Table of Contents


```
clear
clc
```

MNA Circuit Modeling

Adam Heffernan 100977570

This assignment uses the modified nodal analysis method for circuit analysis. This is the underlying framework behind P-Spice or LT-Spice, i.e circuit simulators. All circuit simulators use this basic frameworek in order to analyze a circuit, big or small. This method really cuts down on processing time for a computer as if the computer needed to solve circuits with hundreds of nodes the non-matrix method would be extremely time consuming and costly on the program and taxing on computer resources in order to combat this, we use stamps for each element in our circuit and put them into a matrix, by doing matrix division with a column vector for the current at each node in the circuit we are able to solve for voltage, as shown in the following experiments.

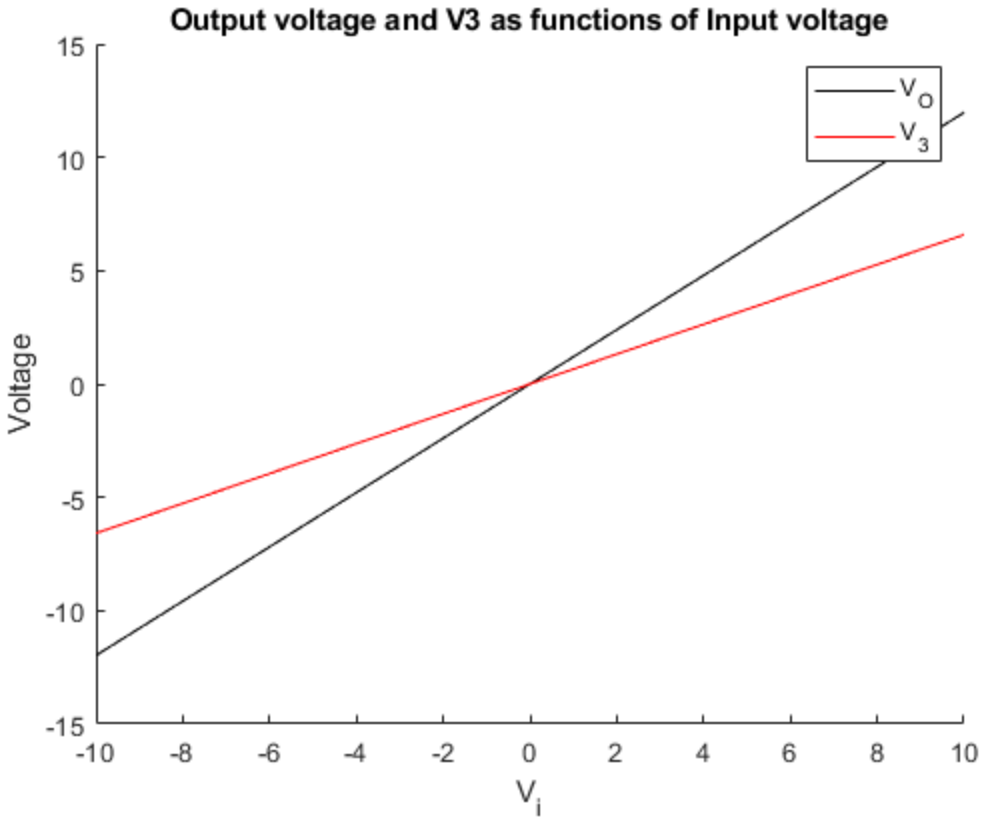
Part1

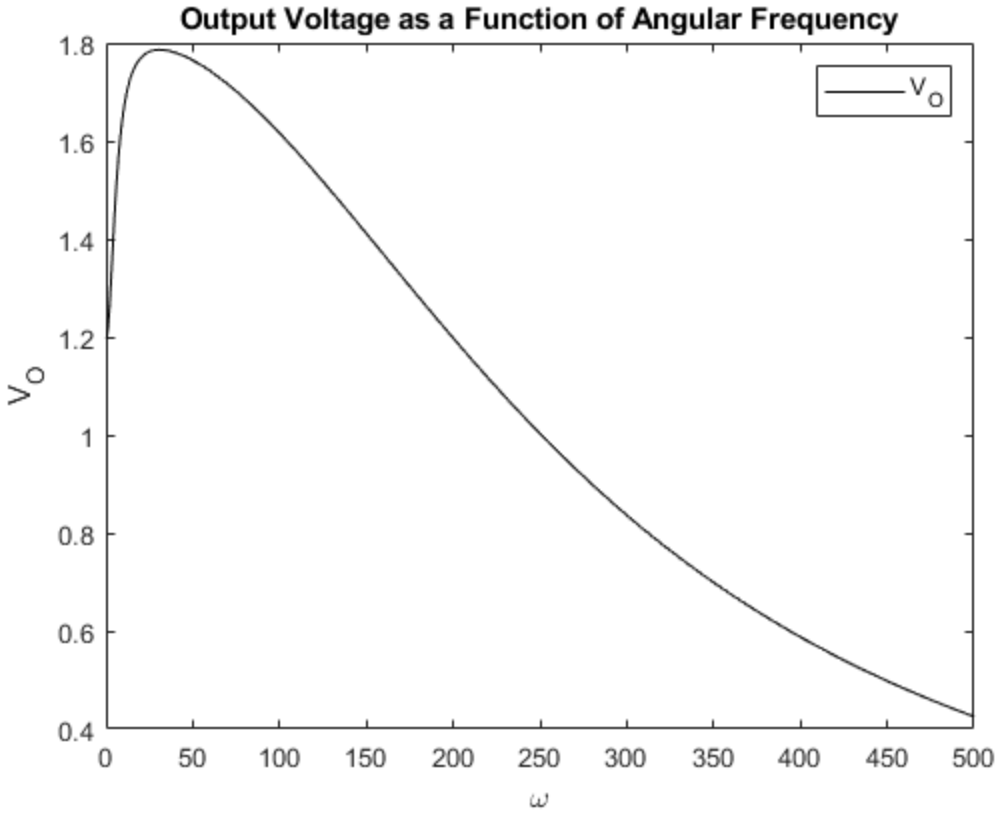
Basic sweep was done using the simulation from part 3, a resistance value of approximately 50 Ohms was found using this method. The rest of the simulation then proceeded. In the MNA PA we learned how to stamp different components into a matrix in order to complete the modified nodal analysis technique. This is a low pass filter circuit. We would expect a frequency response that declines at 40dB/decade as this circuit is a second order filter.

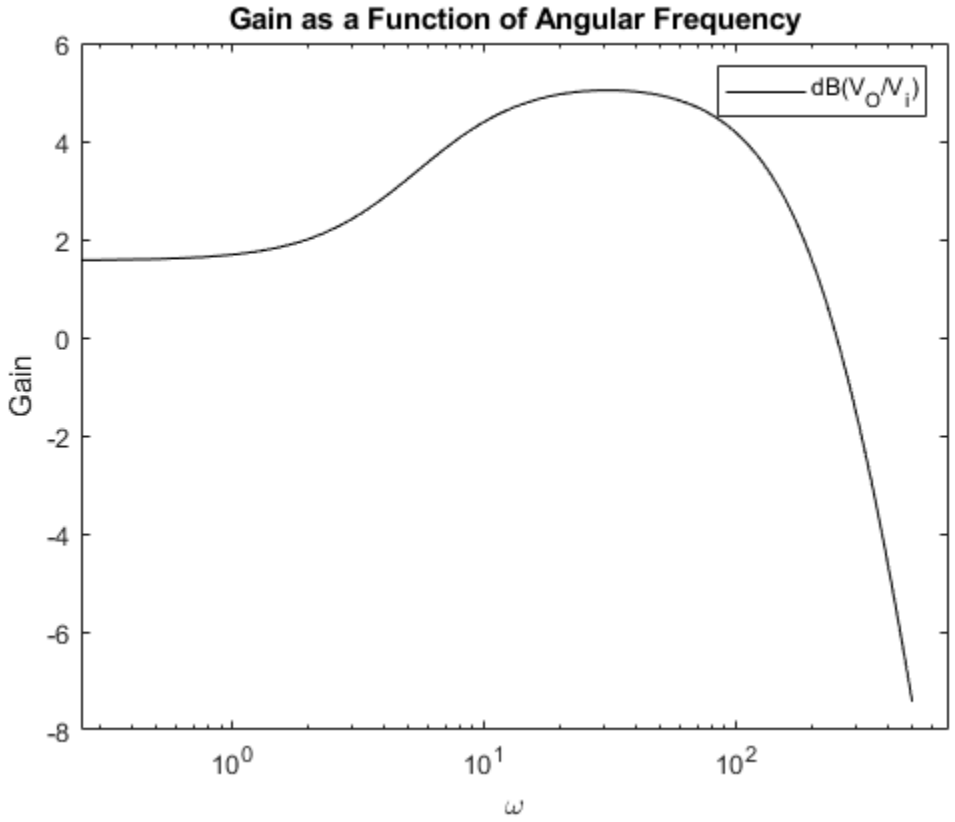
```
Part1;
```

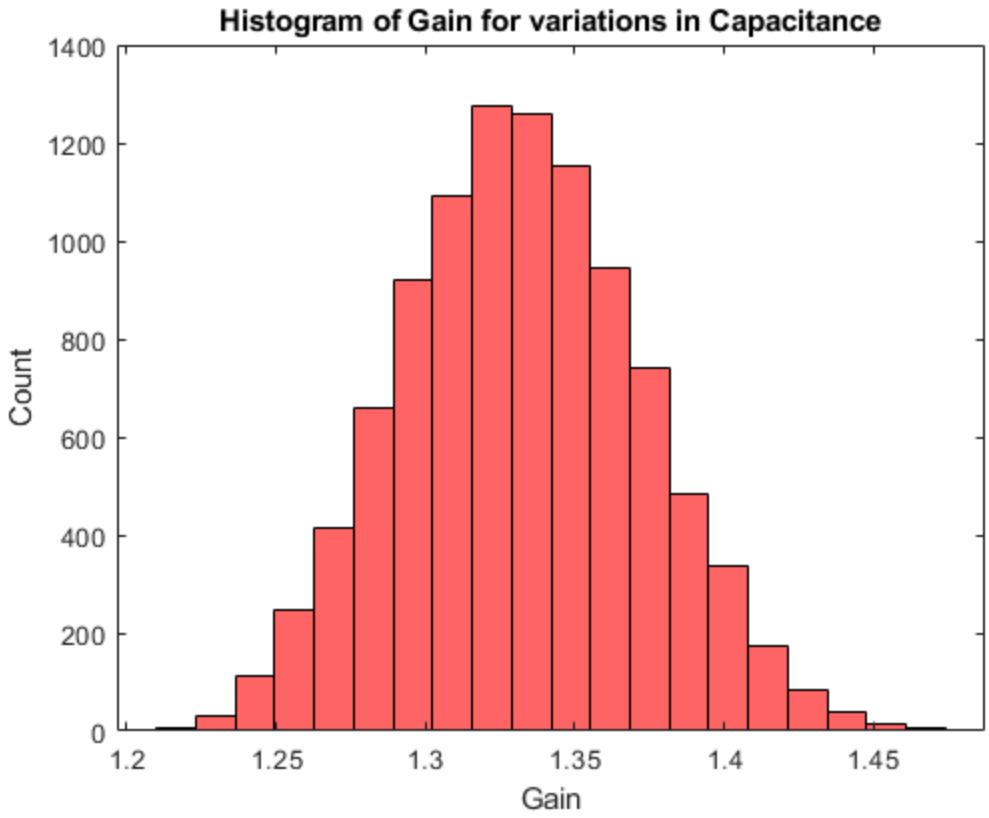
```
C =

         0         0         0         0         0         0
   -0.2500    0.2500         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0    0.2000
```

```
G =

    1.0000         0         0         0         0         0
   -1.0000    1.5000         0         0         0    1.0000
         0         0    0.0182         0         0   -1.0000
         0         0   -1.8182    1.0000         0         0
         0         0         0  -10.0000   10.0010         0
         0   -1.0000    1.0000         0         0         0
```




Output Voltage as a Function of Angular Frequency
V_O
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0
50
100
150
200
250
300
350
400
450
500
ω

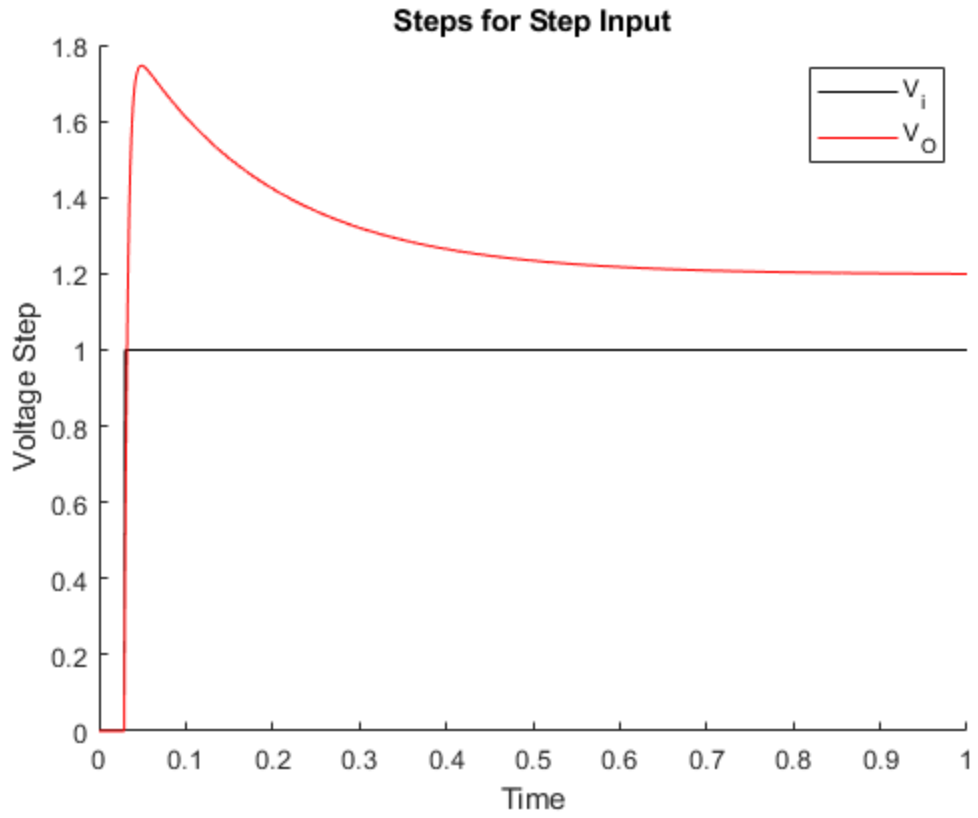

Gain as a Function of Angular Frequency
dB(V_O/V_i)
Gain
6
4
2
0
-2
-4
-6
-8
10^0
10^1
10^2
ω

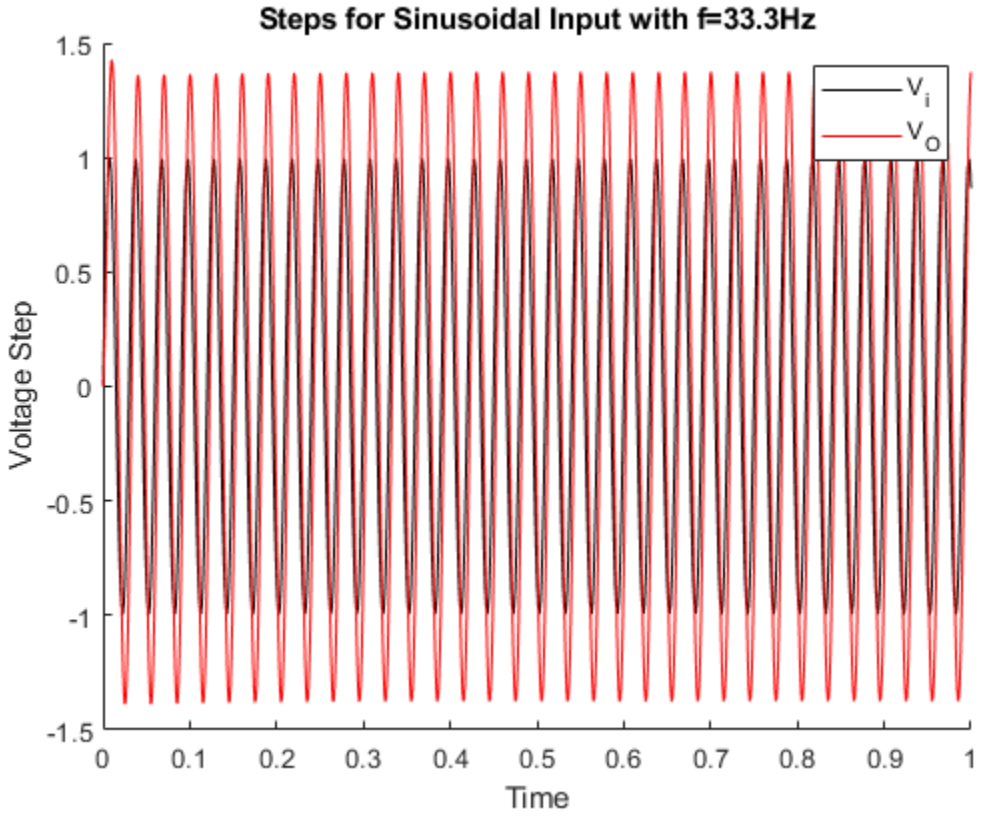


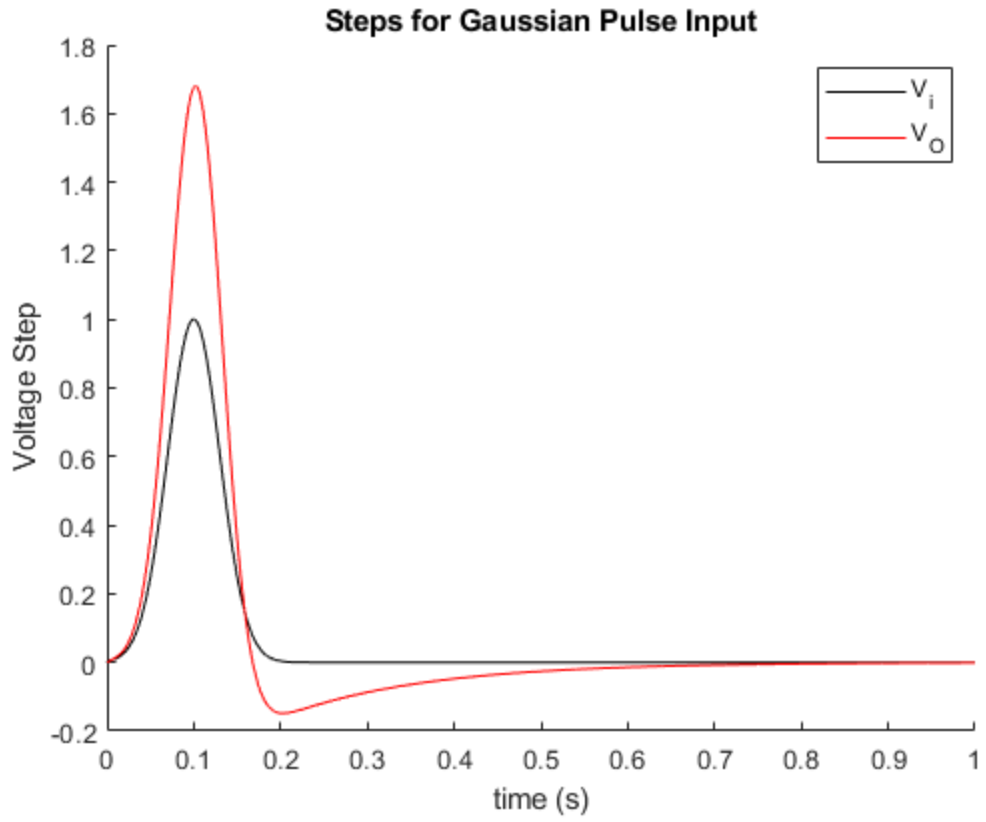
Part2

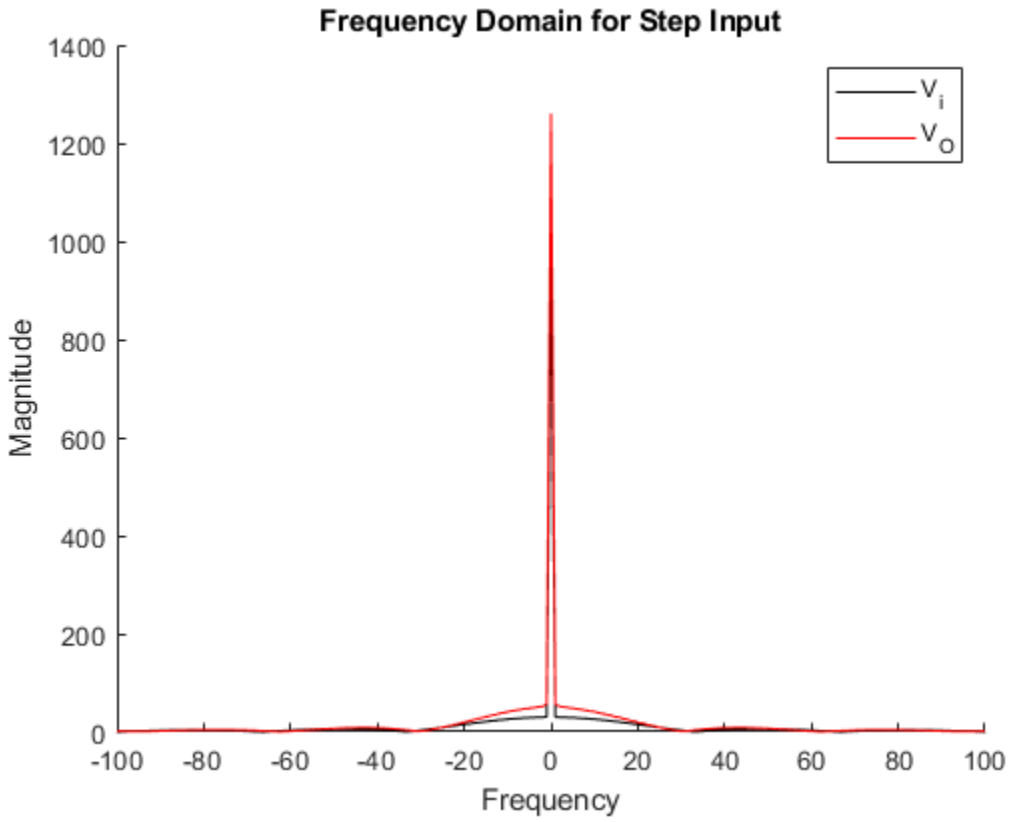
Different inputs with no noise were modeled. A gaussian input a sinusodal input and a step input. The frequency content of the signal was plotted using the FFT command.

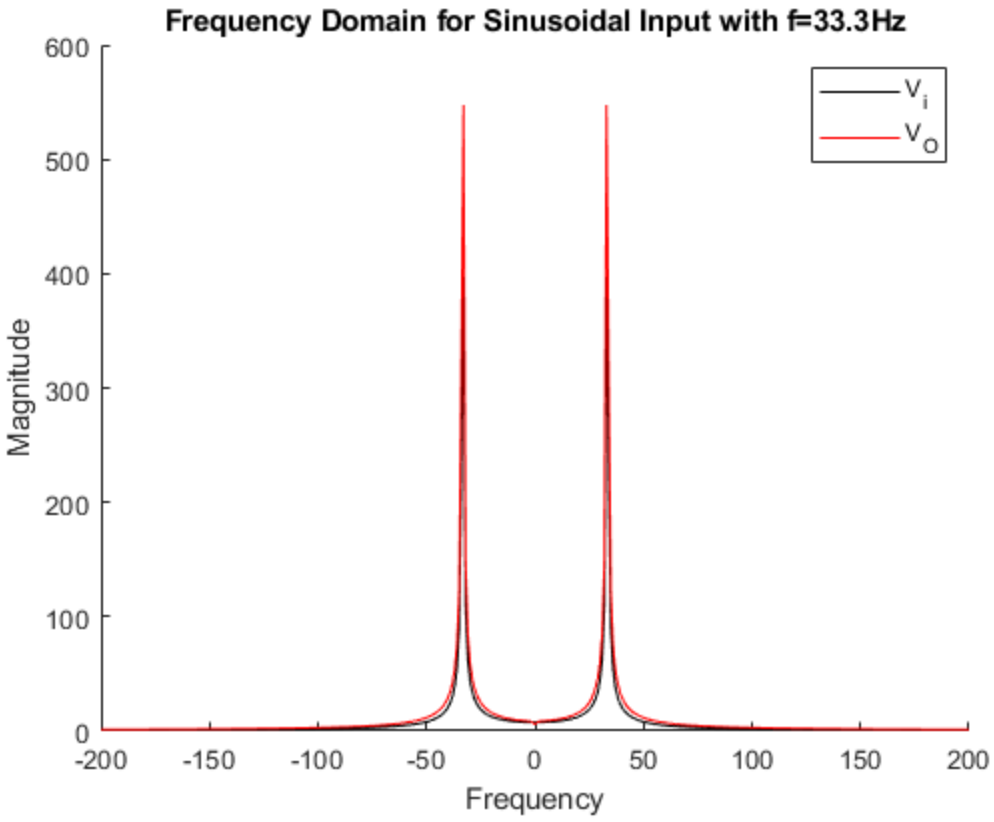
```
Part2;
```

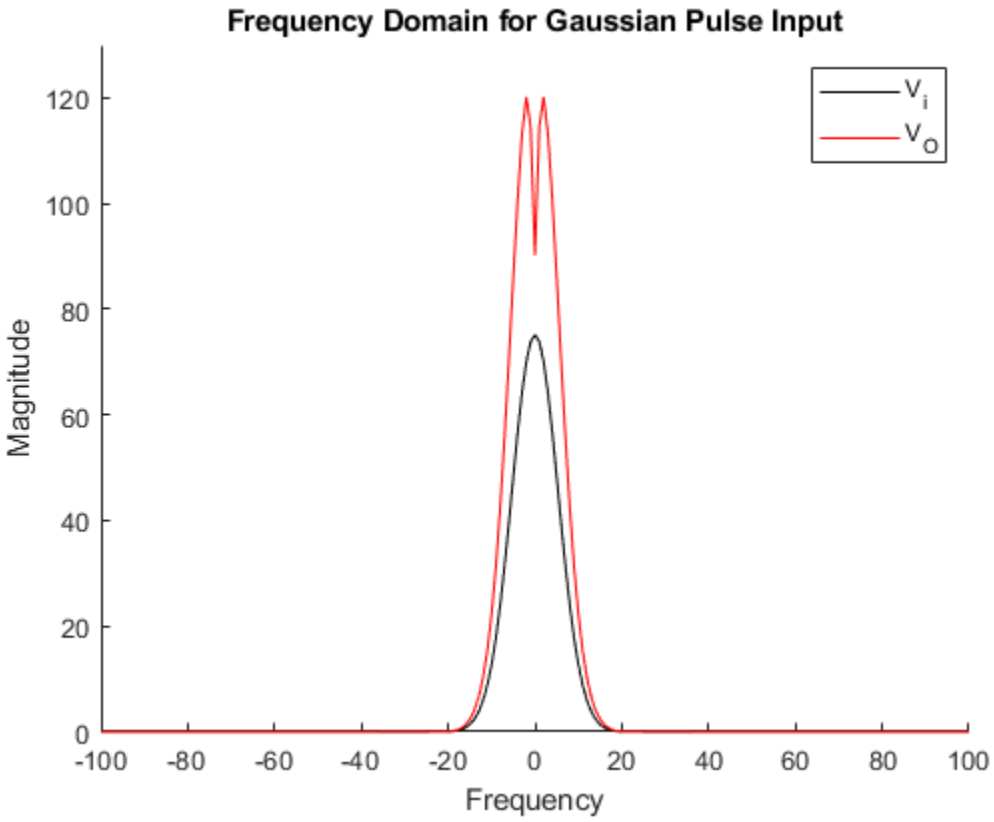

Steps for Step Input
Voltage Step
Time
V_i
V_O
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Steps for Sinusoidal Input with f=33.3Hz
Voltage Step
Time
V_i
V_O
1.5
1
0.5
0
-0.5
-1
-1.5
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Steps for Gaussian Pulse Input
Voltage Step
time (s)
V_i
V_O
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
-0.2
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Frequency Domain for Step Input
Magnitude
Frequency
V_i
V_O
1400
1200
1000
800
600
400
200
0
-100
-80
-60
-40
-20
0
20
40
60
80
100

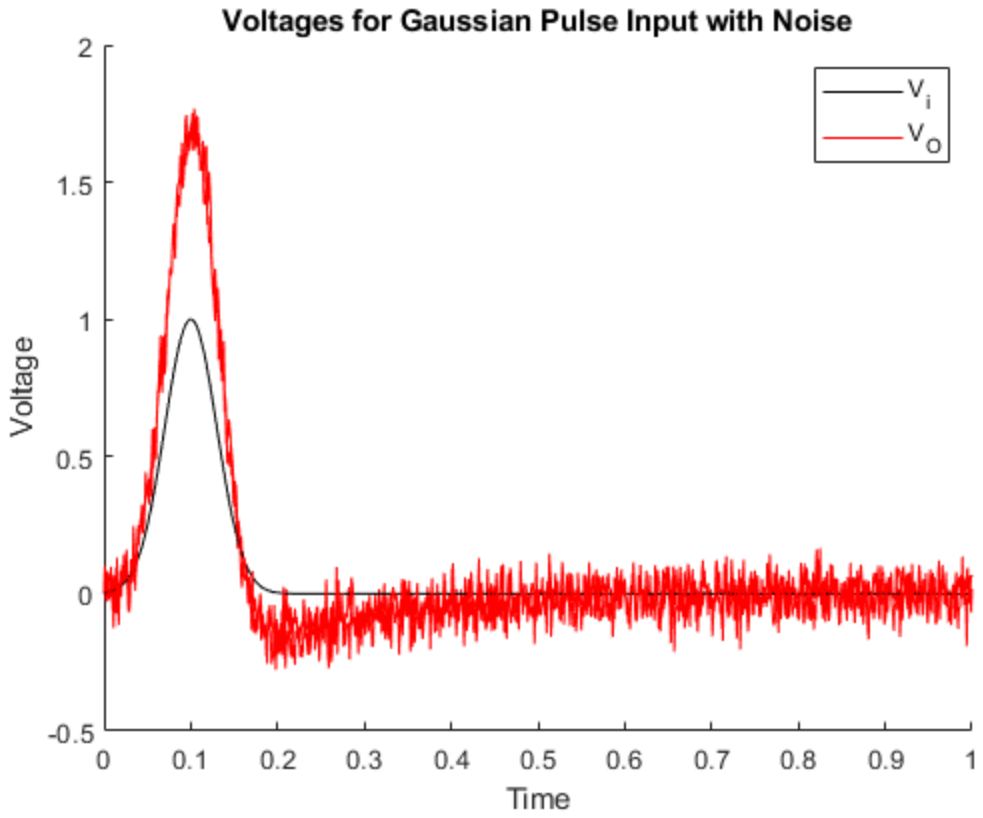

Frequency Domain for Sinusoidal Input with f=33.3Hz
Magnitude
Frequency
V_i
V_O
600
500
400
300
200
100
0
-200
-150
-100
-50
0
50
100
150
200

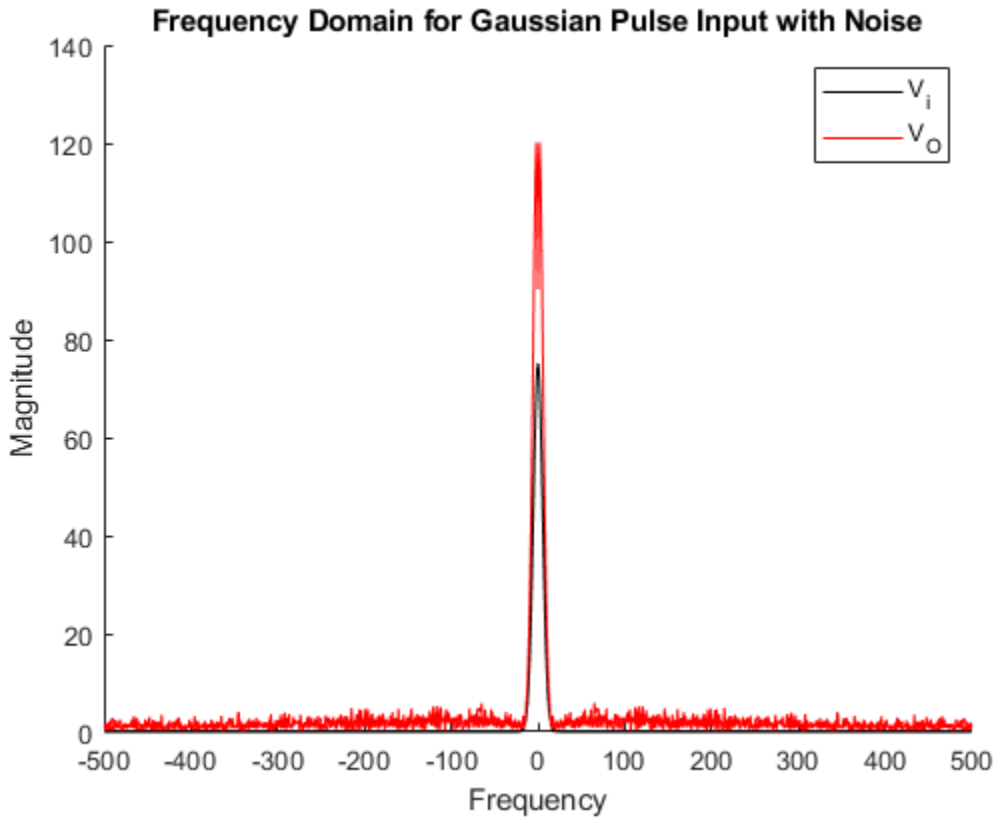

Frequency Domain for Gaussian Pulse Input
Magnitude
Frequency
V_i
V_O
120
100
80
60
40
20
0
-100
-80
-60
-40
-20
0
20
40
60
80
100

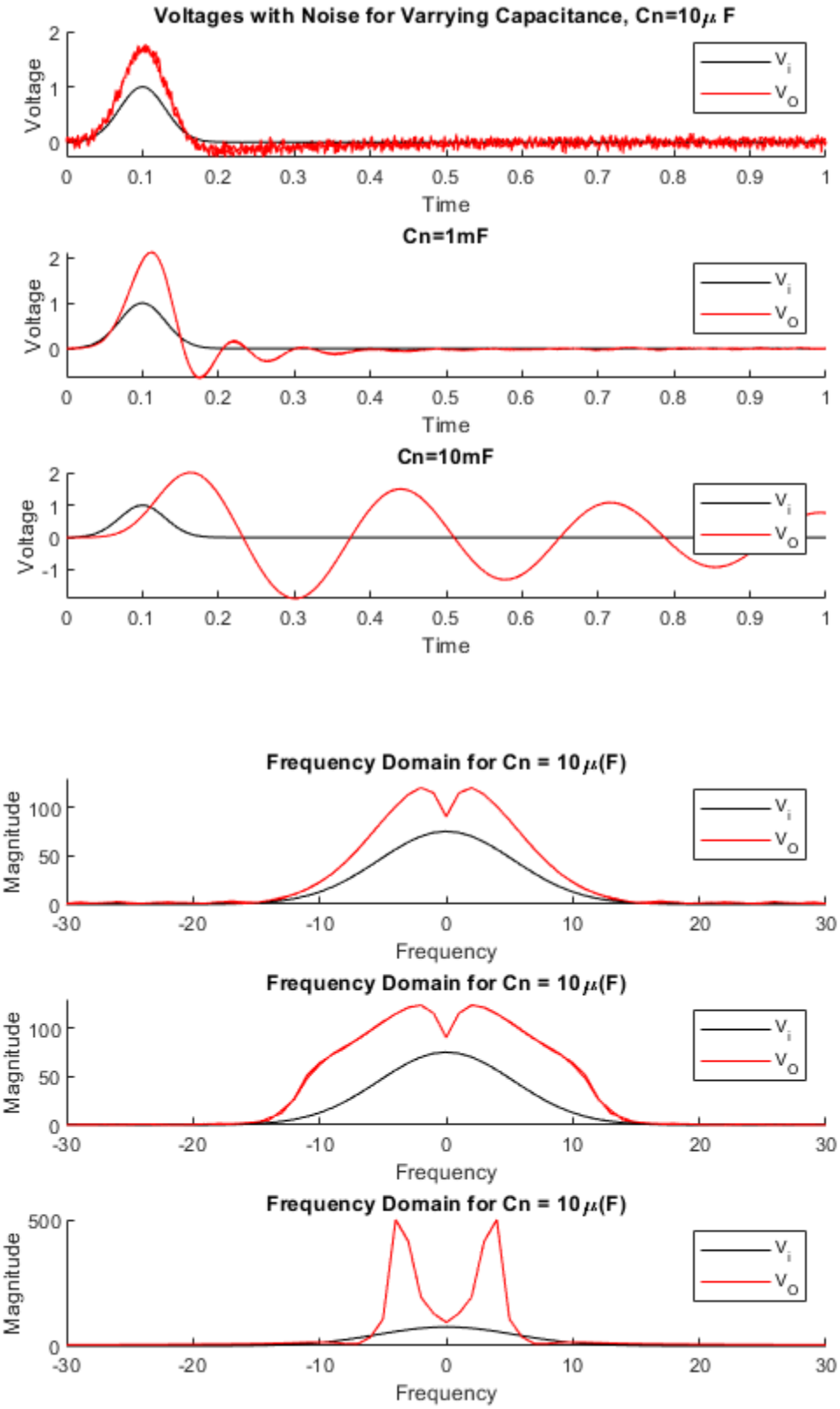
Part 3

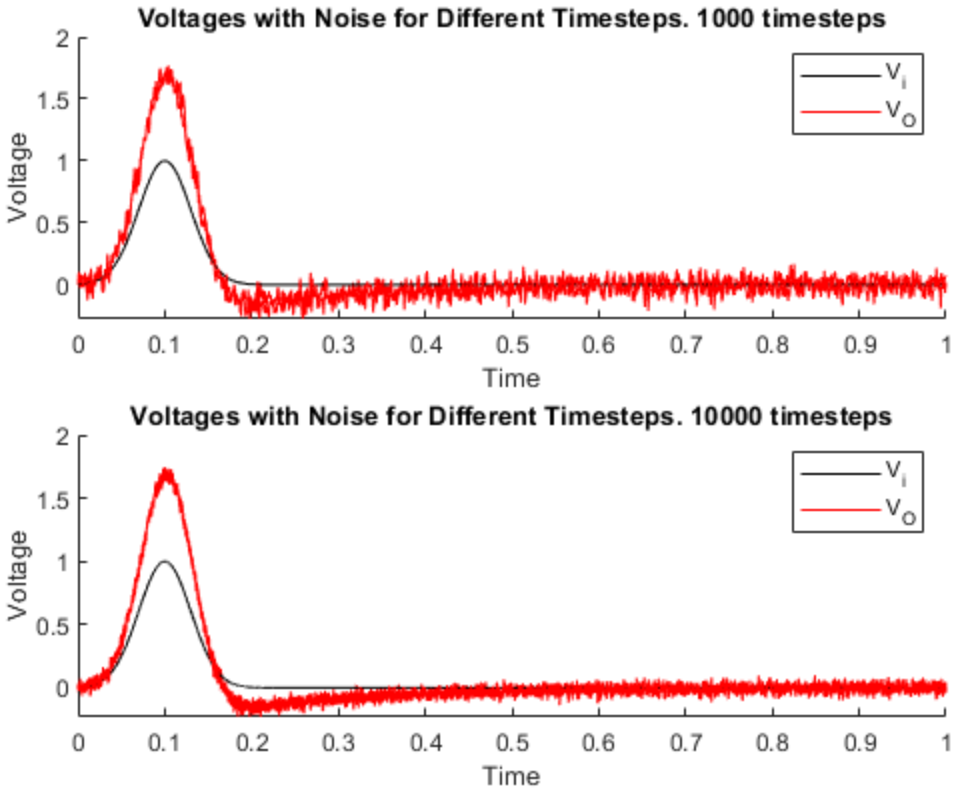
Different inputs with noise were modeled. A gaussian input a sinusodal input and a step input. The frequency content of the signal was plotted using the FFT command. We also varied the capacitance which effects the bandwidth of the signal, and the time step was also varied which alows us to get a higher resolution for the output voltage. Bandwidth decreases as the capacitance of the noise signal is varied. The higher resolution that is created by varrying the time step allows us to get a more accurate representation of the output signal. As the capacitance is varied the signal overshoots and then levels off before finally reaching a steady state, this is because the circuit is transient in nature with energy-storage components like inductors and capacitors. The overshoots increase as the value of the capacitor increases. This makes sense as the time constant for the circuit will also be increasing at the same time.

```
Part3;


C =

         0         0         0         0         0         0
   -0.2500    0.2500         0         0         0         0
         0         0    0.0000         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0    0.2000


G =

    1.0000         0         0         0         0         0
   -1.0000    1.5000         0         0         0    1.0000
         0         0    0.0182         0         0   -1.0000
         0         0   -1.8182    1.0000         0         0
         0         0         0  -10.0000   10.0010         0
         0   -1.0000    1.0000         0         0         0
```


Voltages for Gaussian Pulse Input with Noise
Voltage
Time
V_i
V_o
2
1.5
1
0.5
0
-0.5
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Frequency Domain for Gaussian Pulse Input with Noise
Magnitude
Frequency
V_i
V_o
140
120
100
80
60
40
20
0
-500
-400
-300
-200
-100
0
100
200
300
400
500


Voltages with Noise for Varrying Capacitance, Cn=10μ F
Voltage
Time
Vi
Vo
Cn=1mF
Cn=10mF
Frequency Domain for Cn = 10μ(F)
Magnitude
Frequency

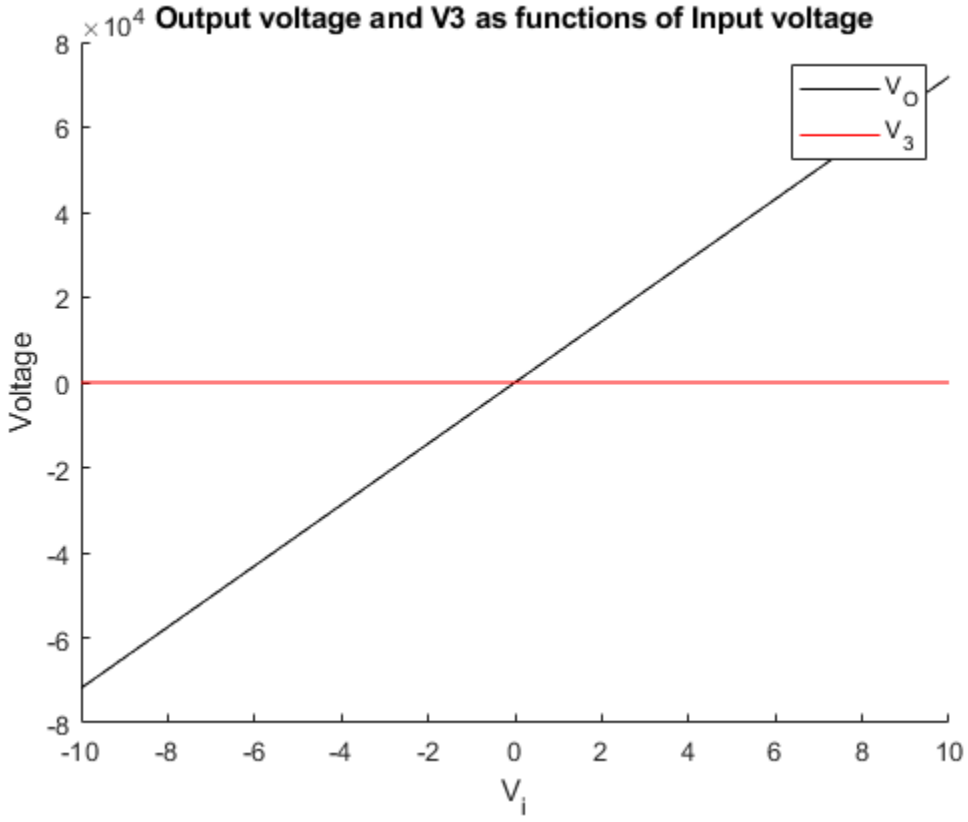


Part 4

Modeling equation would change the equations by adding another vector. A column vector would be added with a non-linear expression for the conductance of the device in question. The updated matrix equation would look like $C * (dV/dt) + GV + B = F(\omega)$ in order to implement this equation we would need to use an iterative approach to solve for the current I_3 and then use this updated value to forward into our system of equations, getting closer and closer to the final solution and finally, depending on the error of convergence the final solution is attained for all practical Engineering purposes.

```
Part4;


C =

         0         0         0         0         0         0
   -0.2500    0.2500         0         0         0         0
         0         0    0.0000         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0         0
         0         0         0         0         0    5.0000


G =

   1.0e+04 *
```

```
    0.0001         0         0         0         0         0
   -0.0001    0.0001         0         0         0    0.0001
         0         0    0.0000         0         0   -0.0001
         0         0   -1.0909    0.0001         0         0
         0         0         0   -0.0010    0.0010         0
         0   -0.0001    0.0001         0         0         0
```



Published with MATLAB® R2019b