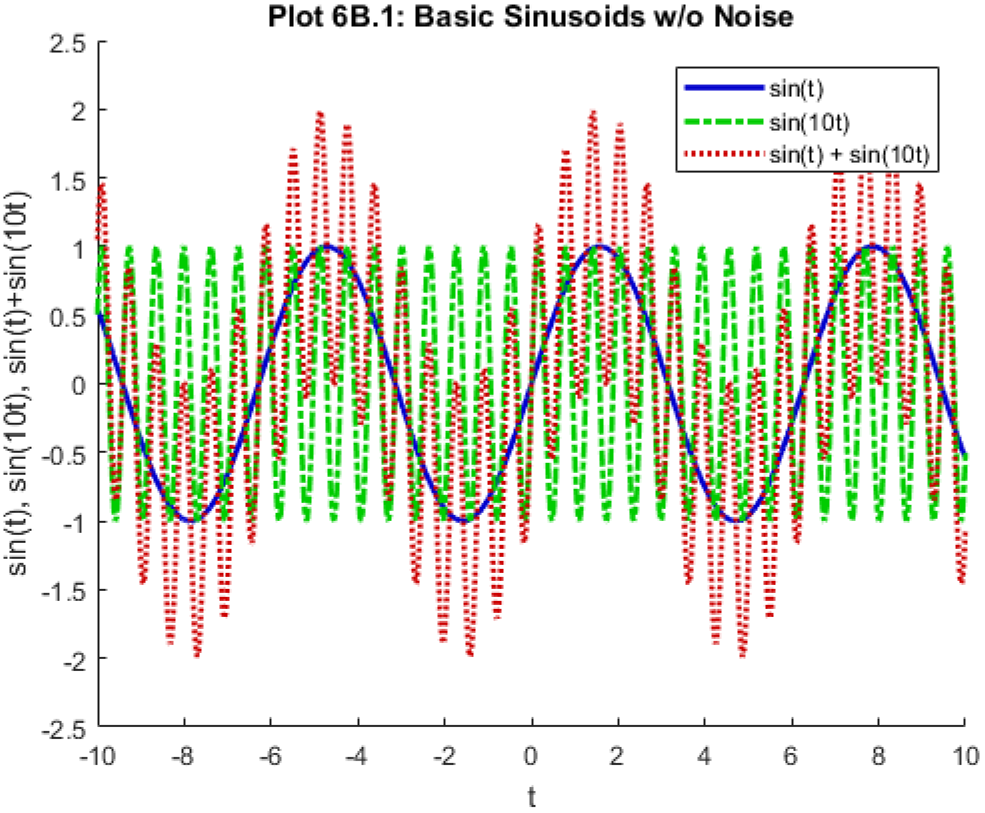
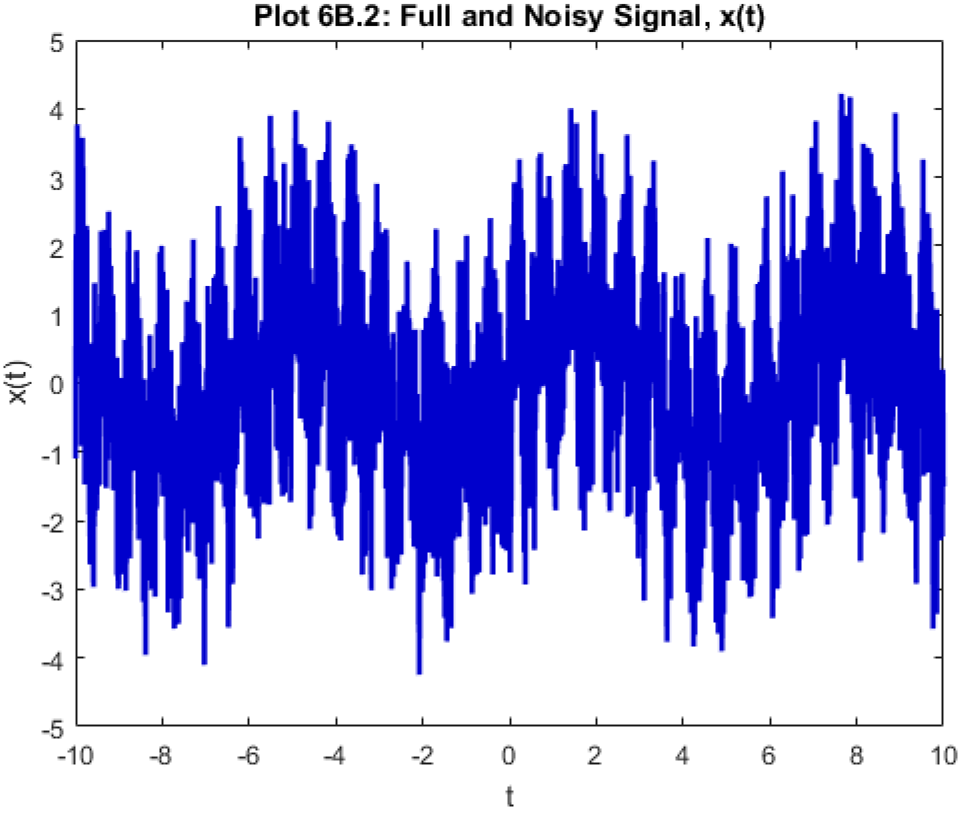
```
% Christopher Brant
% C19816588
% MATLAB Homework 6A Due on 11/29/17

clear; clc; close all;

% This assignment uses the signal x(t) = sin(t) + sin(10t) + N(t)
% N(t) represents random noise.
% t is the time range between -10 and 10
t = -10:0.005:10;

% Plot sin(t), sin(10t), and sin(t)+sin(10t) on the same plot
origin = [0, 0];        % origin values used for plotting
x_lims = [-10, 10];     % x-axis limits
y_lims = [-2.5, 2.5];     % y-axis limits
% Create new graph window
figure();
hold on;
% Plot the different signals
SIN = plot(t, sin(t), 'LineStyle', '-', 'Color',...
    [0,0,.8], 'LineWidth', 2);
SIN10 = plot(t, sin(10*t), 'LineStyle', '-.', 'Color',...
    [0,.8,0], 'LineWidth', 2);
SIN_SIN10 = plot(t, sin(t)+sin(10*t), 'LineStyle', ':', 'Color',...
    [.8,0,0], 'LineWidth', 2);
% Adding labels and axis values to the plot
axis(horzcat(x_lims, y_lims));
title('Plot 6B.1: Basic Sinusoids w/o Noise');
xlabel('t');
ylabel('sin(t), sin(10t), sin(t)+sin(10t)');
legend([SIN, SIN10, SIN_SIN10], 'sin(t)', 'sin(10t)',...
    'sin(t) + sin(10t)', 'Location', 'Northeast');

% x_t will denote the full signal in this case
x_t = sin(t) + sin(10*t) + randn(size(t));

% Plot x(t) on next plot
origin = [0, 0];        % origin values used for plotting
x_lims = [-10, 10];     % x-axis limits
y_lims = [-5, 5];       % y-axis limits
% Create new graph window
figure();
% Plot the signal
plot(t, x_t, 'LineStyle', '-', 'Color',...
    [0,0,.8], 'LineWidth', 2);
% Adding labels and axis values to the plot
axis(horzcat(x_lims, y_lims));
title('Plot 6B.2: Full and Noisy Signal, x(t)');
xlabel('t');
ylabel('x(t)');

% Design an 8th Order Low Pass filter to produce sin(t)+sin(10t)
% I will use Chebyshev Type 1
[z,p,k] = cheby1(8,1,12,'low','s');
% Setting cheby1 to be the desired filter here
sys = zpk(z,p,k);
% y_t will denote the output of the filtered response
y_t = lsim(sys,x_t,t);

% Plotting y_t and sin(t)+sin(10t) on the same graph
origin = [0, 0];        % origin values used for plotting
x_lims = [-10, 10];     % x-axis limits
y_lims = [-2.5, 2.5];   % y-axis limits
% Create new graph window
figure();
hold on;
% Plot the different signals
Y_T = plot(t, y_t, 'LineStyle', '-', 'Color',...
    [0,0,.8], 'LineWidth', 2);
SIN_SIN10 = plot(t, sin(t)+sin(10*t), 'LineStyle', ':', 'Color',...
    [0,.8,0], 'LineWidth', 2);
% Adding labels and axis values to the plot
axis(horzcat(x_lims, y_lims));
title('Plot 6B.3: Low Pass Filter Resulting in sin(t)+sin(10t)');
xlabel('t');
ylabel('sin(t), sin(10t), sin(t)+sin(10t)');
legend([Y_T, SIN_SIN10], 'y(t)',...
    'sin(t) + sin(10t)', 'Location', 'Northeast');

% Plot the system on a Bode plot as well
figure();
bode(sys,{10^-1,10^3});
title('Plot 6B.4: Bode Diagrams for the Low Pass Filter for
 sin(t)+sin(10t)');

% Design an 8th Order Low Pass filter to produce only sin(t)
% Again will use Chebyshev Type 1
[z,p,k] = cheby1(8,1,2,'low','s');
% Setting cheby1 to the desired filter here
sys = zpk(z,p,k);
% y_t will denote the output of the filtered response
y_t = lsim(sys,x_t,t);

% Plotting y_t and sin(t)+sin(10t) on the same graph
origin = [0, 0];        % origin values used for plotting
x_lims = [-10, 10];     % x-axis limits
y_lims = [-2.5, 2.5];     % y-axis limits
% Create new graph window
figure();
hold on;
% Plot the different signals
Y_T = plot(t, y_t, 'LineStyle', '-', 'Color',...
    [0,0,.8], 'LineWidth', 2);
SIN_SIN10 = plot(t, sin(t), 'LineStyle', ':', 'Color',...
    [0,.8,0], 'LineWidth', 2);
% Adding labels and axis values to the plot
```

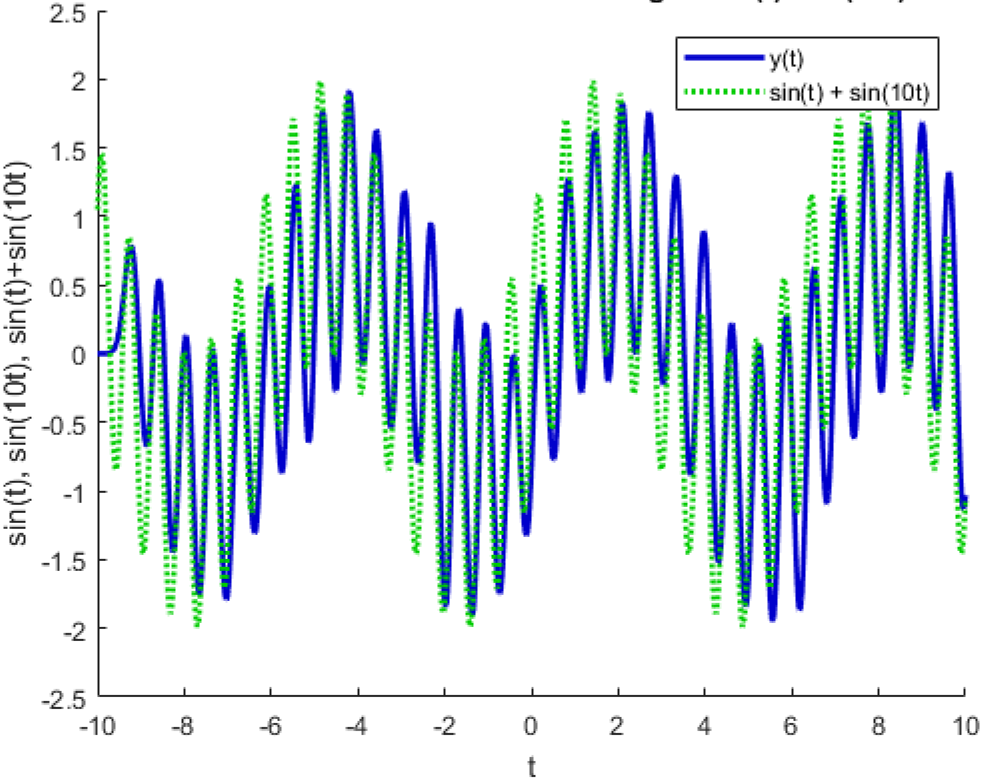
```
axis(horzcat(x_lims, y_lims));
title('Plot 6B.5: Low Pass Filter Resulting in sin(t)');
xlabel('t');
ylabel('sin(t), sin(10t), sin(t)+sin(10t)');
legend([Y_T, SIN_SIN10], 'y(t)',...
    'sin(t)', 'Location', 'Northeast');

% Plot the system on a Bode plot as well
figure();
bode(sys,{10^-1,10^3});
title('Plot 6B.6: Bode Diagrams for the Low Pass Filter for sin(t)');

% Design an 8th Order Band Pass filter to extract sin(10t)
% Again will use Chebyshev Type 1
[z,p,k] = cheby1(8,1,[8,14],'bandpass','s');
% Setting cheby1 to the desired filter here
sys = zpk(z,p,k);
% y_t will denote the output of the filtered response
y_t = lsim(sys,x_t,t);

% Plotting y_t and sin(t)+sin(10t) on the same graph
origin = [0, 0];        % origin values used for plotting
x_lims = [-10, 10];     % x-axis limits
y_lims = [-2.5, 2.5];     % y-axis limits
% Create new graph window
figure();
hold on;
% Plot the different signals
Y_T = plot(t, y_t, 'LineStyle', '-', 'Color',...
    [0,0,.8], 'LineWidth', 2);
SIN_SIN10 = plot(t, sin(10*t), 'LineStyle', ':', 'Color',...
    [0,.8,0], 'LineWidth', 2);
% Adding labels and axis values to the plot
axis(horzcat(x_lims, y_lims));
title('Plot 6B.7: Band Pass Filter Resulting in sin(10t)');
xlabel('t');
ylabel('sin(t), sin(10t), sin(t)+sin(10t)');
legend([Y_T, SIN_SIN10], 'y(t)',...
    'sin(t)', 'Location', 'Northeast');

% Plot the system on a Bode plot as well
figure();
bode(sys,{10^-1,10^3});
title('Plot 6B.8: Bode Diagrams for the Band Pass Filter for
 sin(10t)');
```

```
Warning: Simulation will start at a nonzero initial time.
Warning: Simulation will start at a nonzero initial time.
Warning: Simulation will start at a nonzero initial time.
```

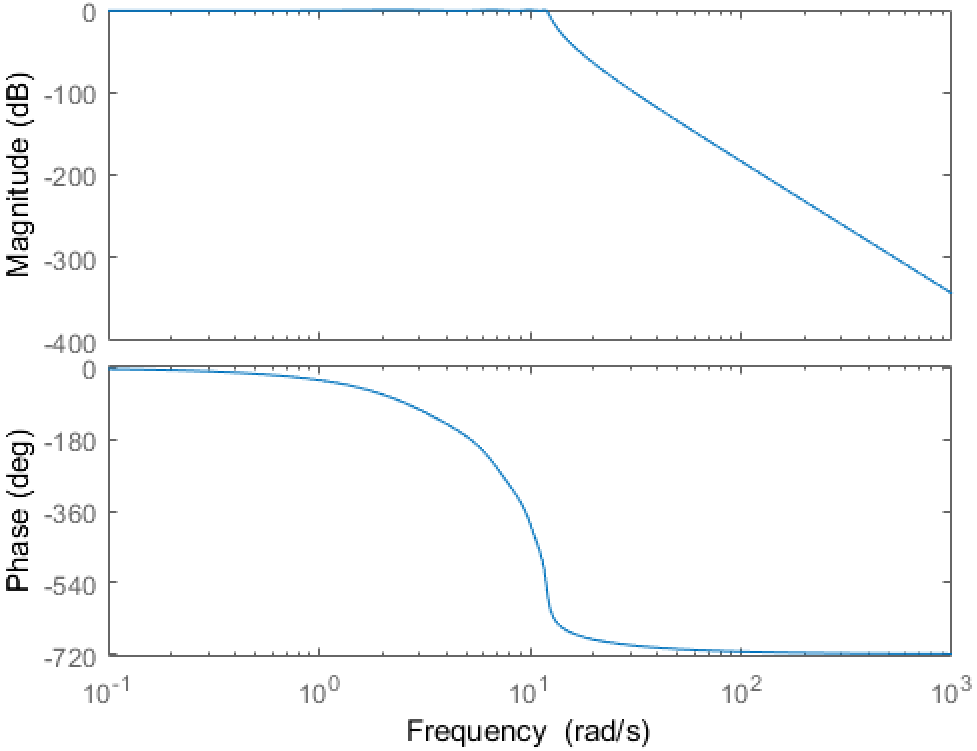

Plot 6B.1: Basic Sinusoids w/o Noise
sin(t)
sin(10t)
sin(t) + sin(10t)
sin(t), sin(10t), sin(t)+sin(10t)
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
-10
-8
-6
-4
-2
0
2
4
6
8
10
t

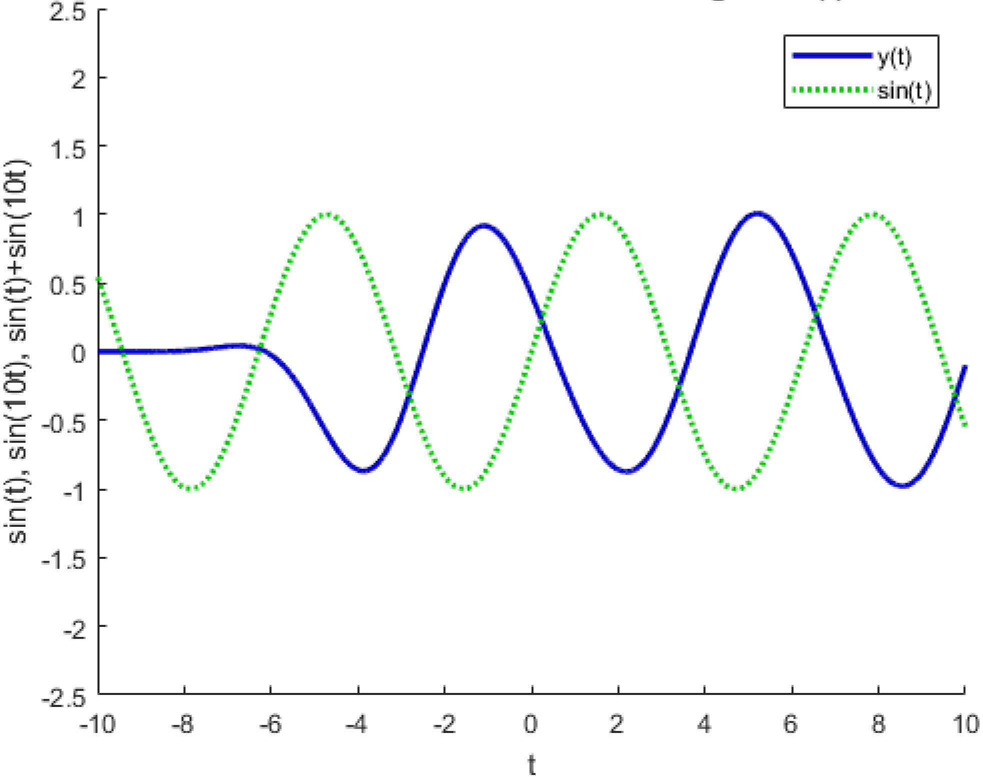

Plot 6B.2: Full and Noisy Signal, x(t)
x(t)
5
4
3
2
1
0
-1
-2
-3
-4
-5
-10
-8
-6
-4
-2
0
2
4
6
8
10
t

Plot 6B.3: Low Pass Filter Resulting in sin(t)+sin(10t)

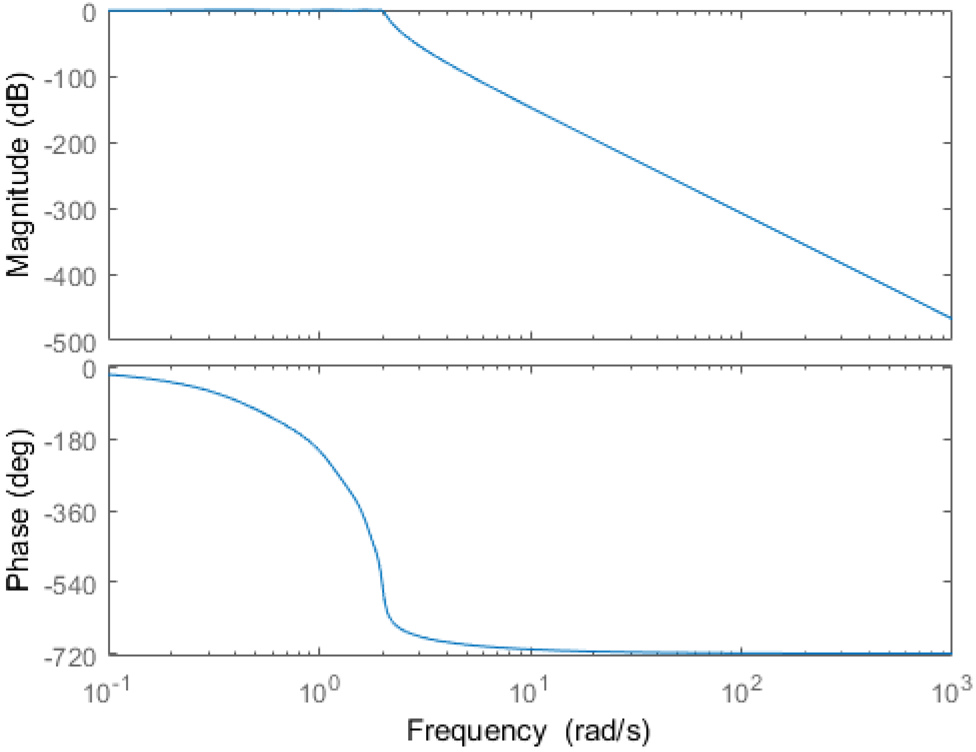


Plot 6B.4: Bode Diagrams for the Low Pass Filter for sin(t)+sin(10t)



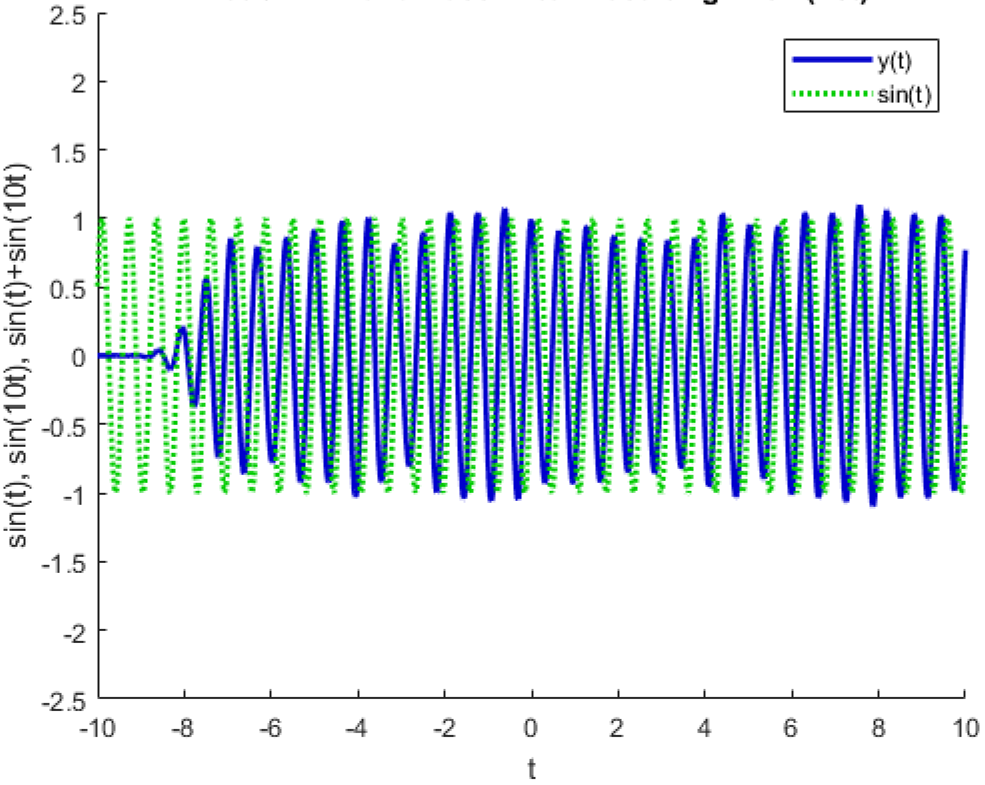


Plot 6B.5: Low Pass Filter Resulting in sin(t)

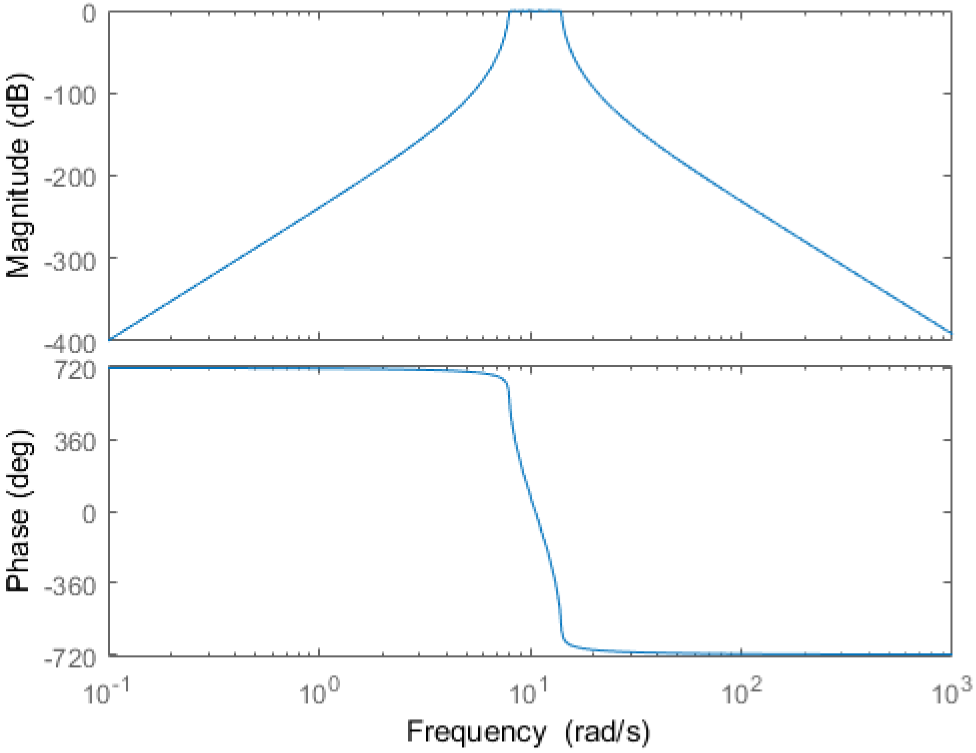


Plot 6B.6: Bode Diagrams for the Low Pass Filter for sin(t)

**Plot 6B.7: Band Pass Filter Resulting in sin(10t)**



**Plot 6B.8: Bode Diagrams for the Band Pass Filter for sin(10t)**



*Published with MATLAB® R2017a*