UNIVERSITI TEKNIKAL MALAYSIA MELAKA

FACULTY OF ELECTRICAL ENGINEERING

FINAL YEAR PROJECT 11

A SMART HOME SECURITY SYSTEM BASED ON ZIGBEE

NAME	NUR FATIHAH BNTI MOHD MARZUKI
MATRIX NUMBER	B011010163
COURSE	BACHELOR OF ELECTRICAL ENGINEERING (CONTROL,INSTRUMENTATION,AUTOMATION)
YEAR	4 BEKC
SUPERVISOR	PROF. MOHD ARIFF B. MAT HANAFIAH

A SMART HOME SECURITY SYSTEM BASED ON ZIGBEE

Nur Fatihah Binti Mohd Marzuki

Bachelor of Electrical Engineering

(Control, Instrumentation & Automation)

June 2014

SUPERVISOR DECLARATION

" I hereby declare that I have read through this report entitle " Home Security System using Zigbee" and found that it has comply the partial fulfilment for awarding the degree of Bachelor of Electrical Engineering (Control, Instrumentation and Automation)"

Signature :

Supervisor's Name : PROF. MADYA MOHD ARIFF B. MAT HANAFIAH

Date : 17/6/2014

A SMART HOME SECURITY SYSTEM BASED ON ZIGBEE

NUR FATIHAH BINTI MOHD MARZUKI

A report submitted in partial fulfilment of the requirements for the degree of Bachelor in Electrical Engineering (Control, Instrumentation and Automation)

Faculty of Electrical Engineering
UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2014

STUDENT DECLARATION

I declare that this report entitle "*A SMART HOME SECURITY SYSTEM BASED ON ZIGBEE*" is the result of my own research except as cited in the references. The report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature : ....

Name : NUR FATIHAH BINTI MOHD MARZUKI

Date : 17/6/2014

To my beloved father and mother who always there for me

MohdMarzuki Bin Ibrahim and KarimahBintiYaakob

To my Supervisor and lecturer, for their guidance and encouragement

Prof MohdAriff Bin Mat Hanafiah

To my friends, for their unconditional support

Dedicated in thankful appreciation for your supporting, encouragement and best wishes.

ACKNOWLEDGEMENT

In the Name of ALLAH, the Most Gracious and the Most Merciful
Praise be to ALLAH

First and foremost, I would like to take this opportunity to express my deepest grateful appreciation to all wonderful people have continuously giving me support, advices, knowledge, understanding and contribution towards the successful completion of this Final Year Project.

In particular, I wish to express my sincere appreciation to my supervisor, Professor MohdAriff Bin Mat Hanafiah for encouragement, guidance, critics, advices, suggestion and motivation on developing this project. I also wish to express my sincere appreciation to, Wan NurShelaEzwane Wan Jusohwho is willing to spend his precious time to give some ideas and suggestion towards this project. This thesis would not have been the same as presented here without continued support and interest from them. They have contributed towards my understanding and thoughts.

My sincere appreciation extends to all my undergraduate friends especially BEKC member of batch 2010-2014 who have helped and shared brilliant ideas throughout the whole year.

Last but not least, I would like to express my sincerest gratitude and deepest thankfulness to my parent MohdMarzuki Bin Ibrahim and KarimahBintiYaakob for their love, support and encouragement that they had given to me.

ABSTRACT

The purpose of this project is to design and develop a security system for monitoring any intruders and others emergency situation around the house and display to the user in the form of computer monitor display. The chosen sensor is PIR motion sensor and the semiconductor sensor. The PIR can be used to detect movements, normally used to detect human movement when passing in or out of range sensor The gas sensor used in this project is MQ2, the semiconductor gas sensor which detects the presence of combustible gas and smoke at concentrations from 300 to 10,000 ppm. These sensors will detect the concentration of the gas according the voltage output of the sensor. To make the sensors operate in the alarm system and data monitoring system, Arduino Uno was used as the microcontroller for the whole system. The circuit also includes LEDs, buzzer, exhaust fan and Xbee. Xbee will send the data reading from gas sensor to data monitoring system that display on computer monitor display.

ABSTRAK

Projek ini bertujuan untuk mereka bentuk dan membangunkan sistem keselamatan dan memantau penceroboh dan keadaan kecemasan di sekitar rumah dan memapaparkan kepada pengguna dalam bentuk komputer monitor. Sensor dipilih adalah PIR sensor gerakan dan sensor semikonduktor. Passive Inframerah PIR boleh digunakan untuk mengesan pergerakan, biasanya digunakan untuk mengesan pergerakan manusia ketika melalui atau di luar julat sensor. Sensor gas yang digunakan dalam projek ini adalah MQ2 , sensor gas semikonduktor yang mengesan kehadiran gas yang mudah terbakar dan asap pada kepekatan daripada 300 kepada 10,000 ppm. Sensor ini akan mengesan kepekatan gas mengikut output voltan sensor. Sensor beroperasi dalam sistem penggera dan sistem pemantauan data, Arduino Uno telah digunakan sebagai pengawal mikro untuk keseluruhan sistem. Litar ini juga termasuk LED, buzzer, kipas ekzos dan XBee . XBee akan menghantar membaca data dari sensor gas dan sensor gerakan untuk sistem pemantauan data yang dipaparkan pada monitor komputer paparan

TABLES OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

PIR	-	Pyroelectric Infrared
WSN	-	Wireless Sensor Network
LED	-	Light Emitting Diode
PPM	-	Parts Per Million
WPAN	-	Wireless Personal Area Network

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Project Background

Home security is a worldwide concerned issue, always emphasized and enhanced system. Different ways of security system have been enhanced such as use a large number of security officers, the use of sophisticated weapons, the use of alarms, monitoring system, through the production of electronic hardware and software and much more. One of the most important safety system and required for all social group is home security. Houses need to be monitored at all times such as from theft, fire and short circuits. Recently, the rate of crimes involving robbery, murder and fires is increasing and worrying all of us. So, home surveillance system must be upgraded to be more effective to keep up with the increasing crime rate. Various methods can be done to improve home security monitoring including the usage of security officers. However, this method is not suitable for all levels, wasteful and less reliability. All these improvements need to work more effectively, giving advantages to the user and can monitor without any errors that may hinder the security process. At present, a lot of study on smart home systems has been done and it covers all aspects. For example, smart home systems study in terms of multimedia, security monitoring, lighting, temperature control and others. In a smart home system, manual methods are no longer used and replaced by an automated system that helped users to monitor the condition of the house, thus facilitating and speeding up daily works. Automatic system can prevent the effects of human error and saving electricity.

1.2 Problem Statement

Security monitoring system requires data transmission system fast receiving data and accurate at a certain distance, so that users can place devices freely at important locations for the data display receiver. In other words, this system must be portable and user friendly. Display system is shown to be straightforward and easy to understand, so that users can take important immediate action. The system must not be hacked by anyone, no matter in various ways including on input source power, the content of data transmission, content of receiving data and location of security sensor device's main processor is stored [2]. The system must also have characteristics such as water resistant, high temperature resistant and robust, so that data transmission process and data receiving will not fail. Many of other security systems have some limitations on the usage of sensor devices. These problems will result in limitations of the security system. However, it is inevitable that a security system requires extensive use of sensors for the system to area of the house. The use of sensor devices is also very important in security systems. Sensors must be sensitive to human motion. Sensors must be working on the most appropriate range, that is not too close and too distant to detect movement and should be according to the human nature. [2]

1.3 Objectives

The main objective of this project are design and develop a security system for monitoring any intruders and others emergency situation around the house and display to the user in the form of computer monitor display.

a) To design and develop the home security system based on Zigbee technology
b) To develop the hardware and the programming the system

1.4 Scope of Research

In effort of achieving the objectives, several scopes have been outlined. The scopes of work in this project are:

a) To study the Zigbee Technology for home networking

b) To investigate the hardware and software appropriate with the Zigbee home automation.

c) To investigate and study the appropriate application software

1.5 Report Outlines

There are five chapters in this thesis which are introduction, literature review, methodology, result and discussion and finally conclusion and recommendation. Each chapter will discuss its own aspects related to the project.

Chapter one is the introduction for the project. Problem statement, object and scope of the project along with the summary of works have been discussed in this chapter. Then, chapter two discusses more on the theory and literature. Besides that, this chapter also discusses the type of Arduino used for the project, the sensor chosen, and also the software involve in programming the Xbee and Arduino.

Chapter three focuses on the methodology and approaches on the project. This includes the programming the software and hardware development of the project. Results and discussion are presented in chapter four. Lastly, chapter five is the conclusion for the whole project. Some future suggestions such as a functional addition and hardware improvement the project are also mentioned.

CHAPTER 2:

LITERATURE REVIEW

2.1 Introduction

This chapter will discuss in details on the components and instruments used for this project. Besides that, there are couple more of past related project or paper work that is related to the project.

2.2 Overview of Zigbee

ZigBee is a specification for wireless personal area networks protocol (WPANs) operating at 868 MHz, 915 MHz and 2.4 GHz. A WPAN is a personal area network in which the device in which the connections are wireless and a network for interconnecting an individual's devices. Using ZigBee, devices in a WPAN can communicate at speeds of up to 250 Kbps while physically separated by distances of up to 100 meters in typical circumstances and greater distances in an ideal environment. ZigBee is based on the 802.15.4 specification approved by the Institute of Electrical and Electronics Engineers Standards Association (IEEE-SA) [1].

ZigBee provides for high data throughput in applications where the duty cycle is low. This makes ZigBee ideal for home, business and industrial automation where control devices and sensors are commonly used. Such devices operate at low power levels, and this, in conjunction with their low duty cycle (typically 0.1% or less), translates into long battery life. Applications well suited to ZigBee include heating, ventilation and air conditioning (HVAC), lighting systems, fire sensing and the detection, intrusion detection and notification of unusual occurrences. ZigBee is compatible with most topologies including peer-to-peer, star network and mesh networks.

2.2.1 ZigBee Alliance

The ZigBee Alliance is an association of companies working together to define an open global standard for making low-power wireless networks. The objective of ZigBee Alliance is to create a specification defining how to build different network topologies with data security features and interoperable application profiles. The association includes companies from a wide spectrum of categories, from chip manufactures to system integration companies.

The first specification was ratified in 2004 and the first generation of ZigBee products had reached the market in 2005. A big challenge for the alliance is to make the interoperability to work among different products. To solve this problem, the ZigBee Alliance has defined different profiles, depending on what type of category the product belongs to [3].

2.2.2 History of ZigBee

1) ZigBee-style networks began to be conceived about 1998, when many installers realized that both WiFi and Bluetooth were going to be unsuitable for many applications. In particular, many engineers saw a need for self-organizing ad-hoc digital radio networks.

2) The IEEE 802.15.4 standard was completed in May 2003.

3) In the summer of 2003, Philips Semiconductors, a major mesh network supporter, ceased the investment. Philips Lighting has, however, continued Philips' participation and Philips remains a promoter member on the ZigBee Alliance Board of Directors.

4) The ZigBee Alliance announced in October 2004 that the membership had more than doubled in the preceding year and had grown to more than 100 member companies, in 22 countries. By April 2005 membership had grown to more than 150 companies and by December 2005 membership had passed 200 companies.

5) The ZigBee specifications were ratified on 14 December 2004.

6) The ZigBee Alliance announces public availability of Specification 1.0 on 13 June 2005, known as ZigBee 2004 Specification.

7) The ZigBee Alliance announces the completion and immediate member availability of the enhanced version of the ZigBee Standard in September 2006, known as ZigBee 2006 Specification.

8) During the last quarter of 2007, ZigBee PRO, the enhanced ZigBee specification was finalized [4].

2.2.3 XBee vs. Other Wireless Standards

Table 2.1 outlines some of the key characteristics of ZigBee and how it stacks up against other common wireless standards [5]

Table 2.1: XBee vs. Other wireless standards

	XBee	**802.11 (Wi-fi)**	**Bluetooth**	**IR Wireless**
Data Rate	20, 40,250 Kbits/s	11,54 Mbits/s	1 Mbits/s	20 – 40 Kbits/s 115 Kbits/s 4 & 6 Mbits/s
Range (meters)	10 – 100	50 – 100	10	< 10
Networking Topology	Ad-hoc, peer to peer, star, mesh	Point to Hub	Ad-hoc, very small Networks	Point to point
Operating Frequency	868 MHz (Europe) 900-928 MHz (NA), 2.4 GHz (Global)	2.4 and 5 GHz	2.4 GHz	800 – 900 nm
Complexity	Low	High	High	Low

Power Consumption	Very low (low power is a design goal)	High	Medium	Low
Other Information	Devices can join an existing network in under 30ms	Device connection requires 3 – 5 sec.	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 such as phones, PDA, laptops, headsets	Remote controls, PC, PDA, phone, laptop links

XBee looks rather like Bluetooth but is simple, has a lower data rate and spends most of its time snoozing. This characteristic means that a node on anXBee network should be able to run for six months to three years on just two AA batteries (see Table2.2)

The operational range of XBee is 1-100m compared to 10m for Bluetooth .XBee sits below Bluetooth in terms of data rate. The data rate of XBee is 250 kbps at 2.4 GHz, 40 kbps at 915 MHz and 20 kbps at 868 MHz whereas that of Bluetooth is 1 Mbps.

Table 2.2: Xbee VS Bluetooth

Market Name (Standard)	XBee® (802.15.4)	Bluetooth™ (802.15.1)
Application Focus	Monitoring & Control	Cable Replacement
Bandwidth (Hz)	20 – 250	1000
Battery Life (days)	100 - 1000+	1 - 7
Network Join Time	0.3 - 18 msec	3 sec
System Resources	4 kb - 32 kb	250 kb+
Transmission Range (meters)	1 - 100+	1 - 10+
Success Metrics	Reliability, Power, Cost	Convenience

Bluetooth's protocol is more complex since it is geared towards handling voice, images and file transfers in ad-hoc networks. Bluetooth devices can support scatter nets of multiple smaller non-synchronized networks (piconets) and it only allows up to 7 slave nodes in a basic master-slave piconet set-up.

When XBee node is powered down, it can wake up and get a packet in around 15 mile second whereas a Bluetooth device would take around 3 second to wake up and respond.

2.2.4 XBee VS XBee Pro

The XBee and XBee-PRO RF Modules were engineered to meet IEEE 802.15.4 standards and support the unique needs of low-cost, low-power wireless sensor networks. The modules require minimal power and provide reliable delivery of data between devices. The modules operate within the ISM 2.4 GHz frequency band and are pin-for-pin compatible with each other.

Table 2.3: XBeeVsXBee Pro

Features	XBee	XBee Pro
Long range Data Integrity	- Indoor : up to 100’ / 30m - Outdoor line-of-sight: up to 300’/90m - Transmit power : 1mW (0dBm) - Receiver Sensitivity :- 92 dBm	- Indoor : up to 300’ / 90m - Outdoor line-of-sight: up to 1mile/ 1600m , 2500’ /750m for international variant 300’/90m - Transmit power : 63mW (18dBm), 10mw(10dBm) for international variant - Receiver Sensitivity : -100 dBm - RF Data Rate : 250,000 bps
Low Power	- TX Peak Current : 45mA (3.3V) - RX Current: 50mA(3.3V) - Power-down Current: <10uA	- TX Peak Current : 250mA - RX Current: 55mA (3.3V) - Power-down Current: <10uA

	- Analog-to-digital conversion, Digital I/OI/O Line Passing
Easy-to-use	- No configuration necessary for out-of box RF communications - Free X-CTU Software (Testing and configuration software) - AT and API Command Modes for configuring module parameters - Extensive command set - Small form factor

2.3 Xbee

Figure 2.1: Xbee Module

Zigbee wireless protocol as shown in Figure 2.1 provides to network a set of autonomous devices with standard radio frequency transceiver to perform some networked task. In the proposed system a vehicular RF takes the role of aXbee end devices while as reader and writer module takes the role of Xbee coordinates [9]. DigiXBee 802.15.4 V as shown in figure 2.1 are the easiest-to-use, most reliable and cost-effective RF devices.

Xbee series 1 is used in this project. To ensure the data are successfully transferred with the other Zigbee receiver, some programming needs to be installed for both the zigbee using X-CTU software. This is will be explained further in the chapter 3. The 802.15.4 XBee modules provide two friendly modes of communication – a simple serial method of transmit/receive or a framed mode providing advanced features. XBees are ready to use out of the package, or they can be configured through the X-CTU utility or from the microcontroller. These modules can communicate point to point, from one point to a PC, or in a mesh network. . XBee 802.15.4 modules are cross-compatible with other 802.15.4 XBee modules, regardless of antenna type or power rating.

Features:

- 3.3V @ 50mA
- 250kbps Max data rate
- 1mW output (+0dBm)
- 300ft (100m) range
- Fully FCC certified
- 6 10-bit ADC input pins
- 8 digital IO pins
- 128-bit encryption
- Local or over-air configuration
- AT or API command set
- Trace Antenna

2.4 USB XBee Adapter

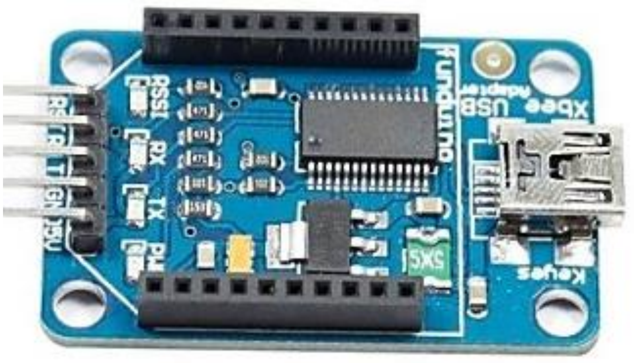

Figure 2.2: Xbee USB adapter

The Xbee USB adapter as shown in figure 2.2 above is used for the XBee module configuration parameters to facilitate or to use as a wireless data transmission. It can be easily connected to a PC via mini USB cable. Further, it has XBee-setting support software X-CTU. It can also be used as a USB-TTL adapter, ArduinoBootloader Programmer, USB-UART Interface.

Features

- Used for the XBee module configuration parameters
- Easily connected to a PC via mini USB cable
- XBee-setting support software X-CTU
- Also used as a USB-TTL adapter
- Voltage: +5V(USB Power)
- XBEE, Bluebee
- Free USB Mini Cable

2.5 Arduino Uno

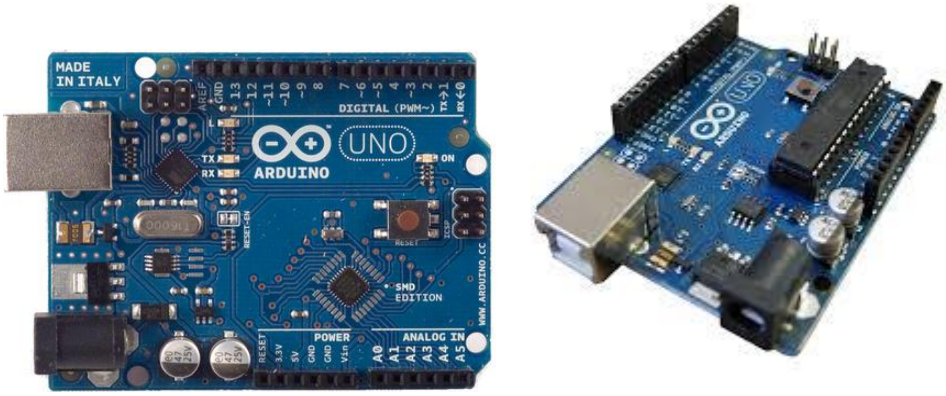

Figure 2.3: Arduino Uno Board

The Arduino Uno as shown in Figure 2.3 is a microcontroller board based on the ATMEL microcontroller ATmega328. It has 14 digital input or output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button [11]. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Arduino Uno can be powered via the USB connection or with an external power supply. The power source is selected automatically.

Arduino is an open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling lights, motors, and other actuators. The microcontroller on the board is programmed using the Arduinoprogramminglanguage and the Arduino development environment. Arduino projects can be stand-alone or they can communicate with software running on a computer.

2.6 Passive Infrared Sensor (PIR)

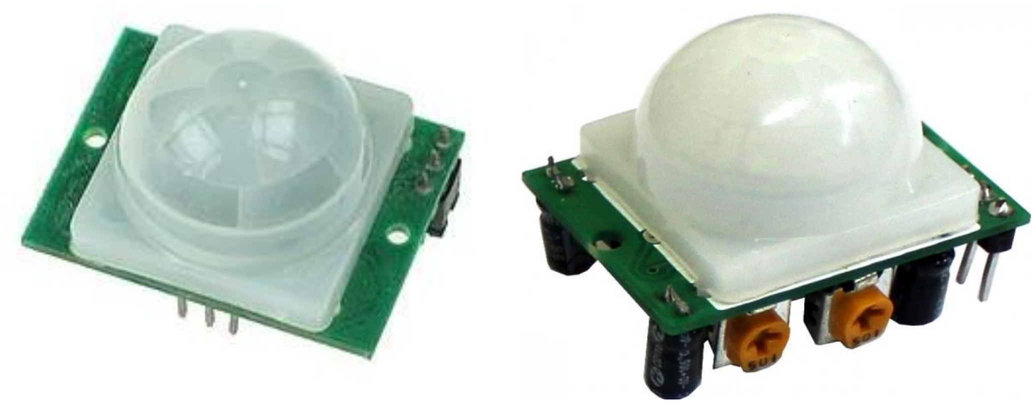

Figure 2.4: Passive Infrared Sensor (PIR)

PIR or the Passive Infrared sensor as shown in Figure 2.4 is an electronic device which is used to measure infrared light radiating from a body in its field of view. The Pyro electric "passive" Infrared (PIR) sensor itself has two slots in it; each slot is made of a special material that is sensitive to Infrared Radiation (IR).

The sensors work when a heat source i.e. a human body is detected within the first half of the viewing area of the PIR which causes a positive differential change between the two halves. When the warm body leaves the sensing area, the reverse happens, which generates a negative differential change. These change pulses are what is detected.

The PIR sensor itself is housed in an airtight metal can to improve protection from noise, temperature and humidity. There is a window made of IR-trans missive material (typically coated silicon) that protects the sensing element. Behind the window are two balanced sensors.

The sensor itself also needs another component to function at its best, a lens. The PIR sensor has a Fresnel and convex lens covering the sensor; these components help to provide a larger viewing area as well channelling any incoming signals directly towards the sensor itself.

2.7 Indicator

Some indicators have been installed with the system in the project. There are including buzzer, LED and exhaust fan. The information details on the devices use explained in the following subsections.

2.7.1 LED

Figure 2.5: LEDs

A light emitting diode (LED) as shown in Figure 2.5 is a semiconductor light source. LED is used as the indicator lamp in the many devices and is increasingly used for lighting. The LED is based on the semiconductor diode. When a diode is forward biased which is switch on, electron are able to recombine with holes within the devices, releasing energy in the form of photon. This effect is called electroluminescence and the colour of the light is determined by the energy gap of the semiconductor. LED are usually integrated optical components are used to shape its radiation pattern and assist in the reflection.

2.7.2 Exhaust Fan

Figure 2.6: Exhaust fan model

Exhaust fan as shown in Figure 2.13 is a fan for ventilating an interior by drawing air from the interior and expelling it outside. This project needs a system combination with exhaust fan as the precaution step before entering the dangerous level. Exhaust fan will suck out all the air inside the room or building that had been installed with the system to the outside of the building. Therefore, the air quality inside the building will maintain in the safe air quality.

2.7.3 Buzzer

A buzzer or beeper is an audio signalling device. Typical uses of buzzers and beepers include alarm devices, timers and confirmation of user input such as a mouse click or keystroke. The project used an electronic type of buzzer which is a piezoelectric element that driven by an Arduino Uno microcontroller signals.

Figure 2.7: Buzzer or beeper

2.8 Related Previous Work

Bridgwater (2005) justified the use of XBee in wireless systems in Hotels wireless system management. Bridgwater listed possible usage of functions that are possible to be implemented wirelessly. He has also showed how the wireless system can be implemented using XBee. Culter (2005) aimed to preserve existing industrial automated networks to save cost. He uses compatible address between Modbus and XBee to demonstrate that XBee can be integrated into the traditional system. Egan (2005) stated his opinion from the commercialization point of view. He stated the advantages of XBee and argued that XBee will become more and more popular in Building Automation and Industrial Control applications. Evans-Pughe, 2003 reviewed on XBee. He stated the advantages of XBee in terms of cost, power consumption, and reliability. The system is also able to be integrated with any home security system to enable the home status be monitored on-line through telephone or internet. This is a new era of home network systems.

CHAPTER 3:

DESIGN METHODOLOGY

3.1 Introduction

The project is divided into two parts which are hardware and software. For the software implementation, it involves writing code and programming the Arduino and Xbee. Meanwhile, hardware implementation involves designing the circuit of the project. After both parts were complete the next was the testing and debugging process. Each part of the project will be discussed in details in this chapter.

3.2 Project Methodology

The flowchart of project methodology is shown in Figure 3.1

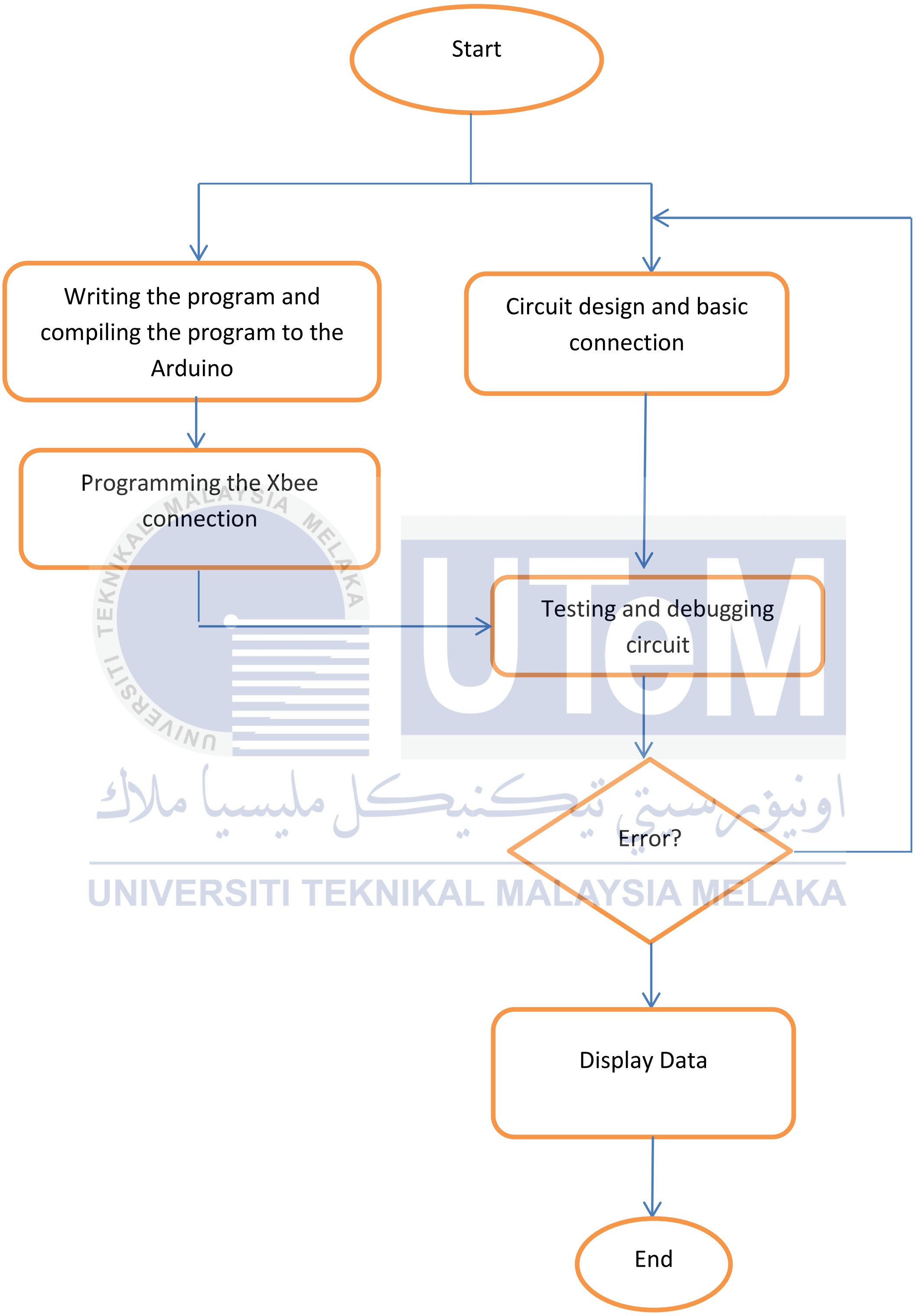

Figure 3.1 Methodology of the Project

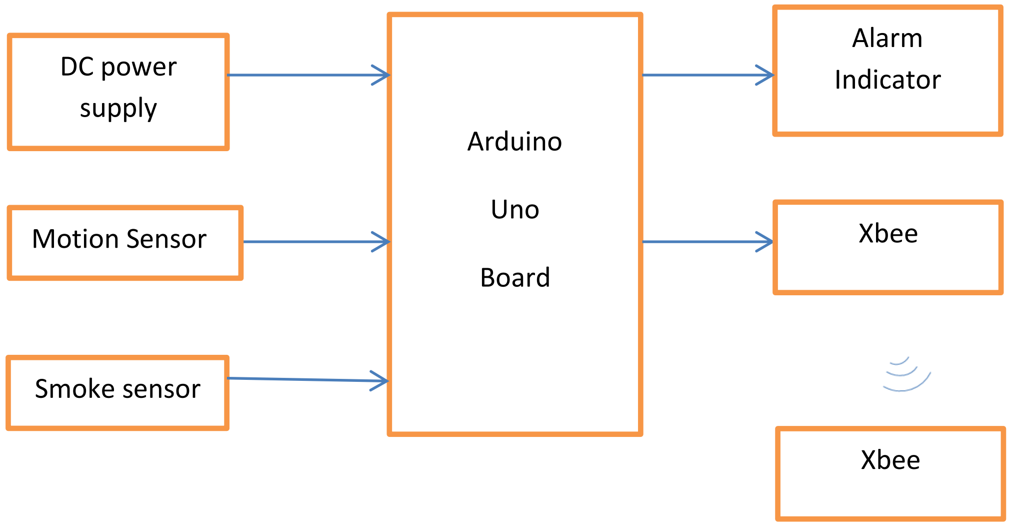

Figure 3.2 Block Diagram

As shown in Figure 3.2, the block diagram is about the connection of the motion detector implementation. In this project, output voltage from the motion sensor will be delivered to the Arduino board. The heart of the system is the Arduino board. All the inputs and the outputs will be connected to the Arduino. When the motion sensor detects the presence of gas, it will send analogue signal to an analog digital converter (ADC) inside the Arduino. An Arduino will process this signal and transfer to Xbee.

3.3 Hardware implementation

In the wireless monitoring system of smoke and PIR sensor, there are a few parts of hardware involved. This section will discuss on the design and function of each component that are connected to the Arduino in order to build the project. This section also explains how the circuitry connection between the components and Arduino Uno. This includes the sensor circuit and also the output circuit which comprises of LEDs, exhaust fan and a buzzer.

3.3.1 Gas sensor

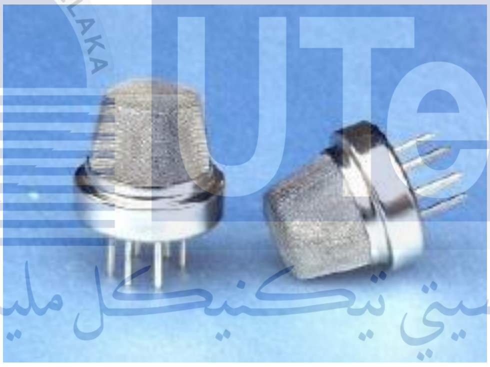

Figure 3.3: Smoke sensor (MQ2)

In this project, MQ2 is used to detect smoke sensor as shown in Figure 3.3.

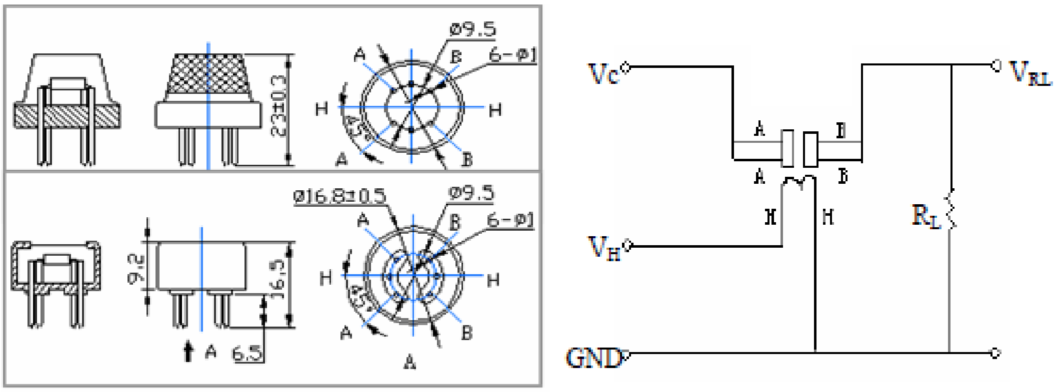

Figure 3.4: Smoke sensor circuit diagram

The circuit diagram for both the sensor is the same as shown in Figure 3.4.There are three output pins from this sensor which are reference voltage (VCC), ground pin (GND) and the output voltage pin. RL is the adjustable resistance. The connection between protection resistor in the sensor circuit and adjustable resistor are in serial which form a load resistor (RL). The sensor's resistance between Rs and RL form a voltage divider. Based on the chart provided in the MQ2 data sheet, Rs in the clean air under given temperature and humidity constant. Figure 3.4 also shows the sensor come out with 6pins. Pin H act as the coil of the gas sensor. Meanwhile, pin A and B are connected in pairing and was connected as in the circuit diagram above.

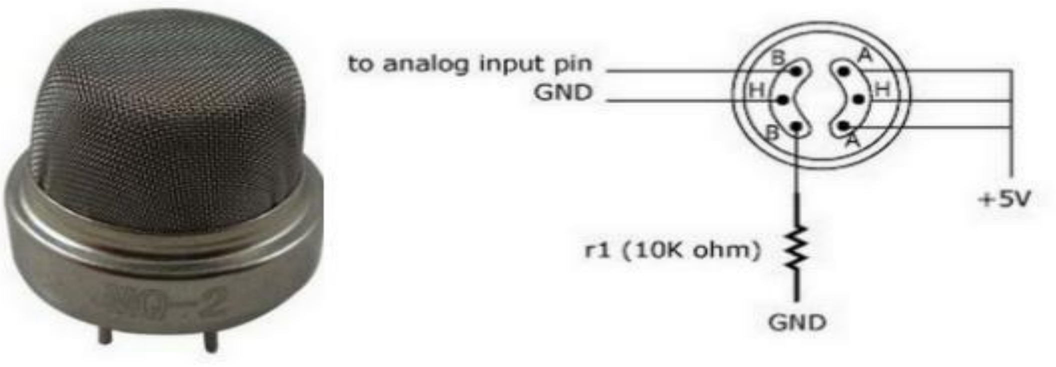

Figure 3.5: The smoke Sensor Connection to Arduino Pins

Figure 3.5 above shown gas sensor connection to the Arduino pins. The A0 pin connected at the Arduino will read the output voltage from the sensor.

3.3.2 Output Circuit

The output circuit consists of four type components which are:

a) Light-Emitting Diode (LED)
b) Buzzer
c) Exhaust Fan
d) Xbee

These components have their own purpose for the project in order to inform the users that the level of dangerous gas in the surrounding areas. There are three LEDs used namely as red LED, yellow LED and green LED. All of these LED indicate three different level of gas concentration in the atmosphere

Table 3.1: Threshold Value for Gas Concentration MQ2

LED	Gas concentration value (ppm)
Green	50 >= X => 300
Yellow	300 >= X => 800
Red	>800

When the red LED is light up, buzzer will also be trigged to inform user that the surrounding areas have reached a dangerous level and emergency evacuation is needed. Meanwhile, exhaust fan will turn on along with the yellow LED to show the level concentration in the warning stage and exhaust fan is used to suck out the dangerous gas as the precaution before entering the dangerous level.The condition of threshold value for gas concentration as shown in Table 3.1.

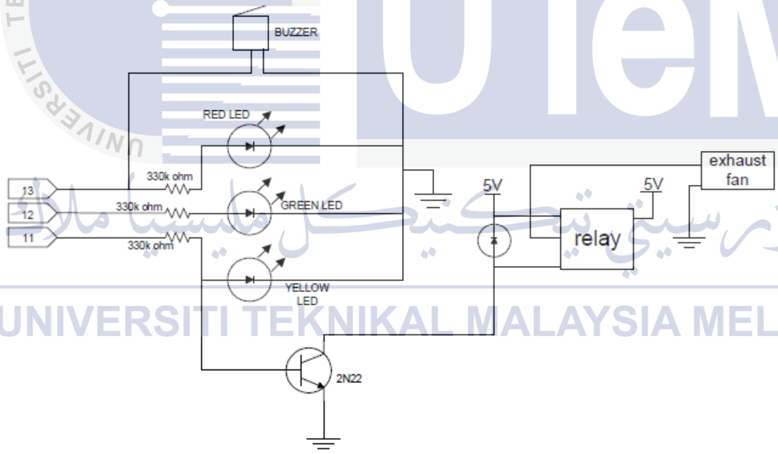

Figure 3.6: Output Circuit Diagram

Figure 3.6 show the output circuit diagram which act as the alarm system to the system. Red LED and buzzer are connected in the same pin from the Arduino, 13. Same as the yellow LED and exhaust fan at the pin 11. At the same time, pin 11 includes the relay circuit act as a switch to the exhaust fan. All the LEDs are covered with a light transparent casing to give visibility to the brightness when the LEDs light up. The output for pin 12 from Arduino is green LED only.

Last but not least, the output circuit from the Arduino is Xbee. Xbee from the output pin RX and TX in the Arduino acts as the transmitter. All the data from the sensor reading will be transferred to another Xbee receiver to interface with computer monitor display in the computer by wireless. Figure 3.7 below shows the circuit output diagram for Xbee from the Arduino.

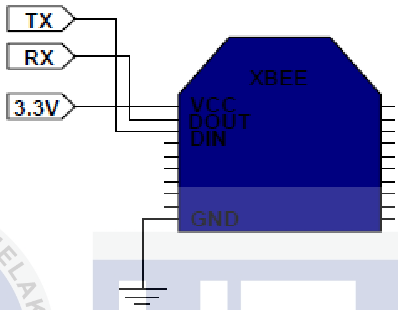

Figure 3.7: Xbee output connection circuit

3.3.3 PCB development

For the PCB development, strip board is used in this project. Before soldering all the components on the board, bread board was first used to ensure that the connection between each component is functional for this project. Donut board have connection to each hole like the bread board. In order to make the connection, solder method is used to connect all components. Besides that, female holders are uses to connect the jumper wires from Arduino to the board circuit. The PCB development circuit diagram is shown in Figure 3.8 below

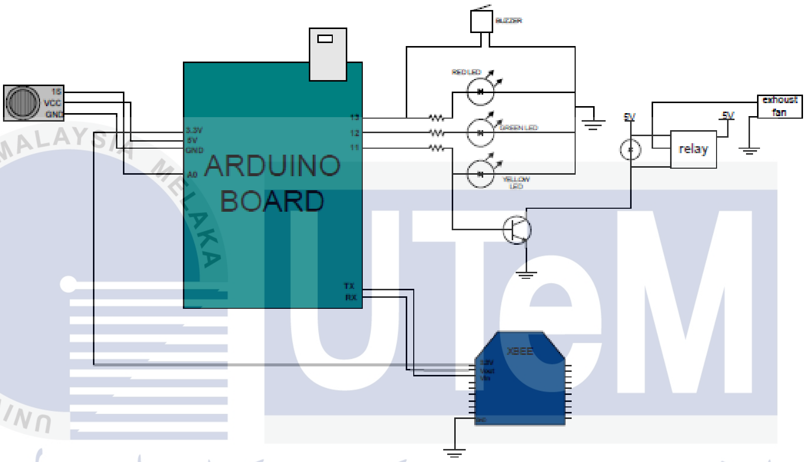

Figure 3.8: Circuit Diagram

3.4 Simulation Approach

3.4.1 Motion Sensor

The flowchart of Arduino programming for PIR motion sensor is shown in Figure 3.9.

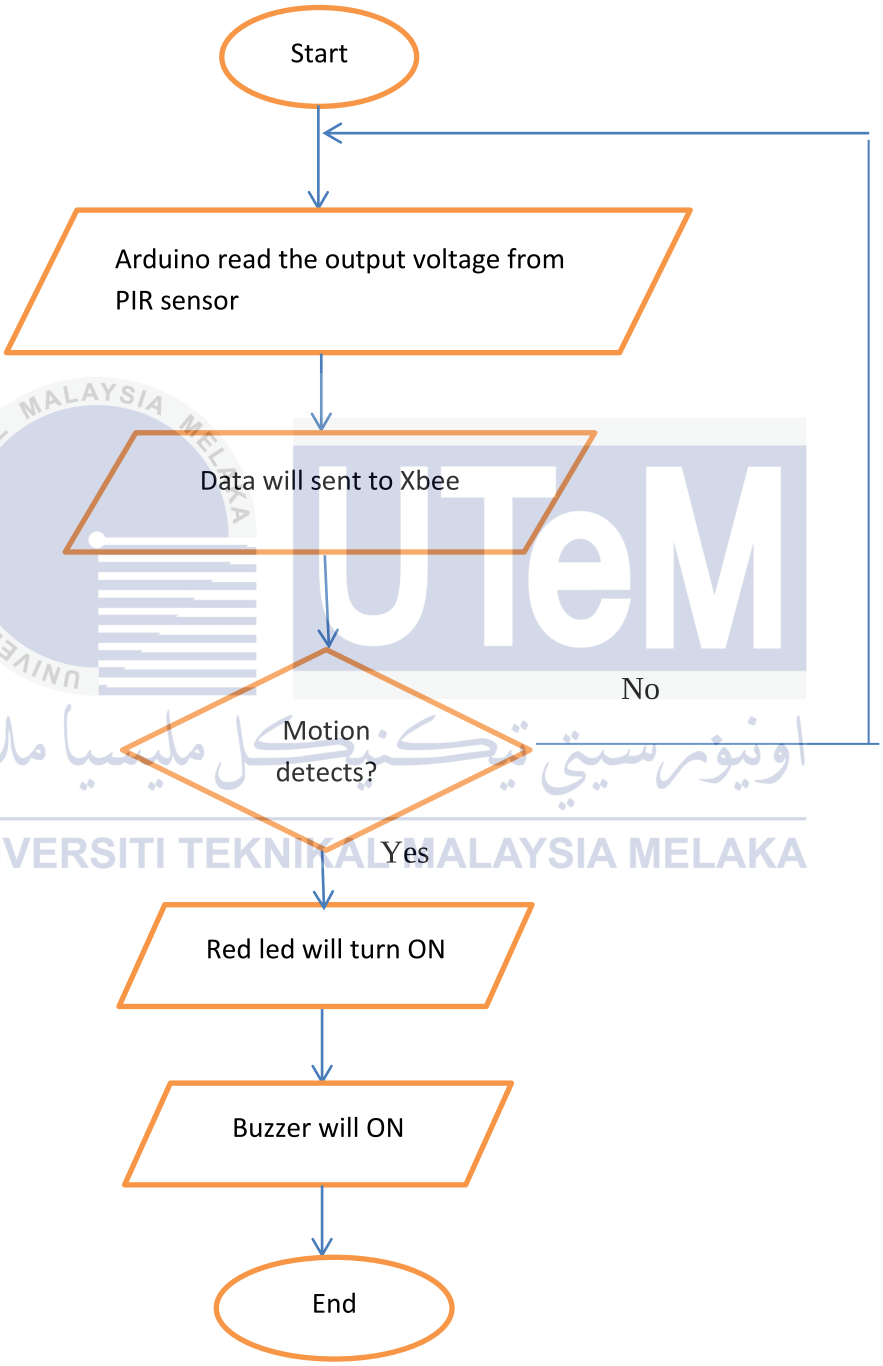

Figure 3.9: Flow chart of Arduino programming for PIR motion sensor

Firstly, all the input and output pin in Arduino must be declare. Then, initial sensor value must be set as 0. The data receive from sensor can calculate in the decimal place as the float sensor declare in line 6 below. Analog input pin that the potentiometer of gas sensor is attached to pin A0. The declaration programming for this step shows below figure 3.10

```
intcalibrationTime = 30;
booleansensorActive = false;
booleanpreviousSensorState = false;
intpirPin = 4;    //the digital pin connected to the PIR sensor's output
intledPin = 13;
```

Figure 3.10: Declaration of the pin

After that, loop is very important which represent all the data will continuously repeat by time. In this project, there are 2 loop include. First loop is to initialize serial communication at the bound rate 9600. This is also sets the digital pin as the output as shown in figure 3.11


```
void setup(){

Serial.begin(9600);

pinMode(pirPin, INPUT);

pinMode(ledPin, OUTPUT);

digitalWrite(pirPin, LOW);
```

```
  //give the sensor some time to calibrate
Serial.println("Sensor Calibration in Progress");
Serial.println("------------------------------");
for(int i = 0; i <calibrationTime; i++){
Serial.print(".");
digitalWrite(ledPin, HIGH);
delay(250);
digitalWrite(ledPin, LOW);
delay(250);
digitalWrite(ledPin, HIGH);
delay(250);
digitalWrite(ledPin, LOW);
delay(250);
  }
Serial.println("");
Serial.println("Sensor Calibration Completed");
Serial.println("Sensor Reading Active");
delay(50);


sensorActive = false;
previousSensorState = false;
}
```


Figure 3.11: Looping

Meanwhile, the second loop is mostly the main process for the alarm system trigger. All the threshold value is declared in this loop. First of all, the system read the sensor value as shown in Figure 3.12. If the sensor value is over or under the limitation of threshold value, some indicator as the output system will be turning on. Finally, the result from the sensor value will be print to the serial monitor.

```
void loop()
{
  // takes the pin value and saves it to the sensorActiveboolean value
if(digitalRead(pirPin) == HIGH)
  {
sensorActive = true;
digitalWrite(ledPin, HIGH);
  }
if(digitalRead(pirPin) == LOW)
  {

digitalWrite(ledPin, LOW);
  }


  // performs action if the state of the sensor changes
  // since this is a loop, here is now it works:
```

```
// if the sensor pin goes HIGH (on) after it being LOW (off), the sensorActive value changes from the previousSensorState value.

// it then turns on the LED. when the pin goes LOW (off) it will do the same thing but opposite values.

// it also prints status to serial. it will print the time of triggering by providing the number of seconds that have passed since the program started.

if(sensorActive != previousSensorState)

{  if(sensorActive == true)

{ previousSensorState = sensorActive;

Serial.println("---")

Serial.print("Motion Detected At: ");

Serial.print(millis()/1000);

Serial.println(" Seconds");

delay(50);    }

if(sensorActive == false)

{  previousSensorState = sensorActive;

Serial.println("---");

Serial.print("Motion Stopped At: ");

Serial.print(millis()/1000);

Serial.println(" Seconds");

delay(50);   }  }

}
```


Figure 3.12

3.4.2 Smoke Sensor

Firstly, all the input and output pin in Arduino must be declare. Then, initial sensor value must be set as 0. The data receive from sensor can calculate in the decimal place as the float sensor declare in line 6 below. Analog input pin that the potentiometer of gas sensor is attached to pin A0. The declaration programming for this step as shown Figure 3.13

```
constintanalogInPin = A0;

constintledred = 13;

constintledyellow = 12;

constintledgreen= 11;

intsensorValue = 0;

float sensor;
```

Figure 3.13: Arduino input and output port initialization

After that, loop is very important which represent all the data will continuously repeat by time. In this project, there are 2 loop include. First loop is to initialize serial communication at the bound rate 9600. This is also sets the digital pin as the output as shown in Figure 3.14

```
void setup()

{

Serial.begin(9600);

pinMode(ledgreen, OUTPUT);
```

```
pinMode(ledred,OUTPUT);

pinMode(ledyellow,OUTPUT);

}
```

Figure 3.14: Declaration of pin

Meanwhile, the second loop is mostly the main process for the alarm system trigger. All the threshold value is declared in this loop. First of all, the system read the sensor value as shown in Figure 3.16. If the sensor value is over or under the limitation of threshold value, some indicator as the output system will be turning on. Finally, the result from the sensor value will be print to the serial monitor. The flow chart of the programming is shown in the Figure 3.15

```
void loop() {

sensorValue = analogRead(analogInPin);

if (sensorValue>= 800)

{

digitalWrite(ledgreen, LOW);

digitalWrite(ledyellow, LOW);

digitalWrite(ledred, HIGH);

delay(200);

digitalWrite(ledred, LOW);

delay(200);

}

else if (sensorValue> 300&&sensorValue< 800)
```

```
{
digitalWrite(ledgreen, LOW); // sets the LED on
digitalWrite(ledyellow, HIGH);
digitalWrite(ledred, LOW); // sets the LED off
delay(200);
digitalWrite(ledyellow, LOW);
delay(200);
}
else if (sensorValue>50&&sensorValue<300)
{
digitalWrite(ledgreen, HIGH); // sets the LED on
digitalWrite(ledyellow, LOW);
digitalWrite(ledred, LOW); // sets the LED off
delay(200);
digitalWrite(ledgreen, LOW);
delay(500);}
Serial.print("sensor Value  = " );
Serial.println(sensorValue);
delay(500);
```


Figure 3.15: Programming Process in Arduino

3.4.3 Xbee Programming

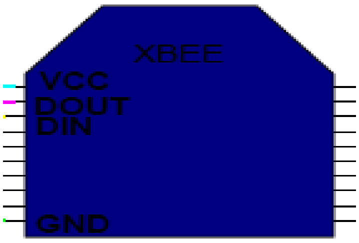

Figure 3.16: Xbee

In order to connect between two Arduino wirelessly, the two Xbee must be configuring first. Both of the Xbee receiver and transmitter must be set up with the programming in order to transfer data from Arduino successfully. The Arduino Uno is shown in Figure 3.6.

XCTU software is needed to run the programming with Xbee. Once the XCTU software is installed and executed in the computer, each COM for each Xbee must be tested by clicking on the button Test/Query as shown in Figure 3.17. XCTU software is support for programming and configuring Xbee, WIFI modules. After that a dialog box will popped up to inform that the COM connection is successful. Figure 3.18 shows the result output for Com test.

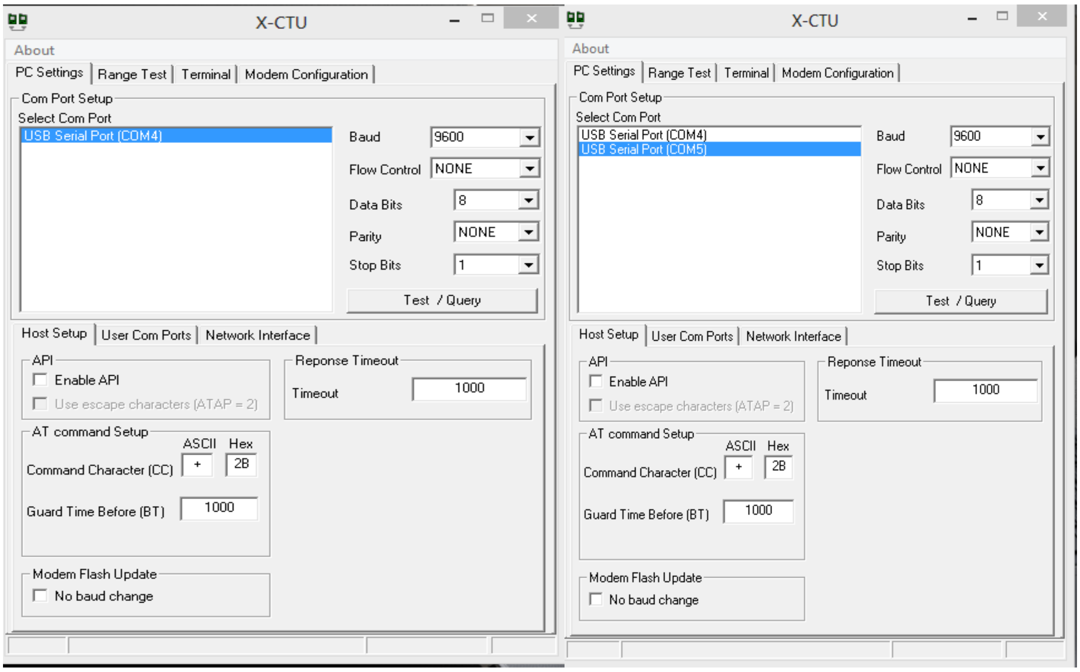

Figure 3.17 : Test the Com of Xbee

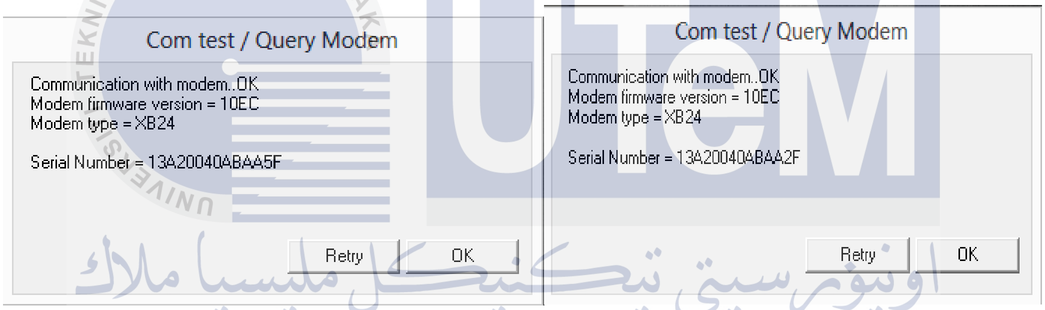

Figure 3.18: Result Output for Com test

The set up for the Xbee data transfer is done by opening the modem configuration at the up right corner of the window XCTU. This step is very important to make sure that the data has been transfer to the exact location. There were four items that need to be considered as shown in Table 3.2. First is PAN ID. This is to show the location number of the port. The value of the ID must be the same. As shown in figure 3.19, the ID for this Xbee is 3332. Then set the destination address high as 13A200 and the destination address low as 40ABAA2F. Serial interfacing will also be the most important things in this step. As we set the bound rate at the Arduino at 9600, the interfacing data rate also must be 9600.

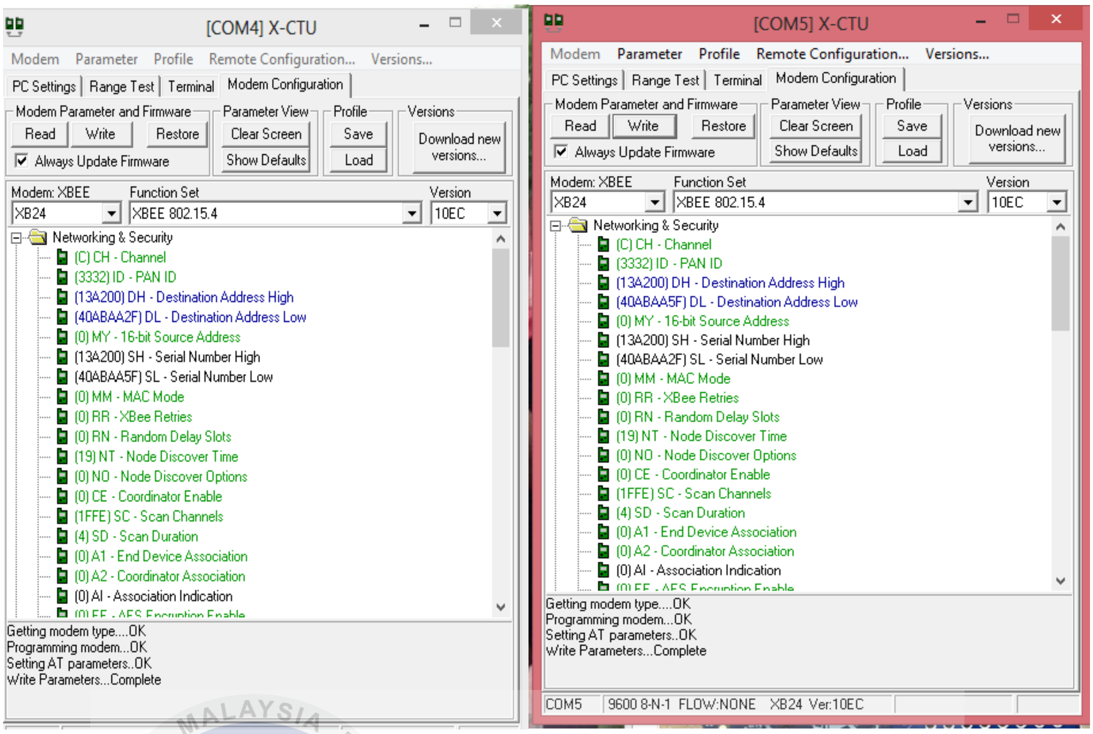

Figure 3.19 : Setting the Xbee

Table 3.2: PAN ID

Coordinator		End device	
SH	13A200	SH	13A200
SL	40ABAA5F	SL	40ABAA2F
DH	13A200	DH	13A200
DL	40ABAA2F	DL	40ABAA5F

The last step is to test the connection between two Xbee. Figure 3.20 shows the data transfer between two Xbeeis successful. As shown in figure 3.20, the writing in blue colour is the data transfer at COM3. Meanwhile, the red colour in COM4 is the receiver and vice versa. Therefore, both of the Xbee can be used as the receiver and transmitter terminal

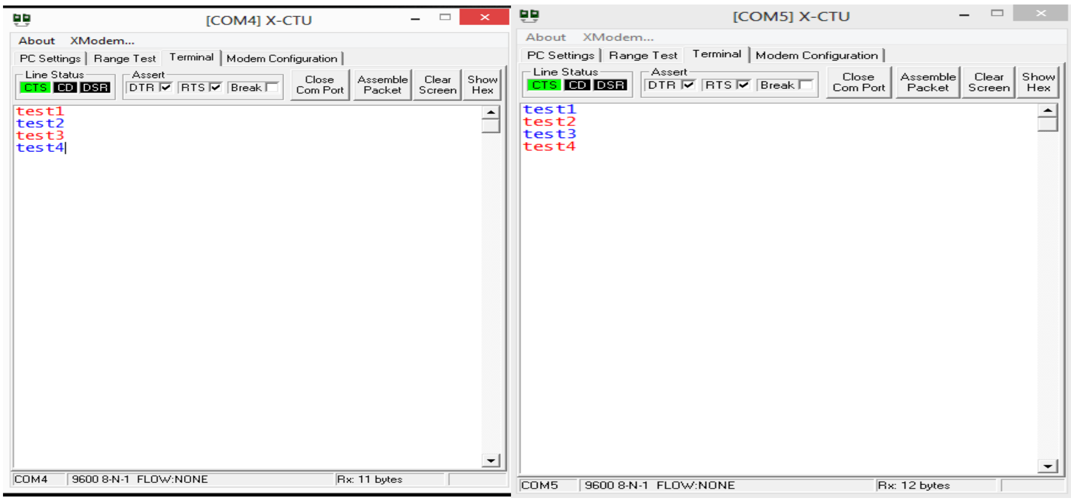

Figure 3.20: data transfer between two Xbee

CHAPTER 4:

RESULT AND DISCUSSION

4.1 Introduction

This chapter explains about the results achieved in the project and a few discussions on problem solving during process and experiment of completing this project

4.2 Project Description

Project done is based on the objectives state before which is design and develop a security system for monitoring any intruders and others emergency situation around the house. The sensor node will detect the heat body for motion sensor and for flammable gas and smoke; sensor node will detect the level of concentration of gas and convert it into analog voltage and directly sending it into Arduino. After transferring the data, Arduino will read the data into the digital format. Arduinoprocesses analog to digital converter (ADC) from 0 to 1024 which are in 10 bits. Voltage output from the motion sensor (0-5) V will be read as (0-1024) decimal output in Arduino.

4.3 Project Result

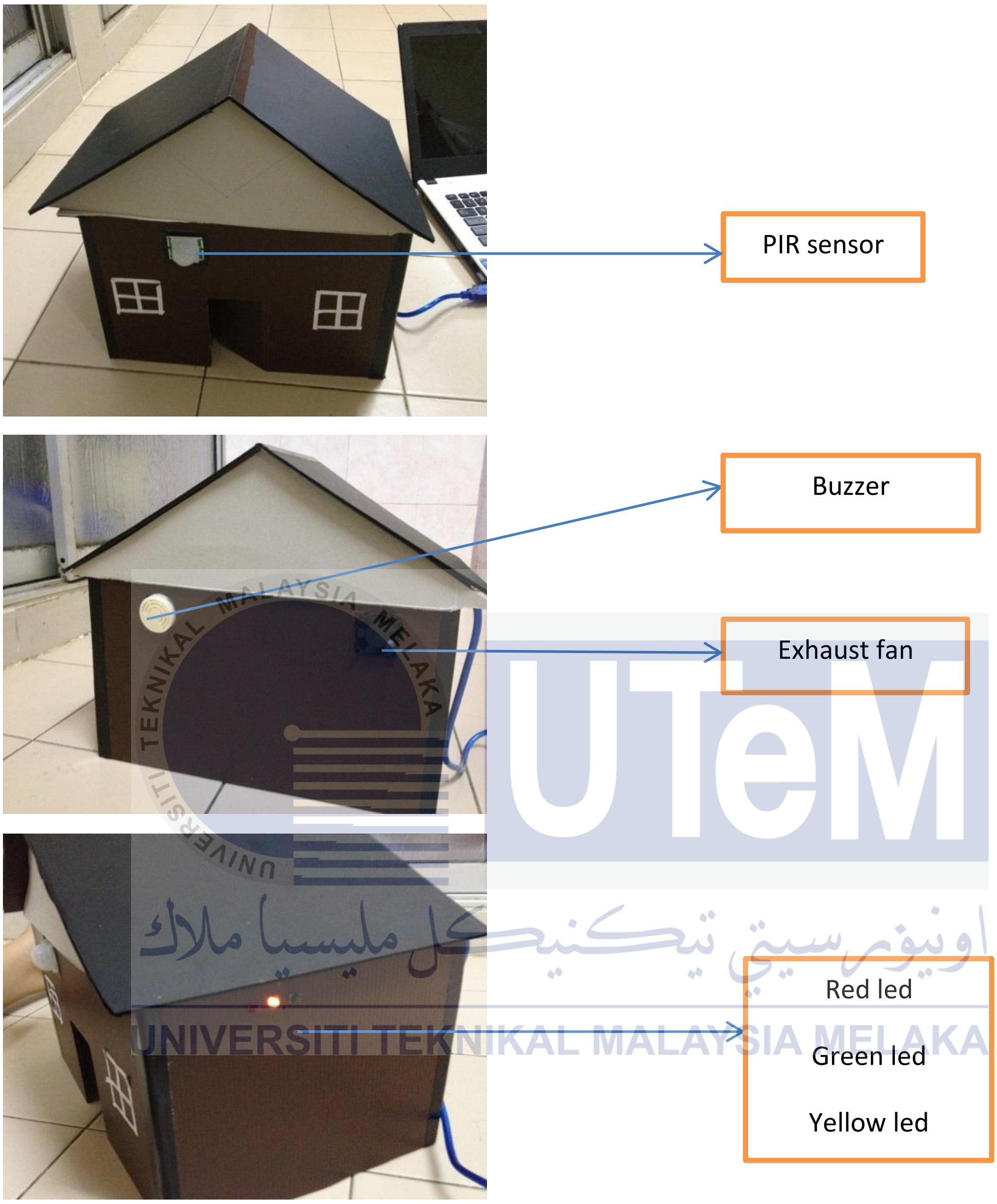

Figure 4.1: Overall Project Configuration

The overall project set up shown in the Figure 4.1, the systems are divided into two part. First part is on the alarm system. The second part is on the software system which is monitoring system.

An experimented was carried out using two type of sensor which are PIR motion sensor and smoke sensor in order to achieve the objective which are design and develop a security system for monitoring any intruders and others emergency situation. The motion sensor is for monitoring any intruders while smoke sensor is for the emergency situation.

Figure 4.2 (a) Figure 4.2 (b)

Figure 4.2 show the Xbee End Device is. Figure 4.2 (a) shown there are no signal from the Xbee Coordinator, which is the Xbee Coordinator does not receive any signal from Arduino Uno Board. In this case, there is no sensor detect in both motionand smoke sensor. While Figure 4.2(b) shown the light will On to shown it receives the signal from the Xbee Coordinator.

4.3.1 Motion Sensor

The first experiment was done with PIR motion sensor. This sensor detects the movements, normally used to detect human movement when passing in or out of range sensor. The result for the motion detection is shown in the Figure 4.3 below. The signal from motion sensor will be wirelessly sent to XCTU from Xbee Coordinator to display to the user in the form of computer monitor display.

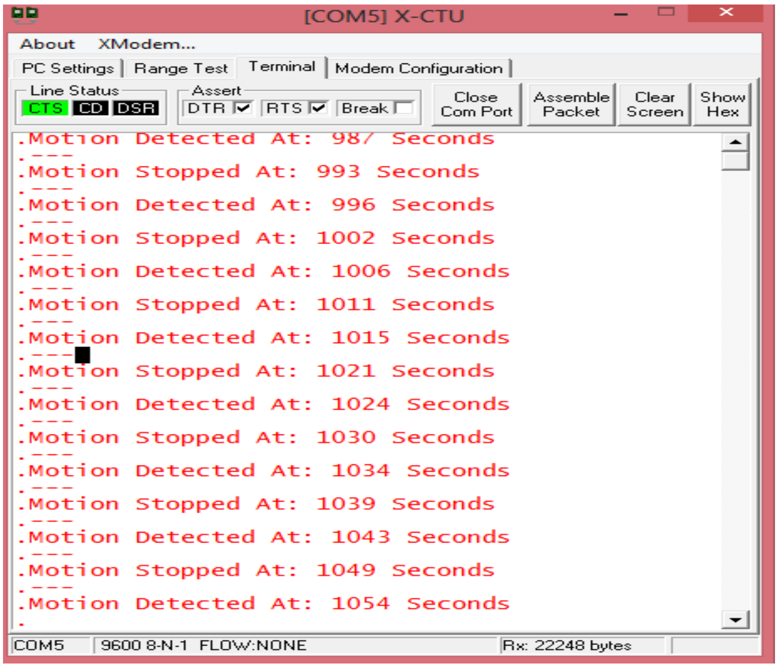

Figure 4.3: Motion sensor display in terminal XCTU of Xbee

4.3.2 Smoke Sensor

The second experiment was done on smoke sensor. The semiconductor gas sensor detects the presence of combustible gas and smoke at concentrations from 300 to 10,000 ppm. The sensor's simple analog voltage interface requires only one analog input pin from Arduino. The signal detected transfer to the Xbee Coordinator which are RX and TX that connected to the Arduino Uno board. Then, from the Xbee Coordinator the signal will wirelessly send to the Xbee End Device and display in the form of computer monitor display X-CTU.

This flammable gas and smoke sensor detects the concentrations of combustible gas in the air and ouputs its reading as an analog voltage. The sensor can measure concentrations of flammable gas of 300 to 10,000 ppm.

Table 4.1: Level concentration of combustible gas

Mode	>50 &< 300	>300&<800	>= 800
Green	High	Low	Low
Yellow	Low	High	Low
Red	Low	Low	High

Table 4.1 shown the level concentration of combustible gas and smoke that set up for this project. When the concentration gas in the computer monitor display as shown in the Figure 4.4 is greater than 50 and below the 300 ppm, the green LED will turn on. This situation means it is in normal condition.

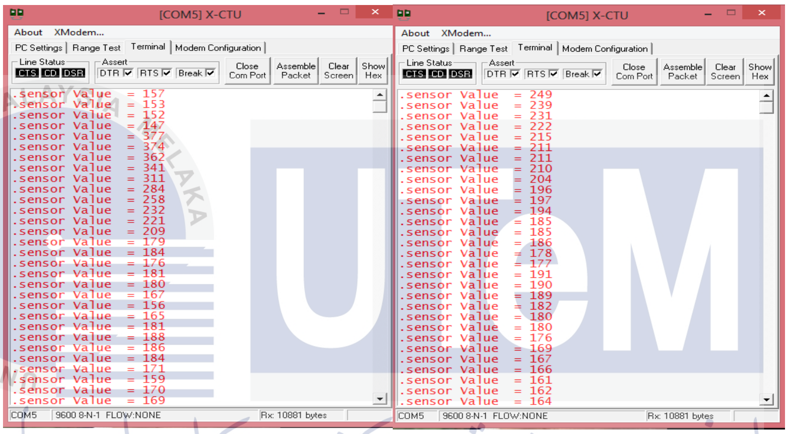

Figure 4.4: Concentration gas >50 and <300

Figure 4.5 shown the result concentration of flammable gas and smoke in the X-CTU computer monitor display. The sensor value is greater than 300 and below 800ppm. The output signal will be turn ON the Yellow LED and exhaust fan as the initially system set up in order to control the air quality in the house. Yellow LED means warning the system.

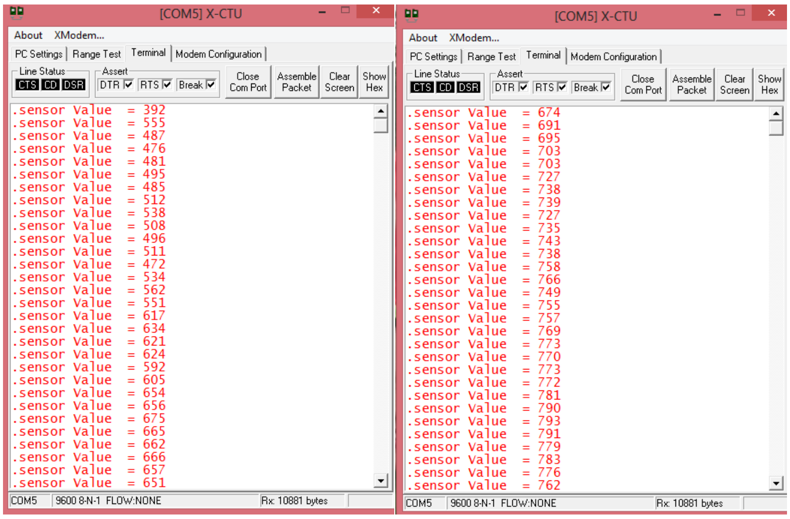

Figure 4.5: Concentrationof gas>300 and < 800

Figure 4.6 shows the concentration gas achieves 800ppm and above. The output result of the alarm system when Arduino threshold in the dangerous mode. The red LED and buzzer alarm will turn on as the initially system set up in order to warning the users its critical condition. Red LED means the condition in the dangerous situation.

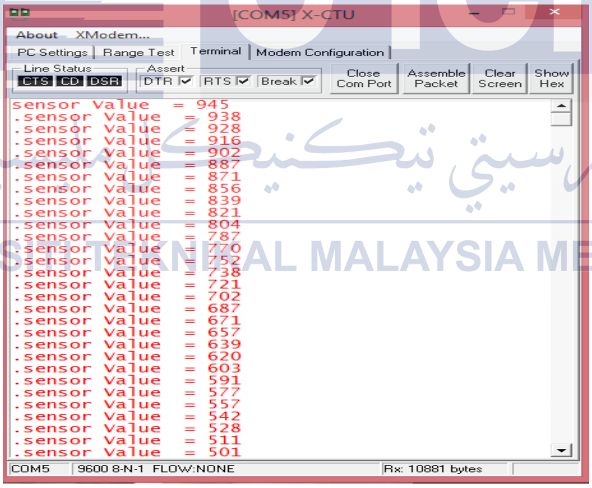

Figure 4.6: Concentration of gas >>800

CHAPTER 5:

CONCLUSIONS

5.1 Conclusions

Overall, this study has been completed properly. All sub-objective of this study has been completed, thus resolved the main objectives and research problems. This system managed to help users to detect human presence around the house precisely and detect the smoke or any combustible gas. This system is a system that meets recently smart home applications in order to function in automation situation. During the study, a lot of knowledge and experience have been learned. Besides the exposure on Arduino Uno microcontroller, this study also has much exposure to radio frequency technology which is Xbee , where it helps in the transfer process for the wireless signal. This device has helped the development of security systems with its high-tech features

5.2 Suggestion

This program should be reviewed in order to improve the information display in terms of accuracy in the future. This study also can be continued in future to make improvements into the existing studies. Some suggestions for improving this study area:

· Increase the number of sensors to enhance the security features
· Implement a different variety of sensors in order to detect certain objects
· Use camera to capture images when PIR detects an object and use LCD monitor for image display

REFERENCES

[1] CHEN, Dechuan and M. Wang. (2006) “A Home Security ZigBee Network for Remote Monitoring Application”, Hangzhou Dianzi University, China, Retrieval November 2008.

[2] Bridgwater, J., 2005. Wireless systems management in hotels with Zigbee technology. J. Elect., 22: 42-43.

[3] ZIGBEE ALLIANCE Web Site. (2006) “ZigBee Specification”, <http://www.zigbee.org/Products/TechnicalDocumentsDownload/tabid/237/Default.aspx>, Retrieval March 2007.

[4] WIKIPEDIA Web Site. (2008) “ZigBee”, <http://en.wikipedia.org/wiki/ZigBee>, Retrieval September 2008.

[5]] SOFTWARE TECH. GROUP Web Site. (2009) “How does ZigBee Compare with Other Wireless Standarts”, <http://www.stg.com/wireless/ZigBee_comp.html>, Retrieval February 2009.

[6] Rozeha A. Rashid, MohdAdib B. Sarijari, MohdRozainiAbd Rahim, NurHijaMahalin "Wireless Home Security and Automation System Utilizing ZigBee based Multi-hop Communication"

[7] Rozeha A. Rashid, NorsheilaFisal, Abdul Hadi Abdul Hamid "Wireless Multimedia Sensor Network Platform for Low Rate Image Video” JurnalTeknologi, 54 (Sains&Kej.)KeluaranKhas, Jan. 2011: 231–254

[8] Abdul HadiFikri Bin Abdul Hamid, RozehaA.Rashid and Prof. Dr.NorsheilaFisal “Development of IEEE802.15.4 based Wireless Sensor Network Platform for Image Transmission”

[9] Dae-Man Han and Jae-Hyun Lim “Smart Home Energy Management System using IEEE 802.15.4 and ZigBee” IEEE Transactions on Consumer Electronics, Vol. 56, No. 3, August 2010

[10] NorsheilaFisal, Ahmad Sharmi Abdullah, Muhammad YusofMohdNor "Development Of WSN And Optical Applications" PenerbitUniversitiTeknologi Malaysia, UTM Press 2008

[11] Datasheet, Uno, T. A. Arduino Uno, *328*.

[12] XBee® Wi-Fi RF Module Data Sheet, Digi International 2011

[13] ZIGBEE ALLIANCE Web Site. (2006) “ZigBee Specification”, <http://www.zigbee.org/Products/TechnicalDocumentsDownload/tabid/237/Default.aspx>, Retrieval March 2007

[14] GUTIERREZ, Jose. (2003) “Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs)”, ISBN 0-7381-3677-5 SS95127, Published by The Institute of Electrical and Electronics Engineers, New York, USA. Retrieval September 2008.

APPENDICES A

1. Arduino Programming for Smoke Sensor

combustible_gas | Arduino 1.0.5-r2

File Edit Sketch Tools Help

combustible_gas

```
const int analogInPin = A0;
const int ledred = 13;
const int ledyellow = 12;
const int ledgreen= 11;
int sensorValue = 0;
float sensor;

void setup()
{
Serial.begin(9600);
pinMode(ledgreen, OUTPUT);
pinMode(ledred,OUTPUT);
pinMode(ledyellow,OUTPUT);
}

void loop() {
sensorValue = analogRead(analogInPin);
if (sensorValue>= 800)
{
digitalWrite(ledgreen, LOW);
digitalWrite(ledyellow, LOW);
digitalWrite(ledred, HIGH);
delay(200);
digitalWrite(ledred, LOW);
delay(200);
}
```

```
else if (sensorValue> 300&&sensorValue< 800)
{
digitalWrite(ledgreen, LOW); // sets the LED on
digitalWrite(ledyellow, HIGH);
digitalWrite(ledred, LOW); // sets the LED off
delay(200);
digitalWrite(ledyellow, LOW);
delay(200);
}
else if (sensorValue>50&&sensorValue<300)
{
digitalWrite(ledgreen, HIGH); // sets the LED on
digitalWrite(ledyellow, LOW);
digitalWrite(ledred, LOW); // sets the LED off
delay(200);
digitalWrite(ledgreen, LOW);
delay(500);
}

Serial.print("sensor Value = " );
Serial.println(sensorValue);
delay(500);
}
```

Done Saving.

```
The sketch name had to be modified. Sketch names can only consist
of ASCII characters and numbers (but cannot start with a number).
They should also be less less than 64 characters long.
```

43 Arduino Uno on COM

2. Arduino Programming for Smoke Sensor

pirsensorA | Arduino 1.0.5-r2

File Edit Sketch Tools Help

pirsensorA §

```
/* PIR sensor tester*/
int alarmPin = 9;
int ledPin = 8; // choose the pin for the LED
int inputPin = 2; // choose the input pin (for PIR sensor)
int pirState = LOW; // we start, assuming no motion detected
int val = 0; // variable for reading the pin status

void setup()
{
pinMode(ledPin, OUTPUT); // declare LED as output
pinMode(inputPin, INPUT); // declare sensor as input
pinMode(alarmPin, OUTPUT);
Serial.begin(9600);
//Blink Led for 10 times before activate alarm (20 second)
for (int i=0; i < 10; i++)
{
digitalWrite(ledPin,HIGH);
delay(1000);
digitalWrite(ledPin, LOW);
delay(1000);
}

}
void loop(){      // Start detecting motion
val = digitalRead(inputPin); // read input value
if (val == HIGH)

{ // check if the input is HIGH
digitalWrite(alarmPin, HIGH); // turn LED ON
digitalWrite(ledPin, HIGH);
if (pirState == LOW)
{
Serial.print("Motion detected!");
delay (60);
pirState = HIGH;
}
}

else
{
digitalWrite(alarmPin, LOW); // turn LED OFF
digitalWrite(ledPin, LOW);
if (pirState == HIGH){
Serial.print("Motion ended!" );
pirState = LOW;
}
}
}
```

APPENDICES B

FYP 1 GANT CHART

APPENDICES C

FYP 11 GANTT CHART

APPENDICES D

COMBUSTIBLE GAS SENSOR