

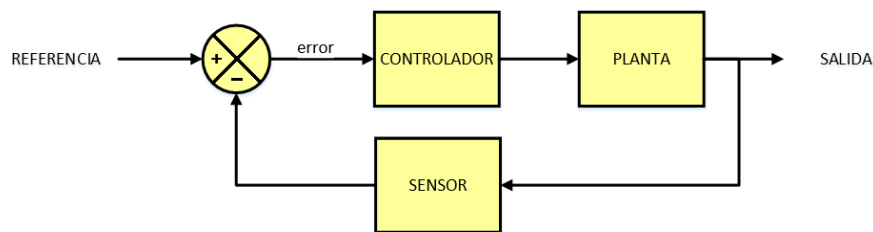
Práctica 4. Control de Actuadores

1. Objetivos

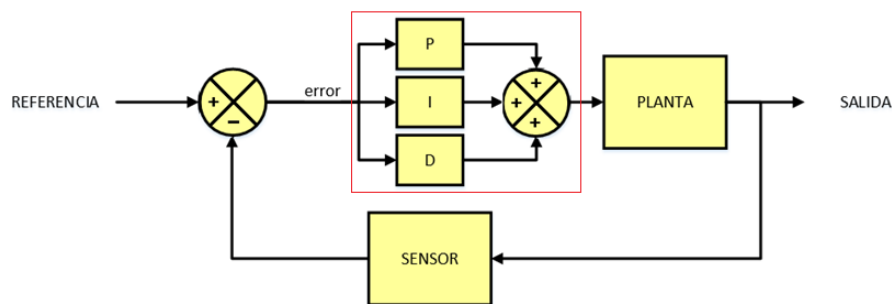
- Aplicar algoritmos y técnicas de control automático a un motor de corriente directa para regular su velocidad.
- Realizar el análisis matemático y comprobar mediante simulación las estrategias de control aplicables a los motores de corriente directa.
- Sintonizar e implementar los controladores P, PI y PID.

2. Antecedentes

- El control automático retroalimentado es una de las bases de la ingeniería moderna, consiste en implementar un controlador el cual interactúa con la señal de error de un sistema en lazo cerrado. La señal de error se constituye como la diferencia aritmética entre una señal de referencia y una señal retroalimentación (misma que ha sido obtenida por un sensor a partir de la señal de salida).



- El controlador por excelencia y que se encuentra presente de manera generalizada en multitud de aplicaciones es el conocido como Proporcional-Integral-Derivativo. El controlador PID incluye tres etapas concurrentes de procesamiento de la señal de error, lo que permite no sólo disminuir las diferencias entre la señal deseada y la señal de salida generada, sino también suavizar las transiciones entre estados del sistema.



- Los motores de corriente directa presentan una velocidad que es de manera práctica, proporcional al voltaje aplicado en sus terminales. A raíz de esta propiedad, las aplicaciones de control más comunes son sobre su posición angular y su velocidad. El controlador PID es aplicable como estrategia de control siempre y cuando se seleccione un método adecuado para leer y reinterpretar eléctricamente la posición y/o velocidad actual del motor a cada momento.

3. Aspectos técnicos

a. Controlador PID

$$u(t) = K_p e(t) + \int_0^t K_i e(\tau) d\tau + K_d \frac{de(t)}{dt}$$

b. Método de Ziegler-Nichols

Identificar un valor estimado de ganancia proporcional y establecer un controlador en modo P ($K_i = K_d = 0$).

Incrementar la ganancia K_p lentamente hasta obtener una respuesta estable con oscilaciones periódicas.

Definir K_u como K_p y el periodo T_u como el periodo de oscilación.

Con base en la siguiente tabla, redefinir los valores de las ganancias K_p , K_i y K_d .

$$k_i = k_p / \tau_i$$

$$k_d = k_p \cdot \tau_d$$

Controlador	K_p	τ_i	τ_d
P	$0.5k_u$	∞	0
PI	$\frac{K_u}{2.2}$	$\frac{P_u}{1.2}$	0
PID	$0.6k_u$	$0.5P_u$	$\frac{P_u}{8}$

c. Bloques útiles de Simulink:

- Scope
- Step
- Gain
- Sum
- PID
- Transf Fcn

4. Procedimientos

SIMULINK



- a. Descargar el archivo correspondiente al modelo de un motor de corriente directa. El archivo se encuentra disponible en https://github.com/BGranadosR/PM_Practica4 y puede abrirse como una nueva simulación en la aplicación Simulink de Matlab.
- b. Establecer el lazo cerrado de control en torno al bloque representativo del motor. Añadir un escalón unitario como señal de entrada.

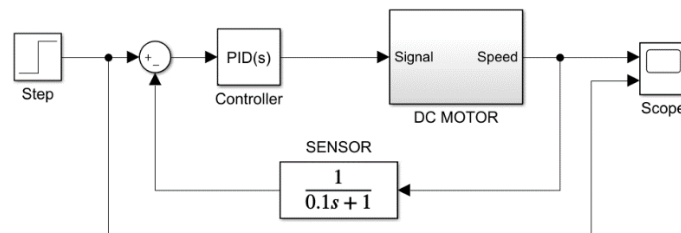


Fig. 1. Diagrama de bloques.

- c. Sintonizar e implementar un controlador tipo P
- d. Sintonizar e implementar un controlador tipo PI
- e. Sintonizar e implementar un controlador tipo PID
- f. Sustituir el bloque PID por controladores individuales y reportar sus observaciones.

CONTROLADOR PID DIGITAL



- g. Realizar el armado del circuito adecuado para controlar un motor de corriente directa mediante los módulos PWM de Arduino.
- h. Añadir al montaje un interruptor óptico a manera de encoder para determinar la velocidad angular del motor de corriente directa.

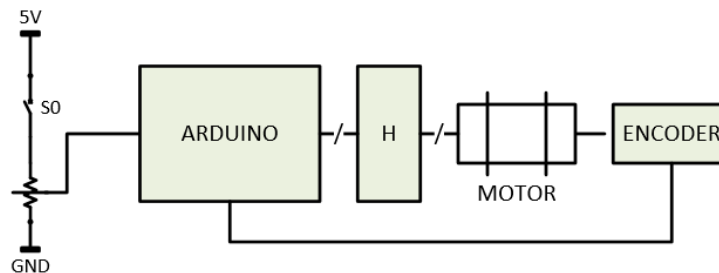


Fig. 1. Montaje de control de velocidad.

- i. Sintonizar un controlador PID por el método de Ziegler-Nichols, e implementar el mismo en código de programación tomando en consideración el rango dinámico de los canales de PWM incorporados.
- j. Activar mediante un push-button una señal escalón como velocidad de referencia y obtener un gráfico de la señal de salida (respuesta al escalón) en el dispositivo físico.
- k. Verificar gráficamente el seguimiento continuo de la velocidad de referencia.

5. Estructura del reporte

El reporte de la práctica de laboratorio debe incluir bajo el formato a dos columnas de conferencia IEEE (https://www.ieee.org/conferences_events/conferences/publishing/templates.html) lo siguiente:

Resumen (abstract), Introducción, Marco Teórico, Desarrollo, Resultados, Conclusiones y Referencias

Incluyendo los puntos resaltados durante los procedimientos de la práctica.

6. Referencias

- Arduino reference
<https://www.arduino.cc/reference/en/>
- XBee/XBee Pro S1 User's Guide
<https://www.digi.com/resources/documentation/digidocs/PDFs/90000982.pdf>
- Simulink
<https://la.mathworks.com/help/simulink/index.html>