**DESIGNING A PID CONTROLLER FOR DC MOTOR POSITION**

**(IN CONTINUOUS AND DISCRETE DOMAIN)**

%% pid controller (continuous and discrete) for motor position

J = 3.2284E-6; %moment of inertia

b = 3.5077E-6; %motor viscous friction constant

K = 0.0274; %Gain

R = 4; %electric resistance

L = 2.75E-6; %electric inductance

s = tf('s'); %define transfer function

p\_motor = K/(s\*((J\*s+b)\*(L\*s+R)+K\*K)) %define transfer function

% Using PID Tuner we get values:

% Kp=3.80979871547419, Ki=7.13510216852499, Kd=0.0473982065747193 Filter

% Coeff=206.117074617444

Kp=3.80979871547419;

Ki=7.13510216852499;

Kd=0.0473982065747193;

k\_pid=pid(Kp,Ki,Kd);

cls\_loop\_pid= feedback(p\_motor\*k\_pid,1);

%%

%figure();

%step(cls\_loop\_sys)

%figure();

%step(p\_motor)

%%

t\_sample = 0.1;

sys\_discrete=c2d(p\_motor,t\_sample,'zoh')

figure();

step(sys\_discrete);

Kp\_discrete=0.17497;

Ki\_discrete=0.12683;

Kd\_discrete=0.0084317;

discrete\_pid=pid(Kp\_discrete,Ki\_discrete,Kd\_discrete,0,t\_sample);

cls\_loop\_pid=feedback(sys\_discrete\*discrete\_pid,1)

figure();

step(cls\_loop\_pid);

p\_motor =

0.0274

-------------------------------------------

8.878e-12 s^3 + 1.291e-05 s^2 + 0.0007648 s

Continuous-time transfer function.

cls\_loop\_pid =

0.001299 s^2 + 0.1044 s + 0.1955

----------------------------------------------------------------

8.878e-12 s^4 + 1.291e-05 s^3 + 0.002064 s^2 + 0.1044 s + 0.1955

Continuous-time transfer function.

sys\_discrete =

2.979 z^2 + 0.5937 z + 2.686e-12

----------------------------------------

z^3 - 1.003 z^2 + 0.002678 z - 7.089e-26

Sample time: 0.1 seconds

Discrete-time transfer function.

cls\_loop\_pid =

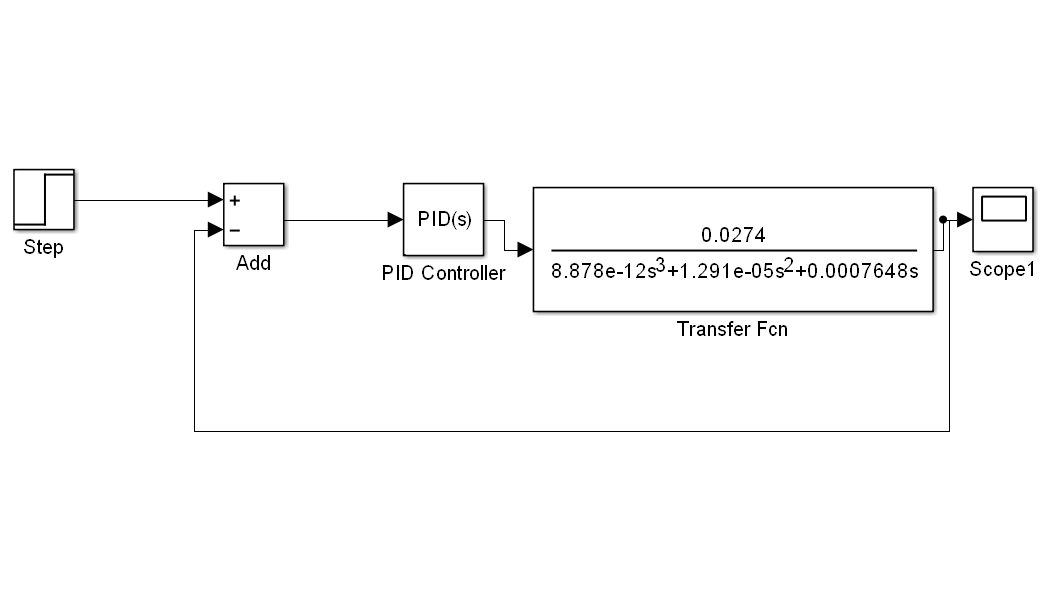
0.2512 z^4 + 0.06894 z^3 - 0.2285 z^2 - 0.04629 z - 2.095e-13

-------------------------------------------------------------

1.251 z^4 - 1.934 z^3 + 0.7768 z^2 - 0.04897 z - 2.095e-13

Sample time: 0.1 seconds

Discrete-time transfer function



In Continuous time domain, the step response of the p\_motor: The system appears to be Unstable as it has unbounded output for bounded input.

**>>step(p\_motor)**



After including PID controller in continuous domain, we get the step response as:



The transfer function then becomes:

cls\_loop\_pid =

0.001299 s^2 + 0.1044 s + 0.1955

----------------------------------------------------------------

8.878e-12 s^4 + 1.291e-05 s^3 + 0.002064 s^2 + 0.1044 s + 0.1955

The system is discretized by using c2d function in MATLAB. The transfer function in Z-Domain becomes and its corresponding step response with a sampling time of 0.1 seconds:

sys\_discrete =

2.979 z^2 + 0.5937 z + 2.686e-12

----------------------------------------

z^3 - 1.003 z^2 + 0.002678 z - 7.089e-26



After including PID controller in continuous domain, we get the step response as:



The transfer function then becomes:

cls\_loop\_pid =

0.2512 z^4 + 0.06894 z^3 - 0.2285 z^2 - 0.04629 z - 2.095e-13

-------------------------------------------------------------

1.251 z^4 - 1.934 z^3 + 0.7768 z^2 - 0.04897 z - 2.095e-13

Sample time: 0.1 seconds

Discrete-time transfer function.