ECGR 2254 – Project 2

By: Luis Umana

Collaborators:

Andrew Hord

Timothy Johnson

Problem 1

A: Determine the Fourier Series

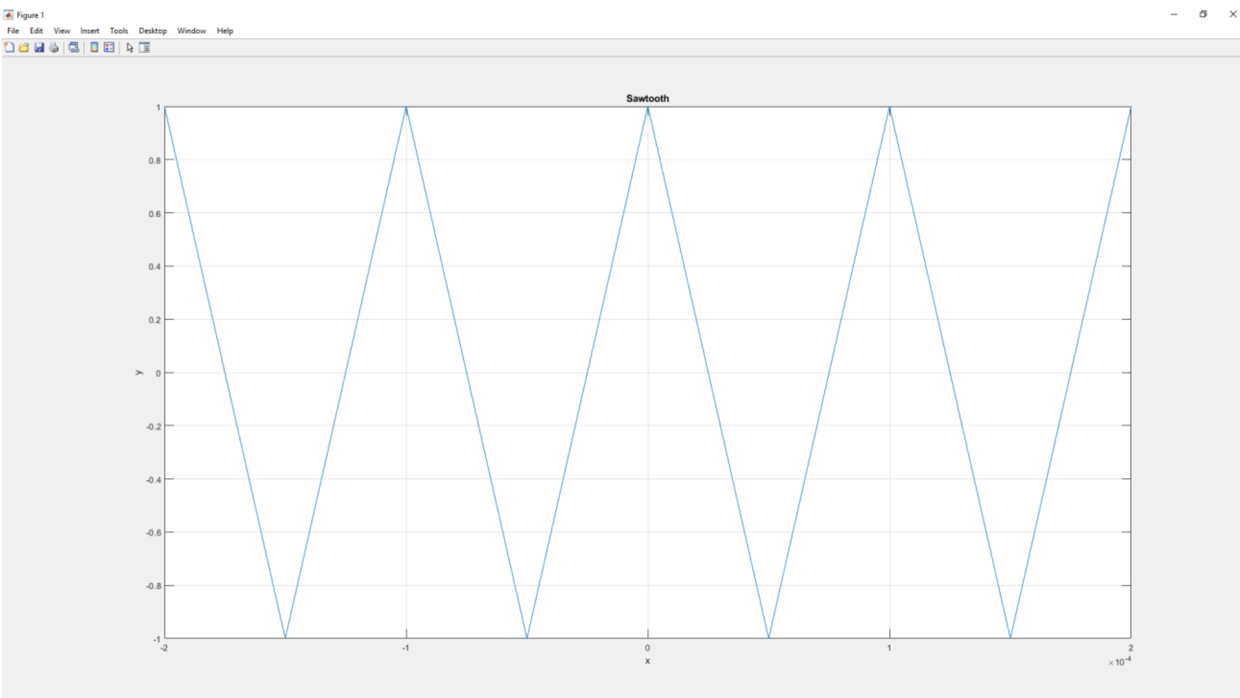
A.1: Determine the Period

The axis for time is scaled to 1*10^-4 because, that is how long it takes for the waveform to repeat itself based on the provided graph.

A.2: Sawtooth Command

```
% Sawtooth Plot
T=1e-4;
omega = 2*pi/T;
t_plot = (-2*T:T/10000:2*T);
S = -1*sawtooth((omega*t_plot)-T/2,0.5);
plot(t_plot,S);
grid on
```

A.3: Does it have the same phase as the figure provided?



After creating my sawtooth, the graph provided in the project and the sawtooth created in Matlab do appear to have the same waveform. The function for the sawtooth use omega multiply by time and subtraction of the waveform by half of the period. This causes the waveform to shift over and lets the voltage by high when time is 0. The second part of the function, 0.5, is used to turn the waveform into symmetrical triangle waves.

Code to calculate Cn and angle_n:

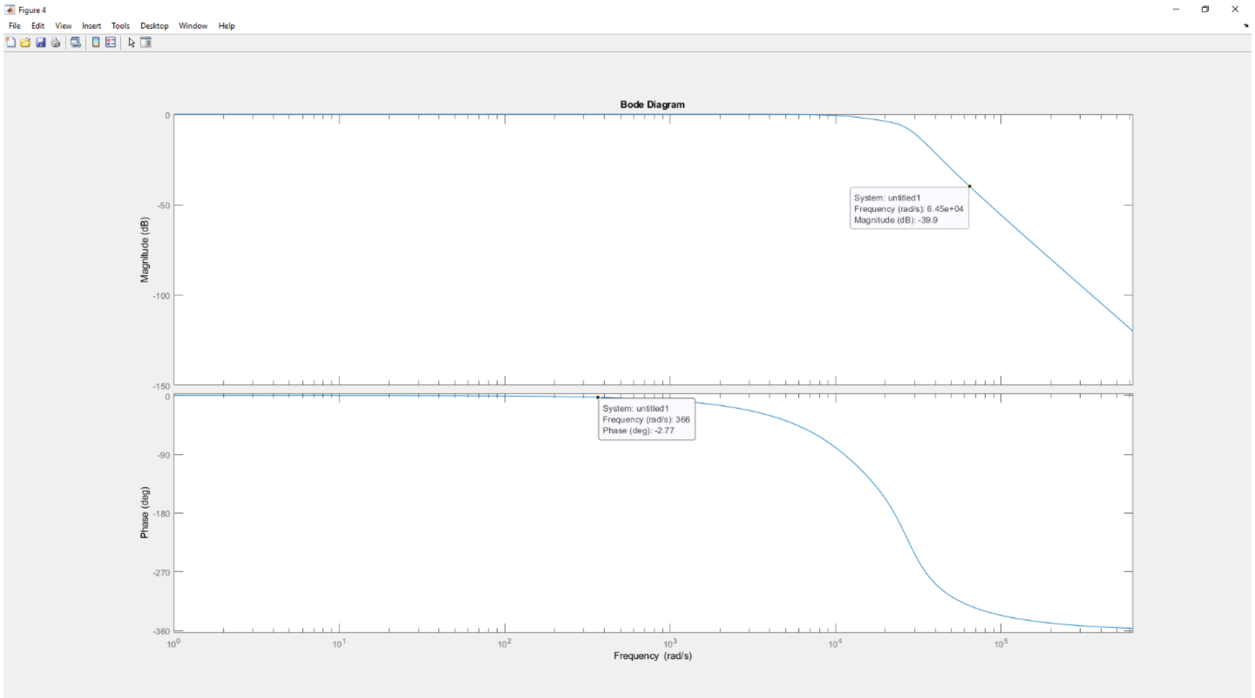
```
n = [-3:1:3];
alpha_n = [];
for k = 1:1:length(n)
    if n(k) == 0;
        int = S;
        alpha_n = [alpha_n trapz(t,int)*(1/T)];
    else
        int = S.* exp(-j*n(k)*omega*t);
        alpha_n = [alpha_n trapz(t, int)*(1/T)];
    end
end
alpha_mag = abs(alpha_n)
angle_alpha = angle(alpha_n)*180/pi

%Had an issues where it would say "Unrecognized function or variable
%'alpha_1'" so I hardcoded the values
alpha1 = 0.450;
alpha2 = 0;
alpha3 = 0.4053;
angle1 = 0;
angle2 = -29.7448;
angle3 = 0;


C1 = 2*alpha1;
C2 = 2*alpha2;
C3 = 2*alpha3;
angle_1 = rad2deg(angle1)
angle_2 = rad2deg(angle2)
angle_3 = rad2deg(angle3)
```

B: Design the Filter

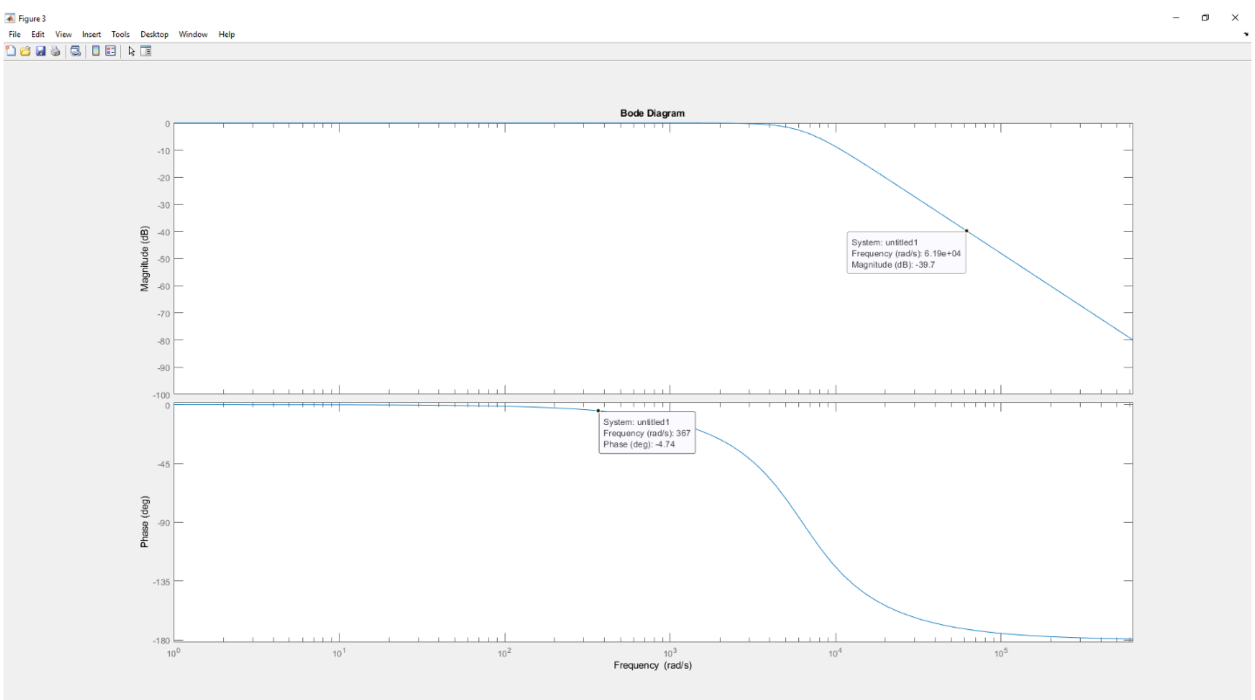
B.1: n(t) must be reduced to about 1%

B.2: Select an appropriate filter to meet the requirements.

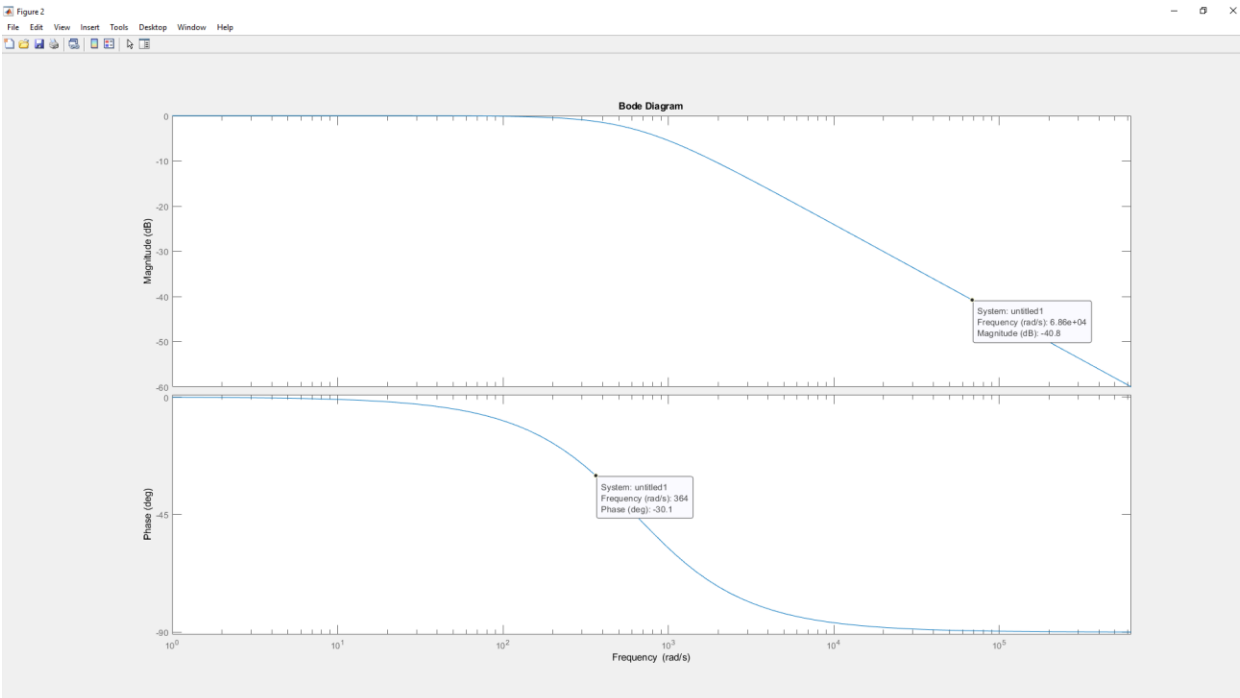
```
%H_4
wc1 = 2*pi*3150;
num3 = [wc1^4];
den3 = [1 2*wc1 3.414*wc1^2 2.6132*wc1^3 wc1^4];
figure (4);
bode(tf(num3,den3),{1,2*pi*100000})
```



H4 had a w_c value of 6300*pi which helped meet the requirement of having the magnitude reach -40 at the frequency 20000*pi. The phase shift of 120*pi also meets the requirement of being less than 4 degrees with a value of 2.77 degrees. Compared to the two filters below, it is clear that H4 was the correct filter.



H2 can reach the required magnitude, however it barely fails to meet the requirement of the phase shift needing to be less than 4 degrees.



While H1 does appear to have a similar magnitude to the expected results, its phase angle is too large to be considered the correct filter.

Problem 1 Code:

```
clc
%Variables
T=1e-4;
omega = 2*pi/T;
S = -1*sawtooth(-omega*t,0.5);
t = (-T/2:T/10000:T/2);

%Problem 1A
n = [-3:1:3];
alpha_n = [];
for k = 1:1:length(n)
    if n(k) == 0;
        int = S;
        alpha_n = [alpha_n trapz(t,int)*(1/T)];
    else
        int = S.*exp(-j*n(k)*omega*t);
        alpha_n = [alpha_n trapz(t, int)*(1/T)];
    end
end
```

```
alpha_mag = abs(alpha_n)
angle_alpha = angle(alpha_n)*180/pi

%Had an issues where it would say "Unrecognized function or variable
%'alpha_1'" so I hardcoded the values
alpha1 = 0.450;
alpha2 = 0;
alpha3 = 0.4053;
angle1 = 0;
angle2 = -29.7448;
angle3 = 0;


C1 = 2*alpha1;
C2 = 2*alpha2;
C3 = 2*alpha3;
angle_1 = rad2deg(angle1)
angle_2 = rad2deg(angle2)
angle_3 = rad2deg(angle3)

% Sawtooth Plot
t_plot = (-2*T:T/10000:2*T);
S = -1*sawtooth(omega*t_plot,0.5);
plot(t_plot,S);
grid on

title ('Sawtooth')
xlabel ('x')
ylabel ('y')

 VA = C1*cos(angle_1)+C2*cos(angle_2)+C3*cos(angle_3)
t_plot1 = (-T/2:T/10000:T/2);
 Z = -1*sawtooth(omega*(t_plot1-T/2),0.5);
 VS = 2*cos(2*pi*60*t_plot1)+Z+VA;

%Problem 1B
%H_1 -WRONG FILTER
num1 = [1];
den1 = [1/(2*pi*100) 1];
figure (2)
bode(tf(num1,den1),{1,2*pi*100000})

%H_2 -WRONG FILTER
wc = 2*pi*1000;
num2 = [wc^2];
den2  = [1 (2/sqrt(2))*wc wc^2];
figure (3);
bode(tf(num2,den2),{1,2*pi*100000})

%H_4- CORRECT Filter
wc1 = 2*pi*3150;
num3 = [wc1^4];
den3 = [1 2*wc1 3.414*wc1^2 2.6132*wc1^3 wc1^4];
figure (4);
bode(tf(num3,den3),{1,2*pi*100000})
```

Problem 2

A: Determine the differential equation for i(t)

$$Vs - K\Omega = L * \left(\frac{di}{dt}\right) + R * i$$

The circuit contains two voltage sources, Vs and KΩ. The circuit is a first order RL circuit that is determined by voltage in the resistor and inductor.

B: Determine the transfer function

$$H(w) = \frac{\Omega w}{V_a w}$$

$$\Omega w(w) = \frac{K_{Ia} w}{jwJ + B}$$

$$V_a = Ia * R + j * w * L * I_a(w) + K\Omega(w)$$

$$H(w) = \frac{\frac{K_{Ia} w}{jwJ + B}}{Ia * R + j * w * L * I_a(w) + K\Omega(w)}$$

$$H(w) = \frac{K}{(j * w * J + B) * (R + j * w * L + \frac{K^2}{j * w * J + B})}$$

C: Plot the magnitude of the transfer function

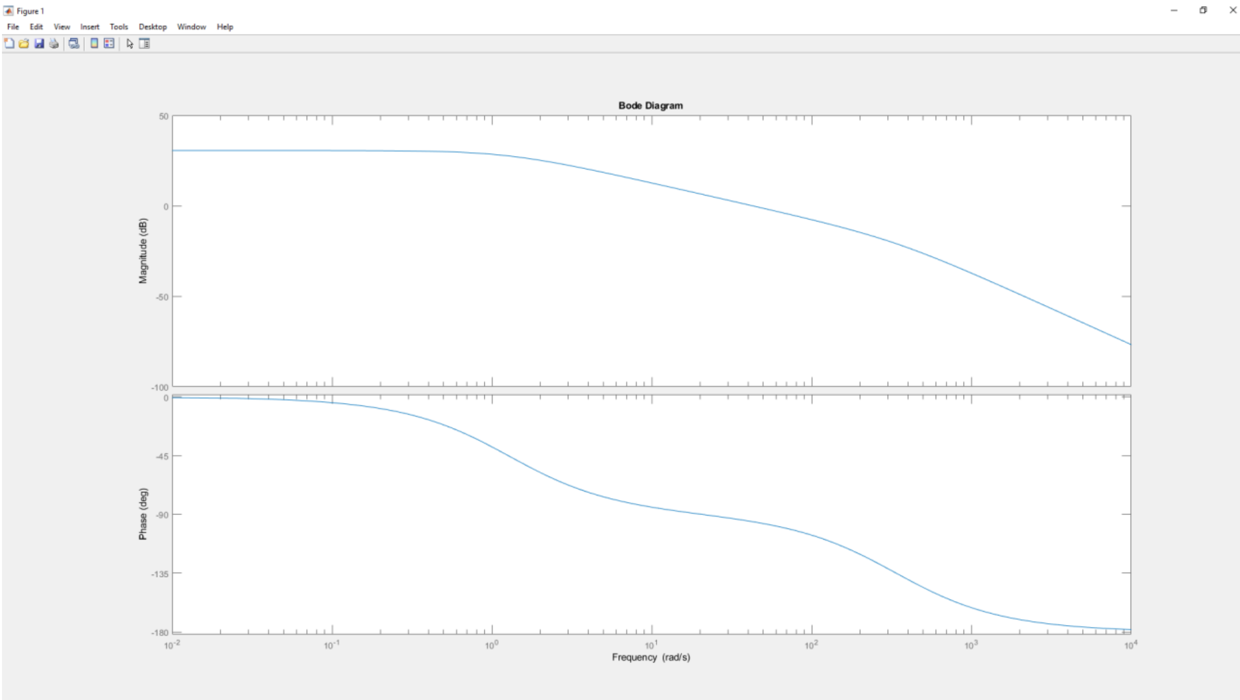
C1: Use the logspace command to create 10,000 frequencies

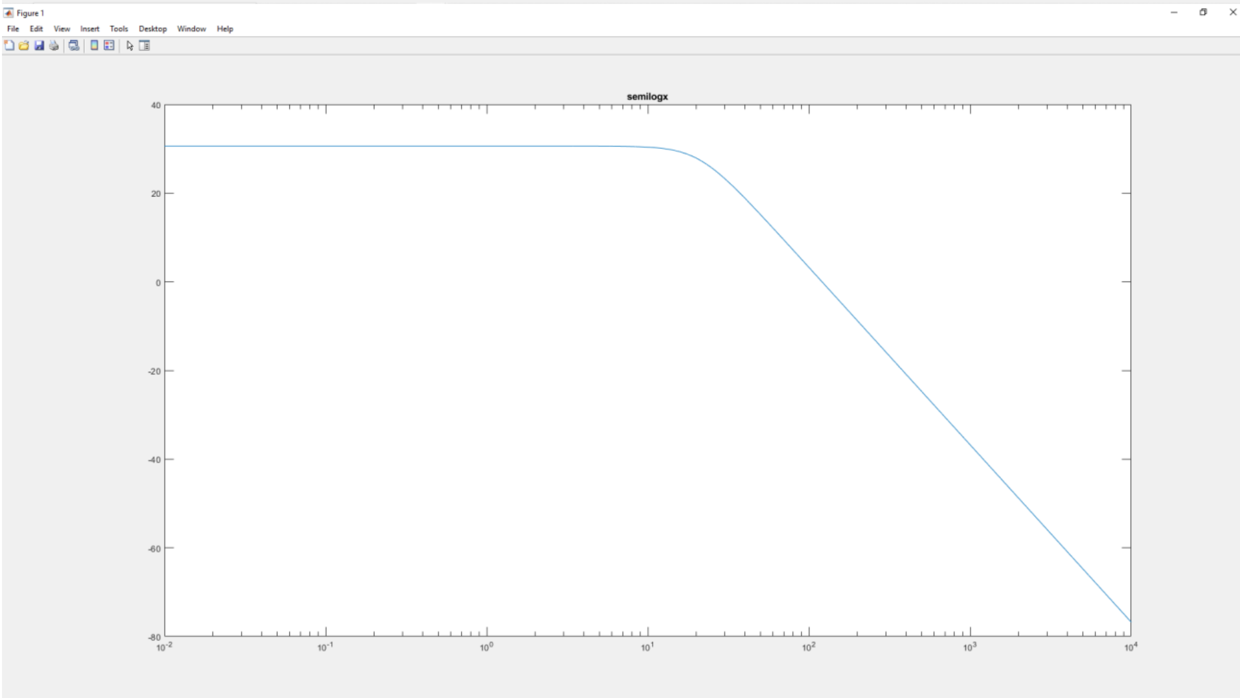
```
W = logspace(-2,4 , 10000);
H = K./((j*W.^2*L*J)+((j*W)*(L*B*R*J))+(R*B+K^2))
semilogx(W, 20*log10(abs(H)));
title('semilogx')
```

C2: Generate the vector containing the elements of the complex function

```
W = logspace(-2,4 , 10000);
H = K./((j*W.^2*L*J)+((j*W)*(L*B*R*J))+(R*B+K^2))
semilogx(W, 20*log10(abs(H)));
title('semilogx')
```

C3: Plot the transfer function using the semilogx

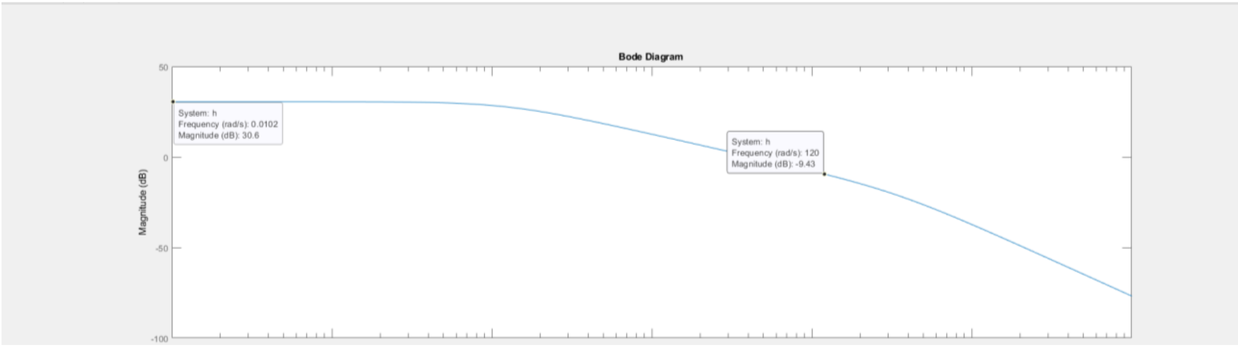




The first graph was made with the bode plot method and the second graph is made with the semilogx method. The transfer function in the bode diagram relates the different equation of the output and input, Ω(w) and V(w). The logspace commands creates 10,000 points with a range from 0.01 to 10,000 Hz. The logspace command helps provide a cleaner view of what the semilogx command magnitude plot should look like.

D: Using your plot from Part c

D1: Determine the switching frequency



The switching frequency can be found when the graph is near 1% of the initial voltage. The bode plot starts at 30.6dB applying the -4dB will cause the switching frequency to have a magnitude of -9.4Db.

Switch = 120Hz

D2: Determine the DC Volt required to turn the motor at 324 rad/sec

$$\frac{\Omega w}{Vw} = \frac{K}{(R * B + K2\,)}$$

$$Vw = \ \Omega w * \frac{R * B + K\verb|^|2}{K}$$

$$Vw = \ \Omega w * \frac{R * B + K\verb|^|2}{K}$$

Using the formula above, we can determine the value of the DC voltage plugging in the values provided by the project. Ωw= 324, R=3.38, B=0.5e^(-5), and K= 0.029, The DC Voltage would be 9.58 Volts

E: Compute the DC term of the Fourier series.

```
%Problem 2E
Voltage = 324 * ((R*B + K^2)/K);
D = Voltage/12;
D = 79.87%
```

D represents the duty cycle of the waveform, and voltage is the area under the curve. The duty cycle will cause the waveform to stay at a high value for a set period of time. Using the formula below:

$$H(w) = \frac{V(w) * T}{12T}$$

We can find duty by canceling out the periods and doing division, which gives out 79.87%. This tells us that for 79.86% of the cycle will stay at a high value before staying low for the rest of the time.

F: What is the amplitude of C1? Assume D=0.5 to keep things simple

$$\int_0^{DT} 12\exp\left(-j * w * T\right)$$

$$= \frac{-12\exp(-j * w * T)}{(j * w * T)}$$

$$= -\left(\frac{12}{j * w * T}\right) * \exp(-j * w * T) + \left(-\frac{12}{j * w * T}\right) * \exp\left(-j * w * 0.5 * T\right)$$

```
Df = 0.5;
a0 = (12*Df *T)/T;
w = 2*pi/T;
Alpha1 = (12/(j*w*T)) + ((-12/(j*w*T)).* exp(-j*w* 0.5 *T))
Angle1 = angle(Alpha1)*180/pi
Mag = 2*Alpha1
```

Alpha1 becomes 3.8197 with an angle of -90 degrees. To get the C1 value we simply multiply the Alpha1 value by 2. So C1 becomes 7.64<-90.

G: Simulate the response of the motor when the voltage is a PWM waveform

```
%Problem 2G
Wave = 6*square(TimeStep* w, 79.87)+6;
figure(619)
plot(TimeStep, Wave)
axis([-5 10 -5 15])
```

Ω

$$J\frac{d\Omega}{dt} = k i_a - B\Omega$$

$$\frac{d\Omega}{dt} = \frac{K}{J} i_a - \frac{B}{J}\Omega(t)$$

$$\frac{\Omega(t+\Delta t) - \Omega(t)}{\Delta t} = \frac{K}{J} i_a - \frac{B}{J}\Omega(t)$$

$$\Omega(t+\Delta t) - \Omega(t) = \Delta t\left[\frac{K}{J} i_a - \frac{B}{J}\Omega(t)\right]$$

$$\Omega(t+\Delta t) = \Delta t\left[\frac{K}{J} i_a(t) - \frac{B}{J}\Omega(t)\right] + \Omega(t)$$

I_a

$$V_a(t) = ir + L\frac{di}{dt} + k\Omega$$

$$L\frac{di}{dt} = V_a(t) - i_a(t)r - k\Omega(t)$$

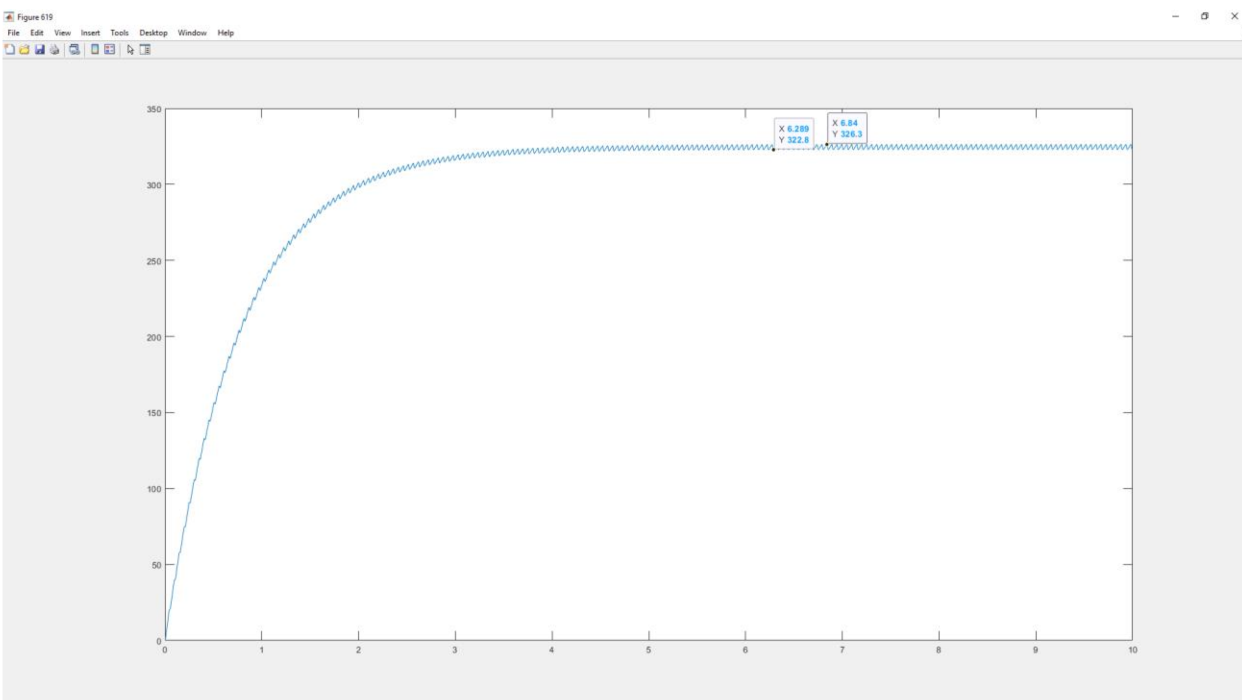
$$\frac{di}{dt} = \frac{V_a(t)}{L} - \frac{i_a(t)r}{L} - \frac{k\Omega(t)}{L}$$

$$\frac{i_a(t+\Delta t) - i_a(t)}{\Delta t} = \frac{V_a}{L} - \frac{i_a(t)r}{L} - \frac{k\Omega(t)}{L}$$

$$i_a(t+\Delta t) - i_a(t) = \Delta t\left[\frac{V_a}{L} - \frac{i_a(t)r}{L} - \frac{k}{L}\Omega(t)\right]$$

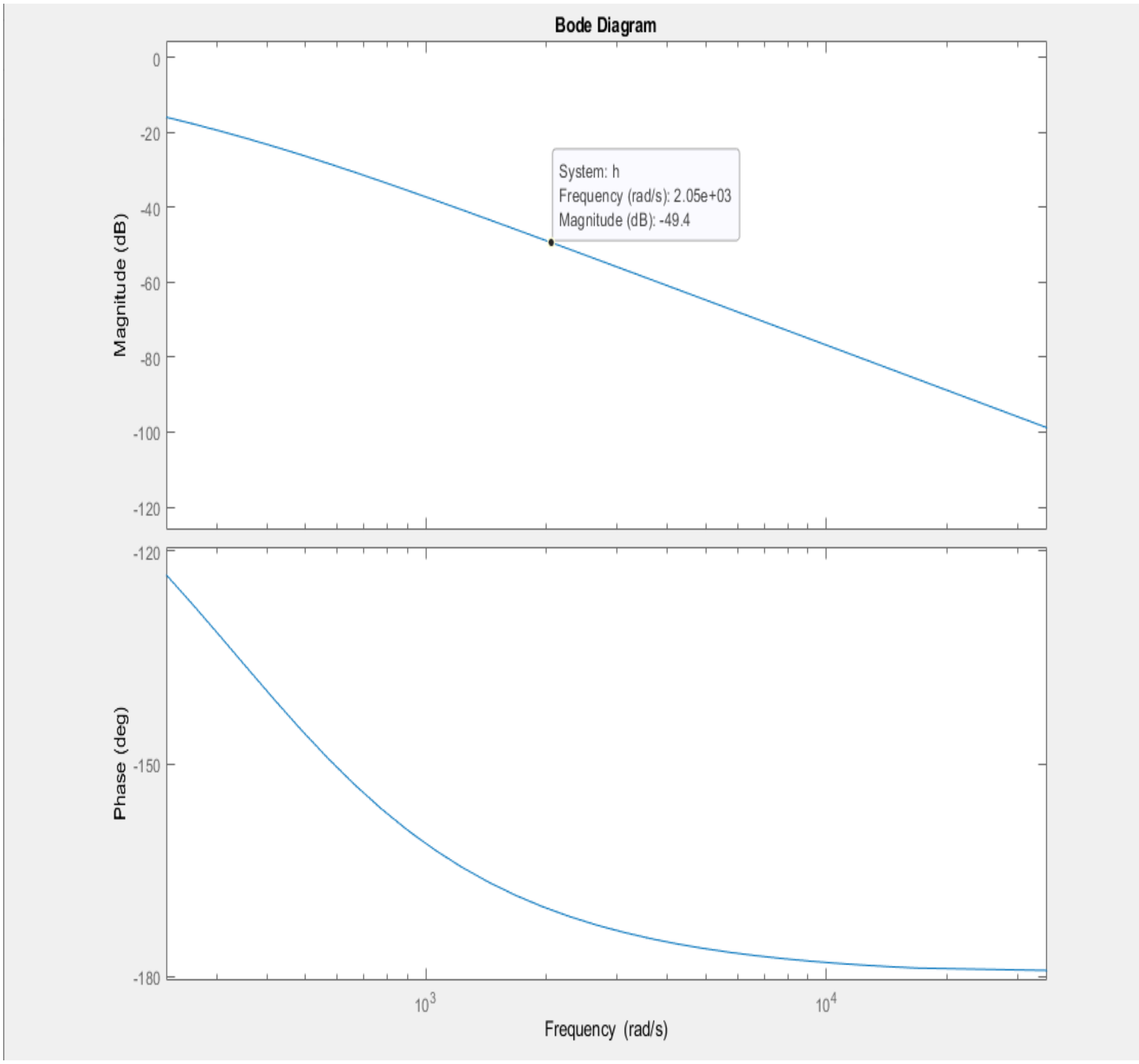
$$i_a(t+\Delta t) = \Delta t\left[\frac{V_a}{L} - \frac{i_a(t)r}{L} - \frac{k}{L}\Omega(t)\right] + i_a(t)$$

Above is the discretization of two differential equation to show the transient and steady state of the waveform. This was done with Andrew, Danil, Matrix and 10 others.



The PWM waveform runs for 10 seconds but reaches its steady state around 4 seconds. It has a trough of 322.8, a peak of 326.3, and am average of 324.55 rad/sec. Using the percent error formula, the results had a percent error of 0.170%. The first possible thing that could have caused this is how many decimal places I used in my calculations and rounding too much of my answers. The second reason could be an error in the code where I may have assigned a slightly off value, which could have caused Matlab to produce a slightly off value.

H: Select a switching frequency such that the amplitude of the speed at the switching frequency is attuned to 0.01% of the DC value.



The the switch frequency is at -80dB which is obtained by doing 20*log10(0.0001). In order to get the true value, we must subtract 30.6, since that were it starts, by -80dB. So, by looking at -49.4dB we see that the initial value will be 2050Hz for 0.01% of the DC value.

Problem 2 Code:

```
clear all
%Variables
L = 0.01;
R = 3.38;
K = 0.029;
```

```
J = 2e-4;
B = 0.5e-5;

%Problem 2A: In report
%Problem 2B
S = tf('s')
h = K/((S^2*L*J) + (S*(L*B + R*J)) + (R*B + K^2))
bode(h)

%Problem 2C
W = logspace(-2,4 , 10000);
H = K./((j*W.^2*L*J)+((j*W)*(L*B*R*J))+(R*B+K^2))
semilogx(W, 20*log10(abs(H)));
title('semilogx')

%Problem 2D
Switch = 120;
Freq = Switch/(2*pi);
T = 1/Freq
TimeStep = (0:T/100000:T);


%Problem 2E
Voltage = 324 * ((R*B + K^2)/K);
D = Voltage/12;

%Problem 2F
Df = 0.5;
a0 = (12*Df *T)/T;
w = 2*pi/T;
Alpha1 = (12/(j*w*T)) + ((-12/(j*w*T)).* exp(-j*w* 0.5 *T))
Angle1 = angle(Alpha1)*180/pi
Mag = 2*Alpha1


%Problem 2G
Wave = 6*square(TimeStep* w, 79.87)+6;
figure(619)
plot(TimeStep, Wave)
axis([-5 10 -5 15])


%Variables 2: electric boogaloo
Omega = 122;
tau = J/K;
Delta=T/10000;
a = tau/Delta;
b = 0.5*10^-5
kb = 5.8000e-07;

%time step%
TimeStep2 = [0:Delta:10];
omega = zeros(size(TimeStep2));
I = zeros(size(TimeStep2));
Y0 = 7.693*cos(Omega*TimeStep2-pi/2);
```

```
WaveofSquare= 6*square(TimeStep2*Omega,80)+6;


%Discretization
for n = 1:1:length(TimeStep2)-1

        I(n+1)=Delta*((WaveofSquare(n)/L)-(I(n)*R/L)-(K*omega(n)/L))+I(n);
        omega(n+1)=Delta*((K/J*I(n))-(b/J*omega(n)))+omega(n);
end

plot(TimeStep2,omega)

%Problem 2H.)
switch_H = 2050 %Hz
```