

Reporte de la Práctica 4: Motor DC & Control

Andrea Marín Alarcón
Luis Felipe Landa Lizarralde
Miguel González Borja

Resumen—Controlar el efecto que una señal de entrada cambiante tiene sobre un motor de corriente directa presenta un problema no trivial. Con una señal cuadrada que va de 0V a 1V se utilizó un control PID (proporcional, integrador, diferenciador) para ajustar el comportamiento del motor para que se acoplara a la señal de entrada. Este PID primero se implementó en Simulink, después en Matlab y por último con amplificadores operacionales acompañados de simuladores de op amps. Se utilizó un motor de corriente directa, un potenciómetro y dos engranes para probar el control. Las simulaciones presentaron las gráficas esperadas pero a causa de errores en los amplificadores operacionales TL-081 la implementación con op amps fue incapaz de producir resultados con el motor de corriente directa

1. INTRODUCCIÓN

El control es un aspecto fundamental en cualquier sistema electromecánico. El control permite no solo que los sistemas se comporten como es deseado, sino mejora su desempeño al ser capaz de recibir retroalimentación y modificar el comportamiento de los componentes dinámicamente.

Al ver un sistema como un conjunto de señales de entrada y otro de señales de salida, se ha podido generar una teoría matemática de control que hace uso del cálculo integral y diferencial. Este cálculo permite hacer modificaciones a las señales de entrada para que las señales medidas de salida se asemejen lo más posible a las señales de salida que deseamos observar.

Con el creciente número de sistemas automáticos y que requieren retroalimentación, esta teoría de control cobra mayor importancia que nunca. Puede haber sistemas con una única señal de entrada y otra de salida que tienen soluciones de control de relativa sencillez e.g. un motor de corriente directa que controla engranes. Sin embargo, también existen sistemas con múltiples señales de entrada y salida que requieren soluciones altamente complejas e.g. el control de un vehículo para que se mantenga a una velocidad constante o hasta el de un avión que requiere una velocidad, ángulo y altura determinada.

Existen posibilidades de implementar sistemas de control digitales y analógicos. Los analógicos pueden ser implementados con op amps, pero el uso de sistemas de control digital pueden llegar a ser de mayor complejidad. Se debe procesar una señal analógica y transformarla en una digital, con procesos muestreo y cuantización el sistema se vuelve más complejo.

A continuación el documento presenta el Marco Teórico: la información necesaria para entender el funcionamiento

de las tecnologías y técnicas utilizadas en el experimento; el Desarrollo: el procedimiento seguido para la implementación del experimento; los Resultados: los datos que resultaron de seguir el procedimiento descrito en la sección de Desarrollo; las Conclusiones: detallan las observaciones y reflexiones de los integrantes del equipo sobre el experimento llevado a cabo; Rol: la participación de cada integrante del equipo; Fuentes Consultadas: la bibliografía consultada para el desarrollo del reporte.

2. MARCO TEÓRICO

En el ámbito de la ingeniería muchas veces es necesario simular el comportamiento de un sistema antes de implementarlo. MATLAB es un lenguaje de alto funcionamiento para computación técnica junto con un software desarrollado para realizar cálculos técnicos y visualización [1]. Una de las herramientas con las que cuenta MATLAB es Simulink, la cual nos permite diseñar y simular sistemas dinámicos y sistemas embebidos [2].

El objetivo de esta práctica era implementar un control PID para un motor de corriente continua, o motor DC. Estos motores son dispositivos que convierten la energía eléctrica en mecánica, provocando un movimiento rotatorio y se compone de dos partes: un estator y un rotor. El primero da soporte mecánico al motor y es donde se encuentran los polos y el segundo, generalmente es de forma cilíndrica el cual se polariza por medio de una corriente la cual le llega mediante dos escobillas [3].

Para poder controlar la velocidad del motor de manera automática se utilizan diversos tipos de controladores. En este caso se implementó un controlador PID (*Proportional Integrative Derivative*), el cual utiliza un sistema de lazo cerrado para controlar las variables del proceso, además de ser de los controladores más precisos y estables [?].

Un sistema de lazo cerrado es aquel en el que existe una retroalimentación de la señal de salida, la cual se compara con la señal deseada para determinar la diferencia entre ambas [4]. Esta diferencia se conoce como señal de error, $e(t)$, y con ella se alimenta el controlador PID el cual emite una señal de control, $u(t)$, al motor para intentar alcanzar el valor deseado (Fig. 1).

El controlador PID está compuesta de tres acciones:

- **Acción de control proporcional:** Da una salida proporcional al controlador

$$u(t) = K_p e(t)$$

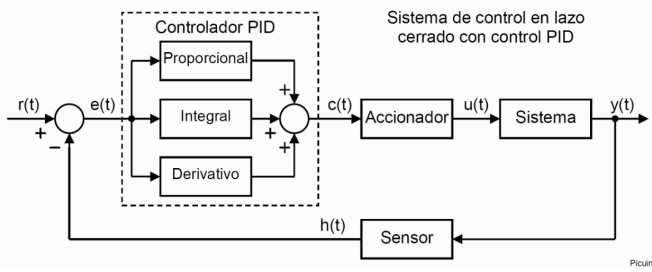


Figura 1: Diagrama de un controlador PID

donde K_p es la ganancia proporcional. El objetivo de esta acción es minimizar el error del sistema. Cuando el error es grande, la acción es grande lo que tiende a minimizar el error.

- **Acción de control integral:** Da una salida proporcional al controlador acumulado, por lo que es un control más lento

$$u(t) = K_i \int_0^t e(t) dt$$

donde K_i es la constante proporcional integral. Conforme pasa el tiempo, aunque los errores sean pequeños, se van sumando lo que ocasiona que la acción integral sea mayor. De esta manera se consigue reducir el error del sistema en régimen permanente.

- **Acción de control derivativa:** Da una salida proporcional a la derivada, o velocidad, del error

$$u(t) = K_d \frac{\delta}{\delta t} e(t)$$

de esta manera, la acción de control derivativa controla qué tan rápido cambia la velocidad del motor, aumentando la estabilidad del sistema.

Por lo tanto, función que describe al controlador PID está dada por:

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

Es necesario encontrar los valores de K_p , K_i y K_d que produzcan el comportamiento deseado, a esto se llama sintonizar el controlador. Una manera de hacerlo es mediante el método de ajuste Ziegler-Nichols.

La regla de Ziegler-Nichols sintoniza el controlador PID de forma empírica sin necesidad de conocer las ecuaciones del sistema [5]. Los únicos parámetros que utiliza son: el periodo de la frecuencia de oscilación en el límite de estabilidad y el margen de ganancia K_u para la estabilidad del ciclo. este método funciona bien cuando se trabaja con un controlador análogo, un sistema lineal, monótono, y que responde lentamente a cambios y una respuesta dominada por algo que se comporta como un retraso unipolar exponencial [6].

Para poder implementar el sistema de manera física se utilizaron Op - Amps, resistencias, capacitores y dos potenciómetros; uno para establecer el valor meta y otro para leer la salida del motor.

Los Op-Amps, o *Operational Amplifiers*, son dispositivos lineales utilizados para condicionamiento de señales, filtrar o para realizar operaciones matemáticas tales como: suma,

resta, integración y derivación. Un Op-Amp es un dispositivo con tres terminales y dos entradas con alta impedancia. Una de ellas se le conoce como entrada inversora y está marcada con un signo menos (-); y la otra se llama entrada no inversora, marcada con un signo positivo (+) (Fig. 2) [7].

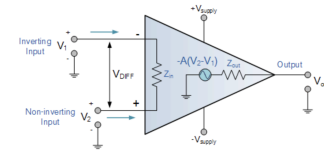


Figura 2: Circuito de un Op-Amp

3. DESARROLLO

3.1. Simulink

Se utilizó *Simulink* para crear un control PID (Figura 3), utilizando bloques amplificadores, integradores, diferenciales y sumadores. La meta del control PID era llevar un motor en 0V a 1V. Para esto, se utilizaron distintos valores para k_p , k_i y k_d para observar el comportamiento del voltaje del motor en el tiempo, como se puede observar en las Figuras 10, 11 y 12.

Después, se utilizó el método Ziegler-Nichols para obtener valores de k_p , k_i y k_d . Para esto, primero utilizo el control P, incrementando k_p hasta notar oscilaciones constantes en el voltaje del motor, como se puede ver en la Figura 13. Después, se encontraron los valores de k_p , k_i y k_d , mostrados en el Cuadro 1, a partir de k_u , el coeficiente usado para las oscilaciones, y P_u , el periodo de las oscilaciones, utilizando las siguientes formulas:

$$k_p = \frac{k_u}{1,7}$$

$$k_i = \frac{P_u}{2}$$

$$k_d = \frac{P_u}{8}$$

Finalmente, se reemplazó el bloque PID por una función de Matlab, que calculaba el control proporcional multiplicando por una constante, el integral sumando los errores, y el diferencial restando al error actual el error previo, y finalmente regresaba la suma de los tres controles. La Figura 5 muestra el diagrama del sistema utilizando esta función.

3.2. Op-amps

Para la implementación del control PID utilizando op-amps, se decidió utilizar los siguientes valores:

$$k_p = 2$$

$$k_i = 2$$

$$k_d = 0,5$$

Con base en estos valores se calcularon las resistencias y capacitores necesarios para cada op-amp. En las Figuras 6, 7, 8 y 9 se muestran los diagramas y simulaciones de

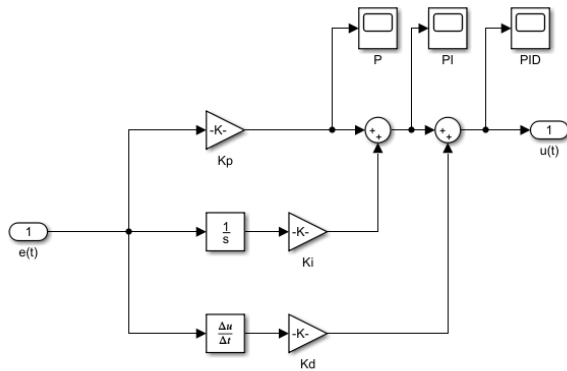


Figura 3: Diagrama del control PID

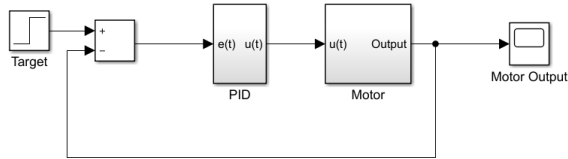


Figura 4: Diagrama del sistema utilizando PID

cada uno de estos elementos, con los valores relevantes para resistencias y capacitores de cada circuito.

Después, se conectaron 2 potenciómetros de $20k\Omega$ a una fuente de 5V, uno midiendo la rotación de los engranes conectados al motor, mientras que el otro determinaba la posición deseada. Una vez que ambos potenciómetros estaban polarizados, se utilizó un op-amp para encontrar la diferencia entre los voltajes. Este error se tomó como entrada para el control PID, cuya salida era el voltaje deseado para el motor.

4. RESULTADOS

4.1. Simulink

En las Figuras 10, 11 y 12 se pueden observar las curvas de voltaje para cada control utilizando distintos valores constantes. Para el método Ziegler-Nichols, la Figura 13 muestra las oscilaciones del voltaje para valores de k_u , el Cuadro 1 muestra los valores para k_p , k_i y k_d y la Figura 14 muestra las curvas de voltaje del motor. En cuanto al bloque

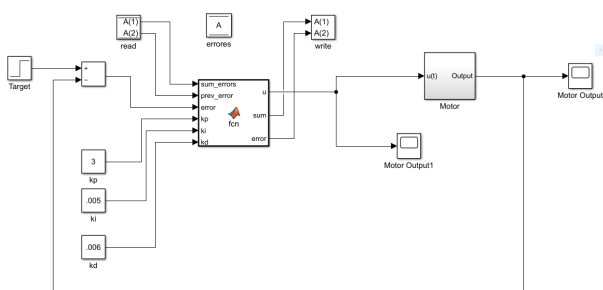


Figura 5: Diagrama del sistema utilizando una función de Matlab

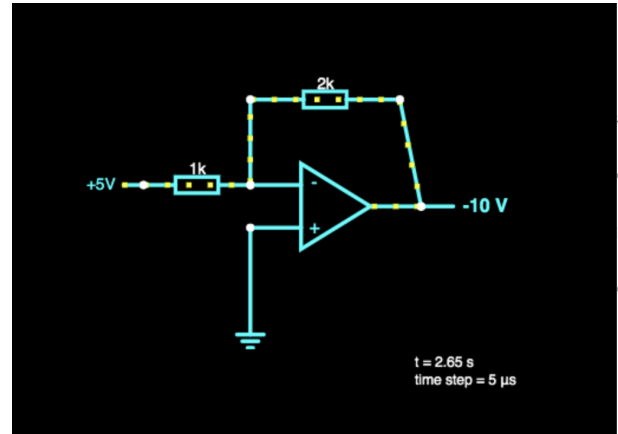


Figura 6: Simulación del op-amp amplificador

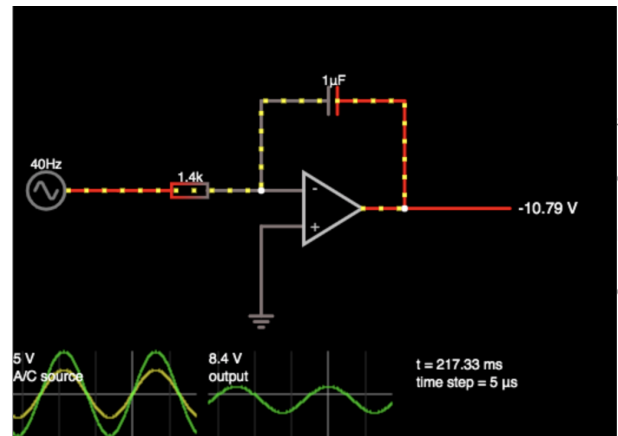


Figura 7: Simulación del op-amp integrador

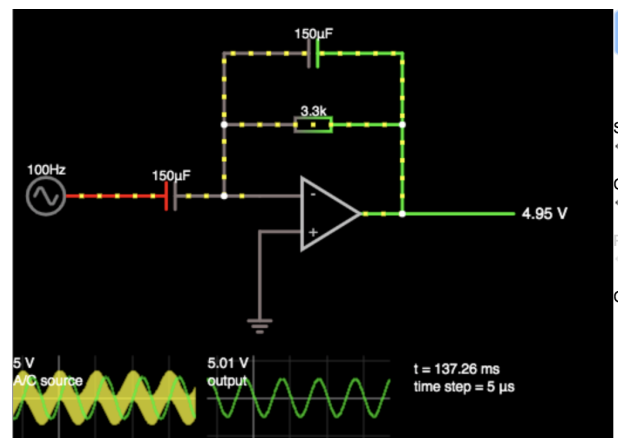


Figura 8: Simulación del op-amp diferencial

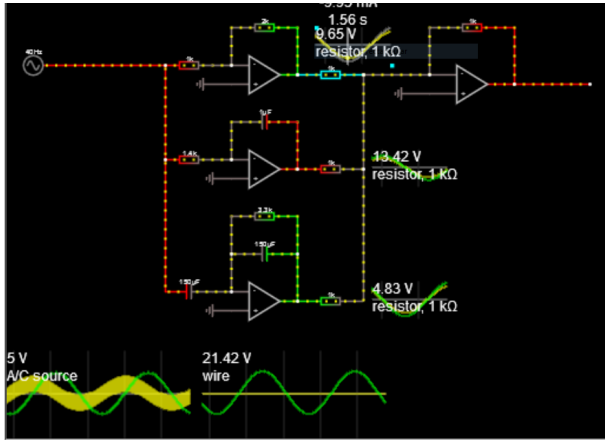


Figura 9: Simulación del control PID

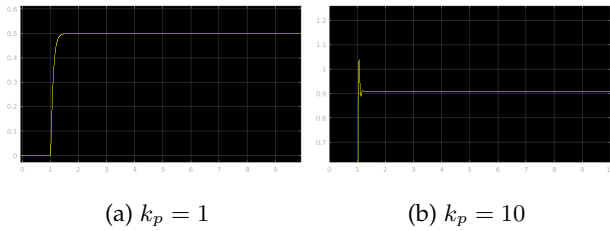


Figura 10: Curvas de voltaje utilizando el control P

de función de Matlab, las curvas reportadas para distintos valores de k_p , k_i y k_d eran iguales a las creadas por el bloque PID creado.

k_u	P_u	k_p	k_i	k_d
100	0.035	58.82	0.017	0.004
35	0.065	20.59	0.032	0.008

Cuadro 1: Valores para el método Ziegler-Nichols

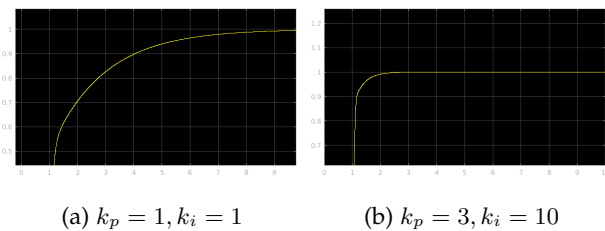


Figura 11: Curvas de voltaje utilizando el control PI

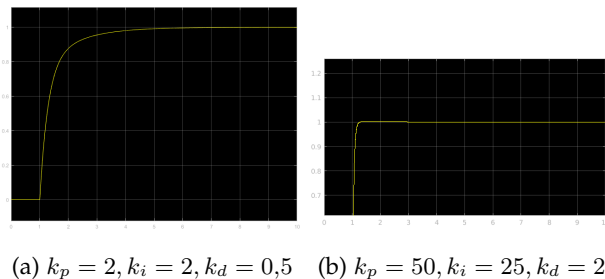


Figura 12: Curvas de voltaje utilizando el control PID

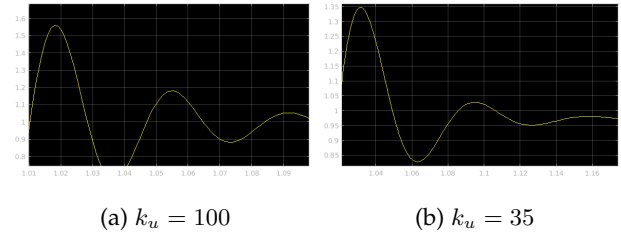


Figura 13: Curvas de voltaje con control P para primer paso de Z-N

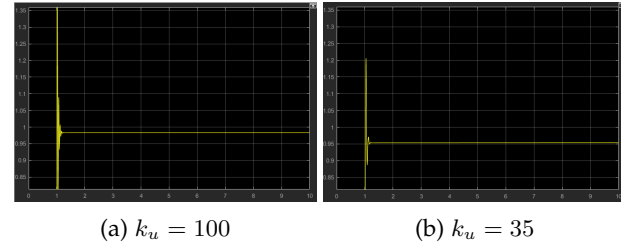


Figura 14: Curvas de voltaje con control PID, método Z-N

4.2. Op-amps

En las Figuras 15 y 16 se muestra la implementación del sistema utilizando op-amps. Debido a fallas en los módulos TL-081, el sistema fue incapaz de controlar el motor para llevar los engranes a la posición deseada. El unico op-amp que funcionó correctamente fue el restador, mostrado en la figura 16. Para probar los otros módulos, se intento reemplazarlos dentro del mismo circuito, pero de los 10 probados ninguno sirvió de manera correcta.

5. CONCLUSIONES

Si bien el uso de un circuito integrado para un control PID es mas rápido y eficiente que un control por software, este es más propenso a fallas, pues si cualquier componente deja de funcionar, el control completo se vuelve obsoleto. Por otro lado, un control PID por software da más flexibilidad en la implementación, a pesar de ser mas lento que uno integrado.

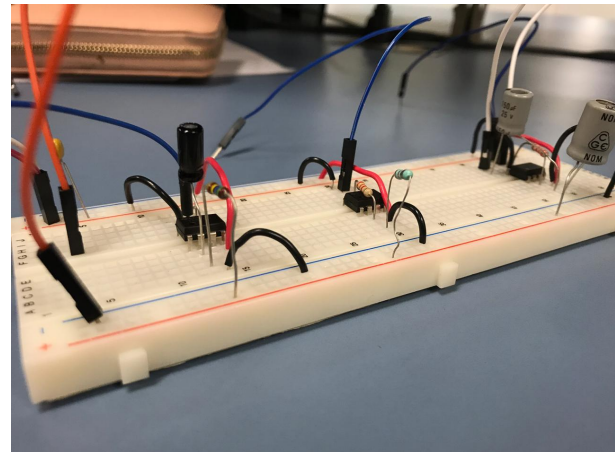


Figura 15: Op-amps para integrar (izq), amplificar (centro) y derivar (der)

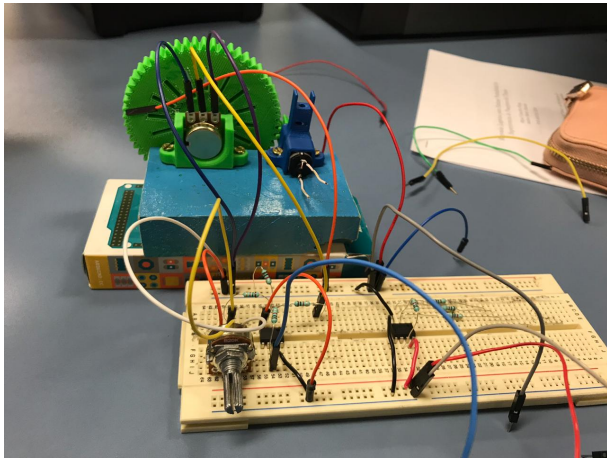


Figura 16: Circuito para comparación de voltajes y suma de controles P, I y D

Miguel González Borja

El uso de simulaciones es de gran ayuda para planificar la implementación física del control PID. Ser capaces de analizar las gráficas de voltaje de salida simulado dieron pauta a que propusiéramos valores de k_p , k_i y k_d . Sin embargo, fuimos testigos de las limitaciones que podemos esperar de los circuitos integrados. Un controlador implementado en software no termina siendo igual de susceptible a estos fallos y puede ser una solución más robusta. De igual manera, ver las múltiples implementaciones que puede haber de un control PID indica que hay circunstancias en las que cierta implementación es más apropiada que otra.

Luis Felipe Landa Lizarralde

Es importante saber utilizar diferentes herramientas de simulación tal como Simulink pues estas te ayudan a diseñar y planear de mejor manera el sistema a implementar. Todo esto evita que se cometan errores, grandes o pequeños, al momento de implementar el sistema en físico, además de que permiten identificar las posibles fallas de los mismos. No obstante, en esta práctica nos pudimos dar cuenta que las simulaciones no siempre arreglan todos los problemas, pues siempre hay que tener en cuenta las posibles fallas en el hardware al momento de implementar el circuito.

Andrea Marín Alarcón

6. ROL

Miguel Gonzalez Borja - Control PID en Simulink, implementación de circuito.

Luis Felipe Landa Lizarralde - Control PID en Simulink e implementación de circuito.

Andrea Marín Alarcón - Control de PID, simulación e implementación del circuito; marco teórico.

7. FUENTES CONSULTADAS

- [1] wikidot, "¿qué es matlab?" accessed 2019-04-11. [Online]. Available: <http://guiamatlabnoobs.wikidot.com/introduccion>
- [2] T. point, "Matlab - simulink," accessed 2019-04-11. [Online]. Available: https://www.tutorialspoint.com/matlab/matlab_simulink.htm

- [3] G. Electronics, "Motores de dc," accessed 2019-04-11. [Online]. Available: <http://www.geekbotelectronics.com/motores-de-dc/>
- [4] L. V. de Guevara, "Sistemas de control de lazo cerrado," accessed 2019-04-11. [Online]. Available: <https://makinandovelez.wordpress.com/2018/02/15/sistemas-de-control-de-lazo-cerrado/>
- [5] Picuino, "Método de ziegler-nichols," accessed 2019-04-11. [Online]. Available: <https://sites.google.com/site/picuino/ziegler-nichols>
- [6] V. Mazzone, "Controladores pid," accessed 2019-04-11. [Online]. Available: <http://www.eng.newcastle.edu.au/~jhb519/teaching/caut1/Apuntes/PID.pdf>
- [7] E. Tutorials, "Operational amplifier basics," accessed 2019-04-11. [Online]. Available: https://www.electronics-tutorials.ws/opamp/opamp_1.html