OLLSCOIL TEICNEOLAÍOCHTA BHAILE ÁTHA CLIATH

DUBLIN

TECHNOLOGICAL UNIVERSITY DUBLIN

Logbook

Guitar effects pedal (analogue)

by

Sajjad Ullah, C17344483

This Report is submitted in partial fulfilment of the requirements of the Honours Degree in Computer & Communications Engineering (DT021A) of the Technological University Dublin

May 23rd, 2022

Supervisor: Kevin Chubb

Contents

Table of Figures

Introduction & Objectives

This project is about building an analogue guitar pedal, this is a device controlled by the guitarist's feet to change the sound of the guitar. In simple terms, a guitar pedal takes in the guitar signal as an input, applies a specified effect to the signal and this becomes the output.

This will require the research of various guitar pedal effects and how they change a guitar's output signal. This research will inform the choice of pedal to implement. Considerations include limited available components and project time window. The project will make use of simulation software such as Cadence PSpice software for circuit design and use signal processing software during design validation before building the pedal effect.

Figure 1 shows an example of a guitar effects pedal called the "VOX V847 Pedal", this is a Wah-Wah pedal and is an example of a typical guitar pedal.

Figure 1 VOX V847 Wah-Wah Pedal [https://voxamps.com/product/v847-wah-pedal/]

Project objective

The project aims to create one analogue guitar pedal. Once built, the aim was to plug a ukulele into the device and have a speaker connected at the output to hear to the effect in action.

A high-level view of various guitar pedal effects will be investigated. This includes an investigation into how they modify a signal in the time and frequency domains. One pedal effect will then be chosen for implementation. This includes more in-depth research later into the chosen effect along with practical considerations that would need to be considered during its implementation. The parameters of the ukulele signal will be investigated, and component values are to be calculated to work within a designed specification. Simulation will be done on PSpice, the chosen pedal circuit then constructed on a breadboard for use with a ukulele and speaker.

Difficulties that could be encountered in creating the analogue guitar pedal are:

- Due to the finite selection of electronic components available on campus, ruling out some pedal designs.
- Other difficulties encountered were electrical noise,
- limitations of components themselves such as operating voltage & tolerances and component failures.

Week 1 starting 25/01/22: Background research on guitar pedal types

Objective: To get an understanding of the different guitar pedals and create objectives and general goals for the project. Start Preliminary Report also.

To understand the scope of the project, which is aiming to create a guitar pedal we started by compiling a list of the different effects that a guitar pedal can create.

A selection of guitar pedal effect:

Table 1 List of guitar pedal effects

• Distortion • Overdrive • Fuzz • Delay • Reverb • Wah • Looper	• Pitch Shifter • Compressor • Noise gate • Tuner • Harmonizer • Sustain • Talk-Box	• Phaser • Flanger • Octave • Tremolo • Acoustic • Chorus • Equalizer • Wah-Wah

Guitar pedals use filters to produce a desired effect, we now compile a selection of pedal effects into a mind map broken down by filter effect employed[2].

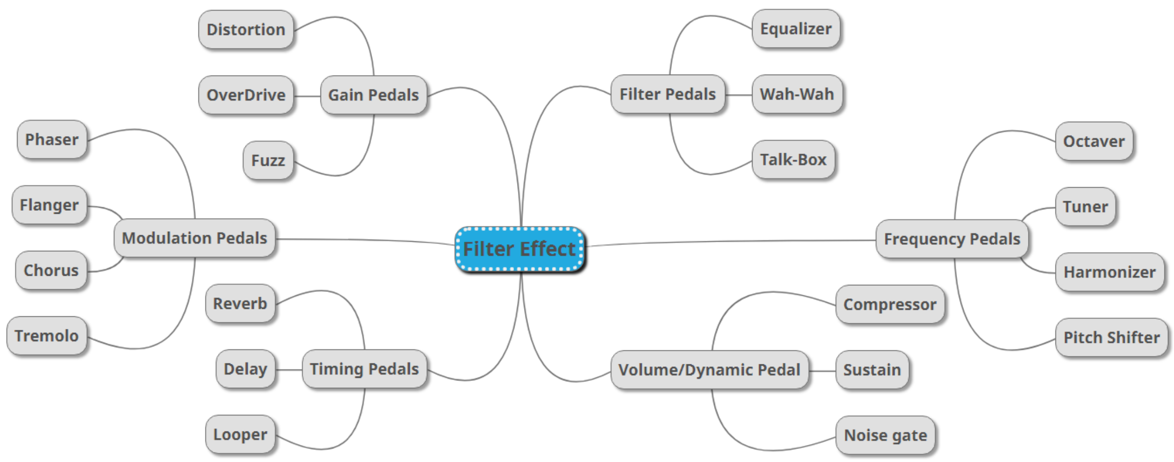


Figure 2 Mind map of filter effects with pedal types

Figure 2 contains a selection of filter effects and names of several pedals that implement the effect.

Pedal Types Explained

Filter Pedals

Equalizer: An equalizer (EQ) is a filter that can isolate a range of frequencies for amplification, attenuation or allow though unchanged[3].

Wah-Wah: Inside this is a device called a potentiometer which is a variable resistor. As the rocker plate in a wah-wah pedal moves the potentiometer will enable to a degree either a low-pass (low frequencies pass through and higher frequencies are blocked) or a high-pass filter (high frequencies pass through and low frequencies are blocked). Rocking the pedal back and forth will create the "Wah effect" [4].

Frequency Pedals

Octaver: The octave effect is where the input guitar signal is mixed with a synthesised signal that is one or more octave lower or higher than the original. An Octave occurs in music when the higher note has a frequency that is twice that of its lower note [5].

Pitch Shifter: As the name suggests it, works in real time to shift the frequencies of the input signal. They can be set to only output the shifted signal or a combination of the shifted and original signal [6].

Volume/Dynamic Pedals

Compressor: These are used to reduce the dynamic range (the ratio of the loudest to the quietest discernible sound) of a signal by reducing the amplitude of the input signal by a certain ratio if it reaches beyond a set threshold [7] .

Noise Gate: This is a common pedal used to attenuate a signal if it drops below a set threshold. It is typically used to prevent unwanted noise from forming and getting amplified by the amplifier [8].

Timing Pedals

Reverb: This pedal simulates the effect caused by sound reflection. Reverb decays the signal as opposed to an echo pedal which delays the signal [9].

Looper: This type of pedal records a time interval of the audio and plays it back in a loop. The pedal itself contains controls to start and stop the recording [10].

Modulation Pedals

Phasor: This pedal creates a copy of the signal; this copy is phase shifted and mixed back in with the original signal for the output [11].

Flanger: This pedal is similar to the Phasor; it also creates a copy of the signal. The copied signal is played at slower speed by modulating the delay time of the copied signal using a low frequency oscillator (LFO). This creates a "whooshing" sound effect due to two recordings playing at the same time [12].

Gain Pedals

Overdrive: when the signal voltage level reaches the max limit of what the hardware can output. This leads to the signal getting truncated and some frequencies getting cut off. When this happens, it's termed "clipping", and this is an overdrive pedal. It causes the signals to clip on purpose [13].

Distortion: Similar to the overdrive pedal except they use multiple stages for more clipping of the signal. The more stages the higher distortion gain [13].

Setting project objectives

Below is a set of initial project objectives that will need to be achieved in order for the project to be considered accomplished.

Objectives of the project:

- Investigate common guitar effects pedals.
- Breakdown pedal effects into categories and identify a suitable effect to implement.
- Thoroughly investigate the underlying mechanism which produces the chosen effect.
- Determine system input and output parameters
- Derive design equations for the filter.
- Use simulation software to validate the design parameters.
- Analyse filter e.g. frequency and phase responses, poles & zeros via MATLAB & PSpice.
- Implement the software-validated design on a breadboard for hardware verification.
- Build onto a prototyping board, attach audio jacks for input and output connections.

Reflection on the week

A better understanding of the scope of the project had been gained, there was discussion on the window of time available to complete the project with the supervisor. The objectives and what general goals would be of the project have become clear.

Minutes from meeting

During the first meeting with the supervisor there was discussion on guitar pedals, viewing images of them and types of effect the pedals are able to create. Supervisor also presented a ukulele during the meeting which we decided it would be a good test device to connect to the pedal once built for testing and demonstration. Filling out the project safety doc was priority during this meeting.

Week 2: Research into Wah-Wah effect

Objective: To continue research on different guitar pedals with the aim of choosing one effect to implement. Also continue work on Preliminary Report.

This week in order to decide on an effect to implement we went on sites such as YouTube to listen to guitarists playing with a pedal effect connected.

On YouTube a channel called "*rolandmedia*" has a playlist which demonstrates a couple of guitar pedal effects.

Link to the playlist on YouTube: https://www.youtube.com/playlist?list=PL081D4BE59AE08F99

Another YouTube video from the channel "*Mykola MrHardGuitar - Guitar Reviews And Lessons*" which demonstrates various guitar effects: https://www.youtube.com/watch?v=Pg6QaQpSoUc

This was helpful in listening to the effect on guitar sound.

Chosen effect to implement

A choice was made to pick the **Wah-Wah pedal** effect because of its clear distinct effect on the guitar sound. While other effects were more subtle such as the chorus.

YouTube video demo chorus effect:
https://www.youtube.com/watch?v=zmN7fK3fKUE&list=PL081D4BE59AE08F99&index=1

YouTube video demo Wah effect:

https://www.youtube.com/watch?v=qTbuDObjZoA&list=PL081D4BE59AE08F99&index=11

By choosing the Wah-Wah pedal we could employ **qualitative** testing, using our ears to see if the effect has worked.

whereas with the chorus, we would require **quantitative** testing with equipment such as an oscilloscope to check if the effect is working. Getting access to an electronics lab is difficult as they are locked to students outside of lab allocated times. Therefore, it would be preferable if we are not reliant on lab equipment for testing and verification.

How a Wah-Wah pedal works

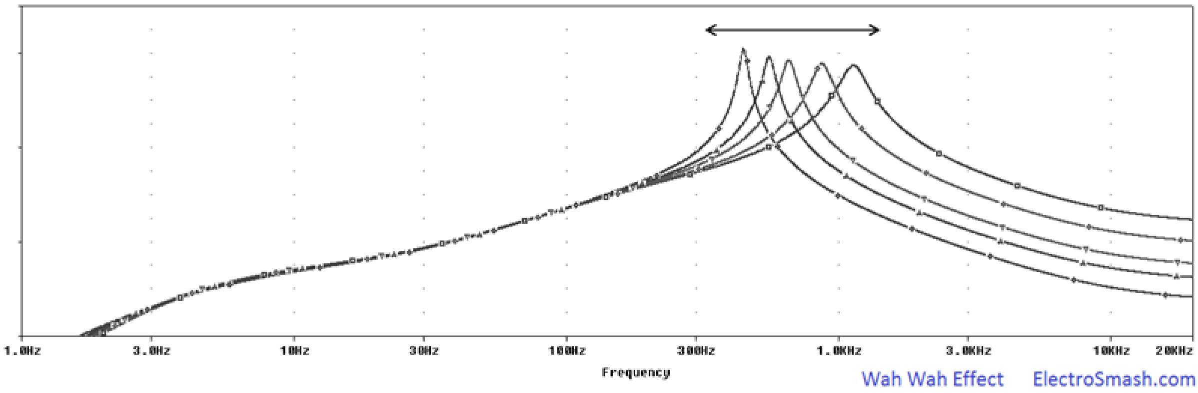


Figure 3 Example Frequency response plot of a Wah-Wah effect

https://www.ultimate-guitar.com/articles/features/guitar_effects_explained_how_does_a_wah_pedal_work-90237

The Wah-Wah effect uses a **bandpass filter** that has a resonant peak, A foot pedal allows the user to move the frequency of the resonant peak up and down, this is known as a spectral glide, which in this case, creates a wah-wah sound

The Q Factor is another aspect of the filter design, it describes the sharpness of the resonant peak. A sharper peak will give a **more pronounced effect.**

A narrower bandwidth will result is a sharper, more pronounced Wah-Wah. As you broaden the bandwidth, the effect becomes less noticeable.

Reflection on the week

We now know which pedal effect we will implement. We know that a bandpass filter is used in a Wah-Wah pedal and that the Q factor makes the effect more pronounced.

Had to divide time between research for this week and work on improving the preliminary report.

Minutes from meeting

The meeting was spent discussing the preliminary report, making improvements and adding possible topics.

Week 3: Research into Wah-Wah effect

Objective: Research Wah-Wah pedal effect.

Downloading owner's manual for a Wah-Wah pedal

With the effect to implement chosen, we went online and downloaded a user manual for a Wah-Wah pedal. The site 'Voxamps.com' was found which sells the V845 & V847 Wah pedals, we downloaded the owner's manual for this and confirmed that the Wah effect is a sweep-able bandpass filter. Where with the foot rocker down the high frequencies are attenuated and, with the rocker up the low frequencies are attenuated.

Download link for V845 & V847 Wah pedal owner's manual: https://voxamps.com/product/v847-wah-pedal/

We found out:

- Confirmed that the Wah effect is a sweep-able bandpass filter.
- It runs on a 9V battery or a DC9V jack
- The output port on the pedal is an output jack, intended to connect into a guitar amp else other pedal.

How to implement Wah effect on signal

The way the Wah-Wah effect is created is by a spectral glide using a bandpass filter, this sweeps between different frequency ranges based on how the pedal is rocked.

To make the Wah effect more pronounced it is best to have a high 'Q' factor.

Choosing Active or passive filter design

Now that we know the type of filter to implement there is a choice of whether to implement the bandpass filter using a passive or active filter.

An active filter uses 'active' components such as Op-Amps or transistor.

Table 2 Active vs Passive filters [14]

	Active filter	**Passive filter**
Component Examples	Transistors, Diodes, Operational amplifiers	Resistors, Capacitors, Inductors,
Power source	Requires external power source to operate in circuit.	Does not require external power source
Power Gain	Can have power gain greater than 1, amplifying signal.	Does not provide power gain, no amplification.
Current and Voltage characteristics	Non-linear characteristics between Current and Voltage	linear characteristics between Current and Voltage

Our decision will be to use an active filter based on an Operational amplifier (Op-amp). An Op-amp's ideal characteristic is such that its input impedance is infinite, and its output impedance is zero. This ideal characteristic would mean that there would be zero loading effects which means that the op-amp does not draw current at its input that would reduce the voltage level of the input voltage.

By using an Op-amp we are minimising the loading effect thereby the signal at the input will not be affected when connected to the output circuit.

We also get the ability to amplify the signal by controlling the gain of the Op-amp via the component values used, this could negate the need for separate amplification circuit is needed.

Sallen-Key Filter

A Sallen-Key filter is a Voltage-Controlled Voltage-Source (VCVS) filter topology that allows *N*th order filter implementations by cascading one or more filters [15]. The Q factor can also be made high which is important in a Wah-Wah pedal as this makes the Wah effect more pronounced.

The sallen and key filter is one that we have worked with during labs and lectures therefore we can make use of our pre-existing knowledge to create a bandpass filter.

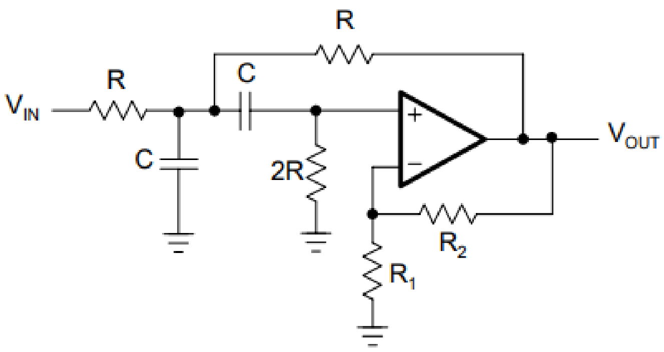


Figure 4 Sallen-Key Band-Pass configuration

Source: "Op Amps for Everyone Design Guide" by Ron Mancini pg. 314

Reflection on the week

We now have a direction on using the sallen-key topology to build a bandpass filter. The two most important aspects of the circuit design will be the sweep-able nature of the filter across frequencies and making the Q factor relatively high so the Wah effect is pronounced.

Minutes from meeting

Discussed with supervisor about creating the bandpass filter using sallen-key since I am already familiar with the sallen-key circuit, spoke about the Q factor and making it variable. Also, about aiming to do the sweep for the Wah-Wah effect using potentiometer ideally. Discussion of having more research for next week on the sallen key type filter and to build one in Pspice Lite.

Week 4: Building sallen-key filter & Researching the Q factor

Objective: build a sallen-key bandpass filter in Pspice, Research and learn more about the “Q” factor.

The Quality factor Q

For a Bandpass filter the quality factor Q is defined as the ratio of the mid frequency (f_m) to the bandwidth (BW) at -3dB points:

$$\text{Quality Factor: Q } = \frac{\mathrm{f_m}}{\mathrm{BW}}$$

It measures the selectivity of a bandpass filter, the higher the value of Q the better selectivity to a particular design frequency.

To show the influence Q has on a bandpass system we can observer the plot in figure 8 which shows the normalized gain response of a second order bandpass filter for different Q values.

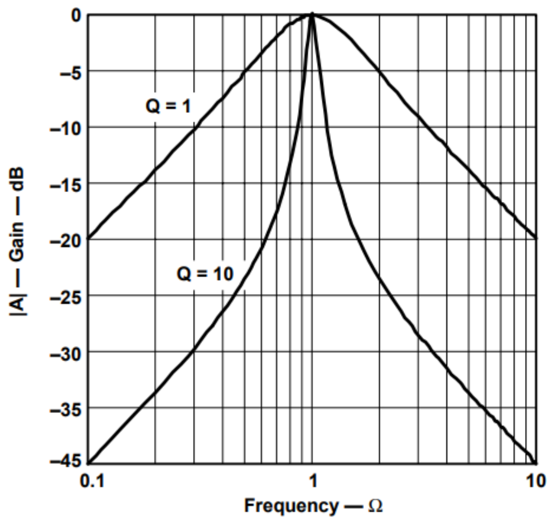


Figure 5 Gain Response of a Second-Order Band-Pass Filter

Source: “Op Amps for Everyone Design Guide” by Ron Mancini pg. 313

It shows that with a higher value of Q the filter response gets steeper at the centre frequency, making it more selective to this frequency than when compared with the response at a lower Q value.

In the Wah-Wah pedal, the effect will become more pronounced with higher values of Q.

Building a sallen-key filter

To build the sallen-key bandpass filter (BPF) we will use the design equations from the book "Op Amps for Everyone Design Guide" by Ron Mancini pg. 314

Refer back to figure 4 for circuit configuration.

Table 3 sallen-key bandpass filter design equations

Source: "Op Amps for Everyone Design Guide" by Ron Mancini pg. 314

mid-frequency: $f_m = \frac{1}{2\pi RC}$

inner gain: $G = 1 + \frac{R_2}{R_1}$

Gain at $f_m : A_m = \frac{G}{3 - G}$

Filter quality: $Q = \frac{1}{3 - G}$

From Table 3 we can observe that the Q factor can be varied independently without changing the mid frequency (fm). However, Am and Q are linked together via the inner gain (G) and cannot be adjusted independently.

Ideally, we would like one component to make variable to change fm, one for filter quality Q and one for the inner gain (G).

Note: when G approaches a value of 3, the circuit becomes unstable due to Am becoming infinite.

The upper limit of frequency rage for the human ear is approximately 20k Hz, therefore for this initial test we set the centre frequency to 10kHz.

Because we do not yet know the frequency rage we are working with in the guitar, we will design a bandpass filter with a mid-frequency of 10kHz.

Putting these equations into an excel page to automate the calculation for mid-frequency of 10khz, resistor R of 100k ohms, resistor R2 of 10k ohms and inner gain G of 2.8, so as not to be greater than a value of 3.

Pick the following:	
fm :	10000 hz
R:	100 k ohms
R2:	10 k ohms
Gain (G)	2.8

Result:	
C ==	1.59E-10 F
R1 ==	5555.56 ohms
Am ==	14
Q ==	5

Figure 6 sallen-key bandpass filter design equations into Excel

Figure 4 we get the values of the capacitor and R2 required to meet the design. It also informs us that the gain at the mid frequency known as *Am* is 14 while the Quality factor is 5.

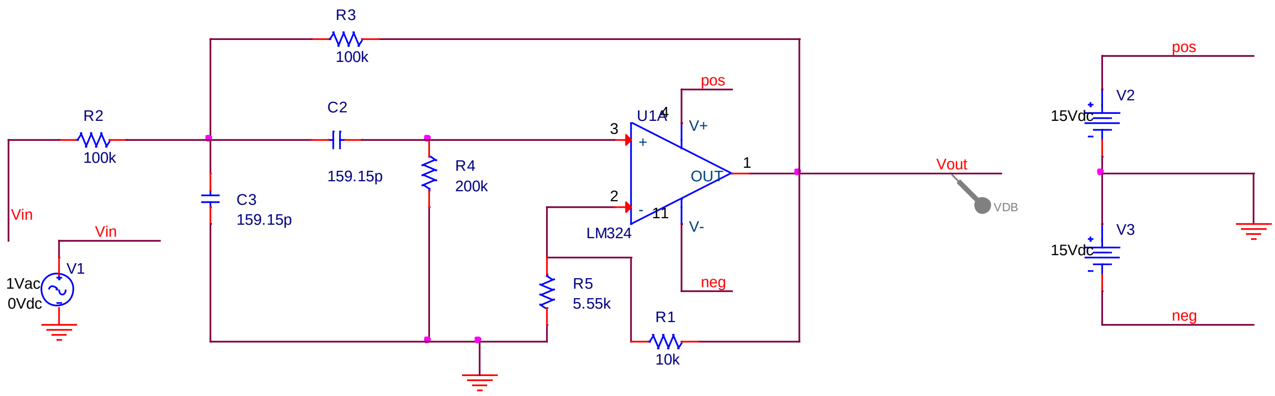


Figure 7 Sallen-Key BPF circuit built in Pspice

The circuit built is shown in figure 7 with an LM324 which is an Op-amp found in the labs and can be used for this project. The power supply is set to +/- 15 volts for now. Using an AC source, we can conduct an AC sweep analysis. The simulation settings used are shown in figure 8.

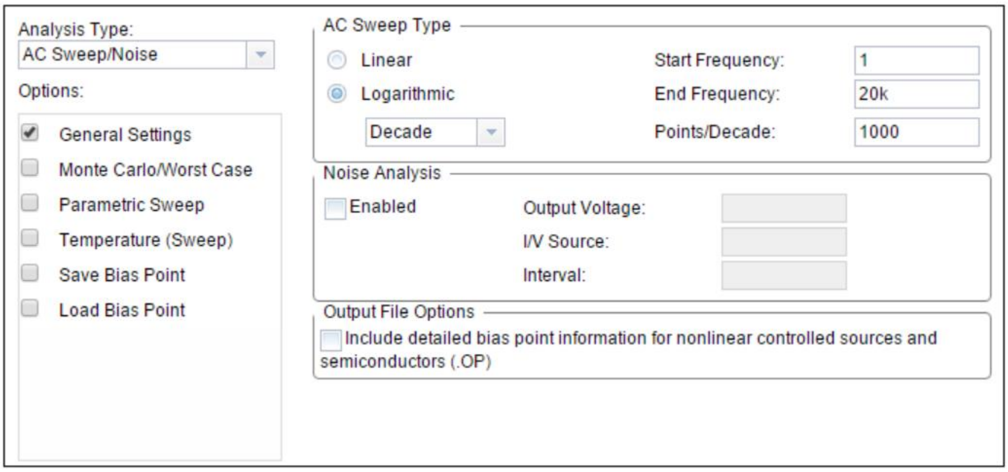


Figure 8 Simulation settings for Sallen-key BPF

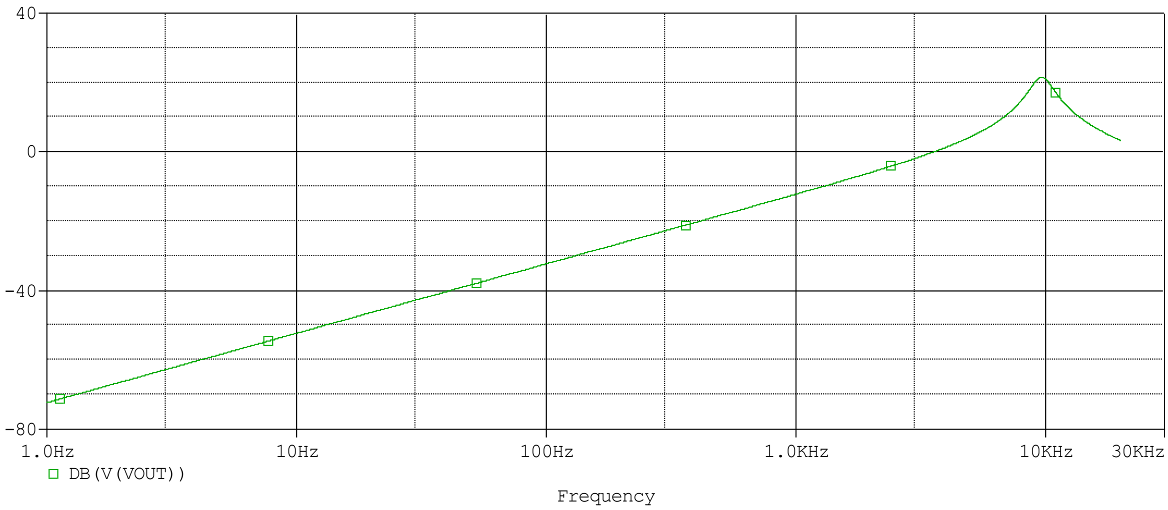


Figure 9 Simulation frequency response result

From figure 9 we observe that a Q factor of 5 has produced a response that should be particularly selective enough for the Wah-Wah effect. Using cursor, we can get the frequency and dB value at the peak mid frequency:

Trace Color	Trace Name	Y1
	X Values	9.662K
CURSOR 1,2	DB(V(VOUT))	21.511

Figure 10

The mid frequency peek should occur at: $20log(G * Q) = X\ dB$

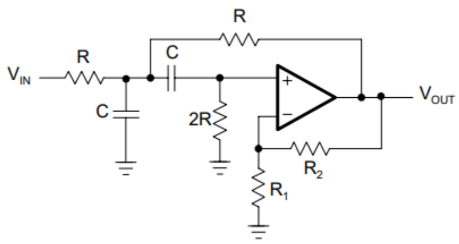
By mathematical calculation: $20log(2.8 * 5) = 22.9\ dB$

By simulation result in figure 10 it is 21.5 dB

The mid frequency is 9.6kHz in the simulation therefore it is 0.4kHz off from our design of 10kHz.

Reflection on the week

This week we have learned more about the Q factor and have begun simulation of a band pass filter (BPF). After taking a second look at the equation for mid frequency $f_m = \frac{1}{2\pi RC}$ and observing the sallen-key BPF circuit (shown below for convenience). In order to allow it to sweep different frequencies, the resistor value R must change in ALL parts of the circuit the same amount. It would be very difficult to achieve this especially considering that one of the resistors is twice R.



Minutes from meeting

The sallen key had some disadvantages, mainly the fact that its mid freq :fm = 1/2piRC therefore all resistor values would have to change at the same time in order for the fm to shift.

Therefore, we will not move ahead with this particular topology for a bandpass filter.

In this meeting we also discussed looking at multiple types of circuit for implementing a bandpass filter and comparing them, then making an informed choice on which topology to use.

Week 5: Compare & Contrast other filter types to implement

Objective: To find and compare other filter topologies for implementing a bandpass filter.

This week we are comparing different bandpass circuits and comparing them, to pick the best one. The most common filter topologies encountered are the sallen-key, multi-feedback and the ones where two first order low and high pass filters are cascaded to form a bandpass filter.

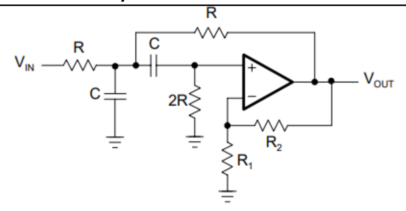
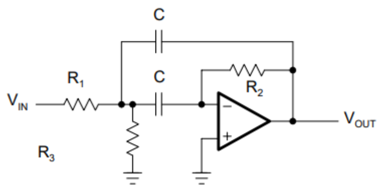
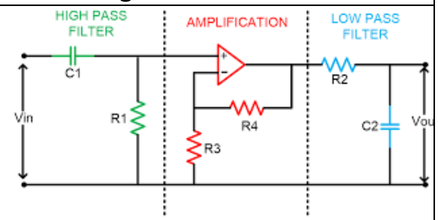
Table 4 Image sources:

Image source for Cascading HPF with LPF circuit design: https://www.electrical4u.com/band-pass-filter/

Image source for Sallen-Key BPF circuit design: Source: “Op Amps for Everyone Design Guide” by Ron Mancini pg. 314

Image source for Sallen-Key BPF circuit design: Source: “Op Amps for Everyone Design Guide” by Ron Mancini pg. 315

Table 4 Comparing different bandpass implementations

	Sallen-Key BPF	Multiple Feedback BPF	Cascading HPF with LPF
circuit design			
Design Equations	mid-frequency: $f_m = \frac{1}{2\pi RC}$ inner gain: $G = 1 + \frac{R_2}{R_1}$ gain at f_m: $A_m = \frac{G}{3-G}$ filter quality: $Q = \frac{1}{3-G}$	mid-frequency: $f_m = \frac{1}{2\pi C}\sqrt{\frac{R_1 + R_3}{R_1 R_2 R_3}}$ gain at f_m: $-A_m = \frac{R_2}{2R_1}$ filter quality: $Q = \pi f_m R_2 C$ bandwidth: $B = \frac{1}{\pi R_2 C}$	sperate eq for HPF & LPF $F_{low} = \frac{1}{2\pi R1C1}$ $F_{high} = \frac{1}{2\pi R2C2}$ $Gain = \frac{R4}{R3}$
advantage	• Q can be varied via the gain without modifying the mid frequency.	• Bandwidth and Gain do not depend on R3. • R3 can be used to modify the mid freq. • For low values of Q, the filter can work without R3 **-5 components used**	• Good for fixed BPF designs • Easy to change fL and fH separately. **-6 components used**
disadvantage	• Q and gain (Am) cannot be adjusted independently • All resistor values in circuit must change same amount **-7 components used**	• Possibility of Transfer function derivation more difficult to get.	• Cannot control filter sweep with one component. (There is 1 for fLow and 1 for fHigh)

Implementing a multi-feedback bandpass filter

From Table 4, after comparing the three common methods for implementing a BPF the multi-feedback topology is the attractive option due to the ability to use one resistor to change the mid frequency. It also uses the minimum number of components.

Table 5 Multi-feedback BPF design equations

Source: "Op Amps for Everyone Design Guide" by Ron Mancini pg. 315

$$\text{mid-frequency: } f_m = \frac{1}{2\pi C}\sqrt{\frac{R_1+R_3}{R_1R_2R_3}}$$

$$\text{Gain at } f_m: \ -A_m = \frac{R_2}{2R_1}$$

$$\text{Filter quality: } Q = \pi f_m R_2 C$$

$$\text{Bandwidth (BW): B} = \frac{1}{\pi R_2 C}$$

$$\text{R2} = \frac{Q}{\pi f_m C}$$

$$\text{R1} = \frac{R_2}{-2A_m}$$

$$\text{R3} = \frac{-A_m R_1}{2Q\text{\^{}}2 + A_m}$$

The multi-feedback topology uses the negative terminal of the op-amp therefore in the design equation the gain will have a negative symbol.

Putting these equations into an excel page to automate the calculation for mid-frequency of 10kHz. The capacitor value will be set to 100nF. This time the Q factor can be set to a particular value. The gain will be set to 1 therefore there should be no amplification of the signal at the mid frequency.

Pick the following:				
Fm	10000 Hz			
C	0.0000001 F			
Q	10			
Am	-1			
Result:				
R2 =	3183.098862 ohms	>>		3.18 k ohms
R1 =	1591.549431 ohms	>>		1.59 k ohms
R3 =	7.997735834 ohms	>>		0.01 k ohms
BW =	500 Hz			

Figure 11 multi-feedback BPF design equations into Excel component result

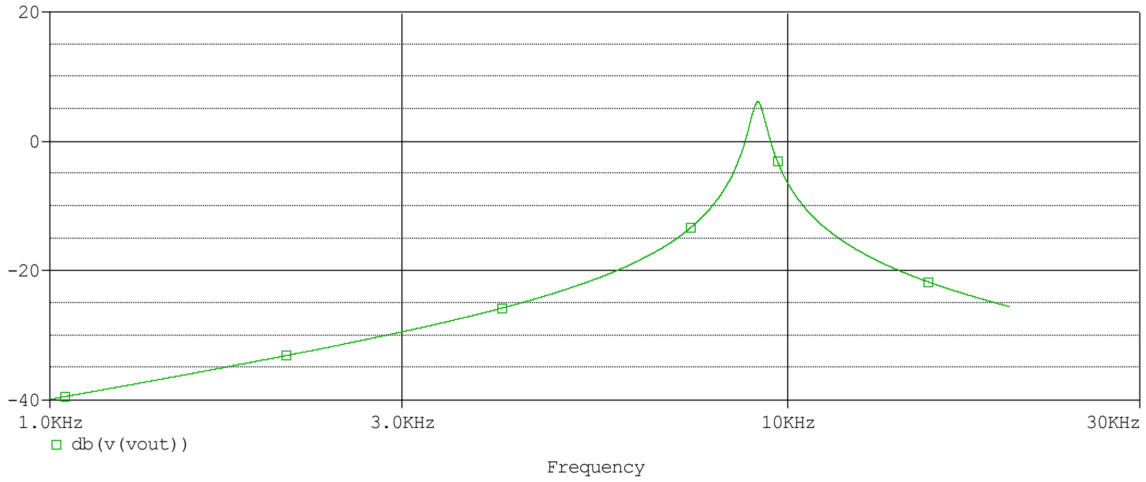


Figure 12 Simulation frequency response result

In figure 12 we can observe that while we designed the system with a gain of -1dB at the centre frequency of 10k Hz, we are getting a gain beyond 5 dB this is a very big difference and we can only assume a mistake as occurred either in the calculation of the component values or the simulation.

Reflection on the week

Need to back over component calculations and check simulation to try and determine why we are getting very high gains.

Minutes from meeting

Getting guitar low and high frequency values, talking about the front end and back end for the filter. Discussed that the system will alter the amplitude and phase of the signal, and this may need to be mitigated using the front and back-end block circuits.

Week 6: Finding out guitar harmonics

Objective: This week we will get the first, second and third harmonics of the guitar so that we can design the BPF for this particular guitar.

Use “finding out guitar harmonics “ in excel

We need to find the low and high frequency range we want our bandpass filter to work at. To do this we w

Methodology:

1. Take an audio recording when the highest frequency cord of the guitar is played i.e., plucked.

2. Using Mathlab code shown in figure 13, plot the frequency and time plots of the recorded file.

3. Find the first, second and third harmonics frequency values in the frequency domain plot.

4. Save these values along with their corresponding amplitude voltage into an excel sheet.

5. Repeat steps 1-4 multiple times, resulting in multiple recordings (v1, v2,v3 etc)

6. Cross-reference the multiple recordings and take the average value for the frequency/Amplitude.

7.Repeat steps 1-6 with the low frequency cord.

```

 [sample_data, sample_rate] = audioread('highv2.mp4');

sample_period = 1/sample_rate;
t = (0:sample_period:(length(sample_data)-1)/sample_rate);
figure(1)
plot(t,sample_data)
title('Time Domain Representation')
xlabel('Time (seconds)')
ylabel('Amplitude')
xlim([0 t(end)])

m = length(sample_data); % Original sample length.
n = pow2(nextpow2(m));
y = fft(sample_data, n);
f = (0:n-1)*(sample_rate/n);
amplitude = abs(y)/n;
figure(2)
plot(f(1:floor(n/2)),amplitude(1:floor(n/2)))
title('Frequency Domain Representation ')
xlabel('Frequency')
ylabel('Amplitude')
```

Figure 13 Mathlab code used to obtain time and frequency representation plots

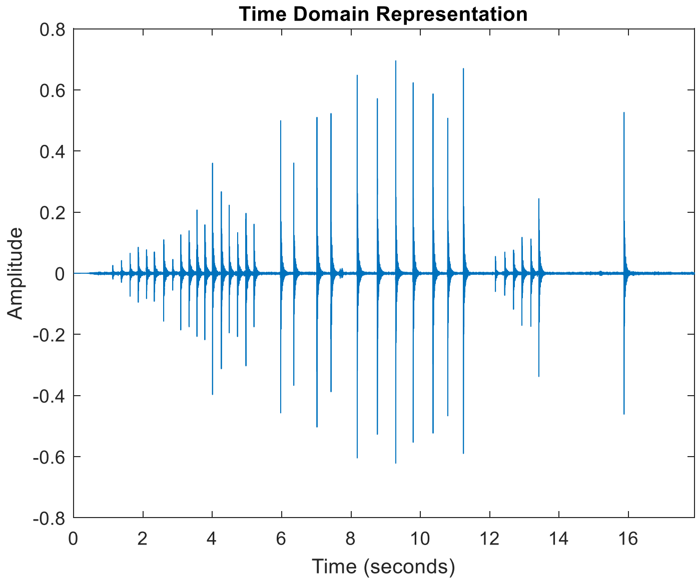


Figure 14 Time domain plot of file: highv2

In figure 14, we have plucked the guitar multiple times during a single recording. We can note that the amplitude varies from each pluck and depends how hard the string is plucked.

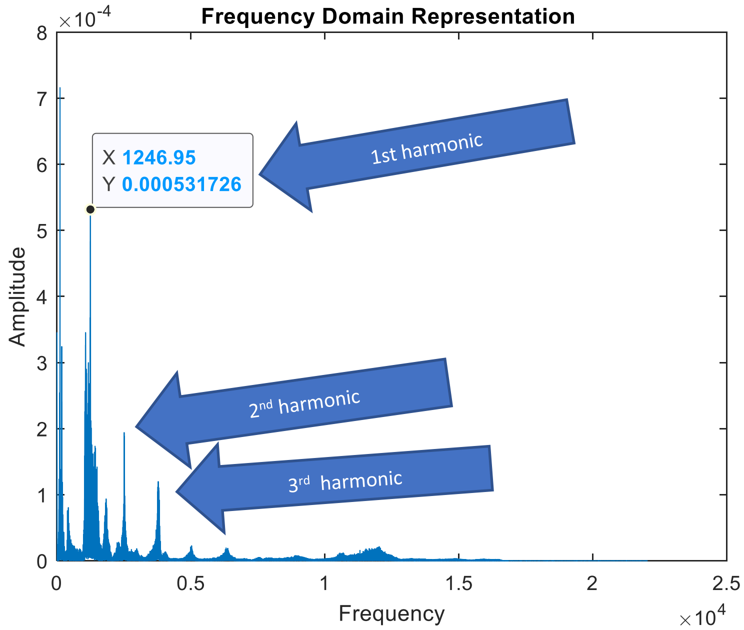


Figure 15 Frequency domain plot of file: highv2

From figure 15, the reason we only collect data from the first 3 harmonics of the signal is because they are easily distinguishable visually from the rest of the signal.

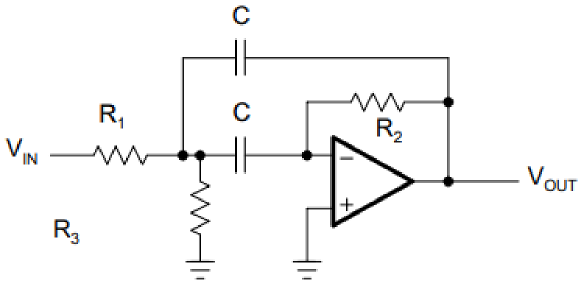
The tabulated results were put into an excel file, the resulting data is shown in Table 6.

		Harmonics (hz)			Amplitude (v)			
	high freq cord	1st	2nd	3rd	1st	2nd	3rd	
*	highv1	1041	2413	3730	7.37E-05	3.08E-05	6.27E-06	
	highv2	1246	2507	3775	0.000532	0.0019	0.000119	
	highv4	1065	2506	3767	0.0002	0.00014	6.13E-05	
	highv5	1061	2505	3773	0.00043	0.00024	0.0001	
	high_cleanv1	1241	2487	3743	0.0013	4.50E-05	9.33E-05	
	average	**1130.8**	**2483.6**	**3757.6**	**0.000507**	**0.000471**	**7.6E-05**	v
					0.507085	0.47116	0.075974	mv
	low freq cord							
	lowv1	257	515	777	0.0018	0.00024	1.50E-04	
	lowv2	257	516	776	0.0017	0.00048	0.00015	
*	lowv3	257	516	776	0.0022	0.00055	3.40E-04	
*	lowv4	257	515	776	0.0022	0.00025	0.00016	
	low_cleanv1	256	512	773	0.0012	0.0004	2.60E-04	
	average	**256.8**	**514.8**	**775.6**	**0.00182**	**0.000384**	**0.000212**	v
					1.82	0.384	0.212	mv

For out bandpass filter we will decide to make it work for the first harmonics of the low and high frequency cords as the first harmonic contains the most energy in terms of voltage. If we observe figure 15 the first harmonic is almost three times more than the second harmonic.

Attempt 1: Deriving the multi-feedback BPF transfer function

The transfer function we are aiming to produce is shown in figure 14 with A(s). we apply Kirchhoff's Current Law (KCL) at two nodes in order to get the transfer function.



$$A(s) = \frac{-\frac{R_2R_3}{R_1+R_3}C\omega_m \cdot s}{1 + \frac{2R_1R_3}{R_1+R_3}C\omega_m \cdot s + \frac{R_1R_2R_3}{R_1+R_3}C^2 \cdot \omega_m{}^2 \cdot s^2}$$

Figure 16 multi-feedback transfer function

Source: "Op Amps for Everyone Design Guide" by Ron Mancini pg. 315

Let one node be called "Vx" and the other "Vz" as shown below:

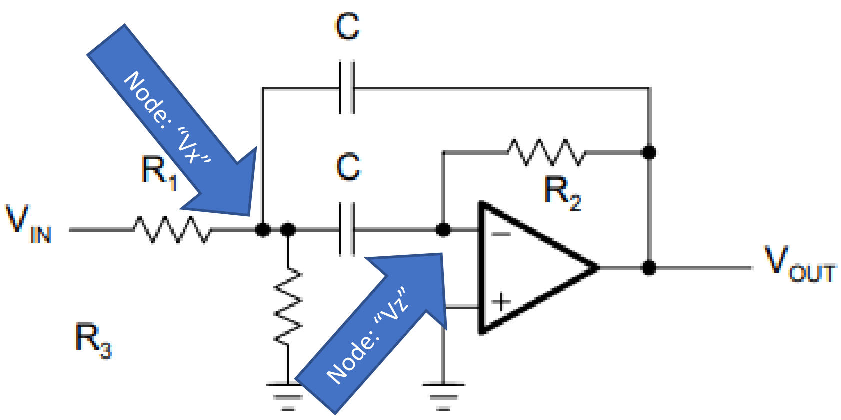


Table 6 deriving multi-feedback transfer function attempt 1

KCL @ node Vx: $$\frac{Vx-Vi}{R1}+\frac{Vx-Vo}{1/sC2}+\frac{Vx-0}{R3}+\frac{Vx-0}{1/sC1}=0$$ $$\frac{Vx}{R1}-\frac{Vi}{R1}+sC2(Vx-Vo)+\frac{Vx}{R3}+sC1Vx\ =\ 0$$ $$\frac{Vx}{R1}-\frac{Vi}{R1}+sC2Vx\ -\ sC2Vo+\frac{Vx}{R3}+sC1Vx\ =\ 0$$ $$-\frac{Vi}{R1}+Vx(\frac{1}{R1}\ -\ sC2+\frac{1}{R3}+sC1)\ =\ sC2Vo$$ $$-\frac{Vi}{R1sC2}+Vx(\frac{1}{R1sC2}\ +1+\frac{1}{R3sC2}+\frac{sC1}{sC2})\ =\ Vo$$
KCL @ node Vz: where Vz is a virtual ground => 0v. $$\frac{Vz-Vo}{R2}+\frac{Vz-Vx}{1/sC1}=0$$ $$-\frac{Vo}{R2}-sC1Vx=0$$ $$-\frac{Vo}{R2}\ =\ sC1Vx$$ $$-Vo\ =\ sC1VxR2$$ $$-Vo\ =\ Vx(-sC1R2)$$
we now have two equations: $EQ1:\quad -Vo\ =\ Vx(-sC1R2)$ $EQ2:\quad Vo\ =\ -\frac{Vi}{R1sC2}+Vx(\frac{1}{R1sC2}\ +1+\frac{1}{R3sC2}+\frac{sC1}{sC2})$ **Got stuck here on how to extract Vo/Vi from the above simultaneous equations**

Reflection on the week

We now know to the range we require the bandpass filter to work within is approx. 250hz to 1100Hz. We got started on the transfer function, which is not complete yet.

Minutes from meeting

In the meeting through my supervisor, I realised the values of amplitude in volts I had collected and put into the excel file was the recording of sound waves felt at the mic. This was an indirect measurement and not accurate because a closer mic to the sound source would record higher amplitude values while if the mic was further away it would record very little sound pressure hitting it and therefore very little amplitude(voltage).

During the meeting we connected an oscilloscope directly to the guitar (this was the correct way I should have done at the start) measured the voltage value for the high and low cords.

We reviewed my transfer function rough work for multi filter BPF. He told me to work on a simple BPF transfer function first and work my way up to the multi-feedback one.

Decided the frequency range to be 200Hz to 2 kHz while the amplitude rage was 0.02v to 0.5 v as measured from the digital oscilloscope.

Week 7: Derivation & Simulation on multi-feedback BPF filter

Objective: continue simulation work on multi-feedback BPF, this time with design values and continue to derive the transfer function.

Since the system will be connected to a guitar speaker (which will already have an amplifier built into it). We will not have any gain in our design. We will aim to ensure the output of the BPF stays within the same range as input guitar signal.

We are building today

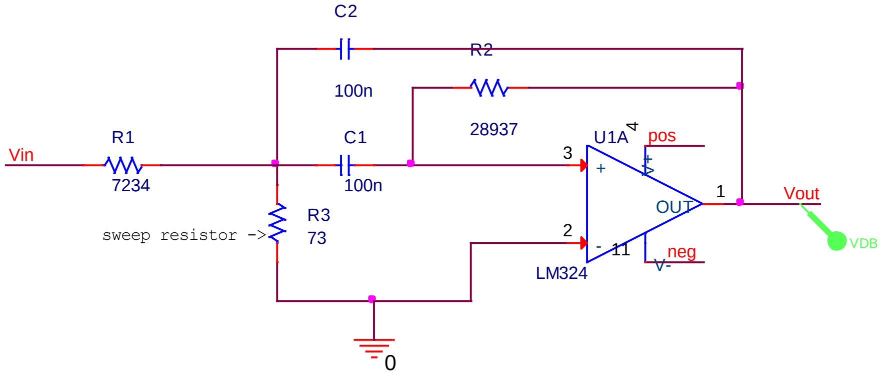


Figure 17 week 7 multi-feedback BPF

Using our multi-feedback excel design sheet, in this test we design the BPF with a gain Am of 1, thus there will be a little bit of gain in system to amplify the signal.

Fm	1100	Hz
C	0.0000001	F
Q	10	
Am	-1	
R2 =	28937.26238	ohms
R1 =	14468.63119	ohms
R3 =	72.7066894	ohms
B	55	Hz

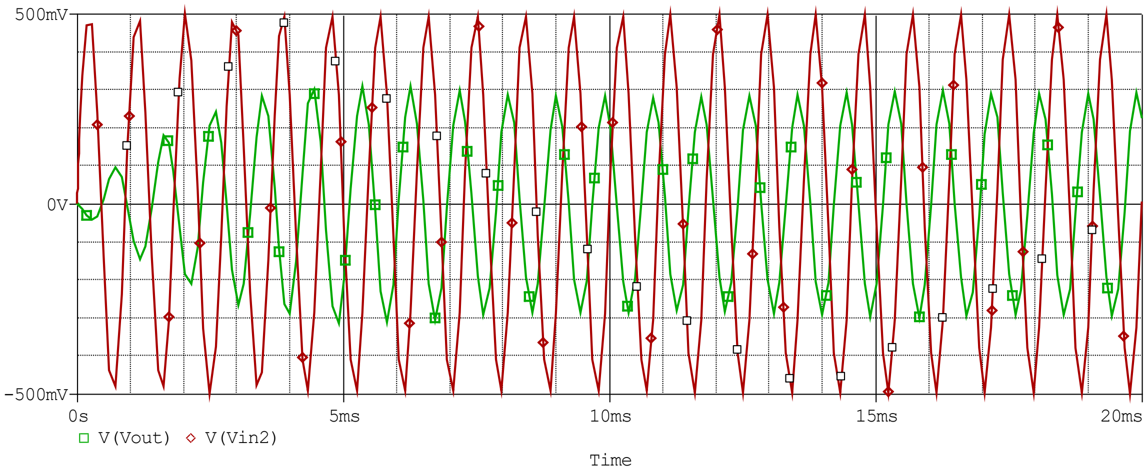


Figure 18

There should be no attenuation between the input signal Vin2 and the output signal Vout when the frequency of Vin2 is set to 1100Hz, the designed mid frequency. There is a reduction of 200mV to the output signal.

Research on how to get Vout/Vin from the simultaneous equation (from table 6) in the first attempt at derivation of the transfer function has come across Cramer's rule. Which is an explicit formula for the solution of a system of linear equations with as many equations as unknowns.

Table 7 Cramer's Rule [16]

Cramer's Rule:

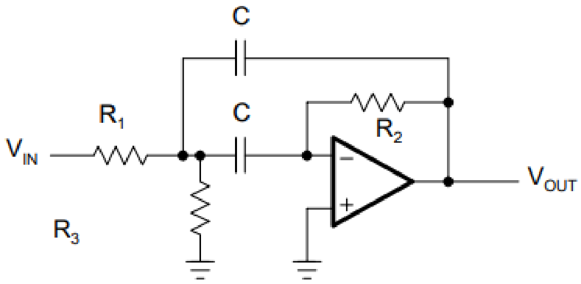
EQ1: $a_1x + b_1y = c_1$
EQ2: $a_2x + b_2y = c_2$

$$x = \frac{\begin{vmatrix} b_1 & c_1 \\ b_2 & c_1 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}} = \frac{c_1b_2 - b_1c_2}{a_1b_2 - b_1a_2}$$

$$y = \frac{\begin{vmatrix} a_1 & c_1 \\ a_2 & c_1 \end{vmatrix}}{\begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix}} = \frac{a_1c_2 - c_1a_2}{a_1b_2 - b_1a_2}$$

Source: C. Moler, "Cramer's rule on 2-by-2 systems", *ACM SIGNUM Newsletter*, vol. 9, no. 4, pp. 13-14, 1974. Available: 10.1145/1206085.1206090 [Accessed 15 March 2022].

Attempt 2: Deriving the multi-feedback BPF transfer function



$$A(s) = \frac{-\frac{R_2R_3}{R_1+R_3}C\omega_m \cdot s}{1 + \frac{2R_1R_3}{R_1+R_3}C\omega_m \cdot s + \frac{R_1R_2R_3}{R_1+R_3}C^2 \cdot \omega_m{}^2 \cdot s^2}$$

Figure 19 multi-feedback transfer function

Source: "Op Amps for Everyone Design Guide" by Ron Mancini pg. 315

KCL @ node Vx:

$$\frac{Vx - Vi}{R1} + \frac{Vx - Vo}{1/sC2} + \frac{Vx - 0}{R3} + \frac{Vx - 0}{1/sC1} = 0$$

$$\frac{Vx}{R1} - \frac{Vi}{R1} + sC2(Vx - Vo) + \frac{Vx}{R3} + sC1Vx = 0$$

$$\frac{Vx}{R1} - \frac{Vi}{R1} + sC2Vx - sC2Vo + \frac{Vx}{R3} + sC1Vx = 0$$

$$-\frac{Vi}{R1} + Vx(\frac{1}{R1} - sC2 + \frac{1}{R3} + sC1) = sC2Vo$$

$$-\frac{Vi}{R1sC2} + Vx(\frac{1}{R1sC2} + 1 + \frac{1}{R3sC2} + \frac{sC1}{sC2}) = Vo$$

KCL @ node Vz: where Vz is a virtual ground => 0v.

$$\frac{Vz - Vo}{R2} + \frac{Vz - Vx}{1/sC1} = 0$$

$$-\frac{Vo}{R2} - sC1Vx = 0$$

$$-\frac{Vo}{R2} = sC1Vx$$

$$-Vo = sC1VxR2$$

$$-Vo = Vx(-sC1R2)$$

we now have two equations:

$EQ1: \quad -Vo = Vx(-sC1R2)$

$EQ2: \quad Vo \ = -\frac{Vi}{R1sC2} + Vx(\frac{1}{R1sC2} + 1 + \frac{1}{R3sC2} + \frac{sC1}{sC2})$
Rearrange them to use Cramer's rule: $EQ1 -> : \quad 0 \ = -Vo \ + \ Vx(-sC1R2)$ $EQ2->: \quad \frac{Vi}{R1sC2} = -Vo \ + Vx(\frac{1}{R1sC2} + 1 + \frac{1}{R3sC2} + \frac{sC1}{sC2})$
Apply Cramer's rule: $Vo \ = \frac{determinant\ of \begin{vmatrix} 0 & sC1R2 \\ \frac{Vi}{R1sC2} & \frac{1}{R1sC2} + 1 + \frac{1}{R3sC2} + \frac{sC1}{sC2} \end{vmatrix}}{determinant\ of \begin{vmatrix} 1 & sC1R2 \\ -1 & \frac{1}{R1sC2} + 1 + \frac{1}{R3sC2} + \frac{sC1}{sC2} \end{vmatrix}}$ $Vo \ = \frac{\frac{-VisC1R2}{sC2R1}}{\frac{1}{R1sC2} + \frac{1}{R3sC2} + \frac{sC1}{sC2} + 1 + sC1R2}$ $\frac{Vo}{Vi} \ = \frac{-sC1R2}{1 + \frac{R1}{R3} + s^2C1C2R1 + sC1C2R1R2}$

Reflection on the week

After researching on how to apply Cramer’s rule we could then apply it to find Vo/Vi transfer function. Now work can move to using this transfer function in MATLAB to view the poles & zeros.

Minutes from meeting

Time was spent reviewing he simulation that had been created and viewing the input/output signal and if they were changed in the expected way by the filter. Discussion on using multiple literature sources (eg the book by Sergio Franco - DESIGN WITH OPERATIONAL AMPLIFIERS AND ANALOG INTEGRATED CIRCUITS) to validate the derived transfer function.

Week 8: review week spent mostly working on report and overhauling logbook.

Week 9: Hardware test & simulation of audio file as input & Continued work on transfer function derivation

Objective: finalise the circuit design and begin building on breadboard**.**

We are going to create a new project for the multi-feedback filter because the current project, when simulated gives values which are significantly far from the expected values for eg, the centre frequency was off by nearly 1k Hz in the VBD plot.

Go to the excel design page and use the calculated resistors values for the parameters chosen.

	A	B	C	D	E	F	G	H	I
1	Choose								
2	fmid	1100	hz						
3	Q	10		, to verify get the -3dB freq values, do fm/ BW					
4	Ho	0	dB	,resonance gain magnitude					
5									
6									
7	Let C ==	100	nF	>>	0.0000001	F			
8	ωo is	6911.504	rad/s						
9									
10									
11				~~that it increases quadratically with~~ Q. If we want $H_0 < 2Q^2$, we must replace R_1					
12	Ho is	1	V/V	with a voltage divider in the manner of Example 3.9. The design equations are then					
13	Is	1	<	200	?, IF we want this to be true then use voltage divider				
14		^Ho val^		^2Q val^					
15									
16									
17									
18	R2 is	28937.26	ohms>>	28.94	k ohms				
19	R1a is	14468.63	ohms>>	14.47	k ohms				
20	R1b	72.71	ohms>>	0.07	k ohms				

Figure 20 Calculated components from Excel

Building the circuit with +/- 9v DC and a 0.5v AC source for the frequency sweep test.

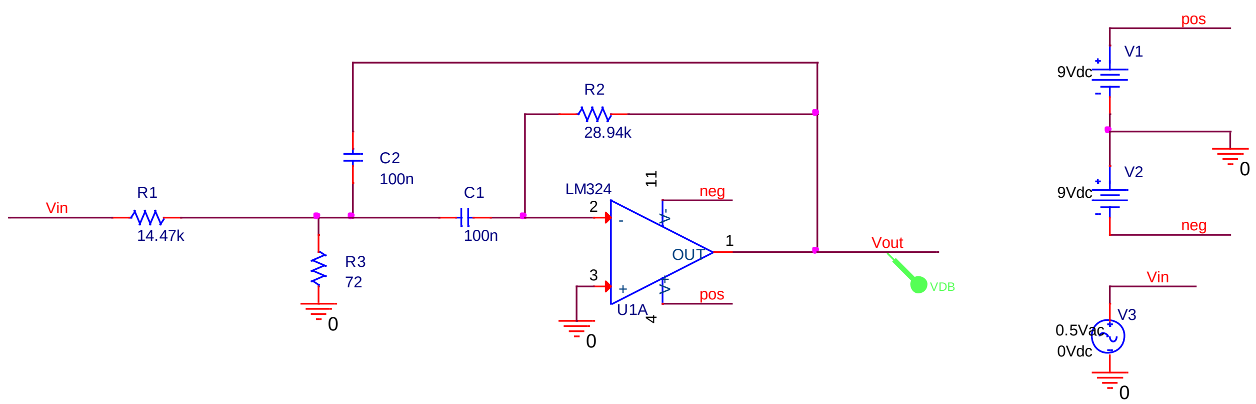


Figure 21 Frequency sweep test set up

The following plot is generated, using cursor we check that the mid frequency is indeed at 1100hz.

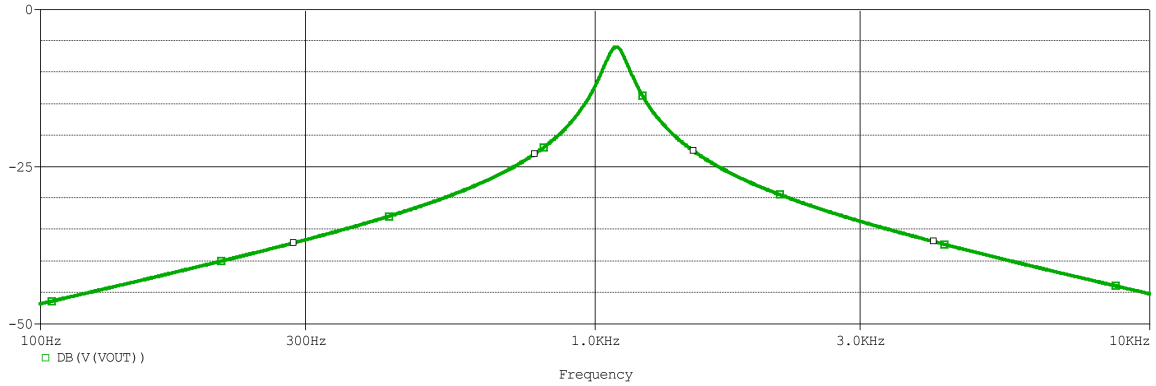
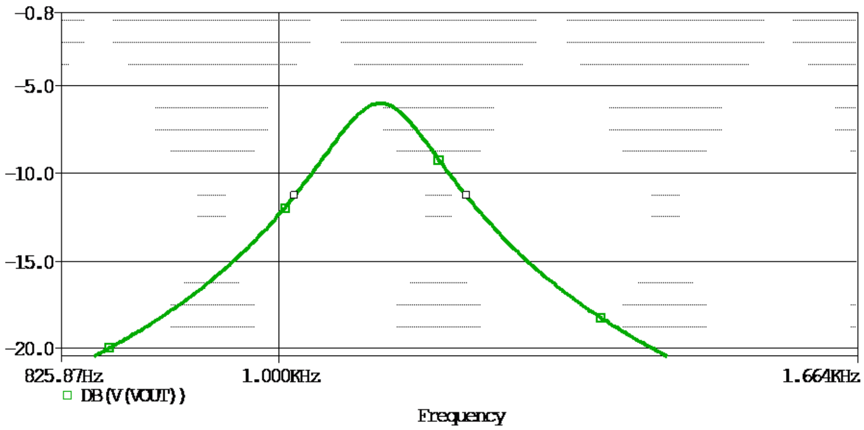


Figure 22 Frequency sweep simulation result

Zoomed in



Using cursor at the mid frequency peek of the graph:

Trace Color	Trace Name	Y1
	X Values	1.0940K
CURSOR 1,2	DB(V(VOUT))	-5.9761

<u>Verifying Q to be 10:</u>

Get the BW at the -3dB points, add -3dB to the peek gain of the system

-5.9761dB -3dB = -8.97 dB,

Get the frequency values at -8.97 dB both sides which are 1.149khz & 1.042khz.

Q = fmid/ BW => 1100/([1.149khz-1.042khz]*1000) = 10.28

To predict output voltage, using input voltage which was 0.5v and voltage gain/loss (dB):

20log(0.5/?) = 6dB, answer is 0.25v for an input of 0.5v

Using a voltage probe instead of VDB probe we can check the voltage behaviour across the frequency sweep:

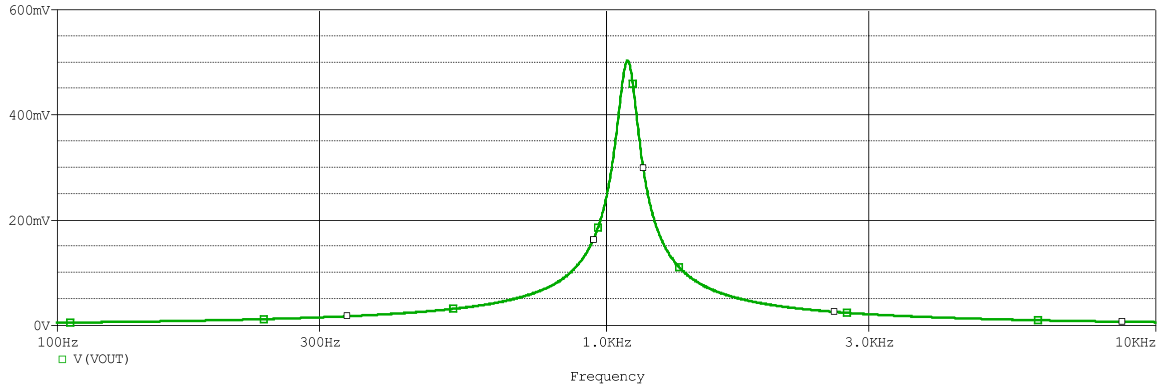


Figure 23 frequency sweep voltage behaviour

At the mid frequency of 1100Hz we get out 0.5v which is the same voltage level that was sent in using the 0.5v AC source.

Now we test with a frequency source at 0.5v @ 1100hz.

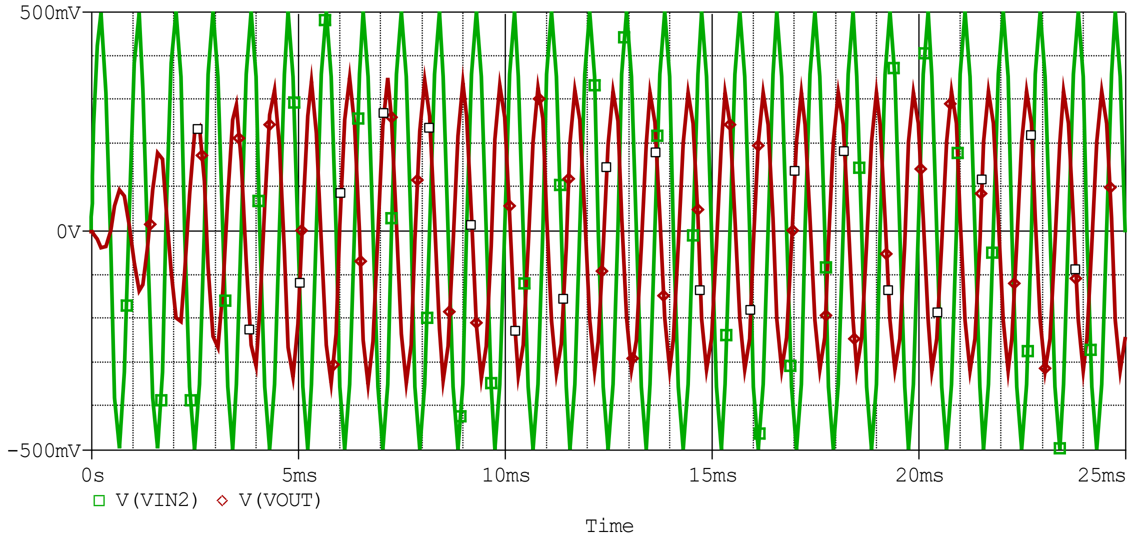


Figure 24 1100 Hz @ 0.5v frequency input test

The first thing that stands out is that the output is not at the same voltage level as the input.

The dB loss here is: -20log(0.5v/0.3v) = -4.43dB

The ratio of out/in is 0.6, therefore we are losing 40% of the voltage.

If we go back to the excel sheet and change the resonance gain magnitude (Ho) value to +6 dB, this should offset the loss of -6dB we are seeing.

	A	B	C	D	E	F	G	H	I
1	Choose								
2	fmid	1100	hz						
3	Q	10		, to verify get the -3dB freq values, do fm/ BW					
4	Ho	6	dB	,resonance gain magnitude					
5									
6									
7	Let C ==	100	nF	>>	0.0000001	F			
8	ωo is	6911.504	rad/s						
9									
10									
11				~~that it increases quadratically with Q~~. If we want $H_0 < 2Q^2$, we must replace R_1					
12	Ho is	1.995262	V/V	with a voltage divider in the manner of Example 3.9. The design equations are then					
13	Is	1.995262	<	200	?, IF we want this to be true then use voltage divider				
14		^Ho val^		^2Q val^					
15									
16									
17									
18	R2 is	28937.26	ohms>>	28.94	k ohms				
19	R1a is	7251.493	ohms>>	7.25	k ohms				
20	R1b	73.07	ohms>>	0.07	k ohms				

Figure 25 Calculated components from Excel with adjusted gain at centre frequency of +6dB

Re-running the simulation, we observe what we initially should have expected from the first test where the centre frequency of 1100 Hz has a gain of 0 dB .

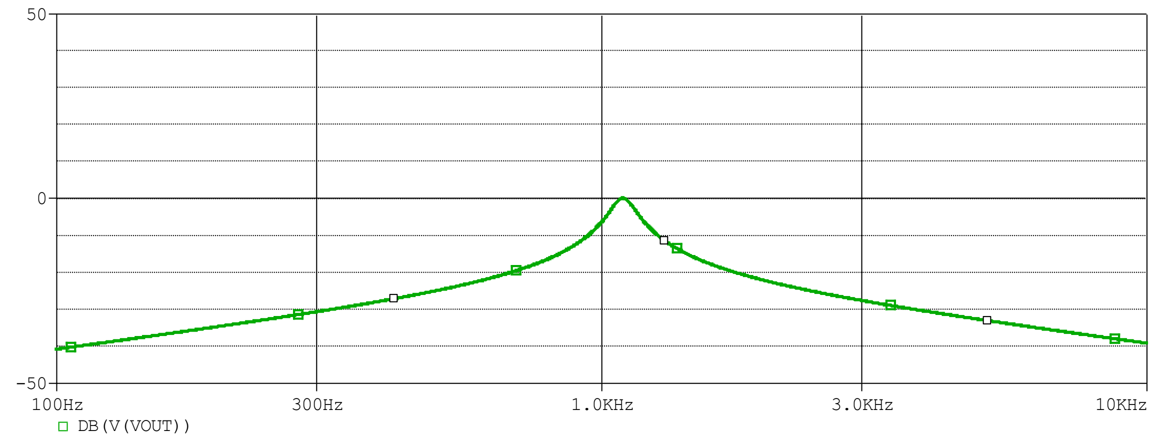


Figure 26 Frequency sweep simulation result with +9 dB Gain

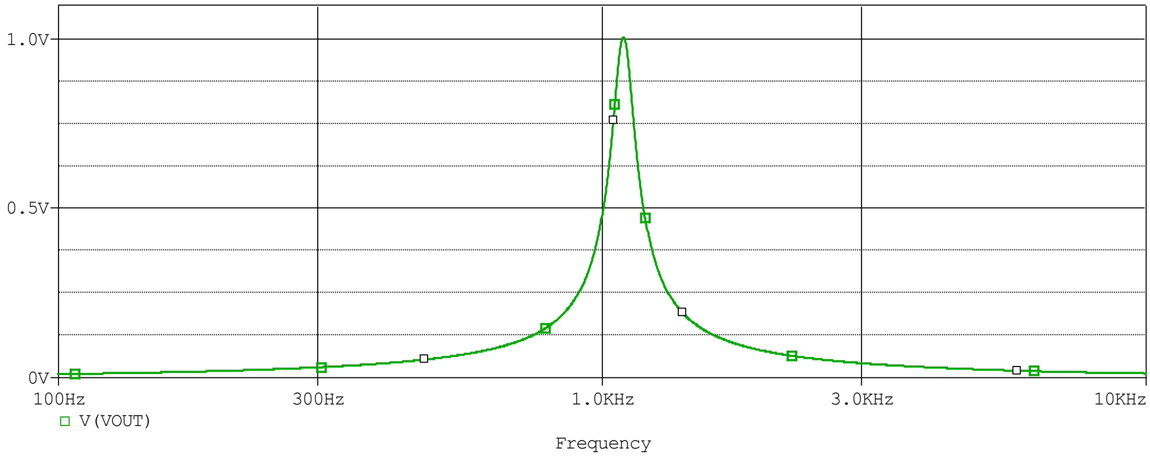


Figure 27 frequency sweep voltage behaviour with +9 dB Gain

With +6 dB gain at the resonate frequency:

20log(Pout/Pin) = +6dB

Pout/Pin = 10^6/20

Pout/Pin = 2

Therefore, there is a doubling of voltage by the system which matches the plot since 2*0.5v = 1v. We observe it is indeed 1v at 1100 Hz. This is not desired as we the voltage going in and voltage going out to be the same.

Applying a frequency of 1100hz at 0.5v:

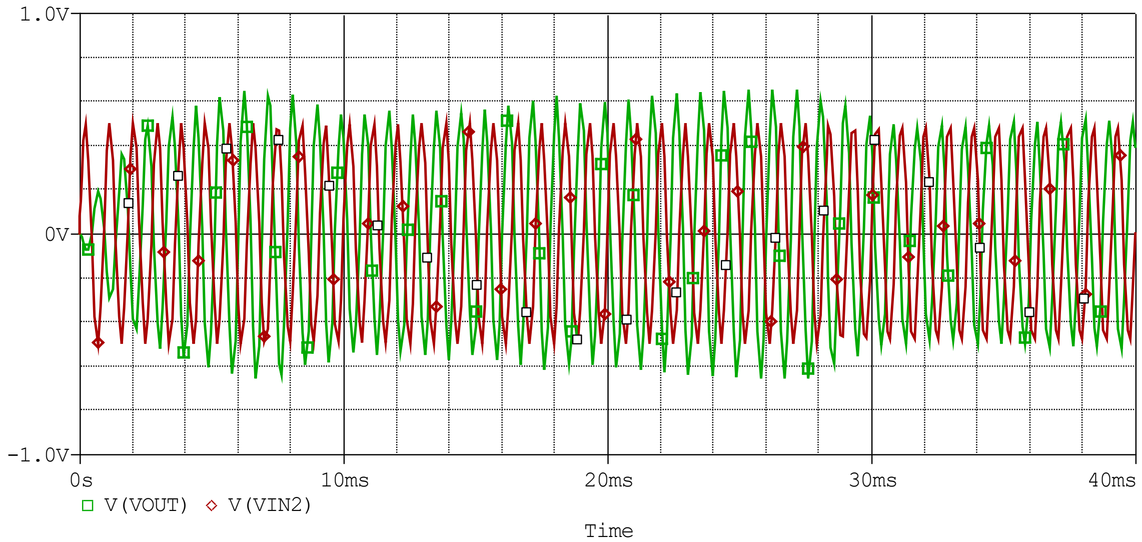


Figure 28 1100 Hz @ 0.5v frequency input test with +6 dB Gain

Zoomed in to view voltage levels:

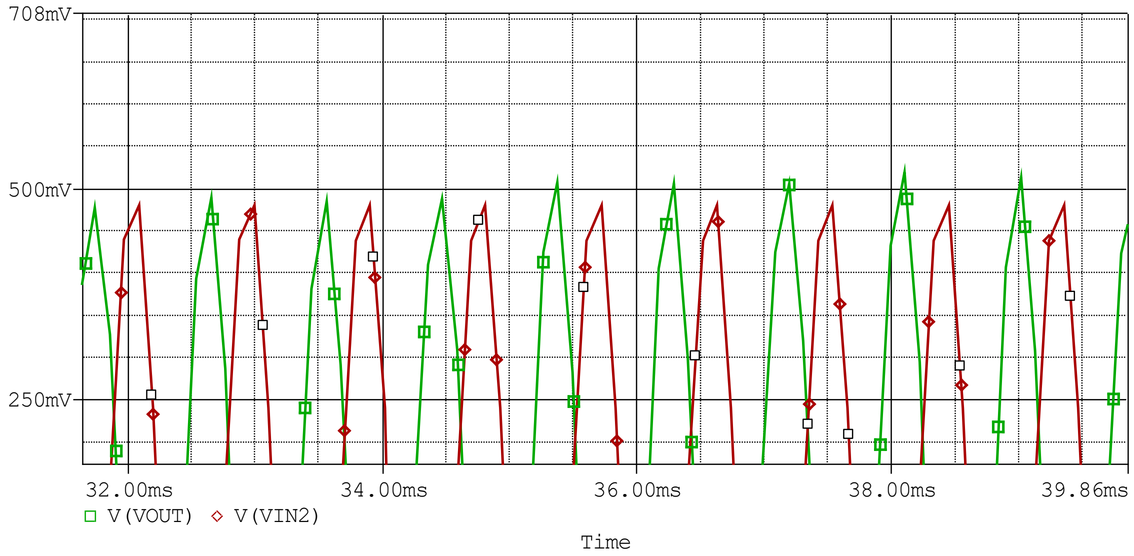


Figure 29 1100 Hz @ 0.5v frequency input test with +6 dB Gain zoomed in to view voltage levels

This part of the test shows the input and output voltages are at the approximately the same level which contradicts what we observed in figure 23 where we expect the output to be double.

Turning the voltage divider resistors into an equivalent resistor:

A Thévenin equivalent circuit can be used to replace a complex section of a circuit with an only a voltage source and resistor.

We will attempt to obtain a Thévenin equivalent circuit to model R1a & R1b which make up a voltage divider (see figure 30). The reason for doing this is to get an equivalent resistor value which can be used in the transfer function, which was derived without R1b (see figure 31) . We can then use the equivalent resistance of R1a & R1b for R1 in transfer function analysis.

R1b is the sweep resistor allowing to sweep the frequency range with a variable resistor.

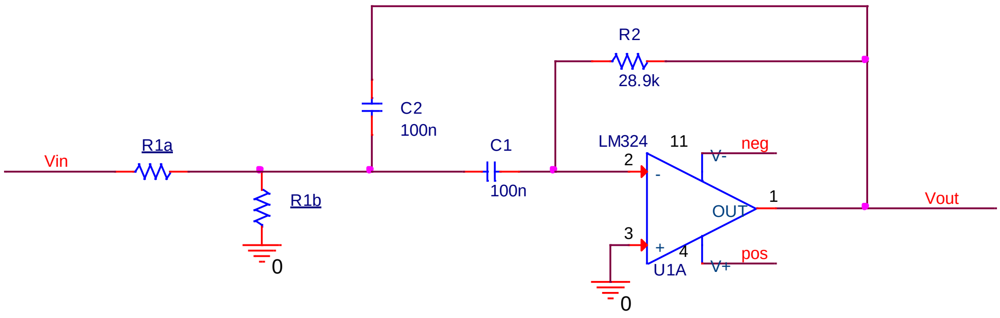


Figure 30 sweep capable multi-feedback BPF

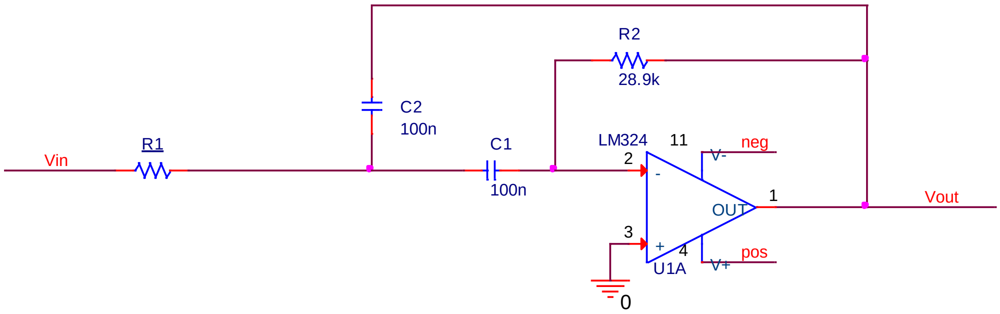
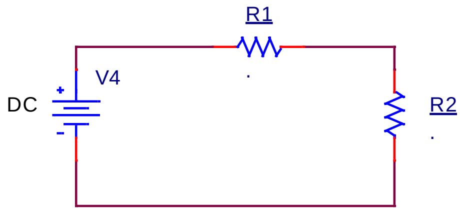


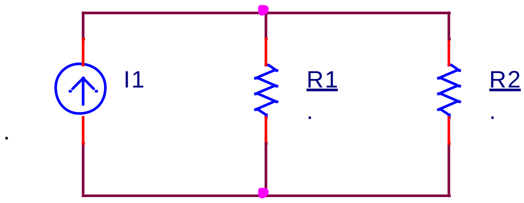
Figure 31 non sweep multi-feedback BPF where R1 is equivalent resistance to R1a & R1b from figure 30

Finding equivalent resistance of R1a & R1b:

<table>
<tr><th>Steps</th><th>state of circuit</th></tr>
<tr><td>Let R1 (from right side circuit) = R1a = 14468 ohms
& R2 (from right side circuit) = R1b = 72 ohms
DC = 0.5v</td><td>

DC = 0.5v
R1 = 14468 ohms
R2 = 72 ohms</td></tr>
<tr><td>Step 1: Transform a Voltage Source into a Current Source

A source transformation changes a voltage source in series with a resistor into a current source in parallel with a resistor. First, we need to find the current going through R1:
using ohms law:
I = VDC/ R1 = >0.5/ 14468 = 3.45e-5 amps

The circuit now becomes:
where the current source = 3.45e-5 amps.</td><td>

I = 3.45e-5 amps
R1 = 14468 ohms
R2 = 72 ohms</td></tr>
<tr><td>Step 2: Replace Parallel Resistors with an Equivalent resistor

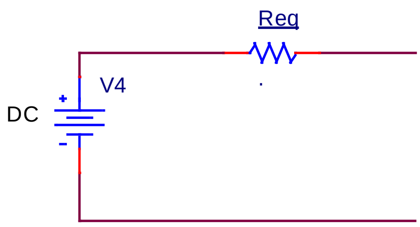
Doing a source transformation reveals a set of parallel resistors. Their equivalent resistance becomes:

Req = (R1*R2)/(R1+R2) => (14468*72)/(14468+72) = approx. 72 ohms</td><td>

I = 3.45e-5 amps
Req = 72 ohms</td></tr>
<tr><td>Step 3: Transform a Current Source into a Voltage Source

A source transformation is used again to change the current source back into a voltage source. Using ohms law, we can find the voltage to be:

V= I*Req => 3.45e-5 * 72 = 0.0025v

The circuit now becomes where VDC = 0.0025v & Req = 72 ohms.</td><td>

DC = 0.0025v
Req = 72 ohms</td></tr>
</table>

Conducting Pspice Parametric Sweep

A resistor sweep will allow us to observe the transfer response at different resistor values.

The circuit we will be testing is shown in figure 31 which is the one with the +6 dB gain at mid frequency.

To set up the sweep we need to create a global parameter with a name and value, then place the name into the component value box within "{ }" brackets.

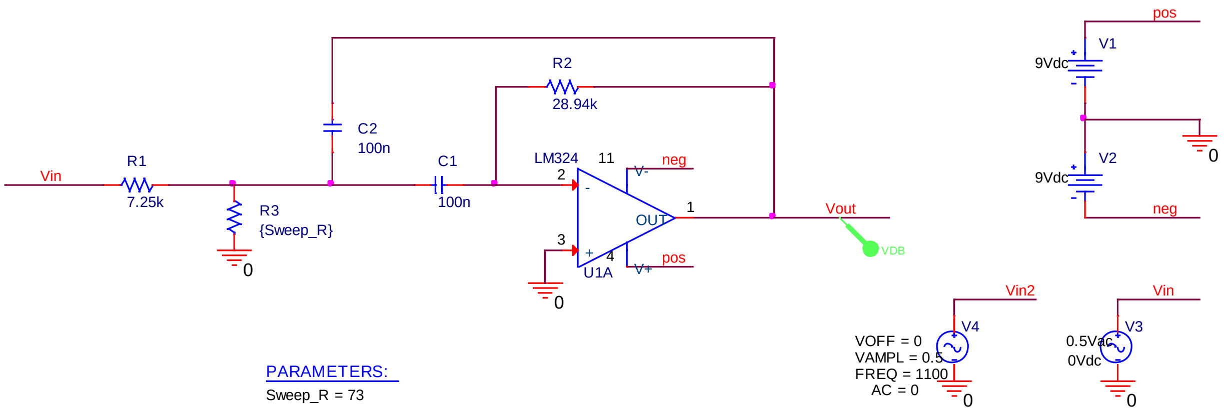


Figure 32 Pspice Resistor Sweep

The parametric sweep settings are to start from 1 to 1k ohms and increment in 10.

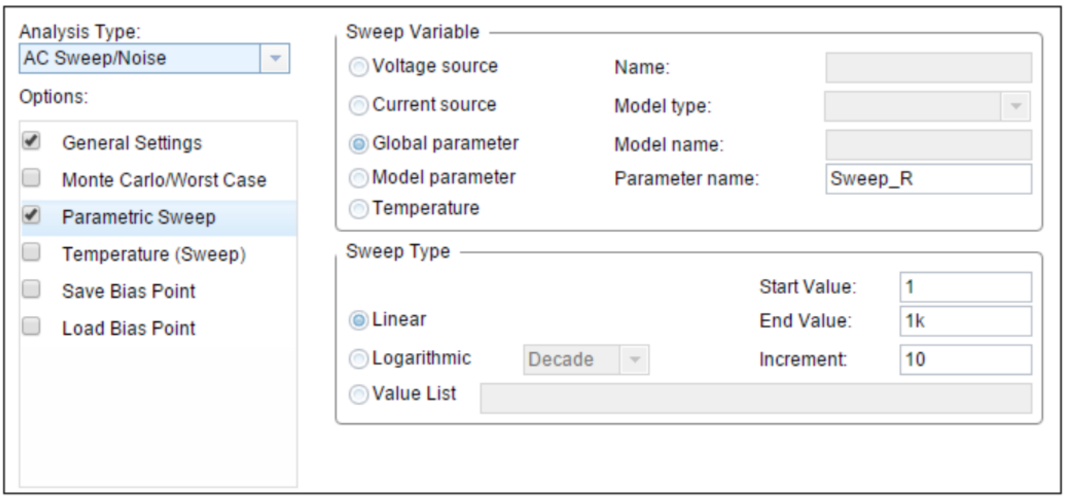


Figure 33 parametric sweep settings

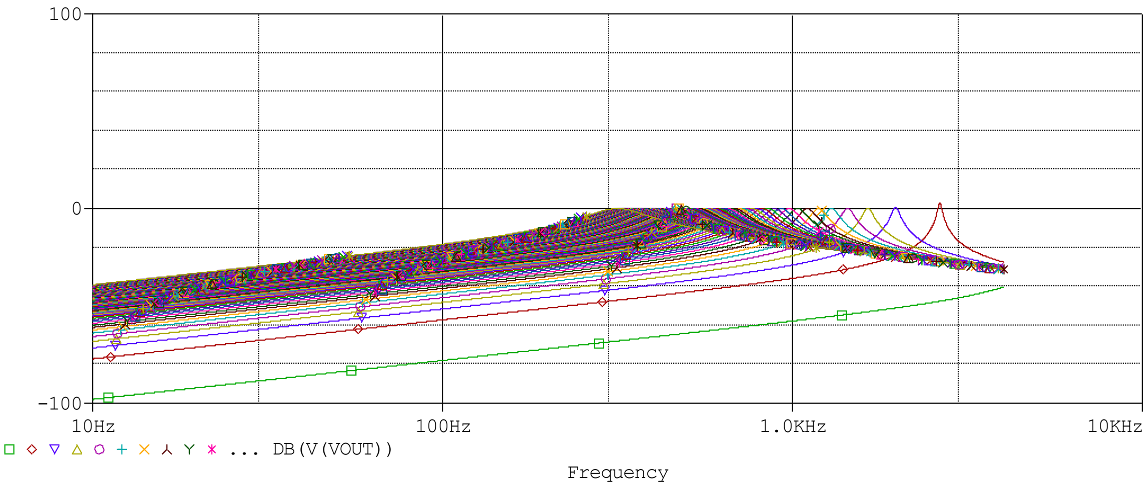


Figure 34 parametric sweep response

The range we design the system to work in is 200 – 2,000 hz, as observed from the plot in figure 29 the important factor to notice is that the gain at the mid frequency stays constant at 0 dB as the frequencies are changing.

We can observe just from the plot that at the lower frequencies the bandwidth is large and Q factor is going to be small which gets better as we increase to higher frequencies where the bandwidth is comparatively small and Q factor is going to be high.

Higher Q factor makes the Wah-Wah effect more pronounced.

The parametric sweep response behaviour was confirmed through mathematical calculations (figure 34) for Q and bandwidth (BW) for different values of R3 in figure 31.

	Fmid:	1100	Hz			
	C:	0.0000001	F	, is 100nF		
	R2:	28937.26	ohms			
	R1a:	14468.63	ohms			
R1b variable:					$Q = 0.5\sqrt{R_2/R_1}$ Req	
Increment		Freq (Hz)	R1b (ohms)	Req of R1a & R1b (ohms)	Q	BW (Hz)
100		200	2188.53	1900.99	1.95	563.88
		300	972.68	911.41	2.82	390.44
		400	547.13	527.20	3.70	296.95
		500	350.17	341.89	4.60	239.13
		600	243.17	239.15	5.50	200.00
		700	178.66	176.48	6.40	171.81
		800	136.78	135.50	7.31	150.55
		900	108.08	107.27	8.21	133.95
		1000	87.54	87.01	9.12	120.64
	fmid->	1100	72.35	71.99	10.02	109.73
		1200	60.79	60.54	10.93	100.63
		1300	51.80	51.61	11.84	92.91
		1400	44.66	44.53	12.75	86.30
		1500	38.91	38.80	13.65	80.56
		1600	34.20	34.12	14.56	75.54
		1700	30.29	30.23	15.47	71.10
		1800	27.02	26.97	16.38	67.16
		1900	24.25	24.21	17.29	63.63
		2000	21.89	21.85	18.19	60.46

Figure 35 Calculating Q, BW at different R1b values (also called R3)

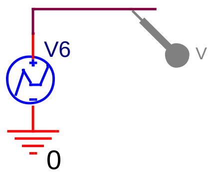
Trying to use audio file in Pspice

https://www.pspice.com/how-use-piece-wise-linear-pwl-voltage-and-current-sources

Using the mathlab code below we are able to convert a mp4 or wav file into a ".dat" file that Pspice can read.

```
[sample_data, sample_rate] = audioread('highv2.mp4');

sample_period = 1/sample_rate;
t = (0:sample_period:(length(sample_data)-1)/sample_rate);

t=t'; % Transpose
y = abs(y);
new_fileHighv2 = [t, sample_data]; % Save for Pspice
save C:\Users\sajja\Documents\guitar_recordings_\new_fileHighv2.dat -ascii

```

To test the .dat file we are using the **VPWL_F_RE_FOREVER** component.



FILE = C:\Users\sajja\Documents\guitar_recordings_\new_fileHighv2.dat
Value = VPWL_F_RE_FOREVER

An error pops us telling us the time values are not monotonically increasing.

```
+ REPEAT FOREVER
+ FILE "C:\Users\sajja\Documents\guitar_recordings_\new_fileHighv2.dat"
-------$
ERROR -- Time not monotonically increasing in File "new_fileHighv2.dat" Line 4
+ ENDREPEAT
.PARAM  sweep_r=73
```

Pspice expects the file contents to be in the format:	While contents of the file are actually in the format below, which is not in the format it expects:
0s 1V 10us 2V 20us 3V 22us 3V 100us 10V 110us 11V 120us 12V 200us 1V	new_fileHighv2 - Notepad File Edit Format View Help 5.0000000e+01 3.4580349e-04 1.1465715e-04 7.6474229e-05 2.5776830e-05 1.3408316e-05 2.6887727e-05 2.1465023e-05 7.0832307e-06 1.3807929e-05 1.0383179e-05 8.5317819e-06 5.1154586e-06 9.9298552e-06 6.9096121e-06 6.2493096e-06 3.2460674e-06 2.3864216e-06 1.5213958e-05

Success, by changing line 36 to line 38, it put the file into the correct format that Pspice expects.

```
36    save C:\Users\sajja\Documents\guitar_recordings_\new_fileHighv2.dat -ascii
37
38    save C:\Users\sajja\Documents\guitar_recordings_\new_fileHighv2.dat new_fileHighv2 -ascii
```

The mathlab code becomes:

```
[sample_data, sample_rate] = audioread('highv2.mp4');

N = (sample_rate/1)
sample_period = 1/(N);
t = (0:sample_period:(length(sample_data)-1)/N);

t=t'; % Transpose
y = abs(y);
new_fileHighv2 = [t, sample_data]; % Save for Pspice
save C:\Users\sajja\Documents\guitar_recordings_\new_fileHighv2.dat new_fileHighv2 -ascii
```

Output file from fixed code in correct format:

new_fileHighv2 - Notepad

File Edit Format View Help

```
0.0000000e+00   0.0000000e+00
2.2675737e-05   0.0000000e+00
4.5351474e-05   0.0000000e+00
6.8027211e-05   0.0000000e+00
9.0702948e-05   0.0000000e+00
1.1337868e-04   0.0000000e+00
1.3605442e-04   0.0000000e+00
1.5873016e-04   0.0000000e+00
1.8140590e-04   0.0000000e+00
2.0408163e-04   0.0000000e+00
2.2675737e-04   0.0000000e+00
2.4943311e-04   0.0000000e+00
2.7210884e-04   0.0000000e+00
2.9478458e-04   0.0000000e+00
3.1746032e-04   0.0000000e+00
3.4013605e-04   0.0000000e+00
3.6281179e-04   0.0000000e+00
```

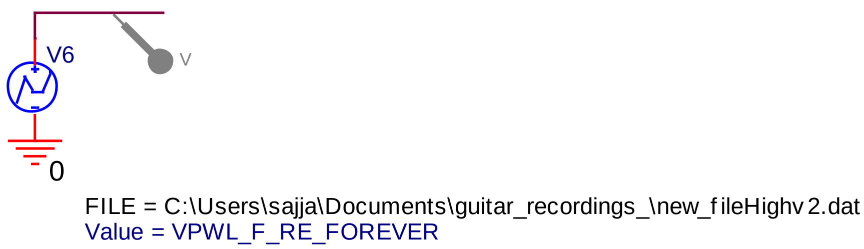
Audio file test: to try and get Pspice to read a wav/mp3 file

We changed the sampling rate from 44.1Khz to 11Khz because it takes very long to process in mathlab. We did this by dividing the sampling rate of the file by an integer say 4. Since the highest frequency component our system will see if 2k Hz we must ensure the sampling rate is not below 4 k Hz which is the Nyquist rule. The Nyquist rate specifies the minimum rate at which to sample a signal in order to faithfully represent the signal.

This is done on line 29 where we divide the sample rate by the desired scale.

```
27 -    [sample_data, sample_rate] = audioread('highv2.mp4');
28
29 -    N = (sample_rate/4)  % reducing sampling rate
30 -    sample_period = 1/(N);
31 -    t = (0:sample_period:(length(sample_data)-1)/N);
32
33 -    t=t'; % Transpose
34 -    y = abs(y);
35 -    new_fileHighv2 = [t, sample_data]; % Save for Pspice
36 -    save C:\Users\sajja\Documents\guitar_recordings_\new_fileHighv2.dat new_fileHighv2 -ascii
```

Pspice reading in a wav file:



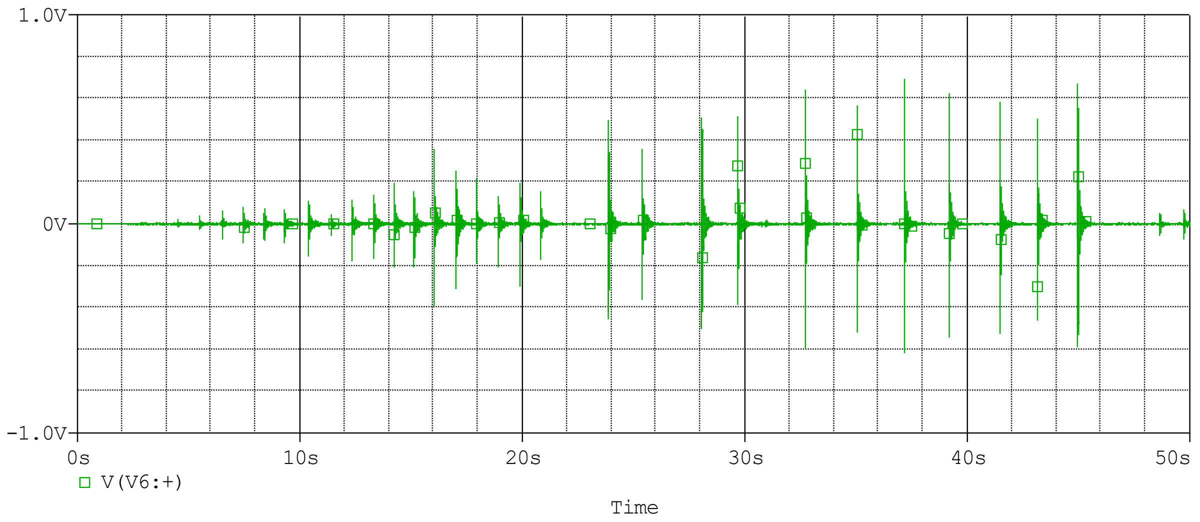


Figure 36 Success of reading in an audio file

Simulating the BPF on a guitar audio signal in PSpice

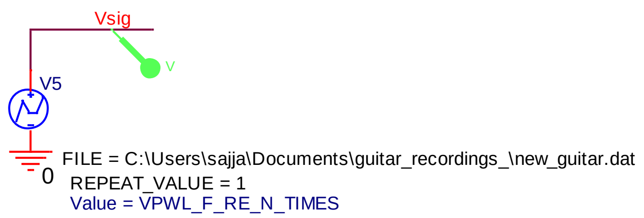
Step 1: Download guitar wav file from online: https://www.findsounds.com/ISAPI/search.dll?keywords=guitar

Step 2: If the audio is **stereo**, convert to **mono** using:

https://audio.online-convert.com/convert-to-wav

Step 3: Use mathlab to convert the file into a .dat file for Pspice

Step 4: Open the project in Pspice and copy the same file directory used in mathlab into the Pspice component for reading the audio file.



The signal with an FTT applied:

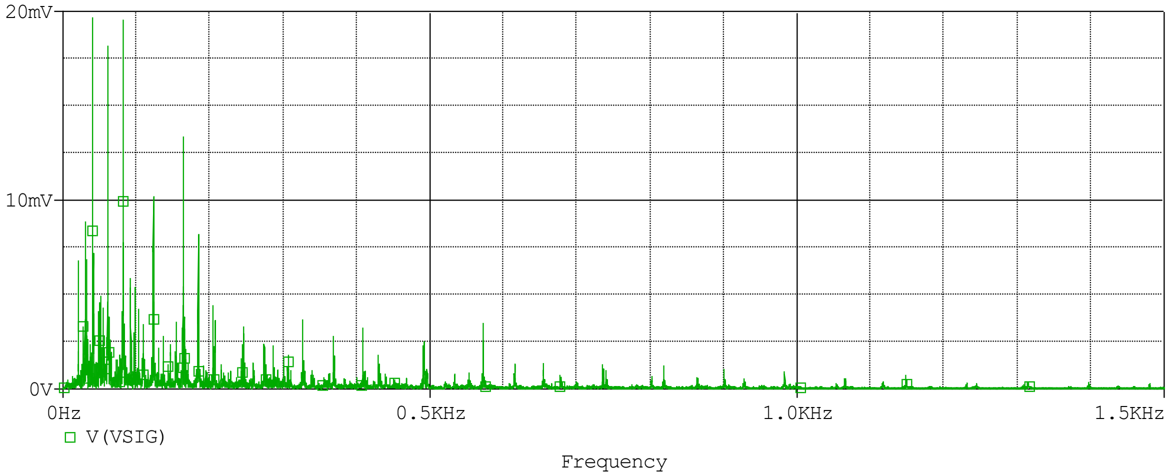


Figure 37 Guitar Audio FFT

The signal is made up of very little frequency components beyond the 0.5K Hz range.

We cannot use the bandpass filter with a centre frequency of 1100Hz as its effect would be imprescriptible.

If we zoom into the plot between 0-400Hz we observe plenty of frequency components in this range so if we apply the BPF with a mid-frequency of 200Hz we should be able to observe the FFT of the output and compare with this original input FFT.

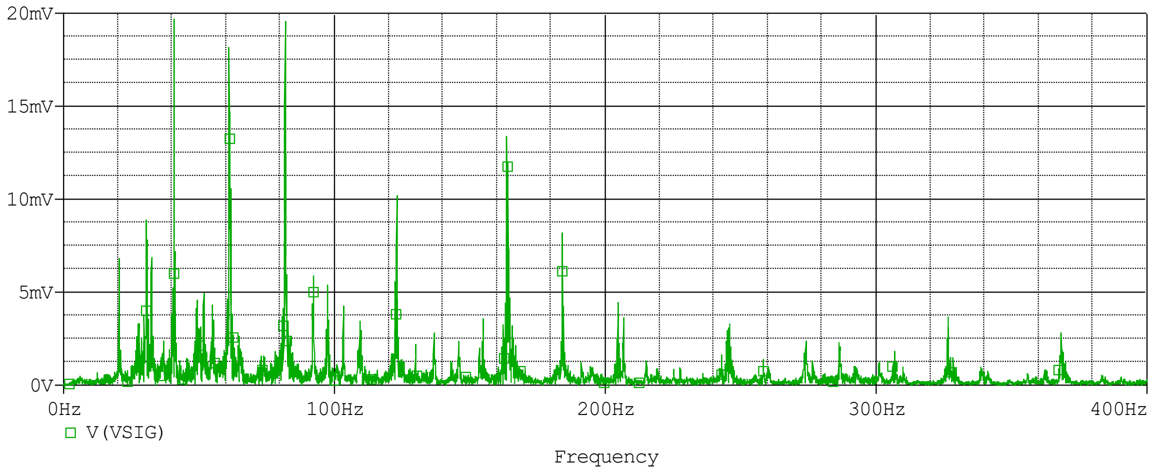
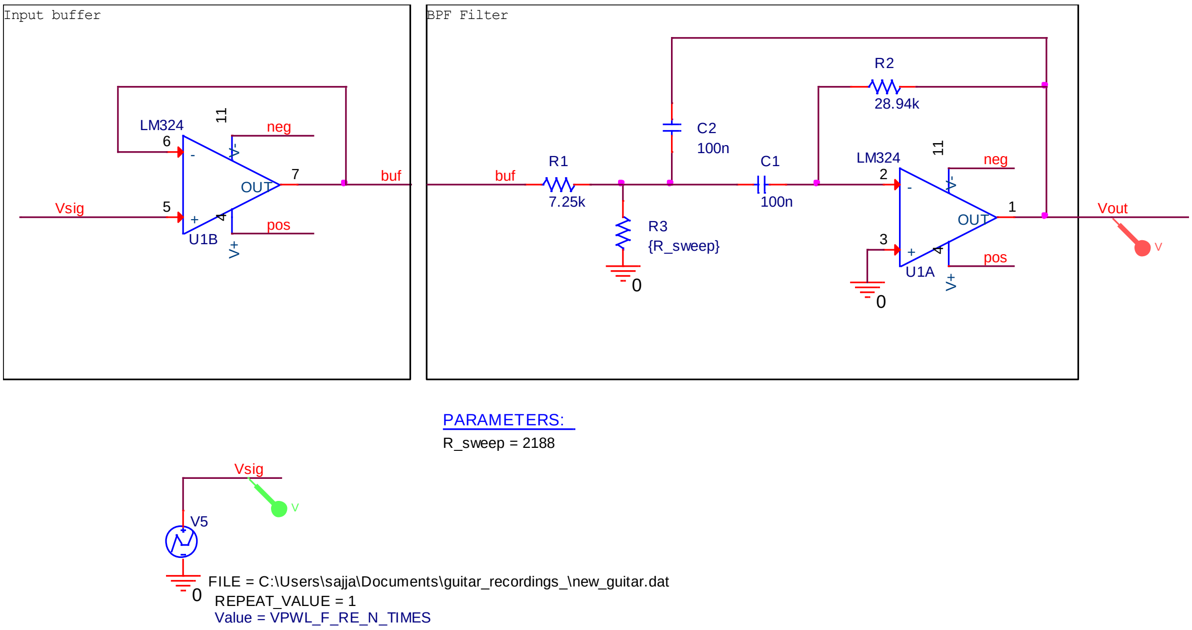


Figure 38 FFT of audio file

Circuit used:



Input and output plots in time domain:

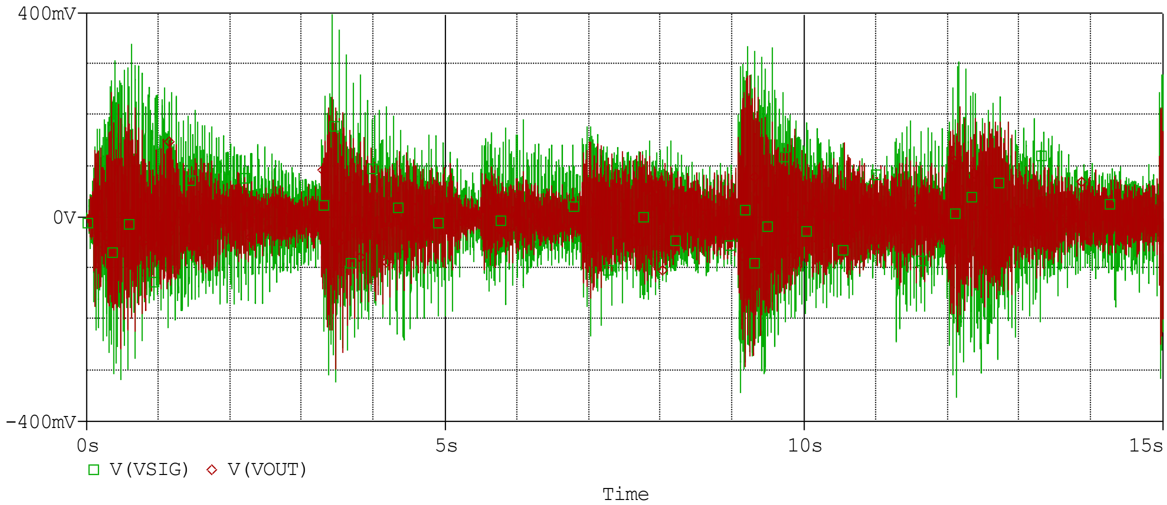


Figure 39 Guitar signal input and output of system

Observing figure 39, there has indeed been an effect applied to the input signal. If we turn on FFT we can check if the effect is the intended one.

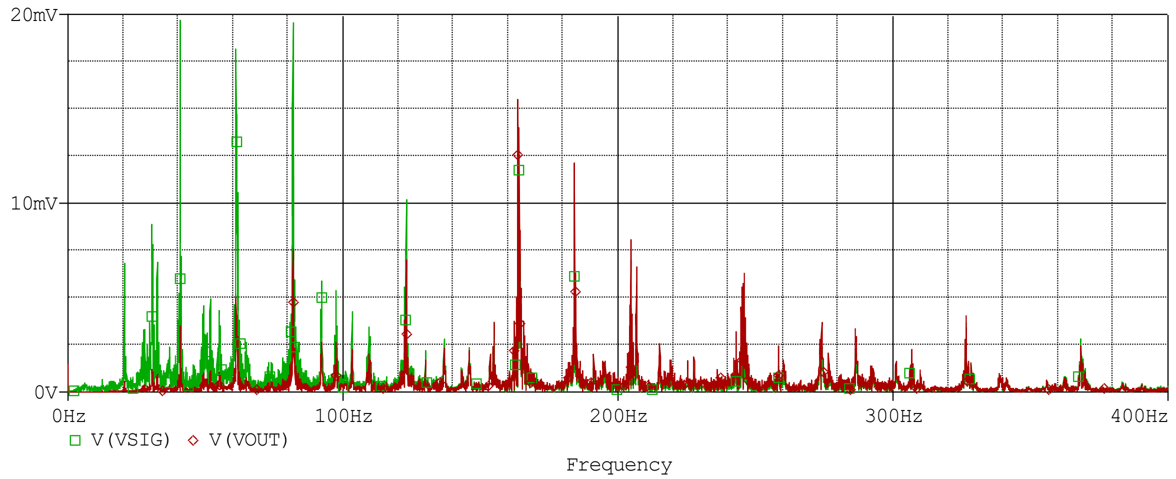


Figure 40 FFT Guitar signal input and output of system

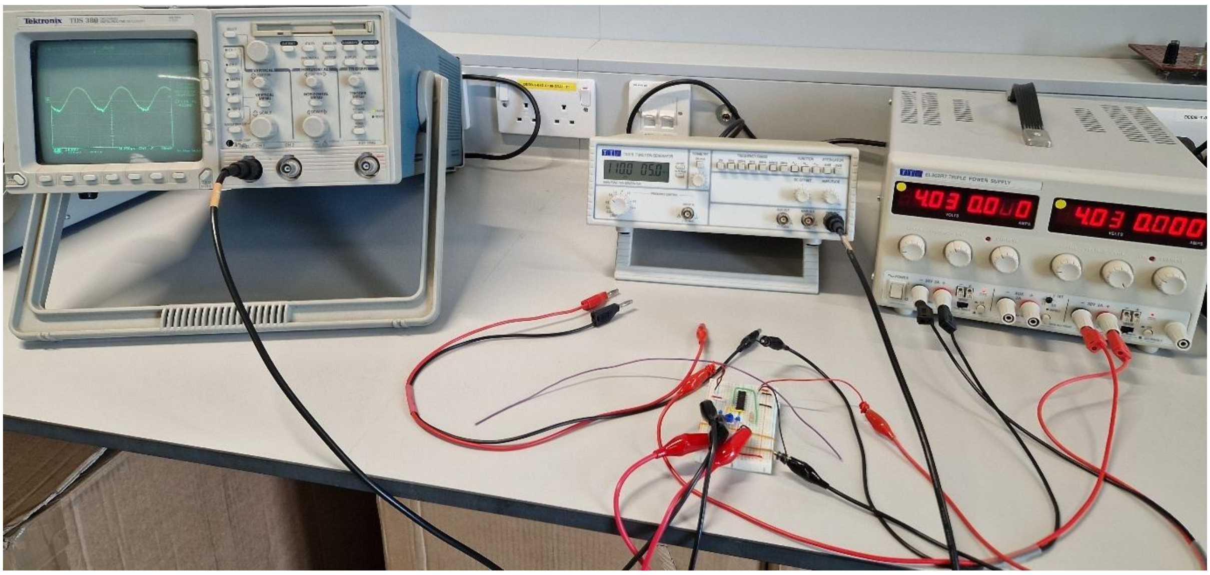
The centre frequency was 200Hz, frequencies above and below this are getting attenuated as intended.

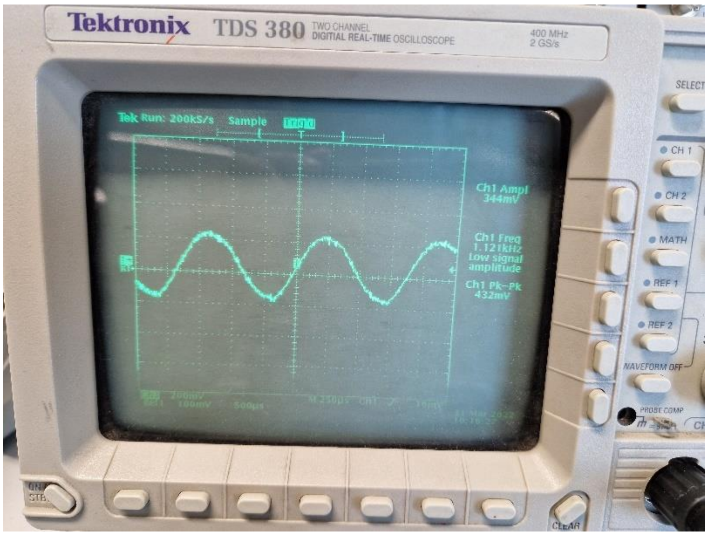
First hardware test

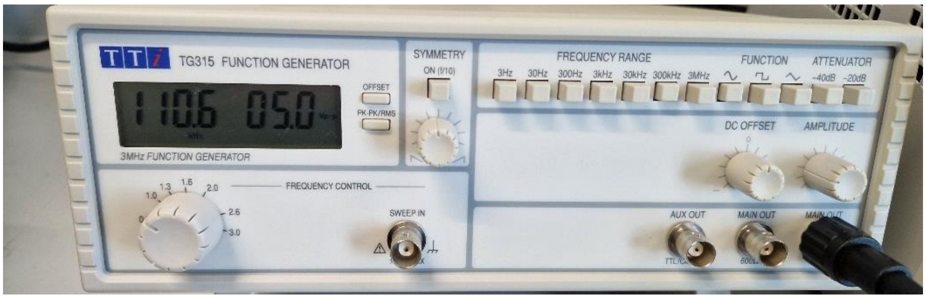
It was connected to +/- 4v DC power source.

Signal generator was set to 1100Khz @ 5.0v p-p

Note this was a mistake as it should have been set to 1100Hz @ 1vp-p (0.5v peek).







The signal source was later fixed with my supervisor who pointed out the misstep to the correct value of 1100Hz @ 1vp-p .

First the signal generator was connected into the oscilloscope to verify the frequency and amplitude of the signal.

Then we connected the signal to the circuit input and the oscilloscope to the output.

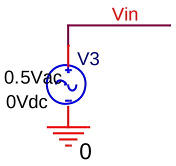
It appeared to be working with the output signal on the oscilloscope showing 1.1Khz @ 0.432v p-p which matches the input signal very closely with a little drop in voltage of 0.068v from the original 0.5v p-p. As the frequency is the resonant frequency of 1100Hz we expect no voltage drop and the signal to pass through unaltered.

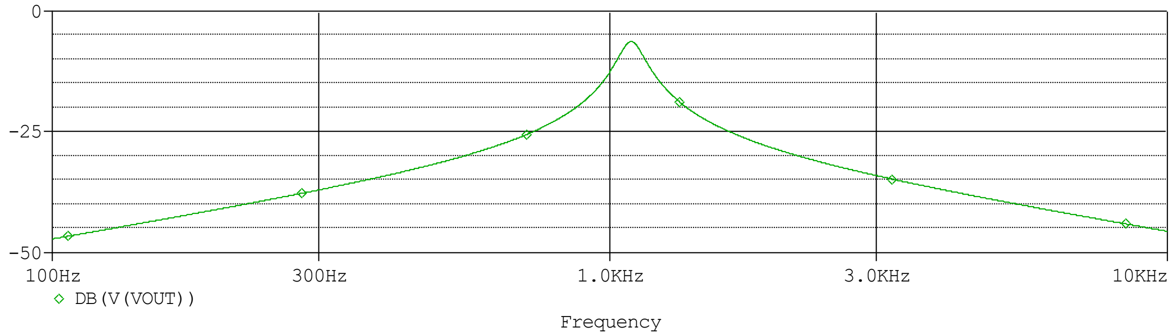
An issue arose when the signal generator was disconnected, and we could still see a 1100Hz waveform at the output.

The supervisor explained it could be noise present in the circuit that when checking the output with no input, the filter filtered the noise and allowed the resonate frequency of 1100Hz to appear at the output.

A sudden mistake discovered that I was doing this whole time:

In simulation, when doing frequency sweep, I set the AC source to 0.5v AC because the max voltage the system will encounter is 0.5v peek. and this produces:

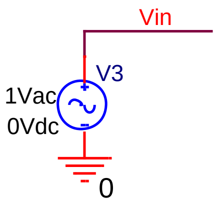


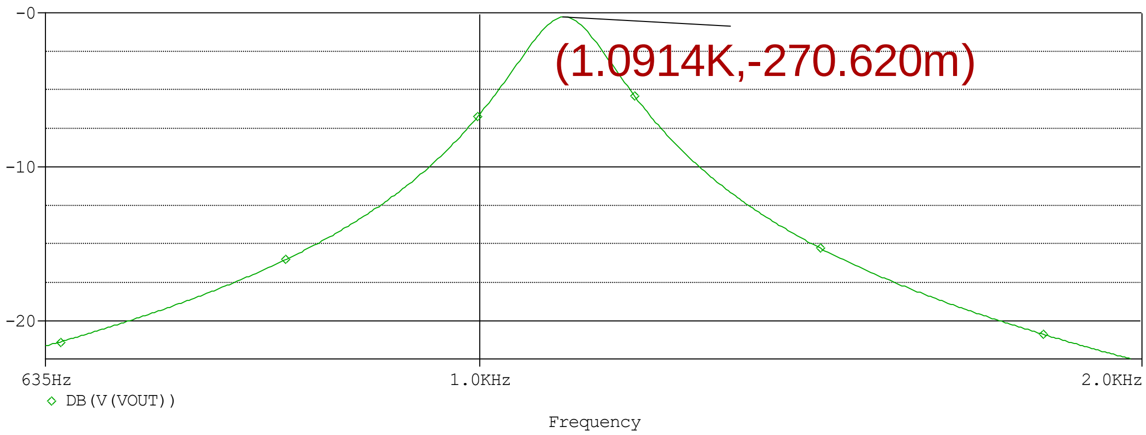


Which is -6 dB at the resonate frequency of 1100 Hz when it was designed to be 0dB at this value.

This was wrong to do.

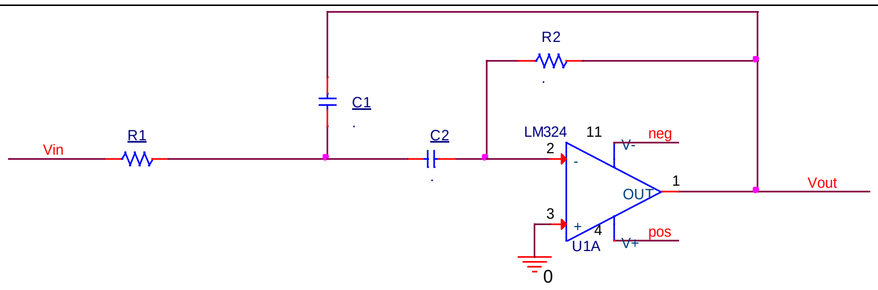
Putting this Vac to 1 instead of 0.5 yeilds:





Now this is the correct behaviour that we expected from the very start.

Fixing the transfer function derivation



$$Node\ Vx: \frac{Vx - Vi}{R1} + \frac{Vx - 0}{1/sC2} + \frac{Vx - Vo}{1/sC1} = 0$$

$$\frac{Vx}{R1} - \frac{Vi}{R1} + sC2\,Vx + sC1\,Vx = sC1\,Vo$$

$$\frac{Vx}{R1sC1} - \frac{Vi}{R1sC1} + \frac{sC2Vx}{sC1} + \frac{sC1Vx}{sC1} = Vo$$

$$-\frac{Vi}{R1sC1} + Vx[\frac{1}{R1sC1} + \frac{sC2}{sC1} + 1] = Vo \qquad , Label\ this\ "A"$$

$$Node\ Vy: \frac{0 - Vx}{1/sC2} + \frac{0 - Vo}{R2} = 0$$

$$-sC2Vx = \frac{Vo}{R2}$$

$$-sC2VxR2 = Vo \qquad , Label\ this\ "B"$$

$$A: \ Vo = -\frac{Vi}{R1sC1} + Vx[\frac{1}{R1sC1} + \frac{sC2}{sC1} + 1]$$

$$B: \ Vo = -sC2VxR2$$

manipulate A & B into a form for use with Cramer's rule:

$$A: \ -Vx\,[\frac{1}{R1sC1} + \frac{sC2}{sC1} + 1] + Vo = -\frac{Vi}{R1sC1}$$

$$B: \qquad Vx[sC2R2] + Vo = 0$$

Now apply Cramer's rule:

$$Vo = \frac{determinate\ of \begin{vmatrix} [\frac{1}{R1sC1} + \frac{sC2}{sC1} + 1] & -\frac{Vi}{R1sC1} \\ [sC2R2] & 0 \end{vmatrix}}{determinate\ of \begin{vmatrix} [\frac{1}{R1sC1} + \frac{sC2}{sC1} + 1] & 1 \\ [sC2R2] & 1 \end{vmatrix}}$$

Do Numerator first:

$$\{\ [\frac{1}{R1sC1} + \frac{sC2}{sC1} + 1]\ *\ [0]\ \} \quad - \quad \{\ [sC2R2] * [-\frac{Vi}{R1sC1}]\ \} = \frac{Vi\, sC2R2}{sC1R1}$$

Denominator is:

$$\{ [\frac{1}{R1sC1} + \frac{sC2}{sC1} + 1]\ *\ [1]\ \} \quad - \{ [sC2R2] *\ [1] \} = \frac{1}{R1sC1} + \frac{sC2}{sC1} + 1 - sC2R2 \}$$

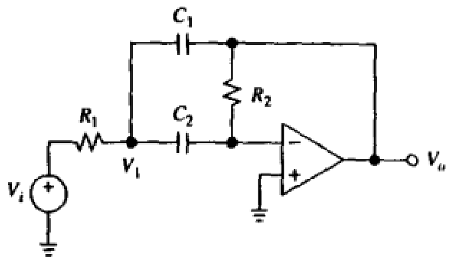
Now Vo/Vi becomes:

$$Vo = \frac{\frac{Vi\, sC2R2}{sC1R1}}{\frac{1}{R1sC1} + \frac{sC2}{sC1} + 1 - sC2R2}$$

$$Vo = \frac{Vi\, sC2R2}{\frac{sC1R1}{R1sC1} + \frac{sC2sC1R1}{sC1} + sC1R1 - sC2R2sC1R1} = \frac{Vi\, sC2R2}{1 - s^2C1C2R1R2 + sR1[C1 + C2]}$$

$$\frac{Vo}{Vi} = \frac{sC2R2}{1 - s^2C1C2R1R2 + sR1[C1 + C2]}$$

This transfer function is correct as we can compare with the transfer function given by Sergio Franco in the book “Design with Operational Amplifiers and analog integrated circuits, pg 141” where it is:



$$H(j\omega) = \frac{V_o}{V_i} = \frac{-j\omega R_2C_2}{1 - \omega^2 R_1 R_2 C_1 C_2 + j\omega R_1(C_1 + C_2)}$$

The only difference is the negative symbol at the top of the fraction which is explained by the fact we are using the inverting input of the op-amp whose output will be negative.

Note that this circuit does not have the third resistor connected to R1 to form a voltage divider.

The transfer function we have derived does not take into account of R1a & R1b, the latter of which is used to sweep the frequency.

R1a & R1b is simply a voltage divider at the input of the circuit therefore:

Vout = Vin * voltage divider * transfer function

$$Vo = \frac{-Vi * \frac{R1b}{R1a + R1b} * sC2R2}{1 - s^2 C1C2R1R2 + sR1[C1 + C2]}$$

$$\frac{Vo}{Vi} = \frac{-s\frac{R2R1b}{R1a + R1b}C2}{1 - s^2 C1C2R1R2 + sR1[C1 + C2]}$$

Let C1 = C2 = C

$$\frac{Vo}{Vi} = \frac{-s\frac{R2R1b}{R1a + R1b}C}{1 - s^2 R1R2C^2 + s\,2R1C}$$

Lastly the equivalent resistance of R1a & R1b is equal to R1 therefore we should substitute in R1a & R1b into the transfer function.

R1 = (R1a*R1b)/(R1a+R1b)

Replace R1 with (R1a*R1b)/(R1a+R1b) to give:

$$\frac{Vo}{Vi} = \frac{-s\frac{R2R1b}{R1a + R1b}C}{1 - s\frac{2\,R1a\,R1b}{R1a + R1b}C + s^2\frac{R2R1aR1b}{R1a + R1b}C^2}$$

Mathlab Bode plots from transfer function

Now we need to get bode and unit circle plots from mathlab.

To do this we use the code in figure 38 where the mathlab function *bode* is used.

```
C = 100e-9; %100nF
R2 = 28.94e+3;  % kohms
R1a = 14.47e+3; % kohms
R1b = 72;

S_Numerator = -(R2*R1b*C)/(R1a+R1b);
S = (2*R1a*R1b*C)/(R1a+R1b);
S2 = ((R2*R1a*R1b*C^2)/(R1a+R1b));


b = [ S_Numerator 0 ];
a = [S2 S 1 ];


H = tf(b,a)
figure(3)
bode(H)
```

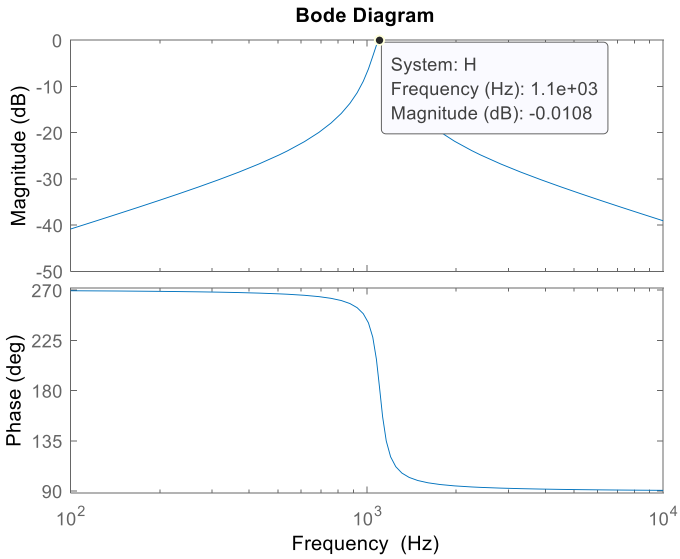
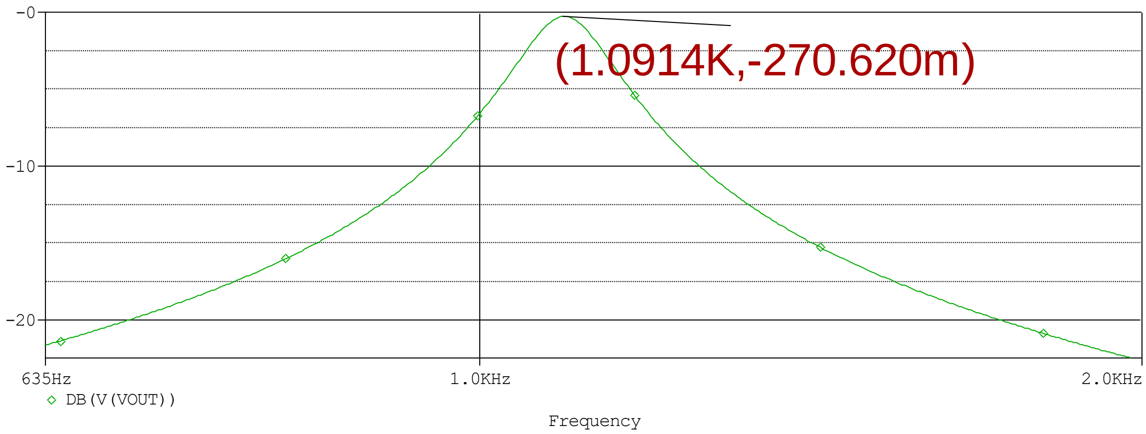


Figure 41 mathlab code & bode plots from transfer function using exact components

We can compare the bode frequency response with response from Pspice (shown below) to observe that they both agree.



Frequency response using measured component resistance

Using measured R2 & R1a resistance values in the simulation:

When measured with a multimeter the resistance of R2 is 33k ohms & R1a is 14.72k ohms.

Putting these values into the simulation to view a more accurate behaviour of the system:

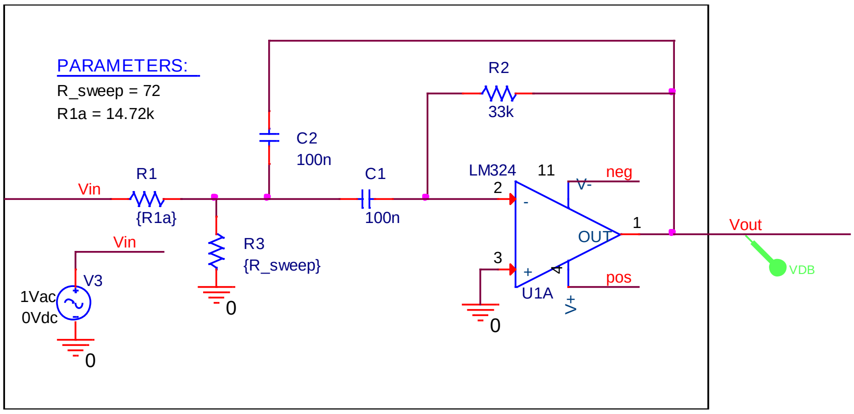


Figure 42 Circuit with measured resistance values

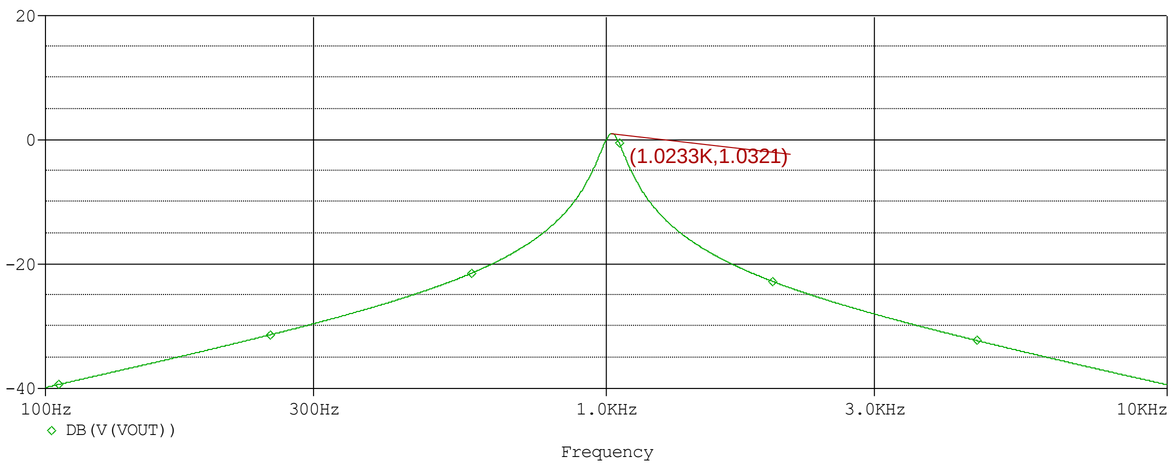


Figure 43 Response with measured resistance values

Figure 40 shows that the system with the resistor values measured in the circuit to have a gain of 1 dB.

Updating the values in the mathlab code for the measured R2 & R1a:

```
C = 100e-9; %100nF
R2 = 33e+3;  % kohms
R1a = 14.72e+3; % kohms
R1b = 72;

S_Numerator = -(R2*R1b*C)/(R1a+R1b);
S = (2*R1a*R1b*C)/(R1a+R1b);
S2 = ((R2*R1a*R1b*C^2)/(R1a+R1b));

b = [ S_Numerator 0 ];
a = [S2 S 1 ];

H = tf(b,a)
figure(1)
bode(H)
```

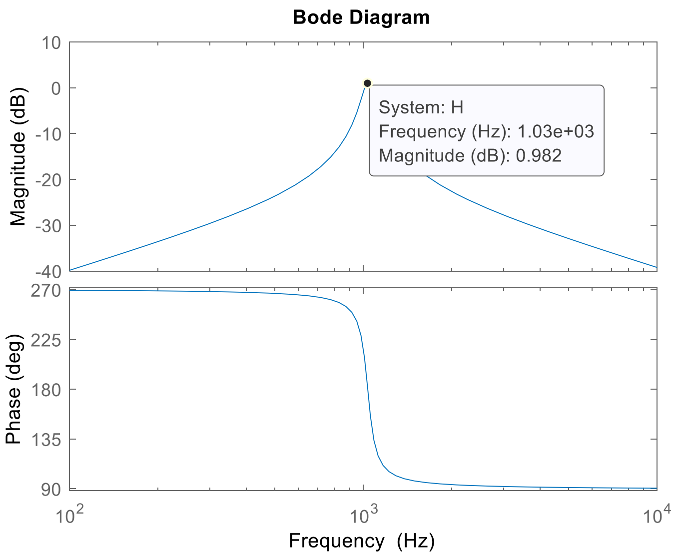


Figure 44 mathlab code & bode plots from transfer function using measured values

Mathlab validates the Pspice simulation as it also shows a gain of almost 1 dB also.

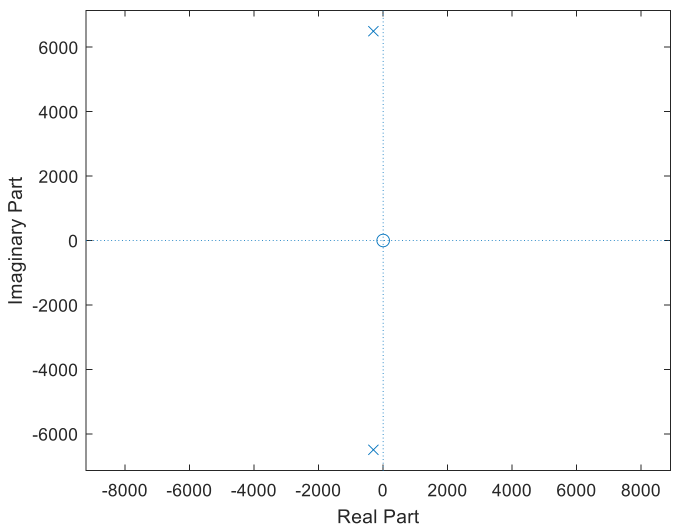
There is a little bit of gain in the system of approx. 1dB therefore:

20 log(Pout/Pin) = 1 dB

Pout/Pin = 10^ (1/20)

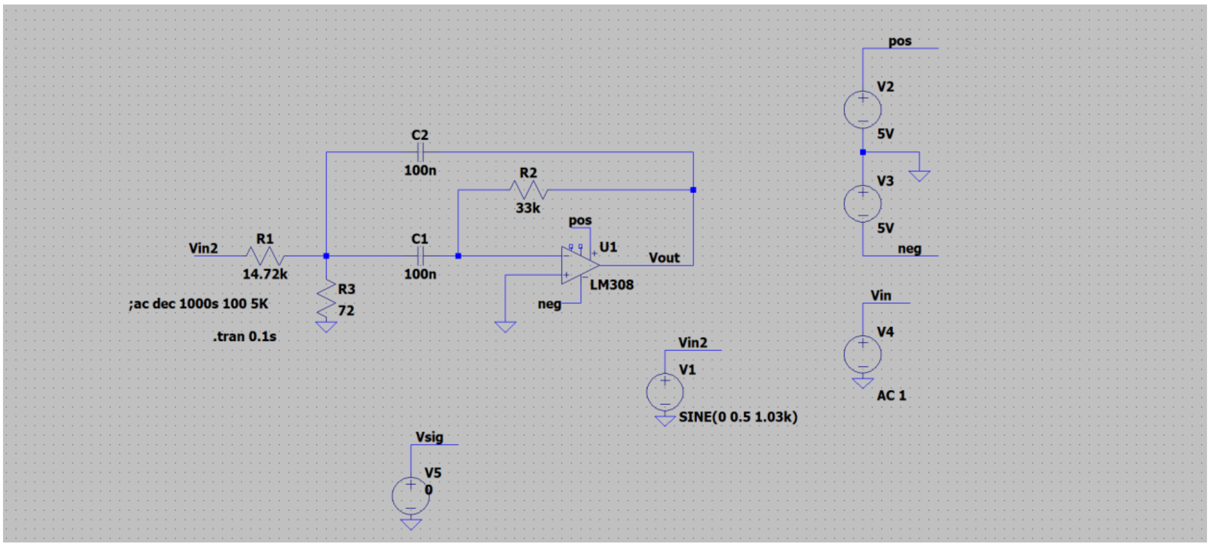
Pout/Pin = 1.122

If Pin = 0.5v the Pout will be 0.56v

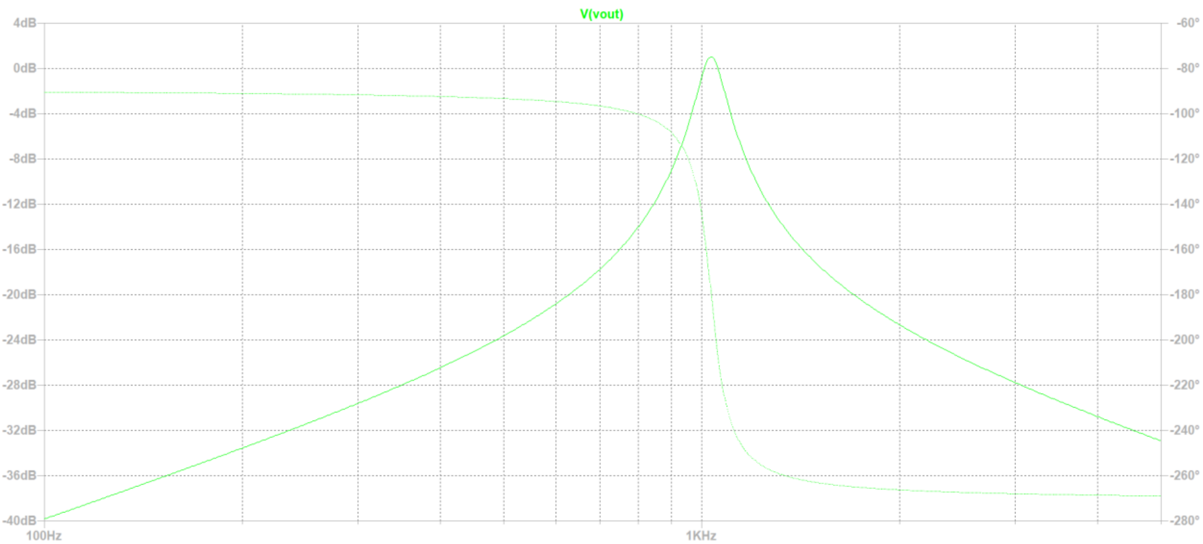


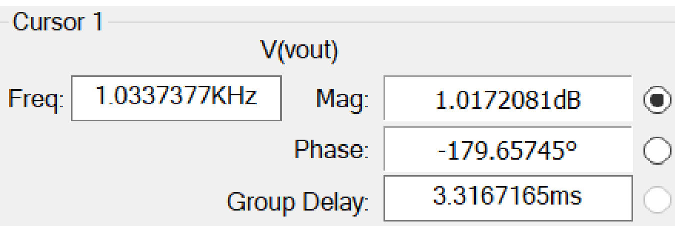

```
zeros =

     0


poles =

   1.0e+03 *

  -0.3030 + 6.4963i
  -0.3030 - 6.4963i


H =

          -1.606e-05 s
  ------------------------------
  2.364e-08 s^2 + 1.433e-05 s + 1

Continuous-time transfer function.
```

Building the circuit in LTspice:



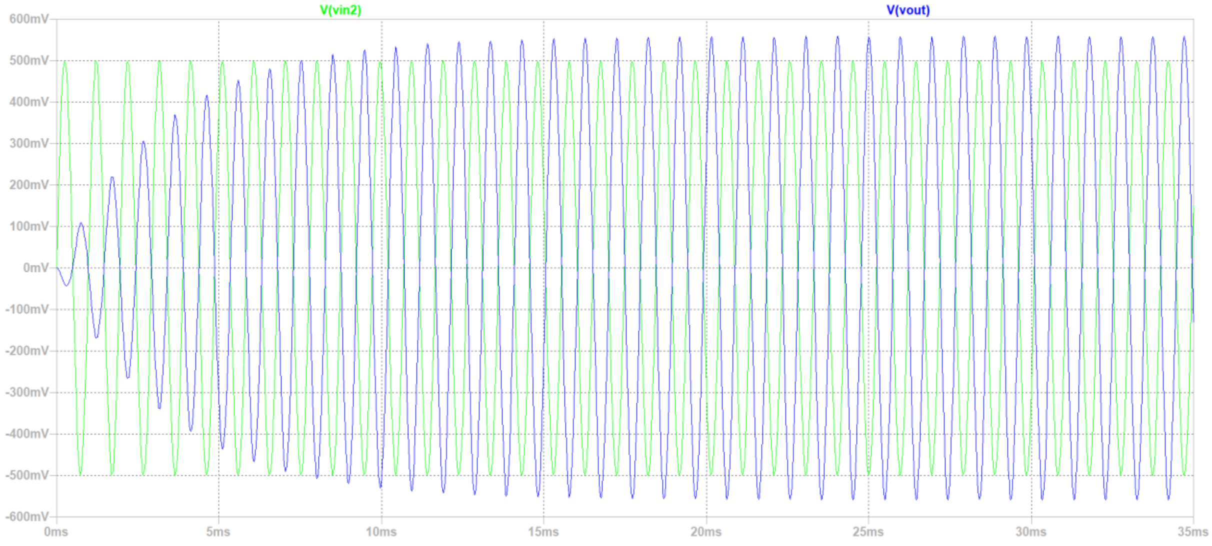
Frequency response:





Using cursor to get the magnitude & resonate frequency shows 1.03 kHz at 1.01 dB, further confirming the previous simulations.

LTspice simulation Frequency at 1.03kHz @ 0.5v

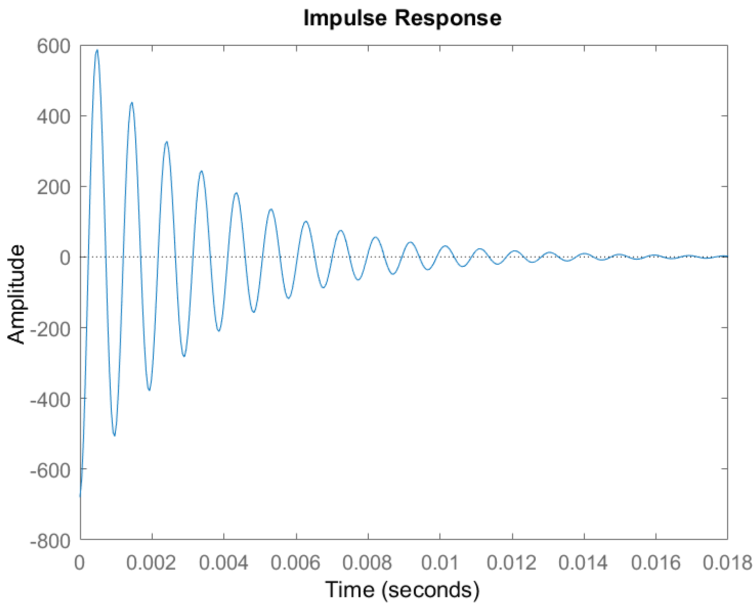


Reflection on the week

A lot of progress was made this week, we now have a simulation and ability to use audio files. The first hardware test showed some difficulty but were later solved with in depth analysis into how the problem could be formed.

Minutes from meeting

All simulations and MATLAB plots agree on the circuit behaviour, I need to find out why there is an output present when no input signal is connected as seen during the hardware test.



Week 10: & simulation of audio file as input and output on LTspice

Objective: To use LT Spice to read in a wav file as the input to the filter then save the output to a wav file, for playback of the effect.

Update: The issue of an output signal being present when no input signal is connected to the circuit was due to the variable resistor being misaligned.

To read in the audio file using a voltage source component, open its attribute editor, in the value section use the syntax: **wavefile=<filename.wav>**

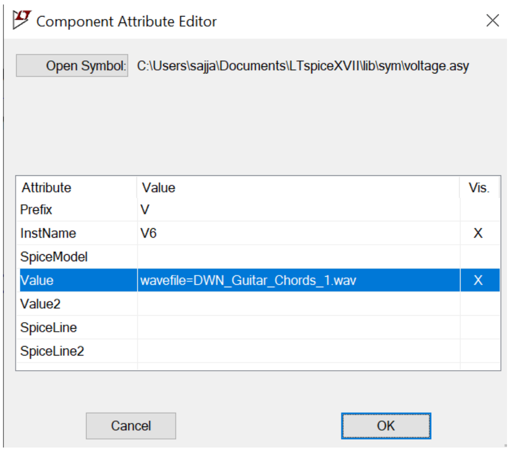


Figure 45 LTSpice voltage source component using wav file as input

The syntax to output individual node voltages as a .wav file is:

.wav <filename.wav> <Nbits> <SampleRate> V(out)

If we want to play the output .wav file your PC sound card, keep in mind that the more popularly supported .wav file formats have 1 or 2 channels, typically at 16 bits/channel and a sample rate of 44100 Hz for consumer audio such as CDs. We will use 16 bits and 44.1kHz to write our wave file.

Using the 'Spice Directive' button on the tool bar we put in the syntax to save the voltage at node *Vout* into a wav file.

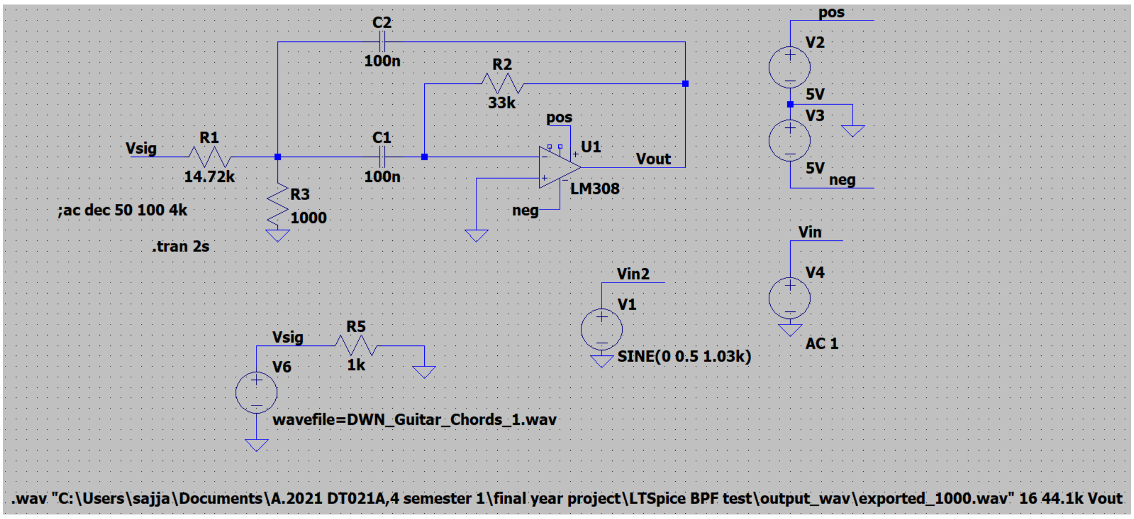
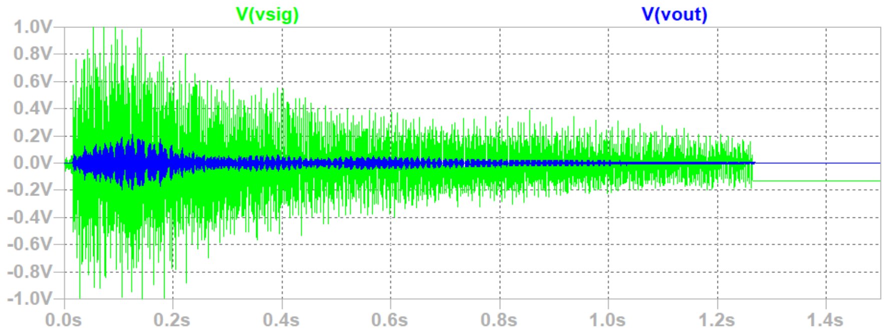
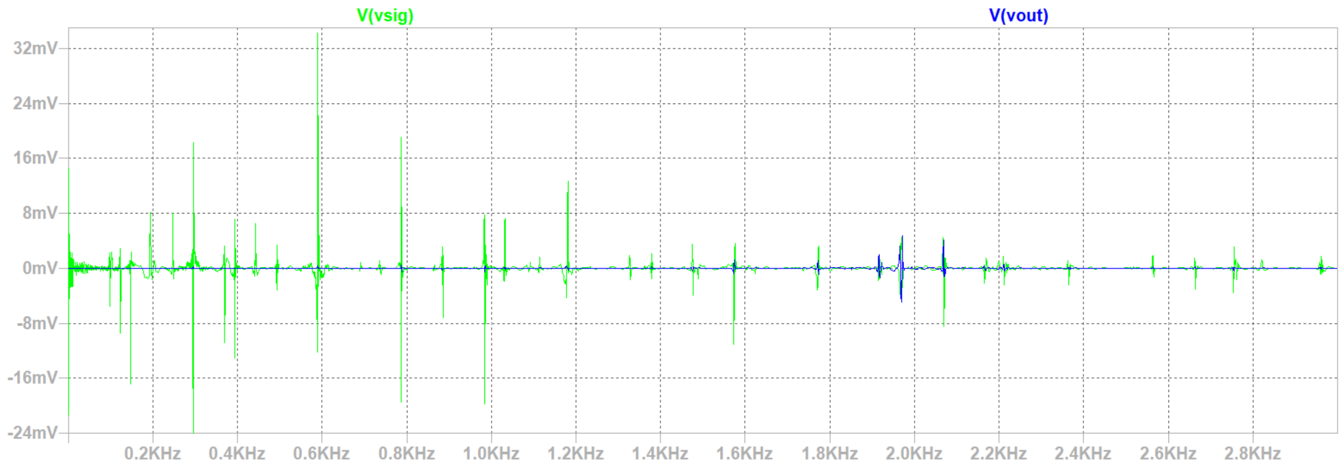


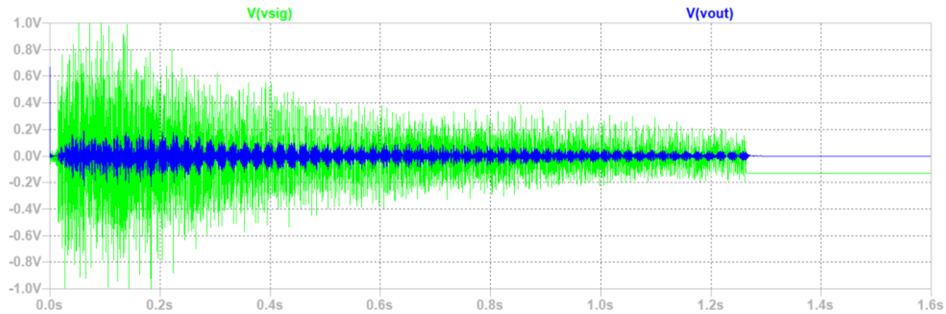
Figure 46 Audio file input and output on LTSpice

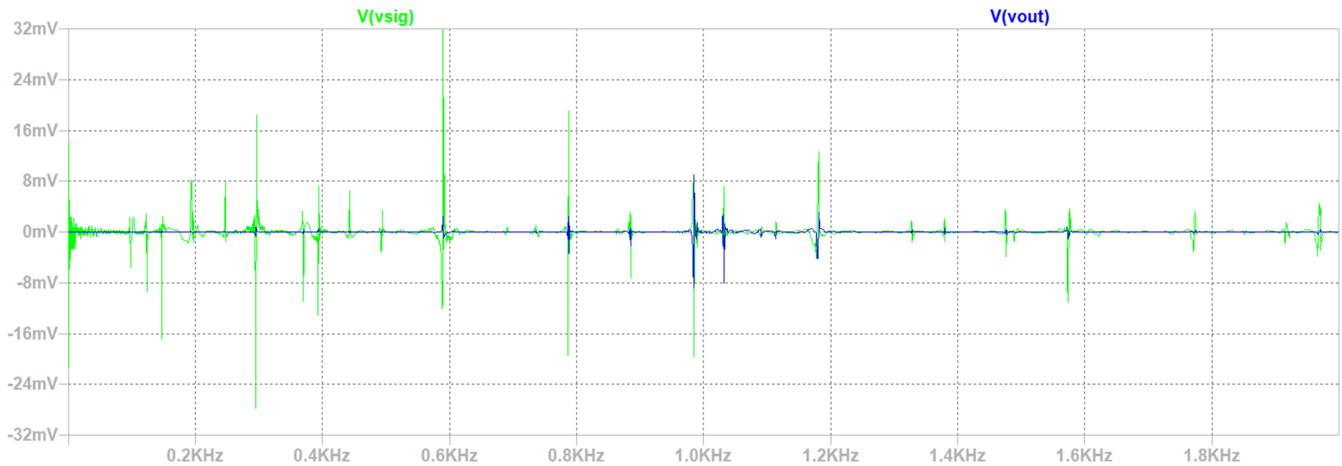
R3 at 20 ohms, resonate frequency = 1960 Hz



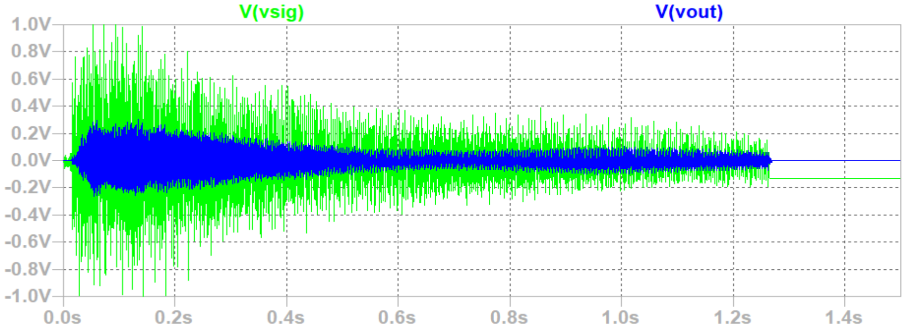


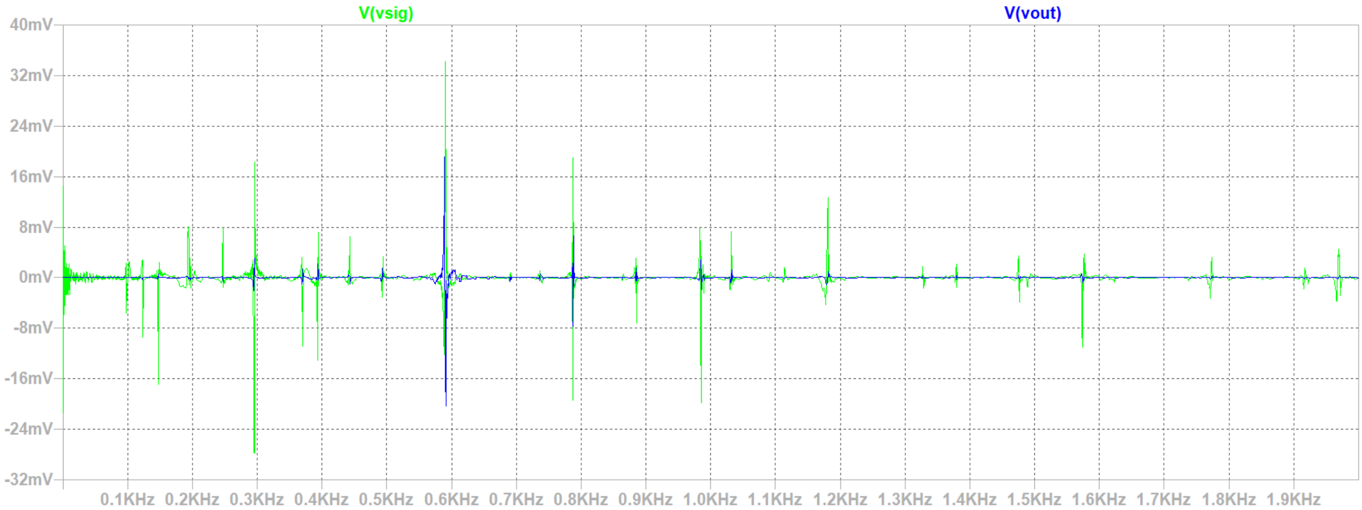
R3 at 70 ohms, resonate frequency = 1047 Hz



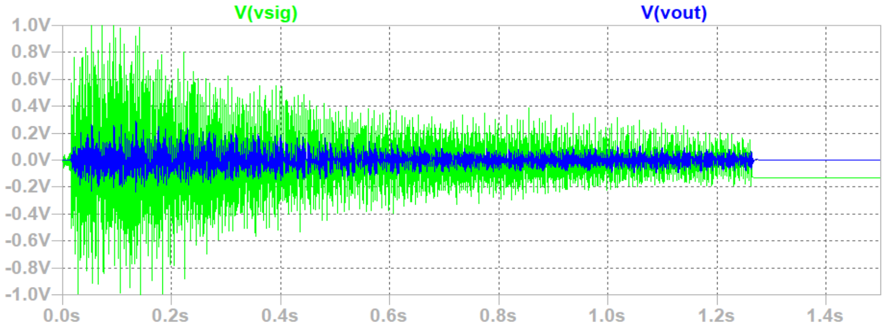


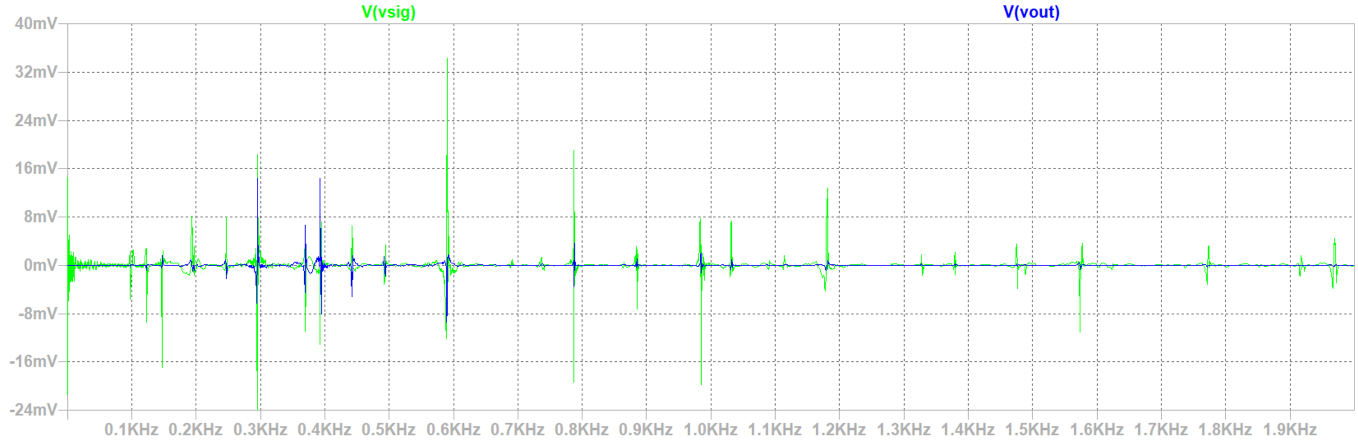
R3 at 200 ohms, resonate frequency = 620 Hz



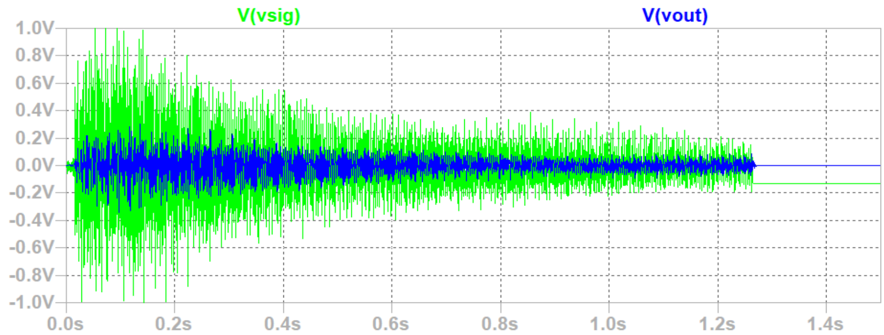


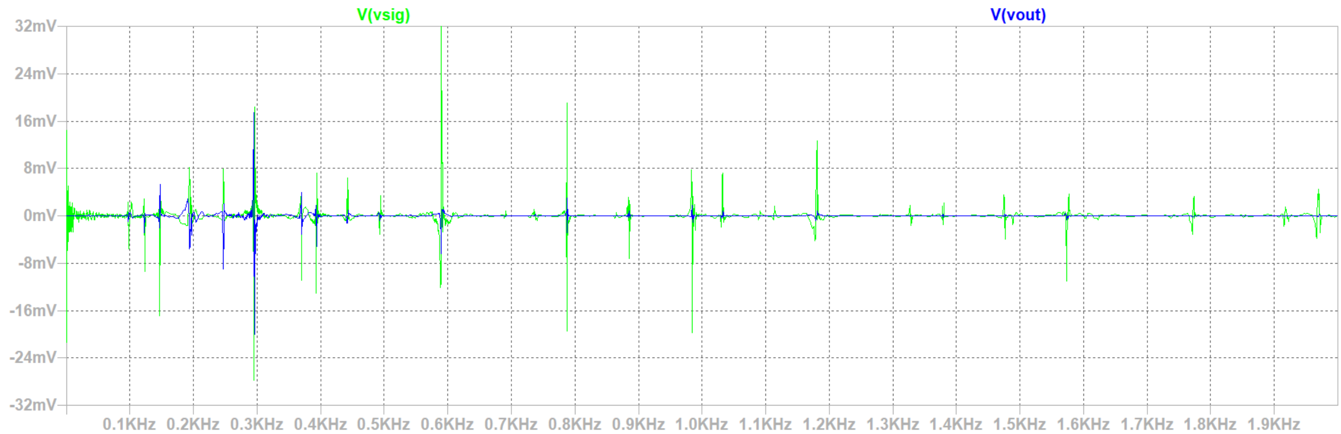
R3 at 500 ohms, resonate frequency = 390 Hz



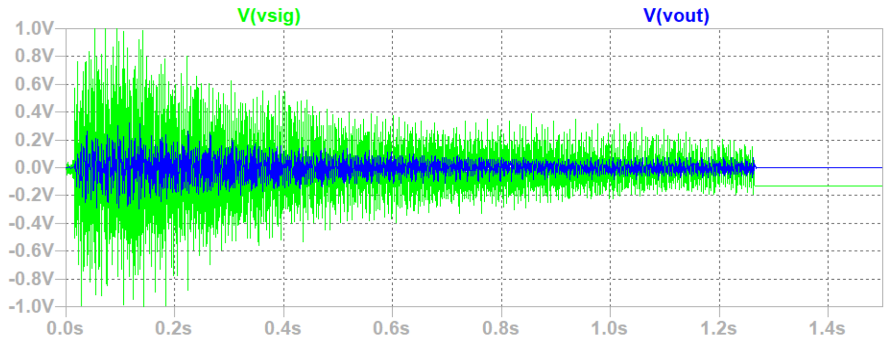


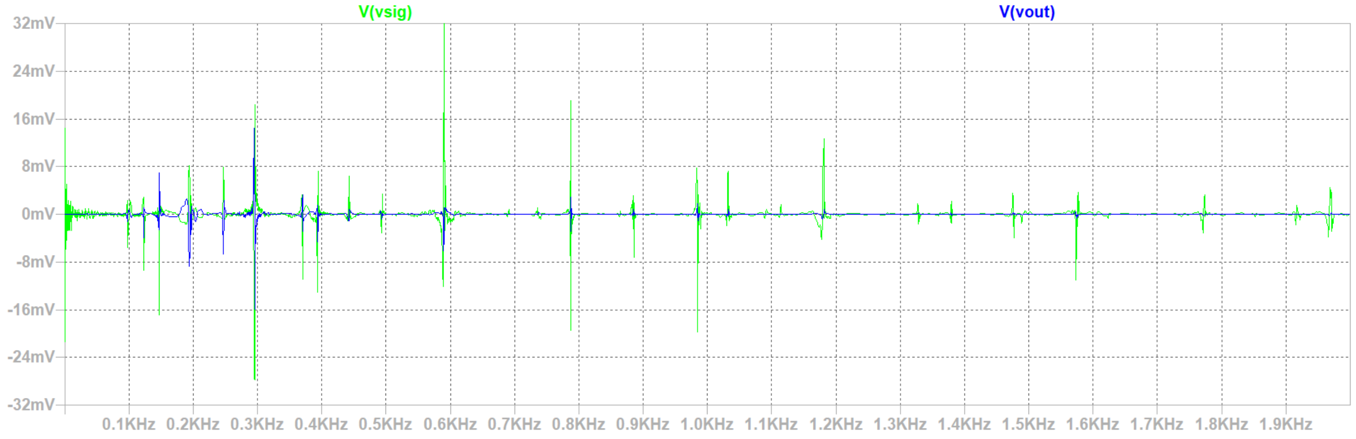
R3 at 1500 ohms, resonate frequency = 226Hz





R3 at 2000 ohms, resonate frequency = 196 Hz





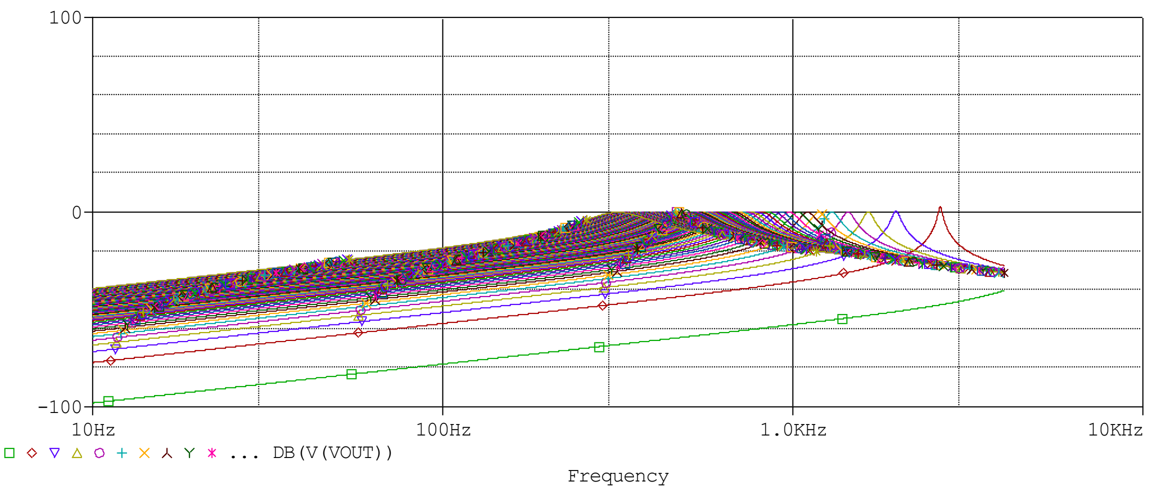
Reflection on the week

After observing the fft plots of the input and output it is clear that in the bandpass filter, as we increase the value of R3 we progressively sweep the frequency domain.

The bandpass filter is most selective at the high frequencies where the Q value is high, and the bandwidth is small. This was observed in figure 33 which has been added below for convenience.

When R3 is set to 20 ohms and the mid frequency becomes 1960 Hz, the bandwidth is ≈50 Hz.

While in the test at R3 set to 2000 ohms, the mid frequency was 196 Hz, the bandwidth becomes ≈500Hz. Allowing more of the side frequencies through.



Minutes from meeting

Work can focus onto testing with the built circuit, gathering more data that verifies the design and simulation.

Week 11: Testing & creating MATLAB model of analogue filter

With the measured resistor values of:

R1 = 14.72k

R2 = 33k

R3 = 150

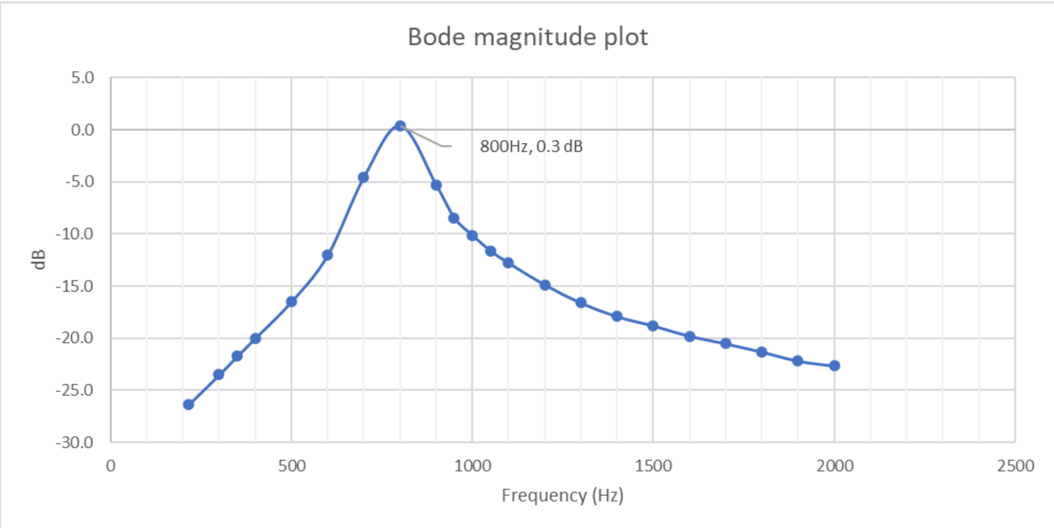
And Assuming C1 = C2 = ideal 100nF

$$\omega_o = \frac{1}{\sqrt{ReqR2}\ C}, where\ \omega_o\ = 2\pi f\ \&\ Req\ = \frac{R1R3}{R1 + R3}$$

$$f\ = \frac{1}{\sqrt{ReqR2}\ C * 2\pi}\ = 719\ Hz$$

The mid frequency should be 719Hz.

The testing gives the magnitude plot shown in figure // where it is 800Hz.



If we take account of the exact values of the capacitor, they are measured as

C1 = 95nF & C2 = 96.4nF

R1 = 14.72k

R2 = 33k

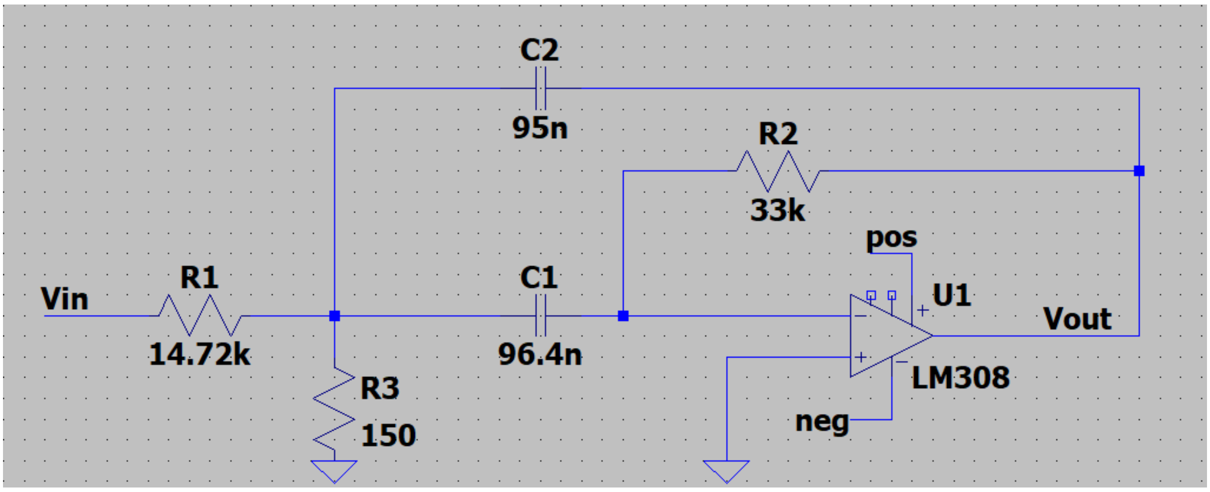
R3 = 150

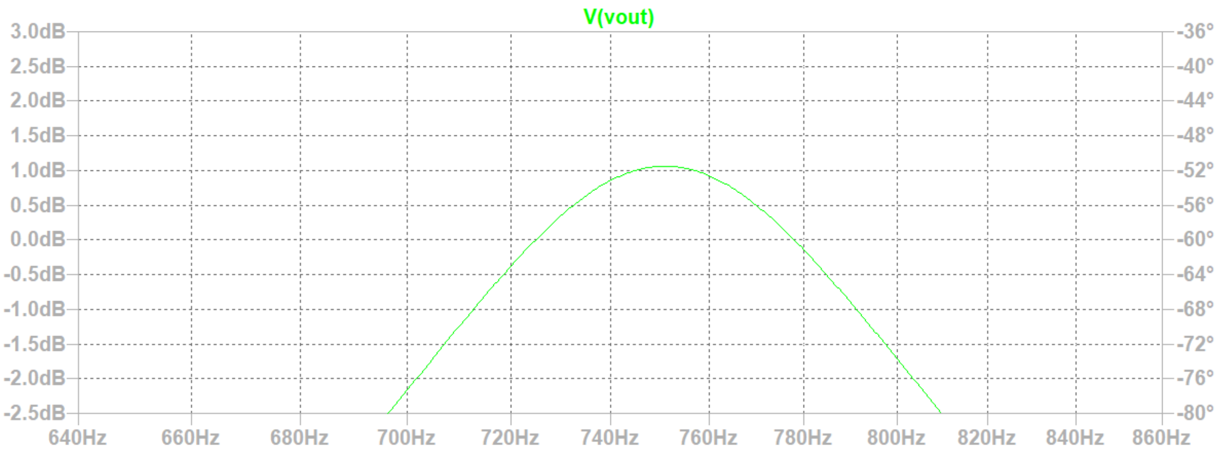
Then the more accurate calculation becomes:

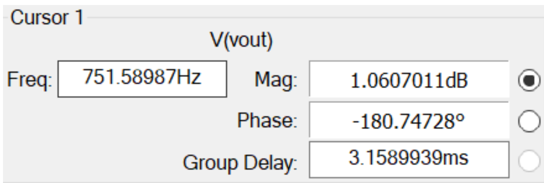
$$\omega_o = \frac{1}{\sqrt{ReqR2C1C2}}, where\ \omega_o\ = 2\pi f\ \&\ Req\ = \frac{R1R3}{R1 + R3}$$

$$f\ = \frac{1}{\sqrt{ReqR2C1C2}\ 2 * \pi} = 751\ Hz$$

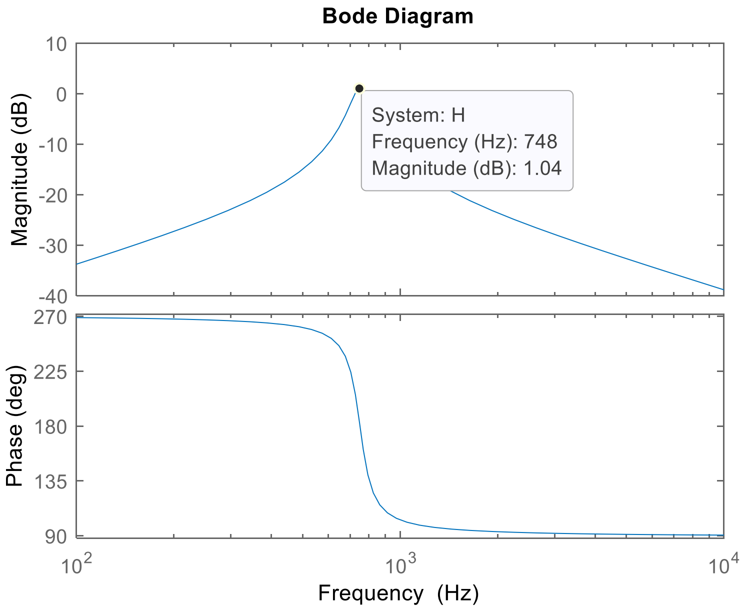
Simulation with measured capacitor & resistor values.







MATLAB, R3 = 150 ohms:



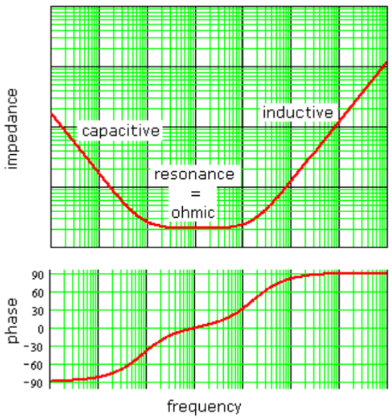


Fig. 6: Impedance and phase characteristic of a capacitor.

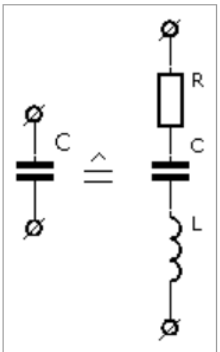


Fig. 4: Theoretical model of a capacitor.

https://meettechniek.info/passive/parasitic.html

The capacitor phase looks much like the above image with the theoretical model a capacitor with resistive and inductive behaviour.

Reflection on the week

The hardware tests are validating the expected values for frequency domain, the bode magnitude is near identical to the expected response with slight deviation found that can be attributed to factors such as electrical noise and non-ideal op-amp behaviours playing a role.

Minutes from meeting

Discussion on the MATLAB model of the pedal will allow cross examination of the effect (using an audio file) with the hardware test (playing in real time).

References

[1]. "Boss DS-1 | The Sound Factory", *The Sound Factory | Musical Instruments and Equipment*. [Online]. Available: https://thesoundfactory.town.ie/product/boss-ds-1/. [Accessed: 31- Jan- 2022].

[2]. C. D. Schiebel, "The 23 Types of Guitar Pedals Explained (2022) - Guitar Lobby", *Guitar Lobby*, 2021. [Online]. Available: https://www.guitarlobby.com/types-of-guitar-pedals/. [Accessed: 04- Feb- 2022].

[3]. "Music 101: What Is an Equalizer? Plus: Best Equalizer Settings for Drums and Guitar", *www.masterclass.com*,2021.[Online].Available: https://www.masterclass.com/articles/music-101-what-is-an-equalizer-plus-best-equalizer-settings-for-drums-and-guitar#:~:text=An%20equalizer%20(also%20called%20an,Car%20stereo%20systems. [Accessed: 03- Feb- 2022].

[4]. "Guitar 101: What Is a Wah-Wah Pedal? Learn About the Different Types of Wah-Wah Pedals", *www.masterclass.com*, 2021. [Online]. Available: https://www.masterclass.com/articles/guitar-101-what-is-a-wah-wah-pedal-learn-about-the-different-types-of-wah-wah-pedals#how-does-the-wahwah-pedal-work. [Accessed: 04- Feb- 2022].

[5]. "Music 101: What Is an Octave?", *www.masterclass.com*, 2021. [Online]. Available: https://www.masterclass.com/articles/music-101-what-is-an-octave#what-is-an-octave. [Accessed: 01- Feb- 2022].

[6]. Fox, "What Are Pitch-Shifting Guitar Pedals & How Do They Work? | My New Microphone", *My New Microphone*. [Online]. Available: https://mynewmicrophone.com/what-are-pitch-shifting-guitar-pedals-how-do-they-work/. [Accessed: 02- Feb- 2022].

[7]. Fox, "What Are Compressor Pedals (Guitar/Bass) & How Do They Work? | My New Microphone", *My New Microphone*. [Online]. Available: https://mynewmicrophone.com/what-are-compressor-pedals-guitar-bass-how-do-they-work/. [Accessed: 02- Feb- 2022].

[8]. S. Beattie, "Guitar Noise Gate Pedals - Your Ultimate Guide from Andertons Music Co.", *Andertons Music Co.*. [Online]. Available: https://www.andertons.co.uk/guitar-noise-gate-pedal-guide#:~:text=A%20noise%20gate%20pedal%20will,to%20prevent%20that%20annoying%20hum. [Accessed: 02- Feb- 2022].

[9]. "How Does A Reverb Pedal Work? - Best Guitar Equipment Q&A", *Best Guitar Equipment*. [Online]. Available: https://www.bestguitarequipment.com/blog/how-does-a-reverb-pedal-work/. [Accessed: 03- Feb- 2022].

[10]. Fox, "What Are Looper Pedals (Guitar/Bass FX) & How Do They Work? | My New Microphone", *My New Microphone*. [Online]. Available: https://mynewmicrophone.com/what-are-looper-pedals-guitar-bass-fx-how-do-they-work/. [Accessed: 01- Feb- 2022].

[11]. J. Chiffey, "Flanger vs Phaser – Understanding the Key Differences - Guitar Space", *Guitar Space*, 2021. [Online]. Available: https://guitarspace.org/sound-pedals/flanger-vs-phaser/#:~:text=Phasers%20work%20is%20frequency%2Dbased,and%20plays%20both%20back%20together. [Accessed: 01- Feb- 2022].

[12]. "Guitar 101: What Are Flangers and Phasers? Learn About the Best Flanger and Phaser Pedals for Electric Guitar Players", *www.masterclass.com*. [Online]. Available: https://www.masterclass.com/articles/guitar-101-what-are-flangers-and-phasers-learn-about-the-best-flanger-and-phaser-pedals-for-electric-guitar-players#:~:text=A%20flanger%20works%20by%20mixing,slightly%20slower%20than%20the%20other. [Accessed: 02- Feb- 2022].

[13]. R. Pulgar, "The Difference Between Overdrive and Distortion Explained | Gearank", *Gearank.com*, 2020. [Online]. Available: https://www.gearank.com/articles/overdrive-v-distortion. [Accessed: 04- Feb- 2022].

[14]. "The Main Difference between Active and Passive Components", *ELECTRICAL TECHNOLOGY*, 2013. [Online]. Available: https://www.electricaltechnology.org/2013/06/the-main-difference-between-active-and.html. [Accessed: 17- Apr- 2022].

[15]. O. B. Olalekan and O. V. Toluwani, "Sallen-Key Topology, MFB and Butterworthy in Bandpass Design for Audio Circuit Design", *Asian Journal of Electrical Sciences*, vol. 6, no. 1, pp. 23-28, 2017. Available: https://www.researchgate.net/profile/Blessed-Oyebola/publication/329781826_Sallen-Key_Topology_MFB_and_Butterworthy_in_Bandpass_Design_for_Audio_Circuit_Design/links/5c1a3b8c299bf12be38af768/Sallen-Key-Topology-MFB-and-Butterworthy-in-Bandpass-Design-for-Audio-Circuit-Design.pdf. [Accessed 17 April 2022].

[16]. C. Moler, "Cramer's rule on 2-by-2 systems", *ACM SIGNUM Newsletter*, vol. 9, no. 4, pp. 13-14, 1974. Available: 10.1145/1206085.1206090 [Accessed 15 March 2022].

[17]. Texas Instruments, “LMx24-N, LM2902-N Low-Power, Quad-Operational Amplifiers,” LM324-N datasheet, March. 2000 [Revised Jan. 2015].