

Especificaciones: $t_s \leq 1.5 \text{ s}, M_p < 5\%, e_{ss} = 0$

Primero para este sistema vamos a obtener los polos y su FDT para llegar a buscar sus mejores respuestas para la estabilización del sistema, así mismo para la obtención de la matriz del sistema para llevar el sistema tanto a Matlab como simulink;

- Usando Matlab

```
>> pretty(expand(sys))
      1
-----
      2
  3   111 s   311 s
s + ---- + ---- + 3
   10    10
```

$$a_0 = 3 \quad a_1 = 31.1 \quad a_2 = 11.1$$

$$b_0 = 1$$

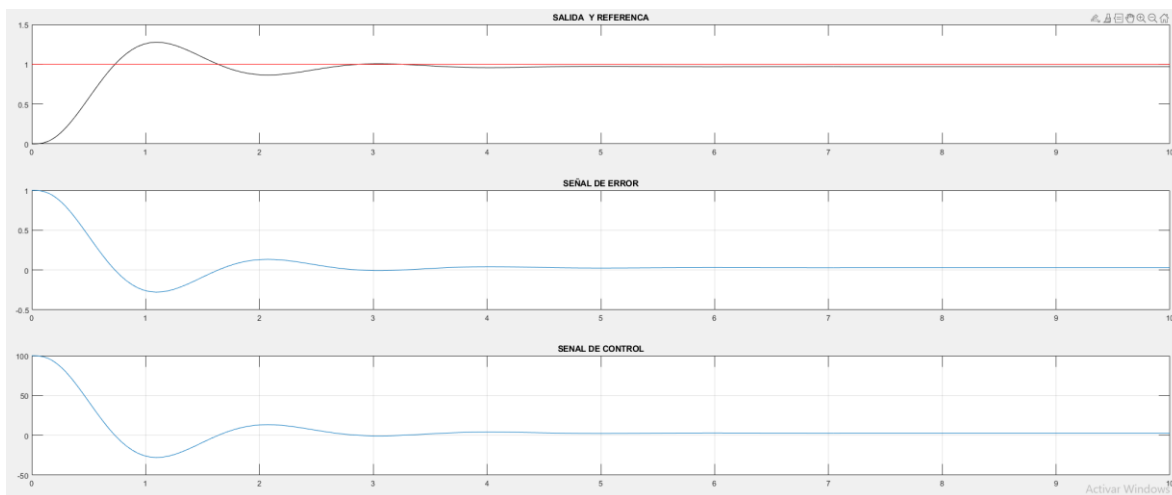
A =

```
      0      1.0000      0
      0      0      1.0000
 -3.0000 -31.1000 -11.1000
```

B =

```
      0 C =
      0
      1      1      0      0
```

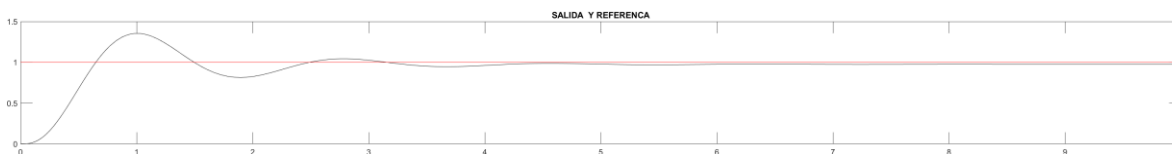
Como se observa en la grafica de resultados, aun no logramos la estabilización en PID que queremos, por lo que ahora buscamos con otros valores de k.



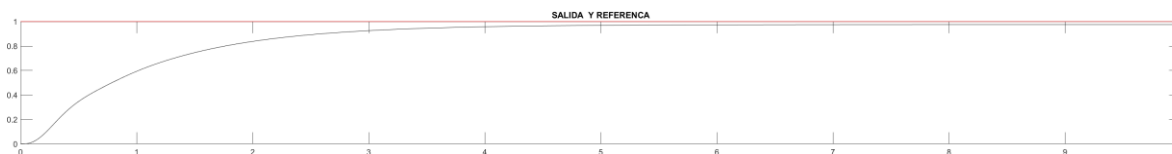
Dichos cambios los haremos siguiendo el siguiente tabla

PID Gain	Percent Overshoot	Settling Time	Steady-State Error
Increasing K_P	Increases	Minimal impact	Decreases
Increasing K_I	Increases	Increases	Zero steady-state error
Increasing K_D	Decreases	Decreases	No impact

- $K_p = 120$ $K_d = 0$ $K_i = 0$



- $K_p = 120$ $K_d = 100$ $K_i = 0$



- $K_p = 120$ $K_d = 100$ $K_i = 20$



- $K_p = 160$ $K_d = 40$ $K_i = 7.6$



CODIGO

```
%% Tarea_24
clc; clear; close all;

tspan = [0,10];
x0 = [0,0,0];
kp = 160;
ki = 7.6;
kd = 40;
k = [kp, ki, kd];

controlador_PID_plot(tspan, x0, k)

%% Functions

function controlador_PID_plot(tspan, x0, k)

global A B C Kp Ki Kd

Kp = k(1);
Ki = k(2);
Kd = k(3);

A = [0, 1, 0; 0, 0, 1; -3, -31.1, -11.1];
B = [0;0;1];
C = [1, 0, 0];

[t,X] = ode45(@controlador_PID_sys, tspan, [0, x0]);

ref = 1;
dref = 0;

figure;

subplot(3,1,1); plot( t, X(:,2:4)*C', 'k'); title('SALIDA Y REFERENCIA');
grid; hold on;
        plot(t, ref*ones(size(t)), 'red'); grid; hold off;

e = ref - X(:,2:4)*C';
subplot(3,1,2); plot(t,e); title('SEÑAL DE ERROR'); grid;

de = dref - X(:,2:4)*A'*C';

ie = X(:,1);

U = Kp*e + Ki*ie + Kd*de;
subplot(3,1,3); plot(t,U); title('SEÑAL DE CONTROL'); grid;
end

function dX = controlador_PID_sys(t,X)
```

```
global A B C Kp Ki Kd

ref = 1;
dref = 0;

e = ref - C*X(2:4);
de = dref - C*A*X(2:4)
ie = X(1);

U = Kp*e + Ki*ie + Kd*de;

dX = [e;
      A*X(2:4) + B*U];

end
```