

ESCUELA POLITÉCNICA DEL EJÉRCITO

DEPARTAMENTO DE ELÉCTRICA Y ELECTRÓNICA

CARRERA DE INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN
Y CONTROL

PROYECTO DE GRADO PARA LA OBTENCIÓN DEL TÍTULO DE
INGENIERÍA

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA DE
ENTRENAMIENTO EN CONTROL DIFUSO UTILIZANDO
FUZZYCONTROL++

ANDREA PAOLA ORELLANA SALAZAR

Sangolquí-Ecuador

2008

CERTIFICACIÓN

Certificamos que el presente Proyecto de Grado titulado “Diseño e implementación de un sistema de entrenamiento en control difuso utilizando FuzzyControl++”, ha sido desarrollado en su totalidad por el señorita: Orellana Salazar Andrea Paola, y elaborado bajo nuestra dirección como requisito previo para la obtención del Título en Ingeniería Electrónica, automatización y Control.

Sangolquí, 12 de Diciembre del 2008.

Ing. Hugo Ortiz
DIRECTOR

Ing. Rodolfo Gordillo
CODIRECTOR

RESUMEN

Este sistema de entrenamiento en control difuso se basa en la utilización del software FuzzyControl++ de Siemens.

Mediante investigación, estudio y análisis de la lógica difusa, se pone a disposición los fundamentos necesarios para la creación y manejo de sistemas difusos. Se explica de manera clara y sencilla como trabaja FuzzyControl++.

Se analiza el hardware requerido y sus características, de la misma manera se proporciona una descripción del software: Step7 Professional, WinCC y FuzzyControl++, destacando sus herramientas principales.

Se describen problemas encontrados durante el funcionamiento de bloques de control FuzzyControl++ incluidos en WinCC, para los cuales se brinda las soluciones respectivas.

Se incluye una aplicación de control de temperatura que abarca todos los conceptos antes mencionados. Se detalla su programación así como su simulación e implementación. Se recopila los datos obtenidos de las pruebas y se los analiza.

Para finalizar se exponen conclusiones y recomendaciones que serán de gran ayuda para quienes opten por este software de control.

DEDICATORIA

Dedico mi tesis a las personas que han estado a mi lado todo el tiempo, que nunca me han abandonado y que siempre me han tendido la mano en los malos momentos.

Con todo mi amor a mi familia.

Andrea Paola Orellana Salazar.

AGRADECIMIENTOS

Agradezco a mis padres, ya que sin su apoyo y sacrificio nada de esto sería posible.

A mis tres hermanos y a mis amigos, que me han ayudado y motivado durante todos estos años.

A Samuelito por haber nacido.

Y a Dios por darme todo lo mejor, por enseñarme, por cuidarme y por amarme tanto.

Andrea Paola Orellana Salazar.

PRÓLOGO

Debido al auge de la lógica difusa dentro de la electrónica en los últimos años, es imprescindible el estudio de la misma en el Ecuador.

La muestra de herramientas como FuzzyControl++ de Siemens, impulsará y apoyará la investigación de esta clase de tecnologías.

El presente proyecto engloba los fundamentos básicos de la lógica difusa, la creación de sistemas difusos con dicho software, y la utilización de programas de soporte para el desarrollo de este tipo de sistemas. Incluye un aporte muy importante, referido a problemas encontrados en el software.

Brinda un modelo práctico de la utilización de FuzzyControl++, exponiendo y analizando los resultados que se obtuvieron de dicha implementación.

Para finalizar, se señala todo lo concluido con este trabajo y se dan algunas recomendaciones para obtener buenos resultados.

ÍNDICE DE CONTENIDO

	Pág.
GLOSARIO	7
CAPÍTULO 1	10
INTRODUCCIÓN	10
1.1 ANTECEDENTES.....	10
1.2 JUSTIFICACIÓN E IMPORTANCIA.....	11
1.3 ALCANCE DEL PROYECTO.....	11
1.4 OBJETIVOS.....	12
1.4.1 General	12
1.4.2 Específicos.....	12
1.5 DESCRIPCIÓN GENERAL DEL PROYECTO.....	12
CAPÍTULO 2.....	14
LÓGICA DIFUSA.....	14
2.1 HISTORIA DE LA LÓGICA DIFUSA.....	14
2.2 FUNDAMENTOS	16
2.2.1 Conjuntos no difusos y difusos.....	16
2.2.2 Función de Pertenencia	17
2.2.3 Tipos de Funciones de Pertenencia.....	17
2.2.4 Operaciones entre Conjuntos Difusos.....	19
2.3 CONTROL DIFUSO.....	21
2.3.1 Características del Control Difuso.....	21
2.3.2 Función de un Sistema de Control Difuso.....	22
2.3.3 Sistema de control difuso de Mamdani	23
2.3.4 Aplicaciones del Control Difuso.....	25
2.2.5 Ventajas y desventajas del Control Difuso.....	26
2.2.6 Ejemplo de un Control Difuso.....	27
CAPÍTULO 3.....	34
HERRAMIENTAS DE HARDWARE	34
3.1 PLC SIMATIC S7-300.....	34

3.1.1 CPU 315F-2 PN/DP	36
3.1.2 Comunicación	38
3.2 TIPOS DE MÓDULOS	39
3.3 TARJETA DE MEMORIA MICRO (MMC)	41
CAPÍTULO 4	43
HERRAMIENTAS DE SOFTWARE	43
4.1 STEP7 PROFESSIONAL.....	43
4.2 WINCC (Windows control center)	48
4.3 FUZZYCONTROL++.....	52
4.3.1 FUZZYCONTROL++ V5: PROBLEMAS Y SOLUCIONES.....	54
CAPÍTULO 5	58
SISTEMA DE ENTRENAMIENTO	58
5.1 DEFINICIÓN DE LA APLICACIÓN	58
5.2 PROGRAMACIÓN	60
5.3 SIMULACIÓN	68
5.4 IMPLEMENTACIÓN.....	74
CAPÍTULO 6	79
PRUEBAS Y RESULTADOS	79
6.1 PRUEBAS.....	79
6.2 RESULTADOS	86
CAPÍTULO 7	88
CONCLUSIONES Y RECOMENDACIONES	88
7.1 CONCLUSIONES	88
7.2 RECOMENDACIONES.....	89
ANEXOS	90
REFERENCIAS BIBLIOGRÁFICAS	131

ÍNDICE DE TABLAS

	Pág.
Tabla. 2.1. Interpretación de la agregación bajo varias implicaciones.....	24
Tabla. 3.1. Datos técnicos CPU 315F-2 PN/DP.....	37
Tabla. 3.2. Tipos de módulos de I/O.....	40
Tabla. 3.3. Tamaños de memoria para las MMC.....	41
Tabla. 5.1. Funciones de pertenencia tanto para la entrada sensor y set_point ..	62
Tabla. 5.2. Valores de las funciones de pertenencia para la salida.....	63
Tabla. 5.3. Direccionamiento de objetos en WinCC.....	66
Tabla. 5.4. Valores del sensor simulados con el Tag Simulation.....	70

ÍNDICE DE FIGURAS

	Pág.
Figura. 2.1. Función de Pertenencia Unitaria o Singleton.....	18
Figura. 2.2. Función de Pertenencia.....	18
Figura. 2.3. Conjunto Difuso A.....	19
Figura. 2.4. Conjunto Difuso B.....	20
Figura. 2.5. Unión entre A y B.....	20
Figura. 2.6. Intersección entre A y B.....	20
Figura. 2.7. Complemento de A.....	21
Figura. 2.8. Configuración básica de un Sistema Difuso Mamdani.....	23
Figura. 2.9. Función de Pertenencia de la variable Distancia.....	30
Figura. 2.10. Función de Pertenencia de la variable Velocidad.....	30
Figura. 2.11. Señal de salida Aceleración.....	31
Figura. 3.1. Autómata programable SIMATIC S7-300.....	35
Figura. 3.2. CPU 315F-2 PN/DP.....	37
Figura. 3.3. Ejemplo de configuración de una red de trabajo para el CPU 315F-2 PN/DP.....	39
Figura. 3.4. Módulos de I/O.....	40
Figura. 4.1. Administrador SIMATIC.....	46
Figura. 4.2. Explorador WinCC.	51
Figura. 4.3. Ventanas de FuzzyControl++	54
Figura. 4.4. Objeto de control FuzzyControl++.....	55
Figura. 4.5. Ventana de Propiedades de FuzzyControl++.....	56
Figura. 5.1. Diagrama del sistema controlador difuso.....	59
Figura. 5.2. Vista general de la ventana de programación.....	60
Figura. 5.3. Funciones de pertenencia para la entrada sensor.....	61
Figura. 5.4. Funciones de pertenencia para la entrada <i>set_point</i>	62
Figura. 5.5. Funciones de pertenencia para la salida.....	63
Figura. 5.6. Definición del bloque IF-THEN (regla base).....	64
Figura. 5.7. Ingreso y escalamiento de valores analógicos.....	65
Figura. 5.8. Escalamiento y salida de valores analógicos.....	65
Figura. 5.9. Habilidad para el <i>trigger</i> del bloque de control difuso.....	65
Figura. 5.10. <i>Timers</i> para generación de pulsos.....	66
Figura. 5.11. Archivo creado en Tag Logging.....	67
Figura. 5.12. Ventana de Propiedades del control de tendencia online de WinC..	68

Figura. 5.13. Simulador de Tags para WinCC.....	69
Figura. 5.14. Simulación del sensor en el Tag Simulation.....	69
Figura. 5.15. Visualización de la simulación en WinCC Runtime.....	70
Figura. 5.16. Plotter de Curva.....	71
Figura. 5.17. Generador de Curva.....	71
Figura. 5.18. Simulación con el Generador de Curva	72
Figura. 5.19. Parámetros de la Curva para la simulación del sensor.....	72
Figura. 5.20. Plotter de Curva con valores simulados.....	73
Figura. 5.21. Actividad de la Regla con valores simulados.....	73
Figura. 5.22. PLC S7-300.....	74
Figura. 5.23. MMC 512KB.....	75
Figura. 5.24. SM334 AI 4/AO 2 8BIT.....	76
Figura. 5.25. Módulo Degem Temperatura.....	76
Figura. 5.26. Sensor IC integrado en el Módulo Degem.....	77
Figura. 5.27. Alimentación del controlador.....	78
Figura. 5.28. Conexiones de la entrada y salida análogas del Módulo SM334 AI 4 / AO 2 8BIT.....	78
Figura. 6.1. HMI de la aplicación.....	79
Figura. 6.2. <i>Trigger</i> apagado.....	80
Figura. 6.3. <i>Trigger</i> encendido.....	80
Figura. 6.4. Ingreso de <i>set point</i>	80
Figura. 6.5. Curvas de tendencia.....	81
Figura. 6.6. Primer caso: temperatura inicial.....	81
Figura. 6.7. Primer caso: ingreso de <i>set point</i>	82
Figura. 6.8. Primer caso: estado estable.....	82
Figura. 6.9. Segundo caso: temperatura inicial.....	83
Figura. 6.10. Segundo caso: ingreso de <i>set point</i>	83
Figura. 6.11. Segundo caso: en proceso de estabilización.....	84
Figura. 6.12. Segundo caso: estado estable.....	84
Figura. 6.13. Tercer caso: temperatura inicial.....	85
Figura. 6.14. Tercer caso: ingreso de <i>set point</i>	85
Figura. 6.15. Tercer caso: estado estable.....	86

ÍNDICE DE HOJAS TÉCNICAS

	Pág.
A2 Hoja técnica de la CPU 315F-2 PN/DP.....	105
A3 Hoja técnica del módulo análogo SM 334; AI 4/AO 2 x 8/8 Bit.....	107

GLOSARIO

ANSI-C

Lenguaje de programación para script en WINCC. Los scripts se emplean para configurar las acciones relacionadas a un objeto (ejecución de funciones).

AUTOMATIZACIÓN BASADA EN COMPONENTES (CBA)

La funcionalidad CBA integrada en la CPU 315F-2 PN/DP permite la modularización tecnológica de máquinas y plantas, lo que aporta una serie de ventajas importantes: se puede estandarizar la planta de mejor forma y se puede reutilizar parte del código o ampliar ésta de forma más sencilla.

BUSINESS INTEGRATION

Describe el camino de integración horizontal y vertical de los procesos en una empresa a través de la repartición de informaciones de diferente contenido entre las soluciones IT empleadas en los diferentes sectores.

HISTORIAN (REGISTRO HISTÓRICO)

Potente base de datos que recoge, archiva los datos de proceso y (a través de una serie de interfaces) los pone a disposición de otras aplicaciones para su procesamiento posterior.

IEC 61131-3

Norma internacional considera un estándar mundial y orientada al futuro en el área de controladores programables.

INFORMATION TECHNOLOGY (IT, TECNOLOGÍA DE LA INFORMACIÓN)

Cualquier sistema, aplicación y estructura relacionada al procesamiento y a

la gestión de informaciones en una empresa.

INTERFACE MULTI PUNTO (MPI)

Protocolo de Siemens, permite la comunicación de los PLCs de esta marca con dispositivos externos como interfaces de operador. Es una versión simple de Profibus.

PROFIBUS

Sistema de bus, potente, abierto y robusto, para la comunicación de proceso y de campo en redes de célula con pocas estaciones y para la comunicación de datos según IEC 61158/EN 50170.

Los dispositivos de automatización tales como PLC, PC, HMI, sensores u actuadores pueden comunicarse a través de este sistema de bus.

PROFIBUS DP

Profibus para un intercambio de datos rápido y cíclico con aparatos de campo.

PROFINET

Estándar Industrial Ethernet abierto y no propietario para la automatización. Con él es posible una comunicación sin discontinuidades desde el nivel de gestión hasta el nivel de campo. Por otro lado cumple las grandes exigencias impuestas por la industria, por ejemplo: sistema de cableado y conexionado apto para ambiente industrial, tiempo real, control de movimiento, ingeniería no propietaria, etc.

PROFINET I/O

Dentro del marco de PROFINET, PROFINET IO es un concepto de comunicación para la realización de aplicaciones descentralizadas y modulares. PROFINET IO se basa en Industrial Ethernet y permite la conexión con la CPU de equipos de E/S y de campo de forma descentralizada.

SILOGISMO

Forma de razonamiento deductivo que consta de dos proposiciones como premisas y otra como conclusión, siendo la última una inferencia necesariamente deductiva de las otras dos.

SISTEMA MONOPUESTO

Los sistemas monopuesto se emplean para aplicaciones pequeñas, pero también pueden manejar y supervisar partes autónomas de la instalación. Un sistema monopuesto trabaja de manera autónoma, es decir que dispone de una comunicación de proceso propia, imágenes propias y archivos propios. A través de una LAN los valores también pueden ser transmitidos a un concentrador de datos (WinCC como servidor OPC).

CAPÍTULO 1

INTRODUCCIÓN

1.1 ANTECEDENTES

Desde siempre la lógica binaria ha sido utilizada en procesos electrónicos de datos, resultando poco adecuada para describir las diversas situaciones de la vida real debido a que las principales variables utilizadas en los procesos industriales son medidas en valores analógicos.

De allí surgió la necesidad de denotar términos que permitan expresar valores intermedios, llegando a intentar aplicar una forma más humana de pensar en la programación de computadoras. La lógica difusa surge como respuesta a este problema, fue iniciada en 1965 por Lotfi A. Zadeh, en 1973 Mamdani usó la lógica difusa para el control de procesos, en 1988 los japoneses le dieron un uso práctico, y solo desde 1991 la lógica difusa ha sido usada en Estados Unidos y Europa, convirtiéndose actualmente en un método muy utilizado para el control de subsistemas y procesos industriales, así también para la electrónica de entretenimiento y hogar, sistemas de diagnóstico y otros sistemas expertos.

La Escuela Politécnica del Ejército recientemente ha adquirido el software FuzzyControl++ y equipos marca Siemens que permiten el desarrollo de control difuso, poniendo así a disposición de sus alumnos una de las más novedosas técnicas de control.

1.2 JUSTIFICACIÓN E IMPORTANCIA

Debido al rápido crecimiento de esta tecnología y a la incesante aparición de aplicaciones, sin mencionar que los artículos electrónicos sin lógica difusa están pasando a ser obsoletos, es imprescindible la inversión de tiempo, dinero, y esfuerzo en la investigación y desarrollo de la lógica difusa en nuestro país.

El análisis y uso de herramientas como el software FuzzyControl++ de Siemens pretende ser una fuente de apoyo para el manejo de estos nuevos recursos, exponiendo los conocimientos necesarios sobre lógica difusa, utilización de comandos para programación, e implementación de un sistema prototipo de control en el que se utiliza esta técnica.

El proyecto procura mostrar de un modo práctico los conocimientos que el estudiante debe poseer para la aplicación de este software, evitando así la pérdida de tiempo en consultas vagas y dando respuesta a los problemas más comunes que pueden suceder durante la programación.

Además será un pilar para el desarrollo de esta tecnología en nuestro país, y un aporte para que la Escuela Politécnica del Ejército en cumplimiento con su misión entregue al país profesionales acorde a las tendencias mundiales.

1.3 ALCANCE DEL PROYECTO

Lo que se busca con este proyecto es documentar de una manera muy clara y específica todo aquello que permita al usuario realizar un control basado en lógica difusa mediante la utilización del software FuzzyControl++. Para esto se detallará los fundamentos básicos que permitan comprender la lógica difusa, los requerimientos del sistema para la instalación del software, guía para crear un sistema difuso, comunicación entre FuzzyControl++ y la familia SIMATIC-S7 y SIMATIC- WinCC, y la estructura de operación de FuzzyControl++.

Además se implementará una aplicación de control *fuzzy* utilizando los recursos existentes en el laboratorio de Robótica y PLCs de la ESPE (PLCs Siemens, Software FuzzyControl++, planta de proceso).

Se diseñarán pantallas HMIs usando WinCC y se programará la lógica del PLC en Step7 Professional.

1.4 OBJETIVOS

1.4.1 General

Diseñar e implementar un sistema de entrenamiento en control difuso utilizando ControlFuzzy++.

1.4.2 Específicos

- Recopilar y analizar los fundamentos necesarios sobre control difuso.
- Sistematizar e integrar en un documento el manejo del software FuzzyControl++ y de sus instrumentos auxiliares: WinCC y Step7, así como del hardware S7-300 y sus módulos de entrada y salida.
- Programar una aplicación de control difuso utilizando FuzzyControl++.
- Simular una aplicación de control difuso en el software respectivo.
- Implementar una aplicación de control difuso sobre una planta de procesos.
- Documentar adecuadamente el proyecto.

1.5 DESCRIPCIÓN GENERAL DEL PROYECTO

Este proyecto contiene conceptos fundamentales sobre lógica difusa y datos esenciales que permiten la aplicación de la misma.

Da a conocer las partes más importantes del hardware que componen el S7 300, como son: CPU 315F-2 PN/DP, módulos de entrada/salida, y la tarjeta de memoria Micro.

Proporciona una vista generalizada del software utilizado: Step7 Professional, WinCC y FuzzyControl++. Enfoca problemas existentes en el software de FuzzyControl++ y brinda una solución eficaz.

Para completar este trabajo se incluye una aplicación enfocada al control de temperatura mediante control difuso, se la pone a prueba y se analiza los resultados obtenidos.

Finalmente se cita todo aquello que se pudo concluir con el proyecto, dando recomendaciones para un mejor desarrollo.

CAPÍTULO 2

LÓGICA DIFUSA

2.1 HISTORIA DE LA LÓGICA DIFUSA

Si bien los conjuntos borrosos (difusos o *fuzzy*) hicieron su aparición en 1965 de la mano de Lotfi A. Zadeh con el fin de representar información y datos imprecisos cabe destacar que la lógica difusa surgió años atrás.

Las Leyes del Pensamiento de Aristóteles permitieron el desarrollo de la lógica y de las matemáticas. La Ley del Tercero Excluido establece que cualquier proposición solo puede ser Verdadera o Falsa y que ningún otro valor de verdad intermedio está permitido. Parménides (300 A.C.) al proponer la primera versión de esta ley encontró serias e inmediatas objeciones. Heráclito propuso cosas que podían ser simultáneamente ciertas y falsas. Pero sería Platón quien colocara la “primera piedra” de la Lógica Difusa señalando que “hay una tercera región entre lo verdadero y lo falso donde los opuestos se presentan juntos”¹.

En el siglo XVIII, en filósofo inglés David Hume habló de la lógica del sentido común (razonamiento basado en la experiencia que la gente comúnmente adquiere de sus vivencias por el mundo)².

Entre 1917 y 1920 J. Lukasiewicz presenta la primera formulación sistemática de una alternativa a la lógica bivaluada de Aristóteles. Esta lógica

¹ FSI Conjuntos Difusos y Lógica Difusa, www.sci2s.ugr.es/docencia/doctoSCTID/FSIConjuntos%20difusos-Introduccion.pdf, página 2, 2008-06-02.

² López, Bruno, Lógica Difusa, <http://www.mitecnologico.com/Main/EIProcesoDeRazonamientoSegunLaLogica>, 2005-08-26, 2008-06-02.

trivaluada describe al término “posible” como un tercer valor de verdad, asignando un valor numérico entre 0 y 1, y proponiendo sus respectivas matemáticas. Lukasiewicz planteó una notación completa y un sistema axiomático a partir de los cuales esperaba derivar “matemática moderna”³. Más tarde exploró lógicas polivaloradas determinando que no existía impedimento para una lógica infinito valorada, lo cual formalizó hacia 1930.

El soviético Lofti Zadeh, actualmente catedrático e investigador en la Universidad de California – Berkeley, hizo la presentación por primera vez (1965) de un trabajo sobre Conjuntos Borrosos (Fuzzy Sets):

"No tiene sentido buscar la solución a un problema no perfectamente definido, por medio de un planteamiento matemático muy exacto, cuando es el ser humano el primero que razona empleando inexactitud"⁴.

“Zadeh introduce una lógica infinito valorada caracterizando el concepto de Conjunto Difuso y por extensión la Lógica Difusa”⁵, considera un número infinito de grados de valor entre cierto y falso. La Lógica Difusa define grados de relación y grados de verdad de algo, algunas cosas pueden ser parcialmente verdaderas y parcialmente falsas simultáneamente. Tras introducir las operaciones básicas sobre los conjuntos difusos, Zadeh convierte repentinamente al álgebra de Boole en un caso particular de la suya propia la que es más general y adaptable a los casos reales.

A inicios de los setenta, la lógica difusa se planteaba muy fuerte. Zadeh formula las bases del razonamiento aproximado a partir de 1971, posteriormente introduce los modificadores lingüísticos, las variables lingüísticas para la representación de la información difusa, obteniendo la regla composicional de

³ FSI Conjuntos Difusos y Lógica Difusa, www.sci2s.ugr.es/docencia/doctoSCTID/FSIConjuntos%20difusos-Introduccion.pdf, página 4, 2008-06-02.

⁴ Universidad Nacional de Cuyo Facultad de Ingeniería, Instrumentación y control automático, <http://fing.uncu.edu.ar/catedras/archivos/control/controldifuso.pdf>, página 2, 2008-06-11.

⁵ FSI Conjuntos Difusos y Lógica Difusa, www.sci2s.ugr.es/docencia/doctoSCTID/FSIConjuntos%20difusos-Introduccion.pdf, página 6, 2008-06-02.

inferencia que insta los criterios para realizar inferencia en entornos de vaguedad e incertidumbre⁶.

En 1974 Ebrahim Mamdani junto con S. Asilian demuestran la aplicabilidad de la lógica *fuzzy* al control de procesos, abriendo un área de aplicaciones de tremendas proporciones⁷.

Zadeh pronto pasó de la inferencia difusa a los sistemas expertos y de ello, al razonamiento aproximado. En los ochenta ya existían grandes escuelas de investigadores y tecnólogos, así como innumerables grupos de investigación universitaria, especialmente en Japón, Estados Unidos, Alemania, España y Francia, Lotfi Zadeh a mediados de esta década establece las reglas del razonamiento silogístico borroso y el razonamiento que él mismo denominó del “sentido común”⁸.

Velozmente la lógica difusa empezó a ser usada en diferentes aplicaciones prácticas, alcanzando su máximo apogeo a principios de los años 90, y extendiéndose hasta la época actual.

2.2 FUNDAMENTOS

2.2.1 Conjuntos no difusos y difusos

La teoría de conjuntos clásica establece que todos los elementos son miembros o no de un conjunto, es decir que solo puede tomar un valor lógico verdadero o falso, alto o bajo, prendido o apagado, cerrado o abierto, en definitiva 1 ó 0, a estos conjuntos se los conoce como conjuntos no difusos. Por ejemplo con referencia al tipo de conjuntos no difusos una puerta sólo tiene dos posibilidades, está abierta o cerrada.

⁶ Gutiérrez, Julio, Laudatio, http://www.upm.es/canalUPM/notasprensa/Discurso_Julio_Gutierrez_E.pdf, página 2, 2007-01-29, 2008-06-02.

⁷ Gutiérrez, Julio, Laudatio, http://www.upm.es/canalUPM/notasprensa/Discurso_Julio_Gutierrez_E.pdf, página 2, 2007-01-29, 2008-06-02.

⁸ Gutiérrez, Julio, Laudatio, http://www.upm.es/canalUPM/notasprensa/Discurso_Julio_Gutierrez_E.pdf, página 2, 2007-01-29, 2008-06-02.

Por otro lado la lógica difusa permite valores intermedios que simulan de mejor manera el pensamiento humano. Los seres humanos emplean conceptos que tienen una transición gradual a su significado opuesto, por lo cual en el anterior ejemplo dentro de un conjunto difuso dicha puerta no tiene porqué estar necesariamente abierta (valor 1) ni cerrada (valor 0), sino que además puede estar abierta a medias (valor 0.5), bastante abierta (valor 0.8), casi cerrada (0.1), etc.

2.2.2 Función de Pertenencia

En contraste con la lógica tradicional que utiliza conceptos absolutos para referirse a la realidad, la lógica difusa define en grados variables de pertenencia a los mismos, siguiendo patrones de razonamiento similares a los del pensamiento humano.

2.2.3 Tipos de Funciones de Pertenencia

Existen diferentes tipos de funciones de pertenencia (función gamma, función sigma, función trapezoidal, función gaussiana), para propósitos prácticos, funciones triangular y trapezoidal son las estándar. Ellas pueden ser descritas con 3 o 4 puntos y por lo tanto no causan mucho gasto computacional⁹.

Una función de pertenencia muy destacada es la unitaria o *singleton*, en la que solo se toma en cuenta la altura de la cima. En otras palabras solo las líneas verticales de las funciones de pertenencia con altura 1, es decir en la posición al 100% sobre el eje de la señal de salida donde se localiza la cima. En la figura. 2.1. se puede ver la función de pertenencia *singleton*, la cual es igual a cero para todos los casos excepto cuando $x = \mu_i$.

⁹ Siemens, FuzzyControl++ User's Manual, http://www.industry.siemens.de/industrial-services/EN/SOLUTION_SERVICES/SIMATIC_ADDONS/fuzzy_download.htm, página 5, 2003, 2008-06-11.

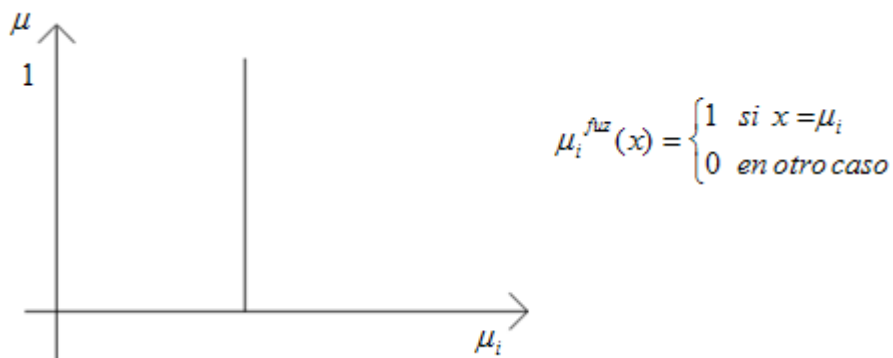


Figura. 2.1. Función de Pertenencia Unitaria o *Singleton*

Ejemplo: Monitorización de presión en una planta de proceso¹⁰.

El operador de una planta de procesamiento determina una presión de 100 bares como óptima para el proceso de producción. Sin embargo, el permitió un rango de operación de 80 a 120 bares como aceptable (“Buena Presión”). La figura. 2.2. muestra la diferencia entre el conjunto de valores de presión “Buena Presión” definiendo tanto un conjunto no difuso y un conjunto difuso. La representación de los valores de pertenencia (grados de pertenencia) mediante los elementos del conjunto (aquí valores de presión) es la función de pertenencia.

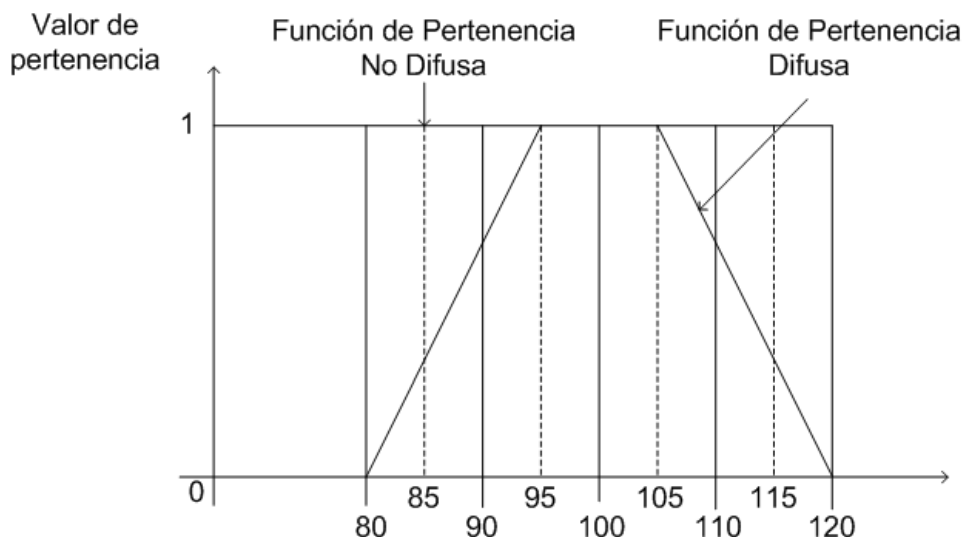


Figura. 2.2. Función de Pertenencia

¹⁰ Siemens, FuzzyControl++ User's Manual, http://www.industry.siemens.de/industrial-services/EN/SOLUTION_SERVICES/SIMATIC_ADDONS/fuzzy_download.htm, página 5, 2003, 2008-06-11.

El operador se prepara para tomar acción cuando la presión mostrada alcanza 115 bares ya que la presión puede solo ser considerada 30% buena. Un control binario con un valor umbral contactado a 120 bares no podría realizar tal operación calificada e interpretaría la situación de repente cuando el valor de presión se vaya al conjunto no difuso. El operador, por otro lado, puede reconocer valores críticos y tendencias con anticipación y puede por tanto reaccionar a tiempo tomando algunas medidas necesarias para contrarrestar. Para imitar este comportamiento es necesario usar un sistema difuso en lugar de un control binario.

2.2.4 Operaciones entre Conjuntos Difusos

Los Conjuntos difusos operan entre sí del mismo modo que los conjuntos clásicos. Ya que los primeros son una generalización de los segundos, es dable definir las operaciones de unión, intersección y complemento haciendo uso de las mismas funciones de pertenencia¹¹: sean $\mu_A(x)$ y $\mu_B(x)$ las funciones de pertenencia correspondientes a los conjuntos difusos A y B, como se indica en las figuras. 2.3. y 2.4. respectivamente, Zadeh propone:

- Unión u OR: $\mu_{A \cup B}(x) = \max(\mu_A(x), \mu_B(x))$. Ver figura. 2.5.
- Intersección o AND: $\mu_{A \cap B}(x) = \min(\mu_A(x), \mu_B(x))$. Ver figura. 2.6.
- Complemento o NOT: $\mu_{\neg A}(x) = 1 - \mu_A(x)$. Ver figura. 2.7.

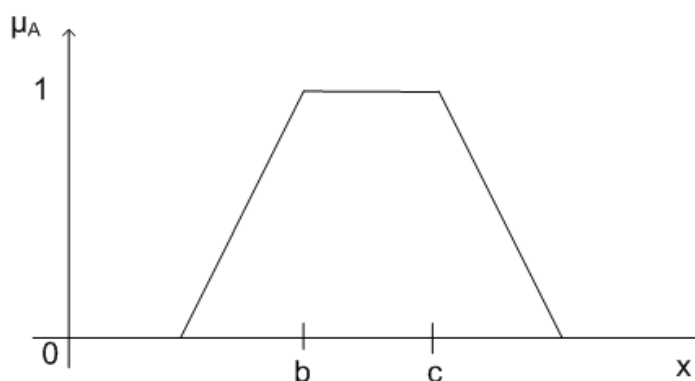


Figura. 2.3. Conjunto Difuso A

¹¹ Corzo, Yuliana, La Lógica, <http://www.monografias.com/trabajos6/lalo/lalo.shtml>, 2008-06-11.

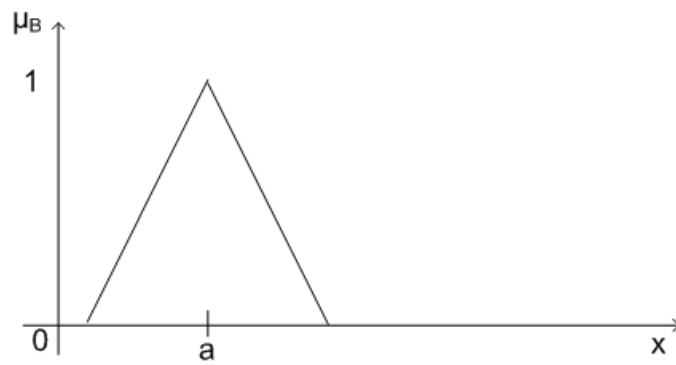


Figura. 2.4. Conjunto Difuso B

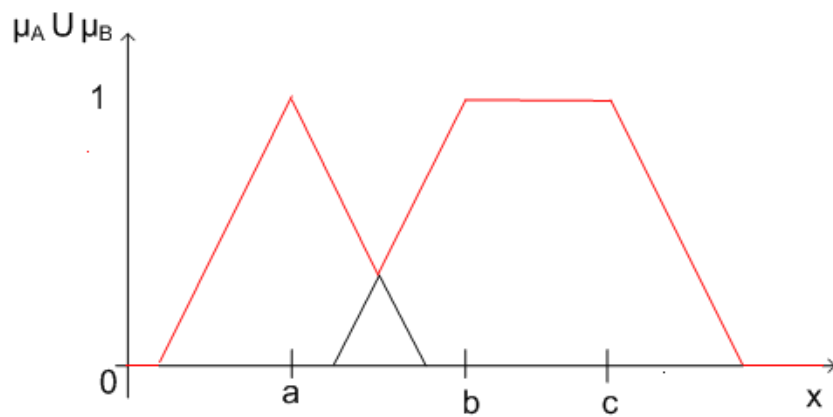


Figura. 2.5. Unión entre A y B

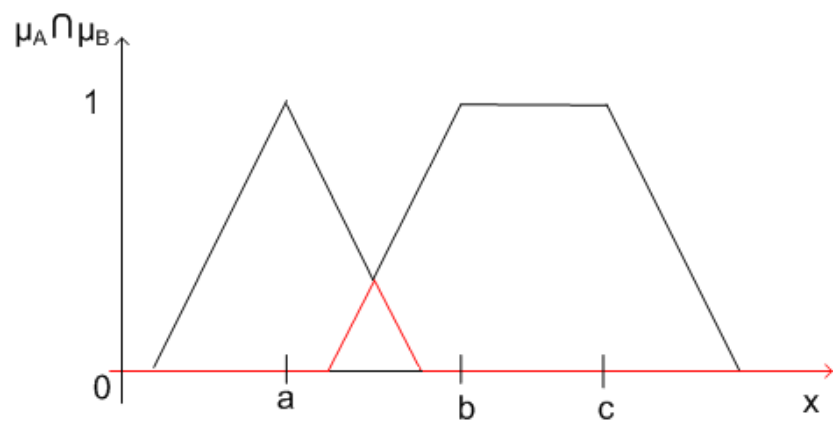


Figura. 2.6. Intersección entre A y B

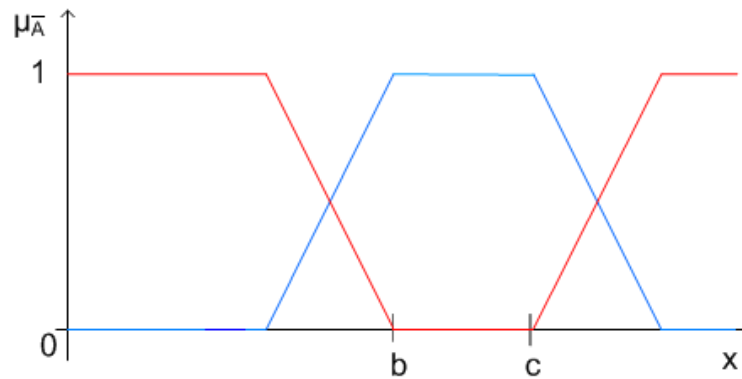


Figura. 2.7. Complemento de A

2.3 CONTROL DIFUSO

2.3.1 Características del Control Difuso¹²

Entre sus principales características están:

1. El control difuso es *lógico*. El denominador de lógico significa que el algoritmo de control utiliza expresiones *IF-THEN*, con las que se pueden mencionar muchas condiciones, combinando expresiones lógicas con *IF* y *AND*.
2. El control difuso es *disperso*. Esta característica, que diferencia básicamente los sistemas de control difuso de los sistemas de control basados en una sola ecuación, admite la coexistencia de controladores con lógicas distintas y su actuación en paralelo.
3. El control difuso es *lingüístico*. Permite el uso de variables lingüísticas imprecisas, en particular en los antecedentes de las reglas. Dicho lenguaje cualitativo es fácil de entender, permite efectuar el control por medio de un diálogo con los operadores, usando sus ojos experimentados en la observación del proceso como entradas externas, e introducir variables como las condiciones del proceso a modo de información útil para el control.

Asimismo el algoritmo de control puede contener los procedimientos inusuales que hay siempre en las operaciones de un proceso real.

¹² Control Difuso, <http://prof.usb.ve/montbrun/ps2320/fuzzy/fuzzy.html>, 2008-06-10.

2.3.2 Función de un Sistema de Control Difuso

Esencialmente un sistema de control difuso, es una estructura basada en conocimiento definida a través de un conjunto de reglas difusas del tipo si-entonces, las cuales, contienen una cuantificación lógica difusa de la descripción lingüística del experto de como realizar un control adecuado. La información se procesa difusamente en la forma de reglas, tal como:

Si valor de entrada “buena presión”, ENTONCES valor de salida “hacer nada”, o

Si valor de entrada “medio”, ENTONCES hacer valor de salida “pequeño”.

“Los valores de entrada y salida son variables lingüísticas y los términos “medio” y “pequeño” no son valores numéricos pero si valores lingüísticos. Ellos son también llamados conjuntos difusos. Una variable lingüística usualmente posee 3, 5 o 7 valores lingüísticos. La colección completa de las reglas forma la regla base”¹³.

No obstante, el sistema difuso es, forzosamente, parte de un sistema técnico que trabaja con valores numéricos (señales), con valores no difusos, en sus entradas y salidas, por lo cual existen dos tipos de sistemas difusos:

- Sistema de control difuso de *Takagi-Sugeno*: Sus entradas y salidas son variables con valores reales.
- Sistema de control difuso de *Mamdani*: Interpone un fusificador en la entrada y un defusificador en la salida. En la figura. 2.8. se ilustra el principio.

¹³ Siemens, FuzzyControl++ User's Manual, http://www.industry.siemens.de/industrial-services/EN/SOLUTION_SERVICES/SIMATIC_ADDONS/fuzzy_download.htm, página 5, 2003, 2008-06-11.

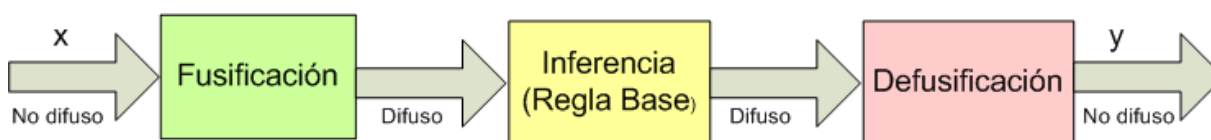


Figura. 2.8. Configuración básica de un Sistema de Control Difuso Mamdani

FuzzyControl++ utiliza el sistema difuso *Mamdani* razón por la cual el proyecto solo hace referencia a este modelo.

2.3.3 Sistema de control difuso de Mamdani

En sistema de control difuso de Mamdani está compuesto por:

- **Borrosificación (Fusificación)**¹⁴

Los grados de satisfacción para los valores lingüísticos (grados de pertenencia del conjunto difuso) de las variables lingüísticas son fijados a los valores de entrada no difusos.

Tal fusificación determina, por ejemplo, a que grado la “presión es buena” si esta es, digamos, 115 bares. Esto se hace usando la función de pertenencia. Este grado es llamado “grado de satisfacción o cumplimiento de la parte SI”.

- **Inferencia**¹⁵

Para cada regla de la regla base, el grado de satisfacción de la parte ENTONCES es formado desde el grado de satisfacción de la parte SI por un cierto método. Este proceso es llamado también Implicación. El grado de satisfacción de la parte entonces es equivalente al grado de satisfacción de la regla, la cual es llamada también intensidad de la regla. Todas estas evaluaciones de reglas individuales puestas juntas resultan en una función de pertenencia para la señal de salida, la cual es también llamada composición.

¹⁴ Siemens, FuzzyControl++ User's Manual, http://www.industry.siemens.de/industrial-services/EN/SOLUTION_SERVICES/SIMATIC_ADDONS/fuzzy_download.htm, página 6, 2003, 2008-06-11.

¹⁵ Siemens, FuzzyControl++ User's Manual, http://www.industry.siemens.de/industrial-services/EN/SOLUTION_SERVICES/SIMATIC_ADDONS/fuzzy_download.htm, página 6, 2003, 2008-06-11.

El resultado de la función de pertenencia describe un “comando de control difuso” que podría, por ejemplo, ser algo como “válvula de posición solo por debajo del centro”.

Cuando la parte SI contiene una declaración combinada “SI....Y...”--- (ENTONCES....), la operación Y de la lógica difusa es ejecutada primero y el grado de satisfacción es usado en la evaluación de la regla total. Todas estas declaraciones juntas son a menudo llamadas una agregación. La elección del operador de implicación condiciona la elección del operador de agregación¹⁶, la tabla. 2.1. indica el tipo de implicación con su respectiva agregación:

Tabla. 2.1. Interpretación de la agregación bajo varias implicaciones

IMPLICACIÓN	AGREGACIÓN
Zadeh Máx-Mín	Y (\wedge)
Mamdani Mín	O (\vee)
Larsen Producto	O (\vee)
Aritmética	Y (\wedge)
Booleana	Y (\wedge)
Producto Obligado	O (\vee)
Producto Drástico	O (\vee)
Secuencia Estándar	Y (\wedge)
Gougen	Y (\wedge)
Gödelian	Y (\wedge)

FUZZYCONTROL++ utiliza como operador de implicación el Mínimo y como operador de agregación el Máximo. Estos métodos los adoptaron por ser los más utilizados y más eficientes desde el punto de vista computacional.

- **Desborrosificación (Defusificación)**

El valor de salida numérico (no difusa) más representativo es calculado para la variable de control desde el comando de control difuso (en la forma de la

¹⁶ Marín Roque, Tareas Inteligentes Control Fuzzy, <http://perseo.dif.um.es/~roque/ti/Material/Tema3.ppt>, diapositiva 23, 2006-06-15.

función de pertenencia resultante)¹⁷.

Las técnicas de defusificación más usadas son¹⁸:

- Centroide o Centro de las Areas (COA o Center of Areas)
- Centro de las Sumas (COS o Center of Sums)
- Media de los Máximos (MOM o Mean of Maxima)
- Promedio de los Centros (CA o Center Average)

La herramienta de programación en estudio utiliza el método del centro de gravedad o centroide¹⁹, el cual es simple cuando los conjuntos de salida son funciones de pertenencia tipo “*singletons*”. De acuerdo este método el valor de salida del sistema difuso se calcula con la ecuación (2.1)²⁰:

$$Salida = \frac{\mu_1 * Sg_1 + \mu_2 * Sg_2 + \dots + \mu_n * Sg_n}{\mu_1 + \mu_2 + \dots + \mu_n} \quad (2.1)$$

Donde μ_n es el grado de activación de las reglas y Sg_n es el valor del “*singleton*” de salida.

2.3.4 Aplicaciones del Control Difuso

El control difuso es aplicado en procesos complejos, en casos en los que no hay un modelo matemático simple. En procesos no lineales, o cuando el ajuste de una variable puede producir el desajuste de otras. Es aplicable si se puede expresar lingüísticamente el conocimiento experto. Permite el procesamiento de información a través de algoritmos expresados verbalmente en forma de la regla SI-ENTONCES. Por ejemplo en procesos de manufactura complicados los operadores expertos toman acciones de control usando el criterio de si-entonces,

¹⁷ Siemens, FuzzyControl++ User's Manual, http://www.industry.siemens.de/industrial-services/EN/SOLUTION_SERVICES/SIMATIC_ADDONS/fuzzy_download.htm, página 6, 2003, 2008-06-11.

¹⁸ Marín Roque, Tareas Inteligentes Control Fuzzy, <http://perseo.dif.um.es/~roque/ti/Material/Tema3.ppt>, diapositiva 31, 2006-06-15.

¹⁹ Debido a que el centroide coincide con el centro de masa, pues el objeto en estudio posee una densidad uniforme y está bajo la influencia de un campo gravitatorio uniforme, el centroide es igual al centro de gravedad.

²⁰ FuzzyComp, <http://www2.unalmed.edu.co/dyna2005/135/fuzzycomp.pdf>, página3, 2002, 2008-06-15.

obteniendo una buena operación con resultados de calidad.

No solo controla procesos en lazo cerrado y en lazo abierto, el control difuso es usado también para el procesamiento de información en la adquisición de valores (señales) y en el manejo y planeamiento (programación y predicción) de operaciones. Hace posible transmitir la experiencia de un experto a una computadora, logrando una solución inicial y automatizada rápidamente²¹. En general, cuando se quiera representar y operar conceptos que tengan imprecisión o incertidumbre²².

En la actualidad se puede encontrar todo tipo de instrumentos, máquinas y procedimientos controlados difusamente, adecuándose "inteligentemente" a cada situación particular: acondicionadores de aire, frigoríficos, lavadoras/secadoras, aspiradoras, hornos microondas, mantas eléctricas, ventiladores, autoenfoques fotográficos, estabilizadores de imágenes en grabadoras de vídeo, transmisiones de automóviles, suspensiones activas, controles de ascensores, dispensadores de anticongelantes para los aviones en los aeropuertos, sistemas de toma de decisiones industriales o económicas, entre muchas otras aplicaciones.

2.2.5 Ventajas y desventajas del Control Difuso

La ventaja primordial es la falta de expresiones matemáticas, tales como ecuaciones algebraicas o ecuaciones diferenciales etc., que describan el proceso a ser automatizado. El manejo de términos lingüísticos como a medias, bastante, casi poco, mucho, algo, etc. permite plantear el problema en las mismas expresiones en las que lo haría un experto humano, facilitando así el diseño y uso de sistemas difusos.

Pero no usar un modelo matemático es también una fuente de incertidumbre, pues los muchos grados de libertad en el diseño de sistemas de

²¹ Siemens, FuzzyControl++ User's Manual, http://www.industry.siemens.de/industrial-services/EN/SOLUTION_SERVICES/SIMATIC_ADDONS/fuzzy_download.htm, página 3, 2003, 2008-06-11.

²² Siemens, FuzzyControl++ User's Manual, http://www.industry.siemens.de/industrial-services/EN/SOLUTION_SERVICES/SIMATIC_ADDONS/fuzzy_download.htm, página 3, 2003, 2008-06-11.

control difusos puede ser una desventaja especial.

Entre sus ventajas se menciona:

- Fácil ejecución de reglas expresadas verbalmente (si...,entonces...) sobre un computador para resolver un problema.
- El comportamiento de un sistema difuso es entendible para los seres humanos.
- En comparación con otros métodos, evita el costoso desarrollo de una descripción matemática.
- El control difuso cubre un amplio rango de parámetros del sistema y puede afrontar la mayoría de las perturbaciones.

Algunas de sus desventajas son:

- La definición de tareas sin suficiente conocimiento del sistema y poco o muy impreciso conocimiento del comportamiento del sistema resultan en malas, posiblemente inutilizables, soluciones difusas.
- Usualmente no posee capacidad de adaptabilidad y aprendizaje si el comportamiento del sistema cambia.
- El diseño de un sistema de control difuso requiere experiencia debido a los muchos grados de libertad.
- Aún no están bien formalizadas las características necesarias para que el control difuso sea robusto y estable.

2.2.6 Ejemplo de un Control Difuso²³

El siguiente ejemplo ilustra los tres típicos bloques de funciones de un controlador difuso de Mamdani. Este enfatiza en la claridad que en la creación realista de valores.

Ejemplo: Control de la aceleración de un vehículo:

²³ Siemens, FuzzyControl++ User's Manual, http://www.industry.siemens.de/industrial-services/EN/SOLUTION_SERVICES/SIMATIC_ADDONS/fuzzy_download.htm, página 7, 2003, 2008-06-11.

- Objetivo: Aceleración automática y frenado adaptados a la situación del vehículo. Las siguientes definiciones son requeridas antes del diseño de la regla base difusa:
 - Señales de entrada de la regla base:
 1. Distancia desde el vehículo en frente (variable lingüística “Distancia”) valores lingüísticos: “pequeña”, “media”, “larga”.
 2. Propia velocidad actual (variable lingüística “Velocidad”) valores lingüísticos: “lenta”, “media”, “rápida”.
 - Variables de control: Aceleración (positiva - aceleración o negativa - frenado) del vehículo (variable lingüística “Aceleración”) valores lingüísticos: “neg_larga”, “neg_peque”, “cero”, “pos_peque”, “ pos_larga”
 - La regla base:
 - R1: SI Distancia = pequeña Y Velocidad = rápida, ENTONCES Aceleración = neg_larga;
 - R2: SI Distancia = pequeña Y Velocidad = media, ENTONCES Aceleración = neg_peque;
 - R3: SI Distancia = mediana Y Velocidad = rápida, ENTONCES Aceleración = neg_peque;
 - R4: SI Distancia = mediana Y Velocidad = media, ENTONCES Aceleración = cero;
 - R5: SI Distancia = mediana Y Velocidad = lenta, ENTONCES Aceleración = pos_peque;
 - R6: SI Distancia = larga Y Velocidad = media, ENTONCES Aceleración = pos_peque;
 - R7: SI Distancia = larga Y Velocidad = lenta, ENTONCES Aceleración = pos_larga;

Las reglas también son representadas en una matriz.

- Definición de las funciones de pertenencia (conjuntos difusos) para los valores lingüísticos:

En este paso de diseño, la forma de los conjuntos difusos se selecciona junto con sus rangos de influencia, el rango de los valores no difusos de la señal de entrada y salida se determinan para que el conjunto difuso tenga un grado de pertenencia más grande que cero.

- Definición del procedimiento de inferencia y defusificación:

Existe un número de métodos, los comunes incluyen inferencia MAX-MIN e inferencia MAX-PROD y defusificación de acuerdo al método centroide. En la práctica, la elección depende de las herramientas de diseño que el programa de computación proporciona. FUZZYCONTROL++ usa un método especial que solo

demanda un bajo esfuerzo computacional en la implementación práctica de un sistema difuso, el principio se describe a continuación.

Para explicar como trabaja la regla base difusa, se plantea un “caso de funcionamiento” en el cual la velocidad es 65 mph y la distancia es 51m. Las funciones de pertenencia se muestran en los siguientes dos diagramas. En la elección de las curvas de las funciones de pertenencia, los grados de libertad disponibles deben ser definidos en una manera significativa (en la forma de “puntos intermedios” de los triángulos o trapecios). Por ejemplo, el conjunto difuso “medio” de la señal de entrada Distancia se refiere al rango de 30 a 120 m.

Fusificación:

Lo siguiente puede ser visto desde las curvas de las funciones de pertenencia elegidas para las dos señales de entrada:

Las declaraciones difusas sobre la distancia, la cual es 51 m, son 30% verdad para “pequeño” y 70% verdad para “media”, como lo grafica la figura. 2.9. (La declaración “larga” es 0% verdad. Tales casos son usualmente no ejercidos.) Similarmente, los grados de pertenencia para “media” (0.8) y para “rápida” (0.2) resultan para los conjuntos difusos de la variable “Velocidad”, como lo indica la figura. 2.10.

Inferencia:

Se considera la regla 1 como un ejemplo:

En la parte SI de la regla, las declaraciones difusas sobre las variables Distancia y Velocidad realizan la operación Y. Distancia es “30% pequeña” y velocidad es “20% rápida”. La parte SI entera de esta regla es por tanto considerada para ser 20% cumplida, después de la aplicación de la operación MÍNIMA para la combinación Y de la lógica difusa (implicación).

La parte ENTONCES de la regla 1 se refiere al conjunto difuso “neg_larga”

de la señal de salida. El más simple método de consecuencia hace la suposición (apropiada) que una consecuencia puede ser cumplida a un grado no más grande que el antecedente. Porque si la parte SI es 20% cumplida, esto puede deducir una aceleración requerida del 20% “neg_larga” (consecuencia).

Este grado de cumplimiento se usa para modificar las funciones de pertenencia de la señal de salida en la parte ENTONCES de la regla 1.

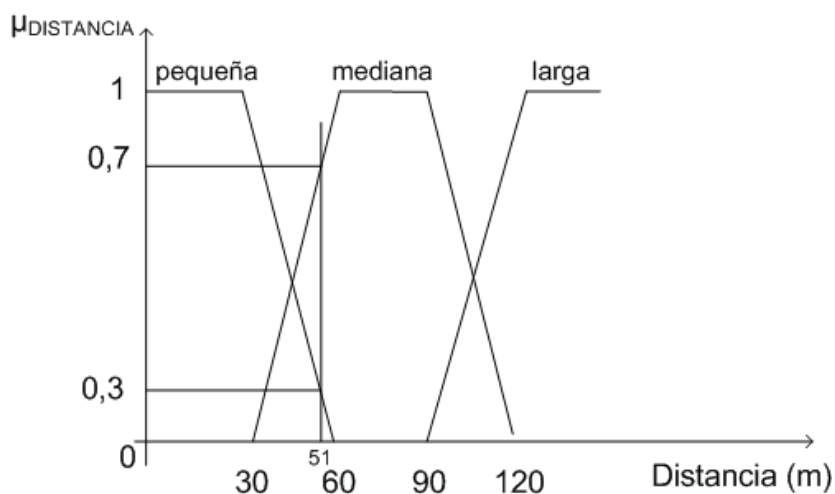


Figura. 2.9. Función de Pertenencia de la variable Distancia

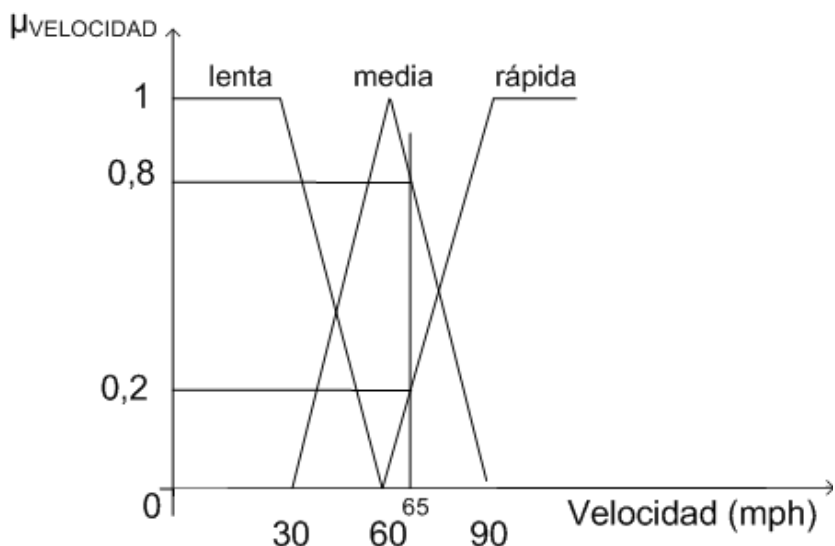


Figura. 2.10. Función de Pertenencia de la variable Velocidad

La figura. 2.11. muestra la señal de salida “Aceleración” con sus 5 conjuntos

difusos. El conjunto difuso “neg_larga” es modificado por el 20% del grado de satisfacción de la regla 1. Un método simple es considerarlos como *singletons*.

Toda esa regla de evaluación acorta la línea, en este ejemplo, al 20%, como se muestra en el diagrama.

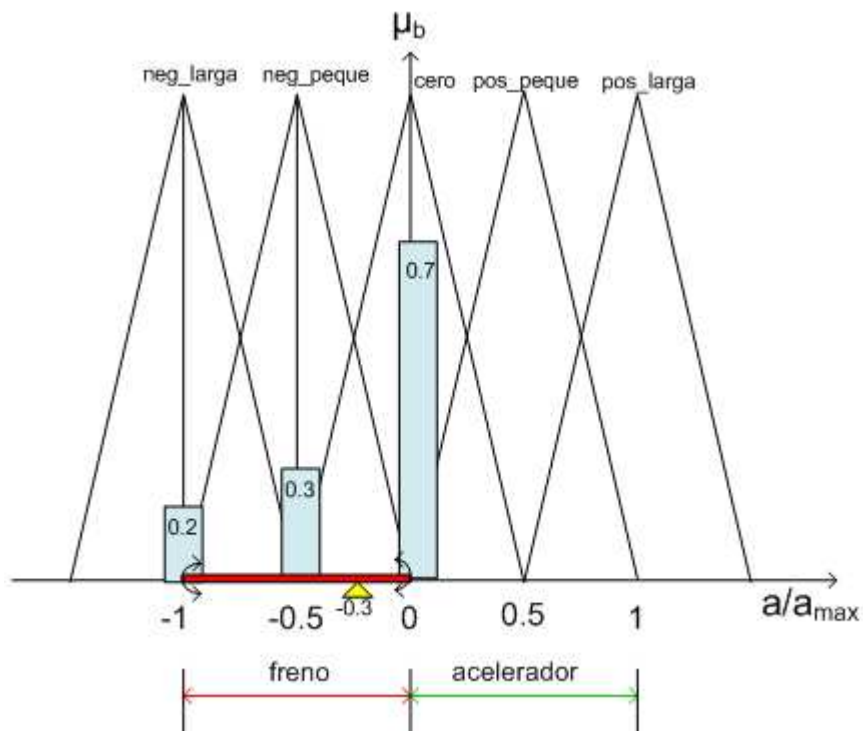


Figura. 2.11. Señal de salida Aceleración

La suma de las reglas restantes a la evaluación para el mismo par de valores de entrada no difusa (51m y 65mph), dan los siguientes grados de satisfacción:

- Regla1: El comando: Hacer aceleración “neg_larga” es 0.2 cumplida
- Regla2: El comando: Hacer aceleración “neg_peque” es 0.3 cumplida
- Regla3: El comando: Hacer aceleración “neg_peque” es 0.2 cumplida
- Regla4: El comando: Hacer aceleración “cero” es 0.7 cumplida

En este caso, las reglas de 5 a 7 no son cumplidas, porque al menos una condición SI no es cumplida, el grado de pertenencia es cero para el valor de entrada no difuso actual (regla 5 con velocidad lenta).

La recomendación de control de la base de la regla difusa consiste de las cuatro declaraciones parciales de arriba en este caso operando. (Se nota que dos diferentes recomendaciones para el mismo conjunto difuso “neg_peque” resultan de las reglas 2 y 3). La redacción en una función de pertenencia resultante se ejecuta mejor por superposición de todas estas “funciones de pertenencia parcial” (composición). Una posible operación (agregación) es la operación MAXIMA. Así, el resultado de la regla 2 cubre el resultado de la regla 3 y la recomendación de control resultante consiste de las tres columnas mostradas en el diagrama. Con FUZZYCONTROL++, se utiliza la operación SUMA para que una intensidad de regla de 0.3 resulte desde las reglas 2 y 3 en la posición $\frac{a}{a_{\max}} = -0.5$.

Sin embargo, la resultante recomendación de control de toda la actividad de la regla es aún difusa y se podría expresar como “de ninguna manera frenar completamente pero no está lista la mitad de la potencia de frenado”

Defusificación:

De esta recomendación de control, la cual es aún difusa, se calcula un valor numérico no difuso para la variable de control, como un valor numérico (a partir del rango manipulado). Este es representativo de la recomendación de control difusa, es decir de la función de pertenencia resultante. Por ejemplo, si la función de pertenencia resultante consiste de la superposición de áreas parciales de triángulos y trapecoides, la posición del centro de gravedad del área total sobre el eje de la señal de salida es el valor no difuso buscado.

En este caso (ver diagrama anterior), se imagina un sistema mecánico en el cual las tres columnas para valores de aceleración, cuyo peso es una función de su altura, son balanceadas sobre un pivote a lo largo del eje (“equilibrio”). Este es el principio en el que se basa el método del centroide.

Para el ejemplo, el siguiente cálculo simple mediante el uso de la formula

(2.1) devuelve un valor no difuso para la variable de control numérica:

$$Salida = \frac{-1 * 0.2 - 0.5 * 0.3 - 0 * 0.7}{0.2 + 0.3 + 0.7} \approx -0.3$$

El valor de salida no difuso para el freno del vehículo se aplica con aproximadamente 30% del máximo rango posible.

CAPÍTULO 3

HERRAMIENTAS DE HARDWARE

3.1 PLC SIMATIC S7-300¹

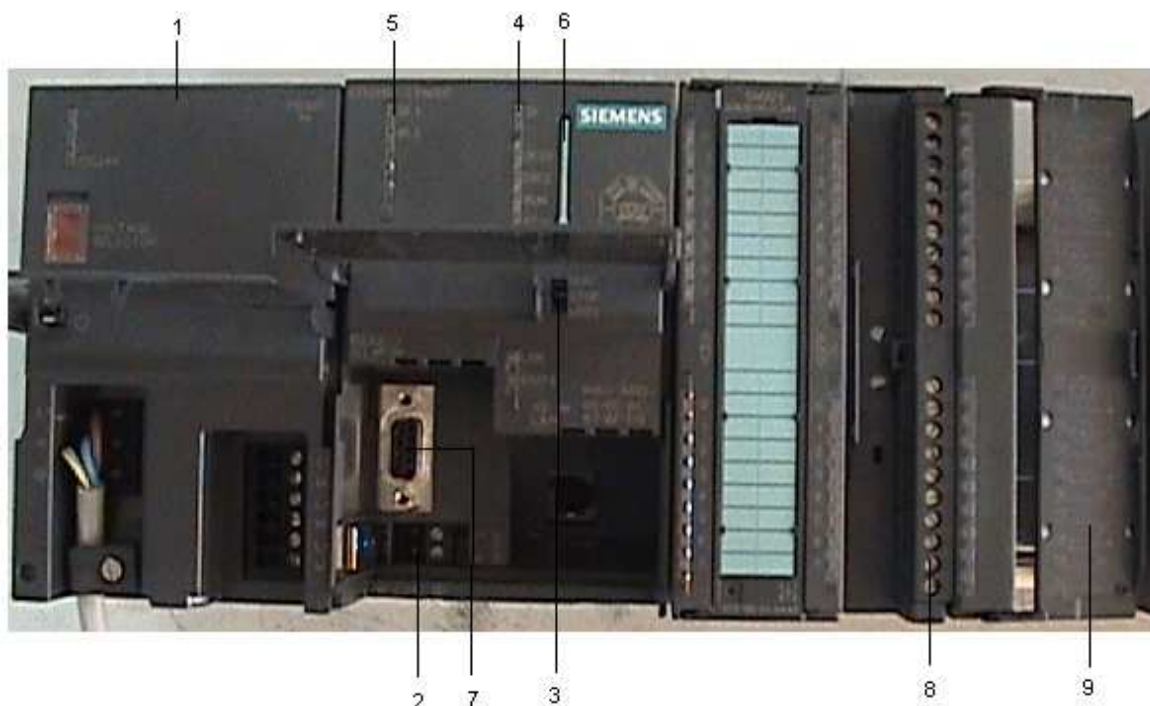
El PLC SIMATIC S7-300 se ha diseñado para las soluciones de sistemas innovadores con enfoque en la ingeniería industrial, y como un sistema de automatización universal, representa una solución óptima para las aplicaciones en configuraciones centralizadas y distribuidas.

El S7-300 pertenece al sistema de miniautómatas modulares para las gamas pequeña y media, permite disminuir los tiempos de ciclo y de respuesta, asegurando la adquisición y tratamiento de señales (analógicas o digitales) a cualquier velocidad y en cualquier forma en que se presenten.

Existen una variedad de módulos para una adaptación óptima de acuerdo a la tarea de automatización en particular. Posee gran flexibilidad en las aplicaciones pues permite realizar estructuras descentralizadas e interconexiones por red. Su sencilla instalación y la falta de necesidad de ventilación facilitan su uso. Si se requiere incrementar las tareas admite la expansión sin complicaciones ya que posee un bus plano integrado en la parte posterior.

¹ Siemens, SIMATIC S7-300 Mid-range PLC, <http://www2.sea.siemens.com/Products/Automation/Programmable-Controllers/SIMATIC-S7-300/?languagecode=en>, 2008-02-08, 2008-07-01.

El importante número de funciones integradas lo convierten en una herramienta muy potente. A continuación la figura. 3.1. presenta al SIMATIC S7-300.



- | | |
|------------------------------------|-------------------------------|
| 1. Fuente de alimentación. | 6. Tarjeta de memoria Micro. |
| 2. Conexión para 24V. | 7. MPI (Interface multipunto) |
| 3. Selector de modo. | 8. Conector frontal. |
| 4. Led de señalización de estados. | 9. Puerta frontal. |
| 5. Led de señalización de fallas. | |

Figura. 3.1. Autómata programable SIMATIC S7-300

El autómata programable se compone de²:

- Unidad central de proceso (CPU), constituye el "cerebro" del sistema y toma decisiones en base a la aplicación programada.
- Módulos para señales digitales y analógicas (I/O).
- Procesadores de comunicación (CP) para facilitar la comunicación entre el hombre y la máquina o entre máquinas. Se tiene procesadores de comunicación para conexión a redes y para conexión punto a punto.

²

- Módulos de función (FM) para operaciones de cálculo rápido.

Se puede hallar componentes que se adaptan a las exigencias de los usuarios:

- Módulos de suministro de energía, el S7-300 requiere una alimentación de 24 VDC. Por ésta razón, los módulos (fuentes) de alimentación de carga transforman la tensión de alimentación de 115/230 VAC en una tensión de 24 VDC. Los módulos de alimentación se montan a la izquierda junto a la CPU.
- Módulos de interfaces para conexión de *racks* múltiples en configuración multi-hilera

Pequeño, extremadamente rápido y universal son las características más importantes de éste PLC, además de su modularidad, sus numerosos módulos de extensión, su comunicabilidad por bus, sus funcionalidades integradas de visualización y operación así como su lenguaje de programación bajo entorno Windows.

3.1.1 CPU 315F-2 PN/DP³

Esta CPU tiene una velocidad 100 veces mayor a las convencionales (no necesita más de 0,3 ms para ejecutar 1024 instrucciones binarias y no mucho más al procesar palabras), posee una memoria para instrucciones de seguridad de 50K, 1024 entradas/salidas digitales, puede integrar 32 módulos dentro de un solo sistema (para tareas especiales se ofrecen módulos específicos), brinda alta potencia de cálculo con aritmética de hasta 32 bits en coma flotante y permite establecer comunicaciones a través de interfaces multipunto o puerto MPI. La

³

figura. 3.2. presenta la CPU 315F-2 PN/DP. La tabla. 3.1. muestra los datos técnicos de la CPU 315F-2 PN/DP.

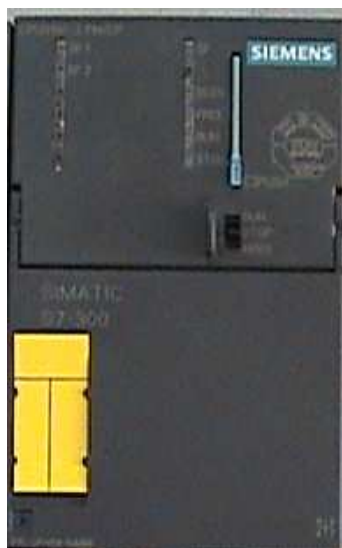


Figura. 3.2. CPU 315F-2 PN/DP

Tabla. 3.1. Datos técnicos CPU 315F-2 PN/DP

DATOS TÉCNICOS	CPU 315F-2 PN/DP
Instrucciones de seguridad	50 K
Memoria de trabajo	
- Integrados	256 Kb RAM
- Enchufables como Memory Card	64kbytes a 8Mbytes
Tiempos de Ejecución	0,3 ms / 1K instrucciones
Marcas	2048
Contadores	256
Temporizadores	256
Reloj Tiempo Real	Si
Direccionamiento Libre	Si
Bytes de Entradas/Salidas Digitales	1024
Bytes de Entradas/Salidas Analógicas	256
Respaldo de datos	
- sin pila	4 Kbytes; de marcas, contadores, temporizadores y datos.
- con pila tampón	Todos los bloques de datos.

El tamaño de la CPU es de 80mm de largo, 125mm de alto y 130mm de profundidad.

3.1.2 Comunicación

Este CPU dispone de un puerto PROFINET integrado. Las posibilidades de configuración y las prestaciones de la CPU 315F-2 PN/DP se corresponden con las de la CPU 315-2 DP, mientras que las prestaciones en cuanto a comunicación son similares a las de una CPU 317-2PN/DP. Adicionalmente se pueden encontrar las siguientes características⁴:

- Comunicación S7 para intercambio de datos entre PLCs (cargable con FBs/FCs).
- Combina una interfaz maestro/esclavo-MPI/PROFIBUS DP (Maestro MPI o DP o esclavo DP) de 12Mbits/s.
- Instalación multi-nivel hasta 32 módulos.
- Ciclo de bus DP equidistante.
- Automatización basada en componentes (CBA⁵) sobre PROFINET.
- PROFINET-proxy para dispositivos inteligentes sobre PROFIBUS DP en Automatización Basada en Componentes (CBA).
- Protocolo de transporte TCP/IP.
- Controlador PROFINET I/O para entradas y salidas distribuidas operando sobre PROFINET.

⁴ Modules Supported from STEP 7 V5.4 SP1 Onwards, http://www.siemens.co.kr/Automation/DATA/FAQ/PLC/200611/New_Modules_in_STEP7_V54_SP1.pdf, 2006-07, 2008-07-15.

⁵ La funcionalidad CBA integrada en la CPU 315F-2 PN/DP permite la modularización tecnológica de máquinas y plantas, lo que aporta una serie de ventajas importantes: se puede estandarizar la planta de mejor forma y se puede reutilizar parte del código o ampliar ésta de forma más sencilla.

La figura. 3.3. da un ejemplo de la CPU 315F-2 PN/DP trabajando dentro de una red a prueba de fallas y dentro de una red estándar, conjuntamente con otros dispositivos SIMATIC⁶.

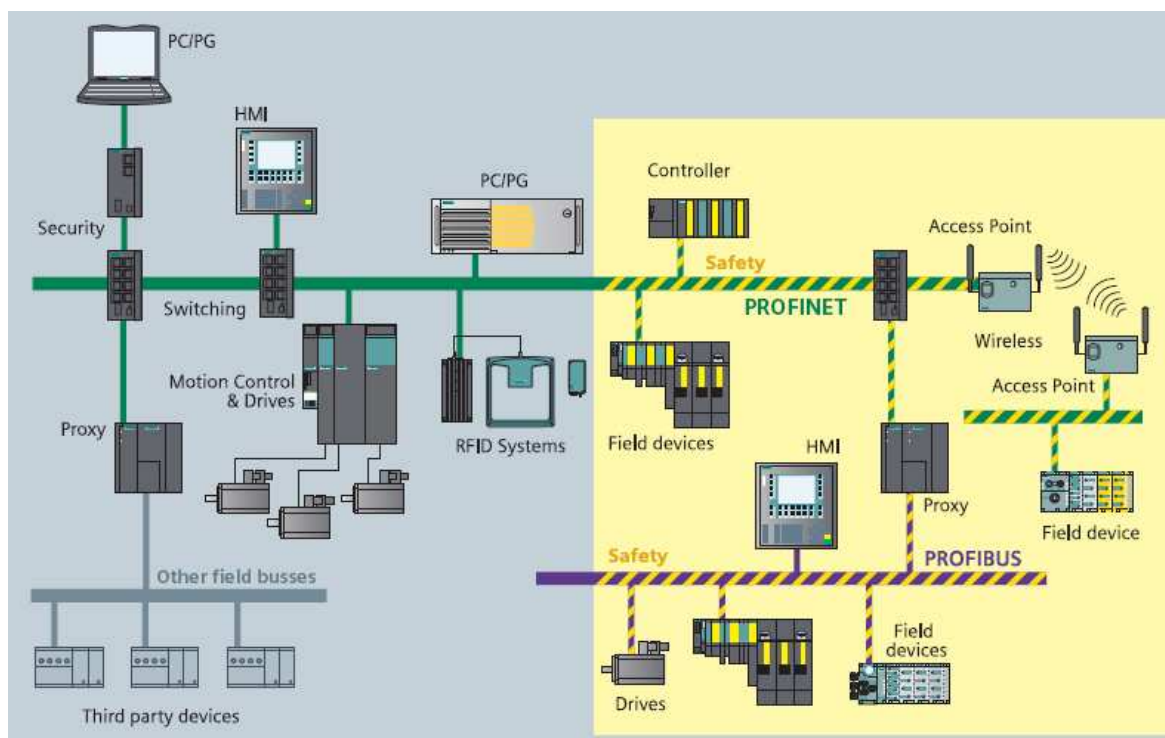


Figura. 3.3. Ejemplo de configuración de una red de trabajo para el CPU 315F-2 PN/DP

3.2 TIPOS DE MÓDULOS⁷

Los módulos difieren en cuanto al número de canales, rangos de voltaje y corriente, aislamiento eléctrico, funciones de diagnóstico y alarma, etc.

En cuanto a las dimensiones de los módulos, sus medidas de ancho, alto y profundidad son 40mm x 125mm x 130mm, respectivamente. Cabe recalcar que se pueden conectar hasta 32 módulos, esto mediante tres *racks* de expansión conectados al *rack* central (8 módulos en cada *rack*). La figura. 3.4. indica los

⁶ Siemens, SIMATIC fail-safe controllers, http://www.siemens.dk/ccmi/bu/ad/download/pdf/service_support/information/brochure/20070119/s7300f_e_12-2005.pdf, página 6, 2005-12, 2008-07-15.

⁷ Siemens, SIMATIC S7-300 I/O Modules, http://www.automation.siemens.com/simatic/controller/html_76/produkte/simatic-s7-300-io.htm, 2008-07-04.

módulos de entrada/salida. La tabla. 3.2. da a conocer algunos tipos de módulos de entrada y salida.



Figura. 3.4. Módulos de I/O

Tabla. 3.2. Tipos de módulos de I/O

TIPOS DE MÓDULOS		
Módulos de Entradas Digitales		
Rango de Voltaje	Canales	
24V DC	16, 32	
48-125V DC	16	
120/230V AC	8, 16, 32	
Módulos de Entradas/Salidas Digitales		
Rango de Voltaje	Canales	
24V DC	8 ó 16 DI/DO	
24V DC	8 DI y 8 DX (parametrizado como DI ó DO)	
Módulos de Salidas Digitales		
Rango de Voltaje	Rango de Corriente	Canales
24V DC	0.5 A	8, 16
24V DC	2 A	8
48-125V DC	1.5 A	8
120/230V AC	1 A	8, 16, 32
120/230V AC	2 A	8
Módulos de Entradas Analógicas		
Rango de Medida	Resolución	Canales
Voltaje	hasta 16 bits	2, 8
Corriente (también HART)	hasta 16 bits	2, 8
Resistencia	hasta 16 bits	1, 4, 8

Elementos de Termocupla	hasta 16 bits	2, 8
Termómetro de resistencia	hasta 15 bits	1, 4, 8
Módulos de Entradas/Salidas Analógicas		
Rango de Medida	Resolución	Canales
Voltaje	hasta 13 bits	2, 4
Corriente	8 bits	4
Resistencia	13 bits	4
Termómetro de resistencia	15 bits	4
Voltaje	14 bits	4
Corriente	14 bits	4
Módulos de Salidas Analógicas		
Rango de Medida	Resolución	Canales
Voltaje	hasta 16 bits	2, 4, 8
Corriente (también HART)	hasta 16 bits	2, 4, 8

3.3 TARJETA DE MEMORIA MICRO (MMC)⁸

La CPU dispone de una ranura para insertar una tarjeta de memoria Micro, la cual es como una memoria de programa y de datos, al mismo tiempo es un redundante respaldo.

Se puede utilizar para guardar la totalidad del proyecto de STEP7, incluyendo los comentarios y símbolos del mismo. Si se desea hacer uso de estas posibilidades será necesario una MMC de mayor tamaño. En la tabla. 3.3. se mencionan algunas tarjetas de memoria con su respectiva capacidad.

Tabla. 3.3. Tamaños de memoria para las MMC

Tamaño de Memoria
128 kBytes
512 kBytes
2 MBytes
4 MBytes
8 MBytes

⁸

El Micro Memory Card (MMC) se puede usar durante la operación para almacenar y acceder a los datos. El sistema de prueba de tarjetas de memoria Micro permite escribir más ciclos que las tradicionales tarjetas de memoria.

La CPU requiere una tarjeta de memoria micro obligatoriamente para trabajar.

CAPÍTULO 4

HERRAMIENTAS DE SOFTWARE

4.1 STEP7 PROFESSIONAL¹

STEP7 Professional es un software de programación y configuración diseñado para uso profesional con controladores SIMATIC. Posee poderosas funciones y herramientas para la variedad de tareas involucradas en un proyecto de automatización.

Dicho software es la herramienta de programación necesaria para los sistemas de automatización:

- SIMATIC S7 300
- SIMATIC S7 400
- SIMATIC WIN AC

Este software incluye todos los lenguajes de programación que obedecen el estándar internacional IEC 61131-3, y por tanto habilita la estandarización del cruce de compañías y ayuda a ahorrar en los gastos de diseño de software de alto nivel.

El funcionamiento intuitivo y el uso de idiomas estándar lo hacen fácil para que programadores y personal de mantenimiento se familiaricen con el software.

¹ Siemens, Step 7 Professional Programming and configuring according to IEC 61131-3, <http://www2.sea.siemens.com/NR/rdonlyres/D50EFC60-1EA6-43AF-A741-26402740B8C4/0/STEP7PROANDBASISPRODUCTBRIEF.pdf>, 2003-04, 2008-07-10.

Apoya al usuario a través de todas las fases de un proceso de desarrollo para las soluciones de automatización como:

- Instalación y manejo de proyectos.
- Configuración y asignación de parámetros.
- Direccionamiento de símbolos.
- Generación de programas para el sistema designado SIMATIC S7
- Carga de programas en el sistema designado.
- Pruebas de la planta de automatización.
- Diagnósticos de fallas de la planta.

Los principales componentes de STEP7 Professional son:

- **Administrador SIMATIC**, para el manejo de todas las herramientas y datos de un proyecto de automatización.
- **Configuración de Hardware**, para ordenamiento y parametrización del mismo.
- **Editor de programas**, para la creación y comprobación del programa de usuario.
- **NetPro**, para montar una transferencia de datos sobre MPI o PROFIBUS/PROFINET.
- Integra **diagnóstico del sistema**, para obtener una vista general del estado del sistema de automatización.
- Documentación del proyecto de manera estándar y sencilla con **DOCPRO**.
- Control PID y control de temperatura PID para parametrización PID simple o controladores de temperatura.
- Prueba de software sin controlador con **S7-PLCSIM** (elemento de STEP7 Professional).
- Creación de programas para controladores a prueba de fallas.
- **Interfaz de llamada de herramientas** (ICT) para la creación de sistemas de ingeniería de otros fabricantes.
- **Interfaz de comando abierto** para importación y exportación de datos desde otras herramientas Windows.
- El **Logon SIMATIC** para gestión de usuarios centralizada.

A continuación se describirá algunos de los componentes antes mencionados.

El **administrador SIMATIC** maneja todos los datos pertinentes a un proyecto de automatización. Además permite crear, copiar, descargar y archivar proyectos.

Todas las aplicaciones pueden ser ejecutadas desde el Administrador SIMATIC, el cual posee un asistente que ayuda a crear el proyecto de STEP7.

Los proyectos se estructuran de tal modo que permiten depositar de forma ordenada todos los datos y ajustes necesarios para su tarea de automatización. Dentro de dicho proyecto, los datos se estructuran por temas y se representa en forma de objetos. Los diversos tipos de objetos están vinculados en el Administrador SIMATIC directamente a la aplicación necesaria para poder editarlos, es decir que permite acceder a todas las funciones y ventanas de STEP7. Gracias a ello no es necesario recordar la aplicación que se debe arrancar cuando se desee editar un objeto determinado.

La figura. 4.1. muestra la pantalla principal del Administrador SIMATIC.

Dentro del administrador se puede encontrar funciones como:

Multiproyectos, que permite crear un proyecto y procesarlo por usuarios diferentes simultáneamente. La convergencia de los proyectos es soportada por el sistema.

Gestión de textos, que permite la creación y manejo de textos en varios lenguajes. Los textos a ser trasladados son exportados desde STEP7, procesados usando un editor ASCII o un programa de hoja de cálculo (Excel), y posteriormente importados de regreso dentro de STEP7.

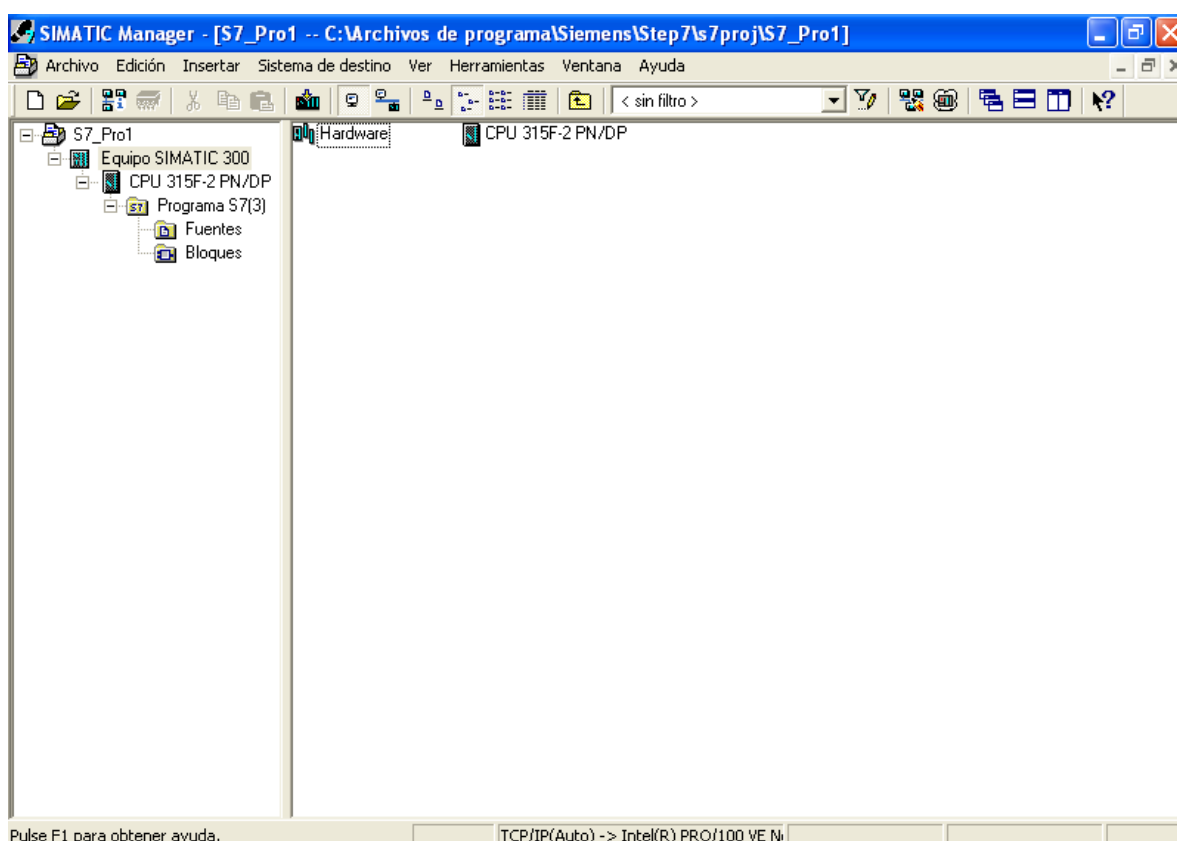


Figura. 4.1. Administrador SIMATIC

Es posible guardar los datos de un proyecto en la CPU (PLC), además del programa de aplicación actual, los datos del proyecto completo pueden ser guardados en la tarjeta de memoria de la CPU. Estos datos están entonces disponibles en el sitio para propósitos de servicio.

Otros de los componentes de STEP7 Professional es **Configuración de Hardware**, con él las CPUs y los módulos de SIMATIC no necesitan cambios mecánicos ni ajuste de tornillos. Todas las configuraciones son implementadas de manera centralizada usando el software, es decir que el hardware es configurado y parametrizado en la herramienta HW Config (Configuración de Hardware).

Esta opción posee opciones especiales como enlace a Internet, con la cual la información más actual con respecto al hardware usado puede descargarse siempre que se requiera accediendo a la información de apoyo del producto en

Internet.

También permite integrar componentes de hardware directamente dentro de STEP7 mediante Internet sin necesidad de un paquete de servicio.

Por otro lado se puede visualizar gráficamente los puertos de comunicación y la distribución de las entradas y salidas.

El **editor de programas** es la interfaz de programación para el usuario. El paquete estándar de STEP7 Professional comprende los lenguajes de programación probados LAD o KOP (diagrama de escalera), FBD o FUP (diagrama de bloques de funciones) y STL o AWL (lista de instrucciones).

El software puede ser ampliado con una extensa gama de paquetes opcionales, por ejemplo:

- S7-GRAPH para la programación gráfica controles secuenciales.
- S7-SCL, el lenguaje de alto nivel con el que se puede resolver tareas complejas sin ningún problema.
- S7-PLCSIM para simulación desconectada de la solución de automatización.

La herramienta **NetPro** de STEP 7 Professional habilita la configuración del sistema de comunicación. Aquí los enlaces de comunicación entre estaciones individuales son configurados gráficamente. NetPro contiene todos los controladores requeridos para PROFINET y PROFIBUS CPs.

S7-PLCSIM es un sistema de simulación de un controlador real que proporciona un apoyo eficaz para el desarrollo de programas y por consiguiente para la aplicación real. Permite descubrir errores y optimizar secciones en el programa.

La **interfaz de llamada de herramientas** facilita la integración de herramientas de ingeniería de otros fabricantes dentro de STEP 7. Habilita de una manera simple la llamada de programas o configuración de instrumentos como sensores o actuadores manejados por medio de controladores SIMATIC.

SIMATIC Logon ayuda a crear accesos con privilegios para proyectos y librerías de STEP 7. Determina que usuario puede usar la licencia y quien podría transferirla desde un servidor.

El software de SIMATIC posee muchos otros paquetes complementarios, los cuales pueden ser adquiridos de acuerdo a la necesidad de automatización.

4.2 WINCC (Windows control center)²

SIMATIC WinCC es un sistema de supervisión sobre PCs estándar en un entorno operativo bajo Windows 2000 y XP. Es un sistema SCADA dotado de potentes funciones de mando y supervisión para procesos automáticos.

WinCC posee una arquitectura absolutamente abierta que permite crear aplicaciones propias sobre las bases de WinCC. Puede trabajar en combinación con otros programas para crear soluciones de mando y supervisión optimizadas que cumplen las exigencias prácticas.

Es un software moderno, ofrece interfaces cómodas para el usuario, es fácilmente configurable, escalable desde las tareas más sencillas a las más complejas y que sirven de plataforma para IT & Business Integration.

WinCC puede conmutar entre los idiomas alemán, inglés, francés, español e italiano. La versión asiática también soporta ideogramas para China, Corea, Japón o Taiwán. Los programas pueden ser configurados en varios idiomas

² Siemens, Simatic HMI WinCC Visualización de proceso y plataforma para IT y Business Integration., http://www.automation.siemens.com/salesmaterial-as/brochure/es/sb_wincc_v6_s.pdf, 2003, 2008-07-18.

Runtime (simultáneamente en varios idiomas conmutables en modo online).

Sirve de plataforma HMI para diversos sistemas de visualización y de control de procesos de Siemens (por ejemplo, Producción y distribución de energía, lógica *Fuzzy*) y de otros fabricantes. WinCC permite la integración con sistemas no pertenecientes a SIMATIC.

Posee funciones de mando y supervisión a la medida de las necesidades industriales para la visualización gráfica completa de los procedimientos y estados del proceso, señalización y confirmación de alarmas, archivo de los valores de medida y mensajes, listado de todos los datos de proceso y de archivo, gestión de los usuarios y sus autorizaciones de acceso.

Está equipado con funciones de configuración sencilla y eficiente, por ejemplo el editor gráfico es cómodo y orientado al objeto, posee extensas bibliotecas, modularidad, modificaciones rápidas por medio de la configuración online, herramienta de configuración para el tratamiento de grandes cantidades de datos, transparencia gracias a la lista de referencias cruzadas.

SIMATIC WinCC ofrece escalabilidad continua y homogénea desde la pequeña solución monopuesto hasta una arquitectura cliente/servidor redundante con registro histórico (Historian) integrado y puestos de operador en la Web.

WinCC es un componente del concepto TIA (Totally Integrated Automation, integración completa de los componentes de automatización individuales), es decir que WinCC, por ejemplo, accede directamente a la configuración de variables y de mensajes de su autómatas SIMATIC y también usa sus parámetros de comunicación. Los datos variables simplemente se introducen una sola vez y pueden ser gestionados y actualizados de forma centralizada. Gracias al acceso directo a los símbolos STEP 7 se evita también un direccionamiento incorrecto en WinCC.

WinCC dispone de una serie de editores con los cuales se pueden configurar individualmente las funcionalidades necesarias para la aplicación pertinente. Entre los editores de WINCC con su tarea o funcionalidad *Runtime* configurable respectiva están:

- **WinCC Explorer**, es el centro de control de WinCC. Gestión centralizada del proyecto para el rápido acceso a todos los datos y los parámetros centralizados del proyecto.
- **Graphics Designer**, sistema gráfico para la configuración de las imágenes del proyecto y los diálogos, permite manipular y dinamizar las propiedades de objeto.
- **Alarm Logging**, editor para la configuración del sistema de alarmas y mensajes. Permite la adquisición y el archivo de eventos con posibilidad de mando y supervisión; libre selección de las clases de mensaje, los contenidos de mensaje, informes y listados
- **Tag Logging**, editor para la configuración del sistema de archivo. Archivo de valores de proceso para la adquisición, compresión y el almacenamiento de valores de medida, por ejemplo para la presentación de tendencias en forma de tablas y su procesamiento posterior.
- **Report Designer**, Sistema de informes y listados para la documentación controlada por eventos de mensajes, operaciones y datos de proceso actuales en forma de informes de usuario o documentación de proyecto.
- **User Administrador**, herramienta para una cómoda gestión de usuarios y sus derechos de acceso.
- **Global Scripts**, funciones de procesamiento con funcionalidad ilimitada gracias a la utilización del compilador ANSI-C.
- **Herramientas diversas**, Text Library, CrossRefence, ProjectDuplicator, Picture Tree Manager, Lifebeat Monitoring, Smart Tools.

La figura. 4.2. presenta al explorador de WinCC.

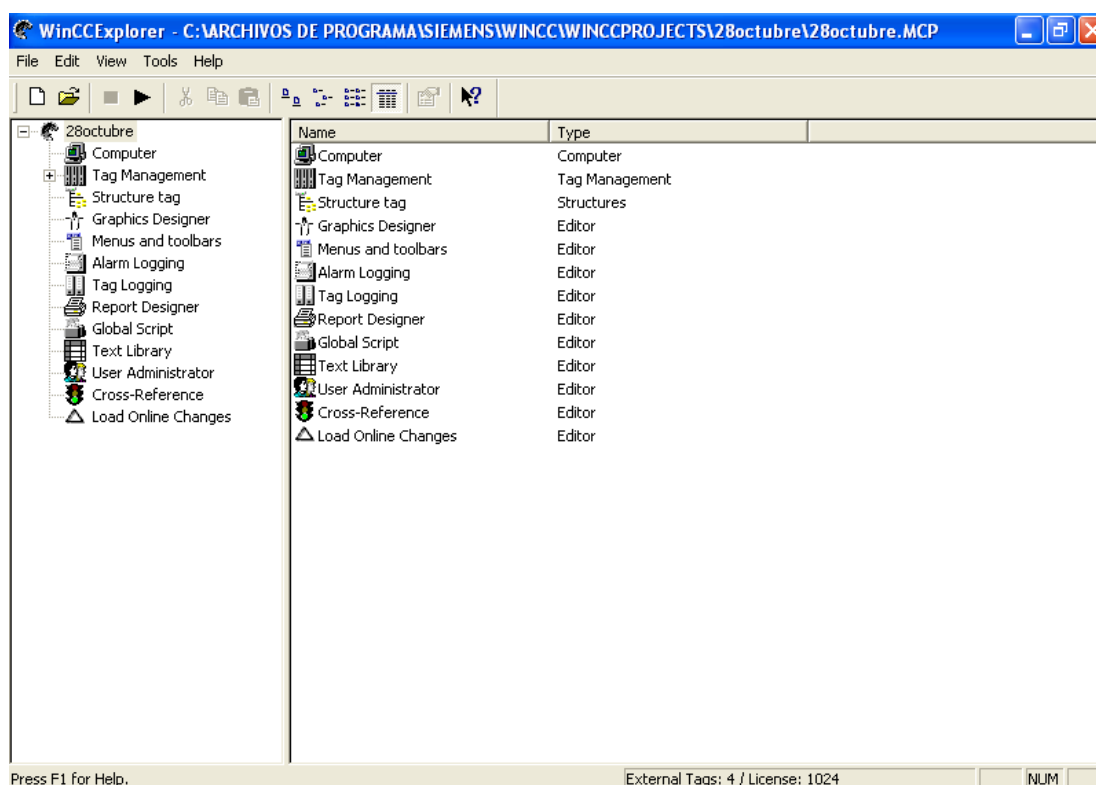


Figura. 4.2. Explorador WinCC

SIMATIC WinCC ofrece la posibilidad de crear menús específicos para un proyecto o un sector industrial, generar objetos personalizados y almacenarlos en una biblioteca, automatizar la configuración y utilizar una serie de interfaces y herramientas para las tareas de definición complejas:

- Asistentes de configuración “Wizards”.
- Bibliotecas y bloques.
- Ingeniería de grandes proyectos.
 - La lista de referencias cruzadas y la visualización centralizada de las propiedades de imagen permiten disponer de informaciones siempre actualizadas.
 - Herramienta de configuración para el tratamiento de grandes volúmenes de datos.
 - Interfaces de importación y exportación.
- Configuración de aplicaciones multilingües.
- Comprobación, puesta en marcha, mantenimiento.
 - Comprobación de proyectos mediante la simulación en *runtime*.
 - Modificación sencilla mediante la configuración online.

- Configuración *offline* y carga de la modificación a un sistema *online* (carga de modificaciones *online*).

4.3 FUZZYCONTROL++³

FuzzyControl++ es una herramienta de configuración para lógica difusa y bloques de función estándar. Ofrece soluciones para regulaciones no lineales y pronósticos de comportamiento de procesos matemáticamente complejos en automatización de procesos, difíciles o imposibles de realizar con herramientas estándar.

Permite desarrollar y configurar eficazmente sistemas difusos para la automatización de procesos técnicos. La habilidad y el conocimiento empírico pueden describirse verbalmente y trasladarse directamente a los controladores, las regulaciones, la detección de patrones, las lógicas de decisión, etc.

Posee una interfaz Windows gráfica fácil de usar. Los bloques de la función están disponibles para el SIMATIC S7-300 y S7-400 así como para SIMATIC WinCC.

La herramienta de configuración (en la PC/PG) se comunica con el SIMATIC S7 a través del bus MPI usando el SOFTNET S7 para software de comunicación PROFIBUS, y para la comunicación con WinCC lo hace a través de un Control ActiveX.

La herramienta de configuración FuzzyControl++ ayuda a crear un sistema difuso. Para usar la herramienta es suficiente tener conocimientos básicos sobre sistemas de estas características, ya que no deben realizarse ajustes matemáticos ni de regulación. Durante la configuración, se puede consultar la amplia ayuda en pantalla.

³ Siemens, FuzzyControl++ for efficient automation of technical processes, http://info.industry.siemens.com/Downloads/start_download.asp?lang=e&file=PB%5F567%5FFuzzyControl%2B%2B%5Fen%2Epdf&did=567&lid=1&kid=10, 2007, 2008-07-15.

El proceso de creación de un sistema difuso bajo FuzzyControl++ básicamente involucra las etapas siguientes:

1. Definición de la configuración externa del sistema difuso.

Esto incluye el número de entradas y salidas y el sistema especificado (controlador programable) para la implementación del comportamiento de las entradas y salidas en el proceso práctico.

2. Definición de las funciones de pertenencia.

Se debe definir los máximos y los mínimos, el número y propiedad de las funciones de pertenencia para cada entrada y salida del sistema difuso.

3. Creación de la regla base.

Aquí se debe resolver y editar la regla base del sistema difuso para el proceso práctico.

FuzzyControl++ permite probar los proyectos creados con el sistema especificado tanto en modo conectado (*online*) como desconectado (*offline*). Entre las funciones que existentes están:

Representación gráfica en 3D de las entradas y las salidas, aquí se puede ver el comportamiento de dos de las entradas y una salida del sistema difuso en una representación espacial.

Otra de las herramientas del software es el *plotter* de curva, el cual proporciona una pantalla gráfica del comportamiento de las entradas y salidas de un sistema difuso. De existir conexión con el sistema especificado, el *plotter* de curva muestra el proceso actual de las variables.

Durante la operación desconectada, las señales de entrada pueden ser generadas automáticamente. Un generador de curva está integrado dentro del *plotter* de curva, el cual puede simular ciertas señales trazadas en el tiempo. Las curvas trazadas pueden ser guardadas en un archivo. El simulador de curva proporciona varios tipos de señal.

La ventana de actividad de regla, muestra la actividad de las reglas individuales para la salida seleccionada como función de las variables de entrada en forma de un diagrama. La figura. 4.3. hace referencia a las ventanas existentes en FuzzyControl++.

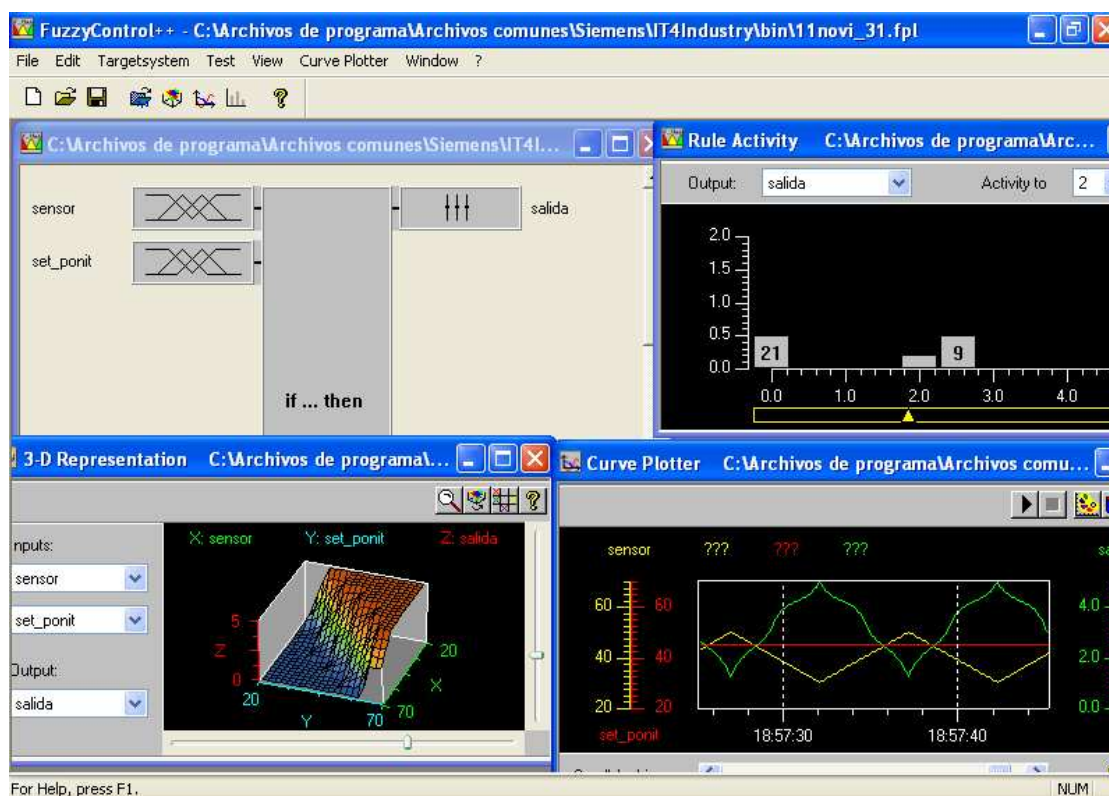


Figura. 4.3. Ventanas de FuzzyControl++

4.3.1 FUZZYCONTROL++ V5: PROBLEMAS Y SOLUCIONES

FuzzyControl++ V5, es la segunda versión generada por SIEMENS, básicamente ofrece dos formas de comunicación, una de ellas es directamente entre el SIMATIC S7, y la otra es con WinCC mediante controles ActiveX desde la versión 6 en adelante.

PROBLEMAS

Debido a la falta de medios de comunicación físicos o habilitaciones como: adaptador MPI/USB, tarjeta PCI 5611, o licencia de comunicación Ethernet no fue

posible comunicar directamente el PLC con el software de control difuso. Se probó con adaptadores de otras marcas que cumplieran con los requisitos eléctricos RS485 que utiliza MPI, pero no se obtuvo ningún resultado positivo.

Por tales razones se utilizó la segunda opción de comunicación. FuzzyControl++ está provisto de una herramienta llamada FuzzyControl++ OCX, cuya instalación permite trabajar en WinCC con controles ActiveX.

Después de realizada la instalación y de poseer las licencias necesarias (licencia de WinCC y licencia de FuzzyControl++ para WinCC) se puede insertar un bloque de control difuso en el Graphics Designer (ver figura. 4.4.) y cargarlo con el archivo *.flp realizado en FuzzyControl++.



Figura. 4.4. Objeto de control FuzzyControl++

En los manuales que vienen incluidos en el paquete de software, el bloque de control insertado en el diseñador gráfico, posee dos modos de funcionamiento: modo automático (solo este viene detallado en el manual) y modo manual. La figura. 4.5 muestra la pantalla de propiedades de FuzzyControl++.

Modo automático

Este supone un correcto funcionamiento sin necesidad de ninguna señal extra más que las entradas al controlador.

Durante la comprobación de su funcionamiento surgieron problemas que se enumeran a continuación:

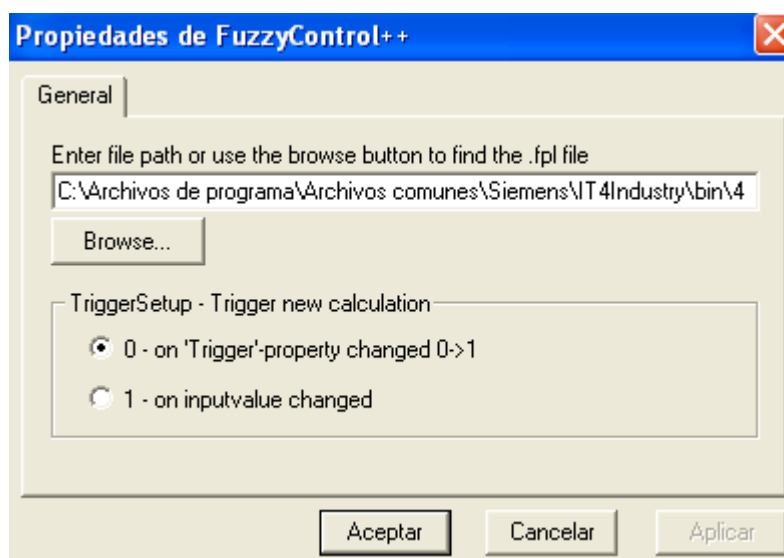


Figura. 4.5. Ventana de Propiedades de FuzzyControl++

1. Si las entradas se direccionan mediante variables internas (*tags* internos) al controlador difuso, la salida no responde.
2. Si las entradas se conectan directamente al bloque de control difuso, la salida no responde.
3. Si se programa las entradas mediante instrucciones en C, la salida no responde.

Se mantuvieron conversaciones con personas de SIEMENS en Alemania, las cuales no supieron responder al problema en cuestión, indicaron que aquellos direccionamientos incluidos en el manual no funcionaban, y se limitaron a esta indicación sin dar solución al problema.

Modo manual

Al probar con el modo manual, cambiando directamente desde WinCC el disparador (*trigger*) no se logró ningún resultado.

SOLUCIONES

La búsqueda de soluciones llevó a la idea de enviar una señal de disparo mediante el Simulador de Tags para WinCC. Este concepto permitió visualizar los valores de salida correspondientes a las entradas, pero si bien devolvía el valor

correcto, un valor basura era también mostrado entre el cambio de 0 a 1 del *trigger*. Sumando a lo anterior el hecho de que el tiempo de respuesta era lento esta solución no fue la más conveniente.

Por último se mandó un disparador desde el STEP7 Professional, arrojó buenos resultados, el valor basura desapareció.

Al usar *timers* de 1 segundo la respuesta todavía resultaba algo lenta por lo que se programaron *timers* de 250 ms, obteniendo una respuesta muy buena.

Únicamente presentó un pequeño inconveniente que es la detección del flanco ascendente debido a la velocidad utilizada, pero si este es detectado todo queda superado.

Esta solución trabaja aún si el bloque de control insertado en el diseñador gráfico está en modo manual o automático, y responde a cualquier tipo de direccionamiento.

CAPÍTULO 5

SISTEMA DE ENTRENAMIENTO

5.1 DEFINICIÓN DE LA APLICACIÓN

5.1.1 Generalidades

La finalidad de esta aplicación es el control de temperatura mediante la utilización de un sistema difuso diseñado en el software FuzzyContol++.

Se podrá monitorear y manejar la planta mediante el computador a través de una interfaz muy sencilla realizada en WinCC.

Para la obtención de datos y control del sistema se utilizará un transmisor de temperatura IC, el mismo que trabajará en el rango de 20 °C a 70 °C.

En la interfaz mostrada en pantalla, el usuario podrá manipular a su criterio el encendido o apagado del controlador difuso mediante el cambio de estado del activador o *trigger* ($0 \rightarrow 1$), también se podrá visualizar la temperatura actual del cilindro (variable procesada), el *set point* deseado y el valor de la salida (variable manipulada).

Se podrá ingresar la temperatura requerida, la cual tendrá una tolerancia de ± 2 °C, y para finalizar se presentará las gráficas de tendencia de las variables. La figura. 5.1. presenta el diagrama del sistema controlador difuso.

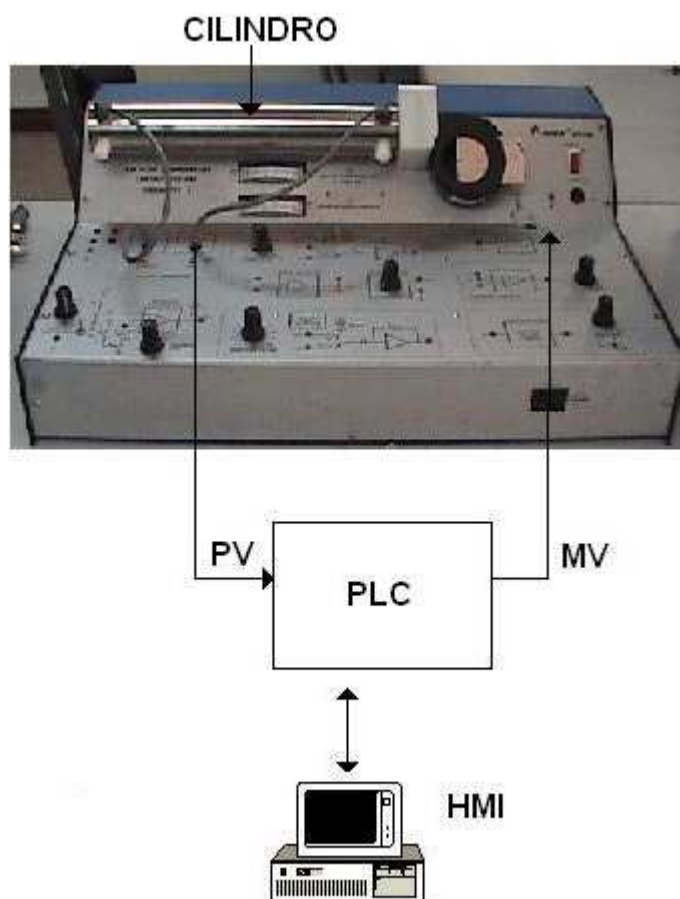


Figura. 5.1. Diagrama del sistema controlador difuso

5.1.2 Descripción del controlador

Este controlador utiliza el método de control difuso, los datos de la temperatura del cilindro son mandados por el sensor IC (0 a 5V) e ingresan al controlador a través de una entrada analógica. A su vez el *set point* es ingresado por el usuario mediante un campo de entrada en la HMI.

El controlador difuso espera el cambio de estado del *trigger* para entrar en funcionamiento, y una vez ocurrido esto devuelve a la salida un valor de voltaje (0 a 5V) que actuará directamente sobre el elemento de calentamiento para conseguir de este modo el valor de temperatura que se solicita para el cilindro. Los rangos de temperatura se encuentran entre 20°C y 70°C.

Así a medida que varía el voltaje en el elemento de calentamiento la temperatura en el cilindro será la solicitada.

Una interfaz de visualización (HMI) permite controlar y monitorear la temperatura del cilindro en cualquier PC. En esta interfaz se verán las curvas de las variables tanto de la temperatura actual en el cilindro, el *set point* y el voltaje enviado al elemento de calentamiento. La respuesta que proporciona la HMI es en tiempo real del proceso, además esta valida los rangos de entrada.

5.2 PROGRAMACIÓN

5.2.1 Código del Programa FuzzyControl++

En la figura. 5.2. se puede ver en general la ventana de programación de FuzzyControl++. Aquí se debe determinar el número de entradas y salidas que va a contener el programa, para este caso las entradas son: sensor y *set_point*, y solo existe una salida.

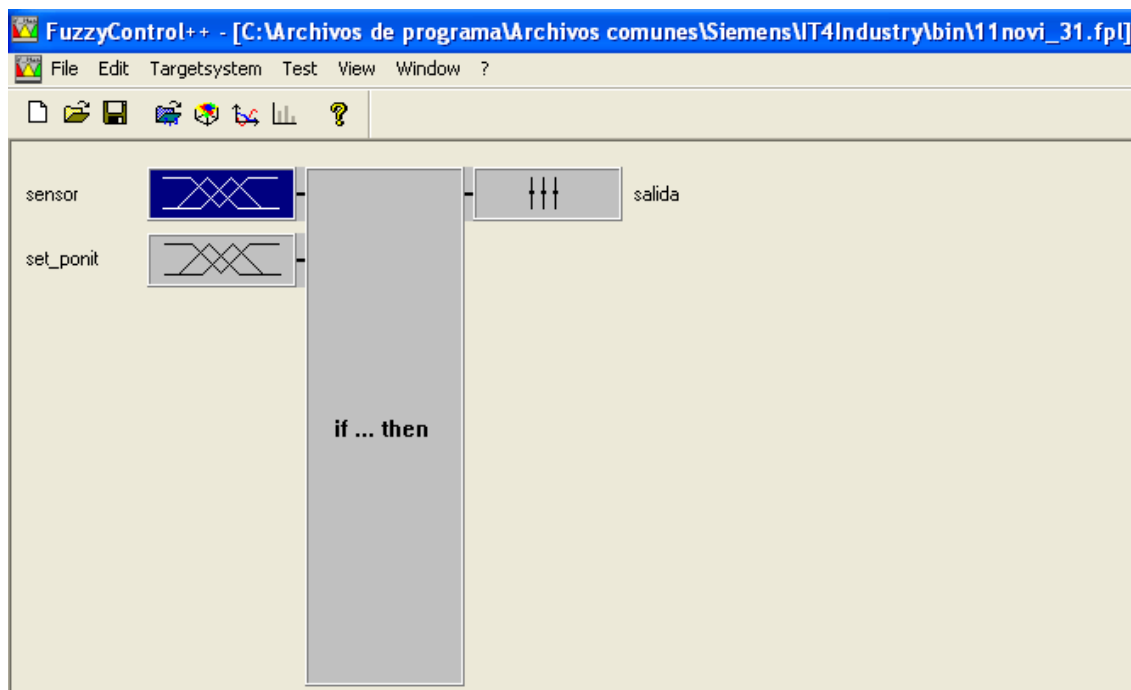


Figura. 5.2. Vista general de la ventana de programación

La figura. 5.3. muestra la ventana de propiedades para la entrada sensor, las funciones de pertenencia utilizadas son de tipo triangular, esto se debe a que ellas permiten manejar de mejor manera los rangos de temperatura para esta aplicación.

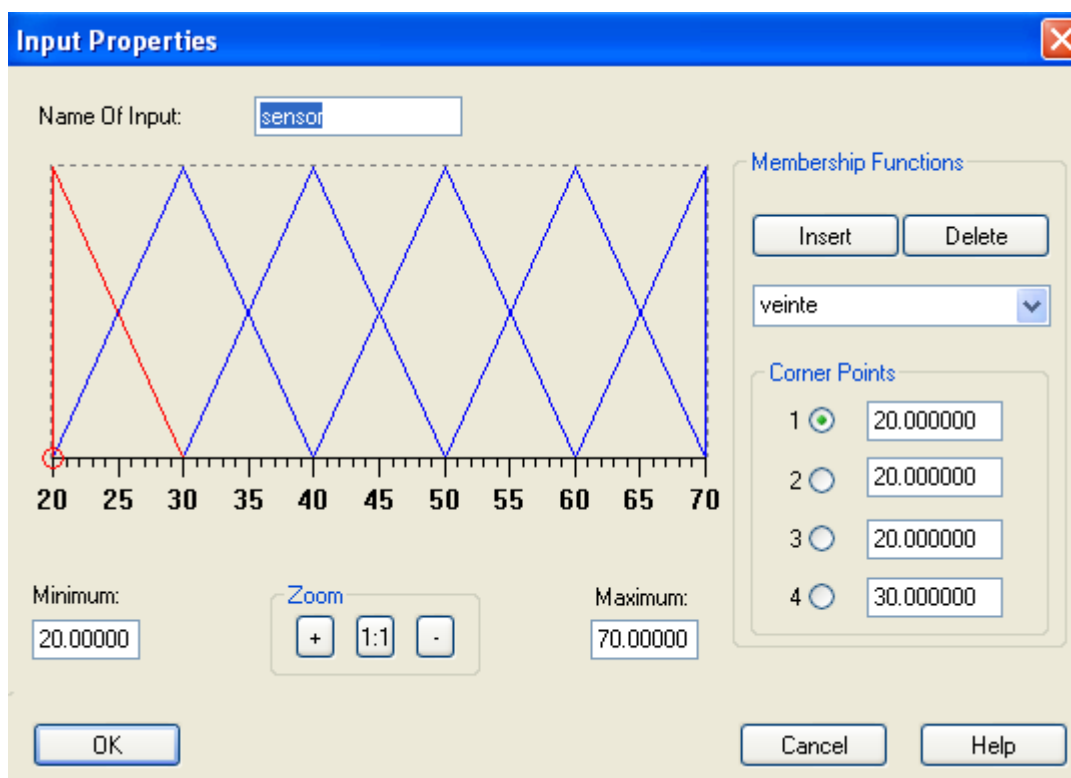


Figura. 5.3. Funciones de pertenencia para la entrada sensor

Para la segunda entrada *set_point*, al igual que en la entrada sensor, se programaron 6 funciones de pertenencia de tipo triangular con sus respectivos rangos de temperatura (figura. 5.4.).

En la tabla. 5.1. se puede ver las funciones de pertenencia creadas tanto para la entrada sensor como para la entrada *set_point* con sus correspondientes rangos de temperatura. Cabe recalcar que la cima de la función de pertenencia equivale al 100% del grado de pertenencia del valor relativo.

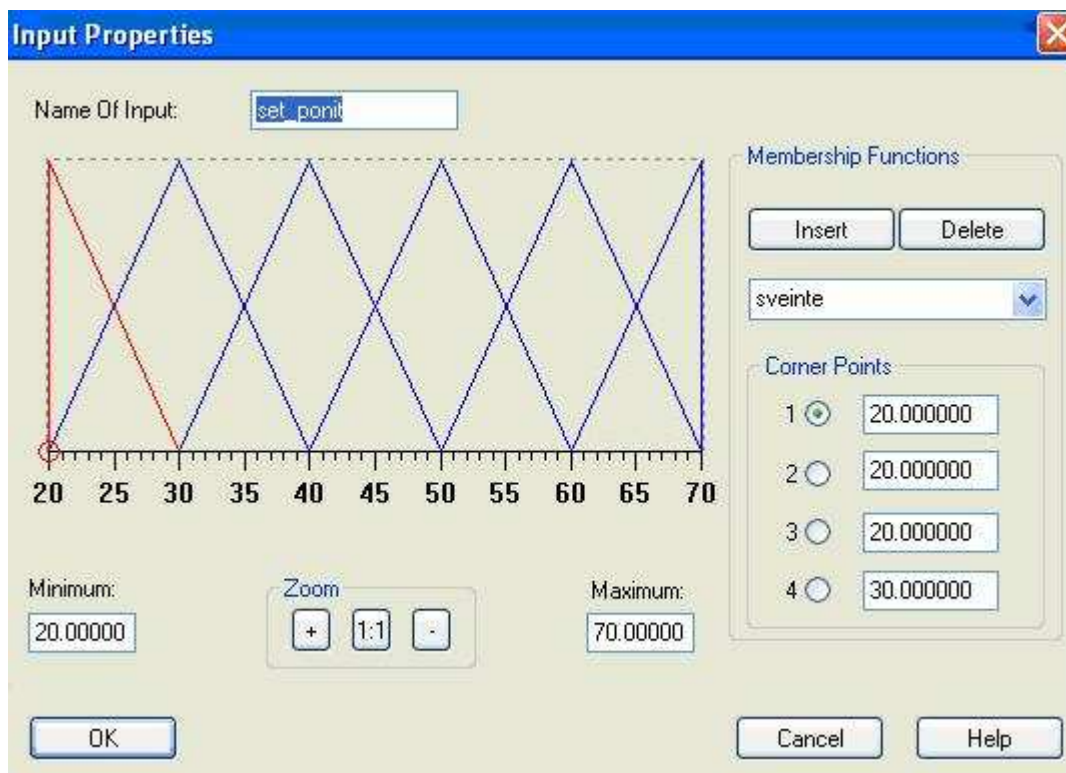


Figura. 5.4. Funciones de pertenencia para la entrada set_point

Tabla. 5.1. Funciones de Pertenencia tanto para la entrada sensor y set_point

Función de Pertenencia	Rango de Temperatura
veinte	20-30 (20 es el valor temperatura con el 100% de grado de pertenencia de esta función)
treinta	20-40 (30 es el valor de temperatura con el 100% de grado de pertenencia de esta función)
cuarenta	30-50 (40 es el valor de temperatura con el 100% de grado de pertenencia de esta función)
cincuenta	40-60 (50 es el valor de temperatura con el 100% de grado de pertenencia de esta función)
sesenta	50-70 (60 es el valor de temperatura con el 100% de grado de pertenencia de esta función)
setenta	60-70 (70 es el valor de temperatura con el 100% de grado de pertenencia de esta función)

Para la variable salida, se programaron 7 funciones de pertenencia de tipo *singleton* (FuzzyControl++ solo permite este tipo de funciones en las salidas),

cada una de ellas posee un único valor que corresponde al 100% del grado de pertenencia de dicha función (figura. 5.5.).

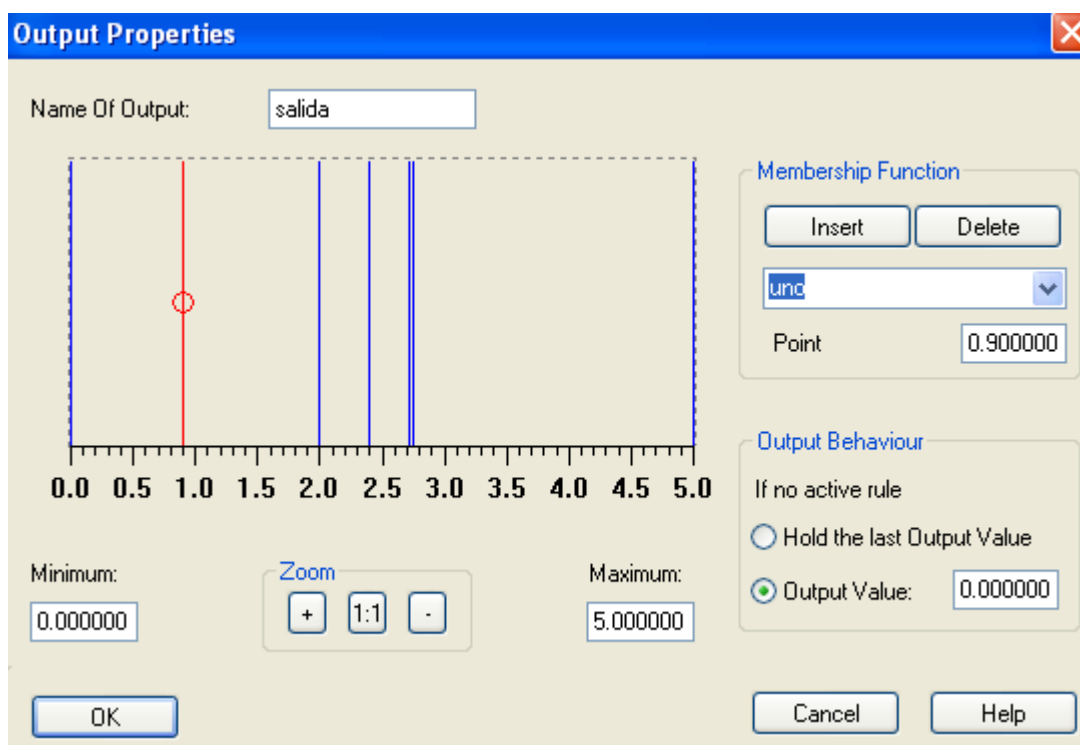


Figura. 5.5. Funciones de pertenencia para la salida

Los valores para cada función son dados de acuerdo a lo que se requiera a la salida, es decir que para otorgar el valor de salida a la función es necesario que el programador sepa o tenga la experiencia necesaria para determinar el valor más conveniente. Seguidamente se muestra en la tabla. 5.2. los valores para cada función de pertenencia de la salida, estos se los obtuvo al alimentar manualmente al elemento de calentamiento y ver que con dicho voltaje el medidor de temperatura marcaba la temperatura correcta:

Tabla. 5.2. Valores de las funciones de pertenencia para la salida

Función de Pertenencia	Punto	Observación
Cero	0	Voltaje necesario para marcar 20°C
Uno	0,9	Voltaje necesario para marcar 30°C
Dos	2	Voltaje necesario para marcar 40°C
Tres	2,4	Voltaje necesario para marcar 50°C

Cuatro	2,71	Voltaje necesario para marcar 60°C
Cinco	2,75	Voltaje necesario para marcar 70°C
Tope	5	Voltaje para alcanzar la temperatura deseada con rapidez

Una vez determinadas las funciones de pertenencia para las entradas y salida, se designa la salida deseada según las entradas existentes en la matriz de la regla base (figura. 5.6.). Por ejemplo:

- Si el sensor es veinte y el set_point es sveinte, entonces la salida es cero.
- Si el sensor es treinta y el set_point es streinta, entonces la salida es uno.
- Si el sensor es cuarenta y el set_point es scuarenta, entonces la salida es dos.

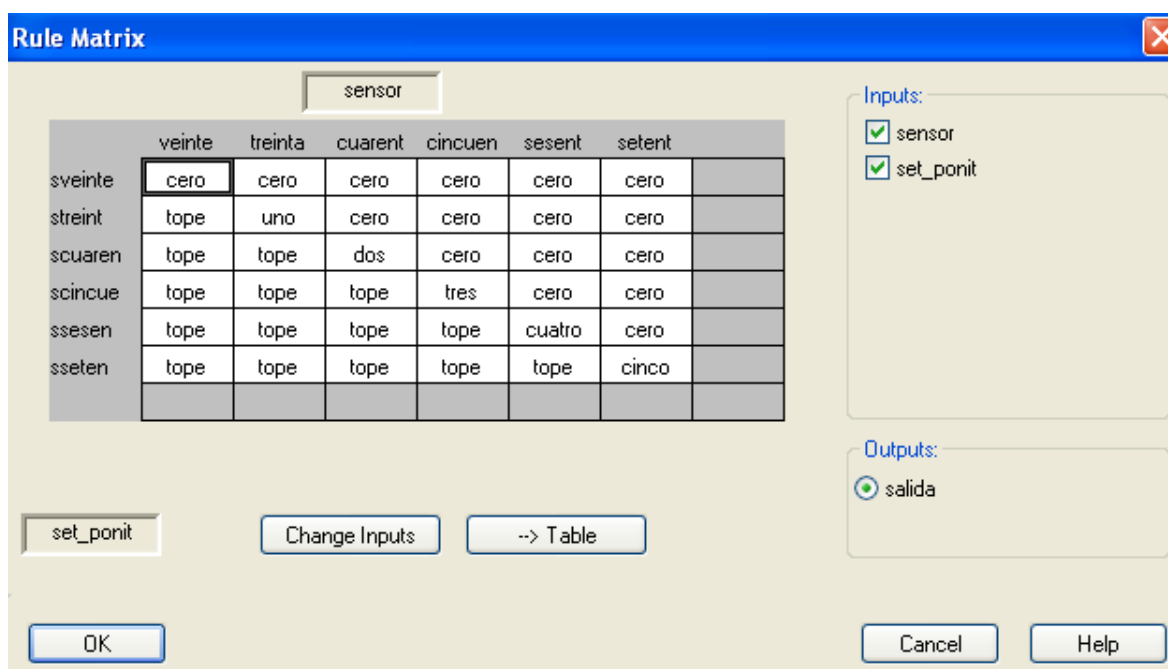


Figura. 5.6. Definición del bloque IF-THEN (regla base)

El archivo del programa se guarda con extensión *.fpl, y el target system seleccionado debe ser WinCC (para la versión 5 de FuzzyControl++).

5.2.2 Código del Programa Step7 Professional

La figura. 5.7. presenta el ingreso y escalamiento de los valores analógicos.

```

L    PEW  272          //Carga de valor análogo al acumulador1
// entre 0 13824 ó 0v a 5v (16 bits).

ITD          //Conversión de entero de 16 bits a 32 bits

DTR          // Conversión de entero de 32 bits a valor real

L    1.382400e+004    //Carga del acumulador1 con 13824 para el escalamiento,
// lo que había en el acumulador1 pasa al acumulador2

/R          //División con el número real 13824, se guarda en el acumulador1

L    5.000000e+001    //Carga 50 para la conversión en el acumulador1,
//lo que había en el acumulador1 pasa al acumulador2

*R          //Multiplicación con el número real 50, lo guarda en el acumulador1

L    2.000000e+001    //Carga 20 al acumulador1,
// lo que había en el acumulador1 pasa al acumulador2

+R          //Suma con el número real 20

T    MD    10          //Transferencia del valor escalado en formato real
    
```

Figura. 5.7. Ingreso y escalamiento de valores analógicos

La figura. 5.8. presenta el escalamiento y salida de los valores analógicos.

```

L    AD    20          //Carga valor esaclado de 0 a 5 en formato real (acumulador1)

L    5.000000e+000    //Carga 5 en el acumulador1,
// lo que había en el acumulador1 pasa al acumulador2

/R          //División con el valor real 5

L    1.382400e+004    //Carga 13824

*R          //Multiplicación con el número real 13824

RND          // Redondeo a entero

T    PAW  272          //Transferencia a la salida,
//el valor entero de 0 a 13824(16 bits)es el valor analógico de la salida.
    
```

Figura. 5.8. Escalamiento y salida de valores analógicos

En la figura. 5.9. se puede ver la habilitación para el *trigger* del bloque de control difuso, y en la figura. 5.10. los *timers* creados para la generación de pulsos. La programación de estos *timers* es de gran importancia, ya que sin ellos no se podrá obtener los valores de salida de la tabla FuzzyControl++ insertada dentro de WinCC.



Figura. 5.9. Habilidad para el *trigger* del bloque de control difuso

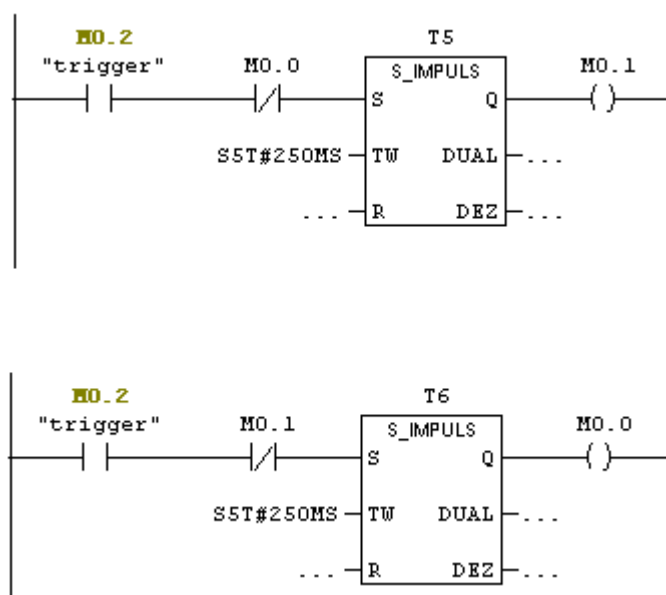


Figura. 5.10. Timers para generación de pulsos

5.2.3 Direccionamiento del HMI realizado en WinCC

La tabla subsiguiente (tabla. 5.2.) muestra las variables usadas para direccionar los objetos insertados en el diseñador gráfico.

Tabla. 5.3. Direccionamiento de objetos en WinCC

Objeto	Pestaña (Propiedades del objeto)	Opción	Propiedades (Opción)	Conexión
Botón	Eventos	Misceláneo	Cambio de objeto	Directa(trigger_hmi, M1.0)
Campo de E/S 1	Propiedades	Entrada/salida	Valor de salida	Variable externa (sensor1, MD10)
Campo de E/S 2	Propiedades	Entrada/salida	Valor de salida	Variable interna (set_pointcopia)
			Tipo de campo	Entrada
	Eventos	Entrada/Salida(Valor de entrada)	Cambio	Directa (FuzzyEn2)
Campo de E/S 3	Propiedades	Entrada/salida	Valor de salida	Variable externa (salida1, AD20)

Campo de E/S 4	Propiedades	Entrada/salida	Valor de salida	Variable externa (trig1, M0.0)
Barra1	Propiedades	Misceláneo	Conexión de control de proceso	Variable externa (sensor1, MD10)
Barra2	Propiedades	Misceláneo	Conexión de control de proceso	Variable interna (set_pointcopia)
Barra3	Propiedades	Misceláneo	Conexión de control de proceso	Variable externa (salida1, AD20)
Control FuzzyControl	Propiedades	Propiedades de control	FuzzyEn1	Variable externa (sensor1, MD10)
	Eventos	Eventos del objeto	Configuración del disparador (trigger) Cambio de FuzzySal1	Variable externa (trig1, M0.0) Directa (salida1, AD20)

Para las gráficas de tendencia fue utilizado el Tag Logging, en él un archivo de tipo valor de proceso es creado, este permitirá relacionar las variables con las curvas (ver figura. 5.11.). La programación de esto se facilita con el uso del asistente Archive Wizard propio del programa.

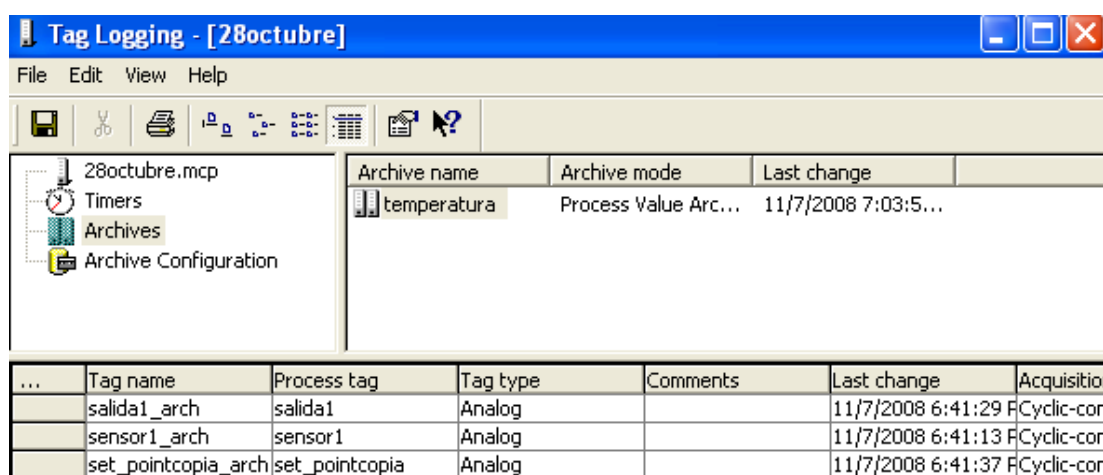


Figura. 5.11. Archivo creado en Tag Logging

Para habilitar la gráfica de tendencia en el diseñador gráfico, se inserta un objeto llamado control de tendencia *online* de WinCC, y se fija las propiedades de las variables seleccionadas (ver figura. 5.12.).

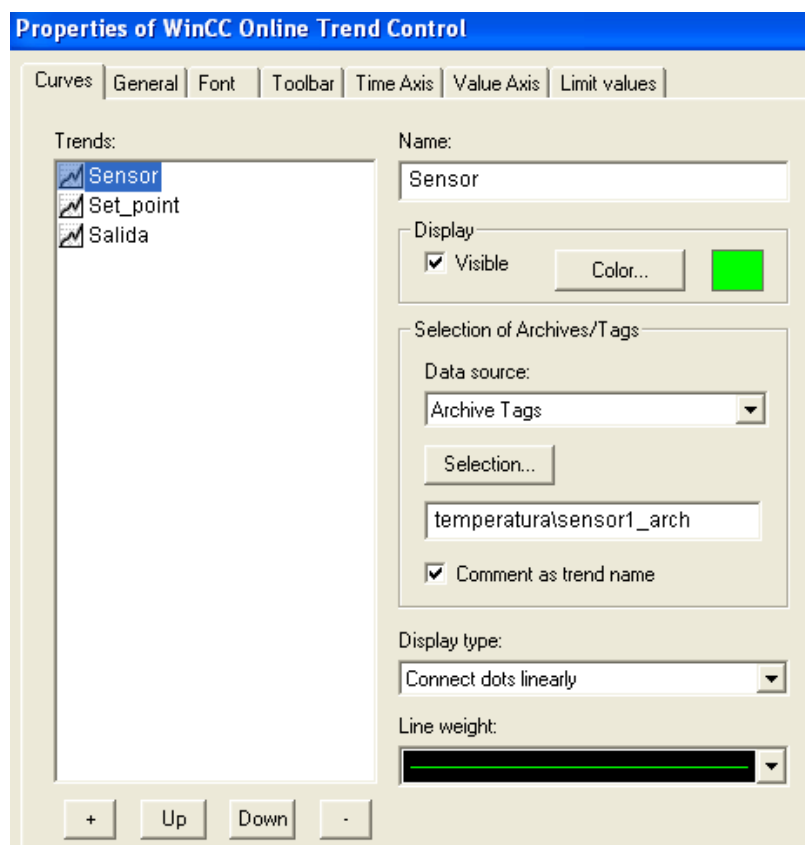


Figura. 5.12. Ventana de Propiedades del control de tendencia online de WinCC

5.3 SIMULACIÓN

WinCC cuenta con el simulador de variables internas o *tags* internos, este permite ingresar valores de acuerdo a lo requerido. La figura. 5.13. muestra el simulador de Tags.

Con el simulador de variables internas se aparentó la entrada del sensor (ver figura. 5.14.):

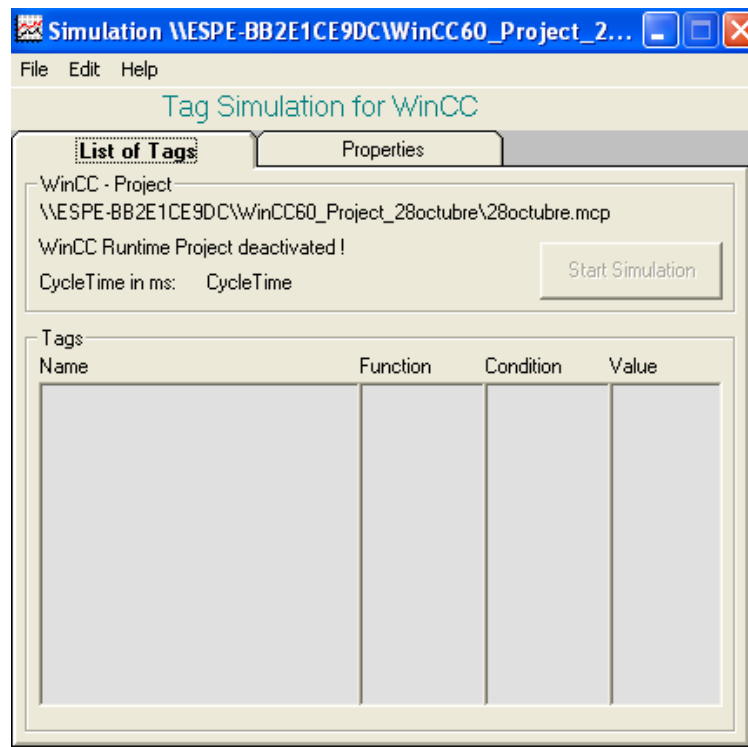


Figura. 5.13. Simulador de Tags para WinCC

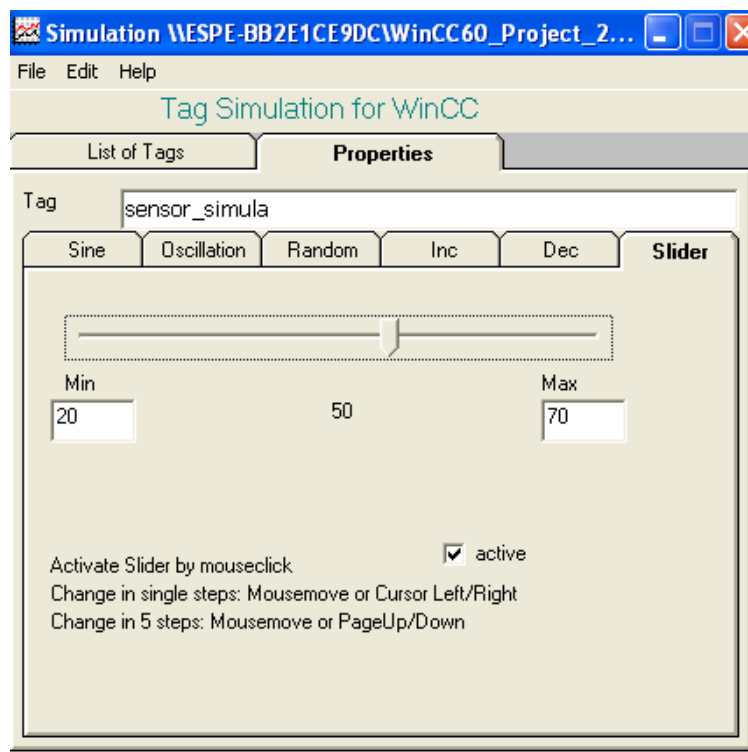


Figura. 5.14. Simulación del sensor en el Tag Simulation

En la figura. 5.15. se puede ver como actúa el sensor simulado en la pantalla del WinCC en modo *runtime*.

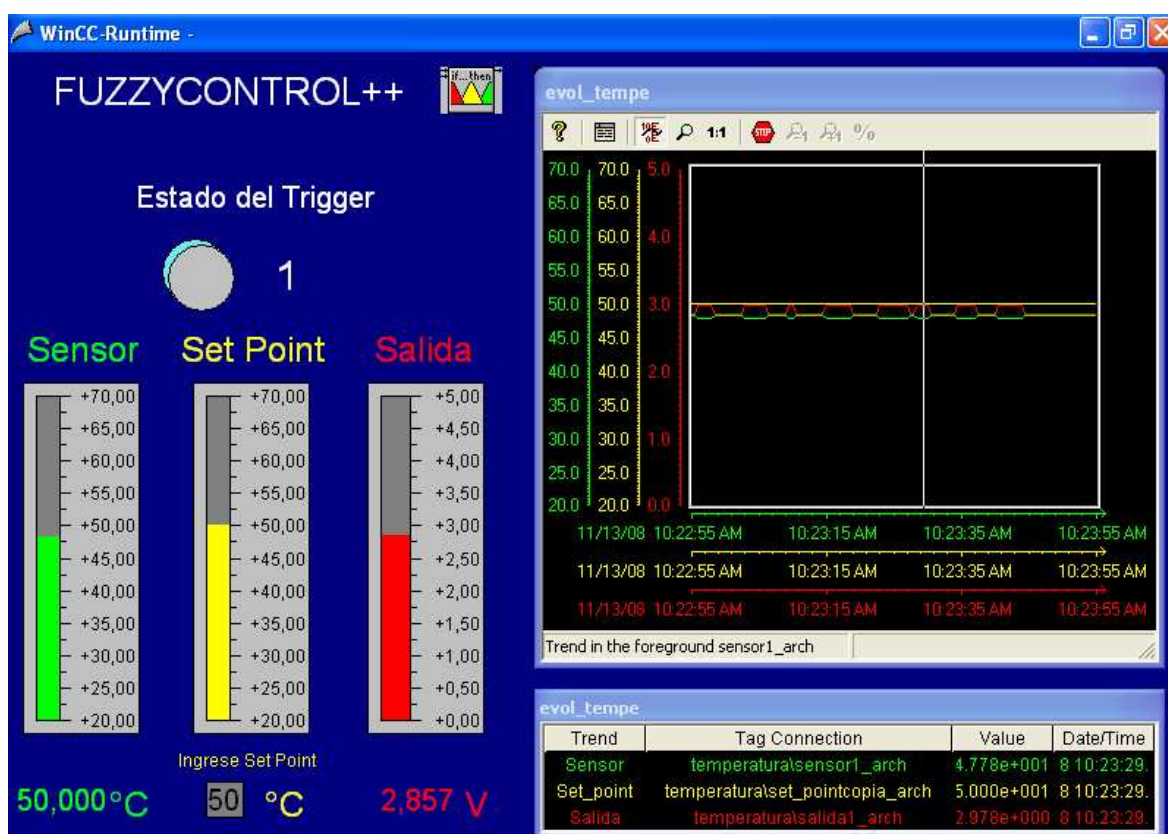


Figura. 5.15. Visualización de la simulación en WinCC Runtime

Al desplazar el *slider* en el simulador de *tags* se visualiza el valor que toma la salida, el cual en la práctica sería aplicado para calentar al cilindro.

La tabla siguiente (tabla. 5.3.) muestra algunos valores que se simularon para el sensor:

Tabla. 5.4. Valores del sensor simulados con el Tag Simulation

Sensor	Set Point	Salida
20	30	1,963
40	50	2,978
50	50	2,857

FuzzyControl++ cuenta con una herramienta llamada *Plotter de Curva* (ver figura. 5.16.), en la cual se puede apreciar el comportamiento de las variables de entrada y salida.

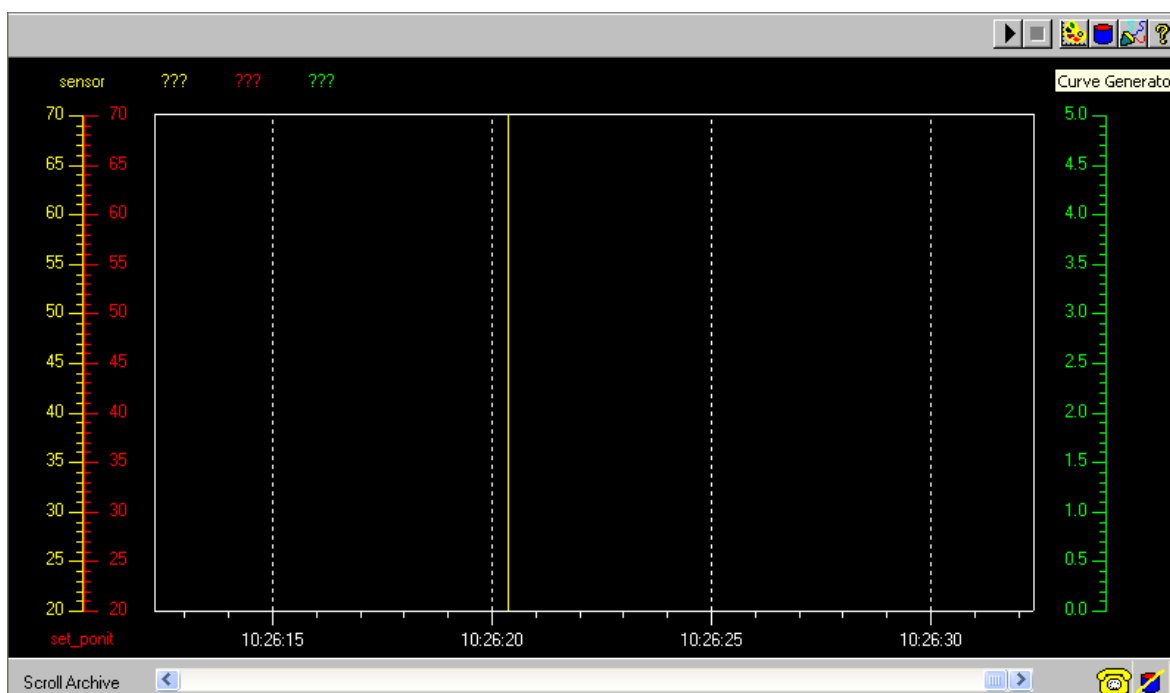


Figura. 5.16. Plotter de Curva

Este a su vez posee un Generador de Curva, el cual como su nombre lo indica genera señales para las entradas (ver figura. 5.17.).

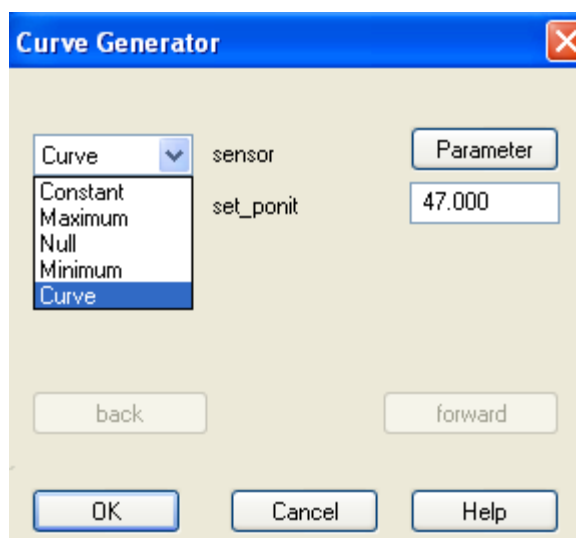


Figura. 5.17. Generador de Curva

Los gráficos consecutivos del *plotter* muestran el comportamiento de una determinada curva para el sensor, al igual que del *set point* y su salida

correspondiente. En las figuras sucesivas se puede ver los tipos de curvas seleccionadas para las variables de entrada (figura. 5.18.), y los parámetros elegidos para la curva del sensor (figura. 5.19.).

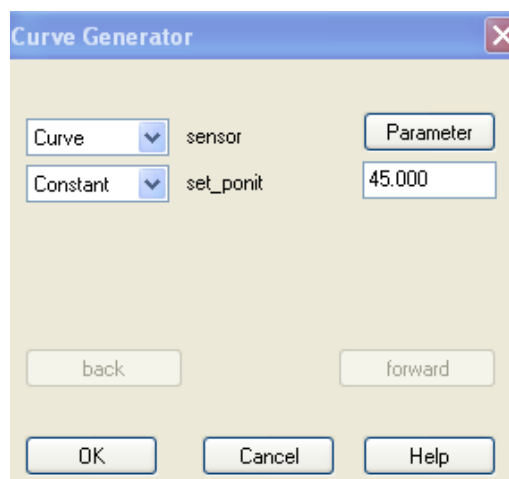


Figura. 5.18. Simulación con el Generador de Curva

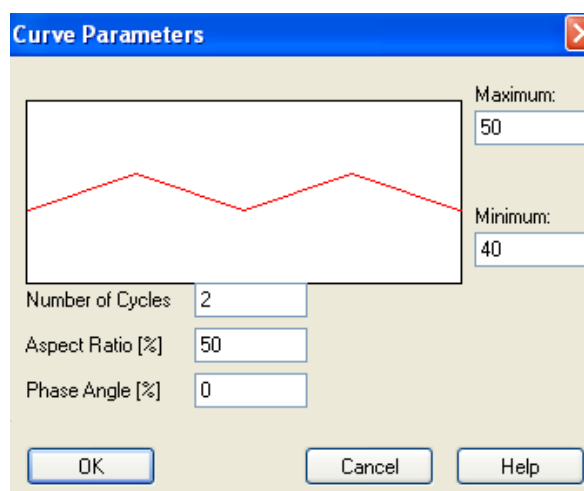


Figura. 5.19. Parámetros de la Curva para la simulación del sensor

La figura. 5.20. muestra el comportamiento de las curva dentro del Plotter de Curva.

Igualmente gracias a otra de las herramientas de FuzzyControl++ llamada Actividad de la Regla, se presenta a modo de barras el porcentaje en el cual la regla actúa, y el valor real de la salida (flecha amarilla). Ver figura. 5.21.

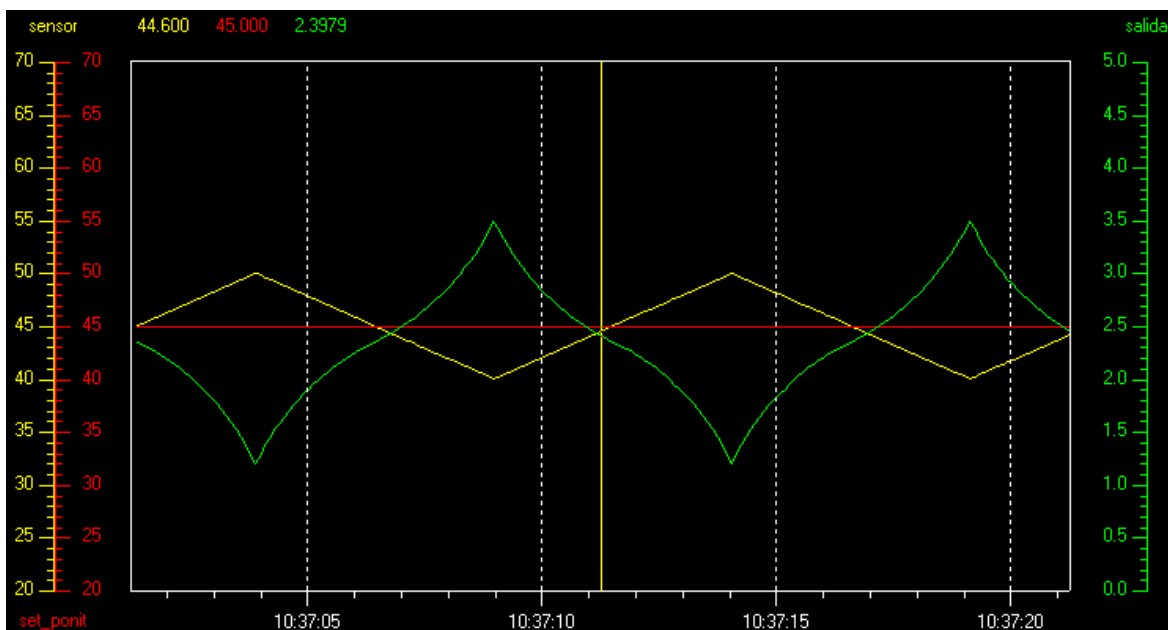


Figura. 5.20. Plotter de Curva con valores simulados

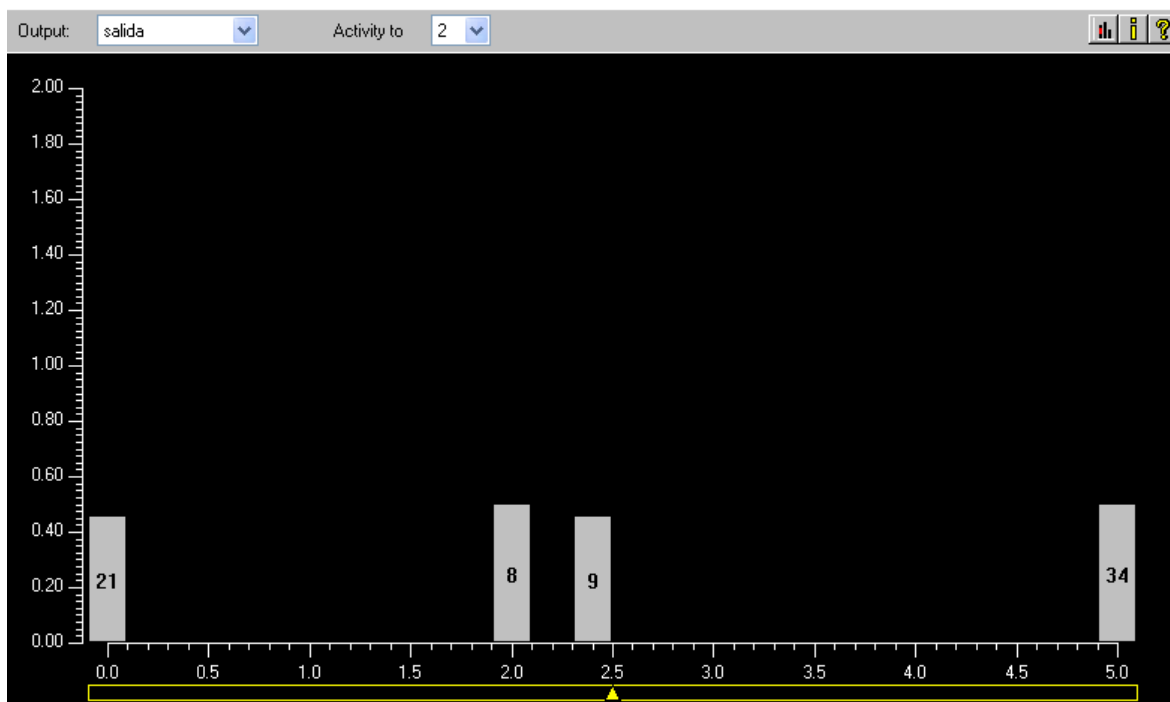


Figura. 5.21. Actividad de la Regla con valores simulados

De este modo se puede ir generando distintos valores para las variables de entrada, y se puede comprobar el correcto comportamiento de la salida.

5.4 IMPLEMENTACIÓN

5.4.1 Dispositivos utilizados en el Sistema

Para la implementación del sistema se requiere:

- PLC Siemens S7-300 (Módulo de alimentación PS 307 5A y CPU 315F-2 PN/DP)
- Tarjeta de memoria Micro 512kb
- Módulo de entradas y salidas análogas SM334 AI 4/AO 2 8BIT.
- Módulo Degem (AIR FLOW TEMPERATURE CONTROL SYSTEM PCT-2).
- Sensor IC
- PC Pentium 4 que posea software de FuzzyControl++, Step7 Professional y WinCC (con licencias), y tarjeta de red.
- Cable para conexión Ethernet.
- Cable de timbre.

PLC Siemens S7-300

La figura. 5.22. muestra el PLC S7-300.



Figura. 5.22. PLC S7-300

Su descripción completa se encuentra en el capítulo 3.

Tarjeta de memoria Micro 512kb

La figura. 5.22. presenta la tarjeta de memoria Micro.



Figura. 5.23. MMC 512KB

Su descripción completa se encuentra en el capítulo 3.

Módulo de entradas y salidas análogas SM334 AI 4/AO 2 8BIT

El módulo análogo de entrada/salida SM334: AI 4/AO 2 x 8/8 bits (ver figura. 5.24.) tiene las siguientes características:

- Cuatro entradas, dos salidas.
- Resolución 8 bits.
- No parametrizable, ajuste del tipo de medición y de salida mediante cableado.
- Rango de medición de 0 a 10 V o 0 a 20 mA.
- Rango de salida de 0 a 10 V o 0 a 20 mA.
- Tanto el voltaje y la corriente son opciones de salida.
- Sin separación galvánica respecto a la conexión del bus posterior.
- Con separación galvánica respecto a la tensión de carga.

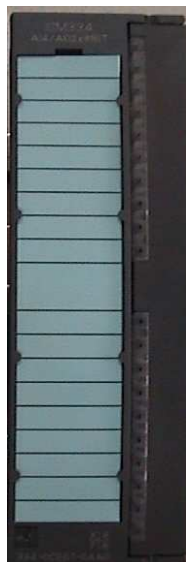


Figura. 5.24. SM334 AI 4/AO 2 8BIT

Módulo Degem

(Air Flow Temperature Control System PCT-2)

Estos módulos simulan la planta en la cual se realizará el control difuso de temperatura. En la figura. 5.25. se muestra el módulo Degem Temperature.

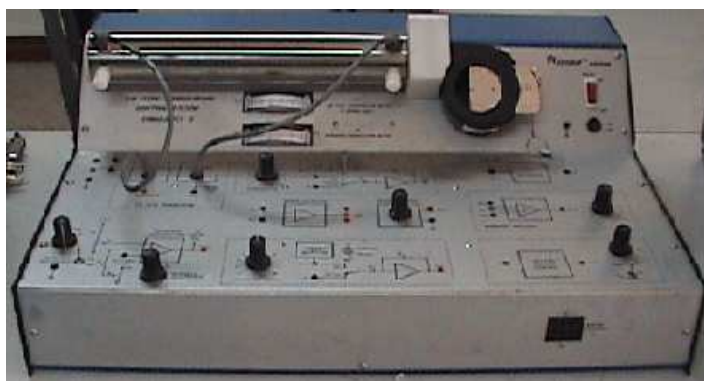


Figura. 5.25. Módulo Degem Temperature

Sensor de circuito integrado IC

“Se fundamenta en la característica de la unión p-n de los semiconductores. Está formado por circuitos integrados sobre un chip, el cual presenta una salida lineal y proporcional a la temperatura. Se consiguen sensores IC que presentan salidas en voltaje analógico y en forma digital. Por estar hechos a base de silicio,

su rango de temperatura está limitado aproximadamente a los 150 °C, son los más lineales, tienen un alto rendimiento y son muy económicos.

Una de las principales ventajas de los sensores IC es su fácil interface. Entre las desventajas se tienen: el limitado rango de temperatura, la necesidad de alimentación y el auto calentamiento¹.

La figura. 5.26. indica el sensor IC integrado en el módulo Degem.

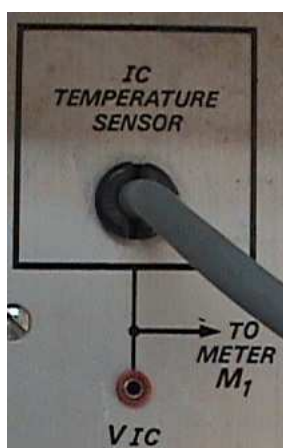


Figura. 5.26. Sensor IC integrado en el Módulo Degem

5.4.2 Modo de uso y diagramas de conexión

Un óptimo funcionamiento del sistema controlador de temperatura se lo obtiene de la siguiente manera:

- ✓ Alimentar tanto al computador, al módulo de simulación del proceso, y al módulo de alimentación PS 307 con 110VAC (ver figura. 5.27.).
- ✓ Conectar el módulo de alimentación a la CPU, y de igual manera al módulo análogo.
- ✓ Enchufar el cable Ethernet tanto al CPU como a la tarjeta de red del computador.
- ✓ Comprobar si los voltajes de entrada en los dispositivos son los correctos. Si estos voltajes llegan a sobrepasarse, los equipos sufrirán averías.

¹ Capítulo VII Sensores, <http://www.unet.edu.ve/~ielectro/sensores.pdf>, página 111, 2008-07-20.

- ✓ Realizar correctamente las conexiones entre el PLC y los demás componentes del sistema como se muestra a continuación (ver figura. 5.28.).

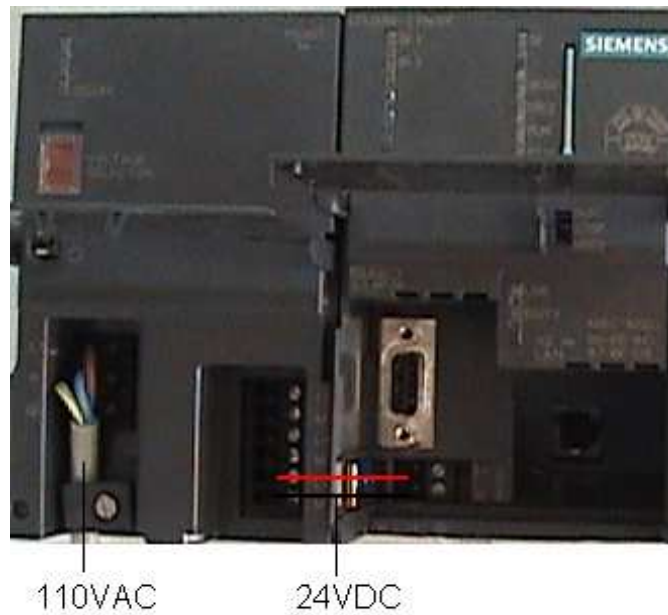


Figura. 5.27. Alimentación del controlador

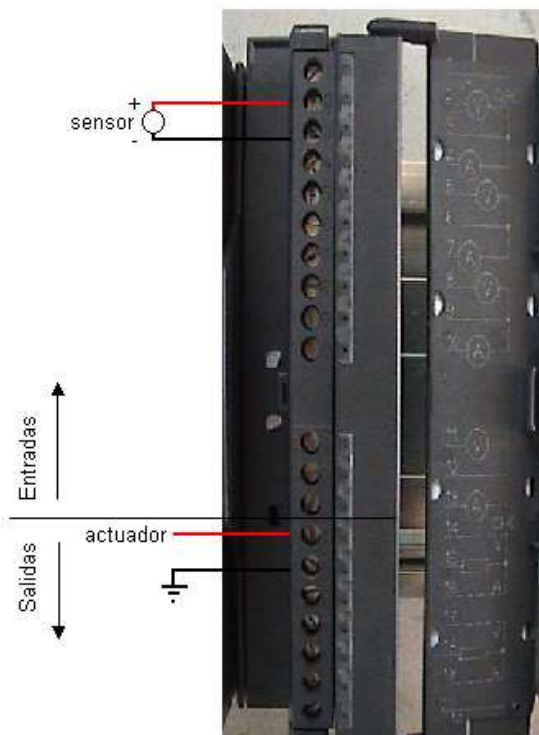


Figura. 5.28. Conexiones de la entrada y salida análogas del Módulo SM334 AI 4/AO 2 8BIT

CAPÍTULO 6

PRUEBAS Y RESULTADOS

En este capítulo se detalla los datos reales obtenidos de las pruebas realizadas con el programa, estos son analizados y comparados entre si.

6.1 PRUEBAS

Para probar el funcionamiento del sistema se ingresaron valores de *set point* (temperaturas deseadas), se observó el tiempo de respuesta, el error en estado estable y el porcentaje de precisión (error porcentual) que presentó el controlador durante estas pruebas. La figura. 6.1. muestra la interfaz HMI creada para la aplicación. Aquí se puede visualizar el valor del sensor, el *set point* y el valor de la salida, tanto en forma numérica como gráficamente, en barras y curvas.

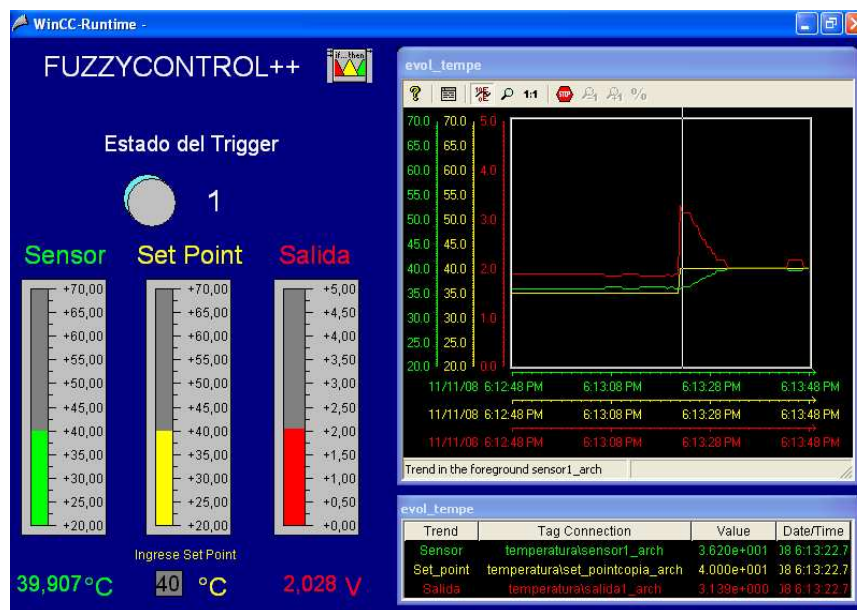


Figura. 6.1. HMI de la aplicación

Cabe destacar que el botón del *trigger* es el encargado del funcionamiento o el no funcionamiento del controlador difuso, su cambio de estado de cero a uno, habilita al sistema (ver figuras. 6.2. y 6.3.).



Figura. 6.2. Trigger apagado



Figura. 6.3. Trigger encendido

El valor de *set point* es ingresado mediante el campo de entrada en la parte inferior de la barra correspondiente (ver figura. 6.4.).



Figura. 6.4. Ingreso de set point

El cuadro de tendencias, es en tiempo real y permite visualizar los valores de las variables (ver figura. 6.5.).

Seguidamente se detallan tres casos de operación del controlador de temperatura:

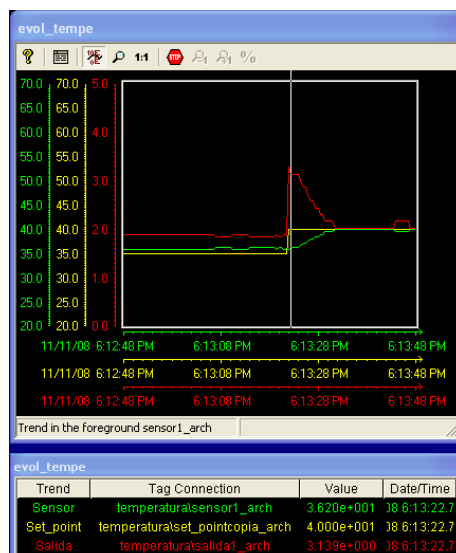


Figura. 6.5. Curvas de tendencia

1. En este caso se inició con una temperatura de 25.556 °C (figura. 6.6.), se ingresó un valor de *set point* de 45 °C, la figura. 6.7. muestra el sobreimpulso el cual en menos de 3 segundos empieza a descender. El tiempo de respuesta fue de aproximadamente 1 minuto con 9 segundos. El error en estado estable fue de 0.926 °C y el porcentaje de precisión fue de ±2.016% (figura. 6.8.).

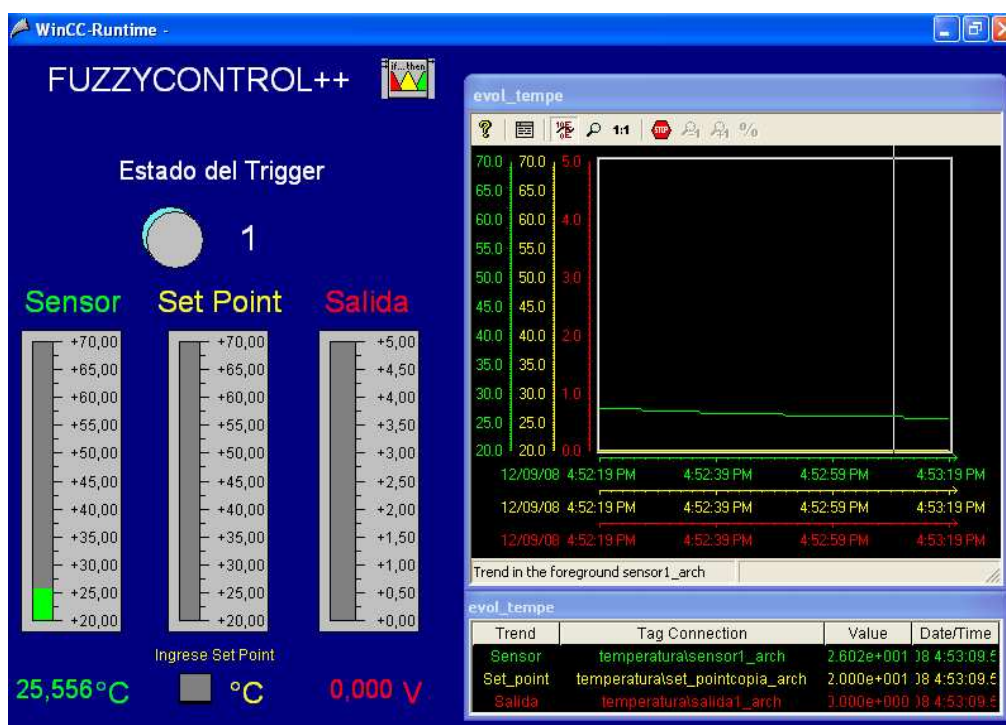


Figura. 6.6. Primer caso: temperatura inicial

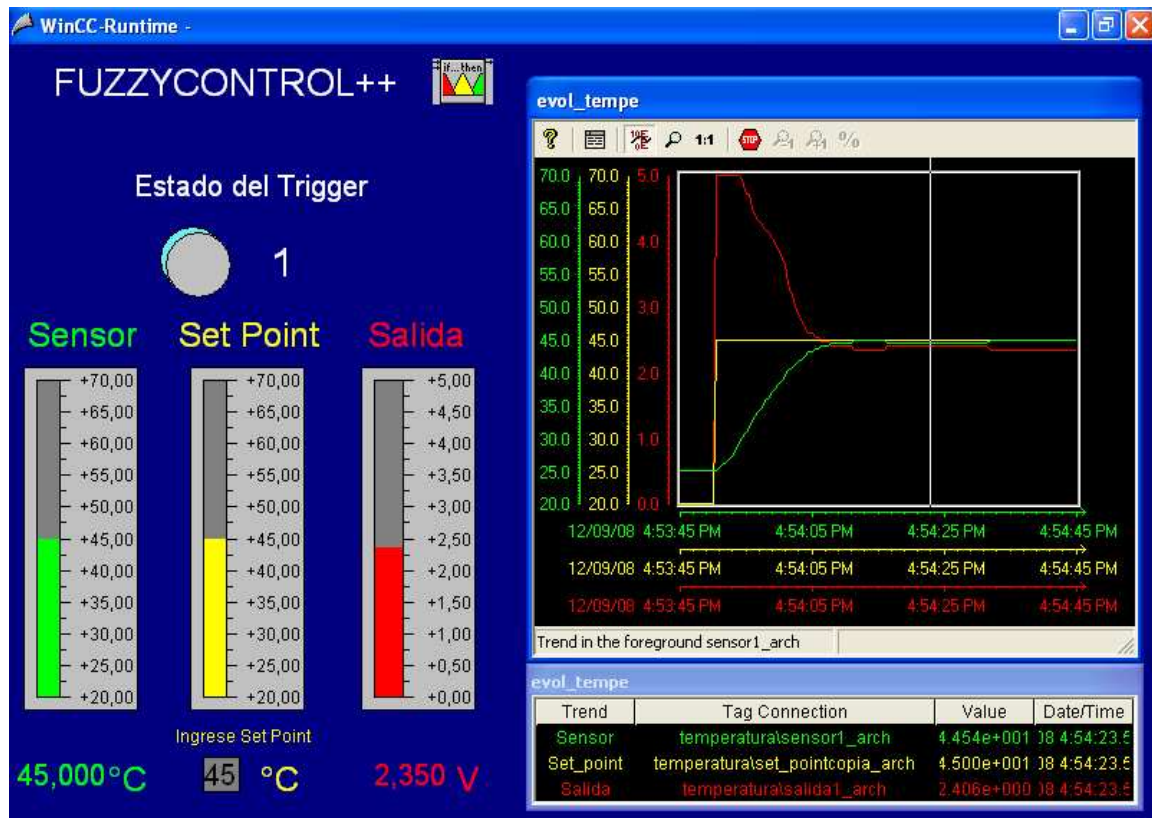


Figura. 6.7. Primer caso: ingreso de set point

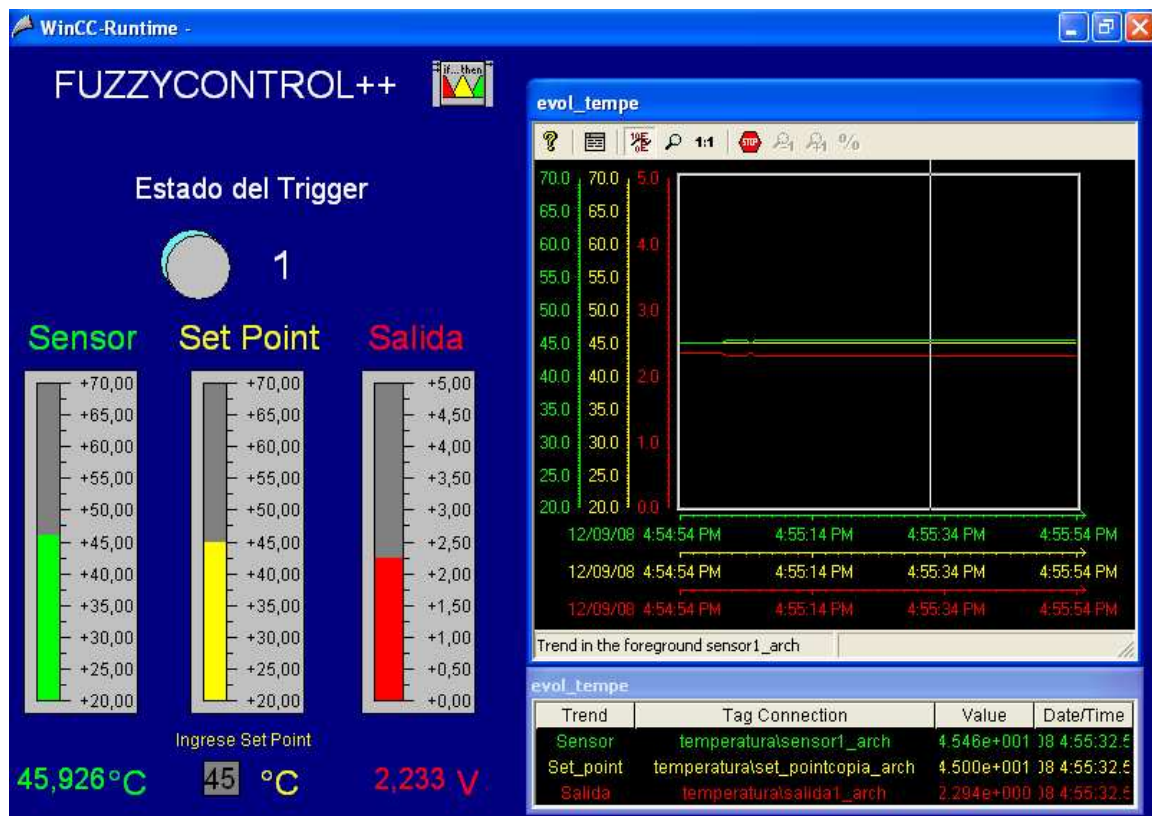


Figura. 6.8. Primer caso: estado estable

- Esta vez el valor inicial fue 40.370 °C (figura. 6.9.). El set point ingresado fue 60 °C, y se produjo un sobreimpulso que inició su descenso después de 10 segundos (figura. 6.10.)

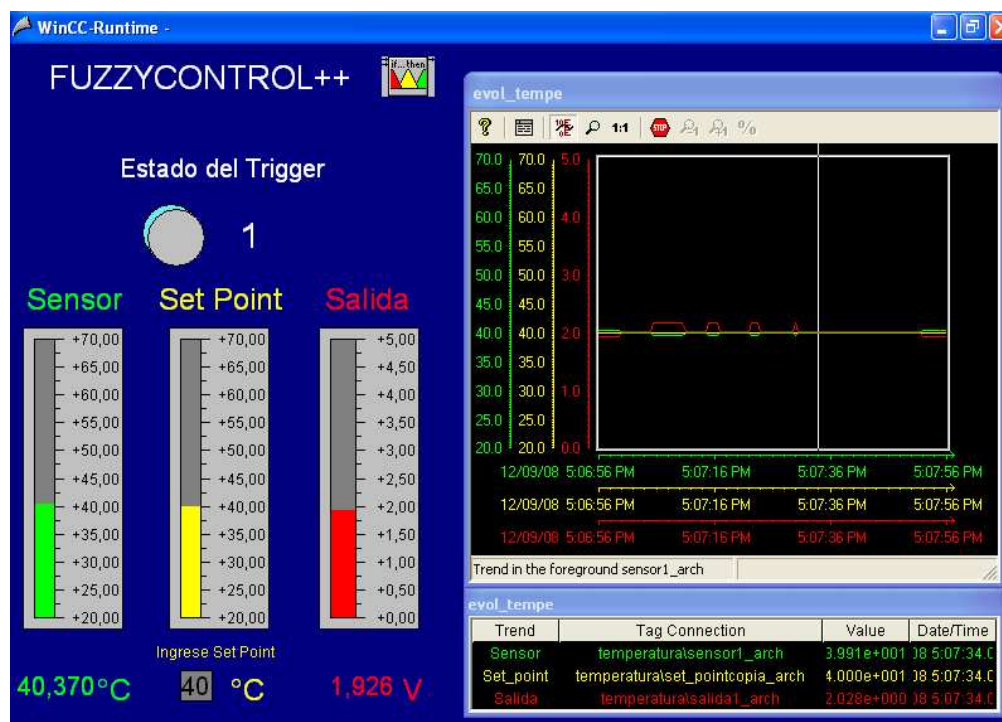


Figura. 6.9. Segundo caso: temperatura inicial

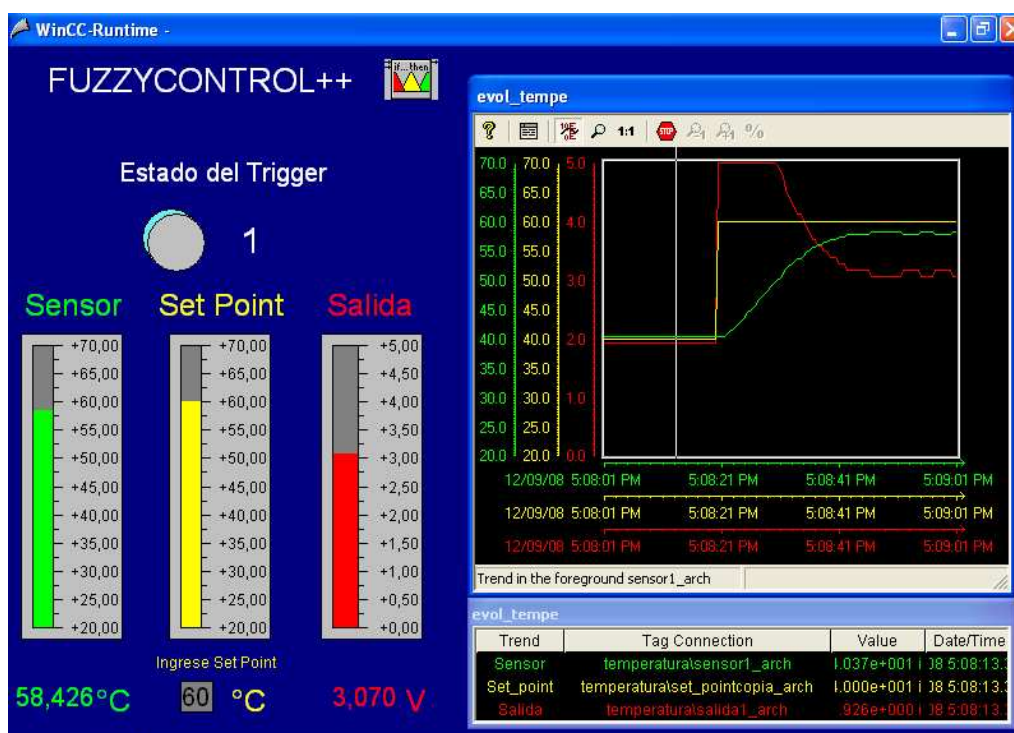


Figura. 6.10. Segundo caso: ingreso de set point

La figura. 6.11. muestra el proceso por el que pasó el controlador hasta alcanzar el estado estable. Finalmente transcurrido un tiempo de respuesta de 3 minutos con 40 segundos, se logró un error en estado estable de 0.185 °C y un error porcentual del $\pm 0.309\%$ (figura. 6.12.).

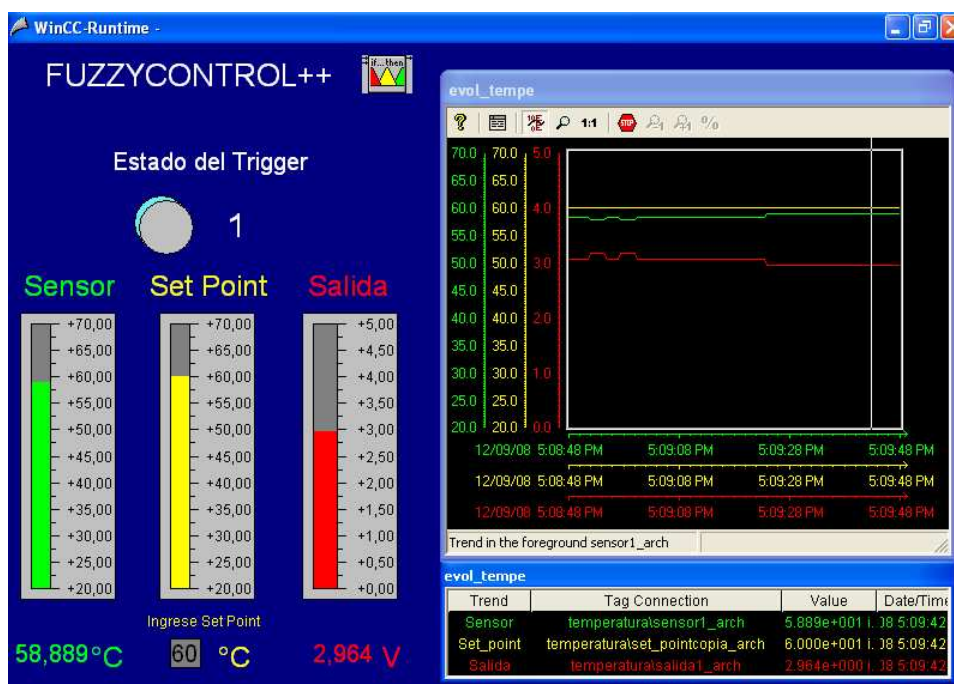


Figura. 6.11. Segundo caso: en proceso de estabilización

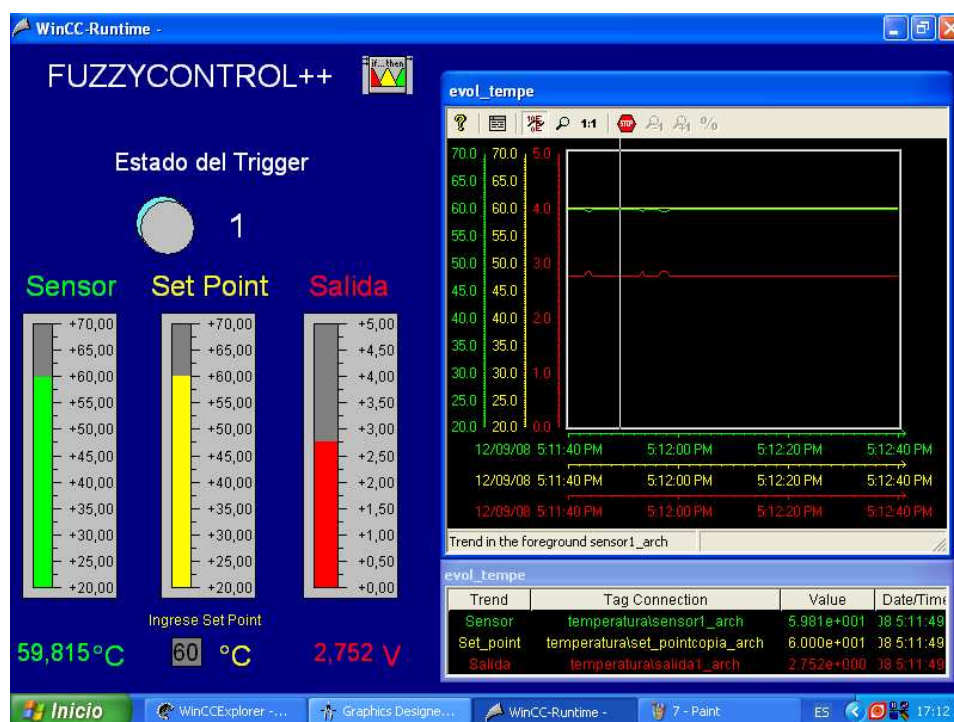


Figura. 6.12. Segundo caso: estado estable

- Por último con una temperatura inaugural de 49.167 °C (figura. 6.13.), se introdujo un *set point* de 42 °C (figura. 6.14.).

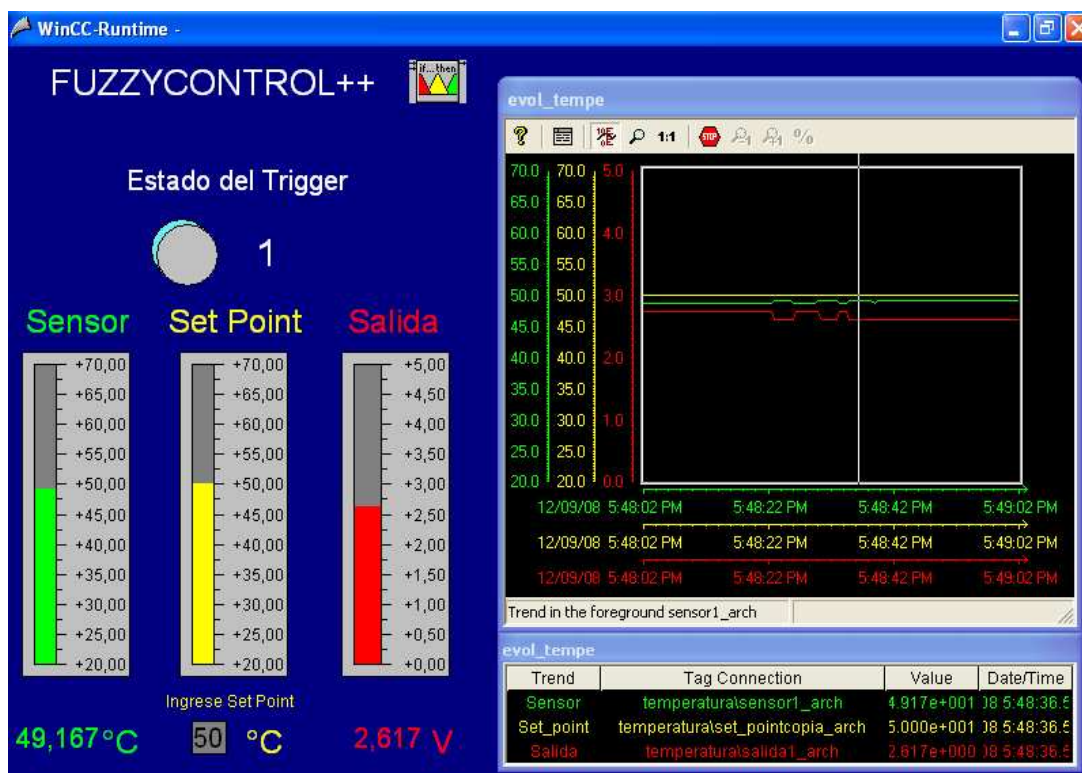


Figura. 6.13. Tercer caso: temperatura inicial

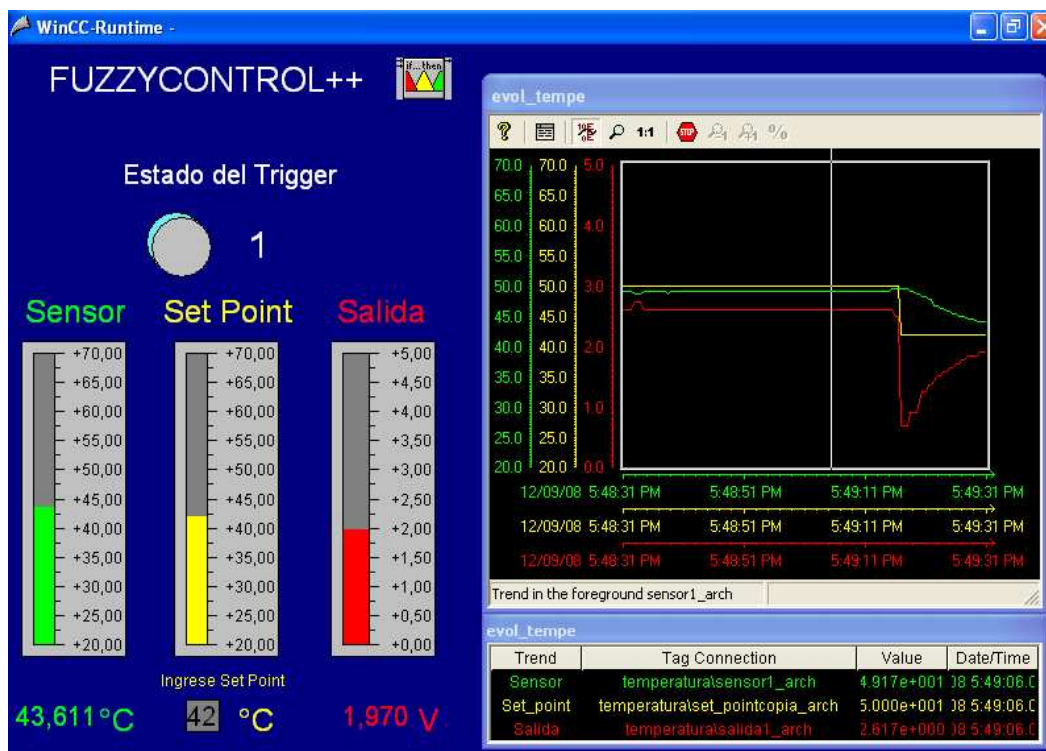


Figura. 6.14. Tercer caso: ingreso de set point

El tiempo de respuesta fue de 30 segundos, el error en estado estable fue de 1.148 °C y el error porcentual fue de ±2.661% (figura. 6.15.).

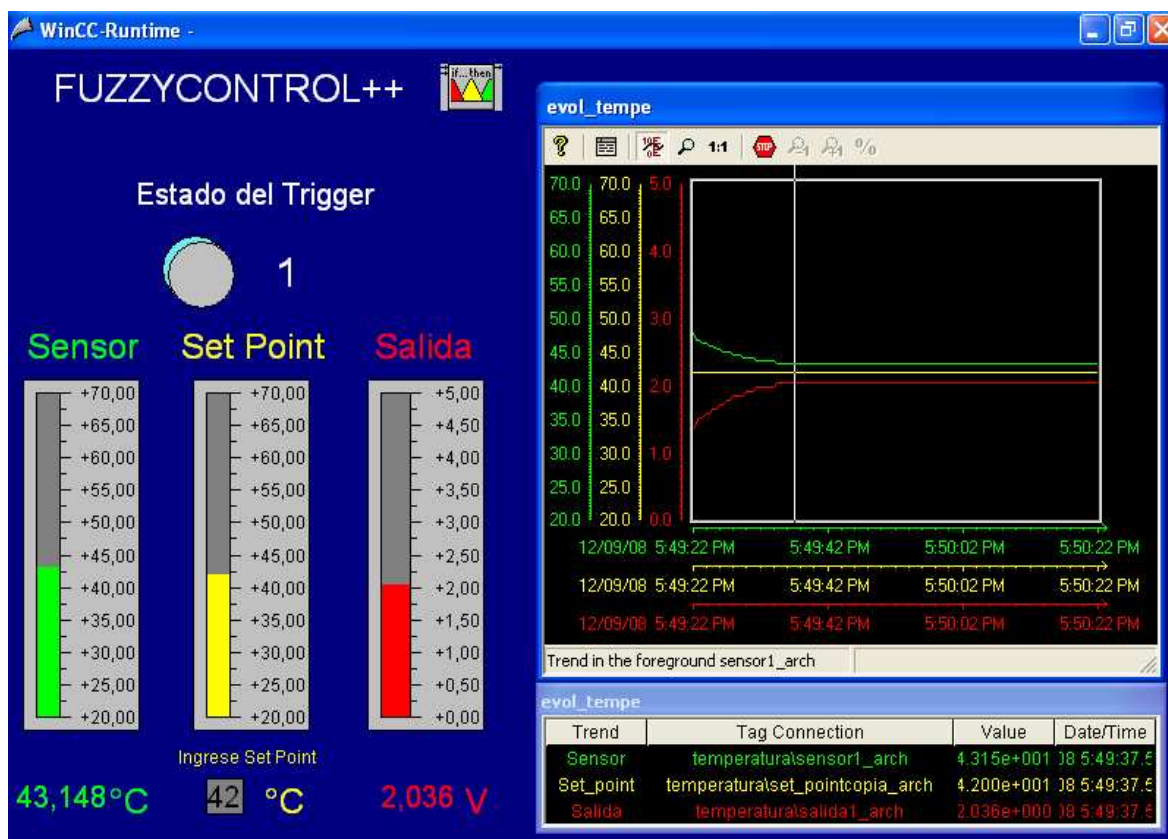


Figura. 6.15. Tercer caso: estado estable

6.2 RESULTADOS

El controlador difuso de temperatura respondió perfectamente a las pruebas, algunas veces el rango de tolerancia fue mucho menor al permitido, y en general se logró lo siguiente:

- Los tiempos de respuesta son moderados para los casos en los que el *set point* es cercano a los valores límites, pero para valores medios los tiempos de respuesta son realmente cortos.

-
- La duración de los sobreimpulsos depende de lo distante que esté la función de pertenencia del sensor respecto a la función de pertenencia del *set point*.
 - El error en estado estable en la mayoría de los casos es muy pequeño, y se encuentra dentro del rango de tolerancia permitido.
 - El error porcentual es muy bueno, pues no sobrepasa el $\pm 3\%$, y por lo general es menor al $\pm 2\%$.

CAPÍTULO 7

CONCLUSIONES Y RECOMENDACIONES

7.1 CONCLUSIONES

Una vez recolectado, analizado y puesto en práctica los conocimientos necesarios sobre control difuso mediante la utilización del software FuzzyControl++, Step7 Professional y WinCC, se logró estructurar un sistema de entrenamiento para FuzzyControl++, cumpliendo a cabalidad con todos los objetivos propuestos.

El manejo y programación de sistemas difusos mediante el software FuzzyControl++ resulta sencillo y comprensible. Si bien no requiere cálculos el diseño de la matriz *IF-THEN* para algunas aplicaciones puede parecer un problema que sin imaginación, sentido común y experiencia se torna complicada su resolución.

El bloque de control FuzzyControl++ utilizado en WinCC V6.2 trabaja con Control ActiveX, dicho objeto posee fallas que perjudican su correcto funcionamiento, y que pueden ser superadas con una solución alterna presentada en este trabajo.

La aplicación realizada produjo resultados satisfactorios, tanto en la simulación como en la implementación, y mostró un ejemplo complejo resuelto mediante el control difuso.

Como resultado de que cada entrada difusa solo permite como máximo 7 funciones miembro, para ciertas aplicaciones con rangos muy amplios y poco lineales esto resulta ser un limitante a la hora de hablar de exactitud. Asimismo debido al calentamiento del sensor se constató la inadaptabilidad que posee el software al momento de existir alteraciones en las variables de entrada.

Este aporte da a conocer de manera rápida, clara y precisa el manejo del software FuzzyControl++, abarca y documenta detalles que no constan en los manuales propios del programa y que solo con investigación propia se los pudo conocer.

7.2 RECOMENDACIONES

Es imprescindible conocer básicamente el funcionamiento de Step7 Professional y WinCC pues ellos permiten la comunicación y utilización del programa FuzzyControl++. Dentro de cada uno de estos existen habilitaciones que deben ser ejecutadas caso contrario será imposible el desarrollo de cualquier aplicación

Para que el bloque de control difuso funcione siempre, se recomienda crear un disparador y conectarlo al *trigger* del bloque de control independientemente de si está seleccionado el modo manual o el automático. El bloque de control debe detectar el flanco ascendente del disparador caso contrario no responderá.

Realizar las instrucciones detenidamente y en el orden indicado.

Inspeccionar las conexiones y el estado de todos los dispositivos del sistema antes de ser energizado.

ANEXOS

A1 GUÍA DE USUARIO

CONTENIDO

1. Instalación del hardware

- 1.1 Dispositivos
- 1.2 Conexiones y alimentación
- 1.3 Condiciones de Temperatura

2. Instalación de software

- 2.1 Step7 Professional
- 2.2 WinCC
- 2.3 FuzzyControl++
 - 2.3.1 Control ActiveX

3. Habilitaciones para la comunicación

- 3.1 Step7 Professional
- 3.2 WinCC
- 3.3 FuzzyControl++
 - 3.3.1 Control ActiveX

4. Descripción del Controlador

5. Puesta en marcha

- 5.1 Programa en WinCC Runtime: Modo de Uso

6. Recomendaciones

Esta guía de usuario corresponde al sistema de entrenamiento en control difuso utilizando ControlFuzzy++.

Por favor lea los capítulos de instalación para un uso seguro y eficiente.

Se espera que este control difuso de temperatura sea de gran utilidad para los requerimientos que usted necesite dentro de su industria.

Soporte es proporcionado por:

andpao79@hotmail.com

1. INSTALACIÓN DEL HARDWARE

1.1 Dispositivos

Para la implementación del sistema se requiere:

- PLC Siemens S7-300 (Módulo de alimentación PS 307 5A y CPU 315F-2 PN/DP)
- Tarjeta de memoria Micro 512kb
- Módulo de entradas y salidas análogas SM334 AI 4/AO 2 8BIT.
- Módulo Degem (AIR FLOW TEMPERATURE CONTROL SYSTEM PCT-2).
- Sensor IC
- PC Pentium 4 o superior, que incluya tarjeta de red.
- Cable para conexión Ethernet.
- Cable de timbre.

1.2 Conexiones y alimentación

Un óptimo funcionamiento del sistema controlador de temperatura se lo obtiene de la siguiente manera:

- ✓ Alimentar tanto al computador, al módulo de simulación del proceso, y al módulo de alimentación PS 307 con 110VAC (ver figura. 1.1.).



Figura. 1.1. Alimentación del PS 307

- ✓ Conectar el módulo de alimentación a la CPU, y de igual manera al módulo análogo (ver figura. 1.2.).

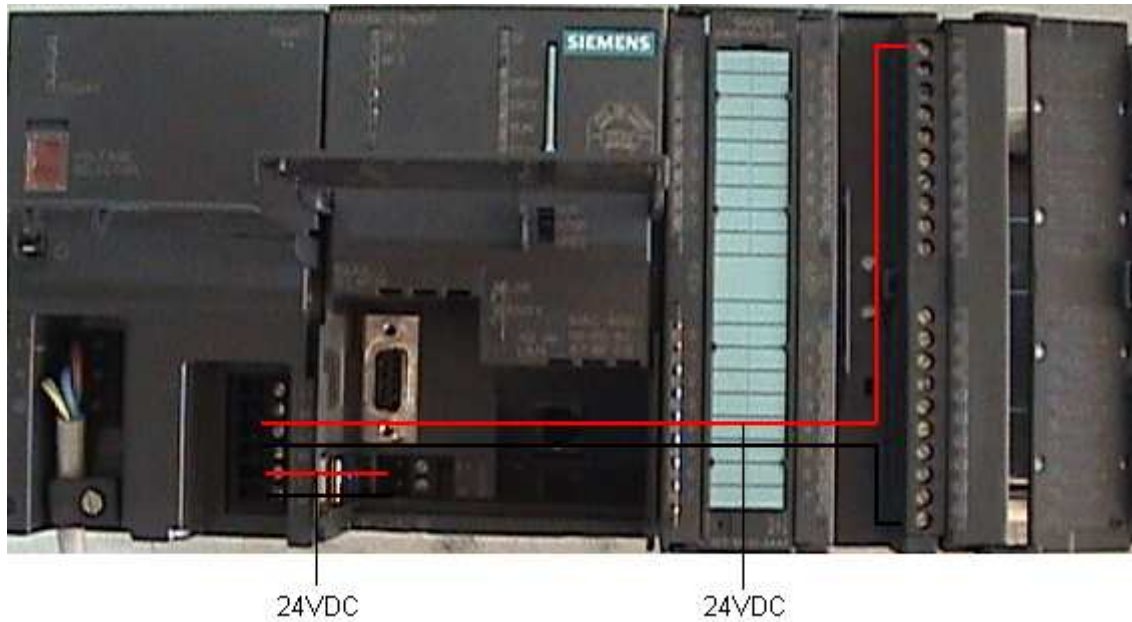


Figura. 1.2. Alimentación de la CPU y del módulo análogo

- ✓ Enchufar el cable Ethernet tanto al CPU como a la tarjeta de red del computador (ver figura. 1.3.).

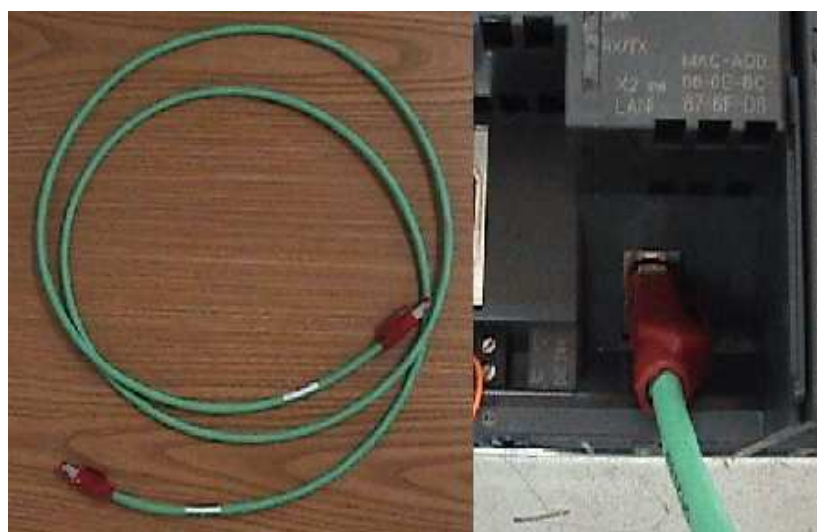


Figura. 1.3. Cable Ethernet y su conexión al CPU

- ✓ Conectar la entrada y la salida análogas que serán utilizadas con su respectivo sensor y actuador en el módulo de temperatura (ver figura. 1.4. y figura. 1.5.).

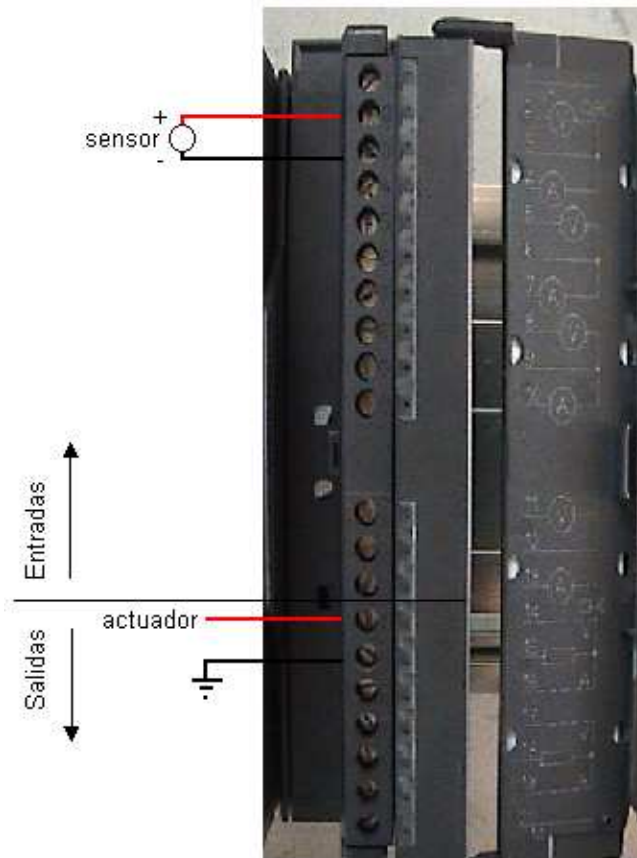


Figura. 1.4. Conexiones de la entrada y salida análogas del módulo SM334 AI 4/AO 2 8BIT.

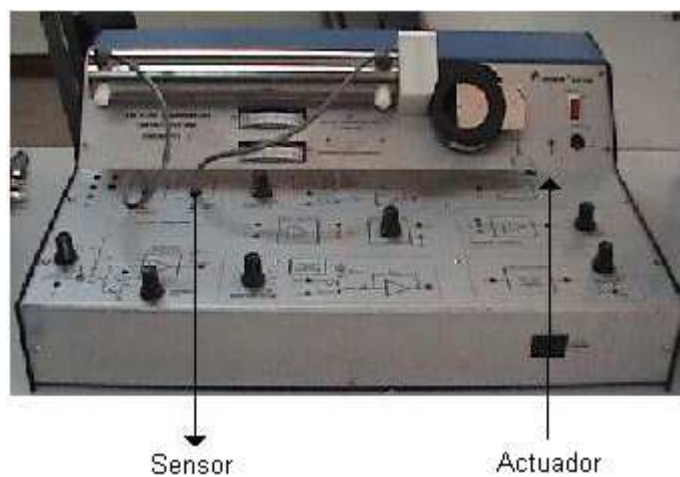


Figura. 1.5. Salida del sensor y entrada del actuador en el módulo Degem

1.3 Condiciones de Temperatura

El sistema controlador difuso de temperatura actúa en el rango de 20 a 70 °C. Con una tolerancia para el set point de $\pm 2^{\circ}\text{C}$.

2. INSTALACIÓN DE SOFTWARE

2.1 Step7 Professional

Instalar STEP7 Professional según el manual propio de este programa, y su respectiva licencia.

2.2 WinCC

Instalar WinCC según el manual propio de este programa, y su respectiva licencia. Es necesario instalar previamente WinCC SQL y el componente de Windows Message Queue Server para poder instalar WinCC. Verificar si la versión de Windows y el Service Pack utilizados son compatibles con WinCC.

2.3 FuzzyControl++

Instalar FuzzyControl++ según el manual propio de este programa, y su respectiva licencia.

2.3.1 Control ActiveX

Instalar el Control ActiveX para ControlFuzzy++ según el manual propio de este programa.

3. HABILITACIONES PARA LA COMUNICACIÓN

3.1 Step7 Professional

Es importante que se realicen algunos ajustes en el computador que será utilizado.

Diríjase a Inicio, panel de control, conexiones de red, y seleccione la conexión que está siendo utilizada, déle un clic derecho y seleccione Propiedades. En la pestaña General seleccione Protocolo Internet (TCP/IP), y de un clic sobre el botón Propiedades (ver figura. 3.1.).

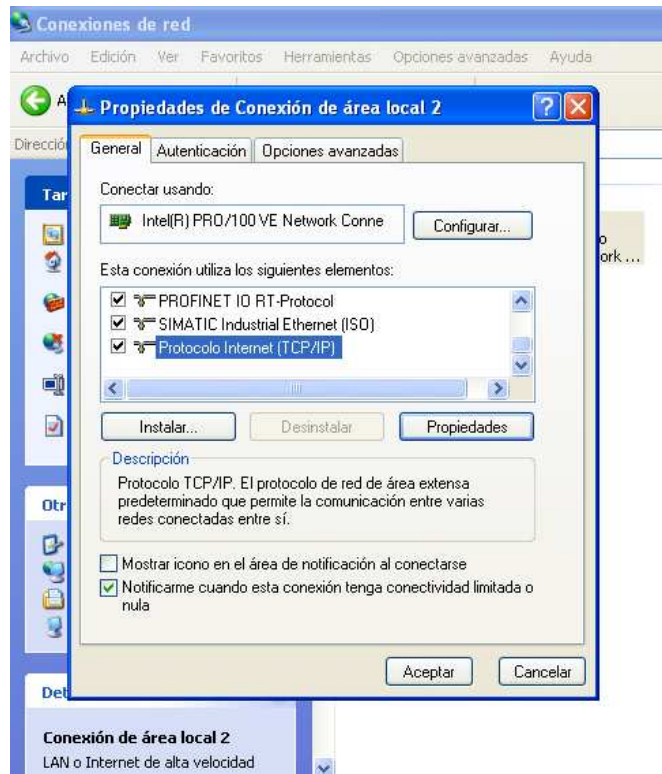


Figura. 3.1. Propiedades de Conexión

En la pantalla Propiedades de Protocolo Internet (TCP/IP), establezca la configuración mostrada en la figura. 3.2.:

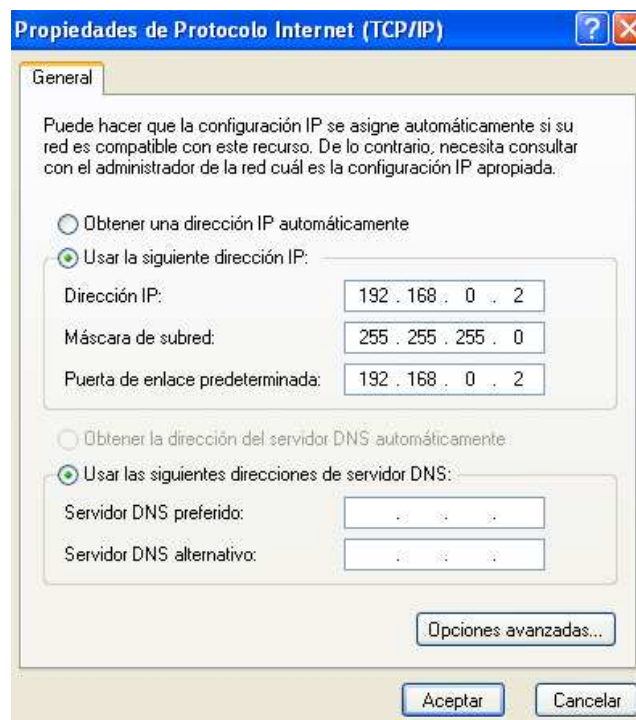


Figura. 3.2. Propiedades de Protocolo Internet (TCP/IP)

Y acepte en todas las ventanas.

Ahora diríjase al Panel de Control e ingrese a Set PG/PC Interface, seleccione el punto de acceso que usted usará (tarjeta de red), así como se señala en la figura. 3.3., y acepte.

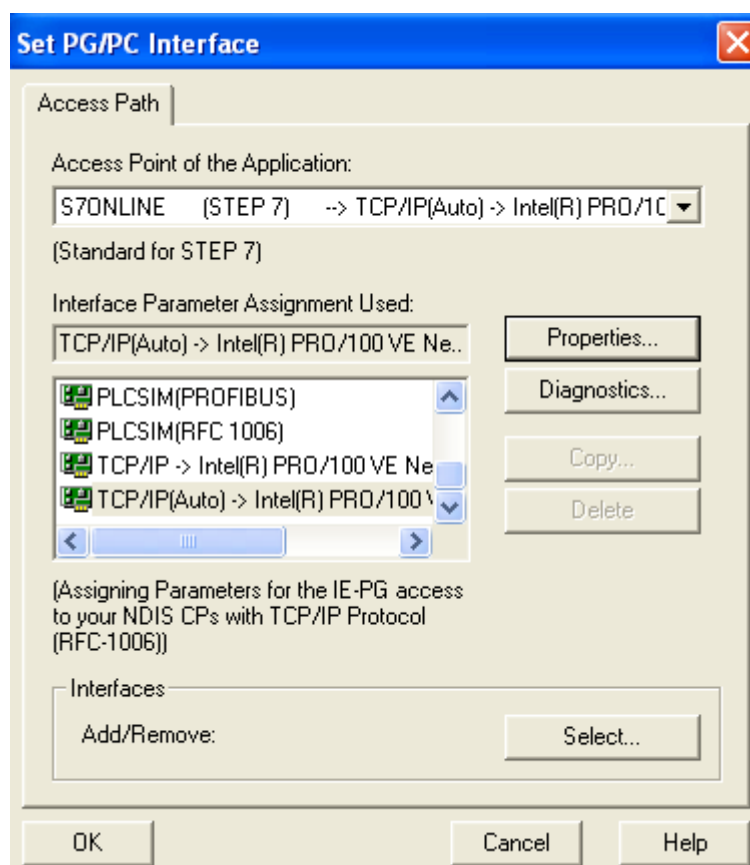


Figura. 3.3. Set PG/PC Interface

3.2 WinCC

Para que WinCC permita realizar todas las acciones de esta aplicación, vaya al explorador de WinCC, seleccione Computer y de un clic derecho, escoja Properties.

En la pestaña Startup de la ventana Computer properties habilite las opciones como se visualiza en la figura. 3.4. Acepte.

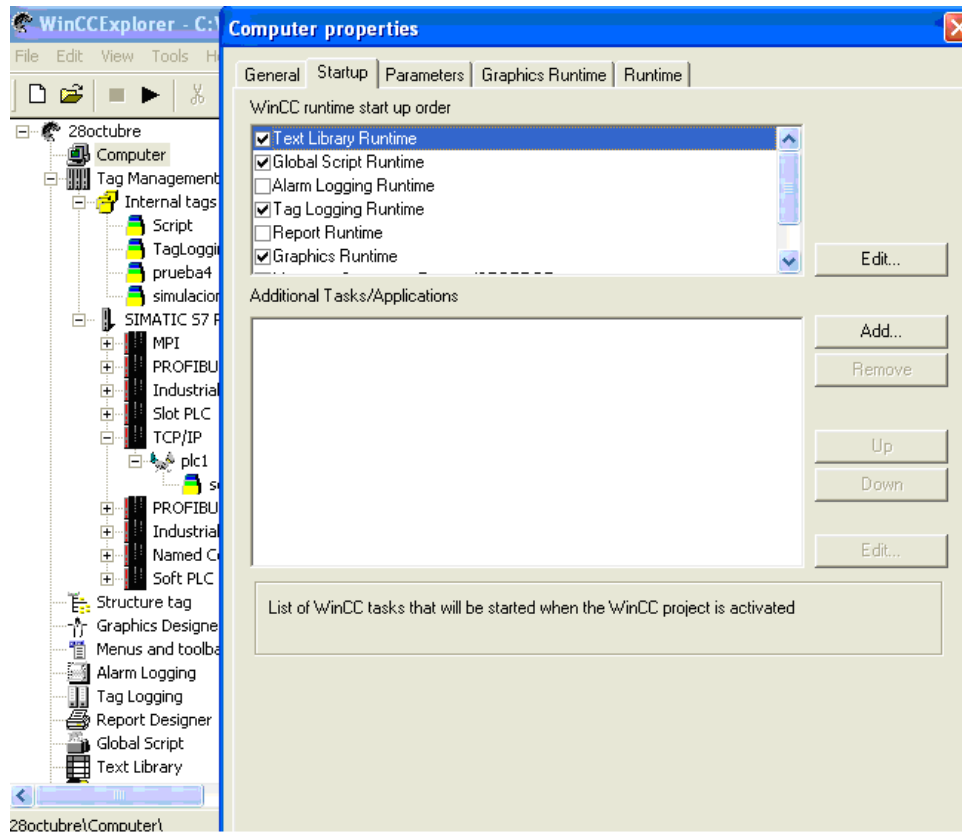


Figura. 3.4. Computer properties

Verifique además que las configuraciones mostradas en las figuras. 3.5 y 3.6. estén presentes:

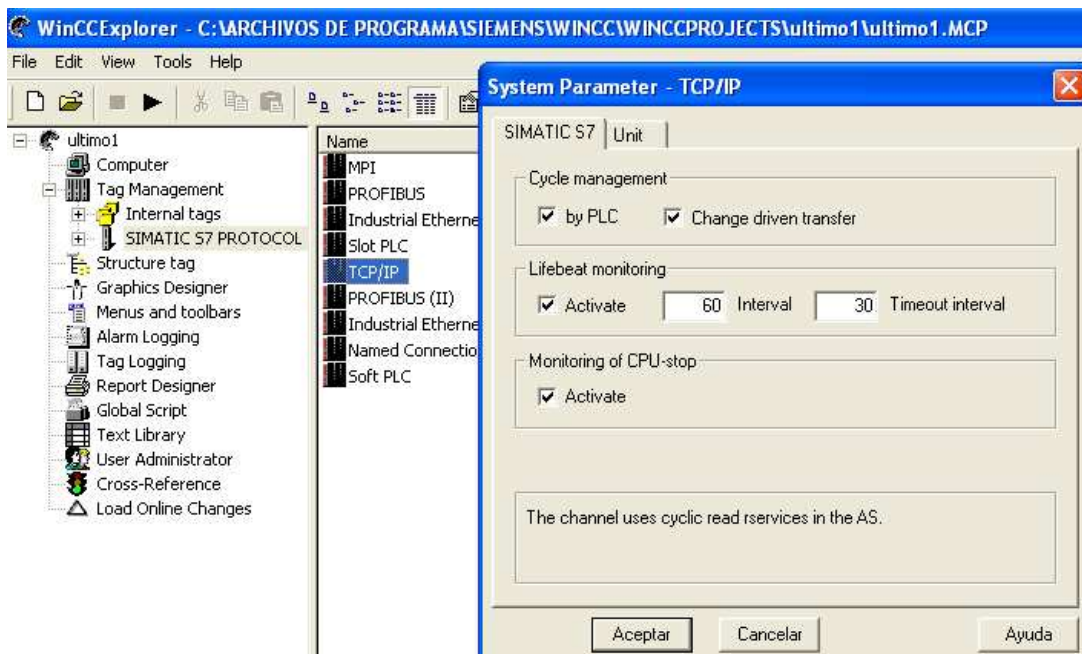


Figura. 3.5. System Parameter: SIMATIC S7

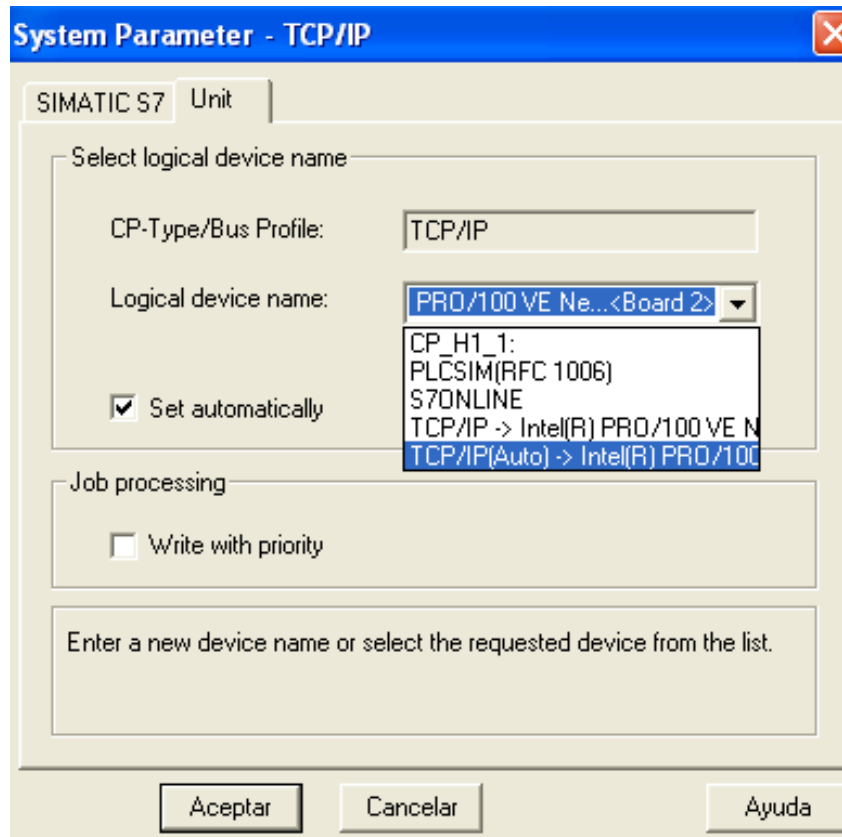


Figura. 3.6. System Parameter: Unit

Ahora en el mismo parámetro TCP/IP (ver figura. 3.7.), en plc1 (ver figura. 3.8.) verifique:

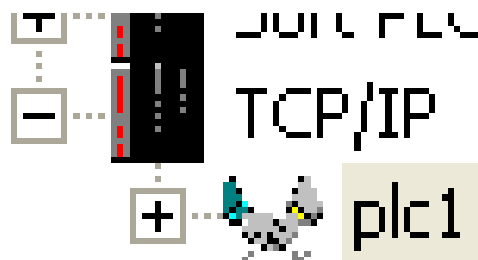


Figura. 3.7. Desplazamiento de TCP/IP

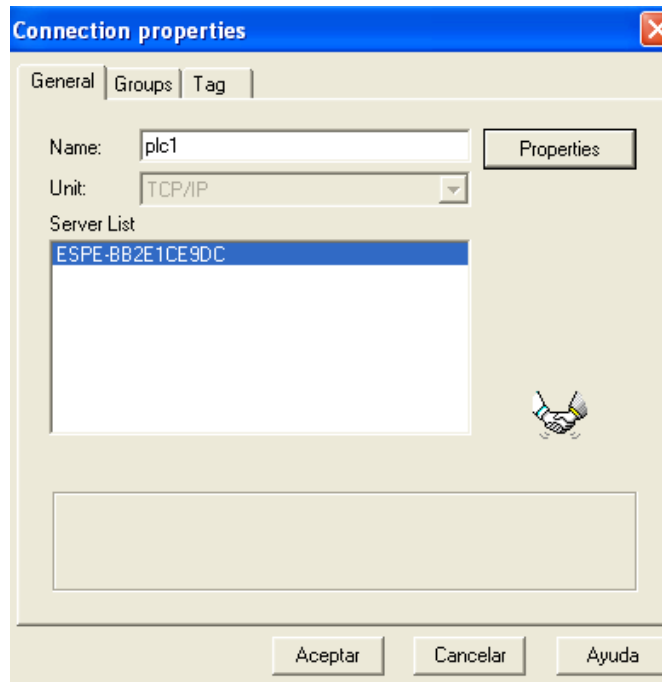


Figura. 3.8. Propiedades de conexión

Presione el botón propiedades, a continuación revise que los parámetros mostrados en la figura. 3.9. se han los correctos:

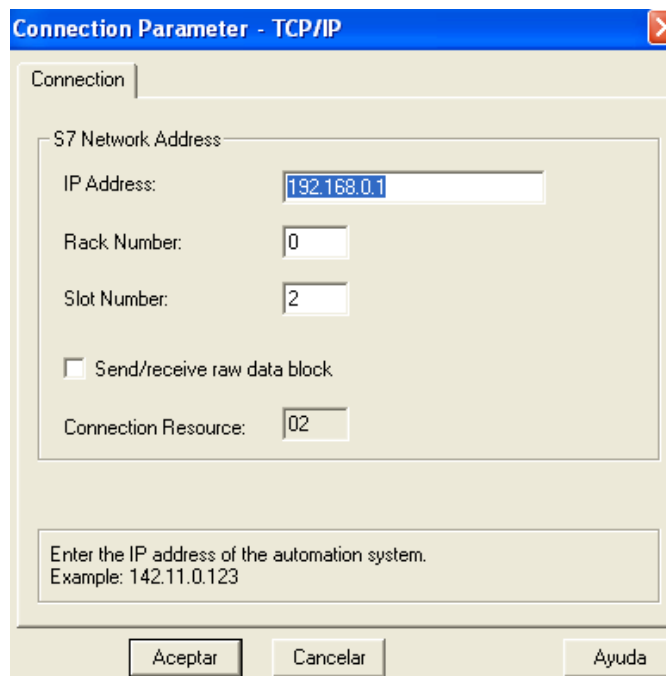


Figura. 3.9. Parámetros de conexión TCP/IP

Acepte.

3.3 FuzzyControl++

3.3.1 Control ActiveX

Para registrar esta herramienta dentro de WinCC vaya al Explorador de WinCC, seleccione el Graphics Designer, de un clic derecho y elija Select ActiveX Control (ver figura. 3.10.).

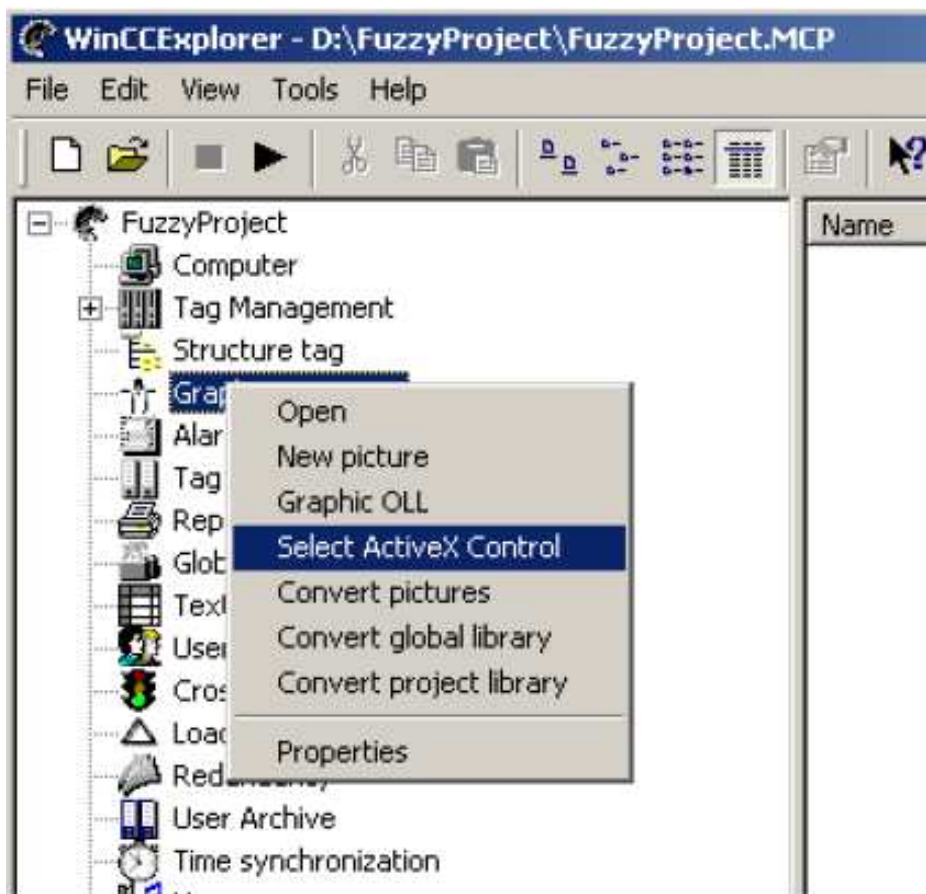


Figura. 3.10. Control ActiveX seleccionar

La siguiente ventana (figura. 3.11.) será mostrada, en ella presione sobre Register OCX....

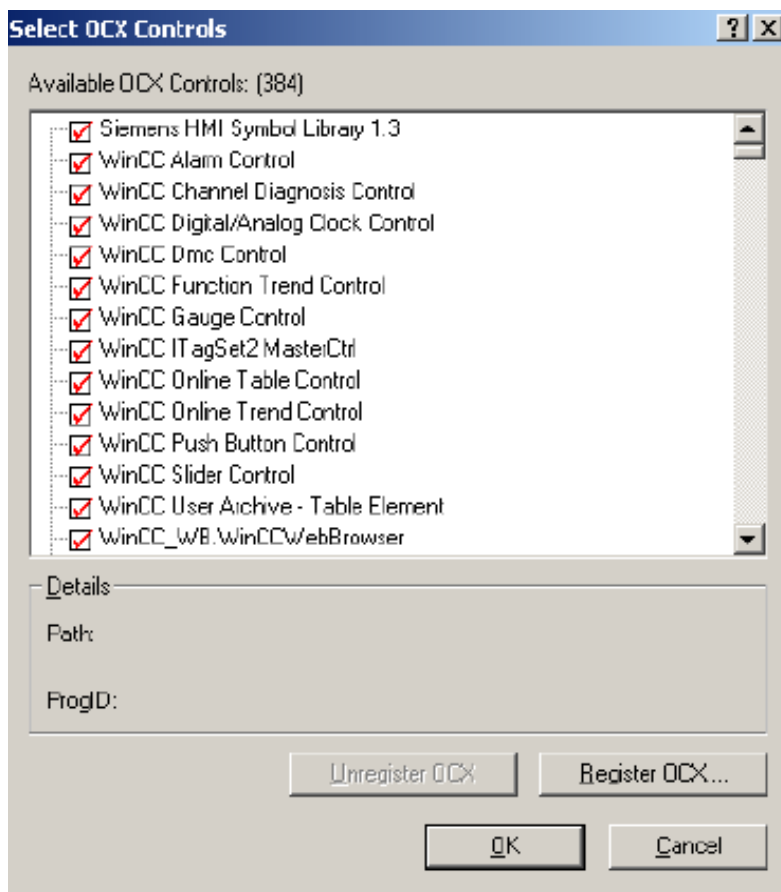


Figura. 3.11. Select OCX Controls

Para finalizar busque el archivo FuzzyControl.ocx y ábralo (ver figura. 3.12.).

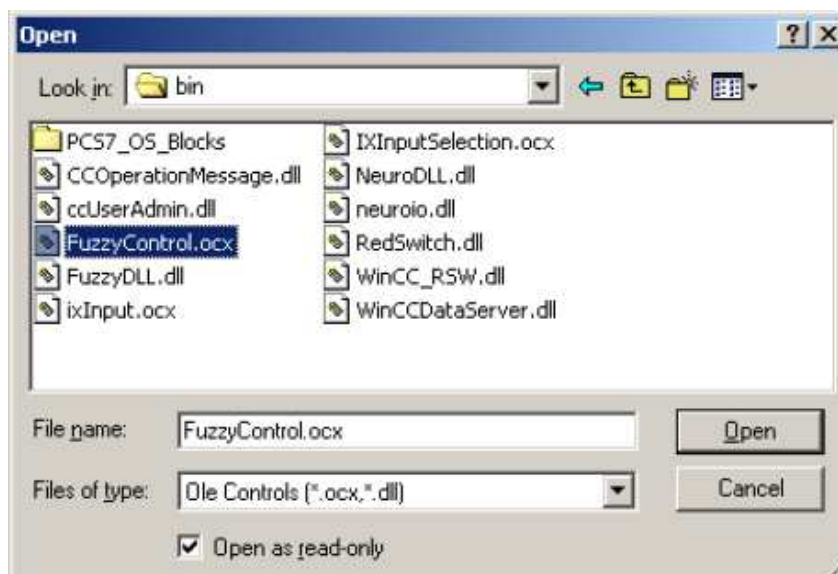


Figura. 3.12. Abrir FuzzyControl.ocx

4. DESCRIPCIÓN DEL CONTROLADOR

Este controlador utiliza el método de control difuso, los datos de la temperatura del cilindro son mandados por el sensor IC (0 a 5V) e ingresan al controlador a través de una entrada analógica. A su vez el *set point* es ingresado por el usuario mediante un campo de entrada en la HMI.

El controlador difuso espera el cambio de estado del *trigger* para entrar en funcionamiento, y una vez ocurrido esto devuelve a la salida un valor de voltaje (0 a 5V) que actuará directamente sobre el elemento de calentamiento para conseguir de este modo el valor de temperatura que se solicita para el cilindro. Los rangos de temperatura se encuentran entre 20°C y 70°C.

Así a medida que varía el voltaje en el elemento de calentamiento la temperatura en el cilindro será la solicitada.

Una interfaz de visualización (HMI) permite controlar y monitorear la temperatura del cilindro en cualquier PC. En esta interfaz se verán las curvas de las variables tanto de la temperatura actual en el cilindro, el *set point* y el voltaje enviado al elemento de calentamiento. La respuesta que proporciona la HMI es en tiempo real del proceso, además esta valida los rangos de entrada.

5. PUESTA EN MARCHA

El programa realizado en Step7 Professional debe ser cargado al PLC, y el selector del S7-300 debe ser colocado en Run.

Conjuntamente el módulo de temperatura debe ser encendido.

El programa realizado en WinCC debe ser abierto, y mandado a correr

presionando el botón



5.1 Programa en WinCC Runtime: Modo de uso

Una vez que se inicialice el programa en modo *runtime*, aparecerá la siguiente pantalla (figura. 5.1.):

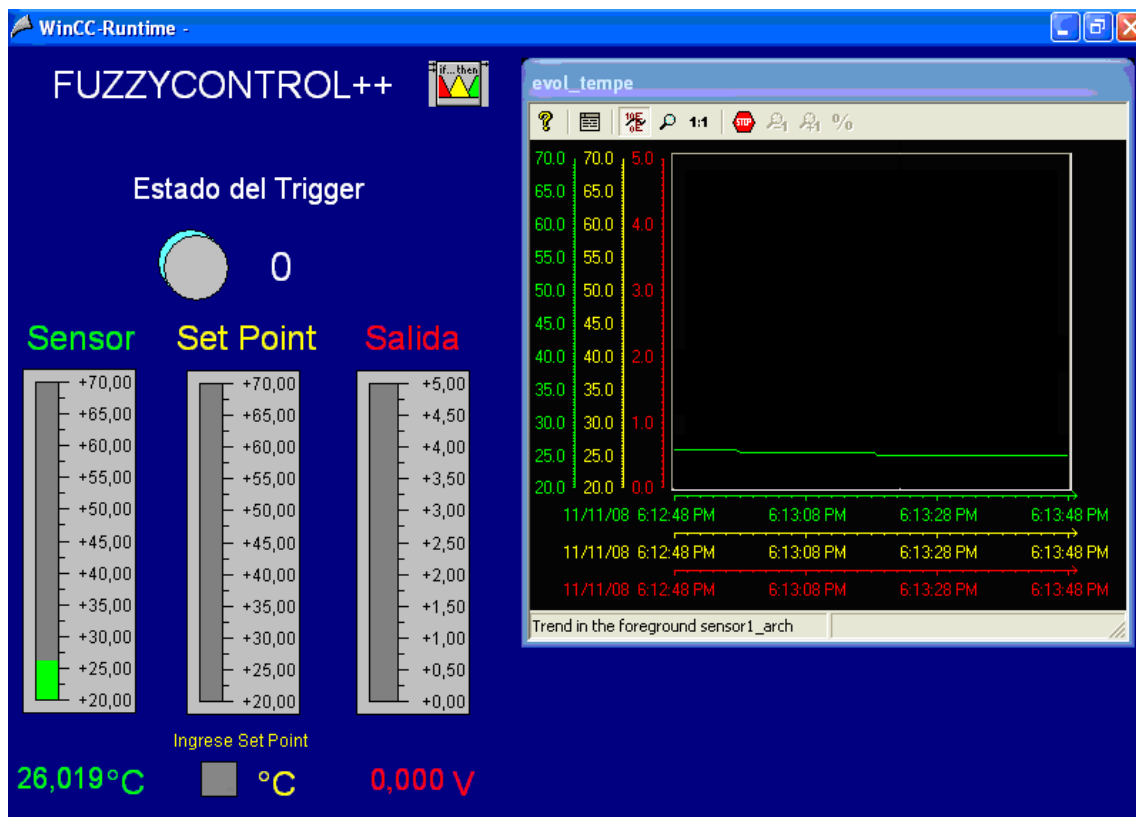


Figura. 5.1. Pantalla inicial

Para que se inicie el control el estado del trigger debe cambiar de cero a uno. La figura. 5.2. indica el estado del *trigger* apagado, y la figura. 5.3. indica el estado del *trigger* encendido.



Figura. 5.2. Trigger apagado

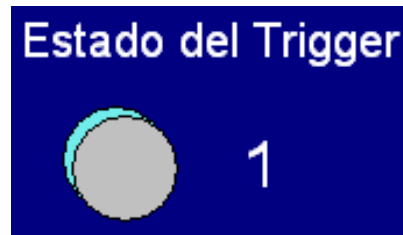


Figura. 5.3. Trigger encendido

Para el ingreso del *set point*, de un clic sobre el cuadro gris y digite un valor entre 20 y 70, presione *Enter*, así como se grafica en la figura. 5.4.



Figura. 5.4. Ingreso de set point

Ahora podrá visualizar el comportamiento del sensor, del *set point* y de la salida en la ventana *evol_tempe* (ver figura. 5.5.).

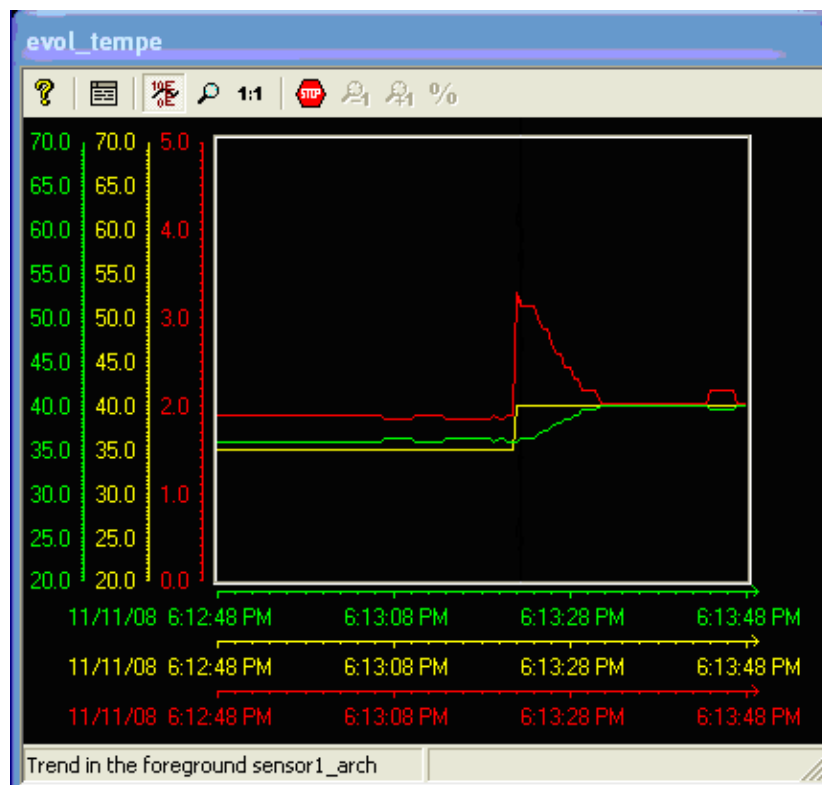


Figura. 5.5. Pantalla evol_tempe


Si presiona el botón  podrá ver el valor de las tres variables en cierto punto del tiempo (ver figura. 5.6.).



Figura. 5.6. Pantalla evol_tempe y valores de variables en el tiempo

Los valores actuales de las tres variables se verán reflejados en sus barras respectivas y sus cifras al pie de estas, tal como lo muestra la figura. 5.7.

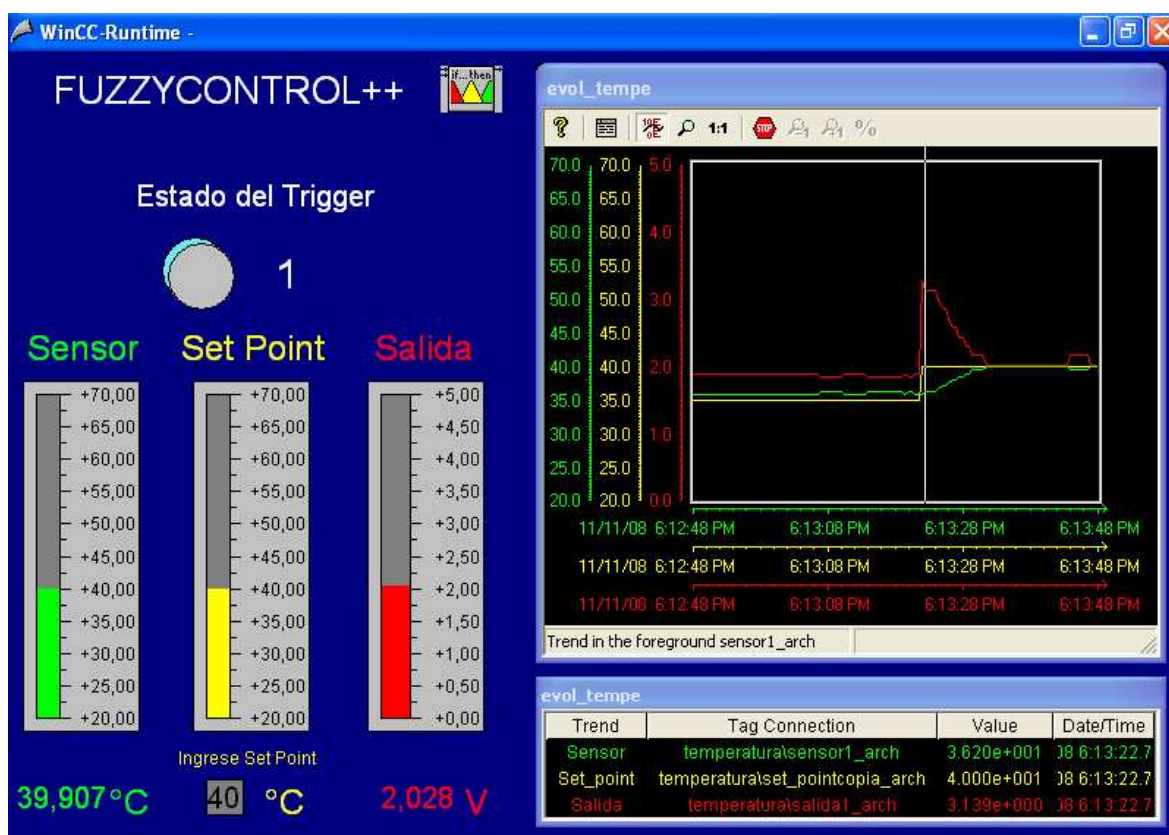


Figura. 5.7. Pantalla en Runtime

6. RECOMENDACIONES

- ✓ Tener cuidado en la conexión de los cables y verificar su funcionamiento.
- ✓ Comprobar si los voltajes de entrada en los dispositivos son los correctos. Si estos voltajes llegan a sobrepasarse, los equipos sufrirán averías.
- ✓ Verificar que la conexión Ethernet esté funcionando.
- ✓ Grabar el programa en el PLC antes de iniciar la aplicación.
- ✓ Lea el manual antes de utilizar el sistema para evitar problemas de uso.
- ✓ Siga todas las instrucciones detenidamente y en el orden indicado.
- ✓ Es importante que no use este sistema cerca de líquidos o similares que pongan en riesgo su funcionamiento.
- ✓ Instale el sistema en un lugar estable para así evitar problemas de control y lesiones que podrían sufrir los operarios.
- ✓ Nunca introducir objetos de ninguna clase por las ranuras y respiraderos de esta unidad. Esto podría causar riesgo de incendio o de descarga. Nunca derramar ningún líquido sobre la unidad.

-
- ✓ No desarmar la unidad. Cuando sea necesario realizar reparaciones se debe llevar la unidad a un centro de soporte técnico autorizado.

A2 HOJA TÉCNICA DE LA CPU 315F-2 PN/DP

CPU	CPU 315F-2 PN/DP
Bus system	PROFIBUS PROFINET (PN)
Design	S7-300 with central and/or distributed fail-safe I/O
Application	Médium performance range
Main memory	256 KB
Load memory (plug-in)	64 KB - 8 MB
Bit memories	16 Kbit
FB/FC/DB	2048/2048/1023
Fail-safe I/O	Up to 320
I/O address area	2 KB/2 KB
Process I/O image	384 Byte /384 Byte
Interfaces	PN and MPI/DP
Number of bus stations	128/124
Processing times	
· Bit operations	0,2 µs
· Word operation	0,2 µs
· Fixed-point arithmetic	2 µs
· Floating-point arithmetic	3 µs
Bit memories/timers/counters	
· Bit memories	2048 byte
· S7 timers / counters	256/256
· IEC timers / counters	✓
Organization blocks (OB)	free cycle (OB 1), real-time interrupt (OB 10), delay alarm (OB 20), (OB 21 [not 315-2 DP, 315-2 PN/DP]) time-triggered (OB 35), (OB 32 - 34 [not with 315-2 DP, 315-2 PN/DP]) interrupt-triggered (OB 40), DPV1 restart (OB 55-57), restart (OB 100) asyn. error (OB 80, 82,83 [315-2 PN/DP and 317-2 PN/DP only], 85-87), syn. error (OB 121,122)
Address ranges	
· I/O address area	2048/2048 byte
· I/O process image	128/128 byte
· Digital channels (central)	1024
· Analog channels (central)	256
Expansions	
· Racks	max. 4
· Modules per rack	8
DP interfaces	
· Number of DP master systems int./CP 342-5	1/4 ✓
· Equidistant	✓
· Activat./deact. DP slaves	✓
· Transmission speed	12 Mbit/s
· No. of slaves per station	124

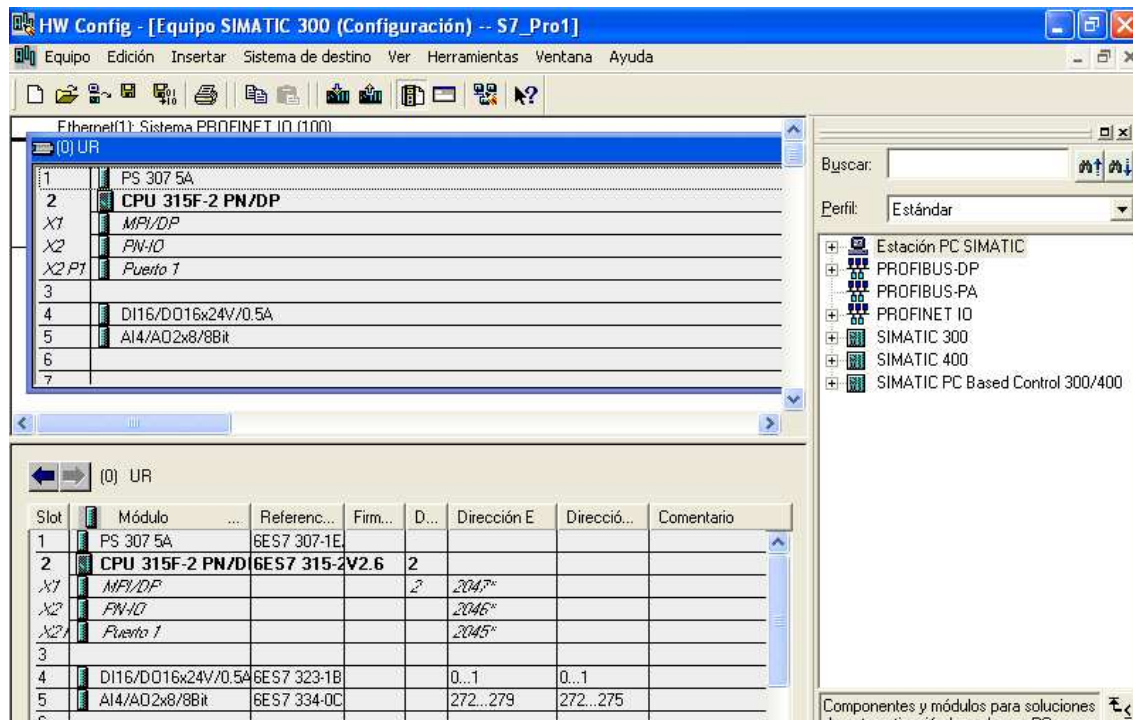
• Lateral communication	✓
PROFINet interface	✓
• Transmission rate	100 MBit/s
• PROFINET CBA	✓
• PROFINET IO	✓
• S7 communication	✓
• PG/OP communication	✓
• TCP/IP	✓
Dimensions	40 x 125 x 130

A3 HOJA TÉCNICA DEL MÓDULO ANÁLOGO SM 334; AI 4/AO 2 x 8/8 Bit

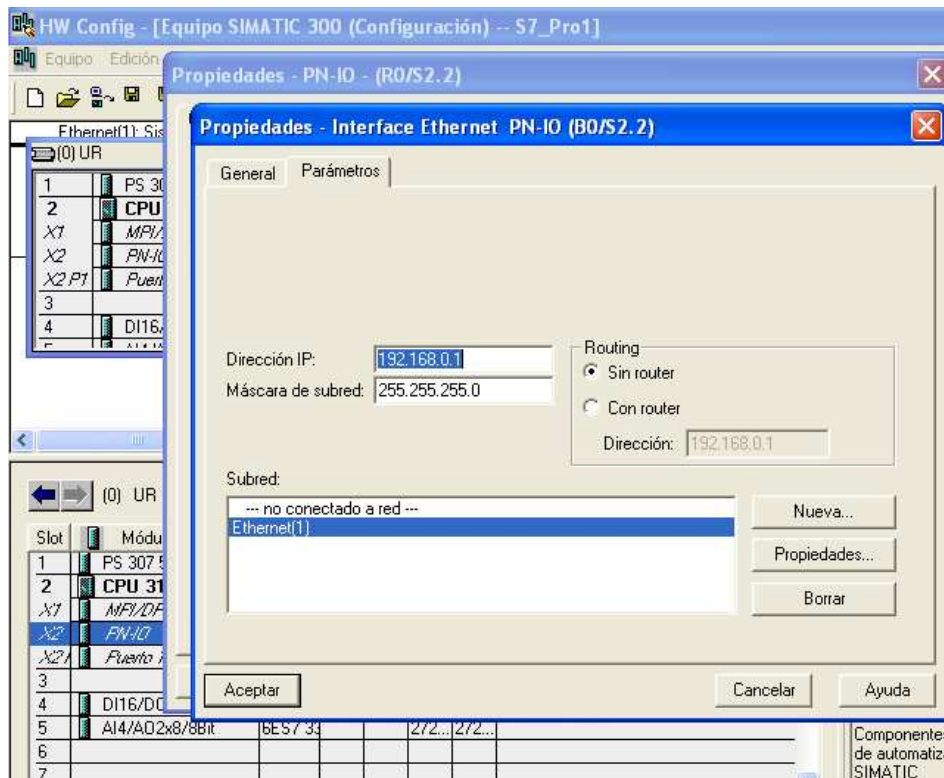
MÓDULO	SM 334; AI 4/AO 2 x 8/8 Bit
Dimensiones y peso	Dimensiones A x A x P (en mm) 40 x 125 x 117 Peso aprox. 285 g
Datos específicos del módulo Soporta operación sincronizada Cantidad de entradas Cantidad de salidas Longitud de cable <ul style="list-style-type: none"> • con pantalla máx. 	No 4 2 200m
Tensiones, intensidades, potenciales Tensión nominal de carga L Tensión nominal de alimentación para electrónica y tensión nominal de carga L+ Separación galvánica <ul style="list-style-type: none"> • entre canales y bus posterior • entre canales y tensión de alimentación de la electrónica • entre los canales Diferencia de potencial admisible <ul style="list-style-type: none"> • entre entradas y MANA (UCM) • entre las entradas (UCM) Aislamiento ensayado con Consumo <ul style="list-style-type: none"> • del bus posterior máx. • de la tensión de alimentación y de carga L+ (sin carga) máx. Disipación del módulo típ.	+ 24 V c.c. 24 V c.c. no si no 1 V c.c. 1 V c.c. 500 V c.c. 55 mA 110 mA 3 W
Datos para la selección de un sensor Margen de entrada (valores nom.)/resistencia de entrada <ul style="list-style-type: none"> • Tensión • Intensidad Tensión de entrada admisible para las entradas de tensión (límite de destrucción) Intensidad de entrada admis. para las entradas de intensidad (límite de destrucción) Conexión de los sensores <ul style="list-style-type: none"> • medición de tensión 	0 a 10 V/100 k Ω 0 a 20 mA/50 W máx. 20 V perman.; 75 V durante máx. 1 s (relación puls./pausa 1:20) 40 mA posible

<ul style="list-style-type: none"> para medición de intensidad como transductor a 2 hilos como transductor a 4 hilos 	<p>imposible posible</p>
<p>Datos para la selección de un actuador</p>	
<p>Márgenes de salida (valores nominales)</p> <ul style="list-style-type: none"> Tensión Intensidad 	<p>0 a 10 V 0 a 20 mA</p>
<p>Resistencia de carga (en el margen nominal de la salida)</p> <ul style="list-style-type: none"> salidas de tensión <ul style="list-style-type: none"> – carga capacitiva salidas de intensidad <ul style="list-style-type: none"> – carga inductiva 	<p>mín. 5 kW máx. 1 mF máx. 300 W máx. 1 mH</p>
<p>Salida de tensión</p> <ul style="list-style-type: none"> protección contra cortocircuitos corriente de cortocircuito 	<p>sí máx. 11 mA</p>
<p>Salida de intensidad</p> <ul style="list-style-type: none"> tensión en vacío 	<p>máx. 15 V</p>
<p>Límite de destrucción por tensiones/intensidades aplicadas desde el exterior</p> <ul style="list-style-type: none"> tensión en las salidas respecto a MANA Intensidad 	<p>máx. 15 V perman. máx. 50 mA c.c.</p>
<p>Conexión de actuadores</p> <ul style="list-style-type: none"> para salida de tensión <ul style="list-style-type: none"> Conexión a 2 hilos conexión a 4 hilos (cable de medición) 	<p>posible imposible</p>
<p>Conexión de los sensores</p> <ul style="list-style-type: none"> medición de intensidad <ul style="list-style-type: none"> Conexión a 2 hilos 	<p>Posible</p>

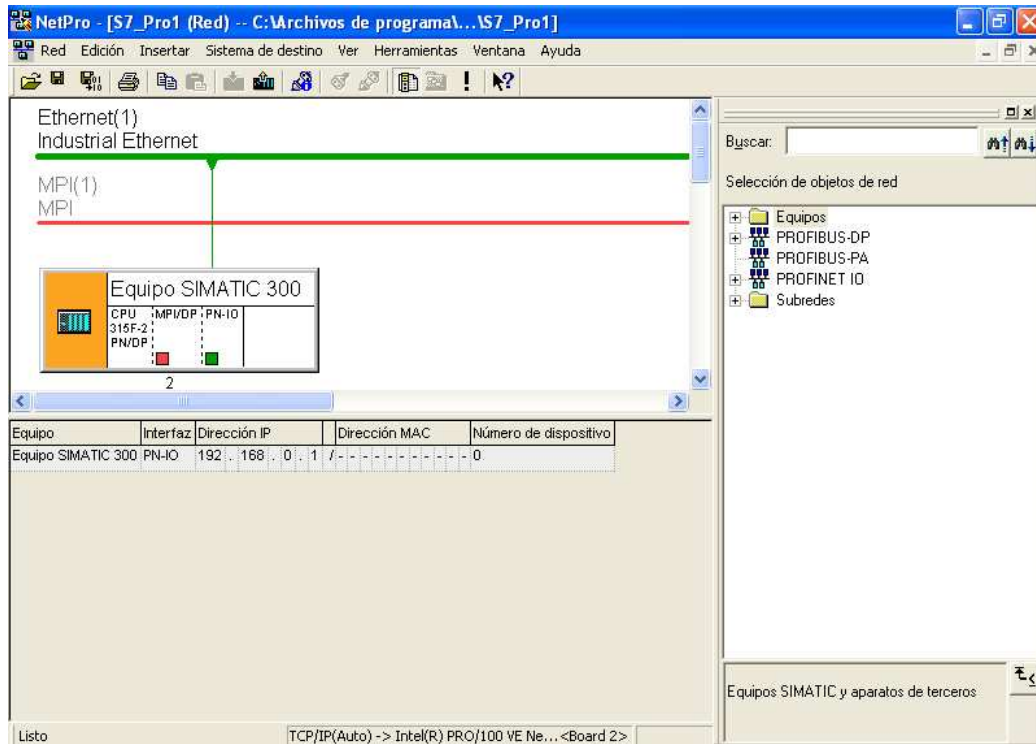
A4 Configuración del Hardware



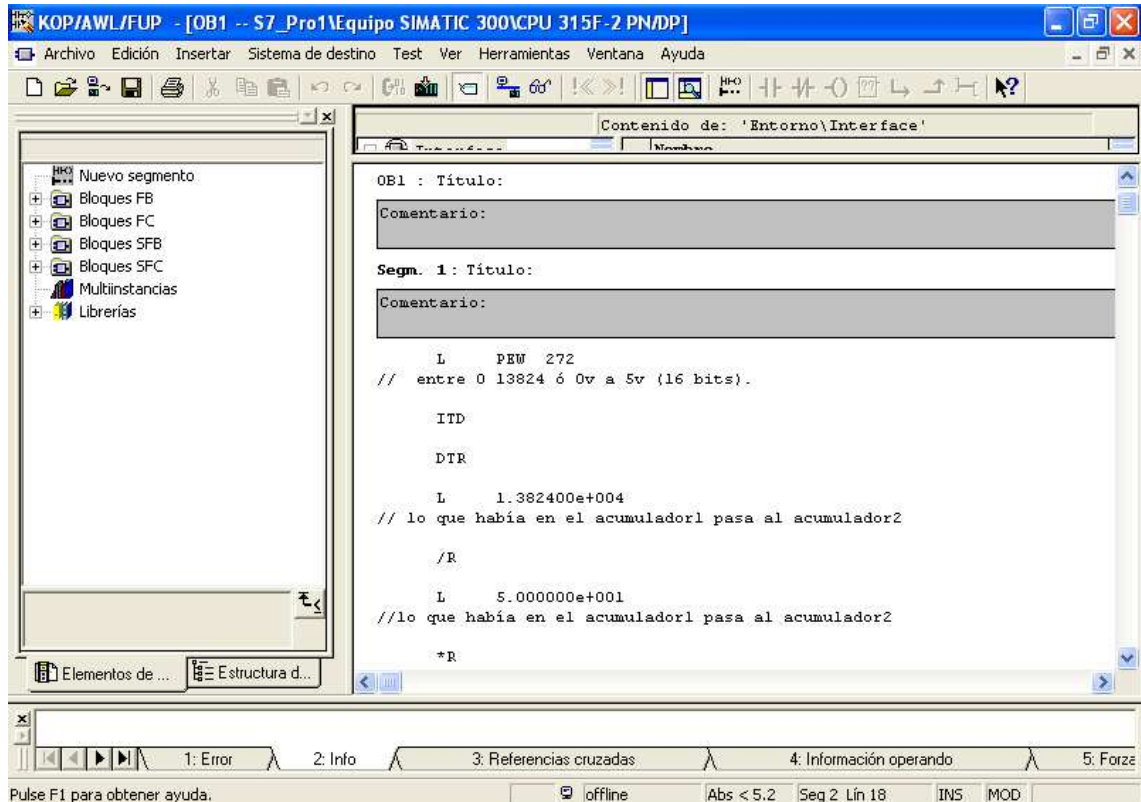
A5 Direccionamiento del PLC



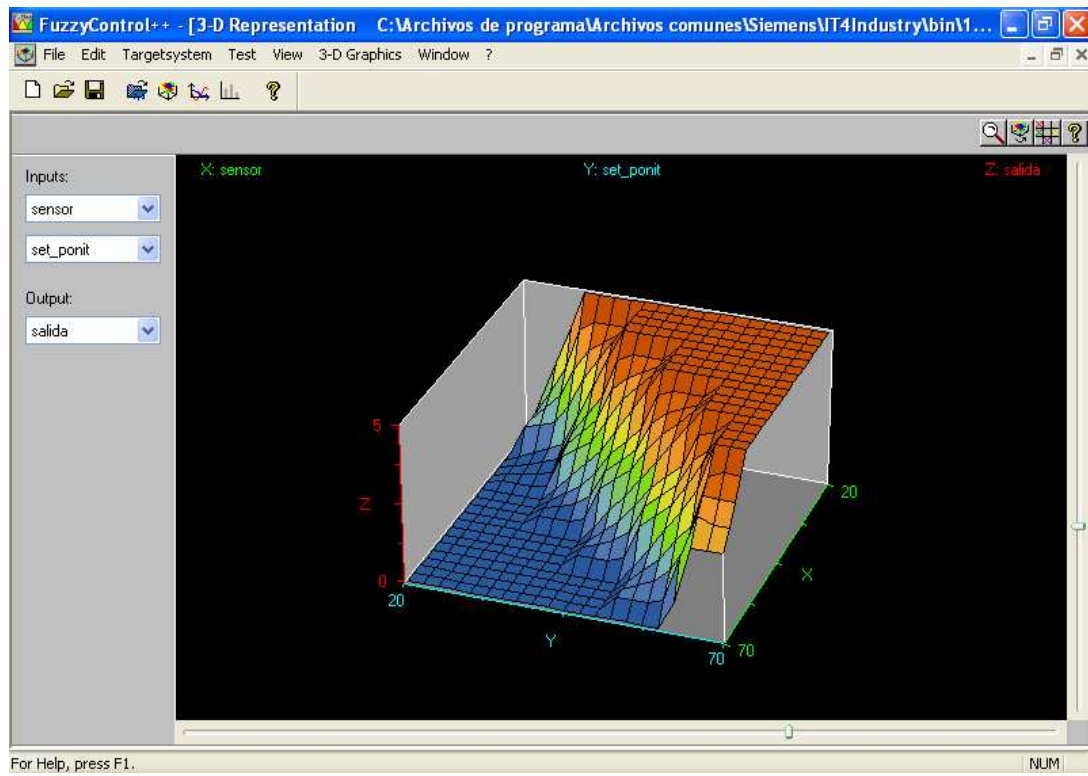
A6 NetPro



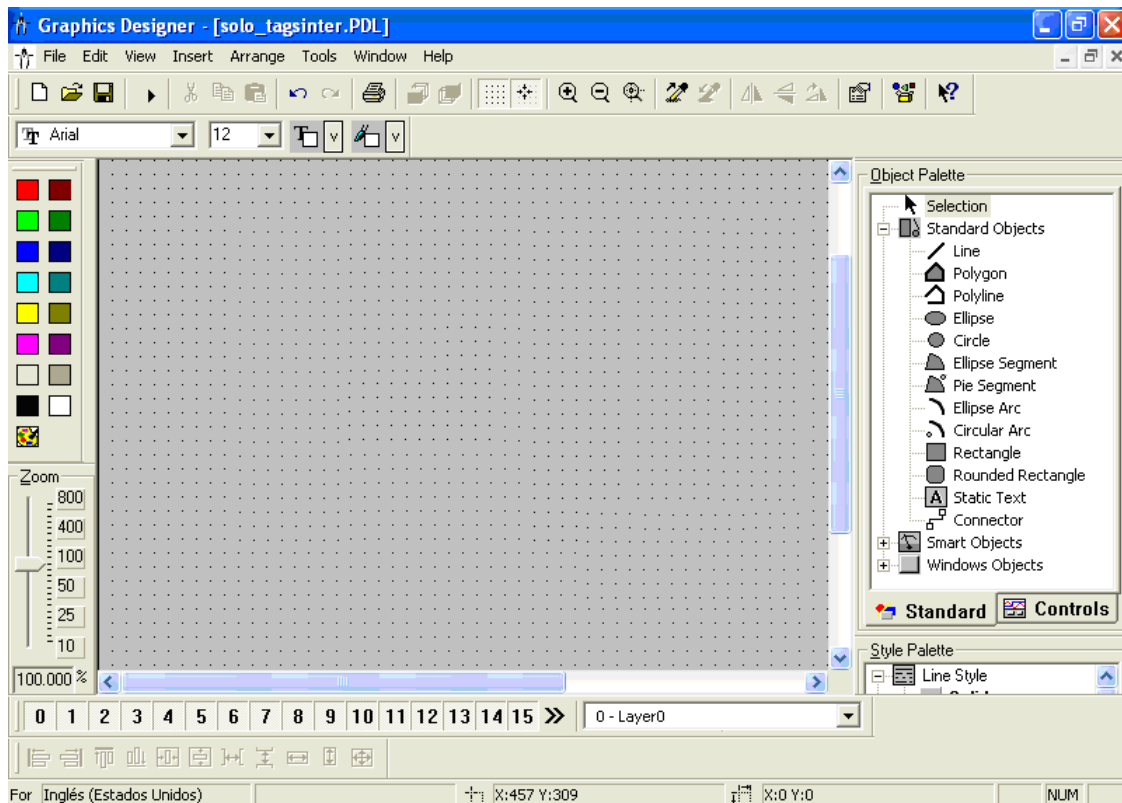
A7 Editor de programa



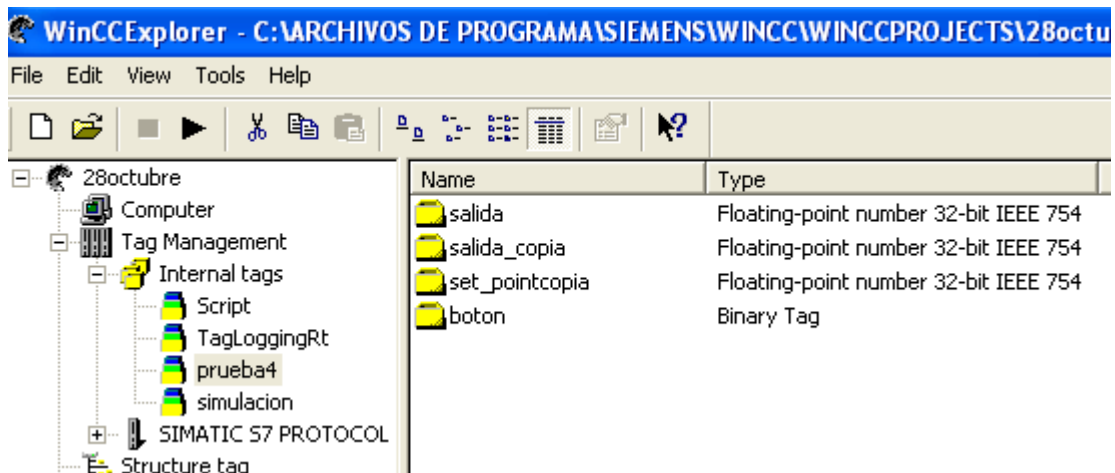
A8 Representación en 3D del programa de control de temperatura



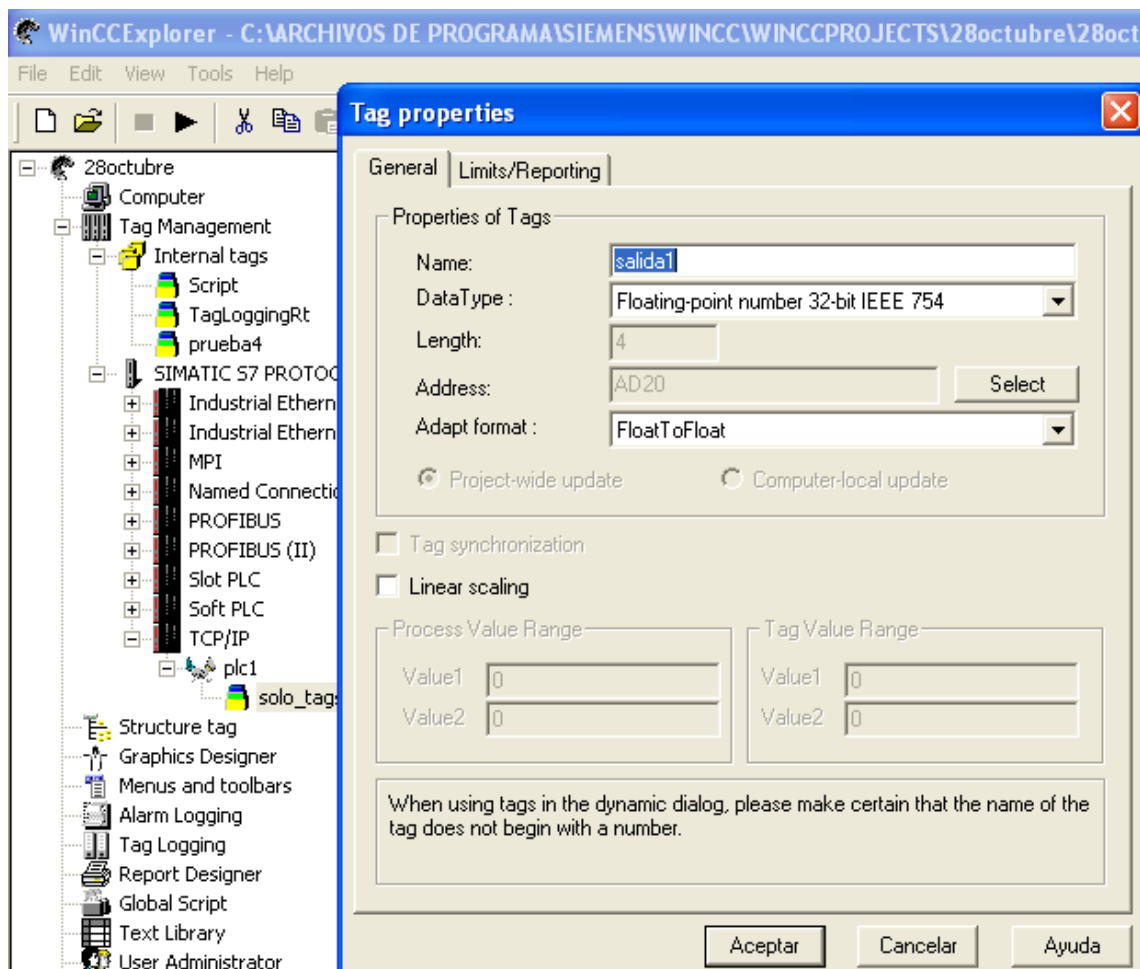
A9 Ventana del Graphics Designer



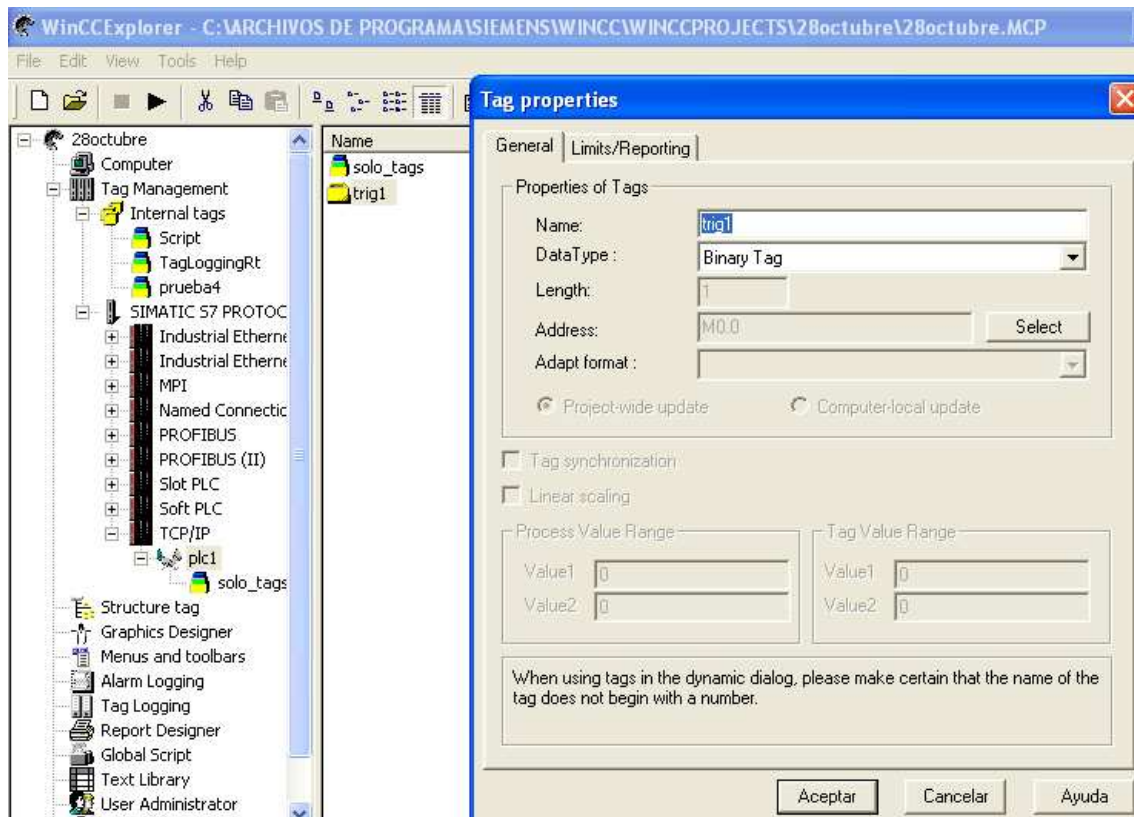
A10 Tags internos



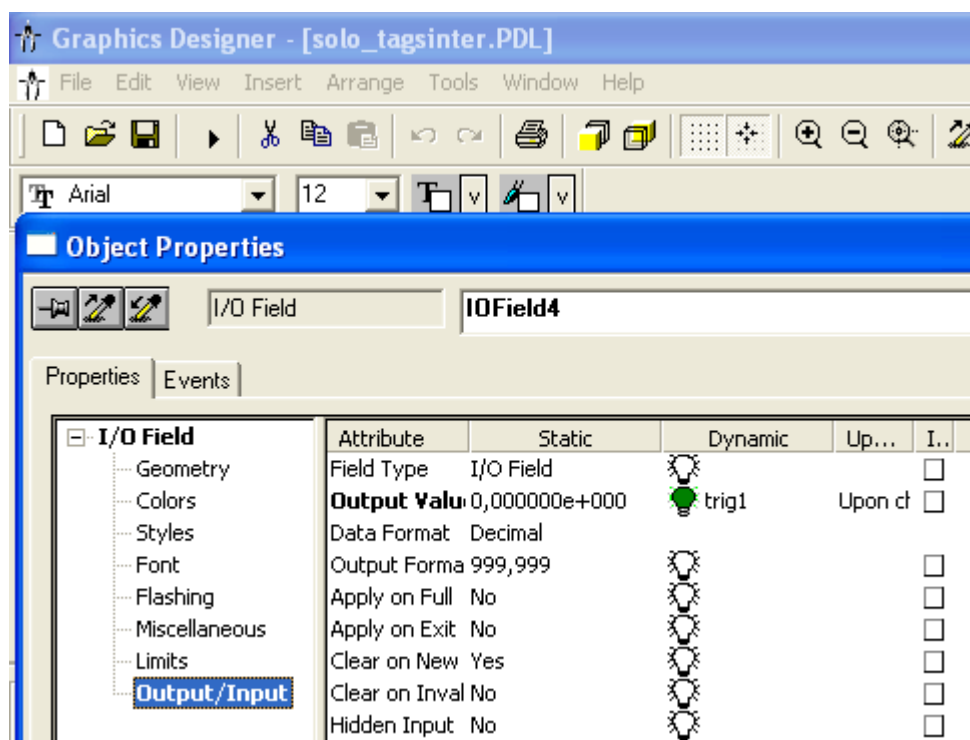
A11 Creación de un tag externo



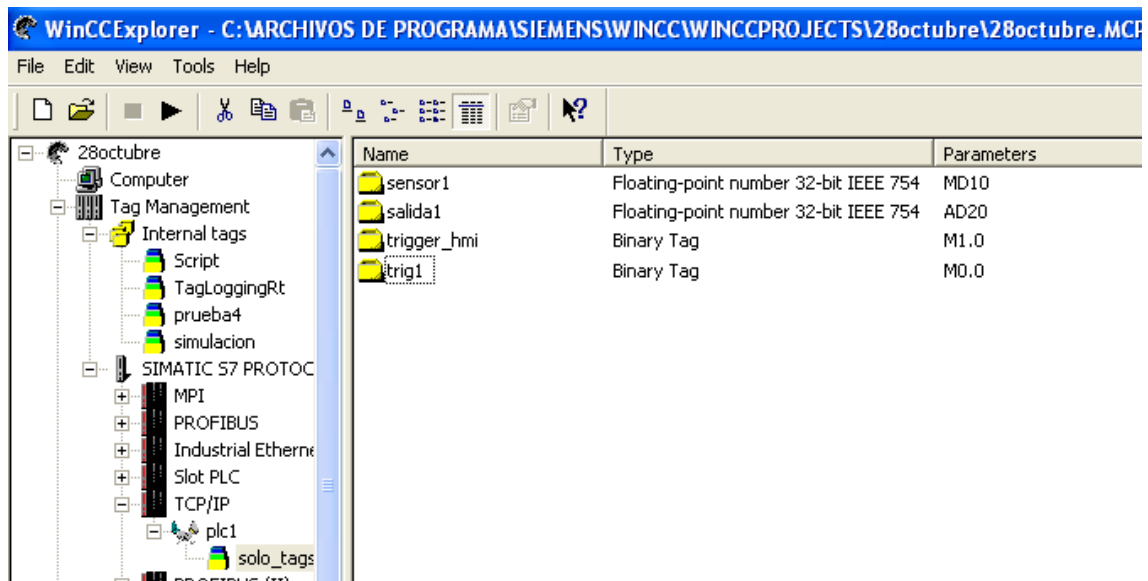
A12 Creación del tag externo para el trigger



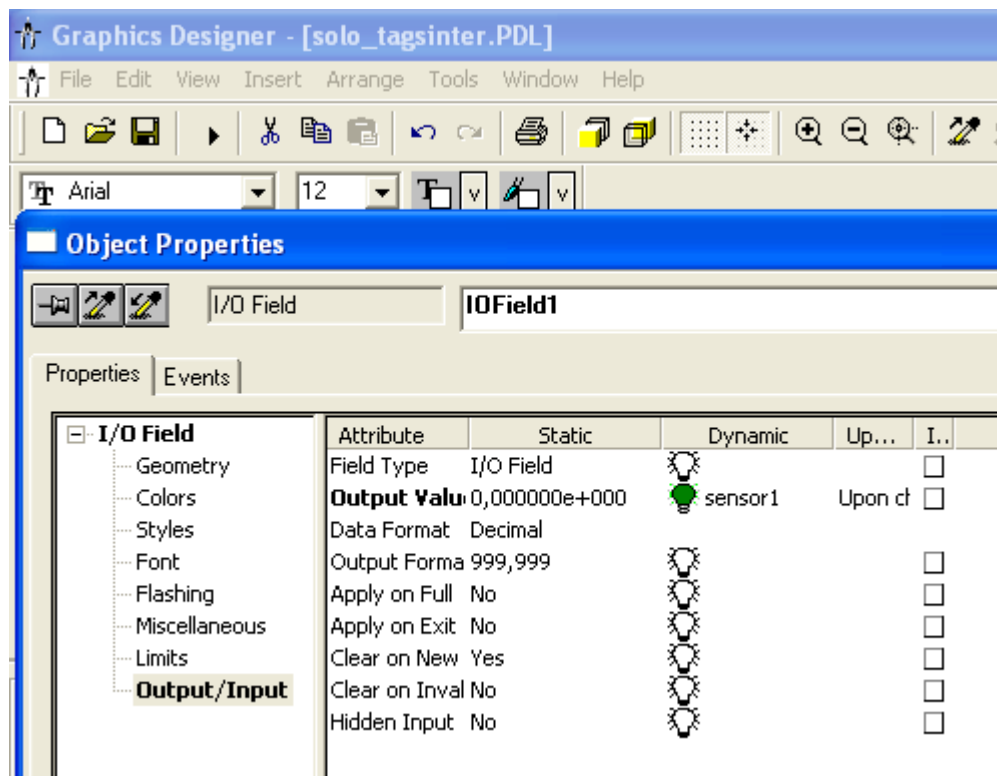
A13 Direccionamientos por tags del trigger el campo de entrada/salida



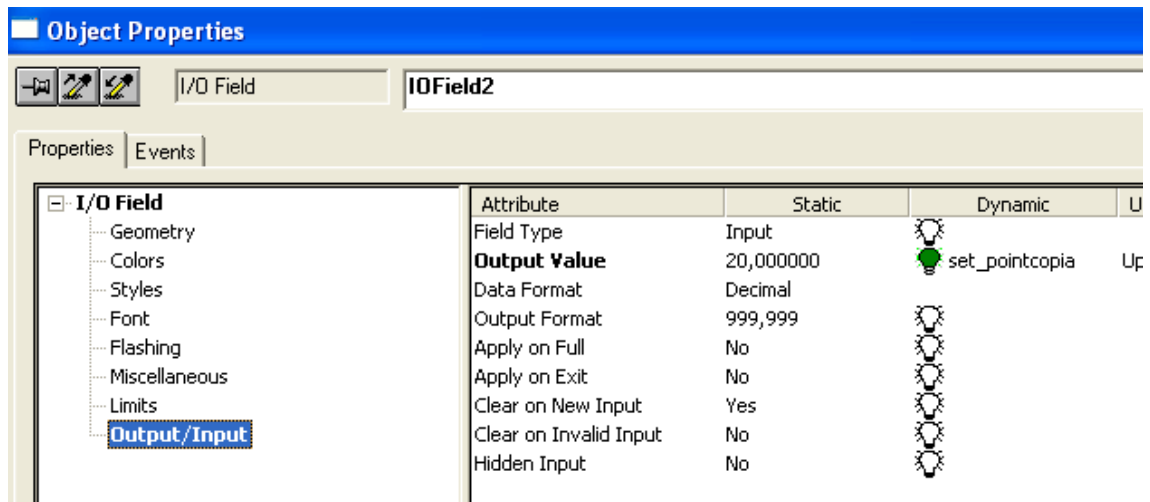
A14 Tags externos



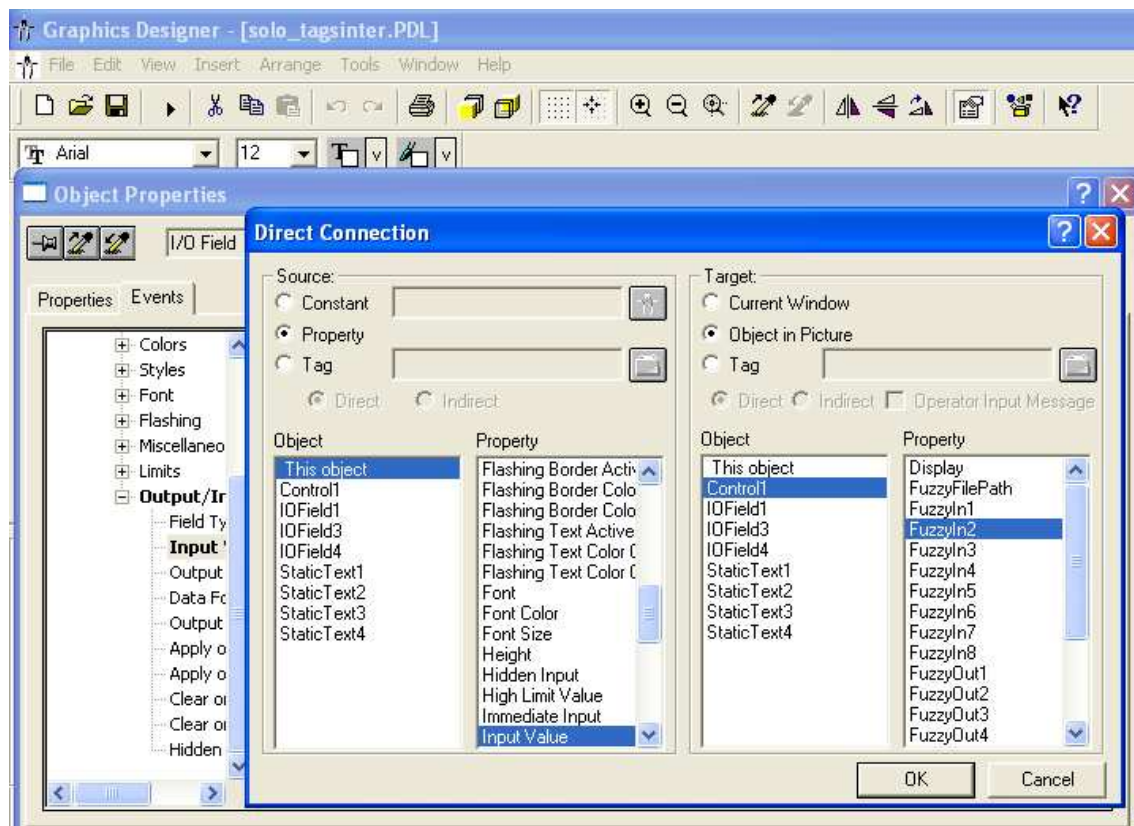
A15 Direccinamiento por tags del sensor para el campo de entrada/salida



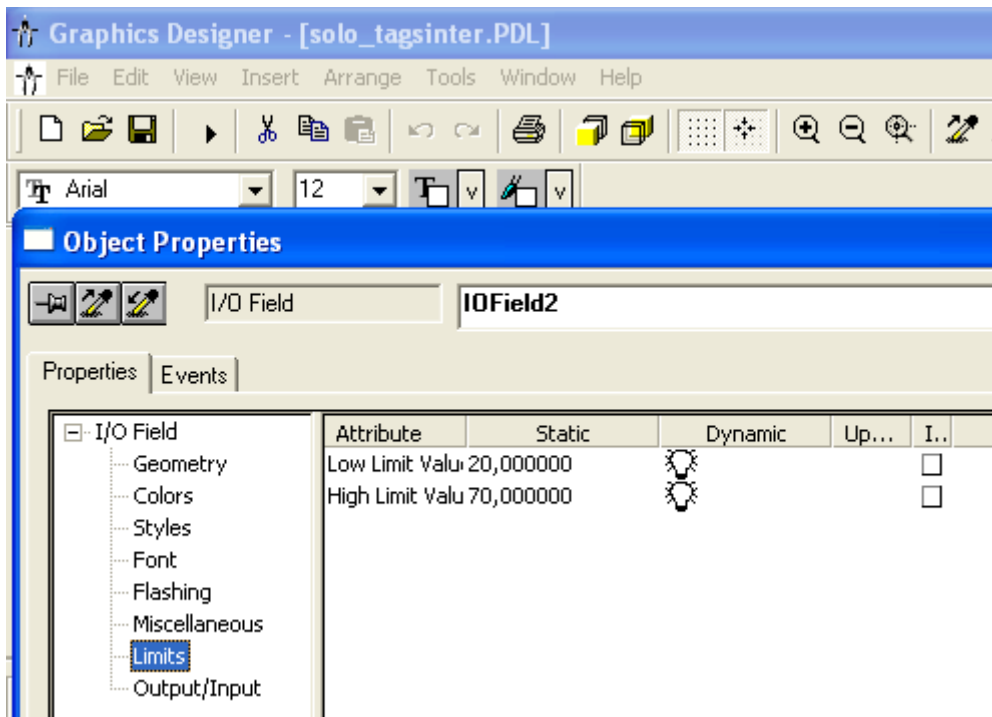
A16 Direccinamiento para el campo de entrada/salida del *set point*



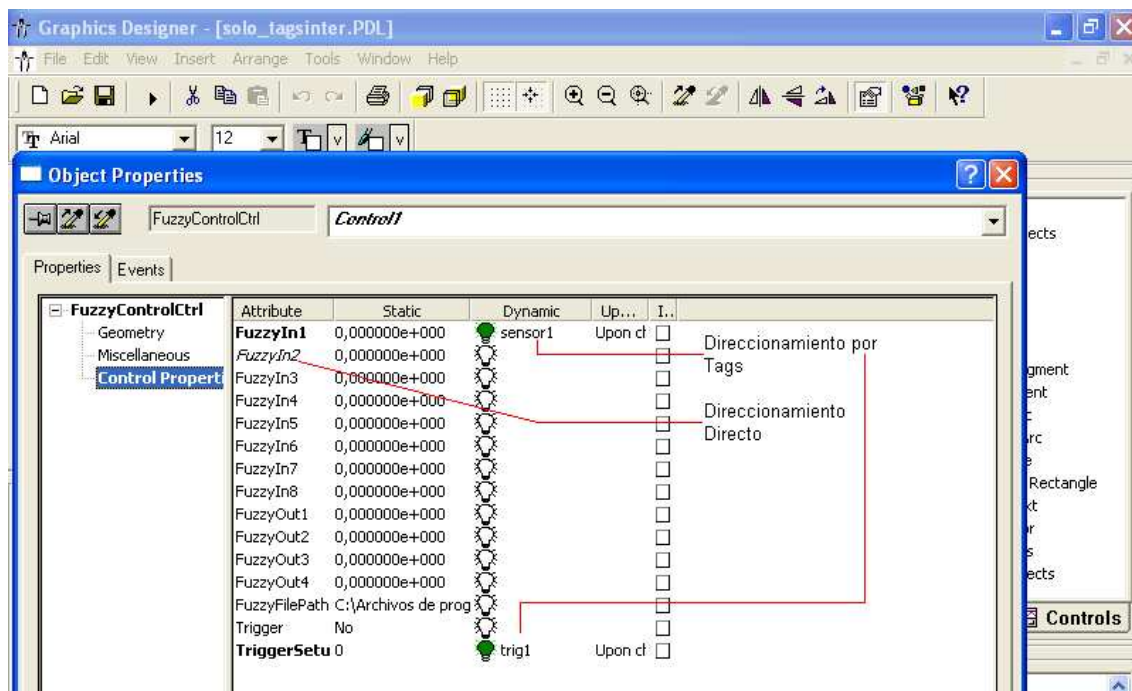
A17 Direccinamiento directo del *set point* hacia la entrada2 del control FuzzyControl



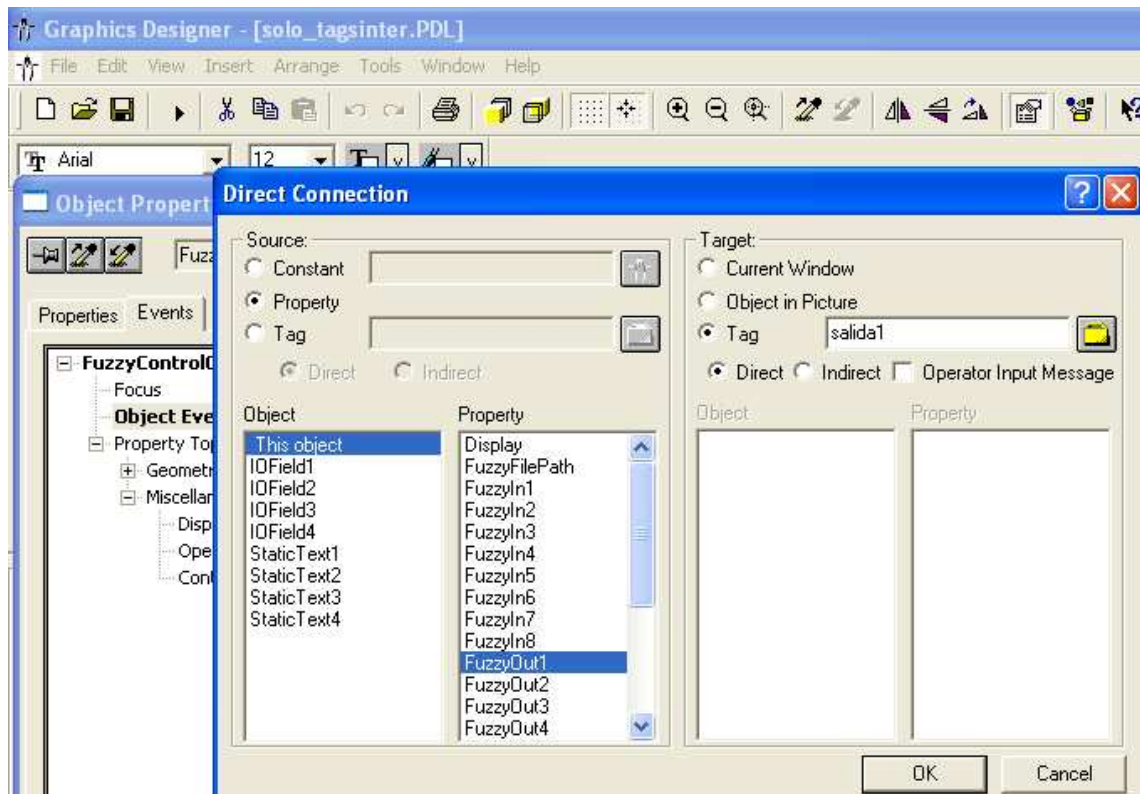
A18 Validación del rango para el set point



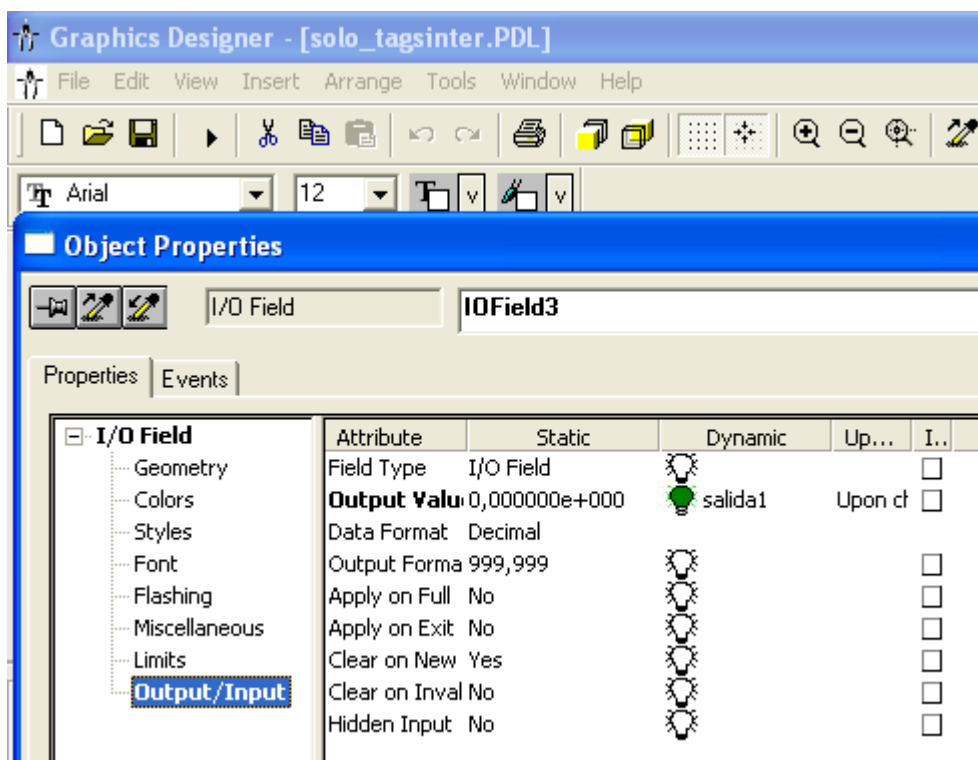
A19 Direccionamiento del control FuzzyControl



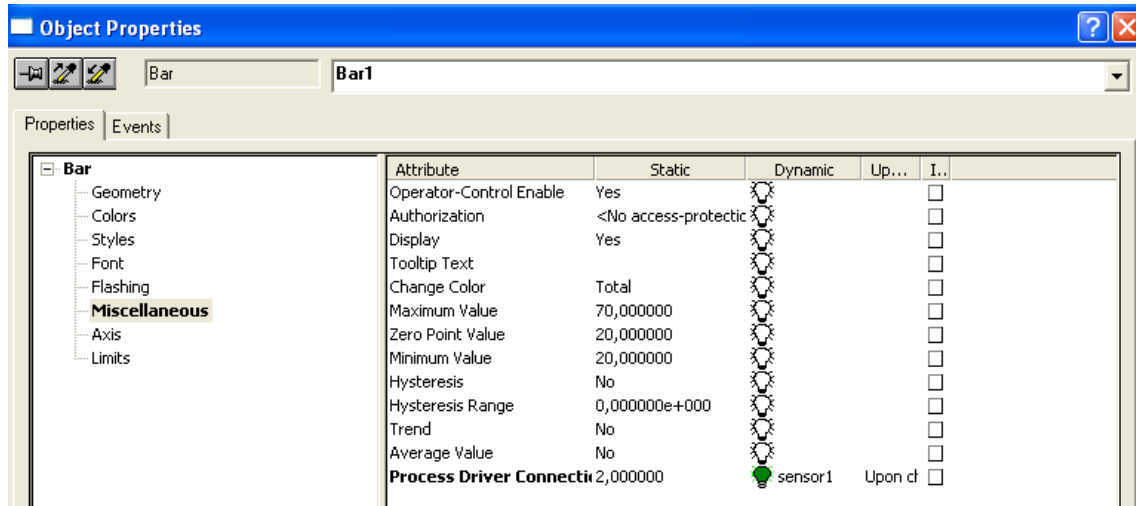
A20 Direccionamiento directo de la salida del control FuzzyControl



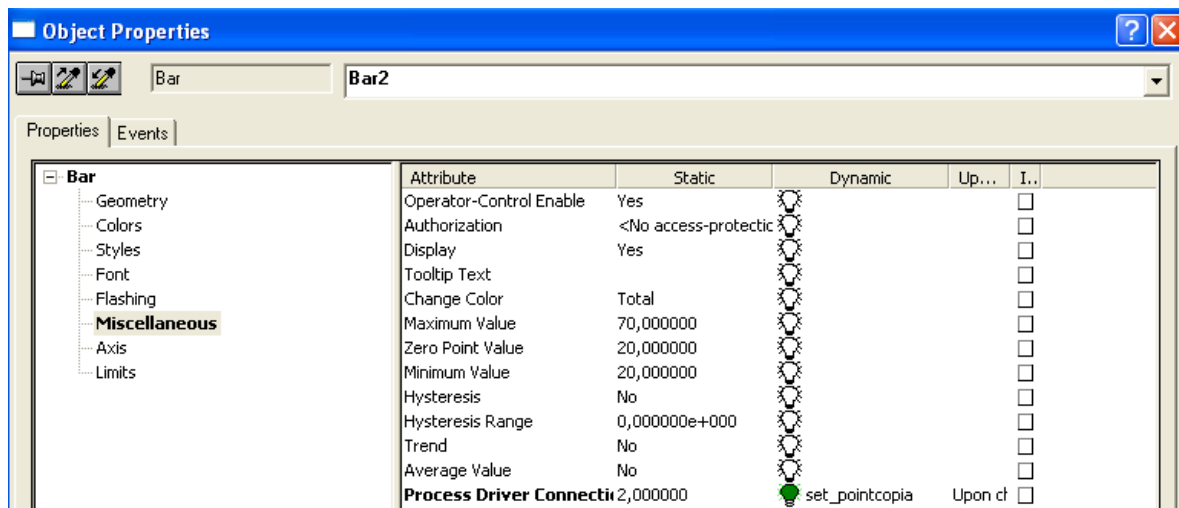
A21 Direccionamiento por tags de la salida para el campo de entrada/salida



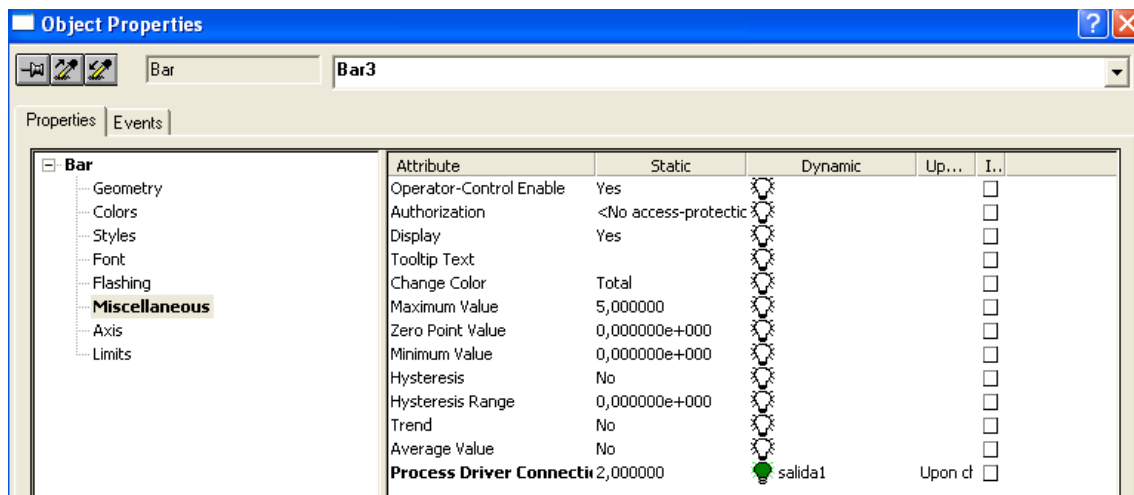
A22 Direcccionamiento de la Barra del sensor



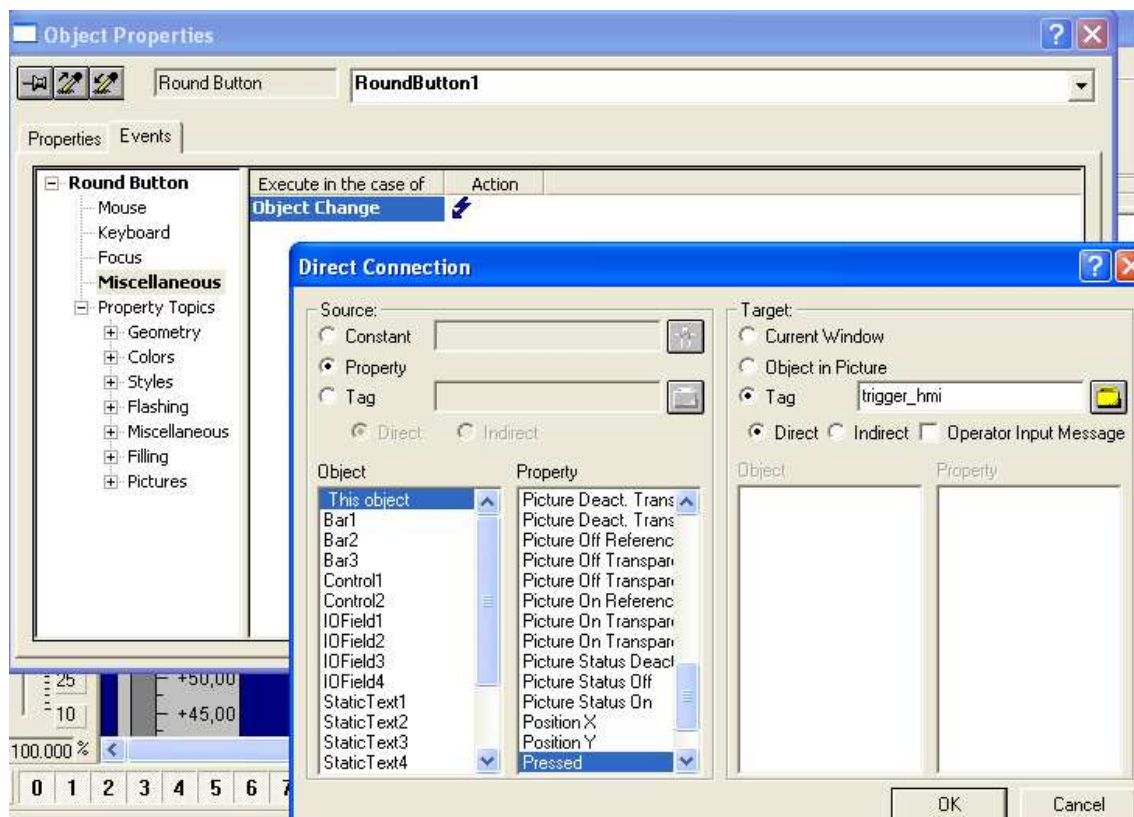
A23 Direcccionamiento de la Barra del set point



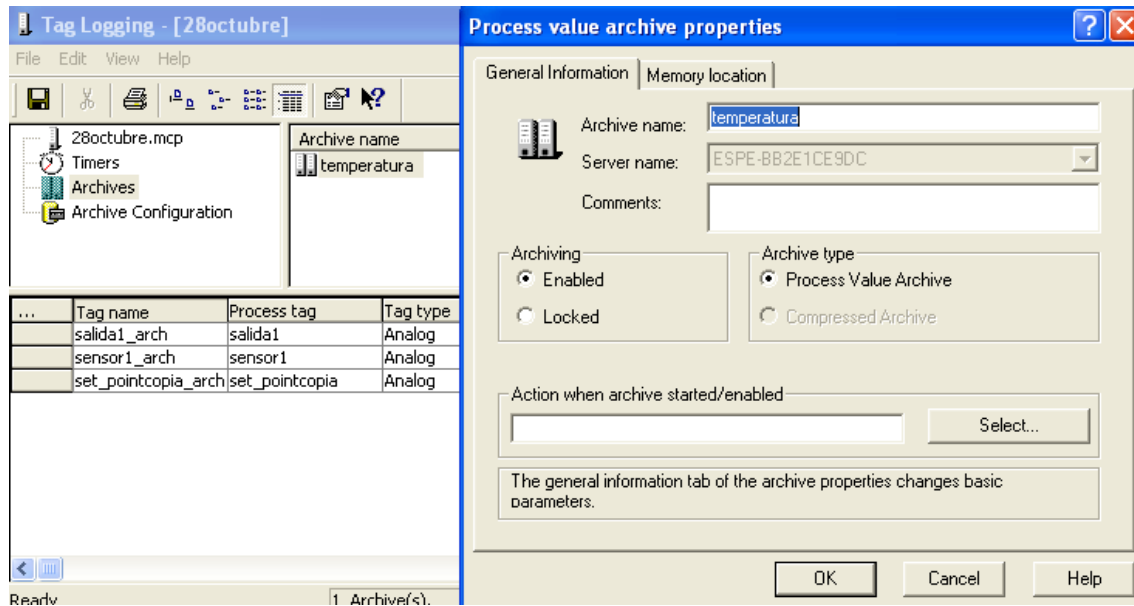
A24 Direcccionamiento de la Barra de salida



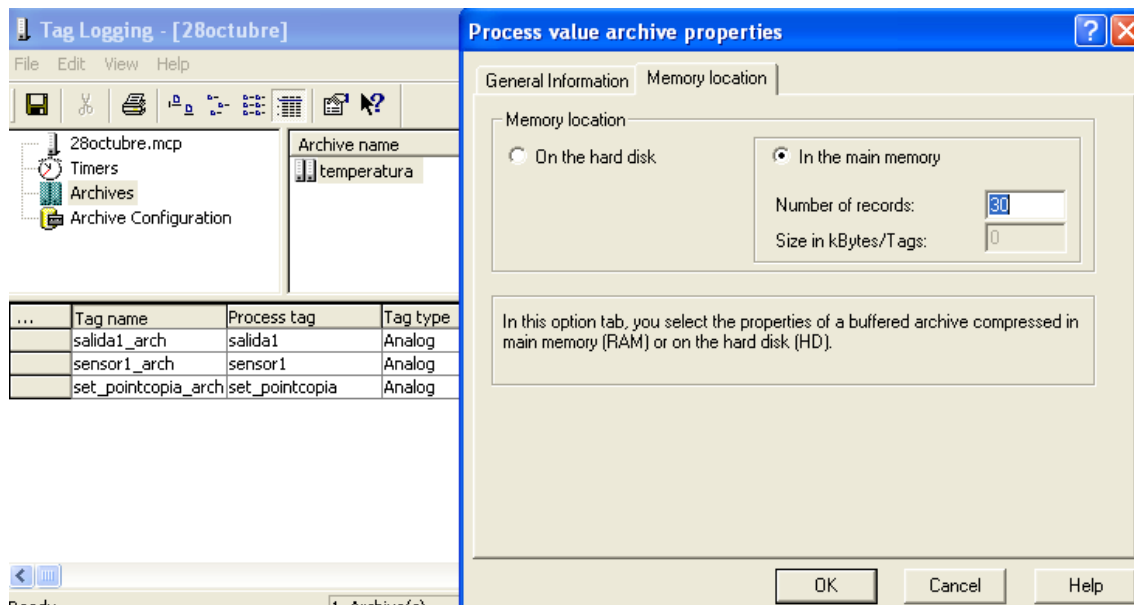
A25 Direcccionamiento del botón del *trigger*



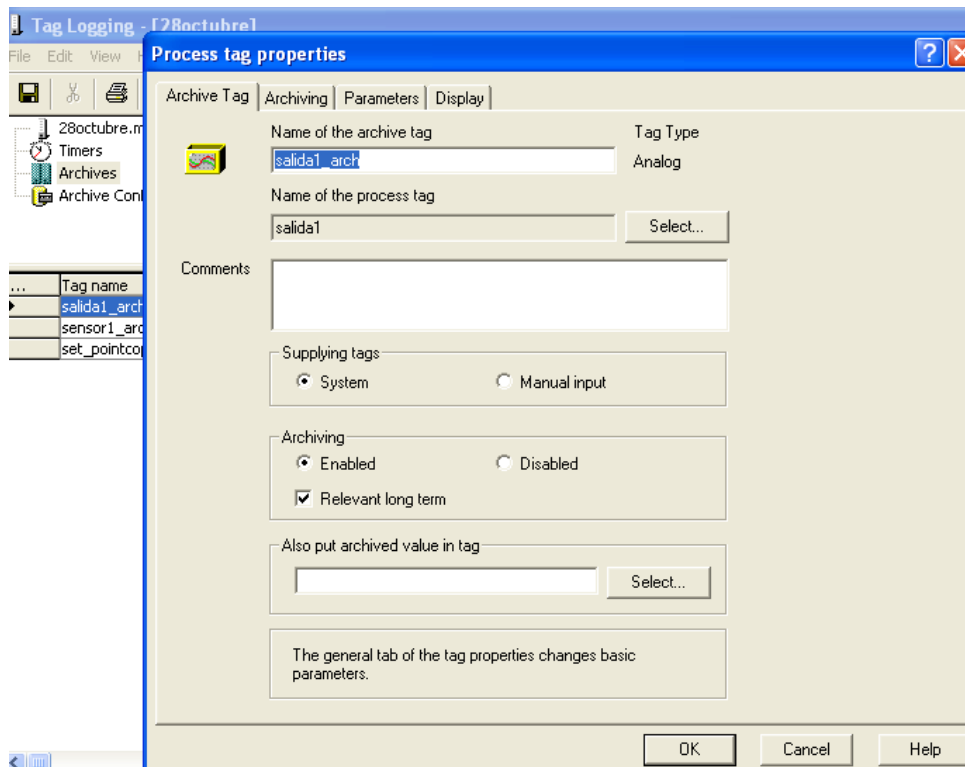
A26 Tag Logging: Propiedades del archivo de valor de proceso, información general



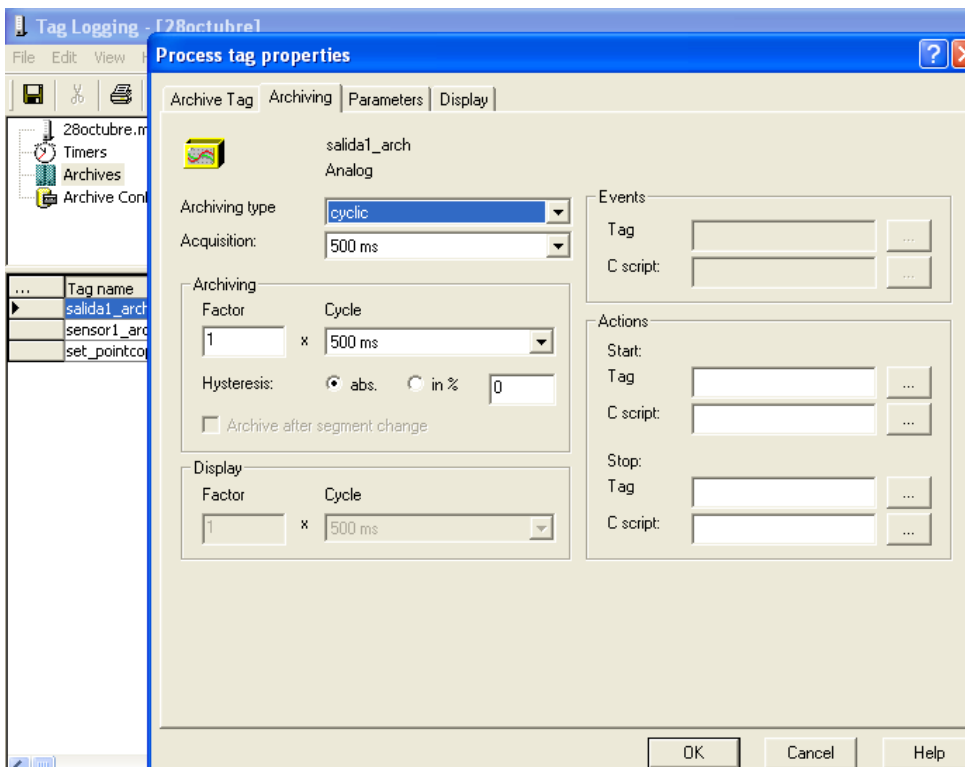
A27 Tag Logging: Propiedades del archivo de valor de proceso, ubicación de memoria



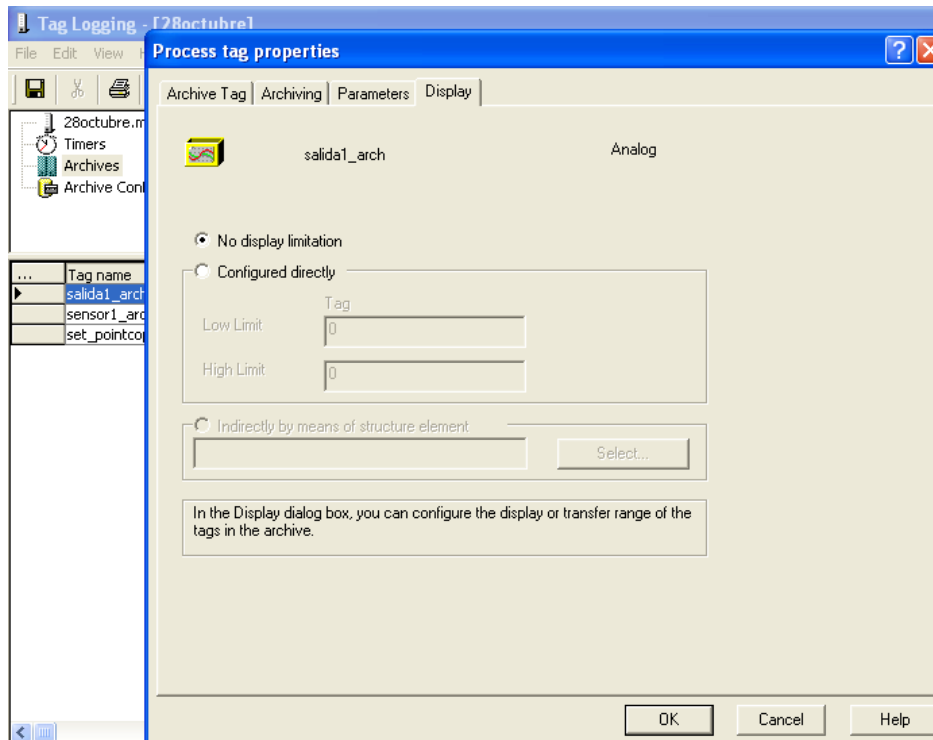
A28 Propiedades de la variable de proceso: variable de archivo salida1



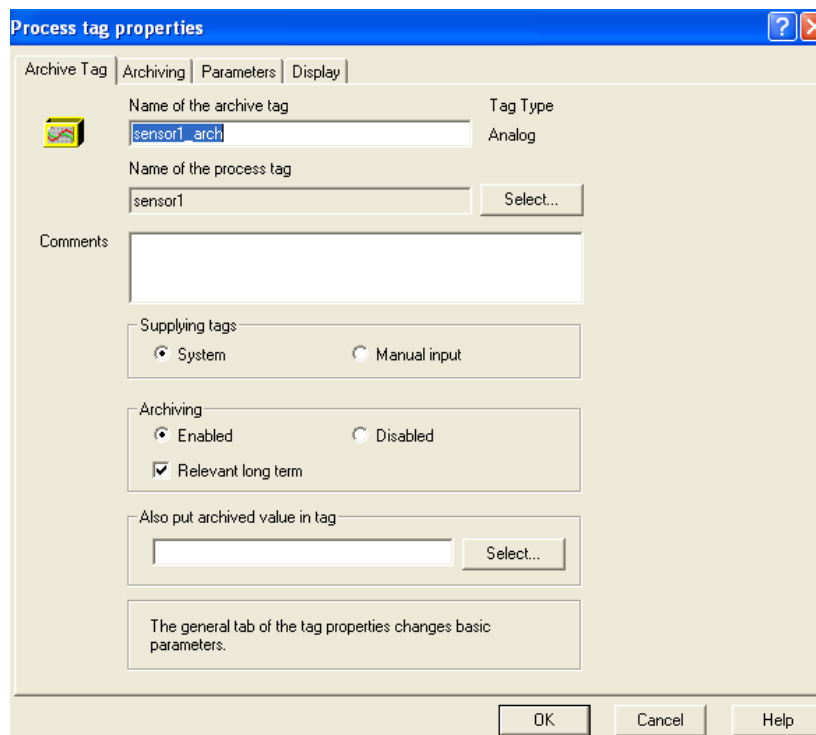
A29 Propiedades de la variable de proceso: archivo salida1



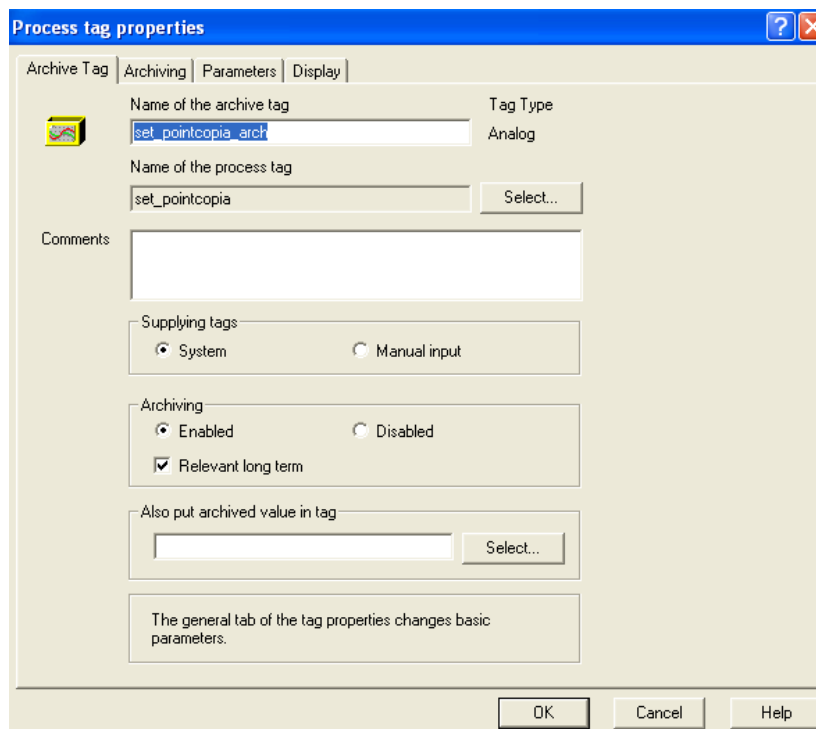
A30 Propiedades de la variable de proceso: display salida1



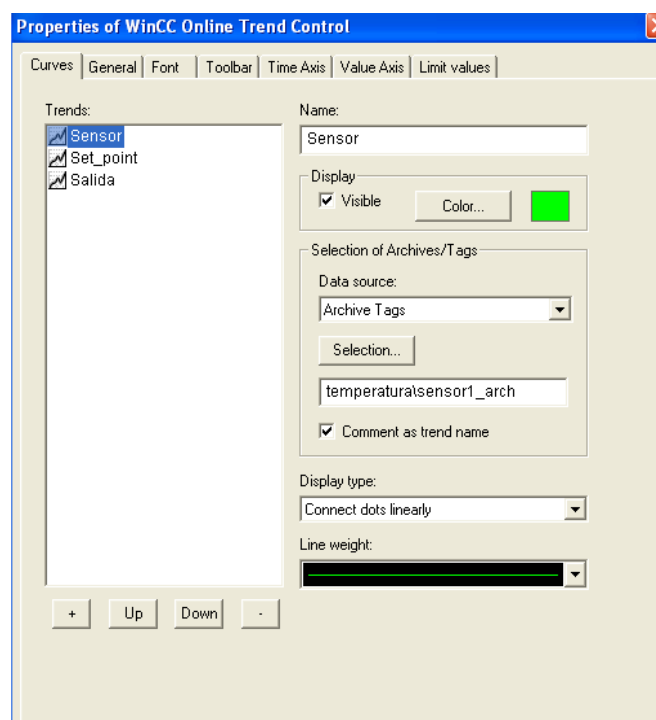
A31 Propiedades de la variable de proceso: variable de archivo sensor1



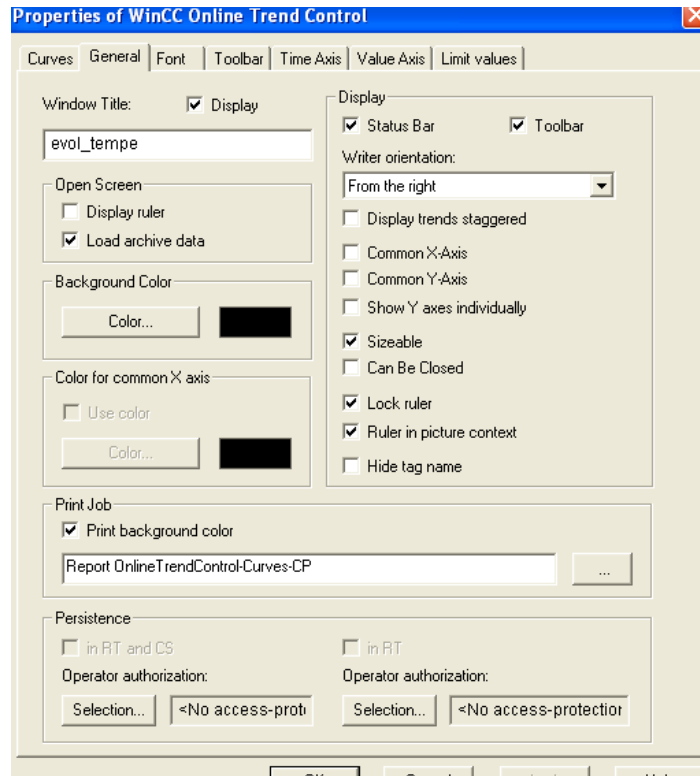
A32 Propiedades de la variable de proceso: variable de archivo set_pointcopia



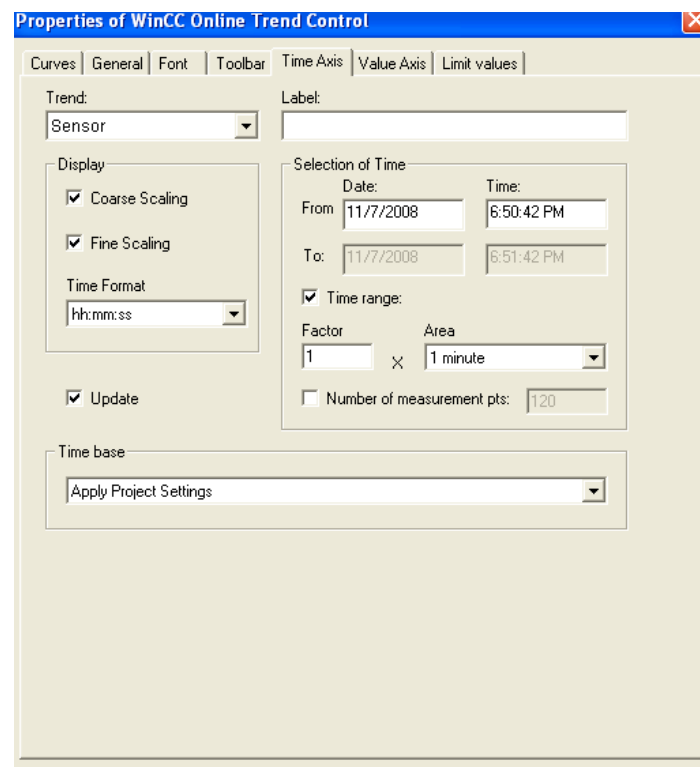
A33 Graphics designer, propiedades de WinCC Online Trend Control sensor



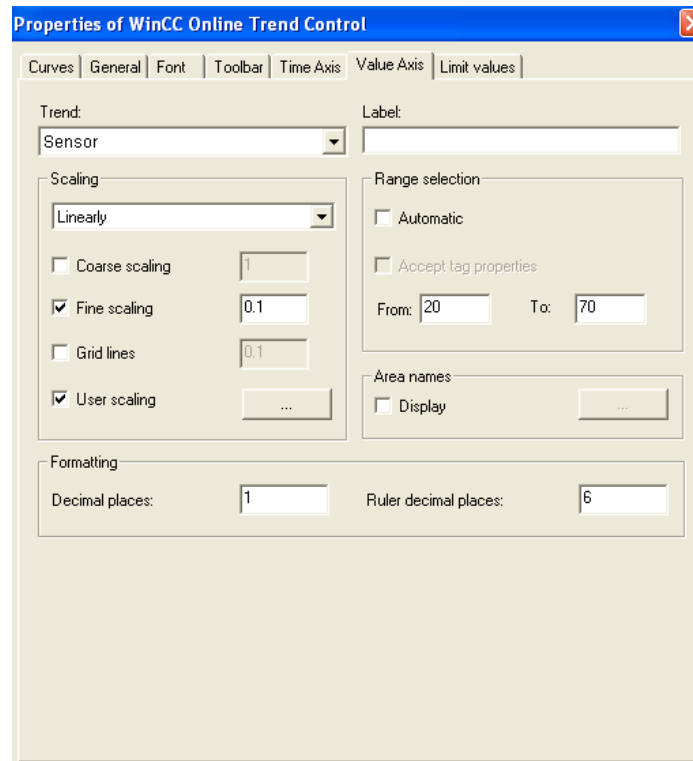
A34 Propiedades de WinCC Online Trend Control: general sensor



A35 Propiedades de WinCC Online Trend Control: time axis sensor



A36 Propiedades de WinCC Online Trend Control: value axis sensor



A37 Vista del Tag Simulation simulando la aplicación

The screenshot shows the WinCC Runtime interface for 'FUZZYCONTROL++'. It features three vertical gauges: 'Sensor' (green, 50.00), 'Set Point' (yellow, 50), and 'Salida' (red, 2.978). A 'Trigger' indicator shows '1'. Below the gauges, the text 'Ingrese Set Point' is visible. An inset window titled 'Simulation WESPE-BB2E1CE9DCWinCC60_Project_2...' shows the 'Tag Simulation for WinCC' properties for the tag 'sensor_simula'. The 'Slider' tab is active, showing a range from 20 to 70 with a current value of 50. The 'active' checkbox is checked. Below the simulation window, a table displays the current values and tag connections for the simulation.

Trend	Tag Connection	Value	Date/Time
Sensor	temperatura\sensor1_arch	4.778e+001	8 10.22.30
Set_point	temperatura\set_pointcopia_arch	5.000e+001	8 10.22.30
Salida	temperatura\salida1_arch	2.978e+000	8 10.22.30

REFERENCIAS BIBLIOGRÁFICAS

- ❖ FSI Conjuntos Difusos y Lógica Difusa, www.sci2s.ugr.es/docencia/doctoSCTID/FSIConjuntos%20difusos-Introduccion.pdf, 2008-06-02.
- ❖ López, Bruno, Lógica Difusa, <http://www.mitecnologico.com/Main/EIProcesoDeRazonamientoSegunLaLogica>, 2005-08-26, 2008-06-02.
- ❖ Gutiérrez, Julio, Laudatio, http://www.upm.es/canalUPM/notasprensa/Discurso_Julio_Gutierrez_E.pdf, 2007-01-29, 2008-06-02.
- ❖ Control Difuso, <http://prof.usb.ve/montbrun/ps2320/fuzzy/fuzzy.html>, 2008-06-10.
- ❖ Gómez, Galindo, Conjuntos y Sistemas Difusos, www.lcc.uma.es/~ppgg/FSS/FSS1.pdf, 2008-06-10.
- ❖ Corzo, Yuliana, La Lógica, <http://www.monografias.com/trabajos6/lalo/lalo.shtml>, 2008-06-11.
- ❖ Montejo, Miguel, Breve curso de lógica difusa y control difuso, www.redeya.com/electronica/cursos/fuzzy/fuzzy.html, 2008-06-10.
- ❖ Universidad Nacional de Cuyo Facultad de Ingeniería, Instrumentación y control automático, <http://fing.uncu.edu.ar/catedras/archivos/control/controldifuso.pdf>, 2008-06-11.
- ❖ Siemens, FuzzyControl++ User's Manual, http://www.industry.siemens.de/industrial-services/EN/SOLUTION_SERVICES/SIMATIC_ADDONS/fuzzy_download.htm, 2003, 2008-06-11.
- ❖ Marín Roque, Tareas Inteligentes Control Fuzzy, <http://perseo.dif.um.es/~roque/ti/Material/Tema3.ppt>, 2006-06-15.

- ❖ Controladores Lógicos Programables,
<http://www.profesores.frc.utn.edu.ar/industrial/sistemasinteligentes/UT3/plc/PLC.html>, 2008-06-16.
- ❖ Siemens, SIMATIC S7-300 Mid-range PLC,
<http://www2.sea.siemens.com/Products/Automation/Programmable-Controllers/SIMATIC-S7-300/?languagecode=en>, 2008-02-08, 2008-07-01.
- ❖ New Modules Supported from STEP 7 V5.4 SP1 Onwards,
http://www.siemens.co.kr/Automation/DATA/FAQ/PLC/200611/New_Modules_in_STEP7_V54_SP1.pdf, 2006-07, 2008-07-15.
- ❖ Siemens,
<http://support.automation.siemens.com/WW/llisapi.dll/19860749?func=ll&objid=19860749&objAction=csView&nodeid0=15326996&lang=es&siteid=cseus&aktprim=0&extranet=standard&viewreg=WW&load=content>, 2004-11-22, 2008-07-01.
- ❖ Siemens, SIMATIC S7-300 I/O Modules,
http://www.automation.siemens.com/simatic/controller/html_76/produkte/simatic-s7-300-io.htm, 2008-07-04.
- ❖ Siemens, SIMATIC fail-safe controllers,
http://www.siemens.dk/ccmi/bu/ad/download/pdf/service_support/information/brochure/20070119/s7300f_e_12-2005.pdf, 2005-12, 2008-07-15.
- ❖ Siemens, Step 7 Professional Programming and configuring according to IEC 61131-3,
<http://www2.sea.siemens.com/NR/rdonlyres/D50EFC60-1EA6-43AF-A741-26402740B8C4/0/STEP7PROANDBASISPRODUCTBRIEF.pdf>, 2003-04, 2008-07-15.
- ❖ Siemens, FuzzyControl++ for efficient automation of technical processes,
http://info.industry.siemens.com/Downloads/start_download.asp?lang=e&file=PB%5F567%5FFuzzyControl%2B%2B%5Fen%2Epdf&did=567&lid=1&kid=10, 2007, 2008-07-15.
- ❖ Siemens, Simatic HMI WinCC Visualización de proceso y plataforma para IT y Business Integration,
http://www.automation.siemens.com/salesmaterial-as/brochure/es/sb_wincc_v6_s.pdf, 2003, 2008-07-18.

- ❖ Capítulo VII Sensores, <http://www.unet.edu.ve/~ielectro/sensores.pdf>, página 111, 2008-07-20.

Referencias bibliográficas de apoyo:

- ❖ Siemens, Software Simatic: Herramientas para configurar y programar controladores SIMATIC, http://www.automation.siemens.com/salesmaterial-as/brochure/es/brochure_simatic-industrial-software_es.pdf, 2008-04, 2008-06-20.
- ❖ Siemens, Simatic HMI WinCC, http://www.automation.siemens.com/salesmaterial-as/brochure/es/sb_wincc_v6_s.pdf, 2003, 2008-07-01.
- ❖ Siemens, Manual de formación para soluciones generales en automatización Totally Integrated Automation (TIA) MÓDULO B2 Procesamiento de valores analógicos, http://www.automation.siemens.com/fea/ftp/module_sp/b02/b02_analog.pdf, 2002-02, 2008-07-15.
- ❖ Siemens, Profinet IO Getting Started: Collection, <http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&objId=19290251&nodeid0=10805255&load=content&lang=es&siteid=cseus&aktprim=0&objaction=csview&extranet=standard&viewreg=WW>, 2006-12, 2008-07-20.
- ❖ Siemens, FAQ about S7 communication via IE, http://setaut.com/Guideline_S7_IE_e.pdf, 2007, 2008-07-28.
- ❖ Siemens, Esquema de contactos (KOP) para S7-300 y S7-400 Manual de referencia, <http://www.elai.upm.es/spain/Asignaturas/Automatizacion/archivos/S7-KOP.pdf>, 2004-01, 2008-07-30.
- ❖ Siemens, Programar con STEP 7 V5.1, <http://www.elai.upm.es/spain/Asignaturas/Automatizacion/archivos/S7-Prog.pdf>, 2004-01, 2008-08-01.
- ❖ Siemens, Sistema de automatización S7-300 Getting Started Collection, <http://support.automation.siemens.com/WW/llisapi.dll?func=cslib.csinfo&obj>

-
- [Id=19860749&nodeid0=15326996&load=content&lang=es&siteid=cseus&aktprim=0&objaction=csview&extranet=standard&viewreg=WW](http://www.esi2.us.es/~gordillo/labpc/ScadaWinCC.pdf), 2008-06, 2008-08-10.
- ❖ Introducción al SCADA WinCC , <http://www.esi2.us.es/~gordillo/labpc/ScadaWinCC.pdf>, 2008-08-20.
 - ❖ Siemens, SIMATIC HMI WinCC V6 Getting Started Manual, http://www.pacontrol.com/siemens-manuals/Simatic-HMI-WinCC_Basics.pdf, 2007-03, 2008-08-30.

Fecha de entrega:

Srta. Andrea Orellana
Autor

Ing. Víctor Proaño
Coordinador de Carrera

