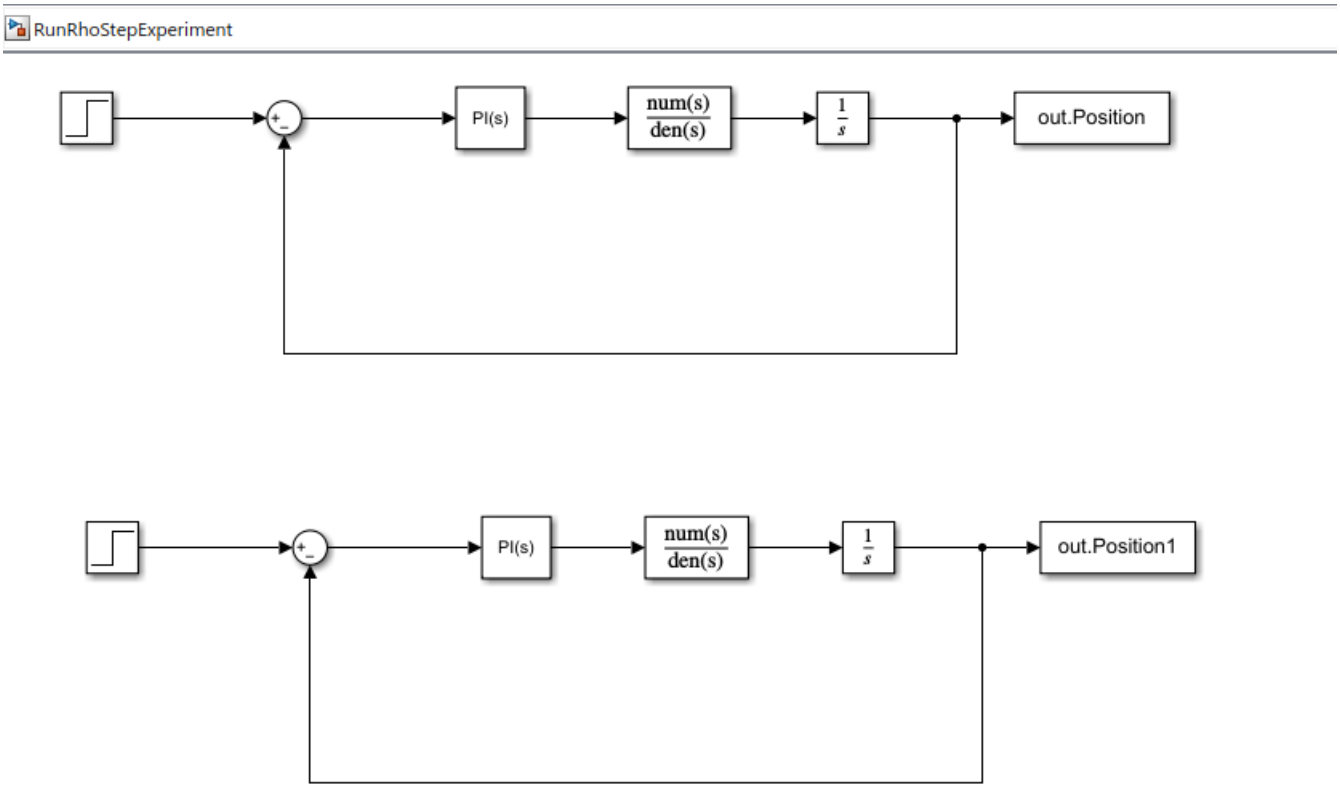
Contents

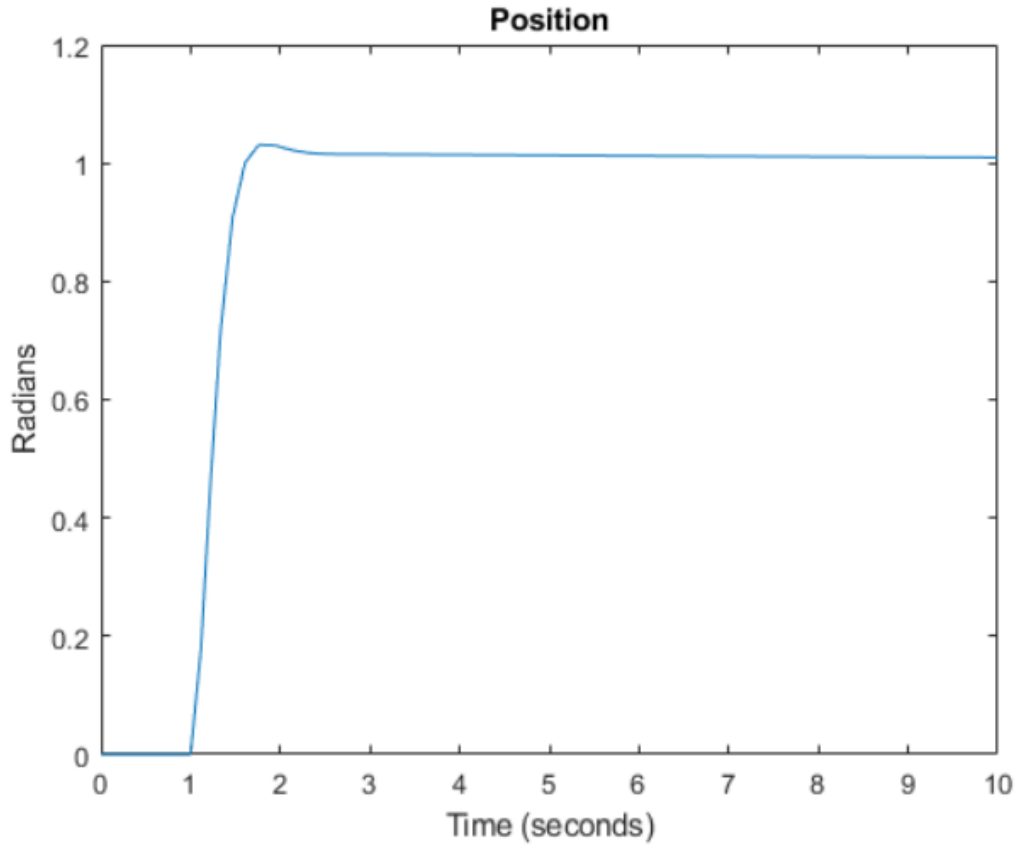

```
%%Step Experiment to find G_rho(s) for both motors
% Group Team 9
%
% This script shows results from both the experimental and simulated open
% loop step response experiments for Demo 1
%
% Required File: RunRhoStepExperiment.slx
%
```

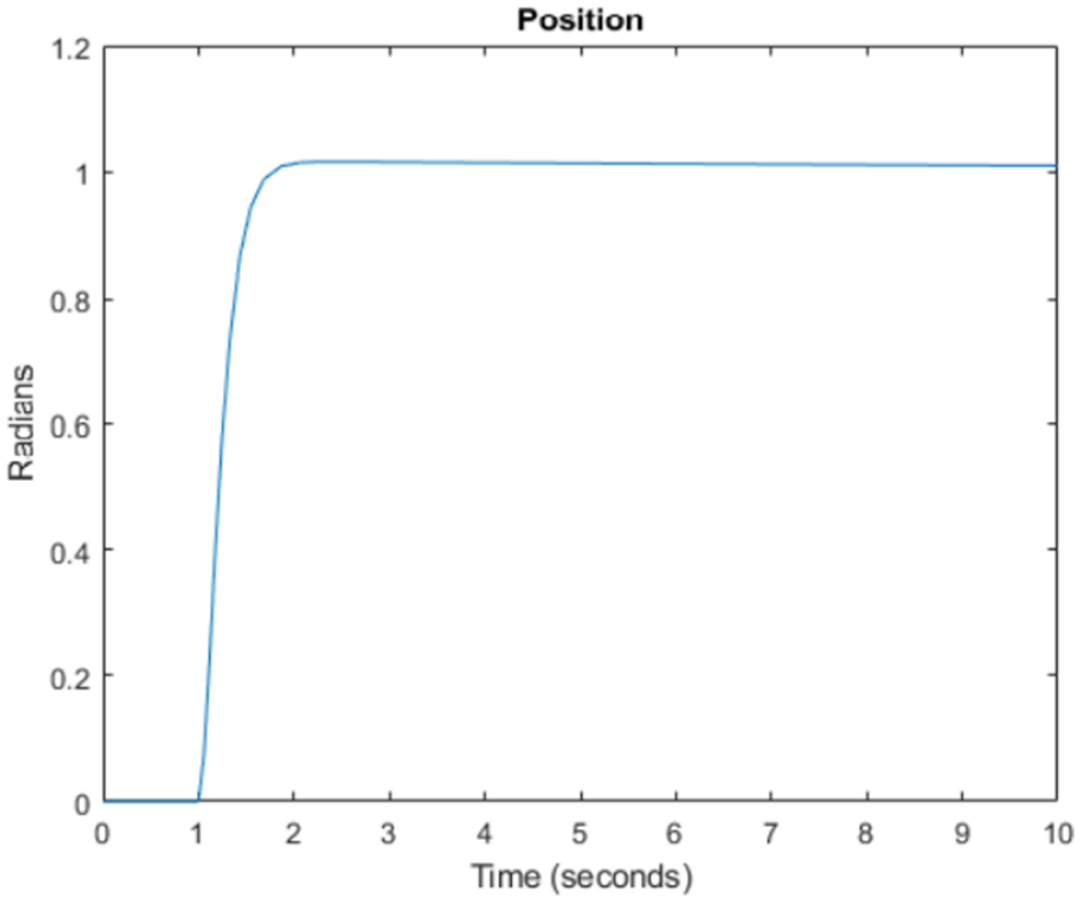
Loop step response

This simulation incorporates a PI controller. This controller can be tuned in Simulink to ensure Rise times of less than 1 seconds that have less than 10% overshoot.

```
% Open Closed Loop step response simulation
open_system('RunRhoStepExperiment')
%
% Run the simulation
%
out=sim('RunRhoStepExperiment');

% Plot the Results for both Wheels

% Right Wheel
plot(out.position1)
title('Position');
% Left Wheel
plot(out.position)
title('Position');
```




Position
Radians
Time (seconds)
0
0.2
0.4
0.6
0.8
1
1.2
0
1
2
3
4
5
6
7
8
9
10


Position
Radians
Time (seconds)
0
0.2
0.4
0.6
0.8
1
1.2
0
1
2
3
4
5
6
7
8
9
10

Controll Simulation for Theta Desired (Outer Loop)

Group Team 9

This script runs a simulation of the robot (Both wheels) with a PI controller. This controller has been tuned to have a rise time of under 1 second and to have less than 10% overshoot

Required File: RunFinalDemo1ControlCode.slx

Contents

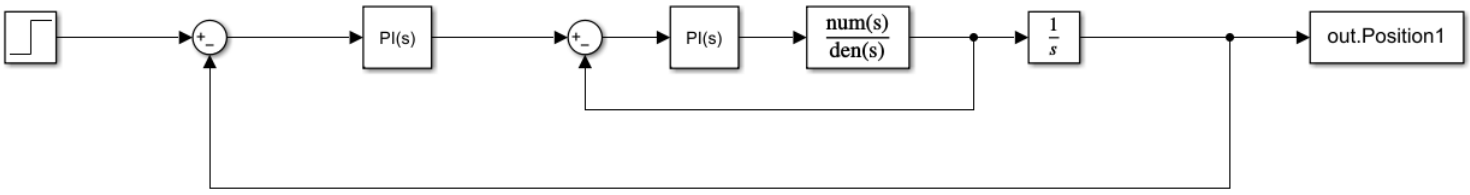
Define Motor Paramters

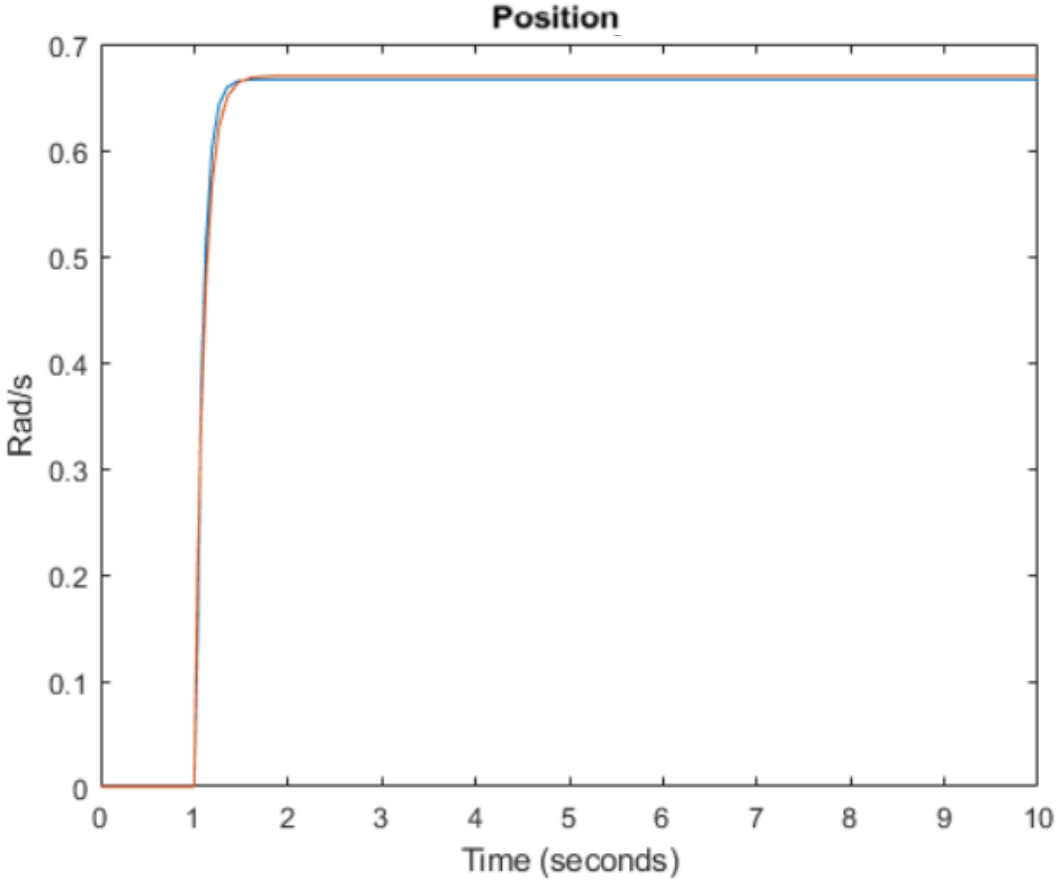
```
Ra=1;  % armaature resistance [Ohms]
Kt=.5; % motor torque constant [Nm/A]
Ke=.5; % back emf constant [Vs/rad]
% J=.05; % Load inertia [Nm^2]          NOT USED - From Assignmnet 2
% b=.5;  % damping [Nm/s]               NOT USED - From Assignmnet 2

% Manual Tuning Paramaters
k = 0.39;
sigma = 3.3;
```

Run a simulation

This simulation applies a PI controller to the transfer function found in RunFeedbackControl and the PI can be tuned to lower the overshoot

Open the block diagram so it appears in the documentation when published. Make sure the block diagram is closed before running the publish function




Position
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
Rad/s
0
1
2
3
4
5
6
7
8
9
10
Time (seconds)