

# SOLUCIÓN FINAL

Profesor: Mario Alejandro Giraldo  
Julian Ricardo Gaona Tovar

[Julian.gaona@eia.edu.co](mailto:Julian.gaona@eia.edu.co)

**Resumen**—En este documento se desarrollara la solución del examen final de control.

## I. SOLUCIÓN PUNTO 1

*I-A. Diseñar(analíticamente) un controlador digital PI por asignación de polos que permita un 70 % de la constante del sistema en lazo abierto y tenga un sobre-impulso de 5 %.*

El primer paso es la construcción de la función de transferencia del sistema, la cual cuenta con los siguientes parámetros y estructura:

$$G_o = \frac{K}{\tau \cdot s + 1} \quad (1)$$

$$\tau_o = 6,25 \quad K = \frac{35}{10}$$

la función de transferencia queda de la siguiente forma

$$G_o = \frac{3,5}{6,25 \cdot s + 1} \quad (2)$$

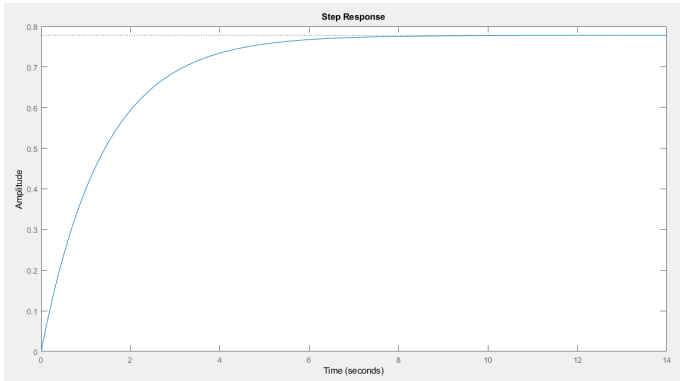


Figura 1: Respuesta ante un escalón

las condiciones deseadas que se requieren cumplir son:

$$\tau_D = 70 \% \cdot \tau_o = 4,37 \quad \%MP = 5 \%$$

con estas condiciones y haciendo uso de criterios y ecuaciones como el criterio de tiempo de establecimiento encontramos los parámetros requeridos como lo son la  $w_n$  y zita.

$$t_s = 5 \cdot \tau_D = 21,85$$

$$\%Mp = e^{\frac{-\pi \cdot \xi}{\sqrt{1-\xi^2}}} \quad t_s = \frac{4}{\xi \cdot w_n}$$

del máximo sobre pico obtenemos:

$$\xi = \frac{1}{\sqrt{1 + \frac{\pi^2}{\ln(Mp)^2}}} = 0,69 \quad (3)$$

y del criterio de tiempo de establecimiento del 2 %

$$w_n = \frac{4}{\xi \cdot t_s} = 0,265 \quad (4)$$

para diseñar el controlador PI por asignación de polos necesitamos la planta del sistema en tiempo discreto, discretizamos por medio del uso de matlab y calculamos el tiempo de muestreo por criterio de tiempo de establecimiento  $t_s$ .

$$0,05 \cdot t_s \leq T_s \leq 0,15 \cdot t_s \\ T_s = 0,2715_s$$

función discreta

$$HG(z) = \frac{0,1488}{z - 0,9575} \quad (5)$$

para la asignación usamos la ecuación

$$1 + D(z) \cdot HG(z) = 0 \quad (6)$$

seguido a esto hallamos la ecuación característica del sistema con controlador y la comparamos con la del desempeño deseado, ubicando los polos con los parámetros deseados haciendo uso de zita y  $w_n$ , los resultados son los siguientes:

$$q_0 = 0,382 \quad q_1 = -0,349$$

el controlador en tiempo discreto seria:

$$HG(z) = \frac{0,382 \cdot z - 0,349}{z - 1} \quad (7)$$

Nota 1: todos los cálculos realizados para llegar a estos resultados se encuentran anexados al final del documento.

*I-B. Diseñar (analíticamente) un controlador digital RST que cumpla el desempeño deseado anterior*

para este controlador modificamos la función en tiempo discreto de la siguiente forma:

$$HG(z^{-1}) = \frac{z^{-1} \cdot (0,1488)}{1 - 0,9575 \cdot z^{-1}}$$

construimos el polinomio deseado

$$Q(z^{-1}) = 1 - p \cdot z^{-1} \quad (8)$$

debido a que el orden de los polinomios dan un valor de 0 tenemos solo las constantes  $R_0$  y  $S_0$

$$R_0 = 1 \quad S_0 = 0,1190$$

$$T(1) = 0,4046$$

Nota 2: cálculos y despejes en anexos al final del documento.

*I-C. Realizar las simulaciones y determinar cualitativamente y cuantitativamente controlador implementar*

Se implementan los controladores hallados al sistema en tiempo continuo con setpoints y perturbación de la siguiente forma

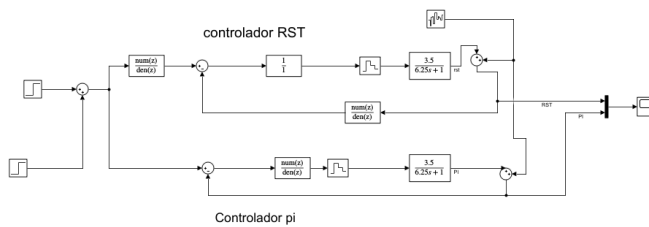


Figura 2: Diagrama de bloques simulink

Controlador PI

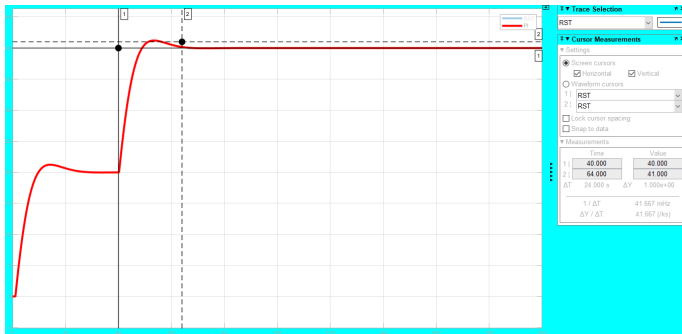


Figura 3: Respuesta controlador PI con planta

Se le ingresa dos setpoint uno de 20 y otro de 40, además se le agrega una perturbación. el controlador se sintoniza modificando los parámetros de  $q_0$  y  $q_1$  de los valores semilla hallados anteriormente, se observa que cumple con los parámetros deseados que es un sobre-impulso del 5 % y una constante de tiempo del 70 % de la del lazo cerrado ya que se estabiliza en el tiempo establecido. Los nuevos coeficientes sintonizados son:

$$q_0 = 0,377 \quad q_1 = -0,349$$

Controlador RST

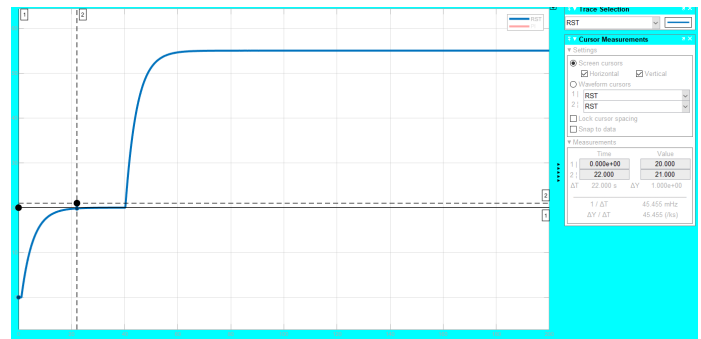


Figura 4: Respuesta controlador RST con planta

Se ingresan los mismo valores que el controlador anterior, este cumple con el tiempo de establecimiento pero no con el sobre-impulso.

Controlador PI vs RST



Figura 5: Respuesta PI vs RST con planta

Se evidencia que el mejor controlador a implementar es el PI por asignación de polos ya que es el único que cumple con todos los requisitos, pese a que ambos cumplen con el tiempo de establecimiento el RST posicional no puede controlar la dinámica del sistema deseado ya que este tipo de controlador es de primer orden y nunca va a alcanzar un sobre-impulso, además se observa que ambos tiene error en estado estable igual a 0.

## II. SOLUCIÓN PUNTO 2

*II-A. Diseñar el filtro de Kalman lineal para estimar la corriente de armadura y la velocidad angular a partir de la corriente medida.*

Para implementar el filtro primero definimos nuestras características del sistema, constantes, variables de estados y demás

```

%% Definición de parámetros motor DC

Ra=99.5833;           %Resistencia de armadura [Ohmios]
La=0.58378;           %Inductancia de armadura [Henrios]
Kv=0.09168;           %Constante de fuerza contraelectromotriz [Voltios/(Rad*Web)]
J=0.001537;           %Momento de inercia [Kg*m^2]
Beta=0.000752;        %Coeficiente de fricción viscosa [N*m*s/Rad]
Va=12;                %Voltaje de armadura [Voltios]
TL=0;                 %torque de carga [N*m]

%% Modelo en espacio de estados

% Los estados son corriente de armadura Ia [A] y velocidad angular Wa [rad/s]
% La entrada es voltaje Va [V]
% La salida medida es la corriente de armadura Ia [A]

A = [-Ra/La -Kv/La;Kv/J -Beta/J]; % Matriz de transición del estado
B = [1/La 0]; % Matriz de entrada
C = [1 0]; % Matriz de salida
D = 0; % Matriz de transferencia directa

```

Figura 6: Definición parámetros iniciales

```

%%inicializamos las variables del modelo a estimar
u=[0 0]';
xs=pinv(A)*(-B*u);

%%

%measurement error matrix
R=cov(Motor(2000:4000,3));

%Error dynamic matrix (model uncertainties)
Q=100*R*eye(2);

%Error covariance matrix
P=1000*eye(2);

%%

```

Figura 9: Parámetros Kalman

seguido de esto cargamos los datos medidos

```

%% Validación de modelo

% Datos de corriente

Motor = load ('current.txt');
tc = Motor(:,1);
Va_t = Motor(:,2);
Ia_t = Motor(:,3);
% Modelo

% Condiciones iniciales de entrada y estados

u=[0 0]';

xv=pinv(A)*(-B*u);

```

Figura 7: Cargar Datos

Tomamos los datos en un rango donde estos se encuentran en estado estable lo cual se observa alrededor de los 20ms hasta que terminan, aplicamos la varianza de estos datos para así conformar nuestra matriz R la cual es la que contiene el ruido de la medición, esta es de 1x1 ya que se tiene una sola medición.

inicialmente se define la matriz Q que es la que tiene la incertidumbre del sistema como 10 veces la matriz R, al simular se evidencia que se obtienen mejores resultados cuando esta es 100 veces la matriz R.

La matriz P se inicializo con un valor de 1000 esta a rain de varias iteraciones que permitieran la correcta sintonización del filtro.

La estructura de este algoritmo es la misma a la usada en las clases anteriores

Definidos todo lo necesario cargamos el modelo

```

% ejemplo de escalon de voltaje (mV)
t_escalon_Va=6.3;
T_escalon=(t_escalon_Va/delta);
Ia(1)=xv(1);
wa(1)=xv(2);

for i=1:n

    cambio_xv = A*xv(:,i) + B*u;

    xv(:,i+1) = xv(:,i) + delta*cambio_xv;

    if i>T_escalon
        u(1) = Va;
    else
        u(1) = 0;
    end
end

```

Figura 8: Modelo

```

for i=1:n

    %cargamos el modelo del sistema
    cambio_xs = A*xs(:,i) + B*u;
    xs(:,i+1) = xs(:,i) + delta*cambio_xs;

    if i>T_escalon
        u(1) = Va;
    else
        u(1) = 0;
    end

    P=A*P*A'+Q;
    %Kalman gain
    K = P*C'*inv(C*P*C'+R);
    %State estimation update
    xs(:,i+1) = xs(:,i+1) + K*(Motor(i,3) - C*xs(:,i+1));
    %Covariance matrix update
    P = (eye(2) - K*C)*P;
end

```

Figura 10: Filtro Kalman

Iniciamos con la receta para poder implementar el filtro de Kalman

Para finalizar graficamos los datos reales de corriente de armadura con los datos estimados por el filtro, a su vez se compara el modo de velocidad del sistema con el modelo estimado por el filtro.

```

% Comparación para corriente
t=0:delta:n*delta;
t2=0:delta2:n2*delta2;
figure(1);
plot(t,xs(1,:), 'b', 'LineWidth', 3);
hold on
plot(tc,Ia_t);
grid on
title('CORRIENTE DE ARMADURA')
xlabel('Tiempo (ms)')
ylabel('Corriente (A)')
legend('Datos estimados', 'Datos experimentales')
%comparacion modelo velocidad
figure(2)
plot(t,xs(2,:), 'b', 'LineWidth', 4);
hold on
plot(tc,xv(2,:), 'lineWidth', 2);
grid on
title('VELOCIDAD ANGULAR ')
xlabel('Tiempo (ms)')
ylabel('velocidad rad/s')
legend('Estimado', 'Modelo')

```

Figura 11: Graficar respuesta

La respuesta de la estimación de la corriente de armadura es la siguiente vs la corriente real medida:

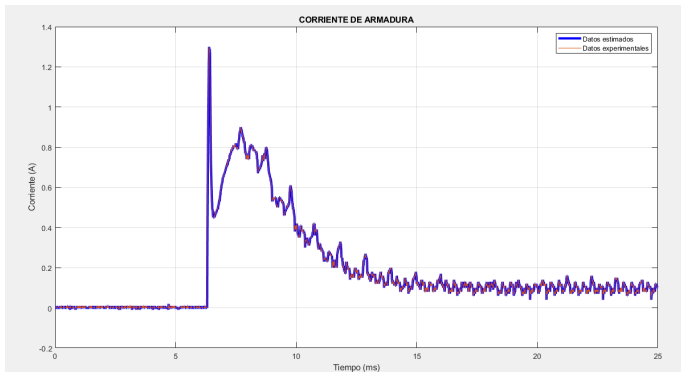


Figura 12: Estimación corriente de armadura

Al realizar el análisis del gráfico obtenido de la comparación entre la medición de la corriente y el valor estimado de ella, se observa que la estimación de los datos medidos es muy óptima y se apega muy bien al comportamiento de los datos, por ende la estimación cumple con lo deseado y nos entrega una buena corrección de estos datos.

La estimación del modelo de velocidad angular que de la siguiente forma:

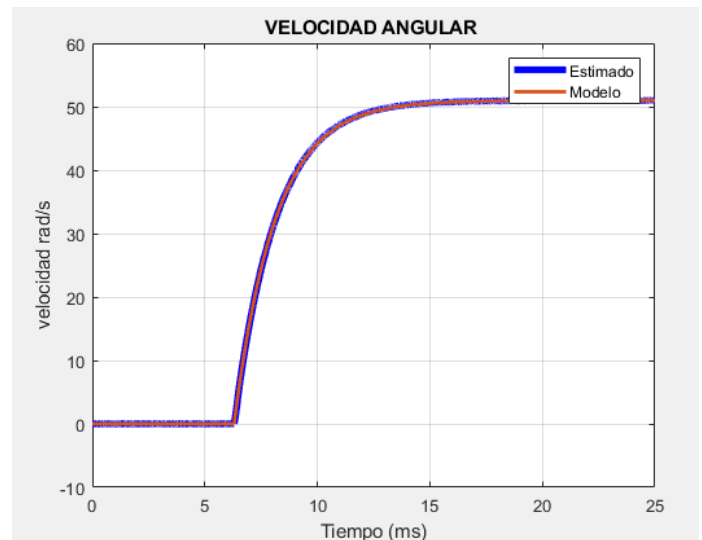


Figura 13: Estimación Modelo de velocidad angular

La estimación del modelo de velocidad angular a partir de la medición de corriente de armadura entrega excelentes resultado ya que estamos obteniendo valores de medida de una variable no medida a partir de la medición de la corriente y estos datos estimados se asemejan al comportamiento del modelo entregado.

El filtro de Kalman es una herramienta capaz de tomar la información del error de entrenamiento y convertirla en un valor de corrección y pronóstico, esta es una de las razones por la que es tan útil y permitió tener importantes usos a lo largo de los años.



# Corrección

$\tau$  es cuando la respuesta alcanza el 63,2%

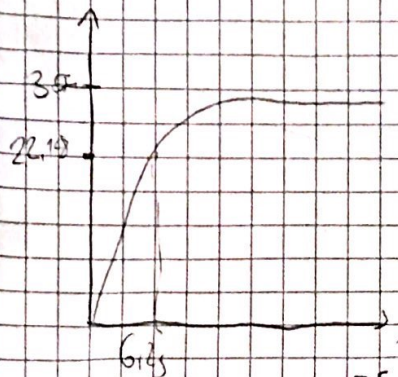
$$G(s) = \frac{K}{\tau s + 1}$$

$$K = \frac{\Delta y}{\Delta u}$$

$$\tau = 0,632 \times \Delta y$$

$$K = \frac{35}{10} = 3,5 \quad \tau_0 = 6,25s$$

$$G(s) = \frac{3,5}{6,25s + 1}$$



1) Condiciones de diseño

$$MP = 5\% \quad \tau = 70\% \times 6,25 = 4,37s$$

para hallar el tiempo de establecimiento del nuevo sistema multiplicamos el  $\tau_0$  deseado por 5

$$\tau_{sa} = 21,85$$

$$MP = e^{-\frac{\pi \xi}{\sqrt{1-\xi^2}}}$$

$$\xi_s = \frac{4}{\xi_{un}} \rightarrow \text{criterio del } 2\%$$

$$\xi = \frac{1}{\sqrt{1 + \frac{\pi^2}{\ln^2(MP)}}}$$

$\rightarrow$  Ecuación para hallar  $\xi$  y  $\tau$

$$\xi = \frac{1}{\sqrt{1 + \frac{\pi^2}{\ln^2(0,05)}}} = 0,69$$

$$\tau_{sa} = 21,85 = \frac{4}{\xi \omega_n} \Rightarrow$$

$$21,85 = \frac{4}{0,69 \times \omega_n} \Rightarrow \omega_n = \frac{4}{0,69 \times 21,85}$$

$$\omega_n = 0,265$$

hallamos el  $T_s$  con criterio de tiempo de establecimiento en lazo cerrado

$$0,05 \tau_{ss} \leq T_s \leq 0,15 \tau_{ss} \quad \tau_{ss} = 5,43s$$

$$\tau_s = 0,2715s$$

Discretizamos en Matlab

$$H(z) = \frac{0,1488}{z - 0,9575}$$

Para el PI por asignación de polos

$$\text{Usamos } 1 + D(z)H(z) = 0$$

$$1 + \frac{q_0 z + q_1}{z - 1} \cdot \frac{0,1488}{z - 0,9575} = 0$$

$$1 + \frac{0,1488 z q_0 + 0,1488 q_1}{z^2 - 1,9575 z + 0,9575} = 0$$



$$z^2 - 1,9575z + 0,9575 + 0,1488zq_0 + 0,1488q_1 = 0$$

$$z^2 + (0,1488q_0 - 1,9575)z + (0,9575 + 0,1488q_1) = 0$$

Para los polos deseados usamos

$$|z| = e^{-\zeta \omega_n T_s}$$

$$= e^{-0,69 \times 0,265 \times 0,2715} = 0,9516$$

Parte real

$$R = |z| \cdot \cos(\omega_n T_s \sqrt{1 - \zeta^2})$$

$$= 0,9516 \cdot \cos(0,265 \times 0,2715 \sqrt{1 - 0,69^2}) = 0,9503$$

$$Im = |z| \cdot \sin(\omega_n T_s \sqrt{1 - \zeta^2})$$

$$= 0,9516 \cdot \sin(0,265 \times 0,2715 \sqrt{1 - 0,69^2}) = 0,0495$$

así formamos la ecuación deseada

$$(z - 0,9503 + 0,0495i)(z - 0,9503 - 0,0495i) = 0$$

$$z^2 - 1,9006z + 0,9503^2 + 0,0495^2 = 0$$

igualamos polinomio deseado a el del sistema

$$-1,9006 = 0,1488q_0 - 1,9575$$

$$0,055 = 0,1488q_1 + 0,9575$$

$$q_0 = 0,382$$

$$q_1 = -0,349$$

$$D(z) = \frac{0,382z - 0,349}{z - 1}$$

b) Control RST

Deseado

$$H(z) = \frac{0,1488}{z - 0,9575} \times \frac{z^{-1}}{z^{-1}}$$

$$\tau = 4,34 \quad MP = 5\%$$

$$= \frac{z^{-1}(0,1488)}{1 - 0,9575z^{-1}}$$

grado

$$nr = 0 + 1 - 1 = 0 \Rightarrow r_0$$

$$z^{-d} \Rightarrow d = 1$$

$$ns = 1 - 1 = 0 \Rightarrow s_0$$

Polinomio deseado

$$Q(z^{-1}) = 1 - p_1 z^{-1}$$

$$\Leftrightarrow p_1 = e^{-\frac{\tau s_0}{T_s}} = e^{-\frac{0,2715 \times 4,34}{0,265}}$$

$$Q(z^{-1}) = 1 - 0,9398z^{-1}$$

$$p_1 = e^{-\frac{0,2715 \times 4,34}{0,265}} = 0,9398$$



$$1 - 0,9398z^{-1} = (1 - 0,9575z^{-1})R_0 + z^{-1}(0,1488)\xi_0$$

$$1 - 0,9398z^{-1} = R_0 - 0,9575z^{-1}R_0 + z^{-1}(0,1488)\xi_0$$

$$R_0 = 1$$

$$-0,9398 = (0,1488\xi_0 - 0,9575)$$

$$\xi_0 = 0,1190$$

$$T(z) = \frac{Q(z)}{B(z)}$$

$$T(z) = \frac{1 - 0,9398z^{-1}}{0,1488} = 0,4046$$