OXYGEN PRESSURE RESCUE-I MONITORING

By

NORHUSAIDIL BIN HUSSAIN

Dissertation Report

Submitted to the Electrical & Electronics Engineering Program

In Partial Fulfillment of the Requirement for the Degree

Bachelor of Engineering (Hons)

(Electrical & Electronics Engineering)

May 2012

Universiti Teknologi PETRONAS

Bandar Seri Iskandar, 31750 Tronoh

Perak Darul Ridzuan

CERTIFICATION OF APPROVAL

Oxygen Pressure Rescue-i Monitoring

By

NORHUSAIDIL BIN HUSSAIN

A project dissertation submitted to

Electrical and Electronics Engineering Programme

Universiti Teknologi PETRONAS

in partial fulfillment of the requirements for the

BACHELOR OF ENGINEERING (Hons)

(ELECTRICAL AND ELECTRONICS ENGINEERING)

Approved:

_____

Ir. Dr. Nursyarizal Mohd Nor

Project Supervisor

UNIVERSITI TEKNOLOGI PETRONAS

TRONOH, PERAK

MAY 2012

CERTIFICATION OF ORIGINALITY

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

_____

NORHUSAIDIL BIN HUSSAIN

ACKNOWLEDGEMENT

First and foremost, I would like to express my gratitude to God for giving me the strength to complete this project. Next, I would like to thanks to my parents and all my family members who give me support and give wise advice when I am in problem.

My sincere appreciation and thanks to supervisor and co-supervisor, Ir Dr Nursyarizal Bin Mohd Nor and Mdm. Hanita Binti Daud, who found time very busy schedule to give me consultation, monitor my progress and guide me throughout the project. Their friendliness and willingness to share the knowledge, experience as well as procedures enhance both my knowledge and skills.

I would like to thank to Dr. Fawnizu Azmadi Hussin for lending me some of the items related to my project. Even the items is quite expensive, he still allows me to use it before I can buy it myself. While utilizing his items, I was able to learn more about the item, hence I managed to complete the project earlier than expected.

I also would like to appreciate to my institution, Universiti Teknologi PETRONAS, whom also just become the first private university to receive accreditation by Standard Malaysia especially Electrical and Electronics Engineering Department, by providing me the good facility and budget for me to accomplish my project.

Finally, I would like to thank my colleagues, who always are there, days and night, willing to spend time to give free consultation and also helping in troubleshooting my projects. Without them, I couldn't finish my project by myself.

ABSTRACT

While performing rescue task, the condition of the rescuer himself must be fit enough in order to face and sustain such dangerous and hazardous environment. There have been few cases that the rescuer could not survive and died before he could save other people. To reduce the risk of rescuer injured or died during that particular event, a mechanism to monitor the condition of the rescuer should be implemented. The data and information of the rescuer will be sending out to the fire engine wirelessly. When any accident or emergency happens, another rescuer from the fire engine will come and help the distress rescuer.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

PASS	Personal Alert Safety System
SCAMP	Supercritical Air Mobility Pack
SCBA	Self-contained Breathing Apparatus
IrDA	Infrared Data Association
UART	Universal Asynchronous Receiver/Transmitter
USB	Universal Serial Bus
GUI	Graphical User Interface
VB	Visual Basic
IDE	Integrated Development Environment
LED	Light Emitting Diode
PC	Personal Computer
FTDI	Future Technology Devices International

CHAPTER 1

INTRODUCTION

1.1 Background of Study

Rescue squad plays an important role in our society. They performed various kinds of rescue activities such as fire rescue, cave rescue, mine rescue and so on. They usually used miscellaneous tools and special devices including search dogs, horses, helicopter, hydraulic machines and others in order to reach and remove humans and animals from tight location. They also required to undergo specific intensive training, and sometime might collaborate with organization such as police or military. Different countries have their own rescue team division and responsibilities. For example, in Malaysia, they have Jabatan Bomba dan Penyelamat (Department of Rescue and Firefighting) while in Brazil they have Para-SAR, the Brazilian Air Force [1,2].

1.2 Problem Statement

Rescuer always get expose towards uncomfortable and hazardous environment. On 6th September 2010, a 51-year-old male volunteer fire fighter (victim) died after being overcome by low oxygen and sewer gases while climbing down into a sewer manhole in an attempt to rescue a village utility worker [3]. There were also few other cases that involve the death of the rescuer while performing rescue task. Hence, it is very crucial to monitor their condition at all time. Some parameters such as pressure level of the rebreather and heartbeat rate is very useful in benchmarking the condition of the rescuer. Once the data is collected, fast and responsive action should be taken immediately.

1.3 Objectives

The main objective of this project is to monitor the pressure level of the rebreather of the rescuer. This can be done by attaching a pressure transducer to the breathing pipe. In this project as well, the mechanism to allow data communication between the rescuers inside the building or closed area with the rescuer at the fire engine, normally stays outside the closed area will be applied. The protocol used for this wireless communication is called zigbee.

As a summary, this project should be able to:

➢ To monitor and transmit the pressure level of the rebreather wirelessly
➢ The alert the rescuer when the pressure level reach certain level
➢ To design a new product while maintaining the design of the pressure tank
➢ To improve the available product in term of cost and performance

1.4 Scope of Study

Generally, there are two types of rescue squads. The first one is to rescue trapped people, and the other one is to rescue people who are having medical emergencies [4]. Medical rescue is everywhere, such as during sports event, when an athletic experiences some injuries, there will be a group of people going to treat that particular athletic. This project will focus on trapped people, normally will be performed by firefighters. For wireless communication, zigbee has a ability to support mesh network topology, which means several nodes, in this case is rescuer, can be monitor at the same time. However, this project will focus on monitoring only one rescuer.

1.5 Feasibility of Project

During FYP 1, the project will focus more on research and study, equipment survey and programming development. For FYP 2, intensive prototype construction, debug and troubleshooting will be the main highlight. Details are shown on Gantt chart in Chapter 3.

CHAPTER 2

LITERATURE REVIEW

Based on small research and survey that had being done, there were some similar products developed in order to ensure that the rescuer is safe. Two of them will be explained briefly in this section. The first one was Personal Alert Safety System (PASS) device and the second one is called Supercritical Air Mobility Pack (SCAMP).

2.1 Personal Alert Safety System (PASS) Device

PASS device will produce a loud audible sound, at least 95 decibels, to notify others in the area that a rescuer is in distress. If the device does not detect any motion for typically 15-30 seconds, it will automatically activate so that the alert will sound if the rescuer is immobile or seriously injured. This device can also be activated manually in an emergency, such as the rescuer is trapped. PASS device will be integrated with SCBA worn by rescuer as shown in Figure 1. It is also capable to operate in flammable or explosive environment [6].
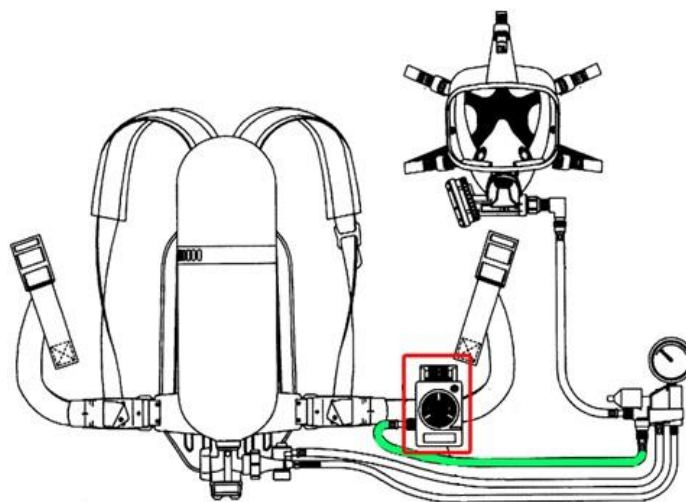
Figure 1: PASS Device integrated with SCBA

Experiment showed that PASS device was able to sustain in an extreme condition. Three types of tests were conducted, namely Static Oven Test, Flow Loop Test and Mannequin Test. For Static Oven Test, PASS device will be suspended inside an electric oven with dimension of 0.44 m x 0.29 m x 0.39 m as shown in Figure 2. The temperature of the oven was varying from 18° C to 250° C [6].
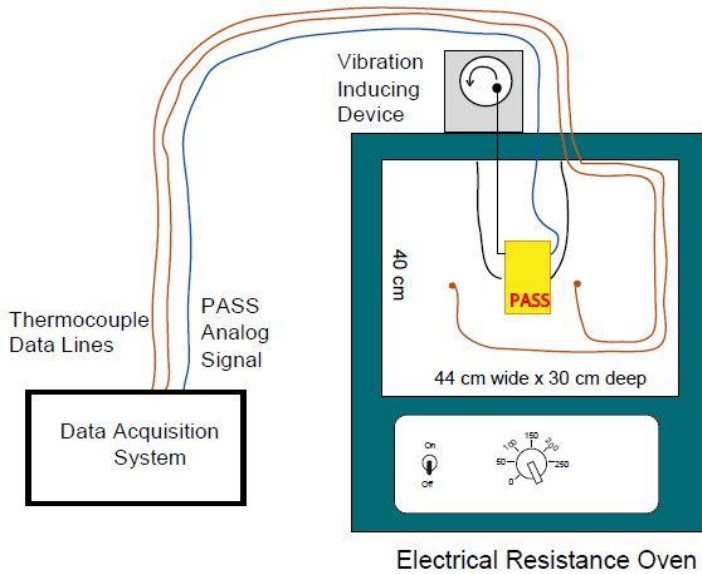
(a)

(b)

Figure 2 (a) & (b): Schematic and setup for Static Oven Test [6]

For Flow Loop Test, PASS device will be suspended inside a flow loop with a dimension of 0.36 m x 0.2 m x 0.2 m as shown in Figure 3. The blower will blow the heat produced by resistance heater in order to maintain the specific temperature of the gases. The gas temperature of the flow loop was varying continuously between 18° C and 150° C [6].
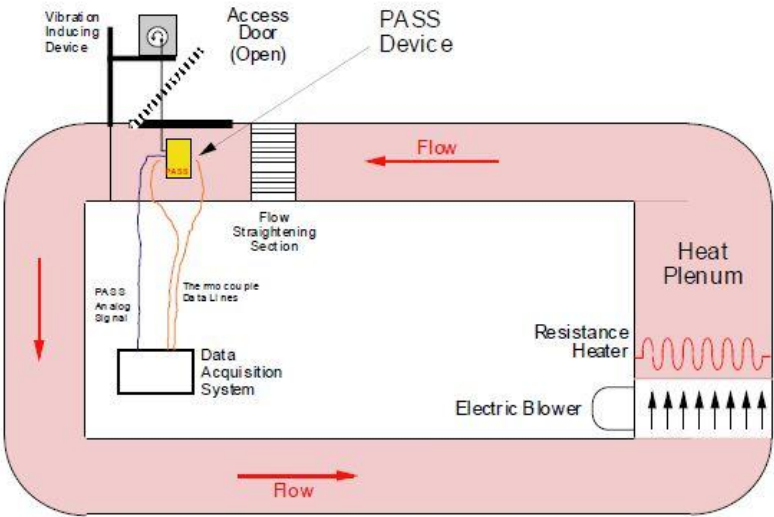
(a)

(b)

Figure 3 (a) & (b): Schematic and setup for Flow Loop Test [6]

And the last test is the Mannequin Test, where the device was tested under more realistic condition as shown in Figure 4. Six PASS devices will be attached on a mannequin. The response of PASS device was continuously monitored throughout different set of fire exposure [6].

(a)

(b)

Figure 4 (a) & (b): Setup for Mannequin Test [6]

However, there is no guarantee that other rescues will alert or might be able to save the distress rescuer due to unexpected environment. To improve this system, there should be some mechanism to inform other rescuers that stay outside from the incident area, so that these rescuers will be able to come inside and save the distress rescuer.

2.2 Supercritical Air Mobility Pack (SCAMP)

SCAMP was developed by Aerospace Design and Development (ADD), Inc. of Niwot, Colorado. This system provide both cooling and breathing from supercritical cooled (cryogenic) air as shown in Figure 5. This cryogenic compressed air is stored in a vacuum bottle called dewar. Heat from rescuer body will warm the cold air to a temperature around 50-60° F at the backpack which consists of heat exchanger. The rescuer wear an undergarment laced with tubes consisting antifreeze mixture and a flowing water. This liquid will be heated by rescuer body, pumped to the heat exchanger and then being cooled by the cold air from dewar. The cooling and breathing air come from the same source.

This system is slightly smaller and lighter than conventional SCBA equipment. It also provide longer operating period due to high fluid density and low pressure in the dewar. A capacitor inside the tank will measures the density of the air and a computer will calculate the remaining air left. The rescuer will be inform to recharge by a beeper alarm and LCD display mounted in the mask [7].

Overall the above system is acceptable, but it can be improvised by installing the monitoring mechanism. No one knows what the condition of the rescuer is once the rescuer is inside the building. The rescuer might trapped or involve in an accident. All those heat exchanger and dewar couldn't do anything for the rescuer. He needs support from outside. Hence, monitoring mechanism should be implemented in this system.

Figure 5: SCAMP System [8]

2.3 Wireless Data Communication

Wireless data communication is a process of data transfer via communication network without any physical connection [9]. In order to send data wirelessly, it required certain protocol that fulfill the user requirement. Bluetooh, IrDA, WiFi and Zigbee are examples of wireless protocol.

Bluetooth technology is the global short-range wireless standard enabling connectivity for broad range of electronic devices. It features include small form factor radio, low power, low cost, built-in security, robustness, ease-of-use and ad hoc networking abilities [10]. It able to transmit data in range of 10 m. Bluetooth also have typically high data rate, make it useful to transmit large file such as audio and video.

IrDA utilize infrared light to transmit data. It transmit low data rate operate in a small distance, which is less than 10 m. It is most commonly used in low data rate transmission such as remote control, computer, wireless USB and so on [11].

Wi-Fi is another wireless protocol. Wi-Fi provides secure, reliable and fast wireless connectivity. A Wi-Fi network can be used to connect electronic devices to each other, to the Internet, and to wired networks which use Ethernet technology. Wi-Fi networks operate in the 2.4 and 5 GHz radio bands, with some products that contain both bands. They can provide real-world performance similar to basic wired networks [12].

Among all these protocol, zigbee was chosen because it is the most suitable for this project. This project requires low data rate, low power consumption and long range data transmission. Figure 6 is the summary of the features of the wireless protocol.

Table 1: Summary of wireless protocol [11]

	ZigBee	802.11 (Wi-Fi)	Bluetooth	UWB (Ultra Wide Band)	Wireless USB	IR Wireless
Data Rate	20, 40, and 250 Kbits/s	11 & 54 Mbits/sec	1 Mbits/s	100-500 Mbits/s	62.5 Kbits/s	20-40 Kbits/s 115 Kbits/s 4 & 16 Mbits/s
Range	10-100 meters	50-100 meters	10 meters	<10 meters	10 meters	<10 meters (line of sight)
Networking Topology	Ad-hoc, peer to peer, star, or mesh	Point to hub	Ad-hoc, very small networks	Point to point	Point to point	Point to point
Operating Frequency	868 MHz (Europe) 900-928 MHz (NA), 2.4 GHz (worldwide)	2.4 and 5 GHz	2.4 GHz	3.1-10.6 GHz	2.4 GHz	800-900 nm
Complexity (Device and application impact)	Low	High	High	Medium	Low	Low
Power Consumption (Battery option and life)	Very low (low power is a design goal)	High	Medium	Low	Low	Low
Security	128 AES plus application layer security		64 and 128 bit encyption			
Other Information	Devices can join an existing network in under 30ms	Device connection requires 3-5 seconds	Device connection requires up to 10 seconds			
Typical Applications	Industrial control and monitoring, sensor networks, building automation, home control and automation, toys, games	Wireless LAN connectivity, broadband Internet access	Wireless connectivity between devices such as phones, PDA, laptops, headsets	Streaming video, home entertainment applications	PC peripheral connections	Remote controls, PC, PDA, phone, laptop links

CHAPTER 3

METHODOLOGY

3.1 Working Principle

Figure 6 shows the overall flow of the project. A pressure transducer will be attached to the pressure pipe of the firefighter to read the amount of pressure that the rescuer breathes. The reading then will converted into electrical analog signal and this signal will be fed as an input for microcontroller. The microcontroller will convert the signal into digital form, process it and pass it to XBee1 through UART module. XBee1 will transmit the data wirelessly to XBee2. The data from XBee2 will be sent to host PC via USB port and will be monitor by using a GUI that being developed using Microsoft Visual Basic 2010. When the reading of the pressure is low, the microcontroller will give logic 1, or 5V to the DC vibration motor to alert the firefighter. When the firefighter feels the vibration, it indicates that the amount of pressure is low. The firefighter should left the current place immediately. If he didn't come out, the standby firefighter will go and rescue that firefighter.
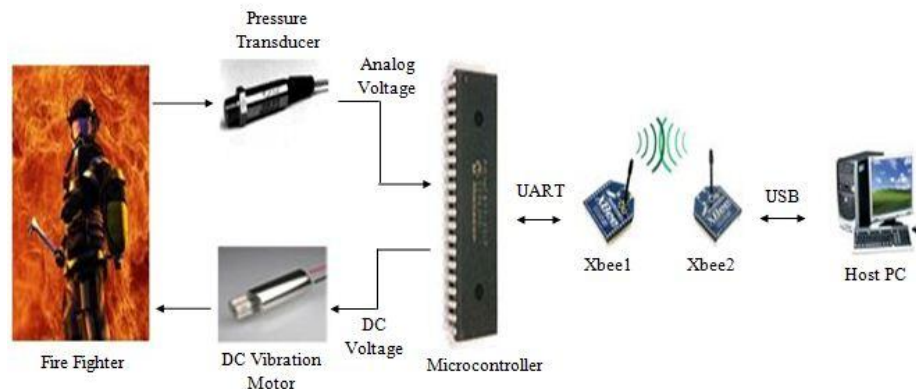

Figure 6: Flow of the project

Figure 7 shows the flow chart of the project. The project started with research and study on the title given so that the main objective is fully understood. Then, the searching and surveying for appropriate tools and equipment is started. With some equivalent devices already in hand, pre-prototype is being developed, generally to understand the basic working principle of the hardware and the coding for the software part. With this effort, any flaws and errors found during pre-prototype can be avoided during prototype development. When the prototype is working fine, it is ready for demonstration and exhibition.

```
                    ┌─────────┐
                    │  Start  │
                    └────┬────┘
                         ▼
              ┌────────────────────┐
              │ Research and survey│
              └─────────┬──────────┘
                        ▼
              ┌────────────────────┐
              │ Develop pre-prototype│◄─────────┐
              └─────────┬──────────┘            │
                        ▼                        │  Troubleshoot/Debug
                     ◇─────────◇                 │
            Yes   ◇  Is pre-prototype  ◇   No     │
              ┌──◇   working?         ◇───────────┘
              │     ◇─────────◇
              ▼
              ┌────────────────────┐
              │ Construct prototype│◄─────────┐
              └─────────┬──────────┘          │
                        ▼                      │  Troubleshoot/Debug
                     ◇─────────◇               │
            Yes   ◇  Is prototype   ◇   No      │
              ┌──◇   working?       ◇───────────┘
              │     ◇─────────◇
              ▼
              ┌────────────────────┐
              │   Demonstration    │
              └─────────┬──────────┘
                        ▼
                    ┌─────────┐
                    │   End   │
                    └─────────┘
```
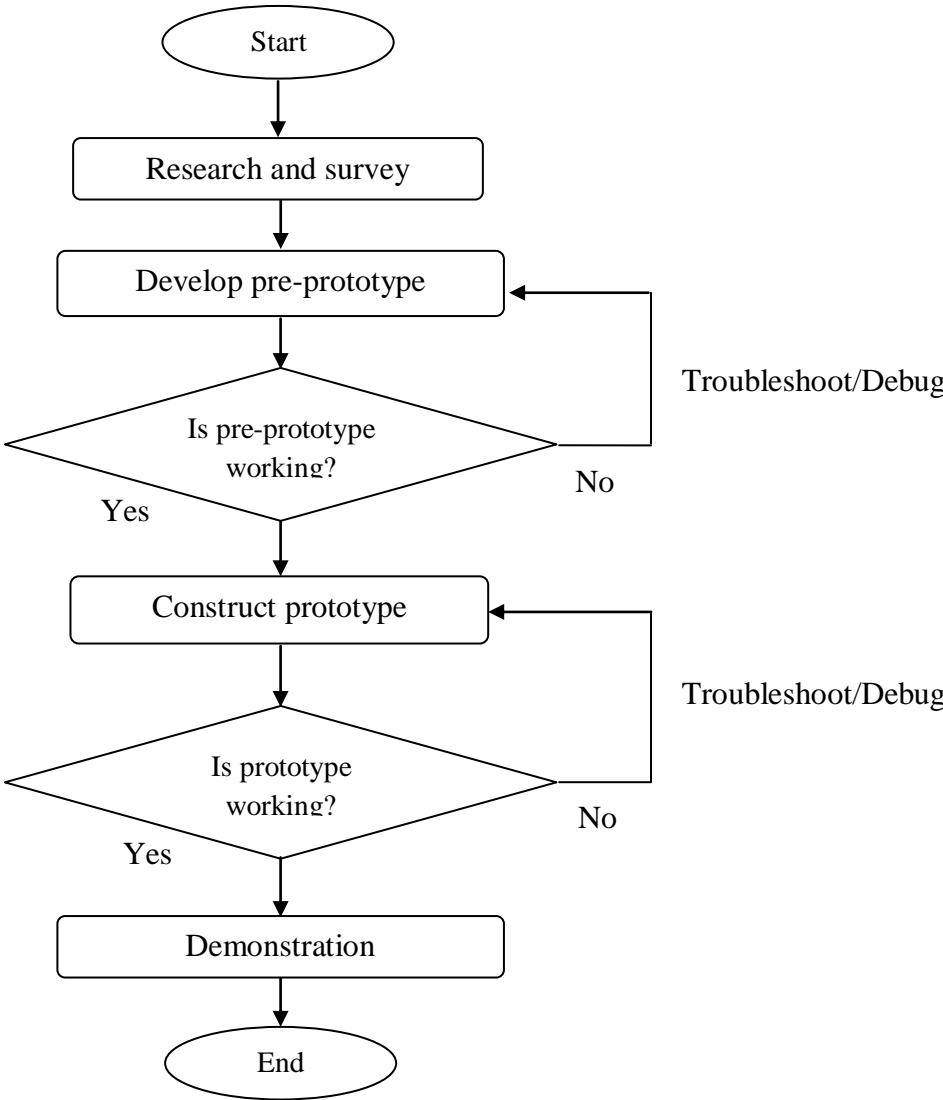
Figure 7: Project flow chart

3.2 Circuit Design

The circuit was designed by using CadSoft Eagle Professional. On the left side of the schematic shown in Figure 8 is the voltage regulator circuit for microcontroller and Xbee module, which is 5V and 3.3V respectively. Diode is added to protect the circuit from reverse polarity. Transducer will also be power up by 5V source. The function of the capacitors is to increase the stability and filter out the noises. Note that the chosen of power supply must not be too large because the efficiency of the regulator will decrease. Efficiency = V_{out}/V_{in}. The magnitude of the power supply also must not too close with V_{CC}, as there will be a voltage drop across the regulator just like a diode. Generally, power supply should typically 1 to 2V greater than the V_{CC}. 20 MHz crystal clock is responsible to provide clock signal for the microcontroller. Transistor is added as an electronic switch for vibration motor.

For Xbee module, minimum of 4 connections are required, which are 3.3V V_{CC}, GND, DIN and DOUT. V_{CC} and GND are for power supply, while DIN and DOUT is for transmitting and receiving the data.
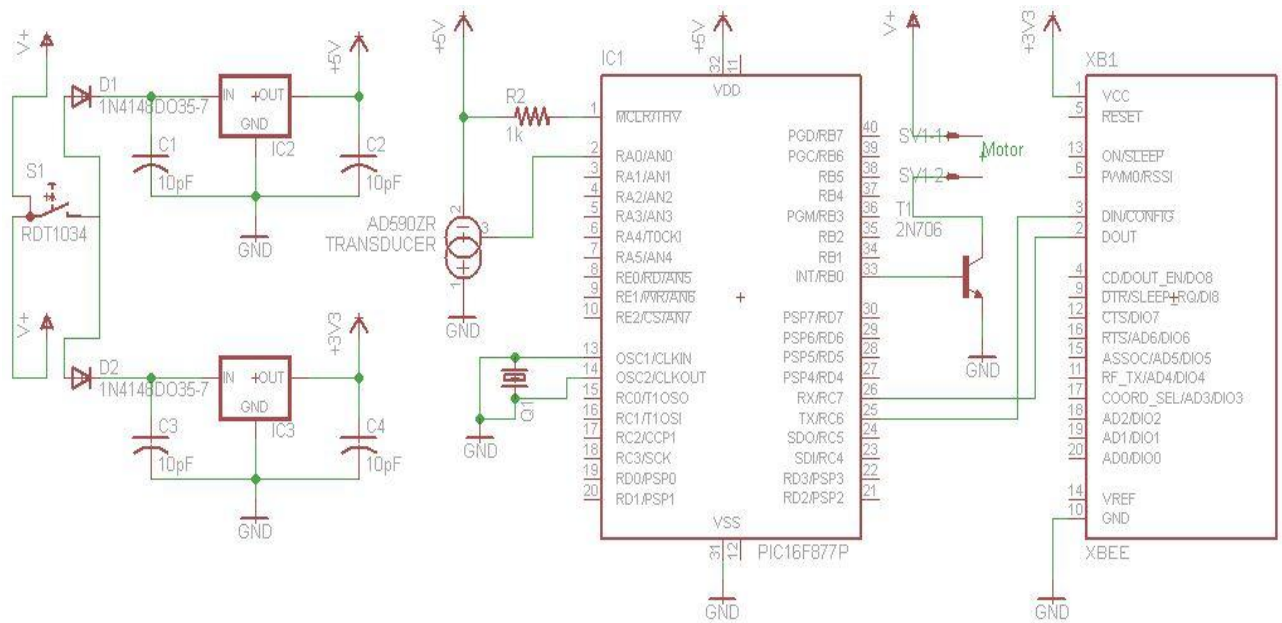

Figure 8: Circuit design

3.3 Tools and Equipment Required

3.3.1. Rescuer

Rescuer is a person who performs the rescue task. In this project, the rescuer may refer to the firefighter. The rescuer is divided into two types, which is the one who enter the building and the one who stay outside of the building, at the fire engine. Rescuer wears a SCBA to provide breathable air. The SCBA will be attached on the jacket. The pressure transducer will be attached on the same jacket, at the SCBA pipe. Data will be sent out from the rescuer inside the building to the fire engine outside the building. The rescuer at the fire engine will monitor the rescuer inside the building. When the pressure level is low, the rescuer inside the building will be alerted by DC motor vibrator and he need to leave the building as soon as possible. If he still does not come out, or maybe stuck inside the building, the rescue at the fire engine will go in and save him. This method may save the rescuer life.

3.3.2. XBee

XBee is a brand name from Digi International which utilizes zigbee protocol. It was created to meet IEEE 802.15.4 standards and support the unique need of low-cost, low power wireless sensor network. It consumes minimal power and provides reliable data transmission between devices. This module operates within ISM 2.4 GHz frequency band and is pin-for-pin compatible with each other [13]. Since this project is going to monitor only one rescuer, so two modules of XBees are going to be used and both of them act as a transceiver, become a transmitter and a receiver alternately.

3.3.3. DC Motor Vibrator

A DC motor vibrator is a device that responsible to produce a vibration by a given electric energy. The vibration will be generated by a motor when there is unbalance mass on its driveshaft. This vibrator will be the alerting device for the rescuer [14].

3.3.4. Pressure Transducer

Pressure transducer is a device that measure and convert pressure into electrical analog signal. The actual element consist of strain gauge, which is basically a resistive element whose resistance changes with the amount of strain apply on it. The conversion of pressure into an electrical signal is achieved by the physical deformation of strain gauge which are bonded into the diaphragm of the pressure transducer and wired into Wheatstone bridge configuration. The pressure that applied to the pressure transducer will produces a deflection of the diaphragm which introduces strain to the gauges. The strain will produce an electrical resistance change proportional to the pressure [15].

3.3.5. PIC16F877A Microcontroller

The microcontroller is a programmable chip that will produce desire result by processing the given input. The input in this case is an analog voltage from the transducer and the output will be 5V. The microcontroller will process and prepare the data which is going to be sent to XBee module via UART module. This microcontroller is developed by Microchip Technology [16].

3.3.6. MPLAB IDE

MPLAB IDE is free software developed by Microchip Technology used as platform to develop the coding for the microcontroller. It able to find the any error in the code, and then convert it into hex file that to be loaded into the microcontroller.

3.3.7. Microsoft Visual Basic 2010 Express

Microsoft Visual Basic 2010 Express is free software from Microsoft Corporation, used to create a GUI for this project. This allows the user to interact with electronic devices with images and graphical representation instead of text command. It also allows the user to interact with outside world via serial port of the computer. If the computer does not have any serial port, USB port can be emulated to become a virtual serial port. This software used VB as a programming language [17].

CHAPTER 4

RESULT AND DISCUSSION

4.1 Hardware Setup

Figure 9 shows the hardware setup for transmitter of this project. The circuit on the strip board is the one that going to be attached on the rescuer. The circuit is power up by using two high current 3.7V batteries that normally used for laptop and flashlight. The circuit is controlled by toggle switch. LED indicator indicates the power status for the circuit. Three pin female headers on the left of the microcontroller are used to connect the pressure transducer. For demonstration purpose, a 5kΩ potentiometer is used to show the variation of the signal. Another three pin female headers on the right are added to test the functionality of the microcontroller when mounted on the strip board. All the wires are soldered at the back of strip board.
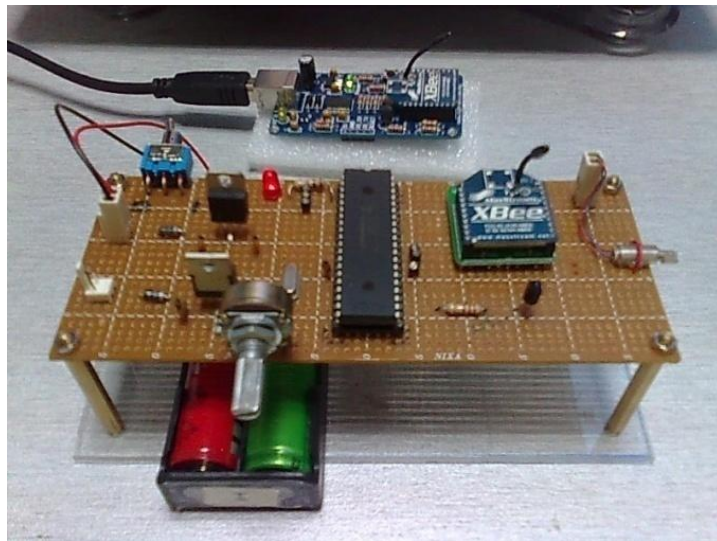
(a)
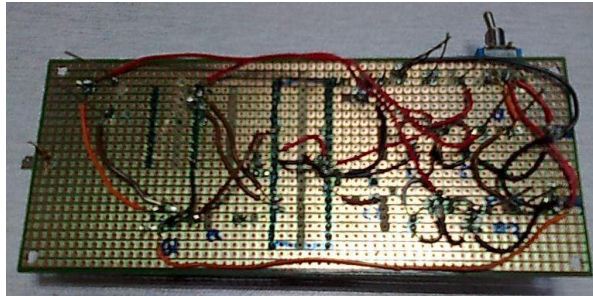
(b)

Figure 9 (a) and (b): Top and bottom view of the prototype

Figure 10 shows the hardware setup for receiver of this project. The board is called SKXBee from Cytron Technologies. The board consists of chip from FTDI, responsible for two tasks, which is performing voltage conversion between 5V and 3.3V and also signal conversion between USB and UART signal. The XBee module will be plugged onto the board, and then connect the board to the PC by using USB cable [18]. Red and yellow LED show the transmitting and receiving status respectively while green LED indicate the power status of the board. Reset button also available on the board. Certain board doesn't have reset button, so with this board, external reset button circuitry is unnecessary.

Figure 10: Circuit for receiver

4.2 Software Setup

4.2.1 FTDI Driver Installation

To establish communication between PC and SKXBee which consist of FTDI chip, the driver must be installed first [19]. Note that the steps are for Windows 7. The steps are as follows:

a) Go to the device manager

Control Panel >> Hardware and sound >> Device Manager

b) In the Device Manager window, there will be a device under Other Devices with warning symbol to indicate a problem. The text next to this device is depending on the device attached. In this case the device is TTL232R. Right click on the Other Device as shown in Figure 11.
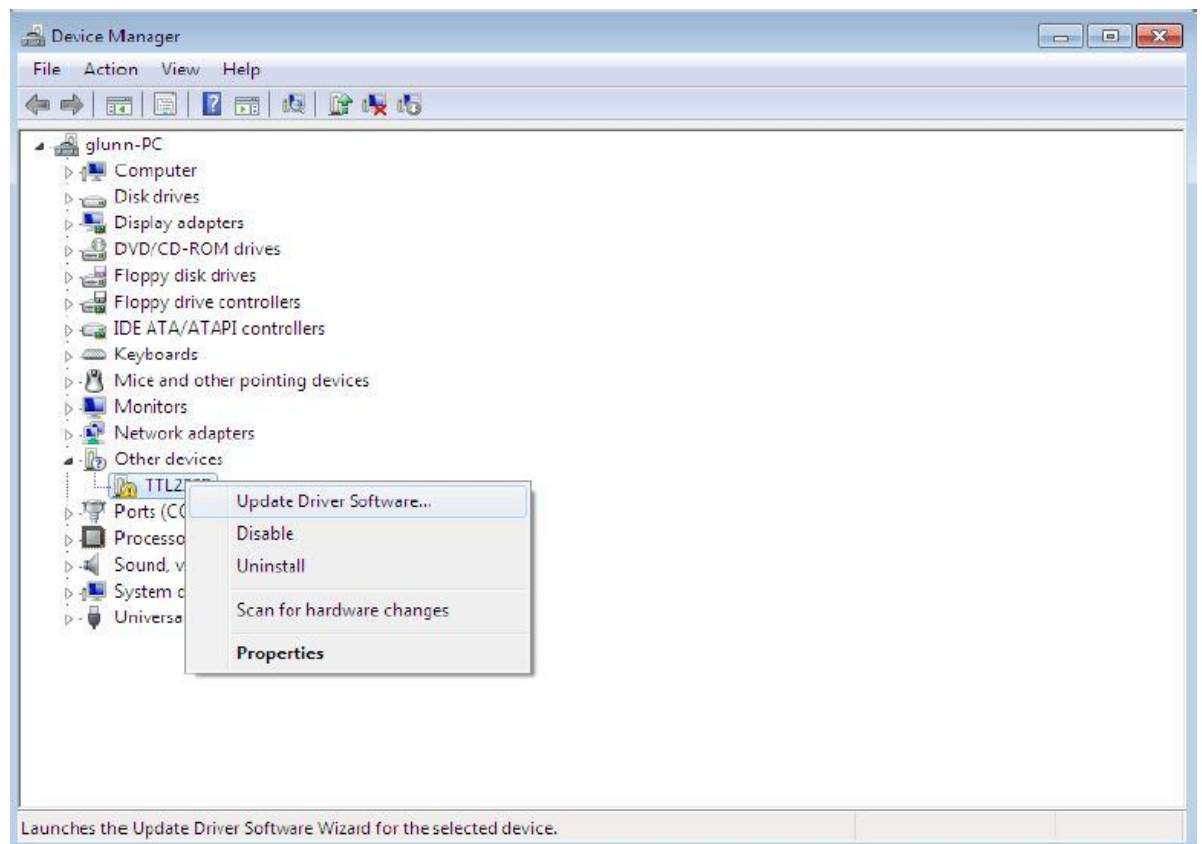
Figure 11: Updating driver software

c) Choose "Browse my computer for driver software". After browsing the driver, just click "Next" to start the installation

d) Wait until the installation is complete then go back to the Device Manager

e) The Device Manager will still show a device under Other Devices but in addition, there is a new entry under USB Controllers as shown in Figure 12. This indicates the bus layer of the driver is installed. The next step is to install Virtual Com Port layer. Just repeat step (b) until (d)
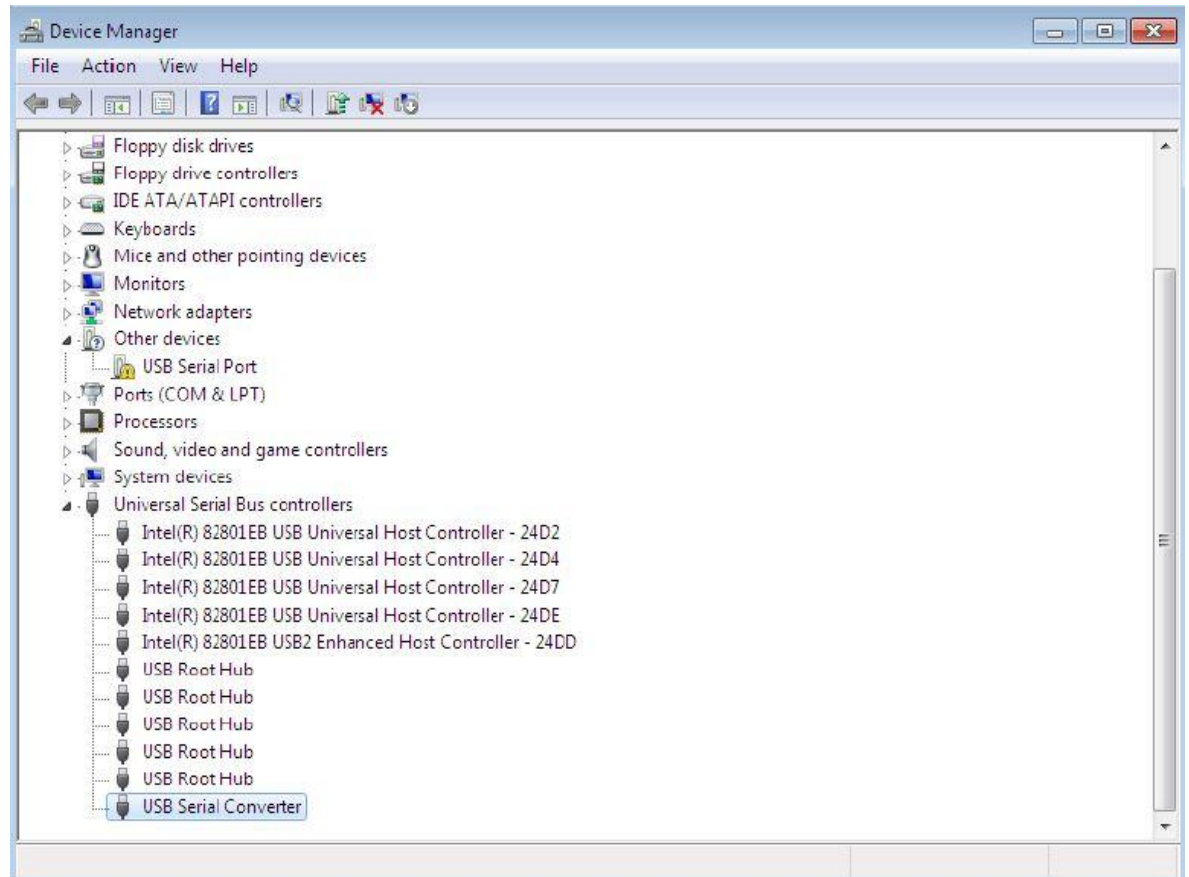
Figure 12: New entry of USB Controllers

f) Figure 13 shows the complete and correct installation of the driver. The device is now ready to be used on COM3.
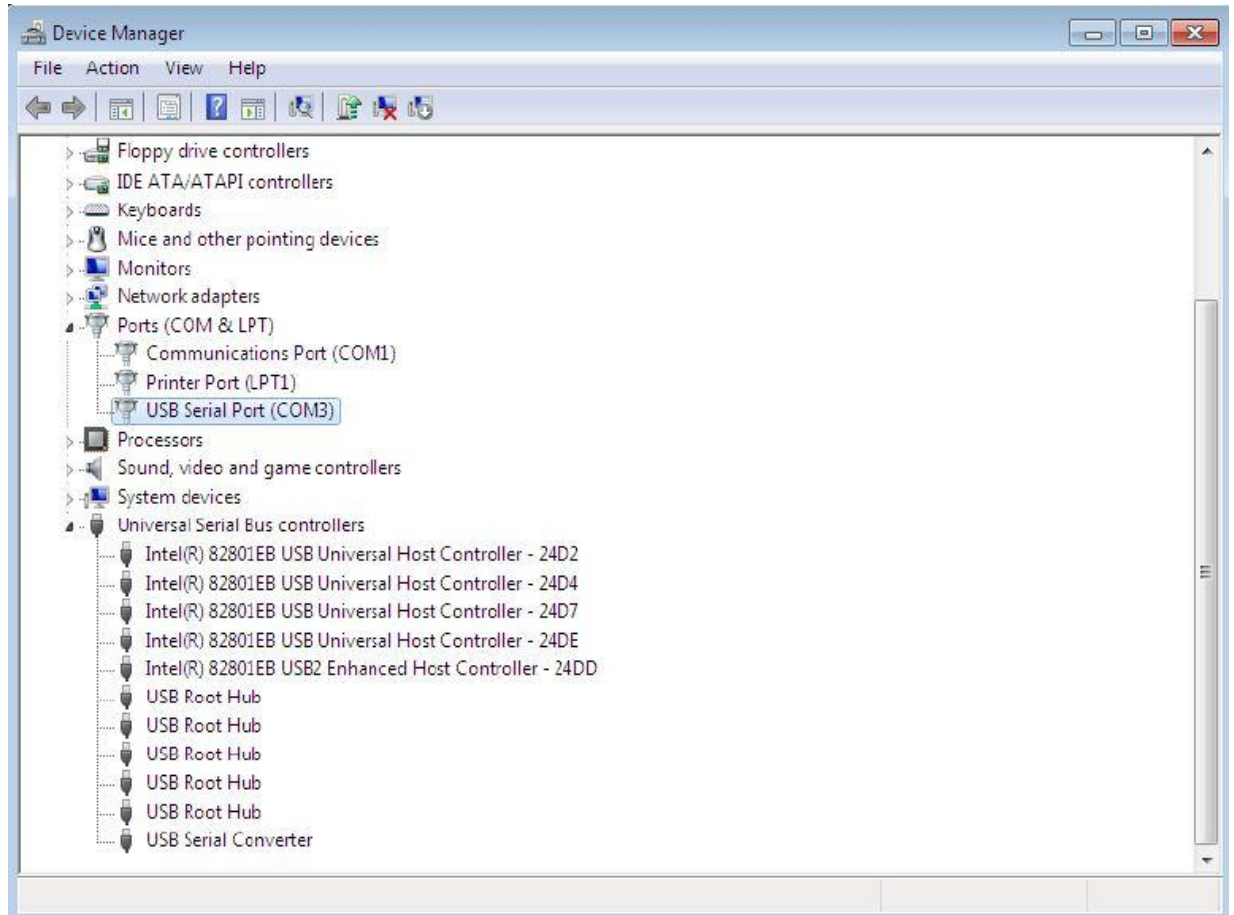
Figure 13: Complete installation of the driver

4.2.2 XBee Configuration by using X-CTU

  X-CTU is a software provided by Digi International, designed to enable a simple and user friendly GUI as well as to interact with the firmware files found on Digi's RF products [20]. Since XBee module doesn't need any programming, it needs to be configured so that each module recognizes each other. The steps are as follows:

a) Launch X-CTU, select COM PORT and then click "Test / Query"
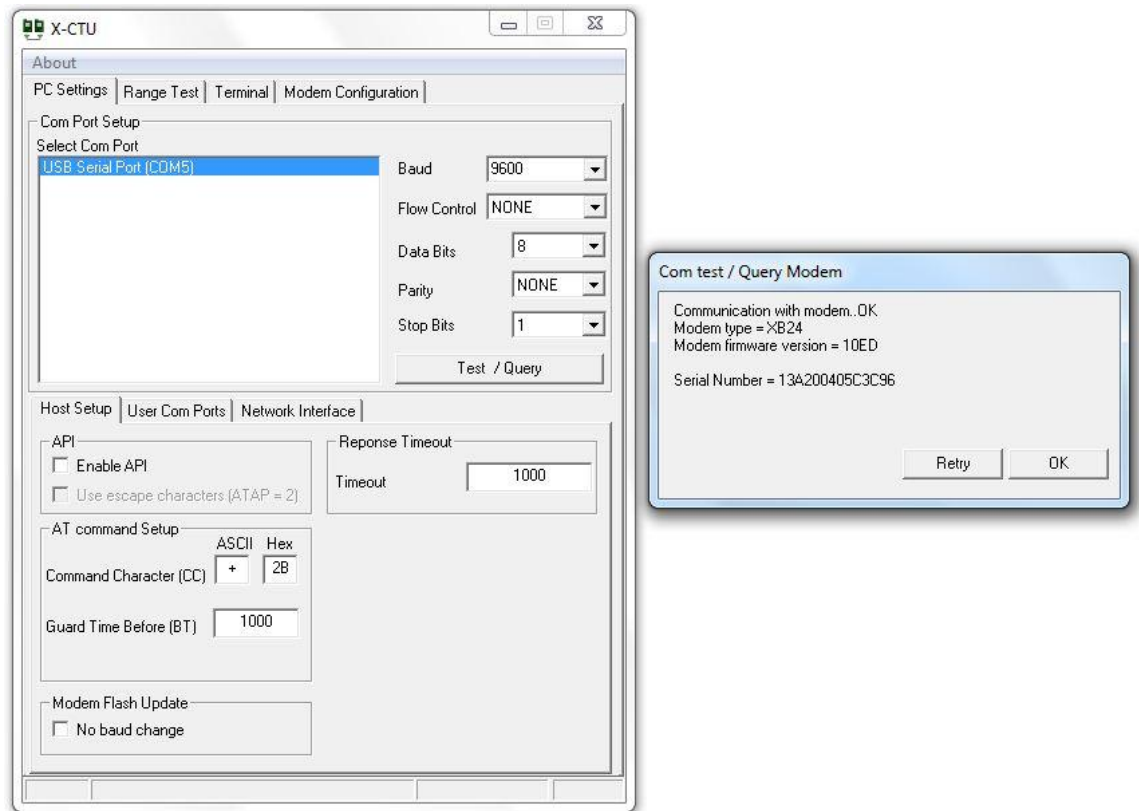
Figure 14: Launching X-CTU

b) Click "Modem Configuration" tab and then click "Read". The parameters of the XBee module will be displayed.

c) Under "Networking & Security", copy the Serial Number High and Serial Number Low, then paste it at the second XBee module. Also, copy the Serial Number High and Serial Number Low of the second XBee, paste at first XBee module. The purpose of this step is to ensure the both XBee modules recognize each other so that point to point communication can be establish.
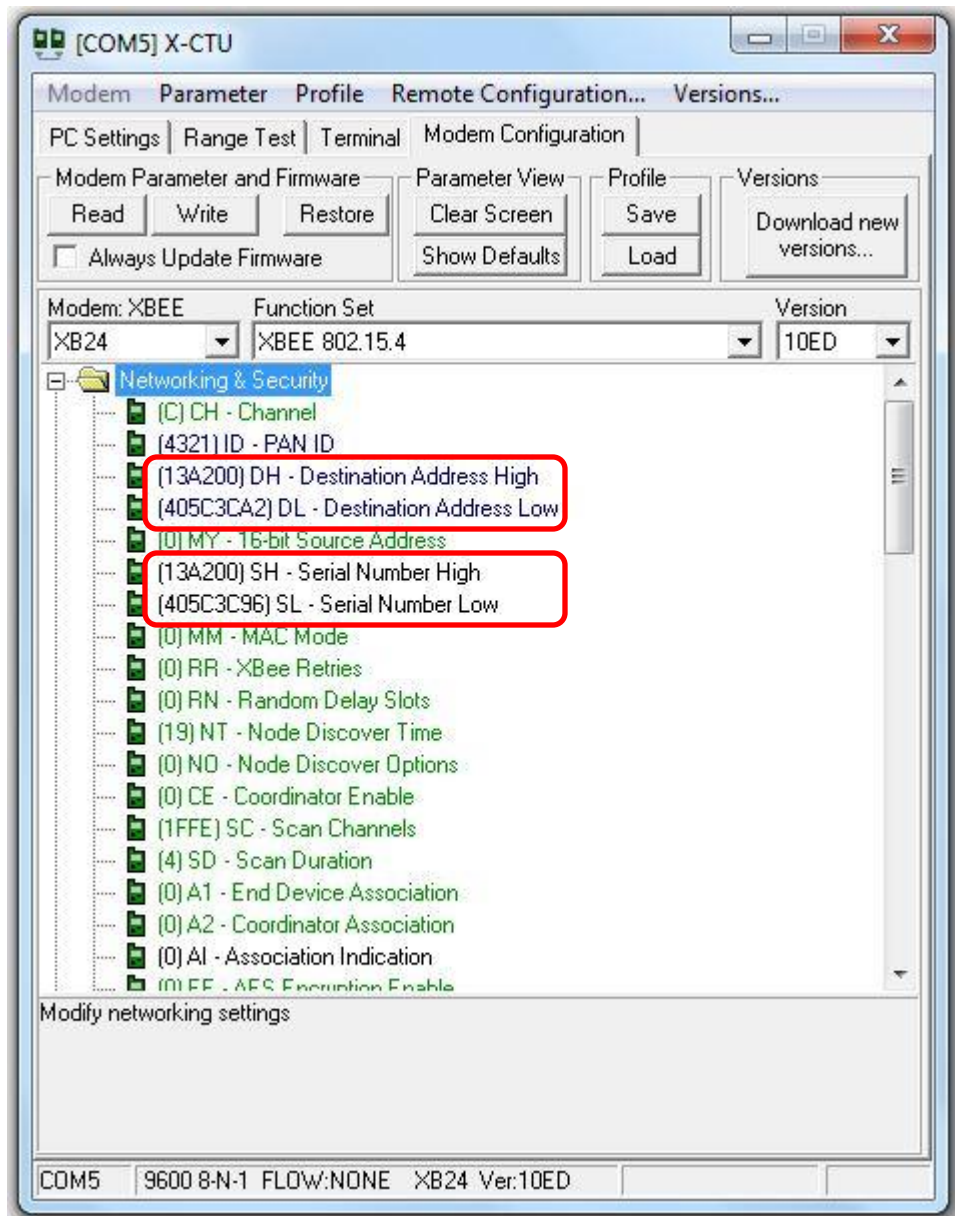
(b)

Figure 15: Steps to configure XBee module by using X-CTU

4.3 Prototype Testing

4.3.1 Functionality Test

Once the hardware and software had been setup, the system needs to be tested, to ensure that the system is working as desired especially at the transmitter circuit. The circuit at the receiver side is working fine. During first few testing, the transmitter circuit failed to function due to some mistake of connection at the strip board. The circuit had undergone several stages of troubleshooting before it able to function.

When the circuit is alive, the wireless transmission needs to test because this is the main part of the system. The circuit is successfully able to transmit the data wirelessly to the PC. The PC is able to read the data that being sent from the transmitter.
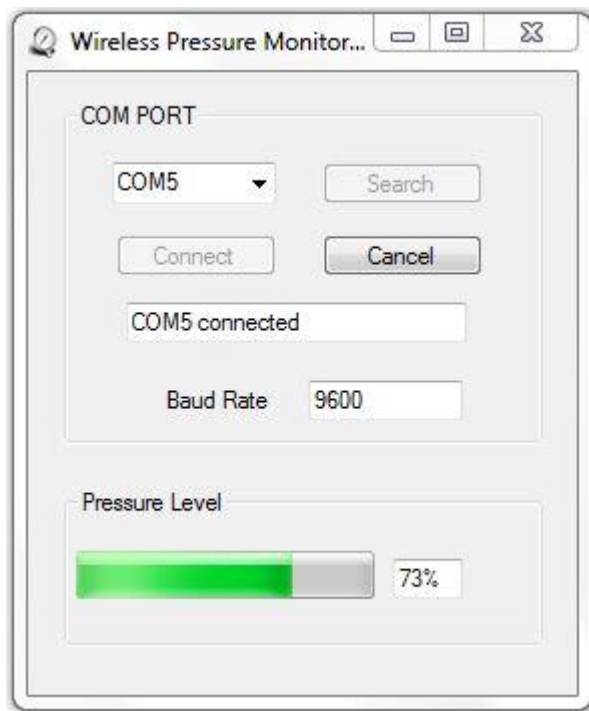
Figure 16: GUI of the system

4.3.2 Distance Test

Another crucial part of the testing is the distance testing. There are two types of distance test, as specify in the datasheet, namely indoor and outdoor range test. Normally the indoor range will be much lesser than outdoor range due to obstacles. Table 3 illustrates the difference between datasheet result and actual result.

Table 3: Distance test result

Type of Test	Datasheet	Actual
Indoor/Urban Range	Up to 30 m	Up to 15 m
Outdoor RF line-of-sight range	Up to 100 m	Up to 70 m

During indoor test, the receiver (PC) is place inside an opened room to have better signal transmission. Although it is not the worst case, but in actual situation the receiver is most likely to be place the fire engine and there will be an open space for sure. At the same time, the transmitter (strip board circuit) is moving away slowly from the PC and the data receive is being monitored at the PC accordingly.

The receiver read the data perfectly when the distance is below 10 m. As the transmitter go beyond 10 m, the signal strength gradually decrease and the receiver response slower than before. When the transmitter goes beyond 15 m, the receiver completely failed to read the data.

The test is repeated at outdoor. This time the range is extended than indoor test due to fewer obstacles. The receiver read well the data when the distance below 60 m. Again, the signal strength falls gradually and the response of the receiver is slower. When the transmitter goes beyond 70 m, the receiver completely failed to read the data.

4.4 Cost Analysis

Table 4 shows the price of each individual item and total cost of the project. In term of the availability, all equipment except pressure transducer is available in local market. Initial estimation of total cost is RM712. After intensive survey, there is some equipment that is cheaper than estimation, so the total cost is reduced by 10.19 % to RM639.45.

Table 4: Summary of budget

Item	Price (RM)	Quantity	Subtotal
XBee modules	112	2	224
SKXBee	65	1	65
Pressure transducer	295.40	1	295.40
DC vibration Motor	5	1	5
Electronic Component	44.05	Abundance	44.05
TOTAL			**633.45**

XBee module used in this project is "XBee 1mW Wire Antenna Series 1", in which only allows point-to-point communication. This type of module is the cheapest thus it provide the shortest communication range compare to other type. To extend the range, "XBee Pro 60mW Wire Antenna Series 1" may be chose and it will cost 71.42 % more expensive, which is RM192.

Series 1 module only support for point-to-point communication. To establish mesh network so that multiple rescuer can be monitored, Series 2 shall be chosen. The product is keep on update in improvise by the manufacturer from time to time. The type of module chose is depending on the application.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

This wireless monitoring system is capable of reducing the risk of injury and death of the rescuer, hence able to improve the overall rescuing activities. As mention before, in order to save the victims, the rescuer himself must be save first. When the rescuer is save, the probability to save the victim is very high, so losses due to injury and fire could be reduced.

5.2 Recommendation

Several actions can be taken in order to improve the performance of the system such as:

a) Substitute XBee with XBee Pro. XBee Pro is the improvise version of XBee with greater indoor and outdoor range, as well as higher transmit power output [13]. However, XBee Pro is slightly expensive than ordinary XBee module.

b) Expand the system to monitor more than 1 rescuer. This can be done by establishing mesh wireless network. XBee module will be attached to each of the rescuer. A systematic database also needs to be created in order to store all the information of each rescuer.

REFERENCES

[1]   "Jabatan Bomba Dan Penyelamat Logo". [Online] Retrieved February 18, 2012, from http://www.bomba.gov.my/

[2]   "Para-SAR". [Online] Retrieved February 18, 2012, from http://en.wikipedia.org/wiki/Para-SAR

[3]   "Volunteer Fire Fighter Dies During Attempted Rescue of Utility Worker from a Confined Space". [Online] Retrieved February 18, 2012, from http://www.cdc.gov/niosh/fire/reports/face201031.html

[4]   "Rescue Squad". [Online] Retrieved February 18, 2012, from http://en.wikipedia.org/wiki/Rescue_squad

[5]   "Understanding Zigbee". [Online] Retrieved February 18, 2012, from http://www.zigbee.org/About/UnderstandingZigBee.aspx

[6]   Bryner N., Madrzykowski D., Stroup D., "Performance of Thermal Exposure Sensor in Personal Alert Safety System (PASS) Devices", 2005

[7]   Emergency Response Breathing Apparatus, NASA Spinoff, 2000

[8]   "Slim air pack cools firefighters, aims to replace SCBA". [Online] Retrieved February 28, 2012, from http://www.firerescue1.com/fire-products/fire breathingapparatus/articles/600098-Slim-air-pack-cools-firefighters-aims   to-replace-SCBA/

[9]   "Getting to Know Wireless Networks and Technology". [Online] Retrieved February 28, 2012, from http://www.informit.com/articles/printerfriendly.aspx?p=98132

[10]   "The Essentials of the Bluetooth SIG". [Online] Retrieved February 28,   2012, from https://www.bluetooth.org/About/bluetooth_sig.htm

[11]   "How does ZigBee compare with other wireless standards?" [Online] Retrieved February 28, 2012, from http://www.stg.com/wireless/ZigBee_comp.html

[12]   "Discover and Learn". [Online] Retrieved February 28, 2012 from http://www.wi-fi.org/discover-and-learn

[13]   Maxstream, "XBee™/XBee-PRO™ OEM RF Module", 2006

[14]   "Vibrator (mechanical)". [Online] Retrieved March 2, 2012, from http://en.wikipedia.org/wiki/Vibrator_%28mechanical%29

[15]   "Introduction to Pressure Transducer". [Online] Retrieved February 28, 2012, from http://www.omega.com/prodinfo/pressuretransducers.html

[16]   "PIC16F877A". [Online] Retrieved March 2, 2012, from http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en010242

[17]   "Visual Basic 2010 Express". [Online] Retrieved March 5, 2012, from http://www.microsoft.com/visualstudio/en-us/products/2010-editions/visual basic-express

[18]   "XBee Starter Kit Without Module". [Online] Retrieved March 5, 2012, from, http://www.cytron.com.my/viewProduct.php?pcode=SKXBEEBOARD &name=XBee20Starter%20Kit%20without%20module

[19]   FTDI Chip, "FTDI Drivers Installation Guide for Windows 7", 2009

[20]   Digi International, "X-CTU Configuration & Test Utility Software", 2008

Project Gantt chart

Activity/Week for FYP I	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Research and study	■	■												
Equipment Survey			■	■	■	■								
Extended proposal						■								
Developing pre-prototype							■	■	■					
Proposal defense									■					
Continue develop pre-prototype										■	■	■	■	■
Draft report													■	
Final report														■

Activity/Week for FYP II	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Develop, troubleshoot & testing	■	■	■	■	■	■	■	■							
Finalize prototype									■	■	■				
Pre-Sedex											■	■			
Draft report													■		
Final report & Technical Paper														■	
VIVA															■

APPENDIX B

C Coding for Microcontroller

```
//=============================================================
//      Author              : Norhusaidil Bin Hussain
//      Project             : VB-XBee Communication
//      Project description : Sending Analog Value To Visual Basic 2010
//=============================================================


//      include & config
//=============================================================
#include <pic.h>                 // header file for pic
__CONFIG ( 0x3F32 );             //configuration for the  microcontroller


//      define
//=============================================================
#define motor             RB1
#define led1              RD7
#define led2              RD6


//      function prototype      (every function must have a function prototype)
//=============================================================
void delay(unsigned long i);
void uart_init(void);
unsigned char uart_rec(void);
void uart_send(unsigned char data);
int read_a2d(unsigned char channel);
void set_digital(void);
//=============================================================


//      main function                           (main fucntion of the program)
//=============================================================
void main(void)
{
        int i,adc;
        TRISA = 0b00000001;                     // configure PORTA I/O direction
        TRISB = 0b00000000;                     // configure PORTB I/O direction
        TRISC = 0b10000000;                     // configure PORTC I/O direction
        TRISD = 0b00000000;                     // configure PORTD I/O direction
        PORTA = 0x00;                           // clear PORTA
```

```
        PORTB = 0x00;                                   // clear PORTB
        PORTC = 0x00;                                   // clear PORTC
        PORTD = 0x00;                                   // clear PORTD
        set_digital();
        INTCON   = 0b11100000;
        PIE1     = 0b00100000;

        led1 = 0;                                       // initialize led1
        led2 = 0;                                       // initialize led2
        motor = 0;                                      // initialize motor
        uart_init();                                    // initialize UART module

        while(1)                                        // infinite loop
        {
                adc = read_a2d(0);
                led1 = adc;
                uart_send(adc);                         // sending analog value to VB

                if(adc<=10)                             // if ADC value less than 10%
                {
                        motor = 1;                      // turn ON the motor
                }
                else                                    // if ADC value greater than 10%
                {
                        motor = 0;                      // turn OFF motor
                }
        }

}

//      functions
//=============================================================================
void delay(unsigned long i)
{
        for (; i>0; i-=1);
}

void uart_init(void)
{
        SPBRG=129;                                      // set baud rate as 9600 baud
        BRGH=1;                                         // baud rate high speed option
```

```
        SYNC=0;                                                 // enable asychrounous mode
        TXEN=1;                                                 // enable transmission
        TX9 =0;                                                 // 8-bit transmission
        RX9 =0;                                                 // 8-bit reception
        CREN1;                                                  // enable reception
        SPEN=1;                                                 // disable serial port
}


unsigned char uart_rec(void)                                    // receive uart value
{
        unsigned int rec_data;
        while(RCIF==0);                                         // wait for data
        rec_data = RCREG;
        return rec_data;                                        // return the data received
}
void uart_send(unsigned char data)
{
        while(TXIF==0);                                         // only send the new data after
        TXREG=data;                                             // the previous data finish sent
}


int read_a2d(unsigned char channel)             // function for ADC module
{
ADCON0=0b00000001;
ADCON1=0b10000000;
ADCON0=(ADCON0&0xC7)|(channel<<3);
delay(5);
ADGO=1;
while(ADGO==1) continue;
GODONE=1;
while(GODONE==1) continue;
return((256*ADRESH+ADRESL)/10.2);
}


void set_digital(void)
{
ADCON1=0b00000110;
}
```

APPENDIX C

VB Coding for GUI

```vb
Public Class Form1

    Dim transmitt As Integer
    Dim rec_data As String

    'define com port name
    Dim WithEvents ComPort As New IO.Ports.SerialPort
    Private Delegate Sub AsyncReceiver(ByVal sender As Object, ByVal e As System.IO.Ports.SerialPort)

    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        For i As Integer = 0 To My.Computer.Ports.SerialPortNames.Count - 1
            ComPortBox.Items.Add( _
            My.Computer.Ports.SerialPortNames(i))
        Next
        Cancel.Enabled = False
        Timer1.Interval = 10
    End Sub

    Private Sub Form1_FormClosed(ByVal sender As Object, ByVal e As
System.Windows.Forms.FormClosedEventArgs) Handles Me.FormClosed
        Try
            ComPort.Close()
            transmitt = 0
        Catch ex As Exception
        End Try
    End Sub
    Private Sub Connect_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
Connect.Click
        Search.Enabled = False
        If ComPort.IsOpen Then
            ComPort.Close()
        End If
        Try
            With ComPort
                .PortName = ComPortBox.Text
                .BaudRate = 9600 '115200
                .Parity = IO.Ports.Parity.None
```

```vbnet
        .DataBits = 8
         .StopBits = IO.Ports.StopBits.One
         ' .Encoding = System.Text.Encoding.Unicode
      End With
      baudrate_status.Text = ComPort.BaudRate


      ComPort.Open()
      comport_status.Text = ComPortBox.Text & " connected"
      Connect.Enabled = False
      Cancel.Enabled = True
      transmitt = 1
      Cancel.Enabled = True
      Timer1.Start()
   Catch ex As Exception
      'MsgBox(ex.ToString)
      MsgBox("Please select COM PORTS")
   End Try
End Sub



Private Sub Cancel_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Cancel.Click
   Search.Enabled = True
   Try
      ComPort.Close()
      comport_status.Text = ComPort.PortName & " disconnected"
      Connect.Enabled = True
      Cancel.Enabled = False
      transmitt = 0
   Catch ex As Exception
      'MsgBox(ex.ToString)
      comport_status.Text = ComPort.PortName & " disconnected"
      Connect.Enabled = True
      Cancel.Enabled = False
      Cancel.Enabled = False
      Timer1.Stop()
   End Try
End Sub
```

```vbnet
    Private Sub Search_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Search.Click
        ComPortBox.Items.Clear()
        For i As Integer = 0 To _
          My.Computer.Ports.SerialPortNames.Count - 1
            ComPortBox.Items.Add( _
                My.Computer.Ports.SerialPortNames(i))
        Next
        Cancel.Enabled = False
    End Sub


    Private Sub Timer1_Tick(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Timer1.Tick
        Try
            If ComPort.IsOpen = True Then
                ComPort.ReadTimeout = 1000
                Dim x As Integer = ComPort.ReadByte
                ProgressBar1.Value = x
                pb_value.Text = x & "%"
                ComPort.Close()
            Else : ComPort.Open()
            End If


            If ProgressBar1.Value = 10 Then
                MessageBox.Show(" The pressure level is low ", "Warning", MessageBoxButtons.OK,
MessageBoxIcon.Warning)
            End If


        Catch ex As TimeoutException
            MessageBox.Show(" Signal loss! ", "Warning", MessageBoxButtons.OK, MessageBoxIcon.Warning)
            ComPort.Close()
            Timer1.Stop()
        End Try
    End Sub
End Class
```