**Final Exam – Simulation Results**

ECEn 483/ ME 431

Fall 2014

Name:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

At the end of the exam, print this file and staple it to the handout portion of the exam.

*Make sure your plots are large enough to be easily legible. Plots should be at least half the page width wide.*

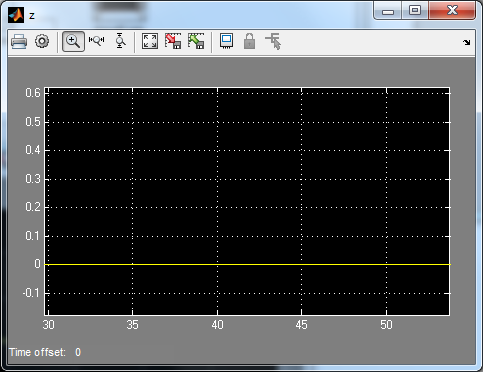
# Part 1. Equations of Motion – Simulation Model

Nothing to do for this part.

# Part 2. Design models

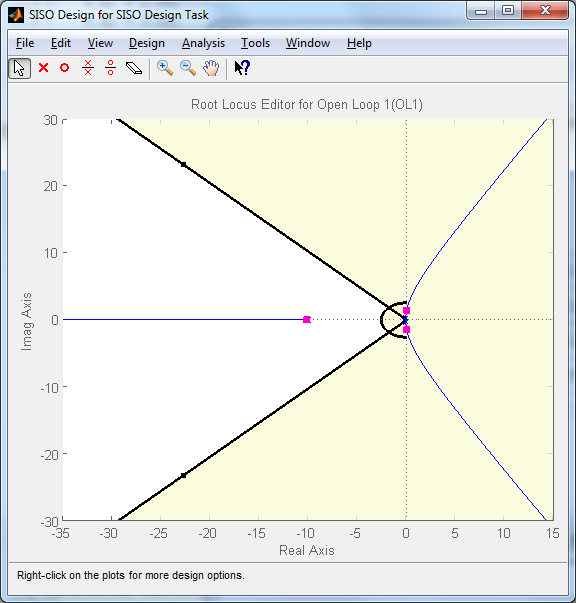
2.2 Insert plot of the output of the simulation model with initial condition *z*(0) = *ze* and input *Ve* directly below this line.

Ve = -3.4684

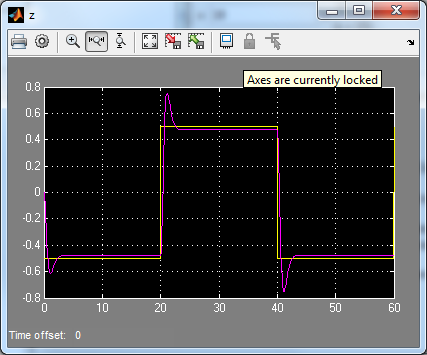


# Part 3. Root Locus Design

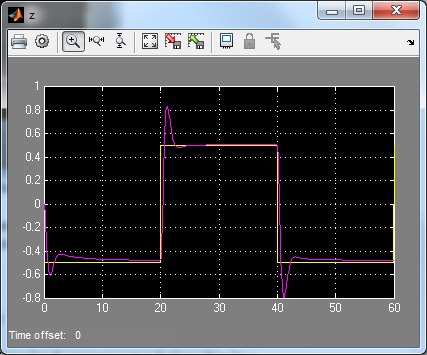
3.1 Insert a plot of the root locus for the open loop plant alone showing the acceptable regions of the s-plane to achieve the desired response below this line.



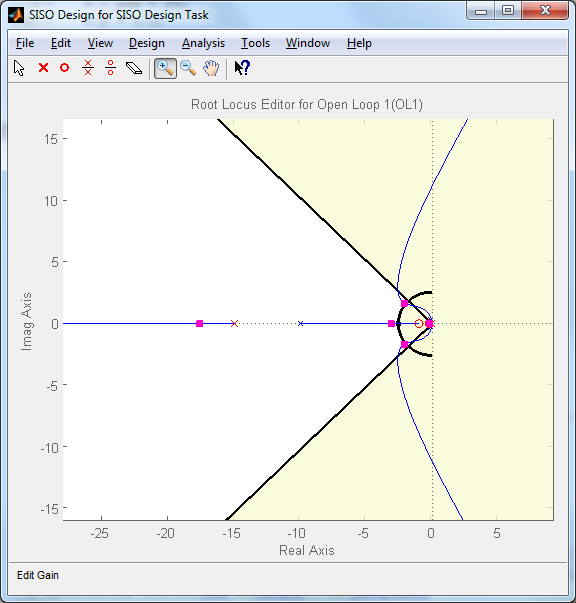
3.3 Insert a plot showing simulation results for a lead compensator below this line.



3.5 Insert a plot showing simulation results for a lead-lag compensator below this line.

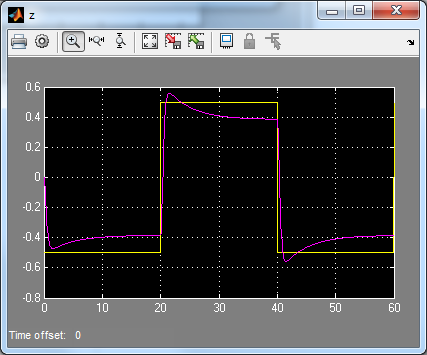


3.6 Insert a plot of your root locus lead-lag design showing the closed-loop pole locations and the acceptable regions of the s-plane directly below this line.

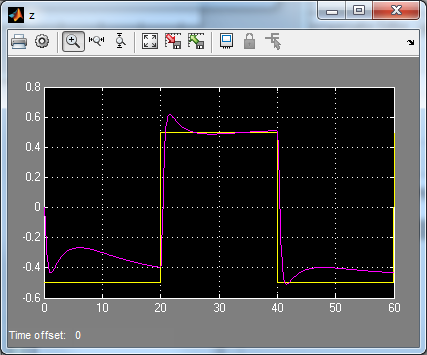


# Part 4. Frequency Response Design

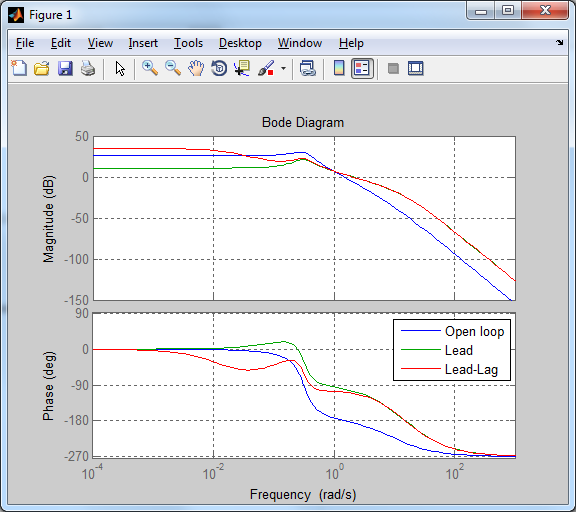
4.3 Insert a plot showing simulation results for a lead compensator below this line.



4.5 Insert a plot showing simulation results for a lead-lag compensator below this line.

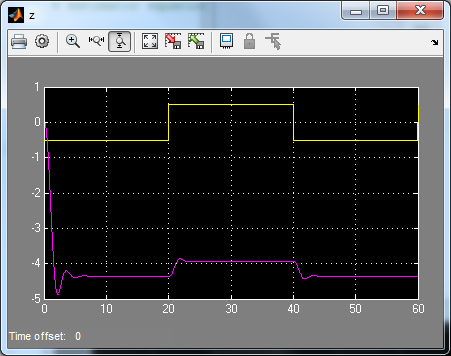


4.6 Insert the Bode plots for the original plant, the lead-controlled plant, and the lead-lag controlled plant below this line.

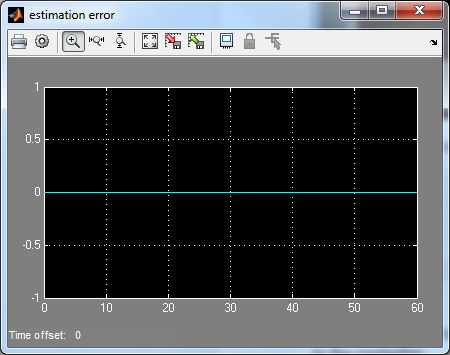


# Part 5. State-space Design

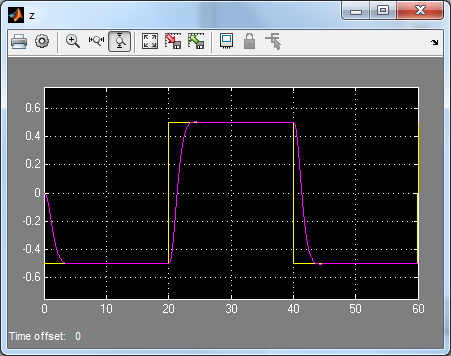
5.5. Insert a plot of the step response of the system when the true state is being used.



5.6 Insert a plot of the state estimation error when the true state is being used.



5.7 Insert a plot of the step response of the system when the estimated state is being used in the controller.



5.8 Insert a copy of ctrl\_est.m.

|  |
| --- |
| function out=ctrl\_est(in,P)  z\_c = in(1);  z = in(2);  t = in(3);  x = in(4:6);    % equilibrium state, input, output  F\_e = P.k1\*z\_c + P.k2\*(z\_c^3)-P.m\*P.g\*sin(45\*pi/180);  V\_e = F\_e;  x\_e = [z\_c; 0; 0];    % estimator  persistent xhat\_  persistent V    % estimator equations go here...    N = 10;    if t<P.Ts,  xhat\_ = [0;0;0];  V = 0;  else  for i = 1:N,  xhat\_ = xhat\_ + P.Ts/N\*(P.F\*(xhat\_-x\_e) + P.G\*(V-V\_e) + P.LL\*(z-P.H\*xhat\_));  end  end    xhat = xhat\_; %use estimated state  % xhat = x; %use true state    V = -P.K\*(xhat-x\_e)+V\_e;    out = [V;xhat];    end    function out = sat(in,limit)  if in > limit, out = limit;  elseif in < -limit, out = -limit;  else out = in;  end  end |

# 6. Insert your param.m or other design m-files here.

|  |
| --- |
| clear all;    % system parameters  P.g = 9.81;  P.theta = 45\*pi/180;  P.m = 0.5;  P.k1 = 0.05;  P.k2 = 0.02;  P.V\_max = 25;  P.b = 0.1;  P.tau\_a = 0.1;    % sample time for controller  P.Ts = 0.01;    % steady-state equilibrium force, voltage for z\_e = 0;  P.z\_e = 0;  P.F\_e = P.k1\*P.z\_e + P.k2\*(P.z\_e^3)-P.m\*P.g\*sin(45\*pi/180); % Put expressions for the calculated equilibrium values  P.V\_e = P.F\_e; % of the force and voltage here.    % initial conditions corresponding to equilibrium  P.z0 = P.z\_e;  P.zdot0 = 0;  P.F0 = P.F\_e;    % Calculations for control designs    %==============================  % ROOT LOCUS DESIGN  %==============================  tm = P.tau\_a\*P.m;  num = 1/tm;  den = [1 (P.tau\_a\*P.b+P.m)/tm (P.tau\_a\*P.k1+P.b)/tm P.k1/tm];    S = tf('s');  H = tf(num, den);    % rltool(H)      %==============================  % FREQUENCY RESPONSE DESIGN  %==============================    w\_c = 2;    phase\_desired = 70;  phi\_max = 80;  phi\_max\_rad = phi\_max\*pi/180;    alpha\_lead = (1-sin(phi\_max\_rad))/(1+sin(phi\_max\_rad));  z\_lead = w\_c\*sqrt(alpha\_lead);  p\_lead = w\_c/sqrt(alpha\_lead);    gain = 22.9;  D\_lead = tf(gain\*[1 z\_lead], [1 p\_lead]);    alpha\_lag = 15;  z\_lag = w\_c/10;  p\_lag = z\_lag/alpha\_lag;  D\_lag = tf([1 z\_lag], [1 p\_lag]);    % sisotool(H)    % hold off;  % hold on;  % bode(H)  % bode(H\*D\_lead)  % bode(H\*D\_lead\*D\_lag)  % legend('Open loop', 'Lead', 'Lead-Lag')  % grid on      %==============================  % FREQUENCY RESPONSE DESIGN  %==============================    P.F = [0 1 0; -P.k1/P.m -P.b/P.m 1/P.m; 0 0 -1/P.tau\_a];  P.G = [0; 0; 1/P.tau\_a];  P.H = [1 0 0];  P.J = 0;    pp = [-1.8 -1.5+j -1.5-j];  P.K = place(P.F,P.G,pp);    NN = inv([P.F P.G; P.H P.J])\*[0; 0; 0; 1];  P.Nx = NN(1:3);  P.Nu = NN(4);  P.Nbar = P.Nu + P.K\*P.Nx;    pe = 5\*pp;  P.LL = place(P.F', P.H', pe)'; |