Ambulatory Electrocardiographic Monitoring

Genesis Garay and Arath Sanchez

Faculty Advisor: Dr. Carl Greco

Introduction

- Cardiovascular disease (CVD) is the world's leading cause of death, especially prevalent in low- and middle-income countries. Up to 80% of premature heart attacks and strokes are preventable.
- Current ambulatory electrocardiographic (ECG) monitors, while effective in diagnosing heart rhythm issues, are often cost-prohibitive and have limitations like short monitoring periods, user-initiated recording, and offline data analysis.
- This poster summarizes the foundation of the creation of a monitoring system that attempts to address these issues.

Objectives

(1) To create a 3-lead patch ambulatory electrocardiographic device and send data via Bluetooth to a cellular device

(2) Create an algorithm to detect QRS of waveform and detect common arrhythmias

(3) Create an Android application to receive data and display data for the user

Methodology

The system includes (4) main components.

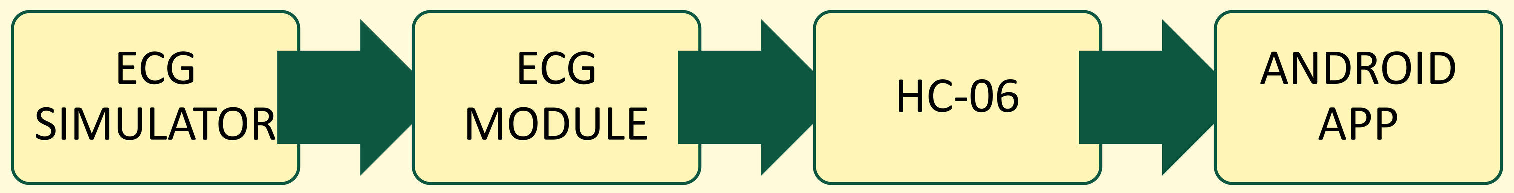

Figure 1. General System Block Diagram

ECG signals were simulated using MATLAB code from PhysioNet to replicate the physiological heart rhythms.

The signal is sent to ECG Module via serial communication to be processed.

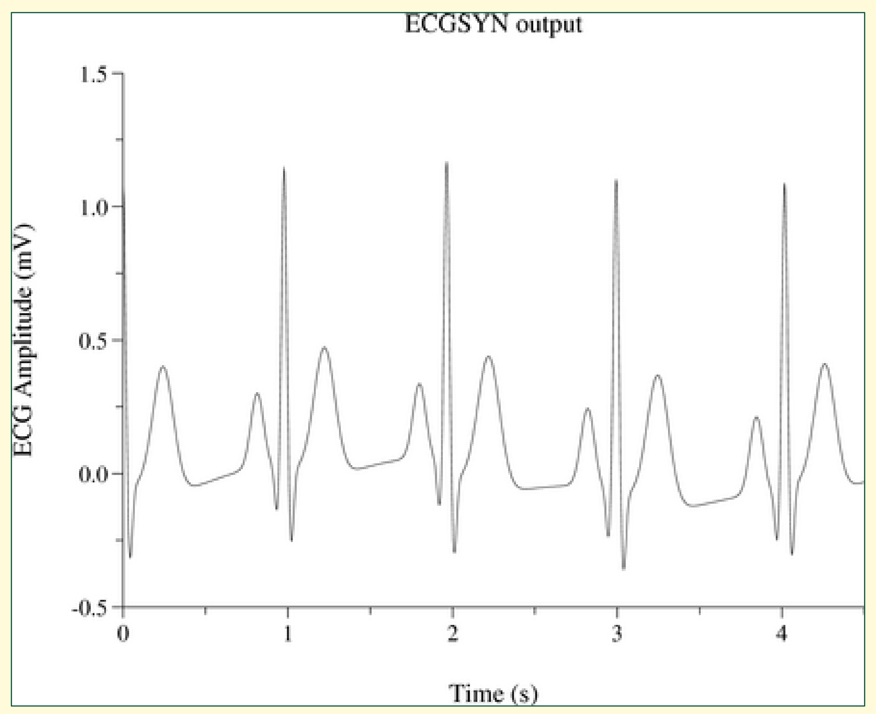

Figure 2. ECG Simulator Waveform

The hardware components of the ECG Module included Arduino Nano, AD8232, and HC-06.

Android Studios IDE was employed to develop the application in Kotlin language. Arduino IDE was used to program a QRS detection algorithm to aid in the detection of common heart rhythm problems.

Results

The results of this project are a functional ECG module that reads signals in real-time and cellular application designed to be worn as a patch and intended for home health care.

We utilized the Pan-Tompkins algorithm for our QRS detection. The overall steps of this technique are band pass filtering, differentiation of the signal, squaring the signal, integrated over a fixed sample size, and applied an adaptive threshold. Our program additionally contains a detection hold-off period to prevent detecting multiple QRS complexes in rapid successions.

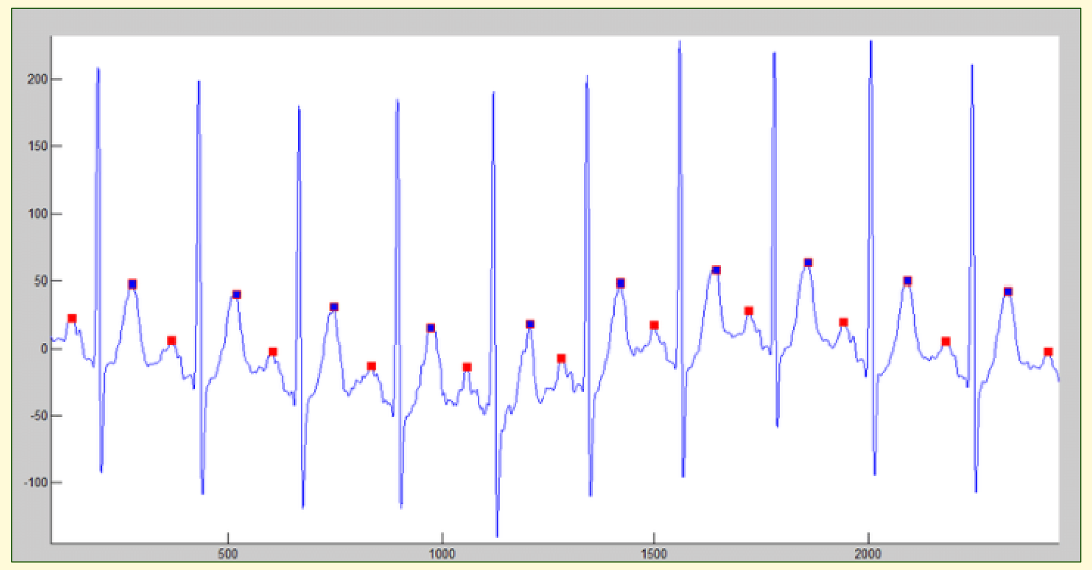

Figure 3. Visual of Program Pin-pointing Intervals

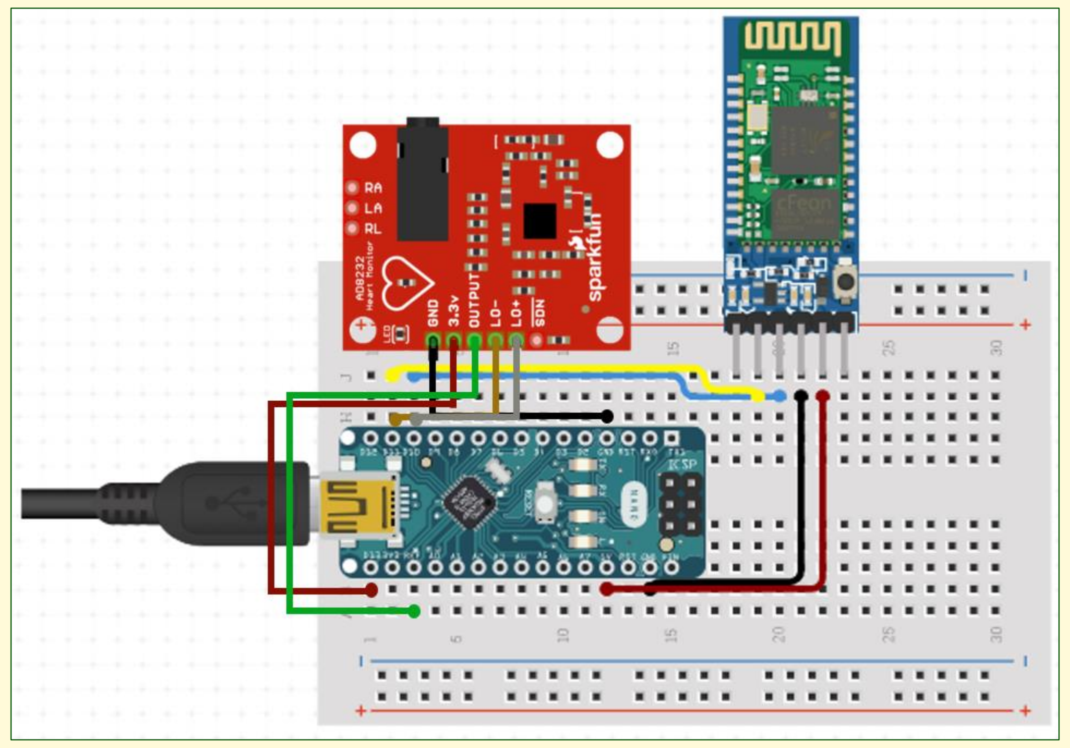

Figure 4. ECG Module Hook-Up

MATLAB feeds the module with the signal and after the detects the QRS it converts data to numeric data points and graphs in real time.

Figure 5. illustrates the Android app that is integrated with the ECG Module. The user can navigate between three pages. The main page displays the user's real time heart rate and ECG waveform. The log page stores string data of patient symptoms during arrhythmia. The history/ setting page stores personal information.

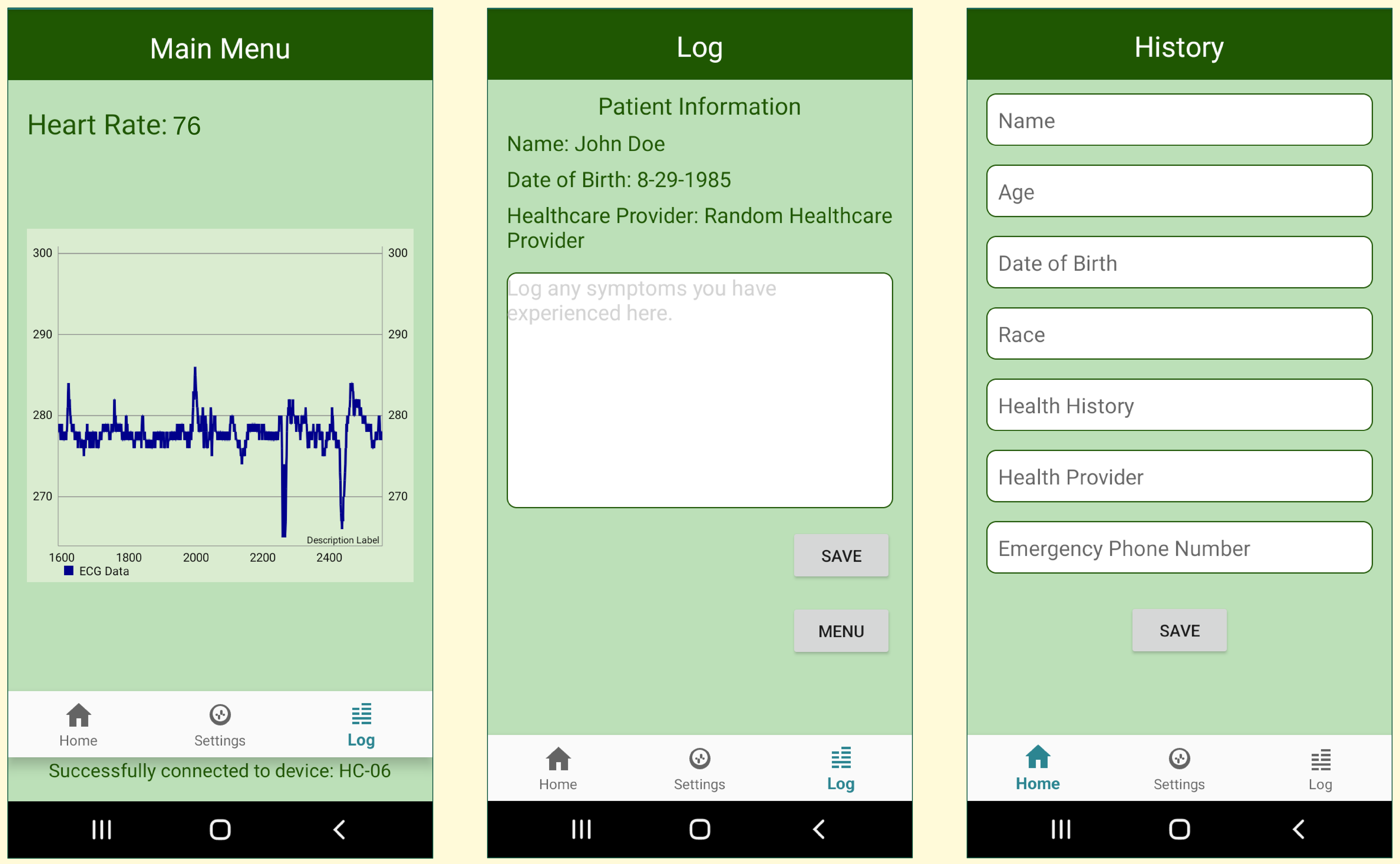

Figure 4. Android ECG Application

Discussion/Conclusion

The ECG module program takes up 25% percent of the microprocessor program storage space which leaves room for common arrhythmias detection. Due to lack of time, we were not able to implement a logic to differentiate between arrhythmia. It consumes minimal power and the Arduino Nano operates at 5V with a current program drawing ~50mA. HC-06 draw ~8mA. Since the device has only been tested with an ECG simulator data on AD8232 was not taken.

Total Power = (Arduino Power)+(HC-06 Power) ≅ 290mW

Total Current = (290mW) / 3.7V* ≅ 78.38mA

Operating Time (hours) = Battery Capacity (mAh) / Total Current (mA)

= 1000 mAh / 78.38mA ≅ 12.76 hr

COST ESTIMATES	
ITEM	COST ($)
Nano	$24.90
HC-06	$9.49
AD8232	$21.50
Other*	$30.00

Total cost estimate of the current prototype ≅ $85.89

In conclusion, this project led to the basic foundation for an ambulatory electrocardiographic monitoring system. It begins to address issues such as utilizing real-time analysis and cost-efficiency while consuming minimal power. However, this system has much room for improvement.

*With 3.7V we assume a Li-ion battery and 'Other' include the cost of items such as miscellaneous electrical components, sensors(wet), battery, chassis, etc.

Future Directions

- Further develop ECG module algorithm to use QRS intervals to detect patterns of common arrhythmias
- Add a way on Android application to store data through cloud-based services rather than locally to avoid overload.
- Redesign application to be visibly more appealing and user-friendly
- Obtain approval to perform clinical testing on humans
- Look into ways to reduce power consumption for longer life longevity.

Acknowledgements

We would like to give a huge thanks to Dr. Carl Greco for his guidance and assistance, AASC by ASGC and The Center for Undergraduate Research at ATU for funding.

Scan QR code for References