



**Tecnológico de Monterrey, Campus Guadalajara**

**Ingeniería en Mecatrónica**

**Diseño de sistemas de control (Gpo 502)**

**Entregable final**

- Octavio Feria Cavargas A01640437
  - Julia Fritz A01760509
- Edgar Adolfo Ochoa Mendoza A00344644
  - Jesús Ramírez Zárate A01639933

## **Introducción**

Controlar los sistemas industriales siempre ha sido una de las búsquedas más impetuosas en la automatización de cientos de aplicaciones en amplios mercados de producción. El poder alcanzar niveles de eficiencia automatizada es el punto a lograr de cualquier proceso industrial.

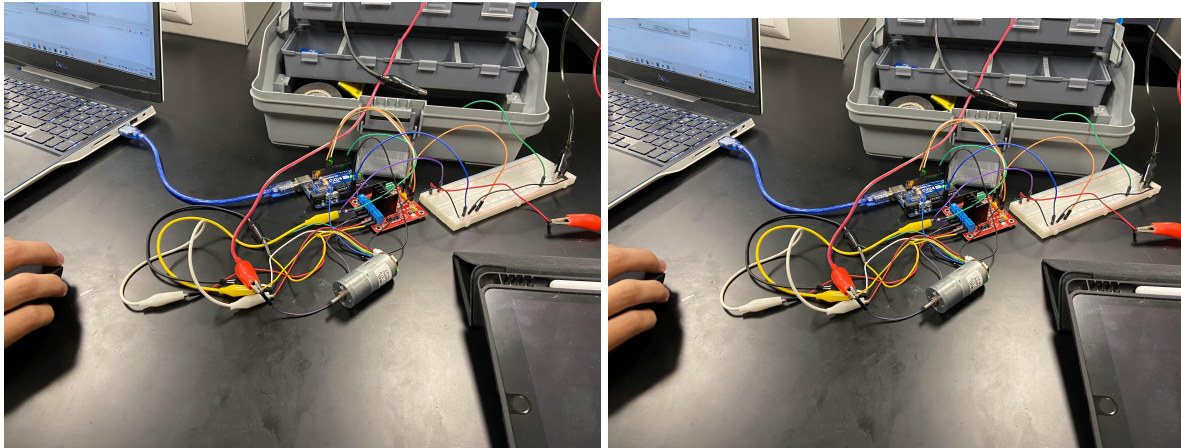
En los inicios de los sistemas de control, el pionero fue el señor Maxwell, el cual en el año de 1868 formuló un modelo matemático para el control de la máquina de vapor, inventada casi un siglo antes. En la invención y patentado de la máquina de vapor, ya existía un sistema de control, pero la aportación de Maxwell representa la primera vez que un sistema real se plasmó en la matemática.

En la actualidad, existen distintos métodos para poder controlar el comportamiento de distintos sistemas, dependiendo las necesidades y las características. En este trabajo presentaremos un controlador proporcional, integrador y derivados (PID) para un motor, midiendo las revoluciones por minuto (RPM) con el encoder integrado.

## **Desarrollo:**

Para obtener la función de transferencia del sistema a controlar se realizó un experimento en el cual se busca obtener el comportamiento del motor con el que se está trabajando, registrando los datos de entrada del motor, en este caso el voltaje de entrada y la velocidad en revoluciones por minuto (RPM) que entrega este mismo, para esto se hizo uso de diferentes materiales como lo son un “puente H” que nos permite controlar el motor, un encoder, el cual ya está integrado en el motor, que se utiliza para medir la velocidad de este mismo. , un microcontrolador para controlar el motor en conjunto con el puente H, así como poder hacer el registro de datos de entrada al motor así como de la velocidad generada con respecto a esta entrada, así mismo se hizo uso de una fuente de voltaje para poder alimentar todos los componentes utilizados. Para realizar el experimento se realizaron las conexiones adecuadas entre todos los componentes anteriormente mencionados, de manera que con ayuda del microcontrolador se pueda generar una entrada escalón al motor, además con ayuda del microcontrolador se hizo la lectura de la entrada al sistema, la velocidad del

motor y el tiempo transcurrido, en este caso se realizó una entrada escalón en la que el sistema iba de 0 a 127 cuentas del Arduino uno, las cuales eran traducidas a voltaje en la salida del microcontrolador al segundo tres de iniciar el programa.



Una vez obteniendo el registro de los datos de entrada y salida del motor, así como el tiempo transcurrido, dichos datos se exportaron a MATLAB, esto con el objetivo de poder obtener una gráfica de los datos ya mencionados para observar el comportamiento del sistema en lazo abierto y poder obtener la función de transferencia del sistema, el cual se validó con los datos experimentales, donde una vez obtenida la función de transferencia se realizó un controlador PID para controlar el motor.

Este es el código que se utilizó para definir la entrada al motor y medir la velocidad

```
#include <TimerOne.h>

//Declaración de Variables
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////

// Pin 2 para señal de encoder
const int interruptPin = 2;
//const int SeñalA = 10; //señal a puente h
//const int SeñalB = 9; //señal b puente h
//const int Enable = 11; //señal Enable puente H

// Contador: cuenta las interrupciones debido a un flanco de subida en
el encoder
long unsigned int counter = 0;

// Variable para registrar la velocidad del motor.
```

```

double VelocidadEnRPM = 0.0;
// Tiempo transcurrido
float tiempo=0.0;
// PWM
int entrada=0;

////////////////////////////////////
////////////////////////////////////
//Configuración de Pines
void setup() {

    // El Pin interruptPin es entrada, se conecta a la señal del encoder
    Canal A (tren de pulsos).
    pinMode(interruptPin, INPUT);
    pinMode(10, OUTPUT);//señal A
    pinMode(11, OUTPUT);//Señal pWM
    pinMode(7, OUTPUT);//Señla B

    digitalWrite(10, LOW);
    digitalWrite(7 HIGH);

    // Interrupción Encoder: cada vez que haya un flanco de subida en el
    interruptPin se ejecutará la
    // función counting que es una Interrupt Service Routine (ISR).

    attachInterrupt(digitalPinToInterrupt(interruptPin), ISR_counting,
    RISING);

    // Timer1: Se dispara cada 20000 microsegundos=0.02 Segundos.
    Timer1.initialize(20000);
    // Activa la interrupción: Ejecuta la función RevolucionesPorMinuto
    Timer1.attachInterrupt(ISR_RevolucionesPorMinuto);

    //Inicializar la transmisión serial bit/s (9600)
    Serial.begin(9600);
}

//Loop infinito
void loop() {

    //Entrada escalon con retraso 3s
    if (tiempo<=3)

```

```

{
    entrada=0;
}
else
{
    entrada=80;
}

analogWrite(11,entrada); //enviamos la señal escalon

}

//Funcion Contador
// Se incrementa la variable counter
//cada vez que el encoder tiene un flanco de subida

void ISR_counting() {
    //Contador
    counter++;
}

void ISR_RevolucionesPorMinuto() {
    // Velocidad en Revoluciones por minuto: esta parte
    "(counter/CuentasPorRevolucion)/Ts"
    // da las revoluciones por segundo a las que gira el encoder,
    // se multiplica por 60 segundos para obtener esta velocidad en RPM
    VelocidadEnRPM = 60*counter/(11*45*0.02); //pulsos por
    revolucio,relacion del moto reductor,tiempo

    tiempo = tiempo + 0.02;

    // Se imprime el valor del DutyCycle y de la velocidad del motor en
    RPM
    //Serial.println(VelocidadEnRPM);

    // Se reinicia el contador para que vuelva a contar hasta la
    siguiente interrupción.
    counter = 0;
}

```

Código para medir velocidad

```

void ISR_counting() {
    //Contador
    counter++;
}

void ISR_RevolucionesPorMinuto() {
    // Velocidad en Revoluciones por minuto: esta parte
    "(counter/CuentasPorRevolucion)/Ts"
    // da las revoluciones por segundo a las que gira el encoder,
    // se multiplica por 60 segundos para obtener esta velocidad en RPM
    VelocidadEnRPM = 60*counter/(11*45*0.02);    //pulsos por
    revolucio,relacion del moto reductor,tiempo

    tiempo = tiempo + 0.02;

    // Se imprime el valor del DutyCycle y de la velocidad del motor en
    RPM
    //Serial.println(VelocidadEnRPM);

    // Se reinicia el contador para que vuelva a contar hasta la
    siguiente interrupción.
    counter = 0;
}

```

## Simulación del sistema en lazo abierto:

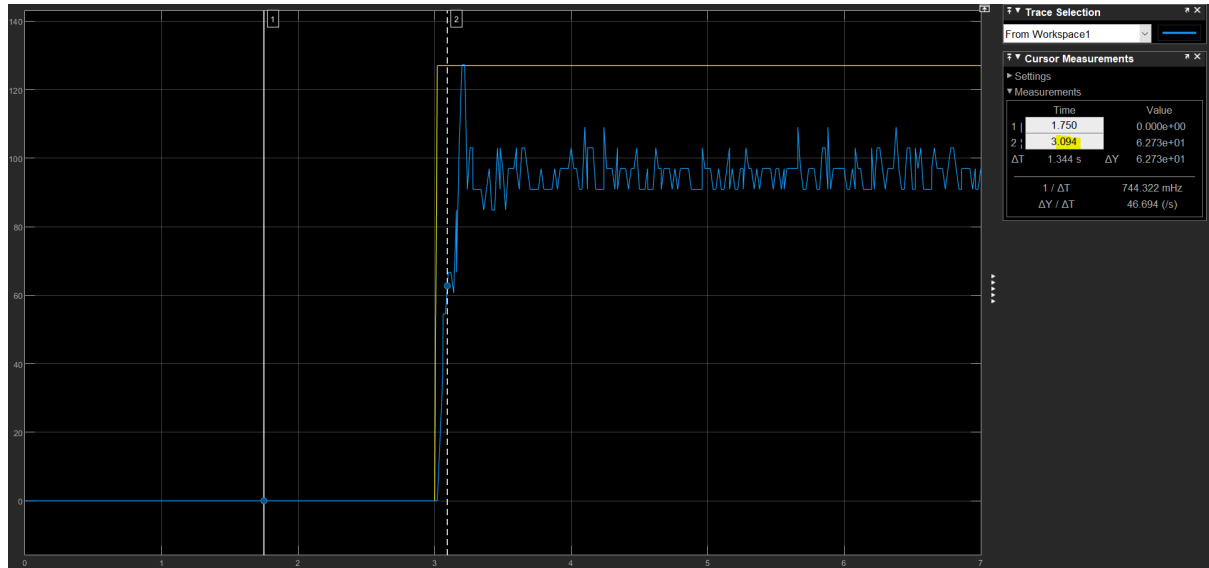


De estos datos obtenemos la ganancia de nuestro sistema.

$$K = \frac{\text{Valor final medido}}{\text{Valor final referencia}} = \frac{97}{127} = 0.7638$$

Definimos  $\tau$  como el 63% del valor final; cuando nuestro sistema alcanza ese valor habremos encontrado el tiempo de establecimiento.

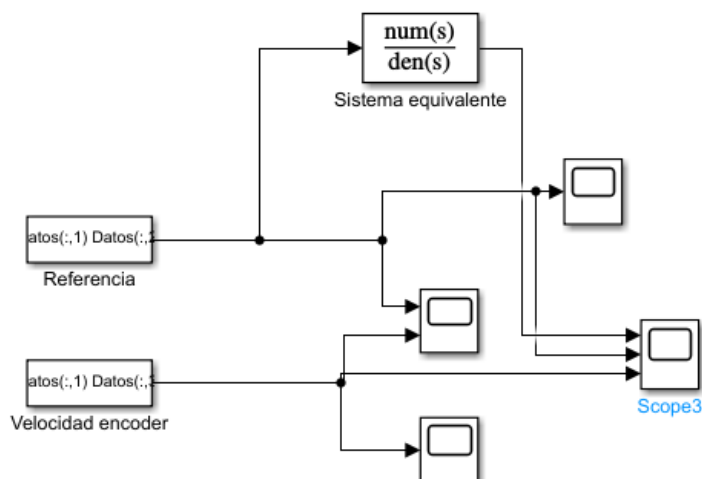
$$\tau = 97 \cdot 0.63 = 61.11$$



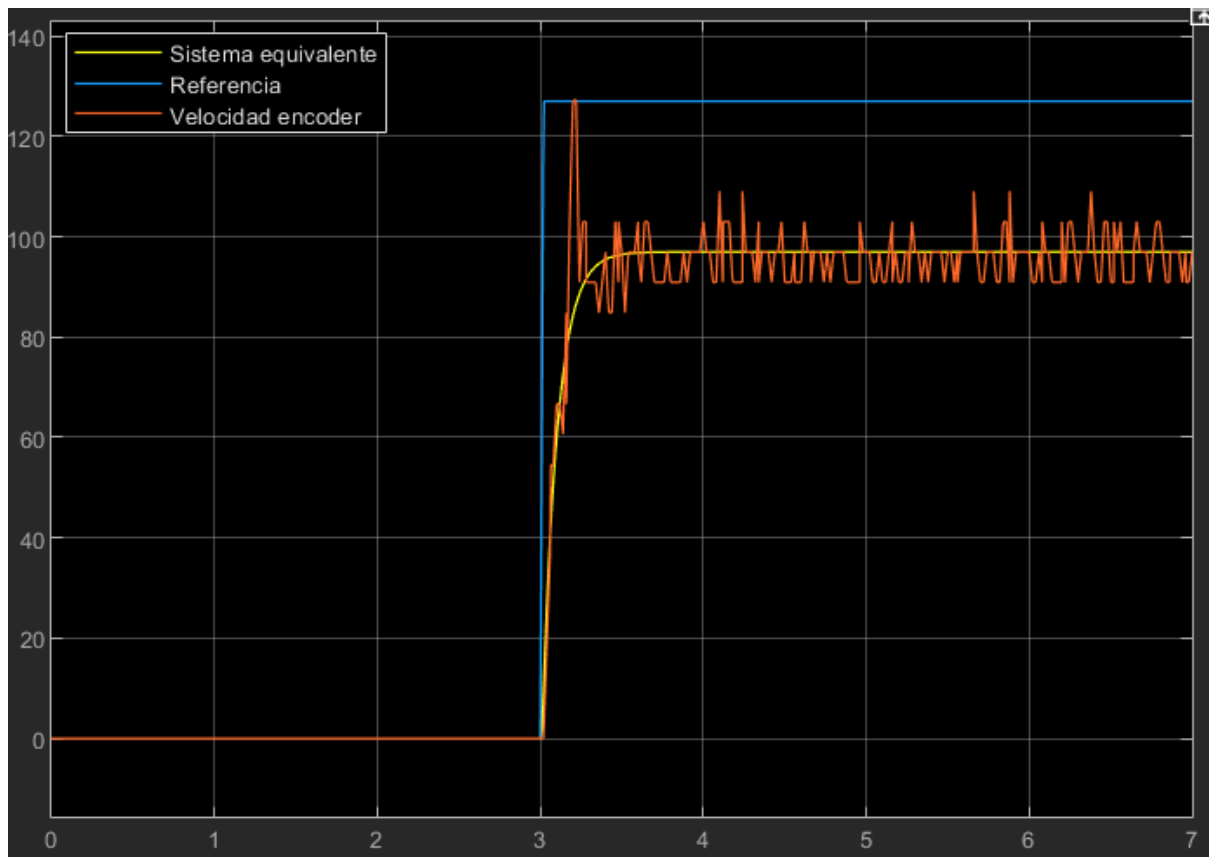
$$T_{ss} = 0.094s$$

Con estos datos podemos proponer una función de transferencia que simule nuestro sistema.

$$tf = \frac{97/127}{0.094s + 1}$$



### Validación de la función de transferencia con los datos experimentales:



En esta gráfica comprobamos que nuestro sistema si se asemeja al resultado medido en las pruebas.

**6.- Elabore una discusión en donde justifique la importancia de tener un modelo matemático del motor que está acoplado a una llanta de un vehículo como el Rover de Marte descrito en la situación problema. ¿Por qué es necesario obtener un modelo matemático del motor?**

El obtener un modelo matemático del motor es fundamental para poder entender de mejor manera cómo se comportará este bajo diversas condiciones trabajo que este pueda llegar a tener, así como para poder diseñar un controlador adecuado para cumplir con el objetivo deseado, en el caso de un vehículo Rover de marte es de suma importancia poder diseñar un controlador adecuado y preciso, ya que la tarea de este es recorrer y obtener información de un planeta que el ser humano no ha conocido aun, ademas la mayoría de los motores de este conforman una parte importante del vehículo, ya que es el sistema de desplazamiento de este, por lo que es de vital importancia tener un adecuado entendimiento de los motores utilizados, ya que el sistema de control debe ser capaz de hacer que los motores respondan adecuadamente a los obstáculos que el vehículo pueda cumplir con su objetivo de manera adecuada.



## **Conclusión**

Al realizar el experimento para entender el comportamiento del motor y obtener su función de transferencia se logró tener un mejor entendimiento de cómo es que se logran obtener los modelos matemáticos de los sistemas a partir de la observación y la experimentación, siendo esta, una etapa de vital importancia en cualquier proyecto que requiere de un sistema control, ya que a partir de esto se obtiene información de suma importancia, la cual será sumamente importante para lograr que el proyecto tenga el éxito deseado. Por otra parte durante esta práctica se logró observar y comprender aún más la importancia de las herramientas computacionales en el desarrollo de dichos modelos matemáticos y en la realización de experimentos que nos permiten conocer la caracterización de ciertos componentes o sistemas, como en este caso lo fue matlab, ya que fue utilizado como herramienta de apoyo durante todo el experimento para encontrar información del comportamiento del motor como lo es su función de transferencia, lo cual es de gran utilidad en la vida profesional para controlar una infinidad de sistemas, algo que es fundamental para un ingeniero en mecatrónica.