



ACTIVIDAD ADELANTO NOTA (CONTROL II) - INFORME 2

MIACON | MÓDULO 2: Identificación y Control de una Planta de Primer Orden [1]

Diego Andrés García Díaz - 2195533

diego2195533@correo.uis.edu.co

Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones

Universidad Industrial de Santander

Octubre 15, 2025

Resumen

En este informe se documenta el proceso de diseño de los diferentes controladores teniendo como base el modelo previamente identificado de la planta del sistema de primer orden.

1. Objetivos

1.1. Objetivo General

Diseñar los controladores (analógicos y/o digitales) para lograr ejercer el adecuado control y eliminar en lo posible el error en estado estacionario.

1.2. Objetivos Específicos

1. Diseñar el controlador proporcional (P).
2. Diseñar el controlador proporcional-integral (PI).
3. Diseñar el controlador proporcional-derivativo (PD).
4. Diseñar el controlador proporcional-integral-derivativo (PID).

2. Marco Teórico

2.1. Control PID

El controlador PID (Proporcional-Integral-Derivativo) es una de las estrategias de control más empleadas en la industria debido a su simplicidad y efectividad para regular sistemas lineales y no lineales. Su acción de control se compone de tres términos:

- **Acción proporcional (P):** responde de manera inmediata al error actual, proporcionando una corrección proporcional a la diferencia entre el valor de referencia y la variable medida. Su efecto es reducir la magnitud del error, aunque no lo elimina completamente.
- **Acción integral (I):** acumula el error a lo largo del tiempo y ajusta la salida del controlador en consecuencia. Gracias a esta acción, es posible **eliminar el error en estado estacionario**, algo que los controladores P o PWM simples no logran.
- **Acción derivativa (D):** anticipa el comportamiento futuro del error al considerar su tasa de cambio. De este modo, mejora la estabilidad del sistema y reduce la tendencia a oscilar, aunque también lo hace más sensible al ruido de la señal de medición.

La ecuación general del controlador PID en el dominio del tiempo es:

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (1)$$

donde $u(t)$ es la señal de control, $e(t)$ el error de seguimiento y K_p , K_i , K_d los parámetros de sintonización. En el dominio de Laplace, el controlador PID puede expresarse como:

$$C(s) = K_p + \frac{K_i}{s} + K_d s \quad (2)$$

2.1.1. Forma discreta del PID

Cuando se implementa en un microcontrolador o computadora digital, el PID debe expresarse en forma discreta. Una representación común utilizando el método de diferencias hacia atrás es:



2.3. Diseño de Controladores Clásicos por ubicación de polos

$$u[k] = u[k-1] + K_p(e[k] - e[k-1]) + K_i T_s e[k] + \frac{K_d}{T_s}(e[k] - 2e[k-1] + e[k-2]) \quad (3)$$

donde:

- $u[k]$ es la salida de control en el instante k ,
- $e[k]$ es el error en el instante k ,
- T_s es el tiempo de muestreo.

Este esquema permite implementar el controlador PID en plataformas como Arduino, ESP32 o en simulaciones discretas en Simulink, garantizando un comportamiento similar al controlador continuo siempre que el tiempo de muestreo T_s sea suficientemente pequeño.

2.2. Modelado de un Sistema de Primer Orden

La planta térmica se modela como un sistema de primer orden con retardo (ver ecuación (4)). Para identificarlo, se puede utilizar el método de Dos Puntos de Smith. Este método permite obtener los parámetros T (constante de tiempo) y L (tiempo muerto o retardo) a partir de la respuesta del sistema a un escalón. Con la gráfica obtenida, se miden los tiempos t_1 y t_2 y se calculan los parámetros de la planta. [2] [3]

$$G_p(s) = \frac{K_p}{T_{p1} \cdot s + 1} \cdot e^{-T_d \cdot s} \quad (4)$$

Debido a que $e^{-30 \cdot s}$ es una *función trascendental*, no se puede hacer la igualación, por ello se hizo uso de la **Aproximación de Padé**, con el fin de hacer una *aproximación racional*. En este caso, se hizo uso de una aproximación de Padé de **Primer Orden** (ver ecuación (5)).

$$e^{-\theta \cdot s} \approx \frac{-\frac{\theta}{2} \cdot s + 1}{\frac{\theta}{2} \cdot s + 1} \quad (5)$$

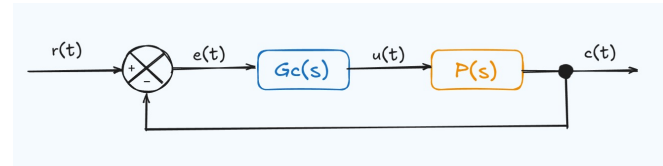


Fig. 1: Diagrama de bloques para el diseño de los diferentes controladores

Idealmente se busca que la función de transferencia en lazo cerrado sea igual o aproximadamente igual a un **Sistema General de Segundo Orden**, tal y como se observa en la ecuación (6).

$$G_{LC}(s) = \frac{C(s)}{R(s)} = \frac{G_C(s) \cdot P(s)}{1 + G_C(s) \cdot P(s)} \approx \frac{\omega_n^2}{s^2 + 2 \cdot \zeta \cdot \omega_n \cdot s + \omega_n^2} \quad (6)$$

La función de transferencia del controlador se puede generalizar, esto se observa en la ecuación (7) y la planta o sistema se generaliza como se observa en la ecuación (8).

$$G_C(s) = \frac{U(s)}{E(s)} = K \frac{(s + z_1) \cdot (s + z_2) \cdots (s + z_m)}{(s + p_1) \cdot (s + p_2) \cdots (s + p_n)} \quad (7)$$

$$P(s) = \frac{N(s)}{D(s)} \quad (8)$$

A continuación, se opera la función de transferencia de lazo cerrado (ver ecuaciones (9) y (10)) para poder hacer la igualación con la forma general de un sistema de segundo orden (ver ecuación (12)).

$$G_{LC}(s) = \frac{C(s)}{R(s)} = \frac{K \frac{(s + z_1) \cdot (s + z_2) \cdots (s + z_m)}{(s + p_1) \cdot (s + p_2) \cdots (s + p_n)} \cdot \frac{N(s)}{D(s)}}{1 + K \frac{(s + z_1) \cdot (s + z_2) \cdots (s + z_m)}{(s + p_1) \cdot (s + p_2) \cdots (s + p_n)} \cdot \frac{N(s)}{D(s)}} \quad (9)$$

$$G_{LC}(s) = \frac{K \cdot (s + z_1) \cdot (s + z_2) \cdots (s + z_m) \cdot N(s)}{(s + p_1) \cdot (s + p_2) \cdots (s + p_n) \cdot D(s) + K \cdot (s + z_1) \cdot (s + z_2) \cdots (s + z_m) \cdot N(s)} \quad (10)$$

$$\frac{K \cdot (s + z_1)(s + z_2) \cdots (s + z_m) \cdot N(s)}{(s + p_1)(s + p_2) \cdots (s + p_n) \cdot D(s)} + K \cdot (s + z_1)(s + z_2) \cdots (s + z_m) \cdot N(s) \approx \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (11)$$

$$(s + p_1) \cdot (s + p_2) \cdots (s + p_n) \cdot D(s) + K \cdot (s + z_1) \cdot (s + z_2) \cdots (s + z_m) \cdot N(s) \approx s^2 + 2\zeta\omega_n s + \omega_n^2 \quad (12)$$

3. Metodología

Se desarrolló un **Live Script (.mlx)** en Matlab para mostrar de una forma interactiva, el desarrollo del diseño de los diferentes controladores. Este archivo se encuentra en el repositorio donde se encuentra almacenado todo el proceso realizado para esta colaboración del proyecto de grado MIACON. [4]

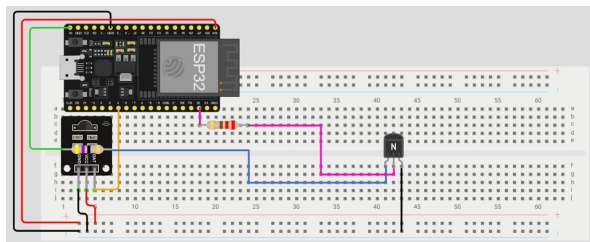


Fig. 2: Circuito usado para la identificación de la planta y posterior diseño de los controladores

En general para el diseño de cualquier controlador (P, Pi, PD o PID) de forma analítica, se hace teniendo en cuenta el siguiente procedimiento que en este caso se aplicó para el diseño del **Controlador Proporcional (P)**:

$$G_{LC}(s) = \frac{C(s)}{R(s)} = \frac{G_C(s) \cdot P(s)}{1 + G_C(s) \cdot P(s)} \approx \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (13)$$

Para la función de transferencia de $G_C(s)$ se tiene como referencia la ecuación generalizada del controlador (ver ecuación (7)). Igualmente, para la Planta o Sistema se tiene la ecuación generalizada (ver ecuación (8)).

Haciendo la respectiva igualación (ecuación (12)), reemplazar los valores, se obtiene el resultado de la ecuación (14):

$$(170,65 \cdot s + 1) + K_P \cdot (0,11864 \cdot e^{-30 \cdot s}) = s^2 + 2 \cdot \zeta \cdot \omega_n \cdot s + \omega_n^2 \quad (14)$$

Debido a la función trascendental, se procede a realizar una aproximación usando la ecuación (5) y se obtiene el resultado de la ecuación (15):

$$e^{-30 \cdot s} = \frac{-\frac{30}{2} \cdot s + 1}{\frac{30}{2} \cdot s + 1} = \frac{-15 \cdot s + 1}{15 \cdot s + 1} = \frac{-s + 0,06667}{s + 0,06667} \quad (15)$$

A continuación se reemplaza $e^{-30 \cdot s}$ de la ecuación (14) por lo obtenido en la ecuación (15). El resultado se visualiza en la ecuación (16):

$$(170,65 s + 1) + K_P \left(0,11864 \frac{1 - 15s}{1 + 15s} \right) = s^2 + 2\zeta\omega_n s + \omega_n^2 \quad (16)$$

Operando la parte izquierda de la ecuación se obtiene el resultado de la ecuación (17):

$$s^2 + (0,07253 - 0,00069K_P)s + (0,0003907 + 0,0000463K_P) = s^2 + 2\zeta\omega_n s + \omega_n^2 \quad (17)$$

Haciendo las respectivas igualaciones con la parte derecha de la ecuación (17), se tiene lo siguiente:

$$2\zeta\omega_n = 0,07253 - 0,00069K_P \quad (18)$$

$$\zeta\omega_n = 0,03627 - 0,00035K_P \quad (19)$$

Ahora, se asume un parámetro para el diseño del controlador, en este caso se asume un *Overhoot* $\leq 20\%$, esto para obtener el **Factor de Amortiguamiento** (ζ), tal y como se visualiza en la ecuación (20):

$$\zeta \geq \sqrt{\frac{\ln(OV)^2}{\pi^2 + \ln(OV)^2}} = 0,456 \quad (20)$$

Debido a que $\zeta < 1$, el sistema resultante es **subamortiguado**, más adelante se calcula el **Tiempo de Establecimiento** (t_s) con la ecuación (21):

$$t_{s(2\%)} = \frac{4}{\zeta\omega_n} \quad (21)$$

Ahora, con el valor encontrado de ζ (ver ecuación (20)) se puede despejar ω_n , obteniendo la ecuación (23):

$$(0,456) \cdot \omega_n = 0,03627 - 0,00035K_P \quad (22)$$

$$\omega_n = 0,03627 - 0,00035K_P \quad (23)$$

Finalmente, se hace la última igualación de la ecuación (17) para obtener la ecuación (24):

$$\omega_n^2 = 0,0003907 - 0,0000463K_P \quad (24)$$

El siguiente paso es reemplazar la ecuación (23) en la ecuación (24), agrupar términos semejantes y despejar la respectiva constante (K_P) del controlador, haciendo las respectivas operaciones se obtiene la siguiente ecuación (25):

$$(0,07954 - 0,00077K_P)^2 = 0,0003907 + 0,0000463K_P \quad (25)$$

Resolviendo la ecuación (25), se obtienen dos posibles resultados para la ganancia proporcional (K_P), la cual corresponde al **Controlador Proporcional (P)**:

$$K_{P_1} = 243,5872 \quad (26)$$

$$K_{P_2} = 41,1009 \quad (27)$$

El siguiente paso es obtener los respectivos valores para ω_n , esto se realiza utilizando la ecuación (24):

$$\omega_{n_1} = 0,10802 \frac{\text{rad}}{\text{s}} \quad (28)$$

$$\omega_{n_2} = 0,04789 \frac{\text{rad}}{\text{s}} \quad (29)$$

Por último, se puede calcular el **Tiempo de Establecimiento** (t_s) usando la ecuación (21) ya que previamente se calculó ζ y ω_n :

$$t_{s(2\%)_1} = 81,20653[s] \quad (30)$$

$$t_{s(2\%)_2} = 183,1683[s] \quad (31)$$

El siguiente paso consiste en validar el controlador diseñado mediante simulaciones en MATLAB y/o Simulink, así como en realizar su sintonización utilizando el método de Ziegler-Nichols y una herramienta de sintonización automática. Posteriormente, se llevarán a cabo las comparaciones entre los resultados obtenidos por ambos métodos.

Finalmente, se resalta que este mismo procedimiento

se aplicará para el diseño analítico y “automático” de los demás controladores, con el fin de evaluar y contrastar los diferentes resultados de cada uno de ellos.

```
1 % Hecho por: Diego Andres Garcia Diaz.
2 %Codigo: 2195533.
3 % Asignatura: Control II (Adelanto de Nota)
4 % -----
5 % Diseno Controladores para Control de
   Temperatura
6 % -----
7 clc; clear; close all;
8 s = tf('s');
9
10 % Definicion de parametros del sistema (
   Planta) identificada
11 Kp = 0.11864; % Ganancia del sistema
12 Tp1 = 170.65; % Tiempo de establecimiento
13 L = 30; % Tiempo muerto o Delay
14
15 % Funcion de Transferencia del modelo
   identificado (Actuador + Planta +
   Sensor)
16 P_s = ( (Kp * exp(-L*s) ) / ((Tp1*s) + 1) )
17
18 % Funcion de Transferencia factorizada
19 P_s_fact = zpk(P_s);
20
21 % Abrir 'sisotool' para diseno y analisis
   de los controladores
22 sisotool(P_s)
```

Listing 1: Código en MATLAB para la definición de la planta.

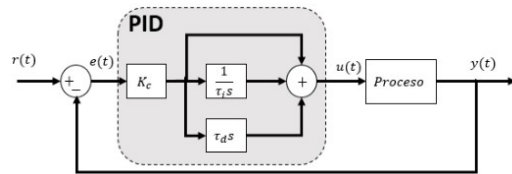


Fig. 3: Diagrama de bloques para diseño de los controladores

Controlador	K_p	T_i	T_d
P	$\frac{T}{K_L}$	∞	0
PI	$\frac{0.9T}{K_L}$	$\frac{L}{0.3}$	0
PID	$\frac{1.2T}{K_L}$	$2L$	$0.5L$

Fig. 4: Tabla para diseño de controladores con el método de Ziegler-Nichols

Si los controladores se diseñan usando el método de Ziegler-Nichols se debe tener en cuenta la figura 4 y las ecuaciones (32) y (33):

$$G_{C_{PID}}(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right) \quad (32)$$

$$K_i = \frac{K_p}{T_i} \quad y \quad K_d = K_p T_d \quad (33)$$

3.1. Diseño controlador Proporcional (P)

$$G_c(s) = K_p \quad (34)$$

Usando el método de **Ziegler-Nichols** se obtuvo una ganancia proporcional de $K_p = 47,9462$, también se asumieron dos ganancias y usando el método de prueba y error se obtuvo la ganancia máxima antes de que el sistema se haga inestable (se comprobó la estabilidad del sistema usando la función `isstable()`), que para este caso fue de $K_{p_{max}} = 104,3000$.

Adicionalmente, se diseñó el controlador de forma analítica, obteniendo como resultado una ganancia proporcional de $K_P = 41,1009$.

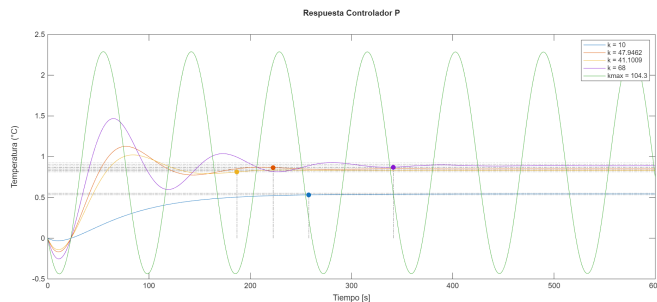


Fig. 5: Respuesta al escalón para diferentes ganancias proporcionales

En la figura 5 se pueden observar las graficas con diferentes ganancias proporcionales:

- $K_{P1} = 10$ -> Se asume
- $K_{P2} = 47,9462$ -> Método Z-N
- $K_{P3} = 41,1009$ -> Método analíticamente
- $K_{P4} = 68$ -> Se asume

- $K_{P_{max}} = 104,3$ -> Calculada por prueba y error

Para las respectivas ganancias, se tienen los siguientes tiempos de establecimiento: 258 [s], 223 [s], 187 [s], 341 [s] y para la ganancia máxima no se define ningún tiempo de establecimiento, puesto que siempre estará oscilando (ver figura 5).

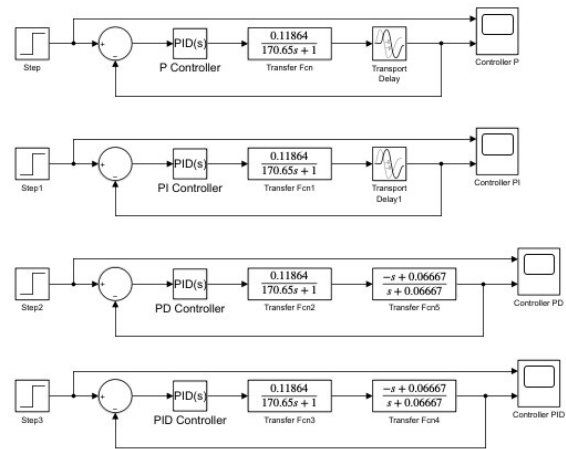


Fig. 6: Validación en Simulink para los diferentes controladores (P, PI, PD y PID) diseñados

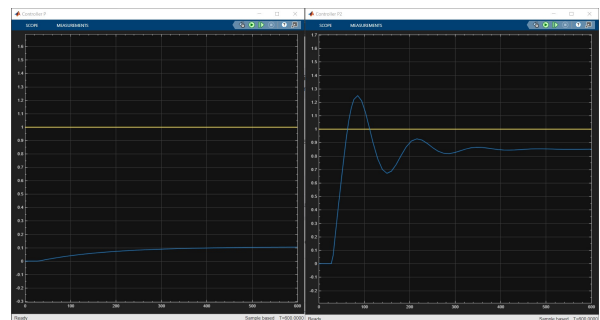


Fig. 7: Validación controlador P con ganancias $K_p = 1$ (izquierda) y $K_p = 47,9462$ (derecha) en Simulink

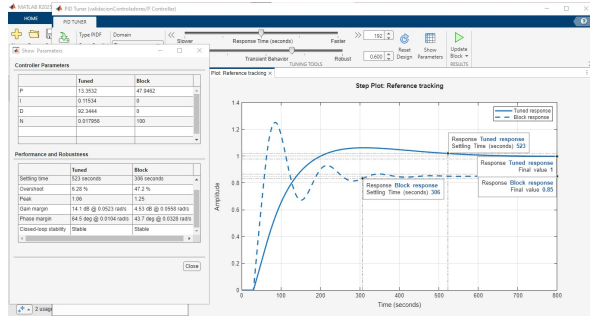


Fig. 8: Diseño “automático” del controlador usando ‘PID Tuner’ de Matlab

3.2. Diseño controlador Proporcional-Integral (PI)

$$G_c(s) = K_p + \frac{K_i}{s} \quad (35)$$

Forma alternativa (con constante de tiempo integral T_i):

$$G_c(s) = K_p \left(1 + \frac{1}{T_i s} \right) \quad (36)$$

- $K_{P1} = 43,1516$ -> Calculada con Z-N
- $K_{i1} = 0,4315$ -> Calculada con Z-N
- $G_{CPI2} = 43,1516 + \left(\frac{0,4315}{s} \right)$ -> Método Z-N
- $G_{CPI3} = 41,1009 + \left(\frac{0,88}{s} \right)$ -> Se asumen
- $G_{CPI4} = 13,3532 + \left(\frac{0,11534}{s} \right)$ -> PID Tuner
- $G_{CPI5} = 39,2173 + \left(\frac{0,18749}{s} \right)$ -> PID Tuner
- $G_{CPI6} = -241,6 + \left(\frac{8,28}{s} \right)$ -> Método analítico

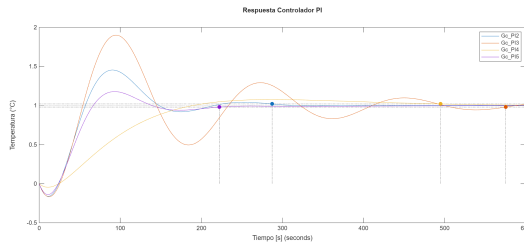


Fig. 9: Respuesta al escalón para diferentes ganancias para el controlador PI

De la figura 9 los respectivos tiempos de establecimiento son: 287 [s], 575 [s], 495 [s] y 222 [s].

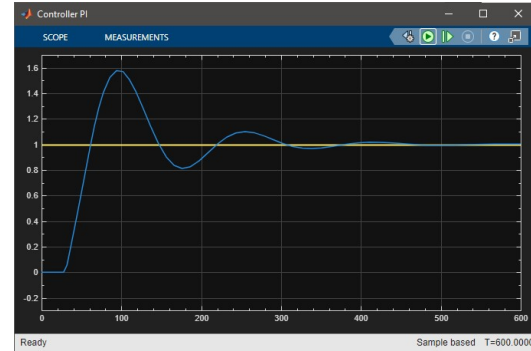


Fig. 10: Validación controlador PI con ganancias $K_p = 43,1516$ y $K_i = 0,4315$ en Simulink

También se usaron las ganancias K_p y K_i generadas por la herramienta **PID Tuner** (ver figura 8) y las respectivas respuestas al escalón se visualiza en la figura 9 en la señal de color *amarillo*. Por último, de forma analítica se obtuvieron las ganancias $K_p = -241,6$ y $K_i = 8,28$, pero al hacer la respectiva respuesta al escalón el sistema se observó inestable (esta respuesta no se graficó).

3.3. Diseño controlador Proporcional-Derivativo (PD)

$$G_c(s) = K_p + K_d s \quad (37)$$

Forma alternativa (con constante de tiempo derivativo T_d):

$$G_c(s) = K_p (1 + T_d s) \quad (38)$$

- $K_{P1} = 57,5354$ -> Calculada con Z-N
- $K_{d1} = 863,0310$ -> Calculada con Z-N
- $G_{CPD2} = 57,5354 + (863,0310 \cdot s)$ -> Método Z-N
- $G_{CPD3} = 47,5354 + (600 \cdot s)$ -> Se asumen
- $G_{CPD4} = 13,3532 + (92,3444 \cdot s)$ -> PID Tuner
- $G_{CPD5} = 35,91 + (-102,8 \cdot s)$ -> Método analítico

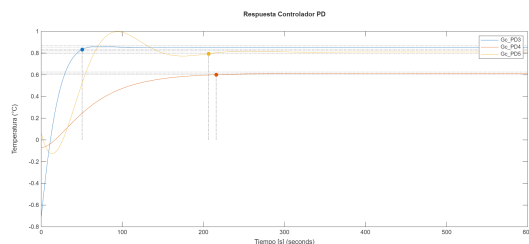


Fig. 11: Respuesta al escalón para diferentes ganancias para el controlador PD

De la figura 11 los respectivos tiempos de establecimiento son: 50.5 [s], 216 [s] y 206 [s]. Para el controlador obtenido mediante Ziegler-Nichols (G_{CPD2}) se generó una respuesta al escalón visualizando un sistema inestable (esta respuesta no se grafico).

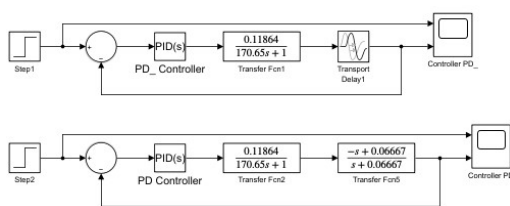


Fig. 12: Validación en Simulink para el controlador PD sin y con aproximación de Padé

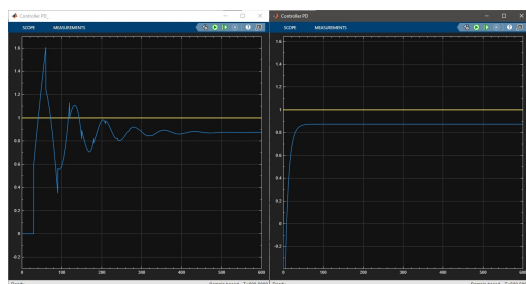


Fig. 13: Validación controlador PD con ganancias $K_p = 57,5354$ y $K_d = 863,0310$ en Simulink, en la izquierda sin aproximación de Padé, en la derecha con aproximación de Padé (ver figura 12)

Usando el método de Ziegler-Nichols se obtuvieron las ganancias $K_p = 47,9462$ y $K_d = 863,0310$, también se asumieron las ganancias para dos controladores, finalmente, también se diseñó el controlador de forma analítica, obteniendo las ganancias $K_p = 35,91$ y $K_d = -102,8$. Se comprobó la estabilidad del sistema usando la función `isstable()`.

3.4. Diseño controlador Proporcional (PID)

$$G_c(s) = K_p + \frac{K_i}{s} + K_d s \quad (39)$$

Forma alternativa (con constantes de tiempo T_i y T_d), ver ecuación (32).

- $K_{P1} = 57,5354$ -> Calculada con Z-N

- $K_{i1} = 0,9589$ -> Calculada con Z-N

- $K_{d1} = 863,0310$ -> Calculada con Z-N

$$- G_{CPID2}(s) = 57,5354 + \frac{0,9589}{s} + 863,0310 \cdot s$$

$$- G_{CPID3}(s) = K_p + \frac{K_i}{s} + K_d s$$

$$- G_{CPID4}(s) = K_p + \frac{K_i}{s} + K_d s$$

$$- G_{CPID5}(s) = K_p + \frac{K_i}{s} + K_d s$$

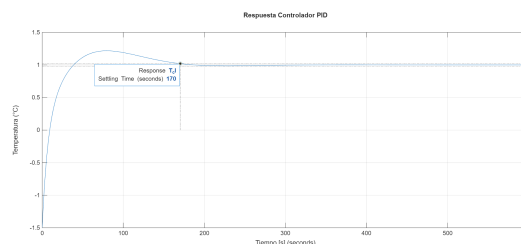


Fig. 14: Respuesta al escalón para diferentes ganancias para el controlador PID

De la figura 14 el respectivo tiempo de establecimiento es: 170 [s].

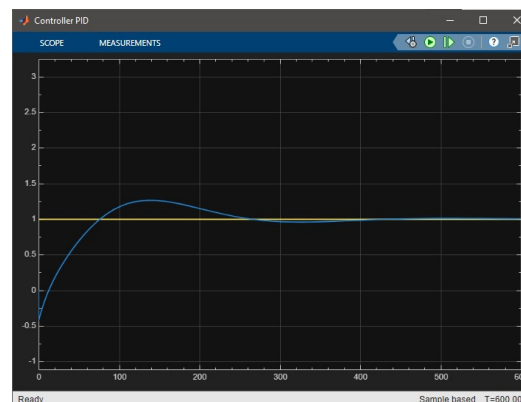


Fig. 15: Validación controlador PID con ganancias $K_p = 57,5354$, $K_i = 0,9589$ y $K_d = 863,0310$ en Simulink



4. Conclusiones

- ME QUEDAN PENDIENTES LAS CONCLUSIONES, LA SECCIÓN “3.4 Diseño controlador Proporcional (PID)”.
- LUEGO AGREGARÉ OTRO CONTROLADOR POR CADA UNO HACINADO USO DE UNA FUNCIÓN PROPIA DE MATLAB “pid()”.
- AGREGAR DE ANEXOS LOS ARCHIVOS “.mlx” y el de Simulink USADOS DURANTE EL DESARROLLO DE ESTA ACTIVIDAD.

Referencias

- den,” 2025. [Online]. Available: <https://sites.google.com/view/miacon-proyectodegrado/gu%C3%ADas-de-laboratorio/m%C3%B3dulo-2-identificaci%C3%B3n-y-control-de-una-planta-de-primer-orden>
- [2] J. J. C. Zagarra, “Método de curva de reacción o de dos puntos de smith,” 2023. [Online]. Available: https://www.youtube.com/watch?v=_6ywp9ryB-Q
- [3] L. F. Rodriguez, “Sintonización pid: Método de smith y corripio,” 2020. [Online]. Available: <https://www.youtube.com/watch?v=8sQChijpKgA>
- [4] D. A. G. Díaz, “Repositorio github.” [Online]. Available: https://github.com/dagdmfc1946/tempControl_Z
- [1] MIACON, “Módulo 2: Identificación y control de una planta de primer or-