

**UNIVERSIDAD POLITÉCNICA SALESIANA**  
**SEDE QUITO**

**CARRERA:**  
**INGENIERÍA ELECTRÓNICA**

**Trabajo de titulación previo a la obtención del título de:**  
**INGENIEROS ELECTRÓNICOS**

**TEMA:**  
**DISEÑO DE UN CONTROLADOR PID DIFUSO PARA LA VARIABLE**  
**CAUDAL, DE LA PLATAFORMA DE ENTRENAMIENTO PARA CONTROL**  
**DE PROCESOS CONTINUOS EXISTENTE EN LA UPS, SEDE QUITO -**  
**CAMPUS SUR**

**AUTORES:**  
**STALIN JAVIER ANDRADE SÁNCHEZ**  
**EDISON PAÚL TIPÁN JIMÉNEZ**

**TUTOR:**  
**WILLIAM MANUEL MONTALVO LÓPEZ**

**Quito, noviembre del 2016**

## CESIÓN DE DERECHOS DE AUTOR

Nosotros Stalin Javier Andrade Sánchez con documento de identificación N° 1720178670 y Edison Paúl Tipán Jiménez con documento de identificación N° 1721055208, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación intitulado: Diseño de un controlador PID difuso para la variable caudal, de la plataforma de entrenamiento para control de procesos continuos existente en la UPS, Sede Quito - Campus Sur, mismo que ha sido desarrollado para optar por el título de: Ingeniero Electrónico, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

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



---

Stalin Javier Andrade Sánchez

C.I 1720178670



---

Edison Paúl Tipán Jiménez

C.I 1721055208

Quito, noviembre del 2016

## DECLARATORIA DE COAUTORÍA DEL DOCENTE TUTOR

Yo, declaro que bajo mi dirección y asesoría fue desarrollado el trabajo de titulación Diseño de un controlador PID difuso para la variable caudal, de la plataforma de entrenamiento para control de procesos continuos existente en la U.P.S, Sede Quito-Campus Sur, realizado por Stalin Javier Andrade Sánchez y Edison Paúl Tipán Jiménez, obteniendo un producto que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana para ser considerados como trabajo final de titulación.

Quito, noviembre del 2016



.....

William Manuel Montalvo López

C.I 1712789989

## **DEDICATORIA**

El presente trabajo de titulación dedico en primer lugar a Dios por haberme permitido llegar a cumplir esta meta, también quiero dedicar a mis padres y hermanos que ellos han sido un apoyo fundamental en el transcurso de esta carrera universitaria.

A mi familia en general, porque siempre con esfuerzo y perseverancia se llega a cumplir sueños que se piensan imposibles

Javier

Primeramente dedico este trabajo de titulación a Dios por darme la bendición tan grande de permitirme llegar a cumplir esta meta, siempre recordando su palabra de esforzarme y ser valiente, también quiero dedicar a mis padres, hermana quienes han sido un apoyo fundamental en el transcurso de mi vida estudiantil.

A mi esposa que siempre ha estado dispuesta a brindarme su apoyo, ayuda y ánimo en todo momento.

Edison

## **AGRADECIMIENTO**

Agradecemos a Dios por habernos bendecido y a la vez a nuestros padres por su apoyo y consejos durante nuestra carrera universitaria.

A la Universidad Politécnica Salesiana por habernos permitido cursar nuestros estudios y cumplir una meta propuesta en nuestras vidas.

Al Ing. William Montalvo sobre todo por su dedicación, apoyo y habernos guiado en la realización del presente trabajo de titulación.

Y por último pero no menos importante agradecer infinitamente a cada uno de los docentes que supieron cultivar su conocimiento y sabiduría en nosotros y así lograr nunca abandonar este hermoso sueño y meta profesional al cual hemos llegado.

## ÍNDICE DE CONTENIDO

CAPÍTULO 1 .....	1
ANTECEDENTES.....	1
1.1 Tema.....	1
1.2 Justificación.....	1
1.3 Delimitación .....	1
1.3.1 Delimitación temporal .....	1
1.3.2 Delimitación espacial.....	2
1.3.3 Delimitación académica.....	2
1.4 Planteamiento del Problema .....	3
1.5 Objetivos .....	3
1.5.1 Objetivo general.....	3
1.5.2 Objetivos Específicos .....	3
1.6 Beneficiarios de la propuesta de intervención.....	4
1.7 Documento Resultante del proyecto.....	4
CAPÍTULO 2 .....	5
FUNDAMENTACIÓN TEÓRICA.....	5
2.1 Sistemas de control.....	5
2.2 Variables de procesos .....	5
2.3 Lazos de control .....	5
2.3.1 Sistema de control en lazo abierto .....	6
2.3.2 Sistema de control en lazo cerrado .....	7
2.4 Controlador Proporcional Integral Derivativo – PID .....	7
2.4.1 Primer método de Ziegler-Nichols. ....	8

2.4.2 Segundo método de Ziegler-Nichols .....	9
2.5 Sensores de Caudal.....	11
2.5.1 Medidores Volumétricos.....	11
2.6 Servo válvulas .....	13
2.7 Controladores Lógicos Programables – PLC .....	13
2.7.1 Introducción al PLC S7-1200.....	13
2.8 Algoritmos de Control Moderno .....	14
2.8.1 Lógica Difusa.....	15
2.9 Controlador PID Difuso .....	16
2.10 Software de Aplicación de Algoritmos Modernos .....	17
2.10.1 Software Matlab.....	17
2.10.2 Toolbox FUZZY .....	18
2.10.3 OPC Toolbox .....	19
2.11 Índice de desempeño de controladores.....	20
2.12 Test estadístico para datos pareados.....	21
<b>CAPÍTULO 3 .....</b>	<b>22</b>
<b>DISEÑO E IMPLEMENTACIÓN DEL PID DIFUSO PARA LA VARIABLE CAUDAL .....</b>	<b>22</b>
3.1 Sensor de caudal Rosemount 1151 .....	22
3.1.1 Calibración del sensor de caudal Rosemount 1151 .....	24
3.2 Diseño del controlador PID difuso .....	25
3.2.1 Controlador PID difuso de la variable caudal.....	25
3.2.2 KEPServerEX .....	25
3.2.3 Programación en Simulink/Matlab .....	26
3.2.4 TIA Portal .....	33

3.3 Implementación del controlador PID difuso .....	33
CAPÍTULO 4.....	46
PRUEBAS Y RESULTADOS .....	46
4.1 Pruebas del controlador PID difuso de caudal .....	46
4.1.1 Respuesta del sistema a la señal del Set Point .....	46
4.2 Pruebas del controlador PID convencional y del PID difuso .....	47
4.2.1 Respuesta de la señal del Set Point en Matlab para el controlador PID convencional .....	48
4.2.2 Respuesta de la señal del Set Point en Matlab para el controlador PID difuso .....	48
4.3 Resultados obtenidos del índice de desempeño IAE entre el PID convencional y el PID difuso.....	49
4.4 Análisis del desempeño de Controladores PID convencionales y PID difusos mediante el error absoluto – IAE .....	50
4.4.1 Test de Wilcoxon del proceso de caudal PID convencional Vs PID difuso....	51
CONCLUSIONES .....	54
RECOMENDACIONES .....	56
REFERENCIAS .....	58
ANEXOS .....	60

## ÍNDICE DE FIGURAS

Figura 1.1. Ubicación de la Universidad Politécnica Salesiana, Sede Quito-Campus Sur. .....	2
Figura 2.1. Elementos de un Sistema de Control en Lazo abierto .....	5
Figura 2.2. Elementos de un Sistema de Control en Lazo cerrado .....	6
Figura 2.3. Diagrama de un control PID de una Planta .....	7
Figura 2.4. Respuesta a un escalón unitario de una planta.....	8
Figura 2.5. Curva de respuesta en forma de S .....	8
Figura 2.6. Diagrama de un controlador proporcional.....	10
Figura 2.7. Oscilación sostenida con periodo $P_{cr}$ .....	10
Figura 2.8. Teorema de Bernoulli .....	12
Figura 2.9. PLC S7-1200 marca Siemens. ....	14
Figura 2.10. Ejemplo de Red Neuronal Unicapa .....	15
Figura 2.11. Relación de un controlador difuso con un lazo de control. ....	16
Figura 2.12. Estructura de un Controlador PID Difuso. ....	16
Figura 2.13. Estructura de un Controlador PID Difuso. ....	17
Figura 2.14. Icono representativo de Matlab .....	18
Figura 2.15. Herramientas de la Lógica Difusa de Matlab. ....	19
Figura 2.16. Ventana de OPC Toolbox de Matlab.....	20
Figura 3.1. Sensor de caudal Rosemount 1151 analógico .....	23
Figura 3.2. Conexión de las puntas del multímetro y los terminales del sensor .....	24
Figura 3.3. Calibración a cero del sensor de Caudal Rosemount 1151.....	24
Figura 3.4. Diagrama de bloques del controlador PID difuso.....	25
Figura 3.5. Pantalla principal del software KEPServerEX .....	26
Figura 3.6. Controlador PID difuso para la variable caudal .....	27
Figura 3.7. Escalamiento de la señal de entrada .....	28
Figura 3.8. Escalamiento y lectura de la señal del sensor.....	28
Figura 3.9. Controlador PID difuso.....	29
Figura 3.10. Bloques de la parte de control del PID convencional.....	29
Figura 3.11. Visualización de valores del Error y del Set Point en el Time Scope .....	30

Figura 3.12. Bloques de la acción de control .....	30
Figura 3.13. Bloques para obtener los valores de la integral del error absoluto (IAE)....	31
Figura 3.14. Pasos para establecer conexión online del PLC .....	34
Figura 3.15. Estado de la conexión entre el PLC Y el KEPServerEX.....	34
Figura 3.16. Pantalla principal del HMI.....	35
Figura 3.17. Pasos para importar el proyecto FUZZY a Matlab.....	36
Figura 3.18. Selección del archivo con extensión fis (.fis) .....	36
Figura 3.19. Pasos para exportar el proyecto FUZZY al Workspace de Matlab .....	37
Figura 3.20. Exportar el proyecto FUZZY al Workspace de Matlab.....	37
Figura 3.21. Pantalla principal del HMI.....	38
Figura 3.22. Interfaz gráfica del controlador PID convencional.....	39
Figura 3.23. Interfaz gráfica del controlador PID FUZZY .....	40
Figura 3.24. Flujo del Proceso del controlador PID.....	41
Figura 3.25. Ejecución fuera de línea del PID convencional y PID FUZZY.....	42
Figura 3.26. Ejecución en tiempo real del controlador PID convencional .....	42
Figura 3.27. Ejecución en tiempo real del controlador PID FUZZY .....	43
Figura 3.28. Comandos para obtener la gráfica comparativa entre los controladores PID convencional y PID FUZZY .....	43
Figura 3.29. Gráfica comparativa entre controladores. PID convencional (curva color azul) y PID FUZZY (curva color rojo). .....	44
Figura 3.30. (a) Valores de PID convencional y (b) Valores de PID FUZZY.....	44
Figura 4.1. Cambios del Set Point.....	46
Figura 4.2. Segunda prueba de cambios del Set Point. ....	47
Figura 4.3. Cambio del Set Point del PID convencional. ....	48
Figura 4.4. Cambio del Set Point del PID difuso.....	49
Figura 4.5. PID convencional (azul) y del PID difuso (rojo).....	50

## ÍNDICE DE TABLAS

Tabla 2.1. Sintonía de Ziegler-Nichols .....	9
Tabla 2.2. Ziegler-Nichols .....	10
Tabla 3.1. Características de trabajo del sensor Rosemount 1151 .....	23
Tabla 3.2. Reglas de inferencia para el control difuso de caudal.....	32
Tabla 3.3. Funciones de la pantalla principal.....	38
Tabla 3.4. Funciones de la interfaz gráfica del controlador PID convencional .....	39
Tabla 3.5. Funciones de la interfaz gráfica del controlador PID FUZZY.....	40
Tabla 3.6. Valores de IAE.....	45
Tabla 4.1. Test de Wilcoxon del Proceso de Caudal.....	51
Tabla 4.2. Resultados del Test de Wilcoxon.....	53

## **RESUMEN**

A través de este proyecto técnico se desea ayudar a los estudiantes y profesores de la Carrera de Ingeniería Electrónica de la Universidad Politécnica Salesiana, Sede Quito-Campus Sur a ampliar los conocimientos adquiridos respecto al uso de controles industriales actuales y al manejo adecuado que se debe tener de equipos usados en las industrias donde sus procesos funcionan mediante instrumentos industriales complejos los cuales necesitan una mejor calibración y un mejor mantenimiento para ampliar su vida útil y que no se detenga el proceso en actividad.

Se crea un control PID (Proporcional-Integrador-Derivativo) difuso para el proceso de caudal, de la Plataforma de Entrenamiento para Control de Procesos Continuos, tomando en cuenta todas las enseñanzas impartidas por los docentes de la carrera de Ingeniería Electrónica, durante el transcurso de la vida estudiantil en la Universidad Politécnica Salesiana Sede Quito-Campus Sur.

Se utiliza el índice de desempeño de la integral del error absoluto (IAE) para comparar el controlador PID difuso para la variable caudal con un controlador PID convencional para la variable caudal, además usando el test de Wilcoxon se determina cuál de estos dos controladores tiene mejor eficiencia de energía, además se efectúa un manual técnico para prácticas con control PID difuso aplicado al control de la variable caudal.

## **ABSTRACT**

Through this technical project wants to help students and teachers the race of Electronic Engineering of the Salesian University, of Quito - South Campus to extend the acquired knowledge regarding the use of current industrial controls and sound management that should be used in industries where their processes work through complex industrial instruments which need better calibration and better maintenance to extend its life equipment and that the process does not stop in activity.

A FUZZY PID control (Proportional-Integrative-Derivative) for the process flow, Platform Training Control Continuous Process is created, taking into account all the lessons taught by teachers of the career of Electronic Engineering, during the course of student life at the Polytechnic Salesian University of Quito-South Campus.

The performance index of the integral absolute error (IAE) is used to compare the FUZZY PID controller for flow variable with a conventional PID controller for flow variable, also using the Wilcoxon test determined which of these two drivers better have energy efficiency, plus a technical manual is made for FUZZY PID control practices applied to the flow control variable.

## INTRODUCCIÓN

El diseño de un controlador PID (Proporcional-Integrador-Derivativo) difuso real a nivel industrial para el proceso de caudal ayudará al alumnado de la Carrera de Ingeniería Electrónica de la Universidad Politécnica Salesiana, Sede Quito-Campus Sur a ampliar sus conocimientos en el manejo de equipos de procesos industriales.

El proyecto se organizó de la siguiente forma:

El capítulo 1, describe el tema del proyecto realizado, planteamiento del problema, justificación, objetivo general, objetivos específicos y los beneficiados del proyecto.

El capítulo 2, detalla el marco teórico, principales generalidades, conceptos, características de los elementos y dispositivos a usarse en este proyecto.

El capítulo 3, muestra el diseño de un controlador PID difuso, como también el diseño de un controlador PID convencional para la variable de caudal, de una plataforma de entrenamiento para control de procesos continuos.

El capítulo 4, presenta las pruebas de funcionamiento del control PID difuso, como también el de un controlador PID convencional para la variable de caudal de una plataforma de entrenamiento para control de procesos continuos implementado en Matlab, además de las pruebas de comparación entre ellas mediante el índice de desempeño IAE usando el Test de Wilcoxon y se describe las conclusiones y recomendaciones del proyecto.

# CAPÍTULO 1

## ANTECEDENTES

### 1.1 Tema

Diseño de un controlador PID difuso para la variable caudal, de la plataforma de entrenamiento para control de procesos continuos existente en la UPS, Sede Quito-Campus Sur.

### 1.2 Justificación

Los sistemas de control de procesos continuos industriales actuales están basados en controladores PID, siendo uno de los más usados, ya que el ajuste de parámetros no es tan complejo.

La variable que se va a controlar y monitorear es usada muy comúnmente en diferentes áreas de la industria ecuatoriana como son procesos de manufactura de alimentos, medicina, textiles, etc. Y puesto que los estudiantes de la Carrera de Ingeniería Electrónica de la Universidad Politécnica Salesiana, Sede Quito-Campus Sur tienen un bajo conocimiento en el manejo de equipos de procesos industriales para realizar control difuso de caudal, se desarrollará este proyecto con el fin de capacitar al alumnado en un proceso real utilizando técnicas de control moderno.

El control PID difuso de variable de caudal repotenciará a la plataforma de entrenamiento para control de procesos continuos existente en la Universidad Politécnica Salesiana, Sede Quito-Campus Sur, en el laboratorio de Teoría de Control, mejorando el desarrollo profesional de los estudiantes de la Carrera de Ingeniería Electrónica.

### 1.3 Delimitación

#### 1.3.1 Delimitación temporal

El presente proyecto tendrá la duración de 6 meses a partir de la aprobación del plan.

### 1.3.2 Delimitación espacial

Este proyecto se llevará a cabo en la ciudad de Quito, en la Universidad Politécnica Salesiana, sede Quito-Campus Sur para el laboratorio de Teoría de Control de la Carrera de Ingeniería en Electrónica.

Mapa por GPS de la ubicación de la U.P.S

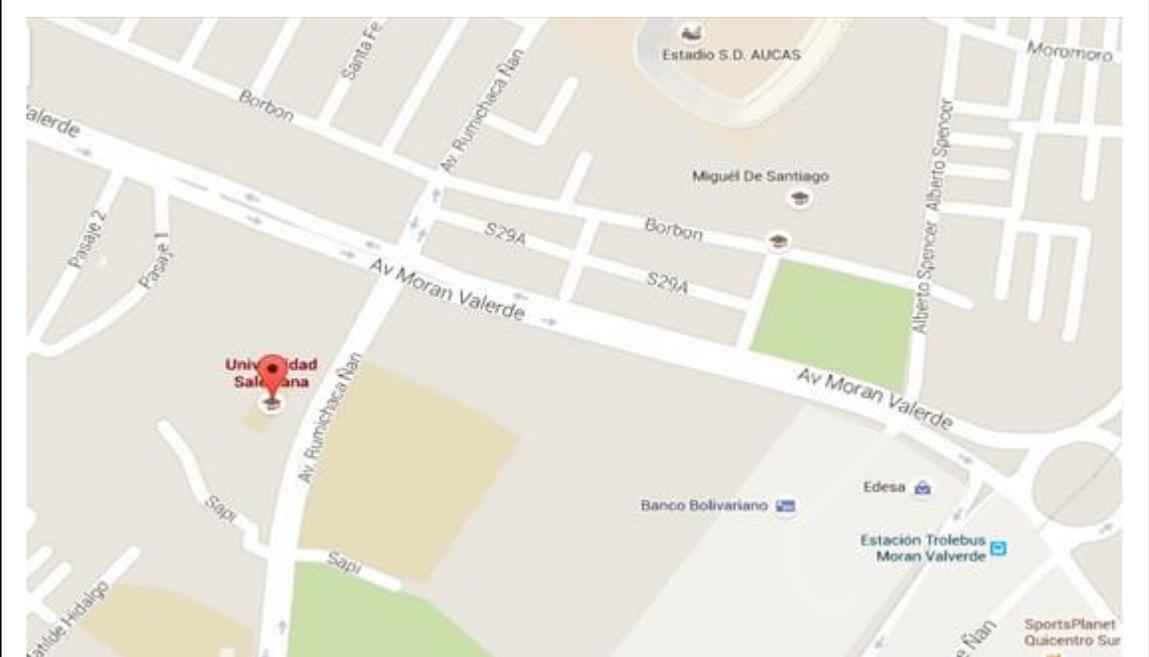


Figura 1.1. Ubicación de la Universidad Politécnica Salesiana, Sede Quito-Campus Sur.

Fuente: Google Maps

### 1.3.3 Delimitación académica

En el presente proyecto se desarrollará un control PID difuso para la variable de caudal de la plataforma de entrenamiento para el control de procesos continuos usando el software Matlab, además se usará el PLC Siemens S7-1200 el cual será el encargado de la adquisición de datos, usando equipos de instrumentación y automatización industrial, además de poner en práctica lo aprendido en la carrera de Ingeniería Electrónica.

## **1.4 Planteamiento del Problema**

En la actualidad los procesos y la electrónica que se utilizan en la industria comprometen tecnología moderna. Estos procesos incorporan sistemas de control convencionales los cuales cumplen con el objetivo previsto, pero son difíciles de diseñar y ajustar puesto que el modelamiento matemático de los mismos se torna complejo. Es por eso que en este proyecto se propone un controlador inteligente basado en lógica difusa y que su comportamiento este basado en un PID convencional.

La Universidad Politécnica Salesiana cuenta con una plataforma de entrenamiento de procesos continuos, la cual no tiene implementado un sistema de control para la variable caudal, ni un sistema de control para el mismo. Este es uno de los procesos más característicos en el ámbito industrial, se necesita entonces mejorar la mencionada plataforma para que se implemente y controle este proceso, y que los estudiantes puedan adiestrarse en todos los ámbitos académicos y prácticos lo más semejantes a la realidad.

A la vez los estudiantes de la Carrera de Ingeniería Electrónica de la Universidad Politécnica Salesiana, Sede Quito-Campus Sur, tienen poca práctica con el uso de controladores aplicados sobre equipos de procesos industriales; por tal motivo se desarrolla en este proyecto, un control PID (Proporcional-Integrador-Derivativo) difuso real a nivel industrial para regular caudal.

## **1.5 Objetivos**

### **1.5.1 Objetivo general**

Diseñar un control PID difuso de la variable de caudal de la plataforma de entrenamiento para el control de procesos continuos.

### **1.5.2 Objetivos Específicos**

- Diseñar e implementar el algoritmo de control PID difuso en el software Matlab para control de procesos de caudal.

- Comunicar el PLC Siemens S7-1200 con el software Matlab mediante el protocolo de comunicación Ethernet.
- Comparar el desempeño del controlador PID convencional con el control PID difuso.
- Elaborar un manual técnico para prácticas con control PID difuso de la variable caudal, en la plataforma de entrenamiento de procesos continuos

### **1.6 Beneficiarios de la propuesta de intervención**

El presente proyecto beneficiará a los docentes y estudiantes de la carrera de Ingeniería Electrónica de la Universidad Politécnica Salesiana, Sede Quito-Campus Sur, puesto que, con el mejoramiento de la Plataforma de entrenamiento para el control de procesos continuos, se podrá adquirir más conocimientos acerca de los procesos industriales que tienen que ver con la variable caudal.

Con esta mejora los estudiantes tendrán la capacidad de realizar prácticas de laboratorio sobre el control de PID difuso orientado a la variable caudal, además de las variables ya implementadas con anterioridad que son la de nivel y temperatura.

### **1.7 Documento Resultante del proyecto**

Como resultado de este proyecto técnico, se propone publicar un artículo académico en la revista INGENIUS de la UPS, con tema “Diseño de un controlador PID difuso para la variable caudal, de la plataforma de entrenamiento para control de procesos continuos”, que pueda ser utilizado como aporte en la rama del control automático.

## CAPÍTULO 2

### FUNDAMENTACIÓN TEÓRICA

En el presente capítulo se describe el marco teórico, principales generalidades, conceptos fundamentales, características de elementos a usar, así como también de dispositivos a emplear en el presente proyecto, utilizando literatura especializada.

#### 2.1 Sistemas de control

El principal objetivo de un sistema de control es el de dirigir la respuesta de una planta, de una manera automática, el operador no interviene directamente sobre la salida. El operador manipula las magnitudes de consigna y el sistema de control es el encargado de dirigir la salida por medio de accionamientos. (Ogata, 2010).

#### 2.2 Variables de procesos

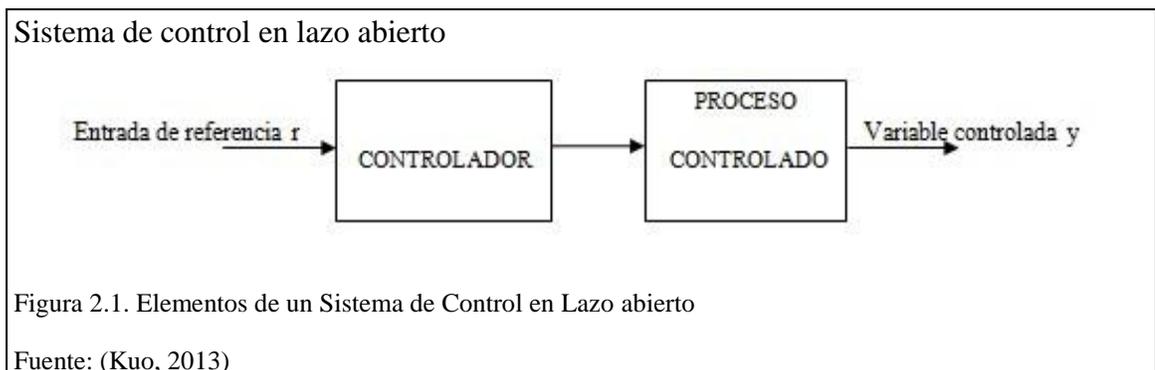
Existen diferentes tipos de variables de procesos que intervienen en un proceso a nivel industrial, los cuales se puede citar: nivel, presión, caudal y temperatura. (Ogata, 2010).

En el presente proyecto se usa la variable caudal en el diseño de un controlador PID difuso.

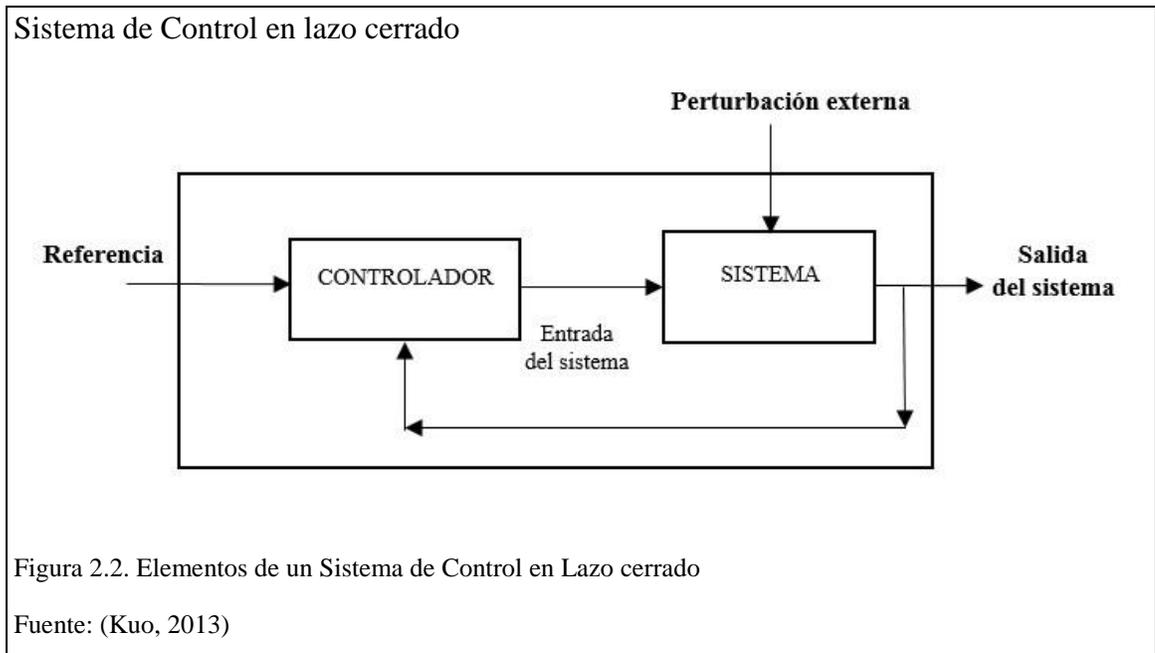
#### 2.3 Lazos de control

Existen dos tipos de lazos de control (Ogata, 2010):

- Lazo de control abierto, como se muestra en la Figura 2.1.



- Lazo de control cerrado, como se muestra en la Figura 2.2.



### 2.3.1 Sistema de control en lazo abierto

Un sistema de control en lazo abierto se da, cuando no tiene realimentación, es decir la entrada no tiene información de la salida. Existen ciertas características de los sistemas de control en lazo abierto como son (Ogata, 2010):

- Cuando existen disturbios internos y externos en la planta, este tipo de sistema no ejecuta el proceso deseado.
- Este tipo de sistema es usado cuando se tiene conocimiento de la relación de entrada y salida.

Algunas ventajas en los sistemas de control en lazo abierto como son:

- Construcción simple y facilidad de mantenimiento.
- Menos costosos que el correspondiente sistema en lazo cerrado.
- Convenientes cuando la salida es difícil de medir o cuando medir la salida de manera precisa no es económicamente viable.

### 2.3.2 Sistema de control en lazo cerrado

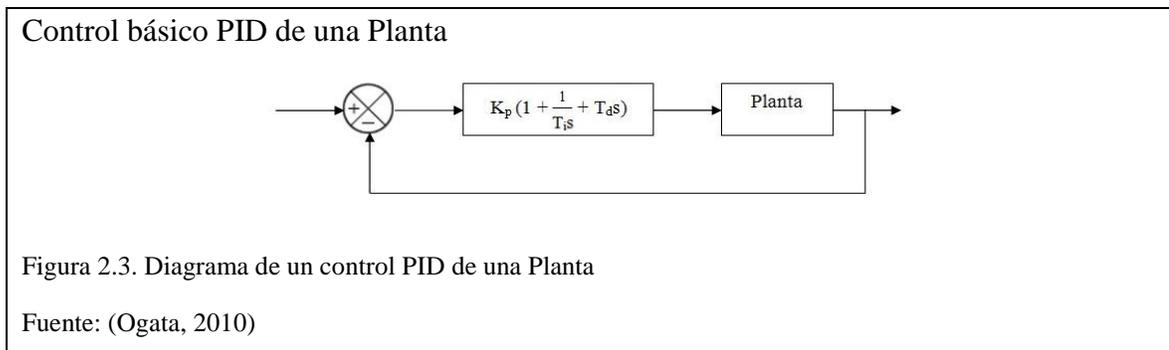
En este tipo de sistemas se produce una realimentación debido a que la entrada recibe información de la salida del sistema.

En el proyecto se usó este tipo de sistema de control para el proceso industrial deseado. Existen ciertas características de los sistemas de control en lazo cerrado como son: (Ogata, 2010).

- En este tipo de sistemas, para tener un valor requerido se hace una diferencia de la señal que ingresa al sistema con la señal realimentada.
- Se usa un control de realimentación para llevar al mínimo el error del proceso.

### 2.4 Controlador Proporcional Integral Derivativo – PID

En la Figura 2.3, se muestra la estructura del control PID de una planta.



Son los más usados en plantas industriales debido a su tipo de estructura la cual es simple.

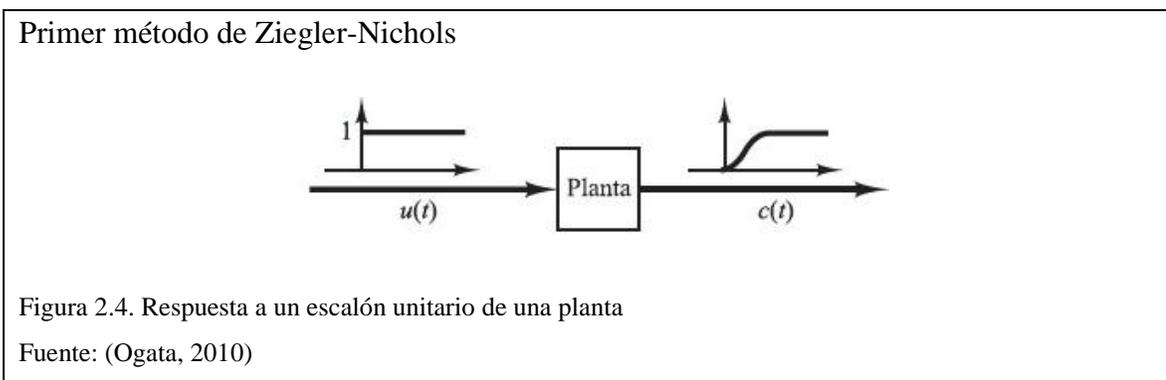
Si la planta es tan complicada que no se hace posible obtener su modelo matemático, se debe recurrir a la sintonía de los controladores PID. Existen reglas de sintonización para controladores PID, según Ziegler y Nichols significa dar valores a  $K_p$ ,  $T_i$  y  $T_d$ , en las plantas industriales se utilizan estas reglas cuando no se conocen sus modelos matemáticos. (Ogata, 2010).

Las reglas de sintonía de Ziegler-Nichols dan una estimación razonable de los parámetros del controlador y proporcionan un punto de partida para una sintonía fina, en lugar de dar los parámetros  $K_p$ ,  $T_i$  y  $T_d$  en un único intento. (Ogata, 2010).

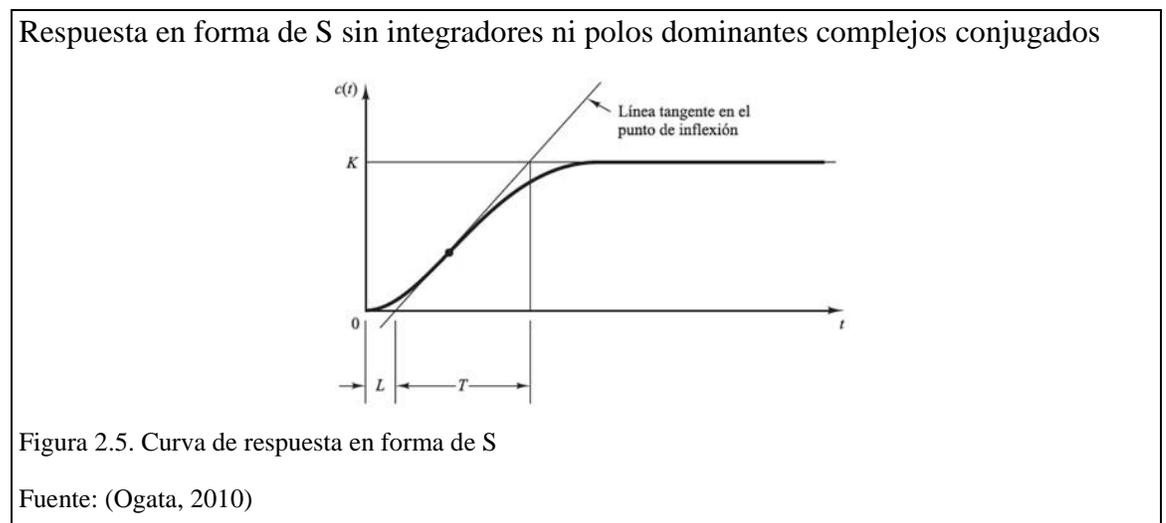
Hay dos métodos denominados reglas de sintonía de Ziegler-Nichols, a continuación se hace una breve presentación de estos dos métodos.

#### 2.4.1 Primer método de Ziegler-Nichols.

En este método la entrada a la planta es un escalón unitario, la cual se obtiene experimentalmente, tal como se muestra en la Figura 2.4. Se puede aplicar si la respuesta muestra una curva con forma de S. (Ogata, 2010).



En la Figura 2.5, se observa la respuesta escalón unitario que tiene forma de S.



En la Figura 2.5 se observa el gráfico donde la L es el tiempo de retardo, T es la constante de tiempo. (Ogata, 2010).

A continuación en la Tabla 2.1, se muestra los valores de las constantes de Ziegler-Nichols por cada tipo de controlador usando el método uno.

Tabla 2.1.

Sintonía de Ziegler-Nichols

Tipo de controlador	$K_p$	$T_i$	$T_d$
<b>P</b>	$\frac{T}{L}$	$\infty$	0
<b>PI</b>	$0.9 \frac{T}{L}$	$\frac{L}{0.3}$	0
<b>PID</b>	$1.2 \frac{T}{L}$	2L	0.5L

Nota: Regla de sintonía de Ziegler-Nichols basada en la respuesta escalón de la planta (primer método).

Fuente: (Ogata, 2010)

Usando el primer método de Ziegler-Nichols se obtiene lo siguiente: (Ogata, 2010)

$$G_c(s) = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right) \quad (2.1)$$

$$G_c(s) = 1.2 \frac{T}{L} \left( 1 + \frac{1}{2LS} + 0.5LS \right) \quad (2.2)$$

$$G_c(s) = 0.6T \frac{(s + \frac{1}{L})^2}{s} \quad (2.3)$$

#### 2.4.2 Segundo método de Ziegler-Nichols

En la Figura 2.6 se observa el diagrama de un controlador proporcional, en donde se tiene a la salida una respuesta constante. (Ogata, 2010).

Controlador Proporcional

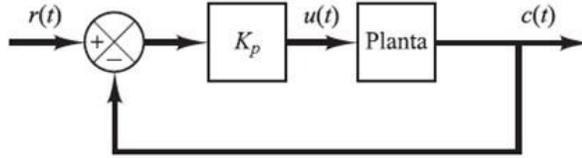


Figura 2.6. Diagrama de un controlador proporcional

Fuente: (Ogata, 2010)

En la Figura 2.7 se muestra un gráfico donde  $K_{cr}$  es la ganancia crítica y  $P_{cr}$  es su período. (Ogata, 2010)

Curva con periodo  $P_{cr}$

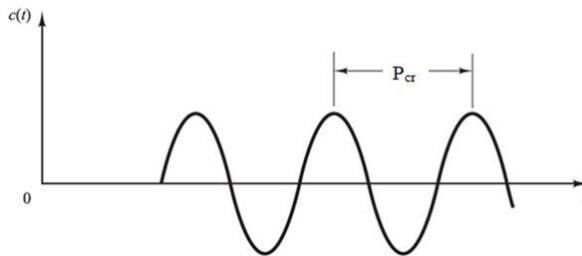


Figura 2.7. Oscilación sostenida con periodo  $P_{cr}$ .

Fuente: (Ogata, 2010)

A continuación en la Tabla 2.2, se muestra los valores de las constantes de Ziegler-Nichols por cada tipo de controlador usando el método dos.

Tabla 2.2.

Ziegler-Nichols

Tipo de Controlador	$K_p$	$T_i$	$T_d$
<b>P</b>	$0.5 K_{cr}$	$\infty$	0
<b>PI</b>	$0.45 K_{cr}$	$\frac{1}{1.2} P_{cr}$	0
<b>PID</b>	$0.6 K_{cr}$	$0.5 P_{cr}$	$0.125 P_{cr}$

Nota: Tipos de controladores por regla de Ziegler-Nichols.

Fuente: (Ogata, 2010)

Usando el segundo método de Ziegler-Nichols se obtiene lo siguiente:

$$G_c(S) = K_p \left( 1 + \frac{1}{T_i S} + T_d S \right) \quad (2.4)$$

$$G_c(S) = 0.6 K_{cr} \left( 1 + \frac{1}{0.5 P_{cr} S} + 0.125 P_{cr} S \right) \quad (2.5)$$

$$G_c(S) = 0.075 K_{cr} P_{cr} \frac{\left( S + \frac{4}{P_{cr}} \right)^2}{S} \quad (2.6)$$

Cuando no se conoce el funcionamiento total de la planta se puede aplicar las reglas de Ziegler-Nichols caso contrario se debe utilizar otros métodos. (Ogata, 2010).

## **2.5 Sensores de Caudal**

Existen dos tipos de medidores o sensores de caudal, los sensores volumétricos se usan para realizar medidas básicas en volumen, mientras que los de masa se usan cuando se requiere medidas exactas en caudal masa. (Creus, 2011)

En este proyecto se usa los medidores volumétricos de presión diferencial debido a que pueden utilizarse para la mayoría de los fluidos, además no son caros, particularmente si se instalan en grandes tuberías y se comparan con otros existentes en el mercado.

### **2.5.1 Medidores Volumétricos**

Funcionan de dos maneras: la primera por desplazamiento y la segunda por deducción. La medida de caudal volumétrico a nivel industrial se lleva a cabo en especial con elementos que dan lugar a una presión diferencial al paso del fluido. (Creus, 2011).

En el presente proyecto se usa un sensor por presión diferencial que está dentro de los medidores por deducción.

#### **2.5.1.1 Medidores de presión diferencial**

Estos medidores funcionan mediante una diferencia de presiones cuando la tubería sufre una contracción. Mediante dos tomas ubicadas en la entrada y en la salida, o por una toma de presión cerca se obtiene la presión diferencial del fluido. La fórmula de caudal tiene como sustento el teorema de Bernoulli. (Creus, 2011).

A continuación se detalla las ecuaciones de este teorema:

$$\frac{V^2_a}{2} + \frac{P_a}{\rho_0} = \frac{V^2_c}{2} + \frac{P_c}{\rho_0} \quad (2.7)$$

$$S_a V_a = S_c V_c \quad (2.8)$$

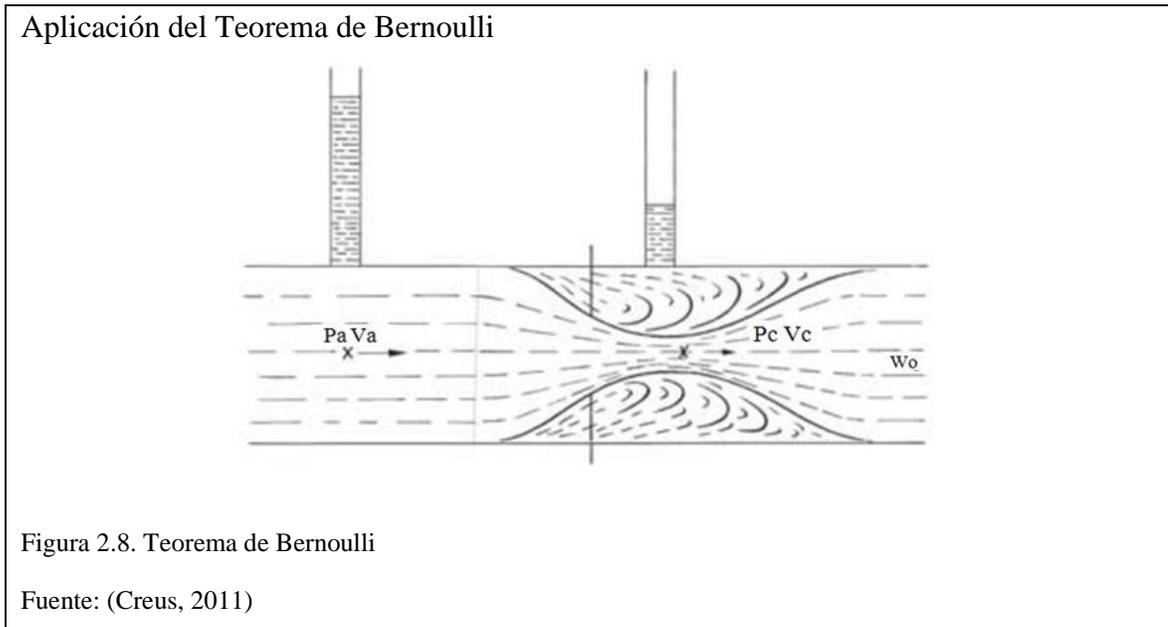
Dónde:  $\rho_0$ : la densidad del fluido.

$P_a, P_c$ : presiones absolutas.

$V_a, V_c$ : velocidades iniciales.

$S_a, S_c$ : secciones intervenidas

En la Figura 2.8, se puede observar la aplicación del Teorema de Bernoulli.



El caudal volumétrico se puede expresar en función de la presión diferencial creada por un elemento medidor de dicha presión como:

$$Q_v = K \sqrt{H} \quad (2.9)$$

Donde la que  $H$  es la diferencia de alturas de presión del fluido y  $K$  es una constante empírica. (Creus, 2011).

## **2.6 Servo válvulas**

Las servoválvulas son accionadores de tipo neumático o hidráulico que conectan dos o más vías por las que circula un fluido.

Las servoválvulas tienen la posibilidad de controlar la presión o el caudal. Las servoválvulas disponen de una retroalimentación interna que cumplen la función de regulación, de esta forma se obtiene en los circuitos hidráulicos que las utilizan un alto grado de exactitud, superior que con las válvulas proporcionales y muy superior que con las convencionales. (Creus, 2011).

## **2.7 Controladores Lógicos Programables – PLC**

La Asociación Nacional de Manufactura Eléctrica - NEMA, define un autómata programable como un aparato electrónico digital que usa una memoria programable para el almacenamiento de instrucciones que implementan funciones lógicas, secuenciales, temporizadores, contadores y aritméticas, para controlar a través de módulos de entradas/salidas digitales y analógicas diferentes tipos de máquinas o procesos. Los autómatas programables se denominan asimismo Controladores Lógicos Programables o PLC (Programmable Controller Logic).

El PLC satisface necesidades de procesos continuos y discontinuos, los que son completamente discretos pueden controlar solo variables discretas y los que son discretos-analógicos a más de regular variables discretas, también pueden regular variables analógicas tales como temperatura, presión, nivel y en el caso del presente proyecto se controlara la variable de caudal.

### **2.7.1 Introducción al PLC S7-1200**

Las principales características que menciona Siemens acerca de su controlador S7-1200 son:

- Controlador modular y compacto
- Versátil que lo hace una inversión segura
- Encaja en una amplia variedad de aplicaciones

- Tiene un diseño escalable y flexible
- Tiene una interfaz de comunicación industrial

La gama S7-1200 incorpora 4 CPU diferentes que ofrecen una variedad de rendimiento y opciones de entrada y salida, cada CPU está disponible con una opción de tensión de alimentación AC o DC y AC, DC, o salidas de relé.

El S7-1200 ofrece varios tipos de comunicación entre los dispositivos entre CPU y de programación, HMI y otros CPU. En la Figura 2.9 se observa este controlador que cuenta con comunicación Profinet que se utiliza para el intercambio de datos a través del programa de usuario con otros compañeros de comunicación a través de Ethernet, también cuenta con comunicación Profibus a diferencia que esta se conecta a través de una red Profibus y no Ethernet como la anterior, por tal motivo la CPU incorpora un puerto PROFINET para la comunicación en la red Profinet.

Módulo de CPU del PLC S7-1200



Figura 2.9. PLC S7-1200 marca Siemens.

Fuente: (Siemens, 2016)

## 2.8 Algoritmos de Control Moderno

En la actualidad existen controles avanzados que cuentan con sistemas muy precisos, donde la inteligencia artificial tiene un protagonismo muy interesante, en estos tipos de sistemas, los más usados son la lógica difusa, las redes neuronales, sistemas neuro-difusos y los algoritmos genéticos. En la Figura 2.10, se describe la estructura de una red neuronal Unicapa.

## Estructura de Red Neuronal

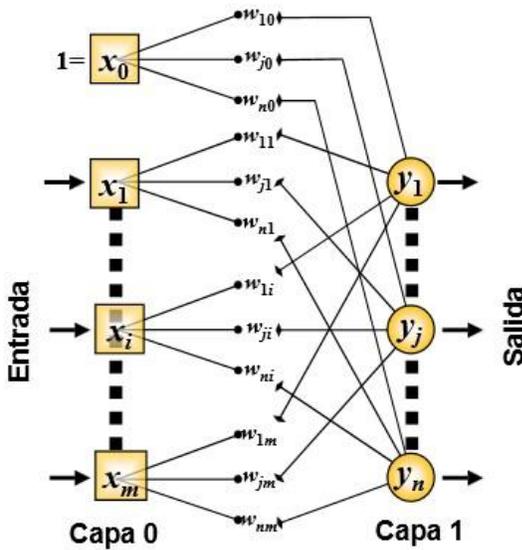


Figura 2.10. Ejemplo de Red Neuronal Unicapa

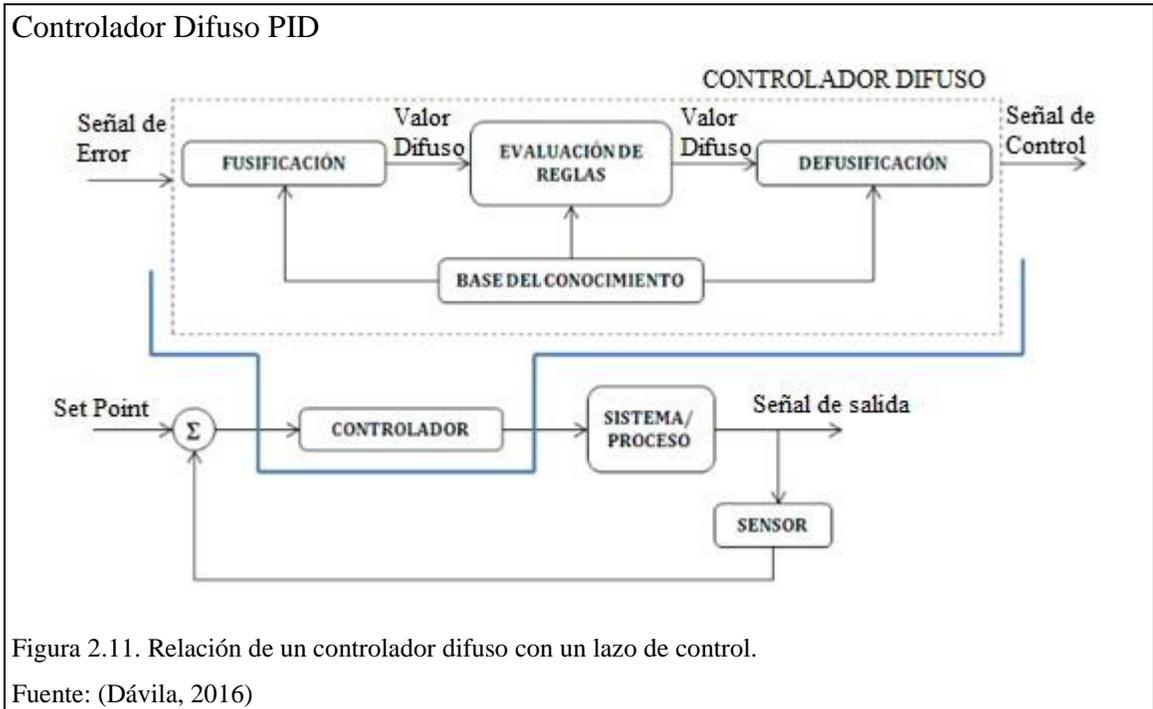
Fuente: (Izaurieta-Saavedra, 2016)

### 2.8.1 Lógica Difusa

Los controles que se encuentran basados en lógica difusa emulan el razonamiento y pensamiento del cerebro humano, las aplicaciones de la lógica difusa se encuentran en múltiples áreas como por ejemplo el desarrollo tecnológico de los electrodomésticos, también se puede encontrar en programas computacionales para tomar decisiones correctas en el momento indicado, y con ello se han extendido a diversos campos específicos como las cámaras de video, seguridad en base al monitoreo de terremotos.

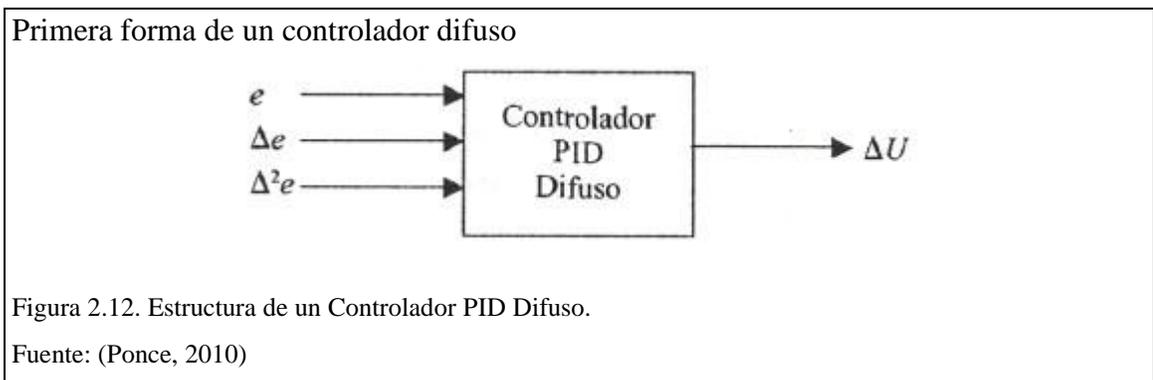
La lógica difusa también conocida como borrosa, es una rama de la Inteligencia Artificial, en la cual las proposiciones pueden ser representadas con diferentes grados entre lo verdadero o lo falso. Este tipo de lógica se basa en reglas basadas en el conocimiento humano, por lo que tolera errores e incluso puede analizar estos para la resolución de problemas o toma de acciones en un determinado sistema de control. (Castillo, 2016).

La Figura 2.11, bosqueja el esquema de la relación de un controlador difuso implementado en un lazo de control.

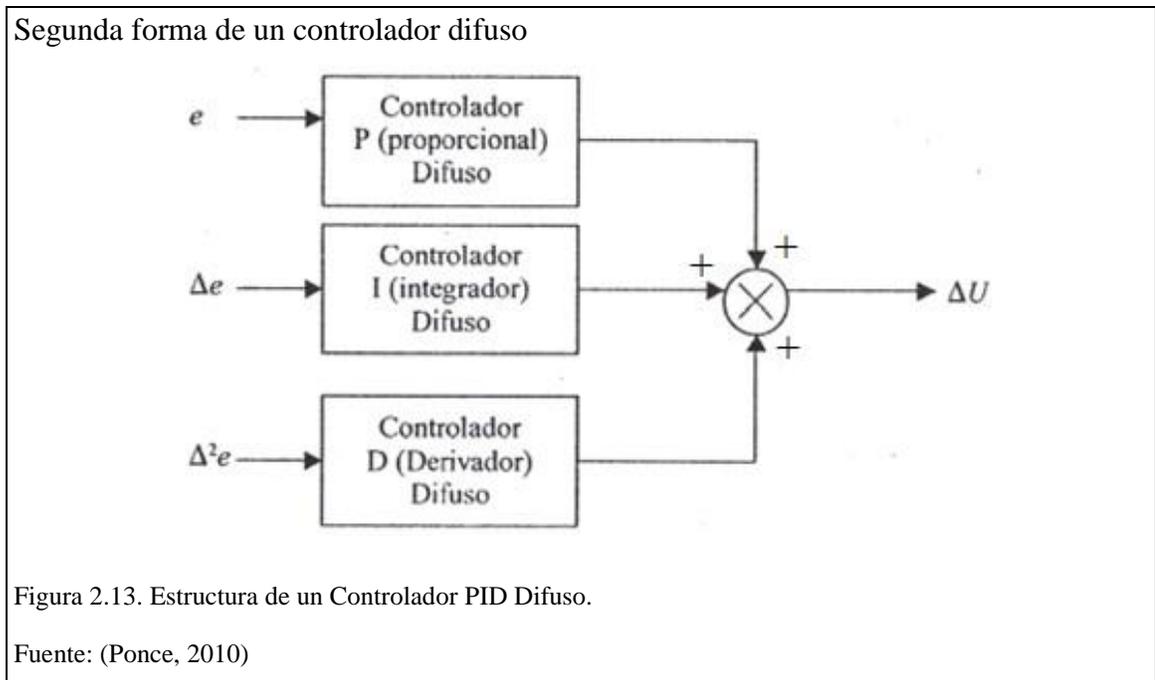


## 2.9 Controlador PID Difuso

Existen dos maneras de realizar un controlador PID difuso, la primera forma es colocando en un solo bloque las variables, error, la derivada del error y la integral del error, en la Figura 2.12, se observa la primera forma mediante un diagrama de bloques. (Ponce, 2010).



La segunda forma es creando cada variable por separado, sumando los bloques y obtener así el PID, en la Figura 2.13 se observa la segunda forma.



## 2.10 Software de Aplicación de Algoritmos Modernos

Con el desarrollo de la tecnología, se ha creado una variedad de software de aplicación para la resolución de algoritmos modernos que en esencia se basan en el manejo numérico de la información, entre los cuales se destaca el software Matlab el cual se basa en programación por matrices, en el presente proyecto se usará el software ya mencionado porque maneja programación de alto nivel para cálculo técnico, gráficos de funciones en 2D y 3D, y además posee herramientas para la creación de interfaces gráficas personalizadas para el usuario.

### 2.10.1 Software Matlab

En la Figura 2.14, se muestra el logo de la empresa del software Matlab.

Logotipo del software Matlab

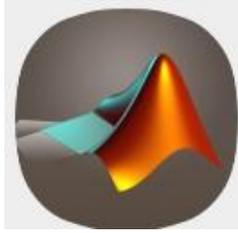


Figura 2.14. Icono representativo de Matlab

Fuente: (Mathworks, 2016)

Reyes en su libro titulado Matlab Aplicado a Robótica y Mecatrónica describe que Matlab es un lenguaje de programación de alto nivel integrado con entorno gráfico amigable para el usuario, tanto como para la visualización de datos, funciones, gráficas 2D y 3D, procesamiento de imágenes, video, computación numérica para desarrollar algoritmos matemáticos con aplicaciones de ingeniería y ciencias exactas.

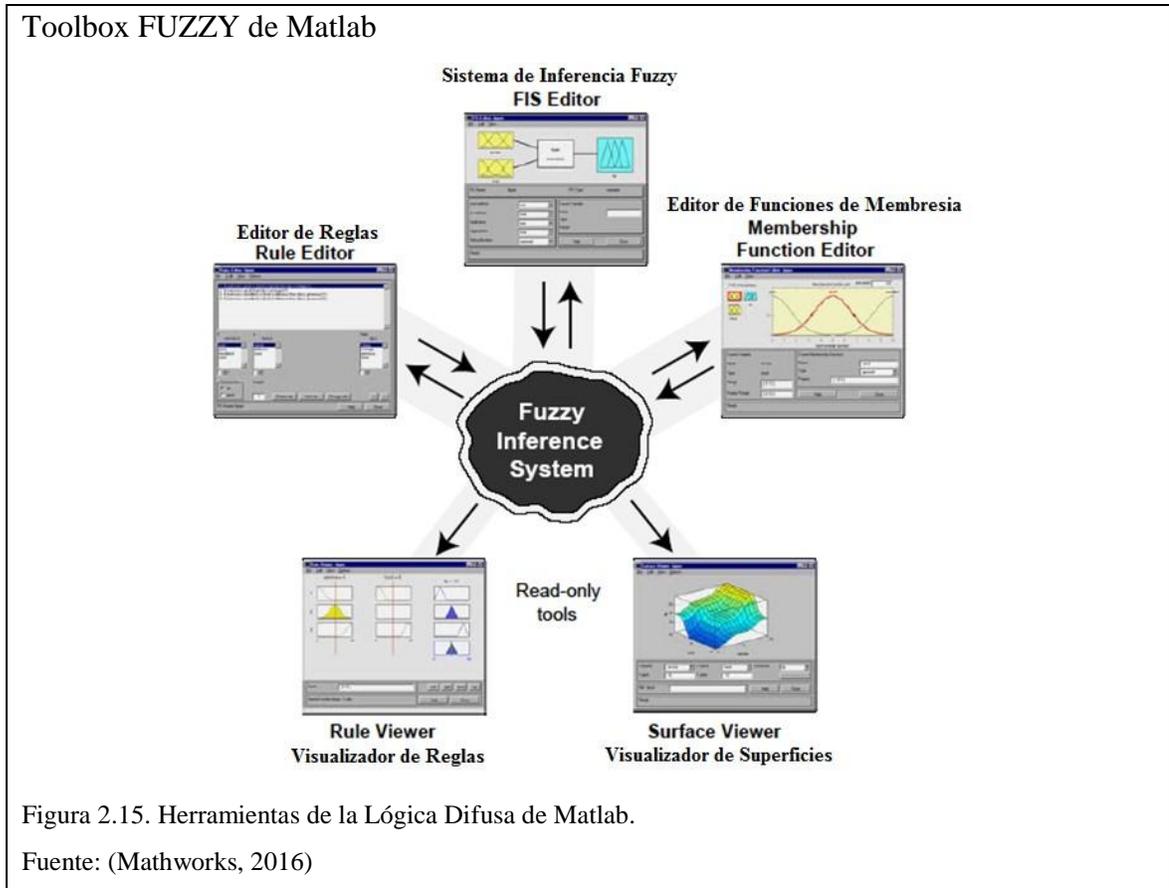
El software Matlab contiene herramientas dedicadas al control industrial moderno, como su toolbox de Lógica Difusa (FUZZY). (Reyes, 2012).

### **2.10.2 Toolbox FUZZY**

En la Figura 2.15, se observa el Toolbox FUZZY el cual tiene aplicaciones y funciones para el análisis y diseño de sistemas basados en lógica difusa; este Toolbox guía paso a paso a través del proceso de diseño de los sistemas de inferencia difusos. (Para una mayor información se recomienda visitar el siguiente enlace: <http://www.mathworks.com/products/FUZZY-logic/>).

Al diseñar estos sistemas difusos es más fácil construir un sistema gráficamente, que trabajar utilizando únicamente la caja de herramientas de la lógica difusa mediante la línea de comandos. Existen cinco herramientas primarias del Toolboz FUZZY las cuales son:

- Editor de reglas
- Editor de FIS (Sistema de Inferencia FUZZY)
- Editor de funciones membrecía
- Visualizador de reglas
- Visualizador de las superficies

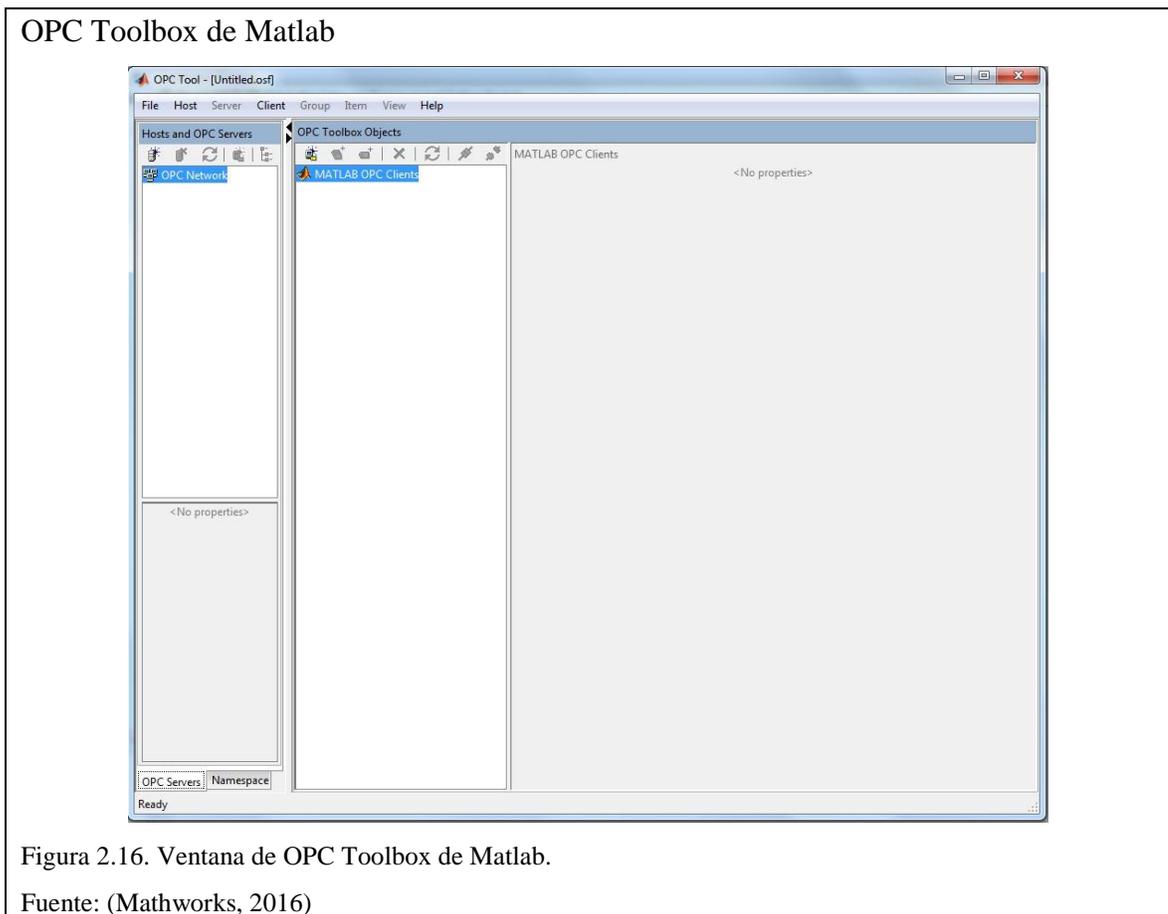


### 2.10.3 OPC Toolbox

La herramienta OLE for Process Control (OPC) permite una conexión con los servidores OPC DA y OPC HDA, los cuales le dan acceso a transferir datos históricos OPC desde Matlab y Simulink; con esta herramienta se puede leer, escribir y registrar datos OPC de dispositivos como los sistemas de control distribuido, supervisión y adquisición de datos y PLC, siempre y cuando cumplan con el estándar de comunicación OPC Foundation

Data Access. (Para una mayor información se recomienda visitar el siguiente enlace: <http://www.mathworks.com/products/opc/>).

La Figura 2.16, muestra la ventana principal de Herramientas de OPC de Matlab, en el lado izquierdo se tiene una pestaña de Hosts y OPC Servers y al lado derecho se tiene una pestaña de Herramientas de objetos de OPC (OPC Toolbox Objects) con sus respectivas configuraciones.



## 2.11 Índice de desempeño de controladores

Los índices de desempeño que regularmente se usan para valorar un controlador son:

- Máximo sobre paso ( $M_p$ ): Es el valor pico máximo de la curva de respuesta. Este valor no debe sobrepasar el nivel determinado debido a que puede causar la pérdida de material en el proceso.

- Tiempo de establecimiento ( $t_s$ ): Es el tiempo que se necesita para que la curva de respuesta llegue a un rango cerca, no al valor final.
- Tiempo de respuesta ( $t_r$ ): Es el tiempo que se necesita para que la respuesta pase del 10 al 90%, del 5 al 95% o del 0 al 100% del valor final.
- Error en estado estable ( $E_{ss}$ ): Es la diferencia entre el punto de ajuste SP y la variable de proceso PV.
- Integral del error absoluto (IAE): Es la energía perdida en el control de un proceso. Cuando se obtiene un valor bajo de este índice se deduce que se la variable de proceso estuvo muy cerca del valor deseado.

La IAE matemáticamente se determina por las siguientes ecuaciones:

$$IAE = \int_0^{\infty} |e(t)| dt \quad (2.11)$$

$$e(t) = r(t) - y(t) \quad (2.12)$$

Dónde:  $y(t)$  es la salida del sistema de control y  $r(t)$  es el valor esperado.

## 2.12 Test estadístico para datos pareados

Los tests estadísticos son pruebas matemáticas que se usan en las estadísticas para poder determinar su grado de certeza.

Uno de los test estadísticos más usados es el Test de Wilcoxon, el cual es usado en este proyecto. Este test es recomendado para ( $n$ ) muestras mayores a 25. (Moore, 2000).

## CAPÍTULO 3

### DISEÑO E IMPLEMENTACIÓN DEL PID DIFUSO PARA LA VARIABLE CAUDAL

En el presente capítulo se describe el diseño e implementación del controlador PID difuso para la variable caudal, de la plataforma de entrenamiento para control de procesos continuos, el mismo que se desarrolla en el software Matlab.

Para la correcta interacción del usuario con la plataforma es necesario realizar una interfaz humano-máquina (HMI), mediante el cual se realiza el control del proceso de caudal, para lograr lo mencionado anteriormente se aplicaron los siguientes softwares:

- **MATLAB:** En este programa se crea el algoritmo de control mediante un PID difuso para la variable caudal
- **TIA PORTAL:** Este programa sirve para la adquisición de datos que se lo realizará en el PLC
- **KEPSERVEX:** este programa establece la comunicación entre el PLC y el Matlab

Para entender el funcionamiento de cada uno de los elementos de control y actuadores de la plataforma de entrenamiento de procesos continuos es necesario conocer la caracterización de la misma, para lo cual se recomienda dirigirse al siguiente enlace del repositorio de la U.P.S: <http://dSPACE.ups.edu.ec/handle/123456789/10188>

#### **3.1 Sensor de caudal Rosemount 1151**

El sensor de caudal que se usa para el desarrollo de este proyecto es el modelo Rosemount 1151 analógico, se elige este sensor porque es un transmisor de baja potencia, maneja un caudal medio, su señal de salida es lineal, es de fácil instalación, este sensor cumple con varias características que se necesita para cubrir las necesidades

del proyecto, además cabe recalcar que es un sensor muy utilizado en las industrias que se dedican a envase o control de líquidos, ya que posee certificaciones Norteamericanas, es decir es un equipo antideflagrante estándar. En la Figura 3.1, se observa el sensor antes mencionado.

Fotografía del sensor de caudal usado



Figura 3.1. Sensor de caudal Rosemount 1151 analógico

Elaborado por: Stalin Andrade y Edison Tipán

En la Tabla 3.1 se muestra las características de trabajo de este sensor de caudal.

Tabla 3.1.

Características de trabajo del sensor Rosemount 1151

Alimentación	24[V]
Salida	4-20[mA]
Fluido de Trabajo	Agua
Rango de Visualización	0-100[%]

Nota: Características de trabajo del sensor de caudal

Elaborado por: Stalin Andrade y Edison Tipán

### 3.1.1 Calibración del sensor de caudal Rosemount 1151

Para el correcto funcionamiento del sensor de caudal se debe calibrar con el multímetro que trabaje con salida de 4-20 [mA], si no se realiza la respectiva calibración el algoritmo programado no corre y por consiguiente el proceso completo no funciona.

En la Figura 3.2 se muestra la conexión de las puntas del multímetro y los terminales del sensor para realizar la calibración.

Calibración de sensor de caudal Rosemount 1151



Figura 3.2. Conexión de las puntas del multímetro y los terminales del sensor

Elaborado por: Stalin Andrade y Edison Tipán

Para calibrar el sensor a 4[mA] y se visualice 0[%], en el sensor mover el tornillo con nombre Z, como se visualiza en la Figura 3.3.

Calibración del sensor



Figura 3.3. Calibración a cero del sensor de Caudal Rosemount 1151

Elaborado por: Stalin Andrade y Edison Tipán

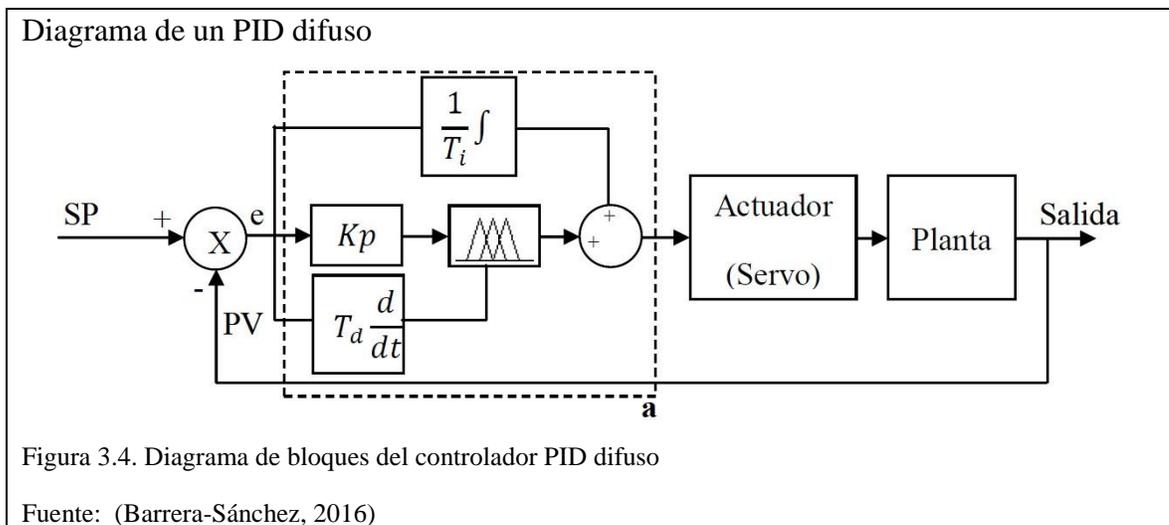
### 3.2 Diseño del controlador PID difuso

A continuación se presenta el diseño del controlador PID difuso para la variable caudal en el software Matlab, el algoritmo de control funciona en tiempo real mediante la programación en Simulink.

#### 3.2.1 Controlador PID difuso de la variable caudal

Para el diseño del control PID difuso de caudal se implementa una estructura FUZZY, donde la parte Proporcional y Derivativa ingresan al bloque FUZZY, y su respectiva salida es sumada con la parte integral. Este modelo de control tiene una gran ventaja ya que solo se utilizan 25 reglas de fusificación, por el contrario usando el modelo general donde la parte integral, proporcional y derivativa ingresan al bloque FUZZY, necesita 125 reglas de fusificación, lo que implica a una acción de control lenta.

El modelo seleccionado permite usar un bloque FUZZY normalizado para cualquier control, como se muestra en el lazo de control de la Figura 3.4.



#### 3.2.2 KEPServerEX

Este software permite enlazar el Matlab con el PLC S7-1200, en este caso en particular el PLC actúa únicamente como una tarjeta de adquisición de datos.

El KEPServerEX permite el intercambio de datos entre clientes OPC y PLC Siemens S7-200, S7-300, S7-400, S7-1200, este programa utiliza protocolos TCP/IP Ethernet.

En la Figura 3.5, se observa la pantalla principal del software KEPServerEX.

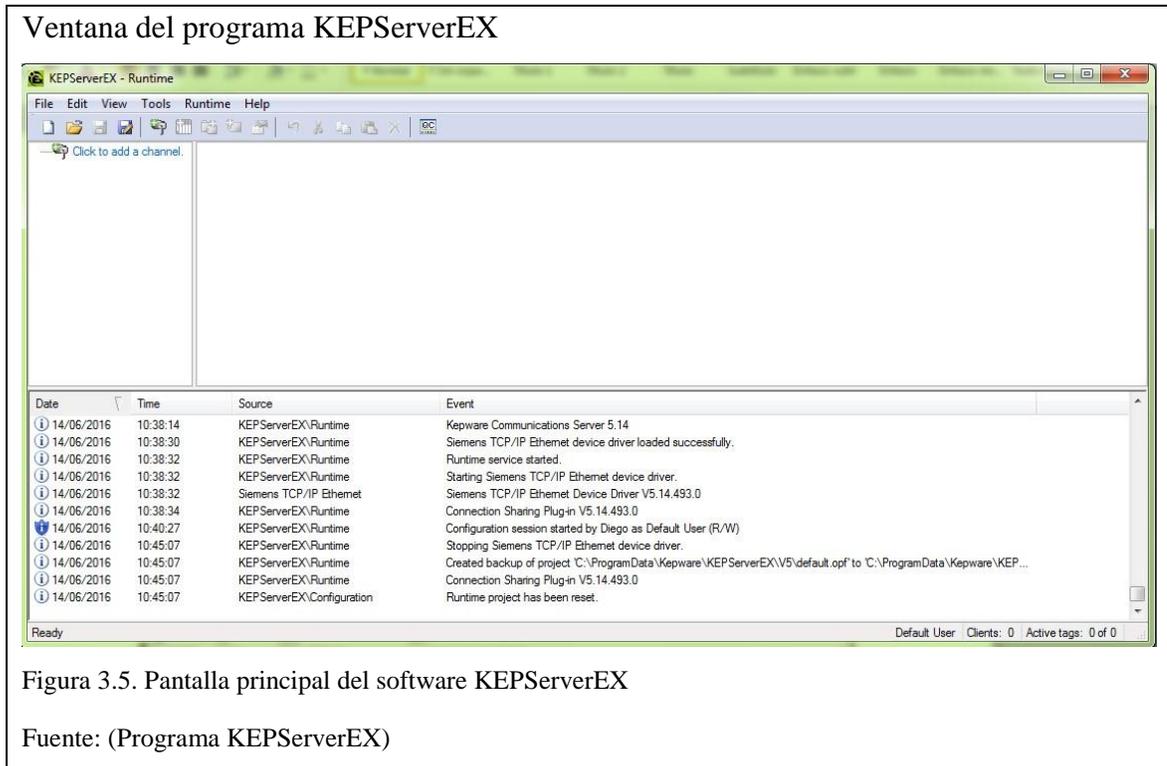


Figura 3.5. Pantalla principal del software KEPServerEX

Fuente: (Programa KEPServerEX)

La comunicación del OPC se configura con el procedimiento descrito en el Anexo 1.

### 3.2.3 Programación en Simulink/Matlab

Lo primero que se debe hacer es configurar el OPC KEPServerEX en Simulink siguiendo los pasos descritos en el Anexo 2.

Una vez configurado los bloques para la comunicación del KEPServerEX con el Simulink se realiza el algoritmo de programación para la variable de caudal, como se observa en la Figura 3.6.

# Algoritmo general del PID difuso hecho en Simulink

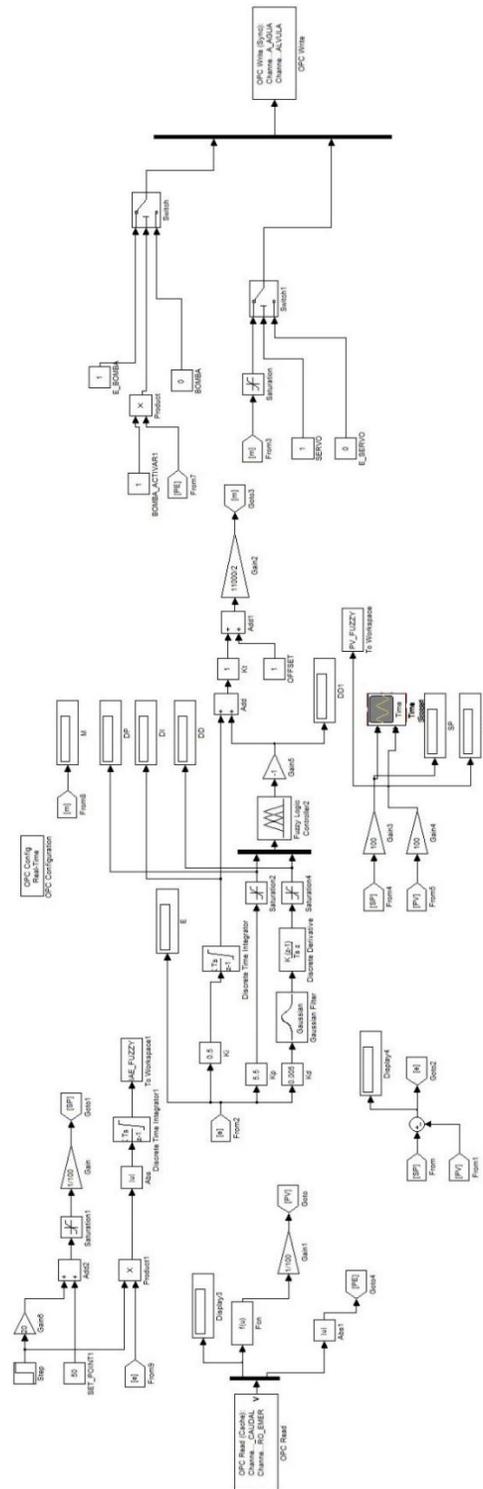
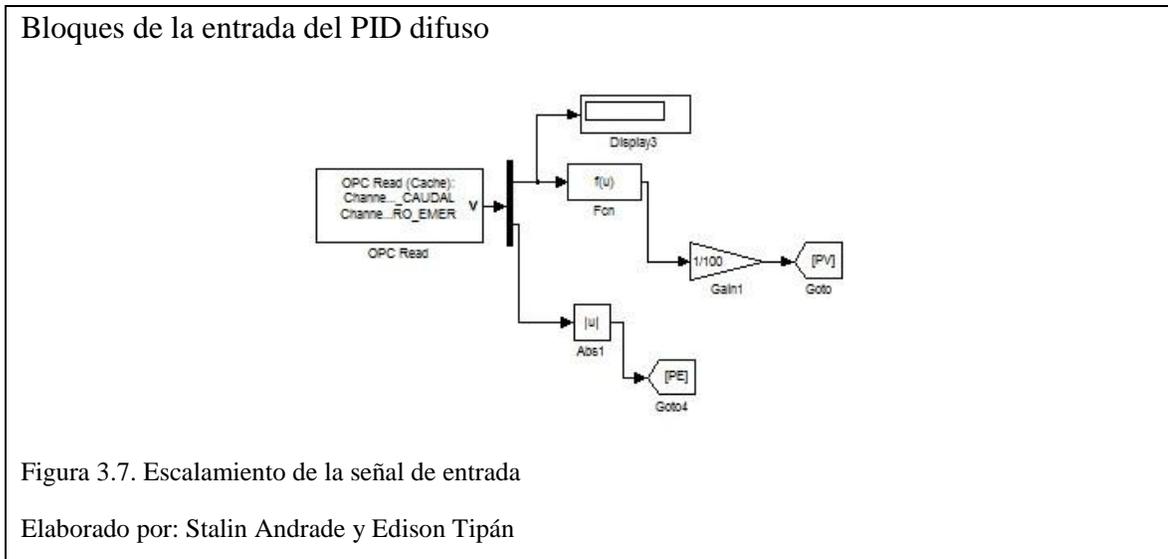


Figura 3.6. Controlador PID difuso para la variable caudal

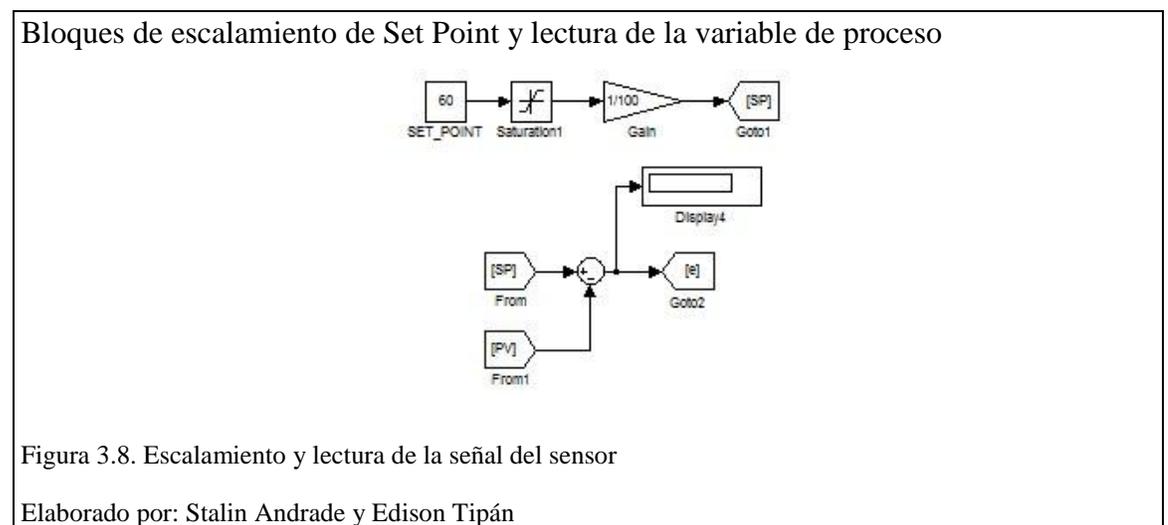
Elaborado por: Stalin Andrade y Edison Tipán

A continuación se detalla el algoritmo de programación en Simulink del controlador PID difuso para la variable caudal.

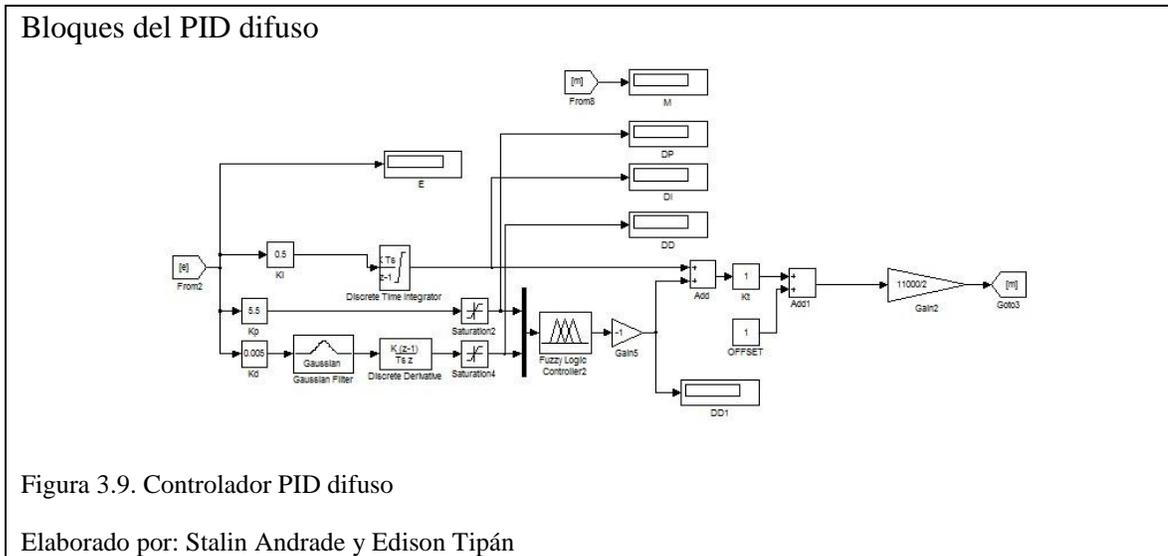
1. En la Figura 3.7, se muestra el escalamiento de 0 a 100[%] de la señal de entrada que viene a ser la lectura de los valores enviados por el sensor de caudal.



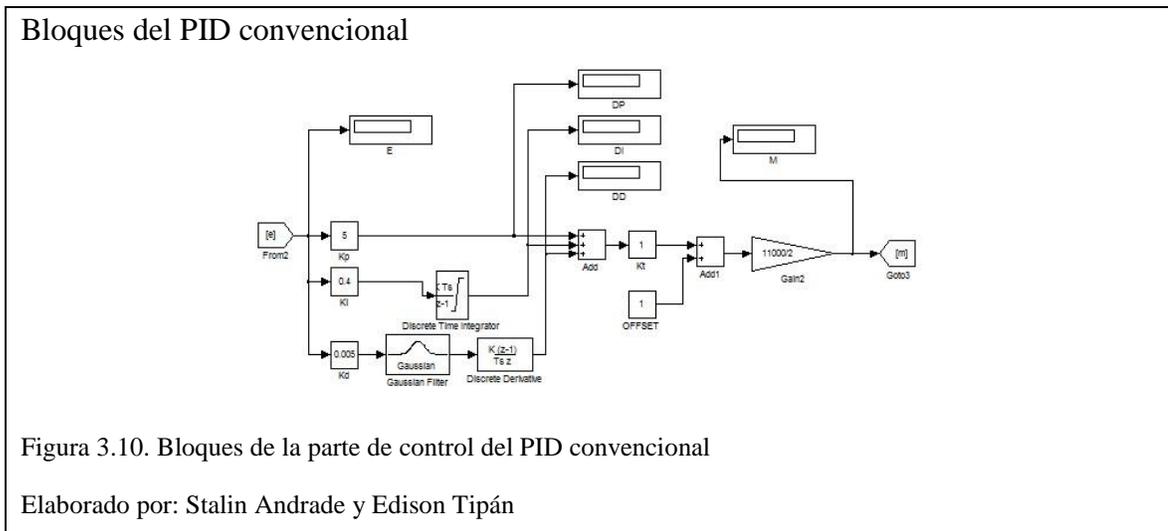
2. En la Figura 3.8, se muestra el valor del Set Point escalado en el rango de 0 a 100[%] y la sumatoria de los valores de Set Point y la lectura del sensor de caudal.



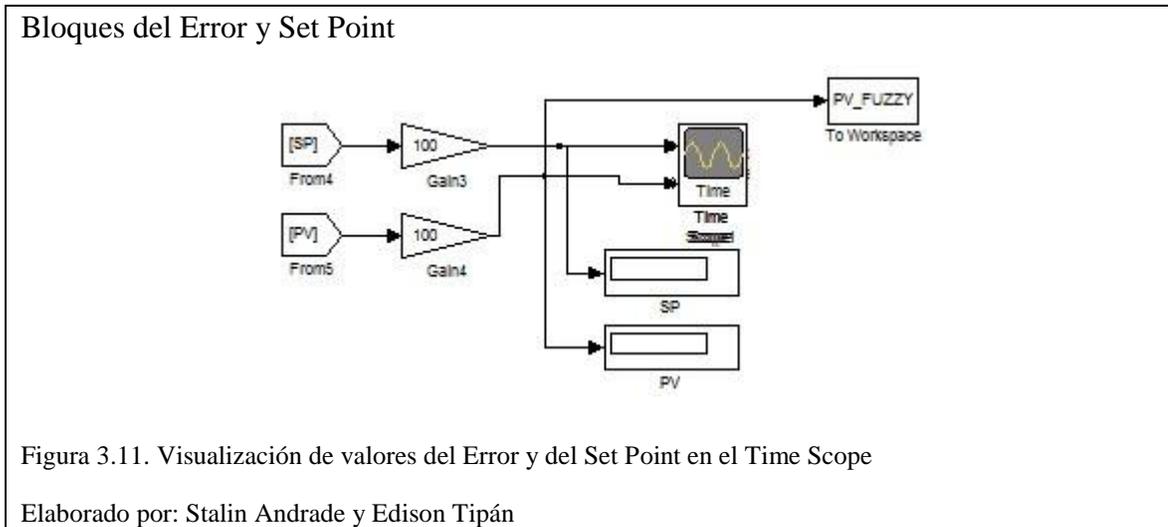
- En la Figura 3.9, se observa el controlador PID difuso, el error y la derivada del error van en un solo bloque, mientras que la integral del error ingresa a la sumatoria como otro bloque.



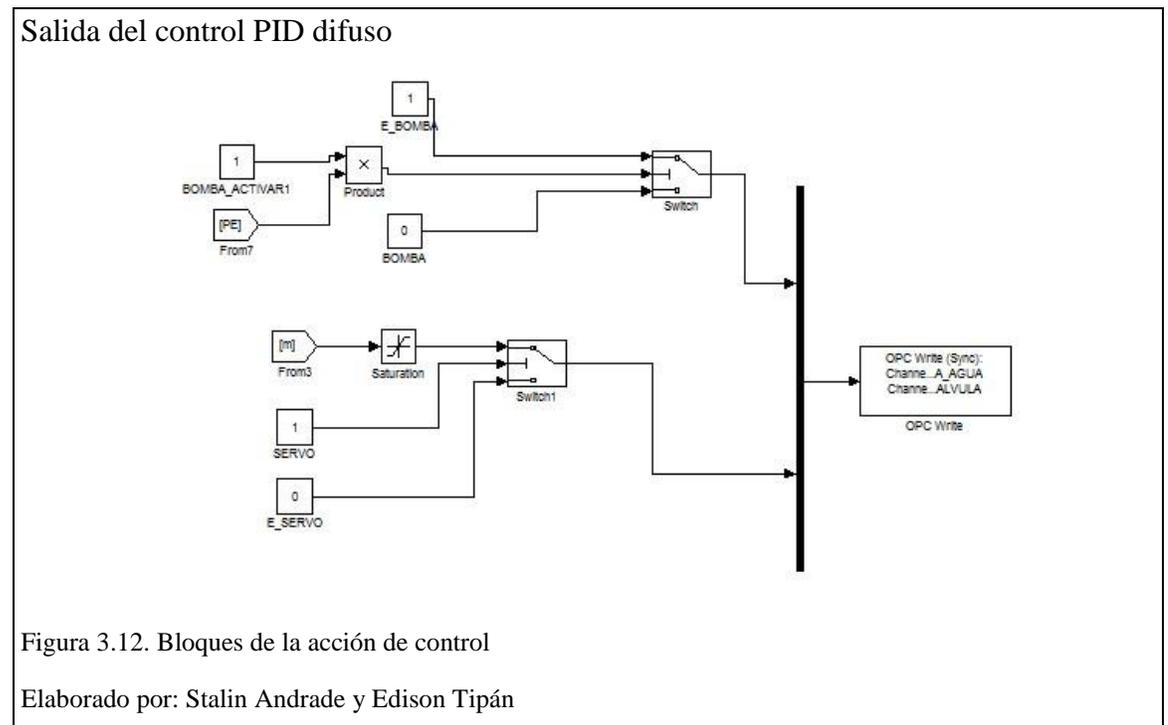
- En el caso del presente proyecto se realiza el algoritmo de un PID convencional para comparar la eficiencia entre ambos controladores, para el caso del PID convencional todo el algoritmo tiene las mismas configuraciones y bloques del PID FUZZY, lo único que varía es la parte de los bloques de la parte de control, como se muestra en la Figura 3.10.



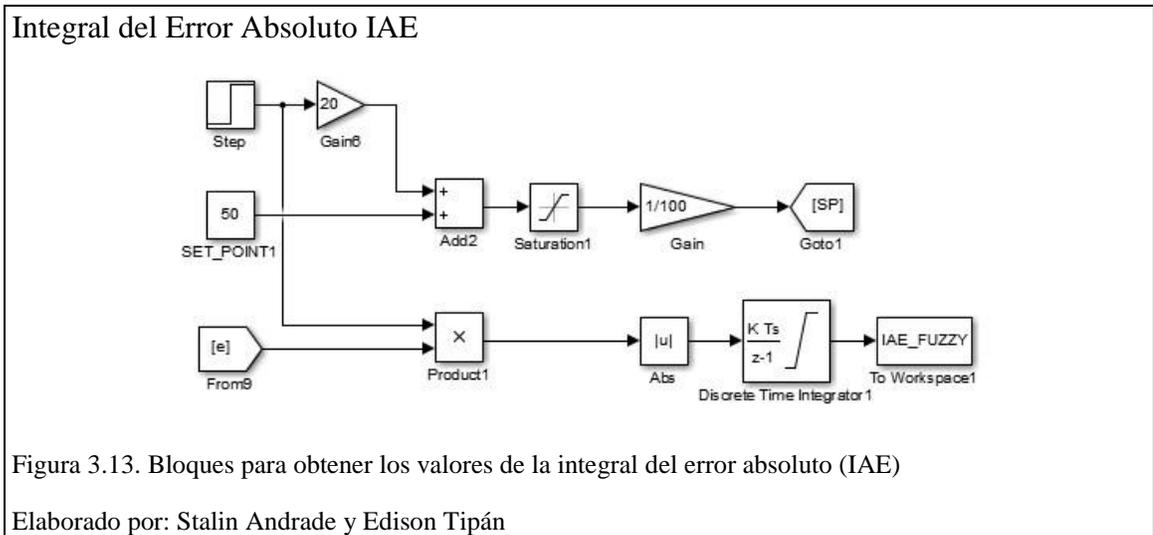
- En la Figura 3.11, se muestra los bloques de visualización los valores del error y del Set Point en el Time Scope, en el cual se observa los valores del proceso en tiempo real.



- En la Figura 3.12, se observa los bloques de la acción de control que en el presente proyecto es la servoválvula.



- En la Figura 3.13, se muestra los bloques para obtener los valores de la integral del error absoluto (IAE), con los cuales se compara entre el control PID convencional y el PID FUZZY.



Para continuar con el funcionamiento del proceso de control de caudal es necesario utilizar el FUZZY Logic Toolbox de Matlab, el cual crea un archivo con extensión fis (.fis) que se carga en el bloque difuso de algoritmo dentro de Simulink, para lograr lo anterior mencionado se debe seguir los siguientes pasos:

- Crear las variables lingüísticas y términos lingüísticos que son los identificadores en el controlador PID difuso. Para el caso del presente proyecto las variables lingüísticas son el error, la derivada del error y la acción de control, y los términos lingüísticos vienen a ser los conjuntos que pertenecen a las variables lingüísticas, a continuación se detalla lo anterior descrito.

- Error  
Conjuntos difusos para el error.  
Muy bajo: Error muy bajo.  
Bajo: Error bajo.  
Medio: Error medio.

Alto: Error alto.

Muy alto: Error muy alto.

- Derivada del error

Conjuntos difusos para la derivada del error

Dermuybajo: Derivada del error muy bajo.

Dervbajo: Derivada del error bajo.

Dervmedio: Derivada del medio.

Dervalto: Derivada del error alto.

Dermuyalto: Derivada del error muy alto.

- Acción de control

Conjuntos difusos para la acción de control.

Contrmuybajo: Salida de control muy bajo.

Contrbajo: Salida de control bajo.

Contrmedio: Salida de control medio.

Contralto: Salida de control alto.

Contrmuyalto: Salida de control muy alto.

2. Realizar la base de reglas de inferencia para el control difuso de caudal, como se muestra en la Tabla 3.2.

Tabla 3.2.

Reglas de inferencia para el control difuso de caudal

Error \ Derivada del error	Error				
	Muy bajo	Bajo	Medio	Alto	Muy alto
Dermuybajo	Contrmuybajo	Contrmuybajo	Contrmuybajo	Contrbajo	Contrmedio
Dervbajo	Contrmuybajo	Contrbajo	Contrbajo	Contrmedio	Contralto
Dervmedio	Contrmuybajo	Contrbajo	Contrmedio	Contralto	Contrmuyalto
Dervalto	Contrbajo	Contrmedio	Contralto	Contrmuyalto	Contrmuyalto
Dermuyalto	Contrmedio	Contralto	Contrmuyalto	Contrmuyalto	Contrmuyalto

Nota: en la tabla se muestra solo las abreviaturas de las variables y términos lingüísticos

Elaborado por: Stalin Andrade y Edison Tipán

3. En la ventana de comandos de Matlab escribir “FUZZY”, y seguir los pasos descritos en el Anexo 3, para crear las variables lingüísticas y funciones de membresía.

Para la realización de la interfaz humano-máquina (HMI) del control PID difuso de caudal, se debe utilizar el Guide de Matlab, para crear lo anterior mencionado se debe seguir los pasos descritos en el Anexo 4.

### **3.2.4 TIA Portal**

Para que el PLC actúe únicamente como tarjeta de adquisición de datos, se debe realizar los pasos descritos en el Anexo 5.

### **3.3 Implementación del controlador PID difuso**

Para la puesta en marcha del controlador PID convencional y PID difuso, implementada en la plataforma de entrenamiento de procesos continuos, se debe seguir los pasos detallados a continuación:

1. Abrir el programa TIA PORTAL con el proyecto creado anteriormente y realizar lo siguiente, como se muestra en la Figura 3.14.
  - a. Click en compilar
  - b. Click en cargar
  - c. Establecer conexión online.

## TIA PORTAL

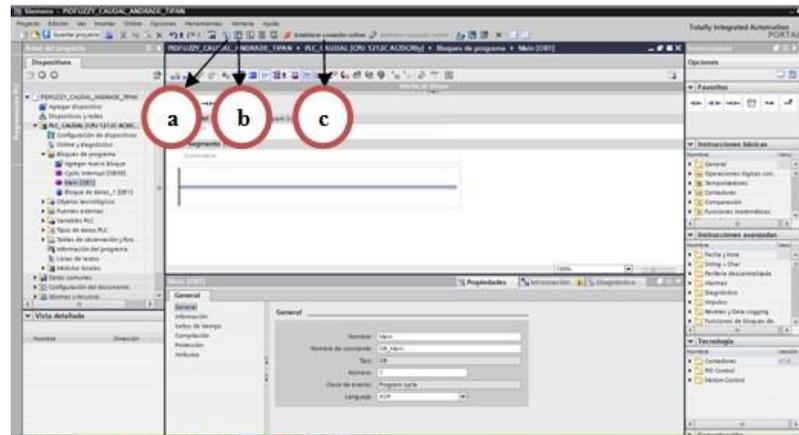
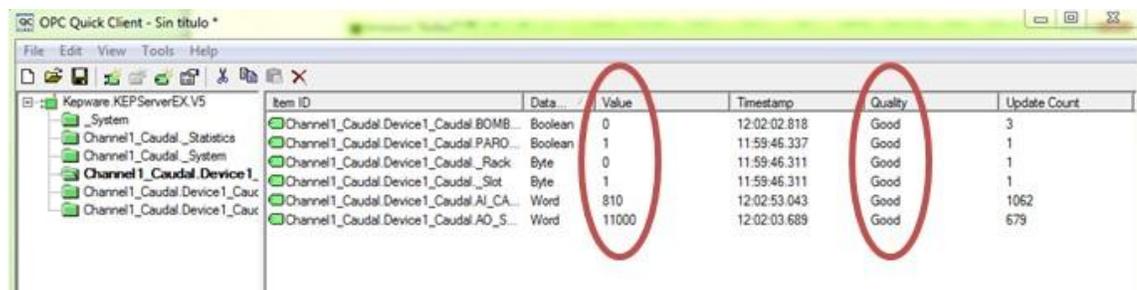


Figura 3.14. Pasos para establecer conexión online del PLC

Fuente: (Programa TIA PORTAL)

2. Abrir el programa KEPServerEX con el proyecto creado anteriormente, abrir la ventana OPC Quick Client y verificar el estado de la conexión entre el PLC Y el KEPServerEX, como se observa en la Figura 3.15.

## Pantalla del KEPServerEX



Item ID	Data...	Value	Timestamp	Quality	Update Count
Channel1_Caudal_Device1_Caudal_BOMB...	Boolean	0	12:02:02.818	Good	3
Channel1_Caudal_Device1_Caudal_PARO...	Boolean	1	11:59:46.337	Good	1
Channel1_Caudal_Device1_Caudal_Rack...	Byte	0	11:59:46.311	Good	1
Channel1_Caudal_Device1_Caudal_Slot...	Byte	1	11:59:46.311	Good	1
Channel1_Caudal_Device1_Caudal_AI_CA...	Word	810	12:02:53.043	Good	1062
Channel1_Caudal_Device1_Caudal_AO_S...	Word	11000	12:02:03.689	Good	679

Figura 3.15. Estado de la conexión entre el PLC Y el KEPServerEX.

Fuente: (Programa KEPServerEX)

3. Abrir el programa Matlab, en la ventana de comandos escribir “CARATULA”, se despliega la pantalla principal desde donde se controla el proceso, como se muestra en la Figura 3.16.

## MATLAB



Figura 3.16. Pantalla principal del HMI

Elaborado por: Stalin Andrade y Edison Tipán

4. Abrir la ventana de editor FUZZY, escribiendo “FUZZY” en la ventana de comandos de Matlab y seguir los pasos que se detallan a continuación y como se observa en la Figura 3.17.
  - a. Click en File.
  - b. Click en Import.
  - c. Click en From file.

### Importación del proyecto en Toolbox Logic FUZZY

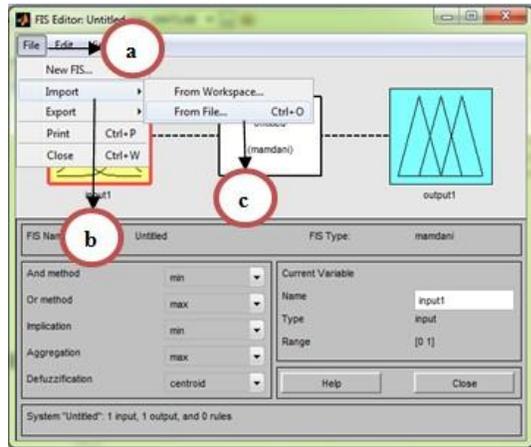


Figura 3.17. Pasos para importar el proyecto FUZZY a Matlab

Elaborado por: Stalin Andrade y Edison Tipán

5. Seleccionar el archivo con extensión fis (.fis), creado anteriormente y dar click en Abrir como se muestra en la Figura 3.18.

### Elección de archivo en Toolbox Logic FUZZY

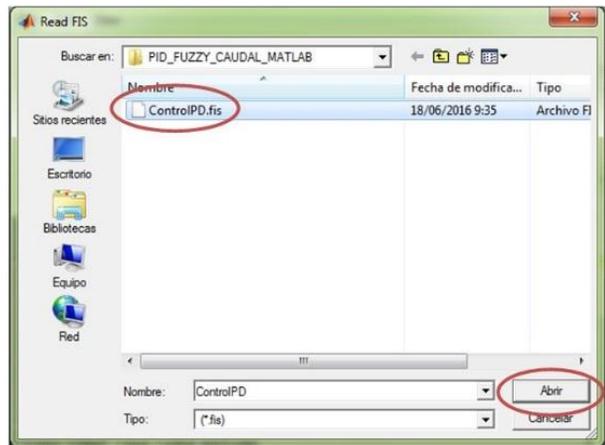


Figura 3.18. Selección del archivo con extensión fis (.fis)

Elaborado por: Stalin Andrade y Edison Tipán

6. Para exportar el proyecto FUZZY al Wokspace de Matlab, en la ventana de editor FUZZY seguimos los siguientes pasos, como se observa en la Figura 3.19.

- a. Click en File
- b. Click en Export
- c. Click en To Workspace

### Exportación del proyecto FUZZY

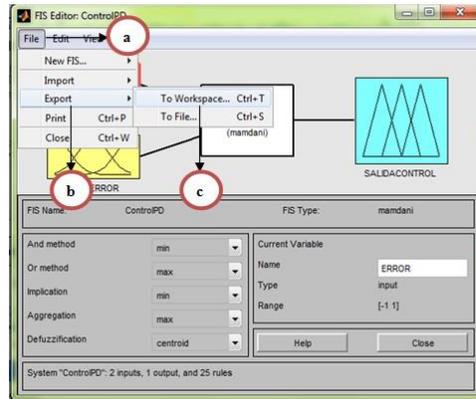


Figura 3.19. Pasos para exportar el proyecto FUZZY al Workspace de Matlab

Elaborado por: Stalin Andrade y Edison Tipán

7. En la nueva ventana que se despliega dar click en OK y cerrar todas las ventanas emergentes abiertas del editor FUZZY, como se muestra en la Figura 3.20.

### Confirmación del proyecto FUZZY

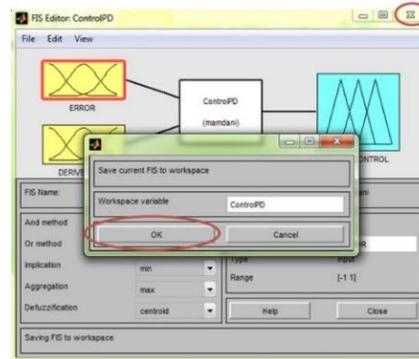


Figura 3.20. Exportar el proyecto FUZZY al Workspace de Matlab.

Elaborado por: Stalin Andrade y Edison Tipán

8. La pantalla principal desplegada con el nombre de CARATULA, permite elegir el control que se desea ejecutar, ya sea PID convencional o PID difuso, como se observa en la Figura 3.21.



A continuación en la Tabla 3.3 se muestra las funciones de los botones de la pantalla principal del HMI.

Tabla 3.3.

Funciones de la pantalla principal

Elemento	Acción
Controlador PID Convencional	Despliega la ventana del controlador del PID convencional
Controlador PID Fuzzy	Despliega la ventana del controlador del PID Fuzzy

Nota: Detalle de la función de los botones de la pantalla principal del HMI

Elaborador por: Stalin Andrade y Edison Tipán

9. Seleccionar uno de los dos procesos que se requiera ejecutar y se despliega las ventanas como se observan en las Figura 3.22 y Figura 3.23.

### Pantalla del HMI del controlador PID convencional



Figura 3.22. Interfaz gráfica del controlador PID convencional

Elaborado por: Stalin Andrade y Edison Tipán

En la Tabla 3.4 se describe las funciones de los botones de la pantalla del controlador PID convencional.

Tabla 3.4.

Funciones de la interfaz gráfica del controlador PID convencional

<b>Elemento</b>	<b>Acción</b>
SET POINT	Ingresa el punto de consigna a controlar
INICIAR	Inicia el control PID Convencional
PARAR	Detiene el control PID Convencional
REGRESAR	Retorna a la venta principal

Nota: Detalle de la función de los elementos de la interfaz gráfica del controlador PID convencional

Elaborador por: Stalin Andrade y Edison Tipán

En la Figura 3.23 se muestra la pantalla del controlador PID FUZZY.

### Pantalla del HMI del controlador PID FUZZY



Figura 3.23. Interfaz gráfica del controlador PID FUZZY

Elaborador por: Stalin Andrade y Edison Tipán

En la Tabla 3.5 se describe las funciones de los botones de la pantalla del controlador PID FUZZY.

Tabla 3.5.

Funciones de la interfaz gráfica del controlador PID FUZZY

Elemento	Acción
SET POINT	Ingresa el punto de consigna a controlar
INICIAR	Inicia el control PID FUZZY
PARAR	Detiene el control PID FUZZY
REGRESAR	Retorna a la venta principal

Nota: Detalle de la función de los elementos de la interfaz gráfica del controlador PID FUZZY

Elaborador por: Stalin Andrade y Edison Tipán

En la Figura 3.24 se observa el diagrama de flujo del funcionamiento de la interfaz gráfica del controlador PID convencional y FUZZY.

Diagrama de Flujo del controlador PID convencional y FUZZY

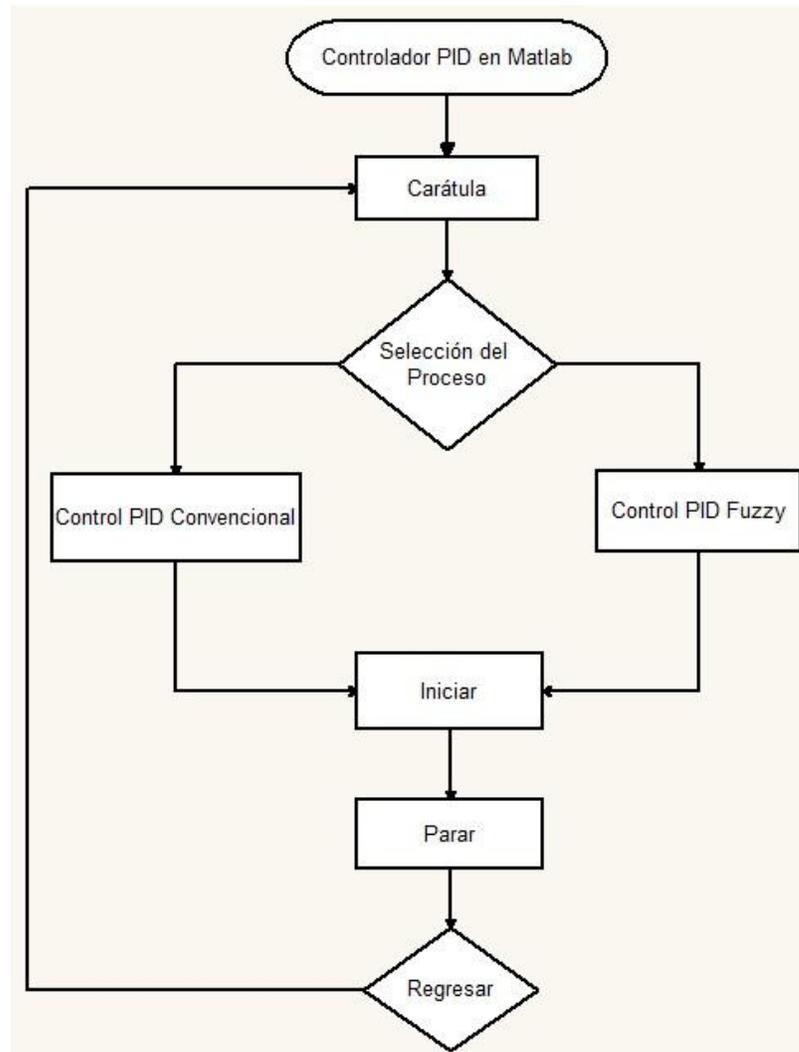


Figura 3.24. Flujo del Proceso del controlador PID

Elaborado por: Stalin Andrade y Edison Tipán

10. Para obtener las gráficas y valores correspondientes a la integral del error absoluto (IAE), se debe seguir los siguientes pasos:

- a. Ejecutar los algoritmos en forma individual como se muestra en la Figura 3.25, es decir ejecutarlos fuera de línea y no directamente desde la interfaz gráfica creada, se debe colocar el mismo tiempo en ambos programas, en este caso se colocó 120[s].

## PID convencional y PID difuso

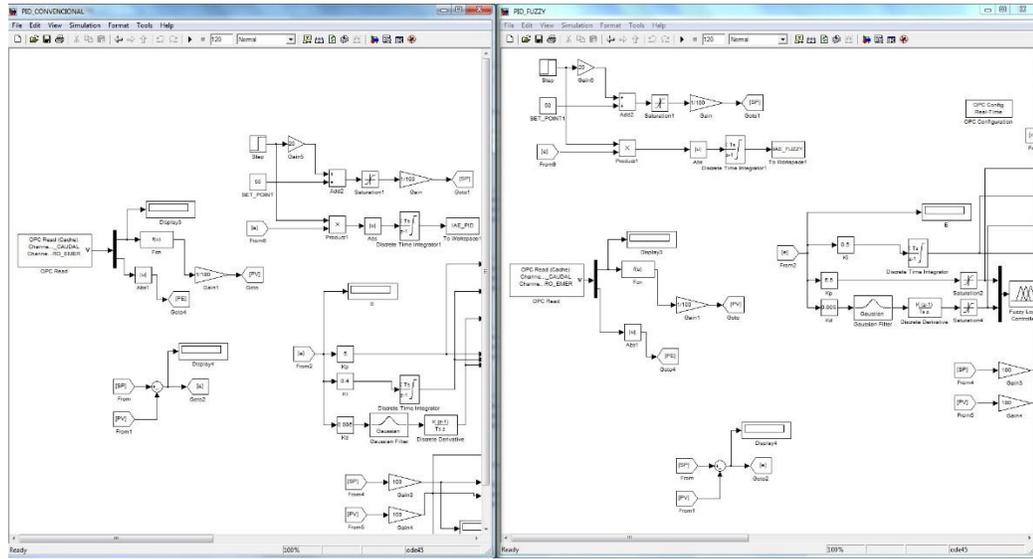


Figura 3.25. Ejecución fuera de línea del PID convencional y PID FUZZY.

Elaborado por: Stalin Andrade y Edison Tipán

En la Figura 3.26 se observa ejecutándose en tiempo real el controlador PID convencional.

## Controlador PID convencional

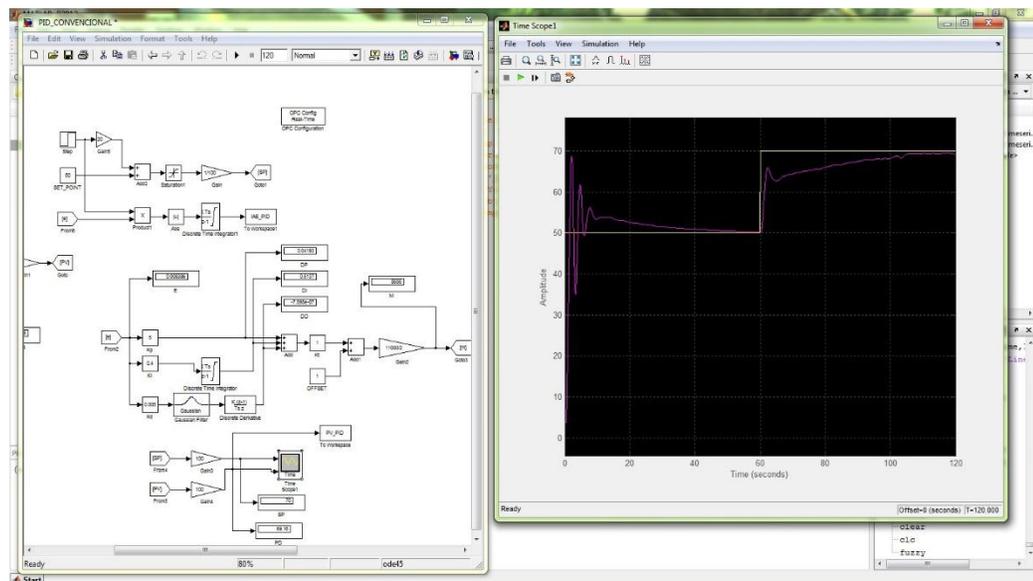
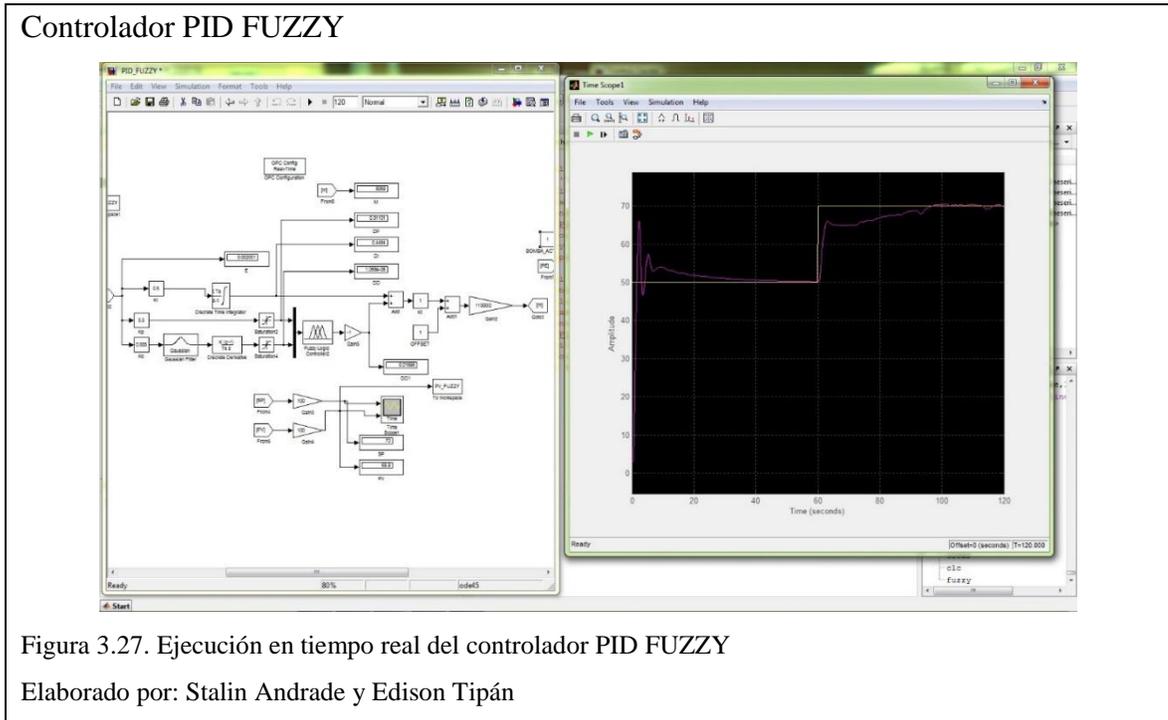


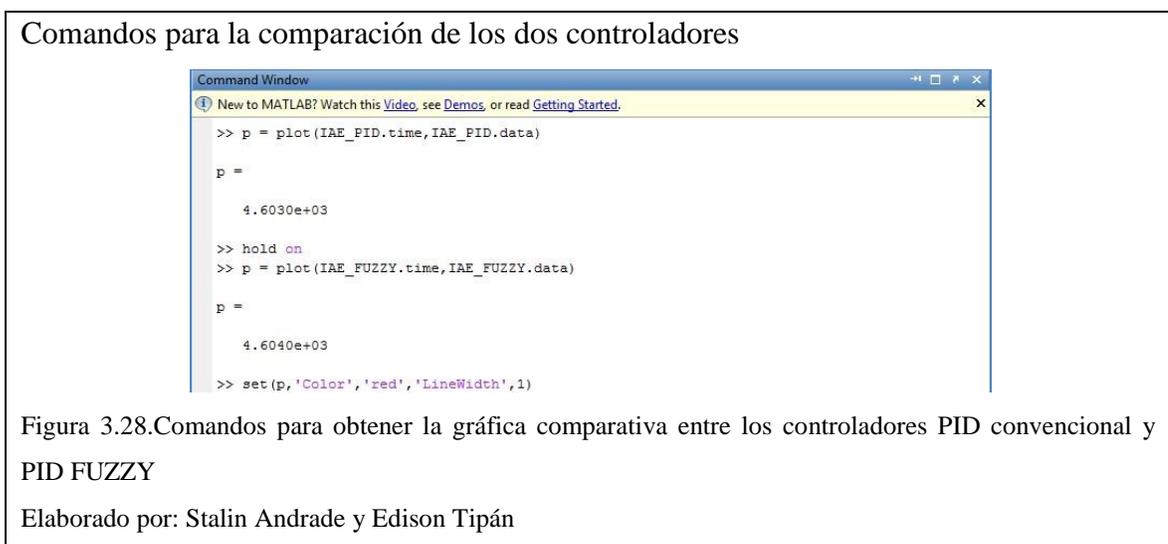
Figura 3.26. Ejecución en tiempo real del controlador PID convencional

Elaborado por: Stalin Andrade y Edison Tipán

En la Figura 3.27 se observa ejecutándose en tiempo real el controlador PID FUZZY.



- b. Una vez ejecutado los dos programas por separado dirigirse a la ventana de comandos de Matlab e ingresar los comandos que se muestran en la Figura 3.28.



La gráfica comparativa se observa en la Figura 3.29.

### Comparación gráfica de los dos controladores

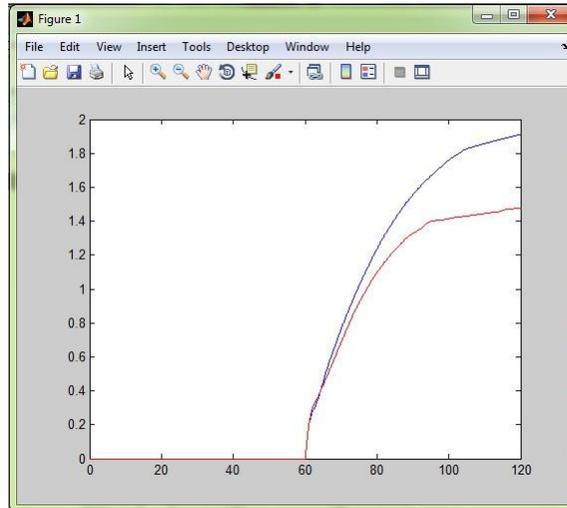


Figura 3.29. Gráfica comparativa entre controladores. PID convencional (curva color azul) y PID FUZZY (curva color rojo).

Elaborado por: Stalin Andrade y Edison Tipán

- c. Para obtener los valores de la integral del error absoluto de cada uno de los controladores PID convencional y PID FUZZY dirigirse a la ventana de comandos de Matlab e ingresar los comandos que se muestran en la Figura 3.30.

### Valores de IAE

```
>> IAE_PID.data
ans =
    0
 0.0583
 0.1166
 0.1749
 0.2332
```

```
>> IAE_FUZZY.data
ans =
    0
 0.0579
 0.1158
 0.1737
 0.2316
```

Figura 3.30. (a) Valores de PID convencional y (b) Valores de PID FUZZY.

Elaborado por: Stalin Andrade y Edison Tipán

Al ingresar cada comando para obtener el valor de IAE se despliega 120 valores porque el tiempo de ejecución que se ingreso fue de 120[s], por tal razón para hacer la

comparación entre el controlador PID convencional y PID FUZZY se elige el último valor desplegado en cada caso respectivamente. En la Tabla 3.6 se observa un ejemplo de los valores de IAE obtenidos en una muestra.

Tabla 3.6.

Valores de IAE

<b>Controlador</b>	<b>Valores de IAE</b>
PID Convencional	1.9125
PID Fuzzy	1.4781

Nota: Ejemplo de valores de IAE obtenidos en una muestra realizada durante la mañana

Elaborado por: Stalin Andrade y Edison Tipán.

## CAPÍTULO 4

### PRUEBAS Y RESULTADOS

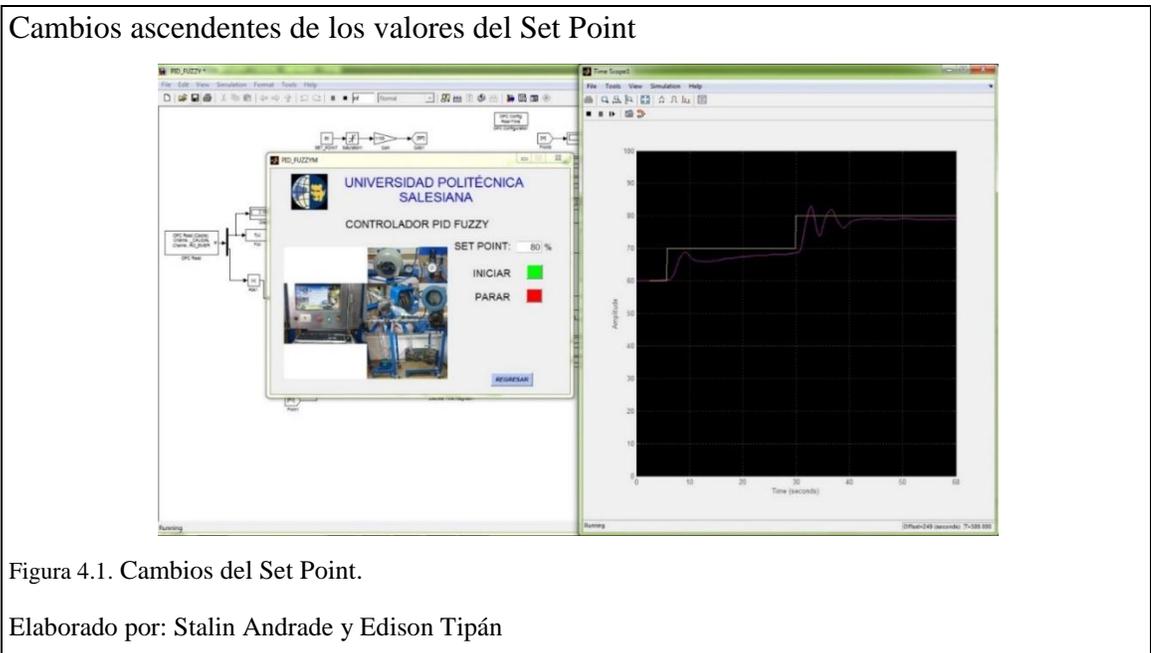
Se especifica en este capítulo las pruebas hechas al controlador PID difuso de la variable de caudal, evaluando el comportamiento del controlador en Matlab, además se indica la comparación entre el PID convencional y el PID difuso mediante el índice de desempeño IAE usando el Test de Wilcoxon.

#### 4.1 Pruebas del controlador PID difuso de caudal

Las pruebas que se realiza en Matlab son la respuesta del sistema de control cuando varía la señal del Set Point.

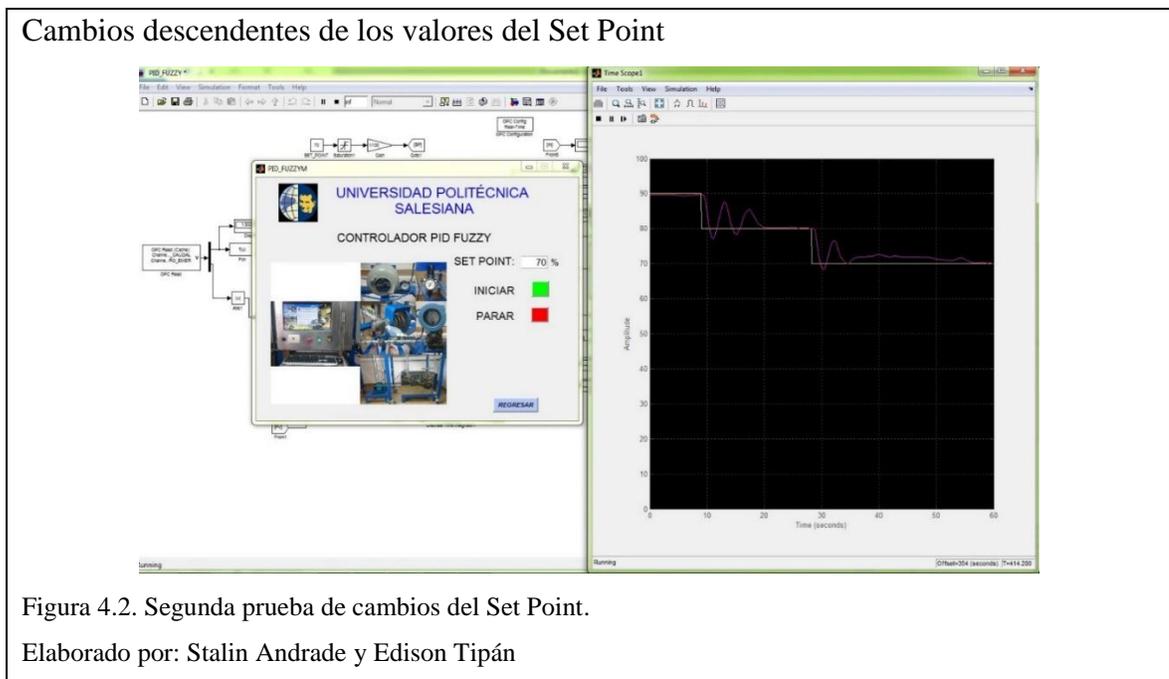
##### 4.1.1 Respuesta del sistema a la señal del Set Point

Esta prueba consiste en mirar la respuesta del sistema de control ante cambios ascendentes de la señal del Set Point, los resultados se observan en la Figura 4.1.



Como se puede observar en la Figura 4.1 el actuador cambia su estado para poder compensar el nuevo valor de la señal del Set Point. El cambio del Set Point esta desde 60% para cambiar a 70% y finalmente a 80%.

La segunda prueba consiste en hacer cambios descendentes del valor del Set Point para observar la respuesta del sistema, el resultado se observa en la Figura 4.2.



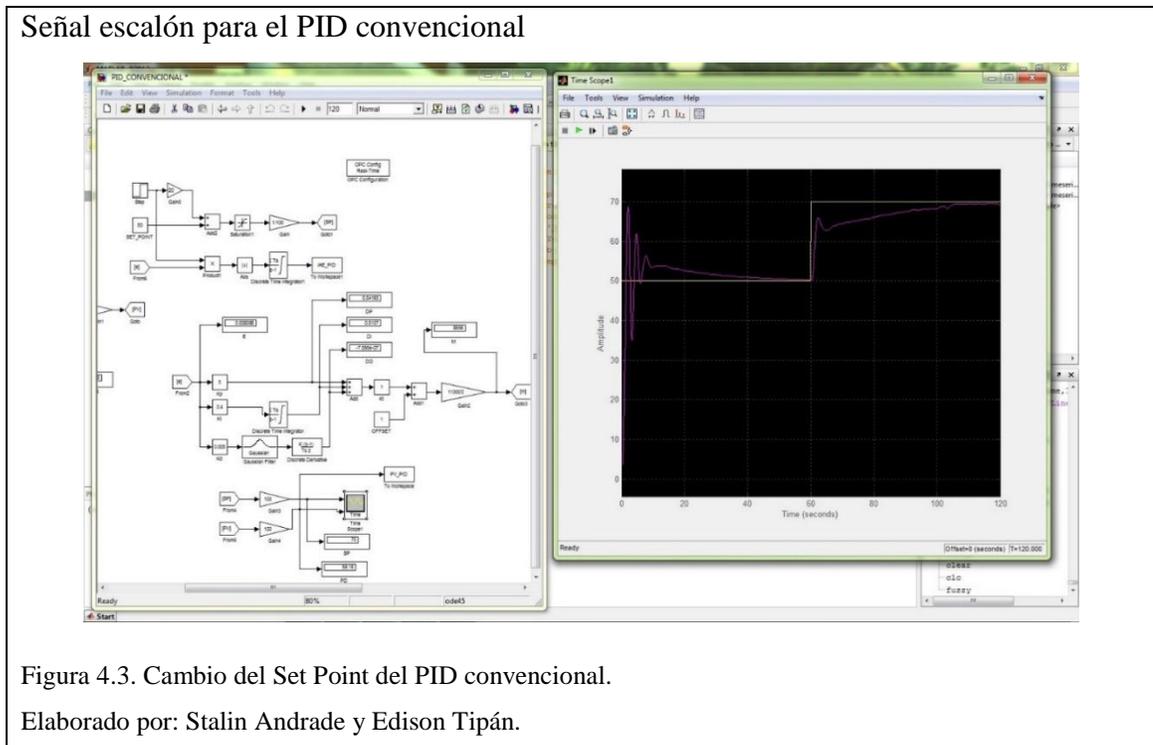
En la Figura 4.2 se observa que el actuador cambia su estado para poder compensar el nuevo valor de la señal del Set Point. El cambio del Set Point esta desde 90% para cambiar a 80% y finalmente a 70%.

#### 4.2 Pruebas del controlador PID convencional y del PID difuso

Las pruebas consisten en ver la respuesta del sistema al cambio de la señal del Set Point, ubicado primero en 50% para luego de un tiempo de 60 segundos que el sistema se estabiliza dar un escalón hasta 70% donde el sistema se vuelve a estabilizar en un tiempo de 120 segundos; estas pruebas se realizan tanto para el PID convencional como para el PID difuso.

#### 4.2.1 Respuesta de la señal del Set Point en Matlab para el controlador PID convencional

En el controlador PID convencional el actuador modifica el estado en que se encuentra para poder compensar el nuevo valor de 70% del Set Point, como se puede observar en la Figura 4.3.



#### 4.2.2 Respuesta de la señal del Set Point en Matlab para el controlador PID difuso

En el controlador PID difuso el actuador al igual que en el PID convencional modifica el estado en que se encuentra para poder compensar el nuevo valor de 70% del Set Point, como se puede observar en la Figura 4.4.

## Señal escalón para el PID difuso

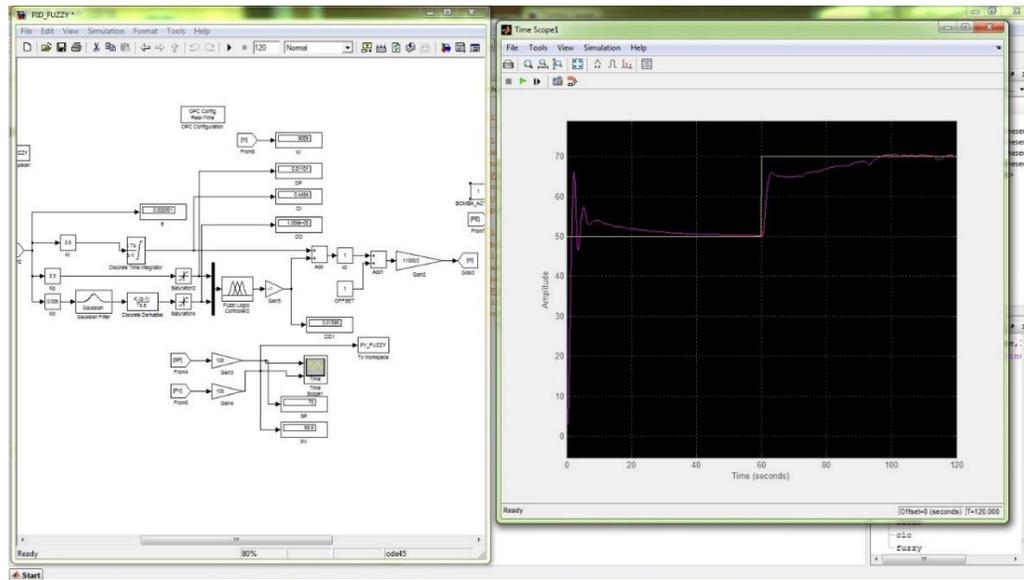


Figura 4.4. Cambio del Set Point del PID difuso.

Elaborado por: Stalin Andrade y Edison Tipán.

### 4.3 Resultados obtenidos del índice de desempeño IAE entre el PID convencional y el PID difuso

En los procesos reales el control de caudal se lo realiza dentro de un rango específico, por tal motivo en el presente proyecto se hizo las pruebas mediante el índice de desempeño IAE, usando el rango de caudal del 50% al 70%, que por lo general es usado dentro de las industrias.

En la Figura 4.5, se observa la gráfica de la diferencia entre los valores obtenidos del IAE del PID convencional (color azul) y el IAE del PID difuso (color rojo).

## Resultado de los IAE convencional y difuso

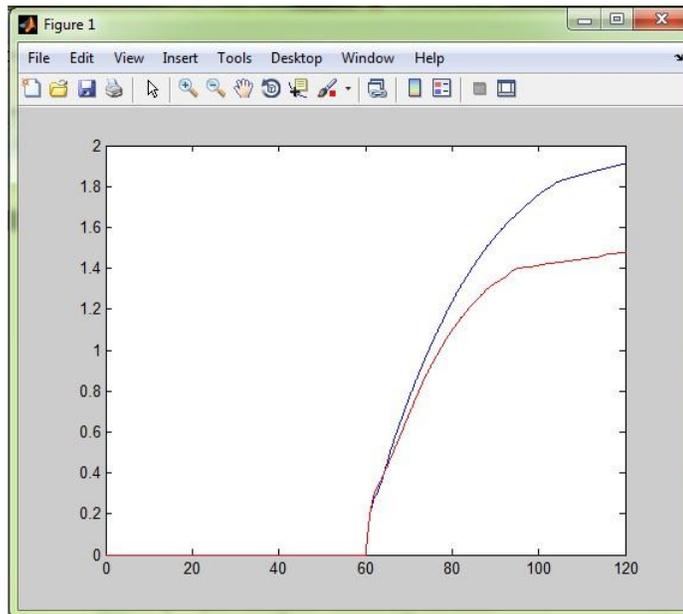


Figura 4.5. PID convencional (azul) y del PID difuso (rojo).

Elaborado por: Stalin Andrade y Edison Tipán.

### 4.4 Análisis del desempeño de Controladores PID convencionales y PID difusos mediante el error absoluto – IAE

Mediante el Test de Wilcoxon se demuestra si el IAE del PID convencional es igual o diferente del IAE del PID difuso.

Se toma en cuenta los siguientes parámetros:

- Si el  $IAE_{conv.}$  es igual al  $IAE_{difuso}$ , no se obtendría la optimización buscada.
- Si el  $IAE_{conv.}$  es mayor al  $IAE_{difuso}$ , se demostraría que se optimiza el controlador PID, ya que ha disminuido la pérdida de energía del sistema.
- Si el  $IAE_{conv.}$  es menor al  $IAE_{difuso}$ , se demostraría que ha empeorado el sistema, ya que aumentaría la pérdida de energía en vez de disminuir.

El Test de Wilcoxon se usa para muestras (n) pequeñas mayores a 25.

#### 4.4.1 Test de Wilcoxon del proceso de caudal PID convencional Vs PID difuso

Para realizar el Test de Wilcoxon se ha tomado en total 30 muestras (n) en diferentes horarios: matutino (Ver Anexo 6), vespertino (Ver Anexo 7) y nocturno (Ver Anexo 8).

En la Tabla 4.1 se presenta los datos obtenidos para el Test de Wilcoxon del proceso de control de caudal, en donde  $IAE_{conv}$  indica el índice del error absoluto del PID convencional y el  $IAE_{difuso}$  indica el índice del error absoluto del PID difuso.

Tabla 4.1.

Test de Wilcoxon del Proceso de Caudal

TEST DE WILCOXON PARA DATOS PAREADOS, EN EL PROCESO DE CONTROL DE CAUDAL									
HIPÓTESIS		Principal (Ho): $IAE_{conv} = IAE_{difuso}$					Media( $IAE_{conv}$ ) = Media( $IAE_{difuso}$ )		
		Alternativa (Ha): $IAE_{conv} > IAE_{difuso}$					Media( $IAE_{conv}$ ) > Media( $IAE_{difuso}$ )		
n	IAE CONV	IAE DIFUSO	DIFERENCIAS	RANGOS ASIGNADOS	T+	T-	LIGADURA	DIFERENCIAS ORDENADAS	RANGOS DE ORDEN
1	1,9125	1,4781	0,4344	26	26			0,2654	1
2	1,9625	1,4681	0,4944	30	30			0,2745	2
3	1,9515	1,5295	0,422	22	22			0,2913	3
4	1,8424	1,4816	0,3608	9	9			0,3161	4
5	1,8853	1,4837	0,4016	19	19			0,326	5
6	1,8419	1,4549	0,387	16	16			0,3535	6
7	1,8568	1,5007	0,3561	7	7			0,3561	7
8	1,9074	1,4843	0,4231	24	24			0,3594	8
9	1,8504	1,4969	0,3535	6	6			0,3608	9
10	1,9327	1,4613	0,4714	29	29			0,3616	10
11	2,0204	1,5899	0,4305	25	25			0,3624	11
12	1,9721	1,5111	0,461	28	28			0,3638	12
13	1,9288	1,5072	0,4216	21	21			0,3703	13
14	1,9421	1,5827	0,3594	8	8			0,3721	14
15	1,8034	1,5289	0,2745	2	2			0,3745	15

Continuación Tabla 4.1.

Test de Wilcoxon del Proceso de Caudal

TEST DE WILCOXON PARA DATOS PAREADOS, EN EL PROCESO DE CONTROL DE CAUDAL									
HIPÓTESIS		Principal (H <sub>0</sub> ): IAE <sub>conv</sub> = IAE <sub>difuso</sub>				Media(IAE <sub>conv</sub> ) = Media(IAE <sub>difuso</sub> )			
		Alternativa (H <sub>a</sub> ): IAE <sub>conv</sub> > IAE <sub>difuso</sub>				Media(IAE <sub>conv</sub> ) > Media(IAE <sub>difuso</sub> )			
n	IAE CONV	IAE DIFUSO	DIFERENCIAS	RANGOS ASIGNADOS	T+	T-	LIGADURA	DIFERENCIAS ORDENADAS	RANGOS DE ORDEN
16	1,9159	1,5456	0,3703	13	13			0,387	16
17	1,9121	1,5196	0,3925	18	18			0,3905	17
18	1,8742	1,5829	0,2913	3	3			0,3925	18
19	1,8985	1,5240	0,3745	15	15			0,4016	19
20	1,9514	1,5876	0,3638	12	12			0,4203	20
21	1,9115	1,4770	0,4345	27	27			0,4216	21
22	1,9512	1,5290	0,4222	23	23			0,422	22
23	1,9053	1,4850	0,4203	20	20			0,4222	23
24	1,8968	1,5807	0,3161	4	4			0,4231	24
25	1,8604	1,4980	0,3624	11	11			0,4305	25
26	1,9904	1,5999	0,3905	17	17			0,4344	26
27	1,9388	1,5772	0,3616	10	10			0,4345	27
28	1,8534	1,5880	0,2654	1	1			0,461	28
29	1,9511	1,5790	0,3721	14	14			0,4714	29
30	1,8990	1,5730	0,326	5	5			0,4944	30

Nota: Muestras obtenidas para el análisis del Test de Wilcoxon PID convencional Vs PID difuso, en el Proceso de Control de Caudal.

Elaborado por: Stalin Andrade y Edison Tipán.

A continuación se realiza un análisis estadístico siguiendo el procedimiento del Test de Wilcoxon, donde las muestras son analizadas, en la Tabla 4.2 se observa los valores obtenidos:

Tabla 4.2.

Resultados del Test de Wilcoxon

<b>Total T+</b>			465
<b>Total T-</b>			0
<b>T=min(T+,T-)</b>			465
$Z = \frac{T - n \frac{(n + 1)}{4}}{\sqrt{\frac{n(n + 1)(2n + 1)}{24}}}$			4,7821389
<b>Nivel de Confianza del 95% (Za)</b>			1,96
<b>Decisión</b>	$Z \leq Z_a$	Se acepta la <b>Ho</b>	
	$Z > Z_a$	Se acepta la <b>Ha</b>	ok
<b>Interpretación</b>			Como el IAE del PID convencional, es mayor al IAE del PID difuso, se concluye que el PID difuso es mejor en el proceso de control de caudal, pues logra disminuir pérdida de energía en el sistema.

## CONCLUSIONES

La conclusión más gratificante del presente proyecto de titulación es que se cumplió en su totalidad el objetivo principal de diseñar un controlador PID difuso de la variable caudal de la plataforma de entrenamiento de procesos continuos, llegando así a complementar el desempeño total de la planta que ya tenían implementado procesos de nivel y temperatura.

El control de caudal de la plataforma maneja un rango de control óptimo de 40 a 95 [%], por la razón que el actuador de la planta, en este caso la servoválvula, necesita de un flujo de agua moderado, ya que al realizar control con bajo caudal se produce un sobrecalentamiento de la bomba de agua y descalibración del transmisor de la servoválvula y por consiguiente no se consigue el control del proceso en la planta.

En el algoritmo de control del PID difuso de caudal, se utilizó las mismas constantes  $K_p$ ,  $K_i$  y  $K_d$  del algoritmo de control PID convencional de caudal, cabe recalcar que para obtener un óptimo desempeño de estos dos controladores se utilizó el ajuste empírico de prueba y error, variando así las constantes de las ganancias  $K_p$ ,  $K_i$  y  $K_d$  hasta llegar a un control eficiente del sistema y se obtuvo un óptimo desempeño analizando el control mediante la Integral del Error Absoluto (IAE).

Dentro de los procesos industriales las perturbaciones que ocurren en el control de caudal son de rango moderado, por tal razón, el control implementado en el proyecto maneja estabilidad para eventuales disturbios dentro de un rango máximo del 30%. Por el contrario si se usa perturbaciones mayores se pierde totalmente el control del proceso y la variable de proceso no llega al punto de consigna designado.

El lazo de control PID FUZZY, trabaja en rangos de estabilidad moderados, debido a que cambios extremos del punto de consigna hacen que se pierda controlabilidad del sistema, a causa del efecto WINDUP (saturación del actuador).

Para el diseño del controlador PID difuso de la variable caudal en Simulink/Matlab, se creó un proyecto en Toolbox de conjuntos difusos normalizados, esto quiere decir que permitirá ser utilizado en cualquier sistema de control PID FUZZY.

El control del proceso PID difuso no necesita un modelo del proceso a controlar, lo cual permite mucha facilidad de implementarlo a diferentes sistemas de control ya que tiene un lenguaje muy común.

La comparación de los resultados obtenidos del controlador PID convencional y del controlador PID difuso mediante el índice de desempeño IAE usando el Test de Wilcoxon, demuestra que el PID difuso es mejor en el proceso de control de caudal pues logra disminuir pérdida de energía en el sistema.

Como consecuencia al diseño de controlador PID difuso de la variable caudal en el software Matlab, se verifica que el control PID difuso proporciona una respuesta rápida y estable, cuando éste es sujeto a variaciones del Set Point.

El algoritmo del controlador PID difuso de la variable caudal es factible implementarlo en el software Matlab; ya que tiene el Toolbox FUZZY Logic, donde permite realizar sistemas de lógica difusa de forma automática, ya que tiene programación gráfica en los elementos que conforman al sistema.

La ventaja de usar un control PID difuso mediante el modelo donde la parte Proporcional y Derivativa ingresan al bloque FUZZY, y su salida es sumada con la parte Integral, es que solo se utilizan 25 reglas de fusificación, por el contrario usando el modelo general donde la parte Proporcional, Integral y Derivativa ingresaran al bloque FUZZY, necesitará 125 reglas de fusificación.

## RECOMENDACIONES

Antes de poner en marcha el controlador PID difuso de la variable caudal, se recomienda calibrar el transmisor de caudal de 4-20[mA], para evitar errores de ejecución y funcionamiento del algoritmo programado en el programa Matlab. (Ver Figura 3.2 y Figura 3.3).

Para entender el funcionamiento del controlador PID difuso de caudal se requiere tener conocimientos de los programas Matlab, TIA Portal y KEPServerEX, ya que de lo contrario se tendrá vacíos en lo que se refiere a conexión e intercambio de datos entre estos tres programas.

Con el fin de no dañar ningún transmisor ni actuador de la plataforma de entrenamiento, se debe consultar características de los mismos como, voltaje de alimentación, tipo de señal que reciben y envían, y correcta instalación.

Para comunicar el software Matlab y el PLC Siemens S7-1200 se usa el OPC KEPServerEX, así se evita el instalar drivers de Matlab para conectarse con muchas fuentes de datos como es el PLC Siemens S7-1200.

Al usar transmisores o actuadores hay que conocer el tipo de señal que usa, ya sea señal de tensión o de corriente; para poder generar los parámetros de tensión o corriente en las entradas y salidas análogas del PLC Siemens S7-1200.

Es importante verificar que las direcciones de las memorias usadas del PLC sean las mismas direcciones que las variables de lectura y escritura colocadas en el OPC KEPServerEX, para que no haya un error de comunicación.

Se recomienda que con el transmisor de caudal se tome valores en el software Matlab y observar su grafica de funcionamiento, de modo que se pueda conocer los rangos en los que se debe trabajar.

El transmisor de caudal no se debe instalar cerca de los codos o uniones de la tubería, ya que si se lo hace se obtiene pequeños picos en la señal de salida porque en estas partes de la tubería la dirección del flujo de agua sufre cambios.

Se recomienda verificar el estado de los racores del transmisor de caudal, para evitar derrames de agua y/o que la manguera se salga a causa de la presión del agua.

El sensor de caudal debe estar bien sujeto para evitar vibraciones de la plataforma, debe ir paralelo a la tubería y con una separación adecuada a los transmisores instalados anteriormente.

Los terminales de alimentación y señal del transmisor de caudal, deben ir bien sujetos y ajustados en sus tornillos, para evitar que la lectura de la señal en el PLC sea errónea y hasta en unos casos tener una señal de salida con valor cero.

## REFERENCIAS

- Barrera-Sánchez. (09 de 07 de 2016). *DISEÑO DE CONTROLADORES PID DIFUSOS DE LAS VARIABLES DE NIVEL Y TEMPERATURA DE UNA PLATAFORMA DE ENTRENAMIENTO PARA CONTROL DE PROCESOS CONTINUOS*. Obtenido de <http://dSPACE.ups.edu.ec/handle/123456789/10188>
- Castillo, A. (06 de 07 de 2016). *Historia de la lógica difusa*. Obtenido de <http://www.dma.fi.upm.es>
- Creus, A. (2011). *Instrumentación Industrial*. México: Alfaomega Grupo Editor.
- Dávila, A. (09 de 07 de 2016). *DISEÑO Y CONSTRUCCIÓN DE UN CONTROL DIFUSO DE LAZO CERRADO DE TEMPERATURA 18F452 PARA LOS LABORATORIOS DE TEORÍA DE CONTROL DE LA UNIVERSIDAD POLITÉCNICA SALESIANA*. Obtenido de <http://dSPACE.ups.edu.ec/bitstream/123456789/5371/1/UPS-ST001041.pdf>
- Izaurieta-Saavedra. (09 de 07 de 2016). *Redes Neuronales Artificiales*. Obtenido de *Redes Neuronales Artificiales*: <http://www.uta.cl/charlas/volumen16/Indice/Ch-csaavedra.pdf>
- Kuo, B. (2013). *Sistemas de control automático*. México: Prentice Hall Hispanoamericana.
- Mathworks. (09 de 07 de 2016). *Mathworks*. Obtenido de Matlab: [www.mathworks.com](http://www.mathworks.com)
- Mathworks. (09 de 07 de 2016). *Toolbox FUZZY*. Obtenido de [http://www.mathworks.com/help/FUZZY/index.html?s\\_tid=srchtitle](http://www.mathworks.com/help/FUZZY/index.html?s_tid=srchtitle)
- Moore, D. S. (2000). *Estadística Aplicada Básica*. Barcelona: Antoni Bosch S.A.
- Ogata, K. (2010). *Ingeniería de Control Moderna*. Madrid: Prentice Hall Hispanoamerica.

Ponce, P. (2010). *Inteligencia Artificial con aplicaciones a la ingeniería*. México: Alfaomega Grupo Editor.

Reyes, F. (2012). *Matlab Aplicado a Robótica Y Mecatrónica*. México: Alfaomega Grupo Editor.

Siemens. (09 de 07 de 2016). *Siemens*. Obtenido de PLC S7-1200: <http://www.festo-didactic.com>

## ANEXOS

### Anexo 1. Configuración de la comunicación OPC

1. Click en el icono de Nuevo y se crea un nuevo proyecto
2. Añadir canal con el identificador Ethernet



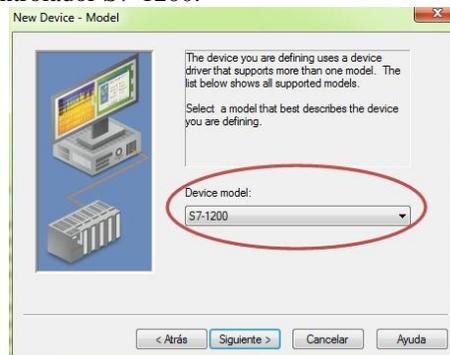
3. Seleccionar el controlador de dispositivo Siemens TCP/IP Ethernet.



4. Seleccionar la tarjeta de red del computador.



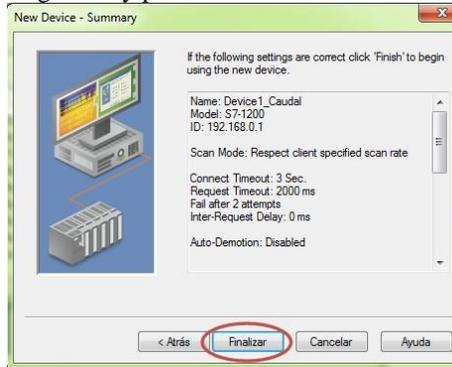
5. Seleccionar el modelo del controlador S7-1200.



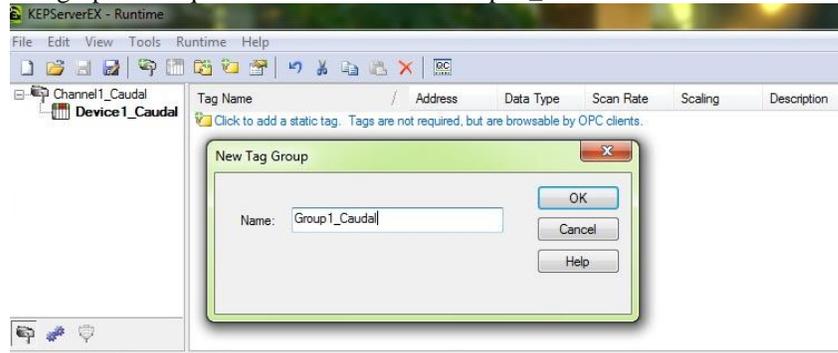
6. Ingresar la dirección IP del PLC, esta dirección IP es la que el PLC tiene por default.



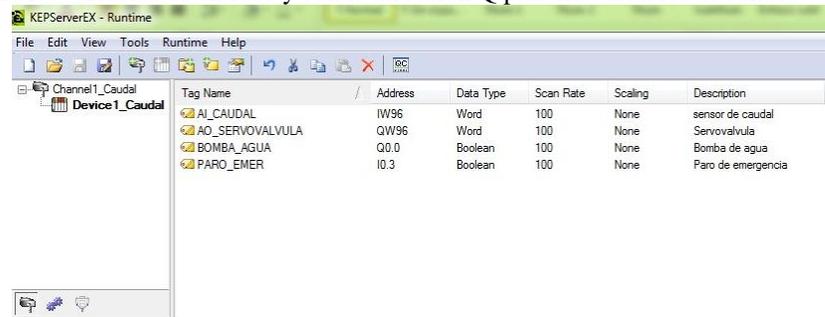
7. Por dos ocasiones dar click en siguiente y por ultimo finalizar.



8. Crear un nuevo grupo de etiquetas con el nombre de Grupo1\_Caudal.



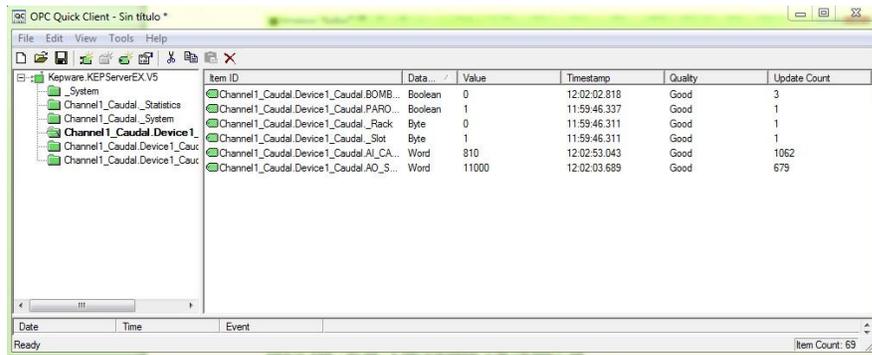
9. Crear las variables correspondientes para la comunicación, las mismas que irán en el PLC, con la dirección I para las variables de entrada y con la dirección Q para las variables de salida.



10. Para verificar la conexión del KEPServerEX con el PLC dar click en el icono de Launch OPC Quick Client.

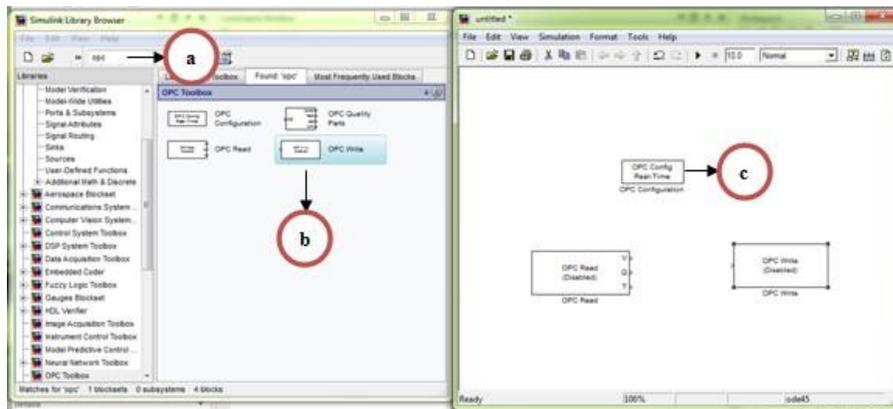


11. Verificar la correcta comunicación de los sensores y actuadores dentro de la ventana de Launch OPC Quick Client, este paso se realiza siempre y cuando el PLC se encuentre en conexión en línea.



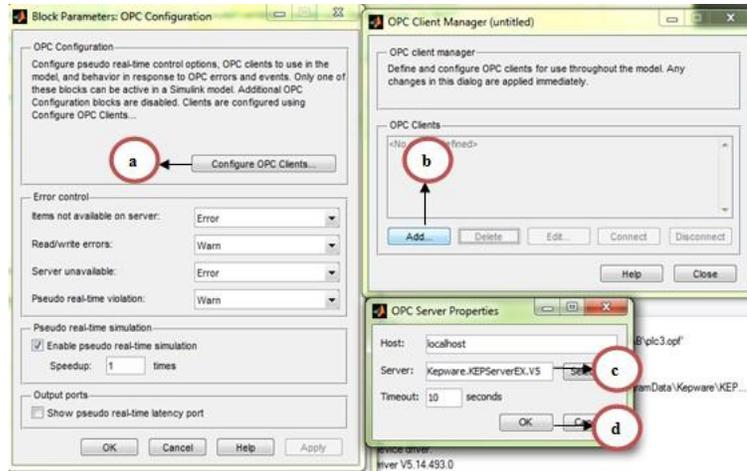
## Anexo 2. Configuración del OPC KEPServerEX

1. Agregar los bloques OPC, seguir los pasos.
  - a. En la ventana de buscar seleccionar la librería OPC
  - b. Seleccionar los bloques de lectura, escritura y configuración.
  - c. Añadir los bloques seleccionados al entorno de Simulink



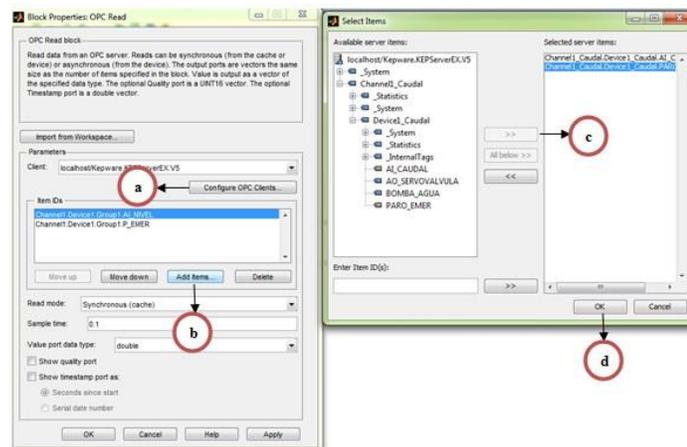
2. En el bloque de OPC Configuración ingresar el OPC cliente creado en el programa KEPServerEX, en el caso del presente proyecto es el PLC S7-1200.

- a. Click en Configure OPC Clients.
- b. Se despliega la ventana OPC Client Manager y hacer click en Add.
- c. En la nueva ventana que se despliega seleccionar el servidor generado en el programa KEPServerEX.
- d. Por último click en OK en cada ventana abierta.



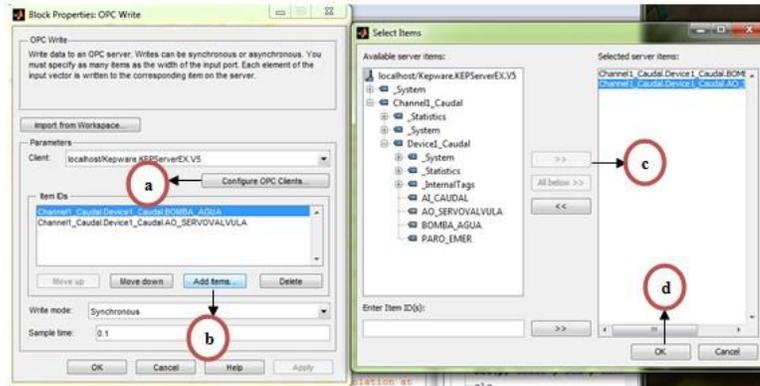
3. En el bloque OPC de lectura añadir las variables que obtiene el PLC del transmisor de caudal y el paro de emergencia.

- Click en Configure OPC Clients y se realiza la configuración de igual manera que los literales b, c y d, del paso 2.
- Click en Add Items
- Agregar las correspondientes variables de lectura.
- Por último click en OK.



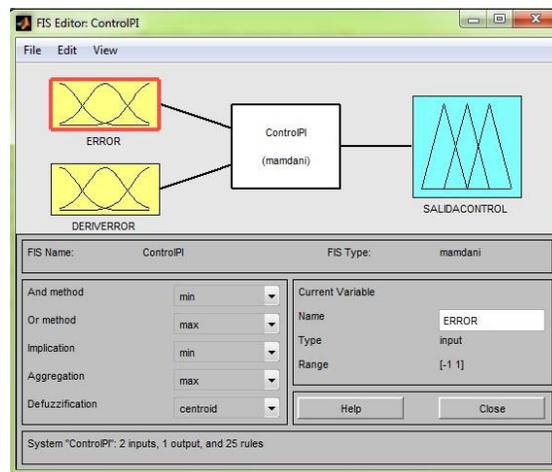
4. En el bloque OPC de escritura añadir las variables de actuadores como la servoválvula y la bomba de agua.

- Click en Configure OPC Clients y se realiza la configuración de igual manera que los literales b, c y d, del paso 2.
- Click en Add Items
- Agregar las correspondientes variables de escritura.
- Por último click en OK.

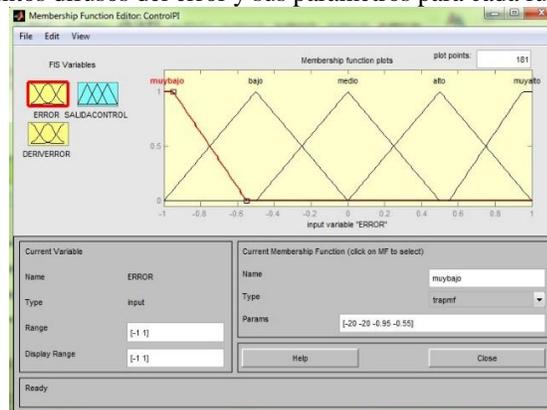


### Anexo 3. Creación de variables lingüísticas y funciones de membrecía

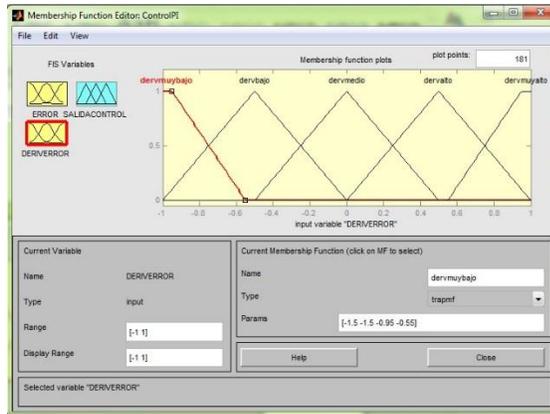
1. Se despliega una ventana y en ella crear las variables lingüísticas.



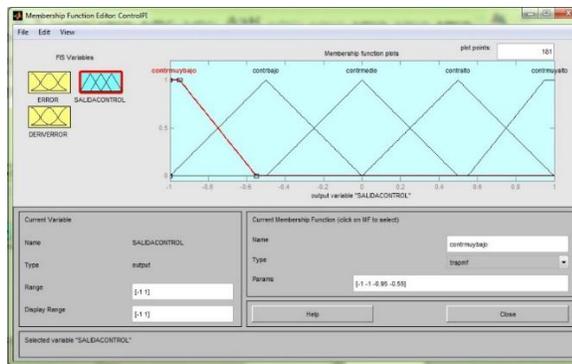
2. Ingresar los conjuntos difusos del error y sus parámetros para cada función de membrecía.



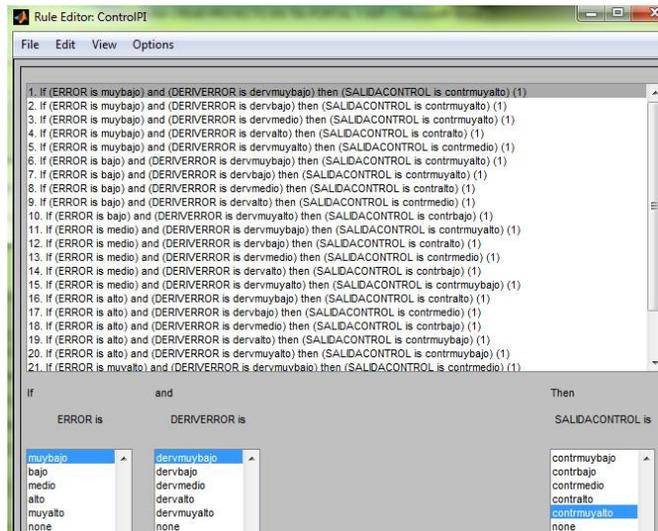
3. Ingresar los conjuntos difusos de la derivada del error y sus parámetros para cada función de membrecía.



- Ingresar los conjuntos difusos de la acción de control y sus parámetros para cada función de membrecía.

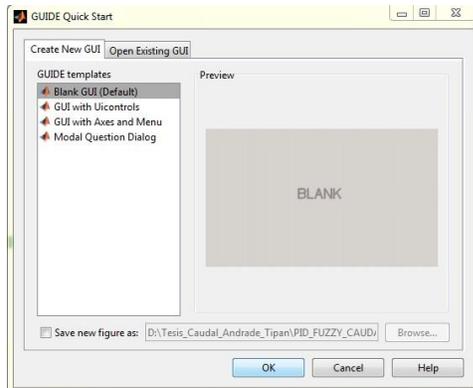


- Ingresar las reglas de inferencia para el control difuso de caudal.

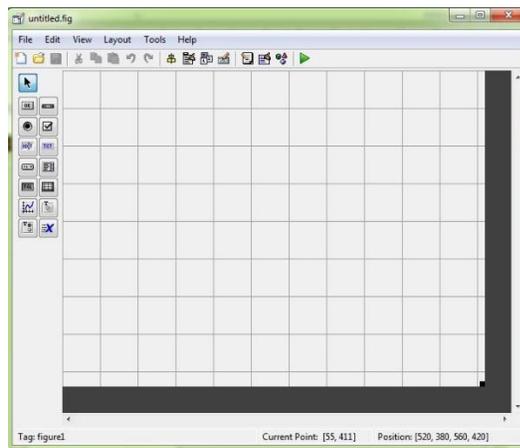


#### Anexo 4. Creación de la interfaz Humano-Máquina (HMI)

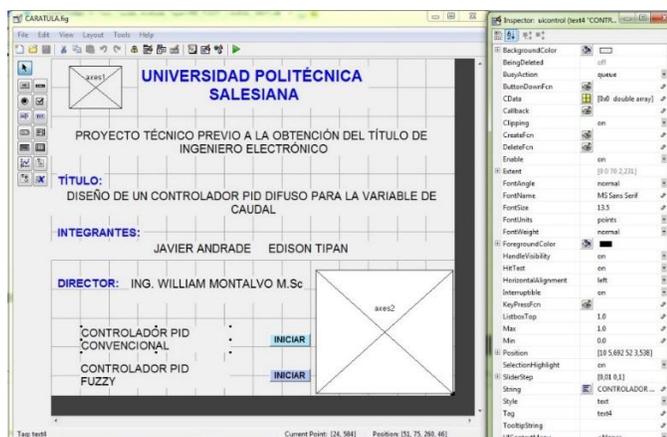
1. Crear un nuevo proyecto ingresando “guide” en la ventana de comandos de Matlab, a continuación se despliega la ventana GUIDE Quick Start, para cambiar la carpeta destino dar click en Browse y elegir donde se desee guardar el proyecto.



2. Al dar click en OK en la ventana Guide se abre una interfaz gráfica en blanco, en el cual se crea el diseño gráfico del control PID FUZZY de la variable caudal.



3. En la interfaz gráfica de usuario añadir los componentes de las diferentes pantallas que se usa y se visualiza en el HMI del control PID FUZZY.



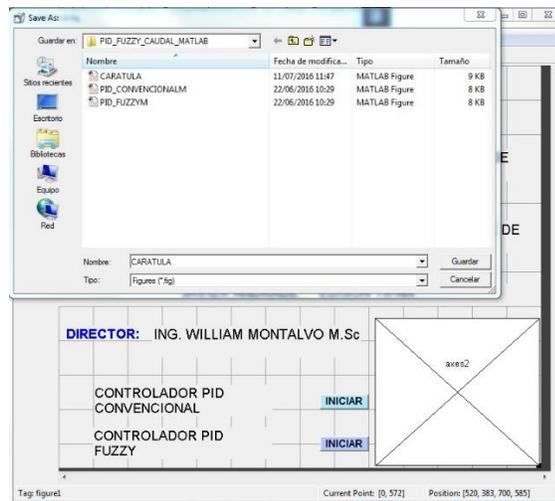
4. Crear otro guide y añadimos los componentes de la pantalla del control PID convencional



5. De la misma manera crear otro guide y añadimos los componentes de la pantalla del control PID FUZZY.

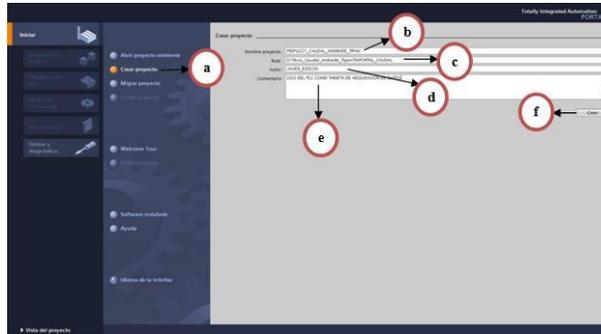


6. Por último al finalizar el diseño de cada una de las pantallas que se visualiza en el control PID convencional y FUZZY, dar click en guardar, ingresamos un nombre y verificar que se guarde con la extensión fig (.fig).

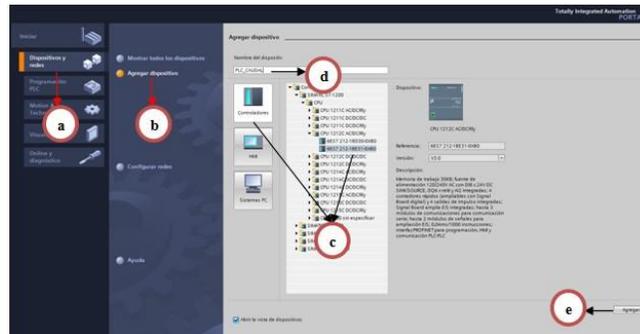


## Anexo 5. Configuración del PLC para que actúe como tarjeta de adquisición de datos

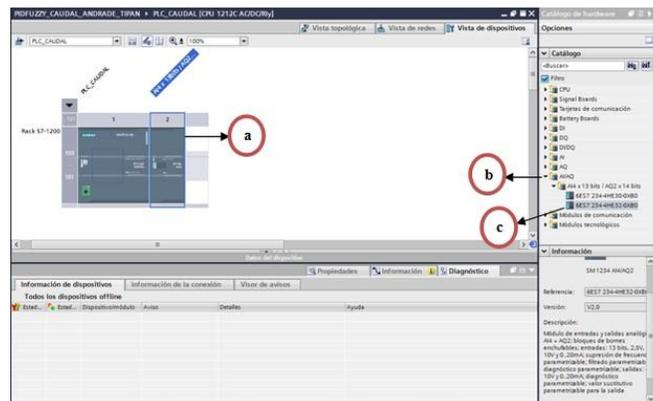
1. Abrir el programa TIA PORTAL V12 y seguir los siguientes pasos.
  - a. Click en crear proyecto
  - b. Ingresar el nombre del proyecto
  - c. Elegir el destino donde se desea guardar el proyecto
  - d. Ingresar el o los autores del proyecto
  - e. Ingresar un comentario del proyecto si se desea
  - f. Click en crear



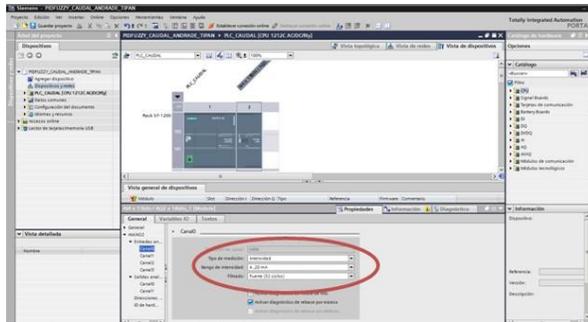
2. Agregar el CPU 1212C AC/DC/Rly, para ello seguir los pasos detallados a continuación.
  - a. Click en la pestaña Dispositivos y redes
  - b. Click en Agregar dispositivo
  - c. Click en la pestaña Controladores y elegir el CPU modelo 1212C AC/DC/Rly.
  - d. Ingresar un nombre al CPU si se desea o dejar el que está por defecto
  - e. Click en agregar



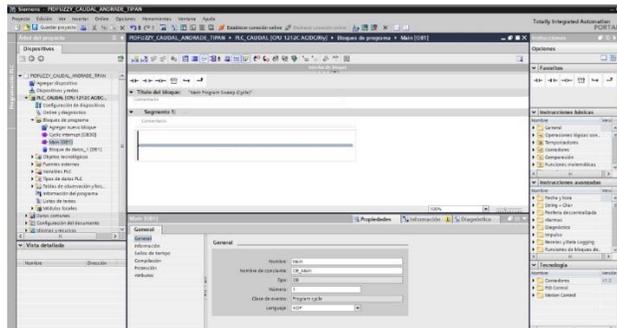
3. Agregar el módulo de expansión de entradas y salidas analógicas, en este caso es el módulo SM-1234, para ello seguir los pasos detallados a continuación.
  - a. Click en el espacio de módulo vacío.
  - b. Seleccionar el módulo de entradas y salidas analógicas
  - c. Doble click en el módulo para que se agregue en el espacio vacío



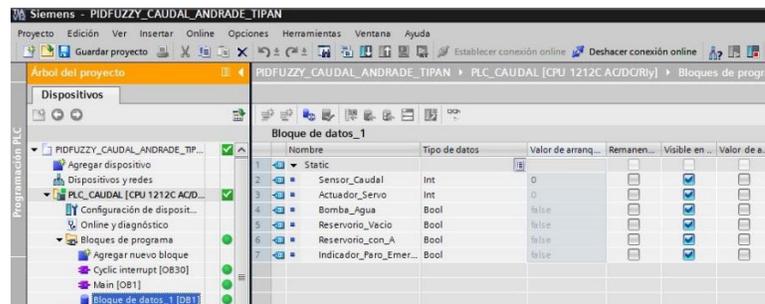
4. Configurar los parámetros ya sean de corriente o tensión de las entradas y salidas analógicas, en el caso del presente proyecto la salida del módulo va ser establecido por corriente, rango de 4-20 [mA], y un filtrado de 32 ciclos.



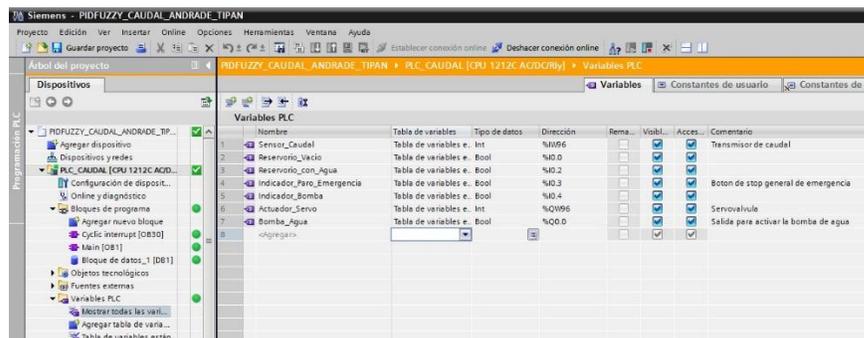
- Para que el proceso del controlador PID difuso funcione correctamente, se procede a crear un programa en blanco dejando los segmentos sin programar.



- Crear un bloque datos e ingresar los nombres y tipos de datos que va recibir y enviar el PLC hacia el programa KEPsServerEX y por consiguiente a Matlab.



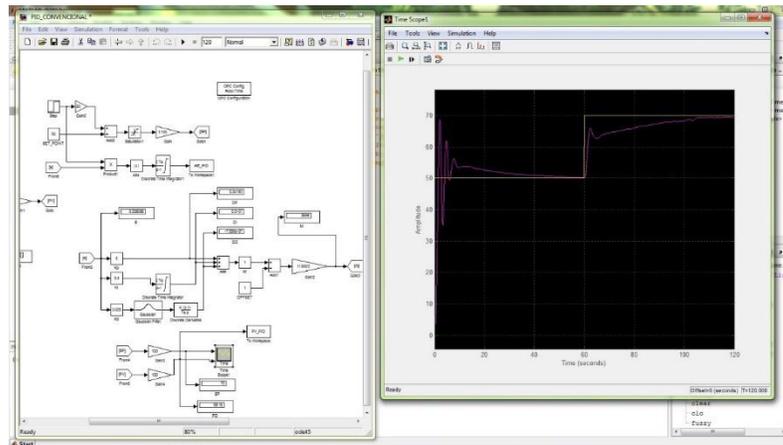
- De la misma manera crear una tabla de variables, ingresando los nombres de las variables y las direcciones de la memoria del PLC donde se guarda los datos.



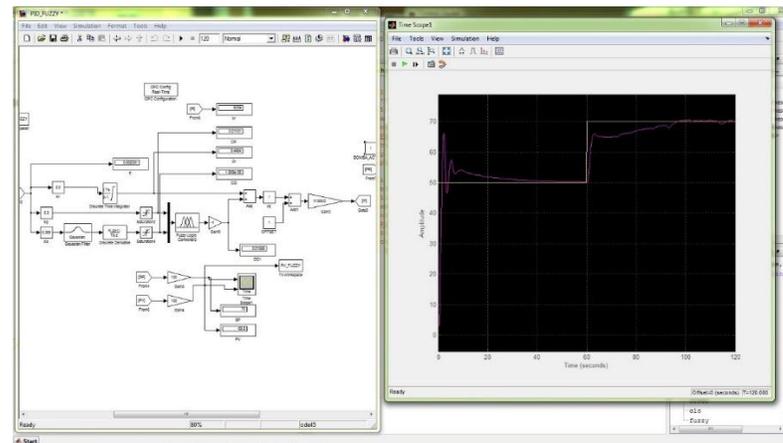
## Anexo 6. Pruebas del IAE en horario matutino

Muestra del PID convencional y del PID difuso tomada en horario matutino

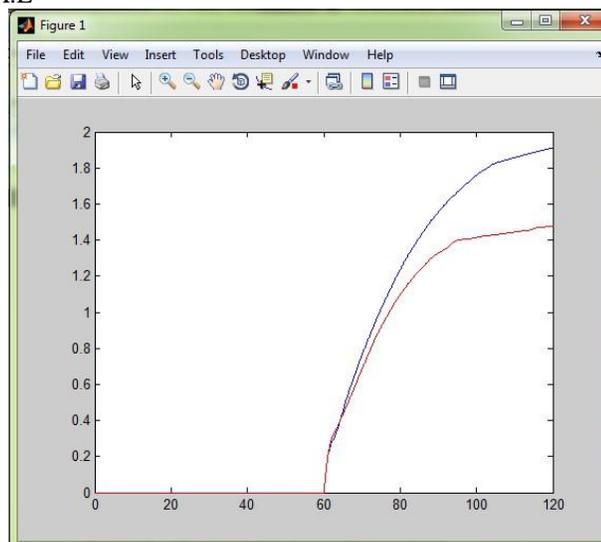
PID Convencional



PID FUZZY

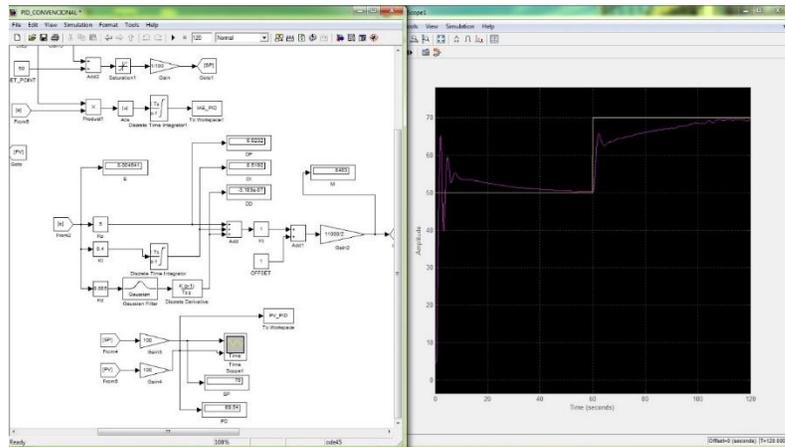


Gráfica de valores de I.A.E

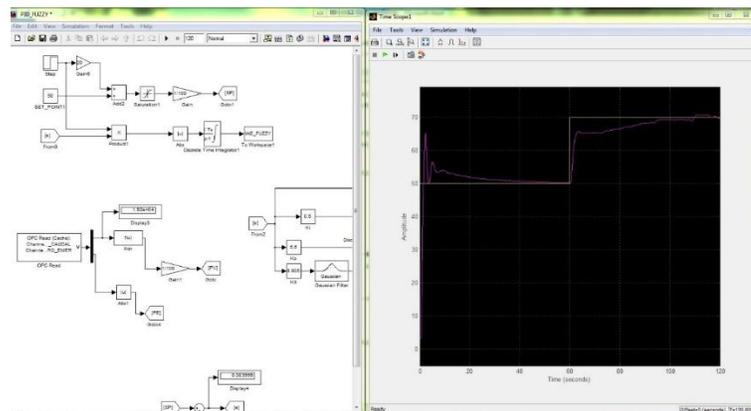


## Anexo 7. Pruebas del IAE en horario vespertino

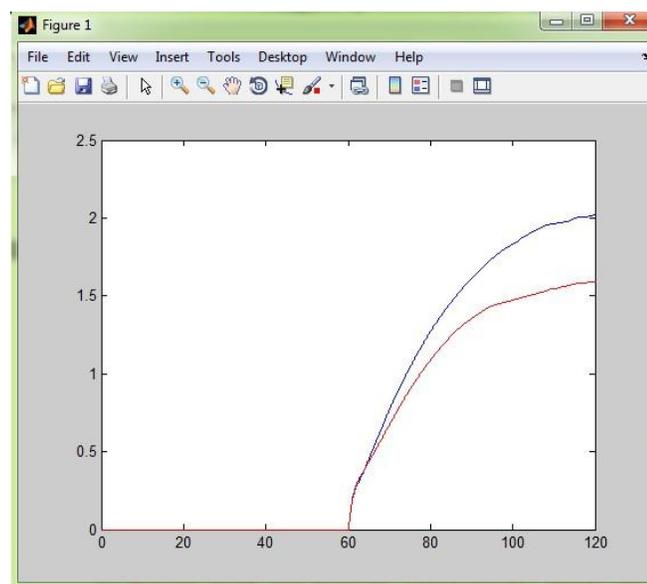
Muestra 1 del PID convencional y del PID difuso tomada en horario vespertino  
PID Convencional



PID FUZZY



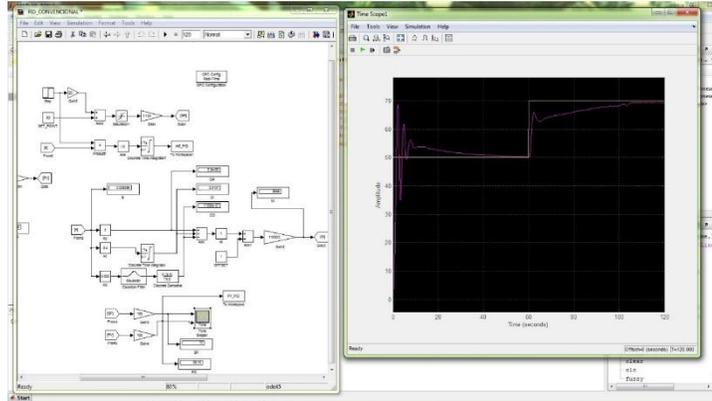
Gráfica de valores de I.A.E



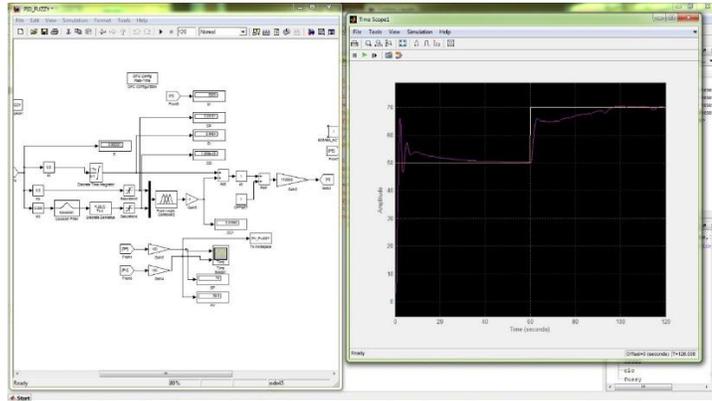
## Anexo 8. Pruebas del IAE en horario nocturno

Muestra 1 del PID convencional y del PID difuso tomada en horario nocturno

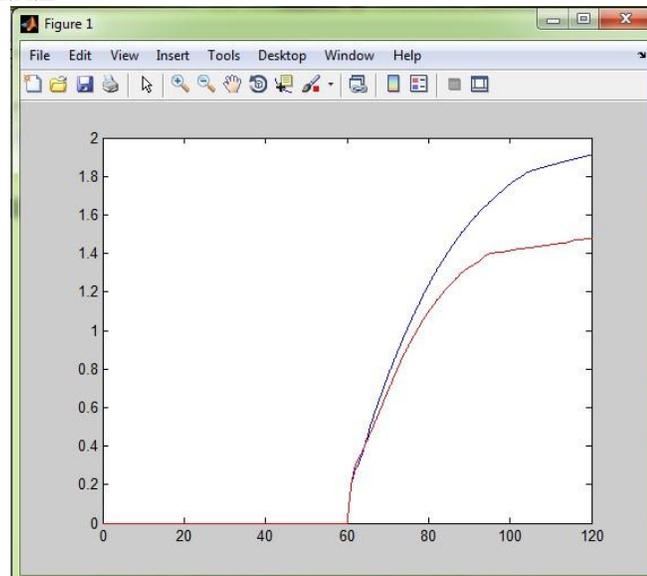
PID Convencional



PID FUZZY



Gráfica de valores de I.A.E



**Anexo 9. Prácticas de laboratorio para el control PID difuso de caudal**

	PRACTICA 1
	COMUNICACIÓN ENTRE EL PLC SIEMENS S7-1200 Y EL SOFTWARE MATLAB
<b>LABORATORIO</b>	CONTROL
<b>CARRERA</b>	ELECTRÓNICA
<b>SEDE</b>	QUITO

**DATOS INFORMATIVOS**

- a) MATERIA:
- b) No DE PRACTICA: 1
- c) NÚMERO DE ESTUDIANTES:
- d) NOMBRE DEL INSTRUCTOR:

**DATOS DE LA PRÁCTICA**

- a) **TEMA:**  
Comunicación entre el PLC Siemens S7-1200 y el software Matlab para poner en funcionamiento el proceso de control de Caudal.
- b) **OBJETIVO GENERAL**  
Comunicar el PLC Siemens S7-1200 con el software Matlab por medio del programa KEPServerEX para manejar variables físicas del PLC desde Matlab y viceversa.
- c) **OBJETIVOS ESPECIFICOS**
  - Abrir el programa creado para el control de caudal hecho en el software TIA PORTAL.
  - Poner en funcionamiento el KEPServerEX con el proyecto creado para el control de caudal y visualizar la ventana OPC Quick Client y
  - Verificar el estado de la conexión entre el PLC Y el KEPServerEX,
- d) **MARCO TEÓRICO**

**KEPServerEx**

Este software permite enlazar el Matlab con el PLC s7-1200, en este caso en particular el PLC actuará únicamente como una tarjeta de adquisición de datos.

El KEPServerEX permite el intercambio de datos entre clientes OPC y PLC Siemens S7-200, S7-300, S7-400, S7-1200, este programa utiliza protocolos TCP/IP Ethernet.

**OPC Toolbox**

La herramienta OPC permite una conexión con los servidores OPC DA y OPC HDA, los cuales le dan acceso a transferir datos históricos OPC desde Matlab y Simulink; con esta herramienta se puede leer, escribir y registrar datos OPC de dispositivos como los sistemas de control distribuido, supervisión y adquisición de datos y PLC (Controlador Lógico Programable), siempre y cuando cumplan con el estándar de comunicación OPC Foundation Data Access.

**e) PROCEDIMIENTO**

**Implementación del controlador PID difuso:**

Para la puesta en marcha del controlador PID convencional y PID difuso, implementada en la plataforma de entrenamiento de procesos continuos, se debe seguir los pasos detallados a continuación:

1. Abrir el programa TIA PORTAL con el proyecto creado del caudal y realizar lo siguiente, como se muestra en la Figura 1.
  - a) Click en compilar
  - b) Click en cargar
  - c) Establecer conexión online.

## Establecimiento de la conexión en TIA PORTAL

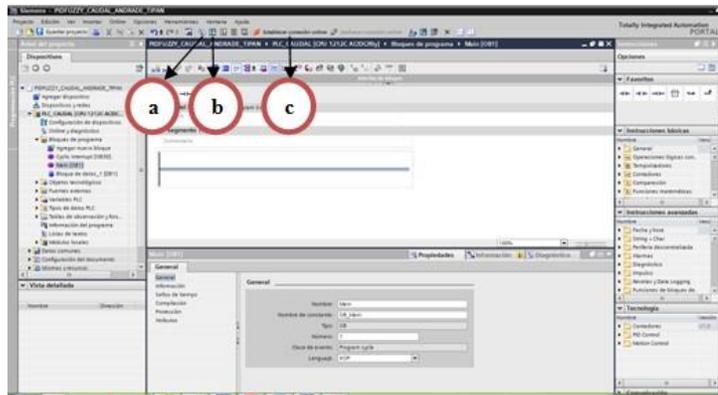


Figura 1. Pasos para establecer conexión online del PLC

Fuente: (Programa TIA PORTAL)

2. Abrir el programa KEPServerEX con el proyecto creado anteriormente, abrir la ventana OPC Quick Client y verificar el estado de la conexión entre el PLC Y el KEPServerEX, como se observa en la Figura 2.

## Conexión entre PLC y KEPServerEX

Item ID	Data	Value	Timestamp	Quality	Update Count
Channel1_Caudal Device1_Caudal BOMB	Boolean	0	12:02:02.818	Good	3
Channel1_Caudal Device1_Caudal PARO	Boolean	1	11:59:46.337	Good	1
Channel1_Caudal Device1_Caudal Rack	Byte	0	11:59:46.311	Good	1
Channel1_Caudal Device1_Caudal Slot	Byte	1	11:59:46.311	Good	1
Channel1_Caudal Device1_Caudal AI_CA	Word	810	12:02:53.643	Good	1062
Channel1_Caudal Device1_Caudal AO_S	Word	11000	12:02:03.689	Good	679

Figura 2. Estado de la conexión entre el PLC y el KEPServerEX.

Fuente: (Programa KEPServerEX)

	PRACTICA 2
	CONTROL PID DIFUSO DE LA VARIABLE CAUDAL EN MATLAB Y APLICACIÓN DEL IAE
LABORATORIO	CONTROL
CARRERA	ELECTRÓNICA
SEDE	QUITO

#### DATOS INFORMATIVOS

- MATERIA:
- No DE PRACTICA: 2
- NÚMERO DE ESTUDIANTES:
- NOMBRE DEL INSTRUCTOR:

#### DATOS DE LA PRÁCTICA

- TEMA:**  
Funcionamiento del controlador PID difuso de la variable de caudal en el software Matlab para el control de procesos continuos y aplicación del IAE.
- OBJETIVO GENERAL**  
Poner en funcionamiento el controlador PID difuso de la variable de caudal en el software Matlab para el control de procesos continuos.
- OBJETIVOS ESPECÍFICOS**
  - Ejecutar el programa de control PID difuso en el software Matlab para control de procesos de caudal.
  - Comparar el desempeño del controlador PID convencional con el control PID difuso mediante el IAE.
- MARCO TEÓRICO**

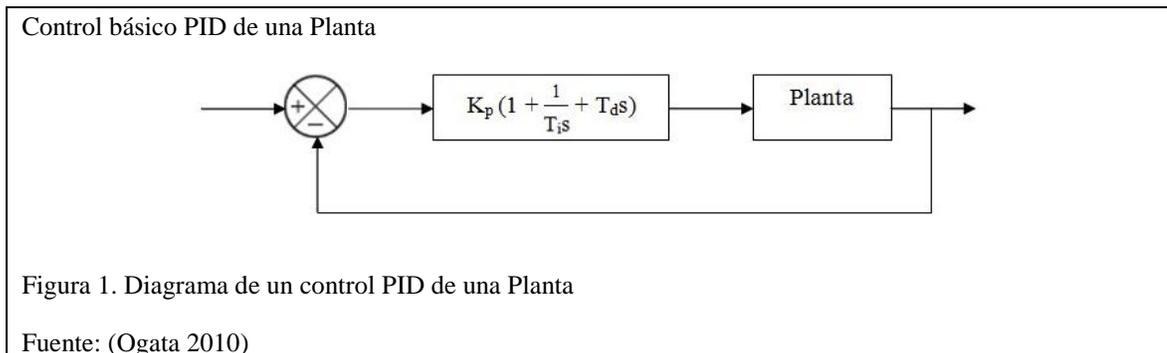
#### Controlador Proporcional Integral Derivativo - PID

Son los más usados en plantas industriales debido a su tipo de estructura la cual es simple.

Este tipo de control es limitado, debido a que es eficiente generalmente en sistemas estables.

Si la planta es tan complicada que no se hace posible obtener su modelo matemático, se debe recurrir a la sintonía de los controladores PID. Existen reglas de sintonización para controladores PID, según Ziegler y Nichols significa dar valores a  $K_p$ ,  $T_i$  y  $T_d$ , en las plantas industriales se utilizan estas reglas cuando no se conocen sus modelos matemáticos. (Ogata, 2010).

Las reglas de sintonía de Ziegler-Nichols dan una estimación razonable de los parámetros del controlador y proporcionan un punto de partida para una sintonía fina, en lugar de dar los parámetros  $K_p$ ,  $T_i$  y  $T_d$  en un único intento. En la Figura 1 se muestra la estructura del control PID de una planta.



## Lógica Difusa

Los controles que se encuentran basados en lógica difusa emulan el razonamiento y pensamiento del cerebro humano, las aplicaciones de la lógica difusa se encuentran en múltiples áreas como por ejemplo el desarrollo tecnológico de los electrodomésticos, también se puede encontrar en programas computacionales para tomar decisiones correctas en el momento indicado, y con ello se han extendido a diversos campos específicos como las cámaras de video, seguridad en base al monitoreo de terremotos. (Castillo 2016), menciona que la lógica difusa también conocida como borrosa, es una rama de la Inteligencia Artificial, en la cual las proposiciones pueden ser representadas con diferentes grados entre lo verdadero o lo falso. Este tipo de lógica se basa en reglas basadas en el conocimiento humano, por lo que tolera errores e incluso puede analizar estos para la resolución de problemas o toma de acciones en un determinado sistema de control.

## Toolbox FUZZY

El Toolbox FUZZY el cual tiene aplicaciones y funciones para el análisis y diseño de sistemas basados en lógica difusa; este toolbox guía paso a paso a través del proceso de diseño de los sistemas de inferencia difusos.

Al diseñar estos sistemas difusos es más fácil construir un sistema gráficamente, que trabajar utilizando únicamente la caja de herramientas de la lógica difusa mediante la línea de comandos. Existen cinco herramientas primarias del Toolbox FUZZY las cuales son:

- Editor de reglas
- Editor de FIS (Sistema de Inferencia FUZZY)
- Editor de funciones membrecía
- Visualizador de reglas
- Visualizador de las superficies

## Integral del error absoluto (IAE)

Integral del error absoluto (IAE) es la energía perdida en el control de un proceso. Cuando se obtiene un valor bajo de este índice se deduce que se la variable de proceso estuvo muy cerca del valor deseado.

La IAE matemáticamente se determina por las siguientes ecuaciones:

$$IAE = \int_0^{\infty} |e(t)| dt$$
$$e(t) = r(t) - y(t)$$

Donde:  $y(t)$  es la salida del sistema de control y  $r(t)$  es el valor esperado.

### e) PROCEDIMIENTO

1. Abrir el programa Matlab, en la ventana de comandos escribir "CARATULA", y se desplegará la pantalla principal desde donde se controlará el proceso, como se muestra en la Figura 2.

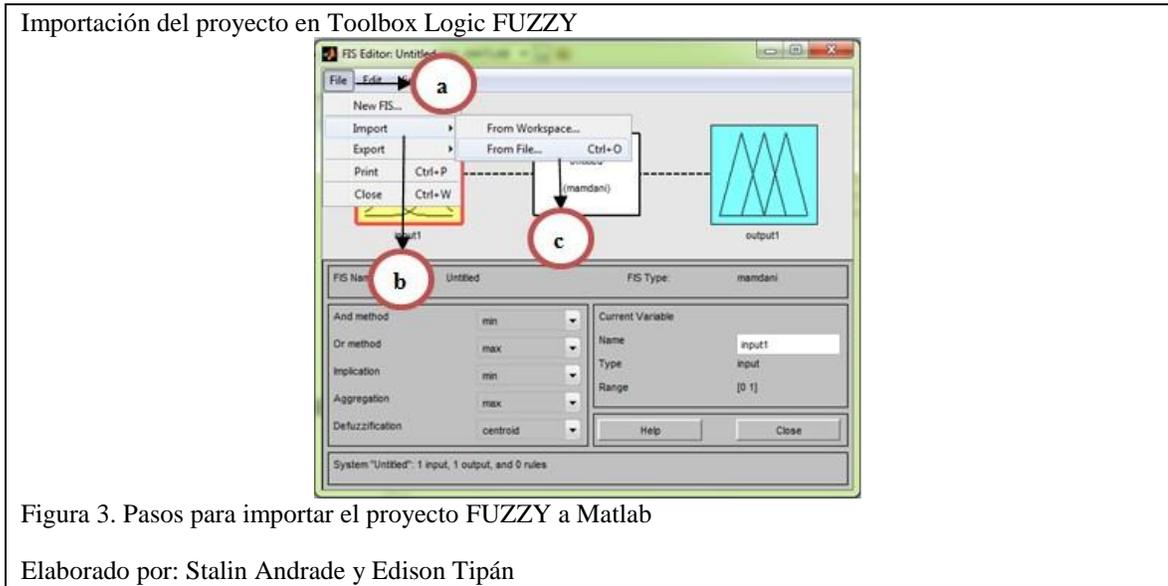
Pantalla del HMI



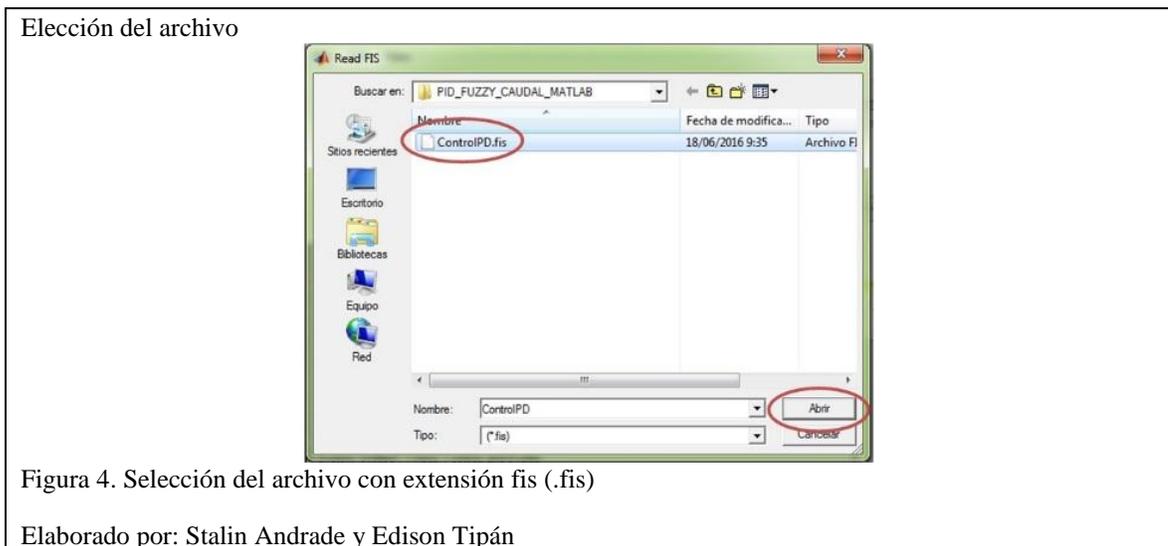
Figura 2. Pantalla principal del HMI

Elaborado por: Stalin Andrade y Edison Tipán

2. Abrir la ventana de editor FUZZY, escribiendo “FUZZY” en la ventana de comandos de Matlab y seguir los pasos que se detallan a continuación y como se observa en la Figura 3.
  - a) Click en File.
  - b) Click en Import.
  - c) Click en From file



3. Seleccionar el archivo con extensión fis (.fis), creado anteriormente y dar click en Abrir como se muestra en la Figura 4.



4. Para exportar el proyecto FUZZY al workspace de Matlab, en la ventana de editor FUZZY seguimos los siguientes pasos, como se observa en la Figura 5.
  - a) Click en File
  - b) Click en Export
  - c) Click en To Workspace

### Exportación del archivo en Toolbox Logiz FUZZY

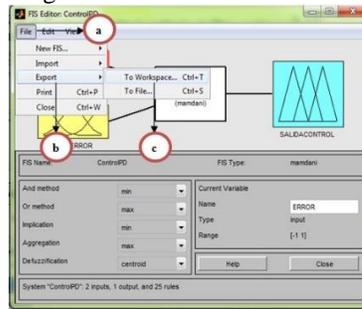


Figura 5. Pasos para exportar el proyecto FUZZY al workspace de Matlab

Elaborado por: Stalin Andrade y Edison Tipán

5. En la nueva ventana que se despliega dar click en OK y cerrar todas las ventanas emergentes abiertas del editor FUZZY, como se muestra en la Figura 6.

### Confirmación del proyecto en Toolbox Logiz FUZZY

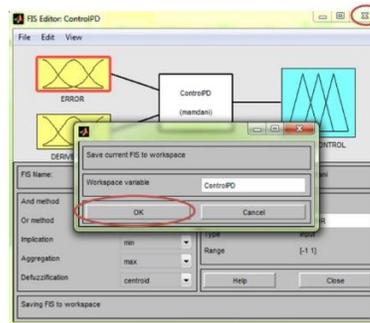


Figura 6. Exportar el proyecto FUZZY al workspace de Matlab.

Elaborado por: Stalin Andrade y Edison Tipán

6. La pantalla principal desplegada con el nombre de CARATULA, permite elegir el control que se desea ejecutar, ya sea PID convencional o PID difuso, como se observa en la Figura 7.

### MATLAB principal



Figura 7. Pantalla principal del HMI

Elaborado por: Stalin Andrade y Edison Tipán

7. Se selecciona el proceso del controlador PID convencional como se observa en la Figura 8.

Pantalla del controlador PID convencional



Figura 8. Interfaz gráfica del controlador PID convencional

Elaborado por: Stalin Andrade y Edison Tipán

Se selecciona el proceso del controlador PID difuso como se observa en la Figura 9.

Pantalla del controlador PID difuso



Figura 9. Interfaz gráfica del controlador PID FUZZY

Elaborador por: Stalin Andrade y Edison Tipán

### IAE del PID convencional y del PID difuso

Para obtener las gráficas y valores correspondientes a la integral del error absoluto (IAE), se debe seguir los siguientes pasos:

- Ejecutar los algoritmos en forma individual como se muestra en la Figura 10, es decir ejecutarlos fuera de línea y no directamente desde la interfaz gráfica creada.

PID convencional y PID FUZZY

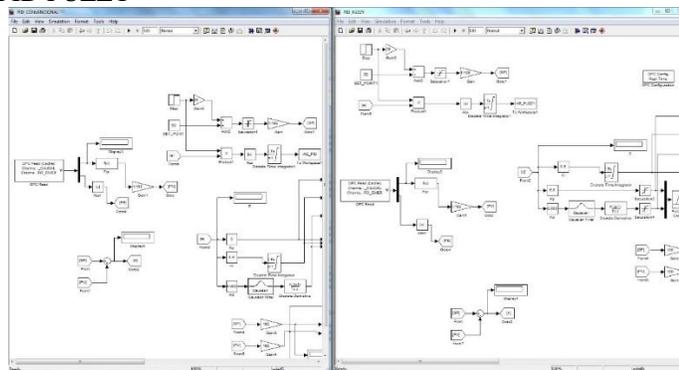


Figura 10. Ejecución fuera de línea del PID convencional y PID FUZZY.

Elaborado por: Stalin Andrade y Edison Tipán

Se debe colocar el mismo tiempo en ambos programas, en este caso se colocó 120[s], en la Figura 11 se observa ejecutándose en tiempo real el controlador PID convencional.

### Ejecución del controlador PID convencional

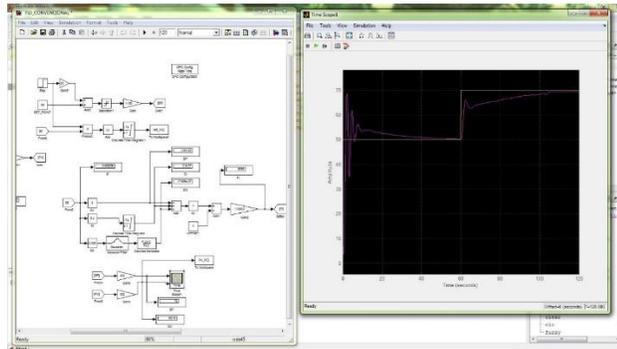


Figura 11. Ejecución en tiempo real del controlador PID convencional  
Elaborado por: Stalin Andrade y Edison Tipán

En la Figura 12 se observa ejecutándose en tiempo real el controlador PID FUZZY.

### Ejecución del controlador PID FUZZY

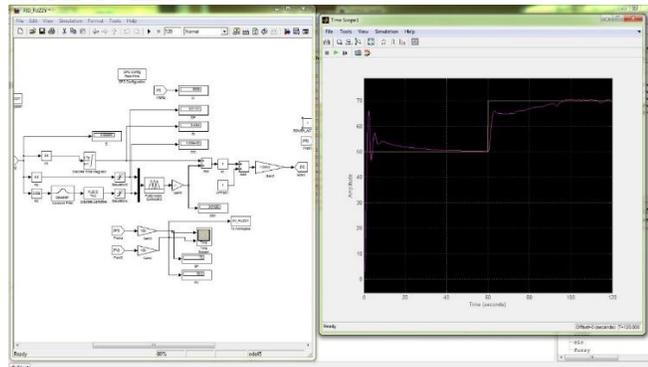


Figura 12. Ejecución en tiempo real del controlador PID FUZZY  
Elaborado por: Stalin Andrade y Edison Tipán

- b) Una vez ejecutado los dos programas por separado dirigirse a la ventana de comandos de Matlab e ingresar los comandos que se muestran en la Figura 13.

### Comandos de comparación de los dos controladores

```
Command Window
New to MATLAB? Watch this Video, see Demos, or read Getting Started.

>> p = plot(IAE_PID.time, IAE_PID.data)

p =

    4.6030e+03

>> hold on
>> p = plot(IAE_FUZZY.time, IAE_FUZZY.data)

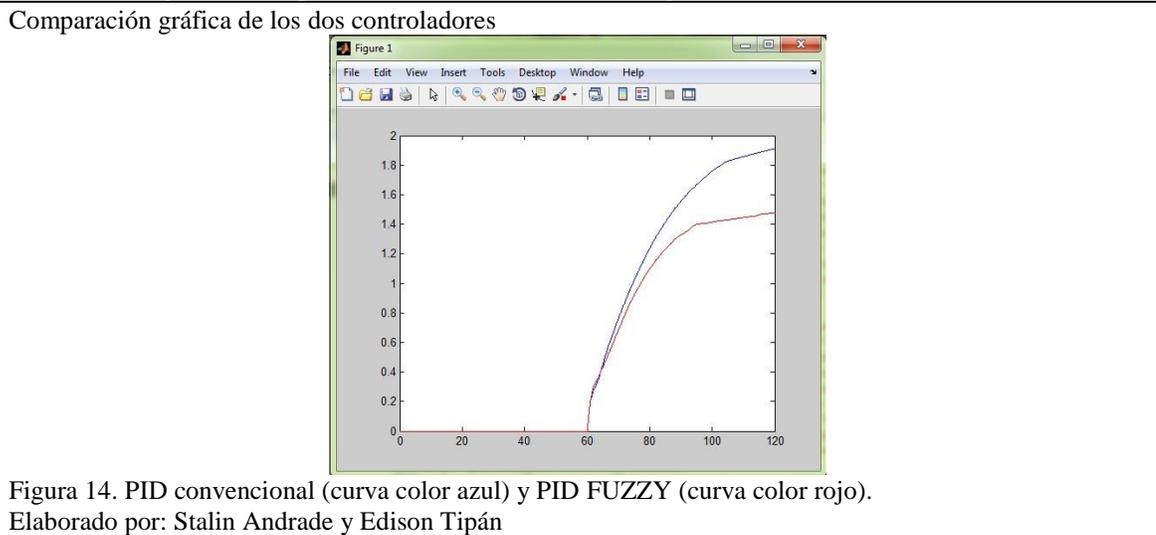
p =

    4.6040e+03

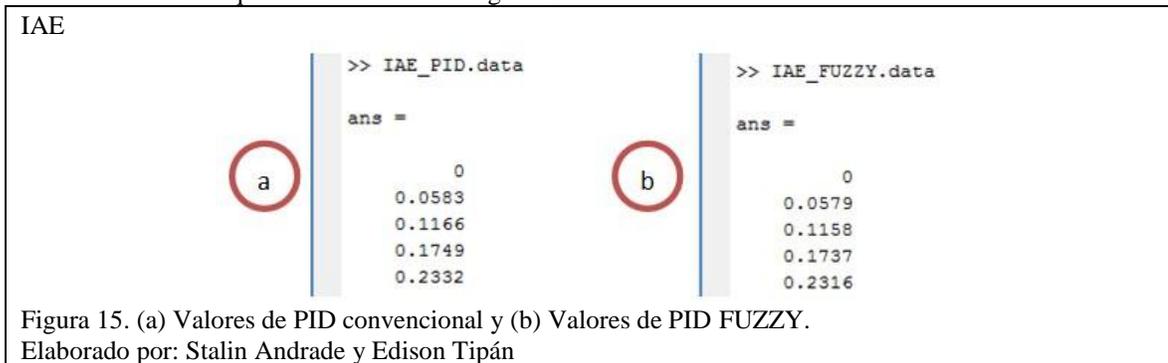
>> set(p, 'Color', 'red', 'LineWidth', 1)
```

Figura 13. Comandos para obtener la gráfica comparativa entre los dos controladores  
Elaborado por: Stalin Andrade y Edison Tipán

Se obtiene las gráficas comparativas como se observa en la Figura 14.



- c) Para obtener los valores de la integral del error absoluto de cada uno de los controladores PID convencional y PID FUZZY dirigirse a la ventana de comandos de Matlab e ingresar los comandos que se muestran en la Figura 15.



Al ingresar cada comando para obtener el valor de IAE se despliega 120 valores porque el tiempo de ejecución que se ingreso fue de 120[s], por tal razón para hacer la comparación entre el controlador PID convencional y PID FUZZY se elige el último valor desplegado en cada caso respectivamente.