

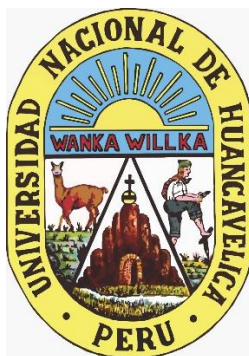
**UNIVERSIDAD NACIONAL DE HUANCABELICA**

*(Creado por Ley N° 25265)*

**FACULTAD DE INGENIERÍA ELECTRÓNICA - SISTEMAS**

**ESCUELA PROFESIONAL DE INGENIERÍA ELECTRÓNICA**

**TESIS**



**IMPLEMENTACIÓN DE UN CONTROLADOR PID  
AUTOSINTONIZABLE PARA PROCESOS  
INDUSTRIALES INTEGRADO A UN PLC CON  
COMUNICACIÓN OPC UA**

**LÍNEA DE INVESTIGACIÓN: AUTOMATIZACIÓN Y CONTROL**

**PRESENTADO POR:**

**BACH. ING. RIKNER RAÚL AQUINO SUAREZ**

**BACH. ING. EMERSON ANCALLE AGUILAR**

**PARA OPTAR EL TÍTULO PROFESIONAL DE:**

**INGENIERO ELECTRÓNICO**

**HUANCAVELICA, PERÚ**

**2022**



"Año del Fortalecimiento de la Soberanía Nacional"

### ACTA DE SUSTENTACIÓN DE TESIS

En la Facultad de Ingeniería Electrónica - Sistemas, Escuela Profesional de Ingeniería Electrónica, local de auditorio de la Facultad FIES- UNH, a los 15 días del mes de Diciembre de 2022 siendo las 10 horas con 05 minutos, se instaló el Jurado Evaluador de Tesis, teniendo como miembros:

1. PRESIDENTE : Dr. Angel Almidon Eles como  
ORCID 0000-0001-9694-8370  
DNI N° 23715269
2. SECRETARIO : MSc Wohler Gonzales Saenz  
ORCID 0000-0003-0728-9479  
DNI N° 23275014
3. VOCAL : Dr. Hipólito Garbajal Morán  
ORCID 0000-0002-1661-5363  
DNI N° 23709147
4. ASESOR : Dr. Hipólito Garbajal Morán  
ORCID 0000-0002-1661-5363  
DNI N° 23709147

Con la finalidad de evaluar la sustentación de tesis, titulado: "Implementación de un controlador PID autosintonizable para procesos industriales, integrado a un PLC con comunicación OPC UA", presentado por la (los) bachiller (es): Rikner Raúl Aquino Suarez, Emerson Analle Aguilar, de la Escuela Profesional de Ingeniería Electrónica, quien(es) ha(n) sido declarado(s) expedito(s) para el presente acto, con Resolución de Consejo de Facultad N° 397-2022-FIES-UNH de fecha 01 de Diciembre del 2022.

Se da inicio a la sustentación de tesis siendo las 10 horas con 10 minutos. Concluido la sustentación, se pasó a la segunda etapa de preguntas y/u observaciones. Enseguida el Jurado deliberó en secreto para obtener el resultado, siendo el veredicto del jurado:

APROBADO ☒ POR Unanimidad


DESAPROBADO ☐

A horas 12:00, se da por concluido el acto, y procedieron a firmar en señal de conformidad.

Ciudad Universitaria Pampas, 15 de Diciembre del 2022

OBSERVACIONES:

  
Presidente

  
Vocal

  
Secretario

**TÍTULO:**

**“IMPLEMENTACIÓN DE UN CONTROLADOR PID  
AUTOSINTONIZABLE PARA PROCESOS  
INDUSTRIALES INTEGRADO A UN PLC CON  
COMUNICACIÓN OPC UA”**

**AUTORES:**

BACH. ING. RIKNER RAÚL AQUINO SUAREZ

BACH. ING. EMERSON ANCALLE AGUILAR

**ASESOR:**

DR. HIPÓLITO CARBAJAL MORÁN

ORCID: <https://orcid.org/0000-0002-1661-5363>

DNI: 23709147

## **AGRADECIMIENTO**

A la Universidad Nacional de Huancavelica por habernos dado la oportunidad de formarnos como profesionales en las aulas de la Escuela Profesional de Ingeniería Electrónica, Facultad de Ingeniería Electrónica-Sistemas.

A los docentes de la Escuela Profesional de Ingeniería Electrónica; quienes nos impartieron sus conocimientos y experiencias, que sirvieron de base para la formulación y el desarrollo de esta tesis.

Al asesor de la Tesis Dr. Hipólito Carbajal Morán, quien nos orientó en todo momento en el marco metodológico que hizo posible con la culminación de este trabajo de investigación.

Finalmente agradecemos a todas las personas que de una u otra manera se involucraron en el desarrollo de esta Tesis; con aportes importantes sobre los PLCs y sus aplicaciones.

# ÍNDICE

TÍTULO .....	iii
AUTORES .....	iv
ASESOR .....	v
AGRADECIMIENTO .....	vi
ÍNDICE .....	vii
ÍNDICE DE TABLAS .....	x
ÍNDICE DE FIGURAS.....	xiv
RESUMEN.....	xviii
ABSTRACT.....	xix
INTRODUCCIÓN .....	xx
CAPÍTULO I.....	22
PLANTEAMIENTO DEL PROBLEMA .....	22
1.1. FUNDAMENTACIÓN DEL PROBLEMA.....	22
1.2. FORMULACIÓN DEL PROBLEMA .....	24
1.2.1. Problema general .....	24
1.2.2. Problemas específicos .....	24
1.3. OBJETIVOS.....	24
1.3.1. Objetivo general .....	24
1.3.2. Objetivos específicos.....	24
1.4. ÁMBITO DE ESTUDIO.....	25
1.5. JUSTIFICACIÓN.....	25
CAPÍTULO II .....	27
MARCO TEÓRICO.....	27
2.1. ANTECEDENTES .....	27
2.2. BASES TEÓRICAS .....	30
2.2.1. Controladores PID .....	30
2.2.2. PLC Simatic S7 1500 .....	38
2.2.3. Procesos industriales .....	51

2.2.4.	Comunicación OPC UA .....	53
2.2.5.	Cientes OPC UA .....	60
2.3.	HIPÓTESIS .....	65
2.3.1.	Hipótesis general .....	65
2.3.2.	Hipótesis específicas .....	65
2.4.	OPERACIONALIZACIÓN DE VARIABLES .....	66
2.4.1.	Variable independiente .....	66
2.4.2.	Variables dependientes .....	66
CAPÍTULO III.....		67
MATERIALES Y MÉTODOS .....		67
3.1.	MATERIALES.....	67
3.1.1.	Equipos .....	67
3.1.2.	Programas de aplicación.....	69
3.2.	TIPO Y NIVEL DE INVESTIGACIÓN.....	70
3.2.1.	Tipo de investigación .....	70
3.2.2.	Nivel de investigación .....	70
3.3.	MÉTODO DE INVESTIGACIÓN .....	71
3.4.	DISEÑO DE INVESTIGACIÓN.....	71
3.5.	POBLACIÓN, MUESTRA Y MUESTREO .....	72
3.5.1.	Población .....	72
3.5.2.	Muestra .....	72
3.5.3.	Muestreo .....	73
3.6.	ANÁLISIS Y PROCESAMIENTO DE DATOS.....	73
CAPÍTULO IV.....		74
DISCUSIÓN DE RESULTADOS .....		74
4.1.	RESULTADOS .....	74
4.1.1.	Implementación del servidor del controlador PID autosintonizable en un PLC S7 1500 con PID_Compact y regla de Ziegler-Nichols, con envío de datos por OPC UA .....	74



4.1.2. Implementación del cliente del controlador PID Autosintonizable.....	87
4.1.3. Determinación del tiempo de asentamiento de la variable <i>Temperatura</i> en un proceso térmico.....	97
4.1.4. Determinación del tiempo de asentamiento de la variable <i>Nivel</i> de líquido en tanque.....	100
4.2. PRUEBA DE HIPÓTESIS.....	104
4.2.1. Prueba de las hipótesis específicas.....	104
4.2.2. Prueba de la hipótesis general.....	111
4.3. DISCUSIÓN.....	111
CONCLUSIONES.....	114
RECOMENDACIONES.....	115
REFERENCIA BIBLIOGRÁFICA.....	116
APÉNDICE.....	122
A. Muestras tomadas para el controlador PID no sintonizado para la variable temperatura del proceso térmico.....	123
B. Muestras tomadas para el controlador PID autosintonizado para la variable temperatura del proceso térmico.....	133
C. Muestras tomadas para el controlador PID no sintonizado para la variable nivel en tanque de 3000 mm de altura.....	143
D. Muestras tomadas para el controlador PID autosintonizado para la variable nivel en tanque de 3000 mm de altura.....	153
E. Fotografías.....	163
F. Diagrama de conexión de entrada/ salida de módulos analógicos para PLC S7 1500.....	170
G. Matriz de consistencia.....	172

## ÍNDICE DE TABLAS

Tabla 1	Regla de sintonía de Ziegler-Nichols basada en la respuesta escalón de la planta.....	35
Tabla 2	Regla de sintonía de Ziegler-Nichols basada en la ganancia crítica $K_{cr}$ y periodo crítico $P_{cr}$ .....	37
Tabla 3	Parámetros de ajuste del controlador.....	38
Tabla 4	Parámetros del controlador PID_Compact.....	45
Tabla 5	Parámetros de entrada del controlador PID_Compact .....	47
Tabla 6	Parámetros de salida del controlador PID_Compact.....	48
Tabla 7	Variables estáticas de PID_Compact.....	49
Tabla 8	Variables almacenadas de ganancia $K_p$ , $T_i$ , $T_d$ , ponderaciones y parámetros calculados durante la optimización de procesos con PID_Compact .....	50
Tabla 9	Tipos de nodos y su descripción.....	58
Tabla 10	Tipo de estructura de un Node ID .....	59
Tabla 11	Sistema de licencias para CPU S7 1500.....	60
Tabla 12	Definición operativa de variables e indicadores.....	66
Tabla 13	Método de sintonización PID de Ziegler-Nichols basada en la ganancia crítica $K_{cr}$ y periodo crítico $P_{cr}$ .....	95
Tabla 14	Resultados de los experimentos con controlador PID no sintonizado para la variable Temperatura de la planta térmica.....	97
Tabla 15	Parámetros PID obtenidos con el método de sintonización PID de Ziegler-Nichols basada en la ganancia crítica $K_{cr}$ y periodo crítico $P_{cr}$ , para la variable controlada Temperatura de la planta térmica .....	98
Tabla 16	Resultados de los experimentos con controlador PID autosintonizado para la variable Temperatura de la planta térmica .....	99
Tabla 17	Resultados de los experimentos con controlador PID no sintonizado para la variable Nivel de líquido en tanque .....	101
Tabla 18	Parámetros PID obtenidos con el método de sintonización PID de Ziegler-Nichols basada en la ganancia crítica $K_{cr}$ y periodo crítico $P_{cr}$ , para la variable controlada Nivel de líquido en tanque .....	102

Tabla 19	Resultados de los experimentos con controlador PID autosintonizado para la variable Nivel de líquido en tanque.....	103
Tabla 20	Resultados de los experimentos con controlador PID no sintonizado y autosintonizado para la variable Temperatura de la planta térmica .....	106
Tabla 21	Prueba t unilateral a la derecha para dos muestras dependientes de la variable Temperatura en planta térmica, para un intervalo de confianza del 95% .....	107
Tabla 22	Resultados de los experimentos con controlador PID no sintonizado y autosintonizado para la variable Nivel de líquido en tanque.....	109
Tabla 23	Prueba t unilateral a la derecha para dos muestras dependientes de la variable Temperatura en planta térmica, para un intervalo de confianza del 95% .....	110
Tabla 24	Mediciones con el controlador PID sin sintonizar para setpoint 25 °C.....	123
Tabla 25	Mediciones con el controlador PID sin sintonizar para setpoint 30 °C.....	124
Tabla 26	Mediciones con el controlador PID sin sintonizar para setpoint 35 °C.....	125
Tabla 27	Mediciones con el controlador PID sin sintonizar para setpoint 40 °C.....	126
Tabla 28	Mediciones con el controlador PID sin sintonizar para setpoint 50 °C.....	127
Tabla 29	Mediciones con el controlador PID sin sintonizar para setpoint 60 °C.....	128
Tabla 30	Mediciones con el controlador PID sin sintonizar para setpoint 70 °C.....	129
Tabla 31	Mediciones con el controlador PID sin sintonizar para setpoint 80 °C.....	130
Tabla 32	Mediciones con el controlador PID sin sintonizar para setpoint 90 °C.....	131

Tabla 33	Mediciones con el controlador PID sin sintonizar para setpoint	
	96 °C.....	132
Tabla 34	Mediciones con el controlador PID autosintonizado para setpoint	
	25 °C.....	133
Tabla 35	Mediciones con el controlador PID autosintonizado para setpoint	
	30 °C.....	134
Tabla 36	Mediciones con el controlador PID autosintonizado para setpoint	
	35 °C.....	135
Tabla 37	Mediciones con el controlador PID autosintonizado para setpoint	
	40 °C.....	136
Tabla 38	Mediciones con el controlador PID autosintonizado para setpoint	
	50 °C.....	137
Tabla 39	Mediciones con el controlador PID autosintonizado para setpoint	
	60 °C.....	138
Tabla 40	Mediciones con el controlador PID autosintonizado para setpoint	
	70 °C.....	139
Tabla 41	Mediciones con el controlador PID autosintonizado para setpoint	
	80 °C.....	140
Tabla 42	Mediciones con el controlador PID autosintonizado para setpoint	
	90 °C.....	141
Tabla 43	Mediciones con el controlador PID autosintonizado para setpoint	
	96 °C.....	142
Tabla 44	Mediciones con el controlador PID sin sintonizar para setpoint	
	250 mm.....	143
Tabla 45	Mediciones con el controlador PID sin sintonizar para setpoint	
	500 mm.....	144
Tabla 46	Mediciones con el controlador PID sin sintonizar para setpoint	
	1000 mm.....	145
Tabla 47	Mediciones con el controlador PID sin sintonizar para setpoint	
	1250 mm.....	146

Tabla 48	Mediciones con el controlador PID sin sintonizar para setpoint	
	1500 mm.....	147
Tabla 49	Mediciones con el controlador PID sin sintonizar para setpoint	
	1750 mm.....	148
Tabla 50	Mediciones con el controlador PID sin sintonizar para setpoint	
	2000 mm.....	149
Tabla 51	Mediciones con el controlador PID sin sintonizar para setpoint	
	2250 mm.....	150
Tabla 52	Mediciones con el controlador PID sin sintonizar para setpoint	
	2500 mm.....	151
Tabla 53	Mediciones con el controlador PID sin sintonizar para setpoint	
	3000 mm.....	152
Tabla 54	Mediciones con el controlador PID autosintonizado para setpoint	
	250 mm.....	153
Tabla 55	Mediciones con el controlador PID autosintonizado para setpoint	
	500 mm.....	154
Tabla 56	Mediciones con el controlador PID autosintonizado para setpoint	
	1000 mm.....	155
Tabla 57	Mediciones con el controlador PID autosintonizado para setpoint	
	1250 mm.....	156
Tabla 58	Mediciones con el controlador PID autosintonizado para setpoint	
	1500 mm.....	157
Tabla 59	Mediciones con el controlador PID autosintonizado para setpoint	
	1750 mm.....	158
Tabla 60	Mediciones con el controlador PID autosintonizado para setpoint	
	2000 mm.....	159
Tabla 61	Mediciones con el controlador PID autosintonizado para setpoint	
	2250 mm.....	160
Tabla 62	Mediciones con el controlador PID autosintonizado para setpoint	
	2500 mm.....	161
Tabla 63	Mediciones con el controlador PID autosintonizado para setpoint	
	3000 mm.....	162

## ÍNDICE DE FIGURAS

Figura 1	Respuesta transitoria de sistemas de control PID .....	31
Figura 2	Tipos de amortiguamiento de un controlador PID .....	32
Figura 3	Control PID de una planta .....	33
Figura 4	Respuesta a un escalón unitario de una planta sin realimentación .....	34
Figura 5	Curva de respuesta en forma de S .....	34
Figura 6	Sistema en lazo cerrado con un controlador proporcional .....	36
Figura 7	Oscilación sostenida con periodo $P_{cr}$ .....	36
Figura 8	PLC Simatic S7 1500 y módulos de expansión.....	39
Figura 9	Lenguaje de programación KOP .....	40
Figura 10	Lenguaje de programación FUB.....	41
Figura 11	Estructura del lenguaje de programación AWL .....	41
Figura 12	Estructura del lenguaje de programación SCL.....	43
Figura 13	Colaboración entre los bloques en un control secuencial.....	44
Figura 14	Diagrama de bloques de PID_Compact.....	46
Figura 15	Tipos de variables en un proceso de control.....	51
Figura 16	Módulo de la planta térmica para el control PID.....	52
Figura 17	Estación Factory IO para control de nivel de líquido en tanque .....	53
Figura 18	Comunicación cliente servidor con OPC UA.....	54
Figura 19	Comunicación OPC UA e intercambio de datos entre dispositivos .....	56
Figura 20	Configuración de nodos y las referencias que las conectan .....	57
Figura 21	Estructura de un Node ID para comunicación OPC UA .....	59
Figura 22	Vista y acceso a datos del servidor OPC UA en UaExpert .....	61
Figura 23	Simatic WinCC Unified en ejecución .....	63
Figura 24	Factory IO: Escena para implementación de procesos.....	64
Figura 25	Ventana de configuración de OPC Client DA/UA.....	65
Figura 26	PLC S7 1500 con módulos de entrada salida digitales y analógicos.....	68
Figura 27	Planta térmica con rango desde temperatura ambiente hasta 100 °C.....	68
Figura 28	Asignación de direcciones para la comunicación del PLC S7 1500 con OPC UA.....	69
Figura 29	Módulo de WinCC Unified V17 .....	70

Figura 30 Diagrama del diseño de investigación experimental antes-después.....	72
Figura 31 Diagrama del servidor del controlador PID autosintonizable implementado en PLC S7 1500 con PID_Compact y envío de datos por OPC UA .....	75
Figura 32 Configuración del “Canal O” del módulo AI 8xU/I/RTD/TC para medición de temperatura en el rango de 0 – 10 V .....	76
Figura 33 Diagrama de normalizado y escalado de la variable “Nivel” para conexión a PID_Compact.....	77
Figura 34 Diagrama de selección de las variables Temperatura y Nivel para ingreso de datos a PID_Compact.....	78
Figura 35 Bloque de instrucción PID_Compact con entradas y salidas previa a la configuración .....	79
Figura 36 Ajuste básico del bloque de instrucción PID_Compact: (a) tipo de regulación “Temperatura” y (b) parámetros de entrada “analógica” y salida en “%” .....	80
Figura 37 Ajustes de los límites y escala del valor real en el bloque de instrucción PID_Compact.....	80
Figura 38 Ajustes de los límites y escala del valor real en el bloque de instrucción PID_Compact.....	81
Figura 39 Ruta para acceso al bloque protegido contra escritura TuneRule de PID_Compact .....	82
Figura 40 Valores que toma TuneRuleTIR para configurar diferentes tipos de sintonización.....	82
Figura 41 Bloque de instrucción PID_Compact con entradas y salidas asignadas a “Tags”.....	83
Figura 42 Tags de entradas y salidas del bloque de funciones “Hmi”.....	84
Figura 43 Configuración del servidor para comunicación OPC UA .....	85
Figura 44 Resultado de la compilación de la estructura del hardware y programa del servidor PID autosintonizable .....	86
Figura 45 Proceso de carga de la configuración del hardware y programa del servidor PID autosintonizable en PLC S7 1500 .....	86

Figura 46 Diagrama general del cliente del controlador PID Autosintonizable implementado en WinCC Unified .....	87
Figura 47 Creación del dispositivo PC-System y selección de SIMATIC WinCC Unified.....	88
Figura 48 Configuración del driver de comunicaciones OPC UA.....	88
Figura 49 Direcciones generadas de las diferentes variables de interface HMI para comunicación OPC UA .....	89
Figura 50 Obtención de nodos e identificador de variables con UaExpert.....	90
Figura 51 Diagrama de flujo de ejecución de imágenes del cliente del controlador PID autosintonizable .....	90
Figura 52 Configuración del rol de los usuarios Administrador y Operador.....	91
Figura 53 Árbol de creación de imágenes en WinCC Unified .....	91
Figura 54 Componentes e íconos de acceso de la imagen de inicio “Tesis” .....	92
Figura 55 Componentes de la imagen “Configuración” .....	93
Figura 56 Configuración del evento para sintonización al pulsar el botón respectivo.....	94
Figura 57 Configuración del evento para sintonización al soltar el botón respectivo.....	94
Figura 58 Componentes de la imagen “Autosintonización” .....	95
Figura 59 Componentes de la imagen “Autosintonización_Temperatura” .....	96
Figura 60 Componentes de la imagen “Autosintonización_Nivel” .....	96
Figura 61 Respuesta sobre amortiguada del controlador PID no sintonizado para la variable Temperatura de la planta térmica.....	98
Figura 62 Respuesta sub amortiguada del controlador PID autosintonizado para la variable Temperatura de la planta térmica .....	100
Figura 63 Respuesta sub amortiguada del controlador PID no sintonizado para la variable Nivel de líquido en tanque .....	102
Figura 64 Respuesta sub amortiguada del controlador PID autosintonizado para la variable Nivel de líquido en tanque .....	104



Figura 65 Representación gráfica de la prueba t de Student unilateral a la derecha para dos muestras dependientes de la variable Temperatura en planta térmica, para un intervalo de confianza del 95% .....	107
Figura 66 Representación gráfica de la prueba t de Student unilateral a la derecha para dos muestras dependientes de la variable Nivel de líquido en tanque, para un intervalo de confianza del 95% .....	111
Figura 67 Diagrama de terminales para medición de tensión proveniente de sensores.....	170
Figura 68 Diagrama de terminales para salida de tensión para activar calefactor en planta térmica.....	171

## RESUMEN

En este trabajo se buscó implementar un controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA, para reducir el tiempo de asentamiento de las variables de manera significativa. Se usó la plataforma de TIA Portal para la implementación del servidor y cliente del controlador PID; la implementación del servidor del controlador PID se hizo en PLC Simatic S7 1500 con PID\_Compact y configurado con la regla de Ziegler-Nichols, con comunicación de datos por protocolo OPC UA con el cliente implementado en WinCC Unified. Para el control de la variable temperatura se empleó la planta térmica constituido por el módulo SO 3536-8T de Lucas Nulle, y para el control de la variable nivel de líquido en tanque se empleó una planta virtual implementado en Factory IO. Para determinar la reducción del tiempo de asentamiento de las variables del proceso térmico y nivel de líquido en tanque; se variaron los factores  $K_p$ ,  $T_i$  y  $T_d$  del controlador PID durante la autosintonización. Para el controlador PID no sintonizado con entradas setpoint de tipo escalón, para temperatura a 50 °C y para nivel a 1500 mm, se establecieron los parámetros:  $K_p= 1.0$ ,  $T_i= 20.00$  s,  $T_d=0.1$  s y  $e_{ss} \leq 2\%$ , obteniéndose; para la variable temperatura un tiempo de asentamiento máximo de 193 s y para la variable nivel un tiempo de asentamiento máximo de 70 s. Luego de autosintonizar el controlador PID para la variable temperatura, con setpoint 50 °C se obtienen los parámetros  $K_p= 6.03$ ,  $T_i= 0.74$  s,  $T_d=0.18$  s para  $e_{ss} \leq 2\%$  siendo la respuesta del tipo sub amortiguado con un tiempo de asentamiento máximo de 54 s; para la variable nivel con setpoint 1500 mm se obtienen los parámetros:  $K_p= 0.80$ ,  $T_i= 2.74$  s,  $T_d=0.68$  s para  $e_{ss} \leq 2\%$  siendo la respuesta del tipo sub amortiguado, con un tiempo de asentamiento máximo de 42 s. En consecuencia, la reducción del tiempo de asentamiento de la variable temperatura fue superior al 60 % y de la variable nivel fue superior al 20%. Por lo que, se concluye que se implementó un controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA que reduce significativamente el tiempo de asentamiento de las variables.

Palabras clave: *Automatización, control PID, sintonización, OPC UA, PLC.*

## ABSTRACT

In this work, we sought to implement a self-tuning PID controller for industrial processes integrated into a S7 1500 PLC with OPC UA communication, to significantly reduce the settling time of the variables. The TIA Portal platform was used for the implementation of the PID controller server and client; The implementation of the PID controller server was done in a Simatic S7 1500 PLC with PID\_Compact and configured with the Ziegler-Nichols rule, with data communication via OPC UA protocol with the client implemented in WinCC Unified. The thermal plant equipped with the Lucas Nulle SO 3536-8T module was used to control the variable temperature, and a virtual plant implemented in Factory IO was used to control the variable level of liquid in the tank. To determine the reduction of the settling time of the variables of the thermal process and liquid level in the tank; the  $K_p$ ,  $T_i$  and  $T_d$  factors of the PID controller were varied during autotuning. For the non-tuned PID controller with step-type setpoint inputs, for temperature at 50 °C and for level at 1500 mm, the parameters are presented:  $K_p = 1.0$ ,  $T_i = 20.00$  s,  $T_d = 0.1$  s and  $e_{ss} \leq 2\%$ , obtaining; for the temperature variable a maximum settling time of 193 s and for the level variable a maximum settling time of 70 s. After autotuning the PID controller for the variable temperature, with a setpoint of 50 °C, the parameters  $K_p = 6.03$ ,  $T_i = 0.74$  s,  $T_d = 0.18$  s are obtained for  $e_{ss} \leq 2\%$ , being the response of the sub-damped type with a settling time 54s maximum; For the level variable with a setpoint of 1500 mm, the following parameters were obtained:  $K_p = 0.80$ ,  $T_i = 2.74$  s,  $T_d = 0.68$  s for  $e_{ss} \leq 2\%$ , the response being of the sub-damped type, with a maximum settling time of 42 s. Consequently, the reduction in the settling time of the temperature variable was greater than 60% and of the level variable was greater than 20%. Therefore, it is concluded that a self-tuning PID controller for industrial processes integrated into a S7 1500 PLC with OPC UA communication was implemented, which significantly reduces the settling time of the variables.

**Keywords:** *Automation, PID control, tuning, OPC UA, PLC.*

# INTRODUCCIÓN

El importante desarrollo que ha tenido la automatización de procesos en los últimos años ha permitido que se desarrollen continuamente nuevas tecnologías y mejores dispositivos. Este desarrollo ha significado que el campo del control y la automatización de procesos debe adaptarse al entorno tecnológico cambiante en el que nos encontramos y empleando técnicas de sintonización de los controladores PID asociado a los PLC con recursos tecnológicos incorporados y comunicaciones eficientes como OPC UA son una buena alternativa para el desarrollo de controladores autosintonizables integrables a procesos industriales por medio de interfaces cliente-servidor.

De esta necesidad, se planteó el objetivo de implementar un controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA, para la reducción significativa del tiempo de asentamiento de las variables; aprovechando los nuevos avances en el campo del control y automatización y las comunicaciones industriales, haciendo posible el desarrollo de nuevos y mejores sistemas adaptables y empleables en la industria.

El contenido del informe está constituido por 04 capítulos:

En el Capítulo I, se aborda el planteamiento del problema; se presenta la fundamentación del problema, formulación del problema, objetivos, ámbito de estudio y la justificación de la investigación.

En el Capítulo II, se expone el marco teórico; presentando los antecedentes de trabajos similares como artículos científicos y trabajos de tesis que guiaron e hicieron posible la ejecución del trabajo. También, se establece las bases teóricas referidas a controladores PID, PLC Simatic S7 1500 y módulos de entrada/salida, procesos industriales, comunicación con protocolo industrial OPC UA y clientes OPC UA, que fueron el fundamento para implementar las diferentes etapas del controlador PID. Así mismo, se establecen las hipótesis y la operacionalización de las variables.

En el Capítulo III, se describe los materiales y métodos; presentando los materiales, equipos y programas de aplicación usados, tipo y nivel de investigación, método de investigación, diseño de investigación, población, muestra, muestreo análisis y procesamiento de datos.

En el Capítulo IV, se expone la discusión de resultados; se explica el proceso de implementación del servidor y del cliente del controlador PID autosintonizable, se determina el tiempo de asentamiento de la variable temperatura en un proceso térmico; así como, el tiempo de asentamiento de la variable nivel de líquido en tanque. Luego se expone la prueba de las hipótesis y la discusión respectiva con los antecedentes.

Finalmente, en el informe se presenta las conclusiones, recomendaciones, referencia bibliográfica y apéndice.

# **CAPÍTULO I**

## **PLANTEAMIENTO DEL PROBLEMA**

### **1.1. FUNDAMENTACIÓN DEL PROBLEMA**

En los últimos años a nivel mundial, la automatización juega un papel fundamental en todas las industrias y tiende a innovarse constantemente, tal como evidencia el estudio “Megatendencias al 2050: grandes retos e implicancias” (CEPLAN, 2021); y que la calidad del producto, la tasa de producción y la satisfacción del cliente se mejoran con la implementación de la automatización.

La situación actual de las industrias es que se enfrentan a problemas de modernizar y automatizar procesos, con su respectiva optimización, para el cual se emplean diversas técnicas (Kamatchi Sundari et al., 2021).

Muchas de las tareas de control de actuadores dinámico requieren realimentación automática y se basan fundamentalmente en el tipo de control proporcional-integral-derivativo (PID). PID se considera como un algoritmo de control computacional simplista. Sin embargo, al igual que todos los problemas de optimización no convexos, ajustar el algoritmo PID para un control de bucle cerrado preciso y estable se convierte en un problema (Somefun et al., 2021). Esto conduce a un dilema, tanto para los usuarios como para los diseñadores, especialmente en las industrias de producción.

Aunque continúan surgiendo varios controladores nuevos, los controladores PID todavía se encuentran en una posición dominante debido a su estructura simple, fácil implementación y buena robustez. En el diseño y la aplicación de controladores PID, uno de los problemas principales es el ajuste de parámetros, seleccionar con precisión y eficacia los mejores parámetros de ajuste del PID es la clave para lograr un controlador PID eficiente operando en procesos industriales (Zhang et al., 2021).

Así mismo, los controladores PID requieren de la autosintonización, bajo diferentes reglas y estrategias, con respuesta de los actuadores que permiten llevar las variables controladas a valores deseados en el menor tiempo posible; mejorando de esta manera los procesos industriales y protegiendo los equipos del desgaste electromecánico que puede ser ocasionado por operaciones defectuosas, traducándose en costos de operación y mantenimiento elevados.

También, es necesario reducir los tiempos de asentamiento de las diferentes variables en los procesos industriales, el cual se logra con una adecuada sintonización de los controladores y para que este proceso sea amigable y seguro, para el operador, se requiere de una interface HMI cliente con comunicación por un protocolo industrial.

Por otro lado, la Arquitectura Unificada OPC (OPC UA) es un protocolo de comunicación independiente del proveedor para aplicaciones de automatización industrial. Se basa en el principio cliente-servidor y permite una comunicación continua desde los sensores y actuadores individuales hasta el sistema ERP o la nube (B&R Industrial Automation, 2020). Este protocolo por la independencia que presenta tiene mecanismos que permite la conexión de manera segura e independiente con otras interfaces, por lo que es considerado el protocolo de comunicación ideal para la implementación de la Industria 4.0 entre otros (Mathias et al., 2020).

Por tanto, en este escenario, fue necesario plantear la factibilidad de la implementación de un controlador PID autosintonizable para procesos industriales integrado a un controlador lógico programable (PLC) con comunicación OPC UA, para lograr la reducción significativa del tiempo de asentamiento de las variables controladas en los procesos industriales.

## **1.2. FORMULACIÓN DEL PROBLEMA**

### **1.2.1. Problema general**

¿Será factible la implementación de un controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA, que reduce significativamente el tiempo de asentamiento de variables controladas?

### **1.2.2. Problemas específicos**

- a. ¿Será factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción significativa del tiempo de asentamiento de la variable temperatura en un proceso industrial térmico para un  $ess \leq 2\%$ ?
- b. ¿Será factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción significativa del tiempo de asentamiento de la variable nivel de líquido en tanque para un  $ess \leq 2\%$ ?

## **1.3. OBJETIVOS**

### **1.3.1. Objetivo general**

Implementar un controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA, que reduce significativamente el tiempo de asentamiento de las variables controladas.

### **1.3.2. Objetivos específicos**

- a. Implementar el servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación



de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción significativa del tiempo de asentamiento de la variable temperatura en un proceso industrial térmico para un  $\text{ess} \leq 2\%$ .

- b. Implementar el servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción significativa del tiempo de asentamiento de la variable nivel de líquido en tanque para un  $\text{ess} \leq 2\%$ .

#### **1.4. ÁMBITO DE ESTUDIO**

El estudio se llevó a cabo en el laboratorio de control y automatización de la Escuela Profesional de Ingeniería Electrónica, para el cual se usó un módulo configurado como servidor en el PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols. El cliente del controlador se implementó en WinCC Unified para PC con comunicación OPC UA, para control de temperatura en planta térmica con conexión directa de datos provenientes del sensor en señal analógica (0 – 10 V) y nivel de líquido en tanque implementado en Factory IO con interconexión de datos por comunicación OPC UA.

#### **1.5. JUSTIFICACIÓN**

Los controladores PID están ampliamente distribuidos en muchas aplicaciones industriales por sus características especiales como la estabilidad, simplicidad escaso error estacionario (Singhal et al., 2012); sin embargo, presenta aspectos críticos relacionados con la optimización de los diferentes parámetros como la constante de proporcionalidad  $K_p$ , constante de integración  $K_i$  y constante de derivación  $K_d$  (Kumar Mishra, A., Kumar Tiwari, V., Kumar, R., Verma, 2013), siendo este aspecto el que da origen a esta investigación.

Por otro lado, se vio la necesidad de usar el protocolo de comunicación de arquitectura unificada para realizar ajustes online de los parámetros de los

componentes del controlador PID para su sintonía de manera interactiva y en tiempo real con los procesos industriales.

Estos aspectos justifican el trabajo, además de incluir equipos tecnológicos emergentes a nivel de hardware como el PLC S7 1500 con PID\_Compact configurable con la regla de Ziegler – Nichols, y soporte para comunicaciones OPC UA. a nivel de software se empleó la plataforma de desarrollo TIA PORTAL de Siemens; el cliente se implementó en WinCC Unified, que interacciona con una planta virtual de procesos industriales implementados en Factory IO y una planta térmica del laboratorio de la Escuela Profesional de Ingeniería Electrónica. Con lo que se obtuvo un controlador PID autosintonizable confiable y adaptable a diversos procesos industriales.

## CAPÍTULO II

### MARCO TEÓRICO

#### 2.1. ANTECEDENTES

López y otros en el año 2020 (López-Rivera et al., 2020), presentan en el “*IX International Congress of Mechatronics Engineering and Automation*” un análisis de una estrategia para el diseño de un controlador PID multivariable para un sistema multivariable no lineal con dos entradas y dos salidas. El sistema multivariado no lineal seleccionado es un prototipo de helicóptero a pequeña escala de dos grados de libertad. El rendimiento del controlador se evalúa en un modelo simulado y en el sistema real para varios casos de cambios de referencia y perturbaciones, donde se muestra la ventaja de usar el control PID multivariable acoplado en comparación con el sistema desacoplado. También se hace una comparación del controlador PID multivariable, con un controlador para la retroalimentación de las variables de estado con el observador en el sistema simulado.

Radu y otros en el año 2016 (Radu et al., 2016) en el trabajo “*Autotuned PID for accurate temperature control — A hot approach*” desarrollan un sistema de control de temperatura preciso, basado en un algoritmo optimizado proporcional - integrativo - derivado (PID). La solución propuesta también incluye un módulo de autoajuste que detecta cambios de carga y ajusta los parámetros del algoritmo para un control de temperatura óptimo. La arquitectura PID autotuned se implementa como un sistema de control para el elemento calefactor de una estación de soldadura. Los resultados experimentales muestran un control de temperatura adecuado, eliminando tanto el exceso de temperatura como la ondulación. Además, el autotuned logra un rendimiento óptimo independientemente del elemento calefactor utilizado para 30W y 48W.

Merzlikina y otros en el año 2021 (Merzlikina et al., 2021) en el trabajo *“Automatic Control System with an Autotuning Module and a Predictive PID-Algorithm for Thermal Processes”* consideran un sistema de control de bucle único que consta de un proceso, un algoritmo PID con un filtro de segundo orden, un módulo de autoajuste y un módulo de predicción. El módulo de predicción está en la retroalimentación; en él se utiliza el algoritmo de predicción lineal. El proceso de control es un proceso térmico típico con un retraso de tiempo. Se analiza el rendimiento del sistema. Se analizan las recomendaciones sobre la elección del tiempo de predicción dadas en la literatura y se dan nuevas recomendaciones para el sistema con el algoritmo PID mencionado anteriormente; Se analiza la dependencia entre el tiempo de predicción y el tiempo de retardo de la planta de control. Se muestran las ventajas del sistema de control con el módulo de predicción y el tiempo de predicción óptimo. El modelo de computadora del sistema se desarrolla en Matlab y Simulink.

Kuzishchin y otros en el año 2018 (Kuzishchin et al., 2018) en el trabajo *“Study of Maximal Rapid Response Algorithm Combined with PID-Algorithm and Autotuning Module”* consideran un sistema de control híbrido, el sistema constó de un algoritmo de respuesta rápida máxima (MRA), un módulo PID y un módulo de autoajuste (ATC). Cuando el punto de ajuste cambia, el módulo MRA funciona, cuando finaliza el proceso transitorio, el módulo PID comienza a funcionar para estabilizar el valor del proceso bajo la influencia de diversas perturbaciones. Se analiza la eficiencia del sistema para plantas con el retardo de tiempo, también se analizan algunas formas de eliminación de auto-oscilaciones. Se muestran las ventajas del sistema MRA sin la acción de control inversa, se dan algunas recomendaciones para el funcionamiento del módulo de predicción.

Figuroa y otros en el año 2017 (Figuroa et al., 2017), desarrollaron el trabajo *“Adaptive PID controller with auto-tuning applied to the agricultural food industry”* para el ahorro de energía al generar vapor para la conservación de

productos agroalimentarios e hidrobiológicos y de los altos costes económicos que produce el uso de estos sistemas. Por lo que diseñaron un sistema de control basado en el algoritmo de un autoajuste PID adaptativo que cuenta con un proceso de aprendizaje, el cual, para demostrar un mejor desempeño de los sistemas de enfriamiento por compresión de vapor y, además, el controlador debe evitar compresor. empieza, durante el cual hay mucho consumo de energía.

Beschi y otros en el año 2015 (Beschi et al., 2015), desarrollan el trabajo “*An Event-based PI controller autotuning technique for integral processes*”, donde proponen una metodología de autoajuste para un controlador PI simétrico send-on-delta para procesos integrales. En particular, una técnica ya propuesta para los procesos autorreguladores se amplía para hacer frente a los procesos no autorreguladores. Se basa en la estimación de los parámetros de una función de transferencia de integrador más tiempo muerto mediante un único experimento de retroalimentación de relé en el que también se utiliza un controlador PI ajustado de forma aproximada en el sistema de retroalimentación. Luego, se pueden seleccionar nuevos parámetros del controlador en función del modelo de proceso identificado. Los resultados de la simulación demuestran según los autores la eficacia de la metodología.

Kolaj y otros en el año 2016 (Kolaj et al., 2016) en el trabajo “*PLC-PIDTuner: Application for PID tuning with SIMATIC S7 PLC controllers*” presentan la aplicación PLC-PID Tuner desarrollada para el ajuste del controlador PID en PLCs S7 Simatic. Permitiendo la identificación de la planta, el cálculo de los parámetros del controlador y la verificación del sistema de control utilizando diferentes índices de calidad. Proporciona recopilación de datos del proceso real y simulación por computadora del sistema de control diseñado. Esta solución permite personalizar las propiedades de los sistemas sin realizar costosos experimentos. La aplicación propuesta permite al usuario desarrollar fácilmente diferentes algoritmos para cada etapa del diseño del sistema de control PID. Finalmente, se presenta un ejemplo del uso de la plataforma PLC-

PID Tuner desarrollada en un proceso simulado descrito por un sistema de tercer orden con retraso.

Por otro lado, Haskamp y otros en el año 2018 (Haskamp et al., 2018) desarrollan el trabajo *“Implementing an OPC UA interface for legacy PLC-based automation systems using the Azure cloud: An ICPS-architecture with a retrofitted RFID system”* aplicado a integrar con nuevos entornos industriales que requieren el cumplimiento de Industria 4.0 con interfaces OPC UA, adaptables a tecnologías comunes recomendadas para migrar sistemas industriales heredados a sistemas compatibles con Industrie 4.0.

## **2.2. BASES TEÓRICAS**

### **2.2.1. Controladores PID**

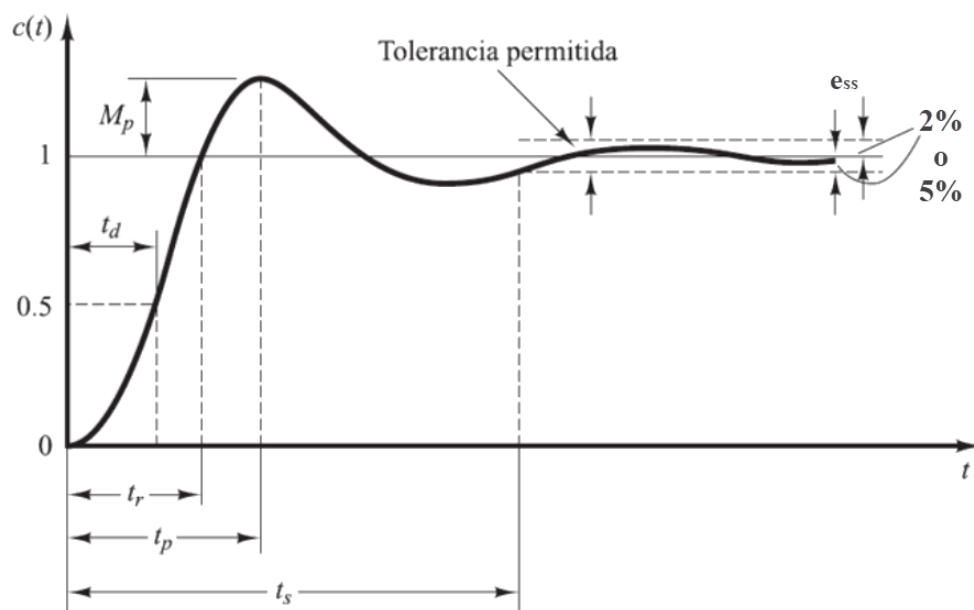
Los controladores PID han sobrevivido a muchos cambios en la tecnología. Comienza con control neumático, mediante control digital directo al DCS. Hoy en día, el controlador PID es muy diferente al de hace 50 años. Normalmente la lógica, función en bloque, selector y secuencia se combinan con el controlador PID; es así que, muchas sofisticadas estrategias de control regulatorio, control multivariable, estrategias de puesta en marcha y parada pueden diseñarse en torno al control PID clásico (Natale et al., 2017). Esto proporciona la base para una buena regulación, un funcionamiento seguro, transitorio suave y una rápida puesta en marcha y parar procesos. En este escenario los controladores lógicos programables proporciona características especiales para la sintonización, programación de ganancias y cambio de modelo y reglas a los controladores PID (Yu, 2007).

La respuesta transitoria de un sistema de control digital puede caracterizarse no sólo por el factor de amortiguamiento relativo y la frecuencia natural amortiguada, sino también por el tiempo de levantamiento, los sobrepasos máximos, el tiempo de asentamiento y así sucesivamente, en respuesta a una entrada escalón (Ogata, 1996). En la especificación de distintas características

de la respuesta transitoria, es necesario especificar los siguientes: Tiempo de retardo  $t_d$ ; Tiempo de levantamiento  $t_r$ ; Tiempo pico  $t_p$ ; Sobrepasso máximo  $M_p$ ; Tiempo de asentamiento  $t_s$ ; error de estado estable  $e_{ss}$ .

**Figura 1**

*Respuesta transitoria de sistemas de control PID*



*Nota.* Figura obtenida de sistemas de control en tiempo discreto (Ogata, 1996)

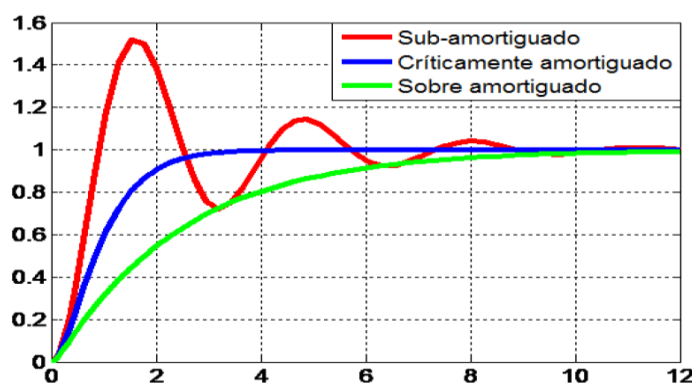
1. **Tiempo de retardo  $t_d$** , el tiempo de retardo es el tiempo requerido para que la respuesta llegue a la mitad del valor final la primera vez.
2. **Tiempo de levantamiento  $t_r$** , el tiempo de levantamiento es el tiempo que requiere la respuesta para pasar de 10% hasta 90%, de 5% a 95% o de 0% a 100% de su valor final. según la situación. Para sistemas de segundo orden subamortiguados, por lo regular se utiliza el tiempo de levantamiento de 0% a 100%. Para sistemas sobreamortiguados y sistemas con atraso de transporte, comúnmente se utiliza el tiempo de levantamiento de 10% a 90%.
3. **Tiempo pico  $t_p$** , el tiempo pico es el tiempo requerido para que la respuesta llegue a la primera cresta del sobrepasso.

4. **Sobrepaso máximo  $M_P$** , el sobrepaso máximo es el valor máximo de la curva de respuesta medido a partir de la unidad. Si el valor final en estado permanente de la respuesta difiere de la unidad, entonces es común utilizar el sobrepaso porcentual máximo.
5. **Tiempo de asentamiento  $t_s$** , el tiempo de asentamiento es el tiempo requerido para que una curva de respuesta llegue y se quede dentro de un rango alrededor del valor final de un tamaño especificado, en función de un porcentaje absoluto de valor final, por lo general 2%. El tiempo de asentamiento está relacionado con la constante de tiempo de mayor valor en el sistema de control.
6. **Error de estado estable  $e_{ss}$** , el error de estado estable es determinado por la diferencia entre el setpoint y el valor de la variable de proceso, es el error límite permitido para el control de un proceso; se considera como 2% o 5%.

La respuesta transitoria de un sistema de control PID puede ser sub amortiguado, críticamente amortiguado y sobre amortiguado (ver Figura 2).

**Figura 2**

*Tipos de amortiguamiento de un controlador PID*



*Nota.* Figura obtenida de Ortiz y otros (2014)

### 2.2.1.1. Control PID de una planta

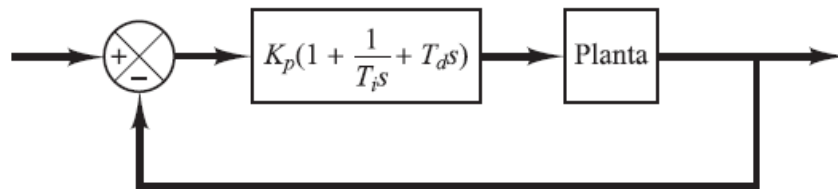
Si se puede obtener un modelo matemático de la planta, es posible aplicar diversas técnicas de diseño con el fin de determinar los parámetros del



controlador que cumpla las especificaciones del transitorio y del estado estacionario del sistema en lazo cerrado. Sin embargo, si la planta es tan complicada que no es fácil obtener su modelo matemático, tampoco es posible un método analítico para el diseño de un controlador PID. En este caso, se debe recurrir a procedimientos experimentales para la sintonía de los controladores PID” (Ogata, 2010, p. 568). En la Figura 3 se presenta el diagrama de control PID de una planta con los factores  $K_p$ ,  $T_i$  y  $T_d$ .

**Figura 3**

Control PID de una planta



*Nota.* Figura obtenida de ingeniería de control moderno (Ogata, 2010)

Donde:  $K_p$  = ganancia proporcional,  $T_i$  = constante de tiempo integral y  $T_d$  = constante de tiempo derivativa.

#### **2.2.1.2. Métodos de sintonización PID**

Existen diversos métodos de ajuste para la sintonización PID, entre los más usados se tienen:

- Métodos de Ziegler-Nichols (Z-N)
- Método Tyreus-Lubeyn (T-L)
- Método de Chien-hrones-Reswick (C-H-R)

En este trabajo de tesis se empleó el método de Ziegler-Nichols basados en los pulsos críticos.

##### **a. Métodos de Ziegler-Nichols para sintonizar controladores PID**

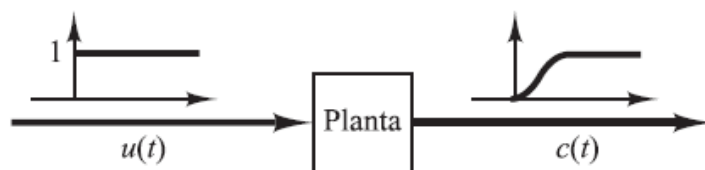
Ziegler y Nichols propusieron reglas para determinar los valores de la ganancia proporcional  $K_p$ , del tiempo integral  $T_i$  y del tiempo derivativo  $T_d$ , basándose

en las características de respuesta transitoria de una planta dada (Ogata, 2010). Tal determinación de los parámetros de los controladores PID o sintonía de controladores PID se puede realizar mediante experimentos sobre la planta. Son dos los métodos denominados reglas de sintonía de Ziegler-Nichols.

**Primer método.** En el primer método, la respuesta de la planta a una entrada escalón unitario se obtiene de manera experimental (ver Figura 4). Si la planta no contiene integradores ni polos dominantes complejos conjugados, la curva de respuesta escalón unitario puede tener forma de S, como se observa en la Figura 5. Este método se puede aplicar si la respuesta muestra una curva con forma de S. Tales curvas de respuesta escalón se pueden generar experimentalmente o a partir de una simulación dinámica de la planta.

**Figura 4**

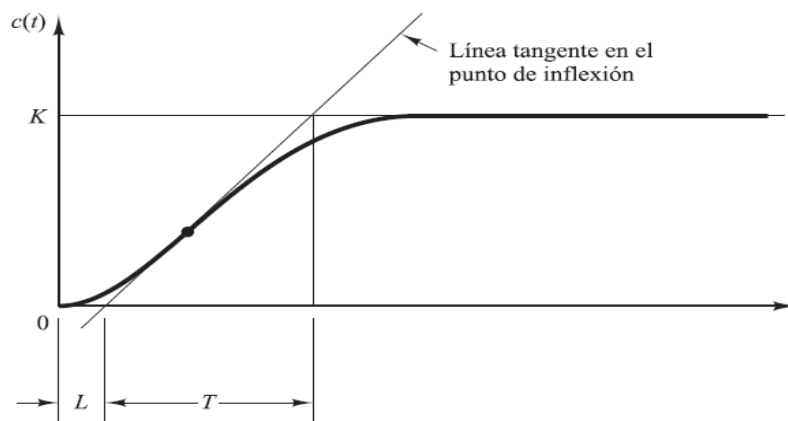
*Respuesta a un escalón unitario de una planta sin realimentación*



*Nota.* Figura obtenida de ingeniería de control moderno (Ogata, 2010)

**Figura 5**

Curva de respuesta en forma de S



*Nota.* Figura obtenida de ingeniería de control moderno (Ogata, 2010)

Para un escalón unitario la función de transferencia  $C(s)/U(s)$  se aproxima mediante un sistema de primer orden con un retardo del modo presentado en la ecuación 1.

$$\frac{C(s)}{U(s)} = \frac{K e^{-Ls}}{Ts + 1} \quad (1)$$

Ziegler y Nichols sugirieron establecer los valores de  $K_p$ ,  $T_i$  y  $T_d$  de acuerdo con la fórmula de la Tabla 1, donde se observa que el controlador PID sintonizado mediante el primer método de las reglas de Ziegler-Nichols produce la ecuación 2.

$$G_s(s) = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right) \quad (2)$$

$$G_s(s) = 1.2 \frac{T}{L} \left( 1 + \frac{1}{2Ls} + 0.5Ls \right) \quad (3)$$

$$G_s(s) = 0.6T \frac{(s + \frac{1}{L})^2}{s} \quad (4)$$

**Tabla 1**

*Regla de sintonía de Ziegler-Nichols basada en la respuesta escalón de la planta*

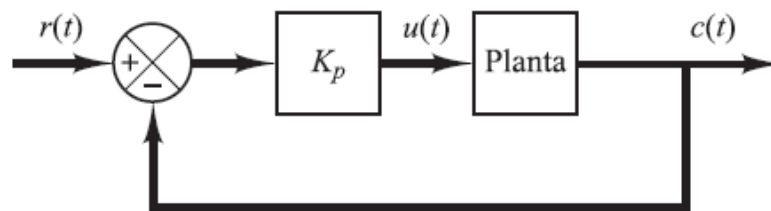
Tipo de controlador	$K_p$	$T_i$	$T_d$
P	$\frac{T}{L}$	$\infty$	0
PI	$0.9 \frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2 \frac{T}{L}$	$2L$	$0.5L$

*Nota.* tabla obtenida de ingeniería de control moderno (Ogata, 2010)

**Segundo método.** En el segundo método, en la ecuación 2, primero se fija  $T_i = \infty$  y  $T_d = 0$ . Usando sólo la acción de control proporcional (véase la Figura 6), se incrementa  $K_p$  desde 0 hasta un valor crítico  $K_{cr}$ , en donde la salida presente oscilaciones sostenidas. Así, la ganancia crítica  $K_{cr}$  y el periodo  $P_{cr}$  (medido en seg) correspondiente se determinan experimentalmente (ver Figura 7)

**Figura 6**

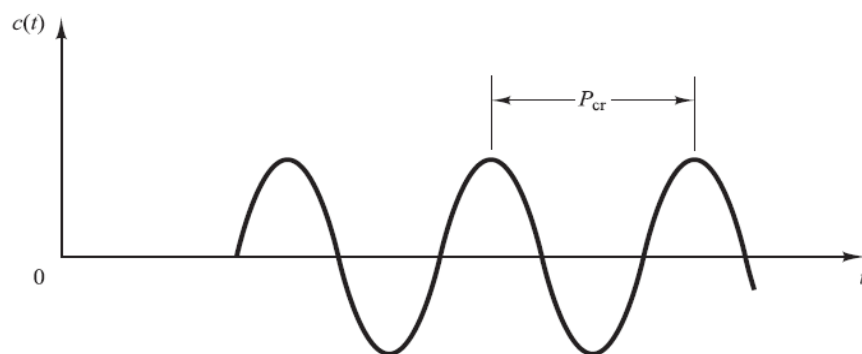
*Sistema en lazo cerrado con un controlador proporcional*



*Nota.* Figura obtenida de ingeniería de control moderno (Ogata, 2010)

**Figura 7**

*Oscilación sostenida con periodo  $P_{cr}$*



*Nota.* Figura obtenida de ingeniería de control moderno (Ogata, 2010)

Ziegler y Nichols sugirieron que se establecieran los valores de los parámetros  $K_p$ ,  $T_i$  y  $T_d$  de acuerdo con la ecuación 5.

$$G_c(s) = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right) \quad (5)$$

$$G_c(s) = 0.6K_{cr} \left( 1 + \frac{1}{0.5P_{cr}s} + 0.125P_{cr}s \right) \quad (6)$$

$$G_c(s) = 0.075K_{cr}P_{cr} \frac{\left(s + \frac{4}{P_{cr}}\right)^2}{s} \quad (7)$$

Obteniéndose los valores que se resumen en la Tabla 2.

**Tabla 2**

*Regla de sintonía de Ziegler-Nichols basada en la ganancia crítica  $K_{cr}$  y periodo crítico  $P_{cr}$*

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

*Nota.* Tabla obtenida de ingeniería de control moderno (Ogata, 2010)

## b. Método Tyreus-Lubeyn

Este método también se denomina método de ajuste en línea. Desarrollaron este método de sintonización en modo de bucle cerrado. Este método de ajuste de circuito cerrado supera las deficiencias del conocido método de ciclo continuo de Ziegler-Nichols y proporciona un rendimiento y una robustez consistentemente mejores para una amplia clase de procesos. El método de diseño del controlador PI/PID se ha discutido ampliamente en la literatura y muestra que la mayor parte del método de ajuste se basa en el procedimiento de dos pasos, que es muy fácil de calcular los parámetros (Ogata, 2010).

### c. Método Chien-Hrones-Reswick:

El método de ajuste automático de Chien-Hrones-Reswick se centra en la respuesta del punto de ajuste y la respuesta a perturbaciones. Este método proporciona fórmulas para un 0% y un 20% de sobreimpulso.

**Tabla 3**

*Parámetros de ajuste del controlador*

Controlador	0% sobre impulso			20% sobre impulso		
	K	T <sub>i</sub>	T <sub>d</sub>	K	T <sub>i</sub>	T <sub>d</sub>
P	$0.3 T_p/\tau$	-	-	$0.7 T_p/\tau$	-	-
PI	$0.6 T_p/\tau$	$4 \tau$	-	$0.7 T_p/\tau$	$2.3 \tau$	-
PID	$0.95 T_p/\tau$	$2,4 \tau$	$0.42 \tau$	$1.2 T_p/\tau$	$2 \tau$	$0.42 \tau$

*Nota.* Tabla obtenida de ingeniería de control moderno (Ogata, 2010)

De la Tabla 3;  $T_p$  es la constante de tiempo de sobre impulso, con el que se obtiene los valores proporcionales, integrales y derivados de Chien-Hrones-Reswick.

### 2.2.2. PLC Simatic S7 1500

SIMATIC S7-1500 es el sistema de automatización modular para rangos de rendimiento medio y superior. Las diferentes versiones de los controladores permiten que el rendimiento sea emparejado con la aplicación respectiva. Dependiendo de los requisitos, el PLC puede ampliarse con módulos de entrada/salida para digital y señales analógicas y módulos de tecnología y comunicación (Berger, 2017). En la Figura 8, se muestra al PLC Simatic S7 1500 y módulos analógicos y digitales de expansión, obtenidos de la plataforma del TIA Portal.

**Figura 8**

PLC Simatic S7 1500 y módulos de expansión



*Nota.* Figura obtenida de la plataforma de TIA Portal V17.

### 2.2.2.1. Características

Las CPU para S7-1500 están disponibles en varias versiones para diferentes aplicaciones. Común a todas las CPU es el alcance de las funciones de control, como las operaciones lógicas binarias, aritmética de punto fijo y de punto flotante, operandos y tipos de etiquetas, así como datos tipos. Las versiones difieren en términos de diseño, expansión de memoria, estructura de cantidad de operandos y velocidad de procesamiento. Es posible conectarse a Industrial Ethernet utilizando la interfaz PN. Cada CPU puede ser tanto un controlador IO como un "inteligente" Dispositivo IO en PROFINET IO. Una CPU con interfaz DP es el maestro DP en PROFIBUS DP. Todas las versiones de CPU están configuradas y programadas con el Software de ingeniería STEP 7 Professional en el TIA Portal.

Todas las versiones de CPU tienen acceso a datos de servidor OPC UA con estándar I4.0/IoT interfaz a partir de la versión de firmware 2.0, que permite autenticados, firmados y comunicación cifrada. Por lo tanto, los datos de

usuario liberados en la estación S7-1500 se pueden acceder desde cualquier dispositivo externo.

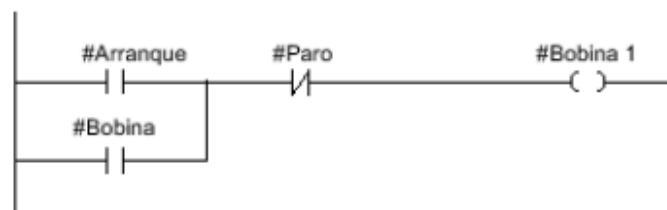
#### 2.2.2.2. *Lenguajes de programación*

El programa de usuario consta de bloques que se guardan en el árbol del proyecto bajo un Estación PLC en la carpeta Bloques de programa. Los bloques de código contienen el código del programa y los bloques de datos contienen los datos de control. Al programar, se crea inicialmente un bloque y posteriormente se llena con datos o un programa. Lógica de escalera (KOP), función diagrama de bloques (FUB), lenguaje de control estructurado (SCL), lista de instrucciones (AWL) y control secuencial (GRAPH) están disponibles como lenguajes para programar el control función. Puede definir el lenguaje de programación individualmente para cada cuadra.

- **Programación KOP (Esquema de contactos):** es un lenguaje de programación gráfico. El programa se mapea en uno o varios segmentos. Un segmento contiene en el margen izquierdo una barra de alimentación de la que parten los circuitos. Las consultas de las señales binarias se disponen en los circuitos en forma de contactos. Si los elementos se disponen en serie en un circuito, se crea una conexión en serie. Si se disponen en ramas paralelas, se crea una conexión en paralelo. Las funciones complejas se representan mediante cuadros (ver Figura 9).

**Figura 9**

*Lenguaje de programación KOP*



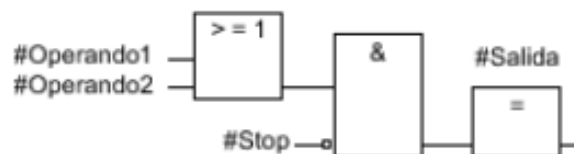
*Nota.* Figura obtenida de ayuda de la plataforma de TIA Portal V17.



- **Lenguaje de programación FUP (Diagrama de funciones):** es un lenguaje de programación gráfico. El programa se mapea en uno o varios segmentos. Un segmento contiene uno o varios circuitos lógicos. Las consultas de las señales binarias se combinan lógicamente mediante cuadros. Para representar la lógica se utilizan los símbolos lógicos gráficos del álgebra booleana (ver Figura 10).

**Figura 10**

*Lenguaje de programación FUB*



*Nota.* Figura obtenida de ayuda de la plataforma de TIA Portal V17.

- **Lenguaje de programación AWL (lista de instrucciones):** es un lenguaje de programación basado en un texto con el que se podrán programar bloques de código. El programa AWL se divide en segmentos. Cada segmento puede tener una o varias líneas. La numeración de las líneas comienza en cada segmento con 1 y continuará de manera ascendente con cada nueva línea. En las líneas del segmento se programarán las instrucciones individuales de AWL, pudiéndose indicar sólo una instrucción AWL por línea. Cada instrucción representa una prescripción de trabajo para la CPU. La CPU ejecuta las instrucciones de arriba abajo. La Figura 11 muestra la programación de AWL de un segmento.

**Figura 11**

*Estructura del lenguaje de programación AWL*

AWL	Explicación
A "Tag_Input_1"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual
A "Tag_Input_2"	// Consultar si el estado lógico del operando es "1" y combinar el resultado lógicamente con Y con el RLO actual
S "Tag_Output"	// Poner el operando a "1" si el RLO es "1"

*Nota.* Figura obtenida de ayuda de la plataforma de TIA Portal V17.

- **Lenguaje de programación SCL (Structured Control Language):** es un lenguaje de programación de alto nivel que se orienta a PASCAL. Este lenguaje se basa en la norma DIN EN-61131-3 (internacional: IEC 1131-3). Esta norma estandariza los lenguajes de programación para autómatas programables. El lenguaje de programación SCL cumple el PLCOpen Basis Level del lenguaje ST (texto estructurado) definido en esta norma.

### **Elementos del lenguaje**

Además de incluir elementos típicos del PLC, como entradas, salidas, temporizadores o marcas, SCL incluye elementos de lenguajes de programación de alto nivel:

- Expresiones
- Asignaciones de valor
- Operadores

### **Control del programa**

SCL ofrece instrucciones prácticas para el control del programa que permiten realizar, por ejemplo, ramas, bucles o saltos del programa.

### **Campo de aplicación**


Por estos motivos, SCL es especialmente adecuado para los siguientes campos de aplicación:

- Gestión de datos
- Optimización de procesos
- Tareas matemáticas/estadísticas

En la Figura 12 se presenta el programa en SCL, empleado para el control de un proceso con motores.

## Figura 12

### *Estructura del lenguaje de programación SCL*

SCL 

```
IF "Marcha_izquierda_S1" OR "Marcha_derecha_S3" THEN
"MOTOR_ON" := 1;
"MOTOR_OFF" := 0;
END_IF;

IF "Paro_izquierda_S2" OR "Paro_derecha_S4" THEN
"MOTOR_ON" := 0;
"MOTOR_OFF" := 1;
END_IF;
```

*Nota.* Figura obtenida de ayuda de la plataforma de TIA Portal V17.

- **GRAPH:** es un lenguaje de programación gráfico para crear controles secuenciales. Permite programar controles secuenciales de manera clara y rápida utilizando cadenas secuenciales. El proceso se descompone en etapas individuales con un alcance funcional delimitable y se organiza en las cadenas secuenciales. En las etapas individuales se determinan las acciones que se deben ejecutar. El paso entre las etapas lo forman las transiciones. Estas contienen condiciones para pasar a la etapa siguiente.
  - **Bloques del control secuencial:**

Un control secuencial controla el proceso en un orden predeterminado y en función de ciertas condiciones. La complejidad del control secuencial depende de la tarea de automatización. Un control secuencial está formado siempre por tres bloques como mínimo:
  - **Bloque de función GRAPH**

En el bloque de función GRAPH se describen las diferentes etapas y transiciones del control secuencial dentro de una o varias cadenas secuenciales.
  - **Bloque de datos de instancia**

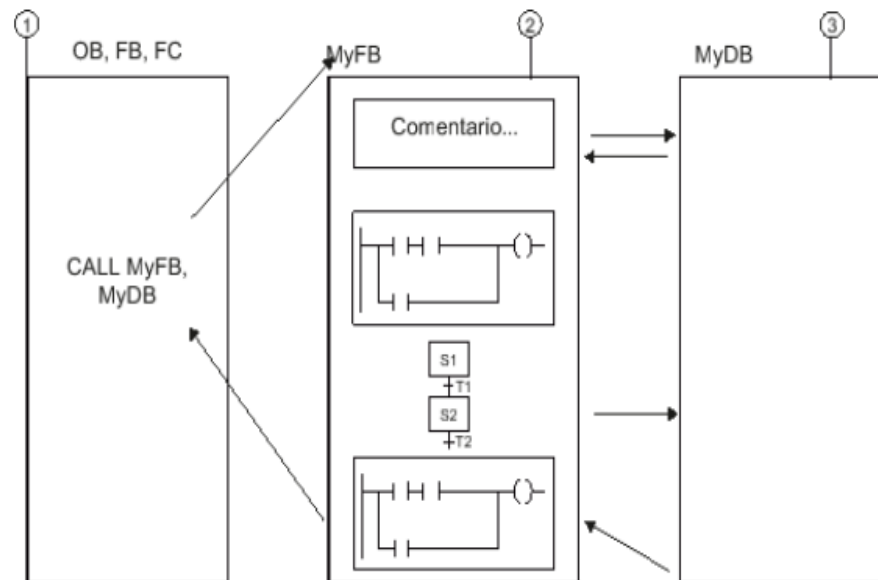
El bloque de datos de instancia contiene los datos y parámetros del control secuencial. Está asignado al bloque de función GRAPH y puede ser generado por el sistema automáticamente.

- **Bloque lógico que efectúa la llamada**

Para que el bloque de función GRAPH pueda ejecutarse dentro del ciclo, debe ser llamado por un bloque lógico de nivel superior. Dicho bloque puede ser un bloque de organización (OB), una función (FC) u otro bloque de función (FB). El bloque de función GRAPH se llama siempre como instancia individual. La Figura 13 muestra la colaboración entre los bloques en un control secuencial.

**Figura 13**

*Colaboración entre los bloques en un control secuencial*



① Bloque que efectúa la llamada, ② Bloque de función GRAPH y ③ Bloque de datos de instancia

*Nota.* Bloques obtenidos de ayuda de la plataforma de TIA Portal V17.

### 2.2.2.3. *PID\_Compact*

La instrucción PID\_Compact ofrece un regulador PID con optimización integrada para actuadores de acción proporcional. Por la practicidad de su aplicación, en la tesis se empleó esta instrucción.

Se dispone de los modos de operación siguientes: Inactivo, Optimización inicial, Optimización fina, Modo automático y Modo manual.

**Algoritmo PID:** PID\_Compact es un regulador PID con Anti-Windup y ponderación de las acciones P y D. El algoritmo PID funciona de acuerdo con la ecuación 8.

$$y = K_p \left[ (b \cdot w - x) + \frac{1}{T_i s} (w - x) + \frac{T_d s}{a \cdot T_d s + 1} (c \cdot w - x) \right] \quad (8)$$

La Tabla 4 muestra los parámetros del controlador PID\_Compact.

**Tabla 4**

*Parámetros del controlador PID\_Compact*

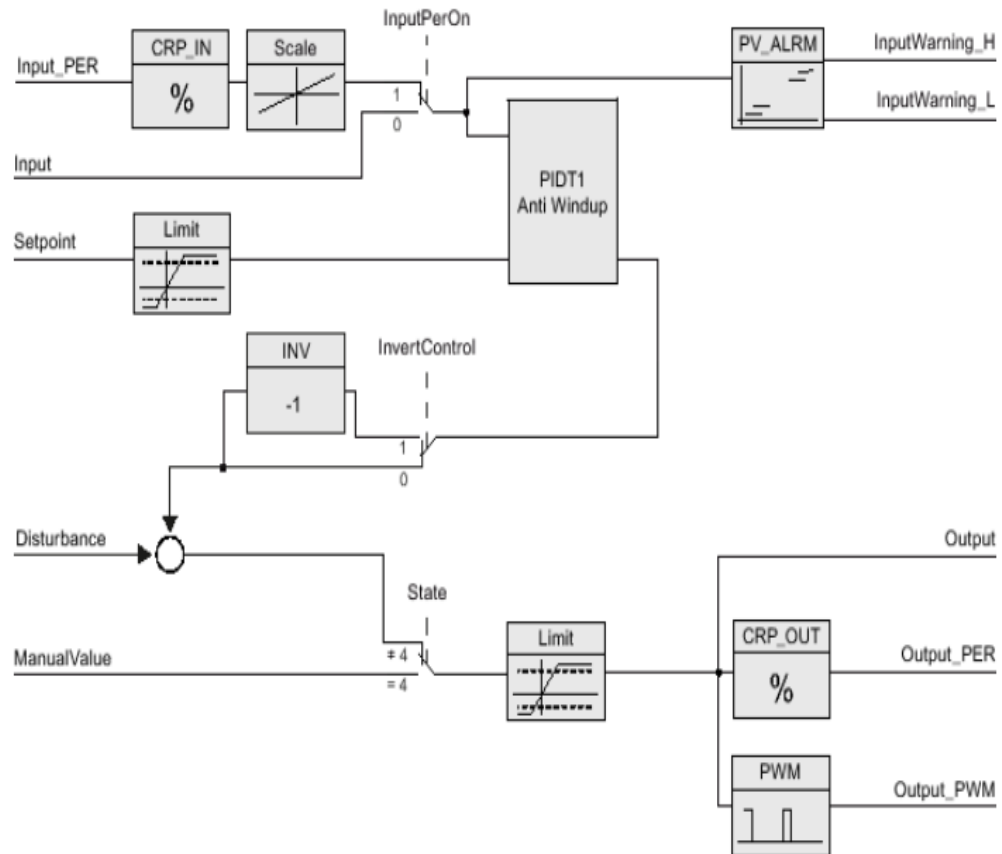
Símbolo	Descripción
y	Valor de salida del algoritmo PID
K <sub>p</sub>	Ganancia proporcional
s	Operador laplaciano
b	Ponderación de la acción P
w	Consigna
x	Valor real
T <sub>i</sub>	Tiempo de integración
T <sub>d</sub>	Tiempo derivativo
a	Coeficiente para el retardo de la acción derivada (retardo de la acción derivada T <sub>1</sub> = a × T <sub>d</sub> )
c	Ponderación de la acción “d”

*Nota.* Tabla adaptada de ayuda para módulos PID Compact de la plataforma de TIA Portal V17.

El diagrama de bloques del controlador PID\_Compact se visualiza en la Figura 14.

**Figura 14**

*Diagrama de bloques de PID\_Compact*



*Nota.* Diagrama de bloques obtenidos de ayuda para PID Compact de la plataforma de TIA Portal V17.

#### **2.2.2.4. Parámetros de entrada de PID\_Compact**

Los parámetros de entrada al módulo del PID\_Compact V2 para PLC Simatic S7 1500 son: Setpoint, Input, Input\_PER, Disturbance, ManualEnable, ManualValue, ErrorAck, Reset y ModeActivate, estos parámetros se detallaron en la Tabla 4. La señal proveniente de la planta donde se lleva a cabo el proceso ingresa por Input con su respectiva unidad de medida si fue previamente procesado, en caso contrario la señal analógica de los sensores ingresa por Input\_PER; esta señal es normalizado y escalado de acuerdo a la configuración que se da al sistema de control. Otro de los parámetros importantes es el Setpoint que representa la consigna del controlador sobre la que se realizaran

los ajusten en los actuadores ubicados en planta. Así mismo, *Disturbance* es un parámetro sobre la que se introduce las perturbaciones generadas por factores externos al proceso de control. Estos parámetros se presentan en la Tabla 5.

**Tabla 5**

*Parámetros de entrada del controlador PID\_Compact*

Parámetro	Tipo de datos	Ajuste predeterminado	Descripción
Setpoint	REAL	0.0	Consigna del regulador PID en modo automático
Input	REAL	0.0	Una variable del programa de usuario se utiliza como origen del valor real.
Input_PER	INT	0	Una entrada analógica se utiliza como origen del valor real.
Disturbance	REAL	0.0	Magnitud perturbadora o valor de control anticipativo
ManualEnable	BOOL	FALSE	El flanco FALSE-TRUE; activa el modo de operación "Modo manual", State = 4. El flanco TRUE-FALSE; activa el modo de operación especificado en Mode.
ManualValue	REAL	0.0	Valor manual. Este valor se utiliza en modo manual como valor de salida. Se admiten valores entre Config.OutputLowerLimit y Config.OutputUpperLimit
ErrorAck	BOOL	FALSE	Flanco FALSE - TRUE
Reset	BOOL	FALSE	Realiza un rearranque completo del regulador.
ModeActivate	BOOL	FALSE	Flanco FALSE - TRUE, PID_Compact cambia al modo guardado en Mode.

*Nota.* Tabla adaptada de ayuda para módulos PID Compact de la plataforma de TIA Portal V17.

### 2.2.2.5. Parámetros de salida de PID\_Compact

Los parámetros de salida del módulo PID\_Compact que interactúan con la planta son: Output en el rango de 0 a 100%, Output\_PER en el rango de 0 a 27648 y Output\_PWM que es señal modulada, se detallan en la Tabla 6.

**Tabla 6**

Parámetros de salida del controlador PID\_Compact

Parámetros	Tipo de datos	Ajuste predeterminado	Descripción
ScaledInput	REAL	0.0	Valor real escalado
Las salidas "Output", "Output_PER" y "Output_PWM" pueden utilizarse paralelamente.			
Output	REAL	0.0	Valor de salida en REAL
Output_PER	INT	0	Valor de salida analógico
Output_PWM	BOOL	FALSE	Valor de salida modulado por ancho de impulso
SetpointLimit_H	BOOL	FALSE	Si SetpointLimit_H = TRUE, significa que se ha alcanzado el límite superior de la consigna.
SetpointLimit_L	BOOL	FALSE	Si SetpointLimit_L = TRUE, significa que se ha alcanzado el límite inferior de la consigna.
InputWarning_H	BOOL	FALSE	InputWarning_H = TRUE, significa que se ha alcanzado o rebasado el límite superior de advertencia del valor real.
InputWarning_L	BOOL	FALSE	InputWarning_L = TRUE, significa que se ha alcanzado o rebasado por defecto el límite inferior de advertencia del valor real.
State	INT	0	El parámetro State indica el modo de operación actual del regulador PID. State = 0: Inactivo; State = 1: optimización inicial; State = 2: Optimización fina; State = 3: Modo automático; State = 4: Modo manual; State = 5: Valor de salida con muestra de errores
Error	BOOL	FALSE	Si Error = TRUE, existe al menos un mensaje de error en el ciclo actual.
ErrorBits	DWORD	DW#16#0	El parámetro ErrorBits muestra qué mensajes de error existen.

*Nota.* Tabla adaptada de ayuda para módulos PID Compact de la plataforma de TIA Portal V17.



Las variables de configuración se presentan interna del módulo PID\_Compact se presentan en la Tabla 7.

**Tabla 7**

*Variables estáticas de PID\_Compact*

Variable	Tipo de datos	Ajuste predeterminado	Descripción
IntegralResetMode	INT	Para V2.2: 1, y V2.3: 4	Posibilidades: 0, 1, 2, 3 y, 4
RunModeByStartup	BOOL	TRUE	Activar Mode después del re arranque de la CPU
LoadBackUp	BOOL	FALSE	Si LoadBackUp = TRUE, se carga parámetros PID.
PhysicalUnit	INT	0	Unidad física del valor real y de la consigna.
Progress	REAL	0.0	Progreso de la optimización en porcentaje (0,0 - 100,0)
Config.InputPerOn	BOOL	TRUE	TRUE, se utiliza Input_PER, FALSE, se utiliza el parámetro Input.
Config.InvertControl	BOOL	FALSE	Inversión del sentido de regulación.
Config.InputUpperLimit	REAL	120.0	Límite superior del valor real.
Config.InputLowerLimit	REAL	0.0	Límite inferior del valor real.
Config.InputUpperWarning	REAL	3.402822e+38	Límite superior de advertencia del valor real.
Config.InputLowerWarning	REAL	-3.402822e+38	Límite inferior de advertencia del valor real.
Config.OutputUpperLimit	REAL	100.0	Límite superior del valor de salida.
Config.OutputLowerLimit	REAL	0.0	Límite inferior del valor de salida.
Config.SetpointUpperLimit	REAL	3.402822e+38	Límite superior de la consigna.
Config.SetpointLowerLimit	REAL	-3.402822e+38	Límite inferior de la consigna.
Config.MinimumOnTime	REAL	0.0	El tiempo mínimo de conexión en segundos.
Config.MinimumOffTime	REAL	0.0	El tiempo mínimo de desconexión en segundos.
Config.InputScaling.UpperPointIn	REAL	27648.0	Escalado Input_PER arriba.
Config.InputScaling.LowerPointIn	REAL	0.0	Escalado Input_PER abajo.
Config.InputScaling.UpperPointOut	REAL	100.0	Valor real superior escalado.
Config.InputScaling.LowerPointOut	REAL	0.0	Valor real inferior escalado.
CycleTime.EnEstimation	BOOL	TRUE	Calcula el tiempo de muestreo PID.

*Nota.* Tabla adaptada de ayuda para módulos PID Compact de la plataforma de TIA Portal V17.

Las variables indicadoras internas del PID\_Compact se muestra en la Tabla 8.

**Tabla 8**

*Variables almacenadas de ganancia  $K_p$ ,  $T_i$ ,  $T_d$ , ponderaciones y parámetros calculados durante la optimización de procesos con PID\_Compact*

Variable	Tipo de datos	Ajuste predeterminado	Descripción
CtrlParamsBackUp.Gain	REAL	1.0	Ganancia proporcional almacenada.
CtrlParamsBackUp.Ti	REAL	20.0	Tiempo de integración [s] almacenado.
CtrlParamsBackUp.Td	REAL	0.0	Tiempo derivativo [s] almacenado.
CtrlParamsBackUp.PWeighting	REAL	1.0	Factor de ponderación almacenado de la acción P.
CtrlParamsBackUp.DWeighting	REAL	1.0	Factor de ponderación almacenado de la acción D.
PIDSelfTune.SUT.CalculateParams	BOOL	FALSE	Indica propiedades del sistema regulado.
PIDSelfTune.SUT.TuneRule	INT	0	Halla parámetros durante la optimización inicial.
PIDSelfTune.SUT.State	INT	0	La variable SUT.State muestra la fase actual de la optimización inicial.
PIDSelfTune.TIR.RunIn	BOOL	FALSE	Con la variable RunIn se determina que se realice una optimización.
PIDSelfTune.TIR.TuneRule	INT	0	Calcular los parámetros durante la optimización fina
PIDSelfTune.TIR.State	INT	0	La variable TIR.State muestra la fase actual de la Optimización fina.
PIDCtrl.IntegralSum	REAL	0.0	Acción I actual.
PIDCtrl.PIDInit	BOOL	FALSE	Si PIDCtrl.PIDInit = TRUE en el "Modo automático",
Retain.CtrlParams.Gain	REAL	1.0	Ganancia proporcional activa.
Retain.CtrlParams.Ti	REAL	20.0	Ti es remanente.
Retain.CtrlParams.Td	REAL	0.0	Td es remanente.
Retain.CtrlParams.TdFiltRatio	REAL	0.2	Coefficiente activo para el retardo de la acción derivada.
Retain.CtrlParams.PWeighting	REAL	1.0	Ponderación activa de la acción P.
Retain.CtrlParams.DWeighting	REAL	1.0	Ponderación activa de la acción D.
Retain.CtrlParams.Cycle	REAL	1.0	Tiempo de muestreo activo del algoritmo PID.

*Nota.* Tabla adaptada de ayuda para módulos PID Compact de la plataforma de TIA Portal V17.

### 2.2.3. Procesos industriales

El proceso se refiere a los métodos de cambiar o depurar materias primas para crear productos finales. Las materias primas, que pasan a través o permanecen en el fluido de proceso, pueden ser transferidas, medidas, mezcladas, calentadas o enfriadas, almacenadas o manipuladas de una u otra manera para obtener el producto final (Smith & Corripio, 1991).

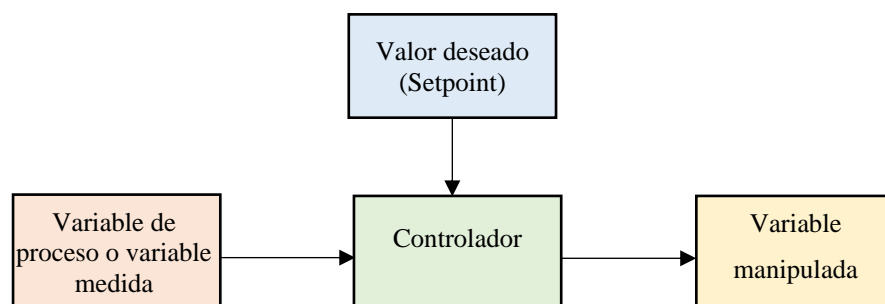
El control de proceso engloba a los métodos que se emplean para controlar variables de proceso cuando se elabora un producto. Factores tales como la proporción entre componentes que se mezclan, su temperatura y presión, etc., pueden afectar significativamente la calidad del producto final. Los fabricantes requieren controlar el proceso de producción por tres razones:

- Disminuir la variabilidad.
- Aumentar la eficacia.
- Garantizar la seguridad.

La Figura 15, presenta las diferentes variables de un proceso, estas son: variable medida, variable de proceso y variable manipulada. La variable medida es la condición del fluido de proceso que tiene que mantenerse en el valor deseado predeterminado. A veces, la variable medida no es la misma que la variable de proceso. El factor que cambia, para mantener la variable medida en el valor deseado, se denomina variable manipulada.

**Figura 15**

*Tipos de variables en un proceso de control*

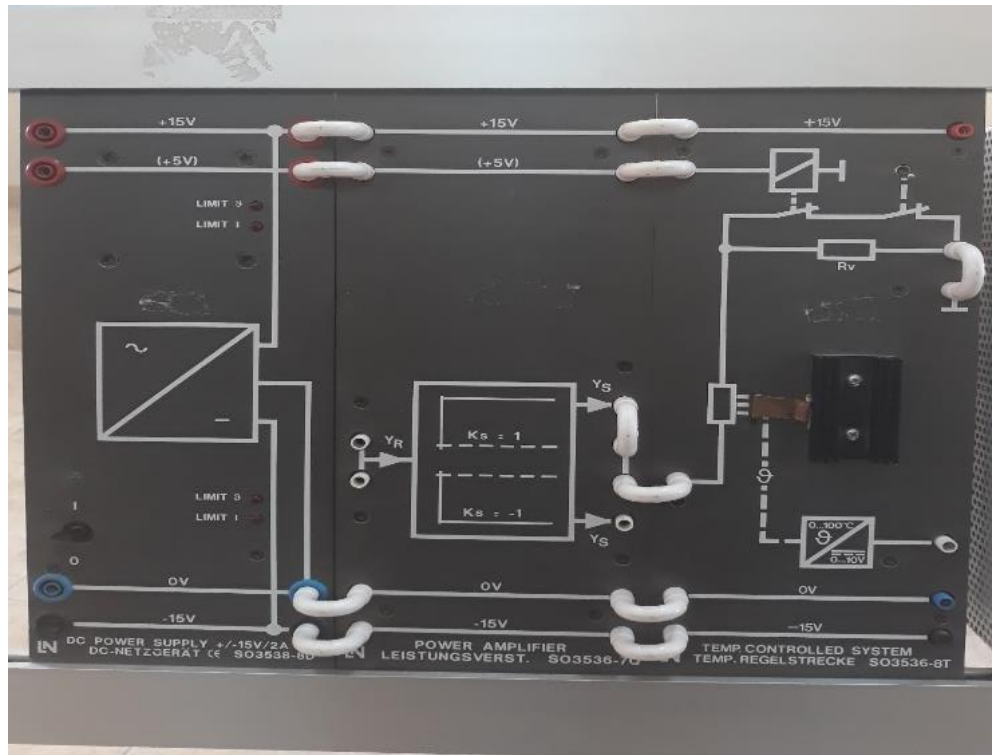


### 2.2.3.1. Planta térmica

Para el control PID la planta térmica debe ser lineal, para esto, se empleó la planta térmica del laboratorio de Control y Automatización de la EPIE-FIES-UNH basado en el módulo SO3536-8T (Alberto & Mendoza, 2021). Al aplicar una tensión de 0 a 10 V en la entrada una resistencia calefactora se activa y calienta la planta; un sensor ubicado en la misma planta mide la temperatura en el rango de 0 a 100 °C. La Figura 16 muestra el módulo de la planta.

**Figura 16**

*Módulo de la planta térmica para el control PID*



*Nota.* Figura obtenida del laboratorio de control y automatización de la EPIE-FIES-UNH

En el módulo de la planta térmica el sensor se encuentra en serie con la planta, y para una temperatura ambiente de 25 °C se da una función de transferencia representado por la ecuación 9.

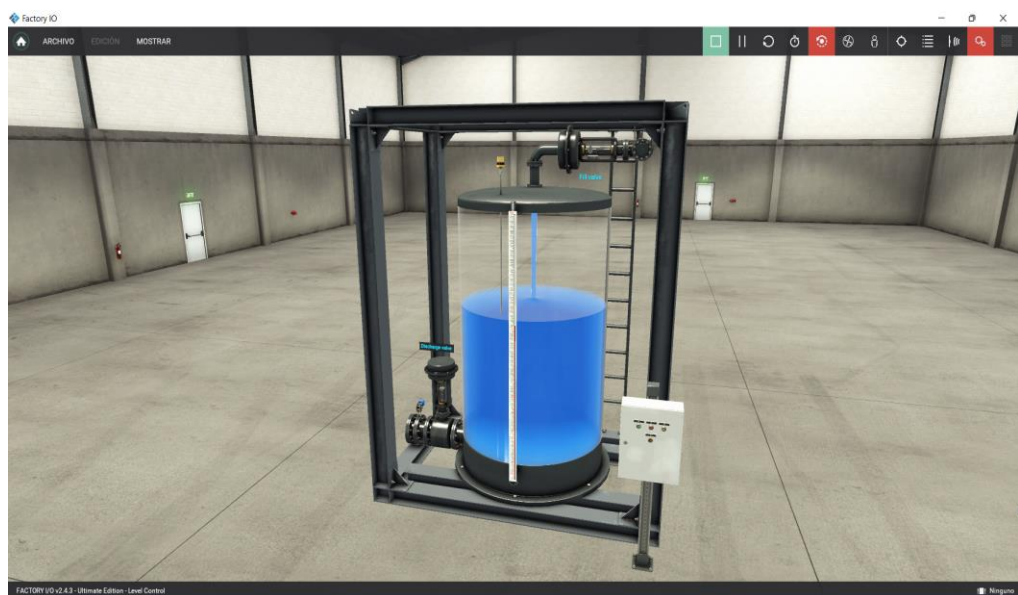
$$G(s) = \frac{4.76e^{-7s}}{24s + 1} \quad (9)$$

### 2.2.3.2. Planta de nivel en tanque

Para el desarrollo de esta tesis se empleó la estación de nivel, que posee un tanque para líquido, 01 electroválvula de llenado, 01 electroválvula de salida, 01 sensor de flujo y 01 sensor de nivel con capacidad de medición de hasta 3000 mm. Las electroválvulas son configurables para operaciones tipo On/Off o analógico. Además, esta estación posee un tablero de control local que permite activar la conexión de los parámetros (ver Figura 17) a los nodos que interactúan con el PLC S7 1500 por medio de la comunicación OPC UA.

### Figura 17

*Estación Factory IO para control de nivel de líquido en tanque*



*Nota.* Figura obtenida de captura de la escena control de nivel de líquido de Factory IO

### 2.2.4. Comunicación OPC UA

Productivity Collaboration Unified Architecture (OPC UA) se desarrolló para superar las limitaciones presentadas por su predecesor el OPC clásico (Portilla, s. f.). El mismo que fue diseñado para funcionar solo con sistemas operativos Windows. OPC UA es actualmente una plataforma independiente para fabricantes y por lo tanto abierta ya que asegura un flujo de información entre componentes de diferentes proveedores, siendo la Fundación OPC la

asociación responsable del desarrollo de esta norma OPC UA cuenta con dos innovaciones o características a destacar del protocolo OPC clásico (Caiza et al., 2020).

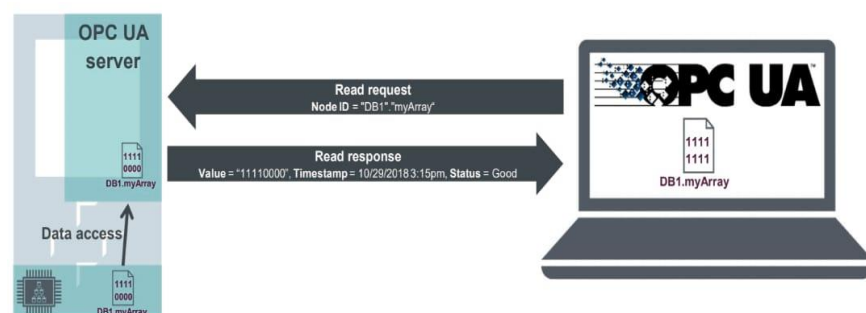
La primera innovación sustituye el protocolo COM y DCOM, específico de Windows, por protocolos abiertos e independientes para que puedan funcionar bajo otros sistemas operativos. Con una implementación importante no solo a nivel de equipos informáticos como Windows, Linux, Mac, etc.; pero, también a nivel de dispositivos de telefonía móvil como Android y diferentes tipos de controles y dispositivos de monitorización, sensores, controladores, etc. Interaccionan con el mundo real incorporando mecanismos de seguridad adicionales.

La segunda característica es que incorpora el sistema orientado a objetos modelo de información que incorpora las tradicionales funcionalidades OPC (tales como acceso a datos, datos históricos, alarmas, eventos, condiciones, etc.) y otras nuevas e innovadoras orientadas a tipos y métodos de datos (García et al., 2016).

El hecho de que la estructura del direccionamiento en los servidores OPC UA están orientados a objetos y el hecho de que la interfaz para acceder a la estructura sea genérica significa que OPC UA se considera un lenguaje de comunicación industrial muy potente (ver Figura 18).

### Figura 18

*Comunicación cliente servidor con OPC UA*



*Nota.* Figura obtenida de OPC UA Server & Client (Siemens, 2019b, p. 5)

#### ***2.2.4.1. La OPC UA en procesos industriales***

La OPC UA define el modo de intercambio de información entre procesos y sistemas de ingeniería industrial. OPC UA mejora el antiguo OPC Classic con funciones ampliadas en términos de modelado de datos, arquitectura de espacio de direcciones, funcionalidades de descubrimiento y seguridad. En OPC UA, los servidores contienen el modelo de información estructurada que representa los datos y el modelo de comunicación es cliente-servidor (Hildebrandt et al., 2020).

La OPC UA se viene incluyendo en gran número de máquinas y sistemas, es un método que facilita la comunicación de proceso a proceso. OPC UA es la interfaz para exportar datos de los sistemas mientras que, internamente, la automatización a nivel de campo todavía se implementa con redes de bus de campo tradicionales. Con OPC UA, los datos en el espacio de direcciones pueden modelarse sin estar restringidos a un protocolo de comunicación específico, obteniendo flujos de información entre sistemas heterogéneos con diferentes modelos de datos. Por lo que, OPC UA es el principal protocolo troncal para la armonización de diferentes redes y sistemas de automatización industrial (Lee et al., 2017).

#### ***2.2.4.2. Esquema OPC UA***

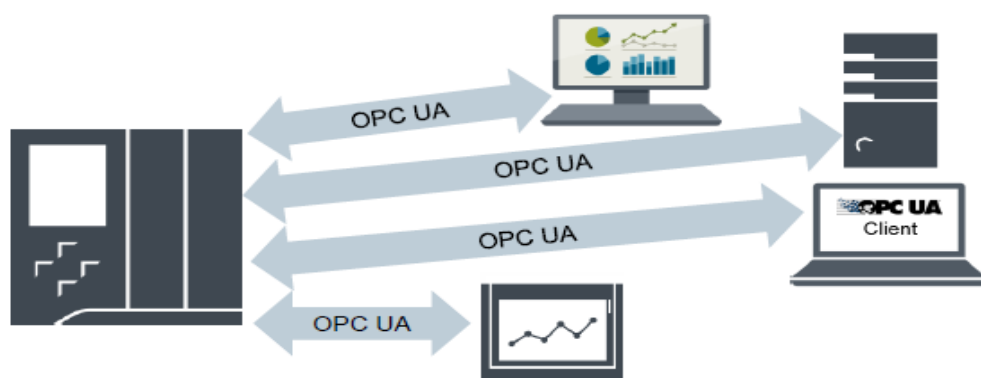
OPC UA ha sido diseñado para facilitar el intercambio de información a través de la jerarquía de sistemas y coexisten con los recursos de planificación empresariales (ERP); sistemas de ejecución de fabricación (MES); sistemas de control; así mismo, con los dispositivos de campo. OPC UA tiene una comunicación basada en mensajes y una arquitectura orientada a servicios (SOA) con clientes y servidores conectados a cualquier tipo de redes.

Una aplicación cliente puede utilizar la API de cliente OPC UA (interfaz de programa de aplicación) para enviar/recibir solicitudes/respuestas de servicio OPC UA hacia/desde el servidor OPC UA. Desde el punto de vista del

programador, la API del cliente OPC UA es como una interfaz que desacopla el código de la aplicación del cliente de la pila de comunicación OPC UA del cliente. En la API de OPC UA, hay un servicio de descubrimiento que se puede utilizar para encontrar servidores OPC UA disponibles y explorar su espacio de direcciones. Claramente, la pila de comunicación de OPC UA convierte las llamadas a la API de OPC UA en mensajes adecuados para las capas de red subyacentes. La Figura 19 muestra la comunicación OPC UA e intercambio de datos por nodos (Ferrari et al., 2018).

### Figura 19

*Comunicación OPC UA e intercambio de datos entre dispositivos*



*Nota.* Figura obtenida de “Evaluation of communication delay in IoT applications based on OPC UA”, (Ferrari et al., 2018)

En los servidores, la API del servidor OPC UA y la pila de comunicación OPC UA son muy similares a las del cliente. Como característica adicional, el servidor dispone del denominado “espacio de direcciones” en el que puede exponer el objeto a intercambiar. En OPC UA, puede existir una multiplicidad de estructuras de datos (llamadas "nodos") que representan: variables, objetos complejos, funciones llamadas remotamente y definiciones de nuevos tipos para crear nuevos metadatos OPC UA.

Se puede crear una estructura jerárquica de complejidad arbitraria con OPC UA, ya que un nodo de objeto puede contener otras variables, objetos, métodos, etc. En otras palabras, el espacio de direcciones OPC UA es el modelo de



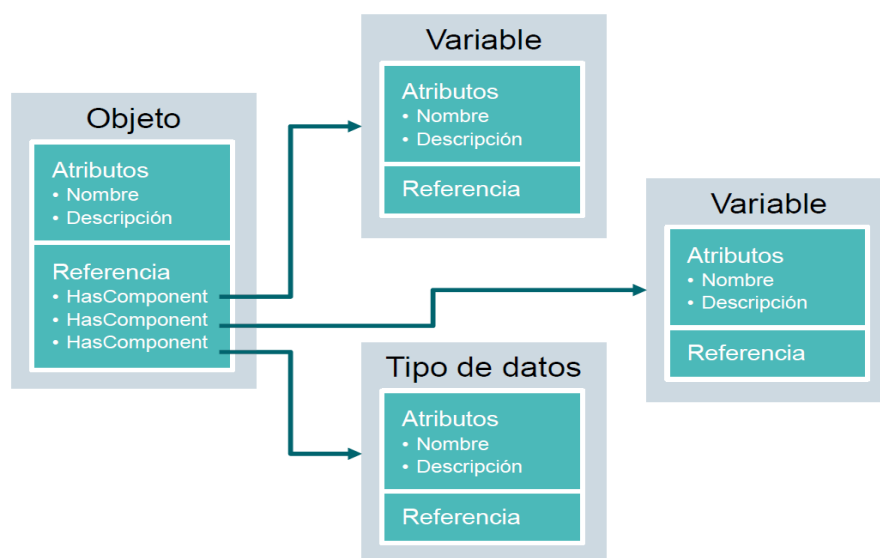
información para la comunicación: dispositivos de hardware reales o “objetos” de software reales como: sensores, actuadores, aplicaciones de software, etc.

#### 2.2.4.3. *Espacio de direcciones OPC UA*

Los nodos en el espacio de direccionamiento OPC UA corresponde a un objeto, variable o método y se describe con una lista de atributos (ver Figura 20). Los nodos tienen atributos comunes, como el nombre o la descripción, y atributos específicos, que representan el valor de una variable. La lista de atributos no se puede ampliar. Es posible añadir información adicional como Property (propiedad) a través del nodo. Las Properties (propiedades) son un tipo especial de variables. Los nodos se conectan entre sí con referencias, las cuales están tipificadas. Hay dos grupos principales: referencias jerárquicas o referencias no jerárquicas, como p. ej. HasTypeDefinition para la conexión de una instancia de objeto a un tipo de objeto (Siemens, 2019a). La Figura 20 muestra la configuración de nodos y su respectiva conexión.

**Figura 20**

Configuración de nodos y las referencias que las conectan



*Nota.* Figura adaptada de “Documentación didáctica/ para cursos de formación, Módulo TIA Portal 092-300 OPC UA con SIMATIC S7-1500 como servidor OPC, así como OPC SCOUT y SIMIT como clientes OPC” (Siemens, 2019a, p. 9)

#### 2.2.4.4. Tipos de nodos disponibles

La Tabla 9 muestra los tipos de nodos definidos para comunicación OPC UA.

**Tabla 9**

*Tipos de nodos y su descripción*

Tipo de nodo	Descripción
Objeto	Sirve como contenedor tipificado o carpeta para variables, métodos y eventos.
Variable	Representa los datos de los objetos o, como atributo, las propiedades de un nodo.
Método	Son componentes de objetos y pueden contener una lista de parámetros de entrada y salida. Los parámetros se describen con atributos definidos.
View	Los nodos tipo View representan una parte del espacio de direcciones. Un nodo hace las veces de punto de acceso y de filtro al navegar.
Tipo de objeto	Suministran información relativa a la estructura y los componentes de un objeto.
Tipo de variable	Describen qué atributos o tipos de datos se pueden encontrar en una instancia de una variable.
Tipo de referencia	Definen los posibles tipos de referencias entre nodos.
Tipo de datos	Describen el contenido del valor de una variable.

Nota. Adaptada de “Documentación didáctica/ para cursos de formación, Módulo TIA Portal 092-300 OPC UA con SIMATIC S7-1500 como servidor OPC, así como OPC SCOUT y SIMIT como clientes OPC” (Siemens, 2019a, p. 10)

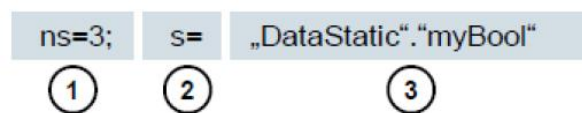
#### 2.2.4.5. Namespaces y Node ID

Cada nodo del espacio de direcciones OPC UA se identifica de manera unívoca mediante una Node ID (ID de nodo). Este Node ID se compone de un

Namespace (espacio de nombres), para diferenciarla de los identificadores de distintos subsistemas, y un identificador que puede ser un valor numérico, un string o un GUID, por lo general se utilizan strings. Los valores numéricos se utilizan para espacios de nombre estáticos. OPC UA define un Namespace con su correspondiente Namespaceindex (índice del espacio de nombres) para los nodos definidos por la OPC Foundation. Los servidores OPC UA definen además uno o varios Namespaces con índice. Los Namespaces definidos por el servidor son variables y pueden cambiar (ver Figura 21).

**Figura 21**

*Estructura de un Node ID para comunicación OPC UA*



La estructura de un Node ID se presenta en Tabla 10.

**Tabla 10**

*Tipo de estructura de un Node ID*

1.	Namespace-Index
2.	Tipo de Node ID (s = string; i = numérica; g =GUID)
3.	ID

#### **2.2.4.6. Servidor OPC UA del S7-1500**

El servidor OPC UA del S7-1500 soporta los siguientes servicios para el acceso a datos: Read, Write, Registered read/write, Subscriptions.

Para leer/escribir variables de un PLC S7-1500, y aumentar notoriamente el rendimiento estructurando de las variables se debe usar arrays y estructuras. Los arrays ofrecen el mejor rendimiento, son de 2 a 3 veces más rápidos que las estructuras. Estas son aproximadamente de 10 a 100 veces más rápidas que

los accesos individuales. Se utiliza "Registered read/write" en el caso de accesos recurrentes para aumentar más el rendimiento.

Es necesario contar con licencias de acuerdo a las versiones de los PLC estas licencias pueden ser cortas, medianas o largas (ver Tabla 11).

**Tabla 11**

*Sistema de licencias para CPU S7 1500*

<b>Tipo de CPU</b>	ET 200SP CPU a S7-1513(F)	1515/1516(F)	1517/1518(F)
<b>Licencia necesaria</b>	Small	Medium	Large

### 2.2.5. Clientes OPC UA

Los softwares clientes utilizados son el UA Expert y la aplicación implementada en Wincc Unified.

#### 2.2.5.1. UaExpert

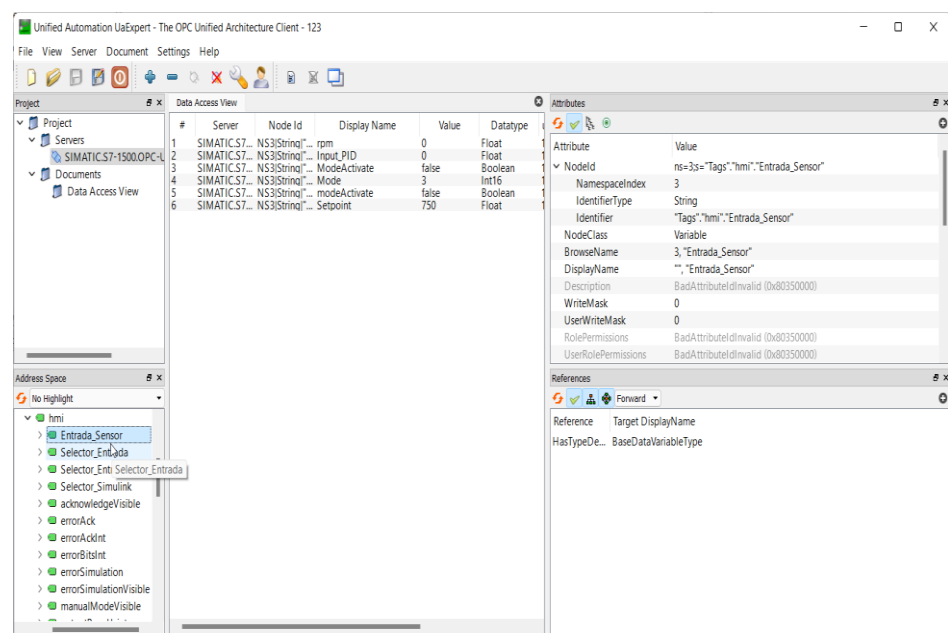
El UA Expert, es un cliente OPC UA con todas las funciones para DataAccess, Alarmas y condiciones, Acceso a históricos de datos y llamada de métodos UA. OPC UA es multiplataforma programado en C++ (Unified Automation GmbH, 2018). Las versiones libres de UaExpert viene con los siguientes complementos:

- Vista y acceso a datos OPC UA
- Vista de alarmas y condiciones de OPC UA
- Vista de tendencia histórica de OPC UA
- Vista de diagnóstico del servidor
- Complemento de rendimiento OPC UA
- XMLNodeSet-Export View (requiere licencia)
- El UaExpert está disponible para Windows y Linux.

1. **Vista y acceso a datos OPC UA:** Este complemento se muestra en el panel central de UaExpert de forma predeterminada. Puede seleccionar múltiples nodos UA en la ventana del espacio de direcciones y arrastrarlos y soltarlos en la vista DA. La Vista DA crea una suscripción y monitorea los Nodos. La frecuencia de muestreo y el intervalo de suscripción se pueden cambiar haciendo clic derecho en la vista DA. Al hacer doble clic en la columna Valor de un nodo específico, puede escribir nuevos valores en ese nodo, se admite la escritura de tipos escalar, matriz y matriz. La vista DA se diseñó para mostrar la vista clásica en los servidores OPC, concentrándose únicamente en la supervisión de elementos y la visualización de valores, la marca de tiempo y el estado de los nodos individuales (ver Figura 22).

**Figura 22**

*Vista y acceso a datos del servidor OPC UA en UaExpert*



2. **Vista de alarmas y condiciones de UaExpert:** El documento de Vista de eventos se puede agregar usando el botón Agregar documento en la barra de menú. El complemento de eventos se mostrará en el panel central y consta de tres grupos principales, la configuración, la vista de eventos/alarmas y la vista de detalles que muestra la información detallada

de una alarma seleccionada individualmente. Debe seleccionar un nodo UA en el navegador de espacio de direcciones y arrastrarlo y soltarlo en el grupo de configuración de la vista de eventos. Ahora puede seleccionar EventFields que le interesen para este objeto de eventos, algunos campos ya están configurados de forma predeterminada. Siempre que el objeto active un evento, se mostrará en el grupo central de la Vista de eventos. Aquí puede cambiar las pestañas entre Eventos y Alarma y mostrarle la lista histórica de eventos o el estado actual de alarma pendiente. Al hacer clic en un evento, el grupo del panel inferior le mostrará todos los detalles de este evento en particular de acuerdo con los campos de eventos seleccionados que marcó en la configuración.

3. ***Vista de tendencia histórica de UaExpert:*** El documento Vista histórica se puede agregar usando el botón Agregar documento en la barra de menú. El complemento Histórico se mostrará en el panel central y consta de dos grupos principales, la Configuración y la vista de datos históricos que muestran los valores en una vista de tendencia gráfica relacionada con el marco de tiempo solicitado.
4. ***Vista de rendimiento de UaExpert:*** El documento Vista de rendimiento se obtiene usando el botón “Agregar documento” en la barra de menú. El complemento de rendimiento se mostrará en el panel central y consta de tres grupos principales, la configuración, la lista de nodos utilizados y los resultados que muestran la medición en una vista gráfica.

#### ***2.2.5.2. Simatic WinCC Unified***

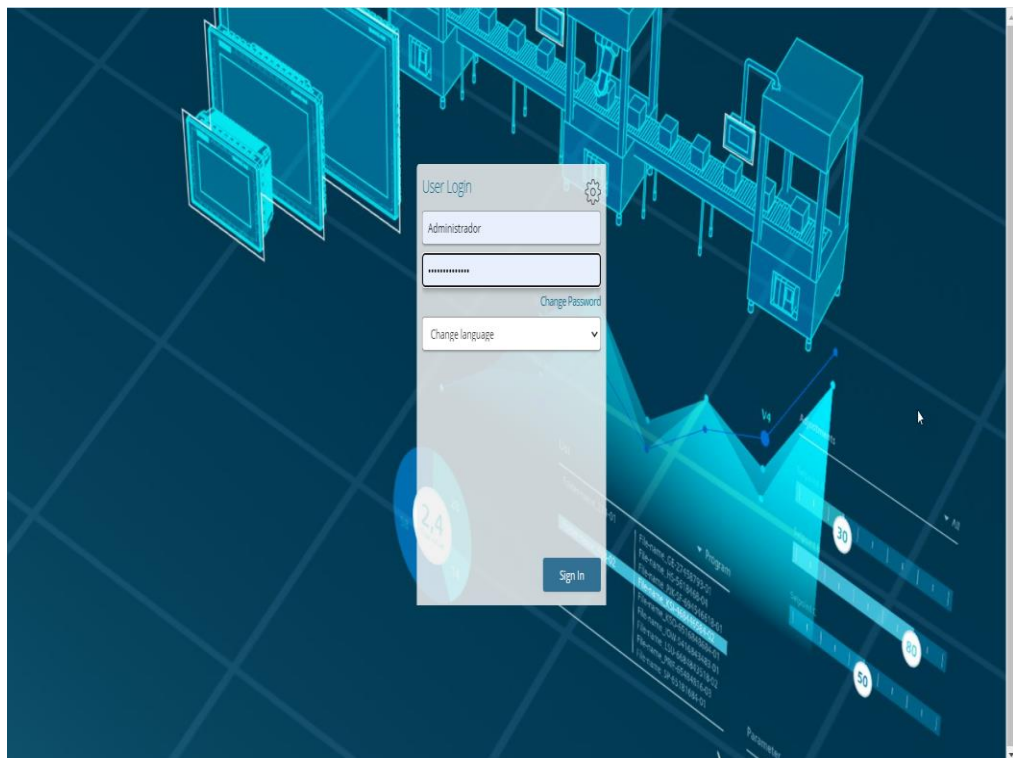
WinCC Unified es un nuevo software utilizado en TIA Portal a partir de la versión 16, lanzado en 2019. La diferencia más significativa con el software WinCC anterior; en Unified es su naturaleza basada en navegador. Se basa en tecnologías de Internet como HTML5, SVG y JavaScript. JavaScript actúa como un lenguaje de scripting en todos los dispositivos, tanto en los paneles como en las computadoras de la sala de control. Las secuencias de comandos se pueden utilizar para todas las funciones, como el procesamiento de datos, la

creación de objetos, etc. Las tecnologías de Internet permiten interfaces abiertas (Siemens, 2019d).

Los usuarios pueden acceder al sistema a través de varios dispositivos, como computadoras, tabletas y teléfonos inteligentes, y no se requieren descargas por separado. El sistema unificado es escalable, lo que significa que los programas escalan desde paneles hasta sistemas SCADA completos. En la Figura 23 se presenta la interface de Simatic WinCC Unified en ejecución a través de Simatic Runtime, esta ejecución es posible en todos los navegadores que soporten HTML5 (Siemens, 2019c). Siemens recomienda usar Google Chrome, pero también se ha probado la compatibilidad de Microsoft Edge, Mozilla Firefox ESR y Apple Safari con WinCC Unified. La tendencia de los programas de WinCC Unified también estarán disponibles en servicios en la nube (Siemens, 2019c).

### **Figura 23**

*Simatic WinCC Unified en ejecución*

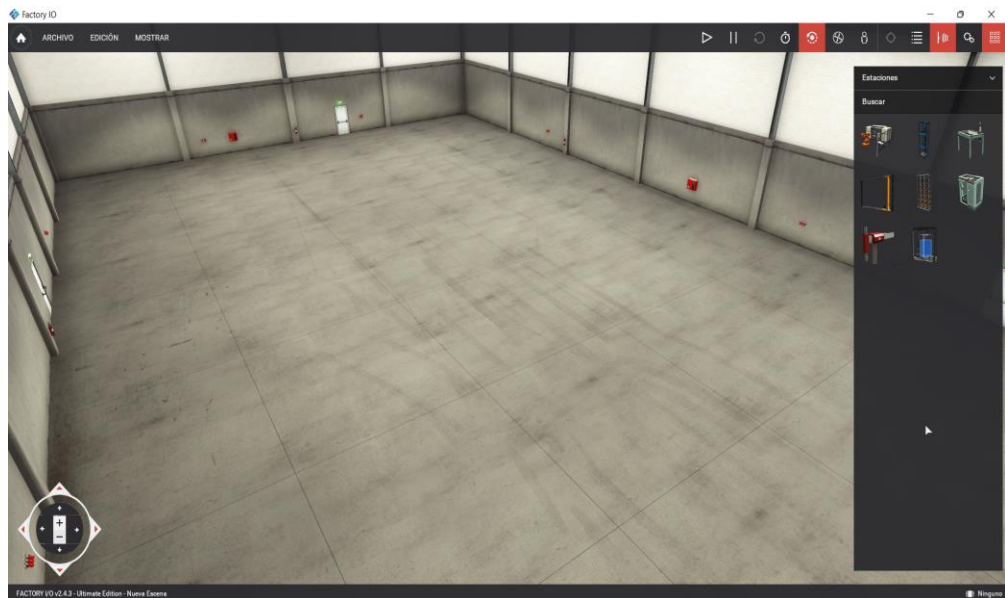


### 2.2.5.3. *Factory IO*

Es un simulador 3D de plantas industriales, los componentes que se pueden utilizar para la Automatización industrial en Factory IO son estaciones, operadores, dispositivos de advertencia, pasarelas, partes de carga y otras partes para configurar libremente un entorno cualquiera (Real Games, 2022). Posee ventanas para implementación de escenas y configuración de driver (ver Figura 24)

#### **Figura 24**

*Factory IO: Escena para implementación de procesos*



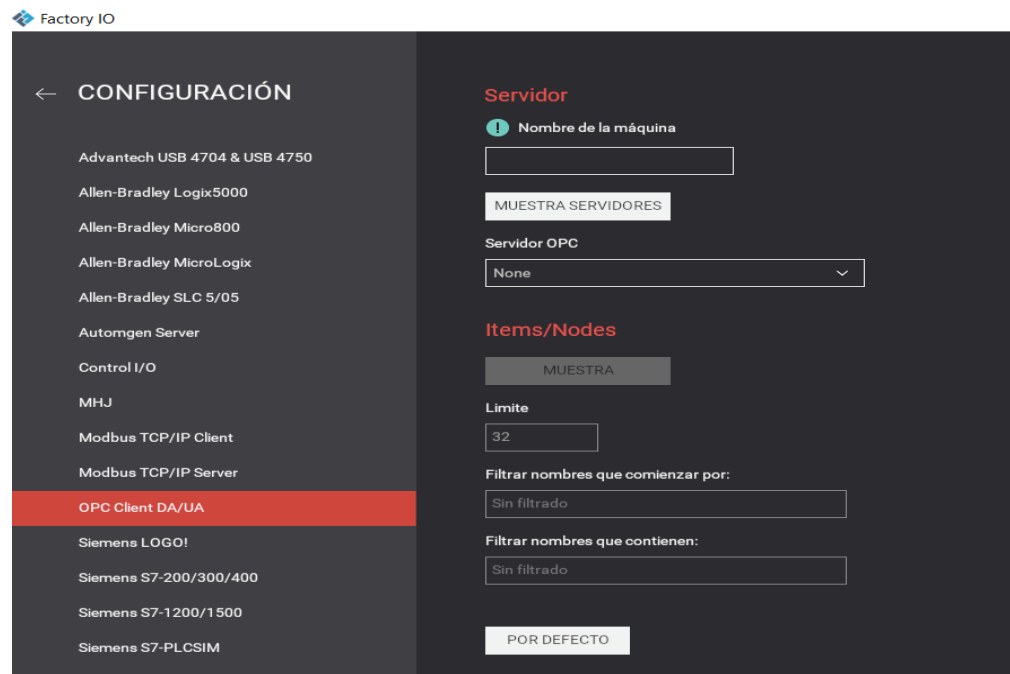
Para construir un proceso en Factory IO, se tiene varios tipos de sensores: capacitivos, difusos, inductivos, retroreflectivos, de visión, arrays de luz y otros sensores. Los actuadores se pueden encontrar integrados en estaciones como centro de maquinado, elevador, pick & place, transelevador, rack, paletizador y tanque.

El controlador de cliente OPC DA/UA (ver Figura 25), permite que Factory I/O intercambie datos con PLC, SoftPLC o cualquier otra tecnología a la que se pueda acceder a través de un Servidor OPC DA/UA. Es compatible con las especificaciones OPC DA 1.0, 2.0, 3.0 (Classic) y UA (Unified Architecture).



**Figura 25**

*Ventana de configuración de OPC Client DA/UA*



## 2.3. HIPÓTESIS

### 2.3.1. Hipótesis general

Es factible la implementación de un controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA, que reduce significativamente el tiempo de asentamiento de variables controladas.

### 2.3.2. Hipótesis específicas

- a. Es factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción en más del 60% del tiempo de asentamiento de la variable temperatura en un proceso industrial térmico para un  $\text{ess} \leq 2\%$ .

- b. Es factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción en más del 20% del tiempo de asentamiento de la variable nivel de líquido en tanque para un  $ess \leq 2\%$ .

## 2.4. OPERACIONALIZACIÓN DE VARIABLES

Las variables identificadas que permitieron demostrar las hipótesis son los siguientes:

### 2.4.1. Variable independiente

- Implementación de un controlador PID autosintonizable.

### 2.4.2. Variables dependientes

- Reducción de tiempo de asentamiento de variables controladas en procesos industriales.

En la Tabla 12 se presenta la definición operativa de variables e indicadores.

**Tabla 12**

*Definición operativa de variables e indicadores*

Variable independiente	Definición operativa	Indicadores	Instrumento/Regla
Implementación de un controlador PID autosintonizable	El controlador PID autosintonizable es implementado en PLC S7 1500 con módulo PID_Compact y comunicación OPC UA bajo las reglas de Ziegler Nichols.	- Kp - Ti - Td	- Regla de Ziegler - Nichols
Variable dependiente	Definición conceptual	Indicadores	Instrumento
Reducción de tiempo de asentamiento en variables de procesos industriales.	La reducción del tiempo de asentamiento de variables controladas en procesos industriales está en relación con el proceso y los parámetros (Kp, Ti y Td) que se establecen para el controlador PID durante la autosintonización desde la interface del cliente implementado en WinCC Unified que soporta comunicación OPC UA.	- Porcentaje (%).	- Interface en WinCC Unified con comunicación OPC UA

## **CAPÍTULO III**

### **MATERIALES Y MÉTODOS**

#### **3.1. MATERIALES**

Los materiales empleados en el desarrollo de la tesis, está constituido por equipos y programas de aplicación.

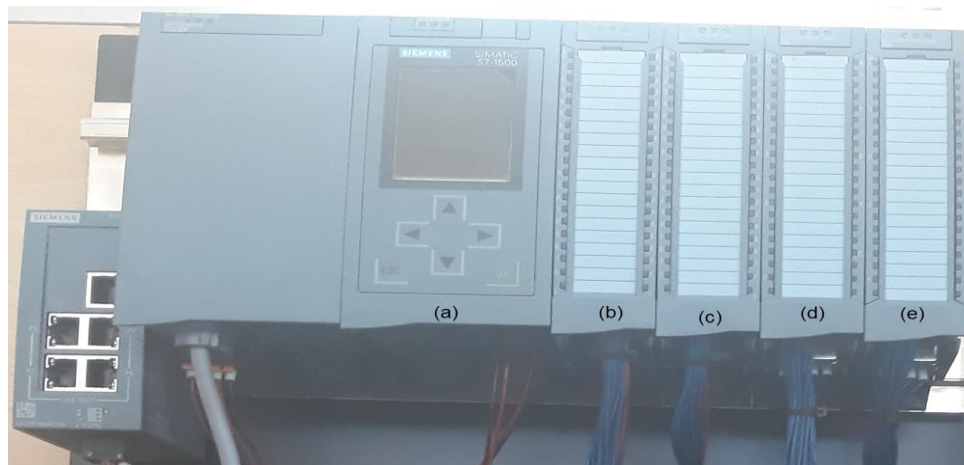
##### **3.1.1. Equipos**

Se empleó el PLC S7 1500, con CPU 1516-3 que soporta comunicación Profinet y Profibus (PN/DP) e instrucción PID\_Compact V2: CPU con display; memoria de trabajo 1 MB para código y 5 MB para datos; tiempo de operación con bits de 10 ns; protocolo de transporte TCP/IP. Incorpora servidor y cliente OPC UA con firmware V2.6. Para la lectura del sensor analógico instalado en la planta térmica se usó el módulo de entradas analógicas AI8 x U/I/RTD/TC de 16 bits, en grupos de 8; 4 canales con medición RTD; la entrada de tensión se parametrizó a 10 V.

Para activar el calefactor en la planta se usó el módulo de salidas analógicas AQ4 x U/I de 16 bits; en grupos de 4; con salida parametrizada a 10 V. La Figura 26 muestra los componentes de la estación de control y comunicación: (a) CPU S7 1500, (b) Módulo de entrada digital de 32 canales, (c) Módulo de salida digital de 32 canales, (d) Módulo de entradas analógicas AI8 x U/I/RTD/TC y (e) Módulo de salidas analógicas AQ4 x U/I.

**Figura 26**

*PLC S7 1500 con módulos de entrada salida digitales y analógicos*

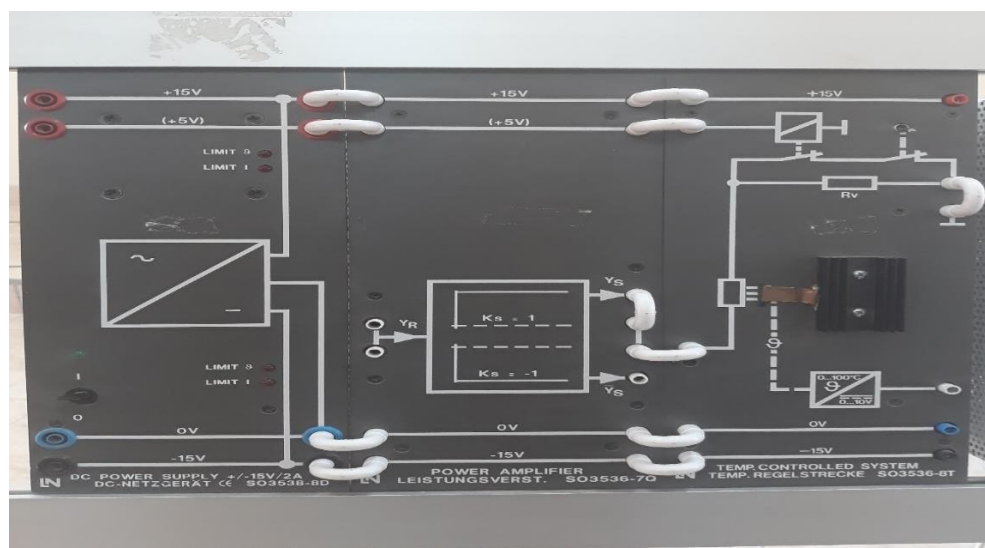


*Nota.* Imagen obtenida del Laboratorio de Control y Automatización de la EPIE-FIES-UNH.

La planta usada para la sintonización térmica está constituida por el módulo SO 3536-8T (ver Figura 27), con rango desde la temperatura ambiente hasta 100 °C, y siendo la temperatura de proceso máximo aproximadamente 96 °C. La tensión de salida del sensor es lineal y es proporcional a la temperatura (0 – 100 °C/0 – 10 V)

**Figura 27**

*Planta térmica con rango desde temperatura ambiente hasta 100 °C*



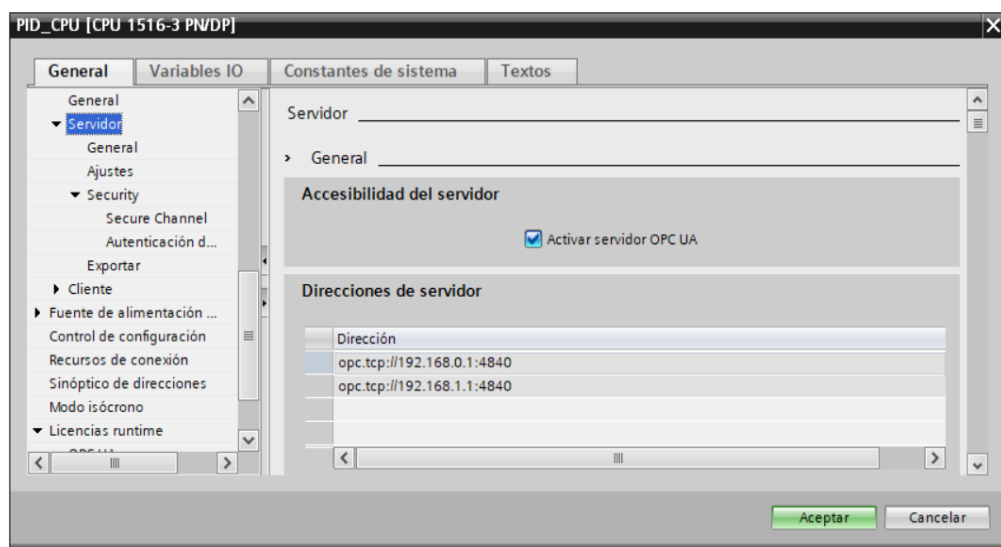
*Nota.* Imagen obtenida del Laboratorio de Control y Automatización de la EPIE-FIES-UNH.

### 3.1.2. Programas de aplicación

En TIA Portal V17 se desarrolló la programación del PLC S7 1500, así como, la activación de la comunicación OPC UA asignando las direcciones `opc.tcp://192.168.0.1:4840` y `opc.tcp://192.168.1.1:4840` (ver Figura 28). La instrucción PID\_Compact fue insertado en TIA Portal para ser configurado en tiempo real desde una estación cliente por medio de la comunicación OPC UA. La programación del PLC se hizo en lenguaje escalera (KOP).

#### Figura 28

*Asignación de direcciones para la comunicación del PLC S7 1500 con OPC UA*

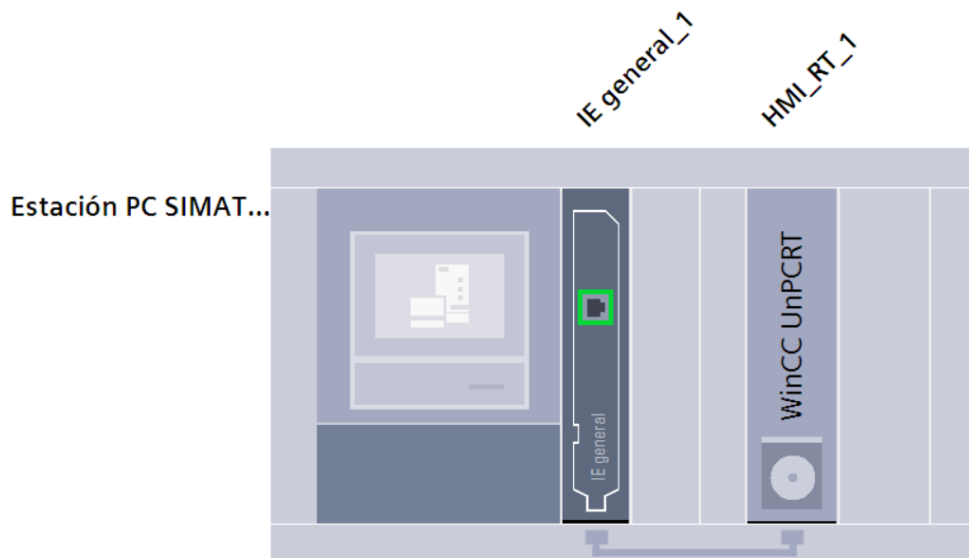


*Nota:* Imagen obtenida de TIA Portal V17.0

En WinCC Unified V17 se implementó las seis imágenes de interface para la autosintonización de procesos industriales. Este software está integrado a TIA portal y está diseñada para ejecución en estaciones de PC, la comunicación se realiza por medio de la tarjeta ethernet (ver Figura 29). La interface entre el PLC y este software se ejecutan en tiempo real.

**Figura 29**

*Módulo de WinCC Unified V17*



*Nota:* Imagen obtenida de TIA portal V17.

## **3.2. TIPO Y NIVEL DE INVESTIGACIÓN**

### **3.2.1. Tipo de investigación**

El trabajo de investigación es del tipo tecnológico; este tipo de investigación tiene por finalidad la creación de artefactos, innovación o desarrollo de procesos con el fin de solucionar un problema (Cegarra, 2004; Espinoza, 2014). Se abordó al problema de sintonización de actuadores de las variables controladas en procesos industriales, mejorando la adaptabilidad de un controlador PID en PLC con interface cliente desarrollado en WinCC Unified con comunicación OPC UA, con la reducción del tiempo de asentamiento de las variables controladas, llevado a cabo en el laboratorio de Control y Automatización de la Escuela Profesional de Ingeniería Electrónica.

### **3.2.2. Nivel de investigación**

Por los objetivos, el nivel de investigación a la que corresponde es aplicativo, ya que se evaluó la factibilidad de la implementación del controlador PID

autosintonizable para procesos industriales integrado a un PLC, comprobando la eficiencia de su operación con la reducción del tiempo de asentamiento de las variables controladas en un proceso térmico y de control de nivel de líquido en tanque.

### **3.3. MÉTODO DE INVESTIGACIÓN**

En esta investigación se empleó el método de procesos lógicos inductivo-deductivo. La implementación del servidor y cliente del controlador PID autosintonizable, al igual que las respuestas del controlador frente a los estímulos hicieron uso del proceso inductivo y la determinación de la disminución del tiempo de asentamiento de las variables controladas, corresponden al proceso deductivo.

La implementación total del controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA, para la reducción significativa del tiempo de asentamiento de las variables, contempló dos aspectos; primero: implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con envío de datos por OPC UA; y segundo: implementación del cliente del controlador PID en WinCC Unified con comunicación OPC UA.

### **3.4. DISEÑO DE INVESTIGACIÓN**

En este trabajo se usó el diseño experimental antes-después (causa – efecto), como establece Hernández y Mendoza (Hernández-Sampieri & Mendoza Torres, 2018, p. 149), se manipuló intencionalmente la variable independiente seleccionados en 02 grupos comparables con participación pre establecida y el otro grupo obtenido por autosintonización, lo que permitió determinar la reducción del tiempo de asentamiento de las variables controladas.

El diseño de la investigación empleado, para el logro de los objetivos de la tesis, se presenta en el esquema de la Figura 30, con el que se determina la reducción del tiempo de asentamiento de las variables controladas.

### Figura 30

*Diagrama del diseño de investigación experimental antes-después*

Grupo experimental:	A	O <sub>1</sub>	X	O <sub>2</sub>
---------------------	---	----------------	---	----------------

Donde: A= Asignación preestablecida de Setpoint para las variables controladas; O<sub>1</sub> = Observaciones de tiempo de asentamiento de variables controladas con parámetros Kp, Ti y Td pre establecidos para el controlador PID; X= tratamiento valor de parámetros Kp, Ti y Td obtenidos por autosintonización y O<sub>2</sub> = Observaciones de tiempo de asentamiento de variables controladas con parámetros Kp, Ti y Td obtenidos al autosintonizar el controlador PID.

Efecto del tratamiento, reducción del tiempo de asentamiento de las variables controladas: (O<sub>1</sub> – O<sub>2</sub>)

## 3.5. POBLACIÓN, MUESTRA Y MUESTREO

### 3.5.1. Población

El trabajo se desarrolló con equipos del laboratorio de Control y Automatización de la Escuela Profesional de Ingeniería Electrónica, donde se realizaron mediciones del tiempo de asentamiento para diferentes consignas de las variables de los procesos industriales, estando constituida la población por el promedio de 20 diferentes tiempos de asentamiento de la variable temperatura en un proceso térmico y promedio de 20 tiempos de asentamiento de la variable nivel en tanque; ambos para un  $e_{ss} \leq 2\%$ .

### 3.5.2. Muestra

Por la naturaleza del estudio la muestra estuvo constituido por toda la población; es decir por el promedio de 20 tiempos de asentamiento de la variable temperatura, 10 con parámetros Kp, Ti y Td pre establecidos y 10 con parámetros obtenidos por autosintonización del proceso térmico para setpoint:



25 °C, 30 °C, 35 °C, 40 °C, 50 °C, 60 °C, 70 °C, 80 °C, 90 °C y 96 °C; cada tiempo de asentamiento se calculó de 35 mediciones, teniendo en total 700 mediciones para esta variable (ver apéndices: A y B)

Por otro lado, para la variable nivel en tanque también se tuvo el promedio de 20 tiempos de asentamiento, 10 con parámetros  $K_p$ ,  $T_i$  y  $T_d$  pre establecidos y 10 con parámetros obtenidos por autosintonización para setpoint: 250 mm, 500 mm, 1000 mm, 1250 mm, 1500 mm, 1750 mm, 2000 mm, 2250 mm, 2500 mm, 2750 mm y 3000 mm; cada tiempo de asentamiento se calculó de 35 mediciones, teniendo en total 700 mediciones también para esta variable (ver apéndice: C y D)

### **3.5.3. Muestreo**

El muestreo fue intencional con fijación de las consignas (Setpoint) de las variables controladas desde la interface HMI del controlador PID autosintonizado, que fue implementado en WinCC Unified.

## **3.6. ANÁLISIS Y PROCESAMIENTO DE DATOS**

El análisis y procesamiento de datos de los tiempos de asentamiento de las variables controladas temperatura y nivel se inició con la sistematización de las muestras; registrando la hora, valor de la entrada de la variable controlada, el Setpoint y el error de estado estable ( $e_{ss}$ ) (ver tablas del anexo A, B, C y D). Se formularon las hipótesis de investigación con sus respectivas hipótesis nulas y alternativas. Las hipótesis nulas se sometieron a prueba mediante el estadístico  $t$  de Student.

## CAPÍTULO IV

### DISCUSIÓN DE RESULTADOS

#### 4.1. RESULTADOS

Como resultados del trabajo de investigación se tiene, primero: la descripción de la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact V2 y regla de Ziegler-Nichols, con envío de datos por OPC UA; segundo: la descripción de la implementación del cliente del controlador PID en WinCC Unified con comunicación OPC UA; y tercero; se evalúa el funcionamiento del mismo, con la determinación del tiempo de asentamiento de la variables: “*Temperatura*” en un proceso térmico y “*Nivel*” de líquido en tanque, para un  $e_{ss} \leq 2\%$  en ambos casos.

##### 4.1.1. Implementación del servidor del controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con envío de datos por OPC UA

El servidor del controlador PID autosintonizable se implementó en el PLC S7 1500, utilizando el bloque de instrucción PID\_Compact V2 y la segunda regla de Ziegler-Nichols, el módulo de entrada analógico se empleó para la adquisición de datos desde los sensores y el módulo de salida digital para envío de señal modulada al actuador. Todos los valores del PLC son transparentes a la comunicación por OPC UA.

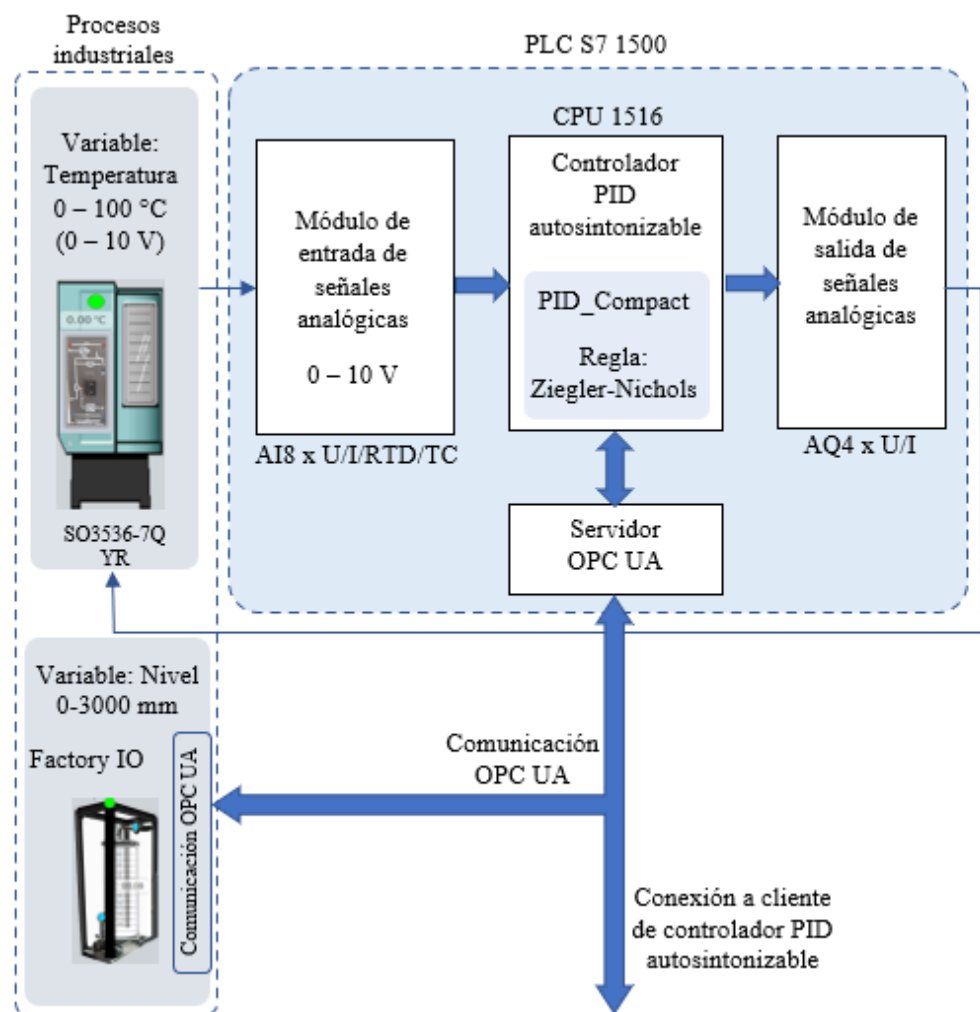
Del diagrama del servidor que se presenta en la Figura 31, las variables a controlar se encuentran en los diferentes procesos, los valores de estas variables son obtenidos por los sensores analógicos en rangos de 0 – 10 V equivalente a los valores físicos (Temperatura de 0 – 100 °C y Nivel de 0 – 3000 mm). La “*Temperatura*” es obtenida por el módulo de entrada analógica y el “*Nivel*” de líquido en tanque es obtenido de los datos provenientes por comunicación OPC UA. Estos valores son enviados al controlador PID implementado usando la

instrucción PID\_Compact; previa selección, normalización y escalado de los valores de las variables.

El controlador PID realiza la regulación de la variable bajo control, enviando una señal hacia el actuador en planta. Para el control de temperatura envía una señal en el rango de 0 – 10V que se conecta al terminal YR del amplificador de potencia SO3536-7Q y para el control de nivel envía valores digitales por OPC UA para apertura o cierra de las válvulas de llenado y vaciado.

### Figura 31

*Diagrama del servidor del controlador PID autosintonizable implementado en PLC S7 1500 con PID\_Compact y envío de datos por OPC UA*

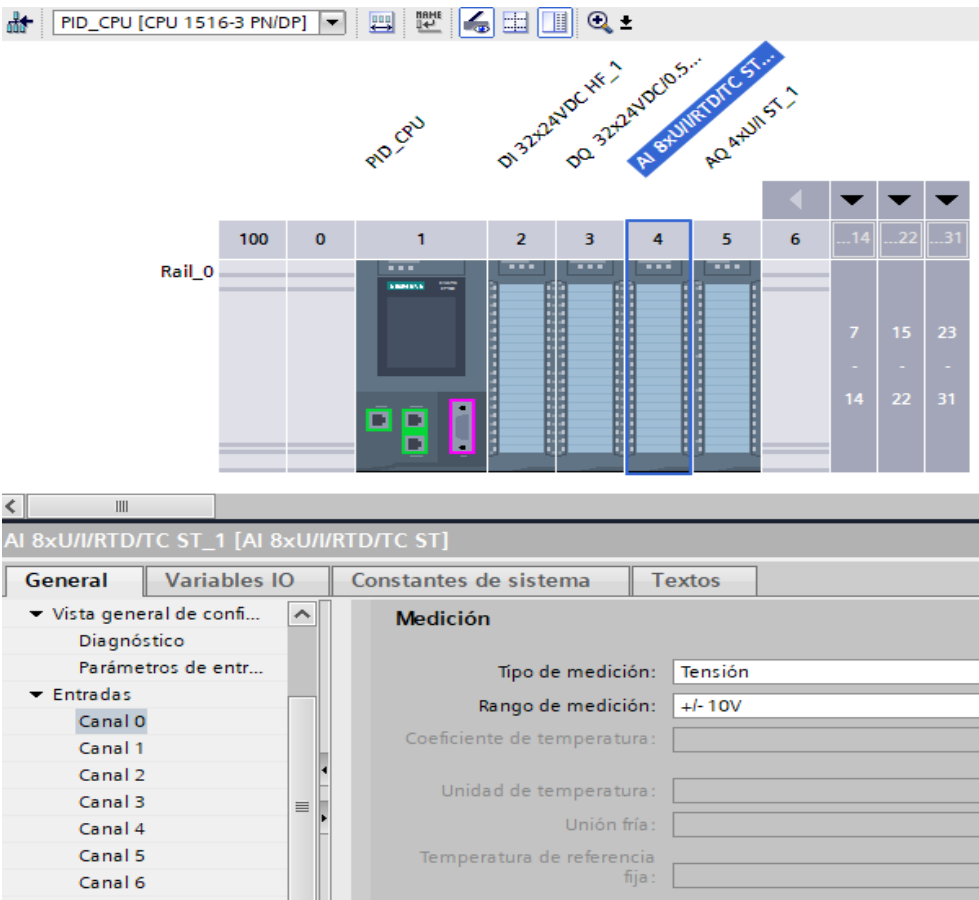


#### 4.1.1.1. Acondicionamiento y selección de variables de procesos industriales

Las señales analógicas de la variable “Temperatura” del proceso térmico, fue adquirido por medio del “Canal 0” del módulo de entrada analógico del PLC S7 1500 (AI 8xU/I/RTD/TC), configurado para medición de tensión en el rango 0 – 10 V en TIA Portal (ver Figura 32); este módulo da como resultado valores discretos proporcionales a las entradas en el rango de 0 a 27648. Estos valores discretos ingresan al módulo de instrucciones PID\_Compact para ser normalizados y escalados de 0 a 100 °C.

**Figura 32**

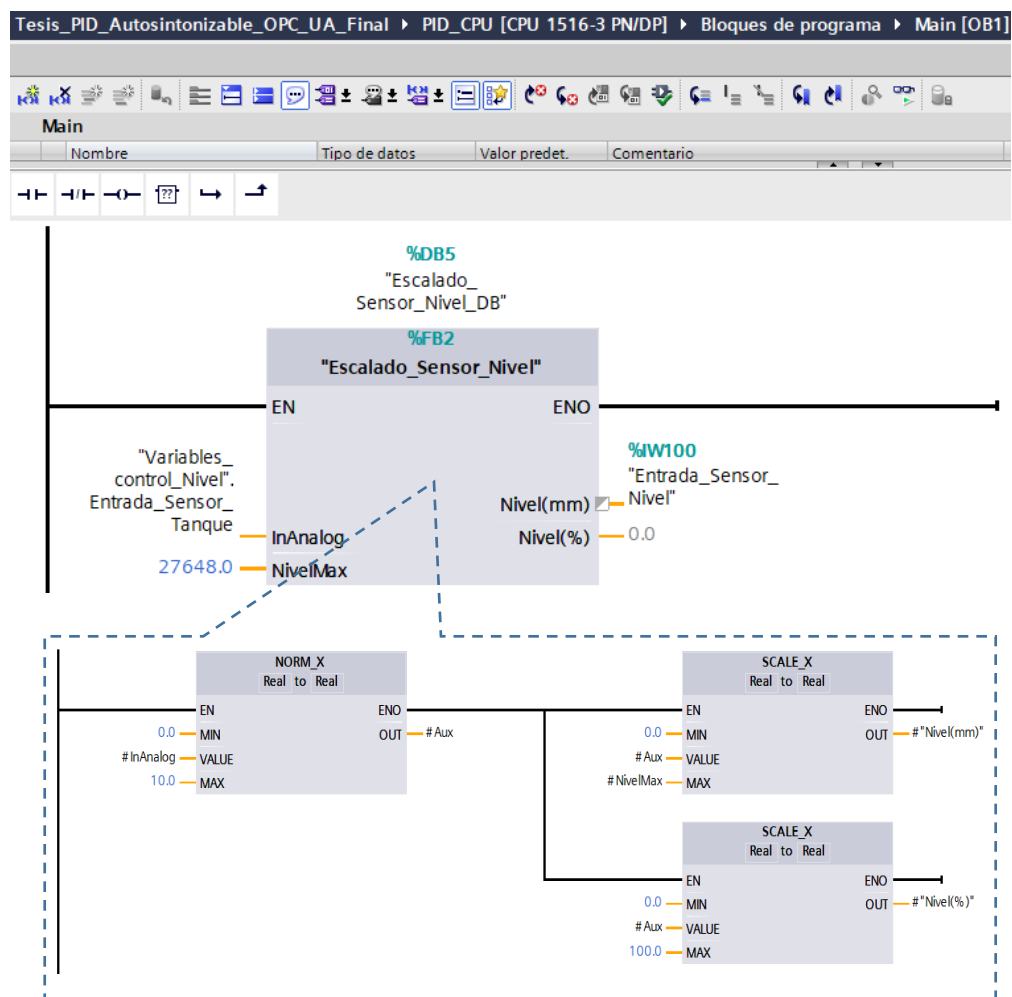
Configuración del “Canal 0” del módulo AI 8xU/I/RTD/TC para medición de temperatura en el rango de 0 – 10 V



La variable “Nivel” tuvo como escala de instrumentación de 0 – 10 V, en relación directa con el rango del nivel de 0 - 3000 mm. Esta señal proveniente de Factory IO transmitido por medio de la comunicación OPC UA, fueron recibidos en valores numéricos de 0.0 – 10.0, en formato real. Estos valores se acondicionaron al rango de 0 – 27648, para la conexión al módulo de instrucciones PID\_Compact. El acondicionamiento de estas señales se llevó a cabo con los bloques de instrucción Norm\_X y Scale\_X e integrado en un bloque de funciones (FB2), ubicado en el bloque de organización Main (OB1) (ver Figura 33).

**Figura 33**

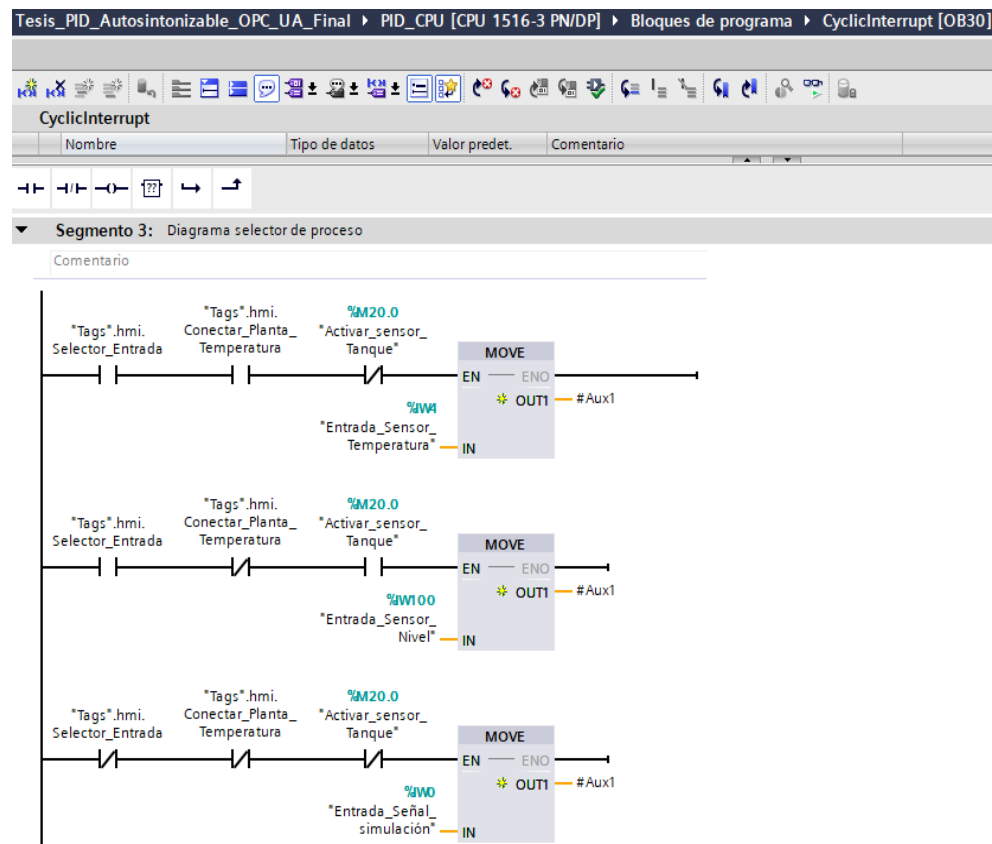
*Diagrama de normalizado y escalado de la variable “Nivel” para conexión a PID\_Compact*



La selección de una de las señales ingresantes al controlador PID\_Compact (Figura 34) se llevó a cabo en el bloque de organización CyclicInterrupt (OB30).

**Figura 34**

*Diagrama de selección de las variables Temperatura y Nivel para ingreso de datos a PID\_Compact*



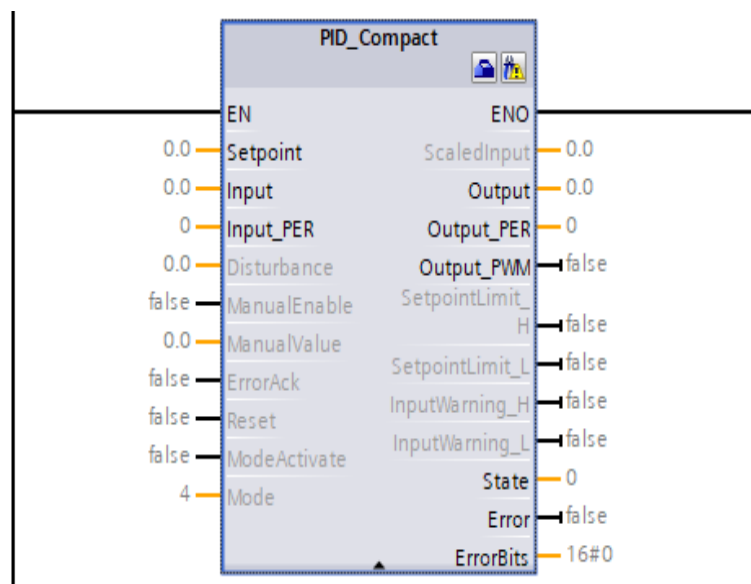
#### 4.1.1.2. Configuración del módulo PID\_Compact

El controlador PID autosintonizable se implementó en el módulo de instrucción PID\_Compact V2 del PLC S7 1500. El bloque de PID\_Compact se insertó en el bloque de organización CyclicInterrupt que es de respuesta rápida (OB30), con almacenamiento de datos de entradas/salidas en el bloque DB1130.

PID\_Compact presenta entradas y salidas definidas: la entrada Setpoint, permite desde la interface del cliente fijar el valor deseado de la variable a controlar; Input permite el ingreso de valores de las variables en el rango de 0 – 100 %; Input\_PER permite el ingreso de los datos analógicos en el rango de 0 – 27648. La salida Output da valores porcentuales (0 – 100) de la señal manipulada o de control que va a los actuadores; Output\_PER da valores para salida analógica (0 - 27648) para control de actuadores; Output\_PWM da señal modulada por ancho de pulso; así mismo, a la salida se tiene el indicador de estado, Error y ErrorBits (ver Figura 35).

**Figura 35**

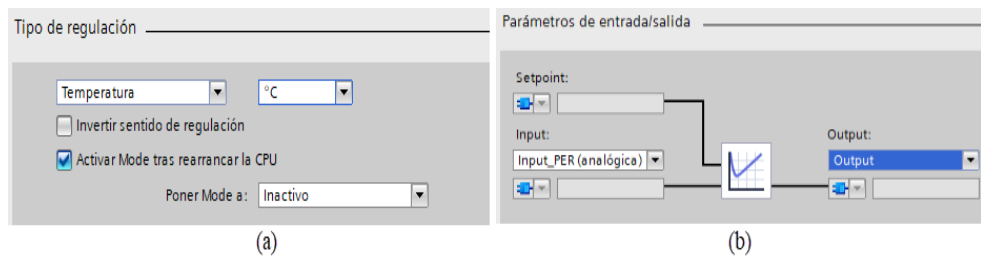
*Bloque de instrucción PID\_Compact con entradas y salidas previa a la configuración*



La configuración de PID\_Compact para aplicación en procesos industriales fue realizado desde la plataforma de interface del *Cliente del controlador PID autosintonizable*, se realizó una configuración inicial para una de las variables como es “Temperatura”, que iniciará en el PLC S7 1500 por default. Se realizó el ajuste básico del bloque de instrucción PID\_Compact para esta variable, siendo el parámetro de entrada tipo analógico (0 - 27648) y la salida regular (%) (ver Figura 36).

**Figura 36**

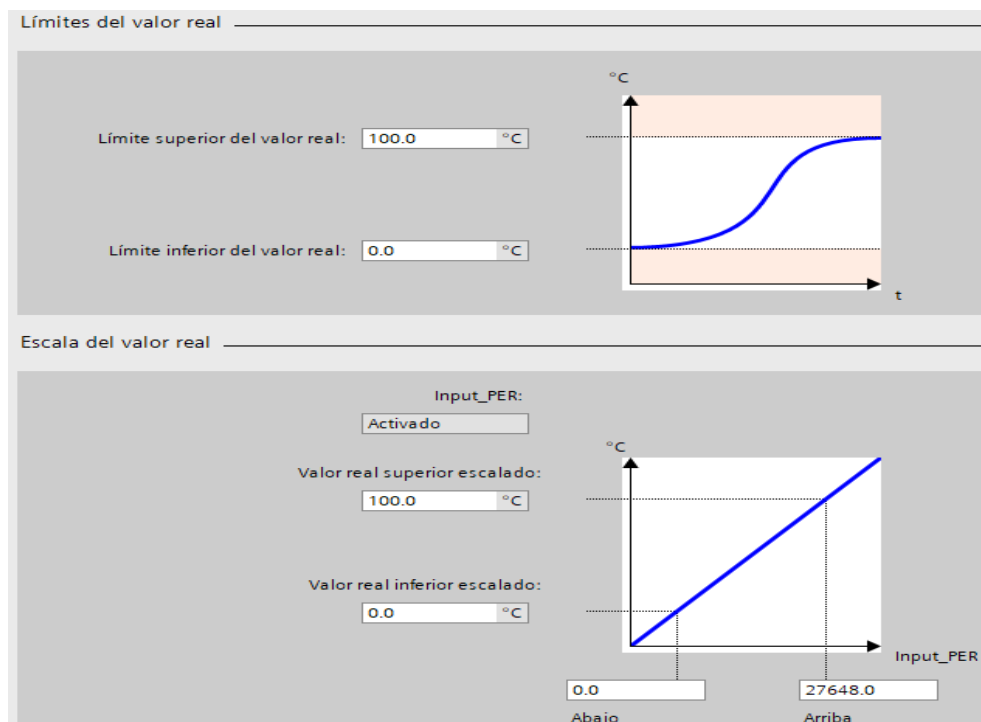
*Ajuste básico del bloque de instrucción PID\_Compact: (a) tipo de regulación “Temperatura” y (b) parámetros de entrada “analógica” y salida en “%”*



El ajuste de los límites para la variable “Temperatura” se da de 0 – 100 °C, la entrada analógica para la escala del valor real tiene el valor numérico que abarca de 0 – 27648, lo que se corresponde de forma directa con los límites de la variable (ver Figura 37).

**Figura 37**

*Ajustes de los límites y escala del valor real en el bloque de instrucción PID\_Compact*





En ajustes avanzados se activa entrada manual, lo que permitirá realizar la autosintonización desde la interface del cliente del controlador PID. Los datos de los parámetros PID no fue necesario modificar ( $K_p = 1.0$ ,  $T_i = 20.00$  s,  $T_d = 0.1$  s), esto sirvió como referencia para determinar las mejoras obtenidas con la autosintonización; además se seleccionó la estructura del regulador como: PID (ver Figura 38).

### Figura 38

*Ajustes de los límites y escala del valor real en el bloque de instrucción PID\_Compact*

Parámetros PID

☒ Activar entrada manual

Ganancia proporcional: 1.0

Tiempo de integración: 20.0 s

Tiempo derivativo: 0.1 s

Coeficiente retardo derivativo: 0.2

Ponderación de la acción P: 1.0

Ponderación de la acción D: 1.0

Tiempo muestreo algoritmo PID: 1.0 s

Regla para la optimización

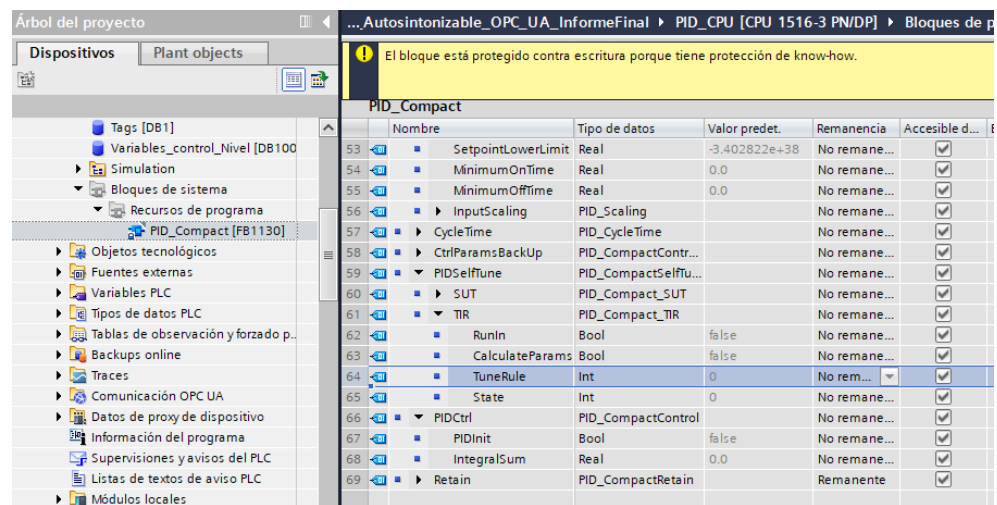
Estructura del regulador: PID

En el bloque protegido contra escritura con ruta: *Bloque de sistema-Recursos de programa-PID\_Compact* (FB1130), se encuentra la variable TuneRule (*PIDSelfTune-TIR-TuneRule*) que está destinado para seleccionar la regla de sintonización definida (ver Figura 39). Esta variable TuneRule toma valores de 0 – 5, lo que permite configurar diferentes tipos de sintonización: 0- la sintonización PID es automática, 1- la sintonización PID es rápida, 2- la sintonización PID es lenta, 3- la sintonización PID usa la regla Ziegler-Nichols, 4- la sintonización PI usa la regla Ziegler-Nichols y 5- la sintonización P usa la regla Ziegler-Nichols, esta configuración fue realizada desde la interface del cliente del controlador PID autosintonizable,

seleccionando para el estudio la opción 3 de Text\_List\_TuneRuleTIR (ver Figura 40).

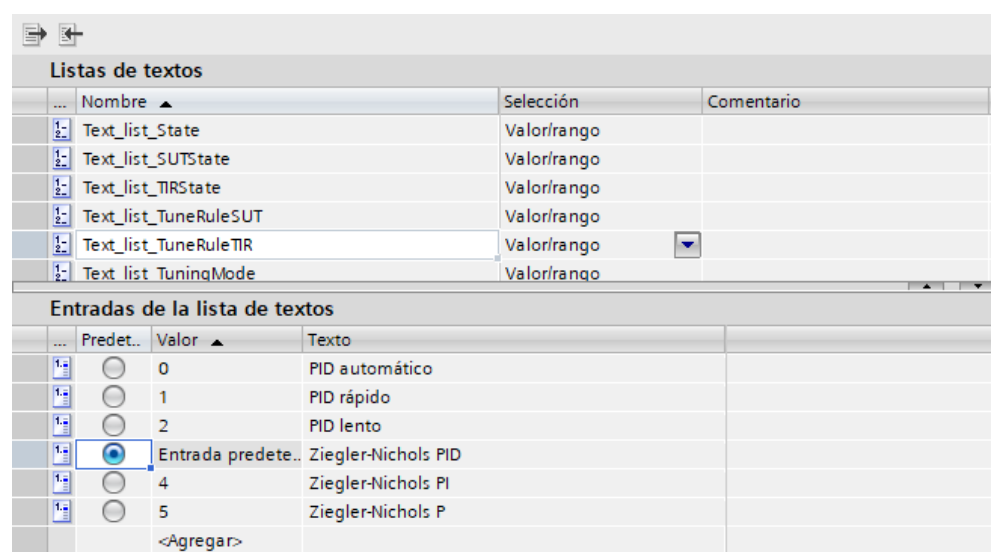
**Figura 39**

*Ruta para acceso al bloque protegido contra escritura TuneRule de PID\_Compact*



**Figura 40**

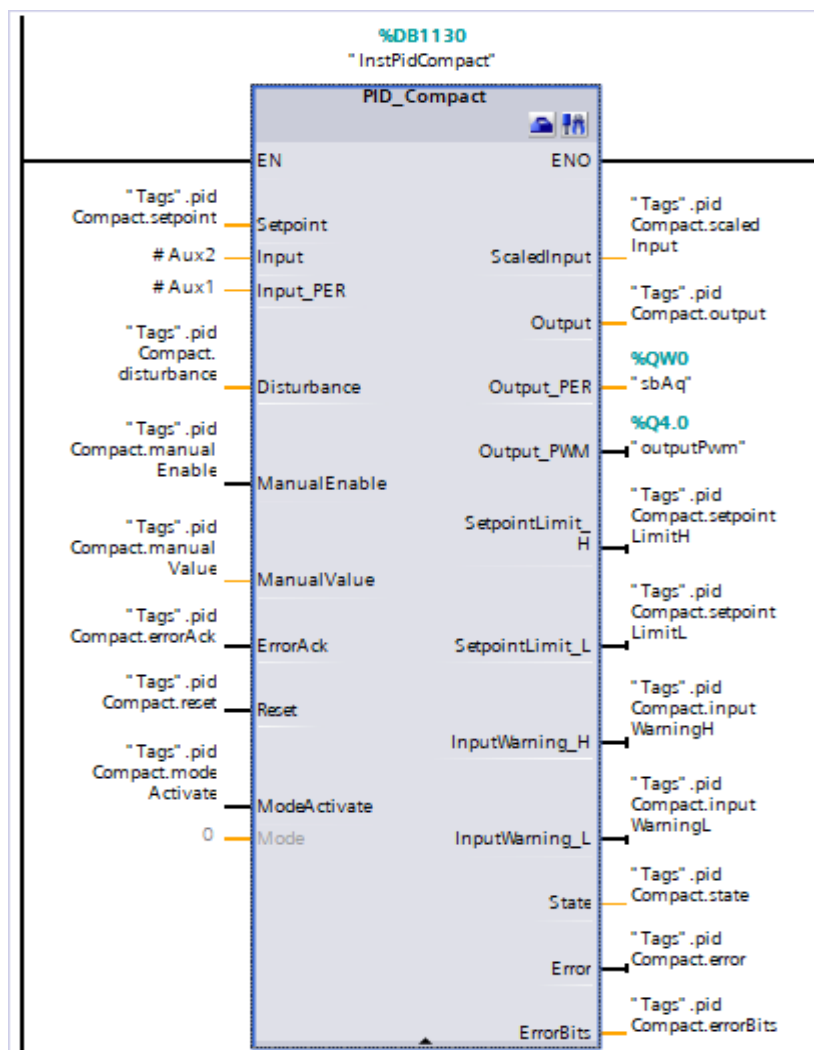
*Valores que toma TuneRuleTIR para configurar diferentes tipos de sintonización*



Una vez configurado el controlador PID\_Compact se asignó “Tags” tanto a las entradas como a las salidas (ver Figura 41).

**Figura 41**

*Bloque de instrucción PID\_Compact con entradas y salidas asignadas a “Tags”*

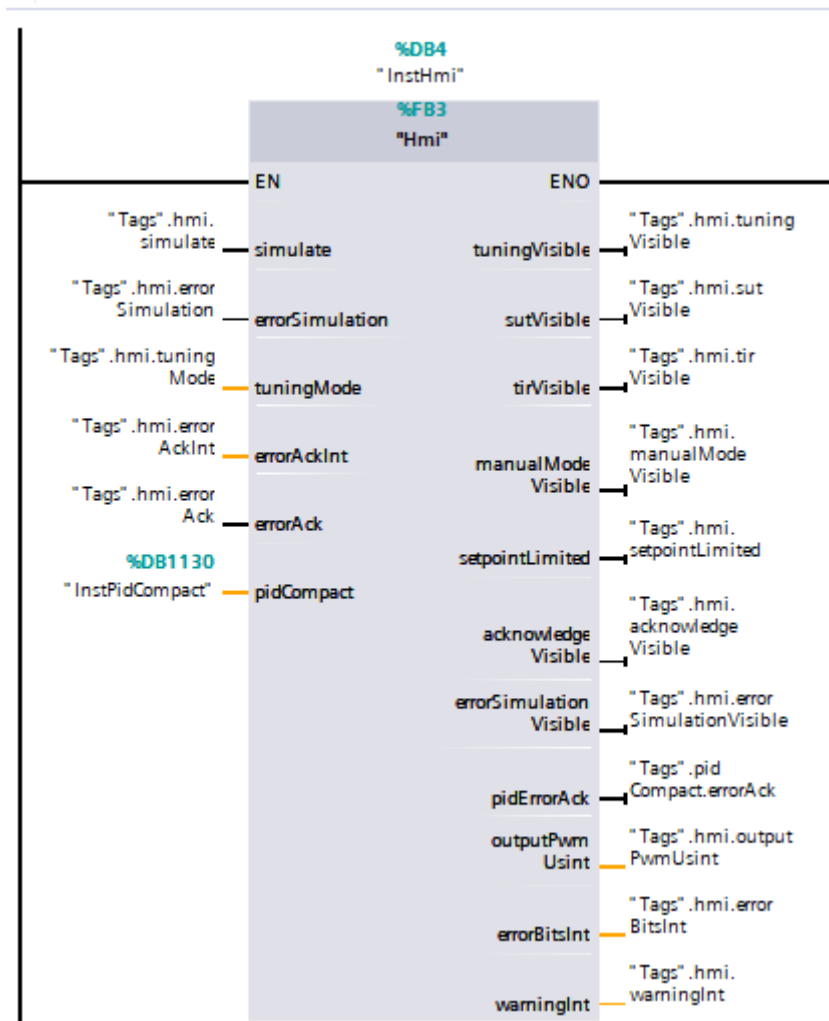


Para facilitar la conexión de las variables entre el servidor y cliente del controlador PID autosintonizable se tuvo que generar un bloque de funciones de interface “Hmi” (FB3) (ver Figura 42). Este bloque permite configurar y poner en marcha al controlador desde el cliente del controlador PID

autosintonizable; así mismo, visualizar el comportamiento de las variables controladas de los procesos.

**Figura 42**

*Tags de entradas y salidas del bloque de funciones “Hmi”*



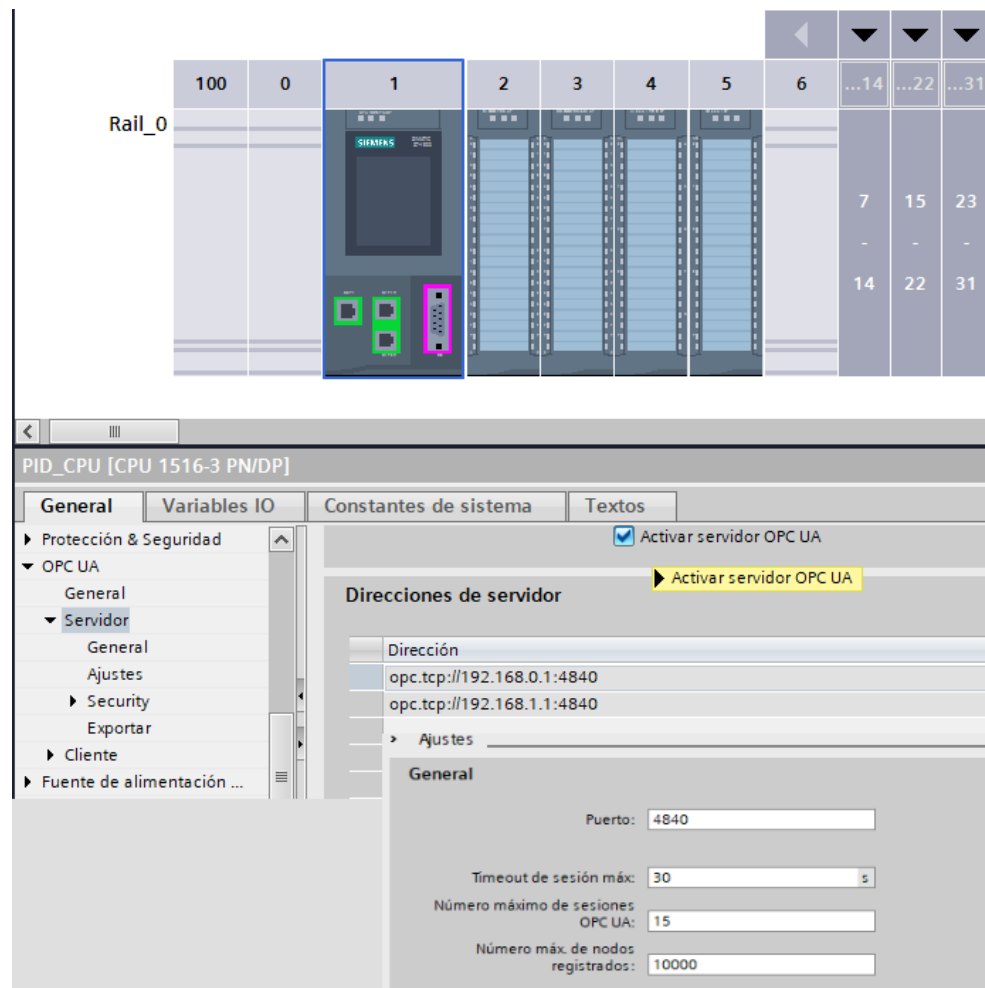
#### 4.1.1.3. Configuración del servidor para comunicación OPC UA

La comunicación del servidor OPC UA se configuró en el PLC S7 1500, siendo la dirección del servidor el generado en forma automática en base a la dirección ethernet asignada al PLC. La Figura 43 muestra que al seleccionar “Activar servidor OPC UA” se genera las direcciones del servidor, siendo estas `opc.tcp://192.168.0.1:4840` y `opc.tcp://192.168.1.1:4840`, donde la

comunicación es por el puerto 4840, con hasta 15 sesiones en simultáneo y 10000 nodos registrados.

**Figura 43**

*Configuración del servidor para comunicación OPC UA*

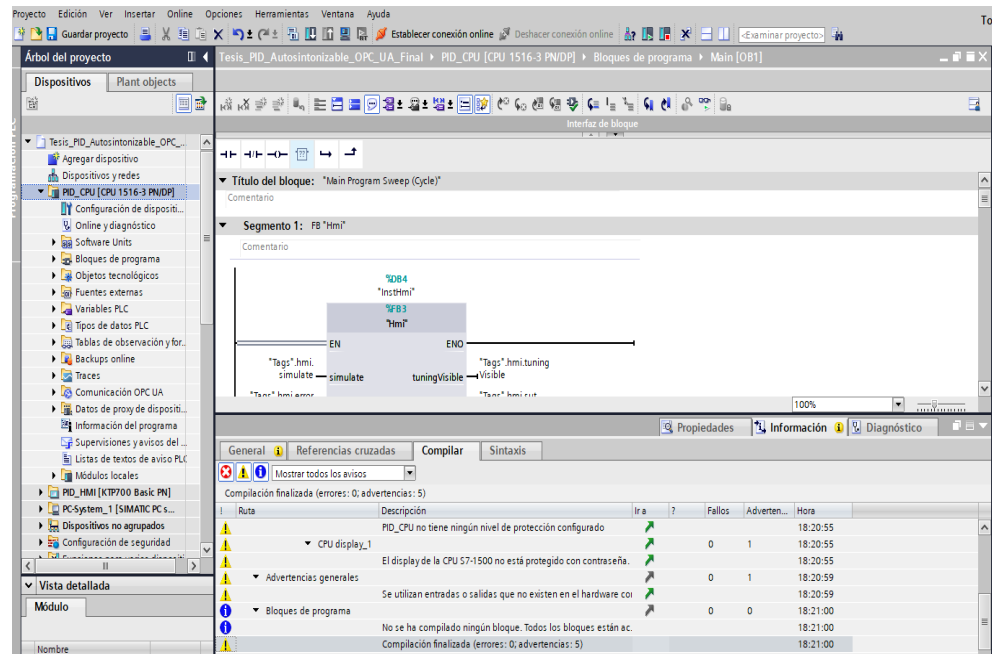


Luego, en TIA Portal, se procedió a compilar toda la estructura del hardware y el programa que corresponde al servidor PID autosintonizable, no se encontró error alguno (ver Figura 44), por lo que está listo para la carga en el PLC.

La Figura 45 presenta la carga de la configuración del hardware y el programa del servidor PID autosintonizable en el PLC S7 1500.

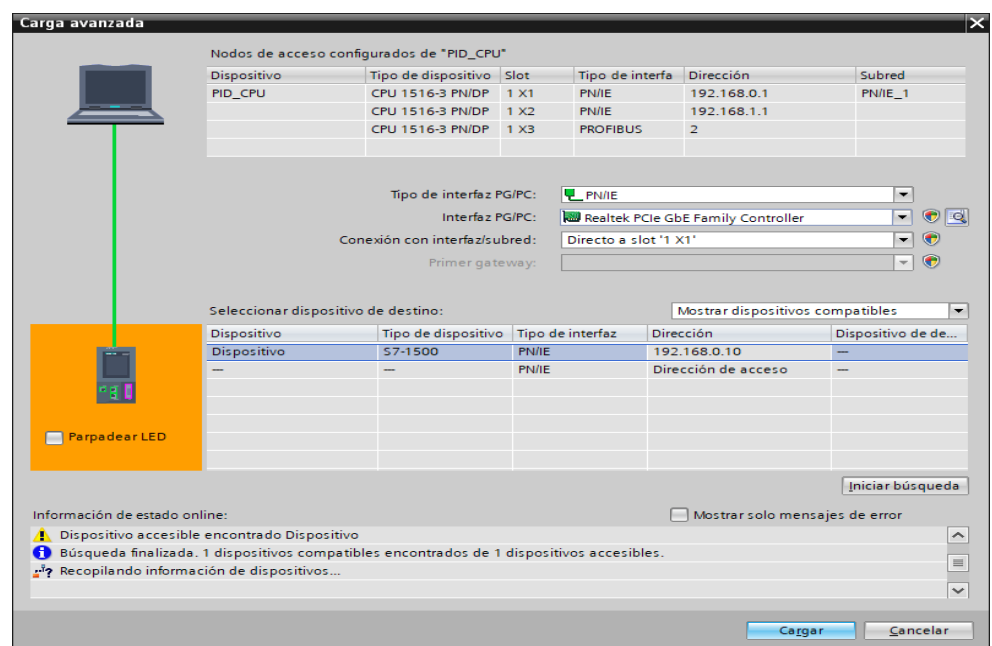
**Figura 44**

*Resultado de la compilación de la estructura del hardware y programa del servidor PID autosintonizable*



**Figura 45**

*Proceso de carga de la configuración del hardware y programa del servidor PID autosintonizable en PLC S7 1500*

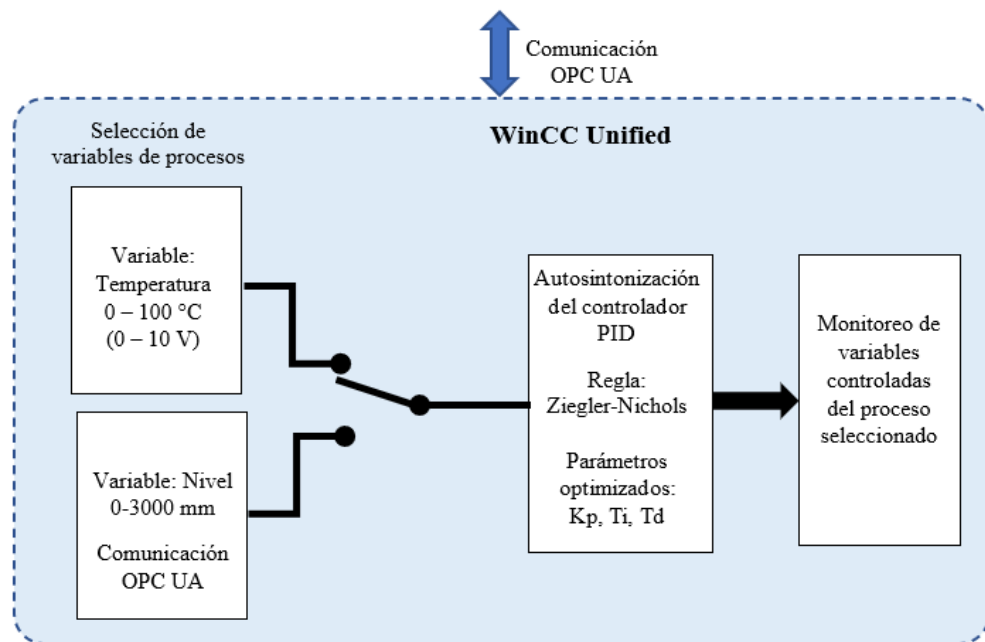


#### 4.1.2. Implementación del cliente del controlador PID Autosintonizable

El cliente del controlador PID autosintonizable se implementó en WinCC Unified V17.0; que mantiene comunicación por medio del protocolo OPC UA con el servidor (ver Figura 46). Permite seleccionar las variables controladas de los procesos para el ingreso al controlador PID configurado en PID\_Compact que es parte del servidor. Así mismo, desde esta interface es posible seleccionar la regla de Ziegler-Nichols, activar el proceso de autosintonización, visualizar los parámetros  $K_p$ ,  $T_i$ , y  $T_d$  optimizados. También es posible el monitoreo de las variables controladas del proceso seleccionado.

**Figura 46**

*Diagrama general del cliente del controlador PID Autosintonizable implementado en WinCC Unified*

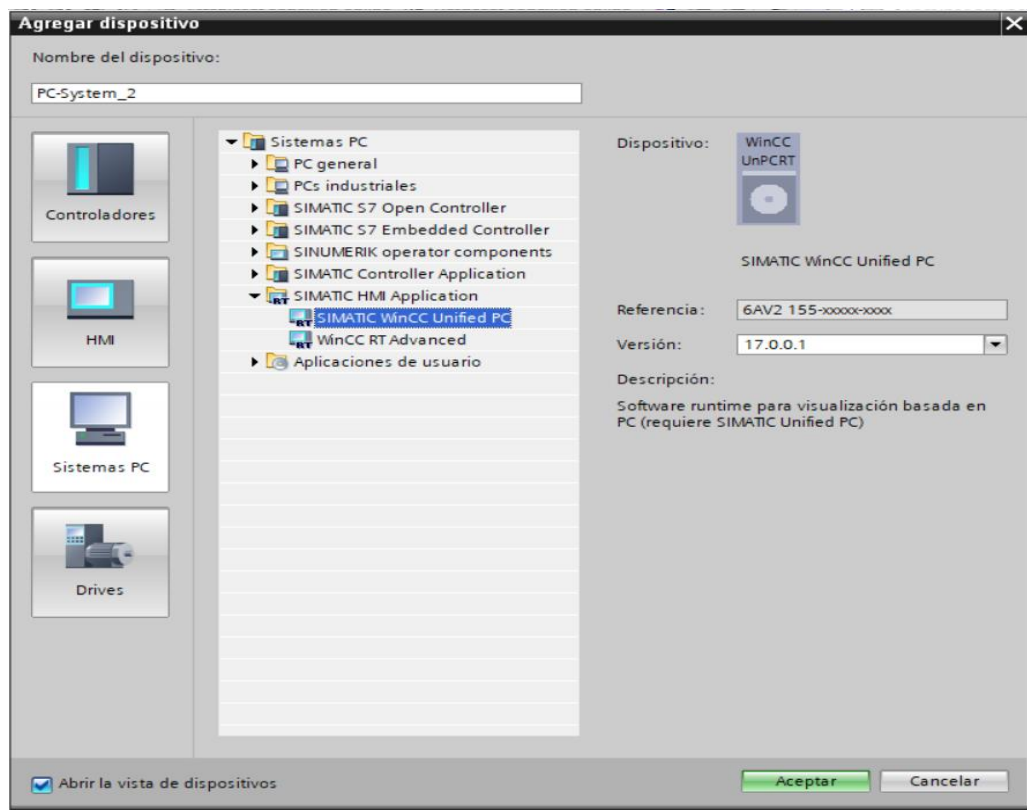


##### 4.1.2.1. Configuración de WinCC Unified

En TIA Portal, se creó como dispositivo un Sistema PC "PC-System", tal como muestra la Figura 47, seleccionando SIMATIC WinCC Unified PC.

**Figura 47**

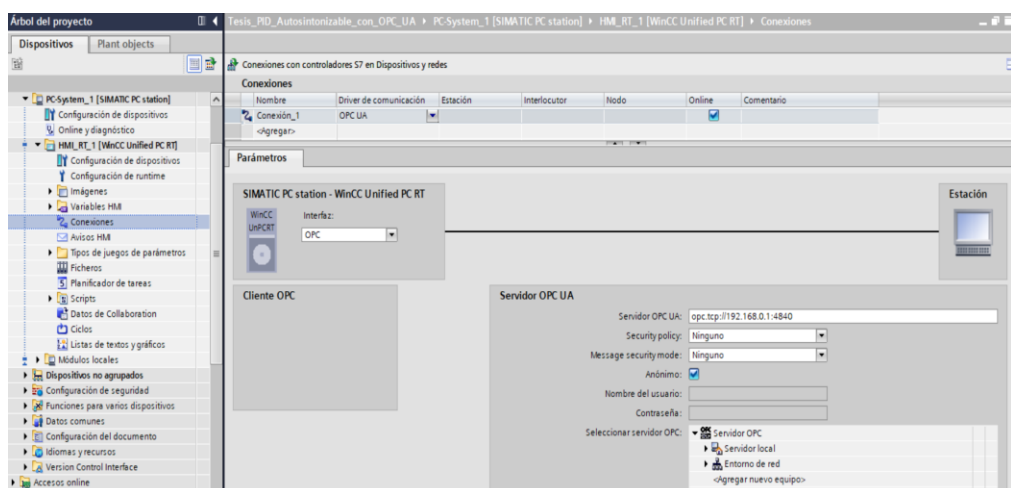
*Creación del dispositivo PC-System y selección de SIMATIC WinCC Unified*



En “Conexiones” se configuró el driver de comunicación entre el servidor OPC UA y WinCC Unified, tal como se muestra en la Figura 48.

**Figura 48**

*Configuración del driver de comunicaciones OPC UA*

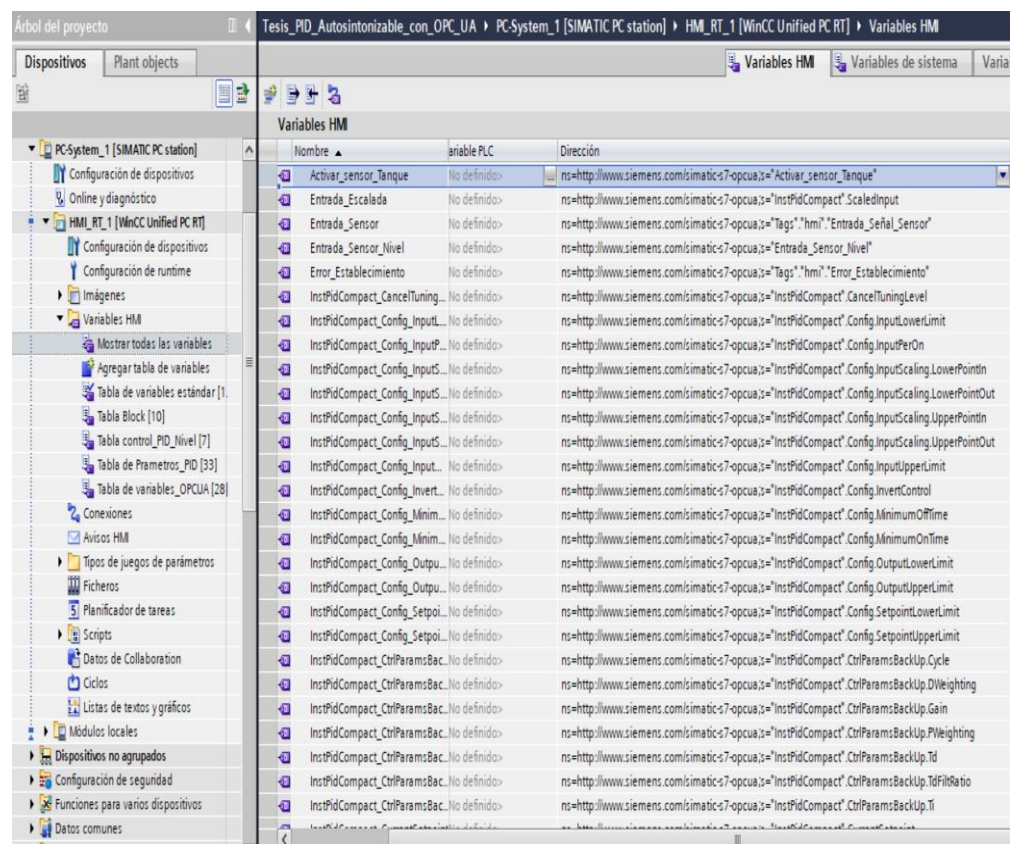




En seguida, se generaron las direcciones para comunicación OPC UA de las diferentes variables HMI (ver Figura 49); tanto para la configuración del controlador PID, como de las variables controladas *temperatura* de la planta térmica y del *nivel* de líquido en tanque.

**Figura 49**

*Direcciones generadas de las diferentes variables de interface HMI para comunicación OPC UA*

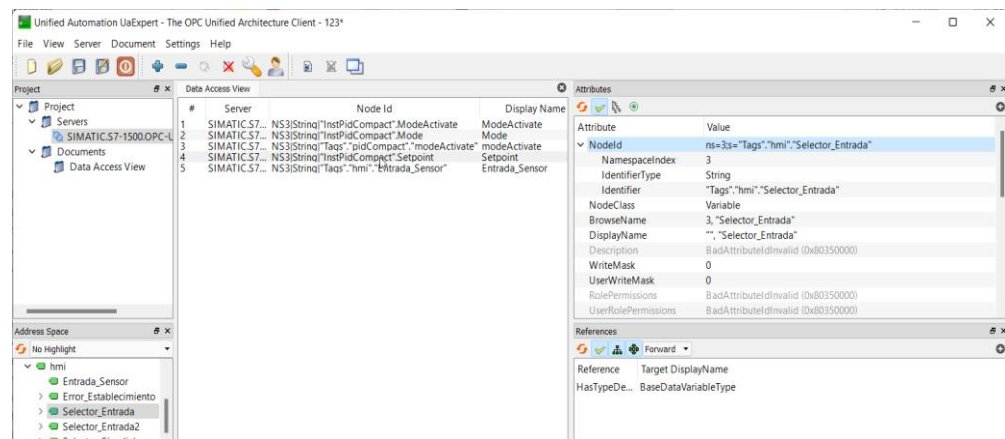


Nombre	Variable PLC	Dirección
Activar_sensor_Tanque	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="Activar_sensor_Tanque"
Entrada_Escalada	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".ScaledInput
Entrada_Sensor	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="Tags".hmi."Entrada_Señal_Sensor"
Entrada_Sensor_Nivel	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="Entrada_Sensor_Nivel"
Error_Establecimiento	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="Tags".hmi."Error_Establecimiento"
InstPidCompact_CancelTuning...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".CancelTuningLevel
InstPidCompact_Config_Input...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.InputLowerLimit
InstPidCompact_Config_Input...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.InputPerOn
InstPidCompact_Config_Input...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.InputScaling.LowerPointIn
InstPidCompact_Config_Input...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.InputScaling.LowerPointOut
InstPidCompact_Config_Input...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.InputScaling.UpperPointIn
InstPidCompact_Config_Input...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.InputScaling.UpperPointOut
InstPidCompact_Config_Input...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.InputUpperLimit
InstPidCompact_Config_Invert...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.InvertControl
InstPidCompact_Config_Minim...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.MinimumOffTime
InstPidCompact_Config_Minim...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.MinimumOnTime
InstPidCompact_Config_Outpu...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.OutputLowerLimit
InstPidCompact_Config_Outpu...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.OutputUpperLimit
InstPidCompact_Config_Setpoi...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.SetpointLowerLimit
InstPidCompact_Config_Setpoi...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".Config.SetpointUpperLimit
InstPidCompact_CtrlParamsBac...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".CtrlParamsBackUp.Cycle
InstPidCompact_CtrlParamsBac...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".CtrlParamsBackUp.DWeighting
InstPidCompact_CtrlParamsBac...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".CtrlParamsBackUp.Gain
InstPidCompact_CtrlParamsBac...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".CtrlParamsBackUp.Td
InstPidCompact_CtrlParamsBac...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".CtrlParamsBackUp.TdFiltRatio
InstPidCompact_CtrlParamsBac...	No definido	ns=http://www.siemens.com/simatic-7-opcua:s="InstPidCompact".CtrlParamsBackUp.Ti

Las direcciones de las variables HMI fueron verificadas con Unified Automation UaExpert para obtener el nodo y su identificador (NodeId) de cada variable o tags a configurar o escribir y monitorear (ver Figura 50).

**Figura 50**

*Obtención de nodos e identificador de variables con UaExpert*

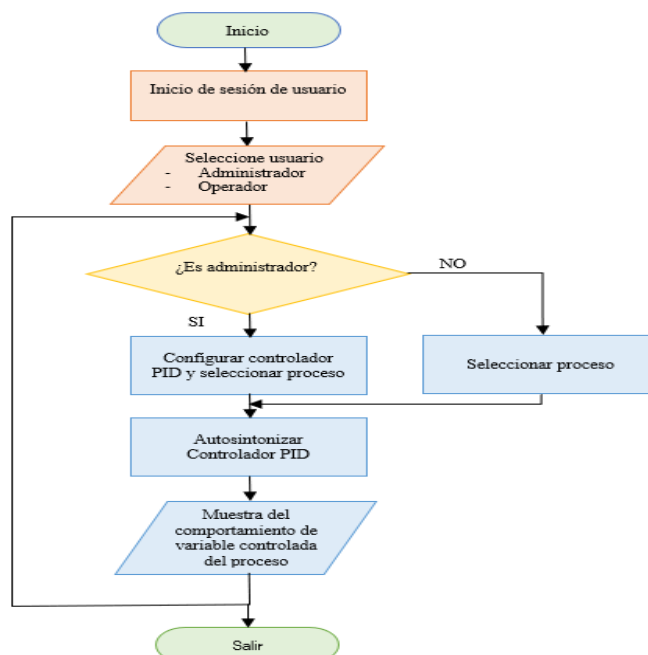


#### 4.1.2.2. Creación de imágenes en WinCC Unified

En la carpeta *Imágenes* ubicado dentro de la interface HMI\_RT\_1 (WinCC Unified PC RT), se crearon las diferentes imágenes cuyo funcionamiento global obedece al diagrama de flujo de la Figura 51.

**Figura 51**

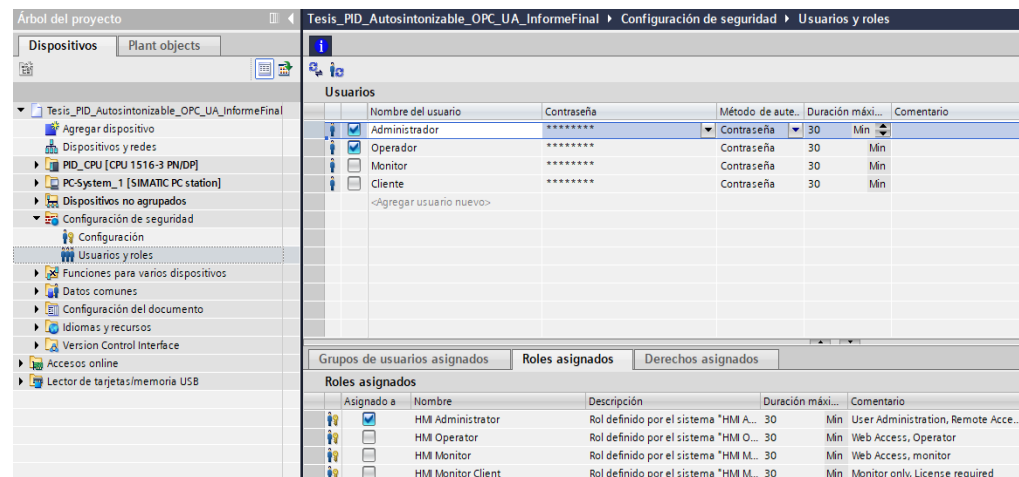
*Diagrama de flujo de ejecución de imágenes del cliente del controlador PID autosintonizable*



Para el inicio de sesión de usuario se crearon 02 usuarios siendo el primero el *Administrador* y el segundo el *Operador* (ver Figura 52). El administrador tiene privilegios para configurar el controlador y realizar todas las acciones incluidas acceso remoto, mientras que el operador se limita a seleccionar el proceso industrial, autosintonizar el controlador y monitorear el comportamiento de la variable controlada.

**Figura 52**

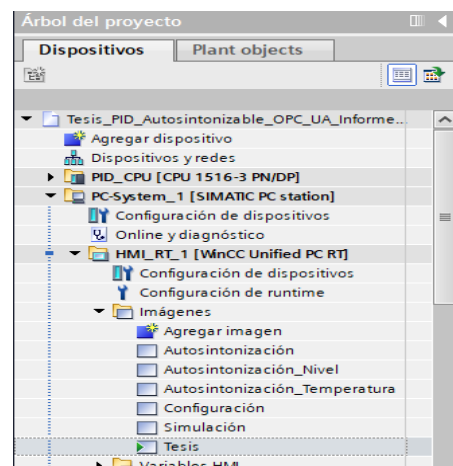
*Configuración del rol de los usuarios Administrador y Operador*



Luego de configurar el rol de los usuarios se procedió a crear 06 imágenes que se muestran en la Figura 53.

**Figura 53**

*Árbol de creación de imágenes en WinCC Unified*

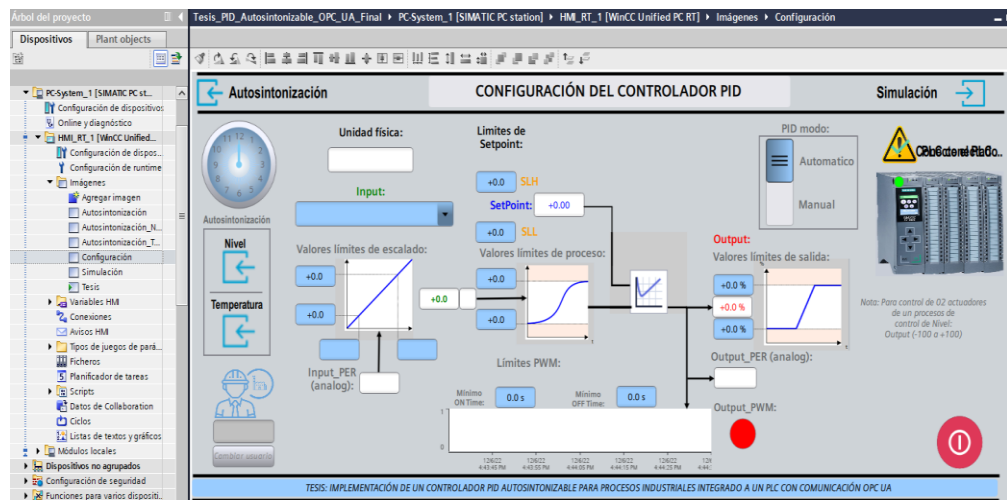





de las imágenes de autosintonización de la temperatura y nivel (ver Figura 55). Así mismo informa el estado de conexión/desconexión del PLC.

**Figura 55**

*Componentes de la imagen “Configuración”*



### ***c. Implementación de la imagen “Autosintonización”***

La imagen “Autosintonización” está implementada para autosintonizar procesos industriales teniendo como señales de entrada de variables controladas en porcentaje (0 – 100%) o del tipo analógico (0 - 27648). Así mismo el setpoint es configurable en tiempo real. La salida y el error de estado estable se da en porcentaje. Así mismo, esta imagen está implementada con un visor de curvas que permite ver el comportamiento de *Setpoint*, *Entrada*, *Salida* y *Error*. El proceso de autosintonización fue configurado para iniciar al pulsar el botón sintonización de la imagen (  ) con el botón izquierdo del mouse, siguiendo la siguiente secuencia:

1. Guardar el estado actual del controlador para volver si se cancela la sintonización con: `Tags_hmi_stateSaved := Tags_pidCompact_state` (0=Inactivo, 1=SUT, 2=TIR, 3=Automático, 4=Manual)
2. Guardar el modo de sintonización con: `InstPidCompact_Mode := Tags_hmi_tuningMode` (1=SUT / 2=TIR))

- Se activa el bit 0 de la variable Tags\_pidCompact\_modeActivate para concretar las secuencias 1 y 2. (ver Figura 56)

**Figura 56**



*Configuración del evento para sintonización al pulsar el botón respectivo*

Activado	Nombre	Valor
Desactivado	▼ DefinirValorVariable	
Clic con botón izquierd...	Variable	Tags_hmi_stateSaved
Pulsar tecla	Valor	Tags_pidCompact_state
Soltar tecla	▼ DefinirValorVariable	
 Pulsar	Variable	InstPidCompact_Mode
 Soltar	Valor	Tags_hmi_tuningMode
Clic con botón derec...	▼ ActivarBitEnVariable	
	Variable	Tags_pidCompact_modeActivate
	Número de bit	0
	<Agregar función>	

Y al soltar el botón izquierdo del mouse se desactiva el bit 0 de la variable Tags\_pidCompact\_modeActivate (ver Figura 57), iniciando de esta manera el proceso de sintonización.

**Figura 57**

*Configuración del evento para sintonización al soltar el botón respectivo*

Activado	Nombre	Valor
Desactivado	▼ DesactivarBitEnVariable	
Clic con botón izquierd...	Variable	Tags_pidCompact_modeActivate
Pulsar tecla	Número de bit	0
Soltar tecla	<Agregar función>	
 Pulsar		
 Soltar		
Clic con botón derec...		

La imagen “Autosintonización” completa se presenta en la Figura 58, la autosintonización de una variable controlada de un proceso industrial inicia con fijar el Setpoint, para luego establecer el modo en sintonía fina con el método Ziegler-Nichols basado en la ganancia crítica  $K_{cr}$  y periodo crítico  $P_{cr}$  (ver Tabla 13); y pulsando el botón “Sintonización” se obtiene los valores adecuados de  $K_p$ ,  $T_i$  y  $T_d$  del módulo PID\_Compact para un proceso en específico.

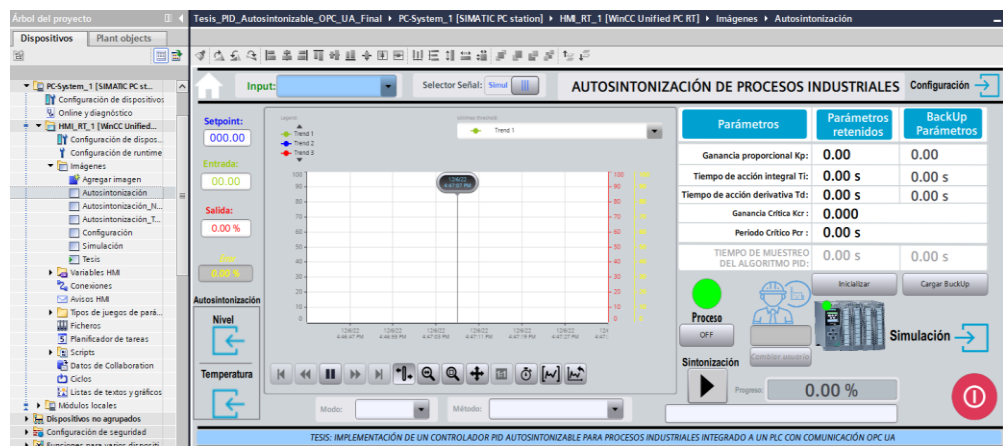
**Tabla 13**

*Método de sintonización PID de Ziegler-Nichols basada en la ganancia crítica  $K_{cr}$  y periodo crítico  $P_{cr}$*

Tipo de controlador	$K_p$	$T_i$	$T_d$
PID	$0.6K_{cr}$	$0.5P_{cr}$	$0.125P_{cr}$

**Figura 58**




*Componentes de la imagen “Autosintonización”*



#### ***d. Implementación de la imagen “Autosintonización\_Temperatura”***

A parte de la imagen “Autosintonización” que opera de manera genérica se implementó la imagen “Autosintonización\_Temperatura” que se muestra en la Figura 59; permite encontrar los valores adecuados para los parámetros  $K_p$ ,  $T_i$  y  $T_d$ , de la variable temperatura en la planta térmica de rango de temperatura ambiente a 100 °C, previa selección del modo sintonía fina con el método Ziegler-Nichols basado en la ganancia crítica  $K_{cr}$  y periodo crítico  $P_{cr}$ ; para esta sintonización se sigue la secuencia 1, 2 y 3 de la sección “c”.

Antes de sintonizar, se debe conectar la entrada analógica correspondiente a la

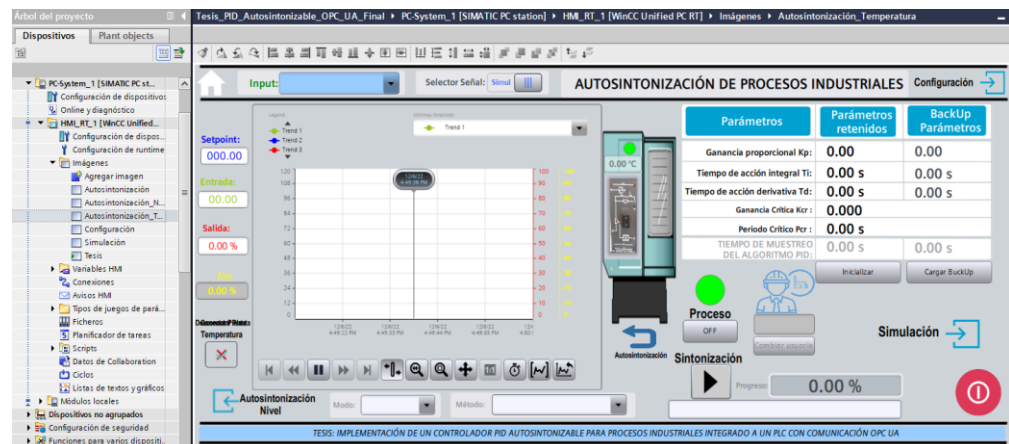
planta de proceso térmico (); en seguida, se establece el Setpoint de la temperatura () y se procede con la sintonización () , durando este proceso un determinado tiempo, al cabo del cual se obtienen los valores



adecuados de  $K_p$ ,  $T_i$  y  $T_d$  en el módulo PID\_Compact ubicado en el servidor PID.

**Figura 59**

*Componentes de la imagen “Autosintonización\_Temperatura”*

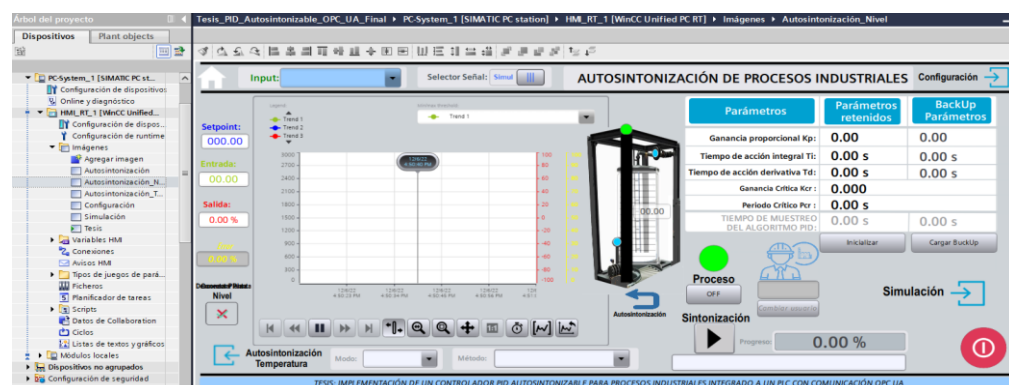


**e. Implementación de la imagen “Autosintonización\_Nivel”**

La imagen “Autosintonización\_Nivel” que se muestra en la Figura 60; permite encontrar los valores adecuados para los parámetros  $K_p$ ,  $T_i$  y  $T_d$ , para el control PID de la variable nivel de líquido en tanque de altura máxima de 3000 mm, previa selección del modo sintonía fina con el método Ziegler-Nichols; y pulsando el botón “Sintonización” (que sigue la secuencia 1, 2 y 3 de la sección “c”).

**Figura 60**

*Componentes de la imagen “Autosintonización\_Nivel”*















#### 4.1.3. Determinación del tiempo de asentamiento de la variable *Temperatura* en un proceso térmico

El resultado de la respuesta del controlador PID no sintonizado a entradas setpoint escalón, con parámetros preestablecidos:  $K_p = 1.0$ ,  $T_i = 20.00$  s,  $T_d = 0.1$  s para  $e_{ss} \leq 2\%$ , para la variable temperatura del proceso en la planta térmica es del tipo sobre amortiguado; siendo los valores de sobrepaso máximo ( $M_p$  %) negativos que asciende hasta  $-1.99\%$  y un tiempo de asentamiento máximo de 193 s (de la Tabla 14).

**Tabla 14**

*Resultados de los experimentos con controlador PID no sintonizado para la variable Temperatura de la planta térmica*

N° Experimento	Setpoint (°C)	Tiempo de asentamiento (s)	Tiempo de asentamiento normalizado (%)	Sobrepaso máximo ( $M_p$ %)
1	12 	109	56.48	-2.00
2	12 	146	75.65	-1.99
3	12 	166	86.01	-1.99
4	12 	169	87.56	-1.98
5	12 	173	89.64	-2.00
6	12 	176	91.19	-1.97
7	12 	177	91.71	-1.98
8	12 	179	92.75	-14.20
9	12 	184	95.34	-2.00
10	12 	193	100.00	-1.99

*Nota.* Valores obtenidos con parámetros PID preestablecidos:  $K_p = 1.0$ ,  $T_i = 20.00$  s,  $T_d = 0.1$  s, para  $e_{ss} \leq 2\%$ . Planta térmica con temperatura de proceso de temperatura ambiente hasta 96 °C. Datos obtenidos de las muestras del Apéndice A.

La Figura 61 muestra la curva de color verde, que es la temperatura controlada para un setpoint de 50 °C; en esta figura se evidencia el tipo de respuesta sobre amortiguada del controlador PID.

**Figura 61**

*Respuesta sobre amortiguada del controlador PID no sintonizado para la variable Temperatura de la planta térmica*



Al autosintonizar el controlador PID por el método de sintonización PID de Ziegler-Nichols basada en la ganancia crítica  $K_{cr}$  y periodo crítico  $P_{cr}$ , para un setpoint a 50 °C de la variable controlada *Temperatura*, dio como resultado los parámetros  $K_p$ ,  $T_i$  y  $T_d$  que se presenta en la Tabla 15, con las cuales se determinó el tiempo de asentamiento para diferentes valores de setpoint de esta variable controlada.

**Tabla 15**











*Parámetros PID obtenidos con el método de sintonización PID de Ziegler-Nichols basada en la ganancia crítica  $K_{cr}$  y periodo crítico  $P_{cr}$ , para la variable controlada Temperatura de la planta térmica*

Tipo de controlador	$K_p$	$T_i$	$T_d$
PID	6.03	0.74 s	0.18 s

Así mismo, el controlador PID autosintonizado da respuesta a entradas setpoint escalón con parámetros PID de la Tabla 14 para  $e_{ss} \leq 2\%$  para la variable *Temperatura* del proceso en la planta térmica, obteniendo una respuesta del tipo sub amortiguado; siendo los valores de sobrepaso máximo (Mp %) descendientes de 8.97% a 0.13% con un tiempo de asentamiento máximo de 54 s (ver Tabla 16).

**Tabla 16**

*Resultados de los experimentos con controlador PID autosintonizado para la variable Temperatura de la planta térmica*

Nº Experimento	Setpoint (°C)	Tiempo de asentamiento (s)	Tiempo de asentamiento normalizado (%)	Sobrepaso máximo (Mp %)
1	12 	24	12.44	7.57
2	12 	27	13.99	8.97
3	12 	29	15.03	8.12
4	12 	31	16.06	7.64
5	12 	34	17.62	5.08
6	12 	37	19.17	3.33
7	12 	39	20.21	2.27
8	12 	42	21.76	1.45
9	12 	49	25.39	0.63
10	12 	54	27.98	0.13

*Nota.* Valores obtenidos con parámetros PID autosintonizado:  $K_p= 6.03$ ,  $T_i= 0.74$  s,  $T_d=0.18$  s, para  $e_{ss} \leq 2\%$ . Planta térmica con temperatura de proceso de temperatura ambiente hasta 96 °C. Datos obtenidos de las muestras del Apéndice B.

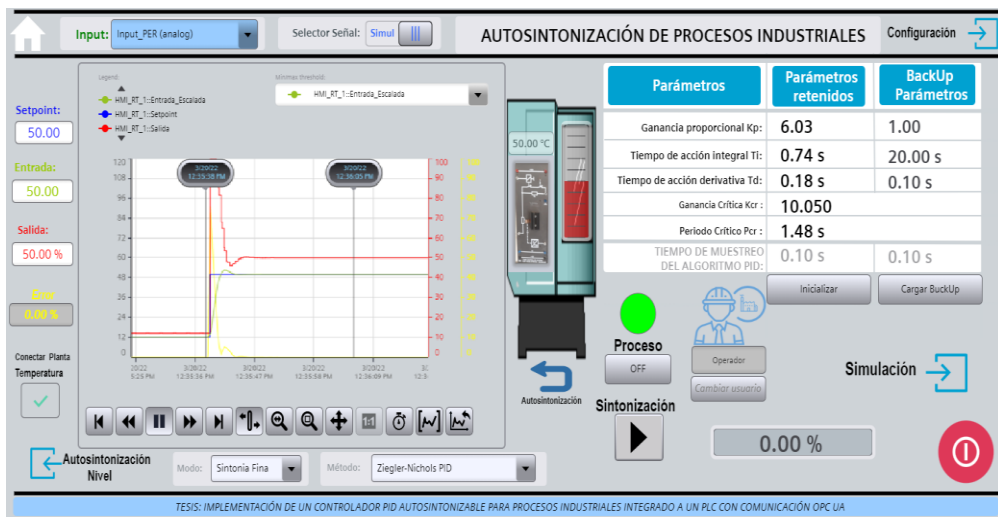
Los datos que contiene la Tabla 16 se obtuvieron de las muestras presentadas en las tablas de Apéndice B, el tiempo de asentamiento máximo (54 s) ocurre en el experimento 10; este valor representa el 27.97 % del máximo encontrado

en el controlador PID no sintonizados para la variable *Temperatura* de la planta térmica mostrado en la Tabla 14.

La Figura 62 muestra la curva de color verde, que es la *Temperatura* controlada para un setpoint de 50 °C; en esta figura se evidencia el tipo de respuesta sub amortiguada del controlador PID.

**Figura 62**

*Respuesta sub amortiguada del controlador PID autosintonizado para la variable Temperatura de la planta térmica*

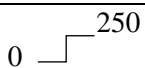
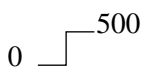
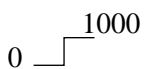
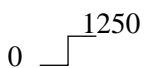
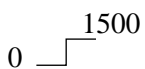
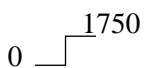
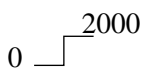
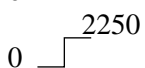
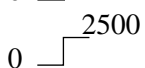
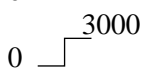


#### 4.1.4. Determinación del tiempo de asentamiento de la variable *Nivel* de líquido en tanque

El controlador PID no sintonizado da respuesta, a entradas setpoint escalón con parámetros:  $K_p = 1.0$ ,  $T_i = 20.00$  s,  $T_d = 0.1$  s y  $e_{ss} \leq 2\%$  para la variable nivel de líquido en tanque, del tipo sub amortiguado; siendo los valores de sobrepaso máximo ( $M_p$  %) descendientes de 19.60 % a 0.33% con un tiempo de asentamiento máximo de 70 s (de la Tabla 17).

**Tabla 17**

*Resultados de los experimentos con controlador PID no sintonizado para la variable Nivel de líquido en tanque*

N° Experimento	Setpoint (mm)	Tiempo de asentamiento (s)	Tiempo de asentamiento normalizado (%)	Sobrepaso máximo (Mp %)
1	0 	35	50.00	18.01
2	0 	41	58.57	19.60
3	0 	43	61.43	6.11
4	0 	44	62.86	5.18
5	0 	47	67.14	8.15
6	0 	52	74.29	6.89
7	0 	56	80.00	4.97
8	0 	59	84.29	2.68
9	0 	63	90.00	3.62
10	0 	70	100.00	0.33

*Nota.* Valores obtenidos con parámetros PID preestablecidos:  $K_p=1.0$ ,  $T_i=20.00$  s,  $T_d=0.1$  s, para  $\text{ess} \leq 2\%$ . Tanque de nivel máximo 3000 mm, flujo de electroválvulas de llenado y descarga 60 l/s. Datos obtenidos de las muestras del Apéndice C.

Los datos que contiene la Tabla 17 se obtuvieron de las muestras presentadas en las tablas del Apéndice C, el tiempo de asentamiento máximo (70 s) de la variable *Nivel* de líquido en tanque, ocurre en el experimento 10; este valor se considera el máximo para la normalización (100 %).

La Figura 63 muestra la curva de color verde, que es el nivel controlado para un setpoint de 1500 mm; en esta figura se evidencia el tipo de respuesta ligeramente sub amortiguada del controlador PID.

**Figura 63**

*Respuesta sub amortiguada del controlador PID no sintonizado para la variable Nivel de líquido en tanque*



Al autosintonizar el controlador PID por el método de sintonización PID de Ziegler-Nichols basada en la ganancia crítica  $K_{cr}$  y periodo crítico  $P_{cr}$ , para un setpoint a 1500 mm de la variable controlada *Nivel* de líquido en tanque, dio como resultado los parámetros  $K_p$ ,  $T_i$  y  $T_d$  que se presenta en la Tabla 18, con las cuales se determinó el tiempo de asentamiento para diferentes valores de setpoint de esta variable controlada.

**Tabla 18**











*Parámetros PID obtenidos con el método de sintonización PID de Ziegler-Nichols basada en la ganancia crítica  $K_{cr}$  y periodo crítico  $P_{cr}$ , para la variable controlada Nivel de líquido en tanque*

Tipo de controlador	$K_p$	$T_i$	$T_d$
PID	0.80	2.74 s	0.68 s

Finalmente, el controlador PID autosintonizado da respuesta a entradas setpoint escalón con parámetros PID de la Tabla 18 para  $e_{ss} \leq 2\%$  para la variable nivel de líquido en tanque, obteniendo una respuesta del tipo sobre amortiguado; siendo los valores de sobrepaso máximo ( $M_p$  %) descendientes de 25.85 % a 0.50% con un tiempo de asentamiento máximo de 42 s (de la Tabla 19).

**Tabla 19**

*Resultados de los experimentos con controlador PID autosintonizado para la variable Nivel de líquido en tanque*

Nº Experimento	Setpoint (mm)	Tiempo de asentamiento (s)	Tiempo de asentamiento normalizado (%)	Sobrepaso máximo ( $M_p$ %)
1	0 	25	35.71	20.00
2	0 	26	37.14	25.85
3	0 	26	37.14	10.68
4	0 	26	37.14	16.56
5	0 	32	45.71	12.82
6	0 	36	51.43	9.56
7	0 	38	54.29	7.54
8	0 	38	54.29	6.61
9	0 	38	54.29	4.87
10	0 	42	60.00	0.50

*Nota.* Valores obtenidos con parámetros PID autosintonizado:  $K_p=0.80$ ,  $T_i=2.74$  s,  $T_d=0.68$  s, para  $e_{ss} \leq 2\%$ . Tanque de nivel máximo 3000 mm, flujo de electroválvulas de llenado y descarga 60 l/s. Datos obtenidos de las muestras del Apéndice D.

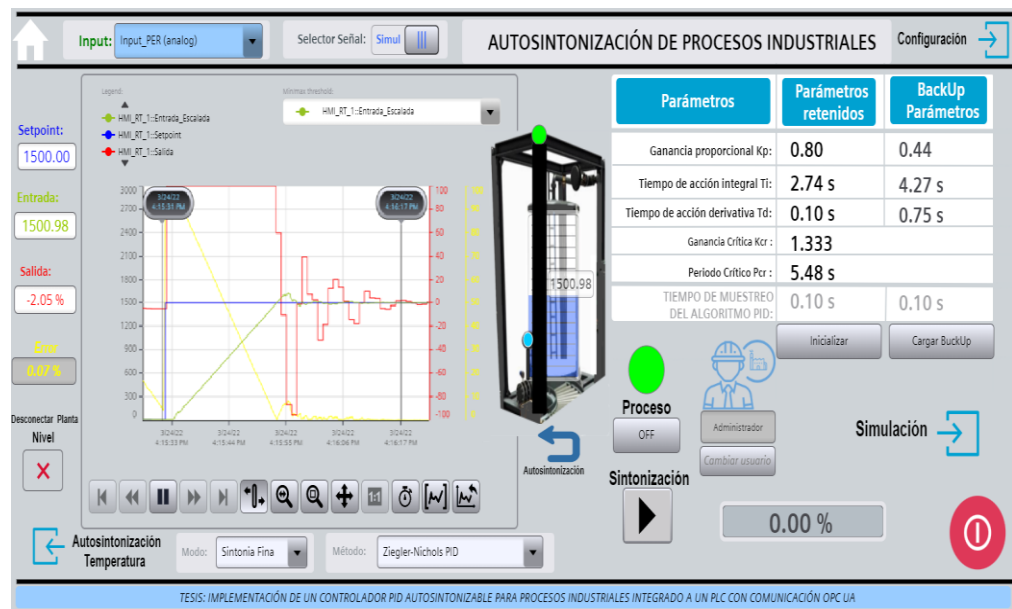
En la Tabla 19, los datos que contiene se obtuvieron de las muestras presentadas en las tablas de Apéndice D, el tiempo de asentamiento máximo de la variable nivel es 42 s, ocurre en el experimento 10; este valor representa

el 60% del máximo normalizado del tiempo de asentamiento del controlador PID no sintonizado.

La Figura 64 muestra la curva de color verde, que es el nivel controlado para un setpoint de 1500 mm; en esta figura se evidencia el tipo de respuesta sobre amortiguada del controlador PID.

**Figura 64**

*Respuesta sub amortiguada del controlador PID autosintonizado para la variable Nivel de líquido en tanque*



## 4.2. PRUEBA DE HIPÓTESIS

### 4.2.1. Prueba de las hipótesis específicas

#### 4.2.1.1. Prueba de la hipótesis específica a

**Hipótesis específica a.** Es factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un



cliente desarrollado en WinCC Unified, para la reducción en más del 60% del tiempo de asentamiento de la variable temperatura en un proceso industrial térmico para un  $\text{ess} \leq 2\%$ .

Las hipótesis estadísticas de trabajo planteados para la prueba de la hipótesis específica son:

**Hipótesis nula  $H_0$ :** Es factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción máxima en 60% del tiempo de asentamiento de la variable temperatura en un proceso industrial térmico para un  $\text{ess} \leq 2\%$ .

$$H_0: \mu_d \leq 60\%$$











**Hipótesis alternativa  $H_a$ :** Es factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción en más del 60% del tiempo de asentamiento de la variable temperatura en un proceso industrial térmico para un  $\text{ess} \leq 2\%$ .

$$H_a: \mu_d > 60\%$$

La Tabla 20 presenta los resultados de los experimentos con controlador PID no sintonizado y autosintonizado para diferentes setpoint de la variable Temperatura de la planta térmica.

**Tabla 20**

*Resultados de los experimentos con controlador PID no sintonizado y autosintonizado para la variable Temperatura de la planta térmica*

N° Experimento	Setpoint (°C)	Tiempo de asentamiento normalizado (%)	
		Con controlador PID no sintonizado	Con controlador PID autosintonizado
1	12  25	56.48	12.44
2	12  30	75.65	13.99
3	12  35	86.01	15.03
4	12  40	87.56	16.06
5	12  50	89.64	17.62
6	12  60	91.19	19.17
7	12  70	91.71	20.21
8	12  80	92.75	21.76
9	12  90	95.34	25.39
10	12  96	100.00	27.98

**Nota:** Los valores de tiempos de asentamiento se obtuvieron de la Tabla 14 y Tabla 16.

Por tratarse de grupos de muestras dependientes se empleó la prueba t de Student unilateral con cola a la derecha presentada en la ecuación 10.

$$t = \frac{\bar{d} - \mu_d}{S_d/\sqrt{n}} = 2.734 \quad (10)$$

Siendo la desviación estándar ( $S_d$ ), calculada con la ecuación 11; en base al cuadrado de la diferencia ( $d_i$ ) y la media de las diferencias ( $\bar{d}$ ) y el grado de libertad ( $n - 1$ ).

$$S_d = \sqrt{\frac{(d_i - \bar{d})^2}{n - 1}} = 8.868 \quad (11)$$

Con:  $n = 10$ ,  $S_d = 8.868$ ,  $\bar{d} = 67.668$ ,  $\mu_d = 60\%$ ; se calculó el valor del estadístico  $t = 2.734$ . Adicionalmente, la diferencia de medias, el valor crítico de  $t$ , los grados de libertad (GL) y valor-p unilateral para un intervalo de confianza del 95% se presentan en la Tabla 21.

**Tabla 21**

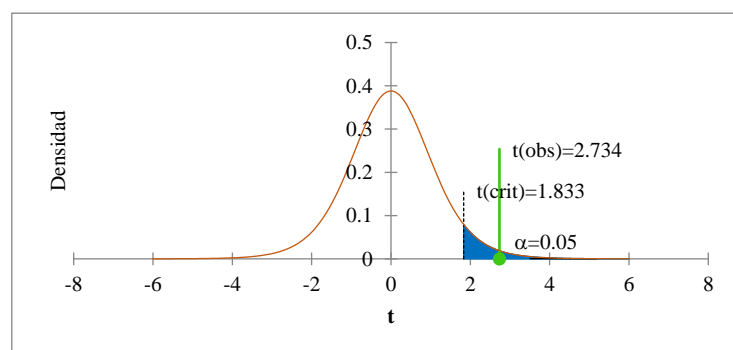
*Prueba  $t$  unilateral a la derecha para dos muestras dependientes de la variable Temperatura en planta térmica, para un intervalo de confianza del 95%*

Diferencia	67.668
t (Valor observado)	2.734
t (Valor crítico)	1.833
GL	9
valor-p (unilateral)	0.012
alfa	0.05

La representación gráfica de esta prueba se ve en la Figura 65.

**Figura 65**

*Representación gráfica de la prueba  $t$  de Student unilateral a la derecha para dos muestras dependientes de la variable Temperatura en planta térmica, para un intervalo de confianza del 95%*



**Interpretación de la prueba para la Hipótesis específica a.** Puesto que el valor-p obtenido = 0.012, es menor que el nivel de significancia  $\alpha=0.05$ , se debe rechazar la hipótesis nula  $H_0$  y aceptar la hipótesis alternativa  $H_a$ .

#### **4.2.1.2. Prueba de la hipótesis específica b**

**Hipótesis específica b.** Es factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción en más del 20% del tiempo de asentamiento de la variable nivel de líquido en tanque para un  $ess \leq 2\%$ .

Las hipótesis estadísticas de trabajo planteados para la prueba de la hipótesis específica b son:

**Hipótesis nula  $H_0$ :** Es factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción máxima en 20% del tiempo de asentamiento de la variable nivel de líquido en tanque para un  $ess \leq 2\%$ .

$$H_0: \mu_d \leq 20\%$$











**Hipótesis alternativa  $H_a$ :** Es factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción en más del 20% del tiempo de asentamiento de la variable nivel de líquido en tanque para un  $ess \leq 2\%$ .

$$H_a: \mu_d > 20\%$$

La Tabla 22 presenta los resultados de los experimentos con controlador PID no sintonizado y autosintonizado para la variable *Nivel* de líquido en tanque.

**Tabla 22**

*Resultados de los experimentos con controlador PID no sintonizado y autosintonizado para la variable Nivel de líquido en tanque*

N° Experimento	Setpoint (mm)	Tiempo de asentamiento normalizado (%)	
		Con controlador PID no sintonizado	Con controlador PID autosintonizado
1	0  250	50.00	35.71
2	0  500	58.57	37.14
3	0  1000	61.43	37.14
4	0  1250	62.86	37.14
5	0  1500	67.14	45.71
6	0  1750	74.29	51.43
7	0  2000	80.00	54.29
8	0  2250	84.29	54.29
9	0  2500	90.00	54.29
10	0  3000	100.00	60.00

**Nota:** Los valores de tiempos de asentamiento se obtuvieron de la Tabla 17 y Tabla 19.

Por tratarse de grupos de muestras dependientes se empleó la prueba t de Student unilateral con cola a la derecha presentada en la ecuación 12.

$$t = \frac{\bar{d} - \mu_d}{S_d/\sqrt{n}} = 2.612 \quad (12)$$

Siendo la desviación estándar ( $S_d$ ), calculada con la ecuación 13; en base al cuadrado de la diferencia ( $d_i$ ) y la media de las diferencias ( $\bar{d}$ ) y el grado de libertad ( $n - 1$ ).

$$S_d = \sqrt{\frac{(d_i - \bar{d})^2}{n - 1}} = 7.43 \quad (13)$$

Con:  $n = 10$ ,  $S_d = 7.43$ ,  $\bar{d} = 26.144$ ,  $\mu_d = 20\%$ ; se calculó el valor del estadístico  $t = 2.612$ . Adicionalmente, la diferencia de medias, el valor crítico de t, los grados de libertad (GL) y valor-p unilateral para un intervalo de confianza del 95% se presentan en la Tabla 21.

**Tabla 23**

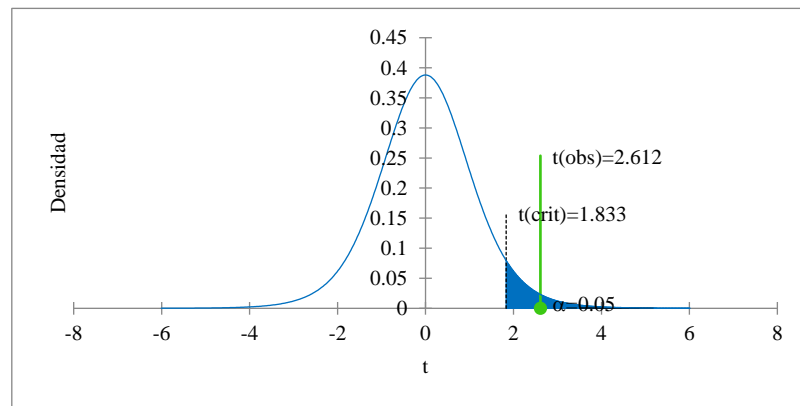
*Prueba t unilateral a la derecha para dos muestras dependientes de la variable Temperatura en planta térmica, para un intervalo de confianza del 95%*

Diferencia	26.144
t (Valor observado)	2.612
t (Valor crítico)	1.833
GL	9
valor-p (unilateral)	0.014
alfa	0.05

La representación gráfica de esta prueba se ve en la Figura 66.

**Figura 66**

*Representación gráfica de la prueba t de Student unilateral a la derecha para dos muestras dependientes de la variable Nivel de líquido en tanque, para un intervalo de confianza del 95%*



**Interpretación de la prueba para la Hipótesis específica b.** Puesto que el valor-p obtenido = 0.014, es menor que el nivel de significancia  $\alpha=0.05$ , se debe rechazar la hipótesis nula  $H_0$  y aceptar la hipótesis alternativa  $H_a$ .

#### 4.2.2. Prueba de la hipótesis general

Probadas las hipótesis específicas a y b; con aceptación de las hipótesis estadísticas alternativas, por inferencia queda probada la hipótesis general; en consecuencia, se afirma que:

*Es factible la implementación de un controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA, que reduce significativamente el tiempo de asentamiento de variables controladas.*

#### 4.3. DISCUSIÓN

De los resultados fue factible implementar el controlador PID autosintonizable; constituido por un servidor configurado en PLC S7 1500 que tiene como elemento fundamental el módulo PID\_Compact V2 configurable con la regla de Ziegler Nichols, con comunicación OPC UA que permitió la interacción con

el cliente de este servidor PID implementado en WinCC Unified, siendo capaz de controlar, desde su interface HMI, a la variable *Temperatura* en una planta térmica y a la variable Nivel de líquido en tanque implementado en Factory IO por medio de la comunicación OPC UA.

Se comprobó el funcionamiento determinando la reducción del tiempo de asentamiento de las variables controladas. Esta comprobación se llevó a cabo con respecto a parámetro PID pre establecidos, y luego de la autosintonización se obtuvieron parámetros PID diferentes para cada proceso; lo que evidencia que el controlador PID debe ser sintonizado de forma exclusiva para cada tipo de proceso, lo que concuerda con los resultados obtenidos por López y otros (2020), que determinan el rendimiento del controlador PID en un modelo simulado y en un sistema real para varios casos de cambios de referencia y perturbaciones.

Por otro lado, el control de la variable *Temperatura* en la planta térmica al ser autosintonizado presenta alta eficiencia (mayor a 60%) siendo los parámetros óptimos encontrados  $K_p = 6.03$ ,  $T_i = 0.74$  s,  $T_d = 0.18$  s para  $e_{ss} \leq 2\%$ , con señal de la temperatura controlada del tipo sub amortiguado, permitiendo una estabilización adecuada frente a diferentes valores del setpoint escalón; lo que concuerda con el trabajo de Radu y otros (2016) que desarrollan un sistema de control de temperatura preciso, basado en un algoritmo optimizado proporcional - integrativo - derivado (PID) con auto ajuste para un calefactor de soldadura; y también con el trabajo de Merzlikina y otros (2021) que analiza la dependencia entre el tiempo de predicción y el tiempo de retardo de la planta de control PID en un modelo de computadora del sistema que se desarrolla en Matlab y Simulink.

Así mismo, el controlador PID autosintonizado da respuesta, a entradas setpoint escalón con parámetros:  $K_p = 0.80$ ,  $T_i = 2.74$  s,  $T_d = 0.68$  s y  $e_{ss} \leq 2\%$  con una eficiencia superior al 20%, para la variable *Nivel* de líquido en tanque que es del tipo sub amortiguado lo que hace que el tiempo de asentamiento sea menor con respecto al controlador con parámetros PID predefinidos en PLC



Simatic S7 1500. La autosintonización permite encontrar parámetros óptimos con facilidad, tal como indica Kolaj y otros (2016), operados de forma remota desde interfaces HMI por comunicación OPC UA que es fácil de integrar con nuevos entornos industriales inclusive de industria 4.0 (Haskamp et al., 2018).

## CONCLUSIONES

Se implementó el servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID\_Compact configurable con la regla de Ziegler-Nichols, intercomunicando los datos mediante el protocolo industrial OPC UA con un cliente desarrollado en WinCC Unified, logrando reducir el tiempo de asentamiento de la variable controlada *Temperatura* de una planta térmica en más del 60 % para el error de estado estable  $e_{ss} \leq 2\%$ . Siendo la planta térmica el módulo SO3536-8T de Lucas Nulle, con rango de proceso nominal de 0 °C a 100 °C.

Así mismo, con el controlador PID autosintonizable implementado, se logró reducir el tiempo de asentamiento de la variable controlada *Nivel* de líquido en tanque en más del 20% para el error de estado estable  $e_{ss} \leq 2\%$ . Siendo la planta de control de nivel de líquido en tanque implementado en Factory I/O con comunicación OPC UA, con rango de proceso de 0 mm a 3000 mm.

Finalmente, se concluye que se implementó un controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA, que reduce significativamente el tiempo de asentamiento de las variables controladas.

## RECOMENDACIONES

El controlador PID debe sintonizarse en función de la variable temperatura en procesos térmicos al 50 % del rango de proceso, para reducir la desviación de la respuesta frente a cambios de consignas o setpoint extremos. Así mismo, la respuesta del calefactor debe ser lineal para un correcto funcionamiento en el ambiente de trabajo.

También, el controlador PID debe sintonizarse en función de la variable nivel de líquido en tanque al 50 % del rango de proceso, para reducir la desviación de la respuesta frente a cambios de consignas o setpoint extremos. Así mismo, la respuesta de las válvulas de llenado y descargo de líquido deben ser idénticos y lineales.

La interface HMI, desarrollados en WinCC Unified con comunicación OPC UA, debe ser usado de acuerdo a las credenciales y roles establecidos tanto para: administrador y operador; el cual permitirá realizar las acciones y tareas asignadas evitando errores de maniobra.

Para la réplica del proyecto se debe usar un PLC S7 1500 con CPU de versión de firmware 2.0 o superior, los programas y simuladores deben ser de versiones específicas: para la ejecución del programa y algoritmos PID la plataforma de TIA Portal V17, para creación de interfaces HMI WinCC Unified V17, para generación de escenas de ambientes virtuales de proceso industriales con comunicación OPC UA Factory I/O 2.4.3

Para trabajos futuros, queda pendiente probar la eficiencia del controlador PID autosintonizable en modelos de plantas industriales creados en Simulink-Matlab con la comunicación OPC UA. Así mismo, realizar la integración con aplicaciones remotas por internet que involucren la industria 4.0 que está en plena expansión.

## ***REFERENCIA BIBLIOGRÁFICA***

Alberto, F., & Mendoza, C. E. (2021). *Diseño y Comparación de Controladores: PID, Difuso y Robusto para un Sistema Térmico*. 2-5.

<https://www.docsity.com/es/comparacion-entre-controladores-pid-difuso-y-robusto-para-una-planta-termica/8277742/>

B&R Industrial Automation. (2020). *OPC UA*. <https://www.br-automation.com/es/tecnologias/opc-ua/>

Berger, H. (2017). *Automating with SIMATIC S7-1500* (2.<sup>a</sup> ed.).

Beschi, M., Dormido, S., Sánchez, J., & Visioli, A. (2015). An Event-based PI controller autotuning technique for integral processes. *2015 International Conference on Event-based Control, Communication, and Signal Processing (EBCCSP)*, 1-6. <https://doi.org/10.1109/EBCCSP.2015.7300684>

Biblioteca DUOC UC. (2018). *Definición y propósito de la Investigación Aplicada*. <http://www.duoc.cl/biblioteca/crai/definicion-y-proposito-de-la-investigacion-aplicada>

Caiza, G., Nuñez, A., Garcia, C. A., & Garcia, M. V. (2020). Human Machine Interfaces Based on Open Source Web-Platform and OPC UA. *Procedia Manufacturing*, 42, 307-314. <https://doi.org/https://doi.org/10.1016/j.promfg.2020.02.089>

Cegarra, J. (2004). Metodología de la Investigación Científica y Tecnológica. En *Metodología de la investigación científica y tecnológica*. [https://books.google.com.pe/books/about/Metodología\\_de\\_la\\_investigación\\_cient.html?hl=es&id=8SA8KZyurk4C](https://books.google.com.pe/books/about/Metodología_de_la_investigación_cient.html?hl=es&id=8SA8KZyurk4C)

CEPLAN. (2021). *Megatendencias 2050: grandes retos e implicancias*. [https://cdn.www.gob.pe/uploads/document/file/3294036/CEPLAN.Megatendencias 2050%3A grandes retos e implicancias.pdf](https://cdn.www.gob.pe/uploads/document/file/3294036/CEPLAN.Megatendencias%2050%3A%20grandes%20retos%20e%20implicancias.pdf)

- Espinoza, C. (2014). *Metodología de investigación tecnológica Pensando en sistemas*.
- Ferrari, P., Flammini, A., Rinaldi, S., Sisinni, E., Maffei, D., & Malara, M. (2018). Evaluation of communication delay in IoT applications based on OPC UA. *2018 Workshop on Metrology for Industry 4.0 and IoT*, 224-229.
- Figueroa, A. A. P., Silupu, J. J. M., & Garcia Zabaleta, R. S. (2017). Adaptive PID controller with auto-tuning applied to the agricultural food industry. *2017 CHILEAN Conference on Electrical, Electronics Engineering, Information and Communication Technologies (CHILECON)*, 1-7.  
<https://doi.org/10.1109/CHILECON.2017.8229714>
- García, M. V, Irisarri, E., Pérez, F., Estévez, E., & Marcos, M. (2016). OPC-UA communications integration using a CPPS architecture. *2016 IEEE Ecuador Technical Chapters Meeting (ETCM)*, 1-6.  
<https://doi.org/10.1109/ETCM.2016.7750838>
- Haskamp, H., Orth, F., Wermann, J., & Colombo, A. W. (2018). Implementing an OPC UA interface for legacy PLC-based automation systems using the Azure cloud: An ICPS-architecture with a retrofitted RFID system. *2018 IEEE Industrial Cyber-Physical Systems (ICPS)*, 115-121.  
<https://doi.org/10.1109/ICPHYS.2018.8387646>
- Hernández-Sampieri, R., & Mendoza Torres, C. P. (2018). *Metodología de la investigación: las rutas cuantitativa, cualitativa y mixta*. McGRAW-HILL.
- Hildebrandt, M., Lamshöft, K., Dittmann, J., Neubert, T., & Vielhauer, C. (2020). Information hiding in industrial control systems: An OPC UA based supply chain attack and its detection. *Proceedings of the 2020 ACM Workshop on Information Hiding and Multimedia Security*, 115-120.
- Kamatchi Sundari, V., Nithyashri, J., Kuzhaloli, S., Subburaj, J., Vijayakumar, P., & Subha Hency Jose, P. (2021). Comparison analysis of IoT based industrial

automation and improvement of different processes – review. *Materials Today: Proceedings*, 45, 2595-2598.

<https://doi.org/https://doi.org/10.1016/j.matpr.2020.11.338>

Kolaj, W., Mozaryn, J., & Syfert, M. (2016). PLC-PIDTuner: Application for PID tuning with SIMATIC S7 PLC controllers. *2016 21st International Conference on Methods and Models in Automation and Robotics (MMAR)*, 306-311.

<https://doi.org/10.1109/MMAR.2016.7575152>

Kumar Mishra, A., Kumar Tiwari, V., Kumar, R., Verma, T. (2013). Speed Control of DC Motor Using Artificial Bee Colony Optimization Technique. *Universal Journal of Electrical and Electronic Engineering* 1, 68–75.

<https://doi.org/doi:10.13189/ujeee.2013.010302>

Kuzishchin, V. F., Merzlikina, E. I., & Van Va, H. (2018). Study of Maximal Rapid Response Algorithm Combined with PID-Algorithm and Autotuning Module. *2018 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM)*, 1-6. <https://doi.org/10.1109/ICIEAM.2018.8728821>

Lee, B., Kim, D.-K., Yang, H., & Oh, S. (2017). Model transformation between OPC UA and UML. *Computer Standards & Interfaces*, 50, 236-250.

<https://doi.org/https://doi.org/10.1016/j.csi.2016.09.004>

López-Rivera, M., Camilo Cortés-Villada, A., Hincapié-Franco, F., Andrés Duque-Giraldo, J., & Giraldo, E. (2020). Análisis e implementación de un controlador PID en un helicóptero no tripulado de dos grados de libertad. *2020 IX International Congress of Mechatronics Engineering and Automation (CIIMA)*, 1-6. <https://doi.org/10.1109/CIIMA50553.2020.9290320>

Mathias, S. G., Schmied, S., & Grossmann, D. (2020). An Investigation on Database Connections in OPC UA Applications. *Procedia Computer Science*, 170, 602-609. <https://doi.org/https://doi.org/10.1016/j.procs.2020.03.132>

Merzlikina, E., Van Va, H., & Farafonov, G. (2021). Automatic Control System with

an Autotuning Module and a Predictive PID-Algorithm for Thermal Processes. *2021 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM)*, 525-529.  
<https://doi.org/10.1109/ICIEAM51226.2021.9446467>

Natale, L. F. C., Campos-Mercado, E., Fernández, J. C. R., Márquez-Vera, M. A., & Ac, I. A. C. (2017). Comparación experimental de controladores PID clásico, PID no lineal y PID difuso para el caso de regulación. *Res. Comput. Sci.*, 135, 173-187.

Ogata, K. (1996). *Sistemas de control en tiempo discreto* (2da ed.). Prentice Hall Hispanoamericana S.A.

Ogata, K. (2010). *Ingeniería de control moderna* (5.<sup>a</sup> ed.). Pearson.

Ortiz, P. S. C., Dutan, B. R. N., Jaramillo, L. G. T., Ávila-Campoverde, R. S., Aleman, M. A. C., Montesdeoca-Contreras, J. C., & Morales-Garcia, J. A. (2014). Linearization of mass spring damper system for applying linear control PID techniques. *2014 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, 1-5.  
<https://doi.org/10.1109/ROPEC.2014.7036300>

Portilla, J. (s. f.). *Introducción a OPC UA para Sistemas de Comunicación Industrial*.

Radu, V., Avram, A., Anghel, V., & Brezeanu, G. (2016). Autotuned PID for accurate temperature control — A hot approach. *2016 International Semiconductor Conference (CAS)*, 209-212.  
<https://doi.org/10.1109/SMICND.2016.7783088>

Real Games. (2022). *Features - Factory I/O*. <https://factoryio.com/features>

Siemen. (2018). *S7-1500/ET 200MP Módulo de entradas analógicas AI 8xU/I/RTD/TC ST (6ES7531-7KF00-0AB0)*.  
[https://cache.industry.siemens.com/dl/files/205/59193205/att\\_112068/v1/s7150\\_0\\_ai\\_8xu\\_i\\_rtd\\_tc\\_st\\_manual\\_es-ES\\_es-ES.pdf](https://cache.industry.siemens.com/dl/files/205/59193205/att_112068/v1/s7150_0_ai_8xu_i_rtd_tc_st_manual_es-ES_es-ES.pdf)

- Siemens. (2018). *Módulo de salidas analógicas AQ 4xU/I ST*.  
[https://support.industry.siemens.com/cs/attachments/59191850/s71500\\_aq\\_4xu\\_i\\_st\\_manual\\_es-ES\\_es-ES.pdf?download=true](https://support.industry.siemens.com/cs/attachments/59191850/s71500_aq_4xu_i_st_manual_es-ES_es-ES.pdf?download=true)
- Siemens. (2019a). *Documentación didáctica/ para cursos de formación, Módulo TIA Portal 092-300 OPC UA con SIMATIC S7-1500 como servidor OPC, así como OPC SCOUT y SIMIT como clientes OPC*.  
<https://www.automation.siemens.com/sce-static/learning-training-documents/tia-portal/basics-programming-s7-1200/sce-031-300-iec-times-and-counters-s7-1200-r1709-es.pdf>
- Siemens. (2019b). *OPC UA Server & Client*. <https://www.infopl.net/descargas/105-siemens/automatas/s7-1500/3111-simatic-s7-1500-opc-ua-server-client>
- Siemens. (2019c). *Web-based visualization system sets new standards for industrial operator control and monitoring*.  
<https://press.siemens.com/global/en/pressrelease/web-based-visualization-system-sets-new-standards-industrial-operator-control-and>
- Siemens. (2019d). *Which browser can you use for SIMATIC WinCC Unified? - ID: 109757952*.  
<https://support.industry.siemens.com/cs/document/109757952/which-browser-can-you-use-for-simatic-wincc-unified-?dti=0&pnid=16743&lc=en-US>
- Singhal, R., Padhee, S., & Kaur, G. (2012). *Design of Fractional Order PID Controller for Speed Control of DC Motor*.
- Smith, C. A., & Corripio, A. (1991). *Control automático de procesos teoría y práctica*. Editorial Limusa SA.
- Somefun, O. A., Akingbade, K., & Dahunsi, F. (2021). The dilemma of PID tuning. *Annual Reviews in Control*.  
<https://doi.org/https://doi.org/10.1016/j.arcontrol.2021.05.002>
- Unified Automation GmbH. (2018). *UaExpert—A Full-Featured OPC UA Client*.



<https://www.unified-automation.com/products/development-tools/uaexpert.html>

Yu, C.-C. (2007). *Autotuning of PID Controllers* (2.<sup>a</sup> ed.). Springer.

Zhang, H., Assawinchaichote, W., & Shi, Y. (2021). New PID Parameter Autotuning for Nonlinear Systems Based on a Modified Monkey–Multiagent DRL Algorithm. *IEEE Access*, 9, 78799-78811.

<https://doi.org/10.1109/ACCESS.2021.3083705>

## ***APÉNDICE***

## A. Muestras tomadas para el controlador PID no sintonizado para la variable temperatura del proceso térmico

**Tabla 24**

*Mediciones con el controlador PID sin sintonizar para setpoint 25 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable ( $e_{ss}$ )
1	11:34:16	12.62659	25	13.113	26.26592
2	11:34:22	13.26678		21.249	10.55651
3	11:34:23	13.28848		21.346	10.73736
4	11:34:23	13.2921		21.441	10.76751
5	11:34:23	13.28848		21.53	10.73736
6	11:34:23	13.28487		21.62	10.70722
7	11:34:24	13.26678		21.874	10.55651
8	11:34:25	13.25593		21.954	10.4661
9	11:34:25	13.23785		22.108	10.31539
10	11:34:26	13.20891		22.395	10.07427
11	11:34:26	13.20529		22.461	10.04412
12	11:34:27	13.19444		22.526	47.22222
13	11:34:27	13.19083		22.591	47.23669
14	11:34:28	17.03921		22.771	31.84317
15	11:34:28	17.39728		22.828	30.41088
16	11:34:28	18.29065		22.883	26.83739
17	11:34:29	19.72656		23.093	21.09375
18	11:34:29	19.7989		23.142	20.8044
19	11:34:30	20.0557		23.19	19.7772
20	11:34:31	20.35229		23.369	18.59086
21	11:34:31	20.37399		23.41	18.50405
22	11:34:31	20.44271		23.452	18.22917
23	11:34:31	20.45718		23.491	18.1713
24	11:34:32	20.59462		23.64	17.62153
25	11:34:33	20.71398		23.774	17.1441
26	11:34:34	20.75738		23.806	16.97049
27	11:34:34	20.76461		23.837	16.94155
28	11:35:54	24.47555		24.98	2.097801
29	11:35:54	24.47917		24.981	2.083336
30	11:35:55	24.48278		24.983	2.068871
31	11:35:56	24.4864		24.982	2.054398
32	11:35:56	24.49002		24.981	2.039932
33	11:35:57	24.49363		24.982	2.025467
34	11:35:59	24.49725		24.983	2.010994
35	11:36:05	24.50087		24.984	1.996529
<b><math>t_s = 0:01:49</math></b>		<b><math>M_p = -1.997 \%</math></b>		<b><math>e_{ss} \leq 2 \%</math></b>	

*Nota: Parámetros PID por defecto:  $K_p = 1.0$ ,  $T_i = 20.0$  s,  $T_d = 0.1$  s*

**Tabla 25***Mediciones con el controlador PID sin sintonizar para setpoint 30 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (ess)
1	11:53:28	9.621	30	8.706	862.095
2	11:53:40	7.328		37.450	632.784
3	11:53:40	7.310		29.643	630.975
4	11:53:41	7.245		23.650	624.465
5	11:54:31	24.790		29.951	17.366
6	11:54:31	24.805		29.954	17.318
7	11:54:36	25.427		29.970	15.244
8	11:54:36	25.438		29.972	15.208
9	11:54:37	25.485		29.974	15.051
10	11:54:37	25.496		29.975	15.015
11	11:54:37	25.543		29.973	14.858
12	11:54:37	25.557		29.974	14.810
13	11:54:38	25.604		29.976	14.653
14	11:54:38	25.611		29.977	14.629
15	11:54:38	25.658		29.978	14.472
16	11:54:38	25.669		29.977	14.436
17	11:54:39	25.716		29.977	14.280
18	11:54:39	25.727		29.978	14.243
19	11:54:39	25.770		29.979	14.099
20	11:54:39	25.781		29.981	14.063
21	11:54:40	25.825		29.982	13.918
22	11:54:41	25.984		29.984	13.387
23	11:54:41	25.995		29.985	13.351
24	11:55:51	29.358		29.983	2.139
25	11:55:52	29.362		29.983	2.127
26	11:55:52	29.366		29.984	2.115
27	11:55:52	29.369		29.985	2.103
28	11:55:52	29.373		29.985	2.091
29	11:55:53	29.376		29.986	2.079
30	11:55:53	29.380		29.987	2.066
31	11:55:53	29.384		29.988	2.054
32	11:55:53	29.391		29.988	2.030
33	11:55:54	29.395		29.985	2.018
34	11:55:54	29.398		29.986	2.006
35	11:55:54	29.402		29.987	1.994
<b>ts = 0:02:26</b>		<b>Mp = -1.994%</b>			<b>ess ≤ 2%</b>

Nota: Parámetros PID por defecto: Kp= 1.0, Ti= 20.0 s, Td=0.1 s

**Tabla 26***Mediciones con el controlador PID sin sintonizar para setpoint 35 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (ess)
1	12:06:36	6.496	35	5.002	100.000
2	12:08:23	31.767		23.451	9.237
3	12:08:24	31.796		23.737	9.154
4	12:08:24	31.807		24.024	9.123
5	12:08:24	31.843		24.304	9.020
6	12:08:24	31.850		24.578	8.999
7	12:08:25	31.879		24.846	8.916
8	12:08:29	32.227		28.643	7.924
9	12:08:29	32.234		28.806	7.903
10	12:08:30	32.259		28.966	7.831
11	12:08:30	32.270		29.120	7.800
12	12:08:39	32.863		32.755	6.105
13	12:08:39	32.867		32.813	6.095
14	12:08:40	32.888		32.868	6.033
15	12:08:40	32.896		32.924	6.012
16	12:08:40	32.917		32.977	5.950
17	12:08:40	32.921		33.028	5.940
18	12:08:41	32.943		33.079	5.878
19	12:08:41	32.950		33.129	5.857
20	12:08:41	32.972		33.176	5.795
21	12:08:41	32.975		33.224	5.785
22	12:08:42	32.997		33.269	5.723
23	12:08:42	33.001		33.313	5.713
24	12:08:42	33.022		33.357	5.651
25	12:08:56	33.608		34.597	3.977
26	12:08:59	33.731		34.727	3.625
27	12:09:19	34.245		34.965	2.158
28	12:09:19	34.248		34.967	2.147
29	12:09:20	34.252		34.966	2.137
30	12:09:20	34.256		34.967	2.127
31	12:09:21	34.285		34.974	2.044
32	12:09:22	34.292		34.972	2.023
33	12:09:22	34.295		34.973	2.013
34	12:09:22	34.299		34.974	2.003
35	12:09:22	34.303		34.976	1.992
<b>ts = 0:02:46</b>		<b>Mp = -1.992%</b>		<b>ess ≤ 2%</b>	

Nota: Parámetros PID por defecto: Kp= 1.0, Ti= 20.0 s, Td=0.1 s

**Tabla 27***Mediciones con el controlador PID sin sintonizar para setpoint 40 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (ess)
1	12:23:48	1.074	40.000	1.069	7.422
2	12:25:40	36.429		29.847	9.017
3	12:25:40	36.440		30.107	8.927
4	12:25:40	36.476		30.361	8.899
5	12:25:40	36.487		30.608	8.809
6	12:25:42	36.665		32.374	8.357
7	12:25:43	36.697		32.569	8.339
8	12:25:43	36.704		32.760	8.257
9	12:25:43	36.740		32.946	8.239
10	12:25:43	36.751		33.128	8.149
11	12:25:44	36.784		33.304	8.122
12	12:25:44	36.791		33.476	8.040
13	12:25:44	36.824		33.644	8.022
14	12:25:44	36.834		33.807	7.941
15	12:25:46	36.994		34.971	7.534
16	12:25:47	37.023		35.100	7.516
17	12:25:49	37.214		36.222	7.046
18	12:25:49	37.222		36.319	6.964
19	12:25:50	37.247		36.414	6.946
20	12:25:50	37.254		36.507	6.883
21	12:25:50	37.283		36.596	6.865
22	12:25:53	37.489		37.508	6.349
23	12:25:53	37.496		37.573	6.277
24	12:25:54	37.522		37.635	6.259
25	12:25:54	37.525		37.697	6.196
26	12:25:54	37.554		37.754	6.187
27	12:25:54	37.558		37.812	6.114
28	12:25:55	37.583		37.869	6.105
29	12:25:55	37.590		37.924	6.042
30	12:26:36	39.182		39.971	2.054
31	12:26:37	39.189		39.973	2.045
32	12:26:37	39.193		39.974	2.027
33	12:26:37	39.200		39.976	2.018
34	12:26:37	39.204		39.974	2.000
35	12:26:37	39.207		39.975	1.991
<b>ts = 0:02:49</b>		<b>Mp = -1.982%</b>			<b>ess ≤ 2%</b>

Nota: Parámetros PID por defecto: Kp= 1.0, Ti= 20.0 s, Td=0.1 s

**Tabla 28***Mediciones con el controlador PID sin sintonizar para setpoint 50 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (ess)
1	17:19:20	1.859	50	2.689	85.909
2	17:19:20	1.855		57.835	85.547
3	17:20:25	43.099		57.832	13.802
4	17:20:30	43.924		57.829	12.153
5	17:20:30	43.942		57.826	12.117
6	17:20:30	44.003		57.824	11.994
7	17:20:34	44.538		57.821	10.923
8	17:20:34	44.593		57.818	10.815
9	17:20:34	44.607		57.815	10.786
10	17:20:35	44.665		57.813	10.670
11	17:20:36	44.882		57.810	10.236
12	17:20:37	44.936		57.807	10.127
13	17:20:37	44.951		57.804	10.098
14	17:20:37	45.001		57.801	9.997
15	17:20:37	45.012		57.799	9.975
16	17:20:45	45.949		57.796	8.102
17	17:20:46	45.992		57.793	8.015
18	17:20:46	46.003		57.790	7.993
19	17:20:46	46.043		57.788	7.914
20	17:20:50	46.398		44.845	7.205
21	17:20:50	46.434		44.977	7.133
22	17:20:54	46.795		46.773	6.409
23	17:21:02	47.363		48.522	5.273
24	17:21:08	47.392		48.559	5.216
25	17:21:12	47.396		48.597	5.208
26	17:21:18	47.425		48.633	5.150
27	17:21:25	47.432		48.667	5.136
28	17:21:30	48.658		49.898	2.684
29	17:21:42	48.962		49.964	2.076
30	17:21:45	48.966		49.966	2.069
31	17:21:47	48.976		49.965	2.047
32	17:21:52	48.980		49.966	2.040
33	17:21:55	48.991		49.968	2.018
34	17:22:11	48.995		49.970	2.011
35	17:22:13	49.002		49.968	1.997
<b>ts = 0:02:53</b>		<b>Mp = -1.997%</b>			<b>ess ≤ 2%</b>

Nota: Parámetros PID por defecto: Kp= 1.0, Ti= 20.0 s, Td=0.1 s

**Tabla 29***Mediciones con el controlador PID sin sintonizar para setpoint 60 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (ess)
1	17:55:25	2.333	60	2.196	133.290
2	17:55:37	1.975		70.123	97.483
3	17:57:15	54.420		41.313	9.391
4	17:57:18	54.854		46.676	8.601
5	17:57:19	54.908		47.018	8.577
6	17:57:19	54.923		47.351	8.486
7	17:57:19	54.973		47.676	8.462
8	17:57:33	56.474		57.053	5.894
9	17:57:33	56.518		57.203	5.822
10	17:57:34	56.554		57.276	5.804
11	17:57:37	56.814		58.006	5.370
12	17:57:37	56.854		58.106	5.298
13	17:57:40	57.060		58.578	4.912
14	17:57:41	57.129		58.685	4.839
15	17:57:41	57.136		58.718	4.785
16	17:57:41	57.165		58.751	4.773
17	17:57:42	57.201		58.814	4.713
18	17:57:45	57.411		59.133	4.357
19	17:57:45	57.444		59.177	4.303
20	17:57:45	57.451		59.198	4.261
21	17:57:46	57.483		59.238	4.200
22	17:57:46	57.509		59.258	4.194
23	17:57:46	57.516		59.277	4.152
24	17:57:47	57.548		59.314	4.092
25	17:57:47	57.574		59.331	4.086
26	17:57:51	57.787		59.535	3.730
27	17:57:54	57.954		59.660	3.447
28	17:57:54	57.982		59.685	3.369
29	17:57:55	58.030		59.709	3.314
30	17:58:11	58.702		59.945	2.187
31	17:58:11	58.706		59.948	2.163
32	17:58:14	58.796		59.960	2.024
33	17:58:16	58.800		59.960	2.006
34	17:58:20	58.814		59.962	2.000
35	17:58:21	58.818		59.964	1.976
<b>ts = 0:02:56</b>		<b>Mp = -1.970 %</b>		<b>ess ≤ 2%</b>	

Nota: Parámetros PID por defecto: Kp= 1.0, Ti= 20.0 s, Td=0.1 s



**Tabla 30***Mediciones con el controlador PID sin sintonizar para setpoint 70 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (ess)
1	18:17:47	1.034	70	1.034	3.443
2	18:18:45	44.430		80.360	36.637
3	18:18:46	44.708		80.359	36.529
4	18:18:46	44.774		80.358	36.131
5	18:19:11	56.959		70.357	18.806
6	18:19:29	61.860		61.972	11.743
7	18:19:29	61.885		47.899	11.629
8	18:19:39	63.744		53.370	8.963
9	18:19:45	64.648		59.856	7.666
10	18:19:46	64.706		60.116	7.645
11	18:19:48	64.985		64.706	7.180
12	18:19:49	65.101		64.842	7.072
13	18:19:52	65.480		64.976	6.472
14	18:19:52	65.527		65.104	6.457
15	18:19:53	65.585		65.352	6.374
16	18:20:10	67.206		67.165	4.002
17	18:20:11	67.238		67.756	3.992
18	18:20:23	68.001		69.890	2.881
19	18:20:23	68.009		69.893	2.855
20	18:20:24	68.030		69.896	2.845
21	18:20:28	68.229		69.899	2.535
22	18:20:28	68.251		69.937	2.504
23	18:20:29	68.269		69.938	2.499
24	18:20:33	68.443		69.940	2.251
25	18:20:33	68.446		69.941	2.225
26	18:20:33	68.461		69.943	2.220
27	18:20:33	68.464		69.944	2.199
28	18:20:34	68.482		69.945	2.194
29	18:20:34	68.486		69.948	2.168
30	18:20:34	68.500		69.947	2.163
31	18:20:36	68.580		69.950	2.034
32	18:20:37	68.594		69.951	2.029
33	18:20:42	68.598		69.952	2.008
34	18:20:43	68.613		69.955	2.003
35	18:20:44	68.616		69.953	1.982
<b>ts = 0:02:57</b>		<b>Mp = -1.977 %</b>		<b>ess ≤ 2%</b>	

Nota: Parámetros PID por defecto: Kp= 1.0, Ti= 20.0 s, Td=0.1 s

**Tabla 31***Mediciones con el controlador PID sin sintonizar para setpoint 80 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (ess)
1	18:32:24	2.022	80	2.111	106.163
2	18:32:36	1.747		80.661	76.866
3	18:33:07	13.520		80.161	35.272
4	18:33:51	59.762		79.661	15.158
5	18:33:53	60.305		48.184	14.249
6	18:33:54	60.413		43.929	14.218
7	18:33:55	60.796		46.910	13.525
8	18:33:59	61.704		54.772	12.192
9	18:34:00	61.798		55.163	12.161
10	18:34:01	62.008		56.629	11.851
11	18:34:02	62.326		58.856	11.272
12	18:34:03	62.413		59.142	11.252
13	18:34:11	63.929		65.631	8.921
14	18:34:13	64.153		66.165	8.673
15	18:34:13	64.236		66.453	8.467
16	18:34:14	64.316		66.632	8.353
17	18:34:15	64.450		66.885	8.234
18	18:34:19	65.072		68.100	7.247
19	18:34:20	65.126		68.149	7.227
20	18:34:20	65.198		68.288	7.061
21	18:34:21	65.252		68.332	7.041
22	18:34:25	65.784		68.982	6.198
23	18:34:26	65.831		69.008	6.183
24	18:34:26	65.842		69.034	6.116
25	18:34:27	65.987		69.151	5.940
26	18:34:27	65.998		69.174	5.883
27	18:34:28	66.041		69.195	5.868
28	18:34:39	67.072		69.763	4.307
29	18:34:40	67.104		69.769	4.297
30	18:34:45	67.495		69.873	3.682
31	18:34:46	67.520		69.876	3.677
32	18:34:50	67.614		69.894	3.532
33	18:34:59	67.622		69.897	3.496
34	18:35:22	68.638		69.954	2.018
35	18:35:23	68.641		69.954	1.998
<b>ts = 0:02:59</b>		<b>Mp = -1.941%</b>			<b>ess ≤ 2%</b>

Nota: Parámetros PID por defecto: Kp= 1.0, Ti= 20.0 s, Td=0.1 s

**Tabla 32***Mediciones con el controlador PID sin sintonizar para setpoint 90 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (ess)
1	18:49:01	2.966	90	2.774	196.586
2	18:49:13	2.434		100.000	143.417
3	18:49:18	2.246		67.713	124.609
4	18:49:19	2.242		68.285	124.248
5	18:49:19	2.232		68.842	123.163
6	18:49:23	2.094		76.754	109.418
7	18:49:24	2.091		77.094	109.057
8	18:51:02	83.482		80.050	7.243
9	18:51:03	83.554		80.307	7.162
10	18:51:07	84.277		84.088	6.359
11	18:51:08	84.339		84.238	6.290
12	18:51:13	85.030		86.576	5.523
13	18:51:13	85.044		86.664	5.507
14	18:51:13	85.095		86.749	5.450
15	18:51:13	85.106		86.833	5.438
16	18:51:14	85.160		86.914	5.378
17	18:51:14	85.171		86.994	5.366
18	18:51:21	86.024		88.622	4.417
19	18:51:23	86.227		88.882	4.192
20	18:51:29	86.773		89.401	3.586
21	18:51:30	86.806		89.416	3.549
22	18:51:36	87.305		89.703	2.995
23	18:51:39	87.507		89.783	2.770
24	18:51:40	87.543		89.799	2.730
25	18:51:40	87.569		89.804	2.701
26	18:51:41	87.609		89.819	2.657
27	18:51:46	87.927		89.898	2.304
28	18:51:49	88.079		89.923	2.135
29	18:51:50	88.108		89.929	2.103
30	18:51:50	88.126		89.931	2.083
31	18:51:51	88.155		89.936	2.050
32	18:51:51	88.173		89.938	2.030
33	18:51:51	88.180		89.939	2.022
34	18:52:03	88.198		89.941	2.002
35	18:52:05	88.202		89.942	1.998
<b>ts = 0:03:04</b>		<b>Mp = -1.998%</b>			<b>ess ≤ 2%</b>

Nota: Parámetros PID por defecto: Kp= 1.0, Ti= 20.0 s, Td=0.1 s

**Tabla 33***Mediciones con el controlador PID sin sintonizar para setpoint 96 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (ess)
1	19:04:33	5.389	96	4.954	438.918
2	19:04:45	4.170		100.000	317.028
3	19:06:37	89.178		78.332	7.106
4	19:06:39	89.424		80.487	6.850
5	19:06:41	89.757		83.402	6.503
6	19:06:41	89.833		84.040	6.424
7	19:06:45	90.444		88.114	5.787
8	19:06:46	90.520		88.513	5.708
9	19:06:48	90.795		89.920	5.422
10	19:06:48	90.809		90.076	5.407
11	19:06:48	90.860		90.228	5.354
12	19:06:52	91.323		92.093	4.872
13	19:06:52	91.370		92.193	4.823
14	19:06:52	91.381		92.291	4.812
15	19:06:55	91.717		93.215	4.461
16	19:06:57	91.934		93.738	4.235
17	19:06:58	92.039		93.962	4.126
18	19:07:00	92.249		94.387	3.907
19	19:07:01	92.300		94.469	3.855
20	19:07:04	92.567		95.317	3.576
21	19:07:07	92.875		95.538	3.256
22	19:07:08	92.907		95.549	3.222
23	19:07:10	93.099		95.653	3.022
24	19:07:11	93.175		95.687	2.943
25	19:07:14	93.356		95.758	2.754
26	19:07:14	93.359		95.765	2.751
27	19:07:14	93.388		95.771	2.721
28	19:07:15	93.428		95.788	2.679
29	19:07:23	93.913		95.907	2.174
30	19:07:25	93.938		95.913	2.148
31	19:07:28	93.960		95.914	2.125
32	19:07:35	94.036		95.926	2.046
33	19:07:36	94.065		95.930	2.016
34	19:07:40	94.068		95.933	2.012
35	19:07:46	94.086		95.934	1.993
<b>ts = 0:03:13</b>		<b>Mp = -1.993%</b>		<b>ess ≤ 2%</b>	

Nota: Parámetros PID por defecto: Kp= 1.0, Ti= 20.0 s, Td=0.1 s

## B. Muestras tomadas para el controlador PID autosintonizado para la variable temperatura del proceso térmico

**Tabla 34**

*Mediciones con el controlador PID autosintonizado para setpoint 25 °C*

N° medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable ( $e_{ss}$ )
1	12:12:59	11.997	25	11.993	0.023
2	12:13:13	17.299		57.83	30.802
3	12:13:13	18.627		58.455	25.492
4	12:13:14	23.046		47.629	7.813
5	12:13:14	23.853		44.947	4.586
6	12:13:14	26.009		35.502	4.036
7	12:13:14	26.320		33.43	5.281
8	12:13:15	26.880		27.192	7.523
9	12:13:15	26.891		26.062	7.566
10	12:13:15	26.623		23.181	6.496
11	12:13:15	26.511		22.741	6.047
12	12:13:16	25.998		22.173	3.993
13	12:13:16	25.871		22.143	3.486
14	12:13:16	25.426		22.594	1.707
15	12:13:16	25.332		22.849	1.331
16	12:13:17	25.057		23.574	1.231
17	12:13:17	25.007		23.833	1.029
18	12:13:17	24.891		24.409	1.434
19	12:13:17	24.873		24.622	1.506
20	12:13:18	24.858		24.964	0.564
21	12:13:18	24.862		25.014	0.549
22	12:13:18	24.891		25.158	0.434
23	12:13:18	24.898		25.238	1.405
24	12:13:19	24.938		25.24	0.246
25	12:13:19	24.949		25.176	0.203
26	12:13:19	24.978		25.148	0.087
27	12:13:19	24.981		25.181	0.072
28	12:13:20	25.000		25.127	0.005
29	12:13:20	25.004		25.062	0.014
30	12:13:20	25.007		25.094	0.028
31	12:13:20	25.011		24.983	1.043
32	12:13:21	25.007		24.988	0.028
33	12:13:22	25.004		24.979	0.014
34	12:13:22	25.000		24.974	1.001
35	12:13:23	24.996		24.968	0.014
<b>ts = 0:00:24</b>		<b>Mp = 7.567%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID:  $K_p= 6.03$ ,  $T_i= 0.74$  s,  $T_d=0.18$  s*

**Tabla 35***Mediciones con el controlador PID autosintonizado para setpoint 30 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (e <sub>ss</sub> )
1	12:24:14	11.997	30	11.955	0.023
2	12:24:28	19.126		77.889	36.246
3	12:24:28	21.050		76.900	29.832
4	12:24:29	27.293		61.954	9.023
5	12:24:29	28.429		58.237	5.237
6	12:24:29	31.467		44.770	4.890
7	12:24:29	31.905		41.818	6.349
8	12:24:30	32.682		32.984	8.941
9	12:24:30	32.690		31.504	8.965
10	12:24:30	32.306		27.345	7.687
11	12:24:30	32.143		26.775	7.145
12	12:24:31	31.413		25.893	4.709
13	12:24:31	31.232		25.936	4.106
14	12:24:31	30.599		26.622	1.997
15	12:24:31	30.469		26.903	1.563
16	12:24:32	30.078		27.978	1.260
17	12:24:32	30.009		28.305	1.031
18	12:24:32	29.843		29.177	1.523
19	12:24:32	29.821		29.413	1.596
20	12:24:33	29.800		29.916	0.668
21	12:24:33	29.803		30.055	0.656
22	12:24:33	29.847		30.236	0.511
23	12:24:33	29.857		30.327	0.475
24	12:24:34	29.915		30.283	0.282
25	12:24:34	29.926		30.318	0.246
26	12:24:34	29.970		30.243	0.101
27	12:24:34	29.977		30.219	0.077
28	12:24:35	30.002		30.126	0.007
29	12:24:35	30.006		30.099	0.019
30	12:24:36	30.013		30.059	0.043
31	12:24:36	30.017		29.983	1.055
32	12:24:37	30.013		29.935	0.043
33	12:24:38	30.009		30.026	0.031
34	12:24:40	30.006		29.971	0.019
35	12:24:41	30.002		29.964	0.007
<b>ts = 0:00:27</b>		<b>Mp = 8.965%</b>			<b>ess ≤ 2 %</b>

*Nota: Parámetros PID: Kp= 6.03, Ti= 0.74 s, Td=0.18 s*

**Tabla 36***Mediciones con el controlador PID autosintonizado para setpoint 35 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (e <sub>ss</sub> )
1	12:29:04	12.001	35	12.058	0.007
2	12:29:19	19.871		90.726	43.225
3	12:29:19	22.193		89.445	36.591
4	12:29:19	29.944		75.251	14.445
5	12:29:19	31.431		71.253	10.198
6	12:29:20	35.637		55.832	1.821
7	12:29:20	36.299		52.385	3.712
8	12:29:20	37.731		41.227	7.804
9	12:29:20	37.844		39.230	8.125
10	12:29:21	37.695		33.365	7.701
11	12:29:21	37.551		32.553	7.288
12	12:29:21	36.813		30.724	5.179
13	12:29:21	36.614		30.604	4.611
14	12:29:22	35.876		30.963	2.503
15	12:29:22	35.713		31.321	2.038
16	12:29:22	35.214		32.429	1.612
17	12:29:22	35.124		32.713	1.353
18	12:29:23	34.878		33.860	1.349
19	12:29:23	34.845		34.022	1.442
20	12:29:23	34.784		34.734	2.618
21	12:29:23	34.780		34.936	2.628
22	12:29:24	34.813		35.239	2.535
23	12:29:24	34.827		35.253	1.494
24	12:29:24	34.885		35.389	1.329
25	12:29:24	34.903		35.278	1.277
26	12:29:25	34.950		35.332	1.143
27	12:29:25	34.965		35.193	1.101
28	12:29:25	34.993		35.193	0.019
29	12:29:26	35.001		35.109	0.002
30	12:29:27	35.012		35.095	0.033
31	12:29:28	35.015		35.020	0.043
32	12:29:29	35.012		35.011	1.033
33	12:29:30	35.008		34.998	0.023
34	12:29:32	35.004		35.014	0.012
35	12:29:33	35.001		34.964	0.002
<b>ts = 0:00:29</b>		<b>Mp = 8.125%</b>		<b>ess ≤ 2 %</b>	

*Nota: Parámetros PID: Kp= 6.03, Ti= 0.74 s, Td=0.18 s*

**Tabla 37***Mediciones con el controlador PID autosintonizado para setpoint 40 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable ( $e_{ss}$ )
1	12:32:10	11.997	40.000	11.994	0.023
2	12:32:25	20.374		100.00	49.065
3	12:32:25	22.985		88.822	42.537
4	12:32:25	32.237		84.672	19.406
5	12:32:25	34.093		67.484	14.768
6	12:32:26	39.572		63.494	1.069
7	12:32:26	40.488		50.005	1.219
8	12:32:26	42.669		47.434	6.671
9	12:32:26	42.907		39.775	7.268
10	12:32:27	43.056		38.477	7.639
11	12:32:27	42.951		35.681	7.377
12	12:32:27	42.231		35.464	5.577
13	12:32:27	42.014		35.445	5.035
14	12:32:28	41.178		35.648	2.946
15	12:32:28	40.990		36.791	2.476
16	12:32:28	40.375		37.116	1.939
17	12:32:28	40.260		38.380	1.649
18	12:32:29	39.931		38.663	1.174
19	12:32:29	39.880		39.600	1.300
20	12:32:29	39.771		39.651	1.571
21	12:32:29	39.768		40.085	0.581
22	12:32:30	39.786		40.291	2.535
23	12:32:30	39.793		40.322	2.517
24	12:32:30	39.858		40.379	2.354
25	12:32:30	39.873		40.362	2.318
26	12:32:31	39.931		40.280	1.174
27	12:32:31	39.945		40.214	1.137
28	12:32:31	39.985		40.177	1.038
29	12:32:32	39.992		40.093	1.020
30	12:32:33	40.010		40.060	1.025
31	12:32:35	40.014		40.024	0.034
32	12:32:36	40.017		40.010	0.043
33	12:32:38	40.014		40.018	1.034
34	12:32:40	40.010		39.967	0.025
35	12:32:41	40.007		39.953	0.016
<b>ts = 0:00:31</b>		<b>Mp = 7.639 %</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID:  $K_p= 6.03$ ,  $T_i= 0.74$  s,  $T_d=0.18$  s*



**Tabla 38***Mediciones con el controlador PID autosintonizado para setpoint 50 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable ( $e_{ss}$ )
1	12:35:24	11.997	50	11.993	0.023
2	12:35:39	14.884		100.00	75.998
3	12:35:39	25.510		97.523	70.233
4	12:35:39	27.951		85.136	48.980
5	12:35:39	36.943		81.520	44.097
6	12:35:40	39.012		67.913	26.114
7	12:35:40	45.768		64.893	21.976
8	12:35:40	47.059		55.201	8.464
9	12:35:40	50.687		53.383	5.881
10	12:35:41	51.251		48.373	1.374
11	12:35:41	52.449		47.711	2.503
12	12:35:41	52.539		46.099	4.897
13	12:35:41	52.373		46.103	5.078
14	12:35:42	52.239		46.763	4.745
15	12:35:42	51.400		46.972	4.478
16	12:35:42	51.226		48.047	2.799
17	12:35:43	50.608		48.256	2.452
18	12:35:43	50.481		49.138	1.215
19	12:35:43	50.094		49.427	0.962
20	12:35:43	50.029		49.978	0.188
21	12:35:44	49.859		49.968	0.058
22	12:35:44	49.834		50.260	0.282
23	12:35:44	49.805		50.245	0.333
24	12:35:44	49.812		50.323	0.391
25	12:35:45	49.848		50.254	0.376
26	12:35:46	49.863		50.245	1.304
27	12:35:47	49.913		50.157	0.275
28	12:35:48	49.928		50.131	0.174
29	12:35:51	49.967		50.106	0.145
30	12:35:52	49.978		50.005	0.065
31	12:35:54	50.000		50.033	0.043
32	12:35:55	50.004		49.986	1.000
33	12:35:56	50.011		50.039	0.007
34	12:35:57	50.014		49.960	0.022
35	12:35:58	50.011		50.017	0.029
<b>ts = 0:00:34</b>		<b>Mp = 5.078%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID:  $K_p= 6.03$ ,  $T_i= 0.74$  s,  $T_d=0.18$  s*

**Tabla 39***Mediciones con el controlador PID autosintonizado para setpoint 60 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (e <sub>ss</sub> )
1	12:39:53	12.001	60	12.058	0.007
2	12:40:08	14.884		100.00	80.005
3	12:40:08	25.510		92.505	75.194
4	12:40:08	27.951		89.895	57.483
5	12:40:09	36.943		78.660	53.414
6	12:40:09	39.012		76.148	38.428
7	12:40:09	46.625		67.149	34.980
8	12:40:09	48.376		65.316	22.291
9	12:40:10	54.539		60.107	19.373
10	12:40:10	55.783		59.358	9.101
11	12:40:10	59.502		57.316	7.028
12	12:40:10	60.127		57.078	0.831
13	12:40:11	61.661		56.994	0.212
14	12:40:11	61.842		57.161	2.768
15	12:40:11	62.001		57.830	3.070
16	12:40:11	61.936		58.072	3.335
17	12:40:12	61.491		58.890	3.226
18	12:40:12	61.353		59.084	2.485
19	12:40:12	60.807		59.664	2.256
20	12:40:12	60.681		59.791	1.345
21	12:40:13	60.272		60.115	1.134
22	12:40:13	60.192		60.078	0.453
23	12:40:13	59.968		60.227	0.321
24	12:40:13	59.932		60.232	1.053
25	12:40:15	59.856		60.191	0.113
26	12:40:16	59.849		60.220	0.240
27	12:40:18	59.856		60.111	0.252
28	12:40:20	59.867		60.135	0.240
29	12:40:22	59.903		60.039	0.222
30	12:40:23	59.914		60.073	0.162
31	12:40:25	59.954		60.013	0.143
32	12:40:26	59.961		60.003	1.077
33	12:40:27	59.990		59.928	0.065
34	12:40:28	59.993		60.024	0.017
35	12:40:30	60.004		59.921	0.011
<b>ts = 0:00:37</b>		<b>Mp = 3.335%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID: Kp= 6.03, Ti= 0.74 s, Td=0.18 s*

**Tabla 40***Mediciones con el controlador PID autosintonizado para setpoint 70 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (e <sub>ss</sub> )
1	12:42:57	12.001	70	12.057	0.007
2	12:43:12	14.884		100.00	78.738
3	12:43:12	17.676		97.906	74.749
4	12:43:12	27.951		89.845	60.069
5	12:43:12	30.313		87.569	56.695
6	12:43:13	39.012		79.468	44.269
7	12:43:13	41.012		77.765	41.411
8	12:43:13	48.376		72.273	30.891
9	12:43:13	50.069		71.225	28.473
10	12:43:14	56.301		68.596	19.571
11	12:43:14	57.733		68.233	17.524
12	12:43:14	63.010		67.593	9.986
13	12:43:14	64.222		67.551	8.255
14	12:43:15	68.063		67.902	2.767
15	12:43:15	68.779		68.151	1.744
16	12:43:15	70.721		68.727	1.030
17	12:43:15	71.007		68.930	1.438
18	12:43:16	71.560		69.473	2.229
19	12:43:16	71.586		69.602	2.265
20	12:43:16	71.398		69.992	1.997
21	12:43:16	71.307		69.986	1.867
22	12:43:17	70.880		70.197	1.258
23	12:43:17	70.775		70.120	1.108
24	12:43:17	70.392		70.244	0.560
25	12:43:19	70.309		70.135	0.441
26	12:43:20	70.067		70.118	0.095
27	12:43:22	70.023		70.150	0.033
28	12:43:23	69.915		70.119	0.122
29	12:43:26	69.900		70.056	1.043
30	12:43:28	69.878		70.030	0.174
31	12:43:30	69.882		70.023	0.168
32	12:43:33	69.904		69.995	1.137
33	12:43:34	69.915		69.987	0.122
34	12:43:35	69.944		70.027	0.081
35	12:43:36	69.954		69.983	0.065
<b>ts = 0:00:39</b>		<b>Mp = 2.265%</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID: Kp= 6.03, Ti= 0.74 s, Td=0.18 s*

**Tabla 41***Mediciones con el controlador PID autosintonizado para setpoint 80 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable ( $e_{ss}$ )
1	12:46:33	11.910	80	11.957	0.746
2	12:46:47	22.985		100.00	68.113
3	12:46:47	25.510		98.778	56.489
4	12:46:48	34.809		92.458	53.821
5	12:46:48	36.943		90.828	43.979
6	12:46:48	44.817		85.307	41.718
7	12:46:48	46.625		84.167	33.390
8	12:46:49	53.288		80.712	31.478
9	12:46:49	54.818		80.144	24.425
10	12:46:49	60.460		78.645	22.806
11	12:46:49	61.755		78.480	16.838
12	12:46:50	66.529		78.212	15.469
13	12:46:50	67.625		78.287	10.414
14	12:46:50	71.669		78.611	9.257
15	12:46:50	72.595		78.810	4.980
16	12:46:51	76.016		79.229	3.994
17	12:46:51	76.805		79.355	0.947
18	12:46:51	79.243		79.756	0.404
19	12:46:51	79.677		79.764	0.988
20	12:46:52	80.791		79.988	1.174
21	12:46:52	80.939		80.032	1.449
22	12:46:52	81.160		80.153	1.431
23	12:46:52	81.145		80.115	1.169
24	12:46:53	80.935		80.159	1.074
25	12:46:53	80.859		80.097	0.681
26	12:46:53	80.545		80.040	0.586
27	12:46:53	80.469		80.127	0.265
28	12:46:54	80.212		80.023	0.197
29	12:46:55	80.158		80.062	1.012
30	12:46:56	80.009		80.021	0.020
31	12:46:58	79.984		80.016	0.097
32	12:47:02	79.923		79.998	1.101
33	12:47:05	79.919		79.993	0.106
34	12:47:10	79.915		79.974	0.101
35	12:47:15	79.919		80.034	0.079
<b>ts = 0:00:42</b>		<b>Mp = 1.449%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID:  $K_p= 6.03$ ,  $T_i= 0.74$  s,  $T_d=0.18$  s*

**Tabla 42***Mediciones con el controlador PID autosintonizado para setpoint 90 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable ( $e_{ss}$ )
1	12:51:59	11.997	90	12.058	0.007
2	12:52:06	34.809		100.00	0.007
3	12:52:06	36.943		99.996	0.023
4	12:52:07	44.817		96.804	0.007
5	12:52:07	46.625		95.986	0.023
6	12:52:07	53.288		93.164	0.007
7	12:52:07	54.818		92.513	86.666
8	12:52:08	60.460		90.680	74.461
9	12:52:08	61.755		90.315	71.656
10	12:52:08	66.529		89.435	61.323
11	12:52:08	67.625		89.364	58.952
12	12:52:09	71.669		89.149	50.203
13	12:52:09	72.595		89.111	48.194
14	12:52:09	76.016		89.329	40.791
15	12:52:09	76.805		89.319	39.091
16	12:52:10	79.698		89.592	32.822
17	12:52:10	80.364		89.636	31.383
18	12:52:10	82.816		89.845	26.079
19	12:52:10	83.380		89.856	24.861
20	12:52:11	85.453		90.001	20.368
21	12:52:11	85.930		89.972	19.339
22	12:52:11	87.688		90.050	15.537
23	12:52:11	88.090		90.052	14.661
24	12:52:12	89.413		90.060	11.446
25	12:52:15	89.656		90.050	10.707
26	12:52:16	90.303		90.079	7.982
27	12:52:18	90.397		90.020	7.355
28	12:52:20	90.564		90.044	5.052
29	12:52:33	90.567		89.980	4.522
30	12:52:38	90.488		90.011	2.569
31	12:52:40	90.451		90.008	2.123
32	12:52:43	90.299		89.999	0.652
33	12:52:45	90.263		89.996	1.383
34	12:52:47	90.126		89.987	0.337
35	12:52:48	90.101		89.985	0.441
<b>ts= 0:00:49</b>		<b>Mp = 0.630 %</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID:  $K_p= 6.03$ ,  $T_i= 0.74$  s,  $T_d=0.18$  s*

**Tabla 43***Mediciones con el controlador PID autosintonizado para setpoint 96 °C*

Nº medición	Hora	Entrada °C	Setpoint °C	Salida %	Error de estado estable (e <sub>ss</sub> )
1	10:38:35	1.002	96	0.948	0.188
2	10:38:40	41.923		100.00	56.330
3	10:38:40	43.826		99.951	54.348
4	10:38:40	50.839		99.590	47.043
5	10:38:40	52.449		98.365	45.366
6	10:38:41	58.384		98.132	39.183
7	10:38:41	59.751		96.970	37.759
8	10:38:41	64.775		96.828	32.526
9	10:38:41	65.929		96.101	31.324
10	10:38:42	70.182		96.022	26.893
11	10:38:42	71.159		95.699	25.876
12	10:38:42	74.758		95.754	22.127
13	10:38:42	75.586		95.713	21.265
14	10:38:43	78.635		95.626	18.089
15	10:38:43	79.333		95.747	17.361
16	10:38:43	81.916		95.725	14.671
17	10:38:43	82.509		95.887	14.053
18	10:38:44	84.690		95.873	11.782
19	10:38:44	85.192		95.974	11.258
20	10:38:44	87.041		95.947	9.333
21	10:38:44	87.467		95.998	8.888
22	10:38:45	89.030		96.010	7.260
23	10:38:45	89.392		96.031	6.884
24	10:38:45	90.715		96.040	5.505
25	10:38:45	91.019		96.045	5.188
26	10:38:46	92.140		95.988	4.020
27	10:38:46	92.397		96.076	3.753
28	10:39:24	96.123		96.003	2.003
29	10:39:24	96.000		96.004	2.000
30	10:39:25	96.003		95.940	1.003
31	10:39:25	96.000		96.002	0.000
32	10:39:26	96.003		96.003	0.003
33	10:39:26	96.000		96.004	1.000
34	10:39:27	96.003		96.004	0.003
35	10:39:29	96.000		96.002	0.000
<b>ts = 0:00:54</b>		<b>Mp = 0.128%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID: Kp= 6.03, Ti= 0.74 s, Td=0.18 s*

### C. Muestras tomadas para el controlador PID no sintonizado para la variable nivel en tanque de 3000 mm de altura

**Tabla 44**

*Mediciones con el controlador PID sin sintonizar para setpoint 250 mm*

N° medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	16:52:50	57.50868	250	-13.18	99.60938
2	16:52:51	80.29514		100.00	63.02083
3	16:52:51	106.6623		100.00	67.88194
4	16:52:51	113.3898		100.00	57.33507
5	16:52:51	159.6137		100.00	54.64409
6	16:52:53	263.6719		100.00	6.510413
7	16:52:53	267.7951		100.00	5.46875
8	16:52:53	295.0304		100.00	7.118054
9	16:53:01	237.0877		-10.06	4.644092
10	16:53:01	236.1111		-79.81	5.164929
11	16:53:02	235.8941		-100	5.555554
12	16:53:02	236.4366		-64.73	5.642358
13	16:53:02	236.5452		-36.5	5.425342
14	16:53:02	236.2196		1.2255	5.381939
15	16:53:03	235.5686		1.2656	5.512152
16	16:53:03	235.6771		0.894	5.772564
17	16:53:03	236.2196		-0.285	5.685761
18	16:53:03	235.8941		-0.459	5.512152
19	16:53:04	235.5686		-0.183	5.642358
20	16:53:04	235.6771		0.2461	5.772564
21	16:53:08	239.2578		0.5521	4.340277
22	16:53:08	239.9089		0.7697	4.296875
23	16:53:09	240.0174		0.7685	4.036456
24	16:53:09	240.4514		0.1544	3.862842
25	16:53:09	240.8854		-0.183	3.81944
26	16:53:10	241.1024		-0.314	3.645831
27	16:53:21	244.3577		0.1625	2.213538
28	16:53:22	244.2491		0.3595	2.25694
29	16:53:22	244.3577		0.5108	2.300342
30	16:53:23	244.4662		0.4658	2.25694
31	16:53:23	244.5747		0.3285	2.213538
32	16:53:24	244.6832		0.0485	2.170135
33	16:53:24	244.7917		-0.14	2.126733
34	16:53:24	245.0087		-0.052	2.039929
35	16:53:25	245.1172		0.1581	1.996527
<b>ts = 0:00:35</b>		<b>Mp = 18.012%</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID por defecto:  $K_p=1.0$ ,  $T_i=20.0$  s,  $T_d=0.1$  s*

**Tabla 45***Mediciones con el controlador PID sin sintonizar para setpoint 500 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	9:11:42	1.085	500	-25.950	8.507
2	9:11:58	92.990		100.00	86.719
3	9:12:09	472.765		100.00	5.382
4	9:12:09	473.416		100.00	5.447
5	9:12:10	479.384		100.00	5.317
6	9:12:10	477.648		100.00	4.123
7	9:12:14	486.545		100.00	2.821
8	9:12:14	486.003		78.144	2.691
9	9:12:14	487.413		5.035	2.799
10	9:12:15	486.979		-65.047	2.517
11	9:12:15	488.390		-100.00	2.604
12	9:12:15	488.064		-61.740	2.322
13	9:12:16	485.894		-19.230	2.387
14	9:12:16	486.437		-0.181	2.821
15	9:12:16	486.654		4.095	2.713
16	9:12:16	486.545		4.296	2.669
17	9:12:17	485.352		5.107	2.691
18	9:12:17	485.243		2.882	2.930
19	9:12:17	484.918		0.394	2.951
20	9:12:17	485.677		-2.415	3.016
21	9:12:19	485.569		2.232	2.908
22	9:12:19	485.135		2.046	2.886
23	9:12:19	485.460		1.233	2.973
24	9:12:20	485.894		0.084	2.908
25	9:12:20	548.000		-0.783	2.821
26	9:12:20	598.000		-0.861	2.799
27	9:12:20	520.000		0.051	2.734
28	9:12:21	500.000		1.457	2.561
29	9:12:21	488.064		1.152	2.452
30	9:12:21	488.498		0.550	2.387
31	9:12:22	488.824		-0.148	2.300
32	9:12:22	489.366		-0.566	2.235
33	9:12:22	489.475		-0.400	2.127
34	9:12:23	490.126		0.200	2.105
35	9:12:23	490.343		0.608	1.975
<b>ts = 0:00:41</b>		<b>Mp = 19.600%</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID por defecto:  $K_p = 1.0$ ,  $T_i = 20.0$  s,  $T_d = 0.1$  s*



**Tabla 46***Mediciones con el controlador PID sin sintonizar para setpoint 1000 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	9:28:19	0.977	1000	-40.263	2.344
2	9:28:29	151.910		100.00	89.301
3	9:28:29	159.722		100.00	84.809
4	9:28:29	185.004		78.000	84.028
5	9:28:29	184.679		75.000	81.500
6	9:28:38	812.609		-91.586	21.691
7	9:28:38	819.987		-55.959	18.739
8	9:28:38	850.152		-5.878	18.001
9	9:28:38	857.530		13.474	14.985
10	9:28:39	887.153		14.905	14.247
11	9:28:39	894.531		17.019	11.285
12	9:28:39	924.696		8.550	10.547
13	9:28:39	932.075		-1.094	7.530
14	9:28:40	961.697		-10.851	6.793
15	9:28:40	998.589		-4.014	3.092
16	9:28:40	1006.619		2.570	0.141
17	9:28:41	1032.552		4.523	0.662
18	9:28:41	1031.576		5.381	3.255
19	9:28:41	1053.277		4.493	3.158
20	9:28:41	1061.089		2.187	5.328
21	9:28:42	1031.141		-0.615	6.109
22	9:28:42	1043.728		-2.580	3.114
23	9:28:42	1022.895		-2.403	4.373
24	9:28:42	1025.716		-0.031	2.290
25	9:28:43	1004.015		1.641	2.572
26	9:28:43	996.528		2.167	0.401
27	9:28:43	964.952		2.822	0.347
28	9:28:54	976.020		0.146	2.496
29	9:28:54	976.888		1.017	2.398
30	9:28:54	977.865		1.418	2.311
31	9:28:55	978.950		1.713	2.214
32	9:28:55	979.275		1.373	2.105
33	9:28:55	979.818		0.770	2.072
34	9:28:55	980.903		-0.030	2.018
35	9:29:02	982.964		-0.153	1.714
<b>ts = 0:00:43</b>		<b>Mp = 6.109%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID por defecto:  $K_p = 1.0$ ,  $T_i = 20.0$  s,  $T_d = 0.1$  s*

**Tabla 47***Mediciones con el controlador PID sin sintonizar para setpoint 1250 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	9:42:47	1.953	1250.000	-23.449	99.922
2	9:42:50	199.870		100.00	85.425
3	9:42:50	209.635		76.779	84.010
4	9:42:50	239.692		76.779	83.229
5	9:42:50	251.845		76.779	80.825
6	9:42:51	276.150		75.000	79.852
7	9:43:02	1101.997		75.000	12.431
8	9:43:02	1132.270		75.000	11.840
9	9:43:02	1139.648		74.000	9.418
10	9:43:03	1169.162		74.000	8.828
11	9:43:03	1176.541		74.000	6.467
12	9:43:03	1206.163		74.000	5.877
13	9:43:03	1213.542		-54.412	3.507
14	9:43:04	1243.056		-92.340	2.917
15	9:43:04	1250.434		-50.477	0.556
16	9:43:04	1280.599		4.093	0.035
17	9:43:04	1287.977		23.099	2.448
18	9:43:05	1306.098		20.336	3.038
19	9:43:05	1314.779		19.245	4.488
20	9:43:07	1226.997		2.436	2.170
21	9:43:07	1213.433		-0.720	1.840
22	9:43:08	1208.876		-3.965	2.925
23	9:43:08	1200.955		-3.784	3.290
24	9:43:08	1202.908		-0.512	3.924
25	9:43:08	1199.544		1.718	3.767
26	9:43:09	1212.348		2.059	4.036
27	9:43:09	1208.225		2.460	3.012
28	9:43:09	1205.078		1.658	3.342
29	9:43:09	1206.163		0.587	3.594
30	9:43:10	1211.480		-0.711	3.507
31	9:43:12	1213.542		-1.240	3.082
32	9:43:15	1220.052		-0.661	2.917
33	9:43:18	1224.718		0.394	2.396
34	9:43:21	1231.011		0.949	2.023
35	9:43:31	1233.290		1.173	1.519
<b>ts = 0:00:44</b>		<b>Mp = 5.182%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID por defecto:  $K_p = 1.0$ ,  $T_i = 20.0$  s,  $T_d = 0.1$  s*

**Tabla 48***Mediciones con el controlador PID sin sintonizar para setpoint 1500 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	16:15:27	11.610	1500	-4.788	16.102
2	16:15:29	11.176		100.00	11.762
3	16:15:49	1174.045		100.00	21.730
4	16:15:50	1315.104		100.00	12.326
5	16:15:51	1323.134		100.00	11.791
6	16:15:51	1358.833		100.00	9.411
7	16:15:51	1388.997		100.00	7.400
8	16:15:52	1397.678		59.662	6.821
9	16:15:52	1428.494		-10.386	4.767
10	16:15:52	1466.037		-95.836	2.264
11	16:15:53	1473.416		-1.272	1.772
12	16:15:53	1504.232		39.433	0.282
13	16:15:53	1511.610		7.232	0.774
14	16:15:53	1541.775		2.675	2.785
15	16:15:54	1550.347		-31.741	3.356
16	16:15:54	1580.512		-18.540	5.367
17	16:15:54	1587.999		12.620	5.867
18	16:15:54	1604.601		20.020	6.973
19	16:15:55	1595.378		13.712	6.359
20	16:15:55	1606.771		1.338	7.118
21	16:15:55	1622.287		-14.232	8.152
22	16:15:55	1609.158		-17.987	7.277
23	16:15:56	1520.508		9.911	1.367
24	16:15:56	1541.341		4.373	2.756
25	16:15:56	1534.071		3.982	2.271
26	16:15:57	1528.646		-4.846	1.910
27	16:15:57	1481.120		-5.340	1.259
28	16:15:57	1467.990		-2.299	2.134
29	16:15:57	1459.852		1.771	2.677
30	16:15:58	1506.293		2.036	0.420
31	16:15:58	1512.912		1.169	0.861
32	16:15:58	1491.862		-0.384	0.543
33	16:15:58	1473.633		-1.659	1.758
34	16:16:09	1474.501		-1.379	1.700
35	16:16:14	1495.117		0.537	0.326
<b>ts = 0:00:47</b>		<b>Mp = 8.152%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID por defecto:  $K_p = 1.0$ ,  $T_i = 20.0$  s,  $T_d = 0.1$  s*

**Tabla 49***Mediciones con el controlador PID sin sintonizar para setpoint 1750 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	10:38:18	2.821	1750	-4.654	182.118
2	10:38:20	2.496		100.00	149.566
3	10:38:24	67.708		100.00	97.266
4	10:38:24	94.076		100.00	96.131
5	10:38:24	96.354		100.00	94.624
6	10:38:24	126.736		100.00	94.494
7	10:38:48	1839.084		100.00	4.805
8	10:38:48	1865.451		100.00	5.091
9	10:38:48	1870.551		100.00	6.597
10	10:38:48	1839.193		100.00	6.889
11	10:38:49	1817.383		100.00	5.097
12	10:38:52	1694.336		87.586	3.075
13	10:38:52	1701.063		13.931	3.181
14	10:38:52	1711.589		-61.339	2.796
15	10:38:53	1715.278		-100.00	2.195
16	10:38:53	1729.492		-41.901	1.984
17	10:38:53	1738.932		34.924	1.172
18	10:38:53	1748.155		54.519	0.632
19	10:38:54	1757.053		44.252	0.105
20	10:38:54	1774.414		12.071	0.403
21	10:38:54	1782.769		-28.994	1.395
22	10:38:54	1785.265		-50.260	1.873
23	10:38:55	1792.101		-23.363	2.015
24	10:38:55	1803.060		16.197	2.406
25	10:38:56	1780.057		-25.334	1.606
26	10:38:56	1759.766		-13.761	1.718
27	10:38:57	1749.349		6.743	0.558
28	10:38:57	1734.484		14.124	0.037
29	10:38:57	1724.718		11.622	0.887
30	10:38:57	1729.167		4.939	1.445
31	10:38:58	1728.516		-5.909	1.190
32	10:38:58	1719.944		-12.270	1.228
33	10:39:02	1709.201		-7.778	1.718
34	10:39:05	1720.812		2.488	2.331
35	10:39:10	1722.439		6.731	1.668
<b>ts = 0:00:52</b>		<b>Mp = 6.889%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID por defecto:  $K_p = 1.0$ ,  $T_i = 20.0$  s,  $T_d = 0.1$  s*

**Tabla 50***Mediciones con el controlador PID sin sintonizar para setpoint 2000 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	10:50:18	2.279	2000	-24.76	99.951
2	10:50:19	66.298		100.00	99.121
3	10:50:19	93.099		100.00	96.685
4	10:50:19	94.727		100.00	95.345
5	10:50:19	101.454		100.00	95.264
6	10:50:20	135.525		100.00	94.927
7	10:50:20	150.608		100.00	93.224
8	10:50:20	184.462		100.00	92.470
9	10:50:20	184.787		100.00	90.777
10	10:50:35	1285.699		96.740	37.196
11	10:50:41	1695.204		24.682	16.748
12	10:50:41	1702.583		-50.293	15.240
13	10:50:41	1732.096		-100.00	14.871
14	10:50:45	1992.622		-48.687	1.845
15	10:50:45	2000.000		37.637	0.369
16	10:50:45	2029.514		64.470	0.000
17	10:50:45	2036.458		54.103	1.476
18	10:50:46	2064.996		19.359	1.823
19	10:50:46	2071.615		-29.240	3.250
20	10:50:46	2099.392		-57.134	3.581
21	10:50:46	2086.697		-29.712	4.970
22	10:50:47	2052.951		20.154	4.335
23	10:50:47	2025.933		36.952	2.648
24	10:50:47	2048.503		31.477	1.297
25	10:50:51	1971.571		13.664	1.557
26	10:50:52	1992.405		-14.371	1.421
27	10:50:56	1963.325		-31.489	1.877
28	10:50:56	1941.515		-19.600	1.834
29	10:50:56	1943.902		9.161	2.924
30	10:50:57	1954.861		21.334	2.805
31	10:50:57	1968.533		17.898	2.257
32	10:50:57	1954.861		8.813	1.573
33	10:51:05	1956.163		-7.484	2.257
34	10:51:10	1964.735		-17.421	2.192
35	10:51:14	1969.944		-12.039	1.763
<b>ts = 0:00:56</b>		<b>Mp = 4.970%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID por defecto:  $K_p = 1.0$ ,  $T_i = 20.0$  s,  $T_d = 0.1$  s*

**Tabla 51***Mediciones con el controlador PID sin sintonizar para setpoint 2250 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	11:12:43	55.013	2250	-28.044	99.957
2	11:12:45	158.095		100.00	92.940
3	11:13:11	2079.427		100.00	8.893
4	11:13:11	2087.457		100.00	7.581
5	11:13:11	2116.970		100.00	7.224
6	11:13:11	2124.349		100.00	5.912
7	11:13:12	2153.971		100.00	5.584
8	11:13:12	2161.350		100.00	4.268
9	11:13:12	2191.515		100.00	3.940
10	11:13:12	2198.893		100.00	2.599
11	11:13:13	2228.407		100.00	2.271
12	11:13:13	2235.894		100.00	0.960
13	11:13:13	2266.059		100.00	0.627
14	11:13:13	2273.438		100.00	0.714
15	11:13:14	2302.951		100.00	1.042
16	11:13:14	2310.330		100.00	2.353
17	11:13:14	2291.341		45.705	2.681
18	11:13:14	2302.734		45.705	1.837
19	11:13:15	2300.890		-28.620	2.344
20	11:13:15	2310.004		-89.399	2.262
21	11:13:16	2212.023		33.493	1.403
22	11:13:16	2201.823		-20.417	1.688
23	11:13:17	2156.576		-55.536	2.141
24	11:13:17	2158.746		-42.377	4.152
25	11:13:20	2243.490		-7.265	0.251
26	11:13:20	2255.317		-21.023	0.289
27	11:13:20	2256.836		-17.129	0.236
28	11:13:21	2278.754		4.866	0.304
29	11:13:21	2285.156		16.848	1.278
30	11:13:24	2202.365		6.715	1.943
31	11:13:25	2207.031		5.735	2.117
32	11:13:26	2195.855		3.744	1.910
33	11:13:28	2199.653		-1.440	2.406
34	11:13:35	2200.521		-5.020	2.238
35	11:13:42	2209.527		0.896	1.394
<b>ts = 0:00:59</b>		<b>Mp = 2.681%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID por defecto:  $K_p = 1.0$ ,  $T_i = 20.0$  s,  $T_d = 0.1$  s*

**Tabla 52***Mediciones con el controlador PID sin sintonizar para setpoint 2500 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	11:24:50	48.503	2500	-28.147	99.961
2	11:24:51	115.017		100.00	96.285
3	11:24:51	125.434		100.00	95.399
4	11:24:52	157.878		100.00	94.983
5	11:24:52	163.737		100.00	93.685
6	11:24:52	191.189		100.00	93.451
7	11:25:05	1132.270		100.00	55.030
8	11:25:05	1161.784		100.00	54.709
9	11:25:08	1390.300		100.00	44.583
10	11:25:12	1688.368		100.00	32.786
11	11:25:13	1717.882		100.00	32.465
12	11:25:23	2504.340		100.00	0.122
13	11:25:24	2534.505		100.00	0.174
14	11:25:24	2541.992		100.00	1.380
15	11:25:24	2557.834		100.00	1.680
16	11:25:24	2566.949		100.00	2.313
17	11:25:25	2586.046		100.00	2.678
18	11:25:25	2590.603		72.758	3.442
19	11:25:25	2536.024		72.758	3.624
20	11:25:25	2543.620		72.758	1.441
21	11:25:26	2493.490		-71.643	0.061
22	11:25:27	2409.722		74.591	1.866
23	11:25:27	2403.212		64.737	3.611
24	11:25:28	2414.714		-50.023	3.533
25	11:25:32	2533.312		15.790	0.760
26	11:25:34	2478.190		-18.698	0.217
27	11:25:35	2470.812		1.361	0.872
28	11:25:36	2438.368		16.058	1.168
29	11:25:38	2443.034		-2.461	1.780
30	11:25:48	2446.181		-12.428	2.205
31	11:25:42	2442.817		1.292	2.153
32	11:25:43	2442.708		10.928	2.287
33	11:25:44	2442.491		6.779	2.183
34	11:25:47	2463.867		-1.907	2.300
35	11:25:53	2465.386		-7.948	1.445
<b>ts = 0:01:03</b>		<b>Mp = 3.624%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID por defecto:  $K_p = 1.0$ ,  $T_i = 20.0$  s,  $T_d = 0.1$  s*

**Tabla 53***Mediciones con el controlador PID sin sintonizar para setpoint 3000 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	11:41:41	48.394	3000	-55.594	99.967
2	11:41:42	84.418		100.00	98.441
3	11:41:42	92.773		100.00	97.186
4	11:41:42	115.668		100.00	96.908
5	11:42:10	2165.039		100.00	28.838
6	11:42:10	2172.417		100.00	27.832
7	11:42:10	2202.040		100.00	27.586
8	11:42:11	2246.311		100.00	25.369
9	11:42:17	2683.051		40.024	11.552
10	11:42:17	2690.430		19.214	10.565
11	11:42:17	2719.944		-7.940	10.319
12	11:42:17	2727.322		-23.853	9.335
13	11:42:18	2757.487		-18.338	9.089
14	11:42:18	2764.865		11.170	8.084
15	11:42:18	2794.488		27.852	7.838
16	11:42:18	2801.866		25.165	6.850
17	11:42:20	2941.732		17.249	2.890
18	11:42:20	2948.025		12.496	1.942
19	11:42:21	2970.269		0.505	1.732
20	11:42:21	2974.935		-8.689	0.991
21	11:42:21	2990.451		-9.033	0.835
22	11:42:21	2993.598		2.543	0.318
23	11:42:22	2998.915		11.426	0.213
24	11:42:22	3010.000		10.560	0.036
25	11:42:22	3000.000		8.275	0.000
26	11:42:22	2991.211		0.715	0.412
27	11:42:23	2995.334		-5.853	0.369
28	11:42:24	2966.580		0.956	0.156
29	11:42:25	2967.231		6.922	1.114
30	11:42:27	2923.611		-0.322	2.431
31	11:42:33	2926.541		1.248	2.546
32	11:42:36	2933.919		1.373	2.568
33	11:42:46	2933.377		0.378	2.203
34	11:42:47	2946.181		-0.495	2.221
35	11:42:51	2945.855		-0.848	1.794
<b>ts = 0:01:10</b>		<b>Mp = 0.333%</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID por defecto:  $K_p = 1.0$ ,  $T_i = 20.0$  s,  $T_d = 0.1$  s*



## D. Muestras tomadas para el controlador PID autosintonizado para la variable nivel en tanque de 3000 mm de altura

**Tabla 54**

*Mediciones con el controlador PID autosintonizado para setpoint 250 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	16:52:50	57.50868	250	-13.18	99.60938
2	16:52:51	80.29514		100.00	63.02083
3	16:52:51	106.6623		100.00	67.88194
4	16:52:51	113.3898		100.00	57.33507
5	16:52:51	159.6137		100.00	54.64409
6	16:52:53	263.6719		100.00	6.510413
7	16:52:53	267.7951		100.00	5.46875
8	16:52:53	295.0304		100.00	7.118054
9	16:53:01	237.0877		-10.06	4.644092
10	16:53:01	236.1111		-79.81	5.164929
11	16:53:02	235.8941		-100	5.555554
12	16:53:02	236.4366		-64.73	5.642358
13	16:53:02	236.5452		-36.5	5.425342
14	16:53:02	236.2196		1.2255	5.381939
15	16:53:03	235.5686		1.2656	5.512152
16	16:53:03	235.6771		0.894	5.772564
17	16:53:03	236.2196		-0.285	5.685761
18	16:53:03	235.8941		-0.459	5.512152
19	16:53:04	235.5686		-0.183	5.642358
20	16:53:04	235.6771		0.2461	5.772564
21	16:53:08	239.2578		0.5521	4.340277
22	16:53:08	239.9089		0.7697	4.296875
23	16:53:09	240.0174		0.7685	4.036456
24	16:53:09	240.4514		0.1544	3.862842
25	16:53:09	240.8854		-0.183	3.81944
26	16:53:10	241.1024		-0.314	3.645831
27	16:53:21	244.3577		0.1625	2.213538
28	16:53:22	244.2491		0.3595	2.25694
29	16:53:22	244.3577		0.5108	2.300342
30	16:53:23	244.4662		0.4658	2.25694
31	16:53:23	244.5747		0.3285	2.213538
32	16:53:24	244.6832		0.0485	2.170135
33	16:53:24	244.7917		-0.14	2.126733
34	16:53:24	245.0087		-0.052	2.039929
35	16:53:25	245.1172		0.1581	1.996527
<b>ts = 0:00:35</b>		<b>Mp = 18.012%</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID:  $K_p=0.80$ ,  $T_i=2.74$  s,  $T_d=0.68$  s*

**Tabla 55***Mediciones con el controlador PID autosintonizado para setpoint 500 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	17:21:40	10.417	500	-2.158	4.167
2	17:21:44	100.803		100.00	87.348
3	17:21:44	88.542		63.571	79.839
4	17:21:44	114.909		-9.684	82.292
5	17:21:45	121.745		-94.372	77.018
6	17:21:45	168.077		-100.00	75.651
7	17:21:45	165.907		-70.120	66.385
8	17:21:45	191.081		-52.385	66.819
9	17:21:46	387.587		-42.981	23.915
10	17:21:46	417.209		-28.401	22.483
11	17:21:46	426.975		-10.796	16.558
12	17:21:46	456.597		3.546	14.605
13	17:21:47	462.999		12.985	8.681
14	17:21:47	490.234		23.097	7.400
15	17:21:47	498.372		55.172	1.953
16	17:21:47	528.537		66.009	0.326
17	17:21:48	536.458		54.723	5.707
18	17:21:48	566.081		51.120	7.292
19	17:21:48	573.134		-17.605	13.216
20	17:21:48	603.624		-65.200	14.627
21	17:21:49	611.003		-90.707	20.725
22	17:21:49	616.428		-49.291	22.201
23	17:21:49	629.232		-22.888	23.286
24	17:21:49	502.062		-20.836	0.347
25	17:21:54	501.628		-15.660	0.412
26	17:21:54	501.519		-3.525	2.326
27	17:21:55	500.434		6.366	0.304
28	17:21:55	500.760		11.690	2.087
29	17:21:56	500.651		16.412	0.152
30	17:22:02	500.434		14.142	0.130
31	17:22:02	500.109		6.777	0.087
32	17:22:05	499.783		-5.483	1.022
33	17:22:05	500.000		-14.279	1.043
34	17:22:06	499.783		-13.623	0.000
35	17:22:06	499.675		-6.297	0.043
<b>ts = 0:00:26</b>		<b>Mp = 25.846%</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID:  $K_p=0.80$ ,  $T_i=2.74$  s,  $T_d=0.68$  s*

**Tabla 56***Mediciones con el controlador PID autosintonizado para setpoint 1000 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	17:28:39	9.657	1000	1.121	3.429
2	17:28:39	52.517		100.00	93.457
3	17:28:39	198.893		45.744	82.650
4	17:28:39	277.344		-34.988	75.141
5	17:28:39	285.373		-100.00	72.266
6	17:28:39	315.864		-33.483	71.463
7	17:28:39	325.087		-0.168	68.414
8	17:28:39	399.523		-9.260	60.948
9	17:28:39	430.013		-13.391	60.048
10	17:28:39	437.283		1.158	56.999
11	17:28:39	464.193		13.967	56.272
12	17:28:39	539.822		16.431	49.110
13	17:28:39	546.875		18.553	46.018
14	17:28:39	576.497		7.554	45.313
15	17:28:39	583.876		-4.185	42.350
16	17:28:39	613.824		-16.982	41.612
17	17:28:39	696.289		-13.931	31.120
18	17:28:39	724.067		-0.499	30.371
19	17:28:39	731.445		3.056	27.593
20	17:28:39	843.425		-0.248	16.395
21	17:28:39	911.241		-0.247	11.892
22	17:28:39	918.620		-1.887	8.876
23	17:28:39	948.785		-0.004	8.138
24	17:28:39	956.163		1.733	5.122
25	17:28:39	1021.484		1.557	0.868
26	17:28:39	1096.571		1.737	6.641
27	17:28:39	1103.950		0.183	9.657
28	17:28:39	1091.905		-0.950	13.835
29	17:28:39	1106.771		-1.887	9.191
30	17:28:39	1067.925		-0.004	10.677
31	17:28:40	1035.156		1.733	4.460
32	17:28:40	1001.845		1.557	3.516
33	17:28:40	998.915		1.737	0.184
34	17:28:58	1004.449		0.183	0.109
35	17:29:05	1001.194		-0.950	0.445
<b>ts = 0:00:26</b>		<b>Mp = 10.677%</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID:  $K_p=0.80$ ,  $T_i=2.74$  s,  $T_d=0.68$  s*

**Tabla 57***Mediciones con el controlador PID autosintonizado para setpoint 1250 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	17:43:39	9.983	1250.000	0.102	0.174
2	17:43:40	144.097		100.00	91.293
3	17:43:40	158.203		76.179	88.472
4	17:43:40	189.562		2.840	87.344
5	17:43:40	194.770		-84.525	84.835
6	17:43:41	218.099		-100.00	84.418
7	17:43:41	218.316		-42.637	82.552
8	17:43:41	259.115		1.626	82.535
9	17:43:41	263.889		-5.473	79.271
10	17:43:42	294.813		-16.998	78.889
11	17:43:42	299.154		-6.908	76.415
12	17:43:42	329.644		11.558	76.068
13	17:43:42	336.806		17.115	73.628
14	17:43:43	368.381		18.392	73.056
15	17:43:43	374.458		9.126	70.530
16	17:43:43	404.188		-3.908	70.043
17	17:43:43	411.350		-15.921	67.665
18	17:43:44	442.057		-13.711	67.092
19	17:43:44	449.978		1.326	64.635
20	17:43:44	479.058		6.386	64.017
21	17:43:44	484.050		1.985	61.675
22	17:43:45	514.540		0.439	61.276
23	17:43:45	521.714		-4.644	58.837
24	17:43:45	551.975		-3.381	58.264
25	17:43:45	559.353		1.124	55.842
26	17:43:46	589.518		2.508	55.252
27	17:43:46	1238.607		1.579	52.839
28	17:43:57	1457.000		0.587	52.240
29	17:43:59	1238.607		1.985	1.415
30	17:43:59	1247.830		0.439	0.911
31	17:44:00	1253.038		-4.644	0.174
32	17:44:00	1265.625		-3.381	0.243
33	17:44:00	1224.935		1.124	1.250
34	17:44:00	1234.158		2.508	2.000
35	17:44:05	1242.622		1.579	1.267
<b>ts = 0:00:26</b>		<b>Mp = 16.560%</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID:  $K_p=0.80$ ,  $T_i=2.74$  s,  $T_d=0.68$  s*

**Tabla 58***Mediciones con el controlador PID autosintonizado para setpoint 1500 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	16:15:27	11.610	1500	-4.788	16.102
2	16:15:29	11.176		100.00	11.762
3	16:15:49	1174.045		100.00	21.730
4	16:15:50	1315.104		100.00	12.326
5	16:15:51	1323.134		100.00	11.791
6	16:15:51	1358.833		100.00	9.411
7	16:15:51	1388.997		100.00	7.400
8	16:15:52	1397.678		59.662	6.821
9	16:15:52	1428.494		-10.386	4.767
10	16:15:52	1466.037		-95.836	2.264
11	16:15:53	1473.416		-1.272	1.772
12	16:15:53	1504.232		39.433	0.282
13	16:15:53	1511.610		7.232	0.774
14	16:15:53	1541.775		2.675	2.785
15	16:15:54	1550.347		-31.741	3.356
16	16:15:54	1580.512		-18.540	5.367
17	16:15:54	1587.999		12.620	5.867
18	16:15:54	1604.601		20.020	6.973
19	16:15:55	1595.378		13.712	6.359
20	16:15:55	1606.771		1.338	7.118
21	16:15:55	1692.287		-14.232	8.152
22	16:15:55	1609.158		-17.987	7.277
23	16:15:56	1520.508		9.911	1.367
24	16:15:56	1541.341		4.373	2.756
25	16:15:56	1534.071		3.982	2.271
26	16:15:57	1528.646		-4.846	1.910
27	16:15:57	1481.120		-5.340	1.259
28	16:15:57	1467.990		-2.299	2.134
29	16:15:57	1459.852		1.771	2.677
30	16:15:58	1506.293		2.036	0.420
31	16:15:58	1512.912		1.169	0.861
32	16:15:58	1491.862		-0.384	0.543
33	16:15:58	1473.633		-1.659	1.758
34	16:15:59	1474.501		-1.379	1.700
35	16:15:59	1495.117		0.537	0.326
<b>ts = 0:00:32</b>		<b>Mp = 12.819%</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID:  $K_p=0.80$ ,  $T_i=2.74$  s,  $T_d=0.68$  s*

**Tabla 59***Mediciones con el controlador PID autosintonizado para setpoint 1750 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	18:52:15	9.983	1750	-10.584	0.174
2	18:52:15	1027.995		100.00	41.679
3	18:52:15	1614.909		100.00	9.406
4	18:52:15	1622.938		56.040	7.719
5	18:52:15	1650.608		-22.609	7.261
6	18:52:15	1657.986		-100.00	5.680
7	18:52:15	1687.609		-99.929	5.258
8	18:52:15	1695.638		7.692	3.565
9	18:52:15	1725.152		52.507	3.106
10	18:52:15	1732.530		13.661	1.420
11	18:52:15	1762.695		-0.096	0.998
12	18:52:15	1770.074		-45.124	0.725
13	18:52:15	1800.347		-30.610	1.147
14	18:52:15	1807.726		22.437	2.877
15	18:52:16	1837.240		35.802	3.299
16	18:52:16	1845.269		24.868	4.985
17	18:52:16	1874.783		3.058	5.444
18	18:52:16	1882.812		-26.032	7.130
19	18:52:17	1882.270		-32.854	7.589
20	18:52:17	1886.719		1.350	7.558
21	18:52:17	1890.191		21.037	7.813
22	18:52:17	1917.314		10.248	8.011
23	18:52:18	1842.122		6.791	8.247
24	18:52:18	1849.609		-11.888	5.264
25	18:52:18	1801.758		-13.045	5.692
26	18:52:20	1787.543		-2.527	2.958
27	18:52:20	1770.182		7.212	2.145
28	18:52:30	1758.789		5.100	1.153
29	18:52:30	1715.603		3.904	0.502
30	18:52:40	1712.131		-2.683	1.966
31	18:52:40	1732.747		-5.053	2.164
32	18:52:44	1723.958		-3.446	0.986
33	18:52:45	1726.129		1.772	1.488
34	18:52:51	1725.694		3.128	1.364
35	18:52:51	1724.501		1.901	1.389
<b>ts = 0:00:36</b>		<b>Mp = 9.561%</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID:  $K_p=0.80$ ,  $T_i=2.74$  s,  $T_d=0.68$  s*

**Tabla 60***Mediciones con el controlador PID autosintonizado para setpoint 2000 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	19:25:36	35.156	2000	-99.022	251.563
2	19:25:49	633.464		100.00	68.327
3	19:25:50	663.303		100.00	66.835
4	19:25:50	670.790		100.00	66.461
5	19:25:50	698.025		100.00	65.099
6	19:25:50	705.404		100.00	64.730
7	19:25:51	735.460		100.00	63.227
8	19:25:57	1218.316		100.00	39.084
9	19:25:57	1225.694		100.00	38.715
10	19:25:58	1255.317		100.00	37.234
11	19:25:58	1262.695		100.00	36.865
12	19:26:03	1635.308		100.00	18.235
13	19:26:03	1665.473		100.00	16.726
14	19:26:03	1672.852		68.267	16.357
15	19:26:04	1703.017		-1.956	14.849
16	19:26:04	1710.395		-82.318	14.480
17	19:26:07	1932.726		-60.318	3.364
18	19:26:07	1962.240		3.590	1.888
19	19:26:07	1970.269		59.239	1.487
20	19:26:08	1999.892		17.529	0.005
21	19:26:08	2007.270		8.131	0.364
22	19:26:08	2037.435		-45.277	1.872
23	19:26:11	2150.813		-33.429	2.241
24	19:26:12	1992.622		15.444	0.369
25	19:26:12	1967.448		34.065	1.628
26	19:26:12	1934.028		21.340	3.299
27	19:26:12	1958.225		4.974	2.089
28	19:26:13	1984.484		-23.159	0.776
29	19:26:13	2004.557		-28.101	0.228
30	19:26:13	1963.325		3.123	1.834
31	19:26:13	1965.495		22.774	1.725
32	19:26:14	1973.199		12.447	1.340
33	19:26:14	1990.126		7.163	0.494
34	19:26:14	1988.715		-13.658	0.564
35	19:26:14	1993.490		-16.348	0.326
<b>ts = 0:00:38</b>		<b>Mp = 7.541%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID:  $K_p=0.80$ ,  $T_i=2.74$  s,  $T_d=0.68$  s*

**Tabla 61***Mediciones con el controlador PID autosintonizado para setpoint 2250 mm*

N° medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	10:47:00	45.356	2250	29.000	97.984
2	10:47:27	2098.199		100.000	6.747
3	10:47:28	2127.821		48.986	5.430
4	10:47:28	2135.851		-29.075	5.073
5	10:47:28	2165.365		-100.00	3.762
6	10:47:28	2172.743		-77.249	3.434
7	10:47:29	2202.908		36.897	2.093
8	10:47:29	2210.287		70.106	1.765
9	10:47:29	2240.560		33.182	0.420
10	10:47:29	2247.938		-8.200	0.092
11	10:47:30	2278.103		-63.315	1.249
12	10:47:30	2285.482		-49.305	1.577
13	10:47:30	2314.996		36.788	2.889
14	10:47:30	2322.374		64.187	3.217
15	10:47:31	2352.539		42.566	4.557
16	10:47:31	2360.026		5.983	4.890
17	10:47:31	2398.672		-47.960	6.163
18	10:47:31	2380.751		-57.463	5.811
19	10:47:32	2369.900		-63.315	5.329
20	10:47:32	2353.841		-49.305	4.615
21	10:47:32	2379.123		36.788	5.739
22	10:47:32	2361.871		64.187	4.972
23	10:47:33	2293.728		42.566	1.943
24	10:47:33	2254.449		5.983	0.198
25	10:47:33	2254.015		-47.960	0.178
26	10:47:33	2249.023		-57.463	0.043
27	10:47:34	2226.020		13.441	1.066
28	10:47:34	2218.750		54.342	1.389
29	10:47:34	2171.550		31.440	3.487
30	10:47:34	2188.151		15.539	2.749
31	10:47:35	2215.278		-33.481	1.543
32	10:47:35	2241.320		-40.393	0.386
33	10:47:36	2205.729		-0.940	1.968
34	10:47:37	2209.635		29.103	1.794
35	10:47:38	2229.709		15.244	0.902
<b>ts = 0:00:38</b>		<b>Mp = 6.608%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID:  $K_p=0.80$ ,  $T_i=2.74$  s,  $T_d=0.68$  s*



**Tabla 62***Mediciones con el controlador PID autosintonizado para setpoint 2500 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	11:22:44	11.827	2500	100.00	99.527
2	11:23:14	2223.199		100.00	11.072
3	11:23:14	2230.577		100.00	10.777
4	11:23:14	2260.091		100.00	9.596
5	11:23:14	2268.121		100.00	9.275
6	11:23:15	2297.635		100.00	8.095
7	11:23:15	2305.013		100.00	7.799
8	11:23:15	2334.635		100.00	6.615
9	11:23:15	2342.665		100.00	6.293
10	11:23:16	2372.179		100.00	5.113
11	11:23:16	2379.557		100.00	4.818
12	11:23:16	2409.722		100.00	3.611
13	11:23:16	2417.101		100.00	3.316
14	11:23:17	2446.723		100.00	2.131
15	11:23:17	2454.102		13.280	1.836
16	11:23:17	2484.267		-65.101	0.629
17	11:23:17	2491.645		-100.00	0.334
18	11:23:18	2521.267		-0.131	0.851
19	11:23:18	2528.646		77.245	1.146
20	11:23:18	2558.811		33.709	2.352
21	11:23:18	2566.189		23.383	2.648
22	11:23:19	2591.797		-50.335	3.672
23	11:23:19	2590.169		-50.319	3.607
24	11:23:19	2611.328		8.061	4.453
25	11:23:19	2620.226		47.308	4.809
26	11:23:20	2621.853		27.003	4.874
27	11:23:20	2572.157		14.499	2.886
28	11:23:20	2535.373		-28.683	1.415
29	11:23:20	2550.239		-34.355	2.010
30	11:23:21	2540.039		1.343	1.602
31	11:23:21	2518.012		29.194	0.720
32	11:23:21	2453.451		16.347	1.862
33	11:23:21	2450.846		12.645	1.966
34	11:23:22	2452.148		-14.451	1.914
35	11:23:22	2501.519		-19.260	0.061
<b>ts = 0:00:38</b>		<b>Mp = 4.874%</b>		<b>ess ≤ 2%</b>	

*Nota: Parámetros PID:  $K_p=0.80$ ,  $T_i=2.74$  s,  $T_d=0.68$  s*

**Tabla 63***Mediciones con el controlador PID autosintonizado para setpoint 3000 mm*

Nº medición	Hora	Entrada mm	Setpoint mm	Salida %	Error de estado estable ( $e_{ss}$ )
1	11:32:40	838.759	3000	100.000	72.041
2	11:33:05	2657.986		100.000	11.400
3	11:33:05	2688.151		100.000	10.395
4	11:33:05	2695.530		100.000	10.149
5	11:33:06	2723.307		100.000	9.223
6	11:33:06	2730.686		100.000	8.977
7	11:33:06	2760.851		100.000	7.972
8	11:33:06	2768.229		100.000	7.726
9	11:33:07	2797.852		100.000	6.738
10	11:33:07	2805.230		100.000	6.492
11	11:33:07	2835.395		100.000	5.487
12	11:33:07	2842.773		100.000	5.241
13	11:33:08	2872.938		100.000	4.235
14	11:33:08	2880.317		100.000	3.989
15	11:33:08	2910.482		100.000	2.984
16	11:33:08	2917.860		100.000	2.738
17	11:33:09	2944.010		100.000	1.866
18	11:33:09	2950.304		100.000	1.657
19	11:33:09	2972.005		100.000	0.933
20	11:33:09	2976.562		100.000	2.781
21	11:33:10	2991.970		97.000	0.268
22	11:33:10	2994.900		98.000	0.170
23	11:33:10	2999.892		60.000	0.004
24	11:33:10	3015.000		9.000	0.000
25	11:33:11	2950.304		-2.000	2.933
26	11:33:11	2972.005		-6.000	0.781
27	11:33:11	2976.562		-5.000	0.268
28	11:33:11	2991.970		0.000	1.170
29	11:33:13	2994.900		6.000	0.004
30	11:33:13	2999.892		-4.000	0.200
31	11:33:13	3000.200		-1.000	1.945
32	11:33:20	2960.304		-5.000	0.790
33	11:33:20	2972.010		5.000	0.275
34	11:33:22	2979.562		-9.000	0.172
35	11:33:22	2999.970		-3.000	0.004
<b>ts = 0:00:42</b>		<b>Mp = 0.500%</b>			<b>ess ≤ 2%</b>

*Nota: Parámetros PID:  $K_p=0.80$ ,  $T_i=2.74$  s,  $T_d=0.68$  s*

## E. Fotografías

### Fotografía 1

*Interconexión del PLC S7 1500 a planta térmica SO3536-8T de Lucas Nulle, para control PID autosintonizado de la variable temperatura*



*Nota:* Imagen obtenida en el Laboratorio de Control y Automatización de la EPIE-FIES-UNH

## **Fotografía 2**

*Interface de bienvenida a aplicación de WinCC Unified*



## **Fotografía 3**

*Interface de selección de roles y contraseña de usuarios*



#### Fotografía 4

*Interface de inicio con acceso a imágenes de configuración, autosintonización y simulación*



#### Fotografía 5

*Interface de la imagen “configuración” para autosintonización*





### Fotografía 6

*Interface de la imagen “Autosintonización”*



### Fotografía 7

*Interface de la imagen “Configuración” para autosintonizar la variable Temperatura*



### Fotografía 8

*Interface de la imagen “Autosintonización\_temperatura”*



### Fotografía 9

*Interface de la imagen “Configuración” para autosintonizar la variable nivel*



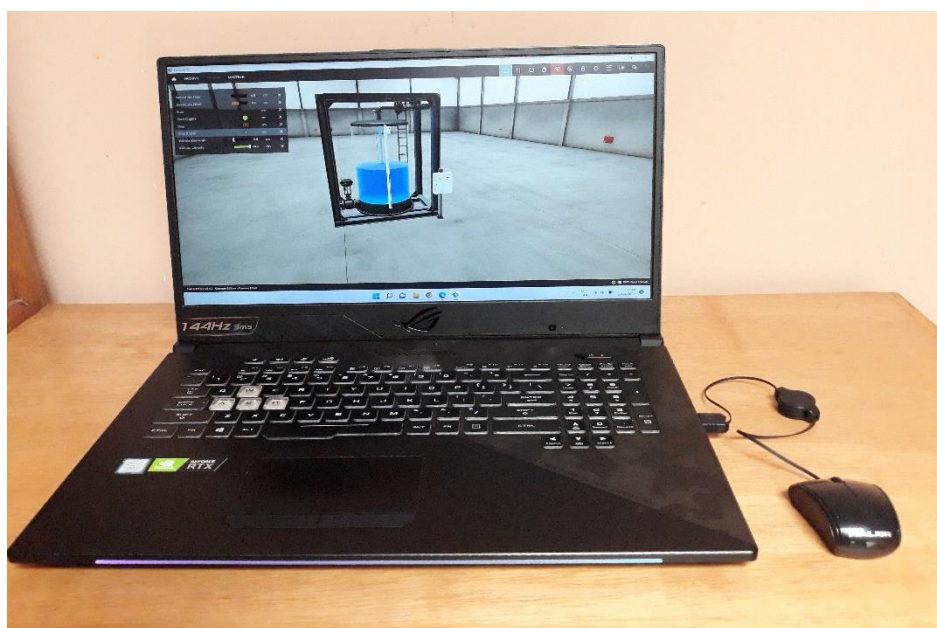
### Fotografía 10

*Interface de la imagen “Autosintonización\_nivel”*



### Fotografía 11

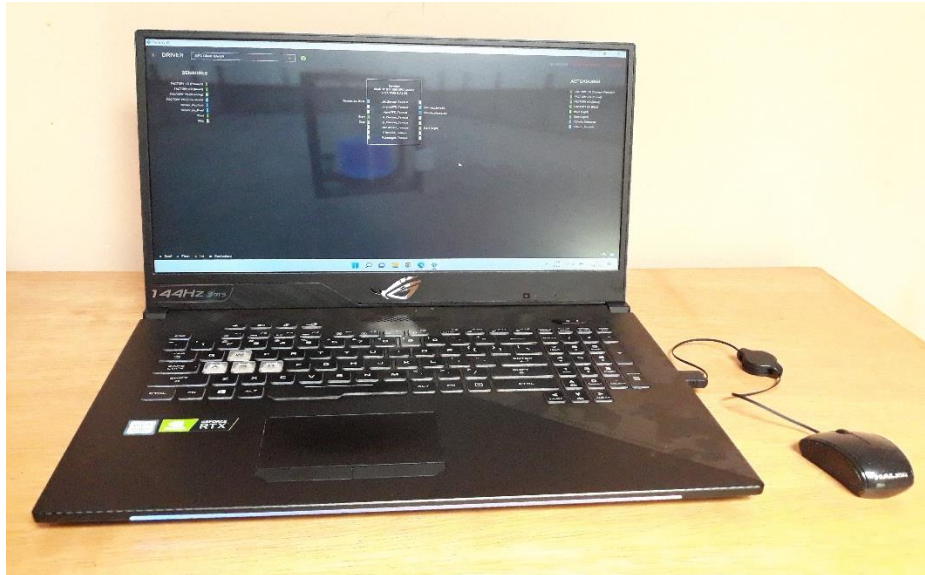
*Escenario de la planta virtual en Factory IO para control de nivel de líquido en tanque*





### Fotografía 12

*Diagrama de comunicación de variables por OPC UA entre la planta virtual en Factory IO y PLC S7 1500*



### Fotografía 13

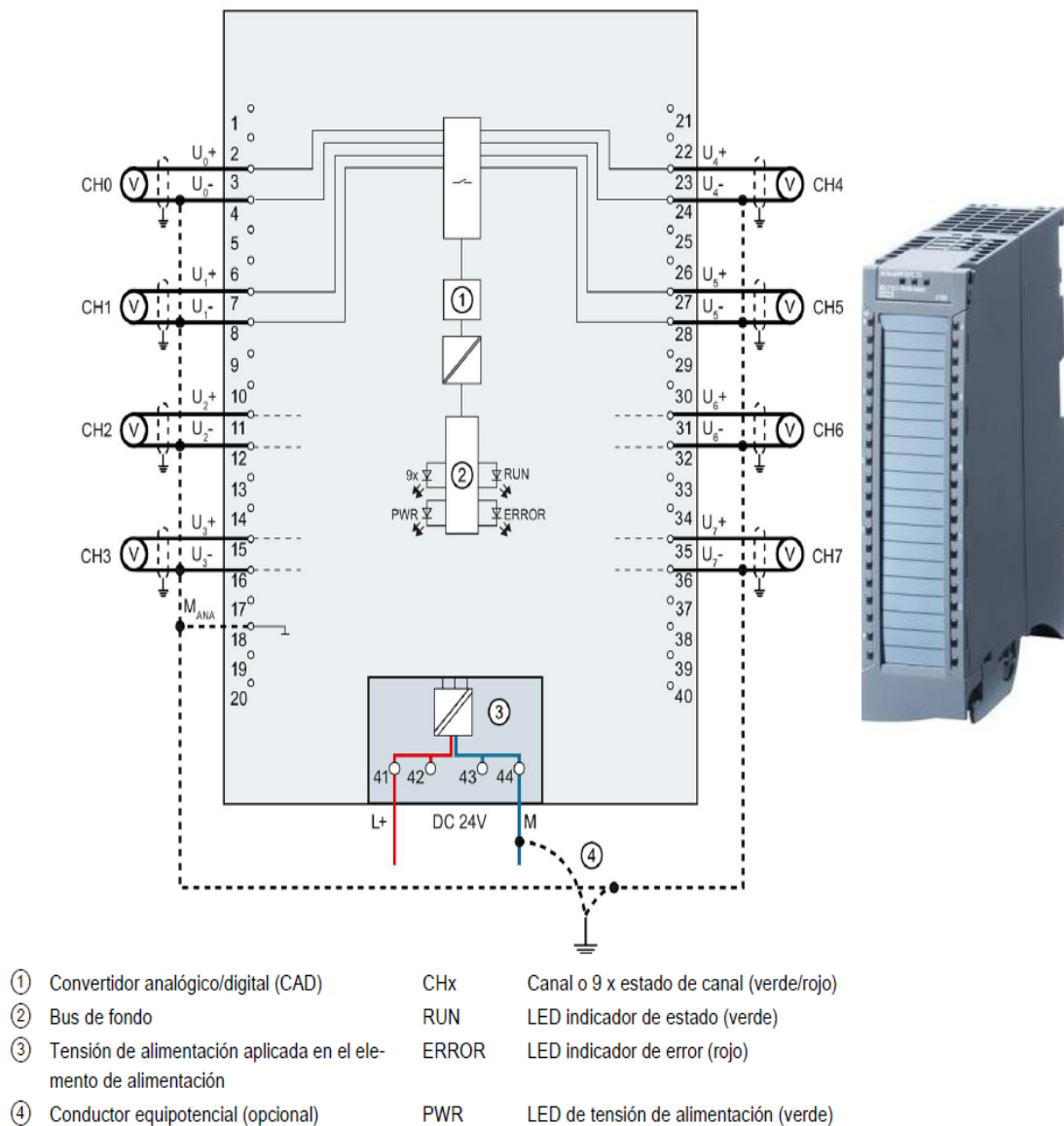
*Interface de la imagen “simulación” de autosintonización de procesos*



## F. Diagrama de conexión de entrada/ salida de módulos analógicos para PLC S7 1500

**Figura 67**

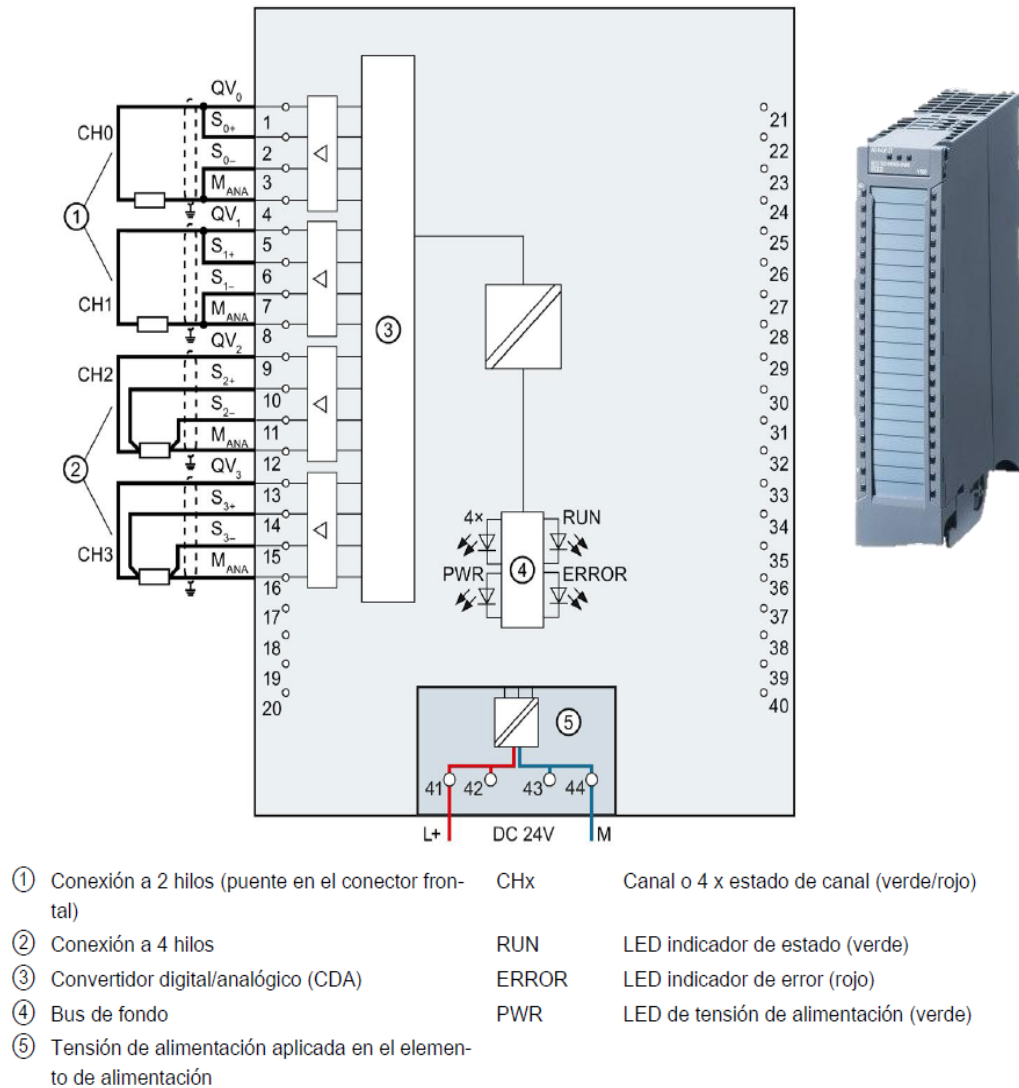
*Diagrama de terminales para medición de tensión proveniente de sensores*



*Nota.* Imagen obtenida de módulo de entrada analógicas AI 8xU/I/RTD/TC ST (Siemen, 2018)

**Figura 68**

*Diagrama de terminales para salida de tensión para activar calefactor en planta térmica*



*Nota.* Imagen obtenida de módulo de salidas analógicas AQ 4xU/I ST (Siemens, 2018)

## G. Matriz de consistencia

TITULO: IMPLEMENTACIÓN DE UN CONTROLADOR PID AUTOSINTONIZABLE PARA PROCESOS INDUSTRIALES INTEGRADO A UN PLC CON COMUNICACIÓN OPC UA					
PROBLEMAS	OBJETIVOS	HIPÓTESIS	DEFINICIÓN OPERATIVA DE VARIABLES		MÉTODO
			VARIABLES	DEFINICION OPERATIVA	
<b>Problema General</b>	<b>Objetivo General</b>	<b>Hipótesis General</b>	<b>Variable independiente:</b>  ➤ Implementación de un controlador PID autosintonizable.	El controlador PID autosintonizable es implementado en PLC S7 1500 con módulo PID_Compact y comunicación OPC UA bajo las reglas de Ziegler Nichols.	➤ En esta investigación se empleó el método científico que utilizó como procesos lógicos la inducción y la deducción, que permitieron implementar un controlador PID autosintonizable para reducir el tiempo de asentamiento de las variables en procesos industriales.
➤ ¿Será factible la implementación de un controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA, que reduce significativamente el tiempo de asentamiento de variables controladas?	➤ Implementar un controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA, que reduce significativamente el tiempo de asentamiento de variables controladas.	Es factible la implementación de un controlador PID autosintonizable para procesos industriales integrado a un PLC S7 1500 con comunicación OPC UA, que reduce significativamente el tiempo de asentamiento de variables controladas.			
<b>Problemas Específicos</b>	<b>Objetivos Específicos</b>	<b>Hipótesis Específicas</b>	<b>Variable Dependiente:</b>		
➤ ¿Será factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción significativa del tiempo de asentamiento de la variable temperatura en un proceso industrial térmico para un $e_{ss} \leq 2\%$ ?	➤ Implementar el servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción significativa del tiempo de asentamiento de la variable temperatura en un proceso industrial térmico para un $e_{ss} \leq 2\%$ .	➤ Es factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción en más del 60% del tiempo de asentamiento de la variable temperatura en un proceso industrial térmico para un $e_{ss} \leq 2\%$ .	➤ Reducción de tiempo de asentamiento de variables controladas en procesos industriales	La reducción del tiempo de asentamiento de variables controladas en procesos industriales está en relación con el proceso y los parámetros ( $K_p$ , $T_1$ y $T_d$ ) que se establecen para el controlador PID durante la autosintonización desde la interface del cliente implementado en WinCC Unified que soporta comunicación OPC UA.	
➤ ¿Será factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción significativa del tiempo de asentamiento de la variable nivel de líquido en tanque para un $e_{ss} \leq 2\%$ ?	➤ Implementar el servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción significativa del tiempo de asentamiento de la variable nivel de líquido en tanque para un $e_{ss} \leq 2\%$ .	➤ Es factible la implementación del servidor de un controlador PID autosintonizable en un PLC S7 1500 con PID_Compact y regla de Ziegler-Nichols, con comunicación de datos por OPC UA con un cliente desarrollado en WinCC Unified, para la reducción en más del 16% del tiempo de asentamiento de la variable nivel de líquido en tanque para un $e_{ss} \leq 2\%$ .			