UNIVERSIDAD POLITÉCNICA SALESIANA ECUADOR

POSGRADOS

MAESTRÍA EN

ELECTRÓNICA Y AUTOMATIZACIÓN

RPC-SO-19-No.277-2018

OPCIÓN DE TITULACIÓN:

PROYECTOS DE DESARROLLO

TEMA:

DESARROLLO DE UN CONTROLADOR INTELIGENTE DIFUSO PARA EL CONTROL DE TEMPERATURA DE MOLDES DE PLÁSTICO CON SISTEMA DE COLADA CALIENTE, MONITOREADA DE FORMA REMOTA DESDE LA NUBE CON HERRAMIENTAS IOT

AUTOR:

DIEGO HERNANDO HIDALGO LLUMIQUINGA

DIRECTOR:

CARLOS GERMÁN PILLAJO ANGOS

QUITO - ECUADOR
2021

COHORTE
2018 - 2020

Autor:

Diego Hernando Hidalgo Llumiquinga.
Ingeniero Eléctrico.
Candidato a Magíster en Electrónica y Automatización, Mención en Informática Industrial por la Universidad Politécnica Salesiana - Sede Quito.
diego_hidalgo81@hotmail.com

Dirigido por:

Carlos Germán Pillajo Angos.
Ingeniero en Electrónica y Control.
Magister en Ingeniería.
cpillajo@ups.edu.ec

DESARROLLO DE UN CONTROLADOR INTELIGENTE DIFUSO PARA EL CONTROL DE TEMPERATURA DE MOLDES DE PLÁSTICO CON SISTEMA DE COLADA CALIENTE, MONITOREADA DE FORMA REMOTA DESDE LA NUBE CON HERRAMIENTAS IOT

Índice general

Índice de Figuras

Índice de Tablas

Resumen

El presente trabajo se realizó con el fin de establecer y diseñar nuevas alternativas para sistemas de Industria 4.0 que permitan el control, monitoreo y análisis de datos de forma remota, usando equipos existentes en el mercado industrial, con una aplicación web de desarrollo propio. Se realiza el diseño de un controlador difuso que es implementado en un PLC Siemens S7-1200, para el control de dos resistencias eléctricas que vienen a ser las cavidades del molde de colada caliente para la inyección de plástico, estas resistencias son controladas mediante relés de estado sólido cuyos pulsos de control los realiza el PLC. La conexión a la nube se implementa en un equipo Siemens Simatic IoT2040, que se lo usa como gateway para la conexión con un broker MQTT que es instalado en una instancia virtual en la plataforma Google Cloud, para permitir la publicación o suscripción de datos, que desde una aplicación web desarrollada en el presente trabajo permite la visualización de las variables de proceso, históricos de datos, control de puntos de consigna y estado del sistema. Finalmente se construyó una planta y el tablero de control para realizar las pruebas que permitieron concluir, que la conectividad de dispositivos móviles y computadores desde la aplicación web en un dominio público contratado, con el sistema de control de temperatura, presenta un retardo poco significativo en tiempo real y adicionalmente, el controlador difuso para el sistema de temperatura mejora la respuesta comparada con el clásico controlador PID, por lo que el proyecto desarrollado es satisfactorio y permite una escalabilidad a sistemas de inyección de plástico a un costo accesible y con una solución personalizada.

Palabras clave: controlador difuso, Internet de las cosas, aplicación web, controlador de temperatura, molde de inyección de plástico.

Abstract

This work was carried out in order to establish and design new alternatives for Industry 4.0 systems that allow the control, monitoring and data analysis remotely, using existing equipment in the industrial market, with a self-developed web application. The design of a fuzzy controller is implemented in a Siemens S7-1200 PLC, for the control of two electrical resistors that come to be the cavities of the hot casting mold for plastic injection, these resistors are controlled by solid state relays whose control pulses are performed by the PLC. The connection to the cloud is implemented in a Siemens Simatic IoT2040 device, which is used as a gateway for the connection with a MQTT broker that is installed in a virtual instance in the Google Cloud platform, to allow the publication or subscription of data, which from a web application developed in this work allows the visualization of process variables, historical data, setpoint and system status. Finally, a plant and the control panel were built to perform the tests that allowed concluding that the connectivity of mobile devices and computers from the web application in a public contracted domain, with the temperature control system, presents an insignificant delay in real time and additionally, the fuzzy controller for the temperature system improves the response compared to the classic PID controller, so the developed project is satisfactory and allows scalability to plastic injection systems at an affordable cost and with a customized solution.

Keywords: fuzzy controller, IoT, web app, temperature controller, plastic injection mold

Capítulo 1

Introducción

Desde los años 80 los sistemas inteligentes difusos han ido desplazando poco a poco a tecnologías convencionales de control en aplicaciones complejas y sistemas de ingeniería, especialmente en los procesos de control. La lógica difusa surge del deseo de explicar sistemas complejos mediante descripciones lingüísticas.

El sistema de control de temperatura a nivel industrial tiene muchas características, como un retraso prolongado, una gran inercia, respuestas no lineales, y muchos factores de perturbación, lo que conlleva a muchas dificultades para configurar un modelo matemático preciso para él. El sistema difuso es una nueva tecnología de control que proviene de la combinación de la teoría de control y la teoría de conjuntos difusos, aprovechando la experiencia operativa y el conocimiento de los expertos en el campo, mediante la selección de reglas apropiadas de control difuso y así garantizando un funcionamiento estable.

1.1. Descripción general del problema

Hoy en día es esencial apoyar la automatización de las plantas industriales enfocadas en el IOT (Internet de las Cosas), debido al desarrollo tecnológico e innovación, sin embargo, este es un punto que en el país aún no se considera como una oportunidad para mejorar la rentabilidad de las empresas. La industria se ha visto forzada a reconfigurar sus procesos pero más no lo enfocan a sistemas de industria 4.0 y manufactura inteligente, que ofrece beneficios como mejora de la gestión, el seguimiento de los activos y de los productos, aumento de la cantidad de datos de información, los mismos que permiten una optimización de equipos o procesos

Adicionalmente, la implementación de controladores basados en lógica difusa tienen como principio utilizar estrategias cualitativas de control y su capacidad de implementar un comportamiento de control flexible, que permiten introducir un enfoque humano al diseño de este tipo de controladores sin necesidad de un modelo matemático, es por eso que representan una alternativa efectiva y razonable a las técnicas de control clásicas en lo que a sistemas complejos se refiere.

1.2. Objetivos

1.2.1. Objetivo general

Desarrollar un Controlador Inteligente Difuso, mediante la utilización de un PLC Siemens S7 1200 y un dispositivo IoT, para aplicarlo en el control de Temperatura de Moldes de Plástico con Sistema de Colada Caliente, monitoreada de forma remota desde la nube.

1.2.2. Objetivos específicos

- Determinar las características de diseño de los controladores inteligentes difusos, mediante el estudio de reglas de control para el análisis del comportamiento de los procesos de temperatura.
- Diseñar el controlador Inteligente difuso utilizando un PLC S7 1200 a través del lenguaje de Programación SCL, para el control de temperatura de Moldes de Plástico con Sistema de Colada Caliente.
- Desarrollar una plataforma mediante un dispositivo IOT para el monitoreo de forma remota del proceso de Temperatura.

- Comparar el desempeño del controlador Fuzzy versus un Controlador PID realizando pruebas de funcionamiento con los algoritmos de control, para evaluar la eficiencia de rendimiento de las técnicas inteligentes y convencionales de control.

1.3. Contribuciones

El presente trabajo promueve el uso de sistemas de internet de las cosas a un costo accesible enfocado a industrias productoras de plásticos así como el uso de algoritmos de control nuevos y eficientes que hacen al proceso más rentable y fiable.

1.4. Organización del manuscrito

El manuscrito se compone de 4 capítulos:

- Introducción: Se compone de los objetivos, descripción del problema y se mencionan las contribuciones del trabajo.
- Marco Teórico: Comprende el estado del arte y formulación del problema.
- Metodología: Comprende el diseño integral y específico de la solución desarrollada.
- Resultados y conclusiones: Comprende el análisis de los resultados obtenidos, adicionalmente, se plantean conclusiones y recomendaciones del proyecto.

Capítulo 2

Marco Teórico

El capítulo 2 abarca el estado del arte, junto con el análisis de los algoritmos de control difusos y su implementación en la industria así como también de las herramientas de Industria 4.0 que permiten un monitoreo remoto de cualquier proceso. Finalmente se formula el problema.

2.1. Estado del Arte

La Industria 4.0 es el nombre principal dada a la cuarta revolución industrial introducida por la era digital como lo menciona el autor [Carvalho et al., 2020]. Aunque muchas de las filosofías y tecnologías de la Industria 4.0, como el internet de las cosas (IoT), sistemas cyber físicos e inteligencia artificial están en uso, muchas empresas e industrias carecen de las nuevas tecnologías ofrecidas por la apliación del internet, que provee dispositivos con gran capacidad de conectividad, inter operatibilidad, visibilidad y capacidad de inteligencia [Kalsoom et al., 2020].

Muchas de las fábricas actuales se encuentran constantemente enfrentando problemas en la adopción de manufactura más limpia, por lo que cada vez es más importante integrar tecnología avanzada que pueda proveer información en tiempo real [Pillajo and Hincapie, 2018] para ser procesada y analizada de forma predictiva y cognitiva [Rajput and Singh, 2020].

[Rajput and Singh, 2020] establece que la literatura existente de la Industria 4.0 combina inteligencia artificial, sensores inteligentes, big data, cloud computing y es considerada como el pilar de infraestructuras de redes inteligentes.

[Petrasch and Hentschke, 2016] establece que el internet de las cosas o mejor conocido por sus siglas en inglés IoT, permite que los dispositivos estén conectados a una red y típicamente a una aplicación en la nube, creando oportunidades para nuevos servicios a través de la integración del mundo físico y digital, este avance tecnológico abarca cualquier elemento, por ejemplo una banda transportadora.

Los autores [Chaves et al., 2020] desarrollan un sistema con lógica difusa con funciones adaptativas de membresía que permite producir objetos de buena calidad en procesos de inyección, encontrando variables de proceso óptimas en un número pequeño de ciclos de inyección. Adicionalmente, realizan un sistema fuzzy híbrido que combina las ventajas del modelo Mamdani y del modelo Tsukamoto, logrando una mejor aproximación al modelo matemático para procesos de inyección, donde no se requiere una discretización de la entrada.

Un estudio ha podido aplicar lógica difusa en el proceso de soldadura de un sistema de inyección por molde, este control está basado en la experiencia de un experto como regla de inferencia, todo el proceso del control difuso lo divide en cuatro secciones, interfaz de fusificación, reglas de producción, lógica de toma de decisiones y la interfaz de desfusificación [Chen et al., 2006].

Los autores [Chen et al., 2009] realizan un análisis de un sistema de

inyección de plástico simple y se establece que se debe tener una temperatura constante en todo el molde de inyección, con el fin de minimizar la deformación y otros defectos causados por fluctuaciones de temperatura. Esto demuestra la importancia de controladores robustos en procesos controlados de temperatura.

Se presenta un estudio cuantitativo de 200 empresas italianas donde el 37.4 % no muestran interés en adoptar tecnologías de Industria 4.0, un 25.6 % de las empresas encuestadas están considerando esta opción, el 18.9 % están ya ejecutando proyectos de Industria 4.0 y el 18.1 % ya han culminado la implementación de sistemas inteligentes [Chiarini et al., 2020], los autores de igual manera establecen que las empresas que han adoptado sistemas de Industria 4.0 han logrado una mejora significativa en los resultados de varias áreas de desempeño, así como también las condiciones de operación que permiten la persecución de diferentes tipos de estrategias.

Los Autores [Devillez et al., 2015] desarrollan un método difuso de clasificación para el diagnóstico de un proceso de inyección de plástico por molde, se basan en una máquina Sandretto de 60 toneladas con un volúmen máximo inyectable de 192 cm^3, el material usado durante el estudio es poliestireno, el estudio concluye que se puede aplicar los métodos de clasificación o reconocimiento de patrones para controlar un proceso de inyección de plástico.

Se diseña un sistema de vigilancia y control utilizando el internet de las cosas aplicado al funcionamiento de dispositivos eléctricos usando un módulo ESP y una Raspberry Pi con conexión a internet, la señal de los sensores es transmitida a un NodeMCU que luego transmite la señal de forma inalámbrica a un terminal de control maestro, el que realiza funciones de control, adicionalmente, el dispositivo maestro está conectado a la nube mediante una aplicación móvil desarrollada con NodeRED [Gupta and Johari, 2019].

Las fábricas tradicionales carecen de las capacidades que permite monitorear y controlar manufactura automatizada y compleja, con el fin de permitir producción eficiente de productos personalizados, [Kalsoom et al., 2020] estas fábricas tradicionales tienen aplicaciones segregadas que operan de forma aislada con menos integración del sistema de producción, menor ciclo de vida del producto y menor cadena de valor, sin embargo, actualmente, el concepto de una fábrica inteligente se muestra en la Figura 2.1. Una fábrica inteligente es una actualización de los sistemas convencionales a un sistema enlazado y flexible que constituye un flujo de datos continuo a través de operaciones conectadas y sistemas de producción que pueden aprender y adaptarse a demandas cambiantes.

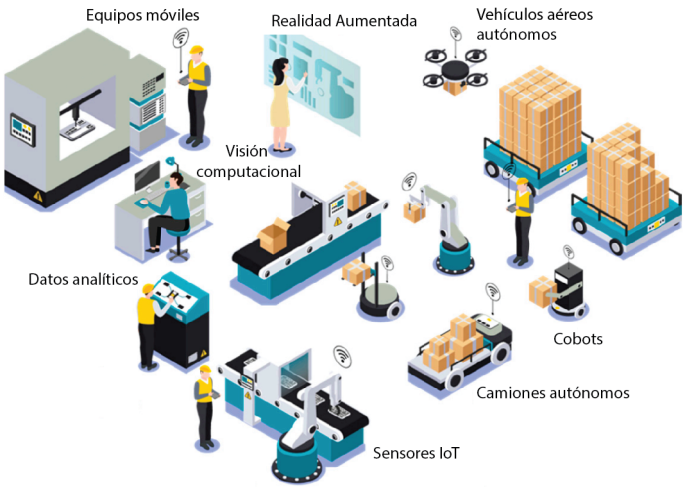


Figura 2.1: Concepto de una fábrica inteligente [Kalsoom et al., 2020]

[Ravi and Balakrishnan, 2010] desarrolla tres tipos de controladores para el control de temperatura en procesos de extrusión de plástico, el típico controlador PID, el controlador Fuzzy y un controlador ANFIS, el proceso es no lineal y los tiempos del sistema controlado se presentan en la Tabla 2.1.

Tabla 2.1: Características de comportamiento de tiempo de PID y Fuzzy [Ravi and Balakrishnan, 2010]

Especificación de tiempo	***PID***	***Fuzzy***
Tiempo de demora Td	170seg	25seg
Tiempo de subida Tr	250seg	80seg
Peak Time Tp	400seg	100seg
Tiempo de establecimiento Ts	1900seg	1800seg
Peak overshoot %	21	0

[Wang et al., 2010] presenta una solución para enfrentar la necesidad de tener productos de alta precisión con eficiencia energética en máquinas inyectoras de plástico, se desarrolla de igual manera un controlador Fuzzy-PID para el control de presión que afecta directamente sobre la calidad de las piezas finales, y concluyen que este controlador es más efectivo y robusto, la Figura 2.2 muestra la configuración del sistema y la Figura 2.3

muestra los bloques de configuración del controlador del sistema.

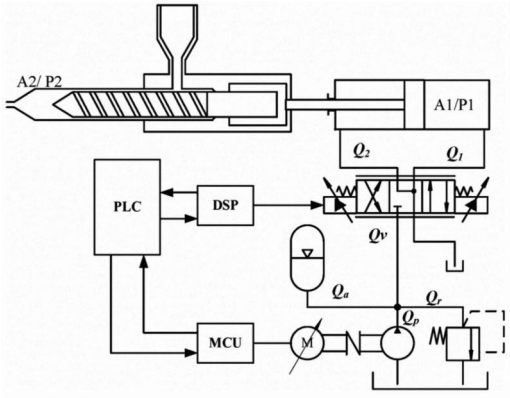


Figura 2.2: Configuración de una máquina de inyección de plástico de eficiencia energética [Wang et al., 2010]

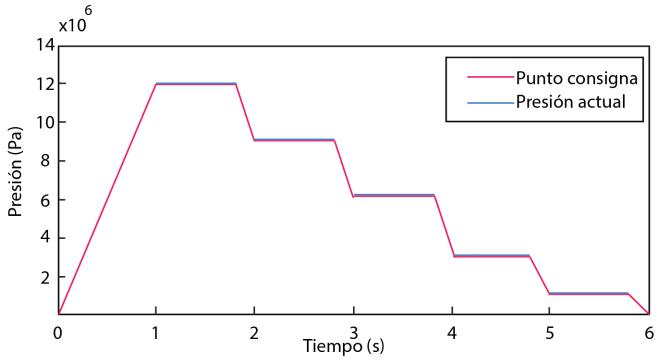


Figura 2.3: Respuesta del controlador Fuzzy-PID [Wang et al., 2010]

2.2. Formulación del Problema

El sistema de colada caliente es la forma más eficiente de producir piezas de plástico inyectado, y consiste en un manifold (distribuidor) y un juego de boquillas con resistencias eléctricas, que mantienen la resina plástica fundida a una temperatura uniforme desde la salida de la maquina inyectora hasta que entra en cada una de las cavidades del molde, esto permite que

el flujo de plástico sea continuo, provocando un efecto representativo de ciclos rápidos, mayor producción en menor tiempo y un mejor acabado de las piezas inyectadas, por lo que el control adecuado de la temperatura es un punto crítico, razón por la cual su funcionamiento debe ser apoyado con un controlador altamente eficiente para mantener la temperatura estable.

La utilización de técnicas de control convencional en procesos de la industria Plástica para el control de Temperatura es el algoritmo PID, en la mayoría de casos resulta ser la más adecuada y utilizada a nivel general hoy en día, sin embargo entre mayor es la precisión requerida por el sistema, el ajuste de este tipo de control es más complicado por lo que se requiere de más recursos computacionales para el desarrollo de algoritmos inteligentes. Razón por la cual se realizará la propuesta en este proyecto, en el desarrollo de un controlador basado en reglas difusas siendo un procedimiento más sencillo, que la ardua tarea de resolver ecuaciones diferenciales no lineales usadas tradicionalmente para modelar sistemas de control.

La implementación del algoritmo inteligente con lógica difusa para controlar la temperatura de un Molde de Plástico con Sistema de Colada Caliente, será implementado en un PLC S7 1200, lo que le hace innovador al proyecto en estudio, ya que se pretende demostrar que se puede desarrollar este tipo de algoritmos en un PLC de gama media, con la utilización del lenguaje de programación SCL (Lenguaje de Control Estructurado), dicho lenguaje de alto nivel está orientado a Pascal que posibilita una programación estructurada, reduciendo costos significativamente al no utilizar PLCs de gama alta como son los s7-1500, s7-300 o s7-400.

También se desea dar un enfoque de Industria 4.0 al sistema desarrollado, aplicando el concepto del IoT que puede ser visto como una combinación de sensores y actuadores que son capaces de proporcionar y recibir información digitalizada y colocarla en redes bidireccionales capaces de transmitir los datos a la nube y ser monitoreados de manera remota desde cualquier ubicación del mundo, por lo que ¿es posible desarrollar un sistema de control difuso con monitoreo y visualización de datos de forma remota usando herramientas de internet de las cosas?.

Capítulo 3

Diseño

Este capítulo contempla el diseño de la solución de monitoreo remoto, del algoritmo difuso y el diseño de la planta de pruebas en base a la metodología seleccionada.

3.1. Metodología

Las metodologías histórico descriptiva e investigación científica son utilizadas en el capítulo 1 y 2 donde se realiza una recolección de información y análisis de los sistemas de control difuso actuales.

Se comienza con el desarrollo del sistema inteligente en base a la metodología experimental, en esta etapa se diseña el controlador inteligente difuso utilizando un PLC S7 1200, a través de la utilización del lenguaje de Programación SCL, que brinda el integrador del Tía Portal de Siemens. Posteriormente, se desarrolla la plataforma web, que comprende la conectividad del equipo a la nube para realizar un monitoreo y control remoto por parte del usuario.

Finalmente, se realiza una evaluación de la solución integral desarrollada usando la metodología deductiva que permite realizar pruebas de validación de prototipo y pruebas de satisfacción de usuario al interactuar con la interfaz desarrollada.

3.2. Diseño general

Se propone un diseño general del sistema, desde el cual se inicia el desarrollo específico de cada componente. La Figura 3.1

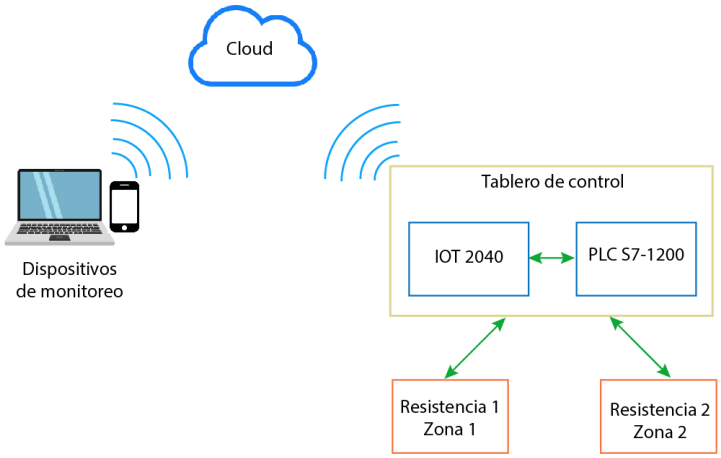


Figura 3.1: Arquitectura general del sistema

Básicamente, el medio de conexión al internet que dispone el PLC

S7-1200 es el módulo IoT 2040 de Siemens, mismo que permite tener una solución confiable y de plataforma abierta para la adquisición, transferencia y procesamiento de datos, funciona como un Gateway.

El PLC S7-1200 es el encargado de receptar y enviar la información desde y hacia el Gateway con el fin de procesar señales de puntos de consigna para los controladores a implementarse, la salida del S7-1200 irá enlazada a un disparador de alta frecuencia que controlará dos resistencias eléctricas de calentamiento de los moldes en cuestión y se retro alimentará por unas termocuplas acopladas a cada resistencia, que podrán tener diferentes puntos de consigna.

3.3. Diseño Eléctrico

El diseño eléctrico contiene el desarrollo de los planos eléctricos de control para que el sistema pueda operar sin problemas. Adicionalmente se diseña el tablero de control, mismo que consta de protecciones eléctricas, fuentes de poder, controlador lógico programable, módulo de expansión para termocuplas, relés de estado sólido y el módulo IOT2040.

Se establece que todo el sistema de control debe estar en un solo gabinete metálico, mismo que es ensamblado a partir del diseño de la presente sección, el tablero de control debe tener conectores exteriores para facilitar la conexión con periféricos tal como las termocuplas y las resistencias eléctricas. Todos los elementos y componentes eléctricos son dimensionados en base a la aplicación en cuestión.

3.3.1. Selección de elementos

El PLC a usar es un PLC S7-1200 mostrado en la Figura 3.2 con las características presentadas en la Tabla 3.1, este PLC es seleccionado en base al criterio de disponibilidad en la mayor parte de inyectoras de plásticos comercializadas e implementadas en el país. Los principales requisitos se presentan a continuación:

- Salidas a transistor para control de relés de estado sólido que comandarán las resistencias eléctricas.
- Interfaz Profinet
- Frecuencia de conmutación de 20KHz
- Posibilidad de agregar módulos de expansión

Figura 3.2: PLC S7-1200 1214C DC/DC/DC

Tabla 3.1: Características técnicas del PLC S7-120.

Característica	***Valor***
Serie	6ES7 214-1AG40-0XB0
Rango de tensión de alimentación	20.4-28.8 VDC
Tensión nominal de alimentación	24VDC
Corriente de entrada máxima	1.5 Amperios
Intensidad de salida	1600mA
Tipo de salida	Transistor
Número de módulos adicionales permitidos	8 de señales
Frecuencia de conmutación máxima	100 KHz
Tipo de interfaz	Profinet

El gateway seleccionado para la conectividad a la nube es el equipo Simatic IOT2040 de Siemens, que permite la implementación de diferentes protocolos desde Modbus RTU, OPC hasta protocolos en la nube como MQTT, este dispositivo posee las características mostradas en la Tabla 3.2 y se presenta en la Figura 3.3

Dado que el PLC y el IOT2040 necesitan una fuente de alimentación de 24VDC se opta por seleccionar una fuente de la marca Siemens que proporciona el voltaje requerido a 2.5 Amperios, corriente suficiente para alimentar el PLC y el Simatic IOT2040, esta fuente se muestra en la Figura 3.4.

El sistema comanda 2 resistencias eléctricas, para realizar un control AC

Tabla 3.2: Características técnicas del Simatic IOT2040

Característica	***Valor***
Rango de tensión de alimentación	9-36 VDC
Tensión nominal de alimentación	24VDC
Interfaces	2xLAN RJ45, 1USB, 2COM
Procesador	Intel Quark X1020, 400 MHz
Temperatura operación	0-50°C
Número de módulos adicionales permitidos	8 de señales
Resistencia	Vibraciones, EMC, golpes, ambientes nocivos
Tamaño	144x90x53mm

Figura 3.3: Simatic IOT2040

Figura 3.4: Fuente de poder 24VDC

Figura 3.5: Relé de estado sólido

de las mismas, se implementan relés de estado sólido para cargas AC, estos son interruptores electrónicos con función de aislamiento, están compuestos de un dispositivo semiconductor activo y un componente pasivo, usan un foto electrón y tecnología micro electrónica para realizar un acoplamiento y aislamiento entre la entrada y la salida [Zhang and Zhang, 2011], permitiendo tener un funcionamiento similar a los relés comunes pero con mejor fiabilidad por la ausencia de contactos y partes mecánicas móviles. Los relés seleccionados se presentan en la Figura 3.5, soportan hasta 25Amperios, por lo que son suficientes para la potencia de la resistencia seleccionada a continuación.

Por otro lado, las resistencias eléctricas a implementarse tienen una longitud de 52mm, con diámetro de 25mm, una potencia de 350W a 110VAC y una densidad de potencia de 5.4 W/cm^2, son resistencias helicoidales que incorporan internamente un termopar tipo J, como se muestra en la Figura 3.6, disponen una protección de fibra de vidrio de 10cm.

Figura 3.6: Resistencia Helicoidal

El elemento sensor para la retroalimentación del controlador a implementar posteriormente, es el termopar de cada resistencia, por lo que se requiere que el PLC disponga de un módulo de expansión para termocuplas. El módulo usado es un SM 1231 TC, con la referencia 6ES7231-5QD30-0XB0,

como se presenta en la Figura 3.7, permite la conexión de 4 termocuplas tipo J, K, T, E, R, S, N, C, TXK/XK(L); con un rango de voltaje entre: ±80 mV con una resolución de 15 bits más el signo.

Figura 3.7: Módulo de entradas de termocuplas

3.3.2. Selección de protecciones

El sistema consta de un disyuntor de corte principal, mismo que se dimensiona en base a la suma de corrientes nominales de todos los elementos del tablero multiplicado por un factor de seguridad de 1.25. La Tabla 3.3 muestra las corrientes de consumo en AC de los elementos del tablero de control. Por lo que se opta por tener un disyuntor de 6 Amperios de dos polos como protección principal.

Tabla 3.3: Corrientes AC por elemento del tablero

Elemento	***Corriente***
Fuente de poder 24VDC	0.67A
Resistencia Eléctrica	3.6A

3.3.3. Diseño de planos eléctricos

Los planos eléctricos se diseñan en base a los requerimientos previos, considerando el tipo de conectividad que disponen los diferentes elementos del tablero y se divide en dos planos, el de la Figura 3.8 muestra la conexión del PLC, Simatic IOT2040 y el módulo de termocuplas, por otro lado, el de la Figura 3.9 muestra la conexión de los relés de estado sólido que comandarán las resistencias helicoidales.

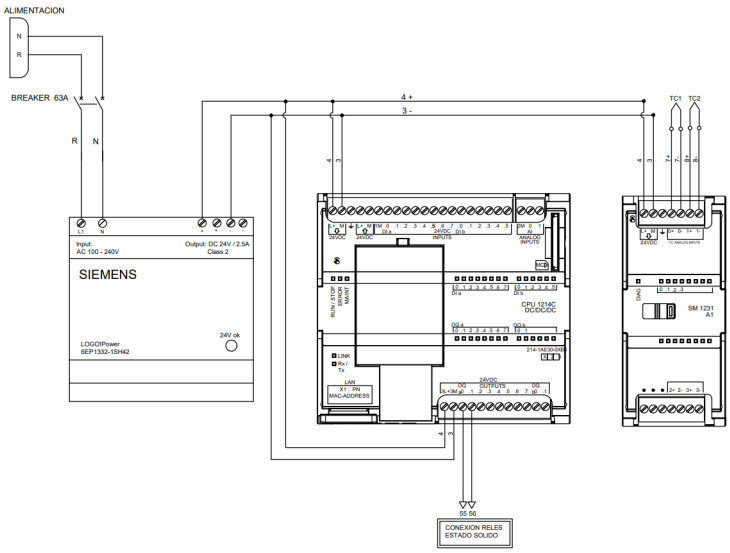


Figura 3.8: Planos de conexiones PLC Siemens

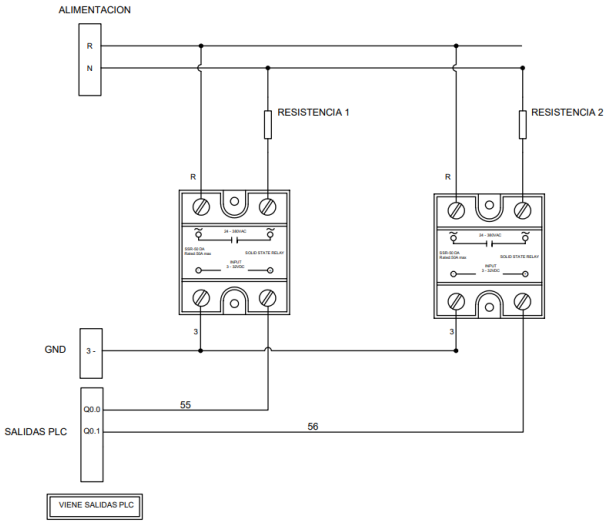


Figura 3.9: Planos de fuerza

Se muestra en la Figura 3.10 el esquemático de montaje de los equipos en el tablero de control para su posterior montaje y construcción.

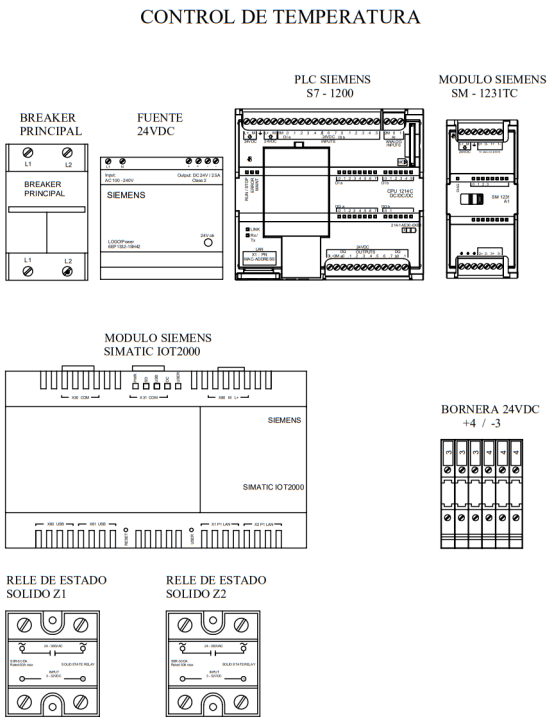


Figura 3.10: Esquemático del tablero de control

3.4. Diseño del controlador

Se procede al diseño del controlador, por lo que se establece funciones de pertenencia, para evaluar con la ayuda de reglas de control y calcular la salida hacia el controlador.

El algoritmo de control se basa en un controlador difuso, el diagrama de bloques del mismo se presenta en la Figura 3.11, un controlador difuso se compone de cuatro elementos [Passino and Yurkovich, 1998]:

- Regla base: Contiene la cuantificación de la lógica difusa en base al experto de cómo obtener un buen control.
- Mecanismo de inferencia: emula la toma de decisión del experto en interpretar y aplicar el conocimiento en la mejor manera de controlar la planta.
- Interfaz de fusificación: convierte las entradas del controlador en

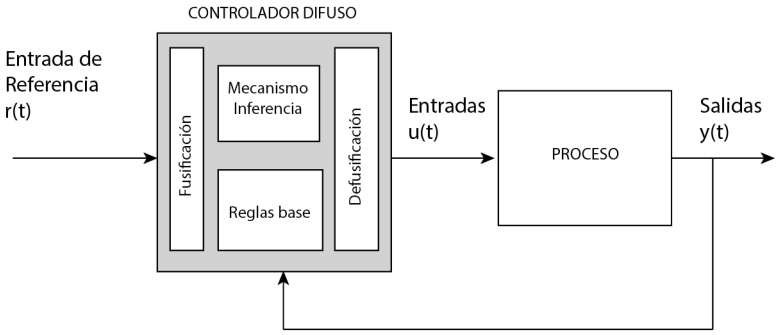


Figura 3.11: Controlador Difuso [Passino and Yurkovich, 1998]

información que el mecanismo de inferencia puede usar para activar y aplicar las reglas.

- Interfaz de desfusificación: convierte las conclusiones del mecanismo de inferencia en entradas del proceso.

Las ventajas de los controladores difusos pueden ser visibles y aparentes para problemas muy complejos, donde se tenga una idea intuitiva de cómo llegar a un alto desempeño del controlador, en estas aplicaciones un modelo matemático exacto es muy complejo de obtener, debido a la cantidad de entradas y salidas, o por su no linealidad, o procesos estocásticos [Passino and Yurkovich, 1998].

El módulo de fusificación realiza las siguientes funciones:

- Transforma las variables físicas de la señal del proceso, la señal de error en un subconjunto difuso normalizado que consiste de un intervalo para el rango de valores de entrada y las funciones de membresía normalizadas que describen el grado de confianza de la entrada perteneciente a este rango [Chen and Tat Pham, 2001].

- Selecciona una buena, razonable e idealmente óptima función de membresía bajo ciertos criterios convenientes de la aplicación.

La estructura general de un controlador lógico difuso consiste de tres elementos principales, la unidad de fusificación en el terminal de entrada, el proceso de inferencia construído sobre las reglas base del control fuzzy y la unidad de desfusificación a la salida [Chen and Tat Pham, 2001] como se muestra en la Figura 3.12.

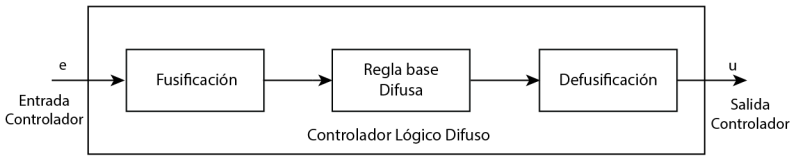


Figura 3.12: Estructura general de un controlador lógico difuso [Chen and Tat Pham, 2001]

3.4.1. Módulo de fusificación

Se establecen las funciones de membresía en la Figura 3.13 para la señal del error e,cuya ecuación 3.1 se presenta a continuación.

$$e(t) = r - y(t) \tag{3.1}$$

Donde:
r : Punto de consigna
$y(t)$: Salida de la planta

Se usa la siguiente notación en las funciones de membresía durante el diseño del control difuso.

- NS: Negativo pequeño
- NL: Negativo largo
- ZU: Cero
- PS: Positivo pequeño
- PL: Positivo largo

3.4.2. Reglas base de la lógica difusa

Se establecen las entradas del controlador, primero se dispone del error e.

La señal del error es una variable de entrada del controlador, sin embargo, para tener una regla base es necesario tener más variables de entrada auxiliares, por lo que se considera el cambio de la señal del error denominada $\dot{e}$, esta ayuda a distinguir si la curva va hacia arriba o hacia abajo. Se presenta la ecuación 3.2 de la derivada del error a continuación:

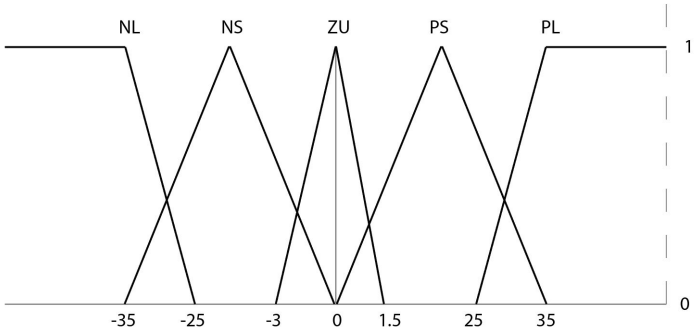


Figura 3.13: Funciones de membresía para el error

$$\dot{e(t)} = \dot{r} - \dot{y(t)} \tag{3.2}$$

Por lo que se consideran las funciones de membresía para la derivada del error que se presenta en la Figura 3.14. Estas funciones de membresía fueron seleccionadas en base a la aplicación de temperatura en cuestión y se buscó un valor a partir del cual se volvía necesario modificar la salida del controlador para evitar sobre impulsos debidos a la inercia térmica de la resistencia eléctrica.

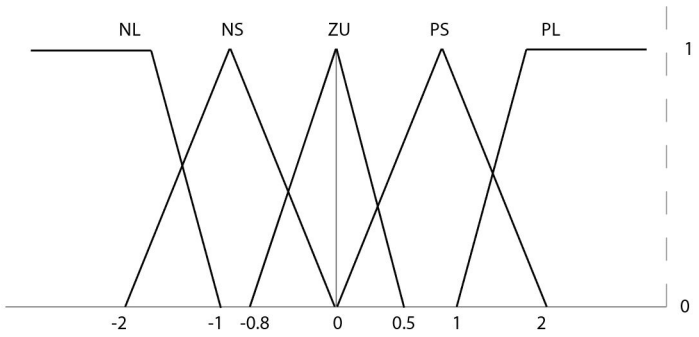


Figura 3.14: Funciones de membresía para la derivada error

Con el fin de determinar la taza de cambio máxima se efectúan pruebas, encendiendo al actuador de la planta, se empieza con una temperatura inicial de 24.7°C y luego de 39.71 segundos se alcanzan 100°C, por lo que se obtiene que la máxima razón de cambio en 1 segundo es $\dot{x} = 1{,}896°C/seg$ y por

esta razón se obtiene un valor máximo de las funciones de membresía en la derivada del error.

Posteriormente, con el fin de facilitar la implementación del algoritmo difuso en el PLC, se separan las funciones de membresía en tres tipos de formas principales que se presentan en la Figura 3.15.

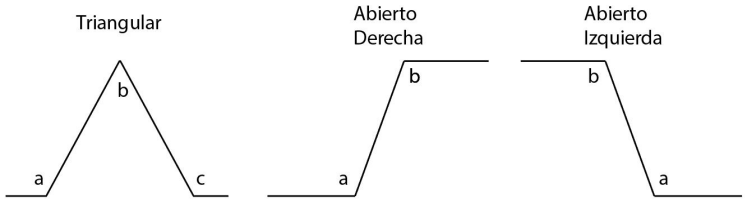


Figura 3.15: Formas bases de las funciones de membresía

La Figura 3.16 muestra las formas abiertas a la derecha e izquierda con la notación adicional requerida para la obtención de las reglas base y la Figura 3.17 para la forma triangular.

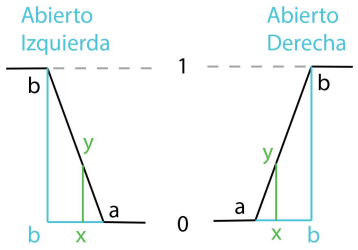


Figura 3.16: Formas abiertas a la derecha e izquierda para análisis del valor de membresía

Luego de analizar las formas de las funciones de membresía se obtiene lo siguiente 3.3, donde y es el valor de membresía, x la variable de proceso y a, b, c los puntos establecidos en las figuras 3.16 y 3.17 .

$$\text{Abierto derecha} \begin{cases} y = 1, & \textbf{si} x < b \\ y = 0, & x \geq a \\ y = \frac{x-a}{b-a}, & \text{Caso contrario} \end{cases} \tag{3.3}$$

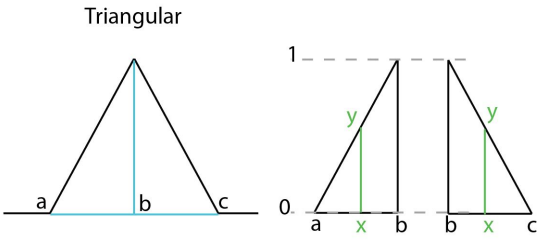


Figura 3.17: Formas triangulares para análisis del valor de membresía

$$\text{Abierto izquierda} \begin{cases} y = 1, & x \geq b \\ y = 0, & x \leq a \\ y = \frac{x-a}{b-a}, & \text{Caso contrario} \end{cases} \tag{3.4}$$

$$\text{Triangular} \quad \begin{cases} y = \frac{x-a}{b-a}, & x > a \wedge x \leq b \\ y = \frac{x-c}{b-c}, & x > b \wedge x \leq c \\ y = 0, & \text{Caso contrario} \end{cases} \tag{3.5}$$

Se establecen las reglas base para el control difuso usando un método alternativo para determinar el peso del control $u(t)$ donde se asignan los valores de Z0, S, M, L, VL de forma que se puede colocar las reglas base de forma tabular [Chen and Tat Pham, 2001] como se muestra en la Tabla 3.4. A esta tabla también se la conoce como look-up table en inglés y puede ser más exacta si se dividiera e y $\dot{e}$ en más sub casos.

La tabla 3.4 representa la regla base para un controlador difuso donde la acción de control es proporcional a ambos errores e y $\dot{e}$.

e/$\dot{e}$	NL	NS	Z0	PS	PL
NL	Z0	Z0	Z0	Z0	Z0
NS	Z0	Z0	Z0	Z0	Z0
Z0	Z0	Z0	S	M	M
PS	Z0	S	L	L	L
NL	M	L	L	VL	VL

Tabla 3.4: Tabla de reglas base

En la tabla 3.4 se usan abreviaciones M significa mediano, $Z0$ cero, S pequeño, L grande y VL muy grande.

3.4.3. Módulo de defusificación

Se procede a realizar el módulo de defusificación, este convierte todos los términos difusos creados por las reglas base a valores numéricos y luego los envía al sistema físico para se ejecute el control del sistema. Se aplica la fórmula de defusificación conocida como center-average [Passino and Yurkovich, 1998] que se presenta en la ecuación 3.6.

$$u^{crisp} = \frac{\sum_i b_i u_{premise_i}}{\sum_i u_{premise_i}} \tag{3.6}$$

Donde:
$u_{premise}$: Valor de membresía
u_{crisp} : Valor de salida
b_i : Centro de la función de membresía

Este método de defusificación es común debido a que los cómputos necesarios son más sencillos que en el método de centro de gravedad y además porque las funciones de membresía de salida son más fáciles de almacenar puesto que la única información que deben proveer es el centro de los valores b_i, no importa su forma, solo el centro [Passino and Yurkovich, 1998].

Finalmente, se seleccionan funciones de membresía para las salidas del controlador, en este caso se dispone una sola salida u, y se usan las abreviaciones de la tabla 3.4. Cabe recalcar que no se dispone control sobre la parte negativa, por ende la Figura 3.18 muestra la función de membresía tomando en cuenta solo la parte positiva. La parte inferior considera en porcentaje la salida del actuador, donde 1000 es el 100 %.

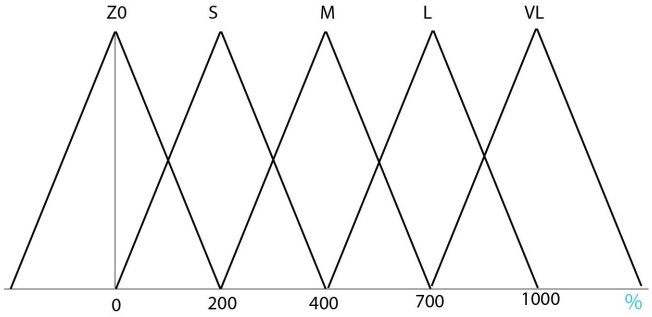


Figura 3.18: Función de membresía para la salida u del sistema

3.5. Algoritmo de control en PLC

Como se mencionó anteriormente, se usa un PLC Siemens S7-1200 con un módulo de expansión de termocuplas como se muestra en los dispositivos del proyecto en TIA Portal 3.19.

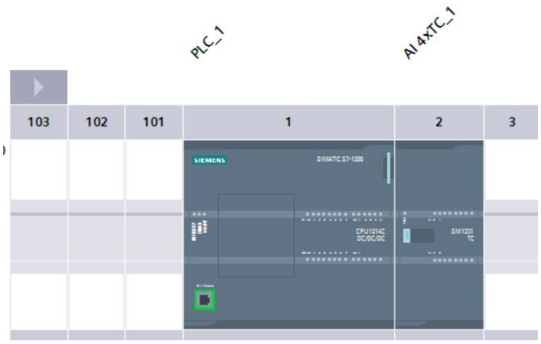


Figura 3.19: Dispositivos en TIA Portal

Con el fin de disponer una programación organizada se crean diferentes bloques de programa como se muestra en la Figura 3.20, estos bloques de programa se subdividen en interrupciones, bloques de datos y bloques de funciones.

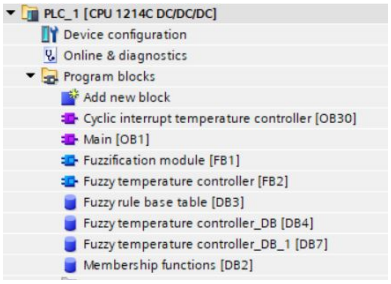


Figura 3.20: Bloques de programa del sistema

Se presenta en la Figura 3.21 el bloque de programa de ejecución del controlador en base a lo obtenido por los demás bloques que se presentan a continuación, este bloque es programado en lenguaje SCL, que básicamente es un lenguaje de control estructurado y es basado en texto para programar PLCs Siemens. Posteriormente se presenta en la Figura 3.22 el módulo de

fusificación, en la Figura 3.23 el módulo de defusificación y en la Figura 3.24 la tabla de las reglas base.

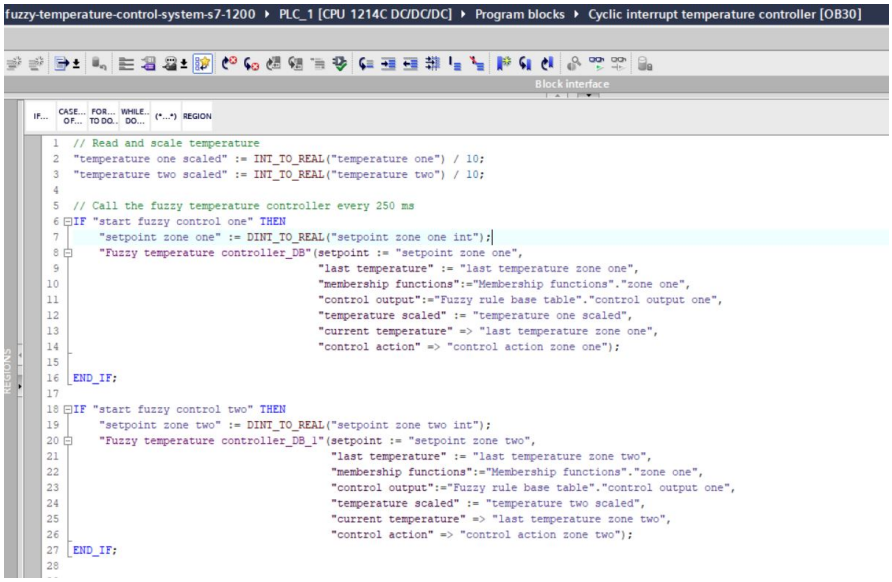


Figura 3.21: Bloque de programa de ejecución del controlador

3.6. Diseño de Comunicación Remota

Para el diseño de la comunicación remota, se establecen principios básicos de internet de las cosas (IoT), se desarrolla la comunicación mediante el uso de protocolo MQTT.

El protocolo MQTT es el protocolo estándar para Internet de las Cosas, está diseñado en una mensajería extremadamente ligera por lo que es ideal para conectar dispositivos remotos usando bajos recursos de código y de ancho de banda [Sinthuja and Thavamani, 2019]

MQTT usa un patrón de publicación y suscripción, requiere de un broker o también conocido como servidor. Todos los clientes establecen una conexión con el broker, y al cliente que envía un mensaje a través del broker se lo conoce como editor o publisher en inglés [Hillar, 2013].

El broker filtra los mensajes entrantes y los distribuye a los clientes que están interesados en el tipo del mensaje recibido. Los clientes que registran al broker como interesados en específicos tipos de mensaje son conocidos

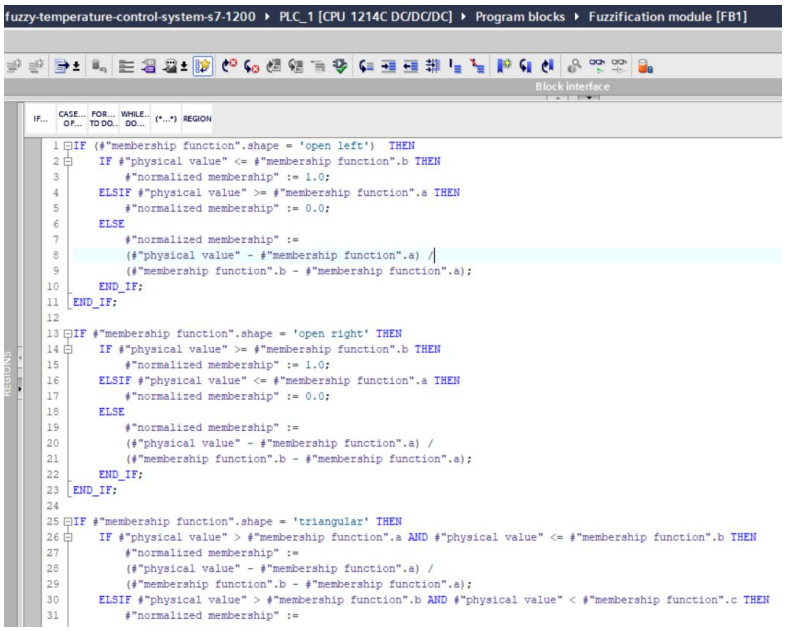


Figura 3.22: Módulo de fusificación en PLC

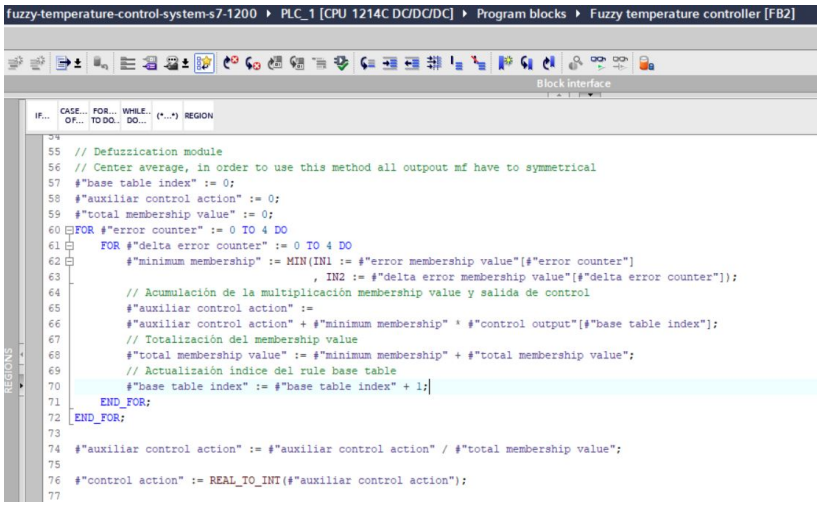
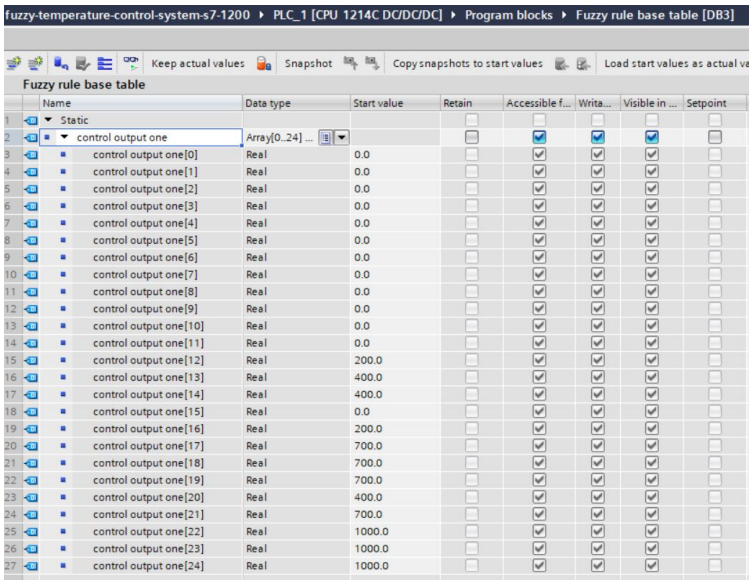


Figura 3.23: Módulo de defusificación en PLC



	Name	Data type	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint
1	Static							
2	control output one	Array[0..24] ...						
3	control output one[0]	Real	0.0					
4	control output one[1]	Real	0.0					
5	control output one[2]	Real	0.0					
6	control output one[3]	Real	0.0					
7	control output one[4]	Real	0.0					
8	control output one[5]	Real	0.0					
9	control output one[6]	Real	0.0					
10	control output one[7]	Real	0.0					
11	control output one[8]	Real	0.0					
12	control output one[9]	Real	0.0					
13	control output one[10]	Real	0.0					
14	control output one[11]	Real	0.0					
15	control output one[12]	Real	200.0					
16	control output one[13]	Real	400.0					
17	control output one[14]	Real	400.0					
18	control output one[15]	Real	0.0					
19	control output one[16]	Real	200.0					
20	control output one[17]	Real	700.0					
21	control output one[18]	Real	700.0					
22	control output one[19]	Real	700.0					
23	control output one[20]	Real	400.0					
24	control output one[21]	Real	700.0					
25	control output one[22]	Real	1000.0					
26	control output one[23]	Real	1000.0					
27	control output one[24]	Real	1000.0					

Figura 3.24: Reglas base de fusificación en PLC

como suscriptores, por ende, ambos suscriptores y editores establecen una conexión con el broker [Hillar, 2013]. Para comprender de mejor manera se presenta la Figura 3.25.

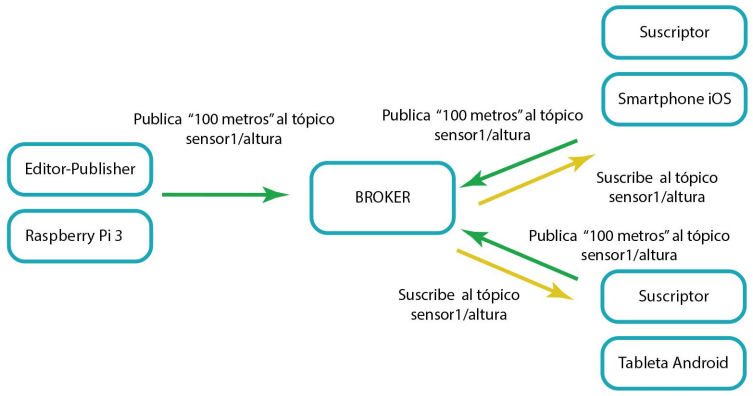


Figura 3.25: Diagrama usando MQTT entre un publisher y dos suscriptores

El tópico es un canal lógico nombrado, el broker enviará a los publishers

solo los mensajes publicados a los tópicos a los que están suscritos.

3.6.1. Instancia Virtual y broker MQTT

Con el fin de disponer de un broker para implementar el protocolo MQTT se establece un computador virtual en la nube, mediante Google Cloud, esta es una plataforma que permite contratar servicios computacionales que se ejecutan sobre la misma infraestructura que Google usa para productos de usuario final.

Se presenta en la Figura 3.26 la interfaz de Google Cloud para la creación de instancias virtuales, la instancia creada lleva el nombre de mqtt-broker-engine.

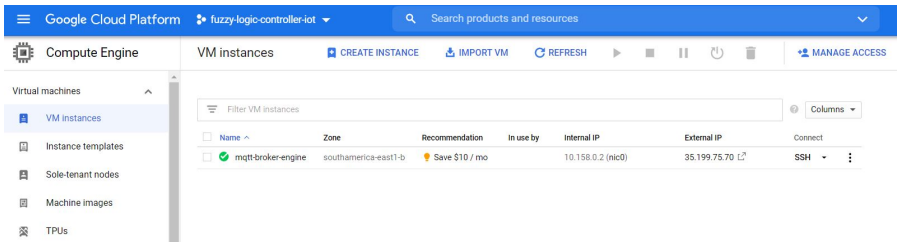


Figura 3.26: Dashboard de Google Cloud para instancias virtuales

La instancia tiene una máquina e2-micro (2 vCPUs, 1 GB memory) con una plataforma de CPU de Intel Broadwell y un sistema operativo ubuntu-1804. La Figura 3.27 presenta el histórico de los paquetes de red de la instancia virtual durante 14 días y de igual manera la Figura 3.28. Adicionalmente, se presenta en la Figura 3.29.

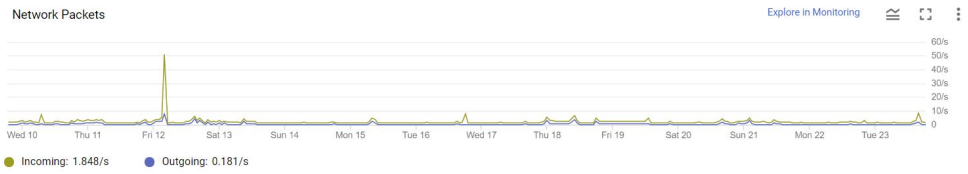


Figura 3.27: Paquetes de red en la instancia virtual

Luego de disponer la instancia virtual, es necesaria reservar una dirección IP externa, posteriormente se requiere obtener un DNS y asociarlo al nombre

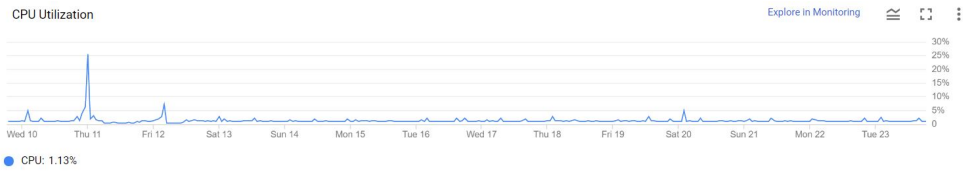


Figura 3.28: Uso del CPU de la instancia virtual

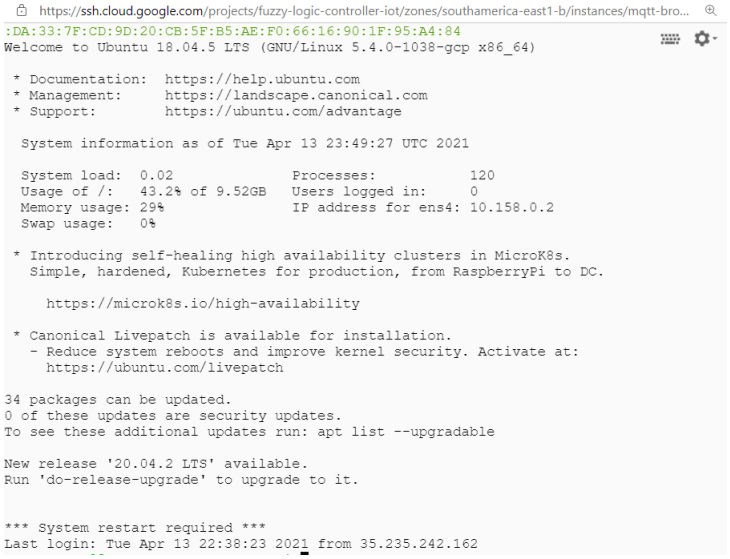


Figura 3.29: Ventana inicio instancia virtual

del dominio de Google Cloud. Cabe recalcar que la dirección IP estática de la VM es *35.199.75.70*.

Se obtiene un dominio de manera gratuita en "*freenom.com*" que es una página web que permite obtener dominios, la Figura 3.30 muestra la página donde se puede administrar el dominio obtenido, se usa el dominio fuzzy-controller-iot.tk.

Posteriormente, en Google Cloud se debe configurar el DNS, por lo que se accede a los servicios de red y se agrega el nombre del DNS como se muestra en la Figura 3.31 y se crea una zona DNS. Una vez establecido el nombre y zona DNS, se procede con la creación de un *record set* como se muestra en la Figura 3.32.

Ahora, nuevamente se debe configurar el dominio en *Freenom.com* y

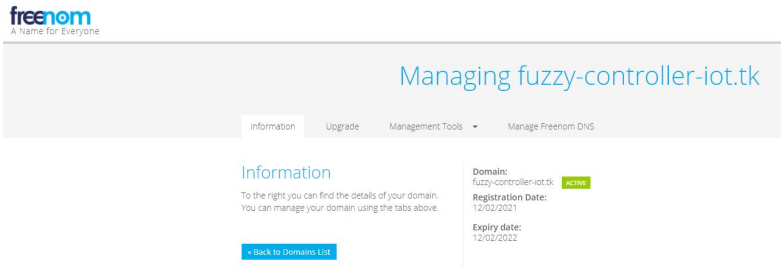


Figura 3.30: Obtención de dominio gratuito

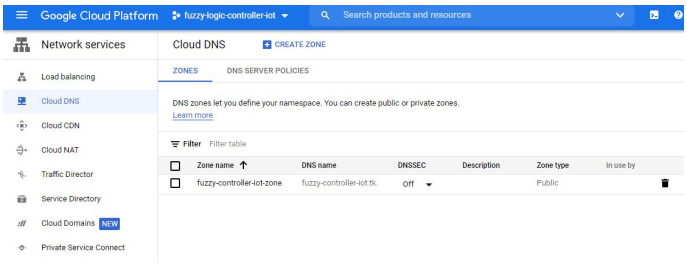


Figura 3.31: Configuración de nombre de DNS

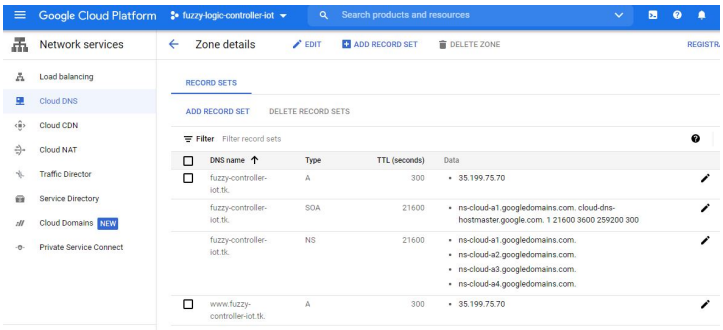


Figura 3.32: Configuración del record set del DNS

agregar los *nameservers* generados anteriormente en Google Cloud como se muestra en la Figura 3.33.

Una vez finalizada la configuración del DNS se abren los puertos 1883 y 8083, cabe recalcar que MQTT usa el puerto 1883 y el 8083 cuando funciona sobe TLS [Sinthuja and Thavamani, 2019], por lo que las conexiones

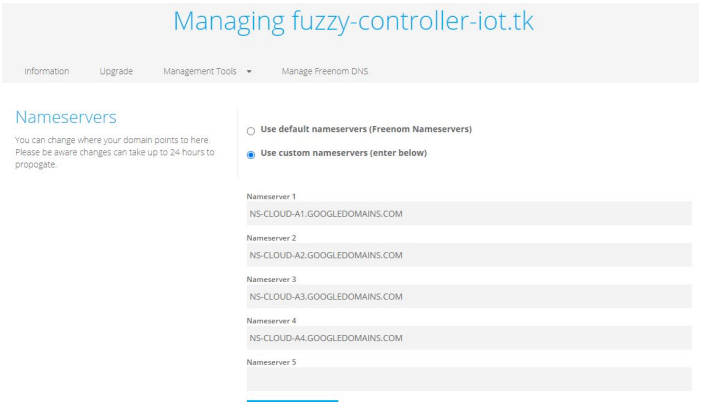


Figura 3.33: Configuración del record set del DNS

entrantes estarán asociadas al puerto 1883 y este debe estar abierto. MQTT usa el protocolo de transporte TCP, el cual usa el puerto 8083 por lo que este puerto también debe estar abierto.

Los puertos anteriormente mencionados deben ser abiertos desde el firewall de la instancia virtual con los comandos *"sudo ufw allow from any to any port 1883 proto tcp"* y *"sudo ufw allow from any to any port 8083 proto tcp"*.

Una vez abiertos los puertos y creada la instancia virtual, se debe instalar *Mosquitto*, este es un broker de código abierto que implementa el protocolo MQTT, es ligero y se lo puede instalar en varios dispositivos [Sinthuja and Thavamani, 2019].

Se procede con la creación de un repositorio *"sudo apt-add-repository ppa:mosquitto-dev/mosquitto-ppa"*, posteriormente con la instalación de mosquitto *"sudo apt-get install mosquitto"* y de mosquitto-clients sobre la instancia virtual *"sudo apt-get install mosquitto-clients"*. Se comprueba en Ubuntu los paquetes instalados y sus versiones como se muestra en la Figura 3.34.

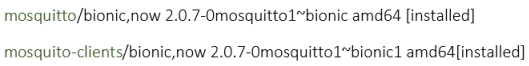


Figura 3.34: Verificación de la instalación de Mosquitto en VM

Se establecen certificados de seguridad SSL con el fin de permitir la

comunicación con websockets ssl. El protocolo ssl consiste de un arreglo de mensajes y reglas de cuando enviar y no enviar mensajes, en este protocolo se tienen clientes y servidores, los clientes inician la comunicación y tienen la responsabilidad de proponer un arreglo de opciones ssl a usarse durante el intercambio; mientras que el servidor selecciona las opciones propuestas por el cliente y decide cuál usar [Thomas and Wiley, 1999].

Cuando se autentifica la identidad, el servidor envía un mensaje de certificado, este mensaje contiene una cadena de certificados que empiezan con el certificado de la clave pública del servidor y termina con el certificado raíz de la autoridad. El cliente tiene la responsabilidad de asegurarse que puede confiar en el certificado que recibe del servidor, esta responsabilidad significa asegurarse de las firmas, tiempos de validación y estado de revocación [Thomas and Wiley, 1999].

Se generan los certificados usando *certbot.eff.org* y se instala sobre la VM, posteriormente cuando se solicite un dominio se ingresa *fuzzy-controller-iot.tk*. Los certificados se almacenan en el directorio */etc/letsencrypt/live/fuzzy-controller-iot.tk*. Una vez obtenidos los certificados sobre la VM, se deben transferior a la carpeta de mosquitto.

Se procede con la configuración del broker Mosquitto. Se accede a la configuración mediante */etc/mosquitto/mosquitto.conf"* y se modifica el archivo para obtener de la manera en que se muestra en la Figura 3.35. Adicionalmente, se agrega un usuario y contraseña y se procede con un reboot de la VM.

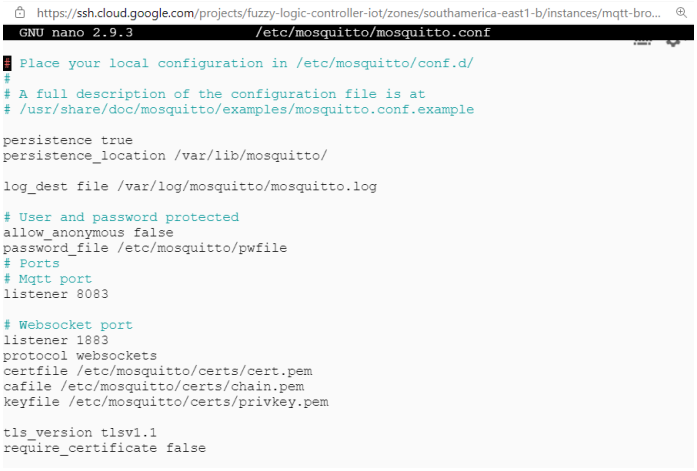


Figura 3.35: Archivo de configuración de Mosquitto en VM

La Figura 3.36 muestra el estado de Mosquitto, al momento de la captura de pantalla, el cliente no estaba conectado por lo que se aprecia una señal de alerta que no permite adquirir la dirección del cliente, sin embargo, el broker se encuentra activo y ejecutándose.

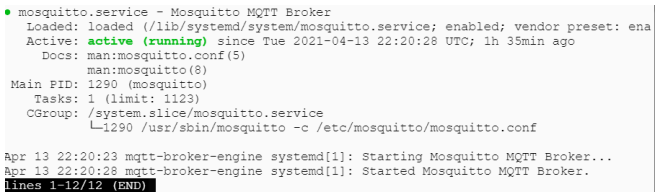


Figura 3.36: Estado del broker Mosquitto

Mosquitto permite observar la actividad del mismo, como se muestra en la Figura 3.37.

Figura 3.37: Logs de Mosquitto en VM

3.6.2. Gateway

Como se mencionó al inicio del capítulo, el sistema dispone de un Gateway que permite tener una puerta de enlace y actúa de interfaz entre la conectividad de diferentes dispositivos o equipos y por ende permite la conexión con el internet. Este gateway implementado es un IoT2040 de Siemens, cuyas características se presentaron anteriormente.

Para empezar la configuración del IoT de Siemens se debe empezar a cargar sobre el mismo una imagen provista por el fabricante, esta imagen del

sistema operativo es descargada en una Micro SD que se la conecta sobre el dispositivo. El sistema operativo viene precargado con Node-RED una herramienta orientada a la programación por nodos que permite una fácil programación del equipo para la conectividad con el PLC y con el broker MQTT. La Figura 3.38 muestra la arquitectura de conexión del sistema.

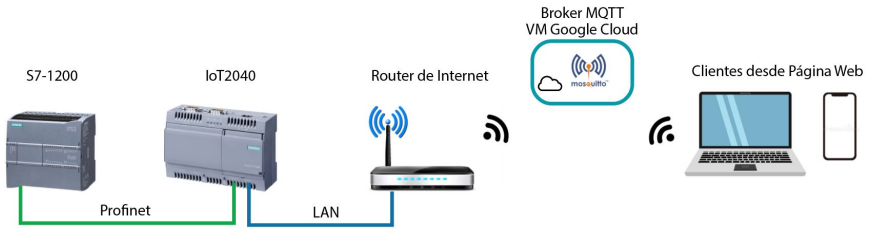


Figura 3.38: Arquitectura de conectividad del Sistema

El IoT2040 fue programado en Node-RED y a continuación se muestra en la Figura 3.39 el arreglo de nodos que permiten la conectividad requerida. El IoT2040 dispone de dos interfaces ethernet, el puerto eth0 está configurado con dirección IP estática *192.168.100.30* y el puerto eth1 como DHCP puesto que este será el que permitirá la conexión con el router y por ende la salida a internet

Cada vez que se apaga y se enciende el sistema se requiere que Node-RED se ejecute automáticamente por lo que se configura las opciones de auto inicio del IoT2040, permitiendo a Node-RED inicializarse cuando se encienda el dispositivo. Debido a la versión de Node-RED con la que cuenta el IoT2040, no se puede instalar los nodos desde la paleta, sino que mediante comandos para la intstalación de versiones antiguas compatibles con la versión de Node-RED, mediante el comando *npm install node-red-contrib-s7@2.2.1.*

Para enlazarse y leer directamente los registros del PLC se usan los nodos *S7* que deben ser configurados en base a la IP del PLC, se deben establecer las variables 3.41, puertos y tiempos de muestreo, como se muestra en la Figura 3.40.

Adicionalmente, sobre Node-RED se usa el nodo MQTT que permite publicar o suscribirse al broker MQTT, la Figura 3.42 muestra la configuración del nodo y la Figura 3.43 muestra cómo se establecen la configuración de los mensajes de nacimiento, cierre y testamento. La Tabla 3.5 muestra los tópicos usados.

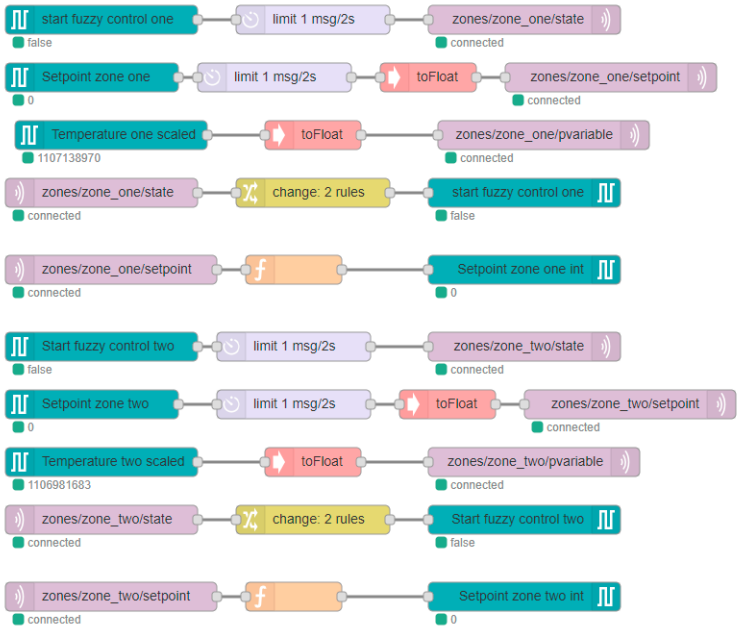


Figura 3.39: Nodos en Node-RED del IoT2040

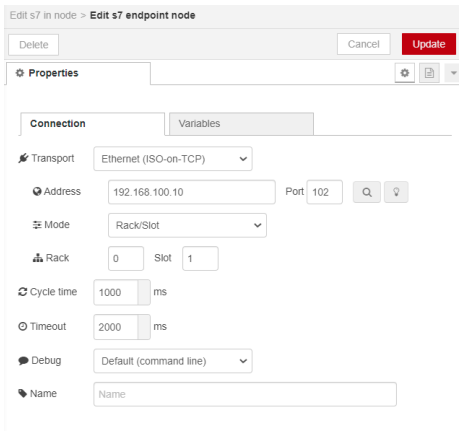


Figura 3.40: Configuración del Nodo S7 en Node-RED

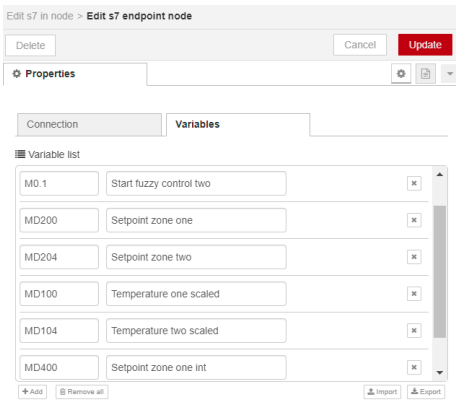


Figura 3.41: Variables del Nodo S7 en Node-RED

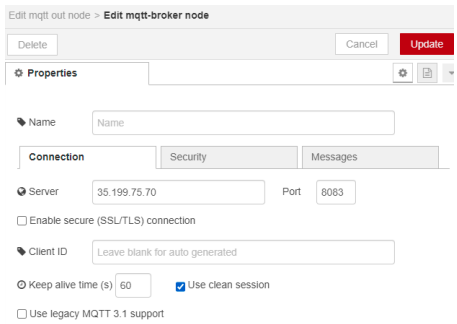


Figura 3.42: Configuración de conexión del Nodo MQTT en Node-RED

Tabla 3.5: Tópicos usados en MQTT

Tópico	***Descripcin***
zones/satus	Estados de las zonas
zones/zone_one/state	Estado de la zona 1
zones/zone_one/setpoint	Punto de consigna de la zona 1
zones/zone_one/pvariable	Variable de proceso de la zona 1
zones/zone_two/state	Estado de la zona 2
zones/zone_two/setpoint	Punto de consigna de la zona 2
zones/zone_two/pvariable	Variable de proceso de la zona 2

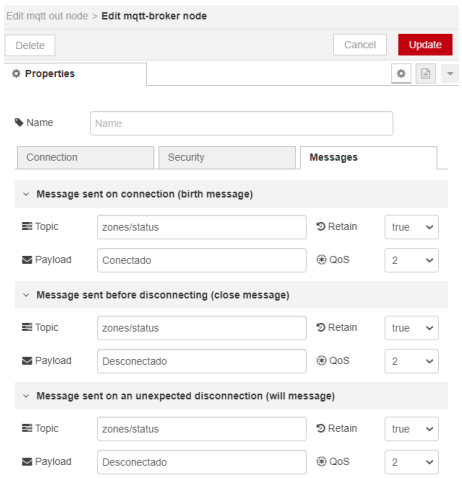


Figura 3.43: Mensajes del Nodo MQTT en Node-RED

3.6.3. Aplicación Web

Se desarrolla una aplicación web sobre la cual se dispone la interfaz con la que el usuario puede visualizar estados de conexión del IoT2040, encender las zonas de calentamiento, establecer puntos de consigna, visualizar el valor actual de la variable de proceso, así como un histórico instantáneo del comportamiento de temperatura que es la respuesta del controlador.

Para la creación de la aplicación web, se usa Angular, esta es un framework de aplicación de diseño y una plataforma de desarrollo que permite crear aplicaciones de una sola página de una manera eficiente,

Para el desarrollo en Angular se establecen tres lenguajes de programación esenciales, el primero es *TypeScript* que es un lenguaje de código abierto que está construido sobre JavaScript, este lenguaje se lo diseñó con el fin de desarrollar largas aplicaciones [Maharry, 2013]. Otro de los lenguajes de programación es *CSS* que viene de sus siglas en inglés cascading style sheets, es un lenguaje usado para describir la presentación de un documento escrito en HTML o en XML, CSS describe cómo los elementos deberán ser presentados en la pantalla [Budd et al., 2006]. HTML por otro lado, es el bloque de construcción más básico de la red, define el significado y estructura del contenido web, proviene del significado en inglés HyperText Markup Language [Gosney, 2004].

Se muestra en la Figura 3.44 el código en TypeScript implementado en Angular para la creación del formulario de inicio de sesión en la aplicación

y en la Figura 3.45 el código HTML para la estructura del formulario de accesos de usuarios.

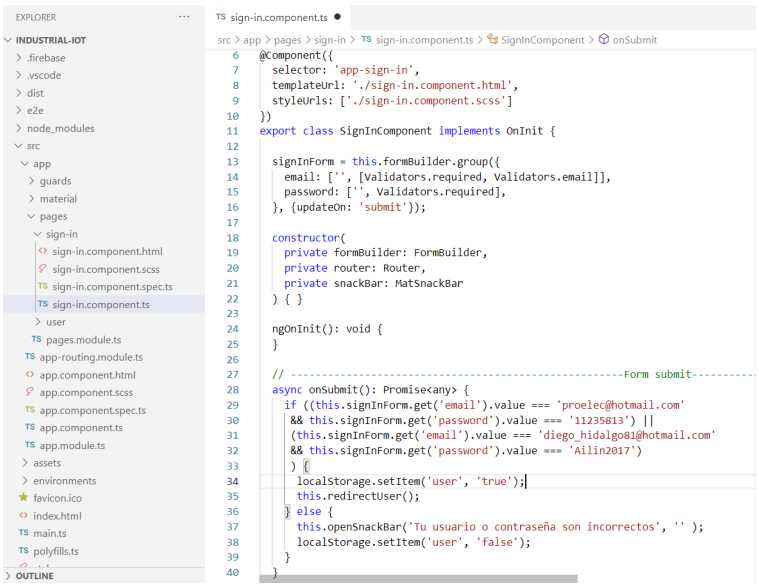


Figura 3.44: Código en TypeScript para formulario de acceso de usuario

Se debe realizar la configuración para la conectividad con el broker MQTT, la Figura 3.46 muestra el código para la conectividad anteriormente mencionada usando WebSockets a la dirección *wss://fuzzy-controller-iot.tk:1883/mqtt'*. Por otro lado, la Figura 3.47 muestra el código usado para la publicación de los valores de puntos de consigna desde la aplicación web.

Con el fin de disponer de un servicio de Hosting, se optó por el hosting provisto por Google, mediante Firebase, que es una plataforma para la creación de aplicaciones web y aplicaciones móviles, sobre esta plataforma se cargaron los archivos de la aplicación web desarrollada en angular, y permite la entrega e contenido mediante una conexión segura y se puede publicar de forma rápida, la Figura 3.48 muestra el dashboard de Firebase. Firebase permite disponer del servicio de forma gratuita con ciertos limitantes, almacenamiento de hasta 10GB y 360MB/día de transferencia de datos, sin embargo, para la aplicación desarrollada, las capacidades gratuitas son suficientes.

Debido a los requerimientos de disponer un inicio de sesión por usuarios

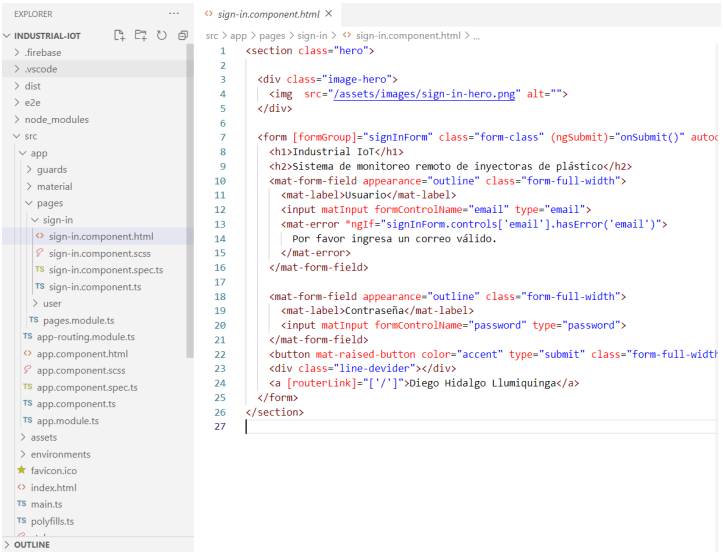


Figura 3.45: Código HTML para el formulario de acceso de usuario

Figura 3.46: Código en TypeScript para formulario de acceso de usuario

3.49, indicadores de estado de conexión, botones que permiten encender las diferentes zonas de calentamiento, ingreso de puntos de consigna,

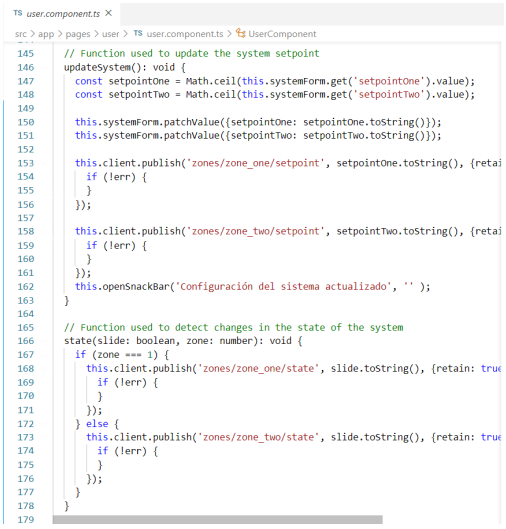


Figura 3.47: Código HTML para el formulario de acceso de usuario

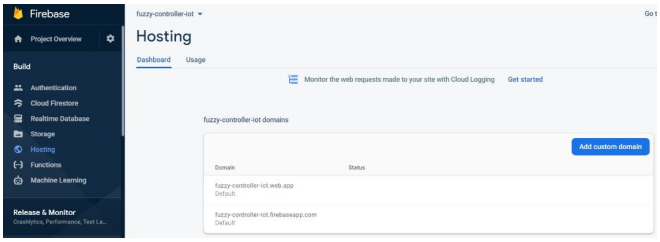


Figura 3.48: Dashboard de Firebase para Hosting

visualización instantánea de variables de proceso e históricos se presenta la Figura 3.50 con la versión final de la aplicación web.

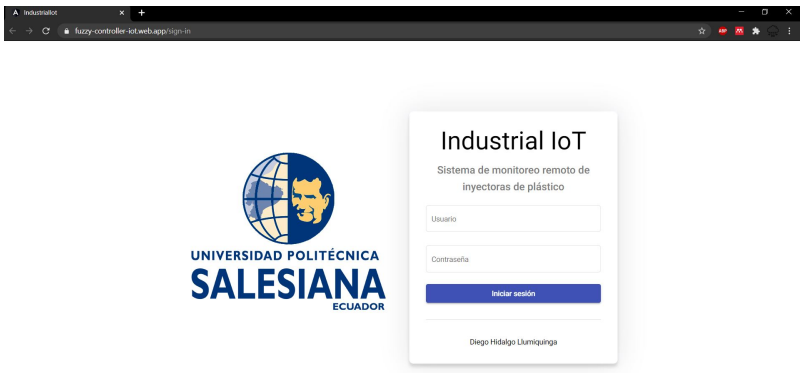


Figura 3.49: Aplicación web en pantalla de inicio de sesión

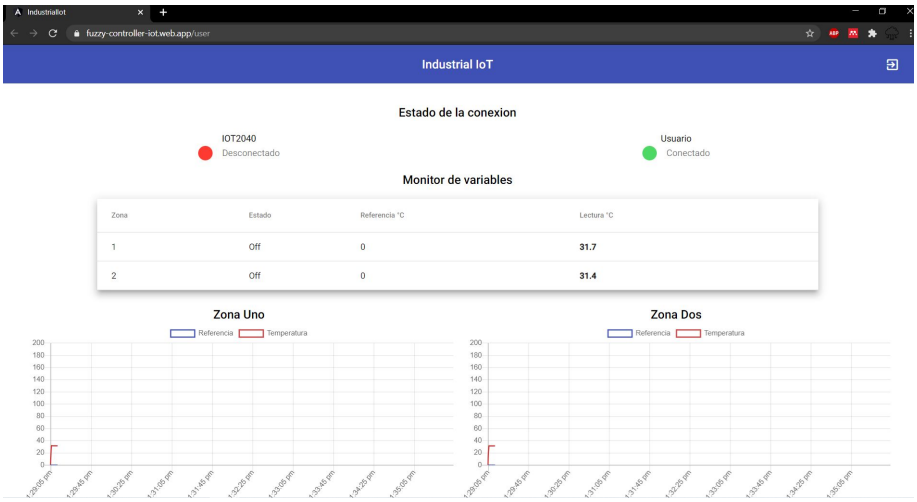


Figura 3.50: Aplicación web en pantalla principal

Capítulo 4

Resultados y Conclusiones

En este capítulo se mencionan los resultados alcanzados en el proyecto desarrollado y las conclusiones posteriores a la implementación y pruebas del sistema controlado por lógica difusa y monitoreado de forma remota usando herramientas IoT.

4.1. Resultados y análisis

Se establecen los resultados y análisis en base a criterios técnicos de funcionalidad y fiabilidad del sistema, del controlador y de la conectividad.

4.1.1. Algoritmo de control

Se establecen pruebas al algoritmo de control implementado con diferentes puntos de consigna, estos valores asignados se los realiza en base a los parámetros de operación de procesos de inyección con moldes de colada caliente o *hot runners* en inglés.

La Figura 4.1 indica gráficamente la respuesta del control difuso y se establece en color verde el punto de consigna y en color rojo la variable de proceso, de forma que se puede apreciar que el sistema no presenta sobre impulsos y es capaz de realizar un seguimiento a diferentes puntos de consigna, con un tiempo de estabilización cercano a los 60 segundos para puntos de consigna superiores al estado actual y de alrededor de 90 segundos para puntos de consigna inferiores, debido a la inercia térmica que disponen las resistencias helicoidales en conjunto con el elemento calefactor del molde. Adicionalmente la Figura 4.2 muestra la respuesta del controlador difuso en base a lo que se puede visualizar en la aplicación web.

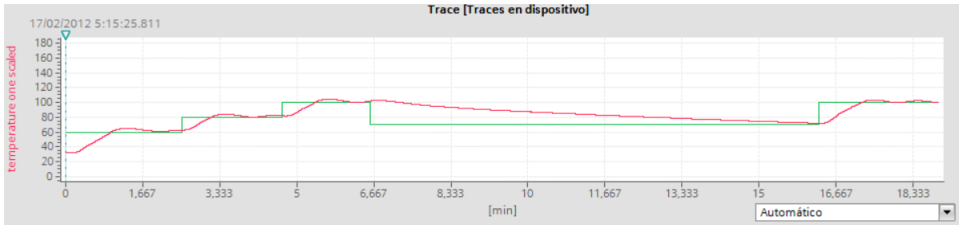


Figura 4.1: Respuesta del sistema con el controlador difuso

Sin embargo, con el fin de poder establecer una idea más clara de la fiabilidad y robustez del controlador implementado, se lo compara con un controlador PID como se muestra en la Figura 4.3, se puede apreciar de forma clara que el controlador PID presenta sobre impulsos y mayor tiempo de estabilización, y a su vez el rastreo del punto de consigna no es inmediato como en el caso del controlador difuso como se muestra en las figuras 4.5 y 4.6. Las gráficas fueron obtenidas usando la herramienta Trace que proporciona el software de siemens TIA Portal.

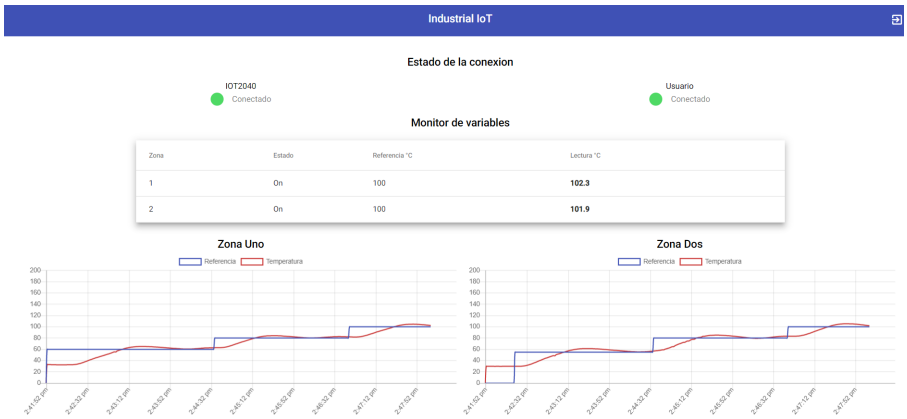


Figura 4.2: Respuesta del sistema con el controlador difuso en la aplicación web

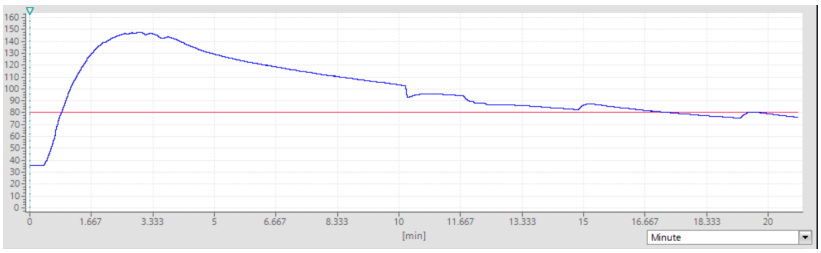


Figura 4.3: Respuesta del controlador PID

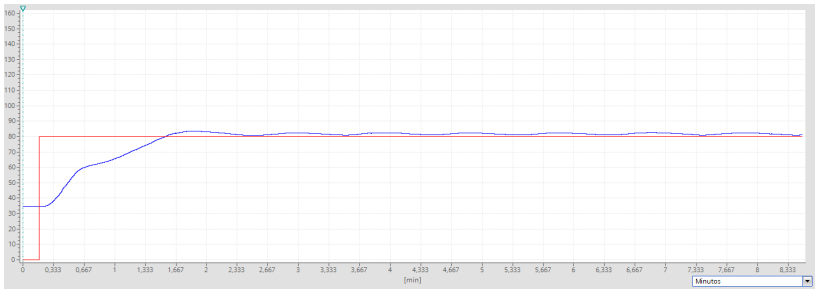


Figura 4.4: Respuesta del controlador Difuso

La Tabla 4.1 presenta los datos de interés críticos entre ambos

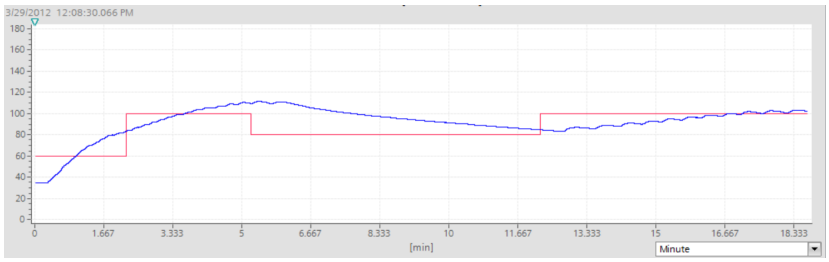


Figura 4.5: Respuesta del controlador PID a diferentes puntos de consigna

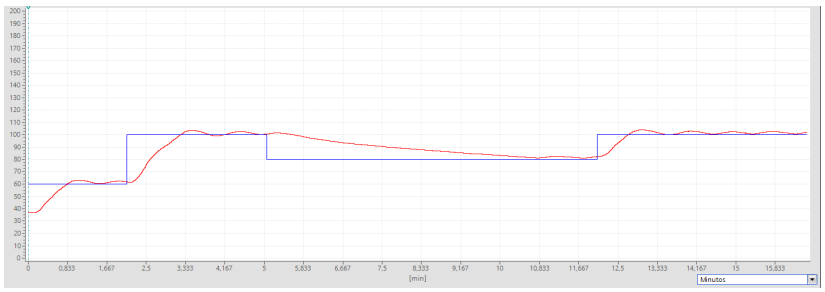


Figura 4.6: Respuesta del controlador Difuso a diferentes puntos de consigna

controladores con el fin de obtener una comparación clara.

Tabla 4.1: Comparativa entre controlador PID y Difuso

Parámetro	***PID***	***Fuzzy***
Tiempo de estabilización	12-15 min	2 seg
Sobre-impulso	Significativo	Poco Significativo
Setpoint-tracking	Regular	Satisfactorio

4.1.2. Solución IoT

Se realizó una solución IoT usando como gateway el Simatic IoT2040, sobre el cual se ejecuta Node-RED, sin embargo, los tiempos de arranque del equipo cuando se desenergiza y se lo vuelve a encender, son considerables, llegando hasta los 200 segundos, tiempo en el cual el equipo no dispone

conexión a internet y los nodos de conectividad con el PLC se encuentran inactivos. Adicionalmente, se analiza que si se desconecta la conexión a internet del equipo, este no detecta dicha desconexión y no envía señales de fallo a la aplicación web, por lo que toca resetearlo de forma manual.

Por otro lado, cuando las condiciones de trabajo son óptimas y se dispone de una buena conexión a internet, el gateway seleccionado trabaja de manera correcta, se enlaza y permite la publicación y suscripción con el broker MQTT, habilitando así la conectividad con la aplicación web, misma que presenta una fácil interacción y operación.

La aplicación web consta de una página principal, donde el usuario debe identificarse para ingresar a operar y monitorear el proceso, esto da seguridad al control remoto permitiendo mantener la confiabilidad a personas autorizadas. Adicionalmente, la aplicación web permite una operación sencilla, para que el usuario puede visualizar las variables de proceso, puntos de consigna en la siguiente Figura 4.7 y el estado operativo de la conexión remota como se muestra en la Figura 4.8.

Figura 4.7: Visualización de datos del sistema en aplicación web

Figura 4.8: Visualización de estado de las conexiones en la aplicación web

El usuario a su vez puede visualizar en forma de histograma la variable actual del proceso y el punto de consigna por cada zona, así como también asignar el valor del punto de consigna, confirmar los datos y encender o apagar las zonas de calentamiento, como se muestra en la Figura 4.9.

Finalmente, el usuario podrá acceder a la aplicación web desde su smartphone o tablet, puesto que la misma fue diseñada de forma responsiva como se puede apreciar en las Figuras 4.10, 4.11 y 4.12.

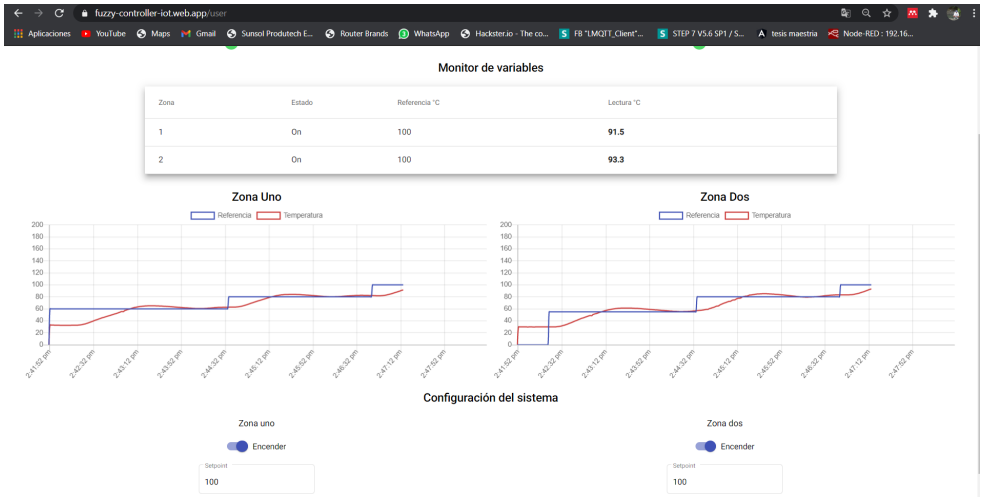


Figura 4.9: Visualización de históricos y modificación de datos del sistema en aplicación web

El hosting de Firebase arroja resultados de almacenamiento mínimos que se presentan en la Figura 4.13 donde se aprecia que se está consumiendo un 0.17 % del almacenamiento gratuito, por otro lado en la Figura 4.14 se observa que se ha realizado un 1.34 % de descargas del total de 10GB gratuitas.

4.1.3. Implementación física

Se presentan a continuación las imágenes de la implementación del tablero eléctrico y la planta del sistema con las resistencias helicoidales. El sistema posee conectores multi pin que permiten un desmontaje sencillo, lo que facilita el mantenimiento y una conexión eficiente y duradera como se muestran en las figuras 4.15, 4.16 y 4.17.

Figura 4.10: Ingreso de usuarios en aplicación web desde smartphone

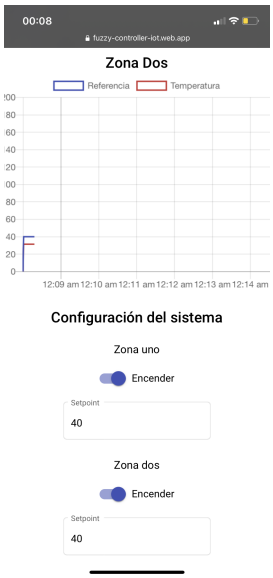


Figura 4.11: Variables y estado en aplicación web desde smartphone

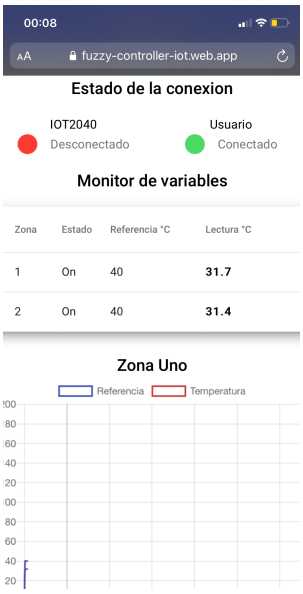


Figura 4.12: Gráficas y puntos de consigna en aplicación web desde smartphone

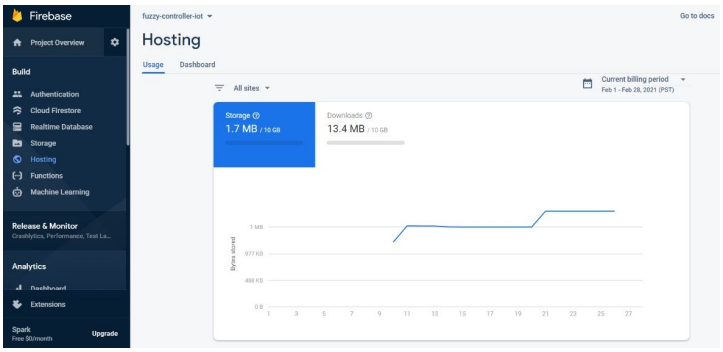


Figura 4.13: Estado del almacenamiento en el Hosting de Firebase

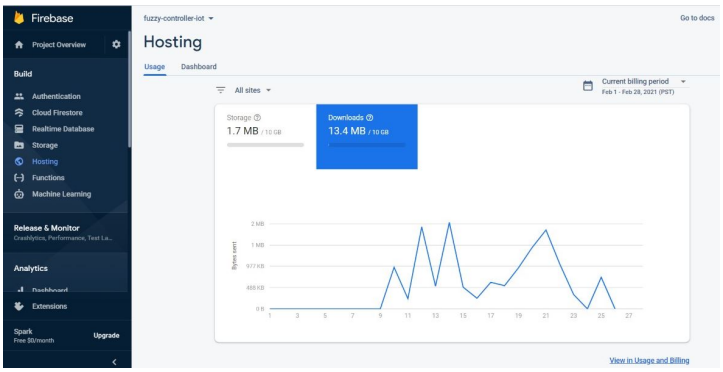


Figura 4.14: Estado de las descargas en el Hosting de Firebase

Figura 4.15: Planta de calentamiento dual del sistema

Figura 4.16: Tablero eléctrico de control vista interior

Figura 4.17: Tablero eléctrico de control vista externa

4.2. Conclusiones

El sistema controlado con lógica difusa entrega una respuesta rápida sin sobre impulso lo que repercute en un controlador más eficiente y mejora la respuesta del sistema ante perturbaciones y puntos de consigna diferentes, usando una sintonización y calibración en base al juicio de expertos sin requerir de la obtención de modelos matemáticos complejos.

La comparación del control difuso versus el control clásico PID presentan varios resultado que se los describe a continuación, el sistema con lógica difusa presenta tiempos de respuesta rápidos lo que ocasiona una pronta estabilidad en la variable de temperatura frente al PID, también el error en estado estable es menor, lo que le da más precisión al control difuso versus el control PID. Lo más importante, que el control difuso responde mejor a las perturbaciones ya que no hay sobre impulso y siempre el seguimiento al punto de consigna no lo sobrepasa.

El proyecto se enfoca en el desarrollo de una solución nueva, utilizando equipos robustos y confiables para procesos industriales, ya que en la actualidad la mayoría de estudios del internet de las cosas se los desarrolla en base a tarjetas electrónicas como Arduino y Raspberry por lo que su aplicabilidad en el sector industrial es reducida.

El uso de lenguaje de programación SCL (Programación Estructurada) que dispone el PLC Siemens S7-1200 de gama media, permite una implementación de controladores robustos sobre procesos industriales, enfocándose en la funcionalidad y confiabilidad que caracterizan a estos controladores lógicos programables.

La implementación del IoT2040 de Siemens, en este proyecto aportó compatibilidad en la comunicación entre el PLC y la Nube, de una manera óptima, entregando conectividad al broker MQTT, permitiendo una suscripción y publicación entre servidores y clientes.

Las plataformas de Google permiten el desarrollo de soluciones innovadoras para el sector industrial y los costos en caso de exceder las condiciones gratuitas son muy convenientes, a su vez que brindan confiabilidad, seguridad, robustez y disponibilidad, dado que estas plataformas son las mismas que Google usa para productos de usuario final. La instancia virtual creada está corriendo 24/7 sin ningún inconveniente, por lo que la disponibilidad del broker MQTT no será una preocupación.

La aplicación web desarrollada permite una conexión con el proceso de forma remota, de una manera sencilla, accediendo a un dominio gratuito adquirido durante el desarrollo del proyecto. Su compatibilidad con smartphones o dispositivos móviles gracias a su diseño responsivo, permite

que el usuario pueda conectarse desde cualquier parte del mundo en todo momento para poder visualizar, monitorear y controlar cualquier proceso.

El proyecto desarrollado aporta significativamente a la matriz productiva, gracias a su enfoque a fábricas inteligentes y a nuevos algoritmos de control que permitan tener plantas más productivas y eficientes que innoven el desarrollo del país y de la tecnología nacional.

4.3. Recomendaciones

El uso de controladores difusos se lo recomienda para procesos cuyo modelo matemático sea complejo y dispongan de varias entradas y varias salidas.

El uso gateways de fabricantes que no proporcionan un soporte técnico actualizado repercute en problemas durante la conectividad con nuevas librerías y dispositivos, dejando a las soluciones de industria 4.0 poco fiables y eficientes.

Este sistema se puede personalizar a cualquier industria e inclusive no se necesita tener una contratación anual por servicios prestados con empresas privadas, el sistema permite tener un bajo tráfico de datos, lo que repercute en gratuidad de los servicios necesarios.

Este proyecto permite la integración de sistemas de Big Data o Cloud Computing generando un mayor acercamiento a los pilares de la Industria 4.0.

Bibliografía

A. Budd, C. Moll, and C. Simon. *CSS Mastery*, volume 53. 2006. ISBN 9788578110796.

A. V. Carvalho, A. Chouchene, T. M. Lima, and F. Charrua-Santos. Cognitive manufacturing in industry 4.0 toward cognitive load reduction: A conceptual framework. *Applied System Innovation*, 3(4):1–14, 2020. ISSN 25715577. doi: 10.3390/asi3040055.

M. L. Chaves, L. Sánchez-González, E. Díez, H. Pérez, and A. Vizán. Experimental assessment of quality in injection parts using a fuzzy system with adaptive membership functions. *Neurocomputing*, 391:334–344, 2020. ISSN 18728286. doi: 10.1016/j.neucom.2019.06.108. URL https://doi.org/10.1016/j.neucom.2019.06.108.

G. Chen and T. Tat Pham. *Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems*. CRC Press LLC, Florida, 2001. ISBN 9781787284395.

M. Y. Chen, Y. C. Chen, and S. C. Chen. Fuzzy control on manufacturing welding systems: To apply fuzzy theory in the control of weld line of plastic injection-molding. *Advances in Industrial Control*, (9781846284687): 315–324, 2006. ISSN 21931577. doi: 10.1007/978-1-84628-469-4_21.

Z. X. Chen, Z. H. Wan, and C. Guo. Comprehensive simulation analysis of plastic injection process. *3rd International Symposium on Intelligent Information Technology Application Workshops, IITAW 2009*, pages 46–49, 2009. doi: 10.1109/IITAW.2009.80.

A. Chiarini, V. Belvedere, and A. Grando. Industry 4.0 strategies and technological developments. An exploratory research from Italian manufacturing companies. *Production Planning and Control*, 31(16): 1385–1398, 2020. ISSN 13665871. doi: 10.1080/09537287.2019.1710304. URL https://doi.org/10.1080/09537287.2019.1710304.

A. Devillez, P. Billaudel, A. Bourmaud, and G. V. Lecolier. Use of the fuzzy pattern matching method for diagnosis of a plastic injection moulding process. *European Control Conference, ECC 1999 - Conference Proceedings*, pages 2114–2119, 2015. doi: 10.23919/ecc.1999.7099631.

J. Gosney. *HTML Professional Projects*. 2004. ISBN 9788578110796.

A. K. Gupta and R. Johari. IOT based Electrical Device Surveillance and Control System. *Proceedings - 2019 4th International Conference on Internet of Things: Smart Innovation and Usages, IoT-SIU 2019*, pages 1–5, 2019. doi: 10.1109/IoT-SIU.2019.8777342.

G. Hillar. *MQTT Essentials - A Lightweight IoT Protocol*, volume 53. Birmingham, 2013. ISBN 9788578110796.

T. Kalsoom, N. Ramzan, S. Ahmed, and M. Ur-Rehman. Advances in sensor technologies in the era of smart factory and industry 4.0. *Sensors (Switzerland)*, 20(23):1–22, 2020. ISSN 14248220. doi: 10.3390/s20236783.

D. Maharry. *TypeScript Revealed*. Apress, 2013. doi: 10.1007/978-1-4302-5726-4.

K. Passino and S. Yurkovich. *Fuzzy control*, volume 517. Addison Wesley Longman, California, 1998. ISBN 9783319021348. doi: 10.1007/978-3-319-02135-5_5.

R. Petrasch and R. Hentschke. Process modeling for industry 4.0 applications: Towards an industry 4.0 process modeling language and method. *2016 13th International Joint Conference on Computer Science and Software Engineering, JCSSE 2016*, (Cc):1–5, 2016. doi: 10.1109/JCSSE.2016.7748885.

C. Pillajo and R. Hincapie. Wireless Network Control Systems De la teoría a la práctica. Technical report, oct 2018. URL http://dspace.ups.edu.ec/handle/123456789/17082.

S. Rajput and S. P. Singh. Industry 4.0 Model for circular economy and cleaner production. *Journal of Cleaner Production*, 277:123853, 2020. ISSN 09596526. doi: 10.1016/j.jclepro.2020.123853. URL https://doi.org/10.1016/j.jclepro.2020.123853.

S. Ravi and P. A. Balakrishnan. Modelling and control of an anfis temperature controller for plastic extrusion process. *2010 IEEE*

International Conference on Communication Control and Computing Technologies, ICCCCT 2010, (1):314–320, 2010. doi: 10.1109/ICCCCT.2010.5670572.

U. Sinthuja and S. Thavamani. Efficient employment of the mqtt and s-mqtt protocol in iot network. *International Journal of Advanced Science and Technology*, 28(17):858–864, 2019. URL https://www.scopus.com/inward/record.uri?eid=2-s2.0-85080145167&partnerID=40&md5=99dd65da22a441f53125de9f80a49145. cited By 0.

S. A. Thomas and J. Wiley. *SSL And TLS Essentials - Securing The Web (2000)*. 1999. URL papers://6b27418f-98b1-4a55-9d4a-81387a1148b8/Paper/p264.

S. Wang, J. Ying, Z. Chen, and Y. Feng. Packing pressure control for energy-saving servo injection molding based on fuzzy-PID controller. *ICMEE 2010 - 2010 2nd International Conference on Mechanical and Electronics Engineering, Proceedings*, 1(Icmee):34–38, 2010. doi: 10.1109/ICMEE.2010.5558599.

X. Zhang and H. Zhang. The accelerated life experiment study of Solid State Relay. *ICQR2MSE 2011 - Proceedings of 2011 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering*, pages 349–351, 2011. doi: 10.1109/ICQR2MSE.2011.5976628.