Elec 4700 Assignment 4

Table of Contents

Joel Demetre (100943543)

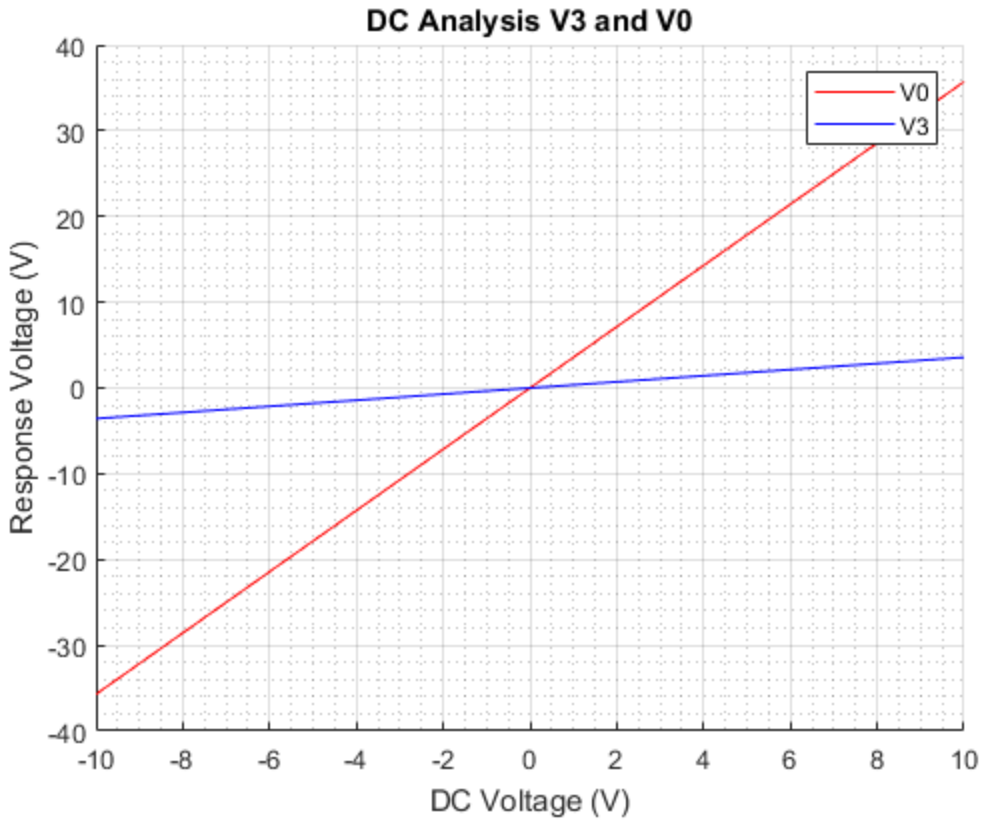
```
clear all;
close all;
%Parameters
G1 = 1;
Cx = 0.25;
G2 = 1/2;
L = 0.2;
G3 = 1/10;
Alpha = 100;
G4 = 1/0.1;
G0 = 1/1000;
```

Part 2 A)

The C and G Matrices

```
C = [ 0 0 0 0 0 0 0;
    -Cx Cx 0 0 0 0 0;
    0 0 -L 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;];


G = [ 1 0 0 0 0 0 0;
    -G2 G1+G2 -1 0 0 0 0;
    0 1 0 -1 0 0 0;
```

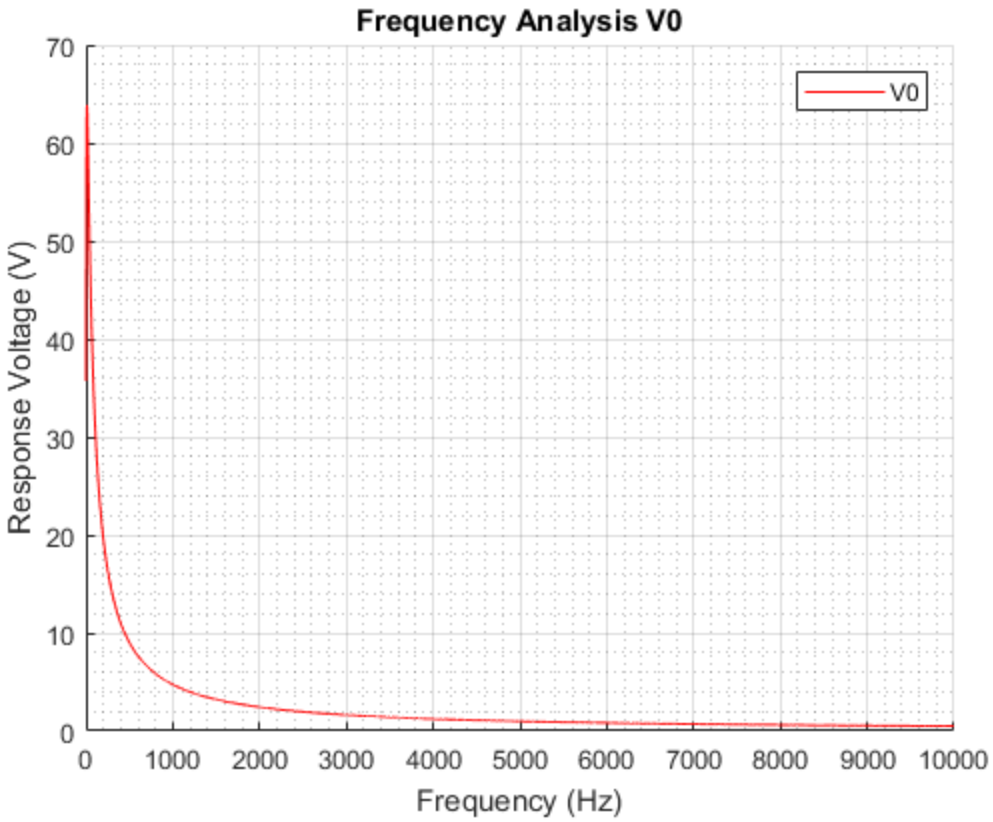
```
    0 0 -1 G3 0 0 0;
    0 0 0 0 -Alpha 1 0;
    0 0 0 G3 -1 0 0;
    0 0 0 0 0 -G4 G4+G0;];

F = [0; 0; 0; 0; 0; 0; 0;];

% Part B) Plot of DC Sweep
counter = 1;
for i = -10:0.2:10
    F(1) = i;
    V = G\F;
    out(counter , 1) = i;
    out(counter , 2) = V(7);
    out(counter , 3) = V(4);
    counter = counter + 1;
end
figure;
title('DC Analysis V3 and V0');
xlabel('DC Voltage (V)');
ylabel('Response Voltage (V)');
hold on;
grid on;
grid minor;
plot(out(:,1), out(:,2), 'r');
plot(out(:,1), out(:,3), 'b');
legend('V0', 'V3');



%Part C
clear out
counter = 1;
for i = 0:1:10000
    F(1) = 10;
    V = (G + i*C)\F;
    out(counter , 1) = i;
    out(counter , 2) = V(7);
    out(counter , 3) =V(7)/V(1);
    counter = counter + 1;
end
```

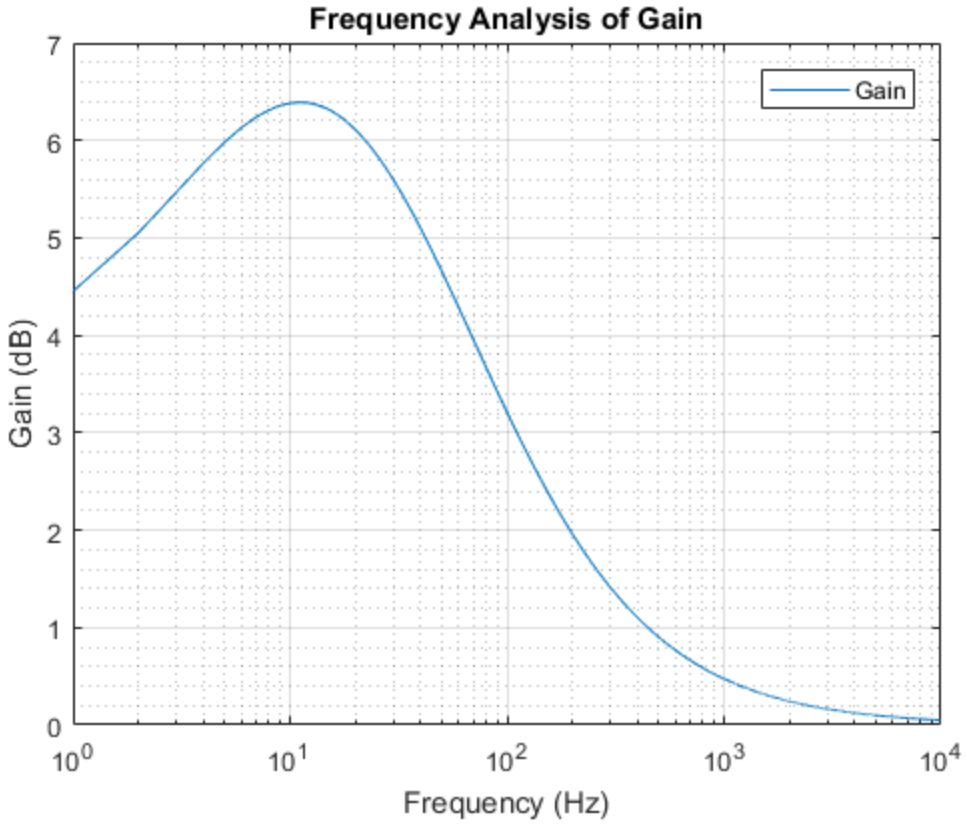


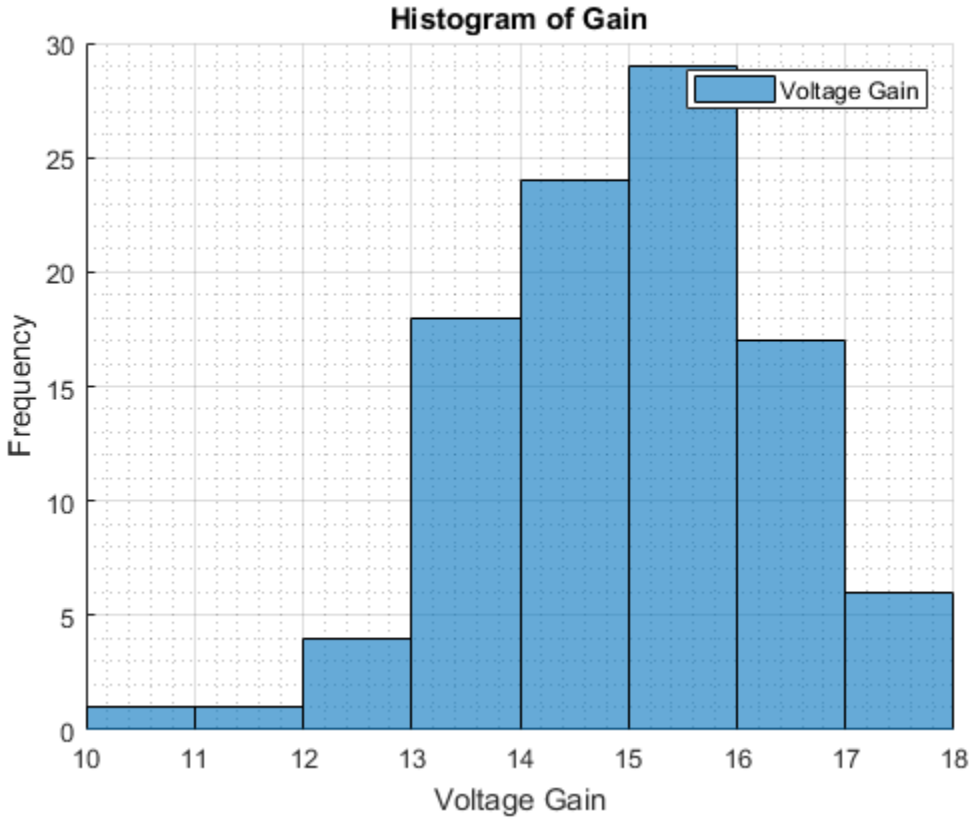
Part C) Plot of AC Analysis

```
figure;
title('Frequency Analysis V0');
xlabel('Frequency (Hz)');
ylabel('Response Voltage (V)');
hold on;
grid on;
grid minor;
plot(out(:,1), out(:,2), 'r');
legend('V0');
hold off;

figure;

semilogx(out(:,1), out(:,3));
legend('Gain');
title('Frequency Analysis of Gain');
xlabel('Frequency (Hz)');
ylabel('Gain (dB)');
hold on;
grid on;
grid minor;
```

```
clear out
counter = 1;
for i = 1:100
    C(2,1) = -normrnd(Cx, 0.05);
    C(2,2) = normrnd(Cx, 0.05);
    F(1) = 10;
    V = (G + pi*C)\F;
    out(counter , 1) = i;
    out(counter , 2) = 20*log10(V(7)/V(1));
    counter = counter + 1;
end
figure;
title('Histogram of Gain');
xlabel('Voltage Gain');
ylabel('Frequency');
hold on;
grid on;
grid minor;
histogram(out(:,2));
legend('Voltage Gain');
```




Frequency Analysis of Gain
Gain
Gain (dB)
Frequency (Hz)
0
1
2
3
4
5
6
7
10^0
10^1
10^2
10^3
10^4


Histogram of Gain
Voltage Gain
Frequency
Voltage Gain
0
5
10
15
20
25
30
10
11
12
13
14
15
16
17
18

Part D and Part E

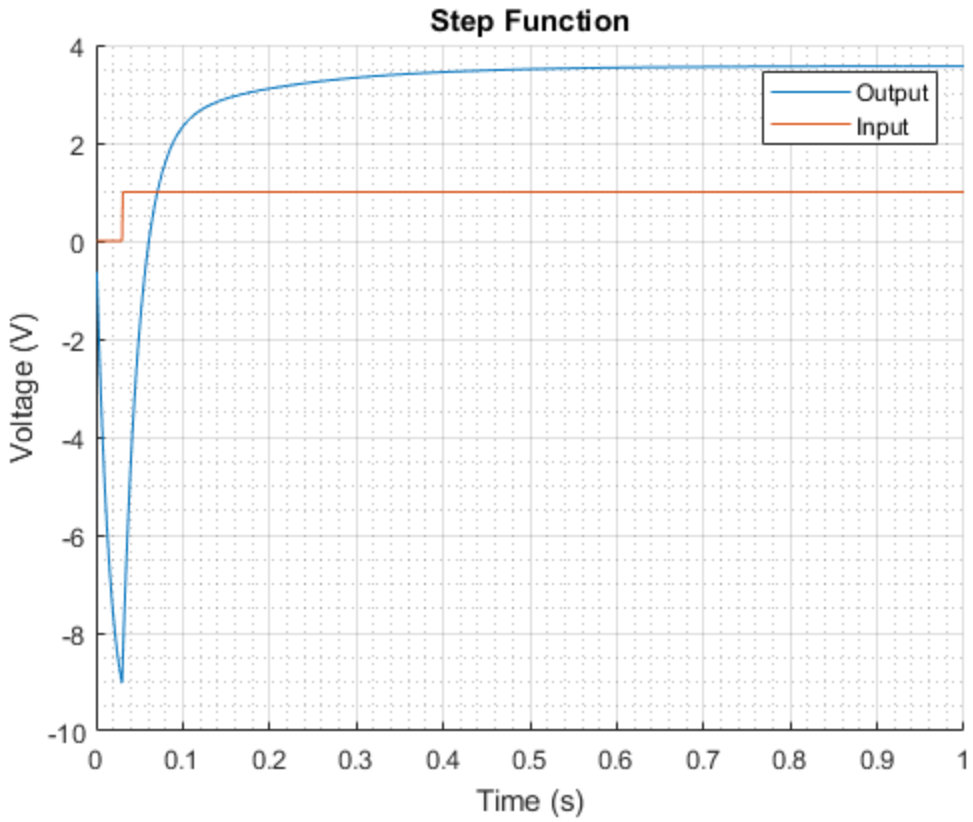
The Analysis of the Vin and Vout as well as the fourier transform for various functions such as step transition, guassian pulse and sine function.

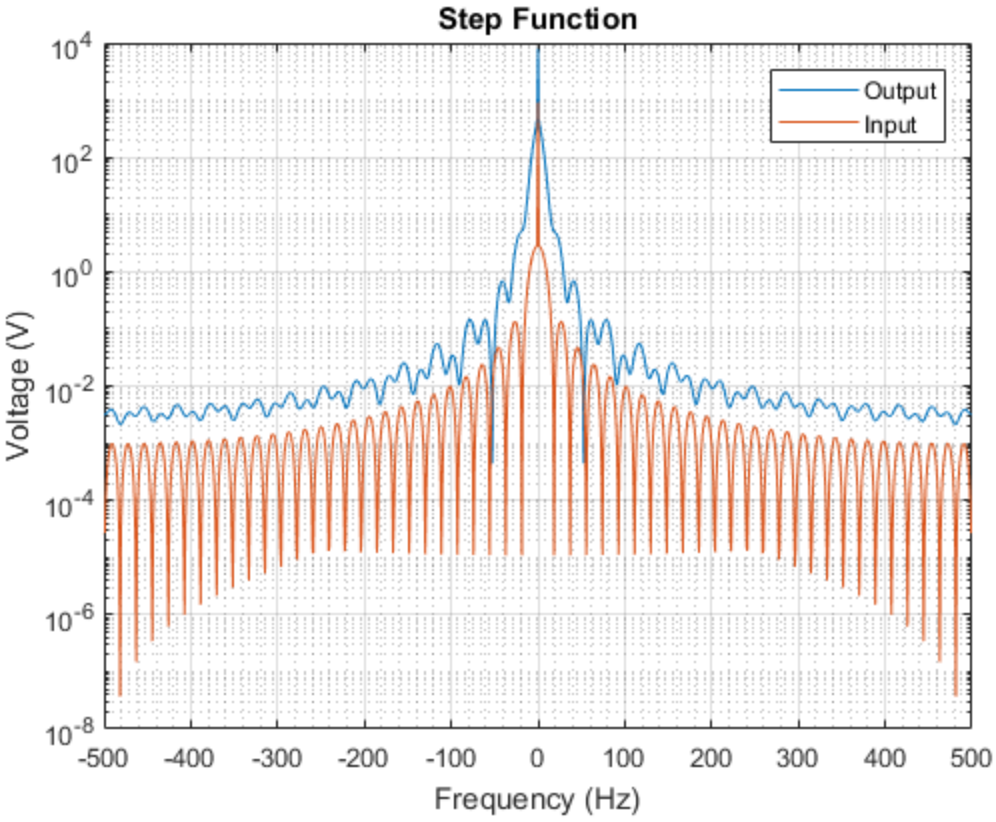
```
L = 1000;

clear out;
deltaT = 0.001;
A = (C/deltaT + G);
F = [0; 0; 0; 0; 0; 0; 0;];
Vold = [1; 0; 0; 0; 0; 0; 0;];
counter = 1;
for i = 0.001:deltaT:L*0.001
    if round(i*1000)>30
        F(1) = 1;
    end
    vNew = inv(A)*(C*Vold./deltaT + F);
    Vold = vNew;
    out(counter , 1) = i;
    out(counter , 2) = vNew(7);
    out(counter , 3) = vNew(1);
    counter = counter + 1;
end
figure;
hold on;
grid on;
grid minor;
title('Step Function');
xlabel('Time (s)');
ylabel('Voltage (V)');
plot(out(:,1), out(:,2));
plot(out(:,1), out(:,3));
legend('Output', 'Input');
hold off;

figure;

Fs = 1000;

n = 2^nextpow2(L);
Y = fft(out(:,2),n);
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
P = abs(Y/n);
semilogy(fshift, powershift) ;
hold on;
grid on;
grid minor;
title('Step Function');
xlabel('Frequency (Hz)');
```

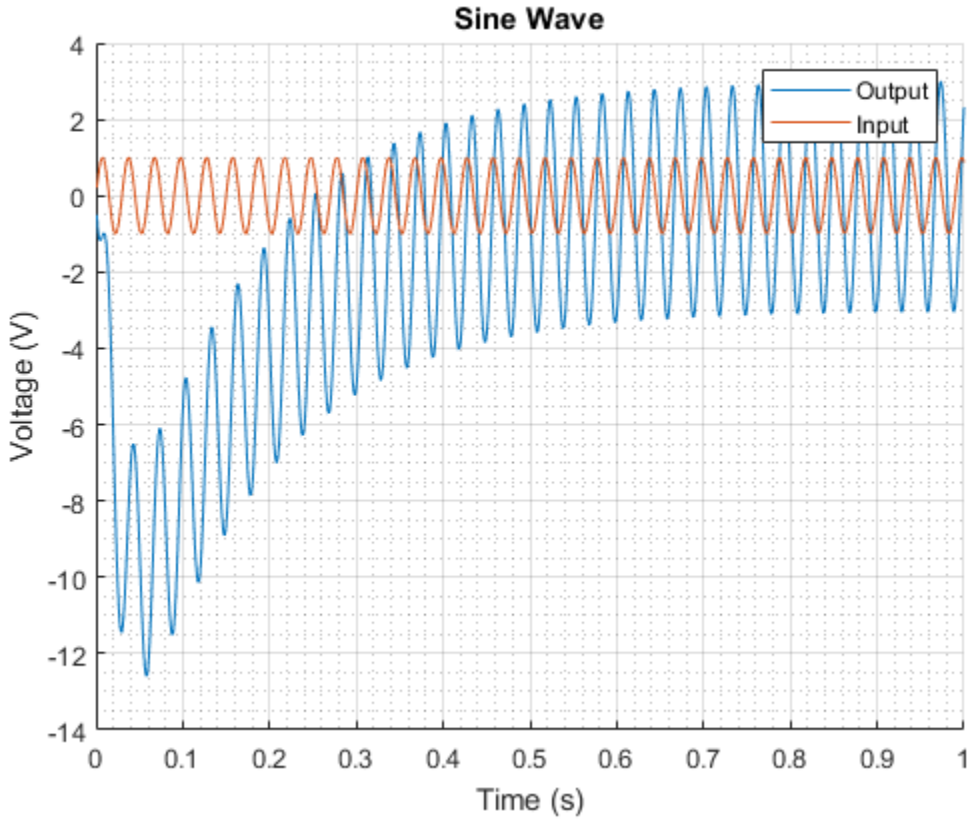
```
ylabel('Voltage (V)');

Y = fft(out(:,3),n);
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
P = abs(Y/n);
semilogy(fshift, powershift) ;
legend('Output', 'Input');
hold off;



clear out;
deltaT = 0.001;
A = (C/deltaT + G);
F = [0; 0; 0; 0; 0; 0; 0;];
Vold = [1; 0; 0; 0; 0; 0; 0;];
counter = 1;
for i = 0.001:deltaT:L*0.001
    F(1) = sin(2*pi*i/(0.03));
    vNew = inv(A)*(C*Vold./deltaT + F);
    Vold = vNew;
    out(counter , 1) = i;
    out(counter , 2) = vNew(7);
    out(counter , 3) = vNew(1);
    counter = counter + 1;
end
figure;
hold on;
grid on;
grid minor;
title('Sine Wave');
xlabel('Time (s)');
ylabel('Voltage (V)');
plot(out(:,1), out(:,2));
plot(out(:,1), out(:,3));
legend('Output', 'Input');
hold off;

figure;

Fs = 1000;

n = 2^nextpow2(L);
Y = fft(out(:,2),n);
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
P = abs(Y/n);
semilogy(fshift, powershift) ;
hold on;
```

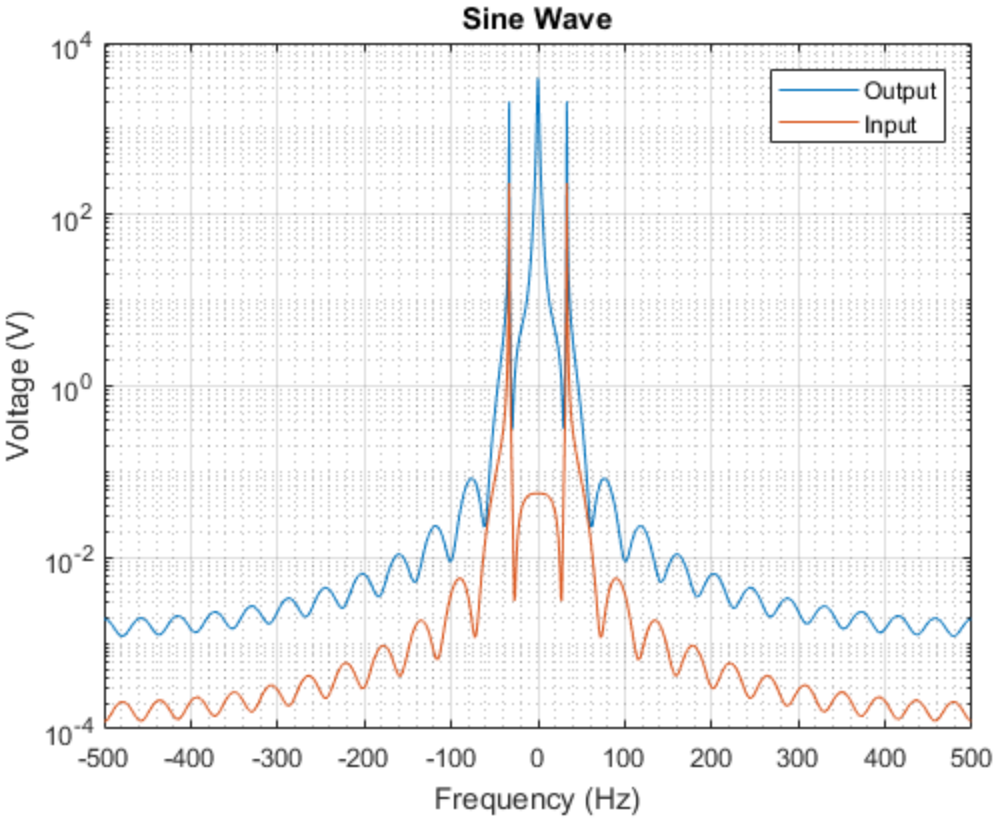
```
grid on;
grid minor;
title('Sine Wave');
xlabel('Frequency (Hz)');
ylabel('Voltage (V)');
Y = fft(out(:,3),n);
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
P = abs(Y/n);
semilogy(fshift, powershift) ;
legend('Output', 'Input');
hold off;


clear out;
deltaT = 0.001;
A = (C/deltaT + G);
F = [0; 0; 0; 0; 0; 0; 0;];
Vold = [1; 0; 0; 0; 0; 0; 0;];
counter = 1;
for i = 0.001:deltaT:L*0.001
    F(1) = normpdf(i,0.06,0.03)/13.2981;
    vNew = inv(A)*(C*Vold./deltaT + F);
    Vold = vNew;
    out(counter , 1) = i;
    out(counter , 2) = vNew(7);
    out(counter , 3) = vNew(1);
    counter = counter + 1;
end
figure;
hold on;
grid on;
grid minor;
title('Gaussian Pulse');
xlabel('Time (s)');
ylabel('Voltage (V)');
plot(out(:,1), out(:,2));
plot(out(:,1), out(:,3));
legend('Output', 'Input');
hold off;


figure;

Fs = 1000;

n = 2^nextpow2(L);
Y = fft(out(:,2),n);
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
```

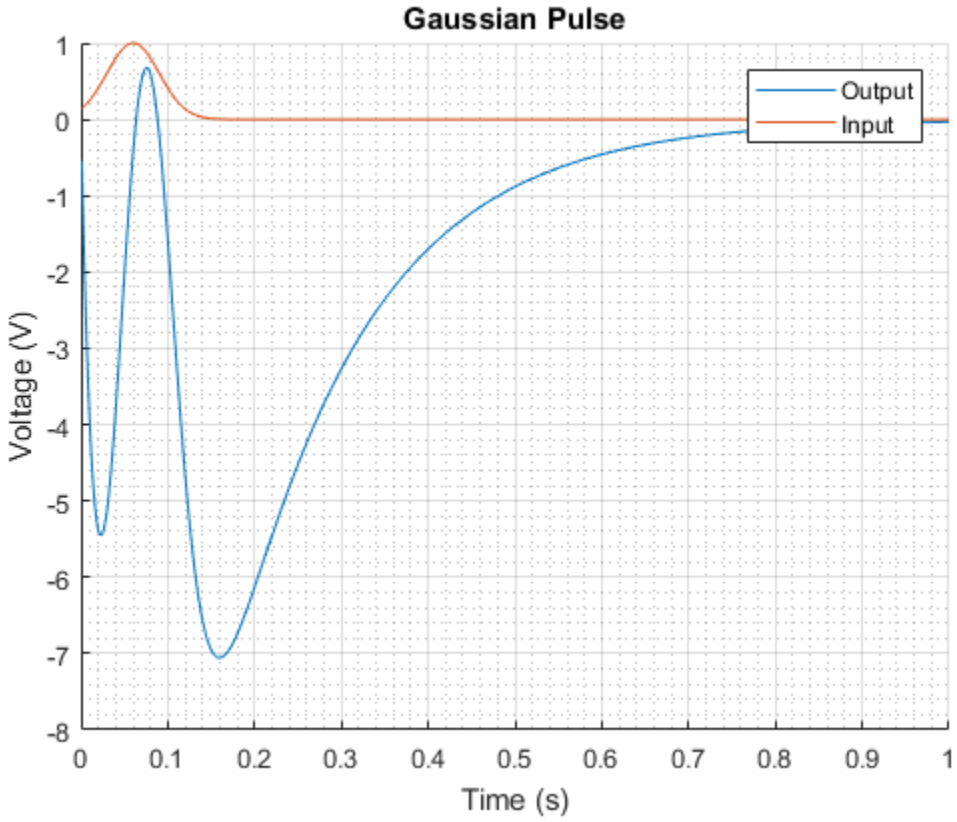
```
P = abs(Y/n);
semilogy(fshift, powershift) ;
hold on;
grid on;
grid minor;
title('Gaussian Pulse');
xlabel('Frequency (Hz)');
ylabel('Voltage (V)');
Y = fft(out(:,3),n);
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
P = abs(Y/n);
semilogy(fshift, powershift) ;
legend('Output', 'Input');
hold off;
```

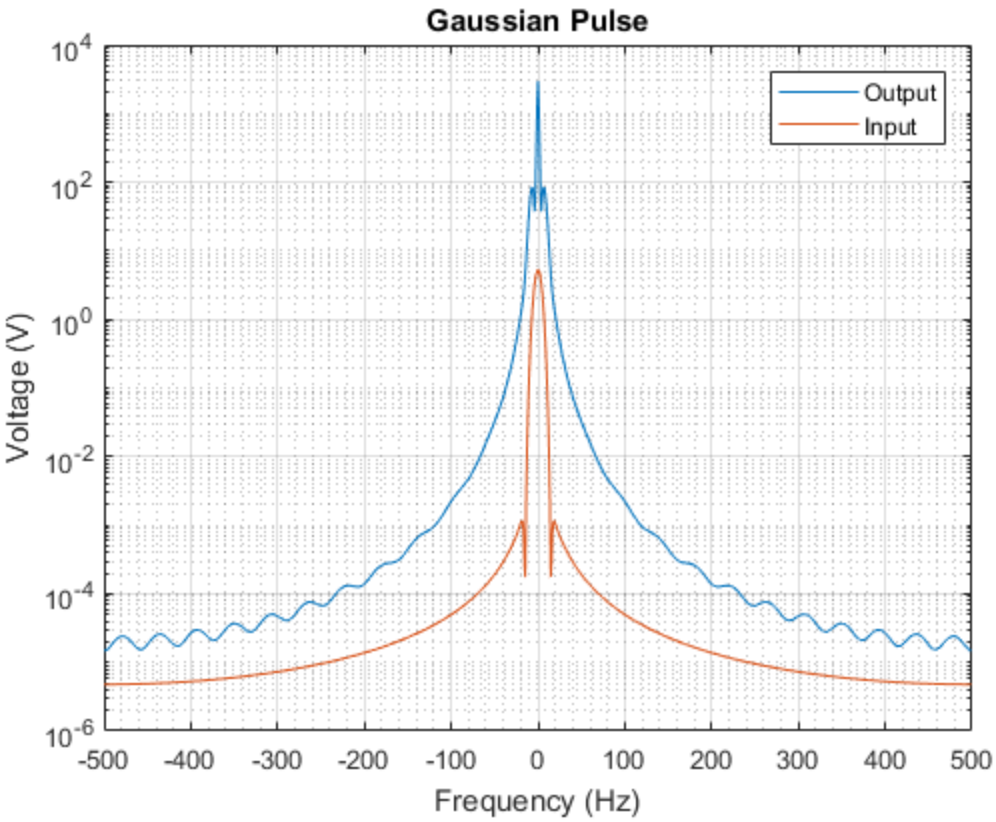



Step Function
Output
Input
Voltage (V)
Frequency (Hz)
10^4
10^2
10^0
10^-2
10^-4
10^-6
10^-8
-500
-400
-300
-200
-100
0
100
200
300
400
500


Sine Wave
Output
Input
Voltage (V)
Time (s)
4
2
0
-2
-4
-6
-8
-10
-12
-14
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Sine Wave
Output
Input
Voltage (V)
Frequency (Hz)
10^4
10^2
10^0
10^-2
10^-4
-500
-400
-300
-200
-100
0
100
200
300
400
500


Gaussian Pulse
Output
Input
Voltage (V)
Time (s)
1
0
-1
-2
-3
-4
-5
-6
-7
-8
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1



Question 2

```
clear all;
close all;


%Parameters
G1 = 1;
Cx = 0.25;
G2 = 1/2;
L = 0.2;
G3 = 1/10;
Alpha = 100;
G4 = 1/0.1;
G0 = 1/1000;
Cn = 0.00001;
```

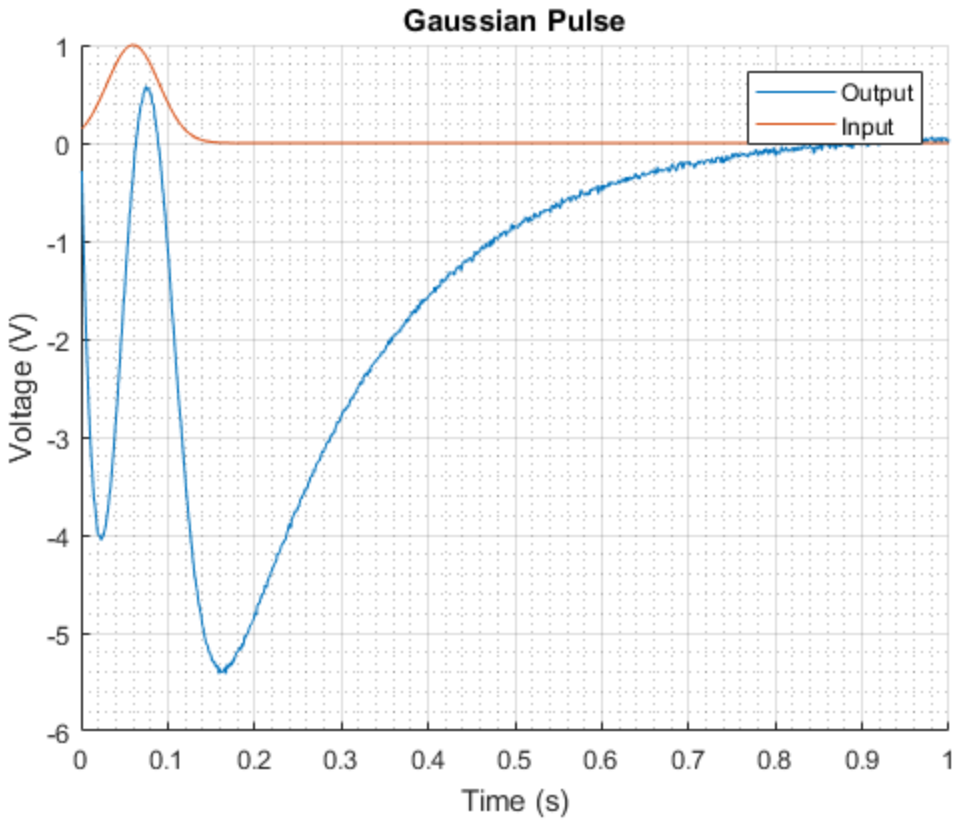
Part A) Updated C Matrix

```
C = [ 0 0 0 0 0 0 0 0;
    -Cx Cx 0 0 0 0 0 0;
    0 0 -L 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 -Cn 0 0 0 0;
    0 0 0 0 0 0 0 0;];


G = [ 1 0 0 0 0 0 0 0;
    -G2 G1+G2 -1 0 0 0 0 0;
    0 1 0 -1 0 0 0 0;
    0 0 -1 G3 0 0 0 0;
    0 0 0 0 -Alpha 1 0 0;
    0 0 0 G3 -1 0 0 1;
    0 0 0 0 0 -G4 G4+G0 0;
    0 0 0 0 0 0 0 1;];

F = [0; 0; 0; 0; 0; 0; 0; 0;];

deltaT = 0.001;
A = (C/deltaT + G);
Vold = [1; 0; 0; 0; 0; 0; 0; 0;];
counter = 1;
L =1000;
for i = 0.001:deltaT:L*0.001
    F(1) = normpdf(i,0.06,0.03)/13.2981;
   F(8) = normrnd(0.001, 0.0002);
    vNew = inv(A)*(C*Vold./deltaT + F);
    Vold = vNew;
    out(counter , 1) = i;
    out(counter , 2) = vNew(7);
    out(counter , 3) = vNew(1);
    counter = counter + 1;
end
```

Part B) Vout with Noise

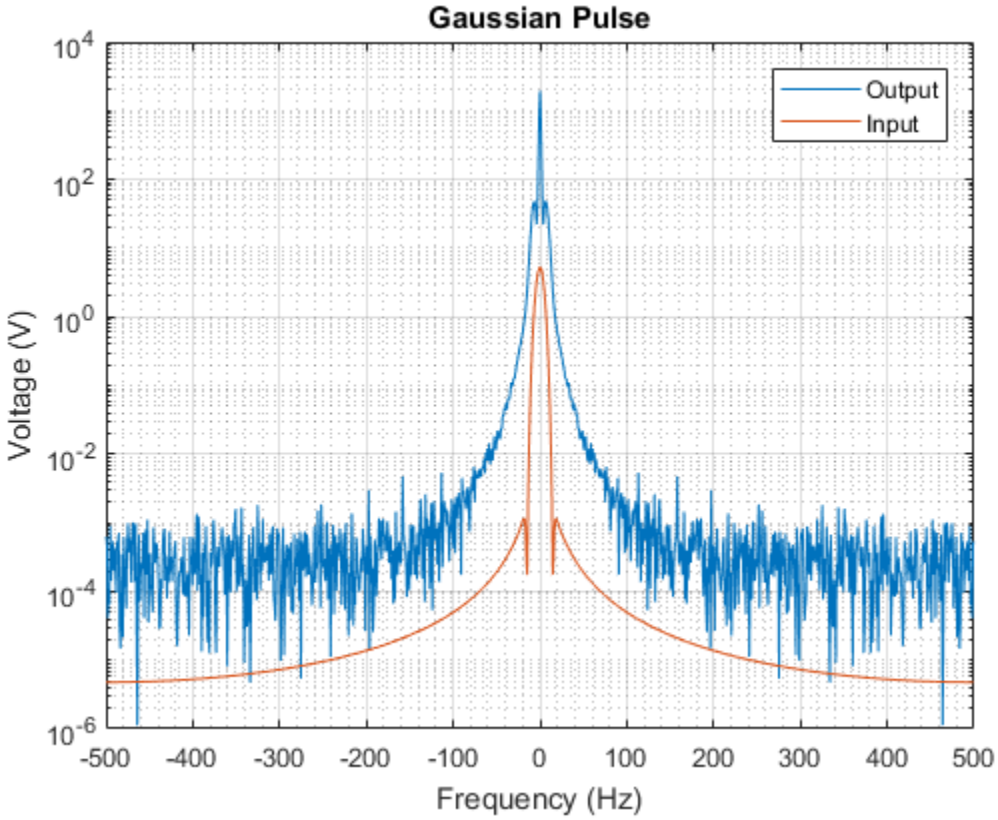
```
figure;
hold on;
grid on;
grid minor;
title('Gaussian Pulse');
xlabel('Time (s)');
ylabel('Voltage (V)');
plot(out(:,1), out(:,2));
plot(out(:,1), out(:,3));
legend('Output', 'Input');
hold off;
```



Part C and D) Fourier Transform with Noise

```
figure;

Fs = 1000;

n = 2^nextpow2(L);
Y = fft(out(:,2),n);
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
P = abs(Y/n);
semilogy(fshift, powershift) ;
hold on;
grid on;
grid minor;
title('Gaussian Pulse');
xlabel('Frequency (Hz)');
ylabel('Voltage (V)');
Y = fft(out(:,3),n);
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
```

```
P = abs(Y/n);
semilogy(fshift, powershift) ;
legend('Output', 'Input');
hold off;
```



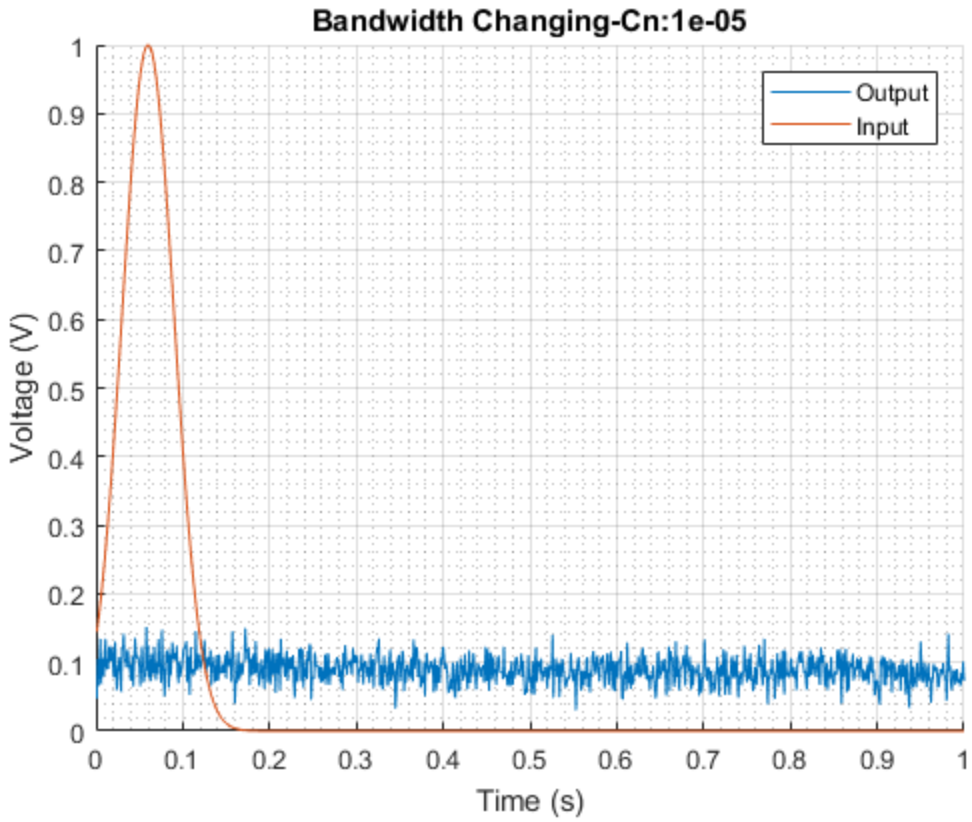
Part E) 3 Plots with Different Couts

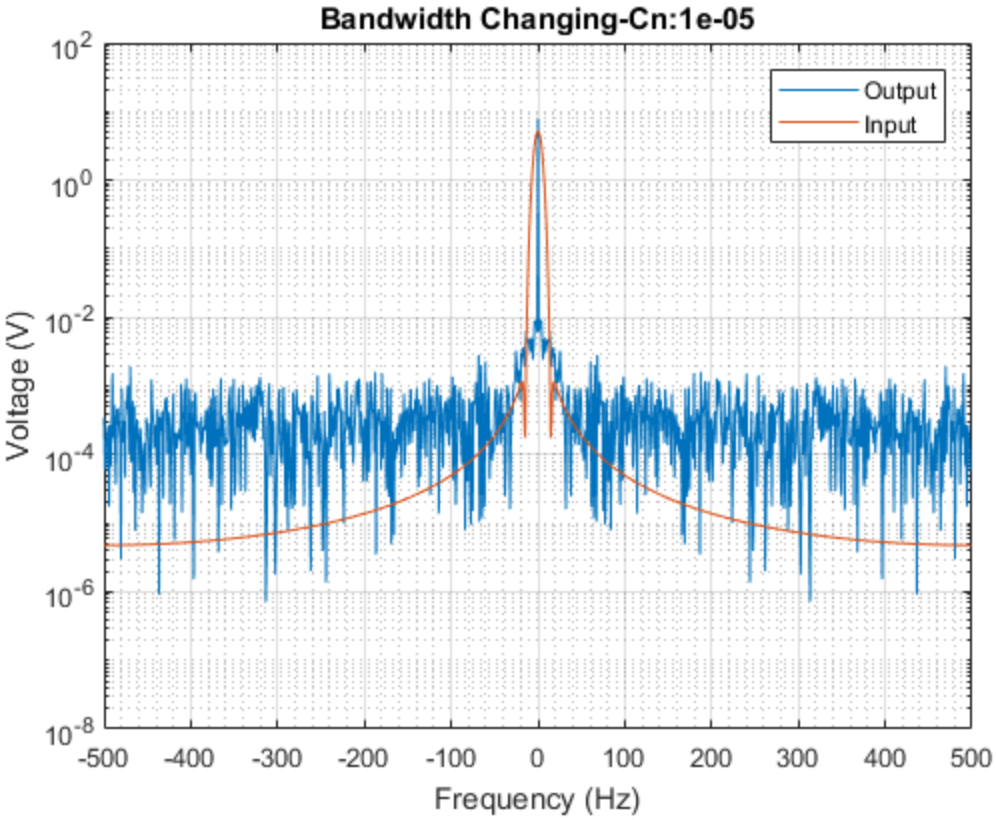
With a higher Cn the bandwidth becomes more broad

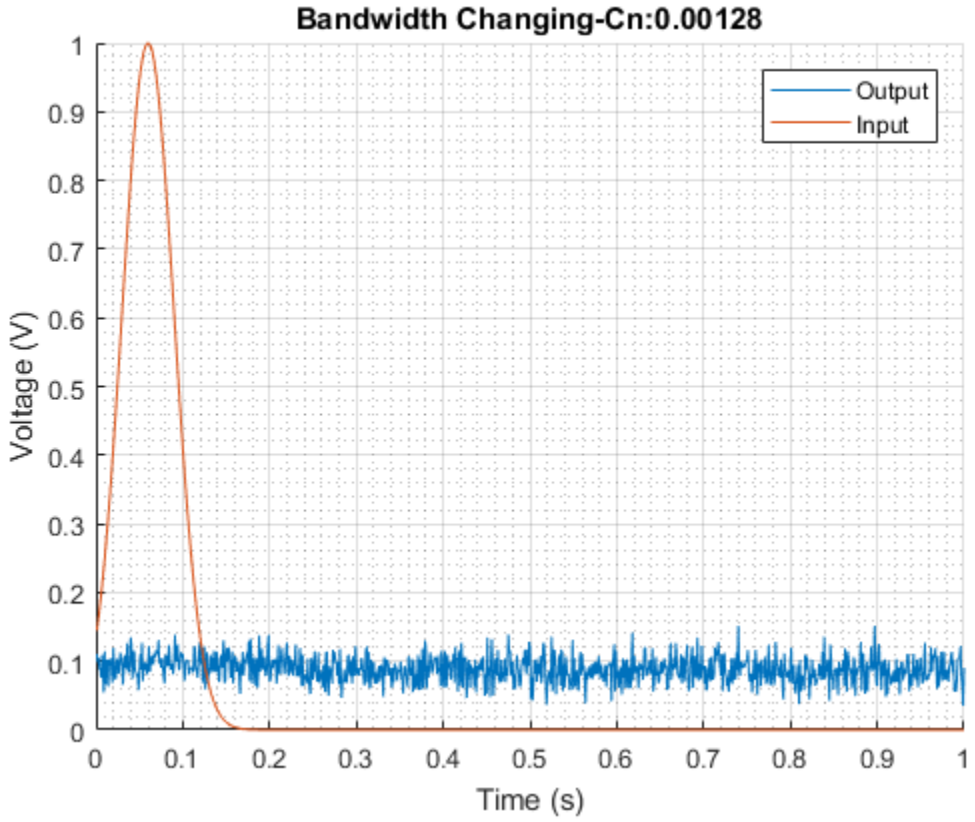
```
for zz = 1:3
Cn = 0.00001*zz^7;


C = [ 0 0 0 0 0 0 0 0;
    -Cx Cx 0 0 0 0 0 0;
    0 0 -L 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 -Cn 0 0 0 0;
    0 0 0 0 0 0 0 0;];


G = [ 1 0 0 0 0 0 0 0;
    -G2 G1+G2 -1 0 0 0 0 0;
    0 1 0 -1 0 0 0 0;
```

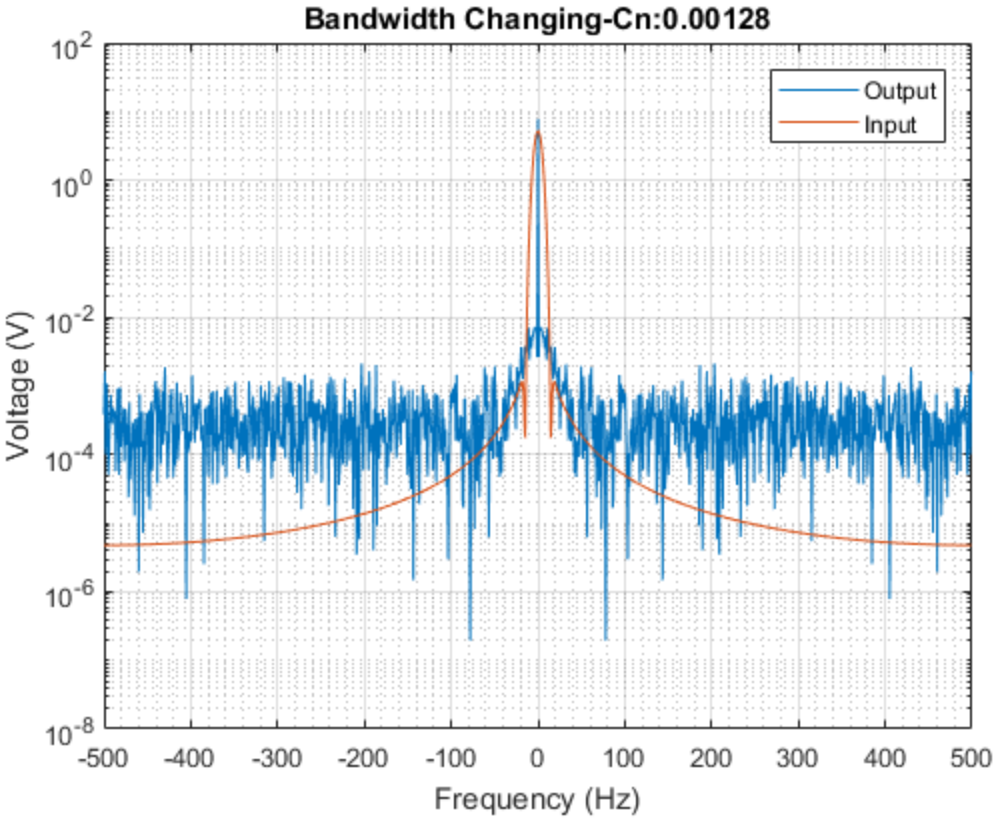
```
    0 0 -1 G3 0 0 0 0;
    0 0 0 0 -Alpha 1 0 0;
    0 0 0 G3 -1 0 0 1;
    0 0 0 0 0 -G4 G4+G0 0;
    0 0 0 0 0 0 0 1;];


deltaT = 0.001;
A = (C/deltaT + G);
F = [0; 0; 0; 0; 0; 0; 0; 0;];
Vold = [1; 0; 0; 0; 0; 0; 0; 0;];
counter = 1;
for i = 0.001:deltaT:L*0.001
    F(1) = normpdf(i,0.06,0.03)/13.2981;
   F(8) = normrnd(0.001, 0.0002);
    vNew = inv(A)*(C*Vold./deltaT + F);
    Vold = vNew;
    out(counter , 1) = i;
    out(counter , 2) = vNew(7);
    out(counter , 3) = vNew(1);
    counter = counter + 1;
end
figure;
hold on;
grid on;
grid minor;
title(strcat('Bandwidth Changing-Cn:', num2str(Cn)));
xlabel('Time (s)');
ylabel('Voltage (V)');
plot(out(:,1), out(:,2));
plot(out(:,1), out(:,3));
legend('Output', 'Input');
hold off;


figure;

Fs = 1000;

n = 2^nextpow2(L);
Y = fft(out(:,2),n);
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
P = abs(Y/n);
semilogy(fshift, powershift) ;
hold on;
grid on;
grid minor;
title(strcat('Bandwidth Changing-Cn:', num2str(Cn)));
xlabel('Frequency (Hz)');
ylabel('Voltage (V)');
Y = fft(out(:,3),n);
```

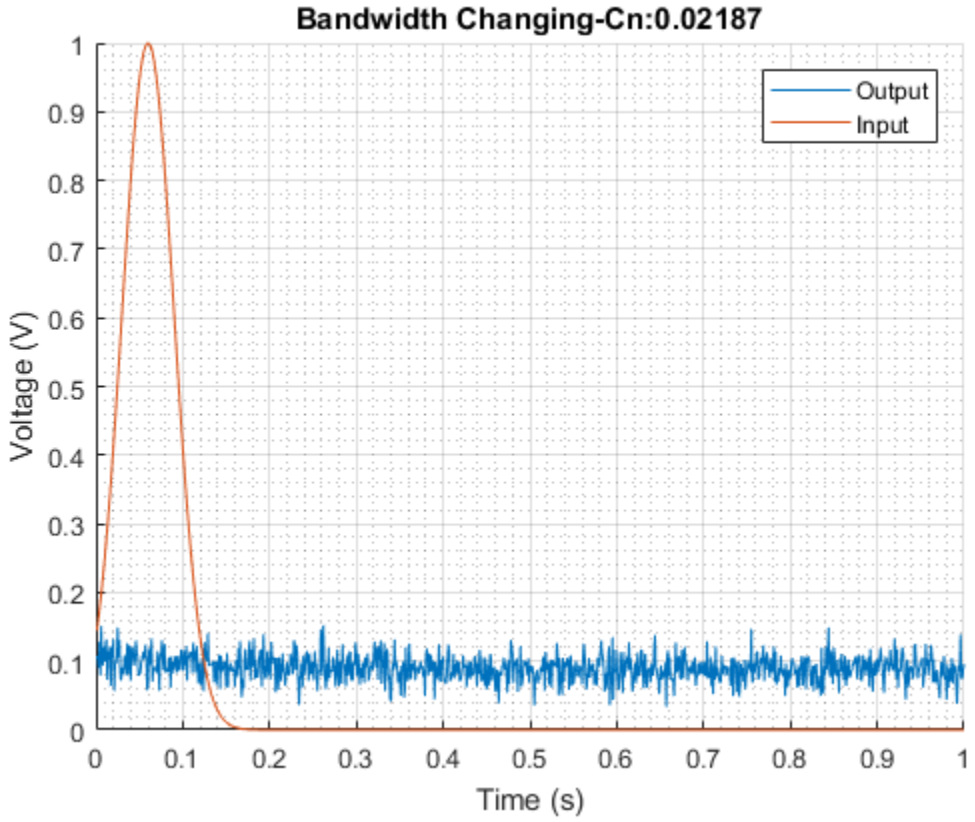
```
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
P = abs(Y/n);
semilogy(fshift, powershift) ;
legend('Output', 'Input');
hold off;


end
```

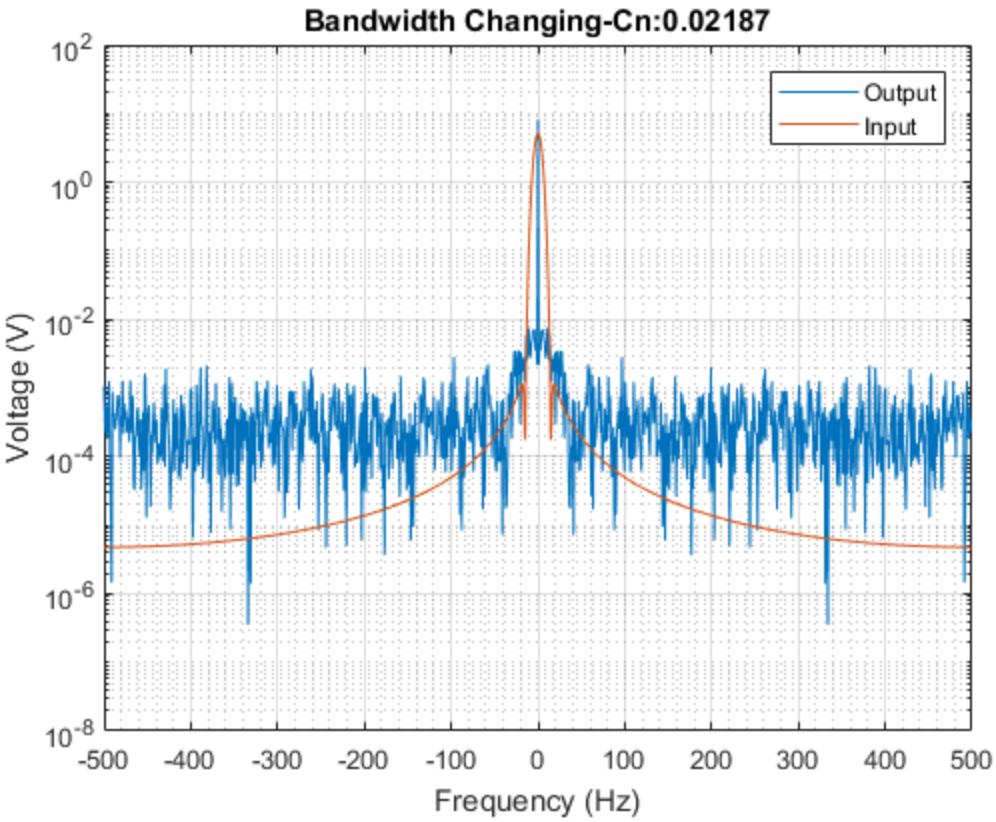



Bandwidth Changing-Cn:1e-05
Output
Input
Voltage (V)
Frequency (Hz)
10^2
10^0
10^-2
10^-4
10^-6
10^-8
-500
-400
-300
-200
-100
0
100
200
300
400
500


Bandwidth Changing-Cn:0.00128
Output
Input
Voltage (V)
Time (s)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1


Bandwidth Changing-Cn:0.00128
Output
Input
Voltage (V)
Frequency (Hz)
10^2
10^0
10^-2
10^-4
10^-6
10^-8
-500
-400
-300
-200
-100
0
100
200
300
400
500


Bandwidth Changing-Cn:0.02187
Output
Input
Voltage (V)
Time (s)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1



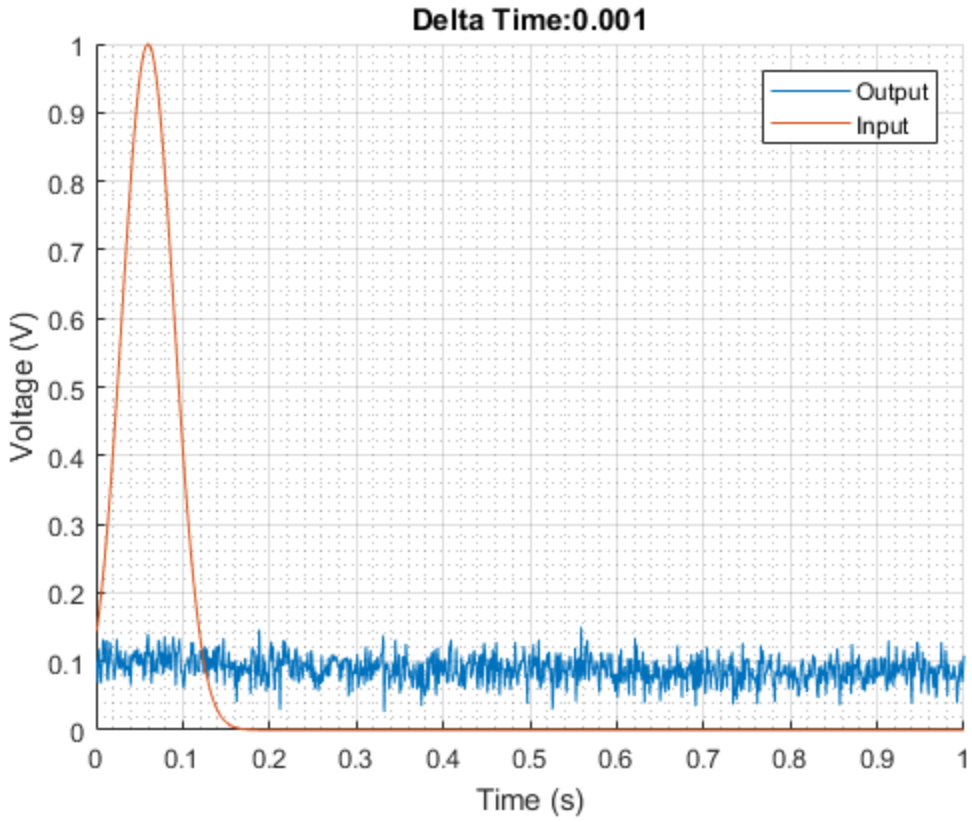
Part F) 2 Plots with different time steps

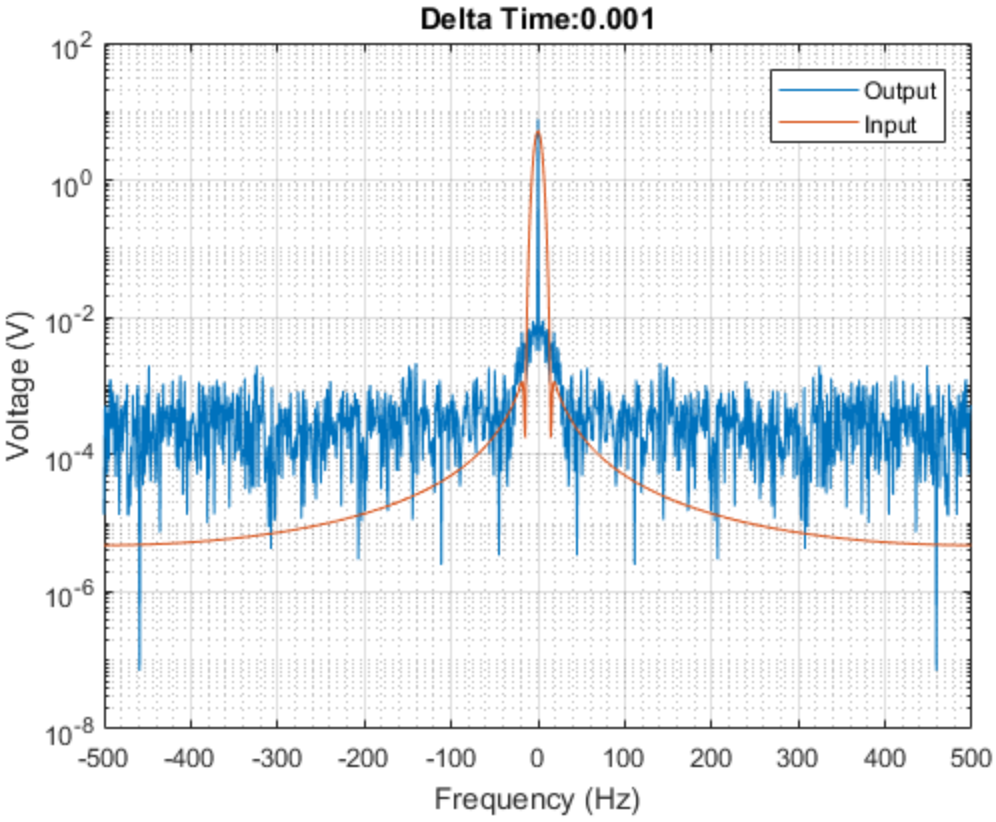
Can see that with larger time steps the accuracy of the plot suffers and less data points are taken not giving good insight into the true signal that is occuring

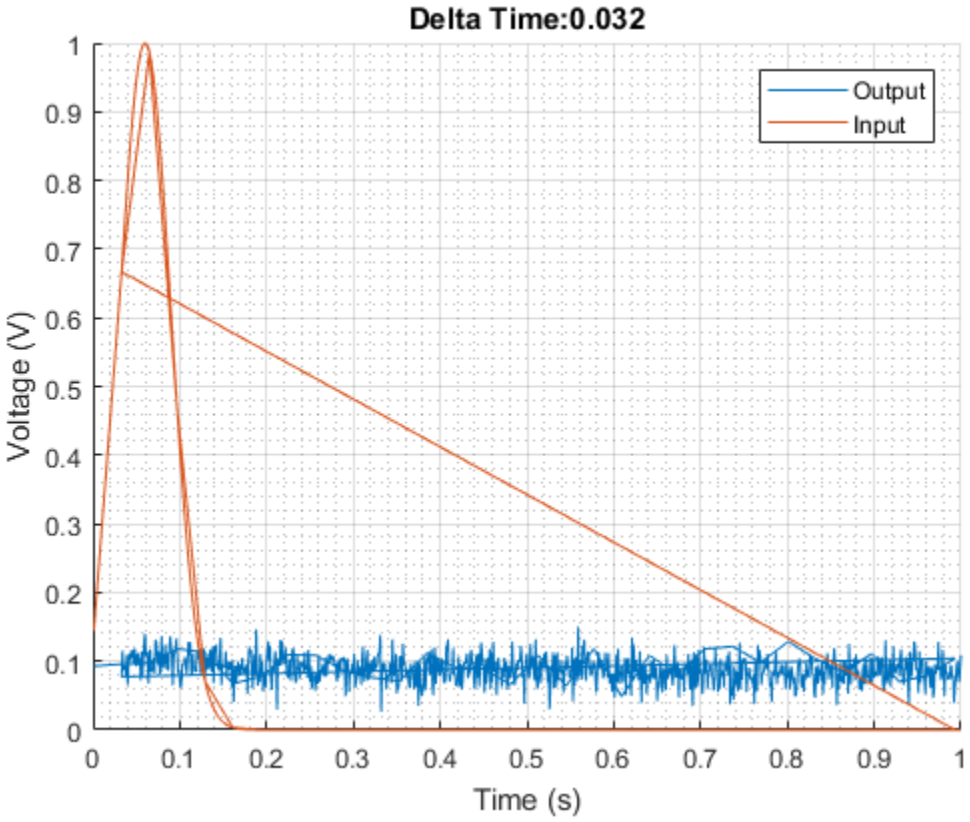
```
Cn = 0.00001;


C = [ 0 0 0 0 0 0 0 0;
    -Cx Cx 0 0 0 0 0 0;
    0 0 -L 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 -Cn 0 0 0 0;
    0 0 0 0 0 0 0 0;];


G = [ 1 0 0 0 0 0 0 0;
    -G2 G1+G2 -1 0 0 0 0 0;
    0 1 0 -1 0 0 0 0;
    0 0 -1 G3 0 0 0 0;
    0 0 0 0 -Alpha 1 0 0;
    0 0 0 G3 -1 0 0 1;
    0 0 0 0 0 -G4 G4+G0 0;
    0 0 0 0 0 0 0 1;];
```

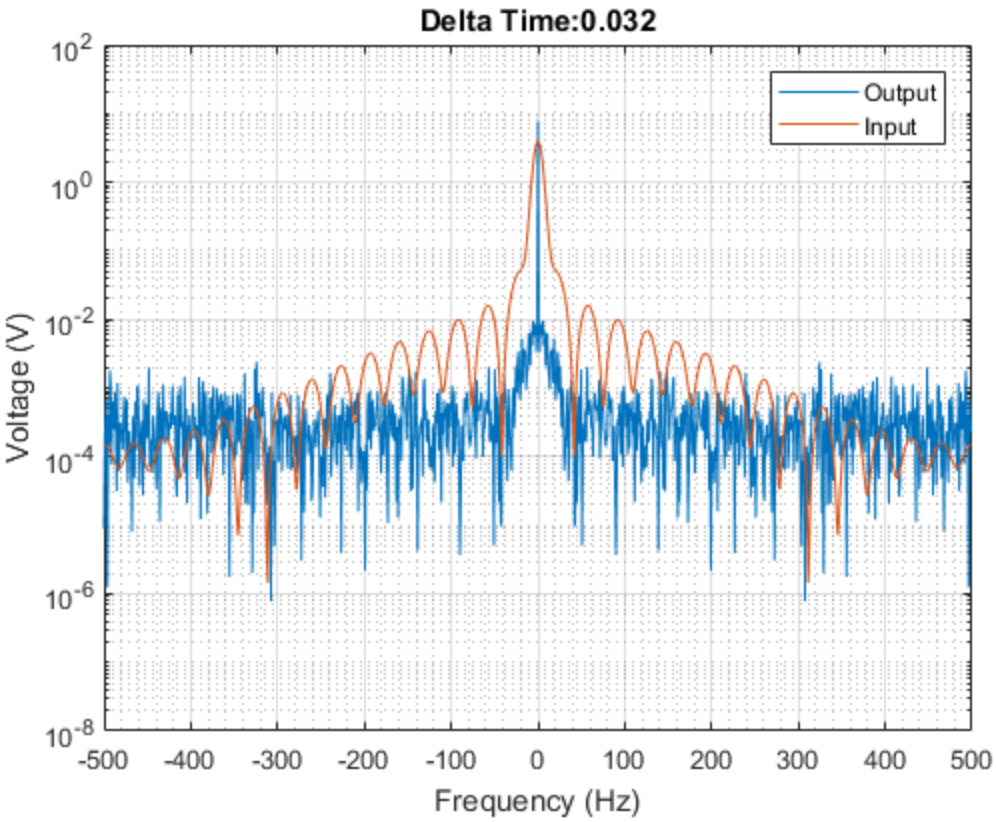
```
for zz = 1:2

deltaT = 0.001*zz^5;
A = (C/deltaT + G);
F = [0; 0; 0; 0; 0; 0; 0; 0;];
Vold = [1; 0; 0; 0; 0; 0; 0; 0;];
counter = 1;
for i = 0.001:deltaT:L*0.001
    F(1) = normpdf(i,0.06,0.03)/13.2981;
    F(8) = normrnd(0.001, 0.0002);
    vNew = inv(A)*(C*Vold./deltaT + F);
    Vold = vNew;
    out(counter , 1) = i;
    out(counter , 2) = vNew(7);
    out(counter , 3) = vNew(1);
    counter = counter + 1;
end
figure;
hold on;
grid on;
grid minor;
title(strcat('Delta Time:', num2str(deltaT)));
xlabel('Time (s)');
ylabel('Voltage (V)');
plot(out(:,1), out(:,2));
plot(out(:,1), out(:,3));
legend('Output', 'Input');
hold off;


figure;

Fs = 1000;

n = 2^nextpow2(L);
Y = fft(out(:,2),n);
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
P = abs(Y/n);
semilogy(fshift, powershift) ;
hold on;
grid on;
grid minor;
title(strcat('Delta Time:', num2str(deltaT)));
xlabel('Frequency (Hz)');
ylabel('Voltage (V)');
Y = fft(out(:,3),n);
```

```
Y = fftshift(Y);
fshift = (-n/2:n/2-1)*(Fs/n);
powershift = abs(Y).^2/n;
f = Fs*(0:(n/2))/n;
P = abs(Y/n);
semilogy(fshift, powershift) ;
legend('Output', 'Input');
hold off;
end
```




Delta Time:0.001
Output
Input
Voltage (V)
Frequency (Hz)
10^2
10^0
10^-2
10^-4
10^-6
10^-8
-500
-400
-300
-200
-100
0
100
200
300
400
500


Delta Time:0.032
Output
Input
Voltage (V)
Time (s)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1



Question 4

Part A) Change to Simulator

In order to account for nonlinear terms one would need to encorporate an additional matrix. In the notes this matrix is defined as the B matrix and is represented by the following equation.

$$ V^{j} = A^{-1}[C\frac{V^{j-1}{\delta t} + F(t^{j}) - B(V^{j})]

This could be implemented by changing the equation that relates to calculating the new V matrix when using the old V matrix. The B matrix would include the squared and cubed term by multiplying the V matrix into the B.

Published with MATLAB® R2017a