ELEC 4700

Assignment 4 – Circuit Modelling

Miranda Heredia

100996160

April 3, 2020

```
%% Assignment 4 - Part 1
% This is an extension of PA 7

clear
close all
%% A) G and C Matrices

global G C b

% initialize matrices
G = zeros(5,5);
C = zeros(5,5);
b = zeros(5,1);

%---------------------------------------------------------------------------
% List of the components (netlist):
%---------------------------------------------------------------------

% Part A) G and C matrices

R1 = 1;
C1 = 0.25;
R2 = 2;
L = 0.2;
R3 = 3.07;
a = 100;
R4 = 0.1;
R5 = 1000;

vol(1,0,1);
res(1,2,R1);
cap(1,2,C1);
res(2,0,R2);
ind(2,3,L);
vcvs(4,0,3,0,(a/R3));
res(3,0,R3);
res(4,5,R4);
res(5,0,R5);
disp('The G matix is:')
disp(G)
disp('The C matix is:')
disp(C)

%% B) DC Sweep

%sweeping the input voltage V1 from -10V to 10V

Vin = linspace(-10, 10, 50);
steps = 50;
for n = 1:steps
    b(6) = Vin(n);
    X = G\b;
    V3(n) = X(3);
    VO(n) = X(5);
    gain(n) = VO(n)/Vin(n);
end
```

```
figure(1)
plot(Vin, V3, Vin, VO);
legend('Voltage @ Node 3', 'Voltage @ Output');
xlabel('Vin')
title('Plot of DC Sweep')

%% B) AC Sweeps

% i) Plot AC sweep VO as a function of w(omega) and the gain
F = logspace(0,9,5000);
OutputNode = 5;
for n=1:length(F)
    w = 2*pi*F(n);
    s = 1i*F(n);
    A = G + (s*C);

    X = A\b;
    Vout(n) = abs(X(OutputNode));
    gain(n) = 20*log(abs(X(OutputNode)));
end
figure(2);
semilogx(F, Vout);
xlabel('Frequency (Hz)');
ylabel('Vout (V)');
title('Frequncy Response')

%Gain Plot

figure(3);
semilogx(F, gain);
xlabel('Frequency (Hz)');
ylabel('Gain (dB)');
title('Gain Response');

% iii) Random Perturbations


Cs = 0.25 + 0.05*randn(1,1000);
for n = 1:1000
    s = 1i*pi;

    C(1,1) = Cs(n);
    C(2,2) = Cs(n);
    C(1,2) = Cs(n)*-1;
    C(2,1) = Cs(n)*-1;

    A = G +(s*C);
    X = A\b;
    gainP(n) = 20*log10(abs(X(OutputNode)));
end
figure(4)
histogram(gainP)
```

A) C and G Matrices

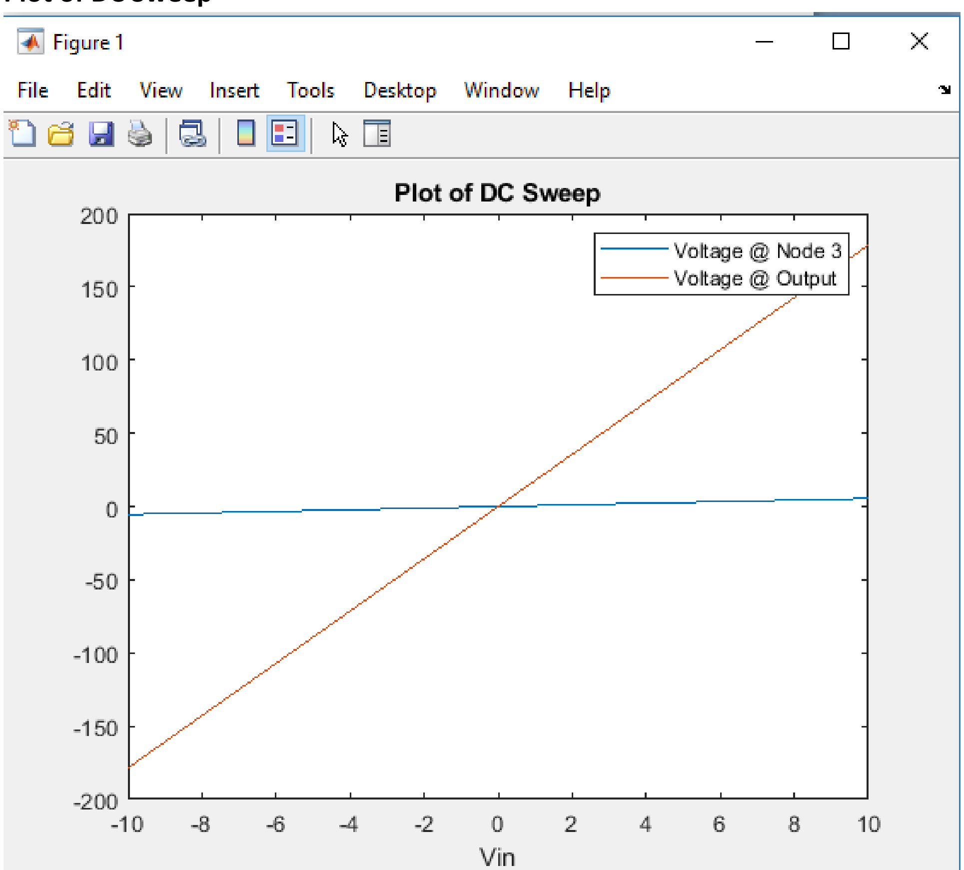
The G matix is:

1.0000	-1.0000	0	0	0	1.0000	0	0
-1.0000	1.5000	0	0	0	0	1.0000	0
0	0	0.3257	0	0	0	-1.0000	0
0	0	0	10.0000	-10.0000	0	0	1.0000
0	0	0	-10.0000	10.0010	0	0	0
1.0000	0	0	0	0	0	0	0
0	1.0000	-1.0000	0	0	0	0	0
0	0	-32.5733	1.0000	0	0	0	0

The C matix is:

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

B) Plot of DC Sweep



C) Gain Response

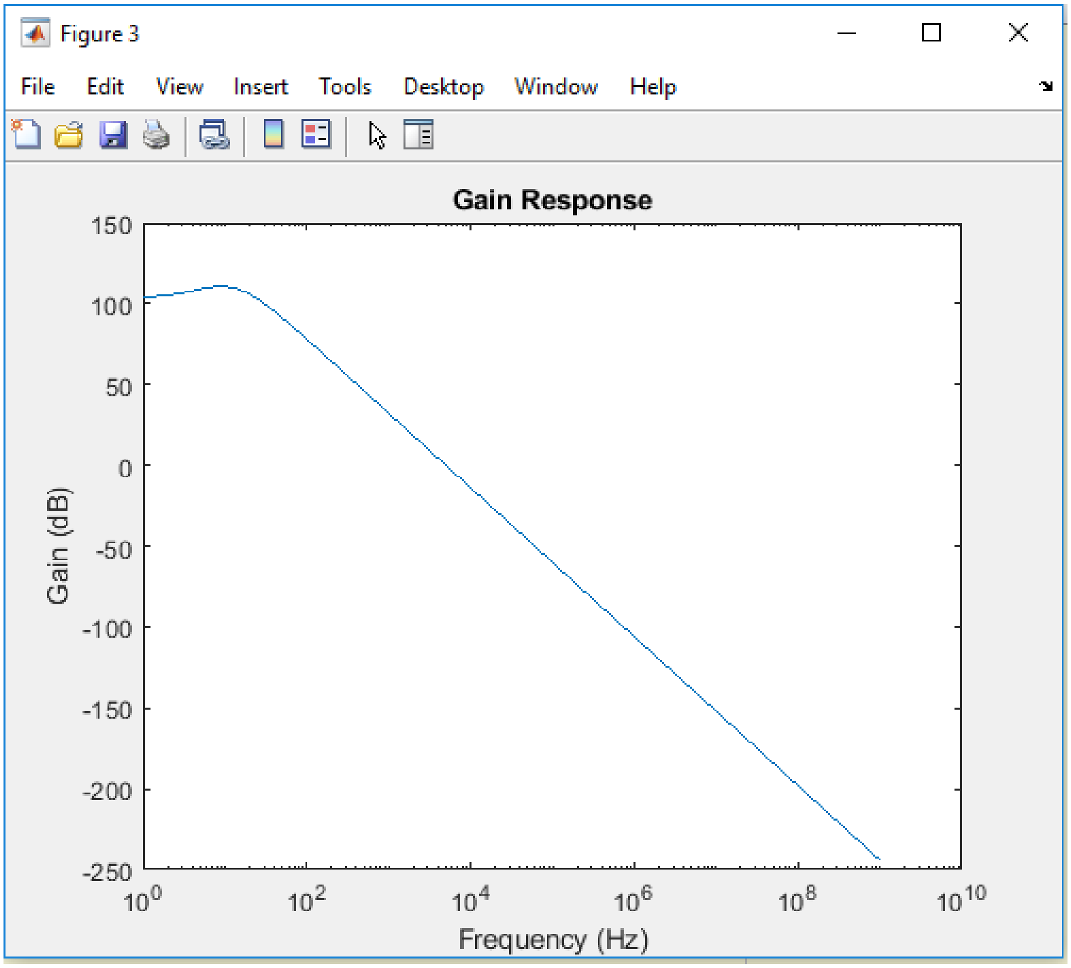
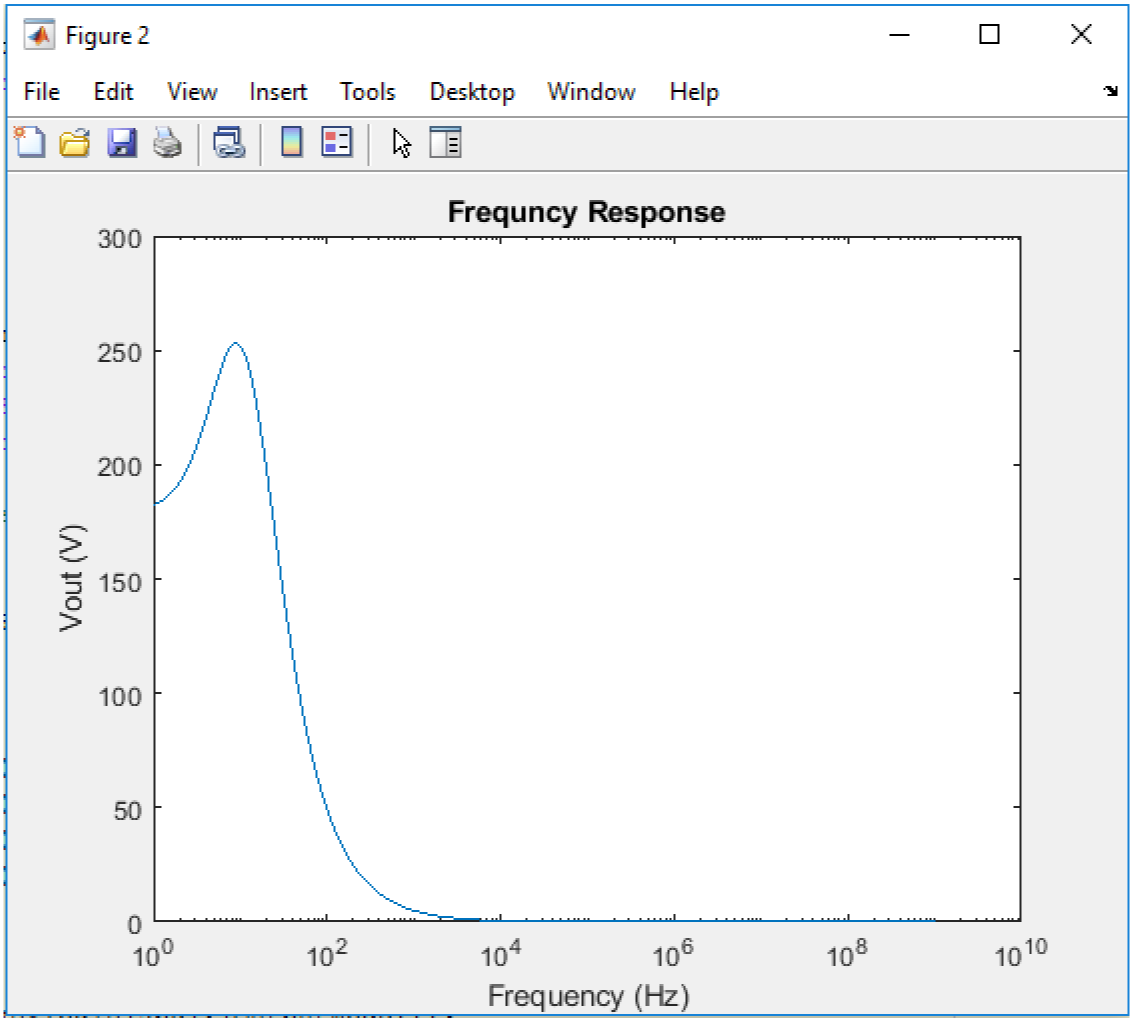


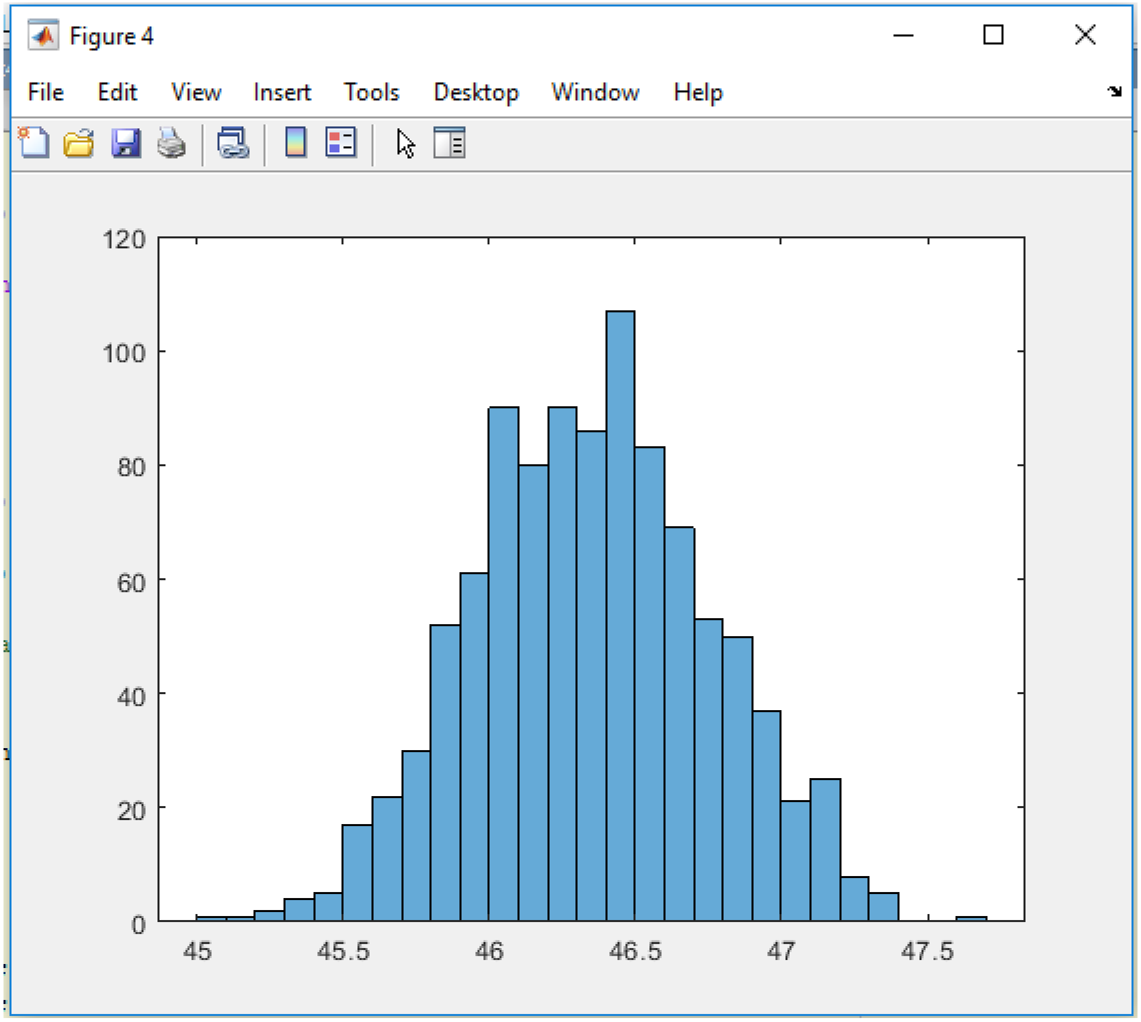
Figure 3
File Edit View Insert Tools Desktop Window Help
Gain Response
150
100
50
0
-50
-100
-150
-200
-250
Gain (dB)
10^0
10^2
10^4
10^6
10^8
10^10
Frequency (Hz)

D)

Frequency Response of AC case plot



E) <u>Gain as Function of Random Perturbations</u>



Part 2: Transient Circuit Simulation

B)

This appears to be an RLC circuit because there are capacitors, resistors and inductors.

C)

RLC circuits go through resonance at a certain frequqency (resonant frequency) and this is when then the capacitors and inductors are equal in magnitude and cancel each other out. At this frequency the gain of the circuit sharply rises, and is low at other frequencies. For this reason, this circuit has a very sharp lowpass response

Code:

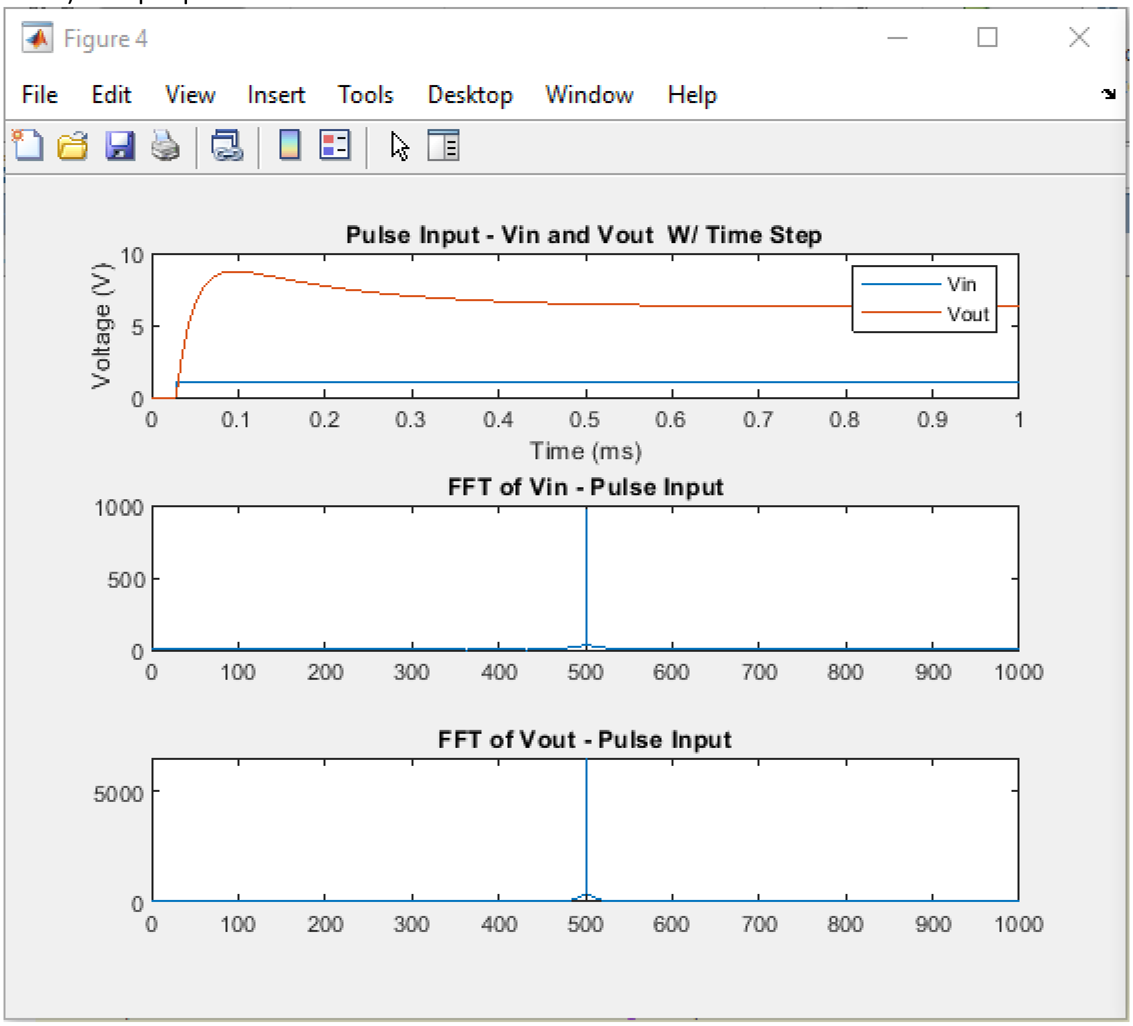
```
clc
clear all

%Using stamps from ELEC 4506
global G C b;

G = zeros(5,5);
C = zeros(5,5);
b = zeros(5,1);

R1 = 1;
C1 = 0.25;
R2 = 2;
L = 0.2;
R3 = 10;
a = 100;
R4 = 0.1;
R5 = 1000;

vol(1,0,1);
res(1,2,R1);
cap(1,2,C1);
res(2,0,R2);
ind(2,3,L);
vcvs(4,0,3,0,(a/R3));
res(3,0,R3);
res(4,5,R4);
res(5,0,R5);

%% Time Step Input 0 to 1 at 0.03s
%Using trapezoidal rule from ELEC 4506

h = 1/1000;
bn=zeros(length(b));
bn1=zeros(length(b));
xn=zeros(length(b));
time = linspace(0,1,1000);

for n=2:numel(time)
    pulse = in(1);
    %Trapezoidal Rule
    bn1(6) = pulse(n);
    bn(6) = pulse(n-1);
    trap =(2*C/h-G)*xn+bn1+bn;
    xn1=(2*C/h + G)\trap;
    Vout1(n) = xn(5)*2;
    xn = xn1;

    Vin(n-1) = xn(1);
    Vout(n-1) = xn(5);
end

Vin(n) = xn(1);
Vout(n) = xn(5);
```

```
figure(4)
subplot(3,1,1)
plot(time,Vin,time,Vout)
title('Pulse Input - Vin and Vout  W/ Time Step')
legend('Vin', 'Vout')
xlabel ('Time (ms)')
ylabel('Voltage (V)')
% Fourier
subplot(3,1,2)
plot(abs(fftshift(fft(Vin))))
title('FFT of Vin - Pulse Input')

subplot(3,1,3)
plot(abs(fftshift(fft(Vout))))
title('FFT of Vout - Pulse Input')

%% Sine Input

%Using trapezoidal rule from ELEC 4506
h = 1/1000;
bn=zeros(length(b));
bn1=zeros(length(b));
xn=zeros(length(b));
time = linspace(0,1,1000);

for n=2:numel(time)
    sinIn = in(2);
    %Trapezoidal Rule
    bn1(6) = sinIn(n);
    bn(6) = sinIn(n-1);
    trap =(2*C/h-G)*xn+bn1+bn;
    xn1=(2*C/h + G)\trap;
    Vout1(n) = xn(5)*2;
    xn = xn1;

    Vin(n-1) = xn(1);
    Vout(n-1) = xn(5);
end

Vin(n) = xn(1);
Vout(n) = xn(5);

figure(5)
subplot(3,1,1)
plot(time,Vin,time,Vout)
title('Sine Input - Vin and Vout')
legend('Vin', 'Vout')
xlabel ('Time (ms)')
ylabel('Voltage (V)')
% Fourier
subplot(3,1,2)
plot(abs(fftshift(fft(Vin))))
title('FFT of Vin - Sine Input')

subplot(3,1,3)
plot(abs(fftshift(fft(Vout))))
title('FFT of Vout - Sine Input')
```

```
%%  Gaussian Pulse Input

%Using trapezoidal rule from ELEC 4506

h = 1/1000;
bn=zeros(length(b));
bn1=zeros(length(b));
xn=zeros(length(b));
time = linspace(0,1,1000);

delay=0.06+(1-0.06)/2;

gdist=makedist('Normal','mu',delay,'sigma',0.03);
gpdf=pdf(gdist,time);
gpulse=gpdf/max(gpdf);

for n=2:numel(time)

    %Trapezoidal Rule
    bn1(6) = gpulse(n);
    bn(6) = gpulse(n-1);
    trap =(2*C/h-G)*xn+bn1+bn;
    xn1=(2*C/h + G)\trap;
    Vout1(n) = xn(5)*2;
    xn = xn1;

    Vin(n-1) = xn(1);
    Vout(n-1) = xn(5);
end

Vin(n) = xn(1);
Vout(n) = xn(5);

figure(6)
subplot(3,1,1)
plot(time,Vin,time,Vout)
title('Gaussian Pulse Input - Vin and Vout')
legend('Vin', 'Vout')
xlabel ('Time (ms)')
ylabel('Voltage (V)')

% Fourier
subplot(3,1,2)
plot(abs(fftshift(fft(Vin))))
title('FFT of Vin - Gaussian Pulse Input')
xlim([440 560])

subplot(3,1,3)
plot(abs(fftshift(fft(Vout))))
title('FFT of Vout - Gaussian Pulse Input')
xlim([440 560])
```

%% Increasing Time Step
 Increasing the time step decreases the accuracy of the model this is because for a bigger time step, the guess is going to be less accurate from FD method (trap)

1) Step Input



2) Sine input

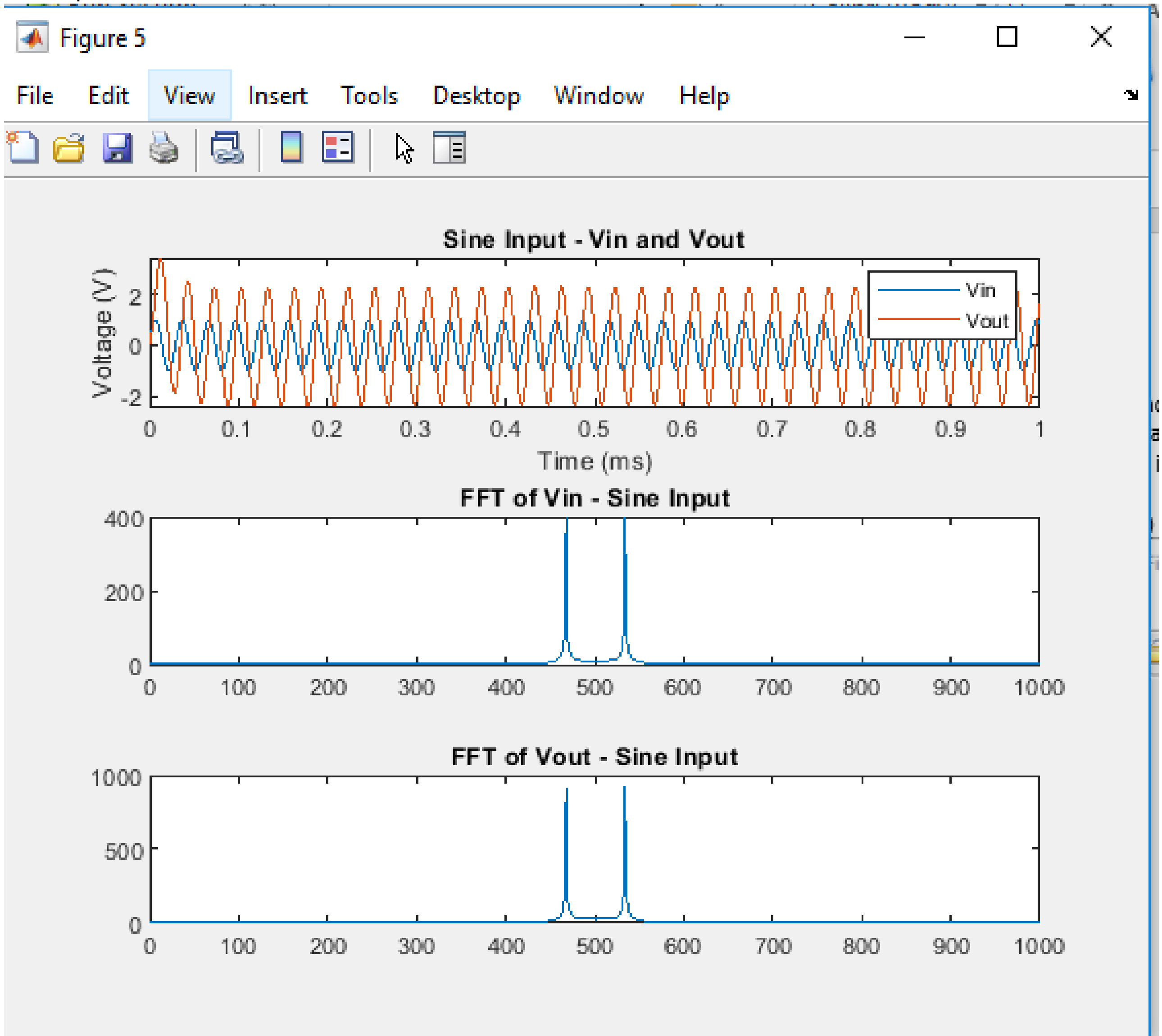
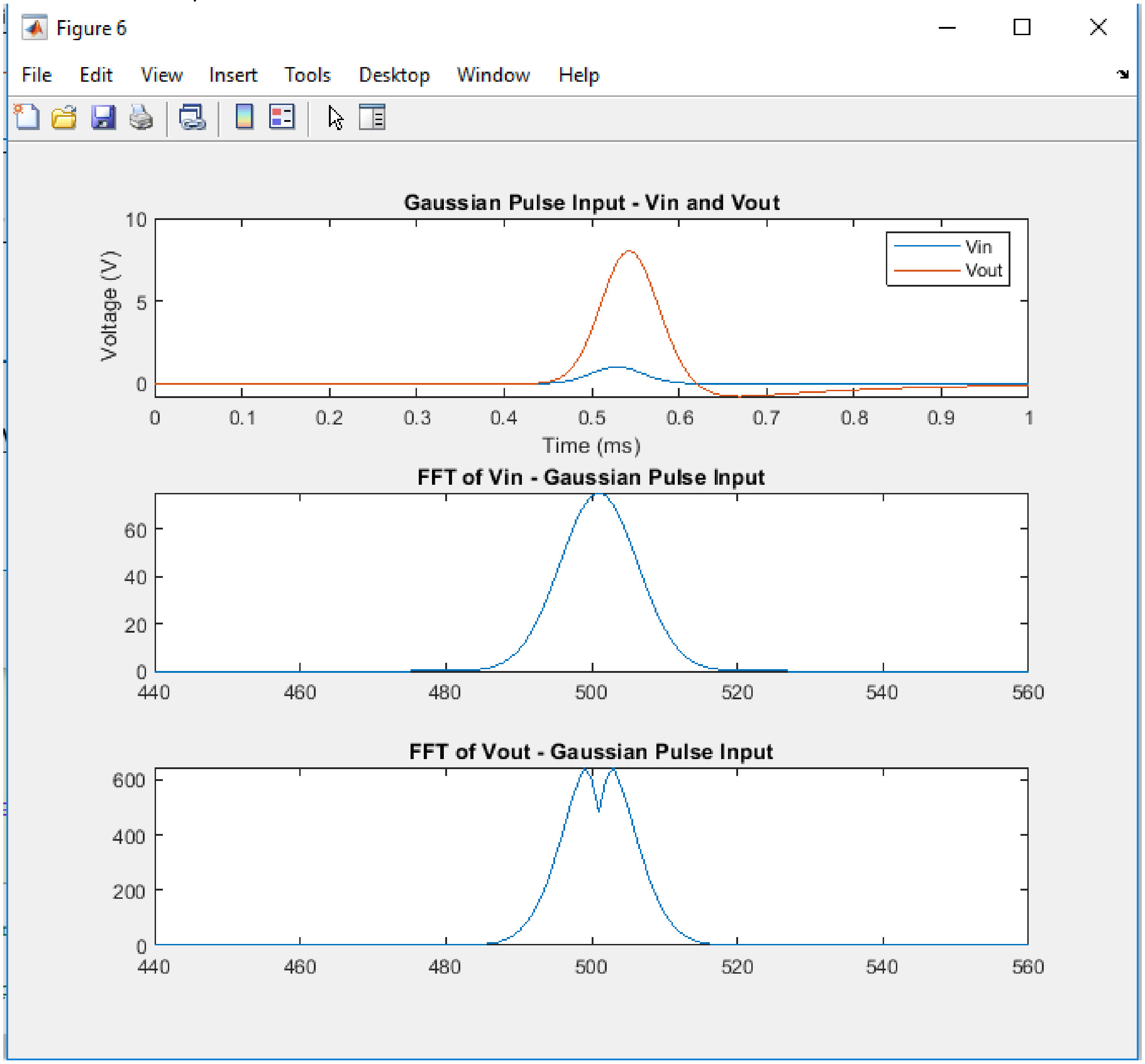


Figure 5
File Edit View Insert Tools Desktop Window Help
Sine Input - Vin and Vout
Vin
Vout
Voltage (V)
Time (ms)
FFT of Vin - Sine Input
FFT of Vout - Sine Input

3) Gaussian Pulse Input



Part 3) Circuit with Noise

Updated C Matrix:

C =

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

```
% Assignment 4

%% Part 3

clearvars;
clear;
clear all;
close all;
clc;

%%
% Use stamps to generate the desired MNA
% matrices. These stamps were developed in ELEC 4506.

global G C b; %define global variables
G = zeros(6,6); % Define G, 5 node circuit (do not include additional
variables)
C = zeros(6,6); % Define C, 5 node circuit (do not include additional
variables)
b = zeros(6,1); % Define b, 5 node circuit (do not include additional
variables)

vol(1,0,10);
cur(3,4,0.001)
% Use stamp for current controlled voltage source
ccvs(5,0, 4,0, 100);

res(1,2,1);
res(2,0,2);
res(3,4,3.07);
res(5,6,0.1);
res(6,0,1000);

cap(1,2,0.25);
cap(3,4,0.00001);
ind(2,3,0.2);

%% A
```

```
% The C matrix is now:
C

%% B
% A noise source is added by setting In to 0.001*randn().

Xold=zeros(10,1);

h=1/1000;
Vin = zeros(1000,1);
Vout = zeros(1000,1);
for count = 1:1000
    t=count*h;
    % Gaussian pulse, shifted by 0.06s and compressed to have std deviation
    % of 0.03.
    Vin(count) = exp(-0.5*((t-0.06)/0.03)^2);
end

b(7) = Vin(1);
for count = 1:1000
    bn1 = b;
    bn1(3)=0.005*randn();
    bn1(4)=-bn1(3);
    bn1(7) = Vin(count);
    Xnew = (G+(2*C/h))\((2*C/h - G)*Xold+b+bn1);
    Vout(count) = Xnew(6);

    b = bn1;
    Xold = Xnew;
end


fftVin = abs(fftshift(fft(Vin)));
fftVout = abs(fftshift(fft(Vout)));
n=length(fftVin);
fs=1/h;
fshift=(-n/2:n/2-1)*(fs/n);

%%
% The figures below contain the time domain and frequency domain response
% of the circuit to a Gaussian input voltage with a noise source.

figure;
plot(linspace(0,1,1000),Vin)
xlabel('Time (s)')
ylabel('Input Voltage (Volts)')
title('Input Voltage Over Time - Cn = 0.00001')

figure;
plot(linspace(0,1,1000),Vout)
xlabel('Time (s)')
ylabel('Output Voltage (Volts)')
title('Output Voltage over Time - Cn = 0.00001')
```

```
figure;
plot(fshift, fftVin);
xlabel('Frequency (Hz)')
title('Frequency Response of Input Voltage - Cn = 0.00001')

figure;
plot(fshift, fftVout);
xlabel('Frequency (Hz)')
title('Frequency Response of Output Voltage - Cn = 0.00001')

%% E
% Obtain 2nd plot of Vout. Use Cn=0.01.

C(3,3)=0.01;
C(4,4)=0.01;
C(3,4)=-0.01;
C(4,3)=-0.01;

Xold=zeros(10,1);
b(7) = Vin(1);
for count = 1:1000
    bn1 = b;
    bn1(3)=0.005*randn();
    bn1(4)=-bn1(3);
    bn1(7) = Vin(count);
    Xnew = (G+(2*C/h))\((2*C/h - G)*Xold+b+bn1);
    Vout(count) = Xnew(6);

    b = bn1;
    Xold = Xnew;
end

fftVin = abs(fftshift(fft(Vin)));
fftVout = abs(fftshift(fft(Vout)));
n=length(fftVin);
fs=1/h;
fshift=(-n/2:n/2-1)*(fs/n);

figure;
plot(linspace(0,1,1000),Vout)
xlabel('Time (s)')
ylabel('Output Voltage (Volts)')
title('Output Voltage over Time - Cn = 0.01')

figure;
plot(fshift, fftVout);
xlabel('Frequency (Hz)')
title('Frequency Response of Output Voltage - Cn = 0.01')


%% E (2)
% Obtain 3rd plot of Vout. Use Cn = 1e-3.

C(3,3)=1e-3;
```

```
C(4,4)=1e-3;
C(3,4)=-1e-3;
C(4,3)=-1e-3;

Xold=zeros(10,1);
b(7) = Vin(1);
for count = 1:1000
    bn1 = b;
    bn1(3)=0.005*randn();
    bn1(4)=-bn1(3);
    bn1(7) = Vin(count);
    Xnew = (G+(2*C/h))\((2*C/h - G)*Xold+b+bn1);
    Vout(count) = Xnew(6);

    b = bn1;
    Xold = Xnew;
end

fftVin = abs(fftshift(fft(Vin)));
fftVout = abs(fftshift(fft(Vout)));
n=length(fftVin);
fs=1/h;
fshift=(-n/2:n/2-1)*(fs/n);

figure;
plot(linspace(0,1,1000),Vout)
xlabel('Time (s)')
ylabel('Output Voltage (Volts)')
title('Output Voltage over Time - Cn = 1e-3')

figure;
plot(fshift, fftVout);
xlabel('Frequency (Hz)')
title('Frequency Response of Output Voltage - Cn = 1e-3')

%%
% As can be seen in the plots above, the thermal noise is able to be seen.
% As Cn increases, the bandwidth of the noise decreases.

%% F
% Increase the time step.
h=1/500;
Vin = zeros(1/h,1);
Vout = zeros(1/h,1);
for count = 1:length(Vout)
    t=count*h;
    % Gaussian pulse, shifted by 0.06s and compressed to have std deviation
    % of 0.03.
    Vin(count) = exp(-0.5*((t-0.06)/0.03)^2);
end

C(3,3)=1e-5;
C(4,4)=1e-5;
C(3,4)=-1e-5;
C(4,3)=-1e-5;
```

```
Xold=zeros(10,1);
b(7) = Vin(1);
for count = 1:length(Vout)
    bn1 = b;
    bn1(3)=0.005*randn();
    bn1(4)=-bn1(3);
    bn1(7) = Vin(count);
    Xnew = (G+(2*C/h))\((2*C/h - G)*Xold+b+bn1);
    Vout(count) = Xnew(6);

    b = bn1;
    Xold = Xnew;
end

figure;
plot(linspace(0,1,length(Vout)),Vout)
xlabel('Time (s)')
ylabel('Output Voltage (Volts)')
title('Output Voltage over Time - t=2ms')

% Decrease the time step.
h=1/2000;
Vin = zeros(1/h,1);
Vout = zeros(1/h,1);
for count = 1:(1/h)
    t=count*h;
    % Gaussian pulse, shifted by 0.06s and compressed to have std deviation
    % of 0.03.
    Vin(count) = exp(-0.5*((t-0.06)/0.03)^2);
end

Xold=zeros(10,1);
b(7) = Vin(1);
for count = 1:(1/h)
    bn1 = b;
    bn1(3)=0.005*randn();
    bn1(4)=-bn1(3);
    bn1(7) = Vin(count);
    Xnew = (G+(2*C/h))\((2*C/h - G)*Xold+b+bn1);
    Vout(count) = Xnew(6);

    b = bn1;
    Xold = Xnew;
end

figure;
plot(linspace(0,1,length(Vout)),Vout)
xlabel('Time (s)')
ylabel('Output Voltage (Volts)')
title('Output Voltage over Time - t=0.5ms')
```

B) Vin with noise source

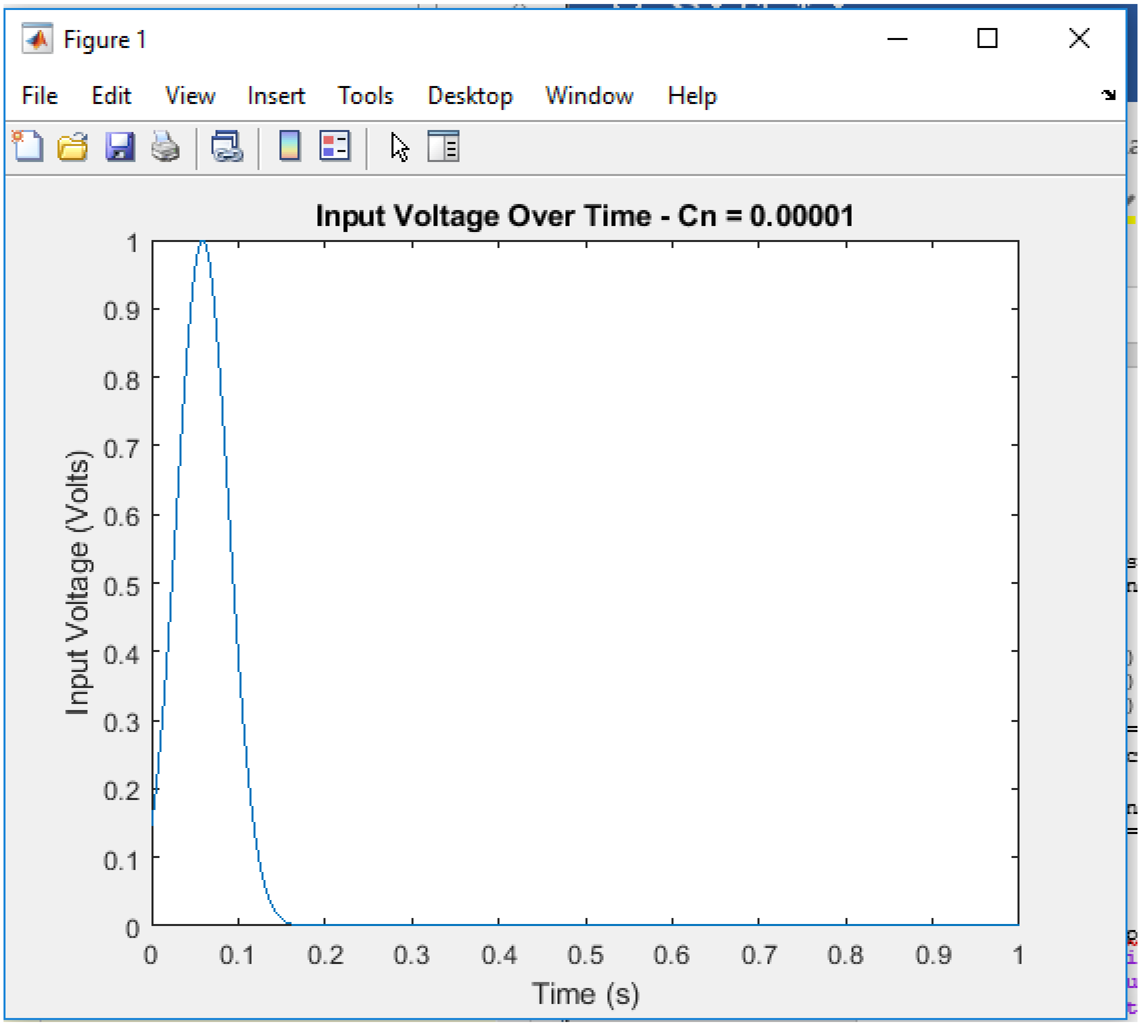
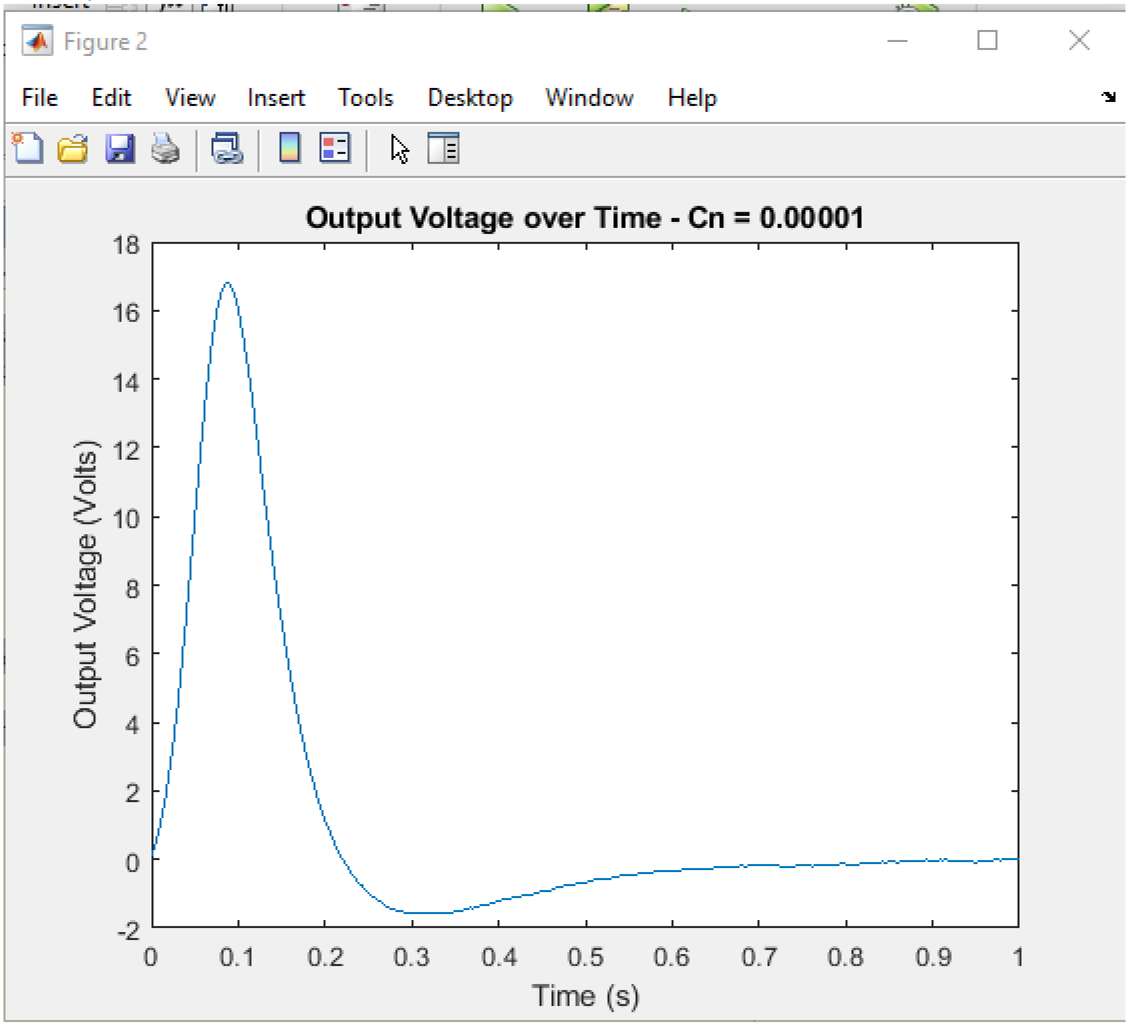
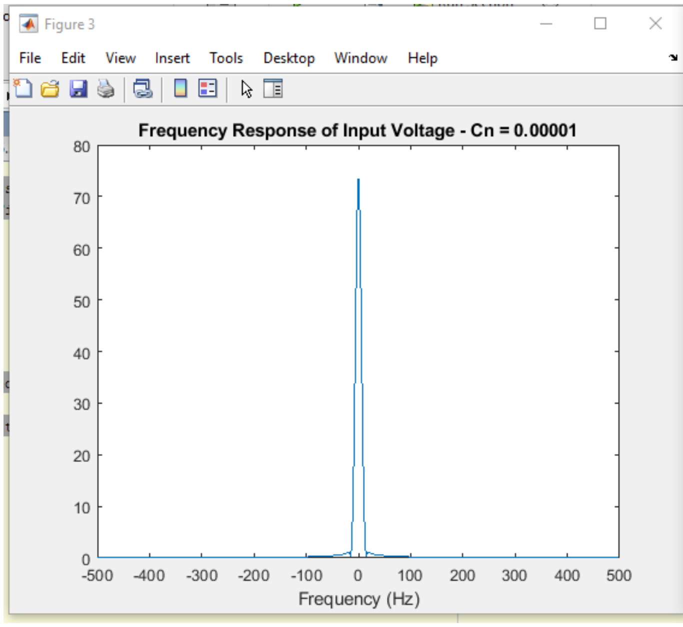


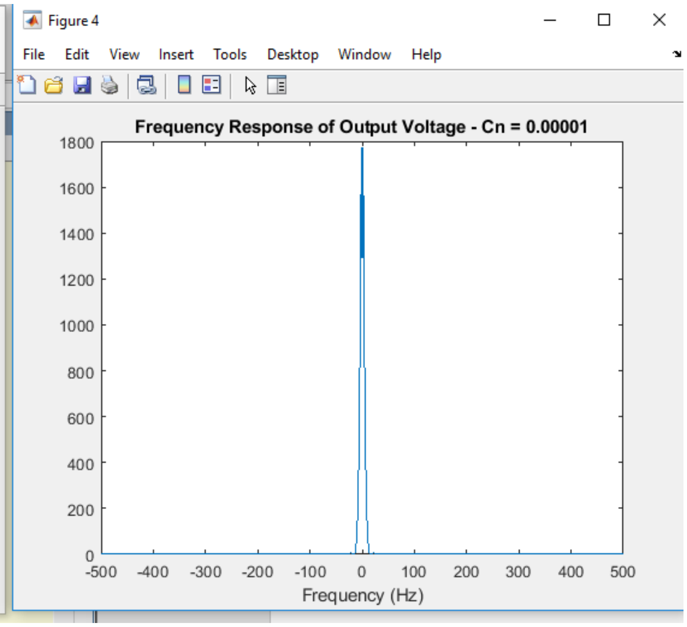
Figure 1
File Edit View Insert Tools Desktop Window Help
Input Voltage Over Time - Cn = 0.00001
Input Voltage (Volts)
Time (s)
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

Vout w/ Noise Source

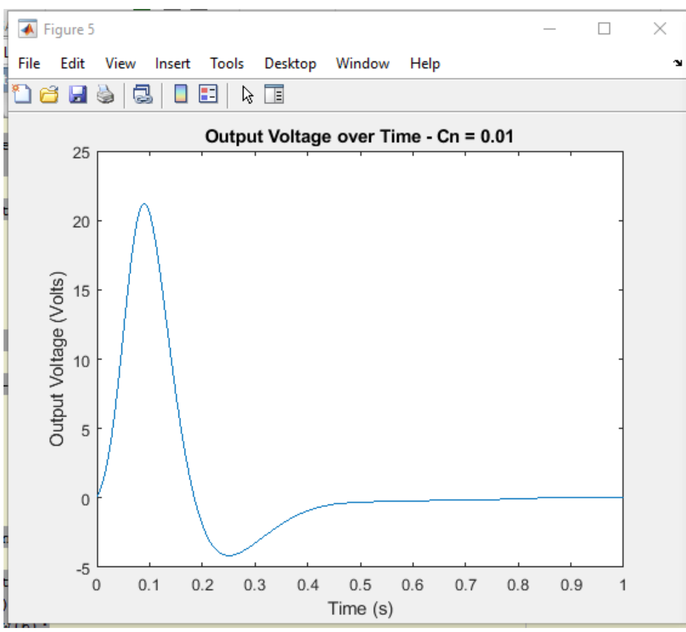


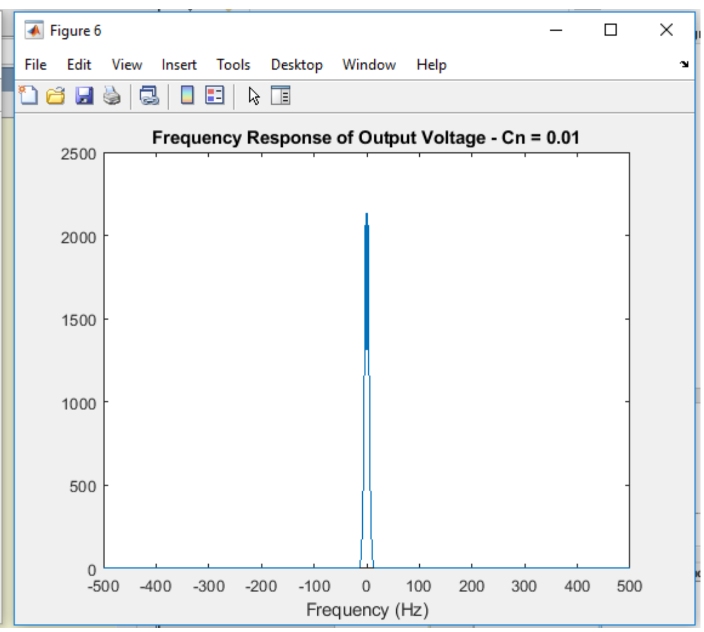
Fourier Transform Plot



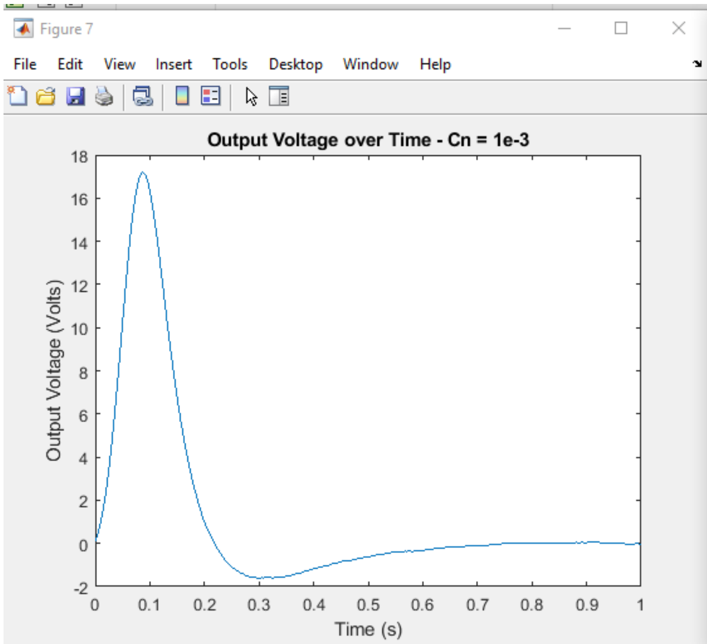


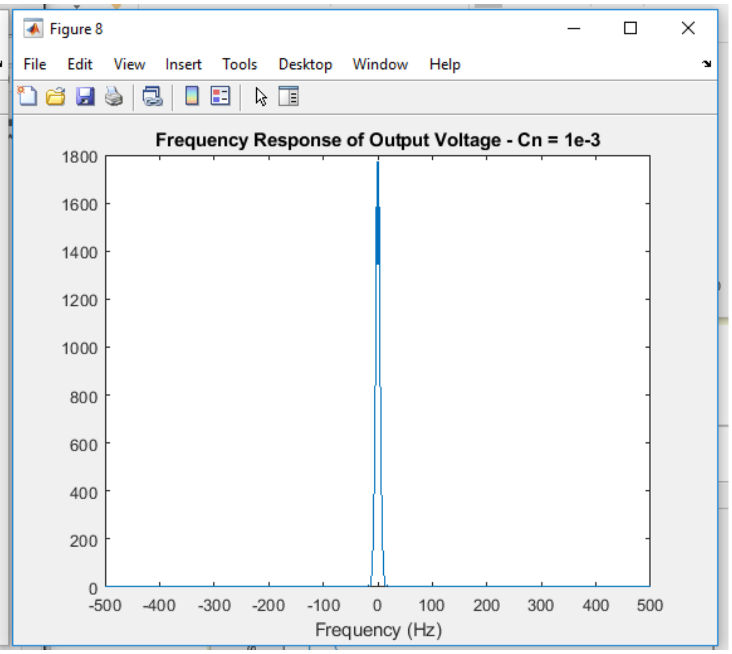
Change of Cn = 0.01



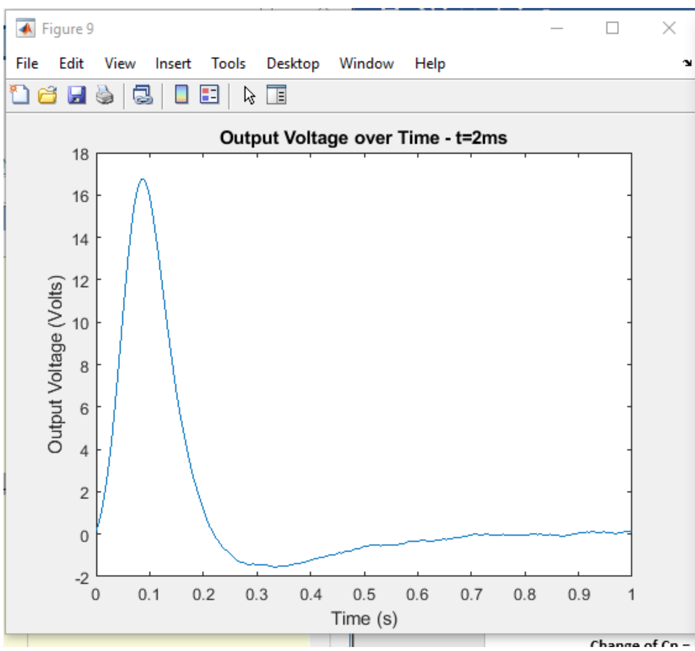


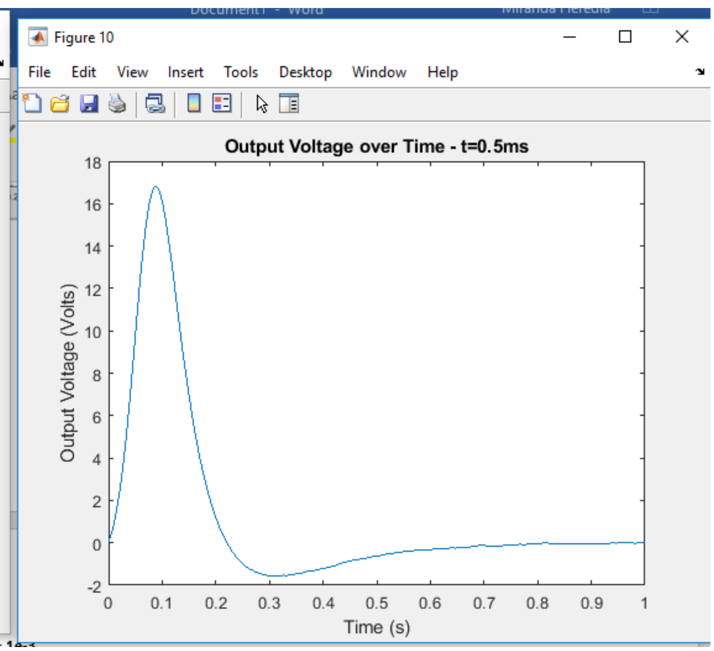
Change of Cn = 1e-3





Time Step Change





Part 4: Non-Linearity

% In order to deal with the non-linear element in the circuit, a new column matrix ('f(x)') is introduced into the equation (Gx +f(x) = b -- DC Case). Then, in order to solve the system, the jacobian of the matrix would need to be calculated. The jacobian is added to the derivative of the new column matrix (for the imaginary values). The sum is then divided by the sum of G*x+f(x)-b, giving the change in the result (delta X). Then, the Newton Raphson method is applied until the change in resulting values is
% sufficiently small, ideally close to 0. The time can be advanced and the jacobian+newton raphson method can be applied again. The trap is to march through time, where the jacobian is to find the DC solution at that time step.