DESIGN AND IMPLEMENTATION OF BLOOD PRESSURE MONITORING SYSTEM

By

MUHAMMAD IZZUDDIN BIN MOHD SANI

FINAL PROJECT REPORT

Submitted to the Department of Electrical & Electronic Engineering
in Partial Fulfilment of the Requirements
for the Degree
Bachelor of Engineering (Hons)
(Electrical and Electronic Engineering)

Universiti Teknologi PETRONAS
Bandar Seri Iskandar
31750 Tronoh
Perak Darul Ridzuan

CERTIFICATION

CERTIFICATION OF APPROVAL

Design and Implementation of Blood Pressure Monitoring System

By

Muhammad Izzuddin Bin Mohd Sani

A project dissertation submitted to the
Electrical and Electronics Engineering Programme
Universiti Teknologi PETRONAS
in partial fulfilment of the requirement for the
Bachelor of Engineering (Hons)
(Electrical and Electronics Engineering)

Approved:

Ir. Dr. Nursyarizal Bin Mohd Nor	Dr. Hanita Binti Daud
Project Supervisor	Project Co-Supervisor

UNIVERSITI TEKNOLOGI PETRONAS
TRONOH, PERAK

May 2013

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.

____

MUHAMMAD IZZUDDIN BIN MOHD SANI

ABSTRACT

This project is about designing blood pressure monitoring system that integrated with wireless technology. The concept design of the system is consisted of multiple blood pressure sensors and a single computer. The blood pressure level is measured by blood pressure sensor than the information is sent through a wireless technology to the computer. Unfortunately, the project was not completed as proposed due to blood pressure measurement accuracy issue.

ACKNOWLEDGEMENTS

Alhamdulillah, with the ascendency and the consent from Allah SWT, I successfully completed this Final Year Project. Although, the objectives of this project are not completely achieved, at least these objectives were achieved partially. I am also very grateful to Allah SWT, because from this project, it is not just the knowledge that I gain from this project. Because of Allah SWT, through this project I also gain the experience to become more responsible and appreciate on what other people can contribute to me in this project.

Therefore, my highest appreciation I give to my supervisor and co-supervisor Final Year Project, Ir. Dr. Nursyarizal Bin Mohd Nor and Dr. Hanita Binti Daud for their support, ideas and guidance towards the completion of this project. Their guidance helps me to makes continuous progress in every week throughout the completion of this project.

I would also like give appreciation to all my colleagues for their support and ideas throughout the period, specifically to Nurul Fauzana Binti Imran Gulcharan, Abdillah Bin Nasib, Muhammad Afham Bin Mohd Azhar, Mohd Zulhelmi Bin Shamsudin, Muhammad Rusydi Bin Mashhor and Muhammad Zulfadhli Bin Abdul Rahman.

Other than that, I also like to thank all laboratory technologists specifically to Mr. Azhar Bin Zainal Abidin and Mr. Mohamad Yasin Bin Baharudin for their cooperation given and also the guidance for me throughout the project completion.

Last but not least, many thanks to my family members who have supported me throughout the project completion. Thank you for your continuous support and cooperation in giving and sharing some ideas for this project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABBREVIATIONS AND NOMENCLATURES

mmHg	millimetres of mercury
PTT	Pulse Transit Time
RFID	Radio Frequency Identifications
WPAN	Wireless Personal Area Networks
VDC	Direct current voltage
PWM	Pulse Width Modulation
USB	Universal Serial Bus
ICSP	In-circuit Serial Programming
ADC	Analogue-to-Digital Converter
AC	Alternate Current
DC	Direct Current
LP filter	Low Pass Filter
BP	Blood pressure
bpm	beats per minute

CHAPTER 1
INTRODUCTION

1 INTRODUCTION

The Design and Implementation of Blood Pressure Monitoring System project is proposed to ease medical monitoring where the vital signs of patients is updated automatically in a single computer rather than the vital signs need to be checked from one patient to the other patients. In this chapter, certain important points have been determined to execute the project.

1.1 Background

Every day in hospitals, there are some patients who need medical monitoring by the hospitals. Most of them who need medical monitoring are those who are required for follow-up treatment and detained by hospital before being allowed to back to their homes.

Nowadays, the monitoring tasks are done by using electronic machines that measure crucial parameters such as body temperature, heartbeat, and blood pressure. Normally, the instruments (for measure vital signs) are carried along during monitoring and certain cases these facilities are placed at every single patient. Nurses and doctors need to check the vital signs at every patient periodically. Thus, monitoring task need more concentration by nurses and doctors and may lead to tiredness.

Therefore, we proposed a system that able to monitor patients via a computer. The computer will received data from sensors that equipped on patients. These sensors are performing several tasks that measure the medical parameters. From a single computer, nurses and doctors able to monitor many patients at the same time. Other than that, this system also able to trigger alarm (alerting that some patient need attention) and keep patients record (the medical parameters).

Due to concern about final date of completing Final Year Project, we selected to specialise one medical parameter. Plus, the system will be designed for general wards. Hence, we proposed a system called Design and Implementation of Blood Pressure Monitoring System.

1.2 Problem Statement

There are several problems need to encounter in order to complete this project. The next paragraphs explained about the determined problems.

Currently, the blood pressure measurement in general wards is done manually where the blood pressure level is taken on every single patient. Plus, current method is not updates the blood pressure automatically.

In normal wards, the blood pressure level are updates every four hours. Within four hours, anything can be happen by means the critical blood pressure level (too low or too high) could be suffered by patients.

1.3 Objectives and Scope of Study

From the problem statement, several objectives or scope of study are outlined as follows:-

1. To integrate blood pressure sensor with network topology for remote monitoring.
2. To design blood pressure monitoring system that will alert (with alarm) if the blood pressure is close to critical level.

CHAPTER 2
LITERATURE REVIEW AND THEORY

2 LITERATURE REVIEW AND THEORY

This literature review and theory chapter is divided into two sections. It is better if we understand the theory of blood pressure measurement and wireless technology first. Hence, the first part will be the theory and background study. The second part is the discussion on literature review of relevance papers.

2.1 Theory and Background Study

In this theory and background study section, it is divided into four subsections. The first subsection is about the description of blood pressure. The second subsection is describing on how the blood pressure is measured. Next, it is the explanation about the methods or techniques used for measuring blood pressure electronically. Lastly, the last subsection is explaining possible ways on integrating blood pressure sensors with computer.

2.1.1 What is Blood Pressure?

Blood pressure is a pressure that exerted on arteries where the blood flows through it and spread to the whole body. Due to this fact, blood pressure also called as arterial pressure [**1**] [**2**].

The early medical practitioners found that the heartbeat is related to measurement of blood pressure. Blood pressure is measured by determining the systolic pressure and diastolic pressure. Systolic pressure is determined when the heart beating and diastolic pressure, it is determined when the heart relaxing [**1**].

To be specific, the heartbeat mechanism is a cyclic mechanism. It is started from ventricle contraction (both right and left simultaneously) and immediately followed by atrium contraction (both right and left simultaneously). At ventricle contraction stage, the systolic pressure is determined. Then, it is followed by relaxing stage where the diastolic pressure is determined. After that, the cycle resumes with ventricle contraction again and so on [**3**] [**4**] [**5**].

TABLE 1 The blood pressure chart according to age group [**6**].

Age	Range	Systolic (mmHg)	Diastolic (mmHg)	Median Normal Systolic/Diastolic (mmHg)
15 to 19	Minimum	105	73	105/73
	Average	117	77	117/77
	Maximum	120	81	120/81
20 to 24	Minimum	108	75	108/75
	Average	120	79	120/79
	Maximum	132	83	132/83
25 to 29	Minimum	109	76	109/76
	Average	121	80	121/80
	Maximum	133	84	133/84
30 to 34	Minimum	110	77	110/77
	Average	122	81	122/81
	Maximum	134	85	134/85
35 to 39	Minimum	111	78	111/78
	Average	123	82	123/82
	Maximum	135	86	135/86
40 to 44	Minimum	122	79	122/79
	Average	125	83	125/83
	Maximum	137	87	137/87
45 to 49	Minimum	115	80	115/80
	Average	127	84	127/84
	Maximum	139	88	139/88
50 to 54	Minimum	116	81	116/81
	Average	129	85	129/85
	Maximum	142	89	142/89
55 to 59	Minimum	118	82	118/82
	Average	131	86	131/86
	Maximum	144	90	144/90
60 to 64	Minimum	121	83	121/83
	Average	134	87	134/87
	Maximum	147	91	147/91

Healthy blood pressure level are various from a person to persons. According to InternationalDrugMart.com, many factors affect the healthy blood pressure level such as age, gender, and physical size [**6**].

2.1.2 How blood pressure is measured?

Nowadays, several methods used by medical practitioners to measure blood pressure. The blood pressure can be measured manually by some early devices (non-electronic) or measured by electronic devices. However, most of the medical practitioners used electronic devices today since it is easy and reliable to measure blood pressure. These electronic devices are perfected by number of inventors who is dedicated to invent the way of measuring blood pressure and were enhancing it from a generation to other generation.

From the history, it can be listed two major techniques on measuring blood pressure which are non-invasively and invasively. From these two major techniques, it is develop into numbers of techniques. The history of early blood pressure measurement techniques (non-electronic) can be tracked between 1733 and 1905. However, between the periods, only non-invasive techniques are safe for human use. Nevertheless, the invasive technique is become safer today as the technique coupled with electronic device such as implanting pressure transducer into an artery. Plus, the technique is applied for special cases in particular hospitals only [**2**] [**4**].

Today, most of the non-invasive blood pressure measurement devices are based on an invention of an Italian, Riva-Rocci. This device paired with auscultatory technique introduced by a Russian surgeon, Dr. Kororkoff in 1905. Or at least, if a medical practitioner does not have a stethoscope, he or she still can measure the blood pressure using palpation technique just like the demonstration done by Riva-Rocci in 1896. However, this technique is only able to measure systolic pressure [**4**].

Furthermore, the auscultatory technique (paired with mercury sphygmomanometer) is considered gold standard for non-invasive blood pressure measurement [**7**].

2.1.3 How existing Electronic Blood Pressure Monitor works?

Recent years, many electronic blood pressure monitor marketed as hospital and home usage. Most of them are based on non-invasive oscillometric technique. It is difficult to trace the origin or history related to this technique or time began where this technique is began to grow in one of the techniques to measure blood pressure. From the references collected, it seems that this technique quite related to kymograph by Carl Ludwig in 1847. The kymograph records the oscillation of the artery pulsation. Later, it has been improved by Vierordt (in 1855) and Marey (in 1860). Similar to oscillometric technique, this method exploit the pulsation of the pressured artery [**4**] [**8**].

Usually, oscillometric devices consist of inflatable cuff and pressure sensor. The devices use the pressure sensor to measure pressure and read the pulsation of the pressured artery [**5**].

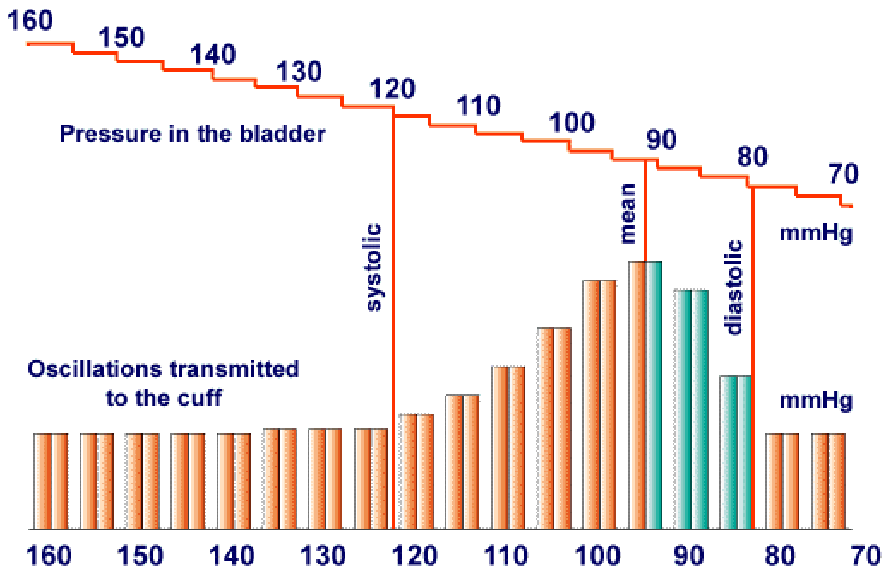


FIGURE 1 The working principle of oscillometric technique [**9**].

The working principal of the oscillometric technique is beginning with wearing inflatable cuff on a subject. Then, the cuff is inflated exceeding the systolic pressure range. After that, the cuff is deflated slowly. The systolic pressure is determined when a significant oscillation appeared. Throughout the deflation, there will be a maximum magnitude of oscillation, at this point it is the Mean Artery Pressure (MAP). Lastly, the diastolic pressure is determined when the last significant oscillation detected before the cuff is deflated completely [**9**].

The other technique to measure blood pressure electronically is Pulse Transit Time technique (PTT). This technique is introduced to design a cuff-less blood pressure monitor. Two types of pulse wave are usually paired and PTT is extracted from the both types. PTT is the time between two pulse waves propagating on the same cardiac cycle from two separate arterial sites. This technique is able to determine the systolic and diastolic pressure continuously too [**5**].

2.1.4 How to integrate Electronic Blood Pressure Monitor with Computer wirelessly?

The system that will be designed is a blood pressure monitoring system where all data about blood pressure of patients can be monitored in a single computer. The integration between electronic blood pressure monitor with computer is actually the communication between blood pressure monitors with computer. To ensure communication wirelessly between sensors and computer several methods has been found.

To think about transmitting data of a group of people blood pressure data, these data need to specify on an identity reflecting a person on that group. Therefore, RFID (Radio Frequency Identification) method might suitable of this project. The RFID method allows a certain subject or tag can be identified its identity automatically. There are two types of RFID tags which are passive RFID tag and active RFID tag. The passive RFID tag is a tag that powered by external source or RFID reader. Meanwhile, the active RFID tag is a tag that have internal power source like battery. In term of communication range between these two types, the active RFID tag has longer communication range compare to passive RFID tag. Hence, the active RFID tag is more suitable than the passive. However, the active RFID tag require internal power source like battery which may cause the short period of usage. Because the battery could not be lasted long until it needs to recharge back. Nevertheless, it is possible this problem can be overcome [**10**] [**11**].

The disadvantage of RFID leads to find other alternative of wireless data transmission which is the ZigBee. The ZigBee technology is the wireless communication between nodes, allows a personal computer share some information with other personal computers within a WPAN (wireless personal area networks). The low power consumption and simplicity of the Zigbee technology leads an experiment on Zigbee technology in this project. A Zigbee module which is compatible with Arduino board has been purchased and tested using simple codes provided by Arduino. Further discussion on the experiment will be discussed in Chapter 4 [**12**].

There is also a research done by Bonifacio Castaño and Maria Rodriguez-Moreno (2010) where the RFID and Zigbee are feasible to be combined. By this method, the RFID and Zigbee technology will be work more effective and efficient for wireless data transmission [**13**] [**14**].

2.2 Literature review

This section is divided into two subsections which are blood pressure measurement related papers and wireless technology selected papers.

2.2.1 Blood Pressure Measurement relevance papers

Recently, a lot of researches have been done to develop a reliable continuous blood pressure measurement system. The continuous BP measurement will make the BP measurement system can be updates continuously and alerts immediately if any critical BP level detected on patients. However, the developed continuous BP measurement techniques still not reliable enough to do BP measurement as the oscillometric method and the gold standard technique, the auscultatory method using mercury sphygmomanometer are chosen in measuring BP.

One of the techniques is the PTT techniques. There have been findings done where the PTT techniques developed are not reliable for BP measurement. Therefore, a clinical evaluation was made by C Dounima, CU Sauter, and R Couronne (2009). It was a controlled clinical evaluation on many PTT techniques. This clinical evaluation is done on 22 samples (sedated patients) but only 14 samples are suitable for the analysis. The data obtained is about 240 hours of measured vital signs using different pairs of signals type. The types signal or wave taken into this research are electrocardiogram (ECG), photoplethysmogram (PPG), invasive blood pressure (IBP), bioimpedence cardiogram (ICG), and bioimpedence plethysmogram (IPG). This research found that the PTT techniques are not ready for BP measurement where the smallest average error achieved is 4.91 mmHg (ECG and IBP pair) [**15**].

Another research was done by M K Ali Hassan, M Y Mashor, A R Mohd Saad, and M S Mohamed (2011), developed a portable continuous BP monitoring kit. The BP monitoring kit is used ECG only for measure BP level continuously. The continuous BP monitoring kit was used the BP measurement by mercury sphygmomanometer which taken early before the prototype use it as reference for measure BP level continuously. The portable continuous BP monitoring kit is working based on a neural network model. Before the portable continuous BP monitoring kit is developed, the ECG signal and BP data is taken form 20 subjects (aged from 15 years old to 60 years old) for training the neural network model. The results from the study, prove that the portable continuous BP monitoring kit is suitable for BP measurement but require more studies to encounter error average of -0.4712 mmHg with standard deviation of 2.204 mmHg [**16**].

Other PTT technique was tested in a research done by Heiko Gesche, Detlet Grosskurth, Gert Kuchler and Andreas Patzak (2011). In order to improve the continuous BP measurement, the aim of this research is to develop an effective function which is based on relation between systolic BP and Pulse Wave Velocity (PWV). The PWV is obtained from PTT which produced by ECG and PPG. Several experiments have been done on 63 volunteers and data from 13 volunteers was selected to find the relationship of PWV and systolic pressure. The volunteers require to ride on different loads of bicycle ergo meter. At the end, the study found the significant relationship between PWV and systolic pressure. However, the error recorded after comparison with mercury sphygmomanometer (auscultatory method) reached 20 mmHg. Therefore, more study required to improve the PWV and systolic BP relation [**17**].

Other than that, other continuous BP measurement system was developed by Md Manirul Islam, Fida Hasan Md Rafi, Abu Farzan Mitul, Mohiuddin Ahmad, M A Rashid, and Mohd Fareq bin Abd Malek (2012). These authors successfully developed continuous BP monitoring system using PPG technique. The high intensity Light Emitting Diode (LED) and Light Dependent Resistor (LDR) is used in this project where both components are attached oppositely on a finger. The systolic pressure is determined when the least light intensity received by LDR and diastolic BP is determined when the maximum light intensity received. This is due to the volume of blood in the artery is maximum when systolic pressure causing the maximum light absorption and makes light intensity received by LDR is minimum. This happens vice versa when diastolic pressure. Nevertheless, the system requires often calibration every time needs to measure BP of a person. This is because every person has different finger and artery sizes. Lastly, the system has proven reliable although the error is up to 4 mmHg when it is compared with mercury sphygmomanometer [**18**].

As conclusion, the oscillometric method will be used for this project as this method found to be more feasible compare to other methods. The complexity of continuous BP measurement techniques make the techniques become less feasible for this Final Year Project.

2.2.2 Wireless Technology relevance papers

There was wireless technology for health monitoring system. Many researchers contributed on various study in this specified bio-medical engineering field. One of the researches is done by Wun-Jin Li, Yuan-Long Luo, Yao-Shun Chang, and Yuan-Hsiang Lin (2010). They successfully developed a wireless BP monitoring system which consists of a BP monitor and a computer. It is one-way communication between the BP monitor and the computer. The BP monitor is based on ARM controller while the data transmission is based on Zigbee technology. This system allows a user to see the blood pressure variation in a line chart. Hence, this system is very convenience for personal health management [**19**].

Other study on wireless technology which was made by Bonifacio Castano and Maria Rodriguez-Moreno (2011), where a hybrid wireless system made combining Zigbee and RFID. This hybrid system is able to monitor people movements inside a building. By this capability, it will benefit a building visitor-based activity like an exhibition hall. The system will generate data on people movements and this data will used by the building management team to increase the attractiveness of an exhibition. The system consists of RFID tags (for exhibition hall example, the RFID tags will be given to visitors), RFID detector, and Zigbee wireless network topology [**13**].

Active RFID tags known for the large power consumption. To encounter this problem, a research done by Qingbin Meng and Jie Jin (2011), proposed a design to lower the power consumption of active RFID tags. The Low-Power Active RFID tag consists of battery, microcontroller, co-processing digital circuit transceiver, and an antenna. The co-processing digital circuit is the key for this low-power active RFID tag. This component will blocks any incorrect information received, so that the microcontroller can stay at sleep mode. Thus, ensure the lower power consumption [**11**].

This project is quite similar to a system which developed by Wun-Jin Li, Yuan-Long Luo, Yao-Shun Chang, and Yuan-Hsiang Lin (2011). The only difference is this project will have multiples blood pressure sensors or tags. The communication also is in multiple-ways into a single computer instead of one-way [**19**].

In this project, the Arduino board is used as the controller of the project system. Therefore, further findings about wireless technology that compatible with Arduino board has been made. It is found there are a Zigbee module that can be attach easily on Arduino board. The discussion about the Zigbee module will be discussed further in Chapter 4 [**12**].

CHAPTER 3
METHODOLOGY AND PROJECT WORK

3 METHODOLOGY AND PROJECT WORK

In this chapter, it is divided into two parts which are methodology part and project work. The methodology part is about the procedure of the project flow. Lastly, the project work part explains hardware components for the project.

3.1 The Methodology

The methodology is used as guideline system for this project. The following flow chart shows the methodology for this project.

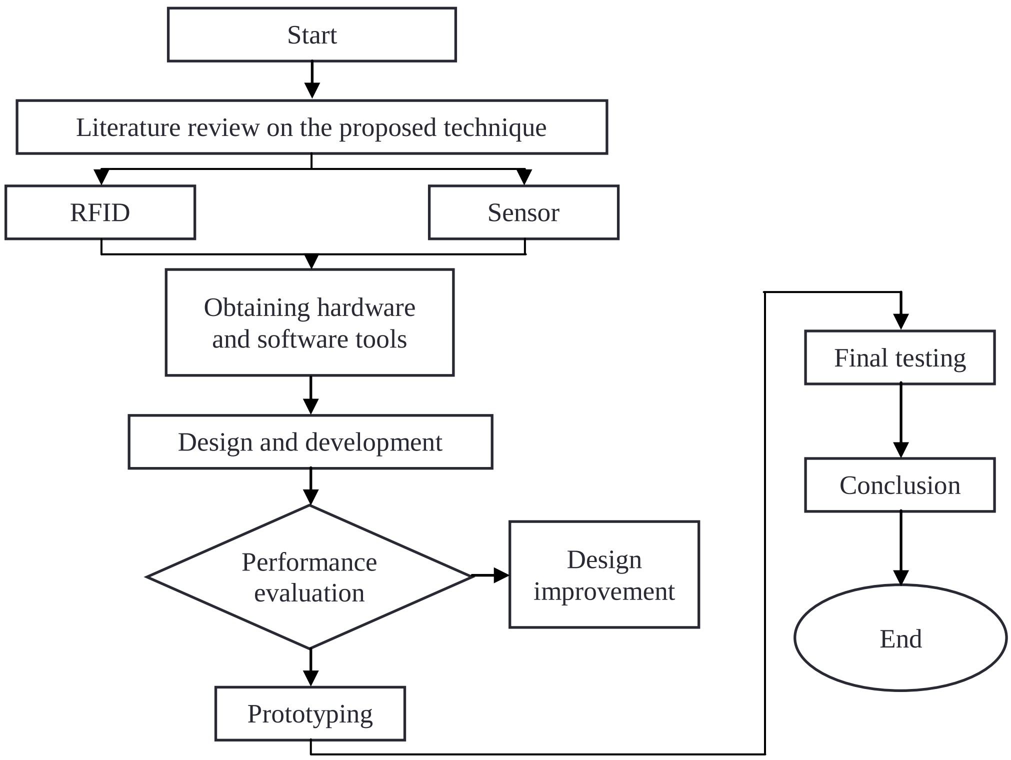


FIGURE 2 The flow diagram of methodology.

3.2 Project Work

In this section, the hardware components will be explained the functionality of major components. Before that, the process flow of the concept design will be described where it is proposed based on background study, literature review and certain experiments (on Freescale pressure sensor).

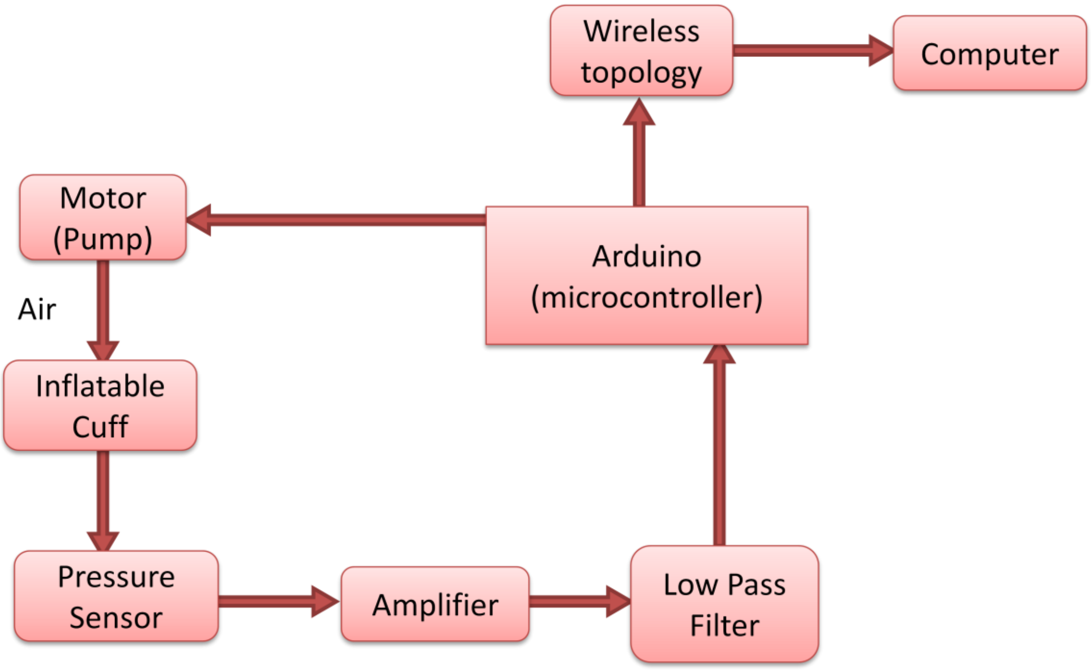


FIGURE 3 The diagram shows the process flow of the concept design.

There are three main parts of the system which are the Blood Pressure sensor, Wireless part and Computer. The BP sensor is consisting of Arduino, pump, inflatable cuff, pressure sensor, amplifier circuit, and low pass filter circuit.

The process will begin with the inflating the inflatable cuff by motor pump which controlled by Arduino. Then, the pressure sensor will measure the pressure inside cuff. The sensor will produce voltage signal which reflect the pressure and oscillation of the pressurised artery. Then, the signal will be amplified by an amplifier before it is filtered by the low pass filter. The amplified and filtered signal then, received by Arduino to determine the blood pressure. Certain programming codes is prepared, so that the Arduino able to determine the blood pressure. At last, the determined blood pressure readings is sent to a computer by a wireless topology.

3.3 Support Tools

There are several important tools that commonly used during the development of this project. These tools can be divided into two major groups, which are hardware tools and software tools. The list and the description of the important tools are shown in Table 2 and Table 3.

TABLE 2 The list of the important hardware tools and the description.

No	Tools	Description
1	Electronic Pressure Gauge	For calibration of the designed BP sensor.
2	DC Power Supply	To supply DC voltage.
3	Digital Multimeter	To measure electrical parameters.

TABLE 3 The list of the important software tools and the description.

No	Tools	Description
1	Open source Arduino Development software	For writing and compiling the programming codes. To upload the programming codes into Arduino board.
2	Microsoft Excel	For analysing the BP data and simulate the data before the programming codes is written.
3	Multisim	To simulate the electronic circuit design before the circuit is built.
4	Pspice	To simulate the electronic circuit design before the circuit is built.

3.4 The Hardware Components

In this subsection, the description of important hardware components for this project will be explained. The following subsections explain some of the main hardware parts. The parts are pressure sensor, air valve, motor pump and microcontroller.

3.4.1 Pressure Sensor

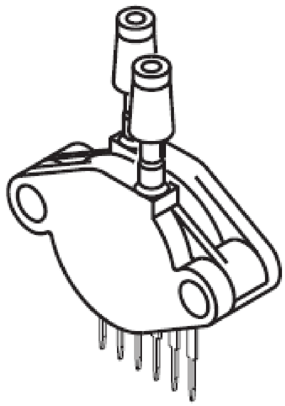


FIGURE 4 The MPX5700DP Pressure Sensor is manufactured by Freescale Semiconductor, Inc.

The pressure sensor is used to measure the air pressure inside the inflatable cuff. The output is in analogue electrical signal representing the exerted pressure. This has been done by the piezoresistive transducer of the sensor. The piezoresistive transducer is actually in a diaphragm form, which it will stretch due to difference pressure and produced certain electrical signal. The stretched piezoresistive transducer wills affect the resistance value hence the output voltage will also differ from applied voltage. This process occurred according to bridge circuit principle [**20**].

As explained before, the output of the pressure sensor is amplified and filtered by amplifier and low pass filter. This is because the output of the pressure sensor is small for blood pressure level (up to 300 mmHg or about 50 kPa) as the Freescale pressure sensor able to measure pressure in the range between 0 kPa and 700 kPa. From the amplified and filtered output signal, the microcontroller wills analyses and determines the systolic and diastolic pressures [**21**].

3.4.2 Motor Pump & Solenoid Air Valve

The motor pump is used to increase air pressure and air valve is used to hold air pressure in the cuff. There are two different rating for both components, which are rated at 6 VDC and 3 VDC. Initially, the design of BP sensor was expected to use 6 V power supply along with wireless circuit. However, further reviewed on various wireless technology like Zigbee and RFID, found that these wireless technology able to operate at lower voltage. Since then, the overall design (see Figure 7) is suggested to use 5 V power supply (use voltage regulator for 3 VDC rated components) instead of 6 V power supply. This will allows the BP sensor to operate with single power supply instead of using two different power supplies.

Despite the changes suggested, the 6 VDC rated components are commonly used throughout the development process. This is because these components have been used from early stage, and it is not completed yet due to obstacles that we faced.

FIGURE 5 The motor pumps and air valve. The left side is the components rated at 3 VDC, and the right side is the components rated at 6 VDC.

3.4.3 Microcontroller

In this project, Arduino Duemilanove is used as controller. The Arduino Duemilanove is a microcontroller board based on Atmega328. It has 14 digital input/output pins, 6 analogue inputs, a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. Notes that out of 14 digital input/outputs pins, there are 6 of them can be used as PWM outputs. The analogue inputs are connected with built-in ADC where capable to convert or quantize 0 V to 5 V analogue input into 1024 level digital input (0 to 1023, 9 bits) [**12**].

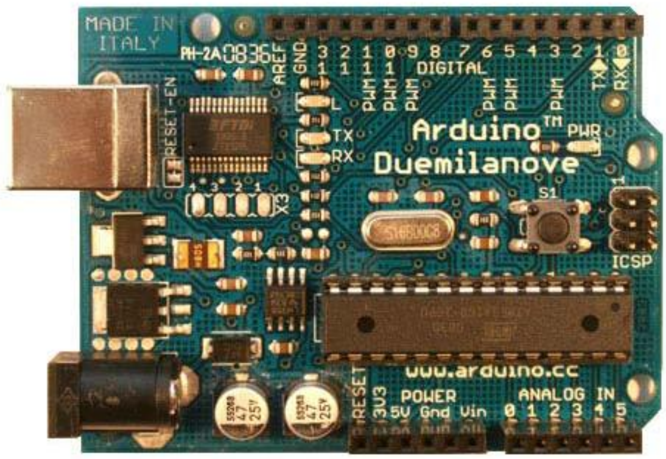


FIGURE 6	The Arduino Duemilanove board.

CHAPTER 4
RESULTS AND DISCUSSION

4 RESULTS AND DISCUSSION

In this chapter, the matters about the technical information of the project and the results will be discussed.

4.1 Results

Throughout the period of the given project time frame, the project is not completed as proposed concept design. This is due to the BP measurement accuracy issue. The following subsections will be discussed more about the designed BP sensor. Currently, the wireless part has been tested using simple codes, while the computer is used for displaying readings from BP sensor.

4.1.1 Overall Circuit

The overall circuit is consisting of Freescale pressure sensor, amplifier circuit (using LM741 operational amplifier), simple low pass filter circuit, solenoid air valve, motor pump, relays, 6 V power supply and Arduino board. The computer which is used to display readings is connected by USB socket on Arduino board. Several components and configurations were tested before, to determine the suitable components for the BP sensor. Hence, some of the parts of the overall circuit are facing changes.

One of the components that required changes is operational amplifier. In earlier stage, the LM358 operational amplifier is used. The changes of operational amplifier are due to the limitations of LM358. The LM358 is able to amplify the output voltage up to 3.66 V. Therefore, the output voltage range is not wide. Hence, the changes to LM741 is necessity where the LM741 operational amplifier able to amplify up to 4.18 V.

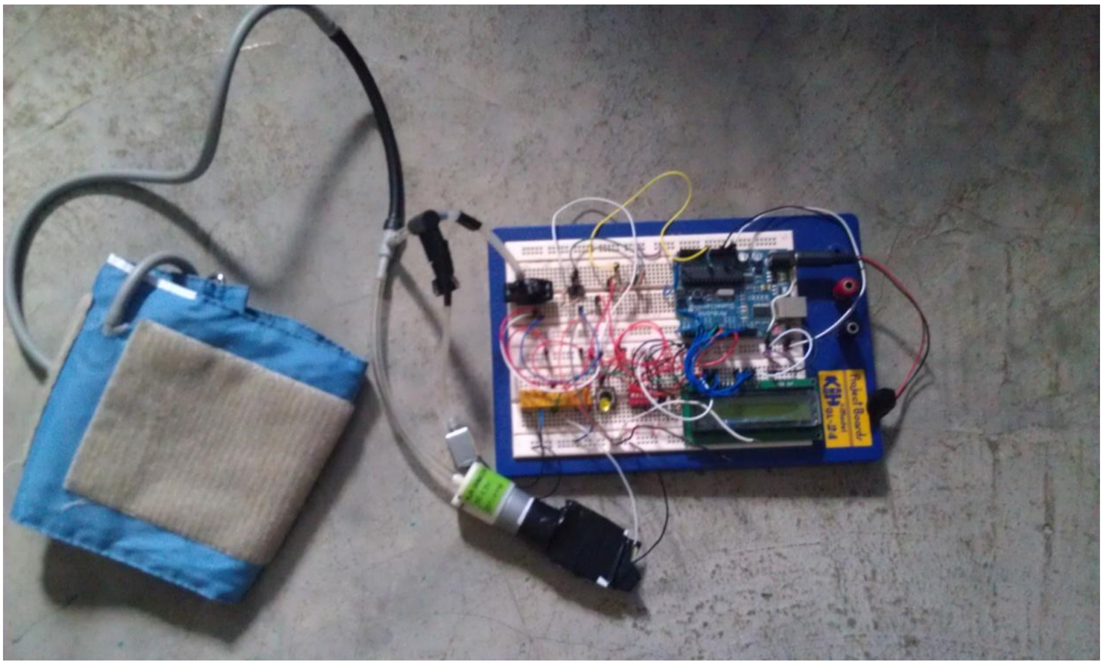


FIGURE 7 The overall design of BP sensor on a testing board.

Other than that, the Low Pass filter design also has facing changes throughout the project development. Early conceptual design, the output signal of pressure sensor was expected to have AC and DC components. This leads the early design for LP filter was designed for AC signal. However, after studying the Freescale pressure sensor mechanism, the pressure sensor output voltage only in DC type voltage signal.

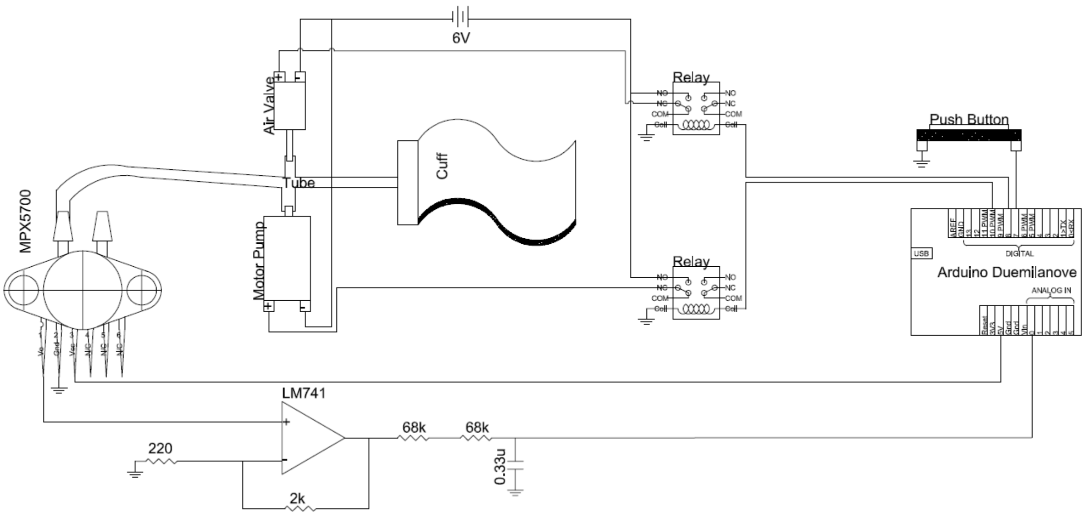


FIGURE 8 The overall wiring diagram of BP sensor.

This overall design was using separated 6 V power supply because the solenoid air valve and motor pump rated at 6 VDC. Therefore, the 6 V power supply is required as an external power supply since the Arduino board only capable to supply 3.3 V and 5 V. Note that, the LM741 operational amplifier uses two power supplies which are 5 V (positive terminal) from the Arduino board and 6 V (negative terminal) form 6 V power supply [**12**].

As explained before, the pressure sensor will produce an output signal reflecting the pressure inside the cuff. Then, the signal will amplify and filter before captured by Arduino. The Arduino will determine the systolic and diastolic pressure by the programme codes. After that the result will be displayed in Arduino Serial Monitor (on computer).

4.1.2 Amplifier Circuit

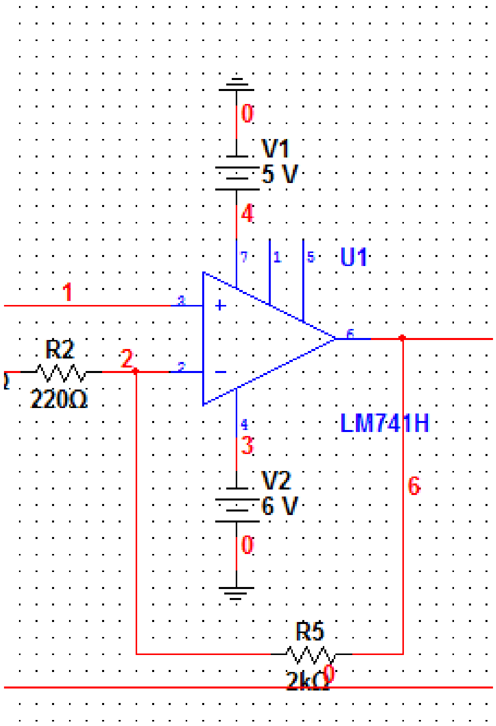


FIGURE 9 The schematic diagram of amplifier circuit.

The circuit is designed to achieve 9.091 gains. This is achieved by using 2000 Ω and 220 Ω resistors. Based on the datasheet of Freescale MPX5700DP datasheet the output range will be around 0 V to 0.5 V (the maximum of BP is estimated at 350 mmHg or 50 kPa). Hence, the signal need to amplify at 9.091 gains for widen the output voltage range. Thus, it will provide better accuracy in pressure measurement of blood pressure [**21**].

Based on simulation in Multisim 10 (see Figure 10), the circuit is working successfully as expected.

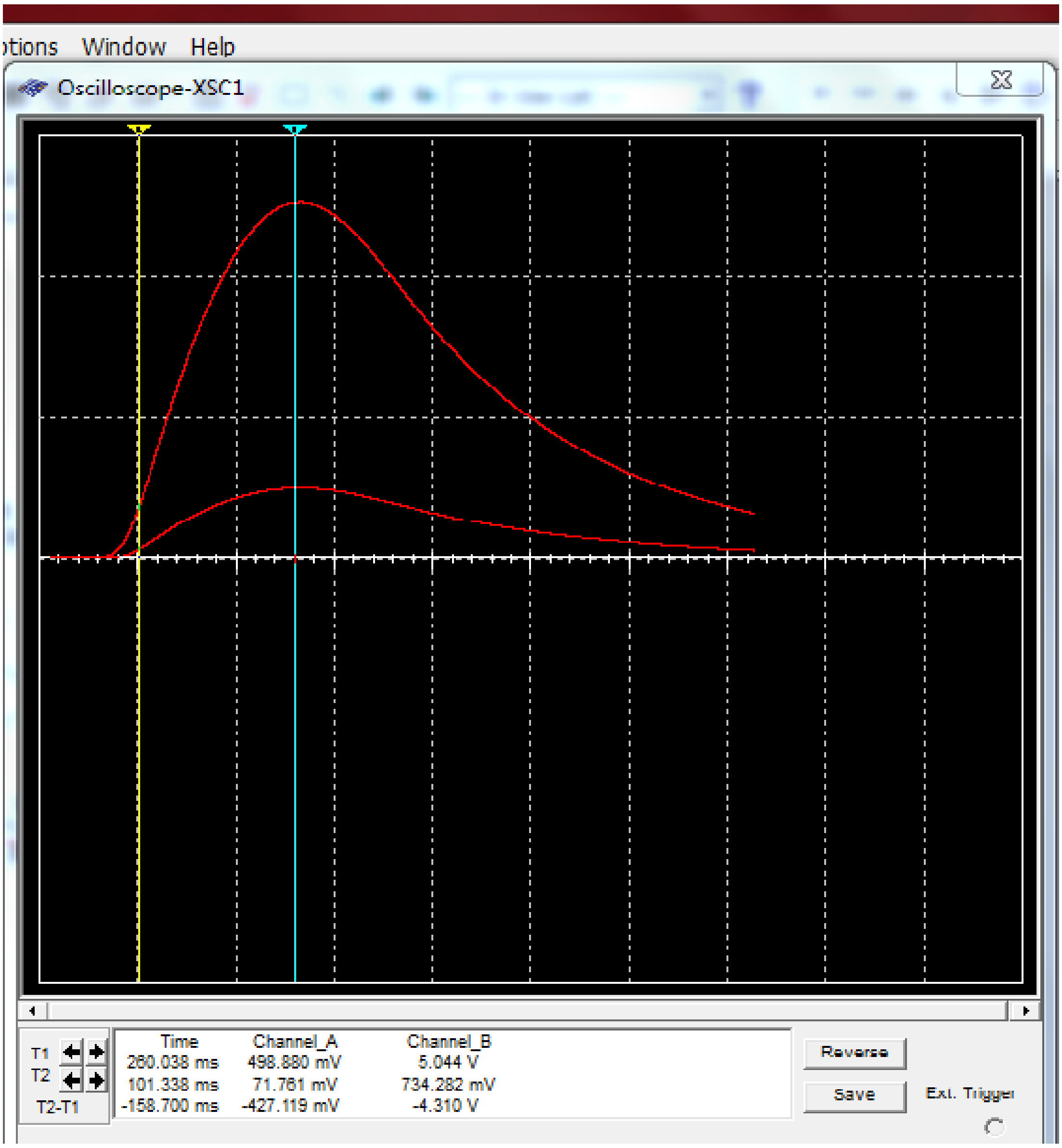


FIGURE 10 The simulation result of the amplifier circuit.

From the experiment conducted in the laboratory, the amplifier circuit able to amplify the sensor output voltage up to 10 gains. As an example, the pressure sensor is yielding 0.124 V (the input for amplifier circuit), then the amplifier circuit will yield 1.24 V.

Note that the operational amplifier has been changed from LM358 to LM741. There are several advantages LM741 against LM358. One of it is the maximum output voltage. With two supply voltages which are +5 V (from Arduino board) and -6 V (from 6 V power supply), the maximum output voltage is about 4.18 V. Compare to LM358, with single supply voltage of 5 V, the maximum output voltage is about 3.66 V.

Nevertheless, the LM358 also have some advantages over LM741. The LM358 is able to operate with single power supply. Plus, the LM358 also has lesser noise compare to LM741. Since the wide range of output voltage is the priority, LM741 is chosen for the project.

4.1.3 Low Pass Filter Circuit

Initially, the design of low pass filter circuit was designed for AC type signal as explained before. However, the MPX5700 pressure sensor only produced DC type signal. Hence, a new low pass filter circuit for DC type signal is required [**21**].

The new low pass filter circuit is a simple circuit. It is only consist of resistors and a capacitor. The value of the LP filter cut off frequency is 3.55 Hz. 3.55 Hz cut off frequency is chosen because the maximum heart beat rate for human is 200 bpm or equivalent to 3.33 Hz. Therefore, 3.55 Hz cut off frequency is sufficient to filter noise which occurred at higher frequency. The design was successful as tested in simulation as well as in laboratory experiment.

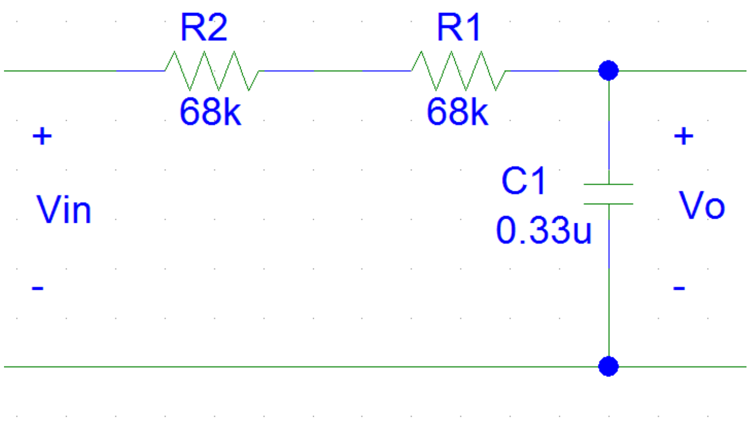


FIGURE 11 The schematic diagram of low pass filter.

4.1.4 Arduino

The Arduino is used for controlling the whole operation of the BP sensor. Many programming codes have been written and tested to ensure suitable programming codes for BP measurement.

The process starts with the motor pump inflating the cuff, then stop. After that, the cuff is deflating gradually where the systolic and diastolic pressure is determined. After the deflation process reaching certain low pressure, the air valve will release the air inside the cuff and the cuff will completely deflated.

During the slow deflation process, the Arduino starts to analyse the pressure sensor signal and stored certain important information to be analyse later. The Arduino receives the analogue signal and will determine which pressure level is reflecting the systolic and diastolic pressure.

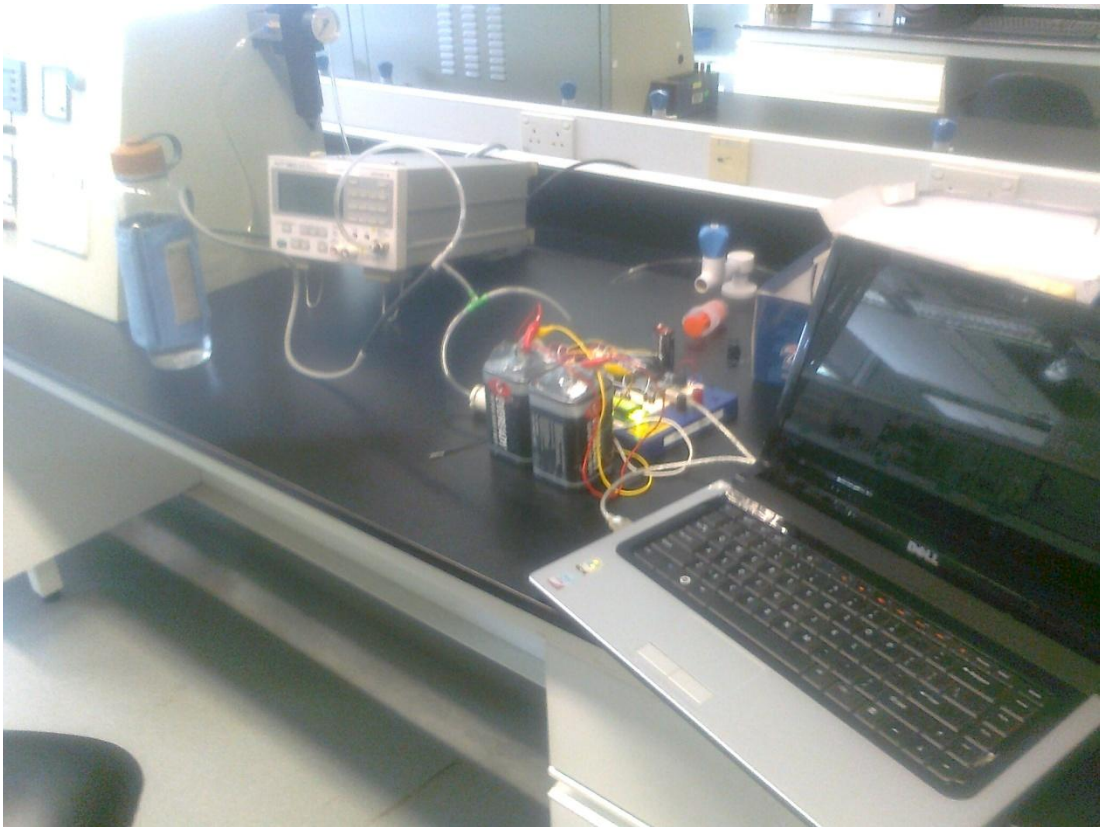


FIGURE 12 The calibration works.

Any microcontrollers should be able to handle only digital signal (not analogue signal) including Arduino. Nevertheless, the Arduino board have built-in ADC to quantize analogue signal into digital signal as explained in Chapter 3.

Some analyses have been done (by comparing with an electronic pressure gauge), to understand the Freescale pressure sensor. After the analyses completed, it is found that the pressure sensor operated by measuring the absolute pressure where the output signal also includes the atmospheric pressure. The pressure level that we need to analyse is actually the pressure inside the cuff. Therefore, to determine the pressure inside the cuff, the atmospheric pressure needs to be subtracted.

Then, the digital signal needs calibration to represent the actual pressure readings. For such purpose, the digital signal is compared with pressure gauge. In this calibration process, a bottle is used to represent human arm. The motor pump is pumped, and then the digital pressure gauge and the digital output of Arduino appeared accordingly. The data tabulated to be analysed. This process is conducted multiple times and the data is analysed to determine right equation (for programming codes). At last, the suitable mathematical equation determined and used for programming codes.

Other than that, the digital signal is sampled at 40 ms (the pressure sensor output voltage is captured every 40 ms). The reason sampling at 40 ms is to capture analogue signal accurately and smoothening the signal by using 3-point moving average technique.

To further clarification about this decision, let we recall the cut off frequency chosen for LP filter part which is 3.55 Hz. 3.55 Hz is equivalent to single peak-to-peak signal at every 282 ms. For 3-point moving average, it used 3 points, and averaged them into one point. Thus 282 ms need to be divided by 3, and yielded 94 ms. The sampling process is needed to be done two times faster than the analogue signal, to avoid lost shape of the analogue's envelope. Hence, the sampling time required is at least at every 47 ms. Therefore, the sampling time at every 40ms is more than enough to capture the analogue signal [**22**].

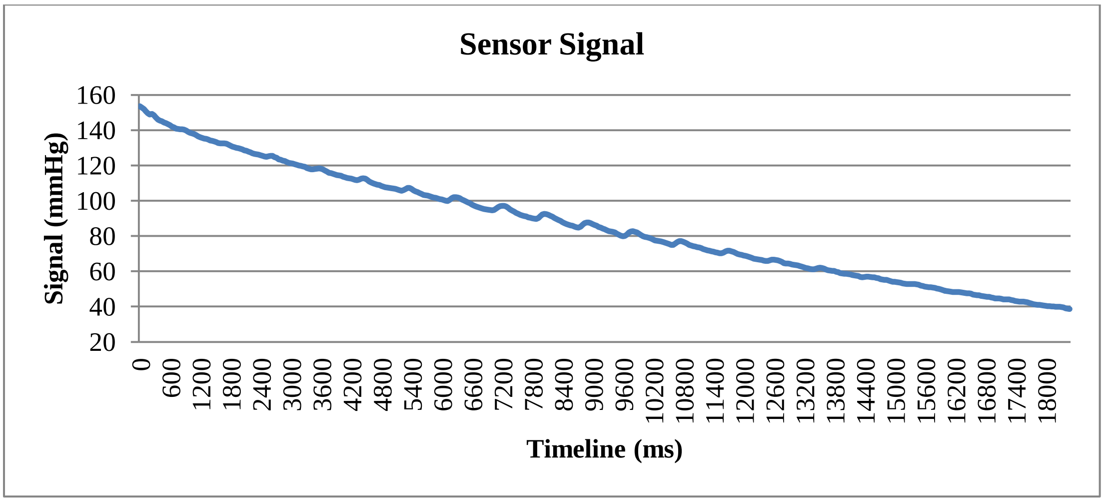


FIGURE 13 The timeline of the pressure sensor signal.

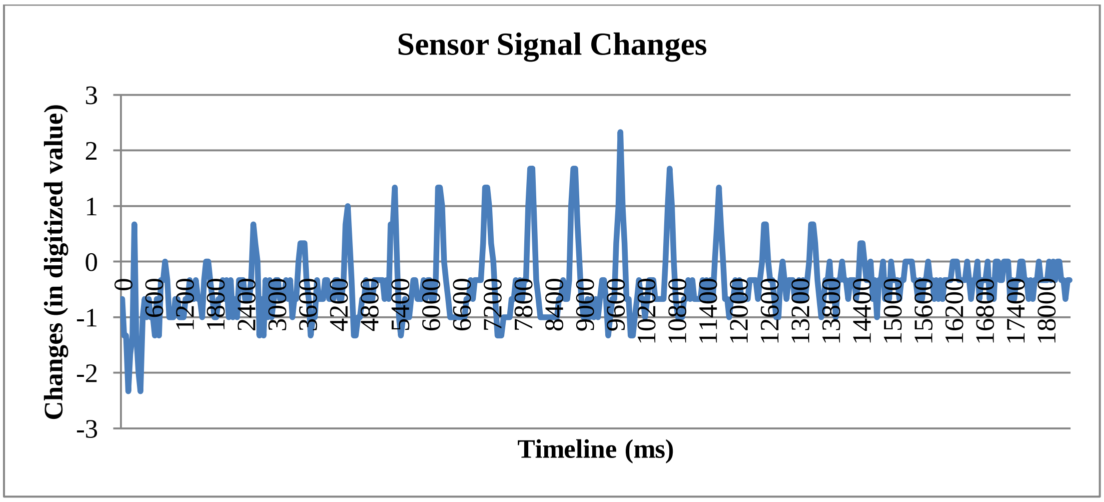


FIGURE 14 The timeline of the signal changes.

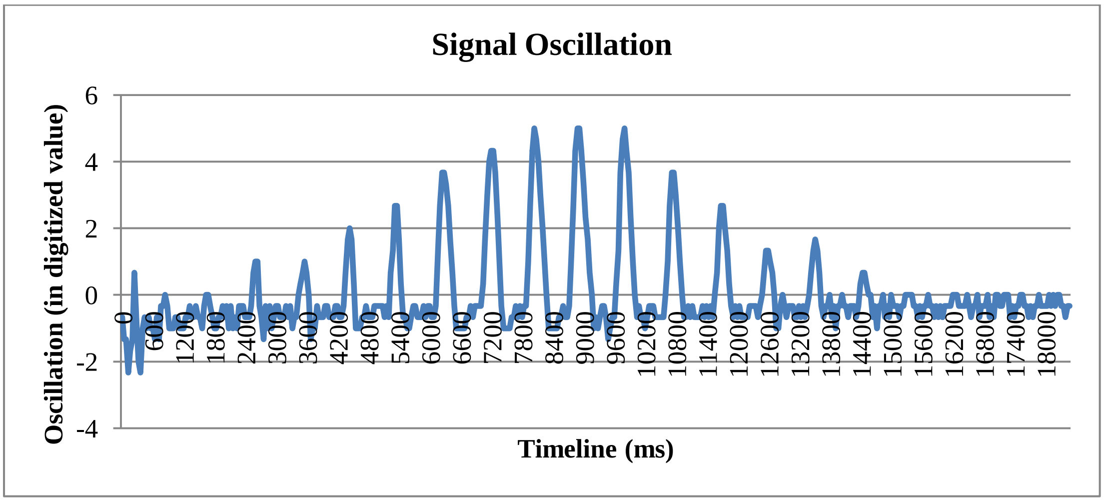


FIGURE 15 This graph shows the timeline of output oscillation. This is formed based on the output changes.

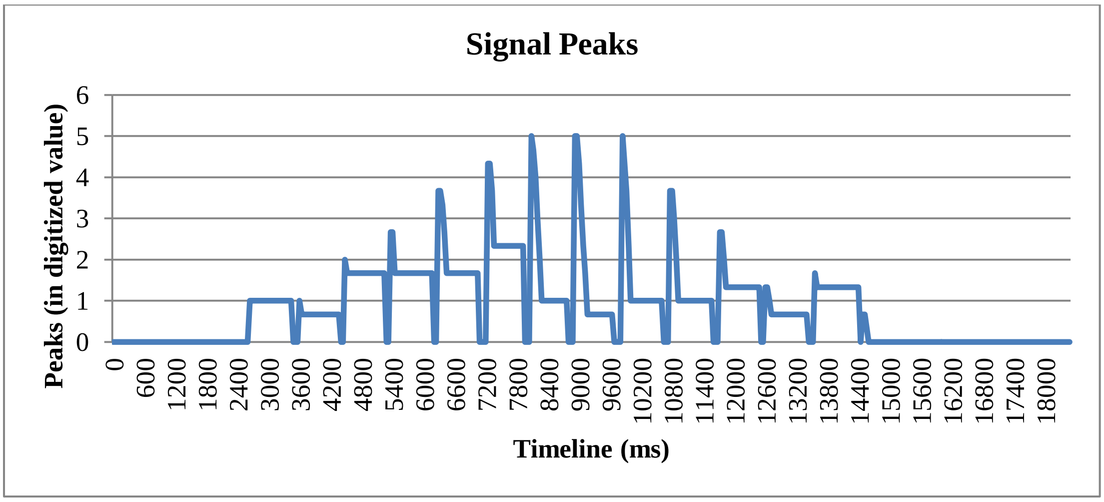


FIGURE 16 The timeline of detected peak from the oscillations.

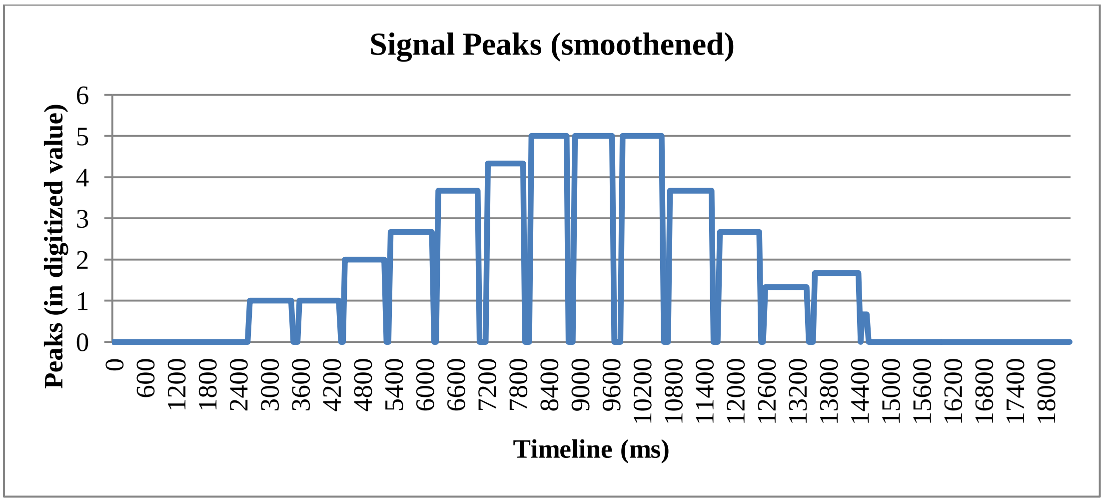


FIGURE 17 The timeline of smoothened detected peak.

To implement appropriate programming code, the data acquisition is crucial. Later, by studying the data gathered, an appropriate programming code can be written for determining BP level. By using Arduino, a programming code is written to tabulate the signal data for every 40 ms, and this can be more understandable in a graph shown in Figure 13. Lastly, the suitable programming code successfully implemented. Figure 13 to 17 show the processes that done by the programming code.

Immediately after the peaks are detected, the controller will store the peaks value and the pressure value accordingly. After the deflation is complete, the controller will analyse the stored peaks and determine the systolic and diastolic pressure. Because of the controller is depending the voltage signal for BP measurement, the controller might also captured the noise. Figure 18 and Figure 19 show the example of peaks value (in digitize value) and the pressure value (in mmHg) respectively.

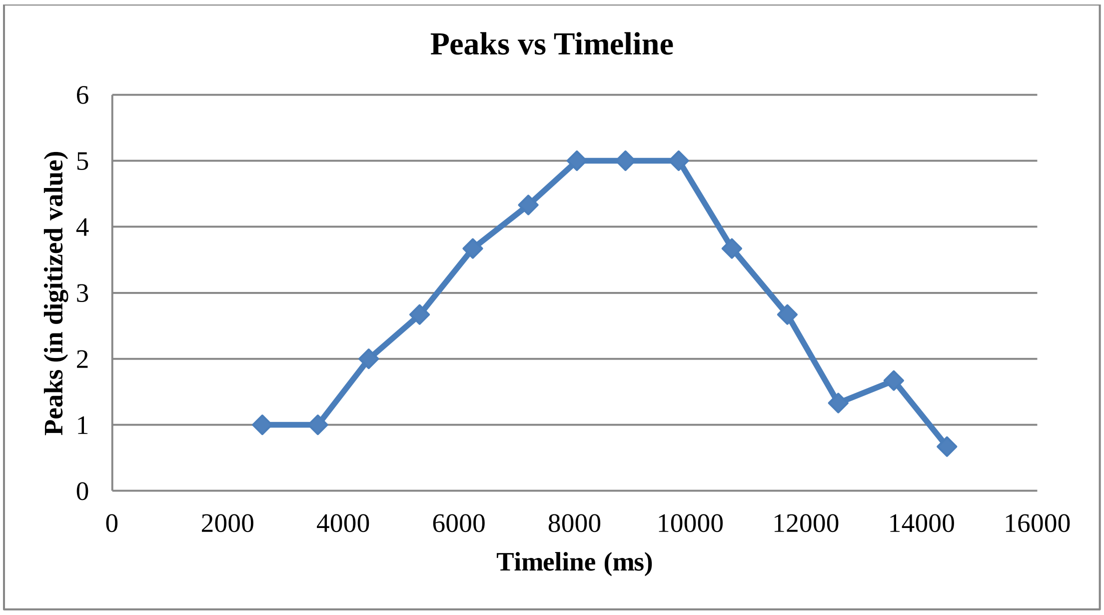


FIGURE 18 The stored peaks value against time (ms).

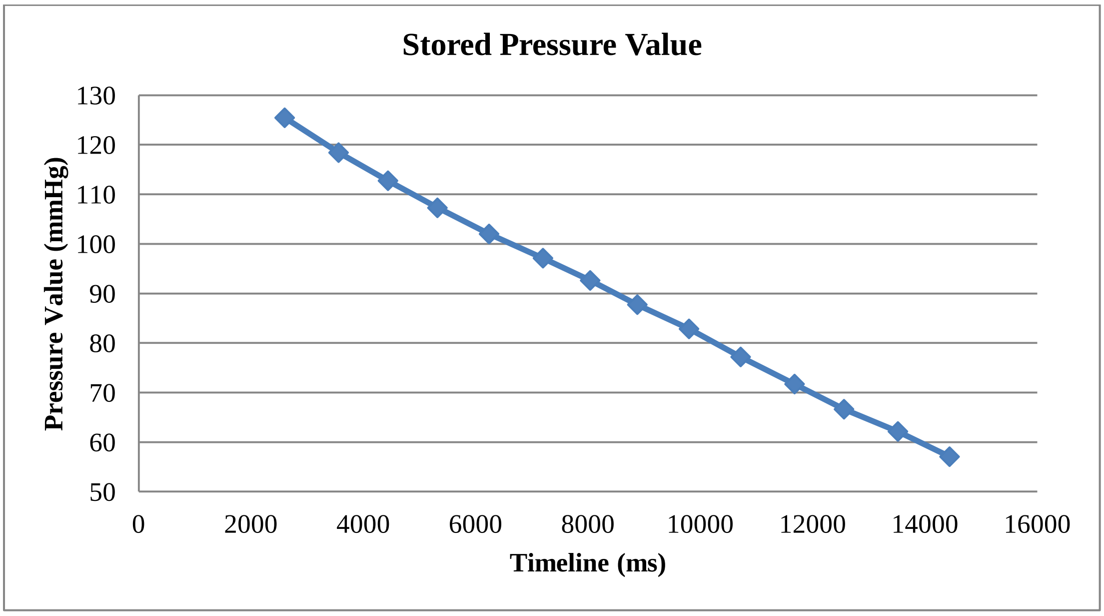


FIGURE 19 The stored pressure value against time (ms).

After that, several programming codes have been developed to determine BP level. Then, the programming codes has been tested and compared with Omron BP monitor. This is done by comparing the readings while the BP sensor and the Omron

BP monitor are run simultaneously. The BP sensor determining the BP level on left arm while the Omron BP monitor determining BP level on right arm.

As explained before, the systolic and diastolic pressure is determined when there are significant peaks. At early stage, based on the data acquired from several experiments conducted, it seems that the significant peaks are the peaks that exceed half of the highest peak. Let the peaks data in Figure 18 as an example. In this peaks data the highest peak is at 5. Therefore, any peaks that are exceeding 2.5 will be considered as significant peaks. The first significant peak, will determine the systolic pressure, and the last significant peak will determine the diastolic pressure. However, the results show that this assumption is not accurate and not consistent. Hence, the programming codes need to be rectified in order to determine BP level.

Other than the programming codes itself as the factor of the inaccuracy and inconsistent, this is also due to some problems that discovered after several tests such as the inflatable cuff noises and the circuit noise. This matter will be discussed further in the next subsection.

Several programming codes have been developed throughout the project development. These codes were named as Mark 1 to Mark 19. From Mark 10 to Mark 19 programming codes, the Andon cuff is used to replace the old cuff due to noise problem. Between Mark 10 to Mark 18 programming codes, Mark 18 gives the best result.

Mark 18 programming codes was developed through a series of experiments to determine the significant peaks. From the data, a ratio between suspected significant peaks and the highest peak was determined. At last, the results become better (for systolic pressure).

Nevertheless, by using Mark 18 programming codes, the determined diastolic pressure is always not accurate. This result is shown in the Table 4 and Table 5 (for systolic and diastolic pressure respectively). Plus, the result of this method (the programming codes) is not consistent when it was tested with difference subjects (persons). Therefore, new methods (the programming codes) need to be identified. It may use the mathematical technique like neural network.

TABLE 4 The comparison of systolic reading between designed BP sensor and Omron BP monitor.

Systolic M18

	BP Sensor (mmHg)	Omron (mmHg)	**Percentage Error (%)**	
1st	134.22	131	**2.458015**	
2nd	129.93	131	**0.816794**	
3rd	131.49	132	**0.386364**	
4th	129.93	132	**1.568182**	
5th	132.47	128	**3.492188**	
	Standard Deviation (mmHg)	Standard Deviation (mmHg)	**Average (%)**	**Standard Deviation (%)**
	1.817339	1.643168	**1.744308**	**1.254968**

Since, the programming codes is much depending on the amplified and filtered signal where noise also could intercept the signal, the controller sometimes captured the noise other than the oscillation of the pressurised artery.

TABLE 5 The comparison of diastolic reading between designed BP sensor and Omron BP monitor.

Diastolic M18

	BP Sensor (mmHg)	Omron (mmHg)	**Percentage Error (%)**	
1st	80.5	73	**10.27397**	
2nd	90.85	83	**9.457831**	
3rd	93.39	81	**15.2963**	
4th	93.98	83	**13.22892**	
5th	85.77	76	**12.85526**	
	Standard Deviation (mmHg)	Standard Deviation (mmHg)	**Average (%)**	**Standard Deviation (%)**
	5.703637	4.494441	**12.22246**	**2.36123**

4.1.5 The noise of the Inflatable Cuff

There are also some experiments done on the cuff. Currently, three cuffs can be used for the project. One of them is the old inflatable cuff which is taken from broken BP monitor. Another cuff is taken Andon BP monitor, and the other is the inflatable cuff from the Omron BP monitor.

The result can be seen in Figure 20 to Figure 22. In these graphs, it is about time interval between pulses in ms, against n^{th} pulses. In this result, it shows that the Omron and Andon cuffs have lesser noise compare to old cuff. Supposedly, the pulse is detected in between 760 ms to 800 ms (about 70 bpm) constantly. On the contrary, the old cuff sometimes is creating pulse lesser time interval.

From the experiments, it is found that the old inflatable cuff has more noise compare to Andon and Omron inflatable cuff. Because of the programme codes are capturing the oscillations signal, the noise makes the controller capturing ‘fake’ oscillations. Plus, the old inflatable cuff tends to yield smaller magnitude of the oscillation of the pressurised artery than the Omron inflatable cuff. Hence, the inflatable cuff also affecting the BP measurement of this circuit.

From the results, the Omron Cuff is the best inflatable cuff because it has lesser noise and more sensitive (provides higher magnitude of oscillation to the controller). However, because of the Omron cuff need to be used for Omron BP monitor, the Andon cuff is selected to replace the old cuff.

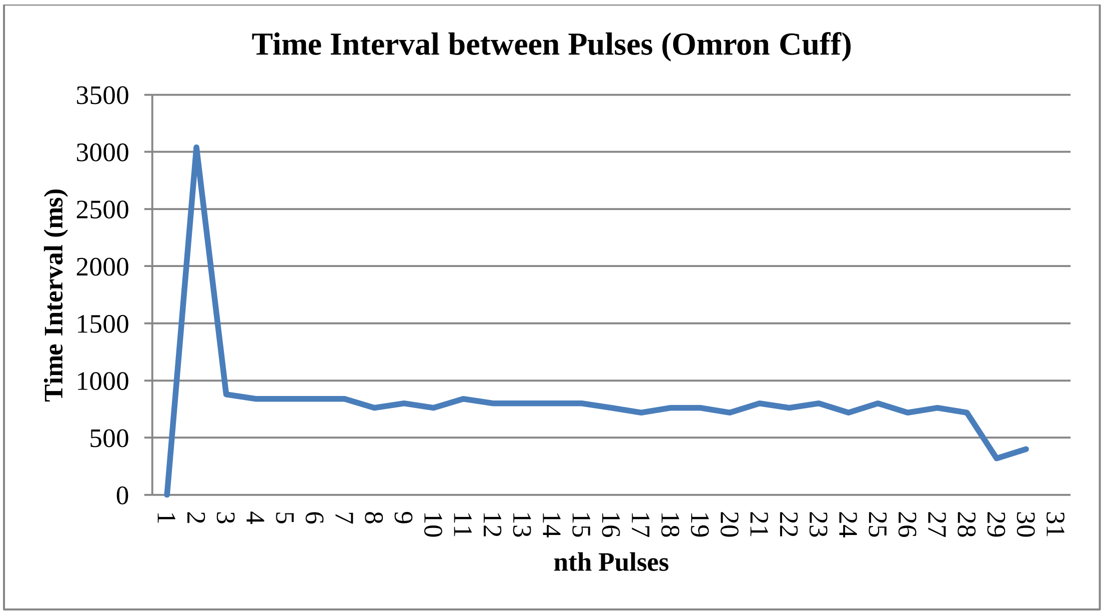


FIGURE 20 Time interval (ms) between two pulses in Omron Cuff.

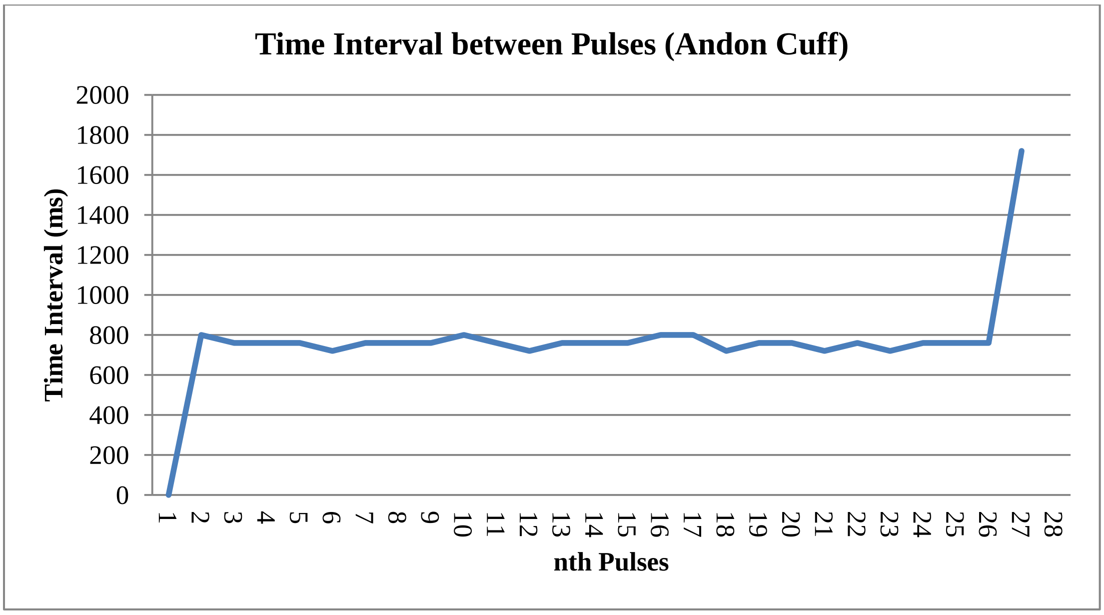


FIGURE 21 Time interval (ms) between two pulses in Andon Cuff.

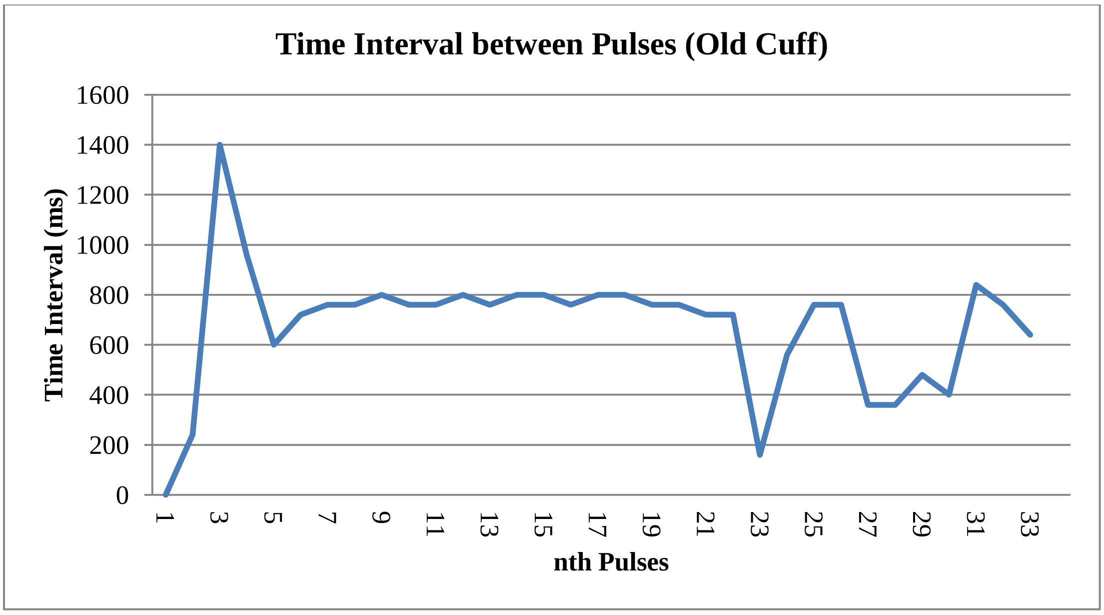


FIGURE 22 Time interval (ms) between two pulses in Old Cuff.

4.1.6 Wireless part

Even though the BP sensor is not completed yet, the wireless part of the project has been made. It is based on Zigbee technology. A simple code has been tested and the wireless part is working as expected. It just simply attaches the Zigbee module on the Arduino board, and a simple code is made. This provides assurance that the wireless part can be implemented. Unfortunately, because of the accuracy of the BP sensor part is not good, the development of wireless part is put on hold.

4.2 Discussion

Based on the result, the accuracy of designed BP sensor is still the main obstacle of the project. The accuracy of direct BP measurement is not sufficient, this also will gives similar result when integrates with wireless part. Therefore, this is the main reason why the wireless part is not integrated yet with the project.

The amplifier circuit is required more attention before improving the other parts of the design. This is due to the noise of the LM741 operational amplifier. Low noise operational amplifier is required for this project. One of the suggested operational amplifiers is the INA333 by Texas Instrument.

Other than that, the BP sensor should be using components rated at 5V and below. This to ensure, the overall design of the project is used single power supply. Thus, it will provide simpler designed of the system.

Last but not least, the integration between BP sensor and wireless technology should be started as the accuracy of the BP sensor is successfully achieved.

CHAPTER 5
CONCLUSION AND RECOMMENDATION

5 CONCLUSION AND RECOMMENDATION

As conclusion, the project is not complete as the proposed design (the concept design flow process, see Figure 3). This is due to accuracy issue of BP measurement of the BP sensor. The project can be divided into two major parts, which are the BP sensor and network topology for remote monitoring. A simple test on wireless part gives confidence that the Design and Implementation of Blood Pressure Monitoring System project is feasible to be done (if the accuracy issue of the BP sensor is successfully rectified).

As recommendation for this project, the improvement of the circuit particularly the amplifier part needs to be done. The selection of suitable inflatable cuff also needs to be done. Other than that, the programming codes also need to be improved. Other than that, the programming codes should use mathematical relations like neural network to determine the BP level. Besides, the resolution of ADC also needs to increase, so that the controller can receive higher resolution of the analogue signal. Then, the BP sensor can be integrated with wireless part and forming network topology for the whole system between multiple BP sensors and a computer.

REFERENCES

[1] Santiago Lopez, Blood Pressure Monitor Fundamentals and Design, 2012, Application Note by Freescale Semiconductor.

[2] Jasa Abdul Rahman, "Design of Portable Blood Pressure Monitor," Universiti Teknologi PETRONAS, Bandar Seri Iskandar, Bachelor Thesis 2007.

[3] Richard N. Fogoros. (2003, December) About.com - Heart Health Center. [Online]. http://heartdisease.about.com/cs/starthere/a/chambersvalves.htm

[4] Jeremy Booth, "A Short History of Blood Pressure Measurement," *Proceedings of the Royal Society of Medicine*, vol. 70, pp. 793-799, November 1977.

[5] Aminurrashid Noordin, "Cuff - less Blood Pressure Meter," Universiti Teknologi Malaysia, Skudai, Master Thesis 2009.

[6] InternationalDrugMart.com. (2013) Blog.internationaldrugmart.com. [Online]. http://blog.internationaldrugmart.com/

[7] S Daochai, W Sroykham, Y Kajornpredanon, and C Apaiwongse, "Non-invasive Blood Pressure Measurement: Auscultatory method versus Oscillometric method," in *The 2011 Biomedical Engineering International Conference (BMEiCON-2011)*, Chiang Mai, 2011, pp. 221 - 224.

[8] Nor Azura Mohamed Yusof, "Portable Digital Blood Pressure Monitor Blood Pressure Coding System," Universiti Teknikal Malaysia Melaka, Hang Tuah Jaya, Bachelor Thesis 2007.

[9] Dr. P. Laurent. (2011, December) Blood Pressure & Hypertension. [Online]. http://www.blood-pressure-hypertension.com/how-to-measure/measure-blood-pressure-8.shtml

[10] Steven Shepard, *RFID: Radio Frequency Identification.*: McGraw-Hill Companies, Inc., 2005.

[11] Meng Qingbin and Jin Jie, "Design of Low - Power Active RFID Tag in UHF Band," *Control, Automation and Systems Engineering (CASE), 2011 International Conference*, pp. 1-4, July 2011.

[12] Arduino. (2013, July) Arduino. [Online]. http://www.arduino.cc/

[13] Bonifacio Castano and Maria Rodriguez-Moreno, "A Zigbee and RFID Hybrid System for People Monitoring and Helping inside Large Buildings," *Industrial Electronics & Applications (ISIEA), 2010 IEEE Symposium*, pp. 16-21, October 2010.

[14] Drew Gislason. (2010, February) EE Times Design. [Online]. http://www.eetimes.com/design/embedded-internet-design/4201087/ZigBee-applications--Part-1-Sending-and-receiving-data/

[15] C. Douniama, C. U. Sauter, and R. Couronne, "Blood Pressure Tracking Capabilities of Pulse Transit Times in Different Arterial Segments: A Clinical Evaluation," *Computers in Cardiology*, vol. 36, September 13-16 2009.

[16] Ali Hassan M. K, Mashor M. Y, Mohd Saad A. R, and Mohamed M. S, "A Portable Continuous Blood Pressure Monitoring Kit," in *IEEE Symposium on Business, Engineering and Industrial Applications (ISBEIA)*, Langakawi, Malaysia, 2011, pp. 503-507.

[17] Heiko Gesche, Detlef Grosskurth, Gert Ku¨chler, and Andreas Patzak, "Continuous Blood Pressure Measurement by Using the Pulse Transit Time: Comparison to a Cuff-based Method," April 2011.

[18] Md Manirul Islam et al., "Development of a Noninvasive Continuous Blood Pressure Measurement and Monitoring System," in *International Conference on Informatics, Electronics & Vision*, Dhaka, Bangladesh, 2012.

[19] Jin Li Wun, Long Luo Yuan, Shun Chang Yao, and Hsiang Lin Yuan, "A Wireless Blood Pressure Monitoring System for Personal Health Management,"

in *32nd Annual International Conference of the IEEE EMBS*, Buenos Aires, Argentina, 2010, pp. 2196-2199.

[20] Dr. Neil Townsend, "Medical Electronics," *Michealmas Term 2001*, pp. 48-54, 2001.

[21] Freescale Semiconductor, Inc., Integrated Silicon Pressure Sensor On-Chip Signal Conditioned, Temperature Compensated and Calibrated, 2007, MPX5700 Series Technical Data.

[22] William Stallings, *Data and Computer Communications*, 9th ed., Micheal Hirsch, Tracy Dunkelberger, and Melinda Haggerty, Eds. New Jersey: Prentice Hall, 2011.

APPENDICES

APPENDIX A

Freescale Semiconductor MPX5700 Series Pressure Sensor Datasheet

Integrated Silicon Pressure Sensor On-Chip Signal Conditioned, Temperature Compensated and Calibrated

The MPX5700 series piezoresistive transducer is a state-of-the-art monolithic silicon pressure sensor designed for a wide range of applications, but particularly those employing a microcontroller or microprocessor with A/D inputs. This patented, single element transducer combines advanced micromachining techniques, thin-film metallization, and bipolar processing to provide an accurate, high level analog output signal that is proportional to the applied pressure.

Features

- 2.5% Maximum Error over 0° to 85°C
- Ideally Suited for Microprocessor or Microcontroller-Based Systems
- Available in Absolute, Differential and Gauge Configurations
- Patented Silicon Shear Stress Strain Gauge
- Durable Epoxy Unibody Element

ORDERING INFORMATION

Device Type	Options	Case Type	MPX Series	
			Order Number	Device Marking
Basic Element	Differential	867	MPX5700D	MPX5700D
	Absolute	867	MPX5700A	MPX5700A
Ported Elements	Differential Dual Ports	867C	MPX5700DP	MPX5700DP
	Gauge	867B	MPX5700GP	MPX5700GP
	Gauge, Axial	867E	MPX5700GS	MPX5700D
	Absolute	867B	MPX5700AP	MPX5700AP
	Absolute, Axial	867E	MPX5700AS	MPX5700A
	Absolute, Axial PC Mount	867F	MPX5700ASX	MPX5700A

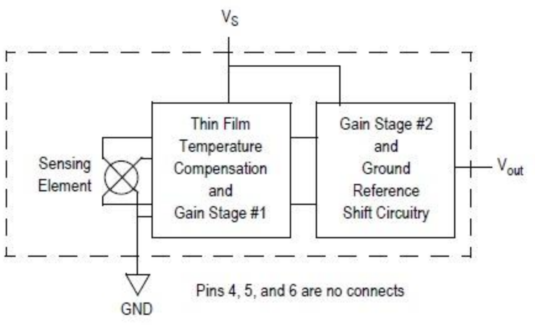
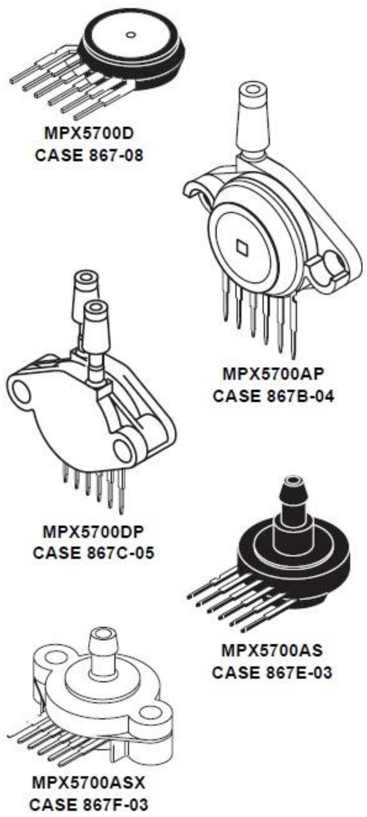


Figure 1. Fully Integrated Pressure Sensor Schematic

MPX5700 SERIES

INTEGRATED PRESSURE SENSOR
0 to 700 kPa (0 to 101.5 psi)
15 to 700 kPa (2.18 to 101.5 psi)
0.2 to 4.7 V OUTPUT



PIN NUMBERS

1	V_{out}	4	N/C
2	GND	5	N/C
3	V_S	6	N/C

NOTE: Pins 4, 5, and 6 are internal device connections. Do not connect to external circuitry or ground. Pin 1 is noted by the notch in the lead.

Table 1. Maximum Ratings(1)

Parametrics	Symbol	Value	Unit
Maximum Pressure(2) (P2 ≤ 1 Atmosphere)	$P1_{max}$	2800	kPa
Storage Temperature	T_{stg}	–40 to +125	°C
Operating Temperature	T_A	–40 to +125	°C

1. Maximum Ratings apply to Case 867 only. Extended exposure at the specified limits may cause permanent damage or degradation to the device.
2. This sensor is designed for applications where P1 is always greater than, or equal to P2. P2 maximum is 500 kPa.

Table 2. Operating Characteristics (V_S = 5.0 Vdc, T_A = 25°C unless otherwise noted, P1 > P2. Decoupling circuit shown in Figure 4 required to meet electrical specifications.)

Characteristic		Symbol	Min	Typ	Max	Unit
Pressure Range(1)	Gauge, Differential: MPX5700D Absolute: MPX5700A	P_{OP}	0 15	— —	700 700	kPa
Supply Voltage(2)		V_S	4.75	5.0	5.25	Vdc
Supply Current		I_O	—	7.0	10	mAdc
Zero Pressure Offset(3)	Gauge, Differential (0 to 85°C) Absolute (0 to 85°C)	V_{off}	0.088 0.184	0.2 —	0.313 0.409	Vdc
Full Scale Output(4)	(0 to 85°C)	V_{FSO}	4.587	4.7	4.813	Vdc
Full Scale Span(5)	(0 to 85°C)	V_{FSS}	—	4.5	—	Vdc
Accuracy(6)	(0 to 85°C)	—	—	—	±2.5	$\%V_{FSS}$
Sensitivity		V/P	—	6.4	—	mV/kPa
Response Time(7)		t_R	—	1.0	—	ms
Output Source Current at Full Scale Output		I_{O+}	—	0.1	—	mAdc
Warm-Up Time(8)		—	—	20	—	ms

1. 1.0 kPa (kiloPascal) equals 0.145 psi.
2. Device is ratiometric within this specified excitation range.
3. Offset (V_{off}) is defined as the output voltage at the minimum rated pressure.
4. Full Scale Output (V_{FSO}) is defined as the output voltage at the maximum or full rated pressure.
5. Full Scale Span (V_{FSS}) is defined as the algebraic difference between the output voltage at full rated pressure and the output voltage at the minimum rated pressure.
6. Accuracy (error budget) consists of the following:
   - Linearity: Output deviation from a straight line relationship with pressure over the specified pressure range.
   - Temperature Hysteresis: Output deviation at any temperature within the operating temperature range, after the temperature is cycled to and from the minimum or maximum operating temperature points, with zero differential pressure applied.
   - Pressure Hysteresis: Output deviation at any pressure within the specified range, when this pressure is cycled to and from the minimum or maximum rated pressure, at 25°C.
   - TcSpan: Output deviation over the temperature range of 0° to 85°C, relative to 25°C.
   - TcOffset: Output deviation with minimum rated pressure applied, over the temperature range of 0° to 85°C, relative to 25°C.
   - Variation from Nominal: The variation from nominal values, for Offset or Full Scale Span, as a percent of V_{FSS}, at 25°C.
7. Response Time is defined as the time for the incremental change in the output to go from 10% to 90% of its final value when subjected to a specified step change in pressure.
8. Warm-up Time is defined as the time required for the device to meet the specified output voltage after the pressure has been stabilized.

Table 3. Mechanical Characteristics

Characteristics	Typ	Unit
Weight, Basic Element (Case 867)	4.0	grams

ON-CHIP TEMPERATURE COMPENSATION, CALIBRATION AND SIGNAL CONDITIONING

Figure 3 illustrates both the Differential/Gauge and the Absolute Sensing Chip in the basic chip carrier (Case 867). A fluorosilicone gel isolates the die surface and wire bonds from the environment, while allowing the pressure signal to be transmitted to the sensor diaphragm. (For use of the MPX5700D in a high-pressure cyclic application, consult the factory.)

The MPX5700 series pressure sensor operating characteristics, and internal reliability and qualification tests are based on use of dry air as the pressure media. Media, other than dry air, may have adverse effects on sensor performance and long-term reliability. Contact the factory for information regarding media compatibility in your application.

Figure 2 shows the sensor output signal relative to pressure input. Typical, minimum, and maximum output curves are shown for operation over a temperature range of 0° to 85°C using the decoupling circuit shown in Figure 4. The output will saturate outside of the specified pressure range.

Figure 4 shows the recommended decoupling circuit for interfacing the output of the integrated sensor to the A/D input of a microprocessor or microcontroller. Proper decoupling of the power supply is recommended.

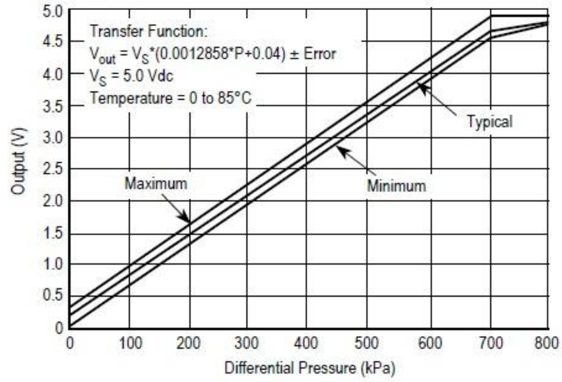


Figure 2. Output versus Pressure Differential

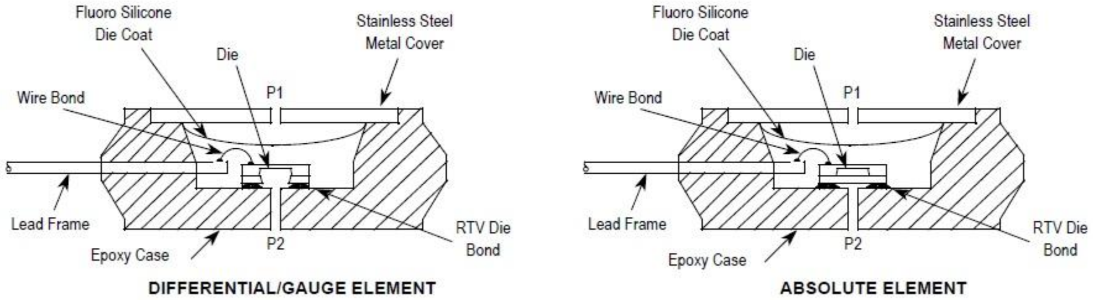


Figure 3. Cross-Sectional Diagrams (not to scale)

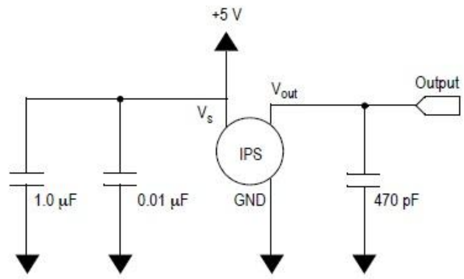


Figure 4. Recommended Power Supply Decoupling and Output Filtering (For additional output filtering, please refer to Application Note AN1646)

APPENDIX B

Blood Pressure Sensor Arduino Code Program (Mark 18)

Blood Pressure Sensor Arduino Code Program (Mark 18)

```
// Mark_18 - based on Mark_11 for Andon Cuff

int pressureSensor = 0;
int airValve = 6;
int airPump = 7;
int button = 8;
int DIP1 = 9;  //for select maximum inflation
int DIP2 = 10;  //for select maximum inflation
int prog;
int state;
int DIP_1;
int DIP_2;
int maxS;  //set maximum inflation
int i;  //for array
int cnt;  //for counting beats for diastolic
float initial;
float x1;
float x2;
float x3;
float xS31;  //3 points moving average
float xS32;  //3 points moving average
float xC;  //changes between past and current reading
float xR1;  //rise value
float xR2;  //rise value
float xP;  //peak value
float xPR1;  //peak_real, previous value
float xPR2;  //peak_real, current value
float SPeak;  //significant peak value
float Systo;
float Diasto;
unsigned long d;
unsigned long dx;

void setup()  {
  pinMode(button, INPUT);
  pinMode(airValve, OUTPUT);
  pinMode(airPump, OUTPUT);
  prog = 1;
  Serial.begin(9600);
  Serial.print("Annyeong haseyo!");
  Serial.println("");
}

void loop()  {
  switch(prog)  {
    case 1:  //initial
    state = digitalRead(button);
    if (state == HIGH)  {
      Serial.print("----------------");
      delay(2000);
      Serial.println("");
      Serial.print("-----START!-----");
      delay(2000);
      Serial.println("");
      prog = prog + 1;
    }
    break;

    case 2:  //exceed Systolic
```

```
    x1 = analogRead(pressureSensor);
    initial = x1;
    // select maximum inflation
    DIP_1 = digitalRead(DIP1);
    DIP_2 = digitalRead(DIP2);
    if ((DIP_1 == LOW) && (DIP_2 == LOW))  {
      maxS = 650;  //about 150mmHg
    }
    if ((DIP_1 == LOW) && (DIP_2 == HIGH))  {
      maxS = 700;  //about 180mmHg
    }
    if ((DIP_1 == HIGH) && (DIP_2 == LOW))  {
      maxS = 750;  //about 210mmhg
    }
    if ((DIP_1 == HIGH) && (DIP_2 == HIGH))  {
      maxS = 800;  //about 240mmHg
    }
    while(x1 < maxS)  {
      digitalWrite(airValve, HIGH);
      digitalWrite(airPump, HIGH);
      x1 = analogRead(pressureSensor);
    }
    digitalWrite(airPump, LOW);
    prog = prog + 1;
    break;

    case 3:  //Check for stable value
    Serial.print("Stablizing...");
    Serial.println("");
    delay(40);  //avoid early fluctuation
    xS31 = 0;
    x1 = 0;
    x2 = analogRead(pressureSensor);
    delay(40);  //sampling at 25ms
    while(xS31 < (maxS - 50))  {  //the smoothing value must exceed
(maxS - 50)
      x3 = analogRead(pressureSensor);
      dx = millis();
      xS31 = (x1 + x2 + x3)/3;
      x1 = x2;
      x2 = x3;
      Serial.print(xS31,2);
      Serial.println("");
      d = 40 - (millis() - dx);  //sampling at 40ms
      delay(d);
    }
    prog = prog + 1;
    break;

    case 4:  //Detecting Peak, store data for analysis
    float Sys[100];
    float sxP[100];
    float xHP[100];
    i = 0;
    Serial.print("Peak Detection");
    Serial.println("");
    Serial.print("x");
    Serial.print("\t");
    Serial.print("sxP");
    Serial.print("\t");
    Serial.print("xHP");
```

```
    Serial.println("");
    xPR1 = 0;  //Initial value of peak_real, 0 means no peak
    xHP[0] = 0;  //Initial value of highest_peak
    Sys[0] = 0;
    sxP[0] = 0;
    i = i + 1;  //ready for next array
    x3 = analogRead(pressureSensor);
    dx = millis();
    xS32 = (x1 + x2 + x3)/3;
    xR1 = xS31 - xS32;
    x1 = x2;
    x2 = x3;
    xS31 = xS32;
    d = 40 - (millis() - dx);  //sampling at 25ms
    delay(d);
    while(xS32 > 450)  {  //Once the Systolic detected, go to
"Diastolic"
      x3 = analogRead(pressureSensor);
      dx = millis();
      xS32 = (x1 + x2 + x3)/3;  //smoothing
      x1 = x2;
      x2 = x3;
      xC = xS32 - xS31;  //the change/difference betweem current and
previous
      if(xR1 < 0)  {
        xR2 = xC;  //forming oscillation
      }
      else  {
        xR2 = xR1 + xC;  //forming oscillation
      }
      if ((xR1 >= 0) && (xR2 >= 0))  {
        if (xR1 >= xR2)  {
          xP = xR1;  //detect peak value
        }
        else  {
          xP = 0;  //no peak detected
        }
      }
      if ((xP <= 0) && (xPR1 <= 0))  {
        xPR2 = 0;  //no peak_real
      }
      if ((xP <= 0) && (xPR1 > 0))  {
        xPR2 = 0;  //no peak_real
      }
      if ((xP > 0) && (xPR1 <= 0))  {
        xPR2 = xP;  //peak_real detected
      }
      if ((xP > 0) && (xPR1 > 0))  {
        xPR2 = xPR1;  //peak_real detected
      }
      //suspecting Sys, store Sys and peak
      if (xPR2 > xPR1)  {
        Sys[i] = xS32;
        Serial.print(Sys[i],2);
        Serial.print("\t");
        sxP[i] = xPR2;
        Serial.print(sxP[i],2);
        Serial.print("\t");
        // determine highest peak
        if (sxP[i] > xHP[(i-1)])  {
          xHP[i] = sxP[i];
```

```
    }
    else {
      xHP[i] = xHP[(i-1)];
    }
    Serial.print(xHP[i],2);
    Serial.println("");
    xHP[(i-1)] = xHP[i];
    i = i + 1;
  }
  xS31 = xS32;
  xR1 = xR2;
  xPR1 = xPR2;
  d = 40 - (millis() - dx);  //samping at 40ms
  delay(d);
}
digitalWrite(airValve, LOW);
SPeak = xHP[(i-1)] * 0.318707;  //determine significant peak
Serial.print(Sys[(i-1)],2);
Serial.println("");
Serial.print(initial,2);
Serial.println("");
dx = millis();
prog = prog + 1;
break;

case 5:  //for systolic and diastolic detection
Systo = 0;
i = 0;
Serial.print("SPeak");
Serial.print("\t");
Serial.print(SPeak,2);
Serial.println("");
while (Systo < 5)  {  //systolic
  if (sxP[i] > SPeak)  {
    Systo = (Sys[i] - initial) * 0.586135;
    Serial.print("Systolic pressure (mmHg)");
    Serial.print("\t");
    Serial.print(Systo, 2);
    Serial.println("");
  }
  i = i + 1;
}
Diasto = 0;
cnt = 0;
while (Diasto < 5)  {  //diastolic
  if ((sxP[i] < SPeak) && (xHP[i] > SPeak))  {
    cnt = cnt + 1;
  }
  else  {
    cnt = 0;
  }
  if (cnt > 3)  {
    Diasto = (Sys[(i-4)] - initial) * 0.586135;
    Serial.print("Diastolic pressure (mmHg)");
    Serial.print("\t");
    Serial.print(Diasto, 2);
    Serial.println("");
  }
  i = i + 1;
}
delay(3000);
```

```
    Serial.print("Done...");
    Serial.println("");
    Serial.println("");
    prog = prog - 4;
    break;

    default:
    digitalWrite(airValve, HIGH);
    digitalWrite(airPump, LOW);
    delay(50);
    digitalWrite(airValve, LOW);
    digitalWrite(airPump, LOW);
    delay(50);
    digitalWrite(airValve, LOW);
    digitalWrite(airPump, HIGH);
    delay(50);
    digitalWrite(airValve, LOW);
    digitalWrite(airPump, LOW);
  }
}
```