University of Washington
Department of Electrical Engineering
EE 433, Winter 2019

Laboratory Project Design #2:
Tuned AC Voltmeter

Student Name	Laboratory Contribution
Minh Ho	Construction, Testing, Troubleshooting, Lab Report
Andrew Nguyen	Lab Report, Procedure
Jack Chicken	Circuit Design, Simulation, Fabrication, Troubleshooting, Lab Report

March 15th, 2019
Professor Lawrence Lam, PhD

Table of Contents

Objective

The objective of this laboratory is to design a tuned AC voltmeter which responds only to frequencies within a certain band. Tuned voltmeters are very common instrumentation systems when significant out-of-band interference or noise must be rejected. The starting point for this design is the digital panel meter (DPM) which was constructed during Laboratory Project Design #0.

Introduction

In this lab our team is tasked with designing, simulating, and constructing an AC voltmeters. In general, AC voltmeters convert the AC signal into DC through a rectifier and display it through the meter reading output. We want to tune this voltmeter so it only reads signals within our desired frequency.

These are the design specifications for the project:

1. The entire system must operate from a single 9 Volt transistor battery or a bench power supply.

2. The output reading is displayed on a 3.5 digit LCD digital panel meter (DPM) or a bench meter.

3. The input impedance of the voltmeter must be 10.0 ± 0.1 MΩ.

4. The meter must feature four ranges, each separated by a factor of 10: 20 VACrms, 2.0 VACrms, 200 mVACrms, and 20 mVACrms. Switching between ranges must be accomplished by changing no more than 2 wires, so that this could be ultimately performed by a simple rotary switch. Changing ranges must also properly reposition the decimal point in the DPM.

5. The maximum count and accuracy of each of these ranges must be: 19.99 ± 0.20 VACrms, 1.999 ± 0.020 VACrms, 199.9 ± 2.0 mVACrms, and 19.99 ± 0.20 mVACrms. This is a basic 1.0% accuracy for each of the four ranges. Within each range, the voltmeter must display the rms value of the sinewave which is being input.

6. The desired frequency response of the meter is over a pass band of 100 Hz to 100 kHz. Within this frequency range, the response of the meter should be flat with frequency to within ±0.1 dB (within a multiplicative factor of 0.9886 to 1.0116 of the nominal value). Beyond one decade outside of this pass band (both high and low) the frequency response must have fallen by at least 50 dB. i.e., at 10 Hz and lower and at 1.0 MHz and higher the response must be attenuated by at least 50 dB (a factor of 316.2).

7. The cost of the parts to create this system must be minimized.

Equipment and Materials Used

- LM324 Quad Op-Amp
- ICL7760 Super Voltage Converter
- 1N4148 diodes

- Resistors
- Potentiometers
- Capacitors

- Power Supply
- Function Generator
- Oscilloscope

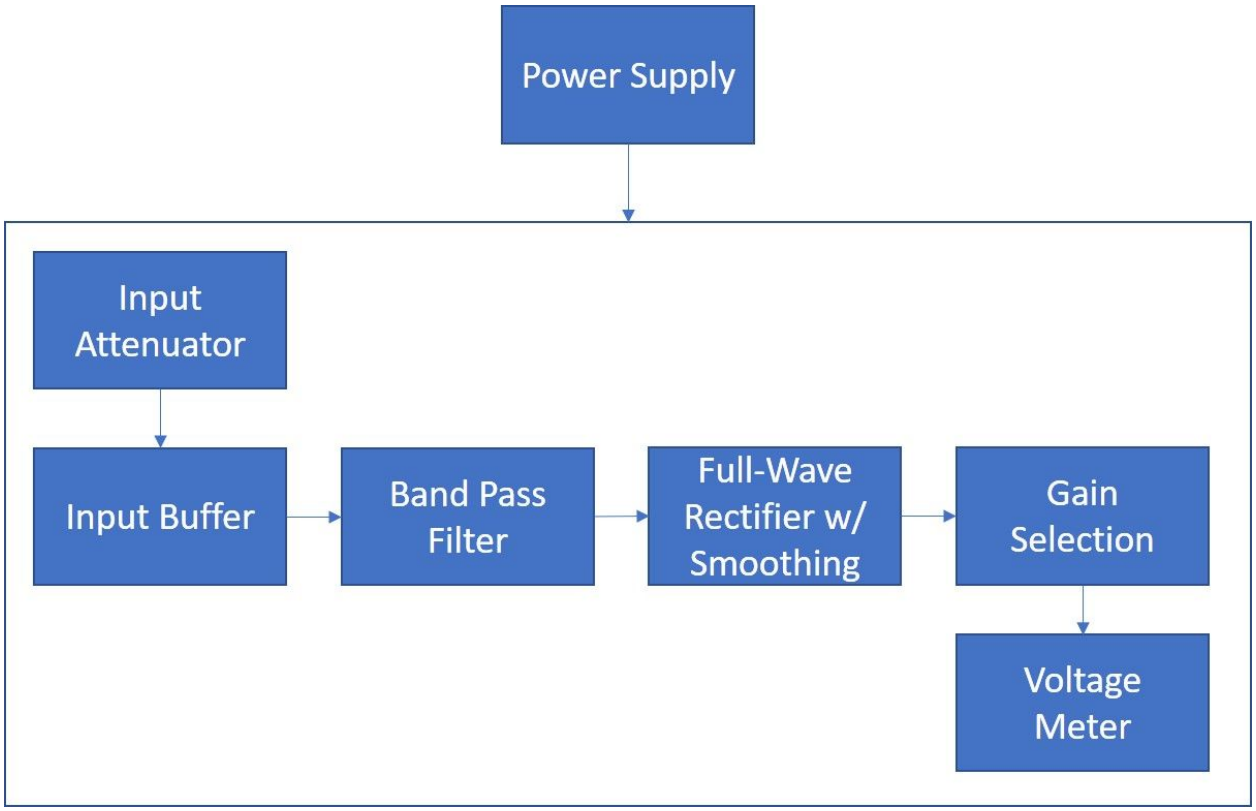
Additional components were designed, bought, and constructed by Jack explained within the lab report.

Procedures

1. Power up the overall system using the built-in voltage regulator of the ICL7106.
2. Use the +5.0 V rail to power any needed logic.
3. Use the bench power supply or the +5.0 V and -5.0 V power supply rails to power any op amps or comparators.
4. Use an input voltage divider that always remains in place, regardless of the range that the voltmeter is set to, to protect against direct exposure to the voltage level.
5. Use the input voltage divider to set the input impedance of the system to its desired value.
6. Select either the in-phase or out-of-phase backplane signal by controlling it with an exclusive-OR (XOR) gate.
7. Use the logic control signals generated by the range selection comparators to control the 2DP, 3DP, and 4DP segments of the display.
8. Use a manual-exchange scheme to switch components when you have different voltage input range.
9. Cascade multiple filter stages of the 50 dB/decade filter roll-off in order to create a higher-order filter.
10. Properly align the filter poles of each stage to produce an overall filter function that meets the specifications.

Circuit Design

To meet the requirements of the project, the circuit was broken into different stages as shown below.



Power Supply

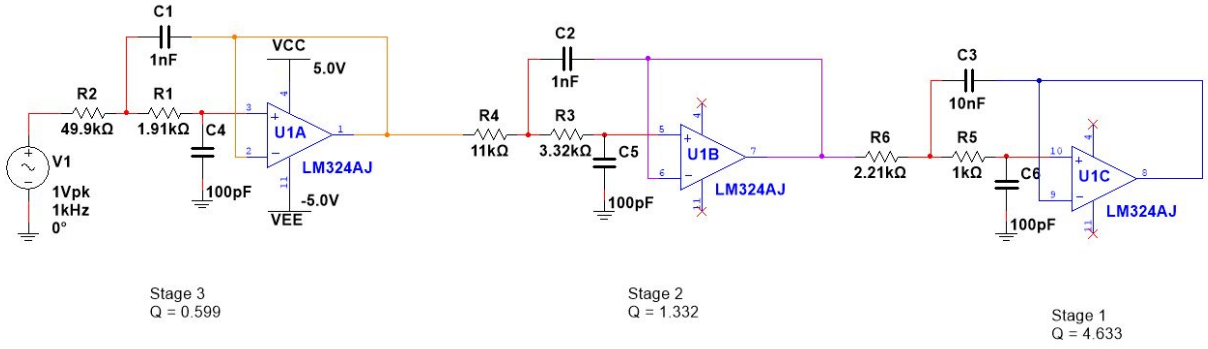
To meet the requirement for the circuit to be battery powered a system voltage of 5VDC was selected. This allowed the circuit to be powered from the USB interface of a rechargeable lithium-ion battery pack.To provide the -5VDC input for the bandpass op-amps, a voltage inverter using a 7066 IC was constructed.

Input Attenuator and Buffer

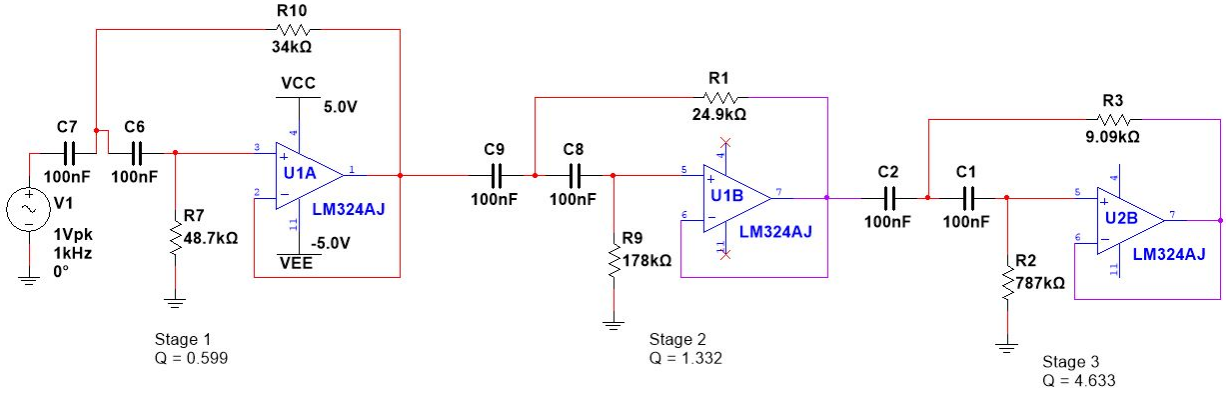
As the circuit was required to accept an input as high as 20 VACrms, a voltage divider was incorporated to step the voltage down by a factor of 10. To also meet the requirement for a 10M ohm input impedance, the voltage divider was comprised of a 9M and 1M ohm resistor. To ensure the input signal was not affected by the operation of the AC voltmeter, a voltage follower was placed following the attenuator to buffer the input signal.

Band Pass Filter

The band pass filter is the heart of the circuit. To aid in development and troubleshooting, the band pass filter was divided into two parts: a Low Pass filter and a High Pass filter. To meet the requirement for a minimum of 50 dB attenuation one decade on either side of the desired corner frequencies, sixth-order Chebyshev filters were selected. To aid in the design of the filters and allow rapid alterations to the design, a Matlab script was written to calculate the resistor and capacitor values for a given corner frequency and filter type, using Unity-Gain op-amp active filter stages. The Matlab script also returned the standard 1% tolerance resistor values to aid in component procurement. To allow the filter performance to be simulated and evaluated, the Low Pass and High Pass filters were modeled in Multisim, as shown below.



6-Order Chebyshev Low Pass Filter (100 kHz Cutoff Frequency)

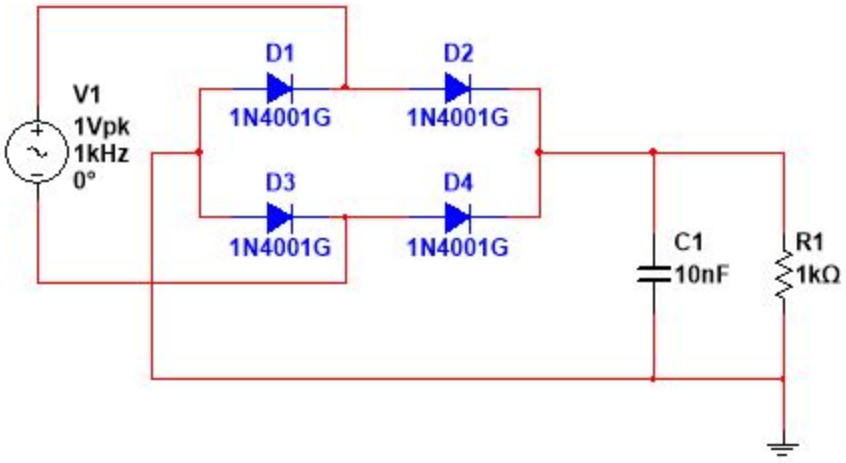


6-Order Chebyshev High Pass Filter (100 Hz Cutoff Frequency)

To simplify circuit construction the LM324 Quad Op-Amp was selected for U1 and U2. Since only three op amps are needed for each filter, one of the remaining op amps was used for the input buffer stage.

Full Wave Rectifier

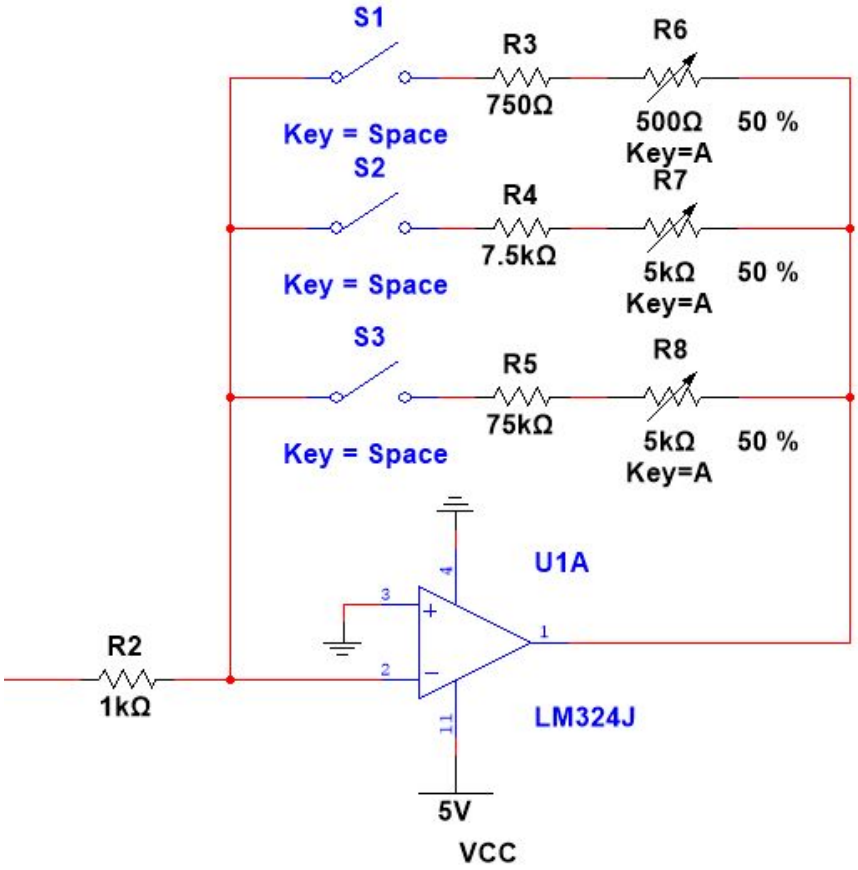
To convert the AC output of the Band Pass filter to a DC signal for display by the voltmeter, a full wave rectifier was used. A schematic of the rectifier as modeled in Multisim is shown below.



Note that this model does not represent the as-built rectifier--1N4148 diodes were used, and a much larger capacitor, 470 uF, was used. Also an LED was placed in series with the resistor. The fabricated rectifier did not perform as expected--the large capacitor, while providing improved smoothing in steady-state conditions, meant that the output was slow to respond to input changes. Also, the input to the rectifier could smaller than the forward bias of the LED, meaning that the LED would not illuminate. If an indicator light was desired, a transistor could be used to amplify the output signal to drive the LED.

Gain Selection

Following rectification, the signal needs to be adjusted to account for the input attenuation and losses through the circuit. This is accomplished through a selectable gain amplifier. A schematic of the amplifier as modeled in Multisim is shown below.



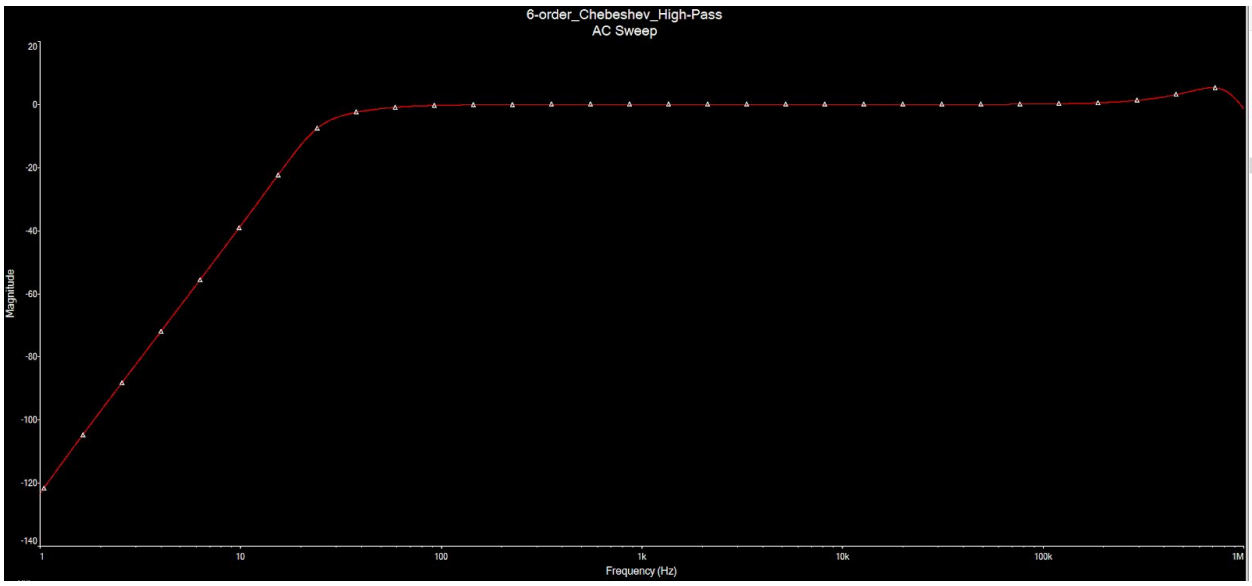
The switch S1 selects the nominal unity gain, the switch S2 selects a gain of 10, and the switch S3 selects a gain of 100. The variable resistor is used to tune the gain to the desired output. A rotary switch was used to select the gains in the as-built circuit, since only one switch should be closed at a time.

Voltage Meter

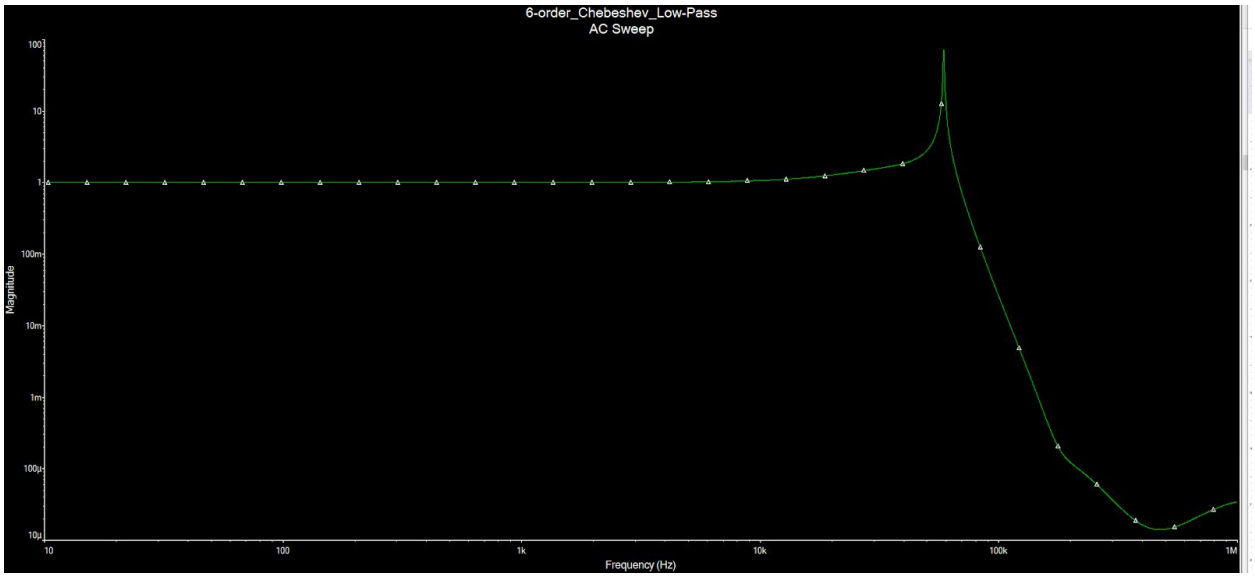
A voltmeter powered by a 5VDC supply was built from a kit to use in the circuit, since the meters constructed for Project 0 required a 9 V input. The connection points for the control of decimal point location were configured to be controlled by the rotary switch as well, so the gain and decimal point display could be coordinated.

Simulation

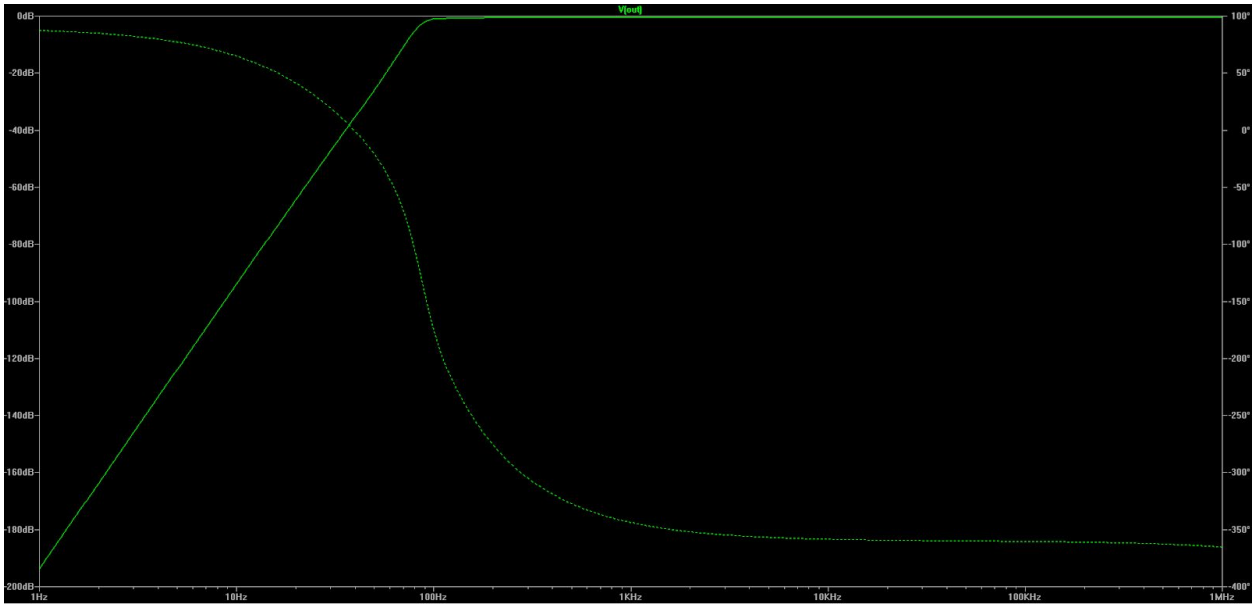
Multisim and LT Spice were used to simulate the behaviour of the High and Low Pass Chebyshev filters. Plots of the simulation results are shown below.



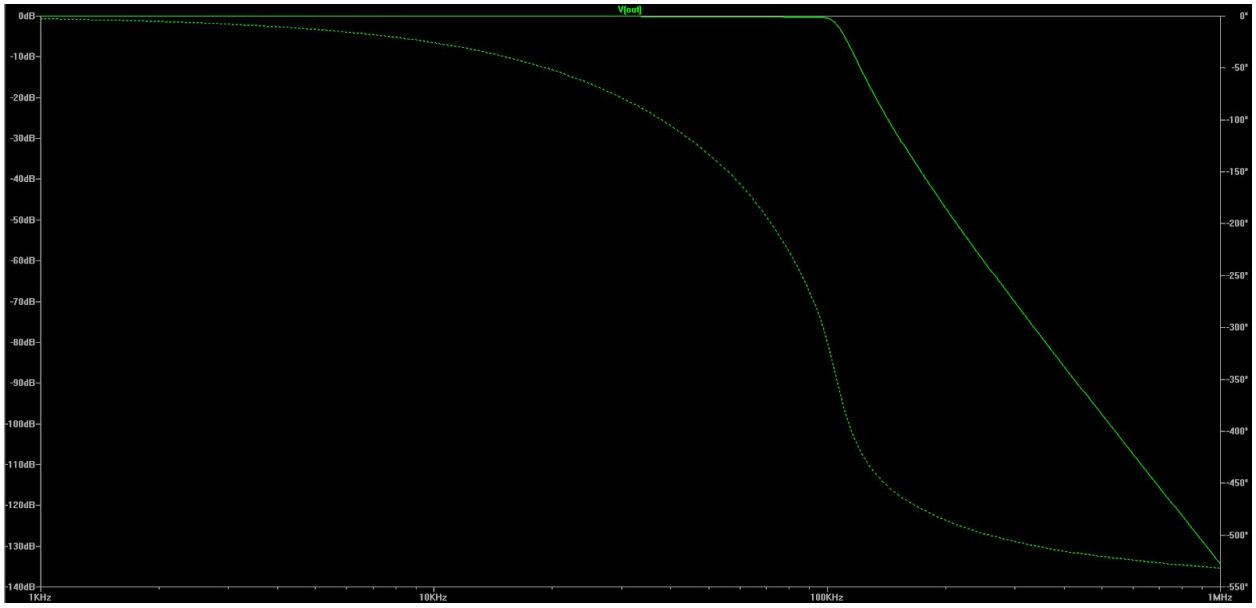
Multisim High Pass FIlter Simulation Results



Multisim Low Pass Filter Simulation Results



LTSpice High Pass Filter Simulation Results



LTSpice Low Pass Filter Simulation Results

Since up to 18 dB of attenuation could be expected at the design corner frequencies due to the 6-order filters, the corner frequencies were shifted slightly to the outside of the specified frequencies to ensure that the attenuation would be zero (or near zero) at the specified corner frequencies, but still give the desired 50 dB attenuation one decade outside the passband. The Matlab script was very useful at calculating new resistor and capacitor values for different design frequencies until the right component values were determined.

It was noted that for the High Pass Filter, the Multisim and LTSpice simulation produced similar plots that matched the expected results. However, for the Low Pass filter, while the LTSpice simulation produced a plot similar to expected results, the Multisim simulation did not. The Multisim result appeared to more closely resemble the output of the final stage in isolation. It is unknown why the two simulation results differed.

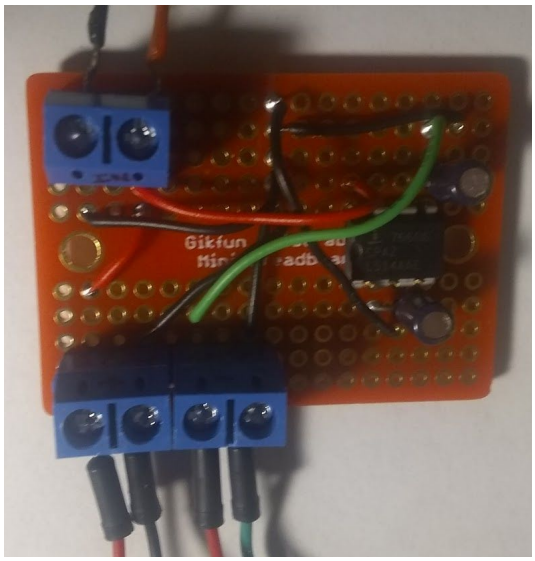
Circuit Construction

As discussed in previous sections, the circuit was divided into several stages, and in general each stage was built as a discrete subcircuit to be integrated into the overall AC Voltmeter system. This was done to simplify construction and troubleshooting. Once a successful design was validated and tested, the circuit could be condensed into a smaller footprint.

Where possible it was desired to build the circuit by soldering components to a solderable breadboard, rather than using a solderless breadboard to improve the prototype circuit's robustness and reduce the potential for the introduction of noise in the circuit. A description of each stage is provided below.

Stage 0: Power Supply

The since the circuit required a +/- VDC power supply for the op-amp filter stages, a 7660S IC was selected to convert a 5 VDC input to -4VDC. This voltage level also allowed the use of a Lithium-Ion Rechargeable battery pack with a USB interface.

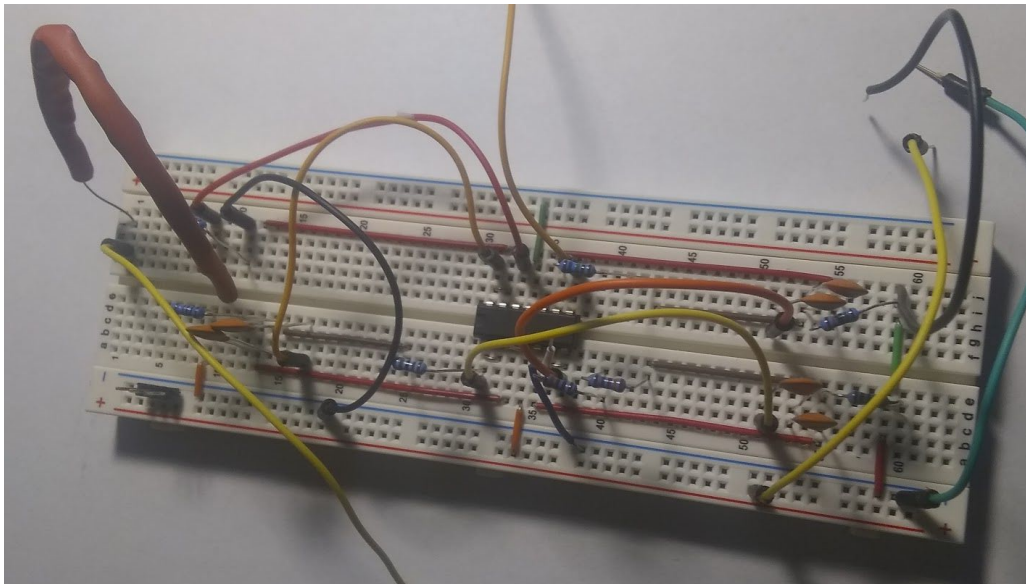


Stage 1: Input Attenuation and Buffering and High-Pass Filter

Since a 1:10 input attenuation was desired as well as a 10M ohm input impedance, a 9M/1M voltage divider was selected. Since the lab supplies did not have any 9M ohm resistors, 9 1M resistors were soldered together in series, and subsequently coated with hot glue and covered with heat shrink tubing for robustness. The 9M resistor assembly can be seen in the upper left of the picture below.

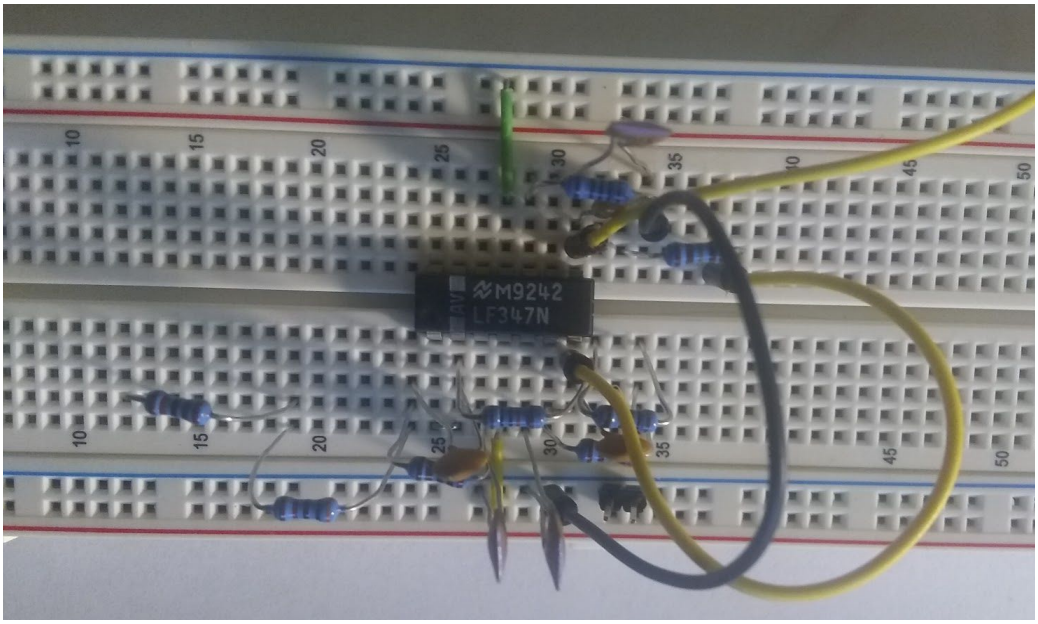
A LM324 Quad Op-Amp was selected for this stage. The fourth op-amp was used for the input buffer, visible in the upper left quadrant of the breadboard below.

The circuit out on the breadboard to aid in troubleshooting, although the jumper wires offered the potential for noise coupling. This could have been reduced by using straight jumper wires on the surface of the breadboard, or making the circuit more compact.



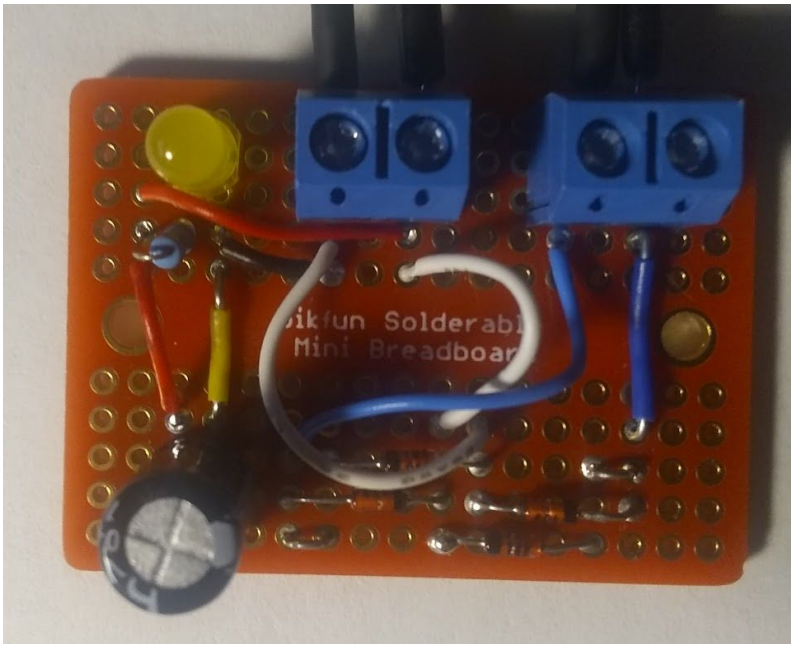
Stage 2: Low Pass Filter

The Low Pass Filter was constructed on a single breadboard. Initially an LM324 quad op-amp was selected, but this was eventually changed to a LF347N quad op-amp because the original op-amp appeared to be defective and did not perform as expected.



Stage 3: Rectification

A full-wave bridge rectifier was constructed on a solderless breadboard. As discussed in the design section, the configuration of the rectifier was not appropriate for the intended use, and while the output was a DC signal, the time constant was too large to be useful.

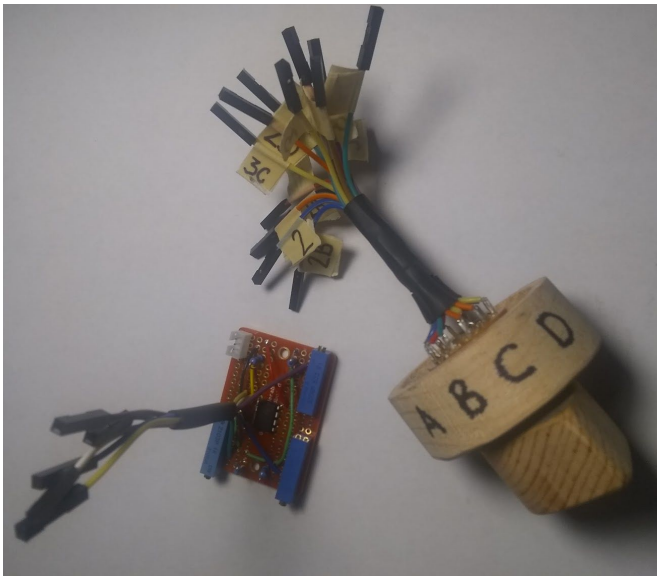


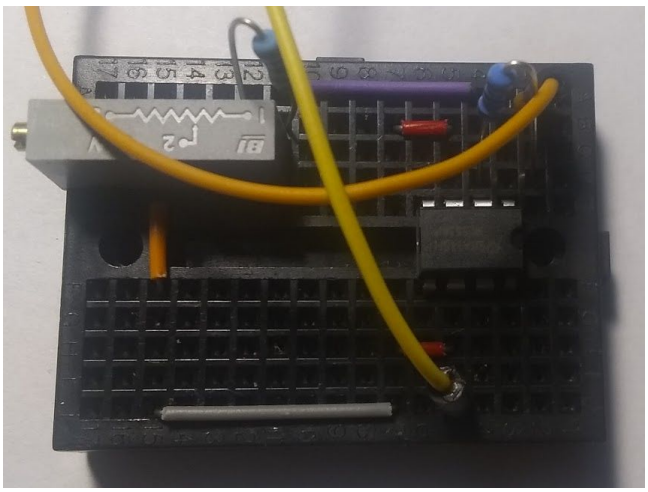
Stage 4: DC Gain Selection

A Texas Instruments NE5532P op amp was used in a 1/10/100 selectable gain inverting amplifier circuit. Three potentiometers were wired together in series with fixed value resistors to allow an range of gain values to adjust the final output to the display. Unfortunately, the op amp appeared to be defective, and did not provide the expected amplification.

In addition, a 4-throw, 3-pole rotary switch was prepared to allow selection of the gian values as well as the position of the decimal point on the voltmeter display. The switch was mounted in a simple wood housing to allow easy rotation of the switch.

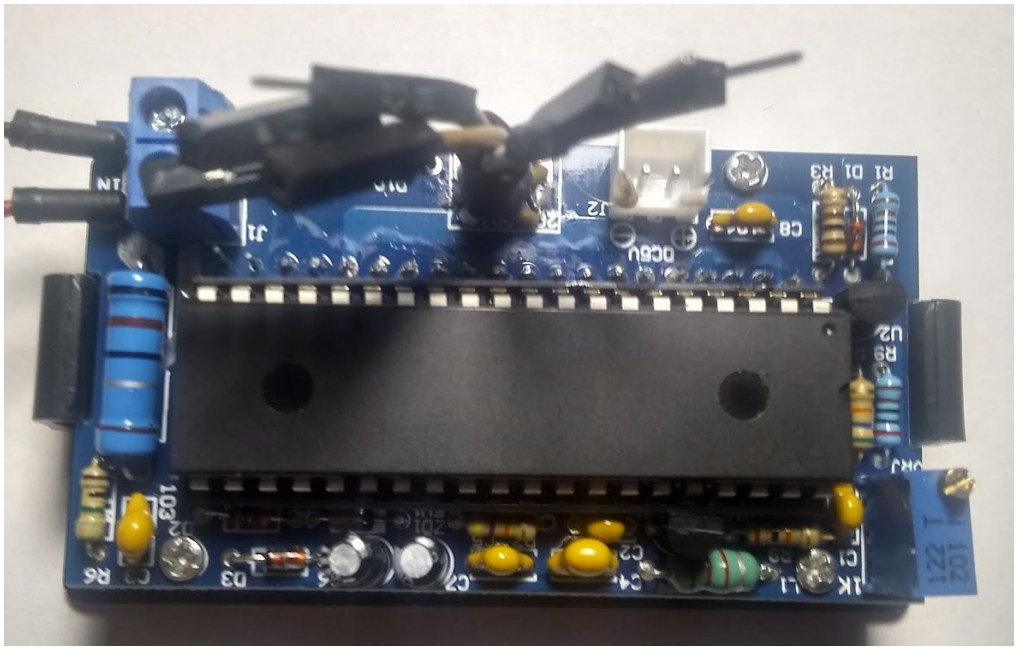
A second adjustable amplifier with a middle gain of 1 was prepared on a mini solderless breadboard for use with the circuit.





Stage 5: Voltmeter and Display

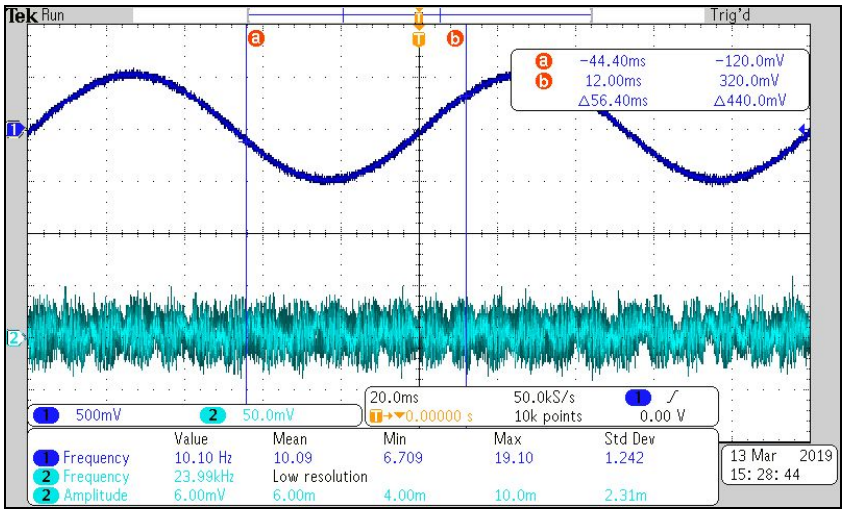
A 3.5-digit voltmeter was constructed from a kit purchased online. Like the voltmeter constructed in Lab 0 this voltmeter incorporates the ICL7107 IC, but is a more compact design and powered by 5VDC, instead of a 9V battery.



Circuit Verification

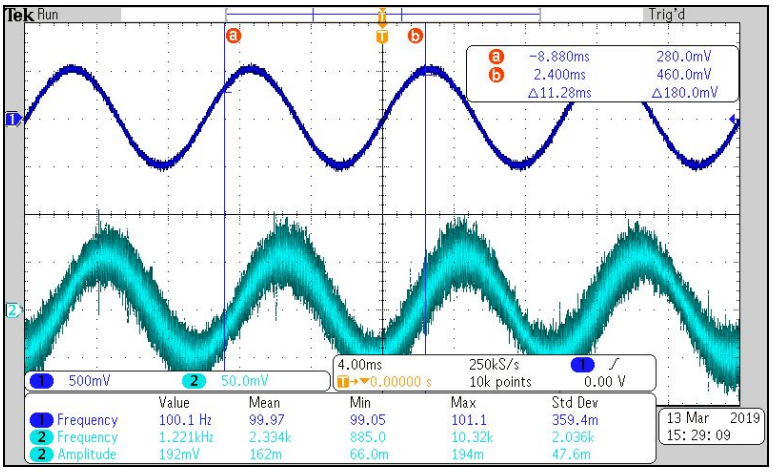
We use the lab oscilloscope, function generator, and power supply to demonstrate our circuit.

- The quad op-amp is powered by a single +/- 5V.
- An oscilloscope measures:
    - Line 1: Input signal
    - Line 2: Output signal
- A function generator to demonstrate that our bandpass signal allows signals 100Hz - 100kHz as per our design specifications. Amplitude is set to 1 V_pp



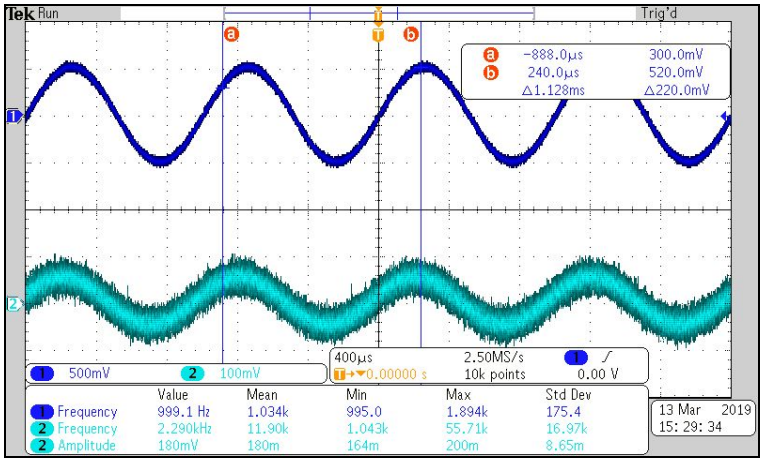
10 Hz input frequency

From the 10 Hz input frequency we see that there is attenuation of the signal by the high pass portion of our bandpass filter. The signal frequency is below our designed bounds.



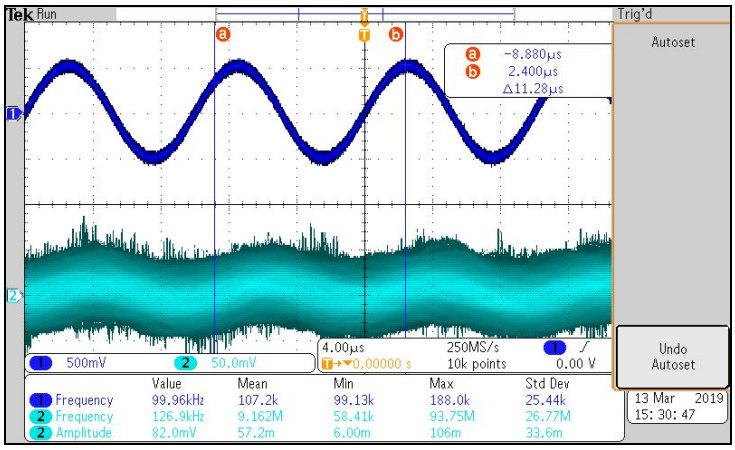
100 Hz input frequency

From the 100 Hz input frequency we can see that the signal is not attenuating as it is within our band pass's designed range. There is still some roll off noise as it is on the edge of our lower bound.



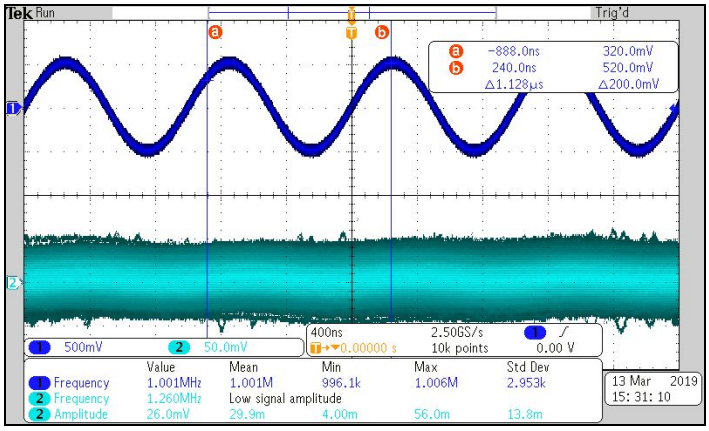
1 kHz input frequency

From the 1 kHz input frequency we can see that the signal is not attenuating as it is within our band pass's designed range. The signal is clean.



100 kHz input frequency

From the 100 kHz input frequency we can see that the signal is starting to attenuate due to roll off as it is within our band pass's designed upper bound.



1 MHz input frequency

From the 1 MHz input frequency we see that there is attenuation of the signal by the low pass portion of our bandpass filter. The signal frequency is above our designed bounds.

Conclusion

From this project our team designed tuned AC voltmeter that filters out noise interference outside of certain frequencies. The project helped us to understand the material we were learning in class by having us design our filters within a certain band pass and attenuation rate. Other design specifications within the project emulated real world parameters that would need to be considered.

While our simulations met the targets we designed for, our final circuit that we designed, simulated, and constructed was only able to meet 5 of the 9 criteria listed in the grading rubric. Due to time constraints we were unable to rectify and select the gain for the output. Our rectifier was completed, but we believe issues arose due to the capacitor we used being too large.

The quad op-amps we were using for our gain selector were faulty and created too much noise. This issue also affected our low-pass filter which we built separately from our high-pass on a different bread board. The high-pass filter worked, while the low-pass had too much noise within our band-pass bounds. We later realized multiple of the op-amps in the batch we ordered were faulty except for the one in our high-pass filter. We were able to get it working by using a different model quad op-amp from another group. Since we built and tested each component at the same time, a lot of time was spent trying to troubleshoot each component.

We believe given a bit more time, we would’ve been able to complete all portions of the circuit to match our design and simulations. We used our power supply we built to power our supply rails. The frequency range and roll off for our band-pass met the design specifications. Overall the lab experience gave us valuable experience with design, simulation, construction, and troubleshooting for project design.

References

[LM324 Quad Op Amp Datasheet]

[ICL7760 Super Voltage Converter Datasheet]

Appendix A

As mentioned above, a Matlab script was developed to calculate component values for Chebyshev filters using unity gain KRC second order op amp stages. The complete script and associated functions are provided below.

```
% Chebyshev Filter Design
% Author: Jack Chicken
% Date: March 5, 2019

clc
clear all
close all


% Define filter order
order = 6;

% Define cut-off frequency,Hz
fc = 100;

% Define filter type (1 = 'high' or 2 = 'low')
type = 1;

% Define Reference Capacitor Value (F)
C2 = 100e-9;


% Define ripple (1 = 0.1-dB, 2 = 1.0-dB)
ripple = 1;

% 0.1-dB ripple Chebyshev low-pass filter
Table_1 =...
    [1.820,0.767,0.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000;...
    1.300,1.341,0.969,0.000,0.000,0.000,0.000,0.000,0.000,0.000;...
    1.153,2.183,0.789,0.619,0.000,0.000,0.000,0.000,0.000,0.000;...
    1.093,3.282,0.797,0.915,0.539,0.000,0.000,0.000,0.000,0.000;...
    1.063,4.633,0.834,1.332,0.513,0.599,0.000,0.000,0.000,0.000;...
    1.045,6.233,0.868,1.847,0.575,0.846,0.377,0.000,0.000,0.000;...
    1.034,8.082,0.894,2.453,0.645,1.183,0.382,0.593,0.000,0.000;...
    1.027,10.178,.913,3.145,0.705,1.585,0.449,0.822,0.290,0.000;...
    1.022,12.522,.928,3.921,0.754,2.044,0.524,1.127,0.304,0.590];

% 1-dB ripple Chebyshev low-pass filter
Table_2 =...
```

```
    [1.020,0.957,0.000,0.000,0.000,0.000,0.000,0.000,0.000,0.000;...
    0.997,2.018,0.494,0.000,0.000,0.000,0.000,0.000,0.000,0.000;...
    0.993,3.559,0.529,0.785,0.000,0.000,0.000,0.000,0.000,0.000;...
    0.994,5.556,0.655,1.399,0.289,0.000,0.000,0.000,0.000,0.000;...
    0.995,8.004,0.747,2.198,0.353,0.761,0.000,0.000,0.000,0.000;...
    0.996,10.899,.808,3.156,0.480,1.297,0.205,0.000,0.000,0.000;...
    0.997,14.240,.851,4.266,0.584,1.956,0.265,0.753,0.000,0.000;...
    0.998,18.029,.881,5.527,0.662,2.713,0.377,1.260,0.159,0.000;...
    0.998,22.263,.902,6.937,0.721,3.561,0.476,1.864,0.212,0.749];

if ripple == 1
    f_Q_vector = Table_1((order-1),:);
elseif ripple == 2
    f_Q_vector = Table_2((order-1),:);
end

qty_2_order_stages = floor(order/2);

if qty_2_order_stages < (order/2)
    component_values = zeros(qty_2_order_stages+1,4);
else
    component_values = zeros(qty_2_order_stages,4);
end

%--------------------------------------------------------------------------
% Low Pass Filter Values
%--------------------------------------------------------------------------
if type == 2;

    ans = sprintf('Component values for %d-order low-pass Chebeyshev filter
with %.2f kHz cut-off frequency:\n',order, fc/1e3);
    disp(ans)

    % Second-Order Stages
    for ii = 1:(qty_2_order_stages)

        jj = ii*2-1;
        f0 = fc*f_Q_vector(jj);
        Q = f_Q_vector(jj+1);

        % Unity-Gain KRC Circuit
        w0 = f0*2*pi();
        n_pre = 4*Q^2;
        n=10^ceil(log10(n_pre));
        C1=n*C2;
        k = n/(2*Q^2)-1;
        m = k+sqrt(k^2-1);
        R2_pre=1/(sqrt(m*n)*w0*C2);
```

```
        R1_pre=m*R2_pre;
        R1 = R_Value(R1_pre);
        R2 = R_Value(R2_pre);
        stage_component_values = [C1, C2, R1, R2];
        component_values(ii,:) = stage_component_values;

        ans = sprintf('Stage %d: KRC Unity-Gain Low-Pass Filter, Q =
%.3f.\n',ii,Q);
        disp(ans)
        ans = sprintf('   Value of Capacitor C1 (nF): %.2f', C1/1e-9);
        disp(ans)
        ans = sprintf('   Value of Capacitor C2 (nF): %.2f', C2/1e-9);
        disp(ans)
        ans = sprintf('   Value of Resistor R1 (kOhms): %.2f', R1/1e3);
        disp(ans)
        ans = sprintf('   Value of Resistor R2 (kOhms): %.2f\n', R2/1e3);
        disp(ans)
    end

    % First-Order Stage (if n = odd)
    if qty_2_order_stages < (order/2)
        f0 = fc*f_Q_vector(qty_2_order_stages*2+1);
        R1_pre = 1/(f0*2*pi()*C2);
        R1 = R_Value(R1_pre);
        R2 = R1;
        C1 = 0;
        stage_component_values = [C1, C2, R1, R2];
        component_values(qty_2_order_stages+1,:) = stage_component_values;

        ans = sprintf('Stage %d: Single Order Unity Gain Low-Pass Filter, Q =
%.3f.\n',...
            ii+1,Q);
        disp(ans)
        ans = sprintf('   Value of Capacitor C2 (nF): %.2f', C2/1e-9);
        disp(ans)
        ans = sprintf('   Value of Resistor R1 (k Ohms): %.2f', R1/1e3);
        disp(ans)
        ans = sprintf('   Value of Resistor R2 (k Ohms): %.2f\n', R2/1e3);
        disp(ans)
    end

end

%------------------------------------------------------------------------
% High Pass Filter Values
%------------------------------------------------------------------------
if type == 1
```

```
    ans = sprintf('Component values for %d-order high-pass Chebeyshev filter
with %.2f kHz cut-off frequency:\n',order, fc/1e3);
    disp(ans)

    % Second-Order Stages
    for ii = 1:(qty_2_order_stages)

        jj = ii*2-1;
        f0 = fc/f_Q_vector(jj);
        Q = f_Q_vector(jj+1);

        % Unity-Gain KRC Circuit
        C1=C2;
        w0 = f0*2*pi();
        R1R2_a=(w0^2*C1*C2)^-1;
        R1R2_b=(2*Q)^2;
        R2_pre=sqrt(R1R2_a*R1R2_b);
        R1_pre=R2_pre/R1R2_b;
        R2 = R_Value(R2_pre);
        R1 = R_Value(R1_pre);
        stage_component_values = [C1, C2, R1, R2];
        component_values(ii,:) = stage_component_values;

        ans = sprintf('Stage %d: KRC Unity-Gain High-Pass Filter, Q =
%.3f.\n',ii,Q);
        disp(ans)
        ans = sprintf('   Value of Capacitor C1 (nF): %.2f', C1/1e-9);
        disp(ans)
        ans = sprintf('   Value of Capacitor C2 (nF): %.2f', C2/1e-9);
        disp(ans)
        ans = sprintf('   Value of Resistor R1 (kOhms): %.2f', R1/1e3);
        disp(ans)
        ans = sprintf('   Value of Resistor R2 (kOhms): %.2f\n', R2/1e3);
        disp(ans)
    end

    % First-Order Stage (if n = odd)
    if qty_2_order_stages < (order/2)
        f0 = fc/f_Q_vector(qty_2_order_stages*2+1);
        R1_pre = 1/(f0*2*pi()*C2);
        R1 = R_Value(R1_pre);
        R2 = R1;
        C1 = 0;
        stage_component_values = [C1, C2, R1, R2];
        component_values(qty_2_order_stages+1,:) = stage_component_values;

        ans = sprintf('Stage %d: Single Order Unity Gain High-Pass Filter, Q =
%.3f.\n',...
```

```
            ii+1,Q);
        disp(ans)
        ans = sprintf('   Value of Capacitor C2 (nF): %.2f', C2/1e-9);
        disp(ans)
        ans = sprintf('   Value of Resistor R1 (k Ohms): %.2f', R1/1e3);
        disp(ans)
        ans = sprintf('   Value of Resistor R2 (k Ohms): %.2f\n', R2/1e3);
        disp(ans)
    end
end

function resistor = R_Value(resistance)
%UNTITLED3 Finds the nearest standard resistor value given a resistance
% and % tolerance
%   Detailed explanation goes here

one_percent = [10.0, 10.2, 10.5, 10.7, 11.0, 11.3, 11.5, 11.8, 12.1,...
    12.4, 12.7, 13.0, 13.3, 13.7, 14.0, 14.3, 14.7, 15.0, 15.4, 15.8,...
    16.2, 16.5, 16.9, 17.4, 17.8, 18.2, 18.7, 19.1, 19.6, 20.0, 20.5,...
    21.0, 21.5, 22.1, 22.6, 23.2, 23.7, 24.3, 24.9, 25.5, 26.1, 26.7,...
    27.4, 28.0, 28.7, 29.4, 30.1, 30.9, 31.6, 32.4, 33.2, 34.0, 34.8,...
    35.7, 36.5, 37.4, 38.3, 39.2, 40.2, 41.2, 42.2, 43.2, 44.2, 45.3,...
    46.4, 47.5, 48.7, 49.9, 51.1, 52.3, 53.6, 54.9, 56.2, 57.6, 59.0,...
    60.4, 61.9, 63.4, 64.9, 66.5, 68.1, 69.8, 71.5, 73.2, 75.0, 76.8,...
    78.7, 80.6, 82.5, 84.5, 86.6, 88.7, 90.9, 93.1, 95.3, 97.6].';

scaling_factor_raw = log10(resistance);

if scaling_factor_raw >=1
    N = floor(scaling_factor_raw)-1;
else
    N = 0.1;
end

resistance_scaled = resistance/10^N;

resistance_rounded = round(resistance_scaled, 3,'significant');

resistance_rounded_vector = ones(length(one_percent),1)*resistance_rounded;

resistance_rounded_difference = (abs(one_percent-resistance_rounded_vector));

[min_value, resistor_index] = min(resistance_rounded_difference);

resistance_table_value = one_percent(resistor_index);

resistor = resistance_table_value * 10^N;
end
```