

Proyecto Final: Control Digital de un Sistema.

Autor: Fernando Flores Pérez (NUA: 909732). Universidad de Guanajuato

Palabras clave— Aproximación, Arduino, control automático, controladores, derivadas, discretización, Eagle, Euler hacia atrás, Función de transferencia, Matlab, método, PCB, PWM, sintonización, sistema, Tinkercad.

I. INTRODUCCIÓN

Las acciones básicas de control y sus combinaciones generan las diferentes leyes de control que caracterizan a los controladores, normalmente, estas vendrán expresadas como funciones de transferencia. La discretización es un proceso que transfiere funciones continuas a funciones discretas con ayuda de la transformada Z.

Para poder ingresar una función de transferencia de un controlador a una tarjeta programable como Arduino es necesario discretizar la función de transferencia de dicho controlador así podrá ser ejecutada para poder controlar un sistema, que para el caso de esta práctica el sistema es un motor DC.

Hay diferentes métodos que sirven para obtener a partir de un controlador continuo un controlador digital equivalente, para este proyecto se mostrará el método de aproximación denominado Euler hacia atrás, este método servirá para poder implementar una interfaz gráfica la cual se utiliza para la discretización de cualquier controlador.

En la actualidad los controles automáticos se han vuelto de gran importancia ya que son utilizados para controlar desde el sistema más simple, hasta el más complejo de los sistemas como el guiado de proyectiles por dar un ejemplo.

El control automático juega un papel vital en el avance de la ingeniería y la ciencia ya que los controles brindan los medios para lograr el funcionamiento óptimo de sistemas dinámicos, mejorar la calidad y abaratar los costos de producción, liberar de la complejidad de muchas rutinas de tareas manuales respectivas, etc.

Hoy en día en las modernas fábricas e instalaciones industriales, se hace cada día más necesario de disponer de sistemas de control o de mando, que permitan mejorar y optimizar una gran cantidad de procesos, en donde la sola presencia del hombre es insuficiente para gobernarlos. [1]

El objetivo de este proyecto es implementar un controlador discreto a un pequeño motor de DC para poder cambiar o ajustar su velocidad, además el circuito físico implementado para el sistema será simulado con ayuda del software Eagle a manera ilustrativa y posteriormente implementado en una PCB.

II. MARCO TEÓRICO

Métodos de Discretización de Sistemas en Tiempo

Para obtener un controlador digital a partir de uno continuo es deseable que la respuesta transitoria y la respuesta frecuencial del sistema sean equivalentes. Además, las características del controlador digital dependerán del valor de T que afectará a la respuesta del sistema.

A continuación, se describen los métodos de aproximación mencionados en la introducción:

1. Método de Euler hacia Atrás

El método aproxima la integral bajo $f(t)$ por la suma de rectángulos hacia atrás tal como se muestra en la figura 1. [1]

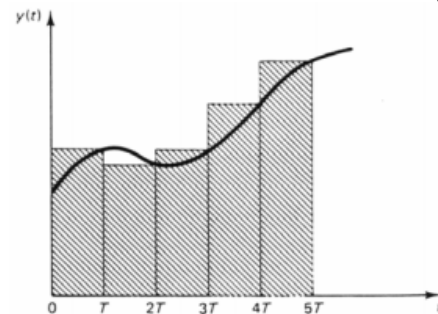


Fig. 1. Aproximación del método de Euler hacia atrás.

Donde:

$$f(x(t), y(t)) = cte = f(x(kT), y(kT))$$

En:

$$(k-1)T < t < kT$$

Obteniendo así:

$$y(kT) = y((k-1)T) + [-a \cdot y(kT) + a \cdot x(kT)] \cdot T$$

Y su función de transferencia en Z del controlador es:

$$G_c(z) = \frac{Y(z)}{X(z)} = \frac{a \cdot T}{1 - z^{-1} + a \cdot T} = \frac{a}{\frac{1 - z^{-1}}{T} + a}$$

Lo que equivale a:

$$s = \frac{1 - z^{-1}}{T} \quad (1)$$

En la figura 2 se muestra el mapeo obtenido con el método de Euler hacia atrás:

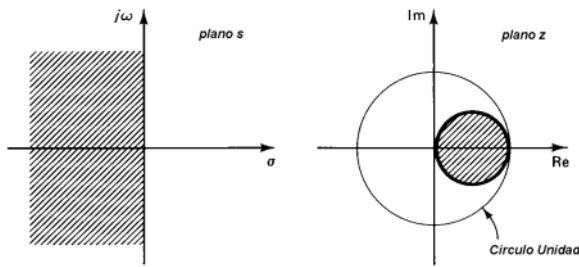


Fig. 2. Mapeo de la aproximación del método de Euler hacia atrás.

Controlador Proporcional Integral PI

Para este controlador el valor de las constantes dependerá del tipo de aproximación que se utilice y como se utilizará el método de Euler hacia atrás.

La función de transferencia para este controlador es la siguiente:

$$G_c(s) = \frac{Y(s)}{E(s)} = kp + \frac{kp}{ti} \quad (2)$$

Euler hacia atrás

Para este método en la ecuación (2) se debe sustituir el valor de s por la ecuación (1) y se pasa al dominio de Z :

$$G_c(z) = \frac{Y(z)}{E(z)} = kp + \frac{kp}{ti \left(\frac{1-z^{-1}}{T} \right)}$$

Despejando $Y(z)$:

$$Y(z) = kpE(z) + \frac{kpE(z)}{\frac{ti}{T}(1-z^{-1})}$$

$$Y(z)(1-z^{-1}) = kpE(z)(1-z^{-1}) + \frac{Tkp}{ti}E(z)$$

$$Y(z) = kpE(z) - kpE(z)z^{-1} + \frac{Tkp}{ti}E(z) + Y(z)z^{-1}$$

$$y(n) = \left(kp + \frac{Tkp}{ti} \right) e(n) - kpe(n-1) + y(n-1)$$

$$y(n) = a_0 e(n) + a_1 e(n-1) + b_0 y(n-1) \quad (3)$$

$$a_0 = kp + \frac{Tkp}{ti}$$

$$a_1 = -kp$$

$$b_0 = 1$$

Discretización de Controladores

Para obtener un controlador digital a partir de uno continuo es deseable que la respuesta transitoria y la respuesta frecuencial del sistema sean equivalentes. Además, las características del controlador digital dependerán del valor de T que afectará a la respuesta del sistema.[1]

Para poder realizar cualquier acción de control se puede crear una especie de formula general para cualquier controlador

utilizando cualquier método de aproximación que se requiera, dicha formula es la siguiente.

$$y(n) = a_0 e(n) + a_1 e(n-1) + a_2 e(n-2) + b_0 y(n-1) \quad (4)$$

En esta fórmula solo cambian los valores de a_0 , a_1 , a_2 y b_0 , esto como se menciona depende del controlador que se utilice y del método de aproximación.

Para un mejor manejo de las constantes obtenidas para el sistema a discretizar del método de Euler hacia atrás para el controlador PI, a continuación, se muestran en una tabla los valores de las constantes obtenidas a partir de la ecuación 3.

<i>Euler Hacia Atrás</i>				
Controlador	a0	a1	a2	b0
PI	$kp + \frac{Tkp}{ti}$	$-kp$	0	1

Tabla. 1. Valores de las constantes del controlador PI obtenidas para la discretización.

Diseño de Sistemas de Control Basados en el Análisis del Dominio de la Frecuencia.

El método de análisis de la respuesta en frecuencia tiene la ventaja de ayudar en gran medida a determinar las funciones de transferencia o en su defecto establecer el espacio de estados de un sistema a partir de mediciones o toma de datos en una planta.

En el diseño de controladores se usan además representaciones gráficas tales como diagramas de Bode cuyo uso se justifica en la posibilidad de representar un amplio espectro de frecuencias en un solo gráfico, o la aparición de parámetros como margen de fase o margen de ganancia, mismos que resultan cruciales para la búsqueda de soluciones óptimas que lleven a los sistemas a la estabilidad o a su vez ayuden a sintonizar diferentes controladores.

El análisis del margen de ganancia y margen de fase para el diseño de controladores supone una estrategia interesante e información crucial sobre las posibles medidas que se pueden tomar frente a la aparición de respuestas inestables, esto ya que son medidas de estabilidad relativa. Esto en otras palabras quiere decir que si algunos de los márgenes obtenidos del sistema son negativo se puede concluir que es un sistema inestable.

Además, los conceptos de margen de ganancia y fase no solo indican la estabilidad en términos absolutos, también permiten dar un margen sobre qué tan lejos está en sistema de la estabilidad o inestabilidad, esto ya que mientras más pequeños se hagan los valores del margen de ganancia y de fase más tenderá el sistema hacia la inestabilidad y viceversa. [2]

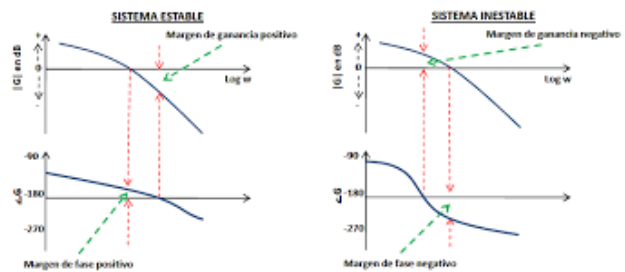


Fig. 3. Márgenes de fase y ganancia de un sistema estable e inestable.

III. METODOLOGÍA

El sistema a controlar, son dos motores de DC que fueron sujetados desde sus ejes de forma horizontal en un pequeño recuadro de madera, uno de los motores actúa como el sistema a controlar mientras que el otro será utilizado como sensor en el cual se podrá medir su velocidad.

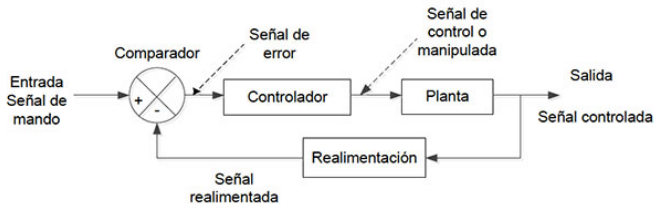


Fig. 4. Diagrama de un sistema de control discreto.

Materiales para la construcción

- 2 motores DC 3V ~ 12V
- 1 transistor TIP 31C
- 1 disipador de calor
- 1 diodo rectificador 1N4007
- 1 resistencia 1 kΩ
- 1 potenciómetro B10K
- Cables jumper
- Alambre para protoboard
- 1 tabla de 19x15cm
- Arduino UNO

Para poder apreciar las conexiones del sistema armado con ayuda de Tinkercad se creó un diagrama de las conexiones implementadas.

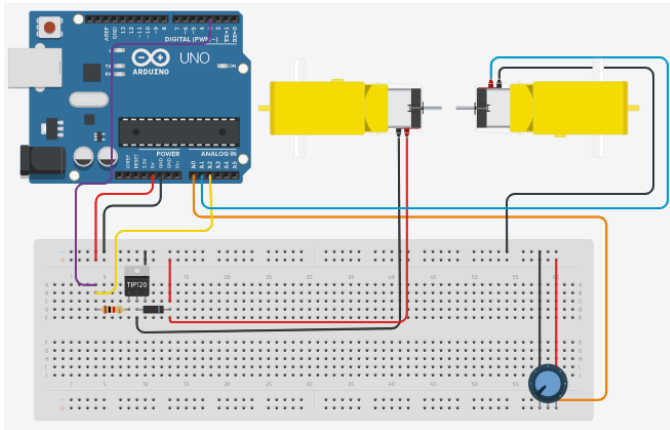


Fig. 5. Diagrama de las conexiones del sistema.

En la figura 6 se muestra una vista frontal de cómo ha quedado el sistema de control armado, el motor que es utilizado como sistema es el que se encuentra a un lado del Arduino y el otro motor es el que se utiliza como sensor.

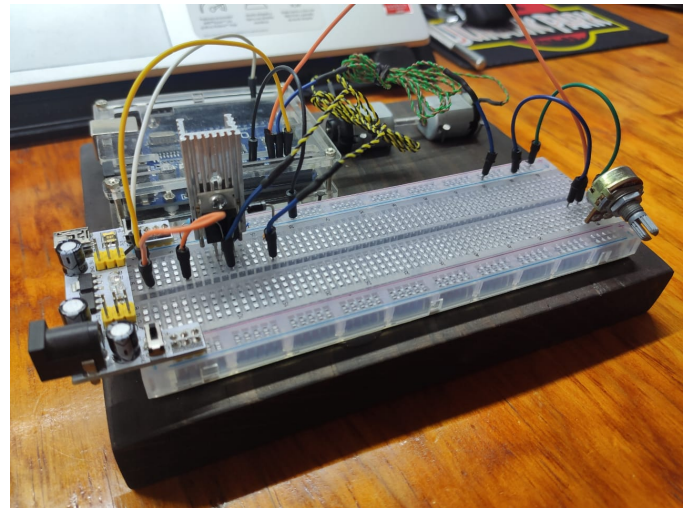


Fig. 6. Sistema de control armado de forma física (vista frontal).

En la figura 7 se muestra una vista trasera del sistema armado, con la finalidad de dar una mejor perspectiva de cómo se encuentra compuesto el sistema.

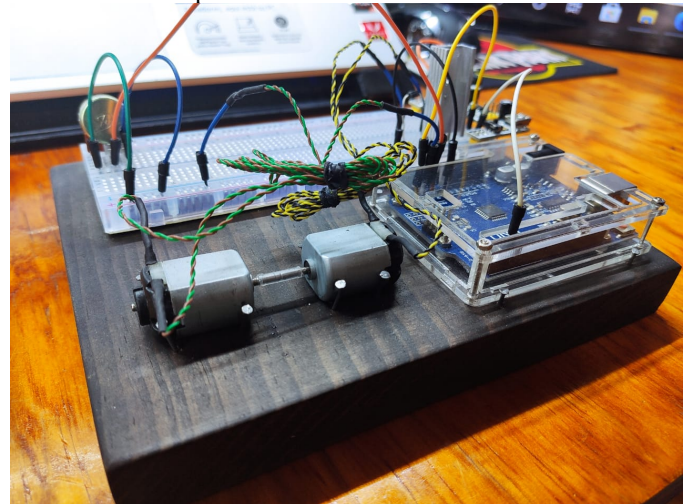


Fig. 7. Sistema de control armado de forma física (vista trasera).

Para poder controlar el sistema que se muestra en la figura 3 se debe contar primeramente con la función de transferencia del sistema.

Para obtenerla primeramente se hizo un programa en Arduino el cual se utilizó para poder leer el voltaje que le llega al sensor cuando al motor del sistema se le ingresan 5 volts, esta medición se hizo por medio de una de las entradas analógicas del Arduino y los datos obtenidos de la medición se guardaron en un archivo de texto para después con ayuda de Matlab poder encontrar la función de transferencia del sistema. En la figura 8 se pueden observar los resultados, en la gráfica se aprecian los 5 volts de referencia ingresados al sistema, la señal leída por el sistema y la señal aproximada que se obtuvo.

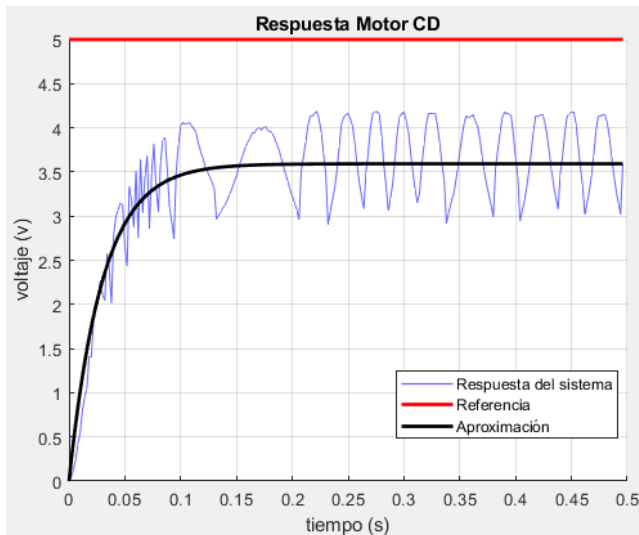


Fig. 8. Función de transferencia del sistema.

Y la función de transferencia en lazo abierto del sistema es la siguiente:

$$G(s) = \frac{0.7185}{0.03 + 0.282s} \quad (5)$$

Sintonización

Una vez obtenida la función de transferencia del sistema, este se debe sintonizar por medio de la interfaz gráfica de Matlab, el método de sintonización utilizado es el método de respuesta en frecuencia por lo que en la interfaz además de ingresar la función de transferencia se deben ingresar también un tiempo de respuesta (en segundos) y un margen de fase (en grados) y los parámetros ingresados fueron un tiempo de respuesta de 0.175 s y un margen de fase de 58°. El controlador que se utilizó para la sintonización fue el controlador proporcional integral (PI) y el resultado de la sintonización es la siguiente:

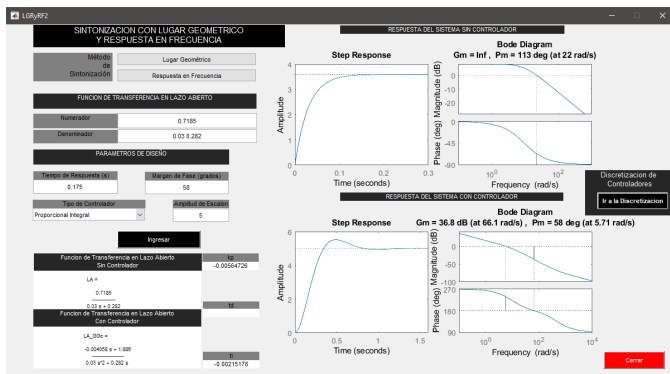


Fig. 9. Sintonización del sistema en la interfaz gráfica.

La razón por la que se eligieron todos los parámetros mencionados es porque con ellos se obtiene la mejor sintonización, es decir, un sistema bastante estable.

Se probaron previamente diferentes parámetros al igual que con los otros controladores, pero con ninguno se obtuvo una respuesta optima o buena, además se puede apreciar como la sintonización obtenida en la figura 9 cumple con lo presente en la figura 3 en la cual se muestran los márgenes de fase y magnitud correspondientes a un sistema estable.

Discretización

Ya con la sintonización optima del sistema, lo que sigue es la discretización del controlador para poder obtener los valores de las constantes de la formula (4) que como ya se mencionó, es la fórmula general para cualquier tipo de controlador y poder pasar y poder entonces pasar los valores de esas constantes al Arduino.

Para discretizar se debe presionar el botón “Ir a la discretización” el cual se encuentra en la parte central del lado derecho de la interfaz de sintonización. Al presionar el botón se despliega otra interfaz en la cual se debe ajustar un tiempo de muestro que para este caso es de 0.1s y elegir el método de discretización el cual se utilizó el método de Euler Hacia Atrás.

El resultado de lo mencionado se muestra en la siguiente imagen.



Fig. 10. Discretización del sistema en la interfaz gráfica.

Los parámetros utilizados para la discretización del controlador se eligieron por ser los que mejor respuesta presentaron, dando unos resultados bastante buenos, al igual que en la sintonización, se hicieron muchas pruebas utilizando los tres métodos de aproximación con distintos periodos de muestreo, pero al momento de observar el comportamiento en las gráficas de la respuesta del sistema, los parámetros mencionados son los que mejor respuesta presentaron.

Un dato interesante observado durante las pruebas es que, si se utiliza un periodo de muestreo grande provoca que el sistema se torne lento, responda con un cierto tiempo de retraso al momento de intentar ajustar la referencia, es por eso por lo que eligió un periodo muy pequeño para que el sistema responda adecuadamente y de forma rápida.

Por último, una vez obtenidos los valores de las constantes del sistema en el software de Arduino se hizo un programa para el controlador ya discretizado, en el cual se ingresan de forma manual los valores de las constantes.

IV. RESULTADOS

Una vez que ya se ha sintonizado, discretizado e implementado el programa en Matlab, para poder observar la respuesta del sistema en la interfaz de discretización se encuentra un botón que dice “Graficar ON/OFF” al presionarlo se inicia la comunicación serial con el Arduino y se despliega una gráfica de matlab en la cual se puede observar la respuesta del sistema, la referencia y el PWM. En la figura 10 se pueden observar la respuesta del sistema.

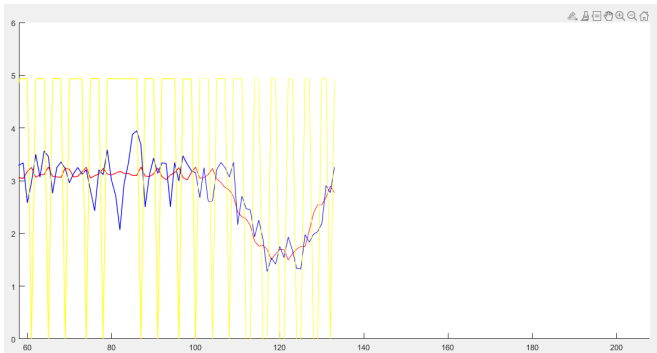


Fig. 11. Comportamiento del sistema.

En la gráfica, la señal amarilla corresponde al PWM, la señal roja a la referencia y la señal azul es la respuesta del sensor. Es apreciable como la señal del sensor cambia al mover o modificar la referencia.

También se hizo una prueba distinta, en la figura 12 se puede apreciar como el ancho de pulso aumenta aun cuando no se ha modificado la referencia, esto sucede ya que el sistema fue sometido a una perturbación, más específicamente se le aplico un poco de fricción con el dedo entre los ejes de los motores.

La razón por la cual el ancho de pulso del PWM aumenta se debe a la fricción sometida en los motores como el sistema detecta que el sensor no está llegando a la referencia hace que el PWM mande más voltaje al motor para compensar esa fricción y que el sensor llegue a la referencia.

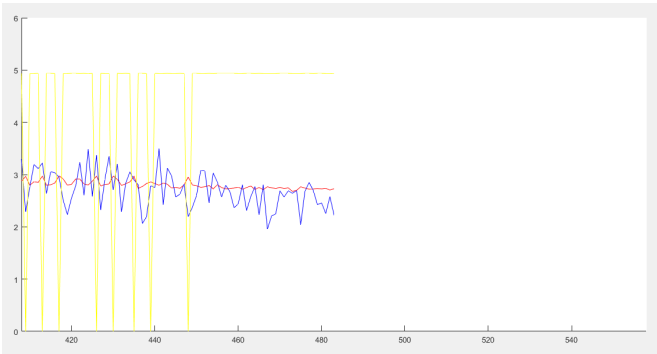


Fig. 12. Comportamiento del sistema ante una perturbación.

Esquemático del Circuito

Un esquemático es una representación grafica de un circuito electrónico o eléctrico. Muestra los diferentes componentes del circuito de manera simple y utilizando símbolos o pictogramas uniformes y las conexiones de alimentación y de señal eléctrica entre los distintos dispositivos que lo conforman.

En la figura 13 se puede apreciar el esquemático del circuito utilizado e implementado para controlar el sistema de los motores.

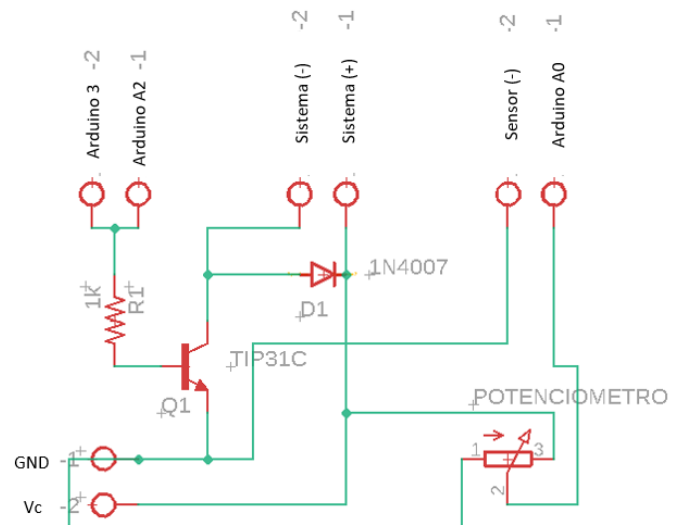


Fig. 13. Esquemático del circuito de sistema.

PCB del Circuito

Un circuito impreso (PCB) es una superficie que cuenta con pistas conductoras por las cuales fluye una corriente eléctrica y que al mismo tiempo conecta a diferentes componentes electrónicos en base a un previo diseño.

En la figura 14 se muestra el PCB del circuito utilizado para el sistema de Motores.

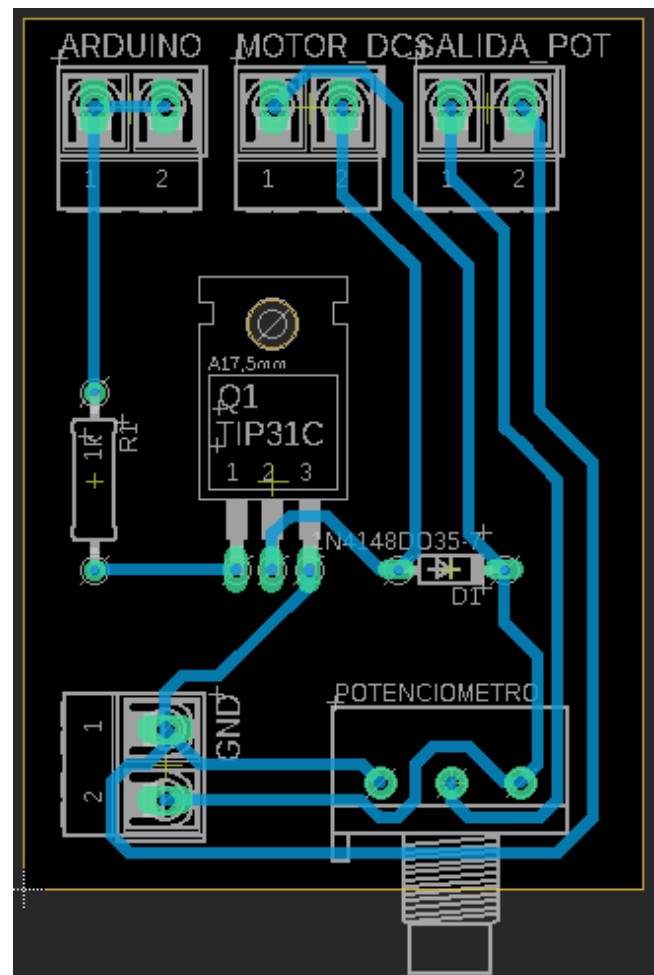


Fig. 14. Circuito impreso (PCB) del circuito de sistema.

V. CONCLUSIONES

El diseño de sistemas de control basados en la respuesta en frecuencia supone la ventaja de permitir predecir el comportamiento de un sistema al incluir un controlador, ya que juntamente con las herramientas gráficas tanto la respuesta en frecuencia de la planta y el controlador pueden ser sumadas permitiendo de forma clara implementar mecanismos de control para garantizar la estabilidad del sistema.

Una de las desventajas que se pudieron apreciar de graficar la respuesta del sistema en matlab es que en un inicio la respuesta observada es en tiempo real pero cuando se sigue graficando después de un tiempo la respuesta que se observa en la gráfica ya no es en tiempo real presenta un retraso en la respuesta muy considerable, por lo que se recomienda observar el comportamiento del sistema solo por un tiempo pequeño.

VI. REFERENCIAS

- [1]. [Www2.isa.uma.es](http://www2.isa.uma.es). 2021. Control por Discretizacion.
- [2]. Revistaenergia.cenace.org.ec. 2021. Vista de Diseño de Sistemas de Control Basados en el Análisis del Dominio en Frecuencia | Revista Técnica