ECG
Electrocardiogram Machine

University of Central Florida
College of Engineering and Computer Science

Senior Design 1 EEL 4914
Dr. Samuel Richie and Dr. Lei Wei

Group 12

Mark Nguyen	***Computer Engineering***	*markngn@knights.ucf.edu*
Jorge I. Duarte	***Electrical Engineering***	*jduarte20@knights.ucf.edu*
Zack Briones	***Electrical Engineering***	*zackbriones@knights.ucf.edu*
Krystian Plaskota	***Electrical Engineering***	*kplaskota@knights.ucf.ede*

Table of Contents

1. Executive Summary

1. Summary 1 page

2. Project Overview

2. Intro 1 page

2.1 Background 1 page

2.2 Roles and Responsibilities 1 page

2.3 Specification and requirements 1 page

2.3.1 Hardware 1 page

2.3.2 Software 1 page

2.4 Standards and limitations 2 pages

2.4.1 Managing Constraints 1 page

2.4.2 Managing functionality constraints 1 page

2.4.3 Managing financial constraints 1 page

2.5 Milestones 1 page

3. Research Related to Project

3. Research related to project 2 pages

3.1 ECG background 22 pages

3.2 Similar products 2 pages

3.3 Hardware 3 page

3.3.1 Power Supply 1 pages

3.3.2 Batteries 4 pages

3.3.3 Wall Outlet 3 pages

3.3.4 Voltage Regulator 1 page

3.3.5 Linear Voltage Regulator 1 page

3.3.6 Switching Regulator 2 pages

3.3.7 Power Usage 1 page

3.4 Developing Software Selection 1 page

3.4.1 IDE for Mobile Device Application Development 2 page

3.4.2 IDEs consideration for MCU 6 page

3.4.3 PCB software 1 page

3.4.4 Mobile Device Application Interfaces & Functionalities 3 page

3.4.5 Mobile Device Application Feature 2 page

3.4.6 Challenges of Mobile Device Application 4 page

3.4.7 Testing Procedures for Mobile Application 1 page

4. Initializing Procedure for GUI software

4.1 Software design implementation 7 pages

5. Related IEEE Standards and Design constraints

5.1 Professional Standards 2 pages

5.2 Medical Device Regulations 1 page

5.3 Power Supply Standards 1 page

5.4 IEEE standard for rechargeable batteries 2 pages

5.5 PCB standards 1 page

5.6 Soldering standards 1 page

5.7 Personal Health Device standards 1 page

5.8 Design Constraints 3 page

6. Project Design

6.1 Hardware Design 5 page

6.2 Software Design 5 page

7. System Testing

7.1 Hardware Testing 1 page

7.1.2 Power System Testing 1 page

7.2 Overall Design Testing 1 page

8. Administrative Content

8.1 Project Diagram 2 page

8.2 Budget (BOM) 2 page

8.3 Milestones 2 page

9. Conclusions

Future Improvements 3 page

10. Appendix

Citation 4 pages

1. Executive Summary

An ECG/EKG is the medical instrument that can measure the heart activity and convert it into an analog signal which can be recorded on paper or digitized on an LCD screen. ECG and EKG are synonymous and used interchangeably, it is short for “electrocardiogram” or “electrokardiogram” which was the original term used by its inventor, Willem Einthoven. An ECG machine can be used to detect abnormal rhythms caused by damage to the conductive tissue or electrolyte imbalances or it can identify damaged heart muscle in specific areas. ECGs work by detecting and amplifying the tiny voltage changes on the skin that are caused when the electrical signal in the heart muscle is charged and spread during each heartbeat. The signal recorded as the difference between two potentials on the body surface is called an “ECG lead”. Each lead is said to look at the heart from a different angle. Typically, ECGs have 3 leads but some can have upwards of 12 leads. Once the signal is amplified and filtered, it gets passed into an ADC so it can be sampled and displayed on an LCD screen using a microprocessor.

Making an ECG is an appropriate idea for senior design because it utilizes a lot of the knowledge learned from previous classes. Filtering out noise, amplifying a heart signal in the low millivolt range, and creating a portable power supply that is rechargeable with a battery management system would be a challenging problem that requires our knowledge gained from electronics 2 and analog filter design. Being able to sample an analog signal and make it a digital waveform will require our knowledge from digital signal processing and embedded systems and computer science 2.

Our ECG is going to be small, low cost, and portable. We will construct a single lead ECG system primarily used for basic heart monitoring, various arrhythmia checking, and pulse measuring. Given the tight time constraints, experience of our group, and how mature the market is for ECGs, our device is purely for educational purposes and should not be mistaken for a credible medical device for diagnosing actual heart problems. The reason we are making a single lead ECG and not three lead is because one lead is sufficient to detect basic heart arrhythmias. One-lead ECGs can also be used to accomplish 3-lead recordings if measured sequentially. We plan on implementing a few safety features such as lead-off detection for electrodes and ESD protection.

2. Project Overview

Biomedical Technology

Biomedical technology is a combination of technology and engineering to solve problems that involve biology and or medicine. Its application is used mostly in developing medical equipment for diagnosing and treatment for various diseases. The term biomedical technology can also be broken down to sub-field such as biomedical science and biomedicine. This project is an example of the combination of engineering and technology to develop a medical device that can be used to diagnose some common cardiac conditions. The knowledge requirement for this project is challenging for us due to its nature as a medical device, since we do not have any medical knowledge background. We have to learn at least a new set of knowledge about biomedical devices that are used for the heart.

Generalized medical instrument system

Every instrumentation system that contains at least some of the functional components from the following figure chart is a medical instrument system. The major difference between a medical instrumentation system and a conventional instrumentation system is that the source of the signal is living tissues or energy applied to living tissues.

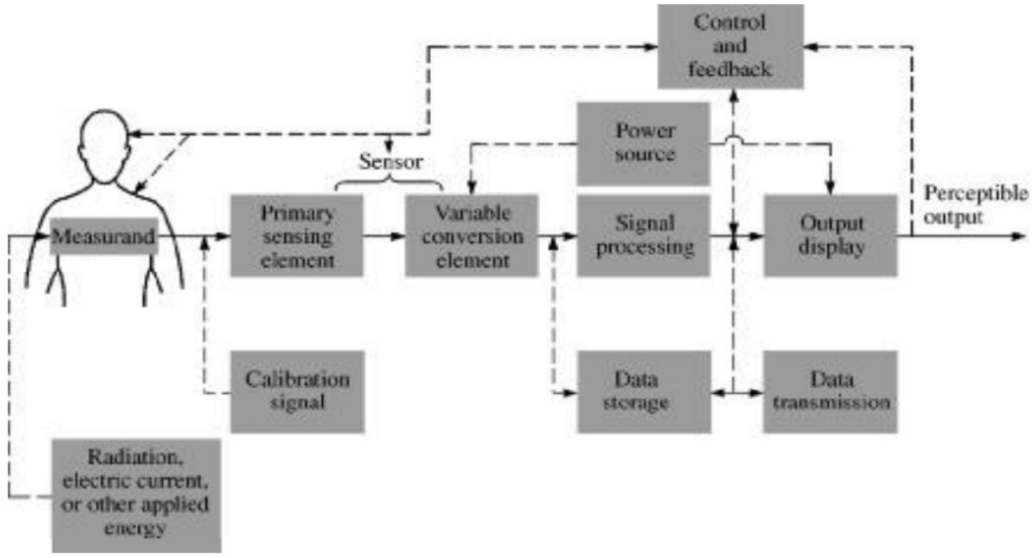


Figure 1: John G. Webster (permission pending)

2.1 Background

This project requires a good background of electrical and software developing knowledge. Our group consists of three electrical engineering majors and one computer engineering major. We would like to have an even number of members with similar knowledge backgrounds, but we cannot find another computer engineering major due to different interests such as time to take senior design 2. However, we are lucky enough to form a group with good members who are willing to learn new skills and work together to achieve our goal.

2.2 Roles and Responsibilities

Nguyen, Mark

Software developing for the GUI portion of the EKG machine. The benefit of using a mobile device as an EKG display is that it provides more expandability for features. Since mobile devices are supporting high level developing languages, it is easier to develop a multifunction application that serves as an EKG display. My responsibility for this part is developing the mobile application and testing the functionalities of the main objective together with the additional features from the design.

Duarte, Jorge

Main Focus is Software programming and Hardware interactions, through the given data acquired by peripherals. Will deploy roles into the system and analyze data as it is being halted through the overall design. Determining compatibility between sensor interactions and various protocols used for communication. Will also support PCB construction, soldering, and component implementation.

Briones, Zack

Primary responsibility is the design of the power supply. This includes being responsible for applying adequate power to the necessary components of the design. The role also includes researching how to properly regulate voltage across the analog front end. Zack's other responsibility is to help Krystian design and test the analog front end of the design. This area is very important because obtaining an accurate heart rhythm is essential to this project. We decided to have a more hands on deck approach when handling the analog front end of our project.

Plaskota, Krystian

Co-designer of the analog front end, analog circuitry of the RLD amplifier, and lead off detection with Zack. This involves researching various chips and amplifiers appropriate

for amplifying ultra low voltages and designing filters that negate noise so the ADC can accurately sample the input signal. The analog frontend is the heart of the ECG and it's vital to accurately tune it such that the critical elements of the final waveform are legible and accurate.

2.3 Specification and requirements

The EKG machine that we are designing consists of 4 groups of components. The first group is the input sensors called electrodes that collect the cardiac signals. The second group is the signal processing unit that will process the input from the electrodes. The third group is the power supply unit that supports both AC and DC sources of powers. The last group is the output group which consists of a mobile application that may support multi interfaces (wire/wireless).

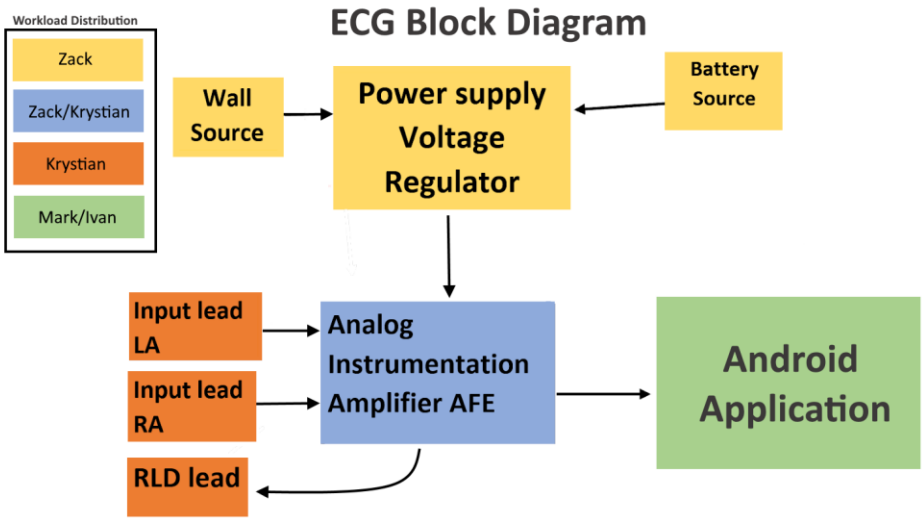


Figure 2: Block diagram

Above is a basic block diagram of our design. It features the main components that we must design to build a functional and portable ECG.

Figure 3 below illustrates a criterion for programming capabilities within the structure of the design.

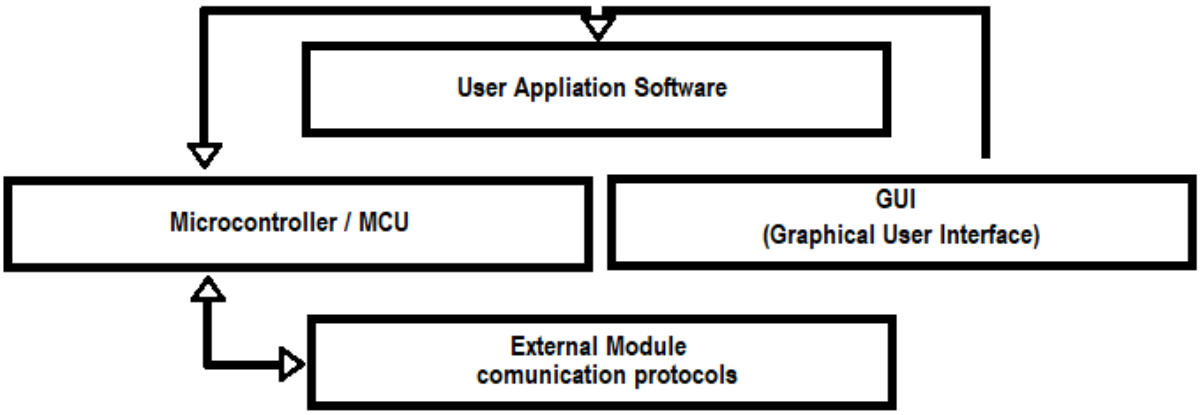


Figure 3: Block Diagram programming criteria

The microcontroller / MCU contains a programmer within the design which self checks abnormalities at the user application level. The MCU and ECG circuit design processes and handles the inputted data from the external Sensors. The censored data is then displayed onto the screen in a real time manner where touch capabilities could then be performed through the Graphical commands from a single user.

House of Quality

Table 1 shows the basic requirements that help our team to determine which trait of the design will benefit the outcome and which trait will hold it back.

	Ease of use	Accuracy	Performance	Goal
Cost	↑↑	↑↑	↑↑	≦ $666
Size	↑↑	N/A	↑	< 12 x12 x 6 (inc)
Quality	↑↑	↑↑	↑↑	≧ 90%
Efficiency	↓	↑↑	↑↑	≧ 80%

Table 1

↑↑ : Strongly positive correlation ↑ : Positive correlation. ↓ : Negative correlation

2.3.1 Hardware

- Metal leads to acquire heartbeat signal from user
- Circuitry to amplify heartbeat signal
- Circuitry to filter heartbeat signal from noise and interference
- Voltage regulator to regulate power
- Protection circuitry for user safety

2.3.2 Software

- MCU should be able to handle robust operation in which outside noise and offset voltages do not trigger analog signals readings into different threshold levels.
- MCU to contain a built-in watchdog timer for system reset on malfunctions.
- MCU to contain I2C and SPI communication software capabilities for external peripherals.
- MCU to possess Software Development tools that can be used in conjunction with C/C++
- Application Software to demonstrate accurate functionality with multiple sensors data reading.
- Application Software to be able to translate the microcontroller data onto an Operating System
- Application Software to display Android charts through JAVA and Kotlin
- Application Software to be compatible with iOS charts (Swift / Objective C)
- Application Software to be compatible through (C# / WPF)
- Application Software to be compatible with Xamarin Charts (C#)
- Application Software to operate RealTime FFT 1024 Pts, Spectrogram and operate 500k points
- GUI to contain Pan in Pan out capabilities
- GUI to provide user select category scrolling functions
- GUI to deliver cursor swifting and hovering roles

2.4 Standards and limitations

Standards in an solid architectural design mostly in the engineering field require the product to be modifiable, bounded complete, normalized, and consistent as well as abstract, verifiable, unambiguous, and practical. This and many other standards. For

engineering projects, the conditions need to be satisfied in order to determine a successful completion of the project.

We follow all the university guidelines and standards from the beginning of every phase of our project. To simplify the process, we divide our project into three major phases: planning, building, and testing. Each major phase has multiple minor phases. Planning includes picking an idea, researching for that idea, and seeking advice from academic advisors. The building phase will take place in the beginning of Senior Design 2. It includes getting parts, assembling prebuilt parts, building our own PCB, developing the software (drivers, mobile application), and combining all components in one container. Testing needs to be conducted after every minor assembling step of the building phase to make sure that every component is working properly. The final testing phase is when the building phase is completed. It will be a tuning period before the project will be shown in our Senior Design showcase.

Planning phase started after we formed our group. According to given guidelines by our advisors, we picked an idea that would fulfill the requirements for a senior design project. The standards that our idea needs to follow consist of requiring enough self-design, challenging enough to show our knowledge, and having a purpose in usability. After we determine the idea, we proceed to do research on this idea based on the standards that are attached. Because the nature of our project requires a human as a testing subject, we were informed to reach out to the office of compliance and risk management for permission. However, since our team members will be the only people to conduct testing on our project, they give us the green light to proceed with this idea. While we are still in the planning phase, we keep scheduling to meet up with our advisors to ensure that our project is following the preset standards. The planning phase which includes actual planning, permission, and research is set to be done while we are in senior design 1. This is the first time constraint that we must follow.

Building phase will start at the beginning of senior design 2. We will have a complete list of components by this time. Each component comes with a set of specifications, and they must meet all compatible standards. Besides that, some components that have specific standards that vary in different regions. For example, a power supply unit (PSU) that takes input from a wall outlet from Asia or Europe will have a different operating voltage range from the PSU that is used in the US. We must review all of these standards when we order parts. Even though the manufacturers apply those standards in their products, there are still chances that we may get a mislabeled or defective product. Before every step of assembly, we will test each component to ensure they would work as their intended design. These testing are also according to safety

practice. This phase must finish within 6 weeks from the beginning of senior design 2. This is the second time constraint.

Testing phase is referred to the tuning period after the ECG machine is built and the software is developed. There are multiple tests run for every assembling process. When we assemble some components together, we will conduct some test runs to ensure that those components work together properly. These testing are not counted in the testing phase. When the building phase completes, we will start the testing phase. The electrical components will be tested with all the safety standards that we obtain from their manufacturers. The software that we develop will be tested with the standards that we will obtain from advisors who have more experience with software development. The tuning period will take a couple of weeks to make sure that every designed function of the ECG machine works properly. We would like to have at least 3 weeks of testing before the showcase in senior design 2. All testing will be conducted under strict safety procedures.

2.4.1 Managing Constraints

When arriving with the idea of an ECG (Electrocardiogram) many thoughts were given in respect to flexibility of the design as well as safety that require focused attention which were the main challenges that could trigger the possibility of design failure. Besides foreseen challenges, we also must prepare for some unexpected situations that may alter our plans. We need to leave room for changes to our design when things do not go as plans, so that we can at least meet some of the main objectives to deliver a working prototype.

There are many constraints that we have to consider before proceeding with the idea that we picked. Based on the guidelines given by our advisors, we first have to consider the purpose of the prototype that we are going to build. Then, we have to consider the financial options. Since this idea is our own, we do not have many options in this constraint. Finally, we have to consider the time constraint which is one very important thing that will dictate many decisions that we will have to make down the road.

2.4.2 Managing functionality constraints

Initially the heart rate monitor practically simulates the electrical signals coming from each heart pulse. In order to collect data into the ECG human testing of analog signals are to be required for diagnostic capabilities. In order to proceed with this idea, the UCF (University of Central Florida) department of health professions and Sciences needed to be consulted in order to obtain an approval for safe testing in humans. Although

the heart rate monitor does not cause any health effects to the human body because the electrical components do not transmit signals but receive instead. It is still safe to remove potential doubts about risk for project disqualification. We were advised to seek approval from the office of compliance and risk management for this project, and we were given a green light to proceed as we explained the procedure which does not pose any risk.

After obtaining the approval from the office of compliance and risk management, we continue our planning phase. The next step is considering financial options. We have our own idea, and it does not really give much incentive in the market for a new product. Finding a sponsor who is willing to pay for a project that may not go to market at all is nearly impossible. The option that we choose is self financing this project and sharing the cost among our team members. This option gives us some financial constraints.

2.4.3 Managing financial constraints

Financial constraints were not an issue for the design decision. The estimated budget was $700, after the parts and rough draft was implemented. As we decided to carve and shave out unnecessary components, we can keep the cost of producing this prototype within the overall budget. In the first design decision of the ECG, we had a vision to place a TFT 7 inch screen module to the design that was about a $200 price if the TFT included a processor. After revisiting the design and noticed that it wouldn't be effective to have to carry a bulky screen around we came out with the idea of a wired or bluetooth interface application that would allow for vision of heart readings through the phone screen. Assuming that every single person with cognitive abilities to use an ECG sensor possesses a phone then that would reduce the cost of the design by $200.

With the new approach, we save a considerable amount from the overall cost of producing this prototype. We decided that our group will share the cost among ourselves. We will start to discuss the decisions on how to manage the fund when we start to have the blueprint inplace. After we have the blueprint, we will start to order parts. There are some parts of this project that are sensitive to current and have potential to fail if we apply too much current during the testing. We will have to consider ordering multiple parts for those components. These insurance parts will save us from delaying the progress of the building phase which is bounded with a time constraint.

2.4.4 Managing time constraints

We picked a relatively hard idea to proceed for this project. This idea requires extensive additional learning for the scope of 2 semesters of senior design 1 and 2. We decided that the first semester of senior design 1 will be the research and prepare period

for this project. We have a weekly meeting schedule for the entire semester, but things have changed during the second half of this semester. Everyone is facing a hard time during the COVID-19 pandemic that forces the strict rule of isolation. Online collaboration is a new way that our group has to adapt to progress with the time remaining for this semester.

A challenge for online collaboration is that sometimes we miscommunicate in one meeting session without realizing it, and that leads to delaying our progress by wasting more time correcting the mistakes through more meeting sessions. Besides this, we have to manage time for other courses that are also changed to online courses, and some of us are having more responsibilities than just school work. However, we are slowly adapting to the new working method and keep working to get as much work done as possible.

As prospecting from the school official announcement, we will have to keep working from home for the next semester in senior design 2. This will be a much harder challenge for us because we need to assemble the parts and conduct testing which require some special equipment that we do not own or too expensive to acquire. Besides the equipment, we also need to work together physically to conduct testing. We are still waiting for more instruction from our advisors to make a plan on how we can proceed to the next phase of our project.

2.5 Milestones

Number	Milestones	Date	Status	Responsible
Senior Design 1				
1	Project selection	1/10/20	Completed	Everyone
2	Divide & Conquer V1	1/31/20	Completed	Everyone

3	Research requirements and update DV document	2/7/20	Completed	Everyone
4	Divide & Conquer V2	2/14/20	Completed	Everyone
5	Begin writing draft paper and researching assigned roles	2/24/20	Completed	Everyone
6	Submit 60 page draft	3/20/20	Completed	Everyone
7	Work on 100 page draft	3/25/20	In progress	Everyone
8	Breadboard testing project components	4/1/20	In progress	Zach - Krystian
9	Begin research app development and screen interface/build	4/1/20	In progress	Mark - Ivan
10	Recording & Data abstract	4/3/20	In Progress	Everyone
11	Submit 100 page paper	4/3/20	In Progress	Everyone
12	Submit Final Senior Design 1 document	4/21/20	In Progress	Everyone

3. Research Related to Project

An understanding of medium

As defined by Signal Theory Textbook version 2012

"Signal: The word signal originates from the Latin word signum, which means an object, a sign, a token, sometimes a gesture. The understanding that is associated with the word signal, therefore, is very old and dates back to the prehistory. Electrical signals, however, have appeared in engineering only since the 19th century . Every mathematical function, of which one of the variables is time, can be considered as a signal. This definition, however, seems to be too wide. The acoustical sound from the dial tone in a telephone system indeed can be regarded as being a signal. The physical description of the sound will require a mathematical function of which the time will be a parameter. Planets moving around the sun can be described also by a mathematical function with time dependency, but are not regarded as being signals. Hence, the first attempt to propose a definition must be refined. A mathematical function depending on the time will be called a process. So, signals are processes, but the reverse is not necessarily true. The second less general attempt to define a signal could be: a signal is a process that can propagate in a certain medium. This definition confirms that a signal is a process, but indicates further that it is to be prepared by a source, that it will propagate in a physical medium further to reach a destination, where it will be received. An alternative definition that takes the previous concerns into consideration but that will introduce the new concept of information is of Frederic de Coulon: A signal is the physical description of the information that it carries from a source towards a receiving destination."

As described in the article above, a signal is an information process that propagates through a certain medium, usually in the form of an electrical wave. In the human body electrical waves travel through the body; However, water is the top constituent of a human, where the male body is composed of by 65 percent of, while the female body is composed of 55 percent of water. All of the body organs and tissues are composed of water and a body requires vast amounts of it to function properly. Water is very important for a strong and healthy heart. The heart being the main organ in the body that it is needed to pump blood throughout the body through a system where the rest of the body can function properly. It is crucial to acquire information on an occasional basis from such an organ in order to assure a healthy state for human life.

Information for the heart travels through a watery medium since is what the body is most abundant with. For this reason when designing an ECG (electrocardiogram), there are plenty of factors that need to be considered. The transmission through the human body is quite impressive and complex since the information of interest is a heart

pulse. Difficulties arrive in which Information is not given in a binary form and it does not give information through an analog sine wave, rather its information is given relatively in the form of an impulse function.

A pulse is observed as an information packet where the information acquired in the impulse can be expanded after compression and analyzed carefully for irregularities in human health. Wavelet transforms are efficient for detecting QRS morphology changes by time and noises in the signal.

3.1 ECG background

ECG is a growing application space. Traditionally, ECG is measured in the hospital environment for patient monitoring. However, we have seen a recent trend in more mobile and wearable ECG systems as well. This section will provide an overview of the inner workings of an ECG, as well as the key functional blocks and specifications needed to measure a human heart.

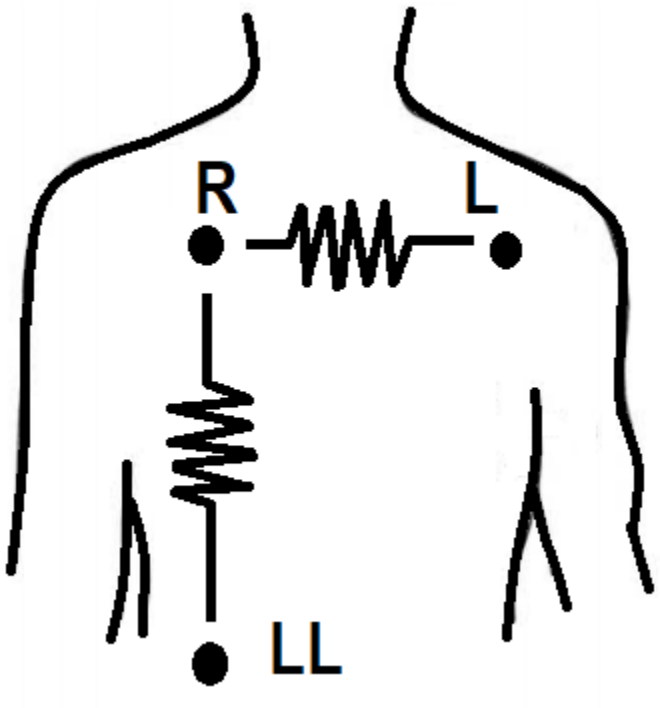


Figure 4: Human body electrodes

The ECG (Electrocardiogram) diagnoses signals and voltage potentials relating to heart conditions. As mentioned previously electrical signals are triggered by pulses generated by the heart muscle.

There are two ways of detecting the heart pulse. One is by implanting a device directly to the heart muscle which is difficult to do due to the cavity placement, and abstracting readings from there which is not recommended because of the heart's delicacy. Also when placing the electrode directly to the heart, the positioning matters The second option is to place electrodes indirectly from the heart to check potential differences, extract the pulses, filter the noise amplify and process the signal for better analysis.

A pulse acquired through the nerve fiber or named central process in the figure above moves at a velocity speed of lambda over tao; v = (lambda/tao), and with a height of about 110milivolts, while the time of the pulse depends on the time constant tau.

For the electrocardiogram position of the electrode placing indirectly for three leads are usually on the left shoulder, the right shoulder and the left leg (negative electrode / indiferent). Three leads are placed in order to obtain the potential difference and DC offset consideration. Resistors are added between the electrode positioning in order to obtain augmented lead configuration

Electrocardiography (ECG) is a graphical representation of electrical activity of the heart over a period of time. One cardiac cycle of ECG signal consists of P wave, QRS complex, along with T wave shown in figure below:

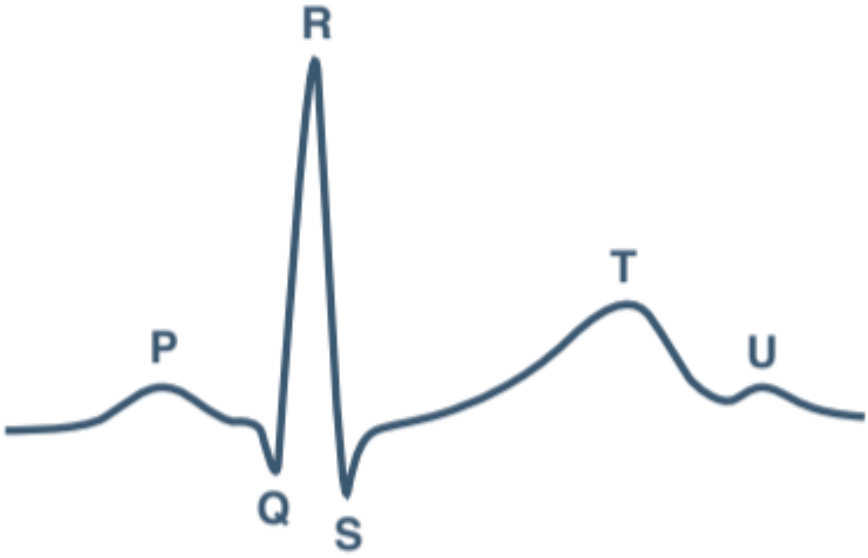
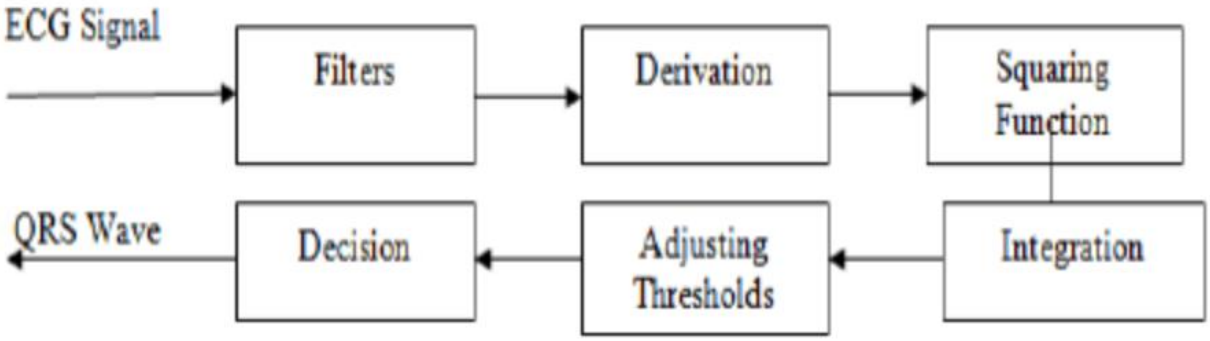


Figure 5: By Enrique Company-Bosch
[enrique.combos@analog.com]
Eckart Hartmann [eckart.hartmann@analog.com]
(Permission Pending)

The P wave represents depolarization, and QRS represents ventricular depolarization. T wave represents rapid repolarization of the ventricles. Out of these three characteristics of the ECG cycle, QRS complex is the most significant part in ECG because of the high amplitude of R peak. Thus, the R peak is used to detect the heart rate of a person.

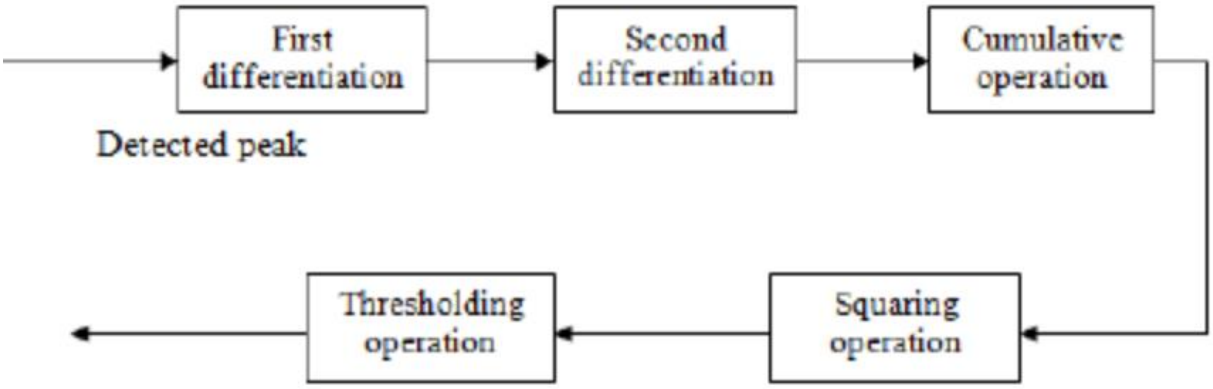
The detection of the QRS complexes in the ECG signal is developed into many algorithms that are helpful to determine a person's heartbeats based on R peaks. Two most popular algorithms are Pan Tompkins and Derivative Based QRS Detector. The first method that was introduced by Pan and Tompkins used a series of low-pass and high-pass circuits to filter out noises in the ECG signal. The filtered signal then goes through a series of derivative, squaring and window integration phases. After that the Q,R,S peaks will be detected by thresholding technique. Following is the block diagram of Pan Tompkins algorithm:



(Balambigai Subramanian
Department of Electronics and Communication Engineering
Kongu Engineering College, India)

Figure 6: (Permission Pending)

The second method uses a band pass filter with cut off frequencies of 5 Hz and 25 Hz. The band pass suppresses the other components of the ECG signal which are P waves, T waves, and noises. The QRS complexes are detected by comparing thresholds with filtered signals. Following is the block diagram of derivative based QRS detector:



(Balambigai Subramanian
Department of Electronics and Communication Engineering
Kongu Engineering College, India)

Figure 7: (Permission Pending

The Heart

The heart is responsible for maintaining adequate blood flow throughout the body to deliver oxygen and nutrients, while removing carbon dioxide and waste. As deoxygenated blood returns from the “systemic circuit”, i.e. the body, it must pass through the heart on its way to the “pulmonary circuit”, the lungs. Oxygenated blood then returns to the heart and is pumped back out to the systemic circuit. During each heartbeat, the cardiac muscle tissue contracts in a specific sequence in order for blood flow in the proper direction; passing from one chamber to the next. The contraction of each segment in the heart produces its own depolarization waveform, like the ones shown below:

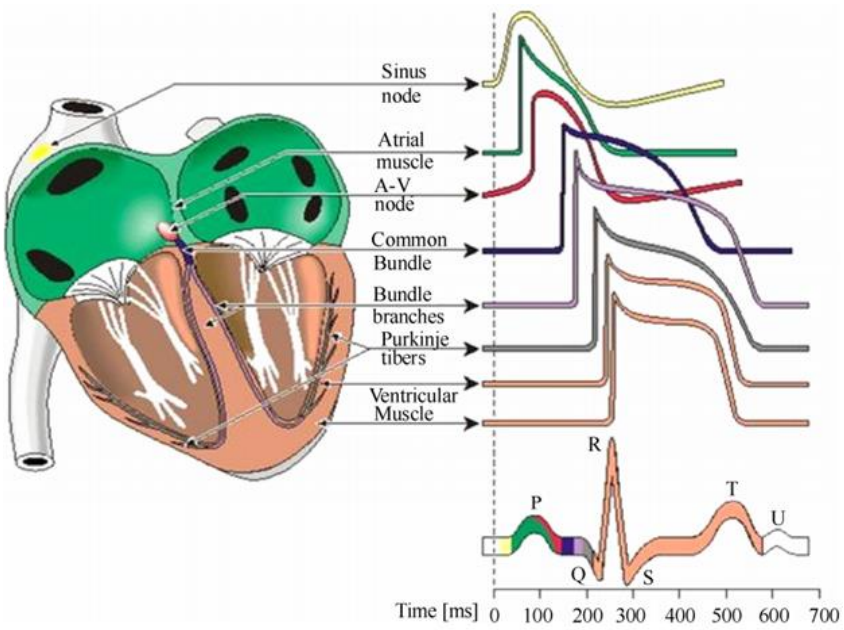


Figure 8: (Permission Pending)

The sum of each of these contractions generates the final resulting wave pattern shown above. This waveform has unique characteristics, which is very recognizable even to non-physicians.

A normal sinus rhythm starts in the sinoatrial(SA) node and spreads down to the atrioventricular(AV) node as the atria contract and force blood into the ventricles. The ventricles then contract and pump blood out of the heart as electrical signals reach ventricular muscle cells. The heart has specialized cells in the sinoatrial node responsible for generating autonomously the electrical impulses used to contract these muscles. Normal rhythm is 60-100 beats per minute.

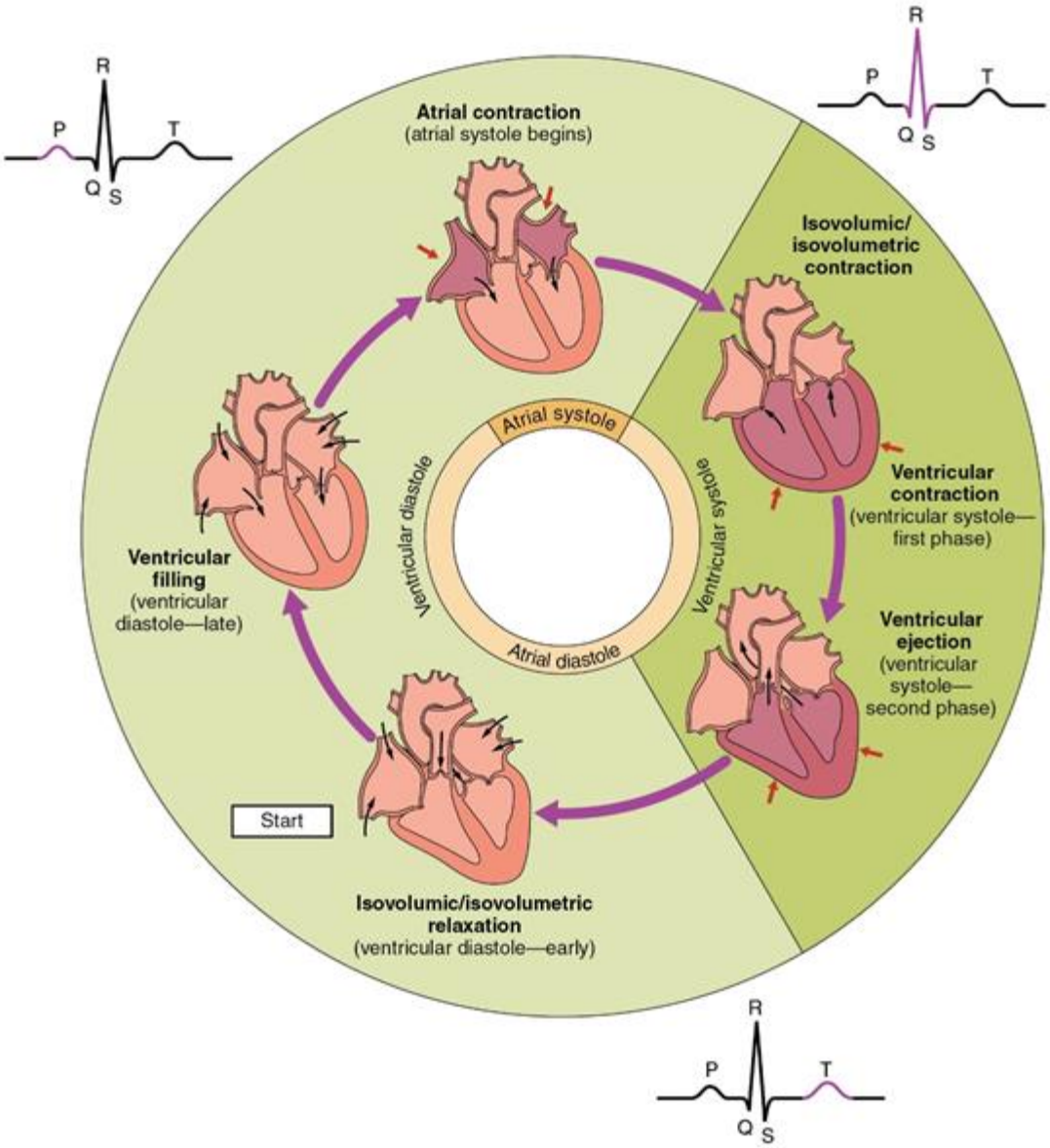


Figure 9: https://creativecommons.org/licenses/by/3.0/ https://upload.wikimedia.org/wikipedia/commons/3/38/2027_Phases_of_the_Cardiac_Cycle.jpg

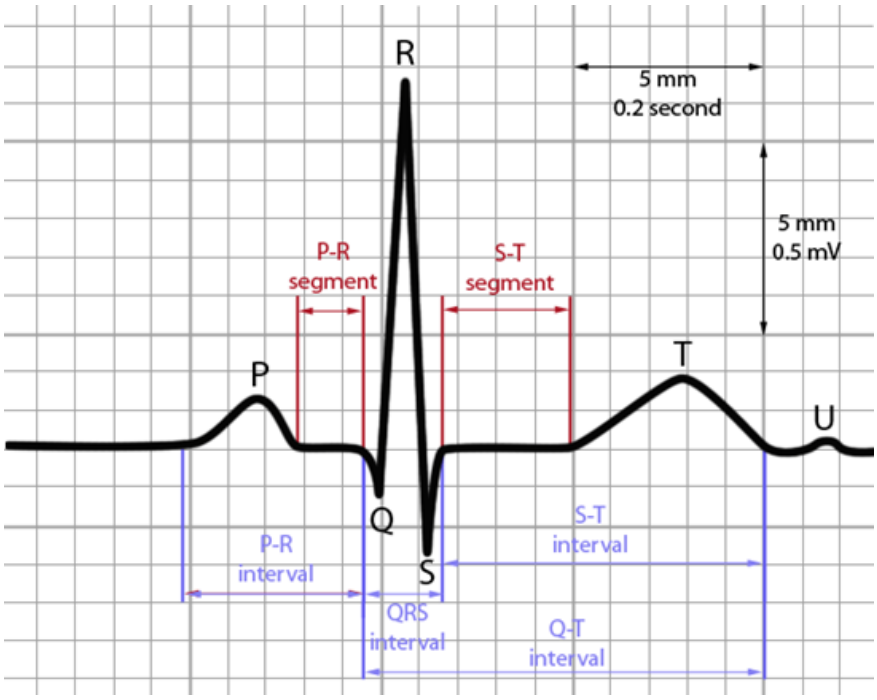


Figure 10: (Permission Pending)

QRS complex

It is essential to understand what the ECG signal characteristics are and where they come from. By looking at the interval between repeated segments of the waveform, you can easily determine an individual's heart rate. To calculate a patient's heart rate, the distance between two R peaks is measured and converted into beats per minute. For example, if the measured distance of one cardiac cycle is 24 millimeters, you would divide 24 millimeters by 25 millimeters per second, then take the reciprocal and multiply by 60 seconds per minute. This results in a heart rate of 62 beats per minute, or BPM. Additionally, as each segment of the ECG waveform corresponds to a specific portion of the cardiac cycle, trained cardiologists can use these wave patterns to diagnose a range of conditions and diseases affecting the heart's health. The range and information from heart rate to diagnostic data is why such a wide variety of health-related applications may require ECG measurements.

Traditionally, the ECG signal is plotted on a special type of graph paper. In Figure 2, 10 millimeters on the x-axis represents 0.4 seconds, which translates into 25 millimeters per second. On the y-axis, 10 millimeters represents 1 millivolt. As Figure 4 shows, a typical ECG has a peak-to-peak amplitude of just a few millivolts, which is measured by an electrode on the skin surface.

The ECG waveform can be subdivided into several parts or stages: P, Q, R, S, and T. The most identifiable part of the signal are the parts Q, R, and S, which is denoted as the **QRS complex**. The QRS complex is heavily useful in determining any abnormalities with the heart. The Q, R, and S waves occur in rapid succession, do not all appear in all leads, and reflect a single event and thus are usually considered together. In adults, the QRS complex normally lasts 0.06–0.10 s. From figure 3h:

- When the initial deflection of the QRS complex is negative (below the baseline), it is called a Q wave.
- The first positive deflection in the QRS complex is called an R wave.
- A negative deflection following the R wave is called an S wave.

With respect to the cardiac cycle the following are the definitions of each segment of the PQRST signal:

- P Wave: Atrial Depolarization
- PR Segment: Delay in the AV node
- PR interval: All electrical activity in the heart before impulse reaches ventricles
- Q Wave: First negative deflection after the P wave but before the R wave
- R Wave: First positive deflection following the P wave
- S Wave: first negative deflection after the R wave
- QRS complex: Signifies ventricular depolarization
- T Wave: Ventricular repolarization

The fundamental frequency for the QRS complex at the body surface is ≈10 Hz, and most of the diagnostic information is contained below 100 Hz in adults, although low-amplitude, high-frequency components as high as 500Hz. Filtering of the ECG signal to within the band between 1 to 30 Hz produces a stable ECG that is generally free of artifacts, but this bandwidth is unacceptable for diagnostic recording because it produces distortions of both high- and low-frequency components of the signal.

ECG artifacts and noise

There are some common challenges to measuring ECG. The ECG signals contain many types of noise artifacts; baseline wander, powerline interference, electromyographic (EMG) noise, electrode motion artifact noise. Baseline wander is a low-frequency noise of around 0.5 to 0.6 Hz. The baseline of an EEG waveform is the DC offset voltage that develops between two measurement electrodes, which can change over time due to things like electrode contact quality, respiration, and patient movement. To remove it, a high-pass filter of cut-off frequency 0.5 to 0.6 Hz can be used. Powerline interference (50 or 60 Hz noise from mains supply) can be removed by using a notch filter of 50 or 60 Hz cut-off frequency. EMG noise is a high frequency noise of above 100 Hz

and hence may be removed by a low-pass filter of an appropriate cut-off frequency. Electrode motion artifacts can be suppressed by minimizing the movements made by the subject. Some of these challenges can be overcome with a good circuit design. But in most cases, digital post-processing is still required to bandpass filter the ECG and remove DC drift or high-frequency interference.

Types of Artifacts:

Wandering Baseline: A slow wander of the baseline.

Cause of artifact:

- Body Movement
- Respiratory swing

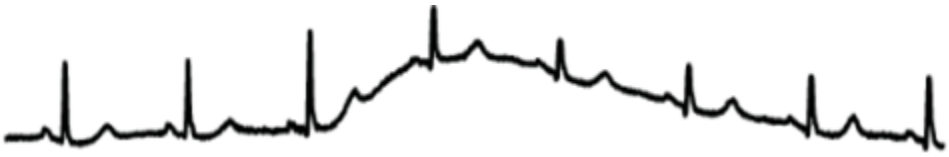


Figure 11: Wandering Baseline (public domain)

AC interference: Varying amplitude of ECG and indistinct isoelectric baseline.

Cause of artifact:

- Electrical power leakage
- Improper equipment grounding
- Close proximity of other electrical equipment

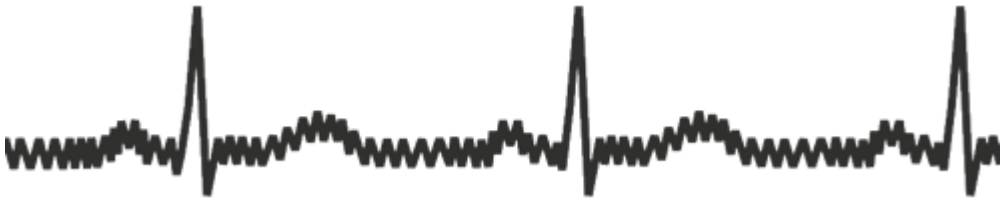


Figure 12: Ac interference (Public Domain)

Muscle tremor: Narrow and rapid spike of ECG

Cause of artifact:

- Effect of EMG signal
- Shivering

- Parkinson’s disease

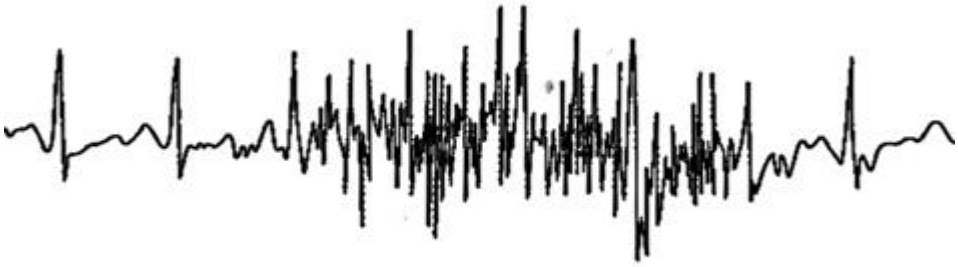


Figure 13: Muscle tremor (Public Domain)

Motion artifact: Large swing in the baseline, uncertainty of large amplitude signals

Cause of artifact:

- Effect of epidermal signal
- Stretching the epidermis
- Coughing
- Ambulation

Figure 14: Motion Artifact (Public domain)

The AC component of the ECG waveform is relatively low in frequency, usually between 0.05 Hertz and 40 Hertz. Diagnostic quality ECG applications may require up to 150 Hertz or more to extract additional information from the waveform. Figure 15 shows where ECG falls in the frequency spectrum, relative to other biopotential measurements. Again, note the magnitude of the ECG is only a couple millivolts.

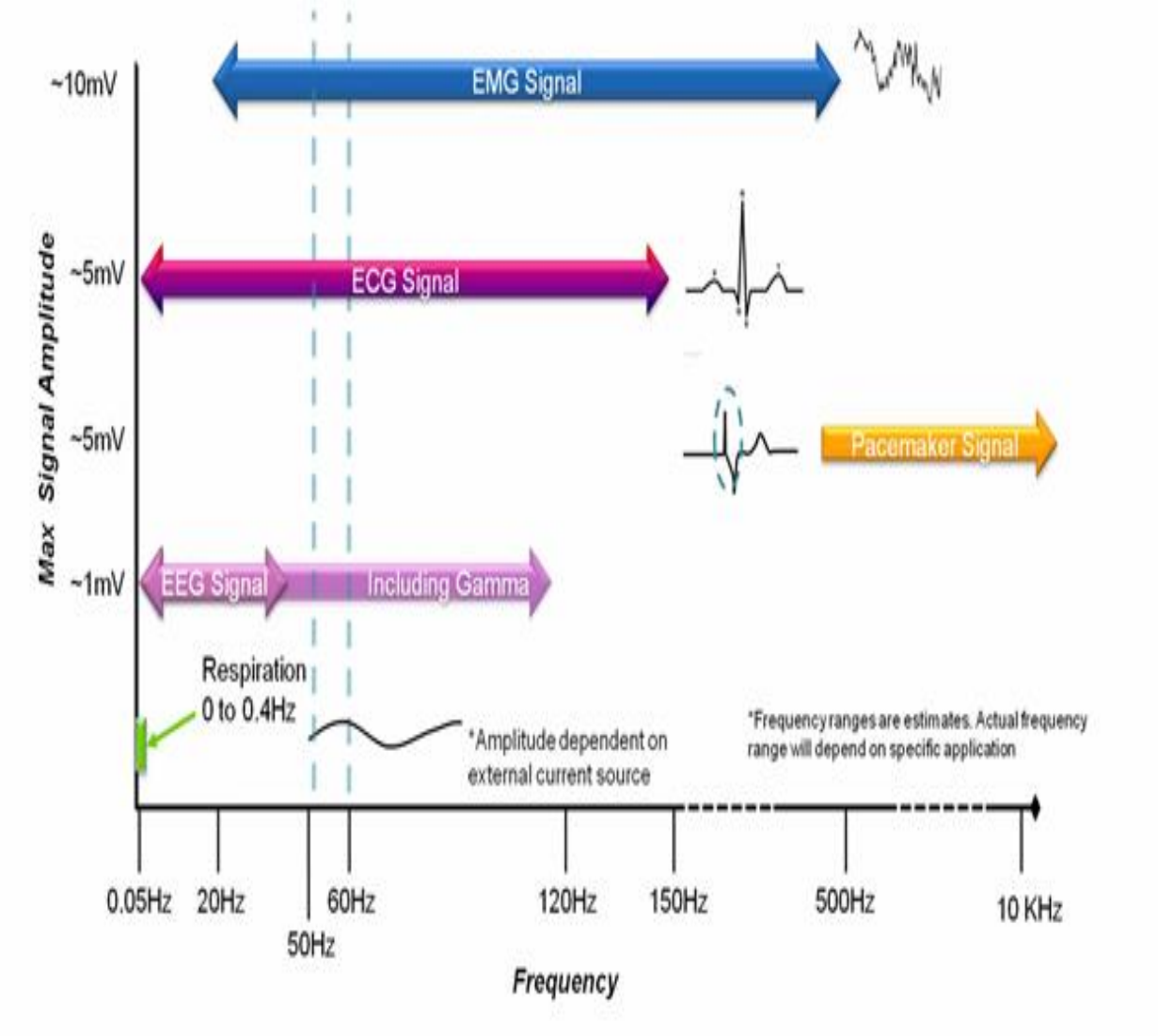


Figure 15: Frequency spectrum of ECG signals versus other biopotential signals (Permission Pending)

ECG leads

The standard electrocardiograph (the instrument that generates an ECG) uses 3, 5, or 12 leads. The greater the number of leads an electrocardiograph uses, the more information the ECG provides. The term “lead” may be used to refer to the cable from the electrode to the electrical recorder, but it typically describes the voltage difference between two of the electrodes. You can think of different leads as different viewing angles that provide unique information about the heart's electrical activity. The number of leads in an ECG system can vary widely. Simpler systems generally use no more than the first two or three primary leads. This is typical in consumer applications and small portable

ECG equipment meant for basic diagnostic purposes. High quality diagnostic systems are more complex and may use up to 12 or more leads.

For a 12 lead ECG system, the leads are organized in the following manner: three primary leads, three augmented leads, and six chest leads. The three primary leads are derived using what is known as Einthoven's triangle, named after the physiologist who invented the first electrocardiogram, Willem Einthoven. The three main measurement electrodes are placed on the **left arm**, **right arm**, and the **left leg**. These electrodes are abbreviated as LA, RA, and LL.

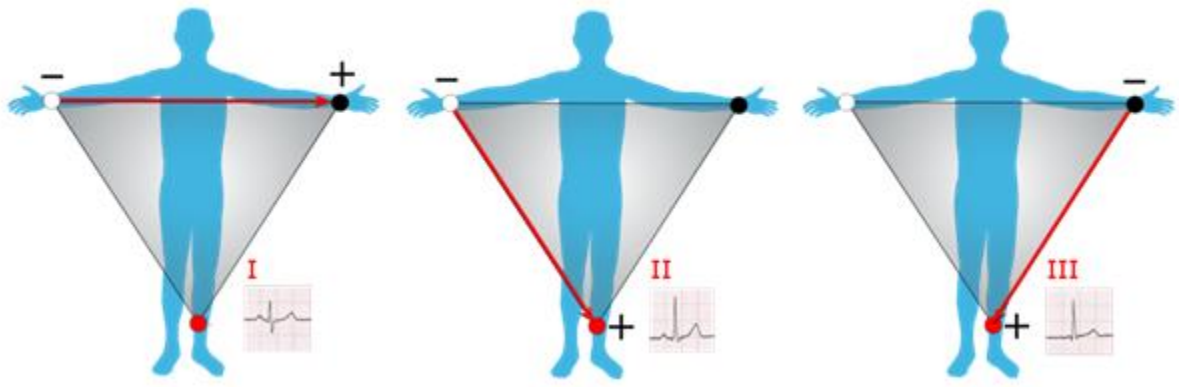


Figure 16: Einthoven's triangle
https://en.wikipedia.org/wiki/Einthoven%27s_triangle#/media/File:Limb_leads_of_EKG.png
https://creativecommons.org/licenses/by-sa/4.0/

From these three primary electrodes, we can derive the three primary limb leads, which are Lead I, Lead II, and Lead III. These three leads are calculated as follows:

- Lead I: V_{LA}-V_{RA}
- Lead II: V_{LL}-V_{RA}
- Lead III: V_{LL}-V_{LA}

Einthoven's Law states "if electrocardiograms are taken simultaneously with the three limb leads, at any given instant the potential in lead II is equal to the sum of the potentials in leads I and III." In other words, you only need to measure two leads, because the third can always be calculated.

Right Leg Drive

In addition to the three primary electrodes, a fourth electrode is commonly used for DC coupled ECG systems. This electrode is known as the **right leg drive**, or RLD. The common mode voltage of the human body is floating with respect to the ECG measurement system. Without a way to bias the body, the input signals from the

electrodes may not be within the acceptable input range of the ECG's analog to digital converter (ADC). Typically, the voltage on each pin, or an ADC, must be within the range of the ADC's power supply, VDD and VSS. The right leg drive electrode drives a voltage to the patient that will cause the DC level of the other electrodes to fall within the supply range of the ECG equipment so they can be measured properly.

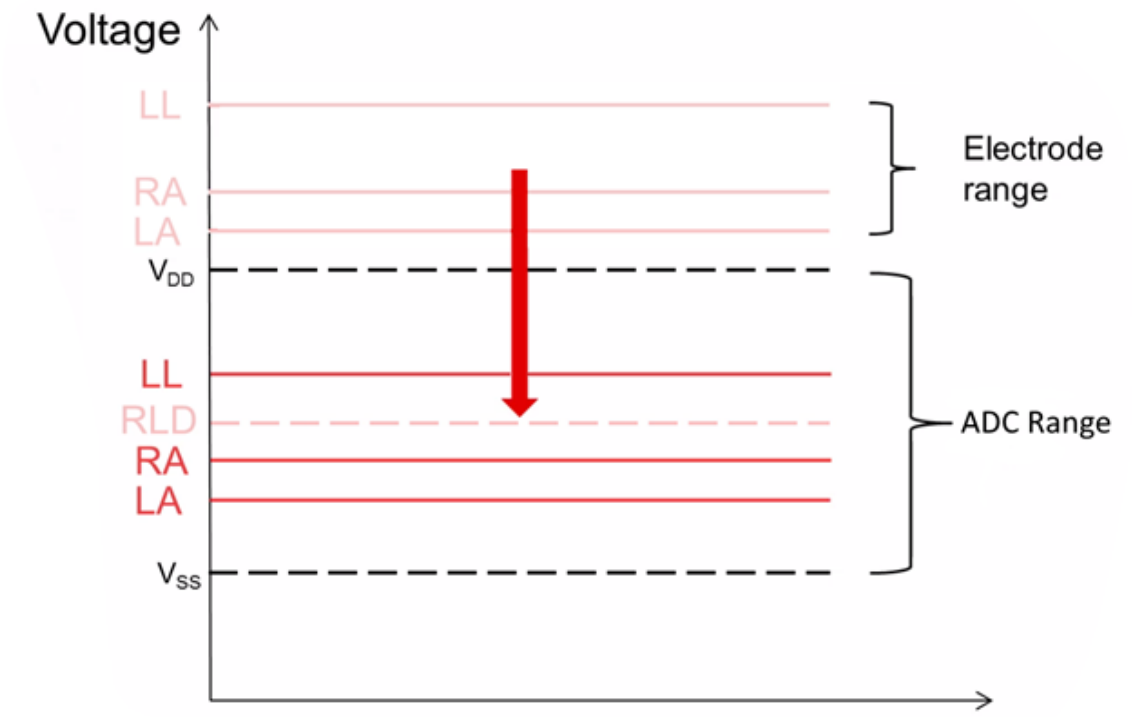


Figure 17: RLD biasing electrodes to be within ADC range (Permission Pending)

Heart Arrhythmias

The following is a description of common types of heart arrhythmias, their causes, and how to recognize them on an ECG. An arrhythmia is defined as any rhythm other than normal sinus rhythm.

Atrial fibrillation: An abnormal heart rhythm (arrhythmia) characterized by rapid and irregular beating of the atrial chambers of the heart. It often begins as short periods of abnormal beating which become longer or continuous over time.

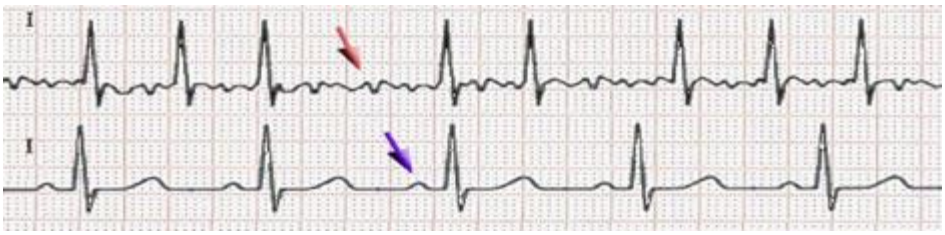


Figure 18: Atrial Fibrillation (top) vs Normal(bottom)
https://en.wikipedia.org/wiki/Atrial_fibrillation#/media/File:Afib_ecg.jpg
https://creativecommons.org/licenses/by-sa/3.0/

Sinus bradycardia: a sinus rhythm with a rate that is lower than normal.

ECG characteristics:

- Rate: Less than 60 beats per minute.
- Rhythm: Regular
- P waves: Upright, consistent, and normal in morphology and duration.
- PR interval: Between 0.12 and 0.20 seconds in duration.
- QRS complex: Less than 0.12 seconds in width, and consistent in morphology

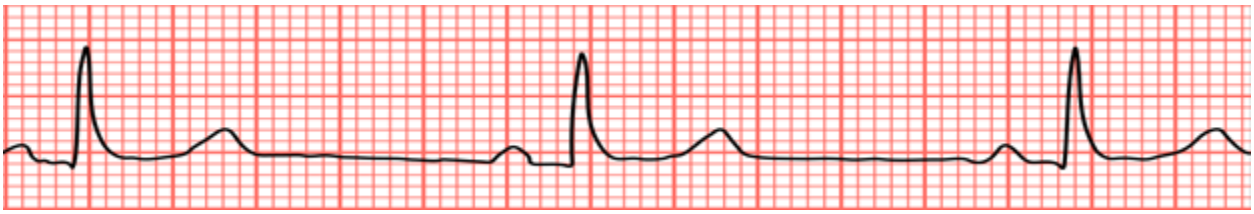


Figure 19: Sinus bradycardia
https://en.wikipedia.org/wiki/Sinus_bradycardia#/media/File:Sinus_bradycardia_lead2.svg
https://creativecommons.org/licenses/by-sa/3.0/

Sinus Tachycardia: defined as a heart rate greater than 100 beats/min (bpm).

ECG characteristics:

- Rate: Greater than or equal to 100.
- Rhythm: Regular.
- P waves: Upright, consistent, and normal in morphology (if no atrial disease)
- P–R interval: Between 0.12–0.20 seconds and shortens with increasing heart rate
- QRS complex: Less than 0.12 seconds, consistent, and normal in morphology.

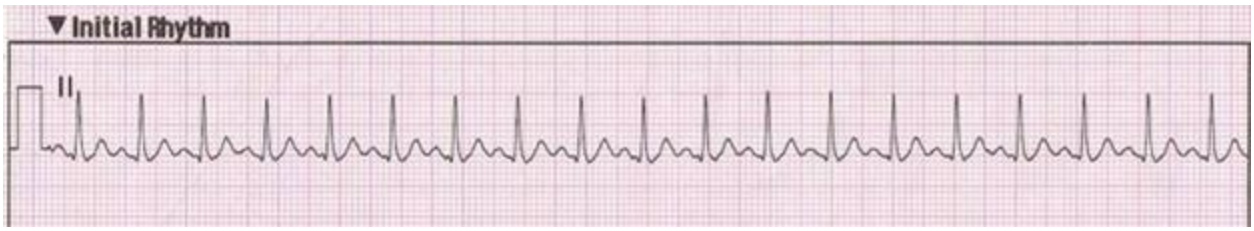


Figure 20: Sinus Tachycardia
https://creativecommons.org/licenses/by-sa/3.0/
https://en.wikipedia.org/wiki/Sinus_tachycardia#/media/File:SinusTach.jpg

Atrial Flutter: Common abnormal heart rhythm that starts in the atrial chambers of the heart. When it first occurs, it is usually associated with a fast heart rate and is classified as a type of supraventricular tachycardia.

ECG characteristics:

- Narrow complex tachycardia
- Atrial rate is between 250-350bpm
- Flutter waves ("saw-tooth" pattern) best seen in leads II, III, aVF
- Flutter waves in V1 may resemble P waves
- Loss of the isoelectric baseline

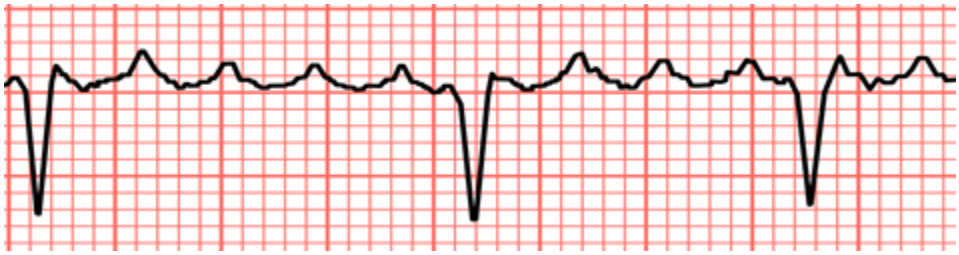


Figure 21: Atrial Flutter
https://en.wikipedia.org/wiki/Atrial_flutter#/media/File:Atrial_flutter34.svg
https://creativecommons.org/licenses/by-sa/3.0/

Ventricular Fibrillation: Multiple foci in the ventricles become irritable and generate uncoordinated, chaotic impulses that cause the heart to fibrillate rather than contract.

ECG characteristics:

- No discernable QRS complexes
- Baseline chaotic
- Heart rate indeterminate

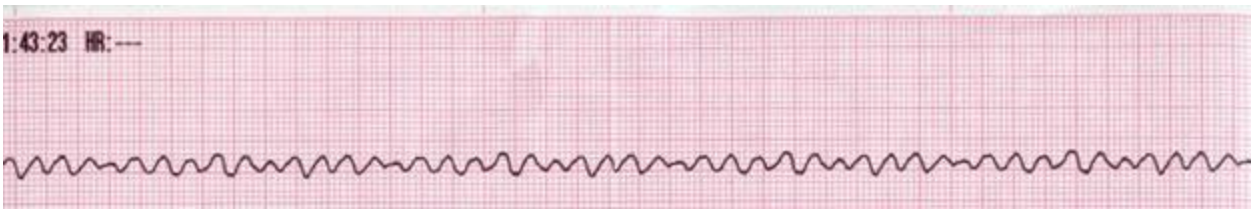


Figure 22: Ventricular fibrillation (Public Domain)
https://en.wikipedia.org/wiki/Ventricular_fibrillation#/media/File:Lead_II_rhythm_generated_ventricular_fibrilation_VF.JPG

Asystole: The heart has lost its electrical activity. There is no electrical pacemaker to initiate electrical flow. The most serious arrhythmia.

ECG characteristics: No electrical activity

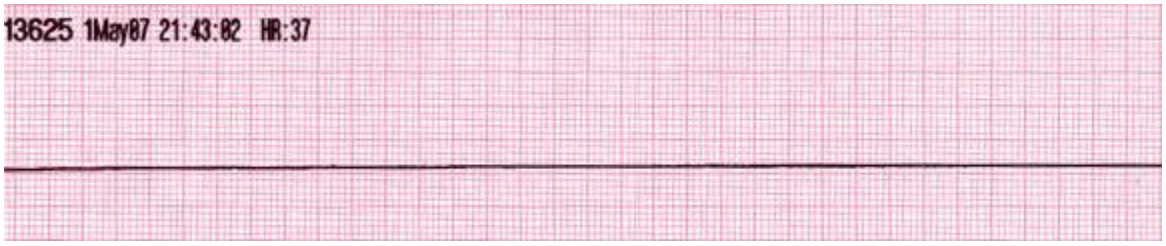


Figure 23: Flat line or Asystole (Copyright Public domain)
https://en.wikipedia.org/wiki/Asystole#/media/File:Lead_II_rhythm_generated_asystole.JPG

Pacemaker Detection

Pacemakers are electronic medical devices used by patients with irregular heart rates, or arrhythmias. These devices start the cardiac cycle in a well-controlled fashion so that the heart can have a more normal degree of function. It is important for many ECG systems to detect if a pace artifact exists in the ECG data. This helps doctors decide how to treat those patients and know whether a defibrillation shock can be administered. Pacemaker signals appear as high frequency pulses right before the ECG waveform. The pacing signals have much higher frequency content than other bio-potential signals due to their narrow pulse width.

The pace pulse is bipolar, with fast rising edge about 10µs; amplitude on the patient skin surface varying between few hundred µV to several hundred mV; and width of pace artifacts between 100µs and 2ms (see figure 6). There are different medical standards with variable requirements regarding the height and width of the pace pulse that must be captured and indicated on the screen of the device. According to ANSI/AAMI

EC11 the features of the pacemaker pulses that should be obligatory detected are as follows:

- duration from 100 µs to 2ms
- amplitude from 2 mV to 250 mV
- frequency up to 100 pulses per minute
- rising edge duration less than 100ms

The IEC60601-2-27 standard states duration from 0.5ms to 2.0ms and amplitude from 2 mV to 700 mV. Modern pacemakers could generate smaller pace pulse amplitudes that could fall below the requirements set in the standards and lead to complications in the algorithms for pace pulses detection.

Texas Instruments' TIPD111 is a reference design that detects pace signals via hardware. The circuit combines three individual circuits: a differentiator circuit used for slope detection, a window comparator to monitor for an event and SR latch to indicate an event has occurred. A pacemaker signal at the input will latch a digital I/O pin high to indicate the signal is present in the waveform. Below is a schematic for the reference design:

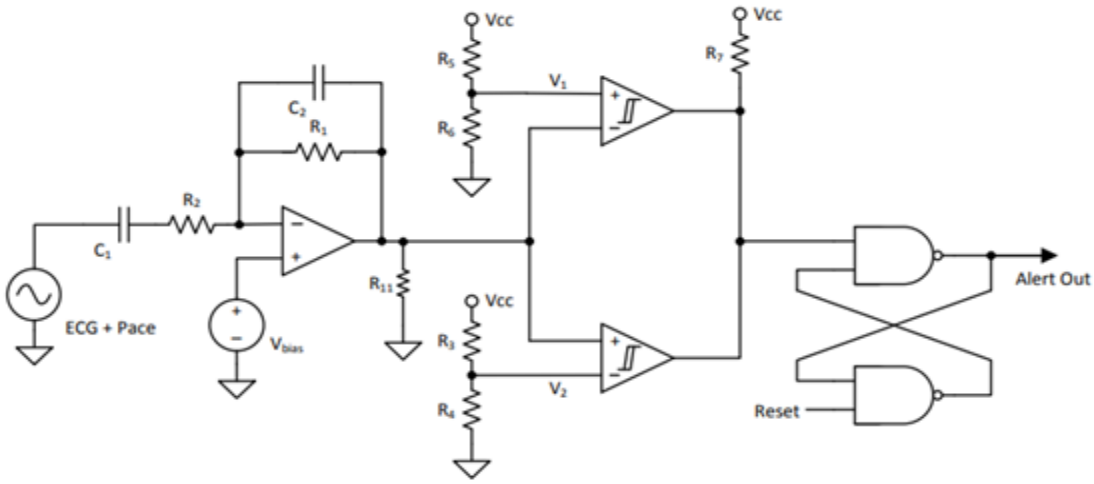


Figure 24: Pace detector

The first stage monitors for the leading edge of the pace signal while attenuating the QRS complex. A differentiator circuit is selected to monitor for the steep leading edge

of the pacemaker signal. This circuit takes advantage of the characteristic of current across a capacitor and uses the op amp feedback resistor to develop a proportional voltage. The output from the circuit will reflect a signal relative to the slope, dV/dt, of the input, shown in Figure 25:

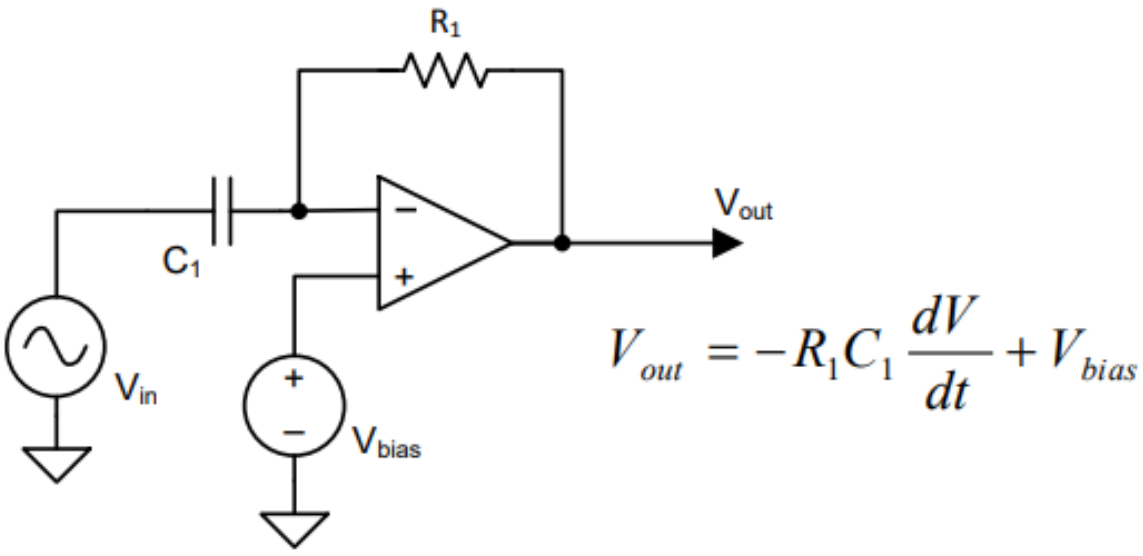


Figure 25: The differentiator

The passive components will set the poles and zeros for the transfer function and must be selected to ensure stability. A second capacitor, C2, may be placed in parallel with the resistor in the feedback loop to help stabilize the circuit, along with a second resistor, R2, in the input path.

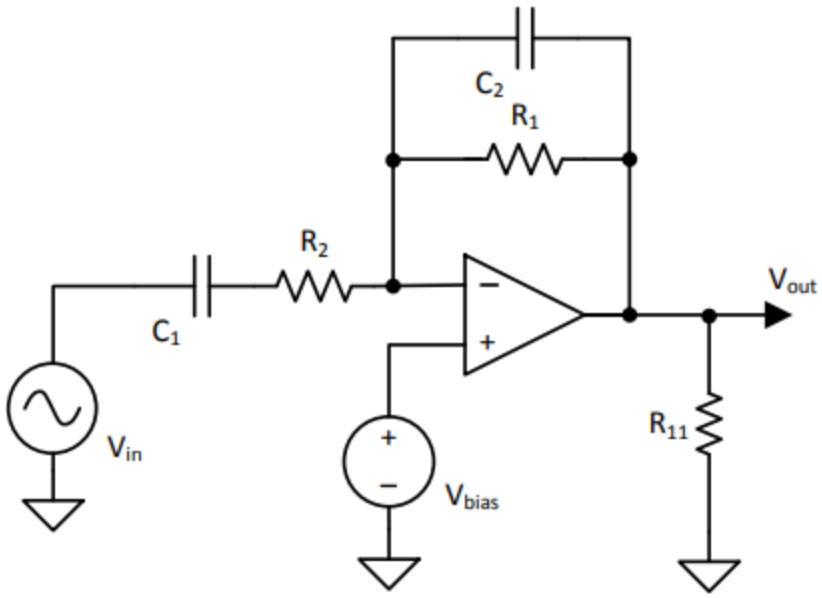


Figure 26: Differentiator with compensation

The output of the differentiator circuit, discussed above, will produce a pulse when a pace event occurs which must then be recognized to register an alert. By design, this pulse from the output of the differentiator can range anywhere from a few hundred millivolts to a couple volts depending on the pacemaker signal characteristics. The OPA348 makes a good choice for the differentiator's amplifier because it has 1MHz bandwidth, can reach within 25mV of the rail and maintains sufficient bandwidth.

Capturing the differentiator pulse is done using a window comparator circuit designed to output a logic low signal when the comparator input pulse exceeds either a high or low threshold voltage. The values chosen are setting V1 to 2.77V and V2 to 2.48V, requiring R3 set to 10.2kΩ, R4 set to 10kΩ, R5 set to 8.06kΩ, and R6 set to 10kΩ. R7 is a weak pull up resistor for the open drain output and will be set to 10kΩ. Vcc is the 5V supply used to power the system. The open drain output window comparator circuit used in the design is shown in Figure 27:

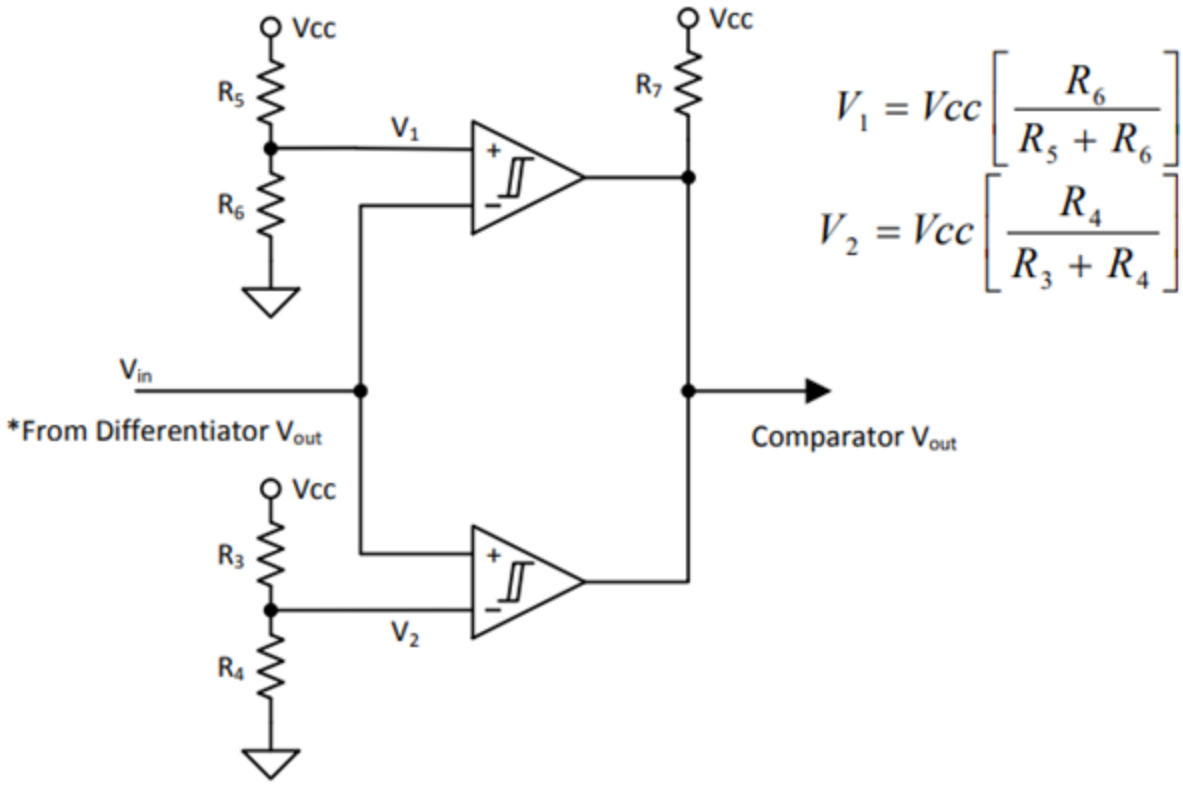


Figure 27: Window comparator circuit

The width of the comparator output pulse is determined by the time that Vin (from Figure 19) exceeds the window comparator boundaries. This duration can be as small as a few milliseconds and may be missed if not latched. The TLV3401 can be used as a window comparator.

Placing a SR-latch circuit after the window comparator will latch the signal in a steady state to be read back by a microcontroller or DSP until a reset is performed. The SR latch will perform two functions: latching the window comparator out signal and serving as an inverter to create an active high alert out signal. The design of the SR-latch uses two NAND gates as shown in Figure 28. The SN74LVC2G00 was chosen for its two devices per package and tpd max of under 5ns.

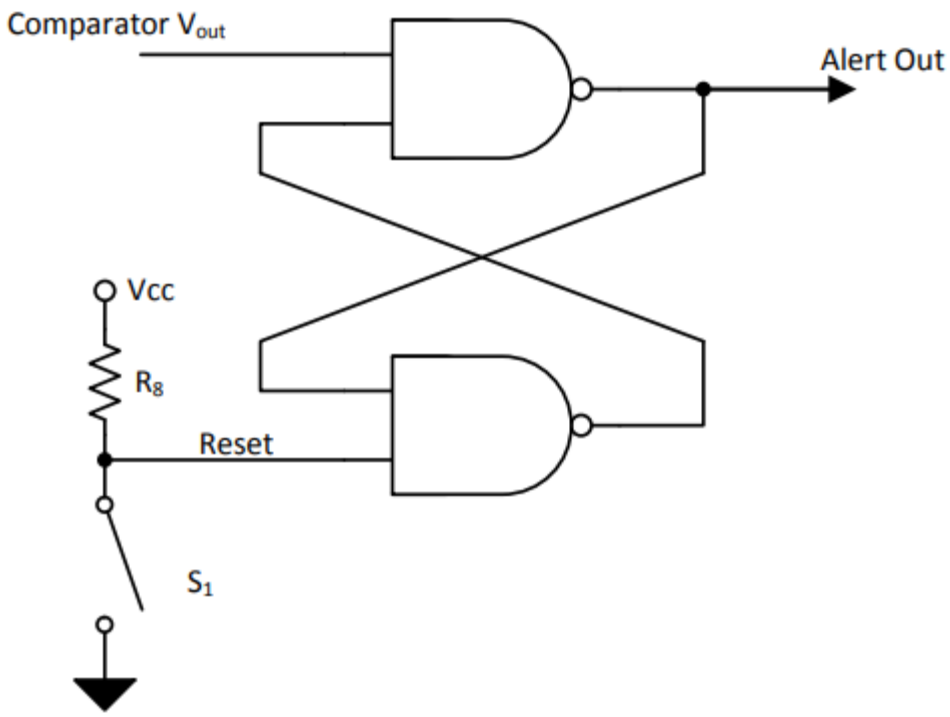


Figure 28: SR Latch

The alert signal can be routed to a GPIO and the state monitored for a pace event. The R8 resistor and S1 switch are used to reset the latch. R8 is a weak pull up resistor and will be set to 10kΩ, and S1 is a single pull, single throw (SPST) push button switch.

Item	Qty	Value	Designator	Description	Manufacturer	Part Number	Supplier Part Number
1	2	1uF	C1, C7	CAP CER 1UF 25V 10% X7R 0603	Taiyo Yuden	TMK107B7105KA-T	587-2984-1-ND
2	1	10000pF	C2	CAP CER 10000PF 50V 5% X7R 0603	AVX	06035C103JAT2A	478-5007-1-ND
3	4	0.1uF	C3, C4, C5, C6	CAP CER 0.1UF 50V 10% X7R 0603	TDK	C1608X7R1H104K	445-1314-1-ND
4	2	Red	J1, J2	TEST POINT PC MULTI PURPOSE RED	Keystone	5010	5010K-ND
5	1		J3	CONN RCPT .100" 10POS DUAL TIN	Samtec	SSW-105-01-T-D	SAM1212-05-ND
6	1		JP1	3 Position Jumper _ .1" spacing	Samtec	TSW-103-07-T-S	SAM1035-03-ND
7	1	392k	R1	RES 392K OHM 1/10W 1% 0603 SMD	Yageo	RC0603FR-07392KL	311-392KHRCT-ND
8	1	0	R2	RES 0.0 OHM 1/10W 0603 SMD	Yageo	RC0603JR-070RL	311-0.0GRCT-ND
9	1	10.2k	R3	RES 10.2K OHM 1/10W 0.1% 0603 SMD	Panasonic	ERA-3AEB1022V	P10.2KDBCT-ND
10	4	10k	R4, R6, R7, R8	RES 10K OHM 1/10W 0.1% 0603 SMD	Panasonic	ERA-3AEB103V	P10KDBCT-ND
11	1	8.06k	R5	RES 8.06K OHM 1/10W 0.1% 0603 SMD	Panasonic	ERA-3AEB8061V	P8.06KDBCT-ND
12	1	1M	R11	RES 1.00M OHM 1/10W 1% 0603 SMD	Yageo	RC0603FR-071ML	311-1.00MHRCT-ND
13	2	2M	R9, R10	RES 2.00M OHM 1/10W 1% 0603 SMD	Yageo	RC0603FR-072ML	311-2.00MHRCT-ND
14	1		S1	SWITCH TACTILE SPST-NO 0.02A 15V	Panasonic	EVQ-P2002M	P12296SCT-ND
15	4	Red	TP1, TP2, TP3, TP4	TEST POINT PC MINI .040"D RED	Keystone	5000	5000K-ND
16	2	Black	TP5, TP6	TEST POINT PC MINI .040"D BLACK	Keystone	5001	5001K-ND
17	1		U1	IC OPAMP GP R-R 1MHZ SGL SC70-5	TI	OPA348AIDCKT	296-27989-1-ND
18	2		U2, U3	IC COMPARATOR SGL OD OUT SOT23-5	TI	TLV3401IDBVR	296-13376-1-ND
19	1		U4	IC DUAL 2-IN POS-NAND GATE SM8	TI	SN74LVC2G00DCTR	296-13257-1-ND

Figure 29: Bill of materials for a pace detector

3.2 Similar Products

There are many similar products that are already in the market. Many of them have successfully proven to be effective for early detection of cardiac related problems. According to tfah.org (Trust for America's Health), 1 out of 3 American has obesity which is one of the most contributing factors to increase the risk of having heart diseases. Heart disease is the leading cause of death in America in recent years. Therefore, early detection of heart related problems is one of the best ways to help better American health. This is the incentive for us to work on an EKG machine project. This project is our challenge to apply the knowledge that we have learned to produce a prototype of an EKG machine. The goal of this prototype is not aiming to compete with any product in the market, but we are trying to demonstrate our knowledge by producing a working prototype with as many features as we can add. Below are some similar products we found during our research.

EMAY Portable ECG Monitor

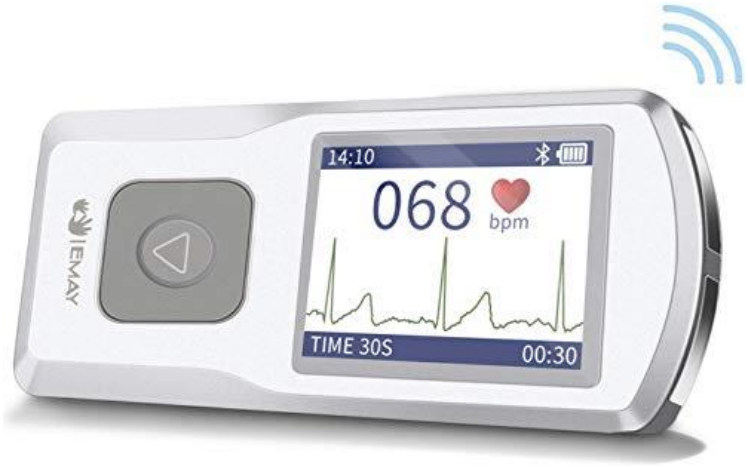


Figure30: (Permission Pending)

EMAY is a portable ECG monitor that costs about $99 on amazon. The monitor has bluetooth capability and can send useful information directly to the user's physician. It allows the user to see their heart rhythm in 30 seconds. EMAY is powered by lithium polymer batteries and is portable. The EMAY app tracks the monitor's information and is compatible with iphone and android users. Seeing other products that can transmit data to a phone application gave us the idea that we could potentially do the same.

HEATHWOOD Portable ECG Monitor

Figure 31: (Permission Pending)

The heathwood portable ECG has almost all the same features as the EMAY but prices in at about $60. This product does not have a phone app but can still record the useful data and upload it to a computer to be shared with the user's physician. Both of

these products seem to be handheld and can fit in the user's pocket. This product, however, runs on lithium ion batteries unlike the lithium polymer batteries that are used for the EMAY device.

Both products gave us some insight on what is possible when building an ECG. We are not trying to compete with either of these products nor trying to sell it on the market. Our team's goal is to show the skills we have learned and apply those skills to building a safe and reliable ECG.

3.3 Hardware Selection

Within this section a detailed description for every potential hardware instrumentation is layed out, focusing on their limitations and capabilities. The strongest contender for hardware devices would then be determined by group vote on engineering requirements and prerequisites that would best bring our design forward. Depending on outcome results adjustments may occur from a hardware standpoint as the design progresses throughout the length of the design.

Hardware specifications

The process for this design began with the entire discussion of how to complete our goal during our weekly meetings. Our goal for this design was to build an Electrocardiograph that could be used portably from the comfort of the user's home. In terms of hardware, our design will follow that of a normal ECG. This includes input leads to acquire the heart signal, the analog front end (AFE) which will amplify and filter the input signal, and a power supply to provide sufficient power to the components of the analog front end. The ECG will be a three-lead input to produce an accurate reading of the user's heart rhythm. If a lead isn't properly connected to the user or a lead is malfunctioning, the ECG will notify the user. This is done by implementing lead off detection at the input leads. The analog front end (AFE) will be soldered onto a printed circuit board which will be housed inside a small case that will be convenient for use.

Once the signal is acquired and passed through the AFE, the waveform will be transmitted to a phone application and processed using computer software to collect information we deem useful for our design. This information will be displayed on the user's device through the application. All wiring and power protection will follow National code and standards to ensure the user is safe from electrocution. Safety is our biggest priority because to properly test our design, we must test on live subjects. Even though we will be the only people testing our design, protecting ourselves is extremely important to us.

The AFE will be the only component that requires power. The AFE includes various operational amplifiers that need regulated voltage to be powered. Our design requires voltage sources from either rechargeable batteries or a wall outlet. Both of these options will be inside the protected casing of the ECG. To protect the user and to follow the standards for medical instruments, we implemented protection schemes into the AFE to be certain that no harmful currents will be sent back into the user's body.

Hardware Issues

Our team knew that if we wanted to go forward with this design, one of our biggest priorities would be building protection within the system to protect the user from currents going back into the user or commonly called fault currents. Although our team will be the only people testing the design, we wanted to be absolutely sure we would be protected. From our experiences in Electronics 2 and from prior research about protection schemes, our design at the input leads will include series resistors to limit current and protection diodes to protect components. Another issue we face is being able to put our printed circuit board safely and securely inside a casing. Since our ECG will be portable, it will be subjected to a lot of movement. We want to make sure that the components won't be damaged or compromised from sudden movement

ECG / ECG Standardizations

In certain modern advances within the medical field, certain requirements and specifications need to be improved and updated in the analysis of an Electrocardiogram. Lot of issues have been given light upon because of this outdated result.

First about 49 million EKGs were managed by a computer in 1986 which is 5 million more than 5 years earlier. Because of the fast-growing population and a high demand in the economy there is a vast number of machines requested daily and computer aided Electrocardiogram machines are more frequent in hospitals and emergency clinics. There were several factors in the use of these popular machines and it mostly has to do with cost, performance and demand. In this new era the semiconductor chips that compose this once expensive machine which is about $5k U.S. dollars helps the user facilitate job accurately to obtain accurate readings.

Another reason is most Electrocardiogram systems which after filtering processing and amplifying signals allow for higher quality resolution then previous systems.

A third issue is that high and low frequencies in the Electrocardiogram are more insightful and difficult to understand and for that reason researchers need a requirement for routine recording inspection for recognition.

ECG SystemStandardization steps

For Routine Visual Reading

1. A representation of the input signal will grant deviation to output recorded,which may not exceed 24 gV or 4.9%, whichever is greater.

For Morphological Diagnosis by Digital Program

2. The root means Square error over the PQRST complex should not exceed 10uV
3. The error in Vpp ions may not exceed 10 gV or 2.1 percent or larger value.
4. Root mean squared error divided by mean squared amplitude of a deflection should not exceed 2%
5. QRS deflections with .19 uV amplitude and greater than 6 ms duration should be detected
6.The MPE shouldnt surpass 10% during a number of any QRS deflection . 19 uV and . 11 ms.

For Digital Transmission or Storage/Retrieval of Data

7. The samples of data being reconstructed should differ by 10 uV and half QRS detections

3.3.1 Power Supply

Our goal is to design a portable and reliable ECG to be used in the comfort of the user's home. In today's world, technology needs to work efficiently and consistently especially when a product is involved in a person's wellbeing. This is where power supply plays a huge role in modern technology. Not only is adequate power supply necessary for our design to function properly, but also the components that make up our design could be compromised leading to delays and more money being spent. Safety is also a big concern of ours so avoiding severe damages to ourselves and our design is one of our top priorities. To make sure we are safe, we are following all the necessary standards in regard to power distribution.

Our design will have two options for power supply. The user will either plug the ECG into a wall outlet or be able to use the device on rechargeable batteries depending on the user's needs. We made this decision because we wanted to give the user the flexibility to use the device in a variety of scenarios. If the user is traveling and is not able to use a wall outlet or perhaps the user forgot to recharge the batteries thus forcing them to use a wall outlet. This in our eyes makes the ECG portable and gives flexibility to the user in whichever way they prefer.

3.3.2 Batteries

For our design, rechargeable batteries will act as a backup power supply to our ECG in the event that the user is not able to connect to a wall outlet. Rechargeable batteries give our design the portable feature we are looking for. Our research for this aspect of the design centers around finding the correct battery type that will not only provide sufficient power to our system but also be durable and reliable for the user. There are four types of rechargeable batteries that we researched: Nickel-cadmium (Ni-Cd), Nickel metal hydride (Ni-MH), Lithium-ion (Li-Ion), and Lithium polymer (Li-Po).

Nickel-cadmium (Ni-Cd)

Of the four types of batteries we researched, Ni-Cd batteries have been around the longest. Ni-Cd batteries were for the longest time used in power tools and portable electronics devices because of its wide range of sizes and good cycle life. Being the oldest of the battery types comes with some disadvantages. First off is its memory effect. Memory effect is when the battery is charged numerous times back to the same state of charge and begins to suffer from sudden drops in voltage over time. Some devices may be able to withstand the drop while other devices suffer from the lack of energy. Another problem is that cadmium is a very toxic metal and harmful to the environment. In the last 30 years, great strides have been made to start protecting our planet and thinking long term to protect the environment so it's obvious that these kinds of batteries are being replaced by better options.

Nickel metal hydride (Ni-MH)

Ni-MH batteries are the more updated version of the Ni-Cd batteries. These batteries hit the market in the 70's and were used in satellite applications. Ni-MH batteries became available for commercial use in 1989. Compared to its predecessor, Ni-MH batteries offer more recharge cycles with less memory effect. They also are better used in high current drawn applications due to low internal resistance. Cameras with flashlights that require higher current levels typically use these batteries because they can produce these high levels of current without loss of capacity. Ni-MH batteries last longer than Ni-

Cd batteries however Ni-MH batteries discharge at a higher rate when not in use. These types of batteries can lose up to 20% of their battery life after a week of not being used.

Lithium-ion (Li-Ion)

Lithium-ion batteries, or Li-Ion for short, are the newest type of battery that is being used today. These batteries are widely used in electronics and electric vehicles. Li-Ion batteries have a high energy density making them very useful in electric vehicle applications. The major improvement of these batteries is that they have no memory effect with low self-discharge rates. This is a huge improvement from its predecessors since these were the factors holding those battery types back. However, Li-Ion batteries lose roughly 10% of its capacity every year of its use from chemical breakdown. On top of this, this battery type can be a safety hazard if damaged or improperly charged. Li-Ion batteries have a small history of explosions and fires due to damaged batteries.

Lithium polymer (Li-Po)

Li-Po batteries are of the same family as Li-Ion batteries because they use the same technology. The only difference is that Li-Po batteries use a polymer electrolyte while Li-Ion batteries use a liquid electrolyte. These batteries can store about 350% more energy than the nickel based batteries mentioned earlier with all the advantages mentioned with the Li-Ion batteries. Its main advantage is that Li-Po batteries are lightweight which can be advantageous to certain applications that require minimum weight. Some of these applications include radio-controlled aircrafts and portable electronics. Since this battery type is in the same family as the Li-Ion battery, Li-Po batteries also can be a safety hazard if improperly handled or damaged.

Figure 32 shows a comparison of the different types of rechargeable batteries with some useful information that will help us in deciding which battery type will be best for our design of a portable ECG. Our decision will primarily be focused around choosing some type of lithium based rechargeable battery because of their high capacity and performance.

Comparison of Rechargeable Batteries

Battery	Memory	Self-Discharge Rate	Capacity (mA-hrs)	Toxic	High-Drain Device Performance	Available Recharge Cycles	Cost	Notes
Nickel Metal Hydride (NiMH)	No	Moderate	Moderate to High	No	Good	Moderate	Mod	**Best for:** All high-current draw devices. Use only LSD (low self discharge) variety for low-use devices like emergency flashlights and smoke detectors
Nickel Cadmium (NiCad)	Maybe	High to Moderate	Moderate	Yes	Moderate	Moderate to High	Mod	**Best for:** All high-current draw devices. **Not for use** in low-drain, low-use devices like emergency flashlights and smoke detectors Being replaced by NiMH
Alkaline	Yes	Moderate to Low	Low	No	Poor	Low	Low	**Best for:** LED flashlights (provide brighter light), other infrequently-used devices, items that require extra voltage (i.e., more than 2 batteries)
Lithium Ion	No	Low	High	Yes	Good	Moderate	High	**Best for:** Digital cameras, laptops, cell phones

Memory: Batteries that are susceptible to developing a memory suffer from degraded battery performance over time and multiple charges
Self-Discharge Rate: How quickly the batteries lose their charge sitting in a drawer
Available Recharge Cycles: The number of times a battery can be recharged
Toxic: Toxic batteries have special disposal requirements and must be turned in to special recycling centers

Len Penzo dot Com (www.LenPenzo.com)

Figure 32: (Permission Pending)

Our research then led to comparing the different types of Lithium Ion batteries that we could purchase. The cost of these batteries typically depended on the overall battery capacity and power output. Table below shows some of the characteristics of different battery capacities and other logistic comparisons.

Characteristics	Li-Ion (1)	Li-Ion (2)	Li-Ion (3)
Capacity	500mAh	1200 mAh	2500 mAh
Cost	~$7.95	~$9.95	~14.95
size	1.15” x 1.4” x .19”	1.3” x 2.4” x .2”	1.15” x 1.4” x .2”
weight	10 g	23 g	10.5 g
Output	4.2 V - 3.7V	4.2 V - 3.7V	4.2 V - 3.7V

	(1.9Wh)	(4.5Wh)	(10 Wh)

Table 2

Although batteries are easy to use with low cost, the more the battery is drained and recharged, its battery capacity decays over time. Figure # shows battery capacity decay over a different number of cycles. We used this figure to help forward our research and help us make a decision on what type of battery to use. This is largely one of the reasons we decided that our design will incorporate two separate power supply options. We wanted to give the user the ability to use ECG in any situation. Batteries are not always reliable when they've been used quite frequently. Giving our design this flexibility allows for either power supply source to be the backup for the other.

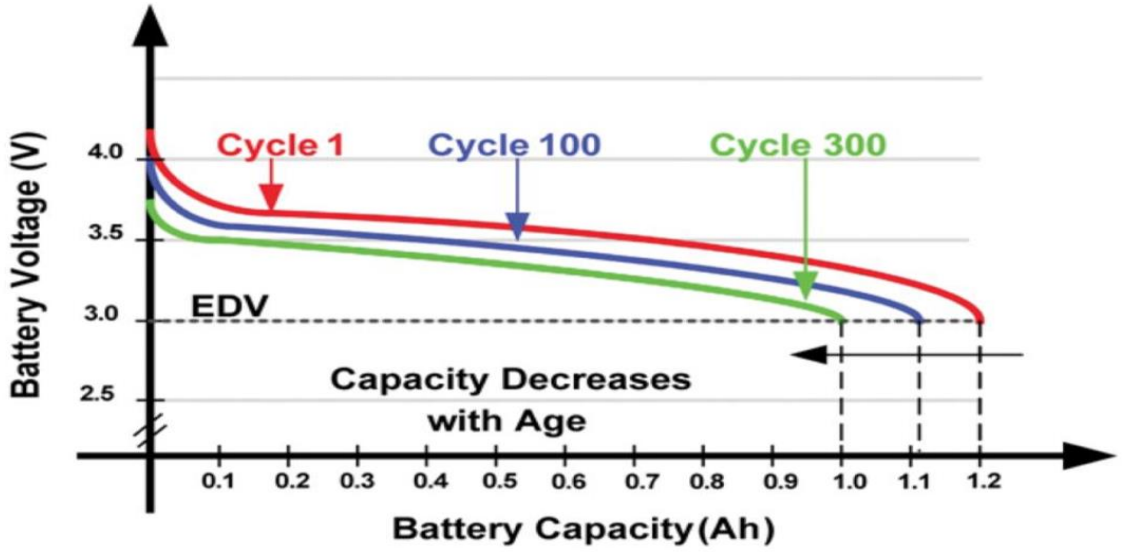


Figure 33: Battery voltage vs capacity (permission pending)

3.3.3 Wall Outlet

The primary power supply will be from a wall outlet. Even though we mentioned earlier that rechargeable batteries will act as the backup power supply, the wall outlet could also be the backup plan for the user in the case that he or she forgets to charge the batteries, or the batteries die. Either power supply option will be viable depending on the

user's situation. Safety is one of our top priorities so anytime we have to deal with high voltages, we need to be certain that our design and ourselves will be protected.

AC/DC Adaptor

Most devices you find in your home will have some sort of AC adaptor implemented to appropriately power these devices. An AC to DC adaptor does exactly what it says it does. The voltage coming from the wall outlet is AC and way too high for certain devices that may only need 5V DC to operate. Our design will only need about 3.3V to 5V to power the components in our design so using the 120V straight from the wall outlet will destroy our design. An adaptor is needed to convert and step down the AC voltage to DC voltage. The figure below shows a simple AC to DC converter to enhance the basics behind this concept.

The AC input coming from the wall outlet at 120V 60Hz will be sent through a step-down transformer and a full wave rectifier. A full wave rectifier is required to convert the AC voltage to a pulsating voltage waveform seen above in the diode rectifier output waveform. To improve the DC output, a capacitor is connected in parallel to the load. The capacitor will not only increase the average output, but also will smooth or filter the output from distortions. This can be seen in the bottom right waveform of the figure above. The larger the capacitor, the smoother the DC output will be. For safety reasons and to ensure we power our design correctly, we will be buying an AC/DC adaptor instead of building our own. As we mentioned earlier, safety is at the top of our priorities so buying an adaptor that already meets the international standards will be beneficial for our design moving forward. The image below is an example of an AC/DC adaptor that is plugged into a wall outlet.

There are many different AC adaptors on the market with different voltage output ratings. These ratings can range from 3.3V all the way to 12V. Some products on the market have adjustable output ratings but at higher cost. Since we will know exactly what voltage level our design will need to operate, we can save on money and purchase an adaptor with one constant output rating. So when choosing an adaptor to purchase, we will make sure the output matches our desired voltage rating or is at least greater so that we can regulate it to the appropriate rating. This thought process also applies to the current rating of the adaptor. Since we are still in the design phase of our project, we don't know exactly how much current our ECG will need to operate. As long as the current rating of the adapter is above the amount of total current our design needs, then our ECG will have enough power to operate. Typically, a higher power rating is more expensive so this constraint will be discussed amongst our group. Adaptors also have unique plug sizes ranging from 3.5x1.35mm to 6.3x4.4mm. Since we are building from the ground up, we

can choose the plug size to whatever suits our design best.Typically, the plug of the adaptor has the positive polarity in the middle of the plug while the negative polarity is connected on the outside of the plug. This is better explained with the figure below:

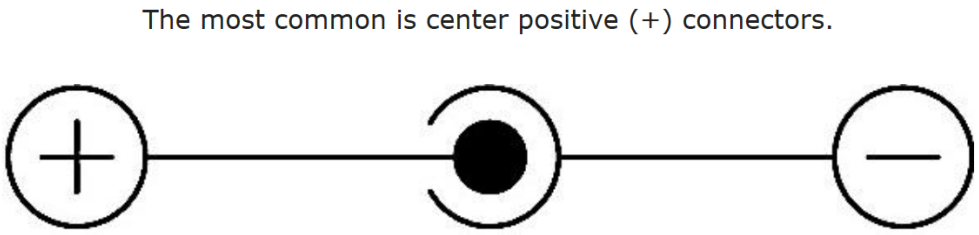


Figure 34: (permission pending)

Although it is less common to see the polarities flipped, it is something we will need to make sure about when purchasing the plug and the jack it goes into. Through our research we have come across two possible options depending on what our final design will need. Below in table # are two AC adaptors from triad magnetics. Both have the same size with the same plug size.

product	WAU12-1000	WAU12-500
Cost	$15.28	$13.85
Voltage output	12V	12V

Current output	1A	500mA
Efficiency	80%	80%

Table 3

In the next sections, we discuss our research of voltage regulators that will be needed to regulate the voltage for our design components from either the rechargeable batteries or the wall outlet.

3.3.4 Voltage Regulator

Some components of our design require constant voltage to operate. Whether the power is coming from the wall outlet or from the rechargeable battery, the voltage will eventually have to be regulated at a constant voltage for the certain components to function properly. To do this, a voltage regulator will be implemented to provide our operational amplifiers inside the AFE with the correct level of voltage to operate. If done properly, this will ensure that no component will be damaged from high voltages. There are two types of voltage regulators that we researched: Linear Voltage regulators and switching regulators.

3.3.5 Linear Voltage Regulator

A linear voltage regulator is a type of voltage regulator that maintains a steady output voltage by use of a regulating device that acts like a variable resistor. Linear voltage regulators are often called series regulators because the elements that are used to control the output voltage are configured in series. Following figure below shows the idea of a simple linear voltage regulator.

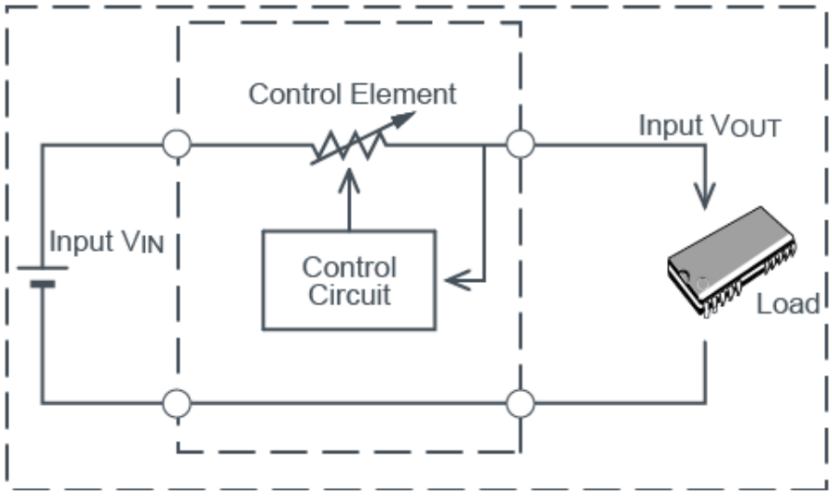


Figure 35: (Permission Pending)

The control elements vary the voltage drop to keep a regulated voltage. The simplest way to build one of these regulators can be done using a transistor and an operational amplifier. Even though this can be easily done, there are small three terminal regulators devices on the market that are simple to use. Some advantages of linear voltage regulators are the simple configurations which result in low noise. The disadvantages, however, are that these types of regulators have poor efficiency with a considerable amount of generated heat from the regulator. Also, this type of regulator can only be used to step down the voltage which limits its use in certain applications.

3.3.6 Switching Regulator

A switching voltage regulator is a regulator that varies its duty cycle at high speeds to turn on and off a switching element to output a constant voltage. The incoming supply of power is transformed into a pulsed voltage using a Pulse width Modulator. Figure 36 shows a basic diagram of a switching regulator.

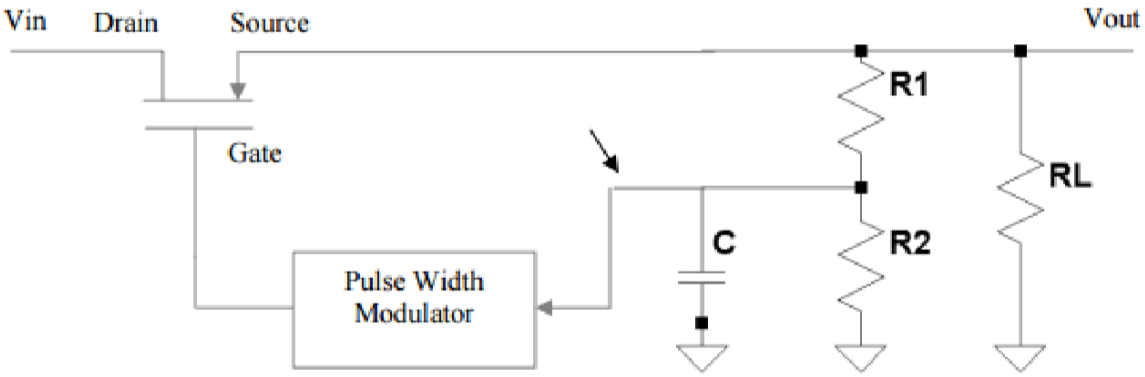


Figure 36: (Permission Pending)

Figure 3.2 shows a MOSFET working with a pulse width modulator(PWM) connected to the gate terminal. At high speeds, the MOSFET turns on and off rapidly to keep the output voltage constant. If the output reaches the regulated voltage level then the MOSFET turns off and when the voltage drops too low, the MOSFET turns back on. These methods of regulating the voltage gives the switching regulator some advantages. Some of these are high efficiency ratings which are a result of low generation of heat. Also, unlike the linear voltage regulator, switching regulators can not only step down the voltage but also step up the voltage giving it more possible application uses. These advantages, however, come with equal disadvantages. Higher efficiency ratings come with a more complicated design. A more complicated design then results in an increase of noise interference with a higher cost then the typical linear voltage regulator.

Regulator type	Linear	Switching

Efficiency	Typical < 60%	Typically 70-90%
Noise Interference	Low	High
Heat generation	High	Low
Cost	Low	Medium

Table 4

Comparison of Voltage Regulators

Supplier	TI	Polulo	Digi-Key
price	$4.49	$4.95	$4.31
Max voltage	5.5V	5.5V	5.5V
Min voltage	.5V	.5V	4.5V
Max current	600mA	1.2A	200mA

Table 5

3.3.7 Power Usage

Supplying power to all the components of our project is an essential task and needs to be one of our top priorities. Insufficient power will make our components useless and using the wrong level of voltage will destroy our components forcing us to spend more money. The power to our project will be supplied from either a 9V lithium ion rechargeable battery or from an AC wall adapter(12V) depending on what the user wants. Power will be evenly distributed to all parts of our ECG. First, we need to properly power the components of our analog front end. This includes the instrumentation amplifier, which will amplify the heartbeat signal, and the operational amplifiers, which will be used in the

filter stages to filter noise and interference. Both of these components can operate on 3.3 V which will be obtained from regulating voltage from either power supply.

3.4 Developing Software Selection

As Hardware Decisions are chosen Software applications and programming languages will go through the same agenda. Within this section a detailed description for every potential low level and high level language will be rehearsed, focusing on their limitations and capabilities. The strongest contender for software programs either high level or low level languages would then be determined by group vote on engineering requirements and prerequisites as well hardware compatibility. Depending on outcome adjustments, adjustments may occur from software standpoint, calibrations will always be performed and rounds of test will be simulated in order to acquire correct data points. Analysis would be performed through the design completion.

After considering time constraints for this project, we decided to start the software development in one platform first. Since there are only two major mobile OS platforms out there which are Android OS and iOS. Android OS is based on Java and is supported by many mobile phone manufacturers. On the other hand, iOS is based on Objective C which is a prioritized programming language from Apple. The number of Android phones out there is outnumbered iOS phones by a large margin. Since we are not experienced programmers, we would start with an easier option. We have decided to start with an Android Application because we are more familiar with Java.

There are many options for developing software. We specifically chose to build an Android Application that will work on any phone that supports Android OS. For this purpose of the task, we weigh options for several IDEs that would benefit us the most since we are not experienced programmers. We know that Android Application is based on Java as its developing language. Technically we can use any Java IDE to develop this Application; however, there are many IDEs out there that are dedicated for Android App development. We would like to have an IDE that supports the preview option, so that we can track the elements that we have visually.

We choose to develop an Android Application and use any supported Android phone as our EKG display device. Since it is easier to add more functions to an App than implement more hardwares that supports additional features for our design, we choose this approach over a dedicated LCD display. Android App which requires Java as a developing language, and Java is one of the more popular programming languages that are taught in school. With this advantage, it will be easier for us to start with an Android

App. We also consider to expand our design to support iOS devices if we have enough time down the road. The main function of this App will be the waveform display for the EKG machine, and additional features will most likely be software implementation.

3.4.1 IDE for Mobile Device Application Development

Android Studio is an excellent IDE choice for our Android App development. There are many more options, but Android Studio is an official IDE for Google's Android OS. It is free and also very easy to use. It has a built in graphical preview of every component while the App is being developed. The IDE is supported in many major computer OS such as Windows, Mac, and Linux; therefore, it is easy to collaborate in one software project from multiple platforms. Our group has two members who are assigned to develop the App that is served as the EKG machine display, and we use different computer OS. This compatibility trait of this IDE is the main reason that we choose this IDE over others.

The Android Application is the front end of the output after we process all the inputs from the sensors. Those inputs have to go through a series of filter and amplifier circuits before reaching the MCU which is a final stage of the input for the display. The MCU needs to be programmed to process the inputs, and it usually takes a different programming language as a developing tool. We would choose a MCU that utilizes C as its developing language because it is a basic programing language that most people know how to use. For this part, we have many choices for a good IDE to develop and test our MCU programs. The following section is the selection of the IDEs for the MCU that we take into consideration.

The integrated Development software Android Studio is the key tool to launch the application for any needed project in automation and sensor implemented designs. Though the new versions and updates are being developed for this freeware. The software becomes easier to use at every future time that early systems could not support. The basic System requirements for this tool require Microsoft windows 7/8/10 32bit or 64bit, Mac OS 10.10 or higher, or linux GNOME or KDE desktop with a minimum of 4GB RAM or 8GB recommended. The Screen Resolution has a minimum size of 1281 by 800

	Linux	Windows	Mac
Operating Systems	KDE or GENOME	Windows 7,8,10	Mac OS 10.10 + up
Required Version	JDK version 8 (Java Development Kit)	JDK version 8 (Java Development Kit)	JDK version 8 (Java Development Kit)

RAM (Random Access Memory)	4GB or higher 8GB recommended	4GB or higher 8GB recommended	4GB or higher 8GB recommended

Table 6: Android Studio Standard Requirements

3.4.2 IDEs consideration for MCU

PlatformIO (Latest Version 4.3.0)

C/C++ intelligent code completion and Smart code linter for professional development. Control Device monitor output with filters and text transformations fixes are constructed. The software platform is compatible for ARM processors embedded architectures with JSON support. Compatible with the Arduino ecosystem and contains correct dependencies and correct libraries for hardware within the library manager. JSON fault issues fixed from previous versions. Python language intelligent, and serial monitor implemented 15 and up Frameworks, as well as 30 and up Development platforms.

Eclipse

Developed by Eclipse foundation while languages can be written in Java and C. This software defines a set of frameworks which is a comprehensive tool for code integration. It is a great project model for managing additive compilers and language independent debugging infrastructure. The platform can be used for software libraries and languages like Scala typically used for Hadoop linux, PDT, and PHP. It is an open source freeware, However it does not have easy accessibility since linux and Solaris are the main operating system. Python , LaTeX. This platform supports database management systems and configuration management, which is a systems engineering process for establishing consistency of performance in a product. Not convenient for microcontroller use, and more for desktop applications. Can also be used in Mac Operating Systems

Sloeber

Sober is a platform plug-in for MCU software integration that uses JAVA as an interpreted language, meaning that when the program is run, then the application can only be used inside a java interpreter (Java Runtime) inside a Windows machine. This plug - in is handy; However, it is recommended to be used after experience has been acquired with starter platforms such as Arduino IDE before going along this track. The .jre version is available for Microsoft Windows users. A version for Eclipse is also available in

which the scripts can also be run on C / C++ that makes it perfect to launch an Arduino file role that can be installed in 1 package. Can be used in Linux and Mac Systems.

EmbedXcode

This the most widely used template within the Apple MacOS users. It is beneficial for embedded computing boards. It supports the Adafruit based boards which contain Atmel Atmega microcontrollers, Cortex - M0 - SAMD and Nordic nRF52 , Arm Cortex, Intel galileo, currie and edison boards. Same features as Arduino IDE with added functionalities. Supports C / C++, Embedded Xcode handles Texas instruments MSP430 MCU family

Zeus IDE

This IDE is constructed for the beneficial use of popular languages like C#, C / C++, Python and many others used in Microsoft Windows. It consists of plenty of configurable features and support for any toolset. It is easy to write code, with plenty of documentation and faster to deploy script times. It is one of the Top powerful environment platforms compared to others. Seamless Git integration, and FTP file editing, FTP secure protocols which includes SSH and SSL/TLS.
This contains smart pasting and editing which benefits from leaving the editor while building the code.

Atmel Studio

This is an integrated development platform in which applications can be developed and debugged in all AVR and SAM microcontrollers. Applications can be written in C/C++ and Assembly code could be used within compiler integration. It is easy to import Arduino sketches. It includes drivers and communication stacks. It has full Chip simulation for accurate model of CPU interrupts and interfaces for programming are easy accessible for various programmers hardware architectures inside the development boards

Code composer

An Integrated development environment which uses C / C++ languages and may be used for Texas Instrument Family packages MCU's: MSP Low Power MCUs C2000 Real-time MCUs, SimpleLink Wireless MCUs, TM4x MCUs, TMS570 & RM4 Safety MCUs, Sitara (Cortex A & ARM9) Processors, Multicore DSP and ARM including KeyStone Processors, F24x/C24x devices, C3x/C4x DSPs, Jacinto™ automotive

processors (TDAx/DRAx). This platform is able to debug code while a programmer is set into the design, also having the advantage of combining the Eclipse software framework, having a profiler as well as an editor.

Visual studio

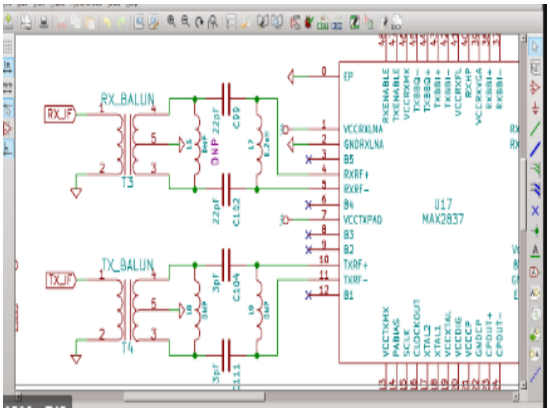
This development platform from Microsoft is quite different than others, in the fact that while analysing and integrating new commands, the user can be connected to Cloud. C / C++, C#, .NET, javascript, HTML, XML, CSS, and Python are amongst the most used out of the 36 different languages within this platform. The debugging can happen through call stacks and breakpoints instead of having to exit the editor window. Extensions can be added to the platform in order to complete requested functions for specific designs. They run in separate processes which do not slow down the machine, or editor. This platform can be easily run on Microsoft Windows, and MacOS and is convenient through processors because of Intel's cooperation with Visual Studio

3.4.3.0 PCB Design (Hardware) Software

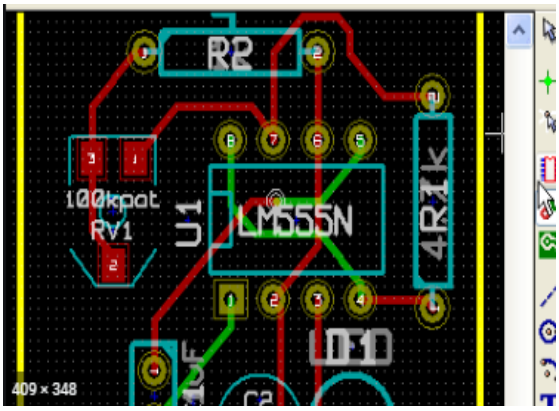
There are plenty of PCB design software platforms available for creating an electronics design for creating a marketable product that provide a sharp look and be a rigid design. The platforms that the group will consider regarding the experiments will be free of charge to students and two potential candidates are KiCAD and Autodesk eagle. The software should be an open source electronics suite without limitations and computer manageability for multiple operating systems. This main software would reduce the cost of the design where the money could better be used for more quality products across the ECG project other than this platform.

KiCAD

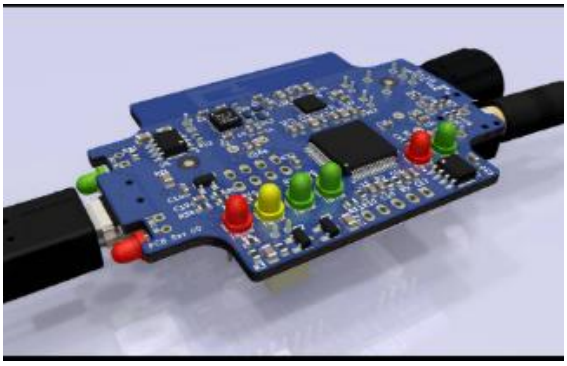
This is a free software PCB design platform used tof electronic automation design. It was created in 1992 and operated by a group of volunteers and paid contributors. This platform is composed of five main essentials. Project manager (KiCad), Schematic capture editor (Schema), PCB layout program (PCBnew), tool to convert images as footprints on PCB (Bitmap - 2 Component),and Gerber lookout viewer (Gerver-view).
KiCAD iis a cross platform that is composed using the C++ high level language that can be ran in Microsoft Windows Mac OS or Linux operating systems



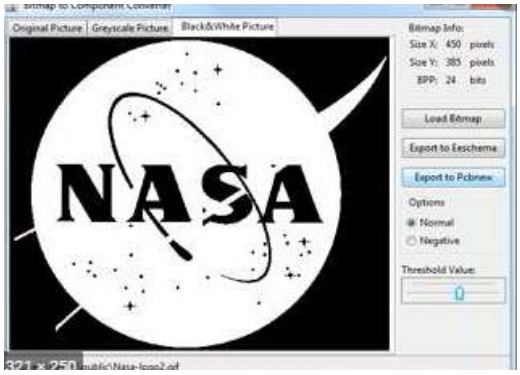
Schematic Capture



PCBnew



3D Viewer



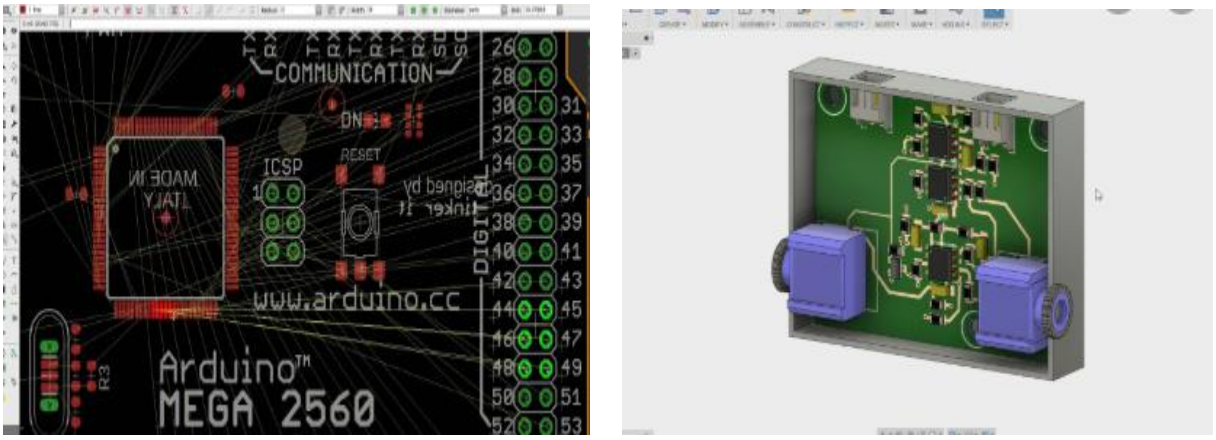
Bitmap2Component

Figure 37: (Permission Pending)

With the KiCAD software you can expect to transition between file formats in an accessible manner where file conversions like PDF or SVG are available for deployment. The gerber files can be produced in this application along with the bill of materials and integrated libraries.

Eagle

Eagle Autodesk is an electronic Design Automation application that can design projects through scripting. This platform provides similar applications through that of KiCAD but with several less extensions. The platform is a freeware for students that can be used in Microsoft Windows, Linux or Mac OS 64bit machines. The software contains a schematic editor for creating circuit layouts and computer aided manufacturing. Eagle Autodesk contains a lot of documentation that can easily be used with electronic circuit microcontroller modules such as Adafruit and Sparkfun, Easy EDA and other partners in electronics as long as the gerber extension files have been produced.



schematicPCB view 3D viewer

Figure 38: (Permission Pending)

TI MSP430 MCU Route

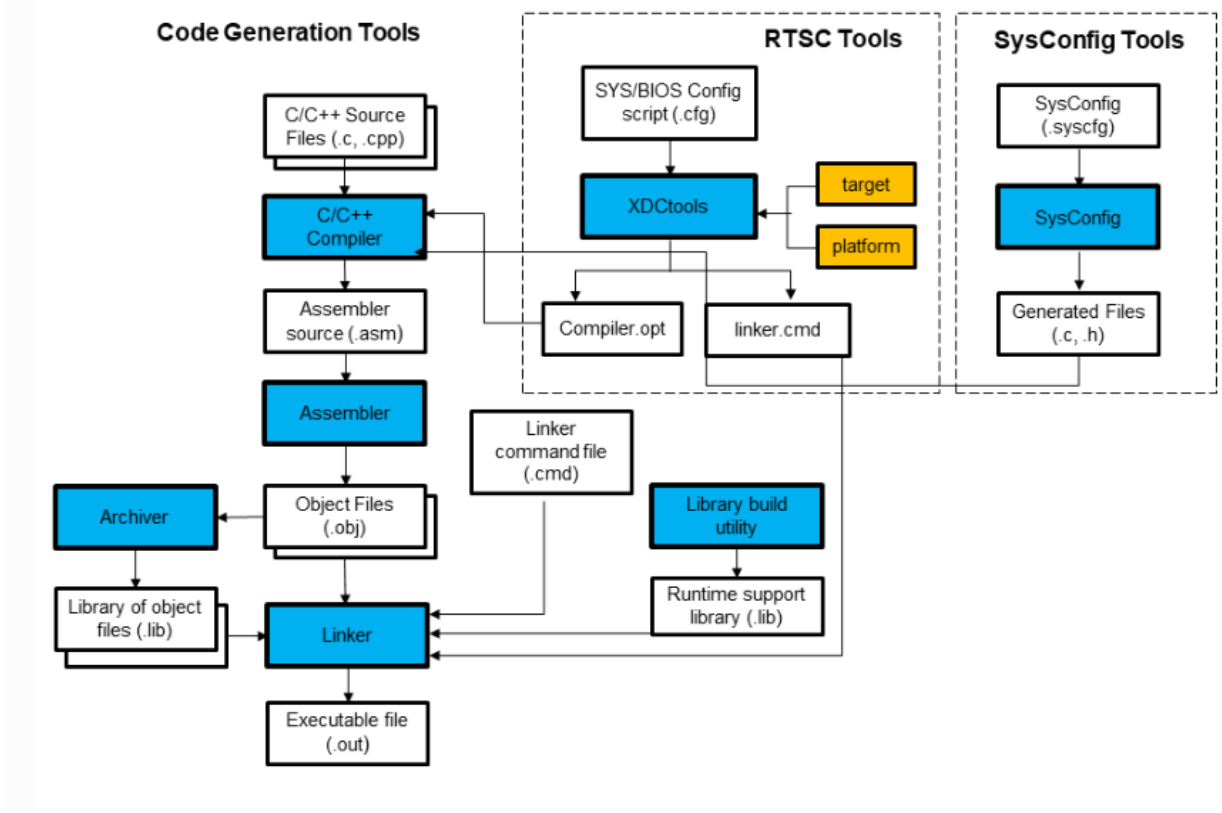


Figure 39: Microcontroller Design Software Table (Permission Pending)

A typical Build in a CCS (Code Composer Studio) Project requires deep understanding of the program structure. As shown on the picture above the blue highlighted boxes illustrate the tools and components while the white boxes are the input, output files. The Assembler in blue is the low level code program that is not shown through the user which interprets software written in assembly language. It performs the binary functions through machine language with Logical commands. The Code Generation Tools provide the user language interconnection between machine and user, which mostly requires maintenance and complexity The RTSC tool dotted box shows that they construct output files and then processed through the linker as well as the compiler. Lastly the Sysconfig tools create output files that will then be inserted in the compiler.

This design implementation software is used in unison with an Arduino microcontroller development board, developed by Aristotle University. The software was developed as a transmission from analog signals into a digital device, in order to observe compression of signals and perform analysis in external interferences (noise).

3.4.3.1 Discussing PCB platforms

KiCAD vs Eagle

Through several research journals and review commentaries it is safe to say that if one of the design tools is learned, then the transition between each other is not brutal. KiCAD is free software created by volunteers and its open source. Eagle which is owned by Autodesk only gives free software for students. However, it provides some limitations.

KiCAD seems to be more reliable because of its keyboard shortcuts, and for that you might find yourself to work more effectively with KiCAD. However, they are both very similar toolboxes that seem to provide the same service. Since in the University of Central Florida we were trained to use EAGLE in several projects. This might be the top one we may choose, but this might change if the implementation requires a KiCAD tool not provided by EAGLE's free student version.

After choosing the appropriate software to design the PCB, we need to focus on producing a PCB that will serve as a motherboard that holds all the electrical components of the front end hardware. The parallel process of this phase on the software development side would be choosing the developing software for the drivers of the PCB and also the front end of mobile Application for android devices to be used as the ECG display later.

3.4.4 Mobile Device Application Interfaces and Functionalities:

The GUI layout for our EKG display will have a standard autoload screen for the waveform output when the mobile device is connected to the machine, either via a USB cable or via Bluetooth connection. Since the main purpose of this project is mobile EKG self checking and basic diagnoses as a feature, there will not be any database to store users' information in the background. Therefore, the main GUI of our App will be minimal compared to others. The app will support additional features such as heart rate monitor and some basic heart conditions detection. Following is the sketching of the main screen for the GUI of our App (figure 13). There are two buttons on the left side, and there are two sub-screens for heartbeat monitor and temperature on the right side. The extended screens will be activated after the on screen button clicked.

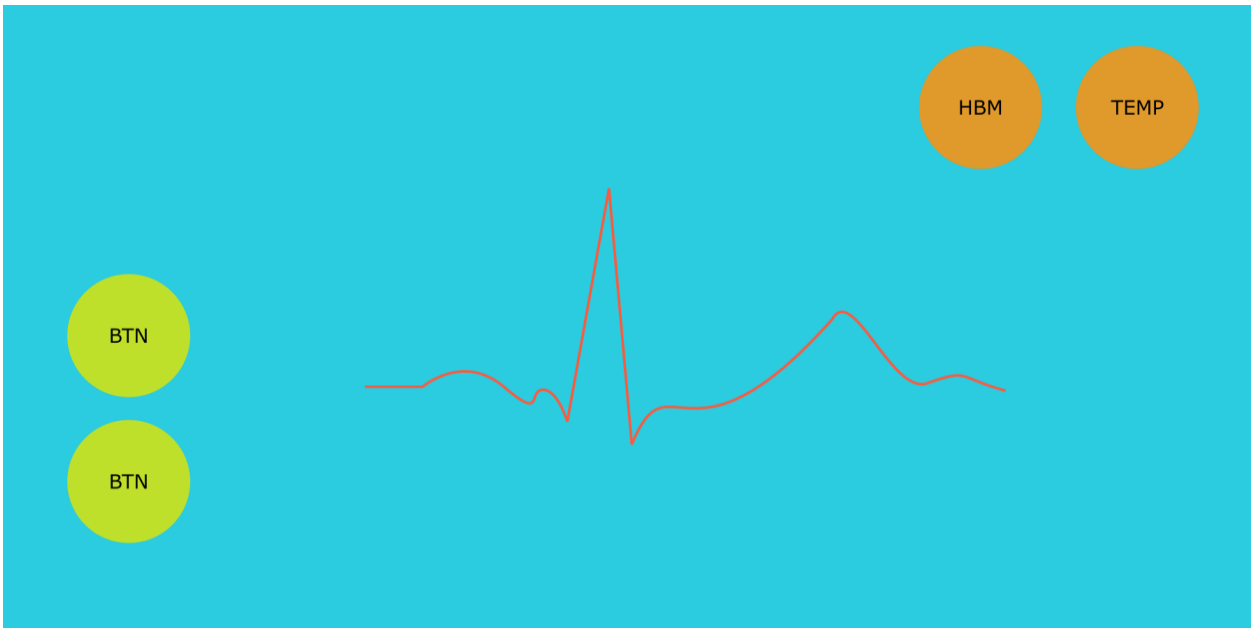


Figure 40

The home page of the App will be a full screen waveform display with some on screen buttons to activate additional features. The waveform output window will be either built on HTML5 technology or the latest flash player that Android Studio supports. HTML5 is a newer and more advanced technology that works much better than flash player. Graphical display with smooth refresh rate is required for more accurate waveform representation. This requirement is also required for better heart problem detection feature since we use recorded on screen waveform to compare with referent pictures to determine the conditions. Besides the refresh rate requirement, the processing ability of the mobile device to process background activities of the App quickly is also very important.

Background processes that have registered to listen for an implicit broadcast even if these processes may not do much work. This can cause a really bad impact on both device performance and user experience. If the app uses any extensive broadcast, developers should consider implementing a better background optimization. Since Android 7.0 (API level 24), the system restricts connectivity action broadcasts if they declare their broadcast receiver in the manifest. Apps cannot send or receive ACTION_NEW_PICTURE or ACTION_NEW_VIDEO broadcasts. This affects all apps, not only those targeting Android 7.0 or higher. When we develop our app, we have to consider these restrictions to ensure our background processes are optimized.

Android Application is built as a combination of individual components that can be invoked separately. When users click on the App icon, the main activity is invoked. Other components, such as broadcast receiver or services, help the background tasks to be performed without the UI. Android environments allow developers to provide different resources for different devices. We can specify our app's feature requirements and control which types of devices can install our app. However, we aim to create a dynamic Android app that will work on a wide range of android devices. For example, a robust App will have to work in different layouts for different screen sizes. The system determines which layout to use based on the screen size of the current device.

Android Apps can be written in Java, C++, and Kotlin. The code and any of its data, resource files are compiled into the APK (Android Package) by the Android SDK. One APK file contains all the contents of an Android App and is the file that Android supported devices use to install the App. Each Android App exists in its own security space called sandbox. The sandbox is protected the Android security features such as:

- Each Android App is a different user in the Android operating system which is a multi-user Linux system.
- The system assigns each app a unique Linux user ID by default. The system sets permissions for all the files in the app so that only the user ID assigned to that app can be accessed.
- To ensure that an app's code runs in isolation from other apps, each process has its own virtual machine.
- Every app runs in its own Linux process by default. When any of the app's components need to be executed, the Android system starts the process. The Android system also shuts down and recovers memory of the process when it is no longer needed.

The principle of least privilege is an Android implementation that gives an app access only the components that it needs to do its work and no more. This creates a very

secure environment in which an app cannot access part of the system for which it is not given permission. However, There are many different ways for an app to share data with other apps and to access system services:

- Sharing Linux user ID is possible for 2 apps, in which case the apps are able to access each other's resources. They can arrange to run in the same process and share the same virtual machine to conserve system resources. They must be signed by the same certificate.
- Given permissions explicitly by users, an app can access device data such as GPS, camera, connections.

Android is designed to run on many different types of devices, from phones to tablets and televisions. Android provides a dynamic app framework in which developers can provide configuration specific app resources such as XML layouts for different screen sizes. The system then loads the appropriate resources based on the detected device's spec. With some forethought to the app design, developers can publish a single package (APK) that provides an optimized user experience on a variety of devices. Different devices may run different versions of Android platforms. Each platform version specifies an API level. The API level allows the developers to declare the minimum version which their app will be compatible with.

Every Android app has a main thread which is in charge of handling UI, coordinating user interactions and receiving lifecycle events. Any long-running operations should be done on a separate background thread to keep cohesive user experience. Some of these tasks may be required to be performed while the user is actively interacting with the app. Applications may also require some tasks to run even when the user is not actively using the app. They may also require services to run immediately to completion even after the user has finished interacting with the app. Background tasks consume a device's limited resources such as RAM and battery. This may lead to poor user experience, if the background tasks are not handled correctly. In order to conserve these resources, the system restricts background work when the app is not actively used by the user.

The heartbeat monitor feature is based on the real time process of the waveform to detect the R peaks of the ECG cycles. The unit of measurement for this feature is BPM (Beats Per Minute). This feature does not update the counter every minute, but it should update on the change of the heart rate while the test person is being measured. Because this feature is very popular in many heart rate monitor devices, we have many algorithms out there for us to choose to implement for our design. However, since it is only an

additional feature, we prioritize optimization of the code that requires the least resources and the most lightweight algorithm.

There is an additional screen that is activated after one of the on screen buttons clicked. For now, we only planned on one additional feature. Using the recorded image of an ECG cycle from a test subject to detect some basic heart conditions is one feature that requires a lot of medical knowledge and coding experience. The following section is detailing the plans that we consider to implement this feature. However if by the time that the hardware is fully working, this feature proves to be too challenging for us. We may abandon this feature.

3.4.5 Mobile Device Application Feature

The feature of heart problems detection requires some additional software for edge detection and comparison. The on screen recorded waveform will be saved to a temporary file as a picture of one cycle of the ECG, and the background process will compare the recorded picture with the pictures in a stored pictures database. This process sounds very simple, but it requires quite a lot of processing power from the CPU. Background processes can be memory and battery intensive. This can have a substantial impact on both device performance and user experience. This problem is noticed by Android developers, and they release updates to help app developers to optimize background tasks within the system. Started with Android 7.0 (API level 24), the system will restrict apps that have extensive background processing.

The first step of this process is taking a picture of one cycle of the ECG waveform and then saving it to a temporary file. It is a challenge to develop an algorithm that can detect and take a picture of a full cycle of the ECG from the continuing waveform. Since the characteristics of some conditions that have minimal change in the regular shape of the components of one ECG cycle, a high quality picture is required to work with a good edge detection algorithm. In the second step of the process, the edge detection software will run in the background to create a map of the recorded picture and then save it to another temporary file. This map file will be compared with the stored database that contains the map files of some regular heart conditions. This is the third step of the process. If the map file matches with one of the store files, there will be a result window popped up that has additional information to inform the users about their condition. The flowchart below shows the basic steps of this feature.

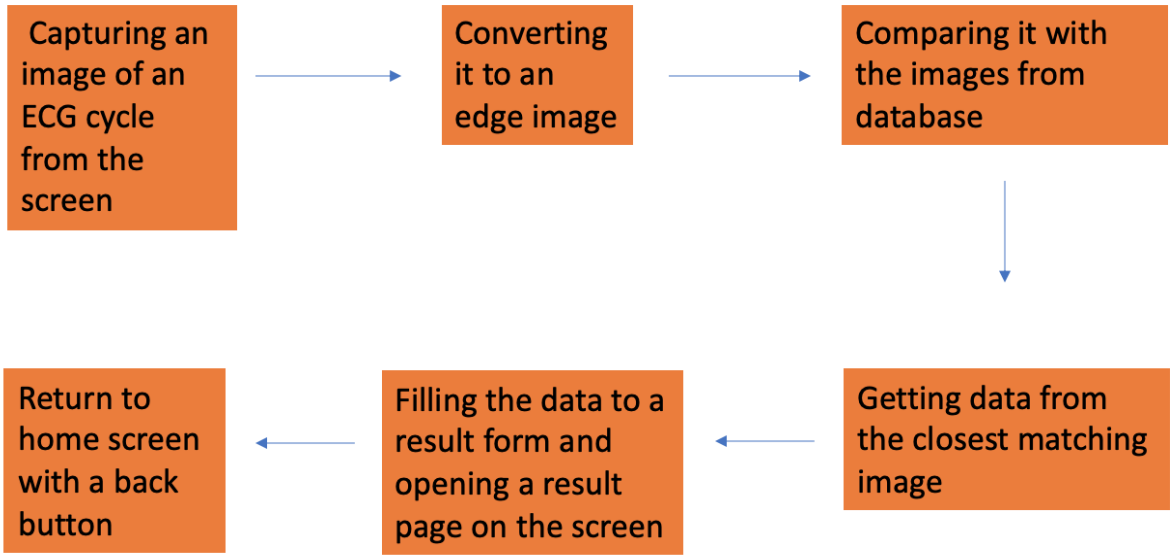


Figure 41

Edge detection algorithm selection is based on the needs of the application. For our need, we want an algorithm that is not taxing heavily on the background process. Since the pictures that we need to compare consist of simple edges, we do not need a sophisticated algorithm for this task. However, the accuracy of the edge maps is very important to determine the correct condition. There are two good options of edge detection algorithms that will probably be good for our project: Sobel and Canny edge detections. Sobel is a very light algorithm that fits well in mobile applications, but it does not have the accuracy that Canny offers. Sobel algorithm computes the gradient magnitude of an image with 3x3 filters to compare the threshold from fixed filters. Canny algorithm takes an extra filtering step before using the same method of Sobel to create an edge map. The Canny algorithm uses a low pass filter to reduce noises then applying matrix filters like in Sobel. After that, it uses a non maximum suppression to pick out pixels that are most suitable for edges from many possibilities to create a gradient picture called edge map.

The images of some basic heart problems are stored in a small database within the mobile App. These are pre compiled images after the original pictures were processed by the same edge detection algorithm. These images are represented as matrices of

integers that can be easily compared with a referent picture to find a close match. Each stored image has its own name and attached information that can be shown in the result window to the user when a match is found. Different mobile devices have different screen sizes, so we need to take that into consideration when we take the picture of the waveform. The taken pictures need to be scaled to the size of the referent stored pictures before going through the edge detection process. There will be another algorithm for comparing the processed images. The values in the new taken picture will be compared with those in every matrix in the database. The differences will be stored in a new matrix with certain thresholds indicating certain heart problems.

This approach gives us a lot of advantages such as flexibility in expanding features and appealing for portability; however, it also gives us a lot of challenges. Since our group does have any computer science major, we have to learn a lot to keep up with the demand of programming required for this approach. We are still in the planning phase, and we will weigh the options after the hardware is fully implemented. With the main feature that requires a lot of experience with programming and medical knowledge, we have a major challenge in implementing it.

3.4.6 Challenges of Mobile Device Application

With these considerations for the additional features, they pose a lot of challenges for us to fully develop the mobile Application that will support multiple OS platforms. The Android App will be focused on first to make sure that we will have a working EKG display. The fully functional feature of heart problems detection will be added or removed based on our progress in the coming months. The feature of a heart rate monitor is simple enough for us to be guaranteed to add. So far, these are the main features that we considered. There are some other good ideas that we may add such as a temperature measure or an alarm system for users with serious heart conditions.

Since we have multiple components that serve different purposes within an App, we have to make sure that they do not create too much resource competition while the App is running with other Apps that run in the background of the mobile device. Even though newer devices accommodate a lot of resources such as more RAM, faster CPU, they are still limited for the sake of efficiency. Because if an App is not working efficiently while there are many Apps running at the same time, the mobile device will throttle the performance of the App that requests the most resources. Therefore, optimization is a challenge for us to make this App working as efficiently as possible.

App components are the building blocks of an Android app. Each component serves as an entry point for the system or a user to enter the app. Some components

work independently, and some components depend on others. There are four different types of app components. Each of them serves a distinct purpose and has a distinct lifecycle that depends on how the components are created and destroyed:

- Activities
- Services
- Broadcast receivers
- Contents providers

An activity is the entry point for the user to interact with an app. A single screen is a representation of an activity with a user interface. Many activities combine together to create a cohesive user experience, however; each of them is usually independent of the others. An activity facilitates the following key interactions between the operating system and the app:

- Tracking the user current focus (on screen contents)to ensure the system keeps running the process that is hosting the activity.
- Keeping records of previous processes (stopped activities) in case the user wants to revisit previous activities.
- Helping the app handle its process killed, so if the user wants to return to previous activities, the previous states would be restored.
- Providing a shared environment for apps to implement users flows between each other, and for the system to coordinate these flows.

A service is a general purpose entry point for the app that keeps running in the background. This component performs long running operations or work for remote processes. A service does not provide a user interface. For example, a service may keep the system updated while the user is using a different app. Another component can start the service and let it run or bind to it in order to interact with it. Bound services are dependent processes that one process keeps running for another process by providing an API from one process to another. Because of their flexibility, services are a really useful building block of many kinds of high level system concepts. Many applications such as notification listeners, screen saver, input methods, accessibility services, and many other core system features are implemented through services, and the system binds to when they should be running.

A broadcast receiver is the component that enables the system to deliver events to the app outside of regular user flow, and it allows the app to respond to system-wide broadcast announcements. Because it is also a well defined entry point for an app, the system can deliver broadcasts even to apps that are not currently running. For example,

an alarm app that posts notifications to tell the user about an upcoming event. By delivering that alarm to a broadcast receiver to the app, the app does not need to be running until the alarm goes off. Many broadcasts originate from the system, but apps can also initiate broadcasts. Although broadcast receivers do not display a user interface, they may create a notification status bar to alert users when a broadcast event occurs. Moreover, a broadcast receiver is a gateway to other components and is intended to do a very minimal amount of work.

A content provider manages a shared set of data that developers can store in the file system. Through the content provider, other apps can query or modify the data if the content provider allows it. For example, the Android system provides a content provider that manages a user's location. As such, any app with proper permissions can query the content provider to read the user's location. To the system, a content provider is an entry point into an app for publishing named data files, identified by URL scheme. Thus, an app can decide how it maps the data it contains to a URL namespace. There are a few particular ways that system use to manage URL in apps:

- An URL is assigned even if an owning app is not running, so URLs can persist after their owning apps exitted. The system only needs to ensure that the owning app is still running when it has to retrieve the app's data from the corresponding URL.
- These URLs also provide a deep level of security model.

A unique aspect of the Android system design is that any app can start another app's component. When the system starts a component, it starts the process for an app and initiates the classes needed for the component. Unlike apps for most other systems, Android apps do not have a single entry point. Because the system runs each app in a separate process with file permissions that restrict access to other apps, an app cannot activate a component of another app directly. However, the system can do that. In order to activate a component from another app, the user needs to deliver a request to the system that specifies the permission to start a particular component. If the permission is granted, the system then will activate that component.

Android uses a file system that is similar to disk-based file systems on other platforms. The system provides several options for developers to save their app's data:

- App-specific storage.
- Shared storage.
- Preferences.
- Databases.

To choose the right solution, developers need to find answers for the following questions:

- **How much space does your data require?** Internal storage has limited space for app-specific data, so considering a solution for app data storage from the beginning of development is a good practice.

- **How reliable does data access need to be?** App-specific data that are stored in external storage are not always accessible because some devices allow users to remove a physical device that corresponds to external storage. Therefore, if we need our app data to be accessible while it is in use, we need to ensure that the app data to be stored in an internal storage directory or database.

- **What kind of data do you need to store?** For structured data, use either preferences or database. For shareable data, use shared storage so other apps can access the data. Our app data is only meaningful for this specific app, so it should be stored in app-specific storage. Since there is no shareable media content, using shared storage is a not a good approach in our case.

- **Should the data be private to your app?** When storing sensitive data which should not be accessible by other apps, we should use internal storage, preferences, or a database. Internal storage also has the benefit of hidden data from users.

Both of our software development team members do not have much experience with mobile Application development. We have to put more effort into learning how to make a mobile App that will not only work, but it has to work efficiently. Our team does not have any computer science members, so software development is a new subject that we have to learn fast to fulfill our part in this project. Since we are working on this part while we are still learning, it will most likely be that we are going to make a lot of mistakes. However, this is still a planning phase before we actually work on the project. There are still chances that we may have to change the entire approach for the ECG display. Due to the constraints of time and budget, we take this approach into consideration as our best option as of right now.

One feature that we choose to implement is the self checking for heart conditions. This feature requires some extra processing that will add into the challenges of this project. The ECG image that the App captures will be used to compare with the referenced images in a saved database within the App through a process called edge

detection. The algorithm for edge detection is complicated for us since we are not software engineering majors. Adding this algorithm to the code for the ECG display may create some bugs that may compromise the main objective of the App which is the ECG display. We will have to wait until the hardware is fully assembled before we can test the main function App. After the main function is working properly, then we will add this feature and test if any unexpected bug occurs.

3.4.7 Testing Procedures for Mobile Application

Android Studio supports a visual preview of the front end design of the App, so we can actually see what we have while we are working on the program. The main screen of the App consists of a full screen of ECG display and a few buttons to navigate to different features. Due to inconsistency of binary codes in different devices, after every build we will test the basic functionality of the App on multiple devices to make sure that the App will work on every supported device. The final build will have to wait until the hardware portion of this project is finished to continue the testing. The following chart shows the testing procedure that we will utilize:

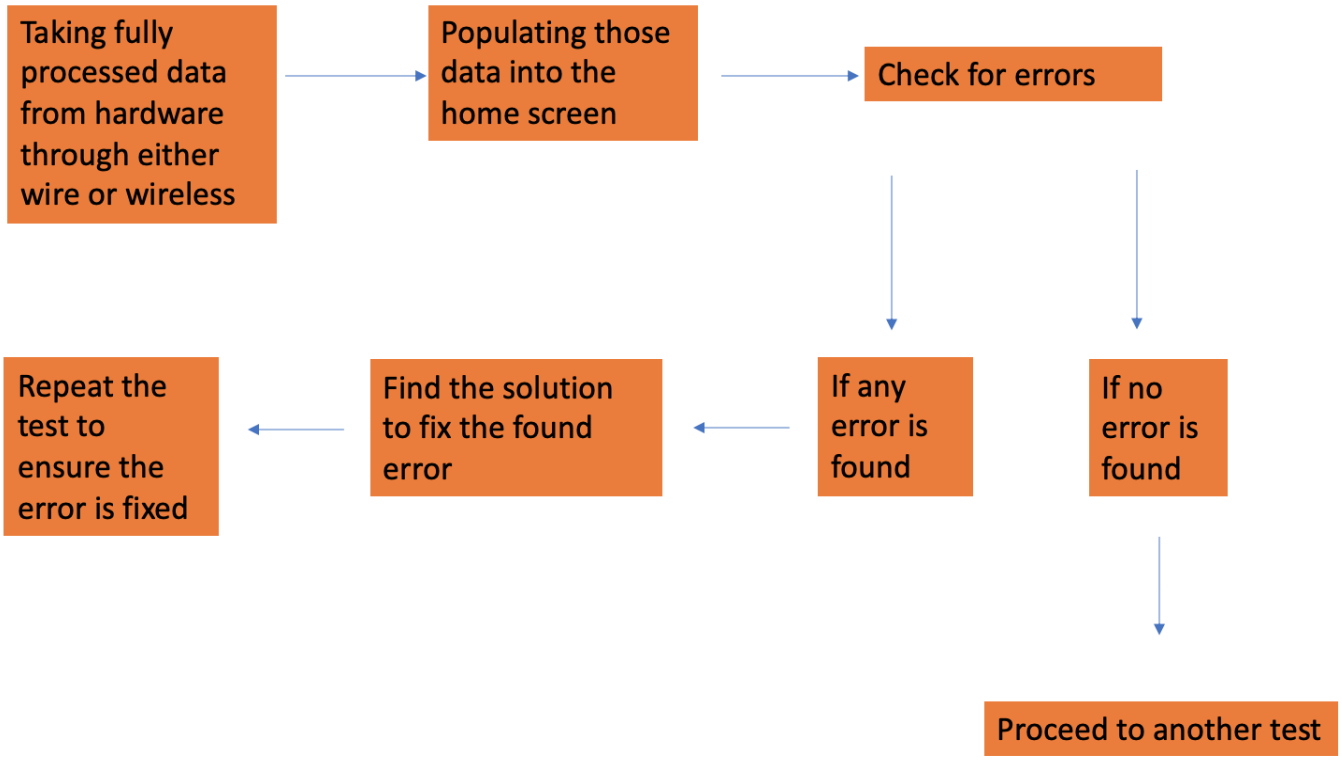


Figure 42

When the hardware portion of this project is working properly, we will start the testing phase on the App. The App only takes fully processed data from the hardware and displays the waveform on its main screen. There will be inconsistency in input data value at the beginning that requires multiple adjustments in the code to make it working. After the main functionality is working properly, we will start working on the additional feature/s. The main feature is a challenge for us, so it will be tested with stock photos without our hardware input. This process will take at least a couple of weeks. After the code for the additional feature is working properly, we will continue testing with the input from our hardware. We will have to dedicate the last 3 weeks before the final showcase of the prototype for testing the software design together with the hardware.

4. initializing procedures for GUI software

- Go to the Android developers website for download and proceed with reading and accepting terms and conditions
- Open Android Studio freeware once downloaded
- Follow steps to install android studio and launch it
- Click on do not import settings and continue with installation
- Once installed create a new android studio project and create an application title.
- Choose between using Kotlin, JAVA or C++ for coding development and finalize folder creation.
- Once all above procedures have been performed, the window should look something like the figure below.
- Initialisation of android application can now be developed
- Open a new project and name it with a desired title.
- Set the correct location for the file to be saved to.
- The phone and tablet are the only boxes that should be checked.
- When creating an app on the phone or tablet make sure that the SDK requirements are set under the correct level of the operating system.

The SPI Interface

A most probable interface that can be used in a wide variety of projects and could be used in the ECG/ EKG implementation would be the SPI protocol. SPI stands for Serial Peripheral Interface “SPI” for short is an interface that gets transmitted through synchronous serial communication. This type of communication usually works best in short distance transmission such as embedded microcontroller systems. It was initially developed by a multinational telecommunication company named Motorola and became one of the standard types of communication within the electrical and electronics field. The interface is one of the top interfaces where it can be used to send data between sensor modules, SD cards, shift registers, small computers on a single chip, and multiple other devices and components. Research was done on several protocols to acquire the best performance and result for the ECG project. However, before a schematic can be constructed, an understanding of the protocol needs to be set in place.

A 4 wire SPI protocol is a point to point communication system with two or more devices where communication can happen with the devices in both directions. a two-way communication synchronous master slave interface. The digital signal can be triggered between rising edge and falling edge sites, while the data can be passed between devices at the same time without interruption.

4 Interface pins:

MOSI – Master Out Slave In

MISO – Master In Slave Out

SCLK/SPI – Serial Clock

SS/CS – Slave Select/Chip Select

The clock signal being generated by the apparatus. The SPI protocol is a good transmission interface because it can accept higher frequencies than other protocols. They can also have lots of slaves but only one master. With the chip select signal a slave device can be chosen selectively at a time from the master by a high or a low digital offset. If multiple slave devices are used at the same time a designated slave, a select signal is transmitted for each slave. MOSI (Master Output Slave Input), and MISO (Master Input Slave Output) are the transmission lines where the data flows.

A 3 wire SPI protocol is also the data transmission communication protocol; However, the protocol is only half duplex since it is missing a pin. The transmission gets initiated by setting the pin to either “lo” or “hi”. This happens through a bidirectional transmission. If the setting is placed to read then the slave transmits the signal while the master receives it, and if setting is written, then the master pin transmits the signal while

the slave pin receives it. The master can interchange information through various slave devices because each slave device has its own chip select pin.

The master and slave have to be set on the same routine in order for them to exchange information. While the protocol has four operational modes, they are all based on two parameters which are the clock phase and clock polarity. Depending on whether the clock is set to high or low then different functions would be performed. If the CPHA from the SDIO

3 Interface pins:

SDIO – Serial Data In/Out

SCLK/SPI – Serial Clock

SSn/CSn – (Slave Select/Chip Select) HI/LO

When the Data transfer is initiated, a signal is sent from a master device onto one of the slave devices by enabling the slave select pin the signal from the master is active low and the signal is triggered high when a break wants to be performed between the master and the slave devices. With the SPI protocol synchronization can be performed and changes can be made while a positive edge trigger, or a negative edge trigger happens.

Polarity in SPI Transmission:

The number of bits of data that could be passed through the bus can be seen on the datasheet of the device being worked on in this case would be the MCU for sensor manipulations in the ECG. The master also has the advantage of choosing clock phase and polarity. There are four modes with CPOL (Clock Polarity) and CPHA (Clock Phase): SPI mode 0, SPI mode 1, SPI mode 2, SPI mode 3.

With mode 0 clock polarity is at 0, clock phase is at 0 and clock polarity in an idle state is logic low, data is sampled on a rising edge and it is shifted to a falling edge.

With mode 1 clock polarity is at 0, clock phase is at 1 and clock polarity in an idle state is logic low, data is sampled on a falling edge and it is shifted to a rising edge.

With mode 2 clock polarity is at 1, clock phase is at 1 and clock polarity in an idle state is logic high, data is sampled on a falling edge and it is shifted to a rising edge.

With mode 3 clock polarity is at 1, clock phase is at 0 and clock polarity in an idle state is logic high, data is sampled on a rising edge and it is shifted to a falling edge.

The rising and/or falling edge for the clock shift sample data is contingent on the clock phase, thus making the master select the clock phase and clock polarity and depending where the parity bits sit then various options are displayed for different modes explained.

Interface Wiring Options:

The SPI interface contains two daisy chain features being the Basic serial communication Interface, and the Daisy Chain option.

Basic serial communication Interface:

Devices in a Serial communication protocol require hardware and software changes in order to fit several needs in an overall design. Different types of architecture modes. Various electronic devices are not selectively addressable and for this reason, a basic serial communication protocol is a good option for it being standard.

In A regular Serial peripheral interface option when the chip select signal of a device transmits a logic “0” then this means that the master and slave devices are ready for transmission between specified slave devices. In the scenario when two or more slave devices are transmitting data at the same time onto the master device, then the transmission becomes invalid due to failing to detect the slave device that is transmitting the signal, the data sent is received as clutter. However, one of the big advantages of this setup is being able to add as many slave devices as possible below 256, and more if combined slave with another set as daisy chain.

This protocol is not limited but it is on the other hand restricted because of the number of pins on the microcontroller. Slave devices can be added in a large quantity but as that happens then the number of chips select lines will have to increase in the SPI master device. Like one stated previously there are multiple ways to up the number of slave devices in a design. Another would be a common technique which is a multiplexer.

In Our case for the ECG design, one of the key focuses is to try to minimize the surface area of the PCB and for that to happen smaller components would need to be used and less pins if possible. If in the design the group can come up with a plan to bring forward a smaller microcontroller and add a mux device to it to include various peripheral devices, then this could be kept in mind when reducing space for our machine.

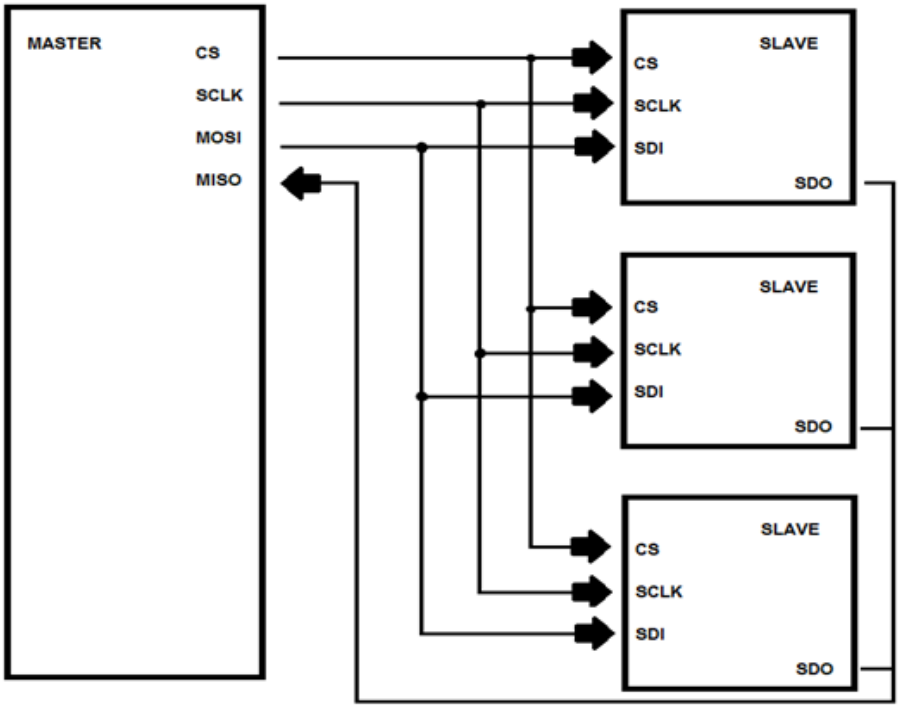


Figure 43: (Permission Pending)

Daisy Chain option

For a Serial Peripheral original interface, the SDO (Slave Data Out) went into one bus which led to one pin straight onto the microcontroller. All the slave devices were connected to the same bus for this data information. For the Daisy Chain Method, the data the pins are the exact same as from the previous method except for the data bus for serial data out is arranged differently. The serial data out from the first slave device does not go into a bus bar and then into the MCU, this time the slave output information gets transmitted straight through slave serial data input of the following slave device. For this method all the slave devices get the same exact serial peripheral interface clock, and the same corresponding chip select.

It is stated through the Motorola community that the number of clock cycles that are needed to send information, is directly proportional to the slave location in the daisy chain trend. Also since SPI daisy chain option requires slave devices to combine imputed data and outputted data from predecessor devices then not all devices support the daisy chain function; Therefore, it is best beneficial to peruse through the correspondent datasheet when a component is being placed to use to check for SPI daisy chain capabilities.

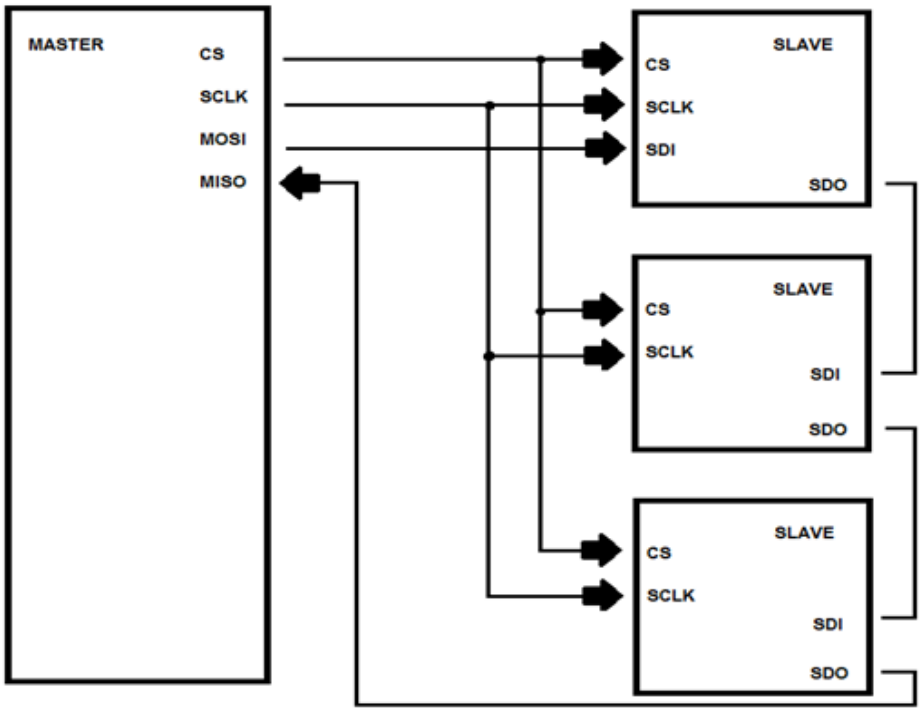


Figure 44: (Permission Pending)

There are several different other devices that relieve the problem of having multiple GPIO on the microcontroller and worrying about the hassle of making sure enough pins are available. Extensions like an SPI enable switches provide high performance where a serial to parallel converter is set in place to act as an intervention between the slave and the master device. The system outputs signals to other devices where the devices can be controlled through SPI. Multiple more slave devices can now be placed into the ECG design if desired by performing the daisy chain method.

Comparing the Serial versus parallel method of the interface is the speed. SPI protocol is efficient because of its high-speed capabilities and other multiple functions like multiple slave device implementation. The slave devices are also simple to acquire because of its simple adaptability with hardware and software. However, one of the downfalls is that the clocking inside the slave is not always sync together at a given time causing problems for the device's communication. These issues can also cause noise interference which causes defects in the system.

I2C protocol

Another alternative to SPI protocol is I2C (inter integrated Circuit) which was created by Philips semiconductor now NXP semiconductors. This protocol is half duplex due to the fact that it only contains two wires. It is synchronous meaning that the changes that happen in memory components are synchronized in line by the same clock.

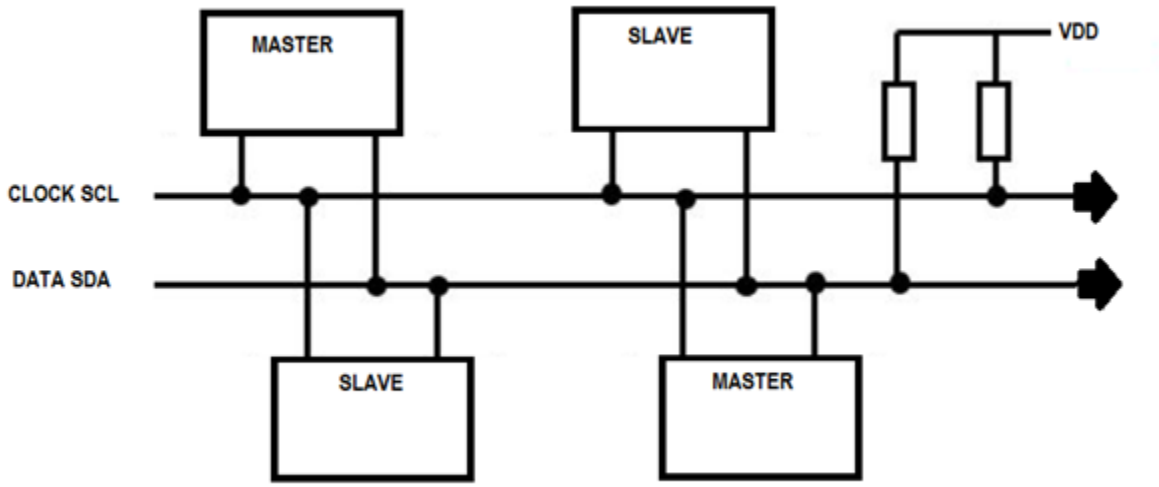


Figure 45: (Permission Pending)

This protocol has a huge advantage over other types of protocols in the way that it works on packet switching where the data is enclosed into packets for transmission into a serial manner instead of sending data parallel requiring multiple wires. When using this protocol multiple master and/or slave devices can be placed in the bus without any difficulties.

There is a serial and parallel method within the I2C protocol, where the address bits can be sent in a single or multiple wire at a time.

The general rule of signal communication:

Chip select is not required for point to point serial data communication When no external clock signal is used or required it is called Asynchronous. Read and Write and chip select pins can share the same bus wire when performing different functions. Slave devices cannot talk when they are not instructed by the master device, and slave devices cannot communicate with one another before passing data into the master device first. When the number of slave devices increases so does the number of wires.

Parallel on the ECG design is good for our multiple components but it contains many wires. Using the Clock line and the data line transmission could happen. It all depends on the data-line, it all happens through open allocations. A variety of peripherals on the same bus could be added like an Arduino, a temperature sensor, an EEPROM, an accelerometer etc…

The way it starts it is through a start bit being 1 bit going from master device to slave device. The clock is set to 0. After the start bit is sent then the control line is sent, the serial data contains the data bit for the address, but the serial clock line is always doing a pulse width 1 and 0.

The Standard baud rate for the I2C protocol is 100k bit per second and for full speed is 400kb per second. The correct order in which the data transmission is performed is shown:

Start……1 bit……S register

Device save address……8 bits……..A6, A5, A4, A3, A2, A1, A0

Read/Write……1 bit……R/W

Acknowledge……1 bit……A register

Internal register Address…….8 bits……..B7, B6, B5, B4, B3, B2, B1, B0

Acknowledge……1 bit……A register

Data……8 bits……..D7, D6, D5, D4, D3, D2, D1, D0

Acknowledge…… 1 bit……A register

Stop……1 bit……S register

UART communication

UART (Universal Asynchronous Receiver Transmitter) is a circuit packet responsible for setting up serial communication, in other words, it acts as a conversion device from serial to parallel data or parallel to serial. While the UART is usually a system that can be implemented outside of the MCU it is already implemented in multiple microcontrollers.

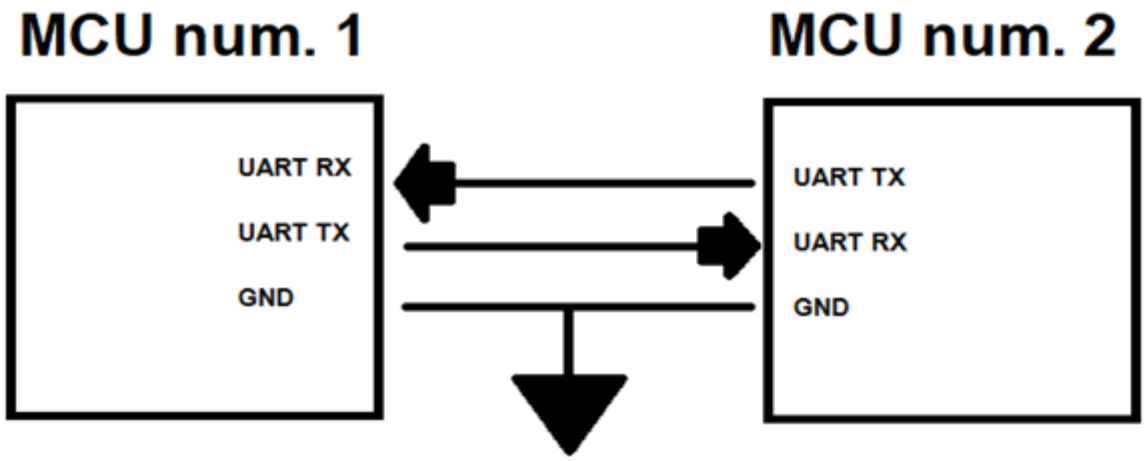


Figure 46: (Permission Pending)

The UART interface is an economic way to perform communication between two devices because it is Asynchronous. It is high speed, contains simplistic hardware and requires a receiver and a transmitter receiver to initialize the communication. Although UART contains lots of good benefits and it is commonly used for internet access in computers and used for slow services, it is starting to be replaced by more sophisticated protocols and transmission mediums such as Ethernet and USB for Personal Computers and SPI, or I2C for integrated chips.

Programmable setting alternatives

Parity : Odd, Even None, Mark, Space

Data : 7 & 8

Baud Rate : 300, 1200, 2400 ,4800 ,9600, 19.2k…

In order to implement a UART transmission in the ECG design parameters need to be set. A seven data bit 300 baud rate 1 stop odd or even parity bit could be set. With a parity generator having a D flip flop being imputed to it with a 7-bit data line an array has to be built.

The transmitter as a state machine has the information from the host being received while the 300Hz timer and a modulator counter work alongside for the UART transmitter to work properly. A shift register is receiving information from the transmitter state machine in order to perform the shift operation and to load the information into each block. The parity bit needed for the shift register comes from the parity bit generator where

the generator has a parity select where it generates a parity-bit according to the selected value of the input. A parity bit generator can also be built from 8 XOR gates that by using Demorgan's law it can perform the logical switches 0 and 1 for even or odd parity-bit

Microcontrollers / MCU

Micron Infineon

The XMC4500 relax kit and the XMC4500 Relax lite are development evaluation boards which are used in the XMC4000 microcontroller. The key piece in the microcontroller is running on 120 megahertz with a ARM cortex m4 SOC chip. It contains 1 gigabyte flash integration with a micro - USB plug and LEDs integrated on the board for feedback. Two headers on the borders of the development board are mapped to the communication ports from the chip. The board can be powered up by a USB plug or voltage pins. The large part of the board where the division starts involves the microcontroller chip where all the programming code can be deployed. On the other side of the division there is an extra chip which contains the programmer for the chip that can as well perform debugging in code, in order to troubleshoot problems and defects in the code.

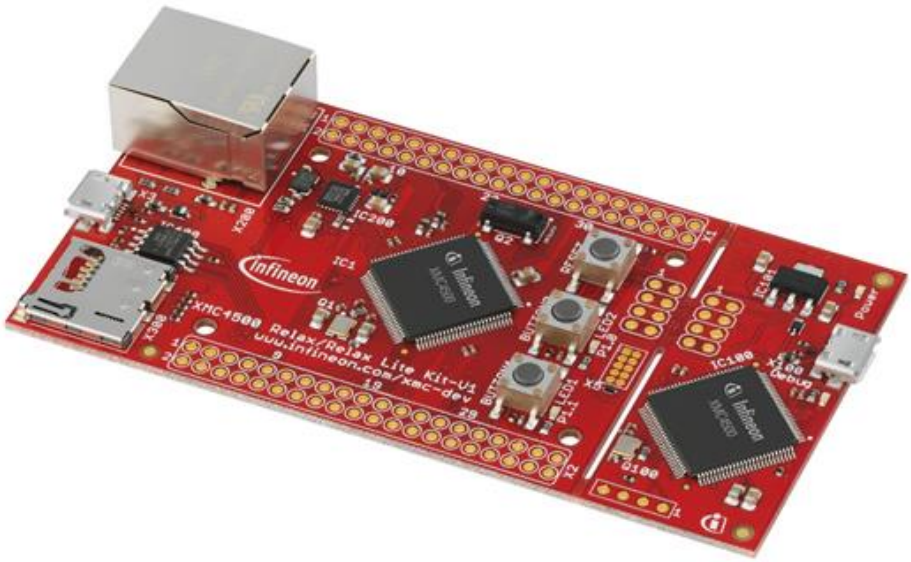


Figure 47: (Permission Pending)

- FPU with Arm Cortex -M4 backed by single cycle DSP and 144Mhz

- Analog
- 4 ADC (analog to digital converters) of 12 bit 8 channels
- 1 DAC (digital to analog converter) of 12 bit 2 channels
- 1 demodulator of 4 channels

These microcontrollers are based off of the Arm Cortex - M4 and a peripheral tool set which is constructed to fit the needs of connection and controllability solution systems.

There are several position interfaces that control the motor positioning and an external screen for watchdog timer interruptions for safety critical applications, six multi master serial bus standard for connecting ECU nodes in a network for communication and RTC modules with alarm trigger compatibility.

This system is specifically made to compare and capture feedback from power conversion and motor control functions.

This core is supported by a digital signal processor which multiplies and accumulates dual operations and it is supported by a floating-point unit as well.

This XMC47xxx unit is compatible with all series including XMC4800 while it adds ethernet field bus systems for eliminating needs for separate implementations on the design.

Specs and Ratings:

- Temperature rating of 39 to 119 degrees Celsius
- High performance interrupt handling
- Memory
- 352 kB SRAM
- 2 MB Flash memory

Alternate Microcontroller platform / MCU

Particle Photon

Specs and Ratings:

IC…….....BCM43362 chip from Broadcom

Pinout……....5 Analog and 8 Digital

Flash Memory…… 1MB

RAM…….....128 KB

GPIO……..18 pins

SRAM……...…128KB

Wi-Fi……..800.22b/g/n

Particle photon is a hardware device with cloud capability The particle photo allows for cloud computing applications and IOT functions. The variables function and algorithms allow for it to be externally accessible through another device connected in the cloud. The particle photon. It is an IOT device where projects can be created, and software updates can be updated without having to remove the hardware from the design for deploying roles like a regular microcontroller. The Flash can be processed through the cloud IDE platform.

Several other shields could be adapted into the Photon Device such as adapter shields. It is considered a good pick for our project because of the compact capability and the interconnection between different devices with an on-cloud access. The processor is an STM32F205RGY6 high performance ARM Cortex family. It is a 32 bit RISC processor which operates with up to 120 MHz through high speed embedded memories. Aside from the electrical power supply through the USB port it is 100% wireless

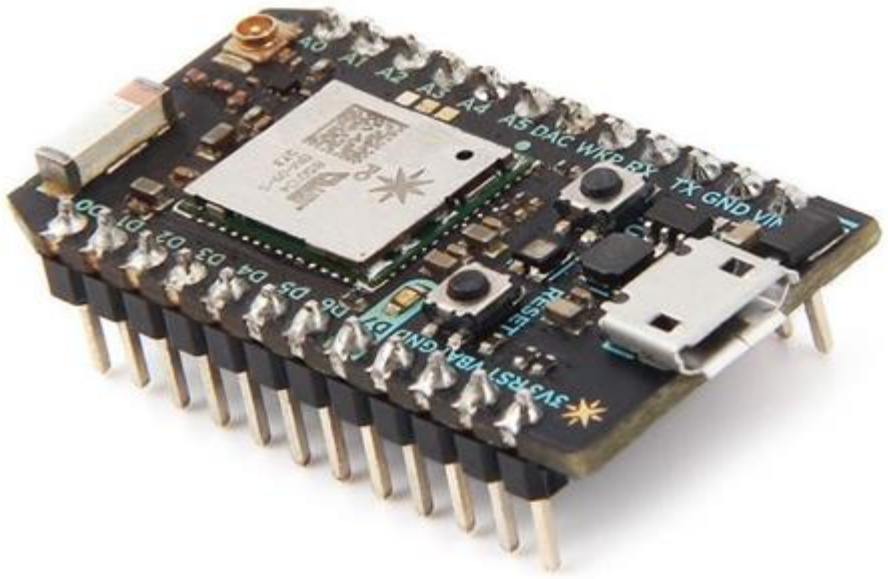


Figure 48: (Permission Pending)

Wi-Fi Setup

Logging into the Board Manufacturer “Particle Application”

The Wi-Fi name needs to be searched for set-up. It will be called “HOME”. The Wi-Fi connection needs to be configured for the photon following the commands, and the new photon controller could be found under the “Particle Application”.

When the Particle Photon is connected to a power source then an LED on the board would be blinking green, meaning it is searching for Wi-Fi signal. This module contains a real time operating system with certified license for CE, FCC, and IC. There is a Soft AP setup and it contains mixed signal GPIO additional peripherals as well as status LED’s right on the platform.

5. Related IEEE Standards and Regulations

Standards are an established requirement for specifications related to a certain field. Engineering standards have been put into place to require all types of design meet a certain criterion. This ensures that all engineering practices follow a practice of safe engineering design and constituent protocols across all fields of engineering. There are quite a few standards that apply to our design of an ECG. The Institute of Electrical and Electronics Engineers, or IEEE, provides professional documents of standards that need to be followed to ensure that designs are safe and reliable. Since our design involves medical measurements on top of typical electrical engineering design procedures, there are quite a number of standards that we must follow to ensure our safety and to follow the professional guidelines.

5.1 Professional standards

The understanding of standard systems in conformity assessment is very important in establishment of regulations for producing medical equipment. Standards are documented agreements about technical specifications to be used as rules, guidelines to ensure that services, products, and processes are met with their purposes. Standards can be enforced as mandatory, or they can be voluntary. In fact, most standards are voluntary. When a standard is enforced as mandatory, it is called a regulation. However, this mandate may or may not have a global basis. There are several important purposes that standards serve:

- They give reference criteria for product, service, and process to be met.
- They provide information for better safety, reliability, and performance for products, processes, and services.
- They ensure that consumers will receive products, services with reliability and performance.

As the world has become more connected, the need for standardization also has become more and more important internationally. Common references to the kind of process, sevice, and product are provided by quality systems and other management standards. Professional standards that are usually given by ISO (International Organization for Standardization) are the guidelines that are applied on almost every service, product, and process. There are four types of specifications in standards:

1. Prescriptive specifications are guidelines for characteristics such as products ementions, materials, calibration procedures as well as definitions of terminologies.

2. Design specifications are guidelines for technical characteristics of a product that guarantees compatibility when it is produced by many different manufacturers.

3. Performance specifications are guidelines for testing to ensure that products will meet their prescribed specification, services will meet their expectation, and processes will pass all requirements.

4. Management specifications are guidelines for processes and procedures for companies to put in place so that an organization has its management systems that can be learned if replacement is needed.

A standard may contain more than one type of specifications. The first three types: prescriptive, design, and performance specifications are usually common combinations in standards. In recent years, management specifications are gaining their provenance rapidly since organized structures of management systems proved the effectiveness to teer many big companies to their successes. For our project, we need to follow the regulations for medical devices that are standards that are given by different organizations such as ISO, IEC, ITU, ANSI, ect. Medical devices are almost used globally. The performance, safety, and consistent quality of medical devices are common interests of international public health communities. Therefore, medical device standards and regulations are critical as global harmonization.

5.2 Medical Device Regulations

Regulatory authorities can recognize a standard fully or partially, provided that they specify and publicize their intent. z. In many countries, medical device standards are mandated by the publication of government recognized standards. They are called medical product compliance. Medical products that are produced with the intent for international use should follow international standards. International standards are building blocks for harmonized regulatory processes to ensure the safety, quality, and performance of medical devices. There are several principles that the WHO medical device regulations handbook suggest to ensure the purposes of harmonized regulatory of medical devices:

- “The Essential Principles of Safety and Performance of Medical Devices” (Essential Principles, GHTF document SG1 NO20R5). Regulatory Authorities and

industry should encourage and support the development of international standards for medical devices that complies with the Essential Principles.

- International standards should be encouraged to be used by Regulatory Authorities while developing new medical device regulations.

- Regulatory Authorities should provide a mechanism for recognizing international standards to provide manufacturers with a method to demonstrate their compliance with the Essential Principles.

- It is acceptable when an international standard is not applied or not applied in full, if an appropriate level of compliance of the Essential Principles can be demonstrated.

- It may be appropriate for Regulatory Authorities to accept the use of regional standards or industry standards as a means of demonstrating compliance while it may be preferable for harmonization purposes to use international standards.

- While not discouraging the use of new technologies, the use of standards should reflect current, broadly applicable technologies.

- Standards should reflect the current state of the art in the technological field. However, not all medical devices may be addressed by recognised standards, especially new developed devices and emerging technologies.

It can be very expensive in terms of time and resources to implement a full regulatory system. The trend to use international standards is, in effect, tackling this problem by steering more and more manufacturers to produce medical products with uniform standards. The methods and procedures relating to governmental regulations are also converging. These developments create opportunities for many countries to develop low cost systems based on the work that have already been done in this field by other countries. A good approach for expanding harmonized recommendations is to establish a comprehensive national policy or guidelines on medical device management.

Besides regulations and standards that are applied for medical devices, there are many requirements and standards that need to be applied on components that are used in those devices. With the rapid advancement of technologies, electronic medical devices that can be used with less power consumption, so the market for rechargeable batteries is also emerse as a new industry that needs to be regulated and managed with standards and regulations.

ECG Regulations

1. A representation of the input signal will grant deviation to output recorded,which may not exceed 24 gV or 4.9%, whichever is greater.

2. The root means Square error over the PQRST complex should not exceed 10uV
3. The error in Vpp ions may not exceed 10 gV or 2.1 percent or larger value.
4. Root mean squared error divided by mean squared amplitude of a deflection should not exceed 2%
5. QRS deflections with .19 uV amplitude and greater than 6 ms duration should be detected
6.The MPE shouldnt surpass 10% during a number of any QRS deflection . 19 uV and . 11 ms.

5.3 Power Supply Standards

The table below highlights some of the major standards in regards to proper power supply standards. Misuse of the power supply of a design can not only damage key components, but also may cause a safety hazard in a worst case scenario.

Standard	**Description**
IEC 60950-1	Information of tech equipment
IEC 60601-1	Medical electrical equipment

UL 1310	Safety standards for power units
IEC-61010-1	Safety guidelines for lab equipment

Table 7

5.4 IEEE standard for rechargeable batteries

IEEE Standard for Rechargeable Batteries is a standard for the use and manufacturing of lithium-ion rechargeable battery packs. The standard defines approaches to designing and testing battery cells or battery packs to prevent system failure. The reliability of the system is dependent on these guidelines and failure to do so may result in damage to the battery system. Manufactures rely on these standards to ensure that their product will be safe to use for their customers. For our design, the specific standard number IEEE 1625 provides specific standards for rechargeable batteries for portable computing. The standard focuses on a few topics that we are able to control and few that are meant for the production side of these batteries. Quality control, manufacturing process, and packaging methods are some of the topics that are meant for the manufactures to ensure proper production and handling. Battery use is growing exponentially in the world and because these standards need to be constantly reviewed and updated to ensure proper use. The U.S consumer product safety commission or CPSC states that:

- Components and battery-powered products comply with applicable voluntary standards
- New components and products that are not yet subject to voluntary standards be designed considering the best practices from similar voluntary standards;
- Battery-powered products be designed with a system approach addressing thermal protection, charge and discharge protection , and use in product, including:
    - Cells suitable for intended loads and conditions and manufactured with good quality control
    - Battery packs with proper Battery Management Systems, including charge control, short-circuit protection and cell balancing

- Chargers that comply with applicable voluntary standards and are suitable for product
- End-product systems (including cells, batteries, chargers, and product) are tested together for safe function and appropriate conditions.

For us, the standards that focus on the testing methods and energy capacity are areas that we can monitor in our own design to avoid hazards and protect ourselves from harm.

These standards can only provide the necessary documents and information for the manufactures and customers. The IEEE standards do not guarantee full protection for whomever uses the batteries. It can only provide the recommended guidelines to ensure reliable and consistent manufacturing processes and test procedures. The responsibility falls upon us and the manufacturers to abide by these standards to ensure the safest course of action when designing or testing. If the end user follows all the standards and uses proper battery usage, then damage and hazards can be avoided.

The best course of action to prevent hazards is to follow the batteries rated max voltage and max current values. These are values that are provided by the manufacturer and have been tested thoroughly. We do not have the time to go through the extensive testing procedures that the battery manufactures to do their products. This also applies to all areas of our design. We will follow the voltage and current limits for our components to not cause unwanted damage to any component and to avoid safety hazards at all cost. The minimum discharge voltage will be determined by the lowest voltage limit of the battery cell as specified by the manufacturer in order to prevent over-discharge.

5.5 PCB standards

Aside from IEEE, the Institute for Printed Circuits (IPC) provides standards specifically focused on printed circuit boards. One of the requirements for design is a printed circuit board must be implemented. All the electrical devices that you use have printed circuit boards implemented within them. It is unreliable to build any design with a breadboard due to size constraints and loose wiring. This is why our design must follow printed circuit board standards to ensure our design can work consistently. These standards cover everything from production to assembly. Listed below are a few of the IPC printed circuit board standards we should adhere to:

- IPC-2221A indicates the necessary general standard used within Printed Circuit Board Design.

- IPC-6011 indicates the necessary general performance specification for Printed Circuit Boards.

● IPC 2615 establishes the standard for the Printed Circuit Board Dimensions and tolerances.

● IPC J-STD-004B indicates the necessary requirements for soldering fluxes

When discussing PCB standards, soldering standards are also included because PCB's require components to be soldered to them. The next section discusses soldering standards that must be followed.

5.6 Soldering standards

Adequate soldering methods are essential to our project. An incorrect soldered component can cause our whole design to not function properly. Special care must be taken during this phase of the design process to ensure no problems come about. The National Aeronautics and Space Administration (NASA) provides free guidelines and standards for proper soldering techniques. The document we researched provided many tables and figures for all aspects of soldering electrical components.We will use the document to get a better understanding of soldering since most of us have not had much experience in soldering onto a printed circuit board. Courtesy of learn.sparkfun.com, the figure below accurately illustrates good and bad soldering techniques and we will be referencing it when the time comes to construct our PCB.

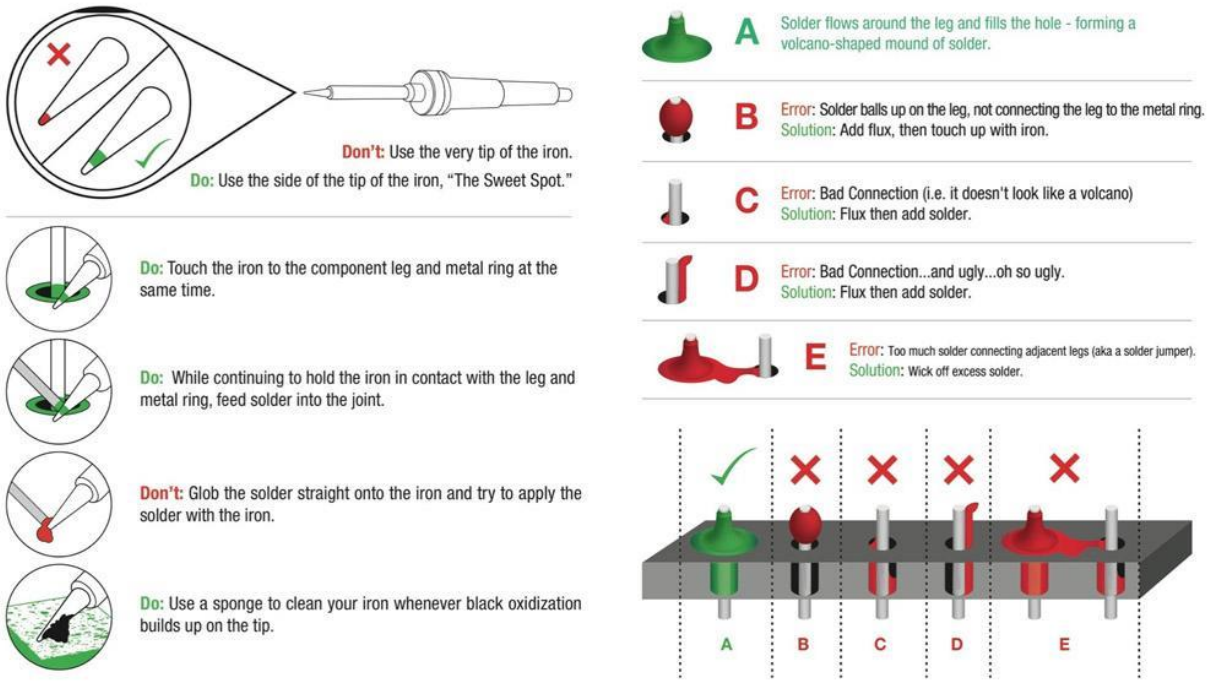


Figure 49: courtesy (Permission pending)

(https://learn.sparkfun.com/tutorials/how-to-solder-through-hole-soldering/all)

5.7 Personal Health Device standards

As technology progresses, health devices are becoming portable and for use at home. The aging population and an increase in diseases are factors that are requiring these types of devices to be easy to use and do not require a visit to a health professional. Personal health devices are growing exponentially and with us comes the need for updated standards that include personal health devices. The IEEE 11073-20601 standard covers personal health devices and our ECG design falls under this category. The standard states:

"...addresses a need for an openly defined, independent standard for converting the information profile [of personal health devices] into an interoperable transmission format so the information can be exchanged to and from personal telehealth devices and compute engines (e.g., cell phones, personal computers, personal health appliances, and set top boxes)."

Essentially the standard requires that personal health devices transmit health information or measurements in a clear and legible manner. The whole point of a personal health device is so that the user can receive adequate and accurate information at a given moment without having to wait to meet with a health professional. Some key points to the standard include:

- A point-to-point connection between device and user
- Use Bluetooth, USB, or Zigbee
- The device is self-describing (i.e legible and easy to comprehend for the user)

5.8 Design Constraints

Any design idea will come across various constraints that must be kept in mind throughout the entire design process. These constraints help us think of impending problems that may arise throughout this process. Other factors such as, Economic, Time, health, and safety constraints are also factors that may or may not have a negative impact on our final design.

One of the main problems that we were able to identify early on during our design process was the need to acquire a clean heartbeat waveform. This is the basis of our project and must be dealt with early on in the design process. To acquire the signal, we must be able to amplify the low voltage signal and filter that signal from unwanted

interference. Another problem we faced early on was that our project needed to be approved by the research department. This was something we did not think of beforehand and could've stalled our design process or even canceled it all together. Since our design idea would require a human to test, we needed the stamp of approval to make sure no one in the UCF engineering department would be held accountable for any safety hazards that may occur. Luckily, we were able to get approval quickly since our design project was for Senior Design and not for actual research purposes.

Since we started early on in our design phase that our project is not meant to compete with real life portable ECG products, we can avoid ethical constraints such as the promise of accurate heartbeat signals. If we stated that this was a product that would compete on the market, then we must be beyond all reasonable doubt that our ECG would provide accurate medical grade heartbeat signals. This design is strictly for Senior Design and to showcase the skills we have learned in our engineering courses. Our ECG design is not meant to give the user accurate readings that they could've gotten in a hospital. Such a feat is beyond the scope of our design process and would require more time and effort that we do not have during our Senior Design classes. That being said, our goal is to provide the user with a clear and easy to comprehend heartbeat signal. As stated earlier, our design is meant to show our teachers the electrical and computer engineering skills we have learned throughout our time in the engineering curriculum at UCF. We can avoid ethical constraints that could make us liable if we promised that our design is a medical grade ECG.

Economic and Time Constraints

Our design is a self-funded project which gives us a lot of freedom in the choices we can make for our design. That being said, the economic constraints will limit us from spending a lot of money to build the ECG. There are products out there that can do a lot of the tasks we need to for us, however, these products would increase our final cost bill. Also purchasing and using a product that would give us a shortcut in the design process would make our Senior design project illegitimate. One of our goals is to showcase what we have learned in school and purchasing a chip that would do a lot of the amplifying and filtering for us would take away from that goal.

The largest constraint of our design process is time. This is something that every Senior Design group faces as a factor that may have a negative impact on our design. In our case, we will be completing Senior Design 2 in summer which is only eleven weeks long unlike the fall or spring semester that is sixteen weeks long. Five week difference is a significant amount of time and if we don't manage our time properly then it may have a very huge impact on the final design outcome. Each group member is

either working while attending school, balancing other course work, or both. These are factors that need to be kept in mind throughout the entire design phase. Due to the current circumstances happening across the globe, we are not able to use the lab to test our design. This may stall our design process down the line during Senior Design 2 and we will need to make up lost time in order to succeed in our goals for a final ECG design. Regardless of the current circumstances, it is important for us to continue to practice adequate time management skills and meet our deadlines appropriately.

Health and Safety Constraints

Our design project will be responsible for taking health measurements of the user. To do this, a person must be connected right to our design. Even though we will be the only people testing the project, our safety is very important to us. As we mentioned earlier, permission was needed to proceed with our project due to this potential safety hazard. All our electrical components need to be handled and implemented correctly while following the IEEE guidelines. We also need to follow the manufacturer's rated maximum and minimum voltage/ current values to ensure we do not damage any component. Since our project will be connected directly to the person to test and ultimately function, electrical shock is our main concern. Figure # shows what will happen if different current levels were to travel back into the whomever is connected to the ECG.

Current	Reaction
1 milliamp	Just a faint tingle.
5 milliamps	Slight shock felt. Disturbing, but not painful. Most people can "let go." However, strong involuntary movements can cause injuries.
6-30 milliamps	Painful shock. Muscular control is lost. This is the range where "freezing currents" start. It may not be possible to "let go."
50-150 milliamps	Extreme pain, respiratory arrest, severe muscular contractions. Individual cannot let go. Death is possible.
1,000-4,300 milliamps (1-4.3 amps)	Ventricular fibrillation (uneven uncoordinated pumping of the heart.) Muscular contraction and nerve damage begins to accur. Death is likely.
10,000 milliamps (10 amps)	Cardiac arrest and severe burns occur. Death is probable

Figure 50:

Courtesy (https://www.scienceabc.com/humans/how-many-volts-amps-kill-you-human.html) (Permission Pending)

Skin irritation may also be a factor we should consider. We haven't made a decision on what the leads to the ECG will look like or be made of. The leads need to be reliable so that a constituent signal can be acquired. Any slight discrepancies in the lead or the lead connection to the user will cause irregular signals. A poorly made ECG lead could cause skin irritation and is something we will keep in mind when designing the leads.

6. Project Design

6.1 Hardware Design

ECG Block Diagram

The basic structure of an ECG system consists of protected electrodes connected into an instrumentation amplifier that is passed into an ADC to be digitized and prepared for signal processing via the onboard CPU. The output of the instrumentation amplifier is also fed back into the RLD amplifier to bias the body and keep the electrode voltage in range of the ADC. This block diagram is scalable to create ECG systems that have 3, 6, and even 12 leads.

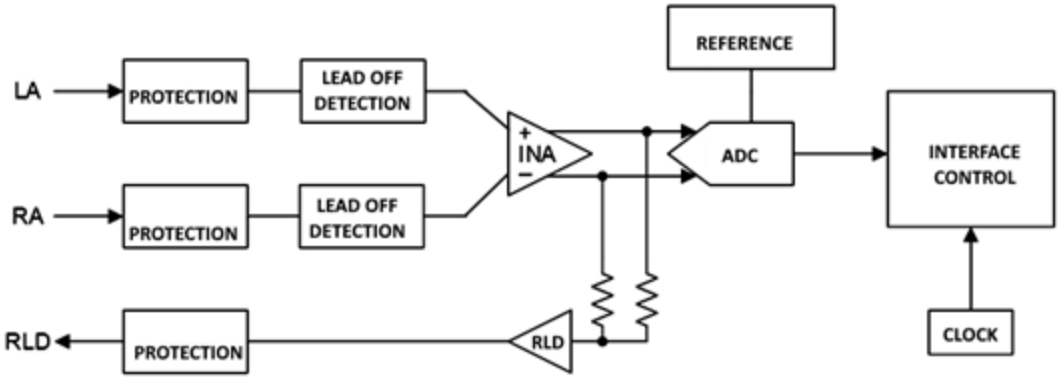


Figure 51:3 Lead ECG block diagram

ECG Front End

The front end of an ECG must be able to handle weak signals ranging from 0.5 mV to 5.0 mV, combined with a dc component of up to ±300 mV(from the electrode-skin contact) and a common-mode component of up to 1.5 V, resulting from the potential between the electrodes and ground. For a monitoring application in intensive care units, the bandwidth of an ECG signal can range from 0.5 Hz to 50 Hz. The bandwidth can go upwards of 1 kHz for late-potential measurements (pacemaker detection).

A standard clinical ECG has a bandwidth of 0.05 Hz to 100 Hz. Our circuit design must account for several different sources of noise: Power line interference, electrode contact noise, motion artifacts, muscle contractions, respiration, electromagnetic interference, and high frequency noise coupled from other electronic devices. The ECG front end analog filtering stages are shown in figure 53, and the complete system design is shown in figure below and requires several components that will be described in figure 52.

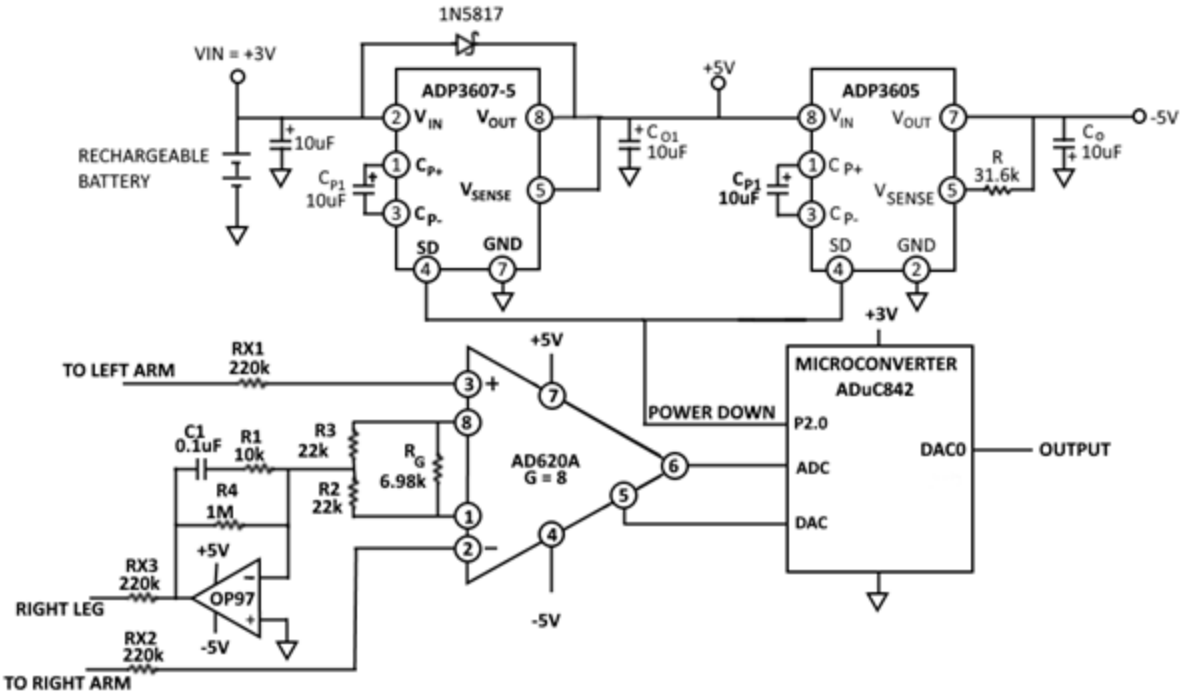


Figure 52: Frontend System schematic

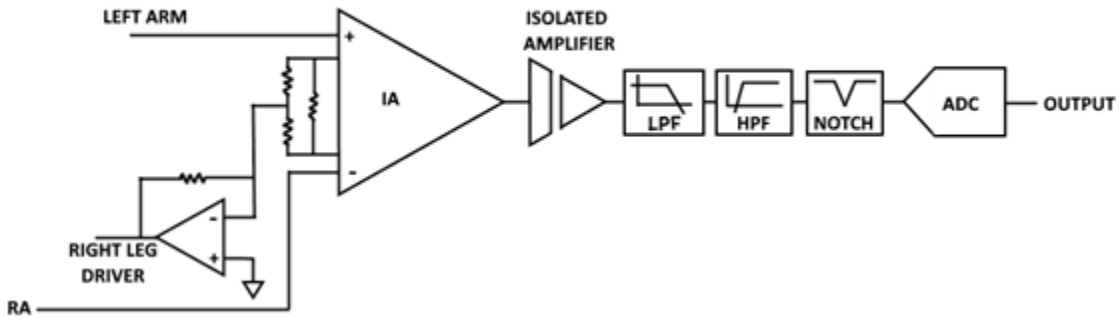


Figure 53: Front end signal chain

Instrumentation Amplifier

The AD621 is a low cost, high accuracy instrumentation amplifier (IA/INA), with excellent dc performance. It only requires one external resistor to set gains of 1 to 10,000. It features a minimum common mode rejection ratio of 100dB up to 1kHz, 50-µV max offset voltage, low input bias current (1 nA max), and low input voltage noise (0.28 µV from 0.1 Hz to 10 Hz). The equation for the gain based on R2 & R3 from figure 13 is:

$$Gain = 1 + \frac{49.4k\Omega}{R_G} + \frac{\frac{49.4k\Omega}{2}}{22k\Omega}$$

The gain is limited by the output swing and the maximum input voltage to the instrumentation amplifier. With a ±5V power supply, the output swing of the AD620 is about ±3.8V; and the maximum input is ±5 mV plus a variable dc offset of up to ±300 mV, allowing a maximum gain of 12.45. We choose to set the gain to 8, using RG = 8.45 kΩ.

ADuC842 MicroConverter

We will use an ADuC842 MicroConverter in our design which simplifies our circuit by combining the ADC, filters, and microprocessor into a single integrated circuit. This gives us better flexibility in implementing filters and isolating the signal digitally. This IC well suited for the main signal processing tasks. It features a fast, 12-bit ADC and other high-performance analog peripherals, a fast 8052 microprocessor core, integrated 62KB flash memory for code, and several other useful peripherals.

The key components of the MicroConverter for this design are the ADC and the 8052 core. The ADC converts the analog output of the instrumentation amplifier to a digital signal. The software written for the 8052 core processes the digitized signal to produce the data for the ultimate ECG trace. The software includes both high level code written in C and time sensitive routines written in assembly code. In this case, the implementation of band-pass filters and notch filters are written in C, while the ADC is controlled by assembly code. Assembly code, combined with converter speed, enables the accumulation of multiple samples, improving the effective resolution of the ADC well beyond its typical 12 bits. The Block diagram for the ADuC842 is shown in figure 54:

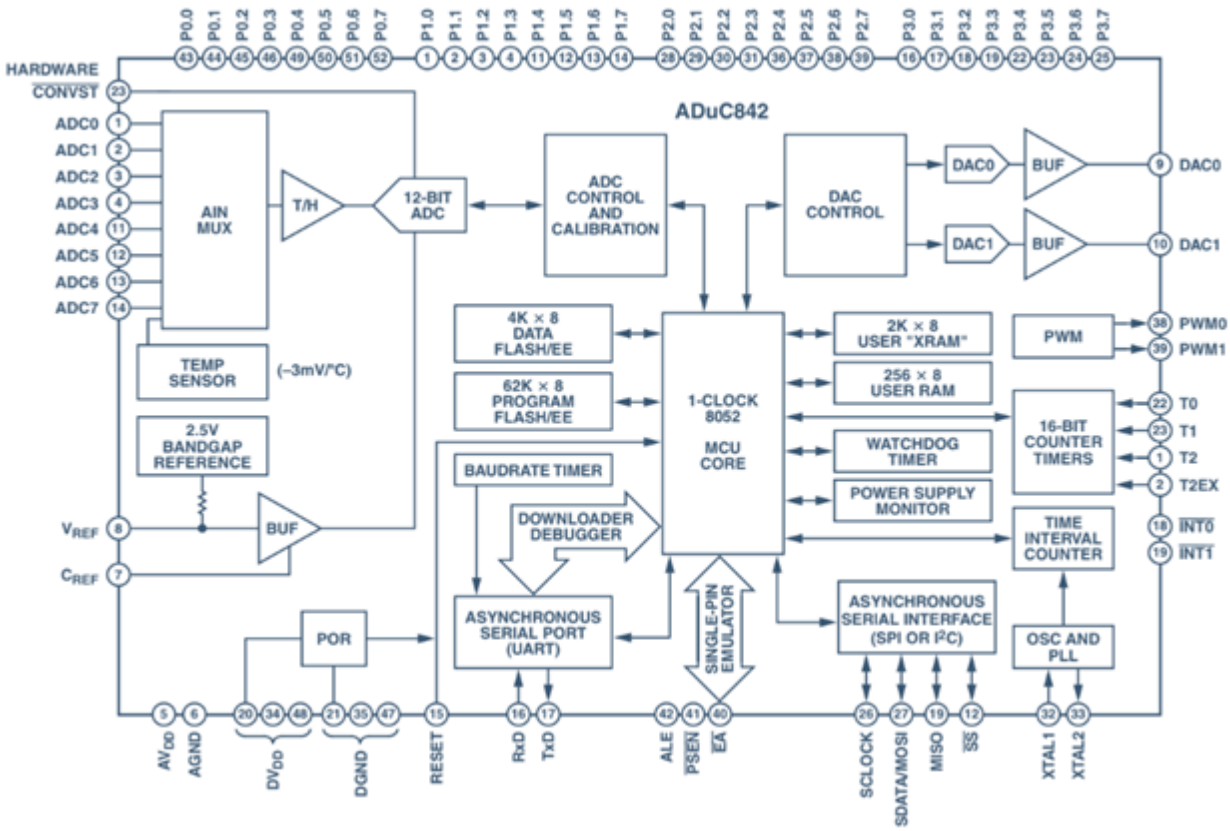


Figure 54: The MicroConverter (permission pending)

Right Leg Drive Circuit

The component used for the right leg drive is the OP97. It's a high precision, low power operational amplifier with a very high common mode rejection ratio (114dB minimum). This circuit applies an inverted common mode interference to the patient's right leg and cancels out the interference generated by the body. The op amp has a voltage gain for the common-mode voltage of 91 with a 1.6-Hz roll off and a low-pass cutoff at about 160 Hz for stability.

$$Gain = \frac{R4}{R2||R3} = \frac{1M\Omega}{11k\Omega}$$

$$3dB\ Frequency = \frac{1}{2\pi * (10k\Omega * 0.1\mu F)}$$

Power supply circuitry

The system's isolated power is supplied by a battery. A ±5V dual supply is needed for the AD629 and the OP97 to handle a bipolar input signal. We can use the ADP3607-5 booster regulator and the ADP3605 inverter to serve as a regulated dual supply that provides positive and negative regulated voltages from a single 3V battery. The ADP3607 is a regulated-output switched-capacitor voltage doubler capable of providing up to 50 mA.

Capable of operating from an input voltage as low as 3 V, it is offered in a version with the regulation fixed at 5 V (ADP3607-5). The ADP3605 switched-capacitor voltage inverter, with a regulated output voltage, is capable of providing up to 120 mA. It is offered with the regulation fixed at –3 V (ADP3605-3) or adjustable via external resistors over a –3-V to –6-V range. A –5V supply is needed, with an input voltage of +5 V, so we set R = 31.6 kΩ, and use the equation:

$$Vout = -1.5\frac{R}{9.5k\Omega}$$

Both supply voltages (±5 V) are generated by capacitive charge pumps, which cannot generate unsafe voltages—even under fault conditions—because they do not require any inductors. These devices also feature a shutdown mode, which allows the MicroConverter to power down the devices when the system is not in use.

Patient Isolation

In order to comply with AAMI (Association for the Advancement of Medical Instrumentation) standards for safe current levels, the series resistors, Rx1, Rx2, and Rx3, provide protection for the patient. These standards require that rms ground currents or fault current from the electronics must be less than 50 μA.

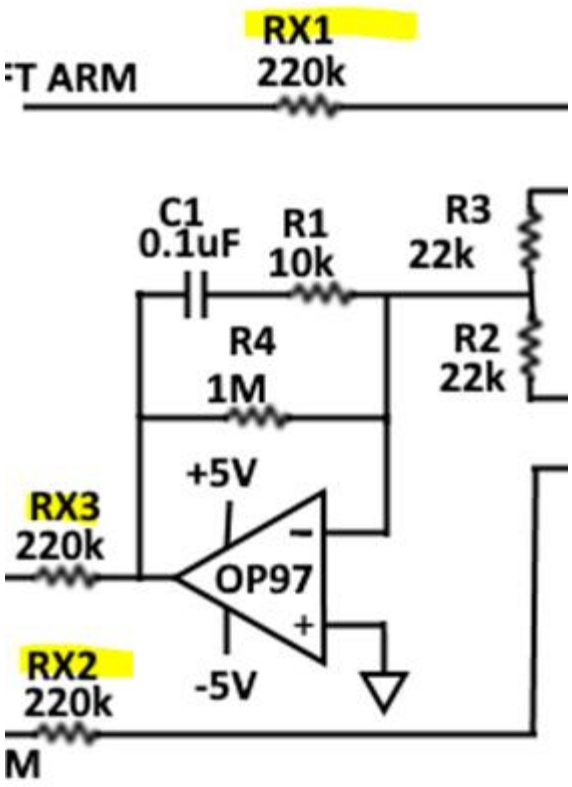


Figure 55: Safety Resistors

Digital signal Isolation

After the signal has been filtered, conditioned, and digitized by the ADC, it passes through a digital isolator so it can then be processed by the host (i.e. a computer or MCU) to be displayed on screen. The purpose of a digital isolator is to reduce any possible ground loop noise and conform to safety regulations. The ADuM1301 is a bidirectional isolator that achieves high data rates with lower power consumption than optocouplers. The power supply for the measurement side of the ADuM1301 is taken from the ADP3607-5, which provides a fixed 5 V output.

Digital Processing and Filtering in C

The acquired signal is processed by digital filtering in the MicroConverter. For this purpose, we utilized two second-order digital infinite impulse-response (IIR) filters, based on a sampling frequency of 500 Hz. A notch filter was designed to suppress the 50-Hz interference. The chosen design procedure was the pole-zero placement method, with a notch frequency of 50 Hz and notch width 10 Hz. To achieve this required the following transfer function:

$$H(z) = \frac{1 - 1.618z^{-1} + z^{-2}}{1 - 1.5164z^{-1} + 0.8783z^{-2}}$$

This transfer function may be expressed as a recursive algorithm:

$$NOut_k = NIn_k - 1.618NIn_{k-1} + NIn_{k-2} + 1.5164NOut_{k-1} - 0.8783NOut_{k-2}$$

In this equation k, means the present value, k-1 means the value in the previous instant, and so on. C coding was the choice for this arithmetic-intensive processing, as programming it in assembly would have been too time consuming. Implementing the filter equations directly would be inefficient with the ADuC842, since it is not specifically designed for floating-point calculations. To implement this second order filter in C code we scale the coefficients by 4096 and write:

```
iNOut = (4096L*iNIn-6627L*iNIn1+4096L*iNIn2+6211L*iNOut1-3598L*iNOut2)/4096;
```

The second filter was a Butterworth pass-band filter with a 0.05-Hz low cutoff frequency and a 100Hz high cutoff frequency. The transfer function and recursive algorithm are:

$$H(z) = \frac{0.4206 - 0.4206z^{-2}}{1 - 1.5182z^{-1} + 0.1582z^{-2}}$$

$$BOut_k = 0.4206BIn_k - 0.4206BIn_{k-2} + 1.1582BOut_{k-1} - 0.1582BOut_{k-2}$$

This algorithm implemented in C is given by:

```
iBOut = (1723L*iBIn-1723L*iBIn2+4745L*iBOut1-650L*iBOut2)/4096;
```

The outputs can be scaled simply by changing the coefficients of the inputs. Also note that, if the signals are all positive, the division by 4096 is accomplished at the end by shifting 12 right.

The implementation shown in the code below is for a cascade of five band-pass filters and two notch filters. The signal is scaled up by a factor of 4 in each of the first and second band-pass filters. The 12-bit right shift accomplishes the divide-by-4096.

```
while(1)
{
while(c25ms<64);     //Wait for 64 measurements to be done.
iBIn = iAdc0>>3;     //Save accumulated measurement.
iAdc0 = 0;           //Zero for new measurement accumulation.
c25ms = 0;           //Reset synchronization timer.
//5 Band pass 0.05 - 100Hz Fs=500 first 2 with gain of 4 each.
iBOut = (6891L*lBIn-6891L*lBIn2+4745L*iBOut1-650L*iBOut2)>>12L;
iBO10 = (6891L*iBOut-6891L*iBOut2+4745L*iBO11-650L*iBO12)>>12L;
iBO20 = (1723L*iBO10-1723L*iBO12+4745L*iBO21-650L*iBO22)>>12L;
iBO30 = (1723L*iBO20-1723L*iBO22+4745L*iBO31-650L*iBO32)>>12L;
iNIn = (1723L*iBO30-1723L*iBO32+4745L*iNIn1-650L*iNIn2)>>12L;//2Notch filters
iNOut = (4096L*iNIn-6627L*iNIn1+4096L*iNIn2+6211L*iNOut1-3598L*iNOut2)>>12L;
iN3O = (4096L*iNOut-6627L*iNOut1+4096L*iNOut2+6211L*iN3O1-3598L*iN3O2)>>12L;
iBIn2 = iBIn1;       //Save delayed values for filters.
iBIn1 = iBIn;
iBOut2 = iBOut1;
iBOut1 = iBOut;
iNIn2 = iNIn1;
iNIn1 = iNIn;
iNOut2 = iNOut1;
iNOut1 = iNOut;
iBO12 = iBO11;
iBO11 = iBO10;
iBO22 = iBO21;
iBO21 = iBO20;
iBO32 = iBO31;
iBO31 = iBO30;
iN3O2 = iN3O1;
iN3O1 = iN30;
if(iBIn>24000) iDac -= 1; //Control AD620 output level.
if(iBIn<8000) iDac += 1;
iOut1 = (iDac)&0xfff;
DAC0H = (iOut1>>8)&0xf;
DAC0L = iOut1&0xff;
if((iN3O+iOfs)>3000) iOfs -= 1; //Control output level.
if((iN3O+iOfs)<1000) iOfs += 1;
iOut = ((iN3O+iOfs))&0xfff;
DAC1H = (iOut>>8)&0xf; //Output to oscilloscope for evaluation.
DAC1L = iOut&0xff;
if(!(c2++&3)) printf("%4d\r\n",iOut); //Output to PC.
```

The lines: `if (iAdc00>24000)iDac-= 1`, and `if(iBIn<8000) iDac += 1`, adjust the DAC output of the ADuC842 to drive the level-shifting input of the AD620 to shift the AD620 output to an acceptable value for the MicroConverter's ADC input. This is desirable to reduce the effects of the variable dc offsets that result from slight differences in the way the electrodes are applied to the skin. The same idea is used to ensure that the output voltage is centered within the output range.

Further Processing using Assembly code

The assembly code's main functions are to measure the input signal at regular intervals and to ensure that the C code calculations are repeated at the required rate of 500 times per second. In the first instance, we programmed Timer0 to run continuously

and generate its interrupts at 1ms intervals. Each interrupt restarts Timer0, gets an ADC conversion result, and increments a variable, c2ms, which is used to synchronize the C code. At this stage of code development, the first few lines of C code were:

```
while(c2ms<2); //Used in first phase.
c2ms = 0;
iAdc00 = iAdc0;
```

Initially, c2ms = 0, and the C code will wait at the line: while(c2ms<2);. After 1ms, a Timer0 interrupt occurs, and c2ms is incremented to 1. After another 1ms, c2ms is incremented to 2. Now while(c2ms<2); is no longer satisfied, and the C code continues by resetting counter c2ms to 0 and doing the filter calculations. Thereafter, the C code shifts the results down the chain of variables representing the various delayed results ready for the next iteration of the loop. The final part of the loop is the printf(...), which sends the result to the PC for display.

To improve the result, the Timer0 interrupt rate was shortened to 1/32ms, and the data was accumulated in iAdc0, to make use of multiple measurements instead of just a single measurement. At the same time, the while was changed to while(c2ms<64) so that the C code would wait for 64 measurements to be accumulated before doing each filter loop. The value in iAdc0 is saved in iAdc00 for further processing, and then iAdc0 is cleared and ready to accumulate the next 64 measurements. Below is the assembly code:

```
IntT0:      push    ACC
            push    PSW
            clr     TR0         ;Stop T0.
            mov TH0,#0fdh       ;Reload 1/32ms.
            mov TL0,#0f6h
            setb TR0            ;Restart T0.
            mov a,ADCDATAL
            add a,iAdc0+3
            mov iAdc0+3,a
            mov a,ADCDATAH      ;Add a conversion to iAdc0.
            anl a,#0fh
            addc a,iAdc0+2
            mov iAdc0+2,a
            clr a
            addc a,iAdc0+1
            mov iAdc0+1,a
            mov ADCCON2,#0      ;Start a conversion.
            mov ADCCON2,#10h
            inc c25ms           ;Increment ms counter.
IntT0R:     pop PSW
            pop ACC
            reti
```

Gain settings

Signal gain is always an important consideration in an ECG signal chain. In the above-described design, it depends on several factors. The analog gain is set to 8x, as discussed previously. Next, a gain of 64× results from accumulating 64 measurements of this signal. Next there is a signal loss of 8× from the code `iBIn = iAdc0>>3;`, and finally, a gain of 4× twice from the scaling of the first two band-pass-filter equations. This results in a total gain of G = (8 × 64/8) × 4 × 4 = 1024, which is typical of analog ECG circuits.

7.0 Testing

The testing section shows our plans of how we will test various components of the hardware and software of our design. Our initial plan was to begin testing some parts of our design during the end of senior design 1, however, due to the national pandemic not allowing us to go on campus we are currently waiting to begin testing and building our design. Everything in this section describes the various tests we can and will do during this phase of the design process.

7.1 Hardware Testing

This section explains our plans for testing the hardware components of our design. The senior design lab will be utilized to perform these tests to ensure every component we have acquired functions properly. Testing is a crucial phase in our design process. If proper and adequate testing is not done, then problems later on in our design process may arise.

The first set of components we will test are the components that make up our analog front end. This includes all the op-amps, resistors, capacitors, and the instrumentation amplifier. Resistors and capacitors can be tested easily with a multimeter that can measure resistance and capacitance. Thoroughly checking these parts first is important because using a faulty or incorrect resistance/capacitance could damage other components or give us incorrect ECG readings. Each resistor and capacitor will be tested and checked to make sure they are measuring around the manufacturing rating.

Two of the most important components of our AFE are the instrumentation amplifier and op amps. These components are used for the amplification and filtering of the signal. Our project comes down to whether or not we can acquire a clean signal so adequate testing of these two components is crucial. We will use a multimeter and a

power supply to test these amplifiers. The MCP6004 datasheet says to use the test circuit shown below:

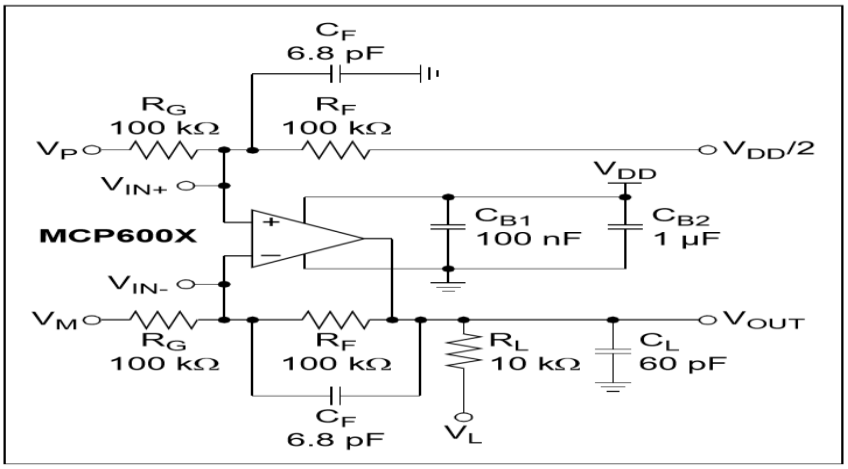


Figure 56:(permission pending) Courtesy of http://ww1.microchip.com/downloads/en/DeviceDoc/MCP6001-1R-1U-2-4-1-MHz-Low-Power-Op-Amp-DS20001733L.pdf

This test circuit could be used for a thorough test of the op amp, however, depending on time constraints we may decide to test with a simple non-inverting amp circuit to see if the component shows us the correct gain. The same method will be used for the instrumentation amplifier to check whether the component is faulty.

7.1.2 Power System testing

This section will highlight our plans for testing the components of our design involved with the power supply. These include the rechargeable battery pack, wall outlet adaptor, and voltage regulator.

To test the battery pack, we will charge our battery for the recommended amount of time and then use a multimeter to measure the direct current or voltage. This will immediately tell us if the battery pack is faulty or is an incorrect rated battery. The same process can be used for the wall outlet adaptor. Since we are buying this component and not building our own, then it'll be much easier to measure the voltage rating and see if it matches the manufacture rating.

Our final power system component to test is our voltage regulator. The regulator is in charge of accurately regulating the voltage levels from the power supply to 3.3V for the components of the analog front end. This is a very crucial test because incorrect voltage levels can lead to damaged parts. A simple test would be to connect our regulator to a breadboard and apply a voltage similar to what is to be expected for our design. We

will then use a multimeter to measure the input and output. As long as the output stays constant as the input is increased, then our regulator is functioning properly.

7.2 Overall Design Testing

After testing of the individual components of our design, we will be doing more tests as we build our ECG. One of the most important tasks of our design is to acquire a clean heart signal. To accomplish this task, we must test our AFE as a whole and see what kind of signal we receive after using our initial design. Changes to our design may need to happen if we need to either amplify the signal more or filter the signal more. Unfortunately, due to the given circumstances, we can not accomplish this until we can use the senior design lab.

We will be testing our AFE on ourselves since we are the ones building the design. Since we always need a person to be connected to the design to test our AFE, it is more convenient to have one of us be connected to it rather than having someone else be with us throughout the entire testing phase.

Some tests we will do include testing the durability of the electrode contact leads. Our goal is to see how much contact is needed for a good signal and at what point is the signal compromised. This is where our lead off detection will help us in this test phase. We can also test the accuracy of the BPM being acquired by the software. Beats per minute will be taken by one of us who is resting and one of us who has a higher BPM. These tests will help in both the hardware software aspects.

8.0 Administrative Content

This section of our document will be used to explain how our group managed our time to the best of our ability and to describe the budget of the design. This portion also portrays an abbreviated timeline of senior design 1 and our plans for senior design 2.

8.1 Project Diagram

The figure below shows a diagram of our design as a whole and how we distributed the workload amongst us. The different colors signify the specific group members workload. As we went through our design process, each group member looked over each other’s work to make sure we were all on schedule.

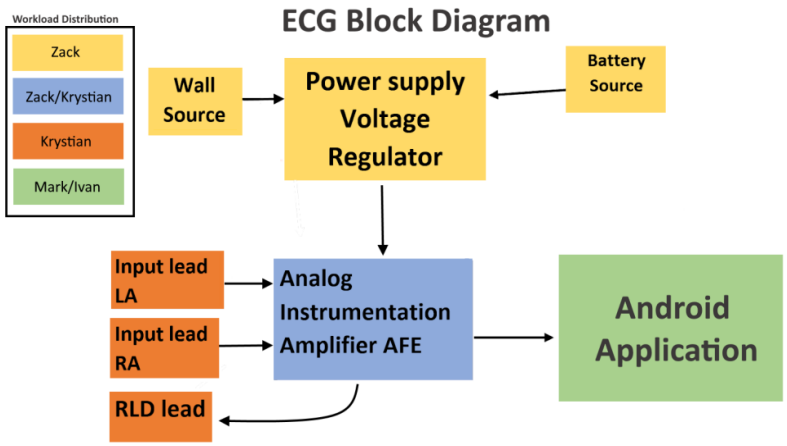


Figure 57:Workload breakdown

The division of work was divided into two groups: hardware and software. The hardware team consisted mostly of Kyrstian and Zack. This group was in charge of developing the power system and Analog front end. The power system included the power supply options and regulator. The AFE included the electrode contact lead along with the AFE itself. This group is also in charge of testing these hardware components and making sure nothing is faulty for the software team. Even though PCB would be considered a hardware task, all group members will be helping out with the final PCB layout and testing.

The second group is the software team. This group consisted of Mark and Ivan because they had the most experience in this area. The software group was in charge of the code for the application and the processing of information from the hardware. This was an area that we all did not have much experience in so all members were in charge of helping the software team during the research phase.

We thought that splitting the workload like this would be the most efficient route to succeed. That being said, we also agreed that each group would help the other group during the research phase. This allowed all us to have a full understanding of the entire design. The testing phase will also be a joint effort because we believe that if there is a problem, then we all can tackle the issue and resolve it quickly.

8.2 Budget

Bill of Materials(BOM)

Below is a compilation of all the parts necessary for the project. Due to the coronavirus quarantine, tools such as the oscilloscope, soldering station, and power supplies weren't accessible at the UCF's senior design lab. So we had to purchase our own.

Quantity	Unit	Description	Price($)
Analog Front End			
1	ADP3607-5	Voltage doubler	1.50
1	ADP3605	Voltage inverter	1.50
1	AD620A	Instrumentation Amplifier	5.24
1	OP97	Operational Amplifier	11.82
1	EVAL-ADUC841QSZ	Microconverter Dev Board	56.97
3	Data Source Lead Cable	ECG leads	44.00
3	Electrodes	ECG Electrodes	15.91
Power supplies & Tools			
1	SOIC to DIP Adapter	SOIC to DIP Adapter	2.95
2	3.7V LG3400G	Li-on Batteries	28.99
4	Breadboard	Breadboard	8.99
1	Breadboard Jumper Kit		12.00
1	Oscilloscope	ETEPON Digital Oscilloscope	41.99
1	X-Tronic's 3060-PRO	Soldering station	70.00

1	KAIWEETS DC Power Supply	Power Supply	99.99
Components			
1	Capacitor Assortment		14.99
1	1N5817	Schottky Diode	0.45
1	Resistor Assortment		14.99
1	AUX Cable 3.5mm		8.99
1	3.5mm Stereo Female block	Jack for AUX cable	7.59
Total			433.42

Table 7B

8.3 Milestones

The tables below show the basic and abbreviated milestones for Senior Design 1 and Senior Design 2. Our group tried our best to practice good time management skills and keep each other on track during the semester. Once we begin senior Design 2, we will be starting immediately to ensure we give ourselves the most amount of time possible during the shortened summer semester.

Senior Design 1

Week	Tasks
1	Project Idea
2-4	Divide and Conquer
5-6	Research/Updated D&C
7-10	60-page draft
11-12	100-page draft
13-15	Final draft

Table 8

Senior Design 2

Week	Tasks
1-2	Test
3-6	Build
7-8	Test whole design
9-11	Finalize and prepare
12	Present ECG

Table 9

9.0 Conclusions

During our Senior Design 1 journey, taking the first steps to designing and building a functional ECG has been challenging. We chose this idea because we all had interest in this field and thought it could be a very rewarding experience. We also thought it would be a good Senior Design project that could really highlight some of the skills we have learned while getting to learn or enhance other skills. Due to the circumstances going on in the world at the moment, we have not been able to go to campus and use the Senior Design lab to begin testing our design. The current situation will most likely roll over into the summer semester so we are waiting to see what happens and if we will be able to use the labs for our design.

Our ECG design project highlights various fields in the Electrical and Computer Engineering curriculum. To build a working prototype, the ECG must include proper power system usage, amplification/filtering techniques, and coding/application development. These were all items that we as a group have been introduced to and felt confident that we could accomplish our goal. The area that most had little to no experience in was the medical side of this project. The design required us to research not only how ECG's function, but also how the heart works. We needed to understand how the heart beats, the QRS topic, and how to actually measure a person's heart signal. All of these topics were very new to us and we spent most of our beginning research phase reading about this field.

Our main motivation for this project was to build something challenging that could truly highlight the skills we have learned. We had no plan of actually making an ECG that could compete on the market because of time and budget constraints. Even though the project was not intended for this, we still made design decisions that would be useful if someone were to actually use our design to see their heartbeat signal.

Future improvements

During our research phase, we came up with a lot of ideas and possible alternatives to our design. This allowed us to come together and choose a final idea that would be best for us. Just having some of the most basic features was already challenging so including more features may not have been possible due to the current circumstances and time constraints. One idea that we liked a lot in the beginning but decided to not go with it was incorporating a heart arrhythmia detection into the software. This would require software that could take the given heart signal and compare it to known heart arrhythmia signals. At the very least, the feature could notify the user that their heart signal may have some irregularities to it. However, we thought the basic features were already challenging enough and with the current timeframe we did not think it was possible.

Another improvement could be in the portability category. Some ECG's on the market are very portable and can fit securely in a person's pocket. At home personal use medical devices are becoming more popular and there is a growing demand for them. Most of the improvements could go into the aesthetic aspects of the device. To be a personal medical device, it must not only be appealing but also be easy to use without the help of a medical professional.

10. Appendix

Citations

Dah, O. Microcontroller Programming, Build Electronic Circuit. 2013. Web, https://www.build-electronic-circuits.com/microcontroller-programming/

Texas Instruments. Code compressor studio V7,x. User Guide, TI. 2017. Web, http://www.ti.com/lit/ug/slau157ap/slau157ap.pdf

Release notes, Platform IO core4. PlatformIO. 2020. Web. https://docs.platformio.org/en/latest/core/history.html

PlatformIO IDE, PlaformIO. 2014. Web https://docs.platformio.org/en/latest/integration/ide/pioide.html

Zues IDE Alternative to Atom, Zues An IDE designed by programmer's for programming. 2020. Web, https://www.zeusedit.com/index.html

Trust for America's health. Obesity. 2019. web, https://www.tfah.org/issue-details/obesity/

Subnamrian, B. ECG signal classification and parameter estimation using multiwavelet transform. Biomedical Research an International Journal of Medical Science, 2017. Web, https://www.biomedres.info/biomedical-research/ecg-signal-classification-and-parameter-estimation-using-multiwavelet-transform.html

Signal Theory. Signal and Information and Classification of Signal. 2012. Web, http://homepages.vub.ac.be/~imarkovs/st/notes-leo/Signal_theory_Chapter_1.pdf

Dah, O. Kicad vs Eagle Which one is best? 2018. Web. https://www.build-electronic-circuits.com/kicad-vs-eagle-2018-comparison/

ROHM. Difference between Linear Regulation and Switching Regulator? 2020. Web. https://www.rohm.com/electronics-basics/dc-dc-converters/linear-vs-switching-regulators

Renesas. Linear vs. Switching Regulators. 2020. Web, https://www.renesas.com/us/en/products/power-management/linear-vs-switching-regulators.html

Texas Instrument, Code Composer Studio 10.0.0. Projects and Build. Create and Managing Projects. 2019. Web, http://software-dl.ti.com/ccs/esd/documents/users_guide/ccs_project-management.html#building-projects

Basic Electrical. Different Types of Rechargeable Batteries. 2020. Web https://studyelectrical.com/2017/09/types-of-rechargeable-batteries.html

Electronics Tutorials. Full Wave Rectifier and Bridge Rectifier Theory. 2020. Web, https://www.electronics-tutorials.ws/diode/diode_6.html

BYU Computer Engineering and Electrical Engineering. UART. 2006. web

https://www.unm.edu/~zbaker/ece238/slides/UART.pdf

Jimblom. Sprakfun start something. Serial Communication. Web,

https://learn.sparkfun.com/tutorials/serial-communication/uarts

Philips Semiconductor. I2C manual. 2003. Web,

https://www.nxp.com/docs/en/application-note/AN10216.pdf

Texas Instruments. Keystone Architecture UART. 2010. Web,

http://www.ti.com/lit/ug/sprugp1/sprugp1.pdf

Infineon. Kit_XMC47_Relax_V1. 2020. Web,

https://www.infineon.com/cms/en/product/evaluation-boards/kit_xmc47_relax_v1/?redirId=114294

Micron. Infeneon. 2020. Web,

https://www.micron.com/solutions/micron-valued-partner-program/chipset-partner/infineon

Corelis. SPI Tutorial. Serial Peripheral Interface. 2020. Web,

https://www.corelis.com/education/tutorials/spi-tutorial/

Insight Medical Publishing. Biomedical Technology. 2020. Web,

https://www.imedpub.com/scholarly/biomedical-technology-journals-articles-ppts-list.php

Dr. Michael C. Medical Device Regulations Global Overview and Guiding Principles. 1999. Web,

https://www.who.int/medical_devices/publications/en/MD_Regulations.pdf

Sunny. How to choose the right AC Adapter for your gadget. 2019. Web,
https://blog.banggood.com/how-to-choose-the-right-ac-adapter-for-your-gadget-29870.html

United Consumer Product Safety Commission. Batteries. Web,
https://www.cpsc.gov/Regulations-Laws--Standards/Voluntary-Standards/Topics/Batteries/

James J.B., Alan S.B., Arthur J. Jr., Leo G.H., Peter W.M., David W.M., Christoph Z., Recommendations for standardization and specifications in automatic Electrocardiography: Bandwidth and Digital signal processing. 2020. Web,
https://www.ahajournals.org/doi/pdf/10.1161/01.CIR.81.2.730

Android Application Fundamental. Android developers. Web,
https://www.developer.android.com/guide/components/fundamentals

Data and File Storage Overview. Android developers. Web,
https://developer.android.com/training/data-storage

Device Compatibility Overview. Android developers. Web,
https://developer.android.com/guide/practices/compatibility

Background optimizations, Documentation. Web,
https://developer.android.com/topic/performance/background-optimization

Webster John G., Medical Instrumentation. Application and Design. 3rd edition, Wiley, 1998.

Firth J. and Errico P., “Low-Power, Low-Voltage IC Choices for ECG System Requirements,” Analog Dialogue, Volume 29, Number 3, 1995.

AAMI, American National Standard, Safe Current Limits for Electromedical Apparatus (ANSI/AAMI ES1-1993). Association for the Advancement of Medical Instrumentation, 1993.

AD620 Data Sheet revision F. Analog Devices, Inc., ©2003.

https://www.scienceabc.com/humans/how-many-volts-amps-kill-you-human.html