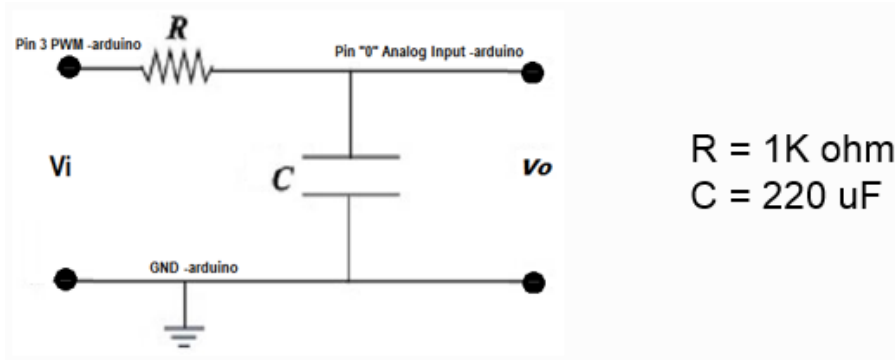


CONTROL PID



Implementar el siguiente Sistema eléctrico:



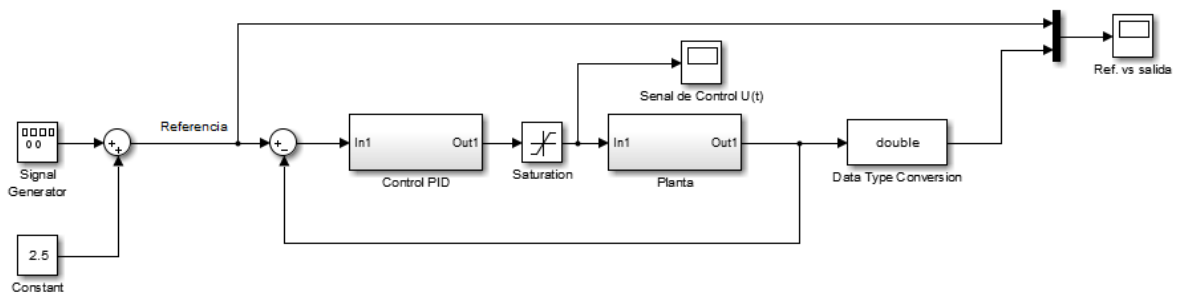
El modelo matemático de la planta es

$$H(s) = \frac{1}{RCs + 1} = \frac{1}{s + \frac{1}{RC}}$$

Diseñar un control PID para cumplir con las siguientes características de desempeño:

$$T_s = 1 \text{ seg} \quad ; SP = 10\% = 0.1$$

Se adjunta el programa en Matlab-Simulink para implementar el controlador diseñado.



Probar con un circuito RC pasabajo donde $R=1\text{Kohmio}$ y $C=220\text{uF}$.
Deben definir el periodo de muestreo (T_m) en el workspace de matlab
Usen $T_m=10\text{e-}3$

Con el siguiente código es posible diseñar y simular el controlador

```
% CIRCUITO RC DE PRIMER ORDEN

% R -> Resistencia
% Co -> Capacitancia

%          Kt          1
% G(s) = ----- = -----
%          tao*s+1     R*Co*s+1

%Kp -> Ganancia proporcional
%Ki -> Ganancia integral
%Kd -> Ganancia derivativa
%Kt -> Ganancia de la planta
%tao -> constante de tiempo de la planta
%s -> variable de Laplace

%% CONTROL PI
syms Kp Ki Kt tao s

C=(Kp*s+Ki)/s; %controlador
G=Kt/(tao*s+1); %planta (circuito RC de primer orden)

T=C*G/(1+C*G); %Función de lazo cerrado
T=simplify(T);
T=collect(T,s);
pretty(T)

[n, d] = numden(sym(T)); %extraer numerador y denominador

%se divide numerador y denominador en el coeficiente de s^2 que es tao
nume=n/tao;
nume=simplify(nume);
nume=collect(nume,s);

deno=d/tao;
deno=simplify(deno);
deno=collect(deno,s);

T=nume/deno;
pretty(T)

%% Diseno Control PI

%Sistema RC de primer orden
clear,clc
R=1e3; %R=1Kohmio
Co=220e-6; %c=220uF
s=tf('s');
tao=R*Co;
```

```

G=1/(tao*s+1) %planta continua
G=zpk(G)      %planta en la forma cero-polo-ganancia
Tm=10e-3;    %periodo de muestreo
Gd=c2d(G,Tm) %planta discreta
step(G,Gd)   %comparar respuesta planta continua y discreta

```

```

Kt=1; %ganancia del circuito RC
tao=R*Co; %constante de tiempo
Ts=1; %tiempo de establecimiento deseado
SP=0.1; %sobrepico maximo deseado

```

```

sita=abs(log(SP))/sqrt(pi^2+(log(SP))^2);
Wn=4.5/(Ts*sita);

```

```

%Se igualan los coeficientes del denominador a los de un sistema de
segundo orden
%normalizado:

```

```

%      Kp Kt      Ki Kt
%      ---- s + ----
%      tao      tao
% ----- = T(s)
% 2      (Kp Kt + 1) s      Ki Kt
% s + ---- + ----
%      tao      tao

```

```

%      Wn^2
% T(s) = ----
%      s^2+2.sita.Wn.s+Wn^2

```

```

%      Kp.Kt+1
% ----- = 2.sita.Wn  -> Despejar Kp
%      tao

```

```

%      Ki.Kt
% ----- = Wn^2  -> Despejar Ki
%      tao

```

```

Kp=(2*sita*Wn*tao-1)/Kt
Ki=(Wn^2*tao)/Kt
Kd=0 %como es control PI la ganancia derivativa es cero
N=100; %ubicacion polo lejano del derivador

```

```

C=(Kp*s+Ki)/s; %controlador PI
T=C*G/(1+C*G); %funcion de lazo cerrado
T=minreal(T)
step(T)

```

```

%% CONTROL PID
syms Kp Kd Ki Kt tao s

C=(Kd*s^2+Kp*s+Ki)/s; %controlador PID con derivador ideal
G=Kt/(tao*s+1); %planta (conjunto actuador-Motor-sensor)

T=C*G/(1+C*G); %Función de lazo cerrado
T=simplify(T);
T=collect(T,s);
pretty(T)

[n, d] = numden(sym(T)); %extraer numerador y denominador

%se divide numerador y denominador en el coeficiente de s^2 que es tao
nume=n/(tao + Kd*Kt);
nume=simplify(nume);
nume=collect(nume,s);

deno=d/(tao + Kd*Kt);
deno=simplify(deno);
deno=collect(deno,s);

T=nume/deno;
pretty(T)

%% Diseño PID
Kt=1; %ganancia del circuito RC
tao=R*Co; %constante de tiempo

Ts=1; %tiempo de establecimiento deseado
SP=0.1; %sobrepico maximo deseado

sita=abs(log(SP))/sqrt(pi^2+(log(SP))^2);
Wn=4.5/(Ts*sita);

%Se igualan los coeficientes del denominador a los de un sistema de
segundo orden
%normalizado:

%
%          2
%      Kd Kt s      Kp Kt s      Ki Kt
%      ----- + ----- + -----
%      tao + Kd Kt   tao + Kd Kt   tao + Kd Kt
%      ----- = T(s)
%      2      (Kp Kt + 1) s      Ki Kt
%      s  + ----- + -----
%          tao + Kd Kt      tao + Kd Kt

%
%          Wn^2
%      T(s) = -----
%          s^2+2.sita.Wn.s+Wn^2

```

```

%   Kp.Kt+1
%   ----- = 2.sita.Wn   ->  Despejar Kd
%   tao+Kd.Kt

```

```

%   Ki.Kt
%   ----- = Wn^2   ->  Despejar Ki
%   tao+Kd.Kt

```

%NOTA: No olvide ubicar paréntesis en los denominadores al escribir
%ecuaciones en Matlab

```

Kp=1.25   %Ganancia proporcional asumida
Kd=(Kp*Kt+1-2*sita*Wn*tao)/(2*Kt*sita*Wn)
Ki=Wn^2*(tao+Kd*Kt)/Kt

```

```

s=tf('s'); %variable de Laplace
G=Kt/(tao*s+1); %planta
N=100; %para ubicar polo lejano en el derivador
C=Kp+(Ki/s)+Kd*s*N/(s+N); %controlador PID
T=C*G/(1+C*G); %funcion de lazo cerrado
T=minreal(T)
step(T)

```

```

%% CONTROLADOR PI diseñado con la herramienta rltool
s=tf('s');
Kp=0.47319;
Ki=7.2919;
Kd=0;
C=(Kp*s+Ki)/s %control PI diseñado con rltool(G)

```

```

N=100; %ubicacion polo lejano derivador

```

```

T=C*G/(1+C*G); %sistema en lazo cerrado
T=minreal(T) %realizacion minima (simplificar T)
step(T) %respuesta a escalon en lazo cerrado

```