

**DESARROLLO DE UN CONTROLADOR FUZZY PARA IMPLEMENTAR EN UN
PLC SIEMENS S7-300**

BRIAN ALEXANDER PEDRAZA RODRÍGUEZ

HÁROL ALEXÁNDER SUÁREZ ORTIZ

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICOMECÁNICAS
ESCUELA DE INGENIERÍA MECÁNICA
BUCARAMANGA**

2018

**DESARROLLO DE UN CONTROLADOR FUZZY PARA IMPLEMENTAR EN UN
PLC SIEMENS S7-300**

BRIAN ALEXANDER PEDRAZA RODRÍGUEZ

HÁROL ALEXÁNDER SUÁREZ ORTIZ

TRABAJO DE GRADO PARA OPTAR EL TÍTULO DE INGENIERO MECÁNICO

DIRECTOR

JORGE ENRIQUE MENESES FLÓREZ

M.Sc. Ingeniería mecánica

CODIRECTOR

LUIS MIGUEL ZABALA GUALTERO

M.Sc. Ingeniería automotriz

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICOMECÁNICAS**

ESCUELA DE INGENIERÍA MECÁNICA

BUCARAMANGA

2018

DEDICATORIA

“A mi hija Helena, cuyo ser ilumina e irradia de energía todo mi existir.

*A mi Esposa Julieth, que sin importar las circunstancias, ni el cómo ni el dónde
siempre decidió jugársela conmigo.*

A mi Madre Alba, el ser al que le agradezco la vida y gran parte de lo que soy.

*A mis compañeros, que siempre estuvieron para brindar apoyo, ayuda y lo más
importante "MOMENTOS".*

*A mi compañero Brian, por brindar la ayuda y la posibilidad de culminar esta
etapa”*

Hárol Alexander Suárez Ortiz

*“A mi madre Alejandra, a mi padre Carlos, por su apoyo incondicional y la
motivación que me entregan en cada una de las etapas.*

*A mis amigos y compañeros de universidad quienes brindaron su apoyo, su
disposición y su alegría.”*

Brian Alexander Pedraza Rodríguez

AGRADECIMIENTOS

Al profesor Jorge Enrique Meneses Flórez, director del proyecto y pieza fundamental en el desarrollo del mismo, agradecemos la confianza puesta en nosotros, los consejos y las enseñanzas que nos brindó en cada una de las etapas, siendo de gran utilidad para nuestra formación personal y profesional.

A los compañeros del laboratorio de Automatización Industrial, por su disposición y su apoyo brindado para resolver inconvenientes.

TABLA DE CONTENIDO

	Pág.
INTRODUCCION	20
1. FORMULACION DEL PROBLEMA	22
1.1 DESCRIPCION DEL PROBLEMA	22
1.2 JUSTIFICACION PARA LA SOLUCIÓN DEL PROBLEMA	23
1.3 ANTECEDENTES INVESTIGATIVOS	23
2. OBJETIVOS DEL TRABAJO DE GRADO	25
2.1 OBJETIVO GENERAL	25
2.2 OBJETIVOS ESPECÍFICOS	25
3. SINTESIS DEL DESARROLLO DEL CONTROLADOR FUZZY IMPLEMENTADO	28
3.1 INTERFAZ GRÁFICA	30
3.1.1 Módulo de Entradas y Salidas.	31
3.1.2 Módulo de reglas.	32
3.1.3 Módulo de carga.	32
3.1.4 Módulo de visualización.	33
3.2 COMUNICACIÓN OPC	34
3.3 COMPONENTE DE CONTROL	35
3.3.1 Preparación de variables del modelo Fuzzy.	39
3.3.2 Bloque FB50 "FuzzyControl".	40
3.3.3 Preparación de las variables de salida.	42
4. DISEÑO DE LA INTERFAZ GRAFICA	43
4.1 ADMINISTRACION DE LOS DATOS EN LA INTERFAZ GRAFICA	44
4.1.1 Estructura de datos de las variables de entrada.	45

4.1.2	Estructura de datos de las variables de salida.	47
4.1.3	Estructura de datos de la tabla de reglas.	48
4.1.4	Estructura de datos de conexión.	49
4.1.5	Estructura de datos de tablas de nombres.	50
4.1.6	Flujo de datos en la interfaz.	51
4.2	FUNCIONAMIENTO DE LA INTERFAZ	52
4.2.1	Edición del modelo Fuzzy.	52
4.2.2	Formulación de la base de reglas del modelo.	55
4.2.3	Transferencia del modelo Fuzzy al PLC	57
4.2.4	Visualización de las señales de la planta.	58
5.	IMPLEMENTACION DE LA COMUNICACIÓN OPC	60
5.1	CONFIGURACION DE LOS ELEMENTOS DEL SERVIDOR OPC	61
5.2	COMUNICACIÓN ENTRE LA INTERFAZ GRAFICA Y EL SERVIDOR OPC	62
5.3	COMUNICACIÓN ENTRE EL SERVIDOR OPC Y EL PLC	64
6.	DESARROLLO DEL COMPONENTE DE CONTROL EN EL PLC	66
6.1	ESTRUCTURA DE LOS BLOQUES DE DATOS GLOBALES	69
6.1.1	Estructura de datos de las variables de entrada.	69
6.1.2	Estructura de datos de las variables de salida.	70
6.1.3	Estructura de datos de la base de reglas.	71
6.1.4	Bloque de datos de parámetros del modelo.	73
6.1.5	Bloque de datos de visualización.	73
6.1.6	Tabla de variables del PLC	74
6.2	BLOQUE DE EJECUCION POR INTERRUPCION OB35	76
6.3	PREPARACIÓN DE VARIABLES DE ENTRADA AL MODELO FUZZY	77

6.3.1	FB30. CapturaSeñales.	77
6.3.2	FB20. SetupReferencias:	78
6.3.3	FB15. Normalización:	78
6.3.4	FB10. SetupVariables.	80
6.4	EJECUCIÓN DE FUZZYCONTROL	81
6.4.1	FB51. B01Pertenencia.	83
6.4.2	FB52. B02Implicación.	85
6.5	PREPARACIÓN DE LAS VARIABLES DE SALIDA A LA PLANTA	89
7.	IMPLEMENTACIÓN DEL CONTROLADOR FUZZY	90
7.1	SISTEMA DE CONTROL DE POSICIÓN ANGULAR (SISO)	90
7.1.1	Caracterización del sistema SISO.	91
7.1.2	Creación de los conjuntos de las variables de entrada.	93
7.1.3	Creación del conjunto de la variable de salida.	95
7.1.4	Creación de la base de reglas de control.	96
7.1.5	Ejecución del modelo Fuzzy creado mediante el FuzzyControl desarrollado.	97
7.1.6	Análisis de la ejecución del modelo Fuzzy creado mediante el FuzzyControl desarrollado.	101
7.2	SISTEMA DE CONTROL DE VELOCIDAD Y TEMPERATURA DEL AIRE (MIMO)	102
7.2.1	Caracterización del sistema MIMO.	103
7.2.2	Creación de los conjuntos de las variables de entrada para sistema MIMO.	106
7.2.3	Creación del conjunto de las variables de salida MIMO.	110
7.2.4	Creación de la base de reglas de control para sistema MIMO.	112

7.2.5	Ejecución del modelo Fuzzy creado para el sistema MIMO mediante el FuzzyControl desarrollado. _____	113
7.2.6	Análisis de la ejecución del modelo Fuzzy para el sistema MIMO creado mediante el FuzzyControl desarrollado. _____	116
8.	CONCLUSIONES _____	118
9.	RECOMENDACIONES _____	119
	BIBLIOGRAFIA _____	120

LISTA DE FIGURAS

	Pág.
Figura 1. Esquema de configuración para control de posición angular (SISO).	26
Figura 2. Esquema de configuración para el control regulado de velocidad y temperatura del aire (MIMO).....	27
Figura 3. Esquema global del controlador Fuzzy.	29
Figura 4. Esquema global de la interfaz gráfica.	30
Figura 5. Visualización de los conjuntos de entrada y salida del modelo Fuzzy. ...	31
Figura 6. Esquema general comunicación OPC.	34
Figura 7. Esquema de control a la planta.....	35
Figura 8. Esquema algoritmo de control.	36
Figura 9. Etapas del algoritmo de control.....	38
Figura 10. Esquema preparación de variables del modelo Fuzzy.....	39
Figura 11. Esquema disposición del FuzzyControl.	41
Figura 12. Esquema preparación variables de salida.	42
Figura 13. Interfaz gráfica diseñada.....	43
Figura 14. Estructura de la tabla de las variables de entrada.	46
Figura 15. Estructura de la tabla de las variables de salida.	47
Figura 16. Estructura de la tabla de las bases de reglas.	48
Figura 17. Estructura de datos variable A y conexión A.....	49
Figura 18. Estructura de la tabla de nombres.	50
Figura 19. Edición de variables de entrada y salida en la interfaz.	52
Figura 20. Modulo creación conjuntos de variables de entrada.	53
Figura 21. Asignación de tipo de conjunto.	54
Figura 22. Modulo creación conjuntos de variables de salida.....	55
Figura 23. Modulo creación base de reglas.	56
Figura 24. Carga del modelo Fuzzy al PLC.	57

Figura 25. Esquema de comparación direcciones – datos para la transferencia de información.	58
Figura 26. Módulo de Visualización.	59
Figura 27. Servidor OPC softing.	60
Figura 28. Zonas de trabajo servidor OPC.....	61
Figura 29. Esquema general de conformación de protocolos de comunicación. ..	62
Figura 30. Configuración del canal de comunicación.....	64
Figura 31. Activación del canal de comunicación y punto de acceso.....	65
Figura 32. Verificación del canal de comunicación.	65
Figura 33. Lazo de control del sistema SISO.....	66
Figura 34. Estructura de bloques de datos de las variables de entrada.....	69
Figura 35. Estructura de bloques de datos de las variables de salida.	71
Figura 36. Estructura de bloques de datos de la matriz de reglas.	72
Figura 37. Estructura de bloques de datos de los parámetros.....	73
Figura 38. Estructura de bloques de datos de los valores de visualización.	74
Figura 39. Flujo de ejecución periódica OB35.....	76
Figura 40. Esquema general de flujo en los bloques de instancia de la captura de señales.	77
Figura 41. Esquema general de flujo en los bloques de instancia de la actualización de referencias.	78
Figura 42. Esquema general de flujo en los bloques de instancia de la normalización.....	79
Figura 43. Tablas de datos de los valores normalizados.	79
Figura 44. Esquema general de flujo en los bloques de instancia de la actualización de variables.....	80
Figura 45. Esquema general de datos en los bloques de instancia del FuzzyControl.	81
Figura 46. Flujo de ejecución del FuzzyControl.	82
Figura 47. Proceso de cálculo de la pertenencia.	84
Figura 48. Proceso de cálculo funciones MVP_IZQ y MVP_DER.....	85

Figura 49. Esquema general de datos en los bloques de instancia de la implicación.	86
Figura 50. Proceso de implicación de reglas Fuzzy.	86
Figura 51. Aplicación del método MIN-MAX.	87
Figura 52. Esquema general de datos en los bloques de instancia defuzificación.	88
Figura 53. Defuzificación por método de pesos concentrados.....	89
Figura 54. Esquema general del proceso de acumulación de la variable de salida.	89
Figura 55. Configuración para sistema de control de posicionamiento angular (SISO).....	91
Figura 56. Gráfica del comportamiento en la caracterización del sistema SISO...	92
Figura 57. Definición de los conjuntos de la variable de entrada "Error de ángulo".	93
Figura 58. Definición de los conjuntos de la variable de entrada "Variación del error".	94
Figura 59. Definición de conjuntos de la variable de salida "Aumento velocidad del motor".	95
Figura 60. Edición de base de reglas determinadas en la interfaz gráfica.	97
Figura 61. Edición del primer valor de referencia para ejecución de pruebas SISO.	98
Figura 62. Edición del segundo valor de referencia para ejecución de pruebas SISO.	99
Figura 63. Edición del tercer valor de referencia para ejecución de prueba SISO.	99
Figura 64. Ejecución de prueba de seguimiento para el sistema SISO.	100
Figura 65. Configuración para sistema de control de velocidad y temperatura del aire (MIMO).....	103
Figura 66. Gráfica del comportamiento de la velocidad en la caracterización del sistema MIMO.....	104

Figura 67. Gráfica del comportamiento de la temperatura en la caracterización del sistema MIMO.....	105
Figura 68. Definición de los conjuntos de la variable de entrada "Error de velocidad".	107
Figura 69. Definición de los conjuntos de la variable de entrada "Variación error de la velocidad".....	107
Figura 70. Definición de los conjuntos de la variable de entrada "Error de temperatura".	108
Figura 71. Definición de los conjuntos de la variable de entrada "Variación error de temperatura".	109
Figura 72. Definición de conjuntos de la variable de salida "Cambio el motor"...	110
Figura 73. Definición de conjuntos de la variable de salida "Cambio en resistencia".	111
Figura 74. Edición del primer conjunto de valores de referencia para ejecución de pruebas MIMO.	114
Figura 75. Edición del segundo conjunto de valores de referencia para ejecución de pruebas MIMO.	115

LISTA DE TABLAS

	Pág.
Tabla 1. Información general dentro de los bloques de datos globales.....	37
Tabla 2. Listado de tablas que componen la base de datos SQL.....	44
Tabla 3. Bloque de programación en el PLC.....	68
Tabla 4. Tabla de variables del PLC.....	74
Tabla 5. Estructura de la variable MP y MVP.....	83
Tabla 6. Designación de nombres para sistema SISO.....	94
Tabla 7. Estructuración de la base de reglas para el sistema SISO.....	96
Tabla 8. Valores de referencia para ejecución de pruebas del sistema SISO.....	97
Tabla 9. Designación de nombres para sistema MIMO.....	109
Tabla 10. Estructuración de la base de reglas para el sistema MIMO.....	112
Tabla 11. Designación adicional de nombre para aplicación de reglas sistema MIMO.....	113
Tabla 12. Valores de referencia para ejecución de pruebas del sistema MIMO..	113

LISTA DE ANEXOS

(Ver anexos adjuntos en el CD y pueden visualizarlos en la base de datos de la Biblioteca UIS)

ANEXO A. Marco teórico relacionado a control Fuzzy

ANEXO B. Representación de números reales bajo estándar IEEE754

ANEXO C. Práctica No.1: “Caracterización de sistemas SISO”

ANEXO D. Práctica No.2: “Sintonización de sistemas SISO”

ANEXO E. Práctica No.3: “Caracterización de sistemas MIMO”

ANEXO F. Práctica No.4: “Sintonización de sistemas MIMO”

RESUMEN

TITULO: DESARROLLO DE UN CONTROLADOR FUZZY PARA IMPLEMENTAR EN UN PLC SIEMENS S7-300*

AUTORES: PEDRAZA RODRÍGUEZ BRIAN ALEXANDER**

SUÁREZ ORTIZ HÁROL ALEXÁNDER

PALABRAS CLAVES: Control Fuzzy, Controlador Lógico Programable PLC, FuzzyControl, lenguaje SCL, interfaz gráfica de usuario, comunicación OPC.

DESCRIPCIÓN:

El objetivo del proyecto consiste en desarrollar un controlador fuzzy orientado a PLC's Siemens S7-300 que permita a los estudiantes de la escuela de ingeniería mecánica poner en práctica los conocimientos adquiridos en control de procesos bajo lógica fuzzy, desarrollando las competencias necesarias para una formación competitiva y productiva en el futuro egresado.

El proyecto integra una interfaz gráfica que permite al usuario la edición del modelo fuzzy y posteriormente puede visualizar el comportamiento de la planta en funcionamiento para efectuar el proceso de análisis y sintonización del control, un programa de control dentro de un PLC el cual contiene un bloque de funciones encargado del proceso de lógica fuzzy y un servidor de datos OPC que se encarga del proceso de comunicación y transferencia de información entre estos dos componentes.

El controlador es puesto a prueba en el ducto de viento para recrear eventos de automatismos, disponible en el laboratorio de Automatización Industrial de la universidad. Dicho banco puede configurarse para un control de posición angular de una compuerta a partir de un flujo de aire (control SISO), o también para el control de la velocidad y temperatura del aire en un ducto de sección circular (control MIMO).

*Proyecto de Grado

** Facultad de Ingenierías Físico-mecánicas. Escuela de Ingeniería Mecánica. Director: MSc. Jorge Enrique Meneses Flórez, Co-director: MSc. Luis Miguel Zabala Gualtero

ABSTRACT

TITLE: DEVELOPMENT OF A FUZZY CONTROLLER TO IMPLEMENT IN A SIEMENS S7-300 PLCⁱ

AUTHORS: PEDRAZA RODRÍGUEZ BRIAN ALEXANDERⁱⁱ

SUÁREZ ORTIZ HÁROL ALEXÁNDER

KEYS WORDS: Fuzzy Control, programmable logic controller plc, FuzzyControl, SCL Language, graphical user interface, OPC communication.

DESCRIPTION:

The aim of the project is to develop a fuzzy controller oriented to PLCs Siemens S7-300 that allows students of the school of mechanical engineering to put into practice the knowledge acquired in process control under fuzzy logic, developing the necessary skills for a competitive training and productive in the future graduate.

The project integrates a graphic interface that allows the user to edit the fuzzy model and later can visualize the behavior of the plant in operation and then carry out the process of analysis and control tuning, a control program inside a PLC which contains a function block responsible for the fuzzy logic process and an OPC data server that is responsible for the process of communication and transfer of information between these two components.

The controller is put to the test in the wind pipe to recreate automation events, available in the Industrial Automation laboratory of the university. This bank may be configured for angular position control of a gate from an airflow (SISO control), or also to control the speed and temperature of air in a duct of circular section (MIMO control).

ⁱBachelor Thesis

ⁱⁱFacultad de Ingenierías Físico-mecánicas. Escuela de Ingeniería Mecánica. Director: M.sc. Jorge Enrique Meneses Flórez, Co-director: MSc. Luis Miguel Zabala Gualtero

INTRODUCCION

El desarrollo tecnológico de la industria colombiana se ha visto enmarcado en la premisa de conseguir que sus procesos productivos sean cada vez más eficientes y con una calidad que permita ser competitivo en el panorama económico internacional e internacional. Para conseguir estas mejoras, se han implementado procesos cada vez más complejos, donde el control de éstos se queda corto ante las estrategias tradicionalmente usadas. De ahí la necesidad de desarrollar nuevas técnicas basadas en inteligencia artificial para el control de procesos industriales.

Dentro de las estrategias de control basadas en inteligencia artificial, el control fuzzy se posiciona como una estrategia ampliamente usada para procesos tecnológicos que dada su complejidad resulta complicado obtener un modelo matemático preciso. Dicha estrategia se basa en el uso de expresiones lingüísticas que permiten manejar la incertidumbre y la ambigüedad utilizadas en la cotidianidad, tales como “temperatura fría” o “presión alta”, lo que permite la formulación de controladores industriales basados en reglas heurísticas que un operador humano usaría para controlar un proceso.¹

El proyecto nace de la necesidad de fortalecer la formación de los estudiantes de la Universidad Industrial de Santander en las diferentes técnicas de control de procesos industriales (siendo el control fuzzy una de ellas) a partir del desarrollo de un objeto de aprendizaje que integra un bloque de funciones en PLC para control fuzzy y una interfaz gráfica que permite la edición de estos modelos y su estudio desde la práctica.

Con el desarrollo de este objeto de aprendizaje se pretende en primera instancia que el estudiante aplique los conocimientos adquiridos en lógica difusa, cuente con la herramienta tecnológica para la elaboración de sus propios modelos de control y

¹ MENESES Jorge. Inteligencia artificial en el control de procesos. Control Fuzzy basado en PLC's Simatic S5. Bucaramanga. Universidad Industrial de Santander, 1998. 212 p.

pueda aplicarlos en un entorno tangible donde pueda hacer el proceso de análisis, para ello se dispuso el ducto de viento para recrear eventos de automatismos disponible en el laboratorio de automatización industrial de la universidad. Por otra parte, el hecho de desarrollar los propios algoritmos en lugar del uso de software donde no conocemos su lógica de programación, siendo este una “caja negra”, abre la posibilidad al investigador para que a partir del uso de esta herramienta tenga un punto de partida para proponer mejoras y estrategias de control más avanzadas.

En el capítulo 1 se plantean los objetivos del proyecto, dándole una delimitación al alcance del mismo y marcando el punto de partida del mismo. En el capítulo 2 se da una descripción global del proyecto, abordando las características principales de cada uno de sus componentes. En el capítulo 3 y 4 se hace profundización en el desarrollo de la interfaz gráfica y la comunicación de ésta con el autómata encargado del control, allí se muestra su configuración y su funcionamiento, además de la integración de la comunicación OPC. Posteriormente en el capítulo 5 se especifica el desarrollo de toda la arquitectura de control dentro del autómata, mostrando la estructura de la información, las etapas del control y el funcionamiento del bloque de control fuzzy (objeto principal del proyecto). El capítulo 6 plantea la serie de pruebas realizadas sobre el banco que se dispuso para el proyecto y en los capítulos 7 y 8 se plantean las conclusiones y recomendaciones a las que se llegó.

1. FORMULACION DEL PROBLEMA

1.1 DESCRIPCION DEL PROBLEMA

Una de las bases fundamentales de la academia es buscar la unificación de contextos con la industria para poder brindar soluciones eficaces y efectivas, puesto que es de suma importancia que la industria funcione para que la académica crezca; bajo esta premisa se presentan diversos inconvenientes debido a las herramientas y canales de comunicación que existen entre éstas dos, generando que la academia en varias ocasiones se quede corta debido a la poca experiencia ofrecida a los estudiantes para que puedan relacionarse directamente con prácticas industriales que les puedan brindar la capacidad de intuición y agilidad para resolver problemas cuando estén en su etapa productiva.

El ingeniero mecánico de la Universidad Industrial de Santander ha venido limitando su desempeño profesional al seguir las tendencias industriales modernas, basadas primordialmente en las nuevas tecnologías de automatización industrial, creando un foco importante a tener en cuenta por la academia para la formación de profesionales en función del crecimiento industrial.

La lógica difusa ha generado como instrumento para su uso industrial los controladores Fuzzy, induciendo a la academia a que éstos sean objeto principal de estudio; desarrollando herramientas y habilidades para su aplicación posterior. En la actualidad no se cuenta con un software comercial que permita desarrollar prácticas de control Fuzzy interactuando con un controlador industrial como lo son los PLC`S, creando una desventaja en la formación de profesionales, surgiendo así la necesidad de implementar métodos para crear en los estudiantes una destreza preferente que ayude en el desarrollo y la implementación de estos controles Fuzzy.

1.2 JUSTIFICACION PARA LA SOLUCIÓN DEL PROBLEMA

Actualmente es significativo el desarrollo y avance tecnológico que se tiene a nivel industrial debido a los retos de cambio e innovación a la que están constantemente acostumbradas las empresas, el ingeniero mecánico de la Universidad Industrial de Santander debe tener las capacidades para enfrentar este tipo de retos generando valor agregado en base a su capacidad y contenido de aporte.

El propósito de este proyecto estuvo encaminado a contribuir en el desarrollo de capacidades y destrezas en la interpretación, análisis, diseño y aplicación de controles FUZZY enfocados a ambientes industriales por medio de controladores lógicos programables (PLC's), creando en los estudiantes de ingeniería mecánica prácticas y experiencias cercanas a la industria para permitir que generen y afiancen conocimientos sobre las técnicas y dispositivos aplicados en controles basados en lógica difusa y poder lograr mejor desempeño laboral cuando se esté en la etapa productiva.

1.3 ANTECEDENTES INVESTIGATIVOS

Se buscaron en módulos investigativos a nivel internacional, nacional y regional, aportes o trabajos relacionados los cuales pueden complementar o aportar en el desarrollo de esta investigación.

El trabajo desarrollado GUTIERREZ William y LOPEZ Jesús², se implementó un controlador difuso tipo Mandani* para el control de nivel de un tanque industrial por medio de un autómata programable. El documento muestra una síntesis de la programación del autómata a partir de AOI (Add-On Instruction) para un sistema

² GUTIERREZ William, LOPEZ Jesús. Control difuso para un sistema de nivel implementado en un autómata programable. En: Informador Técnico (Colombia) Enero – Diciembre 2011, Edición 75, p 11-16.

de dos variables de entrada y una salida, se realiza un control posterior por PD y una comparación de la respuesta del control de ambas estrategias y se concluye la ventaja que tiene el control difuso sobre el PD en función de mejoras en tiempo de respuesta, sobreimpulso y esfuerzo en el control.

La investigación realizada por GUARNIZO Cristian³ plantea una metodología para la implementación de un control difuso tipo Takagi Sugeno** en un PLC S7-300. El algoritmo de control fue desarrollado en SCL de STEP 7 para un control de dos variables de entrada y una variable de salida con distribuciones de conjuntos de tipo triangular para posteriormente mostrar el efecto que tiene la respuesta de salida ante la variación de las variables de sintonización del control.

El seminario taller realizado por MENESES Jorge ⁴ aborda inicialmente la fundamentación teórica de la teoría fuzzy y su campo de aplicación, la arquitectura base del controlador difuso y la programación del mismo en el PLC a partir de bloques de funciones haciendo uso de la herramienta de desarrollo PROFUZZY de Matlab y concluye con una serie de ejemplos de aplicación donde incluye la programación de controladores PID, FUZZY y la combinación de los dos bajo diferentes condiciones.

³ GUARNIZO Cristian. Metodología para la implementación de un controlador difuso tipo Takagi Sugeno en PLC S7-300. Medellín, Mayo de 2011. 8 p.

⁴ MENESES Jorge. Inteligencia artificial en el control de procesos. Control Fuzzy basado en PLC's Simatic S5. Bucaramanga. Universidad Industrial de Santander, 1998. 212 p.

2. OBJETIVOS DEL TRABAJO DE GRADO

2.1 OBJETIVO GENERAL

Fortalecer el propósito misional de la Universidad Industrial de Santander, con la generación y adecuación de conocimientos para las necesidades de las diversas ramas industriales y académicas, en especial la mecatrónica y automatización, a través del desarrollo de un controlador Fuzzy para implementarlo mediante un PLC de marca Siemens referencia S7-300.

2.2 OBJETIVOS ESPECÍFICOS

- Desarrollar un software de control FUZZY en PLC que contenga las siguientes características:
 - Capacidad para 4 entradas, 2 salidas y 7 conjuntos Fuzzy por variable.
 - Variables de entrada con distribuciones de los conjuntos Fuzzy de tipo triangular y de tipo shoulder para los conjuntos a los extremos del universo del discurso.
 - Distribución de los conjuntos en tipo singleton para las variables de salida.
 - Inferencia lógica de las reglas composicionales Fuzzy aplicando el método MIN-MAX.
 - Defuzificación de la salida por el método de pesos concentrados.
 - Interfaz de usuario desarrollada en C# (Visual Studio 2013) que permita la edición y visualización gráfica de las variables, los conjuntos de las variables y formulación de las reglas del control.

- Implementación de los protocolos de comunicación entre la aplicación de usuario y el autómatas basados en OPC (Softing OPC Demo Client) para la transmisión de los parámetros del control y las reglas del control.
 - Algoritmos de control desarrollados en TIA Portal V13 basados en SCL ajustable al número de entradas, salidas y reglas formuladas en la interfaz de usuario e implementados en el PLC Siemens S7-300.
 - Guía de usuario del programa.
- Elaborar prácticas y material académico para el desarrollo de competencias en implementación de Control FUZZY aplicados a autómatas industriales, mediante la siguiente metodología:
 - 2 prácticas para caracterización de sistemas SISO⁵ y MIMO.⁶
 - 1 prácticas para implementación de control FUZZY en sistemas SISO.
 - 1 prácticas para implementación de control FUZZY en sistemas MIMO.

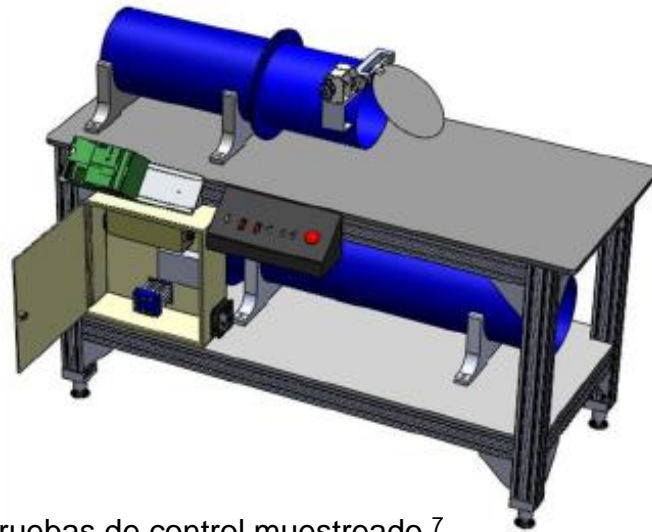
El desarrollo de las practicas se realizó a una planta modelo configurada sobre un banco de pruebas que consta de un ducto de viento tipo modular de sección circular, compuesto por un módulo principal y dos módulos intercambiables; uno para simulación de procesos SISO, realizando el control para su posición angular como se observa en la figura 1, y otro para simulación de procesos MIMO, realizando el control regulado de velocidad y temperatura del aire ver figura 2.

La planta modelo es llamada “Ducto de viento para recrear eventos de automatismo” y está ubicada en el laboratorio de automatización industrial de la escuela de ingeniería mecánica.

Figura 1. Esquema de configuración para control de posición angular (SISO).

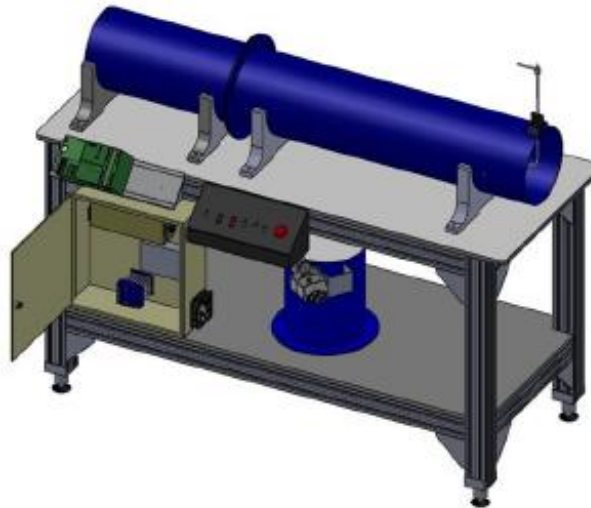
⁵ SISO, De la sigla en inglés single input single output, Una entrada y una salida.

⁶ MIMO, De la sigla en inglés multiple input multiple output, Múltiple entrada y múltiple salida.



Fuente. Banco de pruebas de control muestreado.⁷

Figura 2. Esquema de configuración para el control regulado de velocidad y temperatura del aire (MIMO).



Fuente. Banco de pruebas de control muestreado.³

⁷TARAZONA Oscar, QUINTERO Jeisson. Diseño y Construcción de un objeto de Aprendizaje para prácticas de sistemas de control muestreado, basado en controlador industrial PLC. Tesis de Grado Ingeniero Mecánico, Escuela de Ingeniería Mecánica, Facultad de Ingenierías Físico-Mecánicas, Universidad Industrial de Santander, 2012, 181 p.

3. SINTESIS DEL DESARROLLO DEL CONTROLADOR FUZZY IMPLEMENTADO

El controlador Fuzzy se estructuró en un bloque de funciones único llamado *FuzzyControl* desarrollado en lenguaje SCL⁸ para ser utilizado en PLC's⁹ Siemens de la gama S7-300. Los datos requeridos para la ejecución del control se almacenan en una serie de bases de datos asociados al mismo. En principio, la ventaja principal que posee este módulo radica en que su configuración de uso es totalmente parametrizable, de esta manera puede ser utilizado para realizar control en cualquier planta en donde quiera implementarse, que sirva como una guía de aprendizaje técnico-práctico para los estudiantes de ingeniería mecánica, afianzando durante su proceso de formación los conocimientos en áreas de mecatrónica y control de procesos.

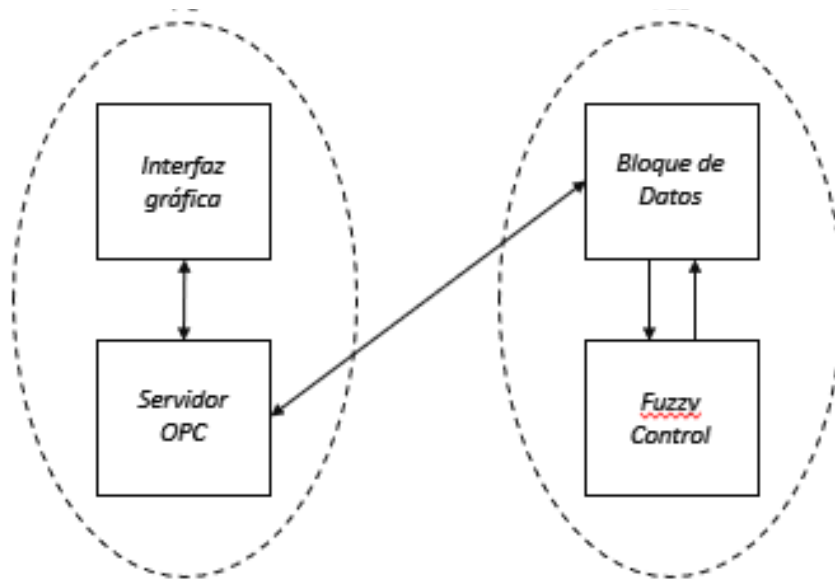
El controlador se desarrolló bajo las premisas de adaptabilidad, interfaz visualmente agradable, veracidad en la información obtenida, comunicación estable con el PLC⁵, respuesta rápida, fácil manejo y utilización de software gratuito o de bajo costo, con el fin de generar un interés latente del usuario y mayor accesibilidad a la herramienta académica para el desarrollo de las prácticas.

En la figura 3, se observa el esquema identificando tres (3) grandes componentes integrados que conforman el controlador.

⁸SCL. De las siglas en ingles structured control language. Lenguaje de control estructurado.

⁹ PLC, PLC's. De las siglas en ingles programable logic controller. Controlador lógico programable.

Figura 3. Esquema global del controlador Fuzzy.



Una interfaz gráfica que permite al usuario editar toda la información correspondiente al modelo Fuzzy para luego transmitirla al PLC. Esta interfaz también permite visualizar en línea el comportamiento de las variables manipuladas y variables controladas.

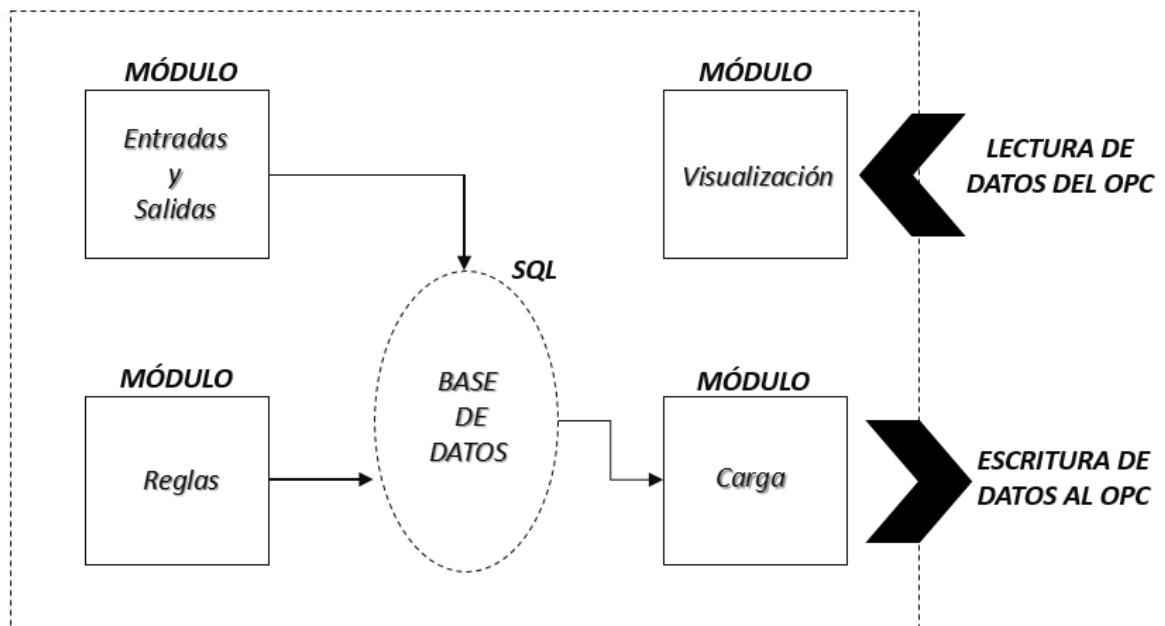
Un servidor OPC que se comunica a partir de unos protocolos de comunicación con la interfaz gráfica y permite la transferencia de datos entre ésta y el PLC.

El bloque *FuzzyControl* es el encargado de realizar el control, el cual busca la información requerida dentro de los bloques de datos. Este está dispuesto dentro de un algoritmo que se ejecuta de en intervalos de tiempo fijos.

3.1 INTERFAZ GRÁFICA

La interfaz de usuario desarrollada en C# comprende 4 grandes módulos que permiten al estudiante editar y visualizar el modelo Fuzzy. Esta tiene integrada una base de datos en SQL que permite almacenar la información del modelo a medida que ocurre la edición para luego ser transmitida a los bloques de datos del PLC en la estructura configurada dentro de dichos bloques.

Figura 4. Esquema global de la interfaz gráfica.

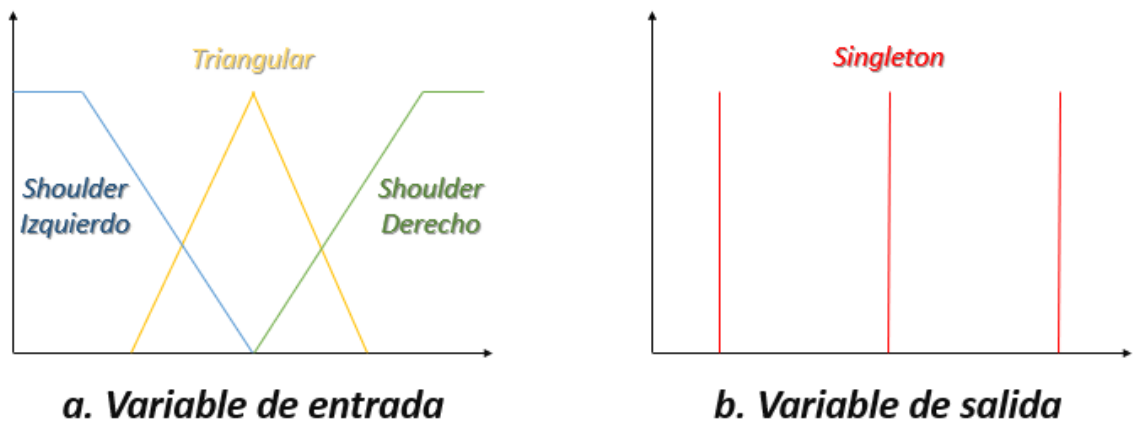


3.1.1 Módulo de Entradas y Salidas.

Este módulo permite inicialmente definir el número de variables de entradas y salidas del modelo Fuzzy con un máximo de cuatro (4) variables de entrada, dos (2) variables de salida y siete (7) conjuntos Fuzzy por variable.

En la edición de las variables de entrada se pueden editar sus valores y configurar los conjuntos en tres (3) diferentes tipos de funciones de pertenencia como se observa en la figura 5a (shoulder izquierdo, triangular o shoulder derecho). En los conjuntos de las variables de salida únicamente se pueden modificar sus valores dentro del universo del discurso, debido a que los conjuntos ya están configurados por defecto para ser una función tipo *singleton*, como se observa en la figura 5b.

Figura 5. Visualización de los conjuntos de entrada y salida del modelo Fuzzy.



Una vez ingresados los conjuntos de las variables, la interfaz tiene la función de almacenar dichos conjuntos dentro de una base de datos en SQL bajo la estructura necesaria para transmitirlos al PLC y enviar sus nombres al módulo de reglas.

3.1.2 Módulo de reglas.

Este módulo permite configurar la base de reglas del modelo Fuzzy. Estas reglas tienen una estructura de tipo “sí – entonces” y están compuestas por una serie de precedentes dados por los conjuntos de las variables de entrada y unos consecuentes dados por los conjuntos de las variables de salida. El módulo contiene una tabla y una serie de controles que permiten ingresar un máximo de 100 reglas.

Al finalizar la edición, se almacena la base de reglas editada dentro de la base de datos SQL bajo la estructura que requiere el bloque de datos en el PLC.

3.1.3 Módulo de carga.

La función principal de este módulo consiste en tomar la información del modelo Fuzzy incluida en la base de datos SQL y transferirla a los bloques de datos dentro del PLC. El módulo contiene una serie de controles que permite al usuario ejecutar las funciones de transferencia y el borrado de la información en los bloques de datos del PLC, además, lleva un control del estado de la información dentro del mismo.

El proceso de transferencia y borrado de datos es posible mediante la función de escritura integrada dentro de los protocolos de comunicación establecida entre la interfaz de usuario y el servidor OPC.

3.1.4 Módulo de visualización.

En este módulo se puede visualizar el comportamiento de las variables medidas y las variables manipuladas en la planta que se está controlando. Para la planta propuesta en este proyecto (Ducto de viento para recrear eventos de automatismo), el módulo contiene un selector que permite visualizar las variables dependiendo de si está configurado como un sistema SISO o un sistema MIMO. La visualización de estas variables es posible a través del evento de lectura incluida dentro de los protocolos de comunicación entre la interfaz y el servidor OPC.

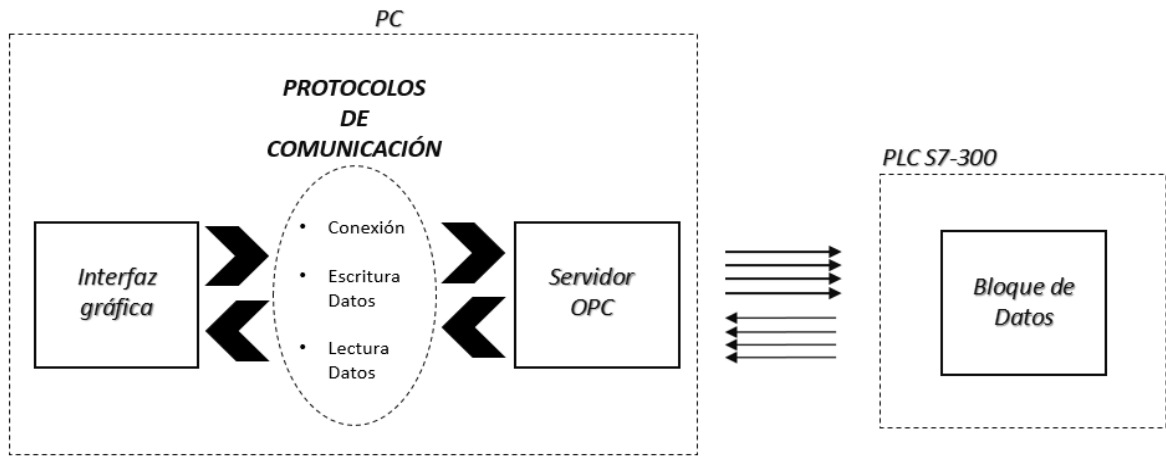
Cuando se selecciona el modo SISO, se indica que la planta está configurada para un control de posición angular (ver figura 1), y el módulo habilita los controles para visualizar el comportamiento de la posición angular en el tiempo siendo esta la variable controlada, y el comportamiento de la potencia del ventilador como variable manipulada.

Para el caso del modo MIMO, la planta está configurada para el control de velocidad y temperatura del aire, por tanto, el módulo habilita las gráficas para visualizar el comportamiento de la velocidad junto con la temperatura del aire y el comportamiento de la potencia del ventilador junto a la potencia entregada por la resistencia.

3.2 COMUNICACIÓN OPC

La comunicación entre la interfaz gráfica y el PLC se realizó a través del estándar OPC (*OLE for Process Control*) utilizando una arquitectura cliente – servidor.

Figura 6. Esquema general comunicación OPC.



Bajo esta arquitectura el servidor funciona como una fuente de datos con la información correspondiente a las variables incluidas dentro de los bloques de datos y variables globales declaradas en el PLC utilizando las direcciones que estas tienen asignadas, por ejemplo, las marcas, las variables de entrada, las variables de salida tanto analógicas como digitales.

Se integró un servidor de acceso a datos (OPC DA) disponible en internet llamado *Softing OPC DA* bajo una versión Demo gratuita, el cual contiene las funciones esenciales para poder comunicar aplicaciones de escritorio con éste, además, permite habilitar el canal de comunicaciones con el PLC vía MPI¹⁰. Por su parte la interfaz gráfica contiene dentro de su programación los protocolos de comunicación

¹⁰ MPI. De las siglas en inglés multi point interface. Interfaz multipunto.

necesarios para conectarse con el servidor y acceder a la información dentro de la memoria de datos del PLC.

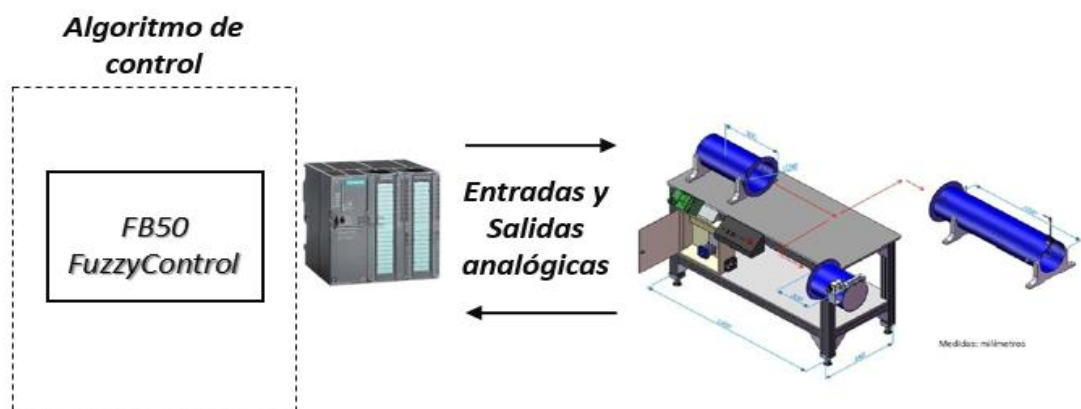
La estructura de estos códigos se puede subdividir en tres partes:

- Establecimiento de la conexión con el servidor.
- Eventos de lectura de datos.
- Funciones de escritura de datos.

3.3 COMPONENTE DE CONTROL

El modelo de control implementado sobre la planta propuesta en el proyecto consiste en un control Fuzzy donde las entradas del modelo son el error y la variación del error de cada una de las variables medidas, y las variables de salida del modelo es un cambio en la potencia entregada al actuador. Al salir del bloque Fuzzy, esta variable se suma al valor total de la salida en el ciclo anterior, en entrega a la planta y se almacena en la memoria del PLC para el siguiente ciclo.

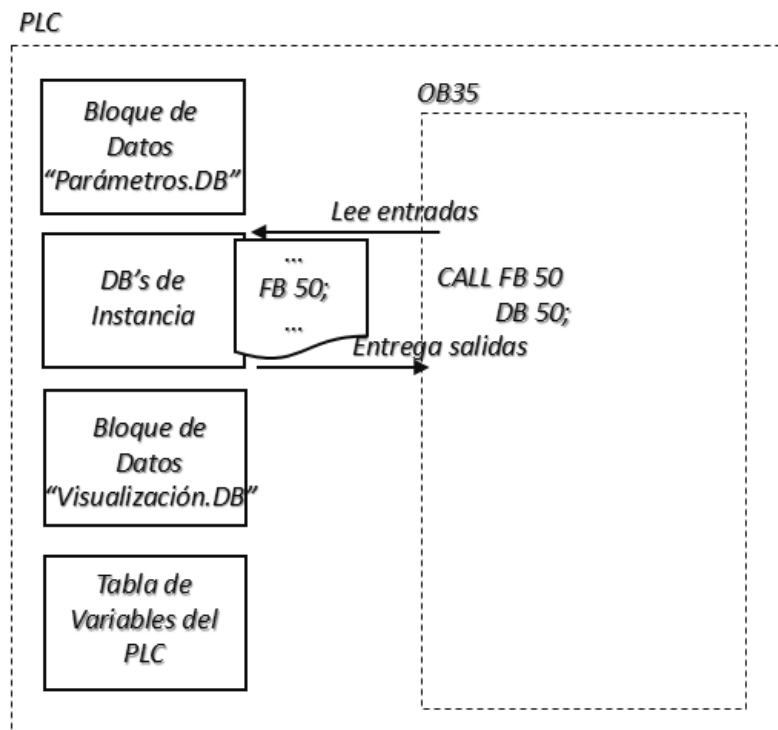
Figura 7. Esquema de control a la planta.



En ese sentido cuando la planta está configurada para el control de posición angular, las variables de entrada son el error de ángulo y la variación del error, y la variable de salida es el cambio en la velocidad del ventilador. Por su parte, cuando la planta está configurada para el control de temperatura y velocidad de aire las variables de entrada son error de velocidad, variación de error de velocidad, error de temperatura y variación del error de temperatura, y las variables de salida son el cambio a la potencia de la resistencia y el cambio a la velocidad del ventilador.

El control Fuzzy es realizado por un bloque de funciones llamado *FuzzyControl*. La información relacionada al modelo Fuzzy, las lecturas de los valores medidos en los sensores que deben enviarse de vuelta a la interfaz y las variables de proceso, deben almacenarse en unos bloques de datos globales y bloques de datos de instancia.

Figura 8. Esquema algoritmo de control.



La figura 8, se muestran los diferentes elementos que componen el algoritmo de control dentro del PLC. El bloque “Parámetros.db” son unos conjuntos de bloques de datos globales que contienen la información del modelo Fuzzy (“conjuntos de entradas, conjuntos de salidas y reglas), que han sido transmitidos desde la interfaz gráfica a través del OPC. Por su parte, el bloque “Visualización.db” almacena la información de las mediciones reales de los sensores y las magnitudes enviadas a los actuadores para que sean leídos en la interfaz gráfica. Los bloques de funciones son llamados a un bloque de ejecución OB35 y sus bloques de instancia definen las entradas, salidas y variables intermedias de cada uno.

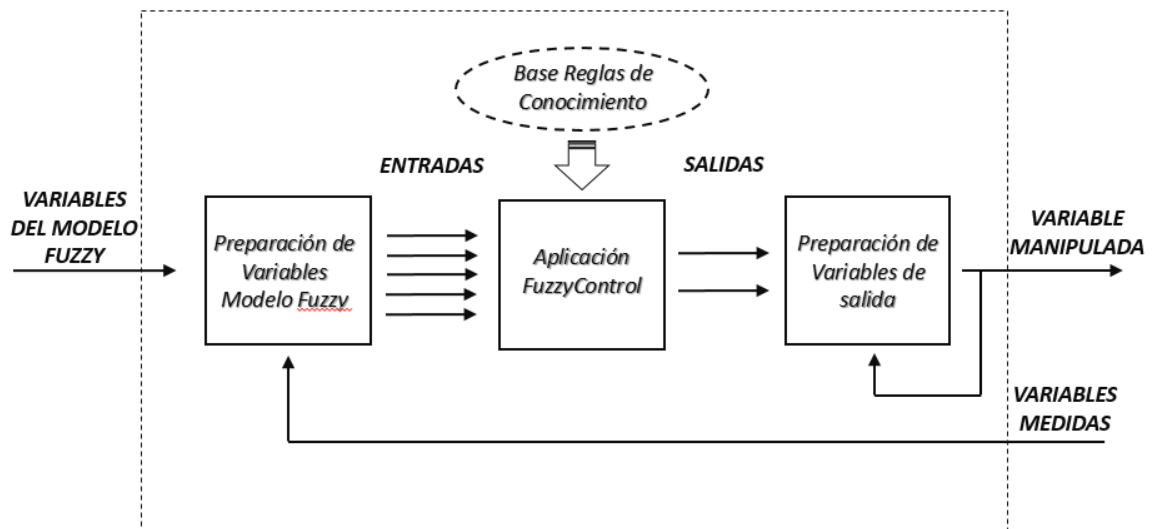
La información dentro de los bloques de datos globales que almacenan el modelo Fuzzy e interactúan con la interfaz se encuentran en la siguiente tabla:

Tabla 1. Información general dentro de los bloques de datos globales.

DB	Nombre	Descripción
DB1	VariableA	Contiene la información de los conjuntos de la entrada A.
DB2	VariableB	Contiene la información de los conjuntos de la entrada B.
DB3	VariableC	Contiene la información de los conjuntos de la entrada C.
DB4	VariableD	Contiene la información de los conjuntos de la entrada D.
DB5	SalidaX	Contiene la información de los conjuntos de la salida X.
DB6	SalidaY	Contiene la información de los conjuntos de la salida Y.
DB7	Base de Reglas	Contiene la lista de reglas composicionales del modelo Fuzzy.
DB8	Parámetros	Contiene información relacionada al número de entradas y salidas, número de reglas y los <i>Setpoint</i> para el control.
DB9	Visualización	Almacena el valor real de las variables medidas y el valor de la salida a los actuadores para visualizarlos vía OPC.

El algoritmo de control se puede dividir en tres (3) etapas como se muestra en la figura 9.

Figura 9. Etapas del algoritmo de control.



Una primera etapa llamada “preparación de variables del modelo Fuzzy” que consiste en tomar la información del modelo Fuzzy, las señales de la planta y normalizarlas a una escala determinada. Posteriormente se evalúan el error y variación del error de cada variable física medida con respecto a su valor de referencia para ingresar al bloque Fuzzy.

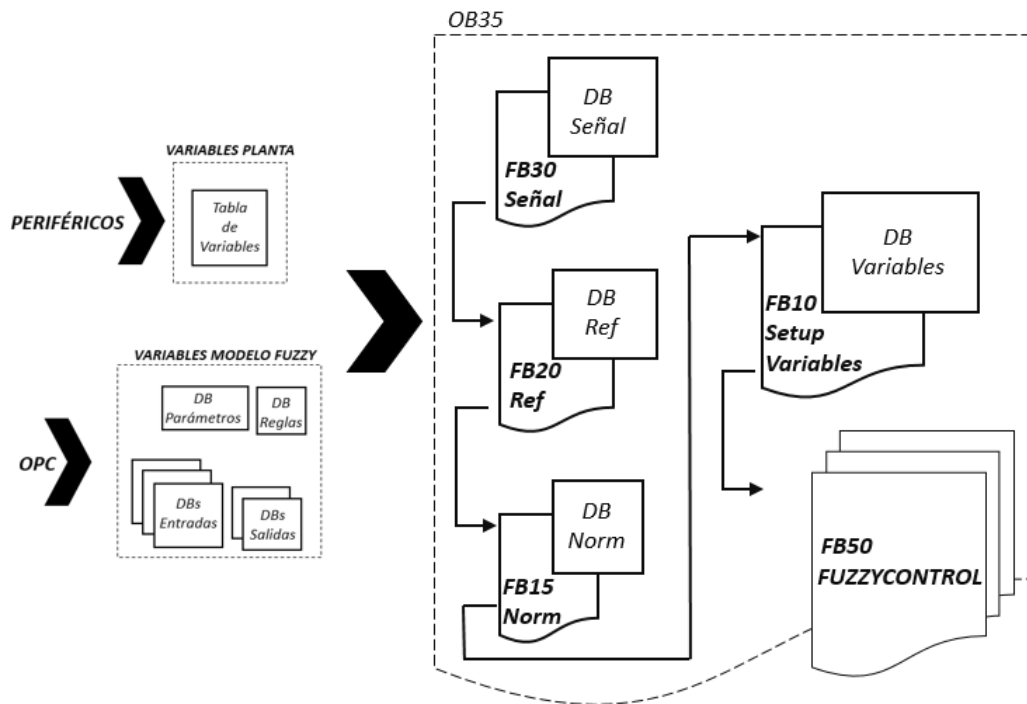
Una segunda etapa en la cual el bloque de ejecución cíclica llama a *FuzzyControl* para ejecutar el proceso de control por lógica Fuzzy.

Una tercera etapa llamada “preparación de variables de salida” en la cual se calcula el valor de la señal de salida a cada uno de los actuadores de la planta.

3.3.1 Preparación de variables del modelo Fuzzy.

Esta etapa es la encargada de recibir el modelo Fuzzy de los bloques de datos y las mediciones de los sensores, normalizar estos datos a una escala ampliada para facilitar la operación de las mismas. En la figura 10, se muestra el orden en que el OB realiza las llamadas a cada uno de los bloques de funciones, en la llamada del bloque de instancia de datos respectivo se especifican las entradas, salidas y los bloques de datos de donde toma dicha información.

Figura 10. Esquema preparación de variables del modelo Fuzzy.



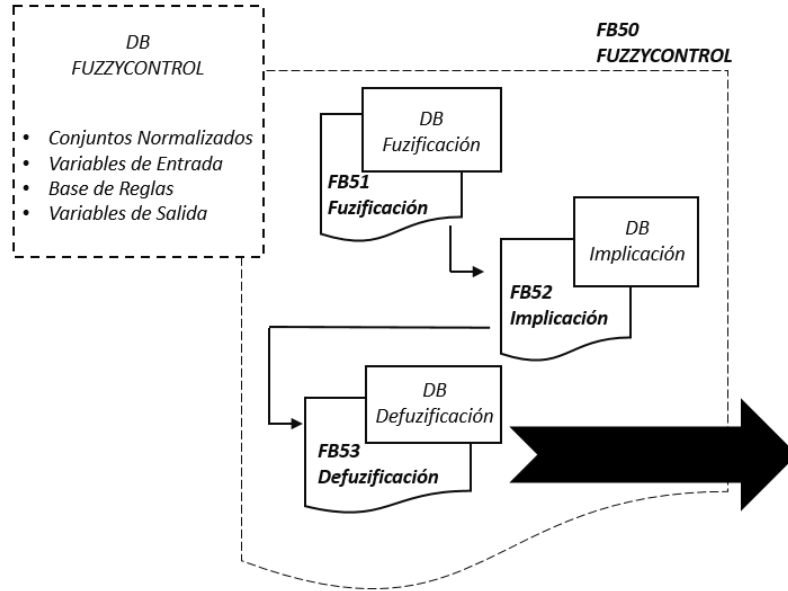
- El FB30 de captura de señales obtiene el valor del canal análogo y lo convierte en un valor entero para determinar la medida de cada uno de los sensores.
- El FB20 de captura de referencias toma el valor de referencia de la variable a controlar y amplifica su valor para ser comparado con los valores medidos en los sensores.
- El FB15 de normalización toma los conjuntos de las entradas y salidas del modelo Fuzzy y los amplifica a una escala determinada para ser comparados con las referencias y las señales medidas.
- El FB10 de Setup de variables que toma las mediciones y los valores de referencia normalizados, calcula las variables de error y derivada del error para que éstas ingresen al bloque de control Fuzzy.

3.3.2 Bloque FB50 “FuzzyControl”.

Este bloque es el encargado de realizar el proceso de control aplicando la lógica Fuzzy. Toma las variables de entrada calculadas en el bloque de Setup de variables, los conjuntos de datos de entrada y salida normalizados, la base de reglas y parámetros como número de entradas, número de salidas y número de reglas. Información ubicada en el bloque de datos “Parámetros.db”. Al finalizar la operación, el bloque entrega los valores de las variables de salida del modelo.

En la figura 11, se observa el orden en que el FB50 “FuzzyControl” hace la llamada de tres (3) funciones en específico.

Figura 11. Esquema disposición del FuzzyControl.

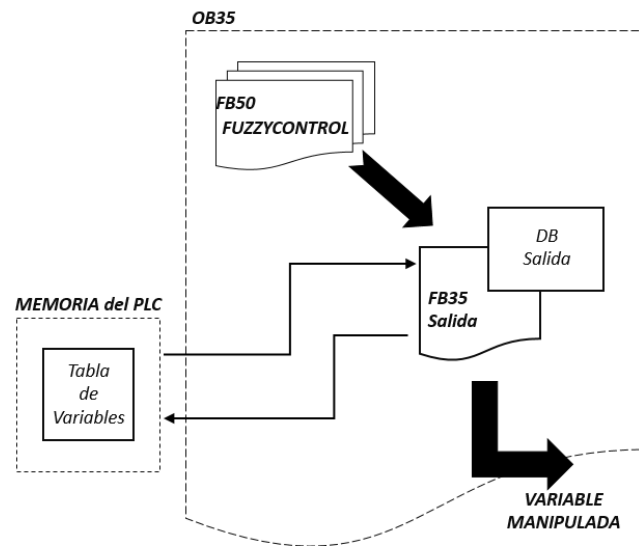


- FB51 “Pertenencia”: Realiza el proceso de inferencia Fuzzy. Toma cada variable de entrada y sus respectivos conjuntos y evalúa el grado de pertenencia (desde 0 a 1) de la variable a uno o varios de sus conjuntos.
- FB52 “Implicación”: Recorre cada una de las reglas composicionales del modelo Fuzzy para verificar cuáles se cumplen y evaluar el valor de pertenencia asociado al conjunto de la variable de salida indicado como consecuencia en la regla dada.
- FB53 “Defuzificación”: Toma los valores de pertenencia asociados a cada conjunto Fuzzy de la variable de salida y evalúa el valor total de la variable de salida del modelo.

3.3.3 Preparación de las variables de salida.

En esta etapa se busca calcular el valor total de la salida que se entrega al canal analógico del actuador. Para ello, el OB35 llama a ejecución al FB35 “Salida”. El cual toma el valor del cambio en la variable de salida proveniente del FB50 “FuzzyControl”, lo suma al valor de la salida total del ciclo anterior, lo almacena de nuevo dentro de una posición de memoria y lo envía al canal analógico siendo ya la variable manipulada.

Figura 12. Esquema preparación variables de salida.

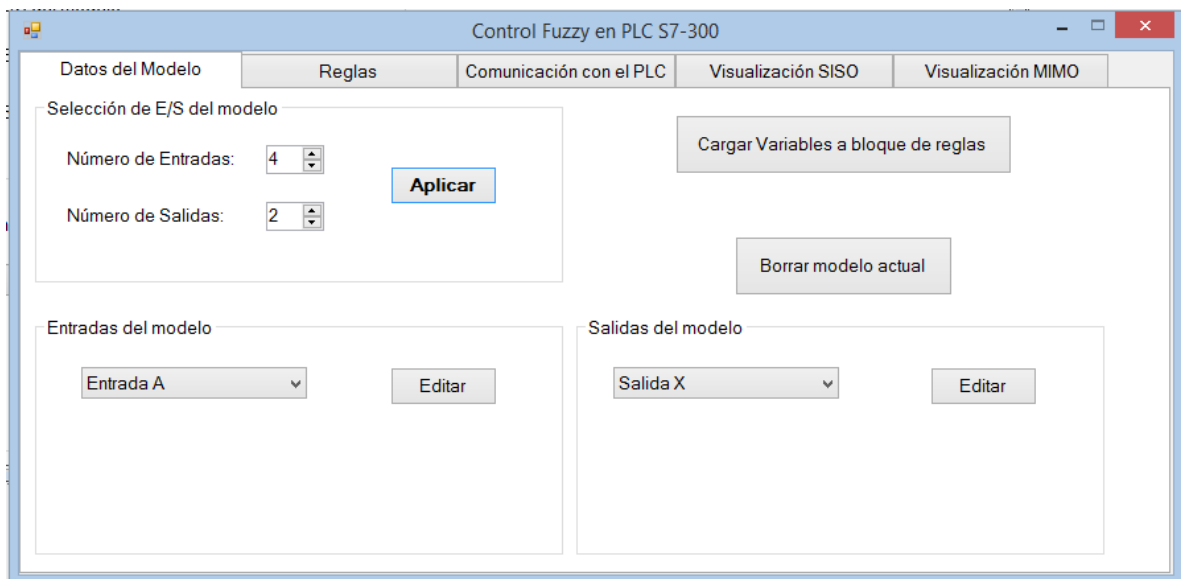


4. DISEÑO DE LA INTERFAZ GRAFICA

El controlador diseñado cuenta con una interfaz gráfica programada de manera externa al PLC como se observa en la figura 13. Esta fue desarrollada en Visual Studio 2013, cuya licencia se encuentra disponible de manera gratuita para la comunidad UIS a través de la plataforma Microsoft imagine.

El objetivo fundamental de esta interfaz es permitir al estudiante la edición del modelo Fuzzy para luego transmitirlo a los bloques de datos que almacenan dicha información en el PLC. Una vez se encuentra en funcionamiento el control, la interfaz permite visualizar el comportamiento de las variables medidas y las variables manipuladas en la planta propuesta.

Figura 13. Interfaz gráfica diseñada.



Cabe resaltar que la interfaz desarrollada otorga al estudiante una fácil edición y análisis del modelo, a través de diferentes herramientas gráficas y tablas que contribuyen a una interpretación visual presentando la información de manera ordenada.

Por este motivo, se especifica el funcionamiento de la interfaz en este capítulo, mostrando cada uno de sus módulos descritos en la sección 3.1 y explicando la estructura de la base de datos SQL integrada al proyecto para la administración de los datos del modelo.

4.1 ADMINISTRACION DE LOS DATOS EN LA INTERFAZ GRAFICA

Adjunta a la interfaz se tiene una base de datos en SQL encargada del almacenamiento y la administración de los datos del modelo Fuzzy. En la tabla 2 se relacionan los nombres de las tablas de datos que componen la base de datos, su tipo y su función dentro del programa.

Tabla 2. Listado de tablas que componen la base de datos SQL.

Nombre	Tipo	Función
Dbo.VariableA	Variable de entrada	Almacena la información de la variable de entrada A.
Dbo.VariableB	Variable de entrada	Almacena la información de la variable de entrada B.
Dbo.VariableC	Variable de entrada	Almacena la información de la variable de entrada C.
Dbo.VariableD	Variable de entrada	Almacena la información de la variable de entrada D.
Dbo.ConexionA	Conexión de entrada	Almacena las direcciones del PLC a donde debe transferirse los conjuntos de la entrada A.
Dbo.ConexionB	Conexión de entrada	Almacena las direcciones del PLC a donde debe transferirse los conjuntos de la entrada B.

Dbó.ConexionC	Conexión de entrada	Almacena las direcciones del PLC a donde debe transferirse los conjuntos de la entrada C.
Dbó.ConexionD	Conexión de entrada	Almacena las direcciones del PLC a donde debe transferirse los conjuntos de la entrada D.
Dbó.SalidaX	Variable de salida	Almacena la información de la variable de salida X.
Dbó.SalidaY	Variable de salida	Almacena la información de la variable de salida Y.
Dbó.ConexionX	Conexión de salida	Almacena las direcciones del PLC a donde debe transferirse los conjuntos de la salida X.
Dbó.ConexionY	Conexión de salida	Almacena las direcciones del PLC a donde debe transferirse los conjuntos de la salida Y.
Dbó.Reglas	Tabla de reglas	Almacena la información de la base de reglas del modelo.
Dbó.ConexionReglas	Conexión de reglas	Almacena las direcciones del PLC a donde debe transferirse la base de reglas.
Dbó.NombreEntradas	Nombres	Almacena los nombres de las variables de entrada.
Dbó.NombreSalidas	Nombres	Almacena los nombres de las variables de salida.

4.1.1 Estructura de datos de las variables de entrada.

Las tablas de datos relacionadas como variable de entrada (ver tabla 2), contienen una estructura idéntica que permite almacenar la información de los conjuntos Fuzzy de su respectiva entrada.

La figura 14 muestra la estructura de la tabla de datos para las variables de entrada, esta se compone de 5 campos que almacenan la siguiente información:

- Nombre: (Tipo *string*) Guarda los nombres de los conjuntos.
- Tipo: (Tipo *entero*) Guarda un valor que representa el tipo de conjunto. Puede tomar el valor 1 si es un conjunto de tipo shoulder izquierdo, 2 si es de tipo

triangular o 3 si es un conjunto de tipo shoulder derecho. El valor 0 se asigna únicamente al primer registro que se guarda por defecto.

- Puntolzquierda: (tipo decimal) Guarda el valor del punto izquierdo del conjunto fuzzy.
- PuntoCentro: (tipo decimal) Guarda el valor del punto central del conjunto fuzzy.
- PuntoDerecha: (tipo decimal) Guarda el valor del punto derecho del conjunto Fuzzy.

Figura 14. Estructura de la tabla de las variables de entrada.

	Nombre	Tipo	Puntolzquierda	PuntoCentro	PuntoDerecha
1	No Interviene	0	0.00	0.00	0.00
2	NG	1	-10.00	-6.00	-4.00
3	NM	2	-6.00	-4.00	-2.00
4	NP	2	-4.00	-1.00	0.00
5	CE	2	-1.00	0.00	1.00
6	PP	2	0.00	1.00	4.00
7	PM	2	2.00	4.00	6.00
8	PG	3	4.00	6.00	10.00

La tabla permite guardar 8 registros, de los cuales el primero está guardado por defecto como se muestra en la figura 14 con el objetivo de poder incluir la opción “No interviene” dentro de la base de reglas. En este caso la variable no está involucrada como sujeto dentro de la regla. Los otros registros 2-8 pertenecen a los 7 conjuntos posibles de la variable de entrada.

4.1.2 Estructura de datos de las variables de salida.

Las tablas de datos relacionadas como variable de salida (ver tabla 2), contienen una estructura idéntica que permite almacenar la información de los conjuntos Fuzzy de su respectiva salida.

La figura 15 muestra la estructura de la tabla para las variables de salida, esta se compone de 2 campos que almacenan la siguiente información:

- Nombre: (tipo string) Guarda los nombres de los conjuntos.
- Valor: (tipo decimal) Guarda el valor del conjunto singleton.

Figura 15. Estructura de la tabla de las variables de salida.

	Nombre	Valor
1	No Interviene	0.00
2	NG	-1.00
3	NM	-0.20
4	NP	-0.05
5	CE	0.00
6	PP	0.05
7	PM	0.20
8	PG	1.00

La tabla permite guardar 8 registros, de los cuales el primero está guardado por defecto como se muestra en la figura 15 con el objetivo de poder incluir la opción “No interviene” dentro de la base de reglas. En este caso la variable no está

involucrada como predicado dentro de la regla. Los otros registros 2-8 pertenecen a los 7 conjuntos posibles de la variable de entrada.

4.1.3 Estructura de datos de la tabla de reglas.

La tabla de reglas contiene los registros de todas las reglas formuladas dentro de modelo Fuzzy de control. De acuerdo a la figura 16, la tabla cuenta con 6 campos donde los 4 primeros son reservados para las variables de entrada y los 2 últimos para las variables de salida, permitiendo guardar hasta 100 registros.

También se muestra en la figura 16, que al crear una regla la tabla crea un registro donde almacena campo por campo un valor de 0 a 7 correspondiente al índice del conjunto Fuzzy de la respectiva variable. Si la variable no forma parte de dicha regla almacena el valor 0 (No interviene), de lo contrario almacenará un número entre 1 y 7 de acuerdo al conjunto seleccionado.

Figura 16. Estructura de la tabla de las bases de reglas.

	Error de Angulo	Variación de error de ángulo	Cambio en potencia del motor			
	NG	No Interviene	NG			
	No Interviene	No Interviene	No Interviene			
	NG	NG	NG			
	NM	NP	NM			
	NP	CE	NP			
	CE	PP	CE			
	PP	PG	PP			
	PM	NP	PM			
	PG	NP	PG			

	VariableA	VariableB	VariableC	VariableD	SalidaX	SalidaY
1	1	0	0	0	1	0
2	2	1	0	0	1	0
3	2	3	0	0	2	0
4	2	4	0	0	2	0
5	2	5	0	0	2	0
6	3	1	0	0	1	0
7	3	2	0	0	2	0
8	3	3	0	0	3	0

4.1.4 Estructura de datos de conexión.

Las tablas de datos relacionadas como “conexión de entrada”, “conexión de salida” o “conexión de reglas”, son tablas prediseñadas que guardan las direcciones del PLC donde se debe transmitir la información del modelo. Dado que el número de conjuntos y el número de reglas puede variar, las tablas contienen las direcciones para poder transferir toda la información dentro de los límites, es decir, las 4 variables de entrada cada una con siete (7) conjuntos, las dos (2) variables de salida cada una con siete (7) conjuntos y una base de máximo cien (100) reglas.

La estructura de datos de las tablas de conexión de entrada se compone de cuatro (4) campos y siete (7) registros, estos son utilizados para transferir únicamente el tipo y los valores de cada conjunto. La figura 17 muestra al lado izquierdo la tabla de datos de la variable A y encierra en un recuadro rojo los datos que se envían al PLC, del lado derecho se muestra la estructura de datos de la conexión A, la cual contiene las direcciones para cada uno de estos datos.

Figura 17. Estructura de datos variable A y conexión A.

<i>Tablas de datos del modelo</i>						<i>Tablas de datos</i>				
<i>Variable A</i>						<i>Direcciones de la Variable A</i>				
	Nombre	Tipo	PuntoIzquierda	PuntoCentro	PuntoDerecha	A0	A1	A2	A3	
1	No Interviene	0	0.00	0.00	0.00	DB1.DBW0	DB1.DBW2	DB1.DBW6	DB1.DBW10	
2	NG	1	-10.00	-6.00	-4.00	DB1.DBW14	DB1.DBW16	DB1.DBW20	DB1.DBW24	
3	NM	2	-6.00	-4.00	-2.00	DB1.DBW28	DB1.DBW30	DB1.DBW34	DB1.DBW38	
4	NP	2	-4.00	-1.00	0.00	DB1.DBW42	DB1.DBW44	DB1.DBW48	DB1.DBW52	
5	CE	2	-1.00	0.00	1.00	DB1.DBW56	DB1.DBW58	DB1.DBW62	DB1.DBW66	
6	PP	2	0.00	1.00	4.00	DB1.DBW70	DB1.DBW72	DB1.DBW76	DB1.DBW80	
7	PM	2	2.00	4.00	6.00	DB1.DBW84	DB1.DBW86	DB1.DBW90	DB1.DBW94	
8	PG	3	4.00	6.00	10.00					

Siguiendo la misma línea, las tablas de conexión de salida contienen un (1) campo con siete (7) registros donde almacena las direcciones para enviar los conjuntos Fuzzy dentro del campo “Valor” de las tablas de variable de salida (Ver figura 15). Por su parte la tabla de conexión de reglas se compone de seis (6) campos y cien (100) registros con las direcciones de cada uno de los registros dentro de la tabla de reglas.

4.1.5 Estructura de datos de tablas de nombres.

Estas tablas se componen un campo llamado “Nombre” donde se almacena el nombre de las variables. La figura 18 muestra del lado izquierdo la tabla de datos que almacena los nombres de las entradas (4 registros) y del lado derecho la tabla donde se almacena los nombres de las salidas (2 registros).

Figura 18. Estructura de la tabla de nombres.

<i>Dbn.NombreEntradas</i>		<i>Dbn.NombreSalidas</i>	
	Nombres		Nombres
1	Error de Angulo	1	Cambio en potencia del motor
2	Variación de error de ángulo	2	Salida Y
3	Entrada C		
4	Entrada D		

4.1.6 Flujo de datos en la interfaz.

El proceso de edición de las variables del modelo, la formulación de la base de reglas y la carga de datos al PLC, requieren de una administración adecuada de la información contenida dentro de las tablas de datos. Retomando la figura 4 (ver página 14), se observa que la información del modelo realiza una trayectoria a través de los diferentes módulos de la interfaz.

El proceso comienza cuando se crean los conjuntos de las variables, cuyos nombres se almacenan en las tablas de datos "*Dbó.NombreEntradas*" y "*Dbó.NombreSalidas*", adicional la información de los conjuntos se almacena en las tablas de variables respectivamente.

Cuando se procede a la formulación de reglas, el módulo accede a las tablas de datos y toma los nombres de las variables para ubicarlos en los cabeceros de las tablas, los nombres de sus conjuntos son utilizados en sus reglas y al finalizar, el módulo almacena los índices correspondientes a los conjuntos Fuzzy dentro de las reglas en la tabla de datos de reglas.

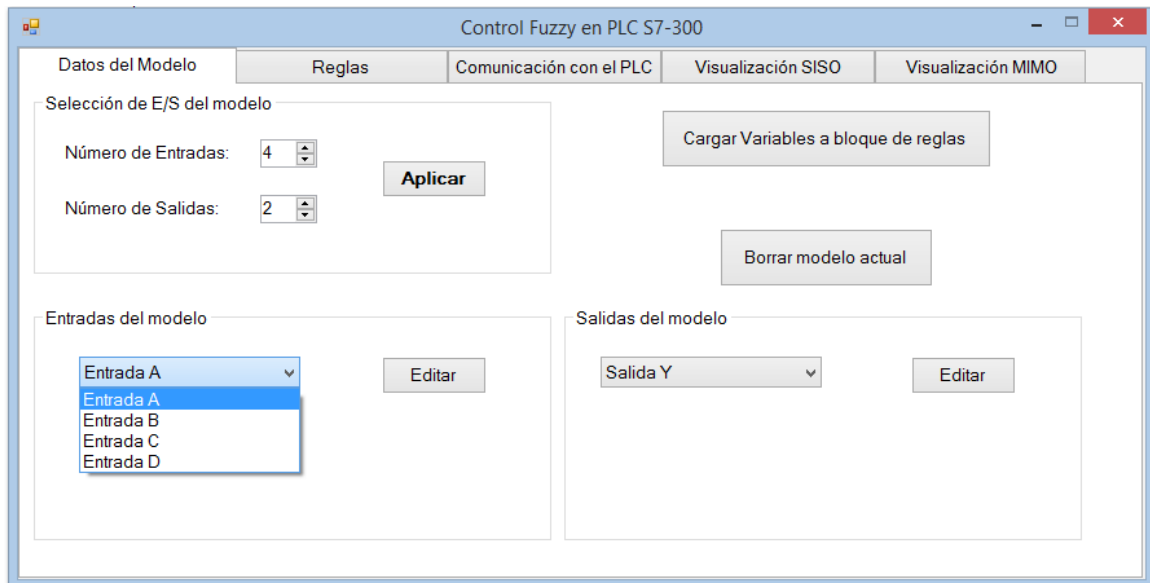
Finalmente, el módulo de comunicación con el PLC toma las tablas de datos de las variables, la tabla de datos de reglas y las tablas de datos de conexión para ejecutar el proceso de transferencia de información al controlador.

4.2 FUNCIONAMIENTO DE LA INTERFAZ

4.2.1 Edición del modelo Fuzzy.

La edición del modelo Fuzzy se hace de manera secuencial a través de los dos primeros módulos; inicialmente se deben generar los conjuntos de las variables de entrada y los conjuntos de las variables de salida para proceder a la formulación de la base de reglas. La figura 19, muestra una captura del módulo de entradas y salidas.

Figura 19. Edición de variables de entrada y salida en la interfaz.



El panel llamado “*Selección de E/S del modelo*” permite seleccionar un máximo de cuatro (4) variables de entrada y dos (2) variables de salida; al pulsar el botón “*Aplicar*”, el módulo habilita los selectores dispuestos tanto en el panel de entradas como en el de salidas con la cantidad de variables fijadas.

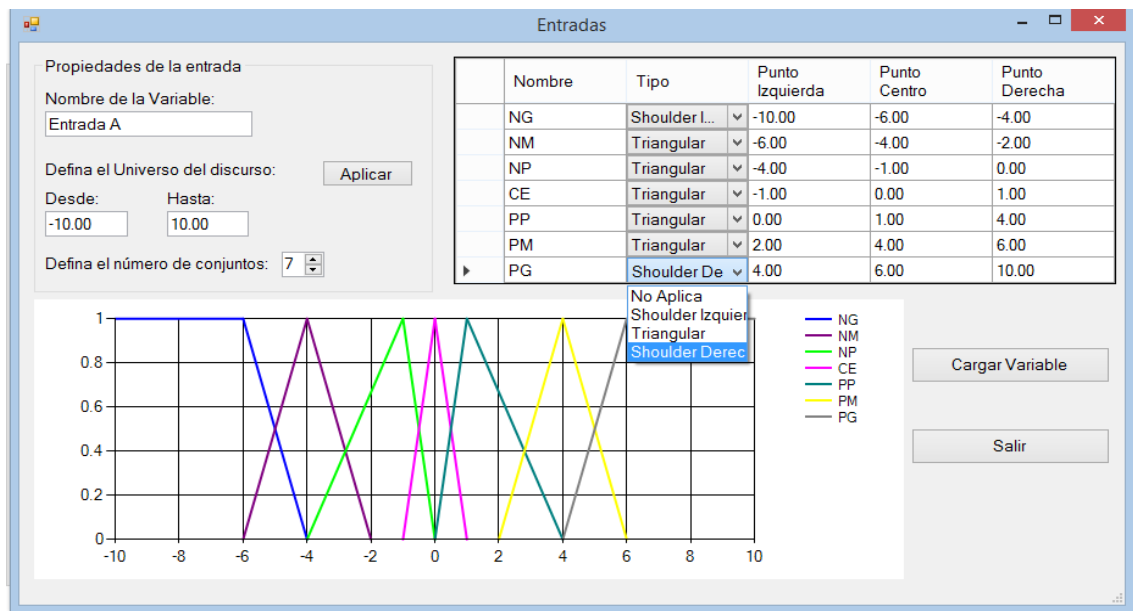
Al elegir una variable del selector, el botón “*Editar*” envía al usuario a una nueva ventana donde se edita los conjuntos de esta.

Una vez se han creado los conjuntos de las variables, el botón “*Cargar variables a bloque de reglas*” permite pasar a la formulación de la base de reglas.

4.2.1.1 Edición de conjuntos de las variables de entrada.

Esta ventana está diseñada para la edición de los conjuntos Fuzzy de las variables de entrada. En la figura 20, se muestran tres (3) zonas de trabajo: Un panel de propiedades de la entrada, una tabla para ingresar los conjuntos Fuzzy y una gráfica para representar los conjuntos.

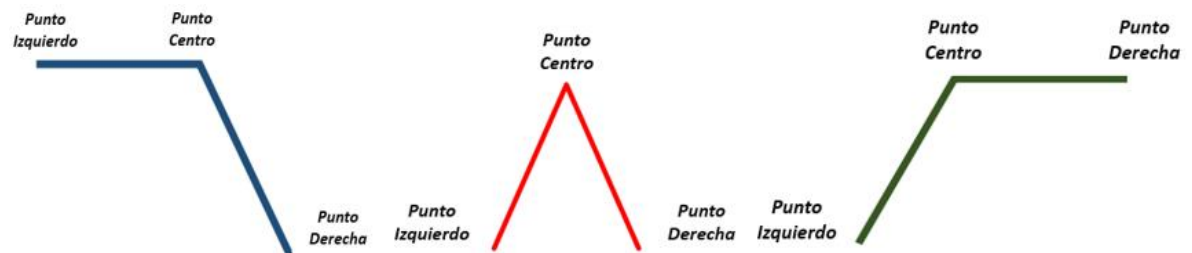
Figura 20. Modulo creación conjuntos de variables de entrada.



En el panel “*Propiedades de la entrada*” el usuario puede asignarle un nombre a la variable y definir tanto el universo del discurso como el número de conjuntos (máximo siete (7) conjuntos). Al pulsar el botón “*Aplicar*”, se despliega la tabla donde se ingresan los conjuntos y se activa el control de la gráfica.

La tabla habilita el número de filas indicado por el número de conjuntos. Dentro de esta, el usuario ingresa el nombre de cada conjunto, elige mediante un selector entre tres (3) tipos de conjuntos: Shoulder izquierdo, triangular o shoulder derecho, definiendo los puntos de cada conjunto como se muestra en la figura 21.

Figura 21. Asignación de tipo de conjunto.



A medida que se crean o se modifican los conjuntos, el control de la gráfica muestra una representación de éstos dentro del universo del discurso.

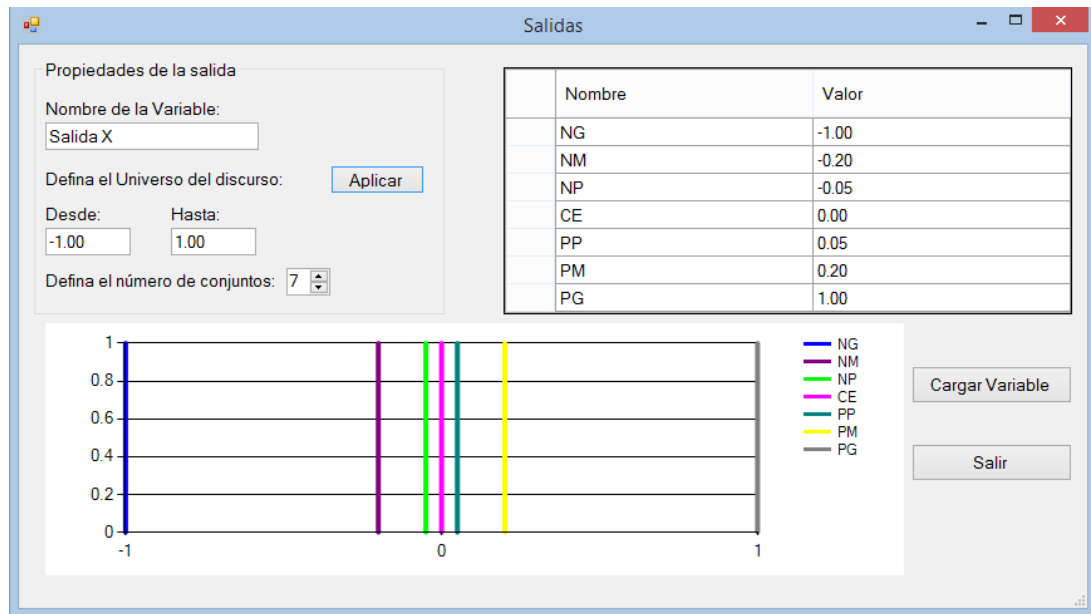
Al finalizar la edición de los conjuntos, el botón “*Cargar variable*” almacena la información de los conjuntos dentro de la respectiva tabla dentro de la base de datos.

4.2.1.2 Edición de conjuntos de las variables de salida.

De manera similar a la edición de los conjuntos de entrada, la ventana mostrada en la figura 22, permite la edición de los conjuntos de la variable de salida.

A diferencia de los conjuntos de entrada, los conjuntos de la variable de salida están predefinidos de tipo *singleton*, permitiéndonos en este módulo modificar y ajustar su valor de asignación para cumplir con la necesidad del control Fuzzy a implementar.

Figura 22. Modulo creación conjuntos de variables de salida.



4.2.2 Formulación de la base de reglas del modelo.

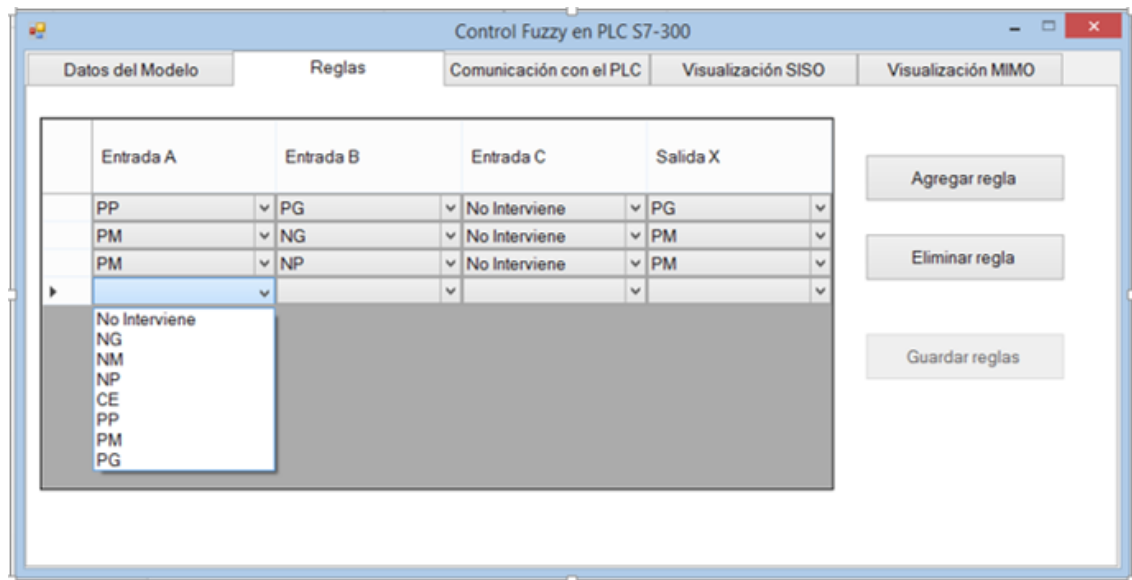
En la figura 23, se muestra el segundo módulo de la interfaz el cual se activa luego de haber creado los conjuntos Fuzzy de las variables y se envían a través del botón

“Cargar variables a bloque de reglas”. La función de este módulo consiste en la creación de la base de reglas.

Este módulo contiene una tabla la cual muestra una columna por cada variable de entrada/salida especificada anteriormente y habilita el control para agregar reglas, eliminar reglas y guardar la base de reglas.

Cuando se agrega una regla, la fila habilita un selector en cada columna que contiene los nombres de los conjuntos de dicha variable y la opción “no interviene”, para que el usuario defina el conjunto que hace parte de la regla o si por el contrario no la afecta.

Figura 23. Modulo creación base de reglas.



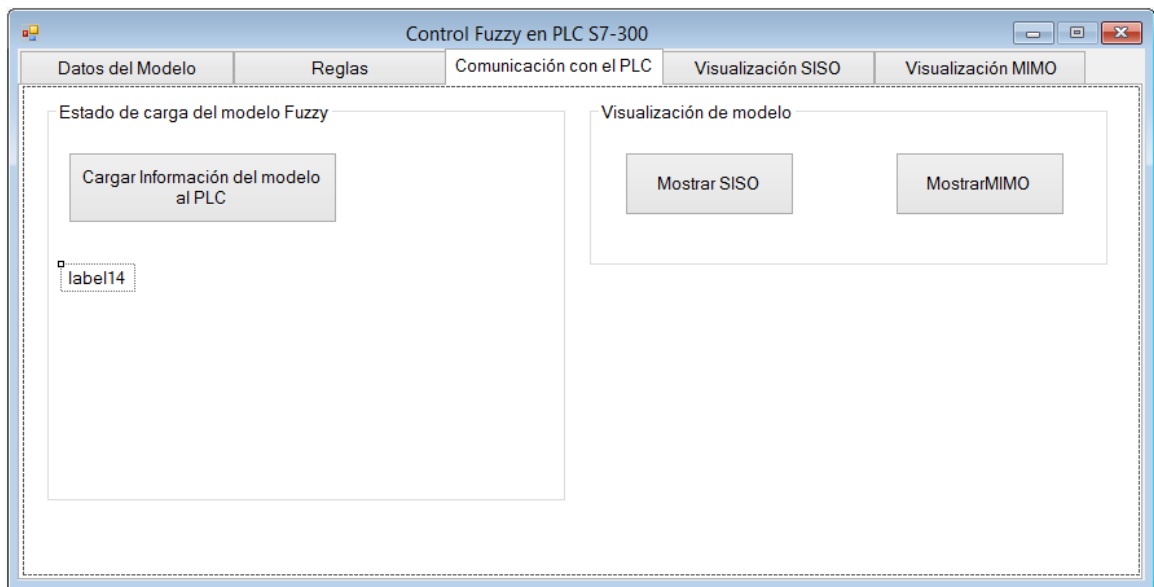
Una vez se han añadido todas las reglas, el botón “Guardar reglas” Ejecuta la función para almacenar la base de reglas dentro de una tabla de la base de datos SQL.

4.2.3 Transferencia del modelo Fuzzy al PLC

Una vez ha finalizado la edición del modelo Fuzzy, el módulo de “*Comunicación con el PLC*” se encarga del proceso de transferencia a los bloques de datos del PLC a través del OPC.

Dentro de éste módulo se encuentran dos (2) paneles de control. El panel “*Estado de carga del modelo Fuzzy*” contiene el botón que ejecuta la función de carga del modelo al PLC y lleva un seguimiento del estado de carga. Si la carga se realiza con éxito, el panel “*Visualización de modelo*” se activa para definir cuál de las dos ventanas de visualización se quiere mostrar, bien sea el control de posición angular “*Mostrar SISO*” o el control de velocidad y temperatura del aire “*Mostrar MIMO*”.

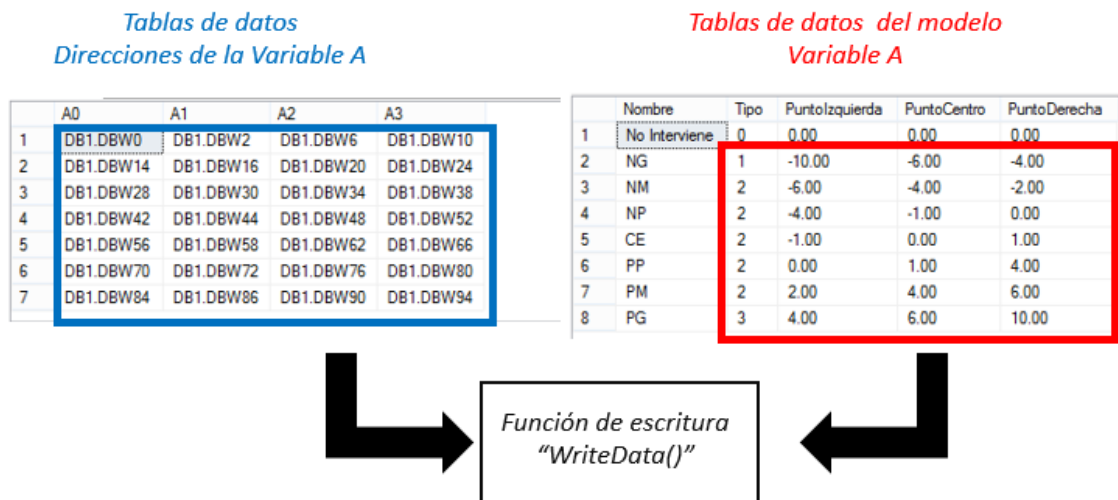
Figura 24. Carga del modelo Fuzzy al PLC.



El proceso de transferencia de la información se da de la siguiente manera:

1. Borra la información existente dentro de los bloques de datos del PLC.
2. Llama a las tablas de datos de cada variable de entrada/salida, la base de reglas y a las tablas de conexiones que contiene las direcciones del PLC donde debe almacenar dicha información. Como se muestra en la figura 25, la función toma las direcciones de la tabla señaladas en el recuadro azul y la usa para transmitir la información del modelo encerrado dentro del recuadro rojo.
3. Ejecuta la función “*WriteData()*” para la escritura en el PLC.

Figura 25. Esquema de comparación direcciones – datos para la transferencia de información.



4.2.4 Visualización de las señales de la planta.

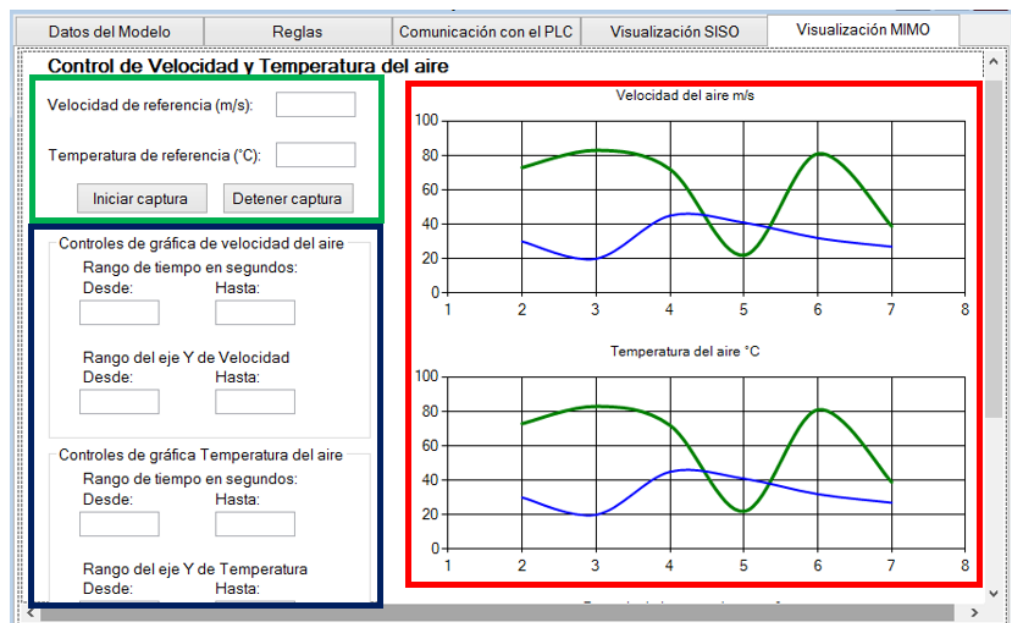
Esta función se realiza luego de haber cargado la información del modelo Fuzzy en la memoria del PLC y de haber definido la configuración del modelo (SISO o MIMO),

Como se observa en la figura 26, el módulo cuenta con tres áreas de trabajo: El área de gráficas (encerrada en rojo) que registra el comportamiento de las señales, un área (azul) donde se ubican los controles de las gráficas y un área (verde) donde se establecen los valores de referencia de las variables controladas y se da la orden de inicio del modelo en el PLC.

El botón “*Iniciar captura*” activa una marca en el PLC para ejecutar el control Fuzzy e inicia un cronómetro que permite registrar el tiempo transcurrido desde dicha orden.

Un temporizador de intervalos da la orden de tomar los valores leídos del PLC de las señales y el tiempo transcurrido para graficarlos en sus respectivas gráficas.

Figura 26. Módulo de Visualización.

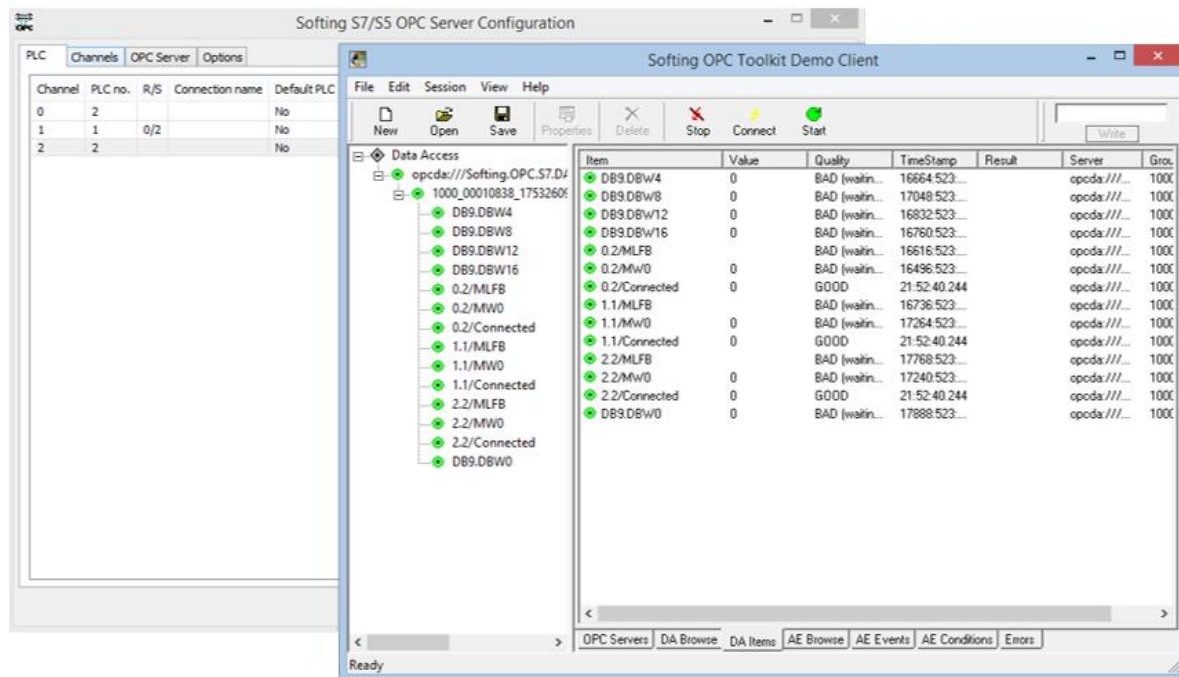


Por su parte, el botón “*Detener captura*” revierte la marca en el PLC que activa el control Fuzzy y detiene el registro de las mediciones, preparando la interfaz para la siguiente toma.

5. IMPLEMENTACION DE LA COMUNICACIÓN OPC

La interfaz gráfica y el PLC se comunican entre sí por medio de un servidor OPC de acceso a datos (Ver figura 27), este servidor funciona como una “*ventana de datos*” que le permite a la interfaz acceder a los bloques de datos y a las variables del PLC, para leerlas y modificar su información.

Figura 27. Servidor OPC softing.



Para que la comunicación interfaz – PLC sea posible debe configurarse una serie de protocolos de comunicación entre la interfaz gráfica y el servidor, además, debe configurarse el canal de comunicación entre el servidor OPC y el PLC.

5.1 CONFIGURACION DE LOS ELEMENTOS DEL SERVIDOR OPC

El servidor está compuesto por tres (3) objetos claves: Un servidor que mantiene la información y unifica los datos dentro de uno o varios grupos, Un grupo que dota al servidor de una estructura para almacenar los elementos de datos y una serie de elementos que corresponden a los datos a los que se tiene acceso. Estos últimos tienen la etiqueta o dirección del PLC, su valor y el tipo de dato nativo.

En la figura 28, se observa la disposición de dos zonas de trabajo en el servidor. La primera que corresponde al árbol de servidor y contiene la URL del servidor, el grupo y los elementos de datos, y una segunda zona que corresponde a la tabla de elementos disponibles en el servidor.

Figura 28. Zonas de trabajo servidor OPC.

The screenshot shows the OPC Data Access interface. On the left, a tree view displays the server structure. A red box highlights the tree view, which is labeled "Árbol del servidor". Arrows point to the "url servidor" (opcda://Softing.OPC.S7.DA), "Grupo" (1000_00010838_1753260), and "Elementos" (DB9.DBW4, DB9.DBW8, DB9.DBW12, DB9.DBW16, DB9.DBW0, 0.2/State) in the tree view. On the right, a table displays the data for the selected elements. A blue box highlights the table, which is labeled "Tabla de elementos".

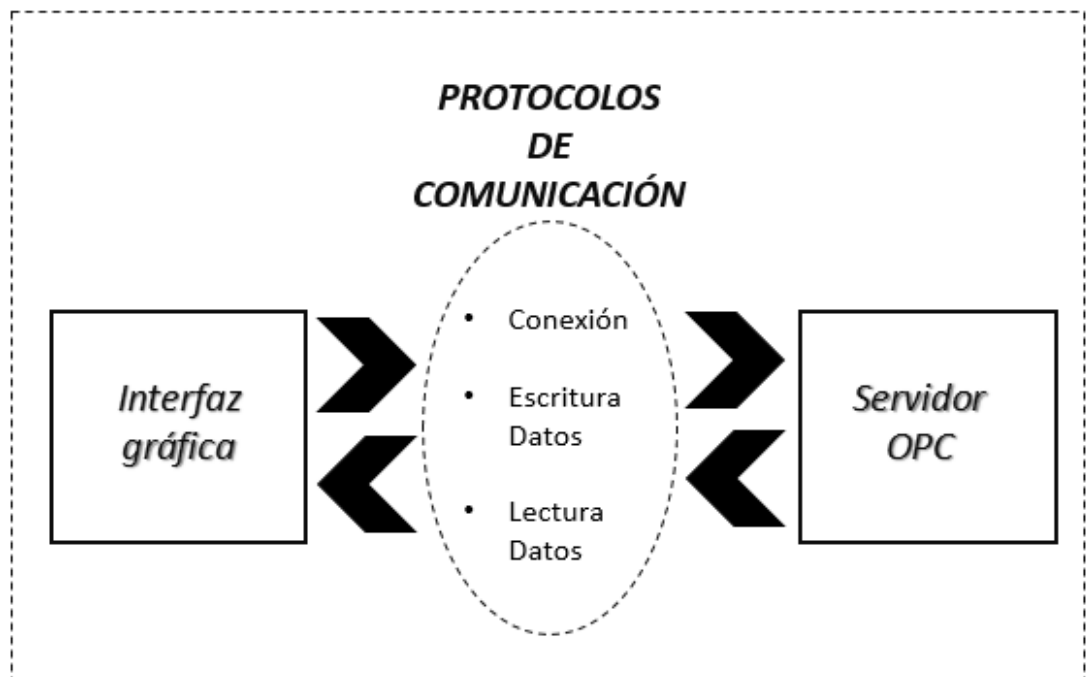
Item	Value	Quality	TimeStamp	Result	Server	Grou
DB9.DBW4	0	BAD (waitin...	6080:541:5...		opcda://...	1000
DB9.DBW8	0	BAD (waitin...	6320:541:5...		opcda://...	1000
DB9.DBW12	0	BAD (waitin...	6128:541:5...		opcda://...	1000
DB9.DBW16	0	BAD (waitin...	5984:541:5...		opcda://...	1000
DB9.DBW0	0	BAD (waitin...	6080:541:5...		opcda://...	1000
0.2/State		BAD (waitin...	5504:541:5...		opcda://...	1000

5.2 COMUNICACIÓN ENTRE LA INTERFAZ GRAFICA Y EL SERVIDOR OPC¹¹

La comunicación entre la interfaz gráfica y el servidor OPC se da vía software a través de unos protocolos de comunicación escritos como parte del código fuente en la interfaz. Dicho código requiere que se integre a la interfaz las referencias “*OpcNetApi*”, “*OpcNetApi.Com*” y “*OpcNetApi.Xml*”, las cuales contienen las clases y funciones necesarias para la comunicación.

Este código se compone de tres partes como se observa en la figura 29.

Figura 29. Esquema general de conformación de protocolos de comunicación.



¹¹ Opc client with C#: an how to video [en línea]. Disponible en internet: <URL: <https://www.mesta-automation.com/opc-client-with-c-an-how-to-video/>> Fecha de consulta: 15/07/2017.

La conexión consiste en configurar el *handshaking* (URL del servidor, suscripción, tasa de transferencia, apertura de la conexión), determinar los elementos del servidor que deben leerse, activar el evento de lectura y la función de escritura de datos.

Por su parte, la escritura de datos "*WriteData()*" consiste en una función que se llama cuando se transfiere la información del modelo al PLC en el módulo de comunicación de la interfaz y cuando ocurre un cambio en los valores de referencia de los controles en el módulo de visualización. La función requiere que se ingrese como parámetro la dirección de memoria del PLC que recibe la información y el valor representado como un entero de tipo UI2 (entero sin signo de 32 bits) bajo el estándar IEEE 754¹².

El evento de lectura ocurre cuando alguno de los valores dentro de la tabla de elementos del servidor cambia y es detectado por la función de lectura integrada en la conexión. El evento toma el valor en su representación UI2 bajo el estándar IEEE754⁷ y lo convierte a su valor real.

¹² Estándar IEEE754 [en línea]. Disponible en internet: <URL: http://www.carlospes.com/curso_representacion_datos/06_01_estandar_ieee_754.php>, fecha de consulta 18/04/2018.

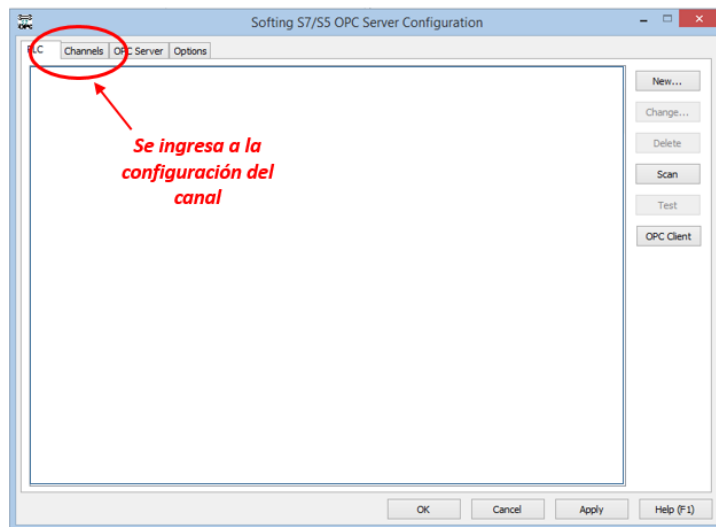
5.3 COMUNICACIÓN ENTRE EL SERVIDOR OPC Y EL PLC

La comunicación entre el servidor de datos OPC y el PLC se da vía hardware a través de una conexión MPI.

Para que el servidor pueda reconocer dicha conexión, se debe configurar el canal de comunicación en la herramienta externa para configuración del servidor de la siguiente manera:

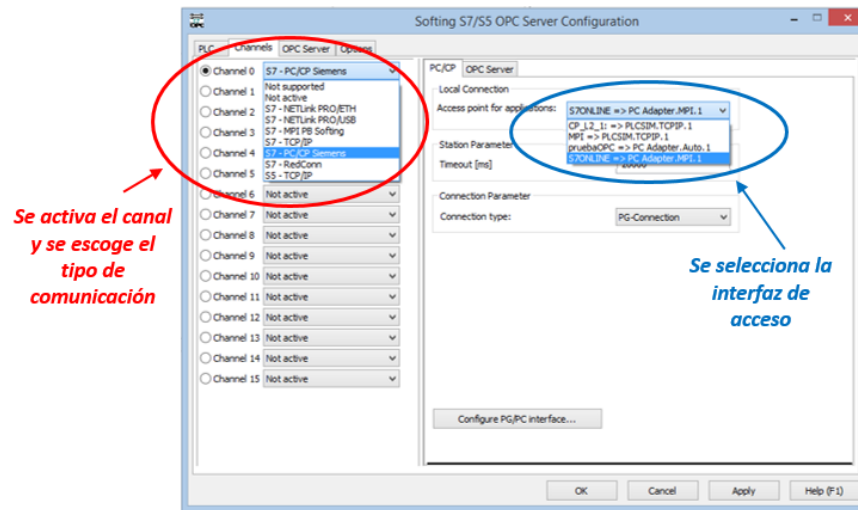
1. Se accede al “Configurator” para configurar el canal. (Ver figura 30)

Figura 30. Configuración del canal de comunicación.



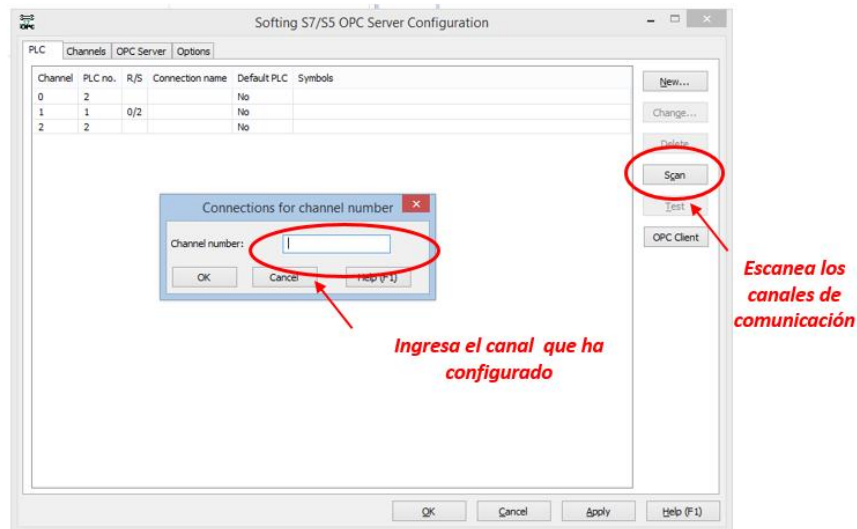
2. Se activa el canal de comunicación a utilizar en la opción “S7 –PC/CP Siemens” y se configura el punto de acceso “S7ONLINE => Pc Adapter.MPI.1”, después “Aplicar”. (Ver figura 31)

Figura 31. Activación del canal de comunicación y punto de acceso.



3. En la pestaña “PLC” pulsa la opción “Scan” e ingresa el número del canal donde ha configurado la comunicación. Finalmente “Aplica”. (Ver figura 32)

Figura 32. Verificación del canal de comunicación.



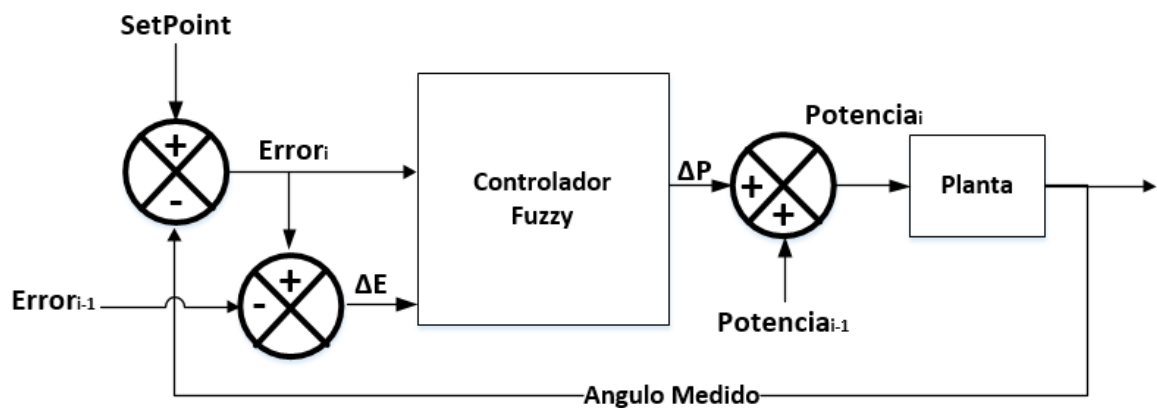
4. Pulsa la opción “OPC Client” para que el asistente de configuración abra el servidor y configure la URL, y el grupo para ingresar los elementos de datos.

6. DESARROLLO DEL COMPONENTE DE CONTROL EN EL PLC

El bloque de control *FuzzyControl* se ha desarrollado como una herramienta parametrizable con el objetivo de poder usarse en diferentes sistemas de control donde quiera implementarse esta técnica de control. Por ello, se integra este bloque a un algoritmo de control para implementarse en el banco “Ducto de viento para recrear eventos de automatismos”.

Este banco permite implementar control Fuzzy en una configuración SISO para el control de la posición angular de una placa al extremo del ducto. La figura 33 muestra el lazo de control donde se mide ángulo a través de un potenciómetro y se realiza el control toma como variables de entrada el error en el ángulo y el cambio en el error del ángulo. Como salida del controlador se tiene un cambio en la velocidad del ventilador, la cual se suma al valor en el ciclo de ejecución anterior, se almacena y se envía a la salida del PLC.

Figura 33. Lazo de control del sistema SISO.



De manera similar, se aplica un control al banco en su configuración MIMO donde el control se efectúa a la velocidad y la temperatura del flujo de aire, teniendo como variables de entrada el error en la velocidad, la variación del error de velocidad, el error de temperatura y la variación del error de temperatura. Como variables de salida del bloque de control se tiene el cambio en la potencia de la resistencia y el cambio a la velocidad del ventilador que se suman a su valor total en el ciclo anterior para entregar la salida por el canal análogo a la planta. Sin embargo, no resulta conveniente evaluar un control de velocidad de flujo y temperatura bajo un mismo intervalo de ejecución debido a que el control de temperatura tiene una respuesta significativamente más lenta, por ello se optó por evaluar el control de temperatura cada veinte (20) ciclos de ejecución.

En la sección 2.3. Se describe la funcionalidad de los principales bloques de datos y bloques de funciones que componen el algoritmo de control. Con intención de explicar en profundidad su funcionamiento, se tratará a lo largo de este capítulo la estructura de datos de los bloques de datos que reciben la información del modelo Fuzzy proveniente de la interfaz y el funcionamiento de las tres etapas de ejecución del algoritmo mencionadas en dicha sección, dando atención especial al funcionamiento de *FuzzyControl* mostradas.

Se relacionan en la siguiente tabla las funciones, los bloques de funciones y bloques de datos de instancia que componen la programación que explicarán a través de este capítulo. Los bloques de datos globales que almacenan la información del modelo Fuzzy y los valores de visualización se han mencionado en la tabla 1 (ver página 21).

Tabla 3. Bloque de programación en el PLC.

	Nombre	Etapa
OB35	CYC_INT5	
FB10	SetupVariables	Preparación de variables de entrada al modelo Fuzzy.
FB15	Normalización	Preparación de variables de entrada al modelo Fuzzy.
FB20	SetupReferencias	Preparación de variables de entrada al modelo Fuzzy.
FB25	MediaMovil	Preparación de variables de entrada al modelo Fuzzy.
FB30	CapturaSeñales	Preparación de variables de entrada al modelo Fuzzy.
FB35	Salida	Preparación de las variables de salida a la planta.
FB50	FuzzyControl	Ejecución de FuzzyControl.
FB51	B01Pertenencia	Ejecución de FuzzyControl.
FB52	B02Implicación	Ejecución de FuzzyControl.
FB53	B03Defuzzificación	Ejecución de FuzzyControl.
FC1	MVP_IZQ	Ejecución de FuzzyControl.
FC2	MVP_DER	Ejecución de FuzzyControl.
DB10	SetupVariables_DB.	Bloque de instancia asociado a FB10.
DB15	Normalización_DB.	Bloque de instancia asociado a FB15.
DB20	SetupReferencias_DB.	Bloque de instancia asociado a FB20.
DB25	MediaMovil_DB.	Bloque de instancia asociado a FB25.
DB30	CapturaSeñales_DB.	Bloque de instancia asociado a FB30.
DB35	Salida_DB.	Bloque de instancia asociado a FB35.
DB50	FuzzyControl_DB.	Bloque de instancia asociado a FB50.
DB51	B01Pertenencia_DB.	Bloque de instancia asociado a FB51.
DB52	B02Implicación_DB.	Bloque de instancia asociado a FB52.
DB53	B03Defuzzificación_DB.	Bloque de instancia asociado a FB53.

6.1 ESTRUCTURA DE LOS BLOQUES DE DATOS GLOBALES

Los bloques de datos globales mencionados en la tabla 1 (Ver página 21), almacenan la información relacionada al modelo Fuzzy (conjuntos de entrada, conjuntos de salida, base de reglas, valores de referencia del control) y almacenan las mediciones que se deben enviar de regreso a la interfaz para visualizarlas.

Estas bases de datos obedecen a la siguiente estructura:

6.1.1 Estructura de datos de las variables de entrada.

Los bloques de datos *DB1.EntradaA*, *DB2.EntradaB*, *DB3.EntradaC*, *DB4.EntradaD*, almacenan los conjuntos de cada una de las variables de entrada, estos bloques de datos permiten almacenar un máximo de siete (7) conjuntos. La figura 34 muestra la estructura de datos que conforma cada uno de estos bloques.

Figura 34. Estructura de bloques de datos de las variables de entrada.

EntradaB			
	Nombre	Tipo de datos	Offset
1	Static		
2	B	Array[1..7] of Struct	0.0
3	B[1]	Struct	0.0
4	B0	Int	0.0
5	B1	Real	2.0
6	B2	Real	6.0
7	B3	Real	10.0
8	B[2]	Struct	14.0
9	B0	Int	0.0
10	B1	Real	2.0
11	B2	Real	6.0
12	B3	Real	10.0
13	B[3]	Struct	28.0
14	B[4]	Struct	42.0
15	B[5]	Struct	56.0
16	B[6]	Struct	70.0
17	B[7]	Struct	84.0

Dicha base contiene una variable de tipo *Array[] of Struct*, la cual permite almacenar dentro de cada elemento del vector un estructura con diferentes tipos de datos.

Se observa que la variable de entrada B declarada como un vector de siete (7) elementos, donde en cada elemento se almacenan los siguientes tres (3) datos:

- El elemento B0 de tipo entero que guarda el valor correspondiente al tipo de conjunto Fuzzy de entrada mostrado en la sección 3.1.1.
- Los elementos B1, B2, B3 de tipo real almacenan los valores izquierdos, centro y derechos del conjunto Fuzzy mostrados en la figura 15.

La tercera columna del bloque de datos muestra la dirección que dicho dato ocupa en la memoria, entendiéndose de la siguiente manera:

- “El elemento B[1].B1 del bloque de datos DB2 tiene una posición de memoria dada por la dirección *DB2.DBW2.*”
- “El elemento B[2].B1 tiene una posición de memoria dada por la dirección *DB2.DBW16.*”

Estas direcciones son la información que requiere la interfaz en sus bases de datos de conexión para el proceso de transferencia de datos.

6.1.2 Estructura de datos de las variables de salida.

Los bloques de datos *DB5.SalidaX* y *DB6.SalidaY*, almacenan los conjuntos de cada una de las variables de salida. Estos bloques de datos permiten almacenar un máximo de siete (7) conjuntos. La figura 35 muestra la estructura de datos que conforma cada uno de estos bloques.

Figura 35. Estructura de bloques de datos de las variables de salida.

SalidaX				
		Nombre	Tipo de datos	Offset
1	Static	Static		
2	X	X	Array[1..7] of Real	0.0
3	X[1]	X[1]	Real	0.0
4	X[2]	X[2]	Real	4.0
5	X[3]	X[3]	Real	8.0
6	X[4]	X[4]	Real	12.0
7	X[5]	X[5]	Real	16.0
8	X[6]	X[6]	Real	20.0
9	X[7]	X[7]	Real	24.0

Dado que estos conjuntos son de tipo *singleton*, solo requieren el valor década uno de ellos, por tanto, se declara como un vector de reales (*array[] of Real*).

La columna “*offset*” muestra la dirección que dicho dato ocupa en la memoria de manera similar a las variables de entrada. Ejemplo:

“La variable X[2] del bloque de datos DB5 tiene una posición de memoria dada por la dirección *DB5.DBW4*.”

6.1.3 Estructura de datos de la base de reglas.

El bloque de datos *D75.Base de Reglas*, almacena la base de reglas del modelo Fuzzy. Este bloque de datos permite almacenar un máximo de cien (100) reglas. La figura 36 muestra la estructura de datos que conforma este bloque.

Figura 36. Estructura de bloques de datos de la matriz de reglas.

base de reglas			
	Nombre	Tipo de datos	Offset
1	Static		
2	reglas	Array[1..100, 1..6] of Int	0.0
3	reglas[1,1]	Int	0.0
4	reglas[1,2]	Int	2.0
5	reglas[1,3]	Int	4.0
6	reglas[1,4]	Int	6.0
7	reglas[1,5]	Int	8.0
8	reglas[1,6]	Int	10.0
9	reglas[2,1]	Int	12.0
10	reglas[2,2]	Int	14.0
11	reglas[2,3]	Int	16.0
12	reglas[2,4]	Int	18.0
13	reglas[2,5]	Int	20.0
14	reglas[2,6]	Int	22.0
15	reglas[3,1]	Int	24.0
16	reglas[3,2]	Int	26.0

La variable “Reglas” dentro de esta base de datos se declara como un matriz de cien (100) filas con seis (6) datos de tipo entero por columna, que corresponde a una estructura idéntica a la mencionada en la sección 3.1.3, donde los primeros cuatro (4) campos están reservados para los conjuntos de cada una de las variables de entrada y los dos (2) últimos son para las variables de salida.

6.1.4 Bloque de datos de parámetros del modelo.

El bloque “*DB8.Parámetros*” almacena los valores de número de entradas, número de salidas, y los valores de referencia del control requeridos como parámetros del bloque Fuzzy, provenientes de la interfaz.

La figura 37 muestra la estructura de dicho bloque.

Figura 37. Estructura de bloques de datos de los parámetros.

Parámetros			
	Nombre	Tipo de datos	Offset
1	Static		
2	Nentradas	Int	0.0
3	Nsalidas	Int	2.0
4	Nreglas	Int	4.0
5	SetPointAngulo	Real	6.0
6	SetPointVelocidad	Real	10.0
7	SetPointTemperatura	Real	14.0

6.1.5 Bloque de datos de visualización.

El bloque “*DB9.Visualización*” almacena el valor real de las señales de la planta (ángulo medido, velocidad del aire, temperatura, potencia entregada al motor y a la resistencia) para entregarlas a la interfaz vía OPC. Este bloque es puramente del almacenamiento y no proporciona ninguna función en el algoritmo de control.

Está configurado bajo la siguiente estructura:

Figura 38. Estructura de bloques de datos de los valores de visualización.

VISUALLIZACION			
	Nombre	Tipo de datos	Offset
1	Static		
2	EntradaPotenciómetro	Real	0.0
3	EntradaVelocidad	Real	4.0
4	EntradaTemperatura	Real	8.0
5	SalidaMotor	Real	12.0
6	SalidaResistencia	Real	16.0

6.1.6 Tabla de variables del PLC

La tabla de variables del PLC comprende aquellas direcciones del PLC disponibles para el control, definidas como entradas, salidas y marcas. Cada una de estas cumple una función específica dentro del algoritmo del control. En la tabla se relaciona la tabla de variables del PLC.

Tabla 4. Tabla de variables del PLC.

Nombre	Clasificación	Dirección	Descripción
Potenciómetro	Entrada	IW752	Variable de tipo WORD, Entrada analógica que recibe la señal del potenciómetro para la medición del ángulo.
Pitot	Entrada	IW754	Variable de tipo WORD, Entrada analógica que recibe la señal de la sonda pitot para la medición de la velocidad de aire.
Pt100	Entrada	IW760	Variable de tipo WORD, Entrada analógica que recibe la señal de la termorresistencia para la medición de la temperatura de aire.

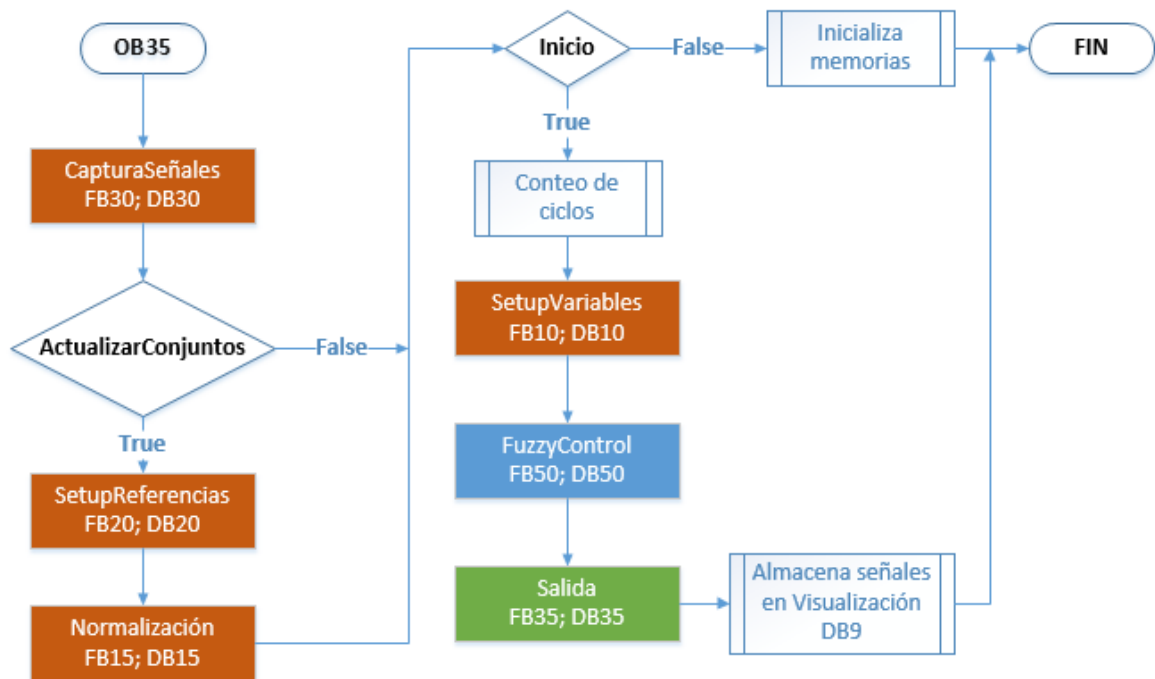
Inicio	Entrada	M0.0	Marca booleana que se activa desde la interfaz para la ejecución del control.
IndicadorSISO	Entrada	M1.0	Marca booleana que se activa desde la interfaz e indica que la planta está configurada en modo SISO.
ActualizarConjuntos	Entrada	M2.0	Marca booleana que se activa desde la interfaz para ejecutar procesos de normalización por cambios en los conjuntos o valores de referencia.
Motor	Salida	QW752	Variable de tipo WORD. Salida analógica que envía un porcentaje de la potencia máxima al motor.
Resistencia	Salida	QW754	Variable de tipo WORD. Salida analógica que envía un porcentaje de la potencia máxima a la resistencia calefactora.
Memoria1	Memoria	MW2	Variable de tipo entero. Almacena el valor de error de ángulo o velocidad para utilizarla en el siguiente ciclo.
Memoria2	Memoria	MW4	Variable de tipo entero. Almacena el valor de error de temperatura para utilizarla en el siguiente ciclo.
Señal1ant	Memoria	MW6	Variable de tipo entero. Almacena el valor de la salida al motor para utilizarlo en el siguiente ciclo.
Señal2ant	memoria	MW8	Variable de tipo entero. Almacena el valor de la salida a la resistencia para utilizarlo en el siguiente ciclo.

6.2 BLOQUE DE EJECUCION POR INTERRUPCION OB35

Este bloque está desarrollado en lenguaje SCL y realiza las llamadas a cada uno de los bloques de funciones, El bloque se ejecuta cada 50 milisegundos. La figura 39 se muestra el diagrama de flujo de la ejecución de todo el OB, identificando las tres etapas del algoritmo:

- Preparación de variables de entrada al modelo Fuzzy, señalado en color marrón.
- Ejecución de *FuzzyControl*, señalado en color azul.
- Preparación de las variables de salida a la planta, señalado en color verde.

Figura 39. Flujo de ejecución periódica OB35.



6.3 PREPARACIÓN DE VARIABLES DE ENTRADA AL MODELO FUZZY

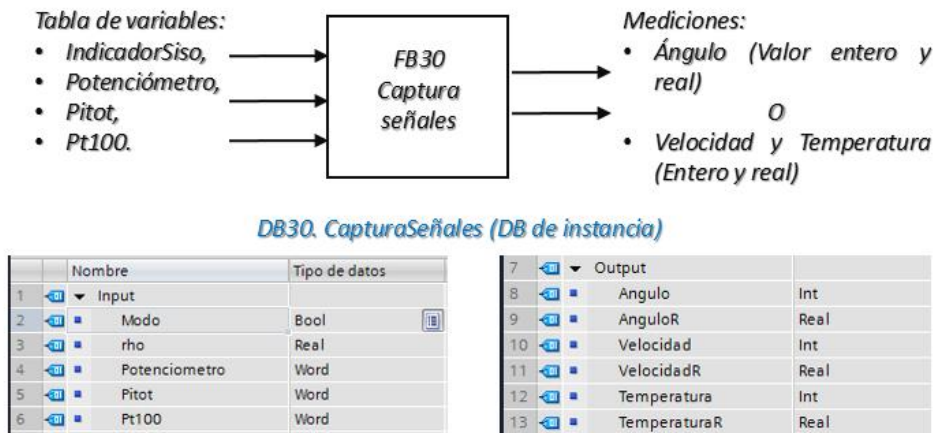
Retomando la figura 39, se observa la secuencia en que el bloque de ejecución llama a las funciones encargadas del proceso de captura de señales de la planta, valores de referencia del control, normalización de las variables Fuzzy y cálculo de las variables de entrada (error y variación del error).

6.3.1 FB30. CapturaSeñales.

Esta función toma las señales de medición de ángulo del potenciómetro, la presión diferencial del tubo pitot o la temperatura medida por la termorresistencia, y de acuerdo a la configuración de la planta dada por el “IndicadorSISO”, entrega su conversión a un valor digital del ángulo o el valor de velocidad y temperatura capturadas por el canal analógico.

La figura 40 muestra de manera esquemática las entradas y salidas de la función, además la estructura de datos que conforma su bloque de datos de instancia.

Figura 40. Esquema general de flujo en los bloques de instancia de la captura de señales.

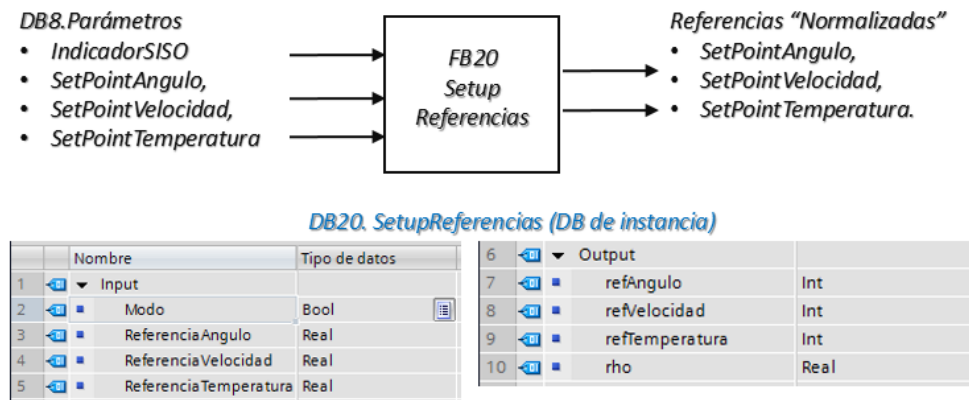


6.3.2 FB20. SetupReferencias:

Esta función se ejecuta cuando la marca “*ActualizarConjuntos*” se encuentra activada. De acuerdo a la configuración dada por “*IndicadorSISO*”, toma los valores de referencia del control provenientes del “*DB8*” y aplica un proceso de amplificación similar al realizado en la captura de señales para que éstos y las mediciones sean comparables entre sí.

La figura 41 muestra de manera esquemática las entradas y salidas de la función y la estructura de datos de su bloque de datos de instancia.

Figura 41. Esquema general de flujo en los bloques de instancia de la actualización de referencias.

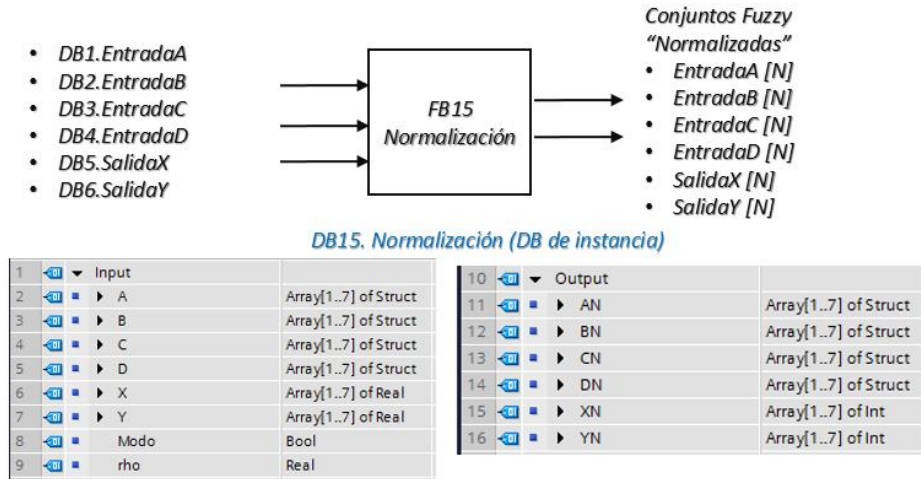


6.3.3 FB15. Normalización:

Esta función se ejecuta cuando la marca “*ActualizarConjuntos*” se encuentra activada. De acuerdo a la configuración dada por “*IndicadorSISO*” toma las estructuras de datos de los conjuntos Fuzzy (DB1 al DB6) y realiza un proceso de amplificación de cada uno de los puntos de los conjuntos Fuzzy tal que puedan ser comparados con los valores de referencia y los valores medidos.

La figura 42 muestra de manera esquemática las entradas y salidas de la función y la estructura de datos de su bloque de datos de instancia.

Figura 42. Esquema general de flujo en los bloques de instancia de la normalización.



La estructura de datos entre la variable de entrada A antes y después de ser normalizada ($A \Rightarrow AN$) conserva el mismo tamaño, sin embargo, los tipos de datos cambian pues pasan de ser reales a enteros.

Figura 43. Tablas de datos de los valores normalizados.

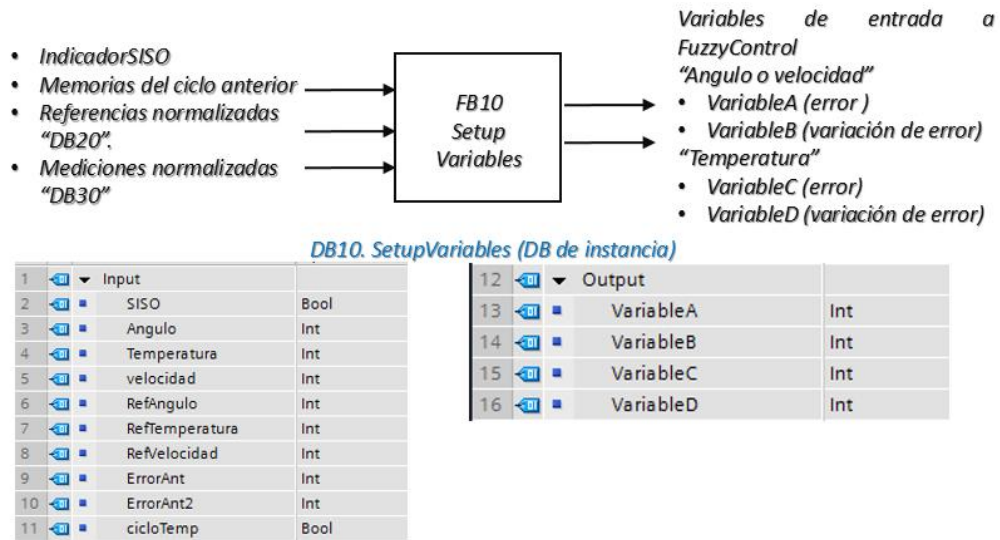
Variable A sin normalizar		Variable A normalizada (AN)	
1	Input	21	Output
2	A	22	AN
3	A[1]	23	AN[1]
4	A0	24	AN0
5	A1	25	AN1
6	A2	26	AN2
7	A3	27	AN3

6.3.4 FB10. SetupVariables.

Esta función se ejecuta cuando la marca “Inicio” da la orden de ejecución del control Fuzzy, de acuerdo a la configuración dada por “IndicadorSISO”, Toma los valores de referencia y las mediciones debidamente normalizadas en el “FB20” y “FB30”, toma los valores de error del ciclo anterior almacenados en la memoria y calcula el error y variación del error de las señales de entrada tomadas.

La figura 44 muestra de manera esquemática las entradas y salidas de la función y la estructura de datos de su bloque de datos de instancia.

Figura 44. Esquema general de flujo en los bloques de instancia de la actualización de variables.

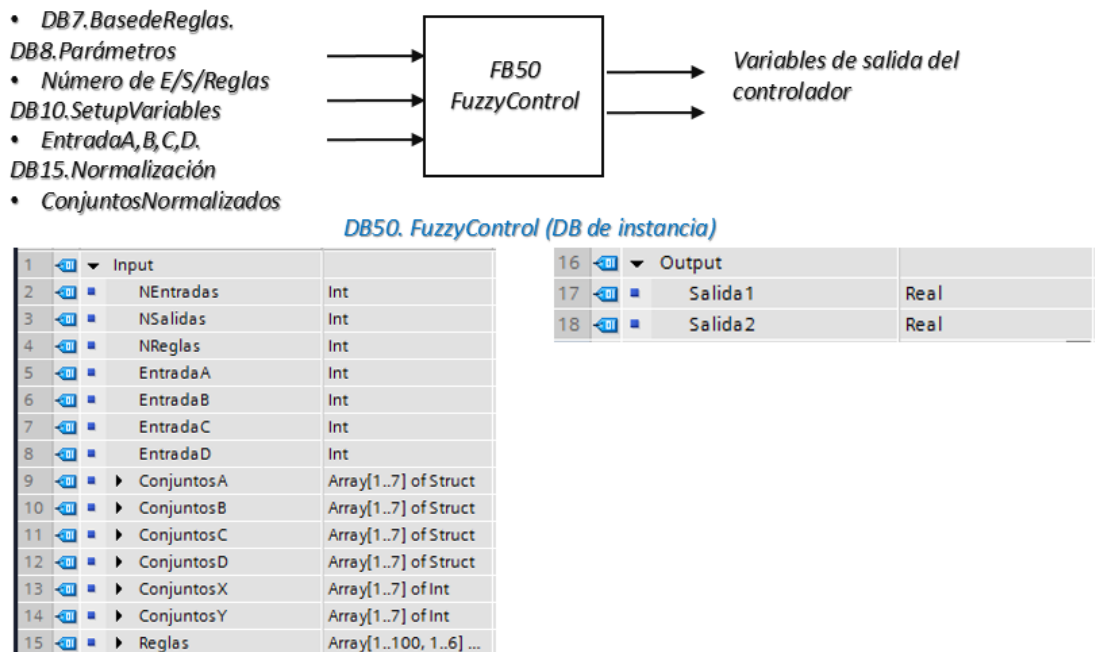


6.4 EJECUCIÓN DE FUZZYCONTROL

FuzzyControl es el bloque encargado de realizar todo el proceso de lógica Fuzzy para determinar las acciones de control. El bloque se encarga de hacer la llamada los bloques de funciones encargados de hacer el proceso de inferencia Fuzzy, implicación y defuzificación de la variable de salida. Para ello, requiere las variables de entrada (*DB10*), los conjuntos de las variables de entrada y de salida debidamente normalizados (*DB15*), la base de reglas (*DB7*), el número de entradas, de salidas y de reglas del modelo (*DB8*). Al final entrega las variables de salida del control.

La figura 45 muestra de manera esquemática las entradas y salidas de la función, además la estructura de datos de su bloque de datos de instancia.

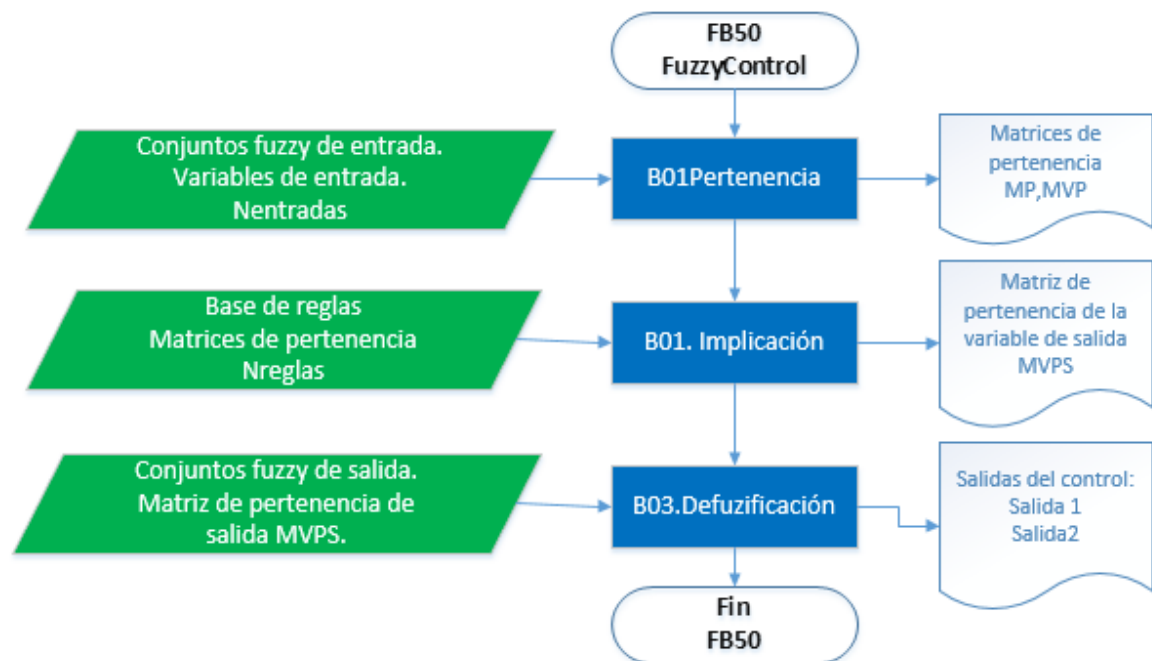
Figura 45. Esquema general de datos en los bloques de instancia del FuzzyControl.



En la siguiente figura, se observa el diagrama de flujo con la secuencia en que *FuzzyControl* realiza la llamada a los bloques de funciones de inferencia, implicación y defuzificación, identificando los datos que necesita en cada etapa para su funcionamiento.

Los bloques de funciones llamados toman los datos de entrada desde el bloque de datos de instancia de *FuzzyControl*.

Figura 46. Flujo de ejecución del FuzzyControl.



6.4.1 FB51. B01Pertenenencia.

Este bloque es el encargado de determinar el grado de pertenencia de cada variable de entrada dentro de los conjuntos Fuzzy en los que fue categorizada.

El bloque de datos de instancia define como entradas los conjuntos de entrada y las variables de entrada. Como salida entrega dos matrices MP y MVP cuya estructura se muestran en la siguiente tabla.

Tabla 5. Estructura de la variable MP y MVP.

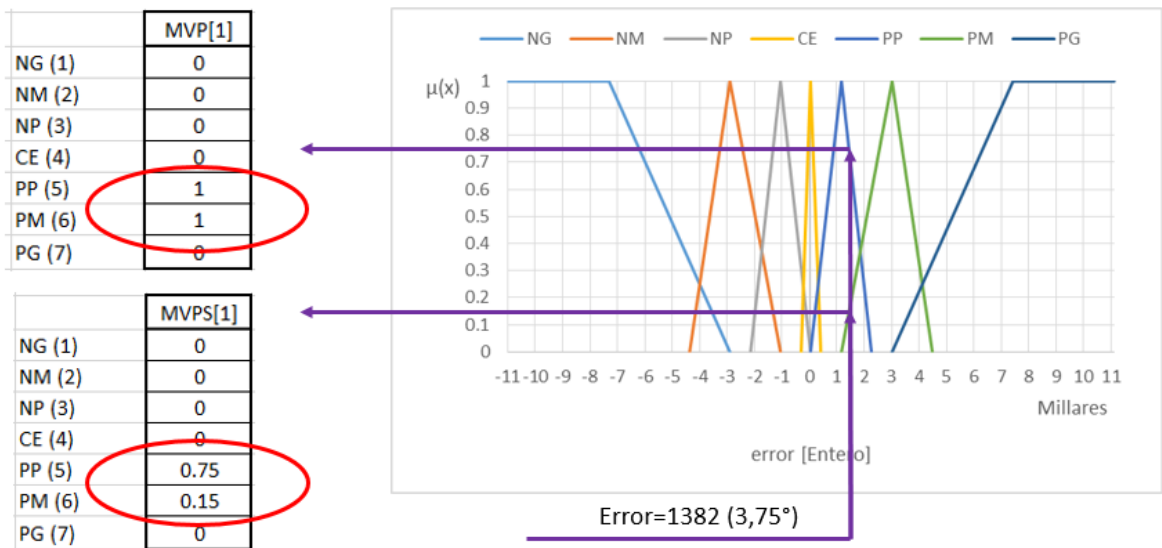
Índices	1 (Variable A)	2 (Variable B)	3 (Variable C)	4 (Variable D)
0 No Interviene	1	1	1	1
1 Conjunto 1	0	0	0	0
2 Conjunto 2	0	0	0	0
3 Conjunto 3	0	0	0	0
4 Conjunto 4	0	0	0	0
5 Conjunto 5	0	0	0	0
6 Conjunto 6	0	0	0	0
7 Conjunto 7	0	0	0	0

La matriz MP guarda la pertenencia de la variable a dicho conjuntos (valor 0 o 1), mientras la matriz MVP almacena el grado de pertenencia (valor de 0 hasta 1). Cada columna de la matriz hace referencia a una variable y las filas hacen referencia a cada uno de los conjuntos. La fila 0 resaltada en amarillo se ha reservado para incluir la opción cuando la variable no interviene en la formulación de reglas y se establece por defecto en 1.

El proceso de cálculo de pertenencia se realiza a través del siguiente proceso para cada variable de entrada:

1. Evalúa el número de conjuntos Fuzzy en donde debe comparar la variable. Para ello, descarta aquellos conjuntos cuyo tipo (elemento A0) este evaluado en cero.
2. Realiza un barrido del valor de la variable de entrada a través de sus conjuntos, la condición para que este valor tenga un grado de pertenencia a algún conjunto consiste en Punto Izquierda <valor< Punto derecha. Al pertenecer, marca el valor 1 en el elemento de la matriz MP para identificar la pertenencia a dicho conjunto (Ver figura 47).

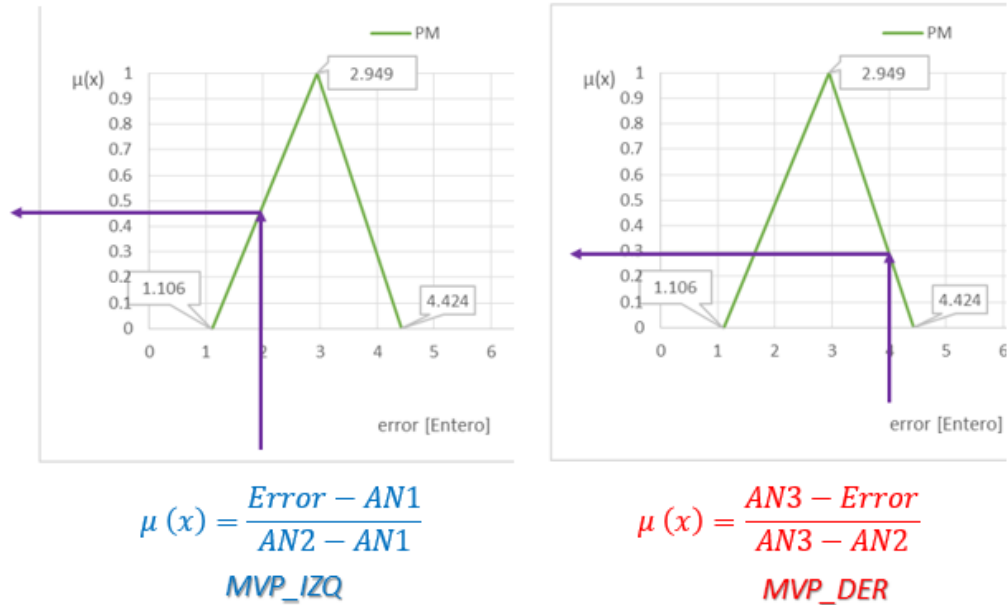
Figura 47. Proceso de cálculo de la pertenencia.



3. Realiza el cálculo del grado de pertenencia a de acuerdo al lugar donde la variable corte al conjunto. Si lo corta del lado con pendiente positiva utiliza la función FC1.MVP_IZQ que realiza la interpolación entre el punto izquierdo y el punto central del conjunto.

En cambio, si la variable corta al conjunto del lado con pendiente negativa, utiliza la función FC2.MVP_DER que realiza la interpolación entre el punto central y el punto derecho del conjunto (Ver figura 48).

Figura 48. Proceso de cálculo funciones MVP_IZQ y MVP_DER.

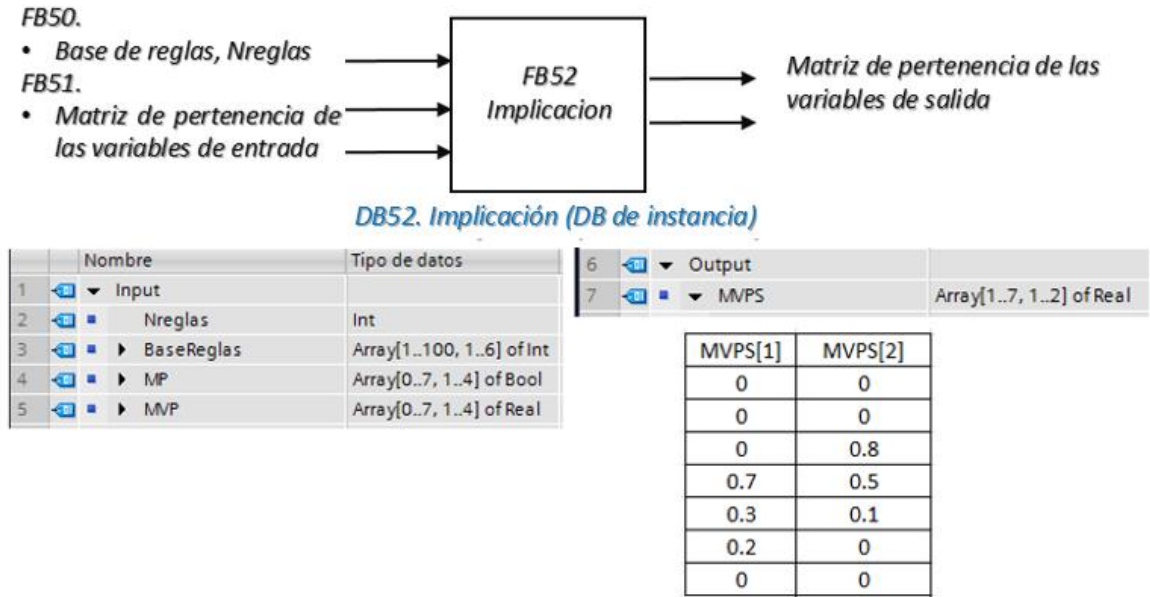


6.4.2 FB52. B02Implicación.

Este bloque evalúa cada una de las reglas del modelo Fuzzy para determinar las acciones de salida del control como consecuencia de una serie de reglas que se cumplen.

La figura 49 muestra de manera esquemática las entradas y salidas de la función y la estructura de su bloque de datos de instancia. La salida de este bloque de funciones es una matriz de dos (2) columnas que me entrega el valor de pertenencia de los conjuntos de cada una de las variables de salida.

Figura 49. Esquema general de datos en los bloques de instancia de la implicación.



El proceso de implicación es el siguiente:

1. Para cada regla toma el valor de sus cuatro (4) conjuntos que conforman las condiciones de cada variable de entrada y evalúa en la matriz si estos índices están activados (Ver figura 50).

Figura 50. Proceso de implicación de reglas Fuzzy.

Base de reglas	Entrada 1	Entrada 2	Entrada 3	Entrada 4	Salida 1	Salida 2
Regla 1	1	2	0	0	1	0

MP(MVP)	Entrada 1	Entrada 2	Entrada 3	Entrada 4
0	1/1	1/1	1/1	1/1
1	1(0.3)	0	0	0
2	0	1(0.5)	0	0
3	0	0	0	0
4	0	0	0	0
5	0	0	0	0
6	0	0	0	0
7	0	0	0	0

2. Compara el grado de pertenencia MVP [conjunto, entrada], para determinar cuál es el menor de todos:

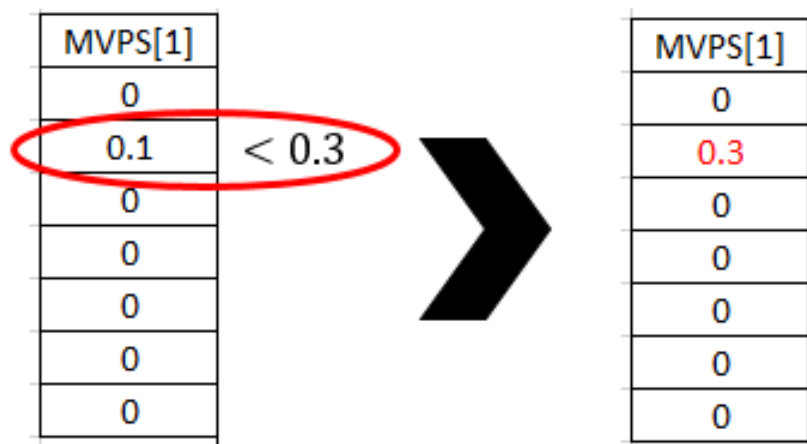
$$MVP[1,1] < MVP[2,2] < MVP[0,3] < MVP[0,4]$$

$$Valor = MVP[1,1]$$

3. Toma el índice del valor de las salidas y compara el grado de pertenencia obtenido del paso 2 con el valor dentro de la matriz de pertenencia de la variable de salida.

Sí $MVPS[\text{índice}] < Valor$, entonces $MVPS[\text{índice}] = Valor$ (Ver figura 51).

Figura 51. Aplicación del método MIN-MAX.

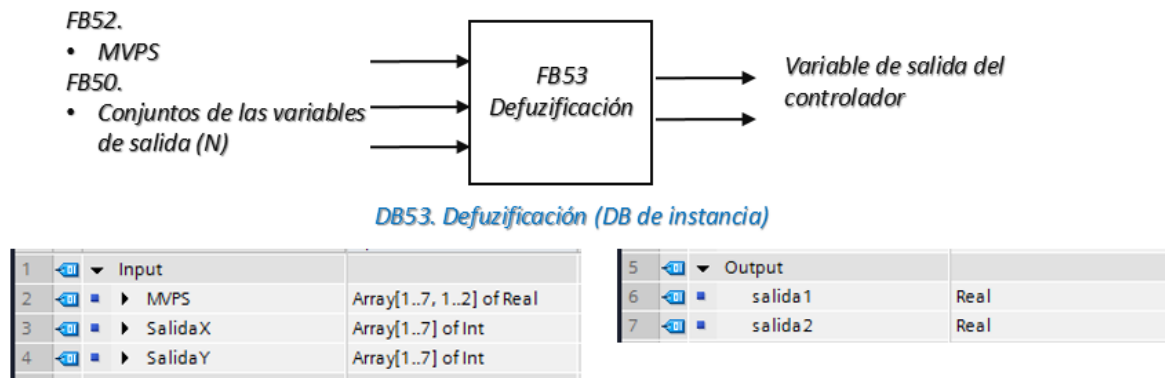


6.4.3 FB53. B03Defuzificación.

Este bloque calcula el valor de la variable de salida a partir de la matriz de pertenencia obtenida en el bloque de implicación de reglas.

La figura 52 muestra de manera esquemática las entradas y salidas de la función y la estructura de su bloque de datos de instancia.

Figura 52. Esquema general de datos en los bloques de instancia defuzificación.



El proceso de defuzificación consiste en realizar una ponderación de sus conjuntos en función de los valores de pertenencia de cada uno de ellos. Para ellos se aplica la fórmula mostrada en la figura:

Figura 53. Defuzificación por método de pesos concentrados.

XN	MVPS[1]	MVPS[1] * XN	
-1382	0	0	
-553	0	0	
-276	0	0	
0	0	0	
276	0.75	207	
553	0.15	83	
1382	0	0	Salida X
Suma	0.9	290	323

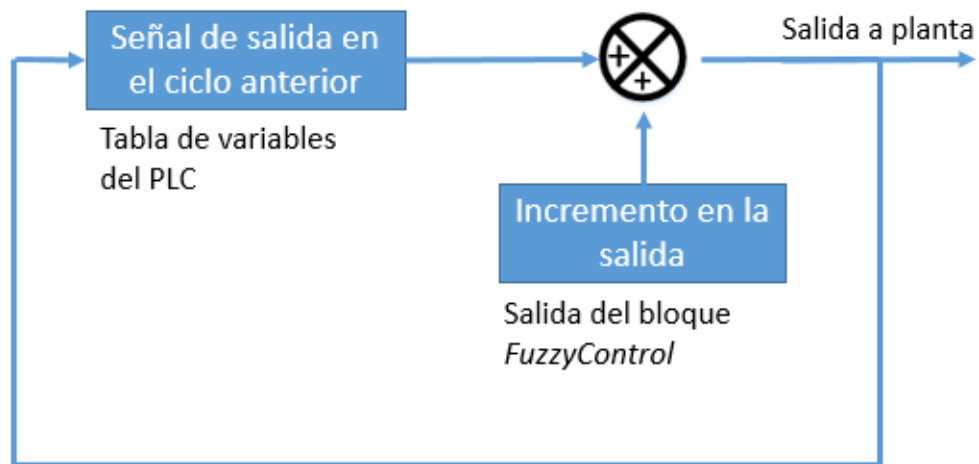
$$Salida X = \frac{\sum MVPS[i] * XN[i]}{\sum MVPS[i]}$$

6.5 PREPARACIÓN DE LAS VARIABLES DE SALIDA A LA PLANTA

Esta etapa busca calcular el valor total de la señal de salida que se debe enviar al actuador. Toma el valor incremental resultante del proceso de lógica Fuzzy y lo agrega al valor de la salida del ciclo anterior, que se encuentra dentro de una posición de memoria del PLC identificada en la tabla (Ver en la tabla 4 página 58).

En la figura se muestra de manera esquemática el proceso de acumulación de la variable de salida.

Figura 54. Esquema general del proceso de acumulación de la variable de salida.



7. IMPLEMENTACIÓN DEL CONTROLADOR FUZZY

Las pruebas llevadas a cabo se ejecutaron en el banco “Ducto de viento para recrear automatismos” que está ubicado en el laboratorio de automatización industrial de la escuela de ingeniería mecánica de la universidad industrial de Santander.

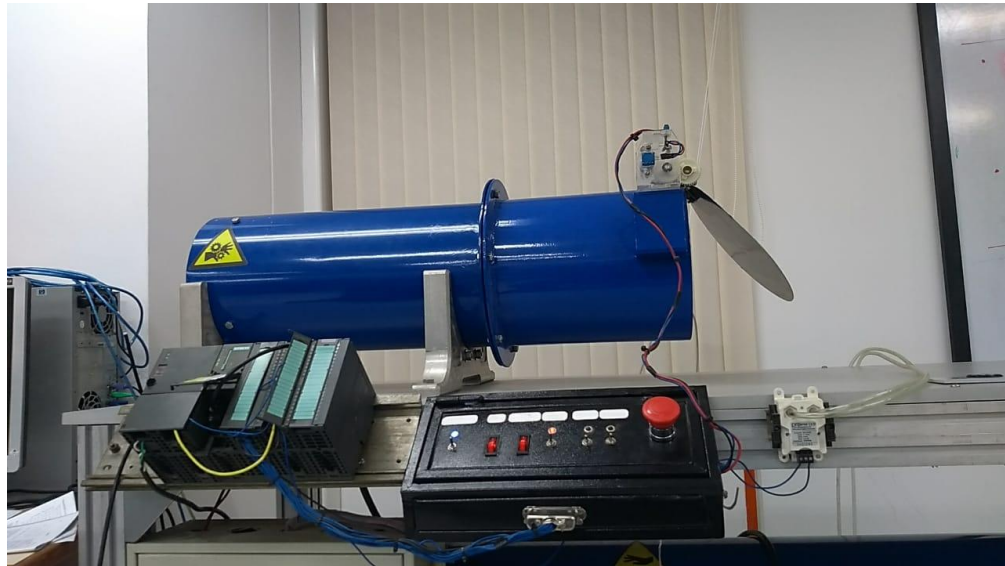
Las pruebas consistieron en realizar el control Fuzzy correspondiente para la simulación de un proceso SISO (una variable de entrada y una variable de salida) y un proceso MIMO (múltiples variables de entrada y múltiples variables de salida), configurables en el banco de pruebas.

Vale aclarar que en el desarrollo de las pruebas se ajustaron constantemente las bases de reglas, los conjuntos de entrada y los conjuntos de salida, en función de obtener el mejor comportamiento de la ejecución del control, mostrando a continuación la mejor sintonización para cada sistema a evaluar.

7.1 SISTEMA DE CONTROL DE POSICIÓN ANGULAR (SISO)

La simulación del proceso SISO radicó en controlar la posición angular como variable de entrada, para la configuración mostrada en la figura 55 del banco. Dicho control se genera mediante la graduación de la velocidad del flujo de aire generado por el motor del ventilador incorporado en el ducto. La velocidad del motor se regula por un preactuador que emite señales PWM en relación con el porcentaje de potencia del motor, generando la variable de salida para el modelo Fuzzy a implementar.

Figura 55. Configuración para sistema de control de posicionamiento angular (SISO).

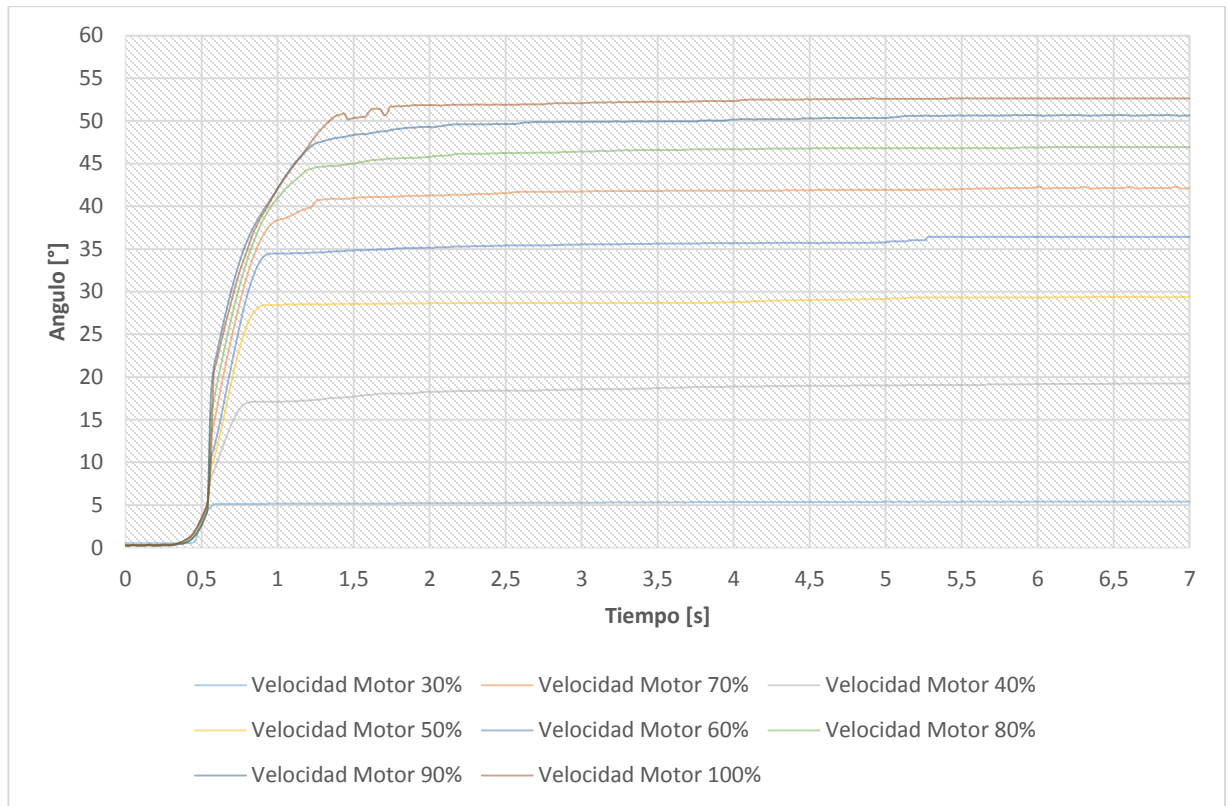


7.1.1 Caracterización del sistema SISO.

Se caracterizó el sistema para identificar los rangos de operación, los tiempos de respuesta, el tiempo de estabilización y el comportamiento dinámico en general para alcanzar el estado estable.

La caracterización consistió en encender el motor del ventilador a unos porcentajes determinados de su potencia, visualizando el comportamiento generado en la posición angular efectuada por el área de impacto del flujo de aire en la compuerta de salida cuando ésta ha alcanzado su estado estable. En la Figura 56, se puede observar el comportamiento del sistema en los porcentajes de potencia ejecutados.

Figura 56. Gráfica del comportamiento en la caracterización del sistema SISO.



En la figura 56, se pudo observar que el tiempo que transcurre desde que el motor del ventilador inicia hasta donde se observa la estabilización por el área de impacto del flujo del aire en la compuerta de salida del ducto, es pequeño debido a la poca inercia del mismo; en su caso crítico, cuando la potencia del motor es del 100% la estabilización de la compuerta en un ángulo de 52° sucede en 1.5 segundos.

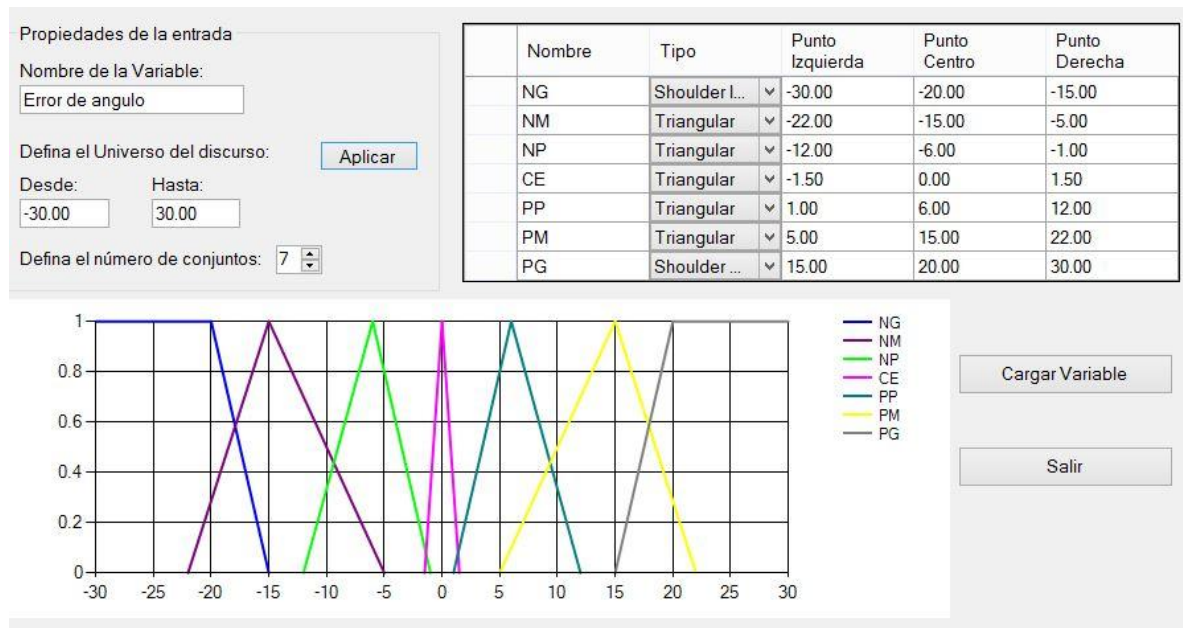
Observando el comportamiento del sistema para llegar a estado estable obtenemos las herramientas necesarias para poder determinar un modelo Fuzzy inicial y ajustar las variables para obtener un tiempo de respuesta similar al de estabilización.

7.1.2 Creación de los conjuntos de las variables de entrada.

Las variables de entrada del modelo Fuzzy determinado para realizar el control de la posición angular de la compuerta están explícitas por el error, definido como la diferencia entre el ángulo deseado y el medido; y la variación del error. Esto debido a que se procedió a diseñar un modelo que pudiese controlar el ángulo de la compuerta con un tiempo de respuesta igual o similar al de caracterización.

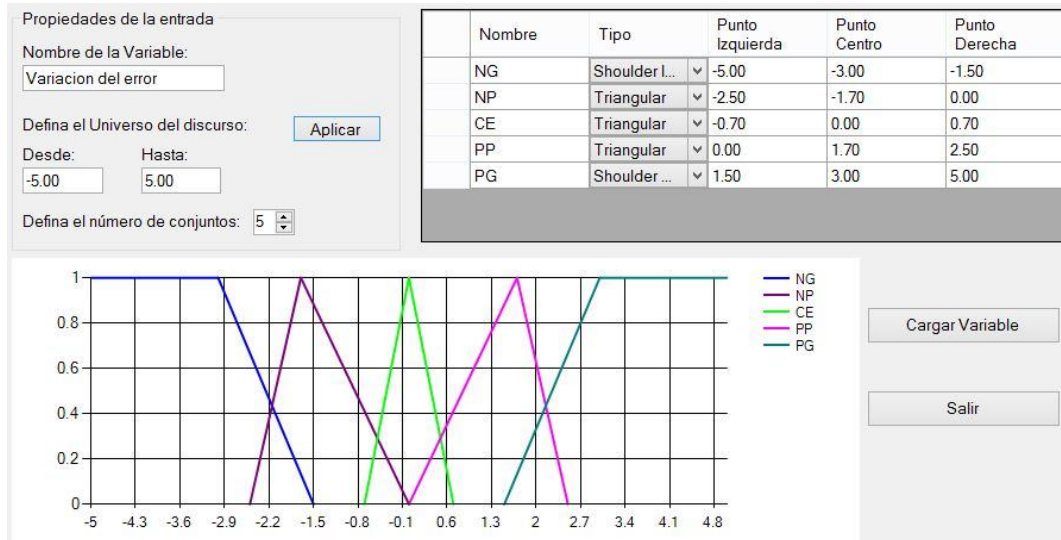
En la variable error se emplearon siete (7) conjuntos difusos definidos como se observa en la figura 57.

Figura 57. Definición de los conjuntos de la variable de entrada "Error de ángulo".



En la variable variación del error se emplearon cinco (5) conjuntos difusos definidos como se observa en la figura 58.

Figura 58. Definición de los conjuntos de la variable de entrada "Variación del error".



Los conjuntos de las variables de entrada se definieron en nombre bajo la notación descrita en la tabla 6.

Tabla 6. Designación de nombres para sistema SISO.

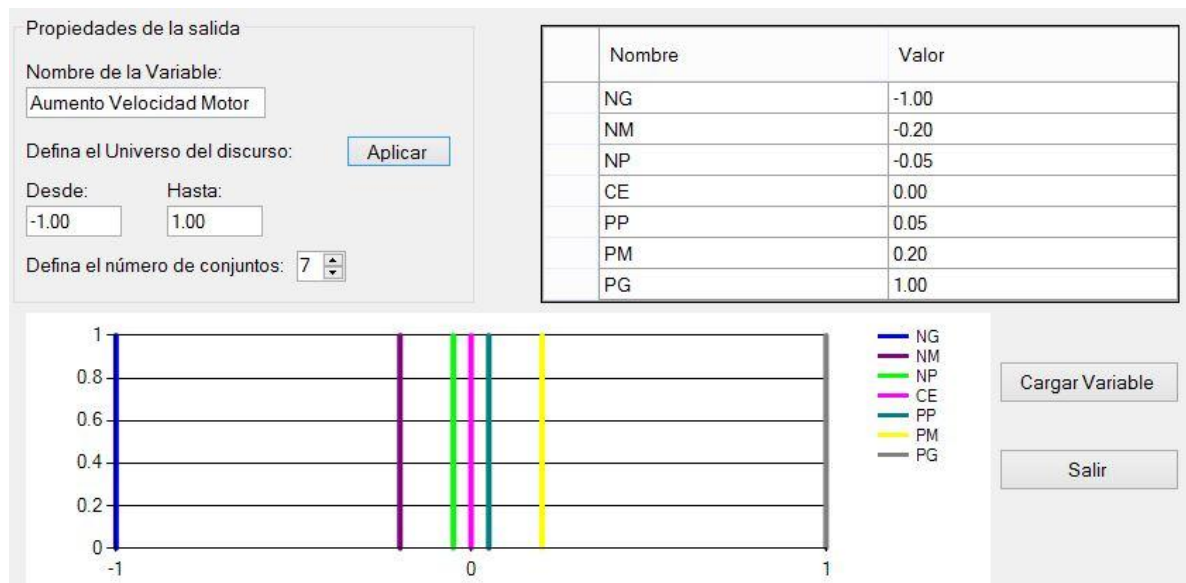
Nombre	Descripción
NG	Negativamente grande
NM	Negativamente mediano
NP	Negativamente pequeño
CE	Cercano
PP	Positivamente pequeño
PM	Positivamente mediano
PG	Positivamente grande

7.1.3 Creación del conjunto de la variable de salida.

La variable de salida o manipulada del modelo Fuzzy determinado para realizar el control de la posición angular de la compuerta está proporcionada por el aumento de la velocidad del motor, definida con el cambio de 0% a 100% de la potencia enviada a la tarjeta de regulación (preactuador de la hélice añadida al motor).

En la variable de aumento de velocidad del motor se emplearon siete (7) conjuntos difusos predeterminadamente definidos de tipo singleton, asignándoles valores como se observa en la figura 59.

Figura 59. Definición de conjuntos de la variable de salida "Aumento velocidad del motor".



Para definir los nombres de los conjuntos de la variable de salida utiliza la notación descrita en la tabla 6, donde se definen los nombres de los conjuntos de las variables de entrada.

7.1.4 Creación de la base de reglas de control.

La base de reglas de control para el modelo Fuzzy determinado se configuró como se observa en la tabla 7, constituida a partir del comportamiento observado en la caracterización del sistema, partiendo que la base obedece a la estructura “SI – ENTONCES”.

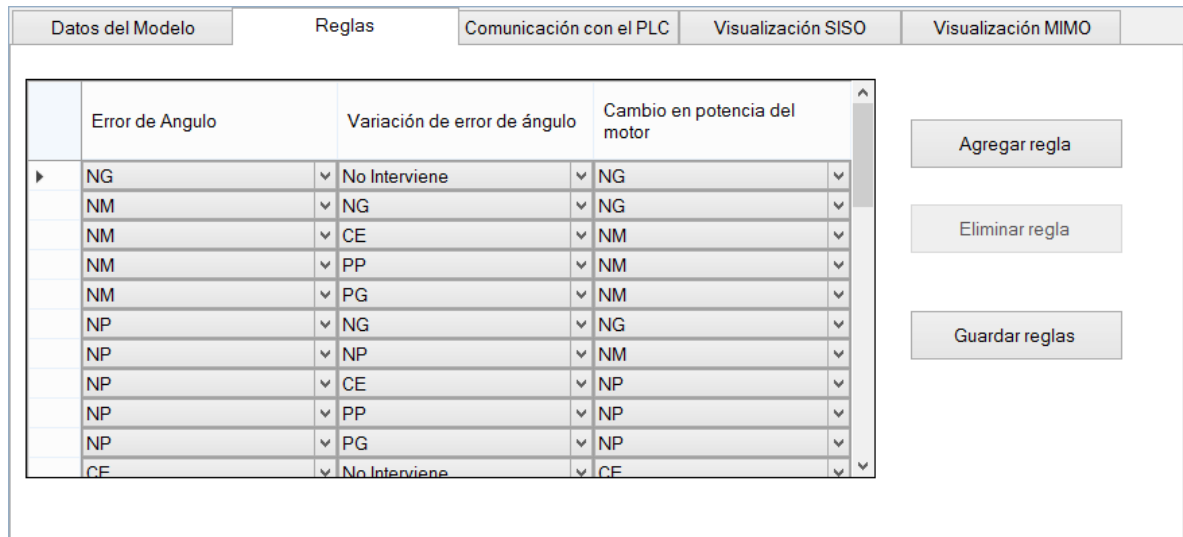
Tabla 7. Estructuración de la base de reglas para el sistema SISO.

BASE REGLAS		VARIACIÓN DEL ERROR				
		NG	NP	CE	PP	PG
E R R O R	NG	NG	NG	NG	NG	NG
	NM	NG	NG	NM	NM	NM
	NP	NG	NM	NP	NP	NP
	CE	CE	CE	CE	CE	CE
	PP	PP	PP	PP	PM	PG
	PM	PM	PM	PM	PG	PG
	PG	PG	PG	PG	PG	PG

La notación determinada para la definición de las reglas obedece a la tomada para los conjuntos de las variables de entrada y los conjuntos de las variables de salida, como se especifica en la tabla 6.

Una vez definidas las reglas se editaron en el módulo correspondiente dentro de la interfaz creada como se observa en la figura 60, y se realizó la ejecución del controlador Fuzzy con el modelo creado.

Figura 60. Edición de base de reglas determinadas en la interfaz gráfica.



7.1.5 Ejecución del modelo Fuzzy creado mediante el FuzzyControl desarrollado.

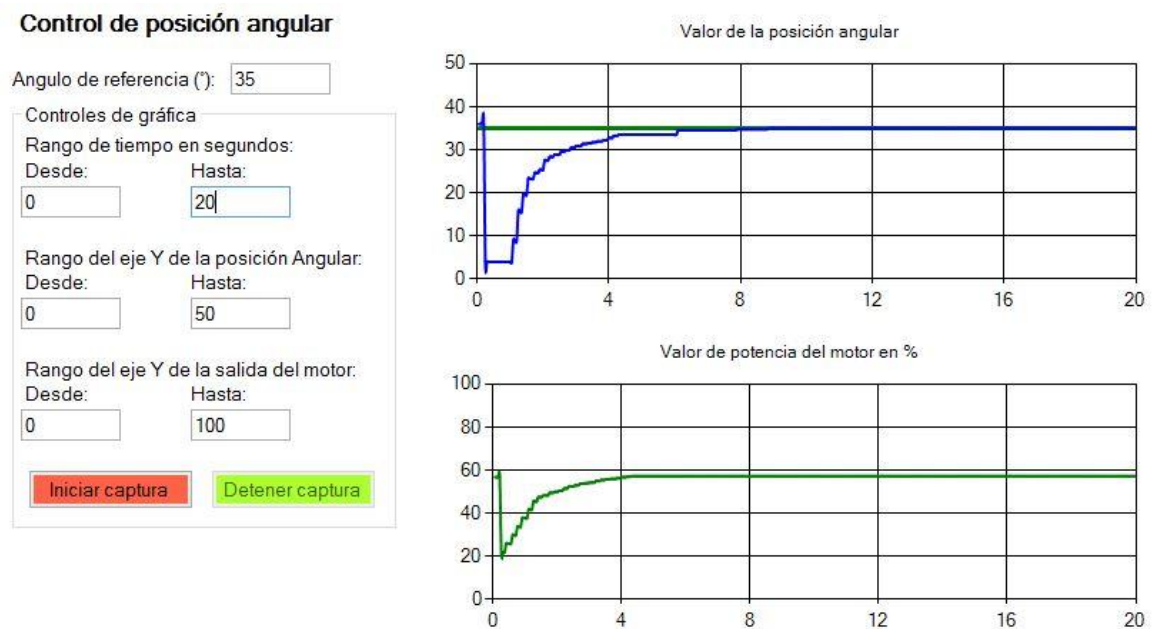
Se ejecutó el modelo Fuzzy creado para el control de la posición angular de la compuerta introduciendo tres (3) valores de referencia independientes como se observa en la tabla __, con el fin de observar el comportamiento y si es el caso sintonizar la base de reglas o los conjuntos de entrada y salida.

Tabla 8. Valores de referencia para ejecución de pruebas del sistema SISO.

Ítem	Angulo de referencia
1	25°
2	35°
3	45°

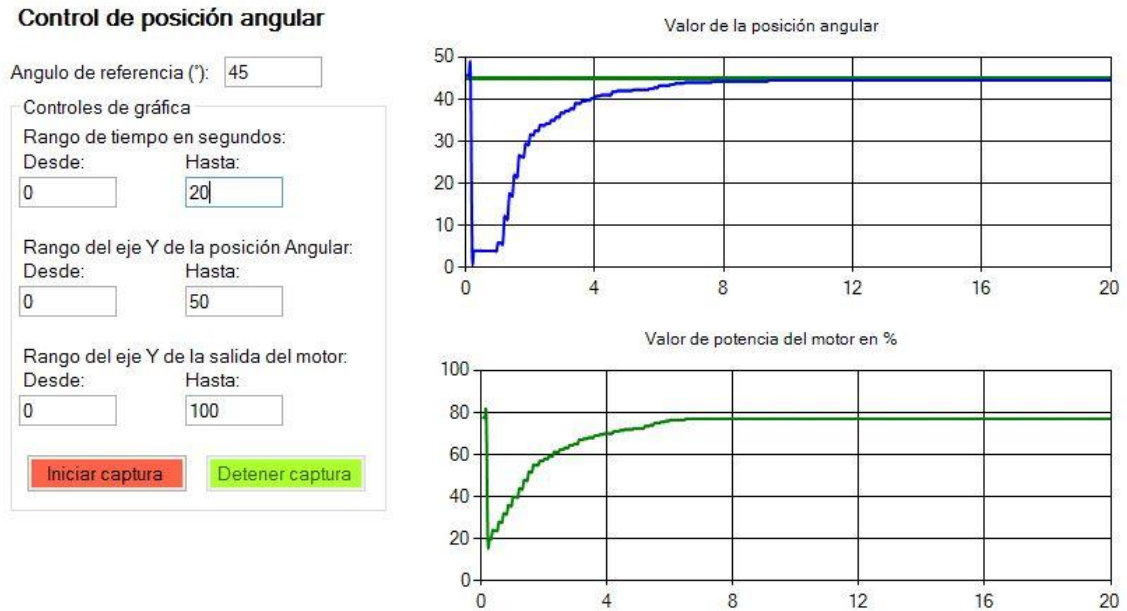
Se introdujo inicialmente el ángulo de referencia de 35°, editando el valor en el módulo de visualización de la interfaz como se puede observar en la figura 61, se observó que el modelo Fuzzy creado realizaba el proceso de control posicionando la compuerta en el ángulo deseado entrando en un régimen estable a los 5 segundos.

Figura 61. Edición del primer valor de referencia para ejecución de pruebas SISO.



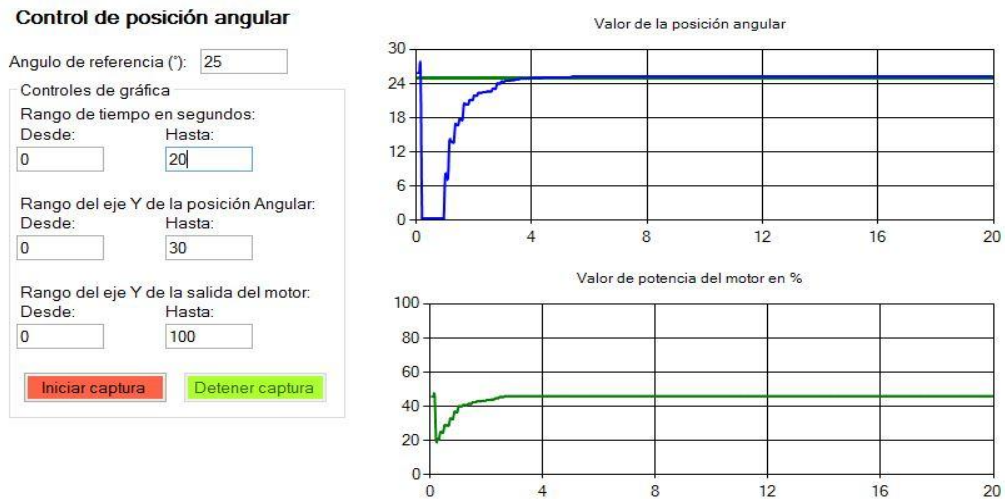
Posteriormente se asignó el ángulo de referencia de 45°, editando el valor en el módulo de visualización como se observa en la figura 62, el comportamiento observado se asemeja al encontrado anteriormente, posicionando la compuerta con un tiempo de estabilización alrededor de los 6 segundos.

Figura 62. Edición del segundo valor de referencia para ejecución de pruebas SISO.



Para observar el comportamiento del modelo en una posición crítica se asignó el ángulo de referencia en 25° como se observa en la figura 63, siendo un ángulo cercano a su posición natural con ausencia de flujo de aire. El comportamiento obtenido se asemeja a los anteriores, controlando la posición de la compuerta en el ángulo deseado con un tiempo de respuesta no superior a los 4 segundos.

Figura 63. Edición del tercer valor de referencia para ejecución de prueba SISO.



Para validar la funcionalidad del modelo Fuzzy se decidió realizar una última prueba donde se ejecutaba el modelo cambiando constantemente los ángulos de referencia como se observa en la figura 64, con el fin de verificar el seguimiento del control críticamente. Obteniendo como se esperaba, una respuesta adecuada en el posicionamiento de la compuerta en los ángulos deseados.

Figura 64. Ejecución de prueba de seguimiento para el sistema SISO.



De esta figura se detecta un problema de histéresis en el sistema de medición de posición angular debido al desgaste en el sistema de engranajes conectado al potenciómetro y una desalineación en el eje de la compuerta. Se observa que para una medición inicial de 35° cuando se está incrementando gradualmente el valor de referencia, el motor gira casi a un 60% de su velocidad máxima. Sin embargo, cuando se está reduciendo este valor y se llega de nuevo a los 35°, el motor gira a una velocidad aproximada de un 50% de su velocidad máxima.

7.1.6 Análisis de la ejecución del modelo Fuzzy creado mediante el FuzzyControl desarrollado.

De los resultados obtenidos en la ejecución del modelo Fuzzy creado para el sistema de posicionamiento angular de la compuerta, se puede afirmar lo siguiente:

- La caracterización del sistema evidencia que el posicionamiento angular obedece a un comportamiento de primer orden con respuesta rápida, visualizando un rango de oscilación en la posición angular de la compuerta no mayor a 52° con su máxima capacidad de flujo. Esto evita que se requiera de un modelo Fuzzy riguroso para poder realizar el control adecuado sobre el sistema.
- La variable de entrada “*error*” se categoriza en siete (7) conjuntos, obteniendo un margen más detallado de la posición de la compuerta, con el fin de abordar de manera minuciosa el estado de transición de la posición angular en el sistema. La intención de ubicar conjuntos de tipo shoulder en los conjuntos extremos del universo del discurso garantiza que éstos abarquen todo el rango de la misma.
- La variable de entrada “*variación del error*” se categorizan en cinco (5) conjuntos, permitiendo identificar de manera más oportuna la celeridad con la que se está posicionando la compuerta al valor de referencia seleccionado.
- La base de reglas evidencia la cautela con la que se modela el control Fuzzy, pues solo se ordena ejecutar acciones fuertes siempre y cuando las variables de entrada se encuentren en los límites extremos del universo del discurso. Entre más cercano se encuentre la compuerta de su valor deseado, más prudentes se vuelven las acciones de ejecución.

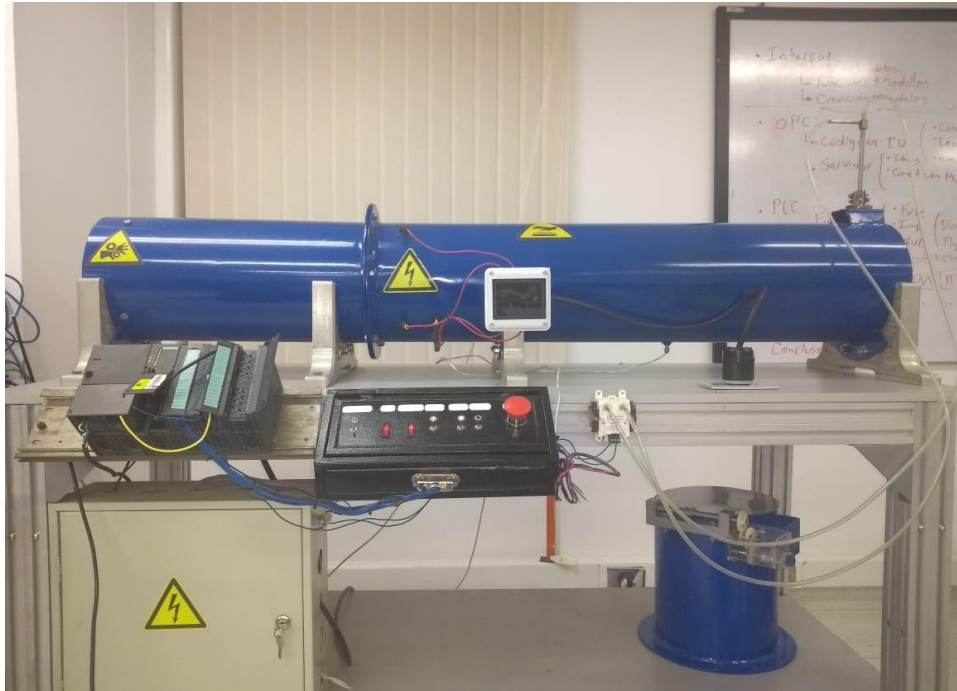
- Si se observa la respuesta del sistema en la posición angular de referencia, en ningún momento se presenta un “*overshoot*”, esto evidencia que las reglas están siendo ejecutadas de manera correcta realizando un incremento pequeño en la potencia del motor en comparación a su incremento mayor en la fase de acercamiento rápido. Sin embargo, es fácil evidenciar que mientras más alejado se encuentre el ángulo inicial de su valor de referencia, este tardará más tiempo en alcanzar el valor deseado y estabilizar.
- La potencia del motor en estado estable obedece a los valores obtenidos dentro de la caracterización del sistema. Esto satisface la configuración de la “*variable de salida*”, categorizada en siete (7) conjuntos, la cual no ejecuta acciones de incremento considerables en su cambio de conjunto a conjunto.
- Por último, se destaca el comportamiento homogéneo del modelo Fuzzy cuando se ejecuta la prueba de seguimiento, pues los valores de referencia son constantemente cambiantes y su funcionalidad no se ve afectada por su forma de acercamiento positivo o negativo al valor de referencia angular deseado.

7.2 SISTEMA DE CONTROL DE VELOCIDAD Y TEMPERATURA DEL AIRE (MIMO)

La ejecución del control MIMO radicó en controlar la velocidad y la temperatura como variables de entrada, para la configuración mostrada en la figura 65 del banco. Dicho control se genera mediante la graduación de la velocidad del flujo de aire generado por el motor del ventilador incorporado y la graduación de la temperatura generada por la resistencia incorporada en el ducto. La velocidad del motor se regula por un preactuador que emite señales PWM en relación con el porcentaje de potencia del motor, generando la primera variable de salida para el modelo Fuzzy la segunda variable está relacionada con la regulación de la temperatura bajo la

misma funcionalidad debido a que existe un preactuador que permite cuantificar y relacionar la potencia de consumo con la capacidad de generación de temperatura.

Figura 65. Configuración para sistema de control de velocidad y temperatura del aire (MIMO).



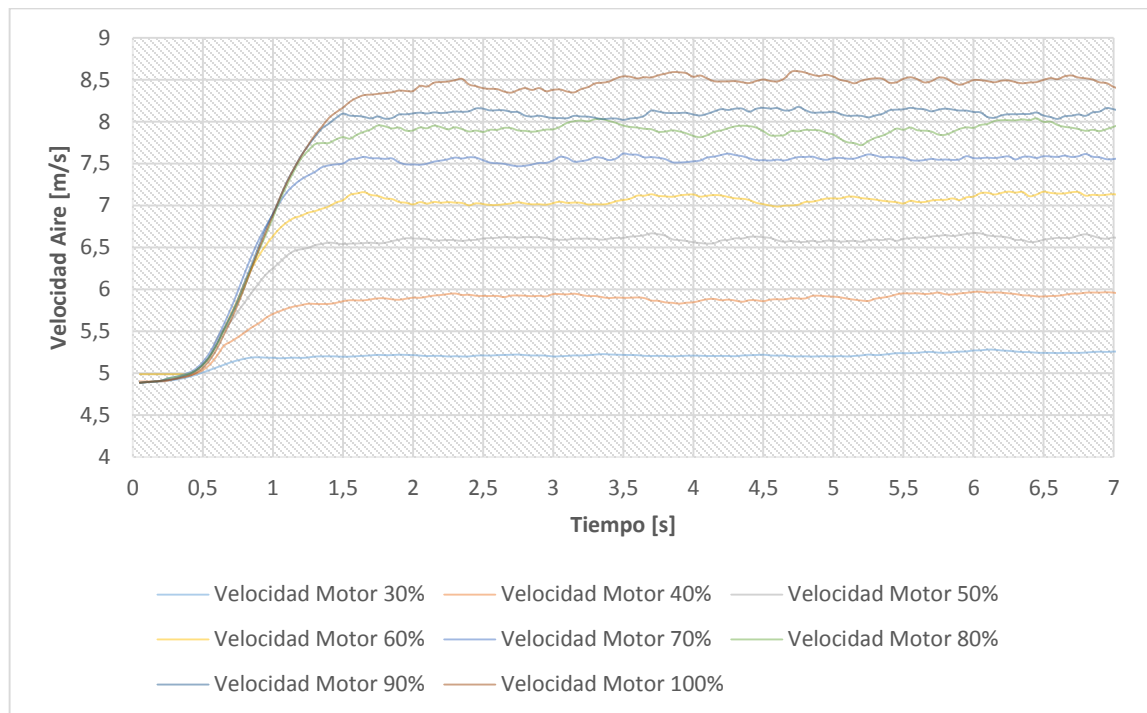
7.2.1 Caracterización del sistema MIMO.

Se caracterizó el sistema MIMO de la misma forma que el SISO, para identificar los rangos de operación, los tiempos de respuesta, el tiempo de estabilización y el comportamiento dinámico en general para alcanzar el estado estable en cada una de las variables deseadas.

La caracterización de la velocidad del flujo consistió en encender el motor del ventilador a unos porcentajes determinados de su potencia, visualizando el comportamiento de estabilización en función de la velocidad del flujo calculada

mediante la variación de la presión captada por el tubo pitot y su aplicación en la ecuación de Bernoulli para un fluido incompresible. En la figura 66 se puede observar el comportamiento del sistema en los porcentajes de potencia ejecutados.

Figura 66. Gráfica del comportamiento de la velocidad en la caracterización del sistema MIMO.

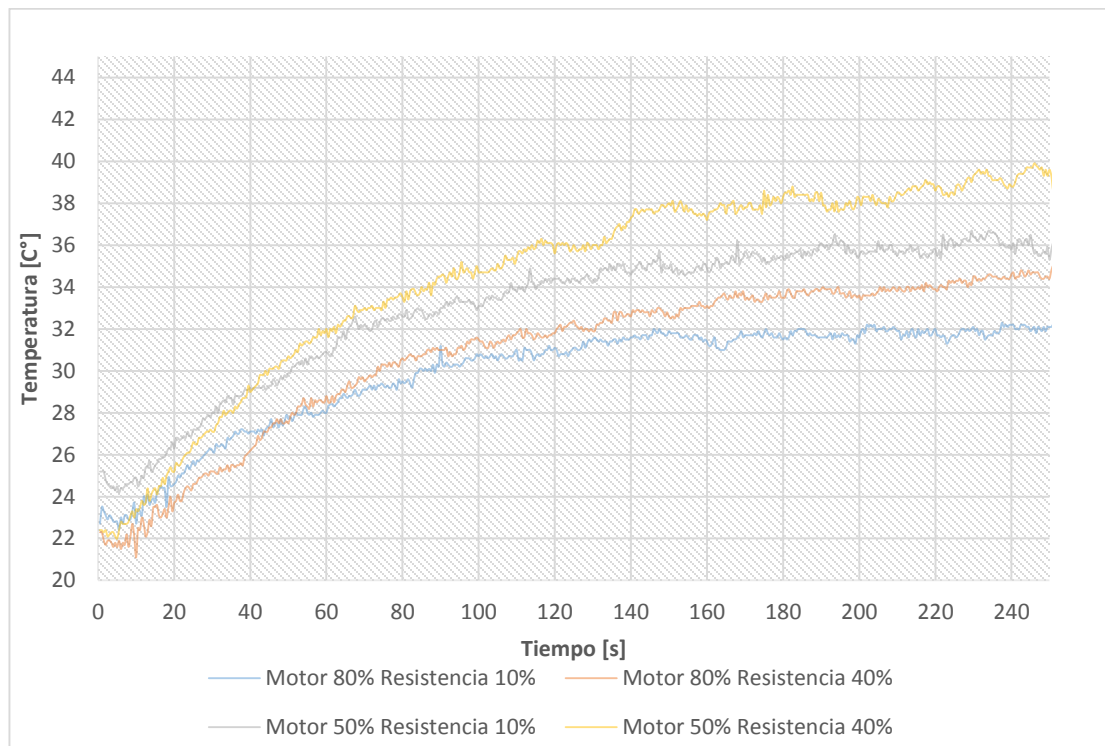


En la figura 66, se pudo observar que el tiempo que transcurre desde que el motor del ventilador inicia hasta donde se observa la estabilización de la velocidad del flujo, es muy corto debido a la sensibilidad y precisión del cálculo, además de la poca inercia a la que obedece el sistema; en su caso crítico, cuando la potencia del motor es del 100% la estabilización del flujo en 8.5 m/s se efectúa en un tiempo no superior a 1 segundo.

La caracterización de la temperatura consistió en encender al mismo tiempo el motor del ventilador a unos porcentajes determinados de su potencia, y la

resistencia a unos porcentajes dentro del mismo valor de referencia asignados, visualizando el comportamiento generado mediante la medición directa de la temperatura por la termorresistencia. En la figura 67, se puede observar el comportamiento del sistema en los porcentajes de potencia ejecutados.

Figura 67. Gráfica del comportamiento de la temperatura en la caracterización del sistema MIMO.



En la figura 67, se pudo observar que, a diferencia del comportamiento obtenido con la velocidad del aire, el tiempo que transcurre desde se enciende la resistencia hasta que se estabiliza la temperatura es mucho más largo, esto debido a que la inercia implícita en la temperatura es muy alta y va aumentando a nivel que se exige un valor más alto de referencia; en su caso crítico cuando la potencia de la resistencia

se eleva a un 40% con una potencia del motor de 50%, el sistema se estabiliza a 40°C transcurrido un tiempo de 250 segundos.

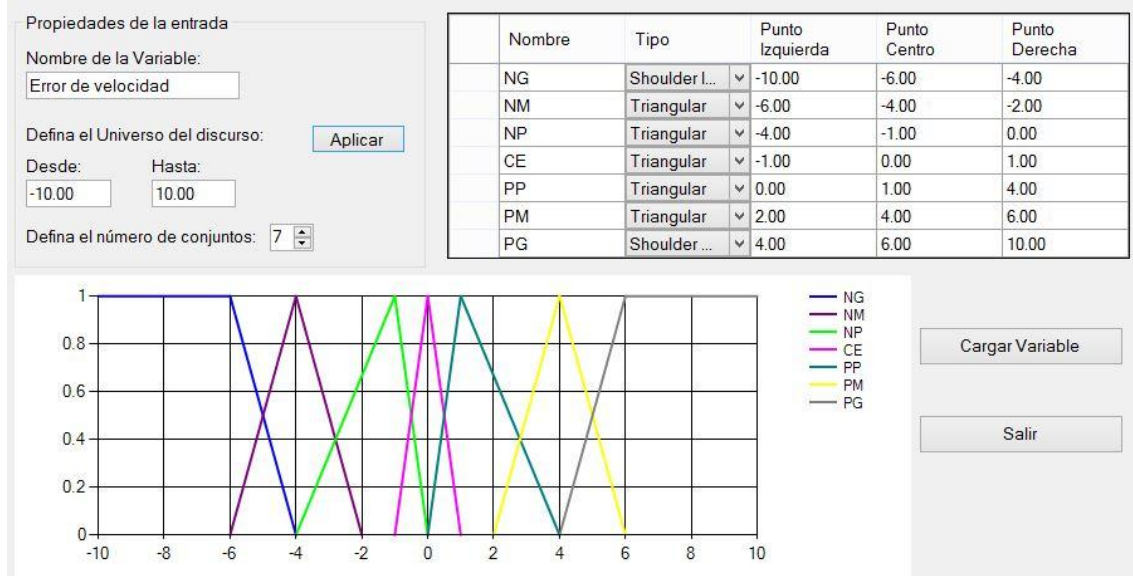
Observando el comportamiento del sistema en las dos variables para llegar a estado estable obtenemos las herramientas necesarias para poder determinar un modelo Fuzzy inicial.

7.2.2 Creación de los conjuntos de las variables de entrada para sistema MIMO.

Las variables de entrada del modelo Fuzzy determinado para realizar el control de la velocidad y temperatura del aire, se efectuaron bajo la misma lógica, tomando el error, definido como la diferencia entre el valor de referencia y el medido; y la variación del error, definido como la diferencia del error actual y el error anterior. Esto nos proporciona determinar cuánto nos falta para poder llegar y como se está llegando al valor de referencia.

En la variable error de velocidad se emplearon siete (7) conjuntos difusos definidos como se observa en la figura 68.

Figura 68. Definición de los conjuntos de la variable de entrada "Error de velocidad".



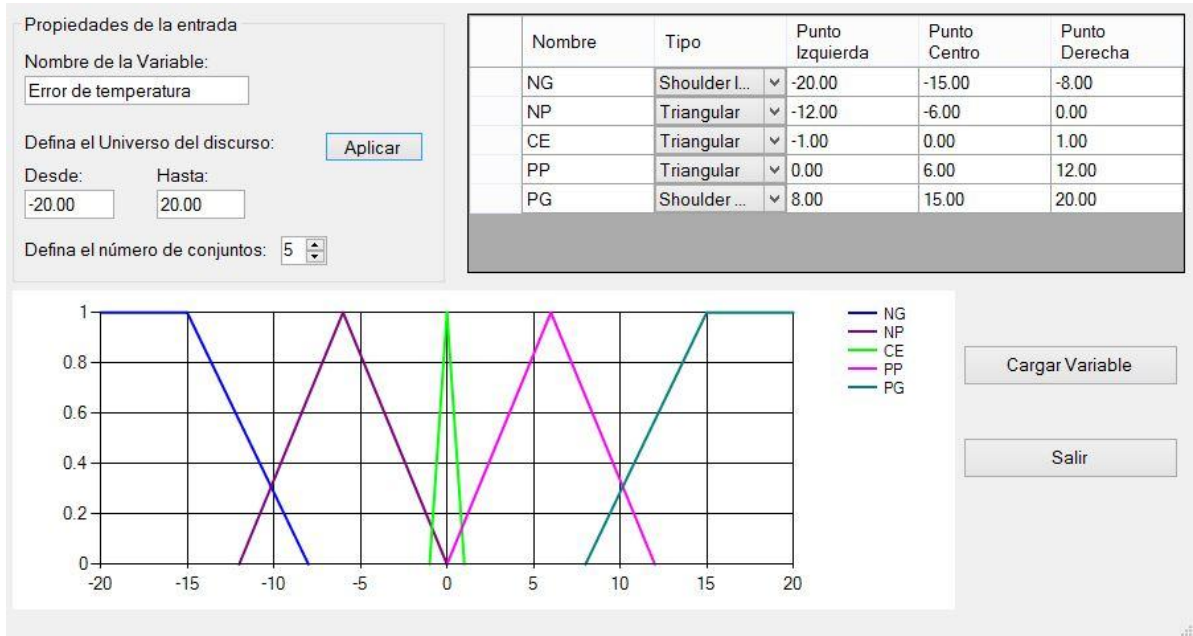
En la variable variación error de velocidad se emplearon cinco (5) conjuntos difusos definidos como se observa en la figura 69.

Figura 69. Definición de los conjuntos de la variable de entrada "Variación error de la velocidad".



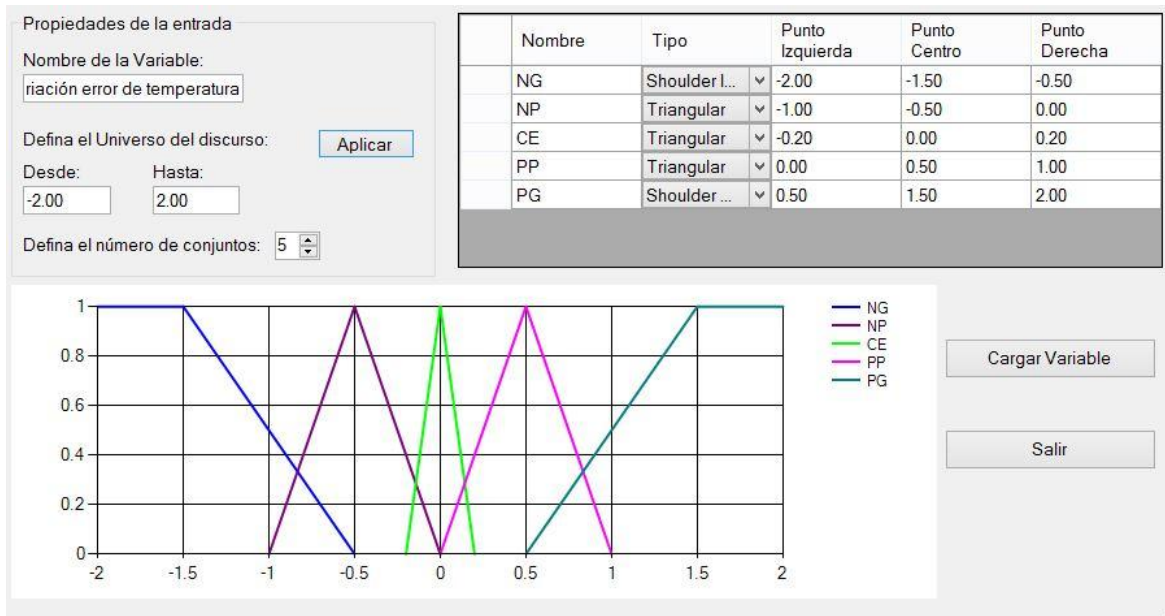
En la variable error de temperatura se emplearon cinco (5) conjuntos difusos definidos como se observa en la figura 70.

Figura 70. Definición de los conjuntos de la variable de entrada "Error de temperatura".



En la variable variación error de temperatura se emplearon cinco (5) conjuntos difusos definidos como se observa en la figura 71.

Figura 71. Definición de los conjuntos de la variable de entrada "Variación error de temperatura".



Los conjuntos de las cuatro (4) variables de entrada se definieron en nombre bajo la notación descrita en la tabla 9.

Tabla 9. Designación de nombres para sistema MIMO.

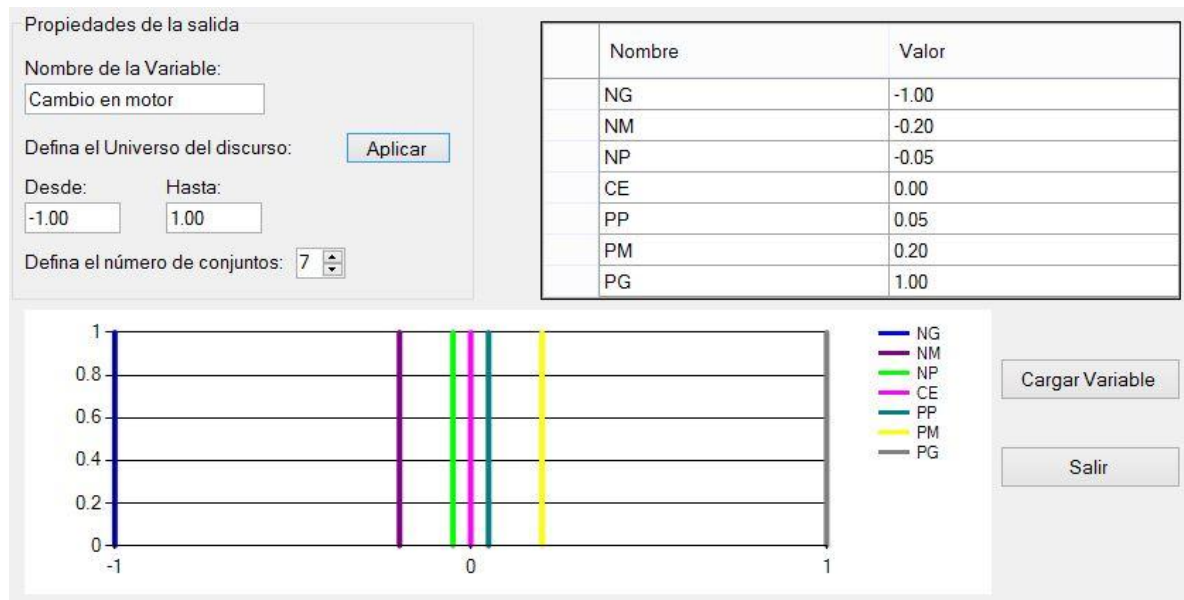
Nombre	Descripción
NG	Negativamente grande.
NM	Negativamente mediano.
NP	Negativamente pequeño.
CE	Cercano.
PP	Positivamente pequeño.
PM	Positivamente mediano.
PG	Positivamente grande.

7.2.3 Creación del conjunto de las variables de salida MIMO.

La variable de salida o manipulada del modelo Fuzzy determinado para realizar el control de la velocidad y temperatura del aire, está proporcionada por el aumento sea de la velocidad del motor o de la potencia eléctrica, definida con el cambio de 0% a 100% de la potencia enviada a la tarjeta de regulación (preactuador del motor) o al variador de potencia eléctrica con control por ángulo de fase (preactuador de la resistencia)

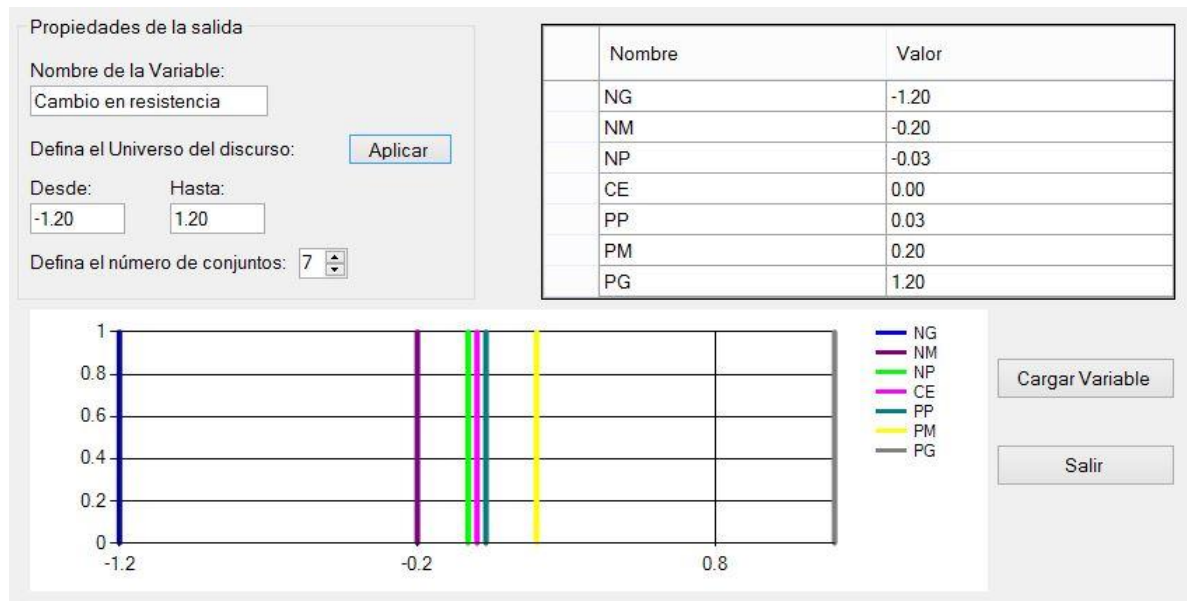
En la variable cambio de motor se emplearon siete (7) conjuntos difusos predeterminadamente definidos de tipo singleton, asignándoles valores como se observa en la figura 72.

Figura 72. Definición de conjuntos de la variable de salida "Cambio el motor".



En la variable cambio de resistencia se emplearon siete (7) conjuntos difusos predeterminadamente definidos de tipo singleton, asignándoles valores como se observa en la figura 73.

Figura 73. Definición de conjuntos de la variable de salida "Cambio en resistencia".



Para definir los nombres de los conjuntos de la variable de salida se preside a la notación descrita en la tabla 9, donde se definen los nombres de los conjuntos de las variables de entrada.

7.2.4 Creación de la base de reglas de control para sistema MIMO.

La base de reglas de control para el modelo Fuzzy determinado se configuró como se observa en la tabla_, constituida a partir del comportamiento observado en la caracterización para la velocidad, independientemente del comportamiento obtenido para la temperatura, partiendo que la base lógica obedece a la estructura “SI – ENTONCES”.

Tabla 10. Estructuración de la base de reglas para el sistema MIMO.

BASE REGLAS		VARIACIÓN DEL ERROR VELOCIDAD					VARIACIÓN DEL ERROR TEMPERATURA				
		NG	NP	CE	PP	PG	NG	NP	CE	PP	PG
ERROR VELOCIDAD	NG	NG	NG	NG	NG	NG	NA	NA	NA	NA	NA
	NM	NG	NG	NM	NM	NM	NA	NA	NA	NA	NA
	NP	NG	NM	NP	NP	NP	NA	NA	NA	NA	NA
	CE	CE	CE	CE	CE	CE	NA	NA	NA	NA	NA
	PP	PP	PP	PP	PM	PG	NA	NA	NA	NA	NA
	PM	PM	PM	PM	PG	PG	NA	NA	NA	NA	NA
	PG	PG	PG	PG	PG	PG	NA	NA	NA	NA	NA
ERROR TEMPERATURA	NG	NA	NA	NA	NA	NA	NG	NG	NG	NG	NG
	NP	NA	NA	NA	NA	NA	NG	NM	NP	PM	PM
	CE	NA	NA	NA	NA	NA	CE	CE	CE	CE	CE
	PP	NA	NA	NA	NA	NA	NM	NM	PP	PM	PG
	PG	NA	NA	NA	NA	NA	PG	PG	PG	PG	PG

La notación determinada para la definición de las reglas obedece a la tomada para los conjuntos de las variables de entrada y los conjuntos de las variables de salida, adicionando un nuevo parámetro como se especifica en la tabla 11.

Tabla 11. Designación adicional de nombre para aplicación de reglas sistema MIMO.

Nombre	Descripción
NA	No aplica

7.2.5 Ejecución del modelo Fuzzy creado para el sistema MIMO mediante el FuzzyControl desarrollado.

Se ejecutó el modelo Fuzzy creado para el control de velocidad y temperatura del aire introduciendo dos (2) valores de referencia combinados como se observa en la tabla 12, con el fin de observar el comportamiento y si es el caso ajustar la base de reglas o sintonizar los conjuntos de entrada y salida.

Tabla 12. Valores de referencia para ejecución de pruebas del sistema MIMO.

Ítem	Velocidad de referencia	Temperatura de referencia
1	6 m/s	40°C
2	7 m/s	32°C

Se editó inicialmente los parámetros de referencia a una velocidad de 6 m/s y una temperatura de 40°C como se observa en la figura 74, se observó que el modelo Fuzzy creado realizaba el proceso de adecuado proceso de control, estabilizando en los valores de referencia o deseados la velocidad y la temperatura.

Figura 74. Edición del primer conjunto de valores de referencia para ejecución de pruebas MIMO.

Control de Velocidad y Temperatura del aire

Velocidad de referencia (m/s):

Temperatura de referencia (°C):

Controles de gráfica de velocidad del aire

Rango de tiempo en segundos:

Desde: Hasta:

Rango del eje Y de Velocidad
Desde: Hasta:

Controles de gráfica Temperatura del aire

Rango de tiempo en segundos:

Desde: Hasta:

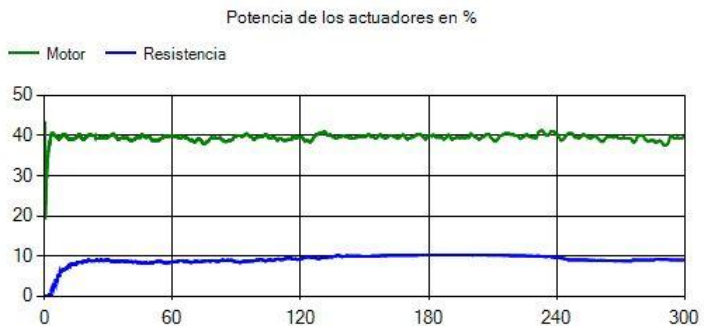
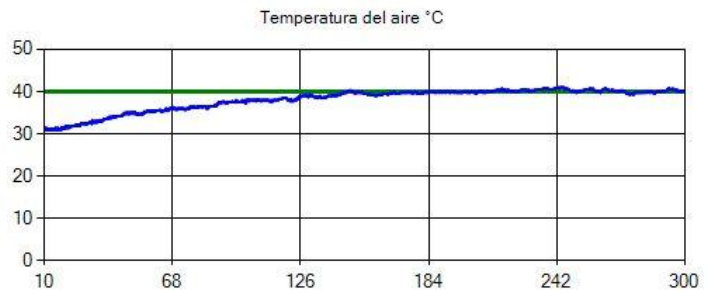
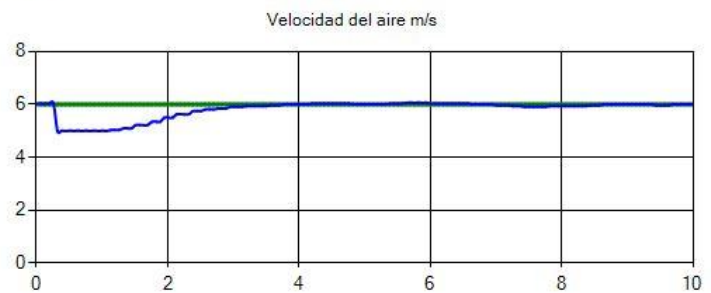
Rango del eje Y de Temperatura
Desde: Hasta:

Controles de gráfica Potencia de actuadores

Rango de tiempo en segundos:

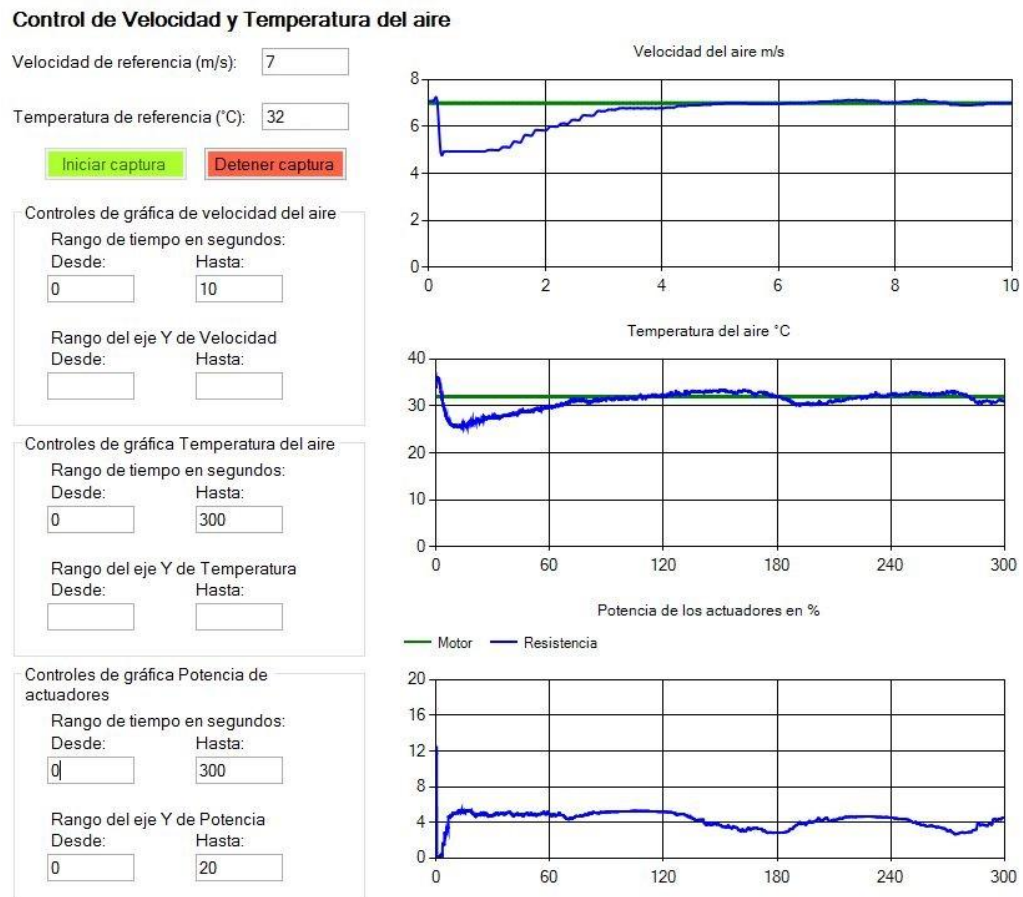
Desde: Hasta:

Rango del eje Y de Potencia
Desde: Hasta:



Posteriormente se asignó un valor de referencia para la velocidad de 7 m/s y la temperatura de 32°C como se observa en la figura 75, el comportamiento observado se asemeja al encontrado anteriormente, estabilizando la velocidad y la temperatura en los valores deseados.

Figura 75. Edición del segundo conjunto de valores de referencia para ejecución de pruebas MIMO.



Mediante las dos pruebas anteriormente realizada se validó la funcionalidad del modelo Fuzzy, Obteniendo como se esperaba, una estabilización adecuada en la velocidad y la temperatura, pero con un tiempo de respuesta muy diferente entre una y la otra.

7.2.6 Análisis de la ejecución del modelo Fuzzy para el sistema MIMO creado mediante el FuzzyControl desarrollado.

De los resultados obtenidos en la ejecución del modelo Fuzzy creado para el sistema de control de velocidad y temperatura del aire, se puede afirmar lo siguiente:

- La caracterización del sistema evidencia que el comportamiento para la velocidad como era de esperarse obedece a un sistema de primer orden igual que en el sistema SISO, sin ningún retardo ni inercias considerables, mientras que para la temperatura se observa claramente como su comportamiento obedece a un sistema de segundo orden, con un alto grado de retardo y una inercia considerable. Esto requiere que el modelo Fuzzy sea riguroso en la implementación del control, teniendo como prioridad el comportamiento de la temperatura.
- La variable de entrada “*erro de temperatura*” se categoriza en cinco (5) conjuntos, a diferencia que la variable de entrada “*error de velocidad*” que se categoriza en siete (7) conjuntos, esto debido a la alta diferencia que existen con los tiempos de respuesta, por lo que no se hace necesario abordar de manera minuciosa la transición de la temperatura, pasando a un concepto más global en este aspecto.
- Las variables de entrada “*variación del error de velocidad*” y “*variación del error de temperatura*” se categoriza en cinco (5) conjuntos, permitiendo identificar de manera más oportuna la celeridad con la que se aproximan a los valores de referencia, sin embargo, en el caso de los conjuntos de la variable “*variación del error de temperatura*” se definen para que esta tenga un valor de pertenencia mucho menor haciendo más prudente la transición de conjunto a conjunto.
- La base de reglas se podría identificar en dos secciones independientes, una para el control de la velocidad y otro para el de la temperatura, esto por la

precaución que se quiso tener ya que el retardo efectuado por el comportamiento de la temperatura generaba una atención prioritaria, por lo mismo las reglas de la sección de temperatura se entrecruzan para contrarrestar el efecto natural que realiza la inercia.

- Observando la respuesta del modelo Fuzzy para el sistema en general, se afirma que en ningún momento se obtuvo “*overshoot*”, esto evidencia la reserva con la que fue diseñado los conjuntos y las reglas del modelo. Sin embargo, es fácil evidenciar que, para la temperatura, entre más alejado se encuentre el valor de referencia, más tiempo se va a tardar en llegar a estabilizarse.
- Por último, se destaca el comportamiento del modelo Fuzzy cuando se ejecutan las pruebas, pues los valores de referencia son constantemente cambiantes y su funcionalidad no se ve afectada por su forma de acercamiento positivo o negativo al valor de referencia deseado para las 2 variables.

8. CONCLUSIONES

- A través del desarrollo e implementación de este controlador Fuzzy se ha puesto a disposición del laboratorio de automatización industrial una herramienta importante en materia de control de procesos y automatización; a partir de estrategias de control basado en la experiencia y conocimientos, enfocada a controladores industriales PLC, permitiendo el desarrollo de competencias y la generación de conocimiento en los estudiantes de la escuela de ingeniería mecánica de la Universidad Industrial de Santander.
- Se demostró la importancia y la capacidad de aplicar controladores Fuzzy a sistemas complejos donde no se tiene un comportamiento preciso, mediante la implementación de modelos de control basados en el conocimiento empírico de su operación, también se comprobó la efectividad en el control de procesos de sistemas multivariables.
- La aplicación de controladores Fuzzy otorga una gran versatilidad tanto para la regulación de sistemas que presentan una respuesta rápida, como para sistemas donde su inercia les impide realizar ajustes en corto tiempo, ya que permite, a través de cambios en su base de conocimientos y sus conjuntos de variables, producir los ajustes de sintonización necesarios para tener un comportamiento más acorde a las necesidades específicas.
- El proceso de defuzzificación a través de pesos concentrados facilita de manera significativa el proceso del cálculo para la salida, reduciendo el costo computacional ejercido por la carga de procesamiento en el PLC, sin observar fallos apreciables en la efectividad del control.

9. RECOMENDACIONES

- Durante la fase de pruebas del control, se consiguió detectar una limitación relacionada con el tipo de licencia del servidor OPC. Si bien el demo utilizado permite cubrir las necesidades inmediatas de una conexión estable, una función de lectura y escritura sobre los datos del PLC, en períodos de trabajo superiores a 2 horas el servidor da por terminada la conexión con la interfaz e impide que el usuario siga manejando el control desde allí, obligándolo a reiniciar el proceso de conexión y operación del mismo. Por tal motivo, la idea de implementar un servidor OPC de licencia completa o una alternativa de comunicación diferente eliminaría esta limitante.
- Conseguir una parametrización desde la interfaz del proceso de normalización garantizaría una menor manipulación del código implementado dentro del PLC. Sin embargo, este proceso obliga al usuario a contemplar todo un nuevo rango de variables que deben relacionarse en la comunicación con el PLC.

BIBLIOGRAFIA

AGUILAR, Carlos y VALENCIA, Jhon. DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR PREDICTIVO BASADO EN PLC, PARA EL BANCO DE PRÁCTICAS DE SISTEMAS DE CONTROL MUESTREADO DEL LABORATORIO DE AUTOMATIZACIÓN INDUSTRIAL., (2017)

Formato de publicaciones IEEEtransactions. Trans-Jour. Doc, [en línea] disponible en <<http://www.ieee.org/organitations/pubs/transactions/stylesheets.htm>>.

KOURO, Samir y MUSALEM, Rodrigo. CONTROL MEDIANTE LÓGICA DIFUSA. Universidad Técnica Federico Santa María. Valparaiso. 2004

MENESES, Jorge., (1998), CONTROL FUZZY BASADO EN PLC'S SIMATIC S5, UNIVERSIDAD INDUSTRIAL DE SANTANDER, BUCARAMANGA, COLOMBIA., Pág 199.

Opc client with C#: an how to video [en línea]. Disponible en internet: <URL: <https://www.mesta-automation.com/opc-client-with-c-an-how-to-video/>>

TARAZONA, Oscar y QUINTERO, Jeisson. DISEÑO Y CONSTRUCCIÓN DE UN OBJETO DE APRENDIZAJE PARA PRÁCTICAS DE SISTEMAS DE CONTROL MUESTREADO, BASADO EN UN CONTROLADOR INDUSTRIAL (PLC). (2012)