
```

%Question 1 & Question 2
%Angular Velocity vs Voltage
clc; close all; clear;
angVelocityTF = tf([50],[0.13, 1]);
figure()
step(angVelocityTF);
title('Angular Velocity Step Response');
fprintf('Angular Velocity Transfer Function | ');
fprintf('Stable System per BIBO definition\n');
%Angular Position vs Voltage
angPositionTF = tf([50], [0.13,1,0]);
figure()
step(angPositionTF);
title('Angular Position Step Response');
fprintf('Angular Position Transfer Function | ');
fprintf('Unstable System per BIBO definition\n');

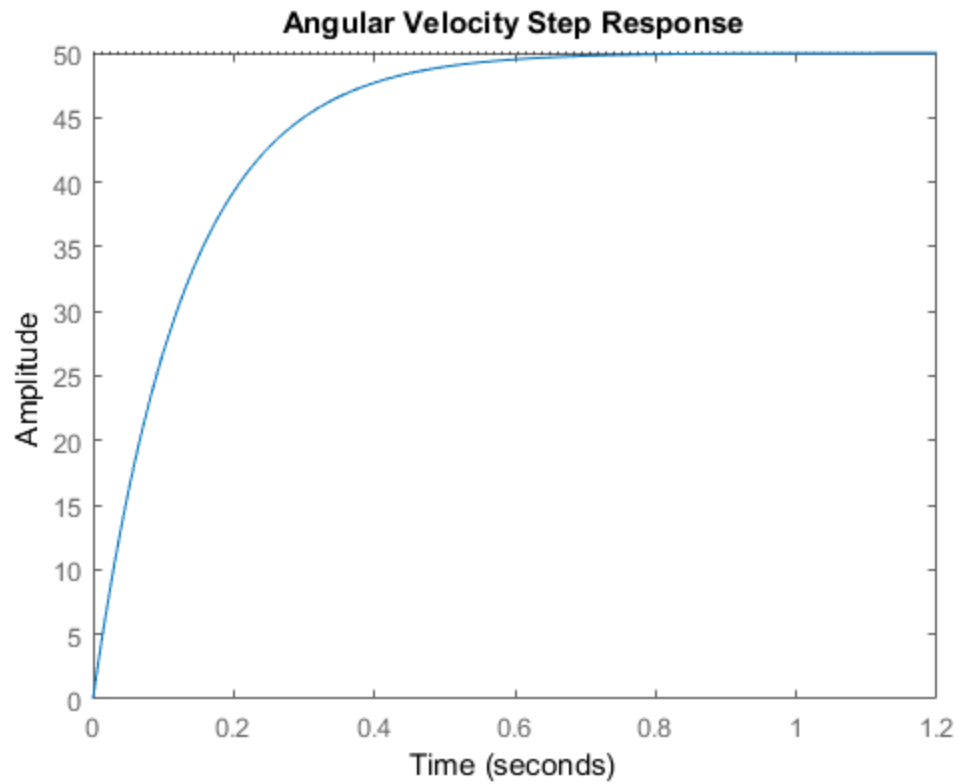
%Question 3 & Question 4
[z1,p1] = zpkdata(angVelocityTF);
z1 = cell2mat(z1);
p1 = cell2mat(p1);
fprintf('Angular Velocity\n');
fprintf('Zeros | %g\n',z1);
fprintf('Poles | %g\n',p1);
fprintf('Stable System\n');
[z2,p2] = zpkdata(angPositionTF);
z2 = cell2mat(z2);
p2 = cell2mat(p2);
fprintf('Angular Position\n');
fprintf('Zeros | %g\n',z2);
fprintf('Poles | %g | %g\n',p2(1),p2(2));
fprintf('Marginally Stable System\n');
figure()
pzmap(angVelocityTF);
title('Angular Velocity Poles and Zeros')
figure()
pzmap(angPositionTF);
title('Angular Position Poles and Zeros');

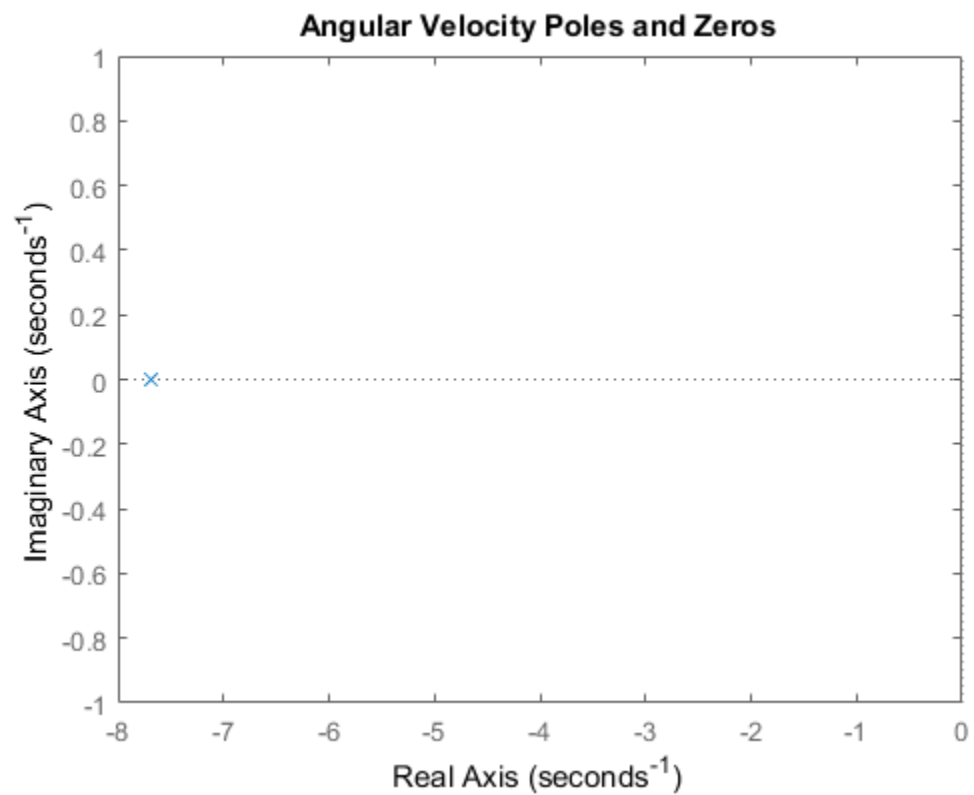
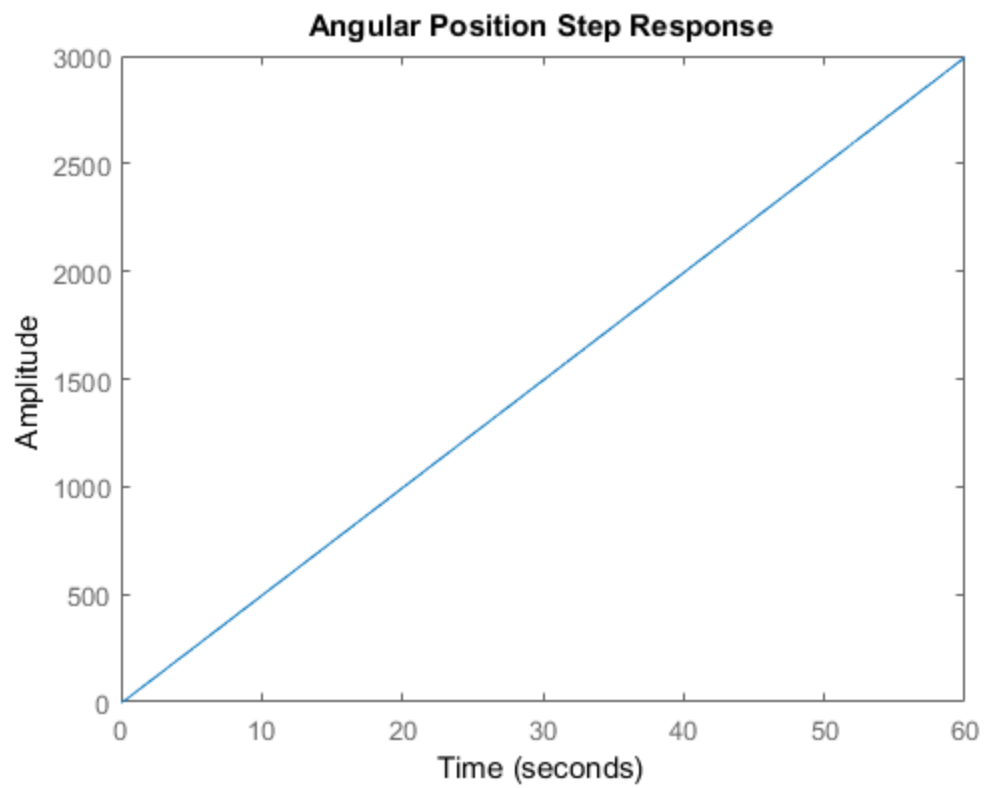
%Question 8
K = 63.881;
tau = 0.096;
closedLoopSystem = tf([K/tau],[1,(1/tau), K/tau]);
figure()
step(closedLoopSystem)
title('Closed Loop Transfer Function Step Response');
fprintf('Closed Loop Transfer Function | ');
fprintf('Stable System per BIBO definition\n');

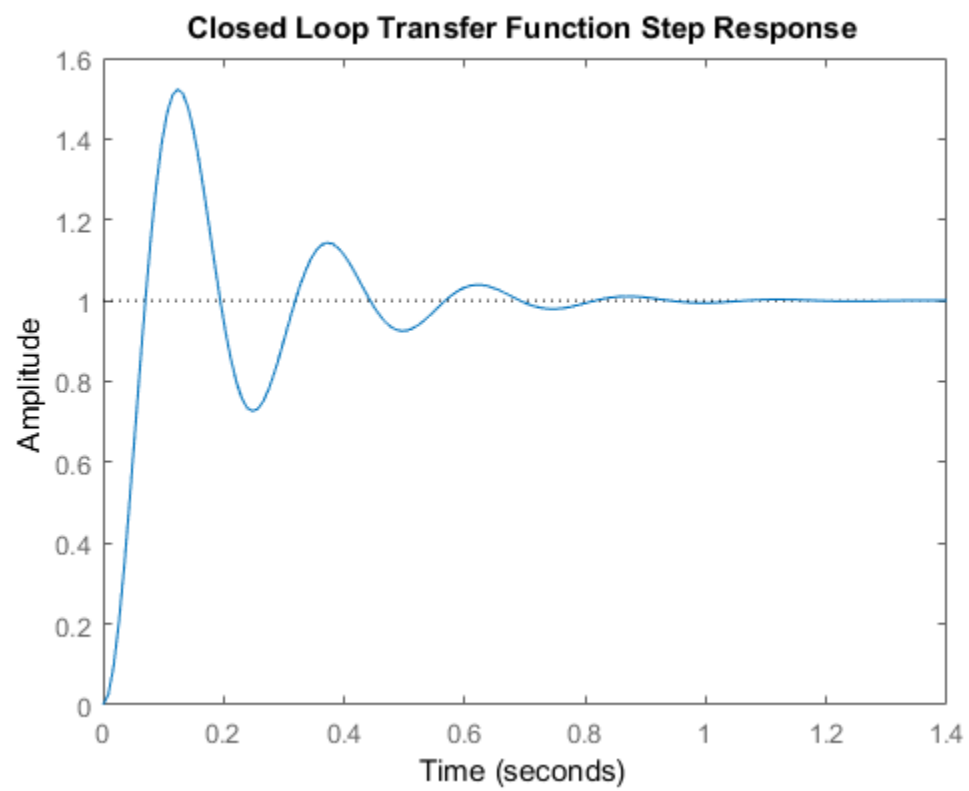
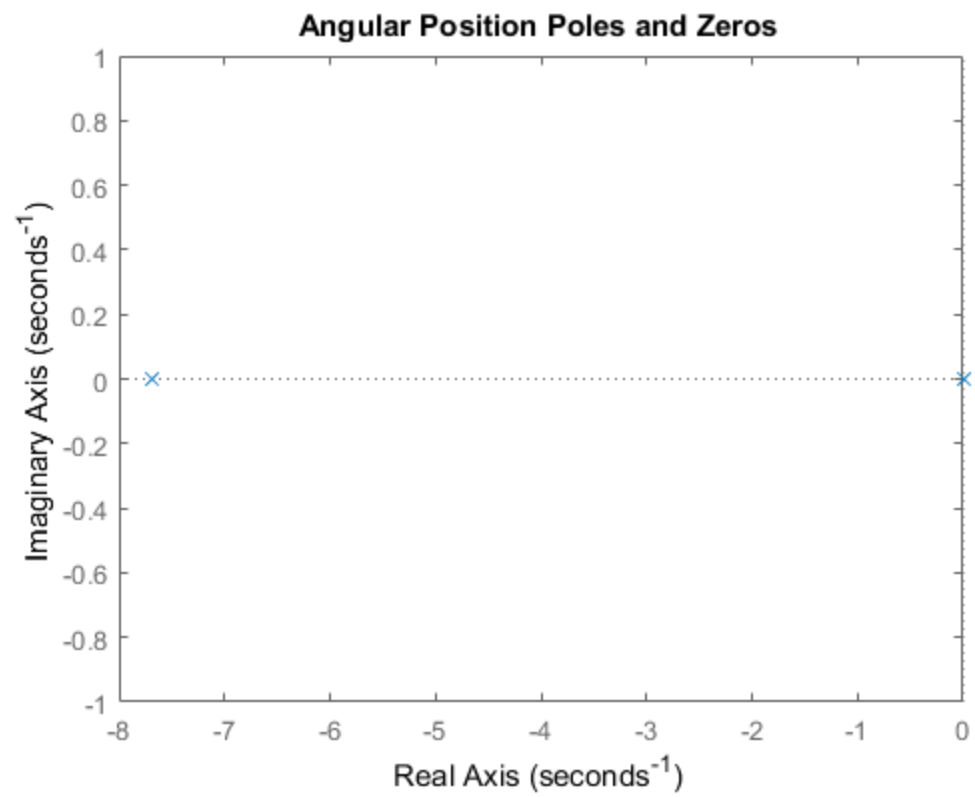
Angular Velocity Transfer Function | Stable System per BIBO definition
Angular Position Transfer Function | Unstable System per BIBO
definition

```

Angular Velocity
Zeros /
Poles / -7.69231
Stable System
Angular Position
Zeros /
Poles / 0 / -7.69231
Marginally Stable System
Closed Loop Transfer Function / Stable System per BIBO definition







Published with MATLAB® R2017a