force (CEMF) when the power supply is turned OFF. The disconnected electromagnet acts like a source of current, building a necessary voltage to keep the original current flowing.

When the power supply is connected, current flows through the windings of the electromagnet. The resting current is determined by the DC resistance of the coil winding and wires. This current gives rise to a magnetizing field which aligns the magnetic domains in the metal of the electromagnet. The field is re-enforced due to
this alignment which makes magnetic force greater and also stores a lot of energy in the process.

**Figure 4.2.3.1a:** When electromagnet is connected to power supply.

When power supply is turned OFF, the magnetic field will collapse. Doing so will generate an EMF or CEMF in the electromagnet’s coil as shown in figure

The generated EMF will produce a very large voltage that can:

- Damage the switch
- Produce interference
- Damage the rest of the circuitry
- Cause loss of data

**Figure 4.2.3.1b:** When power supply is disconnected from electromagnet.
Back EMF cannot be stopped completely, but it can be controlled. When suppressing the back EMF the goal is to restrict very high voltages and dissipate the stored energy in a controlled manner.

### 4.2.3.2 Our protection circuit

When power supply is connected the diode is reversed biased and does not play any role in the circuit. When the power supply is turned OFF a back EMF is produced in the opposite polarity and so the diode comes in series with the electromagnet. The diode does a good job of suppressing the back EMF which is suitable for small solenoids.

![Diagram of protection circuit](image)

**Figure 4.2.3.2a:** Design of our protection circuit, **4.2.3.2b:** Picture of our protection circuit.
4.3 Power management

The source of power for our entire setup is two 12V, 20AH batteries which in series is 24V. We figured that the solenoid requires the highest voltage, which is 24 V DC. The alarm system consists of an emergency light of 12V and a speaker of 12V as well, that sums up to 24V in series. Putting the solenoid and alarm system in parallel means the node will require 24V in total.

By using a 24V to 5V step down voltage regulator we were able to power the Arduino. The Arduino in turn powers the Flame sensors, relay, NAND gate and buzzer with 5V DC.

![Two 12V, 20AH batteries](image1)

**Figure 4.3a:** Two 12V, 20AH batteries.

![24V to 5V DC step down voltage converter](image2)

**Figure 4.3b:** 24V to 5V DC step down voltage converter.
4.4 Design and calculation of the demonstration room

The design of the room depended entirely upon the detection range of the Flame sensor. From the datasheet of the flame sensor and upon carefully experimenting with them we saw that the average range up to which the flame sensors detect small fire was 1 meter. Converting meter to inch we find that 1m equal to 39.3701 inches ≈ 40 inches. Therefore we designed a room that is 40 inches by 40 inches by 40 inches.

Figure 4.4: Demonstration room
4.5 Sprinkler System and calculations regarding the location of the flame sensors

4.5.1 Existing work of sprinkler heads

The mechanism is dependent upon a frangible glass bulb containing a heat responsive liquid. During an occurrence of fire the ambient temperature rises, this causes the liquid in the bulb to expand. When the ambient temperature reaches the rated temperature of the sprinkler head, the bulb breaks. After the bulb breaks it also removes the sealing of the sprinkler. Water is then directed towards the deflector. The deflector is designed to distribute the water in a pattern that is most effective in controlling the fire.

Standard spray coverage is up to: Light Hazard = 225 square feet and Ordinary Hazard = 130 square feet per N.F.P.A 13.

<table>
<thead>
<tr>
<th>Temperature Classification</th>
<th>Normal Sprinkler Temperature Rating</th>
<th>N.F.P.A Maximum Ambient (Ceiling) Temperature</th>
<th>Color of Glass Bulb</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ordinary**</td>
<td>155°F/57°C</td>
<td>100°F/38°C</td>
<td>Orange</td>
</tr>
<tr>
<td>Ordinary</td>
<td>155°F/68°C</td>
<td>100°F/38°C</td>
<td>Red</td>
</tr>
<tr>
<td>Intermediate</td>
<td>175°F/79°C</td>
<td>150°F/65°C</td>
<td>Yellow</td>
</tr>
<tr>
<td>Intermediate</td>
<td>200°F/93°C</td>
<td>150°F/65°C</td>
<td>Green</td>
</tr>
<tr>
<td>High*</td>
<td>286°F/141°C</td>
<td>225°F/107°C</td>
<td>Blue</td>
</tr>
<tr>
<td>Extra High*</td>
<td>360°F/182°C</td>
<td>300°F/149°C</td>
<td>Mauve</td>
</tr>
<tr>
<td>Open*</td>
<td>Open</td>
<td>N/A</td>
<td>No Bulb</td>
</tr>
</tbody>
</table>

Table 8: Ratings of sprinkler head. [25]
Figure 4.5.1: Radius of distribution of water (centimeter) with respect to pressure (Pound per square inch) and mounting height (meter). [25]
4.5.2 Step by step description of our sprinkler system and its calculations

The sprinkler system is place on the ceiling of the room 40 inches above the ground. Dividing the 40 inches by 40 inches ceiling into four equal parts we get a 20 inches by 20 inches square. We placed the sprinkler head exactly at the center of this square.

From figure 4.5.1 we can see the graph of pressure against spreading radius and ceiling height. The pressure of our water supply was about 20 psi, dividing that into four sprinklers gives 5 psi on each sprinkler. While carrying out the experiment we saw that the radius of water coverage was about 40 to 45cm ≈ 18 inches. This was more than enough for our experiment and justified the use of four sprinklers.
Figure 4.5.2b: Placement of a single sprinkler.

Figure 4.5.2c: Top view of sprinkler system.
4.5.3 Placement and calculations regarding the location of the flame sensors

As mentioned earlier in this chapter, four flame sensors were used in our demonstration room. After practically studying the behavior of the one sensor we found that placing the flame sensor like the sprinkler heads right at the center of the 20 inches by 20 inches square after dividing the 40 inches by 40 inches ceiling into four equal parts will cover the detection area. Coincidentally the location of a sprinkler head and a flame sensor were together.

The datasheet of a flame sensor states that the detection angle is 60°. However after several experiments with one we found that in reality its detection angle is about half of that is stated in the datasheet that is 30°.

Here,

H = height of ceiling = 40 inches,

θ = half of that of detection angle = 30° / 2

= 15°

R = radius of detection

Therefore to calculate the radius:

\[
tan 30° = \frac{R}{40 \text{ inches}}
\]

=> \( R = tan 30° \times 40 \text{ inches} \)

=> \( R = 10.7179 \text{ inches} \)

**Figure 4.5.3a:** Radius of detection.
From the figure 4.5.3b above we can see that the red zones indicate the area of fire detection. We can also tell that some areas of the demonstration room do not fall in the detection area. However after carrying out the experiment we found that as fire flickers in a random pattern when it burns it crosses the detection area and triggers the sensors. We placed fire exactly in the center of the room and it was successfully suppressed as shown in the figure 4.5.3c and 4.5.3d below:

**Figure 4.5.3b:** Fire detection area of four flame sensors.
Figure 4.5.3c: Fire detected in the center of the demonstration room. Water is flowing through the solenoid and out the sprinkler heads.

Figure 4.5.3d: Fire suppressed successfully
4.6 Control Algorithm

4.6.1 Arduino Code

```c
const int NANDPin = 7; // the number of the NAND pin
const int RELAYPin = 13; // the number of the RELAY pin
//char myChar = 'RELAY ON';
//char y = "RELAY OFF";
// variables will change:
int NANDState = 0;  // variable for reading the NAND status

void setup() {
    // initialize the RELAY pin as an output:
    pinMode(RELAYPin, OUTPUT);
    // initialize the NAND pin as an input:
    pinMode(NANDPin, INPUT_PULLUP);
    // start serial port at 9600 bps:
    Serial.begin(9600);
}

void loop() {
    // read the state of the NAND value:
    NANDState = digitalRead(NANDPin);

    // check if the NAND is ON. If it is, the NANDState is HIGH:
    if (NANDState == HIGH) {
        digitalWrite(RELAYPin, HIGH);
        Serial.print("RELAY ON");  // prints a label
        Serial.print("\t");  // prints a label
        delay(30000);  // Relay will stay ON for 30 seconds
    }
    if (NANDState == LOW) {
        digitalWrite(RELAYPin, LOW);
        Serial.print("RELAY OFF");
        Serial.print("\t");  // prints a label
        delayMicroseconds(50);
    }
}
```

**Figure 4.6.1a:** Arduino code of an automated fire suppression mechanism controlled using an Arduino.
Figure 4.6.1b: Flowchart of the Arduino code of an automated fire suppression mechanism controlled using an Arduino.
4.6.2 Explanation of the code

First the NAND gate is stated as pin 7 and Relay as pin 13. The initial state of the NAND gate is stated as LOW (0). It is then declared that the Relay is connected as an output and NAND gate is connected as an input to the Arduino. Please refer to Figure 4.2.2b before reading further.

A loop is then run which will constantly read the state of pin 7 that is the output of NAND gate. It checks if the Pin 7 is HIGH (1) or LOW (0).

An “if” condition is run where it states that if the output of NAND gate pin 7 is HIGH (1) the Relay pin 13 will be made HIGH (1). A delay of 30000 milliseconds that is 30 seconds is made. This 30 seconds delay is made so that the Relay pin 13 stays HIGH (1) for 30 seconds.

Again an “if” condition is run which state what happens when the output of NAND gate pin 7 is LOW (0). When the output of NAND gate pin 7 is LOW (0), the input of Relay pin 13 will be made LOW (0).

In the end the delay of 50 microseconds is made just for the Arduino to process the whole information. In both “if” conditions a “Serial.println” command is given, if we connect the Arduino to a computer we will be able to monitor the state of Relay pin 13.
4.7 Conclusion

Main electrical connections, conditions, calculations and working principles were elaborately described in this chapter. The chapter started with the detailed conception of the electrical connections used in our system through step by step description. Then it was followed by the function of the HD74HC20 IC, that is how it compares output signals from all four flame sensors. It was explained how the protection circuit that we used suppresses back EMF. Power management that is the two 12V, 20 AH batteries has been shown. The demonstration of the project including the demonstration room, placement of the sprinklers and flame sensors along with measurements, calculations and diagrams were also shown with the aid of diagrams, tables and calculations. Finally the chapter was concluded with the control algorithm we used to run our system. The code has been sequentially explained along with a flow chart. All in all, this chapter covers the most important and fundamental mechanism of our system.
Chapter 5
Field Test and Analysis

5.1 Wavelength of fire

5.1.1 Effect of sensor triggering wavelength

![Figure 5.1.1a](image)

**Figure 5.1.1a:** The spectral range of an electromagnetic spectrum. [26]

Electromagnetic spectrum is the entire range, scope (spectrum) of frequencies of electromagnetic radiation, their wavelengths and photon energies. The spectrum extends from frequencies that are low and wavelengths that are in nanometer. It starts from ultraviolet radiation and end at infrared radiation with visible light in the middle. Infrared radiation can be further divided into far infrared, mid infrared and near infrared. Figure 5.1.1a shows the near infrared radiation at the spectral range which ranges from around 700nm to 1200nm.

The flame sensor module that we used with the Arduino Uno in our project contains a YG1006 sensor which is a high speed and high sensitive NPN silicon phototransistor in a standard 5mm package. Its black epoxy makes the device sensitive to infrared radiation. This sensor can detect infrared radiation at a wavelength ranging from 760nm to 1100nm.
As we can see in figure 5.1.1b the YG1006 sensor in the flame sensor module can detect light at a wavelength range of 760nm to 1100nm. The range of the color of the light is from light orange to light red. This justifies that the flame sensor module will only respond to flames of color (light orange – light red). This sensor has high photo sensitivity and high response time which enables it for instantly detecting light within its wavelength range within milliseconds.

5.1.2 Effect of non-sensor triggering wavelengths

In topic 5.1.1 we have seen the wavelength range in the electromagnetic spectrum within which the flame sensor module we used in our project can detect radiation. In this chapter we will come across the range of wavelength in which the flame sensor does not trigger.

Figure 5.1.2: The ranges in which the flame sensor does not trigger or cannot detect any infrared. [26]
As we can see from figure 5.1.2, the YG1006 sensor present in the flame sensor module can only detect wavelengths from 760-1100nm, all the other ranges of wavelengths does not trigger the sensor at all. The ranges of non-triggering wavelengths are 200 to 760nm and 1100 to 1200nm. Hence we can conclude that the sensor cannot detect lights of colors purple, different shades of blue, green, and few shades of yellow and orange. All in all, the flame sensor cannot response to any of the ultraviolet and visible light spectrum.

5.2 Distance of triggering

In topic 3.5, we have come across the key features of our flame sensor module which stated that it has a maximum triggering distance of 100cm or 1m. In order to justify this feature, we conducted an experiment to check the distance upto which the sensor detects the presence of fire.

![Figure 5.2a: The flame sensor is ON; the switched OFF LED shows that fire is yet to be detected.](image)
The experiment showed how the flame sensor senses the fire over 100cm. We divided 100cm into five windows of 20cm. Then we placed the fire at each interval up to 100cm.

**Figure 5.2b:** The switched ON LED shows that the flame sensor detects fire at 20cm.
Figure 5.2c: The switched ON LED shows that the flame sensor detects fire at 40cm.

Figure 5.2d: The switched ON LED shows that the flame sensor detects fire at its maximum range of 100cm.
When the fire was placed at a distance more than 100cm, the sensitivity of the flame sensor got weaker with the increasing distance. However flame was being detected but the signal constantly flickered, which could be observed from the continuous switching and ON and OFF of the LED.

5.3 Electrical measures of flame sensor

The flame sensor module that we used in our project consists of certain components which include mainly the YG1006 NPN phototransistor sensor, the LM 393 comparator chip. The sensor typically operates at a voltage range of (3.3V - 5V).

The YG1006 sensor is a high speed and high sensitive NPN silicon phototransistor which is sensitive to infrared radiation. [10]

The absolute maximum readings at 25°C are as follows:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Symbol</th>
<th>rating</th>
<th>units</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collector Emitter Voltage</td>
<td>V_{CEO}</td>
<td>30</td>
<td>V</td>
</tr>
<tr>
<td>Emitter-Collector-Voltage</td>
<td>V_{CEO}</td>
<td>5</td>
<td>V</td>
</tr>
<tr>
<td>Collector Current</td>
<td>I_{c}</td>
<td>20</td>
<td>mA</td>
</tr>
<tr>
<td>Operating Temperature</td>
<td>T_{opr}</td>
<td>-25〜+85°C</td>
<td>°C</td>
</tr>
<tr>
<td>Storage Temperature</td>
<td>T_{stg}</td>
<td>-40〜+85°C</td>
<td>°C</td>
</tr>
<tr>
<td>Lead Soldering Temperature</td>
<td>T_{sol}</td>
<td>260</td>
<td>°C</td>
</tr>
<tr>
<td>Power Dissipation at (or below) 25°C FreeAir Temperature</td>
<td>P_{c}</td>
<td>75</td>
<td>mW</td>
</tr>
</tbody>
</table>

**Table 9:** The electro-optical characteristics of YG 1006 sensor (25°C). [10]
Table 10: The electro-optical characteristics include voltage, time, current and wavelength. [10]
Some typical electro optical curves in accordance with the electro optical characteristics are as follows:
Figure 5.3a: The typical electro optical curves for YG1006 sensor. [10]

Another key component of the flame sensor module is the LM 393 comparator chip.

The LM393 series are dual independent precision voltage comparators capable of single or split supply operation. These devices are designed to permit a common mode range–to–ground level with single supply operation. Input offset voltage specifications as low as 2.0 mV make this device an excellent selection for many applications in consumer auto-motive, and industrial electronics. [24]

- Wide Single–Supply Range: 2.0 Vdc to 36 Vdc
- Split–Supply Range: ±1.0 Vdc to ±18 Vdc
- Very Low Current Drain Independent of Supply Voltage: 0.4 mA
- Low Input Bias Current: 25 nA
- Low Input Offset Current: 5.0 nA
- Low Input Offset Voltage: 2.0 mV (max) LM393A 5.0 mV (max) LM293/393
• Input Common Mode Range to Ground Level

• Differential Input Voltage Range Equal to Power Supply Voltage

• Output Voltage Compatible with DTL, ECL, TTL, MOS, and CMOS Logic Levels

• ESD Clamps on the Inputs Increase the Ruggedness of the Device without Affecting Performance. [24]

**Figure 5.3b:** The schematic diagram of 1 LM 393 comparator chip. [24]
### Table 11: The maximum rating of a LM 393 comparator chip. [24]

<table>
<thead>
<tr>
<th>Rating</th>
<th>Symbol</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power Supply Voltage</td>
<td>$V_{CC}$</td>
<td>$+36$ or $±18$</td>
<td>$V_{dc}$</td>
</tr>
<tr>
<td>Input Differential Voltage Range</td>
<td>$V_{IDR}$</td>
<td>36</td>
<td>$V_{dc}$</td>
</tr>
<tr>
<td>Input Common Mode Voltage Range</td>
<td>$V_{ICR}$</td>
<td>$-0.3$ to $+36$</td>
<td>$V_{dc}$</td>
</tr>
<tr>
<td>Output Short Circuit-to-Ground</td>
<td>$I_{SC}$</td>
<td>Continuous</td>
<td>mA</td>
</tr>
<tr>
<td>Output Sink Current (Note 1)</td>
<td>$I_{Sink}$</td>
<td>20</td>
<td>mA</td>
</tr>
<tr>
<td>Power Dissipation @ $T_A = 25°C$</td>
<td>$P_D$</td>
<td>570</td>
<td>mW</td>
</tr>
<tr>
<td>Denote above 25°C</td>
<td>$1/R_{BJA}$</td>
<td>5.7</td>
<td>mW/$^°C$</td>
</tr>
<tr>
<td>Operating Ambient Temperature Range</td>
<td>$T_A$</td>
<td>$-25$ to $+85$</td>
<td>$^°C$</td>
</tr>
<tr>
<td>LM293, LM393, 393A, LM2903, LM2903V</td>
<td></td>
<td>0 to $+70$</td>
<td>$^°C$</td>
</tr>
<tr>
<td>Maximum Operating Junction Temperature</td>
<td>$T_{J(max)}$</td>
<td>125</td>
<td>$^°C$</td>
</tr>
<tr>
<td>LM393, 393A, 2903, LM2903V, LM293</td>
<td></td>
<td>-40 to $+105$</td>
<td>$^°C$</td>
</tr>
<tr>
<td>LM293</td>
<td></td>
<td>-40 to $+125$</td>
<td>$^°C$</td>
</tr>
<tr>
<td>Storage Temperature Range</td>
<td>$T_{stg}$</td>
<td>$-65$ to $+150$</td>
<td>$^°C$</td>
</tr>
</tbody>
</table>

### 5.4 Conclusion

The effect of sensor triggering and non-triggering wavelengths has been shown along with their diagrams. The triggering distance was explained by and experiment through several pictures and brief descriptions. Also the electrical measures across the YG1006 sensor and the LM393 chip have been looked upon with the aid of graphs, diagrams and tables.
Chapter 6
Conclusion and Future Work

6.1 Summary

Our project named as “An automated fire suppression mechanism controlled using an Arduino” mainly focuses on the idea of developing a system of extinguishing fire, which can be effective and efficient while being cheap and reasonable at the same time. Our system is effective in the sense that it can detect fire at a wide range of distance and efficient in the sense that it can extinguish fire within a short period of time. The components that we used for our project are readily found at the market and are quite reasonable in price. Unlike other projects on automated fire suppression mechanisms, our project focuses greatly on decreasing the delay which occurs between the ignition of fire and the extinguishing process. Through proper coding in the Arduino, we managed to establish a system which causes minimum delay between the time the flame sensors detect the fire and the flow of water out the sprinklers.

If we look at our system at a glance, we can summarize the whole project into these simple steps:

- First the flame sensor detects the fire at its certain wavelength range of 760nm-1100nm.
- The LED lights up and the buzzer goes off simultaneously.
- The flame sensors send the signal to the Arduino Uno through the comparator circuit.
- The Arduino Uno has been set up with required codes previously which come in terms.
- The Arduino provides the optimal voltage to the solenoid valves which then opens up and allows the water stored in the water reservoir beforehand to flow.
- The Arduino is connected to the solenoid valves through relay which says ON for a user defined time span.
• A protection circuit is used to prevent the flow of back EMF through the solenoid valve.
• The fire will be extinguished in no time!

6.2 Conclusion and future work

The project that we developed was intended for many future applications in our country. The main target customers are the Factories especially Garments factories, Banks and Slum areas. These areas will be hugely benefitted if they implement our system into their premises.

![System replaced with Thermo-graphic cameras and Raspberry Pi 3](image)

**Figure 6.2:** System replaced with Thermo-graphic cameras and Raspberry Pi 3.

The major improvement that can be made on this system is the end detector. Instead of using just flame detectors we can use Thermo-graphic cameras. The controlling unit can be replaced with a Raspberry Pi 3 which is more than 40 times
faster than an Arduino. The cameras can detect fire and its intensity. If the fire detected is large enough only then the solenoid valve will be turned ON. This will prevent any unwanted infrared from triggering the system such as cigarette amber. Raspberry Pi will be a better option in case of complex tasks such as image processing.

The slum areas are also vulnerable to such fire incidents at all times. The recent fire that occurred at the Korail slum located at Mohakhali, Dhaka ravaged and destroyed round 500 homes which left hundreds of people homeless. We believe our system can reduce the number of such incidents to a great extent if implemented according to the needs of the areas. For example, the slum areas are usually open areas and the sprinklers that we used have to cover a large distance. In order to achieve that, the water pressure through the water pipes can be increased so that the sprinklers can work effectively to cover the whole ignited area. Also, thin nozzles can be used to increase the pressure of water.

In case of the factories and banks, the flame sensors we used can be replaced by heat detectors for better sensitivity. The flame sensors we used were for our protocol purpose hence it can be replaced by better ones if needed. All in all, our system can be of great value to the government as they can implement this in their designated places and provide safety and security to the general public of our country.


[16] A Flame Sensor with Uniform Sensitivity Over a Large Field of View
CLIFFORD W. ERICKSON. IEEE TRANSACTIONS ON ELECTRON

AUTOMATIC FIRE FIGHTING SYSTEM. [Online] http://repository.sustech.edu,
Available at: http://repository.sustech.edu/bitstream/handle/123456789/13363/DESIGN%20AN

[18] Dual-band UV/IR Optical Sensors for Fire and Flame Detection and Target
10814 Atwell Dr., Houston, TX, 77096. Phone: 713-748-7926, fax : (713) 729-
2867, e-mail : dstarikov@imsensors.com, [Accessed 18 Nov. 2017].

[19] HIGH-TEMPERATURE OPTICAL FLAME SENSOR PE Sims, JD Lesko,
JA Cox Astro Power, Inc. E Chalfin Integrated Circuit Designs, Inc. JB Mc
Kitterick Allied Signal, Inc., Microelectronics & Technology Center [Accessed 18
Nov. 2017].

[20] SHAMS, R., FATEMA, N., PRIYOM, S. and HOSSAIN, S.
(2014). AUTOMATED FIRE EXTINGUISHING SYSTEM WITH GSM ALARM.
[Online] http://dspace.bracu.ac.bd, Available at:
http://dspace.bracu.ac.bd/bitstream/handle/10361/3227/10121051.pdf?sequence=1
[Accessed 20 Nov. 2017].

[21] Raspberry Pi based Smart Fire Management System employing Sensor based
Automatic Water Sprinkler Noorinder, Student Member IEEE Department of
Electronics and Communication Engineering, Punjabi University, Patiala, India
gillnoorinder@gmail.comJaspreetSingh,Member IEEE Department of Electronics
and Communication Engineering, Punjabi University, Patiala, India
singhjaspretpunni@gmail.com Ekambir Sidhu, Member IEEE/IETE Assistant
Professor, Department of Electronics and Communication Engineering Punjabi
University, Patiala, India esidhu@pbi.ac.in,[Accessed 20 Nov. 2017].
[22] Revealing fire survivors hidden behind smoke and flames by IR active imaging systems V. Bianco1, M. Paturzo1, A. Finizio1, L. Miccio1and P. Ferraro1. 1CNR—National Institute of Optics Via Campi Flegrei 34, I-80078, Pozzuoli (Naples), Italy vittorio.bianco@ino.it. [Accessed 20 Nov. 2017].


Appendix A: Electrical connection with all components running for a single sprinkler.

Figure Appendix A: Electrical connection with all components running for a single sprinkler.
Appendix B: Diagram of an automated fire suppression mechanism controlled using an Arduino for a single room.

Figure Appendix B: Diagram of an automated fire suppression mechanism controlled using an Arduino for a single room.
Appendix C: Sprinkler system for single room.

Figure Appendix C: Sprinkler system for single room.
Appendix D: Comparator circuit with buzzer.

Figure Appendix D: Comparator circuit with buzzer and LED.
Appendix E: Single sprinkler and flame sensor front view.

*Figure Appendix E:* Single sprinkler and flame sensor front view.
Appendix F: Controlling circuit along with the Flame sensors.

Figure Appendix F: Controlling circuit along with the Flame sensors.
Appendix G: Flowchart of the fire extinguishing process.

Figure Appendix G: Flowchart of the fire extinguishing process.