ENGN2229 LAB REPORT 1

Written By: Shikhar Vaid (u5922125) and Zeefan Khan

SUBMITTED TO AUSTRALIAN NATIONAL UNIVERSITY

Executive Summary

The given circuit is a notch filter, which can reject a particular frequency band. Findings from the simulations performed on MATLAB show the ‘notch’ or central rejection frequency is roughly around 50Hz. Further simulations of electric circuit done online through circuitlab indicated that the rejection frequency is theoretically 47.9 Hz, which agrees with the MATLAB simulations. It should be noted that this circuit does attenuate the signal close to the notch frequency, and a frequency range of ±30Hz can be cancelled out in the process. Thus to make this band narrower, other circuit elements will have to be used, such as op-amps and so on. The simulations performed on MATLAB were done through making Bode Plots, doing a time domain analysis of output voltage and the Fast Fourier Transform functions of the output voltage. Here it must be pointed out the time step had to be extremely small, especially for analyzing higher frequencies. Similarly very large values for the total simulation time were needed to obtain a smooth bode plot with high resolution whilst avoiding aliasing and leakages. The circuit can be used to cancel out power supply in Australia (50 Hz) for use in communications and biomedical applications.

Background : Analysis Methodology

For the purpose of analysis it was decided that three different kinds of simulations will be performed. These were analysis of transfer function through bode plots, time domain analysis of output voltage and fft function of the output voltage. A function was created filterX.m (explained in depth in Appendix) and it was called in a wrapper function. The RC circuit reproduced below was simulated, using the given ODE.

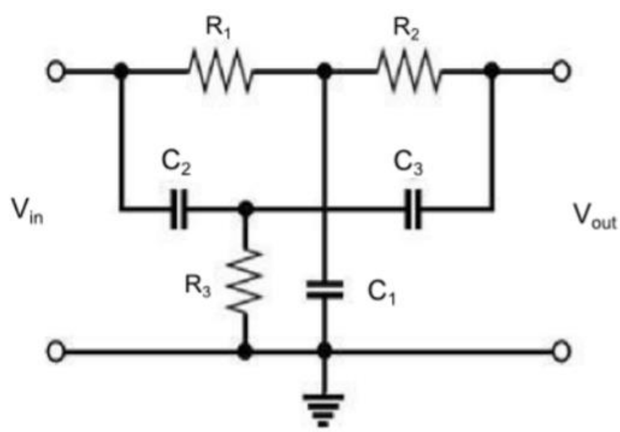


Fig. RC Filter Circuit for analysis. It can be seen here that this circuit is divided into two separate lines. One, is a Resistor Capacitor Resistor (RCR) line, and the other is a CRC line. For certain AC inputs, this circuit gives out Vout, and has two nodes of interest which have been called V1 and V2 in the ODE.

Two scenarios with different values of the circuit’s resistors and capacitors are provided, however, the input and output voltages are unknown. The input voltage is AC; this allows us to express the function of the signal mathematically, however it does not include the magnitude of the signal.

$$\frac{d}{dt}\begin{Bmatrix} V_1(t) \\ V_2(t) \\ V_{\text{out}}(t) \end{Bmatrix} = \begin{bmatrix} -\frac{1}{C_2R_3} & \frac{1}{C_2R_2} & -\frac{1}{C_2R_2} \\ 0 & -\frac{1}{C_1}\left(\frac{1}{R_1}+\frac{1}{R_2}\right) & \frac{1}{C_1R_2} \\ -\frac{1}{C_2R_3} & \frac{1}{R_2}\left(\frac{1}{C_2}+\frac{1}{C_3}\right) & -\frac{1}{R_2}\left(\frac{1}{C_2}+\frac{1}{C_3}\right) \end{bmatrix}\begin{Bmatrix} V_1(t) \\ V_2(t) \\ V_{\text{out}}(t) \end{Bmatrix} + \begin{bmatrix} 0 & 1 \\ \frac{1}{C_1R_1} & 0 \\ 0 & 1 \end{bmatrix}\begin{Bmatrix} V_{\text{in}}(t) \\ \frac{d}{dt}V_{\text{in}}(t) \end{Bmatrix}$$

Fig. Given ODE for circuit. This was used to calculate Vout, through transfer function analysis.

After a circuit analysis using KCL/KVL and a conversion of the system into the phasor domain; the given ODE can be derived and understood. With the help of ENGN2229 LAB02, signal analysis and frequency domain analysis were used to simulate the circuit. The initial values of the ODE are assumed to be 0; this would be the case at t=0 before the circuit has been switched on.

Signal Analysis:

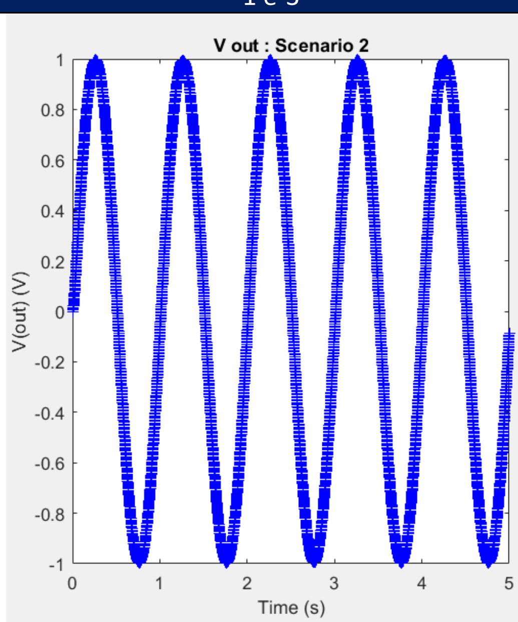
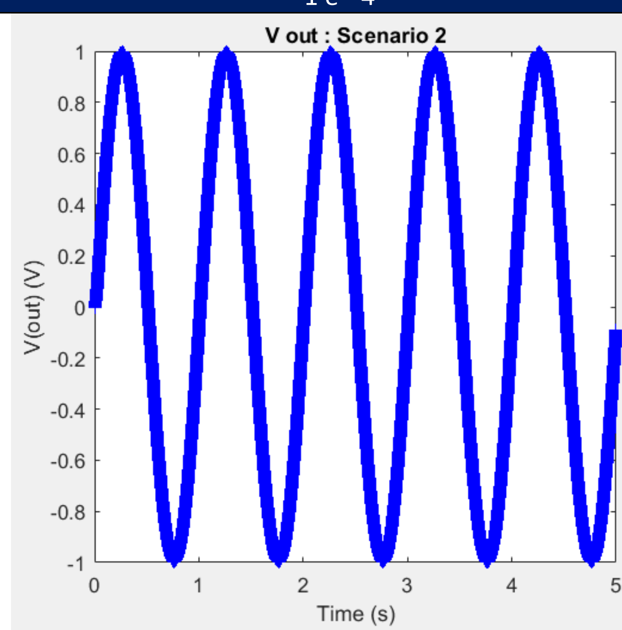
The given ODE can be simplified into a system of 3 equations, and with 3 unknowns (V1, V2, and Vout), we can use an integration script like the one produced in ENGN2229 LAB02.1 to compute values for V(out) in terms of V(in). For such a script to work, it would need certain input arguments:

- ω = Input signal's frequency

The ω (input frequency) was varied between 1Hz – 1kHz as requested in the assignment.

- dt = Time step

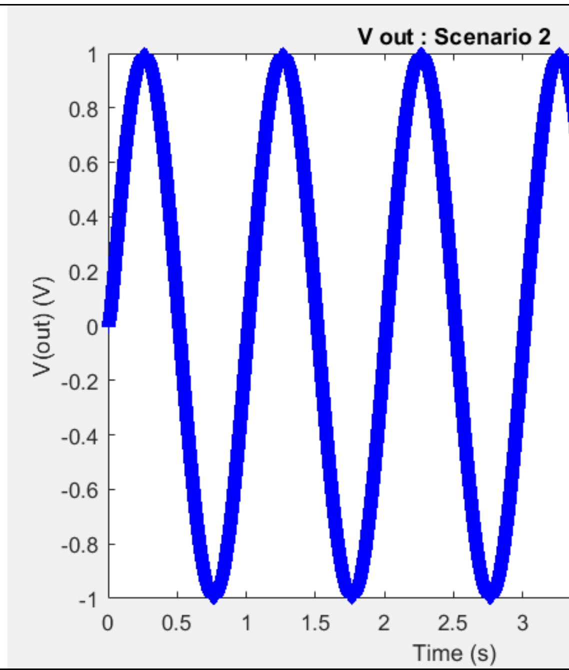
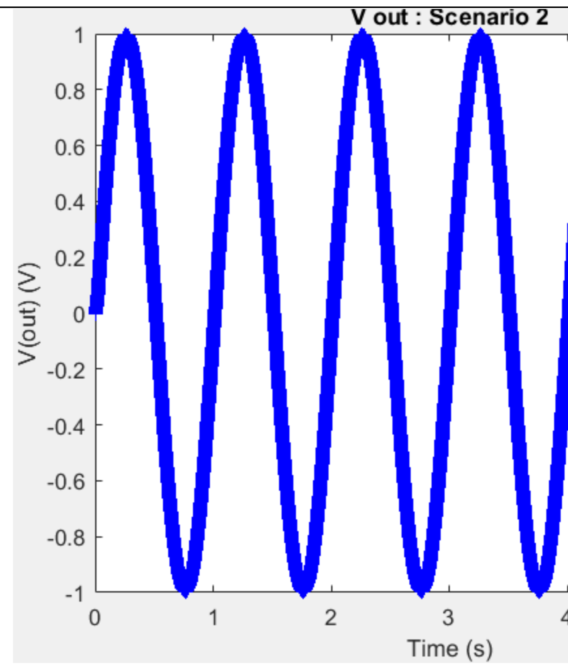
The time step (dt) in the simulation had to be picked with careful consideration. The smaller dt is, the more accurate the numerical solutions (especially in the case of integration) MATLAB can provide, however, if dt is too small, the simulation would require considerable more processing power, and in turn long runtime. Values decreasing in the order of 10s were tested and the results shown below:

dt (s):	1 e-3	1 e^-4
Vout (V) against time (s)		
Comments:	The pattern is as expected, inconsistent noise	Expected pattern, regular pattern of noise

When dt>1*10^-3, V(out) no longer produces a regular pattern, hence smaller values were considered. The table above shows how an approximate value for dt=1*10^{-4} s was initially chosen; any smaller values of dt did not show a visible increase in resolution in the graphs, and smaller values seemed to produce irregular to no pattern.

- t_{max} = Simulation time

As simulation time increases, so does the number of corresponding computed values. This means the more time values there are to input, the more continuous the output signal seems to be. A wide range of t_{max} was tested, and the significance of the value was not seen at this point:

t_{max} (s)	5	500
Vout (V) against time (s)		
Comments:	Expected pattern, little noise	Expected pattern, little noise

The simulation time is kept at an arbitrary point (t=50) until further revised.

Frequency Domain Analysis:

Firstly, a frequency scale needs to be established. The scale has as many points as there are time values; dt is again responsible for the accuracy of our plotted values and may need to be further optimized.

To convert the input and output signals into the frequency domain, the discrete Fourier Transform of the signals is used.

Using FFT (Fast Fourier Transform, MATLAB algorithm) on Vin and Vout, and dividing the absolute value of the FFT by the number of signal values gives the amplitudes of the signal (over the set time values). The largest value (amplitude) of the signal at any given point in time at a specific frequency would be the peak signal value. The max function was used for the final calculation for peak input values and peak output values.

Finally, an analysis of the circuit can be made, by plotting the gain (peak output values / peak input values) against the frequency spectrum.

With a new domain, the effects of varying total simulation time were again tested and the results below

T_{max} Bode Plot	5	500	1000
Comments:	Too few data points	Expected pattern with random data points in frequency rejection range	Expected pattern, ideal value, any larger and runtime is too large

- Transient cut-off time

Trans cut-off (s) in Bode Plot	5	50	100
Comments:	Irregular pattern with inconsistencies mainly	Expected pattern, random noise at higher frequencies	Expected pattern, ideal cut-off time

The transient cut-off time was an ignored parameter up until this point. In Bode plots, the effects of varying cut-off time were quite visible. This parameter causes unpredictable effects on the data if not controlled for.

Simulation of the circuit shows that the runtime of computing data points increases as the input frequency is increased. With analysis of the final Bode plot, it can be deduced that the data points at the non rejection

- numerical accuracy (High Accuracy Compromised processing speed) Time Step needs to be small enough at higher frequencies.
- peak finding, etc. (use max function not peaks, much faster. Prove. Find reference)
- How did you solve them. Large Tmax, Small T_cutoff, Very Small dt. Range of different F, smaller steps in F near rejection frequency, larger steps towards end.

Cut off the transient well enough for your frequency domain analysis. Value at 40. Not needed initial values, since gain is 1, more necessary to simulate the exact values for range 50-80 Hz as that is approximately Rejection frequency response.

The analysis for the values of time and total simulation time were initially based on method of Hit and Trial. However, later simulations of nodal voltages, fft and online simulations of electrical circuit provided the right insight to remove leakages and sampling frequency limitations.

Discussion : Results

As state above, the method decided to study the circuit was a transfer function analysis using bode plots, these are reproduced below for both the scenarios.

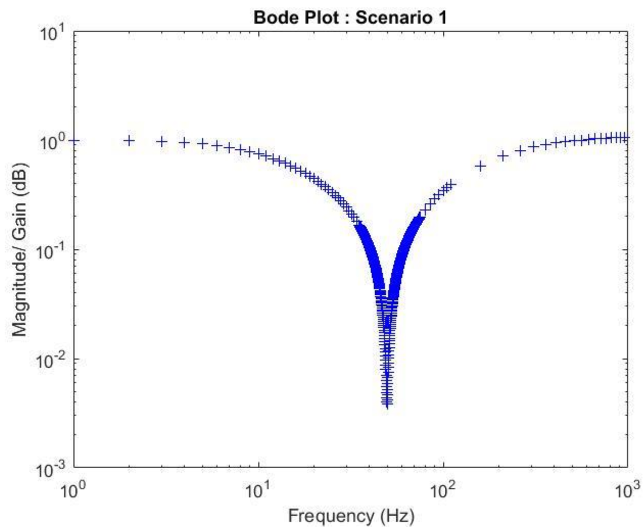
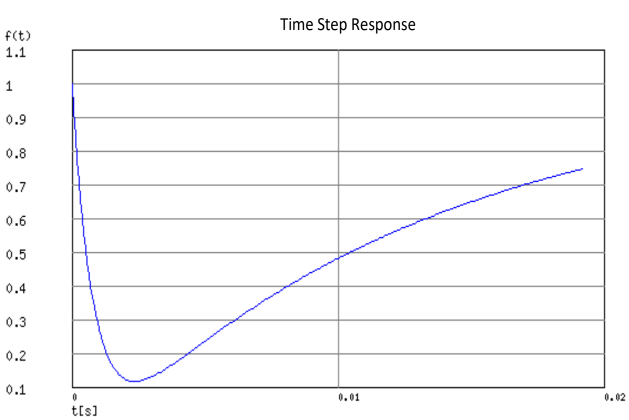


Fig. Bode Plot Scenario 1. It can be clearly seen here that the gain is 1 and never larger than that. However, in the middle roughly about 50Hz frequency the magnitude takes a sudden dip. This is called the rejection frequency. This is because of how the RCR and the CRC lines in the circuit behave.

While analyzing this circuit it must be noted that it is based on the fft functions produced in the matlab code, which by definition cannot have infinite accuracy. Some possible sources of error have been discussed here with the FFT approach:

- Aliasing, sampling rate too slow, not supplying enough data to represent the frequency content of the signal.
- Leakages, simulation too short, not supplying enough data to correctly bin the frequencies.
- Resolution, right amount of samples needed, not supplying enough data to achieve full distinction between close frequencies at the desired level of granularity.

This is why the magnitude does not drop to zero, however it seems that is what the circuit wants to achieve.Online simulations were then performed used circuit lab and the results are produced below:



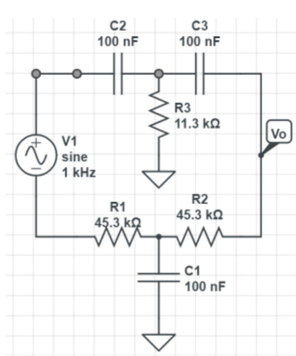


Fig. (LEFT) Time Step Response of Simulated Circuit Performed using Circuit Labs. This indicates the time step needs to be small otherwise many sampling frequencies will be skipped. (RIGHT) Fig. of Simulated Circuited

Notch Frequency = 1/2piRC

Center rejection frequency

f0 = 49.686330913748[Hz]

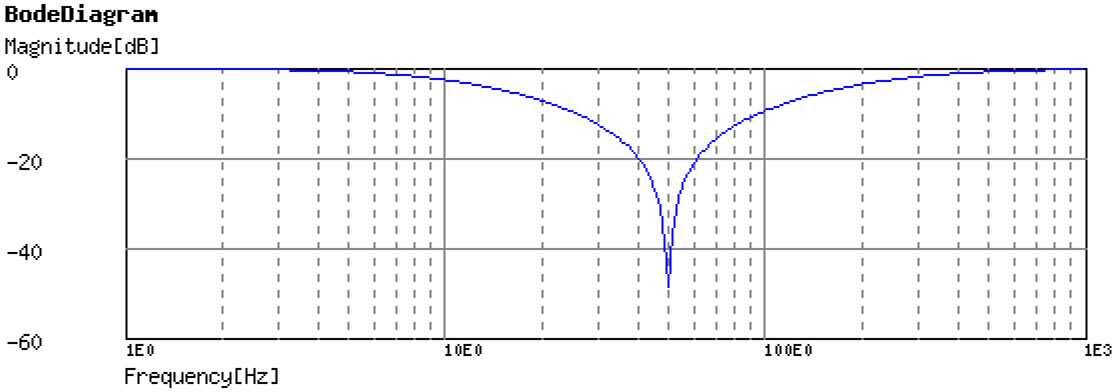


Fig. Simulated Bode Plot. Perfect Match

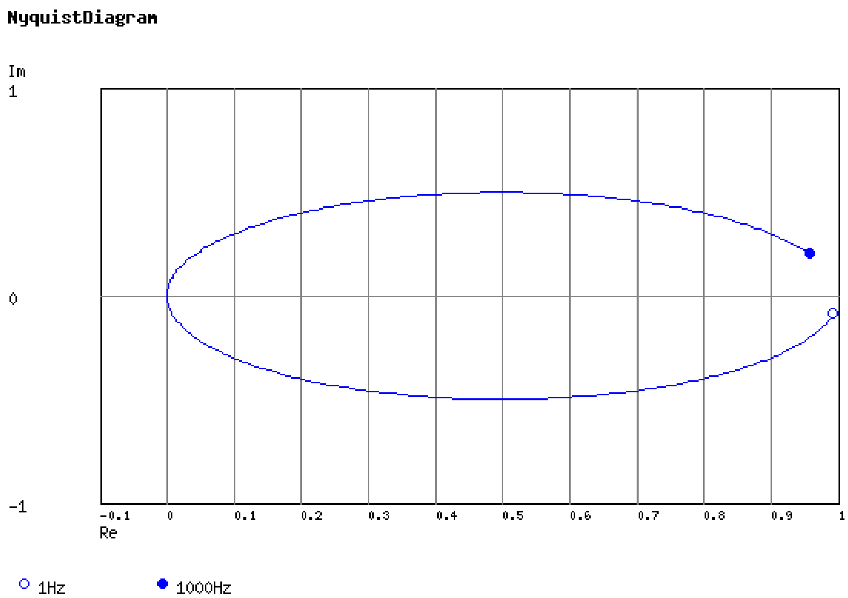


Fig. Nyquist Plot of simulated circuit. It is the graphical representation of a feedback system which shows the relationship between Feedback and Gain in a Complex Graphical format. It can also be used to assess the stability of the closed loop system by applying Nyquist stability criteria.

Nyquist-Shannon theorem limits. Because of the way the FFT algorithm works, it will show us a mirror image of our peaks in the range 5 kHz to 10 kHz, we need to double the amplitude of the peaks in the valid range up to 5 kHz to get the actual frequency content of our signal. In reality we have twice the energy at every frequency than what the FFT plot shows in that range. Algorithm breaks beyond sampling frequency.

Hence from theorem, we need high sampling frequencies to represent signals for numerical analysis, need enough samples.

Scenario 2

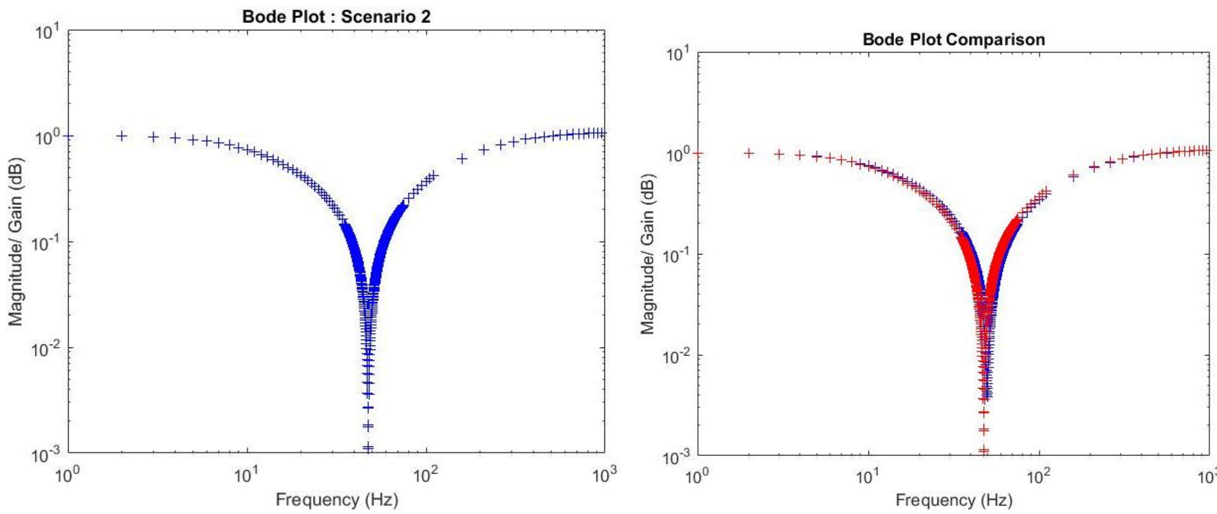


Fig. (Left) Bode Plot for Scenario 2 and (Right) Bode plot Comparsion between Scenario 1 (blue) and 2 (red). The bode plots here have been used to study how the gain changes at different frequencies for the circuit. It can be easily seen that the second scenario has a notch frequency slightly less than that of Scenario 1, this has to do with the RC values in this circuit.

Since the scenario 2 values are standardized less notch frequency is observed, perhaps a couple Hz less than that of scenario 1. This makes it further away from 50Hz, which seems to like a specific frequency that may be needed to be cancelled out in desirable applications (discussed below). Simulations performed on circuitlab online for these RC values in the circuit show that the rejection frequency, is 47.89 Hz, decreased from about 49Hz in scenario 1, thus our simulation agrees with the theoretical result. However, theoretically speaking at the notch frequency the magnitude or gain should be zero, which is not exactly observed here, this has to do with the limitations of FFT discussed above, our chosen time steps for faster simulations and so on.

Possible Uses of the Filter

The simulated circuit can be extremely beneficial if used as a notch filter. The simulation clearly shows how there are two bridges formed in the RC filter. One which allows low frequencies to pass through, and one which allows high frequencies to pass through. This is a very unique kind of filter. This kind of circuit behaviour can be particularly useful in “rejecting” or cancelling out specific frequencies. If there is a scenario with a particular unwanted frequency within a signal, it can be filtered out. Our particular scenarios allow for 50 Hz signals to be rejected,

The Filter response has a minute attenuation within ±30 Hz away from notch frequency, and little to no attenuation at frequencies larger or smaller than rejection frequency. However, the closer the signal gets to notch frequency, the greater the amount of attenuation, providing the generic notch filter response.

It is commonly used for attenuation of a single frequency such as 50 Hz power line frequency hum. In Australia the AC electricity distribution occurs at 50Hz (Aprea, et al., 2006) . This particular kind of filter hence is perfect for cancelling out the power delivered frequency in various instruments for communications and biomedical sensory devices. For instance, this filter could be used in a ECG machine to typically eliminate the 50Hz supplied frequency response. The attenuation quality is not ideal and the range is a little wider than theoretically desirable which may slightly distort the output signal. However, this filter can be used in such an application with addition of some other electrical components such as op-amps and so on.

Appendix

Function filterX

'filterX' is a MATLAB function which was used to simulate the given RC filter. The function has a specific signature and it has been reproduced below. It takes frequency 'omega_in'

```
function [p_in,p_out]= filterX(omega_in,dt,t_max,trans_cutoff);
```

Input Signal and Time Domain

The function starts by setting up the time domain (line 5) which is used throughout the analysis. This is done so that the input signal, output signal and all the simulations can be computed for desired lengths. The input AC signal has been modelled using a sine wave which takes frequency 'omega_in' from the function definition. (lines 12, 13). The differential of the input signal has also been calculated using calculus for the performing the differential function as per the given circuit ODE.

Differential Function

First the variables corresponding to the resistors and capacitors in the circuit were initialized lines (22-27). Then the nodal voltages V1, V2, and Vout were initialized (lines 33-35, reproduced below) and set to be zero vectors of length 't' to match the input signals length. Here the initial conditions at time=0 have been set, which agrees with the logic of how this RC circuit behaves. A for loop is used to perform matrix operations as per the given ODE in the assignment sheet. This loop systematically first computes differential voltages dV1, dV2, dVout and then uses that to calculate the corresponding voltage values (lines 46-48) at each time step (see below).

```
33    V1 = zeros(length(t),1);
48    Vout(count+1) = Vout(count) + dt*dVout;
```

Frequency Scale

Frequency Scale is established (lines 54-57) for generating suitable plots to conduct frequency based analysis. The max frequency is set 1000 Hz, whilst the frequency step is computed using the time step and length of input signal which was calculated previously so matrix dimensions agree while plotting.

Fast Fourier Transform Function

It was decided that the circuit be simulated through bode plot, transfer function and time domain analysis was also conducted. Thus, FFT calculations are needed (line 62) from which we can find peaks and compute gain. FFT_in is divided be length of Vin, and FFT_out(line 67) is restricted to obtain suitable plots starting from transient cut off times to total simulation time.

Finding Peaks for P_in and P_out

To calculate gain in the bode plot, peak values are necessary which are computed fastest using max function. p_in has been restricted in length to perform suitable matrix operation later while calculating gain. These values are the output of the function filterX

```
72    p_in = max(fft_in(1:length(Vin)));
73    p_out = max(fft_out);
```

References

https://www.electronics-tutorials.ws/filter/band-stop-filter.html

http://sim.okawa-denshi.jp/en/TwinTCRtool.php

http://fourier.eng.hmc.edu/e84/lectures/ActiveFilters/node4.html

Tasner, T., Lovrec, D., Tasner, F. & Edler, J. 2012, "COMPARISON OF LabVIEW AND MATLAB FOR SCIENTIFIC RESEARCH", Annals of the Faculty of Engineering Hunedoara, vol. 10, no. 3, pp. 389.

Aprea, C., Mastrullo, R. & Renno, C. 2006, "Experimental analysis of the scroll compressor performances varying its speed", Applied Thermal Engineering, vol. 26, no. 10, pp. 983-992.