

UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA

CARRERA DE INGENIERÍA ELECTRÓNICA

TRABAJO DE TITULACIÓN PREVIO A LA
OBTENCIÓN DEL TÍTULO DE INGENIERO
ELECTRÓNICO

PROYECTO TÉCNICO:

“DISEÑO E IMPLEMENTACIÓN DE UN CONTROL DIFUSO DE
VELOCIDAD PARA EL MOTOR HIDRÁULICO HRE HIDRAULIC”

AUTORES:

DANIEL EDUARDO CALLE CALLE
PABLO FERNANDO GUAMÁN NOVILLO

TUTOR:

ING. WALTER HUMBERTO OROZCO TUPACYUPANQUI, PhD.

CUENCA – ECUADOR

2018

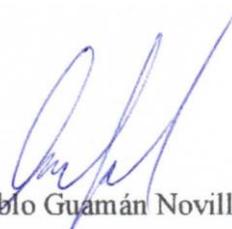
CESIÓN DE DERECHOS DE AUTOR

Nosotros, Daniel Eduardo Calle Calle con documento de identificación N° 0105709547 y Pablo Fernando Guamán Novillo con documento de identificación N° 0105401996, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación: **“DISEÑO E IMPLEMENTACIÓN DE UN CONTROL DIFUSO DE VELOCIDAD PARA EL MOTOR HIDRÁULICO HRE HIDRAULIC”**, mismo que ha sido desarrollado para optar por el título de: Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribimos este documento en el momento que hacemos entrega del trabajo final en formato impreso y digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, octubre del 2018

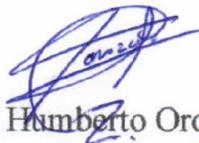

Daniel Calle Calle
CI: 0105709547


Pablo Guamán Novillo
CI: 0105401996

CERTIFICACIÓN

Yo declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **“DISEÑO E IMPLEMENTACIÓN DE UN CONTROL DIFUSO DE VELOCIDAD PARA EL MOTOR HIDRÁULICO HRE HIDRAULIC”**, realizado por Daniel Eduardo Calle Calle y Pablo Fernando Guamán Novillo, obteniendo el Proyecto Técnico que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana.

Cuenca, octubre del 2018



Ing. Walter Humberto Orozco Tupacyupanqui, PhD.

CI: 0102523974

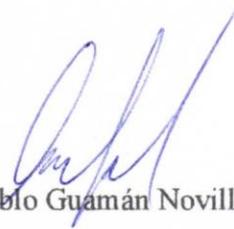
DECLARATORIA DE RESPONSABILIDAD

Nosotros, Daniel Eduardo Calle Calle con número de cédula CI. 0105709547 y Pablo Fernando Guamán Novillo con número de cédula CI. 0105401996, autores del trabajo de titulación: **“DISEÑO E IMPLEMENTACIÓN DE UN CONTROL DIFUSO DE VELOCIDAD PARA EL MOTOR HIDRÁULICO HRE HIDRAULIC”** certificamos que el total contenido del Proyecto Técnico, es de nuestra exclusiva responsabilidad y autoría.

Cuenca, octubre del 2018



Daniel Calle Calle
CI: 0105709547



Pablo Guamán Novillo
CI: 0105401996

AGRADECIMIENTOS

En primer lugar, quiero agradecer infinitamente a Dios por guiarme y bendecirme cada día de mi vida y durante toda mi trayectoria estudiantil.

A mis padres, por apoyarme incondicionalmente en todo momento, y por ser el pilar fundamental para mi realización profesional y personal.

A mi familia, por estar en todo momento junto a mí y brindarme todo su apoyo y atención. Gracias por confiar siempre en mí.

A mi director de tesis, muchísimas gracias por su tiempo y dedicación, y por su esmero por sacar adelante esta tesis.

A mis profesores, ya que gracias a su paciencia, dedicación y enseñanzas he podido obtener los conocimientos fundamentales para mi futura vida profesional.

A mis amigos, por brindarme su amistad y palabras de aliento que me han servido como apoyo moral durante mi carrera universitaria.

Gracias a todos.

Daniel Eduardo Calle Calle

A lo largo de mi vida y de mi carrera universitaria conocí a mucha gente que quiero agradecer por el apoyo que me han brindado. Primero, a toda mi familia, en especial a mis hermanos y a mis padres, Patricio y Carmita, mis cimientos que me empujaron para que no me rindiera.

A todo lo que conforma la Universidad Politécnica Salesiana, los docentes, gracias por el soporte y los conocimientos que me han entregado, en especial al PhD. Walter Orozco por su tutela y tiempo en este trabajo.

También a mis amigos, los buenos amigos que estuvieron cuando uno más los necesita, tanto en la vida universitaria como fuera de ella. A todos ustedes que están leyendo esto, de corazón, muchísimas gracias.

Pablo Fernando Guamán Novillo

DEDICATORIAS

Dedico este trabajo a Dios, por bendecirme y ayudarme en todos los momentos de mi vida, especialmente en los más difíciles.

A mis padres, por estar incondicionalmente en todo momento y por ayudarme a cumplir mis metas personales y estudiantiles.

A mi familia, porque han estado siempre a mi lado brindándome su apoyo.

A mis amigos con los que he compartido buenos momentos en mi vida universitaria.

Daniel Eduardo Calle Calle

Este trabajo va dedicado a mi padre y a mi madre; dos personas que no dejaron que me rindiera y por quienes en algún punto de mi vida me propuse terminar mi carrera. También a toda la familia que ha estado pendiente de mi desarrollo profesional.

A un gran amigo que tuvo que marcharse muy pronto, Lucho me hubiera gustado alcanzar esta meta a la par. Quisimos estudiar electrónica y sé que hubieras sido muy feliz al conseguirlo; solo espero que en donde estés puedas alegrarte por esto. Te recuerdo como fuiste, muy inteligente, pero sobre todo un gran ser humano.

Pablo Fernando Guamán Novillo

ÍNDICE GENERAL

AGRADECIMIENTOS.....	I
DEDICATORIAS.....	III
ÍNDICE GENERAL.....	IV
ÍNDICE DE FIGURAS.....	VIII
ÍNDICE DE TABLAS.....	XII
RESUMEN.....	XIII
INTRODUCCIÓN.....	XIV
CAPÍTULO 1: PROBLEMÁTICA.....	15
1.1 Antecedentes.....	15
1.2 Justificación.....	17
1.3 Objetivos.....	18
1.1.3 Objetivo General.....	18
1.2.3 Objetivos Específicos.....	18
1.4 Delimitación.....	19
1.5 Metodología.....	20
CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA.....	22
2.1 Estado del Arte.....	22
2.2 Motores Hidráulicos.....	29
2.1.2 Tipos de Motores Hidráulicos.....	30
2.3 Sistemas de Control Automático.....	35
2.4 Controladores.....	37
2.1.4 Sistemas de Control en Lazo Abierto.....	37
2.2.4 Sistemas de Control en Lazo Cerrado.....	38
2.3.4 Tipos de Control Automático.....	39
2.4.4 Discretización de Controladores.....	43
2.5 Lógica Difusa.....	45
2.1.5 Lógica Clásica.....	45
2.2.5 Conjuntos Clásicos.....	45
2.3.5 Operaciones con Conjuntos Clásicos.....	46
2.4.5 Inicios de la Lógica Difusa.....	47
2.5.5 Definiciones.....	47
2.6.5 Alcance.....	48

2.7.5 Enfoque	48
2.8.5 Universo de Discurso	49
2.9.5 Membresía en Lógica Difusa	49
2.10.5 Variable Lingüística	49
2.11.5 Proposiciones Difusas	50
2.12.5 Reglas Difusas.....	50
2.13.5 Conjuntos Difusos.....	51
2.14.5 Operaciones con Conjuntos Difusos	52
2.15.5 Funciones de Membresía o de Pertenencia	53
2.16.5 Inferencia Difusa.....	55
2.6 Control Difuso.....	61
2.1.6 Estructura de un Sistema Difuso SISO	61
2.2.6 Controlador Difuso	62
2.7 Identificación de Sistemas (Black Box Model)	63
2.8 Software	65
2.1.8 MATLAB®.....	65
2.2.8 LabVIEW®	67
2.9 Hardware	69
2.1.9 Arduino	69
2.2.9 MyDAQ	71
2.10 Banco HRE Hidraulic	74
2.1.10 Ventajas.....	75
2.2.10 Características Técnicas.....	76
2.3.10 Motor Hidráulico y Transductor	77
2.4.10 Vácula Distribuidora Porporcional 4/3	79
CAPÍTULO 3: DESARROLLO E IMPLEMENTACIÓN.....	81
3.1 Circuito de potencia	81
3.2 Estabilizador de señal.....	83
3.3 Implementación del sistema al Banco HRE Hidráulic.....	99
3.4 Desarrollo de Reglas Difusas	104
3.1.4 Control Proporcional Derivativo Difuso (CPDD)	105
3.2.4 Control Proporcional Integrativo Difuso (CPID).....	108
3.3.4 Control Difuso Proporcional Derivativo + Integrativo (CDPD+I).....	110
3.5 Desarrollo de VI en LabVIEW®	113
3.1.5 VI para el muestreo y promediado de señales	114
3.2.5 VI para el filtrado de señales	115
3.3.5 VI para el Control Clásico	117
3.4.5 VI para el Control Difuso PD, PI y PD+I.....	118
3.5.5 Frecuencia de Muestreo.....	120
CAPÍTULO 4: PRUEBAS DE FUNCIONAMIENTO, ANÁLISIS DE RESULTADOS, CONCLUSIONES Y RECOMENDACIONES	124

4.1 Pruebas de Funcionamiento	124
4.1.1 Diseño del Plan Piloto.....	124
4.2.1 Prueba de Funcionamiento del Controlador Difuso PD: Cambios de Referencia	125
4.3.1 Prueba de Funcionamiento del Controlador Difuso PI: Cambios de Referencia	126
4.4.1 Prueba de Funcionamiento del Controlador Difuso PI: Ingreso de Perturbaciones.....	126
4.5.1 Prueba de Funcionamiento del Controlador Difuso PD+I: Cambios de Referencia	127
4.6.1 Diseño de un Controlador Clásico	128
4.7.1 Comparación entre el Controlador Clásico y el Controlador difuso PD+I: Cambios de Referencia	132
4.8.1 Comparación entre el Controlador Clásico y el Controlador difuso PD+I: Ingreso de Perturbaciones	132
4.9.1 Comparación entre el Controlador Clásico y el Controlador difuso PD+I: Entrada Escalón en 100rpm	135
4.2 Análisis de Resultados	136
4.3 Conclusiones	138
4.4 Recomendaciones.....	139
REFERENCIAS BIBLIOGRÁFICAS	140
APÉNDICES.....	146
APÉNDICE A: CÓDIGO IMPLEMENTADO EN MATLAB® PARA GENERAR FUNCIONES DE MEMBRESÍA.	146
APÉNDICE B: CÓDIGO IMPLEMENTADO EN ARDUINO PARA TRANSFORMAR UNA SEÑAL ANALÓGICA A PWM.	147
APÉNDICE C: CÓDIGO IMPLEMENTADO EN MATLAB® PARA DETERMINAR EL VALOR RMS.	148
APÉNDICE D: CÓDIGO IMPLEMENTADO EN MATLAB® PARA OBTENER LA TRANSFORMADA RÁPIDA DE FOURIER DE UNA SEÑAL Y PARA EL DISEÑO DE FILTROS DIGITALES.....	149
APÉNDICE E: CÓDIGO IMPLEMENTADO EN ARDUINO PARA MEDIR LA VELOCIDAD DEL EJE DEL TRANSDUCTOR MEDIANTE EL SENSOR ÓPTICO <i>H21A1</i>	152
APÉNDICE F: DIMENSIONES DE DISEÑO DE LA CAJA PARA EL HARDWARE DEL SISTEMA EN CONJUNTO	154
APÉNDICE G: REGLAS IMPLEMENTADAS PARA LOS CONTROLES DIFUSOS	155

APÉNDICE H: CÓDIGO IMPLEMENTADO EN MATLAB® PARA LA ESTIMACIÓN DE LA
FUNCIÓN DE TRANSFERENCIA DEL SISTEMA EN CONJUNTO. 157

ÍNDICE DE FIGURAS

Figura 1. <i>Esquema del sistema para el control difuso de velocidad del motor hidráulico.</i>	20
Figura 2. <i>Motor de engranes externos.</i>	31
Figura 3. <i>Motor tipo gerotor de acople directo.</i>	31
Figura 4. <i>Motor tipo gerotor orbitante.</i>	32
Figura 5. <i>Motor tipo gerotor de paletas rotativas.</i>	32
Figura 6. <i>Motor de paletas.</i>	33
Figura 7. <i>Motor de pistones radiales.</i>	34
Figura 8. <i>Motor de pistones axiales, en línea.</i>	34
Figura 9. <i>Motor de pistones axiales, eje inclinado.</i>	35
Figura 10. <i>Sistema de control autónomo.</i>	36
Figura 11. <i>Diagrama de bloques de un sistema de control en lazo abierto.</i>	38
Figura 12. <i>Diagrama de bloques de un sistema de control en lazo cerrado.</i>	38
Figura 13. <i>Control Proporcional.</i>	39
Figura 14. <i>Control Integral.</i>	40
Figura 15. <i>Control Proporcional - Integral.</i>	41
Figura 16. <i>Parámetros del Control PID.</i>	42
Figura 17. <i>Estructura del Control PID.</i>	42
Figura 18. <i>Función de membresía para un conjunto clásico A.</i>	46
Figura 19. <i>Diagrama del límite definido en un conjunto clásico A.</i>	46
Figura 20. <i>Diagrama del límite de un conjunto difuso.</i>	51
Figura 21. <i>Función de mapeo.</i>	52
Figura 22. <i>Partes de una función de pertenencia.</i>	53
Figura 23. <i>Función de pertenencia triangular.</i>	54
Figura 24. <i>Función de pertenencia de distribución normal.</i>	54
Figura 25. <i>Función de pertenencia trapezoidal.</i>	55
Figura 26. <i>Función de pertenencia sigmoideal.</i>	55
Figura 27. <i>Método de Máxima Membresía</i>	59
Figura 28. <i>Método del Centroide</i>	59
Figura 29. <i>Método de promedio ponderado.</i>	60
Figura 30. <i>Método de membresía máxima promedio.</i>	60
Figura 31. <i>Componentes de un controlador difuso.</i>	61

Figura 32. <i>Esquema general de un controlador difuso.</i>	63
Figura 33. <i>Esquema del filtrado digital adaptativo.</i>	64
Figura 34. <i>Entorno de trabajo de MATLAB®.</i>	66
Figura 35. <i>Entorno del TOOLBOX Fuzzy Logic Designer</i>	66
Figura 36. <i>Entorno de Simulink.</i>	67
Figura 37. <i>Panel Frontal en LabVIEW®.</i>	68
Figura 38. <i>Diagrama de bloques.</i>	68
Figura 39. <i>Topología de una comunicación remota en LabVIEW®.</i>	69
Figura 40. <i>Entorno de programación Arduino.</i>	70
Figura 41. <i>Arduino UNO.</i>	70
Figura 42. <i>Aspecto físico de la placa Arduino MEGA 2560.</i>	71
Figura 43. <i>Dimensiones y peso de la tarjeta MyDAQ.</i>	72
Figura 44. <i>Arquitectura interna de la tarjeta MyDAQ.</i>	72
Figura 45. <i>De izquierda a derecha: Puertos de audio, de alimentación, de entradas/salidas analógicos y digitales.</i>	74
Figura 46. <i>Dimensiones y peso de los bancos HLA90 y HLA80.</i>	75
Figura 47. <i>Dimensiones y peso del banco HLA70.</i>	75
Figura 48. <i>Símbolo del Motor Hidráulico.</i>	78
Figura 49. <i>Conexión entre el Motor Hidráulico y el Transductor.</i>	79
Figura 50. <i>Símbolo de la Válvula distribuidora proporcional 4/3.</i>	79
Figura 51. <i>Aspecto físico de la Válvula distribuidora proporcional 4/3.</i>	80
Figura 52. <i>Aspecto físico del Módulo de potencia L298.</i>	81
Figura 53. <i>Esquema de la conexión entre el software, hardware y el sistema hidráulico.</i>	82
Figura 54. <i>Valores RMS obtenidos en el tiempo de operación corto.</i>	84
Figura 55. <i>Valores RMS obtenidos en tiempo de operación medio.</i>	84
Figura 56. <i>Valores RMS obtenidos en tiempo de operación largo.</i>	85
Figura 57. <i>Señal PWM configurada en el Módulo de potencia L298 sin carga.</i>	86
Figura 58. <i>Señal PWM configurada en el Módulo de potencia L298 con carga.</i>	86
Figura 59. <i>Señal producida por el Transductor en voltios.</i>	87
Figura 60. <i>Espectro de la señal ruidosa del Transductor.</i>	87
Figura 61. <i>Señal del Transductor filtrada con $F_c = 10\text{Hz}$.</i>	89
Figura 62. <i>Señal del Transductor filtrada con $F_c = 50\text{Hz}$.</i>	89

Figura 63. Filtrado de la señal del Transductor mediante el filtro Butterworth con $F_c = 10\text{Hz}$.	91
Figura 64. Filtrado de la señal del Transductor mediante el filtro Butterworth con $F_c = 50\text{Hz}$.	91
Figura 65. Filtrado de la señal del Transductor mediante el filtro Chebyshev Tipo 1.	91
Figura 66. Filtrado de la señal del Transductor mediante el filtro Chebyshev Tipo 2.	92
Figura 67. Filtrado de la señal del Transductor mediante el filtro Elíptico.	92
Figura 68. Aplicación del filtro Butterworth con $F_c = 10\text{Hz}$ y 50Hz .	93
Figura 69. Aplicación del filtro Chebyshev Tipo 1 con $F_c = 10\text{Hz}$ y 50Hz .	94
Figura 70. Aplicación del filtro Elíptico con $F_c = 10\text{Hz}$ y 50Hz .	94
Figura 71. Cálculo de la media móvil con $M1 = 0$ y $M2 = 5$ para $x[n]$.	96
Figura 72. Señal del Transductor promediada cada dos y tres muestras.	97
Figura 73. Señal del Transductor promediada cada cuatro muestras.	97
Figura 74. Comparación entre el proceso de filtrado y promediado de la señal del Transductor.	98
Figura 75. Herramientas utilizadas para la obtención de la constante K .	100
Figura 76. Sistema en conjunto implementado.	103
Figura 77. Distribución de la caja que contiene el hardware del sistema en conjunto.	103
Figura 78. Lazo de control implementado para el CPDD.	106
Figura 79. Funciones de membresía implementadas para el CPDD.	107
Figura 80. Superficie de control para el CPDD.	108
Figura 81. Lazo de control implementado para el CPID.	108
Figura 82. Funciones de membresía implementadas para el CPID.	109
Figura 83. Superficie de control para el CPID.	110
Figura 84. Lazo de control implementado para el CDPD+I.	110
Figura 85. Funciones de membresía implementadas para el CDPD+I.	112
Figura 86. Superficie de control para el CDPD+I.	113
Figura 87. Diagrama de bloques para el muestreo y promediado de señales.	114
Figura 88. Panel Frontal para el muestreo y promediado de señales.	114
Figura 89. Panel Frontal para el filtrado de señales.	116
Figura 90. Diagrama de bloques para el filtrado de señales.	116

Figura 91. <i>Panel Frontal utilizado para las pruebas de funcionamiento del Control Clásico.</i>	117
Figura 92. <i>Diagrama de bloques utilizado para las pruebas de funcionamiento del Control Clásico.</i>	117
Figura 93. <i>Panel Frontal utilizado para las pruebas de funcionamiento de los Controles Difusos.</i>	118
Figura 94. <i>Diagrama de bloques utilizado para las pruebas de funcionamiento de los Controles Difusos PD y PI.</i>	119
Figura 95. <i>Diagrama de bloques utilizado para las pruebas de funcionamiento del Control Difuso PD+I.</i>	119
Figura 96. <i>Respuesta del sistema en conjunto al aplicarse una señal escalón.</i>	121
Figura 97. <i>Prueba de funcionamiento del controlador PD con cambios de referencia.</i>	125
Figura 98. <i>Prueba de funcionamiento del controlador PI ante cambios de referencia.</i>	126
Figura 99. <i>Sondeo de funcionamiento del controlador PI ante ingreso de perturbaciones.</i>	127
Figura 100. <i>Prueba de funcionamiento del controlador PD+I con cambios de referencia.</i>	127
Figura 101. <i>Respuesta al escalón del sistema.</i>	128
Figura 102. <i>Comparación entre la respuesta real y estimada del sistema en conjunto.</i>	130
Figura 103. <i>Respuesta en lazo cerrado del control clásico diseñado.</i>	131
Figura 104. <i>Comparación entre el controlador clásico vs el difuso PD+I ante cambios de referencia.</i>	132
Figura 105. <i>Comparación entre el controlador clásico y el controlador difuso PD+I ante perturbaciones.</i>	133
Figura 106. <i>Comparación entre el controlador clásico y el controlador difuso PD+I en un escalón de 100rpm.</i>	135

ÍNDICE DE TABLAS

Tabla I. Operaciones en los conjuntos clásicos.....	46
Tabla II. Operaciones en conjuntos difusos	52
Tabla III. Lógica de la señal de control para la interfaz del motor uno.	82
Tabla IV. Tiempos de desfase entre la señal filtrada y sin filtrar.	89
Tabla V. Orden de los filtros digitales.	93
Tabla VI. Tiempos de desfase de los filtros digitales.	95
Tabla VII. Tiempos de desfase entre la señal promediada y sin promediar.	98
Tabla VIII. Valores medidos con el sensor óptico <i>H2IA1</i>	100
Tabla IX. Valores medidos con el tacómetro óptico.....	101
Tabla X. Determinación del error entre el valor real y medido en la velocidad del Motor Hidráulico.....	102
Tabla XI. Reglas difusas creadas para el CPDD.....	107
Tabla XII. Reglas difusas creadas para el CPID.	110
Tabla XIII. Reglas difusas creadas para el CDPD+I.	112
Tabla XIV. Presenta el plan piloto que se diseñó para las pruebas de los controladores.	124
Tabla XV. Resultados de las pruebas de estimación de la <i>FT</i>	129
Tabla XVI. Publica los resultados obtenidos de las pruebas de todos los controladores.	134
Tabla XVII. Valores obtenidos de los controladores difuso y clásico para una entrada de 100rpm.....	135

RESUMEN

El Sistema Hidráulico HRE HIDRAULIC presenta un comportamiento no lineal debido a diversos factores tanto operacionales como mecánicos, además a las propiedades físicas que presenta debido a su naturaleza. Esto ha impedido obtener un modelo matemático de alta aproximación para su funcionamiento y que posteriormente permita diseñar e implementar un control clásico que admita el manejo adecuado de su velocidad.

Es por este motivo que se optó por aprovechar las ventajas que ofrece un controlador difuso. Como punto de partida se desarrolló el estudio de arte para conocer los métodos y tipos de controladores difusos que se han implementado para abordar esta problemática. Seguidamente, se implementó un método, basado en el promedio móvil, para reducir el ruido de la señal que cierra el lazo de control.

Finalmente, se desarrolló un sistema mediante el software MATLAB® y el software LabVIEW® para diseñar e implementar el controlador difuso respectivamente. Los resultados que se obtuvieron son satisfactorios, el Motor Hidráulico responde eficazmente al sistema de control difuso desarrollado.

Palabras Clave: Control difuso, LabVIEW®, MATLAB®, Motor Hidráulico, Promedio móvil.

INTRODUCCIÓN

En las industrias, los Motores Hidráulicos han incrementado su frecuencia de uso, esto debido a la fuerza y eficiencia que éstos ofrecen. Implementar controladores clásicos a este tipo de motores causa inconvenientes debido a que presentan no linealidades, tales como: relación flujo-presión, cambio en la viscosidad debido a la temperatura del fluido, pérdida de presión, fuga de fluido dentro de los elementos hidráulicos y la influencia de otros parámetros durante su funcionamiento. Estas no linealidades complican la obtención del modelo matemático del motor y el diseño e implementación de un control clásico.

Es por esta razón que en este proyecto de tesis se ha desarrollado el Diseño e implementación de un control difuso de velocidad para el Motor Hidráulico HRE HIDRAULIC, como propuesta de un método de control para los Motores Hidráulicos en general. La ventaja de utilizar controladores difusos es que éstos no necesitan conocer el modelo matemático del sistema sobre el cual van actuar, sino que únicamente requieren conocer el comportamiento de dicho sistema. Este trabajo está estructurado de la siguiente manera:

En el primer capítulo se aborda la problemática, justificación, antecedentes y objetivos de este proyecto de tesis. En cuanto al segundo capítulo, se presenta el sustento teórico a cerca de los elementos utilizados para desarrollar este proyecto. Con respecto al tercer capítulo, se expone el proceso desarrollado para implementar el sistema que controla la velocidad del Motor Hidráulico. Finalmente, en el cuarto y último capítulo se indican y analizan los resultados obtenidos luego de llevar a cabo las respectivas pruebas de funcionamiento.

CAPÍTULO 1: PROBLEMÁTICA

1.1 ANTECEDENTES

Dentro del campo electrónico industrial se han implementado sistemas combinados que puedan ejecutar trabajos que demandan cargas elevadas como el caso de los sistemas electrohidráulicos. Aunque se ha podido cubrir dichas demandas, el nivel de complejidad de estos sistemas ha incrementado, dificultando el análisis e intervención por los métodos tradicionales de control. Considerando esto, uno de los principales problemas que un ingeniero llega a encontrar son casos en donde el control clásico puede resultar poco factible realizarlo, por ejemplo, debido a que la planta que modela al sistema presenta un comportamiento no lineal, en donde el control debe apegarse a esta no linealidad [1].

Los sistemas electrohidráulicos como el HRE HIDRAULIC presentan esta conducta, haciendo que la implementación de un control clásico se vuelva poco práctico. En base a esto, se ha visto la necesidad de optar por aplicar un control difuso con el objetivo de poder ajustar mejor el control de esta planta. También, el control difuso permite evitar análisis matemáticos complicados, en donde, además, no se conoce todos los parámetros que permiten aproximar el modelo, como sucede en el banco HRE HIDRAULIC [1].

En el proceso de investigación de este proyecto se han encontrado casos similares al trabajo realizado en [1]. El primero, refiérase a [2], corresponde a una investigación experimental realizada en un motor hidráulico, en donde, se obtienen los resultados de velocidades medidas sobre el eje de trabajo del motor en conjunto con una válvula proporcional. Los autores usan un PID difuso autoajutable usando como variables de entrada el error y la derivada del error. En este trabajo también se establece que la planta presenta un comportamiento no lineal, por lo que los autores, trabajan sobre un modelo lineal. La aplicación industrial de este proyecto está orientada al corte de metal.

El segundo trabajo estudiado, descrito en [3], trata sobre la capacidad que tienen los sistemas hidráulicos para manejar cargas elevadas y otras ventajas, además de ser muy usadas en industrias como el transporte y fabricación. En este trabajo, se desarrolla un controlador PID híbrido, combinando un control difuso con un convencional, con la meta

de aprovechar los beneficios de ambas partes. Para conmutar entre los dos tipos de controladores se utiliza un selector que depende del valor del error actual, el cual depende del valor de un umbral. Los resultados obtenidos, principalmente en error y tiempos, se midieron en el controlador híbrido y un controlador sin autoajuste difuso (refiérase a la sección 2.1) en donde el controlador híbrido muestra tiempos de respuesta más bajos.

En el último trabajo estudiado, ver [4], se puede encontrar una breve clasificación de los controladores difusos. En este documento se implementan dos diferentes clases de controladores difusos tipo PI. Uno de ellos se utiliza para establecer el motor hidráulico de forma lenta en el punto de operación deseado, en donde el sistema no presenta sobre elongación (sobre-salto). El segundo PI, tiene como objetivo establecer rápidamente el motor en el punto de operación deseado, pero, considerando que no haya demasiado sobre impulso en la respuesta. Al final, los resultados se comparan usando los dos controladores propuestos contra un controlador PID clásico. Se muestra que en cuanto a tiempos las respuestas no difieren, pero, la ventaja de los controladores difusos es que no requieren autoajuste a lo largo del tiempo.

1.2 JUSTIFICACIÓN

Con base a la experimentación realizada por los autores en [1], se encontró que el sistema HRE HIDRAULIC presenta un comportamiento no lineal y el control clásico implementado se regía solamente a un punto de operación de la planta. Es por eso que se propone implementar un control basado en lógica difusa como una propuesta para cubrir un rango más amplio de operación de la planta. Esta estrategia estaría orientada al control de velocidad del motor que comprende el sistema hidráulico (Válvula proporcional-Motor hidráulico-Transductor).

La importancia de este trabajo se desenvuelve en torno al controlador que se propone desarrollar sobre una planta de uso académico que dispone la Universidad Politécnica Salesiana, considerada dentro de los procesos industriales y que se podría encontrar en el campo laboral.

Este proyecto considera dos grupos beneficiarios. El primero es el grupo de Investigación en Interacción, Robótica y Automática (GIIRA) de la Universidad Politécnica Salesiana, y el segundo grupo son todos los estudiantes que estén cursando asignaturas de Control, Automatización, Instrumentación e Informática Industrial; se espera que con este proyecto los estudiantes complementen el conocimiento teórico adquirido en las aulas, ganando experiencia a través de la aplicación de dichos conocimientos en el manejo de sistemas reales.

1.3 OBJETIVOS

1.1.3 OBJETIVO GENERAL

- Diseñar e implementar un control difuso para el control de velocidad del motor hidráulico HRE HIDRAULIC.

1.2.3 OBJETIVOS ESPECÍFICOS

- Realizar un estudio del estado del arte de los métodos difusos aplicados para el control de velocidad de motores hidráulicos.
- Implementar un circuito de potencia adecuado para el funcionamiento del sistema.
- Desarrollar y efectuar un controlador difuso haciendo uso del software MATLAB y LabVIEW.
- Diseñar y llevar a cabo un plan piloto para la realización de las respectivas pruebas de funcionamiento del controlador.
- Recopilar los datos que arrojen las pruebas de funcionamiento para optimizar el controlador o acreditar su adecuado funcionamiento.

1.4 DELIMITACIÓN

El proyecto tiene como punto de partida el estudio en cuanto a motores hidráulicos y a continuación los métodos en lógica difusa implementados para el control de su velocidad. Posterior a esto, se concentrará en el diseño de un controlador de clase difusa junto con la implementación de circuito de potencia, para luego realizar pruebas de funcionamiento, las mismas que servirán como referencia para la optimización de dicho controlador o como evidencia del funcionamiento adecuado de éste.

1.5 METODOLOGÍA

En la Figura 1 se observa, de forma esquemática, el sistema que se va implementar para desarrollar el control difuso para el motor hidráulico.

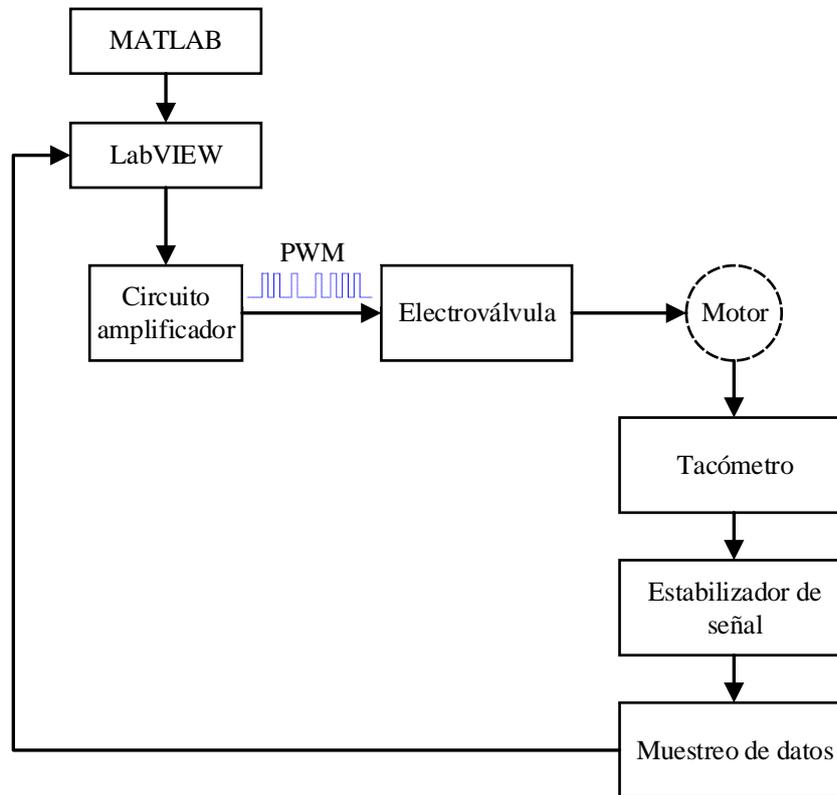


Figura 1. Esquema del sistema para el control difuso de velocidad del motor hidráulico.

A continuación, se desarrolla en qué consiste cada una de las partes:

- **MATLAB®:** en este software se llevará a cabo el desarrollo de la lógica difusa que poseerá el controlador mediante el uso del TOOLBOX FUZZY.
- **LabVIEW®:** en primera instancia en este software se desarrollará un VI (Instrumento Virtual), el cuál tomará tanto la lógica difusa, desarrollada en MATLAB®, como los datos del comportamiento del sistema hidráulico mediante una MyDAQ de la National Instruments. Posterior a esto el VI evaluará los datos adquiridos con la lógica difusa y emitirá una señal de control, la misma que será transmitida hacia la

electroválvula (que controla la velocidad del motor) pasando por un circuito de potencia.

- **Circuito de Potencia:** será el encargado de tomar la señal de control emitida por LabVIEW® y luego amplificarla de tal forma que la electroválvula se active.
- **Electroválvula-Motor hidráulico:** la electroválvula recibirá la señal del Circuito de Potencia y activará su solenoide provocando un desplazamiento positivo o negativo en su carrete, trayendo como consecuencia el aumento o decremento de caudal de fluido, el cual hará que la velocidad del motor hidráulico se eleve o disminuya.
- **Tacómetro (Transductor):** es el instrumento que medirá la velocidad del motor hidráulico y lo convertirá en niveles de tensión (0-10VDC).
- **Estabilizador de señal:** comprende de un circuito o método que estabilice la variación de la señal que proporciona el Transductor.
- **Muestreo de datos:** en esta etapa se hará uso de una MyDAQ para la recopilación de datos, los cuales serán enviados al software LabVIEW®.

CAPÍTULO 2: FUNDAMENTACIÓN TEÓRICA

2.1 ESTADO DEL ARTE

Uno de los objetivos de este proyecto es investigar y establecer los métodos usados para el control difuso para motores hidráulicos y sistemas electrohidráulicos que encierran esta clase de motores, tanto para el conocimiento personal como para posibles trabajos futuros propios y ajenos. A continuación, se desarrollan estos métodos.

En [2], los autores presentan los resultados obtenidos de manera experimental sobre la estabilidad de la velocidad medida sobre el eje de trabajo de un motor hidráulico que opera junto con una válvula proporcional, usando un controlador PID difuso autoajustable. Las principales variables establecidas en esta investigación son: el tiempo de respuesta, el sobre-salto (sobre elongación) y el error de rotación en estado estable.

El sistema con el que trabajaron los autores está constituido por: una válvula proporcional, motor hidráulico, correas de transmisión en V, eje de trabajo, correas de sincronización y el tacómetro.

Los autores recalcan que este tipo de sistemas caen dentro de lo que se conoce como transmisión automática y control de sistemas hidráulicos (ATACOHS, *por sus siglas en inglés*) aplicado a maquinaria, equipo y el desarrollo de tecnología dentro del campo industrial. En este caso la aplicación orientada es como cortador de metal.

Este trabajo menciona que existen muchas medidas a considerar dentro de los sistemas ATACOHS como, por ejemplo; la elasticidad del fluido, la relación entre flujo/presión, el cambio en la viscosidad debido a la temperatura del aceite, pérdida de presión en tuberías, fugas en elementos hidráulicos y otros parámetros que se presentan durante el funcionamiento del sistema. Todas estas cuantificaciones y sus variaciones se presentan como un proceso dinámico *no lineal*, por lo que los autores, para simplificar el proceso de diseño de control, consideran el sistema como uno lineal.

El modelo que los autores hallan se basa en la hipótesis del sistema lineal, en donde se incluyen algunos parámetros como la deformación elástica de la correa de

transmisión entre los ejes, la fricción y el momento de inercia del rotor del motor hidráulico.

También, en [2], se muestra que, en la actualidad el control de velocidad en motores hidráulicos tiene dos enfoques primordiales. El primero, es aquel en donde se cambia el modo de trabajo de la bomba y el motor hidráulico; esto se logra ya sea variando el volumen de la bomba y el motor o alterando la velocidad de la bomba a través de la variación en la velocidad del motor eléctrico trifásico utilizando un inversor. El segundo, consiste en cambiar la resistencia en la ruta del fluido usando una servo-válvula o una válvula proporcional con lo que se logra controlar la velocidad del motor. Comúnmente, este método es más utilizado debido a que el control de lazo cerrado se relaciona con la señal de entrada, la señal de salida y la retroalimentación al controlador, reduciendo los errores del sistema.

En cuanto al controlador, los autores indican que los parámetros K_P , K_I y K_D del PID clásico son fijados para un determinado punto de operación, lo cual no es efectivo en sistemas con parámetros variables que se presentan durante la marcha o por efecto de otros factores. Según esto y con el modelo aproximado, posteriormente se realiza un controlador PID difuso.

Para el diseño y ejecución del controlador difuso usaron un sintonizador difuso (Fuzzy tuner) de MATLAB® en donde se ajustan los parámetros del PID clásico, pero de forma difusa. Las variables de entrada utilizadas para el sintonizador fueron: el error (e) y la derivada del error en función del tiempo (de/dt). Como variables de salida del sintonizador se obtienen los valores de K_P , K_I y K_D difusos.

Las variables de entrada e y de/dt se dividieron en cinco conjuntos difusos: NB (Negative Big), N (Negative), Z (Zero), P (Positive) y PB (Positive Big), de acuerdo al comportamiento que presentaba el sistema. El universo de discurso para e es $[-50 \ 50]$, mientras que para de/dt $[-600 \ 600]$. Para obtener los valores óptimos de K_P , K_I y K_D , la función de implicación utilizada es Mamdani. Los conjuntos difusos para estas salidas son: S (Small), MS (Medium Small), M (Medium), MB (Medium Big) y B (Big) comprendidos en el rango de 0 a 1. Para la simulación del sistema y del controlador los autores utilizan Simulink de MATLAB®.

En la ejecución del experimento usaron un banco de pruebas construido en el Departamento de Ingeniería Mecánica de la Universidad de Ciencia y Tecnología de Danang, Vietnam. Este banco consta de 12 partes:

- Válvulas de seguridad y sobre flujo.
- Alimentación hidráulica con un motor trifásico.
- Amplificador.
- DAC.
- Sistema Generador de Carga.
- Eje de trabajo.
- Medidor de velocidad.
- Filtro de alta presión.
- Acumulador Hidráulico.
- Válvula proporcional.
- Motor hidráulico.
- Arduino (Uno R3).

La función del Arduino es producir una señal digital adecuada, que se entrega al DAC, el cual a su vez se conecta al amplificador para que el banco trabaje en los niveles de tensión aptos.

Finalmente, los resultados obtenidos por los autores en [2], se consiguieron a través de simulaciones y experimentaciones. Los resultados consistían en verificar, para el caso de la simulación, y medir, para el caso del ensayo, la velocidad de respuesta en rpm del eje de trabajo, para una entrada escalón inyectada al sistema (medida en rpm). Para una entrada escalón de 400rpm los autores obtuvieron los siguientes resultados:

- En simulación:
 - Velocidad Actual: 400.2 rpm
 - Tiempo de respuesta: 2.54 s
 - Sobre elongación: 0%
 - Error en estado estable: 3.75% para el límite superior y 2.75% para el límite inferior.

- En el ensayo:
 - Velocidad Actual: 400 ± 15 rpm
 - Tiempo de respuesta: 2.54 s
 - Sobre elongación: 0%
 - Error en estado estable: 3.75% para el límite superior y 2.75% para el límite inferior.

También se realizaron pruebas para 600, 800 y 1000 rpm, refiérase a [2].

Otro trabajo que encierra al control difuso y un sistema electrohidráulico es el que se presenta en [3]. En este estudio se habla sobre la capacidad de los circuitos hidráulicos para manejar fuerzas grandes, y también sobre algunas ventajas como, por ejemplo, precisión de respuesta, propiedades auto-lubricantes y error reducido en posicionamiento. Aunque este tipo de sistemas presentan algunas desventajas como: alto costo, fugas, inflexibilidad y respuestas no lineales que limitan el uso de accionamientos industriales. Sin embargo, son extensamente usados en industrias como el transporte y fabricación.

La planta, clasificada como Sistema Servo Electro-Hidráulico (SEHS, *en inglés*), se caracteriza por que toma tanto las ventajas de la gran potencia de salida de la parte hidráulica como la respuesta rápida de la parte eléctrica. No obstante, esta combinación genera muchas incertidumbres en el sistema, aumenta complejidad y orden, dificultando el control clásico, volviéndolo ineficaz cuando el sistema es perturbado por hechos desconocidos o porque el entorno circundante del sistema varía. La planta está constituida básicamente por tres partes:

- El motor hidráulico
- La servo-válvula
- La carga del sistema.

De forma general, el funcionamiento de este sistema es: un resorte de torsión (carga) es accionado por el motor hidráulico y la rotación del motor es controlada por una servo-válvula. Cuando en la entrada se suministra un nivel de voltaje más alto se produce un mayor caudal de fluido desde la servo-válvula hacia el motor, elevando su velocidad.

Considerando este funcionamiento, el SEHS queda constituido completamente por:

- La planta.
- Una válvula proporcional.
- La bomba (como suministro de presión).
- Un amplificador que recibe 5 voltios de entrada y genera una señal PWM de salida de 24V a 200Hz para la servo-válvula.
- Un codificador de eje óptico (encoder).
- Un DAC de 15 bits de resolución
- Una tarjeta DAQ NI 6221.
- Una PC con un sistema operativo Windows XP que ejecuta LabVIEW® versión 8.6 para el desarrollo de los controladores difuso y clásico.

En lo que se refiere al control en [3], se denota que el control difuso es robusto y adecuado para sistemas que presentan cambios en la dinámica o donde no se requiere información precisa. Dicho control es usado exitosamente, presentando un mejor rendimiento que un controlador PID convencional. Otro atributo del control difuso es que muestra un tiempo de subida reducido y una pequeña sobre elongación. Sin embargo, a favor del PID clásico, está su mejor capacidad de controlar y minimizar el error en estado estable. Debido a esto, los autores desarrollan un controlador híbrido entre las dos estructuras, con la meta de mejorar el rendimiento del controlador, aprovechando los beneficios de ambos arreglos. El controlador PID clásico se ejecuta con base a lo que ya se conoce; encontrar valores adecuados para K_P , K_I y K_D , al linealizar el modelo entorno a diferentes puntos de equilibrio, pero, este PID no puede controlar con precisión la velocidad del SEHS, por lo que se desarrolla el controlador difuso.

Al igual que el estudio anterior, esta investigación establece como variables lingüísticas de entrada del controlador difuso al error (e), y la derivada del error (de/dt), ambas en un universo de discurso comprendido entre -10 y 10. Los conjuntos difusos definidos para estas variables son: Negative Big (NB), Negative (N), Zero (Z), Positive (P) y Positive Big (PB). Para la variable de salida, la señal de control, se presentan los mismos conjuntos, pero, en un universo comprendido entre -5 y 5.

Tanto para las variables de entrada como para la variable de salida las funciones de pertenencia usadas para representar los conjuntos difusos son del tipo triangular. El

resultado difuso se halla a través de la inferencia difusa de Zadeh ($max - min$) y el valor final (defuzificación) se obtiene con el método del centro de gravedad.

Para realizar el cambio entre el controlador difuso y el clásico, los autores definen un interruptor que selecciona uno de ellos. La posición del interruptor se especifica según el valor del error y el valor del punto de ajuste. Si el error en la velocidad tiene un valor por arriba del umbral e_0 , el sistema híbrido selecciona el controlador difuso para corregir la velocidad a la referencia deseada. En el caso de que la velocidad esté por debajo del umbral e_0 o cerca de la referencia (por debajo), el sistema selecciona el PID clásico, mejorando la precisión conforme la velocidad establecida. El ajuste de las constantes K_P , K_I y K_D del controlador convencional se realiza a través de un sintonizador difuso, para mayor detalle refiérase a [3].

Al final, los resultados experimentales presentados por los autores muestran valores obtenidos para la velocidad (rpm), el error (rpm), sobre elongación (%), tiempo de rizo (s), tiempo muerto y el tiempo de establecimiento (s). Estas variables se midieron para el controlador híbrido propuesto y para un controlador híbrido sin autoajuste difuso para PID estándar.

Para la primera prueba el mejor resultado en cuanto tiempos presenta el controlador PID con autoajuste difuso, con un tiempo de establecimiento de 0.525 segundos, sin embargo, el error es ligeramente más alto. En la segunda prueba el resultado dominante corresponde al controlador híbrido; mostrando tiempos más bajos, pero, el error sigue siendo más alto, aunque no se separa demasiado de la referencia. Esto se debe a que el controlador propuesto no requiere reajustar los parámetros del controlador PID, aunque los parámetros del SEHS cambiarán de alguna manera.

En [4], Nayeri, Khosravi y Aliyari muestran una clasificación de los controladores difusos; la primera clasificación corresponde a controladores de acción tipo directa, la segunda a los controladores donde la ganancia difusa es programada y la tercera los controladores híbridos. En este estudio se diseñan e implementan dos diferentes controladores difusos tipo PI independientes del modelo del sistema. El primero, clasificado como de acción tipo directa, consta de dos entradas difusas. El segundo, corresponde a un controlador programado con ganancia difusa, el cual está diseñado para trabajar en una tasa de respuesta máxima del sistema.

La planta hidráulica está compuesta por cinco elementos:

- La unidad de potencia hidráulica.
- Válvula reguladora de presión.
- Servo-válvula.
- Sensor de velocidad angular.
- Tarjeta de entrada/salida (E/S).

El ingreso del sistema es la señal de control generada por el controlador. Ésta determina la velocidad de apertura de la servo-válvula. La salida del sistema es la velocidad del motor medida en rpm a través del sensor de velocidad angular y se recibe al controlador como realimentación. Tanto la señal de entrada como de salida se transmiten a la planta a través de la tarjeta de entrada/salida.

La meta del controlador difuso PI de doble entrada es establecer la velocidad del motor hidráulico paulatinamente en el punto de operación para que no exceda en la respuesta del sistema y la respuesta del motor no cambie debido a la modificación en el punto de trabajo. Para el diseño del controlador se encuentra el rango de las variables de entrada y de salida. Las variables de ingreso son el error definido en $[-900\ 900]$ y su derivada determinado en $[-2000\ 2000]$, mientras que la de salida es la señal de control definida en $[-60\ 60]$. Las funciones de membresía usadas son triangulares tanto para las entradas como para la salida; estas permiten representar cinco conjuntos difusos para el error, siete conjuntos para su derivada y siete para la señal de salida. El sistema de inferencia empleado es Mamdani, se usa el fuzificador de singleton y como defuzificador el método de promedio central.

Por otro lado, el controlador PI con ganancia difusa programada tiene como objetivo establecer rápidamente la velocidad del motor en el punto de operación deseado, pero, considerando que no exista demasiado sobre-impulso en la respuesta. El diseño de este controlador se sujeta a conocer un controlador PI que funcione adecuadamente en un punto de operación. Al establecer el controlador PI, se toma las constantes K_P y K_I y se crean funciones de membresía entorno a estos valores.

Similar al controlador anterior, las funciones usadas son triangulares las cuales representan siete conjuntos difusos tanto para el caso de K_P como de K_I , que corresponden

a las salidas del controlador. De igual manera, se usa la inferencia de Mamdani, el fuzificador singleton y el defuzificador del promedio central.

Los resultados presentados en [4], se realizan al comparar tres controladores PI. Dos de ellos corresponden a los controladores desarrollados por los autores y el tercero a un controlador clásico de Chein-Hrones Roswick. Las deducciones muestran que el controlador PI difuso de doble entrada es más lento. Su señal de control no llega a saturarse y es idóneo para casos en que el motor requiere establecerse de forma pausada. En el caso del PI difuso programado, presenta una respuesta mucho más rápida que el compensador anterior, pero, el costo de esta repuesta acelerada produce un sobre impulso, saturando la señal de control para momentos iniciales en los valores máximos. Para el caso del PI clásico se observa un sobre impulso más grande (saturación) que el PI difuso programado, pero, desde el punto de vista de velocidad de respuesta es una ligera ventaja. Finalmente, los autores realzan que la velocidad de respuesta entre los controladores difuso y clásico no difieren mucho, sin embargo, se debe tomar en consideración que los parámetros del controlador clásico de deben reajustar con el tiempo, mientras que los controlares difusos no requieren este reajuste, ya que, cambiarán gradualmente según los componentes del sistema, obteniendo una respuesta óptima.

2.2 MOTORES HIDRÁULICOS

Los motores hidráulicos son sistemas que obtienen energía mecánica rotativa a partir de energía cinética de un fluido. Esta energía mecánica rotativa es transmitida a una carga mediante un eje. En cuanto a características de diseño, los motores hidráulicos, de manera general, cuentan con las siguientes [5]:

- Manejo de fuerzas y presiones internas.
- Características de fuga.
- Caudal.
- Torque de salida.
- Configuración física.
- Vida útil.

Cuando se trabaja con motores hidráulicos se maneja una extensa terminología, a continuación, se presenta algunos de los términos más utilizados [5]:

Desplazamiento: es el volumen de fluido necesario para hacer girar el eje una revolución; puede ser fijo (torque constante) o variable.

Par de arranque: se refiere a la cantidad de torque necesario para iniciar el movimiento de una carga en reposo. El torque para iniciar el movimiento de la carga es mayor al torque necesario para mantenerla en movimiento.

Torque de arranque: es el torque que un motor puede desarrollar para mover una carga desde el reposo. Se expresa en forma porcentual con respecto al torque teórico (aproximadamente entre el 70% y 80% para motores hidráulicos convencionales del tipo engranajes, paletas y pistones comunes).

Torque de operación: es un porcentaje del torque teórico (aproximadamente el 90% para motores convencionales de paletas, de engranes y de pistón). Toma en consideración la ineficiencia del motor y se define como la carga aplicada al motor o la carga del motor mismo.

Eficiencia mecánica: hace referencia a la relación que existe entre el torque real y el teórico.

Velocidad máxima: es la velocidad rotacional que el motor hidráulico (a una presión determinada en su entrada) puede generar durante un tiempo determinado sin sufrir daños en él.

Velocidad mínima: es la velocidad giratoria más detenida, progresiva y sin interrupción que el motor puede generar.

2.1.2 TIPOS DE MOTORES HIDRÁULICOS

2.1.2.1 Motores de Engranajes

Estos motores pueden ser del tipo engranaje externo o interno [5]:

Motor de engranajes externos: está constituido por dos engranes ajustados entre sí dentro de una carcasa, Figura 2. Estos engranes tienen la misma cantidad y forma de dientes y son movidos mediante un fluido hidráulico. Uno de los engranes está conectado al eje que transmitirá movimiento hacia el exterior, mientras que el otro engrane gira libremente. El fluido hidráulico ingresa a presión

por un lado de la carcasa provocando que los dientes de los engranes se acoplen entre sí, produciendo de esta manera un movimiento rotatorio. Luego, el fluido avanza por la periferia de la carcasa (ruta de menor resistencia) y sale a baja presión por el lado opuesto al de ingreso.

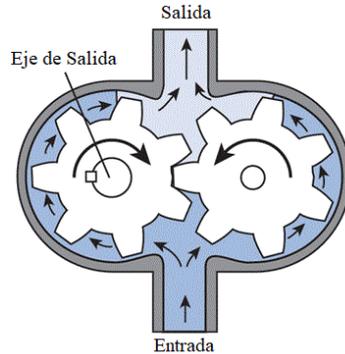


Figura 2. Motor de engranes externos.
Fuente: [5]

Motores de engranaje interno: pueden ser [5]:

Motor tipo gerotor de acople directo: está conformado por un conjunto de engranes interno-externo y un eje, Figura 3. El engrane interno posee un diente menos que el externo; además, los dientes de los engranes externo e interno embonan de forma aproximada.

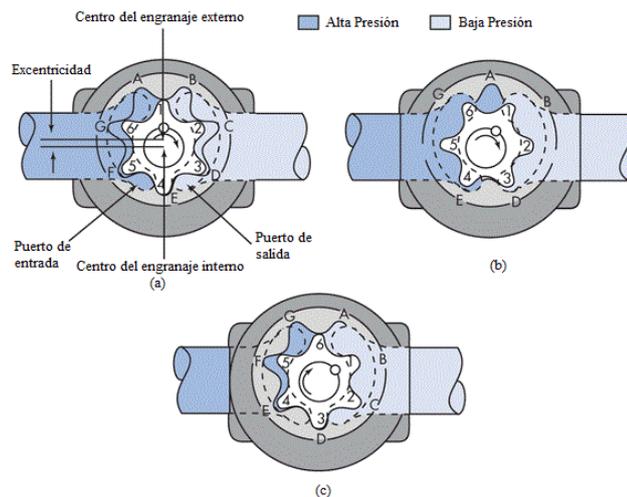


Figura 3. Motor tipo gerotor de acople directo.
Fuente: [5]

Motor tipo gerotor orbitante: está compuesto por un conjunto de engranes (interno-externo), un acople, un eje y un conmutador, como se muestra en la Figura 4. El engrane externo es fijo y además posee un diente más que el engrane

interno. En cuanto al conmutador, éste rota a la misma velocidad que el engrane interno y continuamente distribuye fluido a presión a través de cavidades. Estas cavidades de presión y retorno para cada diente del engrane interno son bloqueadas o proveen fluido a presión durante el movimiento rotacional.

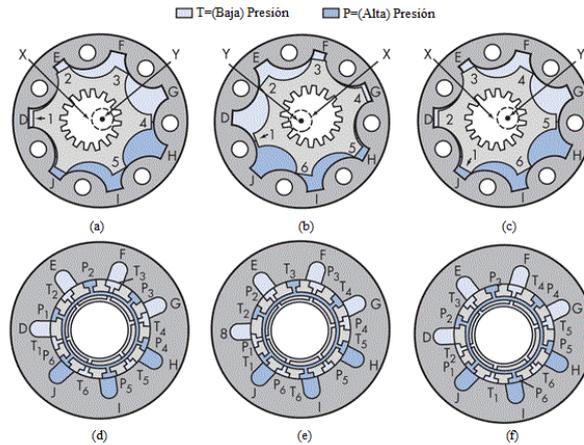


Figura 4. Motor tipo gerotor orbitante.

Fuente: [5]

Motor gerotor de paletas rotativas: su funcionamiento es similar al del motor gerotor orbitante, sin embargo, mecánicamente su diseño interno es diferente, esto con el fin de reducir el desgaste por contacto, Figura 5. En esta versión se incluye paletas rotativas para formar las cámaras de desplazamiento.

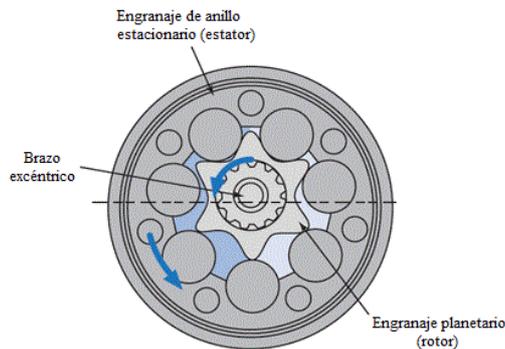


Figura 5. Motor tipo gerotor de paletas rotativas.

Fuente: [5]

2.2.2.1 Motores de Paletas

Estos motores están compuestos por un rotor (con agujeros) que está acoplado a un eje, Figura 6. En las cavidades del rotor se encuentran instaladas paletas que se trasladan radialmente. El movimiento se produce cuando el fluido a presión ingresa a la carcasa del motor (mediante aperturas de ingreso) y mueve al rotor a

través de las paletas, luego el rotor conduce al fluido hacia aperturas de salida con el fin del que el fluido regrese al depósito (tanque).

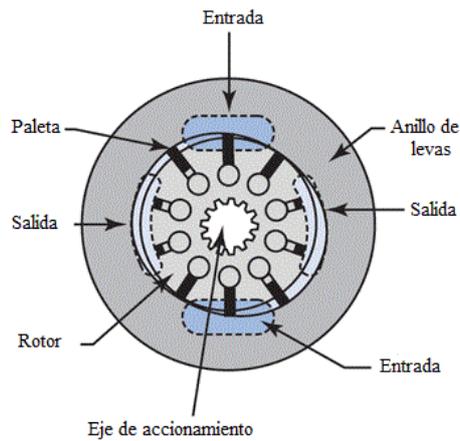


Figura 6. *Motor de paletas.*
Fuente: [5]

2.3.2.1 Motores de Pistones

Pueden ser de 4 tipos [5]:

Motores de pistones radiales: posee un cilindro que está acoplado a un eje, Figura 7. En este eje se encuentran instalados pistones, los cuales (en su extremo externo) se apoyan sobre un anillo impulsor. Estos pistones se mueven en forma recíproca (movimiento hacia arriba y hacia abajo) en cavidades radiales, mediante el fluido a presión que ingresa a través de un pin localizado en el centro del cilindro; los pistones son presionados contra el anillo impulsor y las fuerzas reactivas generan un movimiento rotacional.

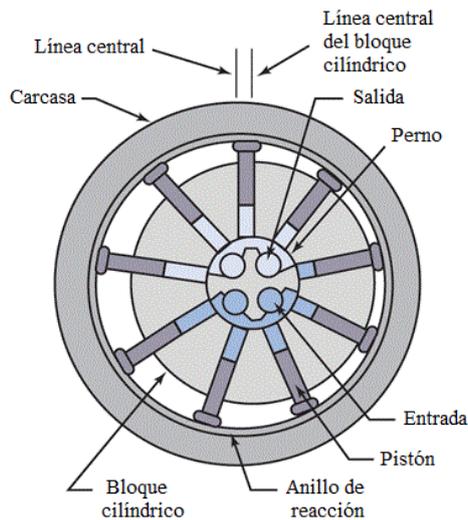


Figura 7. Motor de pistones radiales.
Fuente: [5]

Motores de pistones axiales: el movimiento recíproco también está presente en este motor para accionar el eje principal de forma axial, más no radial. Sus características de eficiencia, vida útil y costos son equivalentes a las de los motores de pistones radiales. Contrario al motor de pistones radiales, este motor presenta excelentes características a altas velocidades, mientras que en bajas velocidades presenta restricciones.

Motores de pistones axiales en línea: el diseño mecánico de este motor cambia con respecto a los anteriormente expuestos, el eje del motor y el cilindro están alineados. El torque se produce debido a una presión que es aplicada en los extremos de los pistones que reciprocán en el cilindro, Figura 8. Esta presión en los pistones produce una reacción en un basculante inclinado, generando así una rotación tanto en el cilindro como en el eje.

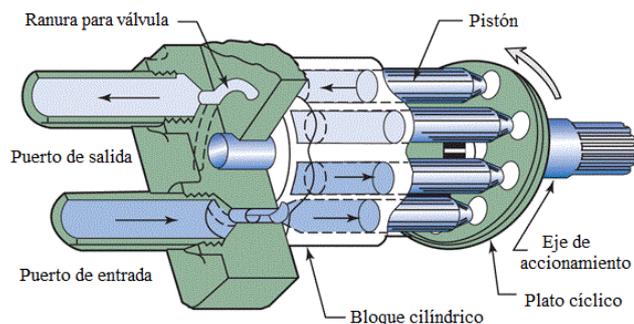


Figura 8. Motor de pistones axiales, en línea.
Fuente: [5]

Otro factor del cual depende el torque son los pistones, éste es proporcional a su área. Además, estos motores pueden ser de desplazamiento fijo o variable.

Motores de pistones axiales de eje inclinado: son motores que producen un torque gracias a la acción de la presión sobre pistones con movimiento recíprocante. El diseño de este motor cambia en cuanto a la posición del cilindro y el eje, Figura 9, éstos se encuentran acoplados a un ángulo. Las variaciones de este ángulo modifican tanto a la velocidad como al torque. Además, también estos motores pueden ser de desplazamiento fijo o variable.

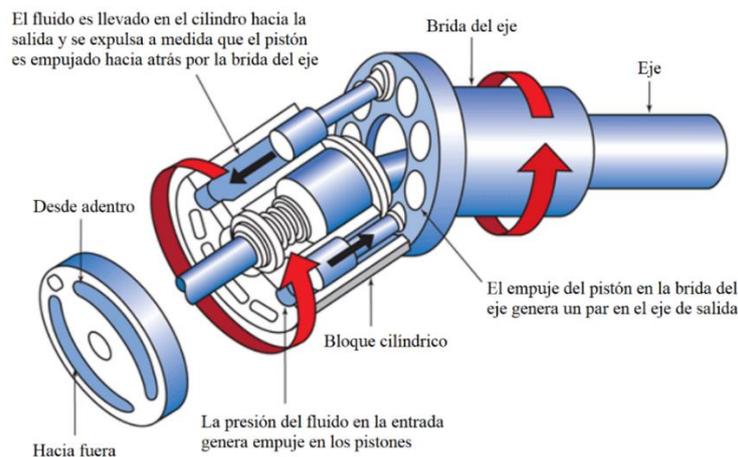


Figura 9. Motor de pistones axiales, eje inclinado.
Fuente: [5]

2.3 SISTEMAS DE CONTROL AUTOMÁTICO

Desde hace tiempo los sistemas de control automático han venido desempeñando un papel importante en todo aquello que mantiene una relación con los avances tecnológicos (robótica, fabricación, operación industrial, etc.) [6]. Con respecto a procesos industriales, los sistemas de control automático son muy importantes debido a que manejan los procesos de producción. Además, estos sistemas han aportado positivamente a la industria debido a que han aumentado considerablemente su productividad [6].

Un sistema de control automático es aquel que recepta las variables de proceso provenientes de la magnitud física a través de instrumentos (sensores, transmisores, etc.), las transforma y procesa para posteriormente ejecutar acciones de control autónomas sobre los elementos finales [7].

Se puede dividir en dos partes [8]:

- **Máquina:** se subdivide en planta y dispositivo de realimentación (sensor/medidor). La planta recibe dos tipos de señales: la de control y la de perturbaciones. En cuanto a la realimentación, ésta es necesaria en plantas cuyo funcionamiento es difícilmente controlable.
- **Controlador:** tiene como objetivo llevar a la operación de la planta a estados deseados, superando perturbaciones internas o externas. Monitorea las señales de salida de la planta para que se comporten de una manera deseada mediante la manipulación de sus señales de entrada.

En la Figura 10 se encuentran estas partes mencionadas.

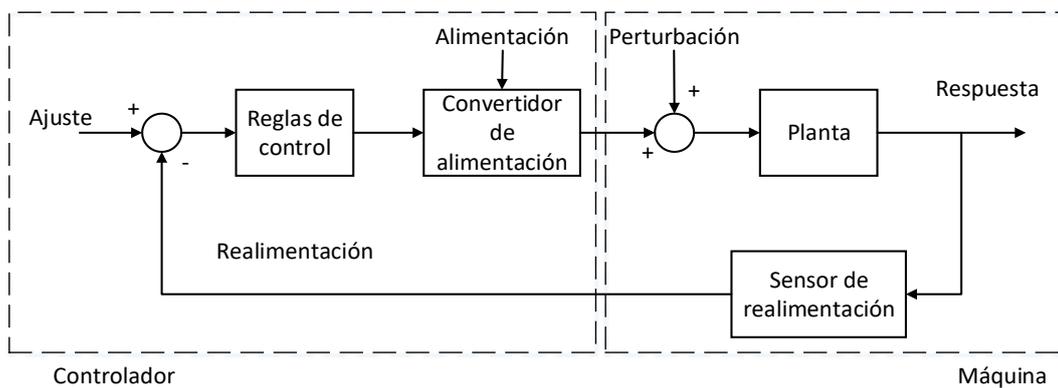


Figura 10. Sistema de control autónomo.

Algunos de los términos que generalmente se manejan en estos sistemas son [9]:

- **Variable controlada:** son variables con las que se mide o controla el proceso y cuyo comportamiento se observa en la salida.
- **Variable manipulada:** son las variables del proceso sobre las cuales se puede actuar, causando efecto en las variables controladas.
- **Planta:** es el conjunto de elementos que constituyen y operan en una máquina o sistema, cuyo objetivo es cumplir una determinada función.
- **Proceso:** ejecución de pasos progresivos que conducen a un resultado luego de pasar por múltiples o pocos cambios.
- **Sistemas:** son la unión de componentes que operan juntos y cumplen un determinado objetivo.

- **Perturbaciones:** son variables sobre las cuales no se puede actuar y que influyen en las variables controladas. Pueden ser internas o externas a un sistema.
- **Control realimentado:** es una operación (que se lleva a cabo cuando hay perturbaciones) que reduce la diferencia existente entre la salida del sistema y una entrada de referencia.

2.4 CONTROLADORES

Tiempo atrás, la modificación del comportamiento de procesos industriales se desarrollaba de forma manual, por ejemplo: se conectaba y desconectaba actuadores hasta lograr un estado deseado. Estas acciones hacían que la estabilidad de un sistema sea subjetiva; además, la calidad del producto final se veía comprometida. Es debido a estos y otros sucesos que se optó por la creación y uso de controladores automáticos [9].

Un controlador cumple la funcionalidad de tomar el valor real de salida de una planta junto con el de una entrada de referencia y compararlas. Luego de llevar a cabo dicha comparación el controlador determina una desviación para posteriormente producir una señal de control (acción de control) que disminuya a cero o a un valor pequeño tal desviación [9].

Los términos importantes que son necesarios conocer cuando se trabaja con controladores son [7]:

- **Variable de proceso:** es la variable sobre la cual se actúa con el objetivo de estabilizarla.
- **Referencia o Setpoint:** es el valor que se espera que tome la variable de proceso.
- **Error:** es la diferencia que existe entre la variable de proceso y la referencia. Ocasionalmente se mide en forma porcentual.

2.1.4 SISTEMAS DE CONTROL EN LAZO ABIERTO

Es un sistema que no mide la salida ni se realimenta para comparar con la entrada [8]. En la Figura 11 se observa su estructura. En el ámbito práctico, este lazo de control es usado únicamente cuando se conoce la relación existente entre la entrada y la salida y las perturbaciones (internas y externas) son nulas [6].

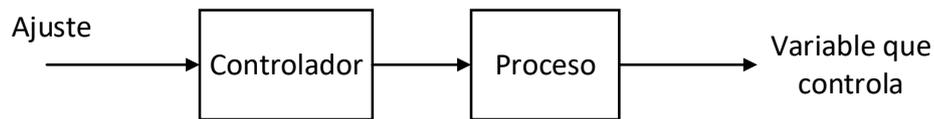


Figura 11. Diagrama de bloques de un sistema de control en lazo abierto.

En cuanto a ventajas y desventajas de este sistema se tienen las siguientes [10]:

- Está expuesto a salidas no deseadas debido a perturbaciones.
- Utiliza menos componentes que el sistema de control en lazo cerrado, por lo tanto, es menos costosa su implementación.
- Puede ser implementado en sistemas en los cuales sus entradas pueden ser anticipadas y donde no causan efecto las perturbaciones.
- Su construcción es más simple.
- No tiene problemas de estabilidad.
- Para mantener un correcto funcionamiento es necesario llevar a cabo una calibración de vez en cuando.

2.2.4 SISTEMAS DE CONTROL EN LAZO CERRADO

A diferencia del sistema de control en lazo abierto este sistema si realimenta su señal de salida, esto con el fin de compararla con una señal de referencia y determinar un error, para luego llevar a cabo una acción correctiva que reduzca dicho error [6], permitiendo de esta manera llevar a la salida a un valor deseado [8]. En la Figura 12 se aprecia como la señal de salida es realimentada al nodo de comparación. Esta realimentación permite llevar a cabo una acción de control, mejorando o manteniendo la estabilidad del sistema.

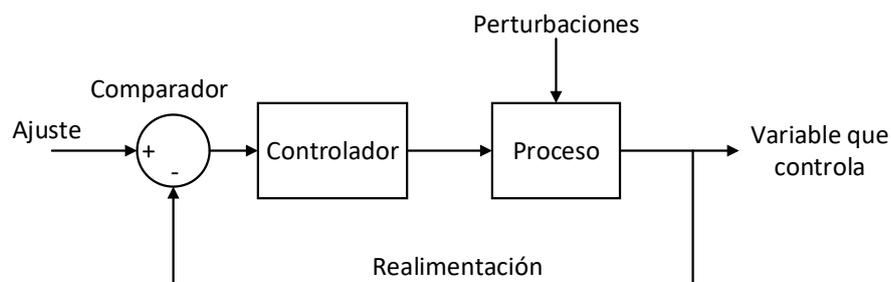


Figura 12. Diagrama de bloques de un sistema de control en lazo cerrado.

Sus ventajas y desventajas son [10]:

- Al poseer una realimentación este tipo de sistema es relativamente insensible a perturbaciones internas o externas.
- En cuanto a implementación, es más costoso que el sistema de control en lazo abierto.
- En la práctica, es uno de los sistemas más usados.

2.3.4 TIPOS DE CONTROL AUTOMÁTICO

Existen varios tipos de controladores, dependiendo de la aplicación será necesario un tipo de controlador. Además, un controlador es elegido dependiendo de la rapidez con la que se desea estabilizar un proceso o de la robustez del mismo [7].

Entre los controles más conocidos se tienen [9]:

- Control ON-OFF.
- Control Proporcional.
- Control Integral.
- Control Derivativo.
- Control Proporcional Integrativo (PI).
- Control Proporcional Derivativo (PD).
- Control Proporcional Integral Derivativo (PID).

A continuación, se desarrolla sobre el control PID y sus variantes.

2.1.4.3 Control Proporcional (P)

Genera en su salida una señal de control ($m(t)$), la cual es proporcional a la señal de error ($e(t)$) [9]:

$$m(t) = k_p * e(t) \quad (1)$$

En la Figura 13 se observa como la señal de error es modificada por k_p luego de pasar por el bloque de acción proporcional.

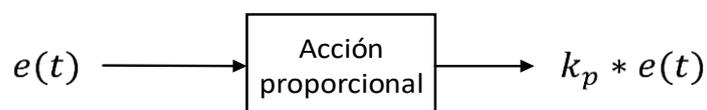


Figura 13. Control Proporcional.

Mientras más grande sea la ganancia proporcional menor será el error, y viceversa [9].

2.2.4.3 Control Integral (I)

Genera una señal de control ($m(t)$) que es proporcional a la integral de la señal de error ($e(\tau)$):

$$m(t) = k_i \int_0^t e(\tau) d\tau \quad (2)$$

Este control entrega una señal correctiva en base a la integración de la señal de error, es decir, la señal correctiva está en función del error. De esta manera se obtiene una señal de control diferente de cero para cualquier valor de la señal de error, incluso cuando ésta tome el valor de cero, Figura 14 [9].

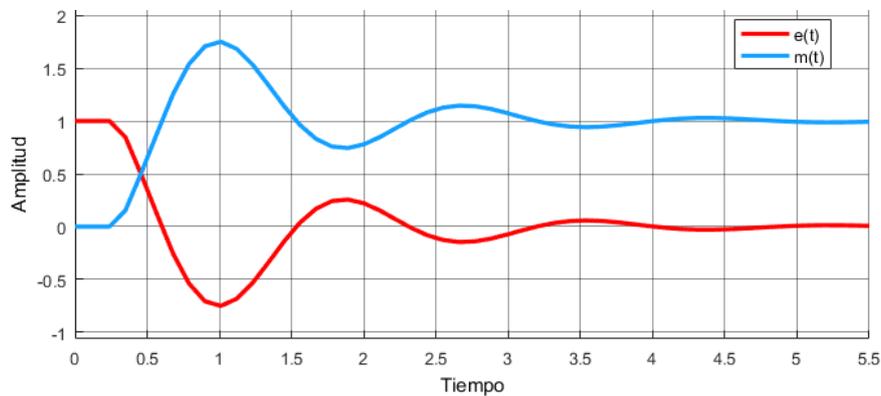


Figura 14. Control Integral.

2.3.4.3 Control Derivativo (D)

Genera una señal de control ($m(t)$) que es proporcional a la derivada de la señal de error ($e(t)$).

$$m(t) = k_d \frac{d}{dt} e(t) \quad (3)$$

Este control trabaja con la derivada de la señal de error y a la vez con las características dinámicas del sistema, modificando la señal de error previo a que ésta aumente su amplitud, es decir, trabaja de forma anticipada. Generalmente, este control trabaja en conjunto con el resto de controles debido a su insuficiencia frente a señales de error constantes [9].

2.4.4.3 Control PI

Produce una señal resultante de la combinación de los controles proporcional e integral. El control integral se encarga de reducir o eliminar el error estacionario, mientras que el control proporcional disminuye el riesgo de inestabilidad que se produce debido al control integral. Habitualmente, si la ganancia proporcional (k_p) e integral (k_i) son pequeñas la señal de control es más estable y la señal de error tiende a cero. En cambio, la señal de control se ve afectada (inestable) si la ganancia integral es mayor, obteniendo así un error diferente de cero [9]. Este comportamiento de la señal de control ($m(t)$) se contempla en la Figura 15.

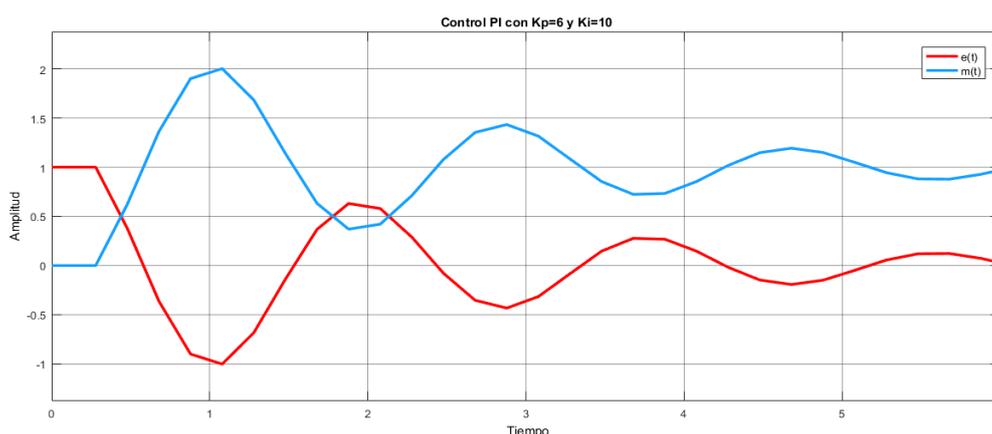


Figura 15. Control Proporcional - Integral.

2.5.4.3 Control PD

Genera una señal resultante de la combinación del control proporcional y derivativo [9].

El control derivativo crea una relación del tipo lineal entre la velocidad en que se modifica la variable controlada y la señal de control (proporcional a la pendiente de la variable) [8].

2.6.4.3 Control PID

Combina los controles proporcional, integral y derivativo [9]. En la Figura 16 se evidencia, en una curva error e versus tiempo t , el comportamiento que toman los parámetros del control PID.

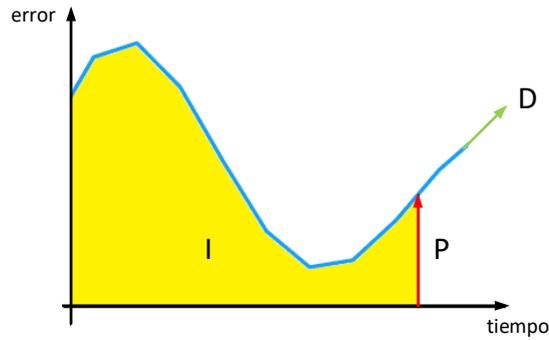


Figura 16. Parámetros del Control PID.

Este control es el más utilizado en la industria debido a su rendimiento robusto en un sin número de condiciones de operación y también debido a su simplificado funcionamiento [9]. En la Figura 17 se puede observar como los controles proporcional, integral y derivativo se unen en el nodo de sumatoria para posteriormente enviar una señal de control hacia el proceso.

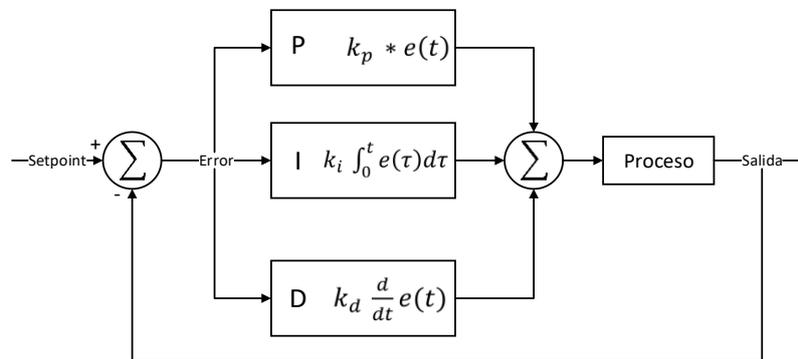


Figura 17. Estructura del Control PID.
Fuente: [7]

La ecuación final del controlador PID es la siguiente:

$$m(t) = k_p * e(t) + k_i \int_0^t e(\tau) d\tau + k_d \frac{d}{dt} e(t) \quad (4)$$

Los parámetros del controlador PID pueden ser calculados matemáticamente a través de la teoría de control. Sin embargo, es necesario obtener previamente la función de transferencia (TF, *por sus siglas en inglés*) del sistema, para esto se requiere que el diseñador determine los diferentes parámetros (mecánicos, físicos, químicos, etc.) que intervienen en dicho sistema, siendo este proceso un tanto tedioso [6].

2.4.4 DISCRETIZACIÓN DE CONTROLADORES

Los controladores discretos pueden ser obtenidos a partir de los controles desarrollados anteriormente, aplicando técnicas de integración numérica [11]:

2.1.4.4 Control Proporcional Discreto

Tomando la ecuación (1) se tiene:

$$m(t) = k_p * e(t)$$

El control proporcional discreto que aproxima esta ecuación es:

$$m(\mathcal{K}T) = k_p * e(\mathcal{K}T)$$

$$P(z) = \frac{M(z)}{E(z)} = k_p \quad (5)$$

Donde T es el periodo de muestreo y \mathcal{K} es el índice que denota el orden del elemento dentro de una secuencia.

2.2.4.4 Control Integral Discreto

De la ecuación (2) se tiene que:

$$m(t) = k_i \int_0^t e(\tau) d\tau$$

Usando la aproximación rectangular se obtiene:

$$m(\mathcal{K}T) = k_i \sum_{i=0}^{\mathcal{K}} T e(iT) = m((\mathcal{K} - 1)T) + k_i T e(\mathcal{K}T)$$

Aplicando transformadas Z se tiene que:

$$I(z) = \frac{M(z)}{E(z)} = \frac{T k_i}{1 - z^{-1}}$$

Con base a esta expresión se puede asumir que el operador integral continuo $\left(\frac{1}{s}\right)$ se transforma en:

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

Entonces el operador derivativo sería:

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

Haciendo uso de la transformada bilineal, que se corresponde con el uso de la estimación trapezoidal de la integral se tiene que:

$$m(\mathcal{K}T) = m((\mathcal{K} - 1)T) + k_i \frac{e((\mathcal{K} - 1)T) + e(\mathcal{K}T)}{2} T$$

Aplicando la transformada Z a esta expresión se obtiene:

$$I(z) = \frac{M(z)}{E(z)} = \frac{T k_i}{2} \frac{z+1}{z-1} \quad (6)$$

2.3.4.4 Control Derivativo Discreto

Tomando la ecuación (3):

$$m(t) = k_d \frac{d}{dt} e(t)$$

Se puede discretizar esta derivada mediante una aproximación por el cociente incremental:

$$m(\mathcal{K}T) = k_d \frac{e(\mathcal{K}T) - e((\mathcal{K} - 1)T)}{T}$$

Aplicando la transformada Z se tiene que:

$$D(z) = \frac{M(z)}{E(z)} = \frac{k_d}{T} \frac{z-1}{z} \quad (7)$$

2.4.4.4 Control Proporcional Integral Derivativo Discreto

Su ecuación en el dominio de s viene dada por:

$$M(s) = k_p \left(1 + \frac{1}{T_i s} + T_d s \right) E(s)$$

Donde k_p es la constante de proporcionalidad, T_i la integral y T_d la derivativa.

Haciendo uso del operador derivada obtenido anteriormente $\left(s = \frac{1-z^{-1}}{T} \right)$ se tiene que:

$$R(z) = \frac{M(z)}{E(z)} = k_p + k_i \frac{z}{z-1} + k_d \frac{z-1}{z} \quad (8)$$

Donde:

$$k_i = \frac{k_p T}{T_i} \quad \text{y} \quad k_d = \frac{k_p T_d}{T}$$

2.5 LÓGICA DIFUSA

2.1.5 LÓGICA CLÁSICA

En la lógica clásica o booleana, una proposición simple P es una declaración lingüística incluida dentro de un universo U de elementos, que puede identificarse como una colección de dichos elementos en U ; y que son verdaderos o falsos. Es decir, la proposición P es un conjunto de componentes donde los valores de verdad, para todos los componentes del conjunto, son todos verdaderos o son todos falsos [12]. En la lógica clásica existe una expresión característica que permite entender de mejor manera la descripción del párrafo anterior [13]:

$$\chi_P = \begin{cases} 1, & x \in P \\ 0, & x \notin P \end{cases} \quad (9)$$

Esta función, χ_P , indica que: si el elemento x pertenece al conjunto P ($x \in P$), entonces la pertenencia es 1, si no pertenece ($x \notin P$) la pertenencia es 0 [13].

2.2.5 CONJUNTOS CLÁSICOS

En los conjuntos clásicos o nítidos, un elemento x en el universo X es o no miembro de algún conjunto nítido A . Esta función de membresía binaria puede ser representada matemáticamente con la función característica o indicadora como se muestra en la ecuación (9) y gráficamente como se indica en la Figura 18 [12].

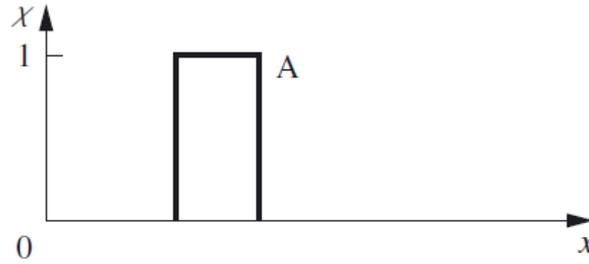


Figura 18. Función de membresía para un conjunto clásico A.
Fuente: [12]

En la Figura 19 se muestra la idea de pertenencia total o nula de los conjuntos clásicos visto desde una perspectiva bidimensional. El punto *a* en la figura es claramente un miembro del conjunto A nítido; el punto *b* no es miembro del conjunto A [12]. A esto se denomina pertenencia total o no pertenencia, respectivamente.

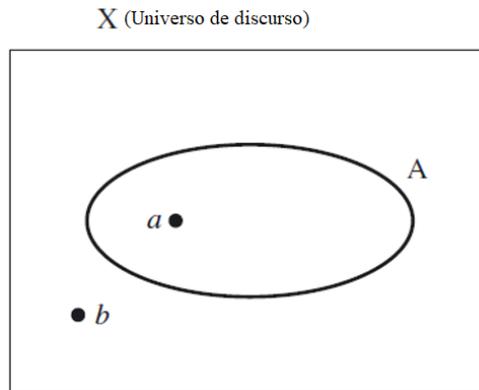


Figura 19. Diagrama del límite definido en un conjunto clásico A.
Fuente: [12]

2.3.5 OPERACIONES CON CONJUNTOS CLÁSICOS

Las operaciones de los conjuntos clásicos son las mismas que rigen en la lógica clásica. Si se establecen los conjuntos nítidos *A* y *B*, las operaciones básicas que se pueden realizar con los conjuntos mencionados se enlistan en la

Tabla I [12].

Tabla I. Operaciones en los conjuntos clásicos.
Fuente: [12]

Principales operadores en los conjuntos clásicos	
Unión	$A \cup B = \{x x \in A \text{ or } x \in B\}$
Intersección	$A \cap B = \{x x \in A \text{ and } x \in B\}$
Complemento	$\bar{A} = \{x x \notin A, x \in X\}$
Diferencia	$A B = \{x x \in A, y x \notin B\}$

2.4.5 INICIOS DE LA LÓGICA DIFUSA

Alrededor de la década de 1920, el polaco Jan Lukasiewicz desarrolló los principios de la lógica multivaluada. En los principios de Lukasiewicz, las proposiciones pueden tomar valores verdaderos fraccionales entre uno y cero de la lógica clásica [13].

Aproximadamente en 1965, el profesor Lofti Zadeh de la Universidad de California en Berkeley, planteó los primeros conceptos de lógica difusa a través de lo que llamó pertenencias parciales y posteriormente como conjuntos difusos [14]. Su publicación denominada *Fuzzy Sets*, muestra las matemáticas de los conjuntos difusos y por extensión de la lógica difusa. Zadeh empleó en la lógica de Lukasiewicz a cada elemento de un conjunto, creando el álgebra necesaria para los conjuntos difusos [13].

Casi una década después Ebrahim Mandani realizó las primeras aplicaciones en la parte de control, en donde aparece el primer controlador difuso implementado para regular un motor de vapor [13], [14]. Después de las primeras aplicaciones y hasta el año 2000, se introdujeron conceptos valiosos como son: la variable lingüística y las reglas si-entonces (if-then), en las cuales se basan la mayoría de aplicaciones de lógica difusa en la actualidad [15].

2.5.5 DEFINICIONES

Definición 1. Es una lógica que permite manipular información en donde existe términos inexactos, intermedios o que son simplemente subjetivos, tratando de emular la manera de cómo procesa dicha información el cerebro humano [13], [14].

Formalmente hablando, la lógica difusa está basada en principios matemáticos que presentan grados de membresía o pertenencia que pueden tomar valores entre 0 y 1, las cuales permiten obtener modelos en base a la información adquirida. Dichos modelos se obtienen en base a reglas lingüísticas que tratan de acercarse a una función mediante la relación de entradas y salidas de un sistema [13].

Definición 2. Se trata de una derivación de la lógica multivaluada que simplifica el modelado de datos atributivos de forma cercana, permitiendo interpretar matemáticamente la incertidumbre y la ambigüedad, suministrando herramientas prudentes para su adecuado procedimiento [15].

2.6.5 ALCANCE

Uno de los alcances de interés que cubre la lógica difusa, son aquellos problemas que se pueden resolver conociendo un conjunto de variables de entrada y obteniendo un conjunto de variables de salida. La lógica difusa permite realizar un mapeo conveniente, según los conjuntos mencionados, apegándose a criterios de significado y no de precisión, es decir, al conocer valores como: Cerca, Medio, Lejos, etc., conocidos como valores lingüísticos [15].

También, dentro de la teoría de los sistemas difusos, su beneficio principal es acercar el comportamiento de sistemas donde las ecuaciones analíticas o las relaciones numéricas no existen. Por lo que, los sistemas difusos tienen un alto potencial para entender sistemas que están privados de ecuaciones, esto es, comprender sistemas complejos. Dichos sistemas complejos pueden ser sistemas nuevos, que no se han probado y de diferente naturaleza. Al mismo tiempo, la relación entre las causas (entradas) y los efectos (salida o salidas) de estos sistemas generalmente no se comprende, pero a menudo se puede observar [12].

Además, a lo largo del estudio de los sistemas difusos, se han establecido como sistemas robustos. Son robustos debido a las incertidumbres incluidas en las entradas y salidas de los sistemas, que son usadas para ajustar sus estructuras en sí, en contraste con el análisis clásico de sistemas en donde primero se propone un modelo, basado en un conjunto de características y parámetros conocidos, para expresar una forma matemática [12].

2.7.5 ENFOQUE

El interés de la lógica es el estudio de la verdad en proposiciones lógicas. En la lógica clásica se toman valores binarios; una proposición puede ser verdadera o puede ser falsa [12].

Bajo este enfoque, la lógica difusa es una técnica para determinar la capacidad humana de razonamiento aproximado. Este razonamiento muestra la disposición humana de razonar de forma cercana y bajo presencia de incertidumbre. En esta lógica, todas las verdades son parciales, donde los extremos binarios de verdadero y falso son interpolados y se representan por la capacidad de la lógica difusa para encapsular verdades fragmentadas [12].

2.8.5 UNIVERSO DE DISCURSO

El universo de discurso es toda la información disponible sobre un problema determinado. Una vez que se define este universo, es posible definir ciertos eventos en este espacio de información [12].

2.9.5 MEMBRESÍA EN LÓGICA DIFUSA

La idea propuesta Lofti Zadeh, explica que la clave para tomar decisiones en donde existe imprecisión es establecer una membresía. El fundamento de instaurar una membresía es vital para representar objetos, dentro de un universo de discurso, por conjuntos contenidos en el mismo universo. Los conjuntos clásicos contienen objetos que satisfacen las propiedades precisas de la membresía; los conjuntos difusos contienen objetos que satisfacen las propiedades imprecisas de la membresía, es decir, la pertenencia a un objeto en un conjunto difuso puede ser *parcial* [12].

Zadeh amplió esta noción de membresía binaria para interpolar varios "grados de membresía" en el intervalo real continuo $[0, 1]$, donde los puntos finales de 0 y 1 se ajustan a no membresía y membresía completa, respectivamente. Al igual que la función indicadora lo hace para conjuntos nítidos, pero donde el número infinito de valores entre los puntos finales puede representar varios grados de membresía para un elemento x en algún conjunto del universo. Los conjuntos en un universo que pueden acomodar "grados de membresía" fueron calificados por Zadeh como conjuntos difusos o valores lingüísticos [12].

2.10.5 VARIABLE LINGÜÍSTICA

Una variable lingüística se diferencia de una variable matemática, en que sus valores no son números, sino palabras u oraciones en un lenguaje natural o artificial. Ya que las palabras, en la mayoría de los casos, son menos precisas que los números, el concepto de una variable lingüística sirve para proporcionar un medio de identificación aproximada de los fenómenos que son demasiado difíciles o están mal definidos como para ser puestos a una descripción en términos cuantitativos convencionales [12]. Ejemplos de estas variables son: Temperatura, Velocidad, Mediano, Pequeño, Muy Pequeño, etc.

2.11.5 PROPOSICIONES DIFUSAS

Una proposición difusa \tilde{P} , es una declaración que involucra algún concepto sin límites claramente definidos. Las proposiciones lingüísticas que tienden a ser ideas subjetivas y que pueden ser interpretadas de forma ligeramente diferente por varios individuos suelen implicar proposiciones difusas. La mayor parte de proposiciones del lenguaje natural es difusa, ya que implica términos vagos e imprecisos [12]. Básicamente existen dos tipos de proposiciones [13]:

- **Atómicas:** Relacionan una sola variable lingüística con un solo valor lingüístico. Por ejemplo: La *Temperatura* es *Alta*.
- **Compuestas:** Son aquellas unidas por un operador o conector lingüístico del tipo “y”, “o”, “no” y relacionan dos o más variables y valores lingüísticos. Por ejemplo: SI *Temperatura* es *Alta* Y *Presión* es *Baja*.

2.12.5 REGLAS DIFUSAS

Son reglas que permitan manejar las variables y valores lingüísticos que se han agregado al universo de discurso. Las reglas son los conectores que unen proposiciones difusas denominadas ANTECEDENTE con otras proposiciones difusas llamadas CONSECUENTE [13], [14]. Son del tipo IF –THEN (SI-ENTONCES) como se muestra a continuación [15] - [17]:

If <proposición difusa (antecedente)> then <proposición difusa (consecuente)>

Cuando son aplicadas generalmente presentan esta forma más sencilla:

If x is A then y is B

En la lógica difusa, cuando el antecedente es verdadero con cierto grado de pertenencia, el consecuente también es verdadero en cierto grado, esto se debe a que todas las reglas difusas se ejecutan parcialmente. Las reglas difusas son capaces de modelar y usar una opinión experta o conocimiento de sentido común a menudo expresado en términos lingüísticos [18].

2.13.5 CONJUNTOS DIFUSOS

Como se mostró en la sección 2.2.5, un conjunto clásico se define por límites claros. En contraste, un conjunto difuso, está prescrito por propiedades vagas, difusas o llamadas también borrosas; por lo tanto, sus límites se especifican de forma ambigua. Esto se muestra en la Figura 20, en donde existe un límite difuso para el conjunto \tilde{A} [12].

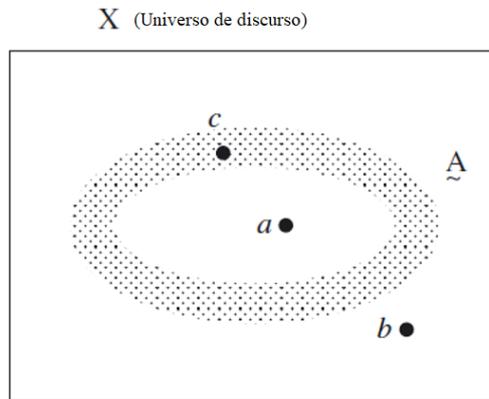


Figura 20. Diagrama del límite de un conjunto difuso.
Fuente: [12]

En la Figura 20, la región sombreada, muestra un límite impreciso para un conjunto difuso \tilde{A} , el cual está contenido en el mismo universo de discurso del conjunto clásico presentado en la Figura 19. En la región central sin sombreadar, el punto a es claramente un elemento con membresía completa (1) del conjunto. Fuera de la región sombreada del conjunto \tilde{A} , el punto b es fácilmente distinguible como un elemento con membresía nula (0). Sin embargo, el punto c , contenido dentro de la región sombreada presenta una membresía ambigua o difusa. Posiblemente, la pertenencia del punto c en \tilde{A} se acerca a un valor de 1 cuando c se mueva hacia la región central de \tilde{A} , mientras que si c tiende a salir de la región sombreada, su pertenencia se acerque a 0 [12].

Según lo anterior, para un elemento en un universo que contiene conjuntos difusos, la transición puede ser gradual. Se puede pensar que esta transición entre diversos grados de membresía se ajusta al hecho de que los límites de los conjuntos difusos son aproximados y confusos. Por lo que, un conjunto difuso contiene elementos que presentan diferentes grados de membresía, y es esta membresía diversa la que permite que los elementos puedan ser miembros de otros conjuntos difusos dentro del mismo universo [12].

Por lo tanto, la pertenencia de un elemento a un conjunto borroso se mide mediante una función denominada pertenencia o membresía que pretende describir la aproximación y la ambigüedad dentro de un intervalo entre 0 y 1. La función que mapea los elementos de un conjunto difuso, en este caso \tilde{A} , a un valor real entre 0 y 1 viene dado por la ecuación (10) [12]:

$$\mu_{\tilde{A}}(x) \in [0,1] \quad (10)$$

La gráfica de esta función de membresía que realiza el mapeo de un elemento en un conjunto difuso típico se ilustra en la Figura 21 [12]:

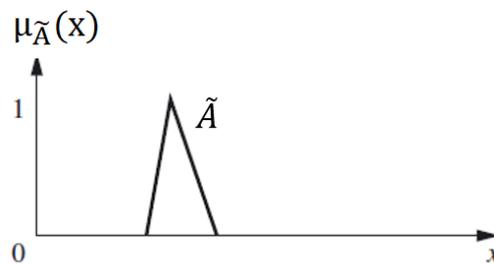


Figura 21. Función de mapeo.
Fuente: [12]

$\mu_{\tilde{A}}(x)$ representa el grado de pertenencia del elemento x al conjunto difuso \tilde{A} que está incluido en el universo X , en este caso. Concretamente, $\mu_{\tilde{A}}(x)$ toma valores en el rango $[0,1]$ que miden e indican cuánto corresponde el elemento x a su conjunto. Una expresión apegada a esta descripción sería: $\mu_{\tilde{A}}(x) = \text{grado al cual } x \in \tilde{A}$ [12].

2.14.5 OPERACIONES CON CONJUNTOS DIFUSOS

Si se establecen dos conjuntos difusos \tilde{A} y \tilde{B} en el universo de discurso X para un determinado elemento x en el mismo universo, las operaciones para los conjuntos \tilde{A} y \tilde{B} en X son [12]:

Tabla II. Operaciones en conjuntos difusos
Fuente: [12]

Principales operadores en los conjuntos difusos		
Operación	Notación	Operador
Unión (OR)	$\mu_{\tilde{A} \cup \tilde{B}}(x) = \mu_{\tilde{A}}(x) \cup \mu_{\tilde{B}}(x)$	$\max\{\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)\}$
Intersección (AND)	$\mu_{\tilde{A} \cap \tilde{B}}(x) = \mu_{\tilde{A}}(x) \cap \mu_{\tilde{B}}(x)$	$\min\{\mu_{\tilde{A}}(x), \mu_{\tilde{B}}(x)\}$
Complemento (NOT)	$\overline{\mu_{\tilde{A}}}(x) = 1 - \mu_{\tilde{A}}(x)$	$\overline{\mu_{\tilde{A}}}(x) = 1 - \mu_{\tilde{A}}(x)$

2.15.5 FUNCIONES DE MEMBRESÍA O DE PERTENENCIA

Como se mencionó anteriormente, una función de membresía o función de pertenencia indica el grado con el que un elemento x pertenece a un conjunto difuso dentro de un universo X [12], [16]. A continuación, se describen algunas de sus partes más significativas y los tipos que existen.

2.1.5.15 Características de las Funciones de Membresía

Las principales partes que caracterizan una función de pertenencia son las siguientes [12], [17]:

- **Núcleo:** Se define como la región del universo de discurso que se especifica por la membresía completa para un conjunto difuso \tilde{A} . Es decir, el núcleo comprende todos los elementos x del universo de modo que $\mu_{\tilde{A}}(x) = 1$
- **Soporte:** Se especifica cómo la región del universo de discurso que se caracteriza por tener una membresía diferente de cero. Esto es, el soporte encierra los elementos x del universo tal que $\mu_{\tilde{A}}(x) > 0$.
- **Límites:** Se definen como aquella región del universo de discurso que encierra elementos que presentan una membresía distinta de cero, pero tampoco una membresía completa. Es decir, los límites comprenden los elementos x del universo tal que $0 < \mu_{\tilde{A}}(x) < 1$. Estos elementos son aquellos dentro del universo que muestran cierto grado de borrosidad o grado de membresía parcial en el conjunto A .

En la Figura 22 se muestran las partes descritas sobre la función de membresía [12].

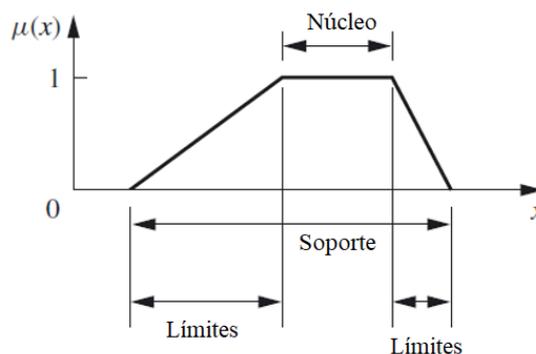


Figura 22. Partes de una función de pertenencia.
Fuente: [12]

2.2.5.15 Tipos de Funciones de Membresía

Las principales funciones de membresía utilizadas en lógica difusa son [16]- [18]:

- **Función de Membresía Triangular:** Esta función se ilustra en la Figura 23. Viene definida por la ecuación (11).

$$\mu_{\tilde{A}}(x) = \begin{cases} \frac{x-a}{b-a}, & a \leq x \leq b; \\ \frac{c-x}{c-b}, & b < x \leq c; \\ 0, & x < a \text{ o } x > c \end{cases} \quad (11)$$

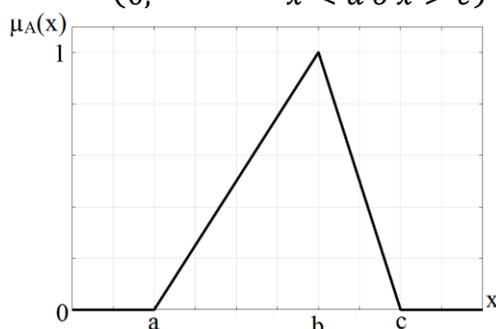


Figura 23. Función de pertenencia triangular.

- **Función de Membresía de Distribución Normal:** Esta función se muestra en la Figura 24. Viene definida por la ecuación (12).

$$\mu_{\tilde{A}}(x) = e^{-\left(\frac{x-a}{b}\right)^2}, \quad b > 0 \quad (12)$$

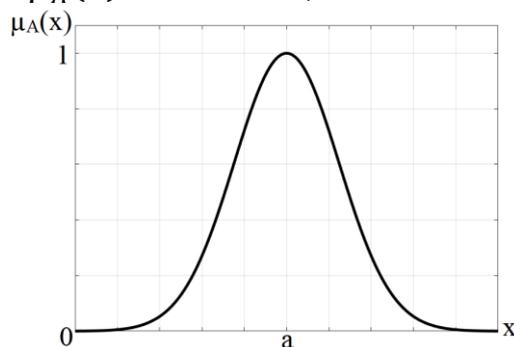


Figura 24. Función de pertenencia de distribución normal.

- **Función de Membresía Trapezoidal:** Esta función se presenta en la Figura 25. Se especifica por la ecuación (13).

$$\mu_{\bar{A}}(x) = \begin{cases} 0, & x < a; \\ \frac{x-a}{b-a}, & a \leq x < b; \\ 1, & b \leq x \leq c; \\ \frac{d-x}{d-c}, & c < x \leq d; \\ 0, & x > d \end{cases} \quad (13)$$

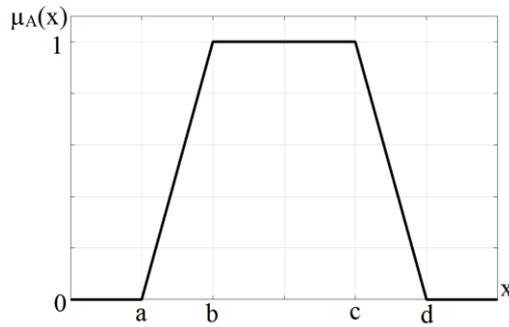


Figura 25. Función de pertenencia trapezoidal.

- **Función de Membresía Sigmoideal:** Esta función se ilustra en la Figura 26. Viene definida por la ecuación (14).

$$\mu_{\bar{A}}(x) = \begin{cases} 0, & x \leq a; \\ 2 \left(\frac{x-a}{b-a} \right)^2, & a < x \leq m; \\ 1 - 2 \left(\frac{x-b}{b-a} \right)^2, & m < x < b; \\ 1, & x \geq b \end{cases} \quad (14)$$

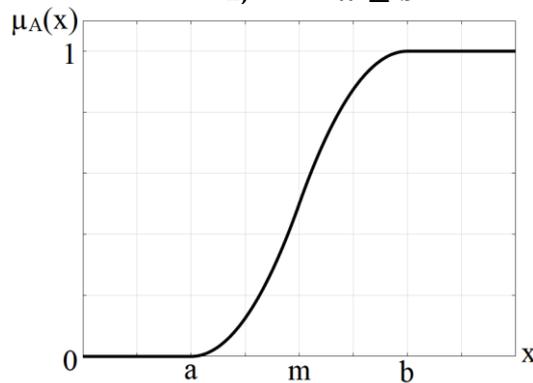


Figura 26. Función de pertenencia sigmoideal.

2.16.5 INFERENCIA DIFUSA

La inferencia difusa se define como el proceso de obtener un valor de salida o conclusión para una entrada dada que posiblemente nunca se haya mostrado antes, usando conjuntos difusos [15], [19]. Existen varios métodos de inferencia difusa, los más

conocidos son los de Mamdani y Takani-Sugeno-Kang (TSK), mejor conocida como Sugeno, como se muestran a continuación [20].

2.1.5.16 Inferencia de Mamdani

Es el método más utilizado en la inferencia difusa. El proceso de inferencia de Mamdani se puede resumir en cuatro pasos [15]:

- **Fuzificación:** consiste en tomar los valores de las entradas y determinar el grado de pertenencia de estas entradas a los conjuntos difusos respectivos.
- **Evaluación de las reglas difusas:** se toman las entradas anteriores y se aplican a los antecedentes de las reglas difusas. Si una regla tiene múltiples antecedentes, se utiliza el operador AND para obtener un único número que represente el resultado de la evaluación. Este número se aplica al consecuente. La evaluación de las reglas se realiza como sigue [20]:
 - **Evaluación del antecedente:** Calcula el grado de verdad del lado izquierdo (LI) de cada regla para los valores de entrada actuales. Si la regla que se calcula es la k -ésima, el grado de verdad viene dado por la evaluación del LI en la función de pertenencia, es decir: $\mu_{LI}(k)$.
 - **Evaluación del consecuente:** Según el grado de verdad calculado, se define la conclusión de la regla. Se establece a la variable de salida del consecuente, el conjunto difuso correspondiente modificado por el grado dado de $\mu_{LI}(k)$. La función de pertenencia del conjunto modificado se representa por $\mu_{LD}(k)(v)$. Donde el lado derecho (LD) es el consecuente de la regla, k es la regla evaluada y v la salida. Esta modificación se ejecuta usando la implicación difusa. La implicación difusa relaciona el antecedente y el consecuente como sigue: Considerando dos conjuntos difusos A y B en un universo de discurso U y V según corresponda, una implicación difusa de A en B ($A \rightarrow B$) es una relación difusa $U \times V$ que se establece por:
 - **Implicación de Zadeh:** Se implementa con el producto acotado. La ecuación (15) muestra esta implicación.

$$\mu_A \rightarrow B(x, y) = \min\{1, 1 - \mu_A(x) + \mu_B(y)\} \quad (15)$$

- **Implicación de Mamdani:** Es el método más usado en implicaciones difusas. Utiliza el mínimo como muestra la ecuación (16).

$$\mu_A \rightarrow B(x, y) = \min\{\mu_A(x), \mu_B(y)\} \quad (16)$$

- **Agregación de las salidas de las reglas:** es el proceso de unificación de las salidas de todas las reglas; es decir, se combinan las funciones de pertenencia de todos los consecuentes previos, para obtener un único conjunto difuso por cada variable de salida. El proceso de agregación se puede explicar de la siguiente manera [20]. Si se tienen las reglas base:

Regla 1: Si x_1 es A_1 y x_2 es A_2 y ... y x_n es A_n Entonces v es B_1

Regla 1: Si x_1 es A_1 y x_2 es A_2 y ... y x_n es A_n Entonces v es B_2

⋮

Regla M: Si x_1 es A_1 y x_2 es A_2 y ... y x_n es A_n Entonces v es B_M

En el proceso se evalúa cada regla, el LI primero y luego el LD. El resultado entrega M conjuntos difusos por las M reglas difusas. Para casos de que dos o más resultados coincidan en el mismo término lingüístico, el resultado en el LD se halla aplicando el operador máximo (OR). Al final, el conjunto difuso total de salida se halla por la operación unión difusa (\cup) de los conjuntos B resultantes:

$$B = B_1 \cup B_2 \cup \dots \cup B_M \quad (17)$$

- **Defuzificación:** En esta etapa se toma como entrada el conjunto difuso anterior (B), para posteriormente dar un valor de salida de forma nítida. Más adelante se detallan los métodos de defuzificación.

2.2.5.16 Inferencia de Sugeno

Takagi, Sugeno y Kang propusieron un modelo que contiene reglas donde el antecedente está constituido por variables lingüísticas y el consecuente se compone por una función lineal de las variables de entrada. El proceso de inferencia de Sugeno se puede describir como sigue [20]:

- **Evaluación de las reglas difusas:** Para el caso del LI se realiza igual que Mamdani como se explicación a continuación:
 - **Evaluación del antecedente:** El LI de las reglas difusas se halla al igual que Mamdani; usando uno de los operadores de implicación se obtiene un grado de pertenencia α para cada regla.
 - **Evaluación del consecuente:** El LD de las reglas se obtiene a través de la combinación lineal de las entradas como se indica:

$$v = f(x_1, x_2 \dots x_n) \quad (18)$$

- **Salidas:** Para el caso de Sugeno, con una base de conocimiento de M reglas, la salida se encuentra como una media ponderada de cada salida v , como indica la siguiente expresión:

$$salida = \frac{\sum \alpha_j v_j}{\sum \alpha_j} \quad (19)$$

Donde $v_j = f_j(x_1, x_2 \dots x_n)$ es la salida de cada regla y α_j el grado de pertenencia que resulta de la inferencia.

2.3.5.16 Métodos de Defuzificación

Existen diferentes métodos de defuzificación entre los cuales se pueden recalcar [12]:

- **Método de Máxima Membresía:** también conocido como el método de altura, se limita a las funciones de salida en pico. Este método está dado por la ecuación (20):

$$\mu(z^*) \geq \mu(z) \text{ para todo } z \in Z \quad (20)$$

Donde z^* es el valor defuzificado (nítido) como se muestra en la Figura 27.

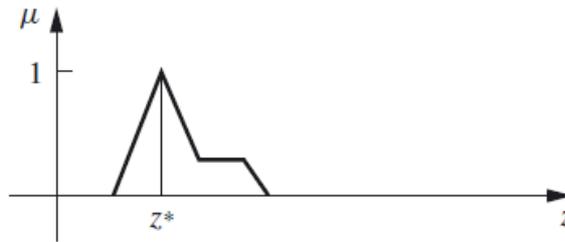


Figura 27. Método de Máxima Membresía

Fuente: [12]

- **Método del Centroide:** También denominado centro de área o centro de gravedad. Es el método de defuzificación más usado. Se define por la ecuación (21):

$$z^* = \frac{\int_a^b \mu(z) * z dz}{\int_a^b \mu(z) dz} \quad (21)$$

Donde \int indica una integración algebraica. Su gráfica se muestra en la Figura 28. Para el caso de que se manejen valores discretos la ecuación del método del centroide viene dada por la ecuación (22) [19]:

$$z^* = \frac{\sum_{k=1}^K \mu_k * f(\mu_k)}{\sum_{k=1}^K f(\mu_k)} \quad (22)$$

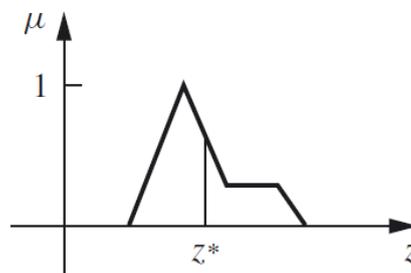


Figura 28. Método del Centroide

Fuente: [9]

- **Método de promedio ponderado:** es el más utilizado en aplicaciones difusas ya que es uno de los métodos más eficientes desde el punto de vista computacional. En la mayoría de las ocasiones está restringido a funciones simétricas de membresía de salida. Está definido por la ecuación (23):

$$z^* = \frac{\sum \mu_{\bar{A}}(\bar{z}) * \bar{z}}{\sum \mu_{\bar{A}}(\bar{z})} \quad (23)$$

Donde \sum indica una suma algebraica y \bar{z} es el centroide de cada función de membresía simétrica. Este método se forma ponderando cada función de membresía en la salida por su valor de membresía máximo respectivo. Es decir, si dos funciones, como se muestran en la Figura 29, darían como resultado la siguiente forma general para el valor defuzificado:

$$z^* = \frac{a(0.5) + b(0.9)}{(0.5) + (0.9)}$$

En donde a y b son los centroides de cada una de las funciones de membresía simétricas.

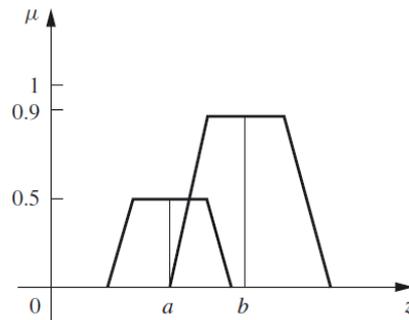


Figura 29. Método de promedio ponderado.

Fuente: [12]

Método de membresía máxima promedio: está estrechamente relacionado con el primer método, excepto que las ubicaciones de la membresía máxima no son únicas. Este método viene dado por la ecuación (24):

$$z^* = \frac{a + b}{2} \quad (24)$$

Donde a y b se son definidos como se indica en la Figura 30.

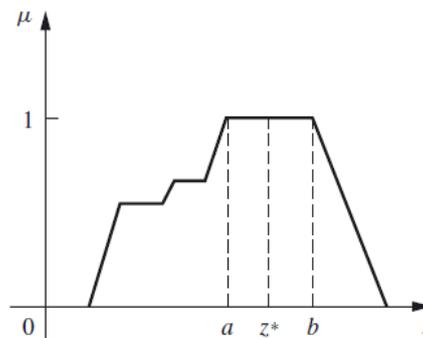


Figura 30. Método de membresía máxima promedio.

Fuente: [12]

2.6 CONTROL DIFUSO

Lo que planteó Lofti Zadeh, fue formular el control sistemáticamente en términos de reglas difusas simples. Esta idea conduce al desarrollo del control difuso y de controladores difusos [17]. Se ingresó por primera vez en la década de los 70 como una tentativa para diseñar controladores para sistemas con modelos complejos, debido a su no linealidad [20].

La meta de un controlador difuso es implementar sistemas de inferencia difusa, dentro de la teoría de control automática clásica. Un controlador difuso está basado en un sistema de inferencia difusa SISO (*Simple Input Simple Output*). Comúnmente, se puede confundir al llamar a un sistema de inferencia difusa SISO como un controlador difuso. El uso de esta terminología es diferente en la teoría de control clásica. El controlador difuso debe tener la salida vinculada a un sistema de control [17], tal y como se describe a continuación.

2.1.6 ESTRUCTURA DE UN SISTEMA DIFUSO SISO

Un sistema difuso de entrada simple y salida simple (Sistema difuso SISO) usa una entrada nítida, la difuzifica, la mapea a través de un sistema de inferencia difusa y la salida difusa que se obtiene se transforma en una salida nítida. A menudo, los sistemas SISO difusos se utilizan en aplicaciones de control, denominados comúnmente controladores difusos [17]. Su estructura se muestra en la Figura 31.

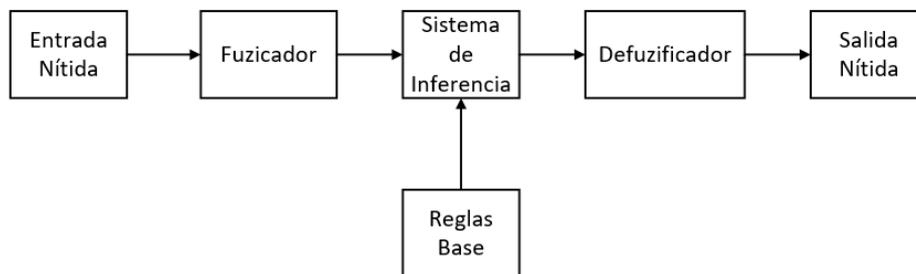


Figura 31. Componentes de un controlador difuso.

Fuente: [17]

En general se tiene [17]:

- **Entrada Nítida:** Una entrada nítida o clara (real). Es fácilmente diferenciable en el lenguaje humano, pero con presencia de incertidumbre.

- **Fuzificador (Fuzificación):** Encargado de convertir los valores nítidos leídos en valores lingüísticos difusos, evaluados en las funciones de membresía. Existen tres tipos de Fuzificación [20]:
 - **Singleton:** simplifica el cálculo que se realiza en el sistema de inferencia para algún tipo de función de pertenencia que pueda adoptar las reglas *si – entonces*.
 - **Gaussiano:** simplifica el cálculo en el sistema de inferencia *solo si* las funciones de pertenencia son del tipo gaussiano (normal).
 - **Triangular:** facilita el cálculo en el sistema de inferencia *solo si* las funciones de membresía son del tipo triangular.
- **Sistema de Inferencia:** Se encarga de calcular las variables de salida a partir de las variables de entrada, a través de las reglas difusas, entregando como salida conjuntos difusos [20]. Aquí se toma las decisiones para el controlador y es el núcleo del mismo, esto permite simular la lógica humana.
- **Reglas Base:** También denominada base de conocimiento. Aquí se encuentran las reglas difusas Si-Entonces. Además, es en donde se almacena el conocimiento dada por un experto.
- **Defuzificador (Defuzificación):** Encargado de convertir los valores difusos de salida en valores nítidos, usando algún método de defuzificación. Pudiendo realizar la acción de control clásica partiendo de una acción difusa.
- **Salida Nítida:** La salida como valor real (nítido, claro).

2.2.6 CONTROLADOR DIFUSO

Los controladores difusos tienen el mismo objetivo que un controlador clásico; hacer que el comportamiento de una planta sea como se espera y no de forma aleatoria. El esquema general de la aplicación de un controlador difuso se muestra en la Figura 32 [20].

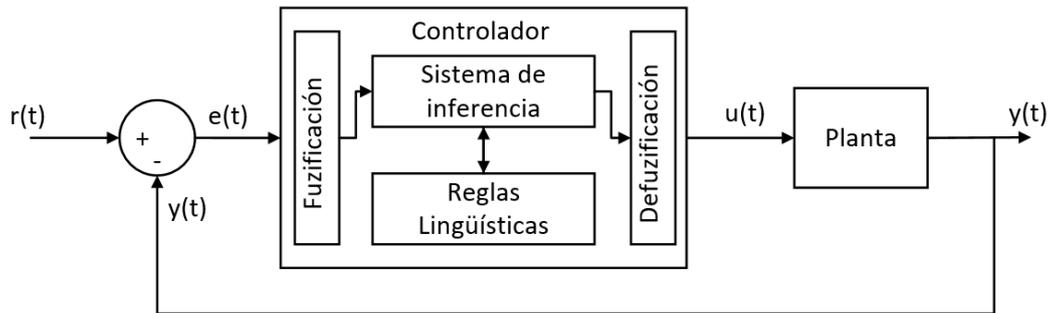


Figura 32. Esquema general de un controlador difuso.
Fuente: [20]

El controlador difuso decide la mejor acción de control $u(t)$ dependiendo de los valores de entrada $e(t)$, apoyándose en la base de conocimientos que trabaja junto con las reglas lingüísticas [20].

2.7 IDENTIFICACIÓN DE SISTEMAS (BLACK BOX MODEL)

Los sistemas o plantas cumplen la función de tomar una entrada para posteriormente producir una salida. Su identificación parte de la observación del comportamiento de la entrada y la salida mientras son sometidas a ensayos [6]. Para identificar totalmente un sistema es necesario conocer sus parámetros internos, debido a que la salida del sistema depende de la forma de interacción de estos [21]. Sin embargo, existen sistemas con una dinámica compleja, en donde sus parámetros internos no pueden ser descritos, es por esto que se usan diferentes técnicas para su identificación, por ejemplo, mediante software. En [22] el autor usa una señal PRBS (*Pseudorandom binary sequence*) y el TOOLBOX IDENT de MATLAB® para obtener la función de transferencia (ecuación que relaciona la salida con la entrada de un sistema).

Otra técnica de identificación es mediante sistemas tipo caja negra (Black Box). Los parámetros internos (que relacionan la entrada con la salida) no son conocidos. Para abordar este inconveniente se lleva a cabo una estimación, esto con el fin de describirlos. Los resultados que se adquieren en este proceso de estimación (el cual depende del error y su actualización) son precisos para que funcione la estructura de identificación, filtrado adaptativo.

El método de identificación de sistemas Black Box permite conocer únicamente su función de transferencia más no acceder a sus dinámicas internas de forma directa. Para conocer las dinámicas internas (de forma indirecta) se lleva a cabo el proceso de filtrado, el cual está compuesto por: la estimación e identificación.

El modelado de la dinámica interna de estos sistemas se realiza a través de ecuaciones diferenciales (sin conocer que sucede internamente). Este modelado está basado en la función de transferencia. En la Figura 33 se ilustra en diagrama de bloques del comportamiento del filtro digital como identificador [21].

En este diagrama, v_k es una señal de entrada tanto para el sistema de caja negra como para el nodo del filtro digital (que se comporta como identificador); w_k es una perturbación externa, y_k es la respuesta del sistema de caja negra adicionada la perturbación, \hat{y}_k es la respuesta del filtro digital. La señal de error e_k se obtiene de la diferencia entre y_k y \hat{y}_k para posteriormente ser comparada con la señal de error anterior e_{k-1} . Luego de la comparación se obtiene uno de dos resultados posibles:

- Si $e_k \leq e_{k-1}$ se obtiene la señal $\hat{y}_{k \text{ C.T.P}}$ que es la cual proporcionará los parámetros para la obtención de la función de transferencia del sistema.
- Si el caso anterior no se cumple, la señal de error se envía hacia el algoritmo de adaptación.

La salida del algoritmo de adaptación se envía al nodo del filtro digital en donde se produce una diferencia con v_k , obteniendo así una señal que es enviada al filtro digital, la cual lo corrige en su proceso de identificación haciendo que el filtro llegue a la convergencia [21].

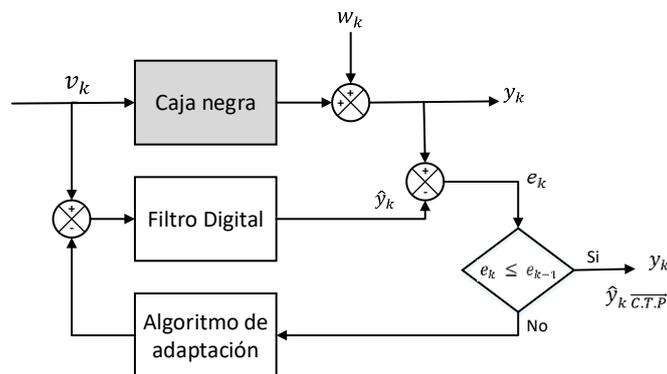


Figura 33. Esquema del filtrado digital adaptativo.
Fuente: [21]

Las técnicas de estimación que se utilizan pueden ser recursivas y no recursivas [21]:

- **No recursivas:** tratan toda la información de entrada-salida utilizable para luego proporcionar un arreglo unidimensional, el cual contiene los parámetros del sistema.
- **Recursivas:** a medida que se va tratando la información se computan los parámetros del modelo; luego de obtener dichos, se desarrolla la correspondiente TF.

El proceso de estimación tradicional no puede llevarse a cabo si la función de transferencia no está acotada. Para esta estimación se considera [21]:

- Condiciones estacionarias.
- Error de seguimiento.
- Criterio para ajustar al estimador.
- Observar la función de transferencia.

Otra opción para llevar a cabo la identificación de sistemas mediante el método Black Box es mediante redes neuronales, tal y como lo hacen en [22] en donde usan funciones de activación sigmoidales y lineales.

2.8 SOFTWARE

2.1.8 MATLAB®

MATLAB® es un software matemático creado en 1970 por Clever Moler [23]. Este software tiene como objetivo ayudar a resolver cálculos científicos a través de sus diferentes herramientas. En sus primeras versiones contaba con herramientas básicas, pero al pasar los años y gracias a los aportes de los usuarios se ha convertido en un software más completo y de uso multidisciplinario [24]. Proviene de la abreviatura MATrix LABoratory (Laboratorio de Matrices); posee un lenguaje de programación propio (lenguaje M) y ofrece un Entorno de Desarrollo Integrado (IDE) [25]. En la Figura 34 se aprecia su entorno de trabajo.

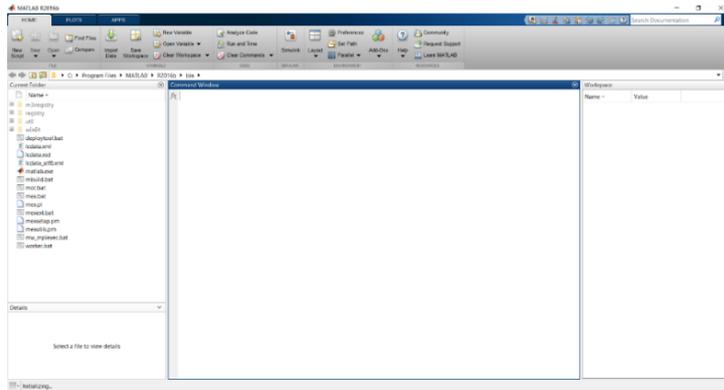


Figura 34. Entorno de trabajo de MATLAB®.
Fuente: [25]

Además, presenta múltiples prestaciones, como [24] - [27]:

- Procesamiento de señales.
- Simulación de sistemas dinámicos.
- Representación de datos y funciones.
- Comunicación con software y hardware externo.
- Interfaces de usuario.

También posee herramientas adicionales que permiten a los usuarios trabajar con mayor facilidad. Algunas de estas herramientas son: TOOLBOX, GUIDE y Simulink [25]. Un ejemplo de TOOLBOX es el Fuzzy Logic Designer y es de gran ayuda en el desarrollo de lógica difusa, Figura 35.

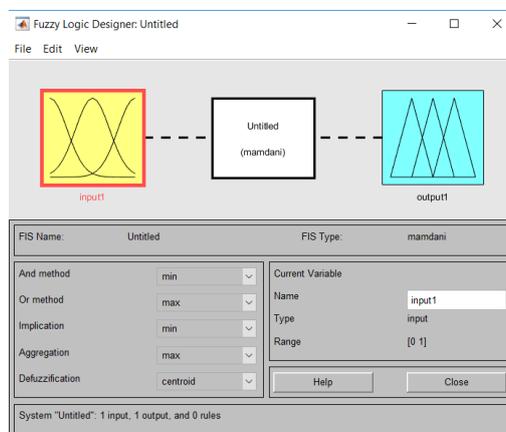


Figura 35. Entorno del TOOLBOX Fuzzy Logic Designer
Fuente: [27]

Por otro lado, Simulink (plataforma de simulación multidominio) [25] es una herramienta que permite realizar simulaciones de diferentes sistemas a través de su librería de bloques (BLOCKSETS), Figura 36. Cabe recalcar que las propiedades de estos bloques son manipulables, por lo que el usuario debe configurarlos de tal forma que el sistema que pretende simular se comporte lo más próximo posible al sistema físico.

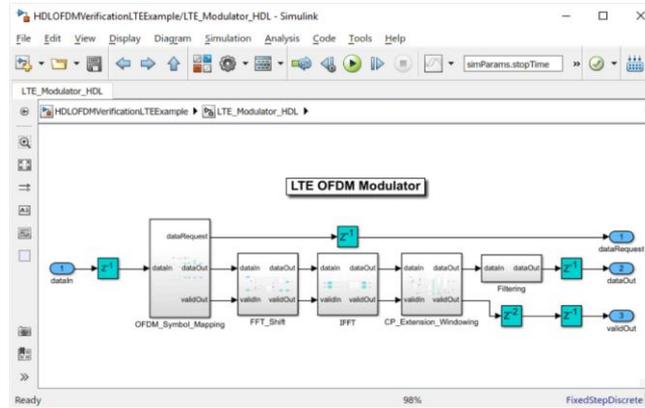


Figura 36. Entorno de Simulink.
Fuente: [29]

2.2.8 LABVIEW®

LabVIEW® es el acrónimo de Laboratory Virtual Instrumentation Engineering Workbench [30]. Su desarrollo comenzó en el año de 1983 por la empresa National Instrument, para luego, en 1986 ser oficialmente puesto en el mercado para los computadores Macintosh. Posteriormente, para el año de 1992 se encontraba disponible para Windows [31]. En la actualidad, LabVIEW® se encuentra disponible para los principales sistemas operativos: Windows, Linux, MAC.

Desde sus inicios LabVIEW® se ha enfocado tanto al desarrollo de soluciones para la industria como para el mundo académico, es por esto que continuamente mejora su entorno, esto con la finalidad de que el usuario pueda resolver sus diseños de la manera más simple y flexible [30]. LabVIEW® es una herramienta de programación que utiliza un lenguaje G (gráfico); trabaja en conjunto con dos ventanas, una llamada Panel Frontal y la otra Diagrama de Bloques [31].

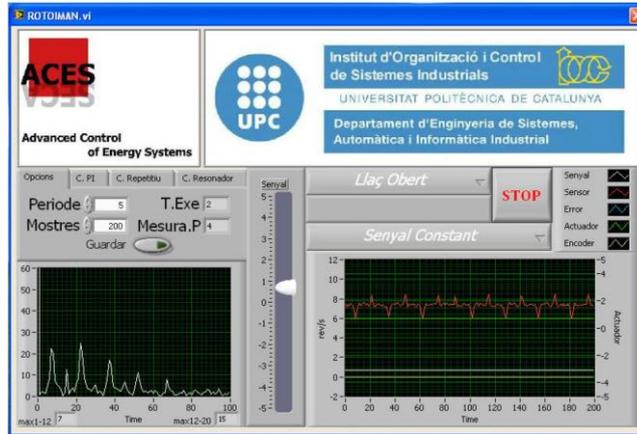


Figura 37. Panel Frontal en LabVIEW®.
Fuente: [30]

El Panel Frontal, Figura 37, es la interfaz con la cual interactuará el usuario, allí dispondrá de botones, pantallas, paneles, etc. Éstos pueden ser controladores o indicadores [31]. La interfaz toma las entradas del usuario (a través de los controles) y responde con salidas proporcionadas por la lógica del programa (mediante los indicadores) [32]. En cuanto al Diagrama de Bloques, Figura 38, es en donde se realizará la programación de forma gráfica a través de los bloques de las librerías multifuncionales que dispone LabVIEW® [31]. La programación que se desarrolla en LabVIEW® se la conoce como Virtual Instrument (VI) debido a que son similares y se comportan como instrumentos físicos [30]. Con respecto al tipo de variables que maneja LabVIEW®, éstas pueden ser: integer, float, double, boolean, bit, string, char y entre otras [31].

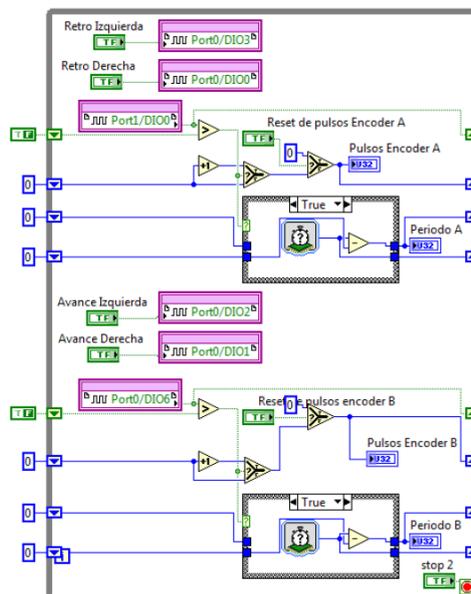


Figura 38. Diagrama de bloques.
Fuente: [32]

Además, también posee múltiples librerías especializadas en diversas aplicaciones, algunas de ellas son [30]:

- Control de flujo.
- Procesamiento de señales.
- Control de procesos.
- Instrumentación.
- Manejo de archivos.

En cuanto a comunicación remota, LabVIEW® ha implementado Paneles Remotos. Estos paneles son proyecciones de Paneles Frontales a través de internet haciendo uso de servidores internos.

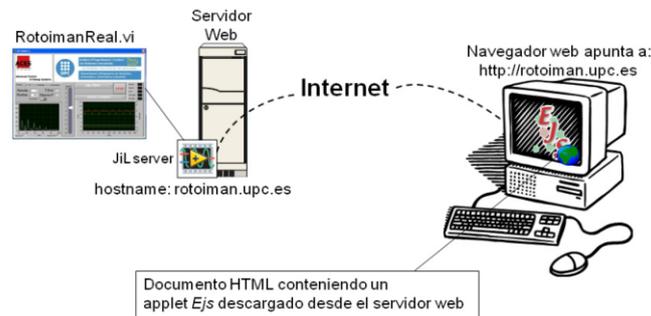


Figura 39. Topología de una comunicación remota en LabVIEW®.
Fuente: [30]

En la Figura 39 se aprecia la proyección de un Panel Frontal en un ordenador a través de un servidor WEB de LabVIEW® e internet. También, LabVIEW® presenta flexibilidad al momento de trabajar con interfaces de comunicación como: USB, TCP/IP, OPC, Bluetooth, puerto serial y paralelo, UDP, entre otros [32].

2.9 HARDWARE

2.1.9 ARDUINO

Es una plataforma de código abierto cuyo hardware está basado en un circuito impreso y un microcontrolador de la marca *ATMEL*. Este microcontrolador cuenta con entradas y salidas digitales y analógicas. Arduino, con respecto a otros microcontroladores, simplifica el proceso de programación y brinda mejores ventajas y características. Esta plataforma se programa a través de un lenguaje propio (programación

de alto nivel processing). Su entorno de programación, Figura 40, es amigable con el usuario, facilita el desarrollo de código y su carga en el microcontrolador [33].

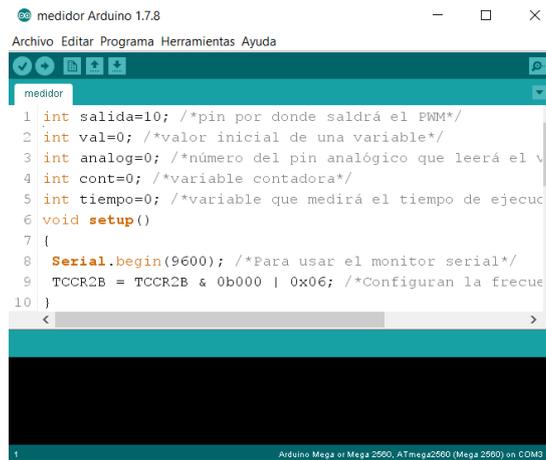


Figura 40. Entorno de programación Arduino.

Arduino está basado en el lenguaje de programación C, pero también soporta algunas funciones de C++. Además, funciona en los sistemas operativos Windows, MAC y Linux [33].

2.1.9.1 Descripción de la Placa Arduino

Existen varios modelos de Arduino que han sido creados de manera oficial y no oficial. Los modelos no oficiales son desarrollados por usuarios que han adaptado el diseño original de la placa Arduino a sus proyectos. Esto es posible debido a que en internet se encuentran los planos electrónicos junto con la licencia del producto (tecnología abierta) [33]. De manera general, una placa Arduino está constituida por una serie de puertos analógicos, digitales, de alimentación (5VDC o 3.3VDC y GND) y comunicación (Tx - Rx). También posee un conector USB que es mediante el cual se cargan los programas al microcontrolador ATMEGA; además también posee un botón de reset y un adaptador AC-DC (6-20 Voltios) [33]. El modelo Arduino UNO se observa en la Figura 41.



Figura 41. Arduino UNO.
Fuente: [34]

2.2.9.1 Arduino MEGA

Esta placa es más grande que la versión UNO, debido a que posee más pines digitales, analógicos y de comunicación; además de ser más potente, Figura 42. Utiliza un microcontrolador ATmega2560 de 256KB de memoria Flash, de los cuales 8KB es utilizado para el gestor de arranque, otros 8KB de SRAM y 4KB de memoria EEPROM. También cuenta con las siguientes características [33]:

- 54 pines I/O (15 pines son dedicados para salida PWM).
- 16 entradas analógicas.
- Oscilador de 16MHz.
- Puertos seriales.
- Conexión USB.
- Cabecera ICSP.
- Conector de alimentación.
- Botón de reset.
- 7 a 12 Voltios de operación.
- 6 a 20 Voltios de entrada.
- Corriente DC 40mA.

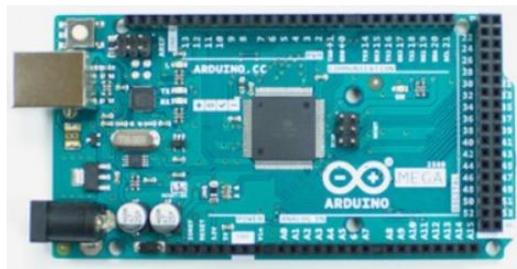


Figura 42. Aspecto físico de la placa Arduino MEGA 2560.
Fuente: [35]

2.2.9 MYDAQ

Es una tarjeta electrónica portátil (de fácil manejo y reducido costo) que usa un puerto USB tanto para comunicación como energización. Puede ser utilizada como un laboratorio portátil, el cual cuenta con ocho instrumentos virtuales definidos por software [36]. Además, incluye un generador de funciones, osciloscopio y multímetro digital [37]. La tarjeta MyDAQ mejora sus prestaciones cuando trabaja en conjunto con el software LabVIEW® y Multisim®, debido a que permite diseñar prototipos de sistemas y circuitos

fuera del laboratorio. La portabilidad de esta tarjeta de adquisición (DAQ) es posible debido a las dimensiones y peso que ofrece su diseño, Figura 43 [7].

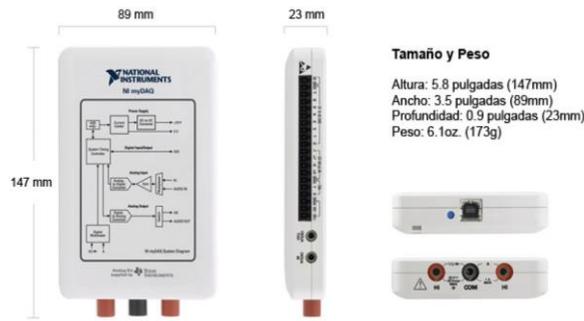


Figura 43. Dimensiones y peso de la tarjeta MyDAQ.
Fuente: [7]

2.1.9.2 Datos de Hardware

MyDAQ cuenta con [7]:

- 8 I/O digitales.
- 2 entradas analógicas.
- 2 salidas analógicas.
- 1 entrada y 1 salida para audio a través de un Jack de 3.5mm.
- 3 terminales para el multímetro digital.

En su interior contiene circuitos integrados de la marca Texas Instruments que trabajan cooperativamente con el controlador principal de National Instruments, Figura 44.

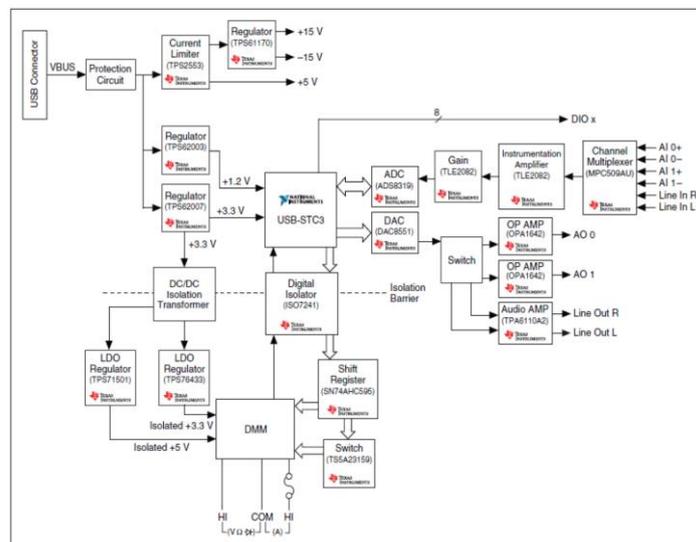


Figura 44. Arquitectura interna de la tarjeta MyDAQ.
Fuente: [7]

2.2.9.2 Entradas Analógicas

MyDAQ tiene dos entradas analógicas diferenciales, las cuales pueden ser configuradas como entradas de tensión diferencial de elevada impedancia de propósito general (+/-10V) o como entradas de audio (señal de audio izquierdo y derecho para un sistema estéreo), Figura 45. Tiene un convertidor ADC de 16bits. En cuanto al proceso de muestreo, puede obtener un total de 200000 muestras por segundo (200 KS/s) [7].

2.3.9.2 Salidas Analógicas

Tiene un convertidor DAC con una resolución de 16bits. Al igual que las Entradas Analógicas, puede muestrear hasta 200000 muestras por segundo. Posee una impedancia de 1Ω en las terminales de salida. Sus dos salidas analógicas están referenciadas a tierra y también pueden ser configuradas como de propósito general (+/- 10V) o de audio (señal de salida estéreo) [7].

2.4.9.2 Entradas/Salidas Digitales

La tarjeta MyDAQ posee un puerto con ocho líneas digitales (DIO). Estas líneas pueden ser configuradas como entradas o salidas de propósito general, como salida de un controlador digital o como entrada o salida de contadores o temporizadores respectivamente (Interfaz de Función Programable, PFI). Las entradas pueden ser hasta de 5VDC TTL, mientras que las salidas son 3.3VDC, es decir LVTTTL (Low Voltage Transistor Transistor Logic). National Instruments no asegura compatibilidad con niveles lógicos CMOS [7].

2.5.9.2 Controladores/Temporizadores

MyDAQ posee un contador/temporizador de 32 bits de resolución que dispone de 5 líneas (DIO 0 – DIO 4, PFI) [7]. Su reloj interno tiene una base de tiempo de 100MHz.

2.6.9.2 Alimentación

Cuenta con tres fuentes de alimentación independientes [7]:

- +/-15V.
- +15V (máximo sin carga +15.3V y mínimo con carga +14V), Imáx 32mA.
- +5V, 100mA (TTL).

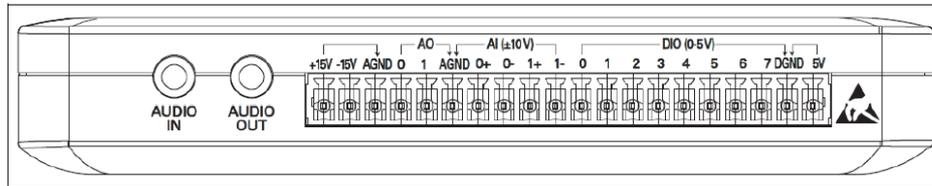


Figura 45. De izquierda a derecha: Puertos de audio, de alimentación, de entradas/salidas analógicas y digitales.
Fuente: [7]

Cabe denotar que la tierra analógica (AGND) es diferente a la tierra digital (DGND), funcionan por separado [7], Figura 45.

2.10 BANCO HRE HIDRAULIC

HRE HIDRAULIC es una empresa dedicada al desarrollo de proyectos ingenieriles en el campo de los fluidos industriales. Además, lleva más de treinta años en el mercado, lo que refleja una amplia experiencia industrial en este ámbito, compaginando la realidad industrial y las nuevas tecnologías. Esta empresa busca satisfacer las necesidades de sus clientes a través de sus dos unidades de negocio complementarias: Industrial y Servicio a la información [38].

En cuanto a educación, HRE HIDRAULIC desarrolla equipos didácticos, específicamente bancos de práctica; actualmente cuenta con tres modelos: HLA90, HLA80 y HLA70. En las Figura 46 y 47 se aprecian las dimensiones y pesos de estos bancos. El principal objetivo del equipo didáctico es fortalecer la literatura que los estudiantes reciben en los centros de formación. Estos equipos proporcionan un soporte al estudiante, para que éste se convierta en un auténtico especialista en la automatización mediante el conocimiento y experiencia en la manipulación de diversos elementos hidráulicos (mandos, accionamientos, regulaciones, etc.) [39].

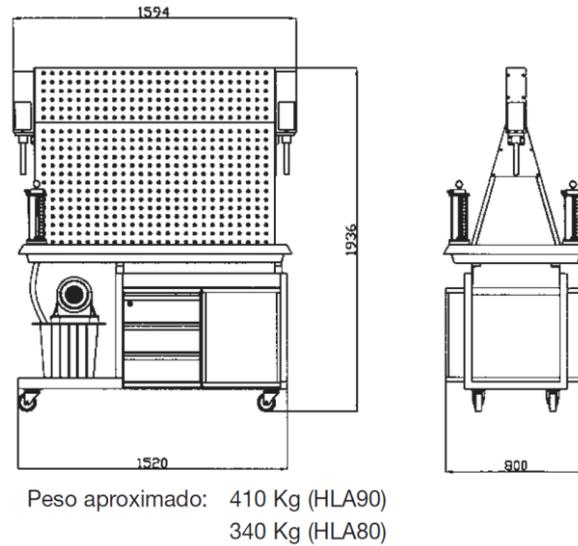


Figura 46. Dimensiones y peso de los bancos HLA90 y HLA80.
Fuente: [39]

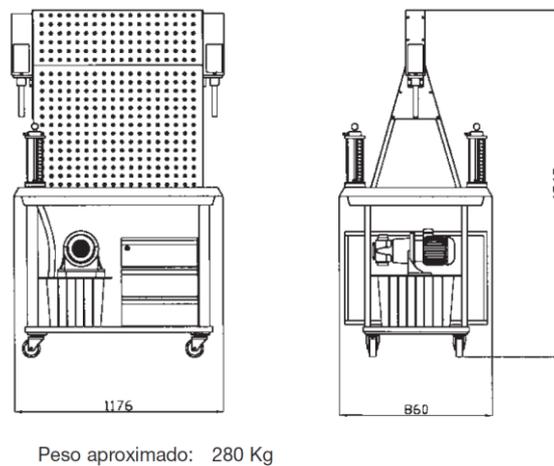


Figura 47. Dimensiones y peso del banco HLA70.
Fuente: [39]

Nota: Para el desarrollo de este proyecto técnico se ha utilizado el banco hidráulico modelo HLA70.

2.1.10 VENTAJAS

Las ventajas que proporcionan estos bancos de trabajo son múltiples, algunas de ellas son [39]:

- Poseen enchufes rápidos auto-obturantes totalmente estancos.
- Sus válvulas son de tamaño nominal (TN6) y sus Bombas son independientes para cada puesto de trabajo.
- Las Bombas son silenciosas (poseen dentado interior).

- Montaje de elementos rápido, cómodo y eficaz, no requiere del uso de herramientas específicas.
- Están incorporados los dispositivos de protección eléctrica, mecánica e hidráulica (cumpliendo con la normativa CE).
- Los bancos pueden ser usados para varias aplicaciones y prácticas, solo requieren de los elementos necesarios.

2.2.10 CARACTERÍSTICAS TÉCNICAS

Los bancos HLA90, 80 y 70 poseen las siguientes características [39]:

- Paneles metálicos totalmente plastificados, de dimensiones 1350 x 1050 mm para los modelos HLA90 y HLA80 y 950 x 1050 mm para el modelo HLA70.
- Grupo hidráulico completo y accesible:
 - Válvulas limitadoras de presión.
 - Visor de nivel de aceite.
 - Refrigerador aire/aceite (disponible únicamente para el modelo HLA90).
 - Filtro de aire.
 - Filtro de retorno.
 - Tapón de llenado del depósito de aceite con filtro de entrada.
 - Manómetros con baño de glicerina.
 - Pulsador de marcha.
 - Pulsador de paro.
- Capacidad de depósito de 120 litros para el modelo HLA90 y de 30 litros para los modelos HLA80 y HLA70.
- Caudal impulsado por Bomba para el modelo HLA90 es de 8 litros/min y de 4.5 litros/min para los modelos HLA80 y HLA70.
- Nivel sonoro aproximado de 60dB a 1m.
- Presión de trabajo típica de 60 bar para HLA90 y de 35 bar para HLA80 y HLA70. (Puede ser ajustada mediante una válvula limitadora de presión).
- Presión máxima para HLA90 es de 100 bar y 60 bar para HLA80 y HLA70.
- Indicadores de presión amortiguados por glicerina (uno por banco de trabajo).
- Motor eléctrico:
 - HLA90: trifásico 380 VCA - 50 Hz de 3Kw

- HLA80 y HLA70: monofásico de 1.2Kw, con condensador de arranque, protección contra inversión de giro. 220 VCA, 50 Hz.
- Interruptor guardamotor automático (viene incluido en la botonera de mando).
- Conexión a red, pulsador de Marcha, Paro y Emergencia en cada puesto de trabajo (corte de suministro eléctrico e hidráulico).
- Tomas de conexión hidráulica 2P y 4T para los modelos HLA90 y HLA80 y 1P y 2T para HLA70.
- Dispositivos de medición, un Recipiente por puesto de 1.2 litros para medidas volumétricas, transparente, con escala graduada y con dispositivo de seguridad anti rebosamiento.
- Aceite HLP46.
- Bandeja metálica para recolección de aceite con retorno directo a tanque y rejilla metálica.
- Ruedas con bloqueo por pedal.

HRE HIDRAULIC cuentan con una gran variedad de elementos que son utilizados para llevar a cabo cualquier práctica (del manual de prácticas) que se desee, estos elementos están disponibles en [39]. El siguiente sustento teórico se concentrará en los elementos principales de interés para este proyecto de tesis: Motor Hidráulico, Transductor, Válvula distribuidora proporcional 4/3.

2.3.10 MOTOR HIDRAÚLICO Y TRANSDUCTOR

El Motor Hidráulico posee las siguientes características [39], [40]:

- De tipo pistones axiales.
- Cubicaje: $12.5 \text{ cm}^3/\text{rev}$.
- Fijación sobre panel mediante tuercas traseras.
- Dos enchufes macho rápidos, R ¼”.
- Presión máxima: 140 bar.

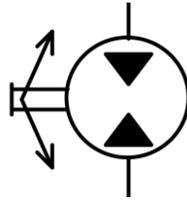


Figura 48. *Símbolo del Motor Hidráulico.*
Fuente: [40]

En la Figura 48 se observa el símbolo del Motor hidráulico, el cual es utilizado en los diagramas de conexión que se encuentran en el manual de prácticas, en [39].

En cuanto al Transductor. Es un elemento electrónico que convierte una señal de un tipo en otro, por ejemplo, transforma una señal física en eléctrica.

Puede clasificarse por su [41]:

- **Fuente de alimentación:** pasivo (requiere de alimentación externa para su operación) o activo (no requiere de dicha alimentación).
- **Señal que proporciona:** analógica (señal que cambia proporcional al efecto que mide) o digital (dos estados lógicos: 0, 1).
- **Variable que capta:** posición, temperatura, presión, velocidad, etc.
- **Fabricación:** mecánicos, resistivos, capacitivos, inductivos, piezoeléctricos, semiconductores.

Específicamente, el Transductor de interés de este proyecto de tesis es el que mide velocidad, estos Transductores pueden ser mecánicos o eléctricos [42]. Para este proyecto se ha utilizado un eléctrico.

Una descripción completa del Transductor usado para este proyecto sería:

Transductor activo eléctrico de velocidad cuyo funcionamiento consiste en medir las revoluciones del Motor Hidráulico y transformar esa medida en una señal eléctrica acotada entre 0 y 10VDC.



Figura 49. Conexión entre el Motor Hidráulico y el Transductor.

En la Figura 49 se aprecia físicamente el acople que existe entre el Motor Hidráulico y el Transductor.

2.4.10 VÁVULA DISTRIBUIDORA PROPORCIONAL 4/3

Esta válvula cuenta con las siguientes características [39], [40]:

- Mando directo, centraje por muelles.
- Posición central: A, B, P y T cerrados.
- Presión máxima: 315 bar.
- Bobinas a 24V, 0 ~ 0.85A.
- Cuatro enchufes macho rápidos, R 1/4".
- Tamaño Normal (TN6).

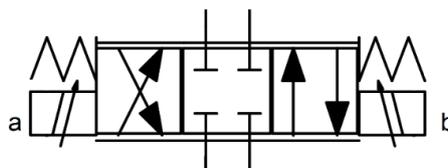


Figura 50. Símbolo de la Válvula distribuidora proporcional 4/3.
Fuente: [40]

El símbolo de Válvula distribuidora proporcional 4/3 se ilustra en la Figura 50; al igual que el símbolo del Motor Hidráulico, éste es utilizado en los diagramas de conexión del manual de prácticas. Por otra parte, en la Figura 51 se observa el aspecto físico de dicha válvula.



Figura 51. Aspecto físico de la Válvula distribuidora proporcional 4/3.

Nota: Para todos los procesos que se describen en capítulos posteriores la bomba de aceite del sistema hidráulico operó a una presión de 47bar.

CAPÍTULO 3: DESARROLLO E IMPLEMENTACIÓN

3.1 CIRCUITO DE POTENCIA

En esta sección se presenta al circuito de potencia junto con sus características y su operación. Debido a las características que posee y satisfacen las necesidades del hardware, se utilizó como Circuito de Potencia el Módulo L298 lógico de accionamiento dual para motor DC, 7A / 160W de grado industrial de alta potencia, Figura 52.

Este módulo cuenta con las siguientes características [43]:

- Doble puente H (7A y alta potencia) el cuál puede operar dos motores DC.
- El rango de voltaje de alimentación es de 6.5-27VDC. Se recomienda usar fusible de 15A cuando se opera a elevadas tensiones y corrientes.
- La interfaz dual de cada motor opera a 7A, 50A pico. Es recomendable usar fusible de 10A en altas tensiones y corrientes.
- Cuenta con dos pines ENABLE, cada uno controla por separado la interfaz de un motor.
- Voltaje de la señal de control 3-6VDC.
- Los pines ENABLE soportan una señal PWM de hasta 10KHz.
- Temperatura de operación -25°C , -80°C .
- La señal GND no puede ser compartida con la PGND (Tierra de alimentación).
- Protección para bajas tensiones (prevención contra altas corrientes instantáneas).
- El pin *IO* está opto acoplado, se puede controlar directamente, sin interferencias.

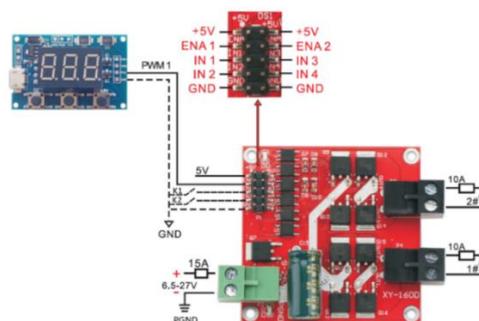


Figura 52. Aspecto físico del Módulo de potencia L298.

Fuente: [43]

De forma general, la operación de este módulo consiste en tomar la señal PWM, que ingresa en el pin ENABLE, y transmitirla a la interfaz del motor con amplitud amplificada; esta amplificación depende de la tensión con la que está siendo alimentado el módulo. En cuanto a la frecuencia de la señal PWM que ingresa al módulo, ésta se mantiene para la interfaz del motor.

Con respecto a los pines IN_1 e IN_2 (para la interfaz del motor uno), éstos controlan el comportamiento de la señal PWM en la interfaz del motor [43]. En la Tabla III se observa las combinaciones posibles de estos pines y el comportamiento que se obtiene en la interfaz.

Tabla III. Lógica de la señal de control para la interfaz del motor uno.

Fuente: [43]

Combinaciones para los pines IN_1 e IN_2 del Módulo L298			
IN_1	IN_2	ENA_1	OUT_1, OUT_2
0	0	X	Detenido
1	1	X	Flotante
1	0	PWM	Marcha hacia adelante
0	1	PWM	Marcha hacia atrás
1	0	1	Marcha hacia adelante (alta velocidad)
0	1	1	Marcha hacia atrás (alta velocidad)

Donde 0 y 1 son 0 y 5VDC respectivamente, y *x* es cualquiera de las dos opciones.

La Tabla III se aplica también para la interfaz del motor dos. Este módulo de potencia forma parte del hardware que vincula el software con el sistema hidráulico, Figura 53.

Como se indicó en la sección 1.5, el módulo toma una señal que es emitida desde el software LabVIEW® (a través de la tarjeta de adquisición MyDAQ y la placa Arduino MEGA 2560) y la transmite al sistema hidráulico amplificando su amplitud.

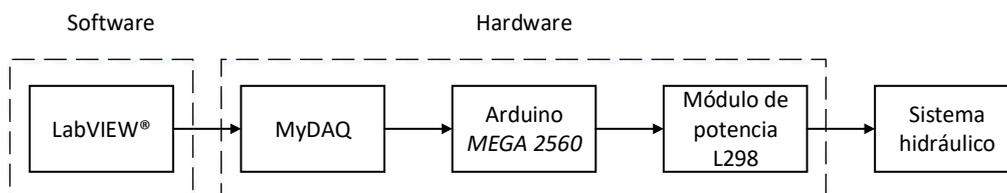


Figura 53. Esquema de la conexión entre el software, hardware y el sistema hidráulico.

En la Figura 53 se observa que la señal proveniente de LabVIEW® pasa a través de la tarjeta MyDAQ y de una placa Arduino, esto es debido a que la generación de la señal PWM en la tarjeta MyDAQ presenta un comportamiento lento eventualmente; este comportamiento podría posteriormente causar retardos en la señal de control.

Es por esta razón que se optó por limitar la señal de control (proveniente de LabVIEW®) entre 0 y 2.5VDC analógicos y enviarla a un puerto analógico del Arduino. Una vez obtenida esta señal en el Arduino se interpretó y convirtió a una señal PWM (mediante software), la cual sería transmitida al pin ENABLE del Módulo de potencia L298. El código implementado en la placa Arduino para este proceso se encuentra en el Apéndice B.

3.2 ESTABILIZADOR DE SEÑAL

En la sección 1.5, específicamente en la Figura 1, se observó que la señal que cierra el lazo de control es la que proviene del Transductor; por lo tanto, es importante que esta señal sea estable (sin variaciones ni ruido) de tal forma que el sistema de control pueda operar sin inconvenientes. En esta sección se expone el proceso desarrollado para obtener dicha señal estable.

Como se mencionó en la sección anterior, la placa Arduino es la encargada de emitir una señal PWM, a una frecuencia determinada, hacia el módulo de potencia. Para determinar esta frecuencia se llevaron a cabo pruebas de funcionamiento, esto debido a que el Motor Hidráulico presenta bajas o altas vibraciones dependiendo de la frecuencia de la señal PWM a la que está operando. Estas vibraciones se ven reflejadas como ruido en la señal que produce el Transductor; esto sucede debido al acople mecánico al que está sometido éste con el Motor Hidráulico, Figura 49.

Las pruebas de funcionamiento se llevaron a cabo en tres condiciones, cuando el Motor Hidráulico había llevado un tiempo de operación: corto, medio y largo.

Estas pruebas consistieron en:

- Muestrear la señal que genera el Transductor (mediante la tarjeta MyDAQ y un VI de LabVIEW®) mientras se varía la frecuencia de la señal PWM cuando el Motor Hidráulico esté operando en su máxima velocidad.

El VI desarrollado para este proceso se indica en la sección 3.5.

- Calcular el valor RMS de la señal obtenida en el punto anterior mediante el software MATLAB®.

El código desarrollado para este proceso se adjunta en el Apéndice C.

Se optó por trabajar con el valor RMS de la señal del Transductor debido a que este valor presenta de forma cuantitativa las variaciones de dicha señal en torno a su media. Cabe mencionar que para realizar estas pruebas se utilizó el módulo variador de frecuencia con el que cuenta propiamente el banco hidráulico; el rango de frecuencias que posee este módulo va desde los 30Hz hasta los 277Hz [39].

Los resultados obtenidos se observan en las siguientes figuras:

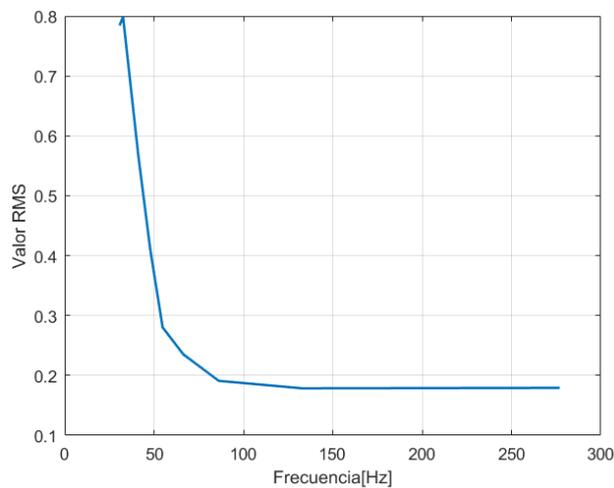


Figura 54. Valores RMS obtenidos en el tiempo de operación corto.

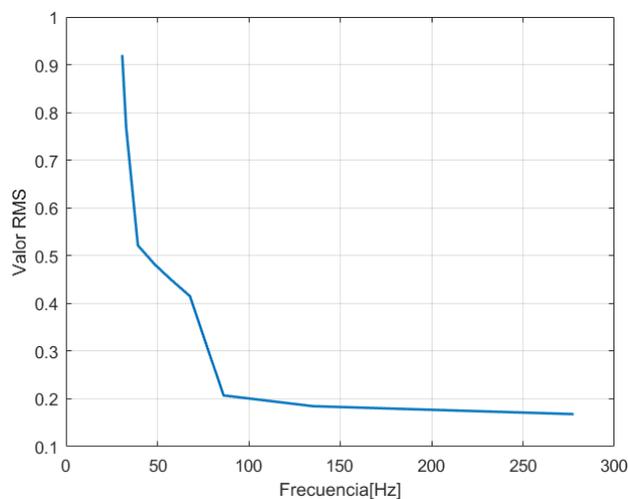


Figura 55. Valores RMS obtenidos en tiempo de operación medio.

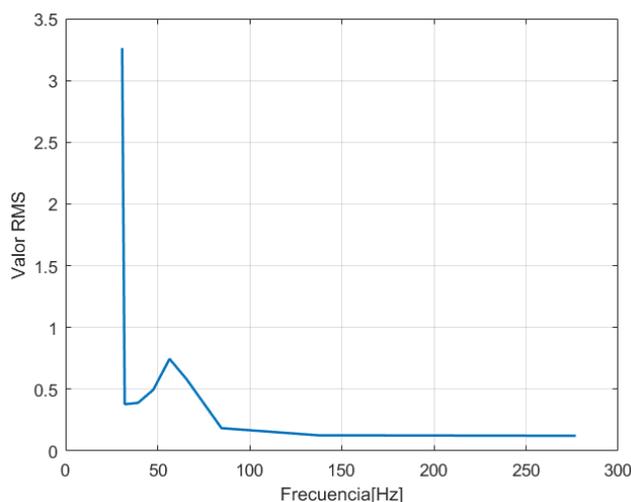


Figura 56. Valores RMS obtenidos en tiempo de operación largo.

Los valores RMS que se observan en las figuras anteriores se obtuvieron de la señal resultante de la resta entre la señal analizada y su media (en unidades de voltaje). En estas mismas figuras, se observa que el valor RMS más bajo se obtiene cuando la frecuencia de la señal PWM se encuentra desde los 100Hz en adelante. También se aprecia un incremento del valor RMS alrededor de los 40Hz-70Hz, cuando la máquina había operado un tiempo medio y largo, Figuras 55 y 56 respectivamente; este comportamiento se debe a que el motor incrementaba sus vibraciones en estos tiempos de operación.

Luego de haber determinado la frecuencia idónea para la señal PWM se procedió a programar en la placa Arduino, esto con el fin de obtener dicha frecuencia en sus pines destinados a PWM. Por defecto la frecuencia de la señal PWM que produce el Arduino *MEGA 2560* es de 490Hz y 977Hz (dependiendo de los pines destinados a PWM), por lo tanto, es necesario modificar la frecuencia (de cualquiera de los pines destinados a PWM) mediante el cambio de factor de división de frecuencia, así como se especifica en [44].

Una vez concluida la programación se procedió a realizar las pruebas de funcionamiento del Arduino junto con el módulo de potencia. La frecuencia de la señal PWM obtenida es de 122Hz, tal y como se observa en la Figura 57. Se ha elegido esta frecuencia para la señal PWM y no la que viene por defecto en la placa Arduino debido a que cuando se incrementaba o reducía la velocidad del motor el cambio se producía de forma un poco brusca; este comportamiento, posteriormente, afectaría al proceso de control. Además, en el módulo de operación del banco hidráulico se recomienda trabajar en las frecuencias que ofrece el módulo variador de frecuencia (30Hz-277Hz) [39].

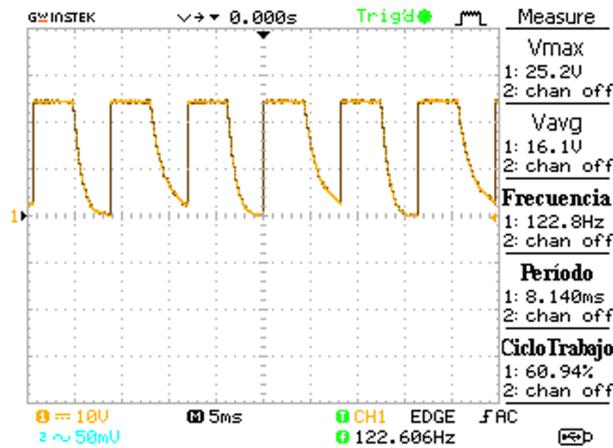


Figura 57. Señal PWM configurada en el Módulo de potencia L298 sin carga.

Es importante denotar que la señal PWM expuesta en la Figura 57 corresponde a la señal que genera el módulo de potencia cuando opera sin carga (sin conectar la válvula proporcional). Esta señal no presenta una forma totalmente cuadrada; sin embargo, ésta cambia su forma a cuadrada cuando la válvula proporcional se conecta al módulo de potencia, Figura 58.

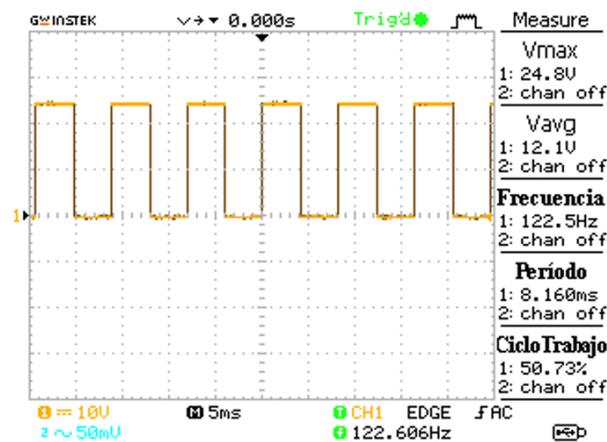


Figura 58. Señal PWM configurada en el Módulo de potencia L298 con carga.

Luego de haber determinado e implementado la frecuencia de la señal PWM, se procedió a muestrear y tratar la señal del Transductor. En la Figura 59 se observa la señal (en voltios) que produce el Transductor. Como se evidencia, esta señal cuenta con la presencia de ruido, el cual produce variaciones en la lectura de la magnitud del voltaje.

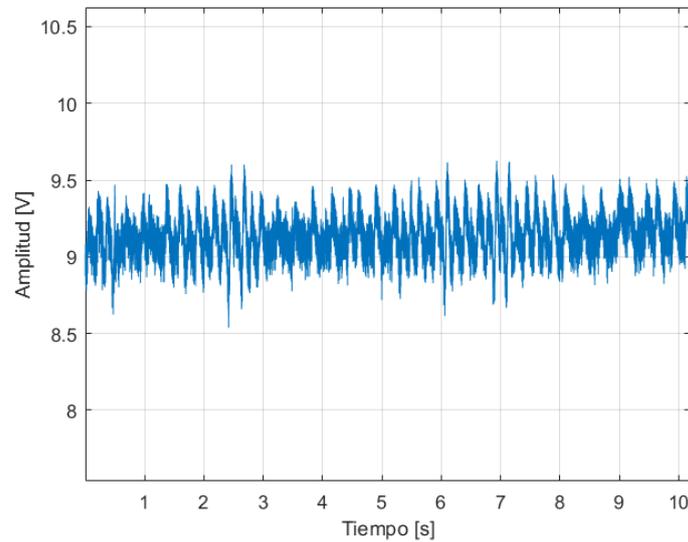


Figura 59. Señal producida por el Transductor en voltios.

Con el fin de determinar a qué frecuencia se encuentra este ruido se procedió a aplicar una Transformada Rápida de Fourier (*FFT*, por sus siglas en inglés) a la señal de la Figura 59, para posteriormente aplicar una técnica que ayude a reducir su presencia.

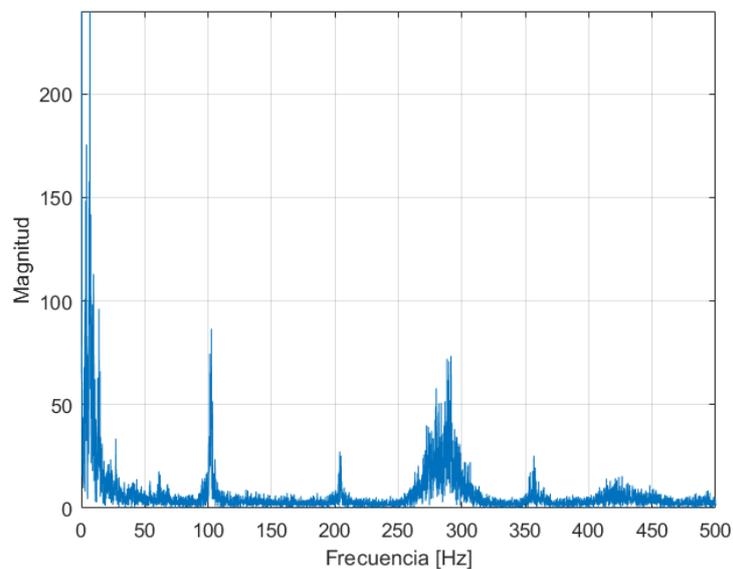


Figura 60. Espectro de la señal ruidosa del Transductor.

En la Figura 60 se evidencia el espectro en frecuencia (magnitud) de la señal de la Figura 59 luego de haber aplicado una *FFT*. En esta figura se observa que en las frecuencias de 100Hz y aproximadamente 300Hz se localizan picos pronunciados. En base a esta identificación se optó por llevar a cabo tres métodos que ayuden a reducir o eliminar la presencia de ruido, para posteriormente utilizar uno de éstos para estabilizar la señal del Transductor. Estos métodos fueron:

- Filtros pasivos paso bajo.
- Filtros digitales paso bajo.
- Promedio móvil.

En primera instancia se implementaron filtros pasivos paso bajo con una frecuencia de corte (F_c) menor a los 100Hz.

Se diseñaron dos filtros pasivos paso bajo, uno con $F_c = 10\text{Hz}$ y otro con $F_c = 50\text{Hz}$, los cálculos se indican a continuación:

$$F_c = \frac{1}{2 * \pi * R * C} \quad (25)$$

$$R = \frac{1}{2 * \pi * F_c * C} \quad (26)$$

Aplicando la ecuación (26) para $F_c = 50\text{Hz}$ y $C = 4.7\mu\text{F}$.

$$R = \frac{1}{2 * \pi * 50 * 4.7 * 10^{-6}} = 677.255\Omega$$

El valor comercial más próximo es de $R = 680\Omega$.

Para $F_c = 10\text{Hz}$ y $C = 4.7\mu\text{F}$.

$$R = \frac{1}{2 * \pi * 10 * 4.7 * 10^{-6}} = 3386.275\Omega$$

El valor comercial más próximo es de $R = 3.3\text{K}\Omega$.

Posteriormente se implementaron estos filtros al sistema (Figura 53) y se desarrollaron pruebas de funcionamiento. Los resultados obtenidos se exponen en las siguientes figuras:

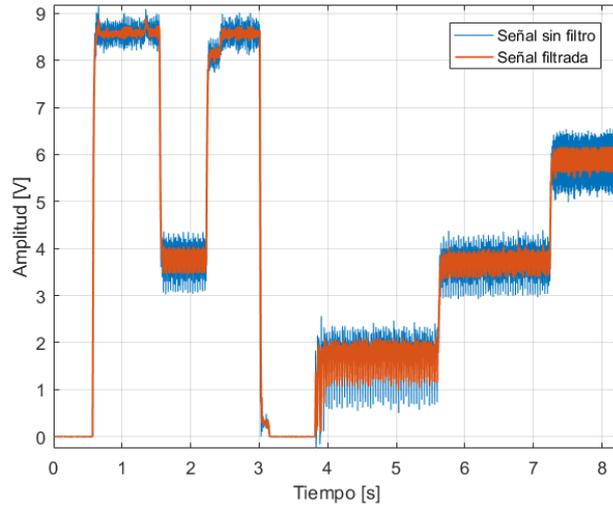


Figura 61. Señal del Transductor filtrada con $F_c = 10\text{Hz}$.

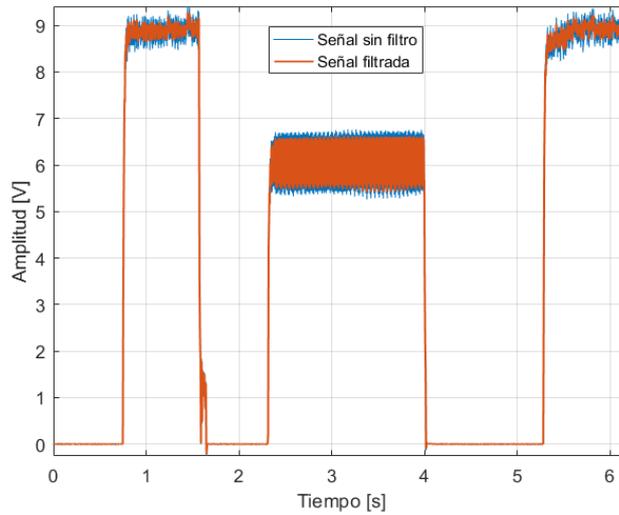


Figura 62. Señal del Transductor filtrada con $F_c = 50\text{Hz}$.

En las Figuras 61 y 62 se observa que los filtros implementados cumplen su objetivo y reducen el ruido de la señal del Transductor; sin embargo, el filtro con $F_c = 10\text{Hz}$ presenta mejores resultados de filtrado que el filtro con $F_c = 50\text{Hz}$. En cuanto a los tiempos de desfase entre la señal sin filtrar y la filtrada, en la Tabla IV se indican la F_c del filtro y la cantidad de desfase que produce éste.

Tabla IV. Tiempos de desfase entre la señal filtrada y sin filtrar.

Tabulación de tiempos de desfase	
F_c (Hz)	Tiempo de desfase (s)
10	0.021
50	0.01

Como se observa en la tabla anterior el filtro pasivo paso bajo con $F_c = 10Hz$, a pesar de ofrecer un mejor resultado de filtrado, genera un mayor tiempo de desfase que el filtro con $F_c = 50Hz$.

Luego de esto se procedió con el diseño e implementación de filtros digitales paso bajo; para esto se tomó como referencia el espectro de la señal del Transductor, Figura 60, para posteriormente diseñar cuatro filtros digitales paso bajo con la ayuda del software MATLAB®. Los filtros que se diseñaron fueron:

- Butterworth.
- Chebyshev Tipo 1.
- Chebyshev Tipo 2.
- Elíptico.

Para el diseño de estos filtros se consideró:

- Atenuación en la banda de paso (R_p) de 0.1dB.
- Atenuación en la banda de rechazo (R_s) de 40dB.
- F_c en 10Hz y en 50Hz (valores tomados en base al análisis de la Figura 60).
- F_s de 30ms (valor que se obtuvo experimentalmente y cuyo proceso se detalla en la sección 3.5.5).

El código implementado en MATLAB® para desarrollar este proceso se adjunta en el Apéndice D.

Cabe mencionar que en primera instancia se procedió a analizar y filtrar la señal de la Figura 60 en el software MATLAB®, para posteriormente aplicar los filtros digitales al sistema mediante el software LabVIEW®. Los resultados de las pruebas se evidencian en las siguientes figuras:

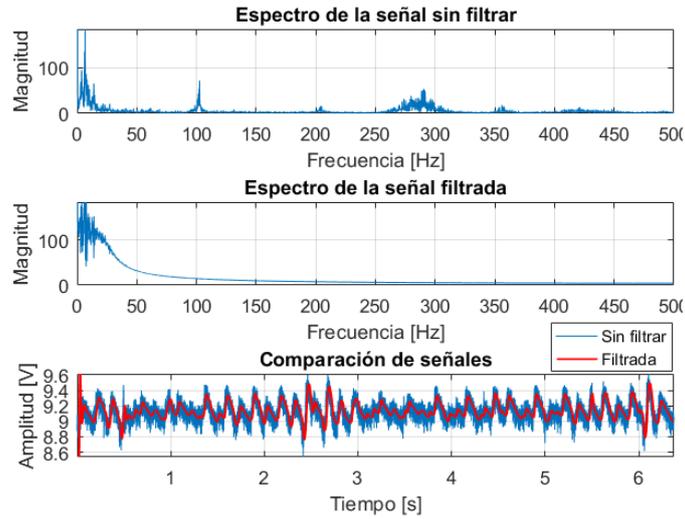


Figura 63. Filtrado de la señal del Transductor mediante el filtro Butterworth con $F_c = 10\text{Hz}$.

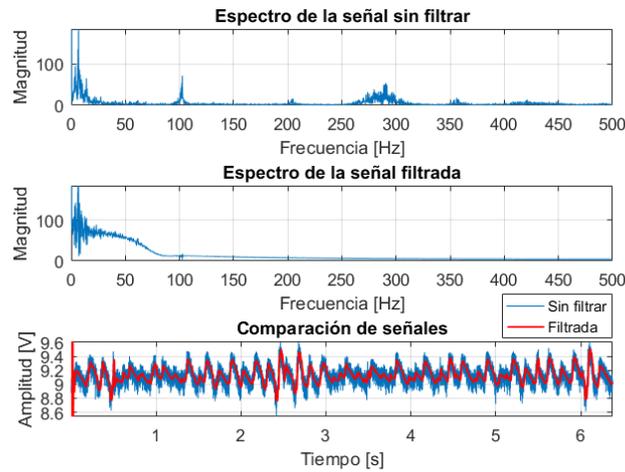


Figura 64. Filtrado de la señal del Transductor mediante el filtro Butterworth con $F_c = 50\text{Hz}$.

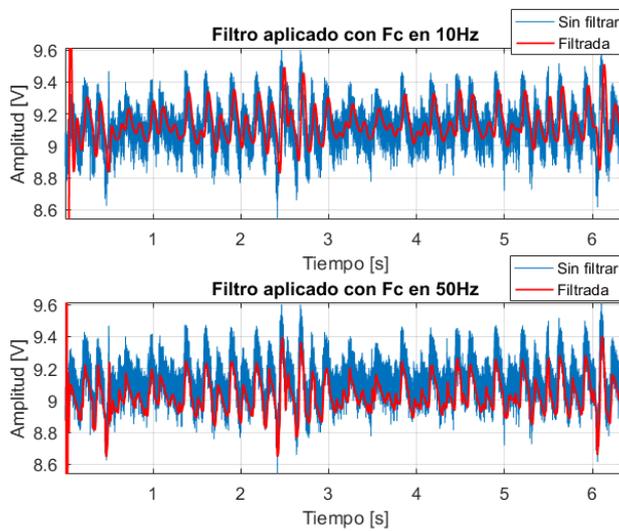


Figura 65. Filtrado de la señal del Transductor mediante el filtro Chebyshev Tipo 1.

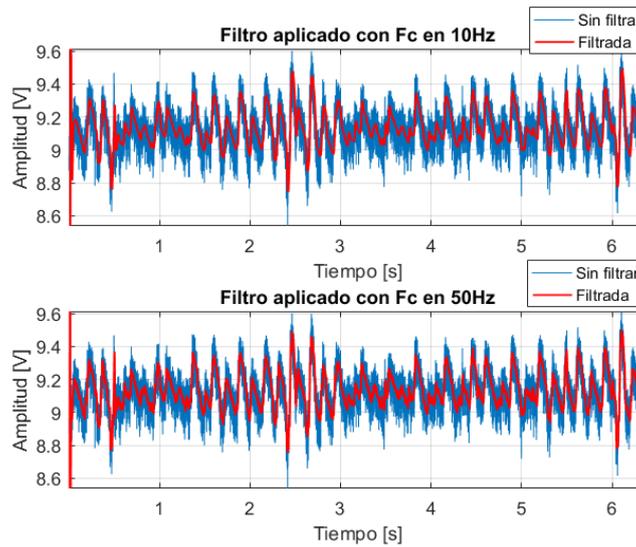


Figura 66. Filtrado de la señal del Transductor mediante el filtro Chebyshev Tipo 2.

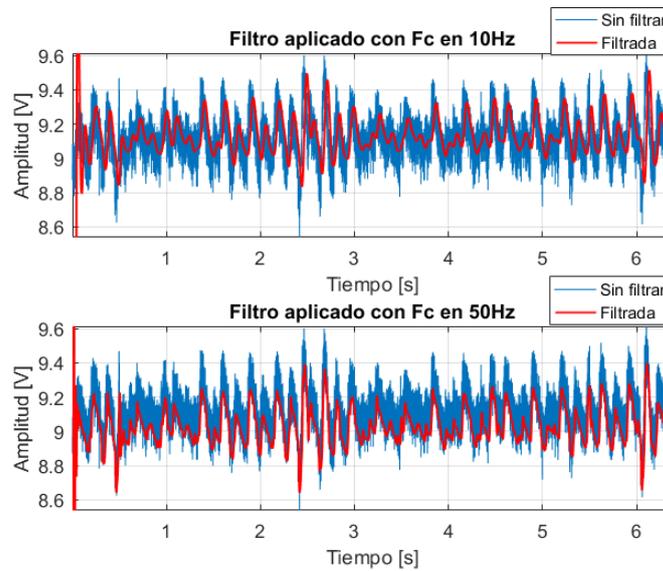


Figura 67. Filtrado de la señal del Transductor mediante el filtro Elíptico.

En la Figuras 63 y 64 se observa el proceso (mencionado anteriormente) que se llevó a cabo para el filtrado de la señal ruidosa del Transductor. Este mismo análisis se realizó con el resto de los filtros digitales; sin embargo, solo se agregaron las figuras en donde se comparan la señal sin filtro con la filtrada en el dominio del tiempo.

De forma general, la implementación de estos filtros digitales ayudaron a reducir la presencia de ruido en la señal del Transductor; a pesar de esto, existen casos particulares en donde la señal filtrada presenta un offset con respecto a la señal sin filtrar, específicamente esto sucede en los filtros Chebyshev Tipo 1 (Figura 65) y Elíptico (Figura 67) cuando $F_c = 50\text{Hz}$ para ambos casos.

También, en la Figura 66 (filtro Chebyshev Tipo 2), la señal filtrada aún cuenta con ruido (en menor proporción), esto cuando $F_c = 50\text{Hz}$. En cuanto al orden de los filtros digitales, obtenidos del proceso de diseño, los podemos encontrar en la Tabla V.

Tabla V. Orden de los filtros digitales.

Tabulación del orden de los filtros digitales		
F_c (Hz)	Filtro	Orden
10	Butterworth	3
	Chebyshev Tipo 1	3
	Chebyshev Tipo 2	3
	Elíptico	3
50	Butterworth	6
	Chebyshev Tipo 1	4
	Chebyshev Tipo 2	4
	Elíptico	4

Luego de probar el rendimiento de estos filtros en el software MATLAB® se procedió a implementarlos al sistema mediante el software LabVIEW®. Los VI desarrollados para este proceso se encuentran detallados en la sección 3.5. Es importante denotar que el software LabVIEW® no cuenta con el filtro Chebyshev Tipo 2, por lo que este filtro no se consideró para las pruebas de funcionamiento en el sistema.

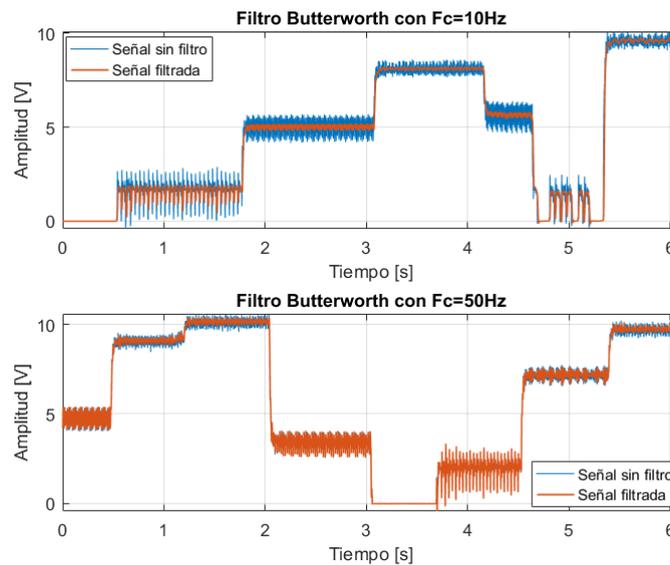


Figura 68. Aplicación del filtro Butterworth con $F_c = 10\text{Hz}$ y 50Hz .

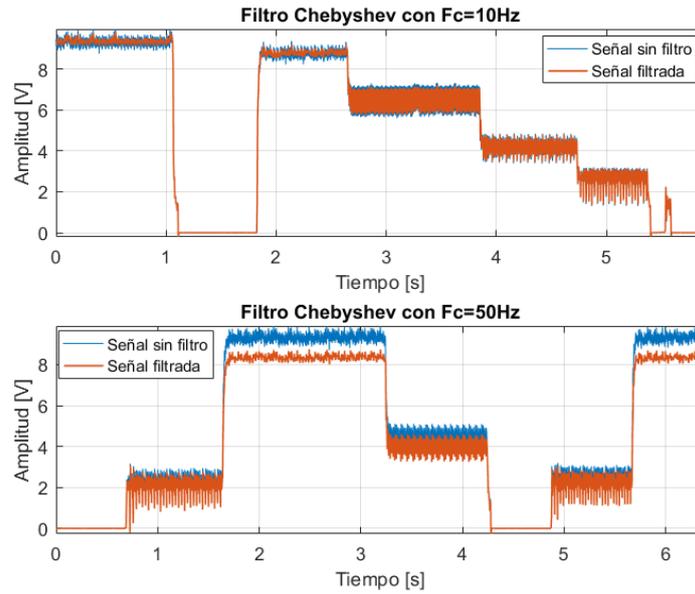


Figura 69. Aplicación del filtro Chebyshev Tipo 1 con $F_c = 10\text{Hz}$ y 50Hz .

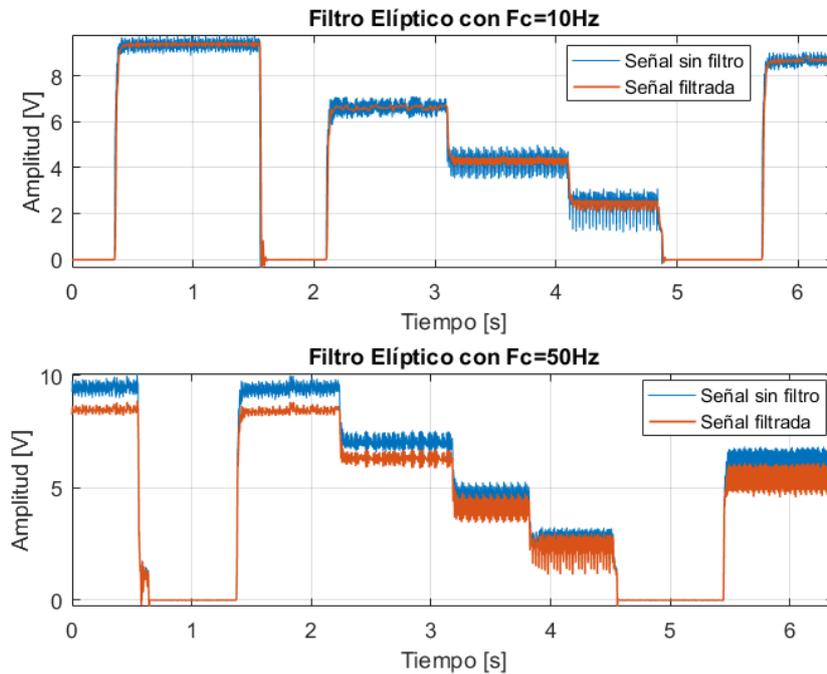


Figura 70. Aplicación del filtro Elíptico con $F_c = 10\text{Hz}$ y 50Hz .

En las Figuras 69 y 70, cuando $F_c = 50\text{Hz}$ para los filtros Chebyshev Tipo 1 y Elíptico respectivamente, se evidencia que el offset que se observó en las Figuras 65 y 67 se mantiene. En las mismas figuras, la frecuencia de corte del filtro Butterworth (Figura 68) no logra filtrar el ruido de la señal del Transductor. Por el contrario (en esta misma figura), cuando $F_c = 10\text{Hz}$ el filtro logra reducir considerablemente el ruido de la señal.

Esta reducción de ruido (en la misma frecuencia de corte) también sucede con el filtro Elíptico, mientras que con el filtro Chebyshev de Tipo 1 no.

En cuanto al desfase que existe entre la señal filtrada y la señal sin filtro, en la Tabla VI se especifican estos tiempos.

Tabla VI. Tiempos de desfase de los filtros digitales.

Tabulación de filtros digitales		
F_c (Hz)	Filtro	Desfase (s)
10	Butterworth	0.017
	Chebyshev Tipo 1	0.011
	Elíptico	0.035
50	Butterworth	0.02
	Chebyshev Tipo 1	0.015
	Elíptico	0.012

Pese a que los filtros digitales ofrecen menores tiempos de desfase que los filtros pasivos paso bajo (tal y como se aprecian en las Tablas V y VI), el filtrado del ruido no ofrece resultados alentadores, exceptuando el caso de los filtros Butterworth y Elíptico cuando $F_c = 10\text{Hz}$, Figuras 68 y 70. Además, es importante mencionar que durante las pruebas de funcionamiento de estos filtros la frecuencia de muestreo se veía afectada y existía retardos eventuales durante la adquisición de la señal. Estos retardos podrían afectar el proceso de control, debido a que su tiempo de respuesta podría incrementarse, complicando de esta manera el control de velocidad. Es debido a esta razón que este método para la supresión o reducción de ruido en la señal del Transductor quedó descartado.

Finalmente, como último método para la disminución del ruido en la señal del Transductor, se utilizó el bloque mean con el que cuenta LabVIEW®; este bloque se basa en el concepto del promedio móvil. A continuación, se expone un breve sustento teórico a cerca de esto.

Dentro de los sistemas en tiempo discreto existe uno muy conocido que maneja la representación de promedio o media móvil. Este sistema evalúa la n -ésima muestra de la sucesión de la variable de salida $y[n]$, como la media de muestras de la sucesión de la variable de entrada $x[n]$ en torno a la n -ésima muestra. Se define por la siguiente ecuación [45]:

$$y[n] = \frac{1}{M_1 + M_2 + 1} \{x[n + M_1] + x[n + M_1 - 1] + \dots + x[n] + x[n - 1] + \dots + x[n - M_2]\}$$

$$y[n] = \frac{1}{M_1 + M_2 + 1} \sum_{k=-M_1}^{M_2} x[n - k] \quad (27)$$

Donde $M1$ y $M2$ son los límites o extremos entre los cuáles se requiere calcular la media. En la Figura 71, se muestra una sucesión de entrada graficada como una función $x[n]$; para $n = 7$, $M1 = 0$ y $M2 = 5$ las muestras involucradas se encuentran entre las líneas verticales entrecortadas [45].

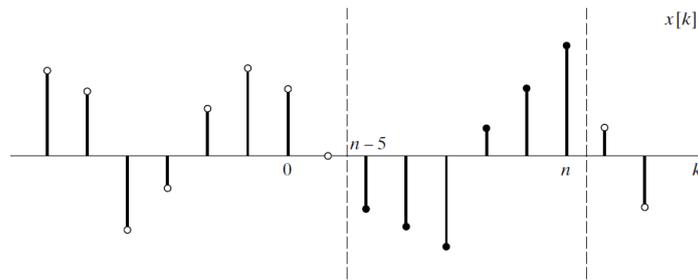


Figura 71. Cálculo de la media móvil con $M1 = 0$ y $M2 = 5$ para $x[n]$.
Fuente: [45]

Por ejemplo, si se desea encontrar $y[7]$ haciendo uso de la ecuación (27) se obtiene:

$$y[n] = \frac{1}{0 + 5 + 1} \sum_{k=0}^5 x[7 - k]$$

$$y[n] = \frac{1}{6} \{x[7 - 0] + x[7 - 1] + x[7 - 2] + x[7 - 3] + x[7 - 4] + x[7 - 5]\}$$

$$y[n] = \frac{1}{6} \{x[7] + x[6] + x[5] + x[4] + x[3] + x[2]\}$$

Se puede observar que la media es $1/6$ de la n -ésima muestra ($y[7]$) multiplicado por todas las muestras en torno a la misma.

Luego de comprender como funciona el bloque mean se desarrolló un VI en el software LabVIEW® para posteriormente aplicarlo al sistema y realizar pruebas de funcionamiento. La estructura de este VI se indica en la sección 3.5. Los resultados obtenidos de las pruebas de funcionamiento se observan en las siguientes figuras:

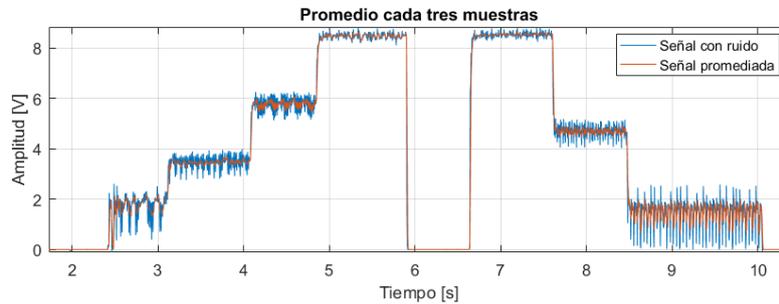
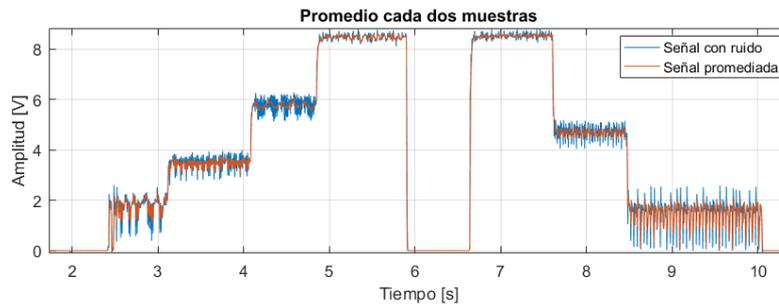


Figura 72. Señal del Transductor promediada cada dos y tres muestras.

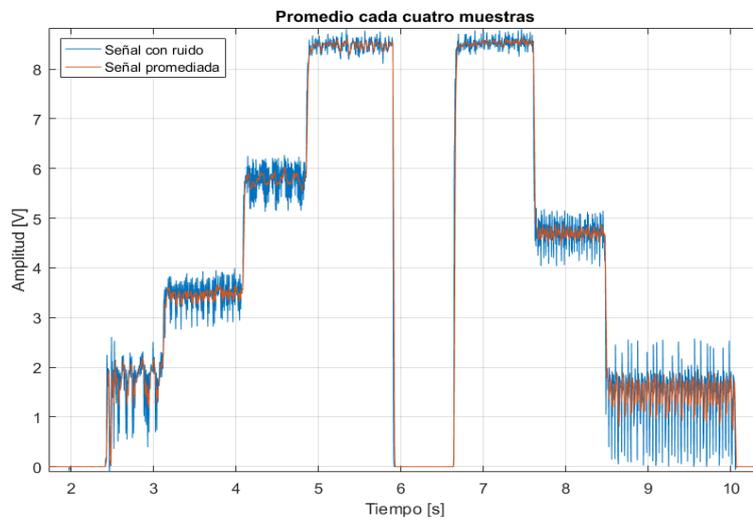


Figura 73. Señal del Transductor promediada cada cuatro muestras.

En la Figura 72 y 73 se observa como el proceso de promediado reduce el ruido en la señal del Transductor. Contrastando estas figuras se observa que el ruido se reduce en mayor proporción cuando el promediado se realiza cada cuatro muestras; sin embargo, cuando se incrementa la cantidad de muestras para el promediado el tiempo de desfase también aumenta, refiérase a la Tabla VII.

Tabla VII. Tiempos de desfase entre la señal promediada y sin promediar.

Tabulación de tiempos de desfase de señales promediadas	
Promedio (#muestras)	Tiempo de Desfase (s)
2	0.02
3	0.025
4	0.055

Comparando los valores de la Tabla VII con los obtenidos para el filtro pasivo paso bajo (Tabla IV) se observa que el promedio de la señal cada tres muestras tiene un tiempo de desfase aproximadamente igual al del filtro pasivo paso bajo con $F_c = 10\text{Hz}$.

Con base en esta información se pusieron a prueba nuevamente al filtro pasivo paso bajo con $F_c = 10\text{Hz}$ y al promediado de la señal del Transductor cada tres muestras; el resultado de esta prueba se observa en la Figura 74.

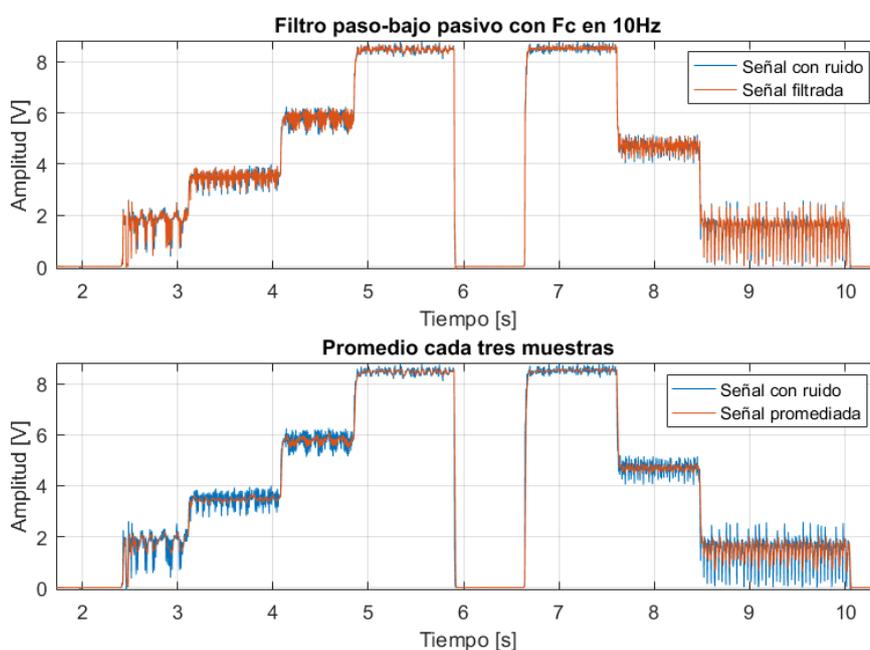


Figura 74. Comparación entre el proceso de filtrado y promediado de la señal del Transductor.

En la Figura 74 se evidencia como el proceso de promediado cada tres muestras ofrece un resultado más satisfactorio en la reducción del ruido que el proceso de filtrado. Por lo tanto, con base a estos resultados obtenidos se optó por usar el bloque mean (promedio cada tres muestras) que posee propiamente el software LabVIEW® para la reducción del ruido en la señal del Transductor, obteniendo de esta manera una señal estable que completa el lazo de control.

Las características de esta señal estable obtenida son aceptables: desfase de 25ms y poca presencia de ruido, esto considerando que se está trabajando con un sistema hidráulico en donde sus vibraciones producen ruido mecánico y eléctrico.

3.3 IMPLEMENTACIÓN DEL SISTEMA AL BANCO HRE HIDRÁULIC

Luego de que se estabilizara la señal que cierra el lazo de control se procedió a implementar el sistema al banco hidráulico. En esta sección se presenta los pasos desarrollados para llevar a cabo dicha implementación.

Como punto de partida se determinó la relación voltaje-velocidad en el Transductor, esto debido a que su señal (realimentación para el lazo de control) debe proporcionar una magnitud de velocidad más no de tensión. Para obtener dicha relación se consideró al Transductor como un motor DC, esto debido a su comportamiento funcional; en base a esta consideración se hizo uso de la siguiente relación:

$$K = \frac{V}{\omega} \quad (28)$$

Donde:

K Constante que relaciona el voltaje (en voltios) en el Transductor y la velocidad angular (en rpm) en su eje.

V Tensión en el Transductor (generado/alimentado).

ω Velocidad angular en el eje del Transductor (en rpm).

Esta relación fue tomada de [46], allí los autores llevan a cabo un proceso experimental para determinar los parámetros y constantes de un motor DC. La ecuación (28) indica que, si se conoce tanto la velocidad angular en el eje del Transductor como la tensión en sus bornes se puede determinar la constante K . Con esta información se procedió a aplicar tensión en los bornes del Transductor y a medir la velocidad angular en su eje, Figura 75.

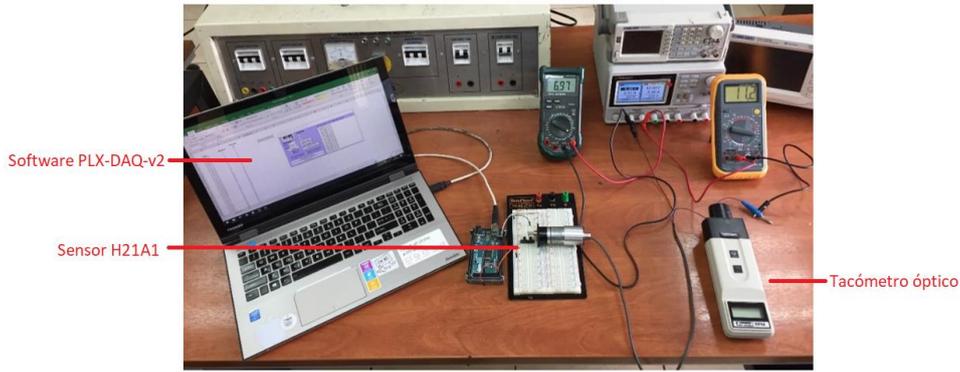


Figura 75. Herramientas utilizadas para la obtención de la constante K .

La tensión aplicada en los bornes del Transductor se midió mediante un Multímetro, mientras que la velocidad angular se obtuvo mediante un tacómetro óptico y un sensor óptico tipo herradura $H21A1$ conectado a un Arduino $MEGA 2560$, el cual previamente fue cargado con un programa que permitía medir los pulsos que produce el sensor óptico. El código se adjunta en Apéndice E. Se utilizaron estos dos métodos para medir la velocidad angular con la finalidad de comparar los valores obtenidos y aproximar mayormente la relación voltaje-velocidad. Los resultados obtenidos se indican en las siguientes tablas:

Tabla VIII. Valores medidos con el sensor óptico $H21A1$.

Tabulación de valores medidos de H21A1		
Voltaje [V]	Velocidad [RPM]	K
1	42	0,02380952
2	84	0,02380952
3	128	0,0234375
4	159	0,02515723
5	216	0,02314815
6	259	0,02316602
7	304	0,02302632
8	348	0,02298851
9	395	0,02278481
10	436	0,02293578
<i>Kprom</i>		0,02342634

Tabla IX. Valores medidos con el tacómetro óptico.

Tabulación de valores medidos de tacómetro óptico		
Voltaje [V]	Velocidad [RPM]	K
1	48	0,02083333
2	85	0,02352941
3	123	0,02439024
4	167	0,0239521
5	211	0,02369668
6	253	0,02371542
7	296	0,02364865
8	342	0,02339181
9	383	0,02349869
10	427	0,0234192
	<i>Kprom</i>	0,02340755

El valor de K , indicado en las tablas anteriores, se obtuvo mediante la ecuación (28). Promediando los valores K_{prom} obtenidos en las Tablas VIII y IX se consigue un valor final de $K = 0,02341695 \approx 0.0235$.

Reordenando y reemplazando el valor de K en la ecuación (28) se tiene que:

$$\omega = \frac{V}{K}$$
$$\omega = \frac{V}{0.0235} \quad (29)$$

La ecuación (29) indica que la velocidad angular del Motor Hidráulico se puede obtener midiendo la tensión en los bornes del Transductor y dividiéndola para el valor de la constante K .

Debido a que estas pruebas se realizaron sin carga en el eje del Transductor la ecuación (29) debe ajustarse cuando se acople el Transductor al Motor Hidráulico, tal y como se observa en la Figura 49. La ecuación ajustada es:

$$\omega = \frac{V}{4*K} \quad (30)$$

Esta ecuación se ajustó de forma experimental.

Luego de obtener esta relación se realizaron pruebas de funcionamiento para determinar el error que existe entre el valor de la velocidad real y la medida. Para obtener

el valor real de la velocidad se usó el tacómetro con el que cuenta el banco hidráulico, mientras que para obtener el valor medido se utilizó la tarjeta de adquisición MyDAQ y un VI desarrollado en el software LabVIEW®. Los resultados obtenidos se indican en la Tabla X.

Tabla X. Determinación del error entre el valor real y medido en la velocidad del Motor Hidráulico.

Tabulación del error absoluto y relativo			
Tacómetro (rpm)	VI-LabVIEW® (rpm)	Error absoluto (rpm)	Error Relativo %
50	49,7	0,3	0,604
86	85,5	0,5	0,585
107	106,55	0,45	0,422
78	77,63	0,37	0,477
51	50,83	0,17	0,334
30	30,16	0,16	0,531
16	16,15	0,15	0,929
60	59,73	0,27	0,452
101	100,46	0,54	0,538
25	25,15	0,15	0,596
39	38,79	0,21	0,541
98	97,43	0,57	0,585
83	82,62	0,38	0,460
57	56,69	0,31	0,547
43	42,81	0,19	0,444
	Promedio	0.315	0,536

Revisando los valores obtenidos en la Tabla X, en la columna del error absoluto, se puede observar que ningún valor supera la unidad; además, el error absoluto en promedio es de apenas 0.315 rpm. Así mismo en la columna del error relativo se aprecian resultados alentadores. Con base a estos datos se concluye que la relación voltaje-velocidad obtenida es suficiente para trabajar con el controlador difuso de velocidad. Una vez obtenida esta relación se procedió con la implementación del sistema en conjunto (software-hardware-sistema hidráulico), Figura 76.

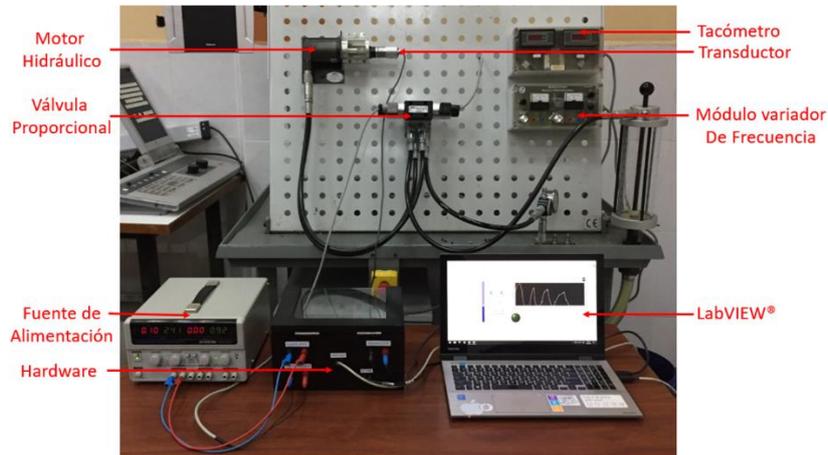


Figura 76. Sistema en conjunto implementado.

En la figura anterior se observa las conexiones realizadas para enlazar el hardware y el software, y estos dos a la vez al sistema hidráulico. En la caja, ubicada entre la fuente de alimentación y el ordenador (en donde se ejecuta el software LabVIEW®) se encuentran instaladas las tarjetas Arduino *MEGA 2560* y MyDAQ junto con el Módulo de Potencia L298, Figura 77b.

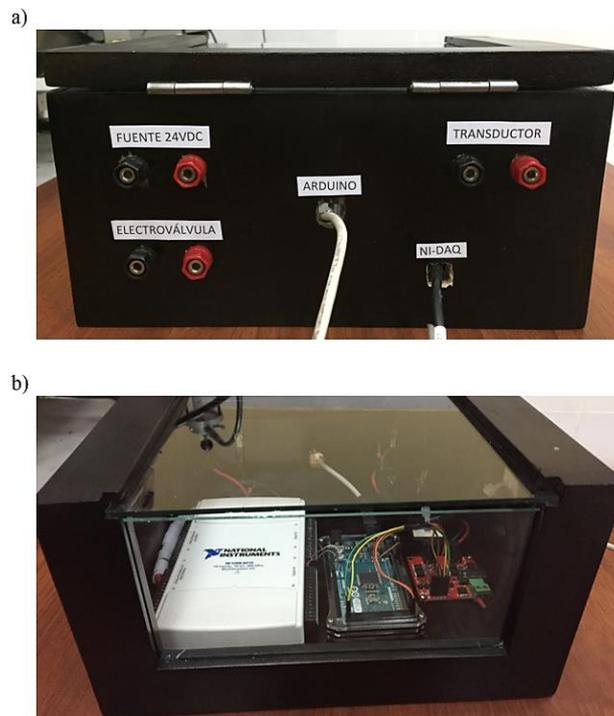


Figura 77. Distribución de la caja que contiene el hardware del sistema en conjunto.

Las dimensiones y el diseño de la caja se adjuntan en el Apéndice F.

En cuanto a la Figura 77a, en ésta se aprecian las borneras instaladas para: alimentar el Módulo de Potencia L298 (Fuente 24VDC), conectar la válvula proporcional (Electroválvula) y el Transductor. Así mismo se observan las aberturas dedicadas para la tarjeta Arduino y MyDAQ.

3.4 DESARROLLO DE REGLAS DIFUSAS

En esta sección se presenta el proceso que se llevó a cabo para el desarrollo de las reglas difusas en el TOOLBOX FUZZY que pertenece al software MATLAB®.

En primera instancia, previo al desarrollo de las reglas difusas se realizaron pruebas de operación con el fin de conocer el comportamiento del Motor Hidráulico junto con el resto del sistema implementado, Figura 76. Además, estas pruebas sirvieron para delimitar los valores máximos y mínimos de las variables de entrada que poseerán los controladores difusos: error (e), cambio del error (derivada: $\frac{de}{dt}$) y la acumulación del error (integral: $\int e * dt$).

Estos valores son:

- Error [-100 100]
- Error derivativo [-70 70]
- Error integrativo [-800 800]

Además, como se especificó en la sección 3.1, la señal de control está limitada entre 0 y 2.5VDC.

Es importante denotar que para este lazo de control se comprende por error a la diferencia que existe entre la señal de referencia r (nivel de velocidad en el que se desea que opere el Motor Hidráulico) y la señal de realimentación y (nivel de velocidad al que está operando el Motor Hidráulico). Luego de esto se procedió con el diseño de los controladores difusos. Se diseñaron un total de tres controladores difusos:

- Proporcional Derivativo (PD).
- Proporcional Integrativo (PI).
- Proporcional Derivativo + Integrativo (PD+I).

Para el diseño de cada uno de estos controles se tomaron las siguientes consideraciones:

- Defuzificación mediante el método del centroide.
- Inferencia mediante el operador *min* (Mandani).
- Agregación mediante el operador *max*.

En cuanto a las funciones de membresía, para su diseño se tomaron en cuenta las observaciones y sugerencias propuestas en [47]:

- Elegir un número adecuado de funciones de membresía, de manera que se representen todos los estados (valores lingüísticos) que la variable de entrada (variable lingüística) pueda tomar.
- Partir con funciones de membresía cuyas formas tengan un ancho mínimo y sean simétricas.
- Distribuir las funciones de membresía a lo largo del universo de discurso de tal forma que éstas se entrecrucen entre sí por la mitad, logrando de esta manera que todos los elementos de dicho universo queden definidos.
- Experimentar con funciones de membresía curvadas y rectas (Triangulares).

Con respecto a las reglas difusas:

- Procurar que las reglas no se contradigan entre sí.
- El número de reglas que se establezcan no deben ser en gran cantidad, esto con el fin de que el controlador no se vuelva complejo e incremente su tiempo de procesamiento.
- Tratar de usar un número reducido o medio de variables y valores lingüísticos, ya que el número de reglas que tendrá el controlador difuso depende de estos dos parámetros.

Tomando en cuenta estas consideraciones se procedió con el diseño de cada uno de los controles difusos.

3.1.4 CONTROL PROPORCIONAL DERIVATIVO DIFUSO (CPDD)

Para este controlador se implementó el lazo de control que se observa en la Figura 78.

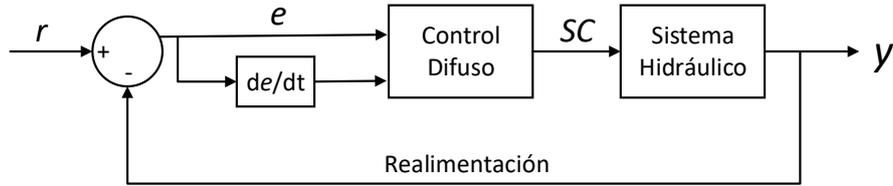


Figura 78. Lazo de control implementado para el CPDD.

En esta figura se aprecia que las variables lingüísticas de entrada para el controlador difuso son el error y su derivada, mientras que como variable lingüística de salida se tiene la Señal de Control (SC) que actuará en el sistema hidráulico.

Los universos de discurso de estas variables son:

- Error [-100 100]
- Error derivado [-40 40]
- Señal de Control [0 2.5]

Para cada una de estas variables se utilizaron un total de cinco valores lingüísticos, éstos son:

- Error y error derivado: Negativo Grande (NG), Negativo Pequeño (NP), Cero (C), Positivo Pequeño (PP) y Positivo Grande (PG).
- Señal de control: Nada, Poco, Medio, Alto y Muy-Alto.

En cuanto a las funciones de membresía, tanto para las variables lingüísticas de entrada como de salida, se hicieron uso de tres funciones triangulares, una en forma de s y una en forma de z , Figura 79.

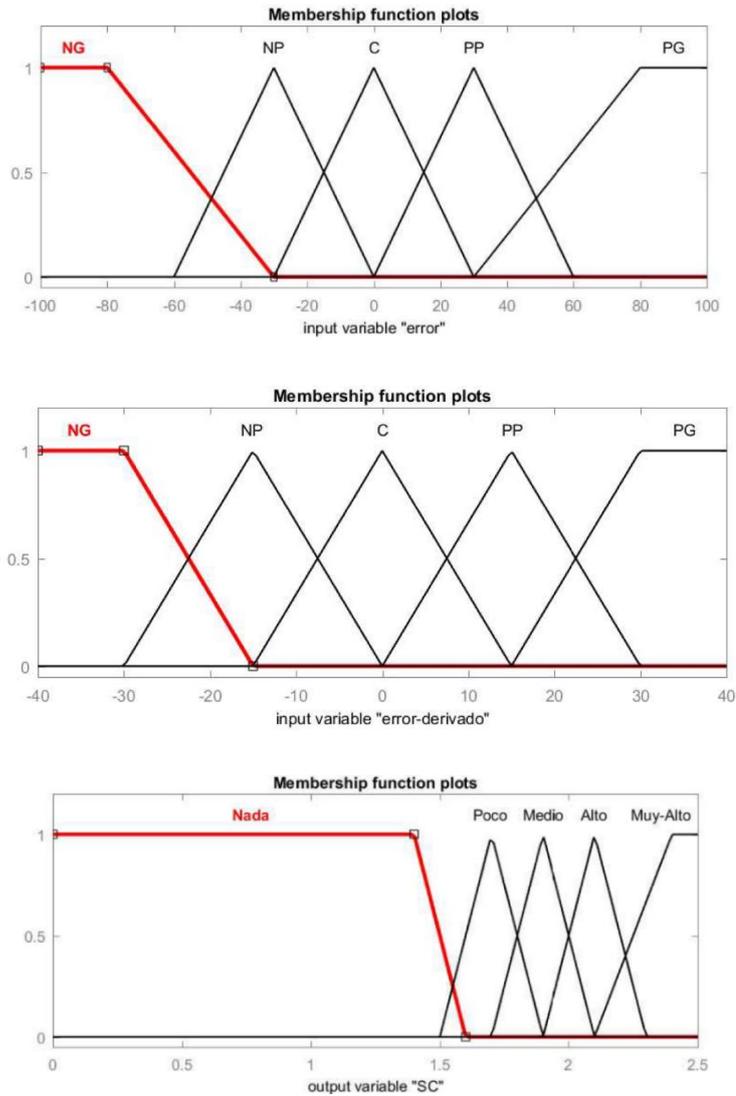


Figura 79. Funciones de membresía implementadas para el CPDD.

Para este controlador se crearon un total de quince reglas, las cuales, de forma resumida, se observan en la Tabla XI.

Tabla XI. Reglas difusas creadas para el CPDD.

Reglas para controlador PD					
$e/SC/\dot{e}$	NG	NP	C	PP	PG
NG	X	X	Poco	X	X
NP	X	Medio	Medio	Medio	X
C	Medio	Medio	Nada	Nada	Medio
PP	X	Medio	Medio	Medio	X
PG	X	Alto	Muy Alto	Medio	X

En el Apéndice G se encuentran adjuntas estas reglas, y las de los otros controladores difusos, implementadas en el TOOLBOX FUZZY de MATLAB®.

Finalmente, en la Figura 80 se aprecia la superficie de control obtenida para el CPDD.

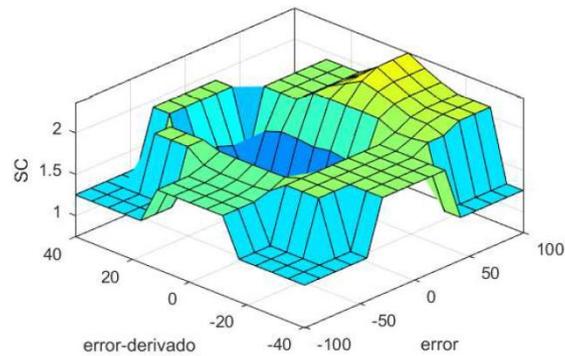


Figura 80. Superficie de control para el CPDD.

3.2.4 CONTROL PROPORCIONAL INTEGRATIVO DIFUSO (CPID)

En este control se utilizaron como variables lingüísticas de entrada al error y su integral, mientras que como variable lingüística de salida se mantuvo la Señal de Control, Figura 81.

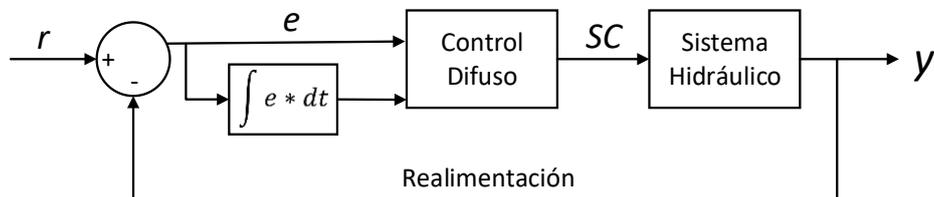


Figura 81. Lazo de control implementado para el CPID.

El rango del universo de discurso para cada una de estas variables es:

- Error: [-100 100]
- Error integrado [-800 800]
- Señal de Control [0 2.5]

En cuanto a sus valores lingüísticos, se utilizaron seis para la variable error, cuatro para el error integrado y nueve para la Señal de control; estos valores son:

- Error: Negativo Grande (NG), Negativo (N), Cero negativo (C-), Cero positivo (C+), Positivo (P) y Positivo Grande (PG).
- Error integrado: Negativo Grande (NG), Negativo (N), Positivo (P) y Positivo Grande (PG).

- Señal de Control: Cero (C), Casi Cero (CC), Bajo (B), Bajo positivo (B+), medio (m), medio positivo (m+), Grande (G), medio Grande (mG), Muy Grande (MG).

Con respecto a las funciones de membresía, se utilizaron las siguientes:

- Error: cuatro triangulares, una *s* y una *z*.
- Error integrado: dos trapezoidales, una *s* y una *z*.
- Señal de Control: una *z*, una *s* y siete gaussianas.

En la Figura 82 se observan estas funciones de membresía y el rango de universo de discurso que ocupan cada una.

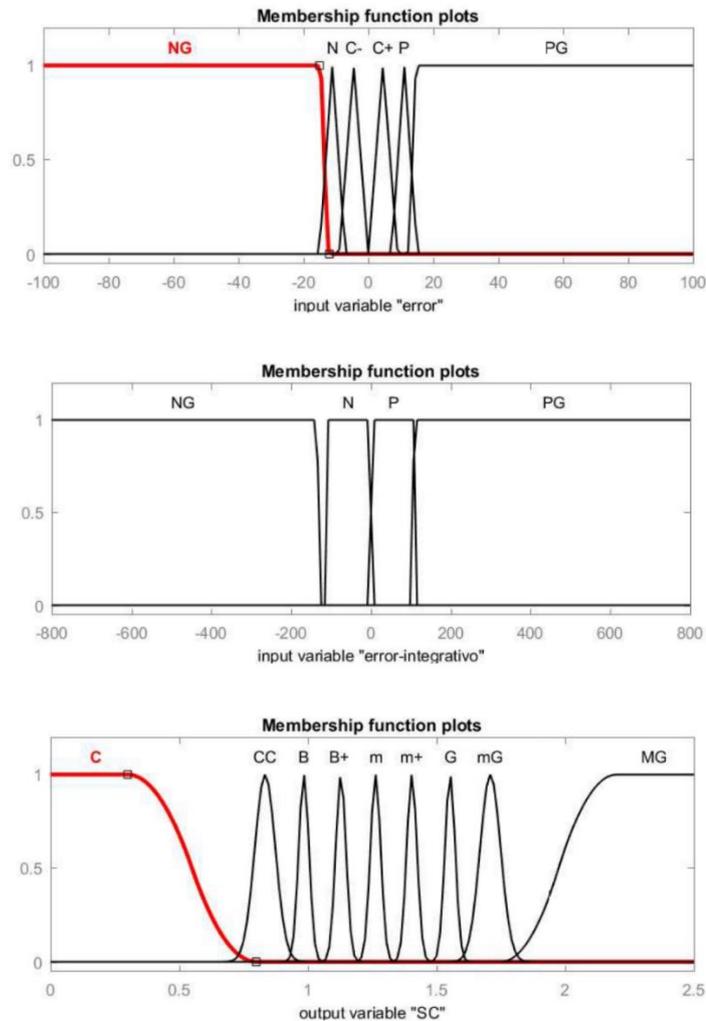


Figura 82. Funciones de membresía implementadas para el CPID.

Una vez definidas las funciones de membresía se procedió con el desarrollo de las reglas difusas; en total se crearon dieciocho, éstas se encuentran resumidas en la Tabla XII.

Tabla XII. Reglas difusas creadas para el CPID.

Reglas para controlador PI				
$e/SC/\int e * dt$	NG	N	P	PG
NG	C	B	MG	MG
N	CC	C	MG	mG
C-	B	B	CC	mG
C+	mG	B+	CC	m
P	mG	G	G	m+
PG	G	B+	m	G

La superficie de control obtenida, como resultado de las reglas difusas creadas, se aprecia en la Figura 83.

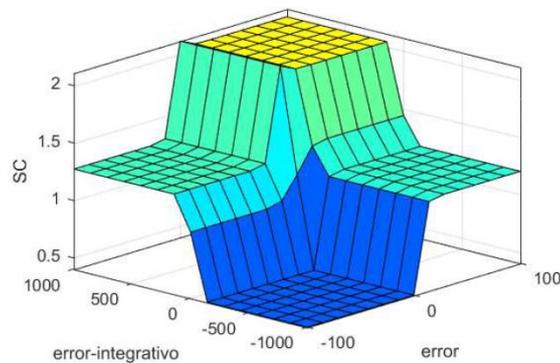


Figura 83. Superficie de control para el CPID.

3.3.4 CONTROL DIFUSO PROPORCIONAL DERIVATIVO + INTEGRATIVO (CDPD+I)

Para este control, a diferencia de los anteriores, se agregó una ganancia integrativa por separado (fuera del control difuso), refiérase a la Figura 84.

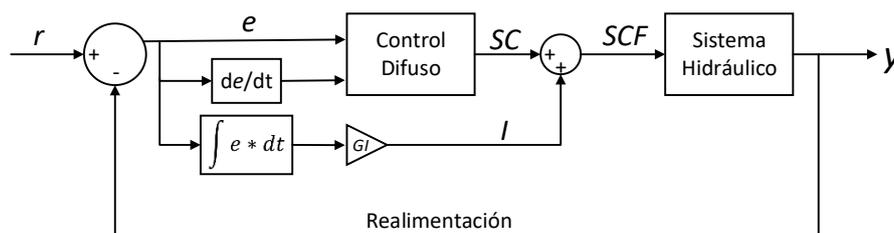


Figura 84. Lazo de control implementado para el CDPD+I.

En la figura anterior se observa que las variables lingüísticas de entrada para el control difuso son el error y su derivada, y como variable lingüística de salida se tiene una señal de control (SC) correspondiente a un control PD; a esta señal se añade una ganancia integrativa (0.0221) para finalmente obtener una Señal de Control Final (SCF). Esta señal será la que actúe sobre el sistema hidráulico. Para la implementación de esta ganancia se usó como referencia la ganancia integrativa de un control clásico diseñado para el sistema hidráulico, para posteriormente ajustarla experimentalmente. El diseño de este control clásico se llevó a cabo con la finalidad de contrastar la respuesta del control difuso con el clásico, refiérase al siguiente capítulo. En cuanto al universo de discurso de cada una de las variables lingüísticas es el siguiente:

- Error: [-100 100]
- Error derivativo: [-70 70]
- Señal de control (SC): [0 2.5]

Sus valores lingüísticos son:

- Error: Negativo Grande (NG), Negativo (N), Cero (C), Positivo (P), Positivo Grande (PG).
- Error derivado: Negativo Grande (NG), Negativo Medio (NM), Negativo (N), Cero (C), Positivo (P), Positivo Medio (PM), Positivo Grande (PG).
- Señal de control: MB (Muy Bajo), mB (medio Bajo), B (Bajo), C (Cero), A (Alto), mA (medio Alto), MA (Muy Alto).

Con respecto a las funciones de membresía, son las siguientes:

- Error: tres triangulares, una z y una s .
- Derivada del error: cinco triangulares, una z y una s .
- SC: cuatro triangulares, una s , una z y una gaussiana.

En la Figura 85 se observan estas funciones.

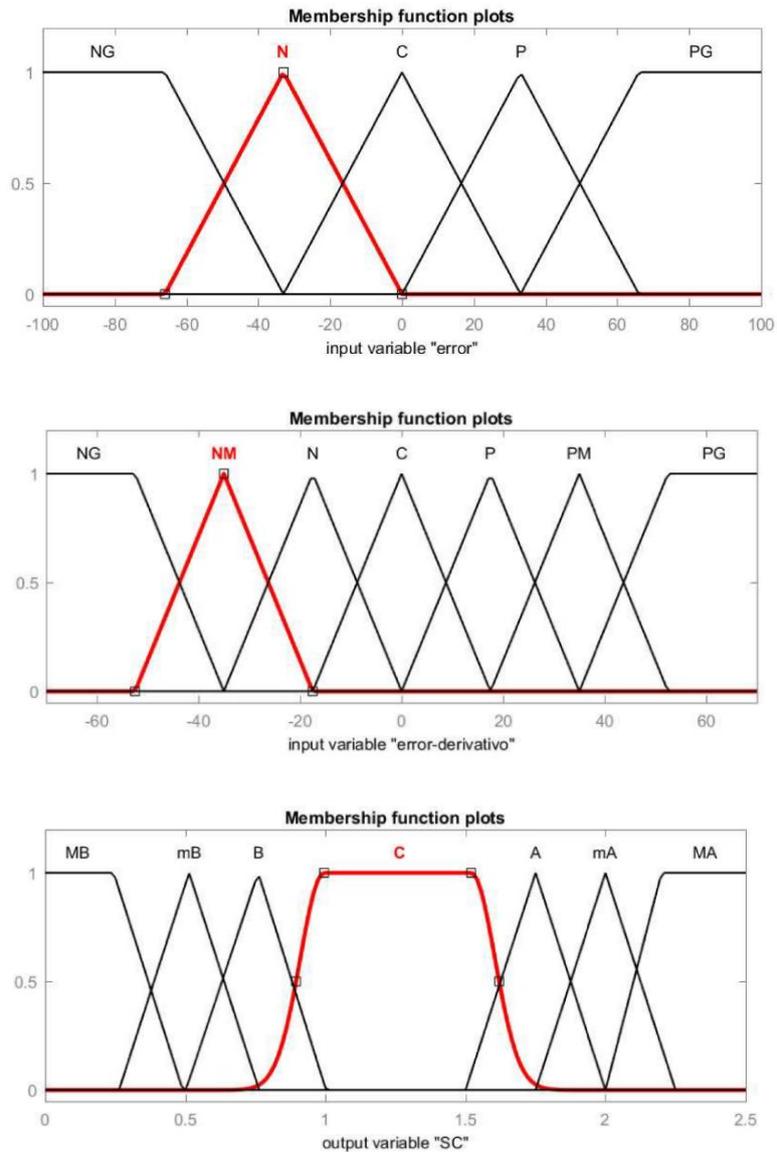


Figura 85. Funciones de membresía implementadas para el CDPD+I.

En cuanto a las reglas difusas, se crearon un total de treinta y cinco reglas, las cuales se indican en la

Tabla XIII.

Tabla XIII. Reglas difusas creadas para el CDPD+I.

Reglas para controlador PD+I							
$e/SC/\dot{e}$	NG	NM	N	C	P	PM	PG
NG	MB	MB	mB	B	B	C	C
N	MB	MB	B	B	C	A	A
C	mB	B	B	C	A	A	mA
P	B	B	C	A	A	MA	MA
PG	C	C	A	A	mA	MA	MA

La superficie de control obtenida debido a dichas reglas se observa en la Figura 86.

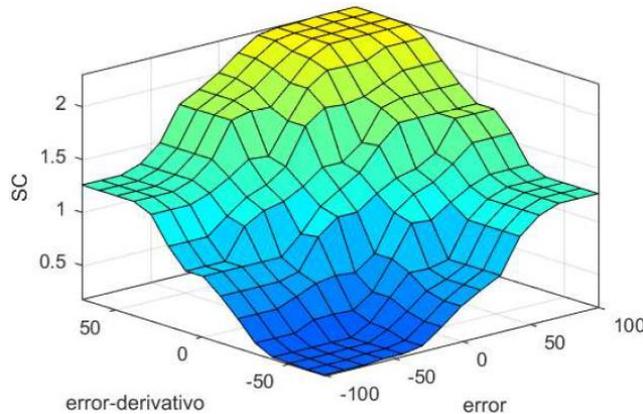


Figura 86. Superficie de control para el CDPD+I.

Los diseños de los controles difusos, expuestos anteriormente, son producto de un largo proceso experimental de: pruebas de funcionamiento, adaptaciones y ajustes de variables lingüísticas, rangos de universos de discurso, etc.

Además, como se mencionó anteriormente, se tomaron como referencia las sugerencias que diferentes autores expusieron en sus literaturas, la cual fue revisada para el desarrollo del estado del arte de este proyecto de tesis.

3.5 DESARROLLO DE VI EN LABVIEW®

Varios de los procedimientos mencionados en secciones anteriores, como: el muestreo, filtrado y promediado de señales, control clásico y difuso del sistema en conjunto, se llevaron a cabo con la ayuda de VI diseñados en el software LabVIEW®. En esta sección se presenta el contenido de estos VI.

Como se mencionó en la sección 2.2.8, el software LabVIEW® trabaja en conjunto con dos ventanas: Panel de Frontal y Diagrama de Bloques; estas dos ventanas componen un VI. El Panel Frontal es la interfaz que se presenta para el usuario, mientras que el Diagrama de bloques es en donde se encuentra implementada la programación gráfica. Los VI desarrollados para los procedimientos anteriormente mencionados están compuestos por bloques que cumplen una funcionalidad en particular y que, conectados con otros bloques proporcionan una funcionalidad más completa. A continuación, se presentan los VI desarrollados para los diferentes procedimientos llevados a cabo con anterioridad.

3.1.5 VI PARA EL MUESTREO Y PROMEDIADO DE SEÑALES

En las Figuras 87 y 88 se observan el Diagrama de bloques y el Panel Frontal, respectivamente, desarrollados para el muestreo y promediado de señales.

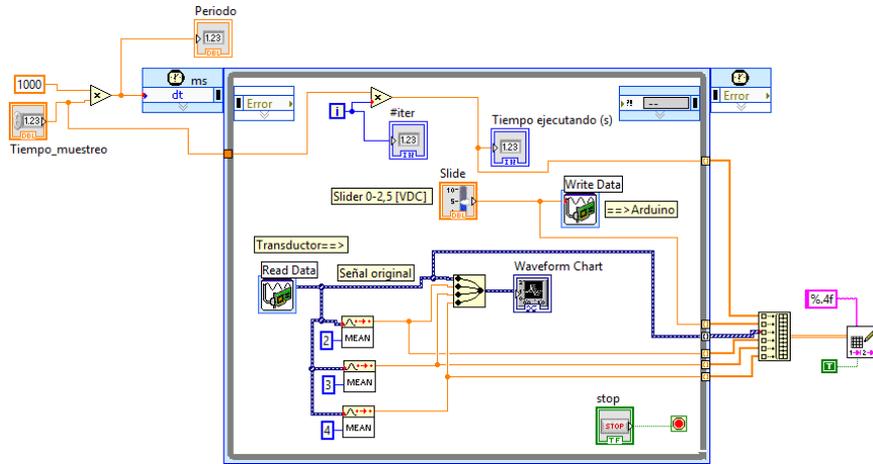


Figura 87. Diagrama de bloques para el muestreo y promediado de señales.

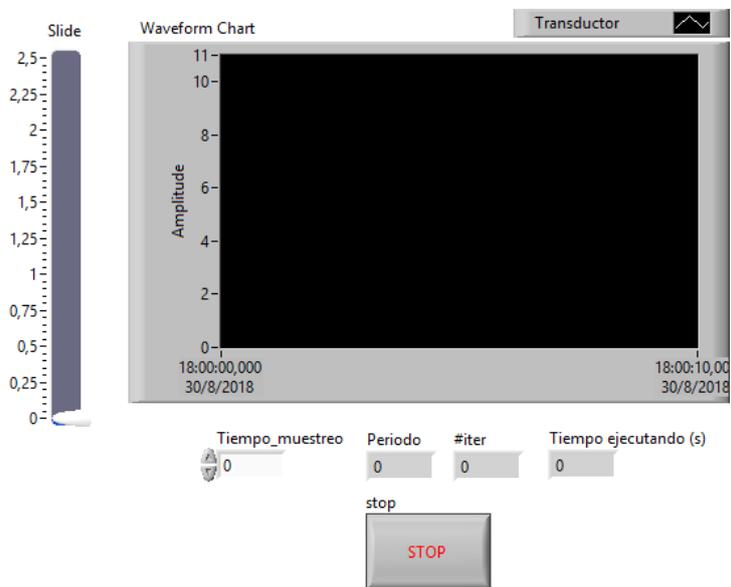


Figura 88. Panel Frontal para el muestreo y promediado de señales.

El funcionamiento de este VI es el siguiente:

Como primer paso, en el Panel Frontal, se ingresa en el control numérico “Tiempo_muestreo” el periodo al cual se va muestrear la señal del Transductor. Este periodo es tomado por el Diagrama de bloques para configurar cada que tiempo se debe ejecutar el lazo temporizado. Después de esto, los indicadores numéricos “Periodo”, “#iter” y “Tiempo ejecutado (s)” muestran el tiempo de muestreo ingresado, el número

de iteraciones que lleva ejecutando el lazo temporizado y el tiempo que se lleva muestreando en segundos, respectivamente.

Para empezar a manipular la señal del Transductor se desliza la barra del “Slide”. Esta barra envía una señal entre 0 y 2.5VDC a la placa Arduino a través del asistente DAQ “Write Data”. Esta señal pasa por todas las etapas especificadas con anterioridad en la sección 3.1.

La señal eléctrica de tensión que produce el Transductor es realimentada a la MyDAQ para que sea interpretada por el software LabVIEW®. Esta señal ingresa al Diagrama de bloques a través del asistente DAQ “Read Data”. Aquí esta señal toma dos caminos diferentes: el primero lleva a la señal ruidosa del Transductor (“señal original”) a imprimirse en el bloque Waveform Chart, mientras que el segundo hace que la señal se promedie cada dos, tres y cuatro muestras, para posteriormente imprimirse en el mismo bloque, Figura 87. Estas señales sin promediar y promediadas, junto con el tiempo de ejecución, son enviadas al bloque que almacena los vectores de datos de las señales (“Write Delimited Spreadsheet”) a través del bloque constructor de vectores. Este bloque que almacena los vectores de datos está configurado para que los valores que acumula tengan una precisión de cuatro decimales y los vectores sean almacenados en columnas.

El proceso de almacenamiento de datos ocurre cuando se presiona el botón “Stop” en el Panel Frontal. Los vectores de datos se guardan en un archivo de texto (.txt) para posteriormente cargar estos vectores al software MATLAB® a través de programación.

3.2.5 VI PARA EL FILTRADO DE SEÑALES

El Panel frontal y el Diagrama de bloques desarrollados para este proceso se observan en las Figuras 89 y 90.

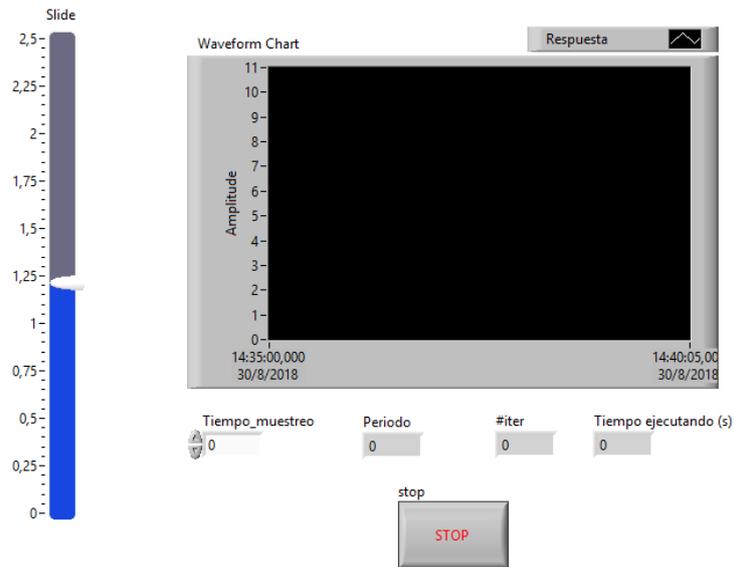


Figura 89. Panel Frontal para el filtrado de señales.

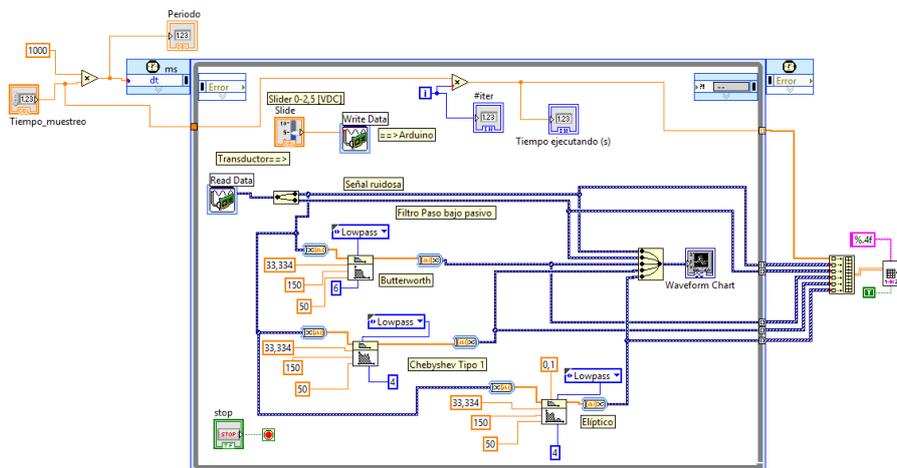


Figura 90. Diagrama de bloques para el filtrado de señales.

El funcionamiento de este VI es similar al expuesto para el muestreo y promediado de señales; sin embargo, se diferencia en que, además de ingresar la señal del Transductor a LabVIEW®, también se ingresa la señal proveniente del filtro pasivo paso bajo configurado a una F_c determinada.

Además, la señal del Transductor ya no es promediada sino más bien es filtrada mediante los filtros digitales Butterworth, Chebyshev Tipo 1 y Elíptico, los cuales están configurados con los datos obtenidos en el proceso de diseño.

3.3.5 VI PARA EL CONTROL CLÁSICO

Las pruebas de funcionamiento del Control Clásico se llevaron a cabo con el Panel Frontal y el Diagrama de bloques expuestos en las Figuras 91 y 92 respectivamente.

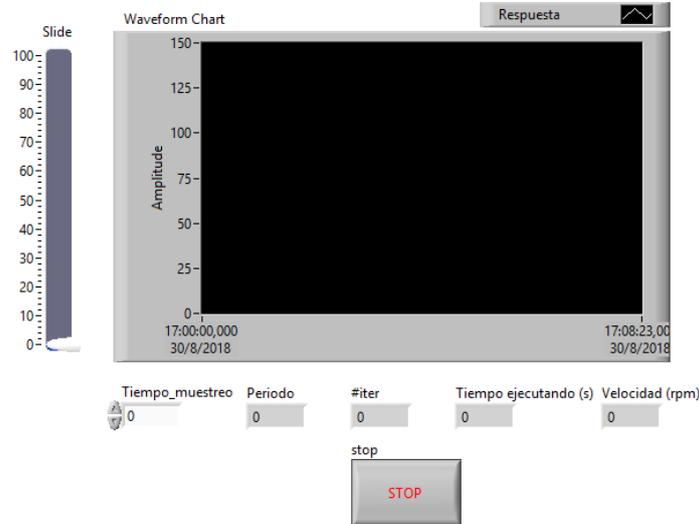


Figura 91. Panel Frontal utilizado para las pruebas de funcionamiento del Control Clásico.

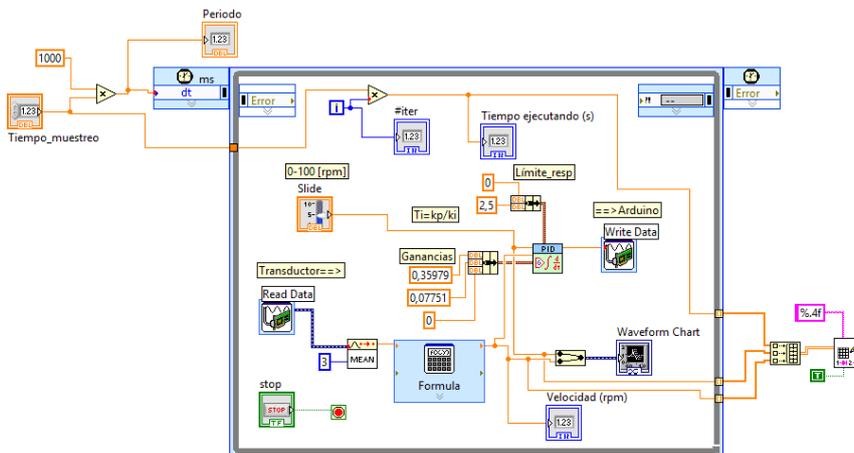


Figura 92. Diagrama de bloques utilizado para las pruebas de funcionamiento del Control Clásico.

El modo de operación de este VI es parecido a los expuestos anteriormente, esto en cuanto a la lectura, escritura y almacenamiento de datos. En este VI la señal del “Slide” no se dirige directamente al asistente DAQ “Write Data” sino que es enviada al bloque PID. Así también, la señal del Transductor es promediada cada tres muestras y enviada al bloque fórmula, que es en donde se encuentra implementada la constante K , la cual indica la relación tensión-velocidad del Transductor (sección 3.3), para posteriormente ser enviada al bloque PID.

En cuanto al bloque PID, a éste se conecta:

- Señal de referencia (señal del “Slide”).
- La variable de proceso (señal del Transductor en unidades rpm)
- Ganancias K_p , K_i y K_d que se obtengan en el proceso de diseño del controlador.
- Asistente DAQ “Write Data”
- Los límites de la señal de control que emitirá este bloque.

Este bloque toma los tres primeros parámetros mencionados y los evalúa, para a continuación producir y efectuar una señal de control para el sistema hidráulico.

3.4.5 VI PARA EL CONTROL DIFUSO PD, PI Y PD+I

El Panel de control que se utilizó para llevar a cabo las pruebas de funcionamiento de los controladores difuso PD, PI y PD+I se aprecia en la Figura 93.

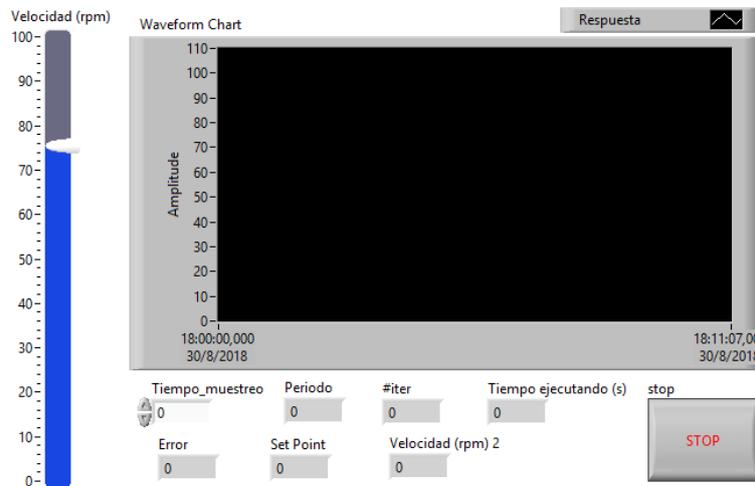


Figura 93. Panel Frontal utilizado para las pruebas de funcionamiento de los Controles Difusos.

En cuanto a los Diagramas de bloques, expuestos en las Figura 94 y 95, son los que se utilizaron para las pruebas de funcionamiento de los controladores PD y PI, y PD+I respectivamente.

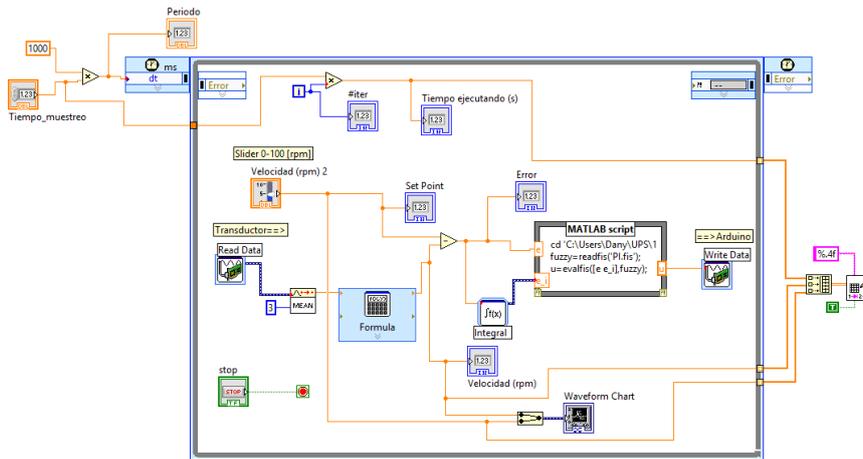


Figura 94. Diagrama de bloques utilizado para las pruebas de funcionamiento de los Controles Difusos PD y PI.

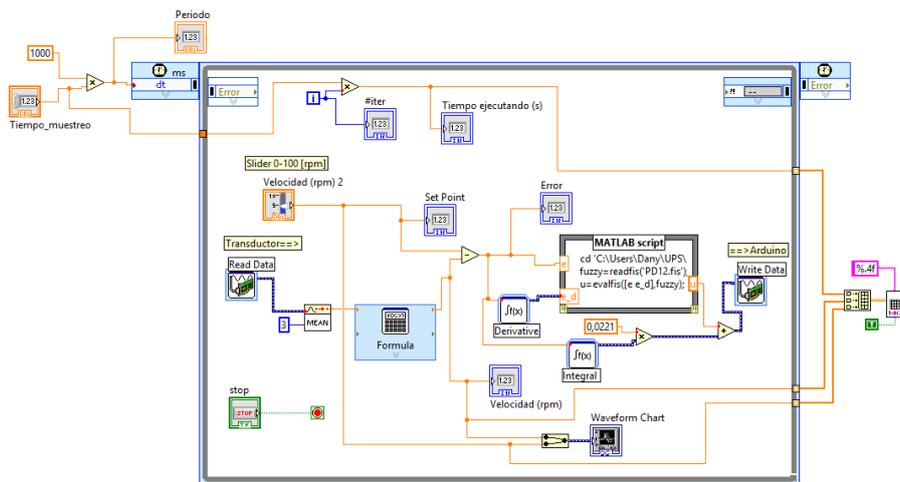


Figura 95. Diagrama de bloques utilizado para las pruebas de funcionamiento del Control Difuso PD+I.

El funcionamiento de estos VI es parecido al del Control Clásico, pero con la diferencia que no se utiliza un bloque PID. Además, estos VI cuentan con otros bloques como Integrativo y Derivativo. Estos bloques, como su nombre lo indica, son los que Integran y Derivan la señal de error. Esta señal es la diferencia entre el Setpoint (referencia regida por el “Slide”) y la señal de retroalimentación (proporcionada por el Transductor y acondicionada en rpm). Para el caso de los controladores difusos PD y PI, Figura 94, el bloque Integral se utiliza para el control PI, mientras que para el control PD este bloque se cambia por el Derivativo. Además, también es necesario cambiar el nombre del archivo con extensión *fis* (el cual se encuentra especificado dentro del Script de MATLAB®). El archivo con esta extensión es en donde se encuentra implementada toda la lógica difusa. Las señales integradas y derivadas respectivamente por estos bloques son enviadas al Script de MATLAB® para que sean evaluadas por la lógica difusa

diseñada y desarrollada con anterioridad (sección 3.4). Luego de esta evaluación este Script produce y efectúa una señal de control para el sistema hidráulico.

En cuanto al control PD+I, se utiliza tanto el bloque Derivativo como Integrativo. Como se explicó con anterioridad en la sección 3.4, este control PD+I utiliza una ganancia integrativa externa a la lógica difusa, por lo que la señal de control final que actuará sobre el sistema hidráulico será la suma de esta ganancia integrativa con la señal obtenida luego de la evaluación difusa en el Script de MATLAB®, Figura 95.

3.5.5 FRECUENCIA DE MUESTREO

Para hallar el valor de la frecuencia de muestreo se consideró aplicar el criterio de Nyquist, el cual enuncia que la frecuencia de muestreo (F_s) de una señal analógica debe ser mínimo dos veces su frecuencia máxima ($F_{m\acute{a}x}$), de esta manera se evita el Alising (solapamiento) de la señal cuando sea digitalizada [48].

Tomando esto en cuenta se procedió a experimentar con la señal del Transductor, ya que esta señal es la que debe ingresarse al ordenador a través de la tarjeta de adquisición de datos MyDAQ. Durante la experimentación se observó que la frecuencia de esta señal es en exceso cambiante, esto debido a la presencia de ruido (sección 3.2); es por esta razón que se optó por llevar a cabo otro método para encontrar una frecuencia de muestreo.

El otro método (experimental, sugerencia docente) que se decidió probar consiste en aplicar un escalón al sistema en conjunto y capturar su respuesta considerándola como de primer orden. Con esta respuesta se mide el valor de τ para posteriormente dividirlo en un número N de partes; de esta manera se establece un periodo de muestreo (T_s).

Se mide la constante de tiempo τ debido a que corresponde al tiempo mínimo que le toma a un sistema cambiar su dinámica. En la Figura 96 se observa el impulso aplicado y la respuesta del sistema en conjunto.

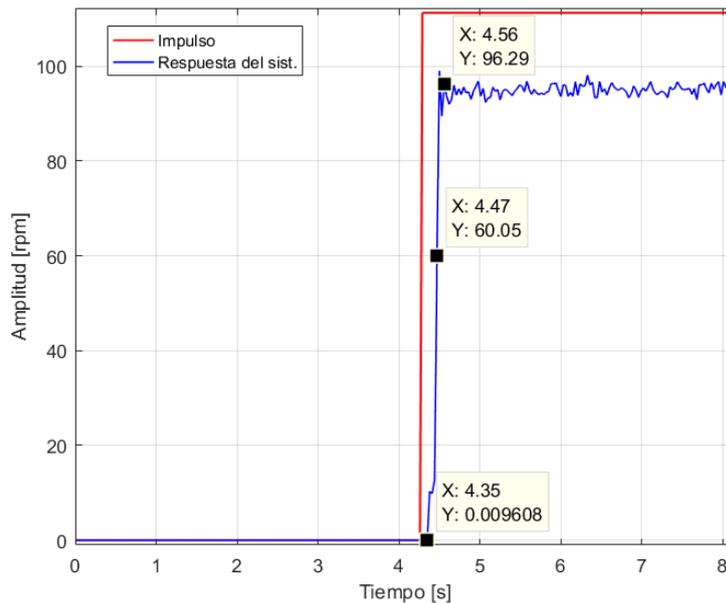


Figura 96. Respuesta del sistema en conjunto al aplicarse una señal escalón.

Con los datos de la Figura 96 se procedió a calcular el valor de τ :

Como se sabe, el valor de τ se mide cuando la amplitud de la señal de respuesta alcanza el 63.2% del valor máximo de su amplitud. Con base a esto, de la Figura 96 se tiene que:

$$\text{Amplitud máxima} = 96.29 \text{ [rpm]}$$

$$\text{Amplitud para medir } \tau = \frac{96.29 * 63.2}{100} = 60.85 \text{ [rpm]}$$

Con este dato se coloca el cursor en aproximadamente esta amplitud y se mide el tiempo τ .

Observando la Figura 96 y con referencia a la señal de respuesta, se aprecia que el tiempo cuando la amplitud alcanza aproximadamente el 63.2% es de 4.47 segundos, mientras que el tiempo en la base es de 4.35 segundos; obteniendo la diferencia de estos dos tiempos se tiene finalmente que el valor de τ es de 0.12 segundos. Una vez calculada esta constante de tiempo se procedió a dividirla en un número $N = 20$ (valor de prueba), obteniendo un periodo de muestreo de:

$$T_s = \frac{0.12}{20} = 6ms$$

Y una frecuencia de muestreo de:

$$F_s = \frac{1}{T_s} = \frac{1}{0.006} = 166.67Hz$$

Esta constante de tiempo τ fue dividida para $N = 20$ debido a que se consideró que era el número mínimo de muestras necesarias para reconstruir la respuesta del sistema hasta el valor de τ . Sin embargo, este número puede ser mayor a veinte, esto produciría que el T_s se reduzca mientras se incrementa F_s ; por el contrario, si N es menor a veinte se corre el riesgo de que el número de muestras no sea suficiente para reconstruir la respuesta del sistema hasta el valor de τ , obteniendo de esta manera una señal diferente a la muestreada.

Luego de esto, se procedió a realizar pruebas de funcionamiento muestreando la señal del Transductor haciendo uso de esta constante de tiempo τ . Durante las pruebas de funcionamiento se observó que el periodo de muestreo obtenido era demasiado bajo para el sistema en conjunto, esto debido a que el número de muestras que se esperaba obtener en un lapso de tiempo determinado no era el correcto.

Ampliando esto. Como se mencionó anteriormente, para utilizar los VI (expuestos anteriormente) es necesario ingresar un periodo de muestreo. Este periodo sirve para determinar cada que tiempo se ejecuta el lazo temporizado, el cual contiene toda la lógica para el muestreo de señales, refiérase a la sección anterior. Con esta consideración se realizaron nuevas pruebas de funcionamiento para determinar el periodo de muestreo mínimo, en el cuál, el número de muestras que se supone obtener en un lapso de tiempo determinado se cumpla. Por ejemplo, si el periodo de muestreo es de 1 segundo, el lazo temporizado se ejecuta cada 1 segundo, obteniendo de esta manera una muestra cada segundo. Ahora, si el periodo de muestreo sigue siendo 1 segundo y el proceso de muestreo dura 10 segundos entonces el número de muestras a obtener es de 10. Esta operación sigue la siguiente relación:

$$Nm = \frac{t_m}{T_s} \quad (31)$$

Donde:

Nm número de muestras a obtener.

t_m duración (en segundos) del proceso de muestreo.

T_s periodo de muestreo (en segundos).

Aplicando la ecuación (31) para un periodo de muestreo de 0.5 segundos y para una duración del proceso de muestreo de 10 segundos se tiene que:

$$Nm = \frac{10}{0.5} = 20 \text{ muestras}$$

Entonces, si se muestrea la señal del Transductor cada 0.5 segundos durante 10 segundos el número de muestras que se espera obtener es de 20. Con esta nueva consideración se desarrollaron nuevas pruebas de funcionamiento.

Durante estas nuevas pruebas de funcionamiento se observó que la ecuación (31) se cumple hasta cuando el periodo de muestreo es de 30ms (0.03 segundos). Cuando el periodo es menor a este valor, el número de muestras que se obtienen no coincide con el número de muestras que se supone se debía obtener. Esto es debido a los bloques que contiene el lazo temporizado; el ordenador no alcanza a realizar todo el procesamiento computacional que debe llevar a cabo para obtener el número de muestras correcto. Es por esta razón que finalmente se estableció como periodo de muestreo $T_s = 30ms$ ($F_s = 33.334Hz$).

CAPÍTULO 4: PRUEBAS DE FUNCIONAMIENTO, ANÁLISIS DE RESULTADOS, CONCLUSIONES Y RECOMENDACIONES

4.1 PRUEBAS DE FUNCIONAMIENTO

4.1.1 DISEÑO DEL PLAN PILOTO

Para poder medir y observar los resultados de los controladores con los que se ha trabajado, se planteó realizar lo que se ha denominado un plan piloto. Este plan se realiza con la mira de establecer y esclarecer los parámetros que se consideraron para obtener los resultados de los controladores. El plan piloto de las pruebas de funcionamiento que se diseñó contiene los ítems expuestos en la Tabla XIV.

Tabla XIV. Presenta el plan piloto que se diseñó para las pruebas de los controladores.

Consideraciones del plan piloto.		
Ítem	Representación	Propósito
Velocidad de Entrada	Setpoint o Referencia (0-100rpm).	Conocer el valor de referencia actual.
Velocidad de Salida	Dato de salida de la planta (0-100rpm).	Conocer el valor de salida para el valor de referencia.
Error en Estado Estacionario (E_{ee})	$E_{ee} = \frac{R}{1 + \lim_{s \rightarrow 0} G(s)}$	Observar el E_{ee} cuando la salida de la planta se estabilice.
Sobre-elongación Máxima (Mp)	$Mp(\%) = \frac{y(tp) - y(\infty)}{y(\infty)} * 100$ Dónde: tp , es el tiempo requerido para que la respuesta alcance el primer pico de sobre-elongación.	Obtener el valor de Mp con cada controlador.
Tiempo de Subida (ts) [s]	Es el tiempo requerido para que la respuesta pase al 95% de su total .	Observar el ts que se presente con cada controlador.
Tiempo de Establecimiento (te) [s]	Es el tiempo que se requiere para que la curva de respuesta alcance un rango alrededor del valor final. Especificado por lo general de 2 o 5% del valor final.	Observar el te que se presente en cada controlador.

Respuesta ante Perturbaciones	Introducir una perturbación al sistema al cerrar parcialmente su válvula principal.	Observar el comportamiento de los controladores ante perturbaciones.
Respuesta ante Cambios de Referencia	Realizar cambios de referencia tanto ascendentes como descendentes.	Observar cómo responde el controlador ante los cambios en niveles de referencia.

4.2.1 PRUEBA DE FUNCIONAMIENTO DEL CONTROLADOR DIFUSO PD: CAMBIOS DE REFERENCIA

El primer ensayo que se realizó sobre el controlador difuso tipo PD, fue el referido a cambios de referencia. Esta prueba se efectuó con la intención de observar su comportamiento y si cumple con las expectativas y necesidades esperadas. En la Figura 97 se muestran los resultados conseguidos.

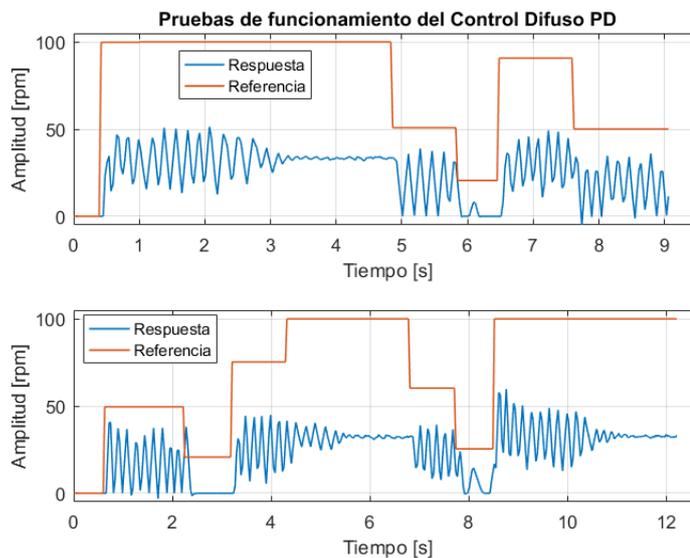


Figura 97. Prueba de funcionamiento del controlador PD con cambios de referencia.

Como se puede observar en la Figura 97, el desempeño del controlador difuso PD es ineficiente, mostrando que en ninguno de los casos llega hasta la referencia. Además, se muestra la presencia de picos antes de llegar a una leve estabilización en el intervalo aproximado entre los 3 y 5 segundos. Sin embargo, esta estabilización no está sobre el nivel de referencia deseado (100rpm).

4.3.1 PRUEBA DE FUNCIONAMIENTO DEL CONTROLADOR DIFUSO PI: CAMBIOS DE REFERENCIA

El segundo controlador analizado es el difuso tipo PI. Al igual que el caso anterior se sometió al ensayo con cambios de referencia. De la misma manera, la intención de la prueba era observar su desenvolvimiento ante estas variantes y extraer las primeras deducciones tanto de la prueba como del controlador. En la Figura 98 se muestra los resultados obtenidos.

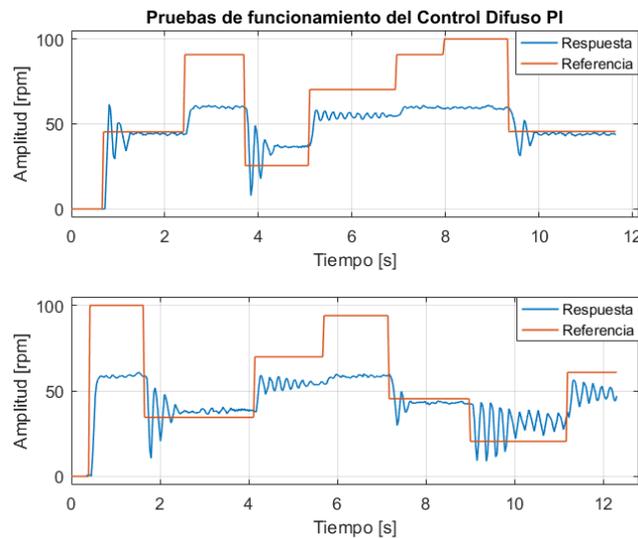


Figura 98. Prueba de funcionamiento del controlador PI ante cambios de referencia.

En la Figura 98, se puede notar que la respuesta del controlador PI no cumple con los cambios de referencia como se esperaba, sin embargo, su desempeño es mejor con respecto al controlador PD. Brevemente, se observa que los picos presentados en la respuesta de controlador son más reducidos y existe un intervalo entre 45 y 55rpm, aproximadamente, en donde el controlador responde correctamente y se estabiliza.

4.4.1 PRUEBA DE FUNCIONAMIENTO DEL CONTROLADOR DIFUSO PI: INGRESO DE PERTURBACIONES

Debido a que el funcionamiento del controlador PI mostró una mejoría con respecto al PD, al menos en un rango, se consideró conveniente realizar pruebas ingresando perturbaciones, dentro del rango funcional del PI, con la mira solamente de observar las respuestas y obtener las primeras retroalimentaciones del camino que estaban tomando el diseño de los controladores. Para realizar las perturbaciones al sistema se

cerró reducidamente la válvula principal del banco hidráulico. En la Figura 99 se muestra la respuesta lograda.

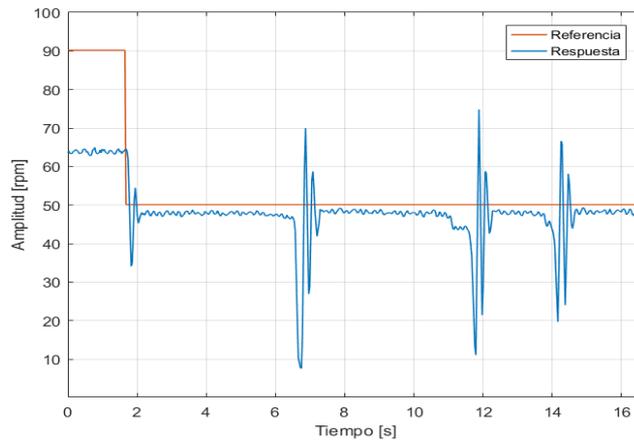


Figura 99. Sondeo de funcionamiento del controlador PI ante ingreso de perturbaciones.

Se puede verificar que el controlador se repone convenientemente ante la presencia de la perturbación, retomando el valor de referencia con el que está trabajando. Esto indicó que el diseño del controlador PI era un avance positivo hacia el cumplimiento del objetivo principal.

4.5.1 PRUEBA DE FUNCIONAMIENTO DEL CONTROLADOR DIFUSO PD+I: CAMBIOS DE REFERENCIA

El tercer controlador ensayado es el difuso tipo PD+I. De igual manera que los controladores PD y PI, se realizó el ensayo en presencia de cambios de referencia. En la Figura 100 se muestra los efectos obtenidos.

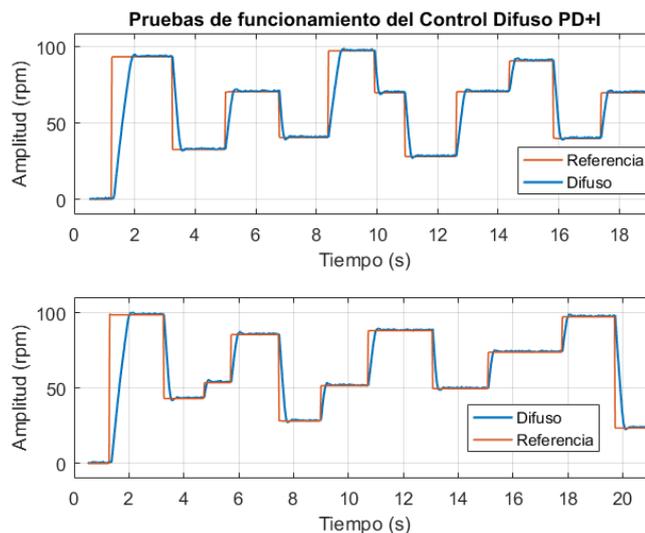


Figura 100. Prueba de funcionamiento del controlador PD+I con cambios de referencia.

Como se expone en la Figura 100, la respuesta del controlador PD+I cumple con los cambios de referencia de la prueba. Este controlador ya no muestra los picos presentados por los controladores anteriores, además, llega y se estabiliza en los niveles de referencia ingresados.

4.6.1 DISEÑO DE UN CONTROLADOR CLÁSICO

Luego del diseño e implementación de los controladores difusos y de obtener uno funcional en la prueba ante cambios de referencia, se procedió con el diseño de un controlador clásico con el fin de comparar la respuesta de éste con el difuso PD+I. La idea principal de la comparación es mostrar la diferencia entre el comportamiento de un controlador tradicional y uno de naturaleza difusa, tratando de evaluar todos los ítems posibles que se mencionaron en la Tabla XIV. A continuación, se presenta el desarrollo del controlador clásico.

En primera instancia, antes de diseñar el controlador clásico se procedió a obtener la Función de Transferencia (FT) del sistema en conjunto en un punto de operación (esto debido a que la respuesta del sistema en conjunto no es lineal). Para obtener la FT se introdujo un escalón con amplitud 110 rpm (provocando que el motor opere en su máxima velocidad) y se muestreó la respuesta del sistema, refiérase a la Figura 101.

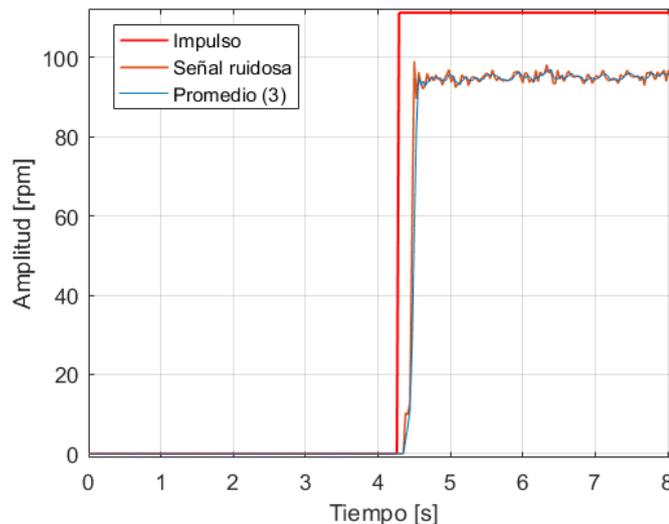


Figura 101. Respuesta al escalón del sistema.

Como se observa en la Figura 101, la señal de respuesta del sistema contiene ruido, por lo tanto, con el fin de reducir este ruido se procedió a utilizar el mismo método de reducción de ruido expuesto en la Sección 3.2; promedio móvil de la señal cada tres

muestras. Luego de obtener la respuesta de este sistema se procedió a ingresarla en MATLAB® para estimar la FT mediante el comando tfest (Estimate transfer function using time or frequency domain data). Este comando permite obtener la FT de un sistema ingresando los siguientes parámetros:

- Señal de entrada (escalón).
- Señal de respuesta (señal promediada cada tres muestras).
- Tiempo de muestreo en segundos (0.03).
- Número de polos que se desea en la FT.
- Número de ceros que se desea en la FT.
- Tiempo muerto (NaN).

De los parámetros anteriores, para el tiempo muerto, se ingresa su valor únicamente si se conoce, caso contrario se deja con el valor NaN para que, junto al numerador y denominador de la FT, éste parámetro sea estimado por MATLAB®. Posteriormente se hicieron pruebas para determinar la FT más próxima a la de este sistema. Para esto se fueron cambiando el número de polos y ceros hasta obtener un alto porcentaje de estimación. Los resultados obtenidos son los siguientes:

Tabla XV. Resultados de las pruebas de estimación de la FT.

Porcentajes obtenidos para la estimación de la FT.			
Número de Polos	Número de Ceros	Porcentaje de estimación [%]	MSE
1	0	98,36	0.006932
1	1	98,66	0.004594
2	1	98,81	0.00364
2	2	98,43	0.006353
3	2	99,05	0.00232
3	3	99,16	0.001776
3	1	99,04	0.002353

En la Tabla XV, se observa que el porcentaje de estimación es más alto cuando se utilizan tres polos y tres ceros, además, en este mismo caso el Error Cuadrático Medio (MSE, por sus siglas en inglés) es el más bajo obtenido en todas las pruebas. La respuesta real y estimada del sistema en conjunto se observan en la Figura 102.

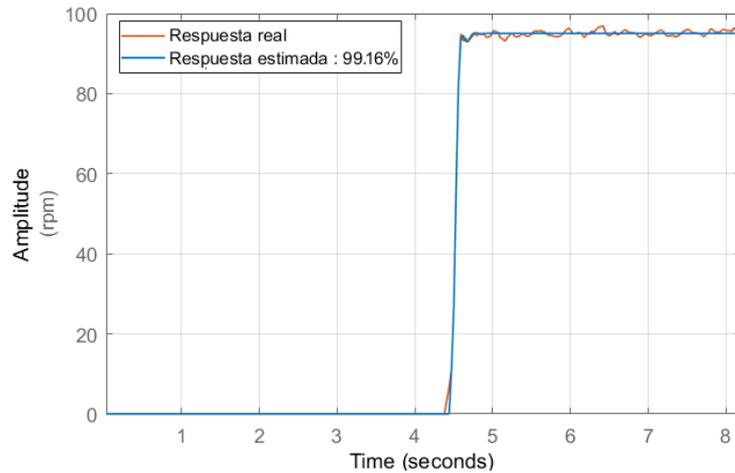


Figura 102. Comparación entre la respuesta real y estimada del sistema en conjunto.

La FT estimada es la siguiente:

$$FT = e^{-0.15*s} * \frac{0.098505 (s+81.78) (s^2+2.44s+3830)}{(s+22.16) (s^2+33.72s+1631)} \quad (32)$$

El código desarrollado en MATLAB® para este proceso de estimación se adjunta en el Apéndice H.

Luego de obtener la FT se utilizó el TOOLBOX PID TUNER de MATLAB® para diseñar el controlador clásico. Durante el proceso de diseño se observó que:

- Si se reduce el sobressalto (Mp) se incrementa el tiempo de subida (ts).
- Si se reduce el tiempo de subida se incrementa el tiempo de establecimiento (te) y el error en estado estacionario (Eee).

Considerando este comportamiento, se procedió a equilibrar estos parámetros para obtener un controlador con una respuesta aceptable, es decir, que no presente: grandes sobre-saltos, tiempos de subida altos, error en estado estacionario elevado, etc. En la Figura 103 se observa el resultado obtenido.

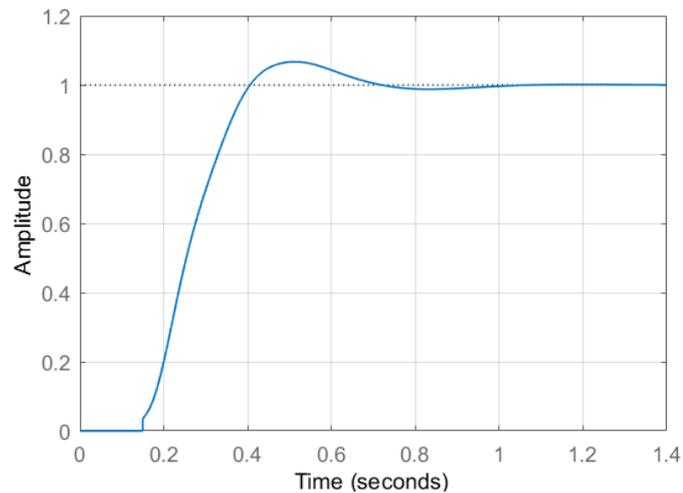


Figura 103. Respuesta en lazo cerrado del control clásico diseñado.

Las ganancias de este controlador son las siguientes:

- $K_p = 0.35979$
- $K_i = 4.6418$
- $K_d = 0$

Estas ganancias se ingresaron al VI de LabVIEW® para llevar a cabo las pruebas de funcionamiento con el control clásico. En cuanto a la respuesta del controlador, se tienen los siguientes resultados:

- Tiempo de subida: 0.184 [s]
- Tiempo de establecimiento: 0.654 [s]
- Sobre-salto: 6.67%
- Valor pico: 1.07
- Margen de ganancia: 7.48dB
- Margen de fase: 60°
- Estabilidad en lazo cerrado: estable

4.7.1 COMPARACIÓN ENTRE EL CONTROLADOR CLÁSICO Y EL CONTROLADOR DIFUSO PD+I: CAMBIOS DE REFERENCIA

Con la intención de contrastar los comportamientos de los controladores, se procedió a realizar pruebas de cambios de referencia en similares condiciones. En la Figura 104 se muestra las secuelas obtenidas.

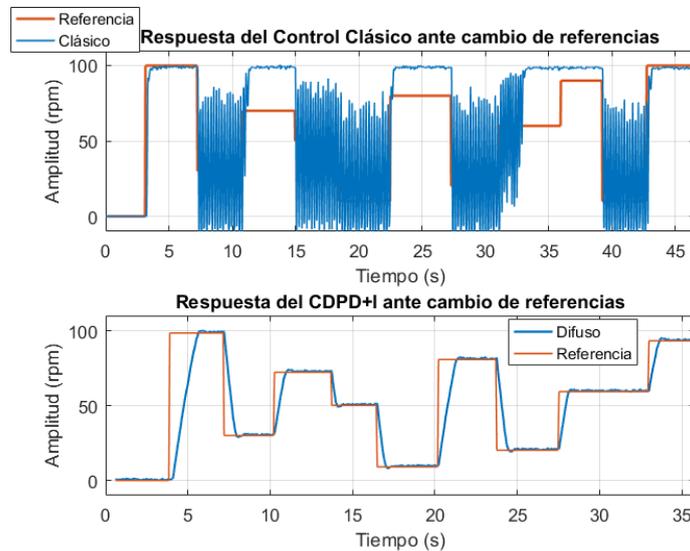


Figura 104. Comparación entre el controlador clásico vs el difuso PD+I ante cambios de referencia.

Brevemente, en Figura 104, se puede verificar que el desempeño del controlador difuso supera por mucho al desenvolvimiento del clásico. De manera similar al caso del difuso PI, el clásico funciona solo en el punto de operación para el que fue diseñado, 100rpm. Mientras que el PD+I cubre de mejor forma todo el rango de operación.

4.8.1 COMPARACIÓN ENTRE EL CONTROLADOR CLÁSICO Y EL CONTROLADOR DIFUSO PD+I: INGRESO DE PERTURBACIONES

Para realizar las pruebas de perturbaciones en el sistema HRE-HIDRAULIC, lo que se efectuó fue interrumpir el flujo normal de aceite del mismo. Esto se logró al cerrar parcialmente su válvula principal. En la Figura 105 se presenta la respuesta de los controladores de esta prueba.

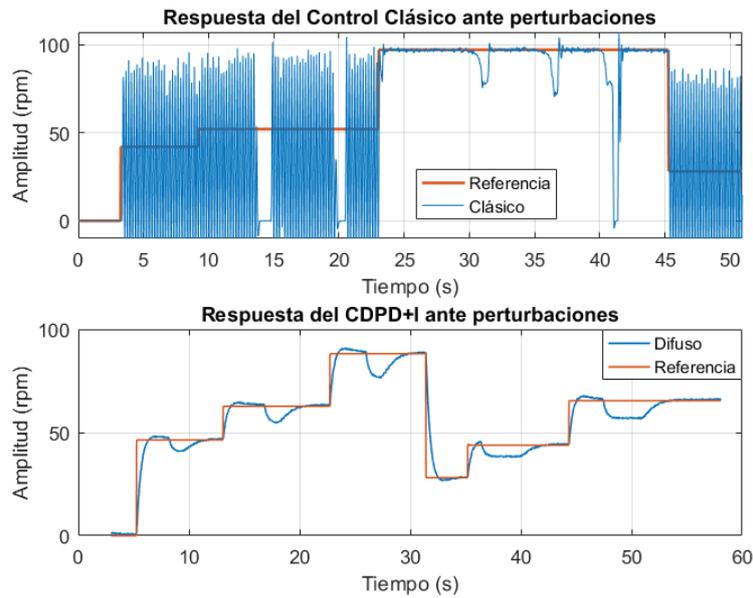


Figura 105. Comparación entre el controlador clásico y el controlador difuso PD+I ante perturbaciones.

En la Figura anterior se puede observar que, para el control clásico, el ensayo de perturbaciones solo es realizable para el nivel de 100rpm. En dicho nivel, el controlador se restaura convenientemente, poco tiempo después de que se anula la alteración. Para el caso del controlador PD+I, se mira que la perturbación es practicable en un rango más amplio de operación, además de que, el controlador retoma correctamente el nivel actual de acción. La contraparte de este controlador es que la reposición para el nivel respectivo es más lenta que la reposición del controlador clásico. Cabe recalcar que en todas las pruebas de perturbaciones la corriente máxima suministrada a la electroválvula fue de 0.65A.

Después de realizar todas las pruebas mostradas en las secciones anteriores, se procedió a medir y tabular los ítems citados en el plan piloto. En la Tabla XVI, se publican los resultados obtenidos de todos los controladores.

Tabla XVI. Publica los resultados obtenidos de las pruebas de todos los controladores.

Resultados medidos y tabulados de los controladores desarrollados según el plan piloto								
Controlador	Velocidad de Referencia (0-100rpm)	Velocidad de salida (0-100rpm)	Error en Estado Estacionario	Sobre-elongación Máxima (Mp%)	Tiempo de Subida (s)	Tiempo de Establecimiento(s)	Respuesta ante Perturbaciones	Respuesta ante Cambios de Referencia
PD Difuso	100	34.02	65.98	No medible	No medible	3 (en 34.02)	No medible	No Medible
PI Difuso	45.37	45.05	0.32	36.58%	0.78	1.8	Si	45-55rpm
PD + I Difuso	93.5	93.27	0.23	2.5%	0.62	1.22	Si	Si
Clásico	100	99.44	0.56	0	0.29	1.08	Si (100rpm)	No

4.9.1 COMPARACIÓN ENTRE EL CONTROLADOR CLÁSICO Y EL CONTROLADOR DIFUSO PD+I: ENTRADA ESCALÓN EN 100RPM

En la Figura 106, se muestra la comparación entre el desempeño de los controladores clásico y difuso. En este caso se consideró el escalón de entrada de 100 rpm dado en el ensayo de cambios de referencia, en donde se han señalado los tiempos y parámetros más relevantes.

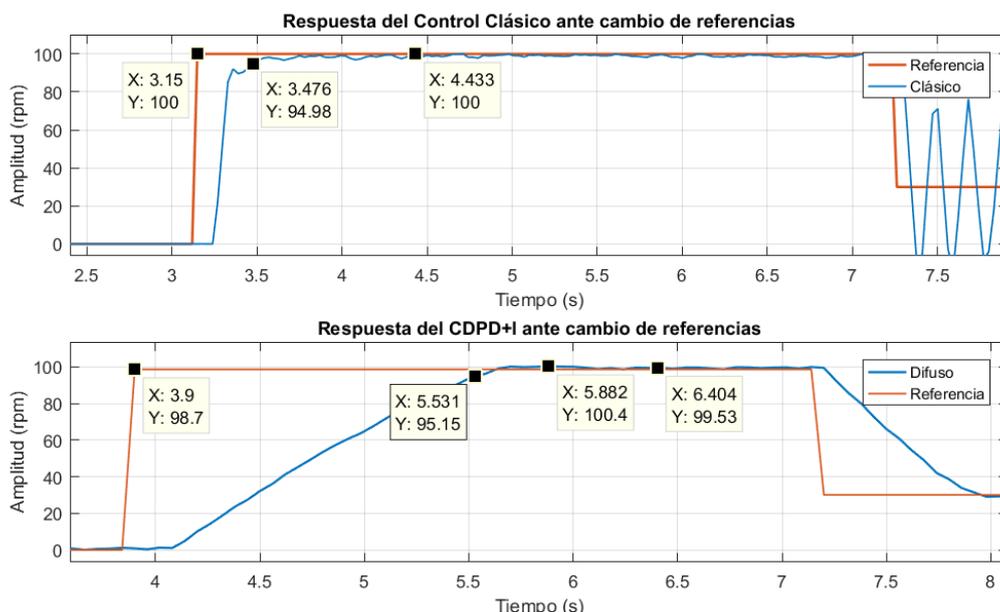


Figura 106. Comparación entre el controlador clásico y el controlador difuso PD+I en un escalón de 100rpm.

Se puede observar rápidamente que la respuesta de control clásico versus el control PD+I presenta mejores tiempos y no tiene sobre elongación para el nivel de entrada mostrado. En la Tabla XVII se muestra los valores obtenidos en esta prueba.

Tabla XVII. Valores obtenidos de los controladores difuso y clásico para una entrada de 100rpm.

Tabulación de valores obtenidos para la comparación en 100rpm						
Controlador	Referencia (0-100rpm)	Salida (0-100rpm)	E_{ee}	Sobreelongación ($Mp\%$)	Tiempo de Subida (s)	Tiempo de Establecimiento(s)
Clásico	100	100	0	0	0.33	1.28
PD + I Difuso	98.7	99.53	-0.83	1.72%	1.63	2.5

4.2 ANÁLISIS DE RESULTADOS

A lo largo del desarrollo de este trabajo, se han podido obtener diferentes resultados que han dependido de los requerimientos tanto del sistema completo como de las partes que lo componen. El primero que se puede mencionar, es con respecto a las pruebas realizadas en el Transductor, en donde, se aproximó la constante K (relación tensión-velocidad) para tratar de obtener una lectura lo más apegada posible a la que presenta el tacómetro del banco hidráulico HLA-70. Como se mostró en la Tabla X, los resultados del error relativo de la aproximación realizada con respecto a la del módulo, se encuentra por debajo del 1% en todas las mediciones, refiérase a la sección 3.3. Estos resultados indicaron que la aproximación ejecutada es apropiada y permitió trabajar con normalidad en el desarrollo posterior del proyecto.

El segundo factor que se puede recalcar, es el acondicionamiento de la señal de trabajo (realimentación). En este punto, se trabajó con tres posibilidades: el filtro pasivo, filtros digitales y la media o promedio móvil. Como se mostró en el capítulo 3, los dos primeros casos no mostraron una respuesta positiva ante el comportamiento de la señal de trabajo dentro del rango completo, filtrando solo en niveles de ingreso específicos, en algunos casos aportando un offset sobre la señal y además de un retraso, revise la sección 3.2. Para el caso del promedio móvil, fue el método que se acopló de mejor manera a la señal del sistema en todo el rango de operación. Para este proceso no se observó el aporte de un offset, a más de reducir los niveles de ruido y el desfase de la señal tratada con respecto a la original fue mínimo, obsérvese a la Figura 72 y Figura 73.

En cuanto a los controladores; como se expuso en las secciones previas de este capítulo, se obtuvieron distintos tipos y comportamientos. El primero de ellos, pertenece al controlador PD difuso. Se le consideró un diseño inestable e ineficiente ya que no respondió en la prueba ante cambios de referencia y tampoco llegaba a los niveles de ingreso esperados, lo que dificultó la medición de los ítems planteados en el plan piloto. En la región superior de la Figura 97, se reveló que el controlador mostraba una leve estabilización en el intervalo comprendido entre 3 y 5 segundos, cuando la planta operaba al nivel máximo (100rpm). Sin embargo, la estabilidad observada se encontraba muy por debajo (34rpm) del nivel límite. Es decir, la permanencia del controlador se daba, pero no en la referencia ingresada; lo que manifestaba principalmente: picos elevados en la

respuesta, un error estado estacionario grande y un tiempo de establecimiento de 3 segundos, el más alto entre los controladores desarrollados.

El controlador PI mostró un comportamiento mejor con respecto al controlador anterior. En este punto del desarrollo, se notó una respuesta en estado permanente ventajosa en el intervalo cercano a los 45 y 55 rpm, permitiendo medir por primera vez todos los ítems que se mencionaron en el plan piloto, Tabla XIV. En la Figura 98, se comprobó que los picos en la respuesta, se expresaban de manera más reducida tanto en amplitud como en tiempo. Los resultados más apreciables revelaron un tiempo de subida de 0.78 segundos, un tiempo de establecimiento de 1.8 segundos, una sobreelongación de 37% y un error en estado estable de 0.32. No obstante, en la prueba de cambios de referencia el controlador no respondió en todo el rango de operación, solamente en el intervalo indicado. Un cuestionamiento que se pudo solventar con el controlador PI, fueron las pruebas frente a perturbaciones. Para estas pruebas se interfirió el flujo normal del aceite, esto se realizó al cerrar reducidamente la válvula principal del sistema, cuando el sistema trabajaba en el rango funcional. Como se mostró en la Figura 99, el controlador se reponía en el nivel de referencia establecido cuando se anulaba la perturbación. Cabe recalcar que, además de los resultados descritos, otra contribución trascendente de éste controlador era el indicativo de que el desarrollo apuntaba hacia el alcance de los objetivos.

El último controlador de naturaleza difusa que se desarrolló fue el denominado PD+I. En la Figura 100, se presentó los resultados ante la prueba de cambios de referencia, en donde, se demostró que el controlador cumplió satisfactoriamente el ensayo, llegando a todos los cambios, redujo la sobre-elongación a 2.5%, el error en estado estacionario disminuyó considerablemente hasta 0.23 y el tiempo de establecimiento fue de 1.22 segundos, el más bajo entre los controladores difusos.

El diseño del controlador clásico se realizó con la meta específica de realizar una comparación con el PD+I. Dicho diseño se hizo cuando el motor operaba en su máxima velocidad. En este punto, como se indicó en la Figura 104, se sometieron tanto el controlador difuso como el clásico al experimento de cambios de referencia, pero considerando condiciones similares, en donde se pudo constatar que el control PD+I funcionaba con normalidad ante estas variaciones, sin embargo, el control clásico solo mostró una respuesta aceptable en el nivel máximo de operación de la planta, es decir, en

las 100 rpm; el nivel de ingreso para el que fue proyectado. La segunda prueba para la comparación entre estos dos sistemas, fue la que corresponde a el ingreso de perturbaciones. En esta prueba, se expone que el controlador PD+I responde favorablemente, retomando el nivel de referencia en el cual se dio la perturbación, después de anular dicha alteración. Para el caso del controlador clásico se observa que su experimentación solo es realizable en el nivel de 100 rpm. En esta amplitud, el control clásico se repone adecuadamente ante la perturbación, recuperándose más rápido que el PD+I en el nivel indicado, refiérase a la Figura 105.

La última confrontación que se cumplió entre el controlador PD+I y el tradicional, corresponde a la prueba ante una entrada de 100rpm. Como se publicó en la Figura 106 y la Tabla XVII, los resultados del controlador clásico fueron mejores que los presentados por el controlador difuso; mostrando mejores tiempos, un error en estado estacionario nulo y no presenta sobre elongación. No obstante, este desempeño mostrado por el clásico solo surgía para una amplitud de 100 revoluciones, mientras que el difuso respondía en todo el rango de operación.

4.3 CONCLUSIONES

En este proyecto de titulación ha sido posible diseñar e implementar un controlador difuso para el mando de velocidad del motor hidráulico perteneciente al banco HRE HIDRAULIC; un sistema caracterizado por su no linealidad. La aplicación del controlador se realizó exitosamente considerando tres pruebas: cambios de referencia, ingreso de perturbaciones y la comparación contra un controlador clásico diseñado en un punto de operación. Las pruebas arrojaron resultados alentadores para el controlador que se propuso, el cual, demostró tener un desempeño en general sobresaliente.

También, en este trabajo se pudieron desarrollar otros factores no menos relevantes para el avance del mismo. El estudio del arte facilitó establecer los métodos usados en el control difuso; del cual se utilizó uno de ellos y se pudo conocer cómo proceder ante sistemas de características similares. Tanto el software como el hardware usados permitieron proceder con el funcionamiento del sistema y el progreso posterior del controlador. Más adelante, se instauraron algunos ítems para tratar de medir o cuantificar los resultados de los controladores, a través del plan piloto. Finalmente, los datos recopilados, graficados y tabulados de los ensayos elaborados acreditaron el funcionamiento del controlador.

4.4 RECOMENDACIONES

Es recomendable que el sistema HRE HIDRAULIC trabaje por debajo de la corriente máxima especificada por el fabricante, es decir, inferior a los 0.8A. Como se especificó en la prueba de perturbaciones, la corriente máxima proporcionada al sistema fue de 0.65A; un valor que puede servir de base para otras aplicaciones o trabajos a futuro.

Para trabajar con el banco hidráulico HLA-70 se recomienda previamente hacer una inspección de las conexiones del sistema, esto es: parte hidráulica, parte eléctrica y la parte mecánica. En esta última, poner más énfasis en el acople (conocido como matrimonio, término mecánico) que se realiza entre el eje del motor y el Transductor, ya que, en el proceso de desarrollo de este trabajo se presentaron leves desajustes en este acople; teniendo que ejecutar en repetidas ocasiones calibraciones para obtener datos apropiados. En este punto se aconsejaría realizar un mantenimiento correctivo completo previo a empezar el avance de un proyecto, o para asegurar su funcionamiento se sugiere cambiar dicho ensamble coordinando con los encargados del laboratorio.

Para los trabajos a futuro; en el caso de que se requiera ampliar el rango de velocidad de trabajo del motor por encima de los 100rpm (el sistema si presenta las condiciones) se recomienda gestionar o realizar una adquisición de un transductor con un rango de lectura más amplio que el rango especificado por el dispositivo en este trabajo (0-10VDC). Si se trabaja con el elemento actual fuera del límite indicado, el mecanismo presenta de valores atípicos (outliers). Otra mejora que se podría considerar estaría en examinar trabajos con sensores virtuales o similares, ya sea para ampliar el rango de operación o añadir una mejora, prescindiendo de esta manera del sensor físico.

El controlador difuso presentado en este trabajo no es definitivo, está dispuesto a mejoras, o referencias futuras tanto para el sistema involucrado como para semejantes.

REFERENCIAS BIBLIOGRÁFICAS

- [1] Erick Narváez, Pablo Sáenz, Daniel Calle, Marco Carpio. “Determinación experimental del modelo matemático de un motor hidráulico e implementación del control de velocidad”, Universidad Politécnica Salesiana Sede Cuenca. (trabajo no publicado), pp. 1-6.
- [2] N. H. Tran, C. Le and A. D. Ngo, "Experimental investigation of speed control of hydraulic motor using proportional valve," 2017 International Conference on System Science and Engineering (ICSSE), Ho Chi Minh City, 2017, pp. 330-335.
- [3] K. Sinthipsomboon, I. Hunsacharoonroj, J. Khedari, W. Pongaen and P. Pratumsuwan, "A hybrid of fuzzy and fuzzy self-tuning PID controller for servo electro-hydraulic system," 2011 6th IEEE Conference on Industrial Electronics and Applications, Beijing, 2011, pp. 220-225.
- [4] M. R. Nayeri, S. Khosravi and M. Aliyari Sh., "Design and implementation double input fuzzy PI and fuzzy gain scheduling PI controller on hydraulic plant," 2013 13th Iranian Conference on Fuzzy Systems (IFSC), Qazvin, 2013, pp. 1-5.
- [5] R. Solorzano, "Principios Ingenieriles Básicos: Motores Hidráulicos", Hydraulics & Pneumatics, 2016. [En Línea]. Disponible en: <https://goo.gl/nF5Ng4>. [Accedido: 20-mzo-2018]
- [6] J. Sánchez y W. Barrera, “Diseño de controladores PID difusos de las variables de nivel y temperatura de una plataforma de entrenamiento para control de procesos continuos”, trabajo de fin de grado, Universidad Politécnica Salesiana, 2015 [En Línea]. Disponible en:
<http://dspace.ups.edu.ec/handle/123456789/10188> [Accedido: 20-mzo-2018]
- [7] A. Rodríguez, “Planta electrónica para entrenamiento de sistemas de control utilizando la interface NI MyDAQ y el software LabVIEW 2010”, trabajo de fin de grado, Universidad Politécnica Salesiana, 2013 [En Línea]. Disponible en: <https://dspace.ups.edu.ec/handle/123456789/4502> [Accedido: 20-mzo-2018]
- [8] J. Orellana y A. Cascante, “Modelado y control de posición de un cilindro de doble efecto utilizando PID y PID-Difuso con MATLAB y LabVIEW”, trabajo de fin de grado, Universidad Politécnica Salesiana, 2017 [En Línea]. Disponible en: <https://dspace.ups.edu.ec/handle/123456789/14554> [Accedido: 20-mzo-2018]

- [9] D. Arteaga y E. Tobar, “Diseño e implementación de módulos de entrenamiento con aplicaciones de control PID de un caldero de una cámara de estabilidad utilizando la interface NI MyDAQ y el software LabVIEW”, trabajo de fin de grado, Universidad Politécnica Salesiana, 2014 [En Línea]. Disponible en: <https://dspace.ups.edu.ec/handle/123456789/7385> [Accedido: 20-mzo-2018]
- [10] K. Ogata, “Ingeniería de Control Moderna”, Madrid, España: Pearson, 2010, pp. 8-9.
- [11] O. Reinoso, J. Zúñiga, R. Aracil y F. Torres, “Control de Sistemas Discretos”, Madrid, España: McGraw-Hill, 2004, pp. 215-217.
- [12] T. Ross, “Fuzzy logic with engineering applications”. Universidad de Nuevo México, USA. Tercera Edición, 2010.
- [13] Pedro Ponce Cruz. "Inteligencia Artificial con Aplicaciones a la Ingeniería". Alfaomega Grupo Editor, S.A. de C.V., México 2010. Primera Edición.
- [14] I. Hernandez, C. Ochoa, “Control difuso y construcción de un mecanismo capaz de golpear con distintos efectos una bola de billar”; Capítulo 4. Sección del Controlador Difuso. Colección de Tesis Digitales, Universidad de las Américas Puebla. [En Línea]. Disponible en: http://catarina.udlap.mx/u_dl_a/tales/documentos/lep/hernandez_b_ii/ [Accedido: 10- Feb- 2018].
- [15] C. González Morcillo. "Lógica Difusa. Una introducción Práctica". Técnicas de Softcomputing. [En Línea]. Disponible en: http://www.esi.uclm.es/www/cglez/downloads/docencia/2011_Softcomputing/LogicaDifusa.pdf [Accedido: 10- Feb- 2018].
- [16] R. Maguiña Pérez. "Sistemas de inferencia basados en Lógica Borrosa: Fundamentos y caso de estudio". Universidad Nacional Mayor de San Marcos. Facultad de Ingeniería de Sistemas e Informática. [En Línea]. Disponible en: <http://revistasinvestigacion.unmsm.edu.pe/index.php/sistem/article/viewFile/3270/2733> [Accedido: 10- Ab- 2018].
- [17] B. Bede, “Mathematics of Fuzzy Sets and Fuzzy Logic”. Springer-Verlag Berlin Heidelberg 2013.
- [18] Orozco-Tupacyupanqui, W.; Nakano-Miyatake, M. & Perez-Meana, H. “A Novel Neural-Fuzzy Method to Search the Optimal Step Size for NLMS”. Beamforming. IEEE Latin America Transactions, 2015, 13, 402-408.

- [19] Orozco-Tupacyupanqui, W.; Pérez-Meana, H. & Nakano-Miyatake, M. “A new fuzzy logic application to search the optimal step size for NLMS beamforming”. 2015 38th International Conference on Telecommunications and Signal Processing (TSP), 2015, 1-6
- [20] Mata Quevedo J.; “Diseño de un controlador difuso para la estabilización de flujos de retorno y de salida en un circuito de molienda de cemento”, trabajo de fin de grado. Maestría en Control y Automatización Industrial, Universidad Politécnica Salesiana, 2016 [En Línea]. Disponible en:
<https://dspace.ups.edu.ec/bitstream/123456789/12675/1/UPS-CT006490.pdf>
[Accedido: 30-abr-2018]
- [21] J. Medel, M. Zagaceta y R. Orozco, “Identificación con estimación para sistemas tipo caja negra”, Rev. Fac. Ing. Univ. Antioquia, vol. 72, pp. 35-46, 2014 [En Línea]. Disponible en: <http://www.redalyc.org/articulo.oa?id=43031750004>
[Accedido: 07-abr-2018]
- [22] X. Ochoa, “Elaboración de un módulo guía para el desarrollo de prácticas en el laboratorio de control de la Universidad Politécnica Salesiana”, trabajo de fin de grado, Universidad Politécnica Salesiana, 2016 [En Línea]. Disponible en:
<https://dspace.ups.edu.ec/handle/123456789/12789> [Accedido: 25-mzo-2018]
- [23] M. Witters and J. Swevers, “Black-box model identification for a continuously variable, electro-hydraulic semi-active damper,” Mech. Syst. Signal Process., vol. 24, no. 1, pp. 4–18, Jan. 2010 [Online]. Available:
<https://www.sciencedirect.com/science/article/pii/S0888327009001241> [Accesed: Apr 07, 2018]
- [24] E. Nolivos, “Control del perfil de voltaje a través de la determinación óptima de taps aplicada en MATLAB”, trabajo de fin de grado, Universidad Politécnica Salesiana, 2015 [En Línea]. Disponible en:
<https://dspace.ups.edu.ec/handle/123456789/8085> [Accedido: 28-mzo-2018]
- [25] F. Meza y P. Ramos, “Modelo matemático motor DC conexión independiente”, trabajo de fin de grado, Universidad Politécnica Salesiana, 2015 [En Línea]. Disponible en: <https://dspace.ups.edu.ec/handle/123456789/10257> [Accedido: 28-mzo-2018]
- [26] “MATLAB®”, MathWorks, Natick, MA, USA, Start Guide, mar. 2018 [En Línea]. Disponible en: https://www.mathworks.com/help/pdf_doc/matlab/getstart.pdf
[Accedido: 03-abr-2018]

- [27] "Fuzzy Logic Toolbox", MathWorks. [En Línea]. Disponible en: <https://goo.gl/X4nX1h>. [Accedido: 02-abr-2018]
- [28] J. Herón y M. Morales, "HMAT: software educativo de visualización y simulación de métodos numéricos con Matlab R2010a", en 2da Conf. Binacional de Investigación, Ciencia y Tecnología de las Universidades, 2013, pp. 586-588. [En Línea]. Disponible en: <https://www.dspace.ups.edu.ec/handle/123456789/11166> [Accedido: 05-abr-2018]
- [29] "Simulink: Simulación y diseño basado en modelos", MathWorks. [En Línea]. Disponible en: <https://la.mathworks.com/products/simulink.html>. [Accedido: 02-abr-2018]
- [30] R. Costa, M. Vallés, L. M. Jiménez, L. Díaz, A. Valera y R. Puerto, "Integración de dispositivos físicos en un laboratorio remoto de control mediante diferentes plataformas: Labview, Matlab y C/C++", Rev. Iberoamericana de Automática e Informática Industrial, vol. 7, pp. 23-34, 2010 [En Línea]. Disponible en: <https://goo.gl/NV2Cqi>. [Accedido: 05-abr-2018]
- [31] J. Lajara y J. Pelegrí, LabVIEW: Entorno gráfico de programación. Barcelona, España: marcombo, 2011 [En Línea]. Disponible en: <https://goo.gl/EJ6PtC>. [Accedido: 04-abr-2018]
- [32] R. Ramírez y R. Reyes, "Diseño e implementación de un robot autónomo móvil usando tecnología FPGA", trabajo de fin de grado, Universidad Politécnica Salesiana, 2015 [En Línea]. Disponible en: <https://dspace.ups.edu.ec/handle/123456789/10429> [Accedido: 05-abr-2018]
- [33] C. Tapia y H. Manzano, "Evaluación de la plataforma Arduino e implementación de un sistema de control de posición horizontal", trabajo de fin de grado, Universidad Politécnica Salesiana, 2013 [En Línea]. Disponible en: <https://dspace.ups.edu.ec/handle/123456789/5522> [Accedido: 03-abr-2018]
- [34] "Arduino Uno Rev3", Arduino. [En Línea]. Disponible en: <https://store.arduino.cc/usa/arduino-uno-rev3>. [Accedido: 03-abr-2018]
- [35] "Arduino Mega 2560 Rev3", Arduino. [En Línea]. Disponible en: <https://store.arduino.cc/usa/arduino-mega-2560-rev3>. [Accedido: 03-abr-2018]
- [36] M. Ugarte, M. Terrón, L. Chávarri y V. Padrón, "Uso de herramientas tecnológicas para el aprendizaje experiencial y la formación integral del ingeniero en el campo de la electrónica", en 11vas Jornadas Internacionales de Innovación Universitaria,

- pp. 338-344. [En Línea]. Disponible en: <http://abacus.universidadeuropea.es/handle/11268/3617> [Accedido: 05-abr-2018]
- [37] J. Cardona, G. Arvizo, M. Carrillo, F. Rodríguez, S. González y G. Ramírez, “Diseño e implementación de un oxímetro de pulso con Labview y la NI MyDAQ”, *Cultura Científica y Tecnológica. CULCYT.*, vol. 55, pp. 199-212, 2015 [En Línea]. Disponible en: <https://goo.gl/a7fXk2> [Accedido: 07-abr-2018]
- [38] "HRE Hidraulic", HRE Hidraulic. [En Línea]. Disponible en: <http://www.hre.es/>. [Accedido: 07-abr-2018]
- [39] “Sistema Didáctico para el Aprendizaje de Tecnologías de Automatización OLEO HIDRÁULICA ELECTRO-HIDRÁULICA HIDRÁULICA PROPORCIONAL”, HRE Hidraulic, Guipúzcoa, España [En Línea]. Disponible en: <https://goo.gl/Ww5h9T> [Accedido: 09-abr-2018]
- [40] “Aplicaciones de Hidráulica Proporcional: PARAMETRIZACION Y MANDO DE AMPLIFICADORES PROPORCIONALES PZD - PWD POR MEDIO DE proPXD”, HRE Hidraulic, Guipúzcoa, España, Manual de Prácticas, abr. 2006.
- [41] J. Bucheli y L. Velásquez, “Diseño e implementación de un módulo didáctico con sistema HMI para el análisis y estudio de sensores y transductores de proximidad analógicos por medio de la tarjeta de adquisición de datos DAQ NI y controlada por el software Labview”, trabajo de fin de grado, Universidad Politécnica Salesiana, 2014 [En Línea]. Disponible en: <https://dspace.ups.edu.ec/handle/123456789/6334> [Accedido: 10-abr-2018]
- [42] A. Creus, “Instrumentación Industrial”, Barcelona, España: MARCOMBO, 2010, pp. 323.
- [43] “7A/160W dual DC motor drive module, high-power industrial grade, positive and negative, PWM speed, L298 logic”, Drok [Online]. Disponible en: <https://images-na.ssl-images-amazon.com/images/I/A1NIP9cXpHL.pdf> [Accedido: 20-ago-2018]
- [44] iluvatar, “Cambio de frecuencia del PWM en el Arduino Mega”, 02-ago-2011. [Entrada en un foro]. Disponible en: <https://forum.arduino.cc/index.php?topic=68281.0>. [Accedido: 20-ago-2018].
- [45] O. Alan y W. Ronald, *Discrete-Time Signal Processing*. Harlow, England. Pearson, 3th ed, 2014, pp. 19-21.
- [46] C. G. Hernández Capacho, M. G. Quijano Ortega y otros, “Obtención experimental de los parámetros del motor que se utilizará en el sistema de locomoción de una

- esfera rodante.”, trabajo de fin de grado, Universidad Pontificia Bolivariana, 2009
[En línea]. Disponible en: <https://goo.gl/9SzJmQ> [Accedido: 22-ago-2018]
- [47] S. Kouro and R. Rodrigo, “Control Mediante Lógica Difusa”, pp. 2–3. [En línea].
Disponible en: <http://www2.elo.utfsm.cl/~elo377/documentos/Fuzzy.pdf>.
[Accedido: 25-ago-2018].
- [48] M. Pérez, J. Álvarez, J. Campo, F. Ferrero y G. Grillo, *Instrumentación Electrónica*.
Universidad de Oviedo: Escuela Politécnica Superior de Ingeniería de Gijón.
Thomson, 2da ed, 2008, pp. 592-593.

APÉNDICES

APÉNDICE A: CÓDIGO IMPLEMENTADO EN MATLAB® PARA GENERAR FUNCIONES DE MEMBRESÍA.

```
function[]=Graficas_FP()

%TRIANGULAR
a=1; b=3; c=4; x=0:0.01:5;
m1=(x>=a&x<=b).*((x-a)/(b-a))+(x>b&x<=c).*((c-x)/(c-b));
figure(1)
plot(x,m1,'k','linewidth',3)
axis([0 5 0 1.1])
xlabel('x')
ylabel('μA(x)')
grid on
%NORMAL
centro=1;
num=x-centro;
den=0.35;
m2=exp(-((num/den)).^2)
figure(2)
plot(x,m2,'k','linewidth',3)
axis([0 2 0 1.1])
xlabel('x')
ylabel('μA(x)')
grid on
%TRAPEZOIDAL
a=1; b=2; c=4; d=5; x=0:0.01:6;
m3=(x>=a&x<b).*((x-a)/(b-a))+(x>=b&x<=c).*1 + (x>c&x<=d).*((d-x)/(d-c));
figure(3)
plot(x,m3,'k','linewidth',3)
axis([0 6 0 1.1])
xlabel('x')
ylabel('μA(x)')
grid on
%SIGMOIDAL
a=1; b=3; p=2; x=0:0.01:5;
m4=(x>a&x<=p).*((2*(((x-a)/(b-a)).^2))+1)+(x>p&x<b).*((1-2*(((x-b)/(b-a)).^2))) +
(x>=b).*1);
figure(4)
plot(x,m4,'k','linewidth',3)
axis([0 4 0 1.1])
xlabel('x')
ylabel('μA(x)')
grid on
```

APÉNDICE B: CÓDIGO IMPLEMENTADO EN ARDUINO PARA TRANSFORMAR UNA SEÑAL ANALÓGICA A PWM.

```
int salida=10; /*pin para el PWM*/
int val=0; /*variable*/
int analog=0; /*número del pin analógico que leerá la señal proveniente de la
MyDAQ*/
void setup()
{
  Serial.begin(9600); /*Para usar el monitor serial*/
  TCCR2B = TCCR2B & 0b000 | 0x06; /*Configura la frecuencia del PWM en
122Hz*/
}
void loop()
{
  val=analogRead(analog); /*Lectura de la señal proveniente de la MyDAQ*/
  analogWrite(salida,val/4); /*Escritura de la señal PWM*/
}
```

APÉNDICE C: CÓDIGO IMPLEMENTADO EN MATLAB® PARA DETERMINAR EL VALOR RMS.

```
clear all; close all; clc;
deci=4; %Número de decimales con los que se trabaja
div=(10^deci);
[t,S1]=textread('30_75.txt','%s %s'); %Carga de datos (nombre y #column)
[t,S2]=textread('32_89.txt','%s %s'); %Carga de datos (nombre y #column)
[t,S3]=textread('39_3.txt','%s %s'); %Carga de datos (nombre y #column)
[t,S4]=textread('48_4.txt','%s %s'); %Carga de datos (nombre y #column)
[t,S5]=textread('56_95.txt','%s %s'); %Carga de datos (nombre y #column)
[t,S6]=textread('67_75.txt','%s %s'); %Carga de datos (nombre y #column)
[t,S7]=textread('86_21.txt','%s %s'); %Carga de datos (nombre y #column)
[t,S8]=textread('135_10.txt','%s %s'); %Carga de datos (nombre y #column)
[t,S9]=textread('277_7.txt','%s %s'); %Carga de datos (nombre y #column)
%Conversión de datos de string a double con precisión de 4 decimales
S1=str2double(S1)/div; S1=S1(1:cant,:);
S2=str2double(S2)/div; S2=S2(1:cant,:);
S3=str2double(S3)/div; S3=S3(1:cant,:);
S4=str2double(S4)/div; S4=S4(1:cant,:);
S5=str2double(S5)/div; S5=S5(1:cant,:);
S6=str2double(S6)/div; S6=S6(1:cant,:);
S7=str2double(S7)/div; S7=S7(1:cant,:);
S8=str2double(S8)/div; S8=S8(1:cant,:);
S9=str2double(S9)/div; S9=S9(1:cant,:);
datos=[S1 S2 S3 S4 S5 S6 S7 S8 S9]; %Array de datos
y=zeros(1,9);
%Cálculo del valor RMS
for i=1:9
a=datos(:,i);
calc_rms=rms(a-mean(a));
y(i)=calc_rms;
end
%Ploteo de datos
y=y';
x=[30.75 32.89 39.3 48.4 56.95 67.75 86.21 135.10 277.7]; x=x';
plot(x,y); grid on; xlabel 'Frecuencia[Hz]'; ylabel 'Valor RMS'
```

APÉNDICE D: CÓDIGO IMPLEMENTADO EN MATLAB® PARA OBTENER LA TRANSFORMADA RÁPIDA DE FOURIER DE UNA SEÑAL Y PARA EL DISEÑO DE FILTROS DIGITALES.

```

clear all; close all; clc;
% =====
% Carga y acondicionamiento de datos
% =====
s=xlsread('out.xlsx');
fs=30/1000; % Frecuencia de muestreo 30ms
L = length(s);
t=[1/fs:1/fs:length(s)/fs];
T=1/fs; % Período de muestreo
% =====
% Transformada Rápida de Fourier
% =====
NFFT=2^nextpow2(L) ; % New Fast Fourier Transform
M=abs(fft(s)); % Magnitud de la FFT
A=angle(fft(s)); % Ángulo de la FFT
xF=[fs/length(s):fs/length(s):floor(fs/2)]; % floor Redondea hacia menos infinito.
M=M(1:length(xF));
A=A(1:length(xF));
% Rangos de los ejes para las graficas en frecuencia
[yfmax,pos]=max(M(2:length(M)));
% freq=xF(1,pos); % Frecuencia del pico mas alto
xfmin=0; xfmax=fs/2;
yfmin=0;
% =====
% Diseño del filtro (pasa bajo)
% =====
fc=input('Ingrese la Frecuencia de corte (fc): '); % Frecuencia de corte para el
filtro
fe=fs/2;
wp=fc;
ws=round(fc+100);
Wp=wp/fe; % Normalización de frecuencias
Ws=ws/fe;
Rp=0.1; % atenuacion en la banda de paso (rizo dB)
Rs=40; % atenuacion en la banda de rechazo (rizo dB)
op=input('Tipo de filtro (1)butter (2)cheb1ord (3)cheb2ord (4)ellipord: ');
if(op==1)
[n,Wp] = buttord(Wp,Ws,Rp,Rs); % Calculo del mínimo orden del filtro
[b,a] = butter(n,Wp); % Diseño del filtro
elseif (op==2)
[n,Wp] = cheb1ord(Wp,Ws,Rp,Rs); % Calculo del mínimo orden del filtro
[b,a] = cheby1(n,Rp,Wp); % Diseño del filtro
elseif (op==3)
[n,Wp] = cheb2ord(Wp,Ws,Rp,Rs); % Calculo del mínimo orden del filtro

```

```

[b,a] = cheby2(n,Rs,Ws); %Diseño del filtro
elseif (op==4)
[n,Wp] = ellipord(Wp,Ws,Rp,Rs);%Calculo del mínimo orden del filtro
[b,a] = ellip(n,Rp,Rs,Wp); %Diseño del filtro
End
[H,w] = freqz(b,a,NFFT); %Respuesta del filtro digital
display('Orden del filtro: ')
n
% =====
% Filtrado de la señal y FFT
% =====
sf = filter(b,a,s); % Se aplica el filtro a la señal en el tiempo
MF=abs(fft(sf)); %FFT de la señal filtrada (magnitud)
MF=MF(1:length(xF));
AF=angle(fft(sf)); %FFT de la señal filtrada (ángulo)
AF=AF(1:length(xF));
% =====
% Impresión de gráficas en Frecuencia
% =====
%Gráfica de la FFT de la señal sin filtrar
figure(1)
subplot(3,1,1)
plot(xF,M); hold on;
title('Espectro de la señal sin filtrar'),...
xlabel('Frecuencia [Hz]');
ylabel('Magnitud');
axis ([xfmin xfmax yfmin yfmax])
grid on
%Gráfica de la respuesta del filtro en frecuencia
subplot(3,1,2)
plot(w*fs/(2*pi),abs(H));
title('Respuesta del filtro');
xlabel('Frecuencia [Hz]');
ylabel('Magnitud');
grid on;
axis ([0 550 -0.1 1.1])
%Gráfica de la FFT de la señal filtrada
subplot(3,1,3)
plot(xF,MF); hold on;
title('Espectro de la señal filtrada'),...
xlabel('Frecuencia [Hz]');
ylabel('Magnitud');
axis ([xfmin xfmax yfmin yfmax])
grid on; hold off;
% =====
% Impresión de gráficas en Tiempo
% =====
xmin=min(t); xmax=max(t);
ymin=min(s); ymax=max(s);
figure (2)

```

```

subplot(3,1,1)
plot(t,s); %Señal sin filtrar
title('Señal sin filtrar'),...
xlabel('Tiempo [s]');
ylabel('Amplitud');
grid on
axis([xmin xmax ymin ymax])
subplot(3,1,2)
plot(t,sf);%Señal filtrada
title('Señal Filtrada'),...
xlabel('Tiempo [s]');
ylabel('Amplitud');
axis([xmin xmax ymin ymax])
grid on
%Señal sin filtrar y señal filtrada transpuestas
subplot(3,1,3)
plot(t,s);
hold on
plot(t,sf,'Color','r','LineWidth',1.1);
title('Comparación de señales'),
legend('Sin filtrar', 'Filtrada');
xlabel('Tiempo [s]');
ylabel('Amplitud');
axis([xmin xmax ymin ymax])
grid on; hold off;

```

APÉNDICE E: CÓDIGO IMPLEMENTADO EN ARDUINO PARA MEDIR LA VELOCIDAD DEL EJE DEL TRANSDUCTOR MEDIANTE EL SENSOR ÓPTICO *H21A1*.

```
const int sensor=40;          // Número del pin al que se conecta el sensor
const int t_Antirebote=10;  //10ms de tiempo de reposición
const int entrada=52;
const int botonR=53;

int c=-1;
int estadoS;
int estadoAS;
int estadoB;
int ban=0;
int reset;
int flag=0;
unsigned int tiempo=0;
int muestras=0;

void setup ()
{
  Serial.begin(56000); /*Velocidad de comunicación serial*/
  Serial.println("CLEARDATA"); /*Limpiamos los datos de la hoja de Excel*/
  Serial.println("LABEL,Hora, Muestra, Velocidad"); /*Nombre que tomará las
columnas de Excel*/
  pinMode(sensor,INPUT); //declaramos el boton como entrada
  pinMode(entrada,INPUT);
}

void loop ()
{
  estadoB=digitalRead(entrada);
  reset=digitalRead(botonR);
  if(reset==HIGH)
  {
    c=-1;
    ban=0;
    muestras=0;
  }
  if(estadoB==HIGH)
  {
    ban=1;
  }

  if(ban==1)
  {
    estadoS=digitalRead (sensor);
    if (estadoS != estadoAS)
```

```

        {
        if (antirebote (sensor))
        {
            c++;
            muestras++;
            Serial.print("DATA,TIME,");
            Serial.print(muestras);
            Serial.print(",");
            Serial.println(c);
        }
        }
        estadoAS= estadoS;
    }
}

/*Función antirebote*/
boolean antirebote (int input )
{
    int contador =0;
    boolean estado;
    boolean estadoAnterior;

    do
    {
        estado = digitalRead (input);
        if (estado != estadoAnterior){
            contador = 0;
            estadoAnterior = estado;
        }
        else
        {
            contador = contador +1;
        }
        delay (1);
    }
    while (contador < t_Antirebote);
    return estado;
}

```

APÉNDICE F: DIMENSIONES DE DISEÑO DE LA CAJA PARA EL HARDWARE DEL SISTEMA EN CONJUNTO

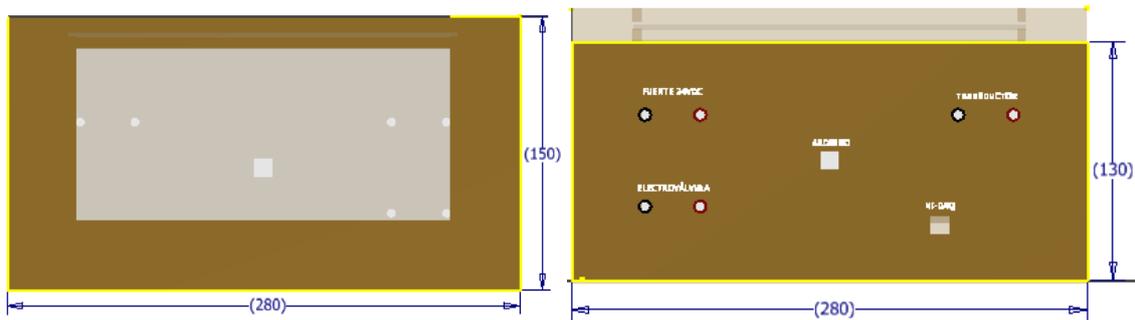


Figura F.1 De izquierda a derecha: Vista Frontal. Vista Posterior.

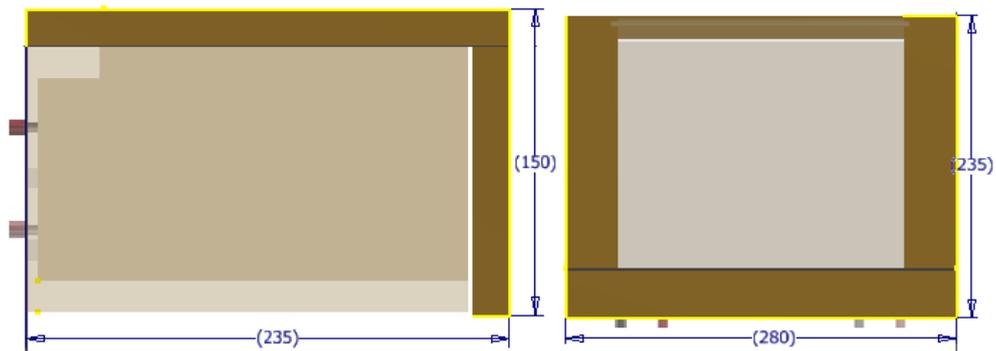


Figura F.2 De izquierda a derecha: Vista Lateral. Vista Superior.

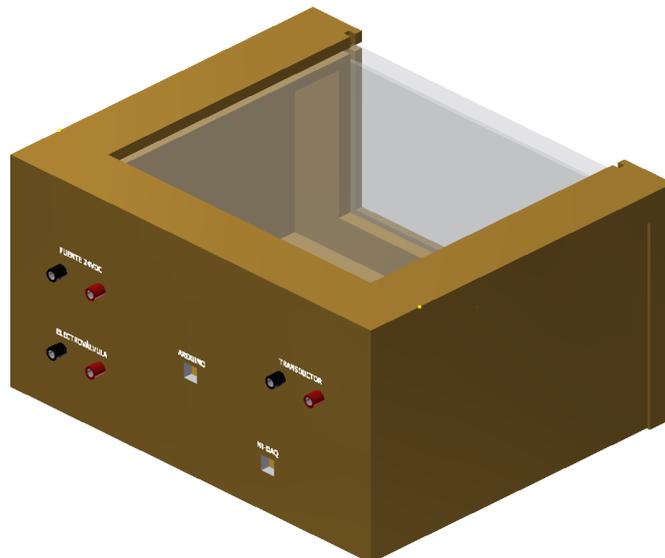


Figura F.3 Caja final implementada.

APÉNDICE G: REGLAS IMPLEMENTADAS PARA LOS CONTROLES DIFUSOS

1. If (error is NG) and (error-derivado is C) then (SC is Poco) (1)
2. If (error is NP) and (error-derivado is NP) then (SC is Medio) (1)
3. If (error is NP) and (error-derivado is C) then (SC is Medio) (1)
4. If (error is NP) and (error-derivado is PP) then (SC is Medio) (1)
5. If (error is C) and (error-derivado is NG) then (SC is Medio) (1)
6. If (error is C) and (error-derivado is NP) then (SC is Medio) (1)
7. If (error is C) and (error-derivado is C) then (SC is Nada) (1)
8. If (error is C) and (error-derivado is PP) then (SC is Nada) (1)
9. If (error is C) and (error-derivado is PG) then (SC is Medio) (1)
10. If (error is PP) and (error-derivado is NP) then (SC is Medio) (1)
11. If (error is PP) and (error-derivado is C) then (SC is Medio) (1)
12. If (error is PG) and (error-derivado is NP) then (SC is Alto) (1)
13. If (error is PG) and (error-derivado is C) then (SC is Muy-Alto) (1)
14. If (error is PP) and (error-derivado is PP) then (SC is Medio) (1)
15. If (error is PG) and (error-derivado is PP) then (SC is Medio) (1)

Figura G.1 Reglas implementadas para el Control Difuso PD

1. If (Error is NegativoGrande) or (ErrorIntegrado is NegativoGrande) then (Control is Cero) (1)
2. If (Error is Negativo) or (ErrorIntegrado is NegativoGrande) then (Control is CasiCero) (1)
3. If (Error is Cero-) or (ErrorIntegrado is Negativo) then (Control is Bajo) (1)
4. If (Error is PositivoGrande) or (ErrorIntegrado is PositivoGrande) then (Control is Grande) (1)
5. If (Error is Negativo) or (ErrorIntegrado is Negativo) then (Control is Cero) (1)
6. If (Error is Cero+) or (ErrorIntegrado is Negativo) then (Control is Bajo+) (0)
7. If (Error is Positivo) or (ErrorIntegrado is PositivoGrande) then (Control is CasiGrande) (1)
8. If (Error is PositivoGrande) or (ErrorIntegrado is Positivo) then (Control is Grande-) (1)
9. If (Error is NegativoGrande) or (ErrorIntegrado is Negativo) then (Control is Bajo) (1)
10. If (Error is Cero-) or (ErrorIntegrado is Positivo) then (Control is CasiCero) (0)
11. If (Error is Negativo) or (ErrorIntegrado is Negativo) then (Control is Cero) (1)
12. If (Error is Cero+) or (ErrorIntegrado is Positivo) then (Control is CasiCero) (0)
13. If (Error is Cero-) or (ErrorIntegrado is Negativo) then (Control is CasiCero) (1)
14. If (Error is Cero-) or (ErrorIntegrado is NegativoGrande) then (Control is Bajo) (1)
15. If (Error is Cero+) or (ErrorIntegrado is PositivoGrande) then (Control is Grande-) (1)
16. If (Error is Cero+) or (ErrorIntegrado is Positivo) then (Control is Grande-) (0)
17. If (Error is PositivoGrande) or (ErrorIntegrado is Negativo) then (Control is Bajo+) (1)
18. If (Error is Positivo) or (ErrorIntegrado is Positivo) then (Control is Grande) (1)

Figura G.2 Reglas implementadas para el Control Difuso PI

1. If (error is NG) and (error-derivativo is NG) then (SC is MB) (1)
2. If (error is NG) and (error-derivativo is NM) then (SC is MB) (1)
3. If (error is NG) and (error-derivativo is N) then (SC is mB) (1)
4. If (error is NG) and (error-derivativo is C) then (SC is B) (1)
5. If (error is NG) and (error-derivativo is P) then (SC is B) (1)
6. If (error is NG) and (error-derivativo is PM) then (SC is C) (1)
7. If (error is NG) and (error-derivativo is PG) then (SC is C) (1)
8. If (error is N) and (error-derivativo is NG) then (SC is MB) (1)
9. If (error is N) and (error-derivativo is NM) then (SC is MB) (1)
10. If (error is N) and (error-derivativo is N) then (SC is B) (1)
11. If (error is N) and (error-derivativo is C) then (SC is B) (1)
12. If (error is N) and (error-derivativo is P) then (SC is C) (1)
13. If (error is N) and (error-derivativo is PM) then (SC is A) (1)
14. If (error is N) and (error-derivativo is PG) then (SC is A) (1)
15. If (error is C) and (error-derivativo is NG) then (SC is mB) (1)
16. If (error is C) and (error-derivativo is NM) then (SC is B) (1)
17. If (error is C) and (error-derivativo is N) then (SC is B) (1)
18. If (error is C) and (error-derivativo is C) then (SC is C) (1)
19. If (error is C) and (error-derivativo is P) then (SC is A) (1)
20. If (error is C) and (error-derivativo is PM) then (SC is A) (1)

21. If (error is C) and (error-derivativo is PG) then (SC is mA) (1)
22. If (error is P) and (error-derivativo is NG) then (SC is B) (1)
23. If (error is P) and (error-derivativo is NM) then (SC is B) (1)
24. If (error is P) and (error-derivativo is N) then (SC is C) (1)
25. If (error is P) and (error-derivativo is P) then (SC is A) (1)
26. If (error is P) and (error-derivativo is PM) then (SC is MA) (1)
27. If (error is P) and (error-derivativo is PG) then (SC is MA) (1)
28. If (error is P) and (error-derivativo is C) then (SC is A) (1)
29. If (error is PG) and (error-derivativo is NG) then (SC is C) (1)
30. If (error is PG) and (error-derivativo is NM) then (SC is C) (1)
31. If (error is PG) and (error-derivativo is N) then (SC is A) (1)
32. If (error is PG) and (error-derivativo is C) then (SC is A) (1)
33. If (error is PG) and (error-derivativo is P) then (SC is mA) (1)
34. If (error is PG) and (error-derivativo is PM) then (SC is MA) (1)
35. If (error is PG) and (error-derivativo is PG) then (SC is MA) (1)

Figura G.3 Reglas implementadas para el Control Difuso PD+I

APÉNDICE H: CÓDIGO IMPLEMENTADO EN MATLAB® PARA LA ESTIMACIÓN DE LA FUNCIÓN DE TRANSFERENCIA DEL SISTEMA EN CONJUNTO.

```
clear all; close all; clc;
deci=4;
div=(10^deci);
[t,imp,SO,SM2,SM3,SM4]=textread('2.5.txt','%s %s %s %s %s %s');
t=str2double(t)/div;%String ==> double
imp=str2double(imp)/div;
SO=str2double(SO)/div;
SM2=str2double(SM2)/div;
SM3=str2double(SM3)/div;
SM4=str2double(SM4)/div;

maxT=max(t);
maxIMP=max(imp);
minIMP=min(imp);

figure
plot(t,imp,'Color','r', 'Linewidth',1.1); axis([0 maxT minIMP-1 maxIMP+1]);
hold on;
plot(t,SO,'Color',[0.8500, 0.3250, 0.0980], 'Linewidth',1);
hold on; plot(t,SM3,'Color',[0, 0.4470, 0.7410], 'Linewidth',0.5);
xlabel('Tiempo [s]');
ylabel('Amplitud [rpm]');
grid on;
legend('Impulso','Señal ruidosa','Promedio (3)');
hold off;

figure
u=imp; %entrada
y=SM3; %respuesta
Ts=0.03; %Tiempo de muestreo en segundos
NP=3; %Número de polos
NZ=3; %Número de zeros
IODELAY=NaN; %tiempo muerto del sistema
data=iddata(y,u,Ts);
sys=tfest(data,NP,NZ,IODELAY)
zpk(sys)
compare(sys,data); grid on;
```