



Birzeit University
Faculty of Engineering & Technology
Department of Electrical & Computer Engineering
ENEE4101

“Signals & Systems Analysis Using Matlab/Simulink”

Modified

Student :Mohamad Bornat #1130842

Instructor:Dr.Jamal Siam

Date: 24-4-2017

Abstract

In these three experiments, Matlab/Simulink software was used in order to simulate and study the linear systems and to model a DC Motor using Simulink to see the characteristics of the motor. Matlab was used also to design different filters to apply it on different signals we have.

Contents

Acronyms and Abbreviations	ii
List of Figures	iii
Chapter 1 Introduction	1
Chapter 2 Procedure	2
2.1 Methods and Materials.....	2
2.1.1 Linear Systems Experiment.....	2
2.1.2 DC Motor Modelling Experiment	5
2.1.3 Filters Design Experiment	6
Chapter 3 Results and Discussion	6
3.1 Linear Systems Results	6
3.2 DC Motor Simulink Model.....	11
3.1 Filters Design Results	15
Conclusion	21
References	22
Appendices	23
Appendix A.....	23

Acronyms and Abbreviations

DC	Direct Current
AC	Alternating Current
T.S	Transfer Function
AWGN	Additive White Gaussian Noise

List of Figures

Fig. 1.1:: Matlab Logo.....	1
Fig. 2.1 : Mesh Circuit	2
Fig. 2.2 : Control System Block Diagram.....	3
Fig. 2.3 : Op-Amp Circuit.....	3
Fig. 2.4 : RLC – Circuit with DC Input	4
Fig. 2.5 : RLC – Circuit with AC Input	4
Fig. 2.6 : Simulink DC Motor Model.....	5
Fig. 3.1 : frequency response using Standard plotting and complex number capabilities	6
Fig. 3.2 : Bode Diagram Mesh Circuit.....	7
Fig. 3.3: Step Response of the Control System.....	8
Fig. 3.4: Function Plot.....	8
Fig. 3.5: Bode Diagram of Op-Amp Circuit	9
Fig. 3.6: Bode Diagram of RLC Circuit.....	10
Fig. 3.7: Step Response of the System.....	11
Fig. 3.8: Step Response of the System for different torque values(0.2,0.4,1) Respectively...	12
Fig. 3.9: Step Response of the System for different resistance Values.	12
Fig. 3.10: Step Response of the System for different Voltage Values.....	13
Fig. 3.11: Step Response of the System for different resistance Values of Moment inertia. .	14
Fig. 3.12: Stem Plot of the Signal.	15
Fig. 3.13: Plot of the Signal in time domain.	15
Fig. 3.14: sine part of the signal.....	15
Fig. 3.15: Plot of the Signal in time domain with AWGN.....	16

Fig. 3.16: Plot of the Signal in time domain using sptool..... 16

Fig. 3.17: Spectrum of the Signal with AWGN..... 17

Fig. 3.18: Low pass Filter and its Spectrum. 18

Fig. 3.19: High pass Filter and its Spectrum..... 19

Fig. 3.20: Band pass Filter and its Spectrum. 20

Chapter 1

Introduction

MATLAB/Simulink is classified as High Programming Language and a special-purpose language that is a suitable choice for writing moderate-size programs that solve problems involving the manipulation of numbers. The design of the language makes it possible to write a powerful mathematical program in a few lines. The problems may be relatively complex, while the MATLAB make them simple in compare to other programming languages such as C++ or Java. As a result, MATLAB is being used in a wide variety of domains from the natural sciences, through all disciplines of engineering, to finance, and beyond, and it is heavily used in industry. Hence, a solid background in MATLAB is a very important skill for electrical engineers these days.



Fig. 1.1:: Matlab Logo.

The following chapters will discuss the use of Matlab to simulate linear systems, Modeling Machine systems such as a DC motor and designing signal's filters.

Chapter 2

Procedure

2.1 Methods and Materials

In these experimnts the following tools were used :

- Matlab program.
- Simulink.

2.1.1 Linear Systems Experiment

Different circuits were given, and the transfer function for all of these circuits was firstly found by hand in order to make the process of writing the desired codes easier.

First the circuit in figure was given to us and the code (A.1.0), (A.1.1) , (A.1.2) were used to plot the frequency response of the circuit and solving it.

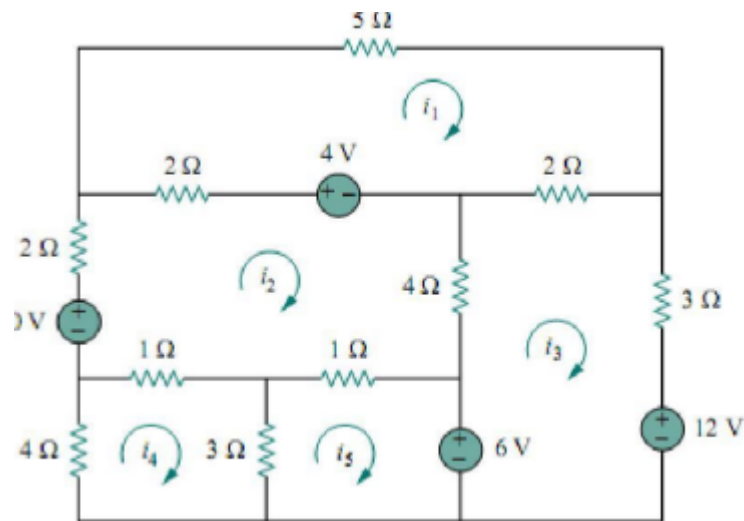


Fig. 2.1 : Mesh Circuit

After that the a matlab code (A.2) was used to build up the control system shown in the next figure.

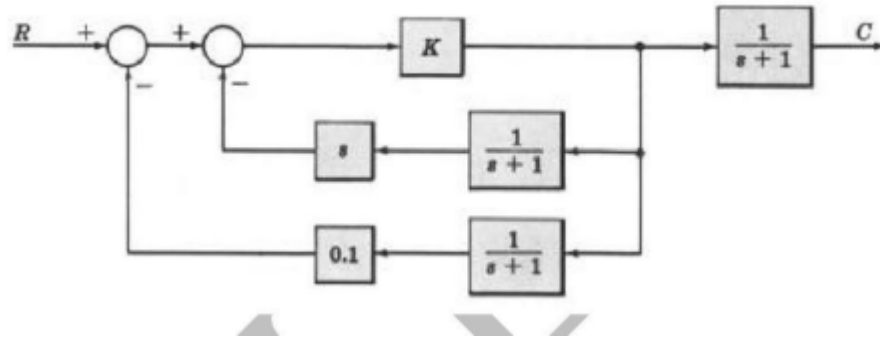


Fig. 2.2 : Control System Block Diagram

Code (A.3) was used as well to plot the function:

$$y(t)=t^2[u(t+1)-u(t)]+(t-1)[u(t)-u(t-1)]+(-1)[u(t-1)-u(t-2)] \text{ for the period } -3\text{sec} < t < 5\text{sec}.$$

Another circuit was given to us in the next figure and a Matlab code (A.4) was used to find the the poles and zeros of the transfer function and to plot the magnitude and phase response as well.

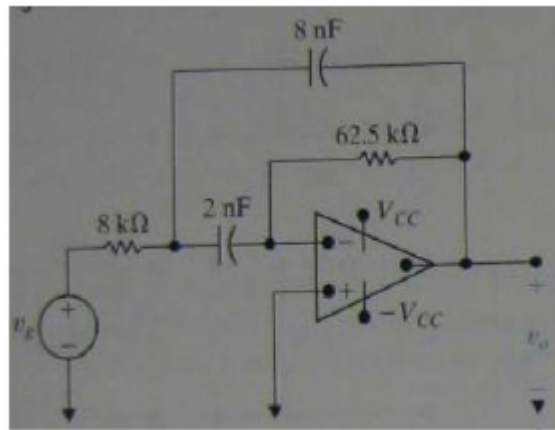


Fig. 2.3 : Op-Amp Circuit

Then the code (A.5) was used to find the roots of the characteristics equation for the following circuit.

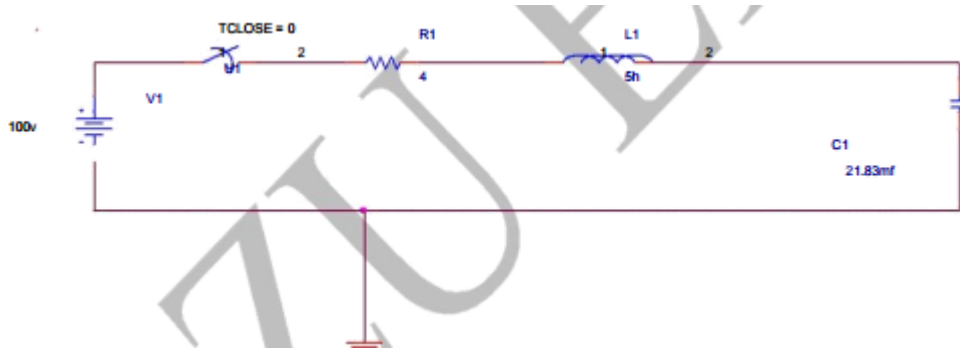


Fig. 2.4 : RLC – Circuit with DC Input

Finally, The Bode Magnitude diagram was plotted using (A.6) code for the following circuit.

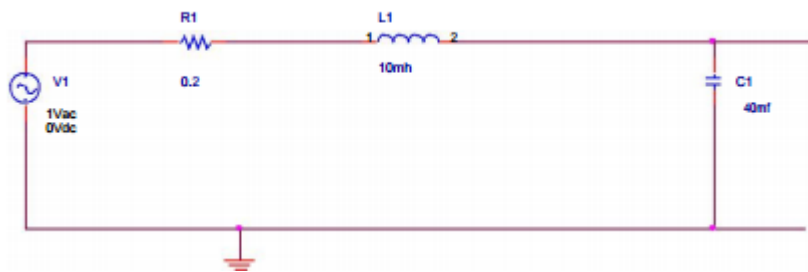


Fig. 2.5 : RLC – Circuit with AC Input

2.1.2 DC Motor Modelling Experiment

In this experiment a DC Motor Model was built using Simulink tools as in the figure above, and some parameters were changed in each part such as torque and resistance to study the effect of these changes on the DC Motor. The output figures were plotted using workspace for clearer results.

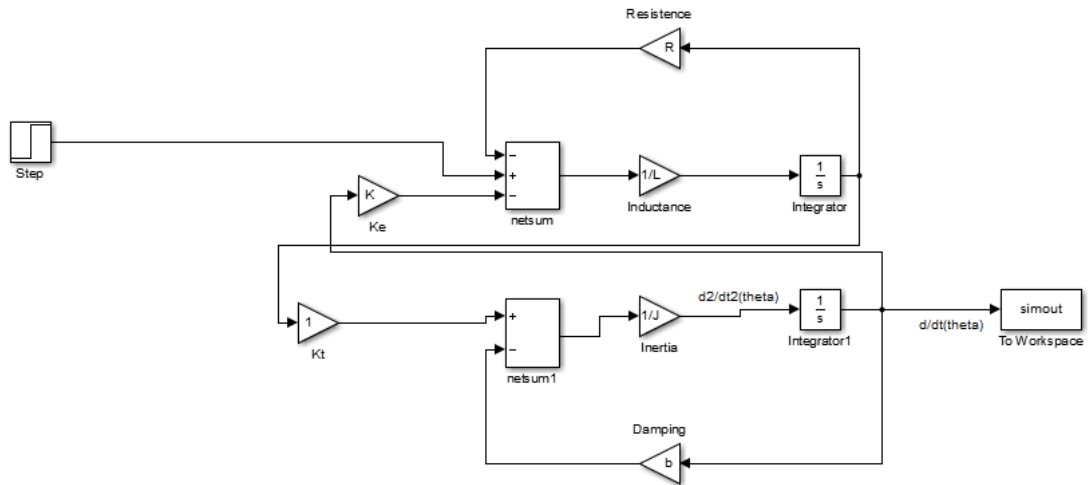


Fig. 2.6 : Simulink DC Motor Model

In order to apply different torques, the constant torque block was opened and the desired torque values were written in the block parameters as a matrix. In order to change other model parameters such as the inertia or the resistance, it was given a symbol reference for each variable (e.g. R for resistance) in the model and a Matlab code was written and applied to the model so we can change the parameters easily using the command window without any need to modify the Simulink model.

2.1.3 Filters Design Experiment

A MATLAB scripts (A.8),(A.9) that generates a discrete-time signal (x) was created. It consists of three sine signals of frequencies 100, 250 and 400Hz and with RMS amplitudes of 1, 2 and 3 respectively. Another signal was generated but this time by adding AWGN to it then the signals were plotted and attached to the report. And all to this was by using one of the Matlab tools called Sptool

Chapter 3

Results and Discussion

3.1 Linear Systems Results

The Mesh currents resulted by using the code (A.1.0) are given by:

$$x = 0.3829, 0.2110, -0.487, -0.7184, -1.9860$$

The frequency response plot generated by code (A.1.1) frequency response using standard plotting and complex number capabilities was plotted as in the following figure:

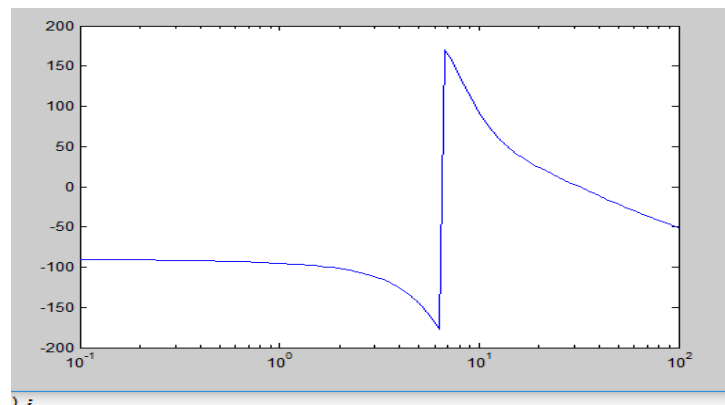


Fig. 3.1 : frequency response using Standard plotting and complex number capabilities

The Phase and magnitude Bode Diagram plot generated using code (A.1.2) is given in the following figure:

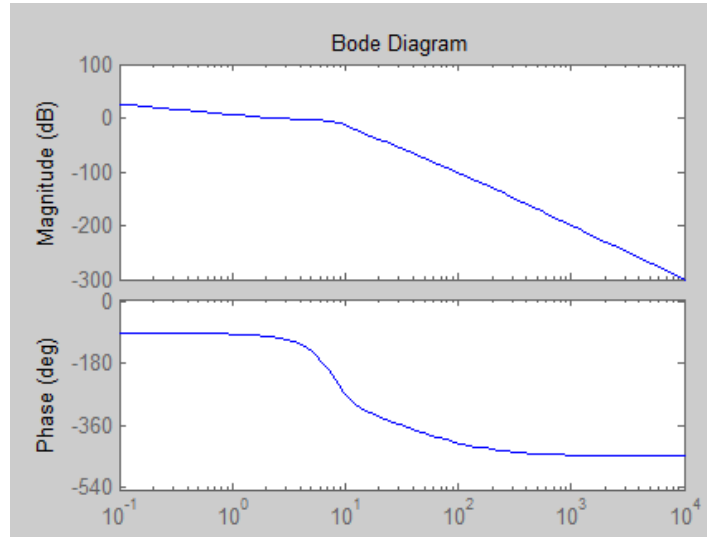


Fig. 3.2 : Bode Diagram Mesh Circuit

For the control system part , the transfer function was computed to be :

```
num/den =
      s^2 + 1.1 s + 0.1
-----
11 s^3 + 24 s^2 + 15 s + 2
>> |
```

While the step response of the systems :

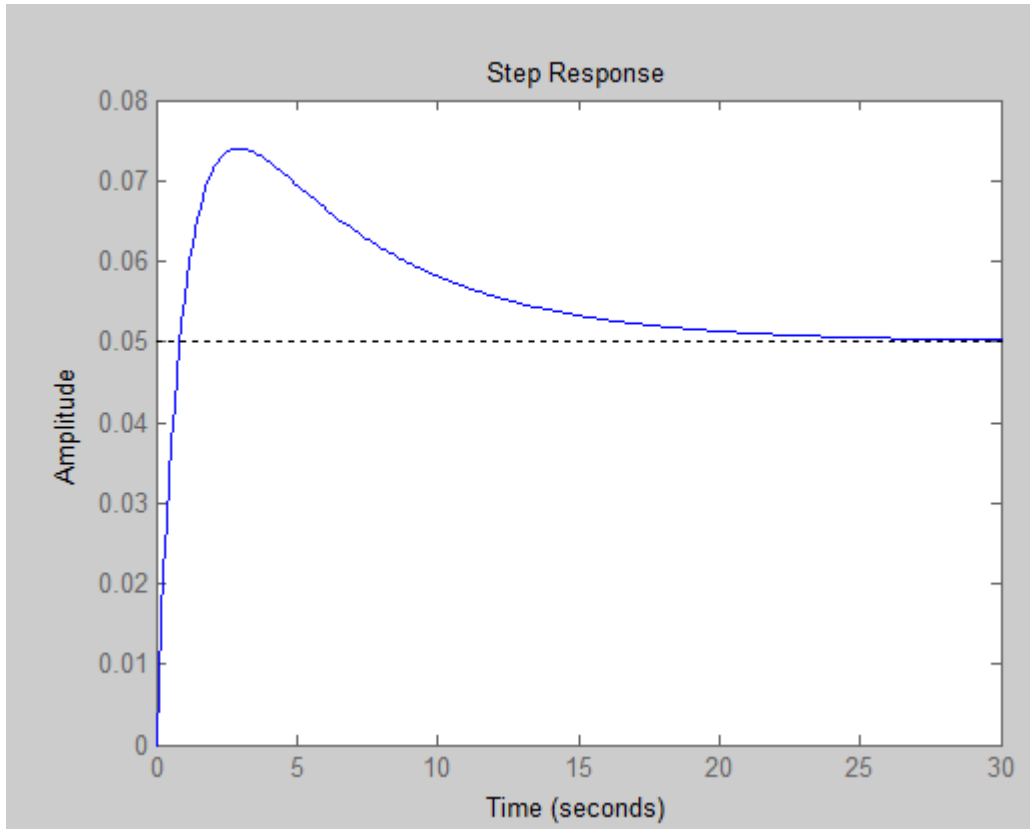


Fig. 3.3: Step Response of the System

The Plot of the given function in part 5 was found as follows :

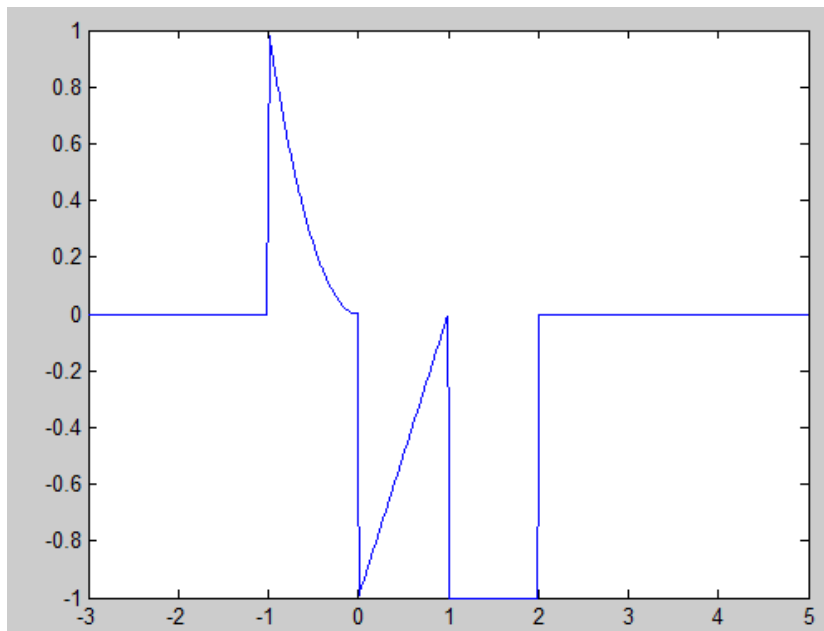


Fig. 3.4: Function Plot

In the next part of the op amp circuit the poles and zeros of the circuit were found to be :

```
n =
     0

m =
  1.0e+04 *
 -0.5000 + 1.0000i
 -0.5000 - 1.0000i
```

And the Bode Diagram of The system was plotted as below :

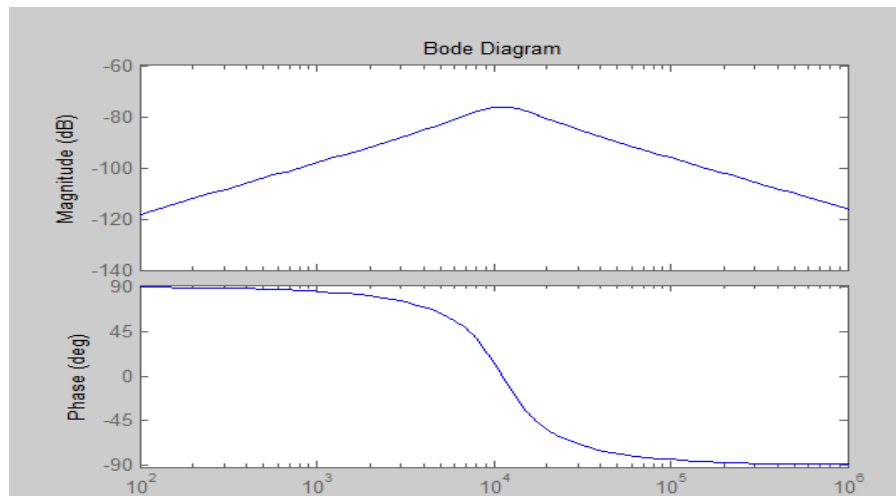


Fig. 3.5: Bode Diagram of Op-Amp Circuit

For the RLC circuit the roots of the characteristic equation were found as follows using the matlab codes :

```
m =
 -0.4000 + 3.0003i
 -0.4000 - 3.0003i
```

The transfer function was computed as :

```
num/den =
      1
-----
s^2 + 20 s + 2500
>> |
```

And finally the Bode Diagram was plotted as in the following figure :

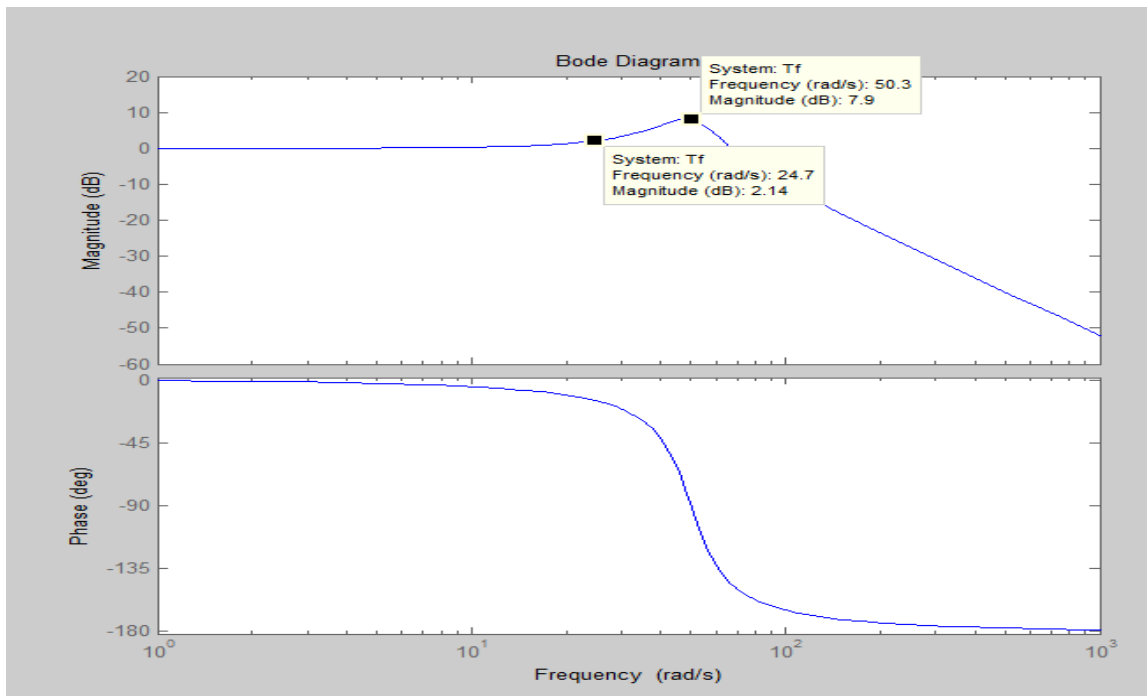


Fig. 3.6: Bode Diagram of RLC Circuit

We can notice that the maximum magnitude was at frequency 50.3 rad/s while and it was equal to 7.9 dB while at half that frequency the magnitude was decreased to 2.14 dB.

3.2 DC Motor Simulink Model

The step response of the model was found to be as in the following figure :

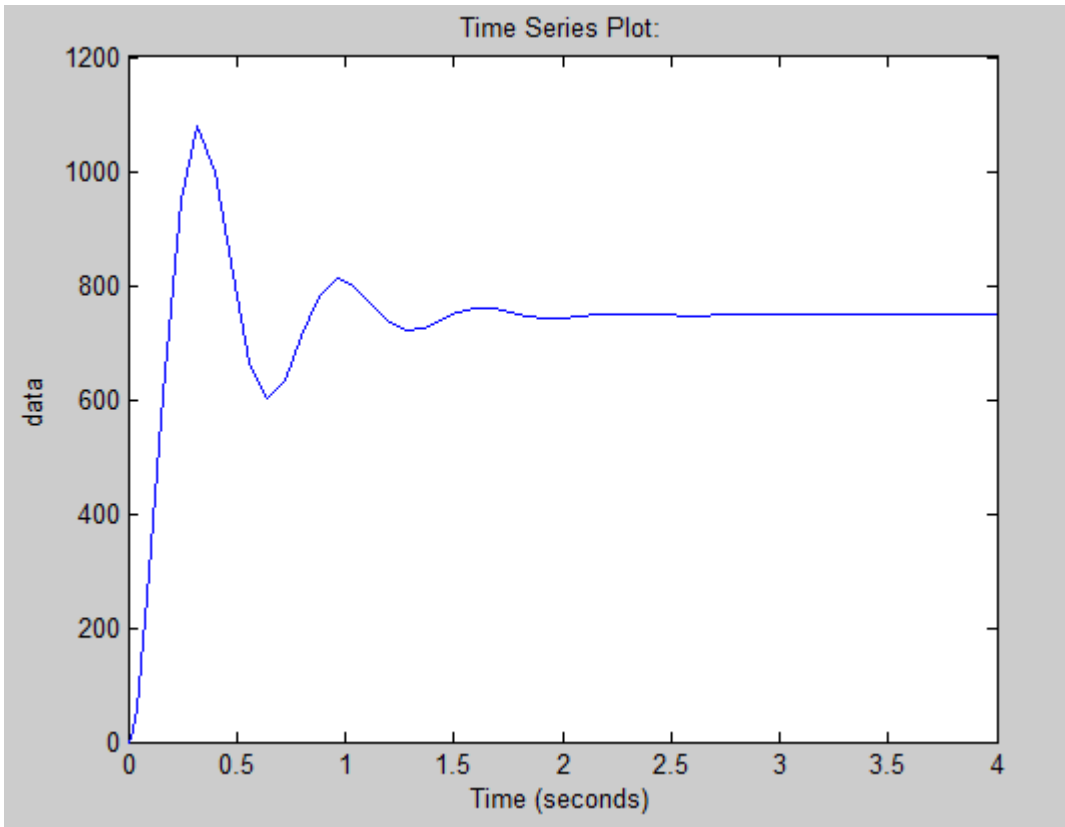


Fig. 3.7: Step Response of the System

We changed the settings for the x axis because the important area that we want to study is the variation at the first few seconds since there is no effective informations in the steady state area.

After that we studied the output for different values of torque and the outputs were as follow :

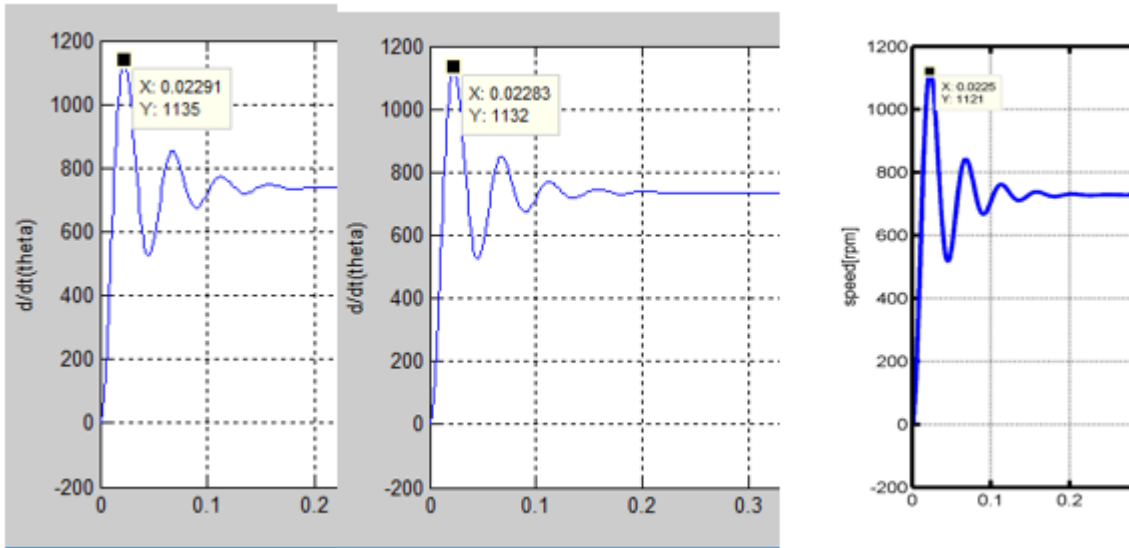


Fig. 3.8: Step Response of the System for different torque values(0.2,0.4,1) Respectively

We can notice that as the torque increases the steady state speed decreases.

The speed response for different values of R was found to be as in the following figure :

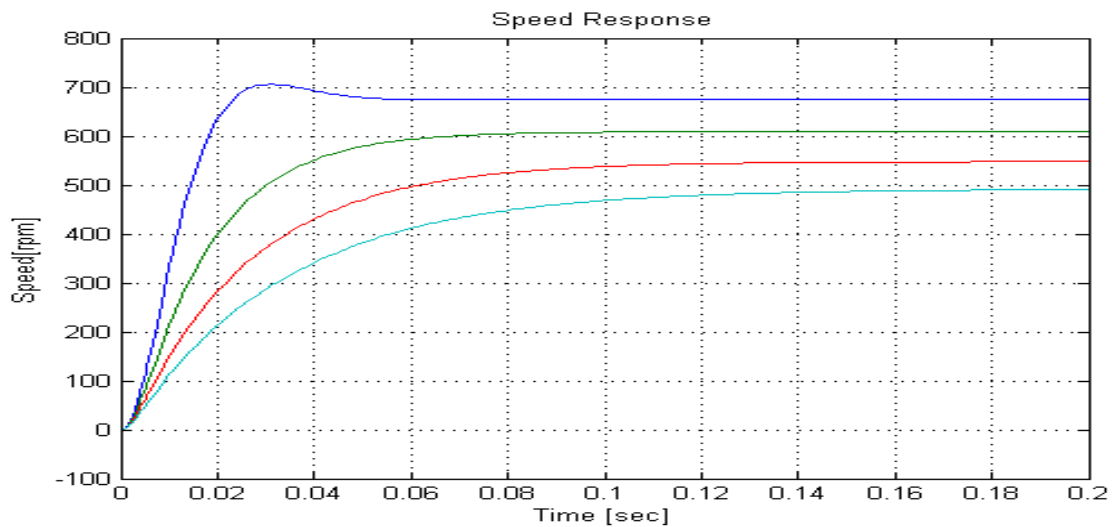


Fig. 3.9: Step Response of the System for different resistance Values.

We can notice that as the resistance increases the steady state speed decreases.

The effect of the voltage source variation on the speed response of the model was found to be as in the following figure :

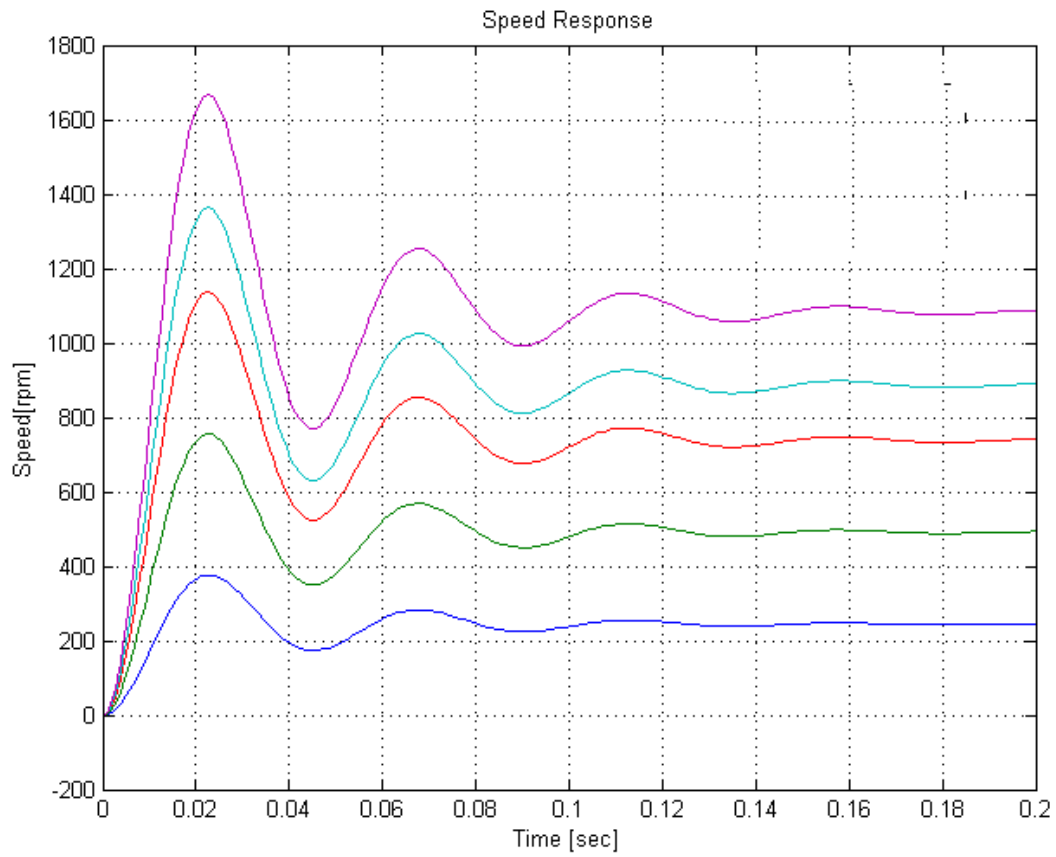


Fig. 3.10: Step Response of the System for different Voltage Values.

We can notice that as the voltage decreases the steady state speed decreases.

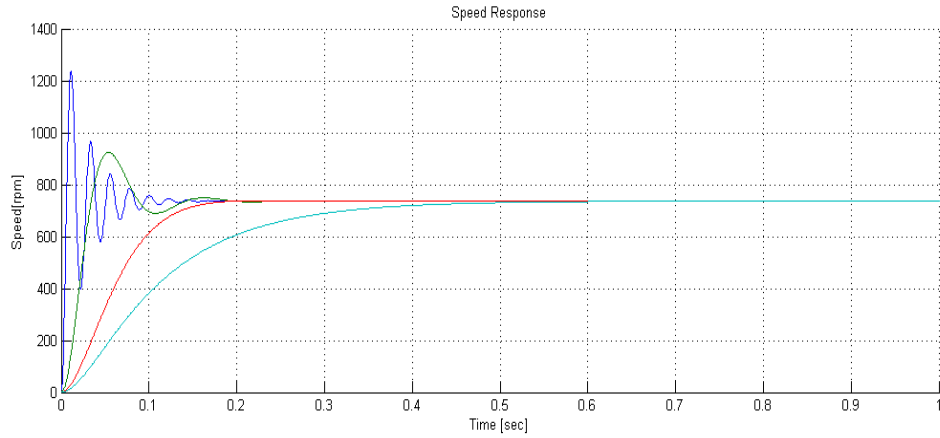


Fig. 3.11: Step Response of the System for different resistance Values of Moment inertia.

Finally we extracted the T.S of the Model using (A.6) code as follows :

$$\frac{100000}{s^2 + 55 s + 2.025e04}$$

3.1 Filters Design Results

We found the following results :

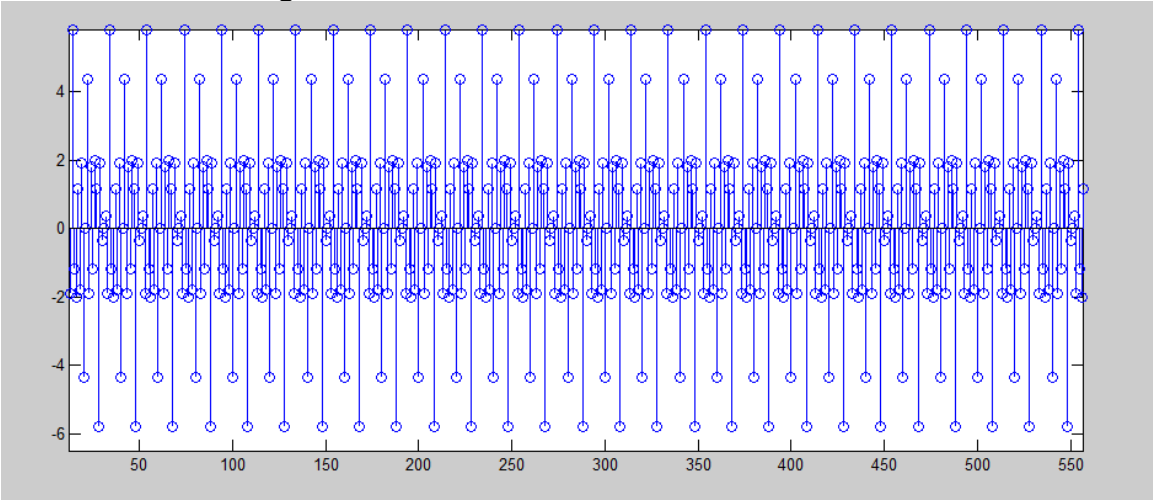


Fig. 3.12: Stem Plot of the Signal.

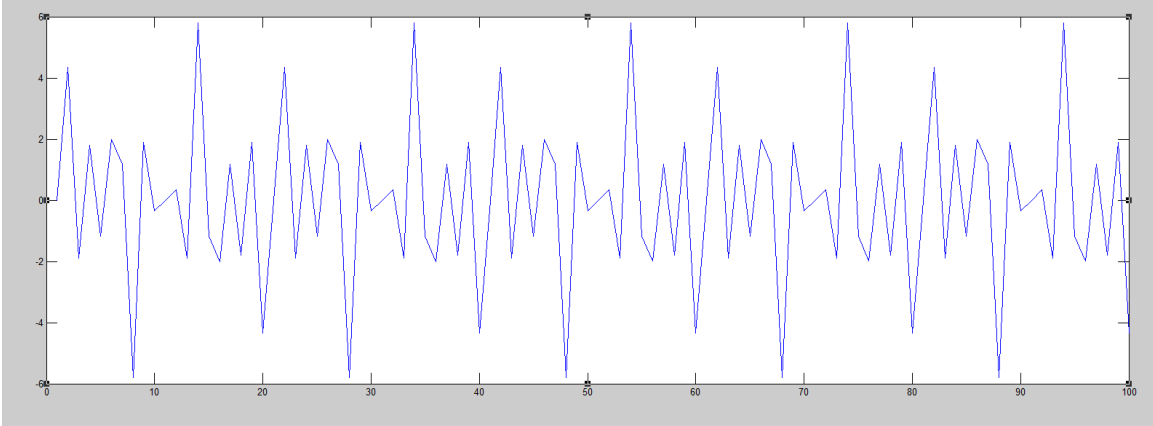


Fig. 3.13: Plot of the Signal in time domain.

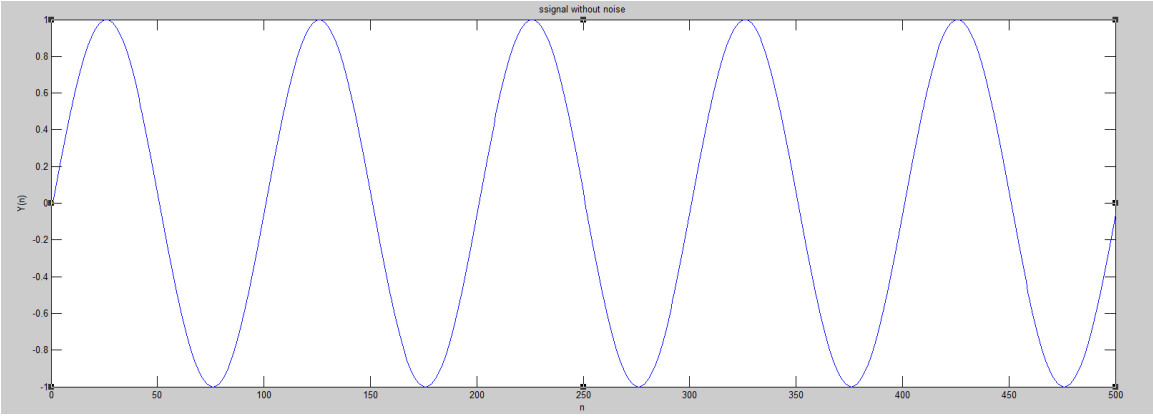


Fig. 3.14: sine part of the signal.

After adding the AWGN we got the following signal :

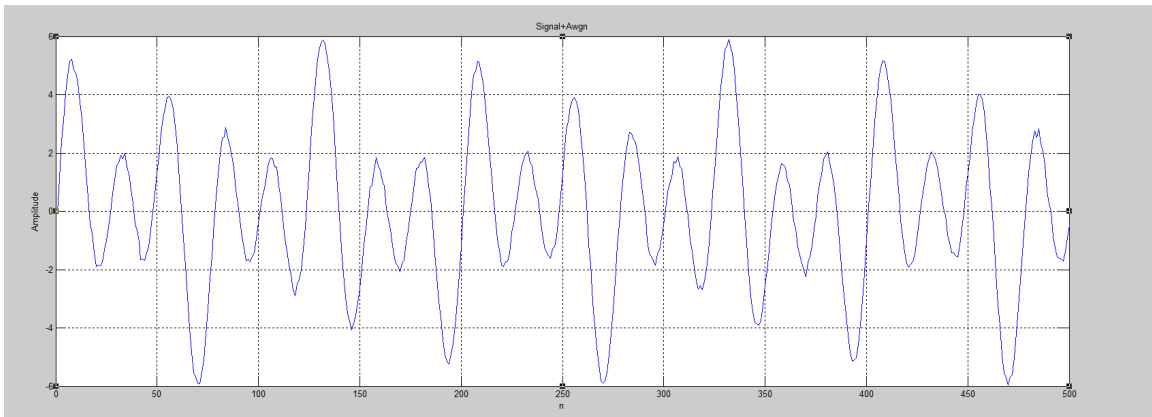


Fig. 3.15: Plot of the Signal in time domain with AWGN.

Using Sptool the following plot was generated :

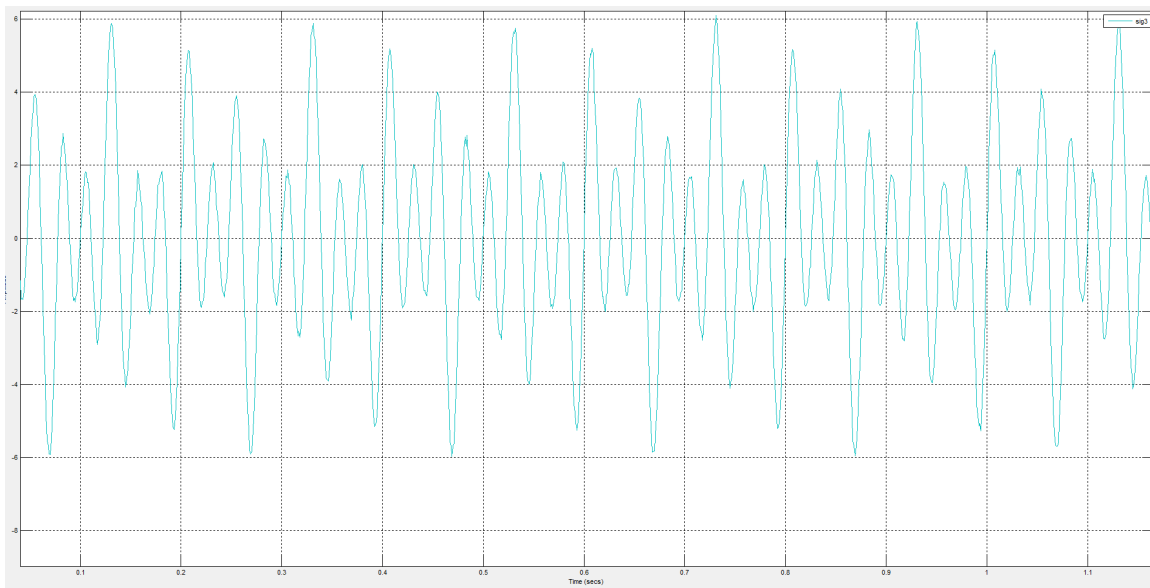


Fig. 3.16: Plot of the Signal in time domain using sptool

After that the spectrum of the signal was created as in the following figure, we can notice the peaks and the time of the signals using vertical and horizontal cursors:

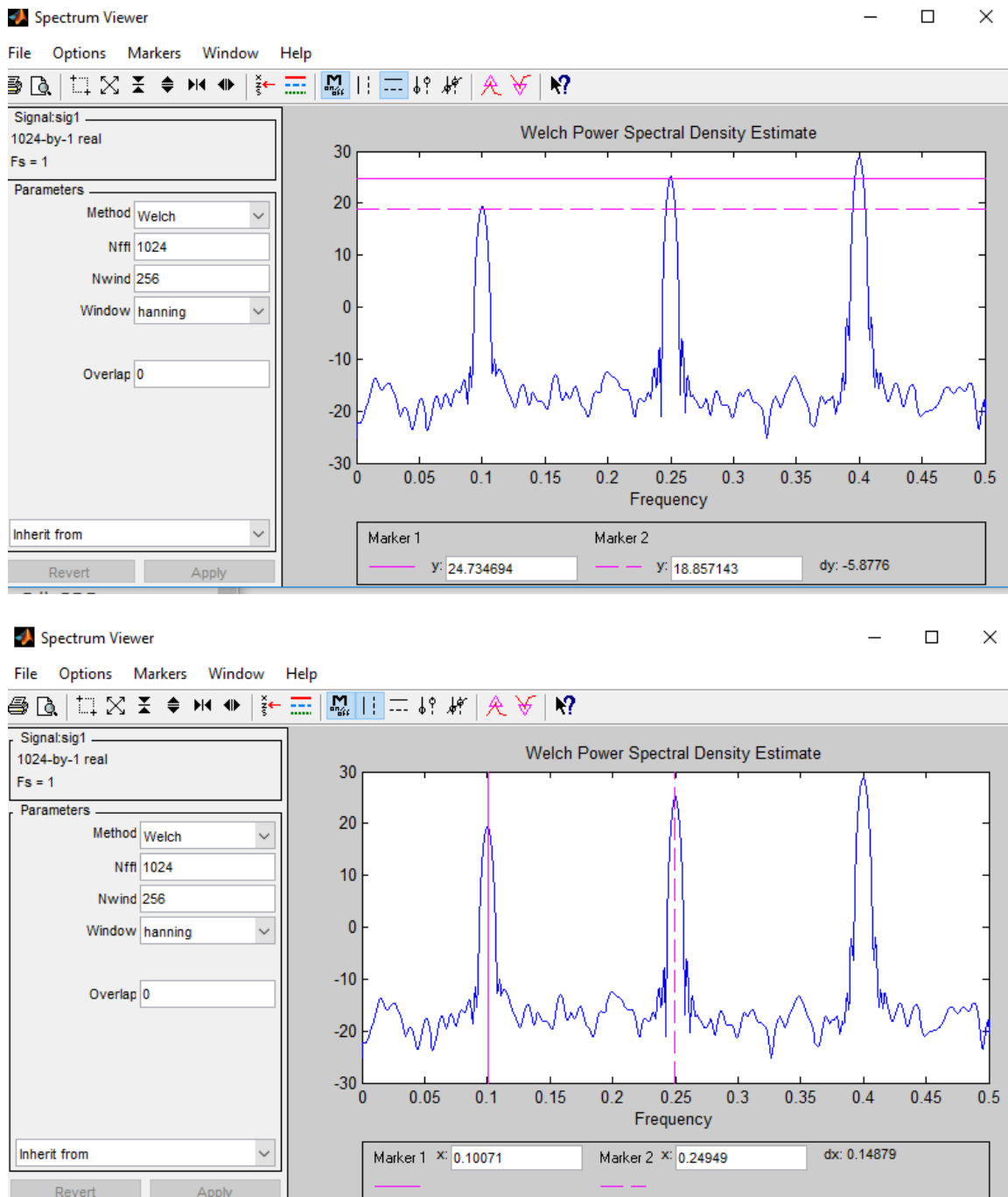


Fig. 3.17: Spectrum of the Signal with AWGN.

Finally , the three filters that were deigned , its parameters and spectrums are in the following plots :

- **LowPass Filter**

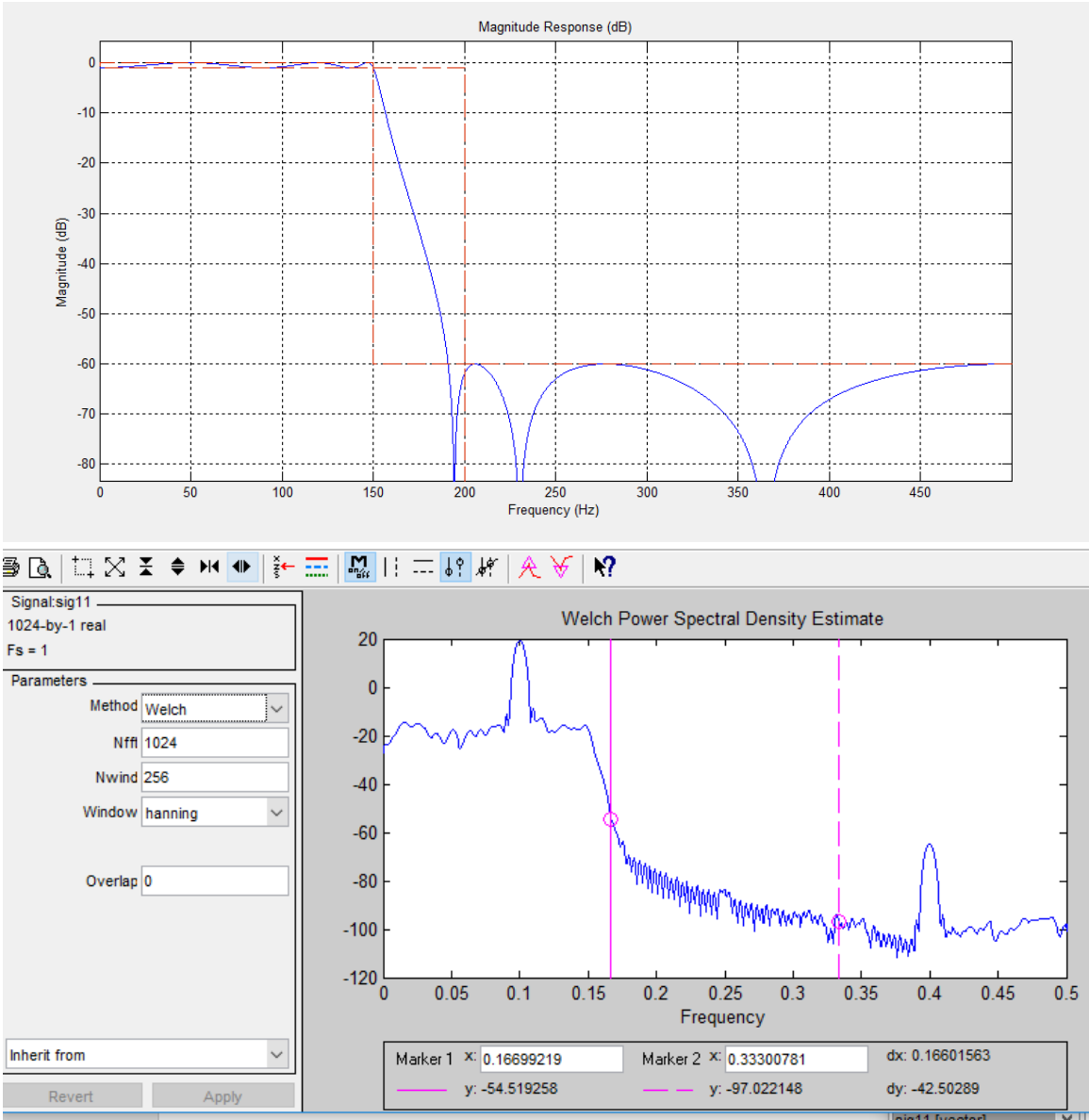


Fig. 3.18: Low pass Filter and its Spectrum.

- **HighPass Filter**

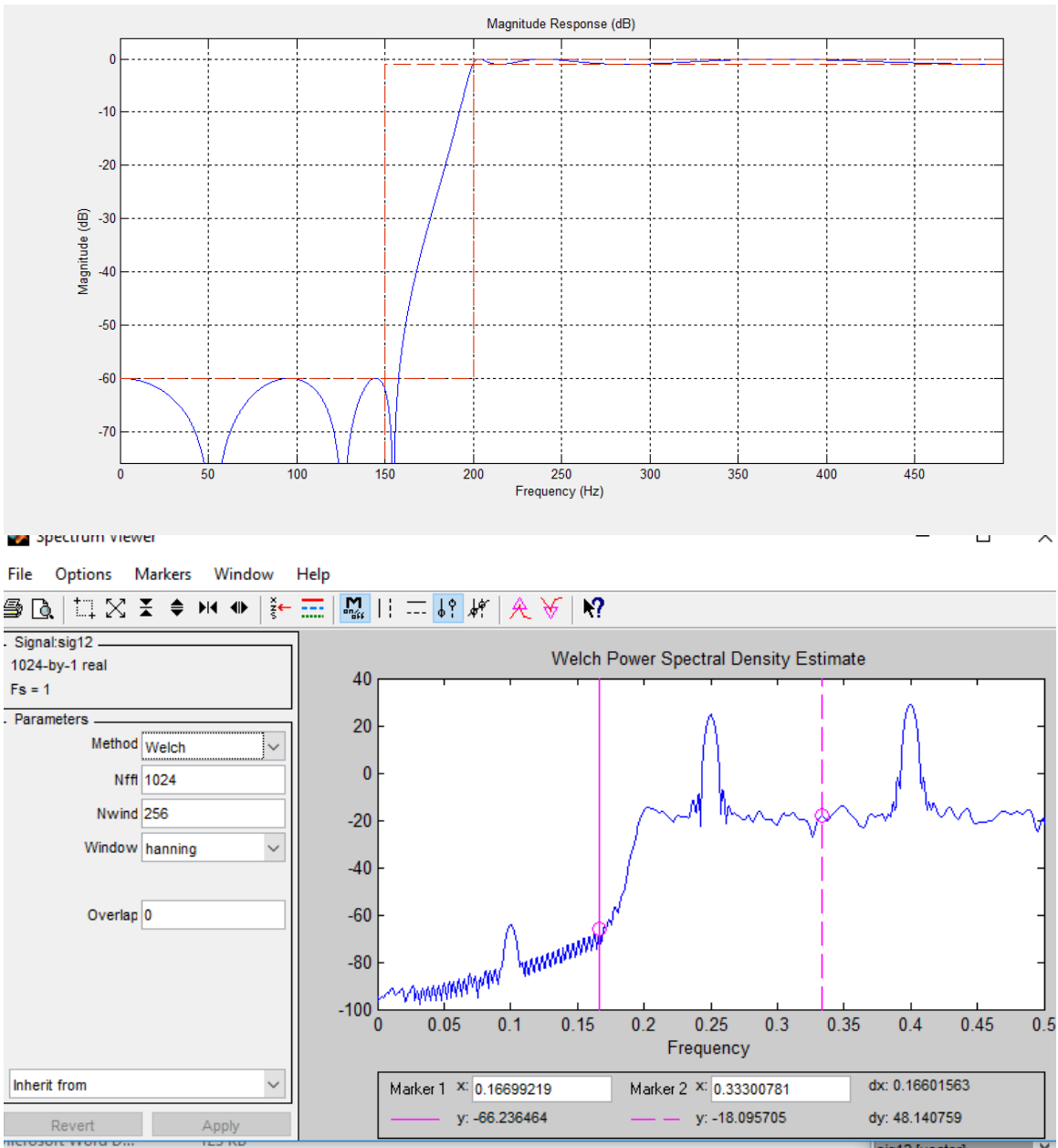


Fig. 3.19: High pass Filter and its Spectrum.

- **BandPass Filter**

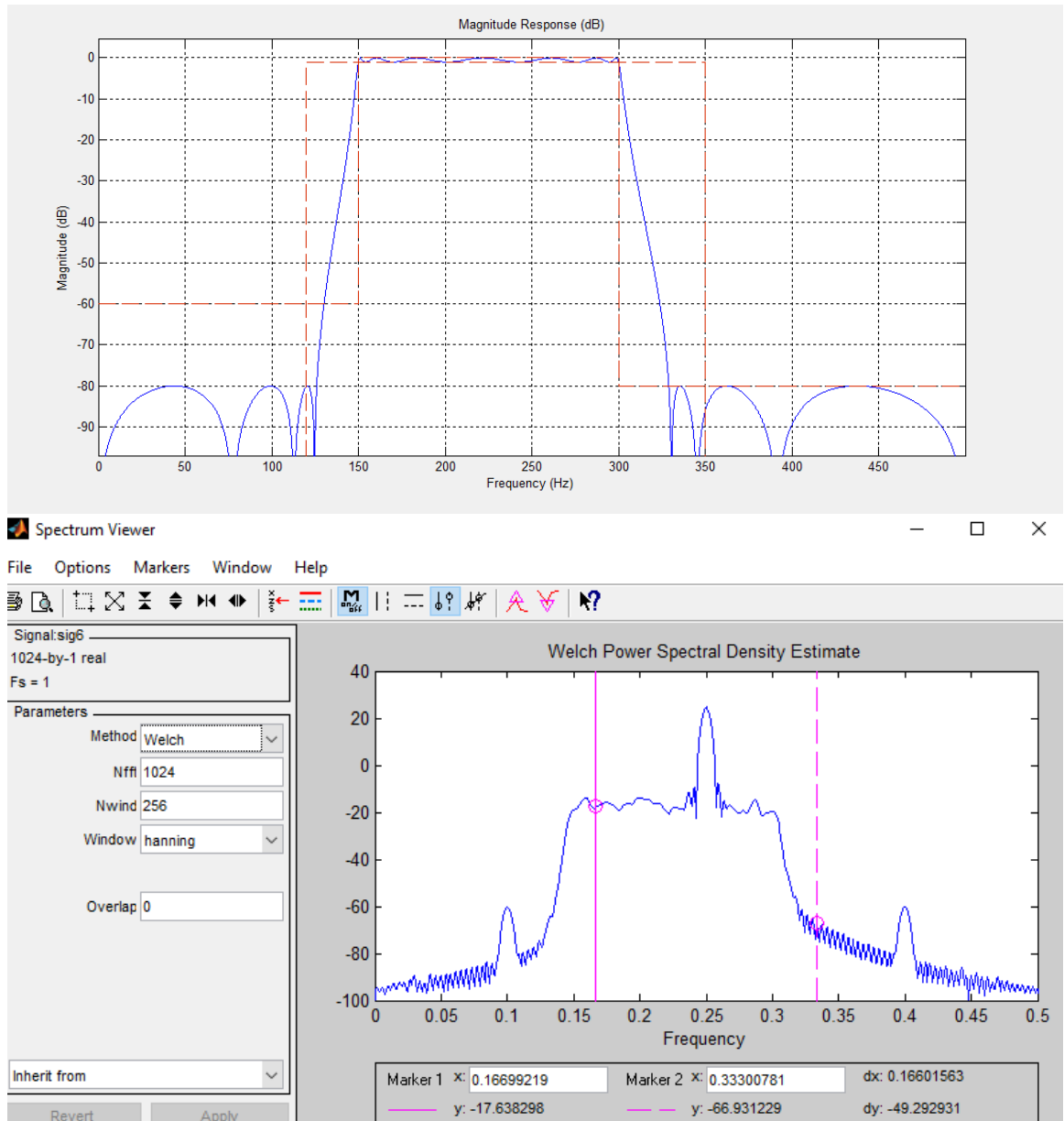


Fig. 3.20: Band pass Filter and its Spectrum.

Conclusion

In this experiments, Matlab/Simulink was used to generate different electrical systems such as dc motor and filters. Matlab approved itself as one of the best softwars used to solve and work on mathematical equations and systems.

Matlab made the learning process much easier and faster especially for the electrical engineering students and teachers and the specialized engineers as well.

References

[1] Lab Manual Booklet

Appendices

Appendix A

```
A = [ 9 -2 -2 0 0;-2 10 -4 -1 -1;-2 -4 9 0 0;0 -1 0 8 -3;0 -1
0 -3 4];
    B = [ 4;6;-6;0;-6];
    C=inv(A);
    x=C*B;
display(x
```

(A.1)

```
num = conv([10^5 7*10^5],[1 13]);
den1 =conv ([1 25 0],[1 55]);
den2 =conv ([1 7 75],[1 7 45]);
dum =conv (den1,den2);
m=logspace(-1,2,101);
Gain=freqs(num,dum,m);
mag=abs(Gain);
db=20*log10(mag);
ph=angle(Gain)*180/pi;
semilogx(m,db)
semilogx(m,ph)
```

(A.2)

```
num = conv([100000 700000],[1 13]);
d1 =conv ([1 25 0],[1 55]);
d2 =conv ([1 7 75],[1 7 45]);
dum =conv (d1,d2);
w=logspace(-1,2,200);
bode(num,dum);
```

(A.3)

```

num1=[1 0];den1=[1 1];
num2=[0 0.1];den2=[1 1];
num3=[0 10];den3=[0 1];
num4=[0 1];den4=[1 1];
[numo1,deno1]=parallel(num1,den1,num2,den2);
[numc1,denc1]=feedback(numo1,deno1,num3,den3);
[numc2,denc2]=series(numc1,denc1,num4,den4);
printsys(numc2,denc2,'s');

```

(A.5)

```

t=-3:0.01:5;
y1=(t.^2).*(heaviside(t+1) - heaviside(t));
y2=(t-1).*(heaviside(t) - heaviside(t-1));
y3=(heaviside(t-1) - heaviside(t-2));
y=y1+y2-y3;
plot(t,y)

```

(A.6)

```

num=[1.5625 0];
den=[1 10000 125000000];
Tf=tf(num,den);
n=zero(Tf)
m=pole(Tf)
bode(Tf)

```

(A.7)

```

n=0:1:1023;
fs=1000;
T1=sin(2*pi*n*100/fs);
T2=2*sin(2*pi*n*250/fs);
T3=3*sin(2*pi*n*400/fs);
s=T1+T2+T3;
stem(s)

```

(A.8)

```

y=awgn(s,20);
plot(y);
grid;
title('Signal+Awgn');
ylabel('Amplitude');
xlabel('n')

```

(A.9)