DUBLIN CITY UNIVERSITY
SCHOOL OF ELECTRONIC ENGINEERING

Tactile Smoke Detector Interface

David Burson
57618514
April 2013

Bachelor Of Engineering

IN

ELECTRONIC ENGINEERING

Supervised by Prof. Barry McMullin

Acknowledgements

I wish to thank my project supervisor, Professor Barry McMullin, for his guidance and tutelage while undertaking this project. His willingness to lend his time to my project while also supervising other projects and also meeting his professional obligations was recognised and greatly appreciated.

I would also like to sincerely thank Mr Billy Roarty of the School of Engineering for being constantly available to provide expert advice, assistance and any tools or components that were required during the technical development of the project.

Finally, I would like to thank my family for their heartfelt support and guidance throughout my final year of study and all my years of education.

Declaration

I hereby declare that, except where otherwise indicated, this document is entirely my own work and has not been submitted in whole or in part to any other university.

Signed: .... Date: ....

Abstract

In the average household, an individual will be alerted to a potential fire danger by an audible alarm emitted by a ceiling mounted detector which detects either smoke particles or heat. If an individual has a hearing impairment or is entirely deaf they will have trouble detecting an audible alarm so must rely on other senses to be alerted to a potential danger.

Currently, visual alerts such as a flashing strobe are available to interface with a smoke alarm that will emit a bright flash of light intermittently to draw the users attention to a detected danger. This solution provides an issue of effectiveness. If a user is asleep or in a non-equipped room the chances of being alerted are greatly diminished and the danger to their well being increases.

This system provides a solution to this challenge through the implementation of a two stage tactile alert. The first stage is a phone call to the users phone providing them with a vibrating alert from the pre-programmed phone number in the system alerting them to a danger. The second is a pillow vibrator which also gives a vibrating alert when the smoke detector is activated. This increases the chances of the user waking from their sleep to presence of a danger.

Table of Contents

Table of Figures

Chapter 1 – Introduction

The definition of deafness or being deaf is; “lacking or deficient in the sense of hearing” [1]. Approximately 17% of the adult population of Ireland suffer from some form of hearing loss at any given time. This loss ranges from the very mild loss to profound or even complete loss of hearing [2]. According to research, approximately 55% of people over the age of 60 in the UK are deaf or hard of hearing [3].

Description of Hearing Loss	% Total Adult Population	Number of Adults
Mild	11.33%	399,201
Moderate	4.99%	175,818
Severe	0.54%	19,026
Profound	0.14%	4,933
Total	17%	598,978

Figure 1: Statistics on Adult Deafness in Ireland [2].

Level of Hearing Impairment	Decibels (dB)
Mild	25-40
Moderate	41-70
Severe	71-95
Profound	96+

Figure 2: Hearing Impairment Classifications [2].

It can be seen from the above figures that some form of hearing impairment affects more than one in every six adults. This is a very significant statistic and one that has many implications for both the individual affected and the people in their lives.

One of the aforementioned implications of hearing impairment is safety. Irish Standard IS409:1988 and British Standard BS5446 state that for a dwelling based smoke-detector, the minimum warning output should be 75dB at 3 metres from the smoke-detector [4]. As a result of this regulation most battery-powered smoke-detectors, intended for use in a dwelling, operate in the 75-85dB output range. The 'Inverse Square Law' states that for every time a person doubles their distance from a sound source, there will be a 6dB decrease in the level of noise [6].

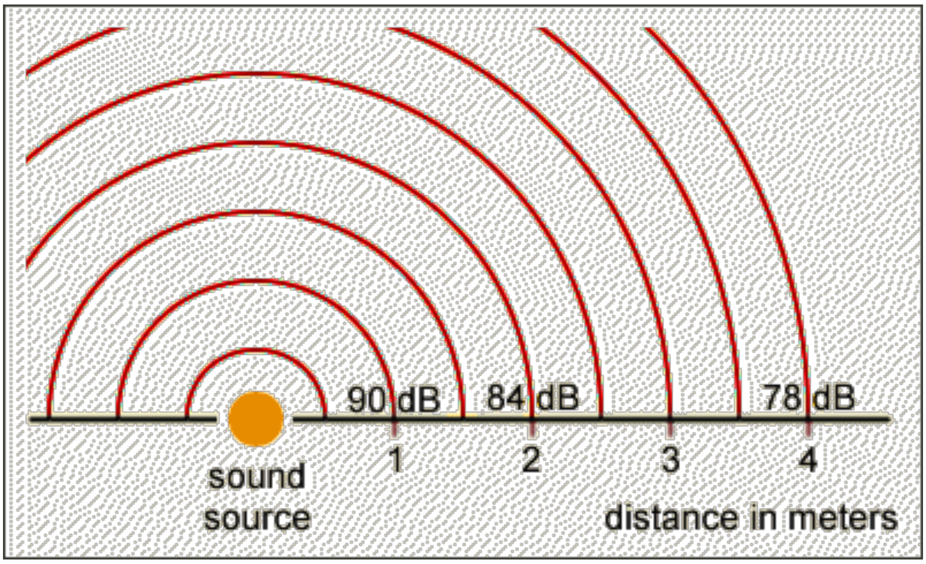

Figure 3: Diagram demonstrating the inverse-square law [6].

This means that, for most household smoke-detectors, if an individual is 6 metres away from the device they hear the emitted warning sound, depending on individual devices, at a level of between 69-79dB, not accounting for any noise absorption between the individual and the device such as curtains, doors and walls.

Considering this range and Figures 1 and 2 above, it can be deduced that individuals who are in the upper range of the Moderate hearing impairment category, as well as individuals who are members of the Profound and Severe hearing impairment categories, are not suitable users of these standardised smoke-detectors and must therefore implement smoke/fire detection devices that are more suited to their health requirements. From the figures provided it can be estimated that standard smoke-detectors are therefore not suitable for approximately 100,000 Irish adults, with this figure set to rise due to Ireland's increasing population[7] and the rise in levels of hearing loss (up 30% from period 1988-1994 to 2005-2006) among younger people, often attributed and referred to as the 'MP3 generation' [8].

Year	Total Cost of Fire Service € millions	Total Number of Fires Attended	Total Number of Fires in Domestic Buildings	Fatalities resulting from Fire
1998	103.62	6,230	11,361	45
1999	119.36	30,210	11,540	51
2000	135.48	31,869	10,194	4
2001	153.79	32,820	10,982	58
2002	167.73	28,099	9,467	48
2003	176.88	31,837	9,438	34
2004	198.53	30,778	8,816	35
2005	209.61	30,537	9,165	41
2006	227.85	33,460	9,394	42
2007	247.74	34,669	9,495	41

Figure 4: Irish Fire Service Statistics 1998-2007 [5]

From Figure 4 above it can be seen that in 2007, the most recent year from which statistics are available, that there were 9,495 fires in domestic buildings. Taking the figure of 17% of the Irish adult population possessing some form of hearing impairment, it can be calculated that, statistically, there are 1,600 fires in the homes of these affected adults in Ireland annually.

41 people lost their lives in fires in 36 incidents in 2007, of these 36 incidents, 35 of them occurred in domestic premises'. There were smoke-detectors found in only 14 of the 35 domestic premises' of these 14 smoke-detectors only 5 were found to be working at the time of the fire [5]. There is an obvious and widely accepted inverse relationship between working smoke detectors in a property and injuries and deaths caused by fire.

Simply put; smoke detectors save lives.

1.1 Project Aim

Project Brief: *People who are deaf or hard of hearing cannot hear a smoke alarm and may fail to see a visual alarm (e.g., a flashing light) if they are not in the room or if they are asleep. Design a hardware/software solution that detects the smoke alarm (either by being directly wired into it or by detecting its emitted signal) and then transmits this to a handheld device that can vibrate and/or show a visual warning. A mobile phone would be ideal because people usually carry them around all the time. A further extension would be to interface with a self-designed or third party bed or pillow vibrator.*

Taking the project brief and the facts regarding hearing loss and fire safety, stated in the introduction, it can be deduced that individuals with hearing impairments cannot rely on auditory-only fire warning devices and that alternative systems of warning must be chosen in the homes of these people. With this in mind, the aim of this project is to develop and deliver a smoke or fire detection system that that takes advantage of other senses to appropriately and effectively alert an individual who is hard, or totally devoid, of hearing to a danger in the location in which the device is installed. This report will, therefore, investigate, discuss and review the design and operation options available and eventually chosen during the development phase.

1.2 Background

At present, there are three widely accepted methods of waking an individual from sleep; auditory, visual and tactile. An auditory alert usually comes in the form of the standardised T-3 smoke detector. T-3 is a temporal pattern that consists of a repetition of three beeps then a pause, which continues in a loop. The T-3 household smoke detectors generally operate at a pure tone of 3100Hz. In a study conducted in 1998 it was found that 100% of adults with healthy hearing woke to the standard smoke-detector signal of 3100Hz with an intensity of 60dB within 32 seconds of alarm initiation. This can therefore be considered an effective system [9]. However, one of the leading indicators of hearing loss is the loss of ability to recognise high frequency sounds. This poses the question; are these devices suitable for the hard-of-hearing?

Visual alerts come in the form of wall or ceiling mounted strobe lights, usually Xenon. Strobes are designed to a US standard minimum of 110 candelas and usually operate in the range of 140-180 candelas. While a visual alert is an extremely effective method of drawing a woken individual's attention to potential danger, there is an obvious question as to the capability of this approach towards waking a sleeping individual.

Figure 5: Example of smoke-detector and strobe pair.

Tactile fire and smoke-detection warning devices come in the form of bed and pillow vibrators or shakers. These devices are activated by an input signal, which can come in the form of an audio signal from a standard smoke-detector audio output, or a wired digital signal, also from a smoke detector. The vibration is generated by an electric motor with an unbalanced mass on its driveshaft, which in turn causes an unstable response during operation thus developing movement in the motor which is transferred through the device in which it is mounted to. This type of alert is perceived as particularly useful for alerting sleeping individuals while not being effective at alerting a woken individual, unless they are in direct contact with the area in which the device is mounted.

Figure 6: Example of a pillow-vibrator.

In 2007, a comprehensive research study into the area of fire-safety for individuals who are deaf or hard-of-hearing was published by Dorothy Bruck and Ian Thomas of Victoria University, Australia on behalf of The United States Fire Protection Research Foundation [10]. The study's aim was to optimize the performance requirements for alarm and signalling systems to meet the needs of high risk groups, in this case, the heard-of-hearing and to test the waking effectiveness of several different auditory signals, a bed shaker, a pillow shaker and a strobe light in a sample of hard of hearing people. Ranges of different intensity levels were tested for each signal. Each device was tested separately.

Participants were 38 volunteers aged 18-77 years (16 males, 22 females) with an average hearing loss of 25-70 dB in both ears (i.e. mild to moderately severe hearing loss). No deaf individuals participated. Each participant was exposed to a range of signals across two non-consecutive nights during slow wave sleep (the deeper part of sleep, stages 3 and 4). Six signals were tested:

- 400 Hz square wave signal in T-3 pulse
- 520 Hz square wave signal in T-3 pulse
- 3100 Hz pure tone in T-3 pulse (the current smoke alarm)
- Bed shaker (under mattress) in T-3 pulse
- Pillow shaker in T-3 pulse
- Strobe light in T-3 pulse (modified)

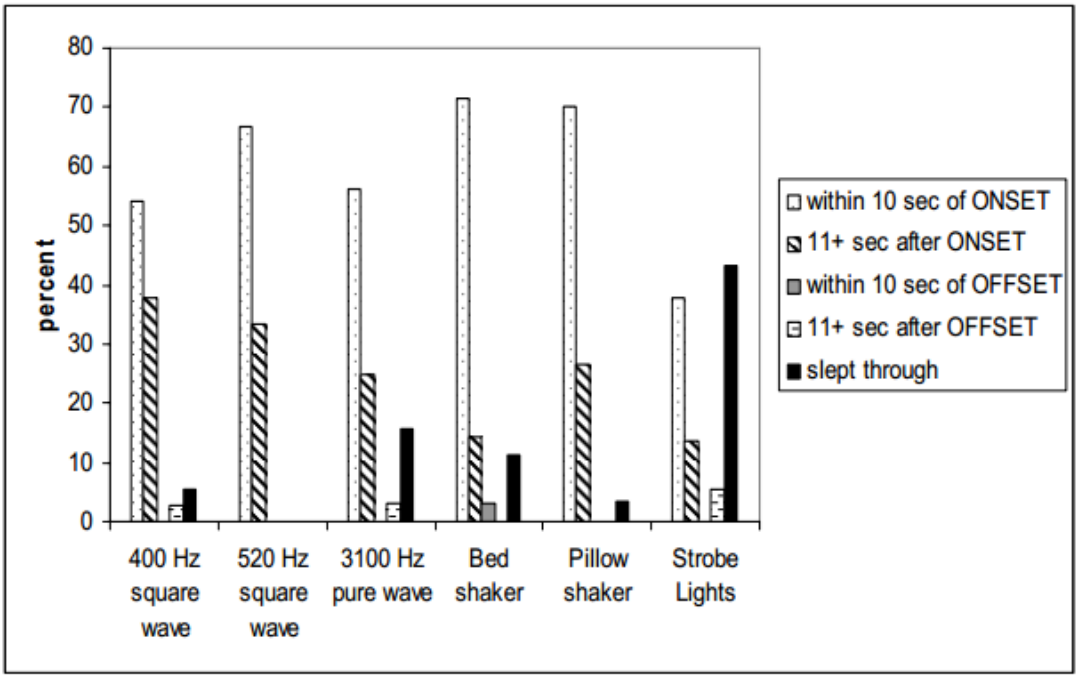

Figure 7: Percentage of awakenings as a function of timing of the onset and offset of each signal [10].

The main conclusions drawn from this study were found to be:

- The 520 Hz square wave T-3 sound is the single most effective signal, awakening 92% of hard of hearing participants when presented at or below 75 dBA for 30 seconds and awakening 100% at 95 dBA. Both the 520 Hz square wave and the 400 Hz square wave were significantly more effective than the 3100 Hz pure tone T-3 sound, which awoke 56% at or below 75 dBA. In addition the 520 Hz square wave signal yielded the lowest hearing threshold when awake for this sample of people who were hard of hearing, from a set of eight alternative sounds with a range of pitch and patterns.

- The bed shaker and pillow shaker devices, presented alone, wake 80-83% of the hard of hearing participants at the intensity level as purchased (vibrating in intermittent pulses).

- Strobe lights are not an effective means of waking this population, with only 27% waking to the lowest strobe light intensity, which was more intense than that required by the standard.

- It is found that, when a signal was presented at a level that caused awakening, most people awoke to the signal within the first 10 seconds of the signal being on. Thus it seems highly probable that a T-3 signal that is alternatively ON for about 10-15 seconds and OFF for a certain period of time (possibly of the same duration) will be more effective than a continuous sounding T-3 signal.

- Worryingly, the questionnaire responses indicate a high level of misplaced complacency among people who are hard of hearing in terms of their need for specialist alerting devices. In view of this, and the fact that many people are not aware of their hearing loss, it is desirable that any standard audible smoke alarm for the general population emit a signal that maximises the chances of awakening for hard of hearing people (provided such a signal presents no increased risk to other sections of the population).

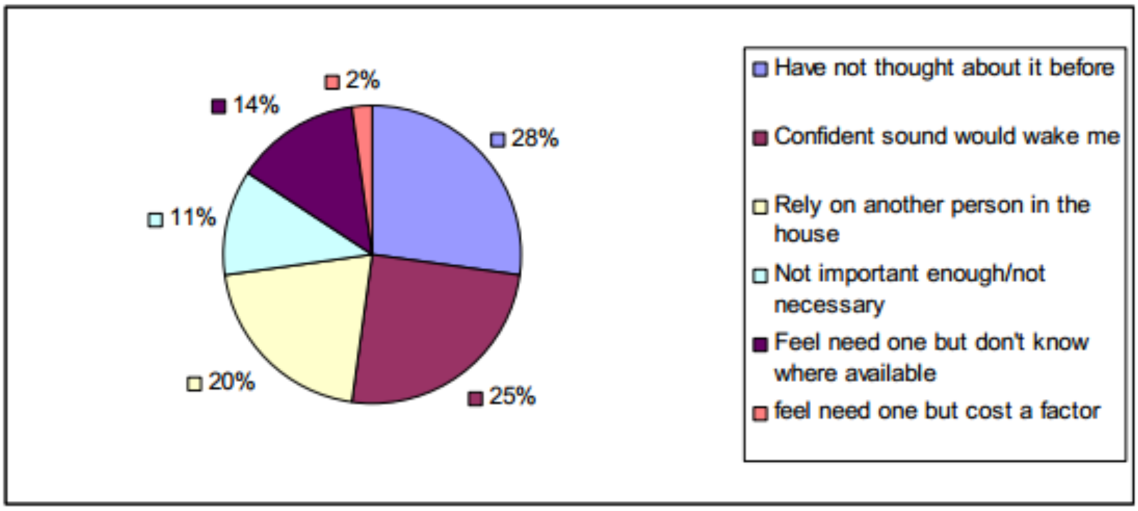

Figure 8: Pie chart of reasons reported by hard of hearing participants who did not have an alternative alarm [10].

The following recommendations can be drawn from this study:

- The current smoke-detector standard of a 3100Hz T-3 signal should be removed and replaced with a 50Hz T-3 signal. This would mean that standard off the shelf smoke detectors are more encompassing in terms of their suitability to the general population and negate the risk of lack of awareness and apathy, indicated in Figure 7 above, which can cause individuals who are hard of hearing not to seek out specialist equipment suitable for their condition. Ideally this sound will be as loud as possible.

- Strobe lights should not be used as a solitary warning device to awaken sleeping users to the danger of a fire as they have an extremely poor effectiveness rating of 27%.
- It is likely that combinations of the three devices, in pairs or all three together, provide the highest likelihood that an individual will be awoken or alerted to a possible danger.
- There should be further investigation of an appropriate means of standardising the measurement of the intensity of bed and pillow shakers and this should inform a new standard [10].

From researching the technology currently approved and recommended for persons possessing a hearing impairment alongside studies such as the one reviewed above, it can be concluded that the optimum combination for waking a sleeping person with a hearing impairment is a T-3 520Hz square wave auditory signal combined with a pillow vibrator.

These studies fail to take into account the case for totally deaf individuals. While a visual indicator such as a strobe is an effective way of alerting a woken individual with severe hearing loss or total deafness to a threat, it is not 100% effective. There are risks to be considered; is it feasible to have a strobe or beacon in every room of the house? How does a person who is, for example, cleaning their bathroom become aware of a fire threat?

The author of this report proposes the investigation and development of a method that can interface with current fire and smoke detection technology to provide a mobile alert and thus increase the chances of a successful warning to the end user at all times and in all scenarios.

1.3 Task Breakdown

In order to investigate and develop a mobile alert system, the following steps were identified to break down the project into sub-tasks.

1. Research commonly sold smoke-detection/warning systems for the deaf and hard of hearing to decide on an initial detection method.
2. Select at least two methods of providing a mobile alert and examine the benefits and limitations of each method in order to choose the best.
3. Identify an appropriate method of communicating the danger from the initial detection device (smoke-detector) to the mobile device. A method of wireless communication would be desirable.
4. Choose a suitable microcontroller to perform processing of detection signal, transmit danger warning and control mobile alert method. At least two microcontrollers should be examined for their capabilities and limitations to choose the correct one.
5. Develop components, circuitry and code so microcontroller receives a signal from detection device, process signal and transmits an alert to a mobile warning device.
6. Package the finished product, with a focus ease of use and aesthetic appearance.

1.4 Project Schedule

Semester 1:

Week 4: Project allocation. Meet with supervisor. Begin preliminary research.

Week 5-6: Complete all required research in order to perform steps 2 - 4.

Week 7-10: Perform steps 2 – 4 and await delivery of components. Perform any more research that is required while awaiting components.

Week 10-12: Begin task 5, technical development of project. Present first semester progress to supervisor in week 12.

Semester 2:

Week 1–5: Continue to perform task 5 and develop the system.

Week 6-7: Testing and Troubleshooting.

Week 8+: Allow a window of time for any delays that occur throughout development. Produce project report during this time.

Week 10-11: Project deadline week 10, report deadline week 11.

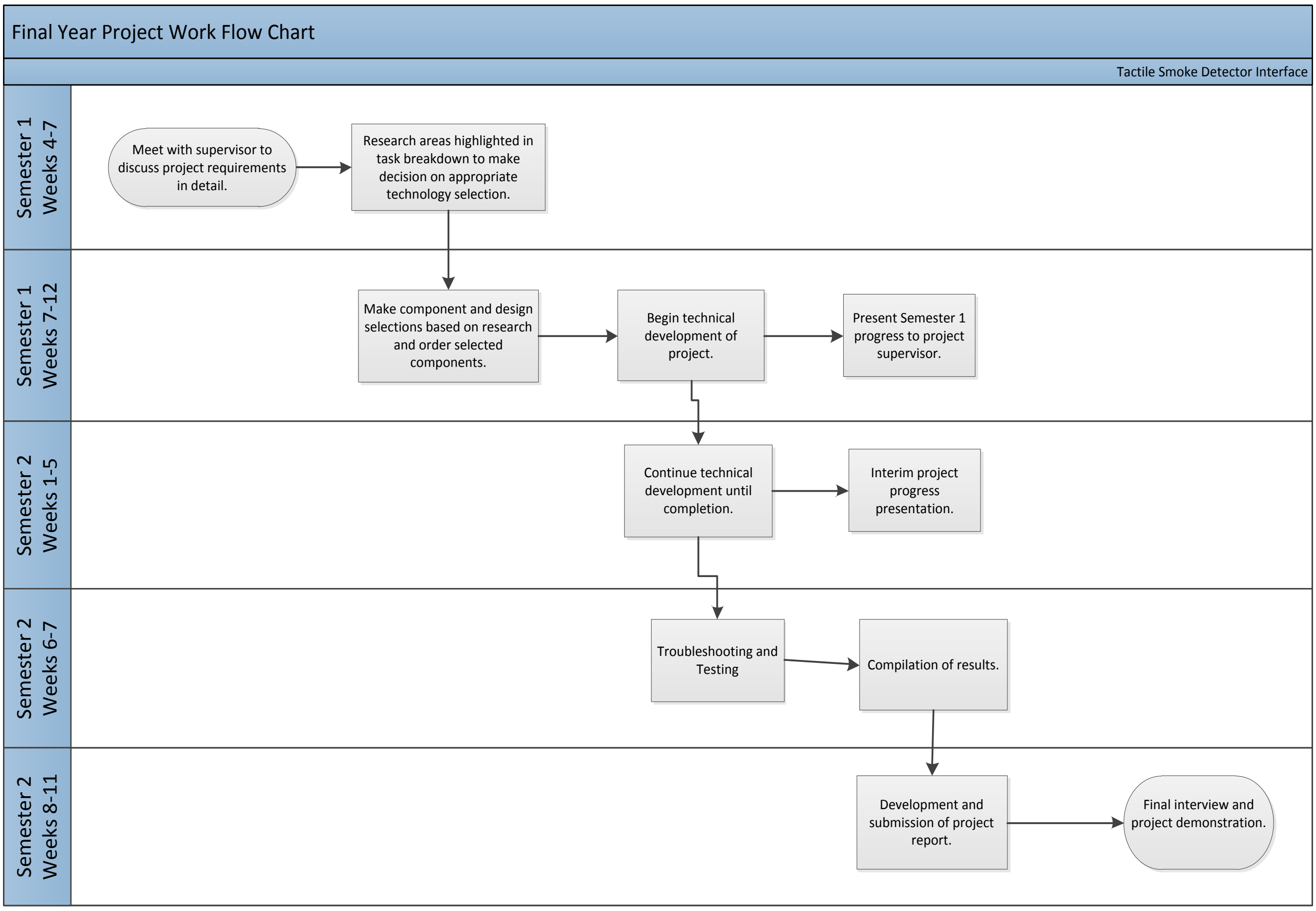

Figure 9: Indicative Project Schedule Flow Chart.

1.5 Document Structure

Chapter 1: This chapter provides an introduction to the report. The chapter discusses the importance of smoke-detectors in the home and the correlation between fire deaths and unsuitable, disused or non-existing smoke-detectors in a home. It also discusses the classifications of hearing loss and how, depending on level of severity, this can affect the safety of the individual in the home when it comes to a danger such as a fire. It also outlines the breakdown of tasks to be performed and the proposed development schedule.

Chapter 2: This chapter discusses technical considerations encountered during the development of the project and the selection choices made for prototyping and development.

Chapter 3: This chapter discusses the design and development of the project in terms of hardware, software and interfacing. The technical challenges encountered and the techniques used to solve each one.

Chapter 4: In chapter 4 the testing methodology developed is explained and discussed and the results of the testing are examined and a conclusion is drawn. Any other technical encountered are also discussed.

Chapter 5: This chapter investigates and ethical considerations, which must be taken on board during the development and production of this, designed solution.

Chapter 6: This chapter concludes the report with final observations on the challenge the project posed and the overall outcome and any further steps that could be taken.

Chapter 2 – Technical Background

This chapter will discuss the selection issues involved in the development of a solution. In particular, it will look at and discuss the options available when it comes to the selection of components to achieve each task set out in the introduction. Finally, it will provide a justification for the final selection of the components and methodology that is used in the development of a finished solution.

2.1 Fire-Detection

In order to develop an effective mobile fire alert system, said system must first be able to detect a potential danger in order to communicate that danger to a user. Extensive research was conducted into the most common and effective methods of fire and smoke detection. From this research, the three most common methods of detection were chosen for review. These methods are; heat detection, ionization smoke detection and photoelectric smoke detection.

2.1.1 Heat Detection

A heat detector is a technically simple but effective device that can work in either one of two ways. Fixed temperature heat detectors operate when the heat sensitive eutectic alloy reaches the eutectic point changing state from a solid to a liquid. Thermal lag delays the accumulation of heat at the sensitive element so that a fixed-temperature device will reach its operating temperature sometime after the surrounding air temperature exceeds that temperature. Fixed temperature detectors activate at a temperature of approximately 47°C.

Rate-of-rise (ROR) heat detectors operate on a rapid rise in element temperature of 67° to 83°C increase per minute, independent of the starting temperature. This detector can operate at a lower temperature condition than would be possible if the threshold were fixed. It has two heat-sensitive thermocouples. One thermocouple monitors heat transferred by convection or radiation. The other responds to ambient temperature. Detector responds when first’s temperature increases relative to the other.

2.1.2 Ionization Smoke Detector

Ionization smoke detectors use an ionization chamber and a source of ionizing radiation to detect smoke, it uses a minute amount of radiation to ionize the air molecules in the chamber and creates a charge between a positive and negative plate to create a small circuit. When smoke enters the chamber, the ions attach to the smoke particles, interrupt the ionization and thus break the circuit or create a current drop causing the alarm to be triggered.

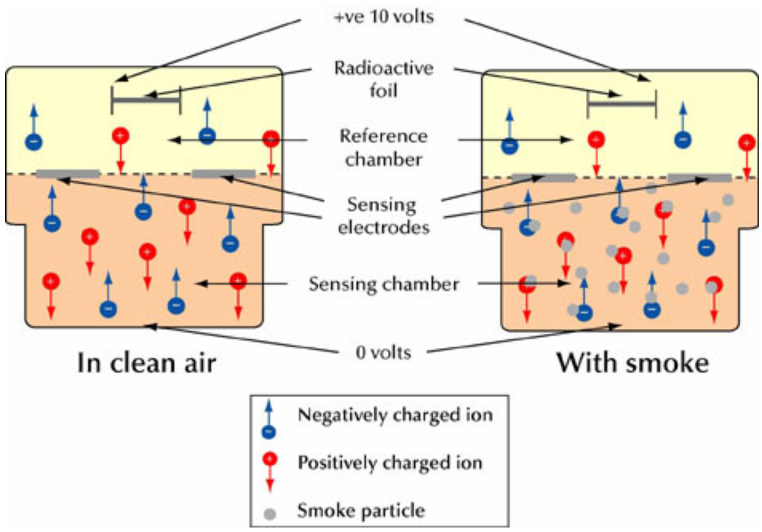

Figure 10: Diagram explaining operation of ionization smoke-detector.

2.1.3 Photoelectric Smoke Detector

A photoelectric smoke detector uses a LED and a photocell, both enclosed in a chamber. The LED emits light across the chamber and the photodiode (or other photoelectric sensor) causes a current when exposed to the emitted light within the chamber. In the case of a fire, or smoke at least, the smoke enters the chamber and causes the light emitted from the LED to be reflected off the smoke particles resulting in a drop in the amount of light picked up by the photodiode thus causing a drop in current which triggers the alarm.

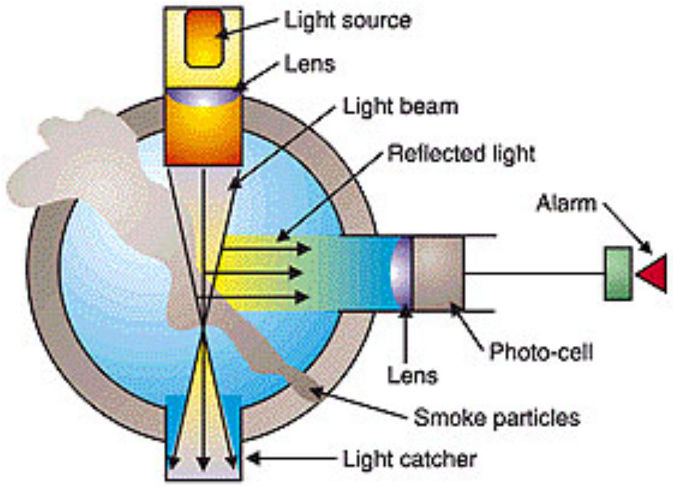

Figure 11: Illustration explaining operation of photoelectric smoke-detector.

Which one?

According to a study conducted in 2003, photoelectric smoke alarm detection times can be up to twice as quick, in most instances, than an ionization smoke alarm [11] while a second study conducted in 2009 found that while the difference in detection times may not be as great as previously found, there is still a faster detection time displayed by a photoelectric smoke detector in most instances apart from kitchen fires [12]. These results are caused by the differing operating principles of both devices. As explained previously, a photoelectric device is triggered by smoke particles meaning that a fire that begins as an initial smouldering that produces smoke will activate these devices much quicker than a ionization alarm as the rate-of-rise of temperature is not sufficiently high to activate them.

In kitchen fires it was concluded that ionization smoke detectors have a faster activation response time than photoelectric, again due to the operating differences between both devices. Kitchen fires often develop rapidly, meaning a fast increase in the temperature of the surrounding air often breaking the ROR threshold that activates the ionization devices before any smoke has had time to develop.

Alarm Type	Average Alarm Time (s)	Standard Deviation (SD) (s)
Ionization Alarm	1205	1102
Photoelectric Alarm	666	537

Figure 12: Results of detection times between ionization and photoelectric smoke alarms [11].

Alarm Type	Average Time to Alarm (s)	Standard Deviation (s)
Ionization (2.6 %/m)	1929	2104
Ionization (4.3 %/m)	1981	2132
Ionization (5.9 %/m)	2006	2138
Photoelectric	1755	1915

Figure 13: Response times found for different fire detection devices [12].

In conclusion due to the development required, there are safety issues raised by opening an ionizing smoke detector with its internal source of radiation and due to photoelectric smoke detectors faster detection time and being recommended for the majority of locations in a house and the negligible cost difference between the two methods, a photoelectric smoke detector was chosen to be used in the initial project prototyping and development.

2.2 Mobile Alert

In order to provide a mobile alert to a user, a method of delivering the alert must be implemented. Today, the term ‘mobile alert’ is synonymous with cell phone technology. There are countless applications available to consumers that provide mobile alerts for every conceivable topic such as news, sports results and weather reports. These alerts are generally delivered to a user’s mobile phone via one of many wireless communication protocols. Short message service (SMS), 3G/4G GSM technology and Wi-Fi are common examples of such communication protocols.

While, in the past, mobile phones may have been of limited use to the deaf and hearing impaired, today’s mobile ‘smart’ phones offer a multitude of features to assist and enhance the quality of life for hearing impaired users [13]. Phone applications that assist with sign-language and video messaging products, such as Skype, have greatly advanced and increased the use of mobile-technology among deaf adults to the point where usage levels are on a par with the general population.

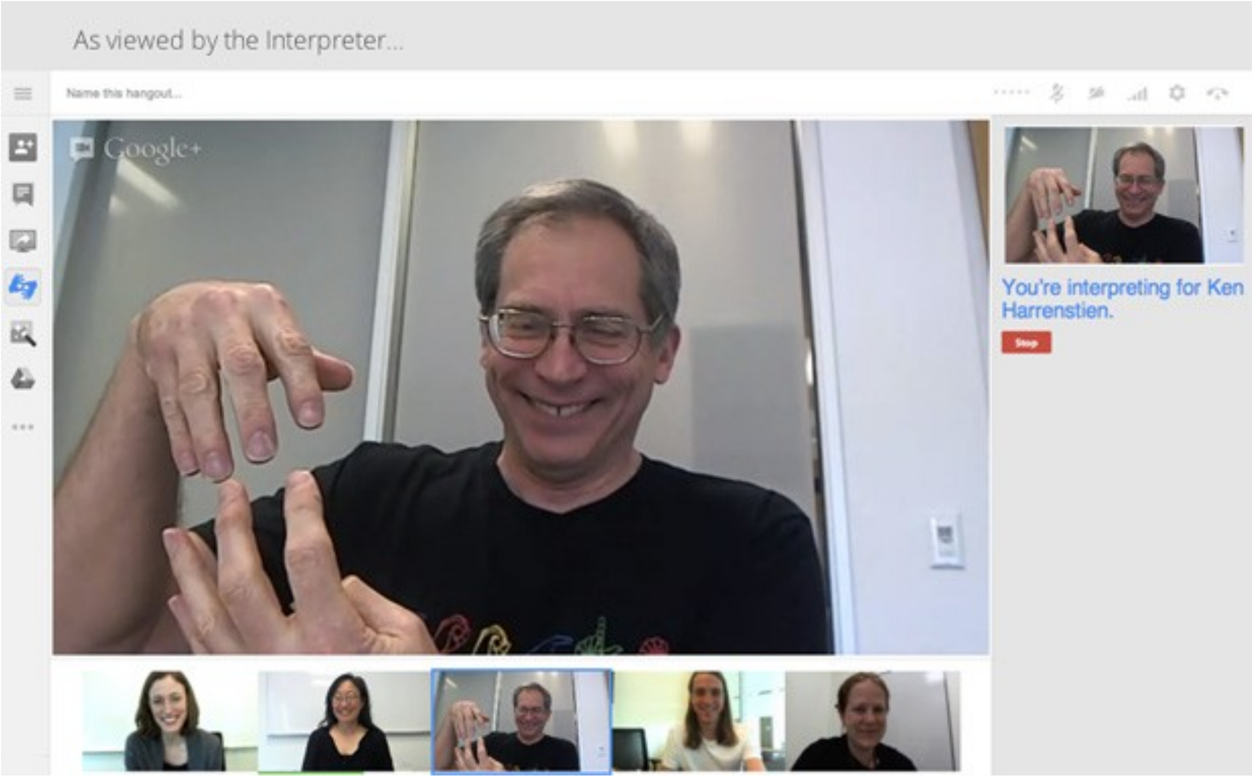

Figure 14: Example of sign-language communication via a video chat application.

The significance of widespread usage of mobile phone technology among the deaf and hard-of-hearing is that the mobile phone provides the perfect platform upon which to deliver a warning alert to a user. A mobile phone is a pre-packaged and professionally produced device which, if can be implemented suitably as a solution in this project, means that the project will have an effective and efficient delivery method for a warning alert as a user of the system will not have to carry any extra devices on their person.

The decided options of delivering an alert to a mobile phone were the following:

2.2.1 Internet Based Alert

An Internet based warning system was considered as a solution. The operating principle behind such a solution would involve an Internet connected microcontroller or server that receives the warning signal from the initial detection device and processes said signal. This processing would potentially be performed by an application or web-application written by the engineer that would then deliver a warning solution such as a voice-over-internetprotocol (VoIP) call or an internet based SMS system such as the API provided by bulksms.com

After consideration, this solution was deemed to be unsuitable for ethical and safety reasons. Firstly, an Internet based solution requires a reliable internet connection in the home. While this is more common than not in developed countries today, it is not an all-encompassing solution and one that would be even less effective if the product were to be developed for users in less economically developed nations.

Secondly, an Internet based solution requires power. A home Internet router needs electricity to operate, an electrical fire in the home may cause a power trip, or worse, that stops the flow of electricity to connected devices, which would in this case cause a loss of internet connectivity and thus render the alert system useless.

It was concluded that a different solution would have to be considered.

2.2.2 Phone Call/SMS Alert

A phone call alert would be an effective method of warning a user to a potential danger as all mobile phones are equipped with a vibration motor that provides a vibrating alert in a multitude of scenarios, in this case when the phone rings. This type of warning system has the added benefit of not only alerting a user when they are in their home but also providing a method of waking a hearing impaired person from their sleep if they choose to locate their phone on their mattress or under their pillow. The third benefit of this type of system is that a user who is not in their home but carrying their phone with them can be alerted to the danger of a fire in their home, early detection meaning an earlier call to the fire service which could potentially limit damage to a home and its contents.

In order to implement an autonomous GSM capable solution into the project, a GSM modem would have to be incorporated. After further research, it was found that GSM modems are controlled by a set of commands called AT commands (these commands will be discussed in further detail later). AT commands can be passed to a GSM modem through a serial connection, an action a microcontroller is capable of performing thus making it a suitable solution for this application.

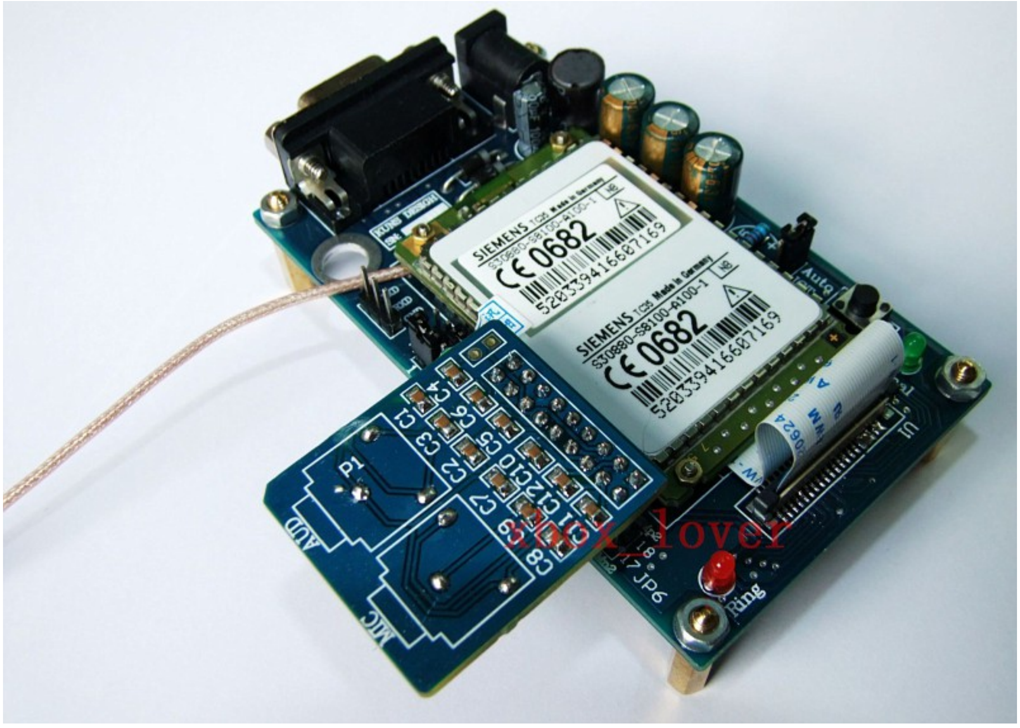

Figure 15: Siemens TC35i GSM development board.

It was concluded that the versatility and functionality of a GSM modem serves as an adequate solution to the problem posed, as a result a Siemens TC35i development board (pictured in figure 14) was purchased for prototyping and development of the project.

2.3 Communication Between Detection Device and Alert Device

To ensure a home smoke detection system effectively warns a user to a danger, it is recommended that multiple smoke detectors are installed throughout a home and in as many locations as possible. As it is not financially practical to integrate a GSM modem in every smoke detector, the devices must be able to provide a warning signal to a centralised mobile alert system, each device must be able to wirelessly communicate its signal with the previously selected GSM modem. To achieve this, it is expected that a form of radio-frequency communication will be implemented. A number of methods were investigated.

Note: Wi-Fi has been previously ruled out for safety reasons

2.3.1 Infrared (IR) Communication

Infrared is a form of wireless communication most commonly used in IrDA, and television remote controls, it uses infrared light-emitting diodes (LEDs) to emit infrared radiation which is focused by a plastic lens into a narrow beam. The beam is modulated to encode the data. The receiver uses a photodiode to convert the infrared radiation to an electric current. The benefits of IR communication are that it is extremely cost effective and easy to implement. The limitations, in terms of this project, are that it has a short range and generally requires devices to be in line-of-sight of each other as IR light cannot pass through most solid object due to its short wavelength [14].

Designing for the requirements of this project, it can be seen that IR communication is not suitable, as each smoke-detection device will have to transmit back to one centralised receiver/server in order to communicate the danger to the user.

2.3.3 Radio Frequency Communication

2.3.3.1 Bluetooth

Bluetooth is a standard for short range, low power, low-cost wireless communication that uses radio technology [15]. Bluetooth operates in the 2400-2480MHz band and implements low cost transceiver microchips with a maximum range of 10 metres for a class 2 device up to a maximum of 100 metres for a class 1 device in optimum conditions, delivering data speeds of up to 721kbit/second.

The advantages of using Bluetooth in an application such as this are clear and it would certainly be a suitable solution. However, the average Bluetooth module has a cost of approximately €35 per unit at present from most component suppliers, reducing when bought in quantity. While worth considering in an application that integrated server connectivity and high rates of data transfer, this is too expensive for an application such as this where a simple on/off signal is required to be transmitted.

2.3.3.2 Transmitter/Receiver Pair

An RF transmitter/receiver pair is a common pairing of electronic components that provide an effective way to transmit data over short range via radio communication. The major benefit of a transmitter/receiver pair is cost. From most component suppliers, an individual transmitter can be purchased for around €3-4 and a pair can be purchased for around €6, with this price reducing when bought in quantity, which a manufacturer would certainly do. In practice, an alert system would incorporate 10-15 smoke detectors throughout a home meaning each smoke detector would have to be equipped with a transmitter and at the designated central location would be a solitary receiver, awaiting the warning signal from the detection devices. The transmission range of a pair is approximately 120-150 metres in open space and 60-90 metres indoors and has a data transfer rate of approximately 1000 bit/second.

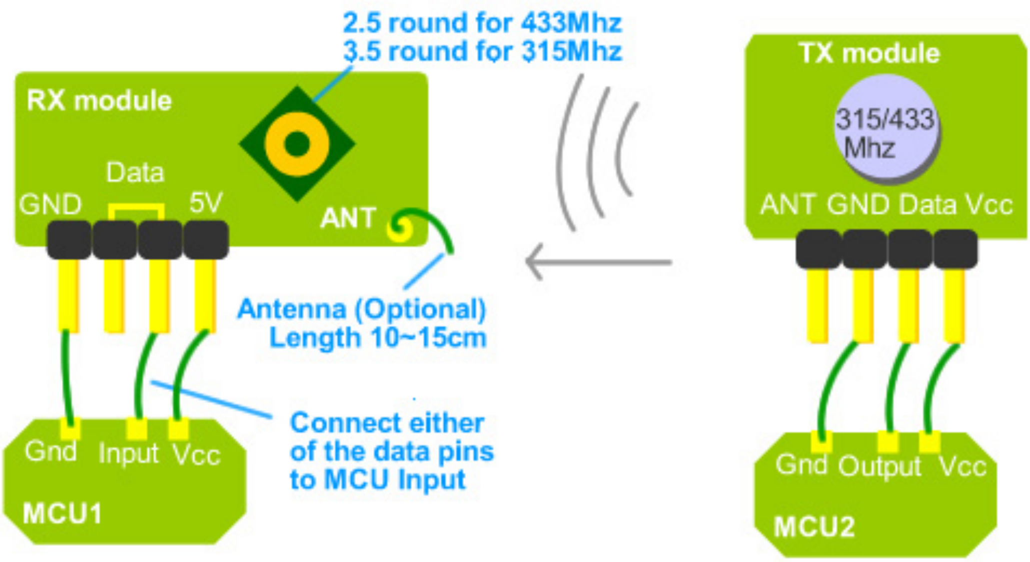

Figure 16: Demonstration of transmitter/receiver pair.

The stated operating capabilities along with the low cost of the devices make the transmitter/receiver pair a suitable solution to the challenge of transmitting a warning signal from a detection device to a central mobile alert server. As a result, a 433MHz pair was ordered for the initial development of a project solution.

2.4 Selection of Controller

As this project involves numerous remote devices communicating with one central receiver device and performing a range of techniques and calculations, a microcontroller is required inside each device to process the information, perform desired actions and control outputs.

There is an extremely large and ever increasing selection of microcontrollers available to the consumer and hobbyist market for prototyping and development, which is advantageous in that the engineer can choose the microcontroller that best, fits the requirements of the project.

2.4.1 PIC Microcontroller

The PIC (Peripheral Interface Controller) microcontroller is a family of Microcontrollers developed by Microchip. PICs are extremely popular in the electronics industry due to their low cost wide range of applications and good performance to power consumption ratio. The most popular PIC, and the one that was considered for use in this project, is the PIC16F87x family of chips. This is a family of 8 bit CMOS Flash microcontrollers [16]. Other advantages of using the PIC is that, due to its compact size, it is easy to embed within a designed circuit. The PIC is also controllable through a set of 35 single word commands meaning it is easy to learn and adapt to for users new to the technology.

Figure 17: PIC16F877 Microcontroller

One aspect of using the PIC in a prototyping environment that could be considered a disadvantage is that you must design your own power supply and control circuitry. This could potentially lead to problems for amateur or novice engineers who design and implement an incorrect circuit which, despite correct programming, could result in unexpected performance, shorting or too much current to components. Another drawback of using a PIC is that it requires a stand-alone programming device that can be quite expensive.

2.4.2 Arduino UNO Microcontroller

The Arduino Uno is a member of the Arduino family of development boards and accessories. Arduino boards are pre-designed development platforms with integrated microcontrollers. Most Arduino boards incorporate Atmel 8-bit AVR microcontrollers, which are similar to PIC controllers in terms of functionality. The Arduino boards provide lots of benefits for users during prototyping of a project, all of the power supply circuitry is integrated as well as providing suitable integrated circuitry to all of the input and output pins meaning an engineer can add external components with minimal effort and less risk of a circuit error causing a performance issue or damage to the components themselves. Another benefit of using the Arduino is that it has a built in boot loader and USB port meaning that it can be programmed, and powered, with a standard USB cable from a computer.

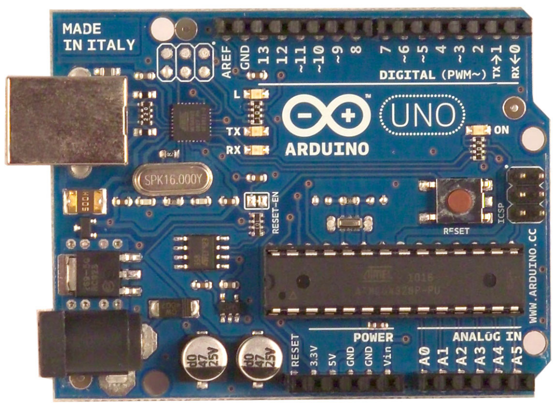

Figure 18: Arduino UNO Development Board.

Arduino boards are programmable via a USB cable using a dedicated integrated development environment (IDE) and in either the C or C++ programming languages [17]. There is also a large on-line community of Arduino developers meaning that there is an ever-increasing library of functions available to developers to perform tasks.

A limitation of using Arduino in a project is that, as it is a development board, it is quite large and not easily integrated, meaning they are often used with a breadboard in order to integrate components and circuits.

After comparing the two options the Arduino Uno board was selected for initial development and prototyping. It was selected for a number of the advantages mentioned in the earlier discussion, such as integrated circuitry meaning the avoidance of circuitry and component related errors in development, multiple easily accessible Input/Output ports and a vast wealth of open-source code available for implementation. It was also chosen as the engineer had a reasonable base knowledge C++ programming. Another advantage of using the Arduino in this instance is that there were a number of Arduinos available immediately for use in the School of Engineering meaning that research and development could begin immediately and replacements were immediately available if there was a hardware failure at any stage. It was decided that if development progressed successfully and a prototype was produced ahead of the project deadline then a more integrated solution involving a PIC could be investigated.

2.5 Conclusion

The task of designing an initial solution to the problems posed by this project and researching and analysing the many different options and components available to solve the those problems showed that, in most technical scenarios, there are multiple ways to develop a solution. Often, two solutions that utilise very different methods can work just as well as each other and the challenge is in the correct overall design.

From the selection stage, there has developed a strong platform of background research and knowledge in the different areas of theory that will be reviewed and investigated during the technical development of a final solution to the project and reinforces the benefit of conducting preliminary research and reviews when undertaking such a task.

Chapter 3 – Design and Development

This chapter will discuss the design decisions and methods of development implemented throughout the project work cycle. Working to the indicative schedule developed in the earlier flow chart, design of a possible solution began in week seven of semester one, once technology selections had been made.

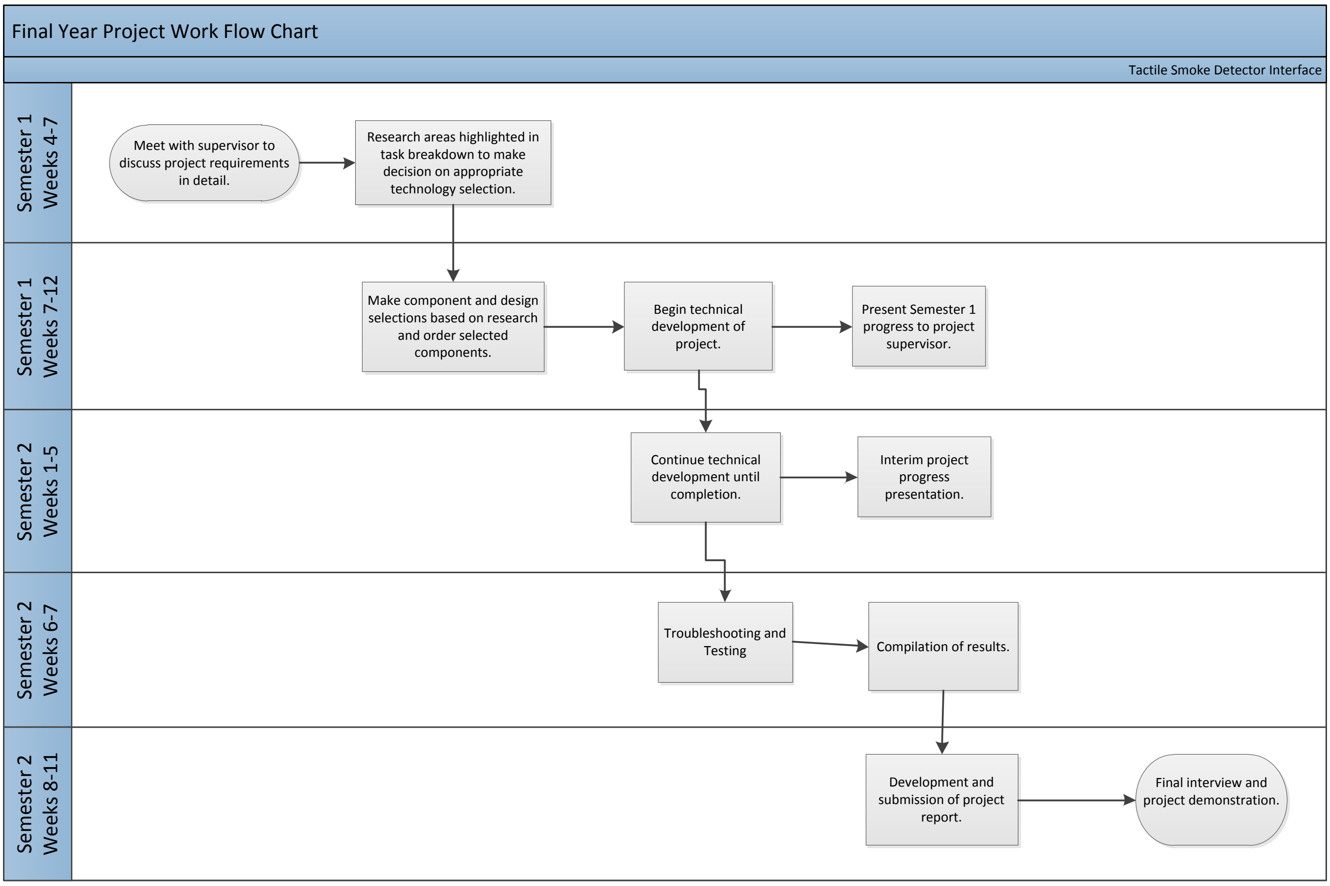

Figure 19: Indicative Project Schedule Flow Chart.

3.1 Working with Arduino

Having already discussed and established the capabilities of the device, the first step in designing and developing a solution required becoming familiar with the Arduino and its development environment. The Arduino Integrated Development Environment (IDE) contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions, a series of menus and a built in compiler. Arduino code is a slightly modified version of C++ and thus incorporates a C++ compiler to compile code. The main feature for a developer to become aware of when writing code for an Arduino is that it is based around two functions; setup and loop. When a program is initialised or run initially, every action inside the setup function is performed once. Here variables can be initiated and actions performed, such as setting a digital output high or low. The loop function is where most of the functionality in a program is developed and inserted. As the name suggests, it loops infinitely allowing methods to be performed or inputs to be analysed and respond accordingly.

A common piece of code recommended for novice programmers to begin understanding Arduino development is a program that commands an LED to blink on and off in one second intervals and is as follows:

```
#define LED_PIN 13

void setup () {
 pinMode (LED_PIN, OUTPUT); // enable pin 13 for digital output
}

void loop () {
 digitalWrite (LED_PIN, HIGH); // turn on the LED
 delay (1000); // wait one second (1000 milliseconds)
 digitalWrite (LED_PIN, LOW); // turn off the LED
 delay (1000); // wait one second
}
```

Once the developer becomes familiar with the Arduino development format, the functions and syntax are the same as C++. For engineers who are inexperienced in C++ development there is a wealth of tutorials and sample exercises available from the Arduino organisation.

3.2 Interfacing TC35 with Microcontroller

In order to communicate a warning signal to a user via a mobile phone, the Arduino must be able to command the GSM module to perform the required actions, this required interfacing of the Siemens TC35i GSM module with the Arduino Uno microcontroller. This initial step posed a number of challenges, the first being that there was scarce availability of technical documentation and information for the Siemens TC35i module and did not arrive with any datasheets or user manuals. After some investigation, most of the literature available online was for the modem itself and not the development board, which was purchased for the project. This meant that there were uncertainties surrounding operating parameters and capabilities of the board itself.

After some further research, a hobby electronics blog was found that provided some introductory information about the Siemens TC35i development board and its operating requirements and capabilities, enough information to get started.

3.2.1 Communicating with GSM Module

The TC35i development board possesses a dedicated serial UART port that can be connected directly to a PC, which can then be used with a desktop program to command the modem. The issue with this approach in terms of this project is that the Arduino, and most microcontrollers, do not possess UART interfaces so this mode of communication is not suitable for this application.

It was then observed that the board had transmit (Tx) and receive (Rx) pins that allow a user or device to input and output information through serial communication. Serial communication is the method of transferring information one bit at a time. This capability meant that a serial communication channel could then be established between the Arduino microcontroller and the GSM module due the Arduinos serial communication ability and its added functionality of a 'Software Serial' library that replicates serial communication through its digital outputs [18].

3.2.2 Powering the GSM Module

The Siemens TC35i modem requires an operating voltage of between 3.3-4.8V and a minimum of 250mA. The development board has either the option of a 9V DC input with an integrated voltage regulator, or four 3.3V input pins that can be connected to an external power source. It is recommended that a power source that can provide up to 500mA as, during certain applications, the modem's consumption can spike.

The Arduino board has a number of options for the supply of power. It can receive power via USB, which is approximately 5V. It also has the option of DC input with a recommended power supply of 7-12V with a range limit of 6-20V. When receiving DC power it implements a 5V voltage regulator to provide power to the Atmel microcontroller on the board. It also has the option of receiving up to 9V through its Vin pin.

One of the many advantages of using the Arduino for prototyping is that it can supply power to a circuit at either 3.3V or 5V via two dedicated pins on the board, which can supply up to approximately 450mA, which would make it a suitable power source for the TC35i module.

There is now in place a power supply to both the Arduino and GSM module and a communication pathway between them. In the circuit diagram below, the red wire is providing power from the 3.3V of the Arduino board and the black wire is the ground connection. The blue wire is the connection between the transmit (Tx) pin of the Arduino to the receive (Rx) pin of the TC35i board. The green is from the Tx pin of the TC35i to the Rx of the Arduino, this allows a two-way communication stream between both devices via the serial communication pathway discussed earlier.

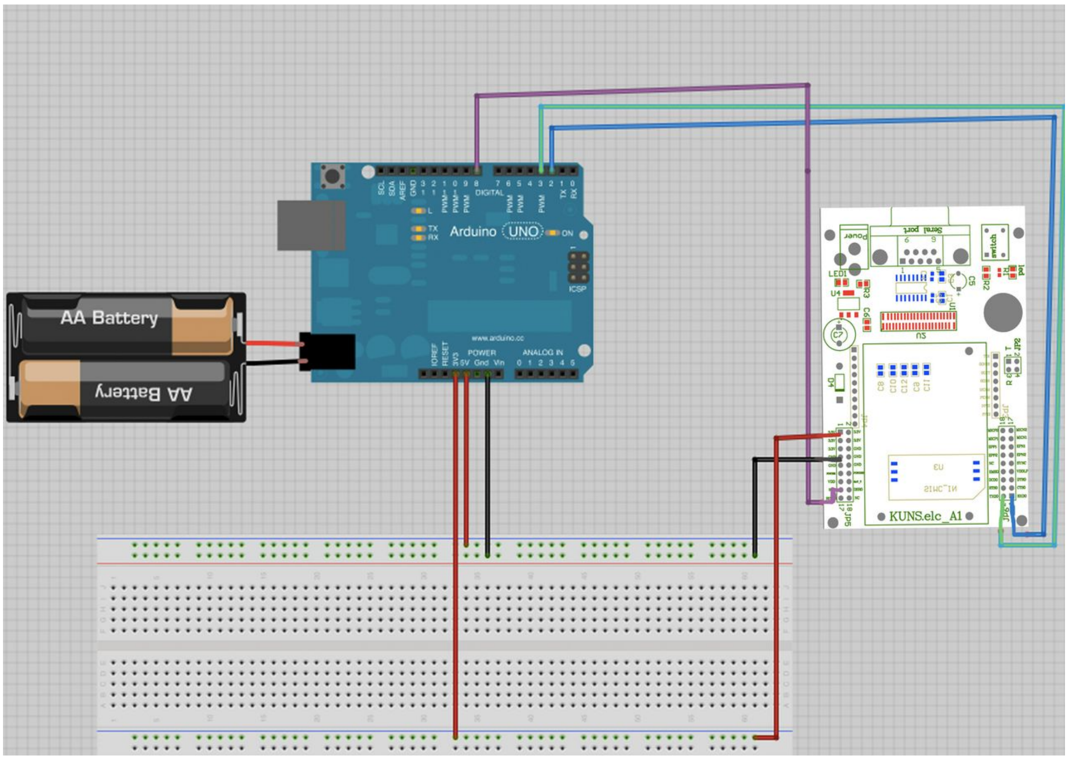

Figure 20: Initial communication circuit between Arduino and TC35i module.

3.2.3 Commanding the GSM Modem

There is a dedicated modem command language available for use in telecommunications in order to command a GSM modem. The Hayes command set, or AT command set, is a language developed originally by Hayes for one of their products in 1981 [19].

All commands begin with the prefix 'AT' which is an abbreviation of attention, this calls the modem to attention. The AT command is then followed by the individual command itself for action. For example, the command 'ATD+353XXXXXXXXX' commands the modem to dial the phone number that is appended to the command after the character 'D'. The command 'ATH' commands the modem to hang up any active dials [20]. There is a vast amount of AT commands and supporting literature as some companies have adapted the language to create a bespoke command set for their own products.

3.2.4 Code

As previously discussed, a software serial connection can be established between the Arduino and GSM modem. This is achieved by including the SoftwareSerial library and declaring which pins will be Tx and Rx.

```
#include <SoftwareSerial.h>
SoftwareSerial gsmSerial(2,3);
```

The begin command establishes a serial communication at the desired baud rate. In this example it is 1200.

```
gsmSerial.begin(1200);
```

Once the serial connection is established, commands can be sent to the GSM modem by printing characters to the serial connection via the 'print' command followed by the characters or command the modem is required to perform. The following AT command sets the modem to message mode, a prerequisite to working with SMS messages.

```
gsmSerial.print("AT+CMGF=1 \r"); // Setting GSM Module to message mode
```

In order to catch and display the characters that the modem returns such as confirmation messages when a command is delivered, the following piece of code is implemented:

```
if(gsmSerial.available() > 0){
    gsm_char_send=gsmSerial.read();    //Store the char in gsm_char.
    Serial.print(gsm_char_send);   //Print it to debug serial
}
```

This functionality assigns any available incoming characters from the modem to the variable 'gsm_char_send' and then prints those characters to the Arduino serial monitor. This is a useful function to implement for troubleshooting and debugging purposes if there are any functionality issues with the modem.

The main modem functions required for this project will be dial, hang up and potentially to send SMS warnings also if required. The dial and hang up commands have been discussed previously, the following piece of code can be implemented to send a SMS to any desired phone number:

```
  gsmSerial.print("AT+CMGF=1\r");
  delay(1000);
 gsmSerial.println("AT+CMGS=\"+353879149715\""); //AT command to send SMS
 delay(500); //the length of this delay is very important. 400 is too low
 gsmSerial.print("Hello World"); //Print the message
 delay(100);
 gsmSerial.println((char)26);
  Serial.println("\r\nSMS Sent\r\n");
```

The command 'AT+CMGF=1' sets the modem to SMS mode. The command 'AT+CMGS= <PhoneNumber>' is used to command the modem to send a SMS to the desired phone number, this is followed by printing the contents of the SMS to the modem via a serial print, which will then be sent out as the SMS body by the modem. After some troubleshooting it was found that a delay is necessary between each command as there is a slight time delay in the modem when performing each action.

Figure 21: Incoming SMS from GSM Modem.

3.3 Interface Smoke Detector with Microcontroller

To deliver a warning alert to a user, a signal of some form must be passed from the recognised smoke detection device to a microcontroller for action. To achieve this, the photoelectric smoke-detector that was purchased was opened and the operating capabilities and circuitry was analysed. Unfortunately, the documentation accompanying the product did not provide any technical, circuit diagram or datasheet information and after conducting some internet searches on the embedded IC present in the circuit it was found to be a Chinese made component with no datasheets publically available online. This meant that the signal outputs had to be deduced through manual analysis.

Figure 22: Lloytron B820 Smoke Detector used in development.

It was observed that the piezo-electric buzzer, which emits the audible output warning signal, received an output signal from pins 8 and 9 of the IC and the flashing LED received signal from pin 11. Through further analysis it was observed that Vdd and Vss were located at pins 5 and 14 respectively. From this the possibility of using either the piezo-electric buzzer or LED outputs as a signal source to a microcontroller was investigated as both sources are in use when a smoke detection event occurs, the alarm emits an audible warning and the LED provides a visual indicator. Some code was produced for the Arduino to process a voltage input signal:

```
x = analogRead(analogPin);     // read the input pin
float voltage = x * (5.0 / 1023.0);

if (voltage>3) {
  Serial.println(voltage);              // debug value
  Serial.println("\r\n");
```

The Arduino has an integrated analog to digital converter (ADC) that will process a voltage input in the range of 0-5.0 volts and represent it digitally in the range 0-1023, indicating that the ADC used in the Arduino is a 10-bit device. In the above code, a variable x is assigned the converted digital input value and a variable 'voltage' is declared which is assigned the value of the original analog voltage value by multiplying the digital value by 5/1023. With this value it is now easy to perform an action based on the input value, in this example the value for voltage is printed to the serial monitor if it is greater than 3V.

A wire was soldered to piezo-electric buzzer output at pin 9 and input to the Arduino analog input pin, a second wire was soldered to the smoke detectors ground and connected to the Arduino ground pin. When the test button was pressed and the smoke detector activated, a 5V signal was sent to the Arduino, which was required. Unfortunately, this voltage draw was too great and lowered the voltage to the buzzer meaning the warning sound was impaired. Obviously this would be unacceptable in operation and extremely unethical so the LED method was then investigated.

Another wire was then soldered to the LED pin on the IC and the same test was performed. This time the LED operated as normal as well as providing an adequate signal to the microcontroller. However, the LED blinks every 50 seconds as a operational and battery life indicator. This led to false signals being sent to the microcontroller resulting in this being an unreliable method to use.

From this it was concluded that a device that contained a known IC would have to be purchased in order to hopefully implement a more suitable solution. While researching popular smoke detector ICs, the MC145010 IC from Freescale Semiconductor was found. It was observed from the datasheet that it had a similar pin layout to the IC in the B820 smoke detector. Upon inspection both IC's seem to be identical and would appear the Chinese produced IC in the B820 is a replica component of the popular MC145010. From here a datasheet was no available that showed the pin layout of the IC [21].

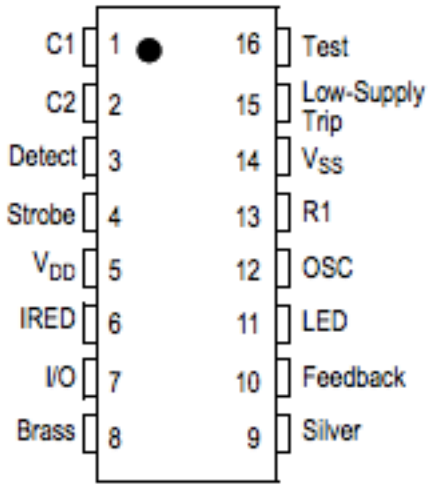

Figure 23: MC145010 IC Pin Layout.

From the technical information provided by the datasheet it was observed that there is an I/0 pin on the IC at pin 7. This pin is provided for the purpose of providing an output voltage to external devices, which is exactly what is required in this application. A wire was added to the pin and connected to the Arduino analog input pin and tested with the code discussed above, this produced a positive result of exactly 5V out and as a result provided a platform for further development of the alert system.

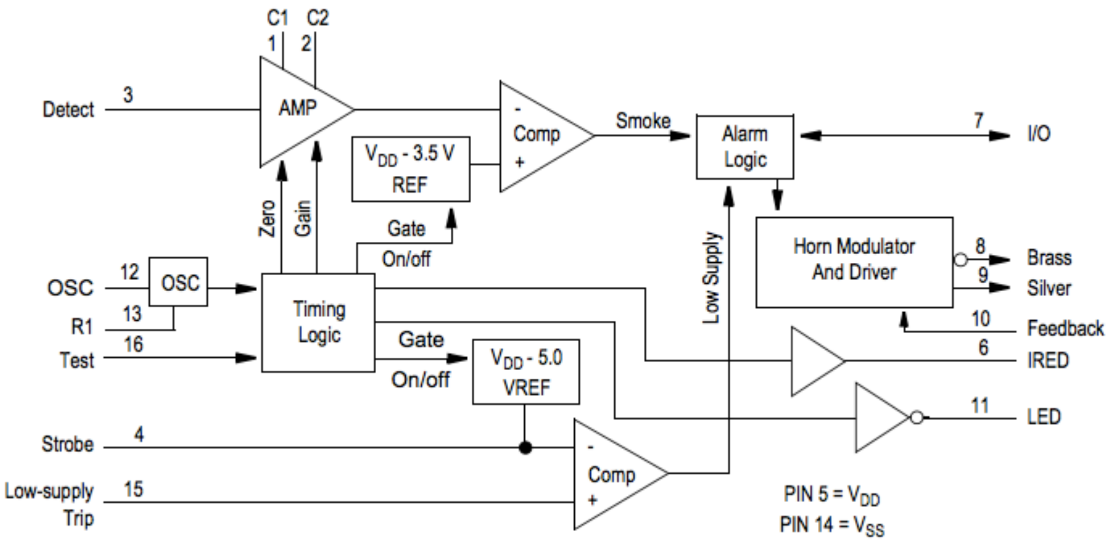

Figure 24: MC145010 IC Block Diagram.

3.4 Wireless Communication Between Smoke Detectors and GSM Module

In the average household scenario there would be numerous smoke detector installed at varying locations throughout the house. In order to mass-produce this type of warning system that provides a mobile phone alert, it would be not be practical or cost effective to install GSM technology in every smoke detector. This means that a cheaper alternative of a radio transmitter will be interfaced with each smoke detector and a radio receiver will be interfaced with a microcontroller and GSM module at a centralised location which will receive any warning signals emitted by the smoke detectors.

As discussed earlier, a 433MHz transmitter/receiver pair was selected for implementation. This pair requires a 5V input, which can be provided by the Arduino board at both the transmitter and receiver ends of the system. The transmitter takes a data input and modulates it within a 433MHz carrier signal. It is then transmitted via a small, integrated antenna, the range of which can be improved by adding a larger antenna. This signal is received by the receiver, where the carrier signal is demodulated and the data is output from the receiver's data out pin. This pin can be connected to the Rx pin of the Arduino to feed the data in.

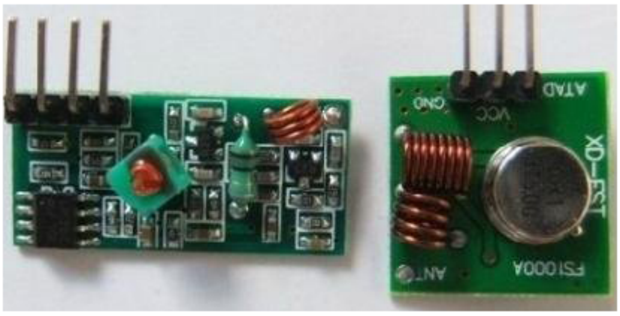

Figure 25: 433MHz Transmitter/Receiver Pair.

In order to transmit data via the RF transmitter, a 'serial.write'command is used followed by the data that is required to be transmitted but since RF receivers are prone to detecting a lot of useless in data in the form of noise, extra bytes were added to the data packet. Two bytes are added to the start of each packet then the byte address is transmitted. This allows great accuracy in terms of transmitting and successfully receiving a data value meaning that in operation there would be no false alarms. A checksum is then sent to confirm the correct data values were received. The following method, ‘writeUInt’ is used to transmit data:

```
const byte g_network_sig[NETWORK_SIG_SIZE] = {0x8F, 0xAA, NET_ADDR};  // Few bytes used to initiate

// Sends an unsigned int over the RF network
void writeUInt(unsigned int val)
{
  byte checksum = (val/256) ^ (val&0xFF);
  Serial.write(0xF0);  // This gets reciever in sync with transmitter
  Serial.write(g_network_sig, NETWORK_SIG_SIZE);
  Serial.write((byte*)&val, VAL_SIZE);
  Serial.write(checksum); //CHECKSUM_SIZE
}
```

The method 'readUInt' is then used to receive and decode the incoming data:

```
// Receives an unsigned int over the RF network
unsigned int readUInt(bool wait)
{
  int pos = 0;          // Position in the network signature
  unsigned int val;     // Value of the unsigned int
  byte c = 0;           // Current byte

  if((Serial.available() < PACKET_SIZE) && (wait == false))
  {
    return 0xFFFF;
  }

  while(pos < NETWORK_SIG_SIZE)
  {
    while(Serial.available() == 0); // Wait until something is avalible
    c = Serial.read();

    if (c == g_network_sig[pos])
    {
      if (pos == NETWORK_SIG_SIZE-1)
      {
        byte checksum;

        while(Serial.available() < VAL_SIZE + CHECKSUM_SIZE); // Wait until something is avalible
        val      =  Serial.read();
        val      += ((unsigned int)Serial.read())*256;
        checksum =  Serial.read();

        if (checksum != ((val/256) ^ (val&0xFF)))
        {
          // Checksum failed
          pos = -1;
        }
      }
      ++pos;
    }
    else if (c == g_network_sig[0])
    {
      pos = 1;
    }
    else
    {
      pos = 0;
      if (!wait)
      {
        return 0xFFFF;
      }
    }
  }
  return val;
}
```

While implementing so many steps may be considered overkill for such a simple application, when dealing with the issue of people's safety it is vitally important to achieve reliability and making sure there are no false alarms is of the upmost importance.

3.5 Vibrating Motor Circuit

Upon researching vibration motors and systems where vibration motors are integrated for the purpose of providing a tactile alert it was observed that the such devices operate by having a vibration motor securely fixed to the device, when the vibration motor receives power it will begin to rotate and the unstable weight offset of the load will cause a vibration in the motor. This vibration increases in line with the rotation of the shaft, with the vibration transferring through the device due to it being securely fastened.

A 5V DC motor was chosen for use in prototyping and to provide the instability, an off centred plastic cylinder was mounted to the shaft.

Figure 26: Example of a vibration motor with unbalanced load.

In order to provide power to a 5V motor from, an external power supply would have to be implemented as the draw on the Arduino's supply would be too great if the motor was drawing from it as well as the RF receiver and TC35i module. In order to provide an external power supply but control the operation from the microcontroller, the following transistor circuit was designed:

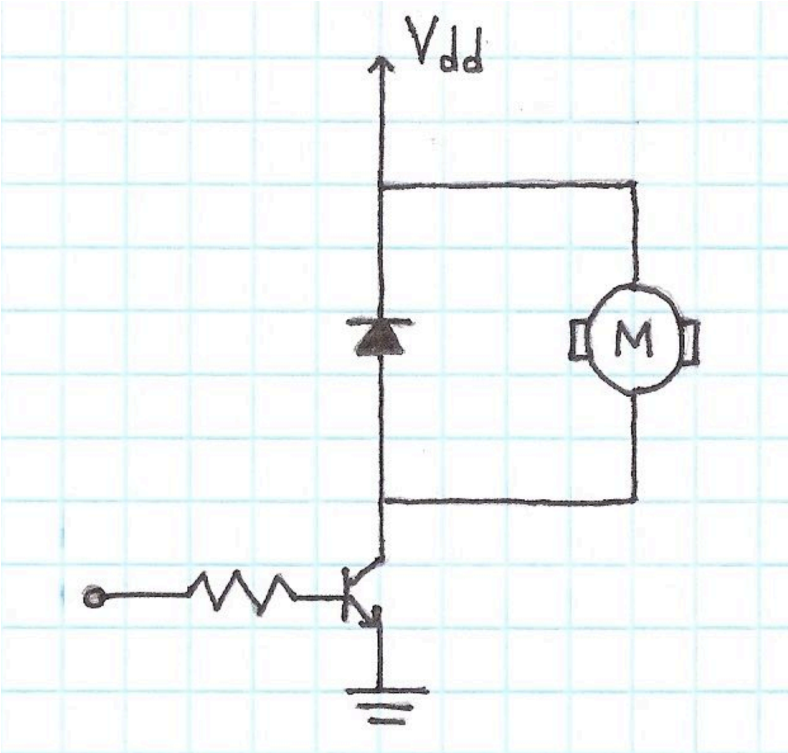

Figure 27: Motor control circuit.

In the circuit above, the motor receives power from an external battery source, Vdd. There is a diode located across the motor to guard the microcontroller from any ‘Back-EMF’. Back-EMF is generated as inductive components like motor winding resist sudden changes in current. That's because the magnetic field caused by the current needs time to build up or decrease. That means that when current is flowing and this is suddenly cut off, the winding will try to maintain that current, and becomes a power source generating a voltage to be able to do so. It gets its power from the built up magnetic field. Since the winding is now a power source instead of a consumer the voltage is reversed for the same current flow direction. This can cause damage to any connected components such as a microcontroller as the flow of current into an output pin may damage or destroy the component. An output voltage from the microcontroller to the base of the transistor turns it on and allows the motor to go to ground, which in turn activates the motor. There is a resistor placed before the base of the transistor to limit current and protect the component.

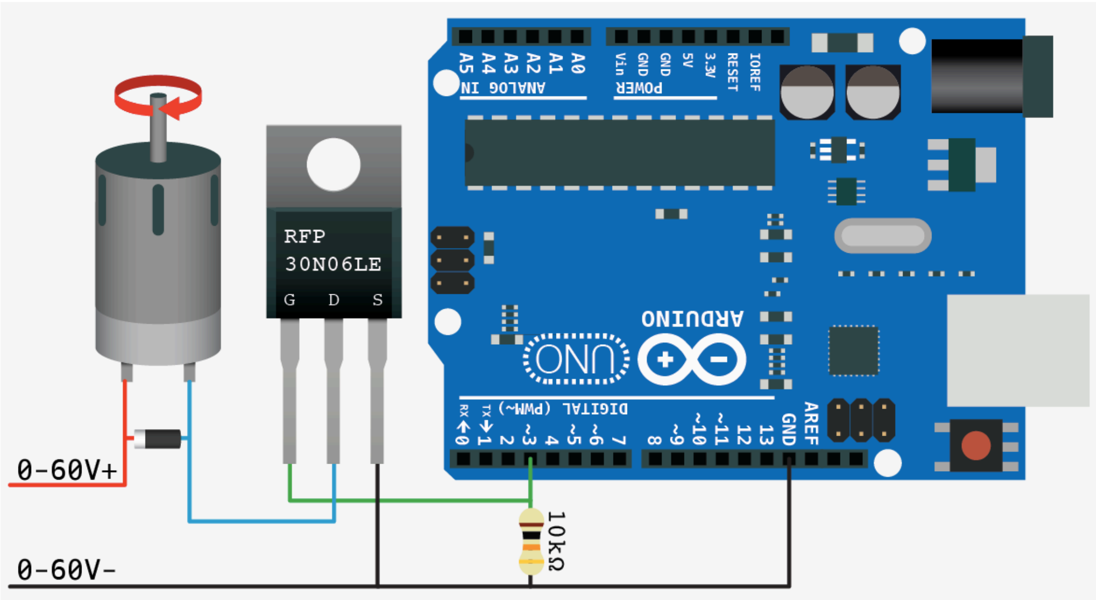

Figure 28: Diagram of motor circuit configuration.

3.6 Assembling Transmitter and Receiver Units

At this point all of the necessary hardware interfacing has been developed and completed and the required code written to add functionality. It was then required to assemble them into the two units; the 'transmitter unit' and the 'receiver unit'. The transmitter unit receives an input signal from the smoke detector and transmits a radio signal with an assigned value in it as previously discussed.

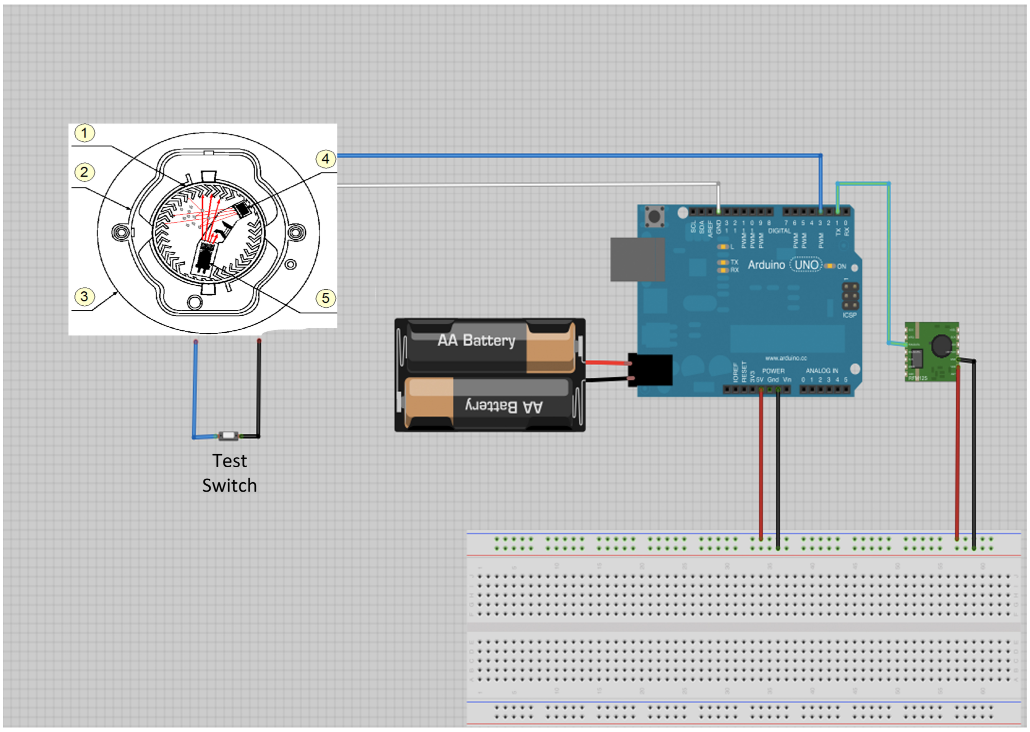

Figure 29: Assembled Transmitter Circuit.

The receiver unit receives the radio signal via the 433MHz receiver and if the value it receives matches the assigned value it performs the programmed warning actions such as make a warning call and activate the vibrating bed alert. This receiver unit can acts as a central 'server' to every smoke detector installed within a home.

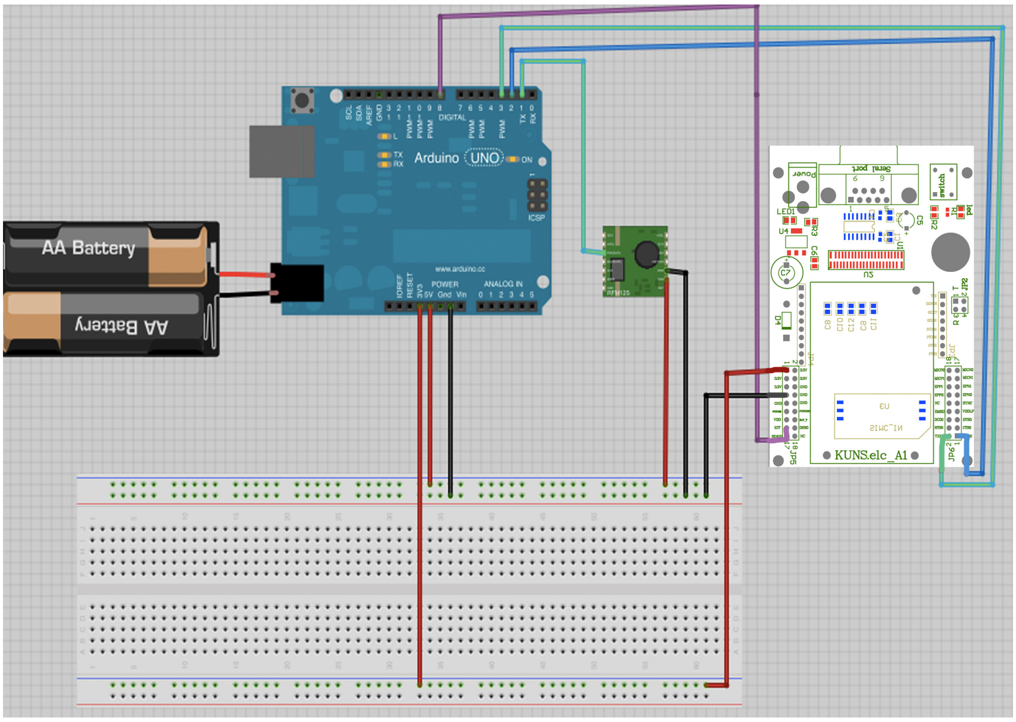

Figure 30: Assembled Receiver Unit.

3.6.1 Power Supply

For this system to perform reliably in the event of a danger of any magnitude it must possess its own suitable and dependable power supply. This means that a mains power supply cannot be used by itself, as there is the possibility of a power supply trip during any fire in a dwelling.

The Arduino board can receive power from four points; the DC jack, the Vin, 5V or 3.3V pins. A 9V power supply is recommended for regular use, which can be provided through the DC jack or the Vin pin, both of which are protected by an integrated 5V regulator. Alternatively, if a regulated 5V or 3.3V power source is available they can be provided via the respective pins of the Arduino board.

It was decided that during development a 9V Lithium battery back would be used to power each Arduino and the performance and battery life degradation would be monitored and compared to the calculated battery statistics.

The capacity of a 9V Lithium battery is 1200mAh and the approximate current consumption of an Arduino running a small amount of code and not subject to any power draws itself is 15mA, this would indicate that the transmitter unit powered by a 9V battery would have an operating life of 80 hours or 3.3333 days [23]. While the transmitter also draws current, it is a negligible amount that rises to approximately 10mA when transmitting data. This performance could be improved upon by instead using 6 1.5V AA batteries in series, where each battery has a capacity of approximately 2000mAh, meaning battery life should be 10 times greater and would give an operating time of 33.333 days.

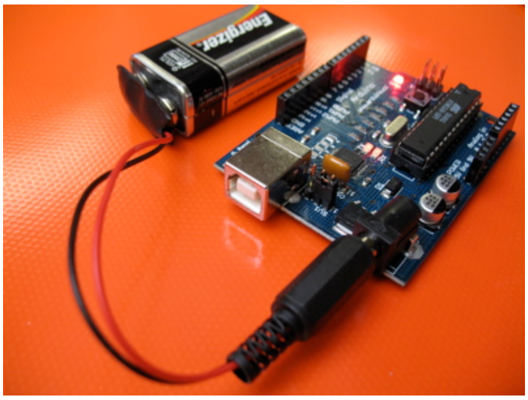

Figure 31: Arduino Uno powered by 9V battery.

The receiver unit has a heavier current draw than the transmitter unit as the RF receiver is constantly active and listening. This GSM modem is also powered on at all times. It was experimented with keeping it powered down until such time as a smoke detection alert is received by the microcontroller but it was decided that the modem start up time which is approximately 35-40 seconds, would result in an alert delay of approximately one minute and is too long to wait to alert a user.

From the transmitter datasheet it was found that the operation current draw is approximately 3mA and from the GSM modem datasheet it was observed that, in IDLE mode, the current draw is approximately 25mA [24]. This gives a total current draw of 43mA meaning for a 9V battery there would be an expected battery life of 27.9 hours or 1.16 days and for the 6x1.5V AA configuration there would be an expected life of 279.1 hours or 11.63 days.

The advantage of using an Arduino is that a battery power source can be used as a failover mechanism to a mains supply. By providing a regulated 5V battery input to the Arduinos 5V input pin, it would automatically prefer the mains supply until such time as that supply is disrupted and will then use the 5V supply.

3.7 Remote Programming of Emergency Dial Number

In order for this system to call out to a user, a phone number to dial is required. This number can be coded into the software as a default but if a consumer was to purchase this product they would require a simple, non-technical method of setting the default call number to their own number.

In order to achieve this, the method deemed most suitable was to make the default number programmable via a text message to the GSM modem. The following method was produced to check the modem for any SMS that may be stored and containing the command ‘Set’. If a message contains ‘Set’, the sender’s number is then stored as the default phone number to call.

```
void checkMessage() {

  gsmSerial.println("AT+CMGR=1 \r");    //Reads the first SMS

        for (x=0;x < 255;x++){
            data[x]='\0';
        }
        x=0;
        do{
            while(gsmSerial.available()==0);
            data[x]=gsmSerial.read();
            x++;
            //if(data[x-1]==0x0D&&data[x-2]=='"'){
                //x=0;
          // }
        }while(!(data[x-1]=='K'&&data[x-2]=='O'));

        data[x-3]='\0';         //finish the string before the OK

        delay(1000); //delay so as CMGR command returns a value before printing data

        Serial.println("\r\nMessage contains: \r");
        Serial.println(data);    //shows the message

}
```

The checkMessage() method utilises the 'CMGR' AT command to read an SMS at the designated position, in this instance position 1. It then ensures the array named 'data' is clear by iterating through it and clearing the contents of each space in the array. It then writes each character that is returned by the modem to the array, this is done so that the characters returned via the serial communication link can be analysed and manipulated. It is then terminated before the OK statement that follows every AT command is returned. The message saved is then printed out.

The readSMS() method was then developed to analyse the contents of the SMS to check if a 'Set' command has been received:

```
void resetSMS() {
  boolean contains = false;
  for (x = 0; x<strlen(data); x++) {
    if (data[x]=='S'&&data[x+1]=='e'&&data[x+2]=='t'){
      contains = true;
      break;
        }
        else {
            contains = false;
        }
    }
  if (contains == true){
        Serial.println("This SMS contains the word; 'Set'. \r");
        for (x = 0; x<strlen(data); x++) {
        if (data[x]=='+'&&data[x+1]=='3'&&data[x+2]=='5'){  //finds characters beginning with +35 and then saves the number
          for(int i=0;i<13;i++){
            ph_number[i]=data[x];
            EEPROM.write(i, data[x]);

            x++;
          }
          Serial.println("This message was sent from: ");
          Serial.println(ph_number); Serial.println("\rThis is the new emergency call number.\r");
          delete_All_SMS();
          }
        }
  }
  else{
    Serial.println("There is no set message at this time. \r");
  }
```

This method implements an iterative loop by first searching for the character ‘S’ then if it is located and the character at the next location is ‘e’ and the following character is ‘t’ a positive match is indicated by way of a Boolean variable titled ‘contains’.

If ‘contains’ has deemed to be true, the array is then looped through once again to identify a senders phone number. If the characters ‘+’, ‘3’, ‘5’ are found in succession then an Irish phone number has been identified and the characters plus the following 10 characters in the array are stored as the new emergency dial number.

Obviously, it would be highly inconvenient for this process to have to be repeated every time the microcontroller is restarted, fortunately there is permanent programmable memory available on the Arduino which can be wrote to with the ‘EEPROM.write’ command followed by the location that is desire to be wrote to followed by the value to be assigned to that location.

The stored phone number can then be read from memory upon start up meaning that the emergency dial number will not be changed until such time as a new set command is issued:

```
  if(EEPROM.read(0)!=255){
 for(int y=0; y<13; y++){
   ph_number[y] = EEPROM.read(y);
 }
 Serial.println("\rThe emergency contact number is: \r");
 Serial.println(ph_number);
 }
```

3.8 Remaining Code Explained

Upon start up of the receiver unit Arduino a ‘vibratePin’ is declared. When the known alert value is received by the RF receiver from a smoke detecting transmitter unit, this system will call out to the programmed emergency alert number and then activate the under-pillow vibration motor by setting the output pin HIGH for 2 seconds and then LOW again followed by a delay of 8.5 seconds before hanging up the warning call. This delay is implemented as there is a 4-5 second delay in dialling out from the modem so this will mean 6-7 seconds of dialling time is achieved:

```
void loop()
{
  boolean light_led = false;

  if (readUInt(true) == 288) // Check to see if we got the 88 test number
  {
    light_led = true;
      Serial.println('Received');              // debug value

//  }

//  if (light_led)
//  {
      digitalWrite(LED_PIN, HIGH);

     gsmSerial.write("ATD+353879149715;\r"); //code to call phone
     digitalWrite(vibratePin, HIGH);
     delay(2000);
     digitalWrite(vibratePin, LOW);
     delay(8500); //11 second delay as takes about 5 seconds to connect call
     gsmSerial.print("ATH0;\r"); //kill call

     digitalWrite(LED_PIN, LOW);

  }
}
```

The TC35i modem possesses an ignition button in order to start the modem's GSM transmission. In order to bypass the button so that it does not have to be pressed manually upon every start up, the IGT pin on the board can be utilised.

If this pin is grounded for a short amount of time it is the same as pressing the button. This is achieved by implementing the following piece of code in the Arduino setup loop to activate the modem.

```
 //--- turn on TC35 ---
// wire pin 8 Arduino to IGT pin on TC35
// it grounds IGN pin for 100 ms
// this is the same as pressing the button
// on the TC35 to start it up

pinMode(8, INPUT);
digitalWrite(8, LOW);
pinMode(8, OUTPUT);
delay(100);
pinMode(8, INPUT);
```

After a new 'Set' message is detected, it is required that all messages are deleted so that the same message is not analysed over and over and thus creates unnecessary computational expense and power consumption. This is achieved by deleting all stored messages from the modem using the following method:

```
void delete_All_SMS(){
  for(int i = 1; i <= 10; i++) {
    gsmSerial.print("AT+CMGD=");
    gsmSerial.println(i);
    Serial.print("deleting SMS ");
    Serial.println(i);
    delay(500);
  }

}
```

The above code iterates through messages stored at location 1 through to 10 and deletes any messages stored at any of these locations by using the message delete AT command 'CMGD'.

Chapter 4 – Testing, Results and Issues

4.1 Introduction

This chapter will discuss the testing methodology designed to ensure reliable operation of the system and to uncover any potential operational or design issues that may arise in differing scenarios. It will then discuss the results of these tests and any issues that happened to be uncovered over the course of the design, development and testing phases of the project.

4.2 Testing and Results

The following tests were designed to ensure the reliability of the system in all scenarios.

4.2.1 Detection Accuracy

In order to guarantee a reliable alert to the user upon every smoke detection, this test was designed to observe the overall alert accuracy of the system and to quantify the performance characteristics.

The smoke detector was activated a total of twenty times by using both the test button for ten activations and canned smoke for another ten activations and the results recorded:

Activation No.	Smoke Alarm Activation	Phone Alert Activation	Vibration Alert Activation	Overall Result
1	Pass	Pass	Pass	Pass
2	Pass	Pass	Pass	Pass
3	Pass	Pass	Pass	Pass
4	Pass	Pass	Pass	Pass
5	Pass	Pass	Pass	Pass
6	Pass	Pass	Pass	Pass
7	Pass	Pass	Pass	Pass
8	Pass	Pass	Pass	Pass
9	Pass	Pass	Pass	Pass

10	Pass	Pass	Pass	Pass
11	Pass	Pass	Pass	Pass
12	Pass	Pass	Pass	Pass
13	Pass	Pass	Pass	Pass
14	Pass	Pass	Pass	Pass
15	Pass	Pass	Pass	Pass
16	Pass	Pass	Pass	Pass
17	Pass	Pass	Pass	Pass
18	Pass	Pass	Pass	Pass
19	Pass	Pass	Pass	Pass
20	Pass	Pass	Pass	Pass

Figure 32: Activation reliability test results.

4.2.1.1 Conclusions

From the above results it can be observed that the system will reliably alert a user in the instance of a smoke detection. In each of the activation cases above, the time between smoke detection and a successful phone alert was 6-7 seconds. This was deemed to be an acceptable performance benchmark as the limitations of GSM technology, in connecting a phone call, enforce such a delay.

4.2.2 Communication Range

In order for this system to work accurately in the average dwelling it was deemed that the transmitter and receiver pair should be able to communicate at up to a range of 40 metres within a household environment containing impedances such as walls, doors and furniture. The datasheets for both components state an approximate range of 150 metres in open space so it was expected that a range of 40 metres could reliably be obtained.

Distance Tested (metres)	Impedances	Result
5	None	**Pass**
10	Door	**Pass**
20	Door, Wall	**Pass**
30	Different levels of house	**Pass**
40	Different levels of house	**Pass**

Figure 33: Range testing results.

4.2.2.1 Conclusions

These results were deemed to be satisfactory as it could ensure the successful operation of the system inside an average dwelling. When the range was pushed to 60 metres and tested three times it passed on one occasion and failed on two. After researching RF transmitters and receivers, it was found that adding a 20cm length wire antenna, slightly increasing the power supply voltage and decreasing the baud rate are all effective methods of increasing the range of the components.

4.2.3 Remotely Programmable Phone Number

The code developed to allow remote programming of the emergency contact number was tested to ensure 100% reliability. This feature is vital to the operation of the system as any user must be able to program their emergency contact number with zero error.

Test No.	Condition	Result
1	No phone number stored.	Pass
2	Overwrite existing number.	Pass
3	Overwrite existing number.	Pass
4	SMS without 'Set' command.	Pass

Figure 34: Emergency number programming test.

4.2.3.1 Conclusion

It was found from testing that the code worked extremely reliably and every time a 'Set' command was sent via SMS the controller saved the senders number as the new emergency contact number. Additionally any SMS command that did not contain the command was disregarded and deleted.

4.3 Issues

4.3.1 Battery Life

As discussed the 'Power Supply' section of the Design chapter, battery life for this project is a concern. The expected battery life when supplied by 6x1.5V AA batteries, as previously calculated, is 33.333 days for the transmitter unit and 11.63 days for the receiver unit. These results would mean that utilising this system in a house, in it's current design, would be impractical as a battery powered only system.

These battery concerns were prevalent throughout the development of the project and as such had an impact on design decisions made during development. One such design consideration, was the implementation of an interface to allow a user to manually input their desired emergency contact number. An LCD interface with a selection of push buttons was obtained and code was developed to take input from the interface and save it within the Arduino EEPROM memory. However the LCD output consumed too great an amount of power and this avenue of inputting a contact number was then discarded in favour of the text message method which is far more efficient.

4.3.1.1 Solutions

There are two considerations to counter the poor power consumption. Firstly, the Arduino can be powered from a mains supply as previously discussed, meaning that a battery backup can be utilised that will support the device in the event of a power outage similar to the mains connected smoke detectors in a lot of buildings today.

Secondly, this is a prototype proof-of concept solution to the challenge posed by the project brief. As discussed in the technology review, the Arduino board and TC35i development board were both chosen for initial development and prototyping and both consume a much greater amount of power than any integrated solution would. For example, the Arduinos voltage regulator and Analog to Digital Converter both draw excess current that would not be drawn if using an embedded PIC in an integrated circuit. There are similar unnecessary components and features on the TC35i board that draw consume excess power.

4.4 Conclusions

The testing methodology implemented covered all of the functionality of the system and ensured reliable operation and performance to the point where it can be accepted that the final prototype design is technically successful.

However, there are some issues to be addressed in terms of power consumption that would ideally be solved when designing an integrated solution containing only essential circuitry and designed for optimum performance.

Chapter 5 – Ethics and Responsibility

5.1 Introduction

This chapter will discuss the ethical issues surrounding the development of this project. The four fundamental engineering principles outlined by the Royal Academy of Engineering's statement of ethical principles are: Accuracy and Rigour, Honesty and Integrity, Respect for Life, Law and the Public Good and Responsible Leadership [25]. The scope of this project covers at least two of these principles.

5.2 Safety and Reliability

By the nature of this system, it is based around the most important thing in existence; the health and safety of a human life. If this system is to be produced as an effective warning to a threat to a human life then only the most robust and thoroughly developed and tested solution must be produced and sold. Such a product must have very clear printed operating capabilities and limitations made available to a user or consumer so that they know exactly the scope of protection this device will serve to them and their family.

5.3 Targeted Testing

Before allowing the final product to go on to the market, extensive testing would be required using participants from the target market, the deaf and hard of hearing. Only by using these targeted test participants can the effectiveness of the device truly be measured and quantified.

5.4 Legal Implications

Fire safety equipment is regulated by national standards in every developed country in the world, as discussed in the introduction to the report. Within Ireland, equipment is covered by both Irish standards and European standards, with a different set of standards in place in the UK.

In developing a safety product it must be ensured that it meets the requirements of national standards in every country it will be sold in, for financial, legal and ethical reasons. If a device does not meet one particular standard and is found to be at fault in the event of an accident, it could have disastrous implications for both the company and developers of the product.

It must also be ensured that any product produced, is developed using designs, code and any other intellectual assets that are entirely the property of the producers of the product or have been licensed to them and recognised accordingly.

Chapter 6 – Conclusions and Further Development

The culmination of the project period has produced a working technical prototype solution to the challenge outlined in the brief. Throughout this period a vast amount of knowledge was learned in the areas of interfacing hardware, microcontroller and software, applying electronics theory to practical applications and developing efficient code for use in a physical system. Research and investigative skills were also greatly improved upon through the research of academic publications, datasheets and web-based services such as IEEE Xplore and Google Scholar.

Important milestones during the development of the project were:

- Interfacing the GSM modem with microcontroller to make calls and send SMS'
- Reverse engineering smoke alarm to emit an output signal when a detection occurs
- Communicating wirelessly through RF components
- Developing code to manage the modems serial output and make the emergency phone number programmable

This project also helped me to develop my problem solving, analytical and organisational skills through iteratively breaking down each challenge posed by the project and researching the optimum solution in each instance and planning work schedules and regular reports to my supervisor.

6.1 Further Development

With the time constraints imposed on this project and by the very nature that it is a solo project and thus a steep learning curve, there is a lot of additional steps I would like to take to develop the solution further.

The first and most significant step I would take if given additional time and opportunity to develop the project would be to transition from Arduino controllers to embedded PIC controllers. The PICs would be embedded in a printed circuit board along with the smoke detector circuitry and RF transmitter so that the entire device could be enclosed within a traditional smoke detector casing. This solution would then be far more efficient in terms of power consumption and thus be realistically operational through the use of batteries.

This step would be repeated with the centralised receiver unit to embed a GSM modem, microcontroller and RF receiver within an integrated circuit, optimised for minimum power consumption. This would then remove the one issue currently present in my solution, battery life.

6.2 Future Plans

I hope to display my project at the Final Year Project Expo hosted by the School of Engineering and demonstrate its functionality to guests of the event.

At the conclusion of my studies in DCU, I will be commencing a graduate role with a large international software company who I have recently signed a contract with. I attribute my success in obtaining this job offer to possessing the required personal and professional which have been developed throughout my studies in DCU and through the huge amount of knowledge gathered over the duration of my Final Year Project.

Works Cited

[1] Merriam-Webster, "Defintion of Deaf",

Available: http://www.merriam-webster.com/dictionary/deaf

[2] National Association for Deaf People, "Information Publication".

Available: http://www.irishhealth.com/clin/documents/information_deafness.pdf

[3] A. Davis, "Hearing in Adults: The Prevalence and Distribution of Hearing Impairment and Reported Hearing Disability in the MRC Institute of Hearing Research's National Study of Hearing", Whurr, 1995.

[4] CFPA Europe, "Smoke Alarms in the Home".

Available: http://www.swissi.ch/uploads/media/CFPAEGuideline10neu_01.pdf

[5] Fire Investigators Association of Ireland, "Fire Statistics for the Republic of Ireland 2007", 2007. Available: http://fireinvestigation.ie/irish-fire-statistics

[6] United States Department of Labour, "Noise and Hearing Conservation: A Guide", Available: http://www.osha.gov/dts/osta/otm/noise/health_effects/physics.html

[7] Central Statistics Office, "This is Ireland, Highlights from Census 2011". Dublin: Stationery Office 2012.

[8] Shargorodsky, J., Curhan, G., Farwell, W., "Prevalence and Characteristics of Tinnitus among US Adults". *The American Journal of Medicine*. 123, 711-718. 2010.

[9] Bruck, D., "Arousal from sleep with a smoke detector in children and adults (Technical report FCRC-TR 98-04)". Sydney, Australia: Fire Reform Center. 1998.

[10] Dorothy Bruck, Ian Thomas, "Waking effectiveness of alarms (audio, visual and tactile) for adults who are hard of hearing." *Optimizing Fire Alarm Notification for High Risk Groups Research Project.* The Fire Protection Research Foundation. June 2007.

Available: http://www.nfpa.org/assets/files/pdf/research/hardofhearing&alarms.pdf

[11] Su, J.Z., Crampton, G.P., Carpenter, D.W., McCartney, C. and Leroux, P.,
''Kemano Fire Studies – Part 1: Response of Residential Smoke Alarms,''
Research Report 108, National Research Council of Canada, Ottawa, Canada,
April 2003

[12] Thomas Cleary. "Performance of Dual Photoelectric/Ionization Smoke Alarms in Full-Scale Fire Tests". Fire Protection Research Foundation's 13th annual Suppression and Detection Research & Applications Symposium (SUPDET), February 24-27, 2009.

Available: http://fire.nist.gov/bfrlpubs/fire09/PDF/f09006.pdf

[13] Erik Jensen. “Smart video phones are a sign of better days for deaf users”. *The Sydney Morning Herald* (March. 8, 2011), Health.

Available: http://www.smh.com.au/digital-life/mobiles/smart-video-phones-are-a-sign-of-better-days-for-deaf-users-20110307-1bl9r.html

[14] Joseph M. Kahn. “Wireless Infrared Communications”. *Proceedings of the IEEE.* (Feb 1997). 85(2), pp.265-298.

Available: http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=554222&tag=1

[15] Patricia McDermott-Wells. “What is Bluetooth?”. *Potentials, IEEE.* (Dec 2004-Jan 2005). 23(5), pp. 33-35.

Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1368913

[16] Microchip, “PIC16F87X Datasheet”.

Available: http://ww1.microchip.com/downloads/en/devicedoc/30292c.pdf

[17] "Programming Arduino Getting Started with Sketches". McGraw-Hill. Nov 8, 2011. Retrieved 2013-03-28.

[18] Arduino Reference Guide, “SoftwareSerial Library”.

Available: http://arduino.cc/en/Reference/SoftwareSerial

[19] Frank Durda, “The AT Command Set Reference – History”. 1993

Available: http://nemesis.lonestar.org/reference/telecom/modems/at/history.html

[20] “AT Commands Reference Guide”, Telit. 2006 Available: https://www.sparkfun.com/datasheets/Cellular%20Modules/AT_Commands_Reference_Guide_r0.pdf

[21] Freescale Semiconductor, “MC145010 Photoelectric Smoke Detector IC Datasheet”.

Available: http://www.freescale.com/files/analog/doc/data_sheet/MC145010.pdf

[22] Jeff Gray, “KLP/KLPA Module Walkthrough”. Spark Fun Electronics.

Available: https://www.sparkfun.com/datasheets/RF/KLP_Walkthrough.pdf

[23] Arduino UNO Datasheet.

Available: http://arduino.cc/en/Main/arduinoBoardUno

[24] Siemens, “TC35i Wireless Module Hardware Interface Description”.

[25] The Royal Academy of Engineering, "Statement of Ethical Principles," [Online]. Available: http://www.engc.org.uk/ecukdocuments/internet/document%20library/Stateme nt %20of%20Ethical%20Principles.pdf. [Accessed 16 04 2012].

Appendices

8.1 Transmitter Unit Code

Transmittercode.ino

```
//Transmitter Code
//David Burson FYP

int analogPin = A0;
int x = 0;
int ledPin = 13;                    // LED connected to digital pin 13

void setup()
{
  pinMode(analogPin, INPUT);
  pinMode(ledPin, INPUT);
  pinMode(ledPin, OUTPUT);
  digitalWrite(ledPin, LOW);

  Serial.begin(1200);  // Hardware supports up to 2400, but 1200 gives longer range
}

void loop()
{
  x = analogRead(analogPin);    // read the input pin
  float voltage = x * (5.0 / 1023.0);

  if (voltage>3) {
    Serial.println(voltage);             // debug value
    Serial.println("\r\n");

    writeUInt(288); // Put any number you want to send here (71 is just a test)

   digitalWrite(ledPin, HIGH);   // sets the LED on
    delay(1000);                  // waits for a second
    digitalWrite(ledPin, LOW);    // sets the LED off
    delay(10000);
  }
}
```

8.2 Debug Code

Debug.ino

```
#define NETWORK_SIG_SIZE 3

#define VAL_SIZE         2
#define CHECKSUM_SIZE    1
#define PACKET_SIZE      (NETWORK_SIG_SIZE + VAL_SIZE + CHECKSUM_SIZE)

// The network address byte and can be change if you want to run different devices in proximity to each other without
#define NET_ADDR 5

const byte g_network_sig[NETWORK_SIG_SIZE] = {0x8F, 0xAA, NET_ADDR};  // Few bytes used to initiate a transfer

// Sends an unsigned int over the RF network
void writeUInt(unsigned int val)
{
  byte checksum = (val/256) ^ (val&0xFF);
  Serial.write(0xF0);  // This gets reciever in sync with transmitter
  Serial.write(g_network_sig, NETWORK_SIG_SIZE);
  Serial.write((byte*)&val, VAL_SIZE);
  Serial.write(checksum); //CHECKSUM_SIZE
}

// Receives an unsigned int over the RF network
unsigned int readUInt(bool wait)
{
  int pos = 0;          // Position in the network signature
  unsigned int val;     // Value of the unsigned int
  byte c = 0;           // Current byte

  if((Serial.available() < PACKET_SIZE) && (wait == false))
  {
    return 0xFFFF;
  }


    while(pos < NETWORK_SIG_SIZE)
    {
      while(Serial.available() == 0); // Wait until something is avalible
      c = Serial.read();

      if (c == g_network_sig[pos])
      {
        if (pos == NETWORK_SIG_SIZE-1)
        {
          byte checksum;

          while(Serial.available() < VAL_SIZE + CHECKSUM_SIZE); // Wait until something is avalible
          val      =  Serial.read();
          val      += ((unsigned int)Serial.read())*256;
          checksum =  Serial.read();

          if (checksum != ((val/256) ^ (val&0xFF)))
          {
            // Checksum failed
            pos = -1;
          }
        }
        ++pos;
      }
      else if (c == g_network_sig[0])
      {
        pos = 1;
      }
      else
      {
        pos = 0;
        if (!wait)
        {
          return 0xFFFF;
        }
      }
    }
    return val;
  }
```

8.3 Receiver Unit Code

Receivercode.ino

```
#include <EEPROM.h>

#include <SoftwareSerial.h>
SoftwareSerial gsmSerial(2,3);

char gsm_char,gsm_char_send, data[256], ph_number[13];//="+353879149715";
int x;

#define LED_PIN 13
#define vibratePin 5

void setup()
{
  pinMode(LED_PIN, OUTPUT);
  pinMode(vibratePin, OUTPUT);
  Serial.begin(1200);  // Hardware supports up to 2400, but 1200 gives longer range

  gsmSerial.begin(1200);

  if(EEPROM.read(0)!=255){
 for(int y=0; y<13; y++){
   ph_number[y] = EEPROM.read(y);
 }
 Serial.println("\rThe emergency contact number is: \r");
 Serial.println(ph_number);
 }

   gsmSerial.print("AT+CMGF=1 \r"); // Setting GSM Module to message mode
   delay(600);

  Serial.begin(1200);  // Hardware supports up to 2400, but 1200 gives longer range
  gsmSerial.begin(1200);

   //--- turn on TC35 ---
  // wire pin 8 Arduino to IGT pin on TC35
  // it grounds IGN pin for 100 ms
  // this is the same as pressing the button
  // on the TC35 to start it up

  pinMode(8, INPUT);
  digitalWrite(8, LOW);
  pinMode(8, OUTPUT);
  delay(100);
  pinMode(8, INPUT);

  delay(1000);

  // Checking for any new messages containing a set command to reset emergency number
  for(int i=0;i<3;i++){
  checkMessage();
  resetSMS();
 delay(500);
  }
```

```
void loop()
{
  boolean light_led = false;

  if (readUInt(true) == 288) // Check to see if we got the 88 test number
  {
    light_led = true;
      Serial.println('Received');             // debug value

//  }

//  if (light_led)
//  {
      digitalWrite(LED_PIN, HIGH);

     gsmSerial.write("ATD+353879149715;\r"); //code to call phone
     digitalWrite(vibratePin, HIGH);
     delay(2000);
     digitalWrite(vibratePin, LOW);
     delay(8500); //11 second delay as takes about 5 seconds to connect call
     gsmSerial.print("ATH0;\r"); //kill call

     digitalWrite(LED_PIN, LOW);

  }
}

void delete_All_SMS(){
  for(int i = 1; i <= 10; i++) {
    gsmSerial.print("AT+CMGD=");
    gsmSerial.println(i);
    Serial.print("deleting SMS ");
    Serial.println(i);
    delay(500);
  }

}

void sendSMS(){

  gsmSerial.print("AT+CMGF=1\r");
  delay(1000);
 gsmSerial.println("AT+CMGS=\"+353879149715\""); //AT command to send SMS
 delay(500); //the length of this delay is very important. 400 is too low
 gsmSerial.print("Hello World"); //Print the message
 delay(100);
 gsmSerial.println((char)26);
  Serial.println("\r\nSMS Sent\r\n");

}
```

```
void checkMessage() {

  gsmSerial.println("AT+CMGR=1 \r");     //Reads the first SMS

        for (x=0;x < 255;x++){
            data[x]='\0';
        }
        x=0;
        do{
            while(gsmSerial.available()==0);
            data[x]=gsmSerial.read();
            x++;
            //if(data[x-1]==0x0D&&data[x-2]=='"'){
                //x=0;
           // }
        }while(!(data[x-1]=='K'&&data[x-2]=='O'));

        data[x-3]='\0';          //finish the string before the OK

        delay(1000); //delay so as CMGR command returns a value before printing data

        Serial.println("\r\nMessage contains: \r");
        Serial.println(data);     //shows the message

}
```

```
void resetSMS() {
  boolean contains = false;
  for (x = 0; x<strlen(data); x++) {
    if (data[x]=='S'&&data[x+1]=='e'&&data[x+2]=='t'){
      contains = true;
      break;
        }
        else {
            contains = false;
        }
    }
  if (contains == true){
        Serial.println("This SMS contains the word; 'Set'. \r");
        for (x = 0; x<strlen(data); x++) {
        if (data[x]=='+'&&data[x+1]=='3'&&data[x+2]=='5'){  //finds characters beginning with +35 and then saves the number
          for(int i=0;i<13;i++){
            ph_number[i]=data[x];
            EEPROM.write(i, data[x]);

            x++;
          }
          Serial.println("This message was sent from: ");
          Serial.println(ph_number); Serial.println("\rThis is the new emergency call number.\r");
          delete_All_SMS();
          }
        }
  }
  else{
    Serial.println("There is no set message at this time. \r");
  }

}
```