



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO  
FACULTAD DE INGENIERÍA

Desarrollo e integración de una  
arquitectura SIMATIC para la evaluación  
de estrategias de control

TESIS

Que para obtener el título de  
Ingeniero Eléctrico Electrónico

PRESENTA

Oscar Valle Xala

DIRECTOR DE TESIS

Dr. Hoover Mujica Ortega



Ciudad Universitaria, Cd. Mx., 2026



**PROTESTA UNIVERSITARIA DE INTEGRIDAD Y  
HONESTIDAD ACADÉMICA Y PROFESIONAL  
(Titulación con trabajo escrito)**



De conformidad con lo dispuesto en los artículos 87, fracción V, del Estatuto General, 68, primer párrafo, del Reglamento General de Estudios Universitarios y 26, fracción I, y 35 del Reglamento General de Exámenes, me comprometo en todo tiempo a honrar a la institución y a cumplir con los principios establecidos en el Código de Ética de la Universidad Nacional Autónoma de México, especialmente con los de integridad y honestidad académica.

De acuerdo con lo anterior, manifiesto que el trabajo escrito titulado DESARROLLO E INTEGRACION DE UNA ARQUITECTURA SIMATIC PARA LA EVALUACION DE ESTRATEGIAS DE CONTROL, que presenté para obtener el título de INGENIERO ELÉCTRICO ELECTRÓNICO es original, de mi autoría y lo realicé con el rigor metodológico exigido por mi Entidad Académica, citando las fuentes de ideas, textos, imágenes, gráficos u otro tipo de obras empleadas para su desarrollo.

En consecuencia, acepto que la falta de cumplimiento de las disposiciones reglamentarias y normativas de la Universidad, en particular las ya referidas en el Código de Ética, llevará a la nulidad de los actos de carácter académico administrativo del proceso de titulación.

---

**OSCAR VALLE XALA**  
Número de cuenta: 317342303



Jurado asignado

Presidente: M.I. Gloria Correa Palacios  
Secretario: Ing. Jeshua Perea Constantino  
Vocal: Dr. Hoover Mujica Ortega  
1<sup>er</sup> suplente: Ing. David Quintanar Villalba  
2<sup>do</sup> suplente: Ing. Luis Arturo Vélez Fajardo

Ciudad Universitaria, Departamento de Control y Robótica, Laboratorio de  
Automatización.

Ciudad de México.

Director de tesis

---

Dr. Hoover Mujica Ortega



## Dedicatoria

**A mis padres, Minerva y Oscar**, por su incondicional apoyo a lo largo de este camino, quienes, sin importar la razón, el momento o la circunstancia, siempre me han brindado su amor y respaldo, asegurándose de que nunca me faltara nada.

**A mi abuelo Francisco**, por haber sentado las bases de mis buenos hábitos de estudio y educación. Sé que estaría muy orgulloso de la persona en la que me he convertido.

**A los profesores** que me acompañaron a lo largo de mi desarrollo académico, no solo por compartir sus conocimientos, sino también por transmitirme su pasión por las materias que imparten.



## Agradecimientos

A mi profesor y asesor de tesis, el **Dr. Hoover Mujica Ortega**, por su confianza, amistad y la valiosa experiencia compartida. Su dedicación y gran compromiso fueron fundamentales para la realización de esta tesis.

**A mis amigos**, por hacer más ameno el tiempo entre clases, los proyectos y las incontables experiencias compartidas. Gracias por hacer que la carrera se convirtiera en una experiencia enriquecedora, llena de momentos gratos, memorables y recuerdos significativos.

A todos los miembros del **Laboratorio de Automatización**, por los valiosos conocimientos compartidos, así como por los momentos vividos durante este tiempo de aprendizaje y colaboración.

**A la Dra. Sofía Avila**, porque con su dedicación y amor por la enseñanza logró que me apasionara el área de control e inspiró a finalizar mi carrera.

**A la Universidad Nacional Autónoma de México** y a la **Facultad de Ingeniería**, que se convirtieron en mi segundo hogar, gracias a lo cual pude conocer a personas maravillosas como Mario, Jesús, Jessica, Mizraim, Allison, Mariana y Duque.



## Resumen

El presente trabajo tiene como objetivo desarrollar e implementar una arquitectura SIMATIC, entendida como la integración estructurada de *hardware*, *software*, señales, comunicación y rutinas de control basadas en equipos Siemens, orientada a la evaluación de estrategias de control en procesos industriales continuos y secuenciales. Asimismo, se busca diseñar, supervisar y validar rutinas de control bajo los lineamientos de la norma IEC 61131-3, proporcionando un entorno para consolidar habilidades orientadas a aplicaciones industriales.

Diversas investigaciones, como [National Academies of Sciences, Engineering, and Medicine, 2024] y [Supriya, *et al.*, 2024], indican que, en la transición de la Industria 4.0 hacia la Industria 5.0, persisten desafíos estructurales y tecnológicos, entre los que destacan la escasez de talento especializado, la fragmentación en los esquemas de formación y las limitaciones en la infraestructura de datos. Ante este panorama, las plataformas experimentales constituyen una alternativa efectiva para la evaluación de soluciones de automatización, ya que permiten validar técnicas de control en entornos controlados sin comprometer la operación de una planta industrial.

A partir de ello, se diseñó una arquitectura de control basada en el controlador SIMATIC S7-1500 y en el entorno de programación *Totally Integrated Automation* (TIA) Portal, integrando un sistema térmico para el análisis de control continuo y una mesa electroneumática para la validación de control secuencial. La propuesta contempla la especificación funcional de los procesos, la organización de variables, el desarrollo de diagramas eléctricos y la estandarización de conexiones.

En el proceso continuo, se implementaron y compararon dos controladores Proporcional-Integral-Derivativo (PID): el bloque tecnológico PID\_Compact y un PID nativo desarrollado a partir de su formulación en tiempo continuo, lo que permitió analizar el comportamiento dinámico del sistema. En el proceso secuencial, se validó la ejecución de una rutina de control ordenada y segura, mediante el monitoreo de cada etapa y la verificación del cumplimiento de las condiciones de transición establecidas. Esto permitió comprobar la correspondencia entre la lógica programada y la respuesta física de la mesa electroneumática, asegurando que la secuencia se desarrollara conforme al comportamiento esperado del sistema.

Los resultados obtenidos demuestran que la arquitectura desarrollada constituye una plataforma robusta, flexible y escalable, capaz de integrar y analizar estrategias de control en procesos de distinta naturaleza. Asimismo, se establece como una base tecnológica adecuada para futuras aplicaciones en automatización industrial, incluyendo su posible extensión hacia funciones de supervisión, monitoreo avanzado y análisis de desempeño.



# Índice general

Índice de figuras	xiii
Índice de tablas	xv
Acrónimos	xvii
<b>1. Introducción</b>	<b>1</b>
1.1. Motivación	2
1.2. Antecedentes	2
1.3. Formulación del problema	5
1.4. Objetivos	6
1.4.1. Objetivo general	6
1.4.2. Objetivos específicos	6
1.5. Contribuciones	6
1.6. Organización de la tesis	7
<b>2. Marco teórico</b>	<b>9</b>
2.1. Procesos industriales	9
2.1.1. Proceso continuo	9
2.1.2. Proceso por lotes	9
2.1.3. Proceso discreto	9
2.1.4. Proceso secuencial	9
2.2. Controlador PID	10
2.2.1. Introducción al PID	10
2.2.2. Fundamentos teóricos del control PID	10
2.2.3. Problemas comunes: <i>windup</i> y respuesta lenta	12
2.2.4. PID nativo	12
2.2.5. PID compact	13
2.3. Arquitectura SIMATIC	13
2.3.1. Introducción a la arquitectura SIMATIC	13
2.3.2. Controladores SIMATIC	14
2.3.3. SIMATIC S7-1500	15
2.3.4. Módulos de entradas y salidas (I/O) de funciones avanzadas	15
2.3.5. Software TIA Portal	15
2.4. Lenguajes de programación bajo el estándar IEC 61131-3	16
2.4.1. <i>Ladder</i> o lenguaje de escalera (LD)	16

2.4.2.	Diagrama de bloque de funciones (FBD) . . . . .	16
2.4.3.	Lenguaje de control estructurado (SCL) . . . . .	17
2.4.4.	Lenguaje de lista de instrucciones (STL) . . . . .	17
2.4.5.	Lenguaje para control secuencial (GRAPH) . . . . .	18
<b>3.</b>	<b>Desarrollo</b>	<b>19</b>
3.1.	Diseño e implementación de la arquitectura de control . . . . .	19
3.2.	Especificación funcional . . . . .	20
3.2.1.	Especificación funcional del sistema térmico . . . . .	20
3.2.2.	Especificación funcional de la mesa electroneumática . . . . .	22
3.3.	Tabla de ordenamiento de variables . . . . .	23
3.4.	Desarrollo de diagramas eléctricos mediante CAD . . . . .	25
3.5.	Conector Centronics . . . . .	27
3.6.	Diseño e implementación del sistema de control continuo . . . . .	30
3.6.1.	Rutina de control proceso continuo . . . . .	30
3.6.2.	Escalamientos . . . . .	30
3.6.3.	Rutina de control LD . . . . .	31
3.6.4.	Implementación del control PID Compact . . . . .	36
3.6.5.	Implementación del control PID nativo . . . . .	38
3.7.	Diseño e implementación del sistema de control secuencial . . . . .	40
3.7.1.	Rutina de control proceso secuencial . . . . .	40
<b>4.</b>	<b>Evaluación y discusión de resultados</b>	<b>47</b>
4.1.	Evaluación funcional del proceso continuo . . . . .	47
4.1.1.	PID Compact . . . . .	48
4.1.2.	PID nativo . . . . .	50
4.1.3.	PID Compact vs PID nativo . . . . .	51
4.2.	Evaluación funcional del proceso secuencial . . . . .	55
4.3.	Discusión . . . . .	60
<b>5.</b>	<b>Conclusiones y trabajo futuro</b>	<b>61</b>
5.1.	Conclusiones . . . . .	61
5.2.	Trabajo futuro . . . . .	62
	<b>Referencias</b>	<b>63</b>

# Índice de figuras

1.1. Evolución de la industria. . . . .	1
1.2. Plataforma educativa basada en un reactor químico. . . . .	3
1.3. Módulo final del centro de excelencia. . . . .	4
1.4. Entorno virtual desarrollado en Unity 3D. . . . .	5
2.1. Diagrama de bloques del control proporcional en el dominio del tiempo. . . . .	11
2.2. Diagrama de bloques del control integral en el dominio del tiempo. . . . .	11
2.3. Diagrama de bloques del control derivativo en el dominio del tiempo. . . . .	12
2.4. Controladores SIMATIC. . . . .	14
2.5. Diferentes versiones de la familia SIMATIC S7-1500. . . . .	15
2.6. Ejemplo del lenguaje de programación Ladder. . . . .	16
2.7. Ejemplo del lenguaje de programación FBD. . . . .	17
2.8. Ejemplo del lenguaje de programación SCL. . . . .	17
2.9. Ejemplo del lenguaje de programación STL. . . . .	18
2.10. Ejemplo del lenguaje de programación GRAPH. . . . .	18
3.1. Arquitectura de control. . . . .	20
3.2. Sistema térmico. . . . .	21
3.3. Mesa electroneumática. . . . .	22
3.4. Diagrama de conexión del módulo de entradas digitales. . . . .	25
3.5. Diagrama de conexión del módulo de salidas digitales. . . . .	26
3.6. Diagrama de conexión del módulo de entradas analógicas. . . . .	26
3.7. Diagrama de conexión del módulo de salidas analógicas. . . . .	27
3.8. Conector Centronics de 36 pines utilizado para la conexión estandarizada de procesos. . . . .	28
3.9. Bloque que invoca la subrutina Escalamiento. . . . .	30
3.10. Peldaño 1 dentro de la subrutina Escalamientos. . . . .	30
3.11. Peldaño 2 dentro de la subrutina Escalamientos. . . . .	31
3.12. Peldaño 1 de la rutina LD. . . . .	31
3.13. Peldaño 2 de la rutina LD. . . . .	32
3.14. Peldaño 3 de la rutina LD. . . . .	32
3.15. Peldaño 4 de la rutina LD. . . . .	32
3.16. Peldaño 5 de la rutina LD. . . . .	33
3.17. Peldaño 6 de la rutina LD. . . . .	33
3.18. Peldaño 7 de la rutina LD. . . . .	33
3.19. Peldaño 8 de la rutina LD. . . . .	34

3.20. Peldaño 9 de la rutina LD. . . . .	34
3.21. Peldaño 10 de la rutina LD. . . . .	34
3.22. Peldaño 11 de la rutina LD. . . . .	34
3.23. Peldaño 12 de la rutina LD. . . . .	35
3.24. Peldaño 13 de la rutina LD. . . . .	35
3.25. Peldaño 14 de la rutina LD. . . . .	35
3.26. Peldaño 15 de la rutina LD. . . . .	35
3.27. Bloque PID Compact y sus configuraciones en TIA Portal. . . . .	36
3.28. Configuraciones basicas del PID Compact. . . . .	36
3.29. Configuraciones de las variables de proceso del PID Compact. . . . .	37
3.30. Configuraciones avanzadas del PID Compact. . . . .	37
3.31. Parametros del PID Compact. . . . .	38
3.32. Peldaño 1 de la rutina LD del proceso secuencial. . . . .	40
3.33. Peldaño 2 de la rutina LD del proceso secuencial. . . . .	40
3.34. Peldaño 3 de la rutina LD del proceso secuencial. . . . .	41
3.35. Vista general de la rutina hecha en GRAPH. . . . .	41
3.36. Paso 1 del proceso secuencial. . . . .	42
3.37. Transición 1 del proceso secuencial. . . . .	42
3.38. Paso 2 del proceso secuencial. . . . .	42
3.39. Transición 2 del proceso secuencial. . . . .	43
3.40. Paso 3 del proceso secuencial. . . . .	43
3.41. Transición 3 del proceso secuencial. . . . .	43
3.42. Paso 4 del proceso secuencial. . . . .	43
3.43. Transición 4 del proceso secuencial. . . . .	44
3.44. Paso 5 del proceso secuencial. . . . .	44
3.45. Transición 5 del proceso secuencial. . . . .	44
3.46. Paso 6 del proceso secuencial. . . . .	44
3.47. Transición 6 del proceso secuencial. . . . .	45
3.48. Paso 7 del proceso secuencial. . . . .	45
3.49. Transición 7 del proceso secuencial. . . . .	45
4.1. Gráfica PID Compact. . . . .	50
4.2. Gráfica PID Nativo. . . . .	51
4.3. Gráfica comparativa entre ambos controladores. . . . .	52
4.4. Acercamiento de la Figura 4.3 en 80 °C. . . . .	52
4.5. Acercamiento de la Figura 4.3 en 60 °C. . . . .	53
4.6. Acercamiento de la Figura 4.3 en 70 °C. . . . .	53
4.7. Acercamiento de la Figura 4.3 en 80 °C. . . . .	54
4.8. Acercamiento de la Figura 4.3 en la señal del actuador. . . . .	55
4.9. Validación de la secuencia de salida del pistón 1. . . . .	56
4.10. Validación de la secuencia de contracción del pistón 1. . . . .	56
4.11. Validación de la secuencia de salida del pistón 2. . . . .	57
4.12. Validación de la secuencia de contracción del pistón 2. . . . .	58
4.13. Validación de la secuencia de expulsión del pistón 3. . . . .	59
4.14. Validación de la secuencia de contracción del pistón 3. . . . .	59

# Índice de tablas

3.1. Tabla de asignación de entradas digitales. . . . .	23
3.2. Tabla de asignación de salidas digitales. . . . .	24
3.3. Tabla de asignación de entradas analógicas. . . . .	24
3.4. Tabla de asignación de salidas analógicas. . . . .	24
3.5. Distribución de pines del conector Centronics . . . . .	29
4.1. Comparación de indicadores de desempeño de ambos controladores . . . . .	60



# Acrónimos

- TIA Portal** Totally Integrated Automation. ix, 2, 13, 14, 15, 16, 19, 30, 36, 61
- PID** Controlador Proporcional, Integral y Derivativo. ix, 2, 3, 7, 10, 12, 13, 31, 34, 35, 36, 37, 38, 47, 48, 50, 51, 52, 53, 54, 60, 61
- PLC** Controlador Lógico Programable. 1, 2, 6, 10, 19, 23, 25, 26, 27, 28, 30
- HMI** Interfaz Humano Máquina. 3
- I/O** Entradas y salidas. 3, 6, 15, 23, 25, 61
- PV** Variable de proceso. 10, 36, 38
- SP** Valor deseado. 10, 13, 34, 35, 38, 47, 48, 49, 50, 51, 52, 53, 54
- LD** Lenguaje de escalera (ladder). 16, 30, 42
- FBD** Diagrama de bloque de funciones. 16
- SCL** Lenguaje de control estructurado. 17
- STL** Lenguaje de lista de instrucciones. 17
- GRAPH** Lenguaje de programación gráfica. 18, 30, 40, 41, 45, 55



# Capítulo 1

## Introducción

Históricamente, la industria se ha enfrentado a constantes transformaciones impulsadas por la evolución tecnológica (ver Figura 1.1). Como señalan en [Vinitha, *et al.*, 2020], la evolución industrial constituye un pilar fundamental del desarrollo mundial, cuyo crecimiento se ha intensificado desde el siglo XVIII. Esto se evidenció con la Primera Revolución Industrial, caracterizada por la incorporación del carbón como fuente de energía y el uso del motor a vapor, lo que permitió mecanizar procesos que antes dependían principalmente del trabajo manual. Posteriormente, la Segunda Revolución Industrial estuvo marcada por la electrificación y la producción en masa. Por su parte, la Tercera Revolución Industrial introdujo la automatización mediante sistemas electrónicos, Controladores Lógicos Programables (PLC, por sus siglas en inglés) y tecnologías de computación.

Actualmente, la Industria 4.0 se desarrolla en un entorno digital impulsado por tecnologías como el Internet de las Cosas (IoT, por sus siglas en inglés), *Big Data* e Inteligencia Artificial, que optimizan la supervisión y la toma de decisiones en tiempo real. A su vez, se transita hacia la Industria 5.0, orientada a la colaboración entre humanos y robots, con un enfoque en la personalización, la sostenibilidad y el bienestar humano [Supriya, *et al.*, 2024]. Este nuevo paradigma exige arquitecturas de control más flexibles, escalables y adaptables, capaces de responder a las nuevas dinámicas que caracterizan a la industria moderna.

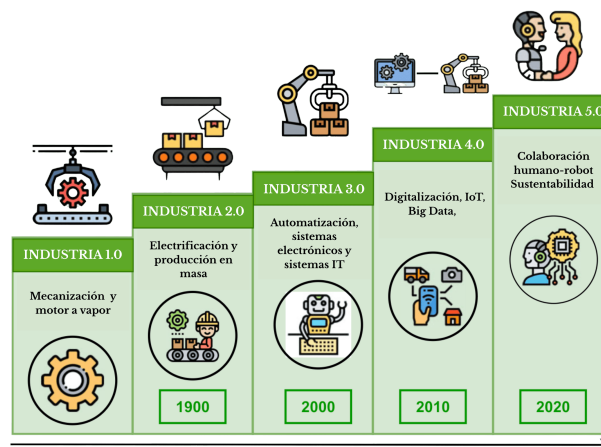


Figura 1.1 Evolución de la industria.<sup>1.1</sup>

<sup>1.1</sup>Tomada de [Supriya, *et al.*, 2024].

Es por eso que diversas instituciones educativas han desarrollado módulos de demostración y plataformas experimentales, con el objetivo de enfrentar los retos que surgen en la transición de la Industria 4.0 hacia la Industria 5.0. Estas iniciativas permiten probar e implementar tecnologías emergentes en entornos controlados, facilitando la evaluación de estrategias de control, la integración de sistemas inteligentes y la formación de profesionales capacitados para responder a las nuevas exigencias del sector industrial.

## 1.1. Motivación

La motivación de este trabajo surge de la necesidad de disponer de una plataforma que fortalezca el aprendizaje y la evaluación de estrategias de control aplicadas a la automatización industrial, particularmente en arquitecturas modernas basadas en PLC para entornos de manufactura. Esto se aborda mediante el desarrollo de una arquitectura SIMATIC orientada a este tipo de procesos, en los cuales predominan el control continuo como el control secuencial, debido a que gran parte de las operaciones industriales requieren, por un lado, la regulación permanente de variables físicas como temperatura, presión, nivel o flujo y, por otro, la coordinación lógica y ordenada de actuadores, etapas de producción y secuencias de operación dentro de sistemas automatizados.

Este trabajo de tesis busca fortalecer la comprensión y aplicación de técnicas de control mediante el uso de herramientas industriales, como la arquitectura SIMATIC S7-1500 y el entorno TIA Portal. Las principales ventajas de realizar este trabajo son:

- Permite implementar y evaluar estrategias de control PID, tanto en su versión Compact como nativa, facilitando la comparación de su desempeño en un entorno práctico.
- Favorece la integración de sistemas de control continuo y secuencial, lo que refleja de manera más realista las condiciones presentes en procesos industriales.
- Proporciona experiencia en el uso de herramientas industriales ampliamente utilizadas, como los controladores SIMATIC S7-1500 y el entorno de programación TIA Portal.
- Facilita la comprensión del diseño e implementación de arquitecturas de control, incluyendo la elaboración de diagramas eléctricos y la especificación funcional de los sistemas.
- Contribuye al desarrollo de habilidades prácticas en automatización, permitiendo al usuario interactuar con un sistema completo desde su modelado hasta su implementación y evaluación.

## 1.2. Antecedentes

La implementación de arquitecturas de control resulta relevante en el ámbito académico, ya que permite a los estudiantes familiarizarse con los equipos, protocolos de comunicación y tecnologías utilizadas en entornos industriales. Este acercamiento fortalece los conocimientos necesarios para afrontar los desafíos actuales de la Industria 4.0, así como aquellos que surjan con la transición hacia la Industria 5.0, donde la integración entre sistemas físicos e inteligentes demanda un nivel de preparación más avanzado.

En este contexto, en 2016 la Universidad de Cambridge desarrolló el sistema Cambridge Weblabs, una plataforma educativa basada en tecnología industrial con acceso remoto [Botero, *et al.*, 2016]. Este

desarrollo se estructuró en torno a un reactor químico (ver Figura 1.2), operado mediante el sistema de control distribuido SIMATIC PCS 7 de Siemens, lo que permitió a los estudiantes monitorear y controlar procesos desde cualquier ubicación con conexión a Internet. Su objetivo principal fue ofrecer una herramienta de aprendizaje en contextos donde el acceso presencial a laboratorios es limitado, ya sea por espacio o por disponibilidad de equipos. Por ello, a través de una interfaz gráfica realista, los estudiantes interactúan con el sistema, realizan experimentos relacionados con la dinámica de procesos y llevan a cabo la sintonización de controladores PID. De este modo, se familiarizan con tecnologías como el monitoreo remoto, la virtualización de procesos, el uso de Interfaz Humano Máquina (HMI, por sus siglas en inglés) y la integración de sistemas distribuidos para el control de plantas en tiempo real.

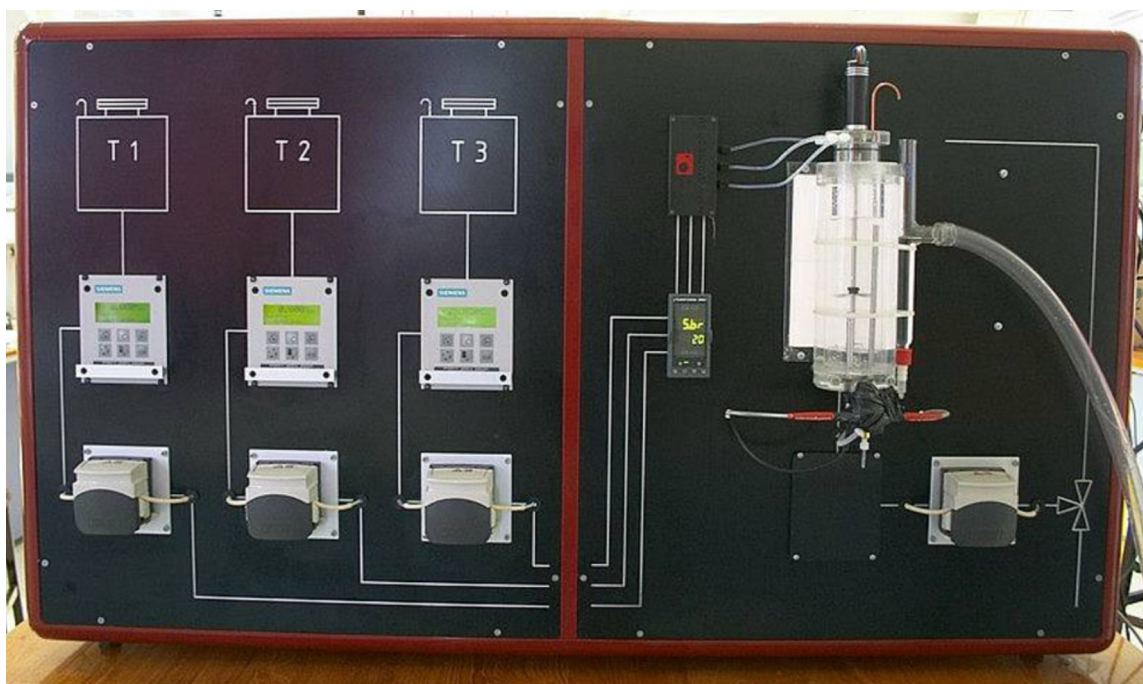


Figura 1.2 Plataforma educativa basada en un reactor químico.<sup>1,2</sup>

De forma similar, en 2022 la Royal University of Phnom Penh, en Camboya, desarrolló un laboratorio como centro de excelencia orientado a la Industria 4.0 [Kuong, *et al.*, 2023]. La Figura 1.3 muestra el módulo final desarrollado. La arquitectura está conformada por el controlador Siemens S7-1500, módulos de entradas y salidas (I/O, por sus siglas en inglés), sistemas HMI y variadores de velocidad. Este laboratorio fue diseñado con fines formativos en el área de automatización industrial, lo que permitió la realización de proyectos piloto. Uno de ellos consistió en un sistema de elevador automatizado, el cual sirvió como caso práctico para el desarrollo de rutinas de control y monitoreo industrial. Los resultados obtenidos demostraron que la arquitectura es capaz de representar un entorno de producción automatizada, confirmando su utilidad como herramienta de formación académica.

<sup>1,2</sup>Tomada de [Botero, *et al.*, 2016]



Figura 1.3 Módulo final del centro de excelencia.<sup>1,3</sup>

Por último, la Universidad Politécnica Salesiana implementó una arquitectura de control orientada al diseño y simulación de procesos industriales mediante un entorno virtual en 3D, conformada por el controlador SIMATIC S7-1500 y Unity 3D [Endara y Rodas, 2024]. Este entorno replica virtualmente estaciones de producción y sistemas de control de movimiento asociados a un proceso de clasificación por cinta transportadora, proporcionando a los estudiantes condiciones operativas similares a las de un entorno industrial real. Como resultado, los alumnos pueden interactuar con procesos virtuales sin necesidad de contar físicamente con los equipos, lo que amplía el acceso a recursos formativos y fortalece su preparación para el ámbito profesional. En la Figura 1.4 se observa el controlador y el entorno virtual desarrollado.

---

<sup>1,3</sup>Tomada de [Kuong, *et al.*, 2023]



Figura 1.4 Entorno virtual desarrollado en Unity 3D.<sup>1.4</sup>

### 1.3. Formulación del problema

Nos encontramos en una etapa de transición de la Industria 4.0 hacia la Industria 5.0. Esta transformación plantea exigencias cada vez mayores sobre los sistemas de automatización industrial, los cuales deben evolucionar para ser más inteligentes, adaptables, colaborativos y compatibles con entornos altamente interconectados.

En este contexto, la automatización eficiente de procesos continuos y secuenciales se vuelve esencial para garantizar la calidad, estabilidad y productividad de los sistemas industriales modernos.

No obstante, existen dos principales limitantes para la mejora o validación de estos sistemas de control:

- La imposibilidad de implementar nuevas técnicas directamente a nivel de planta, ya que detener un proceso para ajustar o evaluar rutinas representa un riesgo operativo y económico considerable.

---

<sup>1.4</sup>Tomada de [Endara y Rodas, 2024]

- Las restricciones para la capacitación del personal en condiciones reales, ya que cualquier error durante este proceso podría derivar en fallos críticos del sistema o comprometer la seguridad del entorno operativo.

Estas problemáticas se enmarcan dentro de los desafíos actuales de la manufactura inteligente. En particular, [National Academies of Sciences, Engineering, and Medicine, 2024] identifica como retos principales del sector la escasez de talento especializado, la fragmentación en los esquemas de formación, las dificultades para integrar tecnologías de operación (OT) y tecnologías de la información (IT), así como la persistencia de datos aislados en sistemas que dificultan su uso para supervisión, control y optimización.

Ante este panorama, se hace necesario el desarrollo de una arquitectura de control que permita simular, validar y ajustar estrategias de automatización continua y secuencial sin comprometer la operación del proceso industrial.

## 1.4. Objetivos

### 1.4.1. Objetivo general

Consolidar una arquitectura SIMATIC, flexible y modular, que permita la implementación y evaluación de técnicas de control para procesos continuos y secuenciales. Asimismo, establecer una comunicación eficaz, sistemática y confiable entre los dispositivos del sistema, garantizando la estandarización mediante el uso de rutinas de control programadas bajo los lineamientos de la norma IEC 61131-3.

### 1.4.2. Objetivos específicos

- Configurar el controlador SIMATIC S7-1500, así como los módulos de entrada y salida, la red de comunicación y los elementos necesarios para la implementación de estrategias de control en procesos continuos y secuenciales.
- Desarrollar el levantamiento técnico y diagramas de conexión con base en dibujos paramétricos utilizando herramientas de diseño asistido por computadora (CAD).
- Diseñar y evaluar rutinas de control de procesos secuenciales y continuos.

## 1.5. Contribuciones

- Se generaron diagramas de conexiones eléctricas que describen la distribución e interconexión de todos los módulos I/O del PLC, permitiendo una visualización clara y precisa de la arquitectura eléctrica del sistema.
- Se elaboró una tabla de distribución de pines para el conector Centronix, con el propósito de estandarizar las conexiones entre dispositivos y facilitar futuras labores de cableado, mantenimiento y expansión del sistema.
- Se diseñó e implementó una arquitectura de control confiable para la evaluación, simulación y programación de técnicas de control aplicadas a procesos secuenciales y continuos, garantizando flexibilidad en su configuración y adaptabilidad a distintas estrategias de automatización.

## 1.6. Organización de la tesis

El contenido de esta tesis se organiza en cinco capítulos. En este primero se presenta un panorama general del planteamiento del problema, así como de los objetivos a alcanzar, además del contexto y la problemática en los que se desarrolla este trabajo. A continuación, se describe el contenido de los capítulos restantes:

- En el Capítulo 2 se presenta el marco teórico del trabajo, abordando los fundamentos de los procesos industriales, el controlador PID y la arquitectura SIMATIC con sus principales componentes y herramientas de programación.
- En el Capítulo 3 se describe el desarrollo del sistema, incluyendo el diseño e implementación de la arquitectura de control, la especificación funcional de los sistemas, la elaboración de diagramas eléctricos y la implementación de las rutinas de control.
- La evaluación y discusión de resultados se presenta en el Capítulo 4, abordando la evaluación funcional del sistema mediante controladores PID, tanto en su versión Compact como nativa, así como el análisis de los resultados obtenidos del sistema secuencial.
- Por último, en el Capítulo 5 se presentan las conclusiones del trabajo realizado, así como las propuestas de trabajo futuro derivadas de los resultados obtenidos.



# Capítulo 2

## Marco teórico

### 2.1. Procesos industriales

En términos simples, los procesos industriales comprenden el conjunto de actividades técnicas y operativas destinadas a transformar materias primas en productos terminados. Entre sus principales tipos se encuentran los procesos continuos, por lotes, discretos y secuenciales [ISA, 1995].

#### 2.1.1. Proceso continuo

El proceso continuo es un modo de operación en el que el sistema de producción recibe un flujo constante de materias primas, las transforma mediante una secuencia específica de operaciones sin interrupciones y genera productos de forma continua. En su forma ideal, este tipo de proceso opera en estado estacionario, es decir, con flujos de entrada y salida que se mantienen constantes a lo largo del tiempo [Nagy, *et al.*, 2020]. Este tipo de operación es común en sectores industriales como el alimentario, químico, petroquímico, farmacéutico y el procesamiento de minerales.

#### 2.1.2. Proceso por lotes

Los procesos por lotes se refieren a la producción de cantidades finitas de material (lotes), mediante una secuencia definida de acciones de procesamiento aplicadas sobre ciertas cantidades de materia prima, utilizando uno o más equipos. El resultado de este tipo de proceso se conoce como lote. Este tipo de proceso se considera discontinuo, ya que no es completamente discreto ni continuo, aunque presenta características de ambos [ISA, 1995].

#### 2.1.3. Proceso discreto

Los procesos discretos son sistemas de pequeña a mediana escala que emplean una combinación de máquinas que operan de forma casi independiente entre sí. Se caracterizan por ejecutar operaciones secuenciales y ensamblajes de componentes para fabricar productos individuales e identificables, como automóviles, aviones, dispositivos biomédicos y equipos electrónicos [Müller y Oehm, 2019].

#### 2.1.4. Proceso secuencial

Un proceso secuencial es un conjunto de etapas, tareas o instrucciones que se ejecutan en un orden específico y predefinido, donde cada paso debe completarse antes de iniciar el siguiente.

Este tipo de procesos se caracteriza por una ejecución paso a paso, en la que el resultado de una etapa condiciona el inicio de la siguiente. Para su implementación, se utilizan diagramas de funciones secuenciales, también conocidos como secuenciadores. Estos facilitan la representación lógica del flujo operativo mediante estados y transiciones dependientes de condiciones o eventos definidos, garantizando así una secuencia clara y estructurada [Siemens, 2020].

## 2.2. Controlador PID

### 2.2.1. Introducción al PID

Un controlador PID se compone de tres acciones de control: Proporcional, Integral y Derivativa. De acuerdo con [Åström y Hägglund, 1995], el controlador PID cumple varias funciones importantes: proporciona retroalimentación, elimina las desviaciones en estado estacionario mediante la acción integral y anticipa el futuro mediante la acción derivativa.

Como menciona [Barrios, 2022], este tipo de controlador mantiene una variable de proceso (PV), como la temperatura, la velocidad o la presión, en un valor deseado (SP), utilizando un algoritmo que calcula continuamente la diferencia entre el SP y el valor actual de la PV, denominada error, y ajusta la salida del controlador para minimizar dicha diferencia.

Según [Lozano Valencia, *et al.*, 2012] y [Ramírez Urrego y Posada Restrepo, 2016], el controlador PID continúa siendo una de las estrategias de control más utilizadas tanto en la industria como en el ámbito académico, a pesar del desarrollo de técnicas de control más avanzadas y del creciente uso de métodos basados en inteligencia artificial. En particular, [Lozano Valencia, *et al.*, 2012, p. 35] señala que “más del 90 % de los lazos de control utilizan control PID, dado que es una estrategia simple, efectiva y no requiere una gran fundamentación teórica para su utilización en los procesos cotidianos”.

Por lo anterior, este tipo de controladores resultan suficientes para resolver una gran cantidad de problemas prácticos de control, razón por la cual se encuentran ampliamente distribuidos en diversos sectores industriales.

### 2.2.2. Fundamentos teóricos del control PID

El controlador PID puede representarse matemáticamente tanto en el dominio del tiempo como en el dominio de Laplace. La representación en el dominio de Laplace es comúnmente utilizada para el análisis de sistemas lineales y el estudio de estabilidad mediante funciones de transferencia. Por otro lado, la representación en el dominio del tiempo describe directamente la relación entre la señal de error y la acción de control en función del tiempo, lo cual resulta especialmente útil para su implementación digital.

En este trabajo se empleará la representación en el dominio del tiempo, dado que el controlador se implementa en un PLC, donde la acción de control se calcula en función del error instantáneo y sus variaciones temporales.

Para comprender el fundamento teórico del controlador, es necesario analizar de manera individual cada una de sus tres acciones: proporcional, integral y derivativa. Cada una de estas acciones desempeña un papel específico en la regulación de la variable de proceso, contribuyendo a la estabilidad, precisión y rapidez de respuesta del sistema de control.

**Control Proporcional (P):** Genera una señal de control directamente proporcional al error. Su expresión matemática se presenta en la Ecuación (2.1), donde  $e(t)$  representa el error entre la referencia y la variable de proceso, y  $K_p$  es la ganancia proporcional.

$$u(t) = K_p e(t) \quad (2.1)$$

La representación de esta relación mediante diagrama de bloques se muestra en la Figura 2.1, ahí se observa que el error  $e(t)$  es multiplicado por la constante  $K_p$  para generar la señal de control  $u(t)$ .

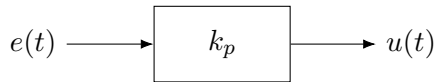


Figura 2.1 Diagrama de bloques del control proporcional en el dominio del tiempo.<sup>2.1</sup>

Un incremento en el valor de  $K_p$  aumenta la rapidez de respuesta del sistema; sin embargo, valores excesivamente altos pueden provocar oscilaciones o incluso inestabilidad.

**Control Integral (I):** Genera una señal de control basada en la acumulación temporal del error, con el propósito de eliminar el error en estado estacionario. Este modo de control realiza un ajuste progresivo de la acción de control mientras el error persiste en el tiempo. Su Ecuación matemática (2.2) se expresa como:

$$u(t) = k_i \int_0^t e(t) dt \quad (2.2)$$

La representación de esta relación mediante un diagrama de bloques se muestra en la Figura 2.2, donde se observa que la señal de error  $e(t)$  es multiplicada por un integrador con ganancia  $K_i$ . Es decir, el error es multiplicado por la constante  $K_i$  y posteriormente integrado en el tiempo, generando así la señal de control  $u(t)$ .

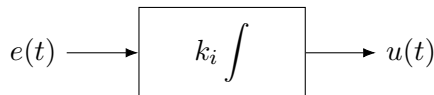


Figura 2.2 Diagrama de bloques del control integral en el dominio del tiempo.<sup>2.2</sup>

Esta característica permite eliminar el error residual que puede mantenerse cuando la referencia es constante. De este modo, el controlador incrementa o disminuye progresivamente la señal de control hasta que el error tiende a cero. Esto evita la necesidad de aumentar excesivamente la ganancia proporcional  $K_p$ , lo cual podría deteriorar el margen de estabilidad del sistema y provocar un comportamiento oscilatorio o incluso inestabilidad.

**Control Derivativo (D):** Genera una señal de control proporcional a la velocidad de cambio del error, anticipando su comportamiento futuro y contribuyendo a mejorar la estabilidad y la respuesta transitoria del sistema Ecuación (2.3). Esta acción se ve representada como:

$$u_c(t) = k_d \frac{de(t)}{dt} \quad (2.3)$$

<sup>2.1</sup>Elaboración propia.

<sup>2.2</sup>Elaboración propia.

La representación de esta relación mediante un diagrama de bloques se muestra en la Figura 2.3, donde se observa que la señal de error  $e(t)$  es aplicada a un bloque derivativo con ganancia  $K_d$ . En este caso, el error es multiplicado por la constante  $K_d$  y posteriormente derivado con respecto al tiempo, generando así la señal de control  $u(t)$ .

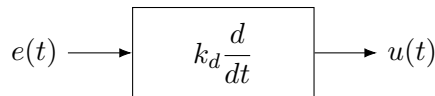


Figura 2.3 Diagrama de bloques del control derivativo en el dominio del tiempo.<sup>2,3</sup>

Una vez definidas las tres acciones del controlador y se estableció el propósito de cada una, la expresión del control PID puede formularse mediante la combinación de las Ecuaciones (2.1), (2.2) y (2.3), teniendo como resultado la ecuación (2.4).

$$u_c(t) = K_p e(t) + K_i \int e(t) dt + k_d \frac{de(t)}{dt} \quad (2.4)$$

La ecuación del control PID se encuentra expresada en forma de ganancias independientes, lo que implica que cada término (proporcional, integral y derivativo) actúa de manera individual sobre la señal de error, sin modificar directamente a las otras.

### 2.2.3. Problemas comunes: *windup* y respuesta lenta

Un problema recurrente en este tipo de sistemas surge al inicio de la operación, donde puede aparecer el fenómeno conocido como *windup*. Este se produce cuando la salida del controlador alcanza los límites de saturación del actuador, impidiendo que la señal de control continúe ajustándose de acuerdo al error del sistema. Como consecuencia, el término integral se acumula de manera excesiva, lo que puede provocar un sobreimpulso significativo y un incremento en el tiempo de recuperación del sistema. [Visioli, 2006]

Por otro lado, una respuesta lenta en el sistema suele estar asociada a una sintonización inadecuada de los parámetros del controlador, manifestándose en un incremento del tiempo de subida y del tiempo de establecimiento, lo que limita la rapidez con la que se alcanza el valor de referencia.

### 2.2.4. PID nativo

Llamaremos PID nativo al algoritmo implementado y desarrollado para realizar la comparación en el Capítulo 4. Este algoritmo se expresa en una forma equivalente de ganancias dependientes y parte de la Ecuación general del controlador (2.4). A continuación, se desarrolla el procedimiento que permite obtener dicha representación.

Partiendo de la Ecuación (2.4) con ganancias independientes, se procede a reorganizar algebraicamente la expresión con el objetivo de obtener su forma equivalente en términos de ganancias dependientes.

**Paso 1: Factorizar la ganancia proporcional  $K_p$ :** Donde se extrae  $K_p$  como factor común de (2.4).

<sup>2,3</sup>Elaboración propia.

$$u(t) = K_p \left[ e(t) + \frac{K_i}{K_p} \int^t e(t) dt + \frac{K_d}{K_p} \frac{de(t)}{dt} \right] \quad (2.5)$$

**Paso 2: Definir parámetros dependientes.** Se introducen parámetros equivalentes (constantes de tiempo) a partir de las razones entre ganancias:

$$T_i \triangleq \frac{K_p}{K_i}, \quad (2.6)$$

$$T_d \triangleq \frac{K_d}{K_p} \quad (2.7)$$

**Paso 3: Sustituir en la ecuación factorizada.** Dado que  $\frac{K_i}{K_p} = \frac{1}{T_i}$  y  $\frac{K_d}{K_p} = T_d$ , al sustituir en (2.5) se obtiene:

$$u(t) = K_p \left[ e(t) + \frac{1}{T_i} \int^t e(t) dt + T_d \frac{de(t)}{dt} \right] \quad (2.8)$$

La expresión (2.8) corresponde a la forma del controlador PID con ganancias dependientes, ya que  $T_i$  y  $T_d$  quedan definidos en función de  $K_P$  y de las ganancias originales  $K_i$  y  $K_d$ , respectivamente.

### 2.2.5. PID compact

El bloque `PID_Compact` de Siemens es un objeto tecnológico del *software* TIA Portal, diseñado para implementar de manera sencilla la función de control PID en los controladores SIMATIC.

Conforme a la documentación de [Siemens, 2023], el `PID_Compact` implementa un algoritmo de control continuo con funciones de optimización automática (*autotuning*). Este bloque permite realizar una sintonización en dos etapas: la optimización inicial (*pretuning*), que determina los parámetros a partir de un cambio brusco en la salida, y la optimización fina (*fine tuning*), que perfecciona la respuesta ante variaciones menores. El bloque adquiere cíclicamente el valor medido (*input*), comparándolo con el SP para determinar el error de control y calcular una señal de salida para estabilizar la variable de proceso con una respuesta rápida y sin error en estado estacionario.

La instrucción `PID_Compact` calcula automáticamente los parámetros proporcional, integral y derivativo del sistema controlado durante la fase de preajuste (*pretuning*). Posteriormente, es posible realizar un ajuste fino (*fine tuning*) para optimizar la respuesta del controlador, sin necesidad de determinar manualmente los parámetros.

## 2.3. Arquitectura SIMATIC

### 2.3.1. Introducción a la arquitectura SIMATIC

De acuerdo con [Siemens, 2024b], la compañía adoptó el uso de transistores en los circuitos de sus controladores en 1958. Ese mismo año, SIMATIC (término derivado de Siemens y *Automatic*) fue registrado como marca comercial. Los nuevos módulos SIMATIC, basados en transistores de germanio, revolucionaron el mercado a partir de ese mismo año. El primer dispositivo lanzado bajo este nombre fue el SIMATIC G.

Hoy en día, SIMATIC es una línea de productos de Siemens que ofrece soluciones integrales para la automatización industrial, abarcando desde pequeños sistemas de control hasta complejas arquitecturas

de automatización. Los productos SIMATIC se utilizan en diversas industrias, como la automotriz, la alimentaria y el sector energético. Gracias a su flexibilidad, escalabilidad y compatibilidad con tecnologías de la Industria 4.0, SIMATIC sigue siendo un referente en el ámbito de la automatización industrial.

### 2.3.2. Controladores SIMATIC

La gama de controladores SIMATIC (Figura 2.4) comprende controladores básicos, avanzados, distribuidos y de *software*, que ofrecen una escalabilidad impresionante e integración de sus funciones. El *software* de TIA Portal permite desarrollar soluciones completas de automatización industrial [Siemens, 2024a].

- **Controladores básicos:** Diseñados para soluciones de automatización compacta con funciones integradas de comunicación. Están disponibles en versiones estándar y de seguridad, siendo el SIMATIC S7-1200 el modelo principal dentro de esta categoría.
- **Controladores avanzados:** Automatizan no solo las plantas de producción completas, sino también las aplicaciones que exigen el mayor rendimiento, flexibilidad y capacidad de red. SIMATIC S7-300, S7-400 y S7-1500 son los modelos dentro de esta categoría.
- **Controladores distribuidos:** Se utilizan para máquinas con una arquitectura distribuida y para máquinas en serie con espacio limitado disponible.



Figura 2.4 Controladores SIMATIC.<sup>2.4</sup>

Por último, tenemos **TIA Portal**, la herramienta central para gestionar los datos de automatización y los editores asociados dentro de un proyecto. Los componentes SIMATIC se configuran, parametrizan y programan mediante el *software* de ingeniería STEP 7, integrado en TIA Portal, lo que permite una gestión eficiente y unificada del sistema de automatización.

---

<sup>2.4</sup>Tomada de [Siemens, 2019]

### 2.3.3. SIMATIC S7-1500

La familia SIMATIC S7-1500 (ver Figura 2.5) es un sistema de automatización modular diseñado para rangos de rendimiento medio y alto. Las diferentes versiones de los controladores permiten ajustar el rendimiento a la aplicación específica. Dependiendo de los requisitos, el controlador programable puede ampliarse con módulos de I/O para señales digitales y analógicas, así como con módulos de tecnología y comunicación [Berger, 2014].



Figura 2.5 Diferentes versiones de la familia SIMATIC S7-1500.<sup>2.5</sup>

### 2.3.4. Módulos de entradas y salidas (I/O) de funciones avanzadas

El controlador SIMATIC S7-1500 se complementa con una variedad de módulos I/O, tanto de señales digitales como analógicas, permitiendo su adaptación a distintas aplicaciones industriales. Estos módulos amplían su capacidad de procesamiento y permiten la integración de sensores y actuadores para el control y supervisión de procesos.

El S7-1500 admite hasta 32 módulos de expansión, lo que le otorga una gran flexibilidad para escalar el sistema según las necesidades de la aplicación. Esta modularidad permite configurar el controlador de manera óptima, asegurando un alto rendimiento en entornos industriales exigentes.

De acuerdo con el manual de sistema de [Siemens, 2023], los módulos de periferia constituyen la interfaz entre el controlador y el proceso. A través de los sensores y actuadores conectados, el controlador adquiere información del proceso y genera las acciones de control correspondientes.

### 2.3.5. Software TIA Portal

TIA Portal es la plataforma de ingeniería unificada de Siemens, cuyo propósito es integrar todas las tareas de automatización (control, visualización, movimiento, seguridad y diagnóstico) en una sola interfaz. Por lo tanto, esta constituye la herramienta central para gestionar los datos de automatización y los editores asociados dentro de un proyecto. Los componentes SIMATIC se configuran, parametrizan

---

<sup>2.5</sup>Tomada de [Berger, 2014]

y programan mediante el *software* de ingeniería STEP 7, integrado en el entorno TIA Portal, lo que permite una gestión eficiente y unificada del sistema de automatización.

En [Siemens, 2024], se define SIMATIC STEP 7, como el software de ingeniería para configurar la familia de controladores SIMATIC S7-1200, S7-1500, S7-300/400 y los controladores por *software* (*WinAC*).

Hay disponibles dos ediciones de STEP 7, dependiendo de las familias de controladores que se desee configurar:

- **STEP 7 Basic:** Destinada a la configuración de controladores SIMATIC S7-1200.
- **STEP 7 Professional:** Permite la configuración de controladores SIMATIC S7-1200, S7-1500, S7-300/400 y controladores por *software* (*WinAC*).

## 2.4. Lenguajes de programación bajo el estándar IEC 61131-3

TIA Portal permite seleccionar entre cinco lenguajes de programación para el desarrollo del proyecto, lo que proporciona al programador la flexibilidad necesaria para elegir el enfoque más adecuado según la naturaleza del proceso y la aplicación de control. En [Siemens, 2025] se describen los siguientes lenguajes:

### 2.4.1. *Ladder* o lenguaje de escalera (LD)

*Ladder*, es un lenguaje de programación gráfico cuya representación se basa en diagramas eléctricos (ver Figura 2.6 ). Los elementos típicos de un circuito son los contactos normalmente abiertos, normalmente cerrados y las salidas. Además, incluye instrucciones en forma de bloques que abarcan una amplia variedad de funciones, como operaciones matemáticas, temporizadores, contadores y transferencias de datos (*move*). Su lectura se realiza de izquierda a derecha y de arriba hacia abajo.

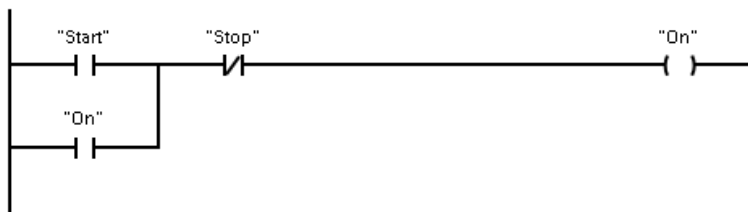


Figura 2.6 Ejemplo del lenguaje de programación Ladder.<sup>2.6</sup>

### 2.4.2. Diagrama de bloque de funciones (FBD)

**FBD**, también es un lenguaje de programación gráfico basado en la interconexión de bloques de función que representan operaciones lógicas, aritméticas y de control (ver Figura 2.7). Su representación se fundamenta en los símbolos lógicos empleados en el álgebra booleana. Sus elementos típicos son las entradas, salidas y bloques funcionales, los cuales se enlazan mediante líneas que definen el flujo de señales dentro del programa.

<sup>2.6</sup>Tomada de [Siemens, 2025]

<sup>2.7</sup>Tomada de [Siemens, 2025]

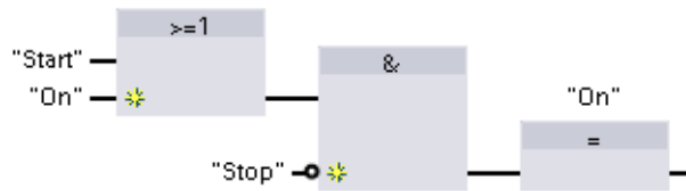


Figura 2.7 Ejemplo del lenguaje de programación FBD.<sup>2.7</sup>

### 2.4.3. Lenguaje de control estructurado (SCL)

**SCL**, es un lenguaje de programación de alto nivel basado en *Pascal* (ver Figura 2.8). Utiliza operadores de programación estándar, como los de asignación ( $:=$ ) y funciones matemáticas (+ para suma, - para resta, \* para multiplicación y / para división). Asimismo, incorpora estructuras de control típicas de *Pascal*, como IF-THEN-ELSE, CASE, REPEAT-UNTIL, GOTO y RETURN, lo que permite implementar algoritmos de control de manera estructurada, legible y eficiente.

```

REGION Scenario1
  #Structure.Control := 7;
  #Structure.Select := 8;
  IF FALSE THEN
    IF FALSE THEN
      "Block_2"();
    END_IF;
  ELSE
    #Static_1 := #Structure.Control;
    #Static_2 := #Structure.Select;
    /***Static_2 should get 8(value of Structure.Select)
    //but it takes 7(Value of Structure.Control).***
  END_IF;
END_REGION

```

Figura 2.8 Ejemplo del lenguaje de programación SCL.<sup>2.8</sup>

### 2.4.4. Lenguaje de lista de instrucciones (STL)

**STL**, es un lenguaje de programación textual de bajo nivel utilizado para crear bloques lógicos mediante instrucciones secuenciales (ver Figura 2.9). Su estructura, similar al lenguaje ensamblador, organiza el programa en filas numeradas, donde cada línea contiene una sola instrucción que la CPU ejecuta de arriba hacia abajo. Este lenguaje permite describir operaciones lógicas de manera compacta y secuencial, por lo que resulta útil en tareas de diagnóstico, optimización y programación de rutinas específicas.

<sup>2.8</sup>Tomada de [Siemens, 2025]

<sup>2.9</sup>Tomada de [Siemens, 2025]

Network 1: .....			
Comment			
1	O	"Start"	§IO.0
2	S	"Auto_mode"	§MO.0
Network 2: .....			
Comment			
1	O	"Stop"	§IO.1
2	R	"Auto_mode"	§MO.0
Network 3: .....			
Comment			
1	O	"FE_Low"	§IO.2
2	A	"FE_High"	§IO.3
3	=	#Large_box	
Network 4: .....			
Comment			
1	O	"FE_Low"	§IO.2
2	AN	"FE_High"	§IO.3
3	=	#Small_box	

Figura 2.9 Ejemplo del lenguaje de programación STL.<sup>2.9</sup>

### 2.4.5. Lenguaje para control secuencial (GRAPH)

**GRAPH**, es un lenguaje de programación gráfico diseñado para describir procesos secuenciales mediante pasos (*steps*) y transiciones (*transitions*) (ver Figura 2.10). Cada paso representa un estado o acción del proceso, mientras que las transiciones definen las condiciones lógicas que determinan el avance entre ellos. Este lenguaje facilita la programación, supervisión y diagnóstico de sistemas secuenciales, mejorando la claridad del programa y reduciendo errores en la lógica de control.

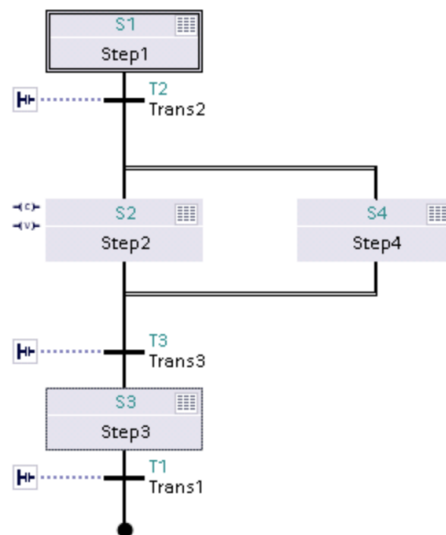


Figura 2.10 Ejemplo del lenguaje de programación GRAPH.<sup>2.10</sup>

<sup>2.10</sup>Tomada de [Siemens, 2025]

## Capítulo 3

# Desarrollo

En este capítulo se aplica la base teórica presentada en el Capítulo 2 para el desarrollo e implementación de la arquitectura de control SIMATIC. Se describen los criterios de diseño considerados, la selección y configuración del controlador, la organización de los módulos de entrada y salida, así como la integración de los procesos continuo y secuencial dentro de una misma estructura de control.

Asimismo, se presenta la forma en que se estandarizaron las señales, variables y rutinas de programación en el entorno TIA Portal, con el propósito de establecer una arquitectura modular, ordenada y funcional. De esta manera, se define la base técnica necesaria para la posterior evaluación de las estrategias de control implementadas.

### 3.1. Diseño e implementación de la arquitectura de control

La arquitectura de control fue diseñada considerando la integración de dos tipos de procesos representativos en el ámbito industrial: un sistema térmico (Figura 3.2), utilizado para el análisis de control continuo, y una mesa electroneumática (Figura 3.3), empleada para la validación de control secuencial. El sistema térmico simula condiciones de calentamiento presentes en múltiples aplicaciones industriales, como hornos, intercambiadores de calor y procesos de regulación de temperatura, lo que lo convierte en un caso de estudio adecuado para la evaluación de estrategias de control continuo. Por su parte, la mesa electroneumática reproduce secuencias de operación típicas en sistemas automatizados, permitiendo analizar la lógica de control discreto y la interacción entre actuadores. Para ello, se incorporó un sistema de conectividad remota que permite acceder al controlador desde cualquier punto dentro de la red local. De esta forma, se adoptó una topología en árbol, la cual ofrece una organización jerárquica que facilita la administración del sistema y permite su expansión de manera estructurada, escalable y eficiente.

El diseño se desarrolló a partir de la infraestructura de red ya existente en el laboratorio, la cual incluye un punto de acceso inalámbrico (*Wireless Access Point*) y un *switch Ethernet*. A partir de esta base, se integró la arquitectura propuesta, compuesta por un *switch* industrial SIMATIC SCALANCE X308-2LD, al cual se conectó el controlador SIMATIC S7-1500. Finalmente, tanto el sistema térmico como la mesa electroneumática se conectan directamente al PLC. El resultado de esta arquitectura se muestra en la Figura 3.1.

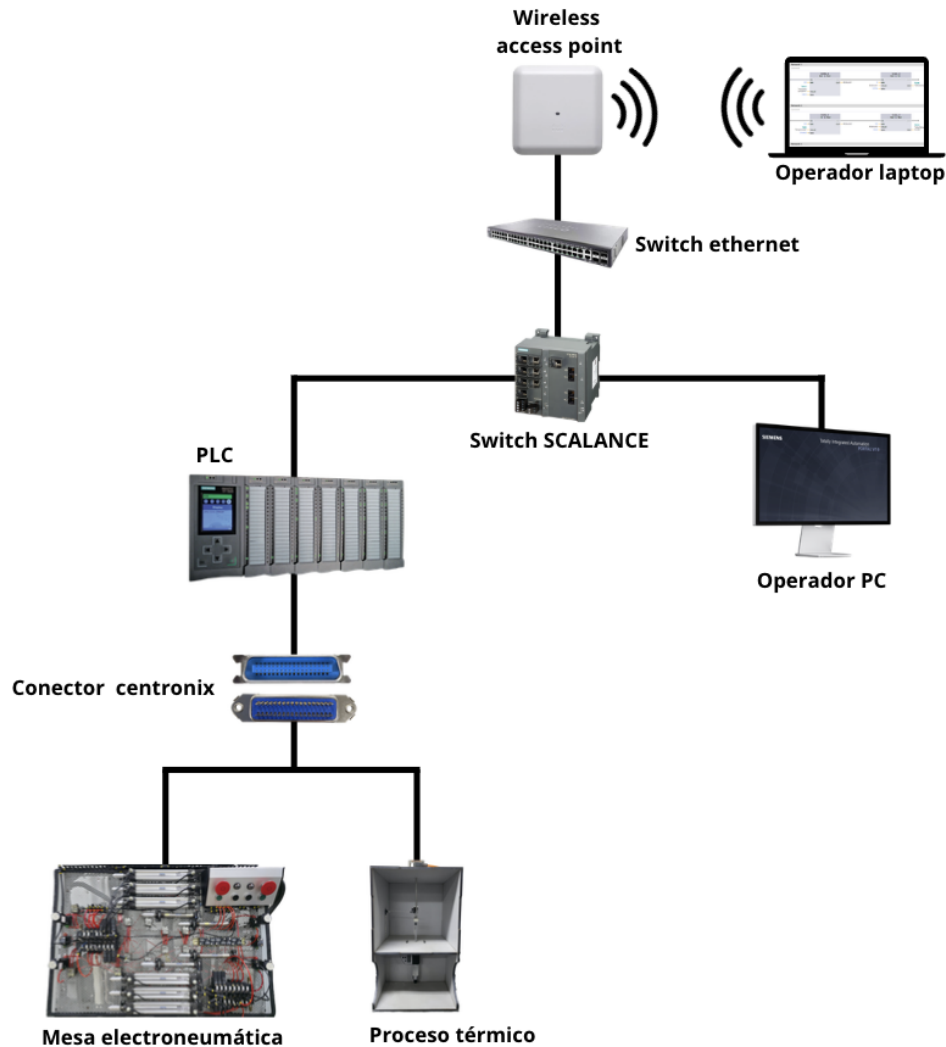


Figura 3.1 Arquitectura de control.<sup>3.1</sup>

## 3.2. Especificación funcional

Se realizó una especificación funcional tanto del sistema térmico (proceso continuo) como de la mesa electropneumática (proceso secuencial), con el objetivo de comprender el funcionamiento y las principales características de cada uno. Este análisis permitió identificar los sensores y actuadores involucrados en ambos procesos, así como clasificar las señales de entrada y salida que intervienen en cada proceso.

### 3.2.1. Especificación funcional del sistema térmico

El sistema térmico que observamos en la Figura 3.2 está diseñado como una plataforma experimental para simular un proceso de calentamiento (proceso continuo) e implementar rutinas de control. Está compuesto por una lámpara de halógeno como fuente de calor, un sensor TD2231 con *display*

<sup>3.1</sup>Elaboración propia.

digital, un termopar tipo K y un ventilador de golpe radial. Todos estos elementos se encuentran contenidos en una estructura construida con fibra de madera de densidad media (MDF, por sus siglas en inglés), lo que garantiza un adecuado aislamiento térmico del sistema.

**El sistema térmico presenta las siguientes características:**

- El sensor TD2231 permite visualizar en tiempo real la temperatura medida, facilitando el monitoreo sin necesidad de sistemas externos de adquisición de datos.
- El ventilador, aunque no forma parte activa del modelo de control, influye en la dinámica térmica al acelerar el enfriamiento del sistema, lo cual permite realizar pruebas repetidas en menor tiempo.
- El volumen interno se encuentra completamente aislado del ambiente exterior, lo que impide la mezcla del aire y contribuye a mantener condiciones térmicas estables y reproducibles.

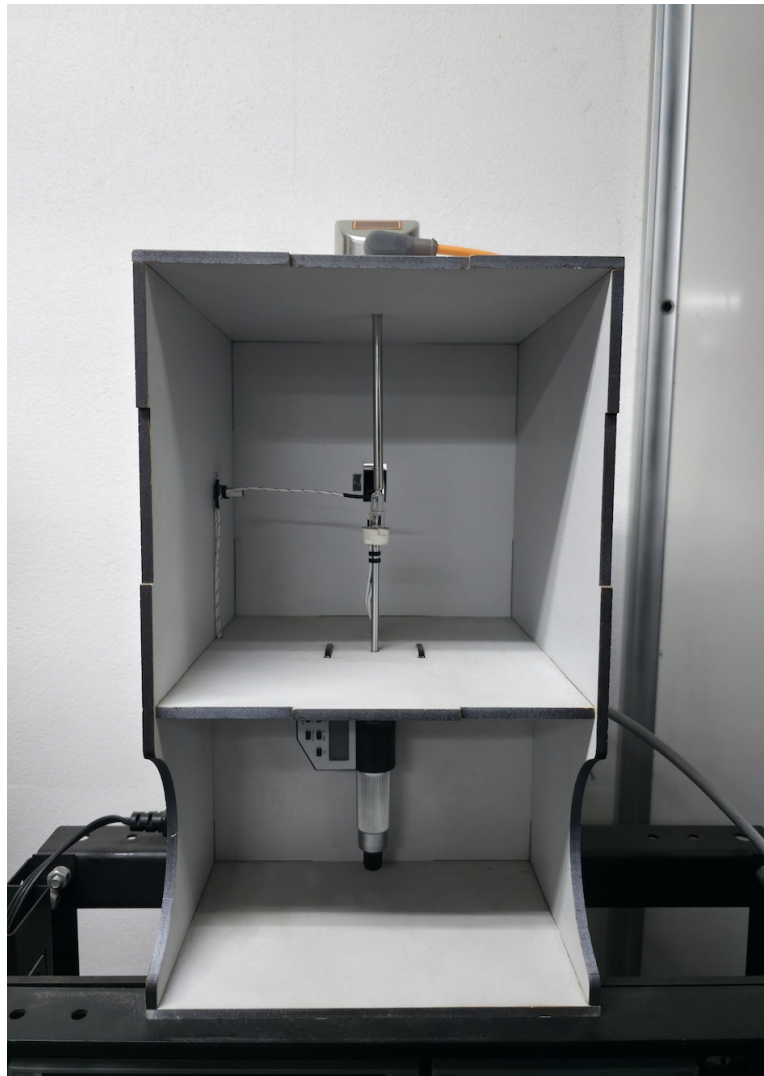


Figura 3.2 Sistema térmico.<sup>3.2</sup>

### 3.2.2. Especificación funcional de la mesa electroneumática

La mesa electroneumática mostrada en la Figura 3.3 integra un conjunto de dispositivos industriales que permiten la implementación y evaluación de rutinas de control secuencial. Está compuesta por seis pistones neumáticos de doble efecto y cuatro pistones de retorno por muelle, organizados en dos grupos simétricos, cada uno con tres pistones de doble efecto y dos de retorno por muelle. Los pistones de doble efecto están acoplados con sensores de posición tipo detector de límite, los cuales permiten identificar si el actuador se encuentra en posición contraída o extendida.

#### El sistema también incluye:

- Compuertas lógicas neumáticas, útiles para implementar funciones lógicas directamente en el circuito neumático sin requerir control electrónico.
- Temporizadores neumáticos, que permiten generar retardos en las señales neumáticas para sincronizar eventos dentro de la secuencia.
- Una botonera que incorpora dos indicadores neumáticos, dos interruptores de enclavamiento con contacto NC (normalmente cerrado) y dos pulsadores con contacto NA (normalmente abierto), diseñada para iniciar, detener o reiniciar las secuencias de manera segura.

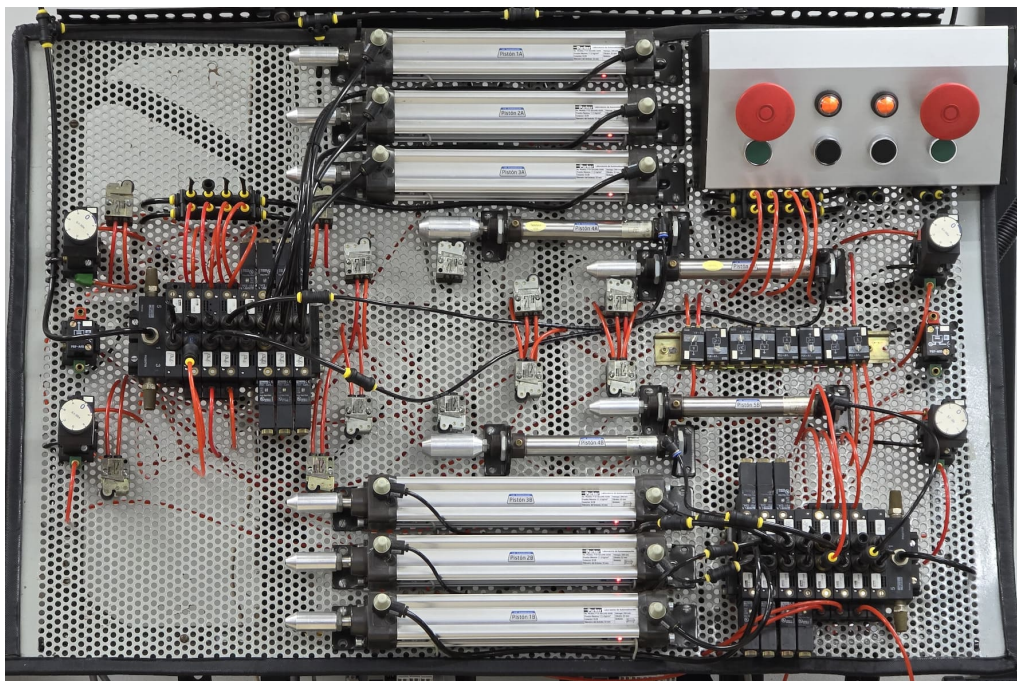


Figura 3.3 Mesa Electroneumática.<sup>3.3</sup>

<sup>3.2</sup>Tomada del archivo fotográfico personal.

<sup>3.3</sup>Tomada del archivo fotográfico personal.

### 3.3. Tabla de ordenamiento de variables

Una vez identificados los dispositivos que componen cada uno de los procesos, fue necesario clasificar su conexión a los módulos de I/O del controlador. Para ello, se elaboraron tablas de ordenamiento de variables que contienen la dirección, símbolo, una breve descripción y comentario; esto permitió organizar de forma clara y estructurada las señales, tanto digitales como analógicas. Como resultado, se obtuvieron las Tablas 3.1, 3.2, 3.3 y 3.4.

En la Tabla 3.1, la dirección inicial de las entradas digitales es I0.0, lo que corresponde al bit 0 del byte de entradas 0. En la nomenclatura de direccionamiento del PLC, el byte IB0 está compuesto por ocho bits, comprendidos entre I0.0 e I0.7. Una vez utilizadas estas posiciones, el direccionamiento continúa en el siguiente byte disponible, IB1, iniciando en I1.0.

Además, en todas las tablas se presenta el nombre asignado a cada dirección, junto con una descripción del equipo conectado a la misma. Por último, se incluye un comentario que indica brevemente la función que realiza cada una.

En la Tabla 3.2, se sigue el mismo criterio de asignación utilizado en las entradas digitales. La dirección Q0.0 corresponde al bit 0 del byte de salidas 0, mientras que Q1.0 corresponde al bit 0 del byte de salidas 1. De esta manera, el byte QB0 comprende las direcciones de Q0.0 a Q0.7, y el byte QB1 comprende las direcciones de Q1.0 a Q1.7.

#### Entradas digitales

DIRECCIÓN	NOMBRE	DESCRIPCIÓN	COMENTARIO
I0.0	Stop	Botón pulsador normalmente cerrado	Desenergiza al sistema
I0.1	Start	Botón pulsador normalmente abierto	Energiza al sistema
I0.2	SC1	Sensor de proximidad en la culata del pistón	Sensor de contracción 1
I0.3	SE1	Sensor de proximidad en la tapa del pistón	Sensor de expulsión 1
I0.4	SC2	Sensor de proximidad en la culata del pistón	Sensor de contracción 2
I0.5	SE2	Sensor de proximidad en la tapa del pistón	Sensor de expulsión 2
I0.6	SC3	Sensor de proximidad en la culata del pistón	Sensor de contracción 3
I0.7	SE3	Sensor de proximidad en la tapa del pistón	Sensor de expulsión 3
I1.0	Stop	Botón pulsador normalmente cerrado	Desenergiza el sistema
I1.1	Start	Botón pulsador normalmente abierto	Energiza el sistema
I1.2	SC4	Sensor de proximidad en la culata del pistón	Sensor de contracción 4
I1.3	SE4	Sensor de proximidad en la tapa del pistón	Sensor de expulsión 4
I1.4	SC5	Sensor de proximidad en la culata del pistón	Sensor de contracción 5
I1.5	SE5	Sensor de proximidad en la tapa del pistón	Sensor de expulsión 5
I1.6	SC6	Sensor de proximidad en la culata del pistón	Sensor de contracción 6
I1.7	SE6	Sensor de proximidad en la tapa del pistón	Sensor de expulsión 6

Tabla 3.1 Tabla de asignación de entradas digitales.

De manera similar a la tabla anterior, se presenta el nombre asignado, la descripción del equipo conectado y su respectivo comentario, lo que permite identificar de manera clara la función de cada salida dentro del sistema.

### Salidas digitales

DIRECCIÓN	NOMBRE	DESCRIPCIÓN	COMENTARIO
Q0.0	EV_E1	Electroválvula de expulsión 1	Expulsa el vástago del pistón 1
Q0.1	EV_C1	Electroválvula de contracción 1	Contrae el vástago del pistón 1
Q0.2	EV_E2	Electroválvula de expulsión 2	Expulsa el vástago del pistón 2
Q0.3	EV_C2	Electroválvula de contracción 2	Contrae el vástago del pistón 2
Q0.4	EV_E3	Electroválvula de expulsión 3	Expulsa el vástago del pistón 3
Q0.5	EV_C3	Electroválvula de contracción 3	Contrae el vástago del pistón 3
Q1.0	EV_E4	Electroválvula de expulsión 4	Expulsa el vástago del pistón 4
Q1.1	EV_C4	Electroválvula de contracción 4	Contrae el vástago del pistón 4
Q1.2	EV_E5	Electroválvula de expulsión 5	Expulsa el vástago del pistón 5
Q1.3	EV_C5	Electroválvula de contracción 5	Contrae el vástago del pistón 5
Q1.4	EV_E6	Electroválvula de expulsión 6	Expulsa el vástago del pistón 6
Q1.5	EV_C6	Electroválvula de contracción 6	Contrae el vástago del pistón 6

Tabla 3.2 Tabla de asignación de salidas digitales.

En la Tablas 3.3 y 3.4, se presentan las direcciones correspondientes a las entradas y salidas analógicas, en este caso IW4 para la entrada y QW4 para la salida.

En ambas tablas se mantienen los mismos títulos, donde se presenta el nombre, junto con la descripción del equipo conectado y su respectivo comentario, lo que permite identificar de manera clara la función de cada salida dentro del sistema.

### Entradas analógicas

DIRECCIÓN	NOMBRE	DESCRIPCIÓN	COMENTARIO
IW4	TE-001	Termopar tipo K	Sensor de temperatura

Tabla 3.3 Tabla de asignación de entradas analógicas.

### Salidas analógicas

DIRECCIÓN	NOMBRE	DESCRIPCIÓN	COMENTARIO
QW4	FC-001	Actuador lámpara	Lámpara de halógeno

Tabla 3.4 Tabla de asignación de salidas analógicas.

### 3.4. Desarrollo de diagramas eléctricos mediante CAD

Una vez identificadas las señales que intervienen en los procesos y definido su direccionamiento, se procedió a la elaboración de los diagramas eléctricos para cada uno de los módulos de I/O del sistema, utilizando herramientas de diseño CAD.

En la Figura 3.4 se presenta el diagrama eléctrico del módulo de entradas digitales, en este se muestra la conexión de los sensores encargados de detectar la expulsión y contracción de los actuadores del sistema. Asimismo, se incluyen el pulsador *Start*, utilizado para la activación del sistema, y el interruptor de enclavamiento *Stop*, empleado para el paro de emergencia; ambos elementos se utilizan en el proceso secuencial. Todos estos dispositivos se encuentran cableados hacia las terminales de entrada del PLC.

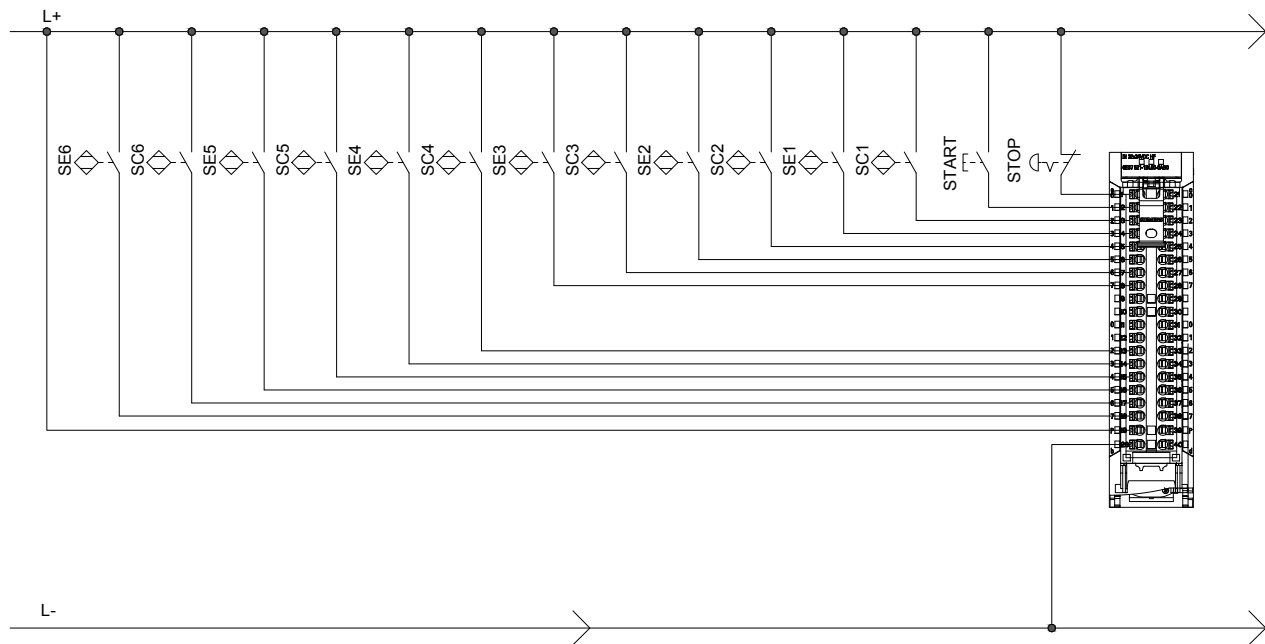


Figura 3.4 Diagrama de conexión del módulo de entradas digitales.<sup>3.4</sup>

En la Figura 3.5 se presenta el diagrama eléctrico del módulo de salidas digitales, en el cual se muestra la conexión de los actuadores del sistema, específicamente las electroválvulas encargadas de controlar la expulsión y contracción de los actuadores. Cada salida del PLC está asociada a una electroválvula (EV), permitiendo su activación de manera individual mediante señales digitales.

<sup>3.4</sup>Elaboración personal.

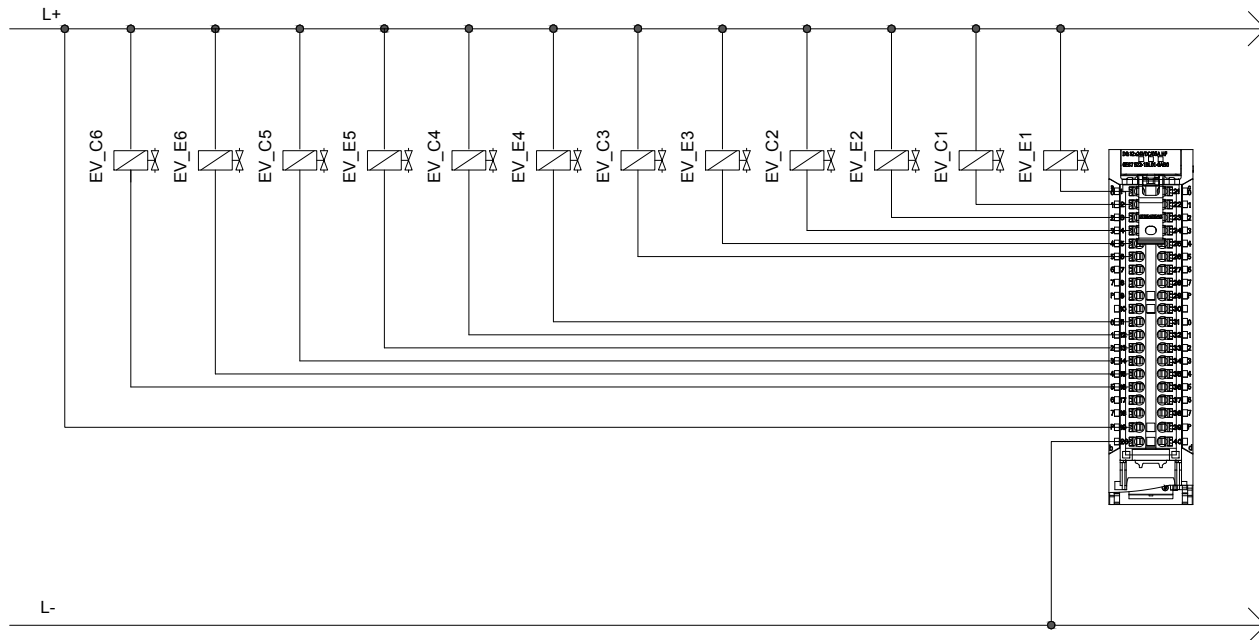


Figura 3.5 Diagrama de conexión del módulo de salidas digitales.<sup>3.5</sup>

En la Figura 3.6 se presenta el diagrama eléctrico del módulo de entradas analógicas, en este se muestra la conexión del sensor de temperatura TE-001 hacia el PLC. Dicho sensor proporciona una señal analógica proporcional a la variable física medida, la cual es adquirida por el módulo para su posterior procesamiento en el sistema de control.

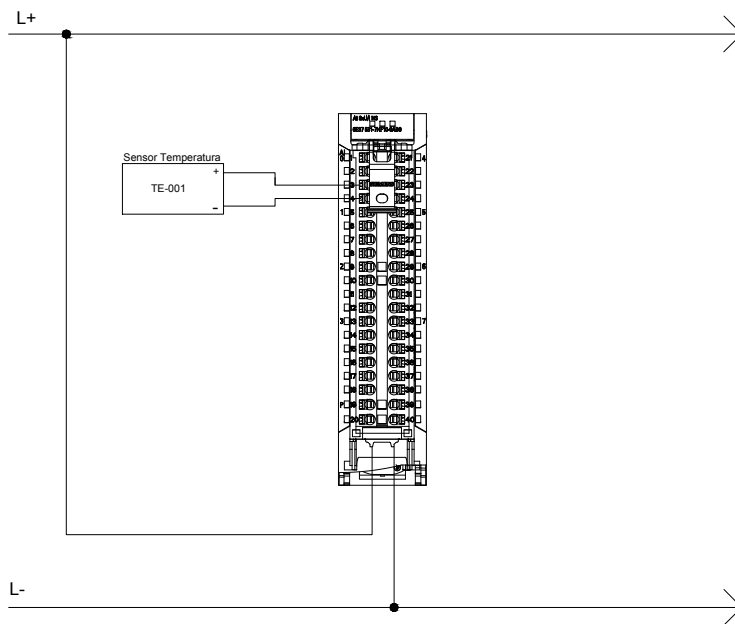


Figura 3.6 Diagrama de conexión del módulo de entradas analógicas.<sup>3.6</sup>

<sup>3.5</sup>Elaboración personal.

<sup>3.6</sup>Elaboración personal.

En la Figura 3.7 se presenta el diagrama eléctrico del módulo de salidas analógicas, en el cual se muestra la conexión del actuador del sistema, específicamente la lámpara FC-001, utilizada como elemento final de control en el proceso. Este actuador recibe una señal analógica proveniente del PLC, lo que permite regular su intensidad de manera continua en función de la señal de control generada.

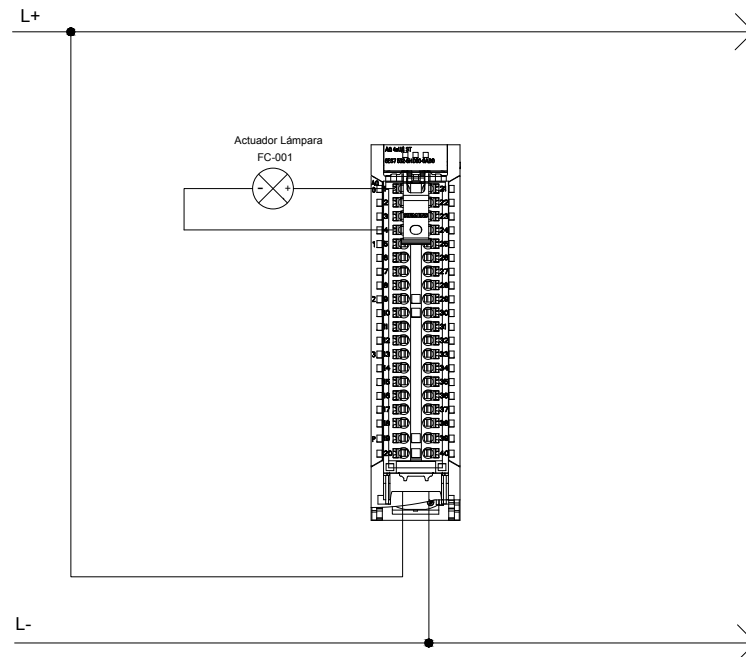


Figura 3.7 Diagrama de conexión del módulo de salidas analógicas.<sup>3.7</sup>

### 3.5. Conector Centronics

Un conector Centronics, como el que se muestra en la Figura 3.8, es un conector físico de tipo paralelo, comúnmente utilizado para transmitir señales digitales de forma simultánea a través de múltiples pines. En este caso, se utilizó un modelo de 36 pines, cuya distribución fue diseñada específicamente para estandarizar la conexión entre el PLC, los procesos implementados y aquellos que puedan integrarse en el futuro.

<sup>3.7</sup>Elaboración personal.

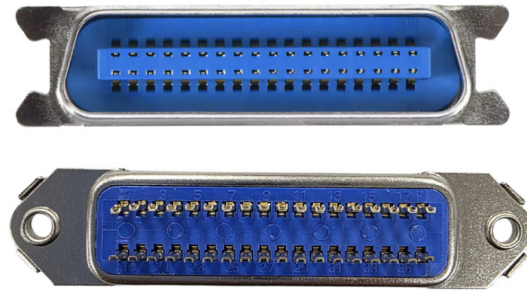


Figura 3.8 Conector Centronics de 36 pines utilizado para la conexión estandarizada de procesos.<sup>3.8</sup>

Para lograr esta estandarización, se definió una distribución de pines que contempla tanto señales digitales como analógicas, además de líneas de alimentación y tierra. Esta organización permite una conexión ordenada, reutilizable y eficiente, lo que evita la saturación de los módulos del PLC y reduce significativamente el tiempo requerido para el cableado y configuración de nuevos procesos. La distribución de pines la podemos observar en la Tabla 3.5.

A manera de resumen, los pines del conector Centronics se organizaron de la siguiente forma: del pin 1 al 8 se asignaron a entradas digitales; del 9 al 16 a salidas digitales; del 17 al 20 se destinaron a líneas de alimentación (V+); y del 21 al 24 a líneas de tierra (GND). Posteriormente, del pin 25 al 32 se distribuyeron las señales correspondientes a entradas analógicas (pares diferencial positivo/negativo), y finalmente, del 33 al 36 a salidas analógicas. Esta distribución estructurada permite una conexión ordenada y facilita el cableado de los módulos del controlador.

---

<sup>3.8</sup>Elaboración personal.

Tabla 3.5 Distribución de pines del conector Centronics

Pin Centronics	Dirección	Comentarios
1	I0.0	Entrada Digital
2	I0.1	Entrada Digital
3	I0.2	Entrada Digital
4	I0.3	Entrada Digital
5	I0.4	Entrada Digital
6	I0.5	Entrada Digital
7	I0.6	Entrada Digital
8	I0.7	Entrada Digital
9	Q0.0	Salida Digital
10	Q0.1	Salida Digital
11	Q0.2	Salida Digital
12	Q0.3	Salida Digital
13	Q0.4	Salida Digital
14	Q0.5	Salida Digital
15	Q0.6	Salida Digital
16	Q0.7	Salida Digital
17	V+	Alimentación
18	V+	Alimentación
19	V+	Alimentación
20	V+	Alimentación
21	GND	Tierra
22	GND	Tierra
23	GND	Tierra
24	GND	Tierra
25	IW4+	Entrada Analógica
26	IW4-	Entrada Analógica
27	IW6+	Entrada Analógica
28	IW6-	Entrada Analógica
29	IW8+	Entrada Analógica
30	IW8-	Entrada Analógica
31	IW10+	Entrada Analógica
32	IW10-	Entrada Analógica
33	QW4+	Salida Analógica
34	QW4-	Salida Analógica
35	QW6+	Salida Analógica
36	QW6-	Salida Analógica

### 3.6. Diseño e implementación del sistema de control continuo

Las rutinas de control desarrolladas para ambos sistemas se implementaron conforme a la norma internacional IEC 61131-3, la cual define los lenguajes de programación estándar para los PLC. Se diseñaron e implementaron dos rutinas específicas para los diferentes procesos: una utilizando el lenguaje LD, orientada al control del proceso continuo, y otra mediante el lenguaje GRAPH, empleada para la gestión del proceso secuencial. También, para el desarrollo de esta última, se adoptó la filosofía GRAFCET, la cual permite representar de forma estructurada y jerárquica la secuencia de pasos, transiciones y acciones del sistema.

#### 3.6.1. Rutina de control proceso continuo

#### 3.6.2. Escalamientos

El escalamiento de las señales analógicas en TIA Portal es un paso esencial dentro de la rutina de control para el proceso continuo, ya que permite convertir las lecturas provenientes de los instrumentos de campo en unidades normalizadas. Este proceso facilita la interpretación de las variables y contribuye al funcionamiento del sistema de control.

En la Figura 3.9 se muestra el bloque principal que invoca la subrutina **Escalamientos**, cuyo propósito es realizar la conversión de las señales analógicas a valores normalizados y porcentuales.

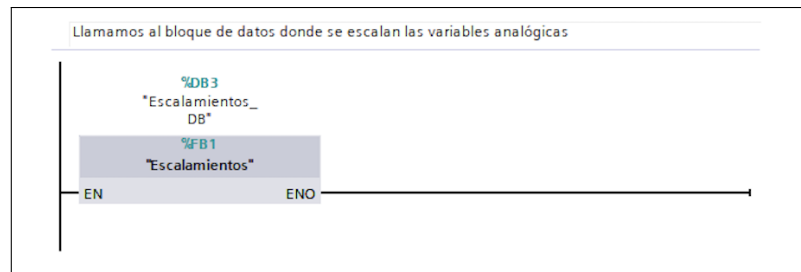


Figura 3.9 Bloque que invoca la subrutina Escalamiento.<sup>3.9</sup>

A continuación, en la Figura 3.10, se observa el primer peldaño de la subrutina **Escalamientos**, donde se emplea el bloque **NORM\_X** para normalizar la señal asociada a la lámpara a un valor real comprendido entre 0 y 1. Posteriormente, dicho valor es utilizado por el bloque **SCALE\_X** para escalarlo en un rango de 0 a 100 %, lo que permite expresar el nivel de intensidad de la lámpara en términos porcentuales y facilitar su control dentro del sistema.

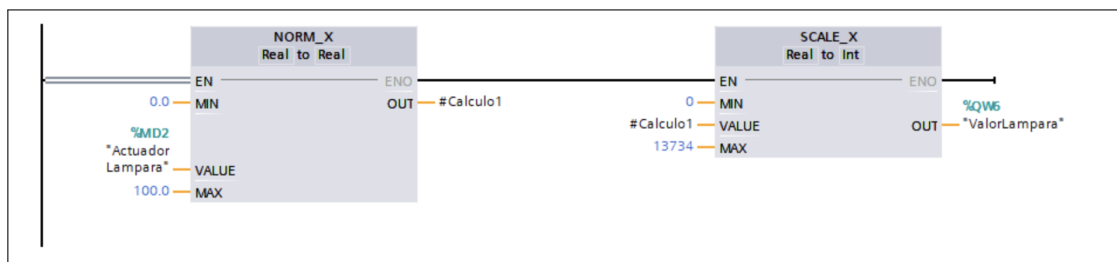


Figura 3.10 Peldaño 1 dentro de la subrutina Escalamientos.<sup>3.10</sup>

<sup>3.9</sup>Captura de pantalla tomada a partir de la programación desarrollada.

La Figura 3.11 muestra el segundo peldaño, el cual realiza un proceso similar para la señal proveniente del sensor de temperatura. La diferencia radica en el bloque `SCALE_X`, que convierte la señal normalizada a su rango físico, en este caso de 0 a 300 °C, obteniendo así la variable de temperatura.

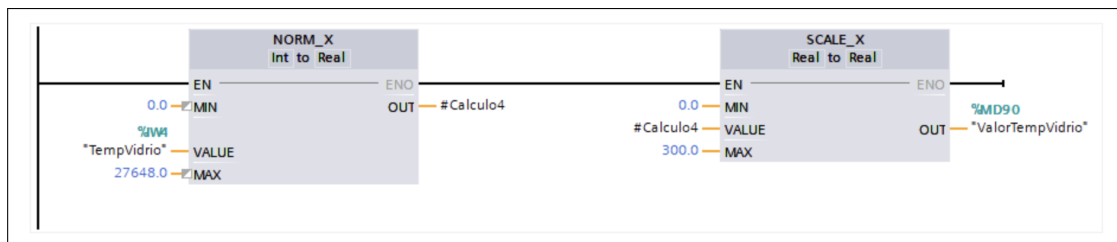


Figura 3.11 Peldaño 2 dentro de la subrutina Escalamientos.<sup>3.11</sup>

### 3.6.3. Rutina de control LD

Una vez escaladas las señales, se desarrolló una rutina de control destinada a realizar las pruebas del sistema térmico bajo la acción del controlador PID. Con este propósito, se propuso la implementación de una estructura lógica que permitiera evaluar el comportamiento dinámico del proceso ante diferentes consignas y condiciones de operación.

La Figura 3.12 muestra el peldaño 1, donde se observa que, para iniciar la ejecución del programa, es necesario presionar el botón `StartPID1`, el cual activa el bloque `PID_Compact`. Una vez activado, este bloque habilita la variable `#PID1Activo`, que permite la ejecución de la siguiente instrucción.

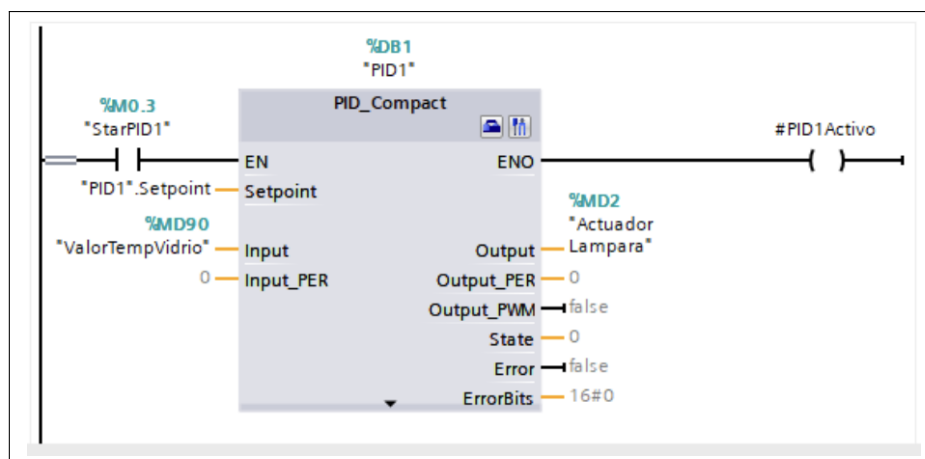


Figura 3.12 Peldaño 1 de la rutina LD.<sup>3.12</sup>

La Figura 3.13 muestra el peldaño 2, donde, mientras la variable `#PID1Activo` se encuentra habilitada, se ejecuta la instrucción `MOVE`, la cual asigna un valor constante de 30.0 °C al parámetro `Setpoint` del bloque `PID_Compact`. Al cumplirse esta condición, se activa la variable `#HabCont1`, que habilita la siguiente etapa de la rutina de control.

<sup>3.10</sup> Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.11</sup> Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.12</sup> Captura de pantalla tomada a partir de la programación desarrollada.

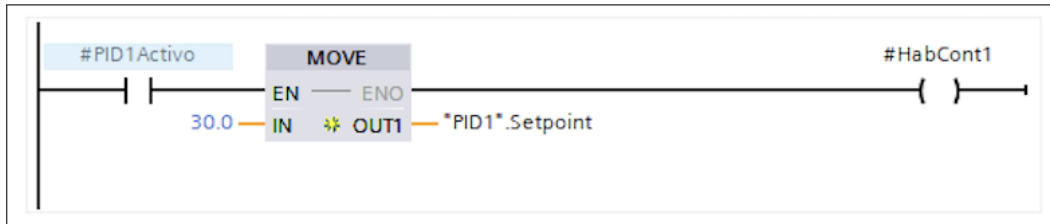


Figura 3.13 Peldaño 2 de la rutina LD.<sup>3.13</sup>

De la Figura 3.14 a la Figura 3.17 se muestran los peldaños 3 al 6, en los cuales se emplean bloques temporizadores del tipo TON, identificados como T1, T2, T3 y T4.

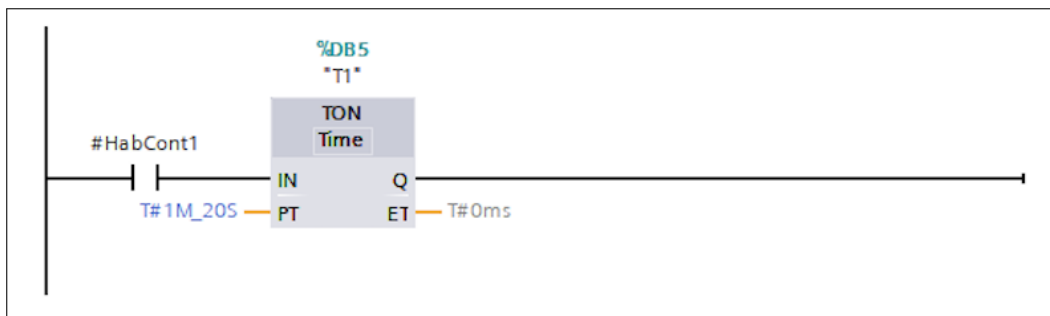


Figura 3.14 Peldaño 3 de la rutina LD.<sup>3.14</sup>

El primer temporizador (T1) se activa cuando la variable #HabCont1 se encuentra activa, iniciando un tiempo preestablecido de 1 minuto y 20 segundos (T#1M\_20S). Dicho intervalo fue definido para garantizar el establecimiento de las condiciones iniciales del sistema. Una vez transcurrido ese tiempo, la salida Q se activa, habilitando el siguiente peldaño de la rutina.

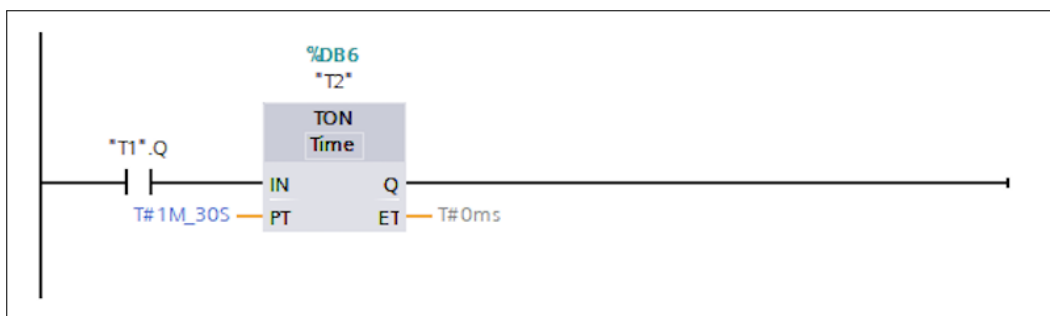


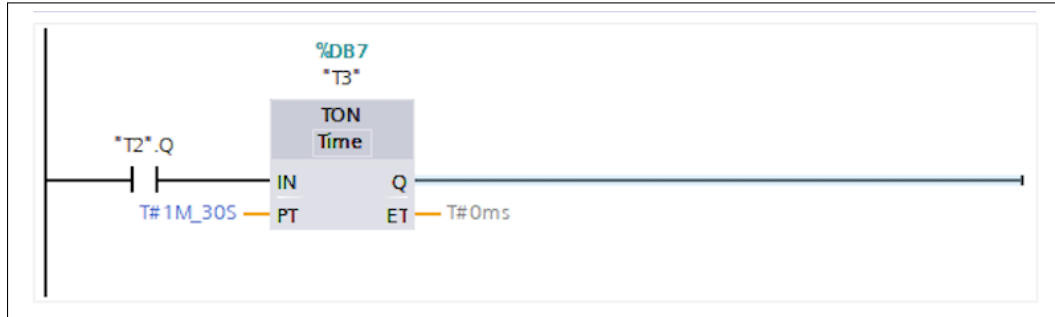
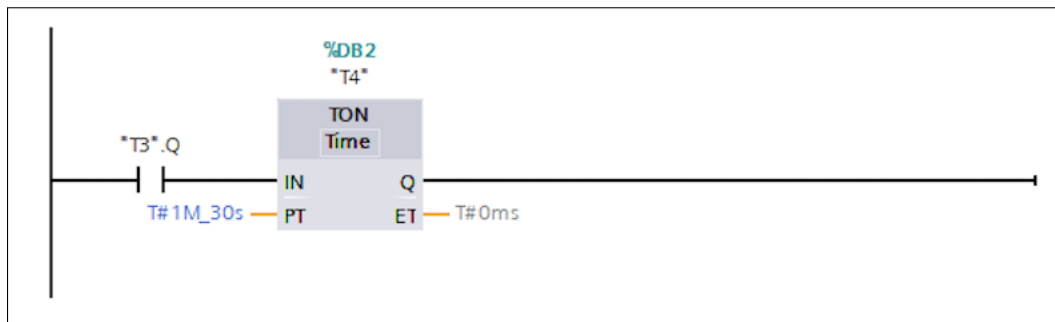
Figura 3.15 Peldaño 4 de la rutina LD.<sup>3.15</sup>

Este comportamiento se repite con los temporizadores restantes, los cuales presentan tiempos configurados de 1 minuto y 30 segundos cada uno. De esta forma, se establece una cadena de temporizadores que controla el cambio de los valores de Setpoint, como se mostrará más adelante.

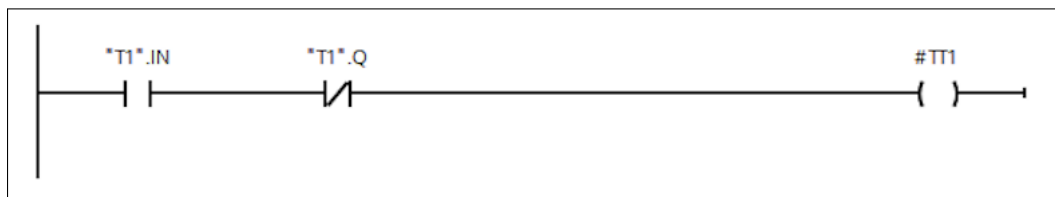
<sup>3.13</sup>Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.14</sup>Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.15</sup>Captura de pantalla tomada a partir de la programación desarrollada.

Figura 3.16 Peldaño 5 de la rutina LD.<sup>3.16</sup>Figura 3.17 Peldaño 6 de la rutina LD.<sup>3.17</sup>

De la Figura 3.18 a la Figura 3.21 se muestran los peldaños 7 al 10. En esta sección se implementa la lógica asociada a las variables #TT1, #TT2, #TT3 y #TT4, las cuales permanecen activas únicamente durante el tiempo de conteo de cada bloque temporizador del tipo TON.

Figura 3.18 Peldaño 7 de la rutina LD.<sup>3.18</sup>

La Figura 3.18 muestra la lógica correspondiente al temporizador T1. El peldaño se activa cuando la entrada del temporizador ("T1".IN) se encuentra activa. Durante este periodo, mientras el temporizador realiza la cuenta del tiempo configurado, la salida "T1".Q permanece desactivada. En esta condición, la variable #TT1 se mantiene activa, indicando que el temporizador T1 se encuentra en proceso de conteo. Una vez finalizado el tiempo preestablecido, la salida "T1".Q se habilita, desactivando la variable #TT1 y señalando que el temporizador ha completado su ciclo.

De manera análoga, las Figuras 3.19 a 3.21 presentan la misma estructura lógica aplicada a los temporizadores T2, T3 y T4, vinculadas con las variables #TT2, #TT3 y #TT4, respectivamente.

<sup>3.16</sup> Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.17</sup> Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.18</sup> Captura de pantalla tomada a partir de la programación desarrollada.

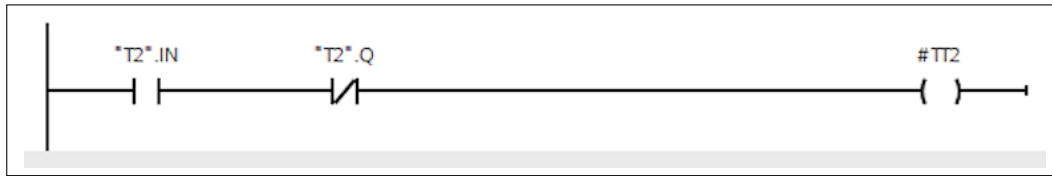


Figura 3.19 Peldaño 8 de la rutina LD.<sup>3.19</sup>

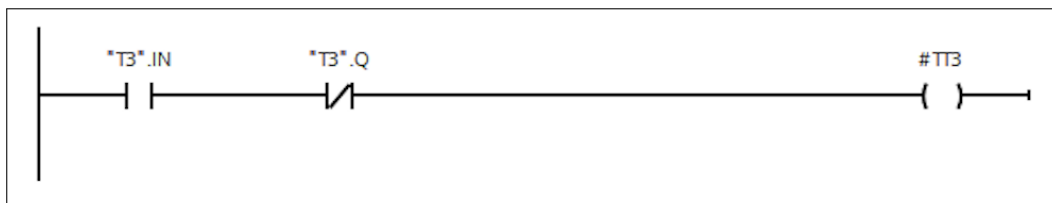


Figura 3.20 Peldaño 9 de la rutina LD.<sup>3.20</sup>



Figura 3.21 Peldaño 10 de la rutina LD.<sup>3.21</sup>

De la Figura 3.22 a la Figura 3.26 se muestran los peldaños 11 al 15, donde se implementa la lógica encargada de modificar el valor del SP del bloque `PID_Compact`. En esta sección, las variables `#TT2`, `#TT3` y `#TT4` se utilizan como condiciones de habilitación para ejecutar la instrucción `MOVE`.

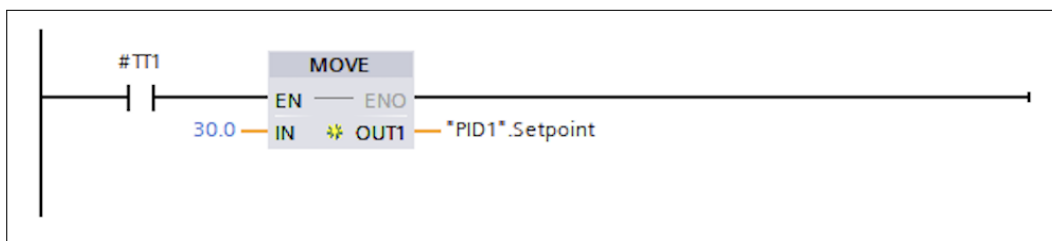


Figura 3.22 Peldaño 11 de la rutina LD.<sup>3.22</sup>

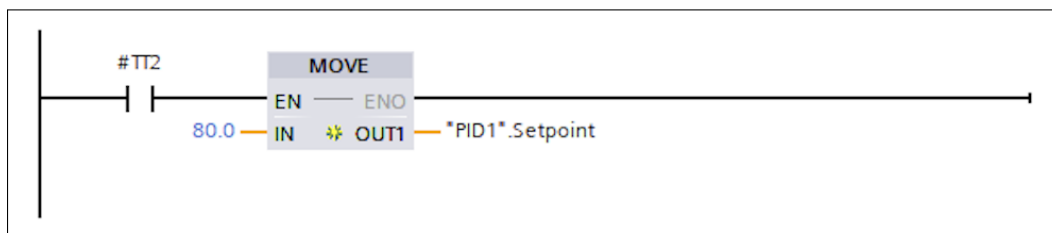
Cada vez que una de estas variables se encuentra activa, el bloque `MOVE` asigna un nuevo valor al parámetro `"PID1".Setpoint`. De esta manera, se generan incrementos escalonados en la variable de referencia con el propósito de analizar la respuesta dinámica del sistema térmico ante variaciones en el punto de ajuste.

<sup>3.19</sup>Captura de pantalla tomada a partir de la programación desarrollada.

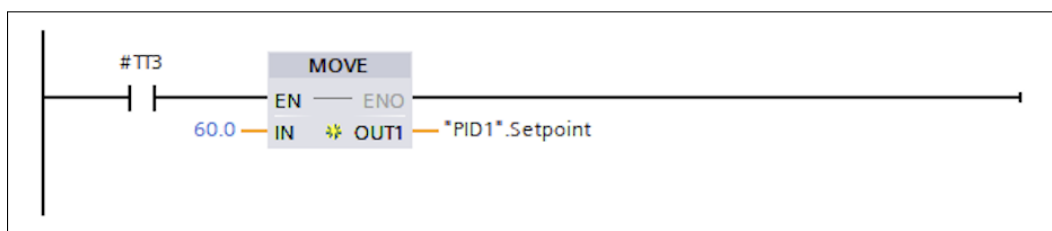
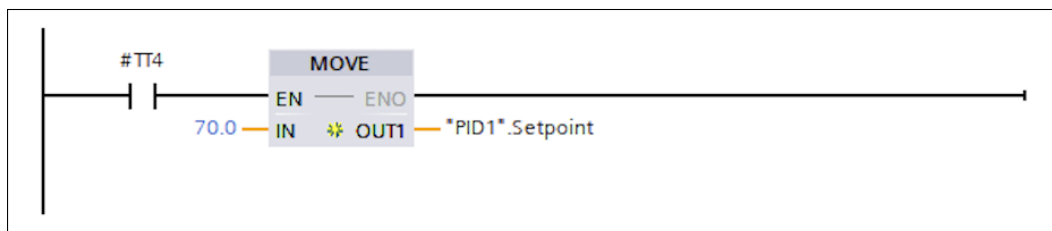
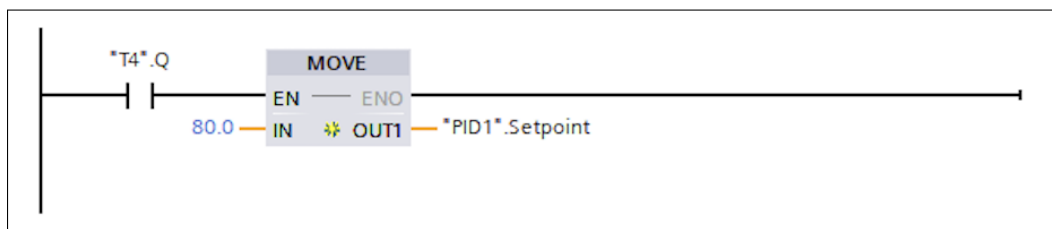
<sup>3.20</sup>Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.21</sup>Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.22</sup>Captura de pantalla tomada a partir de la programación desarrollada.

Figura 3.23 Peldaño 12 de la rutina LD.<sup>3.23</sup>

En las Figuras 3.23 a 3.26 se observa que los valores de SP aplicados corresponden a 80.0 °C, 60.0 °C, 70.0 °C y nuevamente 80.0 °C, respectivamente. Esta secuencia de variaciones permite evaluar la estabilidad, el tiempo de respuesta y el desempeño del controlador PID bajo diferentes condiciones de operación.

Figura 3.24 Peldaño 13 de la rutina LD.<sup>3.24</sup>Figura 3.25 Peldaño 14 de la rutina LD.<sup>3.25</sup>Figura 3.26 Peldaño 15 de la rutina LD.<sup>3.26</sup>

<sup>3.23</sup> Captura de pantalla tomada a partir de la programación desarrollada.

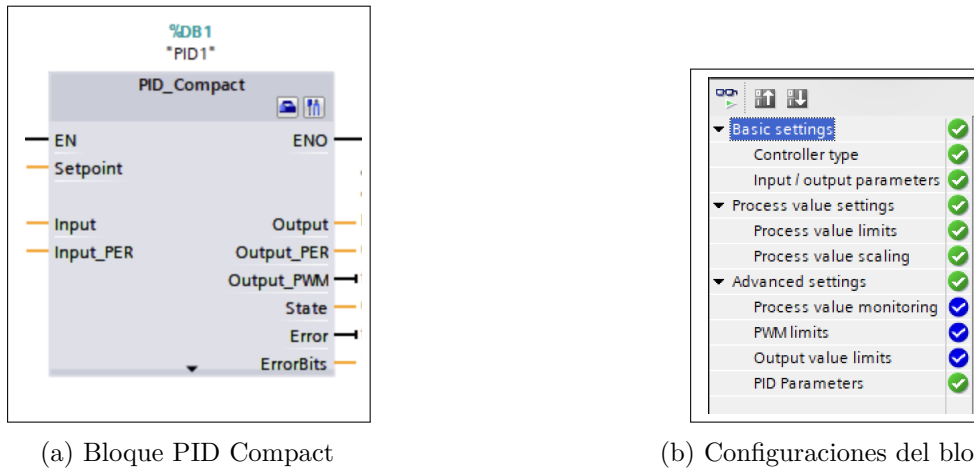
<sup>3.24</sup> Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.25</sup> Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.26</sup> Captura de pantalla tomada a partir de la programación desarrollada.

### 3.6.4. Implementación del control PID Compact

TIA Portal cuenta con un bloque llamado PID\_Compact para la implementación un controlador PID, el cual se muestra en la Figura 3.27a. Las diferentes configuraciones que tiene el bloque se muestran en la Figura 3.27b. Estas configuraciones nos permiten definir parámetros como los límites del proceso, escalado de señales, adaptar el comportamiento del controlador a características específicas, entre otros.



(a) Bloque PID Compact

(b) Configuraciones del bloque

Figura 3.27 Bloque PID Compact y sus configuraciones en TIA Portal.<sup>3.27</sup>

En la configuración básica del bloque se define el tipo de control a utilizar. En este caso, tal como se observa en la Figura 3.28, se seleccionó el modo de control de temperatura, estableciendo como unidad de medida grados Celsius (°C).

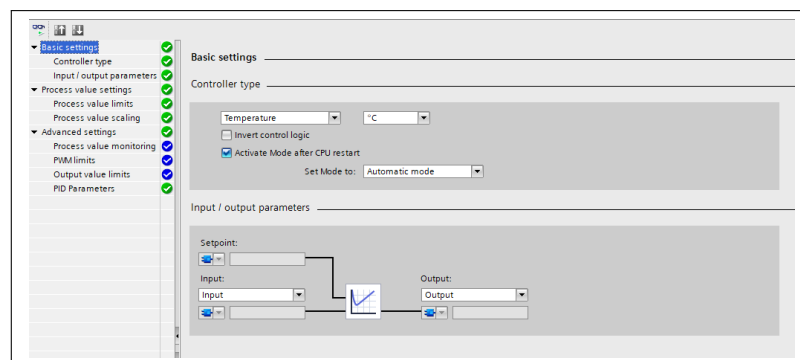


Figura 3.28 Configuraciones básicas del PID Compact.<sup>3.28</sup>

En el apartado de configuración de las variables de proceso, es posible definir los límites máximos y mínimos de la variable controlada, tal como se muestra en la Figura 3.29. Estos parámetros permiten restringir el rango operativo de la PV, evitando que el controlador genere acciones de control fuera del rango permitido. En nuestro caso, los límites que se establecieron fueron 300 °C como máximo y 0 °C como valor mínimo.

<sup>3.27</sup>Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.28</sup>Captura de pantalla tomada a partir de la programación desarrollada.

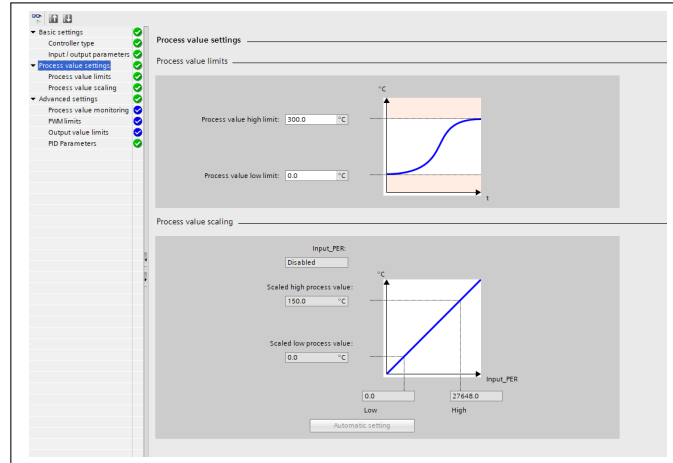


Figura 3.29 Configuraciones de las variables de proceso del PID Compact.<sup>3.29</sup>

En el apartado de configuración avanzada es posible establecer los límites de la señal de salida (*Output value limits*), tal como se observa en la Figura 3.30. Estos parámetros permiten restringir el rango de operación de la salida del controlador, evitando que se generen comandos de control fuera de las capacidades físicas del actuador; en nuestro caso, los límites se definieron entre 100 % como máximo y 0 % como mínimo. Asimismo, se configuró la reacción ante errores (*Reaction to error*) para que el controlador adopte un valor de salida sustituto de 0 % en caso de fallo, garantizando así la seguridad del proceso ante cualquier eventualidad técnica.

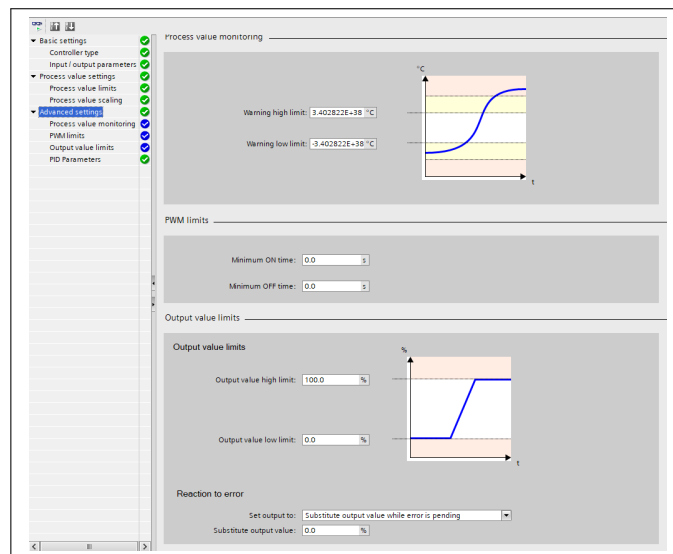


Figura 3.30 Configuraciones avanzadas del PID Compact.<sup>3.30</sup>

Por último, se presentan los parámetros del controlador PID, los cuales incluyen la ganancia proporcional (*Proportional gain*), el tiempo de acción integral (*Integral action time*) y el tiempo de acción derivativa (*Derivative action time*), tal como se ilustra en la Figura 3.31. A partir de lo especificado en el manual de referencia de [Siemens, 2023], se confirma que el bloque opera bajo una

<sup>3.29</sup> Captura de pantalla tomada a partir de la programación desarrollada.

<sup>3.30</sup> Captura de pantalla tomada a partir de la programación desarrollada.

estructura de ganancias dependientes. En este esquema, la ganancia proporcional actúa como un factor global que afecta directamente a las componentes integral y derivativa.

Para esta aplicación, los parámetros iniciales del controlador se obtuvieron mediante la función de *pretuning* del bloque PID\_Compact. Posteriormente, se empleó la función de *fine tuning* con el propósito de refinar la respuesta dinámica del sistema y ajustar el comportamiento del controlador de acuerdo con las características del proceso. Una vez finalizada la sintonización, se habilitó la entrada manual (*Enable manual entry*) para conservar los parámetros obtenidos, correspondientes a una ganancia proporcional de 2.0, un tiempo integral de 12.0 s y un tiempo derivativo de 0.005 s.

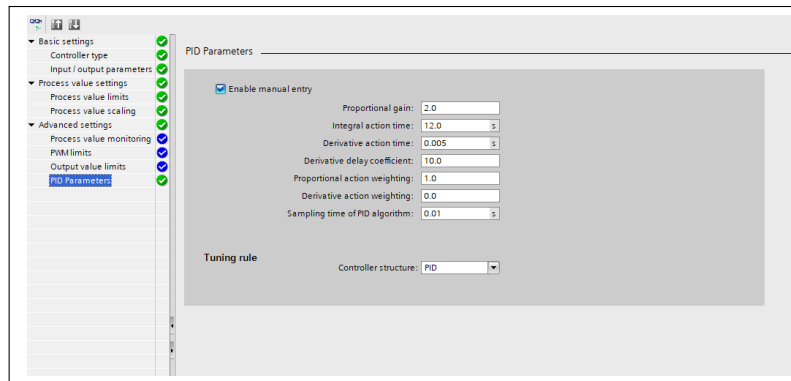


Figura 3.31 Parametros del PID Compact.<sup>3.31</sup>

### 3.6.5. Implementación del control PID nativo

El controlador PID nativo, basado en la Ecuación (2.8) con ganancias dependientes, descrita en el capítulo 2, se observa en el Código 3.1.

En primer lugar, se establecen los mismos parámetros del controlador PID\_Compact, tales como la ganancia proporcional  $K_c$ , el tiempo integral  $T_i$ , el tiempo derivativo  $T_d$  y el tiempo de muestreo  $\Delta t$ . Asimismo, se incorpora un filtro derivativo mediante el parámetro  $N$ , con el fin de reducir el efecto del ruido en la señal derivativa, así como la ganancia de *back-calculation*  $K_b$ , empleada en la implementación del esquema de *anti-windup*.

Posteriormente, se calcula el error del sistema como la diferencia entre el SP y la PV, en este caso la temperatura medida (**ValorTemVidrio**). A partir de este error, el controlador genera una señal de control que define la variable manipulada del sistema, representada por **ActuadorLampara**, la cual regula la potencia suministrada a la lámpara con el objetivo de llevar la temperatura al valor deseado. La acción derivativa es implementada mediante una aproximación discreta con filtrado, mientras que la acción integral se obtiene a través de la integración numérica del error en función del tiempo de muestreo.

La señal de control se calcula considerando la suma de las tres acciones del controlador, dando lugar a la variable **ActuadorLampara**, la cual representa la variable manipulada del sistema y actúa directamente sobre la dinámica térmica del proceso. Esta señal es posteriormente limitada mediante una saturación entre 0 y 100 %, con el objetivo de respetar las restricciones físicas del actuador.

<sup>3.31</sup>Captura de pantalla tomada a partir de la programación desarrollada.

Finalmente, se implementa un esquema de *anti-windup* mediante el método de *back-calculation*, con el propósito de mitigar el efecto de integración acumulativa asociado a condiciones de saturación en la salida del controlador, favoreciendo así un comportamiento más estable del sistema.

```

1 // Proceso Termico con PID de Ganancias Dependientes
2 "dt" := 0.01; // Tiempo de muestreo (segundos)
3 "Kc" := 2; // Ganancia del controlador
4 "Ti" := 12; // Tiempo integral (en segundos)
5 "Td" := 0.005; // Tiempo derivativo (en segundos)
6 "Kb" := 2.2065; // Ganancia de back-calculation
7 "N" := 10; // Filtro derivativo (velocidad de respuesta)
8
9 // Setpoint y error
10 "SP" := 60;
11 "error" := "SP" - "ValorTempVidrio";
12
13 // Derivada filtrada del error
14 "K1" := ("Kc" * "Td") / ("Td" + "N" * "dt");
15 "derror" := "K1" * ("error" - "error_ant");
16 "x1" := "x1" + "K1" * "dt";
17
18 // Parte integral
19 "K2" := "Kc" * ("dt" / "Ti");
20 "ierror" := "x2";
21 "x2" := "x2" + "K2" * "error" * "dt"; // Integracion del error
22
23 // PID con ganancias dependientes
24 "ActuadorLampara" := "Kc" * ("error" + (1 / "Ti") * "ierror" + "Td" * "derror") + "
ierrorBc";
25 "CVsinSat" := "ActuadorLampara";
26
27 // Saturacion de la salida del actuador
28 IF ("ActuadorLampara" >= 100) THEN
29 "ActuadorLampara" := 100;
30 ELSIF ("ActuadorLampara" <= 0) THEN
31 "ActuadorLampara" := 0;
32 END_IF;
33
34 // Anti-windup por back-calculation
35 "errorSat" := "ActuadorLampara" - "CVsinSat";
36 "K3" := "Kb";
37 "x3" := "x3" + "K3" * "errorSat" * "dt";
38 "ierrorBc" := "x3";

```

Código 3.1 Control PID nativo con ganancias dependientes en Siemens ST

### 3.7. Diseño e implementación del sistema de control secuencial

#### 3.7.1. Rutina de control proceso secuencial

La Figura 3.32 muestra el peldaño 1. Este invoca la rutina GRAPH para activar el control secuencial. Esta llamada es indispensable, ya que, sin su ejecución, la lógica secuencial permanecería inactiva.

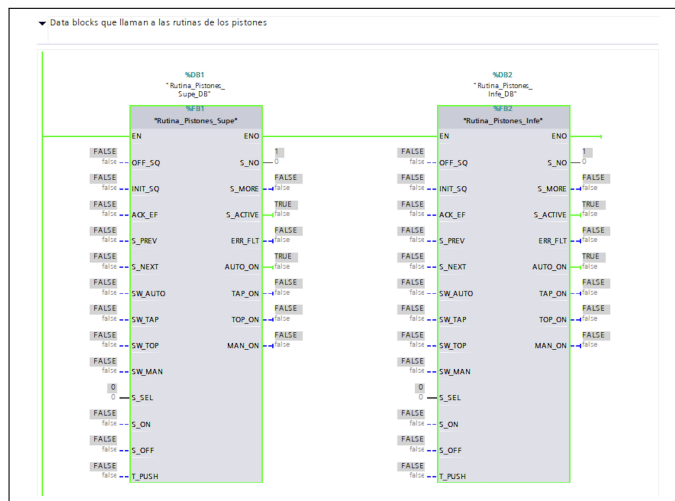


Figura 3.32 Peldaño 1 de la rutina LD del proceso secuencial.<sup>3.32</sup>

La Figura 3.33 muestra el peldaño 2, en el cual se observa la lógica de activación de la rutina. En este peldaño se emplean tres variables en paralelo: **Start\_Supe**, **StartV** y **Paro**, que permiten habilitar la ejecución del bloque correspondiente. Asimismo, se utilizan dos variables en serie: **Stop\_Supe** y **StopV**, las cuales actúan como condiciones de interrupción del proceso. Por un lado, **Stop\_Supe** corresponde al botón físico disponible en la mesa electroneumática; por otro lado, **StopV** corresponde a una marca interna que permite detener el sistema desde la computadora. Esta configuración garantiza que la variable **Paro** tenga prioridad al momento de detener la rutina **Pistones\_Supe**, asegurando un paro controlado cuando se presiona el botón **Stop\_Supe**.

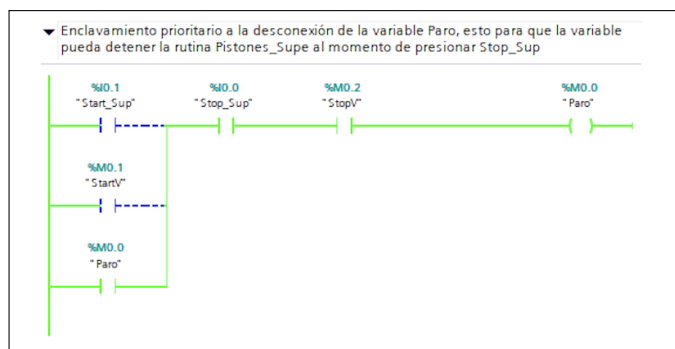


Figura 3.33 Peldaño 2 de la rutina LD del proceso secuencial.<sup>3.33</sup>

<sup>3.32</sup> Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

<sup>3.33</sup> Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

La Figura 3.34 muestra el peldaño 3, en el cual se implementa la misma lógica utilizada en el peldaño 2 (Figura 3.33), con el propósito de mantener el enclavamiento y el control de las variables de paro y arranque correspondientes a los pistones inferiores.

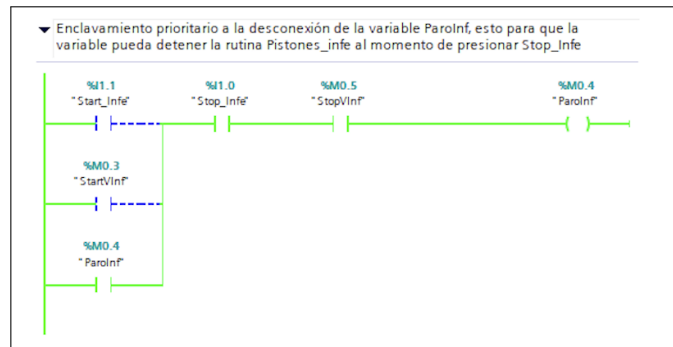


Figura 3.34 Peldaño 3 de la rutina LD del proceso secuencial.<sup>3.34</sup>

La Figura 3.35 muestra de forma general la rutina implementada en el entorno GRAPH, la cual es invocada en el peldaño 1 (Figura 3.32). En ella se observa una estructura secuencial compuesta por siete pasos (*Steps*) y siete transiciones (*Trans*), que representan el flujo lógico de ejecución del proceso. Cada paso contiene las acciones asociadas a una etapa específica del sistema, mientras que las transiciones definen las condiciones bajo las cuales se avanza al siguiente estado, garantizando así un control ordenado y seguro del proceso.

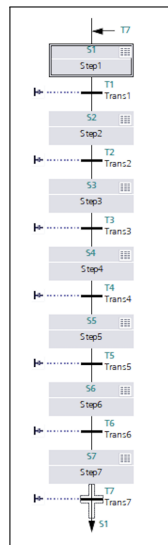


Figura 3.35 Vista general de la rutina hecha en GRAHP.<sup>3.35</sup>

En la Figura 3.36 se muestran las condiciones del primer *Step*, donde se inicializan las variables del sistema y se definen las acciones de salida correspondientes. En este paso se desactivan las señales de extensión de los pistones superiores (*EV\_EX1*, *EV\_EX2* y *EV\_EX3*) y se activan las de contracción (*EV\_CT1*, *EV\_CT2* y *EV\_CT3*), estableciendo así el estado inicial del proceso secuencial. Esta configuración

<sup>3.34</sup>Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

<sup>3.35</sup>Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

garantiza que todos los actuadores comiencen en una posición segura antes de continuar con la siguiente etapa.

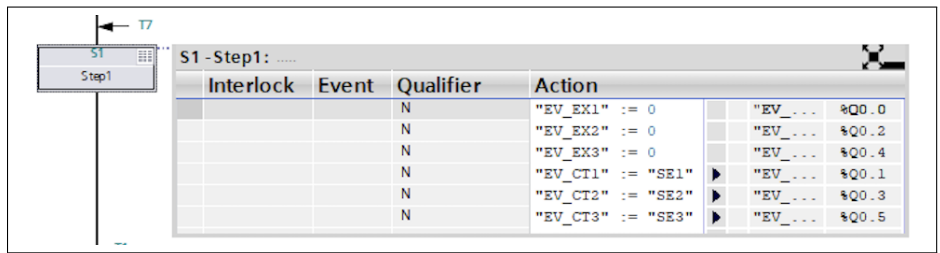


Figura 3.36 Paso 1 del proceso secuencial.<sup>3.36</sup>

En la Figura 3.37 se muestra la Transición 1, la cual corresponde a una lógica implementada en el lenguaje LD. En ella se emplean las variables Start\_Sup y StartV conectadas en paralelo, permitiendo que cualquiera de las dos active el avance desde el Step 1 hacia el Step 2.



Figura 3.37 Transición 1 del proceso secuencial.<sup>3.37</sup>

En la Figura 3.38 se muestra el Step 2, en el cual se ejecuta la acción asociada a la extensión del primer vástago. En este paso, la variable EV\_EX1 se activa mediante la condición SC1, permitiendo el avance del actuador correspondiente. Esta etapa representa el inicio del movimiento secuencial de los pistones dentro del proceso.

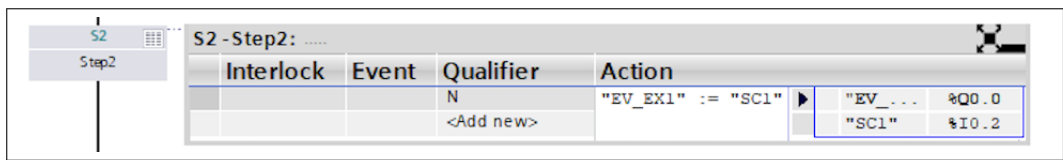


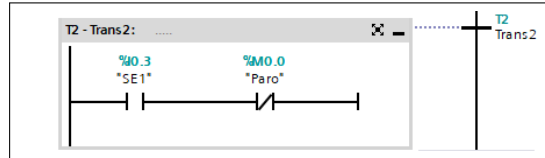
Figura 3.38 Paso 2 del proceso secuencial.<sup>3.38</sup>

En la Figura 3.39 se muestra la Transición 2, la cual se activa cuando la señal del sensor SE1 es detectada y la condición Paro permanece inactiva. Esta lógica permite que el avance al siguiente Step sólo se produzca cuando el primer vástago haya alcanzado su posición extendida y no se haya presionado el botón de paro.

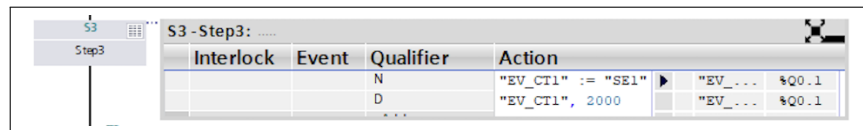
<sup>3.36</sup>Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

<sup>3.37</sup>Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

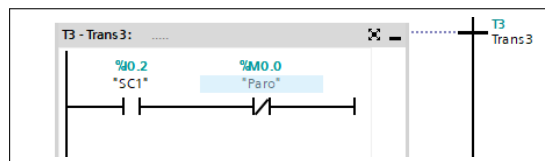
<sup>3.38</sup>Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

Figura 3.39 Transición 2 del proceso secuencial.<sup>3.39</sup>

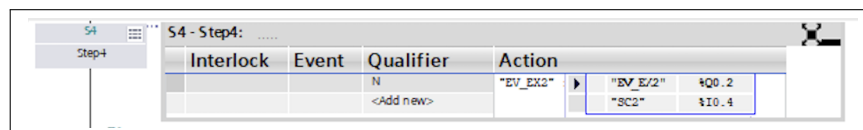
En la Figura 3.40 se muestra el *Step 3*, en el cual se activa la variable *EV\_CT1*, vinculada al sensor *SE1*, provocando la contracción del actuador y permitiendo su retorno a la posición inicial. Además, se aplica un retardo temporal de 2000 ms, con el propósito de retardar la activación de la siguiente transición.

Figura 3.40 Paso 3 del proceso secuencial.<sup>3.40</sup>

En la Figura 3.41 se muestra la Transición 3, la cual se activa cuando el sensor *SC1* detecta la posición completamente contraída del primer vástago, siempre que la condición *Paro* permanezca inactiva. Esta condición permite el avance al siguiente *Step* del proceso secuencial.

Figura 3.41 Transición 3 del proceso secuencial.<sup>3.41</sup>

En la Figura 3.42 se muestra el *Step 4*, en el cual se activa la variable *EV\_EX2*, encargada de extender el segundo vástago del sistema. Esta acción se ejecuta al cumplirse la condición establecida en la transición anterior, permitiendo que dicho actuador realice su desplazamiento de avance hasta alcanzar la posición detectada por el sensor *SC2*.

Figura 3.42 Paso 4 del proceso secuencial.<sup>3.42</sup>

En la Figura 3.43 se muestra la Transición 4, la cual se activa cuando el sensor *SE2* detecta la posición extendida del segundo vástago, siempre que la condición *Paro* permanezca inactiva. Esta lógica permite avanzar al siguiente *Step* del proceso.

<sup>3.39</sup> Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

<sup>3.40</sup> Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

<sup>3.41</sup> Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

<sup>3.42</sup> Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

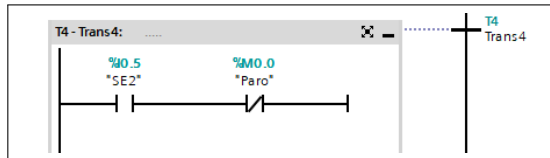


Figura 3.43 Transición 4 del proceso secuencial.<sup>3.43</sup>

En la Figura 3.44 se muestra el *Step 5*, en el cual se activa la variable *EV\_CT2*, asociada al sensor *SE2*, provocando la contracción del segundo vástago y su retorno a la posición inicial. Al igual que en el *Step 3*, se incorpora un retardo temporal de 2000 ms antes de avanzar a la siguiente etapa del proceso secuencial.

Interlock	Event	Qualifier	Action
		N	"EV_CT2" := "SE2"   ▶  "EV_CT2"   %Q0.3
		D	"EV_CT2", 2000   ▶  "EV_CT2"   %Q0.3

Figura 3.44 Paso 5 del proceso secuencial.<sup>3.44</sup>

En la Figura 3.45 se muestra la Transición 5, la cual se activa cuando el sensor *SC2* detecta que el segundo vástago se ha contraído completamente, siempre que la condición *Paro* permanezca inactiva. Esta transición habilita el avance hacia la siguiente etapa del proceso.

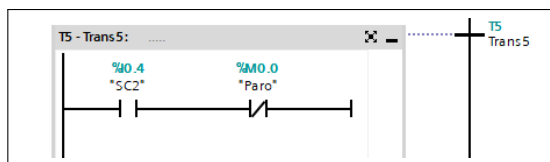


Figura 3.45 Transición 5 del proceso secuencial.<sup>3.45</sup>

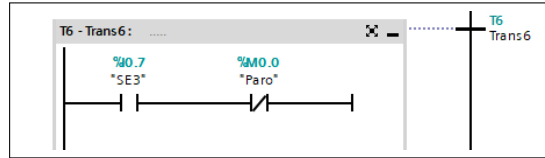
En la Figura 3.46 se muestra el *Step 6*, en el cual se activa la variable *EV\_EX3*, encargada de extender el tercer vástago del sistema. Esta acción se ejecuta cuando se cumplen las condiciones establecidas en la transición anterior, permitiendo el desplazamiento del actuador hasta que el sensor *SC3* confirme que ha alcanzado su posición extendida.

Interlock	Event	Qualifier	Action
		N	"EV_EX3" := "SC3"   ▶  "EV_EX3"   %Q0.4

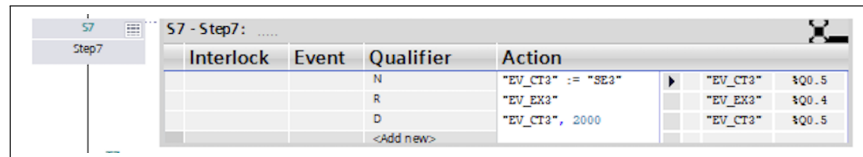
Figura 3.46 Paso 6 del proceso secuencial.<sup>3.46</sup>

En la Figura 3.47 se muestra la Transición 6, la cual se activa cuando el sensor *SE3* detecta que el tercer vástago ha alcanzado su posición completamente extendida, siempre que la condición *Paro* permanezca inactiva. Esta lógica habilita el avance hacia el siguiente *Step*

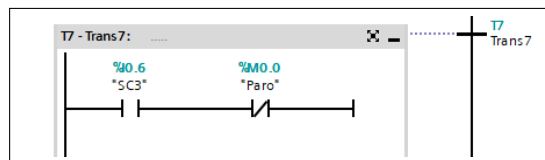
<sup>3.43</sup>Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.  
<sup>3.44</sup>Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.  
<sup>3.45</sup>Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.  
<sup>3.46</sup>Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

Figura 3.47 Transición 6 del proceso secuencial.<sup>3.47</sup>

En la Figura 3.48 se muestra el *Step 7*, en el cual se activa la variable *EV\_CT3*, encargada de realizar la contracción del tercer vástago y devolverlo a su posición inicial. Además, se establece un retardo de 2000 ms antes de concluir la secuencia y de reiniciar el ciclo del proceso.

Figura 3.48 Paso 7 del proceso secuencial.<sup>3.48</sup>

En la Figura 3.49 se muestra la Transición 7, la cual se activa cuando el sensor *SC3* detecta que el tercer vástago se ha contraído completamente, siempre que la condición *Paro* permanezca inactiva. Esta transición marca el final de la secuencia y permite el retorno al *Step 1*, reiniciando el ciclo de operación del sistema.

Figura 3.49 Transición 7 del proceso secuencial.<sup>3.49</sup>

Con ello, se completa el ciclo secuencial del proceso, en el cual los tres vástagos realizan de forma coordinada sus movimientos de extensión y contracción. La estructura desarrollada en GRAPH garantiza una operación ordenada, segura y repetible, cumpliendo con la filosofía *GRAFSET* y la norma IEC 61131-3.

Una vez comprendido el funcionamiento de esta rutina, se aplicó la misma lógica de control para los pistones ubicados en la parte inferior del sistema.

<sup>3.47</sup> Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

<sup>3.48</sup> Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.

<sup>3.49</sup> Captura de pantalla tomada a partir de la programación desarrollada para el control secuencial.



## Capítulo 4

# Evaluación y discusión de resultados

En este capítulo se presenta el análisis de las estrategias de control implementadas, con el propósito de observar su comportamiento ante distintas condiciones de operación y validar el desempeño funcional de los procesos integrados. Para ello, se consideran dos niveles de análisis: por un lado, el proceso continuo asociado al sistema térmico y, por otro, la validación del proceso secuencial correspondiente a la mesa electroneumática.

En el caso del proceso continuo, se comparan los resultados obtenidos mediante el PID\_Compact y el PID nativo, ambos aplicados al proceso térmico descrito previamente. La evaluación se centra en el análisis de la respuesta del sistema en términos de variables características del control, tales como el tiempo de subida, el tiempo de establecimiento, el sobreimpulso y la estabilidad, permitiendo así identificar las principales diferencias en el desempeño de cada controlador.

En cuanto al proceso secuencial, la evaluación consiste en verificar la correcta ejecución de la lógica programada mediante el monitoreo de cada etapa y la observación de la respuesta física de los actuadores. De esta manera, se comprueba que la secuencia implementada se desarrolla de forma ordenada, coherente y repetible, validando así el funcionamiento integral de la arquitectura de control propuesta.

### 4.1. Evaluación funcional del proceso continuo

Para llevar a cabo la evaluación del sistema, se implementó la rutina de control descrita en la Sección 3.6.3. Inicialmente, el sistema parte de una temperatura de 30 °C, con el objetivo de establecer condiciones iniciales similares para ambos controladores, minimizando así variaciones debidas a factores externos como las condiciones ambientales.

Posteriormente, se aplica una secuencia de cambios en el SP con el fin de analizar la respuesta dinámica del sistema. En primera instancia, el SP se incrementa a 80 °C y se mantiene durante un intervalo de un minuto y medio, permitiendo evaluar características como el tiempo de subida, el sobreimpulso y el tiempo de establecimiento.

A continuación, el SP se reduce a 60 °C, para posteriormente incrementarse a 70 °C y finalmente regresar a 80 °C. Esta secuencia de cambios permite analizar el comportamiento del sistema ante variaciones tanto ascendentes como descendentes, así como la capacidad de seguimiento del controlador y su estabilidad en diferentes condiciones de operación.

Con el objetivo de realizar una comparación equivalente entre ambos controladores, se definieron los mismos parámetros de configuración en cada uno de ellos, como se muestra en la Figura 3.31 y en

el Código 3.1. De esta manera, se garantiza que las diferencias observadas en la respuesta del sistema se deban principalmente al alcance de cada controlador.

En ambos casos se utilizaron los mismos parámetros de sintonización del controlador, los cuales se muestran a continuación:

$$K_c = 2.0, \quad T_i = 12.0 \text{ s}, \quad T_d = 0.005 \text{ s}, \quad N = 10.0, \quad dt = 0.01 \text{ s}$$

donde  $K_c$  corresponde a la ganancia proporcional,  $T_i$  al tiempo integral,  $T_d$  al tiempo derivativo,  $N$  al coeficiente del filtro derivativo y  $dt$  al tiempo de muestreo del algoritmo de control.

Con el fin de complementar el análisis de las respuestas obtenidas, se emplearon indicadores clásicos de desempeño para ambos controladores. En particular, se consideraron el sobreimpulso (Ecuación (4.1)), el tiempo de subida (Ecuación (4.2)) y el error en estado estacionario (Ecuación (4.3)), con el propósito de cuantificar las diferencias observadas en la respuesta del sistema.

El sobreimpulso se determinó mediante:

$$M_p = \frac{y_{\text{máx}} - y_{ss}}{y_{ss} - y_0} \times 100 \quad (4.1)$$

donde:

- $y_{\text{máx}}$  representa el valor máximo alcanzado por la respuesta.
- $y_{ss}$  el valor final asociado al SP.
- $y_0$  el valor inicial antes del cambio de referencia.

Por su parte, el tiempo de subida se calculó como:

$$t_r = t_{0.9} - t_{0.1} \quad (4.2)$$

donde:

- $t_{0.1}$  y  $t_{0.9}$  corresponden a los instantes en que la respuesta alcanza el 10 % y el 90 % del cambio aplicado, respectivamente.

Finalmente, el error en estado estacionario se obtuvo de:

$$e_{ss} = |SP - PV_{ss}| \quad (4.3)$$

donde:

- $SP$  es el valor deseado.
- $PV_{ss}$  es el valor de la variable en estado estacionario

#### 4.1.1. PID Compact

En la Figura 4.1 se muestra la respuesta del sistema utilizando el bloque `PID_Compact` ante la secuencia de cambios en el valor de referencia. En la gráfica se presentan la temperatura del sistema, el SP y la señal del actuador.

### **Evaluación del comportamiento dinámico**

El sistema presenta una respuesta subamortiguada ante los cambios de referencia. En el primer escalón de 30 °C a 80 °C, la temperatura asciende con rapidez hasta alcanzar el SP, presentando un sobreimpulso aproximado del 8.64%. Este comportamiento indica que el controlador prioriza una respuesta rápida, aunque con una mayor tendencia al sobrepaso durante la respuesta transitoria.

En cuanto al tiempo de subida, el sistema alcanza el intervalo comprendido entre el 10% y el 90% del cambio en aproximadamente 12.0 s, lo que evidencia una respuesta rápida y adecuada para la dinámica del proceso térmico. Posteriormente, la variable de proceso converge al SP sin presentar oscilaciones, lo que indica un adecuado amortiguamiento para el sistema. Este comportamiento refleja una respuesta estable una vez superado el transitorio principal.

### **Evaluación del estado estacionario**

La respuesta en estado estacionario del controlador presenta un error prácticamente nulo, ya que la temperatura converge y se mantiene cercana al valor de referencia. Asimismo, en los cambios intermedios de 60 °C a 70 °C y de 70 °C a 80 °C, el sistema conserva una adecuada capacidad de seguimiento, alcanzando cada nuevo punto de operación con estabilidad y con sobreimpulsos menores respecto al primer escalón.

Este comportamiento indica que, una vez superada la respuesta transitoria inicial, el controlador mantiene una regulación precisa de la temperatura y una respuesta consistente ante variaciones sucesivas del SP.

### **Análisis de la señal de control**

La señal del actuador presenta incrementos pronunciados durante los cambios de referencia, particularmente en el primer escalón, donde se alcanza la saturación inicial para acelerar el calentamiento del sistema. Conforme la temperatura se aproxima al SP, la señal de control disminuye progresivamente, permitiendo la transición hacia una zona de operación más estable.

La configuración del coeficiente del filtro derivativo dentro del bloque ( $N = 10.0$ ) contribuye a limitar la sensibilidad de la acción derivativa frente a componentes de alta frecuencia y al ruido de medición. Asimismo, el controlador gestiona adecuadamente la saturación de la salida, mitigando el efecto de *windup* y favoreciendo una recuperación estable del sistema durante los transitorios.

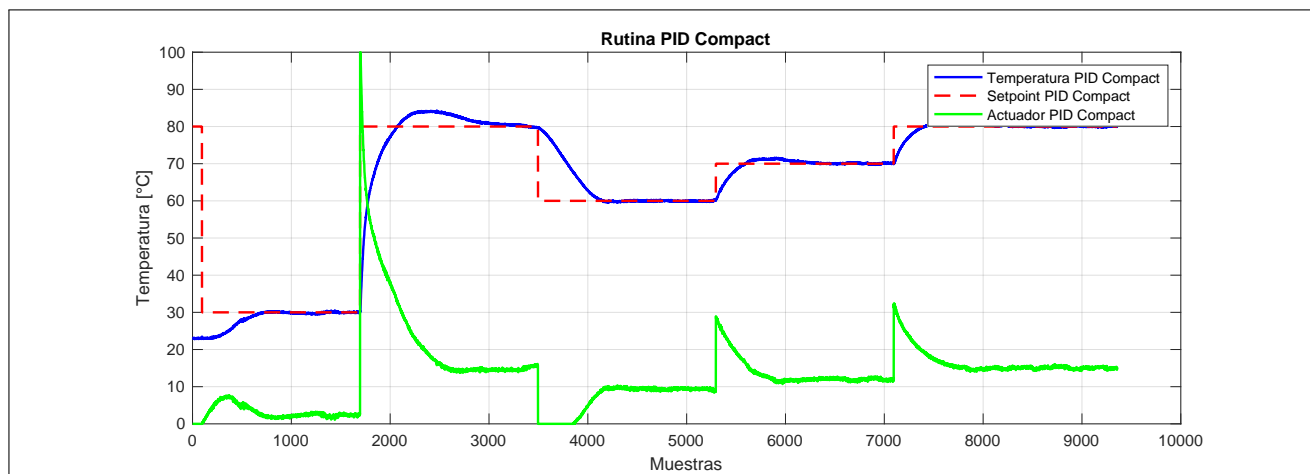


Figura 4.1 Gráfica PID Compact.<sup>4.1</sup>

#### 4.1.2. PID nativo

En la Figura 4.2 se muestra la respuesta del sistema utilizando el PID nativo ante la misma secuencia de cambios en el SP. En la gráfica se presentan nuevamente la temperatura del sistema, el SP y la señal del actuador.

#### Evaluación del comportamiento dinámico

Al igual que en el caso del PID\_Compact, el sistema presenta una respuesta subamortiguada ante los cambios de referencia. En el primer escalón de 30 °C a 80 °C, la temperatura asciende de manera progresiva hasta alcanzar el SP, presentando un sobreimpulso aproximado del 7.88 %. Este comportamiento indica una respuesta moderada ante el cambio de referencia.

En cuanto al tiempo de subida, el sistema alcanza el intervalo comprendido entre el 10 % y el 90 % del cambio en aproximadamente 13.3 s, lo que refleja una respuesta progresiva y adecuada para la dinámica del proceso térmico. Posteriormente, la variable de proceso converge al SP sin presentar oscilaciones, lo que indica un amortiguamiento adecuado del sistema.

#### Evaluación del estado estacionario

De igual forma, en el estado estacionario el controlador presenta un error prácticamente nulo, ya que la temperatura converge y se mantiene cercana al valor del SP. Asimismo, en los cambios intermedios de 60 °C a 70 °C y de 70 °C a 80 °C, el sistema conserva una adecuada capacidad de seguimiento, alcanzando cada nuevo punto de operación con estabilidad y con sobreimpulsos reducidos.

Este comportamiento indica que, una vez superada la respuesta transitoria inicial, el controlador mantiene una regulación precisa de la temperatura y un comportamiento consistente ante variaciones sucesivas del SP.

<sup>4.1</sup>Gráfica generada a partir de datos experimentales recabados.

### Análisis de la señal de control

Como ocurre en el PID\_Compact, la señal del actuador alcanza valores elevados durante el primer escalón con el propósito de acelerar el calentamiento del sistema. No obstante, durante las etapas de transición la señal de control presenta una variación más moderada, lo que sugiere una acción más gradual sobre el actuador.

Conforme la temperatura se aproxima al SP, la señal de control disminuye progresivamente, permitiendo una transición estable hacia el estado estacionario. Asimismo, la implementación explícita del coeficiente del filtro derivativo ( $N = 10.0$ ) contribuye a suavizar la acción derivativa y a reducir la sensibilidad frente al ruido de medición.

De igual forma, el esquema de *anti-windup* implementado mediante *back-calculation* permite limitar la acumulación excesiva de la acción integral cuando la señal de control alcanza los límites de saturación, favoreciendo una recuperación estable del sistema y una transición más adecuada hacia la zona lineal de operación.

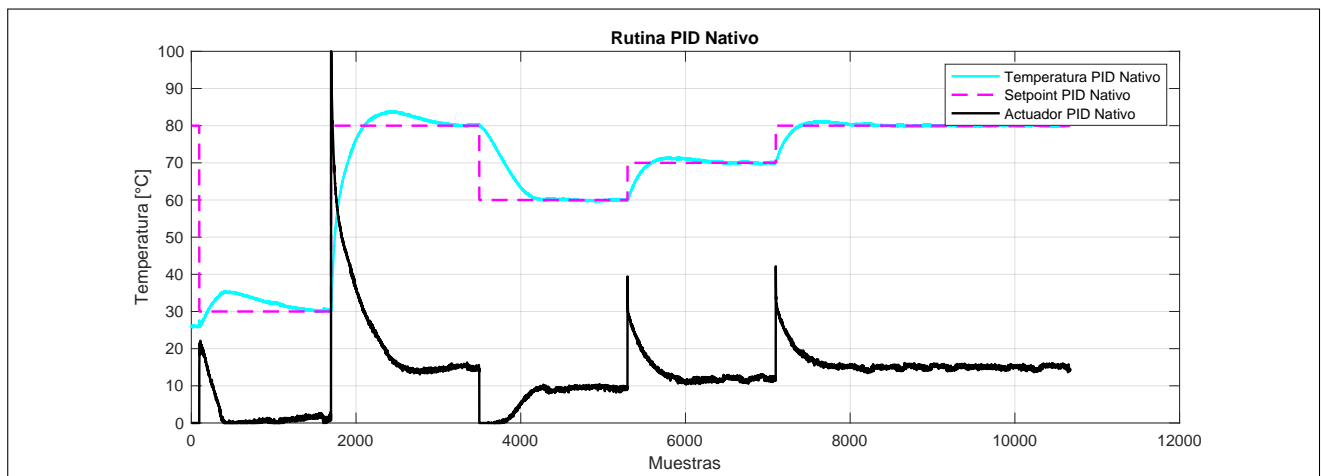


Figura 4.2 Gráfica PID nativo.<sup>4.2</sup>

#### 4.1.3. PID Compact vs PID nativo

A partir de los criterios descritos, se obtuvo que el PID\_Compact presenta un tiempo de subida aproximado de 12.0 s, mientras que el PID nativo alcanza el mismo intervalo en aproximadamente 13.3 s. Asimismo, el sobreimpulso calculado para el PID\_Compact fue de 8.64 %, mientras que para el PID nativo se obtuvo un valor cercano al 7.90 %. Por otra parte, el tiempo de establecimiento fue de aproximadamente 64.9 s para el bloque PID\_Compact y de 64.0 s para el PID nativo.

En términos generales, estos resultados muestran que ambos controladores presentan un desempeño similar en cuanto a la estabilización final del sistema, pero difieren en la forma en que responden durante el estado transitorio. En particular, el PID\_Compact alcanza el valor de referencia con una dinámica ligeramente más rápida, mientras que el PID nativo presenta una evolución más contenida, reflejada en un sobreimpulso menor.

<sup>4.2</sup>Gráfica generada a partir de datos experimentales recabados.

Con el fin de analizar con mayor detalle las diferencias observadas, se superpusieron ambas gráficas en una sola, como se muestra en la Figura 4.3. A continuación, se presentan acercamientos de cada tramo de la respuesta térmica y de la señal del actuador.

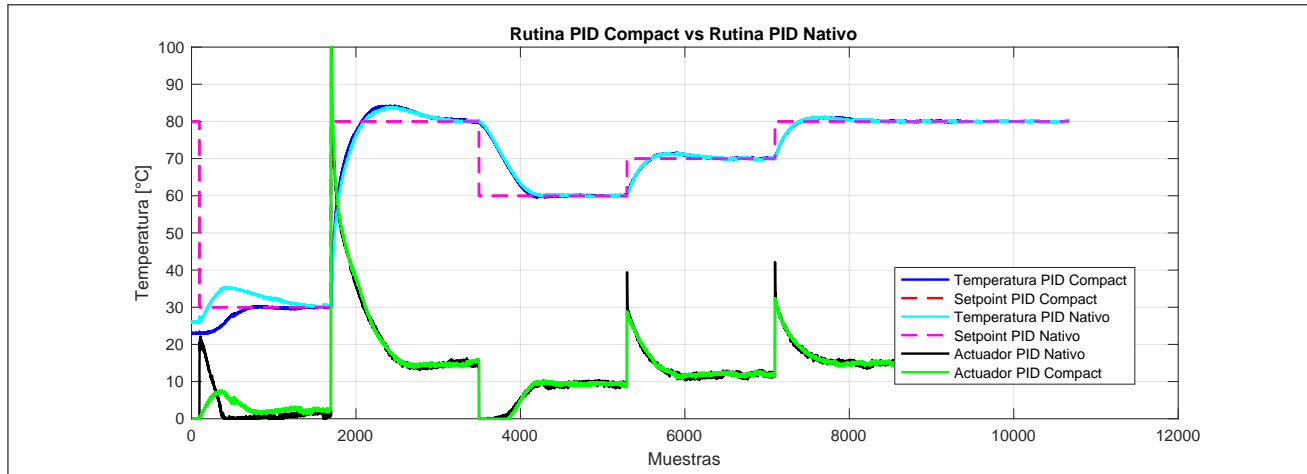


Figura 4.3 Gráfica comparativa entre ambos controladores.<sup>4.3</sup>

En el primer escalón de 30 °C a 80 °C, se observa (ver Figura 4.4) con mayor claridad la principal diferencia entre ambos controladores. El bloque PID\_Compact presenta una respuesta inicial más rápida, ya que la temperatura incrementa con una pendiente ligeramente mayor y alcanza antes la región cercana al SP. No obstante, esta mayor rapidez también se acompaña de un sobreimpulso más pronunciado. Por su parte, el PID nativo sigue una trayectoria similar, aunque con una evolución más progresiva y un pico máximo menor, lo que indica una respuesta más contenida durante el régimen transitorio.

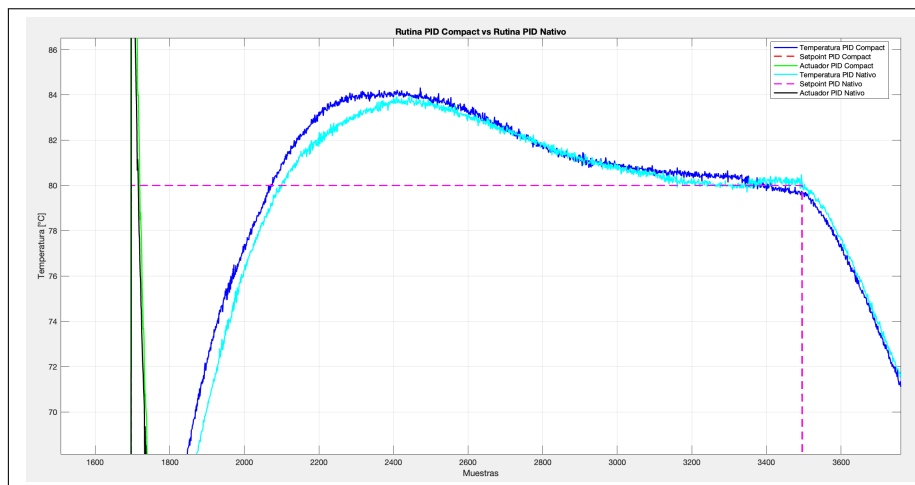


Figura 4.4 Acercamiento de la Figura 4.3 en 80 °C.<sup>4.4</sup>

En la transición descendente de 80 °C a 60 °C (ver Figura 4.5) también se observa un comportamiento muy similar entre ambos controladores. No obstante, el bloque PID\_Compact presenta

<sup>4.3</sup>Gráfica generada a partir de datos experimentales recabados.

<sup>4.4</sup>Gráfica generada a partir de datos experimentales recabados.

una caída ligeramente más directa hacia la nueva referencia, mientras que el PID nativo sigue una trayectoria apenas más progresiva. A pesar de esta diferencia inicial, una vez que la temperatura alcanza la vecindad de 60 °C, ambas respuestas se estabilizan de forma prácticamente equivalente.

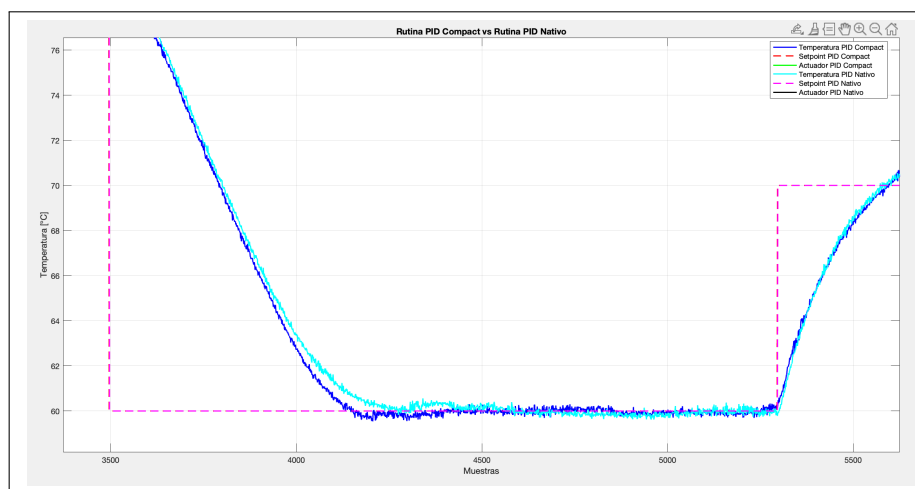


Figura 4.5 Acercamiento de la Figura 4.3 en 60 °C.<sup>4.5</sup>

En la gráfica de acercamiento correspondiente al cambio de 60 °C a 70 °C (Figura 4.6), se observa que ambos controladores presentan un comportamiento muy similar durante la transición hacia el nuevo valor de referencia. Tanto el PID\_Compact como el PID nativo inician el ascenso prácticamente al mismo tiempo y siguen trayectorias muy próximas, con un sobreimpulso ligero en ambos casos. No obstante, se aprecia que el PID\_Compact desarrolla una respuesta ligeramente más rápida en la fase inicial del ascenso, mientras que el PID nativo presenta una evolución algo más gradual. Una vez superado el máximo transitorio, ambas respuestas convergen nuevamente hacia el SP y permanecen cercanas al estado estacionario, lo que indica un desempeño muy similar ante cambios intermedios de referencia.

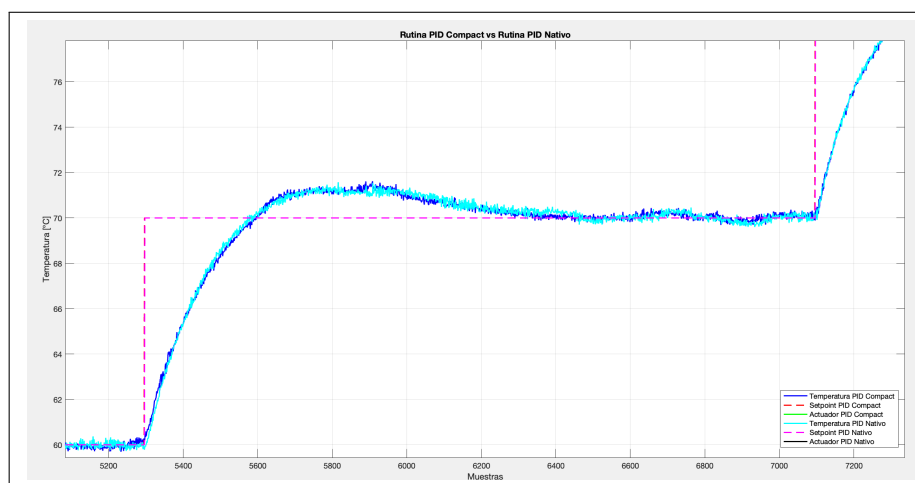


Figura 4.6 Acercamiento de la Figura 4.3 en 70 °C.<sup>4.6</sup>

<sup>4.5</sup>Gráfica generada a partir de datos experimentales recabados.

<sup>4.6</sup>Gráfica generada a partir de datos experimentales recabados.

En la gráfica de acercamiento correspondiente al cambio de 70 °C a 80 °C (Figura 4.7), se observa nuevamente que ambos controladores presentan un comportamiento prácticamente equivalente durante la transición hacia el nuevo valor de referencia. Tanto el PID\_Compact como el PID nativo inician el ascenso al mismo tiempo y siguen trayectorias muy próximas a lo largo de todo el transitorio. Ambas respuestas presentan un sobreimpulso ligero y, una vez superado el máximo transitorio, convergen nuevamente hacia el SP con variaciones pequeñas alrededor del estado estacionario. Esto indica que, para este cambio de referencia, las diferencias dinámicas entre ambos controladores son mínimas y el desempeño de seguimiento resulta prácticamente equivalente.

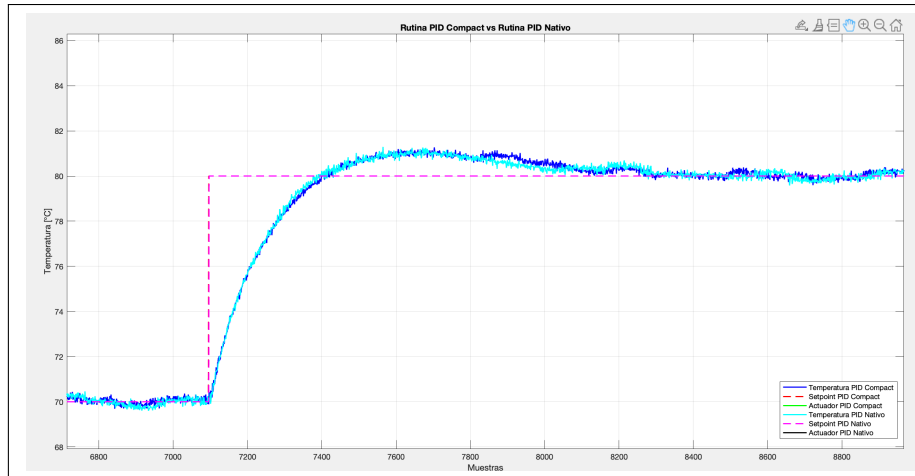


Figura 4.7 Acercamiento de la Figura 4.3 en 80 °C.<sup>4.7</sup>

En el acercamiento de la señal del actuador (Figura 4.8), se observa que ambos controladores presentan una forma similar de generar control, respondiendo con incrementos pronunciados en los instantes donde ocurren cambios bruscos en el SP. No obstante, se aprecian diferencias en la amplitud de los picos y en la presencia de variaciones pequeñas durante la operación.

En particular, el PID nativo genera picos de mayor magnitud en algunos transitorios, especialmente durante el primer escalón y en cambios posteriores de referencia. Por su parte, el PID\_Compact presenta una señal de control más uniforme en ciertos intervalos, aunque mantiene una forma de respuesta similar. Una vez superado cada transitorio, ambas señales convergen a niveles de operación cercanos, lo que indica que los dos controladores alcanzan un esfuerzo de control comparable en régimen de trabajo.

Asimismo, se observa que la señal asociada al PID nativo presenta variaciones pequeñas ligeramente más visibles en algunos tramos, mientras que el bloque PID\_Compact tiende a mostrar una señal algo más uniforme. Estas diferencias sugieren que, aun utilizando los mismos parámetros de sintonización, la forma de implementación influye en la distribución del esfuerzo de control aplicado al actuador.

<sup>4.7</sup>Gráfica generada a partir de datos experimentales recabados.

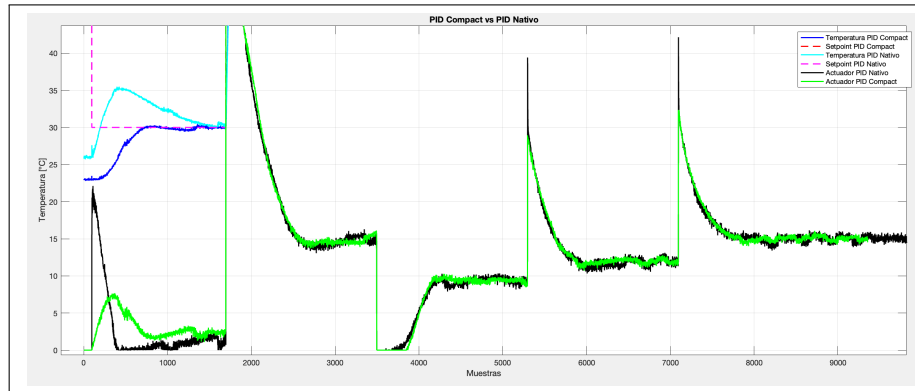


Figura 4.8 Acercamiento de la Figura 4.3 en la señal del actuador.<sup>4.8</sup>

En conjunto, los acercamientos de las respuestas térmicas y de la señal del actuador muestran que la principal diferencia entre ambos controladores se manifiesta en la etapa transitoria inicial, mientras que en los cambios intermedios y en régimen cercano al punto de operación sus comportamientos resultan muy similares. La distinción más clara radica en la forma en que cada controlador distribuye el esfuerzo de control, lo que se refleja en la amplitud de los picos del actuador y en la presencia de variaciones pequeñas en la señal.

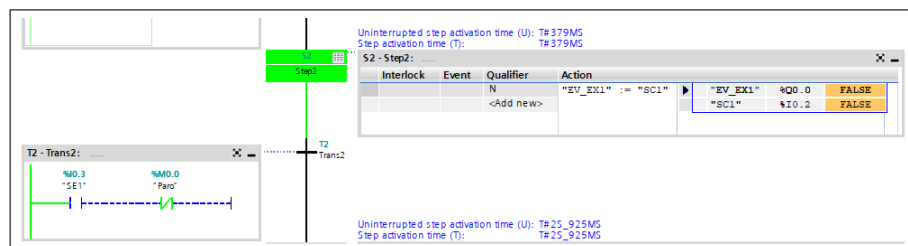
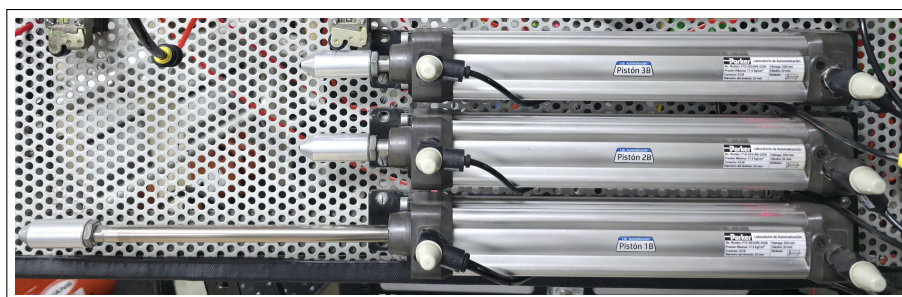
## 4.2. Evaluación funcional del proceso secuencial

La evaluación del proceso secuencial se llevó a cabo con el objetivo de validar el funcionamiento de la rutina de control implementada para la mesa electroneumática, verificando que cada transición y acción se ejecutara en el orden previsto. Para ello, se realizó el monitoreo de los pasos activos dentro de la secuencia desarrollada en GRAPH, contrastando dicho comportamiento con la respuesta física observada en los actuadores neumáticos del sistema.

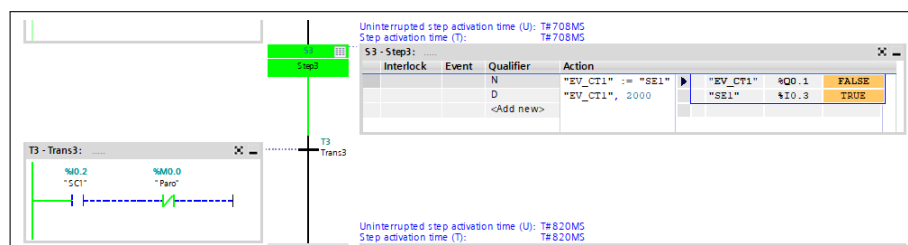
Como parte de esta validación, en la Figura 4.9a se muestra la ejecución del *Step 2*, correspondiente a la etapa de expulsión del pistón 1. En la imagen de monitoreo se observa que el paso se encuentra activo, habilitando la salida asociada a la electroválvula del actuador, mientras que en la evidencia física (Figura 4.9b) se confirma el avance del vástago del pistón 1. Esta correspondencia entre la activación lógica y la respuesta mecánica del actuador permitió comprobar que la secuencia de salida se realiza conforme a lo programado.

Posteriormente, en la Figura 4.10a se presenta la validación del *Step 3*, etapa en la cual se ejecuta la contracción del pistón 1. En este caso, el monitoreo del programa muestra la activación del paso siguiente de la secuencia, mientras que la fotografía del sistema confirma el retorno del vástago a su posición contraída (Figura 4.10b). Lo anterior indica que la transición entre la expulsión y la contracción del pistón 1 se lleva a cabo de forma correcta, manteniendo la coherencia entre la lógica de control y el comportamiento físico del proceso.

<sup>4.8</sup>Gráfica generada a partir de datos experimentales recabados.

(a) Monitoreo del *Step 2* correspondiente a la expulsión del pistón 1.

(b) Pistón 1 expulsado.

Figura 4.9 Validación de la secuencia de expulsión del pistón 1.<sup>4.9</sup>(a) Monitoreo del *Step 3* correspondiente a la contracción del pistón 1.

(b) Pistón 1 contraído.

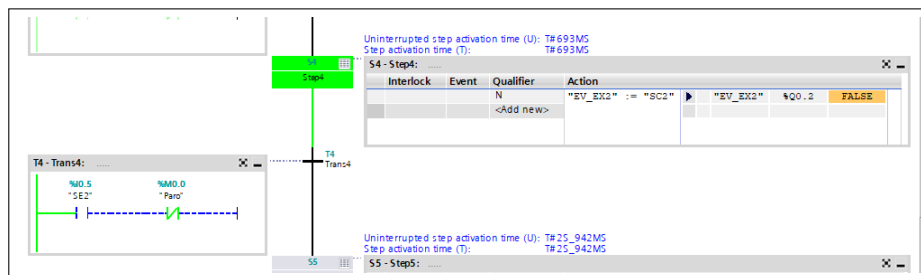
Figura 4.10 Validación de la secuencia de contracción del pistón 1.<sup>4.10</sup>

<sup>4.9</sup>Captura de pantalla de la secuencia programada y fotografía del sistema físico, ambas obtenidas durante la validación experimental.

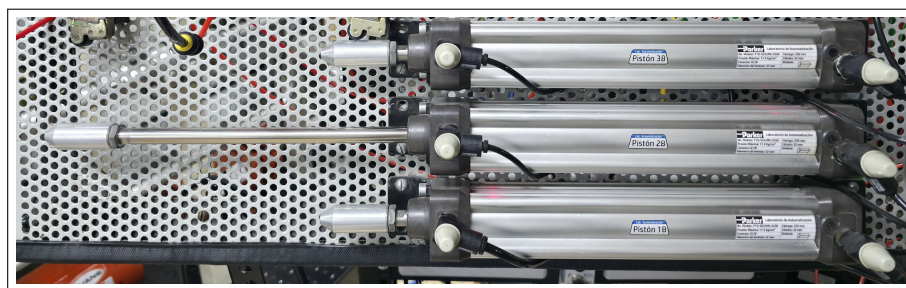
<sup>4.10</sup>Captura de pantalla de la secuencia y fotografía del sistema físico, ambas obtenidas durante la validación experimental.

De manera similar a lo observado en las etapas previas, la evaluación del pistón 2 permitió confirmar el funcionamiento correcto de la secuencia programada. En la Figura 4.11a se presenta el *Step 4*, asociado a la expulsión del pistón 2, donde el monitoreo de la rutina muestra la activación correspondiente y la evidencia física confirma el avance del actuador (4.11b).

En la Figura 4.12a, se observa el *Step 5*, correspondiente a la contracción del pistón 2, verificándose el retorno del vástago a su posición contraída. Su comportamiento físico se observa en la Figura 4.12b. Con ello, se comprueba que la lógica de control mantiene un comportamiento ordenado y coherente con la secuencia establecida.



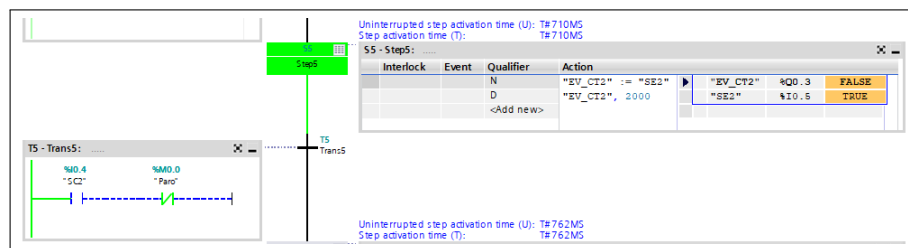
(a) Monitoreo del *Step 4* correspondiente a la expulsión del pistón 2.



(b) Pistón 2 expulsado.

Figura 4.11 Validación de la secuencia de salida del pistón 2.<sup>4.11</sup>

<sup>4.11</sup>Captura de pantalla de la secuencia programada y fotografía del sistema físico, obtenidas durante la validación experimental.

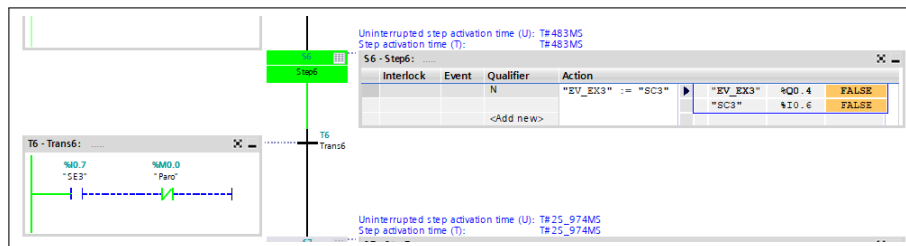
(a) Monitoreo del *Step 5* correspondiente a la contracción del pistón 2.

(b) Pistón 2 contraído.

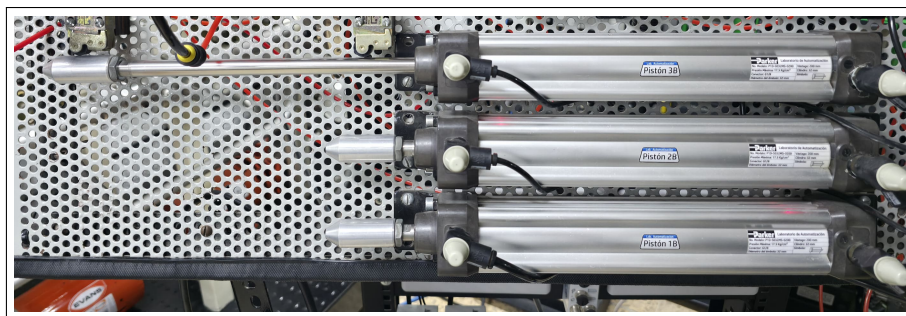
Figura 4.12 Validación de la secuencia de contracción del pistón 2.<sup>4.12</sup>

Finalmente, la validación del pistón 3 permitió corroborar la última parte de la secuencia programada. En la Figura 4.13a se presenta el *Step 6*, asociado a la expulsión del pistón 3 y corroborando de manera física en la Figura 4.13b, mientras que en la Figura 4.14a se muestra la etapa correspondiente a su contracción y de manera física en la Figura 4.14b. Al igual que en los casos anteriores, la correspondencia entre el monitoreo de la rutina y el estado físico del actuador permitió verificar que la lógica implementada se ejecuta de manera consistente, completando así la validación funcional del proceso secuencial.

<sup>4.12</sup>Captura de pantalla de la secuencia programada y fotografía del sistema físico, obtenidas durante la validación experimental.

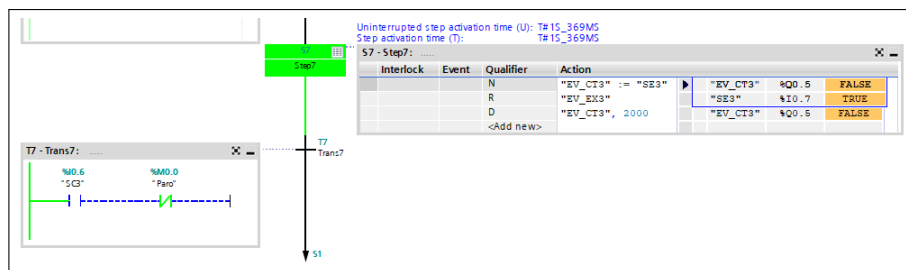


(a) Monitoreo del Step 6 correspondiente a la expulsión del pistón 3.



(b) Pistón 3 expulsado.

Figura 4.13 Validación de la secuencia de salida del pistón 3.<sup>4.13</sup>



(a) Monitoreo del Step 7 correspondiente a la contracción del pistón 3.



(b) Pistón 3 contraído.

Figura 4.14 Validación de la secuencia de contracción del pistón 3.<sup>4.14</sup>

<sup>4.13</sup>Captura de pantalla de la secuencia programada y fotografía del sistema físico, ambas obtenidas durante la validación experimental.

### 4.3. Discusión

La Tabla 4.1 sintetiza el desempeño de ambos controladores. A partir de estos resultados, se observa que el bloque `PID_Compact` presenta un menor tiempo de subida, mientras que el PID nativo exhibe un sobreimpulso ligeramente menor. Asimismo, el tiempo de establecimiento resulta muy similar en ambos casos, lo que indica que los dos controladores alcanzan una estabilización comparable una vez superada la etapa transitoria inicial.

Tabla 4.1 Comparación de indicadores de desempeño de ambos controladores

Parámetro	PID Compact	PID nativo
Tiempo de subida, $t_r$ (s)	12.0	13.3
Sobreimpulso, $M_p$ (%)	8.64	7.90
Tiempo de establecimiento, $t_s$ (s)	64.9	64.0

Estas diferencias sugieren que el bloque `PID_Compact` desarrolla una acción de control más rápida durante el arranque, mientras que el PID nativo presenta una respuesta más moderada. Este comportamiento puede estar asociado no solo a la forma de implementación de cada controlador, sino también al tratamiento interno de la acción derivativa en el bloque `PID_Compact` y a la incorporación explícita del filtro derivativo y del esquema de *anti-windup* mediante *back-calculation* en el PID nativo.

En consecuencia, los resultados muestran un comportamiento clásico entre rapidez de respuesta y sobreimpulso: el `PID_Compact` favorece una transición inicial más rápida, mientras que el PID nativo ofrece una respuesta ligeramente más amortiguada. No obstante, ambos controladores presentan un desempeño satisfactorio para la regulación del proceso continuo.

En términos de aplicación, el bloque `PID_Compact` resulta más favorable cuando se requiere una implementación rápida, con herramientas integradas de ajuste y un buen desempeño general sin necesidad de intervenir explícitamente en la estructura interna del controlador. Por otro lado, el PID nativo es más conveniente en aquellos casos donde se busca un mayor control sobre la lógica del algoritmo, permitiendo incorporar de manera explícita estrategias como el filtrado derivativo y el esquema de *anti-windup* mediante *back-calculation*.

Por otra parte, los resultados obtenidos durante la validación del proceso secuencial permitieron comprobar que la lógica de control implementada ejecuta de manera correcta y ordenada la secuencia programada para la mesa electroneumática. El monitoreo de los *steps* activos y la observación de la respuesta física de los actuadores mostraron que las acciones de expulsión y contracción de los pistones se realizaron conforme a lo establecido en la rutina desarrollada.

Esta evidencia confirma que la arquitectura propuesta es capaz de integrar de forma satisfactoria tareas de automatización secuencial, manteniendo un funcionamiento coherente y repetible. De esta manera, se demuestra que la plataforma desarrollada no solo permite evaluar estrategias de control continuo, sino también validar secuencias discretas dentro de un mismo entorno de automatización.

Si bien la evaluación se enfocó principalmente en la verificación funcional del proceso, los resultados obtenidos son suficientes para respaldar el correcto desempeño de la estrategia implementada. En trabajos futuros, esta validación podría complementarse con análisis relacionados con tiempos de ejecución, robustez ante fallos y respuesta frente a condiciones anormales de operación.

<sup>4.14</sup>Captura de pantalla de la secuencia programada y fotografía del sistema físico, ambas obtenidas durante la validación experimental.

## Capítulo 5

# Conclusiones y trabajo futuro

### 5.1. Conclusiones

En este trabajo se consolidó una plataforma SIMATIC orientada a la evaluación de estrategias de control en procesos industriales continuos y secuenciales, integrando bajo una misma arquitectura un sistema térmico para el análisis de controladores PID y una mesa electroneumática para la validación de rutinas de control secuencial. Esta integración permitió abordar procesos de distinta naturaleza dentro de una estructura común, considerando la configuración del controlador, la organización de señales, la programación de rutinas, la estandarización de conexiones y la comunicación entre los elementos del sistema.

Con ello, se estableció una solución flexible, modular y estandarizada dentro del entorno TIA Portal, capaz de implementar, validar y ampliar distintas tareas de automatización, así como de servir como base para la evaluación de estrategias de control en aplicaciones industriales.

Por un lado, la arquitectura desarrollada demostró ser confiable y escalable, al facilitar la integración del controlador SIMATIC S7-1500, los módulos de I/O, la distribución de señales, el desarrollo de diagramas eléctricos y la implementación de rutinas de control tanto para procesos continuos como secuenciales. Esta integración confirma que la plataforma propuesta no solo habilita la puesta en marcha de distintas estrategias de automatización, sino que también constituye una base adecuada para la incorporación de nuevas funciones y dispositivos.

En lo que respecta al proceso continuo, la implementación del PID\_Compact y del PID nativo hizo posible una comparación funcional entre ambas estrategias aplicadas a la regulación térmica del sistema. Los resultados obtenidos mostraron que ambos controladores presentan un desempeño satisfactorio, confirmando que la arquitectura favorece no solo su implementación, sino también el análisis comparativo de distintas estrategias de control dentro de un entorno controlado.

Por otra parte, la validación del proceso secuencial confirmó que la lógica implementada se ejecuta de manera correcta y ordenada. El monitoreo de los *steps* activos y la observación de la respuesta física de los actuadores comprobaron que cada etapa se desarrolló conforme a la rutina establecida, respaldando así el funcionamiento integral de la arquitectura de control propuesta.

Finalmente, el resultado de este trabajo permitió consolidar una base tecnológica sólida para diseñar, validar y evaluar estrategias de automatización en procesos de distinta naturaleza. Su principal aportación radica en ofrecer un entorno unificado capaz de integrar tareas de control continuo y secuencial dentro de una misma plataforma SIMATIC, facilitando futuras extensiones hacia funciones

de supervisión, monitoreo avanzado, incorporación de nuevas estrategias de control y su eventual transferencia a aplicaciones industriales de mayor complejidad.

## 5.2. Trabajo futuro

Partiendo de la arquitectura desarrollada, esta podría ampliarse hacia funciones que fortalezcan su integración con entornos de manufactura inteligente. Una primera línea de desarrollo podría consistir en incorporar herramientas de supervisión y adquisición de datos más robustas, de modo que la información generada por los procesos pueda almacenarse, organizarse y consultarse de forma estructurada, siguiendo esquemas similares a los empleados en sistemas SCADA o de control distribuido.

Otra línea de trabajo podría enfocarse en desarrollar módulos de análisis y visualización que permitan generar indicadores de desempeño, tendencias operativas, alarmas y reportes asociados al comportamiento de los procesos. Esto permitiría extender el uso de la plataforma más allá de la validación funcional de estrategias de control y emplearla como una herramienta para la toma de decisiones basada en datos, fortaleciendo su utilidad en contextos de digitalización industrial.

De igual manera, la arquitectura podría evolucionar hacia un entorno de experimentación más amplio mediante la incorporación de nuevas estrategias de control, implementar rutinas de diagnóstico, desarrollar esquemas de detección de fallas y utilizar modelos de simulación. En particular, podría considerarse la integración de tecnologías IIoT y entornos de realidad virtual, con el propósito de desarrollar sistemas de supervisión, validación y entrenamiento más avanzados, así como habilitar funciones de monitoreo remoto e interconexión entre dispositivos.

Finalmente, podrían incorporarse nuevos procesos experimentales dentro de la misma estructura, con el fin de reforzar su carácter modular y escalable. Esto permitiría contar con una plataforma unificada capaz de integrar procesos continuos, secuenciales e híbridos, contribuyendo tanto a la validación de estrategias de automatización como a la reducción de limitaciones asociadas a la fragmentación tecnológica y a la necesidad de entornos de prueba más integrados y adaptables.

# Referencias

- [ISA, 1995] (1995). *ISA-88 Batch Control Standard*. International Society of Automation (ISA) <https://gmpua.com/GAMP/ISA-88.pdf>. (Citado en página 9.)
- [Barrios, 2022] Barrios, D. (2022). ¿qué son los controladores pid y cuál es su historia? <https://www.linkedin.com/pulse/quál-son-los-controladores-pid-y-cuál-es-su-historia-darwin-barrios/>. (Citado en página 10.)
- [Berger, 2014] Berger, H. (2014). *Automating with SIMATIC S7-1500: configuring, programming and testing with STEP 7 Professional*. John Wiley & Sons. (Citado en página 15.)
- [Botero, et al., 2016] Botero, M. L., Selmer, A., Watson, R., Bansal, M., & Kraft, M. (2016). Cambridge weblabs: A process control system using industrial standard simatic pcs 7. *Education for Chemical Engineers*, 16, 1–8, <https://doi.org/10.1016/j.ece.2016.04.001> <https://doi.org/10.1016/j.ece.2016.04.001>. (Citado en páginas 2 y 3.)
- [Endara y Rodas, 2024] Endara, L. C. E. & Rodas, K. A. L. (2024). Análisis de la herramienta unity 3d para la creación de procesos virtuales y su automatización con el plc simatic s7-1500. Trabajo de titulación, ingeniería en electrónica, Universidad Politécnica Salesiana, Sede Cuenca, Cuenca, Ecuador. (Citado en páginas 4 y 5.)
- [Kuong, et al., 2023] Kuong, S., Yahoui, H., Thap, T., Tran, T. K., Math, S., & Heng, S. (2023). Industry 4.0: Developing center of excellence for plc lab in the field of automation. In *Proceedings of the 8th International Conference on Digital Arts, Media and Technology (DAMT) and 6th ECTI Northern Section Conference on Electrical, Electronics, Computer and Telecommunications Engineering (NCON)* (pp. 369–373). Phnom Penh, Cambodia: IEEE. (Citado en páginas 3 y 4.)
- [Lozano Valencia, et al., 2012] Lozano Valencia, L. F., Rodríguez García, L. F., & Giraldo Buitrago, D. (2012). Diseño, implementación y validación de un controlador pid autosintonizado. *Revista Tecno Lógicas*, (28), 33–53. (Citado en página 10.)
- [Müller y Oehm, 2019] Müller, R. & Oehm, L. (2019). Process industries versus discrete processing: how system characteristics affect operator tasks. *Cognition, Technology & Work*, 21(3), 337–356, <https://doi.org/10.1007/s10111-018-0511-1> <https://doi.org/10.1007/s10111-018-0511-1>. (Citado en página 9.)
- [Nagy, et al., 2020] Nagy, Z. K., Hagrasy, A. E., & Litster, J., Eds. (2020). *Continuous Manufacturing of Pharmaceuticals*. AAPS Advances in the Pharmaceutical Sciences Series. Springer <https://doi.org/10.1007/978-3-030-41524-2>. (Citado en página 9.)
- [National Academies of Sciences, Engineering, and Medicine, 2024] National Academies of Sciences, Engineering, and Medicine (2024). *Options for a National Plan for Smart Manufacturing*. Washington, DC: The National Academies Press <https://doi.org/10.17226/27260>. (Citado en página 6.)
- [Ramírez Urrego y Posada Restrepo, 2016] Ramírez Urrego, J. A. & Posada Restrepo, N. L. (2016). Aplicación de diseño, simulación, identificación de sistemas e implementación de controladores pid - digitrol. *Revista Politécnica*, 12(23), 27–37. (Citado en página 10.)

- [Siemens, 2019] Siemens (2019). Kits de inicio simatic: Controladores s7-1500, s7-1200 y mini plc logo! 8 <https://assets.new.siemens.com/siemens/assets/api/uuid:673d0f46-0af9-4bf5-9497-62aec13b5606/version:1562315729/folleto-simatic-starter-kit-julio-2019.pdf>. (Citado en página 14.)
- [Siemens, 2020] Siemens (2020). *SIMATIC PCS 7 – Sequential Control Systems: Sequential Function Chart*. In Siemens Automation Cooperates with Education (SCE). (Citado en página 10.)
- [Siemens, 2023] Siemens (2023). *SIMATIC S7-1200/1500: PID Compact, Function Manual*. Siemens, Nuremberg, Germany, version 3.0 edition [https://cache.industry.siemens.com/dl/files/401/100746401/att\\_1124946/v2/100746401\\_S71x00\\_PidCompact\\_DOC\\_V3.0\\_en.pdf](https://cache.industry.siemens.com/dl/files/401/100746401/att_1124946/v2/100746401_S71x00_PidCompact_DOC_V3.0_en.pdf). (Citado en páginas 13 y 37.)
- [Siemens, 2023] Siemens (2023). *SIMATIC S7-1500/ET 200MP Sistema de automatización Manual de sistema*. (Citado en página 15.)
- [Siemens, 2024a] Siemens (2024a). Programmable logic controllers (plc). <https://www.siemens.com/us/en/products/automation/systems/industrial/plc.html>. (Citado en página 14.)
- [Siemens, 2024b] Siemens (2024b). Simatic – 175 years of siemens <https://www.siemens.com/global/en/company/about/history/specials/175-years/simatic.html>. (Citado en página 13.)
- [Siemens, 2024] Siemens (2024). *STEP 7 y WinCC V19: Software de ingeniería para los controladores SIMATIC y los sistemas HMI*. Siemens AG, Digital Industries, Núremberg, Alemania, versión 19 edition [https://cache.industry.siemens.com/dl/files/862/109826862/att\\_1264397/v1/STEP\\_7\\_WinCC\\_V19\\_esES\\_es-ES.pdf](https://cache.industry.siemens.com/dl/files/862/109826862/att_1264397/v1/STEP_7_WinCC_V19_esES_es-ES.pdf). (Citado en página 16.)
- [Siemens, 2025] Siemens (2025). Tia portal documentation – siemens docs [https://docs.tia.siemens.cloud/r/simatic\\_s7\\_1200\\_manual\\_collection\\_enus\\_20/programming-concepts/understanding-data-consistency](https://docs.tia.siemens.cloud/r/simatic_s7_1200_manual_collection_enus_20/programming-concepts/understanding-data-consistency). (Citado en páginas 16, 17 y 18.)
- [Supriya, et al., 2024] Supriya, Y., , et al. (2024). Industry 5.0 in smart education: Concepts, applications, challenges, opportunities, and future directions. *IEEE Access*, (pp. 1–27)., <https://doi.org/10.1109/ACCESS.2024.3401473> <https://doi.org/10.1109/ACCESS.2024.3401473>. (Citado en página 1.)
- [Vinitha, et al., 2020] Vinitha, K., Ambrose Prabhu, R., Bhaskar, R., & Hariharan, R. (2020). Review on industrial mathematics and materials at industry 1.0 to industry 4.0. *Materials Today: Proceedings*, 33, 3956–3960, <https://doi.org/10.1016/j.matpr.2020.06.331>. (Citado en página 1.)
- [Visioli, 2006] Visioli, A. (2006). *Practical PID Control*. Advances in Industrial Control. London: Springer. (Citado en página 12.)
- [Åström y Hägglund, 1995] Åström, K. J. & Hägglund, T. (1995). *PID Controllers: Theory, Design, and Tuning*. Research Triangle Park, NC: Instrument Society of America, 2 edition. (Citado en página 10.)