

**ANÁLISIS COMPARATIVO DEL DESEMPEÑO DE TÉCNICAS DE CONTROL
CONMUTADO IMPLEMENTADAS EN DISPOSITIVOS PROGRAMABLES**



FABIO ANDRÉS MORENO CASTELLANOS

DIEGO ARMANDO CENTENO ARIZA



**ESCUELA DE INGENIERÍAS
ELÉCTRICA, ELECTRÓNICA Y
DE TELECOMUNICACIONES**



UNIVERSIDAD INDUSTRIAL DE SANTANDER

FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS

**ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES**

BUCARAMANGA 2013

**ANÁLISIS COMPARATIVO DEL DESEMPEÑO DE TÉCNICAS DE CONTROL
CONMUTADO IMPLEMENTADAS EN DISPOSITIVOS PROGRAMABLES**

FABIO ANDRÉS MORENO CASTELLANOS

DIEGO ARMANDO CENTENO ARIZA

**Trabajo de grado presentado como requisito parcial para optar al título de
ingeniero electrónico**

Director

RICARDO ALZATE CASTAÑO, PhD

Codirector

CARLOS AUGUSTO FAJARDO ARIZA, MSc

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO-MECÁNICAS
ESCUELA DE INGENIERÍAS ELÉCTRICA, ELECTRÓNICA Y DE
TELECOMUNICACIONES
BUCARAMANGA 2013**

Dedicatoria

A Dios por brindarme siempre su guía y su afecto.

*A mi mamá Gladys por ser siempre mi ejemplo y ser la base de
nuestra familia.*

*A mi papá William por estar siempre presente cuando más lo
necesite y mostrarme el camino a seguir.*

*A mis hermanos que siempre estarán conmigo y prestaron especial
atención a éste trabajo.*

*A mi familia por estar conmigo siempre en especial a mis nonitas
Isaura y Aurora.*

Dedicatoria

A Dios, padre todopoderoso y a la siempre virgen María que me han acompañado en el camino de mi vida hasta este punto con bendiciones, amor y ternura.

A mis padres, hermanos, prima, tía por la paciencia, el amor, y la confianza que han depositado en mí.

A Nidia Mantilla por estar siempre ahí para escucharme, darme fuerzas y alentarme a seguir éste, el camino que he elegido.

A Daría Anfílogova por su dulce compañía, sus palabras de aliento y su presencia.

A mis amigos a todos y cada uno de ellos por su presencia en mi vida, por compartir esos momentos que ayudaron a formar la persona que ahora soy, por su compañía y buenos deseos, también les deseo lo mejor, a todos y a cada uno de ellos, a mi familia y a las personas que llevo en el corazón.

AGRADECIMIENTOS

A Ricardo Alzate, Director del proyecto, que con su paciencia y enorme colaboración con consejos, apoyo y amistad fue el pilar para la elaboración de este trabajo de grado.

A los calificadores Daniel Sierra y William Salamanca por sus consejos e ideas para mejorar el proyecto.

A la Universidad Industrial de Santander, en especial, la escuela de ingenierías eléctrica, electrónica y telecomunicaciones y sus profesores por la educación profesional brindada.

A todos nuestros compañeros con los cuales compartimos y nos llevamos los mejores recuerdos.

A el grupo CEMOS, ya que el presente proyecto de grado forma parte de las actividades del proyecto de investigación denominado: *“Análisis del Comportamiento Dinámico de un Convertidor de Potencia Sometido a Estrategias de Control Conmutado”*, financiado por la VIE-UIS con código 5568.

TABLA DE CONTENIDO

	Pag.
CAPITULO 1	19
1.1 ANTECEDENTES Y JUSTIFICACIÓN.....	19
1.2 PLANTEAMIENTO Y DEFINICIÓN DEL PROBLEMA.....	21
1.3 OBJETIVOS.....	22
1.3.1 OBJETIVO GENERAL.....	22
1.3.2 OBJETIVOS ESPECÍFICOS.....	22
CAPITULO 2	23
2.1 DESCRIPCIÓN GENERAL.....	22
2.1.1 Actuador.....	24
2.1.2 Motor.....	25
2.1.3 Cargas.....	25
2.1.4 Sensor.....	27
2.1.5 Dispositivo programable.....	28
2.1.6 Sistema definitivo.....	30
2.2 INTERFAZ DE MONITOREO.....	30
2.3 PROCEDIMIENTO PARA IDENTIFICACIÓN EXPERIMENTAL DEL MODELO DEL SISTEMA.....	32
2.3.1 Modelo para función de transferencia del sistema.....	34
CAPITULO 3	40
3.1 CONTROLADOR PID DIGITAL.....	40
3.1.1 Cálculo de los parámetros del controlador.....	41
3.1.2 Comportamiento dinámico del sistema sin control.....	42
3.1.3 Comportamiento del sistema controlado.....	44
3.2 CONTROLADOR POR MODOS DESLIZANTES.....	47
3.2.1 Implementación del SMC en MATLAB®.....	48
3.2.2 Implementación del SMC en Arduino MEGA®.....	52
3.3 CONTROL ÓPTIMO CONMUTADO.....	52

3.3.1 Implementación del controlador óptimo en MATLAB®.....	54
3.3.2 Implementación del controlador óptimo en Arduino MEGA®	58
CAPITULO 4.....	60
4.1 DESCRIPCIÓN GENERAL DEL LAZO DE CONTROL	60
4.1.1 Referencia discreta	61
4.1.2 Tiempo de muestreo	61
4.1.3 Lectura de velocidad en RPM	62
4.1.4 Generador de PWM	63
4.1.4.1 Máquina de estados	63
4.2 IMPLEMENTACIÓN DE CONTROLADORES EN VHDL	64
4.2.1 Control PID.....	64
4.2.1.1 Simulación PID	65
4.2.2 Control por modos deslizantes	67
4.2.2.1 Simulación de SMC	68
4.2.3 Control óptimo conmutado	68
4.2.3.1 Simulación de control óptimo	69
4.3 VERIFICACIÓN EXPERIMENTAL DEL DESEMPEÑO DE CONTROLADORES EN VHDL.....	70
4.3.1 Implementación del PID en FPGA.....	70
4.3.2 Implementación del SMC en FPGA.....	72
4.3.3 Implementación del control óptimo en FPGA	74
CONCLUSIONES	76
RECOMENDACIONES Y TRABAJO FUTURO	77
BIBLIOGRAFÍA.....	78

LISTA DE FIGURAS

	Pag.
Figura 1. Diagrama de bloques para el control de velocidad de un motor DC.	24
Figura 2. Circuito actuador del sistema	25
Figura 3. Acoplamiento de motor y generador para producir torque variable	26
Figura 4. Banco de resistencias de carga.	26
Figura 5. Esquema funcional de un encoder óptico.	27
Figura 6. Esquema del sistema de sensado implementado	27
Figura 7. Tarjeta Arduino MEGA 2650	28
Figura 8. Tarjeta Nexy 3 – Spartan 6.	29
Figura 9. Prototipo para pruebas.....	30
Figura 10. Diagrama de bloques para el sistema de adquisición/visualización de datos implementado	31
Figura 11. Panel frontal de la interfaz en LABVIEW	32
Figura 12. Diagrama de bloques del procedimiento experimental	32
Figura 13. Señales de respuesta (superior) y excitación (inferior) ante estímulos binarios pseudoaleatorios en prototipo	35
Figura 14. Identificación de regiones de saturación y banda muerta en el sistema.	35
Figura 15. Correlación de modelos para identificación de la relación de transferencia entre la velocidad angular en el eje del motor y el ciclo útil de entrada al circuito actuador.....	36
Figura 16. Superposición entre datos de salida (simulación + medidas experimentales) para validación ante un estímulo impulso.....	38
Figura 17. Superposición entre datos de salida (simulación + medidas experimentales) para validación ante un estímulo escalón.....	38
Figura 18. Superposición entre datos de salida (simulación + medidas experimentales) para validación ante un estímulo rampa.....	39
Figura 19. Comportamiento del sistema en lazo abierto con carga nominal.....	42
Figura 20. Efecto de perturbaciones de carga para el sistema en lazo abierto.....	43

Figura 21. Efecto de perturbaciones de carga para el sistema en lazo cerrado. ..	44
Figura 22. Comportamiento del sistema controlado por acción PID	45
Figura 23. Comportamiento del sistema controlado por acción PID con ajuste fino	46
Figura 24. Acción de control PID.....	47
Figura 25. Diagrama de flujo para implementación en MATLAB® de la ley de control por modo deslizante	48
Figura 26. Comportamiento del sistema controlado en el plano de fase sin perturbación, obtenido en MATLAB®.	49
Figura 27. Comportamiento de variables de estado para el sistema controlado sin perturbación, obtenido en MATLAB®.	50
Figura 28. Comportamiento del sistema controlado en el plano de fase con perturbación, obtenido en MATLAB®.	51
Figura 29. Comportamiento de variables de estado para el sistema controlado con perturbación, obtenido en MATLAB®.....	51
Figura 30. Comportamiento del sistema controlado por acción SMC	53
Figura 31. Señal de control del sistema controlado por acción SMC.	53
Figura 32. Diagrama de bloques implementación en MATLAB® de la ley de control óptimo conmutado.....	55
Figura 33. Salida de velocidad tras acción de control óptimo obtenida en MATLAB®.....	56
Figura 34. Salida de control óptimo conmutado obtenida en MATLAB®	57
Figura 35. Salida de velocidad tras acción de control óptimo con efecto de perturbaciones en la carga, obtenida en MATLAB®.....	57
Figura 36. Señal de control óptimo conmutado con efecto de perturbaciones en la carga, obtenida en MATLAB®	58
Figura 37. Comportamiento del sistema controlado por acción óptima conmutada.	59
Figura 38. Señal de control del sistema controlado por acción óptima conmutada	59
Figura 39. Circuito de propósito general para control en FPGA.....	60
Figura 40. Circuito de referencia discreta	61
Figura 41. Calculo de velocidad en RPM para control PID	62

Figura 42. Calculo de velocidad en RPM para SMC y óptimo conmutado.....	63
Figura 43. Máquina de estados del generador de PWM	64
Figura 44. Circuito implementado para el controlador PID.....	65
Figura 45. Simulación en <i>Isim</i> del controlador PID	66
Figura 46. Simulaciones en MATLAB y VHDL para control PID	66
Figura 47. Circuito implementado para el SMC.....	67
Figura 48. Simulaciones en MATLAB y VHDL para control SMC	68
Figura 49. Circuito implementado para el controlador óptimo conmutado	69
Figura 50. Simulaciones en MATLAB y VHDL para control óptimo conmutado ...	70
Figura 51. Comportamiento del sistema controlado por acción PID en FPGA.....	71
Figura 52. Acción de control PID en FPGA.....	72
Figura 53. Comportamiento del sistema controlado por acción SMC	73
Figura 54. Acción de control SMC en FPGA.....	73
Figura 55. Comportamiento del sistema controlado por acción óptima conmutada en FPGA	74
Figura 56. Acción de control óptimo conmutado en FPGA	75
Figura 57. Región deslizante en el plano de fase y la acción de control	83

LISTA DE TABLAS

	Pag.
Tabla 1. Características de motor modelo NF5475E	25
Tabla 2. Características de tarjeta Arduino MEGA 2650	28
Tabla 3. Características de la tarjeta Nexys 3 – Spartan 6.....	29
Tabla 4. Errores de validación para las diferentes señales de prueba utilizadas ..	39
Tabla 5. Parámetros de controlador PID calculados analíticamente	41
Tabla 6. Parámetros de controlador PID modificados tras ajuste fino	46
Tabla 7. Valores de parámetro seleccionados para determinar las cotas de incertidumbre	86

LISTA DE ANEXOS

	Pag.
ANEXO A	81
A.1 Fundamentos del SMC.....	81
A.1.1 Existencia del modo deslizante.....	82
A.1.2 Condición de deslizamiento	83
A.2 Diseño del controlador por modos deslizantes.....	84
ANEXO B	88
B.1 Fundamentos del control óptimo	88
B.2 Diseño del control óptimo	90

RESUMEN

Título: ANÁLISIS COMPARATIVO DEL DESEMPEÑO DE TÉCNICAS DE CONTROL CONMUTADO IMPLEMENTADAS EN DISPOSITIVOS PROGRAMABLES¹

Autores: FABIO ANDRÉS MORENO CASTELLANOS, DIEGO ARMANDO CENTENO ARIZA²

Palabras Clave:

Arduino, Control óptimo, FPGA, Modo deslizante, Motor DC.

Descripción:

El presente proyecto de grado aborda el diseño y la implementación de estrategias de control híbrido (conmutado) para manipular la dinámica de rotación del eje un motor DC. Se presenta la implementación de técnicas no-convencionales de control por modos deslizantes y control óptimo conmutado, comparando su desempeño con el control PID convencional. Las implementaciones se realizan tanto en una arquitectura computacional configurada (microcontrolador Arduino) como en una arquitectura configurable (FPGA). De los resultados obtenidos se observa una salida de velocidad uniforme y sin rizado para las técnicas no-convencionales de control y un desempeño superior respecto al PID clásico, principalmente en cuanto respecta al error en estado estacionario ante la acción de perturbaciones. En términos de implementación, la técnica de control óptimo conmutado resulta ser más simple que el SMC. Inicialmente se modela la planta mediante una herramienta computacional de MATLAB, posteriormente se diseña un circuito actuador para operar con los dos dispositivos programables, luego se diseña el circuito de carga que se ubica en el generador, teniendo la planta identificada con capacidad de aplicarle carga, se diseña cada uno de los controles a comparar y se visualizan los resultados mediante una interfaz grafica mediante Labview. Un análisis complementarios de robustez y optimalidad forman parte del trabajo futuro.

¹ Trabajo de grado

² Facultad de ingenierías Físico-mecánicas. Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones. Director: Ricardo Alzate Castaño, PhD. Codirector: Carlos Augusto Fajardo Ariza, MSc.

SUMMARY

Title: COMPARATIVE ANALYSIS OF PERFORMANCE ON SWITCHING CONTROL TECHNIQUES IMPLEMENTED ON PROGRAMMABLE DEVICES³.

Authors: FABIO ANDRÉS MORENO CASTELLANOS, DIEGO ARMANDO CENTENO ARIZA⁴

Keywords:

Arduino, FPGA, DC motor, Optimal control, SMC.

Description:

This project is aimed at implementing switching control techniques on programmable devices as a way to compare the performance of a speed DC-motor system under the action of diferents load disturbances. Two main techniques, namely: sliding mode control (SMC) and optimal switching control are analyzed and compared with a classical PID algorithm. Results show that a smooth shaped profile of velocity is generated for the switching techniques and hence an increased performance compared with very noisy PID responses. Also, computational effort is reduced when implementing the optimal control technique on FPGA and Arduino microcontrollers, showing good results on keeping motor speed close to reference values under disturbances in both cases. An in-depth analysis for robustness and optimality is part of the current ongoing work. Initially the plant is modeled using a software tool, MATLAB, more specifically a toolbox named IDENT ; subsequently the actuator circuit is designed to operate with the two programmable devices, then the load circuit located on the generator was designed, after identify the plant with capacity of applying load, each controls is designed to compare and displays the results through a graphical interface using Labview. A complementary analysis of robustness and optimality are part of future work.

³ Degree work.

⁴ Physico-mechanical Engineering Faculty. School of Electrical Engineering. Supervisor: Ricardo Alzate Castaño, PhD. Co-supervisor: Carlos Augusto Fajardo Ariza, MSc.

CAPITULO 1

1.1 ANTECEDENTES Y JUSTIFICACIÓN

El control conmutado, también denominado control híbrido, estudia sistemas en los cuales existe interacción entre dinámicas continuas y discretas [1]. El estudio de estos modelos híbridos ha ganado reciente interés en la comunidad científica debido a que aplicaciones cotidianas como las redes y los sistemas distribuidos pueden definirse de manera natural en este tipo de contextos. Un ejemplo simple para estrategias de control híbrido y/o conmutado, es el denominado control por modos deslizantes (Sliding Mode Control – SMC [2]) en donde fundamentalmente la evolución de un sistema continuo está determinada por condiciones de frontera asignadas a partir de reglas discretas. En general, estas condiciones de frontera se cumplen en un subconjunto del espacio de estados denominado superficie de deslizamiento. Si el sistema decide forzarse a confinar su evolución dinámica a dicha superficie de deslizamiento, entonces esta última puede configurarse para constituir una trayectoria predefinida aplicable a problemas de seguimiento.

La ventaja del SMC radica en su capacidad para tolerar variaciones paramétricas del modelo; en otras palabras es una técnica robusta. Aplicaciones para esta estrategia de control incluyen: el control de accionamientos eléctricos operados por convertidores de potencia conmutados [3]; el rastreo de trayectorias de Robots móviles y vehículos autónomos [4]; el manejo de servo-mecanismos [5, 6] y vehículos híbridos [7]; entre otras. En Colombia, ejemplos de investigaciones afines con el control híbrido o conmutado incluyen: [8] donde se aplica una estrategia de control (híbrido) no-lineal a un péndulo en el laboratorio de sistemas no-lineales y percepción en la Universidad del Quindío y [9] donde se combinan estrategias de diseño PID multivariable y de conmutación a un robot industrial tipo SCARA. En la Universidad Industrial de Santander, se destacan los trabajos de grado: [10] donde se aplica un control en modo deslizante a un modelo de turbina

hidráulica para comparación con una estrategia H^∞ ; [11] empleando control en modo deslizante para un sistema de segundo de orden con retardo; y [12] en el cual se estabiliza una cola de helicóptero a través de técnicas convencionales de control.

De otro lado, la evolución de las tecnologías asociadas al diseño de sistemas electrónicos (herramientas EDA – Electronic Design Automation) ha contribuido de manera notoria al desarrollo de técnicas cada vez más efectivas para el prototipado de circuitos con aplicación específica, permitiendo desarrollar desde simples filtros RLC hasta complejos Sistemas Embebidos. Lo anterior no solo es producto de un desarrollo apreciable en herramientas de software, sino también a nivel de las plataformas de hardware disponibles para implementación, principalmente de sistemas digitales con cada vez mayores y mejores prestaciones de velocidad y de memoria, entre los cuales se pueden enumerar: a) dispositivos de arquitectura configurada incluyendo microcontroladores y DSP (Digital Signal Processor); b) dispositivos de arquitectura configurable como CPLD (Complex Programmable Logic Device), FPGA (Field Programmable Gate Array) o FPAA (Field Programmable Analog Array); finalmente y de manera reciente c) dispositivos con arquitecturas tanto configuradas como configurables del tipo SoC (System on Chip). En [13, 14, 15] se reportan algunos desarrollos de implementación de algoritmos en este tipo de tecnologías.

La creciente complejidad de los problemas que actualmente debe resolver la ingeniería, implica una constante evolución de las técnicas requeridas para implementar soluciones. Dichas soluciones adquieren la forma de sistemas embebidos en el contexto de realizaciones digitales a gran escala. Al respecto, se han desarrollado estudios que buscan obtener una manera eficaz de diseñar e implementar sistemas computacionales (básicamente de movimiento y transformación de datos digitales) garantizando altos niveles de desempeño y de aprovechamiento de recursos durante su ejecución (implementación hardware)

definitiva. Como ejemplos de resultados reportados en la literatura se tienen metodologías de diseño del tipo top-down [16, 17] y basadas en modelos de computación [18, 19]. Recientemente, en la Universidad Industrial de Santander se desarrolló a nivel de tesis de Maestría una metodología para el diseño de sistemas embebidos [20].

1.2 PLANTEAMIENTO Y DEFINICIÓN DEL PROBLEMA:

El control moderno expandió los horizontes del control clásico hacia el espacio de estados, permitiendo manipular dinámicas multivariadas y de tipo no-lineal. Posteriores desarrollos introdujeron robustez a la incertidumbre, adaptabilidad y optimalidad ante ciertas funciones objetivo. En contraste con su desempeño, los algoritmos empleados en algunas de estas técnicas representan considerable costo computacional, sobre todo si se requieren acciones de control con variación uniforme de amplitud. Como alternativa aparecen las técnicas de control por conmutación, en las cuales se construye una acción de control pulsante a partir de niveles lógicos (digitales), con valores programados en instantes de tiempo determinados por una ley de conmutación.

En cualquiera de los casos, la posibilidad de desarrollar realizaciones circuitales para algoritmos avanzados de control (entre ellos los de control conmutado) está condicionada al aprovechamiento eficiente de los recursos tecnológicos actuales. Lo anterior permite generar las siguientes inquietudes: ¿Cómo aplicar eficientemente una metodología de diseño de sistemas digitales al diseño de un sistema de control? ¿Qué ventajas y desventajas presentan las alternativas tecnológicas disponibles al momento de aplicar una metodología de diseño? ¿Qué técnicas de control avanzado funcionan mejor en una determinada plataforma de implementación?

El presente proyecto de grado busca realizar aportes direccionados a la resolución de este tipo de interrogantes, constituyendo una base para posteriores desarrollos afines al interior de grupos de investigación en la E³T, como una manera de generar medios de apropiación tecnológica para países en vías de desarrollo como el nuestro.

1.3 OBJETIVOS:

1.3.1 OBJETIVO GENERAL

- Realizar un estudio comparativo del desempeño de técnicas de control conmutado implementadas en dispositivos programables.

1.3.2 OBJETIVOS ESPECÍFICOS

- Utilizar una metodología de diseño de sistemas embebidos para implementar algoritmos de control sobre un dispositivo de arquitectura configurable.
- Aplicar técnicas de control conmutado implementadas sobre un dispositivo de arquitectura configurable para controlar la velocidad de un motor de corriente continua.
- Aplicar técnicas de control conmutado implementadas sobre un dispositivo de arquitectura configurada para controlar la velocidad de un motor de corriente continua.
- Comparar el desempeño de las técnicas a partir de las diferentes alternativas de implementación.

CAPITULO 2

A continuación se ilustran la configuración y el modelado matemático para el sistema de trabajo (planta), constituido para realizar la verificación experimental de las técnicas de control digital de velocidad desarrolladas en el presente trabajo de grado. Inicialmente se describirán las etapas constitutivas generales, detalladas con observaciones y parámetros de diseño. Posteriormente se describirá el procedimiento experimental para determinar la función de transferencia (relación entrada-salida) del sistema.

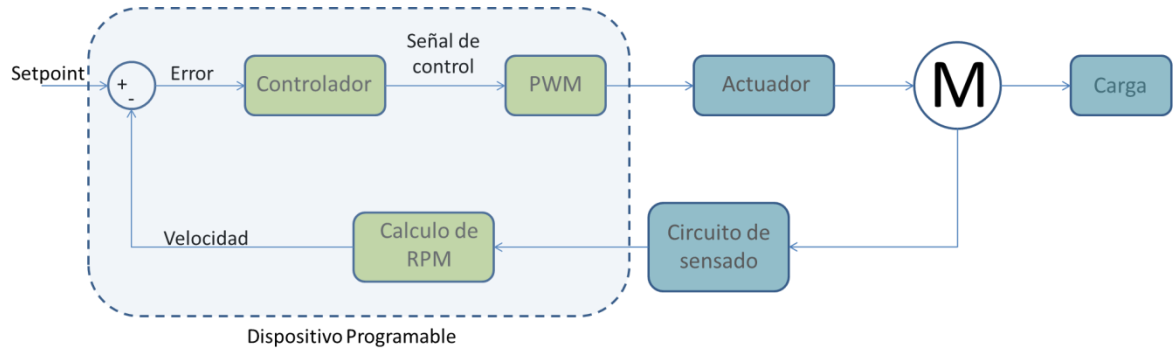
2.1 DESCRIPCIÓN GENERAL DEL SISTEMA

En la Figura 1 se muestra un diagrama de bloques general que permite ilustrar las etapas que constituyen los principales componentes para el sistema implementado de control de velocidad de un motor DC y su respectiva interacción (señales). La función para cada etapa es tal y como se describe a continuación:

- **Actuador:** dispositivo encargado de realizar la conversión entre la señal de control (salida de PWM) y los niveles de tensión y corriente entregados al motor.
- **Motor (M):** dispositivo encargado de realizar la conversión de una señal eléctrica a movimiento mecánico y que por tanto constituye el elemento final de control E.F.C. en el esquema.
- **Carga:** torque variable aplicado en el rotor del motor como perturbación para verificar el efecto de la acción del control de velocidad.
- **Sensor:** dispositivo encargado de realizar la conversión entre la variación angular del rotor (velocidad) y una señal eléctrica proporcional utilizada como medida.

- **Dispositivo programable:** sistema de desarrollo para microcontrolador o FPGA utilizado para calcular la señal de error y con base en ello programar la correspondiente ley de control digital de corrección.

Figura 1. Diagrama de bloques para el control de velocidad de un motor de DC



Fuente: Autores

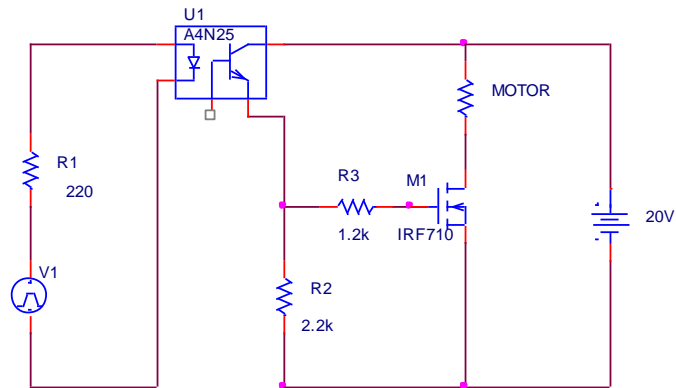
2.1.1 Actuador

El circuito actuador (ver Figura 2) cuenta con un optoacoplador que sirve como protección de aislamiento entre las tierras digital (dispositivo programable) y analógica (actuador) debido a su interconexión óptica. También cuenta con un transistor MOSFET configurado en modo de conmutación para modificar, en proporción a una señal PWM de entrada, la alimentación en los terminales del motor DC alojado como carga. La resistencia de 220Ω a la entrada del optoacoplador garantiza una corriente máxima de 60 mA, ya que la señal de PWM posee una amplitud de 5 V. La resistencia de $2.2k\Omega$ se configura en modo PULL-DOWN. Finalmente la resistencia de $1.2k\Omega$ representa un elemento de protección y polarización para la compuerta del MOSFET.

2.1.2 Motor

El motor DC utilizado en el sistema es un NISCA de referencia NF5475E con alimentación de 24 VDC (ver Figura 3). Otras características técnicas se presentan en la Tabla 1.

Figura 2. Circuito actuador del sistema



Fuente: Autores

Tabla 1. Características de motor modelo NF5475E

Valores a máxima eficiencia					Sin carga	
Torque [mN.m]	Ω [rpm]	I [A]	P [W]	Eficiencia	ω [rpm]	I [A]
74.83	4162	1.256	32.62	68.3	4884	0.218

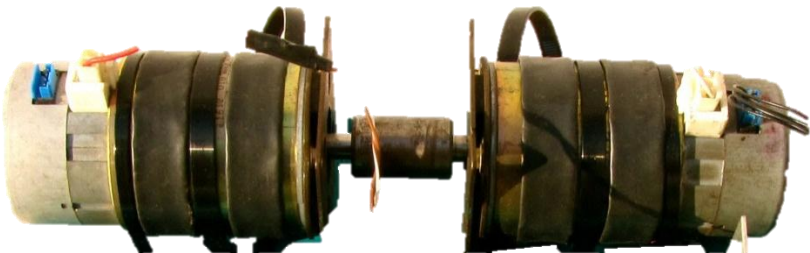
Fuente: Nisca Corporation

2.1.3 Cargas

Para agregar diferentes cargas a manera de torque en el eje del motor, se utilizó otro motor de las mismas características acoplado como generador en serie (ver Figura 3), el cual produce cambios de torque tras modificar la resistencia en su circuito de armadura (terminales de alimentación del generador). Lo anterior se

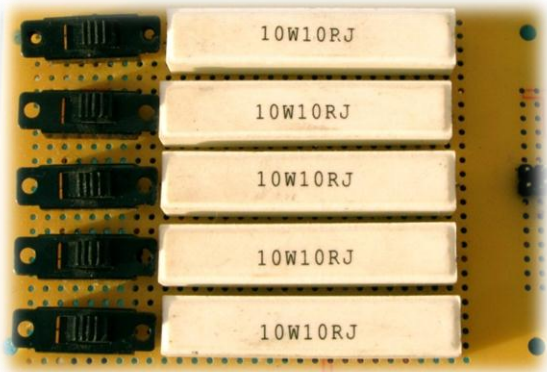
implementó adicionando al generador un conjunto de 5 resistencias de $10[\Omega]/10[W]$, configuradas en serie, seleccionadas a partir de un arreglo de interruptores como se muestra en la Figura 4.

Figura 3. Acoplamiento de motor y generador para producir torque variable



Fuente: Autores

Figura 4. Banco de resistencias de carga

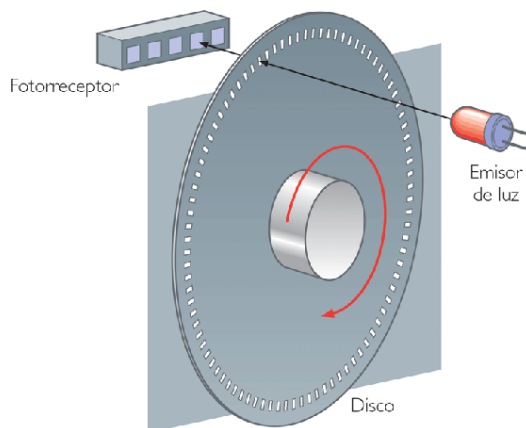


Fuente: Autores

2.1.4 Sensor

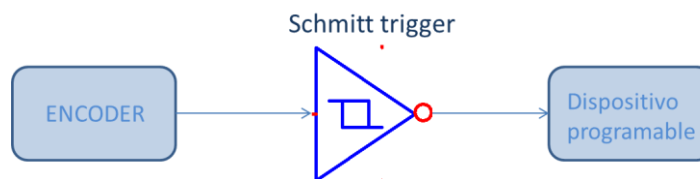
Como circuito de sensado se utilizó un encoder óptico de tipo incremental, acoplado al eje de rotación del motor en cascada con un sistema para conteo de pulsos digitales (ver Figuras 5 y 6). Cada revolución del motor posee 200 pulsos, con lo cual se obtiene una resolución de 1.8° por pulso. De esta manera una aproximación sencilla para la velocidad angular corresponde a la razón entre la cantidad de pulsos y el tiempo correspondiente en que fueron obtenidos. Esta lectura se realiza al interior del dispositivo programable.

Figura 5. Esquema funcional de un encoder óptico



Fuente: Kalipedia, *Encoder óptico*

Figura 6. Esquema del sistema de sensado implementado



Fuente: Autores

2.1.5 Dispositivo programable

Como sistemas digitales para llevar a cabo los procedimientos de cálculo de error y de implementación de las correspondientes leyes de control, se emplearon los siguientes dispositivos comerciales por razones de disponibilidad:

- Arquitectura configurada: sistema de desarrollo para microcontrolador Arduino de referencia MEGA 2560, con características técnicas resumidas en la Tabla 2 e ilustración dada en la Figura 7.

Tabla 2. Características de tarjeta Arduino MEGA 2560

Microcontrolador ATmega2560
54 pines digitales entrada-salida (14 PWM)
16 entradas análogas
4 Puertos UART
Cristal de 16MHz
Puerto USB para programación y alimentación
Conector para alimentación externa de 12 VDC

Fuente: www.arduino.cc

Figura 7. Tarjeta Arduino MEGA 2560



Fuente: www.arduino.cc

- Arquitectura configurable: sistema de desarrollo para FPGA Xilinx Spartan 6 de referencia NEXYS-3, con características técnicas resumidas en la Tabla 3 e ilustración dada en la Figura 8.

Tabla 3. Características de la tarjeta Nexys 3 – Spartan 6

Xilinx Spartan-6 LX16 FPGA cuenta con un empaquetado BGA de 324-pines
16Mbyte RAM (x16)
16Mbytes SPI PCM quad mode
16Mbytes PCM paralelo, memoria no volátil
10/100 Ethernet PHY
Puerto USB2 en la tarjeta para programar
Puerto USB-UART y USB-HID (para mouse/teclado)
Puerto de 8-bit VGA
100MHz oscilador CMOS
72 I/O's que sirven como conectores de expansión
GPIO incluye 8 LEDs, 6 botones,8 switches y un visualizador de siete segmentos de 4 dígitos

Fuente: www.digilent.com

Figura 8. Tarjeta Nexys 3 – Spartan 6

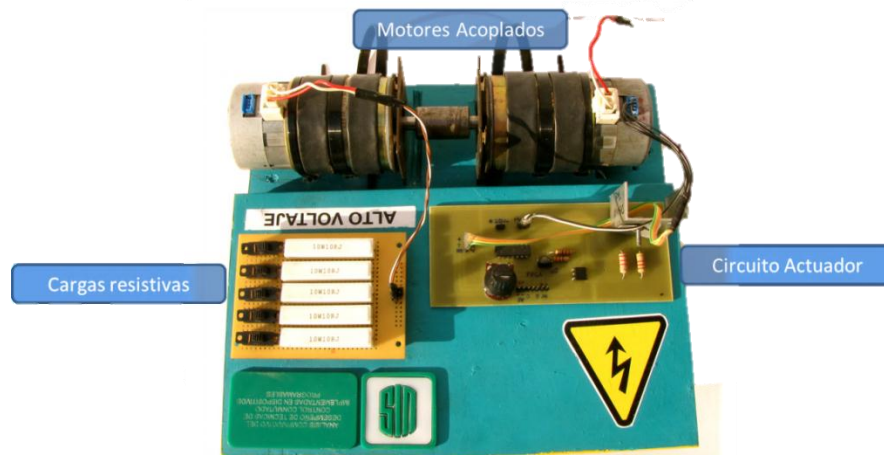


Fuente: www.digilent.com

2.1.6 Sistema definitivo

Una vez constituidas las etapas de actuación y de sensado, es posible interconectar dichas etapas en el prototipo de pruebas mostrado en la Figura 9, el cual consta de una base de madera para alojar los motores y las tarjetas impresas del circuito actuador y del banco de resistencias de carga.

Figura 9. Prototipo para pruebas

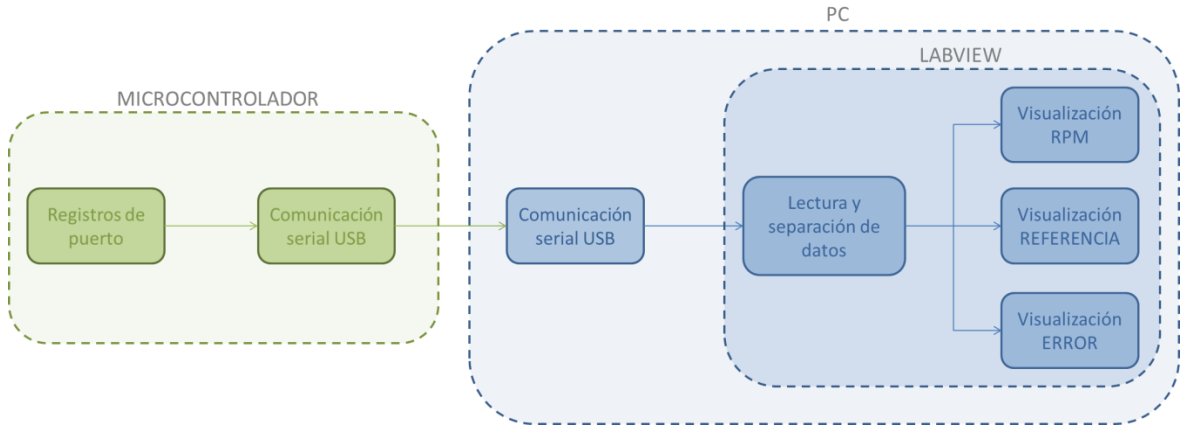


Fuente: Autores

2.2. INTERFAZ DE MONITOREO

Adicional a las descripciones generales del sistema presentadas en la sección 2.1, fue necesario obtener información en línea de las principales señales involucradas en el funcionamiento del sistema para poder realizar un monitoreo apropiado de la evolución dinámica del mismo. Para ello se desarrolló una interfaz en LABVIEW® orientada a visualizar la evolución en el tiempo de las señales de: referencia (set point), salida (velocidad del motor en RPM) y control (porcentaje de ciclo útil para señal de PWM enviada al actuador).

Figura 10. Diagrama de bloques para el sistema de adquisición/visualización de datos implementado.



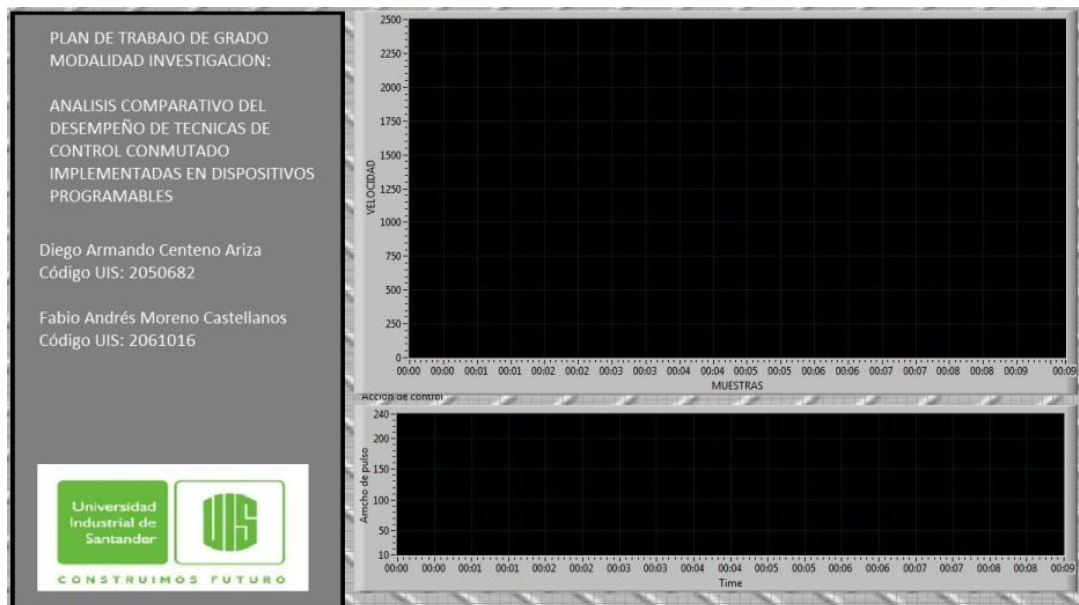
Fuente: Autores

La captura de dichas señales fue realizada a partir de un sistema de adquisición de datos basado en el microcontrolador Arduino MEGA 2560, el cual realiza el empaquetamiento de la información digital adquirida en sus puertos mediante protocolo serial universal (USB). En la Figura 10 se presenta un diagrama de bloques para el sistema de monitoreo (hardware + software) implementado. En el primer bloque (microcontrolador) se realiza la captura y transmisión de datos a través de los periféricos respectivos; es decir registros e interfaz de comunicación USB. Posteriormente, estos datos son enviados a un PC en donde inicialmente son desempaquetados de la correspondiente trama serial, para ser visualizados en LABVIEW® en sus versiones de señal de error, de lectura de velocidad en RPM y de referencia. La Figura 11 ilustra el aspecto de la pantalla principal para la interfaz LABVIEW® desarrollada.

2.3. PROCEDIMIENTO PARA IDENTIFICACIÓN EXPERIMENTAL DEL MODELO DEL SISTEMA

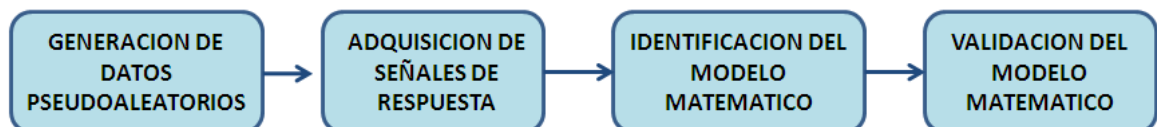
Tomando como base el procedimiento sugerido en [21], la Figura 12 muestra el diagrama de bloques que resume las etapas requeridas para obtener el modelo matemático requerido. A continuación se realizará una descripción de cada una de dichas etapas.

Figura 11. Panel frontal de la interfaz en LABVIEW®



Fuente: Autores

Figura 12: Diagrama de bloques del procedimiento experimental



Fuente: [21]

- **Generación de datos Pseudoaleatorios:** esta etapa se encarga de constituir una señal de excitación al sistema con una cantidad suficiente de componentes frecuenciales (señal de excitación persistente), en modo tal que sea posible activar en el mismo la mayor cantidad de dinámicas reflejadas en su respuesta [21]. En términos prácticos esta señal corresponde con una secuencia binaria pseudoaleatoria aplicada a manera de cambios de ciclo útil de PWM al circuito actuador.
- **Adquisición de señales de respuesta:** esta etapa realiza la captura de la respuesta del sistema ante el estímulo de la señal de excitación persistente. Para facilitar el análisis de los datos existe una sincronía en tiempos entre cada muestra adquirida como respuesta y la correspondiente a la excitación de entrada en dicho instante de tiempo [21]. En términos prácticos la señal obtenida en el sensor (encoder) se envía a un PC mediante la interfaz descrita en el numeral 2.2, que permite almacenar la información (tanto de entrada como de salida) a través de rutinas de código realizadas en LABVIEW®.
- **Identificación del modelo matemático:** empleando técnicas de análisis estadístico es posible inferir la relación matemática entre los vectores de entrada y salida obtenidos para el sistema en los dos procedimientos descritos previamente [21]. Este proceso se desarrolló fuera de línea haciendo uso de la herramienta de identificación de MATLAB® (IDENT – System identification toolbox [22]) empleando principalmente el ajuste de correlación entre modelos paramétricos lineales.
- **Validación del modelo matemático:** una vez obtenida la mejor estimación estadística del modelo matemático para relacionar los conjuntos de datos de interés, se efectuó validación adicional a partir de comparación entre los resultados analíticos y experimentales para un conjunto de datos de prueba constituidos por señales impulso, escalón y rampa [21]. Las señales

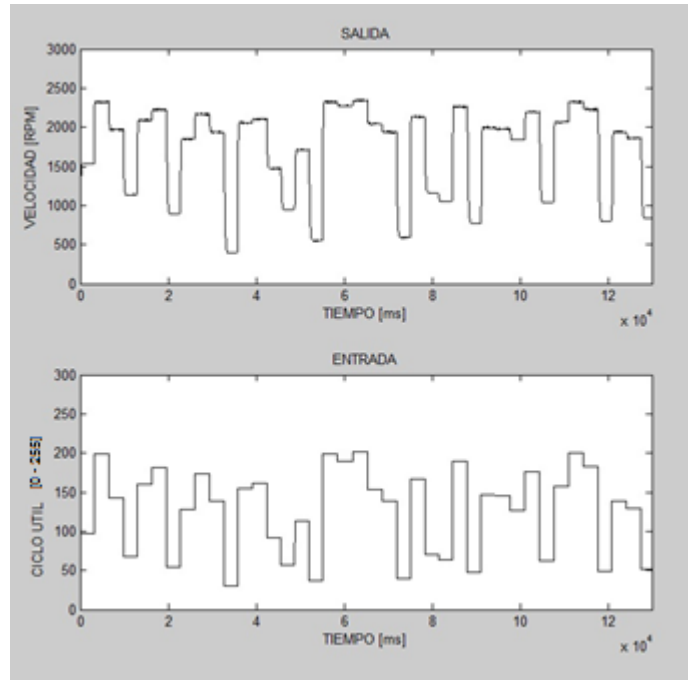
generadas de manera experimental (salidas) fueron comparadas con los resultados obtenidos a partir de evaluar los modelos con el mismo conjunto de datos de excitación. El porcentaje de error de las predicciones teóricas respecto a los valores experimentales permitió corroborar la validez de los modelos matemáticos identificados.

Aplicando los pasos sugeridos en la Figura 12, la primera etapa que debe llevarse a cabo consiste en la generación de los datos de entrada y salida para estímulos del tipo binario pseudoaleatorio. La Figura 13 muestra las señales generadas experimentalmente para excitar las dinámicas del motor en el prototipo con sus correspondientes señales de respuesta (velocidad de su eje en RPM). Note el ajuste de calibración realizado en el rango de la señal de entrada, el cual posee los límites [30, 210] para evitar fenómenos de banda muerta y de saturación en el motor, tal y como se ilustra en la Figura 14. Note también que la señal de entrada se encuentra expresada como un número binario de 8 bits representando un porcentaje del ciclo útil de la señal de PWM aplicada al actuador, entre 30 (11.8%) y 210 (82.4%).

2.3.1 Modelo para función de transferencia del sistema

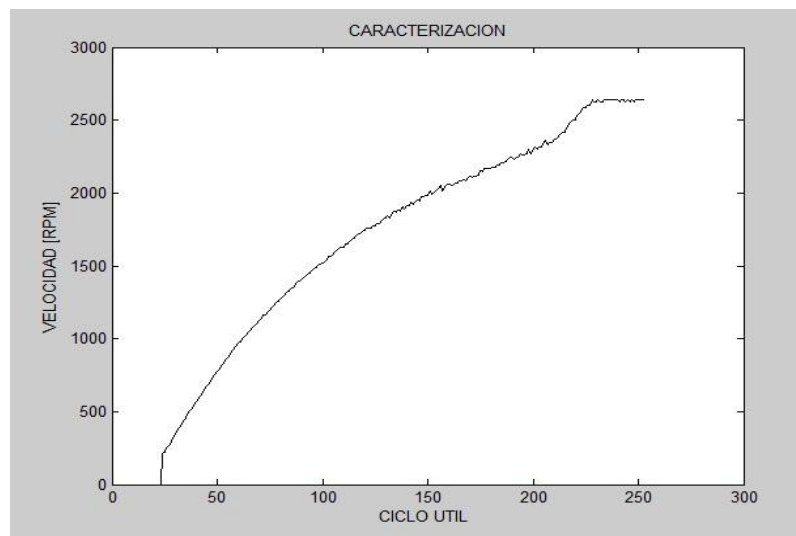
En adelante se ilustrará el procedimiento de identificación y validación del modelo matemático de la relación de transferencia entre la velocidad angular en el eje del motor (medida en RPM) y el ciclo útil de entrada al circuito actuador.

Figura 13: Señales de respuesta (superior) y excitación (inferior) ante estímulos binarios pseudoaleatorios en prototipo



Fuente: Autores

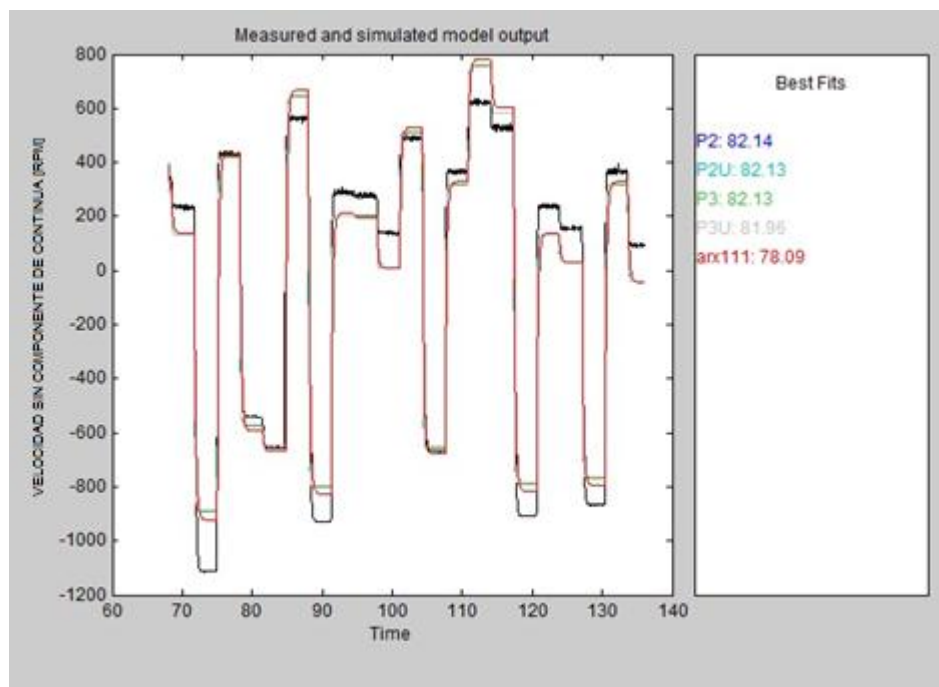
Figura 14: Identificación de regiones de saturación y banda muerta en el sistema



Fuente: Autores

De esta manera, con los datos correspondientes de la Figura 13 se accede a la herramienta de identificación de sistemas de MATLAB® (IDENT – System Identification Toolbox [22]) en donde el primer paso consiste en seleccionar la mitad inicial de los datos como conjunto de estimación y la segunda mitad como conjunto de validación. Con estos subconjuntos de datos se efectuó identificación del modelo matemático que permite relacionar los datos de entrada y salida, para el caso de diferentes estructuras lineales paramétricas entre ellas: función de transferencia con/sin retardos y ARX. Para ilustración de los pasos requeridos en la identificación de sistemas empleando la herramienta IDENT de MATLAB® se recomienda al lector consultar [22]. En la misma medida, para acceder a información relacionada con la justificación estadística de las estructuras de modelos disponibles en la herramienta IDENT se recomienda consultar [23].

Figura 15: Correlación de modelos para identificación de la relación de transferencia entre la velocidad angular en el eje del motor y el ciclo útil de entrada al circuito actuador



Fuente: Autores

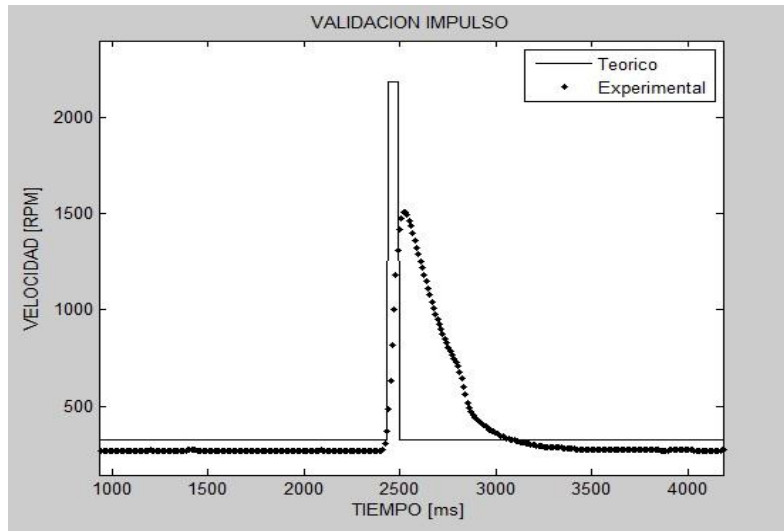
La Figura 15 muestra los resultados de correlación obtenidos por la herramienta de identificación para el caso de los modelos ARX y función de transferencia de dos y tres polos. La elección de estos modelos de prueba se basó en el alto índice de correlación obtenido (best fit) en comparación con otras opciones. Como se observa de la Figura 15, la correlación más alta entre los conjuntos de entrenamiento y validación (82.14 %) se obtuvo para el modelo tipo función de transferencia de dos polos. Por tanto, el modelo obtenido como mejor estimado de la relación entrada-salida del sistema corresponde con:

$$H = \frac{\Omega(s)}{U(s)} = \frac{50000}{s^2 + 1010s + 4575} \quad (2.1)$$

Con esta función de transferencia para el modelo estimado, se procede a realizar una nueva validación con estímulos independientes correspondientes a señales del tipo: impulso, escalón y rampa. De esta manera las Figuras 16-18 presentan las salidas simuladas (evaluación numérica de la ecuación 2.1) respecto a las obtenidas mediante medidas experimentales, para los tres casos de señales de validación considerados. A partir de una valoración cualitativa, es evidente la pertinencia en el comportamiento del modelo estimado respecto a los resultados de las medidas reales del sistema ante cambios de entrada de tipo escalón (Figura 17). Asimismo, el comportamiento comparativo es aceptable ante un cambio de entrada de tipo impulsivo (Figura 16) tomando en cuenta su inviabilidad práctica. En el caso de cambios de entrada con pendiente constante se verifican alinealidades del sistema, no capturadas por el modelo (Figura 18). Este resultado evidencia discrepancias de primer orden entre el sistema real y su modelo, implicando la necesidad de evitar análisis comparativos para entradas de tipo rampa. Sin embargo, tal y como se observará más adelante, el modelo obtenido es un punto de partida sobre el cual se realizarán el diseño y ajuste de técnicas no lineales (conmutadas) de control, que deberán mitigar los efectos de tales aproximaciones en el desempeño del sistema. Con ello, se valida a la

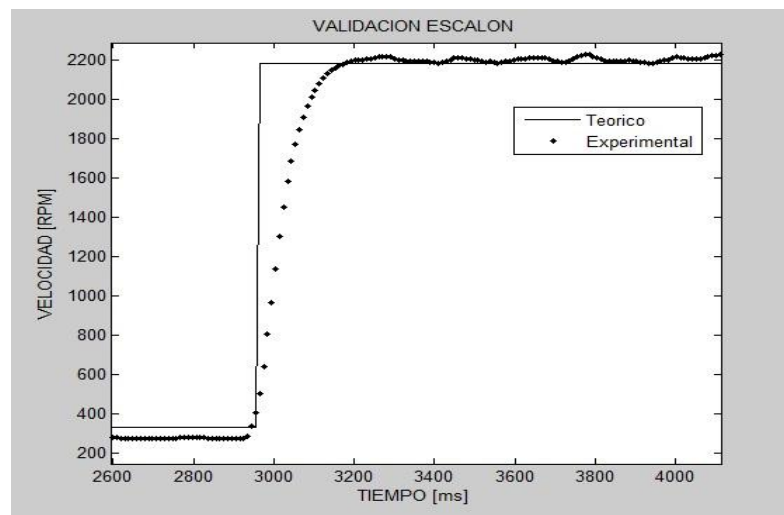
representación matemática del modelo presentado por la ecuación 2.1 como relación para los conjuntos de datos de entrada y salida considerados.

Figura 16: Superposición entre datos de salida (simulación + medidas experimentales) para validación ante un estímulo impulsivo.



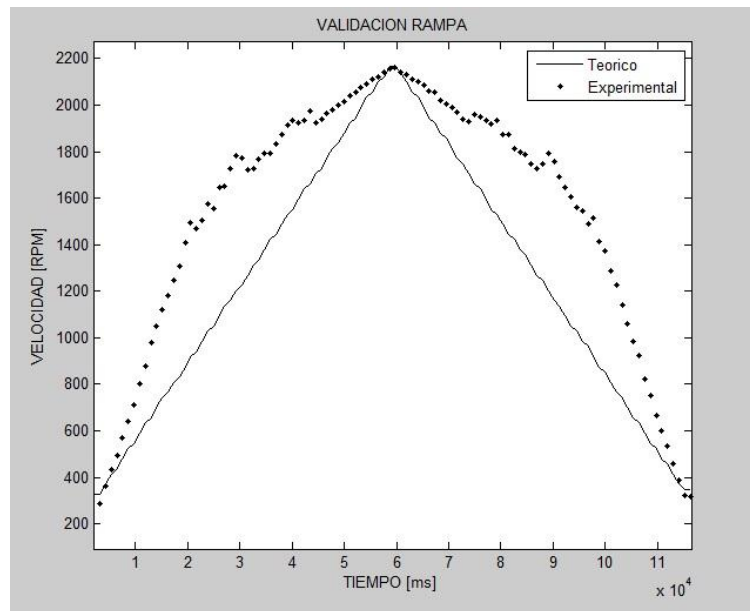
Fuente: Autores

Figura 17: Superposición entre datos de salida (simulación + medidas experimentales) para validación ante un estímulo escalón



Fuente: Autores

Figura 18: Superposición entre datos de salida (simulación + medidas experimentales) para validación ante un estímulo tipo rampa.



Fuente: Autores

La Tabla 3 contiene información de las medidas de error cuadrático medio obtenidas para la validación.

Tabla 4: Errores de validación para las diferentes señales de prueba utilizadas.

Señal de entrada	E.C.M
Impulso	30.5%
Escalón	6.65%
Rampa	35.35%

Fuente: Autores

CAPITULO 3

Una vez definido, construido e identificado el sistema de trabajo (planta) para el sistema de control de velocidad en un motor DC, es momento de realizar el diseño y la implementación de las respectivas acciones de control que permitirán manipular a conveniencia el desempeño del sistema ante la acción de perturbaciones. En el presente capítulo se realizará el análisis de los resultados de implementación de controladores, empleando como plataforma de ejecución computacional una arquitectura configurada: microcontrolador Arduino Mega 2650.

3.1 CONTROLADOR PID DIGITAL

Un controlador proporcional-integral-derivativo (PID) es sin duda la técnica más difundida y empleada en la industria colombiana para el control de procesos. En su versión simple de tiempo continuo, la acción de control PID toma la forma dada por la ecuación (3.1), siendo K_p , K_i y K_d respectivamente las constantes de ganancia proporcional, integral y derivativa, al igual que $e(t)$ representa la señal de error [24].

$$u(t) = K_p e(t) + K_i \int_0^t e(t) dt + K_d \frac{de(t)}{dt} \quad (3.1)$$

Para efectos de la implementación digital del controlador, se considera la función de transferencia equivalente discreta sugerida en [25] (función de transferencia pulso para el controlador PID), dada por la expresión (3.2) donde T equivale al periodo de discretización.

$$C(z) = \frac{U(z)}{E(z)} = \frac{\left(K_p + K_i \frac{T}{2} + \frac{K_d}{T}\right) z^2 + \left(-K_p + K_i \frac{T}{2} - \frac{2K_d}{T}\right) z + \frac{K_d}{T}}{z^2 - z} \quad (3.2)$$

Posterior a efectuar manipulación matemática de la ecuación (3.2) es posible obtener la expresión para la ecuación en diferencias que permite relacionar la

secuencia de control como una función de la secuencia de error, tal y como se observa en la ecuación (3.3). Por tanto, una vez asignados los valores para las constantes del controlador (incluido el intervalo de discretización), la evaluación computacional de dicha expresión permitirá calcular de manera simple la acción de control para cada muestra de error, obtenida a partir de medidas periódicas de las variables del sistema (velocidad de giro del motor).

$$u[k] = u[k - 1] + \left(K_p + K_i \frac{T}{2} + \frac{K_d}{T} \right) e[k] + \left(-K_p + K_i \frac{T}{2} - \frac{2 K_d}{T} \right) e[k - 1] + \frac{K_d}{T} e[k - 2] \quad (3.3)$$

3.1.1 Cálculo de los parámetros del controlador

Empleando MATLAB® y métodos convencionales de diseño, se empleó el modelo de sistema dado por la ecuación (2.1) para determinar los parámetros de un controlador PID que permitieran mejorar la respuesta escalón unitario del sistema en lazo cerrado. Los valores de parámetro obtenidos se presentan en la Tabla 3.1.

Tabla 5. Parámetros de controlador PID calculados analíticamente

VALORES PARAMETROS PID	
Kp	0.7
Ki	6
Kd	0.0086
T	0.010

Fuente: Autores

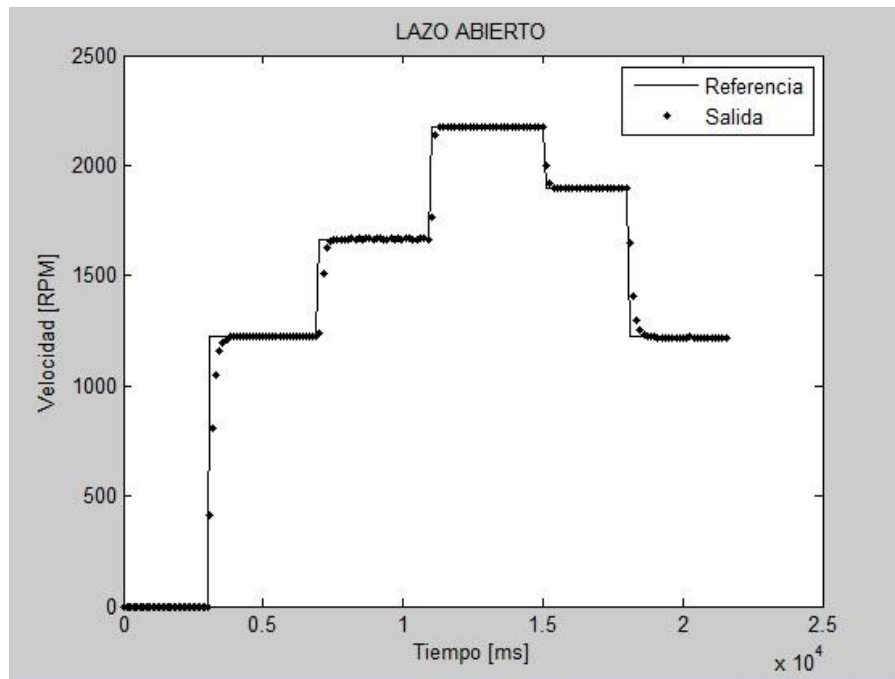
En cuanto al periodo de muestreo, se determinó empíricamente (empleando la interfaz de monitoreo desarrollada y descrita en el Capítulo 2) que su valor mínimo

es de aproximadamente 10 [ms], razón por la cual se definió a este como el intervalo de discretización T .

3.1.2 Comportamiento dinámico del sistema sin control

A partir de los resultados visualizados en la Figura 14 (Capítulo 2), se obtuvo un rango de valores de velocidad dentro del cual es posible actuar (gobernar) la dinámica del sistema para satisfacer los requerimientos de velocidad de giro en el eje del motor. La Figura 19 ilustra los resultados experimentales obtenidos para el sistema en lazo abierto a carga nominal. Nótese como la salida del sistema alcanza rápidamente el valor de la referencia, lo cual evidencia la apropiada calibración de los dispositivos (actuador y sensor).

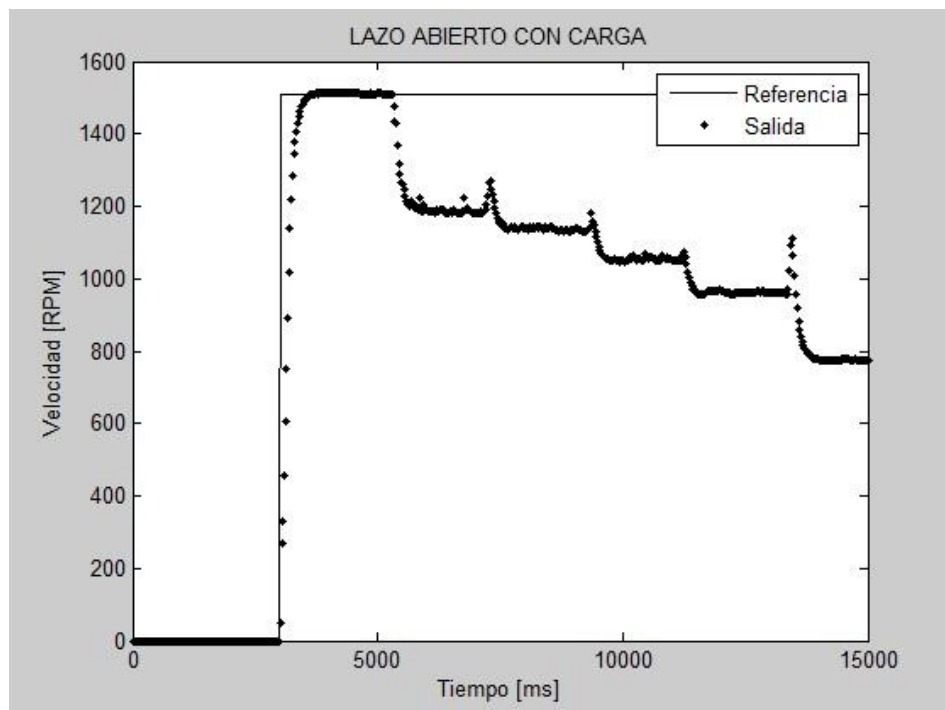
Figura 19. Comportamiento del sistema en lazo abierto con carga nominal



Fuente: Autores

De otro lado, la Figura 20 muestra cómo es afectado el desempeño del sistema al incluir los efectos de adición de carga. En la Figura, se mantiene constante el valor de referencia en 1510 RPM. Posteriormente se agregan 5 valores de carga (ver Sección 2.1.3) incrementados de uno en uno cada 2 [s]. A partir de los resultados es indiscutible el efecto nocivo de las perturbaciones sobre la velocidad.

Figura 20. Efecto de perturbaciones de carga para el sistema en lazo abierto

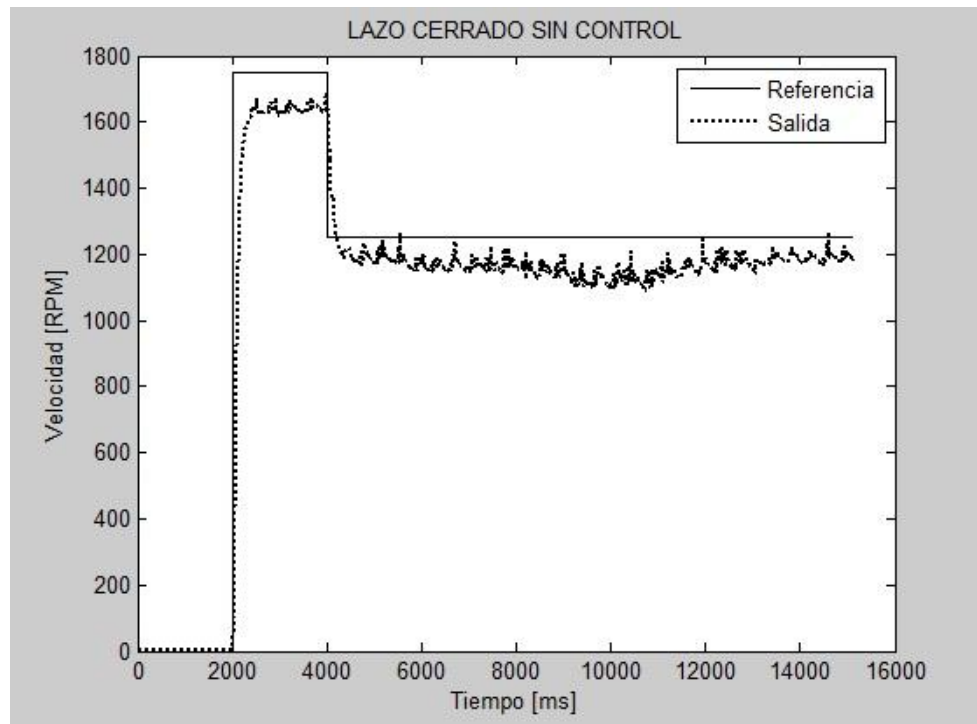


Fuente: Autores

Finalmente, la Figura 21 muestra resultados para el sistema en lazo cerrado sin control. Como era de esperarse, el efecto de cerrar el lazo es la atenuación de la

acción de las perturbaciones. Sin embargo, el desempeño del sistema está lejos de ser aceptable.

Figura 21. Efecto de perturbaciones de carga para el sistema en lazo cerrado



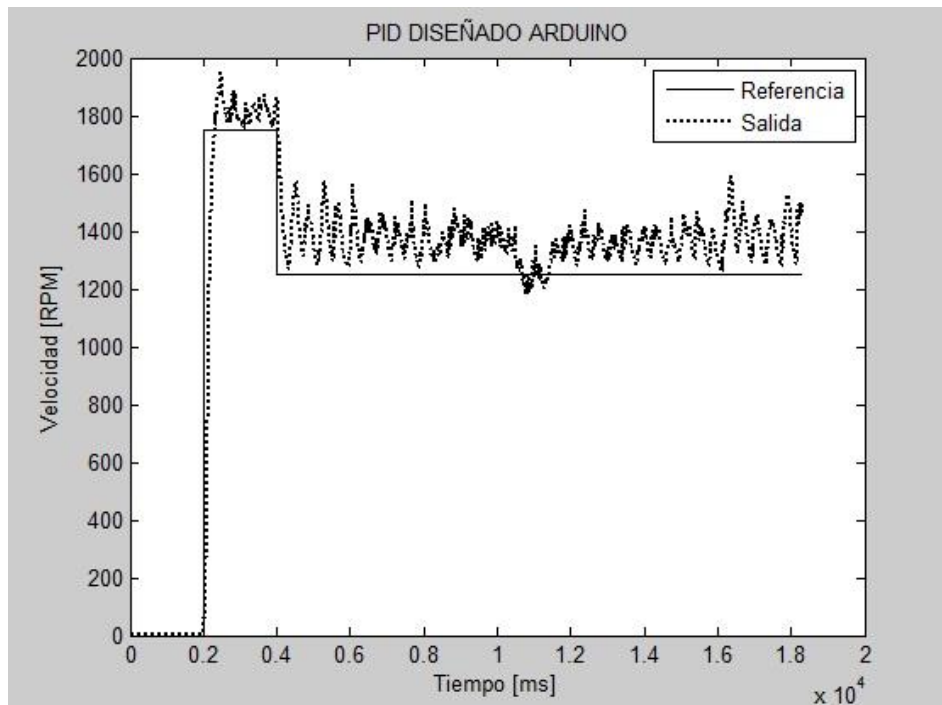
Fuente: Autores

3.1.3 Comportamiento del sistema controlado

Tras incluir la acción de control definida en la ecuación (3.3) junto con los valores de parámetro del controlador incluidos en la Tabla 3.1 y el periodo de muestreo seleccionado, la Figura 22 ilustra el resultado del comportamiento del sistema controlado mediante un PID ante la acción de perturbaciones. A partir de ello se observa una mayor capacidad del sistema para mantener (en promedio) el valor

de referencia, siendo sin embargo afectada la respuesta por una significativa amplitud de oscilación consecuencia de los efectos derivativos del controlador.

Figura 22. Comportamiento del sistema controlado por acción PID

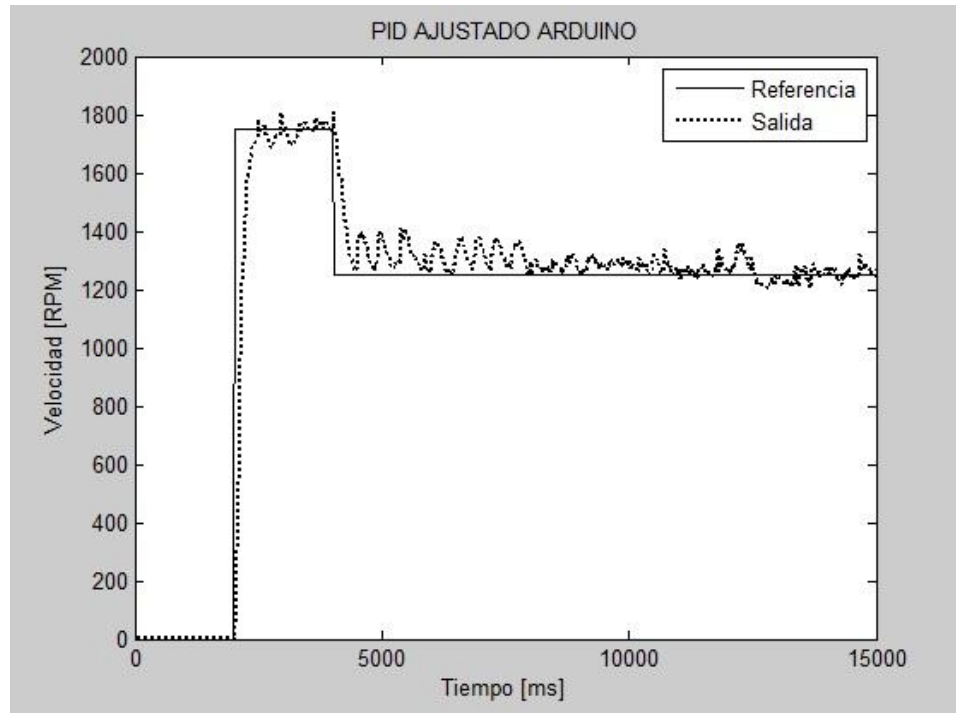


Fuente: Autores

Un ajuste fino (sintonía) en los valores de parámetro del controlador, permitió obtener una respuesta con transitorios suavizados tal y como se muestra en la Figura 23, constituyendo la mejor respuesta experimental para este tipo de controlador. Los nuevos valores de parámetro tras el ajuste están incluidos en la Tabla 3.2. Finalmente, la Figura 24 muestra la forma de onda para la acción de control PID de donde es posible notar incrementos en el esfuerzo de control para ciertos valores de carga conforme a lo presentado en la Figura 23. En este punto es conveniente aclarar que los resultados del controlador PID presentados en la

presente sección, constituyen el punto de partida para comparar el desempeño de este con técnicas no-convencionales de control como las presentadas a continuación: el control por modo deslizante y el control óptimo conmutado.

Figura 23. Comportamiento del sistema controlado por acción PID con ajuste fino



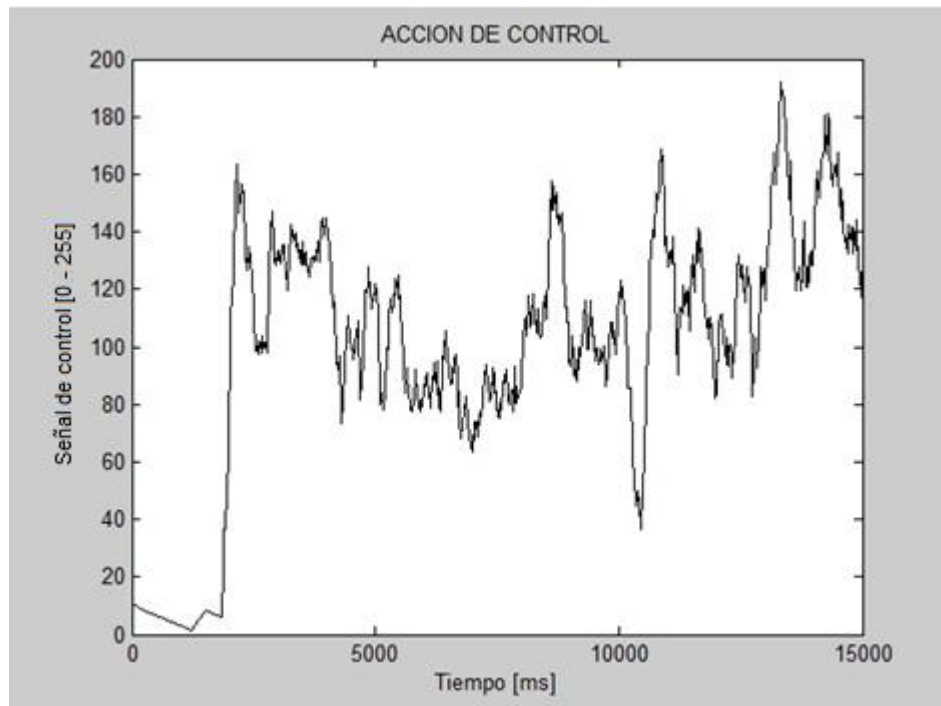
Fuente: Autores

Tabla 6. Parámetros de controlador PID modificados tras ajuste fino

VALORES PARAMETROS PID	
Kp	0.705
Ki	6.35
Kd	0.00078
T	0.010

Fuente: Autores

Figura 24. Acción de control PID



Fuente: Autores

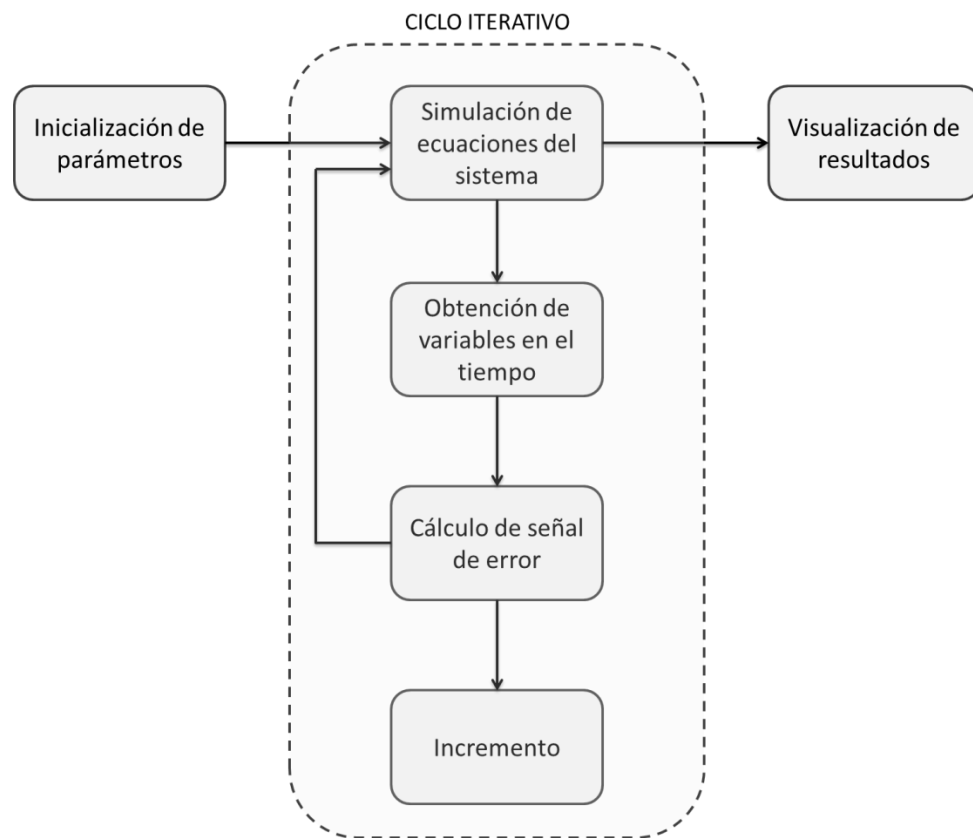
3.2 CONTROL POR MODOS DESLIZANTES

El control por modos deslizantes o SMC de su sigla en inglés (*sliding mode control*) es una técnica de control robusto en el espacio de estados cuyo principio se basa en transformar un problema de error n-dimensional (dimensión del vector de estado) en un problema 1-dimensional o escalar en términos de la denominada "superficie de deslizamiento". A continuación se ilustrará la evaluación de la acción de control obtenida mediante implementación numérica de las ecuaciones en MATLAB®. Una revisión de los fundamentos conceptuales del SMC, al igual que los desarrollos teóricos correspondientes al diseño de la estrategia de control por modos deslizantes para el sistema analizado, se detallan en el Anexo A.

3.2.1 Implementación del SMC en MATLAB®

La Figura 25 ilustra el diagrama de flujo para la rutina de cómputo diseñada, que permitió verificar numéricamente la acción de control por modos deslizantes para el sistema de control de velocidad. Dicha rutina fue ejecutada a partir de la consola de comandos de MATLAB®.

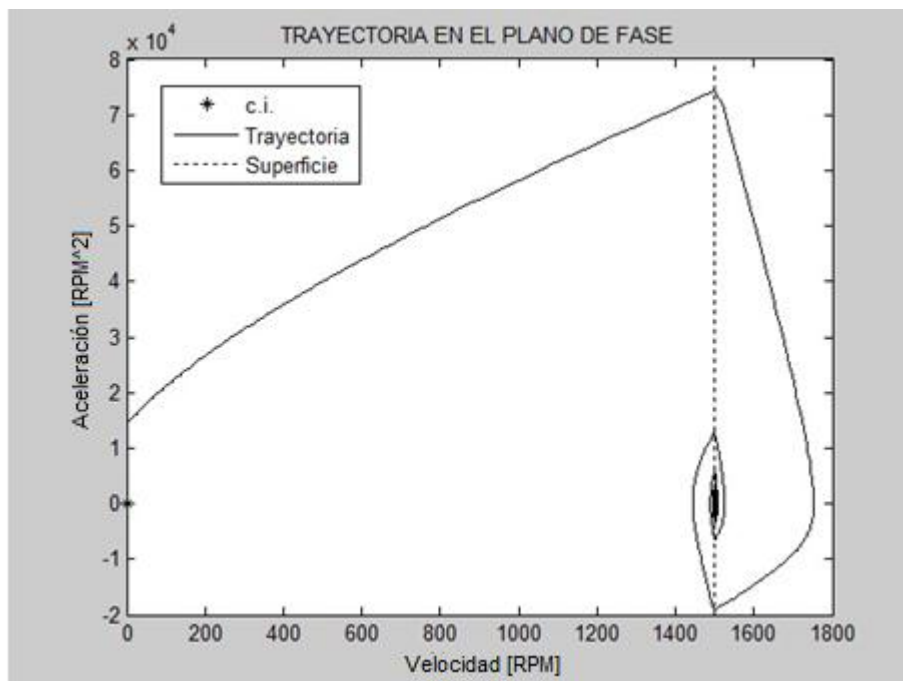
Figura 25. Diagrama de flujo para implementación en MATLAB® de la ley de control por modo deslizante.



Fuente: Autores

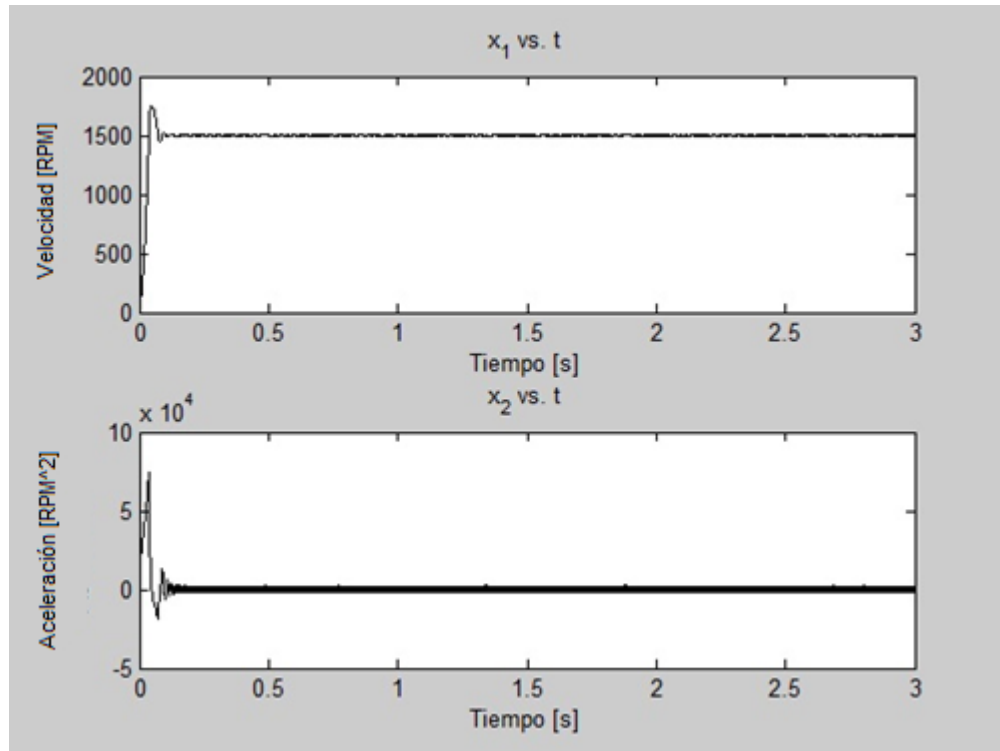
Con base en lo anterior, se realizaron algunas pruebas para verificar la eficiencia de la técnica para mantener la trayectoria en la superficie de deslizamiento deseada. En la Figura 26 se muestra la evolución convergente de la trayectoria en el plano de fase, desde una condición inicial arbitraria hasta una región representada por una línea vertical que corresponde con la superficie deslizante, pues allí para cualquier valor del tiempo se cumple que la velocidad de eje del motor es 1500 RPM, valor de referencia deseado. Para obtener esta gráfica se emplearon como parámetros del controlador $\lambda = 10000$ y $\eta = 100000$ obtenidos mediante ensayo y error. La Figura 27 muestra los resultados correspondientes a cada variable de estado ($x_1(t)$ representa la velocidad angular y $x_2(t)$ aceleración angular) de manera independiente en función del tiempo.

Figura 26. Comportamiento del sistema controlado en el plano de fase sin perturbación, obtenido en MATLAB®



Fuente: Autores

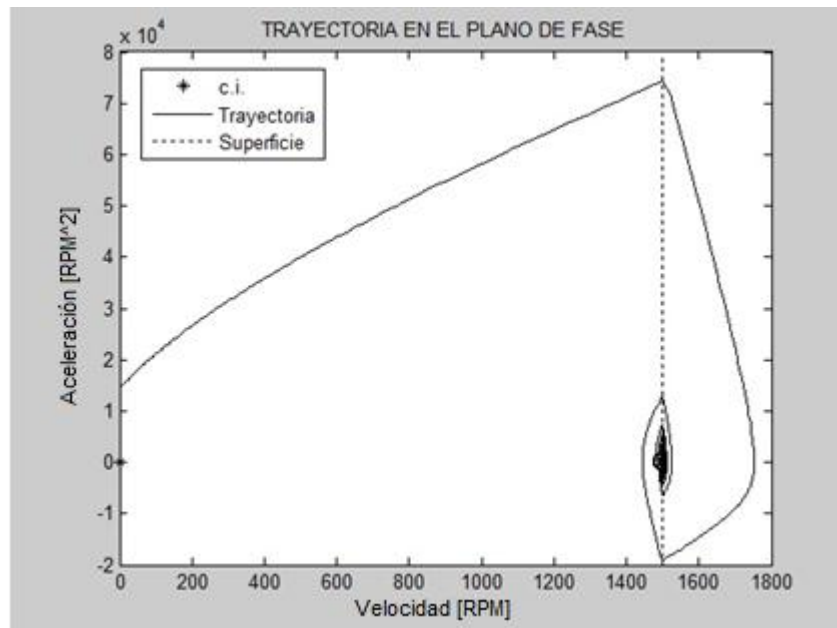
Figura 27. Comportamiento de variables de estado para el sistema controlado sin perturbación, obtenido en MATLAB®



Fuente: Autores

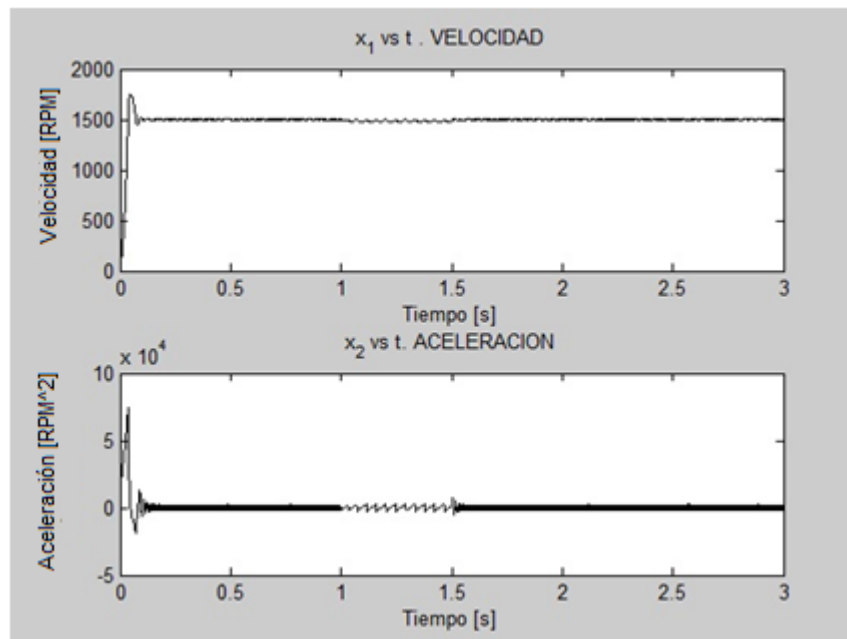
De otro lado, empleando los mismos parámetros de simulación anteriores, se agregó una perturbación al sistema correspondiente con un cambio en la carga equivalente al 20% en su valor nominal durante 0.5 [s]. Los resultados en el plano de fase (Figura 28) y a manera de evolución de variables en el tiempo (Figura 29) demuestran la efectividad de la técnica para forzar al sistema a una convergencia hacia la superficie de deslizamiento ante la acción de perturbaciones, lo cual ratifica la validez del diseño propuesto en la expresión (A.16).

Figura 28. Comportamiento del sistema controlado en el plano de fase con perturbación, obtenido en MATLAB®.



Fuente: Autores

Figura 29. Comportamiento de variables de estado para el sistema controlado con perturbación, obtenido en MATLAB®



Fuente: Autores

El lector interesado puede consultar [26] para resultados del análisis de la incidencia de los parámetros λ y η sobre el desempeño de la estrategia de control. En particular se destaca la verificación de ancho de banda para el caso de λ y de velocidad de convergencia por parte de η . Un análisis profundo para el comportamiento de estos parámetros, forma parte de actividades propuestas como trabajo futuro complementario al presente proyecto de grado.

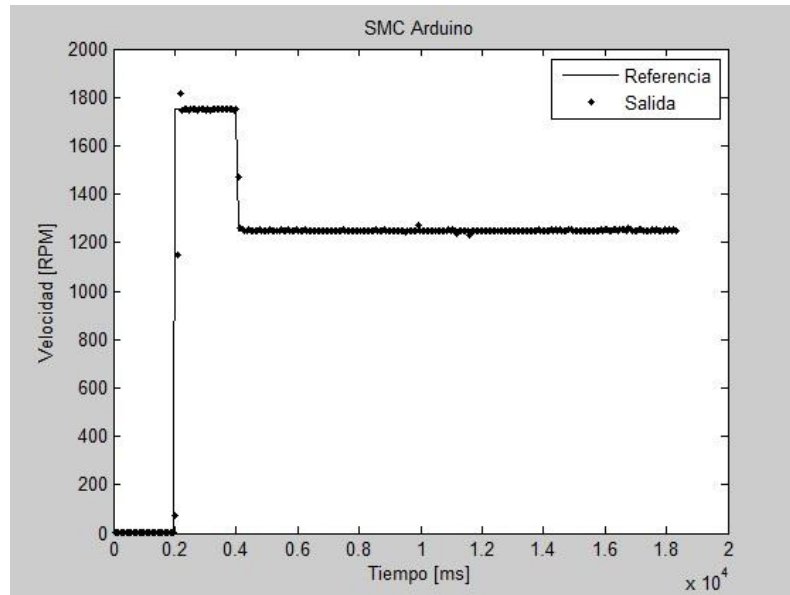
3.2.2 Implementación del SMC en Arduino MEGA®

Empleando el dispositivo digital programable para ejecutar la rutina de cálculo ilustrada en el flujograma de la Figura 25, se generaron con el control por modos deslizantes SMC resultados experimentales similares a los obtenidos para el control PID en la Sección 3.1. Es así como las Figuras 30 y 31 ilustran respetivamente la salida del sistema y el esfuerzo de control mediante un SMC ante la acción de perturbaciones. En comparación con el mismo escenario recreado previamente en la Figura 23, se observa una respuesta suave e invariante ante la acción de perturbaciones.

3.3 CONTROL ÓPTIMO CONMUTADO

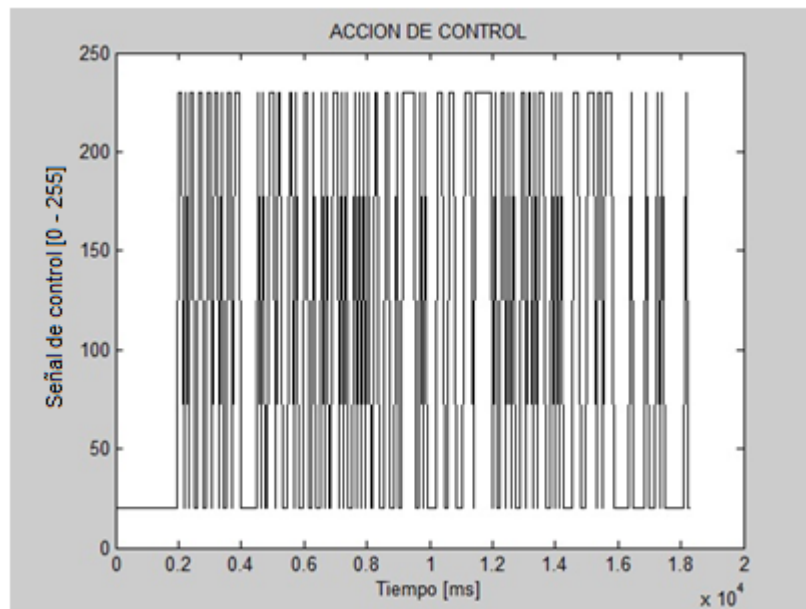
El control óptimo tiene como objetivo principal minimizar un índice de desempeño que permite obtener el mejor resultado posible para una aplicación determinada. A continuación, se visualizarán resultados de cómo, a partir de este concepto, es posible mantener en un valor deseado los niveles de velocidad angular en nuestro sistema. Una revisión de los fundamentos conceptuales del control óptimo, al igual que los desarrollos teóricos requeridos para implementar la técnica en el sistema analizado, se detallan en el Anexo B.

Figura 30. Comportamiento del sistema controlado por acción SMC.



Fuente: Autores

Figura 31. Señal de control del sistema controlado por acción SMC



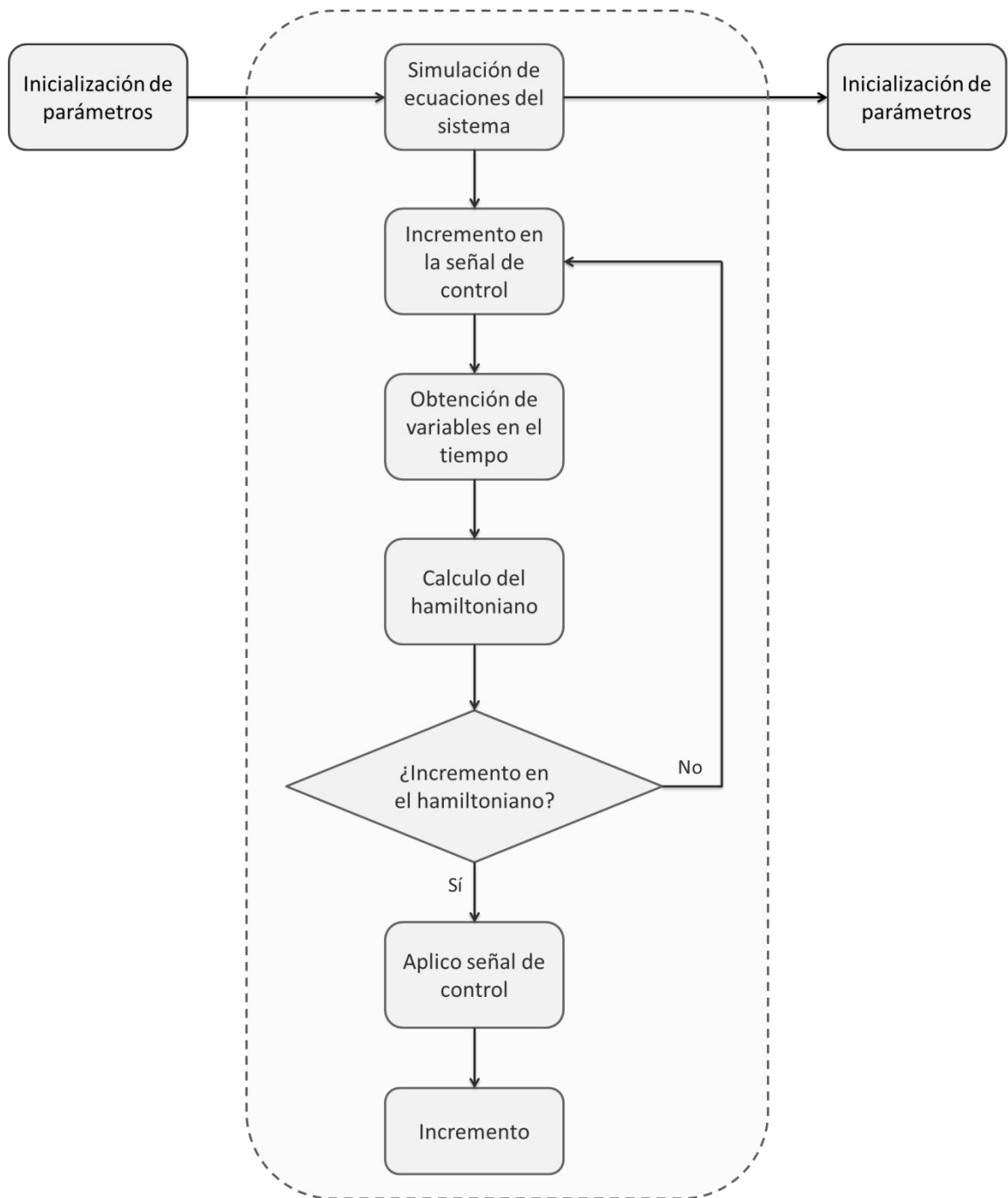
Fuente: Autores

3.3.1 Implementación del controlador óptimo en MATLAB®

El cálculo del Hamiltoniano para el modelo del sistema de control de velocidad fue obtenido mediante simulaciones en MATLAB®. El algoritmo implementado para recrear la solución numérica al problema de control óptimo se resume a manera de diagrama de bloques en la Figura 32. En el algoritmo, se supone un valor cualquiera para la señal de control dentro del intervalo de restricción ($0 \leq u(t) \leq 1$). Con esta condición de control inicial se determina la respectiva trayectoria de estado $x(t)$ dentro del intervalo $0 \leq t \leq T$, a partir de lo cual se calcula un valor del Hamiltoniano al final de dicho periodo. El valor calculado para el Hamiltoniano se compara con el correspondiente a un periodo anterior y si el valor actual es mayor que el anterior, se incrementa el valor de la señal de control al interior del conjunto admisible, en caso contrario se disminuye. El anterior proceso se repite para cada intervalo, determinando numéricamente los valores de la señal de control $u(t)$ que ajustados a las restricciones, maximizan el valor del Hamiltoniano $H(\lambda, X, u)$ (B.14) y por consiguiente minimizan la función objetivo dada en (B.5).

Con base en esta representación numérica fue posible obtener los resultados de simulación que a continuación se presentan. Inicialmente, en la Figura 33 se visualiza la manera como la velocidad de giro del motor se posiciona en el valor requerido de 1500 RPM, en un tiempo considerablemente pequeño tras partir de una condición inicial arbitraria. Lo anterior es consistente con la función objetivo planteada que busca eliminar el error en estado estacionario del sistema. Más aún, la alta velocidad de convergencia hacia el valor final deseado es una clara evidencia del efecto de la optimización en la acción de control.

Figura 32. Diagrama de bloques implementación en MATLAB® de la ley de control óptimo conmutado

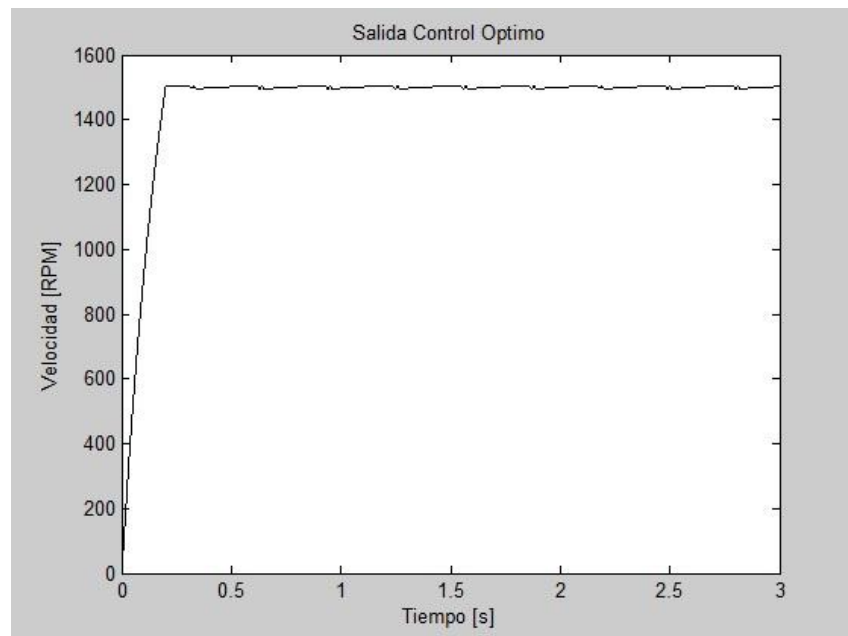


Fuente: Autores

La Figura 34 ilustra el comportamiento de la señal de control por conmutación $u(t)$ en la cual se verifica un valor medio constante en el esfuerzo de control (posterior a un pico inicial) durante el intervalo de simulación, debido a que en este caso no se consideró la influencia de perturbaciones en el sistema. De otro lado, la Figura 35 ilustra el efecto de una perturbación en la carga (correspondiente al 50% de su valor nominal) sobre la salida regulada de velocidad, durante el intervalo [1 – 1.5] [s].

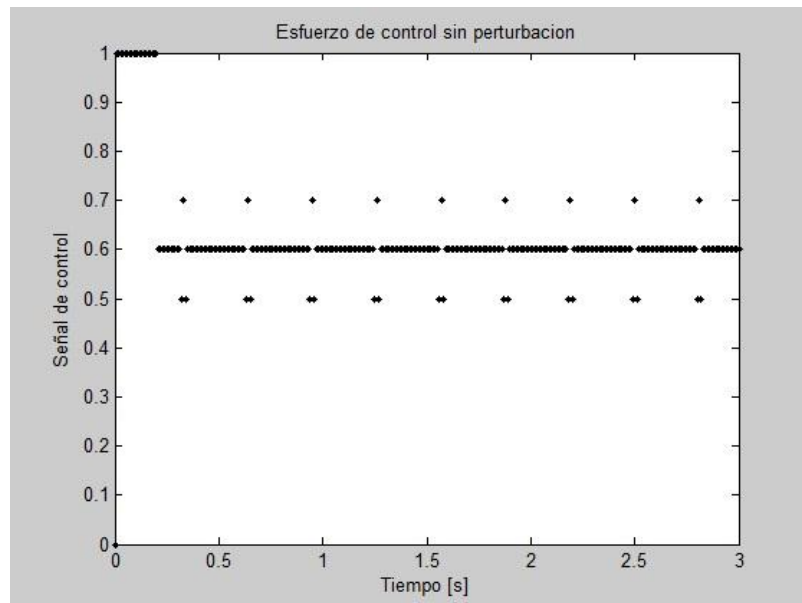
Como se observa en la figura, el efecto de la perturbación sobre la velocidad angular del motor es imperceptible. También es evidente la robustez de la técnica de control implementada a partir del comportamiento de la señal de control en la Figura 36, para la cual se obtiene un incremento significativo en el esfuerzo de control al adicionar cargas. Lo anterior compensa la deficiencia a la salida desvaneciendo el efecto de la perturbación.

Figura 33. Salida de velocidad tras acción de control óptimo obtenida en MATLAB®



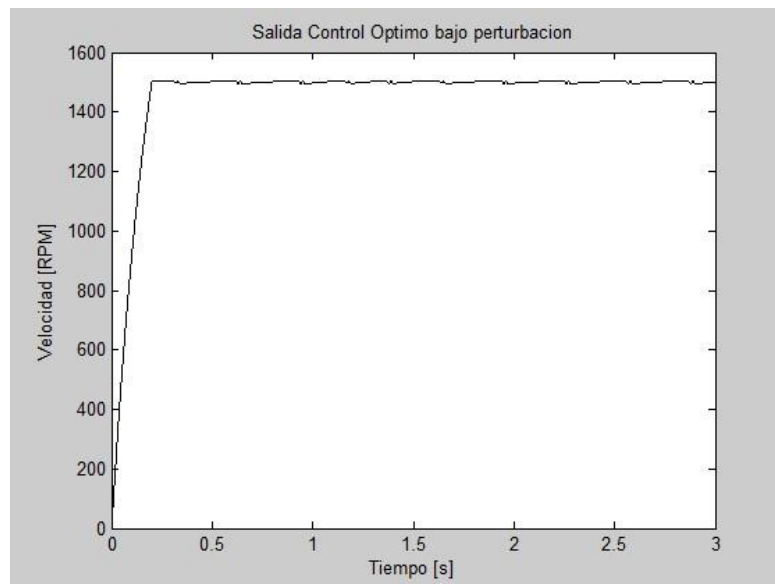
Fuente: Autores

Figura 34. Señal de control óptimo conmutado obtenida en MATLAB®



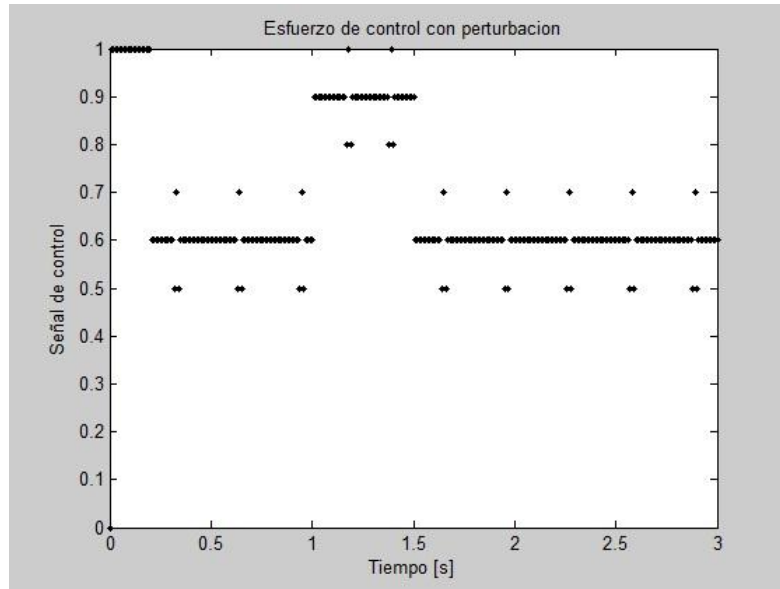
Fuente: Autores

Figura 35. Salida de velocidad tras acción de control óptimo con efecto de perturbaciones en la carga, obtenida en MATLAB®



Fuente: Autores

Figura 36. Señal de control óptimo conmutado con efecto de perturbaciones en la carga, obtenida en MATLAB®

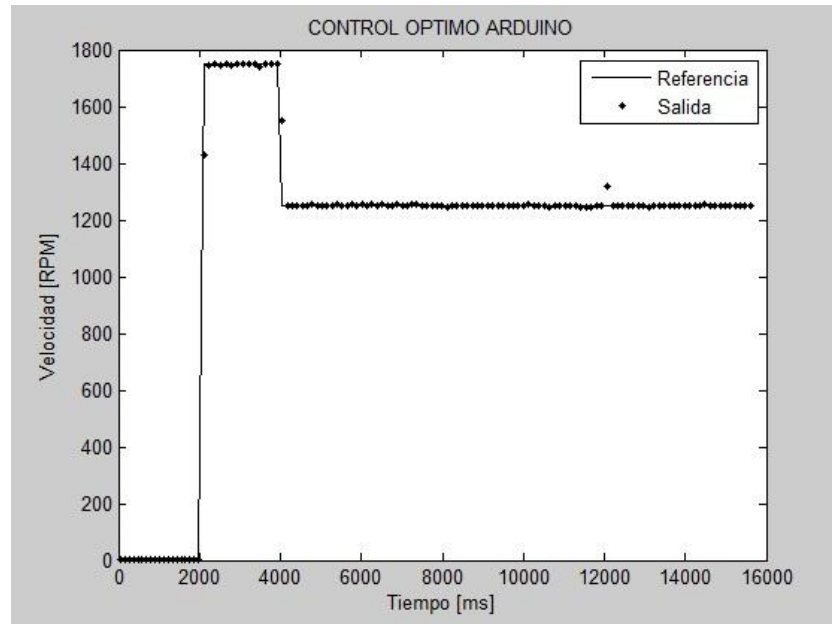


Fuente: Autores

3.3.2 Implementación del control óptimo conmutado en Arduino MEGA®

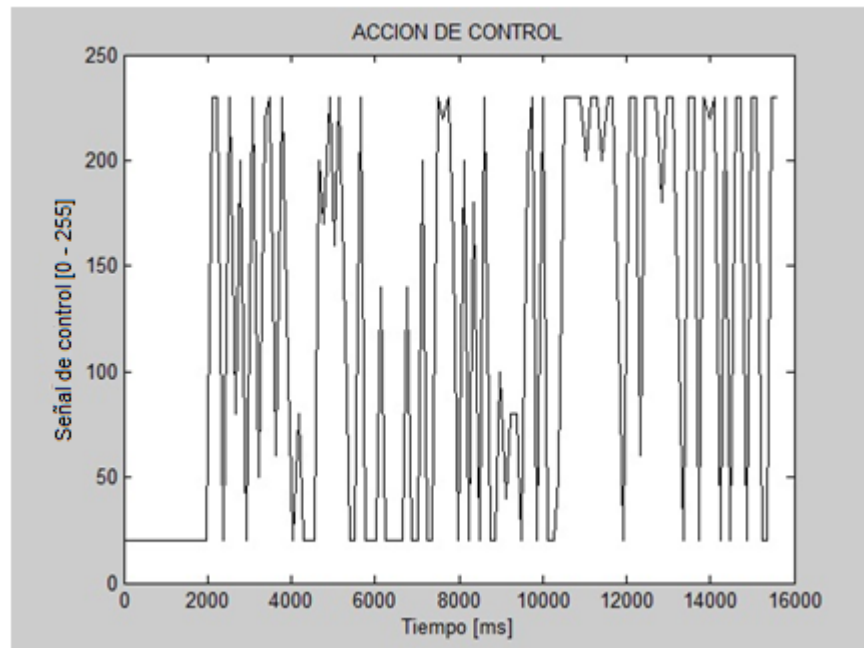
Empleando el dispositivo digital programable para ejecutar la rutina de cálculo ilustrada en el diagrama de bloques de la Figura 32, se generaron con el control óptimo conmutado resultados experimentales similares a los obtenidos para el control PID en la Sección 3.1. Es así como las Figuras 37 y 38 ilustran respectivamente la salida del sistema y la señal de control mediante control óptimo conmutado ante la acción de perturbaciones. En comparación con los resultados ilustrados para las técnicas de control presentadas de manera previa en las Figuras 23 (control PID) y 30 (SMC), se observa que el control óptimo nuevamente elimina la oscilación en la respuesta visualizada con el PID, presentando además un comportamiento levemente más suave que en el caso del SMC.

Figura 37. Comportamiento del sistema controlado por acción óptima conmutada



Fuente: Autores

Figura 38. Señal de control del sistema controlado por acción óptima conmutada



Fuente: Autores

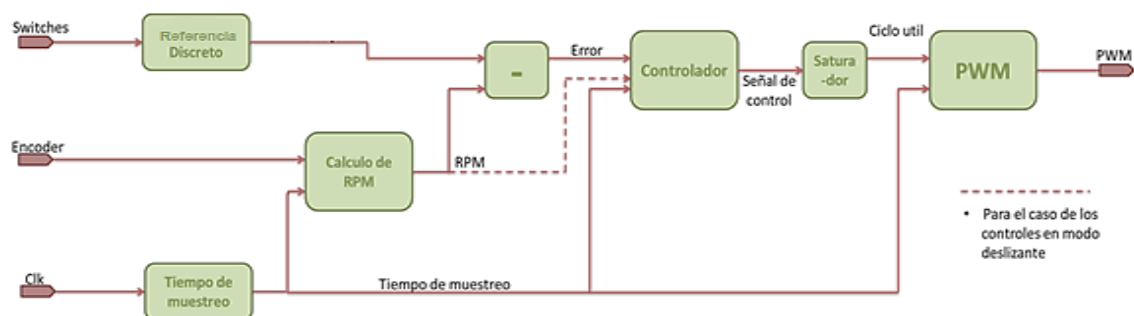
CAPITULO 4

En este capítulo se presenta la configuración y arquitectura de los componentes realizados en un dispositivo de arquitectura programable (Nexys 3 – Spartan 6) para ejecutar las tareas de control diseñadas. Inicialmente se abordarán los componentes comunes a los controles (PID, SMC y óptimo conmutado) particularizando posteriormente en las etapas de cada controlador. En el análisis se presentarán características de funcionamiento y resultados para las distintas pruebas ejecutadas.

4.1 DESCRIPCIÓN GENERAL DEL LAZO DE CONTROL

En el FPGA se realizó un circuito de propósito general para un lazo de control con pequeñas modificaciones para las distintas técnicas implementadas. La idea fue constituir una plantilla de base en la que sólo se reemplazará el algoritmo del controlador sin que los otros circuitos se vean afectados de manera significativa. En la Figura 39 se presenta el diagrama general implementado el cual consta de: un componente que interpreta la referencia; un componente encargado del cálculo de las RPM; otro componente contador que genera la señal con el periodo de muestreo; el componente controlador (que será reemplazado dependiendo del tipo de control) y finalmente un generador de PWM que será la salida del dispositivo programable al circuito actuador.

Figura 39. Circuito de propósito general para control en FPGA

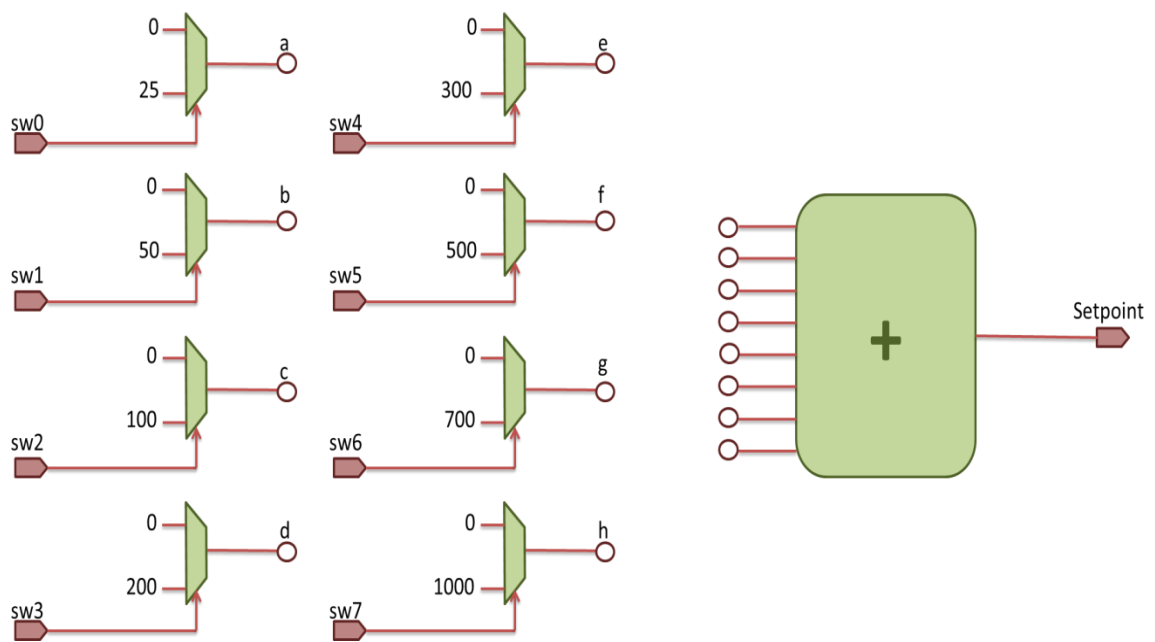


Fuente: Autores

4.1.1 Referencia discreta

Tal y como se ilustra en la Figura 40, el componente que interpreta la referencia discreta es un sumador que asigna un valor específico a un conjunto de interruptores de entrada y realiza la suma de los niveles que se encuentren en valor 1 lógico. Las magnitudes generadas en RPM son: 25, 50, 100, 200, 300, 500, 700 y 1000.

Figura 40. Circuito de referencia discreta



Fuente: Autores

4.1.2 Tiempo de muestreo

El bloque del tiempo de muestreo es sencillamente un contador de pulsos del reloj principal del sistema, el cual posee una frecuencia de 100 MHz, es decir un periodo de 10 ns. Dependiendo del tiempo de muestreo necesario para cada control se asigna el valor final del contador.

4.1.3 Lectura de velocidad en RPM

Para el cálculo de la velocidad en RPM se implementaron dos circuitos distintos: uno para el control PID y otro para los controles en modo deslizante y óptimo. Para el control PID (ver Figura 41) se usó un bloque contador para los pulsos provenientes de la entrada del encoder, que entrega el valor de los pulsos contados en el momento que se detecta un flanco de subida en el reloj del tiempo de muestreo, para luego reiniciar de nuevo la cuenta hasta otro cambio de nivel.

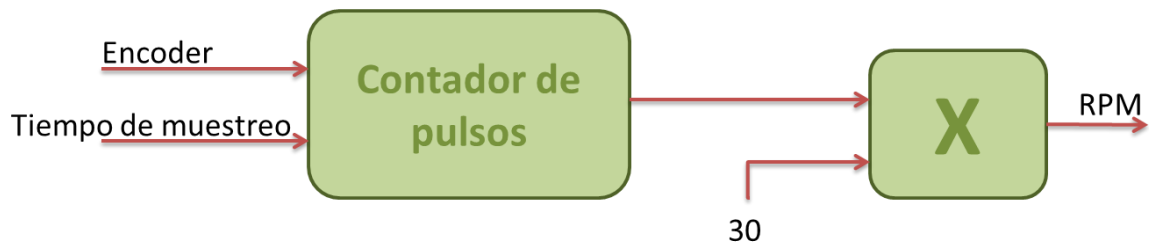
$$\frac{\#PE}{\#Ranuras} = \#Rts$$

$$\frac{\#Rts \times 60}{Ts} = 30$$

$\#PE = \text{número de pulsos del encoder}$

$$Ts = 10 \times 10^{-3}$$

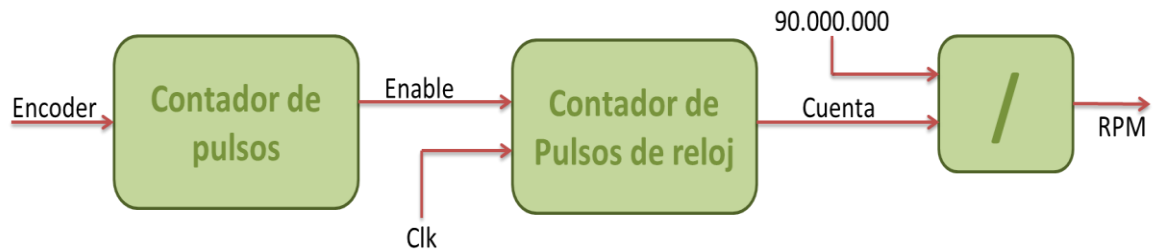
Figura 41. Cálculo de velocidad en RPM para control PID



Fuente: Autores

Para las técnicas de control por modos deslizantes y óptimo conmutado, se toma el tiempo a partir de la cuenta del reloj maestro en una cantidad de pulsos determinados (ver Figura 42). Esto debido a que el tiempo de muestreo en dichos controles es mucho menor, situación contraria al caso PID en donde la resolución se afecta para tiempos de muestreo reducidos.

Figura 42. Cálculo de velocidad en RPM para SMC y óptimo conmutado



Fuente: Autores

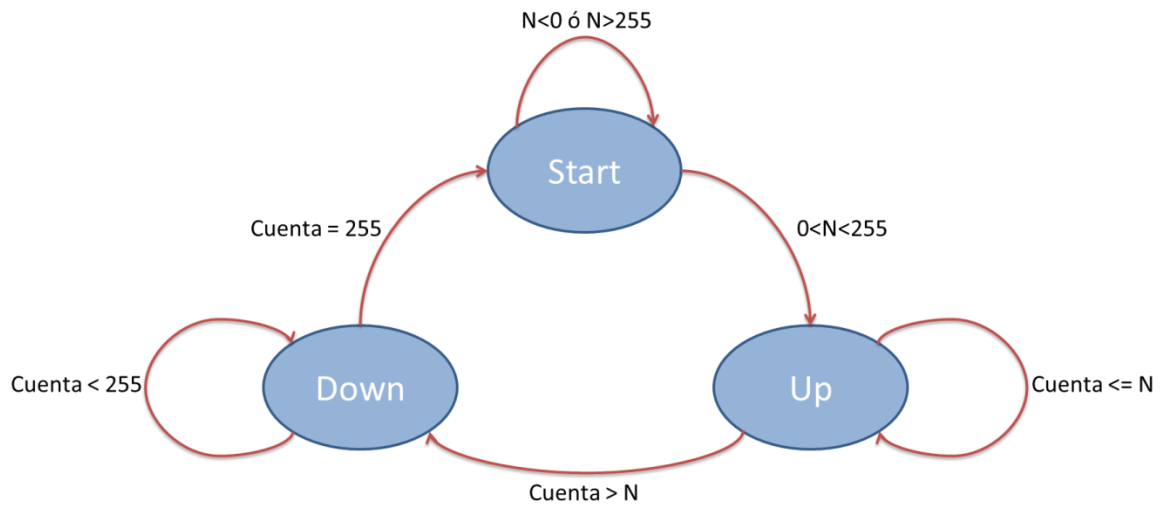
4.1.4 Generador de PWM

El generador de PWM cuenta con un divisor de frecuencia que produce 256 pulsos para el ciclo útil en cada periodo del PWM. Una máquina de estados es la clave de este circuito. La entrada del generador es un valor entre 0 y 255 que representa en escala el ciclo útil del PWM. La máquina de estados realiza la cuenta de ciclos de reloj hasta el número ingresado.

4.1.4.1 Máquina de estados

Tal y como se ilustra en la Figura 43, la máquina está compuesta por tres estados: “*start*”, “*up*” y “*down*”. En el estado de “*start*” se verifica que la entrada sea un valor entre 0 y 255. Después, se pasa inmediatamente al estado “*up*” en donde se realiza una cuenta de ciclos de reloj igual al número de entrada durante la cual la salida del PWM siempre está en un nivel alto. De ésta forma se genera el ciclo útil del periodo del PWM. Cuando la cuenta alcanza el valor del número de entrada ésta no se detiene, pero la maquina cambia al estado “*down*”. En este estado se completa la cuenta hasta 255 con salida en nivel bajo, generando la parte restante del periodo de la señal de PWM.

Figura 43. Máquina de estados del generador de PWM



Fuente: Autores

4.2 IMPLEMENTACIÓN DE CONTROLADORES EN VHDL

4.2.1 Control PID

Para el circuito PID se utilizó un periodo de muestreo de 10 ms. Inicialmente, se realizó el cambio del modelo de la planta continua (2.1) a dominio z en (4.1) para proceder a calcular el controlador PID discreto (4.2)-(4.3) como se ha ilustrado en el capítulo 2. Una vez hecho esto, se representa la ecuación en diferencias del controlador (4.4) a partir de: 3 registros, uno para la salida de la señal de control y otros para mantener los valores de los dos errores anteriores al presente; 3 multiplicadores en los cuales se realiza el producto entre la señal de error y los respectivos coeficientes de la ecuación en diferencias y; 2 sumadores que realizan la adición de los términos de toda la ecuación. Ver Figura 44.

$$G(z) = \frac{0.1581z + 0.1371}{z^2 - 1.631z + 0.652} \quad (4.1)$$

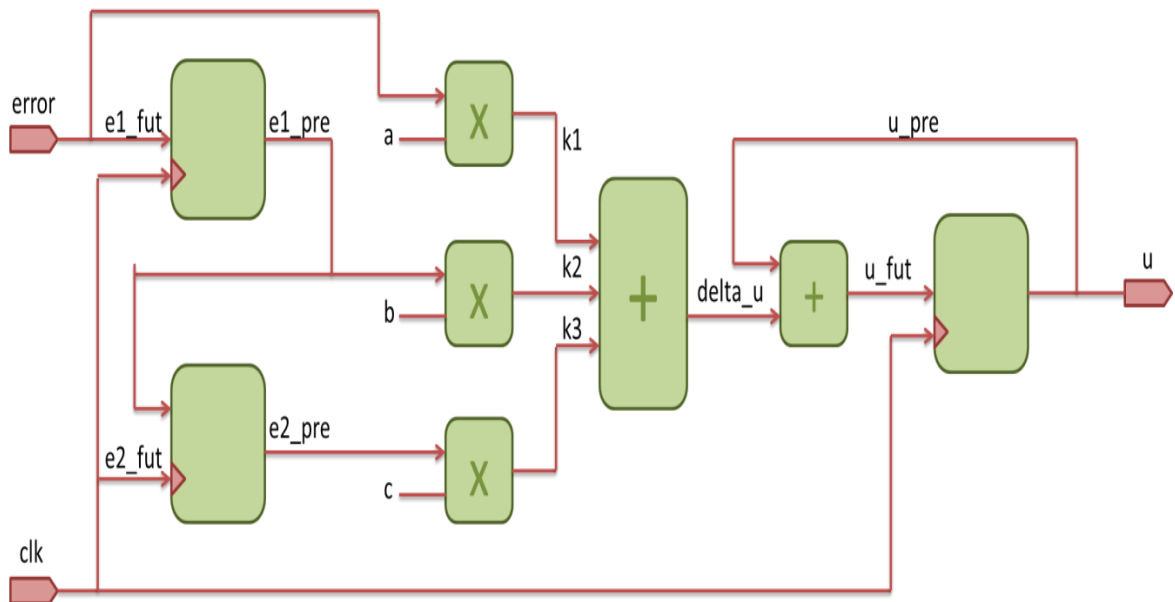
$$D(z) = \frac{1.499z^2 - 2.197z + 0.7655}{z^2 - z} \quad (4.2)$$

$$\frac{U(z)}{E(z)} = \frac{1.499 - 2.197z^{-1} + 0.7655z^{-2}}{1 - z^{-1}} \quad (4.3)$$

$$u[k] = u[k - 1] + 1.5e[k] - 2.197e[k - 1] + 0.7655e[k - 2] \quad (4.4)$$

$$u[k] = u_{pre} + k1 + k2 + k3 \quad (4.5)$$

Figura 44. Circuito implementado para el controlador PID



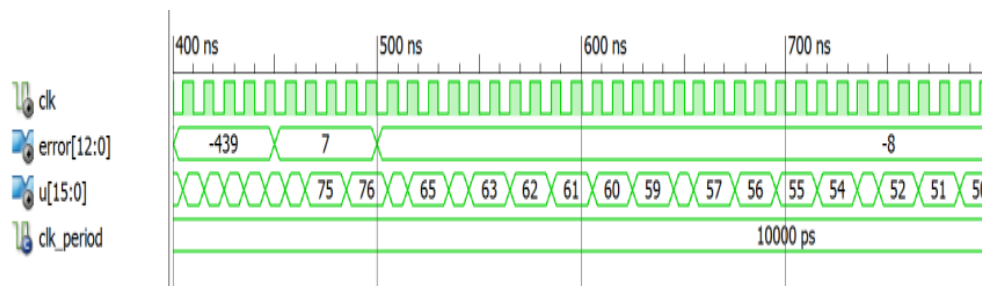
Fuente: Autores

4.2.1.1 Simulación de PID

Como paso previo a las pruebas experimentales, se adoptó una metodología de simulación para verificar la coincidencia numérica de los algoritmos de control en VHDL respecto a los códigos implementados en MATLAB®, presentados en el capítulo 3. Para ello, se seleccionó un conjunto de datos de entrada (señal de error) que una vez aplicado a la rutina de simulación, genera una correspondiente

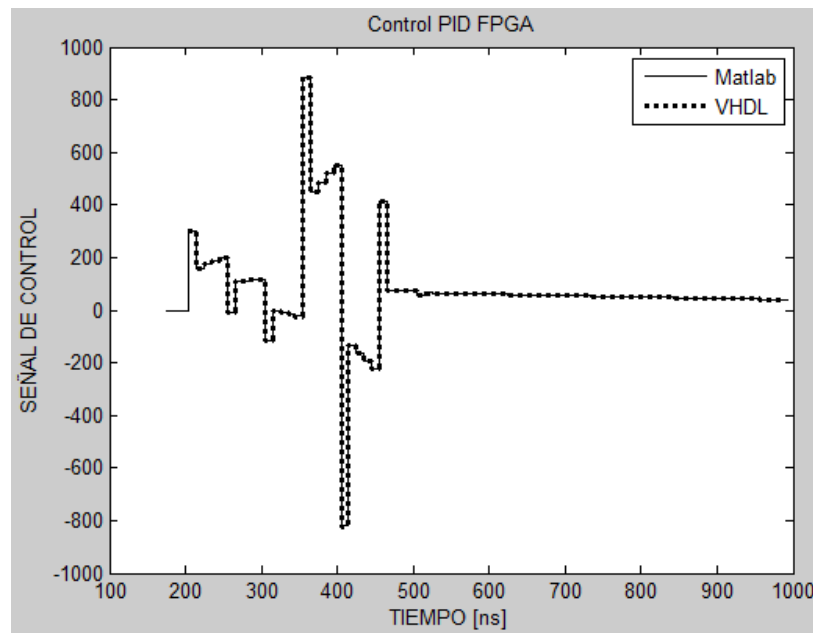
respuesta (señal de control) tras evaluar el algoritmo. Los resultados de simulación para VHDL fueron obtenidos empleando la herramienta *Isim Simulator* de Xilinx® (ver Figura 45). Es importante hacer notar que para comparar los resultados de simulación entre MATLAB® y VHDL fue necesario exportar los datos del *Isim Simulator* a un archivo de texto para posteriormente accederlos desde la ventana de comandos de MATLAB®.

Figura 45. Simulación en *Isim* del control PID



Fuente: Autores

Figura 46. Simulaciones en MATLAB® y VHDL para control PID

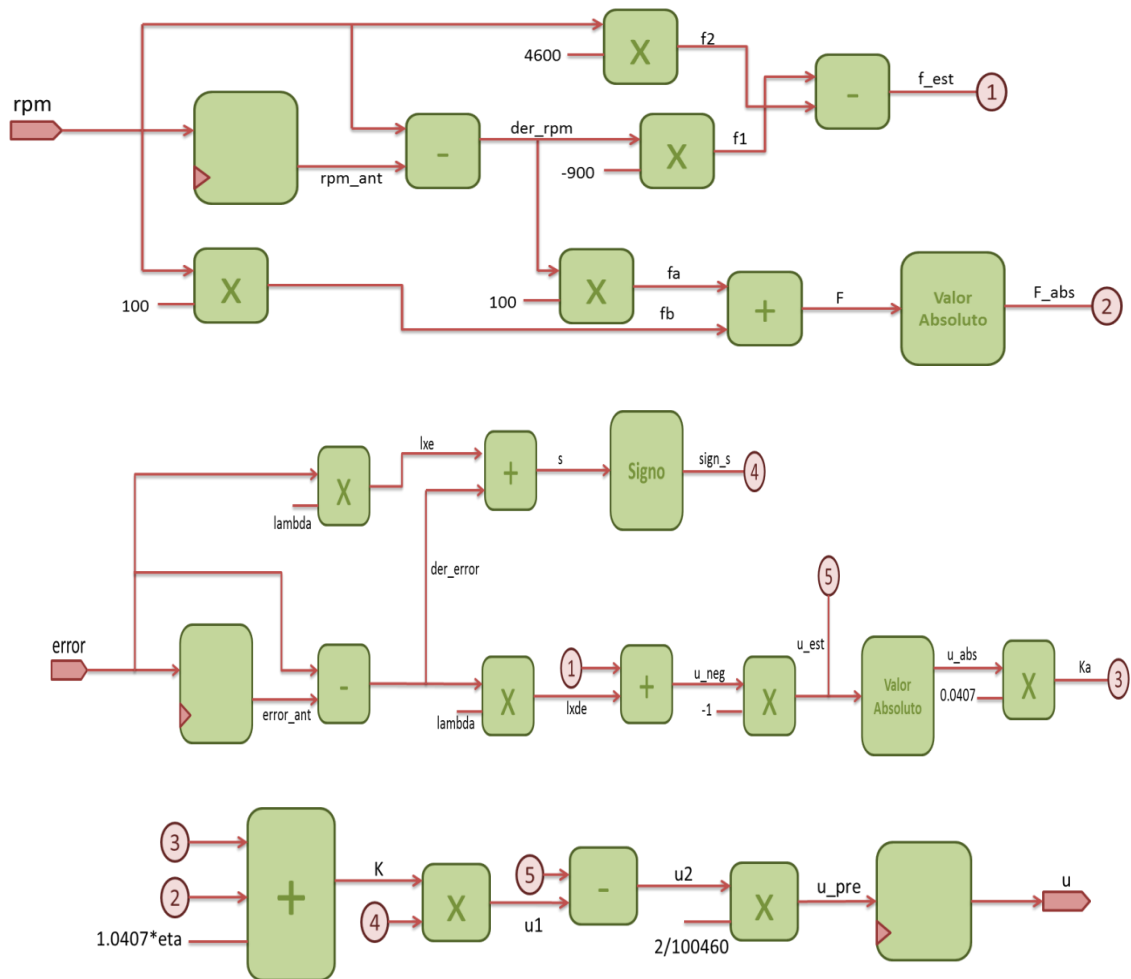


Fuente: Autores

4.2.2 Control por modos deslizantes

Para este circuito se utilizó un periodo de muestreo 1 ms, valor determinado empíricamente para un apropiado desempeño de la rutina en el microcontrolador. El circuito para su realización cuenta con 3 registros para almacenar la medida de RPM anterior, el error anterior y la salida de control actual. Este circuito cuenta además con varios multiplicadores, sumadores y bloques de valor absoluto que funcionan mediante identificación y multiplicación de signos. En la Figura 47 se ilustra el diagrama para el circuito implementado.

Figura 47. Circuito implementado para el SMC

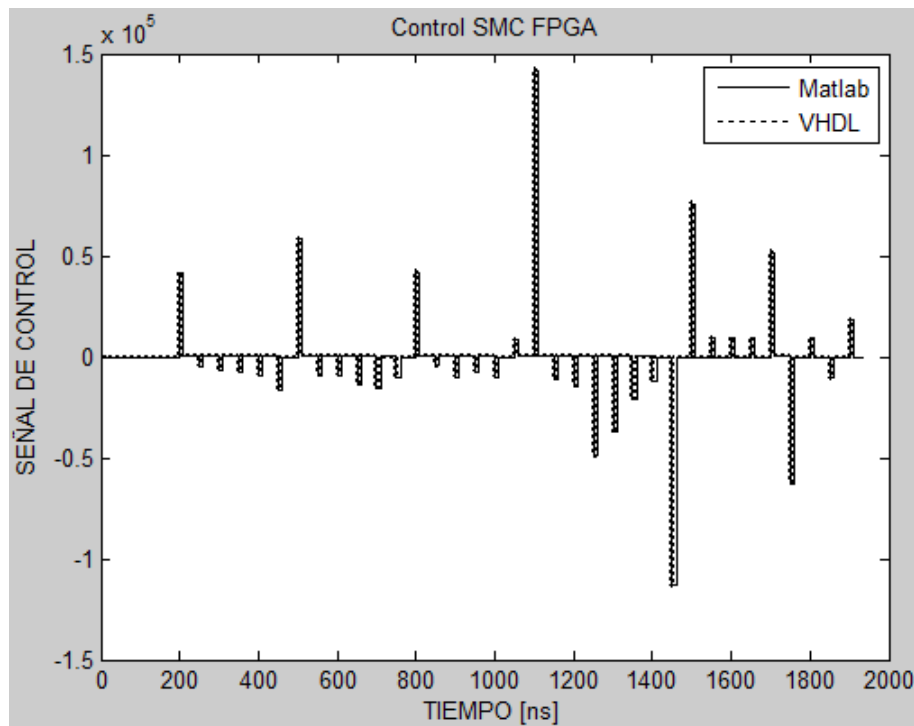


Fuente: Autores

4.2.2.1 Simulación de SMC

Similar al caso descrito en la Sección 4.2.1.1, se realizó comparación entre las simulaciones de los algoritmos de MATLAB® y VHDL para la estrategia de control por modos deslizantes. La Figura 48 presenta los resultados correspondientes a la superposición para ambos resultados.

Figura 48. Simulaciones en MATLAB® y VHDL para control SMC



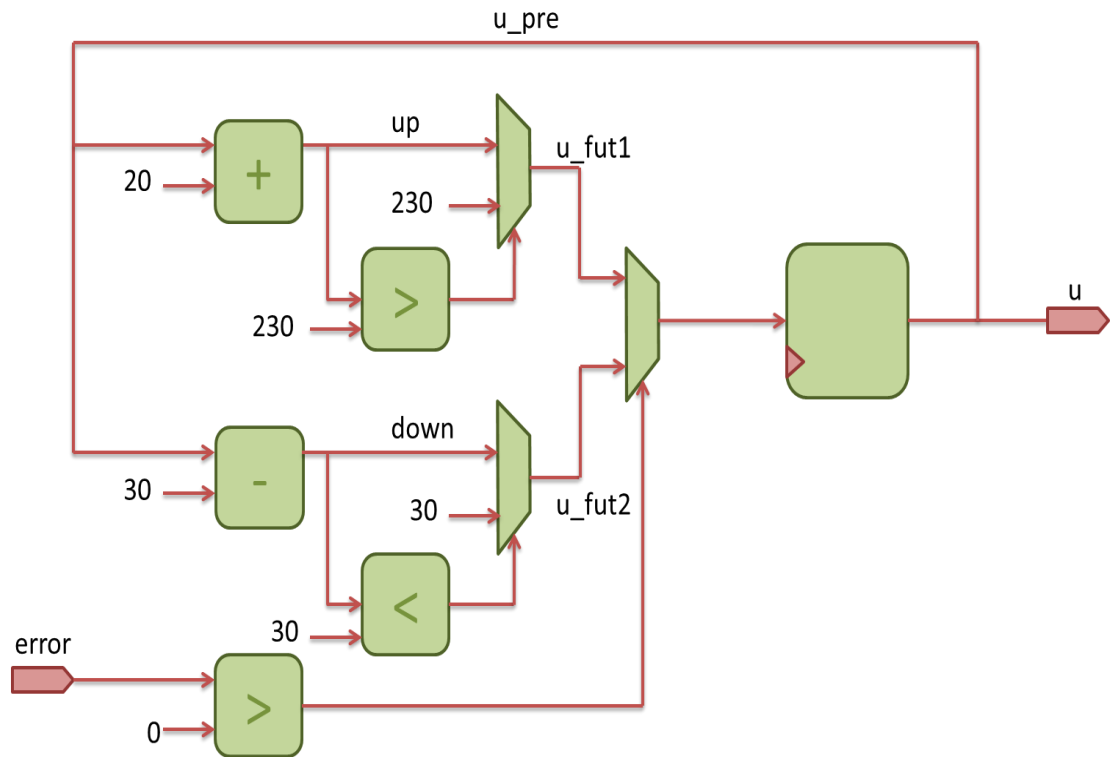
Fuente: Autores

4.2.3 Control óptimo conmutado

La implementación VHDL para la estrategia de control óptimo conmutado se obtuvo de manera relativamente sencilla. Tal y como se presenta en la Figura 49, se realizó una comparación en el error para determinar su signo. De ser positivo, la señal de control debe disminuir. En caso contrario deberá aumentar. La señal

de control aumenta en intervalos de 20 unidades hasta 230 unidades y disminuye en intervalos de 30 unidades, debido a que el motor responde más lento ante desaceleraciones.

Figura 49. Circuito implementado para el controlador óptimo conmutado

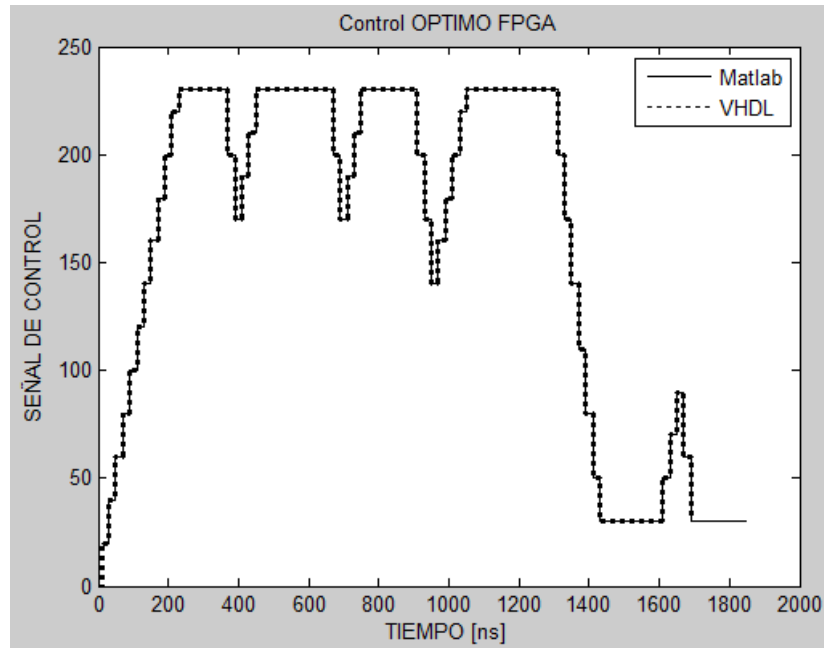


Fuente: Autores

4.2.3.1 Simulación de control óptimo

Similar a los casos descritos en la Secciones 4.2.1.1 y 4.2.2.1, se realizó comparación entre las simulaciones de los algoritmos de MATLAB® y VHDL para la estrategia de control óptimo conmutado. La Figura 50 presenta los resultados correspondientes a la superposición para ambas respuestas.

Figura 50. Simulaciones en MATLAB® y VHDL para control óptimo conmutado



Fuente: Autores

4.3 VERIFICACIÓN EXPERIMENTAL DEL DESEMPEÑO DE CONTROLADORES EN VHDL

Posterior al diseño e implementación en VHDL de las rutinas de control analizadas en el Capítulo 3 y una vez verificada la pertinencia de las implementaciones mediante comparación de los resultados de simulación para los diferentes algoritmos, es momento de presentar los resultados experimentales definitivos de las rutinas ejecutadas sobre la arquitectura configurable de un FPGA.

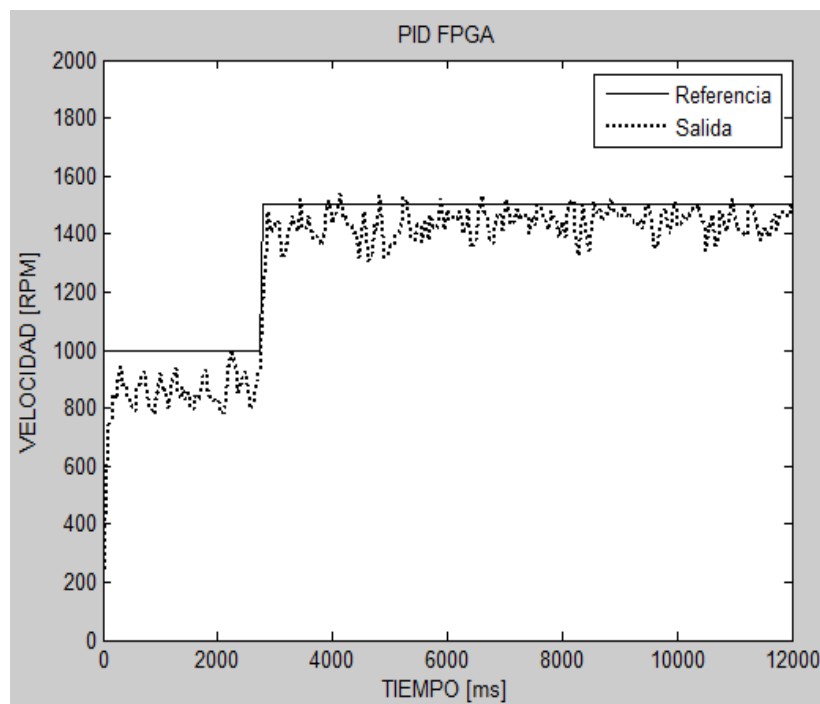
4.3.1 Implementación de PID en FPGA

La Figura 51 muestra la reproducción de las condiciones de operación para el sistema controlado mediante acción PID bajo la acción de perturbaciones

previamente analizado en la Figura 23, esta vez siendo ejecutado en un FPGA. Como se observa de la Figura 51 el control busca seguir la señal de referencia, presentando fuertes oscilaciones. También se verifica la eliminación del efecto de las perturbaciones. El comportamiento del sistema controlado es para este caso comparable con la implementación de la misma técnica en el microcontrolador.

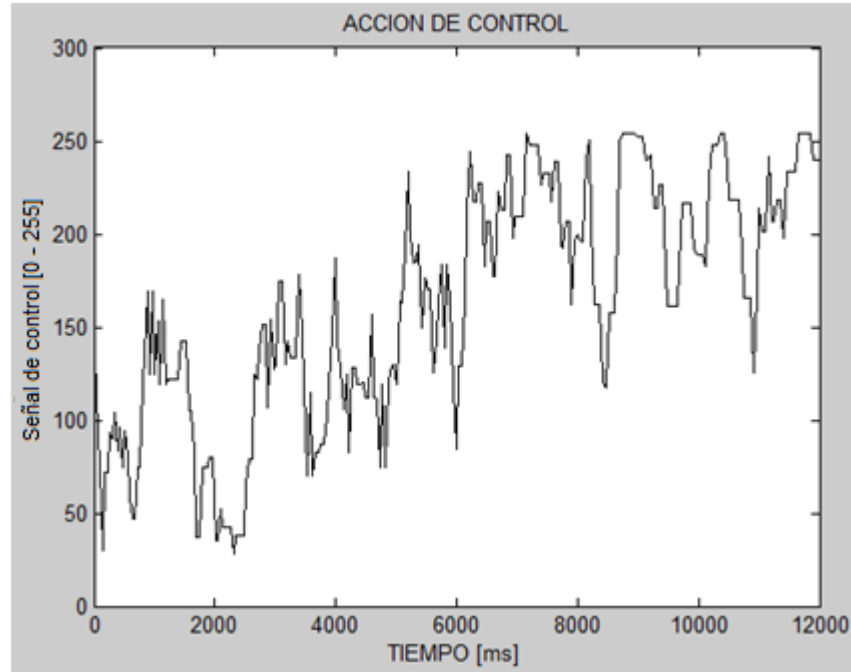
De manera similar, la Figura 52 muestra la señal de control correspondiente donde se verifica el incremento en el esfuerzo de control para la región del tiempo en la cual existe la incidencia de perturbaciones de carga (a partir de 6 segundos).

Figura 51. Comportamiento del sistema controlado por acción PID en FPGA



Fuente: Autores

Figura 52. Acción de control PID en FPGA



Fuente: Autores

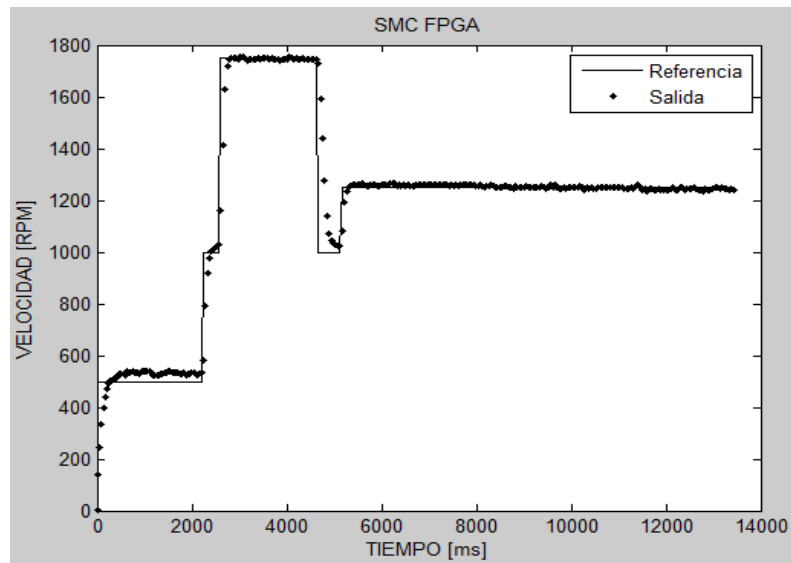
4.3.2 Implementación del SMC en FPGA

Empleando un FPGA para ejecutar la rutina de cálculo ilustrada en el flujograma de la Figura 25, se generaron con el control por modos deslizantes SMC resultados experimentales similares a los obtenidos en la Sección 3.2. Es así como la Figura 53 ilustra el resultado del comportamiento del sistema controlado mediante un SMC ante la acción de perturbaciones.

En comparación con el mismo escenario recreado previamente en la Figura 30, se observa que el control por modos deslizantes nuevamente presenta una respuesta sin oscilaciones significativas que atenúa el efecto de las perturbaciones aplicadas al sistema (a partir de los 8 segundos).

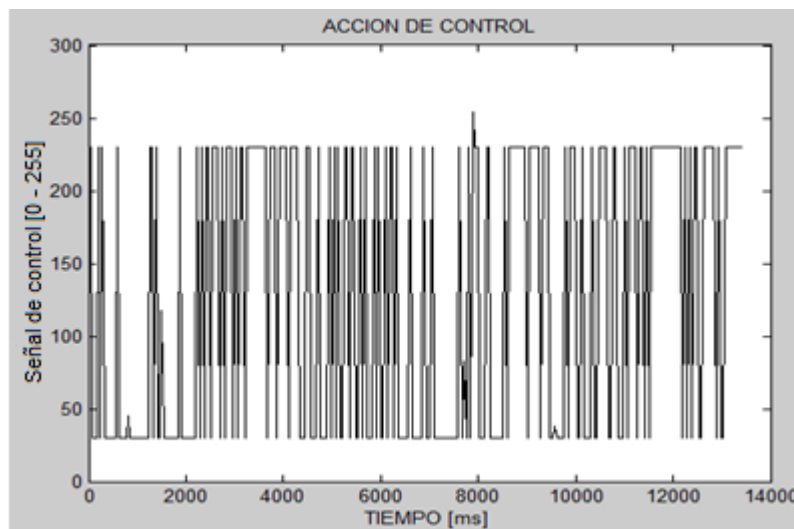
De manera similar, la Figura 54 muestra la señal de control correspondiente en la cual es evidente el aumento de ciclo útil (nivel alto) y por consiguiente un mayor esfuerzo de control, durante la influencia de perturbaciones en el sistema.

Figura 53. Comportamiento del sistema controlado por acción SMC



Fuente: Autores

Figura 54. Acción de control SMC en FPGA



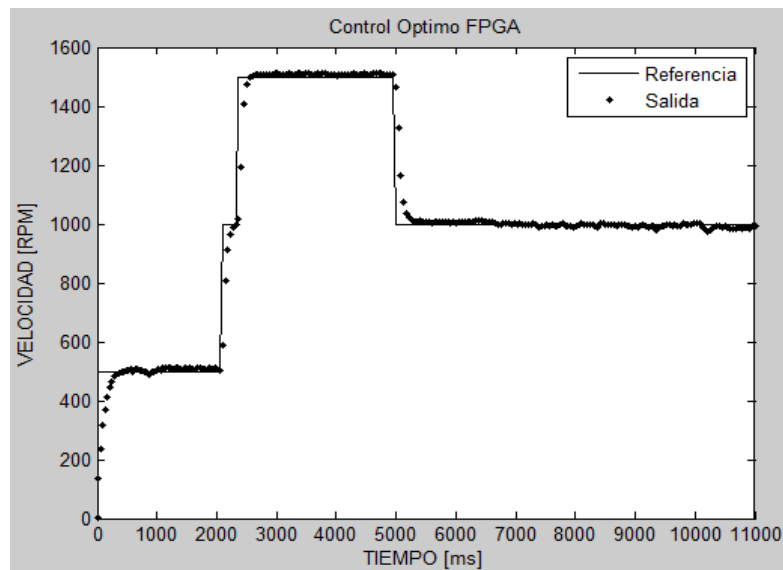
Fuente: Autores.

4.3.3 Implementación del control óptimo conmutado en FPGA

Empleando un FPGA ejecutar la rutina de cálculo ilustrada en el flujograma de la Figura 32, se generaron con el control óptimo conmutado resultados experimentales similares a los obtenidos en la Sección 3.3. Es así como la Figura 55 ilustra el resultado del comportamiento del sistema controlado mediante control óptimo conmutado ante la acción de perturbaciones. En comparación con los resultados obtenidos para las dos técnicas de control presentadas de manera previa en las Figuras 51 (control PID) y 53 (SMC), se observa que la acción de control óptimo presenta un comportamiento similar al SMC. También se nota una leve oscilación durante la influencia de perturbaciones (7 segundos) que sin embargo no deterioran significativamente el desempeño de la respuesta.

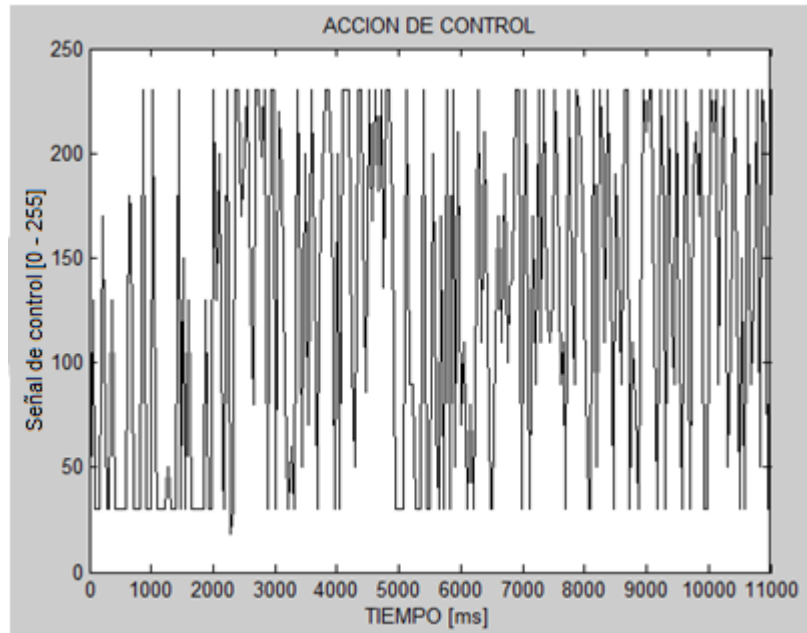
De manera similar, la Figura 56 muestra la señal de control correspondiente en la cual se nota una mayor actividad de conmutación durante los transitorios (cambios en el valor de referencia) y ante la influencia de perturbaciones (a partir 7 segundos en la gráfica).

Figura 55. Comportamiento del sistema controlado por acción óptima conmutada en FPGA.



Fuente: Autores

Figura 56. Acción de control óptimo conmutado en FPGA



Fuente: Autores

CONCLUSIONES

- Se realizó un estudio comparativo del desempeño de técnicas de control conmutado implementadas en dispositivos programables. Se verifica que las técnicas conmutadas (SMC y control óptimo) presentan un desempeño superior al de una técnica PID convencional para seguimiento a valores de referencia de velocidad ante la acción de perturbaciones en la carga del eje del motor.
- Se utilizó una metodología de diseño de sistemas embebidos basada en comparación por simulación de la respuesta del bloque de control, implementado posteriormente en una arquitectura configurable del tipo FPGA.
- Se aplicaron técnicas de control conmutado robusto (SMC) y óptimo, implementadas sobre un dispositivo de arquitectura configurable (FPGA) para controlar la velocidad de un motor de corriente continua. Se pudo verificar un apropiado desempeño de las técnicas a partir de pruebas experimentales.
- Se aplicaron técnicas de control conmutado implementadas sobre un dispositivo de arquitectura configurada (microcontrolador Arduino) para controlar la velocidad de un motor de corriente continua. Se pudo verificar un apropiado desempeño de las técnicas a partir de pruebas experimentales.
- Como era de esperarse, el desempeño dinámico de las técnicas de control implementadas es independiente de la plataforma de implementación en hardware. De otro lado su dificultad de implementación fue un poco más complicada en VHDL que Ansi C® (lenguaje de base para programación del microcontrolador).

RECOMENDACIONES Y TRABAJO FUTURO

Debido a que se utilizó como dispositivo de adquisición de datos del sistema el mismo dispositivo (microcontrolador) empleado para la ejecución de algoritmos, se generaron inconvenientes para obtener muestreos a mayor velocidad, debido a los tiempos de proceso. Para evitar esto, los autores recomiendan la utilización separada de dispositivos de adquisición y proceso.

Los algoritmos implementados fueron adecuados para ser comparados en condiciones de operación similares. Sin embargo, se debe hacer notar que los tiempos de ejecución en el dispositivo de arquitectura configurable (FPGA) son generalmente menores debido a mayores prestaciones de ciclos de máquina.

Como trabajo futuro, se proponen:

- Análisis más estrictos respecto a la robustez de la técnica SMC. En particular, debe realizarse mayor estudio experimental de los parámetros λ y η .
- Análisis más estrictos respecto a la optimalidad generada por el cálculo del Hamiltoniano según el método descrito. En particular, se debe valorar el índice de desempeño J propuesto y su relación con H .
- Optimización de códigos y de asignación de recursos de máquina en las implementaciones digitales.
- Análisis e implementación computacional de técnicas de control de mayor complejidad.

BIBLIOGRAFÍA

- [18]. A. Jantsch and I. Sander. "Models of Computation and Languages for Embedded System Design". IEE Proceedings – Computers and Digital Techniques. Vol 152. Nº 2. pp 114-129. Marzo. 2005.
- [10]. A. J. Bernal, D. Pineda, M. Velázquez. Control en Modo Deslizante con estimación de la perturbación aplicado a un modelo de turbina hidráulica. Trabajo de grado. Universidad Industrial de Santander. 2011.
- [2]. A. Zinober. Deterministic Control of Uncertain Systems. London: Peter Peregrinus Press. ISBN978-0863411700. 1990.
- [25]. CARLOS ALBERTO REY SOTO, "Análisis de Señales y Sistemas en tiempo continuo y en tiempo discreto con aplicaciones en MATLAB" En: Venezuela 2006. ed:Fondo Editorial UNET ISBN: 980-6300-25-4 v. 1 pags. 561.
- [20]. C. A. Fajardo. "Apropiación tecnológica del diseño de Embedded Systems implementados sobre FPGAs y CPLDs". Tesis de Maestría. Universidad Industrial de Santander. 2010.
- [9]. Elena Muñoz. Carlos A. Gaviria. Control PID Multivariable y Modos Deslizantes de un Robot SCARA. Departamento de Electrónica, Instrumentación y control. Universidad del Cauca.
- [1]. ELSO. Jorge, EGUINO. Irene, MOLINS. Carlos, GARCIA. Mario. Estrategia de control conmutado para la mejora de controladores lineales. XXVIII Jornadas de Automática. Universidad de Huelva. 2007.
- [19]. G. Karsai, S. Neema and D. Sharp. "Model-driven Architecture for Embedded Software: a Synopsis and an Example". Science of Computer Programming. Vol 73. Nº 1. pp 26-38. Septiembre. 2008.

- [8]. Ibargüen Francisco, Morales Jairo, Moreno John, Cardona Jaiber, Muños Pablo. Dinámica hamiltoniana y control por modos deslizantes del péndulo de furuta. Ingeniería y Desarrollo. Universidad del Norte. 26: 65-82, 2009.
- [15]. J. Cabrera, J. Velasco. “Implementación de Filtros Análogos usando PSoCs/FPAAs”. Revista colombiana de Física. Vol 40. N° 2. 2008.
- [6]. J. K. Pieper, B. W. Surgenor. “Discrete-time Sliding Mode Control Applied to a Gantry Crane”. On proceedings of the 33th IEEE Conference on Decisions and Control. Lake Buena Vista - FL. USA. 1994.
- [11]. J. Vargas. Control en Modo Deslizante aplicado a un sistema de segundo orden con retardo en la salida. Trabajo de grado. Universidad Industrial de Santander. 2006.
- [24]. Katsuhiko Ogata, Ingeniería de control moderno, cuarta edición, Editorial Prentice Hall, 2003.
- [26]. LADY GISSELIETH FUENTES DIAZ, YULY KARINA TUTIRA CACERES, PEDROJULIO ESTEBAN RIOS, “DISEÑO Y SIMULACION DE ESTRATEGIAS DE CONTROL CONMUTADO EN CONVERTIDORES DE POTENCIA”, Bucaramanga, 2012, 86 p. Trabajo de grado (Ingeniero Electrónico). Universidad Industrial de Santander.
- [17]. L. Avendaño, G. Castellanos, R. Alzate and A. Morales. “Metodología para Implementación de Algoritmos sobre DSP orientados al proceso de Bioseñales”. Scientia et Technica. Vol 10. N° 24. pp 67-72. Mayo. 2004.
- [23]. LJUNG, Lennart. System Identification Toolbox: for use with MATLAB. User’s Guide. Massachussets: The MathWorks Incorporated, 1997. 274 p.
- [22]. MATLAB® (IDENT – System identification toolbox).

- [5]. M. Bouri, D. Thomasset. "Sliding Control of an Electropneumatic actuator using an integral switching surface". IEEE Transactions on Control Systems Technology. Vol 9. N 2. pp 368-375. 2002.
- [7]. M. Y. Ayad, M. Becherif, A. Miraoui. "Sliding mode control of voltage regulated hybrid DC source using fuel cell and supercapacitors for traction system". In: Proc IEEE-ISIE 2007, VIGO-Spain.
- [16]. R. Alzate and G. Castellanos. "Implementación para un DSP de Aplicaciones en Tiempo Real". Ingeniería y Competitividad. Vol 6. N° 1. pp 7-16. Septiembre. 2004.
- [13]. R. Ramos, D. Biel, E. Fossas, F. Guinjoan. "A Fixed-frequency Quasy-Sliding Control Algorithm: Application to Power Inverters by Means of FPGA Implementation". IEEE Transactions on Power Electronics. Vol 18. N° 1. pp 344—355. 2003.
- [4]. R. Solea (2010). "Sliding Mode Control Applied in Trajectory-tracking of WMRs and Autonomous Vehicles". Universidad de Coimbra. PhD Thesis.
- [12]. S. Pinzón, R. Jerez. Implementación del sistema de control de la cola de un helicóptero RC. Trabajo de grado. Universidad Industrial de Santander. 2010.
- [3]. V. Utkin (1993). "Sliding Mode Control Design Principles and Applications to Electric Drives". IEEE Transactions on Industrial Electronics (IEEE) 40 (1): 23–36.
- [21]. VERA ROZO, Edwin. "*Validación experimental del modelo de dos ejes de un avión a escala*". Bucaramanga, 2012, 86 p. Trabajo de grado (Ingeniero Electrónico). Universidad Industrial de Santander.
- [14]. Y. Liu, E. Meyer, X. Liu. "Recent Developments in Digital Control Strategies for DC-DC Switching Power Converters". IEEE Transactions on Power Electronics. Vol 24. N° 11. pp 2567—2577. 2009.

ANEXO A

A.1 Fundamentos del SMC

El control por modos deslizantes se puede definir como la aplicación de una señal de control que conmuta en alta frecuencia y consigue llevar el estado del sistema a un campo escalar $s(X)$, denominado superficie de deslizamiento. La superficie es definida por el diseñador con el propósito de que el estado cumpla las especificaciones deseadas. Las leyes de control para sistemas no-lineales presentan ciertas propiedades atractivas, entre ellas la robustez frente a ciertas incertidumbres de los parámetros y ciertas dinámicas no-modeladas, inmunes a perturbaciones y límites de conmutación dependientes de los estados. Esta última característica permite disminuir la magnitud de micro-oscilaciones (producidas debido a la conmutación de alta frecuencia de la señal de control) asociadas con regímenes deslizantes, mejorando así la regulación sobre la salida del sistema que se desea controlar. Uno de los aspectos más interesantes del modo deslizante es la naturaleza discontinua de la acción de control. Esta característica del sistema da como resultado un excelente rendimiento, ya que incluye baja sensibilidad a las variaciones de los parámetros y un rechazo a las perturbaciones, debido a la robustez.

Para realizar el análisis de esta técnica se considera un sistema no-lineal de la forma:

$$\dot{x}^{(n)} = f(X) + b(X)u \quad (A.1)$$

Donde el escalar x es la salida de interés, el escalar u es la entrada de control, X es el vector de estado $X = [x \ \dot{x} \ \dots \ x^{(n-1)}]^T$, n es la dimensión del sistema, $f(X)$ es la función que describe el sistema y $b(X)$ es la ganancia de control. La dinámica f no es conocida con exactitud, pero si su estimado \hat{f} . La estimación del error sobre f está limitada por una función conocida $F = F(x, \dot{x})$:

$$|\hat{f} - f| \leq F \quad (A.2)$$

El objetivo del control es conseguir que el estado X pueda realizar un seguimiento en los tiempos específicos en que varía el vector de estado dinámico $X_d = [x_d \dot{x}_d \dots x_d^{(n-1)}]^T$, es por esto que el vector de error de seguimiento puede definirse como:

$$\tilde{X} = X - X_d = [\tilde{x} \ \dot{\tilde{x}} \ \dots \ \tilde{x}^{(n-1)}]^T \quad (A.3)$$

La superficie de deslizamiento está definida en el espacio $R^{(n)}$ por la ecuación escalar $s(X) = 0$ donde:

$$s(X) = \left(\frac{d}{dt} + \lambda\right)^{(n-1)} \tilde{x} \quad (A.4)$$

Donde λ es una constante estrictamente positiva correspondiente al ancho de banda del sistema de control. Los límites sobre el escalar $s(X)$ pueden traducirse directamente en los límites del vector de error \tilde{X} , por tanto representan una verdadera medida del desempeño de seguimiento.

A.1.1 Existencia del modo deslizante

La superficie de deslizamiento se caracteriza por las siguientes condiciones de invarianza.

$$s(X) = 0, \quad \frac{ds(X)}{dt} = 0 \quad (A.5)$$

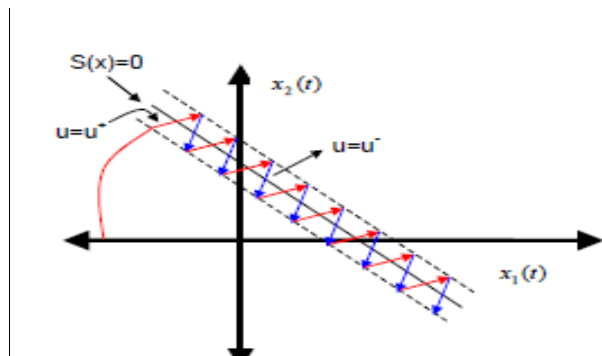
Estas condiciones expresan el hecho que, idealmente las trayectorias de estado están evolucionando sobre la superficie de deslizamiento. Cuando el sistema cumple estas condiciones su comportamiento se puede precisar a través de la dinámica ideal de deslizamiento [26]. Si el sistema está fuera de la superficie de deslizamiento; es decir cuando $s(X) \neq 0$, se aplicará una acción de control $u(t)$ tal

que: $\frac{ds(X)}{dt} < 0$ cuando $s(X) > 0$ y $\frac{ds(X)}{dt} > 0$ cuando $s(X) < 0$, con una forma discontinua dada por la siguiente expresión:

$$u = \begin{cases} u^+ & \text{para } s(X) > 0 \\ u^- & \text{para } s(X) < 0 \end{cases}$$

De esta manera el sistema se encuentra en modo deslizante sobre la superficie $s(X)$ si el control conmuta continuamente entre u^+ y u^- . La Figura 57 muestra un ejemplo de la conmutación cuando se ha introducido una banda de histéresis.

Figura 57. Región deslizante en el plano de fase y la acción de control



Fuente: [26]

A.1.2 Condición de deslizamiento

Lograr que el escalar $s(X)$ sea igual a cero, se puede obtener mediante la elección de la ley de control $u(t)$ en (A.1) de manera tal que fuera de la superficie de deslizamiento se cumpla [26]:

$$\frac{d}{dt} s^2 \leq -\eta |s| \quad (A.6)$$

Donde η es una constante estrictamente positiva que indica la fuerza de atracción de la trayectoria hacia la superficie de deslizamiento. La ecuación (A.6) indica que

una vez el sistema se encuentra sobre la superficie, las trayectorias del sistema permanecerán en ella. También implica tolerancia a perturbaciones o a incertidumbres dinámicas en el modelo.

A.2 Diseño del controlador por modos deslizantes

Para diseñar el SMC se requiere expresar el comportamiento dinámico de la planta (sistema de actuación en velocidad para un motor DC) en variables de estado. Este sistema presenta 2 modos principales de operación, dependiendo del nivel lógico (alto o bajo) aplicado a la entrada mediante la señal de PWM. Lo anterior se resume en modos de energización y des-energización, los cuales alternadamente en el tiempo significarán una condición de excitación regulada que permitirá actuar el sistema en el modo deseado.

Por tanto, retomando la ecuación (2.1) que representa el modelo del sistema obtenido y que se repite aquí en la ecuación (A.7),

$$H = \frac{\Omega(s)}{U(s)} = \frac{50000}{s^2 + 1010s + 4575} \quad (A.7)$$

se obtuvo una representación en el espacio de estados equivalente a través de la forma canónica controlable obtenida tras aplicar transformación de Laplace inversa, asumiendo condiciones iniciales iguales a cero. Como resultado se obtuvo la expresión (A.8) donde $x_1(t)$ representa la velocidad angular en el eje del motor al tiempo que $x_2(t)$ relaciona su derivada (i.e. su aceleración angular); mientras que $u(t)$ es una señal binaria correspondiente a una entrada de tipo PWM.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -4575 & -1010 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 50000 \end{bmatrix} u \quad (A.8)$$

Para formular la ley de control por modo deslizante, se debe expresar el sistema (A.8) en forma de la ecuación (A.1):

$$\ddot{x}_1 = -4575x_1 - 1010\dot{x}_1 + 50000u \quad (A.9)$$

A partir de lo cual:

$$b(X) = 50000 \quad (A.10)$$

$$f(X) \approx -1010\dot{x}_1 - 4575x_1 = -A_1\dot{x}_1 - A_2x_1 \quad (A.11)$$

Con lo cual se define a partir de (A.2), una incertidumbre paramétrica en $f(X)$ dada por:

$$\hat{f} = -p_1\dot{x}_1 - p_2x_1 \quad (A.12)$$

Donde p_1 y p_2 son valores estimados de parámetros que delimitan el límite de incertidumbre dado por F .

En la misma medida, es posible definir un grado de incertidumbre paramétrica multiplicativa en $b(X)$ mediante:

$$\hat{b} = (\sqrt{b_{min}b_{max}}) \quad (A.13)$$

Siendo b_{min} y b_{max} los valores extremos considerados para la función $b(X)$ y a partir de los cuales se define el parámetro de diseño:

$$\beta = \sqrt{\frac{b_{max}}{b_{min}}}$$

En este punto es importante indicar que para el estudio de las ecuaciones del control, se definió una incertidumbre en los parámetros del sistema con valores indicados en la Tabla 7.

Tabla 7. Valores de parámetro seleccionados para determinar las cotas de incertidumbre

Variable	Valor nominal	Valor mínimo	Valor máximo	Valor estimado
A_1	-1010	-700	-1010	-900
A_2	-4575	-4500	-4700	-4600
b	50000	49000	51000	N/A

Fuente: Autores

Luego, aplicando la ecuación (A.4) es posible obtener la siguiente superficie de deslizamiento para $n = 2$:

$$s = \dot{\tilde{x}}_1 + \lambda \tilde{x}_1 \quad (A.14)$$

Siendo $\tilde{x}_1 = x_1 - x_{1d}$ el error de seguimiento en la velocidad angular del motor, al tiempo que λ representa un parámetro de diseño que relaciona el ancho de banda de la acción de control. Por tanto, con base en los resultados sugeridos en la literatura, la mejor aproximación $\hat{u}(t)$ de la ley de control continuo $u(t)$ se obtiene a partir de la expresión (A.15).

$$\hat{u} = -\hat{f} + \ddot{x}_{1d} - \lambda \dot{\tilde{x}}_1 \quad (A.15)$$

Sin embargo, en favor de satisfacer la condición de deslizamiento (A.6), esta última expresión debe complementarse por un término de corrección para compensar la incertidumbre paramétrica \hat{f} , constituyendo una señal discontinua que ejerce influencia en sentido opuesto a la razón de cambio del campo vectorial, generando la expresión (A.16).

$$u = \frac{1}{\hat{b}} (\hat{u} - k * \text{sgn}(s)) \quad (A.16)$$

Siendo $sgn(s)$ la función signo y k un término de ganancia definido como:

$$k = (F + \beta\eta) + (\beta - 1)|\hat{u}| \quad (A.17)$$

Los demás elementos de la expresión han sido previamente definidos en el texto.

En resumen, a partir de la información obtenida del error de seguimiento en (A.14), es posible constituir la acción de control conmutado (robusto) dada por (A.16) en conjunto con (A.15) y (A.17), para forzar la convergencia de $s(X)$ hacia cero empleando las propiedades de convergencia determinadas por (A.6).

ANEXO B

B.1 Fundamentos del control óptimo

El problema de control óptimo en tiempo continuo consta de un sistema dinámico en la forma de la ecuación (B.1):

$$\frac{dX}{dt} = \dot{X}(t) = f(X, u) \quad (B.1)$$

La cual relaciona el vector de control $u(t)$ con el comportamiento dinámico del vector de estado $X(t)$. Para este tipo de control es necesario establecer una función objetivo J también llamada funcional de costo o índice de rendimiento del sistema, la cual depende de la trayectoria solución $X(t)$ y de la función de control $u(t)$. La función objetivo determina el comportamiento del sistema y debe maximizarse o minimizarse dependiendo del objetivo de control. En palabras simples en el control óptimo se establece la trayectoria óptima que maximiza un objetivo. La ecuación (B.2) muestra cómo se encuentra formulada una función objetivo.

$$J = \psi(X(T)) + \int_0^T l(X(t), u(t)) dt \quad (B.2)$$

Dicho funcional de costo se compone de dos partes principales: una contribución del estado final y otra acumulativa a través del tiempo. Para estudiar los efectos de las variaciones en el problema de control óptimo se emplea una función objetivo modificada conocida como Hamiltoniano $H(\lambda, X, u)$, la cual depende de un vector de multiplicadores λ asociado a variaciones del vector de estado $X(t)$ y a la determinación del vector de control $u(t)$:

$$H(\lambda, X, u) = \lambda^T f(X, u) + l(X, u) \quad (B.3)$$

En cada instante de tiempo, el valor óptimo de $u(t)$ tiene la propiedad de maximizar el Hamiltoniano. Este resultado es el principio del máximo de Pontryagin, cuyo teorema principal se resume a continuación:

Principio Máximo de Pontryagin:

Suponga que $u(t) \in U$ y $X(t)$ representan respectivamente la ley de control y la trayectoria de estado solución para el problema de control óptimo. Entonces, existe una trayectoria adjunta $\lambda(t)$ tal que, dados:

$$\dot{X}(t) = f(X(t), u(t))$$

$$X(0) = x_0$$

$$-\dot{\lambda}(t)^T = \lambda^T f_X(X, u) + l_X(X, u)$$

$$\lambda(T)^T = \psi_X(X(T))$$

Se cumple que:

$$H(\lambda(t), X(t), v) \leq H(\lambda(t), X(t), u(t))$$

Para todo t , $0 \leq t \leq T$, y para todo $v \in U$.

A partir de dicho teorema, encontrar el máximo valor de la función objetivo (B.2) equivale a seleccionar una señal $u(t)$ en el conjunto admisible de valores, en modo tal que se maximice el Hamiltoniano a partir de una trayectoria óptima y por tanto de una función de control óptima. Para ilustrar mejor la solución al problema de control óptimo, se considerará el caso de una estrategia de este tipo aplicada a la regulación de la velocidad angular de un motor DC.

B.2 Diseño del controlador óptimo

El objetivo del control consiste en mantener invariante el valor de velocidad angular a pesar de la incidencia de perturbaciones. Para efectos de diseño e ilustración, se conservará como valor de referencia deseado el empleado para el diseño del SMC, es decir 1500 RPM. El modelo del sistema en el espacio de estados obtenido en (A.8) se re-escibe en la ecuación (B.4) donde $x_1(t)$ y $x_2(t)$ constituyen las variables de estado siendo respectivamente la velocidad y aceleración angulares del sistema.

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = f(X(t), u(t)) = \begin{bmatrix} 0 & 1 \\ -4575 & -1010 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 50000 \end{bmatrix} u(t) \quad (B.4)$$

El problema de control óptimo es enunciado a partir del funcional de costo dado por la ecuación (B.5) correspondiente a la minimización de la energía del error, definida como la diferencia entre los valores deseado ω_{ref} y medido $x_1(t)$ para la velocidad del eje del motor (problema de regulación óptima). Las restricciones de la señal de control se incluyen en (B.6) y relacionan un ciclo útil (0% – 100%) ajustable en el dispositivo de conmutación, constituyendo un control óptimo conmutado:

$$\text{maximizar: } J = - \int_0^T (x_1(t) - \omega_{ref})^2 dt \quad (B.5)$$

$$\text{sujeto a: } 0 \leq u(t) \leq 1 \quad (B.6)$$

Procediendo con la formulación del problema de control, se obtendrá la expresión para el Hamiltoniano del sistema. A partir de la representación en espacio de estados dada en (B.4), se obtiene la expresión para la derivada del campo vectorial $f(X(t), u(t))$ respecto al vector de estados $X(t)$, a partir de la expresión (B.7):

$$f_X = f_X(X(t), u(t)) = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -4575 & -1010 \end{bmatrix} \quad (B.7)$$

Comparando las ecuaciones (B.2) y (B.5), se obtiene la igualdad (B.8):

$$\begin{aligned} \psi(X(T)) &= 0 \\ l(X, u) &= -(x_1(t) - \omega_{ref})^2 \end{aligned} \quad (B.8)$$

La derivada de esta última respecto al vector de estados $X(t)$ permite obtener la ecuación (B.9):

$$l_X(X, u) = \begin{bmatrix} \frac{\partial l}{\partial x_1} & \frac{\partial l}{\partial x_2} \end{bmatrix} = [-2x_1 + 2\omega_{ref} \quad 0] \quad (B.9)$$

Reemplazando los resultados de (B.7) y (B.9) en la expresión para la dinámica de $\lambda(t)$ dada por (B.10):

$$-\dot{\lambda}(t)^T = \lambda^T f_X(X, u) + l_X(X, u) \quad (B.10)$$

Se obtiene el conjunto de ecuaciones (B.11):

$$\begin{aligned} \dot{\lambda}_1(t) &= 4575\lambda_2(t) \\ \dot{\lambda}_2(t) &= -\lambda_1(t) + 1010\lambda_2(t) + 2x_1 - 2\omega_{ref} \end{aligned} \quad (B.11)$$

Desde el punto de vista dinámico, estas ecuaciones representan un sistema inestable visto como una relación no-homogénea de $\lambda(t)$ con término forzante dado por $2x_1 - 2\omega_{ref}$. Más aún, a partir de (B.8) es posible determinar las condiciones de frontera (valores finales) para $\lambda(t)$ dadas en (B.12):

$$\lambda(T)^T = \psi_X(X(T)) = \begin{bmatrix} \frac{\partial \psi(x(T))}{\partial x_1} & \frac{\partial \psi(x(T))}{\partial x_2} \end{bmatrix} = [0 \quad 0] \quad (B.12)$$

Ante este panorama de resultados analíticos, la única manera de obtener una expresión convergente para el Hamiltoniano (B.13), dado por la expresión

resultante (B.13), es que la condición final (B.12) para $\lambda(t)$ se satisfaga para todos los valores del tiempo:

$$H(\lambda, X, u) =$$

$$[\lambda_1(t) \ \lambda_2(t)] \left[-4575x_1 - 1010x_2 + 50000u(t) \right] - (x_1(t) - \omega_{ref})^2 \quad (B.13)$$

Es decir, siendo la dinámica de $\lambda(t)$ divergente a partir de (B.11), la única manera de satisfacer la convergencia de (B.13) es haciendo $\lambda(0) = \lambda(T) = \lambda(t) = 0$. Esta condición se pudo comprobar numéricamente empleando simulación numérica en MATLAB®. A partir de ello, la expresión reducida para el Hamiltoniano viene dada por (B.14):

$$H(\lambda, X, u) = l(X, u) = -(x_1(t) - \omega_{ref})^2 \quad (B.14)$$

Por tanto, el problema de control óptimo ha demostrado reducirse a la minimización del error de regulación (debido al principio del máximo que busca maximizar $H(\lambda, X, u)$).