Portable Security System

by

Wong Lee See

Dissertation submitted in partial fulfillment of
the requirements for the
Bachelor of Engineering (Hons)
(Electrical and Electronic Engineering)

DECEMBER 2011

UniversitiTeknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak DarulRidzuan

CERTIFICATION OF APPROVAL

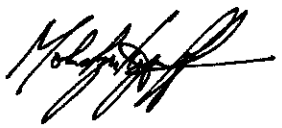
Portable Security Sytsem

by

Wong Lee See

A project dissertation submitted to the
Electrical and Electronic Engineering Department
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
BACHELOR OF ENGINEERING (Hons)
(ELECTRICAL AND ELECTRONIC ENGINEERING)

Approved by,



Dr. Mohd Zuki b Yusoff)

Project Supervisor

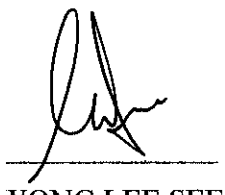
UniversitiTeknologiPETRONAS

Tronoh, Perak

Dec 2011

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.



WONG LEE SEE

ABSTRACT

This project puts the main concerns in designing a portable security system that is able to detect break in(s) through windows or doors in the house. The key point for the project is portability, which reflects the capability to carry or move the system around, so that the users will be able to safeguard their belongings and themselves wherever they are. This system would check for the condition of the sensor(s) continuously to ensure no intrusion. A mobile phone Sony Ericsson T630 is being used in this project. When the sensor is triggered off, the system will then display a text string "help" on PC. On the other side, PC will control the mobile phone Sony Ericsson T630 to send a text message to the pre-programmed phone number which can be the user's mobile or police help line, to inform him/her of the attempted burglary. AT command is used to send the text messages. This project will definitely be beneficial to everyone. It is still further developed and enhanced for more feasible features. It is a handy solution to overcome burglary.

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere appreciation to my supervisor, Dr. Mohd Zuki b Yusoff for his guidance, encouragement and motivation given throughout this project. I did encounter numerous difficulties for the project. However, he patiently discussed the problems with me and provided me with a clearer view for this project.

Besides that, I would like to thank my family members for providing me continual support and understanding throughout this project. Without their encouragement, love and support, it would be quite a struggle to finish this project.

Last but not least, I would also like to thank all my friends who have directly or indirectly helped me in completing my project. Their willingness to share their knowledge is indeed extremely helpful.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATION

ACK	-	Acknowledge
AT	-	Attention
DC	-	Direct Current
GSM	-	Global System for Mobile Communication
I/O	-	Input or Output
IC	-	Integrated Circuit
NO	-	Normally Closed
NC	-	Normally Open
PC	-	Personnel Computer
PDU	-	Protocol Description Unit
PIC	-	Peripheral Interface Controller
RS 232	-	Recommended Standard 232
TTL	-	Transistor-transistor Logic
USB	-	Universal Serial Bus

CHAPTER 1: INTRODUCTION

1.1 Background Study

Every day, when we pick up the newspapers, we see that there are many crime cases such as burglaries, robberies, snatch thieves, drug frauds and many more. We do not know whether the next person who appears in the headline is we ourselves or our relatives or anyone we know. Whatever happens, we are always at the losing end. Hence, it is crucial to have an advanced security system.

A **Security system** is an electrical device that sets off an alarm when someone tries to break in [1]. Thus, by installing a security system, we will be able to secure our premises. There are several types of security system available in the market such as wireless alarm systems, burglar alarm systems, and computerized access control systems. All these security systems act in such a way that whenever there is an intruder inside our premises, an alarm will be triggered off and it will warn us while the intruder panics and runs for his life.

A security system is designed to perform the three basic tasks as follows:

1. **To detect**

   The system consists of a central Control Panel that monitors the various entry points in users' home/building. If an intruder is trying to break into users' house/building through one of the protected entry points, the system will detect this and will trigger the alarm.

2. **To deter**

   When the intruder accidentally triggers any one of the components, either the magnetic contact or vibration sensor, both the Control Panel and Siren Unit will sound immediately with a piercing siren that will deter the intruder from continuing with the break in.

3. To inform

When the transmitter sends a signal to the Control Panel about the intrusion, the system will then inform the owner by calling the pre-programmed phone numbers which can be the hand phone or office phone, or by sending a message to warn the owner about the attempted burglaries.

1.2 Problem Statement

Nowadays, we can find many security alarm systems such as the CCTV and home alarm systems in the market. However, most of the security systems are very expensive and not affordable by poor or middle income families. Usually, it requires special technical personnel to install the security system and it is difficult to remove once it is being installed. Moreover, the security systems that are currently available in the market are not handheld or portable. For instance, when we go for a vacation and stay in a budget hotel that does not provide any personal security system, we need to pay great attention to our valuables and stuffs left in the room. Therefore, it is necessary to design a portable security system with advanced features which can easily be installed and removed. This requires a portable security system without complicated wiring or cabling which we can carry around with us when we go for a vacation or move into a new house/workplace.

1.3 Objectives

The objectives of this project include:

a) To develop a portable security system that can detect (intention of) illegal entry to a house or a building.

b) To safeguard our personal belongings, valuables and ourselves.

1.4 Scope of Study

The project focuses on designing a functioning portable security system that can detect the following:

1. Opening a door/window.

The majority of all entries are made through windows, especially windows that are left open offering a welcome sign to the intruder. When the burglars try to break into the house through the opening of a door or a window, the alarm will sound.

2. Transmission of alarm signals to the owner

Another focus of this project is the transmission of the alarm signals of the security system to the owner. When the alarm is triggered off, the system will send a text message to the user's mobile or police helpline to inform him/ her of the attempted burglaries.

1.5 Limitations of the Project

The limitations of the portable security system project are listed below:

(a) Resource constraint

Although most of the resources (e.g. components) for this project are available in the EE central store, but sometime the components needed might be out of stock and not all components can be gotten from there. Hence, we will need to find other alternatives such as buy it online, buy it from elsewhere, or look for other components to replace it. Then, this will lead to a delay to the project completion time and also the cost of the project – which form the next two constraints.

(b) Time constraint

The period for final year project is very limited which is two semesters. Thus, good time management and planning are crucial. A Gantt chart had been created in this case. The draft and design of the project must be completed within the time frame.

(c) Budget constraint

If given more financial help, better quality materials could be employed in this project.

1.6 Benefits of the Project

(a) To be able to protect our house/building from unwelcomed visitors.

(b) To be able to save cost on installation of the security system.

(c) It is a user friendly project since it does not require complicated wiring or installation.

1.7 Organisation of Report

This thesis is divided into five chapters. In Chapter 1, an introduction of project is presented along with the project objective, problem statement, scope of study, limitations and the benefits of this project. Chapter 2 begin with the literature review on the theories and facts that related to the project. This chapter also provides a review on the research of the components and software that are used in this project. Chapter 3 discusses the related works to this project. In Chapter 4, methodology and approach that used to develop this project are discussed. Chapter 5 discusses on the result and discussion of the project. The last chapter summarizes this project, discusses the limitation of this project and suggests possible recommendations for future works.

CHAPTER 2: LITERATURE REVIEW

2.1 Chapter Overview

This chapter discusses the theories and facts relating to the portable security system project.

2.2 Handphone Communication

There are 3 types of commands available for interfacing with handphone modem:

a) Fbus protocol
b) Mbus protocol
c) AT command

These commands can be used by the computer to control the handphone modem. The Fbus and Mbus protocols are developed by Nokia for its older generation of handphones such as the Nokia 3310 and 3315 while the AT command is the common command that is being used in most handphones nowadays.

2.2.1 Fbus protocol

Fbus is a completely bi-directional bus for both transmitting and receiving data from a phone. It is fast (115,200 bps) and high speed full duplex bus. Figure 2.1 shows the Fbus Connection in Nokia 3310.

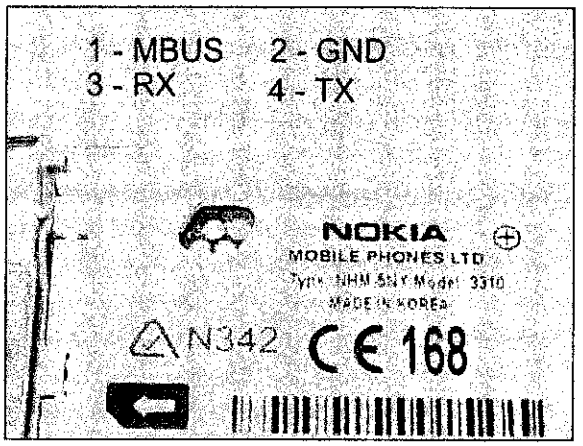


Figure 2.1 Fbus connection in Nokia 3310

An example below shows how Fbus communication in Nokia 3310 takes place [2]:

1) PC sends request to Nokia 3310.

   This sample frame is sent to get the hardware and software version from a Nokia phone.

   1E 00 0C D1 00 07 00 01 00 03 00 01 60 00 72 D5

   Byte 0 : F-bus Frame ID

   Byte 1 : Destination ID (Phone's Device ID byte)

   Byte 2 : Source Address (PC's device ID bype)

   Byte 3 : Command 0xD1 is Get HW & SW version

   Bytes 4 & 5: Message length

   Byte 6 – Byte 13: Data segments

   Byte 14 : Odd Checksum

   Byte 15 : Even Checksum

   The frame structure is as below:

1E	00	0C	Type	Len	Message	Sequence No	Padding byte	Checksum

2) Nokia 3310 replied with acknowledgement.

   1E 0C 00 7F 00 02 D1 00 CF 71

   The frame structure is as below:

1E	0C	00	Type	Len	Message	Sequence No	Padding byte	Checksum

   Notice that in the first line the destination ID and Source Address are swapped.

3) Nokia 3310 sends response to the request (from PC).

   1E 0C 00 D2 00 26 01 00 00 03 56 20 30 34 2E 34 35 0A 32 31 2D 30 36 2D 30 31 0A

   4E 48 4D 2D 35 0A 28 63 29 20 4E 4D

   50 2E 00 01 41 3F A4

   The message type D2 means Get HW & SW version request has been received.

   0x26 which means 38 byte message (highlighted in grey colour) shows:

   0x0003 "V" "firmware\n" "firmware date\n" "model\n" "(c)NMP."

4) PC sends acknowledgement

   1E 00 0C 7F 00 02 D2 01 C0 7C

   The frame structure is as below:

1E	00	0C	Type	Len	Message	Sequence No	Padding byte	Checksum

The PC sends ACK frame to the phone stating that the Message Type D2 is received.

2.2.2 *Mbus protocol*

Mbus is a one pin bi-directional bus for both transmitting and receiving data from phone. It is slow (9600 bps) and only half duplex. There is limited information on how to transmit data using Mbus Protocol; hence it is not discussed here.

2.2.3 *AT commands*

AT commands are a set of instructions used to control modem operations such as dialing, hanging up, and changing the parameters of the connection. The modem can be either a GSM modem or a GSM phone modem. AT commands is simply a string of characters preceded by the AT prefix that is sent to the modem. In this project, AT commands is only used to send out the SMS text messages. The AT commands can be written and tested through the hyper terminal.

AT commands has the following format:

- The command is prefixed with AT (Attention).
- The command is terminated by a carriage return <CR>.
- The commands can be entered in upper case or lower case.
- The AT prefix can be written in either upper case or lower case, but both the A and the T must be written in the same form.
- Characters that precede the AT prefix are ignored.
- Multiple commands can be strung together on a single line and spaces may be included between commands but are not necessary.

General Syntax of AT-Commands:

Basic AT<command> [=] [<parameter>]

Extended AT+<command>= [<parameter>]

Read command AT+<command>?

AT<command>?

Test command AT+<command>=?

Response command AT+<command> : <parameter >

mportant AT commands used to Test and Design:

Table 1: Important AT command

\TD	To dial a voice call from the modem
\TA	To answer an incoming call
\TH	To hang up a voice call
\T+CMGR	To read the message at particular location
\T+CMGS	To send the message
\T+CMGF	To change the message format to PDU or Text mode

:.3 SMS Mode

'here are 2 SMS modes that a GSM modem or mobile phone can operate in. They are called ;MS text mode and SMS PDU mode in which PDU stands for Protocol Data Unit.

'or PDU mode, the message to be sent has first to be converted into HEX code but we can just imply type out our message for the text mode.

}elow is a description of the main difference between SMS PDU mode and SMS text mode in ending text messages.

'or example, "It is easy to send text messages." to the mobile phone number +85291234567. In ;MS text mode, we should enter the following command:

{T+CMGS="+85291234567"<CR>It is easy to send text messages.<Ctrl+z>

However, for GSM modem or mobile phone that is operating in SMS PDU mode, the above command line does not apply. This is because the syntax of the +CMGS AT command is different in SMS PDU mode. To send the same message in SMS PDU mode, the message body, destination phone number header and some other headers are encoded in the hexadecimal sequence. Hence, the following command line should be used instead:

AT+CMGS=42<CR>07915892000000F001000B915892214365F700002149 3A283D0795C3F3
C88FE06CDCB6E32885EC6D341EDF27C1E3E97E72E<Ctrl+z>

The octets of the PDU message contain lots of information. In the above example, the PDU code is divided and different octets signify the following information:

07 Length of SMSC information.

91 Type of address of SMSC.

5892000000F0 Encoded Service center number.

01 First octet of SMS delivery message.

0B Address length of the sender message.

91 Type of address of the sender number.

5892214365F7 Sender number.(number is +85291234567).

00 TP-Protocol Identification Address.

00 TP-Data coding Scheme.

21493A283D 0795 TP-SCTS Time stamp.

C3 TP-User data length.

C88FE06CDCB6E32885EC6D341EDF27C1E3E97E7 Encoded Message "It is easy to send text messages".

2.4 Hardware and Theory

2.4.1 *PIC16F877A*

PIC16F877A microcontroller manufactured by Microchip is used as the brain of this project because it is cheap and easy to be assembled. This microcontroller has a 25 MHz processor, 33 input/output (I/O) pins, (8K*14words) of Enhanced FLASH program memory, (368*8bytes) of RAM, (256*8bytes) of data EEPROM. The PIC does not have an operating system and simply runs the program in its memory when it is turned on. It is used to control the output of the system according to the inputs. The program is loaded into it to be executed as desired. It can be reprogrammed and erased up to 10,000 times. Figure 2.1 shows the pin assignment for PIC16F877A. Different types of PIC series will have different pin assignments.

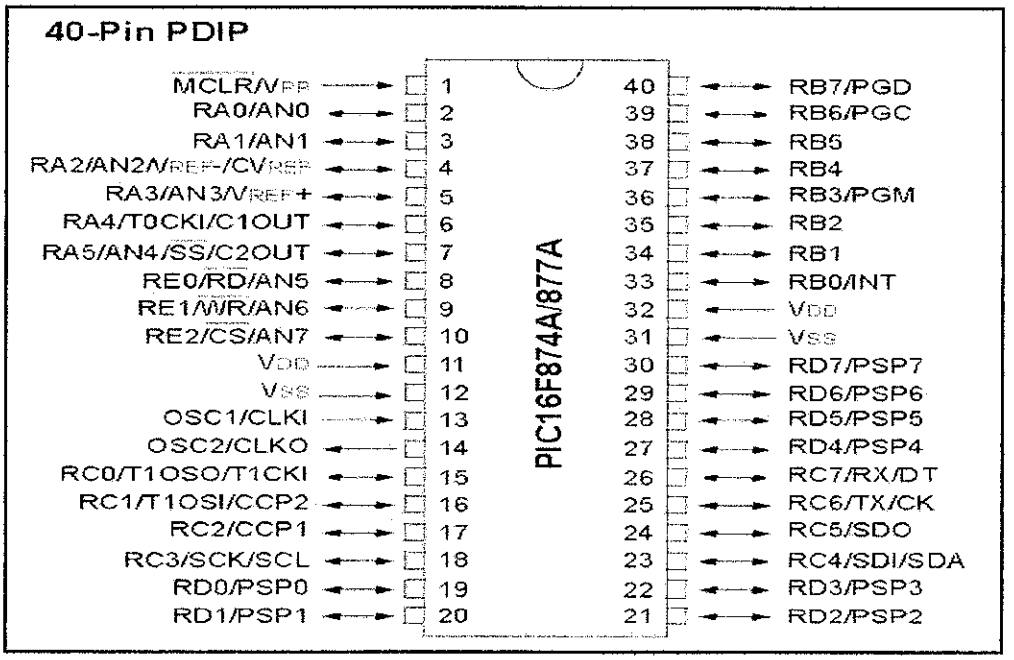


Figure 2.2 Pin Assignment for PIC 16F877A

2.4.2 *Window/Door magnetic sensor*

Magnetic sensors are the most common type of sensor used in security systems. They are often referred to as "contacts" or "switches." They are usually discretely placed on access points, such as windows or doors, so that they are not visible from the outside. Window/Door Magnetic sensors consist of two pieces that form a circuit and when separated, signal the control panel

that an alarm event has occurred. There are two types of magnetic sensors, Normally Closed (NC) or Normally Open (NO). In this project, we use a NC magnetic sensor. The specifications of a NC Magnetic Sensor are as follows:

- Operating distance: 18mm±5mm
- Switch withstood-voltage: 100VDC
- Load current: 500mA
- Life time: 1,000,000 times
- Switch output: NC

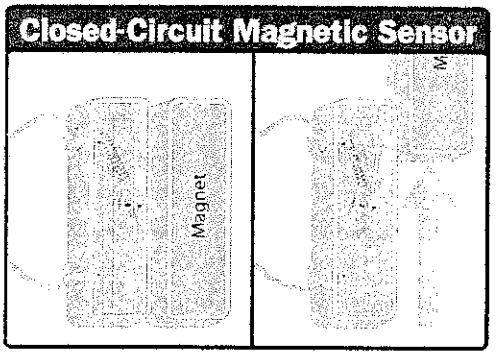


Figure 2.3 Magnetic Sensor

igure 2.3 above shows how a NC magnetic sensor works. When the door/window is closed, ıe electric circuit is closed. This means that as long as the door is closed, electricity can flow ·om one end of the circuit to the other. But if somebody opens the door/window, the circuit is pened, and electricity cannot flow. This triggers an alarm.

.4.3 Voltage regulator

he operating power supply for the PIC microcontroller and sensors are in the range of +4 volts V) to +5 volts (V) DC . Hence, a voltage regulator 7805 IC is used to regulate a 9-volt input ɔwer supply to a 5-volt power supply for the microcontroller and sensor to prevent the ɔmponents and circuits from being overloaded. Figure 2.4 below shows a pinout of the 7805 IC. rom the front side of the IC, we can see three legs or pins. The left pin must be connected to 12- ɔlt DC. The centre pin must be connected to ground. The right pin is the 5-Volt output voltage. heat sink is required for heat dissipation and can be mounted at the back of the IC.

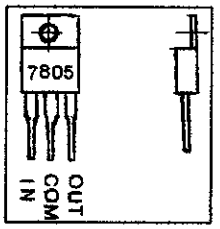


Figure 2.4 Voltage Regulator

There will be noise occurring when regulating the voltage. Therefore capacitors need to be added to get an accurate +5V. Figure 2.5 shows the regulator circuit to step down the voltage source.

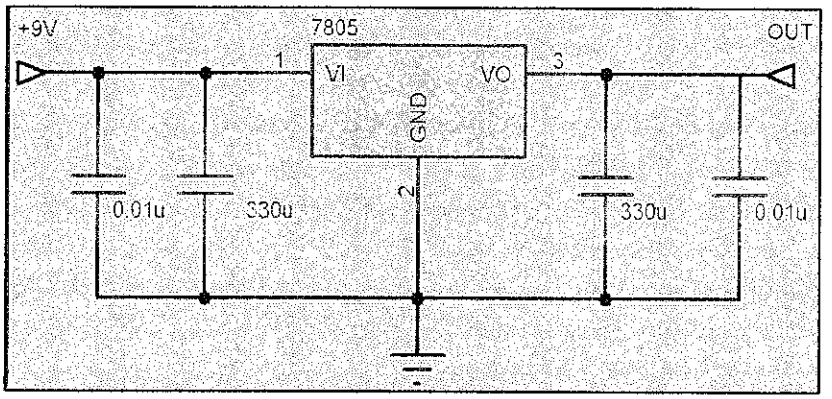


Figure 2.5 Voltage Regulator Circuit

.4.4 Mobile phone (Sony Ericson T630)

n this project, a mobile phone was used as a GSM Modem. Sony Ericsson T630 was chosen. 'his mobile phone has been chosen because it can support both SMS modes - either PDU mode r Text mode. Moreover, this mobile phone is supported with GSM. The mobile phone can be ontrolled and customized in various levels by using the standard AT Command.

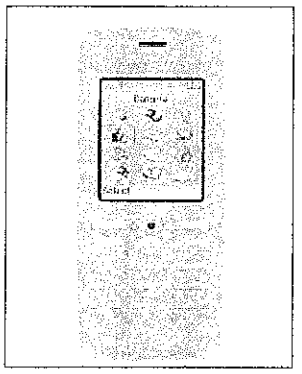


Figure 2.6 Sony Ericsson T630

2.4.5 MAX232

The MAX232 device is a dual driver/receiver. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels while each driver converts TTL/CMOS input levels into EIA-232 levels. It is used widely for level shifting. This IC plays an important role whenever sending data serially to a PC which requires voltage levels as per RS232 standard. Figure 2.7 below shows the MAX 232 diagram.

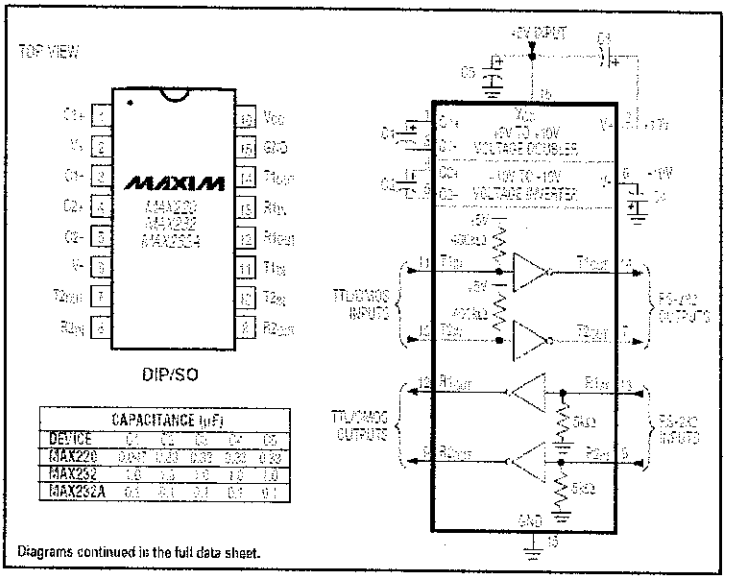


Figure 2.7 MAX 232 Diagram

2.4.6 *RS232*

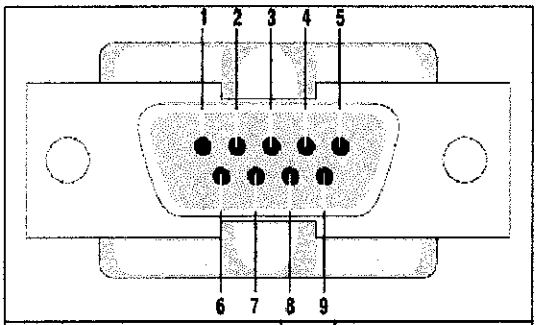
RS232 (Recommended Standard 232) is an electrical signalling specification published by the Electronic Industries Association (EIA). RS232 is often used in computer serial ports and modems. The 9-pin (DB9) connector, with specific pin assignments, is commonly accepted as the RS232 connector or the serial connector."

The general voltage assignment is:

Signal = 0 (LOW) > +3.0V

Signal = 1 (HIGH) < -3.0V

Figure 2.8 shows the RS-232 signals used and pin assignments in a DB 9 Male connector.



Pin	Signal	Pin	Signal
1	Data Carrier Detect	6	Data Set Ready
2	Received Data	7	Request to Send
3	Transmitted Data	8	Clear to Send
4	Data Terminal Ready	9	Ring Indicator
5	Signal Ground		

Figure 2.8 RS-232 DB9 Male Pinout

In this project, only Pins 2, 3 and 5 of the DB9 connector are used while the other pins are ignored.

2.4.7 USB to serial converter cable

Modern computers usually are without serial ports and have USB ports instead. This converter will plug into a computer's USB port and provide a COM port for devices that need a serial port. In this project, we used this USB to Serial Converter Cable to provide COM port and also to allow serial communication between PC and other devices such as microcontroller and mobile phone.

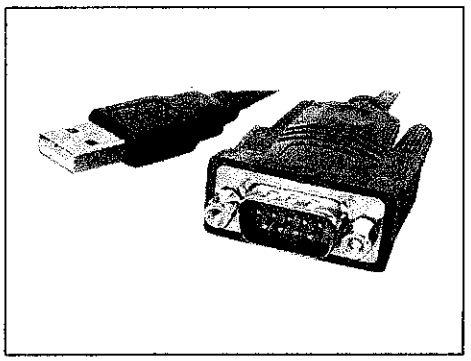


Figure 2.9 USB to Serial Converter Cable

2.4.8 Cytron PIC programmer (UIC00A & UIC-S)

UIC00A is a user friendly PIC USB Programmer, it allow an user to quickly program and debug the source code while the target PIC is on the development board. UIC00A is designed to be plug and play with USB connection. This programmer obtains it power directly from the USB connection, thus no external power supply is required, making it a truly portable programmer. Below are the procedures to program a microcontroller using UIC00A and UIC-S:

By connecting UIC00A to USB port of a computer and another end with optional socket (UIC-S) to program 18 pins, 28 pins and 40 pins PIC Microcontrollers, at which the PIC microcontroller is being located in UIC-S, and then running PICkit 2 programming software to program the PIC microcontroller.

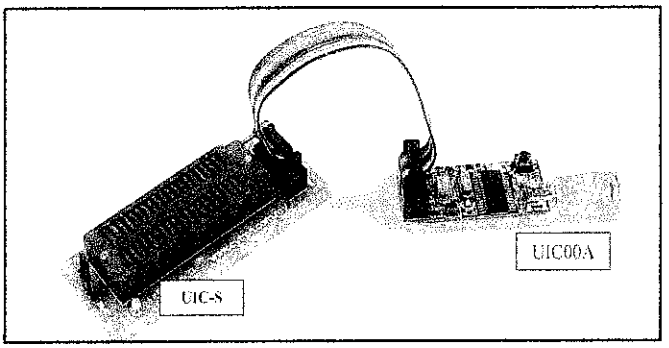


Figure 2.10 Cytron PIC Programmer (UIC00A & UIC-S)

.5 Software and Theory

.5.1 *MPLAB IDE*

MPLAB IDE is a software program that runs on a PC to develop applications for Micro-chip microcontrollers. It is called an Integrated Development Environment, or IDE, because it provides a single integrated "environment" to develop code for embedded microcontrollers.

MPLAB can be used for either Assembly program or C program. In this project, we use MPLAB for a C program. We can either create our own C code as source file via MPLAB or directly insert the C code into the file. The C code will be BUILT (command in MPLAB) to compile the program. If the BUILD is successful, then a hex file for the program will be generated. The hex file generated will be used for PIC programming via PICkit 2 programmer software. Else, we can debug the program or check the validation of the program via simulation mode.

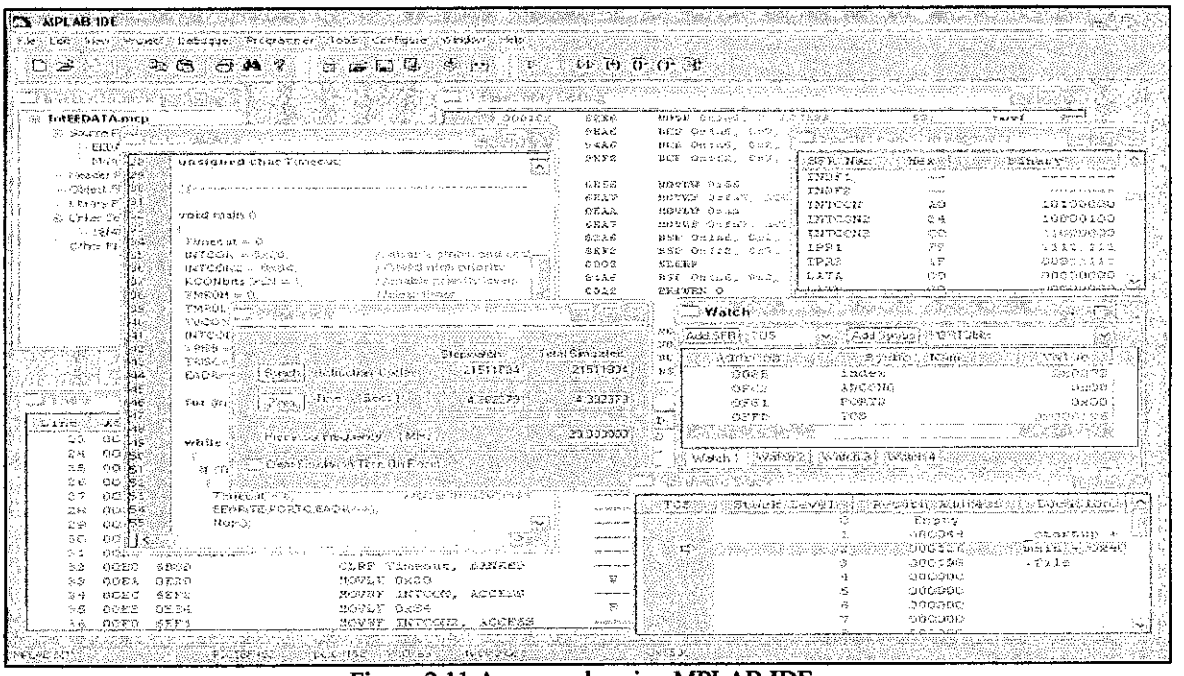


Figure 2.11 An example using MPLAB IDE

.5.2 *PICkit 2 programmer software*

PICkit 2 is programmer software that is used to programs all the current 8-, 16-, and 32-bit Flash PIC microcontrollers. It is easy to use with MPLAB® IDE and other development environments.By using an appropriate programmer tool (e.g. UIC-S), the hex file generated with MPLAB IDE is being imported and then written (using WRITE command) into the appropriate target device, the PIC microcontroller.

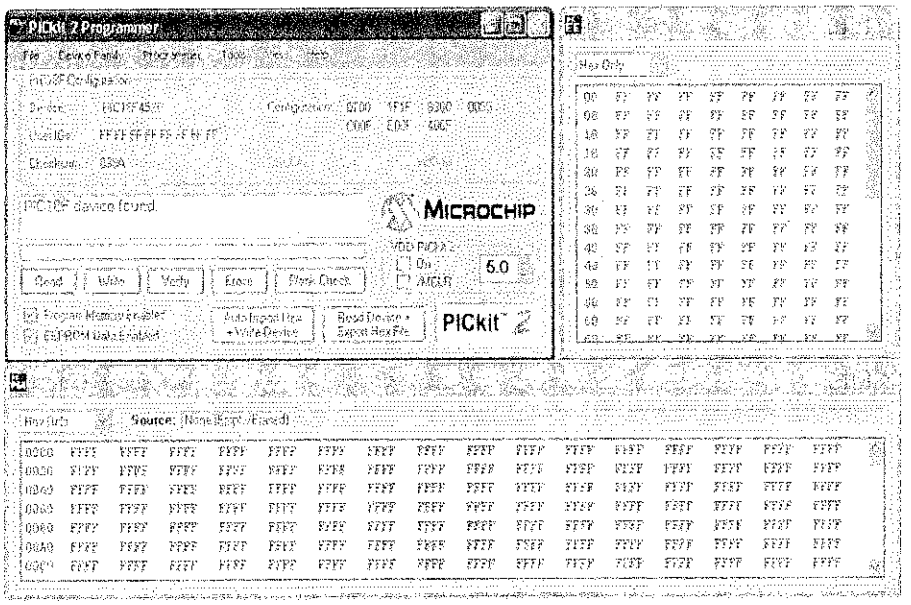


Figure 2.12 PICkit 2 Programmer

2.5.3 *EAGLE*

EAGLE, which stands for Easily Applicable Graphical Layout Editor is a schematic capture and PCB layout tool. It is being used to design the electronic schematics and a printout circuit board (PCB) layout of this project. Using this software tool, we are able to turn the drawn schematics into a PCB design. EAGLE consists of a schematic editor, PCB layout editor and auto-routing module. The schematic editor is for designing circuit diagrams, while the PCB layout editor is for allowing back annotation to the schematic, and auto-routing to automatically connecting traces based on the connections defined in the schematic.

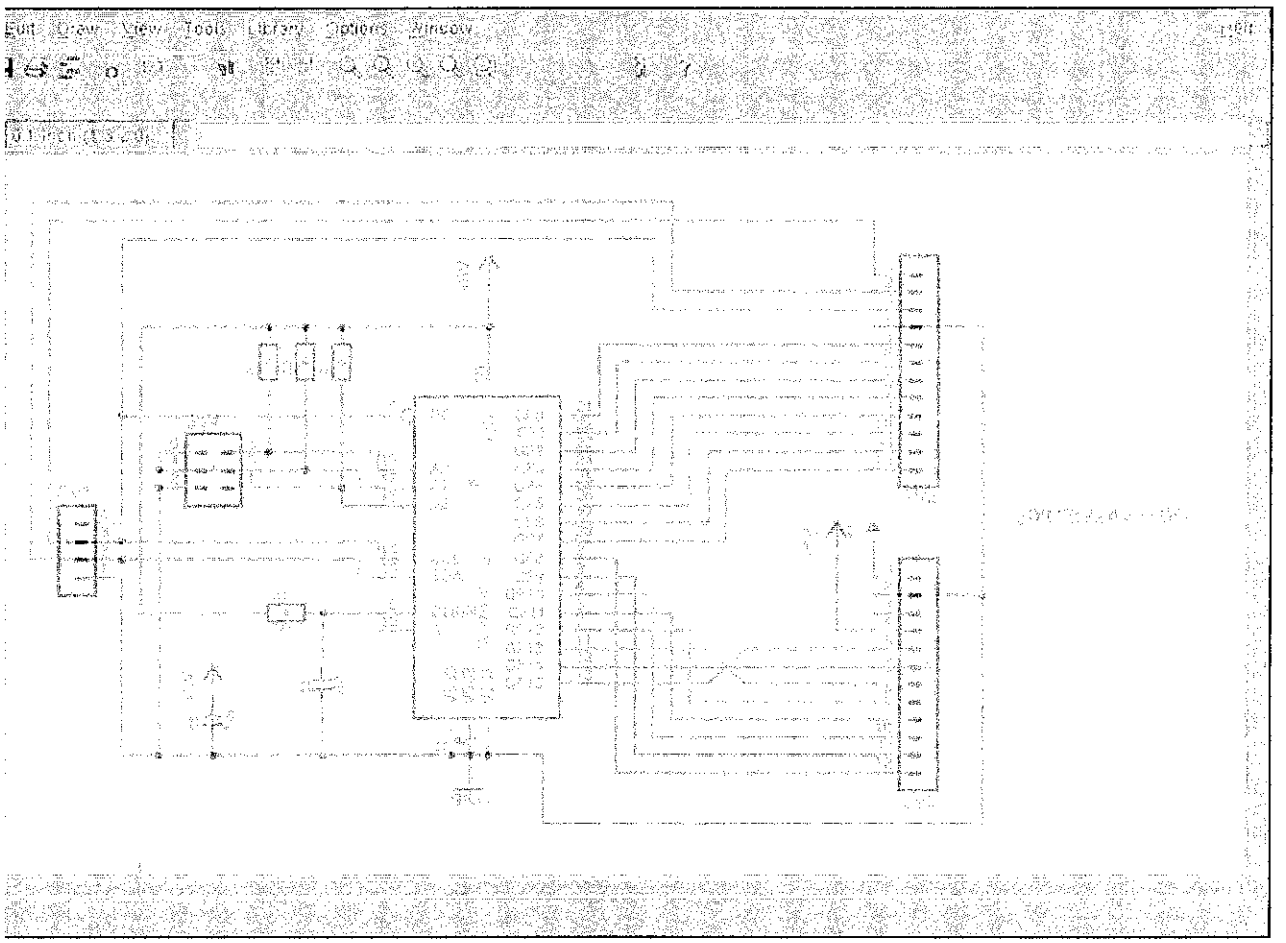


Figure 2.13 An example Using Eagle Software

CHAPTER 3: RELATED WORKS

. Chapter Overview

This chapter discusses projects and paper works related to this project. These related ırks have been reviewed carefully in order to improve the quality and reliability of this project. analyzing the projects developed by other researchers, there is a possibility to know what ıtures are lacking in their projects. Moreover, there are some useful ideas that can be plemented in this project from other similar projects.

! Previous Projects

?.1 *Automatic phone-alert home security system [3]*

ıe project is about designing an automated phone system that, when triggered by an electronic ısor, it initiates a phone call and plays a notification message. The device is compatible with y electronic sensing detector, such as an infra-red detector, motion sensor or an ultrasound tector, and is adaptable to the existing telephone line at home.

; the selected sensor is triggered, a pre-stored number is dialed and a voice message is played, tifying which detector has been activated. The owner or designated contact dials a verification de, in this case the star "*" button and the message is played. If the call is unanswered or the rification is incorrect, the system will continue to make further attempts to call until the essage is played.

ır better security purposes, it is good to have a verification code as we can prevent non-owner pick up the calls without permission but in terms of emergency, it is not practical to have the rification code; the system consumes time to verify the code before playing the message. In signing this project, a system without verification code that will be convenient to the owner is ınsidered.

?.2 *To handphone burglar alarm using PIC 16F877A microcontroller [4]*

his project is similar to the previous project stated in Subsection 3.2.1. But this project focused ı the generation of DTMF (Dual Tone Multi Frequency) tones using a PIC Microcontroller, ccompanied by the R-2R Ladder Network as the Digital to Analog Converter (DAC), and the ɔrresponding circuit in order to inform the owners about intrusion through phone calls.

/hen the sensor is triggered, the sensors sent the information to PIC Microcontroller and the IC will transfer the data to R2R Ladder Network; the R2R will then convert the data into 'TMF tones and an automatic phone call will be received by the owner. The system or circuit ı the house phone will recognize the pressing of the keypad button and generate the different 'TMF tones. The sounds used for touch tone dialing are referred to as DTMF (Dual Tone Iultiple Frequencies) tones. Each number on a phone keypad (as well as the "#" and "*") is epresented by a pair of tones. DTMF is associated with digital telephony, and provides two elected output frequencies (one high band, one low band) for a duration of 100 ms. DTMF eneration consists of selecting and combining two audio tone frequencies associated with the ɔws (low band frequency) and columns (high band frequency) of a pushbutton touch tone elephone keypad. The matrix for selecting the high and low band frequencies associated with ach key is shown in Figure 3.1.

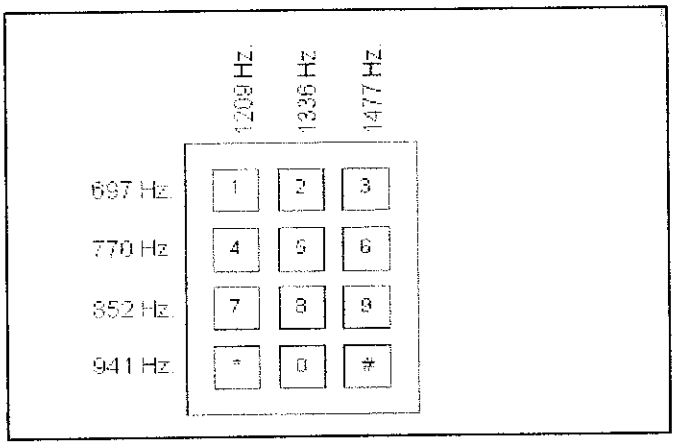


Figure 3.1 The DTMF Frequencies and Corresponding Keys

order to generate the DTMF tone for "1", a pure 697 Hz signal will be mixed with a pure 09 Hz signal (see Fig 3.2) and so on.

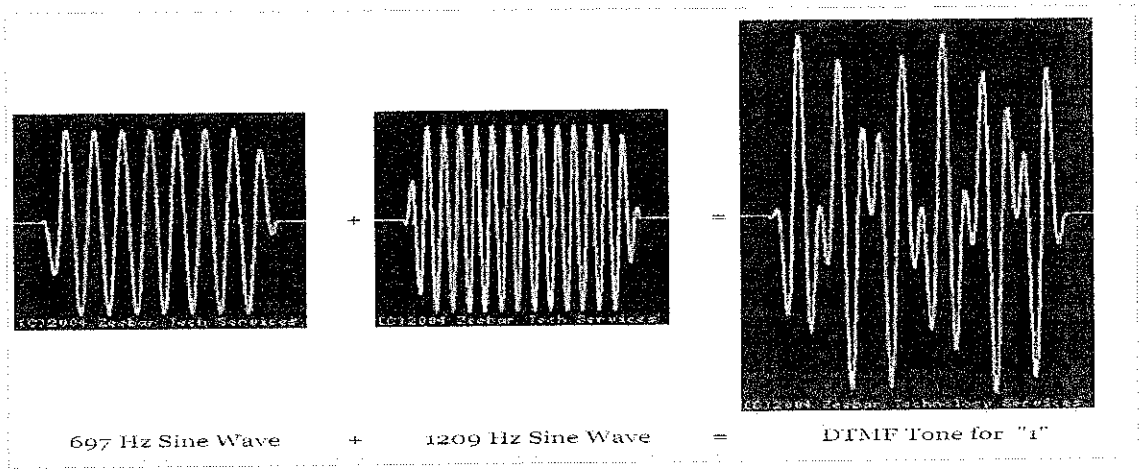


Figure 3.2 Two Pure Sine Waves Combined to Form The DTMF Tone for "1"

this project, modification had been done on house phone by adding an extra wired connection ;tween the house phone and the PIC. The house phone is only applicable at home; hence other ternatives such as cell phone can be considered to replace the house phone as our project main)jective is portability. Moreover, the concept of using a DTMF tone is simple and easy to be ıplemented. But a further research has to be performed; we have to think of the possibility of)ise or other factors that can affect and influence the generation of the DTMF tone.

2.3 Home security system by Chun-Pai Jimmy Hsieh, Yang Cao [5]

This project built a wired home security system using different types of sensors. Besides the nagnetic switch used at the doors and windows, this project also utilized a temperature sensor, moke detectors, and a motion sensor. Thus, the security system will sound the siren for two)urposes namely when there are intruders or when there is smoke or fire. This product had been :quipped with a voice playback chip which will speak when there is alarm to indicate which :ensor contributes to the alarm.

This project is built without considering how the owner of the system could switch off the ;ystem when he/she enters the armed house from outside. This is because the main control unit)f the system is attached inside the house. This central processing unit of the system will input

l the sensors and keypad while the outputs indicate zones and voice alarm. Furthermore, a ired system like this project could not provide portability to operate the system. Thus in wired /stem, the project designer will probably set a delay of around 15 seconds in triggering the arm at the front door in order to provide some time for the owner to disarm the system. The ren will sound if the owner is unable to turn off the system within the given time frame and iis will be inconvenient.

ı designing our project, a wireless system which will be convenient to the owner is considered. lence the system is built with a wireless remote control which has high portability and will ease ıe owner to bring it along when he/she is out of the house. This is because the owner can isarm the system before entering the house and no need to rush into the house to disarm the ystem.

2.4 *A smoke detection system using wireless network [6]*

iis smoke detector project is mainly designed for the hearing impaired. Besides that, this oject can be implemented when there is nobody in a particular place while the alarm for the ıoke detector is sounded. In this paper, the smoke detector will connect to a personal computer C) in order to control the GSM (Global System for Mobile Communications) module/phone to nd SMS (short message Service) from the detection place and receive SMS from the victim.

iis project is conducted using a laptop which has to be in online status all the time for the onitoring purpose. Although the laptop can be easily available, it is not practical to keep the ptop online all the time. This is because it could contribute to the hang up of the operating stem whereby it cannot send signal to the GSM device. Thus, it is recommended to use a icrocontroller such as a Peripheral Interface Controller (PIC) instead of PC as the control unit. y considering this, this project is designed by using a PIC microcontroller as a main device to ıntrol the inputs and outputs. Therefore, it is more convenient to use a PIC that to use a laptop hich consumes a lot more power.

CHAPTER 4: METHODOLOGY

Chapter Overview

is chapter discusses the methodology and approaches that being used in the project. The ıeline of the project are discussed too.

Project Activities

In order to achieve the project's objective, there are various steps that need to be executed. The procedures involved are shown as follows:

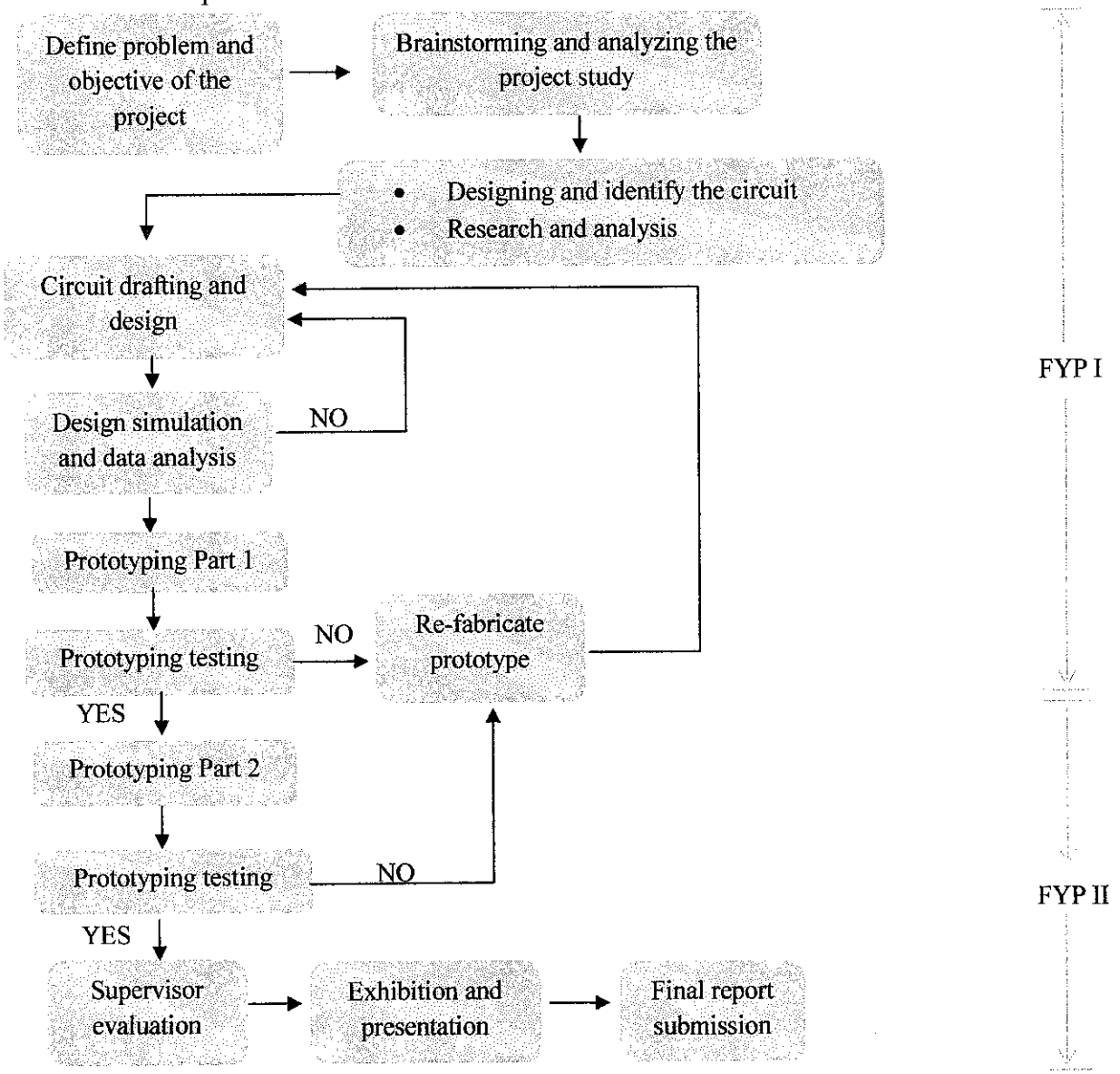


Figure 4.1 Project Activities

Gantt Chart

YP 1

Table 2 Gantt Chart for FYP 1

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
oject title selection														
oblem statement and objective ntification														
ckground study and literature view														
rther research and study														
ntify and purchase required uipments and materials, cost timation														
rcuit drafting and design														
ototype construction Part 1 and ototype testing														
eport compilation														

YP 2

Table 3 Gantt Chart for FYP 2

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
terature review (for prototype uilding)														
ototype building														
ototype testing														
mulation, testing and results														
ecommendation and odification														
eparation for exhibition and esentation														
nal report compilation and esentation														

Tools and Equipment Used

e equipments used throughout the experiment can be categorized into the following:

a) Hardware

Table 4 Components/Hardware Used

Components or hardware	Functions/purposes
PIC Microcontroller 16F877A	Acts as the centre of control and is programmed
PIR sensor	Acts as an input to detects the motion of the surroundings
Magnetic sensor	Acts as a input sensor which will be attached to door or window
Buzzer	Acts as alarm that will triggered when the sensors are on
Reset Switch	Used as the switch to turn off/reset the alarm
LED	Acts as an indicator when the alarm is triggered
DC Supply	Used as the main power source to activate the system
Crystal Oscillator	Provides stable clock signal
Voltage Regulator	Maintain a constant voltage level
Resistor	Restricts the flow of electric current to avoid short circuit
Capacitors	Placed in series with crystal oscillator to reduce DC bias
Diode 1N4007	Allows electrical current to flow in one direction
UIC00A & UIC-S	Programmer tool

b) Software

Table 5 Software Used

Software	Functions/purposes
MPLAB	Build the hex file for the C code
PICKIT 2	Import the hex file and program the PIC microcontroller
EAGLE	Design the electronic circuit for the project

; **Key milestone**

Figure 4.2 showed the draft for the overall system design.

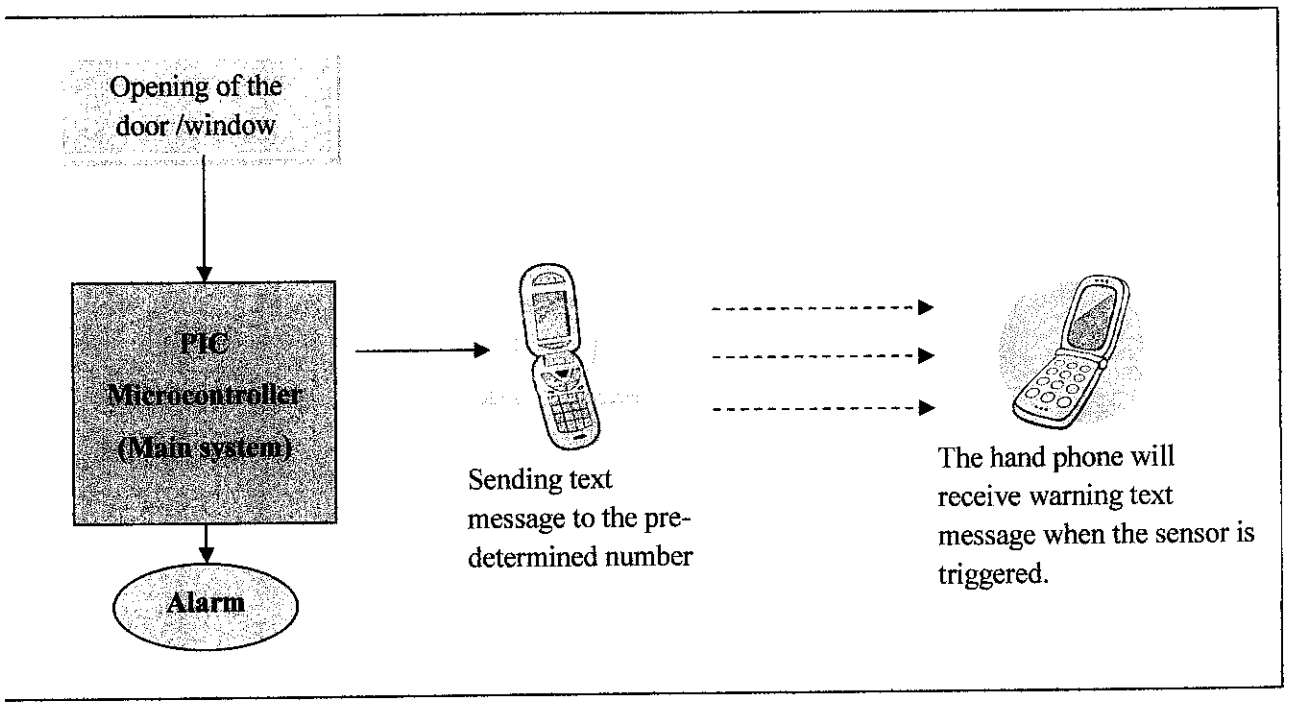


Figure 4.2 Overall System Design for The Project

6 Interfacing for the System

6.1 Interfacing USB to Serial Converter with PC

this project, we need to use a USB to Serial Converter Cable because the laptop used for sting purposes does not have a Serial Port.

1) First, connect the USB to Serial Converter Cable to the PC. Install the driver CD that is provided.

2) After finishing the installation, we need to check which COM Port is used by the PC to interface with the USB to Serial Converter Cable.

3) Go to Start→Control Panel. Look for System, and double click on it.

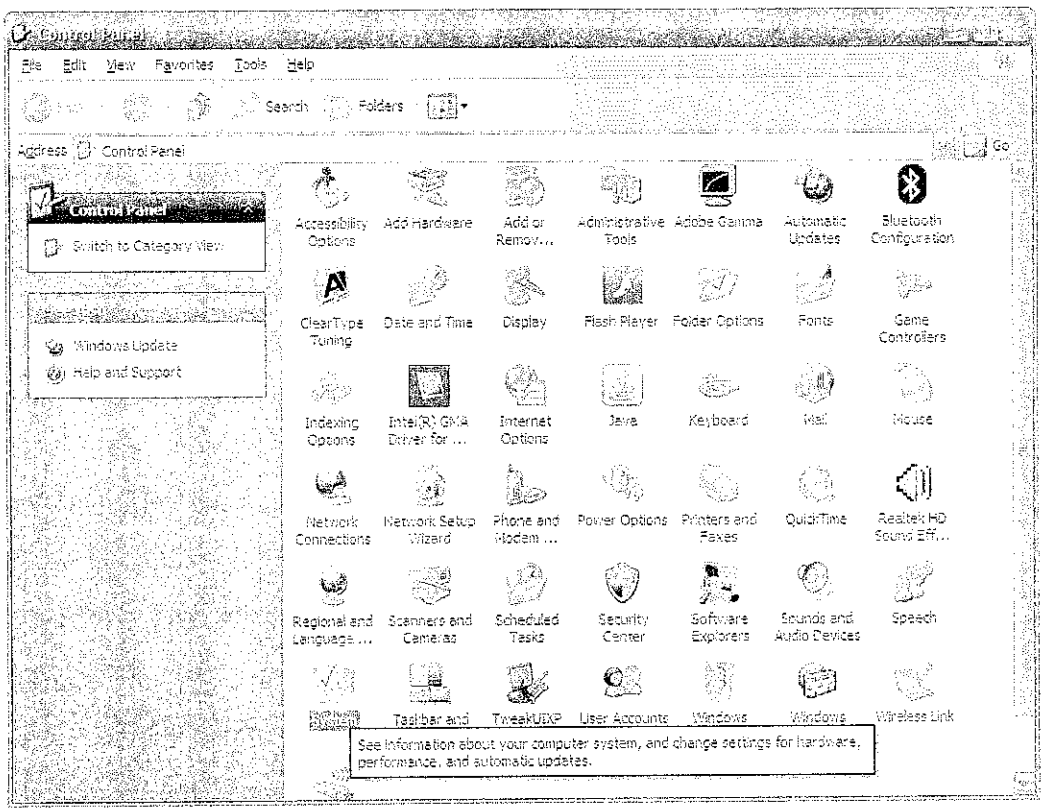


Figure 4.3 Control Panel

4) On the System Properties dialog Box, go to Hardware → Device Manager.

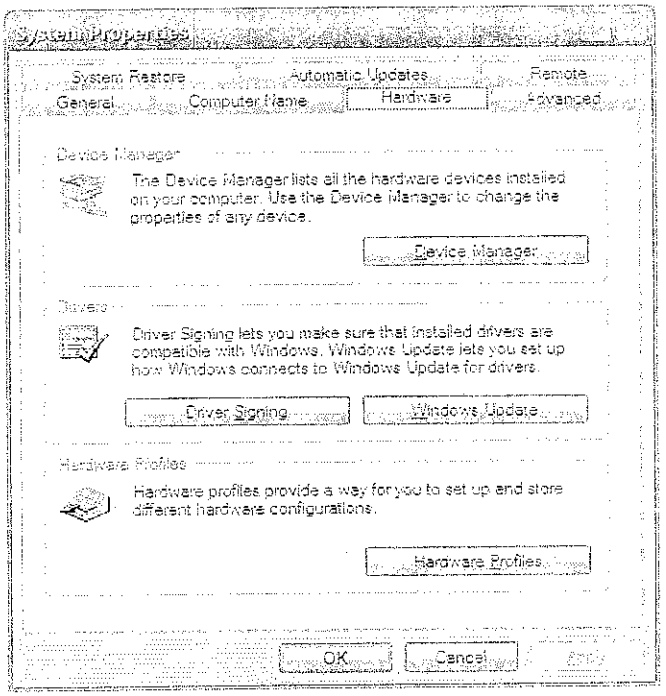


Figure 4.4 System Properties Dialog Box

5) Under Device Manager, look for Ports (COM & LPT). From here, we can check for the COM Port that the PC uses to interface with other devices. In our case, the PC used COM Port to interface with the USB to Serial Converter Cable.

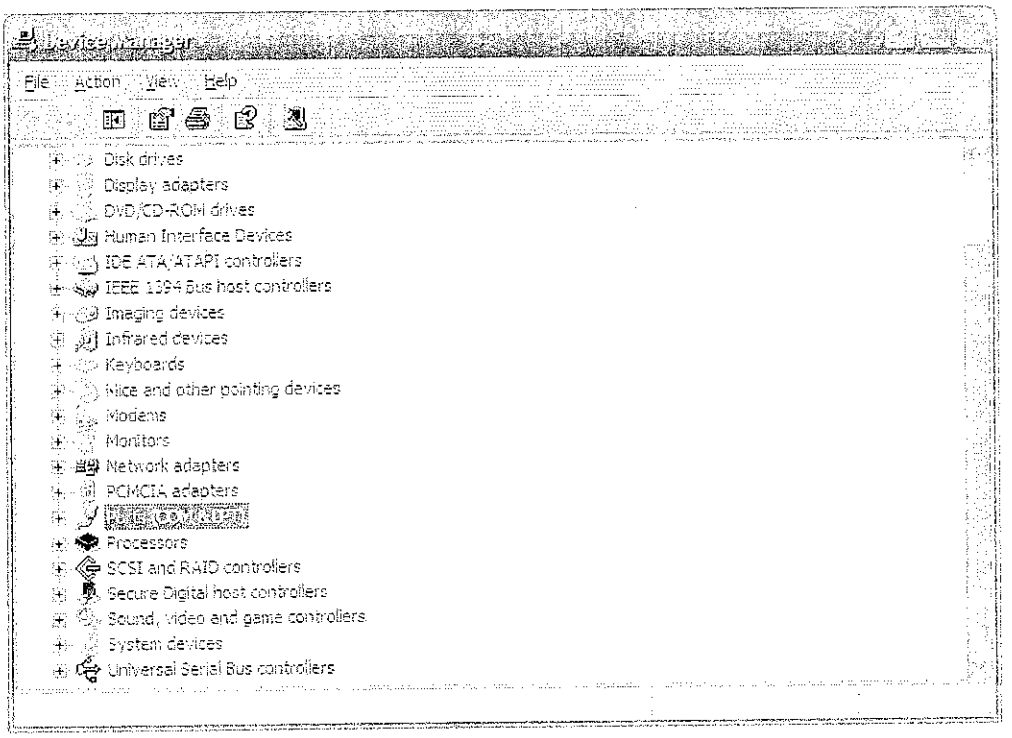


Figure 4.5 Device Manager

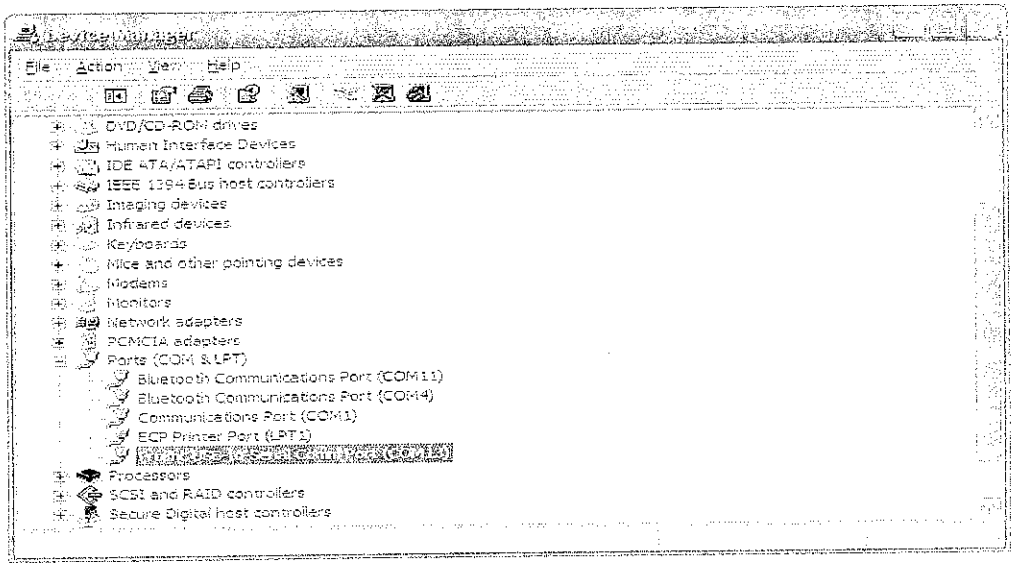


Figure 4.6 COMPort for USB to Serial Converter Cable

5.2 Interfacing PC with handphone

general, there are three methods to interface a computer / PC to a mobile phone:

(a) Via Bluetooth connection

(b) Via a Serial Port cable

(c) Via a USB Cable

ethod (b) and Method (c) are the same as the previous part on Interfacing USB to Serial nverter with PC (subsection 4.6.1), hence it is not discussed here.

Via Bluetooth Connection

1) First, turn on the Bluetooth for both the mobile phone and PC.
2) Next, right click the Bluetooth icon on your desktop, select Add a Bluetooth Device.

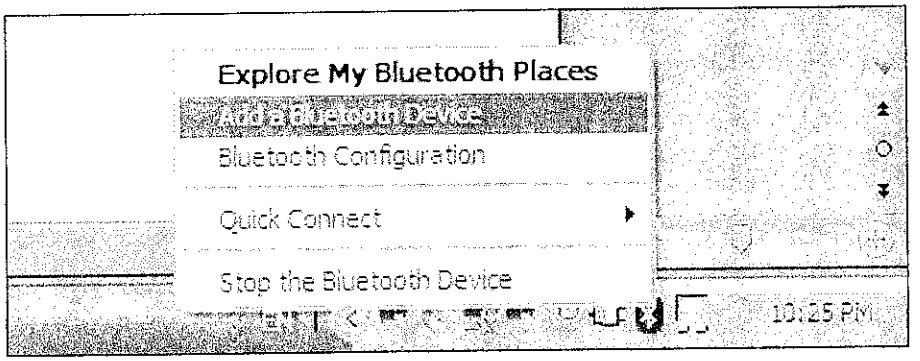


Figure 4.7 Add a Bluetooth Device

ne figure below will then popped up. Click on Next.

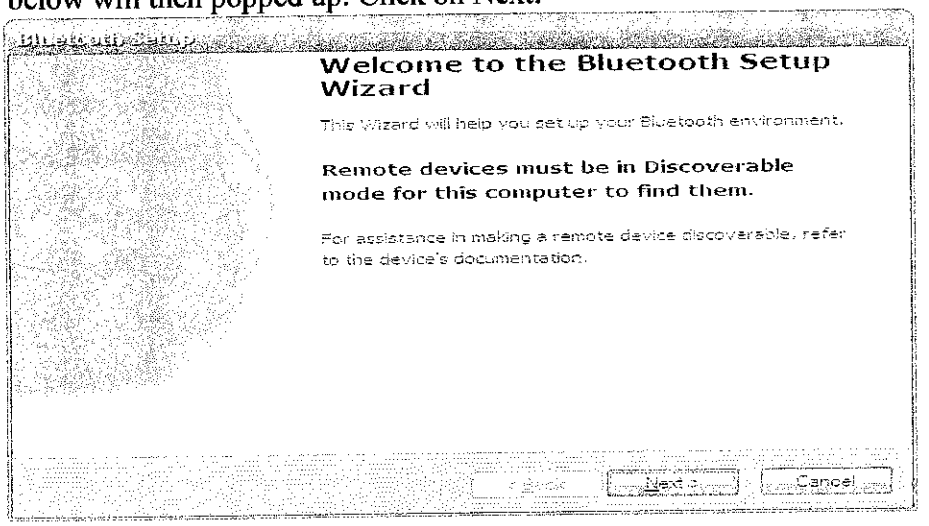


Figure 4.8 Bluetooth Setup Wizard

3) Select the correct Bluetooth device that you wish to interface with PC. Then, click Next.

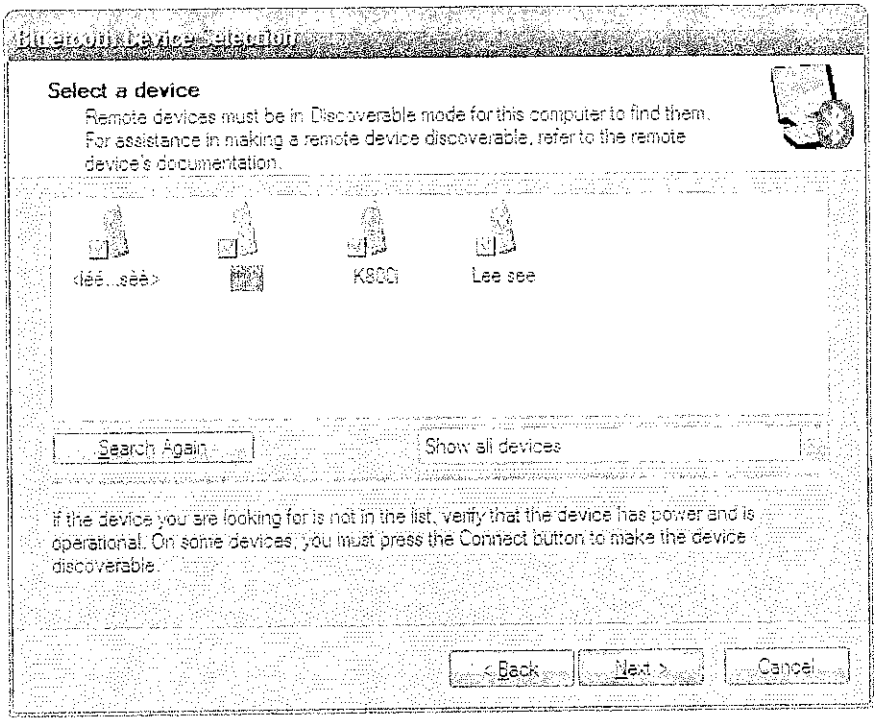


Figure 4.9 Bluetooth Device Selection

4) Select the service that we want to interface with the handphone. In this case, select Serial Port 1 and click Configure.

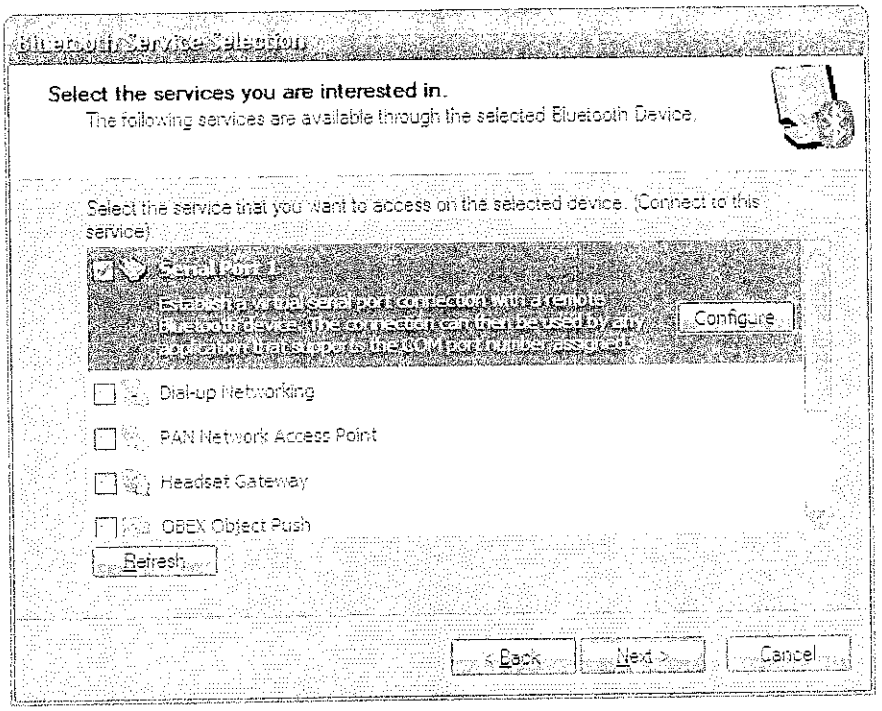


Figure 4.10 Bluetooth Service Selection

5) The following dialog box will appear. Now, we know that the PC is interfacing with the Bluetooth device (Mobile Phone) using COM Port – COM11.

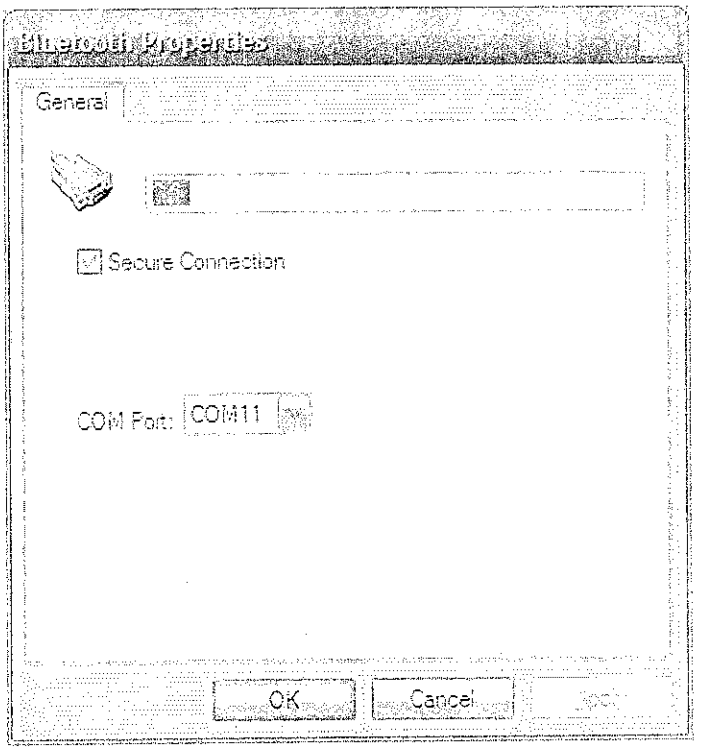


Figure 4.11 Bluetooth Properties

6) Click on OK in the Bluetooth Properties dialog Box. Then, Click next on the Bluetooth Service Selection Dialog Box.

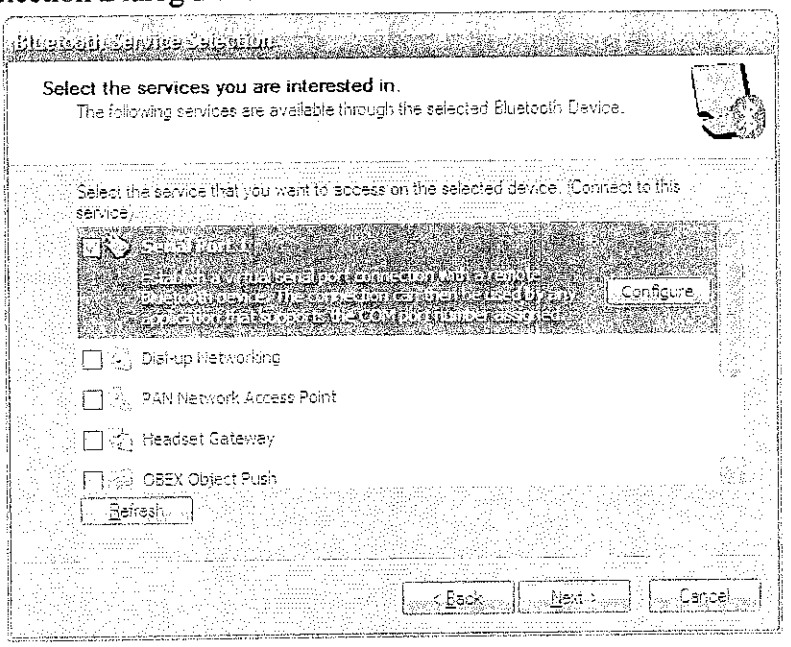


Figure 4.12 Bluetooth Service Selection

7) Click Finish on the Bluetooth Setup Wizard Completion Page. We havenow successfully interfaced the PC and Mobile Phone.

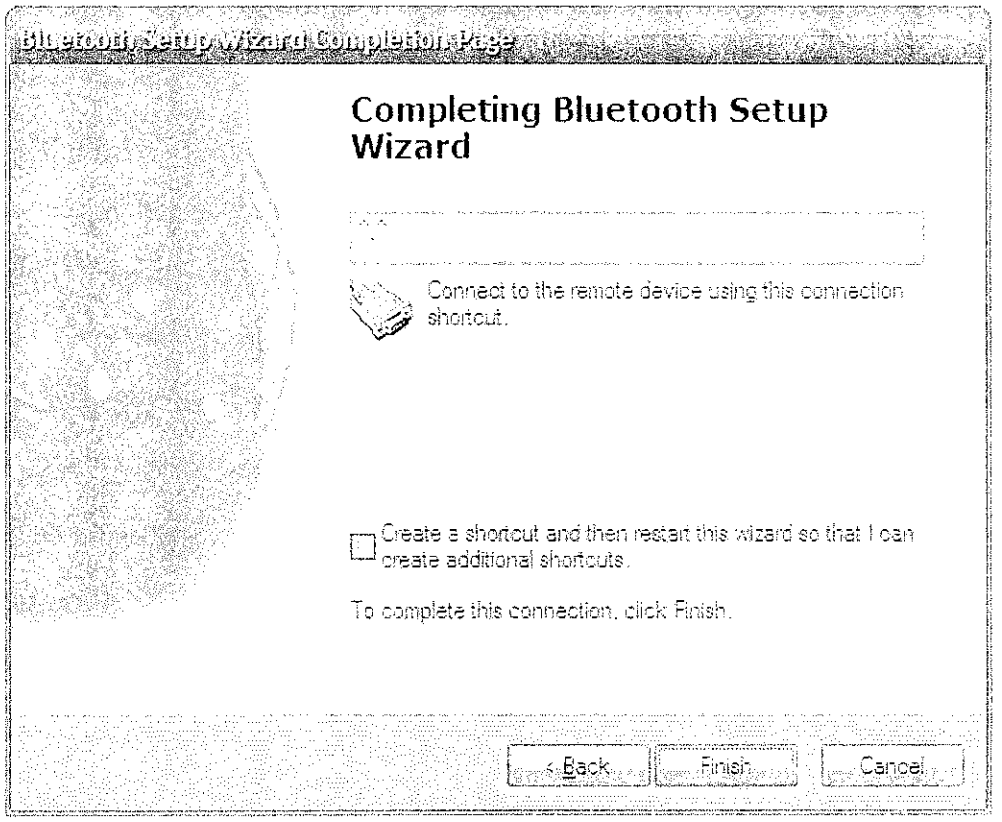


Figure 4.13 Complete Bluetooth Setup Wizard

CHAPTER 5 : RESULTS AND DISCUSSION

1 Chapter Overview

This chapter includes the prototype testing of the project. The results of this project are explained to. In addition, the challenges faced in this project are discussed.

2 The Block Diagram for this Project

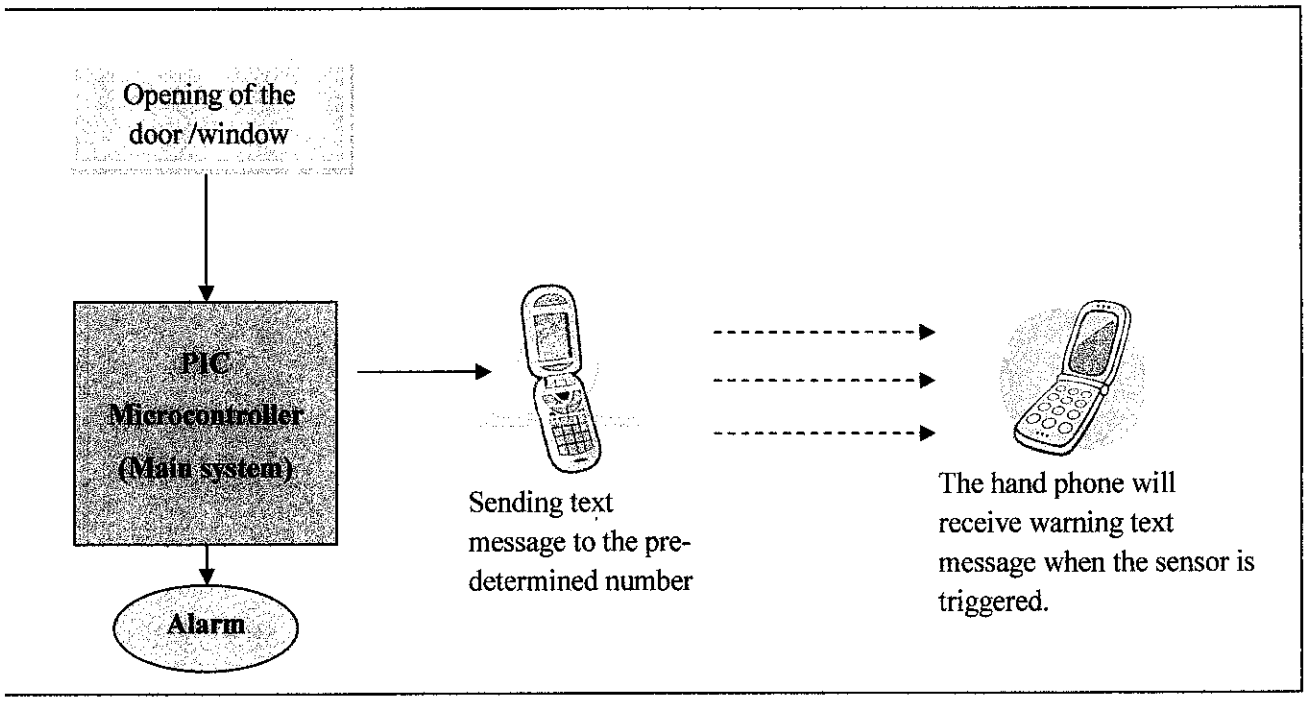


Figure 5.1 Block Diagram for the Project

this project, the expected result is as below:

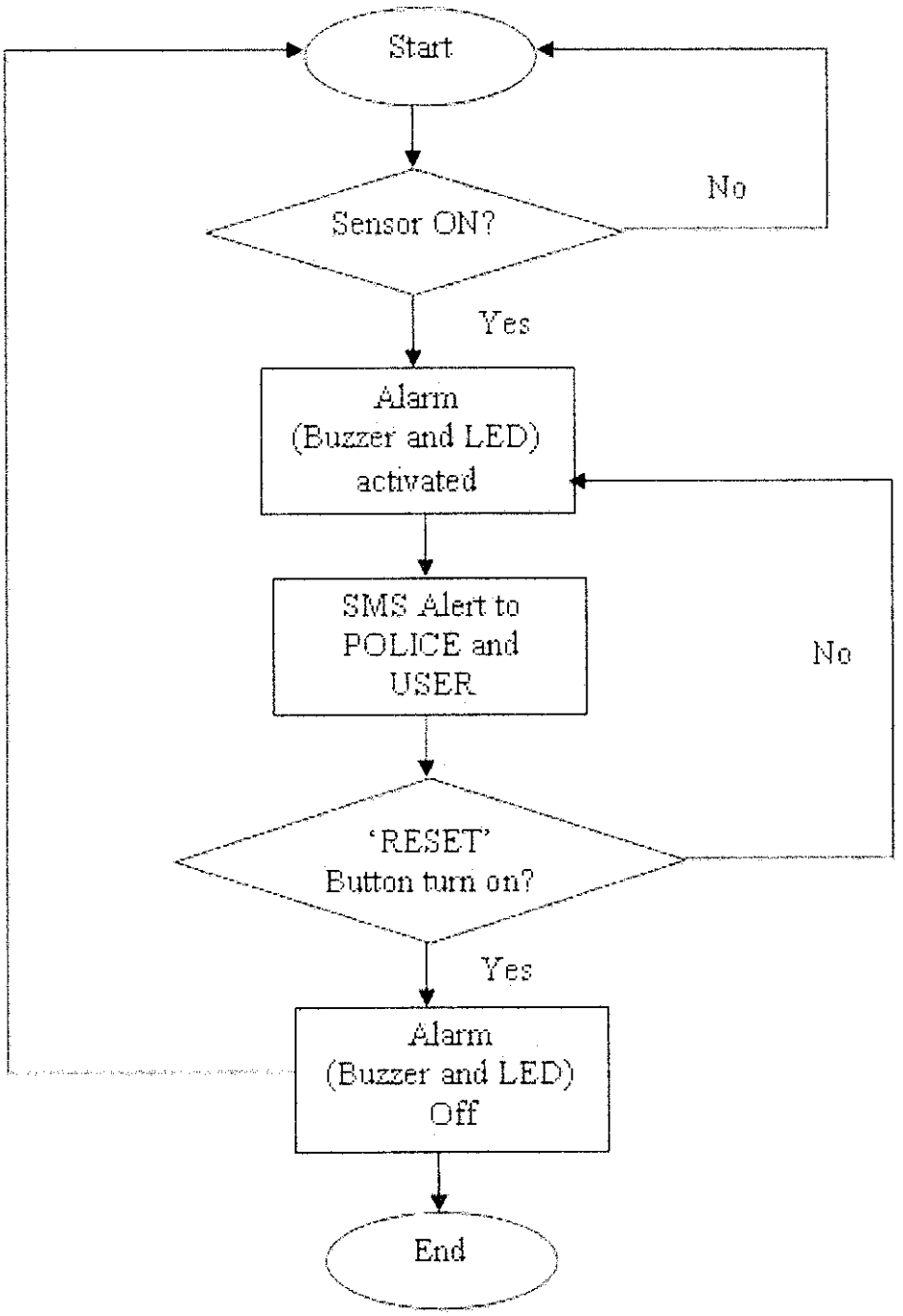


Figure 5.2 Flow Chart for the Project

control a handphone through a programmed microcontroller, we need a Serial Cable for Sony
icsson T630. Unfortunately, the cable is currently out of stock in Malaysia.

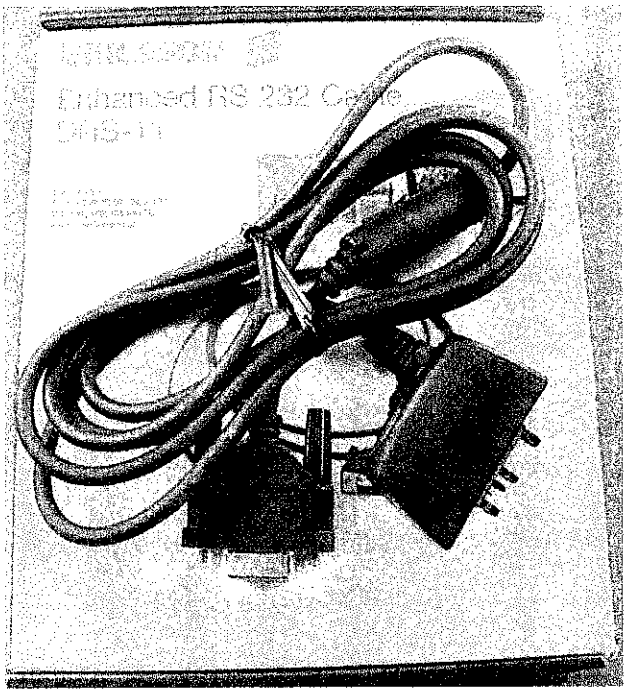


Figure 5.3 Serial Cable for Sony Ericsson T630

ince, the completed project is divided into 2 parts:

rt 1 : Interfacing PIC16F877A microcontroller with PC

Setting Up HyperTerminal and using it to communicate with PIC16F877A

rt 2 : Sending SMS Messages from a PC

ie assumptions being made here is:

- The existence of PC is to replace the unavailability of Serial Cable for Sony Ericsson T630 (as shown in Figure 5.3).

3 Interfacing PIC16F877A microcontroller with PC

ɜure 5.1 below shows the connection from the PC to the microcontroller:

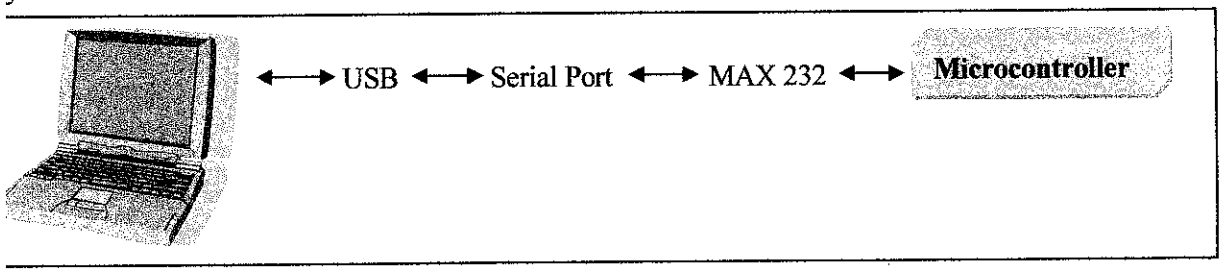


Figure 5.4 Connection from PC to Mircocontroller

the earlier chapter 4.5.1, we discussed about interfacing the USB to Serial Port Cable to PC.

ɔw, we continued on with the interfacing of the microcontroller with PC.

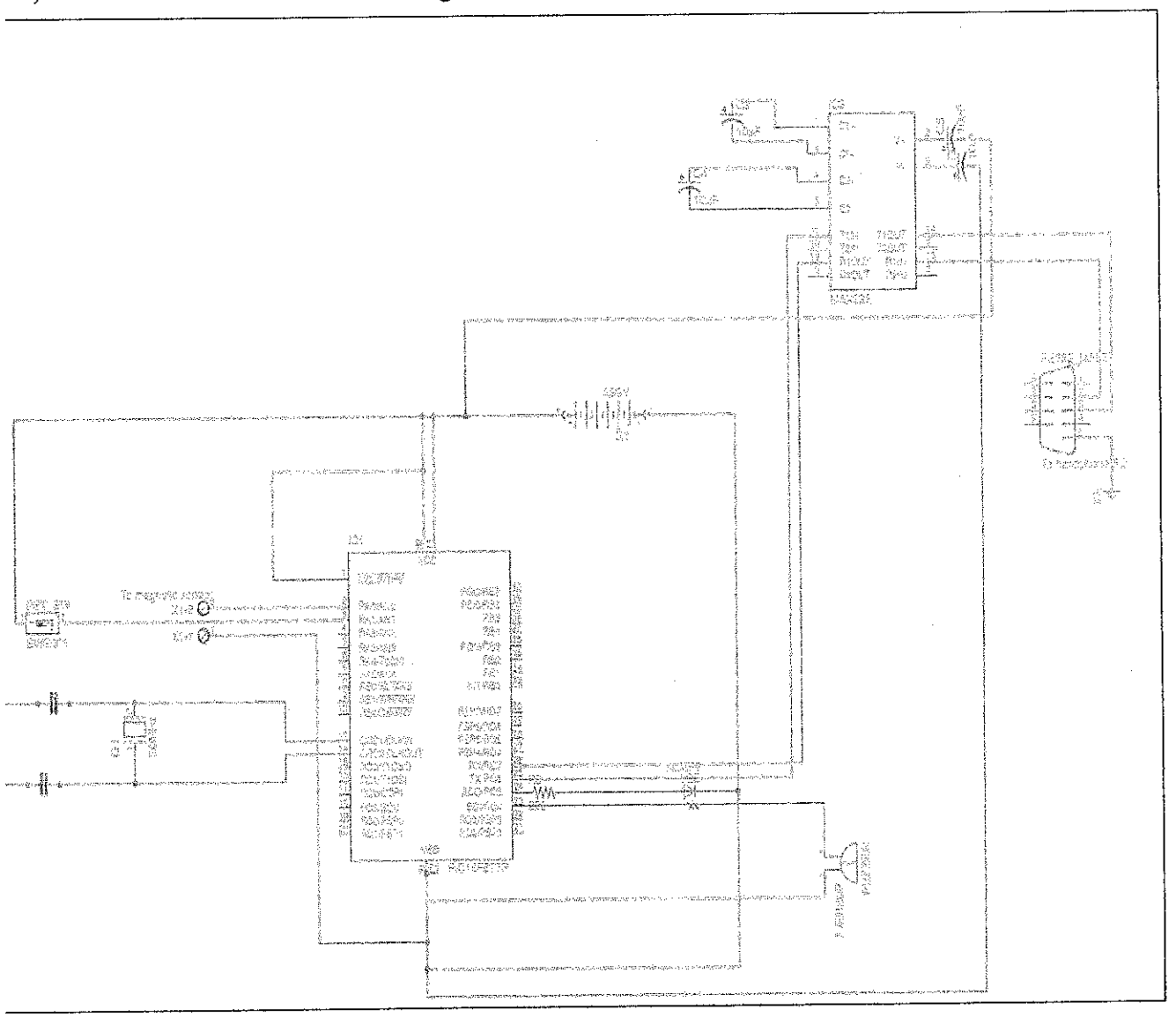


Figure 5.5 Schematic Diagram for the Project

ıe prototype of the circuit is shown below:

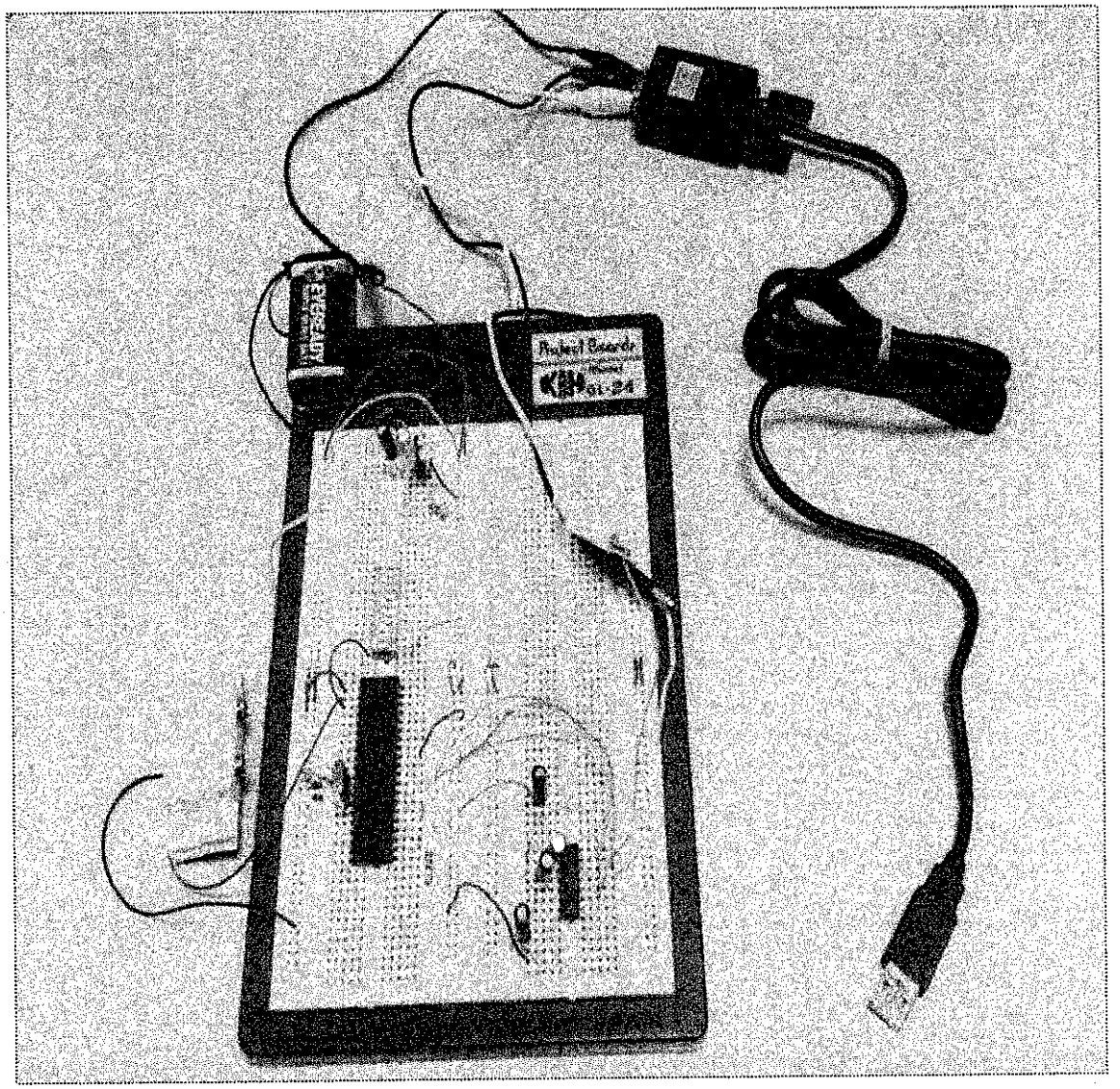


Figure 5.6 The Prototype for the Project

5.1 Setting Up HyperTerminal and using it to communicate with PIC16F877A

After connecting the circuit as in Figure 5.5, we need to set up Hyperterminal.

1) First, go to Start → All Programs → Accessories → Communications → Hyper Terminal

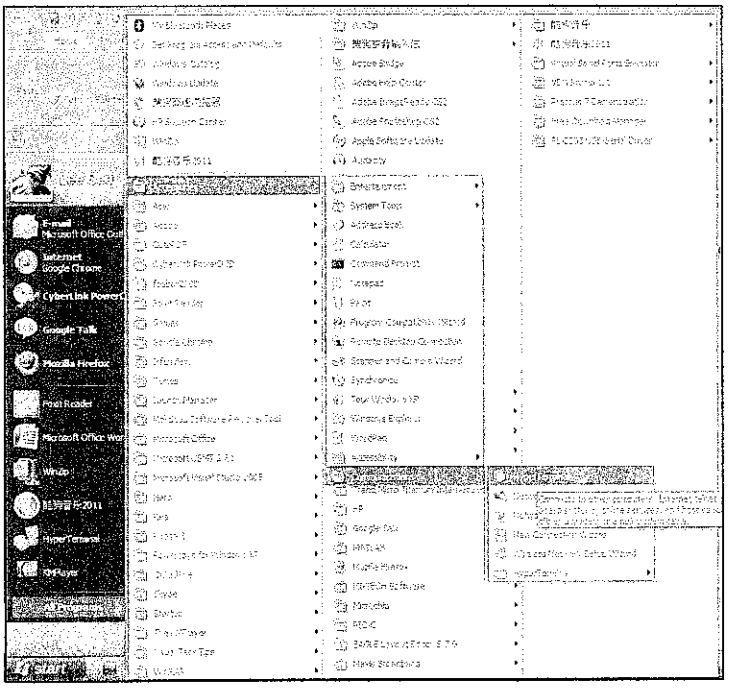


Figure 5.7 Setting Up Hyperterminal

2) On the Connection Description Dialog Box, type a name for the communication, in this case, we typed **usb_usart** and choose the first icon.

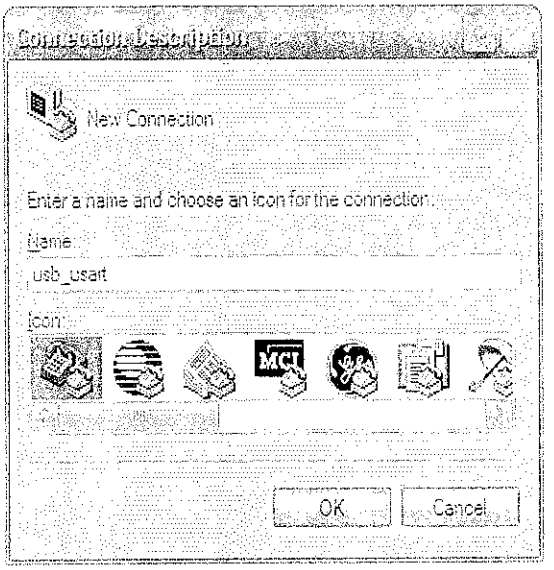


Figure 5.8 Connection Description

3) On the dialog box, Connect using, we choose COM13 (defined earlier for USB to Serial Cable).

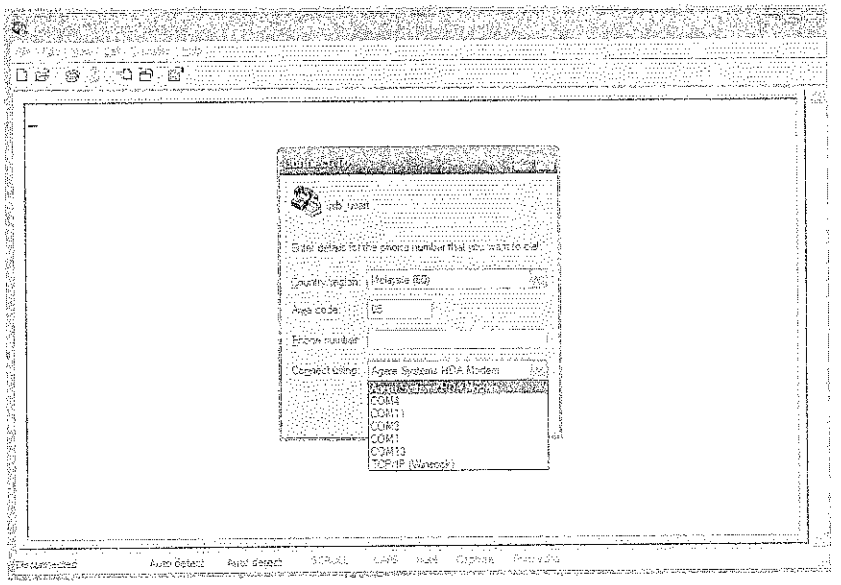


Figure 5.9 Choose the Correct COMPort

4) Define the Port Settings as the following and click OK.

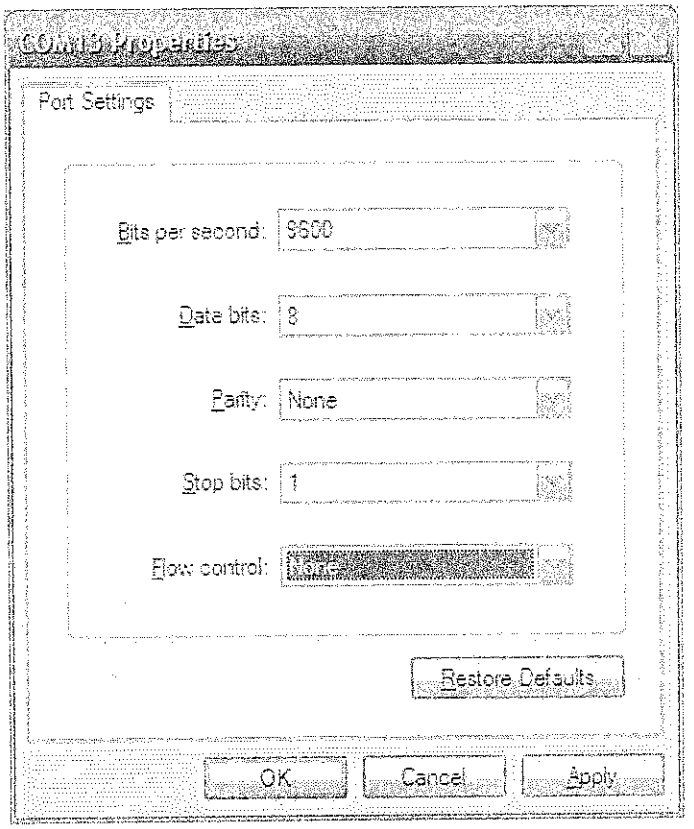


Figure 5.10 Settings for COMPort

5) The hyperterminal is now ready to communicate with the microcontroller.

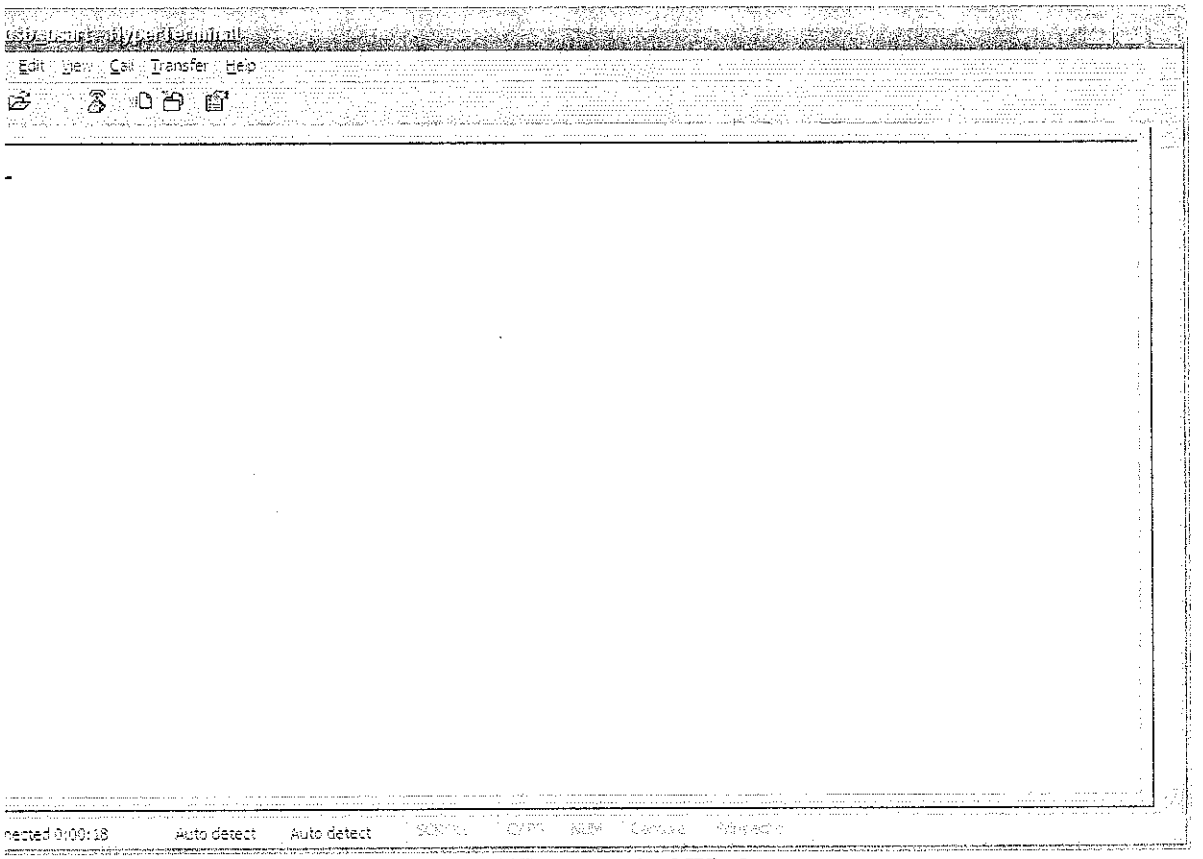


Figure 5.11 Hyperterminal Window

ogram Code:

```
iclude <16f877a.h>
ises HS,NOWDT,NOPROTECT,NOLVP,PUT,NOBROWNOUT
se delay(clock = 20000000)
se RS232(baud = 9600, xmit = PIN_C6, rcv = PIN_C7)

efine sensor PIN_A0
efine sw PIN_A1
efine buzzer PIN_C5
efine led PIN_C4
id main (void)

_TRIS_C(0x00);
_TRIS_A(0xff);
ile(1)

f((input(PIN_A0))&&(!input(PIN_A1)))
 { output_high(led);
 output_high(buzzer);
 delay_ms(10);
 puts("help");

 }
 else if ((input(PIN_A0))&&(input(PIN_A1)))
```

```
{
  output_low(led);
output_low(buzzer); }
else  {
  output_low(led);
output_low(buzzer); }
```

e program code above sets the following situation:

ıen the sensor is triggered off, the LED and Buzzer will be activated and at the same time, a ing of HELP is being displayed in HyperTerminal. If the sensor is being turned off using the et switch on PIN A1, the word "Help" would not be displayed.

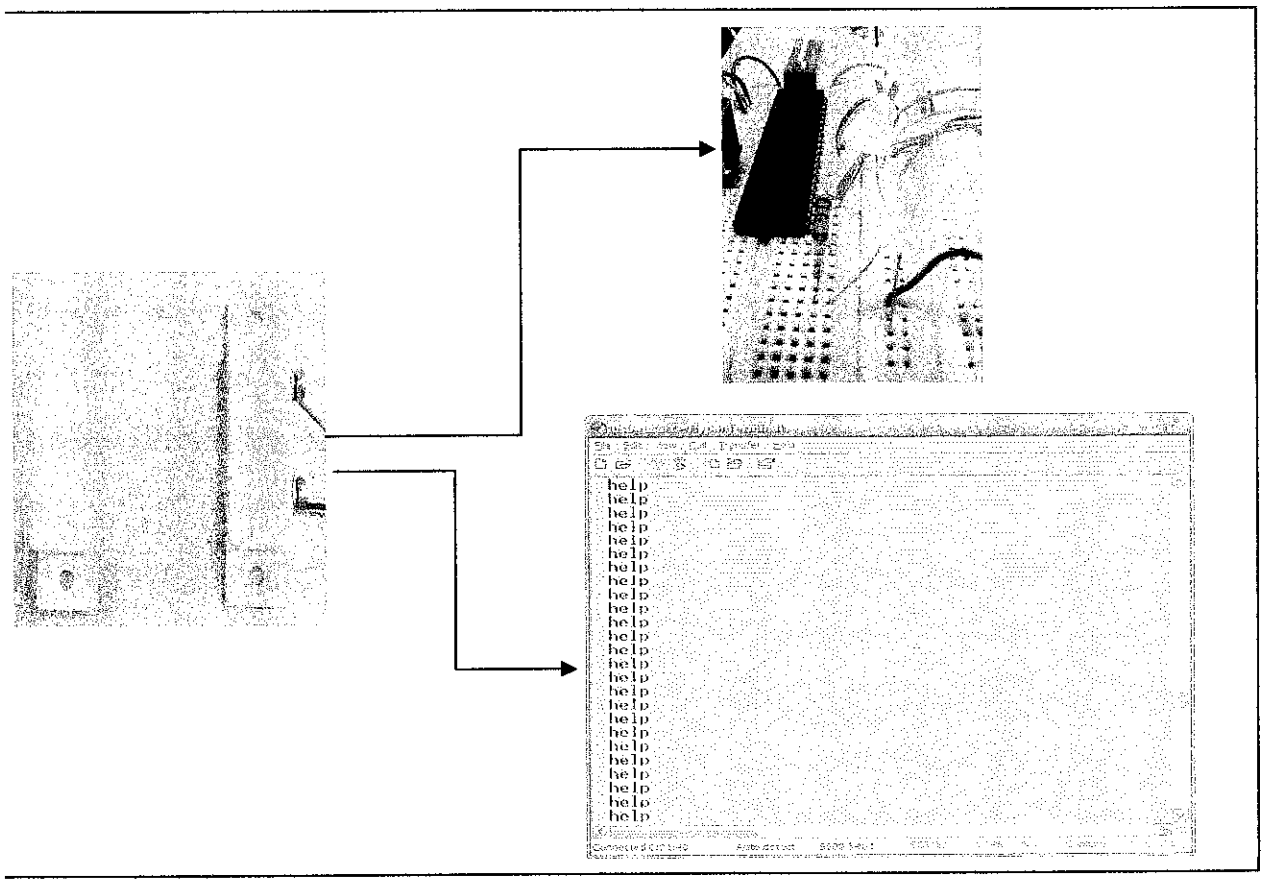


Figure 5.12 Overall Situation

e flowchart below explained the situation:

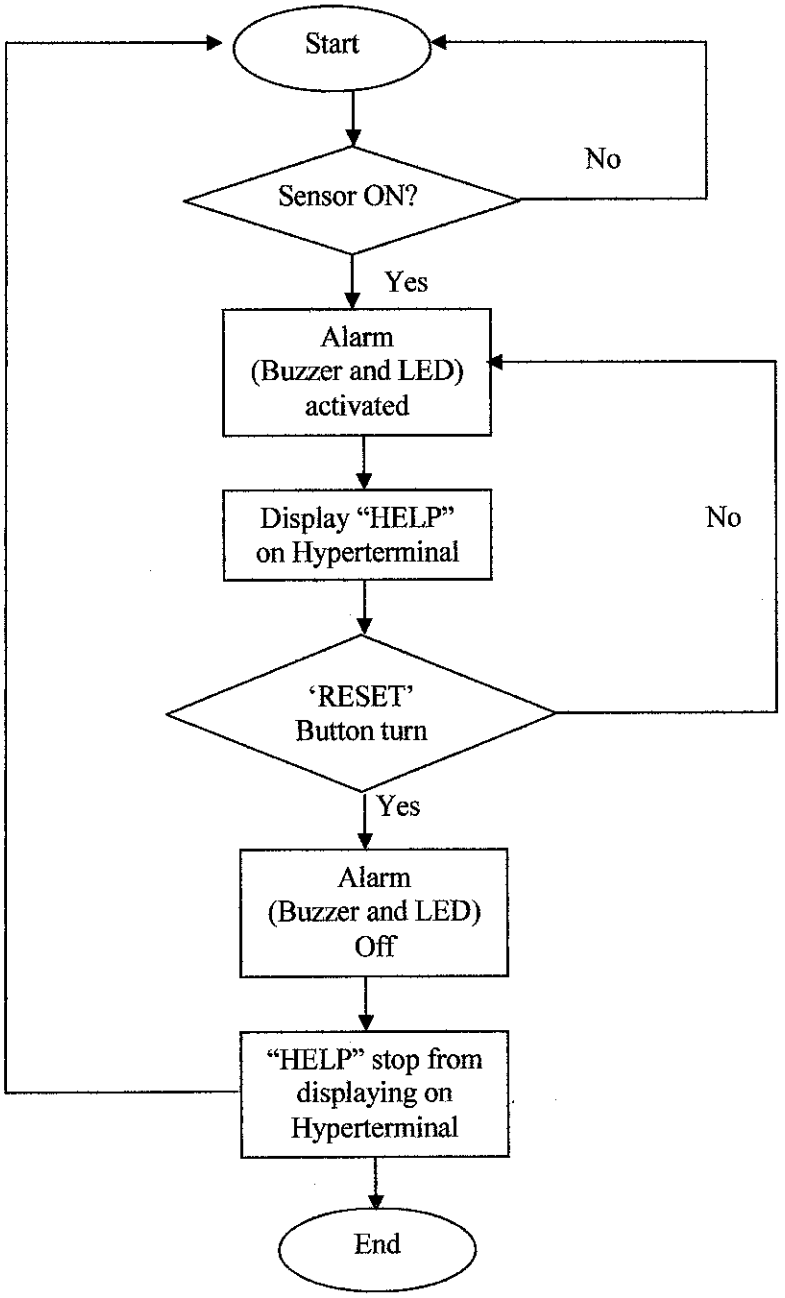
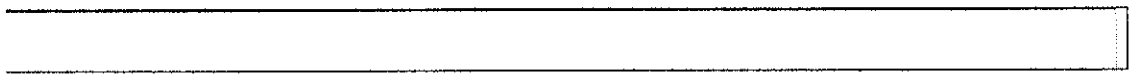


Figure 5.13 Flowchart for Project Part 1

Sending SMS Messages from a PC

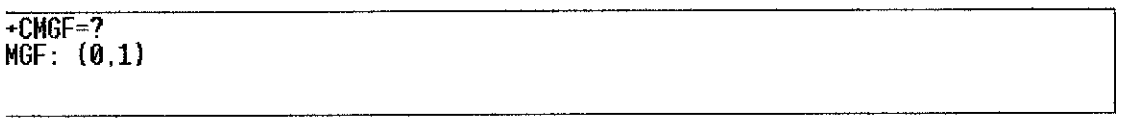
st, we turn on the Bluetooth devices on PC and the mobile phone. Next, we set up perterminal as the step before. But this time, the COM Port being used is COM11.

: start the interfacing by typing AT, and the response is OK which means the PC and Mobile)ne is now connected and we can use AT command to control the mobile phone .



xt, we will check for the SMS text mode that the mobile phone Sony Ericsson T630 is using.

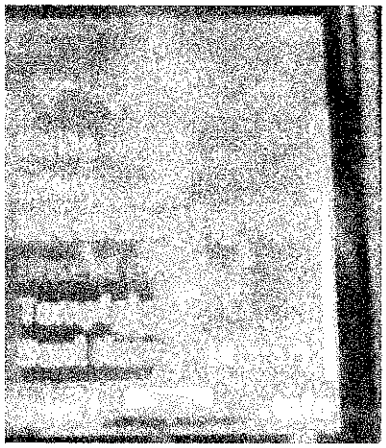
: type AT+ CMGF=? to shows which SMS text mode the phone is using. The response is +CMGF: 1). This means this mobile phone supports both PDU mode and Text mode. Mode 0 refers to PDU de while Mode 1 refers to Text mode.



: are using SMS text mode for this project, hence we need to set the phone to use only Text)de by typing AT + CMGF = 1. The reply of OK means the phone is now set to SMS Text)de.

)w, we start writing our text message to the phone memory by typing the command '+CMGW= "Receiver number" and press enter.

: will see a symbol > being replied, in which we will type our message there and end the :ssage with Ctrl+z. A reply of OK means the message is now stored in SIM card (default).



ure 5.14 Message stored in SIM archive hive

Figure 5.15 Message inside SIM

xt, we will send the message from our storage (SIM) to the receiver by typing AT+CMSS=1. e number 1 represents the message that has just been stored in the SIM card in the position mber 1(default). A reply of OK means the message is being sent.

```
+CMSS=1
MSS: 0
```

e receiver will then receive the text message.

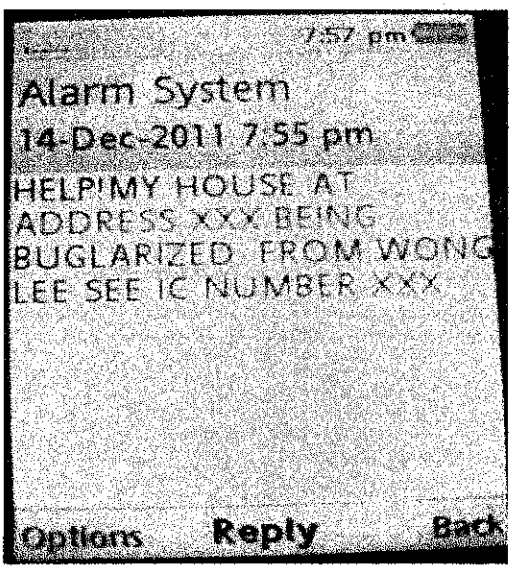


Figure 5.16 Message Received

5 Challenges Faced

here are few challenges faced for the project:

1) The selection of a mobile phone

   Not all mobile phones can be used in this project. Only a mobile phone that can supports SMS text mode is chosen.

2) The unavailability of Serial Cable for Sony Ericsson T630

   Display the output to Hyperterminal, then from Hyperterminal to Handphone.

3) The interfacing from PIC 16F877A to PC (Hyperterminal)

   The setting of baud rate has to be the same for both devices, else the Hyperterminal would not respond.

CHAPTER 6: CONCLUSION AND RECOMMENDATION

Conclusion

e project's main objective is to design and construct a portable security system. The most portant components in the design are the alarm system circuits. Circuit design, circuit testing l C programming testing on the alarm system have been carried out. We have been :cessfully in displaying the output of the alarm system into PC and sending message using PC. e project is still further developed and enhanced for more feasible features.

Recommendation

ıce the project is intended to be portable, some recommendations to improve this project are :eby proposed. The suggestions are listed below:

- More sensors can be included

  Sensors like a smoke detector, a light sensor, amotion detector can be added into the system.

- Remote control circuit

  A remote control circuit can be added to control the alarm system to improve the portability of the system.

REFERENCES

ıin References

Definition of security system, http://www.thefreedictionary.com/security+system

How Fbus protocol work, http://www.embedtronics.com/nokia/fbus.html

Helen Marie Young, Hsiang-Yun Chen. Automatic Phone-Alert Home Security

stem. Columbia University E3390 Final Project, May 7,2007.

Abdul Hassan B Jaafar. To Handphone Burglar Alarm Using PIC 16F877A Microcontroller.
ernational Islamic University Malaysia Engineering Industrial Training, June 2009

] Chun-Pai Jimmy Hsieh, Yang Cao. Home Security System. Cornell University EE476

ıal Project, May 5, 2004.

Dr. Hussein Abdulkadir , Eng. Nada Al-Khatib ,Eng. Osama AlShamma and Eng.

ıhmood Saleh (2006). A Smoke Detection System Using Wireless Network.

ımmunications and Electronics Department /Faculty of Engineering, Philadelphia

ıiversity Amman, Jordan.

ıher References

. Cytron Technologies. PIR Sensor User Manual. Dec 2007.

. Cytron Technologies. UIC00A USB ICSP PIC Programmer User Manual. Oct 2009

. Cytron Technologies. RF Transmitter Module Datasheet. April 2011.

. Cytron Technologies. RF Receiver Module Datasheet. April 2011.

. Holtek Semiconductor Inc. HT12A/HT12E 2^{12} Series of Encoders. February 20, 2009.

. Holtek Semiconductor Inc. HT12D/HT12F 2^{12} Series of Decoders. November 18,2002.

MAXIM. MAX220-249. Revison 16; July 2010.

Lawrence A. Duarte. *The Microcontroller Beginner's Handbook.* 2nd Edition. United ates of America: Prompt Publication. 3-5; 1998.

Steven F. Barrett, Daniel J. Pack .*Microcontrollers Fundamentals for Engineers and ientists*. Morgan & Claypool Publishers, 2006.

). Steven F. Barrett, Daniel J. Pack .*Microcontrollers Fundamentals for Engineers and ientists*. Morgan & Claypool Publishers, 2006.

1. Vivian Capel, *Home Security*. 2nd Edition, First published 1994,1997

2. Zachary Seto, Jackson Yu. Home Security System. Cornell University EE476 Final oject, May 3,1999.

3. Dogan Ibrahim. PIC BASIC Projects: 30 Projects Using PIC BASIC and PIC BASIC RO. Newnes, 2006.

4. "Alarm with pic 16f628, ds275 transceiver, motion sensor and old modem," *tp://www.sistemasorp.es/alarm-with-pic-16f628-ds275-transceiver-motion-sensor-and-old-odem/, Jan 05, 2012*

5. "Wireless Smoke Detector," *tp://www.cytron.com.my/viewProduct.php?pid=LiY4EQULOw4VChYaHicoDReSVd2VAyIdu ctksgAZnU=&store=*, Jan 05, 2012

5. " Planning a Security System/ Burglar System," *tp://www.structuredhomewiring.com/AlarmWiring.aspx, Jan 05, 2012*

7. "Introduction to C and the PIC Microcontroller," *tp://www.swarthmore.edu/NatSci/echeeve1/Ref/C%20for%20PIC/C_Intro.html,* Jan 05,2012

8. "DTMF Tones," *http://lgknowledgebase.com/kb/index.php?View=entry&EntryID=6452,* n 05, 2012

9. "SMS PDU Mode," *http://www.smartposition.nl/resources/sms_pdu.html,* Jan 05, 2012

APPENDICES

ppendix A:PIC16F87XA Pinout Description

PIC16F87XA

TABLE 1-2: PIC16F873A/876A PINOUT DESCRIPTION

Pin Name	PDIP, SOIC, SSOP Pin#	QFN Pin#	I/O/P Type	Buffer Type	Description
OSC1/CLKI	9	6		ST/CMOS[3]	Oscillator crystal or external clock input.
OSC1			I		Oscillator crystal input or external clock source input. ST buffer when configured in RC mode; otherwise CMOS.
CLKI			I		External clock source input. Always associated with pin function OSC1 (see OSC1/CLKI, OSC2/CLKO pins).
OSC2/CLKO	10	7		—	Oscillator crystal or clock output.
OSC2			O		Oscillator crystal output. Connects to crystal or resonator in Crystal Oscillator mode.
CLKO			O		In RC mode, OSC2 pin outputs CLKO, which has 1/4 the frequency of OSC1 and denotes the instruction cycle rate.
MCLR/VPP	1	26		ST	Master Clear (input) or programming voltage (output).
MCLR			I		Master Clear (Reset) input. This pin is an active low Reset to the device.
VPP			P		Programming voltage input.
					PORTA is a bidirectional I/O port.
RA0/AN0	2	27		TTL	
RA0			I/O		Digital I/O.
AN0			I		Analog input 0.
RA1/AN1	3	28		TTL	
RA1			I/O		Digital I/O.
AN1			I		Analog input 1.
RA2/AN2/VREF-/CVREF	4	1		TTL	
RA2			I/O		Digital I/O.
AN2			I		Analog input 2.
VREF-			I		A/D reference voltage (Low) input.
CVREF			O		Comparator VREF output.
RA3/AN3/VREF+	5	2		TTL	
RA3			I/O		Digital I/O.
AN3			I		Analog input 3.
VREF+			I		A/D reference voltage (High) input.
RA4/T0CKI/C1OUT	6	3		ST	
RA4			I/O		Digital I/O – Open-drain when configured as output.
T0CKI			I		Timer0 external clock input.
C1OUT			O		Comparator 1 output.
RA5/AN4/SS/C2OUT	7	4		TTL	
RA5			I/O		Digital I/O.
AN4			I		Analog input 4.
SS			I		SPI slave select input.
C2OUT			O		Comparator 2 output.

Legend: I = input O = output I/O = input/output P = power
— = Not used TTL = TTL input ST = Schmitt Trigger input

Note 1: This buffer is a Schmitt Trigger input when configured as the external interrupt.
2: This buffer is a Schmitt Trigger input when used in Serial Programming mode.
3: This buffer is a Schmitt Trigger input when configured in RC Oscillator mode and a CMOS input otherwise.

Commands

AT+CPMS **Preferred Message Storage (ver.4)**

Description: Set command selects memory storage <mem1>, <mem2> and <mem3> to be used for reading, writing, etc. If chosen storage is not appropriate for the phone (but is supported by the phone), final result code +CMS ERROR: <err> shall be returned.

Test command returns lists of memory storage supported by the phone.

Execution command: AT+CPMS=<mem1>
[,<mem2>
[,<mem3>]]

Execution command response: +CPMS: <used1>,<total1>,<used2>,<total2>,<used3>,<total3>

Read command: AT+CPMS? Displays the current <mem1>,<used1>,<total1>, <mem2>,<used2>,<total2>, <mem3>,<used3>,<total3> settings.

Test command: AT+CPMS=? Shows if the command is supported.

Test command response: +CPMS: (list of supported <mem1>s),(list of supported <mem2>s),(list of supported <mem3>s)

Termination: Phone

Parameters:

<mem1>:

<mem1>	Description
string type	Memory from which messages are read and deleted (commands List Messages +CMGL, Read Message +CMGR and Delete Message +CMGD).
"ME"	phone message storage
"SM"	SIM message storage

<mem2>:

<mem2>	Description
string type	Memory to which writing and sending operations are made (commands Send Message from Storage +CMSS and Write Message to Memory +CMGW).
"ME"	phone message storage
"SM"	SIM message storage

<mem3>:

<mem3>	Description
string type	Memory to which received SMs are preferred to be stored (unless forwarded directly to terminal equipment). Received CBMs are always stored in "BM" (or some manufacturer specific storage) unless directly forwarded to terminal equipment.
"ME"	phone message storage
"SM"	SIM message storage

<used1>,<used2>,<used3>

<used1>,<used2>, <used3>	Description
Integer type	Total number of messages currently in <mem1>, <mem2> and <mem3> respectively.

<total1>,<total2>,<total3>

<total1>,<total2>, <total3>	Description
Integer type	Total number of messages currently in <mem1>, <mem2> and <mem3> respectively.

AT+CMGF Message Format (ver.2)

Description: Set command tells the phone, which input and output format of messages to use. <mode> indicates the format of messages used with send, list, read and write commands and unsolicited result codes resulting from received messages. Mode can be either PDU mode (entire TP data units used) or text mode (headers and body of the messages given as separate parameters).

Execution command: AT+CMGF=<mode>

Read command: AT+CMGF? Displays the current <mode> setting.

Test command: AT+CMGF=? Shows if the command is supported.

Test command response: +CMGF: (list of supported <mode>s)

Termination: Phone

Parameter:

<mode>

<mode>	Description
0	PDU mode
1	Text mode

AT+CMGW	Write Message To Memory (ver.4)
Description:	Execution command stores a message to memory storage <mem2>. Memory location <index> of the stored message is returned. By default message status will be set to 'stored unsent', but parameter <stat> allows also other status values to be given. (phone manufacturer may choose to use different default <stat> values for different message types.) The entering of PDU is done similarly as specified in command Send Message +CMGS. If writing fails, final result code +CMS ERROR: <err> is returned. Note: in text mode character set UCS2 must be used.

Execution command:	if text mode (+CMGF=1):
	+CMGW[=<oa/da>[,<toda/toda>[,<stat>]]]<CR>text is entered<ctrl-Z/ESC>
	if PDU mode(CMGF=0):
	AT+CMGW=<length>[,<stat>]<CR>
	<pdu><ctrl-Z/ESC>

Execution command response:	+CMGW: <index>
Test command:	AT+CMGW=? Shows if the command is supported.
Termination:	Phone
Parameters:	
<oa>:	

<oa>	Description
String type	TP-Originating-Address Address-Value field in string format; BCD number (or GSM 7 bit default alphabet characters) are converted to characters of currently selected terminal equipment character set. Refer to +CSCS.

<da>:

<da>	Description
String type	TP-Originating-Destination Address-Value field in string format; BCD number (or GSM 7 bit default alphabet characters) are converted to characters of currently selected terminal equipment character set. Refer to +CSCS.

<toda>:

<tooa>	Description
Integer type	TP-Originating-Address Type-of-Address octet in integer format. When first character of <oa> is +(IRA 43) default is 145, otherwise default is 129.

<toda>:

<toda>	Description
Integer type	TP-Destination-Address Type-of-Address octet in integer format. When first character of <da> is +(IRA 43) default is 145, otehrwise default is 129.

<stat>:

<stat>	Description
0	Received unread message (i.e. new message)
1	Received read message
2	Stored unsent message (only applicable to SMs)
3	Stored sent message (only applicable to SMs)
16	Template message

<index>:

<index>	Description
Integer type	Value in the range of location numbers supported by the associated memory.

<length>:

<length>	Description
Integer type	Value indicating in PDU mode (+CMGF=0), the length of the actual TP data unit in octets (i.e. the RP layer SMSC address octets are not counted in the length).

<pdu>:

<pdu>	Description
...	In the case of SMS: GSM 04.11 SC address followed by GSM 03.40 TPDU in hexadecimal format: phone converts each octet of TP data unit into two IRA character long hexadecimal number (e.g. octet with integer value 42 is presented to terminal equipment as two characters 2A (IRA 50 and 65)) In the case of CBS: GSM 03.41 TPDU in hexadecimal format

AT+CMGC Send command (ver. 2)

Description:

Text mode:

Execution command sends a command message from a terminal equipment to the network (SMS-COMMAND). The entering of text is done similarly as specified in command Send Message +CMGS, but the format is fixed to be a sequence of two IRA character long hexadecimal numbers which phone converts into 8-bit octets (refer +CMGS). Message reference value <mr> is returned to the terminal equipment on successful message delivery. Optionally (when +CSMS <service> value is 1 and network supports) <sctsa> is returned. Values can be used to identify message upon unsolicited delivery status report result code.

PDU mode:

Execution command sends a command message from a terminal equipment to the network (SMS-COMMAND). The entering of PDU is done similarly as specified in command Send Message +CMGS. Message reference value <mr> is returned to the terminal equipment on successful message delivery. Optionally (when +CSMS <service> value is 1 and network supports) <ackpdu> is returned. Values can be used to identify message upon unsolicited delivery status report result code. If sending fails in a network or an phone error, final result code +CMS ERROR: <err> is returned.

Execution command:

if text mode (+CMGF=1):
AT+CMGC=<fo>,<ct>[,<da>[,<toda>]]
<CR>**text is entered**<ctrl-Z/ESC>
if PDU mode (+CMGF=0):
AT+CMGC=<length><CR>
<pdu><ctrl-Z/ESC

Execution command response:

- If text mode (+CMGF=1) and sending successful:
  +CMGC=<mr>[,<scts>]
- If PDU mode (+CMGF=0) and sending successful:
  +CMGC: <mr>[,<ackpdu>]
- If sending fails:
  +CMS ERROR: <err>
  OK
  ERROR

Test command: AT+CMGC=? Shows if the command is supported.

Termination: Phone

Parameters:

<fo>:

<fo>	Description
Integer type	Depending on the command or result code: First octet of *3rd Generation Partnership Project; Technical Specification Group Terminals; Technical realization of the Short Message Service (SMS)* SMS-DELIVER, SMS-SUBMIT, or SMS-COMMAND.

<ct>:

<ct>	Description
Integer type	TP-Command-Type

<da>:

<da>	Description
String type	TP-Originating-Destination Address-Value field in string format; BCD number (or GSM 7 bit default alphabet characters) are converted to characters of currently selected terminal equipment character set. Refer to +CSCS.

<toda>:

<toda>	Description
Integer type	TP-Destination-Address Type-of-Address octet in integer format. When first character of <da> is +(IRA 43) default is 145, otherwise default is 129.

<mr>:

<mr>	Description
Integer type	GSM 03.40 TP-Message-Reference in integer format.

<scts>:

<scts>	Description
String type	TP-Service-Center-Time-Stamp in time-string format.

<length>:

<length>	Description
Integer type	Value indicating in PDU mode (+CMGF=0), the length of the actual TP data unit in octets (i.e. the RP layer SMSC address octets are not counted in the length)

<pdu>:

<pdu>	Description
...	In the case of SMS: GSM 04.11 SC address followed by GSM 03.40 TPDU in hexadecimal format: phone converts each octet of TP data unit into two IRA character long hexadecimal number (e.g. octet with integer value 42 is presented to terminal equipment as two characters 2A (IRA 50 and 65)); In the case of CBS: GSM 03.41 TPDU in hexadecimal format

<mr>:

<mr>	Description
Integer type	GSM 03.40 TP-Message-Reference in integer format.

<ackpdu>:

<ackpdu>	Description
...	GSM 03.40 RP-User-Data element of RP-ACK PDU; format is same as for <pdu> in case of SMS, but without GSM 04.11 SC address field and parameter shall be bounded by double quote characters like a normal string type parameter

AT — Attention Command

Description: Checks the communication between the phone and any accessory. Determines the presence of a phone.

Execution command: AT

AT+CLAC — List All Available AT Commands

Description: The command causes the ME to return one or more lines of AT Commands.

Note: This command only returns the AT commands available to the user.

Execution command: AT+CLAC

Possible response(s):

<AT Command1><CR><LF>

[<AT Command2><CR><LF>

[...]]

+CME Error: <err>

<AT Command>	Description
AT...	Defines the AT command, including the prefix AT

Test command: AT+CLAC=? Shows if the command is supported.

Example

```
AT+CBAC
AT+CBAC
AT+CBAC
AT+CBAC
...
OK

+AT+CBAC=?
OK
```

ATA Answer Incoming Call Command

Description: Answers and initiates a connection to an incoming call.

Execution command: ATA

Possible responses:

CONNECT

CONNECT <text>

<text>	Description
28800	Connected with data bit rate of 28800 bits/s (HSCSD)
19200	Connected with data bit rate of 19200 bits/s (HSCSD)
14400	Connected with data bit rate of 14400 bits/s (HSCSD)
9600	Connected with data bit rate of 9600 bits/s
4800	Connected with data bit rate of 4800 bits/s
2400	Connected with data bit rate of 2400 bits/s

NO CARRIER The mobile phone is not registered.

ERROR If ATA is unsuccessfully executed by the phone.

ATH Hook Control

Description: Terminates a connection.

Execution command: ATH[<n>]

Parameter:

<n>:

<n>	Description
0	Disconnect data connection

ATD Dial Command

Description: Initiates a phone connection, which may be data, facsimile (+FCLASS> 0), or voice (phone number terminated by semicolon). The phone number used to establish the connection will consist of digits and modifiers, or a stored number specification.

Execution command:

ATD[<dial_string>][;] Dial the phone number specified in the <dial_string> parameter.

ATD>ME<i>[;] Dial the phone number stored in the mobile phone located by the index<i>.

ATD>SIM<i>[;] Dial the phone number stored in the SIM card located by the index <i>.

ATD>LD<i>[;] Dial the number stored in position <i> in the Last Dialled Number list on the SIM card. The most recently dialled number is assumed to have <i>="1".

ATDL Redial the last dialled phone number.

Possible responses:

CONNECT

CONNECT <text>

<text>	Description
28800	Connected with data bit rate of 28800 bits/s (HSCSD)
19200	Connected with data bit rate of 19200 bits/s (HSCSD)
14400	Connected with data bit rate of 14400 bits/s (HSCSD)
9600	Connected with data bit rate of 9600 bits/s
4800	Connected with data bit rate of 4800 bits/s
2400	Connected with data bit rate of 2400 bits/s

NO CARRIER The mobile phone is not registered.

ERROR If ATD is unsuccessfully executed by the phone.

NO DIALTONE The mobile phone is being used for a voice call or is not within coverage of the network.

BUSY The phone number called is engaged; only valid for data and fax connections.

OK Only valid for voice calls.

Parameter:

<dial_string>:

<dial_string>	Description
"0 1 2 3 4 5 6 7 8 9 +"	Valid characters for origination
W	The W modifier is ignored but is included for compatibility reasons only
,	The comma modifier is ignored but is included for compatibility reasons only
:	Informs the Infrared Modem that the number is a voice number rather than a fax or data number
T	The T modifier is ignored but is included only for compatibility purposes
P	The P modifier is ignored but is included only for compatibility purposes

AT+GMI Request Manufacturer Information

Description: Returns the manufacturer information for the infrared modem.

Execution command: AT+GMI

Execution command response: <manufacturer>

Test command: AT+GMI=? Shows if the command is supported.

Parameter:

<manufacturer>: String of characters.

Example:

```
AT+GMI
[illegible]
OK

AT+GMI=?
OK
```

AT+GMM Request Model Identification

Description: Returns the model identification for the infrared modem.

Execution command: AT+GMM

Execution command response: <model>

Test command: AT+GMM=? Shows if the command is supported.

Parameter:

<model>: String of characters.

Example:

```
AT+GMM
[illegible]
OK

AT+GMM
OK
```

AT+GMR Request Revision Identification

Description: Returns the revision identification of the infrared modem.

Execution command: AT+GMR

Execution command response: <revision>

Test command: AT+GMR=? Shows if the command is supported.

Parameter:

<revision>: String of characters.

Example:

```
AT+GMR
[illegible]
OK

AT+GMR
OK
```