Julia Frothingham

Lab 3 - Data Acquisition via the ECG

The ECG data were acquired using a set of electrodes attached to the left and right ankles as well as the right wrist. The ECG signal is electrical activity from the heart measured between the left ankle and right wrist, with the right ankle used as a reference. It has a magnitude on the order of 1mV. The signal measured by the electrodes was displayed on an oscilloscope and put through a filter/amplifier. The oscilloscope was used to visually inspect the signal and make sure that it was as expected--for instance, it took a few seconds for the electrodes to begin picking up a signal. Monitoring the signal on the oscilloscope provided insight on when to begin collecting data.

The signal from the electrodes was also put through a conditioner, where it was amplified and several filters were applied. According to the lab handout, “The ECG does not include important frequency information above 100Hz,” so a low-pass filter was applied. If it was possible to use an ideal low-pass filter, it would have had a cutoff frequency of 100Hz. However, since actual low-pass filters have curved sides, a filter with a cutoff frequency of 300Hz was used instead. This will still filter out much higher unnecessary frequency components without unintentionally filtering out lower frequency components. A high-pass filter was also applied with a fairly low cut-off frequency of 0.1Hz. This was to filter out any DC components or noise with frequencies close to 0Hz. There was also a 60Hz notch filter used to filter out interference from the local power grid. On the equipment, this was labeled as a “LINE FILTER” switch that could be flipped to “IN” (filter on) or “OUT” (filter off). Data was collected with the notch filter both off and on.

The analog signal was amplified by a factor $5x10^3$ for better resolution after conversion to a digital signal. The ADC used in the system utilizes a 16-bit quantizer between -10V and +10V, so the original signal on the order of 10^{-3}V would only be stored in a few bits and would lose a lot of detail. However, too much amplification would put the signal out of the -10V to 10V range and cut off parts of the signal. The amplification factor was chosen by adjusting the amplification setting as high as we could without observing flattened peaks in the signal when plotted on the PC computer indicating overamplification.

After filtering and amplification, the signal was sampled and quantized by an ADC. According to the sampling theorem, the sampling rate must be at least twice the cutoff frequency of the filtered signal. So the sampling rate must be at least 600Hz. We used a sampling frequency of 8kHz. This increased the number of samples available to plot, resulting in smoother plots, without using unreasonable amounts of memory.

The digital signal was captured and stored by a Matlab program running on a PC computer connected to the ADC, utilizing the Data Acquisition Toolbox. This program adjusts the settings of the ADC (channel, sampling rate), controls when data is collected (when manually triggered) as well as how long data is collected (10 seconds) and how much data is collected (samples per trigger), and stores the collected data. It then plots the data vs time.

Figure 1. System diagram.

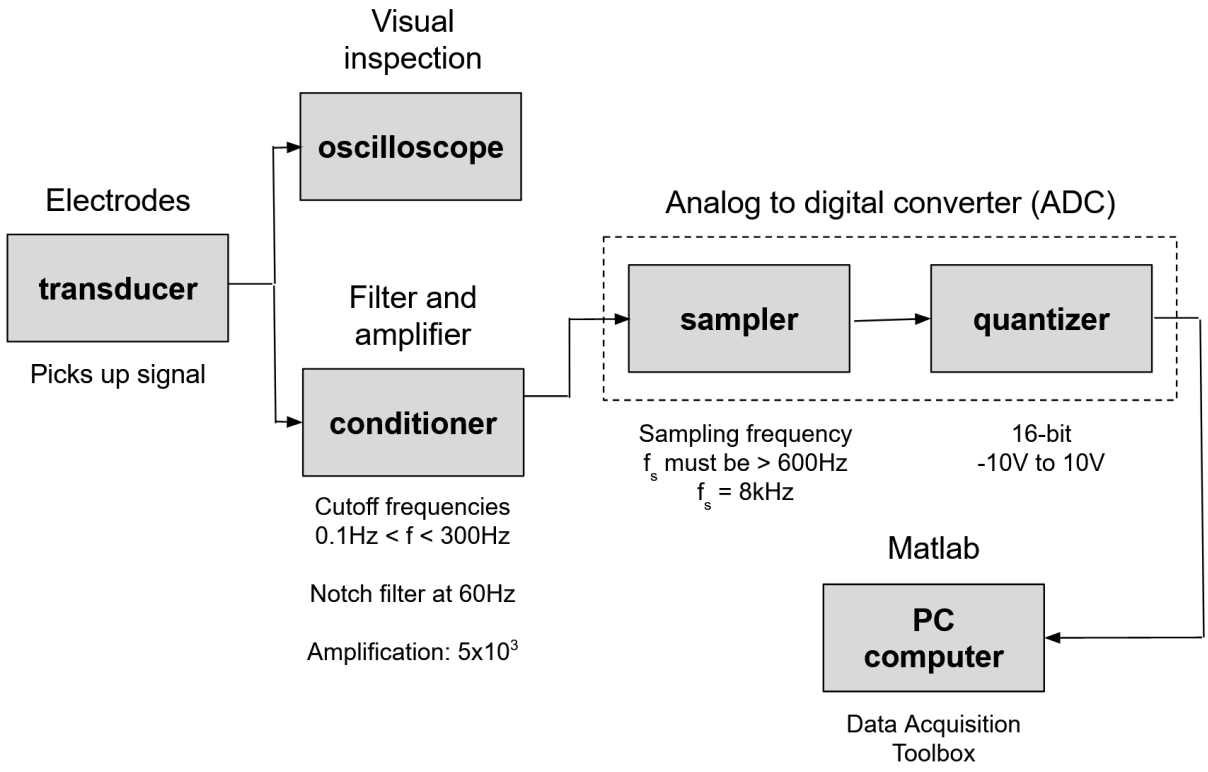


Figure 2. Amplifier and filter settings used.

Figure 3. Matlab code used to collect ECG data.

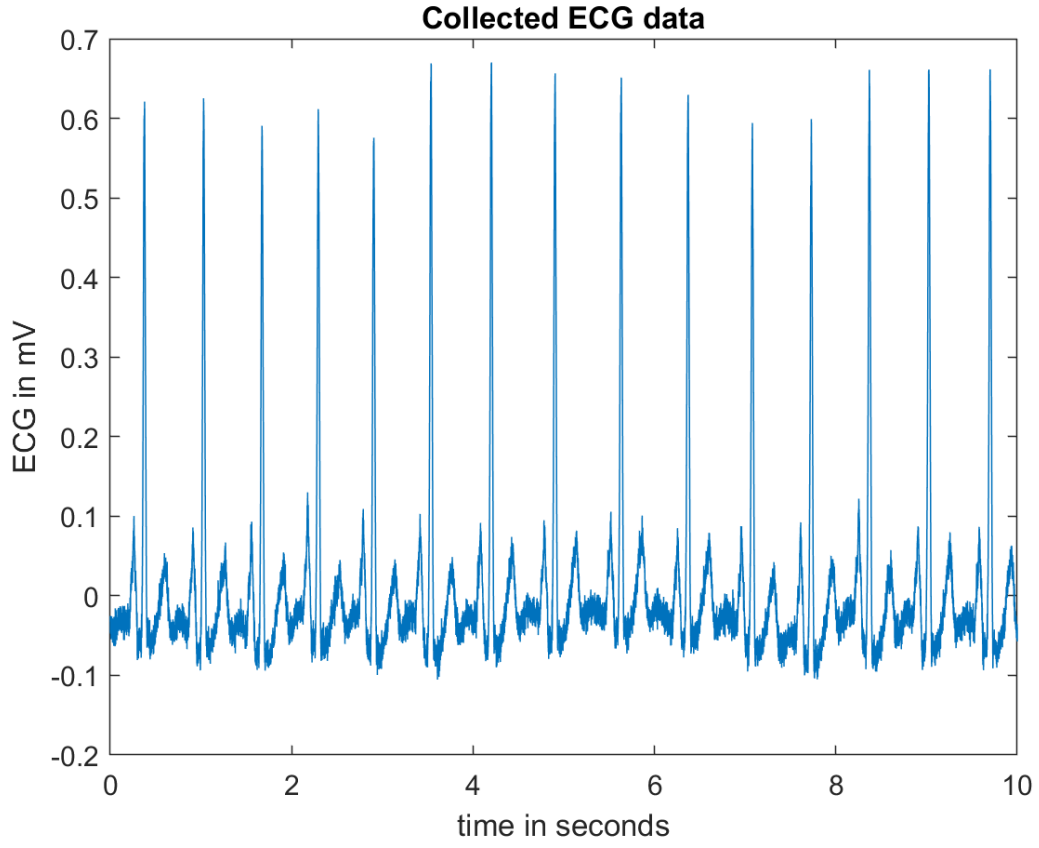
```
% make sure drivers for NI board are set
daqregister('nidaq')

% make sure kernel and variables are reset
% so that previous runs of the program don't
% contaminate collected data
clear all
close all

% assign the NI board as an analog input ai object
ai=analoginput('nidaq','Dev4');
chans=addchannel(ai,0);
% set the sampling rate for the NI board
set(ai,'SampleRate',8000)
% store current ai sampling rate setting in a variable
ActualRate=get(ai,'SampleRate')
% tell ai to start collecting data only when manually triggered
set(ai,'TriggerType','Manual')


duration=10; % measure 10 seconds of data
% set the number of samples ai should take after a trigger
set(ai,'SamplesPerTrigger',duration*ActualRate)
% store current ai samples per trigger setting in a variable
blocksize=get(ai,'SamplesPerTrigger')

% uncomment these to determine properties of the ADC
%prop=get(ai); % lists all properties for ai
%get(ai,'SampleRate') % return ai sample rate
%get(ai.Channel(1)) % return properties of ai channel 1

start(ai) % start ai timer
trigger(ai) % tell ai to start collecting data
ecg_data=getdata(ai) % store collected data in a variable

delete(ai) % delete the ai object
clear ai % remove analog input from workspace

Ts=1/ActualRate; % calculate sampling period (time between samples)
time=(1:length(ecg_data))*Ts % determine time associated w/ each sample

% plot time vs collected samples
figure(1)
plot(time, ecg_data)

% save collected samples in a file
save('julia_notchoff') % edit filename for each run

```

A different Matlab program was used to plot and analyze ECG data in more detail.

Figure 6. Matlab code used to extract and plot ECG data.

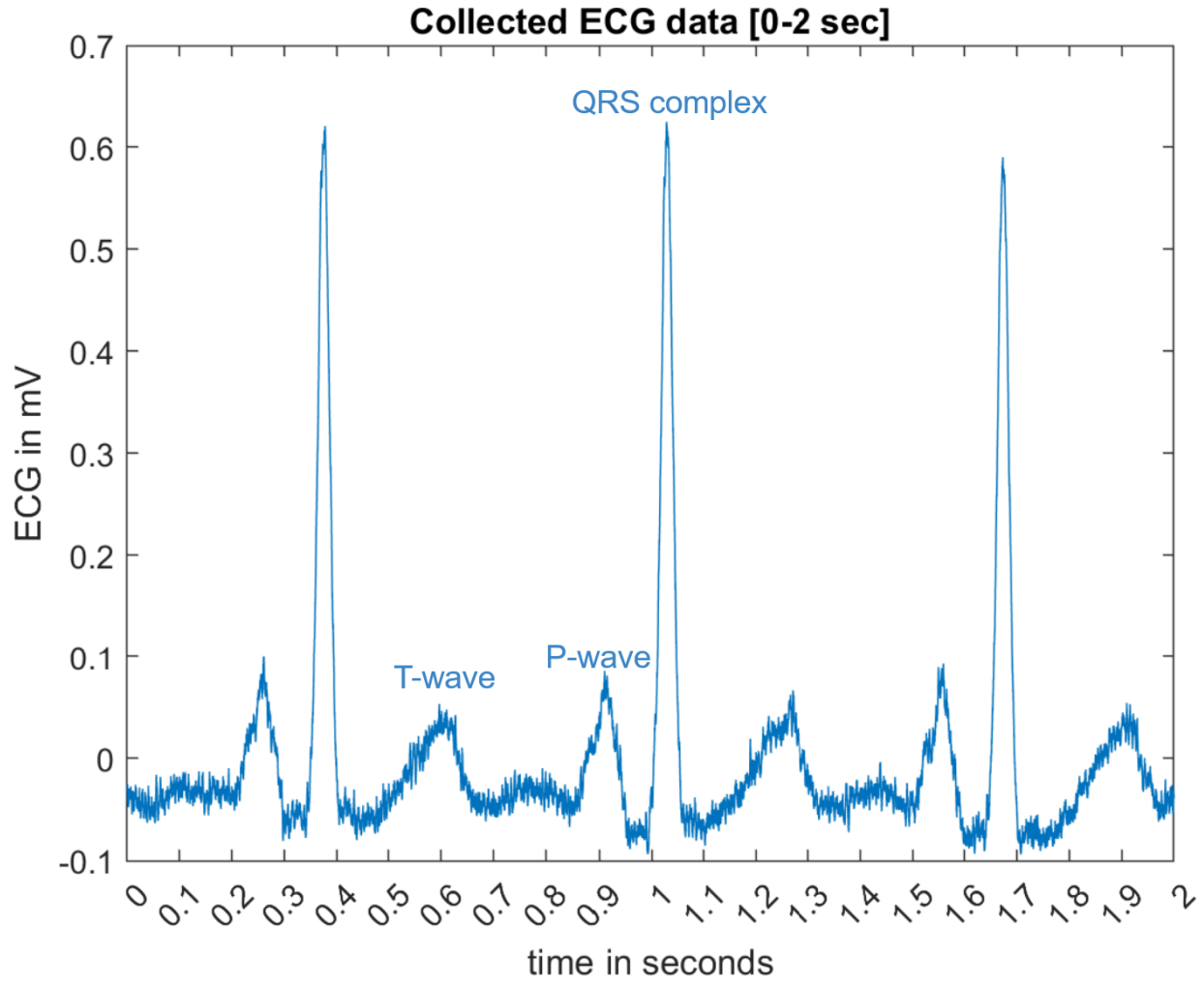
```
clear
close all

% constants
ts = 1.25e-04;
rate = 8000;
samplelength = 10;
samplesize = rate*samplelength;
amp = 5000;
v_to_mv = 1000;

% load relevant data collected with notch filter off
load julia_notchoff.mat;
ecg_notchoff = (ecg_data / amp) * v_to_mv;
time_notchoff = time;

% load relevant data collected with notch filter on
load julia_notchon.mat;
ecg_notchon = (ecg_data / amp) * v_to_mv;
time_notchon = time;

% plot ECG data
figure
plot(time_notchon, ecg_notchon);
title('Collected ECG data');
xlabel('time in seconds');
ylabel('ECG in mV');
```

Figure 5. Ten seconds of ECG data.



It is difficult to distinguish between the features of each heartbeat when all ten seconds of data are plotted, so looking at a plot of only a few seconds of data may be clearer.

Figure 6. First two seconds of ECG data.



There are three main features visible in the signal plotted above. The first feature of a heartbeat is the P-wave. This is shown on the ECG as a low-amplitude (about 0.1mV) peak with a width of about 0.1 seconds. A P-wave is the start of the pump as the atria depolarize and begin to contract. The next feature is the QRS complex. This is the most prominent feature of an ECG. It is shown as a higher-amplitude (just over 0.6mV) peak with a width of less than 0.1 seconds. The QRS complex is the main part of a heartbeat as the ventricles depolarize and begin to contract. The final feature of a heartbeat is the T-wave. This is shown on the ECG as a low-amplitude (less than 0.1mV) peak with a width of about 0.3 seconds. The T-wave is the end of the pump as the ventricles repolarize and relax.

The signal displayed noticeable differences when the notch filter was used compared to when the notch filter was not used.

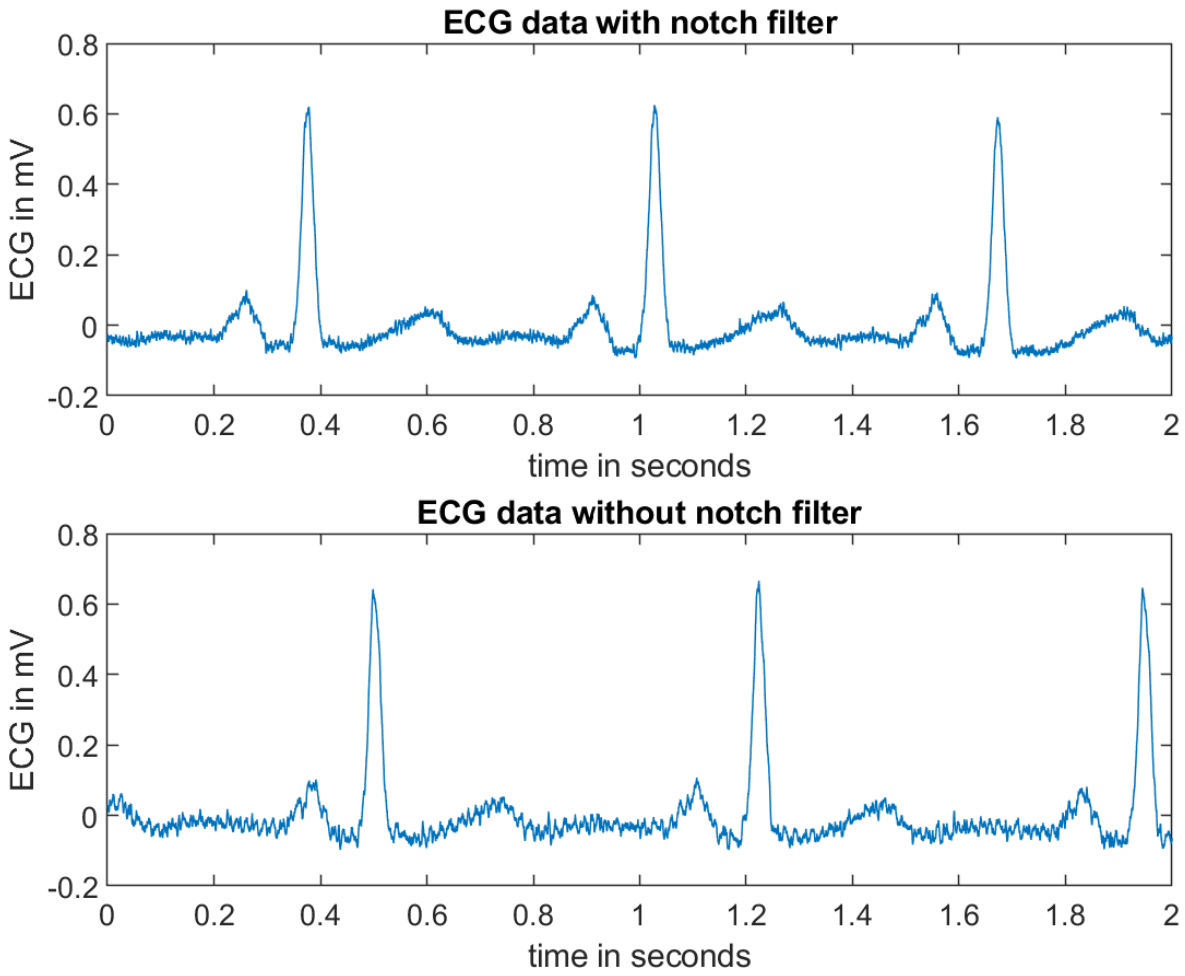
Figure 7. Matlab code used to calculate and plot FFT of ECG data with and without notch filter.

```
% plot ECG with and without notch filter
figure
subplot(2,1,1)
plot(time_notchon, ecg_notchon);
title('ECG data with notch filter');
xlim([0,2]);
xlabel('time in seconds');
ylabel('ECG in mV');

subplot(2,1,2)
plot(time_notchoff, ecg_notchoff);
title('ECG data without notch filter');
xlim([0,2]);
xlabel('time in seconds');
ylabel('ECG in mV');


% calculate FFT with and without notch filter

% make sure omega is same length as fft
k = 0:2047;
omega = k*((2*pi)/2048);

% find the fft for each digit
fft_notchon = fft(ecg_notchon,2048);
fft_notchoff = fft(ecg_notchoff,2048);

% convert omega_k to actual frequency in Hz
conversion = rate / (2*pi);
freq = omega * conversion;
```

```
% plot FFT with and without notch filter -- zoomed in at 0-1000 Hz
figure
subplot(2,1,1)
plot(freq, abs(fft_notchon));
xline(60);
title('FFT with notch filter [0-1000Hz]');
xlim([0,1000])
ylim([0,60])
xlabel('frequency in Hz');
ylabel('magnitude of the FFT');

subplot(2,1,2)
plot(freq, abs(fft_notchoff));
xline(60);
title('FFT without notch filter [0-1000Hz]');
xlim([0,1000])
ylim([0,60])
xlabel('frequency in Hz');
ylabel('magnitude of the FFT');
```

```
% plot FFT with and without notch filter -- zoomed in at 0-400 Hz
figure
plot(freq, abs(fft_notchoff), 'r-', freq, abs(fft_notchon), 'b-');
xline(60)
title('FFT with and without 60Hz notch filter [0-400Hz]');
xlim([0,400])
xlabel('frequency in Hz');
ylabel('magnitude of the FFT');
```

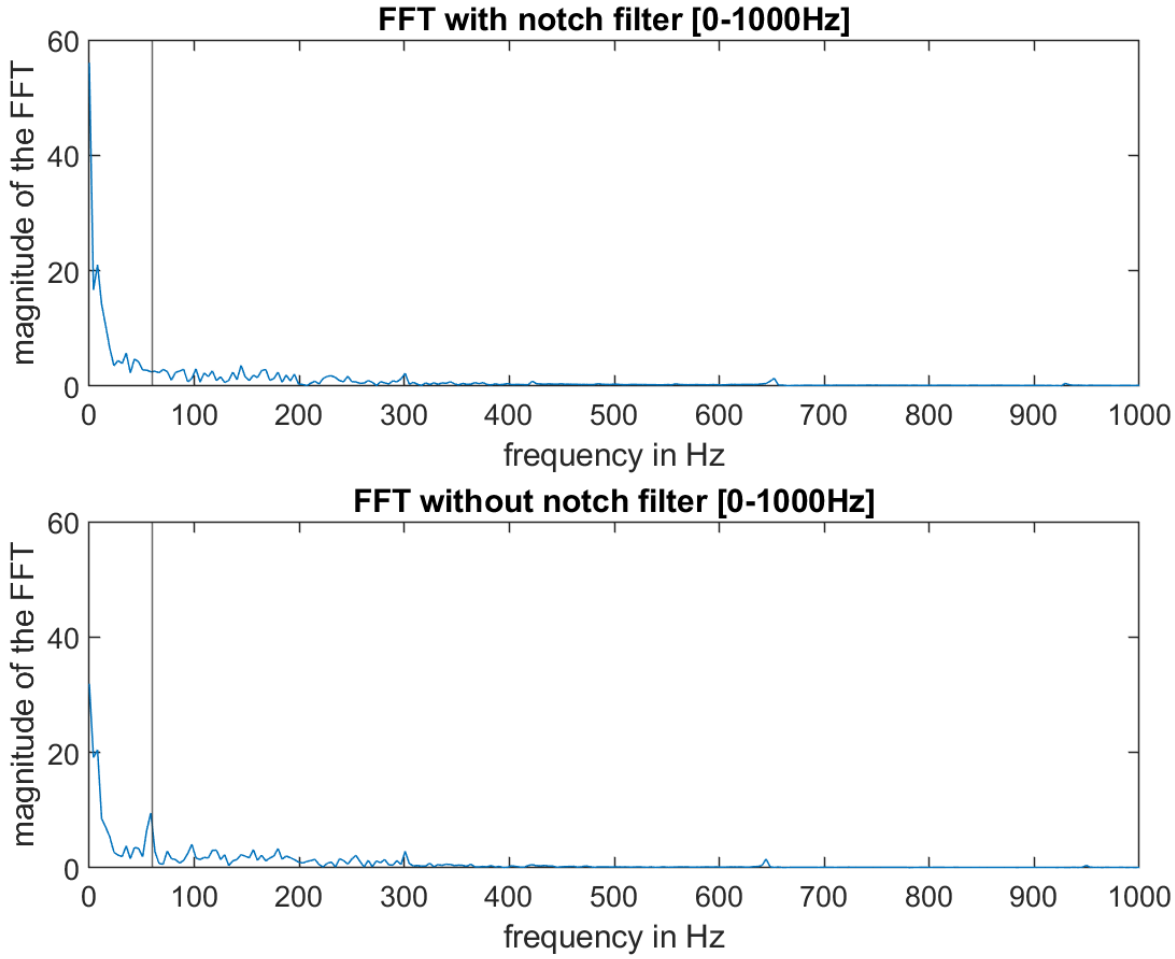
Figure 8. ECG data with and without notch filter.



The two datasets show similar shapes and amplitudes. However, the data collected without the notch filter is much noisier. The amplitude of higher-frequency noise is a lot higher, and it does slightly obscure the low-amplitude T-wave.

Since the notch filter is meant to remove noise based on frequency, it is also useful to compare the Fourier Transforms of each dataset.

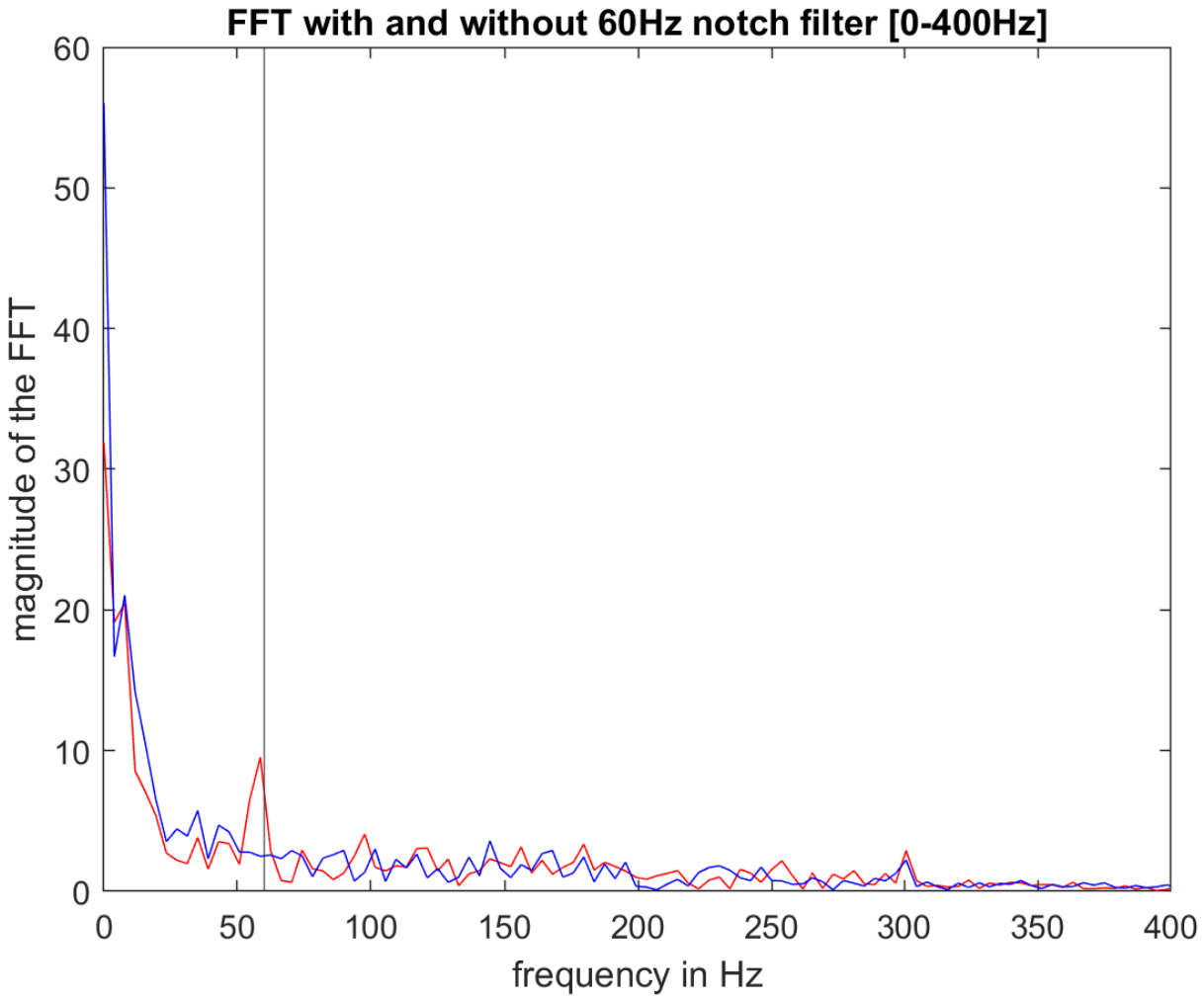
Figure 9. FFT of ECG data with and without notch filter. Vertical line at 60Hz.



These Fourier Transforms show that the dataset with the notch filter has a very low magnitude around 60Hz (indicated with a vertical line on the plots). The dataset without the notch filter has a noticeable spike around 60Hz. Both datasets display a decrease in amplitude for frequencies higher than 300Hz. This is the impact of the low-pass filter. For the most part, the frequency components greater than 300Hz have an amplitude close to or equal to zero. There is one exception - there's a small spike near 650 Hz, present in both datasets. It's possible that there's another source of noise at that frequency with such high magnitude that the low-pass filter could not completely get rid of it.

The low-frequency regions of the two datasets (except at 60Hz) look very similar. However, it is difficult to accurately compare them while they are plotted on different graphs. Plotting both datasets over each other addresses this.

Figure 10. FFT of ECG data with and without notch filter.
Data with notch filter shown in blue. Data without notch filter shown in red. Vertical line at 60Hz.



Plotting both datasets on the same graph, focusing on frequencies below 400Hz, shows that the low-frequency regions (except around 60Hz) are in fact very similar. The two plotted signals do not display spikes at all the same frequencies, but there are some small spikes common between both datasets. I think there may be a few small spikes present in the dataset without the notch filter that are not present in the dataset with the notch filter corresponding to harmonic frequencies (multiples) of 60Hz. However, it is difficult to tell, as all the spikes for both datasets are small and very similar in amplitude. Neither signal is consistently higher or lower amplitude than the other. This suggests that the difference between the data collected with the notch filter and the data collected without the notch filter really is only around 60Hz, and that the noisiness of the plotted ECG signal without the notch filter compared to the plotted ECG signal with the notch filter is only due to 60Hz power grid noise from equipment in the lab.

Think about an experiment or modification you could make to test something that you are curious about. State clearly what aspect of the ECG you are exploring, and explain your findings.

The ECG electrodes are placed on the body such that the left ankle is a positive lead, the right wrist is a negative lead, and the right ankle is a reference lead. I am curious what would happen if the right and left ankle electrodes were switched - so the left ankle would be used as a reference lead, and the right ankle would be used as a positive lead.

Figure 11. Electrodes on ankles and wrist.

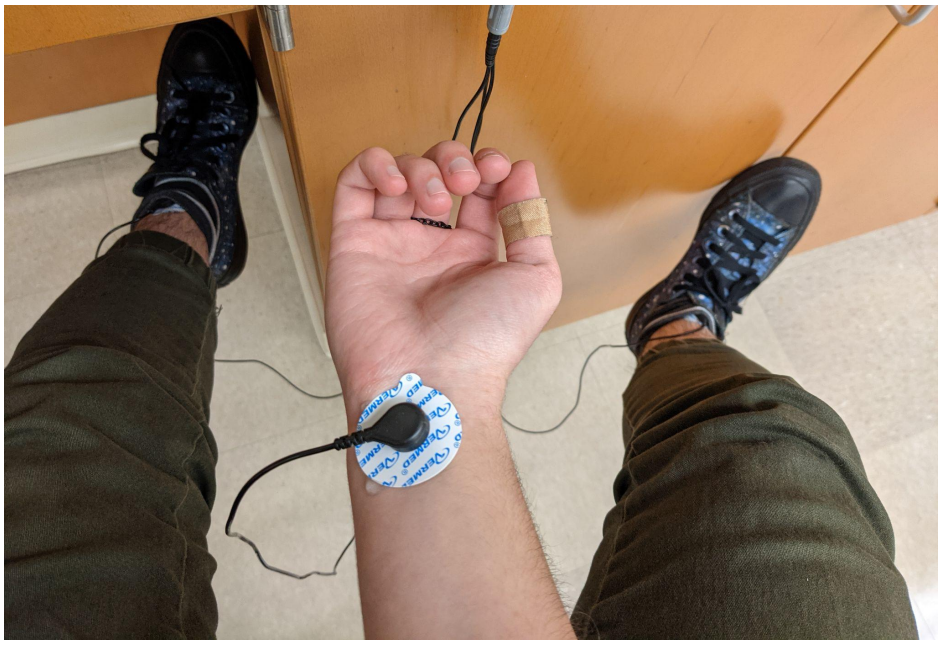


Figure 12. Oscilloscope displaying ECG measured with normal electrode placement.

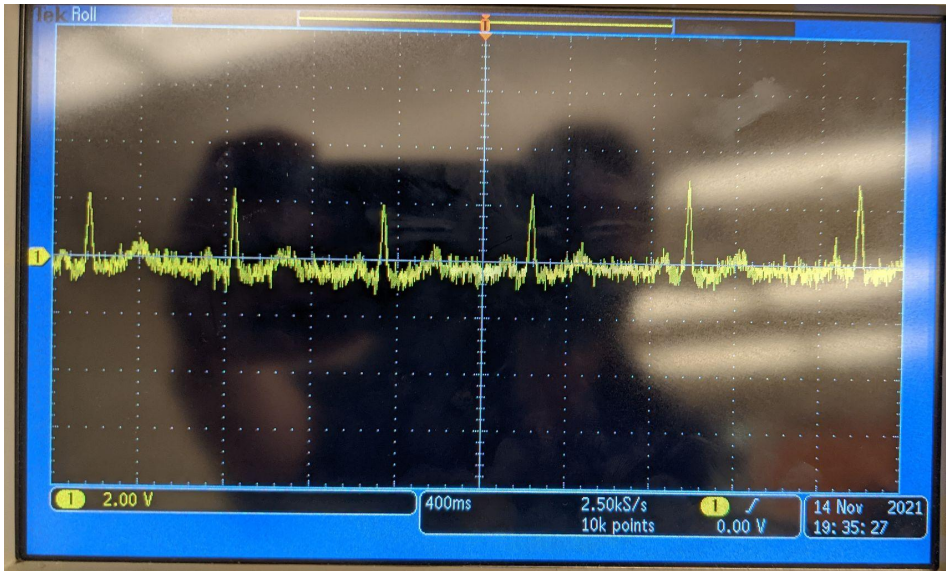
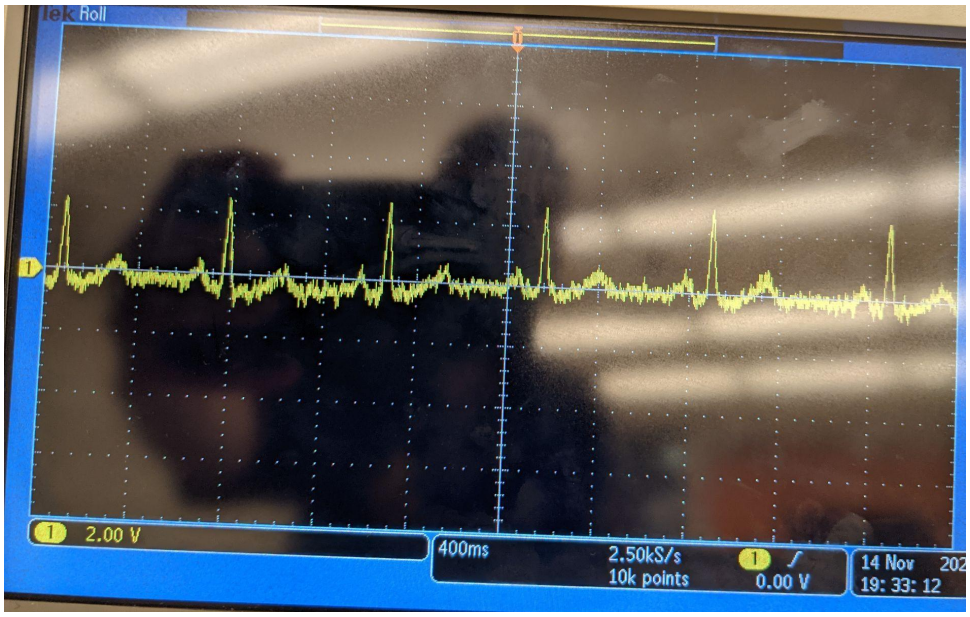
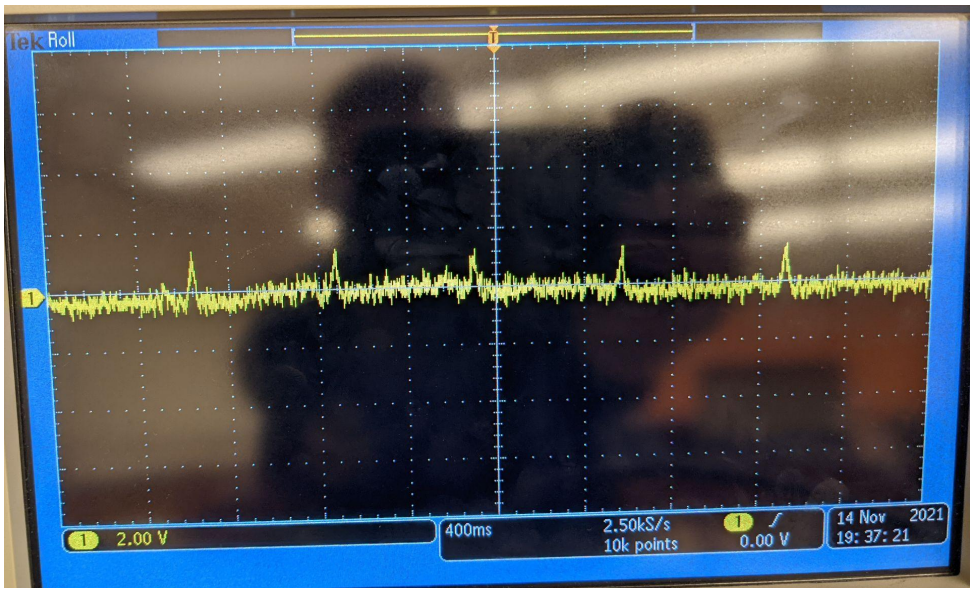


Figure 13. Oscilloscope displaying ECG measured with ankle electrodes switched.



Switching the ankle electrodes does not seem to impact the signal displayed on the oscilloscope. I was hoping to see some kind of difference, so I tried another electrode configuration--this time, I tried it with the ankle electrodes placed normally, but the left wrist used as the negative lead instead of the right wrist.

Figure 14. Oscilloscope displaying ECG measured with left wrist as negative lead.



This is a difference! Only one feature is visible. By its width, it looks like it is the QRS complex, but its amplitude is smaller than measured with previous electrode configurations. The signal

between the QRS complex is fairly flat - if the P-wave or T-wave are being detected, their amplitudes are too low to differentiate from noise.