Lab 4 - EKG
Kenny Jeffris and Irene Kim

Question 1: Why are we using dual supply amplifiers? What might be the problem with using only a positive voltage supply to amplifier an EKG signal?

You need a negative supply amp so that the system can handle bipolar input (both positive and negative voltages)

Question 2: What is the gain of your instrumentation amp (and ONLY the in-amp) at a frequency of 10 Hz? How does it change as you increase the frequency to 1 MHz? Why? At what frequency does the amp go to a gain of 1 (i.e. output is same voltage as input)?

At a frequency of 10 Hz, our gain was 52.1 (Vp-p for output = 3.438 V, Vp-p for function generator = 66 mV → 3.438/0.07). As we increased frequency to 1MHz, our gain was 15.1 (Vp-p for output = 1 V and Vp-p for function generator = 66 mV). The gain decreased because op amps maintain closed loop gain because it has high open loop at low frequency. The closest gain of 1 we got was 13.6 at 4 MHz.

Question 3: What is the gain of these first two stages together at 10 Hz? Start decreasing the frequency. At what frequency does the gain start to drop? What type of filter is this?

At a frequency of 10 Hz, Vp-p for output = 6.5 V. Assume same Vp-p throughout as before. The gain was 98 = 6.5/0.66. At around 1 Hz the gain starts to drop because the Vp-p = 6 V. This indicates high pass filter.

Question 4: What is the overall gain at 10 Hz? Is your signal getting distorted and if so,why? If you increase the frequency, at what frequency does the gain start to drop? What type of filter is this? What’s the overall bandwidth of your system?

Vp-p for output = 14.7 V, so the gain is 222 = 14.7/0.066. Yes, our signal is getting distorted because there cutoff at the peaks (both positive and negative). At around 150 Hz, there is significant drop in gain because Vp-p = 12 V. This is a low pass filter. The overall bandwidth of the system is 150 Hz taking into account both high pass and low pass filters.

Question 5: What is the amplitude of the QRS complex? Freeze the scope display and use the cursors to measure the period. What is this person’s resting heart rate?

We used Irene as the human subject. Amplitude is 2.6V. The period is 1.50 Hz. 1 bpm = 1/60 Hz → 90 bpm is the resting heart rate.

Question 6: What happens when a partner in the group moves their hands in the vicinity of the electrodes? What happens when the “patient” moves? These two kinds of interference sources will look a bit different. Speculate where each comes from.

In both cases (partner moves hands, patient moves), there is an interference in the ECG signal and the output on the oscilloscope is not consistent. When the partner moves their hands in front of the leads, there is a lot of noise and the line between the QRS signals has noise and is not straight. This can be considered extra background noise from partner's own electrical signals. When the patient moved, the entire signal would shift erratically. This can be caused by the inconsistency of the electrical currents and its distances that the leads are picking up.

Question 7: You should see plenty of 60 Hz interference in your signal. What is the peak-to-peak amplitude of this interference? Now have the "patient" connect a third reference electrode from their back to ground and measure the peak-to-peak 60 Hz interference? Did it decrease?

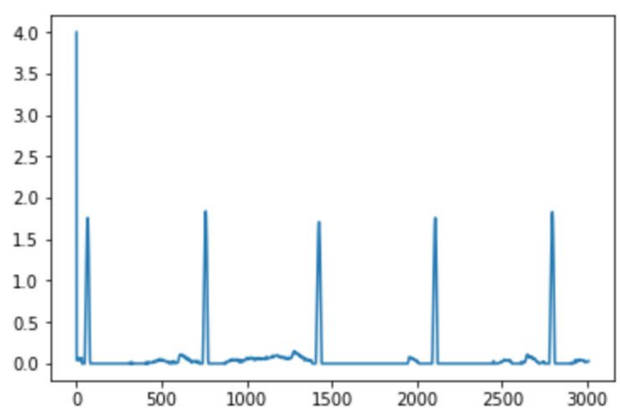
When we connected it to the patient's left and right wrists, the 60 Hz interference was the only signal we saw (about 2V amplitude). Then we changed it to a different configuration of the leads (one on the heart, one below the heart on the stomach, one on the ankle grounded) and we were able to decrease the 60 Hz drastically. Zooming into our signal, the 60 Hz interference amplitude is less than 50 mV.

Question 8: There is also a high frequency interference present in your signal. Tune your scope to the 10ns/div time scale and you should see something that looks very periodic. What is the frequency of this signal (try to estimate within 0.1 MHz)? What do you think this could be? Why would we see such a high frequency interference when we have a low pass filter in our circuit?

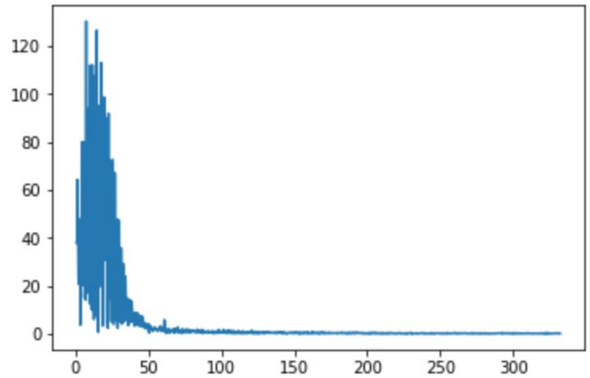
The frequency of the signal is 113.6 MHz. This can be picking up local radio signals, because a typical radio station emits around 100 MHz so it is not uncommon that our ECG is picking this up. For example, KDFC radio station is 89.9 MHz.

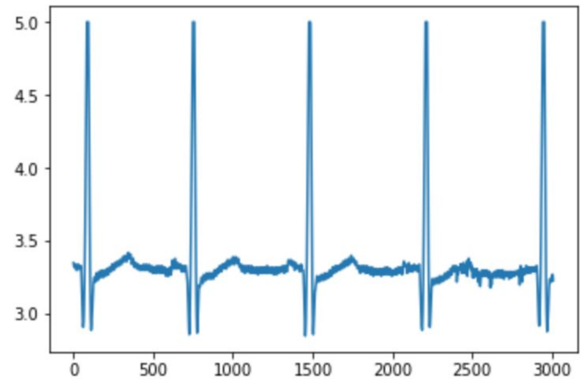
Question 9: Why are we connecting the circuit ground to the 3.3V pin of the Arduino, as opposed to the GND pin? Hint: disconnect the EKG leads from your lab partner, connect circuit ground to the GND pin, and reconnect the leads. How does your EKG signal change?

This allows us to see the full QRS complex, because if we connect ground to ground, we aren't able to see the negative signals from the leads (see following image).



Question 10: Save a plot of your EKG signal and its discrete fourier transform. Based on the bandwidth of your signal, what minimum sampling rate should you be using? Decrease the sampling rate within the send_serial.ino and dft code. At what point do you begin to see aliasing, and does this match your prediction?

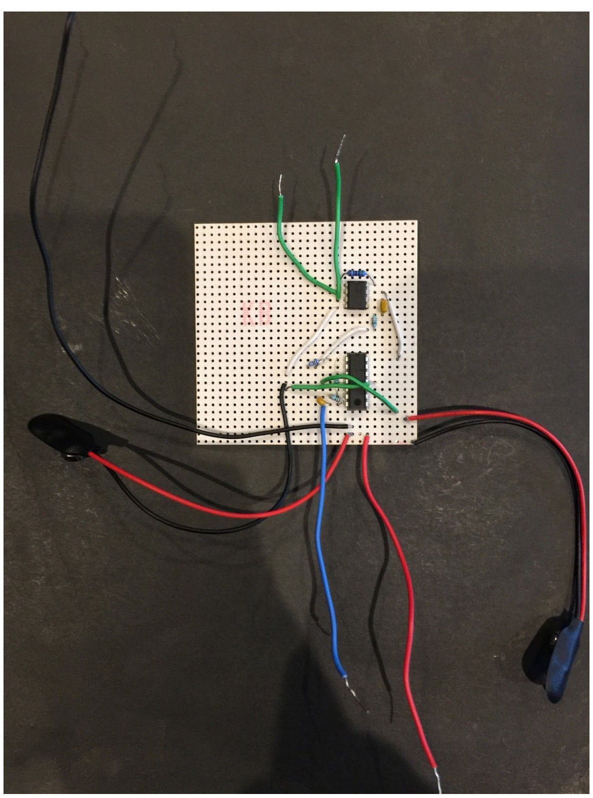


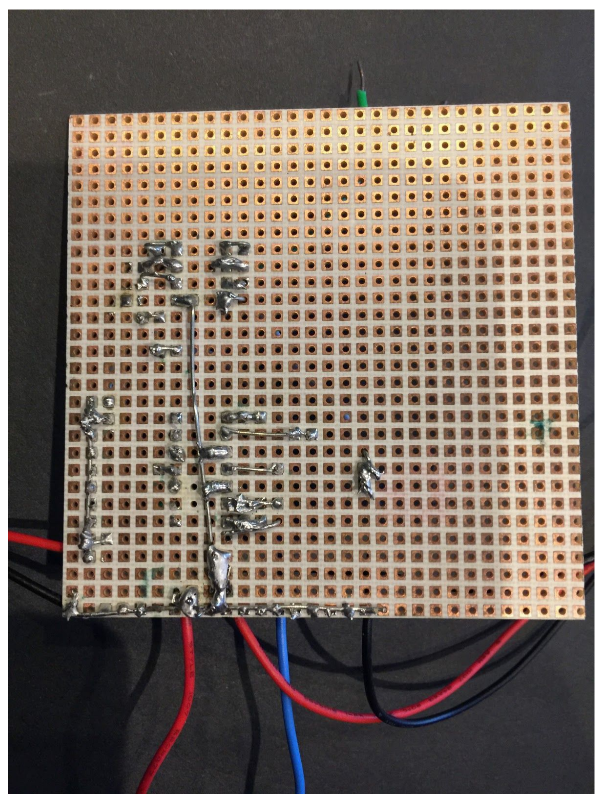


Looking at our dft plot, the majority of the signal lies in the 0-50 Hz range, with a small blip at 60 Hz (surprise surprise). Using Nyquist, we would choose a sampling rate of 100 Hz or greater. After decreasing the sampling rate, we noticed aliasing at the 80 Hz range, which matches our prediction!

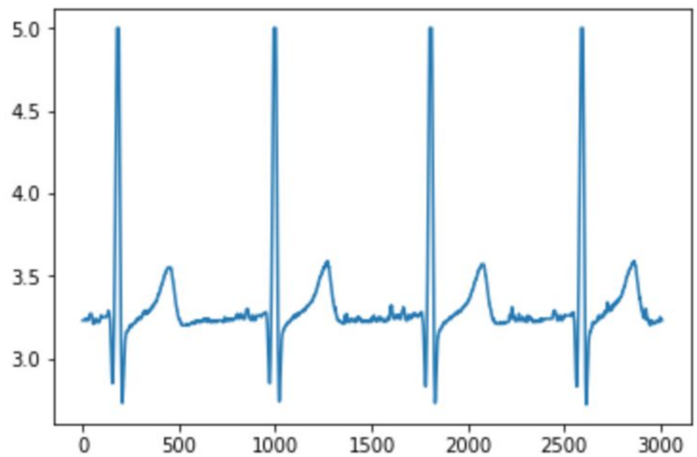
Extra Credit:

Hardware:





Using our PCB ECG, we obtained the following waveform:



The T-Wave is much more noticeable with this circuit!

Software:

We created an algorithm that takes in data from the iPython Notebook and calculates heart-rate in beats per minute:

```
import peakutils
# Must go into terminal and install peakutils by typing pip install peakutils onto your computer
# def heart_rate_monitor(signal):
   # Find R peaks (from QRS complex) within the data using a calibrated threshold (from
experimental trial)
   indices = peakutils.indexes(signal, thres=.25, min_dist=1)
   diffs = [] # Initialize difference list
   for x in range(len(indices) - 1):
      diffs.append(indices[x+1]/1000 - indices[x]/1000) # Find the difference between the points on
# the x-axis (time)
   avg = np.average(diffs) # Find the average of the distance between peaks
   heart_rate = 60*avg # Multiply by 60 to get into units of beats/minute (was in bps)
   print("Your heart rate is %d b/m" % heart_rate) # Print result
   return heart_rate

data = heart_rate_monitor(signal)
```