AEGIS - Body Area Network to Assist Congestive Heart Failure Patients

By

Haroon Ali Akbar *NUST201200848BSEECS60412F*

Jamal Ahmed Khan *NUST201200745BSEECS60412F*

Usama Perviaz *NUST201200290BSEECS60412F*

A Project report submitted in partial fulfillment of the requirement for the degree of

Bachelors in Electrical (Electronics) Engineering

Department of Electrical Engineering

School of Electrical Engineering & Computer Science

National University of Sciences & Technology

Islamabad, Pakistan

2016

CERTIFICATE

It is certified that the contents and form of thesis entitled **"AEGIS-Body Area Networks to Assist Congestive Heart Failure Patients"** submitted by *Haroon Ali Akbar* **(***NUST201200848BSEECS60412F***),** *Jamal Ahmad Khan* **(***NUST201200745BSEECS60412F***)** and *Usama Pervaiz* **(***NUST201200290BSEECS60412F***),** has been found satisfactory for the requirement of the degree.

Advisor: ______________________________

(Dr Osman Hasan)

Co-Advisor: ______________________________

(Dr Khawar Khurshid)

DEDICATION

To Allah the Almighty

&

To our Parents and Faculty

ACKNOWLEDGEMENT

We are deeply thankful to our Advisor and Co-Advisor, Dr. Osman Hassan, and Dr. Khawar Khurshid respectively, for helping us throughout the course in accomplishing our final year project. Their guidance, support and motivation enabled us in achieving the objectives of the project.

We are also thankful to the Dr. Azhar Rafiq (MD NASA) for their support, the lab engineers in SEECS labs, for cooperating with us during the course of this FYP.

Table of Contents

ABSTRACT ……………………………………………………………………………..8

INTRODUCTION...10

1.1.1 DOMAIN…………………………………………………………………...10
1.1.2 HEART FAILURE…………………………………………………………..11
1.1.3 CONGESTIVE HEART DISEASE IN PAKISTAN……………………….11
1.1.4 FACTS…….……………………………………………………………...12
1.1.5 PROBLEM STATEMENT………………………………………………..13
1.1.6 MOTIVATION AND NEED……………….…………………………..13
1.1.7 PROJECT DESCRIPTION……………………………………………..14
1.1.8 SCOPE OF PROJECT…………………………………………………...14
1.1.9 PROJECT PLAN…………………………………………………………16
1.1.9.1 HARDWARE ……………………………………………………………16
1.1.9.2 DATA PROCESSING AND CLOUD STORAGE…………………17
1.1.10 END PRODUCT DESCRIPTION……………………………………………………………18
1.1.10.1 HARDWARE. ………………………………………………………………..18
1.1.10.2 SOFTWARE……………………………………………………………………….18

LITERATURE REVIEW 20

2.1.1 INTRODUCTION………………………………………………………………20

2.1.1.1 QARDIOCORE…………………………………………………………...20

2.1.1.2 QARDIOARM…………………..……………………………………….20

2.1.1.2 VITALITI…………………………………………………………………21

2.1.1.3 BLOOD PRESSURE COMPANION FREE……………………..……..21

2.1.1.4 HEALTHPATCH MD…………………………………………………….23

2.1.1.5 HEART RATE PLUS……………………………………………………….23

2.1.1.6 GRACE 2.0 ACS CALCULATOR……………………………………….24

2.1.1.7 JBS3 RISK CALCULATOR…………………………………………....24

2.1.1.7 ASCVD RISK ESTIMATOR……………………………………………..25

2.1.1.9 CARDIOGRAPH ………………………………………………….25

2.1.2 WEAKNESS OF PREVIOUS SOLUTIONS…………..…………………….……..25

2.1.3 SURVEY TAKEAWAY…………………………………………………………..26

2.2 COMPARISON…………………………………………………………26

FUNCTIONALITY AND DESIGN..28

3.1 ACCELEROMETER …………………………………………………………………….29

3.1.1 FEATURES …………………………………………………………….30

3.2 ECG SENSING CIRCUITR ……………………………………………………..31

3.2.1 FEATURE…………………………………………………………….32

3.2.2 APPLICATIONS …………………………………………………..33

IMPLEMENTATION……………………………………………………………...34

4.1 HARDWARE COMPONENTS ……………………………………………..34

4.1.1 CONTROLLER: RF DIGITAL RFD22301……………………..34

4.1.2 ANALOGUE FRONT END: ANALOG DEVICES AD8232…….36

4.1.3 DIGITAL ACCELEROMETER: ANALOG DESIGN ADXL 345….37

4.1.4 VOLTAGE REGULATOR: STMMICROELECTRONNICS LF33…38

4.2 CIRCUIT CONFIGURATION AND IMPLEMENTATION …………………..39

4.2.1 ANALOGUE FRONT END AD8232 ……………………………..39

4.2.2 DIGITAL ACCELEROMETER ADXL 345 ………………………42

4.2.3 OVERALL CIRCUIT ……………...………………………..43

4.2.4 PCB …………………………………………………………44

4.3 MICROCONTROLLER CODING IDE SETUP ………………………………46

4.3.1 SETTING UP THE IDE …………………………………………… 46

4.3.2 MICROCONTROLLER CODE LOGIC ………………………………48

4.4 SIGNAL PROCESSING AND ALGORITH DESIGN …………………………52

DISCUSSION………………………………………………………………………80

CONCLUSION ……………………………….……………………………………81

FUTURE RECOMMENDATION……………………………………………………83

REFERENCES ……………………………………………………………………84

Table of Figures

Figure 1 – Coronary Heart Patients in Pakistan 12

Figure 2 – Doctor Statistics in Pakistan.. 15

Figure 3- Wearable Device Concept …... 18

Figure 4– Available Solution Comparison.. 27

Figure 5 – 3-Axis Digital Accelerometer ADXL 345/346................................30

Figure 6 – Functional Block Diagram of Accelerometer ADXL 345/346…….31

Figure 7 – Breakout Board AD8232……………………………………………..34

Figure – 8 Controller: RF Digital RFD22301…………………………………35

Figure 9 - nRF51822 BLE chip core……………………………………………..36

Figure 10 – AD8232 circuit breakdown………………………………………..37

Figure 11 – ADXL 345 Configuration…………………………………………38

Figure 12 – LF33 Voltage Regulator Configuration………………………...…39

Figure 13 – AD8232 Breakout Board Configuration…………………………. 41

Figure 14 – AD8232 Frequency Response……………………………………..42

Figure 15 – AD8232……………………………………………………………42

Figure 16 – ADXL345 ..………………………..………………………... 43

Figure 17 – Overall Configuration……….……………………….……….43

Figure 18 – PCB Design 1…………………………………………………….45

Figure 19 – PCB Design 2…………………………………………………….46

Figure 20 – RFDuino Library Installation……………………………………...48

Figure 21 – RFDuino USB Driver Installation ……....………………..……...49

Figure 22 – ECG Signal Breakdown…………………………………………….53

Figure 23 – Signal Conditioning and Detection…………………………………54

Figure 24 – QRS Complex ………………..…………………………………….56

Figure 25 – Pan Tompkins Results (a,b,c,d,e,f,g) ………………………………64

Figure 26 – ECG Evaluation Results of Pan Tompkins…………………………65

Figure 27 – Accelerometer Axes ………………………………………………..66

Figure 28 – Activity Detection Classifier Performance………………………….69

Figure 29 – Activity Detection Classifier Confusion Matrices (a,b,c)………….71

Figure 30 – Activity Detection Classifier Confusion Matrices(d,e,f)…………...75

ABSTRACT

Pakistan is developing country with a struggling economy where basic healthcare and medical facilities take the least priority. According to WHO report as of 2014 coronary heart disease has been ranked as the first cause for deaths in Pakistan, the number of deaths reached 111, 376 that is 9.87% of total deaths. With limited facilities and resources available the ratio of number of cardiac patients to number of doctors available is dangerously rising every day. Due to geographical location this death toll rises further during summer when national newspapers report heightened death toll each year.

In order to cope up with the problem and meet the growing needs of medical attention to the ever increasing number of patients in the country, large reforms are needed on a grass root level. That being said, the solution requires a huge amount of capital to be invested in the health sector but given the terrorism struck economy of it seems almost impossible for the government to implement such proposed solutions. Hence to bridge the gap between patients we're introducing the concept of eHealth and remote monitoring for Cardiac patients for the first time in Pakistan.

The age adjusted death rate of Pakistan is 110.65 per 100,000 of population that is an alarming figure with the leading cause being coronary heart diseases it is the need of the hour to build upon a solution to work towards the solution of minimizing this figure. Hence we at **AEGIS – Body Area Networks to Assist Cardiac Patients** are providing a cost effective solution in form of a wireless body area network based wearable, remote monitoring device.

The aim of our product is to:

1. Help the cardiac patients to remotely connect with their doctors so the latter can closely monitor their patient's vital signs and provide a timely feedback/reaction.

2. Provide the caretaker of the patient with all the useful information about their loved ones health so an upcoming unfortunate event can be taken care of.

Chapter 1

Introduction

This chapter gives an overview of our project and explains the motivation, scope, problem statement and project domain.

1.1.1 DOMAIN

Body Area Networks (BAN), eHealth and Internet of Things (IoT)

***"Body area network (BAN)**, also referred to as a wireless body area network (WBAN) or a body sensor network (BSN), is a wireless network of wearable computing devices. BAN devices may be embedded inside the body, implants, may be surface-mounted on the body in a fixed position wearable technology or may be accompanied devices which humans can carry in different positions, in clothes pockets, by hand or in various bags. [1]*

*"**E-health** is the transfer of health resources and health care by electronic means. It encompasses three main areas:*

1. The delivery of health information, for health professionals and health consumers, through the Internet and telecommunications.

2. Using the power of IT and e-commerce to improve public health services, e.g. through the education and training of health workers.

3. The use of e-commerce and e-business practices in health systems management.

E-health provides a new method for using health resources - such as information, money, and medicines - and in time should help to improve efficient use of these resources." [2]

*"The **Internet of Things (IoT**) is the network of physical objects—devices, vehicles, buildings and other items—embedded with electronics, software, sensors, and network connectivity that enables these objects to collect and exchange data. The*

IoT allows objects to be sensed and controlled remotely across existing network infrastructure."[3]

1.1.2 HEART FAILURE

Heart failure is a condition in which heart cannot pump enough blood to meet the body's need in other cases heart cannot pump blood with enough force. It is the number one cause of worldwide and according to WHO the leading cause of death in Pakistan. Heart failure does not refer to malfunction of heart however a condition that requires immediate medical assistance.

Heart failure is developed gradually as the pumping of blood grows weaker, it affects the right side of the heart or both, the latter in most cases. It leads to 17 million hospitalizations every year. Only in Pakistan alone it contributes to 25% to 30% deaths. Both adults and children alike are prone to this condition. There are 26 million heart patients, and annually 313 billion US dollars are spent on it.

1.1.3 CONGESTIVE HEART DISEASE IN PAKISTAN

Pakistani population ranks at one of the highest on risks of coronary heart disease in the world. The contributing statistics are 30 to 40% due to cardiovascular diseases(CVD). The death toll has reached to 200,000 per annum that is 410/100,1000 of the population. Adults and children are equally vulnerable to CVDs , the risk for which can occur even before birth and progresses due to unhealthy diets and lifestyle.

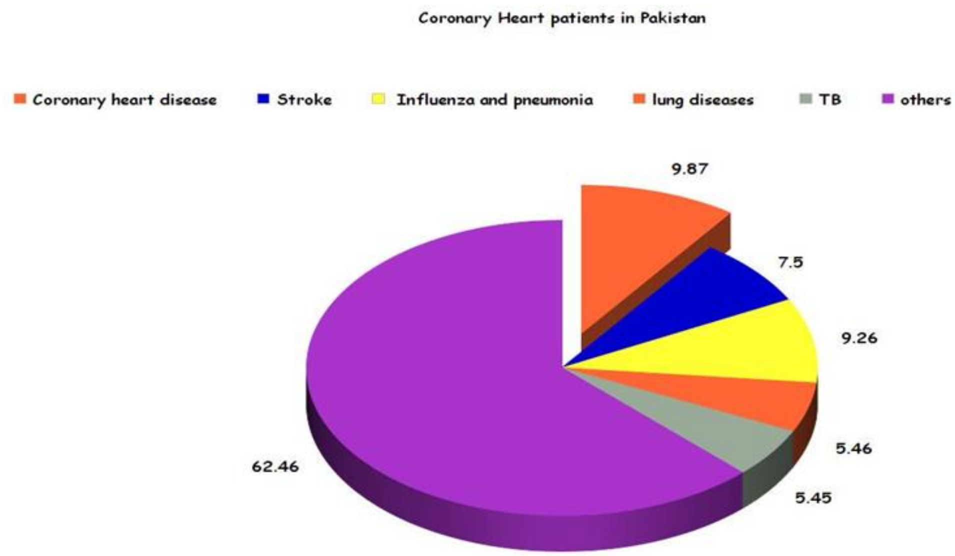


Figure 1 – Coronary Heart Patients in Pakistan

1.1.4 FACTS

- 33% of Pakistani population above the age of 45 has hypertension
- Prevalence of hypertension is 19% in people of age 15 or above
- Pakistan ranks at number six in terms of number of people with diabetes worldwide.
- Diabetic patients will rise to 13.9 million by 2020, leading Pakistan to 4th most populous country.
- 25% of patients over the age of 45 years suffer Diabetes Mellitus.
- Prevalence of smoking has been reported to be 14-21% in adolescents and adults, being more common in men.
- The estimated annual incidences of stroke in Pakistan are 250/100,000, translating to 350,000 new cases every year.
- Every 9th Pakistani is suffering from Liver Diseases. 300,000 patients have dire need of Liver Transplant.

1.1.5 PROBLEM STATEMENT

Congestive heart failure and cardiac diseases are based on a history of symptoms and is confirmed by ECG, a patient would need constant monitoring of his vitals and timely action to save his or her life. Often there is a huge chasm between patients, doctors and the patient's family. Especially in Pakistan the vast majority has limited access to health care facilities and a fast pace lifestyle and the slight callousness results in a loss of life. People with coronary heart diseases need a significant change in lifestyle and dietary changes, with this change they would need to monitor their body's reaction and constantly keep track of them.

To sum it up there is a dire need to bring about a single platform, which is easily accessible to the vast majority of the patients. That would help them monitor themselves, help their family keep track of their progress and also have the doctors on board to act timely.

1.1.6 MOTIVATION AND NEED

1- Remote monitoring of Cardiac Patients

With increasing number of cardiac patients each year and the constant number of available medical resources the need of a remote monitoring solution was more than ever.

2- Cost effective solution

Pakistan is a developing country with a per capita income of a little over than $110 per month. In such case a cost effective solution is the only viable solution.

3- Community development

Cardiac diseases are difficult to deal with ergo a community development feature for people with such conditions helps them in sharing hence growing in terms of social awareness and courage.

1.1.7 PROJECT DESCRIPTION

This platform is specifically designed for congestive heart patients and monitors the vitals necessary for them, it deals with cardiac rehabilitation and fitness monitoring. It continuously records your electrocardiogram, heart rate, heart rate variability, body temperature, body posture, activity including steps respiratory rate and stress levels. The additional diet tracking feature helps in fitness tracking and lifestyle adjustment during recovery period which is essential for recovering patients.

Our solution takes in data from the user from our sensors processes it and converts it into user understandable data, by analyzing the data, constantly monitoring the vitals and giving notifications under serious conditions. We have doctors on board as well as the patient's family or friends or people at work as caretakers, with having doctors on board they are timely notified about the vitals if there is any severity.

1.1.8 SCOPE OF PROJECT

According to the statistics by WHO about 9.87% of patients are due to coronary heart disease, with such an alarming percentage and lack of apt facilities it is the need of the hour to build a platform that would serve the purpose. Our app would target 10% of the population affected by heart diseases and provide timely assistance to them to stop the figure contributing to the death rate of Pakistan.

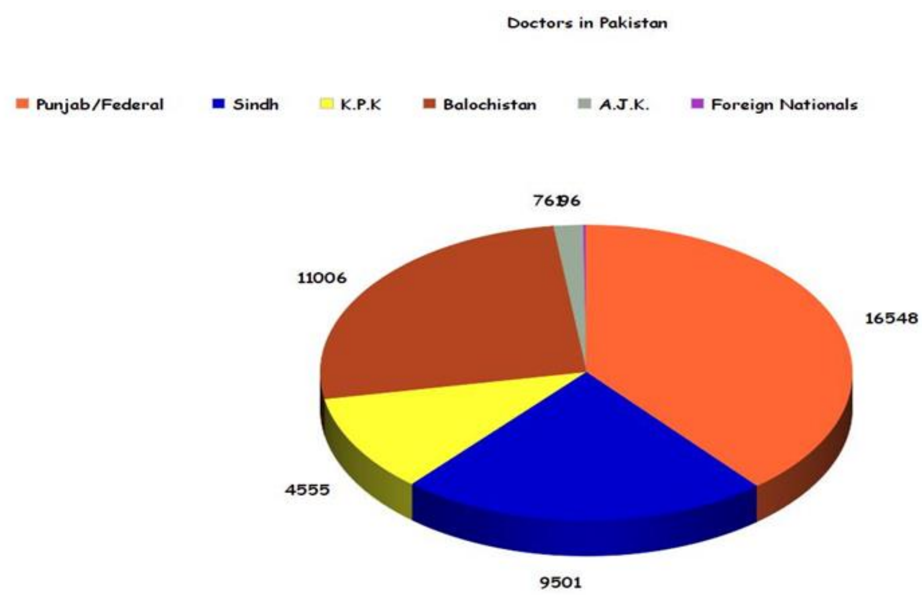


Figure 2 – Doctor Statistics in Pakistan

Apart from patients doctors would also benefit from a platform that allows them to remotely provide assistance, get timely notified in case of any emergency to be able to save lives. Doctors greatly benefit from having patient history, dietary records, tracking patient's activity and remotely gaining ecg graphs of the patients. All of which helps in monitoring a patient's progress studying activity of patients and correlating it with heart rate allows the doctor to detect anomalies well in time. Ecg segments , heart rate variability all of which is vital information to the doctor, which when foreseen timely could significantly mitigate risks. Areas like AJK with very few doctors compared to the population dwelling there could greatly benefit by benefiting more patients from remotely monitoring them, areas like AJK and KPK are greatly affected by natural disasters such as landslide, timely detection of a patient in need could serve to the benefit.

1.1.9 Project Plan

The project will have 3 main parts:

- Hardware for signal acquisition
- PC/laptop for data processing and cloud storage
- Companion Mobile Application to display the processed data

The features described below are tentative and will be changed based on progress made and milestones achieved.

1.1.9.1 Hardware

The hardware portion is the most crucial feature of this whole project. Its purpose is to reliably collect the required data and then transfer it to a central processing station For now, we are targeting 3 features based on which, later on, more features can be added. These are as follows:

Electrocardiogram (ECG)

For acquisition of the ECG signal we're proposing a single lead, 2 electrode solution with the possible addition of a third electrode. The instrumentation shall be done in a 3-stage process first being the instrumentation amplifier, second being a high pass filter and finally Analogue to Digital Conversion with suitable resolution and sampling rate.

Accelerometer

MEMS sensors are both low cost and low power which is exactly what is required for the proposed design. Depending on availability we will use either ADXL345/346 3-Axis Digital Accelerometers manufactured by Analogue Design.

Body Skin Temperature

For this either an NTC thermistor or a digital thermometer like the LM35 will be used.

1.1.9.2 Data Processing and Cloud Storage

This is the brain of the project. Its task is to handle all signal processing and data interpretation. The input is raw data and the output is interpreted data. Following are the features that we plan to extracted or conditions that will be detected:

Electrocardiogram (ECG)

- **Heart Rate**

  This is the most basic feature of the our system that will calculate the heart rate in beats/min or beats/sec.

Arrhythmia Detection

This is related to pattern recognition in the ECG waveform. A few things fall into this category:

- o Ventricular Tachycardia
- o Ventricular Fibrillation
- o Flutter

Accelerometer

This will be used to aid analysis of ECG waveform. Activity and Orientation data will be co-related with the heart status from ECG to observe long term trends and draw better conclusions.

- **Body Orientation/Posture**

  Detection of a person's current orientation e.g. laying on back/front or standing up/sitting etc.

- **Activity Detection**

  Detection of current body activity e.g. running ,walking, jumping etc.

Body Skin Temperature

Derivation of relation between body temperature, current activity and orientation and the heart condition.

1.1.10 End Product Description

The end product has two aspects and below are the end products form factors that we desire to achieve.

1.1.10.1 Hardware

We want to develop a device with a really small form factor that can be worn without any hindrance to daily routines. The device should be a wearable or something that sticks to the body using special adhesives that don’t leave the skin sticky. The Images below explain our idea of the implementation in a good manner. Plus we would like to have a flexible PCB so that the device is somewhat resistance to breaks because of use.

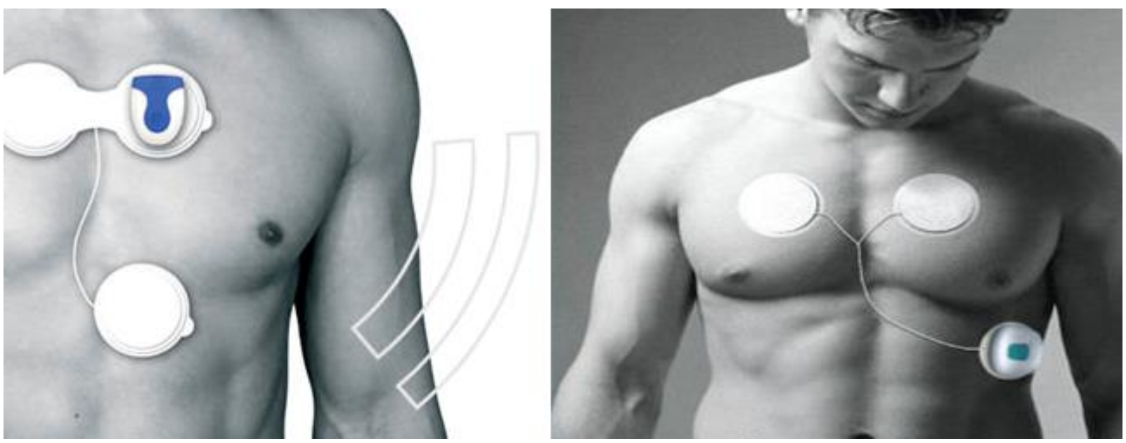


Figure 3- Wearable Device Concept

1.1.10.2 Software

On the software end we have three things; desired goals of all three are described below:

Data collection and processing app

The goal is to have a silent app that runs in the background and on the users input/desire displays the waveform. This is a desktop/Laptop app that will upload the processed data to the server/cloud storage.

Cloud storage

There should be enough space online to store data of patients for at least a month

Mobile Application

This app should will pull all of the data that was processed by the desktop app and upload to the server. There should be an integrated environment which shall allow the patients to view their history and share it too with doctors and other patients. The app shall also notify medical authorities in case of an emergency.

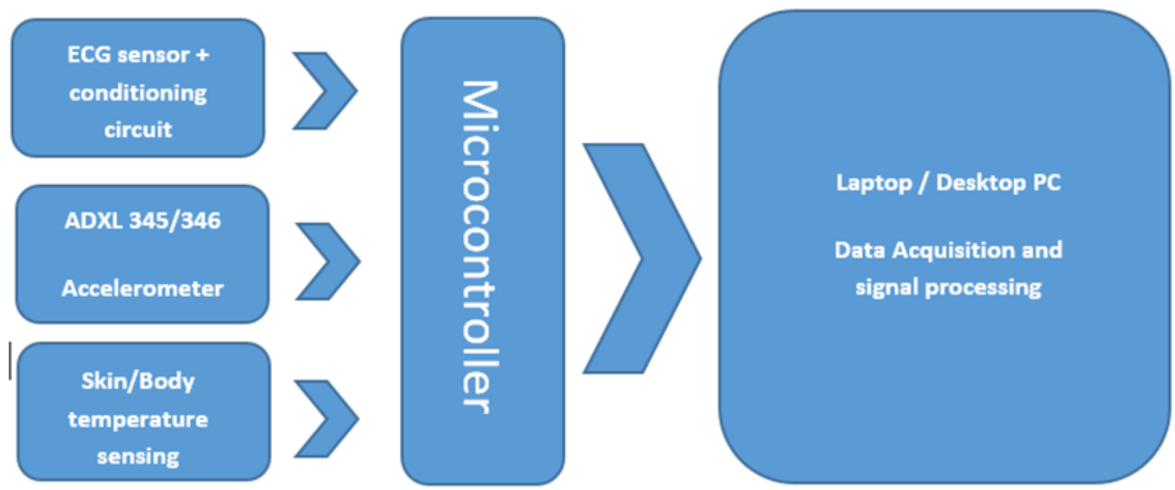


Figure 3 – Hardware and Software Breakdown

Chapter 2

2.1 LITERATURE REVIEW

2.1.1 INTRODUCTION

Through our platform, patients with heart disease can send information about their heart rate straight to their doctors, accessories allow them to monitor their blood sending the results straight to their smartphone, makes a statistics including the previous results, inform the patient if it is out of range.

Keeping track of blood pressure is a good habit to keep healthy. High blood pressure (HBP) is one of most common and frequently-occurring diseases; heredity, unbalanced diet, fat, lack of exercise, and drink etc may cause high blood pressure.

You can use our app to track your blood pressure, you can monitor your blood pressure closely and visual by words, chart and histogram. When you find it abnormal, you can take quick actions to find the reason and keep it from growing high. Also you can share measurements conveniently with your doctors by this app. Following are the previously available solutions in the market at the time of the manufacturing of our product.

2.1.1.1 Qardiocore

- Measures and records your EKG/ECG continuously on three channels.
- Sensors for heart rate, heart rate variability detection, body temperature, stress level via galvanic skin response, respiratory rate and activity tracking.
- Tracks your measurements over time with smart charts, trends and stats.
- Fully integrates with the Qardio product family using the companion Qardio App which automatically stores all your data in Qardio's secure cloud.
- Allows you to share your measurements and progress automatically with your doctor, trainer and family.

2.1.1.2 Qardioarm

- Measures your systolic, diastolic blood pressure as well as heart rate, includes irregular heartbeat detection.

- Tracks your measurements over time with smart charts, trends and stats.
- Supports multiple users.
- Fully integrates with the Qardio product family using the companion Qardio App which automatically stores all your data in Qardio secure cloud.
- Allows you to share your measurements and progress automatically with your doctor and family.
- Works with Apple Health and integrated with Apple HealthKit.
- Compatible with iOS and Android devices.

2.1.1.2 Vitaliti

- It measures the active Heart Rate, resting Heart Rate, heart Rate Variability
- This app can measure respiration rate, core Body Temperature, counts steps taken
- It can calculate calories burned, posture index and sleep cycles
- Gathers advanced vital sign & movement data comfortably, continuously and noninvasively. We use this data to offer you unprecedented insight into your workout, going far beyond typical fitness wearables, or even upcoming tech.
- With this level of continuous data we can show you detailed efficiency metrics, recovery time stats that include heart rate, breathing and blood pressure data, and much more.
- You can help monitor someone vital signs remotely.

2.1.1.3 Blood Pressure Companion Free

- Blood Pressure Companion Free is a blood pressure, heart rate and weight tracker.
- Easily add and edit blood pressure, heart rate, weight and test date.
- Add note to the readings
- Track systolic, diastolic, heart rate and weight by chart.
- Analyze times and percentage of each blood pressure range.

- Calculate Mean Arterial Pressure (MAP) of each reading.
- Create a reminder for measuring BP, doctor visiting or anything else.
- Show MAX, MIN and AVG number of readings by number and histogram.
- Show statistics of measurement by day, week, month or year.
- Create a reminder to measure your blood pressure, doctor visit or others.
- Save snapshots to Photos.
- Export (Email) readings in format of CSV, HTML or PDF.
- Custom date range of readings to be exported.
- Migrate Data to full version.
- Custom your own color range.
- Passcode function to protect your info.
- Supports Touch ID as passcode.
- Support lbs and kgs.
- Clean and humanized interface.
- Supports printing data in app.
- Supports transferring data to HealthKit.

2.1.1.3 Blood Pressure Monitor - Family Lite

- Comprehensive reading input fields collection
- Customizable input data form
- Both oral and injection types of medication tracking and correlation
- Left/right arm tracking and correlation for blood pressure measurement
- Body position tracking and correlation for blood pressure measurement
- Meal type tracking and correlation for glucose measurement
- Data visualization – a chart is worth a thousand words. The graphical charts not only give you a visual trend of the vital signs, but also show warning signs (BP only).

- Email Import/Export – simple yet versatile email/export feature lets you communicate with your healthcare provider at a breeze. Multiple export formats (PDF, CSV, plain message) to choose from
- Selectable vital sign category to export
- Charts included in emails and PDF files
- Flexible format recognition during import
- Built-in reminder, never miss a check up time and seamless integration with iOS 5 reminder app
- Multiple user accommodation, one app to track the whole family (upgrade needed)

2.1.1.4 HealthPatch® MD

- Collects data simultaneously
- Time stamped data for analytics (For example, vital sign changes such as an increase in heart rate or respiratory rate may be viewed relative to a patient's activity and body posture)
- Biosensors continuously tracks data such as increase in heart rate
- Transmits to health care provider to act accordingly
- Connected by bluetooth low energy and WIFI for delivery of data to mobile and server

2.1.1.5 Heart Rate Plus

- Real time pulse graph
- Android Wear support: Measure your pulse using the heart rate sensor in your smartwatch
- Reminder: Automatic remind you to measure your heartbeat everyday
- Export history to CSV file (ads free user only)
- While you perform the breathing exercise, HeartRate+ monitors your heart rate . hold the tip of your finger over the camera lens .
- Measure how well you are doing your breathing exercises by means of displaying the reached level of Coherence and the related average score.

- Breathing Guide
- History of exercises
- Sharing on Twitter, Facebook and email
- Customizable Breathing Pattern

2.1.1.6 Grace 2.0 ACS calculator

- For patients with ACS
- Calculates the risk assessment
- Measures age, heart rate, systolic blood pressure,creatinine levels,Killip class etc for risk assessments
- Population histograms with high-, medium- and low-risk markers
- New calculations provide probabilities directly, bypassing scores

2.1.1.7 JBS3 risk calculator

- Calculates the risk of a stroke
- Empower patients to make appropriate decisions about their lifestyle and drug treatments
- Provides tool available online
- Paid app works for patients and their doctors
- Calculates heart age, makes assessment for lifestyle changes

2.1.1.7 Cardiio

- Monitors heart rate
- Provides weekly view
- Provides daily dashboard
- Gives monthly summary
- Works with your camera flash, calculates heart rate by the amount of light reflected from your face

2.1.1.8 ASCVD Risk Estimator

- This Risk Estimator enables healthcare providers and patients to estimate 10-year and lifetime fatal or nonfatal stroke, based on the Pooled Cohort Equations and lifetime risk prediction tools.
- First, you're asked basic questions related to age and gender, as well as smoking habits, your history of hypertension and diabetes.
- The data is processed and you are given a 10-year risk assessment, as well as an overall lifetime risk estimate. If you have risk factors, check out the app's extra resources for healthy lifestyle tips to reduce the risk of ASCVD.

2.1.1.9 Cardiograph

- It has multiple profile capabilities can store each person's heart rate and track it over time
- Cardiograph works by illuminating your index finger with your device's camera flash, rendering your finger red and measuring the changes in the color caused by your pulse.
- Measure and track heart rate, keeps track of multiple individuals' heart rate, add notes, locations, and print out measurements if desired.

2.1.2 WEAKNESS OF PREVIOUS SOLUTIONS

1. Data entry is tedious that is user manually input the data.
2. Risk estimators are not accurate
3. Camera flash heart rate measurement has varying results so unreliable
4. Not a single platform for measuring all vitals for heart congestive patients
5. Doctors are not notified timely
6. Not for heart congestive patients

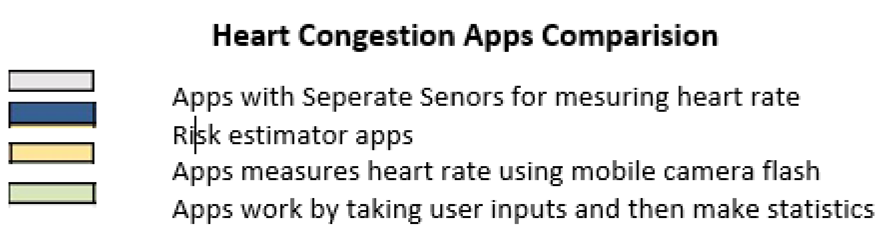
2.1.3 SURVERY TAKEAWAY

We have done a detailed survey about the apps that are related to our work.The apps we reviewed can be categorized into four types. For the *first category*, we found out that most of the apps uses the user's manual input to make the statistic using previous record also. This is very tedious process for the patient to enter every minor detail. *Second category* uses the risk estimator of your heart condition which is not appropriately valid and you never know if the result is correct or not. *Third category* of apps uses mobile camera flash which also do not give appropriate results. *Fourth category* involves the sensors device separately which measure the reading and send it to app and make graphs.

We downloaded the apps and tested it on ourselves but there was such a variations in in results in every one results. Moreover in these apps none of them provide collective features. They are not giving a patient, doctor caretaker interaction platform.

We are providing a patient-doctor portal in which both are able to communicate. A doctor can see the patient result in the form of charts and recommend him. If the measurements are not in range a notification will be sent to the doctor to tell him what immediate actions should be taken. We are also providing him the feature of meal tracking patient will be able to enter the meal he has taken and note down the consequences of him. Next time when the patient enters the data, there will be alert generated of his previous experience.

2.2 COMPARISON



	ECG	Heart Rate	Heart rate variability	Respiratory rate	Skin temperature	Body posture	Activity	Fall detection	accelerometer	Blood pressure	App compatibility	Device	Show data statisti
Qardioarm	Yes	Yes	Yes	Yes	Yes				yes	yes	yes	yes	yes
Qardiocore	Yes	Yes	Yes	Yes	Yes				yes	yes	yes	yes	yes
Healthpatch MD	Yes	Yes	Yes	Yes	Yes	Yes	Yes	yes			Yes	Yes	
Vitaliti	Yes	Yes	Yes	Yes	Yes	Yes	Yes		yes	yes	yes	yes	yes
Holst center and imec	Yes								yes		yes	yes	
gace ACS risk estimator		Yes								yes			yes
ACS calculator		Yes								yes		no	
JBS3 risk calculator		Yes								yes			yes
ASVD risk estimator						yes					yes	no	
Cardio	yes	yes								yes		yes (camera light reflected by the face	yes(su mary n graphs
heart rate + coherence	yes	Yes	Yes	yes			yes (by doing exercise)			yes		Hold the tip of your finger over the camera lens while exercising	
instant Heat Rate	yes	Yes	Yes							yes		yes (camera)	
Cardiograph		yes									Yes	Yes (camera)	

	ECG	Heart Rate	Heart rate variability	Respiratory rate	Skin temperature	Body posture	Activity	Fall detection	accelerometer	Blood pressure	App compatibility	Device	Show data statistic	multi users	Data sharing	Notification to doctor
Blood pressure companion		Yes	Yes							Yes			yes (per week,month,year) graphs	yes (multi patients)	email to Doctor	Yes
Blood pressure monitor		Yes	Yes							Yes			Yes			Yes
iBP Blood pressure		Yes	Yes							Yes		yes . By ome other company	Yes			
Smart Blood pressure		Yes	Yes							Yes			Yes			
HeatWise Blood pressure Tracker		Yes	Yes							Yes			Yes		yes	

Figure 4– Available Solution Comparison

Chapter 3

FUNCTIONALITY AND DESIGN

AEGIS - is designed to assist cardiac patients in order to in enable them to live life like normal person. Our solution carries the following unique features:

Automated system: It will notify the medical authorities in case of need / emergency

Community: Using the mobile application will allow users to share their experience and medical details.

Wearable Product: It will not hinder the patients allowing them to move around normally

First Aid Instructions: In case the patient faints the mobile app shall give out basic first aid instructions to any person in close vicinity.

Internet of Things IOT: We aim to make this product the torch bearer in the Pakistani medical IOT product lineup.

In order to implement this we have gone through a rigorous process of design and development of this product. After checking each and every component in detail and making sure that it the end product is top notch in design, build and quality. To begin with following are the components that were used in the design step.

1. Accelerometer for body posture/orientation detection
2. ECG sensing circuitry for ECG sensing

We shall discuss it further that why and how a specific model of these instruments was chosen and what edge it includes over the other available options. In general we shall look for the following qualities in every component.

1. High efficiency and low cost
2. Easily accessible
3. Long lasting

3.1.1 Accelerometer:

The accelerometer we've employed in our final design implementation is ADXL-345 / ADXL-346 3-Axis Digital Accelerometers by Analogue Design. The ADXL345 is a small, thin, ultralow power, 3-axis accelerometer with high resolution (13-bit) measurement at up to ±16 g. Digital output data is formatted as 16-bit twos complement and is accessible through either a SPI (3- or 4-wire) or I2C digital interface.

The ADXL345 is well suited for mobile device applications. It measures the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion

or shock. Its high resolution (3.9 mg/LSB) enables measurement of inclination changes less than 1.0°.and it involves the following feature.

Several special sensing functions are provided. Activity and inactivity sensing detect the presence or lack of motion by comparing the acceleration on any axis with user-set thresholds. Tap sensing detects single and double taps in any direction. Freefall sensing detects if the device is falling. These functions can be mapped individually to either of two interrupt output pins.

An integrated memory management system with a 32-level first in, first out (FIFO) buffer can be used to store data to minimize host processor activity and lower overall system power consumption. Low power modes enable intelligent motion-based power management with threshold sensing and active acceleration measurement at extremely low power dissipation. [4]

In summary it has following edges to offer.

- MEMS sensors to ensure low volume and low weight system
- Low cost and low power as indicated in the datasheet below
- Breakout Board easily available

Figure 5 – 3-Axis Digital Accelerometer ADXL 345/346

3.1.2 Features:

- Ultralow power: as low as 23 μA in measurement mode and
- 0.1 μA in standby mode at VS = 2.5 V (typical)
- Power consumption scales automatically with bandwidth
- User-selectable resolution
- Fixed 10-bit resolution
- Full resolution, where resolution increases with g range,
- up to 13-bit resolution at ±16 g (maintaining 4 mg/LSB
- scale factor in all g ranges)
- Embedded memory management system with FIFO
- technology minimizes host processor load
- Single tap/double tap detection
- Activity/inactivity monitoring
- Free-fall detection
- Supply voltage range: 2.0 V to 3.6 V
- I/O voltage range: 1.7 V to VS
- SPI (3- and 4-wire) and I2C digital interfaces
- Flexible interrupt modes mappable to either interrupt pin
- Measurement ranges selectable via serial command
- Bandwidth selectable via serial command
- Wide temperature range (−40°C to +85°C)
- 10,000 g shock survival
- Pb free/RoHS compliant

- Small and thin: 3 mm × 5 mm × 1 mm LGA package

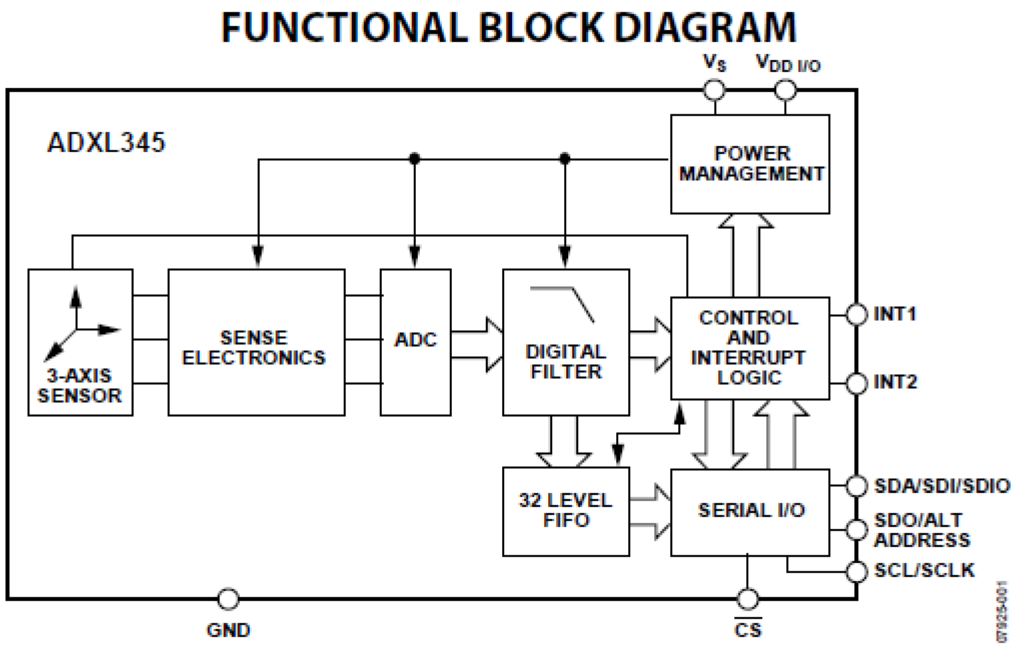


Figure 6 – Functional Block Diagram of Accelerometer ADXL 345/346

3.2 ECG SENSING CIRCUITRY

The ECG sensing cicuitry that has been used in the development of AEGIS consits of Analog Devices AD823. The AD8232 is an integrated signal conditioning block for ECG and other biopotential measurement applications. It is designed to extract, amplify, and filter small biopotential signals in the presence of noisy conditions, such as those created by motion or remote electrode placement. This design allows for an ultralow power analog-to-digital converter (ADC) or an embedded microcontroller to acquire the output signal easily.
The AD8232 can implement a two-pole high-pass filter for eliminating motion artifacts and the electrode half-cell potential. This filter is tightly coupled with the

instrumentation architec-ture of the amplifier to allow both large gain and high-pass filtering in a single stage, thereby saving space and cost.
An uncommitted operational amplifier enables the AD8232 to create a three-pole low-pass filter to remove additional noise. The user can select the frequency cutoff of all filters to suit different types of applications.
To improve common-mode rejection of the line frequencies in the system and other undesired interferences, the AD8232 includes an amplifier for driven lead applications, such as right leg drive (RLD).
The AD8232 includes a fast restore function that reduces the duration of otherwise long settling tails of the high-pass filters. After an abrupt signal change that rails the amplifier (such as a leads off condition), the AD8232 automatically adjusts to a higher filter cutoff. This feature allows the AD8232 to recover quickly, and therefore, to take valid measurements soon after connecting the electrodes to the subject.
The AD8232 is available in a 4 mm × 4 mm, 20-lead LFCSP package. Performance is specified from 0°C to 70°C and is operational from −40°C to +85°C.
The AD8232-EVALZ evaluation board contains an AD8232 heart rate monitor front end conveniently mounted with the necessary components for initial evaluation in fitness applications. Inputs, outputs, supplies, and leads off detection terminals are routed to test pins to simplify connectivity. Switches and jumpers are available for setting the input bias voltage, shutdown (SDN), fast restore (FR), and ac/dc, leads off detection mode. [5]

3.2.1 Features

- Features and Benefits
- Fully integrated single-lead ECG front end
- Low supply current: 170 µA (typical)
- Common-mode rejection ratio: 80 dB (dc to 60 Hz)
- Two or three electrode configurations

- High signal gain (G = 100) with dc blocking capabilities
- 2-pole adjustable high-pass filter
- Accepts up to ±300 mV of half cell potential
- Fast restore feature improves filter settling
- Uncommitted op amp [5]

3.2.2 Applications

- Fitness and Activity Heart Rate Monitoring
- Portable ECG
- Remote health monitors
- Gaming peripherals
- Biopotential signal acquisition

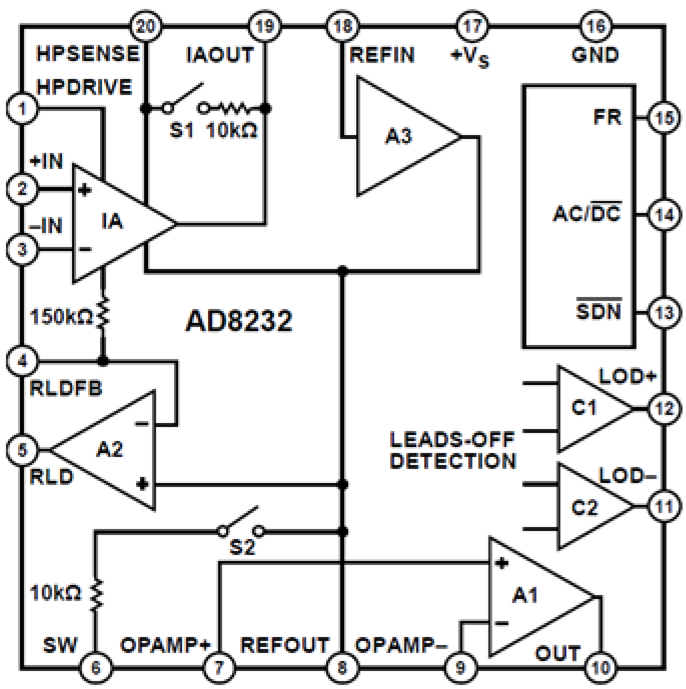


Figure 7 – Breakout Board AD8232

Chapter 4

4.0 IMPLEMENTATION

4.1. Hardware Components

4.1.1. Controller: RF Digital RFD22301

The controller that we chose to use is a hi performance, professional grade (Bluetooth Low Energy) BLE compatible radio transceiver with built-in ARM Cortex M0 microcontroller that can be programmed using the Arduino IDE using RFduino extensions. It comes in both DIP and SMD packages.

Figure – 8 Controller: RF Digital RFD22301

As show in figure above it possesses 7 usable GPIOs all of which can act as digital or analogue pins depending on mode selected. It possesses an inbuilt 10-bit ADC which has a default 3-V voltage reference which can be overridden using pin GPIO0/AREF. However, the reference analogue voltage cannot be above 3 V. I2C, UART and SPI protocols have been built in.

For wireless communication RFduino uses an nRF51822 BLE chip core from Nordic Semiconductors which is a powerful, highly flexible multiprotocol SoC ideally suited for Bluetooth® Smart and 2.4GHz ultra low-power wireless applications. It is built around a 32-bit ARM® Cortex™ M0 CPU with 256kB/128kB flash + 32kB/16kB RAM for improved application performance. The embedded 2.4GHz transceiver supports both Bluetooth Smart and the Nordic Gazell 2.4 GHz protocol stack.

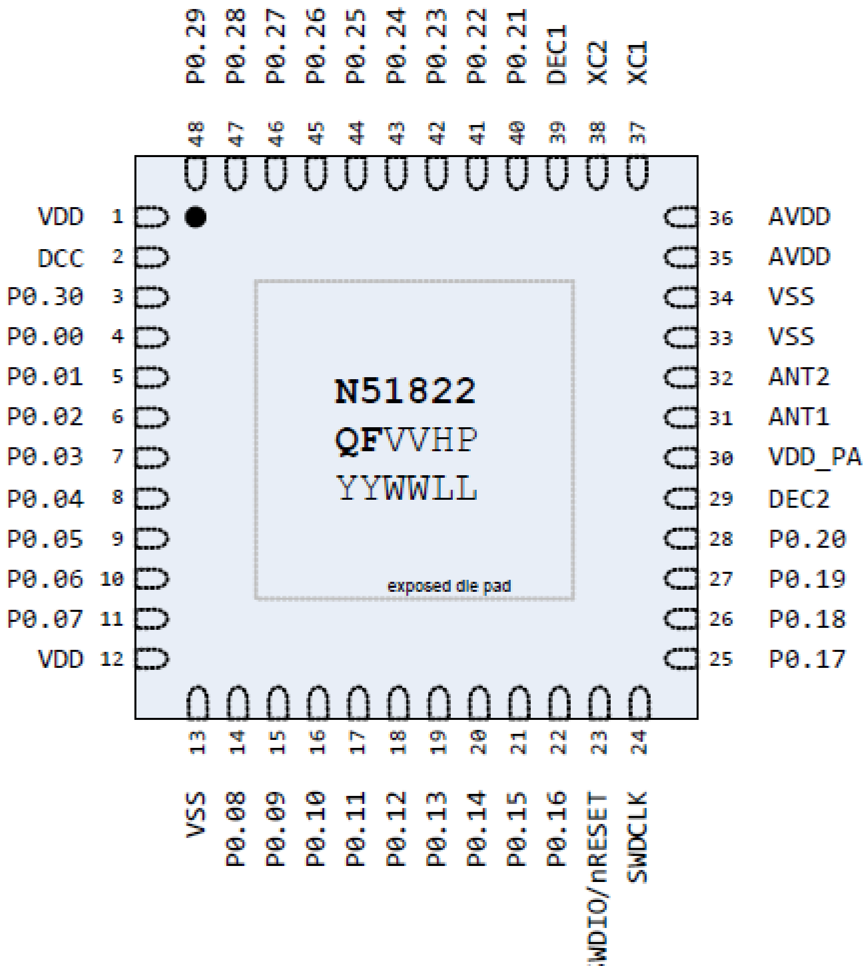


Figure 9 - nRF51822 BLE chip core

4.1.2. Analogue Front End: Analog Devices AD8232

The AD8232 is an integrated signal conditioning block for ECG and other biopotential measurement applications. It is designed to extract, amplify, and filter small biopotential signals in the presence of noisy conditions, such as those created by motion or remote electrode placement. The block diagram below highlights most features.

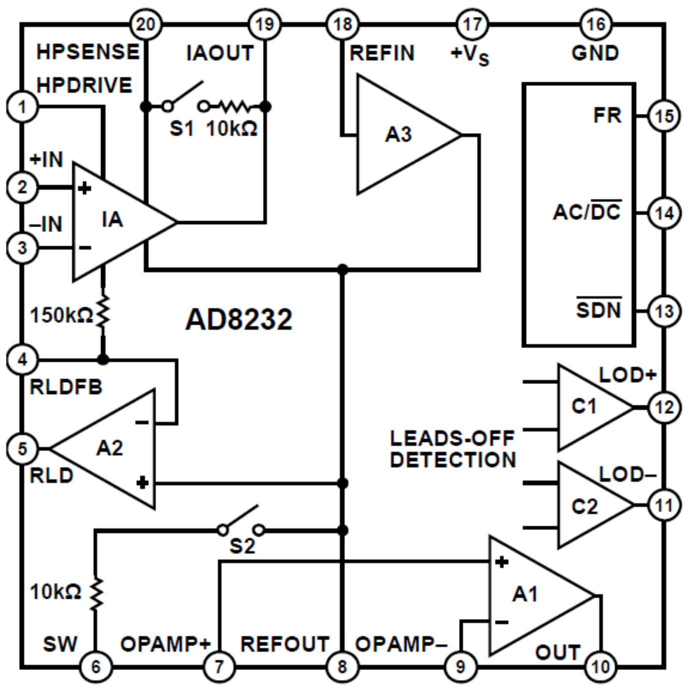


Figure 10 – AD8232 circuit breakdown

It implements a two-pole high-pass filter for eliminating motion artifacts and the electrode half-cell potential. An uncommitted operational amplifier enables the AD8232 to create a three-pole low-pass filter to remove additional noise.

To improve common-mode rejection of the line frequencies in the system and other undesired interferences, the AD8232 includes an amplifier for driven lead applications, such as right leg drive (RLD).

In terms of electrical features it can operate with a low supply current of just 170 µA from a single-supply of 2.0 V to 3.5 V which is compatible with the RFduino operating voltage of 3.0 V nominal.

4.1.3. Digital Accelerometer: Analog Design ADXL345

The ADXL345 is a small, thin, ultralow power, 3-axis accelerometer with high resolution (13-bit) measurement at up to ±16 g. Digital output data is formatted as 16-bit twos complement and is accessible through either a SPI (3- or 4-wire) or I2C digital interface.

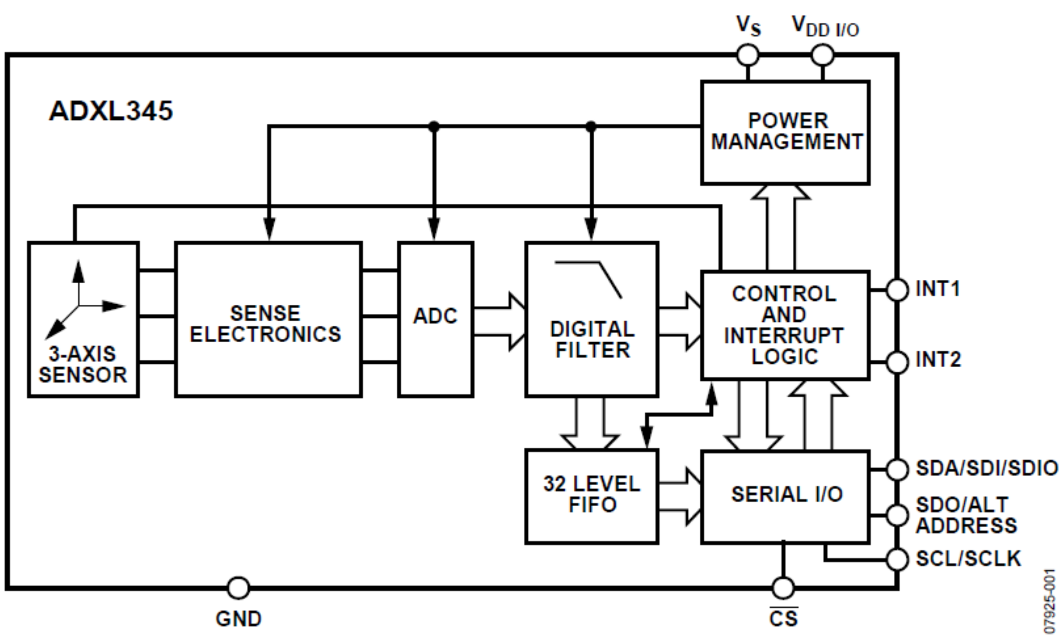


Figure 11 – ADXL 345 Configuration

The ADXL345 is well suited for mobile device applications. It measures the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion or shock. Its high resolution (3.9 mg/LSB) enables measurement of inclination changes less than 1.0°.

Several special sensing functions are provided. Activity and inactivity sensing detect the presence or lack of motion by comparing the acceleration on any axis with user-set thresholds. Tap sensing detects single and double taps in any direction. Free-fall sensing detects if the device is falling. These functions can be mapped individually to either of two interrupt output pins. An integrated memory management system with a 32-level first in, first out (FIFO) buffer can be used to store data to minimize host processor activity and lower overall system power consumption.

It boasts an ultralow power mode with operating current as low as 23 μA in measurement mode and 0.1 μA in standby mode from a supply of 2.0 V to 3.6 V. Power consumption scales automatically with bandwidth and since the resolution is user-selectable power can be effectively managed.

4.1.4. Voltage Regulator: STMicroelectronics LF33

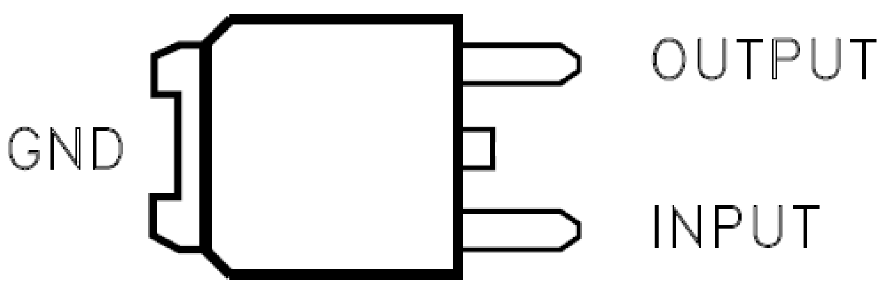


Figure 12 – LF33 Voltage Regulator Configuration

The LFXX series consist of very low drop regulators available in TO-220, TO-220FP, DPAK and PPAK packages and in a wide range of output voltages. The low drop voltage (0.45 V) and low quiescent current make it particularly suitable for low-noise, low-power applications and especially in battery powered systems.

4.2. Circuit Configuration and Implementation

4.2.1. Analogue Front End AD8232

AD8232 can be used in many different configuration depending on the network of capacitors and resistors it is connected to. Since we're interested in the shape of the ECG signal, the cut off frequencies of the implemented filters need to be set such that the entire spectrum of frequencies can pass. Hence, we've chosen to implement the *'Cardiac Monitoring Configuration'* as detailed in the '*Application Circuits'* section of the AD8232 Datasheet.

To obtain the ECG waveform with minimal distortion, the AD8232 is configured with a 0.5 Hz two-pole high-pass filter followed by a two-pole, 40 Hz, low-pass filter. A third electrode is driven for optimum common-mode rejection. The circuit schematic as well as the frequency response of this configuration is shown in the diagrams below:

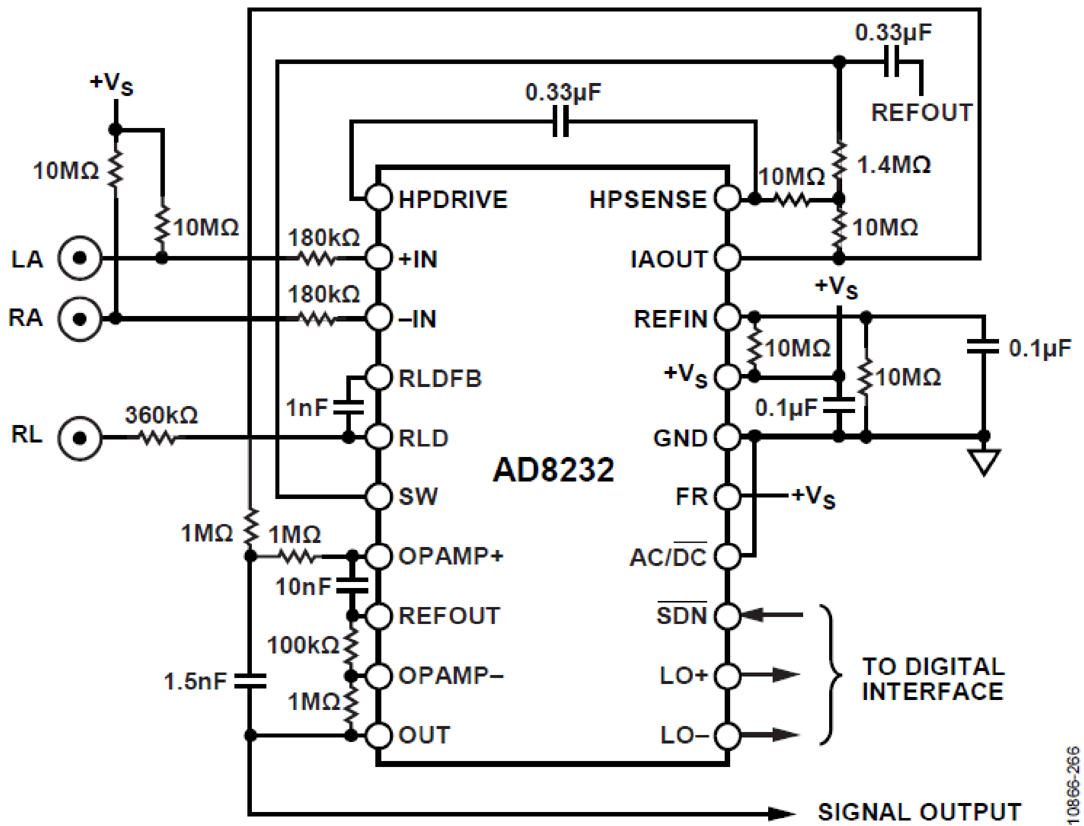


Figure 13 – AD8232 Breakout Board Configuration

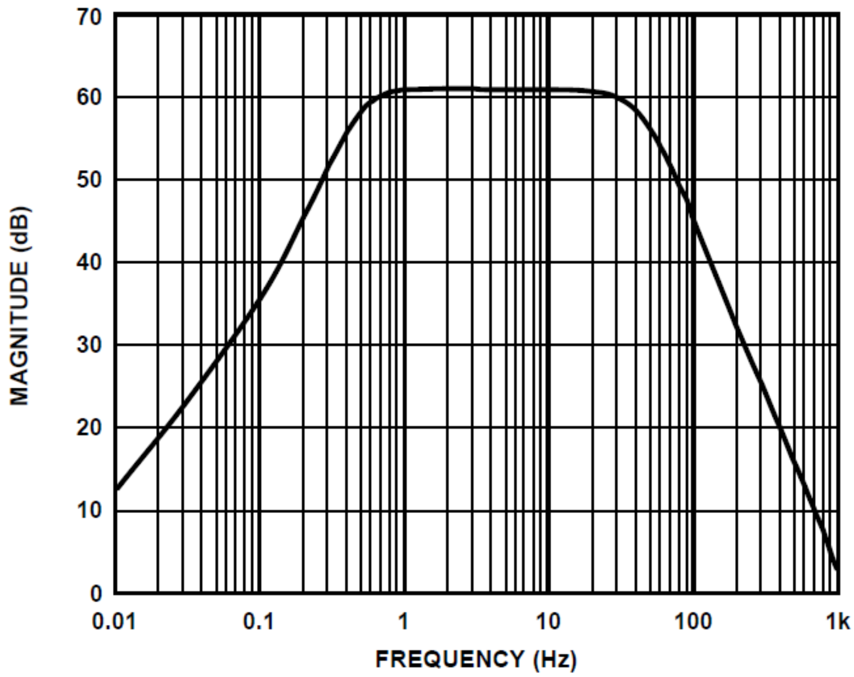


Figure 14 – AD8232 Frequency Response

Luckily the PCB for this very configuration was available from the online retailer Sparkfun under the title of '*Single Lead Heart Rate Monitor - AD8232'*. The form factor is exactly what we wanted along with welcome addition of a 3.5 mm headphone jack to hook up the electrodes easily. The PCB is shown below:

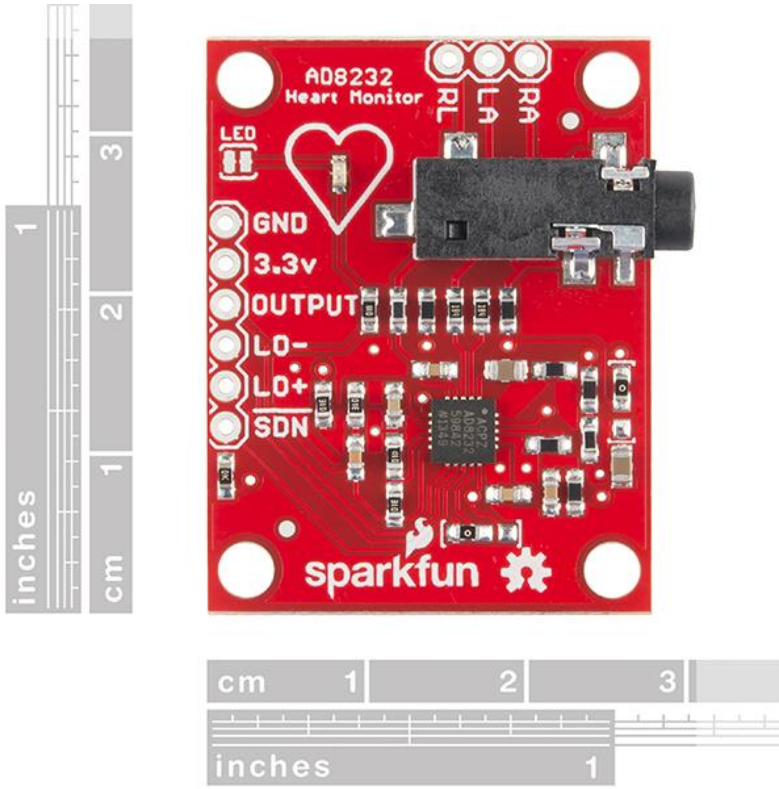


Figure 15 – AD8232

4.2.2. Digital Accelerometer ADXL345

ADXL 345 is a simple to use device with tons of documentation and examples available. We chose to use the ADX345 breakout board from Adafruit which is pictured below because it has been configured to I2C mode by tying the **CS’** pin to V_{DD} and it uses just 2 wires apart from Power and Ground to communicate which is plus point since the RFduino has just 7 GPIOs.

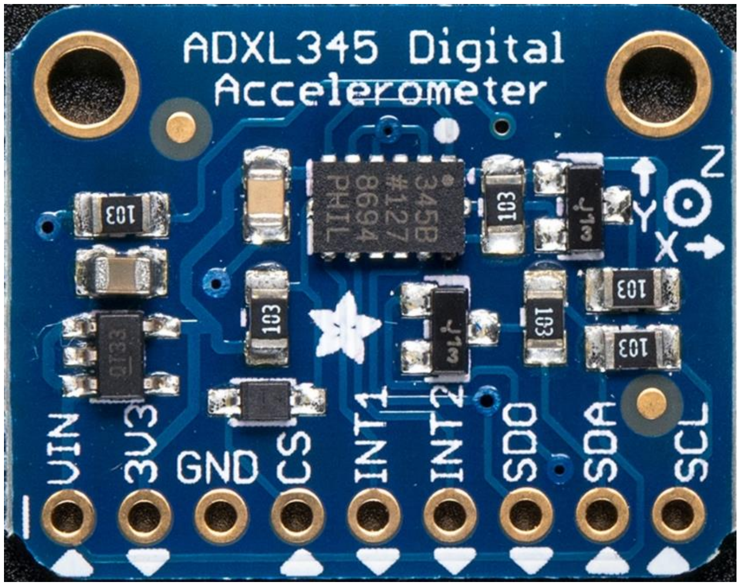


Figure 16 – ADXL345

4.2.3. Overall Circuit

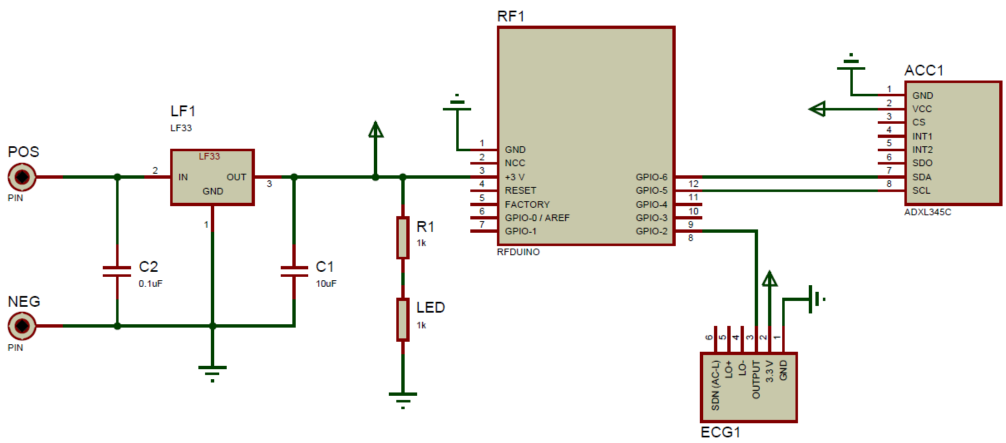


Figure 17 – Overall Configuration

Since the AD8232 provides an analogue output which contains the ECG signal hence it can be connected to any GPIO of RFduino. In our case it has been connected to GPIO-2.

For I2C communication the RFduino has default SDA and SCL pins at GPIO 6 and GPIO 7 respectively which can be remapped to any combination of a pair of GPIOs using the function Wire.beginOnPins(SCL pin, SDA pin).

Other circuitry include an LED to indicate that the device is up and running and the voltage regulator LF33 which will be connected to a Samsung 1350 mAh Li-Po Battery used in the smartphone Galaxy Y.

4.2.4. Printed Circuit Boards

All design files (.pdsprj, Gerber, .sch and .pcb) files have been included in the FYP disc accompanying this report. Given below are the screenshots (not to size) of the PCB designs

4.2.4.1. Design based on breakout boards (Proteus)

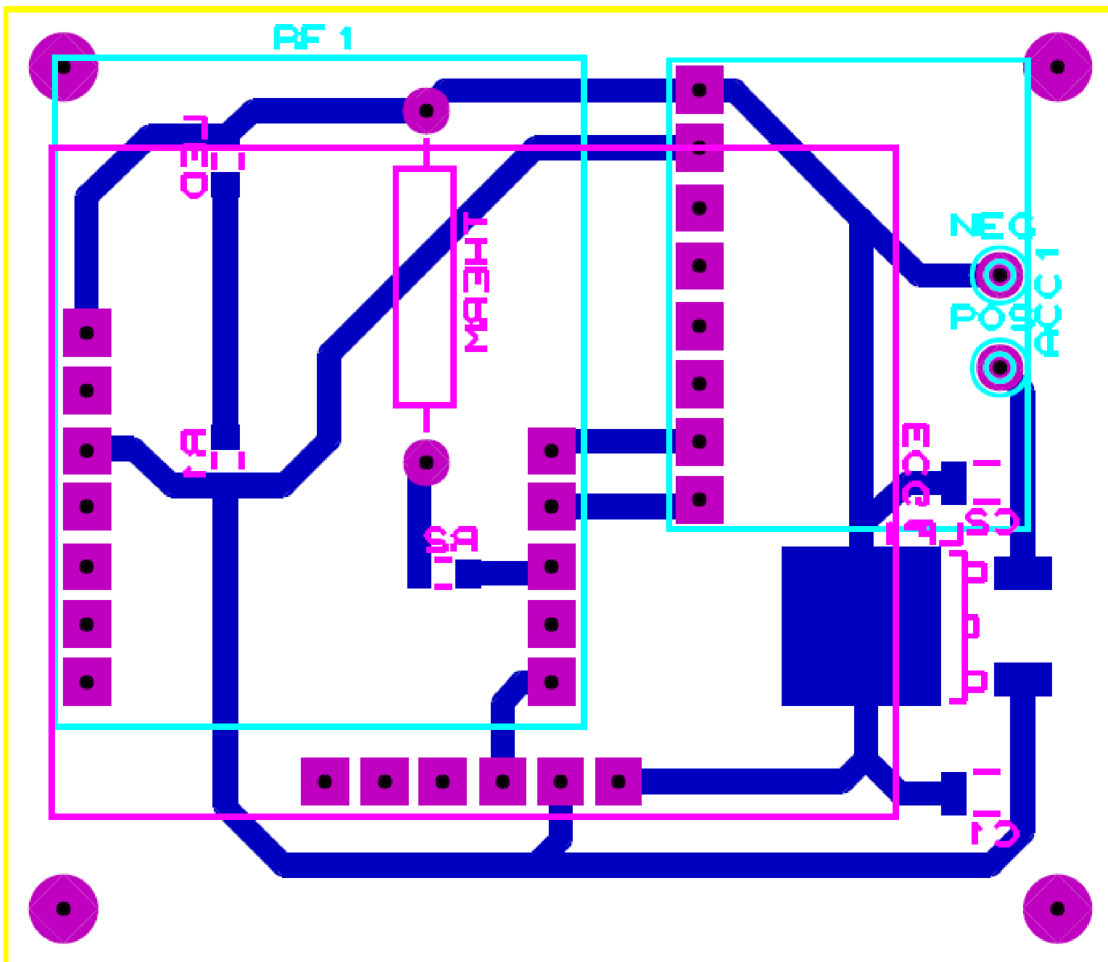


Figure 18 – PCB Design 1

4.2.4.2. Single Board Chip Based Design (Eagle CAD)

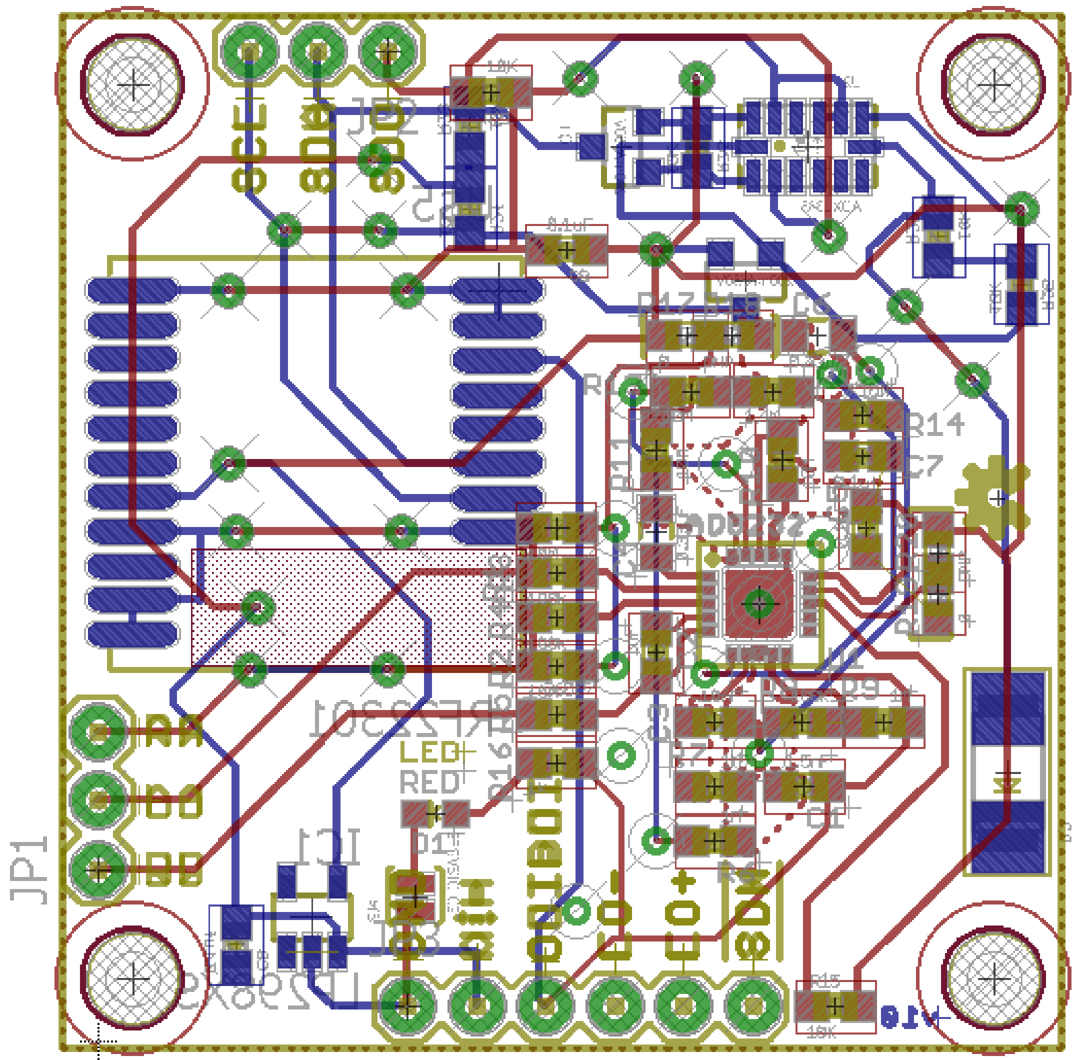


Figure 19 – PCB Design 2

4.3. Microcontroller Coding and IDE Setup

4.3.1. Setting up the IDE

The RFduino is programmable via the Arduino IDE by using library extensions and a custom bootloader developed RFDigital. Before the RFduino can be bootloaded a few steps need to be performed:

1. **RFDuino library installation**

   The RFDuino library can be obtained from RFDigital GitHub repository which is open/free to download from the following link: https://github.com/RFduino/RFduino. Then this folder needs to extracted into the folder “Arduino /hardware /Arduino” as shown in the screenshot:

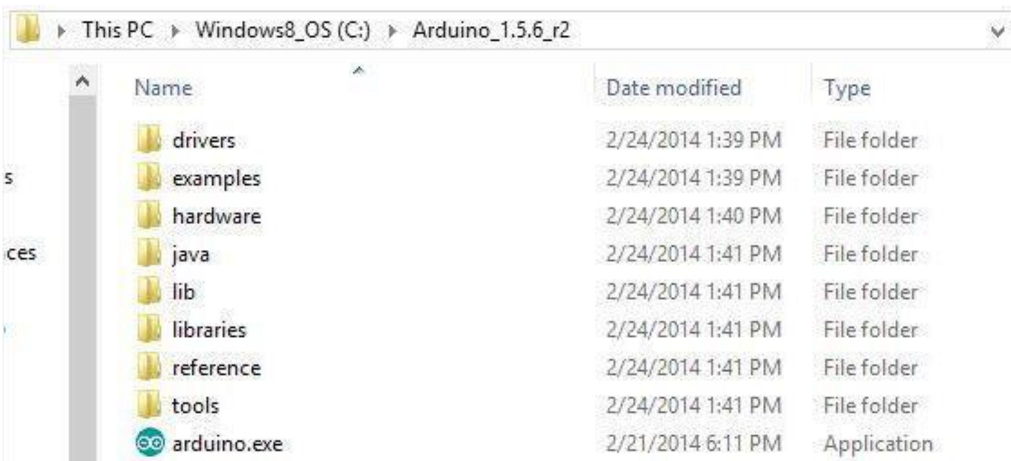


Figure 20 – RFDuino Library Installation

2. **RFduino USB Driver Installation**

   Next up we need to configure the FTDI drivers for bootloader to work properly. All that is required is to download the driver for the current operation system (windows/Mac OS) from the link below and the bootloader should show up as a com port in the system device manager. http://www.ftdichip.com/Drivers/VCP.htm

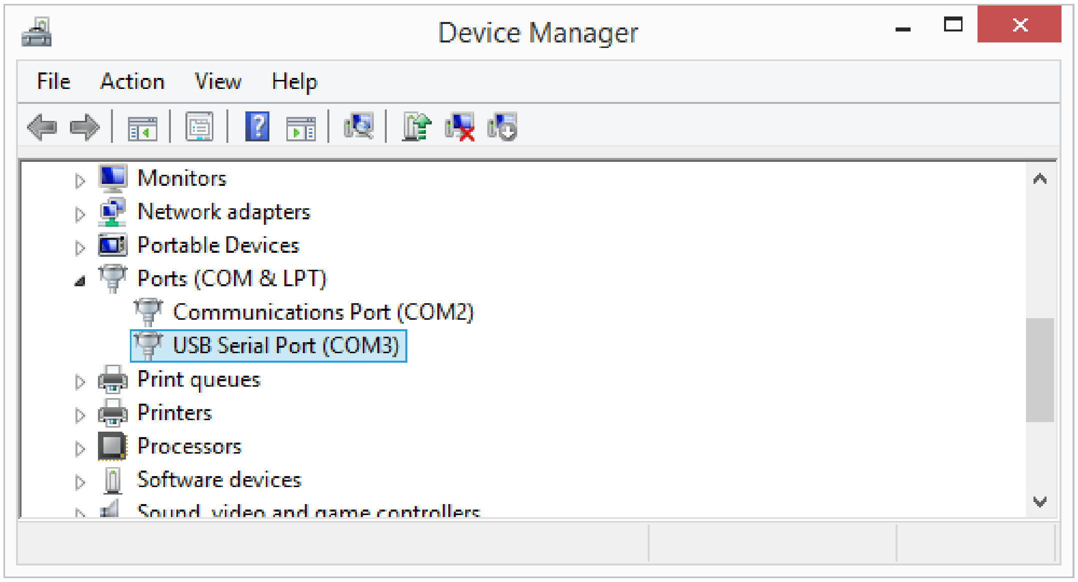


Figure 21 – RFDuino USB Driver Installation

4.3.2. Microcontroller Code Logic

The microcontroller code is rather straight forward and consists only of data collection at a specified rate of 250 Hz.

The full code is available at the end of the book and in the CD included.

4.3.2.1. Timer Configuration

Since the RFD22301 is based on the nRF51822 which has an M0 ARM processor hence the timer code is vastly different from the normal Arduino timer code. We need a timer that will generate an interrupt at 250 Hz which is every 4 ms (4000 µs). Based on that interrupt we will run an interrupt service routine which will trigger the flags necessary to acquire the data. Hence we the timer to be CTC (Count then clear mode) i.e. everytime the count reaches to value which corresponds to 4000 µs reset the count and start again. To achieve this in RFD22301 we do:

```
NRF_TIMER2->CC[0] = 4000;
NRF_TIMER2->INTENSET = TIMER_INTENSET_COMPARE0_Enabled << TIMER_INTENSET_COMPARE0_Pos;
```

```
NRF_TIMER2->SHORTS = (TIMER_SHORTS_COMPARE0_CLEAR_Enabled <<
TIMER_SHORTS_COMPARE0_CLEAR_Pos) & TIMER_SHORTS_COMPARE0_CLEAR_Msk;
attachInterrupt(TIMER2_IRQn, TIMER2_Interrupt);
NRF_TIMER2->TASKS_START = 1;
```

The very first line tell the value we need to count up to i.e. we're using CC[0] as a comparison register. Then in the next line we enable the compare mode and set it position comparison. Finally, in the third line we enable the clear mode and then set it to clear based on the position comparison. The last two lines only attach the ISR and start the timer. Timer 2 has been used because Timer 0 is reserved for BLE stack operation and Timer 1 is reserved for SPI and I2C communications.
The ISR routine is pretty simple and it only changes the value of a boolean variable named *'intrp'* which is used in the built-in loop() function of the Arduino IDE to decide whether or not to take measurements.

4.3.2.2. I2C communication with ADXL345

The ADXL345 breakout board from Adadruit if configured to be in I2C mode. I2C communication for the most part is easy to implement but the ADXL345 is a widely used chip and has a lot of libraries available for use. We have used the official libraries from Adafruit themselves which are:

- **Adafruit Unified Sensor Driver:** Adafruit_Sensor.h
  https://github.com/adafruit/Adafruit_Sensor
  This Library provides a common interface and data type for any supported sensor. It defines some basic information about the sensor (sensor limits, etc.), and returns standard SI units of a specific type and scale for each supported sensor type.
  It provides a simple abstraction layer between your application and the actual sensor HW, allowing you to drop in any comparable sensor

with only one or two lines of code to change in your project (essentially the constructor since the functions to read sensor data and get information about the sensor are defined in the base Adafruit_Sensor class).

The unified sensor abstraction layer is also useful for data-logging and data-transmission since you only have one well-known type to log or transmit over the air or wire.

- **Adafruit ADXL345 Accelerometer Driver:**
  Adafruit_ADXL345_U.h
  http://www.adafruit.com/products/1231
  This driver is for the Adafruit ADXL345 Breakout, and is based on Adafruit's Unified Sensor Library (Adafruit_Sensor).

Once these libraries have bee included we just need to configure the bau rate, ADXL345 data rate and the ADXL345 range in the setup(). Then the values are simple accessible by using 5 lines of code as follows:

```
//set up code
Serial.begin(115200);
accel.setDataRate(ADXL345_DATARATE_400_HZ);
accel.setRange(ADXL345_RANGE_8_G);

//loop code
sensors_event_t event;
accel.getEvent(&event);
accX = 8+event.acceleration.x;
accY = 8+event.acceleration.y;
accZ = 8+event.acceleration.z;
```

Every acceleration value has 8 added to make sure that he value of acceleration stays positive. Since the range is ±8 G, therefore the lowest value will be -8 G which when given a shift of +8 will be 0. Hence we're changing the range from ±8 G to 0 G - 16 G; this is to ease transmission by ensuring a constant buffer size.

4.3.2.3. Communication with AD8232

Communication with the AD8232 is perhaps the easiest one. The AD8232 provide and analog output in the range of 0-3V which need to converted to a digital value to be processed further. For this purpose, we’ve used the inbuilt 10-bit Analog to Digital Converter (ADC) which provides us 1024 digital levels ranging from 0 to 1023. The conversion from analog to digital is achieved using just a just a single line of code.

```
ecg_val=1000+analogRead(ecg_pin);
```

Since the values range from 0 to1023 by adding a 1000 here we make sure that there are always 4 digits; this is to ease transmission by ensuring a constant buffer size. Lastly we don’t need to map the 0-1023 values to a voltage because the algorithm in the smart phone application has been designed to process the waveform regardless of it’s amplitude

4.3.2.4. Bluetooth Low Energy (BLE) Communication with Smartphone

The final step is the transmission of the collected data. Since we have to send both the ADC reading and the three acceleration values. For this purpose, we need to construct a buffer and send that as a packet. The constraint however is that the buffer needs to be of the type char i.e. we need to construct a char array. Hence we proceed as follow:

```
ecg_val_s = String(ecg_val,DEC);
accX_s = String(accX,DEC);
accY_s = String(accY,DEC);
accZ_s = String(accZ,DEC);
final_string = String(ecg_val_s + "," + accX_s + "," + accY_s + "," + accZ_s);
char send_array[final_string.length()+1];
final_string.toCharArray(send_array, final_string.length()+1);
RFduinoBLE.send(send_array, final_string.length()+1);
```

First, Strings are generated for the ECG value and the Acceleration values. Then, these Strings are concatenated using the String() constructor and each values is separated a " , " delimiter to produce a comma separated value (CSV) buffer . Finally a char type array '*send_array*' is constructed of the required size and the '*final_string*' contents are transferred to the '*send_array*' and then it is transmitted using the RFduinoBLE.send() command.

4.4. Signal Processing & Algorithm Design

There are three main portions of the android application that we as the EE group have implemented. The remainder of the application was developed by our CS counter parts Shiza Tariq and Faiza Tahir from CS. The portions are based on the kind of work they perform and in the following sections we will attempt to explain the logic behind the code which is available at the end and in the CD included.

4.4.1. Electrocardiogram (ECG) signal processing

ECG signals are hard to deal with both in hardware and in software. Our aim as to reliably detect the QRS complex in the PQRST wave shape as shown below.

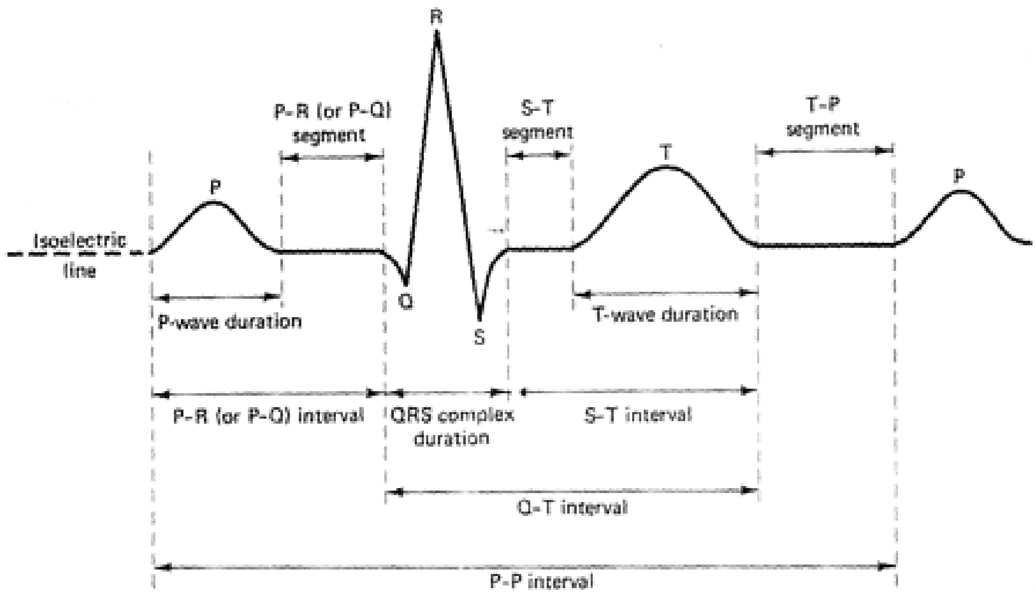


Figure 22 – ECG Signal Breakdown

Two of the most commonly used Algorithms are the Pan Tomkins algorithm and wavelet and Hilbert transform. Both of these algorithms target the detection of the R-peak or the QRS complex. We chose to go along with the Pan Tomkins version. While the a detailed paper is available on it there are no good reliable implementation of it in C++, Java or Matlab script. Hence to ensure proper implementation and functionality we chose to implement the algorithm first in Matlab and then in C++ to generate a reference class based framework which was then ported out to Java for use in Android application. It is easier to code in Java/Android if the algorithm is already in a class based implementation. C++ and Java share a very similar class hierarchy system.

4.4.1.1. Pan Tompkins Algorithm

It reliably recognizes QRS complexes based upon digital analyses of slope, amplitude, and width. It uses low thresholds, thereby increasing detection sensitivity. The algorithm automatically adjusts thresholds and parameters periodically to adapt to such ECG changes as QRS morphology and heart rate.

The algorithm can be divided into 4 phases as shown below:

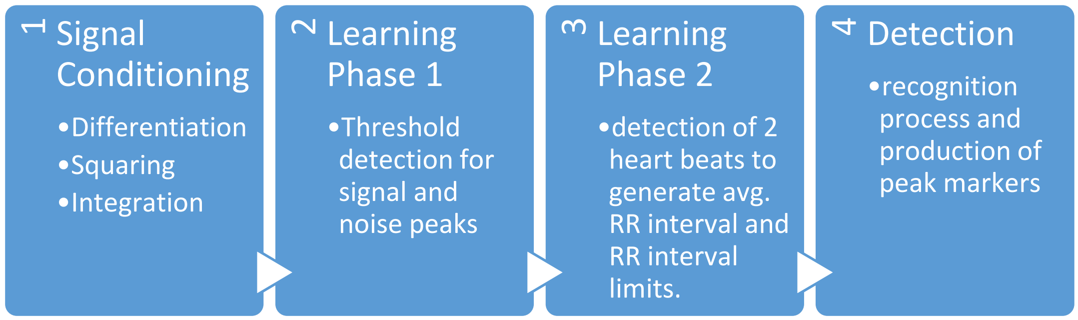


Figure 23 – Signal Conditioning and Detection

As mentioned earlier this algorithm uses two sets of dual thresholds. One set is used on the incoming ECG signal, and the other thresholds the output produced by integration. Since this thresholding is applied to two waveforms, it allows for more reliable QRS complex detection as compared to using one waveform alone.

The detection thresholds float over the noise that is sensed by the algorithm. This approach reduces the number of false positives caused by types of noise that mimic the characteristics of the QRS complex. It is also used to check for missed QRS complexes and therefore reduces the number of false negatives too. The following subsections explain in detail the math and threshold formulas along with the detection logic.

1. Differentiation / Derivative Filter

The Incoming signal is differentiated to obtain the slope information of the QRS complex. Like any other operation on a signal this too can be defined as filter, whose coefficients are generated with which the incoming waveform is convolved to produce the derivative. For this purpose, a 5-point formula has been used whose transfer function is as follows:

$$H(z) = \left(\frac{1}{8T}\right)(-z^{-2} - 2z^{-1} + 2z^{-1} + z^{2})$$

The difference equation corresponding to this is:

$$y(nT) = \left(\frac{1}{8T}\right)[-x(nT - 2T) - 2x(nT - T) + 2x(nT + T) + x(nT + 2T)]$$

2. Signal Squaring

The output of the derivative filter is squared point by point. This is a nonlinear amplification procedure that emphasizes the higher frequencies (P wave, QRS complex and T wave) and makes the output data points positive. The difference equation is as follows:

$$y(nT) = [x(nT)]^2$$

3. Moving Window Integration

As is with the case of the derivative filter moving window integration can be performing via convolution. The difference equation is as follows:

$$y(nT) = \left(\frac{1}{N}\right)[x(nT - (N - 1)T) + x(nT - (N - 2)\,T) + \cdots + x(nT)]$$

Here N is the number of samples in the integration window.

The purpose of moving-window integration is to obtain waveform feature information in addition to the slope of the R wave. The figure below shoes the relationship between the output of the integration filter and the QRS complex:

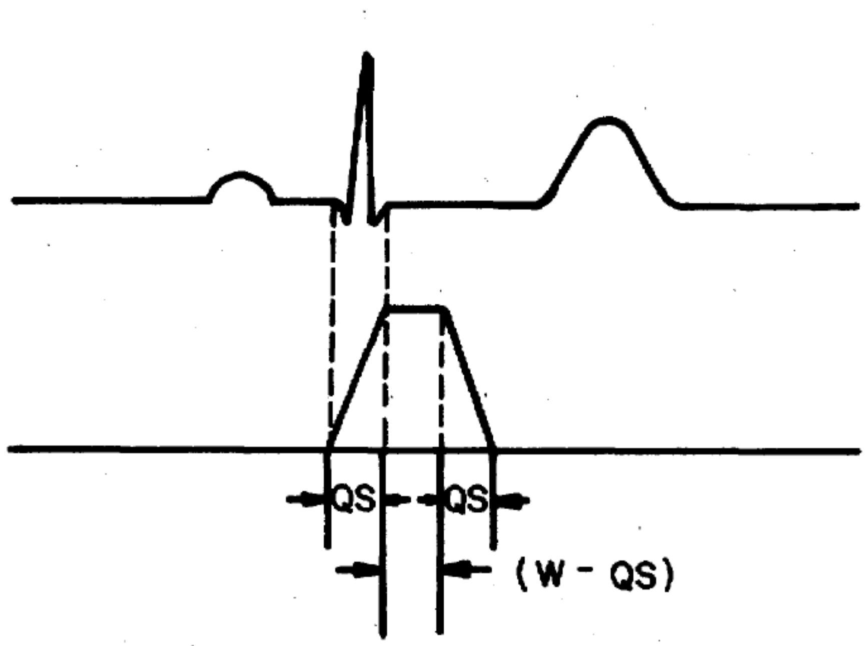


Figure 24 – QRS Complex

The QRS complex corresponds to the rising edge of the integration waveform. Hence the time duration of the rising edge corresponds to the width of the QRS complex i.e. the QS- interval. A fiducial marker for the location of the R-peak can be placed by locating the point of maximum slope.

The N as mentioned earlier is quite important. Generally speaking, the number of samples N should correspond to a time frame that is equal to the widest of the widest QRS complex. However, if the window is too wide then the QRS complex and the next dominant wave i.e. T wave will merge together in the integration waveform. On the other hand, if it is too narrow then some QRS complexes will produced multiple peaks. These can cause problems is the QRS detection process. An ideal windows size for most cases is about 150 ms which at a data rate of 250 Hz corresponds to $N = 37.5 \approx 38$ samples.

4. Adjusting the Thresholds

The thresholds are automatically adjusted to float over the noise. For the first analysis of the signal the higher of the two thresholds is used. The lower one is used if a QRS complex has been missed or hasn't been detected in a certain time interval so that a search back function can be triggered to look back in time for the QRS. The threshold formulas are as follows:

$$SPKI = 0.125\ PEAKI + 0.875\ SPKI \quad (if\ PEAKI\ is\ signal\ peak)$$
$$NPKI = 0.125\ PEAKI + 0.875\ NPKI \quad (if\ PEAKI\ is\ noise\ peak)$$

$$THREASHOLD\ I1 = NPKI + 0.25\ (SPKI - NPKI)$$
$$THREASHOLD\ I2 = 0.5\ THREASHOLD\ I1$$

where all the variables refer to the integration waveform:

- PEAKI is the overall peak
- SPKI is the running estimate of the signal peak
- NPKI is the running estimate of the noise peak
- THRESHOLD Il is the first threshold applied
- THRESHOLD 12 is the second threshold applied.

The signal peak SPKI is a peak that the algorithm has already established to be a QRS complex. The noise peak NPKI is any peak that is not related to the QRS (e.g., the T wave). The thresholds are based upon running estimates of SPKI and NPKI.

Whenever a new peak is found, it must be classified either as a signal peak or a noise peak. To qualify as a signal, the amplitude must be greater than THRESHOLD I1 (if the it is being analyzed for the first time) and THRESHOLD I2 if a search back was triggered. If the second threshold was used, then the update will be:

$$SPKI = 0.25\ PEAKI + 0.75\ SPKI$$

The second set of thresholds are applied to the incoming waveform and are calculated as follows:

$$SPKF = 0.125\,PEAKF + 0.875\,SPKF \quad (if\ PEAKF\ is\ signal\ peak)$$

$$NPKF = 0.125\,PEAKF + 0.875\,NPKF \quad (if\ PEAKF\ is\ noise\ peak)$$

$$THREASHOLD\ F1 = NPKF + 0.25\,(SPKF - NPKF)$$

$$THREASHOLD\ F2 = 0.5\,THREASHOLD\ F1$$

where all the variables refer to the incoming waveform:

- PEAKF is the overall peak
- SPKF is the running estimate of the signal peak
- NPKF is the running estimate of the noise peak
- THRESHOLD Fl is the first threshold applied
- THRESHOLD F2 is the second threshold applied.

As was the case with the threshold set earlier if the QRS complex is detected using the second threshold the formula for updating is:

$$SPKF = 0.25\,PEAKF + 0.75\,SPKF$$

Now that criteria for peak detection in both waveforms has been explained, ***a QRS complex is detected whenever a peak is recognized in both the integral waveform and the incoming waveform.***

In case of irregular heart beat or an ECG waveform in which the R peak amplitude varies a lot, the detection needs be made more sensitive. For this purpose, we reduce the current thresholds by half.

$$THREASHOLD\ I1 = 0.5\,THREASHOLD\ I1$$

$$THREASHOLD\ F1 = 0.5\,THREASHOLD\ F1$$

5. Average RR Interval and Rate Limit Adjustment

Regarding the average RR intervals two different RR interval averages are maintained. The first one the simply the average of the 8 most recent RR intervals:

$$RR\ AVERAGE1 = \frac{1}{8}\sum_{i=1}^{8} RR_i$$

The Second one is based on RR intervals that fall between two specified limits named RR LOW LIMIT and RR HIGH LIMIT as shown below:

$$RR\ LOW\ LIMIT = 0.92\ RR\ AVERAGE2$$

$$RR\ HIGH\ LIMIT = 1.16\ RR\ AVERAGE2$$

$$RR\ AVERAGE2 = \frac{1}{8}\sum_{i=1}^{8} RR_i'$$

For a QRS complex to be considered as missing we use the following condition:

$$RR\ MISSED = 1.66\ RR\ AVERAGE2$$

Hence if the newly calculated RR interval is 66% more than the RR AVERGAE2 that was being maintained then we have a QRS complex. Hence in this case the maximal peak reserved between the two established thresholds is considered to be a QRS candidate.

Now if the RR AVEGRAE1, which was calculated from the 8 most recent RR intervals falls between the RR LOW LIMIT and RR HIGH LIMIT, then the heart rate is considered to be regular and if in addition we have the following condition, the sinus rhythm too is normal:

$$RR\ AVERAGE1 \approx RR\ AVERAGE2$$

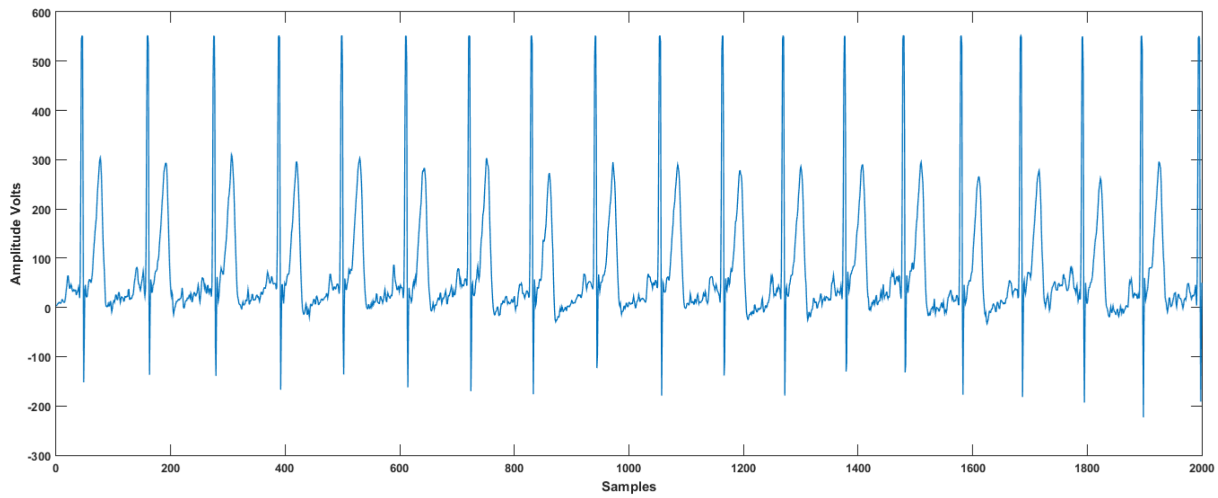
6. T-Wave Detection

T waves often cause problem in detection of QRS complexes because they too have significant amplitude and in some cases might even exceed the ARS complex itself. If the RR interval comes out to be less than 360 ms (which is 90 samples at 250 Hz) then we need to determine whether the particular wave is a QRS complex or a T wave. For this purpose, slope analysis is used. If the max slope (which corresponds to a peak) of the current peak location is less half of the previous peak location (which was a known QRS complex) then the current peak is a T wave and not a QRS complex; otherwise it is a QRS complex. This is quire evident just by looking at the ECG waveform, the T-wave does have a significant amplitude but it's rate of change is far lower than a QRS complex

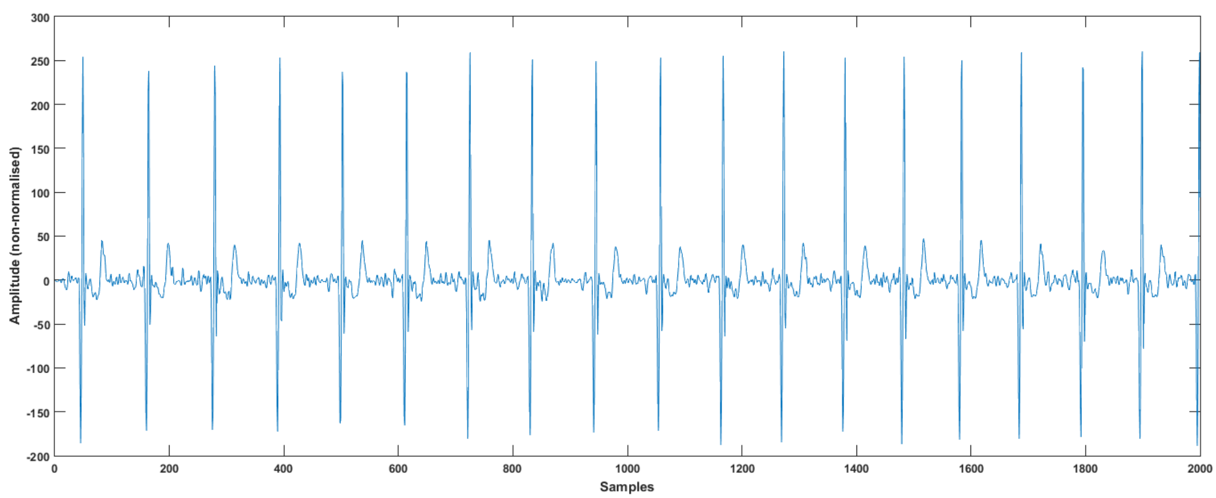
4.4.1.2. Pan Tompkins Implementation Results

The image below show the various waveforms obtained at different steps of the algorithm. The ECG shown is of one of the group members of this project Usama Perviaz. In order the images below show:

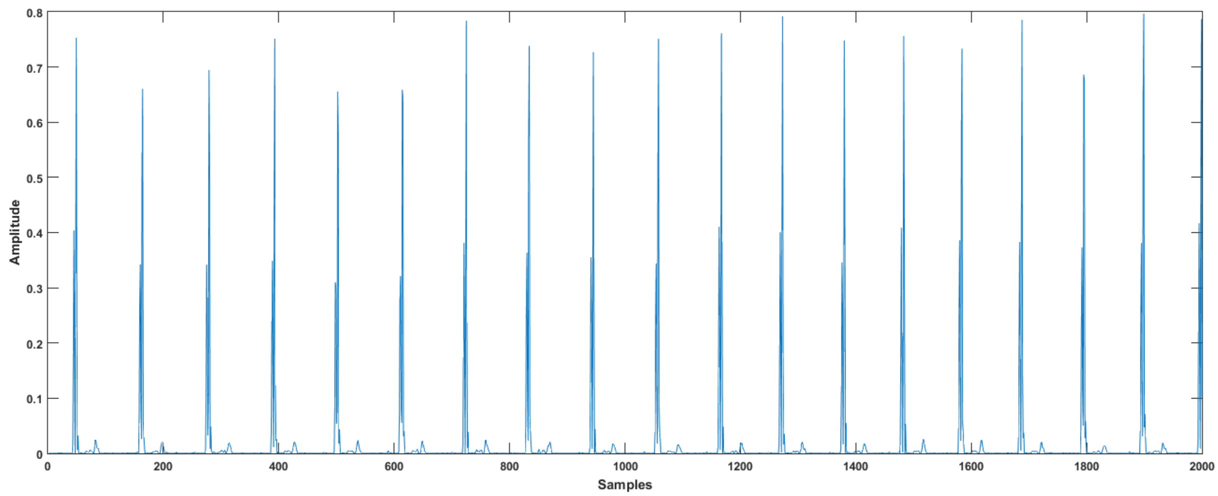
- the incoming ECG waveform
- the derivative filter output
- the squared version of the normalized derivate
- the output of the moving window integration
- fiducial marks of peaks in integral waveform
- the peak location markers drawn to amplitude
- the markers overlapped with the original ECG wave



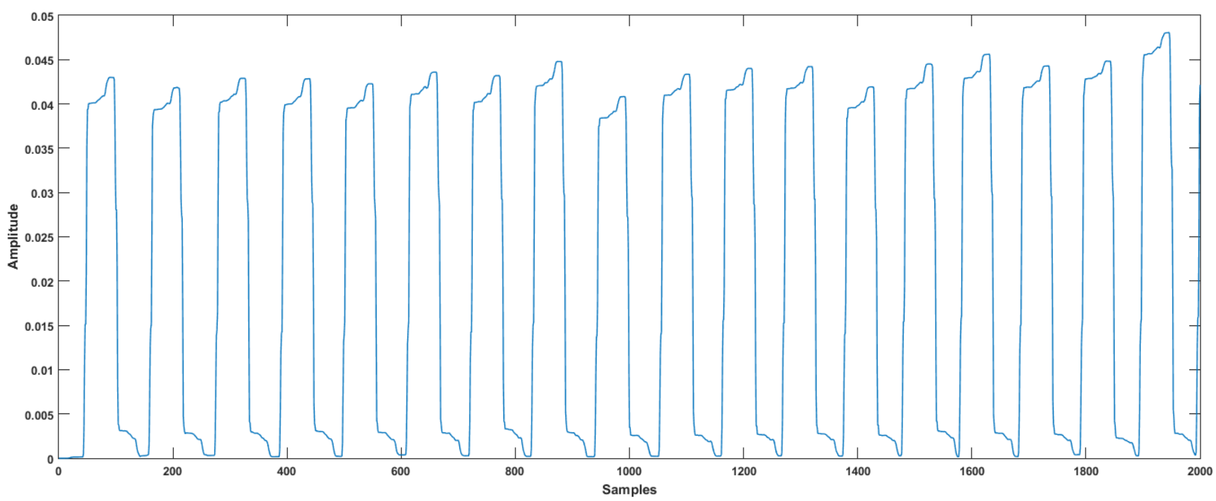
Amplitude Volts
Samples



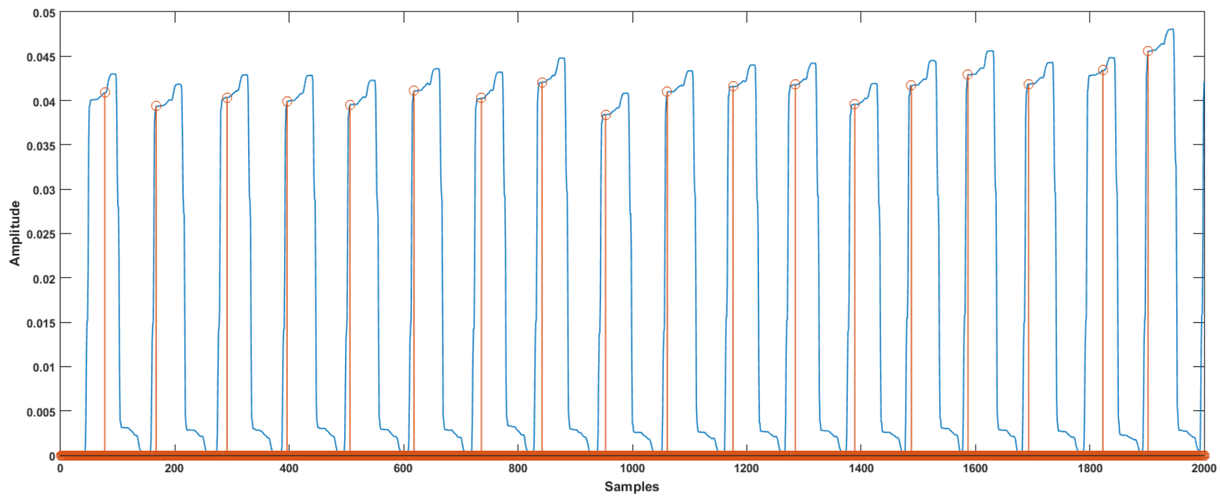
Amplitude (non-normalised)
Samples



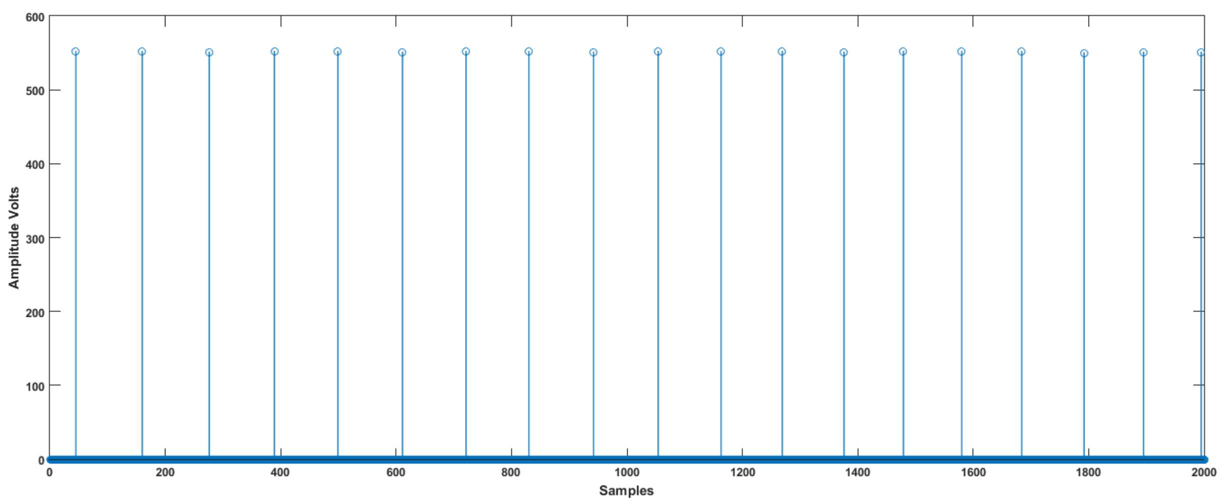
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Amplitude
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Samples



0.05
0.045
0.04
0.035
0.03
0.025
0.02
0.015
0.01
0.005
0
Amplitude
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Samples



Amplitude
Samples



Amplitude Volts
Samples

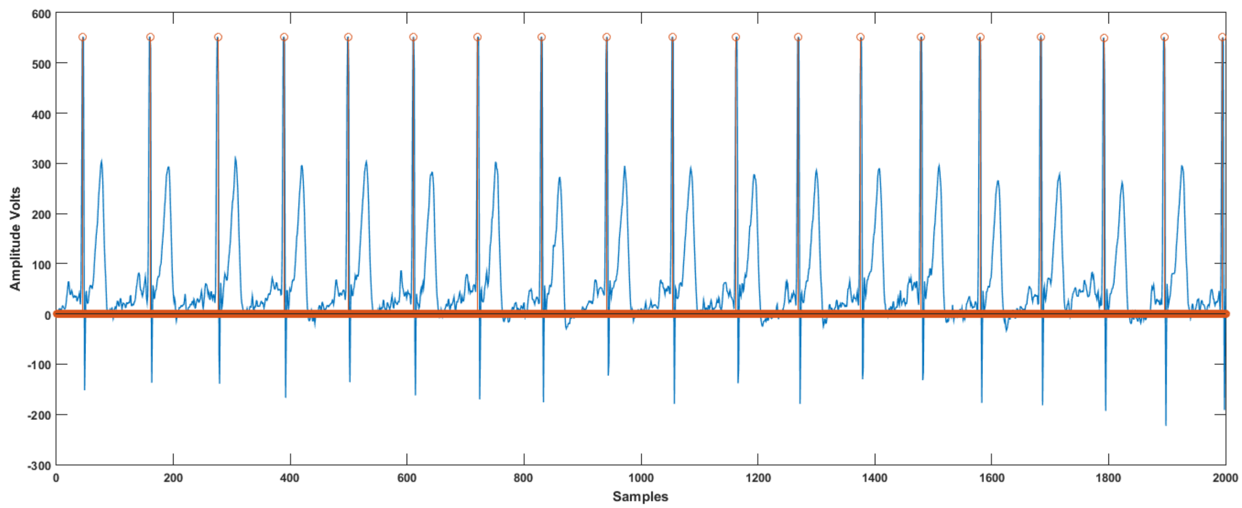


Figure 25 – Pan Tompkins Results (a,b,c,d,e,f,g)

In order to validate the performance of our algorithm implementation we put it to test against the MIT BIH database which contains ECG of a number of patients. The table below show's the performance for not only MIT BIH samples but also for our self-recorded ECG.

	total QRS	Detected QRS	Missed QRS
MIT-BH 100	2273	2273	0
MIT-BH 101	1865	1855	10
MIT-BH 108	1736	1516	220
Self-recorded (Usama Pervaiz)	579	577	2
Self-recorded (Patient 1)	1288	1280	8
Self-recorded (Patient 2)	1124	1112	12
Total	**8865**	**8613**	**252**
Total %	**100**	**97.16**	**2.84**

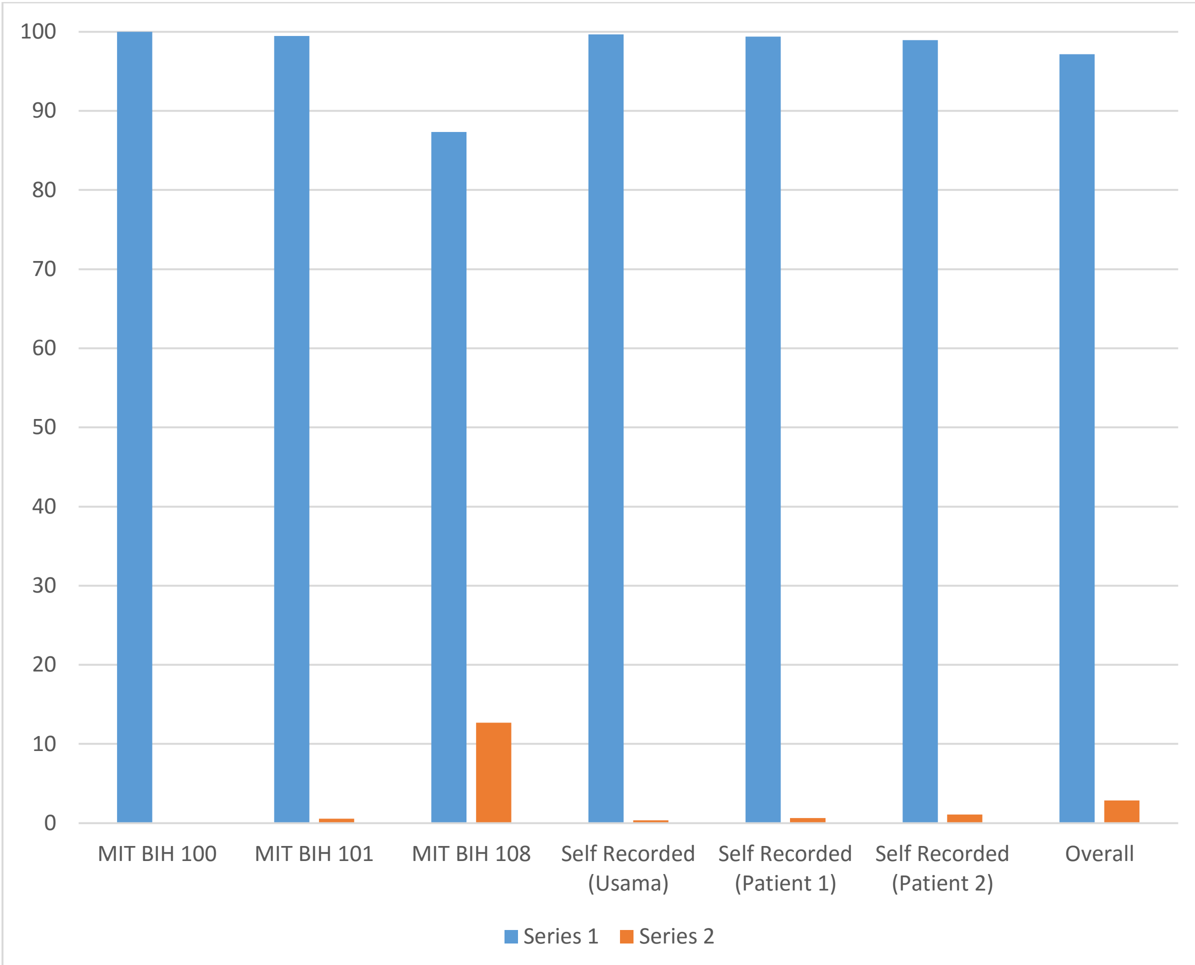


Figure 26 – ECG Evaluation Results of Pan Tompkins (a,b)

4.4.2. Activity Detection: Machine Learning

Activity Detection was a major obstacle to take on since for our application all the calculation was to be performed in real time and also needed to be doable in Java / Android.

Initially we had planned out that the activities would be classified in the following classes:

- Running

- Walking
- Upstairs
- Downstairs
- Idle

However once the project was pitched to doctors at Rawalpindi Institute of Cardiology, it was decided that distinction between upstairs and downstairs was redundant. Hence Upstairs/ Downstairs were merged into one class aptly named Stairs. The following sub sections will explain the entire process spanning from data collection to final implementations:

4.4.2.1. Raw Data Collection

The first and perhaps the most important thing to consider is the placement and the orientation of the accelerometer on the body. The image below shows both placement and the orientation in a single diagram.

Figure 27 – Accelerometer Axes

After the sensor was placed as shown above, the participants performed the basic set of activities in a controlled environment without any distraction to ensure that the date being collected is representative. The data rate was set at 250 Hz. There were 12 participants combined from both genders from whom the data was collected. The following routine was adopted:

- Walk in a straight line at normal/moderate pace for 60 secs
- Run for the same length as was walked (time variable)
- Descend a curved flight of stairs
- Ascend a curved flight of stairs
- Descend a straight flight of stairs
- Ascend a straight flight of stairs
- Idle work:
    - Standing while doing nothing
    - Sitting while doing nothing
    - Work on laptop while sitting
    - Lay on sides

This whole routine has generated a total of 1292 instances of feature vectors for training our desired algorithm.

4.4.2.2. Feature Selection and Model Training in Matlab

The most crucial aspect of machine learning which can make or break the classifier is feature selection. It’s a long tedious process with somewhat trial and error and requires a lot of iterations. Initial we had a list of 15 features as shown below:

Average(X)   Average(combined)   corr(XY)   corr(XZ)   corr(YZ)

Min(X)   Max(X)   Skewness(X)   Kurtosis(X)   Std.dev(X)

Sum(Mag < P_{25})   Sum(Mag < P_{75})   MaxFreq(X)

Sum(FFT peaks < 5Hz) Number(FFT Peaks above threshold)

Even though the final classifier was not implemented in Matlab, the Matlab Machine Learning toolbox played a big role in entire process of classifier design, particularly in feature selection. Now, the Matlab Machine Learning toolbox consists of the Classification Learner App which is basically a GUI which lets you test a variety of classification algorithms without any coding. However, an array consisting of both feature vectors and the corresponding labels need to be calculated beforehand and then imported.

To start off we chose a simple classifier know as Kth nearest neighbor or KNN; most of the feature selection was done in this via iterative process. First the performance indication method was set to 12-fold cross validation because this method of classifier verification gives a realistic view of how the classifier will perform in real time, then the algorithm was trained using all the features available. Now, one by one the features were removed and the classifier retrained. If the classifier prediction accuracy stayed constant or increased by removing a particular feature, then that feature was removed. On the other hand, if the accuracy decreased by removal of a particular feature then that feature was allowed to remain. By the end of this iterative process only 7 features remained out of initial 15 all while maintaining a higher accuracy than at the start. The features that were selected at the end are:

- Co-relation X and Y axis
- Co-relation Y and Z axis
- Minimum value along X axis
- Kurtosis X axis
- Skewness X axis
- Standard Deviation X axis
- Sum of aggregate acceleration A_i, if A_i < 25th Percentile i.e.

  $(\sum_{i=0}^{n} A_i)$ where $A_i = \sqrt{accX_i^2 + accY_i^2 + accZ_i^2}$

4.4.2.3. Classifier Performance in Matlab

Even though most of the feature selection was done on KNN classifier the highest accuracy that was achieved in 12 cross fold test or any test at that was 88.8 % with majority of error coming from misclassification of Upstairs/Downstairs as Walking. Shown below are the:

- App interface the KNN model trained followed by
- Confusion Matrix: Per True Class
- Confusion Matrix: Per Predicted Class
- Confusion Matrix: Overall

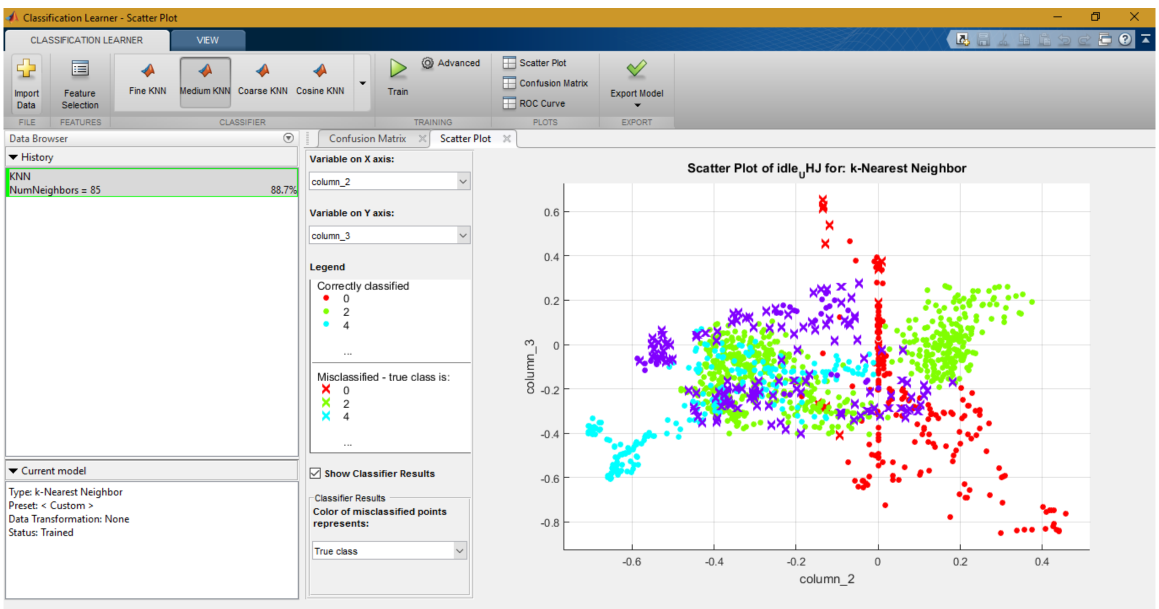
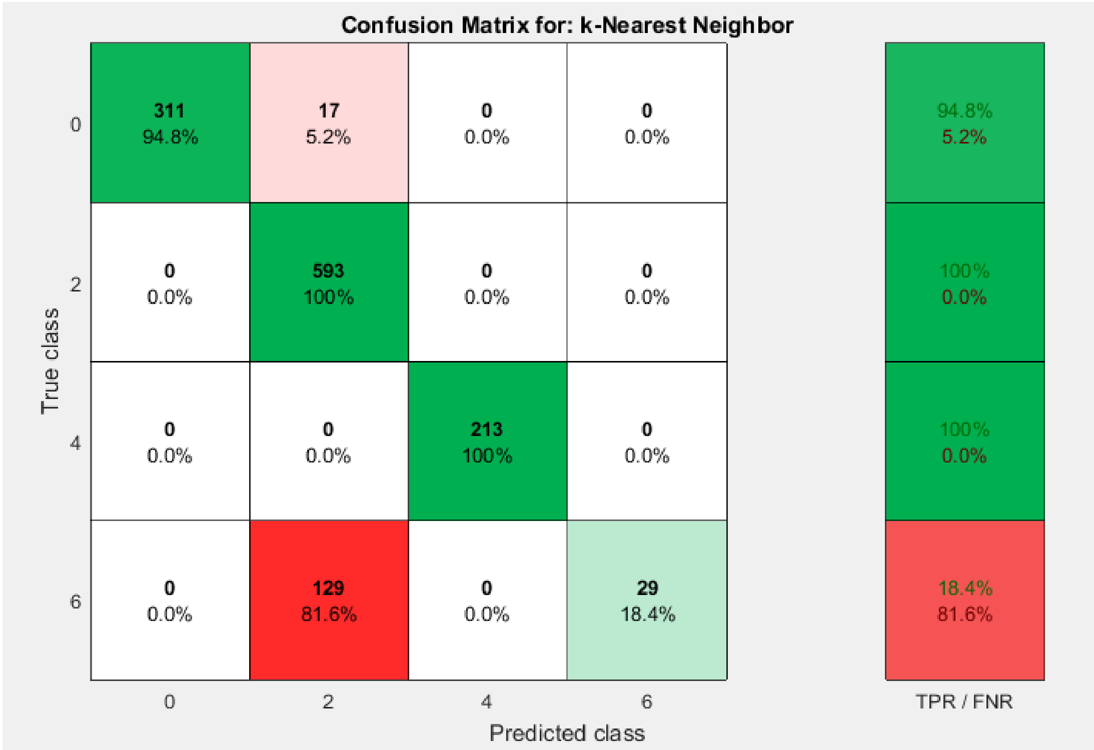
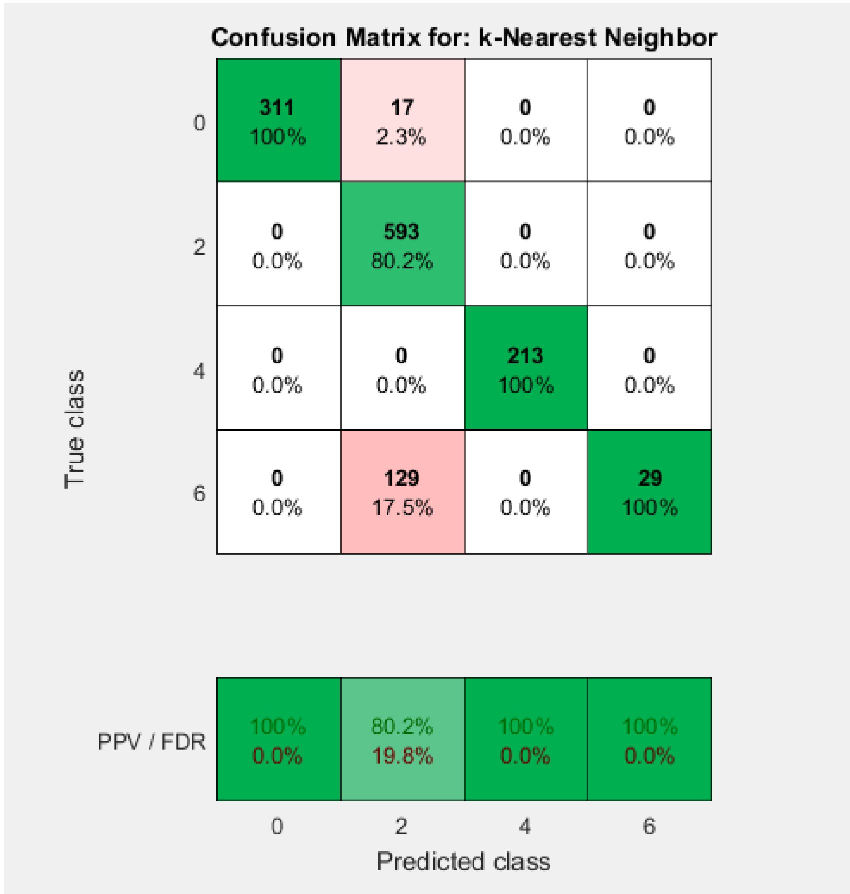


Figure 28 – Activity Detection Classifier Performance


Confusion Matrix for: k-Nearest Neighbor
311
94.8%
17
5.2%
0
0.0%
0
0.0%
0
0.0%
593
100%
0
0.0%
0
0.0%
0
0.0%
0
0.0%
213
100%
0
0.0%
0
0.0%
129
81.6%
0
0.0%
29
18.4%
94.8%
5.2%
100%
0.0%
100%
0.0%
18.4%
81.6%
True class
0
2
4
6
Predicted class
TPR / FNR


Confusion Matrix for: k-Nearest Neighbor
True class
0
2
4
6
311
100%
17
2.3%
0
0.0%
0
0.0%
0
0.0%
593
80.2%
0
0.0%
0
0.0%
0
0.0%
0
0.0%
213
100%
0
0.0%
0
0.0%
129
17.5%
0
0.0%
29
100%
PPV / FDR
100%
0.0%
80.2%
19.8%
100%
0.0%
100%
0.0%
0
2
4
6
Predicted class

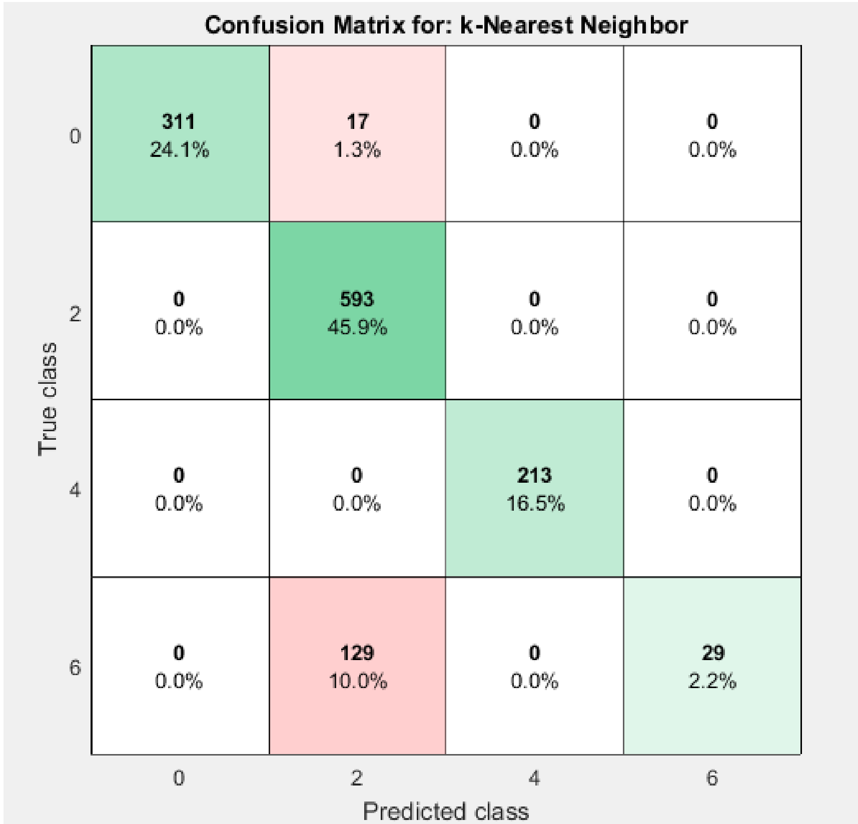
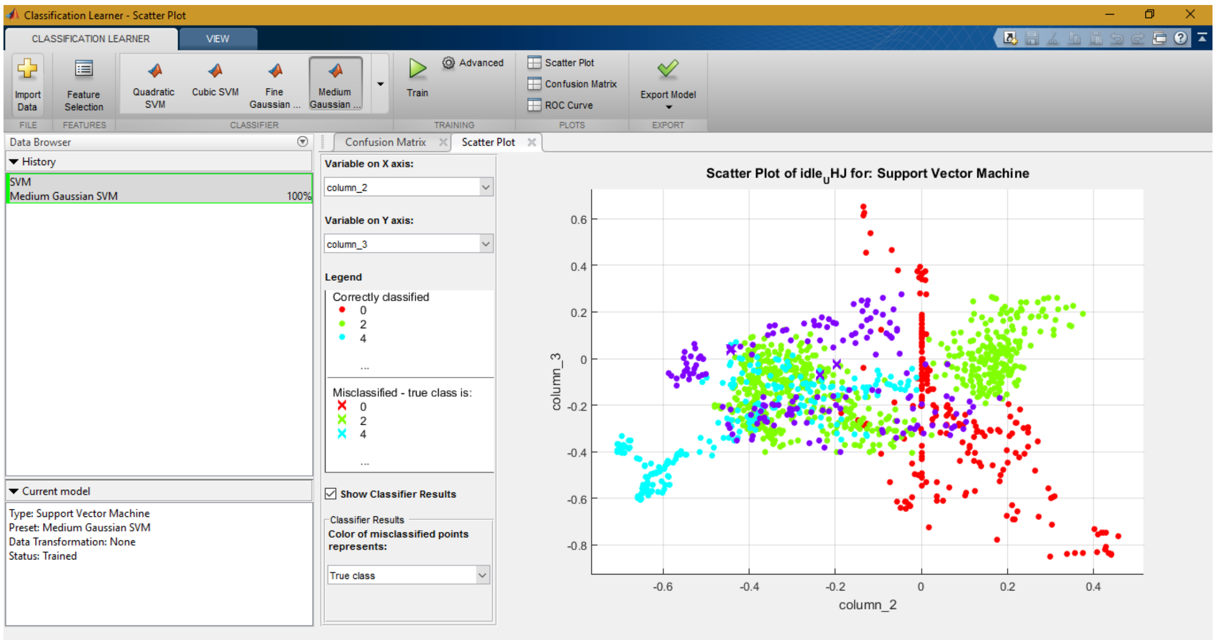


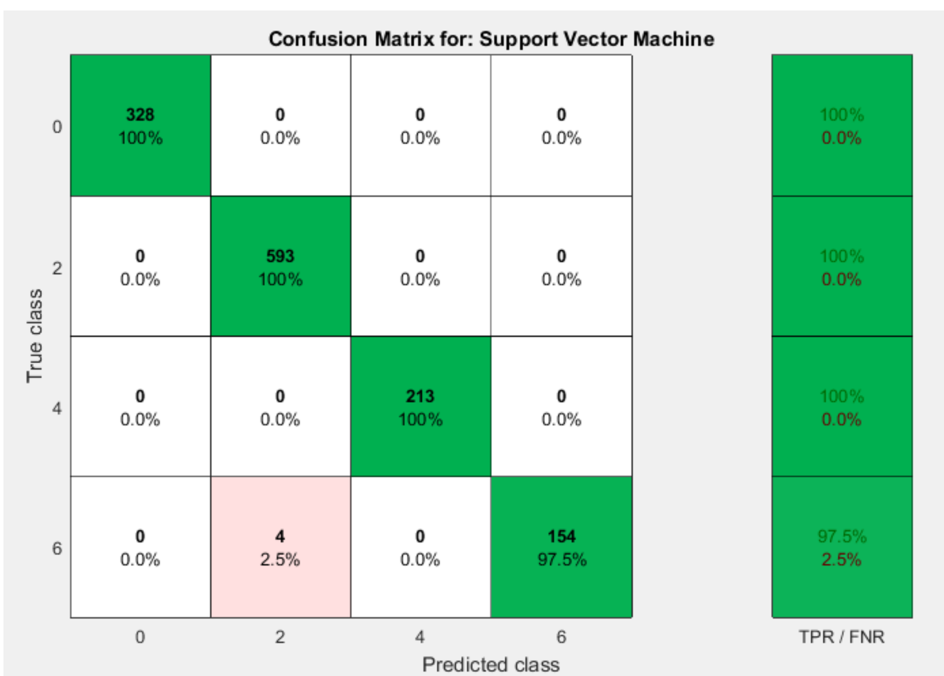
Figure 29 – Activity Detection Classifier Confusion Matrices (a,b,c)

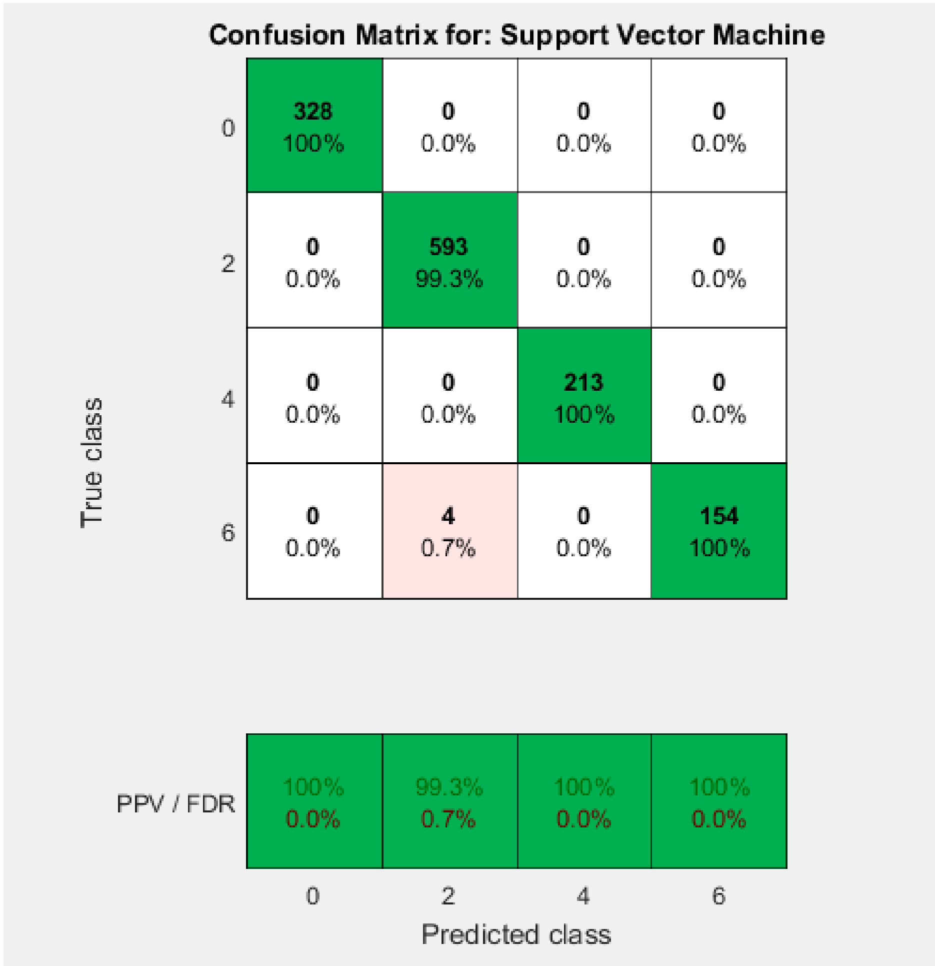
To increase the accuracy further it was evident that a more sophisticated classifier had to opted for. Hence, we moved onto the ever popular Support Vector Machine Implementation. We did try out a lot of different kernel shapes with varying parameters, however the RBF kernel / the Gaussian kernel produces the highest accuracy for any test performed be it hold out or cross fold. Though the first screenshot indicates that the classifier has 100 % accuracy, it is in fact not so! Actually Matlab has rounded off 99.7 % to 100 %. The error again was generated from Upstairs/Downstairs being classified as walking which is understandable since both are the same genre of actions unlike running or jogging. Following images again show:

- The app screenshot with the Gaussian kernel based SVM

- Confusion Matrix: Per True Class
- Confusion Matrix: Per Predicted Class
- Confusion Matrix: Overall







Confusion Matrix for: Support Vector Machine
True class
0
2
4
6
328
100%
0
0.0%
593
99.3%
213
100%
4
0.7%
154
100%
PPV / FDR
100%
0.0%
99.3%
0.7%
Predicted class

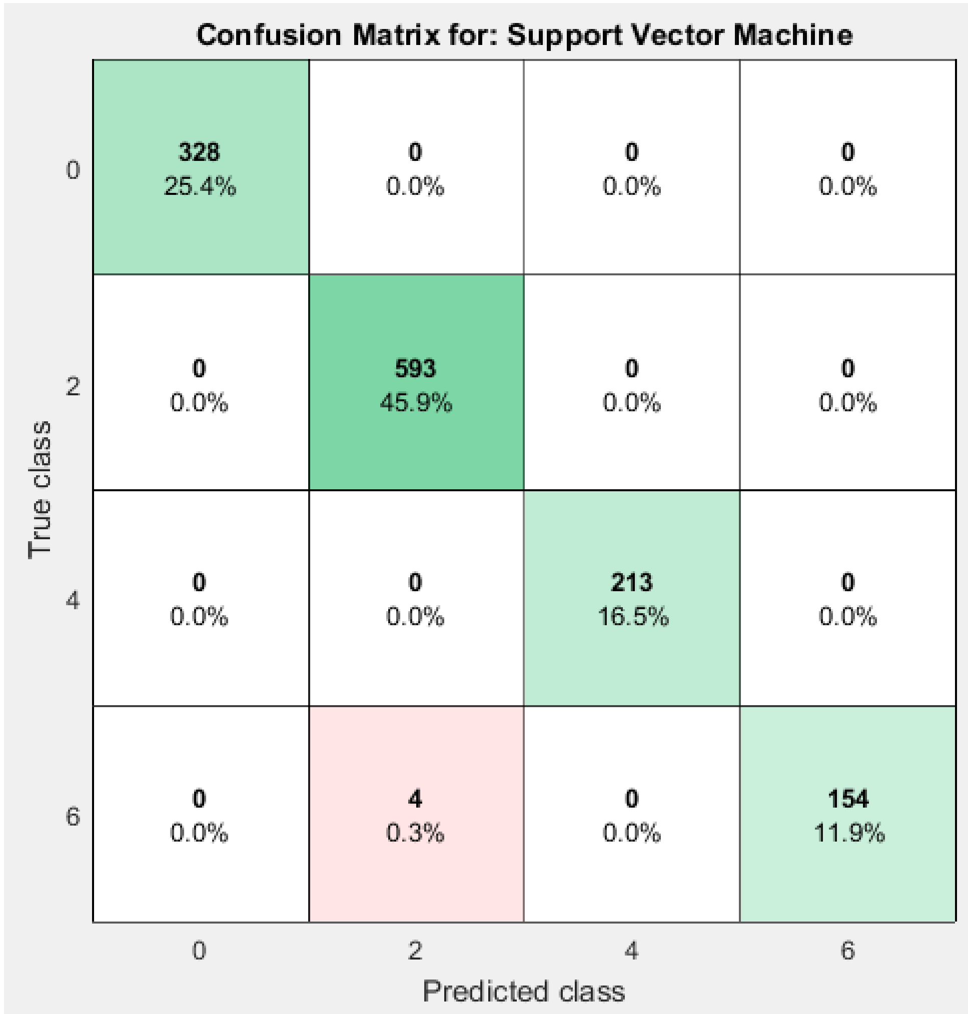


Figure 30 – Activity Detection Classifier Confusion Matrices(d,e,f)

4.4.2.4. Feature Calculation in Java

For Feature calculation in Java we've used the Apache Commons Math Library which contains most of the functions required.

1. **Correlation XY and Correlation YZ**

   The formula used to calculate Correlation is derived from Covariance as follows:

$$Covar(A,B) = \frac{\sum[(A_i - mean(A))(B_i - mean(B))]}{n-1}$$

$$Co.rel(A,B) = \frac{Covar(A,B)}{std.dev(A)\ std.dev(B)}$$

2. **Minimum value of X-axis**

minX=X[0]

For i=0 till i < length(X)

If X[i] <= minx

minX = X[i]

End

End

3. **Kurtosis X axis**

For Kurtosis an *'unbiased non-excess population'* formula is used:

$$3 + \frac{n(n+1)}{(n-1)(n-2)(n-3)} \frac{\sum(X_i - mean(X))^4}{(std.dev(X))^4} - \frac{3(n-1)^2}{(n-2)(n-3)}$$

4. **Skewness X axis**

$$\frac{n}{(n-1)(n-2)} \frac{\sum(X_i - mean(X))^3}{(std.dev(X))^3}$$

5. **Sum of aggregate acceleration Ai < P_{25}**

For i=0 till i < length(X)

Mag[i]=sqrt(X[i]^2 + Y[i]^2 + Z[i]^2)

End

Sum=0

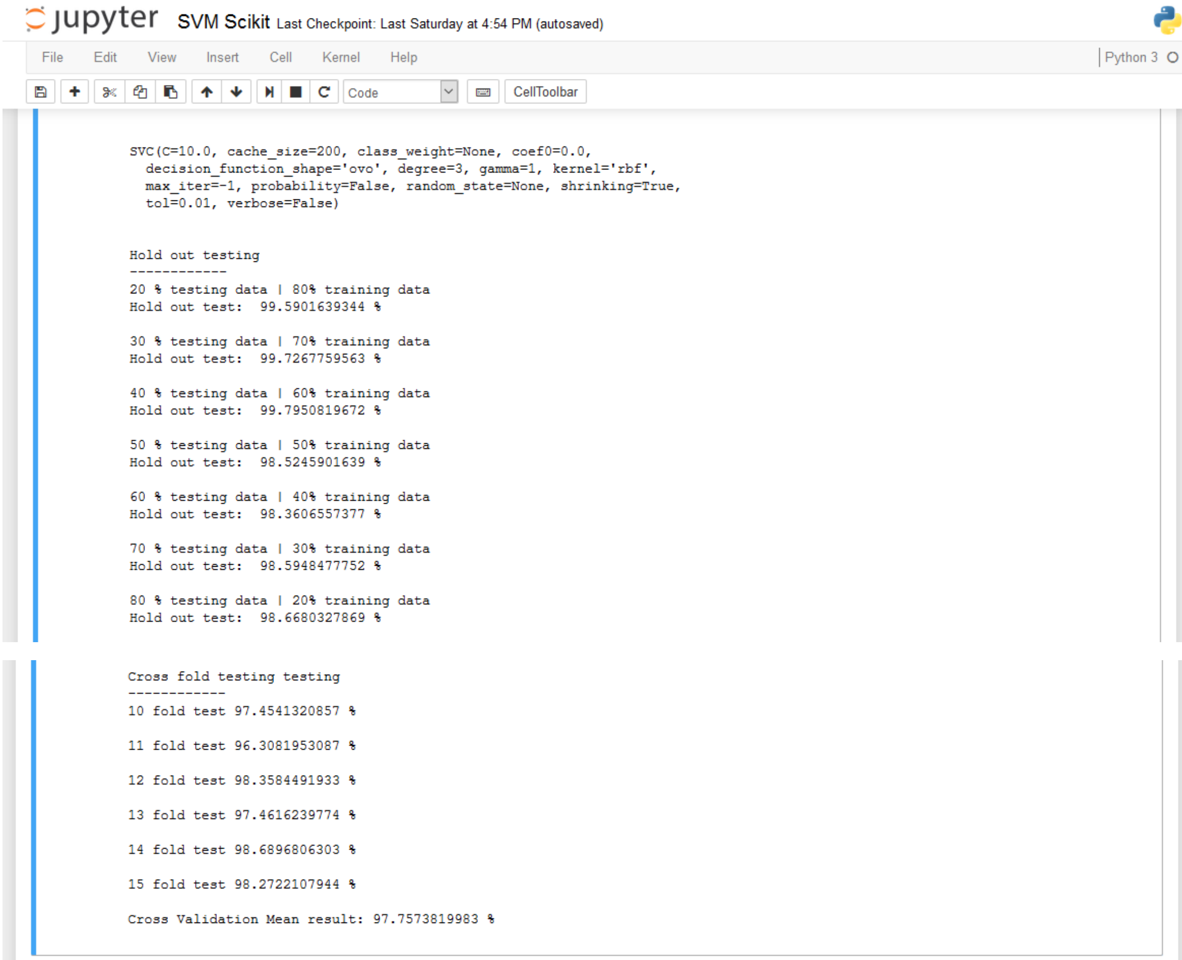
P_{25}=percentile(mag,25)

```
For i=0 till i < length(Mag)
    If Mag[i] <= P25
        sum = sum+Mag[i]
    End
End
```

4.4.2.5. Model Training in Python using Scikit

The classifier model cannot be trained in Java using the feature vector dataset previously calculated in Matlab due to the lack of a good implementations of support vector machine and the framework necessary to test the implementation. Hence we've used Python to implement the model by using Scikit Learn Machine Learning Library which contain efficient implementation of a lot of classification algorithms including SVM. Following subsections will briefly explain the framework used to implement the classifier we had designed in Matlab outside of Matlab in a combination of Java and Python.

The images below show the screenshots of the Python scripts executed in an iPython notebook by treating the local machine as a server and running the kernel. The accuracy achieved here on 12-cross fold testing is a tad bit (only %) lower than that of Matlab but overall the result is very satisfactory.




```
SVC(C=10.0, cache_size=200, class_weight=None, coef0=0.0,
  decision_function_shape='ovo', degree=3, gamma=1, kernel='rbf',
  max_iter=-1, probability=False, random_state=None, shrinking=True,
  tol=0.01, verbose=False)


Hold out testing
------------
20 % testing data | 80% training data
Hold out test:  99.5901639344 %

30 % testing data | 70% training data
Hold out test:  99.7267759563 %

40 % testing data | 60% training data
Hold out test:  99.7950819672 %

50 % testing data | 50% training data
Hold out test:  98.5245901639 %

60 % testing data | 40% training data
Hold out test:  98.3606557377 %

70 % testing data | 30% training data
Hold out test:  98.5948477752 %

80 % testing data | 20% training data
Hold out test:  98.6680327869 %
```

```
Cross fold testing testing
------------
10 fold test 97.4541320857 %

11 fold test 96.3081953087 %

12 fold test 98.3584491933 %

13 fold test 97.4616239774 %

14 fold test 98.6896806303 %

15 fold test 98.2722107944 %

Cross Validation Mean result: 97.7573819983 %
```

4.4.3. Android Function Calling and Thread Management

Now that it is clear what the functions/algorithms are, it is time to explain how they're structured among one another and when what is called and how.

Values that are received form the RFD22301 are in a contaminated string which is in a Comma Separated Value (CSV) format encoded in ASCII. Hence the string is first converted from ASCII then is split on basis of the comma and then the string values are parsed into double type numbers which are stored in double type vectors (variable length arrays) named ecg_raw, accX_raw, accY_raw and accZ_raw. Based on the length of these vectors ECG and Feature calculation functions are called periodically.

For the case of ECG the function is triggered at length of 1500 for ecg_raw. If this happens then the peak detection algorithm is called and the results are returned. Before any new values can be appended to the current buffer i.e. ecg_raw, it is updated by removing the starting (old) 250 values to produce a new length of 1250.

In case of the activity we set the trigger at a length of 1024. This calls the Feature Calculation algorithm which after calculation uploads it's result to the server where the classification algorithm run in Python whose result is returned to the mobile app. As is with the case of the ECG buffer we updated the buffer before new values are appended to it i.e we remove the starting 250 values.

Both these operations are run in a background service in the Android Application in separate threads (parallel processes) so as to not cause the App to lag by overloading the work in the main thread. Hence Using this framework all calculation are performed in real time without any lag. The time between subsequent calculation i.e. the time to fill up the 250 values that were removed is 1 sec or 1000 ms. The ECG and the Feature Calculation algorithms take 17 ms and 10 ms to run respectively leaving us with 1000 – 17 – 10 = 983 ms (if they run sequentially (worst case)) to send and receive the values from the server script.

Chapter 5

5.0 DISCUSSION

5.1 ANDROID APP

The app required a user friendly and extremely convenient interface for the patients to be able to use it without any hindrance. It needed to monitor ECG in real time as well as processes it and store it for displaying it on doctor's end. The app requires calculations to process data both locally and on cloud as well. For this purpose the app had to be constructed carefully and with simplicity.

The API levels for bluetooth compatibility needed to be 22 and most of the libraries used supported API level 23, integrating both components was an arduous task. Next the app needed to be updated iteratively once we got feedback from our clinical trials to suit the patients and the doctors. Separate Apps were to be made for doctors and patients to cater for their separate needs, with patient's app having added features. Like reminders and diet tracking.

We chose graphview for displaying charts as it had a good front end and allowed us to have more features and an interactive graph. As the graph had to be displayed as continuous real time hence our options were limited to graphview and MP chart from a few others.

Chapter 6

6.0 CONCLUSION

The app along with a wearable device is a stepping stone to the advances needed in telemedicine, there is a dire need to fulfill the gap between healthcare facilities and patients. IoT as a domain has a lot of area for exploration, we chose telemedicine as to cater the needs of the society as a whole. The app has a lot of room for advances to be a compact platform for all kinds of heart patients not just heart failure patients.

It is the first solution in Pakistan that has doctors and patients on board and allows mobility of patients by wireless ECG monitoring, as well as remote access to the doctors. The doctors can gain a lot of insights from the data collected overtime which could lead to new foundings. Real time ECG monitoring could save lives, something which did not exist before. As compared to present solutions this is feasible and accurate solution as compared to alternate solutions like risk calculator or measuring heart rate through a camera’s flash.

Some of the insights gained from clinical trials shows the true essence of the app, it proved to be accurate to the traditional ECG monitor system. The heart rate and heart rate variability trends would lend an insight to the doctor about the patient, moreover activity detection and history is equally vital for doctors to correlate the anomalies in ECG as well as pre determine any unlikely incident. The app can be further used for all cardiac patients not just heart failure with extension of other modules which would run on the basis of the present one.

The app is not limited to heart patients as the project was initiated by Dr Azhar from NASA it is well apt for astronauts and other field works where the person has to be constantly monitored for his or her vitals. For the society this is a complete solution especially for a developing country the wearable and app are a low cost solution to other expensive alternatives, as well as feasible in terms of using the app for intended purpose, mobility of the solution and bridging the gap between patient doctor ratio which is very low in our country.

For purpose of commercialization of the project we have already partnered with Rawalpindi Institute of Cardiology after successful complete trials of the product we would be able to roll out the product for funding.

For now we were limited in terms of technical support and funding, both for front end and back end we needed financial assistance as AWS is not free of cost. The doctors were generally apprehensive however we were graciously welcomed at RIC through their support were we able to successfully complete our project.

Chapter 7

7.0 FUTURE RECOMMENDATIONS

As our clinical trials and doctor meetups went we gained further insights to the future work that could be done upon our present solution. Our app can be a solution for all heart patients by introducing relevant features that deal with different kinds of heart diseases. We are searching into medical advances like ICDs and adding features like caloric intake and BP monitoring all of which was recommended by the doctors and is said to greatly improve chances of survival for the patients. Our aim is to gather data overtime and detect anomalies in ECG for detecting conditions like arrhythmia and ventricular fibrillation.

Moreover future recommendation are to

1. Calculate ST interval elevation and respiratory rate from the ECG waveform.
2. In the activity detection portion, we can increase the data set further and can incorporate participants from all age.
3. 3. Clinical testing is a long & rigorous process and can take up to months to detect arrhythmia, so we suggest continuous monitoring of patients for a long period of time and to build a enhance data base for different types of arrhythmia.
4. The form factor of device can be improved further by double side PCB board and the 3D printing facility can be used for better packaging but this will also increase the overall cost of device.

Chapter 8

8.0 References

1. https://en.wikipedia.org/wiki/Body_area_network
2. http://www.who.int/trade/glossary/story021/en/
3. https://en.wikipedia.org/wiki/Internet_of_Things
4. http://www.analog.com/media/en/technical-documentation/data-sheets/ADXL345.pdf

5.http://www.analog.com/en/products/application-specific/medical/ecg/ad8232.html#product-overview

6. http://www.ti.com/product/LM35