Emulación y simulación de un sistema electromecánico en variables de estado

Carlos Mario Paredes Valencia

Universidad Autónoma de Occidente, UAO

Santiago de Cali, Colombia

[carmar141414@hotmail.com](mailto:carmar141414@hotmail.com), [cmparedes@uao.edu.co](mailto:cmparedes@uao.edu.co)

*Abstract*—Este documento presenta la emulación y simulación de un sistema electromecánico, en su representación en variables de estados. En primera instancia, se encuentra el modelo dinámico del sistema mediante las ecuaciones diferenciales que determinan el comportamiento del mismo y a partir de estas se determina la representación en espacio de estado continuo del sistema. Posteriormente se definen las relaciones entrada salida que se desean analizar, encontrando las respectivas funciones de transferencias. Teniendo esto se discretiza debidamente el sistema, encontrando sus funciones de transferencias discretas y su representación en espacio de estado discreto. En ambas situaciones, modelo continuo y modelo discreto, se seleccionan los valores de los parámetros que garantizan la estabilidad del sistema, y se prosigue a la simulación usando la herramienta Matlab. Por último se desarrolla la solución de la ecuación de estado discreto y se implementa en la plataforma de Arduino, obteniendo respuestas iguales a las obtenidas en la simulación

Palabras claves—variables de estado, discretización, función de transferencia

# INTRODUCCION

La representación en variables de estado de un sistema permite usar información de todas las variables que interactúan en el mismo, así como la salida, para poder diseñar sistemas de control con un muy buen desempeño [1]. Este tipo de representación permite además la facilidad de tener un sistema de n-ésimo orden en un conjunto de n ecuaciones diferenciales de primer orden.

Otro aspecto a destacar de este tipo de representación, es que facilita el análisis de sistemas tipo *Multiple-input Multiple-output (MIMO),* diferencia primordial al usar funciones de transferencias. De la misma manera posibilita el análisis de sistemas lineales y no lineales así como sistemas variantes e invariantes en el tiempo [2].

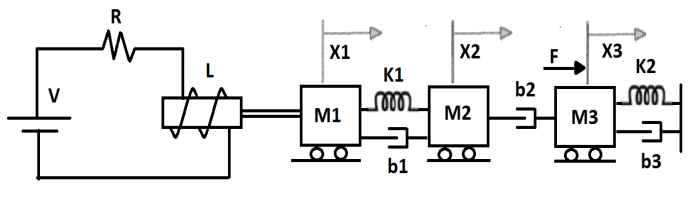
Como se mencionó con anterioridad, en el proceso de emulación, se soluciona la ecuación de estado discreto; esto se realiza mediante un método llamado Método Iterativo.

En este orden de ideas a continuación se muestra la descripción general del sistema que se va a trabajar así como el modelo matemático y posteriormente se realizara su representación en estado continuo y discreto en conjunto con sus respectivas funciones de transferencia usando la transformada de Laplace y Z. Finalmente se muestran los resultados de la simulación realizada en el software de Matlab en conjunto con la herramienta Simulink y se comparan con la implementación en Arduino, en donde se emula el comportamiento de este.

# Sistema electromecánico

El sistema sobre el cual se va a realizar la simulación y emulación se puede observar en la figura 1.

Este sistema está compuesto por una parte eléctrica y una parte mecánica



1. Sistema Electromecánico

La componente eléctrica del sistema consta de una fuente de voltaje que actúa sobre un circuito RL. En este circuito se encuentra una inductancia que genera un voltaje inducido en un electroimán en el momento que circula corriente por el circuito, proporcional a la velocidad lineal de la masa M1. Este electroimán genera una fuerza que realiza acción sobre esta masa, la cual es un elemento de la parte mecánica del sistema. La fuerza generada es directamente proporcional a la corriente que circula por el circuito.

Ahora, si se pone atención a la componente mecánica, el sistema consta de 3 masas: M1, M2 y M3 que interactúan entre ellas. Estas masas se encuentran conectadas a través de resortes y amortiguadores. Además de la fuerza generada por el electroimán, existe una fuerza externa que ejerce acción sobre la masa M3.

Teniendo esta breve descripción, a continuación se muestra el modelo dinámico del sistema, el cual permitirá encontrar la representación en espacio de estado continuo, así como las abreviaciones que se usaran en el resto del documento.

## Abreviaciones

A continuación se muestra la lista de abreviaciones que serán usadas a partir de este momento:

V: tensión de entrada del sistema.

i: corriente eléctrica que circula por el circuito RL.

L: inductancia de la parte eléctrica.

R: resistencia de la parte eléctrica.

Fe: fuerza electromotriz generada por el electroimán.

Vind: tensión inducida por electroimán.

F: fuerza efectuada sobre la masa 3.

Kv: constante de proporcionalidad para Vind.

Kf: constante de proporcionalidad para Fe.

M1: masa 1.

M2: masa 2.

M3: masa 3.

b1: factor de amortiguamiento conectado entre M1 y M2.

b2: factor de amortiguamiento conectado entre M2 y M3.

b3: factor de amortiguamiento conectado en M3.

K1: constante de resorte conectado entre M1 y M2.

K1: constante de resorte conectado entre M2 y M3.

X1: posición de M1.

X2: posición de M2.

X3: posición de M3.

x1: primera variable de estado del sistema.

x2: segunda variable de estado del sistema.

x3: tercera variable de estado del sistema.

x4: cuarta variable de estado del sistema.

x5: quinta variable de estado del sistema.

x6: sexta variable de estado del sistema.

x7: séptima variable de estado del sistema.

u1: primera entrada del sistema.

u2: segunda entrada del sistema.

y: salida del sistema.

A: matriz del sistema.

B: matriz de entrada.

C: matriz de salida.

## Unidades

El sistema de unidades que se va a manejar es el SI.

## Ecuaciones

La representación en variable de estado continuo de un sistema viene dado por (1).

(1)

La representación en estado discreto del sistema, es similar a (1), la diferencia es que en vez de derivadas temporales, se tienen adelantos x(k+1), y las matrices A, B y C se cambian por su semejante en discreto.

Antes de llegar a esta representación son necesarias las ecuaciones diferenciales que modelan el sistema. En (2), (3) y (4) se representa el comportamiento del componente eléctrico del sistema y en (5), (6), (7) se modela la parte mecánica.

(2)

(3)

(4)

(5)

(6)

(7)

Para encontrar la función de transferencia del sistema basta con resolver (8):

H(s)=C(sI-A)-1B (8)

Al tener la representación en continua, la función de transferencia discreta se encuentra con el uso de Matlab, y el adecuado tiempo de muestreo.

# REPRESENTACION EN VARIABLE DE ESTADO CONTINUO

En este apartado se realiza la representación en espacio de estado del sistema. Tomando en cuenta de (2) hasta (7), así pues, los estados de este sistema son:

x1= corriente en el circuito.

x2=posición de M1.

x3= velocidad de M1.

x4=posición de M2.

x5= velocidad de M2.

x6= posición de M3.

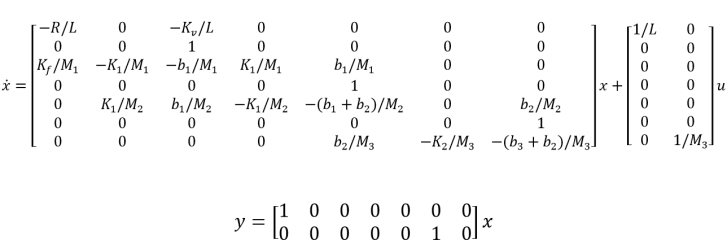
x7= velocidad de M3.

Las respectivas entradas son:

u1= V.

u2= F.

Habiendo definido los estados se procede a encontrar sus respectivas derivadas, encontrando de esta manera la representación en espacio de estado del sistema (1):



Las salidas del sistema que se van a analizar son la corriente en el circuito y la posición de la masa M3.

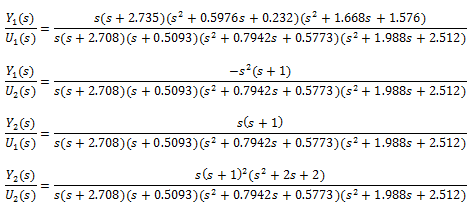
Teniendo la representación en variables de estado del sistema de manera general con sus respectivos parámetros, se procede a usar la herramienta Matlab para encontrar las respectivas funciones de transferencias.

# FUNCIONES DE TRANFERENCIA DEL SISTEMA

Para el cálculo de la función de transferencia, o en este caso la matriz de función de transferencia, se deben asignar valores a los parámetros del sistema. Asignando el valor de 1 a cada uno de los parámetros se garantiza la estabilidad del sistema, la representación de la matriz de función de transferencia es la siguiente:

En el caso se tiene una matriz 2x2 debido a las 2 entradas y 2 salidas que tiene el sistema. Recordar que la primera salida del sistema es la corriente y la segunda salida es la posición de M3, así como U1 y U2 representan las entradas V y F, respectivamente.

Cada una de estas funciones de transferencia de manera específica se muestra a continuación:



Teniendo en el momento tanto la función de transferencia como la representación en variable de estado continuo, se continúa con la discretización del sistema.

# DISCRETIZACION DEL SISTEMA

Para obtener la representación en tiempo discreto, se debe seleccionar el tiempo de muestreo más adecuado. Conociendo la ecuación característica del sistema:



Los polos del sistema en continuo son:

s1=-2.7082

s2=-0.9941 + 1.2342i

s3= -0.9941 - 1.2342i

s4=-0.3971 + 0.6478i

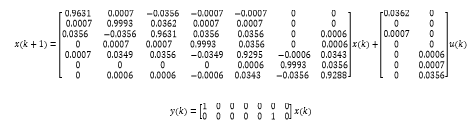
s5=-0.3971 - 0.6478i

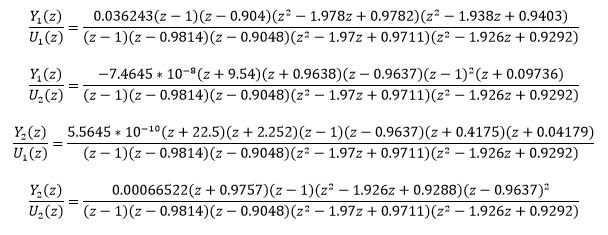
s6= -0.5093 + 0.0000i

s7=0

Se selecciona el polo con parte real más alejada del 0 para el cálculo del tiempo de muestreo, entonces el tiempo de muestreo es de 36.9ms.

De esta manera la representación en variables de estado discreta y su respectiva matriz de función de transferencia en Z son:

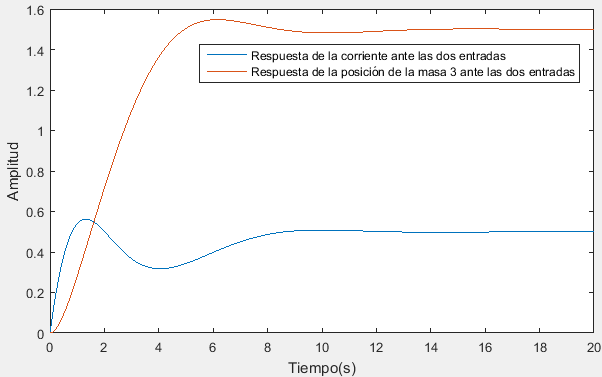




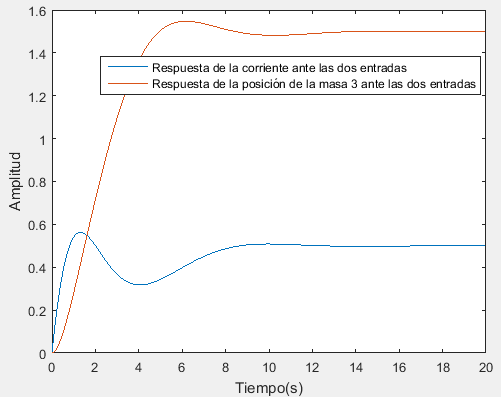
# SIMULACION Y EMULACION

En este apartado se muestran los resultados obtenidos en la simulación y emulación del sistema.

Con respecto a la simulación del sistema, se obtienen las salidas tanto en continuo como en discreto. Ambas entradas se manejan con un escalón de amplitud unitaria. En la figura 2 y 3 se puede observar que ambas salidas son congruentes, dando la misma dinámica tanto en continuo como en discreto.

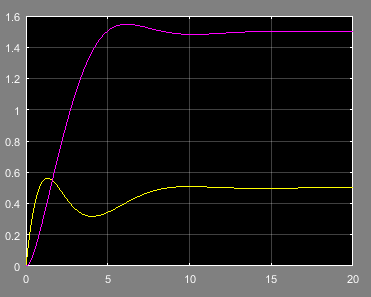


1. Salida del sistema en continuo mediante función de transferencia

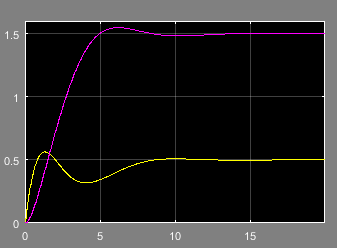


1. Salida del sistema en discreto mediante función de transferencia

Las anteriores graficas corresponden al aplicar el comando step a las funciones de transferencia adecuadas en un script de Matlab. Además de esta respuesta se obtienen por la representación en espacio de estado las mismas respuestas, estas últimas se realizan haciendo uso de la herramienta Simulink.

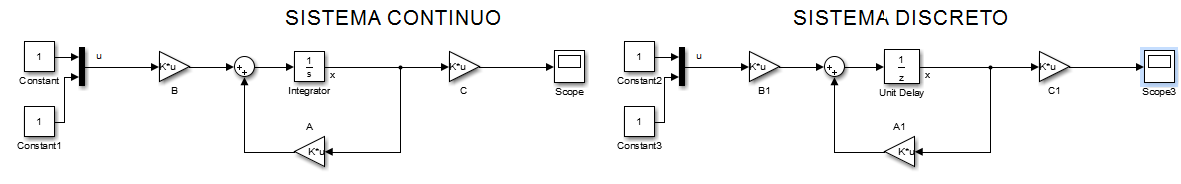


1. Salida del sistema en continuo mediante variables de estado



1. Salida del sistema en discreto mediante variables de estado

El diagrama que corresponde a las dos figuras anteriores es el siguiente:

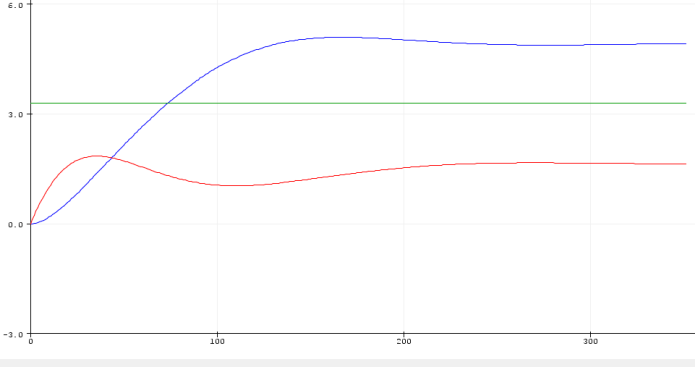


Los resultados obtenidos usando tanto la función en transferencia como la representación en espacio de estado corresponden en continuo y discreto. A continuación se muestra el procedimiento usado para emular este comportamiento en el sistema embebido de Arduino.

Se conoce que para un sistema discreto:

Conociendo las condiciones iniciales del sistema x(0)=0, se pueden estimar la evolución de cada uno de los estados usando el método iterativo:

Haciendo uso de este método se procede a la implementación en la plataforma Arduino y los resultados fueron los siguientes:



1. Salida del sistema: emulación en Arduino

En la figura anterior, se puede observar tres figuras, la línea verde corresponde a la entrada que se le aplica al sistema, ambas en este caso tienen la misma magnitud, la curva azul corresponde a la primera salida, es decir la corriente y por último la curva de tono rojo corresponde a la salida de la posición de M3.

De este proceso se puede notar que observar que los resultados fueron muy similares a los obtenidos en Matlab y Simulink.

##### CONCLUSIONES

El trabajo realizado permite poder describir, analizar y conceptualizar a través de simulaciones y emulaciones lo visto durante las clases.

Se puede observar que la solución de la representación en estados discretos se puede llevar a sistemas embebidos de manera muy fácil e inclusive se puede llevar a cabo en cualquier lenguaje de programación sin ningún inconveniente.

La selección apropiada de los parámetros en cualquier sistema, son los que en última permiten garantizar la estabilidad del mismo así como el comportamiento o la dinámica que este tendrá, dado que estos son los que establecen la ecuación característica del sistema.

La selección de un correcto tiempo de muestreo al momento de discretizar un sistema continuo, es muy importante, porque gracias a esto se puede obtener una representación confiable en tiempo discreto del sistema, de no ser así la dinámica de este podría verse afectada y hasta puede llevar a la inestabilidad.

##### REFERENCIAS

1. H. Victor, O. Ramón, C. Roberto, Control Automático, Teoría de diseño, construcción de prototipos, modelado, identificación y pruebas experimentales, Enero 2013, pp 394.
2. O. Katsuhiko, Ingeniería de Control Moderna, 5 Edición, pp17
3. Notas de clase, Análisis de sistemas lineales y no lineales.