ECE 2660 Spring 2017

Final Project

08 May 2017

Team member Cesar Roucco - Circuit Designer

Team member Anthony Campos - Data Analysis Engineer

Laboratory Purpose

The project is meant to involve both individual and team work in the design and building of a circuit board. The circuit takes an audio signal from a source such as a phone or MP3 player and two LEDs flash according to the frequency content of the music. We needed to design the circuit so that one LED will respond to low frequencies and the other responds to high frequencies. The circuit board is designed to operate from a 9V battery and the input from the selected audio source, which in our case is a phone playing music.

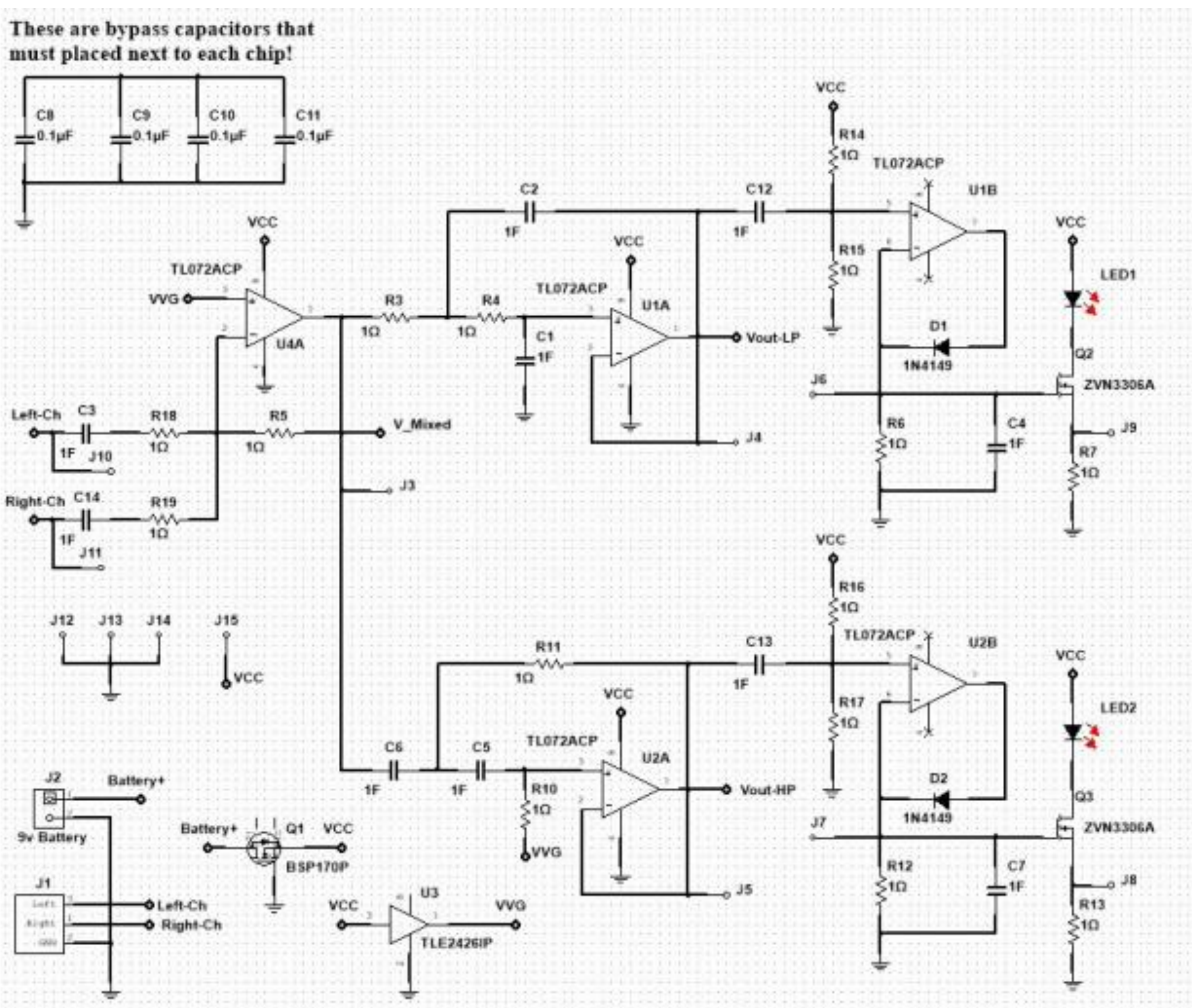
Experimental Equipment Employed

- NI Virtualbench (Power Supply, Function Generator, Oscilloscope)
- Laptop (used for data collection)
- Multisim (used for simulation)
- Resistors: 470 Ω (2), 1.8k Ω (1), 2.2k Ω (3), 10k Ω (4), 12k Ω (1), 33k Ω (2), 39k Ω (2)
- Capacitors: 0.01 µF (4), 0.1 µF (4), 10 µF (4), 220 µF (2)
- Diodes: 1N4149 (2)
- Op Amps: TL072ACP (3)

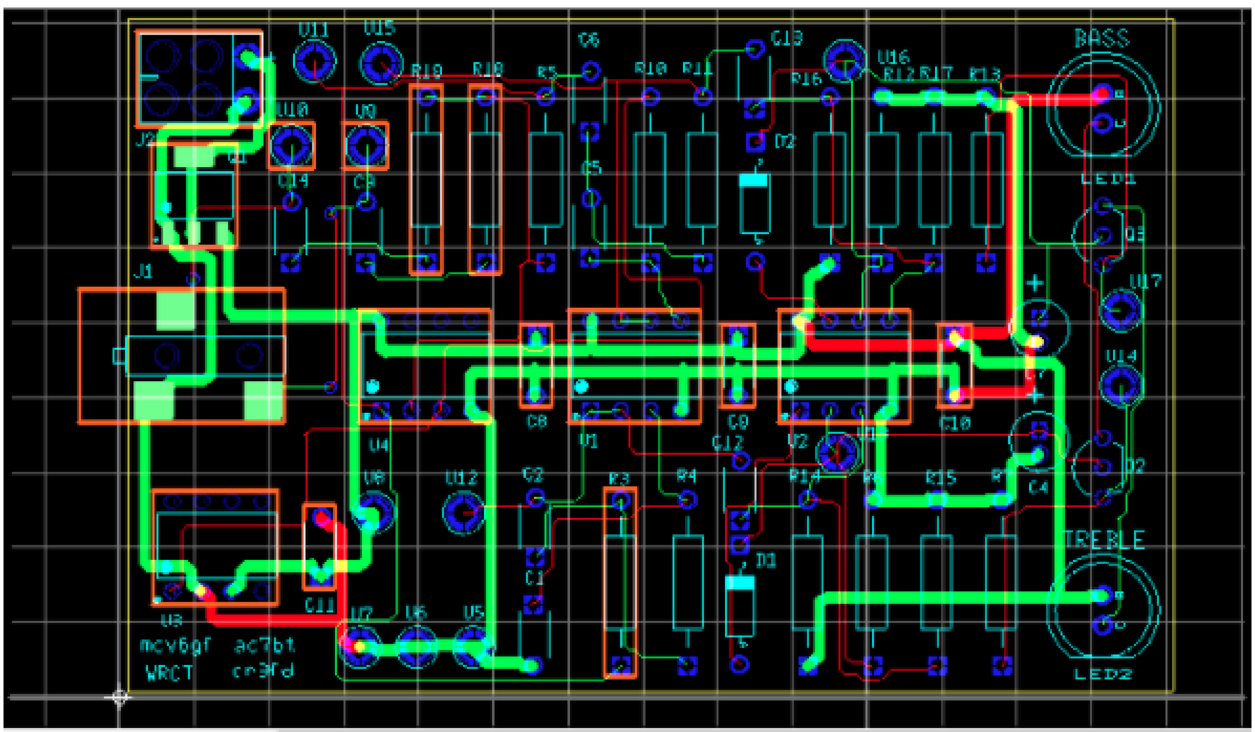
- LEDs: Red(1), Green(1)
- 9V Battery
- Soldering Iron

Experimental Procedures

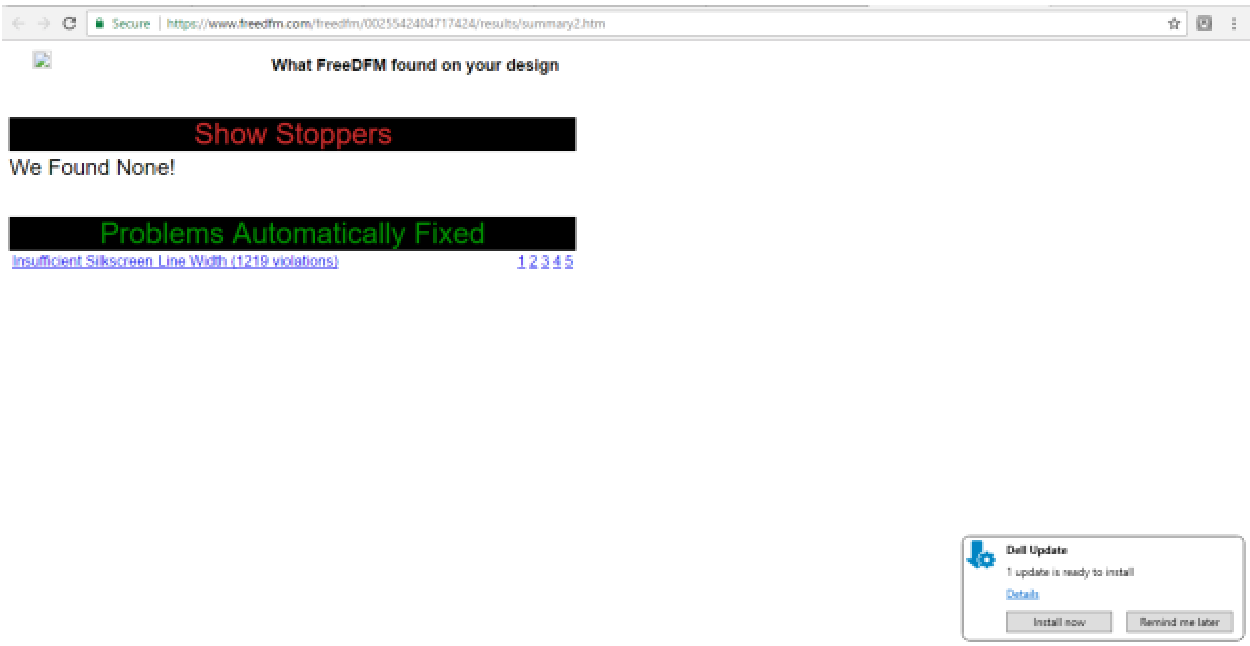
We began this project by downloading the files provided to us on the Collab website and reading the project description so that we could follow along with Professor Powell's tutorial on how to use Ultiboard. Using Professor Powell's tutorial and the Multisim file provided to us (shown below) we completed a board layout, and submitted Coty's board to freedfm.com, whom no longer is in the group.



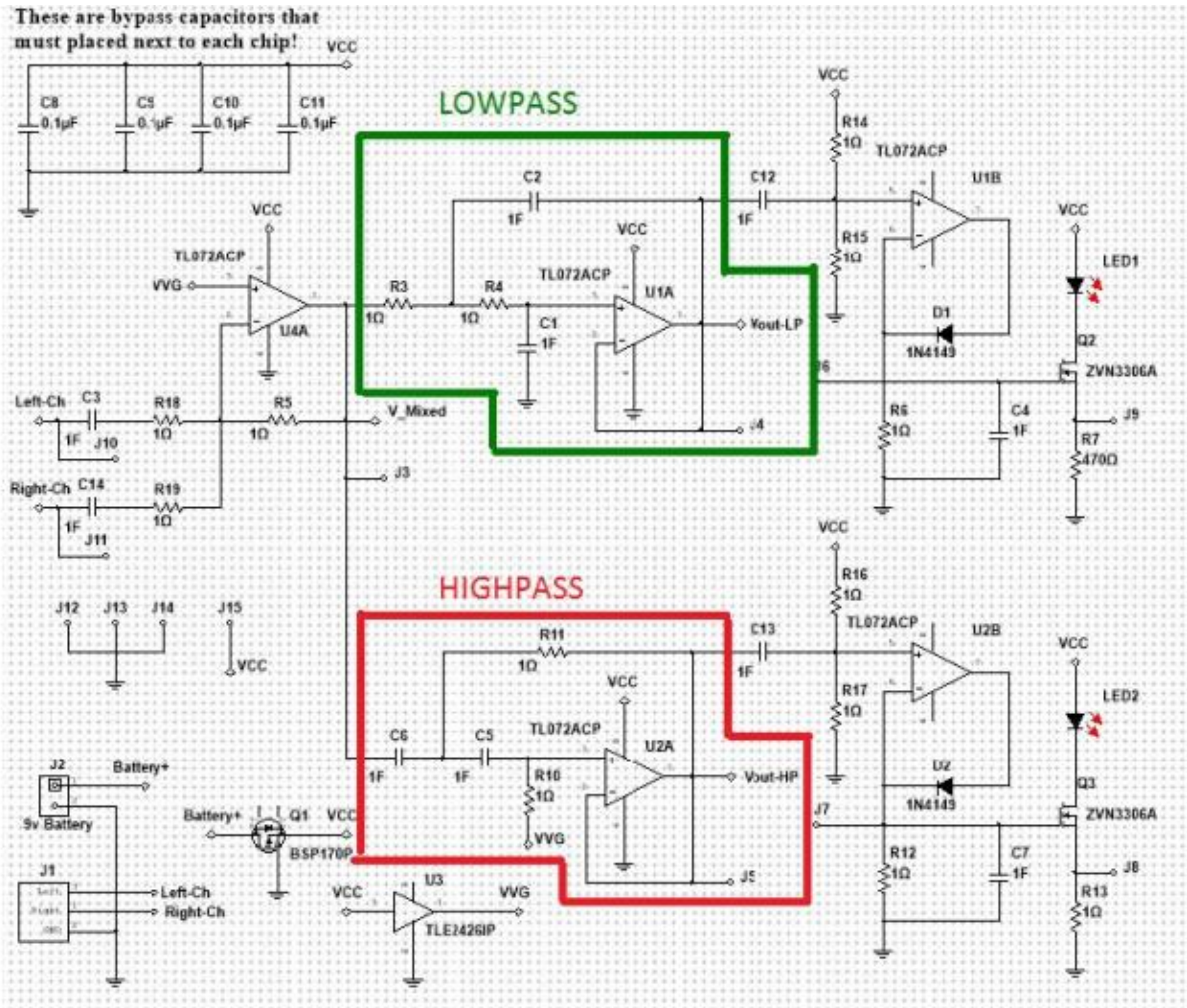
Using the Multisim file shown above, each of us were able to produce a board that had no show stoppers, so it was quite difficult picking which board to use as they were all very well designed. In the end, we decided to go with Coty's board because although our layouts all worked, we liked the layout of Coty's the best as it followed more closely in design to the Multisim simulation, had less vias, was neater and smaller trace routes. Below is a screenshot of the Ultiboard file of the board layout we decided to use.



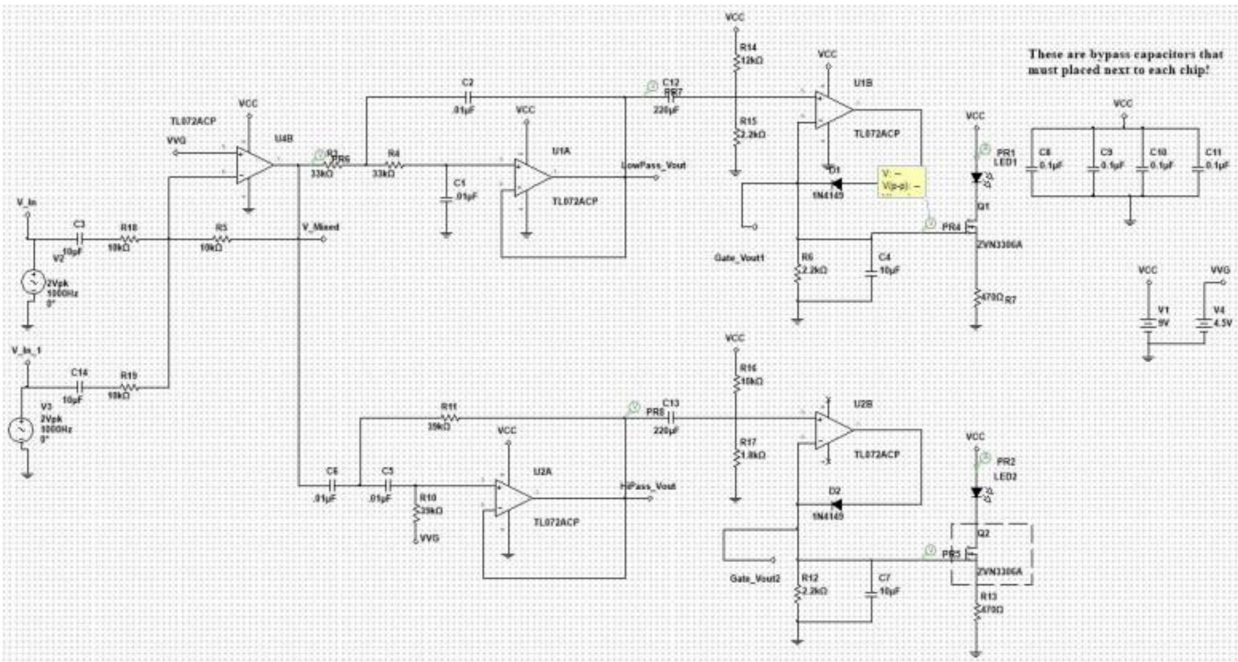
Using this design, the FreeDFM website found no show stoppers as shown below.



Once we had our board layout, the next step was to find the values for the components on our board. We decided that the best way to start this process was by finding resistors and capacitors values for the Sallen key high pass and low pass filters. The picture shown below shows where the high pass and low pass filters are located on our overall design.



The design and calculations sections of this lab report explains in detail how we found the values for the components on our board. Once we had the values for the components of the Sallen key filters, we calculated and tested different values to put in for the rest of the components in our design. After calculating values for our components, we inserted them into our Multisim design and simulated the design. Overall, we calculated the values for the high and low pass filters, the time constant, the voltage dividers that determined how much voltage would go into the LEDs, the current limiting resistor of the LED, and the summing amplifier. For the capacitors that were between the Sallen key filters and the voltage dividers (C12 and C13) we experimented with different values as there was no way that we knew of to calculate these values. We simulated the design in Multisim by putting in different frequencies and seeing when the red and green LEDs would light up. Once the LEDs lit up and turned off at the frequencies that we wanted them to, we decided our design was successful and would work correctly when we built and tested on our PC board. The following image shows what our Multisim design looked like with all our values computed and inserted in their places.



Once we were sure that the values chosen would work for the project's purposes, we began soldering components into our PC board. After testing our board the first time, we saw that our calculations were still a little wrong and that our LEDs were not lighting up how we wanted them to. To fix this, we ran more simulations in Multisim and ended up having to unsolder our R17 component, which we then changed to be 1.8k Ω instead of 3.3k Ω. We tested the board once again after soldering in the new resistor and the board functioned exactly how we had wanted it to.

Design

We began the design process by first choosing a piece of music to base our design off of. We wanted a song that has a large range of frequencies and can expertly show the subtle changes in frequency being played by the song that our device can indicate. It was important that we knew the song beforehand so that we could design the high and low pass filters with the frequency range of the song in mind to obtain the optimal performance of our project. Because me and Cesar usually study with rap music for DLD and FUN2, decided to pick “All We Got” from Chance The Rapper as the piece of music that we would base our design off of. To determine that the song we picked would satisfy our requirement of having a large range of frequencies, we ran the song’s signal through Virtualbench and took the FFT of the signal. We saw from the peaks of the FFT, that the range of our song would run anywhere from 20Hz to 12.804kHz as shown in the screen capture below. Using this knowledge, we decided that we wanted the cutoff for the high and low pass frequency to occur somewhere around 440 Hz (A4) as that was the note heard from a standard tuning fork and our song displayed ranges of frequencies on either side of this value.

Our circuit board was designed with the intention of displaying a range of high frequencies above a cutoff by varying the amount of light emitted by a red LED. Essentially, we configured our circuit to divert all high frequencies above 408 Hz to "pass" through to our red LED. We did this by setting values of two 39k Ω resistors and two 0.01 µF capacitors in our high pass Sallen key filter. A range of low frequencies below a cutoff was then decidedly shown by the varying amount of light emitted by a green LED. We configured our circuit to send low frequencies below the cutoff of 482 Hz to the green LED. The varying voltage delivered to the LED's would then increase or decrease the amount of light emitted by the LED dependent upon its frequency as the audio signal changes with the music. We chose ranges that would slightly overlap that way as one LED begins to dim and turn off, the other LED would be lighting up. Additionally, we would then slightly tweak our values of our voltage divider networks (R14, R15 and R16, R17) so that the brightness of the LED would be clearly seen when the frequency is clearly at a low or high value. Furthermore, we calculate the time constant for our LEDs to assure the LEDs would charge and discharge to show the changes in frequency of our songs. Afterwards, we calculated the current limiting resistor of the LEDs to assure that when we tested and ran our design, the LEDs would not blow up. Finally, after figuring the components on the left side of the circuit, we calculated the values of the summing amp which was important since the audio jack split the audio output from our phone to a left and right channel so we had to recombine those two signals before filtering it out.

So how does the reverse polarity protection, (the P Channel MOSFET) operates in this application?

P-Channel MOSFETs were used for reverse polarity protection. In order to ensure that our circuit would function properly, the (PMOS) FETs job is to ensure that our current is flowing in the proper direction. When the PMOSFET sees reverse polarity, the diode internal to the MOSFET becomes reversed and stops current flow. This acts as protection because any flow of reverse current can result in damaging of the circuit or battery.

Calculations

For both Sallen-Key filters, we decided on definitely using equal and reasonable capacitor values as they were harder to customize. We then used the following calculations to solve for values of resistance, which luckily, we were able to find pairs of identical values for the desired frequencies without having a large variety of parts.

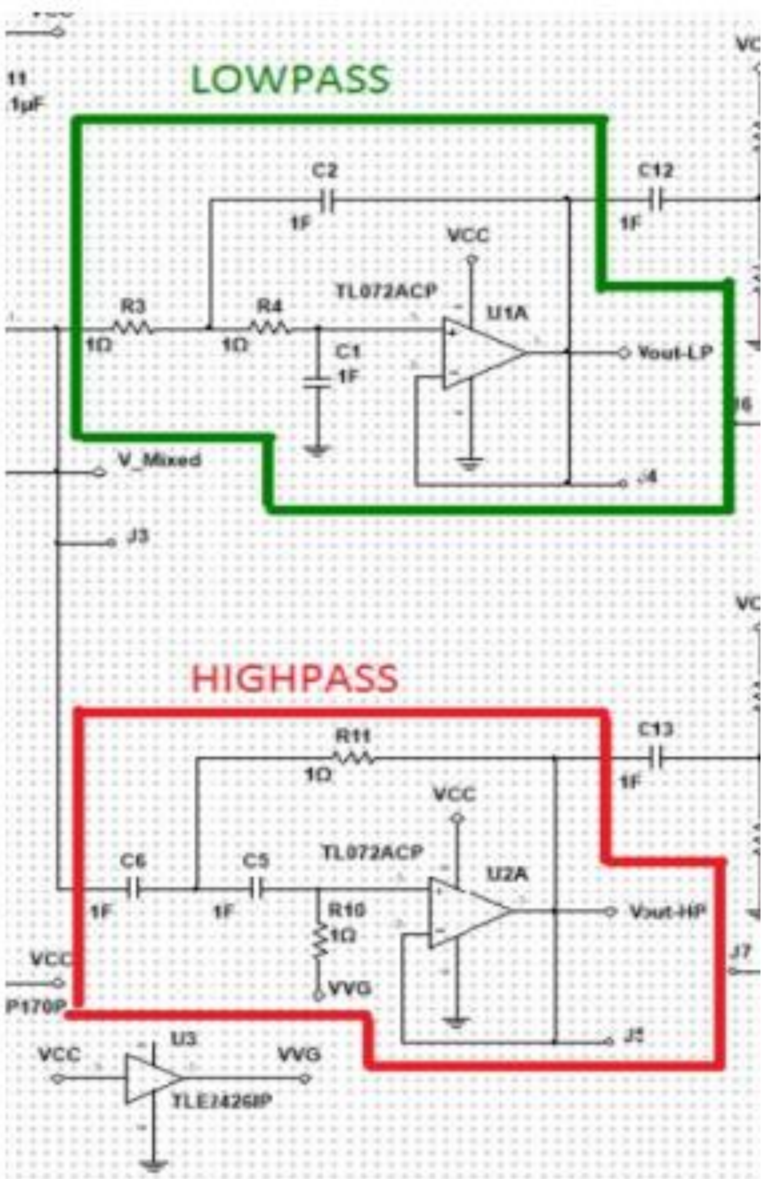
High Pass Sallen-Key Filter:

$\omega_0 = \sqrt{1/(R_{10}R_{11}C_5C_6)}$ and $f = \omega_0/2\pi$, by using values of $R_{10} = R_{11} = 39k$ and $C_5 = C_6 = 0.01uF$, we find that $f = 2564 / 2\pi = 408\ Hz$

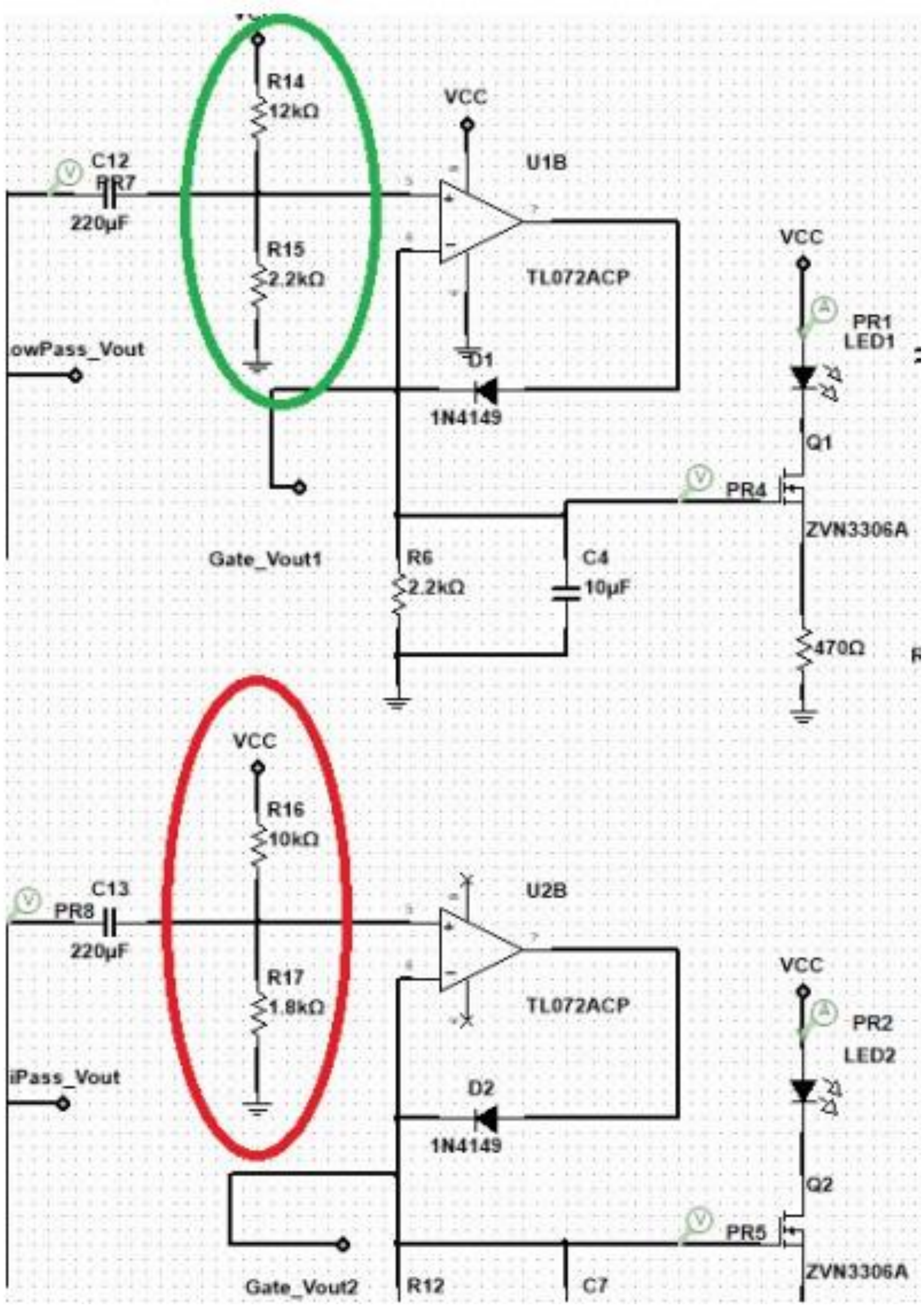
Low Pass Sallen-Key Filter:

$\omega_0 = \sqrt{1/(R_3 R_4 C_1 C_2)}$ and $f = \omega_0/2\pi$, by using values of $R_3 = R_4 = 33k$ and $C_1 = C_2 = 0.01uF$, we find that $f = 3030 / 2\pi = 482\ Hz$

The aforementioned values we calculated were then put into the following portion of our circuit.



Once we determined the cutoff frequencies of our high and low pass filters, we moved on to calculating the voltage divider used to determine the gate voltage of the MOSFET that determines the current flowing through the red and green LEDs shown below.



As you can see from the screen capture above, we decided that for the low pass filter, resistor R14 would be 12kΩ and R15 would be 2.2kΩ. For the high pass filter, resistor R16 and R17 would be 10kΩ and 1.8kΩ. These values were calculated by first finding at what voltage would the green and red LED turn on. We determined this voltage by applying a voltage to the LEDs from 0.0V to until we saw the LEDs turn on. For the green LED, we found that it would turn on at around 1.8V and for the Red LED, we found that it would turn on at around 1.5V. Using this knowledge, we then solved for a voltage divider equation to determine the ratio of the resistors for the high and low pass filter. Below is the voltage divider equation we used to solve for the resistor ratios for each LED.

$$V_{green\_out} = V_{in}\left(\frac{R15}{R14 + R15}\right)$$

$$V_{red\_out} = V_{in}\left(\frac{R17}{R16 + R18}\right)$$

From our experimentation, we knew that Vgreen_out would be 1.8V, Vred_out would be 1.5V and that for both equation Vin would be 9V as the power is coming from a 9V battery. We used this knowledge to perform the calculations below to determine the values of the resistors for the voltage divider.

Green LED

$$1.8V = 9\left(\frac{R15}{R14 + R15}\right)$$

$$0.2V = \left(\frac{R15}{R14 + R15}\right)$$

From this, we knew that we could pick any values for our resistors as long as the ratio is equal to 0.2V and a ratio that would give us 0.2V. We found that with our components available to us, the resistor values that would bring us closest to 0.2V were if set R15 to 2.2kΩ and R14 to 12kΩ.

$$\left(\frac{2.2k\Omega}{2.2k\Omega + 12k\Omega}\right) = 0.155V$$

Red LED

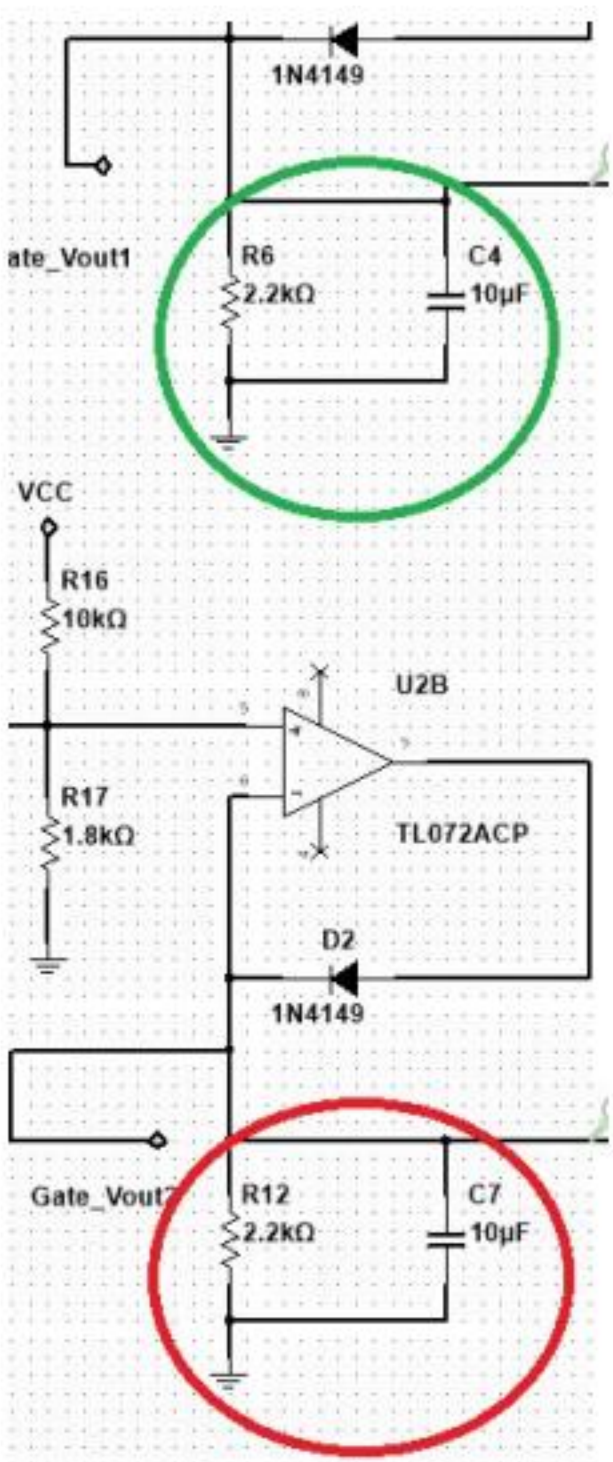
$$1.5V = 9\left(\frac{R17}{R16 + R17}\right)$$

$$0.1667V = \left(\frac{R15}{R14 + R15}\right)$$

Using similar process as for the green LED, we found that with our parts available, using R16 as 1.8k and R17 as 10k would give us a value close to 0.1667V.

$$\left(\frac{1.8k}{1.8k + 10k}\right) = 0.152V$$

After determining the values of the resistors for the voltage divider, we moved on to determine the values for our time constant which is affected by the resistors and capacitors shown below.



For this project, determining the value of time constant is important because the time constant determines how fast the LEDs charged/discharge, as a result, this affected how closely the LEDs will display the changes in frequencies from our song. Consequently, we wanted our time constant to be pretty small to allow the LEDs to charge/discharge in pace with our song. We decided that 20 milliseconds would be an ideal time constant so we solved for the resistor and capacitor value by the following method:

$\tau = RC$

We want tau to be approximately 20mA so:

$0.02 = RC$

We then Plugged in values for R and C that would give us our desired time constant

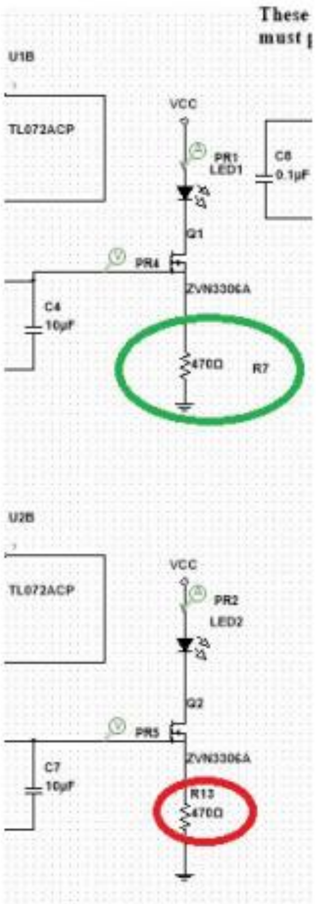
$0.02 = (2.2k)(10uF)$

we used 2.2k instead of 2 ohms because of what was available in our lab kits so:

$R6 = R12 = 2.2\Omega$

$C4 = C7 = 10uF$

After solving for the time constant, we solved for the current limiting resistors of the green and red LED shown in the area highlighted below.



The current limiting resistor is important in order to reduce the current in a circuit. The current limiting resistor placed in series with the LED allows us to control the amount of current going through the LED. A quick research of the basic red and green LED informed us that the max current of each LED was 20 mA so we needed to limit the current going through the LED to be below this. To calculate what resistor to use we simply used ohm's law since we knew that VCC was 9V and the max current was 20mA.

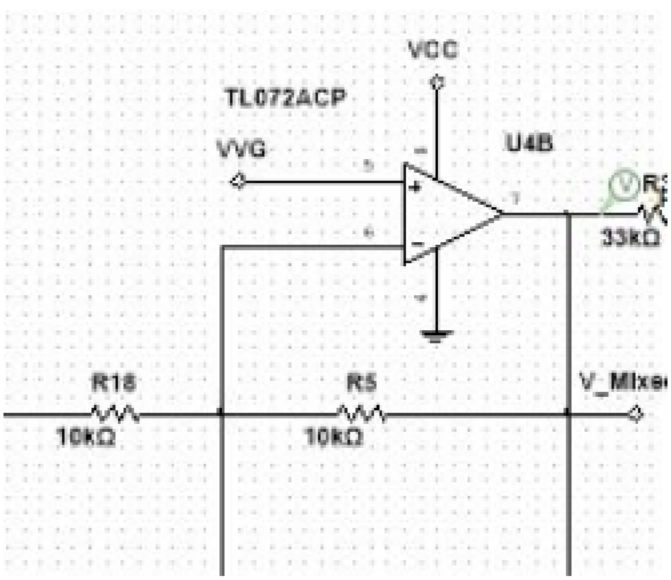
Calculating for the current limiting resistor:

$$V = IR$$

$$\frac{V}{I} = R$$

$$\frac{9V}{20mA} = 450\Omega$$

As you can see from above, we calculated the current limiting resistor to be 450Ω, however because of component limitation, the closest value we had to this was 470Ω so that was what we used for R7 and R13. The final thing we calculated for our design was the values for the summing amp which is shown below.

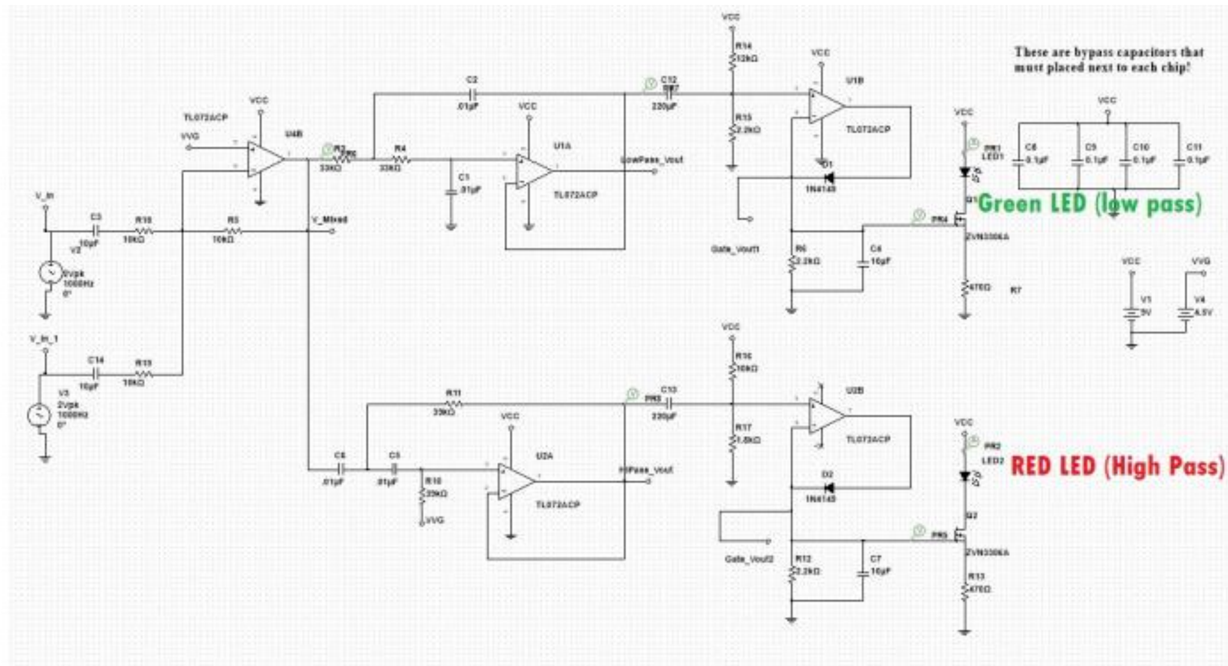


We wanted to design the summing amp to have a gain such that the output of the amp was not greater than the power supply. For this part, we sort of experimentally picked values for the resistor until we obtain gain that was not greater than the power supply. Below shows the calculation we did to determine the value of the summing amp resistors.

$$Vout = (2Vin)\left(\frac{10k}{\frac{1}{\frac{1}{10k}+\frac{1}{10k}}}\right)$$

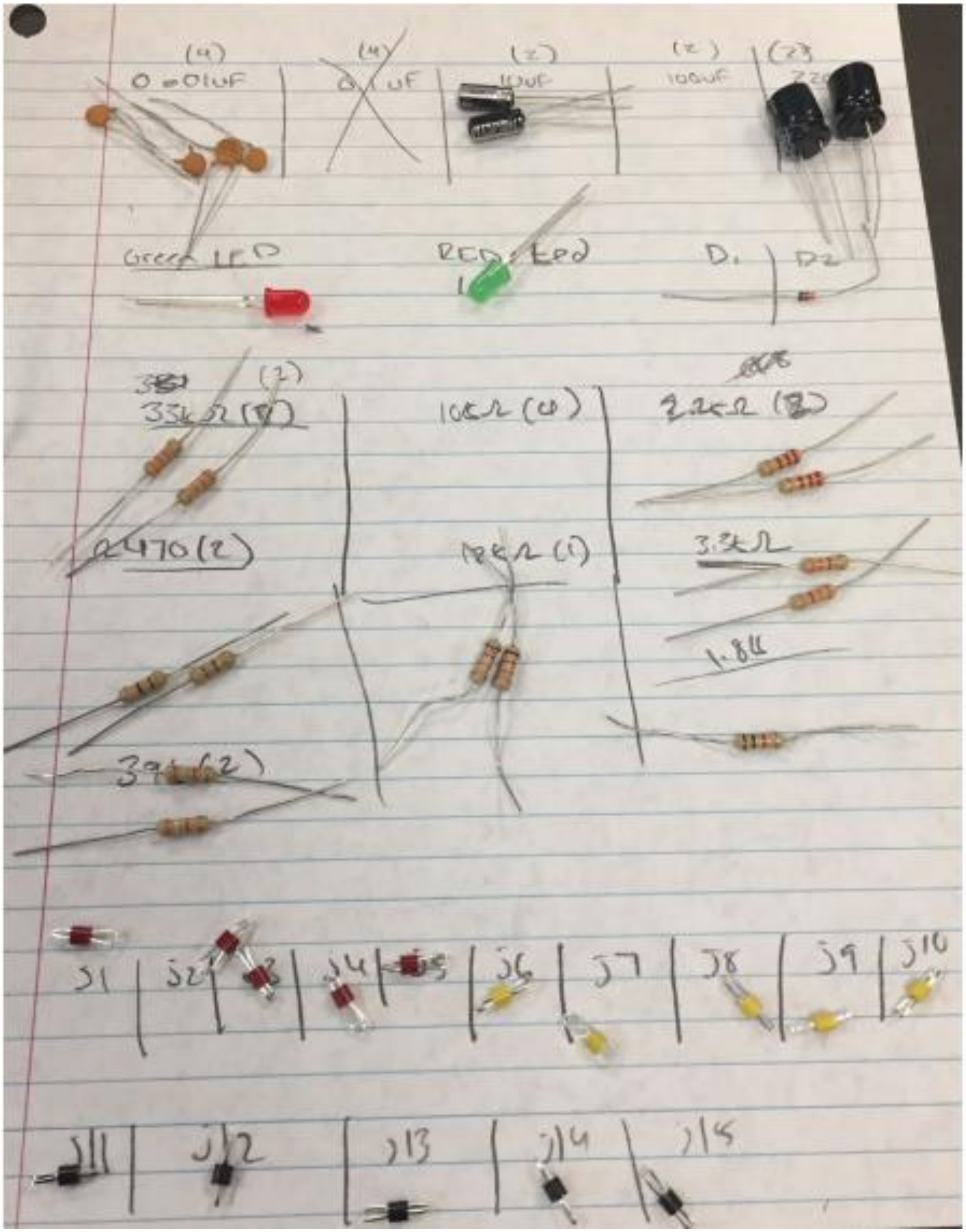
$$\frac{Vout}{Vin} = 2\left(\frac{10k}{5k}\right) = 4$$

As you can see from above, choosing R16, R5 and R19 as 10kΩ sets the gain of the summing amp to 4, which is less than the power supply voltage which was 9V. For this project, we were able to calculate values for almost all of the parts of the circuit except for C3, C14, C12 and C13. Instead, we determined these values experimentally, by plugging different values into Multisim until our circuit behaved as we wanted it to. It was impossible to calculate these values in advance because those values were dependent on the design of our circuit. We just needed to choose capacitor values that were high enough to hold voltage long enough so that it did not dissipate immediately.



Assembly Process

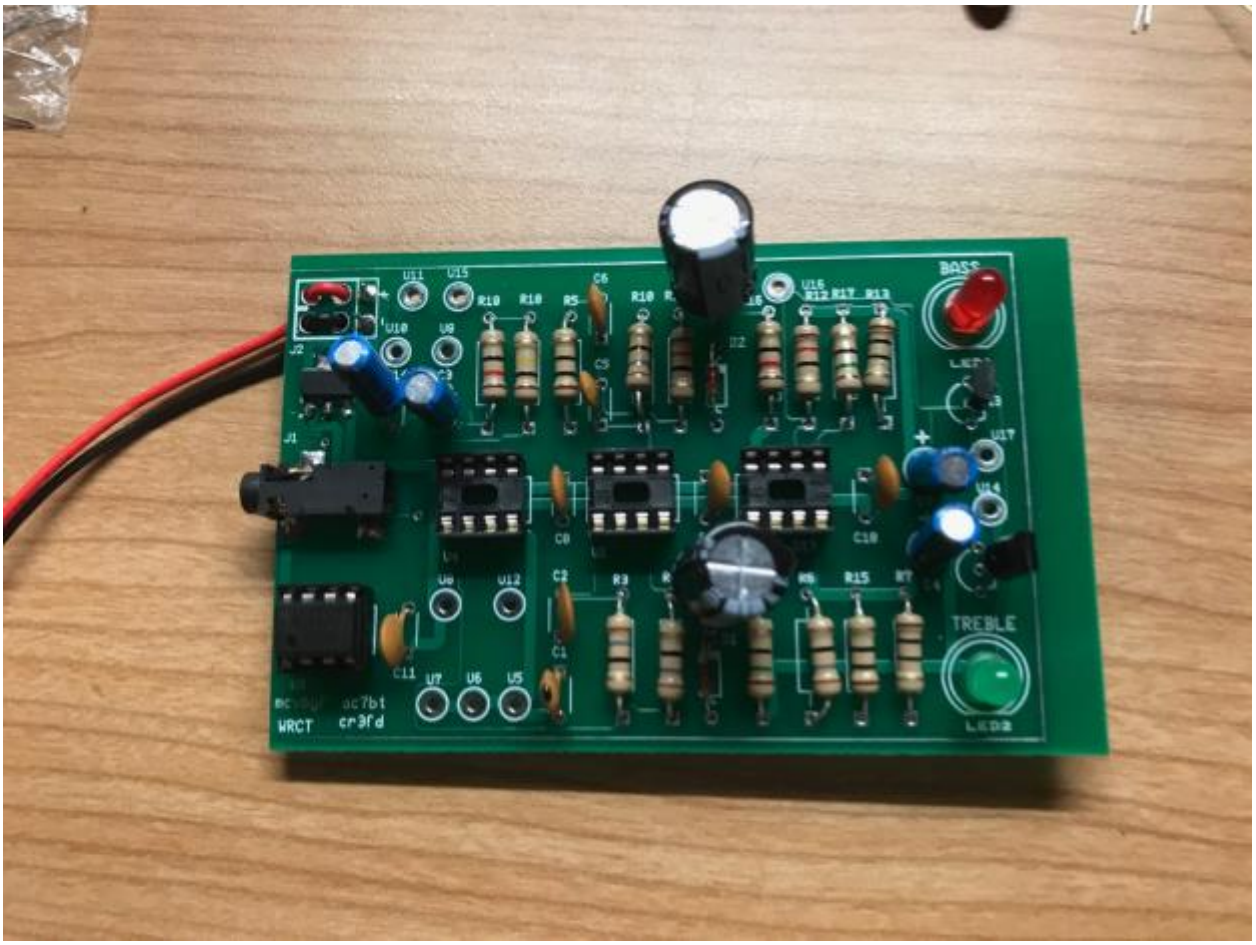
Below are images of our design process. The first image is a picture taken of all our components organized and ready to be soldered into our PC board. We did this so that we could be quick and efficient when soldering in our parts and to prevent us from making a mistake and mixing up resistors.



The first components we added were the ones we knew we couldn’t mess up. We soldered all of the non-valued components first including the audio jack, the battery connector, the p-channel

MOSFET, op amp connectors, and test points. The bypass capacitors were valued but at a fixed value so we did not have to worry about their values being incorrect.

Finally, we soldered the resistors onto the board. The picture to the right is the board right after the soldering was completed. Our next step was to test the board.



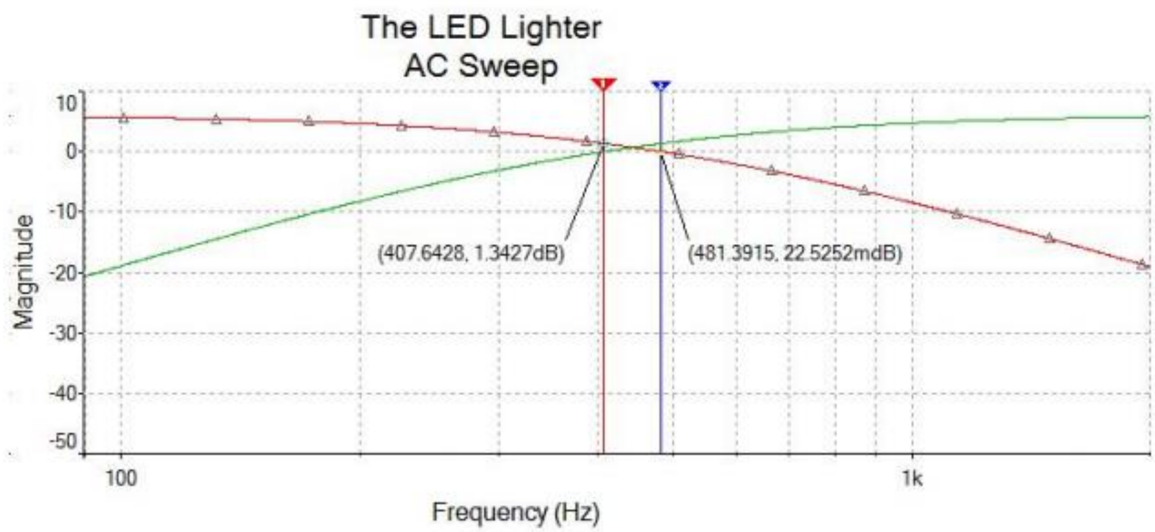
Simulation and Analysis

Below we have copies of our Multisim data collected for analysis.

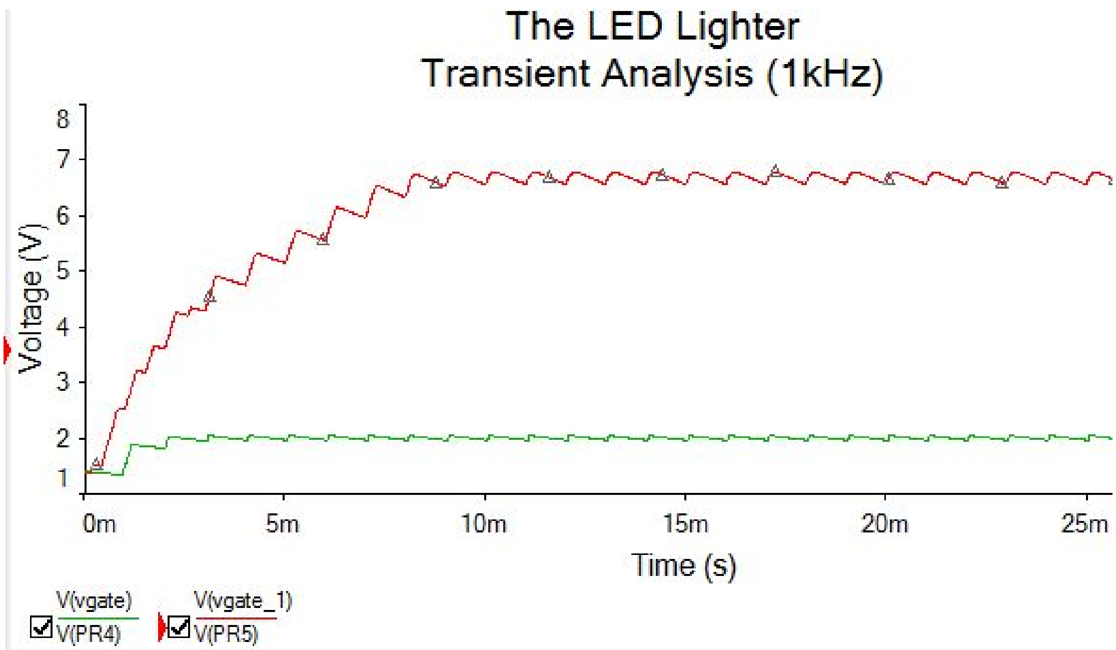
The LED Lighter
DC Operating Point Analysis

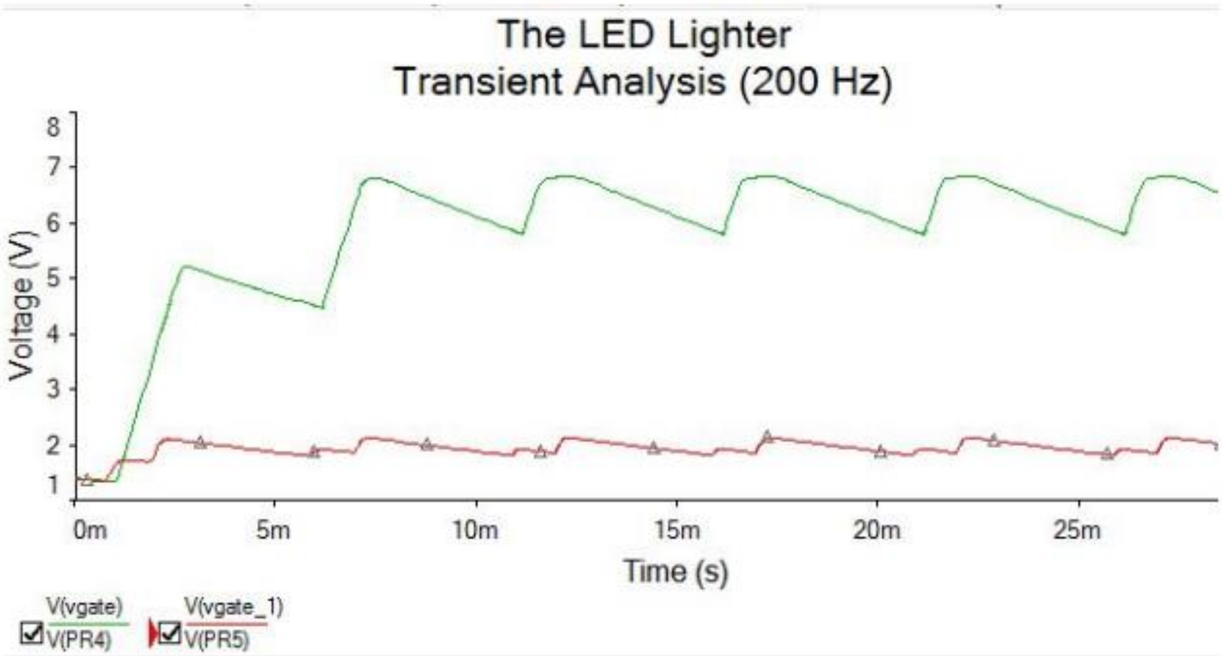
	Variable	Operating point value	
1	I(LED1:A) | I(PR1)	65.29763 n	I_drain (Low Pass)
2	I(R13:1) | I(PR10)	65.26440 n	I_source (High Pass)
3	I(Q2:G) | I(PR11)	1.37299 p	I_gate (HP)
4	I(Q1:G) | I(PR12)	1.39447 p	I_gate (HP)
5	I(LED2:A) | I(PR2)	65.29763 n	I_drain (LP)
6	I(R7:1) | I(PR9)	65.26440 n	I_source (LP)
7	V(vgate) | V(PR4)	1.39447	V_gate (Low Pass)
8	V(vgate_1) | V(PR5)	1.37299	V_gate (High Pass)

In the following AC Sweep analysis, the red line is our low pass filter and the green line is the high pass filter.

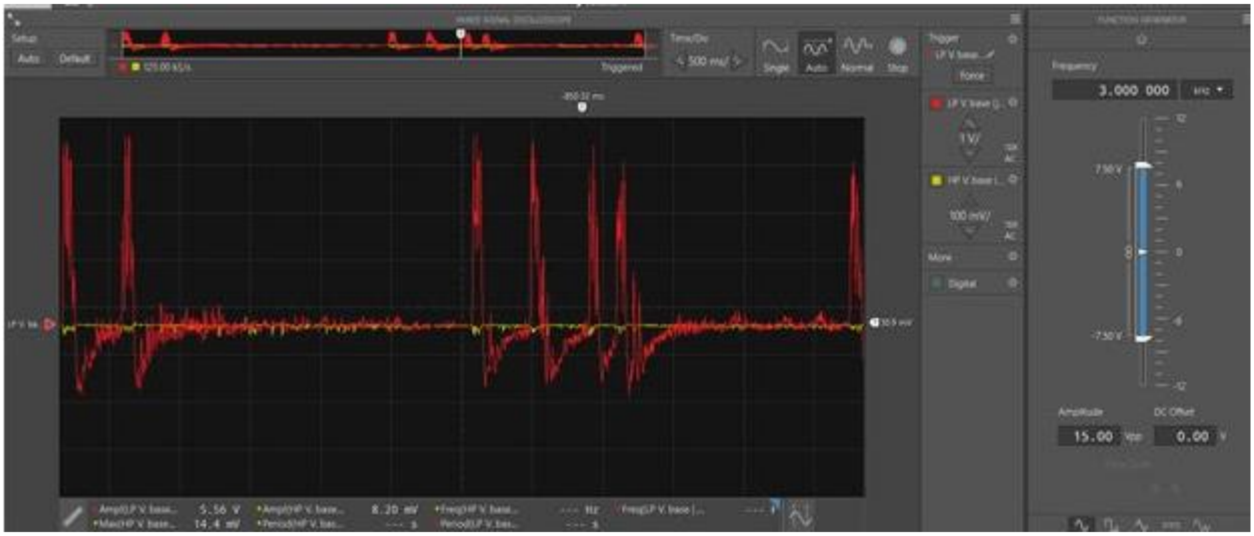


As you can see from the above, the cutoff frequency (which is approximately 3dB from the max gain) was as expected. We predicted our cutoff to be 408 Hz for the low pass and 482 Hz for the high pass. In the following transient analyses, V(PR4) corresponds to the gate voltage for the low-pass filter and V(PR5) corresponds to the gate voltage for the high-pass filter.


The LED Lighter
Transient Analysis (1kHz)
8
7
6
5
4
3
2
1
Voltage (V)
0m
5m
10m
15m
20m
25m
Time (s)
V(vgate)
V(PR4)
V(vgate_1)
V(PR5)


The LED Lighter
Transient Analysis (200 Hz)
8
7
6
5
4
3
2
1
Voltage (V)
0m
5m
10m
15m
20m
25m
Time (s)
V(vgate)
V(PR4)
V(vgate_1)
V(PR5)

Next, we have our oscilloscope readings from Virtualbench. The first screenshot is of an oscilloscope reading taken at the high pass and low pass voltage gates, where channel 1 (red) is for low pass and channel 2 (yellow) is for high pass.



This reading was done with the music playing as our input. We determined that we stopped the reading when the song had been playing low frequencies, which explains why channel 1, which is low pass, is showing more activity on the screenshot than channel 2, which is high pass. The next screenshot is of the oscilloscope reading taken after the Sallen key high pass and low pass filters. Channel 1 (red) shows the reading for the low pass and channel 2 (yellow) shows the reading for the low pass.

However, we later realized that we had taken this oscilloscope reading incorrectly because we had channel 1 set to 10x while channel 2 was set to 1x. We realized this too late to redo the oscilloscope readings, but we can still see from the graph that if we multiplied the readings by 10, we would have gotten a good result.

Conclusion

From the beginning of the project, we were put under heavy time constraints in the design and implementation in order to put out a product that would deliver our original intent. Working efficiently and doing our best to avoid any errors became our focus and our ultimate goal. Although there were definitely some minor missteps incurred throughout, we were able to employ methods of troubleshooting as well as learn from our mistakes. In the end, we were happy with the fact that our final product turned out to work as expected and there was no greater feeling than seeing our LEDs light up synchronously with the music we chose. If we could return and work on this all over again, we would definitely would emphasize more effort in the design process include the ability to work well enough when presented with any type of song. Additionally, the lighting of the LEDs does not work intuitively. By that remark, I am referring to the way the LEDs dim when nearing the cutoff frequencies. When playing the circuit, if there happens to be a lot of sound being input with frequencies near the range of the cutoff frequency, the LED appears to either be off or very dimly lit because we are varying the LED by using the non-saturated region. We believe the circuit could be improved if instead we created the circuit so that at the cutoff frequency, the LEDs would turn on very quickly to a certain level of brightness and then from there they would fluctuate in brightness. In the end, however, we believe that this project was a great testament and application of the ideas presented throughout the course.

Improvements for the Future

One thing that we noticed is that the high pass filter is not quite as strong of a filter with the cutoff frequency as the low pass filter, so in the future we would definitely consider using different resistor values to help us have a crisper cutoff frequency. Tying into this, we would like to in the future use resistors beyond those in our kit, as this would expand our options and help us build a better project. Also, we would like to run more testing with different songs, and perhaps even test with tracks that emit a constant frequency so that we can test the detectors to make even more sure that they work.