**Heart Rate Detection Using A Piezoelectric Sensor**

Authors: Shane Mann | Adam Bohac | Jailyn Battle | Anya Ellis | Alvin Nnaube | Ranye Mclendon | Precious Isidahomen | Oluwafunmilayo Adediwura

Professors: Dr. Makarand Deo and Dr. Sacharia Albin

Norfolk State University

# 

# 

# Table of Contents

Title Page 1

Table of Contents 2

Abstract 3

Introduction 4

Project Objectives 4

Methodology 5

Results 12

Discussion of Results

Conclusions and Future Work

References

# 

# 

# 

# Abstract

The purpose of this project was to develop a wearable heart rate sensor that could be incorporated within a smart and connected city in order to monitor chronic heart disorders. There were 3 research areas that were investigated: design a wearable sensor that will collect heartbeat data using a piezoelectric sensor, develop a data filtering and analysis algorithm in MATLAB, and to design a smartphone Bluetooth interface with and Android application to acquire and process the data from the wearable sensor. This project was broken down into 3 subsections were members of each subsection were assigned one of the three specific research areas. During each stage of the project, the subsections worked independently on their assigned research area while meeting regularly to incorporate the work into the project as a whole. During the course of this project, we were able to successfully collect heart rate data using the piezoelectric sensor, create an algorithm in MATLAB to filter the data that was collected, and build an Android application to display the data from the sensor. Based on the results of this project, it can be concluded that the piezoelectric sensor is not a sufficient heart rate sensor. Furthermore, additional work would be needed within MATLAB to correctly analyze the data to accurately monitor heart conditions.

# Introduction

The purpose of this research was for us to be able to have a better understanding of how to monitor heart rates. Research shows that, “Heart disease is the leading cause of death for both men and women, in the United States about 610,000 people die of heart disease every year” [1]. Building this sensor would allow us to be able to determine whether or not a person has a heart disease, if we are able to detect heart disease as soon as possible then doctors would be able to give treatment before it is too late. An advantage of this project is the ability to make people self aware of their heart condition making it easier for people to take better care of themselves making this a cheap and easy way to incorporate into everyday life. This could also be useful for SMART City applications because a primary care physician can store live data, for a person or family, into a cloud based profile or external hard drive. With this research we also hope to introduce new skill sets for people to develop, encourage teamwork, promote creative thinking, and have a deeper understanding of heart sensors components and applications.

# Project Objectives

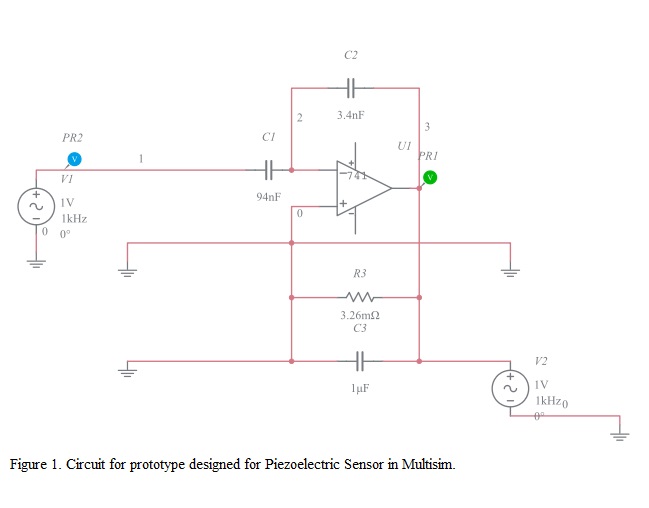
Our project objectives were to build a heart monitor using a piezoelectric sensor and a microcontroller. Our other objectives included creating a circuit, using MatLab to create an algorithm to filter noise, and to use a Bluetooth module to send data wirelessly. We hoped to achieve a wearable device that could detect a heart rate and develop a mobile app to display data from the piezoelectric sensor.

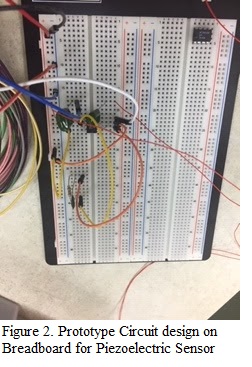
# 

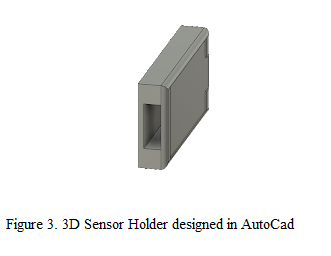
**Subgroup 1**

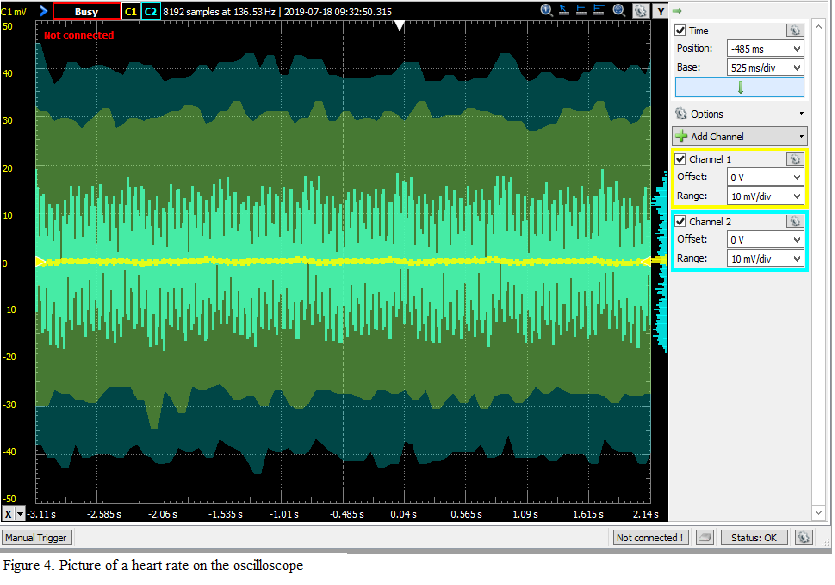
We decided to split our goal objectives by subgroups that included two students and one teacher. Our subgroup goals were to create a circuit capable of detecting a pulse. We chose three participants for wearing the sensor and collecting data. We focused on hardware to send to subgroup two for data collection. As well as, filtering and amplifying the signal, using an Arduino and creating a PCB.

The circuit was created and tested in Multisim and Autodesk Eagle. Afterwards, the circuit was used to make a PCB board that could fit three capacitors, a resistor, pins to be used for the analog discovery and an OpAmp. The analog discovery kit is a portable analog circuit design kit that uses waveforms to display signal data. After creating the circuit, we used the analog discovery kit to display the signal of the detected pulse.

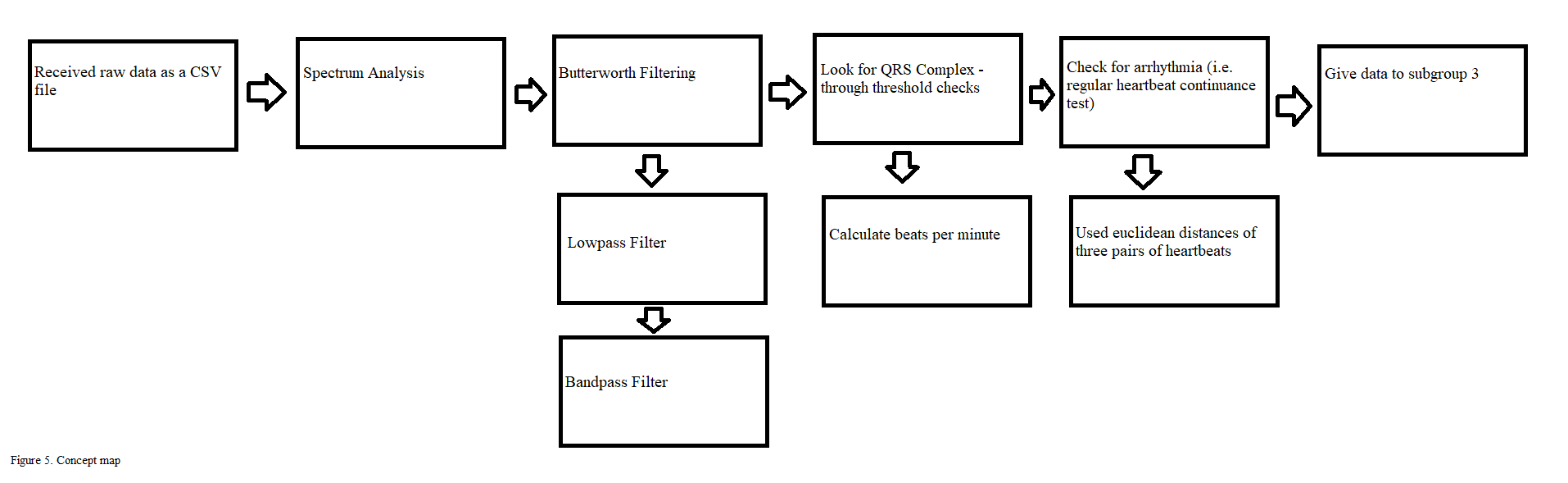




Autodesk Fusion 360 is a 3D modeling software used to create a piezoelectric sensor holder (Fig 3). After printing the sensor holder, velcro strips were to be inserted in the appropriate slots, then worn around the wrist to detect the participants pulse.

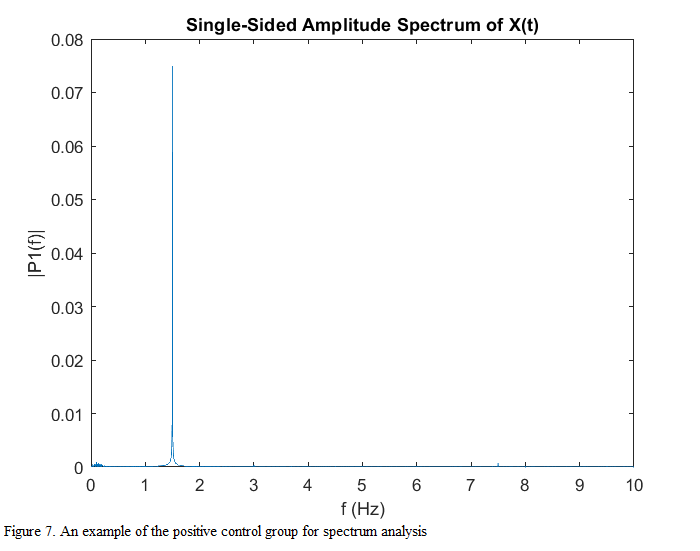
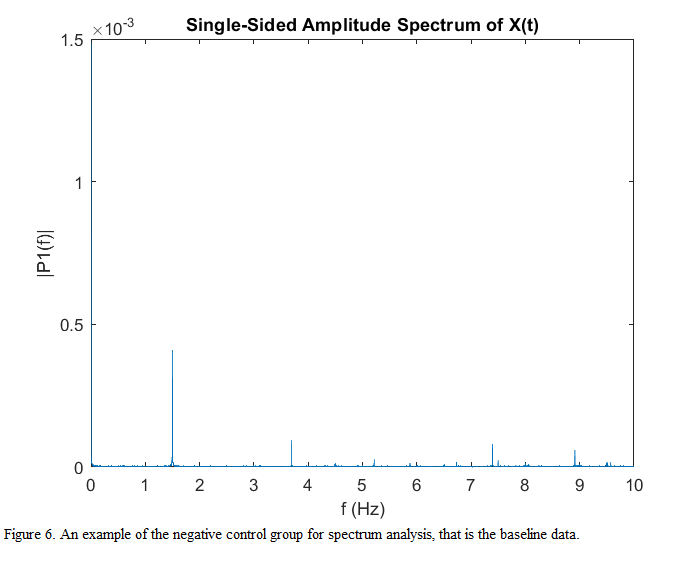
We used a combination of the WaveForms application and the Analog Discovery kit to measure a person’s heart rate. Once we connected the Analog Discovery to the circuit and a computer or laptop, we would then open the WaveForms application. After opening WaveForms we would turn on the Master Enable, which would turn on both the positive and negative voltage supply, then open the dual channel oscilloscope. We then connect the sensor to the circuit so that we can measure a person’s heart rate. We measured three different people, each person had to stay perfectly still while we measured their heart rate by placing the piezoelectric sensor on their wrist. We tested with a 10 second time period and we repeated this with five trials. 

**Subgroup 2**

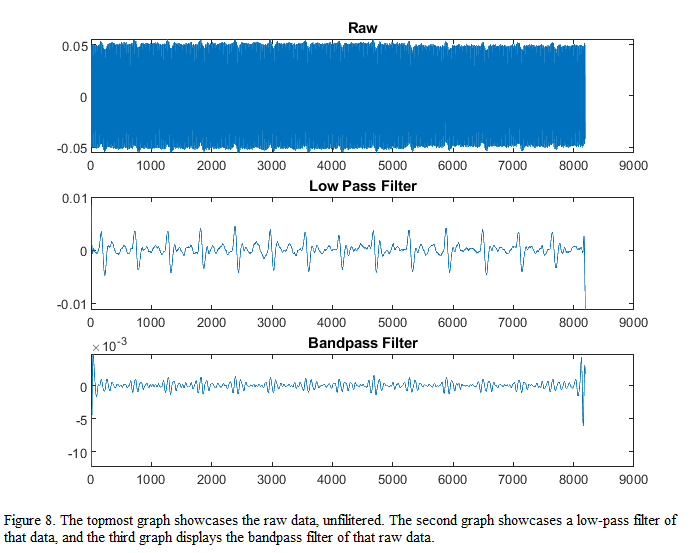
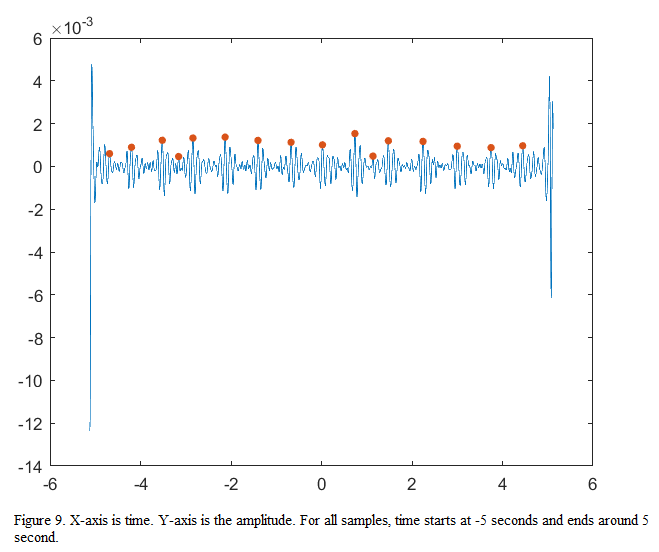
Our goals were to filter electrocardiogram (ECG) received from subgroup 1, detect and analyze QRS complexes, and pass this data onto subgroup 3. All code was written in MATLAB and we received the data as a CSV file, which can be found on GitHub.

In our first iteration of testing, each data sample was around 10 secs long. The data contained the time (in secs), raw (unamplified and unfiltered) sensor output (in Amps) and the amplified and filtered output (in Amps). Before filtering any data, we would chart the spectrum analysis. This is done through the fourier transform method, which changes time domain measurements to frequency domain (Hz).

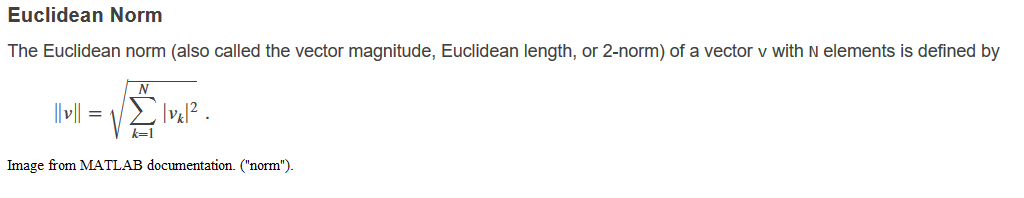
At this point, we came across our first challenge. Each data set, even the negative control group, presented a peak around the range which we expected the QRS complex to be present at. Due to this, we have assumed that two facts are true about the raw data: 1) the actual software or device used by subgroup 1 has a frequency around the range of the expected QRS complex and 2) that the QRS complex in the data is between 0 and 1 hrtz due to "phase shift" from the delay of processing live data.

Moving on, we performed a low-pass butterworth filter and a bandpass butterworth filter. Our reason for using butterworth filtering methods were because butterworth is very capable in flattening the passband (Morisak). After each filtering, we performed a zero-phase digital filtering to minimize changing transients due to the low-pass or bandpass filtering. 

The low pass filter on average would demonstrate an expected similarity to a PQRST wave (fig 8). At the moment, the bandpass filter only works for some test subjects.

Knowing that a heartbeat cannot possible happen within 200 ms of each other (J. Pan and W. J. Tompkins 231), we devised a QRS detection algorithm that finds peaks due to threshold analysis as well as the distance between peaks. The threshold analysis considers all peaks with the standard deviation for highest peaks in the filter data. Peaks that fit this test are marked on the graph. These marked peaks are used to calculate beats per minute of the subjects.

We continued by choosing three pairs of heartbeats within the marked peaks, and perform a euclidean distance test. This information is used to check the possibility for arrhythmia within the subject. If the distance between any two adjacent peaks are within the standard deviation of the euclidean distance, the test subject has a slim chance for arrhythmia. Otherwise, there is a higher possibility.



Sadly to say, this arrhythmia test is flawed. Since we only used three pairs, the analysis does not have enough data to work. This will be changed to four pairs or more in further studies.

**Subgroup 3**

We focused on the software aspect of this project, which was to design and develop an android application that uses the smartphone bluetooth interface to be able to acquire and process heart rate data.



We established bluetooth connectivity on the android app using Java and XML on android studio on an emulator and android phone. The application had three views; the home page which had a button that directed the user to a second page. The second view included button options to turn bluetooth on/off, get visible and show paired devices. Once devices were paired, this page showed these devices.

The final view displayed the result of test sinusoidal signals received as a csv file. We worked on a code to display the filtered data from subgroup 2 on the app. In the absence of the required bluetooth module (HC-05), we could not test the efficiency of the code.

We eventually decided to establish bluetooth connectivity to the android application using micro bits. Upon research, we found out that micro bits could only retrieve data using a micro bit app which defied the use of our android app.

We continued our research and decided to retrieve data through a csv file sent by the data filtering team via email. We graphed the data obtained through the csv using line graph on the android application.

# Results

**Subgroup 1**

The first type of sensor we used was a “home-made” sensor that was made by taking a small piezoelectric strip and attaching it to wires with aluminum tape and laminating it with clear tape. These sensors were not that reliable because they were made differently from other sensors so we switched to better sensors that were pre-made. These sensors allowed us to pick up and record better heartbeats on Analog Discovery.

Analog Discovery was used as a power source for the Active Band Pass filter circuit. By combining the Analog Discovery with the Active Band Pass filter circuit we were able to reduce the amount of noise by filtering frequencies below .5Hz and above 5Hz while amplifying the desired frequencies on the output end. This resulted in a clearly defined signal acquisition displayed on Waveshare.

Using the Arduino to acquire a signal was achievable however, it was not as defined as the first method. When we combined the Active Band Pass Filter with the Arduino it gave comparable results to the first method, but the circuit was to large to incorporate into a wearable device.

While the Arduino is capable of being coded to meet required frequency cut-offs and display the readings, the Arduino was unable to meet the 10V power requirements needed to power the Active Band Pass Filter. Furthermore, it does not have bluetooth capability without the addition of an adaptable module and additional coding.

Better overall project results came from the Micro:bit. This Microcontroller can be coded using multiple languages to meet the frequency cut-offs as well as displaying oscilloscope results on the monitor. The Micro:bit is also to display results with the onboard LED display. The compact size, weight, power requirements and BLE capabilities make it the best fit for this project. Due to the time constraints of the project we were unable to conduct enough testing and evaluation to confirm its capabilities.

The first wearable prototype that was created in AutoCad was only able to be worn vertically. In the second prototype holes were added in both the lengths and widths of the case so it could be worn on the wrist vertically or horizontally. Both prototypes had a small compartment in the bottom of the case so that the sensor would be able to come in contact with the users skin. The reason for this design would be so that it would have a better chance at picking up the heartbeat and calculating the heart rate.

## Subgroup 2

In the end, we created a threshold detection algorithm that was able to mark R-peaks. Hence, we could calculate the beats per minute in our 10-second samples. In addition, we designed the base code for arrhythmia testing. This will be the main subject of improvement in future work for our subgroup.

## Subgroup 3

We created the user experience design to test, research and develop our android application for quality results on Adobe XD. We were also able to connect the bluetooth in the android application to pair with other devices. In the absence of the bluetooth module, we took an alternative route which involved graphing test signals received in a csv file received through an email on the mobile application.

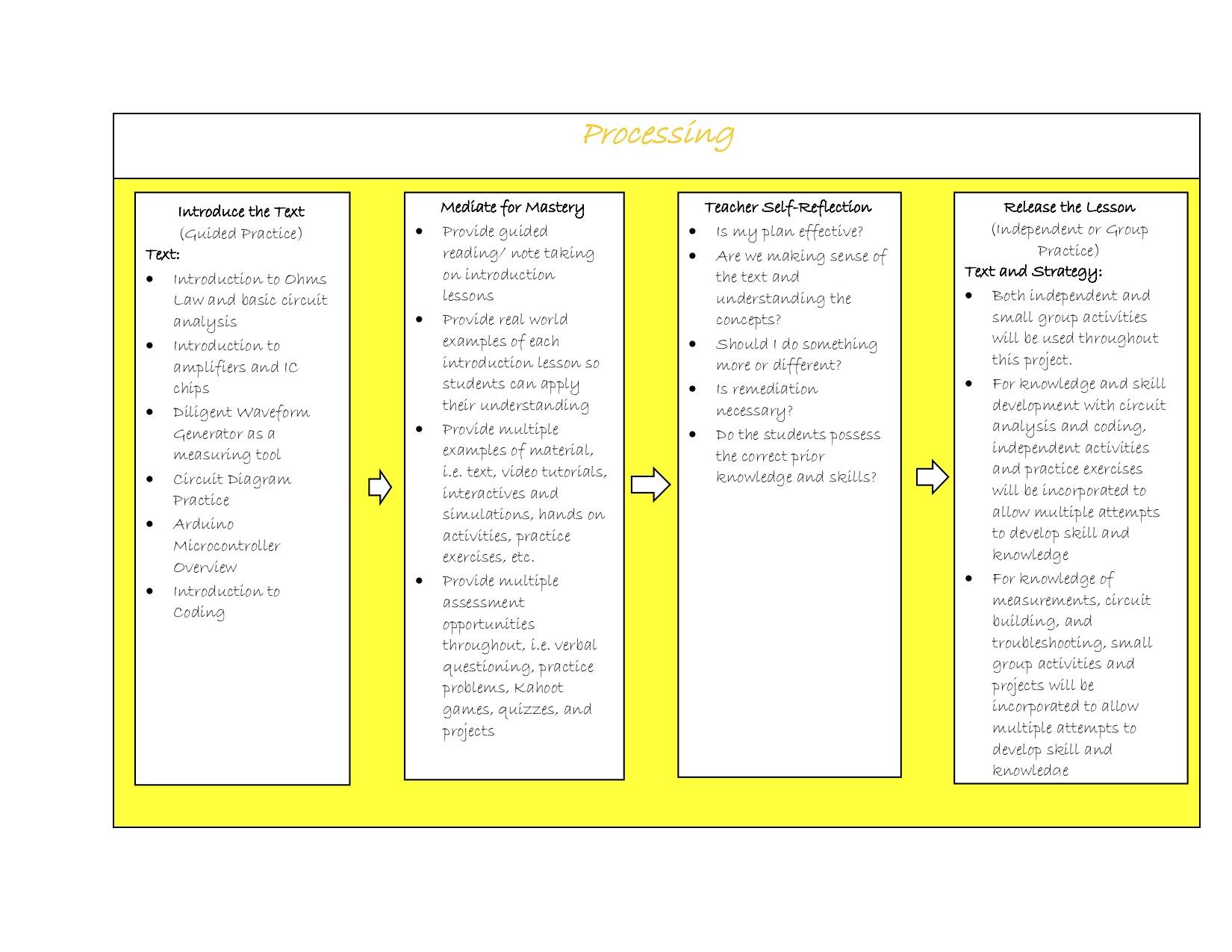
# Conclusions and Future Work

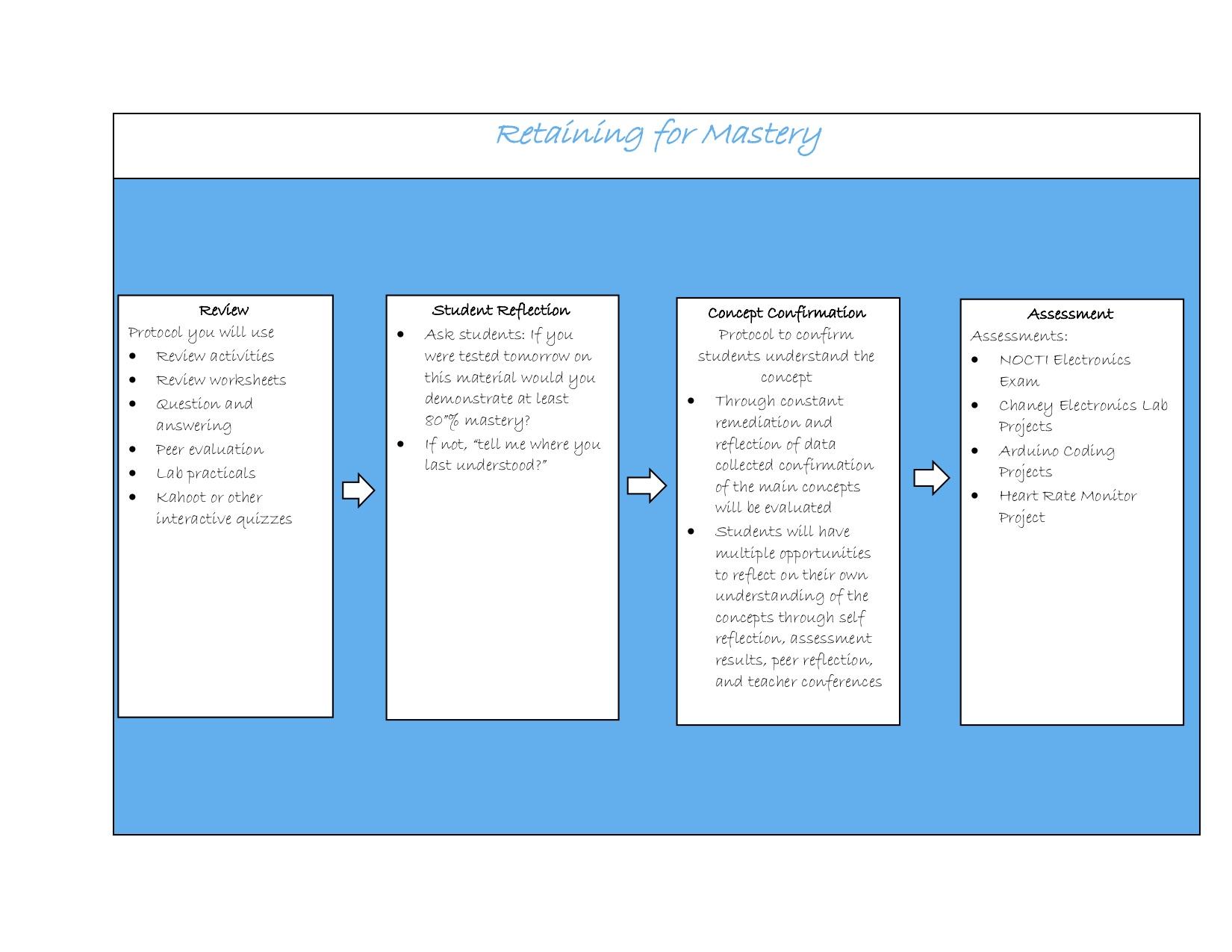
# Piezoelectric sensors have been a hot topic in heart monitoring research, mainly due to its cheap price and accuracy. Our project differs from others by going from the hardware to the software in one project. The hardware portion (i.e. subgroup #1) was able to effectively monitor data. However, the choice of software caused a baseline frequency near the frequency of the R-peaks. For the data analysis portion (i.e. subgroup #2), we were able to calculate R-peaks and test for arrhythmia. Since we used just threshold detection methods, we lack conclusity in differing high P and T peaks within the wave. For the android application process (i.e. subgroup 3), we designed an application that could display graphs and connect the android device to the hardware components, through bluetooth.

Our future works include developing a fully encompassed monitoring system that incorporates a wrist band made with PVDF sensor that holds a display similar to a Fitbit or Apple Watch. The display will sync via Bluetooth to a smartphone application that will collect the data from the sensor, record the data over time, and sync to a primary care doctor for monitoring/health records.

Additionally, future plans would be better testing for the arrhythmia in the future and expanding on the data that is collected by the wearable sensor. Additional sensors or technology would be added to the band to monitor more than just heart rate.

# Deriving Lesson Plans & Activities





# References

All Transistors. Datasheet. Cross Reference Search. (n.d.). Retrieved from <https://alltransistors.com/>

"Heart Disease Facts & Statistics | cdc.gov", *Cdc.gov*, 2019. [Online]. Available: https://www.cdc.gov/heartdisease/facts.htm. [Accessed: 26- Jul- 2019].

How to use a Piezo Vibration Sensor with Arduino. (n.d.). Retrieved from <https://www.ardumotive.com/piezo-vidration-sensor-en.html>

J. Pan and W. J. Tompkins, "A Real-Time QRS Detection Algorithm," in *IEEE Transactions on Biomedical Engineering*, vol. BME-32, no. 3, pp. 230-236, March 1985. doi: 10.1109/TBME.1985.325532

LDT with Crimps - SparkFun Electronics. (n.d.). Retrieved from <http://cdn.sparkfun.com/datasheets/Sensors/ForceFlex/LDT_Series.pdf>

Morisak, J., “Design of a 5th Order Butterworth Low-pass Filter Using Sallen & Key Circuit.” *uic.edu*, 2000. [Online]. Available: <https://www.ece.uic.edu/~jmorisak/blpf.html>. [Accessed: 30- Jul- 2019].

(n.d.). Retrieved from <http://www.learningaboutelectronics.com/Articles/Active-op-amp-low-pass-filter-circuit.php>

(n.d.). Retrieved from <http://www.learningaboutelectronics.com/Articles/Active-op-amp-low-pass-filter-circuit.php>

“norm”, *Mathworks*, 2019. [Online]. Available: <https://www.mathworks.com/help/matlab/ref/norm.html>. [Accessed: 30- Jul- 2019].

Appendix

Ellis, A., Nnabue, A. & Darkwah Akwaboah, A. “Heart Detection and Analysis” 2019. Github Repository, https://github.com/SunWithIssues/Heart-Detection-and-Analysis

Acknowledgement

Akwasi Darkwah Akwaboah (Mentor)

Pascal Yamlome (Mentor)

Douglas Pitts (Engineering Department Staff)

Dr. Deo Makarand (Educational Resource)