

---

**Universidad Michoacana de San Nicolás de Hidalgo**



**Facultad de Ingeniería Eléctrica**



**“IDENTIFICACIÓN DE PARÁMETROS DE UN MOTOR  
DE CD USANDO SERIES WALSH”**

**TESIS**

**Que para obtener el Título de  
INGENIERO ELECTRICISTA**

**Presenta**

**Josué David Román Melchor.**

**Asesor**

**M.I. Isidro I. Lázaro Castillo.**

**Noviembre del 2008**

---

## Agradecimientos

A dios por su protección que tuve desde siempre y que me ayudo a luchar contra todas las adversidades que han surgido en la vida, le pido que me siga guiando por el camino del bien.

A mis padres: David Román Salgado y a M. Yolanda Melchor Luna por el apoyo incondicional y el amor, que me brindaron todos los días de mi vida, les doy gracias por haberme inculcado en mi el deseo de tener una profesión y orientado por un buen camino siempre para ser una persona de bien en la vida.

A mis hermanos Emmanuel y Yolanda Yuritzí por haberme brindado compañía y cariño en la vida, que fueron indispensables para poder desarrollarme como un buen ser humano, gracias por su valioso cariño.

A mis profesores que en el transcurso de mi formación me han brindado su amistad al igual que sus enseñanzas, son una gran herramienta para poder salir adelante en la vida.

Un agradecimiento a mi asesor: M. I. Isidro I. Lázaro Castillo, por el apoyo brindado durante el desarrollo de esta tesis.

A mis amigos: Marcelino, Gabino, Oscar, Gregorio, Miguel, Loera, al igual que todos los demás compañeros de la facultad que me ofrecieron su valiosa amistad, sincera y sin condiciones.

A la familia Espinal Solís por el apoyo incondicional que he recibido de antemano gracias por su valioso apoyo.

*A mis padres, que desde siempre me han enseñado que solo con mucho esfuerzo y dedicación se pueden obtener logros.*

*A mi esposa Nizarindany Espinal Solís y a mi hijo, que con su cariño y su amor han sido un impulso para poder realizar este trabajo.*

## **Resumen**

La identificación de sistemas de la dinámica desconocida ha producido un campo de investigación llamado “Identificación de Sistemas”, dicha identificación se enfoca a la estimación de modelos de sistemas dinámicos a partir de los datos observados o medidos.

Esta tesis esta dirigida principalmente hacia la identificación de parámetros de sistemas lineales invariantes en el tiempo. En ella se presenta la aplicación de una técnica usando series Walsh para realizar el análisis del Motor de CD el cual constituye un ejemplo de un sistema invariante. La metodología esta basada en el uso de las series ortogonales y aprovecha las propiedades ortogonales, estas se resumen en la matriz operacional de integración. Dicha metodología ha sido aplicada satisfactoriamente para resolver problemas tal es la síntesis en el dominio del tiempo en teoría de sistemas; así como la determinación de ganancia de control óptimo y el método directo de solución de problema invariante, además de la identificación paramétrica.

# Contenido

Resumen.....	iv
Lista de Figuras.....	vii
Lista de Tablas.....	ix
Lista de Símbolos y Abreviaciones.....	x

## Capítulo 1. Introducción

1.1 Antecedentes.....	2
1.2 Objetivos de tesis.....	4
1.3 Justificación.....	4
1.4 Metodología.....	5
1.5 Descripción de los Capítulos.....	6

## Capítulo 2. Técnica de Identificación Paramétrica

2.1 Introducción.....	8
2.1.1 Aproximación de Funciones.....	8
2.2 Propiedades Operacionales.....	10
2.2.1 Matriz Operacional de Integración.....	10
2.2.2 Matriz Producto y de Coeficientes.....	11
2.3 Cálculo Operacional Usando Series Walsh.....	12
2.3.1 Series Walsh.....	12
2.3.2 Ejemplo de Expansión de Funciones por Series Walsh.....	14
2.3.3 Matriz Operacional de Integración.....	16
2.3.4 Matriz Producto y de Coeficientes.....	19
2.4 Identificación Paramétrica de Sistemas Lineales Invariantes en el Tiempo.....	23
2.4.1 Redundancia Analítica.....	27

## Capítulo 3. Diseño de la Interfase Gráfica

3.1 Introducción.....	29
3.2 Programación del Instrumento Virtual.....	29

3.3	Descripción del Diseño del Panel Frontal Principal.....	33
3.4	Nodo de MATLAB Script.....	35
3.4.1	Código del Programa Principal.....	38
3.5	Diagrama de Flujo del SubVI “Gráficas”.....	40
3.5.1	Visualización de las Variables de Estado del Motor de CD.....	41

## **Capítulo 4. Pruebas y Resultados**

4.1	Introducción.....	43
4.2	Determinación de las ecuaciones de la dinámica del sistema.....	43
4.3	Pruebas para la identificación de parámetros del Motor de CD.....	46
4.3.1	Prueba 1 Entrada escalón de 24 Volts.....	46
4.3.2	Prueba 2 Aplicando de redundancia analítica.....	49
4.3.3	Prueba 3 Identificación de parámetros de otro Motor de CD.....	51
4.3.4	Prueba 4 Entrada tipo rampa.....	53
4.3.5	Prueba 5 Utilización de la Redundancia Analítica al Proceso de Identificación de Parámetros del Motor de CD del Caso Anterior.....	53

## **Capítulo 5. Conclusiones**

5.1	Conclusiones.....	55
5.2	Trabajos futuros.....	56

Apéndice A.....	57
-----------------	----

Apéndice B.....	74
-----------------	----

Referencias.....	76
------------------	----

# Lista de Figuras

1.1	Diagrama general de la metodología.....	6
2.1	Las primeras 8 funciones base de Walsh.....	13
2.2	Descomposición de la función rampa en series Walsh.....	15
2.3	Aproximación de la función rampa con 8 términos de la FWT y 3 términos de la serie Walsh.....	16
2.4	Aproximación de la función rampa con 8 términos de la serie Walsh.....	16
2.5	Funciones Walsh y sus integrales.....	17
2.6	Estructura de la matriz de integración de Walsh para $m = 32$ .....	19
2.7	Producto de funciones.....	23
2.8	Esquema de redundancia analítica.....	28
3.1	Panel Frontal.....	30
3.2	Paletas de control (panel frontal).....	31
3.3	Diagrama de Bloques.....	31
3.4	Paleta de funciones (ventana de diagrama de bloque).....	32
3.5	Diagrama de flujo del panel frontal principal.....	33
3.6	Panel principal de la interfaz diseñada.....	34
3.7	Nodo MATLAB Script.....	35
3.8	Conexión del nodo MATLAB Script con la captura de datos.....	36
3.9	Interfaz para la captura parámetros reales del Motor de CD.....	37
3.10	Teta para la obtención de los parámetros.....	37
3.11	Interfaz para la visualización de los resultados obtenidos.....	38
3.12	Diagrama de Bloques del Panel Frontal para Entrada Escalón.....	39
3.13	Diagrama de Bloques del Panel Frontal para Entrada Rampa.....	39
3.14	Diagrama de Bloques del Panel Frontal para Redundancia Analítica.....	40
3.15	Diagrama de Flujo de SubVI Gráficas.....	40
3.16	SubVI para la visualización de las gráficas del comportamiento del Motor de CD.....	41
3.17	Diagrama de Bloques del Panel Frontal para SubVI.....	41
3.18	Panel del SubVI para la visualización de las gráficas.....	42

4.1	Circuito equivalente del motor de CD.....	44
4.2	Motor implementado en la simulación.....	46
4.3	Panel frontal de resultados y entrada de datos.....	47
4.4	Parámetros reales vs estimados error existente.....	47
4.5	Panel frontal mostrando las variables de estado.....	48
4.6	Comparación de Parámetros reales y Estimados con Error 5%.....	48
4.7	Comportamiento de Variables de Estado de $m = 1024$ y $n = 128$ .....	49
4.8	Entrada escalón sumada a un tren de pulso y con entrada rampa.....	50
4.9	Estimación aplicando la redundancia analítica con $m = 1024$ y $n = 64$ .....	50
4.10	Estimación aplicando la redundancia analítica con $m = 1024$ y $n = 128$ , con error de 5%.....	51
4.11	Resultados de los parámetros estimados .....	52
4.12	Variables de estado de un Motor de CD diferente.....	52
4.13	Resultado de parámetros estimados con entrada rampa.....	53
4.14	Panel Frontal aplicación de la redundancia analítica a un Motor de CD diferente.....	54

# Lista de Tablas

Tabla 4.1	Datos tomados de la hoja del fabricante.....	46
Tabla 4.2	Parámetros de Motor CD.....	51

# Listas de Símbolos y Abreviaturas

SLVT	Sistema Lineal Variante en el Tiempo.
SLIT	Sistema Lineal Invariante en el Tiempo.
SLPT	Sistema Lineal Periódico en el Tiempo.
$T(t)$	Vector de funciones base de las series ortogonales.
$f \in \mathcal{L}^1$	Función Integrable en el intervalo $[0,1)$ .
$m$	Número de términos de las series ortogonales.
$a_i$	Coefficientes de la serie.
FFT	Transformada Rápida de Fourier.
FHT	Transformada Rápida de Haar.
FWT	Transformada Rápida de Walsh.
$P$	Matriz Operacional de Integración.
$\hat{T} \in \mathcal{L}^1$	Matriz Producto.
[C]	Matriz de Coeficientes.
$\phi_n \in \mathcal{L}^1$	Funciones base de Walsh.
$c_n$	Coefficientes de las funciones Walsh.
$\psi$	Número entero potencia de 2.
$m$	Número de términos para la transformada rápida de Walsh.
$x(t)$	Vector de estado.
$u \in \mathcal{L}^1$	Vector de entrada.
$A \in \mathcal{L}^1$	Matriz de coeficientes.
$B \in \mathcal{L}^1$	Vector de coeficientes.
$a_{ij} \in \mathcal{L}^1$	Aproximación de la matriz de coeficientes.
$b_{ij} \in \mathcal{L}^1$	Aproximación del vector de coeficientes.
$x_i \in \mathcal{L}^1$	Aproximación del vector de estado.
$u_i \in \mathcal{L}^1$	Aproximación del vector de entrada.

$x_0$	Vector de coeficientes iniciales.
$\theta$	Vector de aproximación de la matriz de coeficientes A, B y condiciones iniciales.
Z	Vector de producto de la matriz de coeficientes y la matriz de integración.
$e$	Matriz que representa una constante en Walsh para las condiciones iniciales.
$\hat{\theta}$	Matriz que contiene la aproximación de parámetros.
$V_a$	Voltaje de entrada.
$V_{emf}$	Fuerza contraelectromotriz.
$i_a$	Corriente de armadura.
$R_a$	Resistencia de armadura.
$L_a$	Inductancia de armadura.
$\omega$	Velocidad rotacional de la armadura.
$K_b$	Constante contraelectromotriz.
$K_T$	Constante de torque.
$J$	Inercia del rotor.
$f_r$	Coefficiente de fricción.
V	Voltaje.
rpm	Revoluciones por minuto.
H	Henrios.
seg	Segundos.
CD	Corriente Directa.
VI	Instrumento Virtual.
$\Omega$	Ohm.

# Capítulo 1

## Introducción

El funcionamiento de los procesos industriales ha cambiado drásticamente en las últimas décadas. Este cambio es debido principalmente a la evolución de la tecnología de la computadora. La automatización de los procesos ha originado un aumento en la productividad de algunos sectores industriales, contribuyendo con la industria a adaptarse a las demandas de mercado y aumentar su competitividad.

Para aumentar la competitividad ha sido necesario desarrollar nuevas técnicas que permitan maximizar la eficiencia de los procesos, desarrollar controladores de gran calidad, y maximizar la flexibilidad de los procesos con el menor ajuste de la máquina. Para ello es imprescindible el comportamiento dinámico del proceso, principalmente de las partes críticas [Epssem, 2008].

Para el diseño de controladores, sean de tipo convencional o avanzado, es necesario disponer de un modelo matemático preciso del proceso a controlar, por consiguiente, tanto para el desarrollo de controladores como para el método de diagnóstico y detección de fallas, se hace necesario un profundo conocimiento del sistema o proceso a controlar, que permita analizar el comportamiento del mismo. Esto se conseguirá mediante la utilización de metodologías y protocolos, tales como identificación de sistemas [Sedano, Villar, 2005].

La identificación de sistemas dinámicos conocidos y desconocidos, ha producido un campo de investigación llamado “identificación de parámetros”. Esta identificación de parámetros se basa en el modelo utilizando los datos de entrada o variables de control utilizados en sistemas lineales invariantes en tiempo (SLIT).

## 1.1 Antecedentes.

La identificación de sistemas se refiere al desarrollo de los modelos matemáticos empíricos de sistema basados en las entradas y salidas. El modelo es útil para estudiar la dinámica del sistema y proporcionar esquemas de control adecuados o bien usar esta información para el diagnóstico de falla. El interés práctico en la identificación del sistema, es manejado por dificultades asociadas con derivar modelos de los principios físicos. Construir modelos para sistemas conocidos y desconocidos es un objetivo importante de la ingeniería de control, que se basa en la construcción a partir de técnicas estadísticas, en particular en métodos de regresión lineal y no lineal.

Históricamente puede remontarse los orígenes de identificación de sistema a Gauss, y su método de mínimos cuadrados aplicado a modelos para describir el movimiento de los cuerpos celestes. La mayoría de los adelantos y el desarrollo de procedimientos sistemáticos no tuvieron lugar hasta la aparición de las computadoras digitales y la disponibilidad extendida de instrumentación y hardware a mediados del siglo XX. En la actualidad, cada vez más, el trabajo de un ingeniero consiste en la realización de modelos matemáticos de los procesos estudiados. Los procesos industriales están sujetos a severos requerimientos de eficacia, disponibilidad y seguridad. La complejidad de los mismos crece constantemente y esto hace necesario el desarrollo de herramientas de ayuda al operador humano. Entre las tareas de este tipo, destacan las tareas destinadas a la detección y diagnóstico de fallo [IEEE, 2007].

El término identificación de sistemas fue acuñado por Lofti Zadeh en 1962, como: Identificación es la determinación en base a la entrada y la salida de un sistema, dentro de una clase de sistemas especificados, al cual el sistema aprobado, es equivalente.

El enfoque de la identificación se puede realizar en función de la estructura del modelo, y del comportamiento físico o no del mismo. Se puede distinguir [Sedano, Villar, 2005]:

- (1) *Caja negra*: En la cual los parámetros del modelo no tienen una interpretación física. Un modelo basado en leyes fundamentales es muy complicado o se desconoce.

- (2) *Caja gris*: Algunas partes del sistema son modeladas basándose en principios fundamentales, y otras como una caja negra. Algunos de los parámetros del modelo pueden tener una interpretación física.
- (3) *Caja blanca*: La estructura del modelo se obtiene a partir de las leyes fundamentales. Los parámetros tienen una interpretación física.

Los métodos de identificación pueden clasificarse también en función de los modelos obtenidos, de esta forma se podría diferenciar entre: *técnicas de identificación no paramétrica*, obteniéndose modelos no paramétricos, y *técnicas de identificación paramétrica*, que conducen modelos paramétricos. Dentro de las *técnicas de identificación no paramétrica* se pueden citar como las más importantes [Epsem, 2008]:

- (1) *Análisis de la respuesta transitoria*: Se basa en la obtención de la respuesta del sistema a un impulso o a un escalón. Las señales de prueba a utilizar en este caso son un impulso o un escalón, respectivamente, y la salida registrada da el modelo correspondiente.
- (2) *Análisis de correlación*: Es un método del dominio temporal, útil para sistemas lineales y con señales continuas o discretas. Como resultado del mismo se obtiene la función de correlación entre las variables de interés.
- (3) *Técnicas frecuenciales*: Que son utilizadas directamente para estimar la respuesta frecuencial del sistema. Dentro de las técnicas se pueden diferenciar entre el análisis de Fourier y el de análisis espectral.

Dichas técnicas son aplicables en el caso de considerar procesos lineales o linealizables. Para su utilización no se debe suponer ningún tipo de estructura para el modelo y los resultados obtenidos son de tipos gráficos los cuales pueden ser más o menos fáciles de interpretar.

En el caso de métodos de identificación paramétrica, se debe tener en cuenta una cierta estructura para el modelo. Los parámetros del modelo se calculan minimizando ciertos criterios de error entre el modelo y el proceso. En general se pueden distinguir entre dos tipos de técnicas, [Epsem, 2008].

- (1) *Técnicas frecuenciales*: Las cuales minimizan el error entre la respuesta frecuencial real del proceso y la respuesta frecuencial del modelo.
- (2) *Técnicas temporales*: Las cuales minimizan el error temporal, error de predicción o error de salida, entre el modelo y el proceso. Forman parte de este grupo los métodos de identificación paramétrica clásicos.

Ambas pueden ser utilizadas tanto para la estimación de los parámetros de modelos continuos como discretos [Epsem, 2008].

## 1.2 Objetivo.

Aplicar la técnica de identificación paramétrica basada en matrices operacionales, usando series Walsh a un motor de CD, para determinar sus parámetros, utilizando MatLab®.

Realizar una interfase gráfica usando LabVIEW® y MatLab®, para presentar los resultados de la simulación.

Realizar pruebas para determinar el tipo de entradas que permiten minimizar el error en este tipo de sistemas.

## 1.3 Justificación.

En este trabajo se presenta una técnica que permite la estimación de parámetros de SLIT con parámetros concentrados, cuyo modelo dinámico es conocido. El caso de estudio principal es la estimación de parámetros de un motor de CD controlado por la armadura. La idea principal considera el uso de la expansión en series ortogonales de los datos de operación del motor obtenidos a través de simulación (corriente de armadura y velocidad).

El diagnóstico de fallas para el motor de CD mediante la identificación de los parámetros del sistema, es un método importante para un mantenimiento correctivo y predictivo.

La aplicación de la técnica de identificación de sistemas para un Motor de CD, realizar la simulación para, el mando, monitoreo de comportamiento y el margen de error estimado, así como la predicción de falla.

## 1.4 Metodología.

La identificación paramétrica de sistemas puede considerarse como el proceso de búsqueda de posibles parámetros, localizados en un sistema físico empleado.

- (1) En el primer paso consiste en obtener el modelo matemático del sistema.
- (2) En el segundo paso se pueden identificar los parámetros del modelo anterior; una vez realizado la identificación de las entradas y de las salidas del modelo, éstos pueden ser paramétricos y tienen la ventaja de estar confortados por un conjunto de coeficientes. Para obtener los coeficientes o parámetros del modelo, se utiliza la matriz operacional de integración, empleando series Walsh.

En la identificación de los parámetros se emplea la transformada rápida de Walsh y su inversa para convertir del dominio del tiempo a la frecuencia, la programación se realiza en MatLab®.

Las pruebas se desarrollan implementando una interfaz gráfica diseñada en LabVIEW® versión 7 de *Nacional Instruments*®, donde el usuario, determina los parámetros reales del motor a modelar y las condiciones iniciales, así como el tiempo de respuesta, los términos de la serie de Walsh y los términos de truncamiento de la serie de Walsh.

La Figura 1.1 muestra la metodología seguida. El primer paso consiste en la generación de los datos, se obtienen valores de corriente y velocidad del motor ante una entrada específica. Para ello se utilizan los datos reales del motor y se resuelve el modelo usando ODE45. Posteriormente los datos son aplicados al algoritmo del cual se obtienen los parámetros estimados y es posible usar redundancia analítica, con el fin de reducir el error.

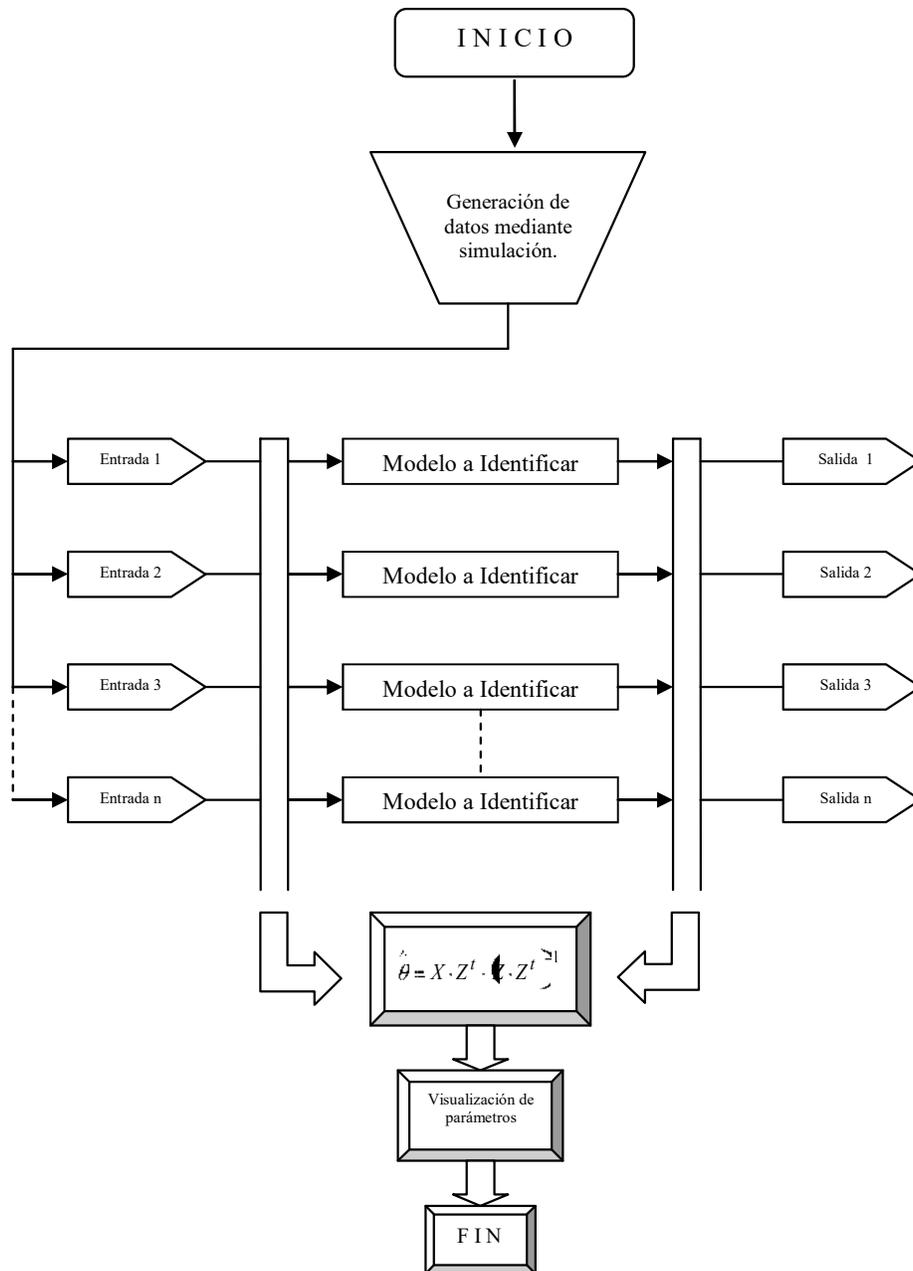


Figura 1.1 Diagrama General de la Metodología.

### 1.5 Descripción de Capítulos.

En el capítulo 1 se presenta, una breve introducción a este trabajo de investigación, la metodología de identificación de parámetros, su importancia en este trabajo, así como los objetivos y la justificación que tiene esta investigación.

En el capítulo 2 se describe la metodología del cálculo operacional, la descripción de las series Walsh y las matrices operacionales, coeficientes e integración, además se muestra la técnica de identificación de parámetros.

En el capítulo 3 se muestra el diseño de la interfaz gráfica realizada con el software LabVIEW® para la manipulación de los datos y visualización de los parámetros estimados al igual que el comportamiento del sistema.

En el capítulo 4 se presentan las pruebas para la identificación paramétrica del modelo de motor de CD, así como los resultados obtenidos y la validación de las mismas, y la comparación de las respuestas del sistema usando parámetros reales *vs* parámetros estimados, en comparación con otro motor de CD.

En el capítulo 5 se muestran las conclusiones obtenidas, se anuncian las desventajas del método empleado y las sugerencias para futuras investigaciones en este campo.

## Capítulo 2

# Técnica de Identificación Paramétrica

### 2.1 Introducción.

En las últimas décadas el uso de la expansión en series ortogonales es una alternativa útil para la aproximación y representación de funciones. Estas series también pueden ser usadas para establecer métodos algebraicos para la solución de problemas descritos por ecuaciones diferenciales, tales como el análisis de sistemas lineales invariantes y variantes en el tiempo, modelos reducidos, control óptimo y la identificación de sistemas. La técnica de identificación de sistemas obtiene un modelo matemático de la dinámica interna del sistema partiendo de la observación de las entradas y salidas. La idea básica de esta técnica es convertir la ecuación diferencial en una ecuación integral vía integración múltiple, con la finalidad de convertir la ecuación integral a una algebraica usando una matriz de integración. Subsecuentemente, las diversas señales involucradas en la ecuación integral son aproximadas por funciones bases ortogonales truncadas. A continuación se mostrarán los conceptos básicos del cálculo operacional y posteriormente se empleará la técnica de identificación paramétrica.

#### 2.1.1 Aproximación de Funciones.

La aproximación de una función, es la combinación lineal de un conjunto de funciones ortogonales y representa una herramienta de uso común en muchas áreas de análisis numérico y de procesamiento de señales. En esta sección se revisa la utilidad de la aproximación de funciones por expansión en series ortogonales.

La función base de las series ortogonales  $T(t)$  es denotada como [Lázaro, 1999]:

$$\phi(t) = [\phi_0(t) | \phi_1(t) | \dots | \phi_r(t)]^T \quad (2.1)$$

Una función  $f(t)$  que es absolutamente integrable en el intervalo  $[0, 1]$  puede ser representada por una serie ortogonal de  $n$  términos:

$$f(t) = \sum_{i=0}^{n-1} a_i \phi_i(t) = a_i^T \phi_i(t) \quad (2.2)$$

Usando la relación ortogonal de  $\phi(t)$  se tiene:

$$\int_0^1 \phi_i(t) \phi_j(t) dt = \begin{cases} 0, & i \neq j \\ 1, & i = j \end{cases} \quad (2.3)$$

Los coeficientes  $a_i$  de la serie se determinan por:

$$a_i = \int_0^1 f(t) \phi_i(t) dt \quad (2.4)$$

Ejemplos particulares de expansión en series ortogonales son:

- Series de Fourier.
- Series Walsh.
- Series Hartley.
- Series de Haar wavelets.
- Polinomios de Laguerre.
- Polinomios de Chebyshev (primera y segunda clase)
- Funciones pulso.

Las Series de Fourier son ampliamente usadas en muchos cálculos computacionales, en el análisis numérico, señales y en la aproximación de funciones, sin embargo, no son la única opción. El análisis que se hace en los apartados siguientes  $\phi(t)$  es una base ortogonal y que posee propiedades operacionales que serán discutidas posteriormente.

Todas las expansiones en series ortogonales mencionadas anteriormente pueden aproximarse a una  $f(t)$  con el mismo error arbitrario. Sin embargo, cada una de ellas puede lograrse con diferentes números de coeficientes. Una de las ventajas de las expansiones en series ortogonales, cuando se utiliza en la aproximación de funciones, es la disponibilidad de poderosos algoritmos para calcular los coeficientes de las series, como las siguientes transformadas FFT (Transformada Rápida de Fourier), FHT (Transformada Rápida de Haar), FWT (Transformada Rápida de Walsh), etc.

## 2.2 Propiedades Operacionales.

La mayoría de las expansiones en series ortogonales poseen las siguientes propiedades que permiten diseñar un cálculo operacional [Lázaro, 1999].

### 2.2.1 Matriz Operacional de Integración.

Si se integra una ecuación (2.5), ésta se puede representar por medio de una expansión de series ortogonales como el producto de una matriz de coeficientes multiplicados por la misma función base, es decir:

$$\int_0^t \phi_i(\tau) d\tau \quad (2.5)$$

$$\int_0^t \phi_i(\tau) d\tau = P_i \phi(t) \quad (2.6)$$

donde:

$$P_i = \begin{bmatrix} | & | & | & | & | \\ a_n & a_{n-1} & \dots & a_1 & a_0 \\ | & | & | & | & | \end{bmatrix} \quad (2.7)$$

Entonces la integral del vector de la función base es:

$$\int_0^t \phi(\tau) d\tau = P \phi(t) \quad (2.8)$$

donde:

$P$  es la matriz operacional de integración. Esto puede ser definido para la mayoría de las bases ortogonales  $\phi(t)$  sin embargo tiene diferentes entradas para cada caso.

Si se integro  $\phi(t)$  dos veces, se tiene que:

$$\int_0^t \int_0^\tau \phi(t)(\tau) d\tau \dots d\tau = P^2 \phi(t) \quad (2.9)$$

El resultado puede ser fácilmente generalizado en caso de k integrales múltiples como se muestra.

$$\int_0^t \dots \int_0^t \phi(t)(\tau) \underbrace{d\tau \dots d\tau}_k = P^k \phi(t) \quad (2.10)$$

### 2.2.2 Matriz Producto y de Coeficientes.

Por definición, el producto de dos vectores base y su transpuesta es llamado “Matriz Producto”  $\hat{T} \mathbf{\Phi}$ , esto es:

$$\hat{T} \mathbf{\Phi} = T \mathbf{\Phi} T^T \mathbf{\Phi} \quad (2.11)$$

donde:

$$T \mathbf{\Phi} = \begin{bmatrix} \phi_0 \mathbf{\Phi} \\ \phi_1 \mathbf{\Phi} \\ \vdots \\ \phi_{n-1} \mathbf{\Phi} \end{bmatrix} \quad (2.12)$$

En donde la forma general del producto es:

$$\hat{T} \mathbf{\Phi} = \begin{bmatrix} \phi_n \cdot \phi_n & \phi_n \cdot \phi_{n-1} & \dots \phi_n \cdot \phi_0 & \dots \phi_n \cdot \phi_{-(n-1)} & \phi_n \cdot \phi_{-n} \\ \phi_{n-1} \cdot \phi_n & \phi_{n-1} \cdot \phi_{n-1} & \dots \phi_{n-1} \cdot \phi_0 & \vdots \phi_{n-1} \cdot \phi_{-(n-1)} & \phi_{n-1} \cdot \phi_{-n} \\ \vdots & \vdots & \ddots & \vdots \vdots & \vdots \\ \phi_1 \cdot \phi_n & \phi_1 \cdot \phi_{n-1} & \dots \phi_1 \cdot \phi_0 & \dots \phi_1 \cdot \phi_{-(n-1)} & \phi_1 \cdot \phi_{-n} \\ \phi_0 \cdot \phi_n & \phi_0 \cdot \phi_{n-1} & \dots \phi_0 \cdot \phi_0 & \dots \phi_0 \cdot \phi_{-(n-1)} & \phi_0 \cdot \phi_{-n} \\ \phi_{-1} \cdot \phi_n & \phi_{-1} \cdot \phi_{n-1} & \dots \phi_{-1} \cdot \phi_0 & \dots \phi_{-1} \cdot \phi_{-(n-1)} & \phi_{-1} \cdot \phi_{-n} \\ \vdots & \vdots & \dots & \vdots & \vdots \\ \phi_{-(n-1)} \cdot \phi_n & \phi_{-(n-1)} \cdot \phi_{n-1} & \dots \phi_{-(n-1)} \cdot \phi_0 & \dots \phi_{-(n-1)} \cdot \phi_{-(n-1)} & \phi_{-(n-1)} \cdot \phi_{-n} \\ \phi_{-n} \cdot \phi_n & \phi_{-n} \cdot \phi_{n-1} & \dots \phi_{-n} \cdot \phi_0 & \dots \phi_{-n} \cdot \phi_{-(n-1)} & \phi_{-n} \cdot \phi_{-n} \end{bmatrix} \quad (2.13)$$

La forma que toma esta matriz depende de la función base seleccionada.

Durante la manipulación de las series frecuentemente se presenta el producto de dos funciones representadas en la forma de series, por lo que es necesaria una expresión concisa del producto para representar implícitamente este estado intermedio. Por lo tanto, la matriz de coeficientes de un producto de funciones se define de tal manera que se satisfaga la siguiente ecuación fundamental,

$$\mathbf{T}^T \mathbf{C} = \left[ \hat{\mathbf{T}} \mathbf{C} \right] c \quad (2.14)$$

donde  $c$  es un vector constante conocido.

Nuevamente la forma de la matriz de coeficientes  $\mathbf{T}^T$  depende de la función base seleccionada.

## 2.3 Cálculo Operacional Usando Series Walsh.

Desde que Corrington construyó tablas Walsh para resolver las ecuaciones diferenciales de orden superior, desarrollando la matriz operacional Walsh para resolver las ecuaciones de estado, el método operacional de Walsh ha sido aplicado satisfactoriamente para resolver varios problemas tales como la síntesis en el dominio del tiempo en teoría de sistemas, así como la determinación de las ganancias en control óptimo y el método directo de solución del problema variacional [Corrington, 1973].

### 2.3.1 Series Walsh.

Las funciones Walsh  $\phi_i(t)$  son un conjunto de ondas cuadradas las cuales son ortogonales. La figura 2.1 muestra las funciones de  $\phi_0(t)$  a  $\phi_7(t)$  en el orden bivalente, las cuales pueden definirse a partir de las funciones de Rademacher.

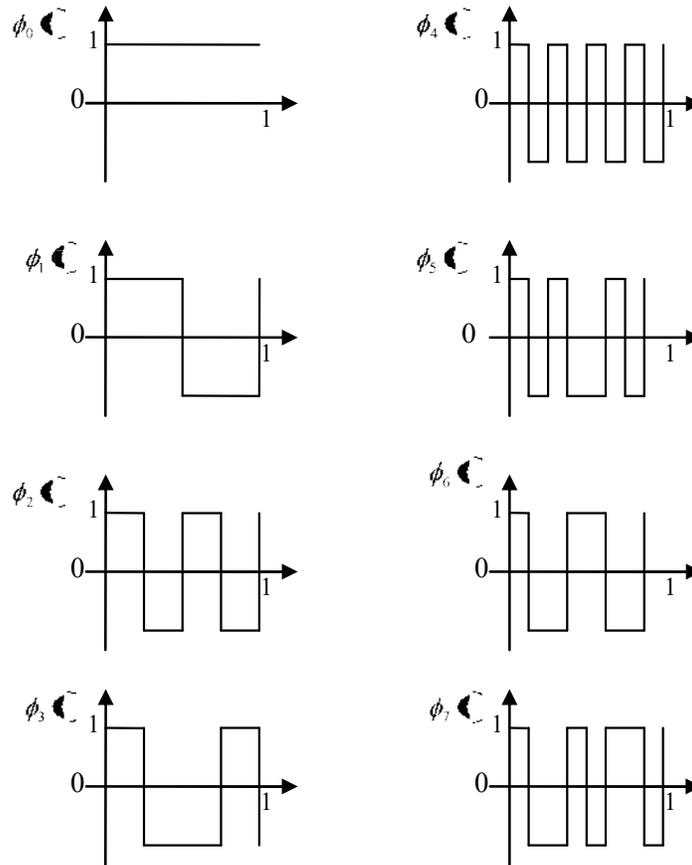


Figura 2.1 Primeras 8 funciones base de Walsh.

Una función arbitraria  $f(t)$  se puede expandir en series Walsh, si la función es absolutamente integrable en  $[0,1)$ , [Lázaro, 1999]:

$$f(t) = c_0\phi_0(t) + c_1\phi_1(t) + c_2\phi_2(t) + \dots + c_n\phi_n(t) \tag{2.15}$$

O bien

$$f(t) = \sum_{n=0}^{\infty} c_n\phi_n(t) = c^t \phi \tag{2.16}$$

Donde  $c^t$  denominan coeficientes de la serie Walsh y esta dada por:

$$c^t = \begin{bmatrix} c_0 & c_1 & c_2 & \dots & c_{n-1} \end{bmatrix}$$

Estos se determinan de manera que la siguiente integral del error cuadrático es minimizada.

$$\varepsilon = \int_0^1 \left[ f(t) - \sum_{n=0}^N c_n \phi_n(t) \right]^2 dt = 0 \quad (2.17)$$

Aplicando el límite.

$$\lim_{N \rightarrow \infty} \int_0^1 \left[ f(t) - \sum_{n=0}^N c_n \phi_n(t) \right]^2 dt = 0 \quad (2.18)$$

Multiplicando por  $\phi_n$  a ambos lados de la Ecuación (2.16) e integrando de 0 a 1 se obtiene:

$$\int_0^1 f(t) \phi_n(t) dt = \int_0^1 c_0 \phi_0(t) \phi_n(t) dt + \int_0^1 c_1 \phi_1(t) \phi_n(t) dt + \dots + \int_0^1 c_n \phi_n(t)^2 dt + \dots \quad (2.19)$$

Cada término del lado derecho de la Ecuación (2.19) es igual a cero, debido a la propiedad ortogonal dada por (2.3), excepto para el término cuadrático de  $\phi_n(t)$  el cual es uno, por lo tanto:

$$c_n = \int_0^1 f(t) \phi_n(t) dt \quad (2.20)$$

### 2.3.2 Ejemplos de Expansión de Funciones por Series Walsh.

#### Función Rampa

Por ejemplo, utilizando series Walsh se aplica la función siguiente:

$$f(t) = t \quad (2.21)$$

Realizando la aplicación en series Walsh:

$$c_0 = \int_0^1 t \phi_0(t) dt = \frac{1}{2}$$

$$c_1 = \int_0^1 t \phi_1(t) dt = -\frac{1}{4}$$

$$c_2 = \int_0^1 t \phi_2(t) dt = -\frac{1}{8}$$

$$c_3 = \int_0^1 t \phi_3(t) dt = 0$$

Por lo tanto la Ecuación (2.16),

$$f(t) = \frac{1}{2}\phi_0(t) - \frac{1}{4}\phi_1(t) - \frac{1}{8}\phi_2(t) - \frac{1}{16}\phi_4(t) + \dots \quad (2.22)$$

La suma parcial anterior de la forma de onda tipo escalera que aproxima a la función  $f(t) = t$  se presenta en la Figura 2.2.

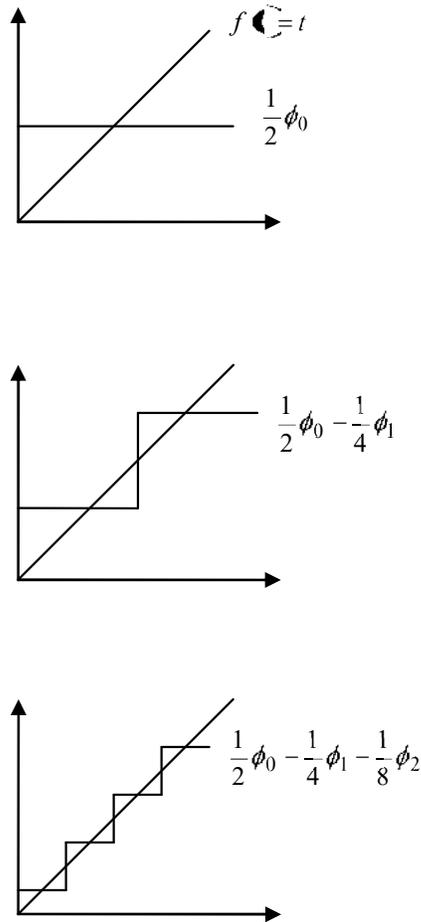


Figura 2.2 Descomposición de la función rampa en series Walsh.

En las Figuras 2.3 y 2.4 se muestra la aproximación de la función rampa para  $m = 8$  puntos de la transformada de Walsh, tomando  $n = 3$  y  $n = 8$  términos respectivamente en la serie Walsh, las cuales fueron simuladas en MatLab®.

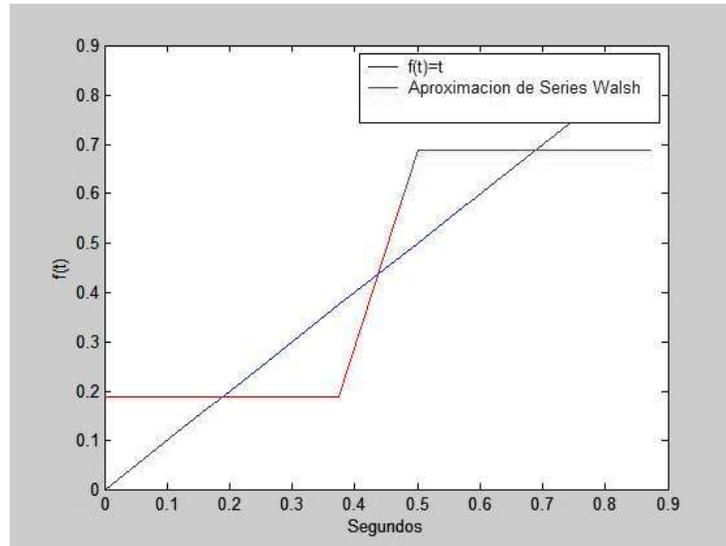


Figura 2.3 Aproximación de la función rampa con 8 términos de la FWT y 3 términos de la serie Walsh.

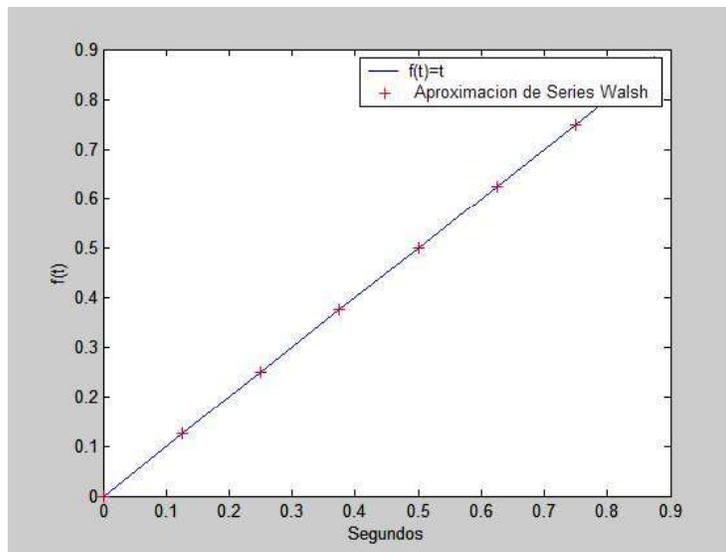


Figura 2.4 Aproximación de la función rampa con 8 términos de la serie Walsh.

### 2.3.3 Matriz Operacional de Integración.

Debido a que las funciones Walsh son un conjunto de ondas rectangulares, sus integrales son ondas triangulares, la Figura 2.5 muestra la integral de las 8 primeras

funciones Walsh. La integral de las funciones Walsh puede ser expresada en funciones Walsh y cada una de ellas puede ser evaluada como se mostró en la sección anterior. Por lo tanto, se pueden escribir la relación entre las funciones Walsh y sus integrales de la forma matricial, para el caso de 4 funciones base se tienen,

$$\begin{bmatrix} \int \phi_0 dt \\ \int \phi_1 dt \\ \int \phi_2 dt \\ \int \phi_3 dt \end{bmatrix} \cong \begin{bmatrix} \frac{1}{2} & -\frac{1}{4} & -\frac{1}{8} & 0 \\ \frac{1}{4} & 0 & 0 & -\frac{1}{8} \\ \frac{1}{8} & 0 & 0 & 0 \\ 0 & \frac{1}{8} & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix} \quad (2.23)$$

De forma compacta se tiene:

$$\int \phi_4(t) dt = P_{4 \times 4} \cdot \phi_4 \quad (2.24)$$

Donde  $P$  es la Matriz Operacional de Integración.  $P$  relaciona las funciones Walsh y sus integrales. Se selecciona como una matriz cuadrada por conveniencia computacional y su dimensión depende del número de componentes elegidos. La Ecuación (2.24) es una ecuación aproximada; su aproximación depende del número de términos tomados, para el caso de 8 funciones base, la integral de dichas funciones se muestra en la Figura 2.5.

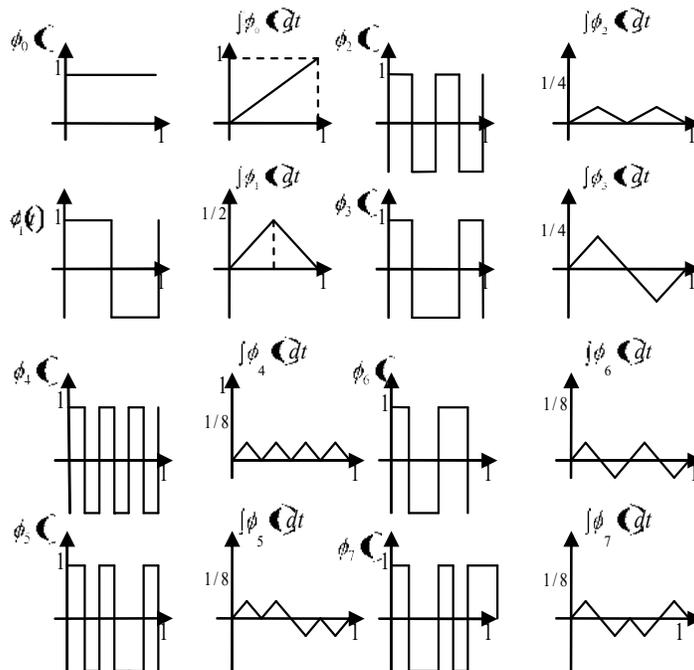


Figura 2.5 Funciones Walsh y sus integrales.

Expresando el resultado en forma matricial, se tiene,

$$\begin{bmatrix} \int \phi_0 dt \\ \int \phi_1 dt \\ \int \phi_2 dt \\ \int \phi_3 dt \\ \int \phi_4 dt \\ \int \phi_5 dt \\ \int \phi_6 dt \\ \int \phi_7 dt \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{4} & -\frac{1}{8} & 0 & -\frac{1}{16} & 0 & 0 & 0 \\ \frac{1}{4} & 0 & 0 & -\frac{1}{8} & 0 & -\frac{1}{16} & 0 & 0 \\ \frac{1}{8} & 0 & 0 & 0 & 0 & 0 & -\frac{1}{16} & 0 \\ 0 & \frac{1}{8} & 0 & 0 & 0 & 0 & 0 & -\frac{1}{16} \\ \frac{1}{16} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{16} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{16} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{16} & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \\ \phi_6 \\ \phi_7 \end{bmatrix} \quad (2.25)$$

o bien

$$\int \phi_8(t) dt = P_{8 \times 8} \cdot \phi_8 \quad (2.26)$$

El subíndice indica la dimensión tomada. Es preferible tomar  $2^{\nu}$ , donde  $\nu$  es un entero, como dimensión. Note que el n-ésimo renglón de la matriz muestra los coeficientes de las series Walsh de la función  $\int \phi_{i-1} dt$ ,  $i = 1, 2, \dots$ , por ejemplo, el primer renglón de la Ecuación (2.25) produce los coeficientes Walsh de  $\int \phi_0 dt$ . El octavo coeficiente puede encontrarse de los 8 términos de la Ecuación (2.22).

La matriz cuadrada  $P$  realiza una integración y es denominada matriz operacional para la integración entera. Es de interés mencionar que si  $P$  es simétrica, y si es dividida en cuatro partes iguales como se muestra a continuación, la parte superior izquierda de  $P_{(8 \times 8)}$  es idéntica a  $P_{(4 \times 4)}$ ; y la parte superior izquierda de  $P_{(4 \times 4)}$  es  $P_{(2 \times 2)}$ . Por lo tanto la regularidad de la estructura de la matriz permite generalizar la matriz  $P$ .

Escribiendo la matriz operacional generalizada para la integración como:

$$P_{(n \times m)} = \begin{bmatrix} \frac{1}{2} & \vdots & \vdots & -\frac{2}{m} I\left(\frac{m}{8}\right) & \vdots & -\frac{1}{m} I\left(\frac{m}{4}\right) & \vdots & -\frac{1}{2m} I\left(\frac{m}{2}\right) \\ \dots & \dots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \dots & \vdots \\ \dots & \dots \\ \frac{2}{m} I\left(\frac{m}{8}\right) & \vdots & \vdots & 0\left(\frac{m}{8}\right) & \vdots & \vdots & \vdots & \vdots \\ \dots & \dots \\ \frac{1}{m} I\left(\frac{m}{4}\right) & \vdots & \vdots & \vdots & 0\left(\frac{m}{4}\right) & \vdots & \vdots & \vdots \\ \dots & \dots \\ \frac{1}{2m} I\left(\frac{m}{2}\right) & \dots & \dots & \dots & \vdots & \vdots & 0\left(\frac{m}{2}\right) & \vdots \end{bmatrix} \quad (2.27)$$

La matriz operacional anterior realiza la integración para operaciones con enteros. Por lo tanto, puede usarse para resolver ecuaciones diferenciales lineales de manera efectiva.

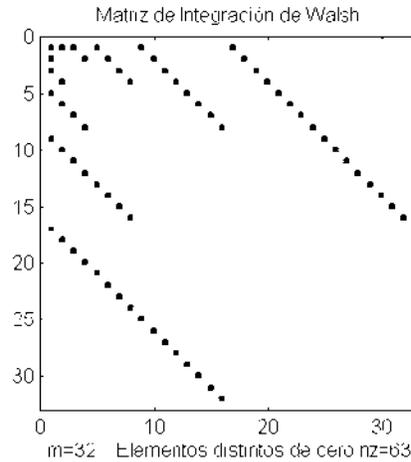


Figura 2.6 Estructura de la matriz de integración de Walsh para m = 32.

La Figura 2.6 muestra la estructura de una matriz de integración cuyo orden es de 32x32, en donde se puede apreciar claramente la dispersidad de ésta, ya que solamente 63 términos de la matriz son distintos de cero.

**2.3.4 Matrices Producto y de coeficientes.**

De acuerdo con la definición de la matriz producto dada por (2.11) para las series Walsh si se toman las dos primeras funciones base se tiene,

$$T = \begin{bmatrix} \phi_0 & \phi_1 \\ \phi_1 & \phi_0 \end{bmatrix} \tag{2.28}$$

Entonces al aplicar la Ecuación (2.28) se tiene,

$$T = \begin{bmatrix} \phi_0 & \phi_0 & \phi_0 & \phi_1 \\ \phi_1 & \phi_0 & \phi_1 & \phi_1 \end{bmatrix} \tag{2.29}$$

De la definición de funciones Walsh dada en la Figura 2.1, se puede obtener,

$$T = \begin{bmatrix} \phi_0 & \phi_1 \\ \phi_1 & \phi_0 \end{bmatrix} \tag{2.30}$$

De manera equivalente al tomar 4 funciones base se tiene,

$$\hat{T} \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix} = \begin{bmatrix} \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \\ \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \\ \dots & \dots & \vdots & \dots & \dots \\ \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \\ \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \end{bmatrix} \quad (2.31)$$

Por lo tanto para el caso de 8 funciones, se tiene una matriz de funciones base como la siguiente:

$$\hat{T} \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} = \begin{bmatrix} \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_4 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_5 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_6 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_7 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \\ \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_5 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_4 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_7 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_6 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \\ \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_6 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_7 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_4 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_5 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \\ \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_7 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_6 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_5 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_4 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \\ \dots & \dots & \dots & \dots & \vdots & \dots & \dots & \dots & \dots \\ \phi_4 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_5 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_6 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_7 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \\ \phi_5 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_4 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_7 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_6 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \\ \phi_6 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_7 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_4 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_5 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \\ \phi_7 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_6 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_5 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_4 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \vdots & \phi_3 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_2 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_1 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} & \phi_0 \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \end{bmatrix} \quad (2.32)$$

De tal manera que la forma general de la matriz producto es:

$$\hat{T} \begin{pmatrix} \cdot \\ \cdot \\ \cdot \\ \cdot \end{pmatrix}_{(n \times m)} = \begin{bmatrix} \hat{T} \begin{pmatrix} \cdot \\ \cdot \end{pmatrix}_{\left(\frac{m}{2} \times \frac{m}{2}\right)} & \hat{T} \begin{pmatrix} \cdot \\ \cdot \end{pmatrix}_{\left(\frac{m}{2} \times \frac{m}{2}\right)} \\ \hat{T} \begin{pmatrix} \cdot \\ \cdot \end{pmatrix}_{\left(\frac{m}{2} \times \frac{m}{2}\right)} & \hat{T} \begin{pmatrix} \cdot \\ \cdot \end{pmatrix}_{\left(\frac{m}{2} \times \frac{m}{2}\right)} \end{bmatrix} \quad (2.33)$$

Donde  $\hat{T} \begin{pmatrix} \cdot \\ \cdot \end{pmatrix}_{\left(\frac{m}{2} \times \frac{m}{2}\right)} = \hat{T} \begin{pmatrix} \cdot \\ \cdot \end{pmatrix}_{\left(\frac{m}{2} \times \frac{m}{2}\right)}$  con subíndice de cada elemento incrementado por  $+\frac{m}{2}$ .

Es conocido que una función la cual es absolutamente integrable en el intervalo  $[0,1)$  puede ser aproximada por una serie Walsh de m términos.

$$f \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} = \sum_{i=0}^{m-1} c_i \phi_i \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} = c^T T \begin{pmatrix} \cdot \\ \cdot \end{pmatrix} \quad (2.34)$$

donde:

$$c_i = \int_0^1 f(\varphi_i) dt \tag{2.35}$$

El vector de coeficientes  $c$  de la Ecuación (2.33) es:

$$c^T = [c_0 \ c_1 \ \dots \ c_{m-1}] \tag{2.36}$$

La matriz de coeficientes  $P$  correspondiente al vector  $c$  de la Ecuación (2.36) es definida como:

$$P_{2 \times 2} = \begin{bmatrix} c_0 & c_1 \\ c_1 & c_0 \end{bmatrix} \tag{2.37}$$

Para los coeficientes de 4 términos se tiene la siguiente ecuación:

$$P_{4 \times 4} = \begin{bmatrix} c_0 & c_1 & \vdots & c_2 & c_3 \\ c_1 & c_0 & \vdots & c_3 & c_2 \\ \dots & \dots & \vdots & \dots & \dots \\ c_2 & c_3 & \vdots & c_0 & c_1 \\ c_3 & c_2 & \vdots & c_1 & c_0 \end{bmatrix} \tag{2.38}$$

De igual manera que el caso anterior cuando se tienen 8 términos,

$$P_{8 \times 8} = \begin{bmatrix} c_0 & c_1 & c_2 & c_3 & \vdots & c_4 & c_5 & c_6 & c_7 \\ c_1 & c_0 & c_3 & c_2 & \vdots & c_5 & c_4 & c_7 & c_6 \\ c_2 & c_3 & c_0 & c_1 & \vdots & c_6 & c_7 & c_4 & c_5 \\ c_3 & c_2 & c_1 & c_0 & \vdots & c_7 & c_6 & c_5 & c_4 \\ \dots & \dots & \dots & \dots & \vdots & \dots & \dots & \dots & \dots \\ c_4 & c_5 & c_6 & c_7 & \vdots & c_0 & c_1 & c_2 & c_3 \\ c_5 & c_4 & c_7 & c_6 & \vdots & c_1 & c_0 & c_3 & c_2 \\ c_6 & c_7 & c_4 & c_5 & \vdots & c_2 & c_3 & c_0 & c_1 \\ c_7 & c_6 & c_5 & c_4 & \vdots & c_3 & c_2 & c_1 & c_0 \end{bmatrix} \tag{2.39}$$

En forma general,

$$C_{m \times m} = \begin{bmatrix} C_{\left(\frac{m}{2} \times \frac{m}{2}\right)} & C_{+\frac{m}{2}\left(\frac{m}{2} \times \frac{m}{2}\right)} \\ C_{+\frac{m}{2}\left(\frac{m}{2} \times \frac{m}{2}\right)} & C_{\left(\frac{m}{2} \times \frac{m}{2}\right)} \end{bmatrix} \quad (2.40)$$

Una propiedad importante de la matriz  $\hat{T}C$  es:

$$\hat{T}C_{m \times m} \cdot c_m = [T]_{m \times m} \cdot Tc \quad (2.41)$$

Para ejemplificar la aplicación de la matriz de coeficientes se considera el producto de dos funciones, en las cuales se emplean una función  $f(t) = t$  y  $g(t) = \sin(t)$ , para determinar el producto de ambas funciones, se puede utilizar la matriz de coeficientes, para ello se aplican las ecuaciones siguientes, usando una aproximación vía series ortogonales en las funciones mencionadas, y aplicando la ecuación (2.16) tenemos:

$$f_j(t) \cong F_j^t \phi(t) \quad (2.42)$$

$$g_{ij}(t) \cong G_{ij}^t \phi(t) \quad (2.43)$$

Por lo que el producto de ambas funciones  $F_j^t \phi(t) \cdot G_{ij}^t \phi(t)$ , puede simplificarse haciendo uso de la ecuación (2.41), aplicando previamente la transpuesta.

$$F_j^t \phi(t) \cdot G_{ij}^t \phi(t) \cong F_j^t \phi(t) \cdot \phi^T(t) \cdot G_{ij} = F_j [T]_{ij} \phi(t) \quad (2.44)$$

Una vez aplicado la ecuación (2.44), y aplicando 8 términos de FWT y truncando 4 términos de la serie de Walsh, se tiene como resultado la ecuación (2.45).

$$f(t) \cdot g(t) \cong \begin{bmatrix} \frac{1}{2} & -\frac{1}{4} & -\frac{1}{8} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \\ \phi_6 \\ \phi_7 \end{bmatrix} \quad (2.45)$$

En donde  $\mathbf{P}$  es la matriz de coeficientes de la función  $g(t) = \sin(\omega \cdot t)$ ,  
 donde el producto:

$$f(t) \cdot G(t) = \begin{bmatrix} 0.3127 & -0.2118 & -0.1012 & 0.0286 & 0 & 0 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} \phi_0 \\ \phi_1 \\ \phi_2 \\ \phi_3 \\ \phi_4 \\ \phi_5 \\ \phi_6 \\ \phi_7 \end{bmatrix} \quad (2.46)$$

En la Figura 2.7 se muestra la comparación del producto las funciones  $f(t)$  y  $g(t) = \sin(\omega \cdot t)$ , calculando en el dominio de la frecuencia y el obtenido en el dominio del tiempo.

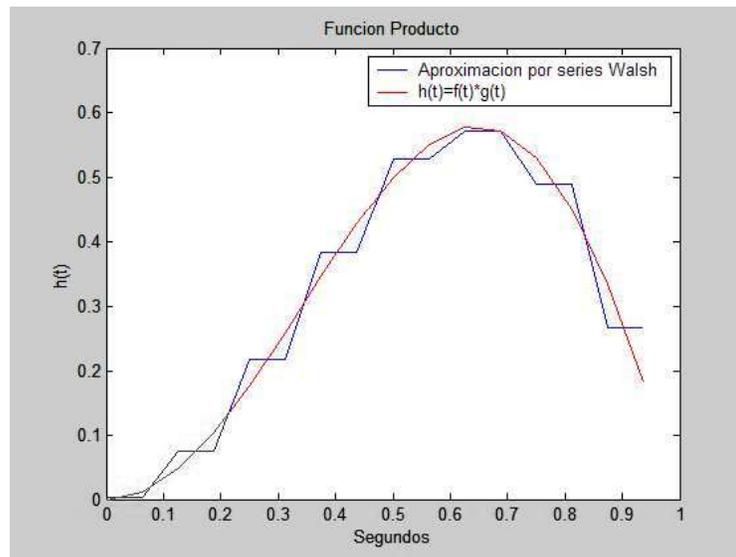


Figura 2.7 Producto de funciones.

### 2.4 Identificación Paramétrica de Sistemas Lineales Invariantes en el Tiempo.

Cualquier circuito eléctrico con elementos variantes e invariantes en tiempo puede ser descrito por medio de las ecuaciones diferenciales, las cuales se pueden escribir como un sistema de ecuaciones de estado de la forma mostrada en la Ecuación (2.47).

Aplicando la metodología de cálculo operacional antes mencionado se puede determinar cualquier vector o matriz de la misma, en este caso se determina la matriz de coeficientes [A] y [B] que a su vez, contienen los parámetros del sistema [Lázaro,Ramírez, 2006].

La técnica propuesta es empleada en SLIT, como el sistema a estudiar es SLIT es posible implementarla ya que estos sistemas invariantes con un caso particular de los SLVT.

$$\dot{x}(t) = A(t) \cdot x(t) + B(t) \cdot u(t) \tag{2.47}$$

donde:

$x(t)$  es el valor de estado y  $u(t)$  el vector de entrada de ordenes  $n \times 1$  y  $q \times 1$ , respectivamente, y las correspondientes matrices de coeficientes invariables A y B de orden  $n \times n$  y  $n \times q$ , respectivamente.

Utilizando aproximación vía series ortogonales las expresiones de los elementos  $a_{ij}(t)$  y  $b_{ij}(t)$  de las matrices  $A(t)$  y  $B(t)$  son:

$$a_{ij}(t) \cong A_{ij}^t \phi(t) \tag{2.48}$$

$$b_{ij}(t) \cong B_{ij}^t \phi(t) \tag{2.49}$$

Donde:

$$A_{ij}^t = \begin{bmatrix} A_{ij,-n} \cdots A_{ij,0} \cdots A_{ij,n} \end{bmatrix} \tag{2.50}$$

$$B_{ij}^t = \begin{bmatrix} B_{ij,-n} \cdots B_{ij,0} \cdots B_{ij,n} \end{bmatrix} \tag{2.51}$$

De igual forma para los elementos del vector  $x(t)$  y  $u(t)$  estos pueden ser aproximados como:

$$x_i \cong X_i^t \phi(t) \tag{2.52}$$

$$u_i \cong U_i^t \phi(t) \tag{2.53}$$

Por lo tanto, el producto de las series ortogonales  $A(t) \cdot x(t)$  resulta:

$$A \cdot x = \begin{bmatrix} A_{11} \phi \cdot X_1 \phi \cdots + A_{1n} \phi \cdot X_n \phi \\ A_{21} \phi \cdot X_1 \phi \cdots + A_{2n} \phi \cdot X_n \phi \\ \vdots \\ A_{n1} \phi \cdot X_1 \phi \cdots + A_{nn} \phi \cdot X_n \phi \end{bmatrix} \quad (2.54)$$

Tomando el factor  $A_{ij} \phi \cdot X_j \phi$ , de cada producto  $A_{ij} \phi$  y  $X_j \phi$  produce un escalar, por lo cual se puede reacomodar el producto como se muestra en la ecuación (2.55), al aplicar la transpuesta al término  $X_j \phi$  y la definición de la Matriz Producto, se tiene:

$$A_{ij} \phi \cdot X_j \phi = A_{ij} \phi \cdot \phi \cdot X_j = A_{ij} K_j \phi \quad (2.55)$$

donde:

$[X_j]$  es la matriz de coeficientes del vector  $x$ .

Reconstruyendo el producto  $A \cdot x$  se obtiene:

$$A \cdot x = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{bmatrix} \cdot \begin{bmatrix} K_1 \\ K_2 \\ \vdots \\ K_n \end{bmatrix} \cdot \phi \quad (2.56)$$

$$A \cdot x = A K \phi \quad (2.57)$$

donde:

$[X_j]$  es una matriz de coeficientes correspondiente a cada vector de coeficientes  $X_j$ , esta tiene dimensiones de  $m \times n$ ,  $K$  es una matriz formada por todos los coeficientes  $[X_j]$  de orden  $m \times m$ . Mientras que  $A$  es una matriz formada por cada vector de constantes  $A_{ij}$  y es de dimensiones  $n \times m$ , y  $\phi$  es el vector de funciones base de dimensiones

$1 \times m$ . Por lo tanto, se puede demostrar que el producto  $B \hat{u}$  se puede aproximar por:

$$B \hat{u} = B \int \bar{\phi} \quad (2.58)$$

En la cual las dimensiones de las matrices con  $B$ ,  $\int$  y  $\bar{\phi}$  son respectivamente  $1 \times m$ ,  $m \times m$  y  $1 \times m$ . Una vez obtenidos estos resultados es posible determinar una expresión para resolver el problema de la Identificación. Al integrar ambos lados de la ecuación (2.47) se tiene:

$$x - x_0 = A \int_0^t x dt + B \int_0^t u dt \quad (2.59)$$

Utilizando las propiedades de la matriz de integración y los resultados obtenidos en la ecuación (2.57) y (2.58), se tiene.

$$X \bar{\phi} = X_0 \bar{\phi} + A \int \bar{P} \bar{\phi} + B \int \bar{P} \bar{\phi} \quad (2.60)$$

Se considera que  $X_0$  es del mismo orden de  $x$  y contiene las condiciones iniciales de cada variable de estado, tal y como se muestra en la siguiente Ecuación.

$$X_0 = \begin{bmatrix} x_1 & 0 & \dots & 0 \\ x_2 & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ x_n & 0 & \dots & 0 \end{bmatrix} \quad (2.61)$$

Cancelando  $\bar{\phi}_0$  en ambos lados de la Ecuación (2.57), se tiene:

$$X = \theta \cdot Z \quad (2.62)$$

donde:

$$\theta = \begin{bmatrix} 1 & B & X_0 \end{bmatrix} \quad (2.63)$$

$$Z = \begin{bmatrix} \bar{X} & \bar{P} \\ \bar{U} & \bar{P} \\ e \end{bmatrix} \quad (2.64)$$

En donde  $e$  se describe como:

$$e = \begin{bmatrix} 1 & 0 & 0 & \dots \\ 1 & 0 & 0 & \dots \\ \vdots & \vdots & \ddots & \dots \\ 1 & 0 & 0 & 0 \end{bmatrix}_{m \times m} \quad (2.65)$$

Cuando se tiene un SLIT las matrices de coeficientes  $[X]$  y  $[U]$  de cada estado y entrada respectivamente, se convierten en una matriz de vectores de coeficientes  $X_j$  y el vector de coeficientes de la entrada  $U$ . Esto hace que las matrices se vuelvan más dispersas.

Es claro que para resolver este problema y obtener los parámetros es necesario aplicar el método de mínimos cuadrados para obtener la pseudoinversa de  $\theta$ , así:

$$X \cdot Z^t = \theta \left( Z \cdot Z^t \right)^{-1} \quad (2.66)$$

Si la inversa de  $\left( Z \cdot Z^t \right)^{-1}$  existe, entonces:

$$\hat{\theta} = X \cdot Z^t \left( Z \cdot Z^t \right)^{-1} \quad (2.67)$$

### 2.4.1 Redundancia Analítica.

Dentro de los sistemas dinámicos es frecuente encontrar una redundancia física, es decir, la implementación de varios dispositivos de las mismas características dentro de un mismo sistema de medición y/o control. Por ejemplo, cuando se utilizan dos o más sensores para prever cualquier eventualidad ante una falla del mismo en ocasiones una inestabilidad del sistema, esta situación ocurre en aquellos sistemas en donde se requieren sistemas de respaldo para proporcionar cualquier confiabilidad del sistema.

Otra alternativa a la redundancia física que suele ser más económica es la redundancia analítica. Este tipo de redundancia supone la posibilidad de realizar varios

experimentos sobre el sistema a identificar, los experimentos realizados incorporan diferentes modos de operación cuando estos se excitan con diferentes tipos de entradas en diferentes intervalos de tiempo, otra variante constituye la aplicación de la misma señal a diferentes frecuencias, corrimientos de fase o cambios de magnitud (Alvarado, 2001).

Posteriormente se utiliza toda esta información en un mismo proceso de estimación, tal y como se observa en la Figura 2.8, la desventaja de esta redundancia analítica es que hace que el proceso de identificación se realice fuera de línea.

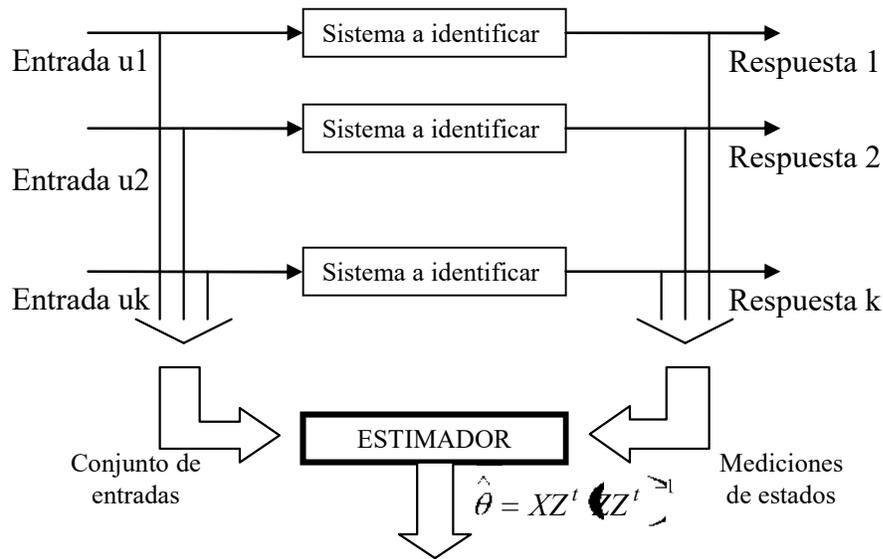


Figura 2.8. Esquema de redundancia analítica.

## Capítulo 3

# Diseño de la Interfase Gráfica

### 3.1 Introducción.

Para realizar la identificación de parámetros, se desarrolló una interfase gráfica amigable para la interacción del usuario con la información que se desea obtener del sistema, esta interfase proporciona resultados fáciles de interpretar.

En el análisis del comportamiento del Motor de CD de imanes permanentes e identificación de parámetros, se utilizó el software LabVIEW® de *National Instruments*, debido a su versatilidad programación para el desarrollo de instrumentos virtuales. Para simplificar la programación se emplean iconos y funciones ya prediseñados para la conexión con MatLab®.

### 3.2 Programación del Instrumento Virtual.

Los Programas en LabVIEW son llamados instrumentos virtuales (VIs), cada uno contiene [Curso de Instrumentación Virtual, 2003]:

- Panel Frontal.- Permite que el usuario interaccionar con el VI.
- Diagrama de bloque.- Es el código fuente que controla el programa.
- Icono/Conector.- Son los medios para conectar un VI con otros VIs.

El panel frontal de un VI es manipulado por el usuario cuando el programa esta corriendo. Los usuarios pueden controlar el programa, cambiar entradas y ver datos actualizados en tiempo real. Se debe hacer énfasis en que los controles son usados como entradas, ajustando controles de deslizamiento para colocar un valor de alarma, encendido o apagado tipo interruptor, o parado del programa.

Los indicadores son usados como salidas. Los termómetros, luces, y otros indicadores indican valores del programa. Esto puede incluir datos, estados de programa u otra información [Curso de Instrumentación Virtual, 2003].

Cada control o indicador de panel frontal tiene una terminal correspondiente en el diagrama de bloques. Cuando un VI se ejecuta, los valores de los controles fluyen a través del diagrama de bloques, en donde éstos son usados en las funciones del diagrama, y los resultados son pasados a otras funciones o indicadores.

En la Figura 3.1 se puede observar el panel frontal de un VI, en éste se colocan los dispositivos de entrada o salida para manipular y visualizar los resultados del programa, en esta área precisamente es donde el usuario interactuará con el programa y la visualización de los resultados.

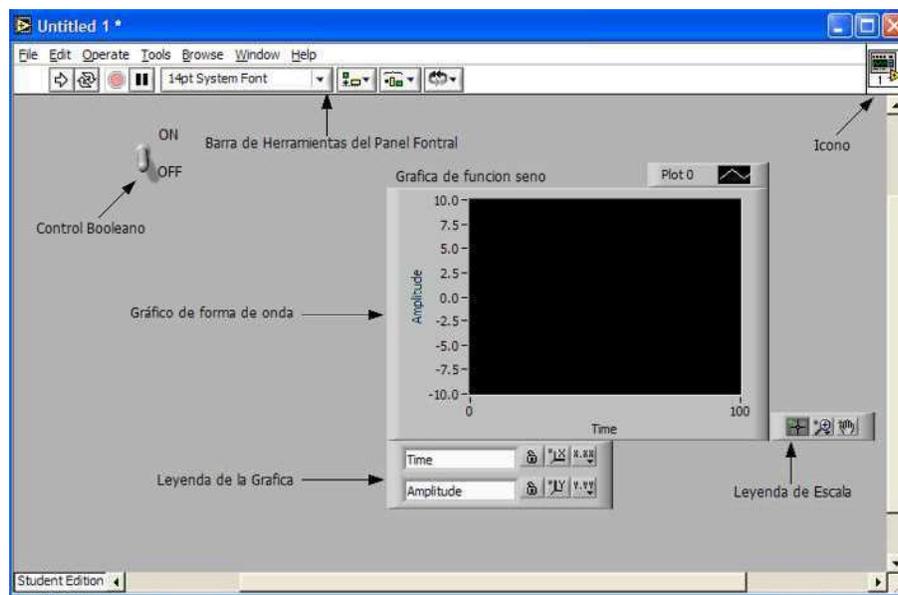


Figura 3.1 Panel Frontal.

En la Figura 3.1, el interruptor de poder (encendido o apagado) es un control booleano. Un valor booleano contiene un valor verdadero o falso. El valor es falso hasta que el interruptor se presione. Cuando el interruptor se presiona, el valor se convierte en verdadero. El indicador del historial de la función seno, es una gráfica de forma de onda, ésta muestra los valores del comportamiento en el tiempo (seg).

En la Figura 3.2 se encuentran los tipos de controles principales para trabajar en esta área, entre los cuales están, interruptores virtuales, así como indicadores, etc:

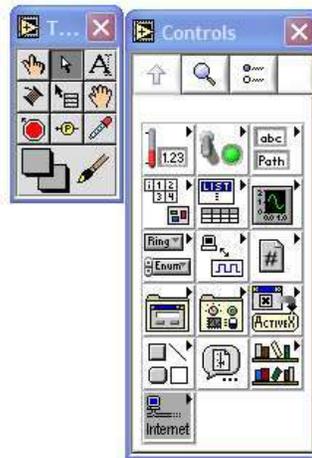


Figura 3.2 Paletas de control (panel frontal).

El diagrama de bloque contiene el código fuente gráfico. Los objetos del panel frontal aparecen como terminales en el diagrama de bloques. Adicionalmente, el diagrama de bloques contiene funciones y estructuras incorporadas en las bibliotecas de LabVIEW VI. Los cables conectan cada uno de los nodos en el diagrama de bloques, incluyendo controles e indicadores de terminal, funciones y estructuras.

En la Figura 3.3 se observa el diagrama de bloques que conforma a un programa de suma y resta de números de tipo double.

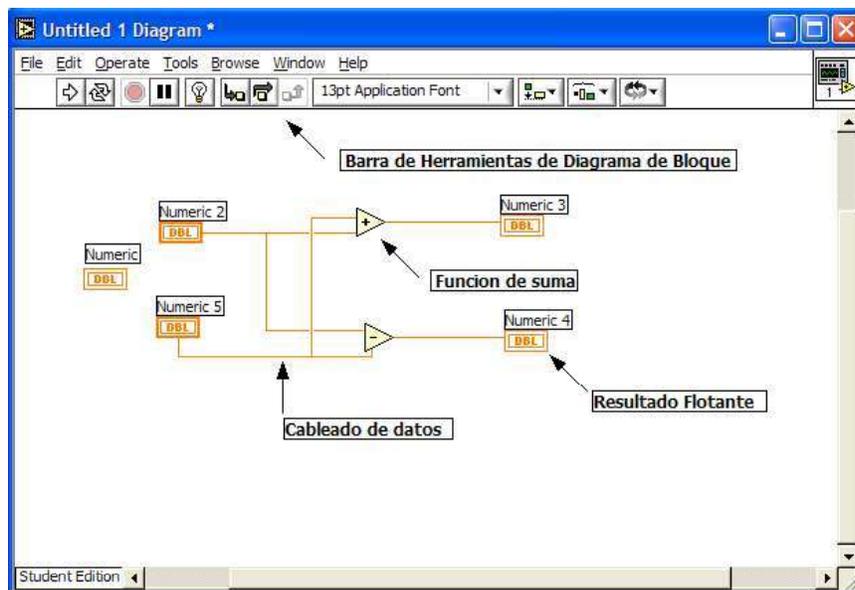


Figura 3.3 Diagrama de Bloques.

Por otro lado el panel frontal el diagrama de bloques cuenta con una paleta de funciones la cual se pueden ver en la Figura 3.4.



Figura 3.4 Paleta de funciones (ventana de diagrama de bloque).

En este programa existe la posibilidad de trabajar con subprogramas denominados subVIs, cada uno de ellos corresponde a una subrutina en lenguaje de programación, algunos de los cuales están diseñados para realizar cálculos específicos en nuestro caso se a utilizó el subVI de Gráficas.

La estructura empleada en la programación de cualquier diseño en LabVIEW® se puede identificar por algunas de las siguientes características.



Recuadro naranja para números o variable de tipo flotante o double.



Recuadros azules para enteros.



Líneas y recuadros rosas para cadenas y arreglos de cadenas.



Recuadro verde entrada o salida de un dato digital.



Recuadros morados para la configuración de la adquisición de datos.

### 3.3 Descripción del Diseño del Panel Frontal Principal.

La estructura del programa en general esta diseñada como se muestra en la Figura 3.5, la cual consta de dos etapas de análisis, la primera destinada al proceso de datos del circuito, la visualización de los parámetros y la segunda, dedicada a la visualización gráfica de las variables de estado.

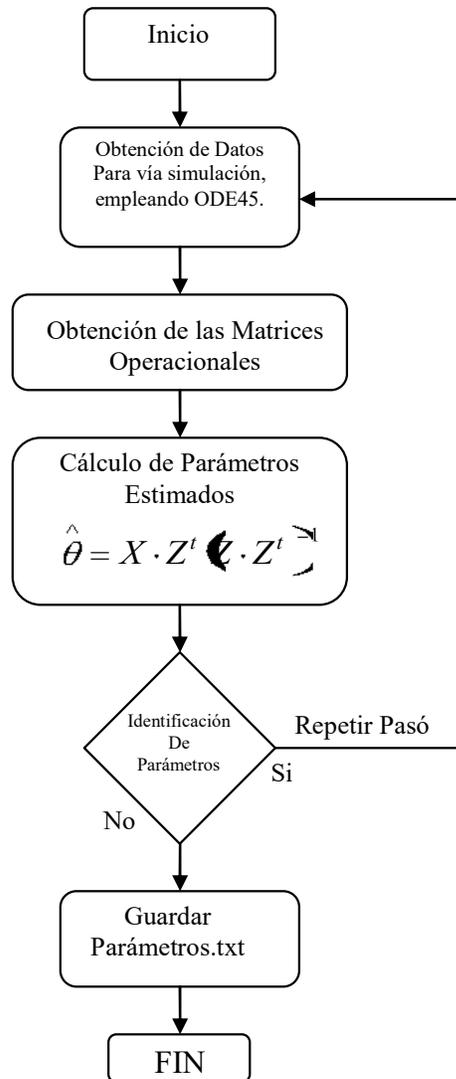


Figura 3.5 Diagrama de flujo del panel frontal principal.

Como primer paso se realiza la obtención de los parámetros reales del Motor de CD, estos son necesarios debido a que el algoritmo se esta probando en simulación, por lo

cual con estos datos se generan los datos de entrada y salida del sistema, conocidos y manipulados por el usuario, los datos de entrada se pueden encontrar en la tabla que muestra el fabricante [Apéndice B]. A continuación se usan los datos para obtener las matrices operacionales a emplear, posteriormente se aplica la ecuación para determinar los parámetros estimados del sistema, si desea emplear otras entradas se repite el procedimiento, finalmente se guardan los parámetros estimados generados.

En lo que respecta al panel frontal, la Figura 3.6 muestra la interfaz diseñada para la obtención de datos, la simulación del sistema y la visualización de resultados. Los cuales son fáciles de interpretar. Además, presenta una interacción de manera amigable con el usuario.

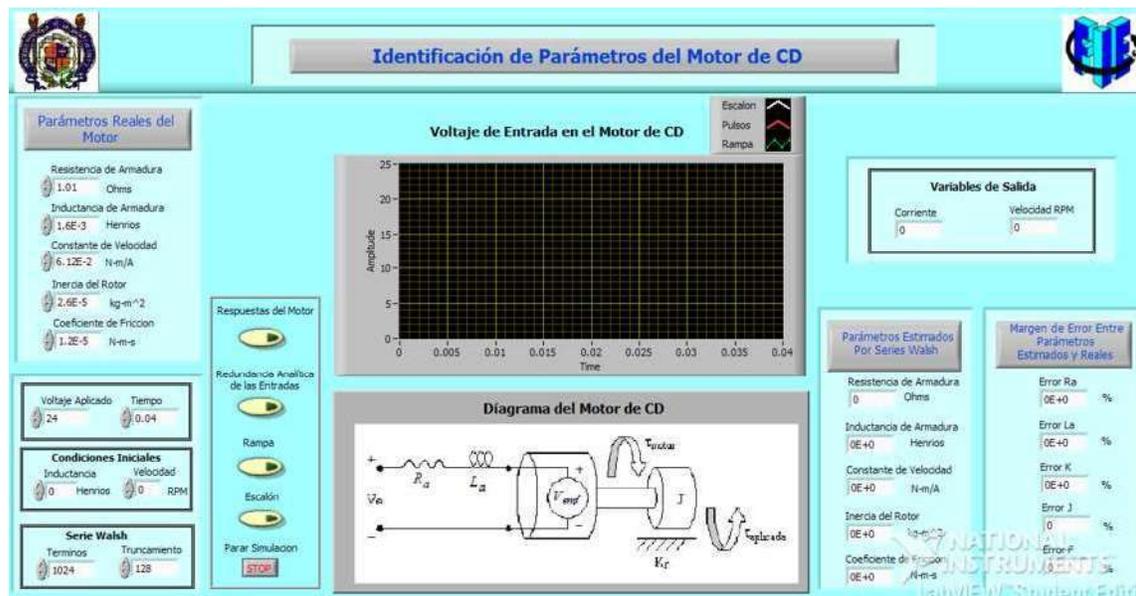


Figura 3.6 Panel principal de la interfaz diseñada.

La interfase mostrada en la Figura 3.6 consta de un panel en donde se proporciona datos y se presenta los resultados, así como la visualización de gráficas; en los siguientes pasos se describe el funcionamiento de la misma:

- Paso 1.- Este paso consiste en proporcionar los parámetros reales del motor, para este caso se guardaron los parámetros del motor de CD por default, estos

permiten generar el conjunto de datos requeridos para la estimación de parámetros.

- Paso 2.- En este paso se determinan las condiciones iniciales del sistema a simular, así como el número de términos para la transformada de Walsh y el de truncamiento de la serie Walsh. En la misma plataforma se proporciona un intervalo de tiempo en segundos para la respuesta del sistema.
- Paso 3.- Una vez realizado los pasos anteriores, se puede proceder a identificar el sistema a través del algoritmo presentado, por lo cual en la interfaz se tienen cinco botones, con ellos se analiza las diferentes entradas señaladas que corresponde al botón mostrando, como resultado nos muestra los parámetros estimados por series Walsh, la gráfica de entrada al sistema, así como el error que hay entre los parámetros reales y los estimados.

### 3.4 Nodo de MATLAB Script.

El uso de este nodo permite ejecutar una función de LabVIEW®, dicha función esta diseñada para interactuar con la aplicación de MatLab®, con la ventaja de utilizar funciones y programas desarrollados en esta plataforma, se encuentra en la paleta de funciones, y tiene una estructura como la mostrada en la Figura 3.4, utilizando la función Mathematics, se elige el icono fórmula, y posteriormente se selecciona el icono de MATLAB Script, ya insertado el icono se importa el programa desarrollado en MatLab, para la ejecución del algoritmo.

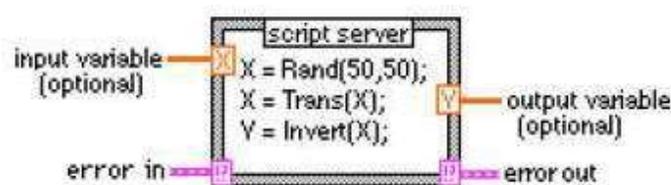


Figura 3.7 Nodo MATLAB Script.

En el apéndice A se muestran los códigos utilizados en este nodo para la estimación de parámetros, al igual que los vectores para la visualización de las gráficas. Una vez obteniendo los resultados en MatLab, se hace uso de este nodo. La configuración del nodo es relativamente simple ya que sólo se deben configurar las

entradas y las salidas con el tipo de dato con el que MatLab trabaja, como se muestra en la Figura 3.7.

En la obtención de los parámetros es necesario configurar el tipo de dato de entrada al nodo de MATLAB Script, ya sea una variable de tipo double, cadena o un arreglo y conectarla al mismo nodo, como se muestra en Figura 3.8. Cabe señalar que este tipo de función de LabVIEW permite correr el código en MatLab, a través de la plataforma Command de MatLab.

```

global n;
global T;
Ts=T/m;
t=0:Ts:T-Ts;
%Parametros del motor de CD
global ra;
global Ia;
global k;
global j;
global fr;
global Va;
options = odeset('RelTol',1e-8,'AbsTol',[1e-6 1e-6]);
[t1,xx1]=ode45('fmotor1',t,xo,options);
z=zeros(size(t1));
P=pwalsch(m);%Creamos la matriz producto
%Diferentes entradas al sistema
u=(ones(size(t1)))^Va;% entrada escalon de 24
V.
%Entradas para visualizar las entradas en
errorK errorF errorJ errorLa errorRa
    
```

Figura 3.8 Conexión del nodo MATLAB Script con la captura de datos.

En la Figura 3.8 se muestra parte del código programado en MatLab, lo cual corresponde a lo remarcado en azul muestra la utilización del ODE45, el cual regresa un vector conteniendo el tiempo y la respuesta de las variables de estado del sistema, para ello hace uso de las condiciones iniciales y del modelo del sistema.

Por otro lado, en la Figura 3.8 se observan los recuadros naranjas que indican las entradas y salidas tipo double, en unos casos se tienen salidas de tipo vector, debido a que estas se utilizan para graficar, cabe mencionar que los datos arrojados por MatLab en algunos vectores no pueden ser leídos en LabVIEW, por lo cual se tiene que hacer una transformación de vectores, en el código del programa principal.

Aplicando los datos de entrada para modelar el sistema se manipulan desde la interfase del panel principal, una parte es mostrada en la Figura 3.9, en ella el usuario puede modificar los parámetros, dependiendo del Motor con el que se desea efectuar las pruebas.



Figura 3.9 Interfaz para la captura parámetros reales del Motor de CD.

El diagrama de flujo, de la Figura 3.10 muestra la composición de teta ( $\hat{\theta}$ ), para la obtención de los parámetros estimados.

```

MATLAB Script
ra1 Va=T*u*P;
k11 elm=zeros(size(IaP));
x222 elm(:,1)=1;
n1 ZT=[IaP;wP;Va;elm];
m1 format long;%Escala para 15 digitos
j1 ct=[X1;X2];
fr1 %Se aplica la pseudoinversa, aproximación de
a1 TOL=1e-8;
k1 Z1inv=ZT'*inv(ZT*ZT');
%este TETA ES MATRIZ
r11 teta=ct*Z1inv;
Va1
errorK errorF errorJ errorLa errorRa
    
```

Figura 3.10 Teta para la obtención de los parámetros.

En la Figura 3.10 el nodo muestra parte del programa, el cual consiste en la obtención de parámetros, las letras remarcadas de azul indican las variables de estado del sistema y la entrada. Las letras de color rojo corresponden a las salidas y por último se tiene en verde los resultados correspondientes a los parámetros estimados usando series Walsh.

En la Figura 3.11 se muestra una interfaz similar a la mostrada en la Figura 3.9, en ésta se pueden apreciar los parámetros estimados por series Walsh y el error que existe entre ellos, una vez ejecutado el algoritmo.

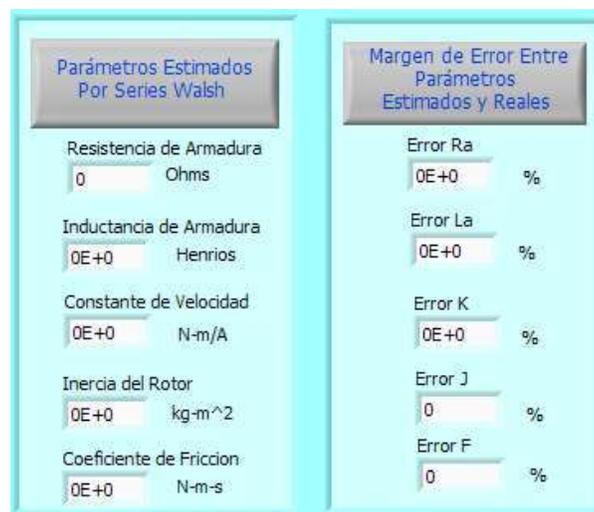


Figura 3.11 Interfaz para la visualización de los resultados obtenidos.

### 3.4.1 Código del Programa Principal.

Al enlazar los códigos mostrados anteriormente, se obtienen cuatro nodos de Matlab Script, los cuales se encargan de ejecutar la condición mediante un botón tipo booleano, estos nodos se encuentran dentro de un While Loop, este lo podemos encontrar en la Figura 3.4. En el primer nodo se tiene que analizar la entrada escalón, en este nodo se tienen entradas y salidas de tipo double, vectores y arreglos de 2D para la visualización de la entrada simulada.

Una vez encontrado teta ( $\hat{\theta}$ ) el programa arroja un resultado de tipo matriz, el cual se descompone como se muestra en las letras remarcadas de azul, ubicadas en el nodo MatLab Script de la Figura 3.12.

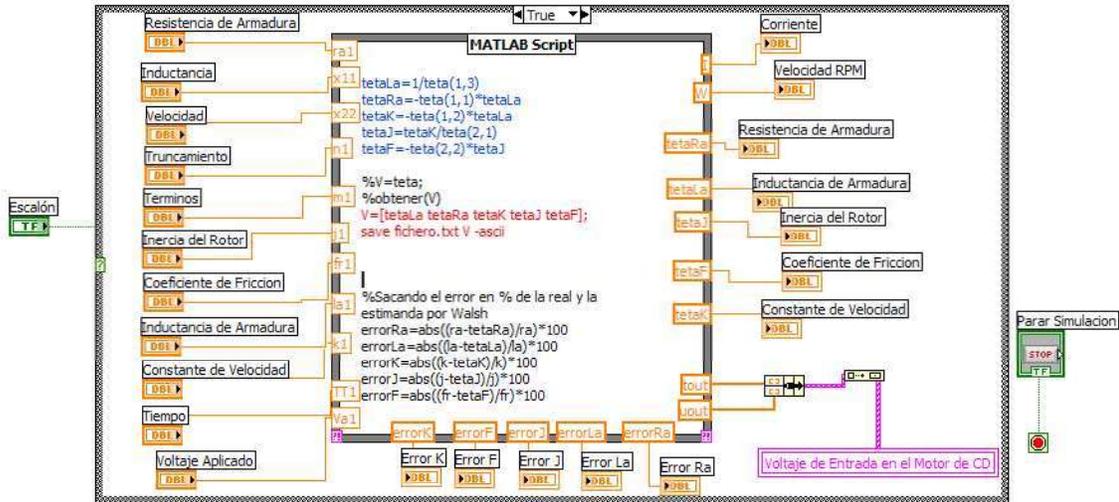


Figura 3.12 Diagrama de Bloques del Panel Frontal para Entrada Escalón.

Para ello se guardan los parámetros estimados en un vector de  $1 \times n$  en un archivo de tipo “txt”, como esta en las letras rojas, estos valores van cambiando cada vez que se repite el paso para el cálculo de los parámetros [MATLAB 6.1, 2001].

En el caso de una entrada rampa se presenta de manera similar a la de escalón, la única diferencia esta en la programación del código en Matlab, en la parte de la definición de la señal aplicada en la Figura 3.13.

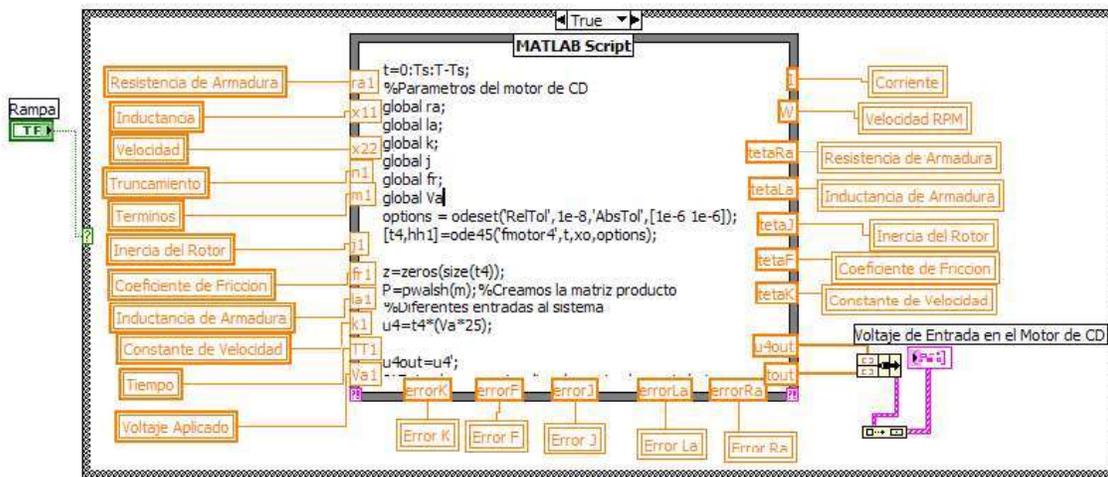


Figura 3.13 Diagrama de Bloques del Panel Frontal para Entrada Rampa.

En la Figura 3.13, se aprecian las variables locales creadas, tomando la referencia de los indicadores y controles de la Figura 3.12.

Para la aplicación de la redundancia analítica de las entradas, se produce de manera similar al anterior, creando las variables locales correspondientes, en donde se activan para leer los datos en las variables de control de color anaranjado, para este caso tenemos la siguiente Figura 3.14.

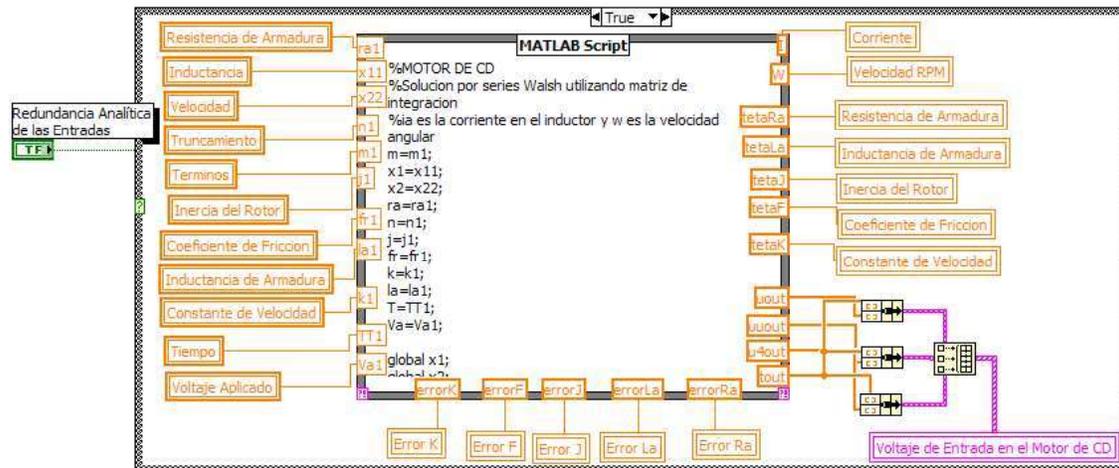


Figura 3.14 Diagrama de Bloques del Panel Frontal para Redundancia Analítica.

### 3.5 Diagrama de Flujo del SubVI “Gráficas”.

La estructura del programa del SubVI de gráficas, es similar al de la Figura 3.5, en este diagrama en particular, realiza el llamado de los parámetros estimados, los cuales fueron calculados en el panel frontal de la Figura 3.6. Por su parte la Figura 3.15 se muestra el diagrama de flujo para el SubVI.

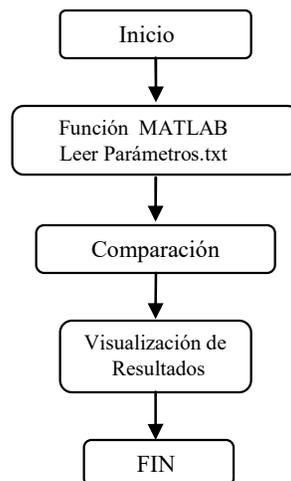


Figura 3.15 Diagrama de Flujo de SubVI Gráficas.

Cuando tenemos los parámetros estimados se guardan en un archivo con extensión “txt” en código ASCII, una función presentada en el nodo del subVI, lee los parámetros estimados de la ejecución aplicada en la Figura 3.6, los cuales grafica y los compara con el comportamiento de los reales.

### 3.5.1 Visualización de las Variables de Estado del Motor de CD.

Para la visualización de las graficas se realiza un subVI llamado “Graficas” en la Figura 3.16 se muestra el subVI que visualiza las variables de estado (Corriente y Velocidad) del Motor de CD, la cual permite hacer una comparación entre la grafica de los parámetros reales y la de los estimados por series Walsh.



Figura 3.16 SubVI para la visualización de las gráficas del comportamiento del Motor de CD.

El nodo de MATLAB Script se encarga del proceso de identificación paramétrica, el cual proporciona los parámetros estimados, el error junto con las gráficas del comportamiento del Motor de CD. En este caso se usa una función creada en MatLab para leer los parámetros que fueron almacenados, como lo indica la Figura 3.12, el nodo implementado se aprecia en la Figura 3.17.

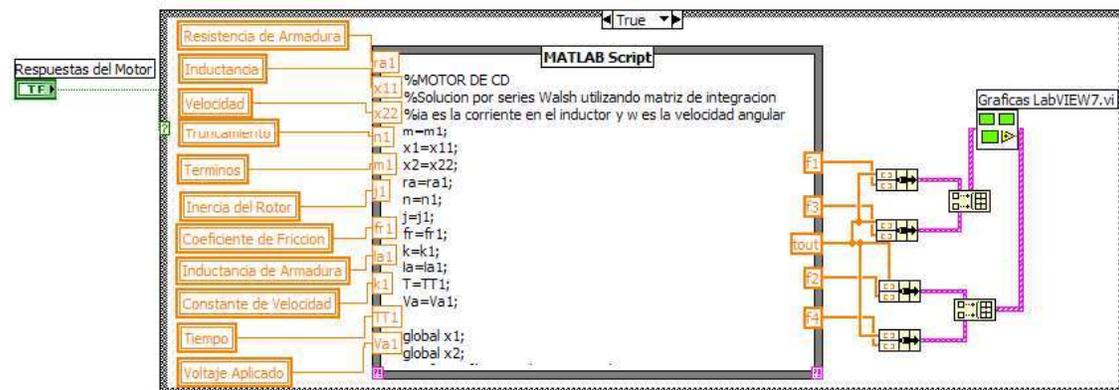


Figura 3.17 Diagrama de Bloques del Panel Frontal para SubVI.

Para poder observar las variables de estado, se tienen que leer los indicadores de las respuestas del nodo al SubVI.

En la Figura 3.18 se presenta una vista del panel frontal para la visualización de las variables de estado del Motor de CD, con el propósito de mostrar los resultados del análisis.

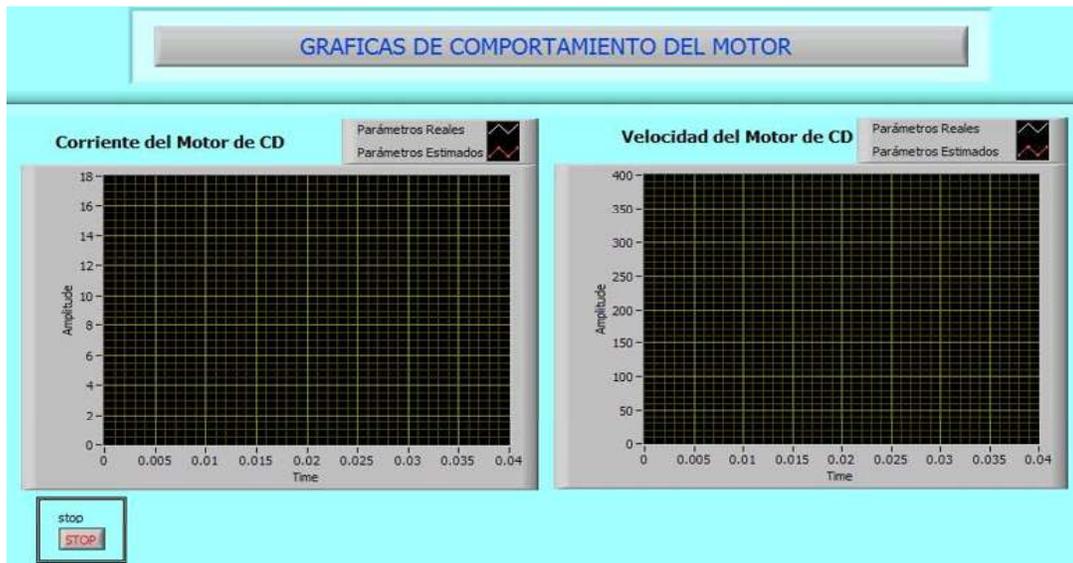


Figura 3.18 Panel del SubVI para la visualización de las gráficas.

## Capítulo 4

# Pruebas y Resultados

### 4.1 Introducción.

Los motores de CD son un elemento importante para aplicaciones en las que se ajusta la velocidad, así como en aquellas que se requieren un par grande. En la actualidad, se siguen utilizando los motores de CD, cuya potencia es de una fracción de caballo, en la industria del transporte como: automóviles, trenes y aviones.

El motor de CD tiene un estator y un rotor (armadura). El estator contiene un devanado por cada polo, los cuales están diseñados para llevar intensidades de corriente directa que establecen un campo magnético.

La armadura y su devanado están ubicados en la trayectoria de este campo magnético y cuando el devanado lleva intensidades de corriente, se desarrolla un par-motor que hace girar al motor. Hay un conmutador conectado al devanado de la armadura, si no se utiliza un conmutador, el motor sólo podría dar una fracción de vuelta y luego se detendría. El par que se produce en un motor de CD, es directamente proporcional a la intensidad de corriente de armadura y al campo del estator; la configuración del motor contiene cinco elementos básicos los cuales son la resistencia de armadura ( $R$ ), la inductancia ( $L$ ), coeficiente de fricción ( $f_r$ ), la inercia del rotor ( $J$ ) y el coeficiente ( $K$ ), para el caso de estudio.

### 4.2 Determinación de las ecuaciones de la dinámica del Motor.

En la Figura 4.1, se muestra el circuito equivalente del Motor de CD, cuando se efectúa un control por armadura. El motor CD se visualiza como un convertidor de energía electromecánica ideal.

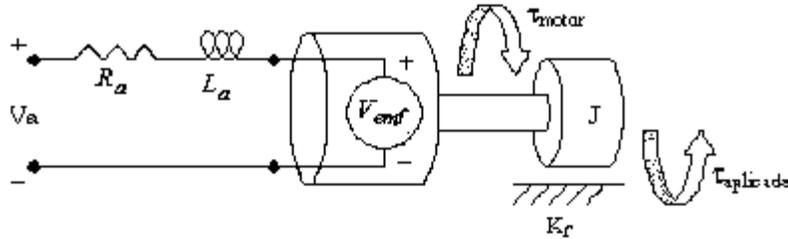


Figura 4.1 Circuito equivalente del motor de CD.

Este sistema se puede modelar a partir de dos variables de estado, la primera se obtiene al aplicar la ley de voltajes de Kirchoff al circuito de armadura, lo cual produce [Herrera, at. al, 2006].

$$V_a = i_a R_a + L_a \frac{di_a}{dt} + V_{emf} \quad (4.1)$$

Donde:

$i_a$  .- Es la corriente de armadura.

$V_{emf}$  .- Es el voltaje generado en el interior de la máquina.

$\omega$  .- Es la velocidad rotacional de la armadura.

$K_b$  .- Es la constante contraelectromotriz.

El voltaje de la fuerza contra electromotriz se considera como:

$$V_{emf} = K_b \omega \quad (4.2)$$

Sustituyendo la Ecuación (4.2) en (4.1) y despejando la derivada  $\frac{di_a}{dt}$ , se tiene la ecuación eléctrica del motor.

$$\frac{di_a}{dt} = \frac{V_a}{L_a} - \frac{i_a R_a}{L_a} - \frac{K_b \omega}{L_a} \quad (4.3)$$

Realizando un balance de energía del sistema mecánico, se obtiene la segunda variable de estado  $\frac{d\omega}{dt}$  que representa la velocidad angular, considerando lo siguiente [Herrera, et. al, 2006]:

$$T_g = T_p + T_f + T_L \quad (4.4)$$

Donde:

$T_g$ .- Es el torque interno generado por la máquina o par electromecánico.

$T_p$ .- Es el par de la aceleración rotacional del rotor.

$T_f$ .- Es el par o torque de pérdidas debidas ala fricción del rotor.

$T_L$ .- Es el par de carga, que en este caso es cero.

El par del motor (4.5) está relacionado con la corriente a través de la armadura por la constante  $K_T$ , la cual representa la constante de torque o par:

$$T_g = K_T i_a \quad (4.5)$$

Desarrollando la ecuación (4.4), toma la forma siguiente:

$$K_g i_a = J \frac{d\omega}{dt} + K_f \omega \quad (4.6)$$

Despejando la derivada  $\frac{d\omega}{dt}$  de la ecuación (4.6), se tiene:

$$\frac{d\omega}{dt} = \frac{K_T i_a}{J} - \frac{K_f \omega}{J} \quad (4.7)$$

Representando en forma matricial la ecuación (4.3) y (4.6) se obtiene una ecuación de la siguiente forma:

$$\begin{bmatrix} \dot{i}_a \\ \dot{\omega} \end{bmatrix} = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{K}{L_a} \\ \frac{K}{J} & -\frac{K_f}{J} \end{bmatrix} \begin{bmatrix} i_a \\ \omega \end{bmatrix} + \begin{bmatrix} \frac{1}{L_a} \\ 0 \end{bmatrix} V_a \quad (4.8)$$

Considerando que  $K = K_T = K_b$ .

### 4.3 Pruebas para la identificación de parámetros del Motor de CD.

Para asegurar que los parámetros estimados sean confiables es necesario, realizar varias pruebas que permitan validar los resultados obtenidos, en este caso de estudio.

El motor de CD utilizado para generar los datos, corresponde al mostrado en la Figura 4.2, el cual es un Motor de CD con imanes permanentes, fabricado por Pittman Power Your Ideas, de las series GM14900 de tipo LO-COG® DC Gearmotors. Los parámetros de acuerdo con los datos del fabricante se presentan en la Tabla 4.1. En el Apéndice B se presenta la hoja de datos por el fabricante.



Figura 4.2 Motor implementado en la simulación.

Tabla 4.1 Datos tomados de la hoja del fabricante.

Voltaje de entrada:	24 Volts.
Resistencia del devanado de armadura:	1.01 Ohms.
Inductancia del devanado:	0.0016 Henrios.
Constante contraelectromotriz:	0.0612 (N-m/A).
Inercia del rotor del motor:	0.000026 (kg-m <sup>2</sup> ).
Constante de fricción viscosa:	0.000012 (N-m-s).

#### 4.3.1 Prueba 1 Entrada escalón de 24 Volts.

En ésta prueba se presenta un análisis aplicando una entrada escalón, en la Figura 4.3 se muestra los datos del motor a modelar, y el resultado de los parámetros estimados, así como las salidas del sistema.

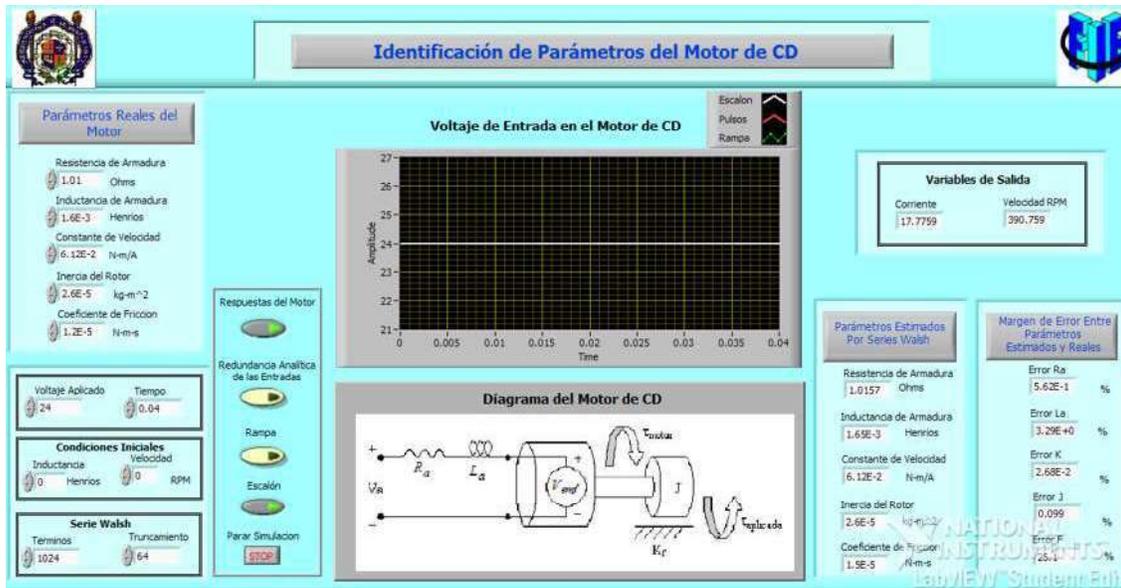


Figura 4.3 Panel frontal de resultados y entrada de datos.

En la Figura 4.3 se muestra el panel frontal, en éste se introducen los parámetros de la Tabla 4.1, así mismo, se introducen las condiciones iniciales. Tomando 1024 puntos y truncando a 64 términos de la Serie Walsh, con un tiempo de 0.04 de respuesta, se obtienen los resultados mostrados en la Figura 4.4.

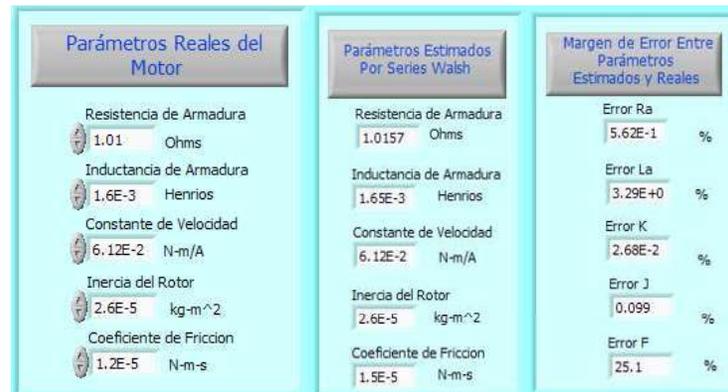


Figura 4.4 Parámetros reales vs estimados y error existente.

Como se puede apreciar en la Figura 4.4, en la parte izquierda se insertan los datos reales del motor a simular que da el fabricante para generar una señal de respuesta usando ODE45, en los datos intermedios se aprecian los parámetros estimados por Series Walsh, los que en esta simulación algunos se acercan a los reales. El margen que

existe entre los reales y estimados se muestran en la parte derecha. La Figura 4.5 corresponde al SubVI que gráfica los parámetros haciendo una comparación.

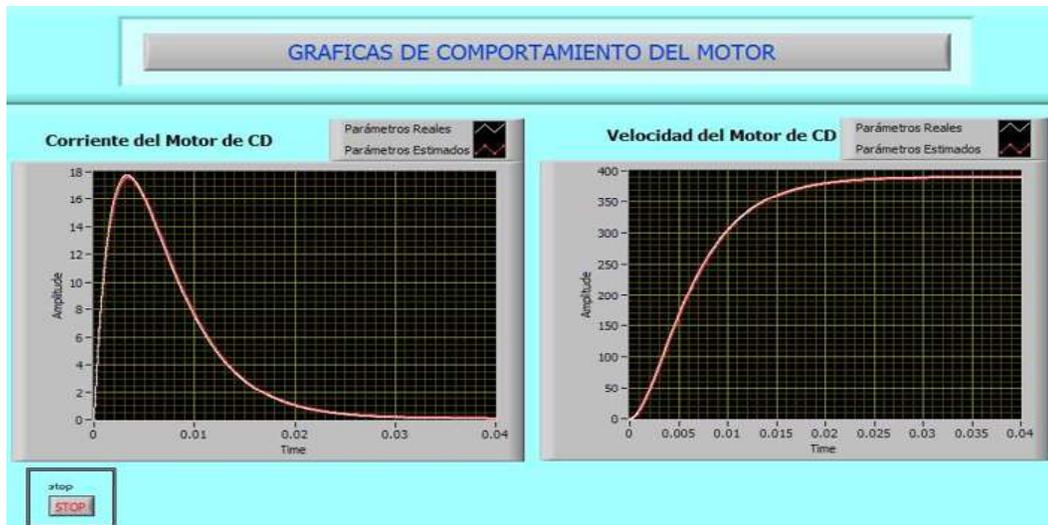


Figura 4.5 Panel frontal mostrando las variables de estado.

En la Figura 4.5 muestra una simulación del error que existe entre los parámetros reales y los estimados por las series Walsh, como se mostró en la Figura 4.4, en la cual existen unos parámetros que sobre pasan el margen de 5% de tolerancia, para tener un resultado mas preciso, se repite la misma prueba sólo cambiando el No. de términos a  $m = 1024$  y de truncamiento a  $n = 128$ , con lo cual se busca reducir el margen del error.

La Figura 4.6 muestra el resultado de los parámetros considerando este número de términos y truncamiento, donde se muestra un resultado favorable. En lo práctico, dado que no se conocen los parámetros reales, lo que se usa es una comparación de la respuesta producida en lazo abierto por el sistema.

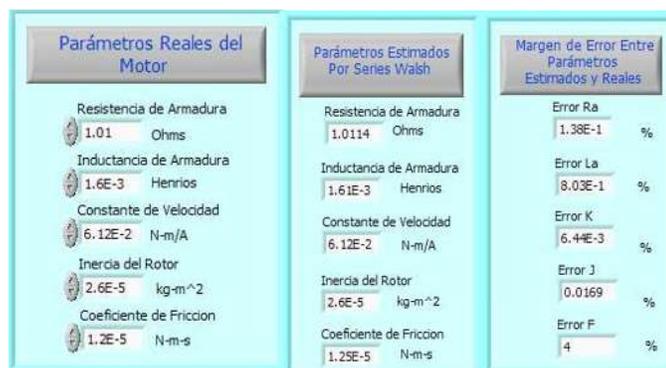


Figura 4.6 Comparación de Parámetros reales y Estimados con Error 5%.

Comparando con la obtenida por simulación del sistema al tomar los parámetros, tal como se muestra en la Figura 4.7.

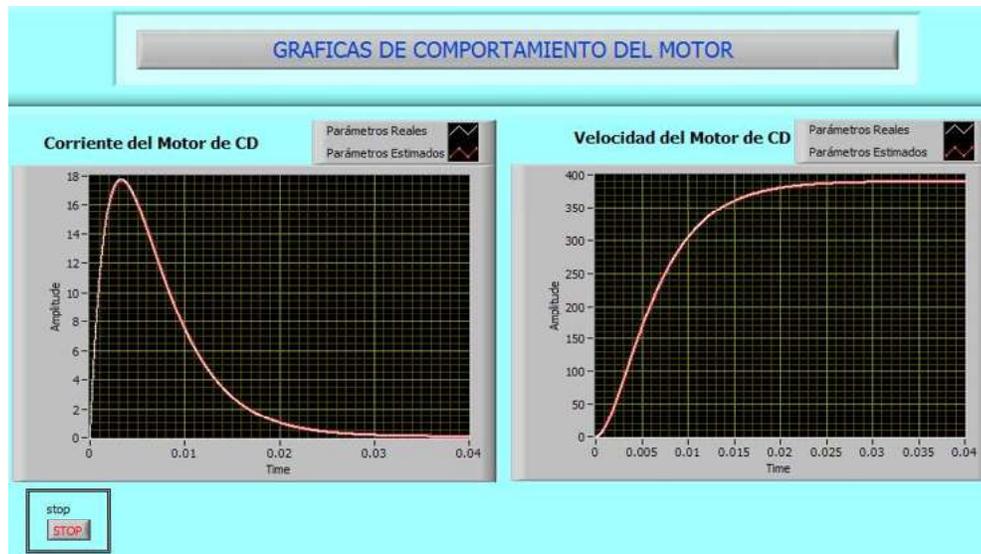


Figura 4.7 Comportamiento de Variables de Estado de  $m = 1024$  y  $n = 128$ .

Tal y como se había dicho, cuanto se aumenta el número de términos de la serie de Walsh el error de los parámetros disminuye considerablemente, como resultado al realizar una comparación entre los reales y estimados hay poco error, por lo que las variables de estado de la Figura 4.7 son muy parecidas.

### 4.3.2 Prueba 2 Aplicación de redundancia analítica.

La Figura 4.8 corresponde a la vista del gráfico del panel frontal, en la cual se observan las entradas que se le están aplicando al motor; en este caso se aplica la redundancia analítica con el fin de obtener el comportamiento con la suma de las diferentes entradas, ya que con algunos tipos de entrada no se excitan algunos parámetros del sistema.

Se requiere aplicar otra entrada diferente, para poder excitar los parámetros que son susceptibles, en este caso aplicamos una entrada escalón, un tren de pulsos que tiene una frecuencia de 95.5 Hz y una rampa.

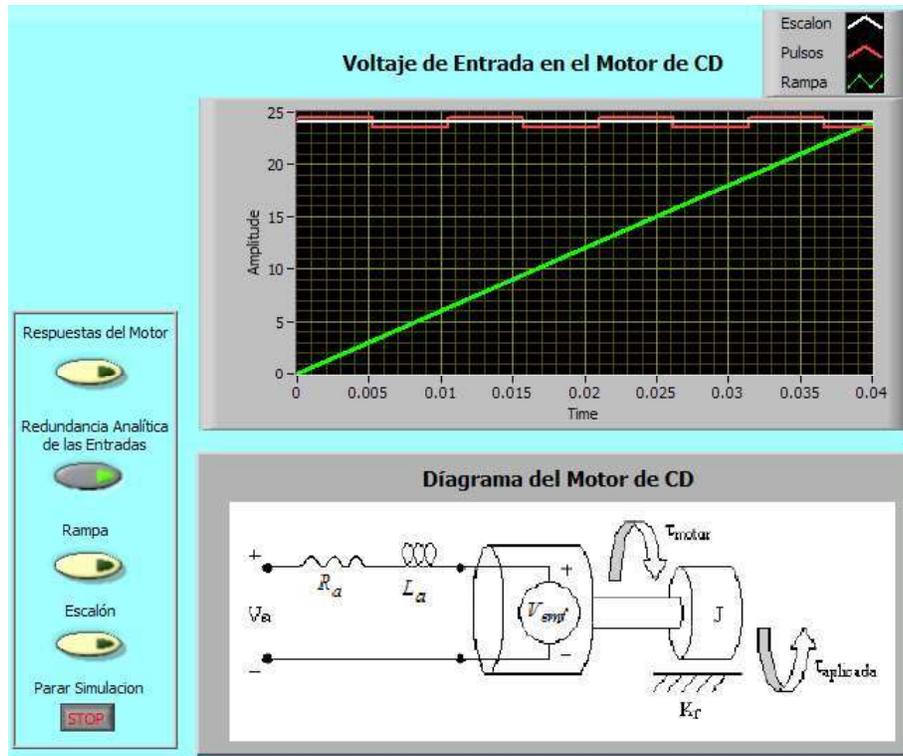


Figura 4.8 Entrada escalón sumada a un tren de pulso y con entrada rampa.

En esta prueba considerando la redundancia analítica, se toma un valor de  $m = 1024$  puntos y se trunca a  $n = 64$  términos de la serie Walsh, el resultado se ve en la Figura 4.9.

Parámetros Reales del Motor	Parámetros Estimados Por Series Walsh	Margen de Error Entre Parámetros Estimados y Reales
Resistencia de Armadura 1.01 Ohms	Resistencia de Armadura 1.0132 Ohms	Error Ra 3.12E-1 %
Inductancia de Armadura 1.6E-3 Henrios	Inductancia de Armadura 1.63E-3 Henrios	Error La 2.03E+0 %
Constante de Velocidad 6.12E-2 N-m/A	Constante de Velocidad 6.12E-2 N-m/A	Error K 2.59E-2 %
Inercia del Rotor 2.6E-5 kg-m <sup>2</sup>	Inercia del Rotor 2.6E-5 kg-m <sup>2</sup>	Error J 0.0533 %
Coefficiente de Friccion 1.2E-5 N-m-s	Coefficiente de Friccion 1.3E-5 N-m-s	Error F 8.02 %

Figura 4.9 Estimación aplicando la redundancia analítica con  $m = 1024$  y  $n = 64$ .

El resultado de la Figura 4.9 se puede considerar aceptable al tener un error menor del 10%. Para reducir el error se incremento el número de términos a 128 dejando  $m = 1024$ , con esto se obtuvieron los resultados de la Figura 4.10.

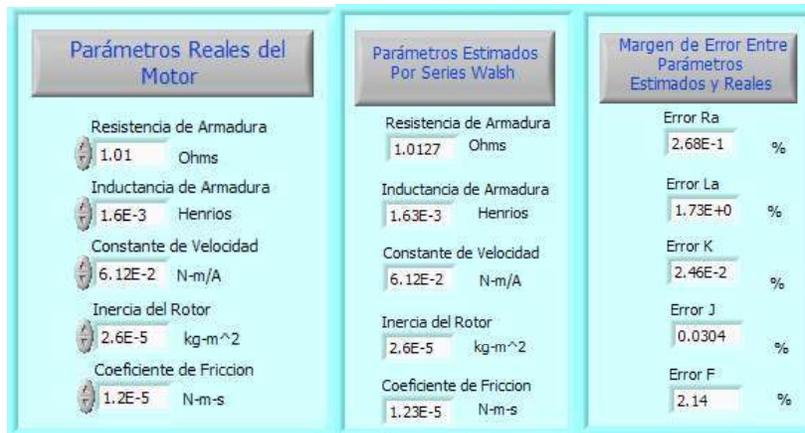


Figura 4.10 Estimación aplicando la redundancia analítica con  $m = 1024$  y  $n = 128$ , con error de 5%.

Al aplicar la redundancia analítica usando como entradas (escalón, tren de pulsos y rampa) en el Motor de CD, la simulación de la Figura 4.10, muestra que los errores en los parámetros disminuyen considerablemente, respecto a la simulación de la Figura 4.9, por lo cual esta prueba es confiable.

### 4.3.3 Prueba 3 Identificación de parámetros de otro Motor de CD.

Con el fin de validar las señales de entrada que se deben aplicar a un motor de CD, para obtener un conjunto de parámetros estimados con un error reducido, se utilizaron los siguientes parámetros de otro motor de CD [Lázaro, Rico, Ocaña, 2007]:

Tabla 4.2 Parámetros de Motor CD.

Voltaje de entrada:	24 Volts.
Resistencia del devanado de armadura:	1.21 Ohms.
Inductancia del devanado:	0.00584 Henrios.
Constante contraelectromotriz:	0.008574(N-m/A).
Inercia del rotor del motor:	0.0000142 (kg-m <sup>2</sup> ).
Constante de fricción viscosa:	0.0000245 (N-m-s).

En la simulación se aplicó una entrada tipo escalón de 24 Volts, dadas las características del motor de CD. Al aplicar las pruebas anteriores haciendo uso de la

interfase gráfica y considerando  $m = 1024$  número de términos y truncar a  $n = 512$ , con un tiempo de respuesta de 1 segundo, se obtuvieron los resultados de la Figura 4.11.

Parámetros Reales del Motor	Parámetros Estimados Por Series Walsh	Margen de Error Entre Parámetros Estimados y Reales
Resistencia de Armadura 1.21 Ohms	Resistencia de Armadura 1.2106 Ohms	Error Ra 4.83E-2 %
Inductancia de Armadura 5.84E-3 Henrios	Inductancia de Armadura 5.98E-3 Henrios	Error La 2.46E+0 %
Constante de Velocidad 8.57E-3 N-m/A	Constante de Velocidad 8.57E-3 N-m/A	Error K 1.9E-2 %
Inercia del Rotor 1.42E-5 kg-m <sup>2</sup>	Inercia del Rotor 1.43E-5 kg-m <sup>2</sup>	Error J 0.493 %
Coefficiente de Fricción 2.45E-5 N-m-s	Coefficiente de Fricción 2.49E-5 N-m-s	Error F 1.8 %

Figura 4.11 Resultados de los parámetros estimados.

Por otro lado el resultado de las variables de estado para este caso, se tienen en la Figura 4.12.

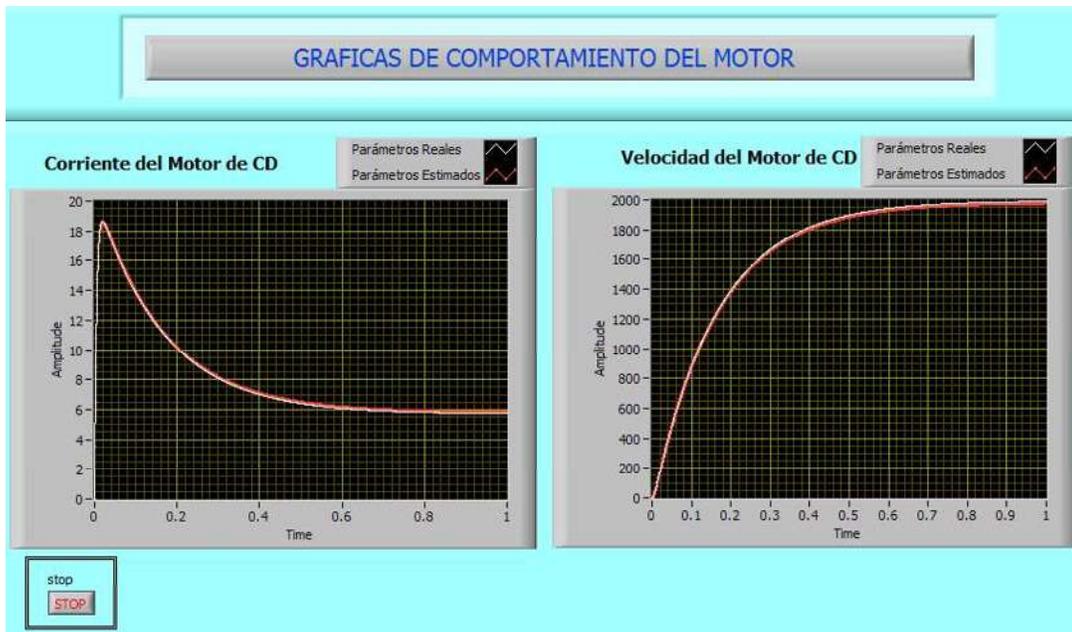


Figura 4.12 Variables de estado de un Motor de CD diferente.

El comportamiento de este Motor es parecido al simulado, en la Figura 4.3, tomando como referencia los parámetros de la Tabla 4.1, aquí la diferencia que se tiene,

es el tiempo de respuesta como en el arranque, así como en la corriente.

#### 4.3.4 Prueba 4 Entrada tipo rampa.

Para observar el comportamiento del sistema se realiza una prueba, con una entrada tipo rampa, los términos  $m = 1024$  de la FWT y truncando a  $n = 512$  de la serie de Walsh, como resultado se tiene la Figura 4.13.

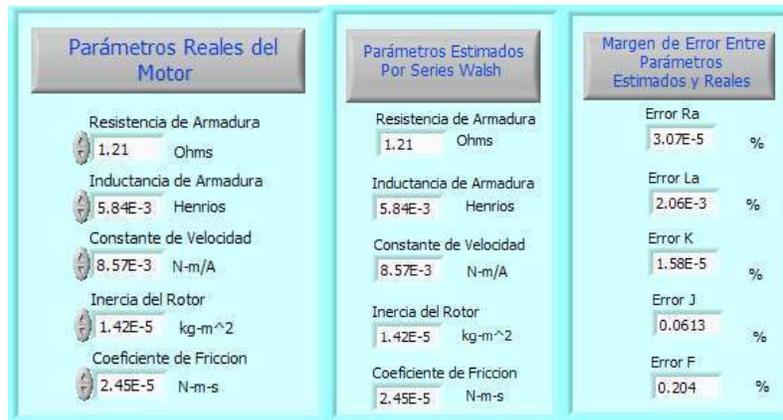


Figura 4.13 Resultado de parámetros estimados con entrada rampa.

Claramente en la Figura 4.13 se aprecian resultados más precisos con la entrada rampa, con respecto a las otras entradas simuladas.

#### 4.3.5 Prueba 5 Utilización de la Redundancia Analítica al Proceso de Identificación de Parámetros del Motor de CD del Caso Anterior.

En esta última prueba se aplica la redundancia analítica, ya que se emplean tres tipos de señales, una de tipo escalón, otra que es un tren de pulsos la cual opera a una frecuencia de 4 Hz y por último de tipo rampa, como se muestra en la leyenda de gráfico de la Figura 4.14. Al analizar el sistema usando la interfase con  $m = 1024$  términos y truncando a  $n = 128$  de la serie de Walsh, se obtuvieron los resultados de la Figura 4.14.

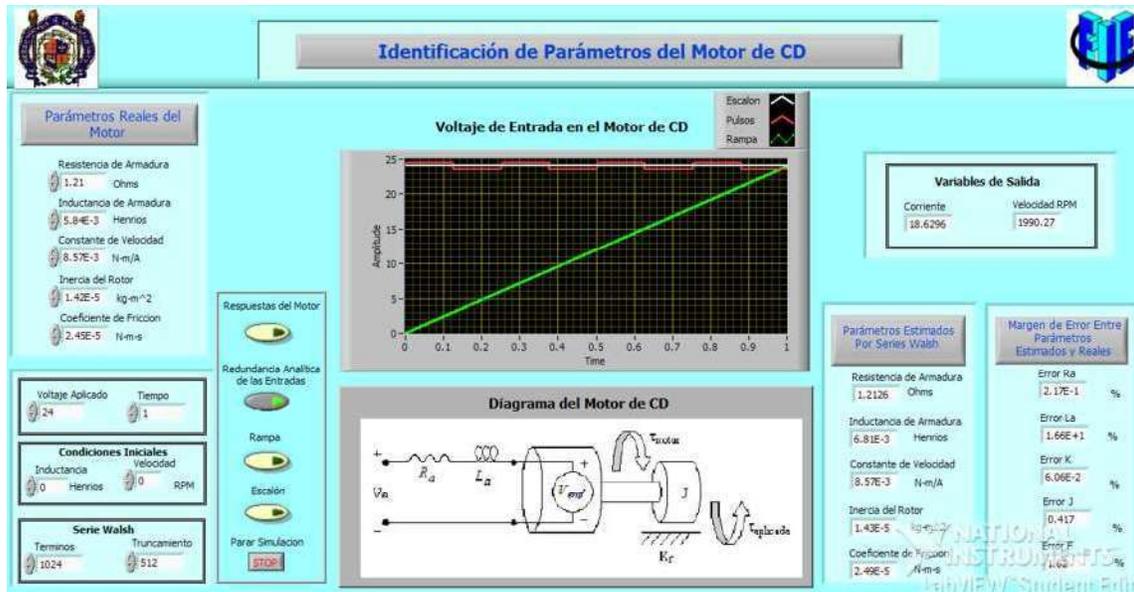


Figura 4.14 Panel Frontal aplicación de la redundancia analítica a un Motor de CD diferente.

Al aplicar la redundancia analítica se hace una suma de entradas, para evitar al sistema y exista una disminución en el error. Para este caso estudio se no se aplica la entrada impulso, debido a que se utilizan entradas de operación. La ventaja de esta plataforma, considerando que la señal de entrada para un Motor de CD, es de tipo escalón, puede simular cualquier tipo de Motor de CD.

# Capítulo 5

## Conclusiones

### 5.1 Conclusiones.

La técnica presentada para la obtención de parámetros, permite obtener una expresión algebraica, que hace uso de las matrices operacionales de integración y la cual permite realizar la identificación de sistemas lineales invariantes en el tiempo, de una manera sencilla.

El error promedio disminuye a medida que el número de términos tomados para la serie ortogonal aumenta, se pueden observar errores menores al 5% en algunos casos, esto depende del número de términos tomados de la serie de Walsh y de la redundancia analítica.

La interfaz gráfica diseñada permite probar la estimación de parámetros de cualquier motor de CD controlado por armadura, siempre que corresponda a una entrada escalón.

Para la aplicación de métodos de diagnóstico y el desarrollo de controladores, para la detección de fallas, es necesaria una identificación paramétrica, para ello es necesario tener un conocimiento dinámico matemático del sistema a identificar, lo cual se puede lograr usando la interfase desarrollada, para el caso específico de los motores de CD.

Se aplica la redundancia analítica al sistema para suponer la posibilidad de realizar varios experimentos sobre el sistema a identificar, esto con el objetivo de excitar algunos parámetros que no son perceptibles con ciertos tipos de entrada. Para algunos casos la redundancia analítica no es tan efectiva, ya que algunas entradas pueden afectar considerablemente a otras entradas, y los resultados de la simulación pueden variar.

Se presenta una interfaz gráfica por la necesidad de tener un instrumento personalizado para la identificación de parámetros de motores de CD, el sistema simula

el comportamiento del motor con respecto a su modelo dinámico, aplicando la formulación matemática que compone este sistema.

Para un motor de CD se observó en esta simulación que puede ser excitado sin ningún problema, utilizando las señales de entrada que corresponde a un escalón, tren de pulsos y una rampa.

En esta plataforma desarrollada en LabVIEW®, se observa la factibilidad de manejo que se tiene directamente con ella, la cual la hace sumamente amigable y susceptible a modificaciones, para futuras adecuaciones en la identificación de parámetros fuera de línea para una captura de datos reales.

## **5.2 Trabajos Futuros.**

Lo desarrollado en este trabajo se puede tomar como referencia para poder realizar una etapa de instrumentación, la cual permita obtener una colección de datos reales, relacionados con las variables de estado del motor y posteriormente realizar la identificación de parámetros del mismo, lo cual permitiría aplicar la toma en tiempo real.

Una de las desventajas que se tuvo, al realizar las pruebas fue en la aplicación de la redundancia analítica, ya que para el análisis de diferentes motores era necesario modificar algunas entradas del código en Matlab; para poder resolver esta desventaja, se podría realizar en un trabajo futuro, modificar la plataforma original para que desde la interfase se realicen las modificaciones a los diferentes tipos de entrada aplicados a motores diferentes.

Se puede aplicar el método de identificación paramétrica junto con otras herramientas que permitan el diagnóstico de fallas.

## Apéndice A

# Código empleado para la Identificación Paramétrica

### A.1 Código principal implementado para el cálculo de los parámetros.

Códigos empleados para la identificación de parámetros usando series Walsh en MatLab®

#### A. 1.1 Código utilizado para el cálculo de los parámetros aplicando una entrada escalón.

```
%MOTOR DE CD
%Solución por series Walsh utilizando matriz de integración
%ia es la corriente en el inductor y w es la velocidad angular
m=m1;
x1=x11;
x2=x22;
ra=ra1;
n=n1;
j=j1;
fr=fr1;
k=k1;
la=la1;
T=TT1;
Va=Va1
```

```

global x1;
global x2;
xo=[x1 x2]'; %condiciones iniciales
global m;
global n;
global T;
Ts=T/m;
t=0:Ts:T-Ts;
%Parámetros del motor de CD
global ra;
global la;
global k;
global j
global fr;
global Va;
options = odeset('RelTol',1e-8,'AbsTol',[1e-6 1e-6]);
[t1,xx1]=ode45('fmotor1',t,xo,options);
z=zeros(size(t1));
P=pwalsh(m);%Creamos la matriz producto
%Diferentes entradas al sistema
u=(ones(size(t1)))*Va;% entrada escalón de 24 V.
%Entradas para visualizar las entradas en Labview
uout=u';
%Algoritmo para obtener la entrada Escalón.
[u,SO]=fwt(u);% Obtenemos los coeficientes de la entrada escalo de 24 volts
u=u/m;%los dividimos en los términos
u(n:m)=z(n:m);
[x1n,SO]=fwt(xx1(:,1));%Transformada de la corriente del modelo
X1an=x1n/m;
X1=X1an;
X1(n:m)=z(n:m);

```

```

[x2n,SO]=fwt(xx1(:,2));%Transformada en el voltaje del modelo
X2an=x2n/m;
X2=X2an;
X2(n:m)=z(n:m);
IaP=T*X1*P;
wP=T*X2*P;
Va=T*u*P;
elm=zeros(size(IaP));
elm(:,1)=1;
ZT=[IaP;wP;Va;elm];
format long;%Escala para 15 dígitos
ct=[X1;X2];
%Se aplica la pseudoinversa, aproximación de puntos.
TOL=1e-8;
Z1inv=ZT'*inv(ZT*ZT');
%Z1inv=ZTT'*inv(ZTT*ZTT');
%Z1inv=pinv(ZT,TOL)
%este TETA ES MATRIZ
%teta=c1'*Z1inv;
teta=ct*Z1inv;
tetaLa=1/teta(1,3)
tetaRa=-teta(1,1)*tetaLa
tetaK=-teta(1,2)*tetaLa
tetaJ=tetaK/teta(2,1)
tetaF=-teta(2,2)*tetaJ
%V=teta;
%obtener(V)
V=[tetaLa tetaRa tetaK tetaJ tetaF];
save fichero.txt V -ascii
%Sacando el error en % de la real y la estimada por Walsh
errorRa=abs((ra-tetaRa)/ra)*100

```

```

errorLa=abs((la-tetaLa)/la)*100
errorK=abs((k-tetaK)/k)*100
errorJ=abs((j-tetaJ)/j)*100
errorF=abs((fr-tetaF)/fr)*100
X1=ifwt(X1,SO);
X2=ifwt(X2,SO);
%XXX1=ifwt(XXX1,SO);
%XXX2=ifwt(XXX2,SO);
f1=xx1(:,1)';
tout=t1';
f2=xx1(:,2)';
%f3=ww1(:,1)';
%f4=ww1(:,2)';
I=max(xx1(:,1));
W=max(xx1(:,2));

```

### **A. 1.2 Código utilizado para el cálculo de los parámetros aplicando una entrada rampa.**

```

%MOTOR DE CD ENTRADA RAMPA
%Solución por series Walsh utilizando matriz de integración
%ia es la corriente en el inductor y w es la velocidad angular
m=m1;
x1=x11;
x2=x22;
ra=ra1;
n=n1;
j=j1;
fr=fr1;
k=k1;
la=la1;
T=TT1;

```

```
Va=Va1;

global x1;
global x2;
xo=[x1 x2]'; %condiciones iniciales
global m;
global n;
global T;
Ts=T/m;
t=0:Ts:T-Ts;
%Parámetros del motor de CD
global ra;
global la;
global k;
global j
global fr;
global Va
options = odeset('RelTol',1e-8,'AbsTol',[1e-6 1e-6]);
[t4,hh1]=ode45('fmotor4',t,xo,options);
z=zeros(size(t4));
P=pwalsh(m);%Creamos la matriz producto
%Diferentes entradas al sistema
u4=t4*(Va*25);
u4out=u4';
%Entradas para visualizar las entradas en Labview
%plot(t4,u4)
%Algoritmo para obtener un entrada de rampa.
[u4,SO]=fwt(u4);
u4=u4/m;
u4(n:m)=z(n:m);
[xxx1n,SO]=fwt(hh1(:,1));
```

```

XXX1an=xxx1n/m;
XXX1=XXX1an;
XXX1(n:m)=z(n:m);
[xxx2n,SO]=fwt(hh1(:,2));
XXX2an=xxx2n/m;
XXX2=XXX2an;
XXX2(n:m)=z(n:m);
I4aP=T*XXX1*P;
w4P=T*XXX2*P;
V4a=T*u4*P;
elm=zeros(size(I4aP));
elm(:,1)=1;
ZT=[I4aP;w4P;V4a;elm];
format long;%Escala para 15 digitos
ct=[XXX1;XXX2];
%Se aplica la pseudoinverza, aproximación de puntos.
TOL=1e-8;
Z1inv=ZT'*inv(ZT*ZT');
%Z1inv=ZTT'*inv(ZTT*ZTT');
%Z1inv=pinv(ZT,TOL)
%este TETA ES MATRIZ
%teta=c1'*Z1inv;
teta=ct*Z1inv;
tetaLa=1/teta(1,3)
tetaRa=-teta(1,1)*tetaLa
tetaK=-teta(1,2)*tetaLa
tetaJ=tetaK/teta(2,1)
tetaF=-teta(2,2)*tetaJ
V=[tetaLa tetaRa tetaK tetaJ tetaF];
save fichero.txt V -ascii
%Sacando el error en % de la real y la estimada por Walsh

```

```

errorRa=abs((ra-tetaRa)/ra)*100
errorLa=abs((la-tetaLa)/la)*100
errorK=abs((k-tetaK)/k)*100
errorJ=abs((j-tetaJ)/j)*100
errorF=abs((fr-tetaF)/fr)*100
f1=hh1(:,1)';
tout=t4';
f2=hh1(:,2)';
I=max(hh1(:,1));
W=max(hh1(:,2));

```

### **A. 1.3 Código utilizado para el cálculo de los parámetros aplicando la redundancia analítica.**

```

%MOTOR DE CD APLICANDO LA REDUNDANCIA ANALITICA
%Solucion por series Walsh utilizando matriz de integracion
%ia es la corriente en el inductor y w es la velocidad angular
m=m1;
x1=x11;
x2=x22;
ra=ra1;
n=n1;
j=j1;
fr=fr1;
k=k1;
la=la1;
T=TT1;
Va=Va1;
global x1;
global x2;
xo=[x1 x2]'; %condiciones iniciales
global m;

```

```

global n;
global T;
Ts=T/m;
t=0:Ts:T-Ts;
%Parametros del motor de CD
global ra;
global la;
global k;
global j
global fr;
global Va;
options = odeset('RelTol',1e-8,'AbsTol',[1e-6 1e-6]);
[t1,xx1]=ode45('fmotor1',t,xo,options);
[t2,yy1]=ode45('fmotor2',t,xo,options);
[t4,hh1]=ode45('fmotor4',t,xo,options);
[t5,jj1]=ode45('fmotor5',t,xo,options);
[t6,kk1]=ode45('fmotor6',t,xo,options);
z=zeros(size(t1));
P=pwalsh(m);%Creamos la matriz producto
%Diferentes entradas al sistema
u=(ones(size(t1)))*Va;% entrada escalon de 24 V.
uu=((sign(sin((t1*(600.000000)))))+(Va*2))/2;%Entrada Formando un tren de pulsos de
24 Volts
%uu=((sign(sin((t1*(30.000000)))))+48)/2;%Entrada Formando un tren de pulsos de 24
Volts
valor=5000;%62831.8530718;
u5=24+.3*sin(valor*t1)+.15*sin(3*(valor*t1))+.01*sin(5*(valor*t1));
u51=.3*sin(valor*t1)+.15*sin(3*(valor*t1))+.01*sin(5*(valor*t1));
u6=u4+u51;
%Entradas para visualizar las entradas en Labview
uout=u';

```

```

uuout=uu';
u4out=u4';
%Algoritmo para obtener la entrada Escalon.
[u,S0]=fwt(u);% Obtenemos los coeficientes de la entrada escalon de 24 volts
u=u/m;%los dividimos en los terminos
u(n:m)=z(n:m);
[x1n,S0]=fwt(xx1(:,1));%Transformada de la corriente del modelo
X1an=x1n/m;
X1=X1an;
X1(n:m)=z(n:m);
[x2n,S0]=fwt(xx1(:,2));%Transformada en el voltaje del modelo
X2an=x2n/m;
X2=X2an;
X2(n:m)=z(n:m);
IaP=T*X1*P;
wP=T*X2*P;
Va=T*u*P;
%Algoritmo para obtener la entrada de tren de pulsos de 24 volts.
[uu,S0]=fwt(uu);
uu=uu/m;
uu(n:m)=z(n:m);
[xx1n,S0]=fwt(yy1(:,1));
XX1an=xx1n/m;
XX1=XX1an;
XX1(n:m)=z(n:m);
[xx2n,S0]=fwt(yy1(:,2));
XX2an=xx2n/m;
XX2=XX2an;
XX2(n:m)=z(n:m);
I2aP=T*XX1*P;
w2P=T*XX2*P;

```

```

V2a=T*uu*P;
%Algoritmo para obtener un entrada de rampa.
[u4,SO]=fwt(u4);
u4=u4/m;
u4(n:m)=z(n:m);
[xxx1n,SO]=fwt(hh1(:,1));
XXX1an=xxx1n/m;
XXX1=XXX1an;
XXX1(n:m)=z(n:m);
[xxx2n,SO]=fwt(hh1(:,2));
XXX2an=xxx2n/m;
XXX2=XXX2an;
XXX2(n:m)=z(n:m);
I4aP=T*XXX1*P;
w4P=T*XXX2*P;
V4a=T*u4*P;
%Algoritmo para aproximar una funcion de rampa formado por un tren de pulsos
[u5,SO]=fwt(u5);
u5=u5/m;
u5(n:m)=z(n:m);
[xxxx1n,SO]=fwt(jj1(:,1));
XXXX1an=xxxx1n/m;
XXXX1=XXXX1an;
XXXX1(n:m)=z(n:m);
[xxxx2n,SO]=fwt(jj1(:,2));
XXXX2an=xxxx2n/m;
XXXX2=XXXX2an;
XXXX2(n:m)=z(n:m);
I5aP=T*XXXX1*P;
w5P=T*XXXX2*P;
V5a=T*u5*P;

```

%Algoritmo para aproximar una función de rampa formando una rampa sumada a un tren de pulso

```
[u6,SO]=fwt(u6);
u6=u6/m;
u6(n:m)=z(n:m);
[xx61n,SO]=fwt(kk1(:,1));
XX61an=xx61n/m;
XX61=XX61an;
XX61(n:m)=z(n:m);
[xx62n,SO]=fwt(kk1(:,2));
XX62an=xx62n/m;
XX62=XX62an;
XX62(n:m)=z(n:m);
I6aP=T*XX61*P;
w6P=T*XX62*P;
V6a=T*u6*P;
elm=zeros(size(IaP));
elm(:,1)=1;

Z1=[IaP;wP;Va;elm];
Z2=[I2aP;w2P;V2a;elm];
Z4=[I4aP;w4P;V4a;elm];
Z5=[I5aP;w5P;V5a;elm];
Z6=[I6aP;w6P;V6a;elm];
ZT=[Z1 Z2 Z4];
format long;%Escala para 15 digitos
c1=[X1;X2];
c2=[XX1;XX2];
c4=[XXX1;XXX2];
c5=[XXXX1;XXXX2];
c6=[XX61;XX62];
```

```

ct=[c1 c2 c4];
%Se aplica la pseudoinversa, aproximación de puntos.
TOL=1e-8;
Z1inv=ZT'*inv(ZT*ZT');
%este TETA ES MATRIZ
teta=ct*Z1inv;
tetaLa=1/teta(1,3)
tetaRa=-teta(1,1)*tetaLa
tetaK=-teta(1,2)*tetaLa
tetaJ=tetaK/teta(2,1)
tetaF=-teta(2,2)*tetaJ
V=[tetaLa tetaRa tetaK tetaJ tetaF];
save fichero.txt V -ascii
%Sacando el error en % de la real y la estimada por Walsh
errorRa=abs((ra-tetaRa)/ra)*100
errorLa=abs((la-tetaLa)/la)*100
errorK=abs((k-tetaK)/k)*100
errorJ=abs((j-tetaJ)/j)*100
errorF=abs((fr-tetaF)/fr)*100
X1=ifwt(X1,SO);
X2=ifwt(X2,SO);
f1=xx1(:,1)';
tout=t1';
f2=xx1(:,2)';
I=max(xx1(:,1));
W=max(xx1(:,2));

```

## **A.2 Código principal implementado para la graficación de los parámetros estimados y los reales.**

```

%MOTOR DE CD
%Solución por series Walsh utilizando matriz de integración

```

%ia es la corriente en el inductor y w es la velocidad angular

```
m=m1;
x1=x11;
x2=x22;
ra=ra1;
n=n1;
j=j1;
fr=fr1;
k=k1;
la=la1;
T=TT1;
Va=Va1;
global x1;
global x2;
xo=[x1 x2]'; %condiciones iniciales
global m;
global n;
global T;
Ts=T/m;
t=0:Ts:T-Ts;
%Parámetros del motor de CD
global ra;
global la;
global k;
global j;
global fr;
global Va;
options = odeset('RelTol',1e-8,'AbsTol',[1e-6 1e-6]);
[t1,xx1]=ode45('fmotor1',t,xo,options);
[t2,yy1]=ode45('fmotor2',t,xo,options);
[t3,ww1]=ode45('fmotor3',t,xo,options);
```

```

tout=t1';
f1=xx1(:,1)';
f2=xx1(:,2)';
f3=ww1(:,1)';
f4=ww1(:,2)';

```

### A.3 Funciones Empleadas.

En estas funciones se emplean los diferentes tipos de entradas para la identificación de parámetros de nuestro motor.

#### A.3.1 Función con una entrada escalón de 24 V.

```

function dydx=fmotor1(t,x)
%Parámetros de los componentes del circuito.
global ra;
global la;
global k;
global j
global fr;
global Va;
u=(ones(size(t)))*Va;
a11=(-ra/la);
a12=(-k/la);
a21=(k/j);
a22=(-fr/j);
b11=(1/la);
A=[a11 a12;a21 a22];
B=[b11 0]';
dydx=A*x+B*u;
end

```

**A.3.3 Función que procesa los parámetros estimados.**

```
function dydx=fmotor3(t,x)
%Parámetros de los componentes del circuito.
global Va;
load fichero.txt -ascii;
fichero;
teRa=fichero(1,2);
teLa=fichero(1,1);
teK=fichero(1,3);
teJ=fichero(1,4);
teF=fichero(1,5);
uu3=(ones(size(t)))*Va;
a11=(-teRa/teLa);
a12=(-teK/teLa);
a21=(teK/teJ);
a22=(-teF/teJ);
b11=(1/teLa);
A=[a11 a12;a21 a22];
B=[b11 0]';
dydx=A*x+B*uu3;
end
```

**A.3.4 Función que simula una rampa para la entrada al motor.**

```
function dydx=fmotor4(t,x)
%Parámetros de los componentes del circuito.
global ra;
global la;
global k;
global j;
global fr;
global Va;
```

```

u4=t*(Va*25);
a11=(-ra/la);
a12=(-k/la);
a21=(k/j);
a22=(-fr/j);
b11=(1/la);
A=[a11 a12;a21 a22];
B=[b11 0]';
dydx=A*x+B*u4;
end

```

### A.3.5 Programa utilizado para el producto de funciones.

```

%Producto de dos funciones
%Aproximación de funciones por series Walsh
m=16;
n=8; %Numero de truncamiento
%T=16.66E-3;
T=1;
Ts=T/m;
t=0:Ts:T-Ts;% Intervalo de tiempo
z=zeros(size(t));
u1=t; % Función entrada rampa
u=u1;
%u2=24*sin(pi*t); % Función entrada senoidal
u2=(ones(size(t)))*24;
us=u2;
u3=sin(pi*t);
up=u3;
P=pwalsh(m);
[u1,SO]=fwt(u1);%Aplicando La transformada rápida de Walsh
u1=u1/m;

```

```
u1(n:m)=z(n:m)
[u3,SO]=fwt(u3);%Aplicando La transformada rápida de Walsh
u3=u3/m;
u3(n:m)=z(n:m)
U3=cwalsh(u3)
X1=ifwt(u1,SO);
X2=ifwt(u2,SO);
X3=ifwt(u3,SO);
UU1=u1;
UU3=u3;
Pro=UU1;
Pro1=U3;
utotal=u.*up
total=Pro*Pro1
ypro=ifwt(total,SO)
plot(t,ypro,t,utotal);
```

# Apéndice B

## Hoja de Datos del Fabricante del Motor Simulado



**GM14904S016**

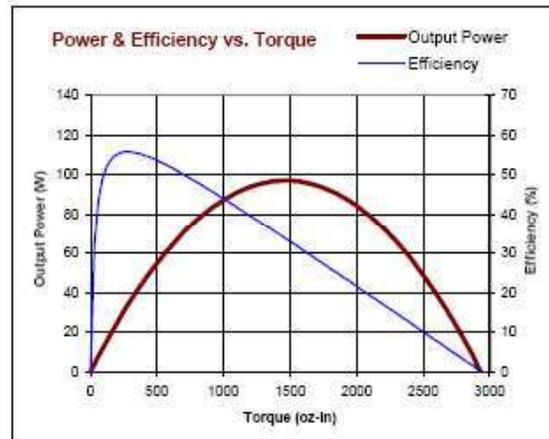
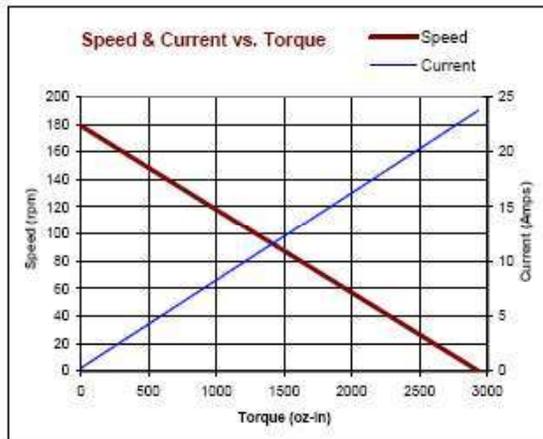
Lo-Cog® DC Servo Gearmotor



Assembly Data	Symbol	Units	Value	
Reference Voltage	E	V	24	
No-Load Speed	$S_{NL}$	rpm (rad/s)	179	(18.7)
Continuous Torque (Max.) <sup>1</sup>	$T_C$	oz-in (N-m)	374	(2.8)
Peak Torque (Stall) <sup>2</sup>	$T_{PK}$	oz-in (N-m)	2934	(20.7)
Weight	$W_M$	oz (g)	44.7	(1268)
Motor Data				
Torque Constant	$K_T$	oz-in/A (N-m/A)	8.67	(6.12E-02)
Back-EMF Constant	$K_E$	V/krpm (V/rad/s)	8.41	(6.12E-02)
Resistance	$R_T$	$\Omega$	1.01	
Inductance	L	mH	1.6	
No-Load Current	$I_{NL}$	A	0.26	
Peak Current (Stall) <sup>2</sup>	$I_P$	A	23.8	
Motor Constant	$K_M$	oz-in/ $\sqrt{W}$ (N-m/ $\sqrt{W}$ )	8.63	(6.09E-02)
Friction Torque	$T_F$	oz-in (N-m)	1.6	(1.1E-02)
Rotor Inertia	$J_M$	oz-in-s <sup>2</sup> (kg-m <sup>2</sup> )	3.7E-03	(2.6E-05)
Electrical Time Constant	$\tau_E$	ms	1.58	
Mechanical Time Constant	$\tau_M$	ms	7.0	
Viscous Damping	D	oz-in/krpm (N-m-s)	0.18	(1.2E-05)
Damping Constant	$K_D$	oz-in/krpm (N-m-s)	55	(3.7E-03)
Maximum Winding Temperature	$\theta_{MAX}$	$^{\circ}F$ ( $^{\circ}C$ )	311	(155)
Thermal Impedance	$R_{TH}$	$^{\circ}F/watt$ ( $^{\circ}C/watt$ )	45.8	(7.7)
Thermal Time Constant	$\tau_{TH}$	min	28.8	
Gearbox Data				
Reduction Ratio			19.7	
Efficiency <sup>2</sup>			0.84	
Maximum Allowable Torque		oz-in (N-m)	500	(3.53)
Encoder Data				
Channels			3	
Resolution		CPR	500	

1 - Specified at max. winding temperature at 25°C ambient without heat sink. 2 - Theoretical values supplied for reference only.  
 2 - Effective gearbox efficiency for this unit improved by use of ball bearings.

<p><b>Included Features</b></p> <ul style="list-style-type: none"> <li>2-Pole Stator</li> <li>Ceramic Magnets</li> <li>Heavy-Gauge Steel Housing</li> <li>11-Slot Armature</li> <li>Silicon Steel Laminations</li> <li>Stainless Steel Shaft</li> <li>Copper-Graphite Brushes</li> <li>Diamond Turned Commutator</li> <li>Motor Ball Bearings</li> <li>Output Ball Bearing</li> <li>Wide Face Gears</li> </ul>
<p><b>Customization Options</b></p> <ul style="list-style-type: none"> <li>Alternate Winding</li> <li>Sleeve or Ball Bearings</li> <li>Modified Output Shaft</li> <li>Custom Cable Assembly</li> <li>Special Brushes</li> <li>EMI/RFI Suppression</li> <li>Alternate Gear Material</li> <li>Special Lubricant</li> <li>Optional Encoder</li> <li>Fail-Safe Brake</li> </ul>



All values are nominal. Specifications subject to change without notice. Graphs are shown for reference only.

© 2001 Pittman.

PITTMAN, 343 Godshall Drive, Harleysville, PA 19438, Phone: 877-PITTMAN, Fax: 215-258-1338, E-mail: [info@pittman.net](mailto:info@pittman.net), Web Site: [www.pittman.net](http://www.pittman.net)

# Referencias

[Lázaro, Ramírez 2008]

Isidro I. Lázaro Castillo, Salvador Ramírez Z., “Identificación de Sistemas Lineales Usando Matrices Operacionales”, Encuentro de Investigación de Ingeniería Eléctrica ENINVIE Abril de 2008, Zacatecas, Zac.

[I. Lázaro, 2008]

Libro. “Ingeniería de Sistemas de Control Continuo” Isidro I. Lázaro Castillo Primera Edición, Facultad de Ingeniería Eléctrica, División de Estudios de Posgrado de la UMSNH.

[Avalos, 2007]

Avalos C.” Identificación de Parámetros de un Controlador CD-CD tipo Boost con series Walsh”, Tesis de Licenciatura, Facultad de Ingeniería Eléctrica de la UMSNH, Febrero 2007.

[National Instruments, 2004]

Libro de LabVIEW, “Learning with LabVIEW 7 Express” Robert H. Bishop The University of Texas at Austin.

[Curso de Instrumentación Virtual, 2003]

Introducción a LabVIEW, “Curso de Seis-Horas de LabVIEW 7 Express” Septiembre 2003 Edición.

[FMI, 2006]

Curso de MatLab, Departamento de Matemáticas Aplicadas, Universidad de Zaragoza. [www.unizar.es/fmi](http://www.unizar.es/fmi) versión actualizada en agosto de 2006.

[MATLAB, 2005]

Libro de MATLAB, “An Introduction With Applications” Amos Gilat Second Edition, Department of Mechanical Engineering. The Ohio State University.

[IEEE, 2007]

IEEE Control Systems Magazine, “System identification comprises a collection of activities whose ultimate purpose is to produce a model meeting user requirements” Daniel E. Rivera, Spilios D. Fassois, October 2007.

[MATLAB 6.1, 2001]

Manual MATLAB, “Aprenda Matlab 6.1 como si estuviera en primero” Javier García de Jalón, José Ignacio Rodríguez, Alfonso Brazález, Madrid Octubre 2001.

[Ljung, 1999]

“*System identification*”. Theory for the user. Prentice Hall. 1999.

[Lázaro, 1999]

Lázaro I. L., “Análisis transitorio y de estado estable en el dominio de la frecuencia de redes eléctricas invariantes y variantes en el tiempo utilizando el enfoque algebraico de las matrices operacionales”, Tesis de Maestría, División de Estudios de Posgrado de la Facultad de Ingeniería Eléctrica de la UMSNH, Febrero 1999.

[Lázaro, Avalos, Anzures, 2008]

Lázaro I. Castillo, Cesar Ávalos C, Juan Anzures M. “Identificación de parámetros de un convertidor de CD-CD”, Encuentro de Investigación en Ingeniería Eléctrica, Zacatecas, Zac, México, Marzo 2008.

[Barrera, Madrigal, 2007]

Edmundo Barrera C, Stalin Madrigal S. “Identificación de Sistemas Lineales Invariantes en el Tiempo Utilizando Matrices Operacionales” ITCH-ELECTRON, Octubre 17 – 19, 2007 Chihuahua, México.

[Herrera, et. al, 2006]

Roberto Herrera, Luis A. González, David Saucedo, Adolfo Esquivel. “Plataforma de Control Para Control de Imán Permanente” Encuentro de Investigación de Ingeniería Eléctrica, Zacatecas, Zac Abril 5 – 7, 2006.

[EPSEM, 2008]

Departamento ESAIL, “Escuela Politécnica Superior de Ingeniería de Manresa” [www.epsem.upc.edu/~esail/assign/ident/tema1 .pdf](http://www.epsem.upc.edu/~esail/assign/ident/tema1.pdf).

[Sedano,Villar, 2005]

Javier Sedano Franco y José Ramón Villar Flecha, “Introducción a la Identificación de Sistemas” Técnica Industrial 256, Marzo 2005.

[Lázaro, Rico, Ocaña, 2007]

Isidro I. Lázaro Castillo, J. Jesús Rico Melgoza, Edgar Ocaña Zavala. “Identificación de Sistemas Lineales Usando Cálculo Operacional” División de Estudios de Posgrado, Facultad de Ingeniería Eléctrica de la UMSNH.

[Corrington, 1973]

Murlan S. Carrington. “Solution of Differential and Integral Equations with Walsh Functions” IEEE Transactions on Circuit Theory, Vol. CT-20, No. 5, September. 1973.

[Alvarado, 2001]

Alvarado N., “Identificación de Cargas ante Condiciones no Senoidales, usando Técnicas de Estimación en el Dominio de la Frecuencia mediante Matrices Operacionales”, Tesis de Maestría, División de Estudios de Posgrado de la Facultad de Ingeniería Eléctrica de la UMSNH, Enero 2001.