



WIRELESS ELECTRO CARDIO GRAPH (ECG)

SAMER HAMEED MAJEED

SEPTEMBER 2014

WIRELESS ELECTRO CARDIO GRAPH (ECG)

**A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED
SCIENCES OF
ÇANKAYA UNIVERSITY**

**BY
SAMER HAMEED MAJEED**

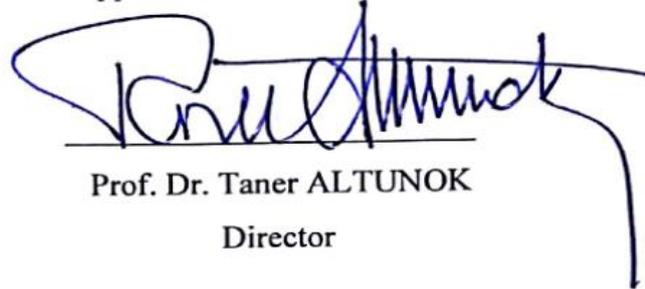
**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
DEGREE OF
MASTER OF SCIENCE
IN
THE DEPARTMENT OF
DEPARTMENT OF COMPUTER ENGINEERING**

SEPTEMBER 2014

Title of the Thesis : **Wireless Electro Cardio Graph (ECG)**

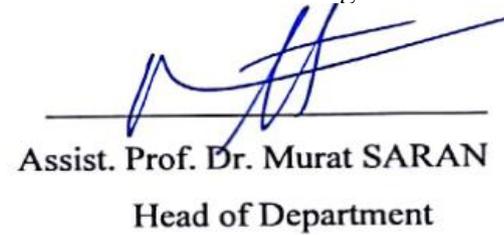
Submitted by **Samer Hameed MAJEED**

Approval of the Graduate School of Natural and Applied Sciences, Çankaya University.



Prof. Dr. Taner ALTUNOK
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science.



Assist. Prof. Dr. Murat SARAN
Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality,



Assist. Prof. Dr. Yuriy ALYEKSYEYENKOV
Supervisor

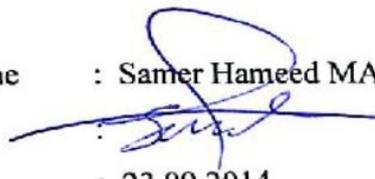
Examination Date: 23.09.2014

Examining Committee Members

Assoc.Prof.Dr. Fahad JARAD	(UTAA)	
Assist.Prof.Dr. Yuriy ALYEKSYEYENKOV	(Çankaya Univ.)	
Assist.Prof.Dr. Nurdan SARAN	(Çankaya Univ.)	

STATEMENT OF NON-PLAGIARISM PAGE

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name : Samer Hameed MAJEED
Signature : 
Date : 23.09.2014

ABSTRACT

WIRELESS ELECTRO CARDIO GRAPH (ECG)

MAJEED, Samer Hameed

M.Sc., Department of Computer Engineering

Supervisor: Assist. Prof. Dr. Yuriy ALYEKSYEYENKOV

September 2014, 62 pages

The aim of this thesis is designing a system that analyze signal produced by the patient heart. This system enables to monitor patients' ECG data remotely instead of keeping them in the emergency care. The system composes of three parts. The first part is electronic circuit that can read the data from patient heart and convert it from analogue to digital by using special analogue to digital convertor and send data to a special server in the clinic care. Electronic equipment is controlled by Arduino microcontroller. The second part of this system is creating monitoring_server that stores the digital signal in a database. MySQL database is used in this part. To read patient data, software tool was developed by using C# language. This tool provides access to the stored data and check the status of the patient. And also this tool contains other features such as analysis of critical cases. The medical specialist will be immediately informed and he can escalate the case to higher level. The last part is creating diagnose system that can classify the patient data to diseases classes by

using data mining techniques. This element of the system helps to check and make a good decision for long data duration such as twenty four hour patient data recording.

Keywords: Electro Cardio Graph (ECG), Arduino, MySQL, C#.

ÖZ

KABLOSUZ ELEKTRO KARDIYO GRAFIĞI (EKG)

MAJEED, Samer Hameed

Yüksek Lisans, Bilgisayar Mühendisliği Anabilim Dalı

Tez Yöneticisi: Yrd. Doç. Dr. Yuriy ALYEKSYEYENKOV

Eylül 2014, 62 sayfa

Bu tezin amacı, hastanın kalp kası tarafından üretilen elektriksel bir sinyal olan EKG sinyalinin analizi için bir sistem tasarlanmasıdır. Bu sistem, hastaların EKG verilerinin acil durumda izlenmesini sağlar. Sistem üç bölümden oluşmaktadır. İlk bölüm, hastanın kalbinden EKG verisinin okunması ve özel bir analog-dijital çevirici kullanılarak digitale çevirilmesi ve klinik bakımına özel sunucuya verinin gönderilmesini sağlayan elektronik devredir. Elektronik cihaz, ARDUINO mikro denetleyici tarafından kontrol edilir. Bu sistemin ikinci bölümü, veri tabanında bulunan dijital sinyalin, bir sunucuda oluşturulması ve izlenmesidir. Bu kısımda MySQL veritabanı kullanılmıştır. Hasta verilerini okumak için, yazılım aracı C # dili kullanılarak geliştirilmiştir. Bu araç depolanan verilere erişimi ve hastanın durumunu kontrol imkanı sağlar. Ve ayrıca bu araç, bu tür kritik durumlarda analizi gibi diğer bazı özellikler de içerir. Eğer yüksek seviyeli bir durum varsa, tıp uzmanı hemen haberdar edilebilecektir. Son bölümde, veri madenciliği teknikleri kullanılarak hasta verilerinin sınıflandırılmasına imkan veren bir teşhis sistemi

oluřturulmuřtur.. Sistemin bu elemanı, yirmi drt saat hasta verisi gibi uzun sreli veri iin iyi bir kontrol yapmaya ve karar vermeye yardımcı olur.

Anahtar Kelimeler: Elektro Kardiyo Grafiđi (EKG), Arduino, MySQL, C#.

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to Assist. Prof. Dr. Yuriy ALYEKSYEYENKOV for his supervision, special guidance, suggestions, and encouragement through the development of this thesis.

It is a pleasure to express my special thanks to my family and my friends for their valuable support.

TABLE OF CONTENTS

STATEMENT OF NON PLAGIARISM.....	iii
ABSTRACT.....	iv
ÖZ.....	vi
ACKNOWLEDGEMENTS.....	vii
TABLE OF CONTENTS.....	viii
LIST OF FIGURES.....	xi
LIST OF TABLES.....	xiii
LIST OF ABBREVIATIONS.....	xiv

CHAPTERS:

1. INTRODUCTION.....	1
1.1. Historical View of Electro Cardio Graph (ECG).....	1
1.2. Heart Cavities and Wave Generation.....	3
1.3. Heart Disease Kinds.....	5
1.3.1. Sinus bradycardia (slow heart rate).....	5
1.3.2. Sinus tachycardia (fast heart rate).....	6
1.3.3. Paroxysmal supraventricular tachycardia (PSVT).....	7
1.3.4. Sinus bradycardia (slow heart rate).....	8
1.4. Literature Survey	9
1.5. Thesis Motivation.....	11
2. BACKGROUND REVIEW.....	13
2.1. Electronic Circuit Components.....	13
2.1.1. INA126 operational amplifier.....	13
2.1.2. High pass filter.....	14
2.1.3. 741 amplifier	14

2.1.4.	Low pass filter.....	15
2.1.5.	Arduino UNO R3.....	16
2.1.6.	Wireless transmission.....	16
2.2.	Signal Diagnose Algorithms.....	17
2.2.1.	Principal component analysis.....	17
2.2.1.1.	Principal component analysis mathematics...	18
2.2.2.	Discrete wavelet transform.....	21
2.2.2.1.	Daubechies.....	21
2.2.2.2.	Coiflets.....	22
2.2.2.3.	Symlets.....	22
2.2.3.	Classification.....	23
2.2.3.1.	Artificial neural network.....	23
2.2.3.2.	K-nearest-neighbors.....	24
2.2.3.3.	Support vector machine (SVM).....	24
2.2.3.4.	Naive bayes.....	24
2.2.3.5.	Decision tree.....	25
3.	SYSTEM DESIGN.....	27
3.1.	Electronic Circuit.....	27
3.1.1.	Attaching electrodes to patient body.....	27
3.1.2.	Using INA126 amplifier.....	28
3.1.3.	High pass filter.....	28
3.1.4.	741 amplifier.....	29
3.1.5.	Low pass filter.....	29
3.1.6.	Circuit connection diagram.....	29
3.1.7.	Arduino UNO R3.....	30
3.2.	Database and Monitoring System.....	33
3.2.1.	Database design.....	34
3.2.2.	Client implementation.....	37
3.3.	Signal Diagnose System.....	40

3.3.1.	Principal component analysis.....	40
3.3.2.	Naive bayes with discrete wavelet transform.....	48
3.3.2.1.	Detecting waves from signal.....	49
3.3.2.2.	Applying naive byes algorithm.....	52
4.	CONCLUSION AND FUTURE WORK.....	61
4.1.	Conclusion.....	61
4.2.	Future Work.....	61
REFERENCES.....		R1
APPENDICES.....		A1
A.	CURRICULUM VITAE.....	A1

LIST OF FIGURES

FIGURES

Figure 1	Heart cavities.....	3
Figure 2	Heart repolarization and depolarization.....	5
Figure 3	Sinus bradycardia.....	6
Figure 4	Sinus tachycardia.....	7
Figure 5	Paroxysmal supraventricular tachycardia (PSVT).....	8
Figure 6	Junctional rhythm.....	9
Figure 7	Thesis motivation steps.....	12
Figure 8	INA126 circuit diagram.....	13
Figure 9	High pass filter frequency response and cutoff frequency.....	14
Figure 10	High pass filter circuit.....	14
Figure 11	Low pass filter connection.....	15
Figure 12	Low pass filter cutoff frequency.....	15
Figure 13	Arduino schematic.....	16
Figure 14	XBee module.....	17
Figure 15	Mean face creation.....	18
Figure 16	Normalized matrix.....	19
Figure 17	Difference between output multiplications.....	19
Figure 18	Selecting K eigenfaces.....	20
Figure 19	Return K eigenfaces to its original size.....	20
Figure 20	Face space for euclidian distance.....	21
Figure 21	Daubechies families.....	22
Figure 22	Coiflets families.....	22
Figure 23	Symlets families.....	23

FIGURES

Figure 24	DWT flow chart.....	23
Figure 25	Data set example.....	25
Figure 26	Decision tree example.....	26
Figure 27	(a)Electrodes and (b) Electrodes positions.....	27
Figure 28	Multisim circuit connection.....	30
Figure 29	Arduino ADC diagram.....	31
Figure 30	ADC conversion clock.....	32
Figure 31	Database and software connection diagram.....	34
Figure 32	Database tables design.....	35
Figure 33	Patient profile update page.....	38
Figure 34	Displaying patient data.....	39
Figure 35	Operator software mailbox.....	40
Figure 36	Selecting training dataset path.....	41
Figure 37	Selecting test dataset path.....	41
Figure 38	Selecting test sample signal.....	42
Figure 39	T matrix data sample.....	43
Figure 40	Plotting meanface.....	44
Figure 41	DWT filter.....	49
Figure 42	Waves peaks.....	51
Figure 43	Typical ECG signal.....	52

LIST OF TABLES

TABLES

Table 1	Sinus Bradycardia Features	6
Table 2	Sinus Tachycardia Features	7
Table 3	Paroxysmal Supraventricular Tachycardia	8
Table 4	Junctional Rhythm Features	9
Table 5	Xbee types and Frequency Modes	16
Table 6	Patient-Daily Table Description	36
Table 7	Eigenfaces Sample Data	45
Table 8	PCA Algorithm Result	48
Table 9	DWT Extracted Features	53
Table 10	Features Time Scale	54
Table 11	High Priority Features	54
Table 12	Features Threshold	55
Table 13	Features Final Shape	56
Table 14	Samples Explanation	57
Table 15	Attributes Values	58
Table 16	Sample Classes Probability	59

LIST OF ABBREVIATIONS

SQL	Structured Query Language
ECG	Electro Cardio Graph
SA	Senatorial Node
BPM	Beat Per Minute
VT	Ventricular Tachycardia
VF	Ventricular Fibrillation
PSVT	Paroxysmal Supraventricular Tachycardia
AV	Atrioventricular Node
mv	Millivolt
Hz	Hertz
INA	Instrument Amplifier
ADC	Analogue to Digital Convertor
PWM	Pulse Width Modulation
ISCP	Integra Serial Control Protocol
GSM	Global System for Mobile Communications
PCA	Principal Component Analysis
DWT	Discrete Wavelet Transform
Kbit/Sec	Kilo Bit Per Second
MCU	Micro Controller Unit
IDE	Integrated Development Environment
USB	Universal Serial Bus
GPRS	General Packet Radio Service
ICD	International Classification Diseases
ANN	Artificial Neural Network
SNN	Simulated Neural Network
OMC	Operation and Monitoring Centre

CHAPTER 1

INTRODUCTION

1.1. Historical View of Electro Cardio Graph (ECG)

In the 1880s, it was discovered that all cardiac contractility produces electrical changes in all parts of the body. Physiologists endeavored to find a way to manufacture, and reliable messaging measurements for this activity decreased because the heart muscle is the electrical impulse generated in the SA node. Keeping the rhythmic beating of the heart through a series of structured discharge originates in the sinus node of the right atrium and through the atria ventricular node and the bundle of nerve and muscle fibers (the owner of the package) to the ventricle. A device for measuring this uses a mercury column, which rises and falls with changes in voltage, but measurements require large amounts of time and many bills that the device was useless for practical comments. The Einthoven¹⁹⁰³ solves the problem with the invention of the string galvanometer, a machine that consists of a very fine quartz wire, which functions in a magnetic field. Being very sensitive, the wire moves in response to even the smallest current. Upon increasing the pointer and recording their movements on film, the Einthoven can make accurate measurements of the electrical activity of the heart. It also the device and it is used in a large number of patients; the Einthoven came to recognize remarkable electrical activities that are consistent with damage or abnormalities in certain areas of the heart. With the knowledge of these electrical patterns and its greatly improved ability to monitor, doctors can diagnose irregularities in the functioning of the heart. Today, with the use of technology pioneered by the Einthoven, physicians routinely make detailed measurements of the electrical activity of the heart by connecting electrodes to specific points in the chest, arms and legs of a person. This is called a record for the

ECG. The planning cycle of the heart of a P wave, QRS complex and T wave, and the foundation, which follows the P wave to appear last. The most important components of the ECG P wave is the deviation caused by the current originating in the atrium; the QRS complex, showing the passage of electrical activity in the ventricles; and the T wave, where the ventricles reset themselves. The waves shown on an ECG trace the path of the pulse as it propagates through the heart. Irregularities in an ECG reflect disorders in the muscle, blood supply or nerves controlling the heart. An ECG can also help doctors to monitor the effects of heart drugs. The senatorial node (SA node) is a dominant pacemaker of the heart. Stimulation and activity is known as a "sinus rhythm". Each pulse from the SA node pacemaker extends through both atria as an advancing wave of depolarization. In normal driving, the cardiac contractions are very organized and programmed to depolarize the best cavity and the contract prior to the bottom cavity and heart rate remains between 60 and 100 beats per minute. The largest portions of the heart, the ventricles, are the areas where potentially the most dangerous problems occur since it is the muscle that is mainly responsible for the circulation of blood to the body's largest parts. Ventricular electrical abnormalities include ventricular tachycardia (VT) and ventricular fibrillation (VF). Ventricular tachycardia is a potentially lethal arrhythmia that can cause the heart to be incapable of pumping adequate amounts of blood through the body. Heart rate does not arise from the node, but from abnormal tissue in the ventricles. It may be 160 to 240 beats per minute and unusual. Ventricular tachycardia occurs in about 2 out of 10,000 people. Ventricular fibrillation is a lethal and disorganized pace and this is synonymous with clinical death. When this occurs, they pump little or no blood from the heart. It may be ventricular fibrillation in itself, and often when myocardial ischemia is present, or after VT continues for a long time. Sudden cardiac death is the final result of the rapid treatment without VT or ventricular fibrillation. Defibrillation is a proven lifesaver; however, it must take place within 10 minutes of cardiac arrest. People who survive an episode of ventricular fibrillation can be treated with transplants Pour defibrillator. Implantable heart defibrillators Pour (ICDs) are devices that provide a variety of treatments for electrical ventricular arrhythmias. DAI is a diagnosis alarm system and emergency medical technique. One of the main tasks is to distinguish

whether the CIE irregular heartbeat is normal; VT VF is an automatically normal sinus rhythm when the heart requires electrical treatment. In recent years, technological advances have produced a significant reduction in the size of the device to simplify the implant as compared to the sub-pectoral implant abdomen that was initially used. However, the current hardware requires refinement in its ability to determine the appropriate treatment for various arrhythmias as soon as possible and avoid inappropriate treatment of up and ventricular tachycardia [1].

1.2. Heart Cavities and Wave Generation.

Heart this strange muscle control most of our body parts because it supply them with oxygen and all items to make it work perfectly, this muscle (heart) is composed of different parts. These parts are responsible for generating the electrical signal (ECG), in other words our heart works like a function generator that generates voltages and signals at all times to keep us alive. In this thesis, these parts are discussed briefly, as shown in the below figure.

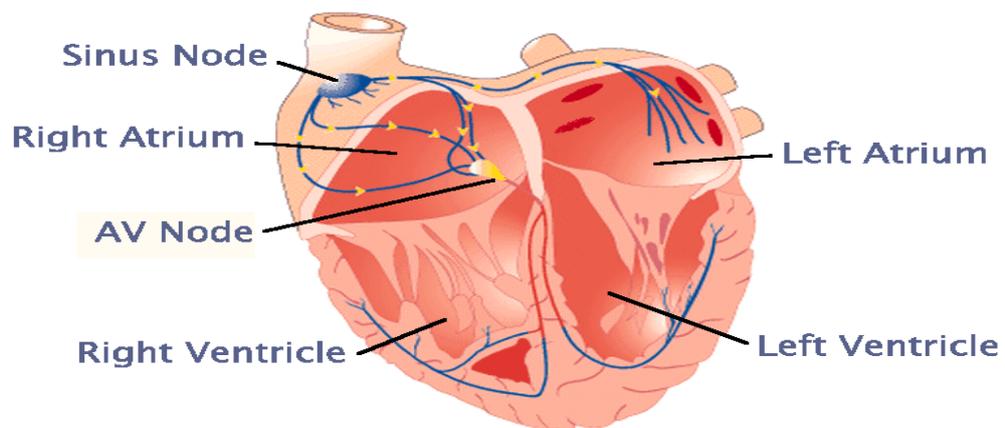


Figure 1: Heart cavities

1. **Sinoatrial node (SA Node):** Is a small piece of tissue in the heart that is also called the peacemaker, which is responsible for impulse generation to beat the heart at regular intervals of 60 to 70 beats per minute for a healthy and resting human being. This tissue is located in the right atrium of the heart. The

pacemaker enables all other heart cells to polarize and depolarize for the pumping of blood.

2. **Right Atrium:** This heart chamber is located on the right side and functions to receive blood that is fully deoxygenated and then pumps this blood to the lungs through the ventricle chamber to be supplied with fresh oxygen. After supplying oxygen to the blood stream, it returns to the heart, but now through the left atrium.
3. **Left Atrium:** This heart chamber is located in the left side of the heart and is receives oxygen-rich blood with from the lungs and then pumps this blood to the left ventricle of the heart to pass it through the body.
4. **Right Ventricle:** This chamber is located in the lower right side of the heart and receives deoxygenated blood from the right atrium and then pumps this blood to the lungs for more oxygen.
5. **Left Ventricle:** This heart chamber is located in the lower left side of the heart and receives oxygenated blood from the upper atrium chamber then pumps it to the body.

There are many heart parts. Some of these parts will be described in this work. As mentioned before, these parts pump blood through depolarizing and repolarizing. This repolarization and depolarization process can be measured through (P-Q-R-S-T) waves, which are composed of very small values (0.5mv) and low frequencies (ranging from 1 to 150 Hz) as shown in Figure 2.

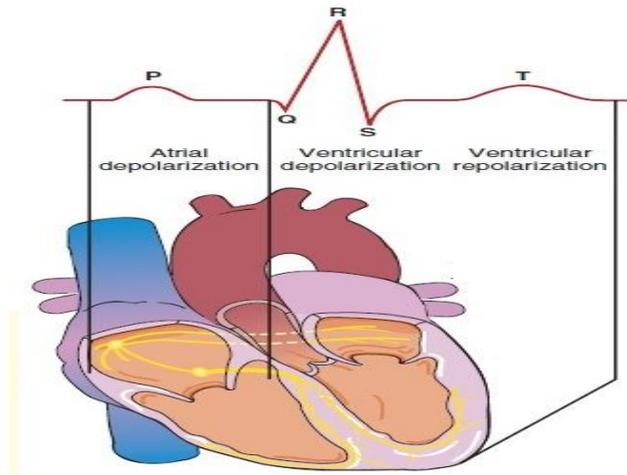


Figure 2: Heart repolarization and depolarization

- 1- Heart repolarization is a chemical action that returns a negative charge to the cells and its resting state.
- 2- Heart Depolarization: is the electrical charge of a cell such that the cells are altered by a shift of electrolytes on either side of the cell membrane. This behavior stimulates the fiber of the muscle to contract.

1.3. Kinds of Heart Disease

There are several types of heart disease, some of which diseases are described briefly in the following pages.

1.3.1. Sinus bradycardia (slow heart rate)

Heart function in healthy people in a rest state should be in normal and fixed interval behavior and that lead to a normal heart beat of 60 to 100 beats per minute (BPM). However, this type of disease has a feature such that the beats are below the normal range. If the heart fails to pump sufficient amounts of fully oxygenated blood, bradycardia can then be a serious problem. Avoiding this problem is possible with an ingrain peacemaker and some treatment. Fig. 3 shows the ECG signal for the sinus bradycardia disease.

Sinus Bradycardia

■ Results from slowing of the SA node.

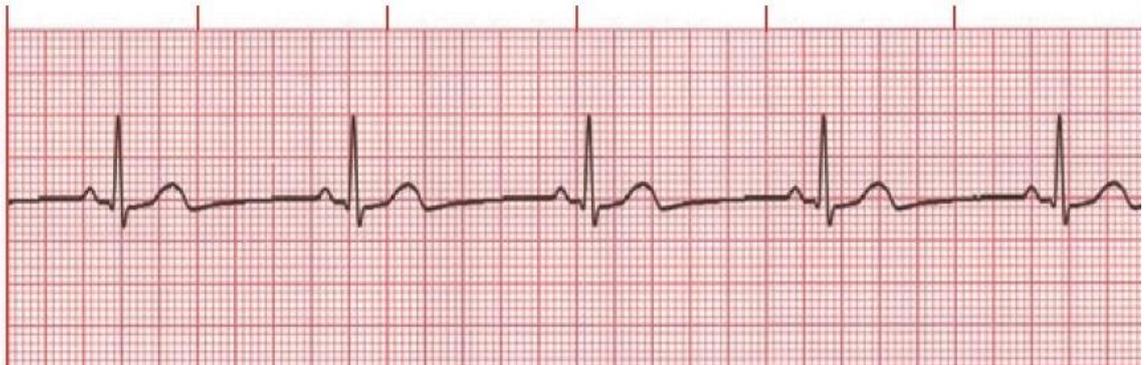


Figure 3: Sinus bradycardia

Table 1: Sinus Bradycardia features

Sinus Bradycardia	
Heart Beat	Less than 60
Rhythm	Regular
P waves	Normal and upright
P R interval	Normal (0.12 to 0.2) second
QRS interval	Normal (0.06 to 0.1) second

1.3.2. Sinus tachycardia (fast heart rate)

Tachycardia is a faster heart rate than normal for an adult human. It occurs when there is something abnormal in the heart to produce rapid beats. This disease increases the probability of stroke or sudden death. Fig.4 shows the heart ECG signal for sinus tachycardia.

Sinus Tachycardia

■ Results from increased SA node discharge.

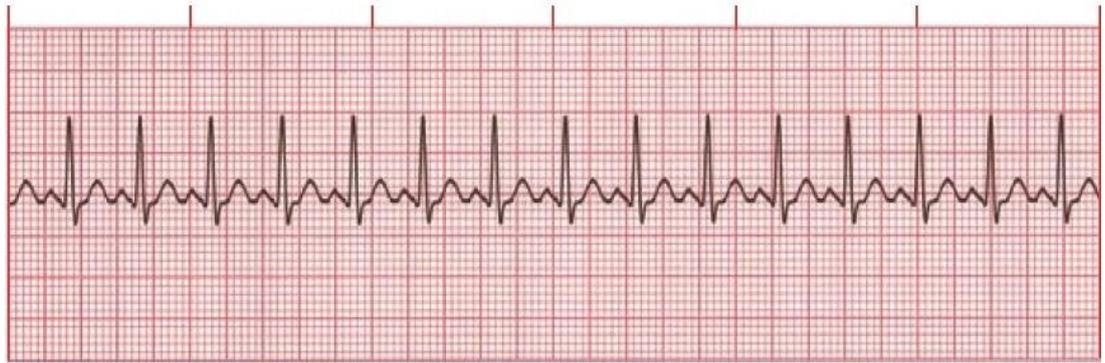


Figure 4: Sinus tachycardia

Table 2: Sinus Tachycardia Features

Sinus Tachycardia	
Heart Beat	More than 100
Rhythm	Regular
P waves	Normal and upright
P R interval	Normal (0.12 to 0.2) second
QRS interval	Normal (0.06 to 0.1) second

1.3.3. Paroxysmal supraventricular tachycardia (PSVT)

Abnormal activity and conduction in the heart causes rapid excessively fast atrial and ventricular beats. This type of disease can appear suddenly and disappear just as suddenly without any warning. Furthermore, when it starts, it may, depending on the patient, be still for a few seconds or for some time, even up to a few hours. Fig. 5 shows the heart ECG signal for this disease.

Paroxysmal Supraventricular Tachycardia (PSVT)

- PSVT is a rapid rhythm that starts and stops suddenly.
- For accurate interpretation, the beginning or end of the PSVT must be seen.
- PSVT is sometimes called paroxysmal atrial tachycardia (PAT).

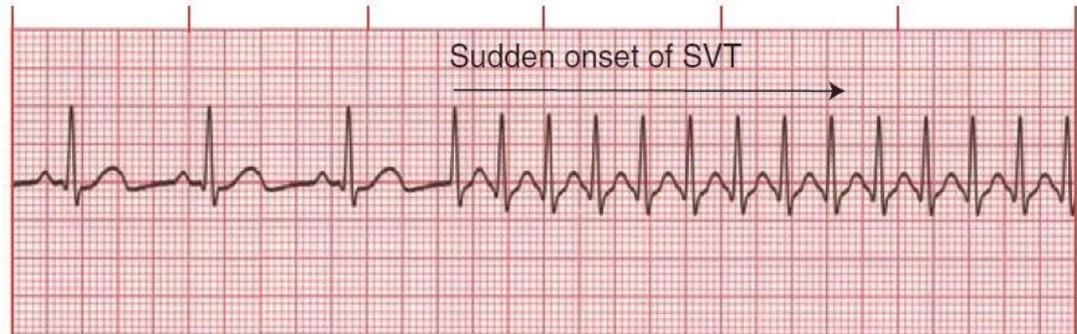


Figure 5: Paroxysmal supraventricular tachycardia (PSVT)

Table 3: Sinus Supraventricular Tachycardia Features

Paroxysmal Supraventricular Tachycardia (PSVT)	
Heart Beat	150 to 250
Rhythm	Regular
P waves	Frequently buried in preceding T waves and difficult to see
P R interval	Usually not possible to measure
QRS interval	Normal (0.06–0.10 sec) but may be wide if abnormally conducted through ventricles

1.3.4. Junctional rhythm

Normal heart conduction activation starts near the SA node, but in cases of junctional rhythm, the signal is generated near the AV node. The heart beats at < 60 and the P wave is frequently absent, hence the name *buried P Wave*. Sometimes this wave will appear as an inverted signal.

Junctional Arrhythmias

- The atria and SA node do not perform their normal pacemaking functions.
- A junctional escape rhythm begins.

Junctional Rhythm

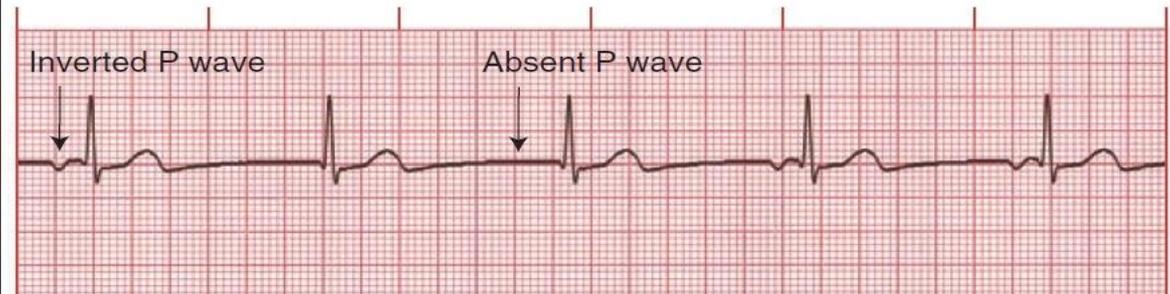


Figure 6: Junctional rhythm

Table 4: Junctional Rhythm Features

Junctional Rhythm	
Heart Beat	40 to 60
Rhythm	Regular
P waves	Absent, inverted and buried
P R interval	None or short
QRS interval	Normal (0.06–0.10 sec)

1.4. Literature Survey

1. The **Wireless ECG/EEG with the MSP430 Microcontroller** thesis by Tamas Hornos [2] designed a hardware system that can collect the ECG signal from a patient and send the data to the local machine wireless by using a special type of micro controller and wireless transmitter. These types of elements are not commercially available on the market. Moreover, the system was designed only to store data in the local machine.

2. **ECG Feature Extraction Using Daubechies Wavelets** by S. Z. Mahmoodabadi^{1,2} (MSc), A. Ahmadian^{1,2} (Phd), M. D. Abolhasani^{1,2} (Phd) [3] designed a feature extraction system that can extract features from an ECG signal by using a specific type of DWT wavelet. This work extracted good results from the signal. The signal was loaded from MIT University. The ECG signal frequency of this work was between 2 and 40 Hz.

3. **ECG Feature Extraction Using Optimal Mother Wavelet** by B. Castro, D. Kogan and A. B. Geva [4] attempted to design an optimal system to extract signal features from a wavelet function. In this work, the authors applied a noise to the signal and then attempted to reconstruct the signal to extract features from different scales of the mother wavelet of this noised signal.

4. **ECG Feature Extraction via Waveform Segmentation** by Antonio Espiritu-Santo-Rincon and Cuauhtemoc Carbajal-Fernandez [5]. The main contribution of this paper is an effective ECG feature extraction method that relies on wavelets and the search for maxima and minima within limited intervals. Due to its relative simplicity, this method seems adequate for its implementation on low power, portable devices. However, further work should be carried out in order to combine manual annotations and an adaptable algorithm that adjusts some parameters depending on the observations made by the specialist.

5. **ECG Feature Extraction Using LCAD** by Rajendra G. Sutar [6]. In this work, Discrete Wavelet Transform is used for feature extraction. DWT allows good time resolution and poor frequency resolution at high frequencies and good frequency resolution and poor time resolution at low frequencies. The algorithm was tested PC and obtained a good performance measure for the extraction of both the time domain as well as the frequency domain ECG parameters on the same platform as a low cost solution.

1.5. Thesis Motivation

The motivation of this thesis is how one can collect patient data remotely, store that patient data in a special type of database and construct a reliable diagnosis system that helps the specialist to make better-informed decisions about patient status. To my knowledge, in the previous works no one appears to have created a dedicated server to store patient daily data. Moreover, in the development of this thesis, the use of data mining techniques to diagnose patient heart status is presented. Below is the work plan of this thesis.

- Design an electronic system that can fetch heart electrical signals (ECG)
- Design and implement a database that can store fetched signals
- Design and implement a software tool that enables the specialist to check patients' recorded data
- Design a system of diagnosis that can assist and support the specialist to analyze patient heart behavior by using a number of data mining algorithms.

Fig. 7 shows the motivation steps of this thesis

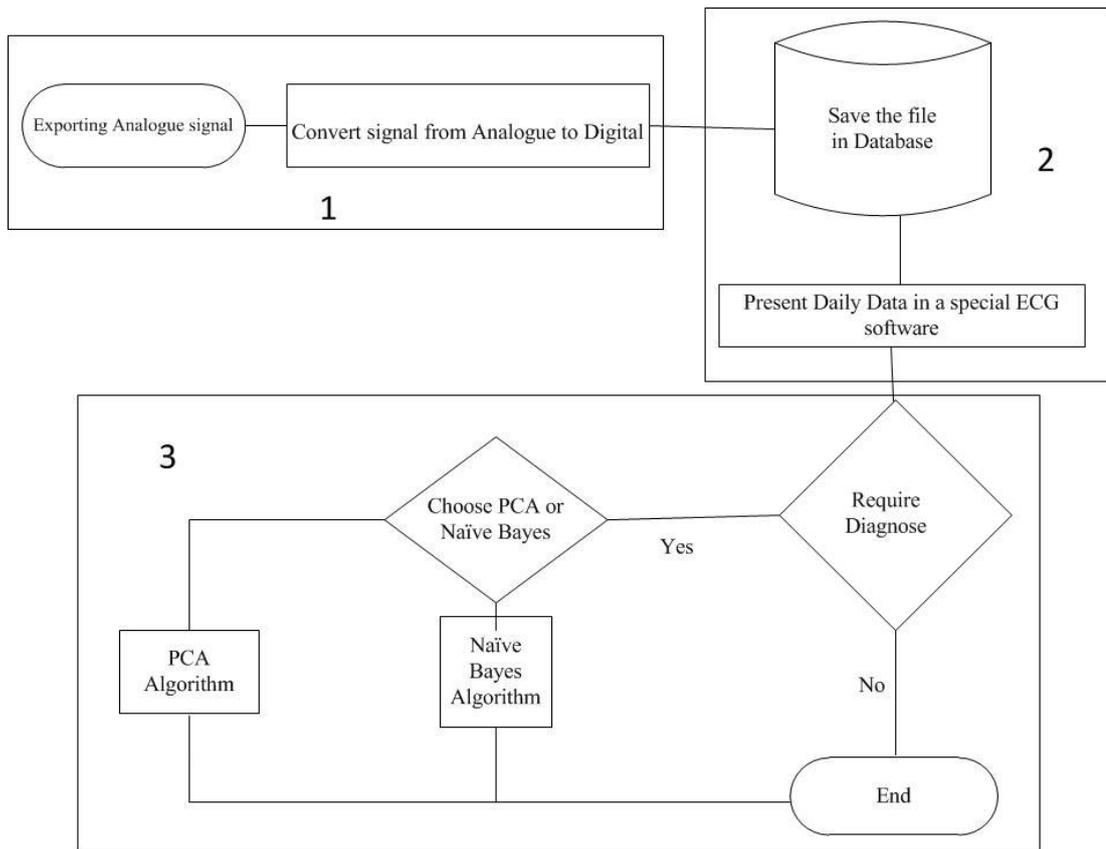


Figure 7: Thesis motivation steps

CHAPTER 2

BACKGROUND REVIEW

2.1. Electronic Circuit Components

The lines below will describe the specification and features of the Electronic circuit components that will be used in the thesis hardware part.

2.1.1. INA126 operational amplifier

INA is an instrument model of amplifiers, 126 is the amplifier type, this amplifier has several features that can be used in our case, one of these features is it's very low voltage sensitivity and high common mode rejection features, also the other feature is the amplification gain, 126 can provide gain up to 1000 times of the input signal. INA Input must be from two sources because the amplifier works as the differential between inputs, in ECG thesis case input will from the right and left chest, and then amplify the output as shown in Fig. 8 of the amplifier

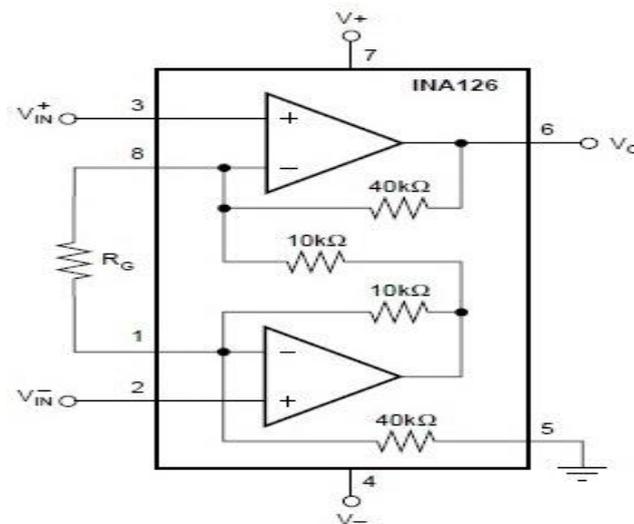


Figure 8: INA126 circuit diagram

2.1.2. High pass filter

A High Pass Filter is used to allow some of the signal with a high frequency component to pass over the limit and cut off the frequencies that are below the limit. High Pass Filters contain many circuits; however, in such circuits, there is no need for complicated circuitry, simple components (such as resistors and capacitors) with a High Pass Filter formula are all that is needed to set the values of these elements. Figure 9 and Figure 10 show the components, how to connect them to make the circuit and the cutoff frequency for this filter.

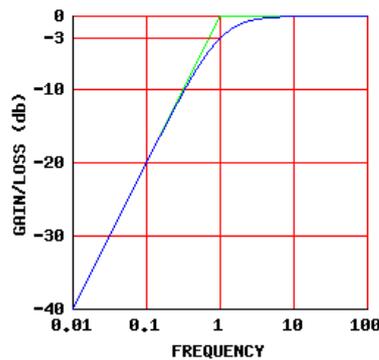


Figure 9: High Pass Filter frequency response and cutoff frequency

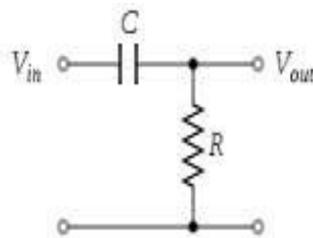


Figure 10: High pass filter circuit

2.1.3. 741 amplifier

This amplifier is well known and can provide high gain. The output of this amplifier depends on the input range. Normally, this amplifier can sense input starting from millivolts, which is useful in the case of this thesis.

2.1.4. Low Pass filter

In this stage, the elements are the same as the High Pass Filter: resistors and capacitors. Moreover, there are several types of Low Pass Filter circuits in the electronic world, but in this ECG case, the filter will be connected in parallel to the 741 amplifier from the output pin as a feedback to the input pin, as shown in Fig. 11.

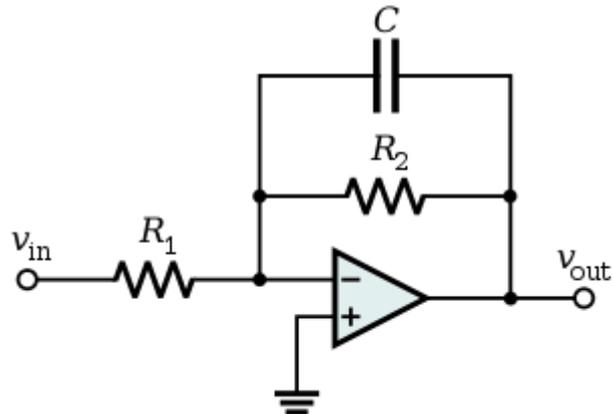


Figure 11: Low pass filter connection

The Low Pass Filter enables the low frequencies to pass through the output pin by setting the cutoff frequency through the values of the resistor and capacitor elements. Fig. 12 shows the cutoff frequency.

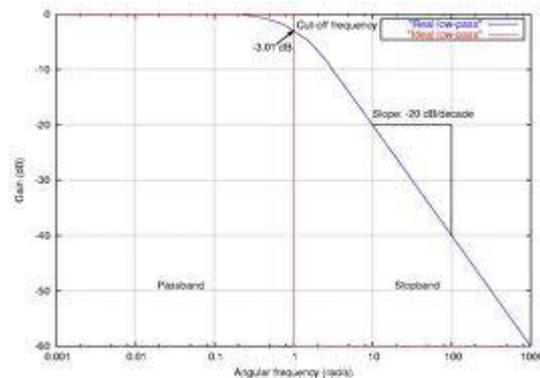


Figure 12: Low pass filter cutoff frequency

2.1.5. Arduino UNO R3

The Arduino is a microcontroller kit that can provide many functions. The kit is composed of several components such as an ADC, a timer, a clock, a PWM. The main element is the Atmel microcontroller. This MCU can be programmed to do several functions. The ADC is a prescaler unit, which means it can generate a multi-type of digital output depending on the clock on the unit. The ADC prescaler can set the clock [7]. This unit can communicate through several interfaces and protocols, such as the serial port (COM Port), through ISCP one or by connecting a specific shield to the unit to communicate by WIFI, ZigBee or even through a GSM SIM Card [7]. Fig. 13 shows the Arduino UNO R3 kit.

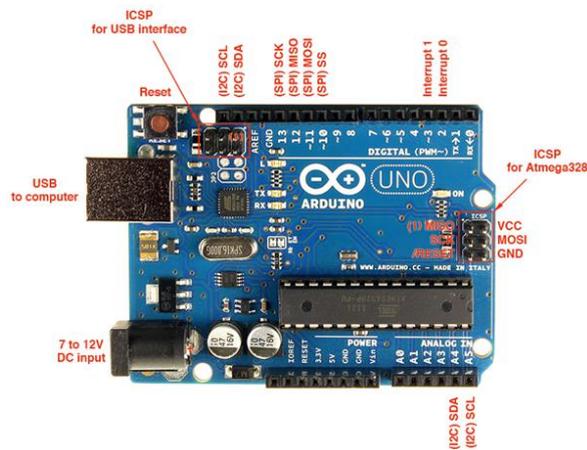


Figure 13: Arduino schematic

2.1.6. Wireless transmission

The Xbee module is a wireless transmitter based on radio frequency that can cover up to 100 meters. There are three varieties of this module [8], which depend on frequency range, as shown in Table 5.

Table 5: Xbee Types and Frequency Modes

Center frequency (MHz)	Number of channels (N)	Channel (k)	Channel center frequency (MHz)
868	1	0	868.3
915	10	1 - 10	$906 + 2(k-1)$
2450	16	11 - 26	$2405 + 5(k-11)$

All these ranges are free license and can be used without any need for permission. However, each type has its own number of channels. The last one uses the 2.4 GHz frequency range and 16 transmission channels. This Xbee module depends on the Zigbee protocol and can transfer data at up to 250 Kbit/Sec [8], which is suitable for ECG requirements using WIFI as a suitable module. However, this is more than necessary. In the ECG case, the data stream needs to be sent continuously from the UNO R3 to the computer. Therefore, the transmission mode should be without the beacon. Fig. 14 shows the XBee module.



Figure 14: XBee module

2.2. Signal Diagnose Algorithms

There are algorithms in data mining science that can be used to analyze and classify the signal from the features that are selected from this signal. Data Mining science is classified into different parts: clustering, classification, time series, etc. In ECG, the signal classification part is used for the patient signal. Two of the classification methods are the Principal Component Analysis (PCA) and the Naïve Bayes algorithm.

2.2.1. Principal component analyses

The Principal Component Analysis (PCA) is a powerful technique that has been used in signal recognition. It is a statistical method that is used to reduce the large dimensionality of data space to a smaller dimensionality of feature space. This

method is used when there is a real and a strong relation within a data space. This method can help us to remove redundancy, to make predictions, as well as perform feature extraction and data compression. Because PCA is a classical technique that can do something in the linear domain, applications with linear models are suitable, these include signal processing, image processing, system and control theory, communications, etc. There are several applications for signal recognition. Moreover, they can be categorized into signal identification or signal classification. The main idea of using the PCA for signal recognition is to express the large 1-D vector of pixels into the compact principal components of the feature space. This can be called eigenspace projection. Eigenspace is calculated by identifying the eigenvectors of the covariance matrix derived from a set of facial signals (vectors). The details are described in the following section. Section 2.2.1.1 describes the mathematical formulation of the PCA [13].

2.2.1.1. Principal Component Analysis mathematics (PCA)

Let us suppose there are M vectors of size 1

$$x_i = [p1 : : : pN]^T ; i= 1 ; : : : ; M \quad (2.1)$$

The signals are mean-centered by subtracting the mean signal from each signal vector. Let m represent the mean signal. In this way, the common features are removed from each signal. Only the unique features are kept, as shown in Fig. 15.

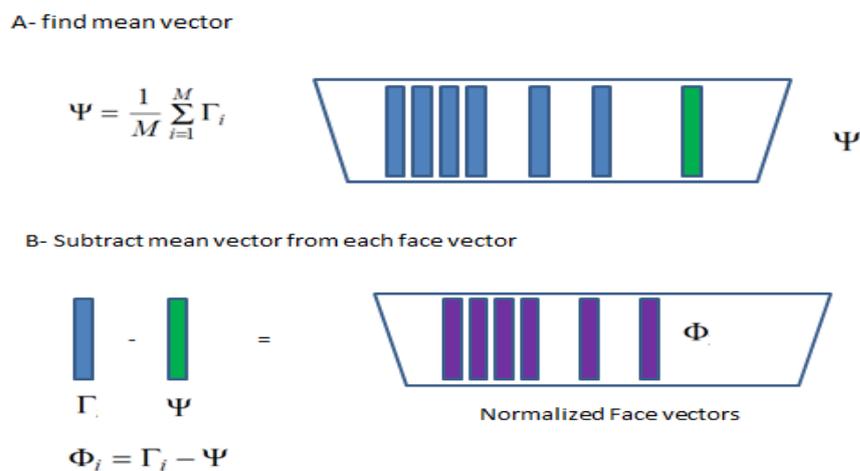


Figure 15: Mean face creation

The output matrix of this subtraction is the normalized face vectors, this matrix must be multiplied by its transpose to create the covariance matrix, as shown in Fig. 16. The importance of this matrix is to calculate **eigenvalues** and **eigenvectors** (**eigenfaces**).

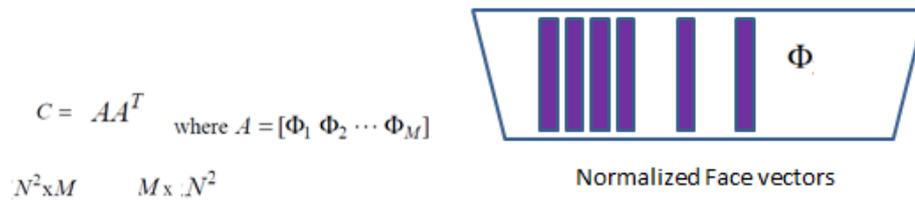


Figure 16: Normalized matrix

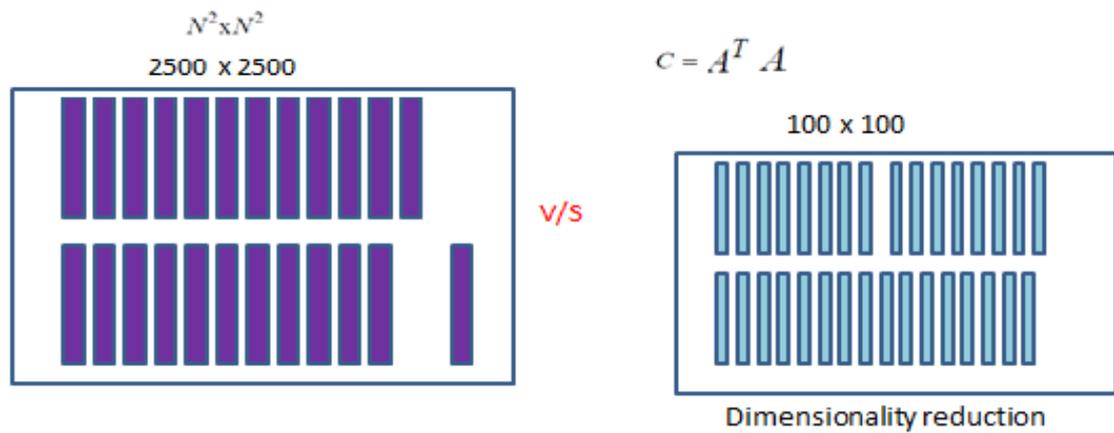


Figure 17: Difference between the output multiplications

$$C = A^T A \tag{2.2}$$

A mathematical trick is used to reduce the complication of multiplication. When multiplying the transpose with the original matrix in normal multiplication, many values and deviations are yielded while on other side the dimensionality is reduced. The eigenvectors corresponding to nonzero eigenvalues of the covariance matrix which produce an orthonormal basis for the subspace within most data, which can be represented with a small amount of error [7]. The eigenvectors are sorted from high to low eigenvalues. The eigenvector which is associated with the largest eigenvalue reflects the greatest variance in the signal. Meanwhile, the smallest eigenvalue is associated with the eigenvector that finds the least variance. They decrease in an

exponential fashion, which means that roughly 90% of the total variance is contained within the first 5% to 10% of the dimensions. From this result, the K eigen faces can be selected. This K must be less than the number of signals in the dataset. Selecting the K depends on the criteria and the threshold value of the diameter value of the covariance matrix. In this study, the eigenvalue is greater than 1.

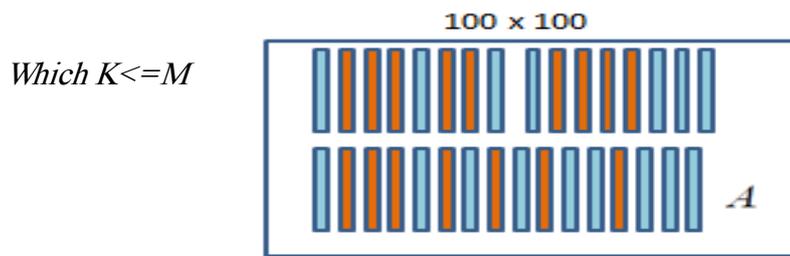


Figure 18: Selecting K eigenfaces

Since these K faces have a low scale value and are to be projected with their original dimension, the K Eigen faces must be multiplied with their normalized matrix A, as shown in Fig. 19.

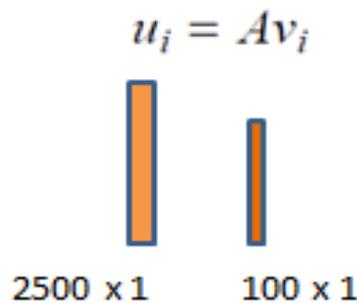


Figure 19: Return K Eigen faces to its original size

In order to measure the signals and specify the closest one to the unknown sample, a weight will be provided for each signal in the dataset. In the comparison step, the Euclidian distance formula can be used to subtract the unknown sample from the K Eigen faces. (Afterwards, we project them and provide them weight.) The smallest value is the nearest signal to the unknown sample, as shown in Fig. 20.

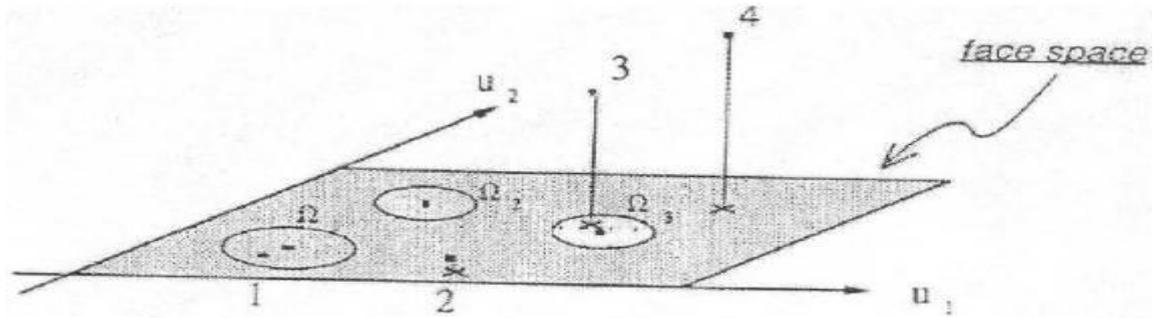


Figure 20: Face space for Euclidian distance

In the Euclidian distance formula, the signal data set and unknown data vector must be considered by using the square root of this vector in the formula to calculate the smallest value, as shown in the formula below.

$$k = || (\Omega - \Omega k) || \quad (2.3)$$

2.2.2. Discrete Wavelet Transform

This is the mathematical tool of choice. In addition to being an efficient, highly intuitive framework for the representation and storage of multi-resolution signals, the DWT provides a powerful insight into frequency characteristics. The wavelet transform is based on small waves of variable frequency and limited duration, which are denominated wavelets. A wavelet decomposition of a signal divides the input signal into several sub bands, which contain features at different scales. The advantage of this is that small features like micro classifications are dominant in one sub band, whereas larger features such as masses prevail in a different sub band. There are many wavelet families that are presented below.

2.2.2.1. Daubechies

Daubechies, one of the brightest stars in the world of wavelet research, invented what are called *compactly supported orthonormal wavelets*, thus making discrete wavelet analysis practicable.

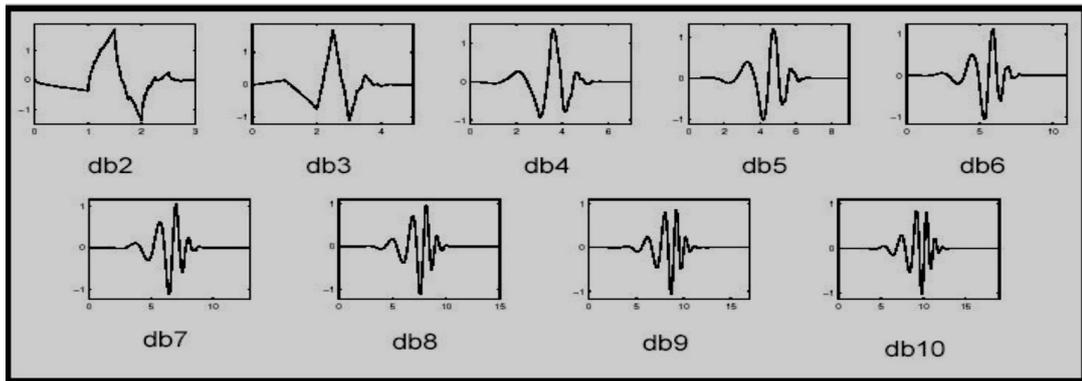


Figure 21: Daubechies families

2.2.2.2. Coiflets

These have been built by Daubechies at the request of R. Coifman. The wavelet function has $2N$ moments equal to 0 and the scaling function has $2N-1$ moments equal to 0, as shown in Fig. 22. The two functions have a support of length $6N-1$. [16]

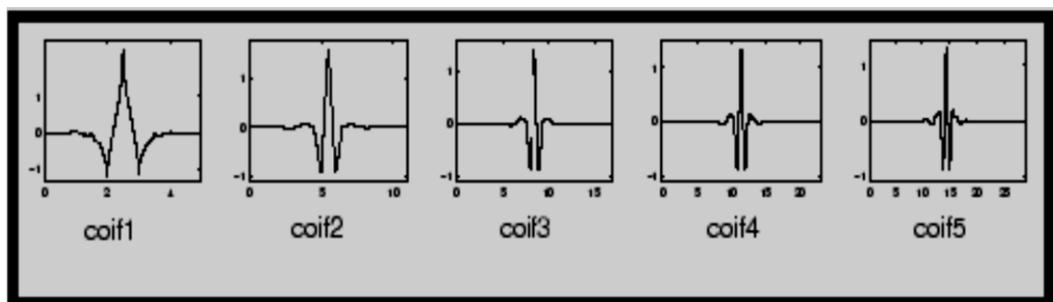


Figure 22: Coiflets families

2.2.2.3. Symlets

Symlets are nearly symmetrical wavelets proposed by Daubechies as modifications to the db family. The properties of the two-wavelet families are similar, as shown in Fig. 23.

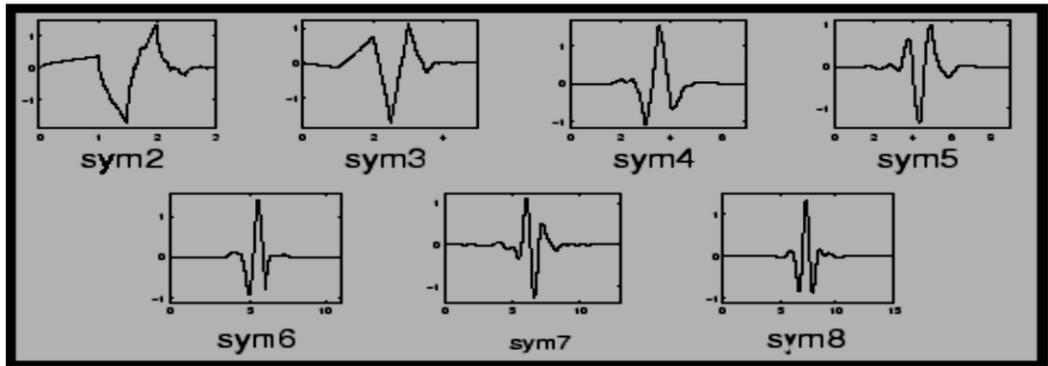


Figure 23: Symlet families

As mentioned above, this method divides the signal to several sub bands. The division is carried out by applying a high pass and low pass filter to the signal to obtain an approximate and detailed coefficient, as shown in Fig. 24.

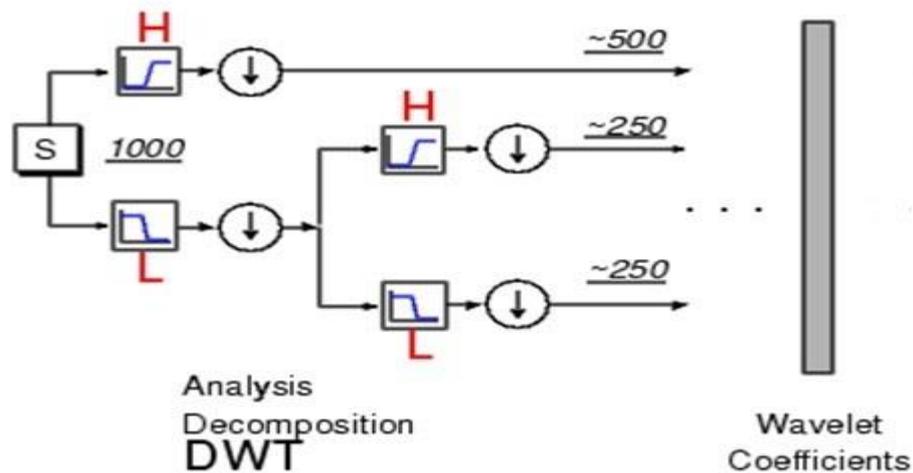


Figure 24: DWT flow chart

These coefficients are important; the values of the required features can be derived from these coefficients [8].

2.2.3. Classification

2.2.3.1. Artificial neural network

This concept was inspired by systems of Biological Learning (BLS), where a huge number of neurons are connected interrelatedly forming a complex network. This net

has a great ability to process information/data in part of second efficiently of BLS due to parallelism in processing which distribute operations over more than one neuron. From here began the concept of ANN neural networks. ANN simulates BLS in distributing and processing operations in parallel. In brief, ANN (also known as simulated neural network (SNN)) is a group of data structures and software that contains a number of processors. Each processor has a small amount of knowledge with access to data in its own memory. Initially, this network must be controlled by a number of rules and some data.

2.2.3.2. K-nearest-neighbors

Based on known similar samples, the classifier (KNN) K-nearest neighbors differentiates unknown patterns by computing distances between every unknown sample to every known sample, and then selects the nearest sample. Unknown patterns are assigned to that class which contains the nearest patterns in the same known sample. For this type of deterministic classifier, it is necessary to have a training set that is not too small, and a good discriminating distance.

2.2.3.3. Support vector machine (SVM)

SVM is a machine learning method that is based on the principle of structural risk minimization and performs well when applied to data outside the training set. Then it formulates micro classification detection as a supervised-learning problem and applies SVM to develop the detection algorithm. SVM is used to detect each location in the image, regardless of the presence of a micro classification. The ability of SVM to outperform several well-known methods developed for the widely studied problem of micro classification detection suggests that SVM is a promising technique for object detection in a medical imaging application.

2.2.3.4. Naïve Bayes

Bayes Theorem comprises a few techniques and Naïve is one of these techniques. Naïve is a probability classifier that considers all attributes independently and

contributes to the probability of a certain decision. Naïve is an efficient technique that works with better results in a complex situation, such as in computer-aided diagnosis, which works better than one might expect. Because independent variables are assumed, only the variances of the variables for each class need to be determined. [13]

$$p(c / f_1, \dots, f_n) = p(c) = \frac{p(c)p(f_1, \dots, f_n / c)}{p(f_1, \dots, f_n)} \quad (2.4)$$

where P is the probability, C is the class variable and F1.....FN are feature variables F1 through Fn. The denominator is independent of C.

2.2.3.5. Decision tree

In the process of finding a model that describes the classes of data and uses this model to specify the class of unknown objects, a non-parametric supervised learning method is used for classification and regression. There are several examples of this algorithm. Decision tree J48 implements Quinlan's C4.5 algorithm to generate a pruned or unpruned C4.5 tree. C4.5 is an extension of Quinlan's earlier ID3 algorithm. The decision trees generated by J48 can be used for classification. J48 builds decision trees from a set of labeled training data using the concept of information entropy. It splits the data into smaller subset and each attribute is used to make a decision of the class type [15].

Predictors				Target
Outlook	Temp	Humidity	Windy	Play Golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mild	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes
Sunny	Mild	Normal	False	Yes
Rainy	Mild	Normal	True	Yes
Overcast	Mild	High	True	Yes
Overcast	Hot	Normal	False	Yes
Sunny	Mild	High	True	No

Figure 25: Data set example

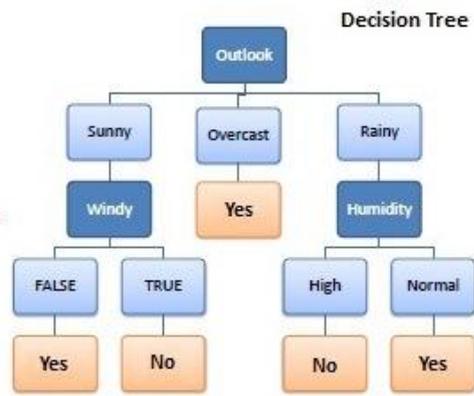


Figure 26: Decision tree example

CHAPTER 3

SYSTEM DESIGN

In this chapter, system implementation will be explained from hardware circuit to the diagnosis part as referenced in Fig.7 in Chapter 1.

3.1. ELECTRONIC CIRCUIT

Hardware is the first part of the system that is referenced as number 1 in Fig.7 and it will be explained in the following sections.

3.1.1. Attaching electrodes to the patient's body

As mentioned in the previous chapter, the human heart works similarly to a function generator such that it continues to generate signals over time. This signal has features that necessitate their extraction from the signal. However, before extracting these features, reading the ECG signal from the body is required.

In order to fetch the ECG signal, normally physical electrodes are required to be attached to the chest at specific places around the heart. Three, five, twelve and even fifteen rods can be used, depending on the patient case, to fetch the data from different angles. Fig. 27 shows the physical rod.

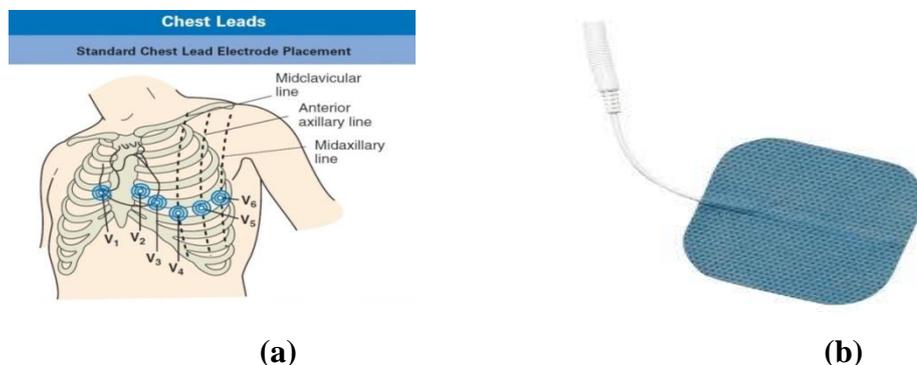


Figure 27: (a)Electrodes and (b) Electrodes positions

After attaching the rods to the chest, the cables from the output of the rods are connected as an input to the circuit. The challenge at this stage is to design a circuit that can read a very low voltage (0.5 mv) and low frequencies (1 to 150 Hz). The input from the cables will be treated as an input to a special type of amplifier designed for industrial and medical cases.

3.1.2. Using the INA126 amplifier

An input is taken from the right and left sides of the chest as an input to the INA126. The amplifier works as a differential between the inputs and then amplifies the output. At this point, after the input has been obtained by the amplifier, care is needed since the signal is not only a heart signal. In fact, it contains noise embedded within the ECG signal. The source of this noise may be anything in the environment such as power lines, microwave devices or television. This noise cannot be avoided; however, it can be reduced by using a band pass filter for the required signal. In the first stage, low amplification gain to the signal is used. This gain is 5, which is the default gain of the amplifier by omitting pins 1 and 8 and leaving them unconnected. The gain is set to 5 because if the signal is amplified to the readable range, the noise will also be amplified, and in this case it will be difficult to control the noise and isolate it from the signal. After amplification, there will be a signal with 2.5 mv plus a new noise signal. This noise is generated by the electronic components and its DC, The value of this noise is less than 1 Hz. Moreover, to reduce environmental noise, it is better to use shield input cables such as coaxial cables. Attaching the electrodes near the shoulders also helps to reduce muscle movement noise.

3.1.3. High pass filter

As mentioned previously, the signal starts from 1 Hz. In order to avoid this DC noise, using a high pass filter will allow all frequencies above one hertz to pass through this filter. The high pass filter is not a complicated one. It is made by using a resistor and capacitor and follows the high pass filter equation as shown in formula (3.1).

$$f_c = \frac{1}{2\pi RC} \quad (3.1)$$

The required frequency is 1 Hz. Then what is needed is to set the R and C values by assuming one of them and calculating the other element value. At this point, after passing the signal through the filter, it should be working at the second amplification stage, which is the real amplification stage by using a normal amplifier such as the 741 series.

3.1.4. 741 amplifier

At this stage, a signal will be received as an input from the high pass filter. This signal needs to be amplified to a readable value since it is still too small at about maximum 2.5 mv. Using a normal amplifier such as the 741 with a gain of 100 provides an output value of 250 mv.

3.1.5. Low pass filter

At this stage, care should be taken from the signal frequency as the work should be in the range of (1 Hz to 150 Hz) and as mentioned previously, there are different frequencies combined with the main signal. Therefore, in order to keep the work in a safe domain, it is preferable to use a low pass filter at this stage. The resistor and capacitor is set to the frequency limit of 150 Hz, as it has been known that the frequency of 150 is all that is required to set the Resistor and Capacitor values, by assuming one of them as a value and calculating the second one by using the formula below.

$$f_c = \frac{1}{2\pi RC} \quad (3.2)$$

3.1.6. Circuit connection diagram

After all the circuit elements and their values have been analyzed and described, the circuit can be drawn now by using Multisim software by downloading an ECG library from the Internet to the software. This library generates an ECG signal from

different sources such as the right chest, left chest and left leg by connecting the wires from two sources to the ECG fetching circuit and checking the output in the oscilloscope, as shown in Fig. 28. In this circuit, the signal is assumed to be pure without any noise. Therefore, even if the filters are removed, the oscilloscope will not show any noise in the output.

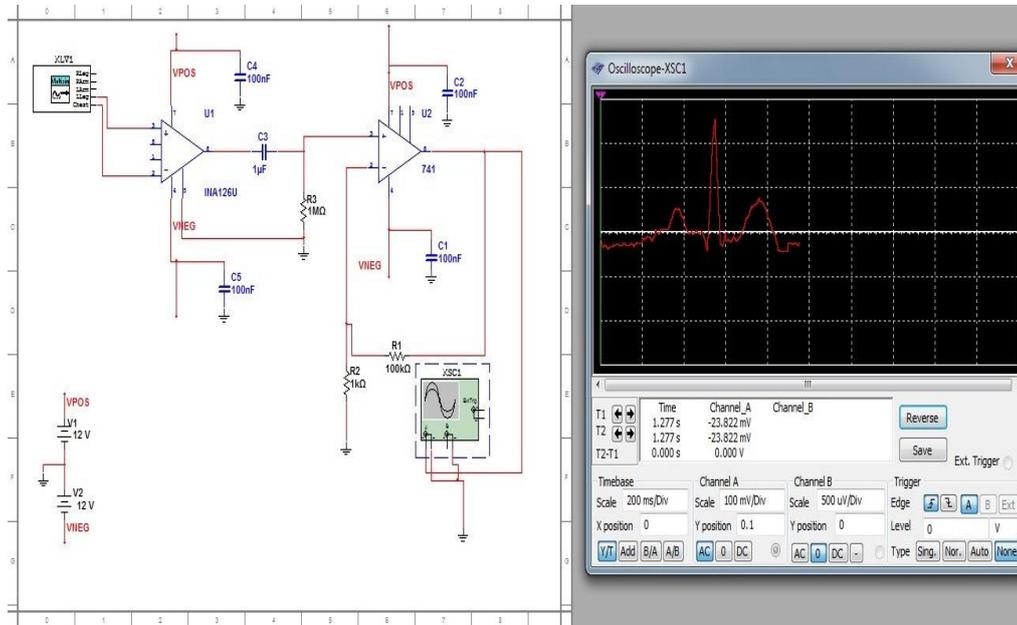


Figure 28: Multisim circuit connection

In this circuit, the noise must be eliminated well since all frequencies will affect the heart signal reading. One of these noises is the muscle noise due to moving arms. This circuit will be used remotely and portably with the patient, putting these rods down the side of the wishbone instead of the arms or hands. This is preferable since the patient cannot move this part of the body. Of importance is the connection using the Einthoven triangle.

3.1.7. Arduino UNO R3

After amplifying the signal in the last stage to its final shape, it is necessary to convert it from the analog to digital. To do this, the Analogue to Digital Convertor (ADC) is used. In the UNO R3 kit, the AD convertor can be found. This convertor has different references to set the max conversion value. For the above circuit, the 5-

volt reference will be selected as a reference to compare with the input signal. For the conversion job, R3 is 10-bit resolution. Now the digital output depends on the sampling frequency and this type of kit is composed of an Atmel micro-controller. Normally, this MCU is programmable. The sampling frequency is set by sending the below commands to the MCU from its IDE.

```
Const unsigned char PS_128 = (1 << ADPS2) | (1 << ADPS1) | (1 << ADPS0);
```

```
ADCSRA |= PS_128;
```

Writing the first command in the definition part of the sketch sets the values for the Prescaler parameters and the second command is written in the setup part of the sketch to use the prescaler of the ADC clock, Fig. 29 shows the prescaler circuit in the R3 kit [7].

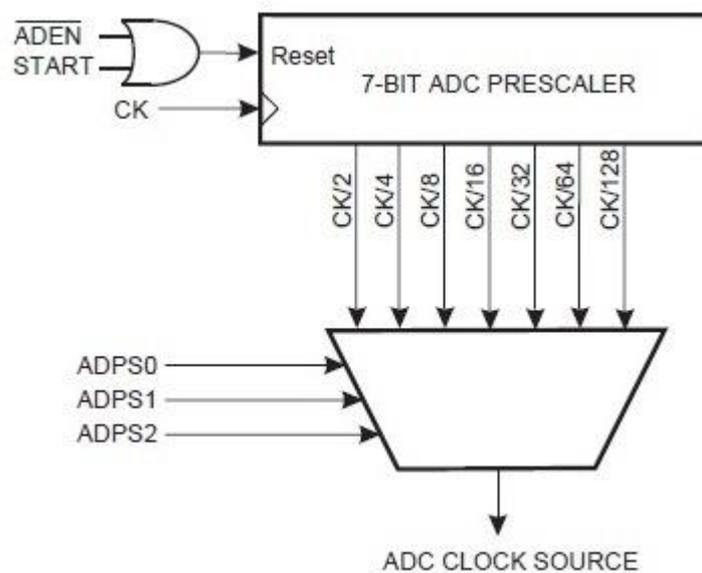


Figure 29: Arduino ADC diagram

As shown in the figure, setting the prescaler parameters sets the frequency of the ADC. The clock of the kit is 16 MHz and the ADC clock operates between 50 kHz and 200 kHz for 10-bit conversion. However, if the conversion resolution needs to be less than 10 bits, it is possible to set the clock to more than 200 kHz. For the ECG circuit conversion case, the clock is set to 125 kHz by selecting the 128 prescaler value and using the formula below.

$$\text{Kit clock} / \text{prescaler} = \text{Sampling frequency} \quad (3.3)$$

Now, after setting the sampling frequency, it is time to know the max conversion values per second. Knowing this value will provide benefits from the conversion time parameter and divide the sampling frequency over the conversion time. The ADC conversion time is 13 ADC clock as shown in the formula below.

$$\text{Sampling frequency} / \text{conversion time} = \text{Conversion values} \quad (3.4)$$

$$125000/13=9600$$

The Arduino R3 ADC needs for the first conversion attempt a 25 ADC clock and then needs a 13 ADC clock for all other conversion attempts since in the first attempt, it needs to update its MUX reference and set the sample and hold element. Moreover, it is very important for us to know the max output value from the ADC. To determine this value, the formula below can be used.

$$\text{ADC} = \frac{V_{IN} * 1024}{V_{REF}} \quad (3.5)$$

Fig. 30 will provide more info about the ADC conversion clocks.

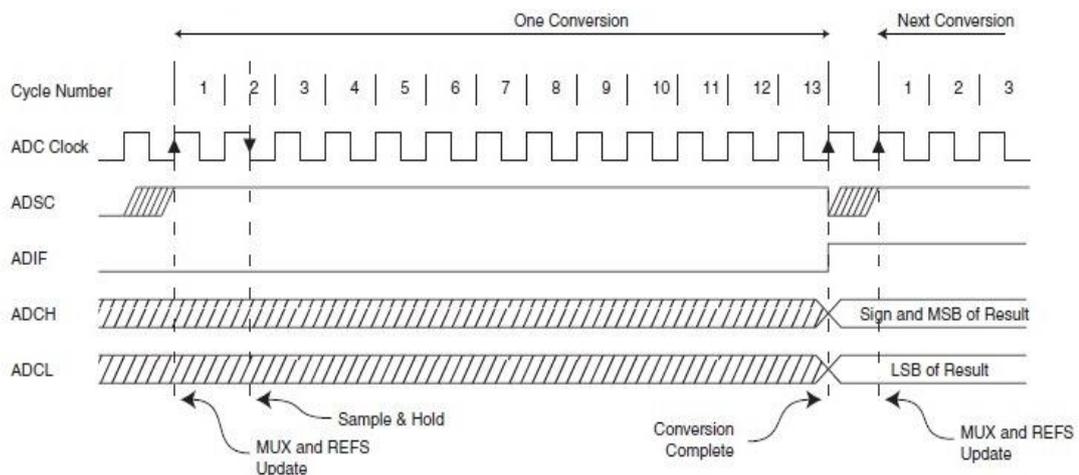


Figure 30: ADC conversion clock

In the ADC module, there is an important issue in the sampling calculation, namely the output and how can it be saved. This output will be saved in a text file. Every

time the sampling ratio increases, its mean sampling period will be decreased and increase its reading data for that will increase the digital output and that leads to increasing text file save, thus needing to take output file size in our consideration in the sampling calculation.

Arduino programming example

```
#define RES 1023
#define AREF 5
Const unsigned char PS_128 = (1 << ADPS2) | (1 << ADPS1) | (1 << ADPS0);
Void setup ()
{Serial begin (9600);
ADCSRA |= PS_128; // set our own prescaler to 128
}
Void loop()
{ Serial println(valtovolt(analogRead(A0)));
Delay (20) ;}
Double valtovolt(double Val)
{Return AREF*(Val/RES);}
```

As shown in the above example, there are three parts to program the Arduino kit. The first part is the definition part, which contains libraries, public variables and other definitions. The second part, which is the setup part, is used for the first power on and loading of software to the kit. It will no longer be accessed after loading, setting the console and selecting the working ports. In the third and final part, an infinite loop is used to read and write data between the Arduino and other peripherals [8].

3.2. Database and Monitoring System

This part will demonstrate the design of the Database, Server and Client part referenced as number 2 in Fig.7. The saved text file in the computer needs to be transferred to a central database. This database is located in a location remote from the patient, such as a hospital OMC room. The server is installed on Windows 7 and

a MySQL database; however, it is more preferable to use Windows 2000 server or 2008 server for its tolerance and services. In the database part, it can continue to use MySQL. Data transfer for each patient will be once per day by sending all patient daily log data. There are several types of file transfer; however, in this thesis, FTP protocol will be used by opening an FTP session between the client and the server every night and then sending the file from the client to the server. In this case, a daily file from any place around the world can be guaranteed. Below is a simple diagram of the project.

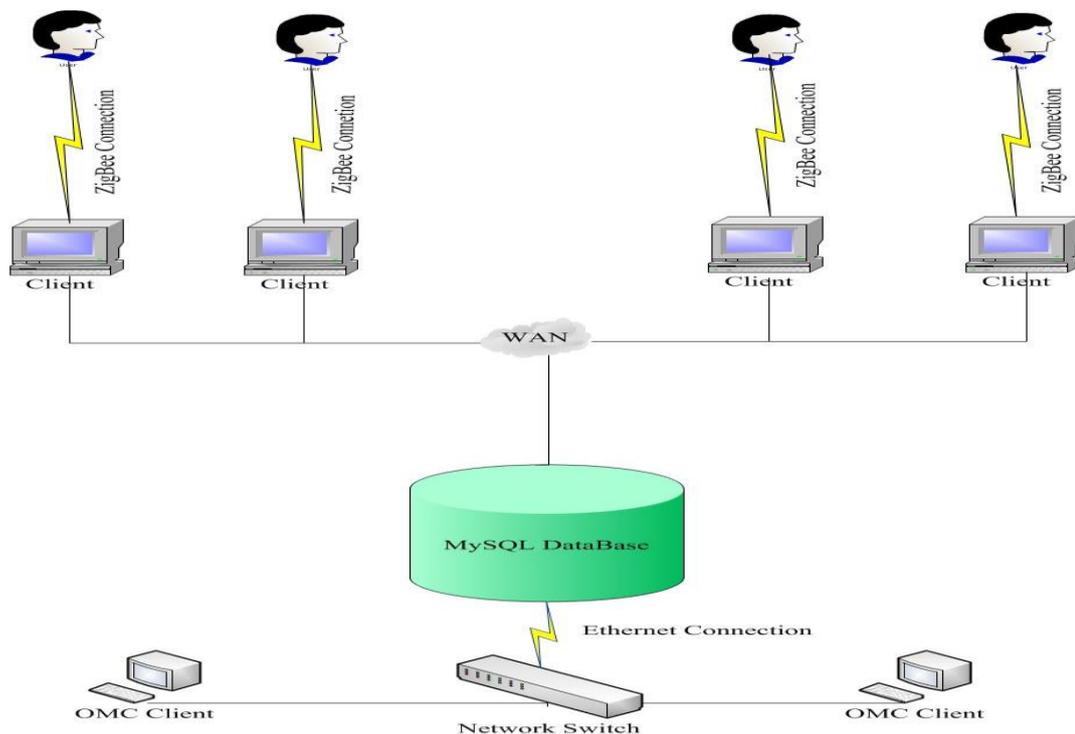


Figure 31: Database and Software connection diagram

3.2.1. Database design

This database comprises two tables: the first one saves **patient profile** data such as name, age, mobile numbe, etc.. This table must be updated before any data is saved from the patient in the second table because the second table depends on the UNO serial number to send the data. This is because the data is unique in each item and cannot be duplicated between devices. In the profile table, it is possible to add,

remove and amend patient data in order to keep it updated with the patient profile; Fig. 32 shows the profile table and its fields.

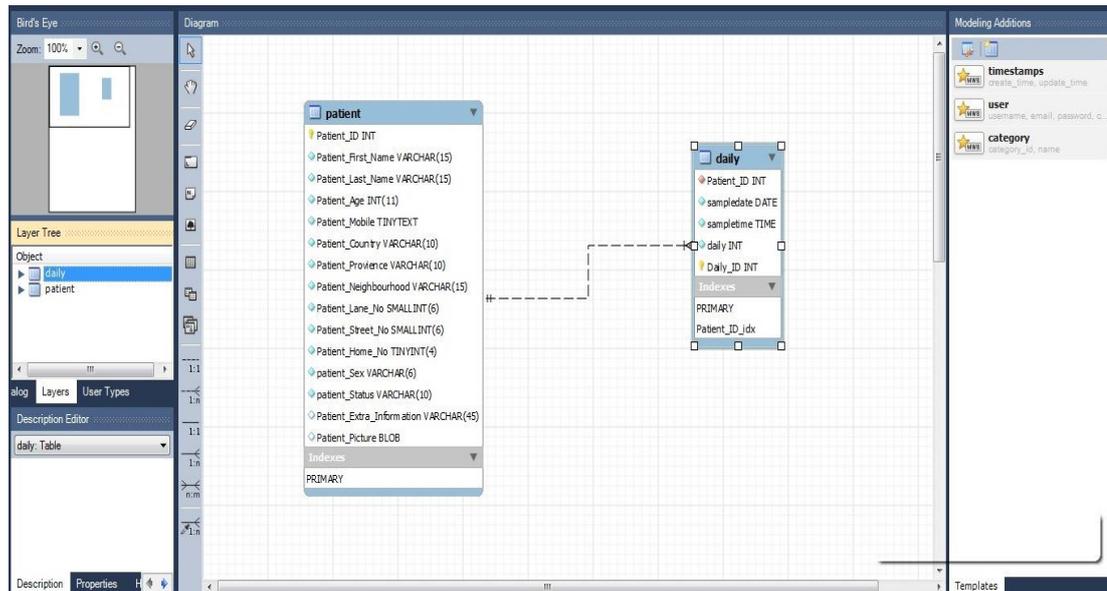


Figure 32: Database tables design

After table creation and inserting each registered patient into the profile table, patients can send his daily data normally to the **patient-daily** table. This table is composed of the fields below. It is necessary to note that this data stream is not real-time data. In fact, it is offline. Normally, the data can be sent after midnight from the patient to the server through the Internet via FTP or any other transfer protocol. To allow an FTP session to work between the patient computer and the Care Server, a batch file needs to be set in the computer. This file contains lines that enable an open FTP session from computer to the server (with Remote side IP and login username and password to the FTP folder of the remote side). This batch file needs to start working for a fixed duration, for example, at midnight (00:00 hours) each day. To set this time for the batch, Windows schedule needs to be edited by providing a batch path to the schedule and the time that it needs to be executed.

Table 6: Patient Daily Table Description

Daily Patient table	
Field name	Field description
Patient-Serial	This field contain UNO Serial Number
Sample-Date	This field contain sample date in the format ddmmyyyy
Sample-Time	This field contain sample time such as hh:mm:mi
Sample-Data	This field contain patient data stream

To load the stream from the test to the database, a normal SQL loader command can be used; this it works from the DOS terminal.

SQL Loader example

*load data local in file 'g:/project/20131225.txt' into table daily fields terminated by ','
lines terminated by '\n';*

The SQL Loader can be done daily and automatically by applying a special schedule according to the required uploading time. This schedule will insert all new data into the database client by client. After uploading all files, the PL-SQL procedure starts to work automatically (Database Job) to check which patient does not send his/her file. It then generates a file containing all patient names and their profile data. This file will help the operator on the server to check the reason for not receiving the file. This complete process enables the system to hold up-to date data.

At this point, the data is transferred from the computer to the server storage and then upload the stream to the database, needing to help the specialist to check their patient daily data that is stored in the server. To do that, it should be implementing the operator software under C# language in order to log in to the database and display the information of the selected patient. This tool has different features such as adding new patients, remove existing patients, updating patient profile data, sending e-mail to with patient log data and most importantly, displaying patient selected timestamp data to enable the specialist to read and diagnose patient status.

3.2.2. Client implementation

- To connect C# with MySQL, a connector should be used by downloading it from the MySQL official web-site because it enables the user to select the reference data source.
- After installing the connector to the C#, the programming side should be started. A login form needs to be created. This form can take its username and password from the MySQL database user privilege, which means that each operator can be provided with a working authority by reading only or reading and writing. Here in this step it must use the connection object from C#. This object comprises the server path and all login information such as username, password and encryption. This object will be sent to the server to compare the information with users' privileges.

Connection object example

```
GlobalVariables.overloadusername= this.Username_txt.Text;  
GlobalVariables.overloadpassword= this.Password_txt.Text;  
MySqlConnection myConn = new  
MySqlConnection("datasource=localhost;port=3306;username="+this  
.Username_txt.Text+";password="+this.Password_txt.Text+");  
myConn.Open();
```

- After the connection object is compared with the MySQL and everything is matched, it can be logged into the client to use the authority.
- To display patient data, we need to create another connection object to allow the user to access the database and retrieve data from the tables.
- Occasionally the retrieved data may be too large, such as when exceeding memory capacity. In this case, the data is saved to a text file on the computer directly and is then read as a new object from a personal

computer, the result of which is displayed on the screen. In this case, the operator will not notice any delay in the query output.

- To add a new patient, there is a specific tab in the client to add new patients with all required information in the profile-table in the database.
- Also there are another two tabs to update or remove patients from the database.

The screenshot shows a web application window titled "Update Patient". The window contains three main sections for data entry:

- Personal Information:** Includes text input fields for "First Name", "Last Name", and "Age". It also features dropdown menus for "Gender" and "Status", and a text input field for "Mobile Number".
- Address Information:** Includes text input fields for "Country", "Province", "Neighbourhood", "Lane", "Street", and "Home".
- Extra Information:** A large, empty rectangular area for additional notes or details.

At the bottom left of the form, there is a "Patient ID" text input field, a "Search" button, and an "Update Patient" button.

Figure 33: Patient profile update page

This page is required to update some patient data such as home address, phone number and any other information. Some time in order to update the profile using only a name is not preferable because there are common names between citizens. Therefore, it is preferable to search in the database for the patient by using as many parameters as possible and then clicking the search button to check whether the result matches the required person.

When the operator completes writing the updated data, he/she simply needs to submit the data to the profile table in the server. Now when the operator wishes to check his/her patient data, it is only necessary to write the required patient ID, sample date and duration, then send this query to the server to receive the results and display

them in the graph window with all the patient profile data. Fig. 34 shows us how to display the patient data.

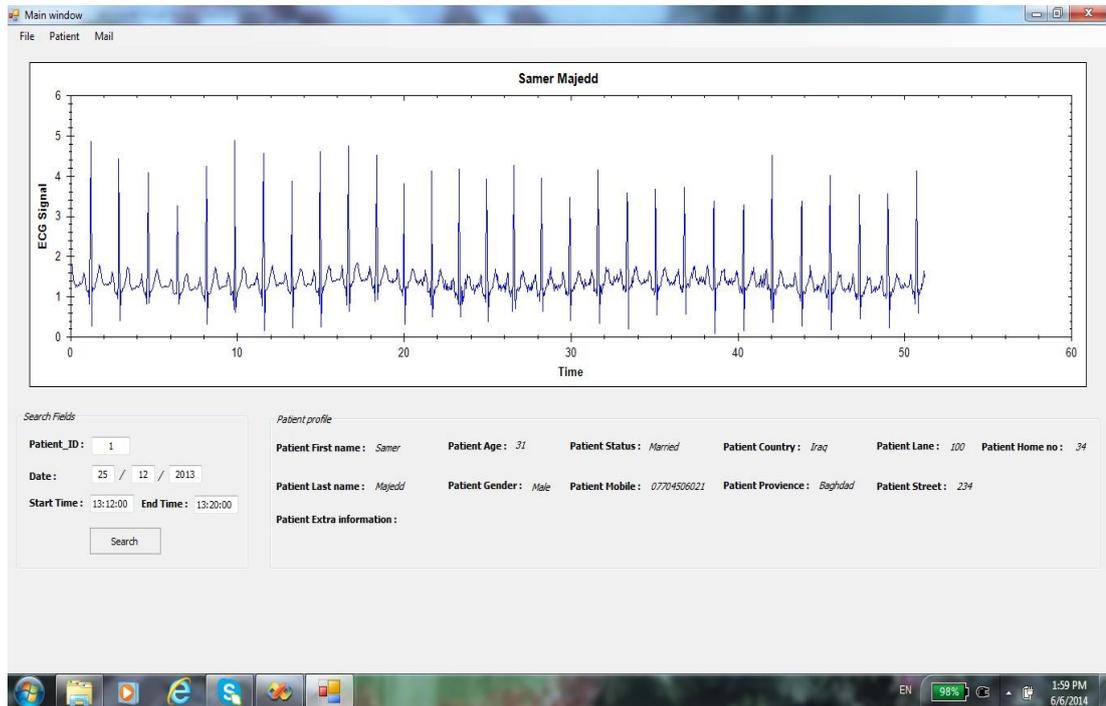


Figure 34: Displaying Patient Data

The last part of this software is the e-mail tab. This tab enables the operator of the software to send any patient data to the higher level, i.e. making escalation to the patient ticket by sending an e-mail that contains a specific duration to make a more detailed diagnosis of the patient case. In this tab, the developer needs to assign the mail type, such as smtp, pop3 or any other mail server type, as shown in Fig. 35.

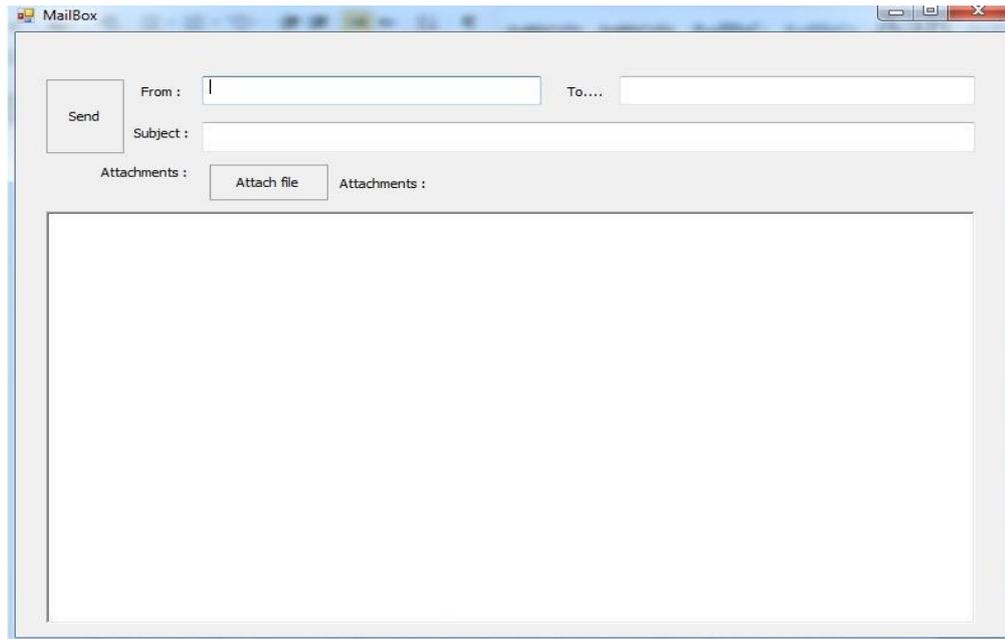


Figure 35: Operator software mailbox

3.3. Signal Diagnose System

After the signal has been generated and saved in the computer, the file is transferred from personal computer to Database Server and provides the operator with visual access to the data stream. The operator should be assisted and supported further by enabling the computer to diagnose the signal conditions and classifying this signal according to the diseases dataset. The signal diagnosis system is the third part of the system that is referenced as number 3 in Fig. 7. It is explained in the following sections. Two algorithms are selected in this thesis and explained in the sections below.

3.3.1. Principal component analysis

In the implementation part of this algorithm, the Matlab software is used. In the computer storage, two folders should be created. The first folder is the **train data set** folder which contains 45 samples. The first 15 samples for the first disease, the second 15 samples for the second disease and the third 15 samples for the third disease. The second folder **test** contains 15 unknown samples that need to be checked with the trained dataset. In the main function of the Matlab code, the **train**

data set folder should be selected and the **test** folder to learn the algorithm, the path of which contains the samples. After that, calling the train function to start reading the training dataset is shown below.

```
clc
```

```
close all
```

```
clear
```

```
% T is the Sample matrix
```

```
% A is the normalized Samples after subtracting the mean sample from them
```

```
% Here we will select the training database and the tested Sample
```

```
TrainDatabasePath = uigetdir('G:\Project\Final update\TrainDatabase', 'Select  
Training database path');
```

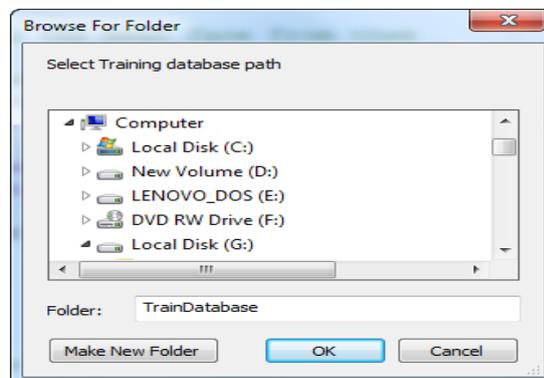


Figure 36: Selecting training dataset path

```
TestDatabasePath = uigetdir('G:\Project\Final update\TestDatabase', 'Select Test  
database path');
```

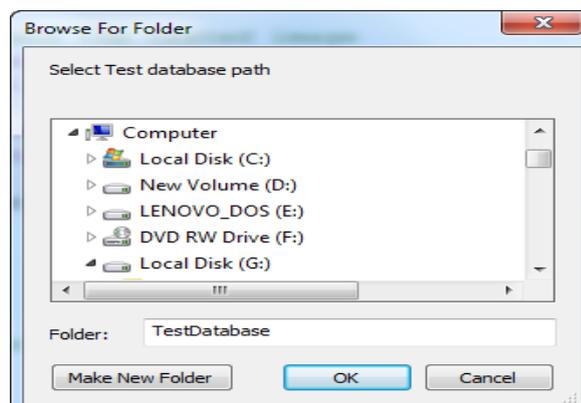


Figure 37: Selecting test database path

```

prompt = {'Enter test Sample name (number between 1 to 45)'};
dlg_title = 'Input of PCA-Based Signal Recognition System';
Testsample = inputdlg(prompt,dlg_title);
Testsample = strcat(TestDatabasePath,'\',char(Testsample),'.txt');

im = load(Testsample);
figure(1);
plot(im);

```

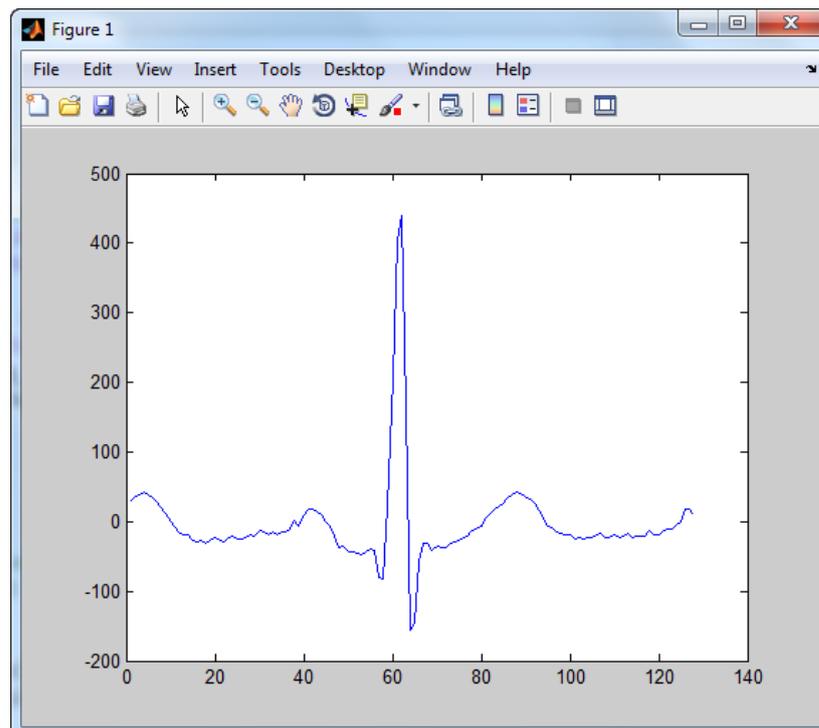


Figure 38: Selecting tested sample signal

```

% Below algorithm functions will be called to the find the
% equivelent signal for the tested one
T = TrainDatabase();

```

Sample dimensions are 128*1 because it is a one-dimensional vector. To create the matrix, all samples should be read and stored one by one. Finally, having 128*45

because the sample number is 45 as explained previously. All of this is done through the `TrainDatabase()`. The code of this function is shown below.

```
function T = TrainDatabase()
filenames = dir(fullfile('G:\Project\Final update\TrainDatabase', '*.txt'));
num_signal = numel(filenames);
T=[];
for n = 1:num_signal
    filename = fullfile('G:\Project\Final update\TrainDatabase', filenames(n).name);
    face = load(filename);
    T=[T face];
End
```

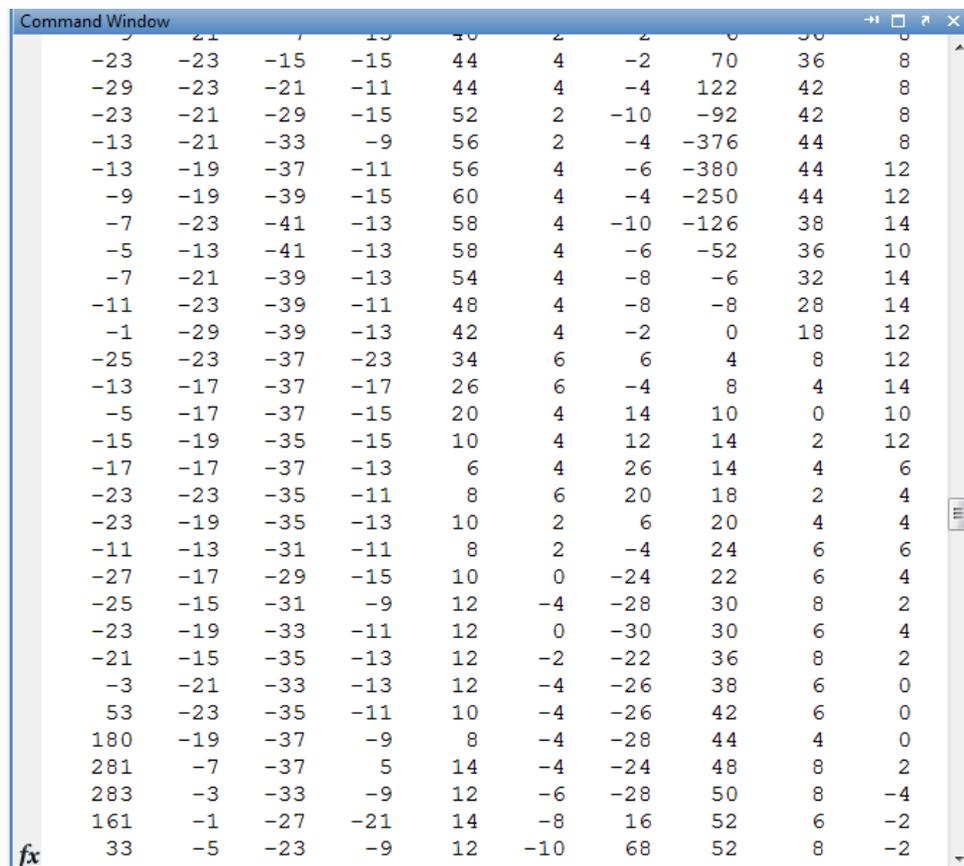


Figure 39: T matrix data sample

After creating the matrix, it should subtract the mean sample from this matrix to keep the unique features, such as the sudden change in the P wave or long duration

between RR peaks. It should also remove the common features, thereby creating a covariance matrix by multiplying the matrix with its transpose to produce the eigenvalues and eigenvectors, and keep the K eigenvectors that are passing the threshold value in the loop procedure. All of this is done through the **Eigenface (T)** function. The code of the **Eigenface (T)** functions is shown below:

% the below commands will use the core of the PCA algorithm which is covariance and it's variables to select the Eigenfaces.

```
function [meanface, Normalized, Eigenfaces] = Eigenface(T)
```

```
meanface = mean(T,2);
```

```
Train_Number = size(T,2);
```

```
Normalized = [];
```

```
for i = 1 : Train_Number
```

```
    temp = double(T(:,i)) - meanface;
```

```
    Normalized = [Normalized temp];
```

```
End
```

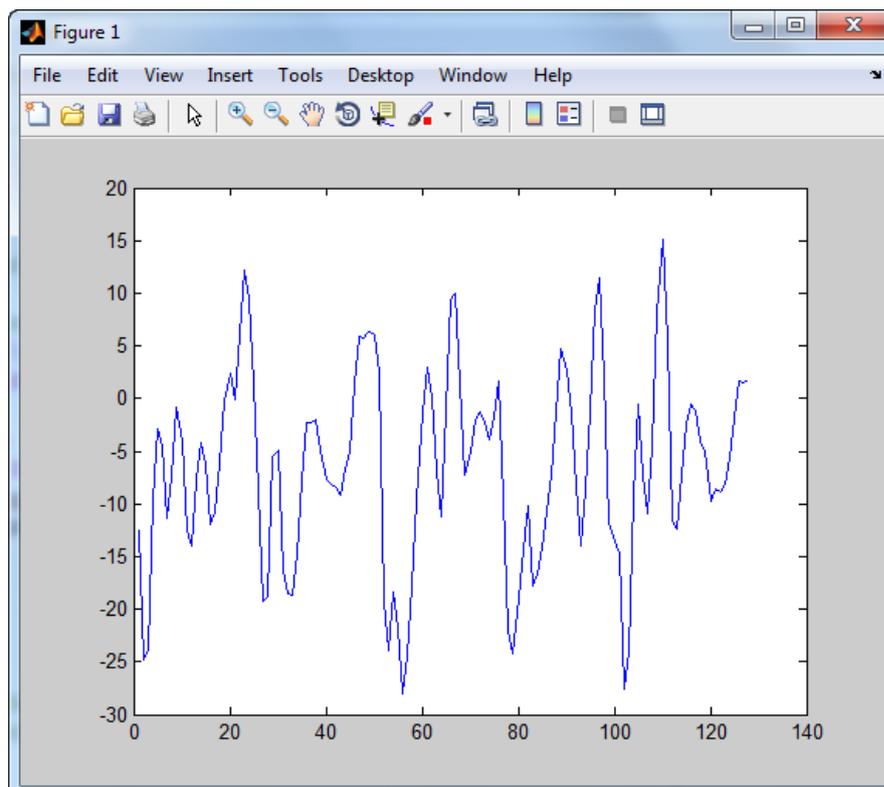


Figure 40: Plotting meanface

```

Covariance = Normalized'*Normalized;
[V D] = eig(Covariance);
L_eig_vec = [];
for i = 1 : Train_Number
    if( D(i,i)>1 )
        L_eig_vec = [L_eig_vec V(:,i)];
    end
end
Eigenfaces = Normalized * L_eig_vec;
End

```

Table 7: Eigen faces Sample Data

K1	K2	K3	K4	K4	K5	K6	K7
-1.3773	1.7814	10.2752	-0.9369	6.6636	5.9264	-2.2324	4.9497
2.0403	3.1271	0.1466	13.6260	2.7448	8.6192	-35.3718	-13.1052
-2.9179	-15.0488	-9.4719	10.5665	9.4731	14.2242	-15.4086	-14.3847
-2.0888	-6.5820	-5.9247	3.7711	4.8897	5.3237	-12.9458	-7.1791
-0.7942	12.9155	12.2231	-6.0959	10.4246	2.4925	-5.9339	-5.0370
0.8973	22.1404	14.9183	-10.5349	14.7336	10.1029	5.2827	-12.8930
2.7004	11.8294	8.9028	-8.7426	15.3949	16.9744	-7.0238	-17.4409
5.0747	-3.1293	-0.1442	-4.6058	6.2863	4.1394	-3.7814	-3.6140
4.1736	4.2833	2.4770	-1.8375	11.2946	8.1459	11.5337	6.5294
-0.0334	-0.8527	6.7675	-1.5634	8.7921	2.1087	3.5226	4.9111
3.3171	0.9713	9.0089	6.6699	3.5427	2.6375	-4.2342	-2.0152
2.7752	4.2688	10.5191	10.6187	-2.4923	3.5875	-6.2563	-0.7750
-0.7806	3.3889	5.0083	8.4913	-2.6037	-1.2113	2.2561	2.9372
1.0203	-1.9619	2.4420	6.3099	2.9439	4.0171	0.7903	2.0943
2.7738	0.2092	3.8325	10.8952	7.6880	4.0556	2.1659	4.8083
4.2786	-4.5933	13.3930	14.0517	4.6123	-3.8128	14.7032	5.8303
1.8256	1.4225	9.4216	14.5666	0.4545	-6.9069	17.5821	4.3885
-4.0380	-0.2376	4.5950	12.6184	-3.4250	-6.3039	21.7217	11.0853

1.8616	5.8097	-9.0016	10.6675	-0.7730	7.1543	-12.5970	-4.6836
3.4648	4.2932	-5.9620	12.9521	5.4482	12.2005	-45.4885	-20.7227
0.0443	-2.5916	4.6572	13.6679	-5.6275	-1.5707	-23.0413	-2.8167
-1.0229	2.8570	0.5619	7.5591	-14.9049	-11.5970	11.5777	15.0727
-0.0638	1.5991	-2.8416	9.5748	-13.5420	-11.3750	22.0719	21.5583
3.8771	0.6430	-4.2129	22.9296	-3.3713	5.5196	10.2023	23.3229
-4.9456	-2.3515	2.6561	1.5897	-19.4370	-10.4268	-30.4091	-1.7581
0.8547	0.6966	2.9871	4.1286	-1.1071	1.1328	-19.6663	-4.0912
2.1479	1.0558	-5.2422	21.3135	-5.3593	-1.0031	42.5428	22.9298
7.3585	-0.5494	-0.2823	12.5855	-13.6829	-2.2751	1.7358	2.6519
-6.0629	-3.1353	1.8392	3.6929	-8.9409	-1.3865	-58.490	-25.1036
-9.3156	-2.1158	2.2985	10.8871	6.6613	10.0452	14.3090	13.1847
7.0276	3.3849	-10.116	-7.6563	-4.5044	-4.8993	-12.474	-12.105

Now after reaching this step, the K samples should be returned to their real dimensions by multiplying each sample with the original samples. To start working with the unknown sample to make it in the same characteristics of the K samples, we use the Euclidean distance function to measure the distance between them, as shown in the code below:

```
% in this function the test signal will be converted to facevector and compare
% it's distance with training database signals, the nearest one is the
% equivalent signal.
function OutputName = Recognition(TestSignal, meanface, Normalized, Eigenfaces)
ProjectedSignals = [];
Train_Number = size(Eigenfaces,2);
for i = 1 : Train_Number
    temp = Eigenfaces'*Normalized(:,i);
    ProjectedSignal = [ProjectedSignal temp];
end
% Below the test sample will be converted to facevector
InputSignal = load(TestSignal);
```

```

Difference = double(InputSignal)-meanface;
ProjectedTestSignal = Eigenfaces'*Difference;
% Below the distance between the test signal and the training Signals in the database
will be compared.

```

```

dist = [];
for i = 1 : Train_Number
    q = ProjectedSignal(:,i);
    temp = ( norm( ProjectedSignal - q ) )^2;
    dist = [dist temp];
end
% Below will return the equivalent signal to the example function to
% display it in the output window.
[Euc_dist_min , Recognized_index] = min(dist);
OutputName = strcat(int2str(Recognized_index));
end

```

Here, the code below will display the equivalent sample that has the smallest distance to the unknown sample.

```

% Below will display the tested sample with its equivalent one.
% SelectedSignal = strcat(TrainDatabasePath,'\',OutputName);
SelectedSignal = strcat(OutputName);
Output=str2double(Selectedsignal);
if(Output<=15)
disp('This signal is belong to first group');
end
if(Output>15 && Output<=30)
disp('This signal is belong to second group');
end
if(Output>30 && Output<=45)
disp('This signal is belong to third group');
end

```

The accuracy of this algorithm is 65%; however, it is possible to increase this ratio to 75% by increasing the number of training dataset samples since more unique features will be acquired from each disease. This will lead to more K Eigen faces in the comparison step. However, the most important action that needs to be taken in this algorithm is the number of diseases in the diseases matrix, as this will increase the probability of knowing more details about the unknown sample. Moreover, in the testing step, the running time should be considered; in this case, increase training dataset sample will cost us running time.

Table 8: PCA Algorithm Result

Unknown sample	Result
1 st sample	Correct
2 nd sample	Correct
3 rd sample	Correct
4 th sample	Wrong
5 th sample	correct
6 th sample	Wrong
7 th sample	Wrong
8 th sample	Correct
9 th sample	Wrong
10 th sample	correct

Correct samples ratio=6 correct/10 samples * 100% =60%

3.3.2. Naïve Bayes with discrete wavelet transform

From the above mother wavelets, it is found that there are different types close to the real ECG signal. This thesis deals with the multi-resolution wavelet transform (db4) from Daubechies to extract features from an electrocardiogram (ECG) signal since it is closer to the real ECG signals. These ECG signals were captured from the Lead II (MLII). In this step, a QRS complex was detected and from this complex, other peak

(P and T) detections will be easy. Also from this complex, the duration between two peaks such as P and R can be measured by subtracting the timestamp of the R peak from the timestamp of the P peak. The peaks and their timestamp can be downloaded from the MIT University database. The main idea in this thesis is to use the first filter coefficients to obtain the features and check whether the outputs are similar to the required dataset [15].

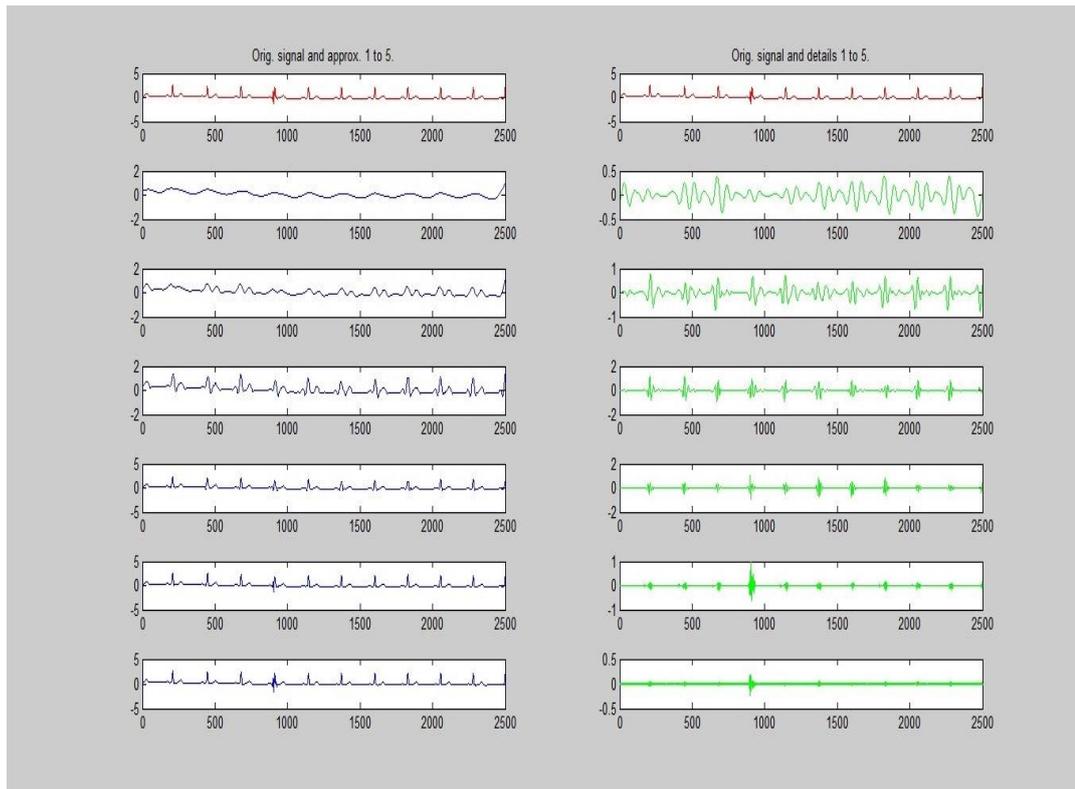


Figure 41: DWT filter

3.3.2.1. Detecting waves from signal

R peak detection

The peaks of R waves are clearer and have the largest amplitudes in MLII leads among other leads. In this case, the peak detection will be performed by keeping the largest value in each second. The peak of this wave will help to calculate the cardiac beats/minute (Hb/m) [19].

Q Peak detection

The Q wave is placed before the R wave and its shape is directed to the negative values. This wave has a peak, which is the smallest negative value of the wave. The detection procedure is carried out by:

1. measuring the duration between R peak and R-1 Peak
2. dividing this duration by 2
3. finding the smallest value between the second half of the duration and the R timestamp (this value will be considered to be the Q peak), and
4. saving the timestamp of the Q peak [19]

P Peak detection

The P wave is placed before the R wave and its shape is directed to the positive values. This wave has a peak, which is the largest positive value of the wave. The detection procedure is carried out by:

1. measuring the duration between the R peak and R-1 Peak
2. dividing this duration by 2
3. finding the largest value between the second half of the duration and the R timestamp (this value will be considered to be the P peak), and
4. saving the timestamp of the P peak

S Peak detection

The S wave is placed after the R wave and its shape is directed to the negative values. This wave has a peak, which is the smallest negative value of the wave. The detection procedure is carried out by:

1. measuring the duration between R peak and R+1 Peak
2. dividing this duration by 2
3. finding the smallest value between the R peak timestamp and the first half of the duration (this value will be considered to be the S peak), and
4. saving the timestamp of the S peak.

T Peak detection

The T wave is placed after the R wave and its shape is directed to the positive values. This wave has a peak, which is the smallest positive value of the wave. The detection procedure is carried out by:

1. measuring the duration between R peak and R+1 Peak
2. dividing this duration by 2
3. finding the largest value between the R peak timestamp and the first half of the duration (this value will be considered as the S peak), and saving the timestamp of the T peak

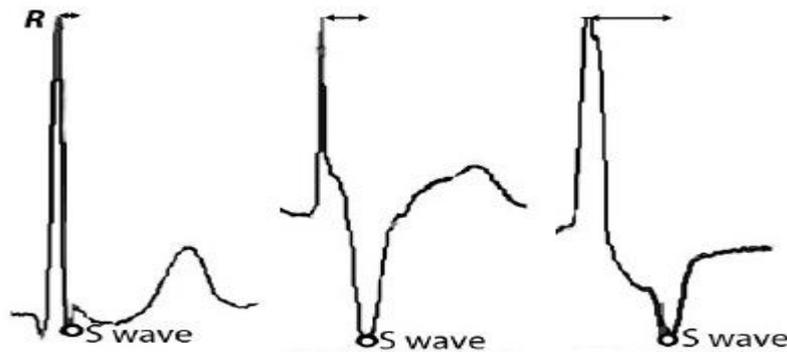


Figure 42: Waves peaks

Now after the detection step, it is necessary to extract the interval feature of the waves. The interval feature is measured by the steps below:

1. *The PR interval is measured by subtracting the timestamp between the P and R wave peaks.*
2. *The QRS interval is measured by subtracting the timestamp between the Q and S wave peaks.*
3. *The RT interval is measured by subtracting the timestamp between the R and T wave peaks.*

4. The RR interval is measured by subtracting the timestamp between the R and R+1 wave peaks.

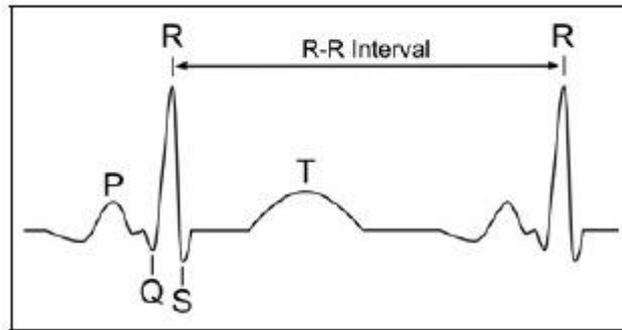


Figure 43: Typical ECG signal

After applying this filter to the signal and attaining the required features, these features of the vector of the unknown signal can be compared with the training dataset to determine to which class it belongs, i.e. in which disease class it is a member.

3.3.2.2. Applying Naïve Bayes Algorithm

After extracting the features from the ECG signal by using DWT filter, now it is important to classify these features according to their probability in the Naïve Bayes algorithm. As mentioned previously, this algorithm uses the formula below:

$$p(c/f_1, \dots, f_n) = p(c) = \frac{p(c)p(f_1, \dots, f_n / c)}{p(f_1, \dots, f_n)} \quad (2.1)$$

The steps of this algorithm formula are explained below.

- Created table composed of all required features

Table 9: DWT Extracted Features

Q_R_S_Interval	R_Peak	R_Position	P_Peak	Q_Min	Q_Min_Position	T_Max	T_Max_Position	P_Max_Position	R_R_Interval	P_R_Interval	S_Min	S_Min_Position	Class
19	2.648	209	0.392	0.099	439	0.771	269	416	240	207	0.093	218	D1
19	2.485	449	0.297	-0.00	671	0.668	507	646	232	197	0.002	458	D1
23	2.350	681	0.724	-1.271	905	0.541	740	900	232	219	-0.1	690	D1
28	1.751	913	0.103	-0.16	1126	0.445	973	1112	232	199	-0.34	928	D1
26	2.126	1145	0.099	-0.22	1363	0.380	1205	1337	227	192	-0.30	1154	D1
101	2.092	1372	0.099	-0.205	1509	0.379	1432	1568	229	196	-0.26	1389	D1
105	1.993	1601	0.099	-0.188	1733	0.345	1662	1792	228	191	-0.308	1610	D1
26	2.202	1829	0.099	-0.200	2047	0.337	1889	2021	227	192	-0.330	1838	D1
14	5.810	145	4.889	4.528	340	4.769	239	309	203	39	4.427	149	D2
14	6.095	348	4.929	4.490	531	4.821	436	500	189	37	4.549	354	D2
11	5.837	537	4.926	4.630	714	4.854	621	682	184	39	4.582	545	D2
13	5.920	721	4.916	4.480	904	4.833	808	876	189	34	4.499	725	D2
13	5.677	910	4.965	4.538	1092	4.878	996	1064	188	34	4.498	917	D2
11	5.706	1098	4.922	4.546	1291	4.867	1003	1253	199	44	4.467	1105	D2
14	5.884	1297	4.857	4.431	1495	4.805	1389	1457	204	44	4.575	1302	D2
14	5.670	1501	4.892	4.465	1691	4.805	1389	1661	196	36	4.462	1509	D2
14	5.836	1697	4.931	4.485	1883	4.828	1782	1852	193	38	4.556	1705	D2
13	5.594	1890	4.952	4.522	2072	4.872	1971	2043	188	35	4.461	1897	D2
13	5.654	2078	4.888	4.506	2264	4.872	1971	2229	194	43	4.463	2085	D2
13	0.800	77	0.201	-0.52	361	-0.24	208	309	294	62	-0.52	85	D3
15	0.893	371	-0.22	-0.55	655	-0.30	518	617	293	47	-0.46	472	D3
17	0.934	664	-0.25	-0.61	937	-0.30	781	885	284	63	-0.56	670	D3
15	0.860	948	-0.24	-0.54	1223	-0.28	1065	1170	284	62	-0.55	954	D3
15	0.818	1232	-0.21	-0.53	1507	-0.28	1369	1465	284	51	-0.50	1238	D3
16	0.870	1516	-0.21	-0.52	1801	-0.27	1653	1747	294	63	-0.51	1522	D3
17	0.855	1810	-0.22	-0.55	2036	-0.27	1653	1988	235	57	-0.58	1817	D3
20	0.889	2045	-0.25	-0.55	2393	-0.30	2057	2340	359	64	-0.53	2053	D3
17	0.87	2404	-0.24	-0.57	2697	-0.30	2185	2657	303	50	-0.53	2413	D3
15	0.779	2707	-0.21	-0.54	2990	-0.28	2834	2952	291	46	-0.62	2714	D3
15	0.777	2998	-0.18	-0.51	3275	-0.26	3119	3224	286	60	-0.60	3005	D3
13	0.800	77	-0.20	-0.52	361	-0.24	208	309	294	62	-0.52	85	D3
15	0.893	371	-0.22	-0.55	655	-0.30	518	617	293	47	-0.46	472	D3
17	0.934	664	-0.25	-0.61	937	-0.30	781	885	284	63	-0.56	670	D3

In this thesis, the intervals between the waves are scaled from their real values to integer values because the real values are very small and will complicate the calculations. However, this swap is not a random scale. Each sample capture is 0.003 seconds and they are scaled to be 1 second. Table 10 explains the meaning of this scaling further.

Table 10: Features time scale

Real sample time	Sample	Scaled Sample time	Sample
'0:00.000'	-0.145	1	-0.145
'0:00.003'	-0.145	2	-0.145
'0:00.006'	-0.145	3	-0.145
'0:00.009'	-0.145	4	-0.145
'0:00.012'	-0.145	5	-0.145
'0:00.015'	-0.145	6	-0.145
'0:00.018'	-0.145	7	-0.145

- These features can provide the behavior of this signal and be used in the classification method. In this thesis, the features with high priority are selected, leaving the others just to facilitate the calculations. Normally, it is preferable to select all of them.
- The R-R-Interval features are important in order to calculate the Beat Per Minute Feature (BPM) by multiplying R-R-Interval with 60, as shown in the formula below:

$$BPM = R_R_Interval * 60 \quad (3.6)$$

Table 11: High Priority Features

QRS_Interval	R_R_Interval	PR_Interval	BPM	P_Peak	Class
0.057	0.72	0.621	43	0.392159	D1
0.057	0.696	0.591	42	0.297756	D1
0.069	0.696	0.657	42	0.724164	D1
0.084	0.696	0.597	42	0.103059	D1
0.078	0.681	0.576	41	0.099224	D1
0.303	0.687	0.588	41	0.099224	D1
0.315	0.684	0.573	41	0.099224	D1
0.078	0.681	0.576	41	0.099224	D1
0.042	0.609	0.117	37	4.889554	D2
0.042	0.567	0.111	34	4.929549	D2

0.033	0.552	0.117	33	4.926978	D2
0.039	0.567	0.102	34	4.916373	D2
0.039	0.564	0.102	34	4.965166	D2
0.033	0.597	0.132	36	4.922369	D2
0.042	0.612	0.132	37	4.857442	D2
0.042	0.588	0.108	35	4.892432	D2
0.042	0.579	0.114	35	4.931812	D2
0.039	0.564	0.105	34	4.952628	D2
0.039	0.582	0.129	35	4.888242	D2
0.039	0.882	0.186	53	-0.20126	D3
0.045	0.879	0.141	53	-0.22731	D3
0.051	0.852	0.189	51	-0.25399	D3
0.045	0.852	0.186	51	-0.24461	D3
0.045	0.852	0.153	51	-0.21482	D3
0.048	0.882	0.189	53	-0.21033	D3
0.051	0.705	0.171	42	-0.22188	D3
0.06	1.077	0.192	65	-0.25432	D3
0.051	0.909	0.15	55	-0.2478	D3
0.045	0.873	0.138	52	-0.2135	D3
0.045	0.858	0.18	51	-0.18509	D3

- After this selection, the features should be compared with the threshold values using those new values in the calculations. Table. 12 will explain the threshold values for each feature:

Table 12: Features threshold

Feature	Threshold
BPM	Normal (60 ~ 100)
P wave	Normal (Upright and Uniform)
PR interval	Normal (0.12 ~ 0.2 sec)
QRS interval	Normal (0.06 ~ 0.1 sec)

- In this step, the conversion process is simple. By comparing each value with its threshold parameter and converting it to its suitable value, the issue is the last one before classification, as shown in Table 13

Table 13: Features final shape

QRS_Interval	PR_Interval	BPM	P_Peak	Class
S	L	S	Pe	D1
S	L	S	Pe	D1
N	L	S	Pe	D1
N	L	S	Pe	D1
N	L	S	Pe	D1
S	L	S	Pe	D1
S	L	S	Pe	D1
N	L	S	Pe	D1
S	S	S	Pe	D2
S	S	S	Pe	D2
S	S	S	Pe	D2
S	S	S	Pe	D2
S	S	S	Pe	D2
S	S	S	Pe	D2
S	S	S	Pe	D2
S	S	S	Pe	D2
S	S	S	Pe	D2
S	S	S	Pe	D2
S	N	S	Pe	D2
S	N	S	Pe	D2
S	S	S	Pe	D2
S	S	S	Pe	D2
S	S	S	Pe	D2
S	N	S	Pe	D2
S	N	S	Ne	D3
S	N	S	Ne	D3
S	N	S	Ne	D3
S	N	S	Ne	D3
S	N	S	Ne	D3
S	N	S	Ne	D3

S	N	S	Ne	D3
N	N	N	Ne	D3
S	N	S	Ne	D3
S	N	S	Ne	D3
S	N	S	Ne	D3

Table 14: Samples explanation

Attribute	Explanation
S	Smaller
N	Normal
L	Larger
Pe	Positive
Ne	Negative

- Start using the classification formula by considering all features above in the calculation, as mentioned previously. It is preferable to use all features in Table 9. However, in order to simplify the calculation, some features are ignored and use only those features with the higher priority.

Another important point is that the number of diseases and the increasing of samples in the dataset will increase the accuracy of finding the real matching of the unknown signal.

- The first step in the calculation is to find the ratio of the classes in the dataset, as shown below.

$$P(class : D1) = \frac{8}{30} = 0.266 \quad (3.7)$$

$$P(class : D2) = \frac{11}{30} = 0.366 \quad (3.8)$$

$$P(class : D3) = \frac{11}{30} = 0.366 \quad (3.9)$$

- The second step of the calculation is to find the percentage of each attribute value from each class, starting with the QRS attribute. It is composed of two values (S & N).

$$P(S : D1) = \frac{4}{8} = 0.5 \quad (3.10)$$

$$P(N : D1) = \frac{4}{8} = 0.5 \quad (3.11)$$

$$P(S : D2) = \frac{11}{11} = 1 \quad (3.12)$$

$$P(S : D3) = \frac{10}{11} = 0.909 \quad (3.13)$$

$$P(N : D3) = \frac{1}{11} = 0.09 \quad (3.14)$$

We maintain the calculation of all other values in all attributes (PR-Interval, PBM and P-Peak) to find the ratio for all of them. The result of the total calculation is shown in Table 15:

Table 15: Attributes values

QRS_Interval		
P(S:D1)=0.5	P(N:D1)=0.5	
P(S:D2)=1	P(N:D2)=0	
P(S:D3)=0.909	P(N:D3)=0.09	
PR-Interval		
P(L:D1)=1	P(N:D1)=0	P(S:D1)=0
P(L:D2)=0	P(N:D2)=0.272	P(S:D2)=0.727
P(L:D3)=0	P(N:D3)=1	P(S:D3)=0
BPM		
P(S:D1)=1	P(N:D1)=0	
P(S:D2)=1	P(N:D2)=0	
P(S:D3)=0.909	P(N:D3)=0.09	
P-Peak		

$P(\text{Pe:D1})=1$	$P(\text{Ne:D1})=0$
$P(\text{Pe:D2})=1$	$P(\text{Ne:D2})=0$
$P(\text{Pe:D3})=0$	$P(\text{Ne:D3})=1$

- Now, after finding the probability of each value for all the attributes, the final part of the theorem is to find the appropriate class in the unknown sample. The unknown sample must have values for all the required attributes. In this thesis, the attributes are only (QRS, PR-Interval, PBM and P_Peak). Below is an example of the unknown sample.

Example: 1st Sample (QRS=S, PR_Interval=N, PBM=S, P_Peak=Ne)

By finding the probability of this sample to all classes to attain the most suitable one, we use the formula below and compare the results.

$$P(\text{Test : Dx}) = P(\text{QRS} = \text{S} : \text{Dx}) * P(\text{PR_Interval} = \text{L} : \text{Dx}) * P(\text{PBM} = \text{N} : \text{Dx}) * P(\text{P_Peak} = \text{Ne} : \text{Dx}) \quad (3.15)$$

Dx mean (D1, D2 and D3), calculating the results for three classes and using selecting the highest value among them.

$$P(\text{Test: D1}) = 0.5 * 0 * 1 * 0 = 0$$

$$P(\text{Test: D2}) = 1 * 0.272 * 1 * 0 = 0$$

$$P(\text{Test: D3}) = 0.909 * 1 * 0.909 * 1 = 0.826$$

- After calculating the probability of all the attributes, now it is important to know the ratio of the three classes by using the formula below.

$$P(\text{Dx : Test}) = P(\text{Dx}) * P(\text{Test : Dx}) \quad (3.16)$$

The result of the above formula is shown in Table 16.

Table 16: Sample classes probability

Class name	Result
D1	0
D2	0
D3	0.302

The result of the above probability shows that the tested signal is closer to the class D3 for its higher probability.

CHAPTER 4

CONCLUSION AND FUTURE WORK

4.1. Conclusion

From the above thesis, it is discovered that patient movement will effect signal shape and add noise to the signal. This is known as muscle noise. It is important to be careful where to put the electrodes on a patient's body. The microcontroller analogue to digital part performed well and yielded a very good resolution when compared with other types of analogue-to-digital circuits such as those used in the literature survey [1]. The PCA algorithm is too systematic an algorithm and needs some specific requirements to work, such as the size of all signals in the data stream; the unknown signal should also be the same. Here, it uses 128 rows for each sample if one of these samples differs in size and if the algorithm fails to work. In addition, this algorithm can increase its efficiency by using a DWT filter instead of a mean signal; however, the Naïve Bayes algorithm is more efficient due to the fact that it deals with the features that are fetched from the DWT filter

4.2. Future Work

In this thesis, the hardware that was used is very specific and can work in different environments. However, it is severe from the mMicrocontroller size so it is preferable to use a smaller type such as a Nanoduino kit. This device has specifications such that it can process the signal and convert it from analogue to digital at the same sampling frequency. The only limit of this kit is the communication ports with the computer and its power supply from the USB connection only. Other improvements can also be carried out on this hardware by using other microcontrollers with 16 digital bits to increase the accuracy of the digital signal. The wireless part can be improved further by using a GPRS shield

instead of the XBee module to send the data stream directly to the clinic care center so as to avoid saving the data stream with the patient's computer and then sending it to the clinic care server. Another method exploits a WIFI module as a shield on the top of the UNO R3 ket to connect the ket directly to the Internet without any need for personal computer storage. This part will grant an opportunity to use the system for live monitoring and diagnoses as well as grant the patient the facility to move without taking any consideration how this affects the data transfer to the server. The Database and Client Design part can be improved by providing Web access to the server to enable specialists to check patients remotely from any place in the world instead of being connected to the OMC of the clinical care or using the software in offline mode. The final part of this thesis can be improved by providing more diseases to the dataset in order to increase the number of classes and increase the likelihood of correct diagnosis. Moreover, this part can be improved by using more classification methods and making intersections between result classes to increase the accuracy of the classification.

REFERENCES

1. <http://medical-dictionary.thefreedictionary.com/Electro-Cardio+Gram>.
2. **Tamas H., (2009)**, “*Wireless ECG/EEG with the MSP430 Microcontroller*”, MASTER THESIS, Department of Electronics and Electrical Engineering, University of Glasgow, England , pp.20- 50.
3. **Mahmoodabadi S. Z., Ahmadian A., Abolhasani M. D., (2005)**, “*ECG Feature Extraction Using Daubechies Wavelets*”, the 5th IASTED International Conference on Visualization, Imaging and Image Processing, pp. 343-348.
4. **Castro B., Kogan D., Geva A. B., (2000)**, “*ECG Feature Extraction Using Optimal Mother Wavelet*”, Electrical and Electronic Engineers, IEEE 1st Convention of the. IEEE, pp. 346-350.
5. **Espiritu A., Carbajal C., (2010)**, “*ECG Feature Extraction Via Waveform Segmentation*”, IEEE, In Electrical Engineering Computing Science and Automatic Control, 7th International Conference, pp. 250-255.
6. **RajendraG., Kothari A., Keskar A., (2012)**, “*ECG Feature Extraction Using LCAD*”, International Conference on Communication Systems and Network Technologies, pp.

7. **Atmel Corporation, (2009)**, “*ATmega48PA/88PA/168PA/328P*”, Data Sheet, pp. 170-230.
8. **Premeaux E., Evans B., (2011)**, “*Arduino Projects to Save the World*”, 1st Edition, Apress, New York, pp. 116-120.
9. **Margolis M., (2011)**, “*Arduino Cookbook*”, O’Reily Media, 1st Edition, United States of America, pp. 165-180.
10. **Wilcher D., (2012)**, “*Learn Electronics with Arduino*”, 2nd Edition, Paul Manning, New York, pp. 77-82.
11. **Riley M., (2012)**, “*Programming Your Home. Automate with Arduino, Android, and Your Computer*”, Pragmatic Bookshelf, 3rd Edition, Texas, pp. 106-114.
12. **Oxer J., Blemings H., (2009)**, “*Practical Arduino: Cool Projects for Open Source Hardware*”, 1st Edition, Apress, New York, pp. 220-222.
13. **Kim K., (1996)**, “*Face Recognition Using Principle Component Analysis*”, In International Conference on Computer Vision and Pattern Recognition, pp. 586-591.
14. **Alsarori F., (2013)**, “*Automatic Detection of Breast Cancer in Mammogram Images*”, MASTER THESIS, Department of Computer Engineering, Çankaya University, Turkey, pp. 37-46.
15. **Han J., Kamber M., (2006)**, “*Data Mining Concepts and Techniques*”, 2nd Edition, Elsevier, San Francisco, pp. 250-270.
16. **Chi Y., Sawaguchi Y., (2011)**, “*A Mobile-Phone-Based Health Management System*”, National United University, Taiwan, pp. 21-39.

17. **Priyadarshee S., (2007)**, “*A Health Decision Support System for Rural India*”, MASTER THESIS, Department of Computer Engineering, National Institute of Technology, Rourkela, India, pp. 22-27.
18. **Malaekheh S. R., (2013)**, “*Clinical Decision Support System for Chronic Pain Management in Primary Care: Usability Testing*”, MASTER THESIS, Department of Information Technology, McMaster University, Ontario, Canada, pp. 22-30.
19. **Swedberg K., Cleland J., Dargie H., Drexler H., Follath F., (2005)**, “*Guidelines for the Diagnosis and Treatment of Chronic Heart Failure*”, European Heart Journal, vol. 26, no. 11, pp. 1115-1140.
20. **Wang J., (2011)**, “*A Service-Oriented Architecture for Integrating Clinical Decision Support in a National E-Health System*”, MASTER THESIS, Department of Computer Engineering, Lund University , Stockholm, Sweden, pp. 8-17.
21. **Billings J. R., (2005)**, “*What Do We Mean by Integrated Care?*”, Journal of Integrated Care, vol. 13, no. 5, pp. 13-20.
22. **Lewis J. R., (1995)**, “*IBM Computer Usability Satisfaction, Questionnaires: Psychometric Evaluation and Instructions for Use*”, International Journal of Human-Computer Interaction, vol.7, no. 1, pp. 57-78.
23. **Dingfei G., (2012)**, “*Study of ECG Feature Extraction for Automatic Classification Based on Wavelet Transform*”, Computer Science & Education (ICCSE), IEEE 7th International Conference, pp. 500-503.

24. **Srinivasan N., Krishnan S. M., (2002)**, “*Cardiac Arrhythmia Classification Using Autoregressive Modeling*”, Department of Biomedical Engineering, Research Centre, Nanyang Technological University, Singapore, pp. 10-13.
25. **Farrokhi F., Moradi M. H., Miri R., (2004)**, “*Automatic. Detection of Premature Complexes in ECG Using Wavelet Features and Fuzzy Hybrid Neural Network*”, Iranian Journal of Electrical and Computer Engineering, vol. 3, no .2, pp. 5-7.
26. **Li C., Zheng C., Tai C., (1995)**, “*Detection of ECG Characteristic Points Using Wavelet Transforms*” Department of Biomedical Engineering, IEEE, pp. 3-4.
27. **Mar T., Zauneder S., Martinez J. P., Llamedo M., Poll R., (2011)**, “*Optimization of ECG Classification by Means of Feature Selection*”, Biomedical Engineering, IEEE Transactions, vol. 58, no. 8, pp. 2168-2177.
28. **Nandi R. J., Nandi A. K., Rangayyan R. M., Scutt D., (2006)**, “*Classification of Breast Masses in Mammograms Using Genetic Programming and Feature Selection*”, Medical and Biological engineering and computing, vol. 44, no. 8, pp. 683-694.
29. **Yuan Y., Giger M. L., Li H., Sennett C., (2008)**, “*Correlative Feature Analysis of FFDM Images*”, International Society for Optics and Photonics in Medical Imaging, pp. 615-622.
30. **Isa M., Ashidi N., Ting S. S., (2012)**, “*Automatic Segmentation and Detection of Mass in Digital Mammograms*”, The 11th International Conference on Signal Processing, Malaysia, pp. 143-146.

APPENDICES A

CURRICULUM VITAE



PERSONAL INFORMATION

Surname, Name: Majeed, Samer Hameed

Date and Place of Birth: 16 July 1982, Baghdad

Marital Status: Married

Phone: +9647704506021

Email: samer1607@yahoo.com

EDUCATION

Degree	Institution	Year of Graduation
M.Sc.	Çankaya Univ., Computer Engineering	2014
B.Sc.	Technology Univ., Computer Engineering	2005
High School	Elegance High School	2000

WORK EXPERIENCE

Year	Place	Enrollment
2009- Present	Itisaluna Telecom Company.	Network Switch Manager
2005-2009	Huawei Telecom.	Network Switch Engineer

FOREIN LANGUAGES

Advanced English, Beginner Turkish