DESIGN AND IMPLEMENTATION OF SONIC BASED FIRE EXTINGUISHER SYSTEM

Mukil Prasath P K ECE
Sri Sairam Engineering college
Chennai, India

Vineeth S ECE
Sri Sairam Engineering College
Chennai, India

Dhilip Kumar M ECE
Sri Sairam Engineering college
Chennai, India

Mr. D. Senthilkumar
Asst. Professor
Sri Sairam Engineering College

Abstract— It is known that the technological advancements are increasing at a faster pace. But the utilization of technologies in various sectors are very low. People have witnessed various accidents caused by fire. The Common system consists of a water based automated fire extinguisher system. This might cause water shortage if the water level is very low. This cannot reduce the fire in critical situations. The system is only partly efficient. So we propose a system where we can efficiently control the fire using Sound based Fire extinguisher. The sound waves are utilized to reduce the fire by performing various processes. The proposed system overcomes the drawback in the existing system. The system here consists of a Frequency generator. The frequency generator is used to generate a signal of particular frequency. Based on the user input a particular frequency is provided to the PWM driver. The PWM driver is controlled by the Arduino controller. Now the PWM driver provides the frequency to the amplifier circuit. The amplifier amplifies the frequency and provides the output to the FET transistor. The output generated is efficient to clear the fire present in an area. If fire occurs in office it will be detected using temperature sensor and sonic fire extinguisher will be activated. Using camera located in that place also the fire will be detected and fire extinguisher will be activated.

Keywords: PWMdriver, FET, Arduino controller.

I .INTRODUCTION

The Common system consists of a water based automated fire extinguishersystem. This might cause water shortage if the water level is very low. This cannot reduce the fire in critical situations.So we have designed a Sonic Fire Extinguisher Extinguish Fire by Sound Waves. Generally, fire is extinguished with the help of water or carbon dioxide but by this Extinguishing fire through sound bass sound. The sound waves are utilized to reduce the fire by performing various processes. The proposed system overcomes the drawback in the existing system.Firefighting in an enclosed space has always been a problem,other than accessibility for the fire extinguisher technology to the closed space is a major challenge. A compact , independent and reliable fire extinguisher is required in order to overcome this problem. Space station and submarine are the main examples of the application that highly required new fire extinguisher technology that will be able to used in a confined and very limited space. Fire manipulation using sound was not a new technique. The interactions between sound and flames was first reported by John Leconte in 1858, who noted flames within anorchestral respond to beats within music. A German physicist, Heinrich Rubens in the 1900's, showed the technique using a section of pipe with holes perforated along the top. Subsequently, the height of the flames could be manipulated. In many offices, shopping malls and many public places the location of fire extinguisher may not be known to the people. The water form sprinklers may cause the fire to spread and chemicals from fire extinguisher may cause any adverse effects.

So this automatic sonic fire extinguishers are used to detect and handle the fire in the absence of human. The proposed system overcomes the drawback in the existing system. The system here consists of a Frequency generator. The frequency generator is used to generate a signal of particular frequency. Based on the user input a particular frequency is provided to the PWM driver. The PWM driver is controlled by the Arduino controller. Now the PWM driver provides the frequency to the amplifier circuit. The amplifier amplifies the frequency and provides the output to the FET transistor. The output generated is efficient to clear the fire present in an area. If fire occurs in office it will be detected using temperature sensor and sonic fire extinguisher will be activated. Using camera located in that place also the fire will be detected and fire extinguisher will be activated. As for the sound wave propagation simulation, it is recommended that the simulation in other Multiphysics software such as COMSOL, to further verify the simulation results. Apart from that, another simulation could be performed, coupled with fluid dynamics of fire and acoustics to study how the fire is being extinguished with sound (acoustics) interaction. In the experiment part, different parameters could be used to further explore is study such as using different intensity of sound (by using different speaker power rating),positioning of sound towards the fire source and size of flame (or flame intensity) & varying design of extinguisher.Fire extinguishers are trying to eradicate one of the elements in the pyramid (a flametetrahedron) in order to eliminate the flame. Firefighting in an enclosed space has been a problem, other than the accessibility for the fire fighte r to access the place,accessing the water, carbon dioxide (CO2) or other fire extinguisher technology to the closed space is a major challenge. A compact, independent and reliable fire extinguisher is required in order to overcome this problem. Space station and submarine are the main examples of the application that highly required new fire extinguisher technology that will be able to be used in a confined and very limited space. Fire manipulation using sound was not a new technique. The interactions between sound and flames was first reported by John Leconte in 1858, who noted flames within anorchestral respond to beats within music. Subsequently, the height of the flames could be manipulated. The amplifier amplifies the frequency and provides the output to the FET transistor. The output generated is efficient to clear the fire present in an area. If fire occurs in office it will be detected using temperature sensor and sonic fire extinguisher will be activated. Using camera located in that place also the fire will be detected and fire extinguisher will be activated.

II METHODOLOGY

It is known that the technological advancements are increasing at a faster pace. But the utilization of technologies in various sectors are very low. People have witnessed various accidents caused by fire. The common method we use for fire reduction is done using water or sand based extinguishers. But in some cases the presence of these, may become shortage enough to reduce the fire. So here comes a system where fire can be efficiently controlled using Sound based Fire extinguisher. This system also consumes less power as the camera is activated and takes snapshots only in the presence of an intruder, unlike the CCTV system, where live feed is recorded round the clock. This system not only notifies the user but also the neighbour so that immediate action could be taken.

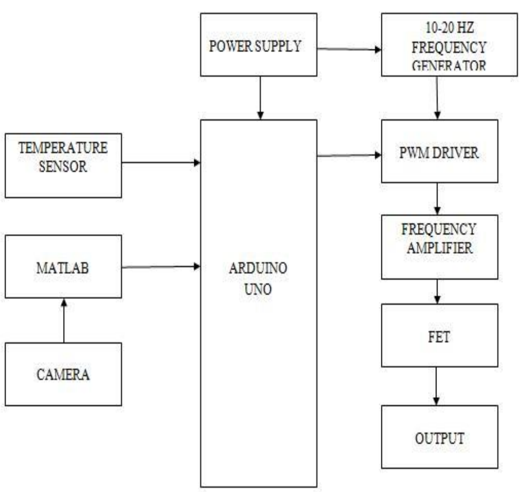


Fig 2.1 Block diagram

III POWER SUPPLY

3.1 TRANSFORMER

The potential transformer will step down the power supply voltage (0-230V) to (0-6V) level. Then the secondary of the potential transformer will be connected to the precision rectifier, which is constructed with the help of op–amp. The advantages of using precision rectifier are it will give peak voltage output as DC, rest of the circuits will give only RMS output. . The amount of amplification provided by an amplifier is measured by its gain: the ratio of output voltage, current, or power to input. An amplifier is a circuit that has a power gain greater than one. An amplifier can either be a separate piece of equipment or an electrical circuit contained within another device. Amplification is fundamental to modern electronics, and amplifiers are widely used in almost all electronic equipment.

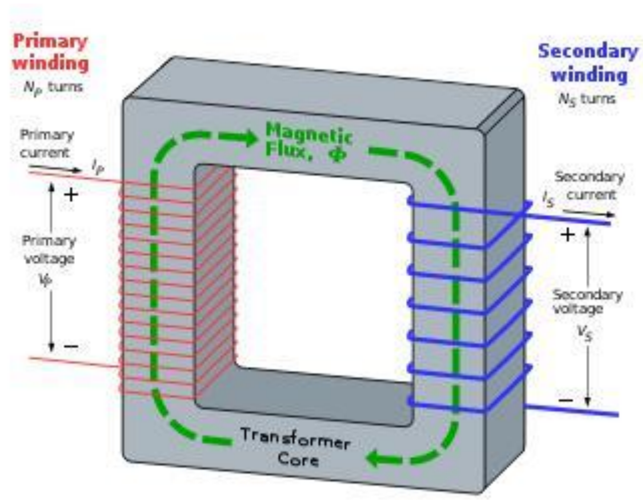


Fig 3.2

Transformer **3.2 BRIDGE RECTIFIER**

When four diodes are connected as shown in figure, the circuit is called as bridge rectifier. The input to the circuit is applied to the diagonally opposite corners of the network, and the output is taken from the remaining two corners. Let us assume that the transformer is working properly and there is a positive potential, at point A and a negative potential at point B, the positive potential at point A will forward bias D3 and reverse bias D4. The negative potential at point B will forward bias D1 and reverse D2. At this time D3 and D1 are forward biased and will allow current flow to pass through them; D4 and D2 are reverse biased and will block current flow.The path for current flow is from point B through D1, up through RL, through D3, through the secondary of the transformer back to point B. this path is indicated by the solid arrows. Waveforms (1) and (2) can be observed across D1 and D3.One-half cycle later the polarity across the secondary of the transformer reverse, forward biasing D2 and D4 and reverse biasing D1 and D3.Current flow will now be from point A through D4, up through RL, through D2, through the secondary of T1, and back to point A. This path is indicated by the broken arrows. Waveforms (3) and (4) can be observed across D2 and D4. The current flow through RL is always in the same direction. In flowing through RL this current develops a voltage corresponding to that shown waveform (5). Since current flows through the load (RL) during both half cycles of the applied voltage, this bridge rectifier is a full-wave rectifier.

3.3 IC VOLTAGE REGULATOR

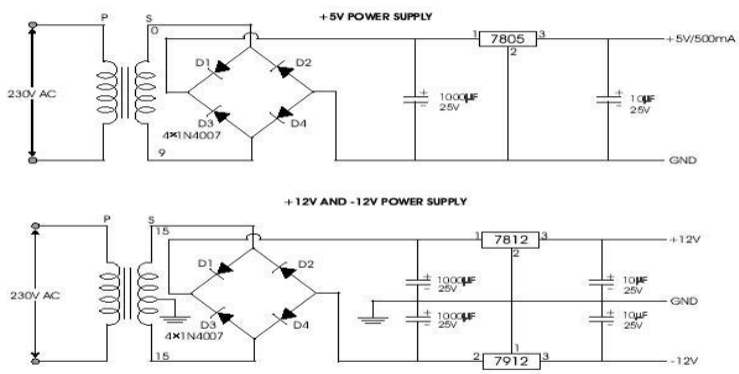
Voltage regulators comprise a class of widely used ICs. Regulator IC units contain the circuitry for reference source, comparator amplifier, control device, and overload protection all in a single IC. IC units provide regulation of either a fixed positive voltage, a fixed negative voltage, or an adjustably set voltage. The regulators can be selected for operation with load currents from hundreds of milli amperes to tens of amperes, corresponding to power ratings from milli watts to tens of watts

Fig 3.3 IC voltge regulator

3.4 ARDUINO UNO

3.4.1 GENERAL DESCRIPTION

Arduino is an opensource project that created microcontroller-based kits for building digital devices and interactive objects that can sense and control physical devices. The project is based on microcontroller board designs, produced by several vendors, using various microcontrollers. These systems provide sets of digital and analog input/output (I/O) pins that can interface to various expansion boards (termed shields) and other circuits. The



boards feature serial communication interfaces, including Universal Serial Bus (USB) on some models, for loading programs from personal computers.

3.4.2 PRODUCT DESCRIPTION

Arduino Uno is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button.

It contains everything needed to support the microcontroller.

Fig 3.4 Arduino UNO

3.5 AMPLIFIER

An amplifier, electronic amplifier or (informally) amp is an electronic device that can increase the power of a signal (a time-varying voltage or current).It is a two-port electronic circuit that uses electric power from a power supply to increase the amplitude of a signal applied to its input terminals, producing a proportionally greater amplitude signal at its output. The amount of amplification provided by an amplifier is measured by its gain: the ratio of output voltage, current, or power to input. An amplifier is a circuit that has a power gain greater than one.An amplifier can either be a separate piece of equipment or an electrical circuit contained within another device. Amplification is fundamental to modern electronics, and amplifiers are widely used in almost all electronic equipment.

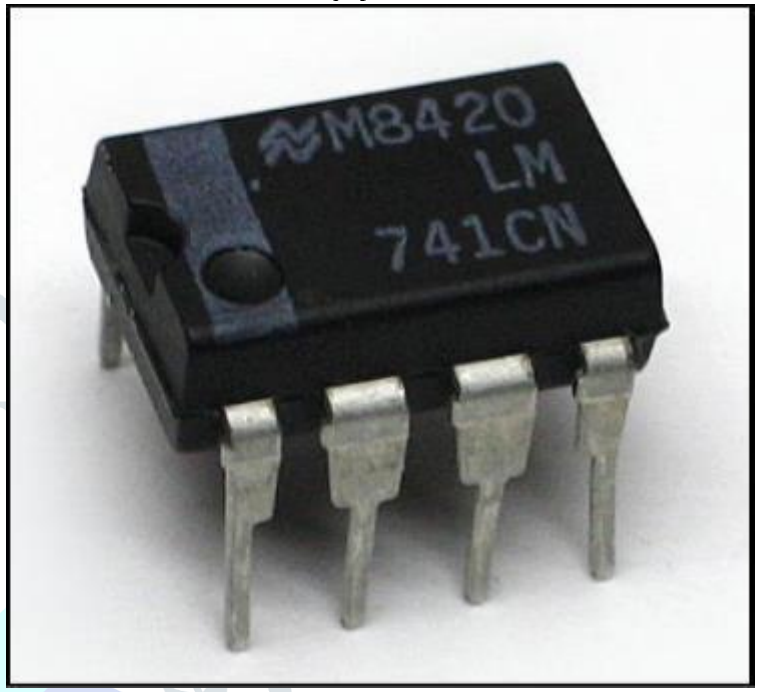


Fig 3.5 LM741

3.6 OPERATIONAL AMPLIFIERS (op-amps)

An operational amplifier is an amplifier circuit which typically has very high open loop gain and differential inputs. Op amps have become very widely used as standardized "gain blocks" in circuits due to their versatility; their gain, bandwidth and other characteristics can be controlled by feedback through an external circuit. Though the term today commonly applies to integrated circuits, the original operational amplifier design used valves, and later designs used discrete transistor circuits. A fully differential amplifier is similar to the operational amplifier, but also has differential outputs. These are usually constructed using BJTs or FETs.

3.6 TEMPERATURE SENSOR USING LM35

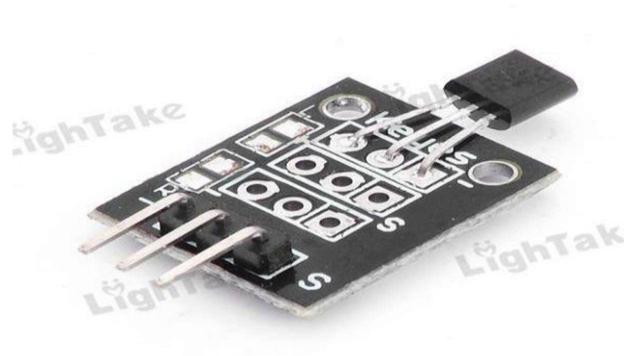


Fig 3.6 LM35

3.6.1 WORKING PRINCIPLE

-> It's a diode type temperature sensor, which is linear, about 2 mV per ºC, which is then amplified and offset so that you can get a direct reading of 10 mV per ºC.

->Laser trimming ensures the specified accuracy.

3.7 SIGNAL GENERATOR

A signal generator is an electronic device that generates repeating or non-repeating electronic signals in either the analog or the digital domain. It is generally used in designing, testing, troubleshooting, and repairing electronic or electroacoustic devices, though it often has artistic uses as well. There are many different types of signal generators with different purposes and applications and at varying levels of expense. These types include function generators, RF and microwave signal generators, pitch generators, arbitrary waveform generators, digital pattern generators and frequency generators. In general, no device is suitable for all possible applications.

3.8 VECTOR SIGNAL GENERATOR

A vector signal generator with the advent of digital communications systems, it is no longer possible to adequately test these systems with traditional analog signal generators. This has led to the development of the vector signal generator, which is also known as a digital signal generator. These signal generators are capable of generating digitally-modulated radio signals that may use any of a large number of digital modulation formats such as QAM, QPSK, FSK, BPSK, and OFDM. In addition, since modern commercial digital communication systems are almost all based on well-defined industry standards, many vector signal generators can generate signals based on these standards. Examples include GSM, W-CDMA (UMTS), CDMA2000, LTE, Wi-Fi (IEEE 802.11), and WiMAX (IEEE 802.16). In contrast, military communication systems such as JTRS, which place a great deal of importance on robustness and information security, typically use very proprietary methods..

3.9 TEMPERATURE SENSOR

The first slave connected to a temperature sensor LM35. This senses the temperature of an engine and provides the level of temperature.

3.9.1 GENERAL DESCRIPTION

The LM35 series are precision integrated-circuit temperature sensors, whose output voltage is linearly proportional to the Celsius (Centigrade) temperature. The LM35 thus has an advantage over linear temperature sensors calibrated in Kelvin, as the user is not required to subtract a large constant voltage from its output to obtain convenient Centigrade scaling.

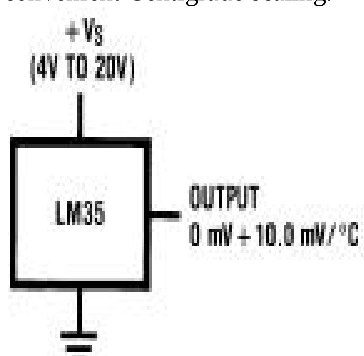


Fig 3.7 Basic Centigrade Temperature Sensor (+2°C to +150°C)

The LM35 does not require any external calibration or trimming to provide typical accuracies of ±1⁄4°C at room temperature and ±3⁄4°C over a full −55 to +150°C temperature range. The LM35's low output impedance, linear output, and precise inherent calibration make interfacing to readout or control circuitry especially easy. It can be used with single power supplies, or with plus and minus supplies. As it draws only 60 µA from its supply, it has very low self-heating, less than 0.1°C in still air. The LM35 is rated to operate over a −55° to +150°C temperature range, while the LM35C is rated for a −40° to +110°C range (−10° with improved accuracy). The LM35 series is available packaged in hermetic TO-46 transistor packages, while the LM35C, LM35CA, and LM35D are also available in the plastic TO-92 transistor package. The LM35D is also available in an 8-lead surface mount small outline package and a plastic TO-220 package.

3.9.2 CAPACITIVE LOAD

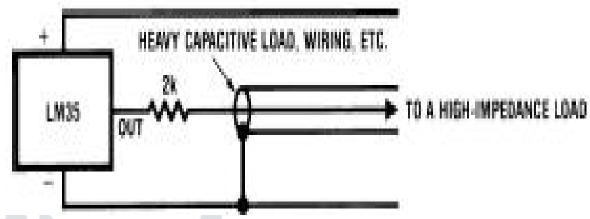


Fig 3.8LM35 With Decoupling From Capacitive Load

Like most micro power circuits, the LM35 has a limited ability to drive heavy capacitive loads. The LM35 by itself is able to drive 50 pf without special precautions. If heavier loads are anticipated, it is easy to isolate or decouple the load with a resistor.

IV SOFTWARE REQUIREMENTS

4.1 EMBEDDED C

High-level language programming has long been in use for embedded-systems development. However, assembly programming still prevails, particularly for digital-signal processor (DSP) based systems. DSPs are often programmed in assembly language by programmers who know the processor architecture inside out. The key motivation for this practice is performance, despite the disadvantages of assembly programming when compared to high-level language programming.If the video decoding takes 80 percent of the CPU-cycle budget instead of 90 percent, for instance, there are twice as many cycles available for audio processing. This coupling of performance to end-user features is characteristic of many of the real-time applications in which DSP processors are applied. DSPs have a highly specialized architecture to achieve the performance requirements for signal processing applications within the limits of cost and power consumption set for consumer applications. Unlike a conventional Load-Store (RISC) architecture, DSPs have a data path with memory-access units that directly feed into the arithmetic units. Address registers are taken out of the general-purpose register file and placed next to the memory units in a separate register file.Communication protocols become more sophisticated and require much more code to implement. In certain markets, multiple protocol stacks are implemented to be compatible with multiple service providers. In addition, backward compatibility with older protocols is needed to stay synchronized with provider networks that are in a slow process of upgrading. Today, most embedded processors are offered with C compilers. Despite this, programming DSPs is still done in assembly for the signal processing parts or, at best, by using assembly-written libraries supplied by manufacturers. The key reason for this is that although the architecture is well matched to the requirements of the signal-processing application, there is no way to express the algorithms efficiently and in a natural way in Standard C. Saturated arithmetic.

4.1.1 DESCRIPTION

Embedded C is designed to bridge the performance mismatch between Standard C and the embedded hardware and application architecture. It extends the C language with the primitives that are needed by signal-processing applications and that are commonly provided by DSP processors.The Embedded C specification extends the C language to support freestanding embedded processors in exploiting the multiple address space functionality, user-defined named address spaces, and direct access to processor and I/O registers. These features are common for the small, embedded processors used in most consumer products. The features introduced by Embedded C are fixed-point and saturated arithmetic, segmented memory spaces, and hardware I/O addressing. The description we present here addresses the extensions from a language-design perspective, as opposed to the programmer or processor architecture perspective.

4.1.2MULTIPLE ADDRESS SPACES

Embedded C supports the multiple address spaces found in most embedded systems. It provides a formal mechanism for C applications to directly access those individual processor instructions that are designed for optimal memory access. The Embedded C extension supports defining both the natural multiple address space built into a processor's architecture and the application-specific address space that can help define the solution to a problem.Embedded C uses address space qualifiers to identify specific memory spaces in variable declarations. There are no predefined keywords for this, as the actual memory segmentation is left to the implementation. As an example, assume that **X** and **Y** are memory qualifiers. The definition:

X int a[25];

Means that **a** is an array of 25 integers, which is locatedin the **X** memory. Similarly (but less common):

X int*Yp;

Means that the pointerp is stored in the Y memory. This pointer points to integer data that is located in the **X** memory. If no memory qualifiers are used, the data is stored into unqualified memory. For proper integration with the C language, a memory structure is specified, where the unqualified memory encompasses all other memories. All unqualified pointers are pointers into this unqualified memory. The unqualified memory abstraction is needed to keep the compatibility of the void ***** type, the NULL pointer, and to avoid duplication of all library code that accesses memory through pointers that are passed as parameters

4.1.3NAMED REGISTERS

Embedded C allows direct access to processor registers that are not addressable in any of the machine's address spaces. The processor registers are defined by the compiler-specific, named-register, storage class for each supported processor. The processor registers are declared and used like conventional C variables (in many cases volatile variables). Developers using Embedded C can now develop their applications, including direct access to the condition code register and other processor-specific status flags, in a high-level language, instead of inline assembly code named address spaces and full processor access reduces application dependency on assembly code and shifts the responsibility for computing data types, array and structure offsets, and all those things that C compilers routinely and easily do from developers to compilers.

4.1.5 I/O HARDWARE ADDRESSING

The motivation to include primitives for I/O hardware addressing in Embedded C is to improve the portability of device-driver code. In principle, a hardware device driver should only be concerned with the device itself. The driver operates on the device through device registers**,** which are device specific. However, the method to access these registers can be very different on different systems, even though it is the same device that is connected. The I/O hardware access primitives aim to create a layer that abstracts the system-specific access method from the device that is accessed. The ultimate goal is to allow source-code portability of device drivers between different systems. In the design of the I/O hardware-addressing interface, three requirements needed to be fulfilled:

->The device-drive source code must be portable.

->The interface must not prevent implementations from producing machine code that is efficient as other methods.

->The design should permit encapsulation of the system-dependent access method.

The design is based on a small collection of functions that are specified in the <iohw.h> include file. These interfaces are divided into two groups; one group provides access to the device, and the second group maintains the access method abstraction itself.

4.1.4 EMBEDDED C PORTABILITY

By design, a number of properties in Embedded C are left implementation defined. This implies that the portability of Embedded C programs is not always guaranteed. Embedded C provides access to the performance features of DSPs. As not all processors are equal, not all Embedded C implementations can be equal For example, suppose an application requires 24-bit fixed-point arithmetic and an Embedded C implementation provides only 16 bits because that is the native size of the processor. When the algorithm is expressed in Embedded C, it will not produce outputs of the right precision. In such a case, there is a mismatch between the requirements of the application and the capabilities of the processor. Under no circumstances, including the use of assembly, will the algorithm run efficiently on such a processor. Embedded C cannot overcome such discrepancies. Yet, Embedded C provides a great improvement in the portability and software engineering of embedded applications.Despite many differences between performance-specific processors, there is a remarkable similarity in the special-purpose features that they provide to speed up applications. Writing C code with the low-level processor-specific support may at first appear to have many of the portability problems usually associated with assembly code. In the limited experience with porting applications that use Embedded C extensions, an automotive engine controller application (about 8000 lines of source) was ported from the eTPU, a 24-bit special-purpose processor.

4.2 MANIPULATING MATRICES

4.2.1 ENTERING MATRICES

The best way for you to get started with MATLAB is to learn how to handle matrices. Start MATLAB and follow along with each example.

You can enter matrices into MATLAB in several different ways:

->Enter an explicit list of elements.

MATLAB displays the matrix you just entered.

16	3	2	13
5	10	11	8
9	6	7	12
4	15	14	1

This exactly matches the numbers in the engraving. Once you have entered the matrix, it is automatically remembered in the MATLAB workspace

V RESULTS

The main Objective of the project is to identify the frequency range that will be able to suppress an open flame and to analyze the physics of sound flame interactions.

Fig 4.1

Fig 4.2

->The proposed system overcomes the drawback in the existing system.

->The system here consists of a Frequency generator. The frequency generator is used to generate a signal of particular frequency.

-> Based on the user input a particular frequency is provided to the PWM driver.

->The PWM driver is controlled by the Arduino controller. Now the PWM driver provides the frequency to the amplifier circuit.

->The amplifier amplifies the frequency and provides the output to the FET transistor.

->The output generated is efficient to clear the fire present in an area.

VI CONCLUSION

In this work, we present a system where we can efficiently control the fire using Sound based Fire extinguisher. The sound waves are utilized to reduce the fire by performing various processes. The existing system consists of a water based automated fire extinguisher system. This might cause water shortage if the water level is very low. This cannot reduce the fire in critical situations. The system is only partly efficient.People have witnessed various accidents caused by fire. The common method we use for fire reduction is done using water or sand based extinguishers. But in some cases the presence of these, may become shortage enough to reduce the fire. So we propose a system where we can efficiently control the fire using Sound based Fire extinguisher. The sound waves are utilized to reduce the fire by performing various processes.

6.1 FUTURE SCOPE

As for the sound wave propagation simulation,it is recommended that the simulation in other Multiphysics software such as COMSOL, to further verify the simulation results.Apart from that, another simulation could be performed, coupled with fluid dynamics of fire and acoustics to study how the fire is being extinguished with sound (acoustics) interaction.In the experiment part,different parameters could be used to further explore is study such as using different intensity of sound (by using different speaker power rating),positioning of sound towards the fire source and size of flame (or flame intensity) & varying design of extinguisher .

REFERENCES

[1] M. Kobes, I. Helsloot, B. de Vries, and J. G. Post, "Building safety and human behaviour in fire: A literature review," *Fire Safety J.*, vol. 45, no. 1, pp. 1–11, 2010.

[2] E. Ronchi, E. D. Kuligowski, R. D. Peacock, and P. A. Reneke, "A probabilistic approach for the analysis of evacuation movement data," *Fire Safety J.*, vol. 63, no. 1, pp. 69–78, 2013.

[3] A. Cuesta, O. Abreu, and D. Alvear, "Methods for measuring collective behaviour in evacuees," *Safety Sci*, vol. 88, pp. 54–63, Oct. 2013.

[4] A. Kneidl, D. Hartmann, and A. Borrmann, "A hybrid multi-scale approach for simulation of pedestrian dynamics," *Transp. Res. C, Emerg. Technol.*, vol. 37, pp. 223–237, Dec. 2016.

[5] E. Kuligowski, "Predicting human behavior during fires," *Fire Technol.*, vol. 49, no. 1, pp. 101–120, 2013.

[6] B. L. Mesmer and C. Bloebaum, "Incorporation of decision, game, and Bayesian game theory in an emergency evacuation exit decision model," *Fire Safety J.*, vol. 67, pp. 121–134, Jul. 2014.

[7] R. Lovreglio, D. Borri, L. dell'Olio, and A. Ibeas, "A discrete choice model based on random utilities for exit choice in emergency evacuations," *Safety Sci.*, vol. 62, pp. 418–426, Feb. 2014.

[8] M. Haghani and M. Sarvi, "Human exit choice in crowded built environments: Investigating underlying behavioural differences between normal egress and emergency evacuations," *Fire Safety J.*, vol. 85, pp. 1–9, Oct. 2016.

[9] N. W. F. Bode and E. A. Codling, "Human exit route choice in virtual crowd evacuations," *Animal Behaviour*, vol. 86, no. 2, pp. 347–358, 2016.

[10] R. Lovreglio, A. Fonzone, and L. dell'Olio, "A mixed logit model for predicting exit choice during building evacuations," *Transp. Res. A, Policy Pract.*, vol. 92, no. 1, pp. 59–75, 2016.