The same equations used in PA9 were the same equations used and can be seen below

$$
\begin{aligned}
V_1 &= V_{in} \\
G_1(V_2 - V_1) + C\frac{d(V_2 - V_1)}{dt} + G_2V_2 - I_L &= 0 \\
V_2 - V_3 - L\frac{dI_L}{dt} &= 0 \\
-I_L + G_3V_3 &= 0 \\
V_4 - \alpha I_3 &= 0 \\
G_3V_3 - I_3 &= 0 \\
G_4(V_O - V_4) + G_OV_O &= 0
\end{aligned}
$$

Matrices:

$$
\mathbf{C} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -C & C & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -L & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}
\quad
\mathbf{G} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -G_1 & G_1 + G_2 & -1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & -1 & G_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\alpha & 1 & 0 \\ 0 & 0 & 0 & G_3 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -G_4 & G_4 + G_O \end{bmatrix}
$$

$$
\mathbf{V} = \begin{bmatrix} V_1 \\ V_2 \\ I_L \\ V_3 \\ I_3 \\ V_4 \\ V_O \end{bmatrix}
\mathbf{F} = \begin{bmatrix} V_{in} \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
$$

Fig 1 Differential Equations

The DC Sweep Vin and Vout plots, along with the frequency spectrums can be seen in figures 1 to 3 below:

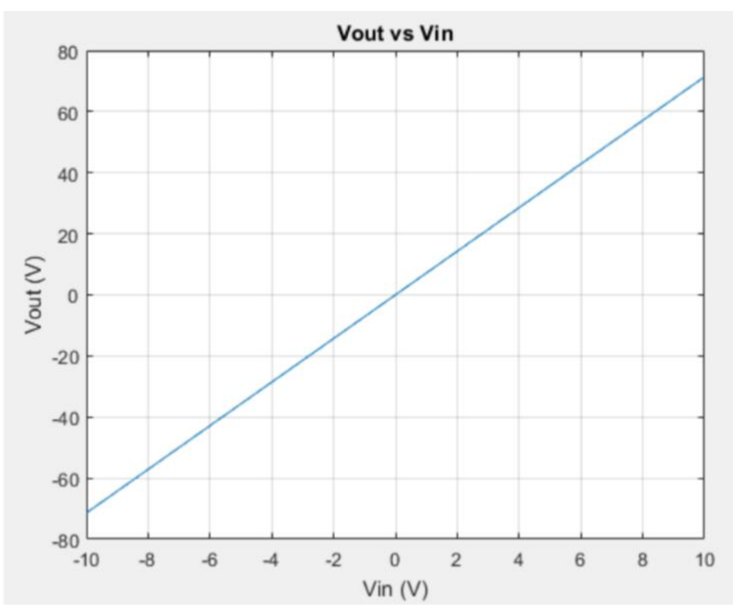

Fig 1 Vout vs Vin

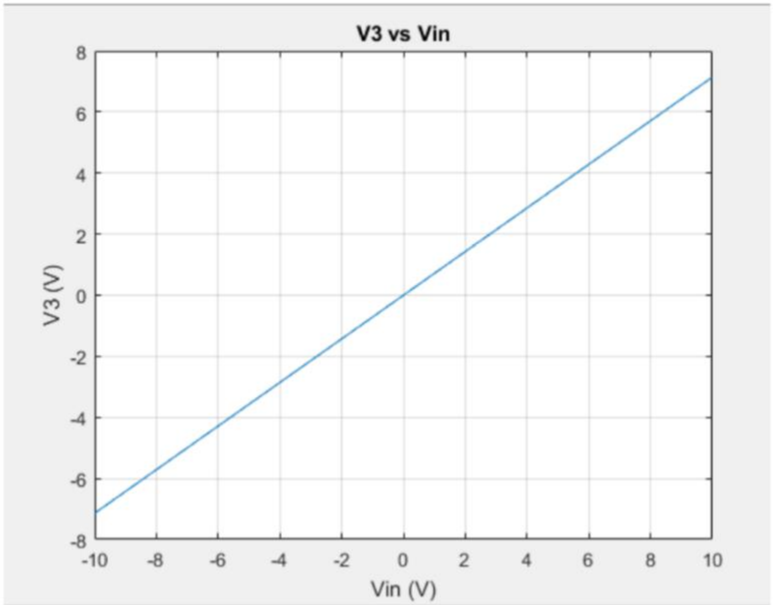

Fig 2 V3 vs Vin

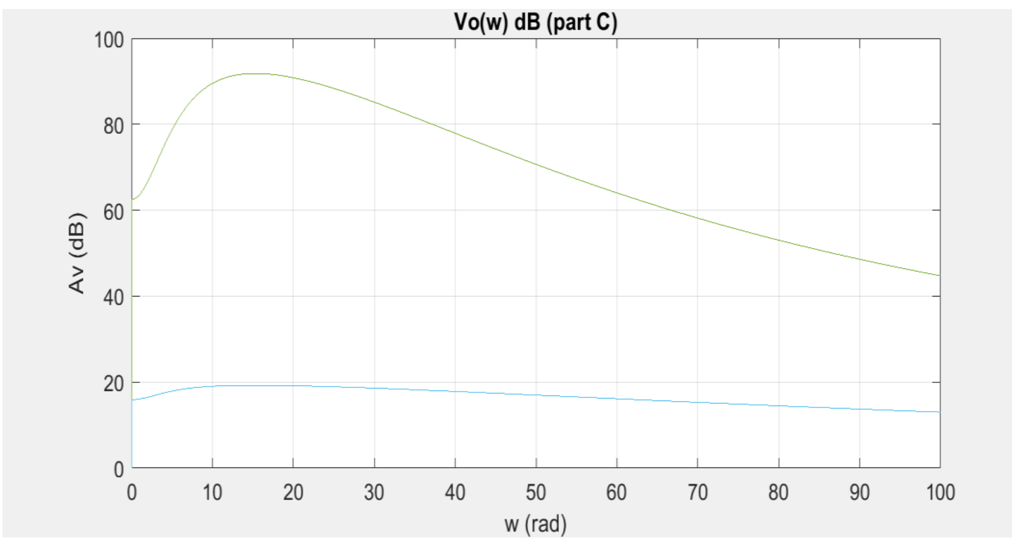

Fig 3 Ac plot of Vo as a function of w

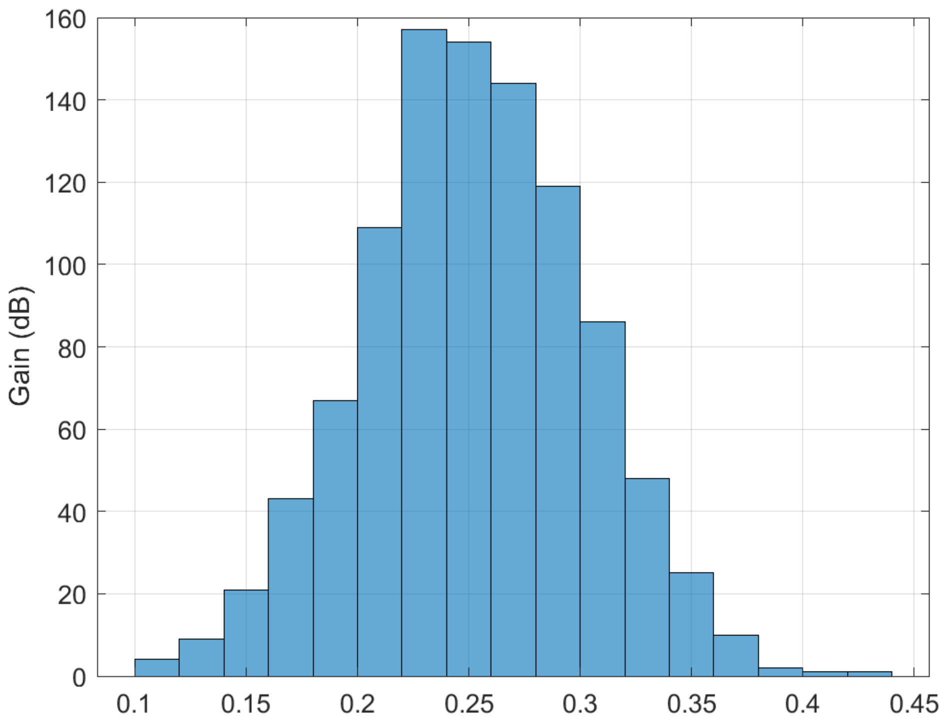

Fig 4

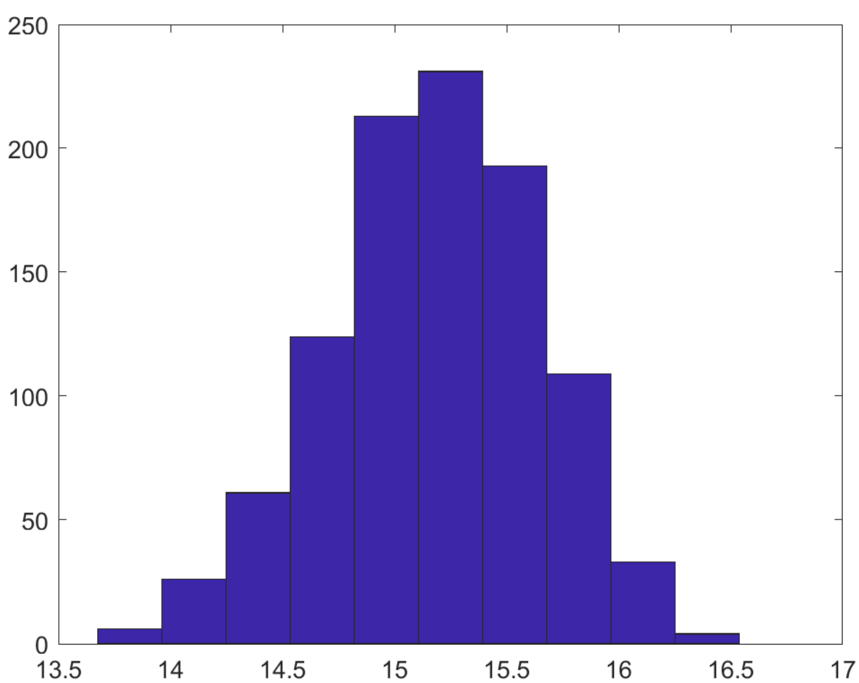

Fig 5 Histogram of Gain

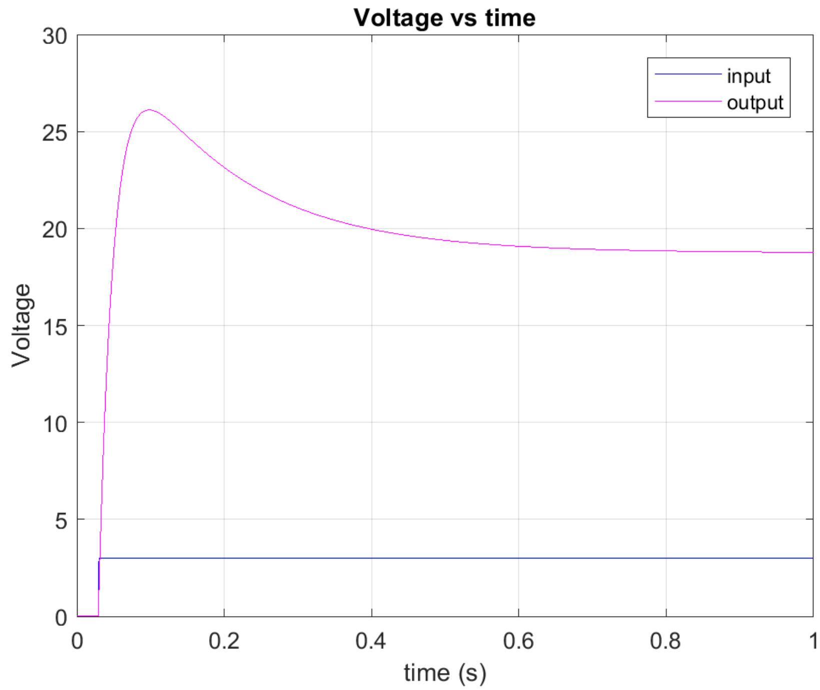

Fig 6 Plot of the Voltage vs Time

a.) By inspection the circuit is a low pass filter

b.) We should expect the frequency response to cut off high frequencies and allow only the low frequencies to go through.

d.) Part v: As the time step is increased the accuracy of the simulation is decreased.

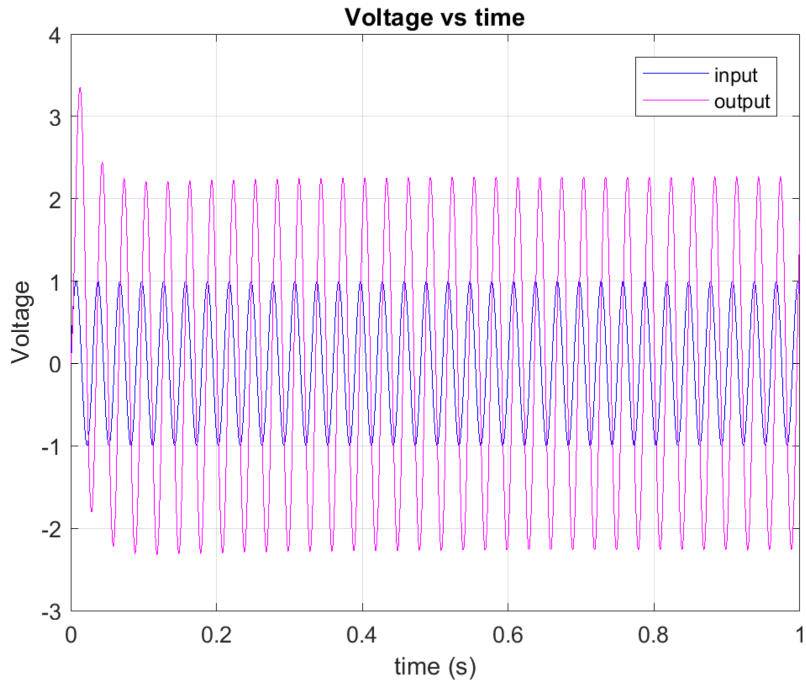

Fig 7 Voltage vs Time Input and Output siSnusoid

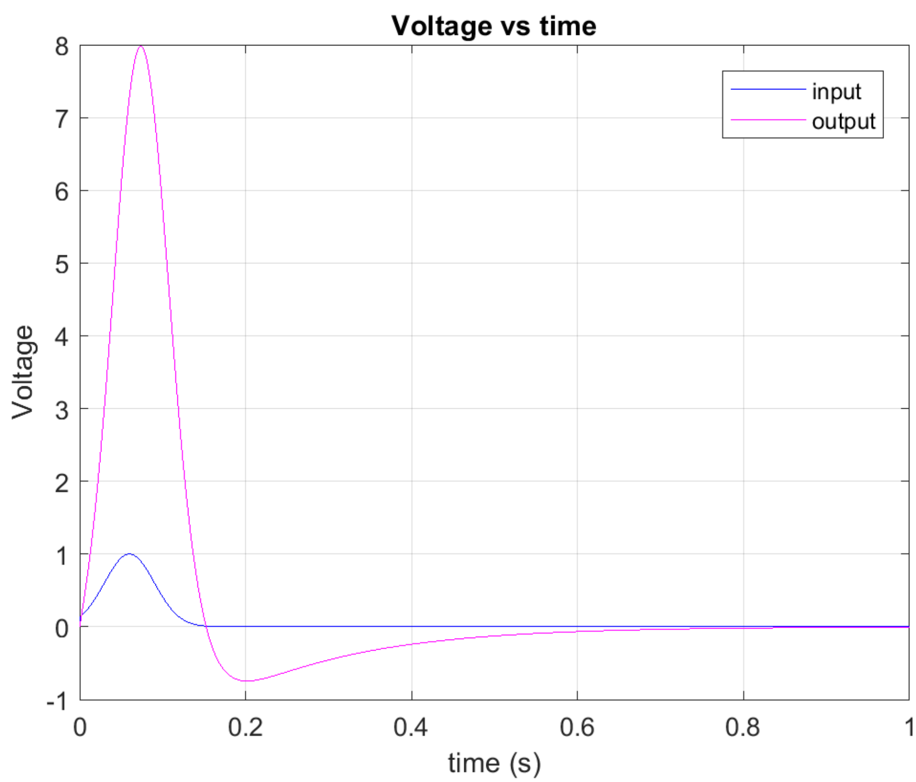

Fig 8 Voltage vs Time with Input and Output

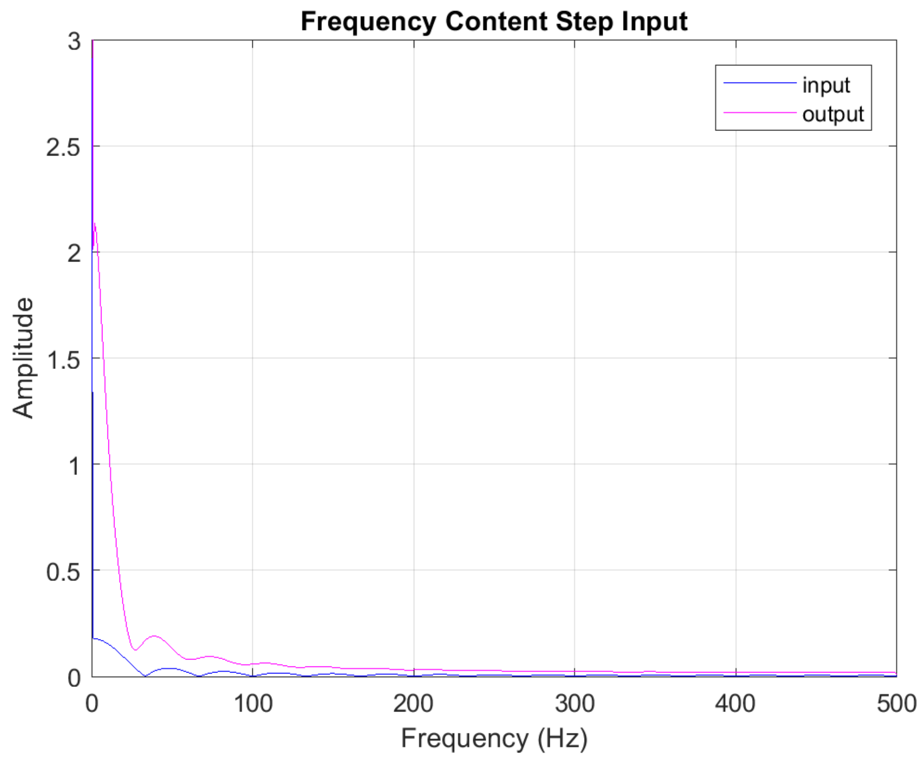

Fig 9 Frequency Content Step Input

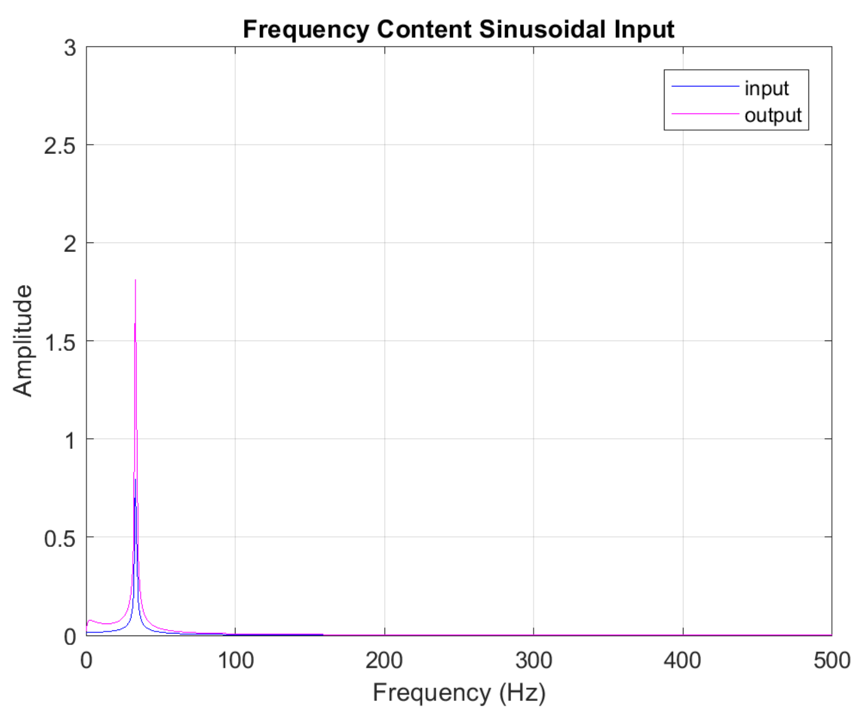

Fig 10 Frequency Content Sinusoidal Input

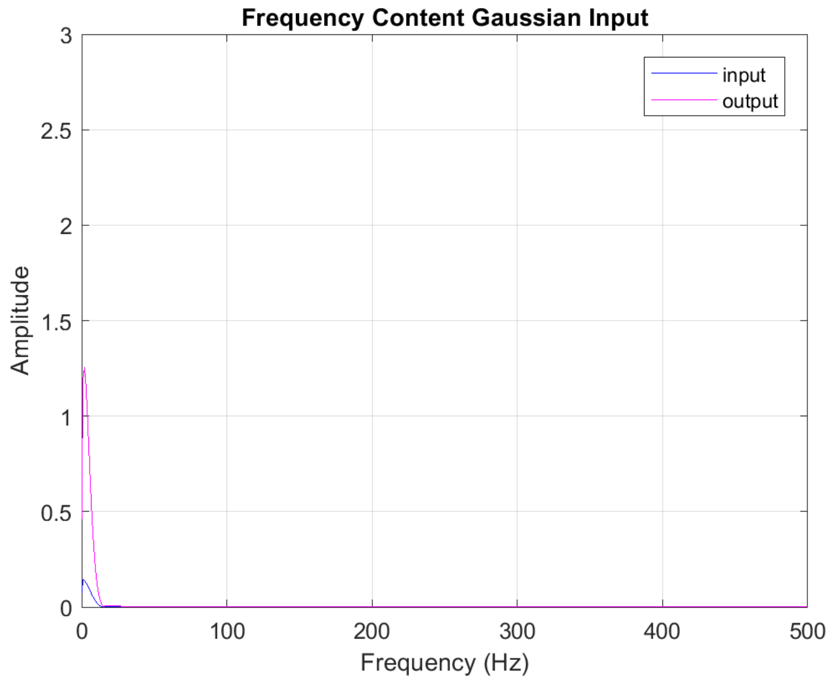

Fig 11 Frequency Content Gaussian Input

```
C1 =

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

C MATRIX

Varying Cn:

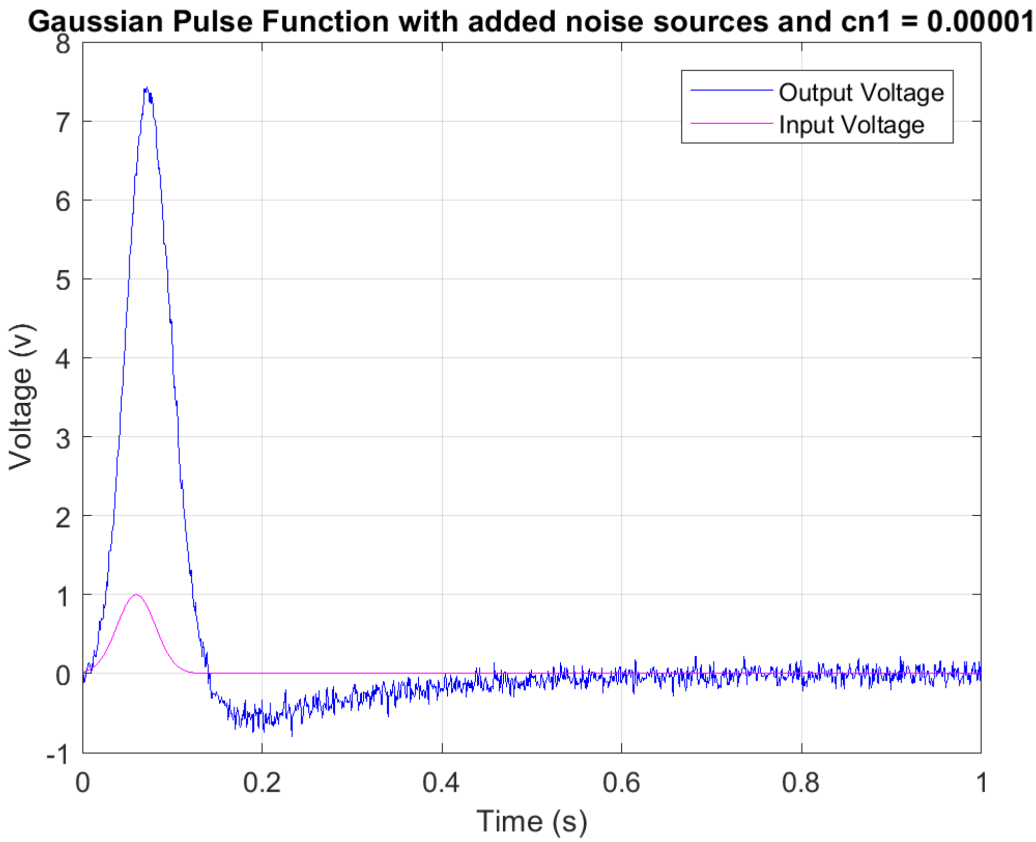

Fig 12 Gaussian Pulse Function with Added Noise Cn = 0.00001

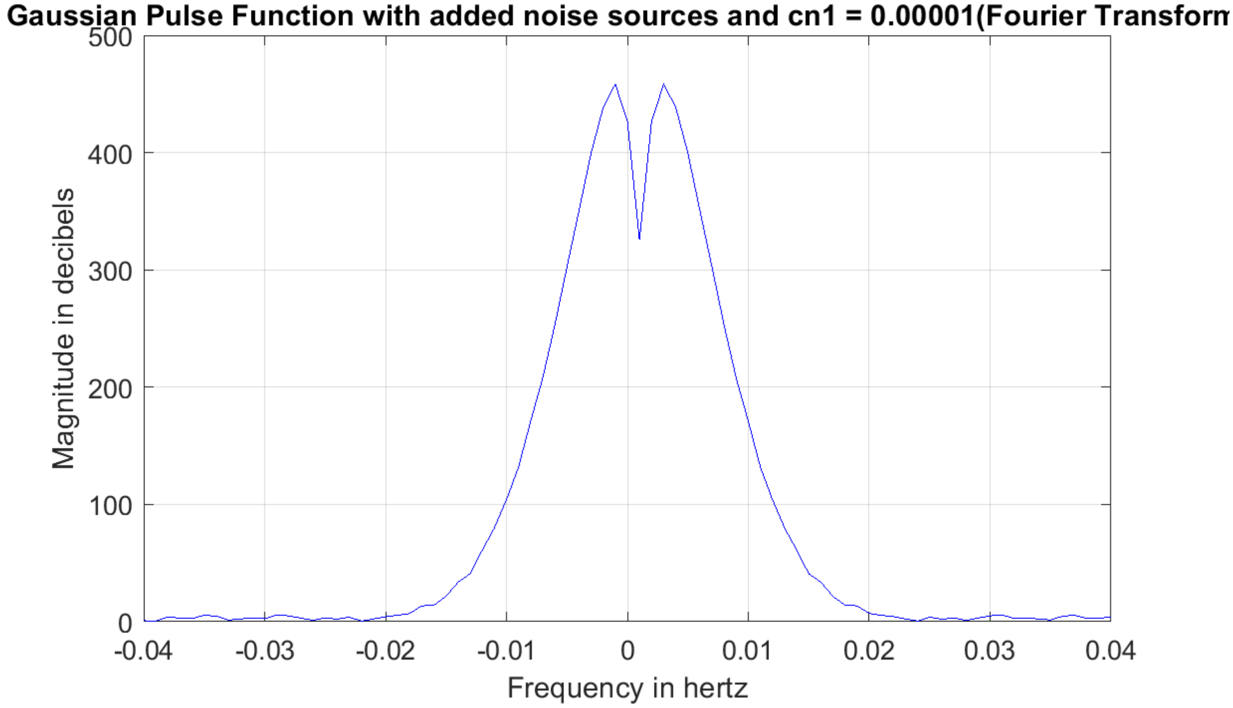

Fig 13 Gaussian Pulse function with added noise sources cn1=0.00001

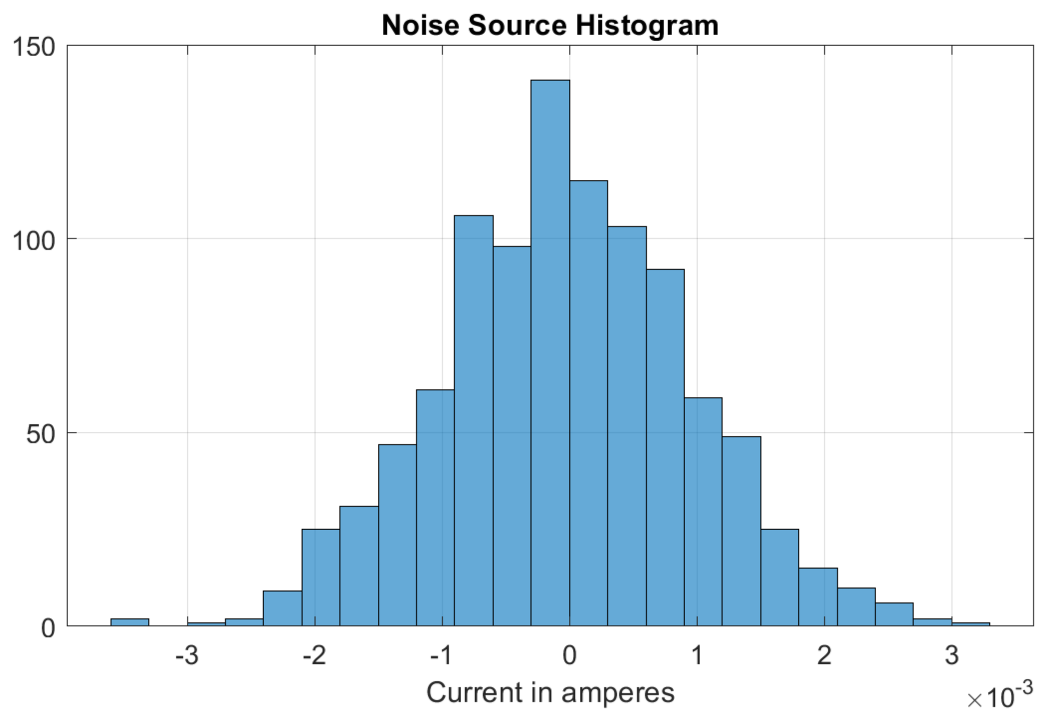

Fig 14 Noise Source Histogram

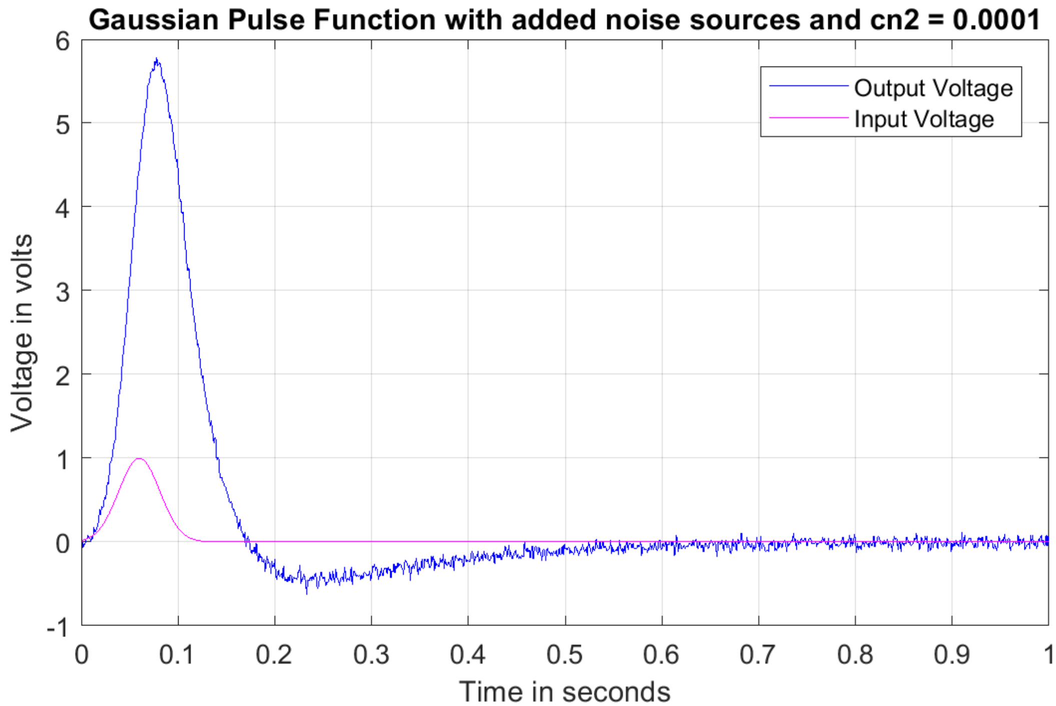

Fig 15

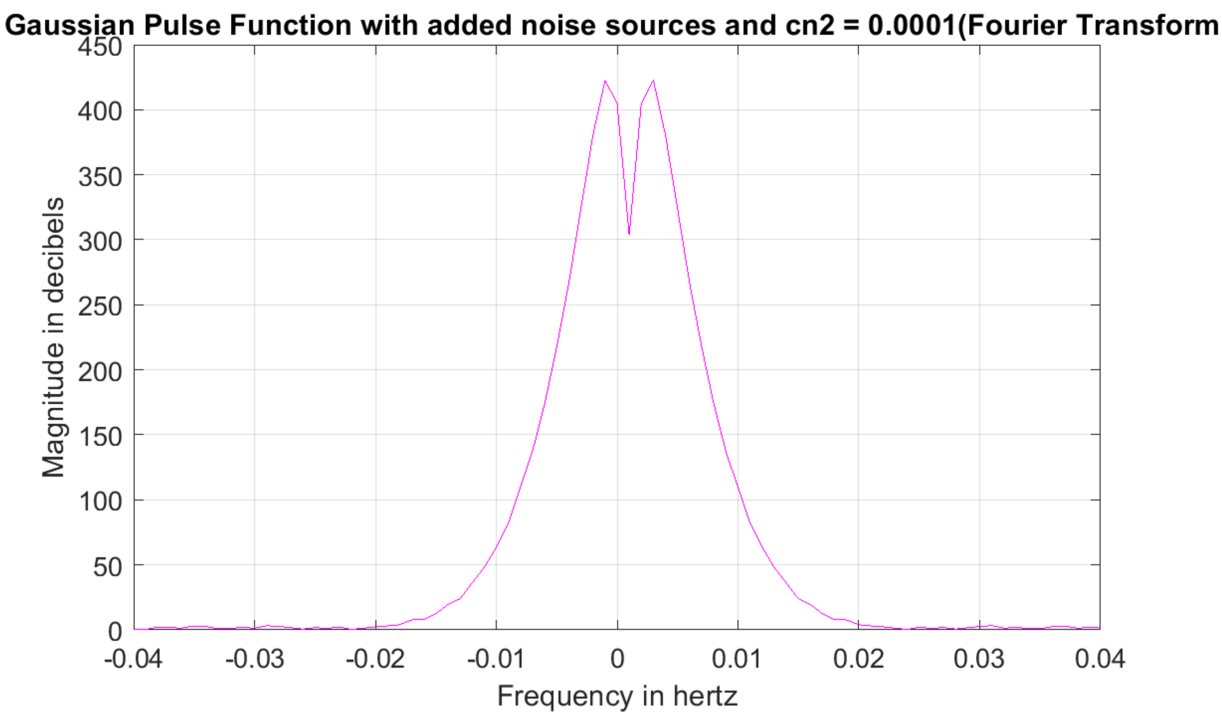

Fig 16 Gaussian Pulse function with added noise sources and cn2=0.0001

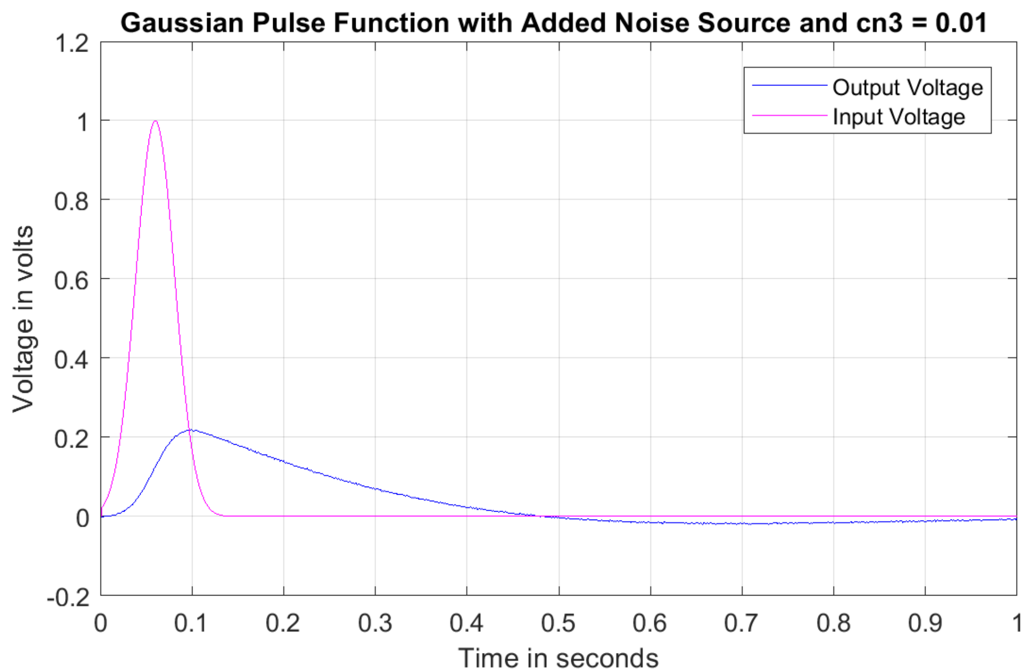

Fig 17 Gaussian Pulse function with added noise sources and cn3=0.01

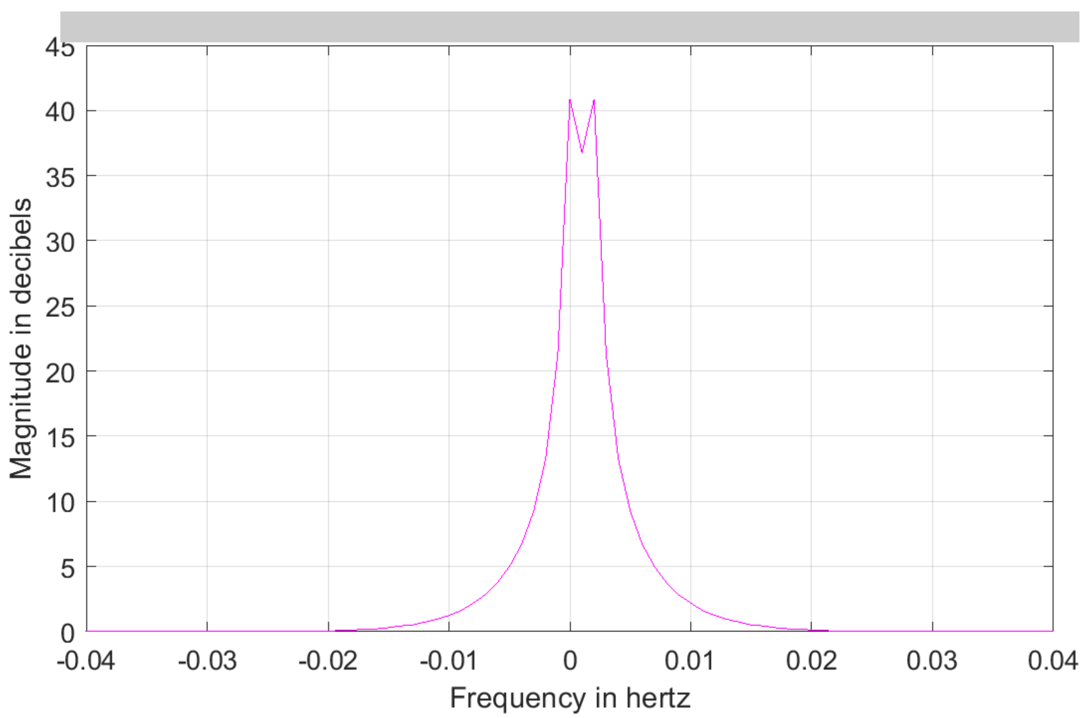

Fig 18 Gaussian Pulse function with added noise sources and cn3=0.01

In the end we can see that as cn gets larger the bandwidth gets smaller.

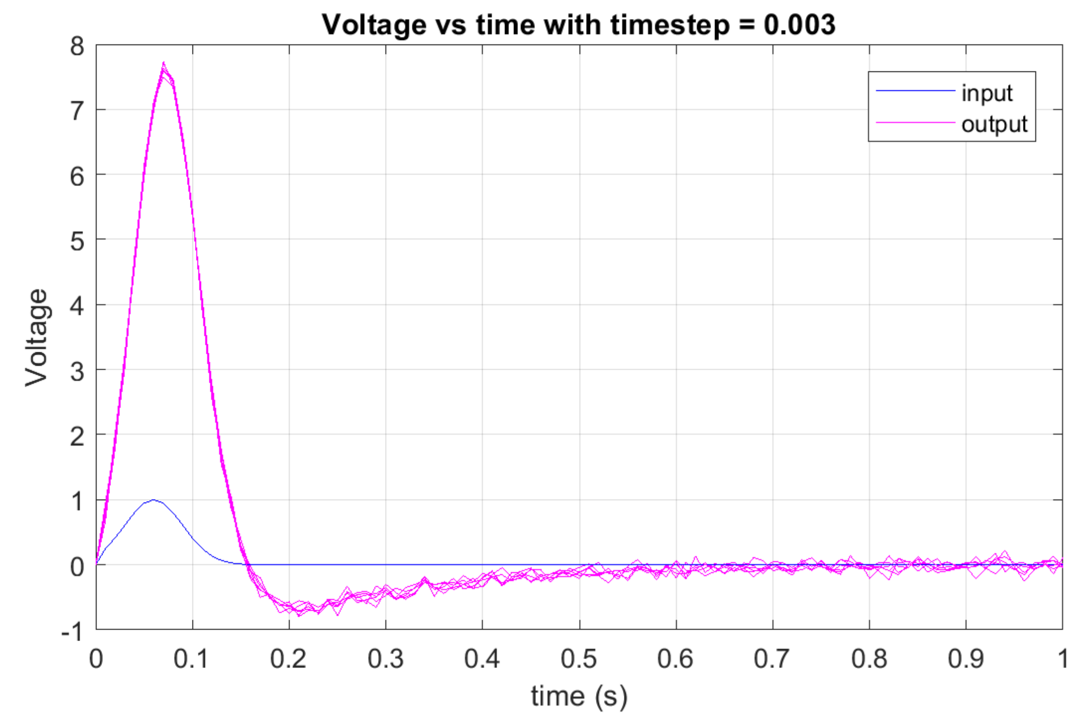

Fig 19 Plot when time step equals 0.003

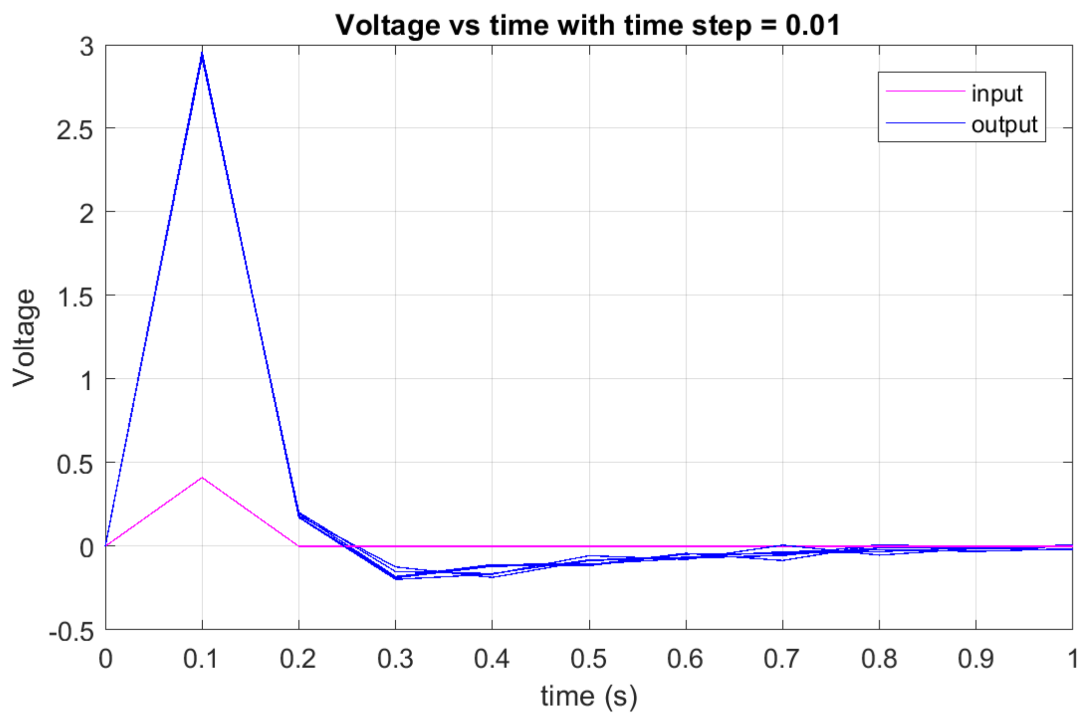

Fig 20 Plot when time step equals 0.01

What was observed was that as the time step is increased the accuracy of the simulation decreases.

Non-Linearity: In order to implement this in MATLAB another matrix is needed to represent the equations for the non-linear elements. A column vector B(V) will have to be added. The current system is non-linear it can't be solved by simple Gaussian elimination and instead the newton Raphson numerical method has to be used instead