

CERTIFICATION OF APPROVAL

Wireless Sensor Device For Health Monitoring,
Using ZigBee Technology

by

Mohamad Zulhilmi Bin Zaini

A project submitted to the
Electrical and Electronics Programme Universiti Teknologi PETRONAS
in partial fulfillment of the requirements for the
BACHELOR OF ENGINEERING (Hons)
(ELECTRICAL AND ELECTRONICS ENGINEERING)

Approved by,

(Dr. Noohul Basheer Zain Ali)

UNIVERSITI TEKNOLOGY PETRONAS
TRONOH, PERAK
May 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 that the original work contained herein have not been undertaken or done by unspecified sources or persons.

MOHAMAD ZULHILMI BIN ZAINI

ACKNOWLEDGEMENT

First of all, I would like to thank Allah for HIS firm hands in guiding me in the course of completing this thesis writing. It is by HIS grace and mercy that I am able to embark on the project within such a limited time. Alhamdulillah.

Second, I would like to express my gratitude and thanks to my supervisor, Dr Noohul Basheer Zain Ali, for his professional guidance, wisdom, endurance, advices, motivation and encouragement throughout the project.

I also would like to thank all my fellow friends for their contribution in giving me a moral support throughout the project development period. Last but not least, to all my beloved family members who were always by my side to encourage, advice, comfort, cherish and support me during this entire project.

Finally, I really appreciate to have this responsibility to finish this project. This task has taught a lot of lesson and knowledge which would be valuable to me in the future.

ABSTRACT

In this project, we develop a portable real-time wireless health monitoring system. Goal of the project is to monitor patient's heart rate. The system was designed and implemented using ZigBee wireless technologies. All data will be transferred within a group of wireless network to database computer system. The sensor modules were designed for low power operation so that it can last long. This project is divided into two phase which is sensor design phase and transmission phase. In the first phase, sensor unit consists of two types of LEDs and photodiode that is facing to a patient's fingertip and microcontroller unit for interacting with ZigBee module. Second phase involve ZigBee module to communicate with heart rate sensor. ZigBee module gets all commands from microcontroller unit and it works on both transmitting and receiving process. Throughput from ZigBee receiver will be displayed at computer for monitoring purpose. This method can be developed into a noninvasive technique to measure body temperature and blood pressure.

TABLE OF CONTENTS

ACKNOWLEDGEMENT.	iii
ABSTRACT	iv
LIST OF FIGURES	viii-ix
LIST OF TABLES	x

CHAPTER 1: INTRODUCTION

1.1 Background of Study	1-2
1.2 Problem Statement	2
1.3 Objective	3
1.4 Scope of study	3

CHAPTER 2: LITERATURE REVIEW

2.1 Heart Rate Sensor		
2.1.1 Introduction of Plethysmography	4-8
2.1.2 Electronic circuit	9-10
2.1.3 Theoretical Results	11
2.1.3.1 Pulse Time Delay Method	12-13
2.1.3.2 Pulse Amplitude Difference Method	13-14

2.2 ZigBee

2.2.1 Network Topology	16
2.2.2 ZigBee Stack Architecture	17
2.2.3 Comparison between ZigBee and Bluetooth	18
2.2.4 Advantages of ZigBee	19
2.2.5 Tools used	20-22
2.2.6 Assumptions	22

CHAPTER 3: METHODOLOGY

3.1 Procedure Identification	24
3.2 Methodology	25-26
3.3 Tools / Software / Equipment Required	26
3.4 Suggested System	26
3.5 Setting Up Xbee Network	27
3.6 Project Milestone	28

CHAPTER 4 : RESULTS AND DISCUSSION

4.1 Sensor module	29-31
4.2 Controller System	31-32
4.3 Transmission system	32-35
4.4 Discussion	36

CHAPTER 5: CONCLUSION

5.1 Conclusion 37

5.2 Recommendations. 38

REFERENCES 39-41

APPENDICES 42-54

LIST OF FIGURES

Figure 1 : Penaz's method of photoplethysmography.....	5
Figure 2 : Light absorption by hemoglobin at different wavelengths.....	6
Figure 3 : Schematic representation of the pulse wave transmit time (PWTT)...	7
Figure 4 : Probe-Transmissive.....	7
Figure 5 : Probe-Reflective.....	8
Figure 6 : Measures of the received signals are processed to yield SpO ₂ values..	8
Figure 7 : Sensing system.....	9
Figure 8 : Schematic diagram for the analog front end plethysmograph.....	10
Figure 9: Basic circuit topology.....	10
Figure 10 : Five seconds recording by plethysmograph.....	11
Figure 11 : Pulse pressure signal in the proximal cuff 1 and photoplethysmograph sensor	12
Figure 12: Pulse time delay as a function of pressure.....	13
Figure 13 : Pulse pressure signal in the proximal cuff 1 and photoplethysmograph sensor	14
Figure 14 : Pulse time delay as a function of pressure.....	14
Figure 15 : ZigBee network topologies.....	16
Figure 16 : ZigBee Stack Architecture.....	17
Figure 17 : System Data Flow in a UART interfaced environment.....	19
Figure 18 : Transmitter and Receiver LED.....	20
Figure 19 : SKXbee Module.....	20
Figure 20 : MikroC Pro software.....	21
Figure 21 : XCTU software.....	22
Figure 22: Flow Chart of The Project.....	24

Figure 23 : Xbee transmitter module.....	26
Figure 24 : Suggested system.....	26
Figure 25 : Sensor module.....	29
Figure 26 : Sensor circuit.....	30
Figure 27 : Result before putting a finger.....	30
Figure 28 : Result after putting a finger.....	31
Figure 29: Key Features for PIC16F877.....	32
Figure 30 : Xbee module attached to socket.....	34
Figure 31 : Overview of SKXbee system.....	35
Figure 32: Transmitter circuit.....	35
Figure 33: SKXbee at PC system.....	35

LIST OF TABLES

Table 1 : ZigBee versus Bluetooth.....18

CHAPTER 1

INTRODUCTION

This chapter will briefly explain on importance of ambulatory monitoring system in daily life. Besides that, this chapter also will give some overview about ZigBee technology as a wireless medium.

1.1 Background Study

Health monitoring systems become an important research field today. Research on health monitoring were applied in many areas such as sports training, homecare unit, military, and emergency monitoring system. Daily health monitoring outside clinical environments or medical offices is instrumental to detect possible health disorders before they become important and for life-style improvement, which can help in preventing illnesses [1]. In this thesis, the project is wearable and real time monitoring system. This system will help doctor to monitor emergency alarm from his patient. There are many parameters covered in health monitoring system such as heart rate, body temperature, blood pressure, and respiratory rate. In this work, we have considered heart rate.

Data from heart rate is important for doctor to monitor patient's health condition. Besides that, data from heart rate also can help doctor to monitor patient with heart rate abnormalities like Tachycardia, Bradycardia, and Arrhythmia. An Australian-led international study of patients with cardiovascular disease has shown that heart beat rate is a key indicator for the risk of heart attack. The study, published in The Lancet (September 2008) studied 11,000 people, across 33 countries, who were being treated for heart problems. Those patients whose heart rate was above 70 beats per minute had significantly higher incidence of heart attacks, hospital admissions and the need for surgery. For the best advantage, health monitoring during daily life should be performed with a minimal intervention of the subject and without disturbing his or her daily activities [2].

University of Sydney professor of cardiology Ben Freedman from Sydney's Concord hospital, said "If you have a high heart rate there was an increase in heart attack, there was about a 46 percent increase in hospitalizations for non-fatal or fatal heart attack"[3].

There are many applications of wireless sensor networks such as Infrared, Bluetooth and ZigBee. Problem with Infrared transmission is the limitation angle. Although Bluetooth is better than ZigBee for transmission rate, but ZigBee has low power consumption[4]. Hence, ZigBee is the best choice for 24 hours monitor of communication system. The first step is we use sensor to measure heart rate from human body, then the measured signal sends to the end device. End device will connect to PC through USB dongle and PC will analyze and store the signal.

1.2 Problem Statement

Emergency and sudden changes in human body lead to a critical condition especially for cardiac patients. Ambulatory monitoring is needed because cardiac problems often occur during normal daily activities. It is important to have a wearable monitoring device which is compact, lightweight and comfortable to wear all the times. Besides that, it must be designed for low power consumption for long term use.

1.3 Objective

The goal of this project is to develop a device to obtain consistent and reliable measurement vital signs in for long term use. A small photoplethysmograph (PPG) device will be designed and tested, which will be used to measure series of data such as heart rate.

1.4 Scope of Study

This project involves study in Computer system and Communication field. Overall project is divided into two parts. The first part have focused on sensor circuit and the second part have focused on data transmission. The first part have covered literature review on the basic concept of such as Photoplethysmograph (PPG) and ZigBee Technology, project designation and construct sensor circuit. Second part have covered on transmission data using ZigBee module.

CHAPTER 2

LITERATURE REVIEW AND THEORY

This chapter will give overview on the previous project that was been made and main components that involve in this project. Furthermore, it will also explain on theory of heart rate sensor and ZigBee technologies in detecting heart rate.

2.1 Heart Rate Sensor

Heart rate is the number of [heartbeats](#) per unit of [time](#) - typically expressed as beats per minute ([bpm](#)) - which can vary as the body's need to absorb oxygen and excrete carbon dioxide changes, such as during [exercise](#) or sleep. The measurement of heart rate is used by [medical professionals](#) to assist in the [diagnosis](#) and tracking of medical conditions[5].

2.1.1 Introduction of Plethysmograph

The photoplethysmograph (PPG) is a non-invasive electro-optical signal to measure pulsations associated in changes of blood volume. The method was first introduced by Penaz. The principle of indirect continuous monitoring of blood pressure waveform is unloading technique proposed first by Marey (1876), later specified by Shirer (1962).

This method is based on idea that if an externally applied pressure in the cuff is equal to the arterial pressure instantaneously, the arterial walls will be unloaded (zero transmural pressure) and the arteries will not change in size. In this condition, the blood volume will not change. This method was attempt to realise for the first time by Penaz (1973) using photoelectric technique of detecting blood flow, equipped with a transparent inflatable cuff controlled by a servocontrol system in the human finger [6].

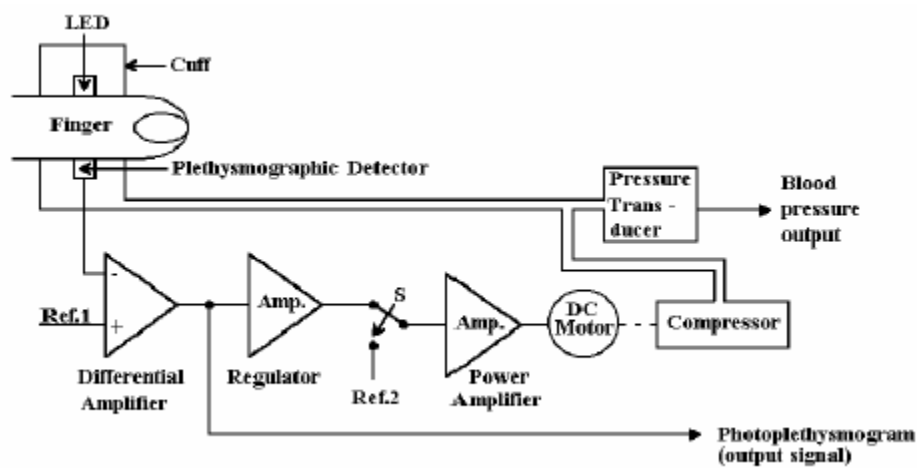


Figure 1 : Penaz's method of photoplethysmography

Contraction of the heart causes a pressure wave which moves along the arteries producing, as a consequence, their expansion during the positive peak. The wave is faster than the blood flow and its speed reaches a few meters per second. The pulse wave can be sensed at a limb as well as the wrist or a finger.

Two possibilities for sensing the pulse wave are via a pressure sensor or through an optoelectronic plethysmograph which uses the physical mechanism of light absorption. Hemoglobin present in the blood absorbs the light emitted in a particular wavelength range (see Figure 2 below). In this system infrared light can be used with no distinction between oxyhemoglobin and deoxyhemoglobin.

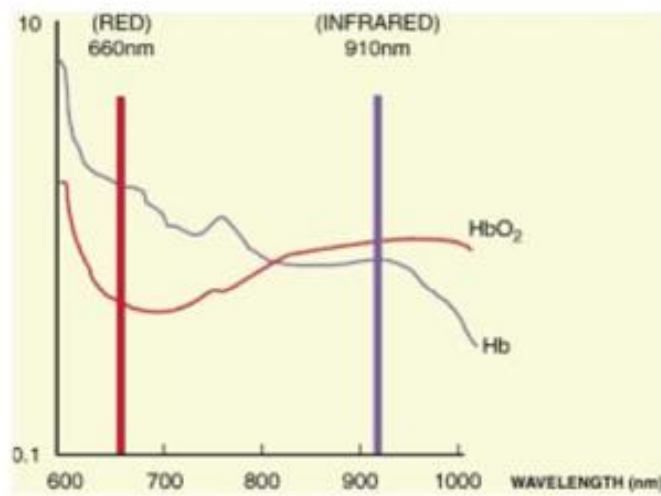


Figure 2 : Light absorption by hemoglobin at different wavelengths

For this reason, the light which is able to pass through the body at a wavelength of 600-900 nm depends on the quantity of hemoglobin flowing in the blood vessels. Therefore, since the quantity of hemoglobin at a given time is proportional to the pulse wave at that time, it is possible to calculate the pulse wave from the transmitted light[7].

Information from the pulse wave is often used in conjunction with a three-lead electrocardiogram (ECG or EKG) with the differential electrodes placed on the thorax. In fact, from the two measurements, it is possible to calculate the PWTT (pulse wave transit time) which is the time interval between the R wave peak of the ECG and the positive peak of the plethysmograph (see Figure 3) and can be correlated with blood pressure.

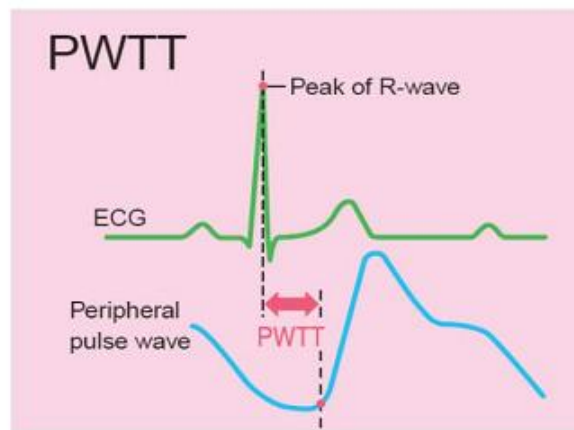


Figure 3 : Schematic representation of the pulse wave transmit time (PWTT)

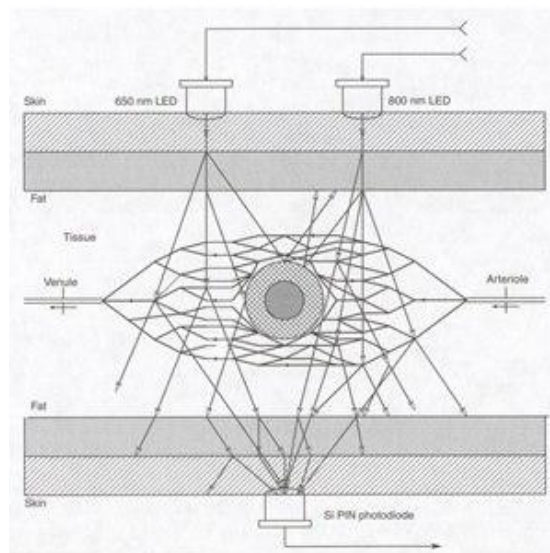


Figure 4 : Probe-Transmissive

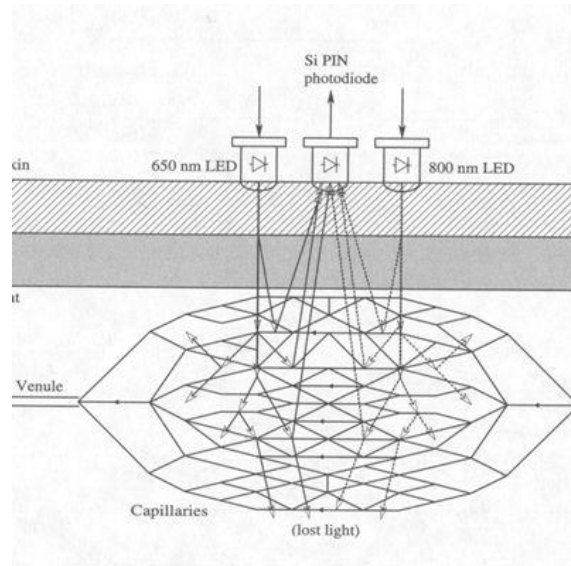


Figure 5 : Probe-Reflective

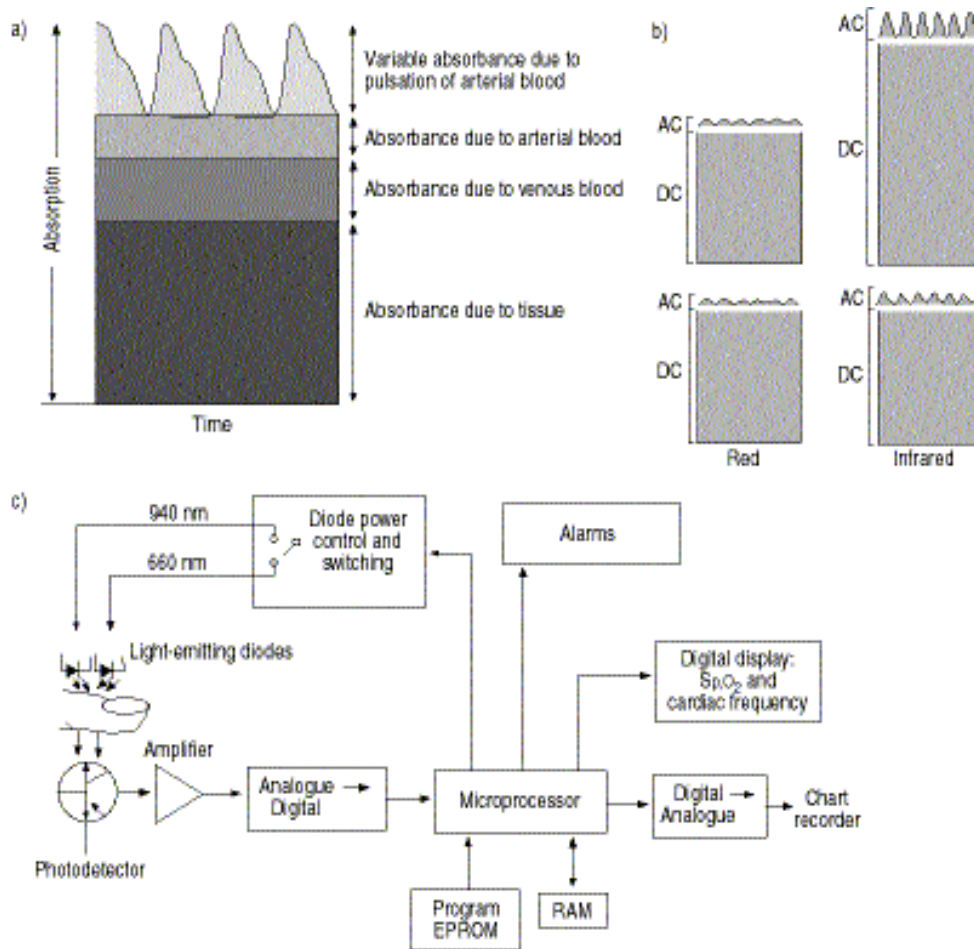


Figure 6 : Measures of the received signals are processed to yield SpO_2 values

2.1.2 Electronic circuit

Sensor circuit consists of an infrared (800 nm) light emitter diode, the SFH309FA, and an NPN phototransistor light receiver, the SFH487. The source and detector have to be mounted side by side on a finger. The light passes through the tissues and the amount of light captured by the photodetector depends on the quantity of hemoglobin. Timed samples of the photodetector output allow us to reconstruct the pulse wave.

An easy way to mount these two components to have steady contact with the finger is to use a wooden clothespin to hold them in a fixed position (see Figure 7 below). The infrared filter of the phototransistor reduces the interference from the fluorescent lights, which exhibit a large AC component in their output.

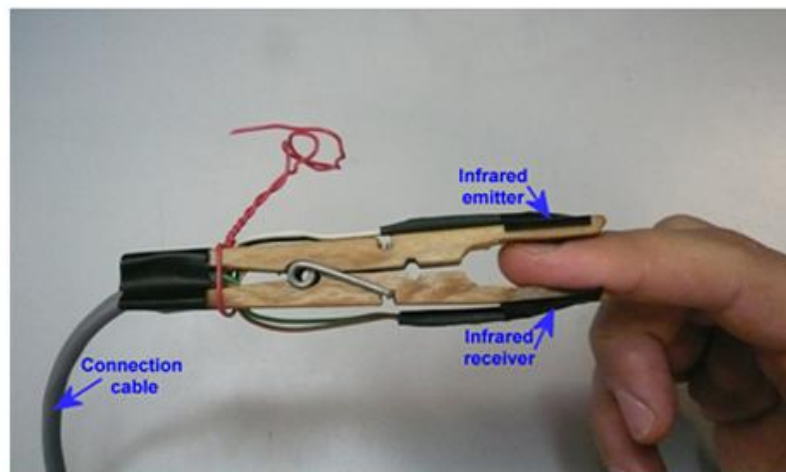


Figure 7 : Sensing system

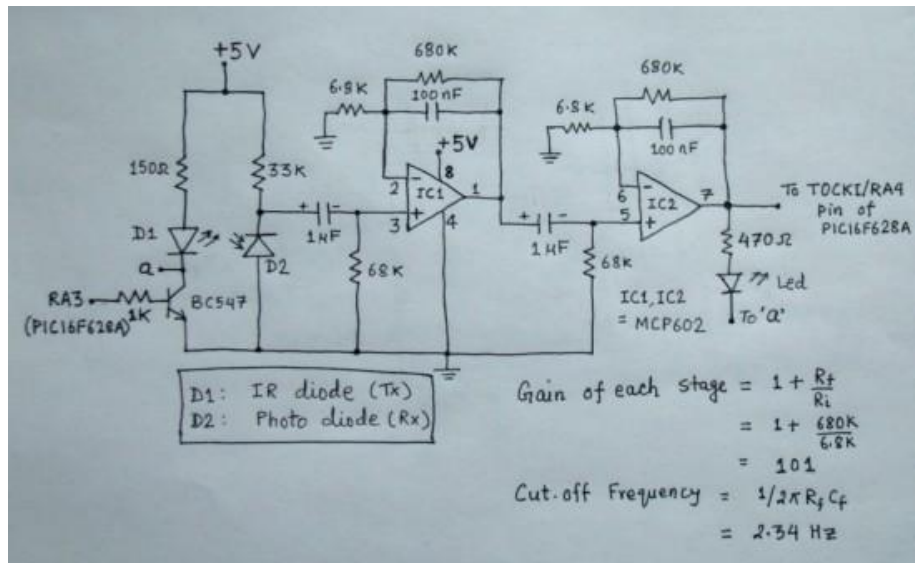


Figure 8 : Schematic diagram for the analog front end plethysmograph

The electronic circuit has four functions:

- ✓ Biasing of the LED
- ✓ Biasing of the photo-transistor
- ✓ Remove low-frequent motion-artefacts and drift
- ✓ Isolate the heartbeat pulse

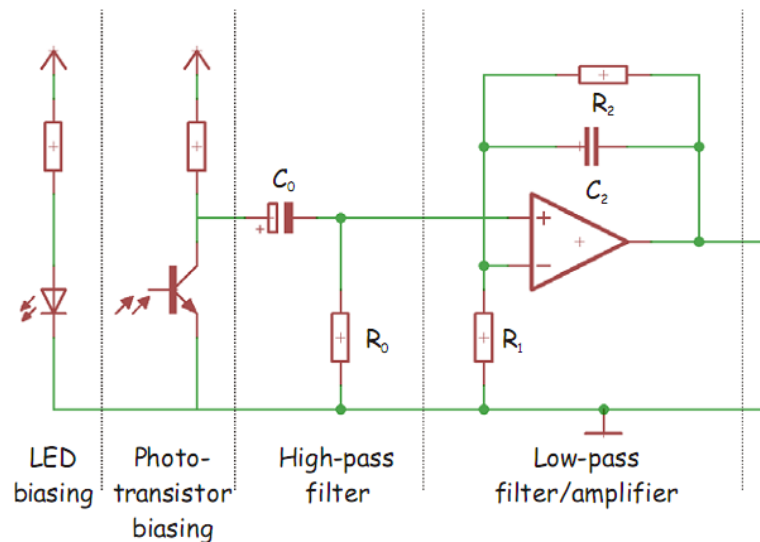


Figure 9 : Basic circuit topology

2.1.3 Theoretical Results

Experiment done by ST Microelectronics on example of the pulse wave measurement as recorded by the plethysmograph is shown in Figure 10. The experiment was carried out on a healthy subject at rest because movement of the finger causes additional compression of the blood volume which varies the DC component of the measurement. The peaks in the signals correspond to a maximum in the blood flow because the greater the voltage, the lesser the light which is received by the phototransistor and the greater the voltage between the phototransistor collector and ground. The time interval between two consecutive peaks can be easily measured by combining the A/D converter with a triggered timer of the microcontroller which computes this interval[7].

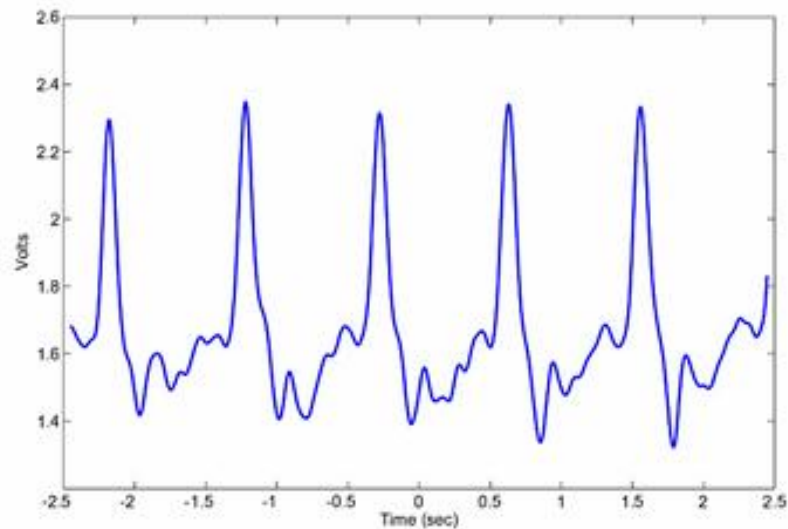


Figure 10 : Five seconds recording by plethysmograph

There are two methods suggested by Ante Fiantii and Miroslav Gaban to measure the diastolic and systolic blood pressure on the finger [8]. One of them is based on measurement of pulse time-delay between proximal cuff and distal photoplethysmograph sensor. The other method is based on pulse amplitude differences which occur during the cuff deflation. The systolic pressure can be easily determined with the sensor on the tip of the finger. The proximal cuff deflates and when the systolic pressure in the proximal cuff is reached, the first bolus of blood succeeds to reach sensor, what indicate systolic pressure. However, it is harder for determination of diastolic pressure measurement.

2.1.3.1 Pulse Time Delay Method

In the previous work of Fiantii and Miroslav Gaban mentions about the method of measuring the time delay of the transition between proximal cuff and sensor until it becomes constant [8].

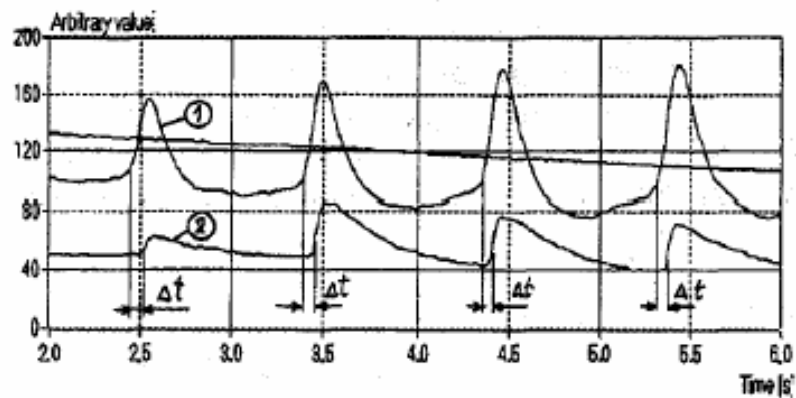


Figure 11 : Pulse pressure signal in the proximal cuff 1 and photoplethysmograph sensor 2

Refer to the figure below; the peak amplitude corresponding to the first minimum of time pulse changes is taken.

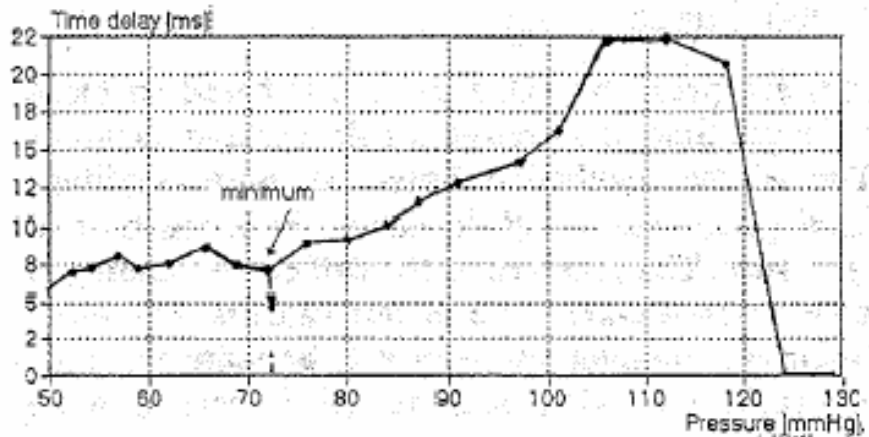


Figure 12: Pulse time delay as a function of pressure

2.1.3.2 Pulse Amplitude Difference Method

The amplitude differences (between cuff pressure and sensor) obtained after the systolic pressure is the highest systolic pressure. The amplitude differences obtained after the diastolic pressure is the lowest pressure. Refer to the figure below to see the graph of pulse signal.

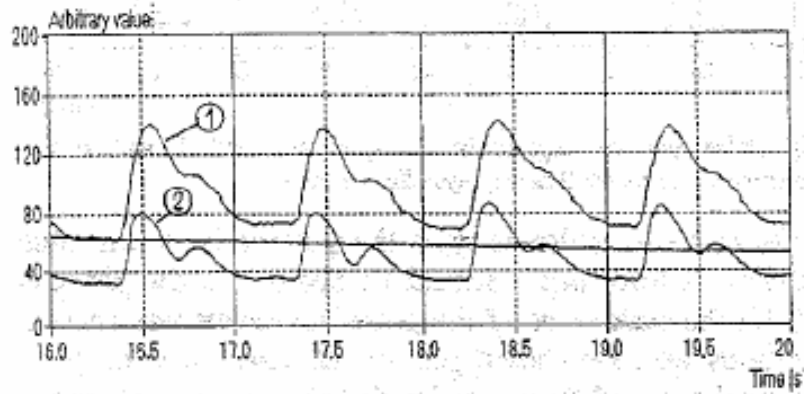


Figure 13 : Pulse pressure signal in the proximal cuff 1 and photoplethysmograph sensor 2

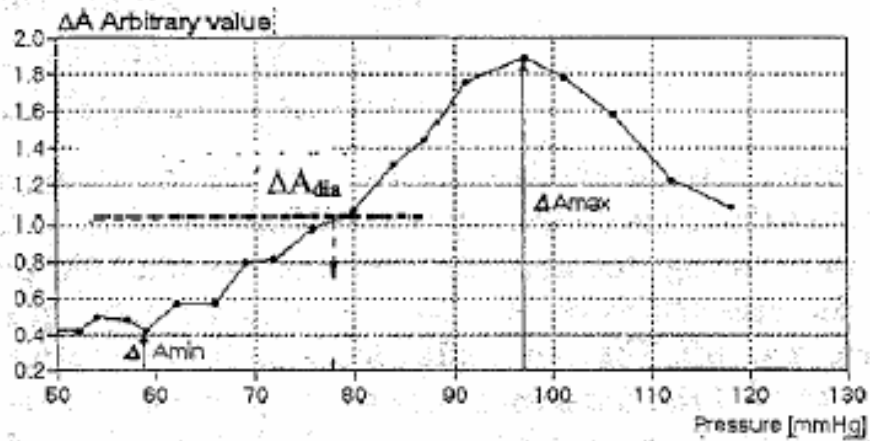


Figure 14 : Pulse time delay as a function of pressure

The mentioned methods require sampling of 10 to 20 data before the diastolic pressure is determined. Obtaining one data sampling requires about one second, hence, time to take an exact diastolic pressure is 20 seconds. This delay of measurement will be fatal to the user during the critical period.

2.2 ZigBee

ZigBee takes its name from the zigzag flying of bees that forms a mesh network among flowers. It is an individually simple organism that works together to tackle complex tasks [9]. ZigBee has built on the IEEE 802.15.4 low-rate, wireless personal area networks (WPAN) standard. The IEEE 802.15.4 defines the physical layer (PHY) and media access control (MAC) layer. The physical layer (PHY) supports three radio bands, those are individually defined 2.4GHz ISM band (Worldwide) with 16 channels, 915MHz ISM band (Americas) with 10 channels, and 868MHz band (Europe) with single channel [10], the data rates are individually defined 250Kbps at 2.4GHz, 40Kbps at 915MHz, and 20Kbps at 868MHz.

The media access control (MAC) layer controls access to the radio channel using the Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA) mechanism. The transmission range is 1-100 meters. The ZigBee defines two types of devices; those are Full Function Device (FFD) and Reduced Function Device (RFD). The FFD can serve as a network coordinator or a regular device. It can communicate with any other device. The RFD is intended for applications that are extremely simple, such as a light switch or a passive sensor device. It can communicate only with the FFD[10]. Theoretically, ZigBee can support up to 65,536 nodes. For security, ZigBee uses 128-bit Advanced Encryption Standard cryptography and trust-center-based authentication [11].

2.2.1 Network Topology

ZigBee supports Star networks, Cluster Tree networks and Mesh networks illustrate in Figure 15.

- 1) Star networks: The devices in the star topology can only communicate via the PAN coordinator.
- 2) Cluster Tree networks: Routers move data and control messages through the network using a hierarchical routing strategy.
- 3) Mesh networks: It shall allow full peer-to-peer communication [12].

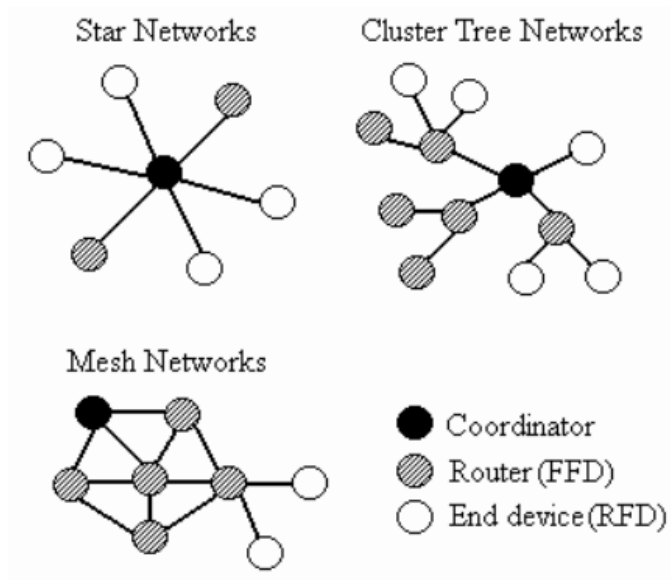


Figure 15 : ZigBee network topologies

2.2.2 ZigBee Stack Architecture

ZigBee stack architecture is shown in Figure 16. The IEEE 802.15.4 defines the physical (PHY) layer and the medium access control (MAC) sub-layer. The ZigBee Alliance builds on this foundation by providing the network (NWK) layer and the framework for the application layer (APL). The NWK and APL as the following:

- 1) Application (APL) layer: It includes application support sub-layer (APS), application framework and the ZigBee device object (ZDO). In the framework are added the user defined application object.
- 2) Network (NWK) layer: The network layer (NWK) handles the network level of the communication. It is managing the network structure and handles routing and security functions for the relayed messages [13].

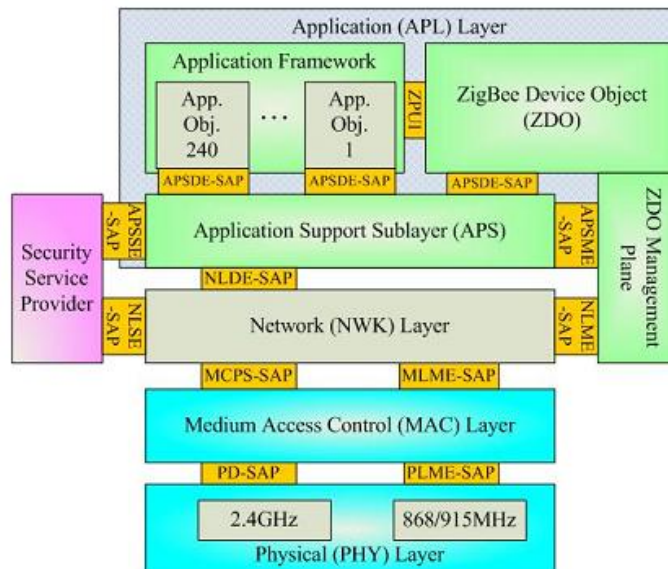


Figure 16 : ZigBee Stack Architecture

2.2.3 Comparison between ZigBee and Bluetooth

Here is some of the specification comparison between ZigBee and Bluetooth that shows ZigBee is the best medium in wireless monitoring system[14].

Table 1 : ZigBee versus Bluetooth

Characteristic	ZigBee	Bluetooth
Range as designed	10-100 meters	10 metres
Special kit outdoor	up to 400 metres	100+ metres
Data Rate	20-250 Kbps	1 Mbps
Network Latency		
New Slave	30 ms	20 s
Sleeping Slave	15 ms	3 s
Active Slave	15 ms	2 ms
Power Profile	Years	
	Optimizes slave power requirements	Maximize adhoc functionality
Security	128 bit AES	64 bit, 128 bit
Operating frequency	868 MHz, 902-928 MHz, 2.4 GHz ISM	2.4 GHz ISM
Complexity	Simple	Complex
Network Topology	Adhod,Star,Mesh,Hybrid	Adhoc Piconets
Number of devices per network	2 to 65,000	8
Scallability / Extendability	Very High / Yes	Low / No
Flexibility	Very High	Medium. Profile dependant
Resilience and reliability	Very High	Medium
Application	Remote Sensing and control	Machine to machine

2.2.4 Advantages of ZigBee

1. No configuration necessary for out-of box RF communications
2. Free X-CTU Software (Testing and configuration software)
3. AT Command Mode for simple configuration of module parameters
4. Small form factor
5. Network compatible with other ZigBee/802.15.4 devices

The XBee/XBee-PRO OEM RF Modules interface to a host device through a logic-level asynchronous serial port. Through its serial port, the module can communicate with any logic and voltage compatible UART; or through a level translator to any serial device (For example: RS-232/485/422 or USB interface board)[15].

Devices that have a UART interface can connect directly to the pins of the RF module as shown in the figure below :

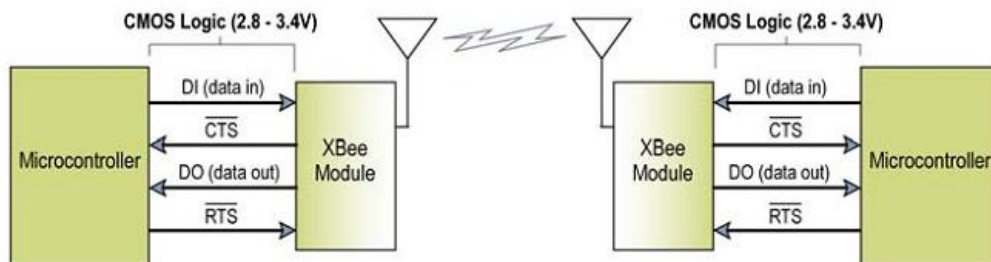


Figure 17 : System Data Flow in a UART interfaced environment

2.2.5 Tools Used

- Hardware :

- Infrared Led and phototransistor LED is used to detect the pulse rate. By passing current through the finger (which provides resistance), the resulting voltage drop can be measured [16].

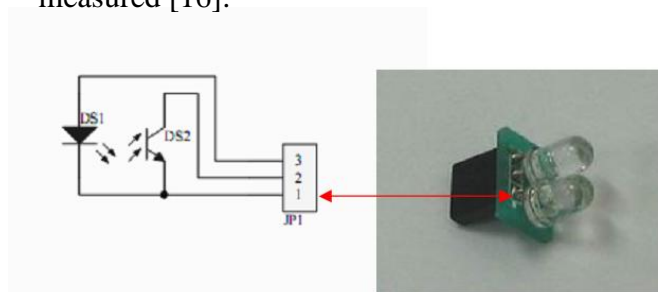


Figure 18 : Transmitter and Receiver LED

- PIC16F887 is used to control sensor and transmission module.
- Xbee transmitter is used to transmit data
- SKXbee module is used to receive data and display the result through PC



Figure 19 : SKXbee Module

▪ Software :

- MATLAB software is used to plot data from sensor and record it in form of video file (.avi).
- MikroC Pro software is used to perform programming task for PIC16F887

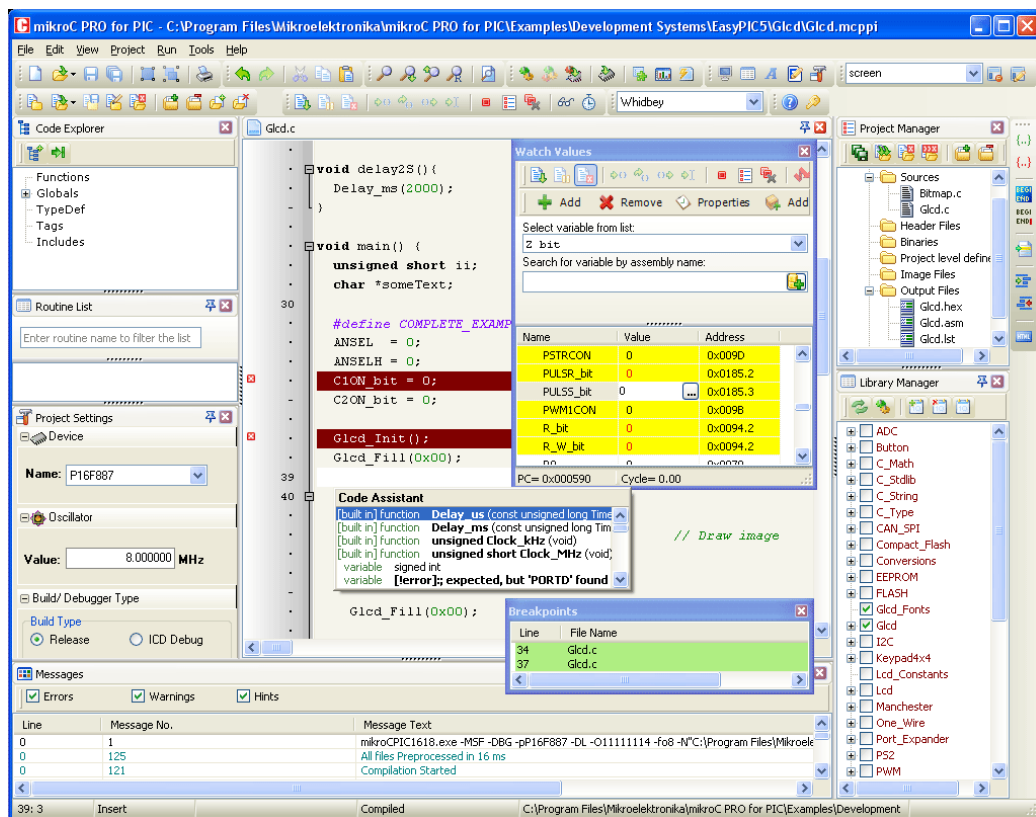


Figure 20 : MikroC Pro software

- X-CTU software act as a medium to connect Xbee device with PC system

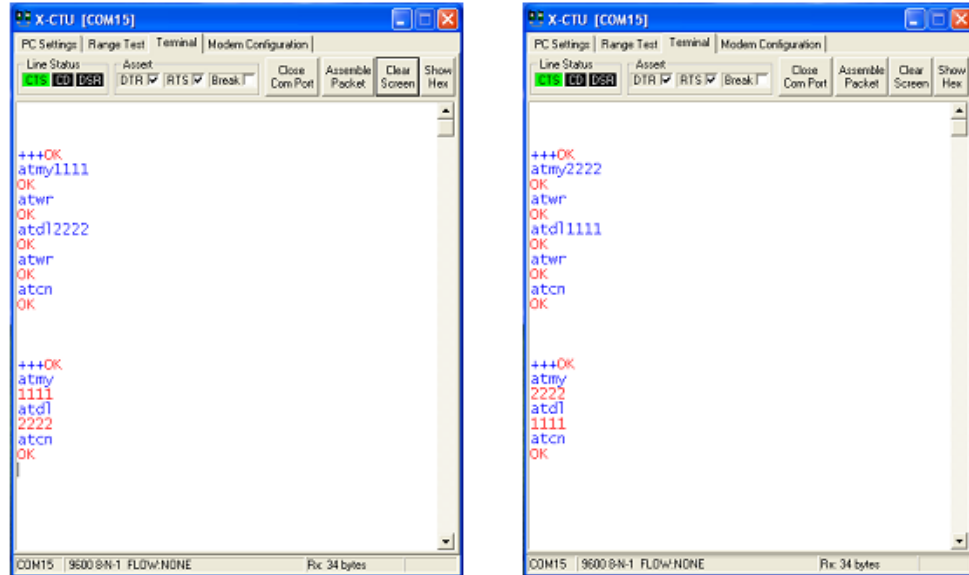


Figure 21 : XCTU software

2.2.6 Assumptions

- To develop this project, there were some assumptions made :
 - Distance between sensor and computer cannot exceed than 1000 meters. The reason is specification of Xbee module covers up to 100 meters only. Theoretically is 100 meters and in practical, it can be less than that.
 - Heart rate cannot be fully accurate due to surrounding effects.

CHAPTER 3

METHODOLOGY AND PROJECT WORK

This chapter will discuss what are the aspects required in order to develop this project. Besides that, this chapter will discuss on the design process such as material selection, electronic design, and hardware design. Methodology for material selection and hardware development also discussed in this chapter. This chapter have covered on basic configuration for transmission device plus software required to run ZigBee module. At the end of this chapter is milestone for this project.

3.1 Procedure Identification

The project activities flow is shown as below:

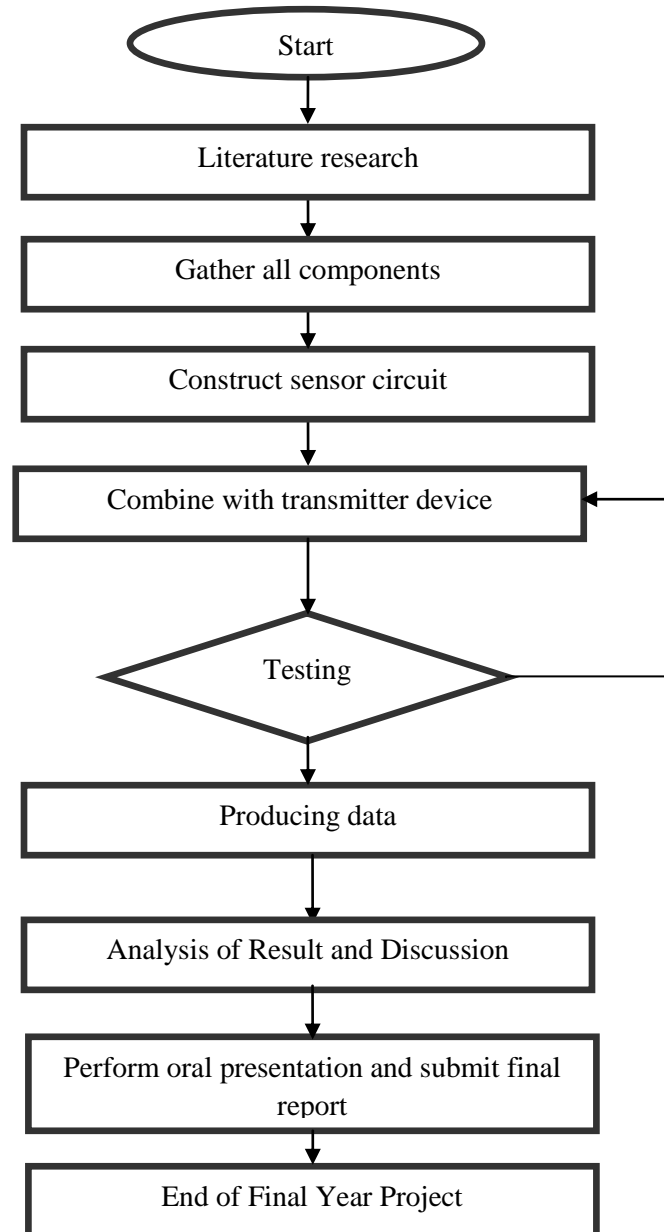


Figure 22: Flow Chart of The Project

3.2 Methodology

- ✓ Research on projects items
 - Finding proper hardware and suitable software for project

- ✓ Study on function system of ZigBee
 - Research on how ZigBee module transmit and receive signal

- ✓ Identify required hardware and software for project development
 - Hardware
 - Find suitable ZigBee device to act as a transceiver
 - Find best components needed for the sensor and transmission circuit
 - Use proper material to wrap device on user's hand

 - Software
 - Write and apply source code in MATLAB
 - Perform programming task using MikroC Pro
 - Display results of heart rate in PC using X-CTU software

- ✓ Hardware Development
 - Construct circuit for heart rate sensor
 - Test circuit on oscilloscope and PC system to verify the operation
 - Combine circuit with Xbee module
 - Run transmission process and debug any error
 - Verify data display on computer
 - Perform practical tests on user

✓ Software Development

- Study and practice programming language for programming purposes

3.3 Tools / Software / Equipment Required

1. Basic electronics components
2. IR emitter and photodiode transistor as sensor
3. Xbee transmitter and SKXbee transceiver
4. MATLAB software
5. MikroC Pro software
6. X-CTU software



Figure 23 : Xbee transmitter module

3.4 Suggested System

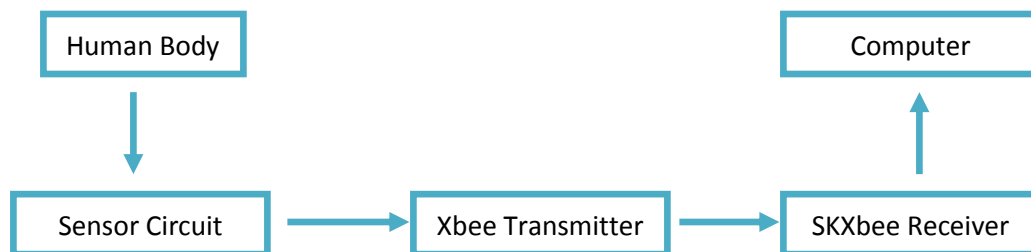


Figure 24 : Suggested system

3.5 Setting Up Xbee Network

In order for two or more XBee radios to communicate, they must

1. Have the same **channel ID**
2. Have the same **network ID**
3. The **source ID** on the receiving radio must match the **destination ID** of the sending radio.

To set the radios for point-to-point communication, there are four things to consider:

- ✓ Source ID - The Source ID is the ID number of your particular radio. You can read or write this parameter using the AT command ATMY.
- ✓ Destination ID - The Destination ID is the ID of the radio that you want to send to.
- ✓ PAN (Personal Area Network) ID is the ID of the network. Your radio will only send to radios with the same PAN ID unless you set your own ID to 0xFFFF, which will make you broadcast across all networks on the same channel.
- ✓ Channel - This is the radio channel of your XBee radio. Radios must be on the same channel in order to communicate. You can reduce interference between different XBee networks by using a different channel.

Radio has two source for IDs:

- ✓ A unique 64-bit serial number that is set at the factory and cannot be changed.
- ✓ A 16-bit ID that you can change.

3.6 Project milestone

The next goal of this project is to add more features on this project and improve its reliability. Continuous test will be conducted to observe the performance with increment of time .Below is the milestone of this project :

- ✓ Integrate blood pressure sensor into this project
- ✓ Improve distance covered for monitoring purpose
- ✓ Apply to more users
- ✓ Reliability test for long term usage

CHAPTER 4

RESULTS AND DISCUSSION

This chapter discuss the results obtain from the system. The result will be focused on output from sensor circuit. Apart from that, this chapter also will explain on controlling process done by PIC and function of Xbee as transmitter. An overall discussion of this project will be made at the end of this chapter

4.1 Sensor module

To create this sensor, it consists of a light source and photodetector. Light is shown whenever the tissues and variation in blood volume alters the amount of light falling on the detector. To see if the sensor were affected by reflected light, we can put both detector and source side by side then see the changes. Figure 25 shows the sensor module in this project.

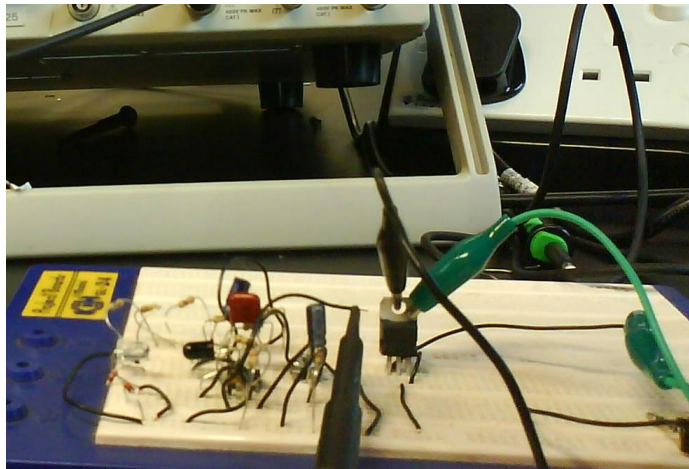


Figure 25 : Sensor module

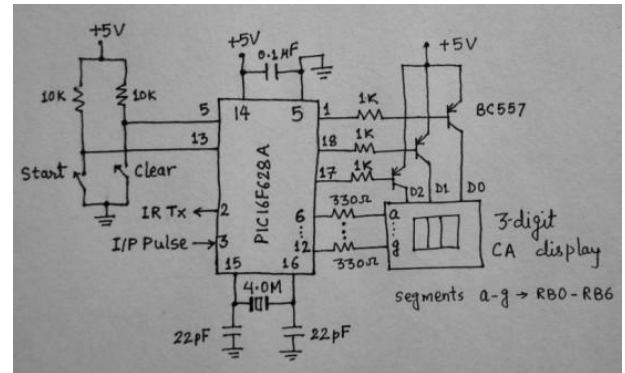
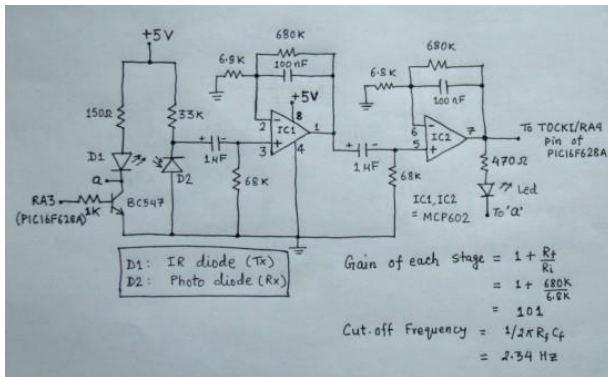


Figure 26 : Sensor circuit

Based on sensor circuit in Figure 26, if there is nothing between IR emitter and photodiode transistor, both components will be grounded and give 0 voltage. But if there is a block in between, both components will be connected and give 5v output. Figure 27 shows result from sensor when there is nothing in between the sensor.

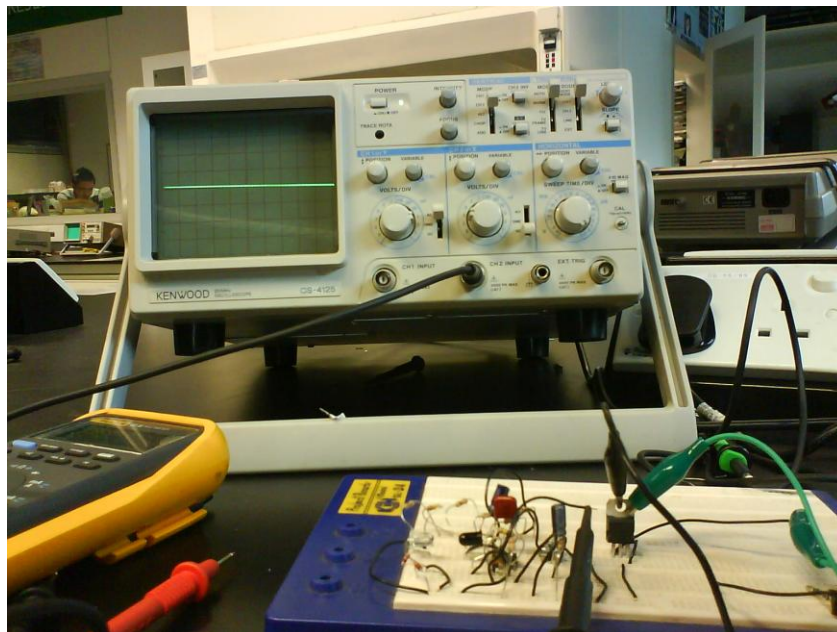


Figure 27 : Result before putting a finger

Figure 28 shows result from sensor circuit when we put a finger in between the sensor.

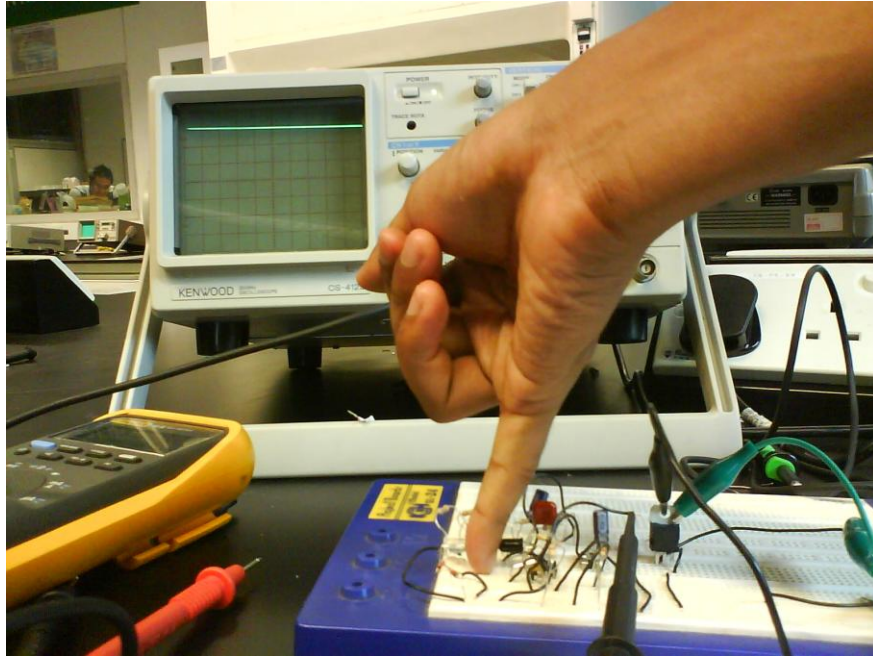


Figure 28 : Result after putting a finger

4.2 Controller System

Control system in this project will be lead by PIC16F877 in both sensor and transmission circuits. Basically, PIC will receive raw data from heart rate sensor and organize it in a proper data. Then PIC will communicate with Xbee module for transmission process. To perform this, we will write a command to PIC using MikroC Pro software to communicate with Xbee module and ready for data transmission.

The reason why PIC16F877 is that it has several features like 8 input channels Analog to Digital Module, Serial and Parallel Communications, 8K FLASH Program Memory and Timers. Below is the table for key features in PIC16F877.

Key Features PICmicro™ Mid-Range Reference Manual (DS33023)	PIC16F873	PIC16F874	PIC16F876	PIC16F877
Operating Frequency	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz	DC - 20 MHz
RESETS (and Delays)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)	POR, BOR (PWRT, OST)
FLASH Program Memory (14-bit words)	4K	4K	8K	8K
Data Memory (bytes)	192	192	368	368
EEPROM Data Memory	128	128	256	256
Interrupts	13	14	13	14
I/O Ports	Ports A,B,C	Ports A,B,C,D,E	Ports A,B,C	Ports A,B,C,D,E
Timers	3	3	3	3
Capture/Compare/PWM Modules	2	2	2	2
Serial Communications	MSSP, USART	MSSP, USART	MSSP, USART	MSSP, USART
Parallel Communications	—	PSP	—	PSP
10-bit Analog-to-Digital Module	5 input channels	8 input channels	5 input channels	8 input channels
Instruction Set	35 instructions	35 instructions	35 instructions	35 instructions

Figure 29 : Key Features for PIC16F877

4.3 Transmission System

Second stage of this project is focusing on data transmission from heart rate sensor to PC system using ZigBee technology. ZigBee module is divided into two parts which are transmission part using Xbee module and receiving part using SKXBee module.

4.3.1 Xbee Module

Basic idea to operate this device is to communicate Xbee module through a logic-level asynchronous serial port.

Through its serial port, the module can interact with any logic and voltage compatible UART or perhaps through a level translator to any serial device. Next part is to configure Xbee module using X-CTU software and a serial port to PC. Below is the sample of command line using X-CTU software :

Method 1(One line per command) :

Send At Command	System Response
++++	OK <CR>
ATDL <ENTER>	{current value} <CR>
ATDL1A0D <ENTER>	OK <CR>
ATWR<ENTER>	OK <CR>
ATCN<ENTER>	OK <CR>

Method 2(Multiple line per command) :

Send At Command	System Response
++++	OK <CR>
ATDL <ENTER>	{current value} <CR>
ATDL1A0D,WR,CN <ENTER>	OK,OK,OK <CR>

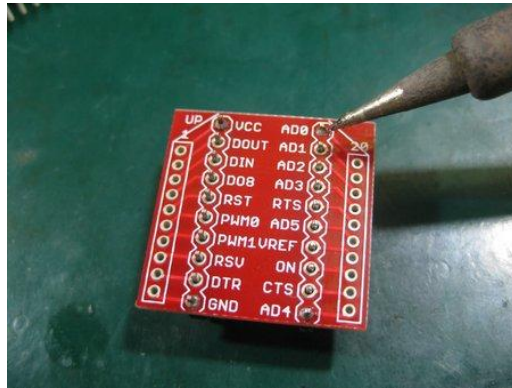


Figure 30 : Xbee module attached to socket

4.3.1 SKXbee Module

Starter Kit Xbee Module as known as SKXbee has been designed for no extra voltage divider, 5V TTL logic interface, and requires minimum interface. The major reason of why we choose SKXBee module is, this device is capable on supporting both Xbee and Xbee PRO modules and also provide USB Plug and Play UART function. USB Plug and Play UART function plays vital role in receiving part whereby it will connected with PC system to display the result.

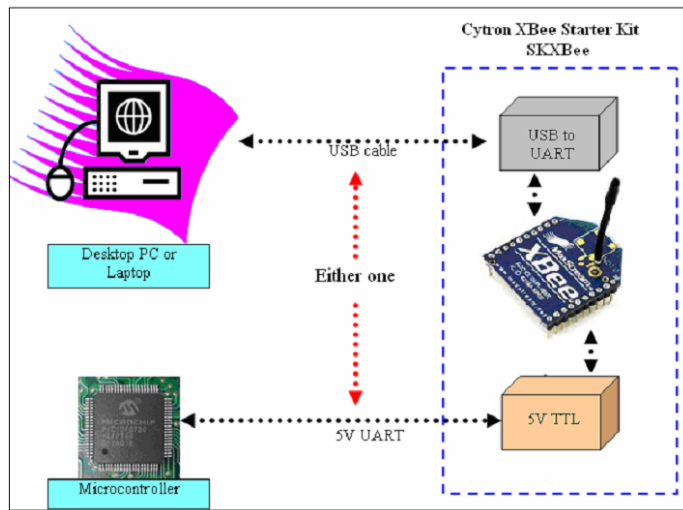


Figure 31 : Overview of SKXbee system

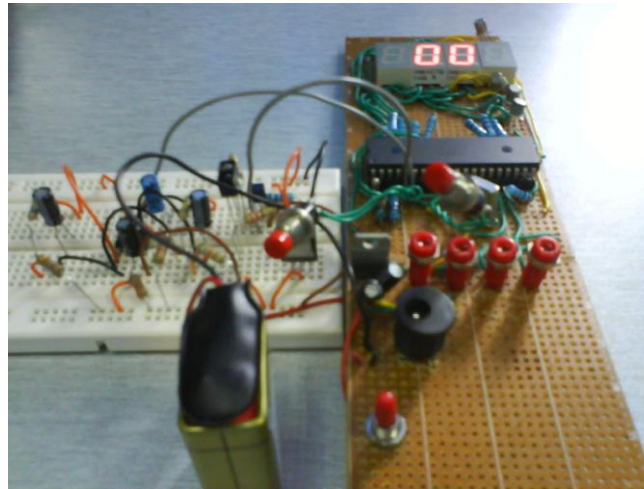


Figure 32 : Transmitter circuit



Figure 33 : SKXbee at PC system

4.4 Discussion

Many problems encountered while developing this project. Major issues on hardware system involving PIC and other devices. There are few issues arose since initial stage and some early forecast issues also occur during second phase of the project. The issues are :

- ✓ PIC cannot communicate with output signal from sensor
- ✓ Low signal from Op Amp to PIC
- ✓ PIC cannot communicate both sensor and ZigBee module

In the middle of this project heart rate sensor was working fine with PIC and 7 segment display whereby it manage to display the number of heart beat. Once we combine ZigBee configuration into PIC, it ruined overall project. To encounter this issue, we perform troubleshooting started with hardware configuration. Below is the list of troubleshooting process :

1. Perform technical checking on hardware component in case there is short circuit somewhere. In this case, there were few components burn out like transistor and IR emitter.
2. Recheck source code in PIC for any error in command. The source code is good and no error retrieve during compilation.
3. Perform signal test on PIC and sensor circuit. Theoretical gain for this circuit is 101 and it seems not enough for sensor to transmit the output.

After all of the troubleshooting process, still the circuit doesn't work in good condition. We suspect the issue came from both coding for PIC and hardware configuration.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

The purpose of this project is to demonstrate the use of wireless technology in providing a better health monitoring system. It has presented the initial idea on constructing heart rate sensor and then combine it with ZigBee module to transmit data to computer system. For Final Year Project 2 (FYP2), we have focused on data transmission from sensor circuit to PC system using ZigBee technology. Xbee module and SKXbee module have been used as the wireless medium for this project. This project shows process of integration between Xbee device as the transmitter and SKXbee device as the receiver.

This project shows high potential in the field of Biomedical technology for future use. Imagine a world with powerful monitoring system which gives benefits to human life. People have no more worries of their health issue because everything is under supervisory.

Lastly, taking into account of issues related to the completion of this project, it is hereby suggested that the project to be continued and further developed in term of hardware and software development.

5.2 Recommendations

In order to improve this project in the future, few suggestion are proposed:

- ✓ Use different types of method to detect heart rate from human body like Electrocardiography (ECG) and optical sensor. This method is more accurate but it might be tough to design.
- ✓ More research should be done for this project and discussion with hardware vendor is also needed for better understanding.

REFERENCES

- [1] M. Ishijima, "Unobtrusive approaches to monitoring vital signs at home", *Medical and Biological Engineering and Computing*, vol. 45, 11, 2007, pp. 1137-1141,.
- [2] M. Ogawa and T. Togawa, "The concept of the home health monitoring," in *5th International Workshop on Enterprise Networking and Computing in Healthcare Industry*, 2003, pp. 71-73.
- [3] [Heartbeat an indicator of disease risk: study](http://www.smh.com.au/news/national/heartbeat-an-indicator-of-disease-risk-study/2008/08/31/1220121048825.html) [Cited: September 1, 2008]
http://www.smh.com.au/news/national/heartbeat-an-indicator-of-disease-risk-study/2008/08/31/1220121048825.html
- [4] Safaric S., Malaric K., "ZigBee wireless standard", *Multimedia Signal Processing and Communications, 48th International Symposium ELMAR-2006*, Zadar, Croatia, June 2006, pp. 259-262.
- [5] Article [Regulation of Human Heart Rate](#). Serendip. Retrieved on June 27, 2007.
- [6] Pavla Holejšovská, Zdeněk Peroutka, Jiří Čengery, "Non-Invasive Monitoring of the Human Blood Pressure", *University of West Bohemia, Pilsen, Czech Republic*.

- [7] Application note, Plethysmograph based on the TS507 from ST product, STMicroelectronics. Retrieved on January 2010.

- [8] Ante Fiantii, Senior Member, IEEE, Miroslav Gaban, “Two Methods for Determination of Diastolic and Systolic Pressures in Fingers”, IEEE-EMBC and CMBEC, 1995.

- [9] Gang Ding, Sahinoglu Z., Orlik P., Jinyun Zhang, Bhargava B., “Tree-Based Data Broadcast in IEEE 802.15.4 and ZigBee Networks”, published by the IEEE transactions on mobile computing, Volume 38, No. 11, November 2006, pp. 1561-1574.

- [10] David Geer, “Users make a Beeline for ZigBee sensor technology”, published by the IEEE computer society, Volume 38, Issue 12, December 2005, pp. 16-19.

- [11] Taehong Kim, Daeyoung Kim, Noseong Park, Seong-eun Yoo, Lopez, T.S., “Shortcut Tree Routing in ZigBee Networks”, Proceeding of the Wireless Pervasive Computing ISWPC '07. 2nd International Symposium, San Juan, February 2007, pp. 42-47.

- [12] David Geer, “Users make a Beeline for ZigBee sensor technology”, published by the IEEE computer society, Volume 38, Issue 12, December 2005, pp. 16-19.

- [13] Ondrej S., Zdenek B., Petr F., Ondrej H., “ZigBee Technology and Device Design”, Proceeding of the International Conference on Networking, International Conference on Systems and International Conference on Mobile Communications and Learning Technologies, April 2006, p. 129.

- [14] Jin-Shyan Lee, Yu-Wei Su, and Chung-Chou Shen, “A Comparative Study of Wireless Protocols: Bluetooth, UWB, ZigBee, and Wi-Fi”, The 33rd Annual Conference of the IEEE Industrial Electronics Society (IECON), Taipei, Taiwan, November 5-8, 2007.

- [15] Product Manual XBee™/XBee-PRO™ OEM RF Modules by Maxstream, Inc. Retrieved on October 28, 2005.

- [16] Webster J.G. 1998, Medical Instrumentation: Application and Design (3rd Ed.). Webster J.G. (Ed). New York: John Wiley& Sons.

APPENDIX A

No.	Detail/ Week	1	2	3	4	5	6	7		8	9	10	11	12	14	15	16	
1	Troubleshoot previous issue on sensor circuit		●						Mid-semester break									
2	Combine sensor with PIC controller				●													
3	Working on command to read input from sensor					●												
4	Setup two model of Xbee. One for transmitter and another one for receiver point.						●											
5	Setup and configure Xbee receiver at PC system and Xbee transmitter sensor circuit										●							
6	Working on command to transmit data from sensor circuit to PC system											●						
7	Display graph or visual output for data collection												●					
8	Wrap up the device in a wearable form													●				
9	Perform validity test and practical test on few respondents														●			

● Suggested milestone

■ Process

APPENDIX B

MATLAB source code to plot data from sensor circuit

```
%Simple analog display
figure(2); clf;
clear all
%trash any existing analog i/o objects
delete(daqfind);

%define the input
adaptor = 'nidaq';
adaptorData = daqhwinfo(adaptor);
%get the adaptor id,
%but note that the brackets are curly {}
%because the struct is an cellarray
id = adaptorData.InstalledBoardIds{1} ;
% Create an analog input object with one channel.
ai = analoginput(adaptor, id);
ch = addchannel(ai, [0]);

dt = 0.1;
maxt = 10;
time = 0:dt:maxt;
%build a plot with zero data
lh = plot(time, zeros(1,length(time)));
set(gca, 'YLim', [-5 5]);
%build a text string on the plot
```

```

th = text(0.5,4,"");
%and draw it
drawnow

%step through time getting samples and plotting them
for index=1:length(time)
    %using getsample does not require that you start the ai object
    %EXCEPT on winsound
    sample = getsample(ai);
    %drop the new sample into the existing plot
    v = get(lh, 'ydata');
    v(index) = sample;
    set(lh, 'ydata', v)
    %update a time-counter
    set(th,'string',num2str(index*dt-dt));
    drawnow
    %delay to make dt the actual time step
    pause(dt);
end
%trash the input object
delete(ai);

```

**MATLAB source code to store result from sensor circuit in video
format file.**

```
fig=figure;
set(fig,'DoubleBuffer','on');
set(gca,'xlim',[-80 80],'ylim',[-80 80],...
      'nextplot','replace','Visible','off')

aviobj = avifile('example.avi')
aviobj.compression='cinepak';
aviobj.quality=100;

x = -pi:.1:pi;
radius = 0:length(x);
for i=1:length(x)
    h = patch(sin(x)*radius(i),cos(x)*radius(i),...
              [abs(cos(x(i))) 0 0]);
    set(h,'EraseMode','xor');
    frame = getframe(gca);
    aviobj = addframe(aviobj,frame);
end

aviobj = close(aviobj);
```

C++ Source Code for Heart Rate Sensor using 7 Segment Display

```
/*  
Project: Measuring heart rate through fingertip  
Zulhilmi Zaini, FYP2  
*/  
  
sbit IR_Tx at RA3_bit;  
sbit DD0_Set at RC4_bit;  
sbit DD1_Set at RC5_bit;  
sbit DD2_Set at RC3_bit;  
sbit start at RC6_bit;  
unsigned short j, DD0, DD1, DD2, DD3;  
unsigned short pulserate, pulsecount;  
unsigned int i;  
//----- Function to Return mask for common anode 7-seg. display  
unsigned short mask(unsigned short num) {  
    switch (num) {  
        case 0 : return 0x7E;  
        case 1 : return 0x30;  
        case 2 : return 0x6D;  
        case 3 : return 0x79;  
        case 4 : return 0x33;  
        case 5 : return 0x5B;  
        case 6 : return 0x5F;  
        case 7 : return 0x70;  
        case 8 : return 0x7F;  
    }
```



```

case 9 : return 0x7B;
} //case end
}

void delay_debounce(){
    Delay_ms(100);
}

void delay_refresh(){
    Delay_ms(5);
}

void countpulse(){
    IR_Tx = 1;
    delay_debounce();
    delay_debounce();
    TMR0L=0;
    Delay_ms(15000); // Delay 15 Sec
    IR_Tx = 0;
    pulsecount = TMR0L;
    pulserate = pulsecount*4;
}

void display(){
    DD0 = pulserate%10;
    DD0 = mask(DD0);
    DD1 = (pulserate/10)%10;
    DD1 = mask(DD1);
}

```

```

DD2 = pulserate/100;
DD2 = mask(DD2);
for (i = 0; i<=180*j; i++) {
    DD0_Set = 0;
    DD1_Set = 1;
    DD2_Set = 1;
    PORTB = DD0;
    delay_refresh();
    DD0_Set = 1;
    DD1_Set = 0;
    DD2_Set = 1;
    PORTB = DD1;
    delay_refresh();
    DD0_Set = 1;
    DD1_Set = 1;
    DD2_Set = 0;
    PORTB = DD2;
    delay_refresh();
}
DD2_Set = 1;
}

void main() {
    CMCON = 0x07; // Disable Comparators
    LATA = 0x00;
    ADCON1 = 0x7;
    TRISA = 0b00110000; // RA4/T0CKI input, RA5 is I/P only
    TRISB = 0b00000000; // RB7 input, rest output

```

```
TRISC = 0b01000000;  
TRISD = 0x00;  
//OPTION_REG = 0b00101000; // Prescaler (1:1), TOCS =1 for counter mode  
T0CON= 0b11101000;  
pulserate = 0;  
j = 1;  
display();  
do {  
    if(~start){  
        delay_debounce();  
        //PORTB = 0x0F;  
        countpulse();  
        j= 3;  
        display();  
    }  
} while(1); // Infinite loop  
}
```

Sending code for Microcontroller from Xbee

```
' serial out is on portc.6
' serial in is on portc.7
' a digital input is on portb.0
'set a constant with the baudmode 9600-8-n-1-true (non-inverted):
    true9600 con 84
' a byte to send out data:
    thisByte var byte
' set portb.0 to input:
    input portb.0
    ' set portd.1 to output for status light:
    output portd.1
configure: ' label to jump back to if configuration times out
' blink status light once on startup
    High portd.1
    pause 200
    low portd.1
    PAUSE 200
' for some reason it seems to help to send an arbitrary character first
    ' then pause for the guard time before requesting command mode
    serout2 portc.6, true9600, ["X"]
    pause 1500
' put the XBee in command mode
    serout2 portc.6, true9600, ["+++"]
' wait for a response from the XBee for 2000 ms, or start
    ' over at the configure label if no valid response comes
```

```

SERIN2 portc.7, true9600, 2000, configure, [WAIT ("OK")]
' set the PAN (personal area network) ID number
    ' this example uses 0x3330, but you'll want to choose your own
    ' unique hexadecimal number between 0x0 and 0xFFFFE
    serout2 portc.6, true9600, ["ATID3330,"]
' set the Destination High to 0x0
    ' to select 16 bit addressing mode. These addresses can
    ' be assigned and changed by sending commands from a
microcontroller
    serout2 portc.6, true9600, ["DH0,"]
' set the Destination Low (16 bit address)
    ' this example uses 0x0 for send and 0x1 for receive but you'll
    ' want to choose your own hexadecimal numbers between 0x0
and 0xFFFFE
    serout2 portc.6, true9600, ["DL1,"]
' exit command mode
    serout2 portc.6, true9600, ["CN",13]
' wait for a response from the XBee for 2000 ms, or start
    ' over at the configure label if no valid response comes
SERIN2 portc.7, true9600, 2000, configure, [WAIT ("OK")]

main:
' read the switch:
    thisByte = portb.0
' convert it to a readable ASCII value, send it out the serial port:
    serout2 portc.6, true9600, [DEC thisByte]

goto main

```

Receiving code for Microcontroller to SKXbee

```
' serial out is on portc.6
' serial in is on portc.7
' a digital input is on portb.0
'set a constant with the baudmode 9600-8-n-1-true (non-inverted):
    true9600 con 84
' set a constant for timeout while waiting for serial input
    timeout CON 2000
' a byte to receive data:
    inByte var byte
' set portb.1 to output:
    OUTPUT portb.1
    ' set portd.1 to output for status light:
    output portd.1
configure: ' label to jump back to if configuration times out
' blink status light once on startup
    High portd.1
    pause 200
    low portd.1
    PAUSE 200
' for some reason it seems to help to send an arbitrary character first
    ' then pause for the guard time before requesting command mode
    serout2 portc.6, true9600, ["X"]
    pause 1100
' put the XBee in command mode
    serout2 portc.6, true9600, ["+++"]
' wait for a response from the XBee for 2000 ms, or start
```

```

        ' over at the configure label if no valid response comes
        SERIN2 portc.7, true9600, timeout, configure, [WAIT ("OK")]
' set the PAN (personal area network) ID number
        ' this example uses 0x3330, but you'll want to choose your own
        ' unique hexadecimal number between 0x0 and 0xFFFFE
        serout2 portc.6, true9600, ["ATID3330,"]
' set the MY (16 bit address)
        ' this example uses 0x0 for send and 0x1 for receive but you'll
        ' want to choose your own hexadecimal numbers between 0x0 and
0xFFFFE
        serout2 portc.6, true9600, ["MY1,"]
' exit command mode
        serout2 portc.6, true9600, ["CN",13]
' wait for a response from the XBee for 2000 ms, or start
        ' over at the configure label if no valid response comes
        SERIN2 portc.7, true9600, timeout, configure, [WAIT ("OK")]
        main:
' get any incoming data:
        SERIN2 portc.7, true9600, timeout, nodata, [DEC inByte]
        ' light the LED if a 1 has been received
        if inByte == 1 THEN

```

HIGH portb.1

'douse the LED if anything else was received

ELSE

LOW portb.1

ENDIF

 nodata: ' label to jump to if no data was received

go to main