

Trabajo Fin de Grado

Ingeniería en Electrónica, Robótica y Mecatrónica

Modelado, simulación y control de un tanque con control de temperatura

Autor: Julio Reyes Quesada

Tutor: David Muñoz de la Peña Sequeda

Dpto. de Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2021



Trabajo de Fin de Grado
Ingeniería en Electrónica, Robótica y Mecatrónica

Modelado, simulación y control de un tanque con control de temperatura

Autor:

Julio Reyes Quesada

Tutor:

David Muñoz de la Peña Sequeda

Catedrático

Dpto. de Ingeniería de Sistemas y Automática

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2021

Resumen

Un sistema dinámico es un sistema cuyo estado evoluciona con el tiempo. Su comportamiento se puede modelar matemáticamente para simular su estructura y funcionamiento, y poder estudiar y experimentar sobre el modelo sin acceder al sistema. En este trabajo se desarrollará la tecnología necesaria para automatizar procesos continuos utilizando Codesys, Vijeo Citect y Simulink. Estos programas se comunicarán entre ellos utilizando el estándar de comunicación OPC. Para ello, se ha elaborado una guía de conexión de los distintos programas con OPC. Se utilizarán a lo largo del trabajo tres sistemas con los que se pretende mostrar todo lo necesario para cumplir los objetivos. Primero, se modelará un tanque con control de nivel, y después, un sistema de control de temperatura. Por ultimo, se utilizará un sistema desarrollado a partir de los dos anteriores, que consistirá en un tanque con un sistema de control de la temperatura del líquido de su interior.

Abstract

A dynamical system is a system whose status evolves over time. Its behaviour can be mathematically modelled to simulate its structure and operation, and be able to study and experiment over the model without access to the system. In this project will be developed the needed technology to automate continuous processes using Codesys, Vijeo Citect and Simulink. Those softwares will be communicated with each other using the communication standar OPC. For that purpose, it has been developed a wiring guide of those softwares using OPC. Three systems will be used along this project with which is intended to show all that is needed to fulfill the objectives. First, a tank with level control will be modelled and, later, a system of temperatura control. Finally, it will be used a system developed from the two previous ones, which will consist of a tank with a system of temperature control of the liquid inside.

Índice

Resumen	v
Abstract	vii
Índice	1
Índice de Tablas	3
Índice de Figuras	5
1 Introducción	9
1.1 Descripción del problema	9
1.2 Objetivos	9
1.3 Estructura de la memoria	9
2 Guía conexión OPC	11
2.1 Configuración Codesys	11
2.1.1 Configuración de símbolos	11
2.1.2 Configuración del OPC Server	12
2.2 Configuración Vijeo Citect	12
2.3 Configuración Matlab Simulink	14
2.4 Prueba de conexión	15
3 Simulación del sistema	17
3.1 Tanque	17
3.1.1 Ecuación diferencial	17
3.1.2 Modelado y sincronización del control	18
3.2 Sistema de control de temperatura	19
3.2.1 Ecuación diferencial	19
3.2.2 Modelado y sincronización del control	21
3.3 Tanque con control de temperatura	26
3.3.1 Ecuación diferencial	27
3.3.2 Modelado y sincronización del control	28
4 Control PLC	35
4.1 Tanque	35
4.1.1 Módulos utilizados	35
4.1.2 Códigos utilizados	35
4.1.3 Resultados obtenidos	37
4.2 Sistema de control de temperatura	38
4.2.1 Módulos utilizados	39
4.2.2 Códigos utilizados	39
4.2.3 Resultados obtenidos	40
4.3 Tanque con control de temperatura	41
4.3.1 Módulos utilizados	41
4.3.2 Códigos utilizados	41

4.3.3	Resultados obtenidos	46
5	SCADA	49
5.1	<i>Guía SCADA</i>	49
5.1.1	Editor gráfico	49
5.1.2	Creación de informes	52
5.2	<i>Tanque</i>	54
5.2.1	Valores mostrados	54
5.2.2	Valores introducidos	54
5.3	<i>Sistema de control de temperatura</i>	54
5.3.1	Valores mostrados	55
5.3.2	Valores introducidos	55
5.4	<i>Tanque con control de temperatura</i>	55
5.4.1	Valores mostrados	55
5.4.2	Valores introducidos	55
5.4.3	Informe generado	55
	Referencias	57

ÍNDICE DE TABLAS

Tabla 3.1: Análisis característica estática control de temperatura	21
Tabla 3.2: Análisis característica estática del sistema	29

ÍNDICE DE FIGURAS

Figura 2.1: Configuración OPC Codesys 1	11
Figura 2.2: Figura 2.3: Configuración OPC Codesys 2	11
Figura 2.4: Configuración OPC CoDeSys 3	12
Figura 2.5: Configuración OPC Codesys 4	12
Figura 2.6: Configuración OPC Codesys 5	12
Figura 2.7: Configuración OPC Vijeo 2	13
Figura 2.8: Configuración OPC Vijeo 1	13
Figura 2.9: Configuración OPC Vijeo 4	13
Figura 2.10: Configuración OPC Vijeo 3	13
Figura 2.11: Configuración OPC Vijeo 5	13
Figura 2.12: Configuración OPC Simulink 1	14
Figura 2.13: Configuración OPC Simulink 2	14
Figura 2.14: Configuración OPC Simulink 3	14
Figura 2.15: Configuración OPC Simulink 4	14
Figura 2.16: “Symbol configuration”	15
Figura 2.17: Bloque OPC read	15
Figura 2.18: Bloque OPC write	15
Figura 2.19: Prueba de conexión Codesys Simulink	16
Figura 2.20: Prueba de conexión Codesys Simulink	16
Figura 3.1: Modelo del tanque en Simulink	18
Figura 3.2: Evolución del nivel de líquido	18
Figura 3.3: Evolución del volumen de líquido	18
Figura 3.4: Control on-off con histéresis	18
Figura 3.5: Simulación final del tanque	19
Figura 3.6: Modelo del control de temperatura en Simulink	20
Figura 3.7: Evolución de la temperatura del aire a la salida del tubo	20
Figura 3.8: Respuesta del control de temperatura ante escalón	21
Figura 3.9: Modelo del control de temperatura con control proporcional	22
Figura 3.10: Respuesta del sistema en bucle cerrado	22
Figura 3.11: Respuesta del sistema en bucle cerrado	23
Figura 3.12: Respuesta del sistema en bucle cerrado	23
Figura 3.13: Modelo del control de temperatura con control proporcional + integrador	24
Figura 3.14: Respuesta del sistema en bucle cerrado	24
Figura 3.15: Respuesta del sistema en bucle cerrado	25
Figura 3.16: Respuesta del sistema en bucle cerrado	25
Figura 3.17: Respuesta del sistema en bucle cerrado	26
Figura 3.18: Simulación final del control de temperatura	26
Figura 3.19: Modelo del sistema de control de temperatura	28
Figura 3.20: Respuesta del sistema ante escalón	28
Figura 3.21: Respuesta del sistema ante escalón	29
Figura 3.22: Modelo del control de temperatura	30
Figura 3.23: Respuesta del sistema en bucle cerrado	30
Figura 3.24: Respuesta del sistema en bucle cerrado	31
Figura 3.25: Respuesta del sistema en bucle cerrado	31
Figura 3.26: Respuesta del sistema en bucle cerrado	32
Figura 3.27: Simulación final del sistema	32

Figura 3.28: Simulación del subsistema resistencia	33
Figura 4.1: Módulo “GVL”	36
Figura 4.2: Programa principal	36
Figura 4.3: Módulo actuadores	37
Figura 4.4: Módulo sensores	37
Figura 4.5: Módulo automático	37
Figura 4.6: Comportamiento del tanque en modo automático	38
Figura 4.7: Comportamiento del tanque en modo manual	38
Figura 4.8: Programa principal	39
Figura 4.9: Documentación bloque PID	40
Figura 4.10: Comportamiento del sistema de control de temperatura	40
Figura 4.11: Programa principal	42
Figura 4.12: Programa automático	43
Figura 4.13: Programa automático	44
Figura 4.14: Programa reiniciar	45
Figura 4.15: Módulo actuadores	46
Figura 4.16: Comportamiento del sistema en modo automático	47
Figura 4.17: Comportamiento del sistema	47
Figura 5.1: Propiedades de “grupo de símbolos”	50
Figura 5.2: Propiedades de “texto” para introducir datos	50
Figura 5.3: Propiedades de “texto” para mostrar datos	50
Figura 5.4: Propiedades de “rectángulo”	51
Figura 5.5: Propiedades de “rectángulo”	51
Figura 5.6: Propiedades de “botón”	51
Figura 5.7: Propiedades de “botón”	51
Figura 5.8: Función “llenado()”	51
Figura 5.9: Configuración dispositivo	52
Figura 5.10: Configuración botón	52
Figura 5.11: Configuración botón	52
Figura 5.12: Configuración dispositivo	53
Figura 5.13: Configuración informe	53
Figura 5.14: Programación informe	53
Figura 5.15: Pantalla de supervisión del sistema	54
Figura 5.16: Pantalla de supervisión del sistema	54
Figura 5.17: Pantalla de supervisión del sistema	55
Figura 5.18: Informe periódico	56
Figura 5.19: Registro de botones	56

1 INTRODUCCIÓN

1.1 Descripción del problema

La motivación de este trabajo es poder desarrollar proyectos y practicas más ambiciosas en las asignaturas de automatización que los alumnos puedan desarrollar en sus casas, sin necesidad de comprar equipamiento, pero usando programas industriales. Para ello, se han desarrollado tres ejemplos de sistemas, simulados utilizando Matlab Simulink, Codesys y Vijeo Citect y conectados mediante el estándar de comunicación OPC.

1.2 Objetivos

El objetivo de este proyecto es desarrollar la tecnología necesaria para realizar practicas de programacion de SCADA, PLC y controladores para sistemas dinámicos de variable mixta (discreta y continua) simulados en Matlab Simulink. Para ello, se han desarrollado tres sistemas con especificaciones distintas.

- Un tanque con una válvula de llenado y un desagüe siempre abierto. Este sistema consta de un único actuador (la válvula de llenado) y un sensor que indica el nivel de líquido en el interior. Además, integrará dos modos de operación. Uno manual, en el que el nivel de líquido se mantendrá al indicado por una referencia desde el SCADA, y otro automático en el que el actuador será controlado directamente por el operario.
- Un sistema de control de temperatura compuesto por un tubo con un ventilador a la entrada que expulsa aire. Al principio del tubo habrá una resistencia que disipa potencia para calentar el aire que entra y al final del tubo habrá un sensor que mide la temperatura del aire que sale. El sistema consta, por lo tanto, de un sensor y un actuador. Integrará un único modo de operación en el que se calienta el aire para alcanzar la temperatura indicada desde el SCADA.
- Un tanque con sistema de control de temperatura con una valvula de llenado, una de vaciado y una resistencia que calienta el líquido del interior. El sistema integra tres actuadores y dos sensores, uno para el nivel y otro para la temperatura. Además, constará de tres modos de operación. Un modo manual, en el que el operario envía los valores de los actuadores al PLC. Un modo automático, en el que, de forma secuencial, el sistema llena el tanque hasta el valor indicado en el SCADA, calienta el líquido hasta el valor indicado en el SCADA, lo deja enfriar 2°C y lo vacía mientras lo mantiene a esa temperatura. Un modo de reinicio para pasar del modo manual al automático en el que el tanque se vacía, se llena hasta un cierto valor y se vuelve a vaciar.

1.3 Estructura de la memoria

Para la estructura de la memoria se ha incluido una guía de conexión de los distintos programas por OPC y se ha dividido el trabajo realizado según estos programas.

- Guía conexión OPC: En esta sección se ha desarrollado un tutorial sobre la conexión por OPC de los tres programas utilizados. Además, se ha incluido una prueba de conexión a modo de ejemplo.
- Simulación del sistema: Esta sección desarrolla el trabajo realizado en Matlab Simulink. Integra tanto el diseño de la ecuación diferencial que simula los distintos sistemas como el modelado y sincronización de los controles realizados.

- Control PLC: Aquí se explican los programas hechos en Codesys para el control de PLC, incluyendo los módulos utilizados, códigos desarrollados y un análisis de los resultados obtenidos.
- SCADA: En esta sección se incluye una guía sobre el uso de Vijeo Citect, tanto del editor gráfico como de la generación de informes. Además, se muestra el SCADA diseñado para cada uno de los sistemas.

2 GUÍA CONEXIÓN OPC

A lo largo de este capítulo se va a explicar paso por paso la conexión de Codesys (versión 3.5 SP 12 provided by Festo), Vijeo Citect (versión 7.4) y Matlab Simulink (versión R2020a) mediante el protocolo de comunicación OPC.

2.1 Configuración Codesys

Para la instalación del software Codesys hay que tener en cuenta que a partir de la versión 3.5 SP 12 el servidor del OPC es de pago con una versión de 30 días. Por ello, se recomienda la instalación de una versión previa desde la web de Festo (FESTO, s.f.). Durante el proceso de instalación será necesario habilitar la instalación del servidor OPC.

2.1.1 Configuración de símbolos

Una vez diseñado el sistema será necesario agregar a la aplicación un objeto del tipo “configuración de símbolos” al que habrá que marcarle la casilla que habilita la compatibilidad con OPC UA.

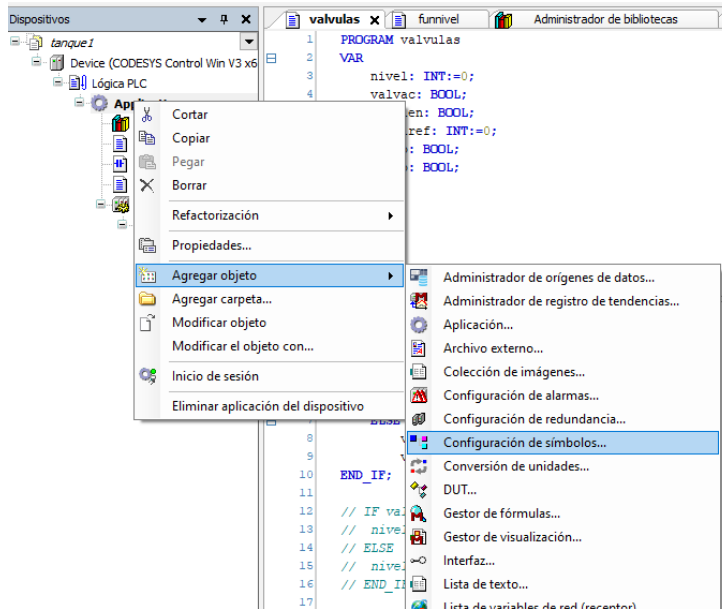


Figura 2.1: Configuración OPC Codesys 1

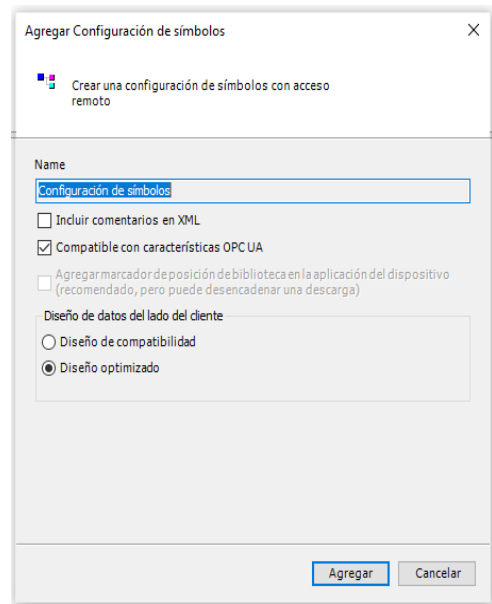


Figura 2.2: Figura 2.3: Configuración OPC Codesys 2

Dentro de este objeto se seleccionarán las variables que se quieran comunicar y se pulsará “crear”.

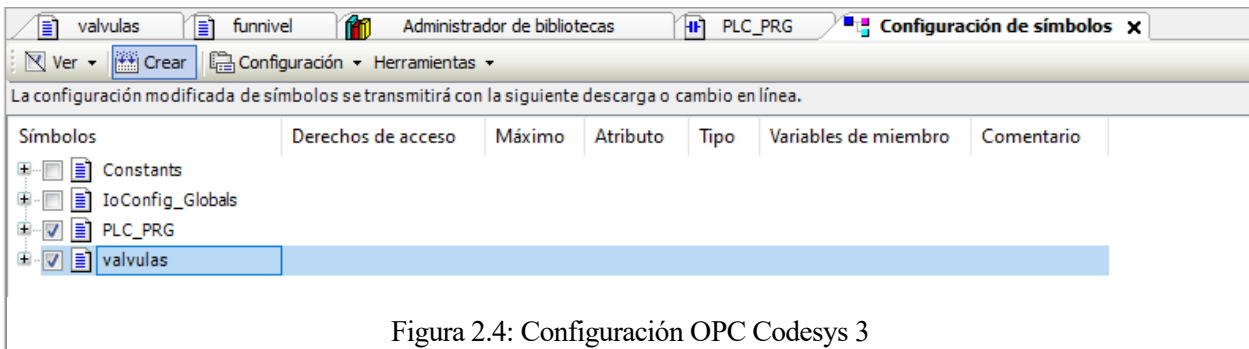


Figura 2.4: Configuración OPC Codesys 3

2.1.2 Configuración del OPC Server

Tras configurar la aplicación Codesys será necesario configurar el servidor de OPC con el que se van a comunicar los distintos programas.

Para ello hay que abrir la aplicación OPCConfig que viene incluida en la instalación de Codesys. Se añadirá un PLC y una conexión en caso de no aparecer ya añadidas y se configurará la conexión para que el nombre del PLC sea el mismo que en el dispositivo de Codesys.

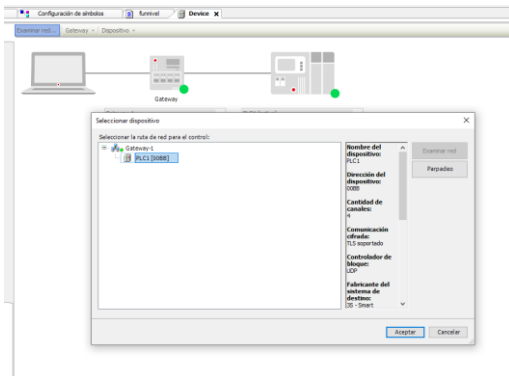


Figura 2.5: Configuración OPC Codesys 4

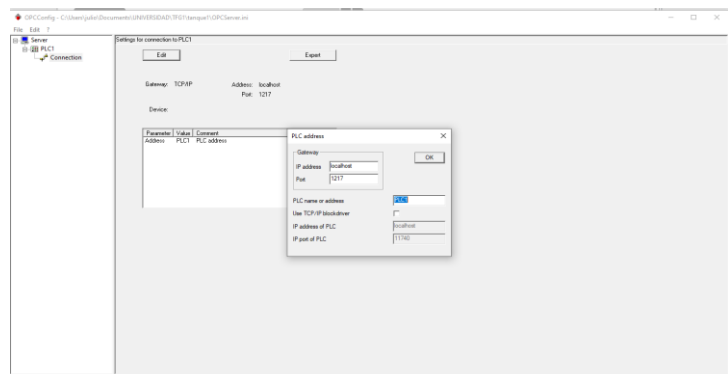


Figura 2.6: Configuración OPC Codesys 5

Cuando se inicie la sesión y se cargue el programa se comunicarán por este servidor todas las variables seleccionadas en la “configuración de símbolos”.

2.2 Configuración Vijeo Citect

Para la configuración de Vijeo se seguirán los pasos indicados en configuración rápida de dispositivo de E/S que se encuentra en el apartado de comunicaciones.

Se seleccionará un dispositivo de E/S externo que utilizará el método de comunicación OPC.

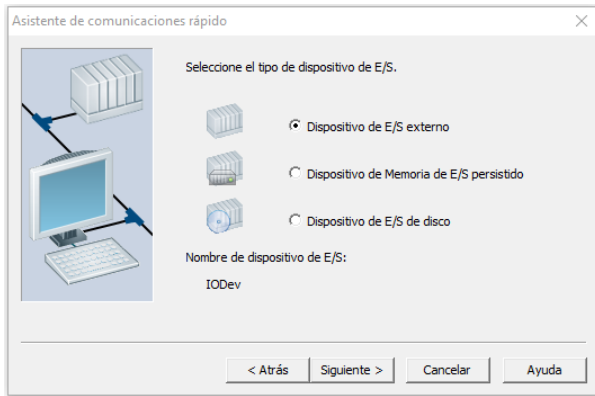


Figura 2.8: Configuración OPC Vijeo 1

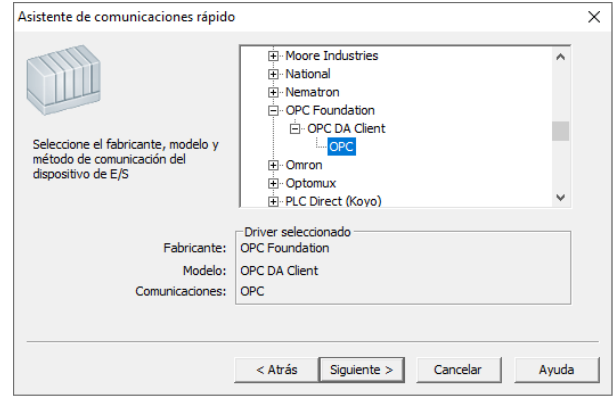


Figura 2.7: Configuración OPC Vijeo 2

Se introducirá la dirección (CoDeSys.OPC.DA) y el tipo de base de datos y se terminará la configuración.

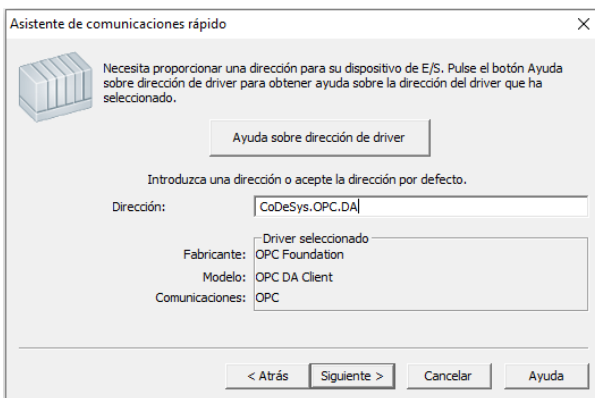


Figura 2.10: Configuración OPC Vijeo 3

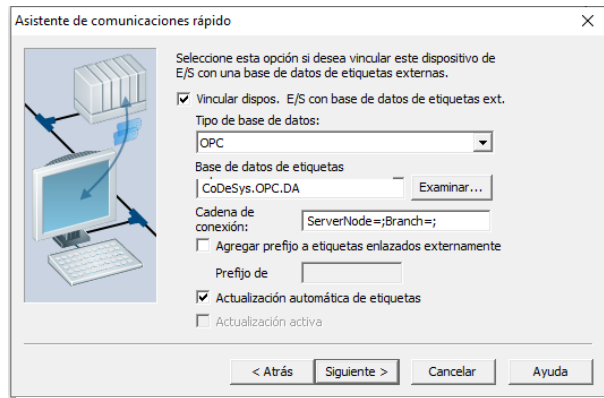


Figura 2.9: Configuración OPC Vijeo 4

Tras esto, se encontrarán todas las variables seleccionadas en CoDeSys en las etiquetas de variable con el nombre con el que se utilizarán en Vijeo.

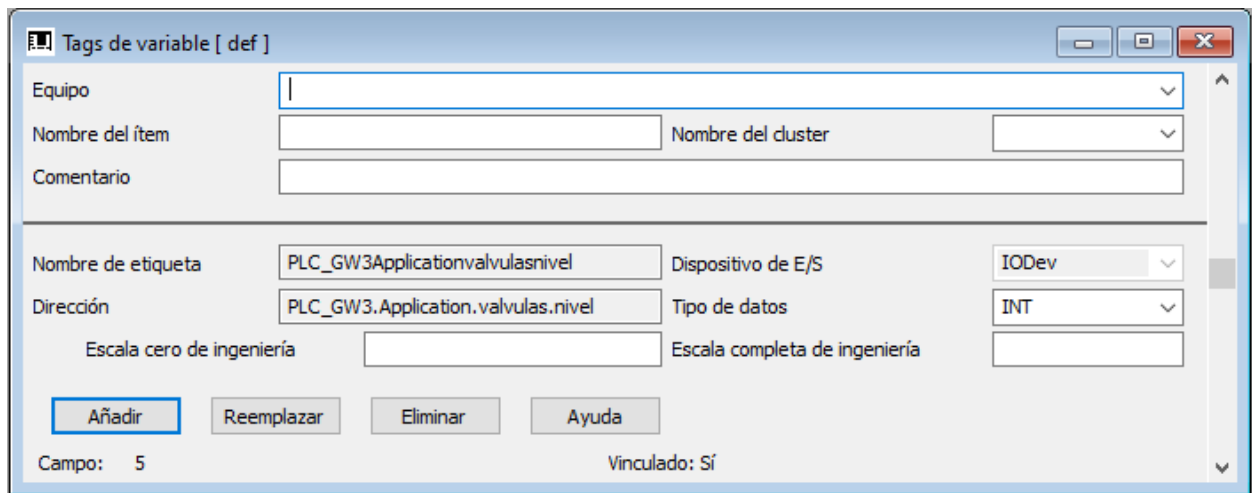


Figura 2.11: Configuración OPC Vijeo 5

2.3 Configuración Matlab Simulink

El primer paso para la comunicación de Simulink con OPC será la instalación del toolbox de OPC.

Se añadirá al sistema el bloque OPC Configuration al que se le añadirá un cliente con servidor CoDeSys.OPC.DA que será el cliente al que habrá que conectarse.

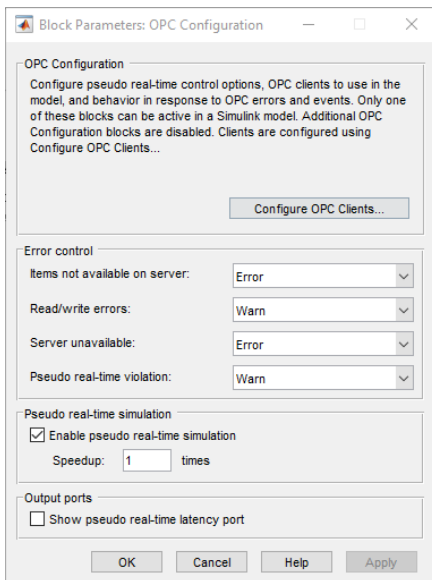


Figura 2.12: Configuración OPC Simulink 1

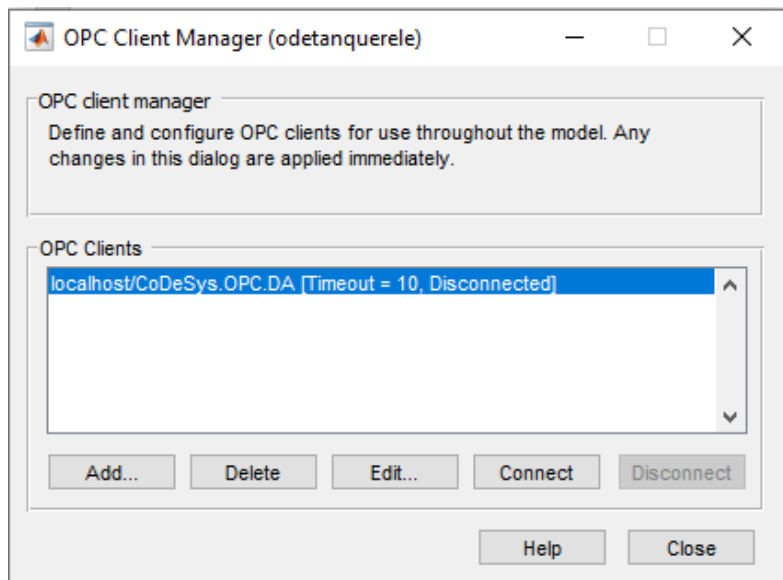


Figura 2.13: Configuración OPC Simulink 2

Se configurarán los bloques OPC Read y OPC Write para leer y escribir respectivamente las variables compartidas desde CoDeSys.

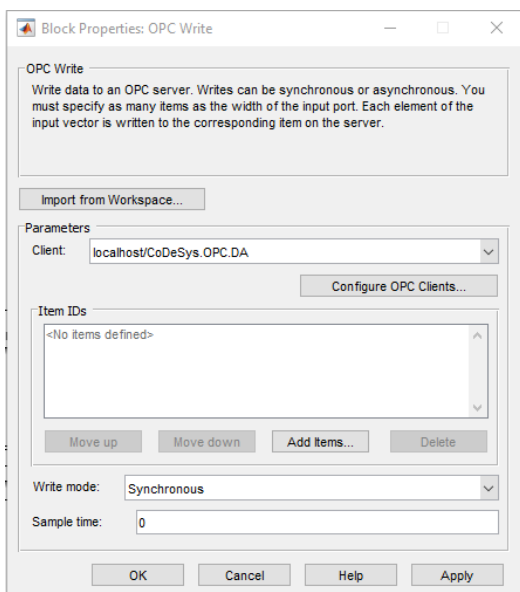


Figura 2.14: Configuración OPC Simulink 3

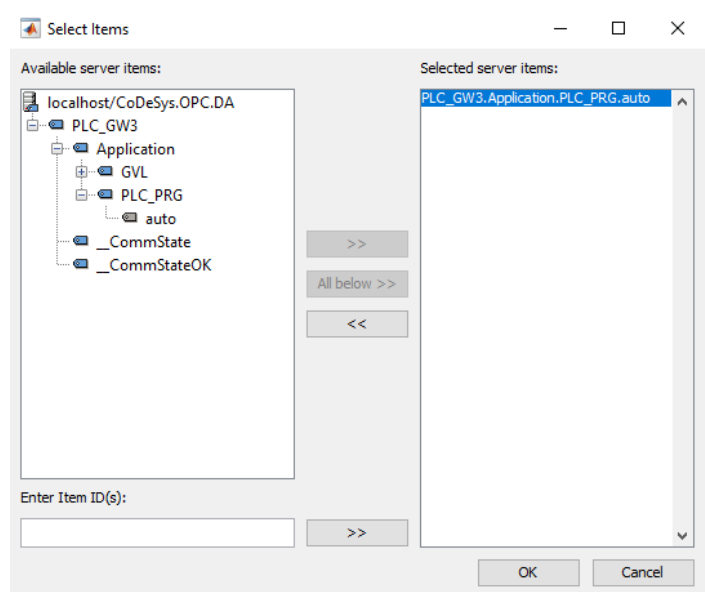


Figura 2.15: Configuración OPC Simulink 4

2.4 Prueba de conexión

A efectos de demostración de la comunicación OPC se ha desarrollado un pequeño programa en el que en Codesys se escribe un valor en una variable “valor” que pasa a Simulink donde se le hace la raíz cuadrada que se escribe en otra variable “resultado”.

Para ello, se ha configurado Codesys como se muestra en la Figura 2.16.

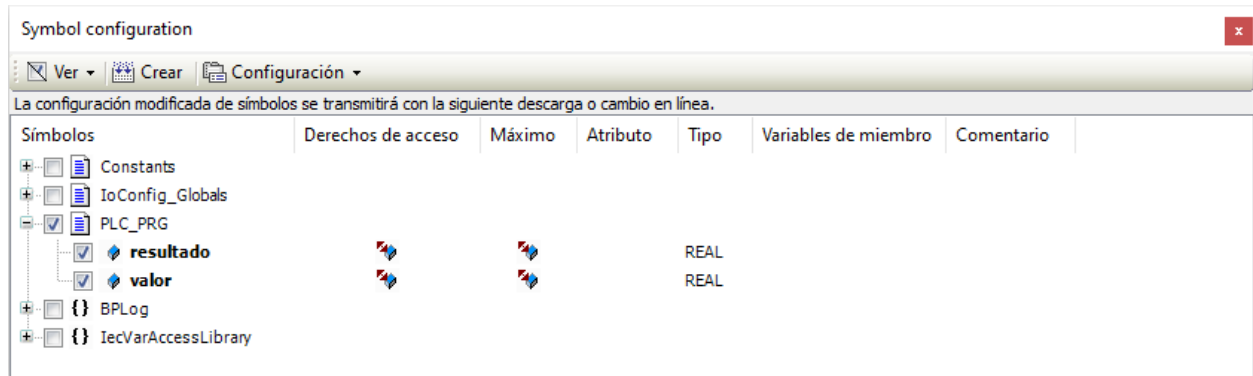


Figura 2.16: “Symbol configuration”

En cuanto a Simulink, se han configurado los bloques “OPC read” y “OPC write” como se muestra en la Figura 2.17 y Figura 2.18 respectivamente.

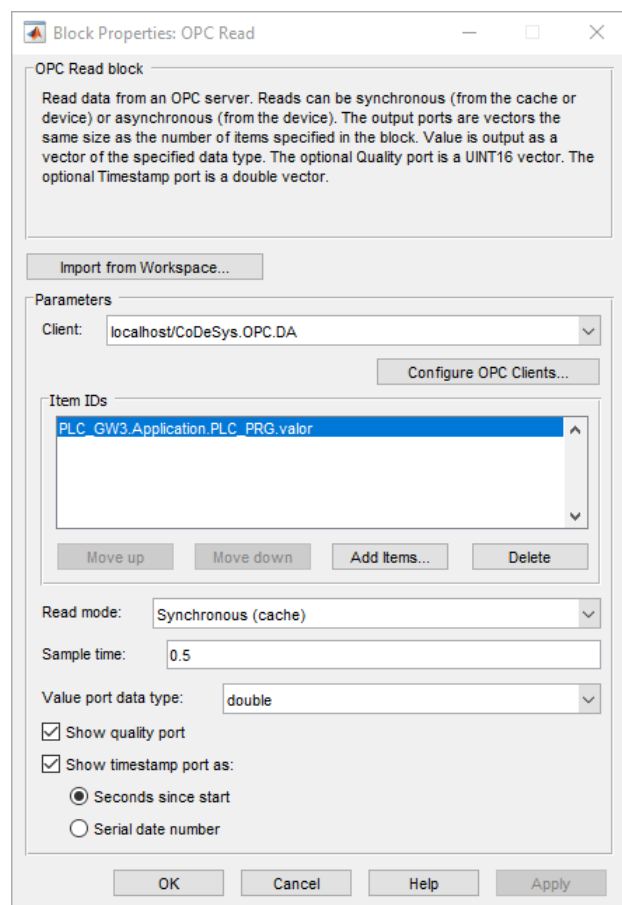


Figura 2.17: Bloque OPC read

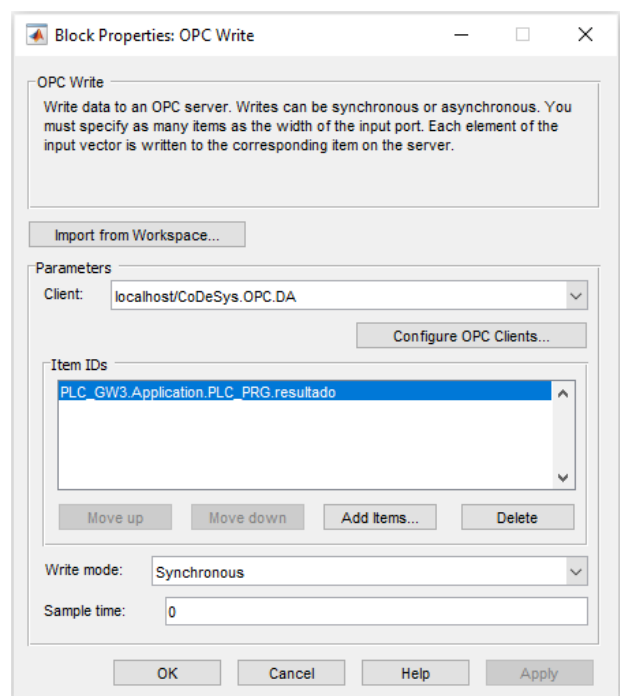


Figura 2.18: Bloque OPC write

El resultado se muestra en las Figura 2.19 y Figura 2.20.

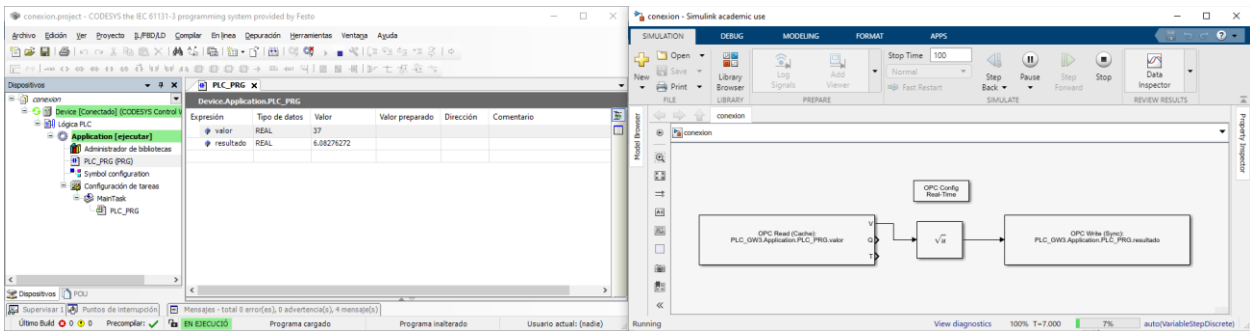


Figura 2.19: Prueba de conexión Codesys Simulink

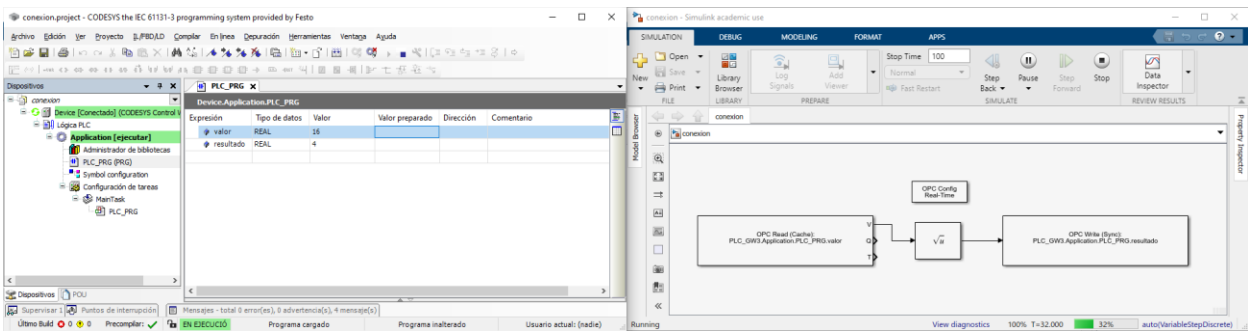


Figura 2.20: Prueba de conexión Codesys Simulink

3 SIMULACIÓN DEL SISTEMA

En este capítulo se va a describir la simulación de los distintos sistemas dinámicos usando Simulink. Para ello se ha partido de unos modelos aproximados al sistema real partiendo de la ecuación diferencial y se le han añadido los sensores y actuadores cuyos valores son escritos y leídos respectivamente por el PLC mediante el protocolo OPC.

3.1 Tanque

Este sistema consiste en un tanque de agua con una valvula de llenado por la que entra un caudal de entrada y que consta de una salida por la que escapa un caudal de salida constante.

3.1.1 Ecuación diferencial

En este apartado se indicará el modelo matemático basado en una ecuación diferencial de primer orden no lineal que describe el funcionamiento del sistema.

Las ecuaciones diferenciales que se utilizarán nos indican que la derivada del volumen (V) es directamente proporcional al caudal de entrada (q_e) menos el caudal de salida (q_s).

$$\frac{dV(t)}{dt} = q_e(t) - q_s(t)$$

La altura o nivel del tanque (h), que será la variable que se utilizará para la representación en el SCADA y como referencia para el control, multiplicado por el área (A) del tanque es igual al volumen.

$$Ah(t) = V(t)$$

El caudal de salida es proporcional a la raíz cuadrada del nivel multiplicado por una constante de descarga (K).

$$q_s(T) = K\sqrt{h(t)}$$

Para comprobar el correcto funcionamiento del modelo se ha simulado el caudal de entrada con un escalón que proporciona un caudal de entrada de $2 \text{ cm}^3/\text{s}$ durante los primeros 5 segundos que luego aumenta a $4 \text{ cm}^3/\text{s}$. El modelo de Simulink quedaría como se muestra en la Figura 3.1.

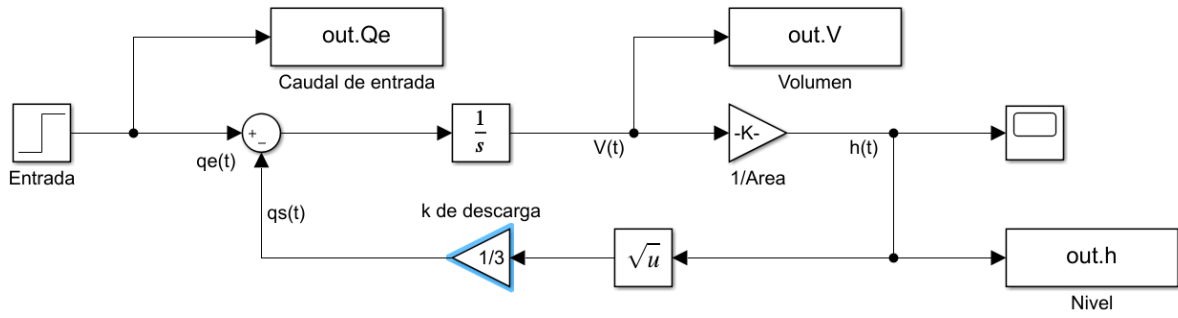


Figura 3.1: Modelo del tanque en Simulink

Los resultados obtenidos en volumen y nivel del líquido son los observados en la Figura 3.32 y Figura 3.23. Debido al diseño del tanque, hay un volumen máximo que se puede alcanzar. Esto se utilizará en el punto 5.22 para la representación porcentual del nivel.

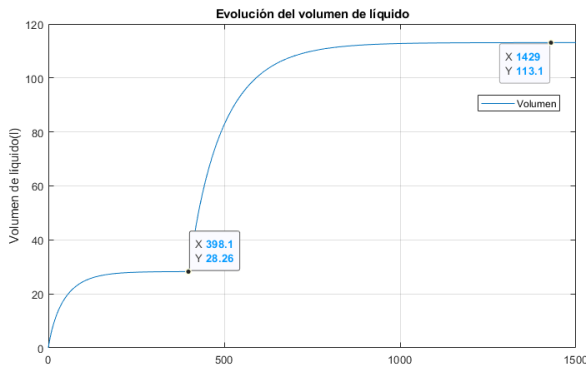


Figura 3.3: Evolución del volumen de líquido

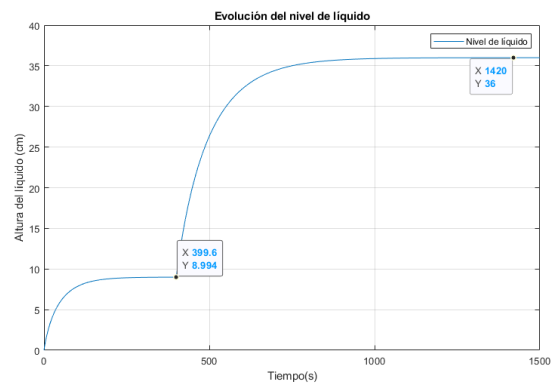


Figura 3.2: Evolución del nivel de líquido

3.1.2 Modelado y sincronización del control

Para el control de este sistema se ha optado por un control on-off regulado por la válvula de llenado. Este control consiste en que la válvula se abre (on) hasta que se alcanza el nivel buscado y en ese momento se cierra (off). Debido a que el desagüe está siempre abierto, el nivel irá bajando y una vez alcanzado un valor igual al buscado menos la histéresis la válvula se vuelve a abrir. Este esquema se ilustra en la Figura 3.4.

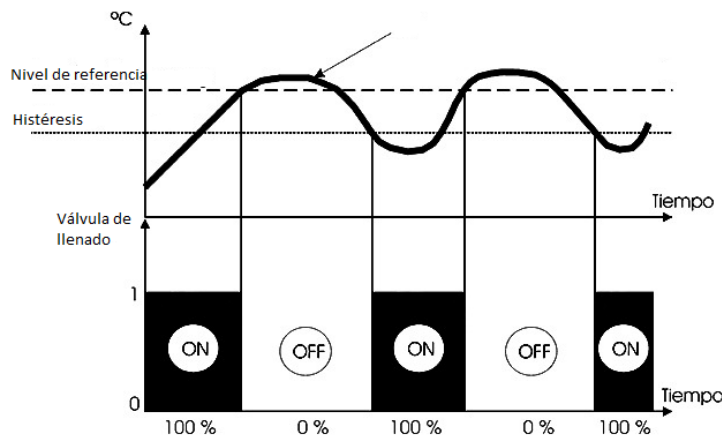


Figura 3.4: Control on-off con histéresis

En la Figura 3.5 se tiene el sistema final utilizado, incluyendo sensores y actuadores. Dentro del subsistema tanque, se encuentra el sistema modelado en este apartado.

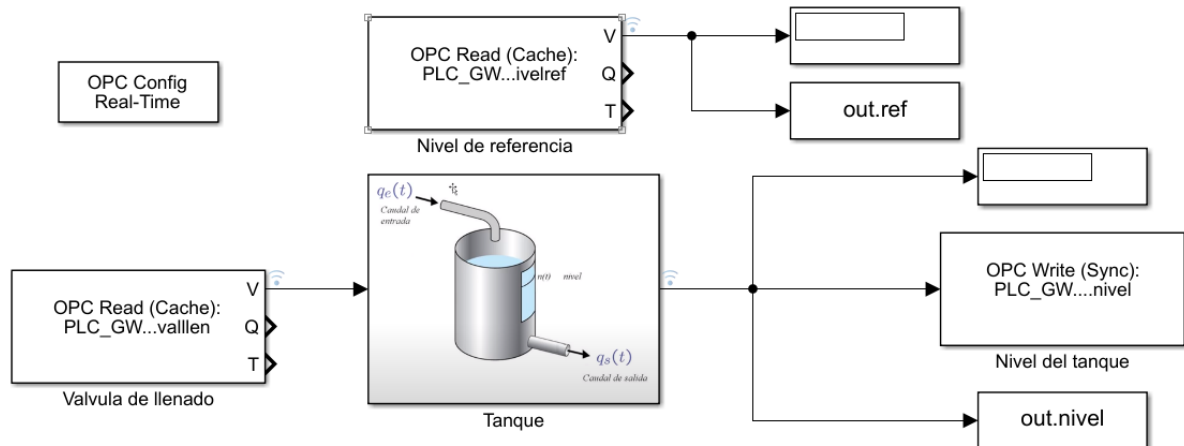


Figura 3.5: Simulación final del tanque

3.2 Sistema de control de temperatura

Este sistema consistirá en un tubo de plástico que tiene a la entrada un ventilador que expulsa aire. Al inicio del recorrido hay una resistencia en la cuál podemos disipar potencia para calentar el aire que entra. Al final del tubo se encuentra un sensor de temperatura que mide el aire caliente que sale del tubo.

3.2.1 Ecuación diferencial

En este apartado se indicará el modelo matemático simplificado basado en una ecuación diferencial de primer orden no lineal que describe el funcionamiento del sistema.

Se ha utilizado una dinámica sencilla en la que la derivada de la temperatura (T) que hay al inicio del tubo dependerá de la transferencia de calor con el ambiente, que es proporcional al caudal de aire (Q) y del volumen del tubo (V), y de la potencia disipada en la resistencia (P).

$$\frac{dT(t)}{dt} = -\frac{Q}{V} (T - T_a) + \frac{1}{\rho V c_p} P$$

La potencia disipada en la resistencia será proporcional a la tensión aplicada (V_{in}), que será la señal de control.

$$P = K_1 V_{in}$$

El problema de que la temperatura medida sea la del aire a la salida y no a la entrada, se modelará como un retraso puro. Esto quiere decir que en cada instante de tiempo se mide la temperatura que había al inicio del tubo hacía un cierto tiempo de retraso (t_d).

$$T_s = T(t - t_d)$$

Para este modelo se han utilizado los siguientes valores:

$$K_1 = 5 \Omega$$

$$\begin{aligned}\rho &= 1.28 \\ c_p &= 1012 \\ V &= 7 * 10^{-4} \text{ l} \\ Q &= 1.2 * 10^{-3} \text{ cm}^3/\text{s} \\ T_a &= 25 \text{ }^\circ\text{C} \\ t_d &= 0.18 \text{ s}\end{aligned}$$

Además, se ha supuesto una temperatura ambiente y caudal de entrada constantes y se ha limitado la tensión aplicada a unos valores entre 0 y 10 voltios.

$$V_{in} \in [0,10] \text{ V}$$

Para comprobar el correcto funcionamiento del modelo se ha simulado la tensión de entrada con un escalón que proporciona una tensión de 0V durante los primeros 2 segundos que luego aumenta a 2V. El modelo de Simulink quedaría como se muestra en la Figura 3.6.

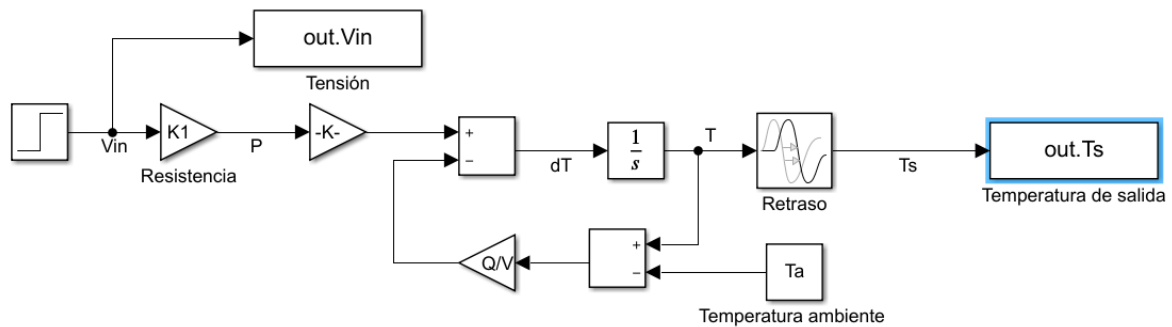


Figura 3.6: Modelo del control de temperatura en Simulink

El resultado obtenido para la temperatura de salida del aire es el observado en la Figura 3.7.

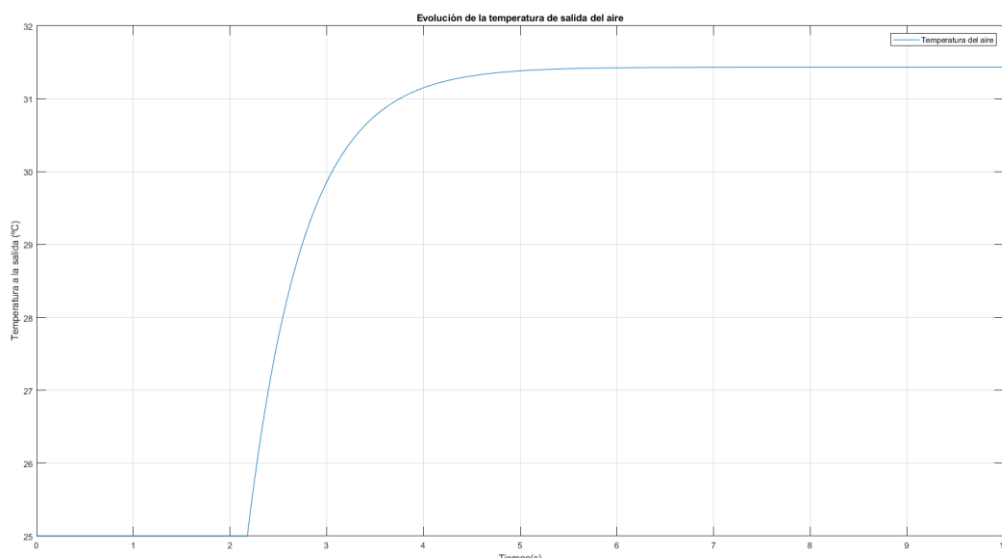


Figura 3.7: Evolución de la temperatura del aire a la salida del tubo

Se puede observar en la Figura 3.7 el retraso existente en la medida de la temperatura debido a que el sensor se encuentra a la salida.

3.2.2 Modelado y sincronización del control

Para la sincronización del control se ha hecho un análisis de la característica estática que nos indica que rangos de temperaturas podemos alcanzar teniendo en cuenta que la tensión aplicada está limitada a valores entre 0 y 10 V. Los resultados son los que se muestran en la Tabla 3.1.

Tabla 3.1: Análisis característica estática control de temperatura

Tensión aplicada	Temperatura de salida
0 V	25 °C
2 V	31.43 °C
3 V	37.9 °C
6 V	44.3 °C
8 V	50.7 °C
10 V	57.2 °C

Dentro de este rango de temperaturas se escoge como punto de operación una tensión de 2V (V_0) por lo que la temperatura del punto de operación es de 31.43°C (T_0).

A continuación se hará un análisis temporal en bucle abierto en torno al punto de operación (Figura 3.8) aplicando un escalón a los 5s entre 2 y 5 V para estimar un modelo simplificado del que poder obtener un sistema de control en bucle cerrado.

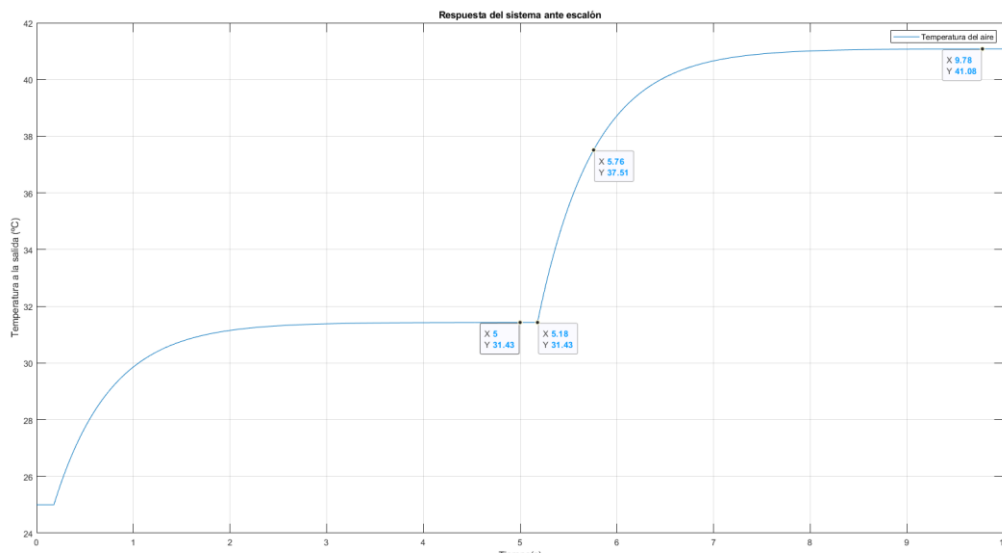


Figura 3.8: Respuesta del control de temperatura ante escalón

En la figura tal observamos el retraso existente en la medida del sistema, la característica estática (K_e) definida como la relación entre el incremento de temperatura entre el incremento de voltaje, y la constante de tiempo (τ) definida como el tiempo que tarda el sistema en alcanzar el 63% del valor final (T_τ).

$$K_e = \frac{y_{rp} - y_{eq}}{U_{rp} - U_{eq}}$$

$$T_\tau = 0.63 * (y_{rp} - y_{eq}) + t_{esc} + t_d$$

De esta forma se obtienen unos valores de $K_e = 3.2167$ y $\tau = 0.58$ s.

3.2.2.1 Control proporcional

En este apartado se implementará un sistema de control proporcional al error. Esto va a hacer que el incremento de la tensión de entrada sea proporcional a la diferencia entre la temperatura de referencia y la temperatura de salida. Se tendrán en cuenta también las limitaciones físicas del sistema, en este caso, la tensión de entrada no podrá ser ni menor que 0V ni mayor de 10V.

El modelo utilizado es el que se muestra en la **¡Error! No se encuentra el origen de la referencia.**

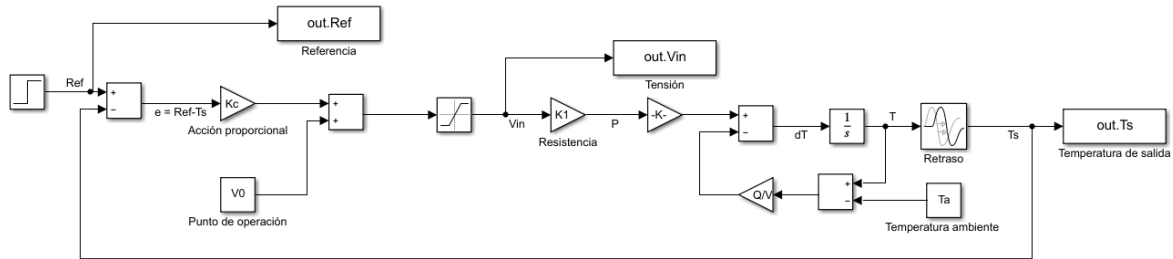


Figura 3.9: Modelo del control de temperatura con control proporcional

Para ver el funcionamiento del control se utilizarán diferentes valores de K_c y se analizará la respuesta del sistema ante una referencia que va de T_0 a $T_0 * 1.25$:

- Para un valor de $K_c = 0.54$, en la Figura 3.10, se observa que la respuesta del sistema en bucle cerrado es subamortiguado. Se puede observar cierta oscilación y error en régimen permanente. Esto es normal cuando se utiliza un control proporcional.

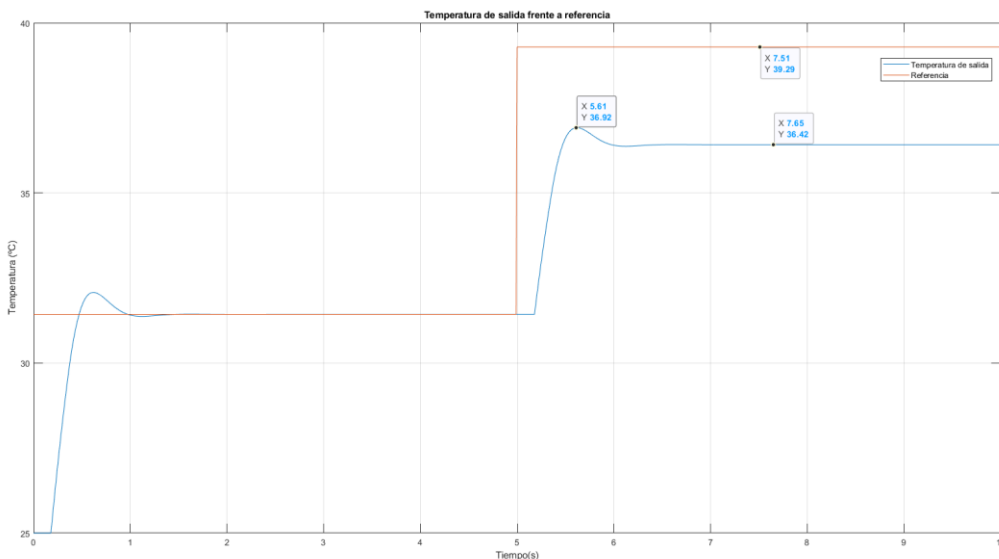


Figura 3.10: Respuesta del sistema en bucle cerrado

- Si se aumenta el valor de $K_C = 0.68$, en la Figura 3.11, se observa que aumenta la sobreoscilación, mientras que disminuyen el tiempo de subida y el error en régimen permanente.

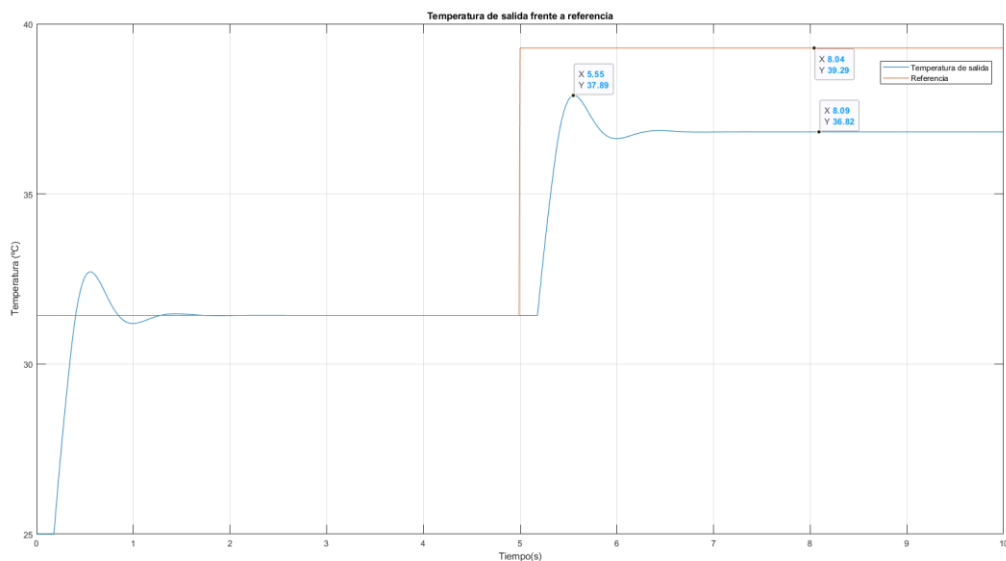


Figura 3.11: Respuesta del sistema en bucle cerrado

- Si se aumenta aún más $K_C = 2$, en la Figura 3.12, se observa que se inestabiliza.

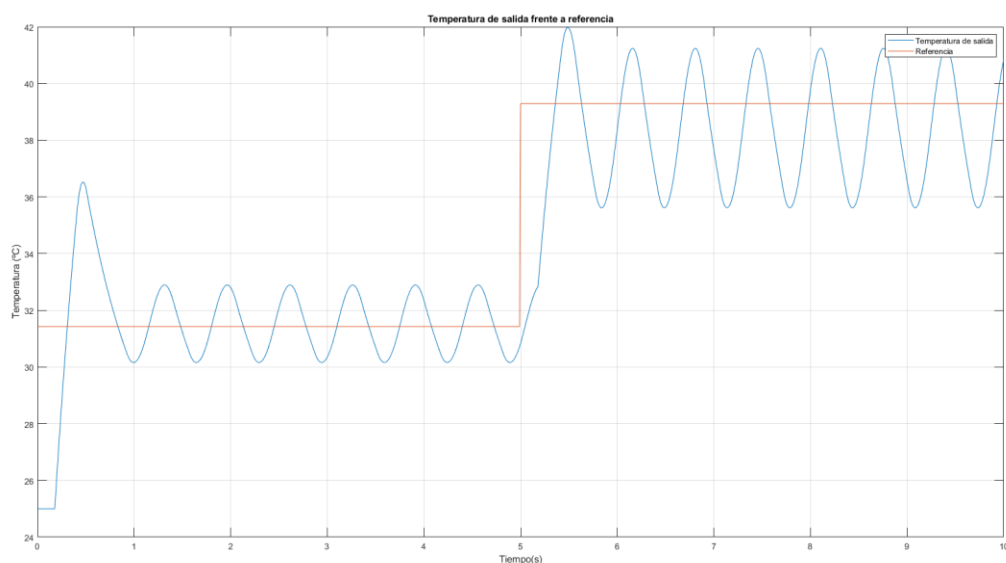


Figura 3.12: Respuesta del sistema en bucle cerrado

Con este análisis se concluye que cuanto mayor es la constante K_C el sistema se vuelve más rápido y el error en régimen permanente más pequeño, pero aumenta la sobreoscilación y llega a saturar. El problema que se observa es que siempre se obtiene un error en régimen permanente por lo que nunca se alcanza la temperatura de referencia.

Para solucionar este problema se va a añadir una acción integral al sistema de control.

3.2.2.2 Control proporcional + integrador

Al añadir una acción integral se consigue que la señal de control no sea sólo proporcional al error sino también

a la integral del error.

Para ello, se añadirá la acción integral al modelo que será inversamente proporcional al tiempo integral T_i . El modelo utilizado es el que se muestra en la Figura 3.13.

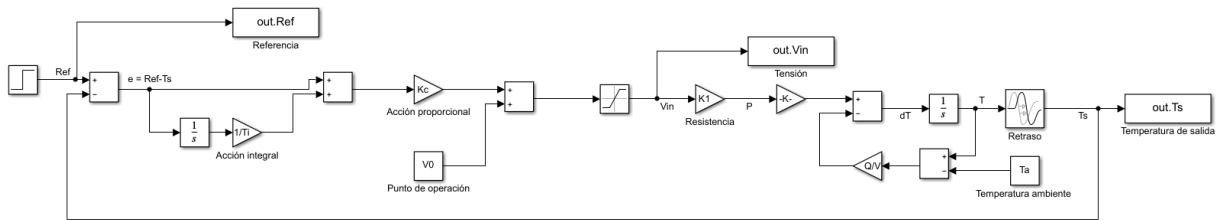


Figura 3.13: Modelo del control de temperatura con control proporcional + integrador

Para ver el funcionamiento del control se utilizarán diferentes valores de K_c y T_i y se analizará la respuesta del sistema:

- Para unos valores de $K_c = 0.54$ y $T_i = 0.58$, en la Figura 3.14, se observa que la respuesta del sistema en bucle cerrado mejora con respecto al control proporcional. Se puede observar que el error en régimen permanente se elimina y que hay una pequeña sobreoscilación.

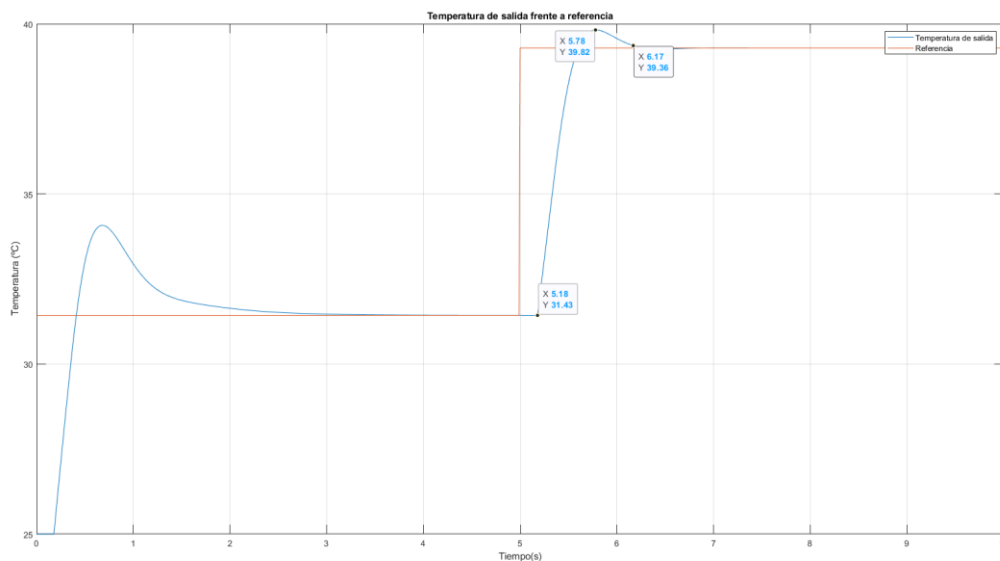


Figura 3.14: Respuesta del sistema en bucle cerrado

- Para unos valores de $K_c = 0.54$ y $T_i = 5.8$, en la Figura 3.15, se observa que la respuesta del sistema en bucle cerrado mejora con respecto al control proporcional. Se puede observar que el error en régimen permanente se elimina y que hay una pequeña sobreoscilación.

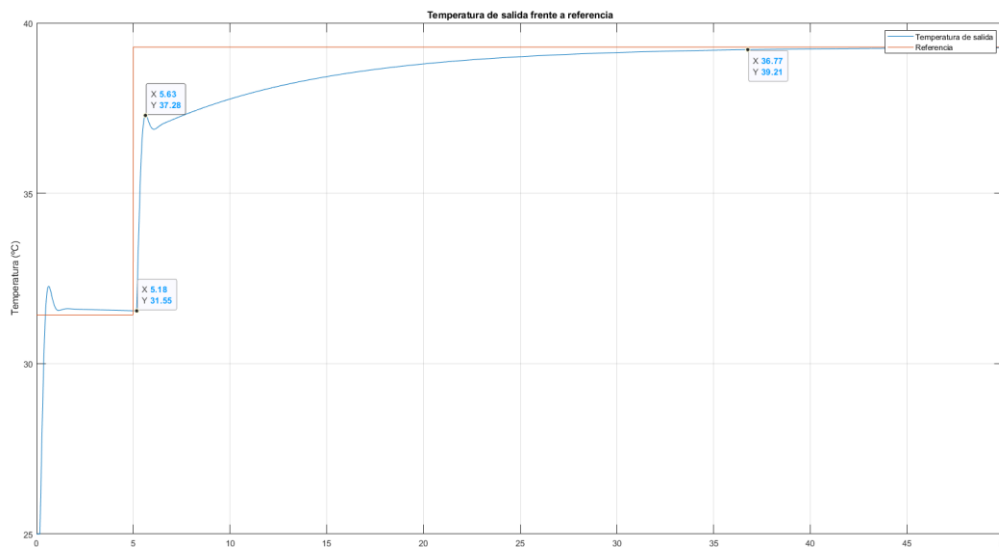


Figura 3.15: Respuesta del sistema en bucle cerrado

- Para unos valores de $K_c = 0.54$ y $T_i = 0.25$, en la Figura 3.16, se observa que la respuesta del sistema en bucle cerrado se vuelve inestable.

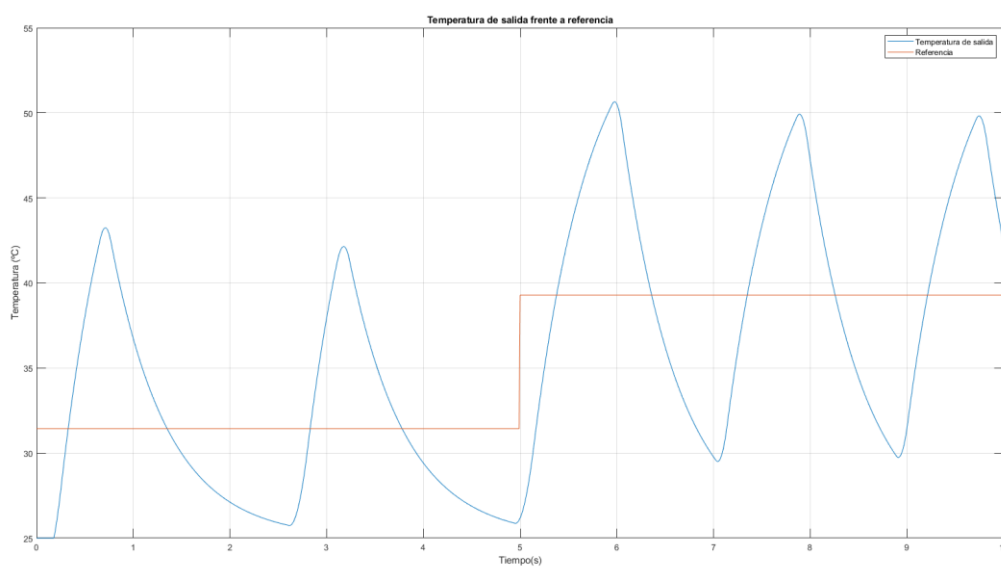


Figura 3.16: Respuesta del sistema en bucle cerrado

- Para unos valores de $K_c = 0.54$ y $T_i = 0.058$, en la Figura 3.17, se observa que la respuesta del sistema en bucle cerrado se vuelve inestable.

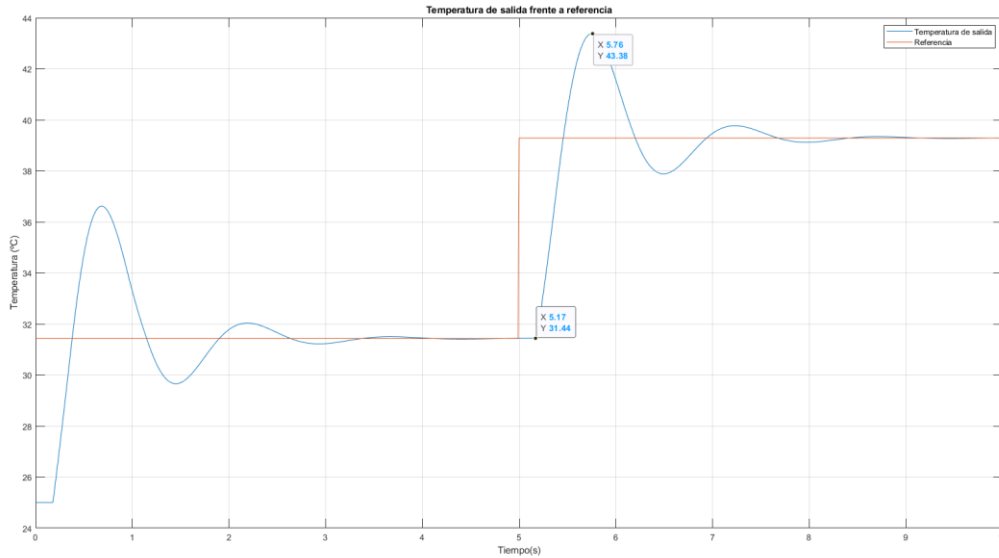


Figura 3.17: Respuesta del sistema en bucle cerrado

Con este análisis se concluye que el término proporcional a la integral del error garantiza que el error en régimen permanente sea 0. Además, cuanto más grande sea el tiempo integral, más se tarda en eliminar el error en régimen permanente (tiempo de establecimiento), este tiempo es del orden de tres veces el tiempo integral. Sin embargo, si se disminuye el tiempo integral, aumenta la temperatura máxima y la inestabilidad, pudiendo llegar a inestabilizarse. En general, se utiliza un tiempo integral del orden de la constante de tiempo del sistema para obtener una buena relación entre tiempo de establecimiento y estabilidad relativa.

Este control se implementará luego en CoDeSys en el punto 4.24.1.1.

En la **¡Error! No se encuentra el origen de la referencia.** se tiene el sistema final utilizado, incluyendo sensores y actuadores. Dentro del subsistema calentador, se encuentra el sistema modelado en este apartado.

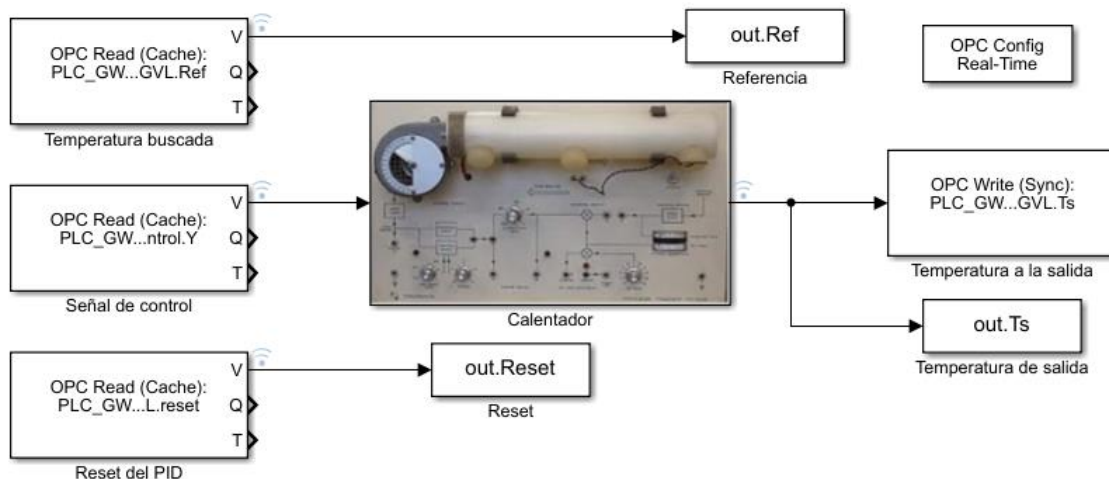


Figura 3.18: Simulación final del control de temperatura

3.3 Tanque con control de temperatura

Este sistema consiste en un tanque con dos válvulas, una de llenado y otra de vaciado, similar al tanque

diseñado previamente. Consta de una resistencia en su interior que permitirá calentar el líquido, similar al sistema de control de temperatura estudiado previamente. Estos dos subsistemas se estudiarán separadamente.

3.3.1 Ecuación diferencial

3.3.1.1 Tanque

Para el modelo matemático de este subsistema se ha utilizado la ecuación diferencial estudiada en el punto 3.1.1.

3.3.1.2 Sistema de control de temperatura

En este apartado se indicará el modelo matemático simplificado basado en una ecuación diferencial de primer orden no lineal que describe el funcionamiento del sistema.

Se ha utilizado una dinámica sencilla en la que la derivada de la temperatura del líquido T_s que hay en el interior del tanque dependerá de las pérdidas de calor con el ambiente (Q_{amb}), que será proporcional a la superficie del tanque en contacto con el ambiente (A_{dep}), y de la potencia (Q_{res}) disipada en la resistencia. Todo ello será proporcional al volumen del líquido que hay en el interior (V).

$$\frac{dT_s}{dt} = \frac{Q_{res} - Q_{amb}}{\rho * c_p * V}$$

La potencia disipada en la resistencia será proporcional a la tensión aplicada, que será la señal de control.

$$Q_{res} = K_1 V_{in}$$

Las pérdidas por el contacto con el ambiente serán proporcionales a la diferencia de temperatura entre el líquido y el ambiente y la superficie de las paredes del tanque.

$$Q_{amb} = U_{dep} * A_{dep} * (T_s - T_{amb})$$

Para este modelo se han utilizado los siguientes valores:

$$K_1 = 1000 \Omega$$

$$\rho = 1.28$$

$$c_p = 1012$$

$$U_{dep} = 4$$

$$R_{dep} = 1 m$$

$$h_{dep} = 1 m$$

Además, se ha supuesto que el líquido entra al tanque con una temperatura constante T_{ent} , una temperatura ambiente T_{amb} , y se ha limitado la tensión aplicada a unos valores entre 0 y 220 voltios.

$$T_{ent} = 24 \text{ } ^\circ\text{C}$$

$$T_{amb} = 20 \text{ } ^\circ\text{C}$$

$$V_{in} \in [0, 220] V$$

Para comprobar el correcto funcionamiento del modelo se ha simulado la tensión de entrada con un escalón que proporciona una tensión de 0V que luego aumenta a 0.1V. El modelo de Simulink quedaría como se muestra en la Figura 3.19.

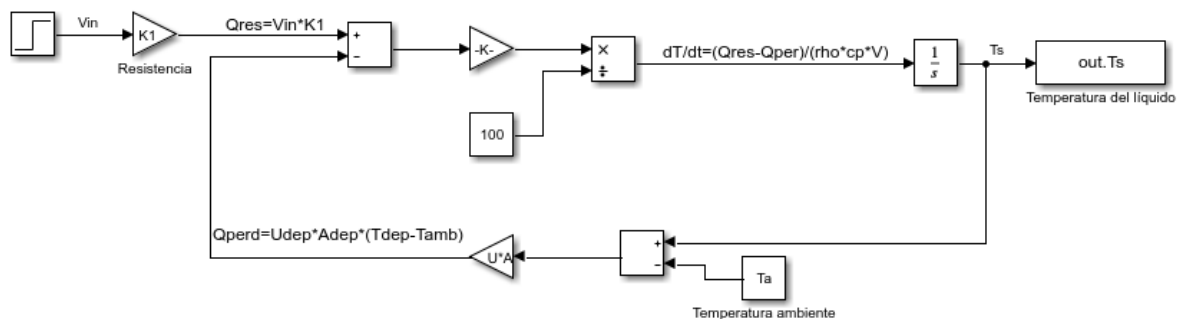


Figura 3.19: Modelo del sistema de control de temperatura

El resultado obtenido para la temperatura del líquido es el observado en la Figura 3.20 donde se observa que la temperatura desciende hasta la temperatura ambiente si no se aplica ninguna tensión.

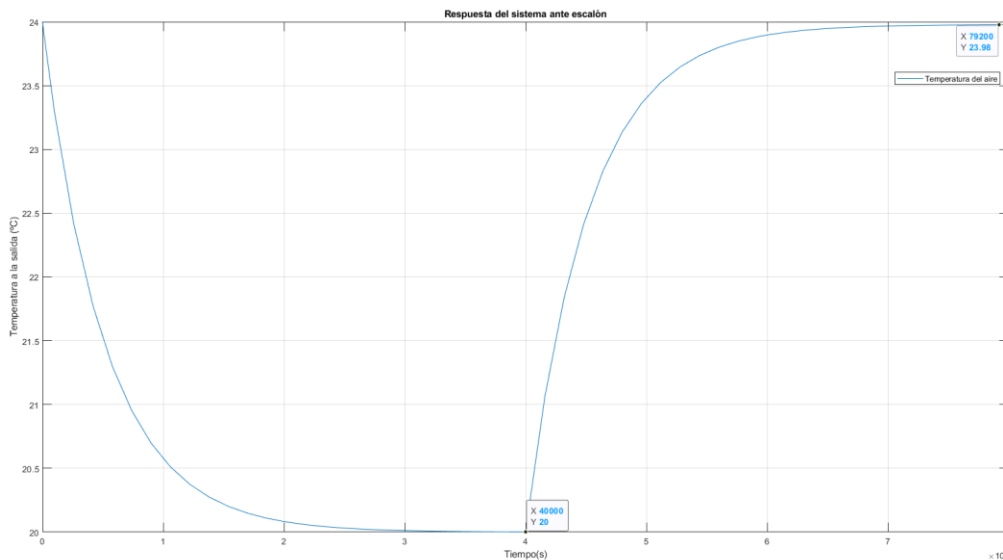


Figura 3.20: Respuesta del sistema ante escalón

3.3.2 Modelado y sincronización del control

3.3.2.1 Tanque

Para este subsistema, de forma similar al tanque estudiado previamente, se ha utilizado un control todo-nada que abre y cierra las válvulas de entrada y salida.

3.3.2.2 Sistema de control de temperatura

Para la sincronización del control se ha hecho un análisis de la característica estática que nos indica que rangos de temperaturas podemos alcanzar teniendo en cuenta que las temperaturas que se van a requerir están entre los 20 y 60°C. Los resultados son los mostrados en la Tabla 3.2.

Tabla 3.2: Análisis característica estática del sistema

Tensión aplicada	Temperatura de salida
0 V	20 °C
0.1 V	24 °C
0.2 V	27.96 °C
0.4 V	35.92 °C
0.6 V	43.87 °C
0.8 V	51.83 °C
10 V	59.79 °C

Dentro de este rango de temperaturas se escoge como punto de operación una tensión de 0.1V (V_0) por lo que la temperatura del punto de operación es de 24°C (T_0).

A continuación, se hará un análisis temporal en bucle abierto en torno al punto de operación (Figura 3.21) aplicando un escalón entre 0.1 y 0.3 V para estimar un modelo simplificado del que poder obtener un sistema de control en bucle cerrado.

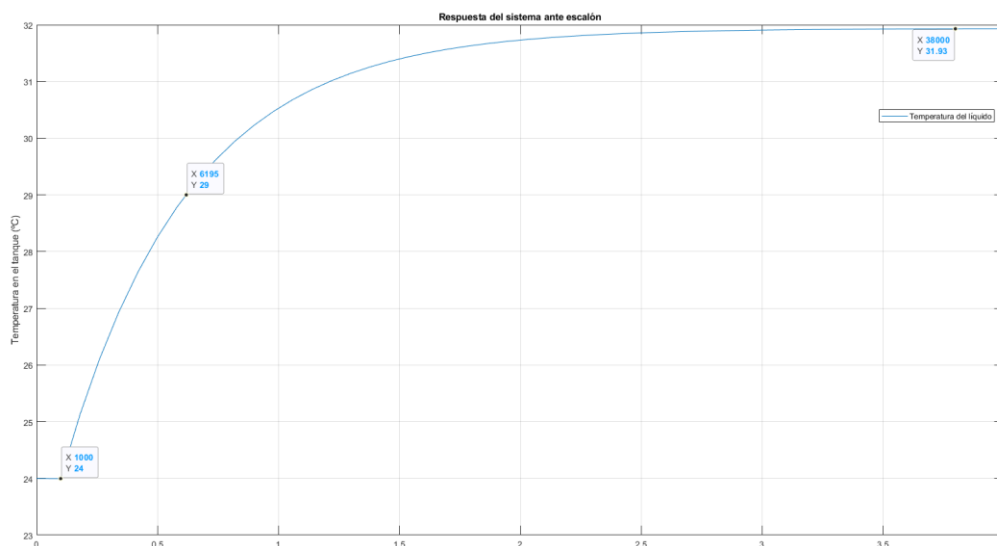


Figura 3.21: Respuesta del sistema ante escalón

Haciendo un análisis similar al hecho previamente con el sistema de control de temperatura se obtienen unos valores de $K_e = 39.65$ y $\tau = 5195$ s.

3.3.2.3 Control del nivel del tanque

Para controlar el nivel del tanque se ha utilizado un relé que abre la válvula de llenado cuando se quiere llenar el tanque y la de vaciado cuando se quiere vaciar.

3.3.2.4 Control de la temperatura del líquido

En este apartado primero se seguirán los mismos pasos que en el punto 3.2.2. Primero se experimentará con un control proporcional.

El modelo utilizado es el que se muestra en la Figura 3.22.

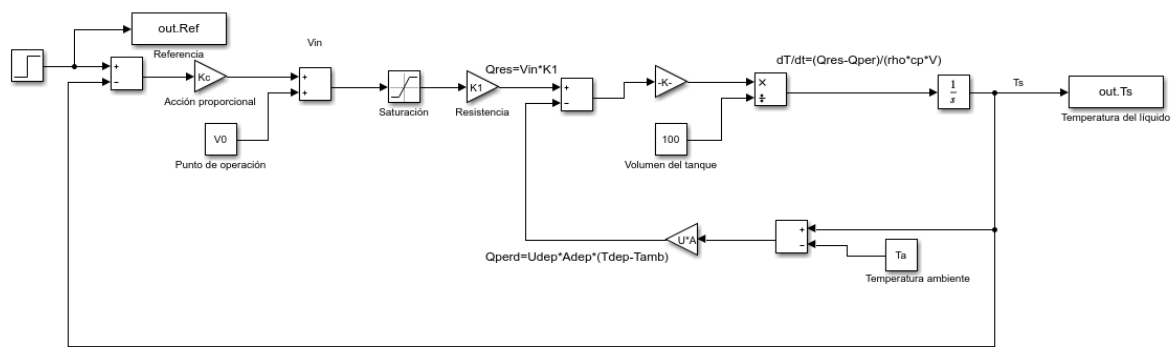


Figura 3.22: Modelo del control de temperatura

Para ver el funcionamiento del control se utilizarán diferentes valores de K_c y se analizará la respuesta del sistema:

- Para un valor de $K_c = 0.5$, en la Figura 2.5, como era de esperar, se observa que la respuesta tiene error en régimen permanente.

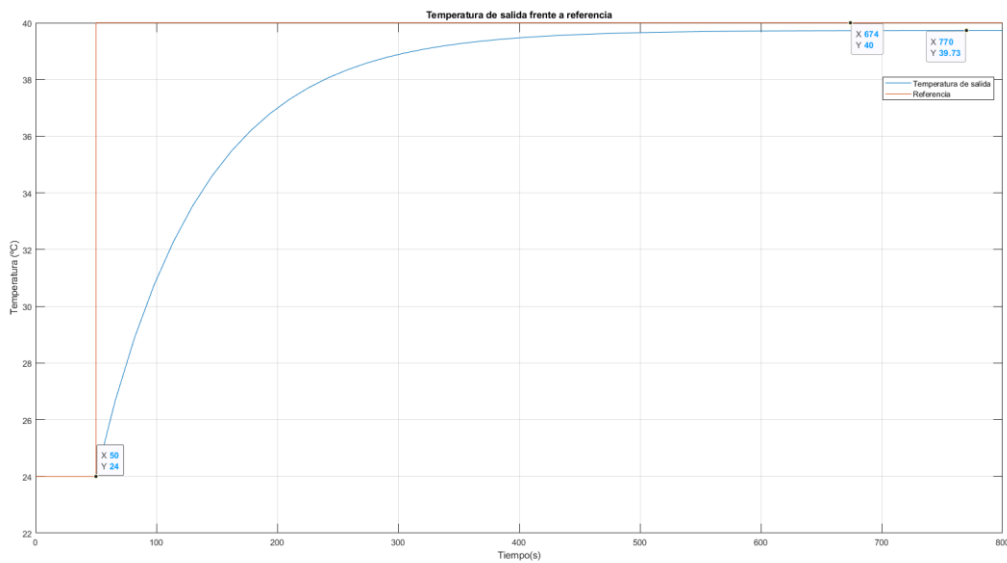


Figura 3.23: Respuesta del sistema en bucle cerrado

- Para un valor de $K_c = 12$, en la Figura 3.24, mejora la velocidad de respuesta pero se sigue observando un pequeño error en régimen permanente.

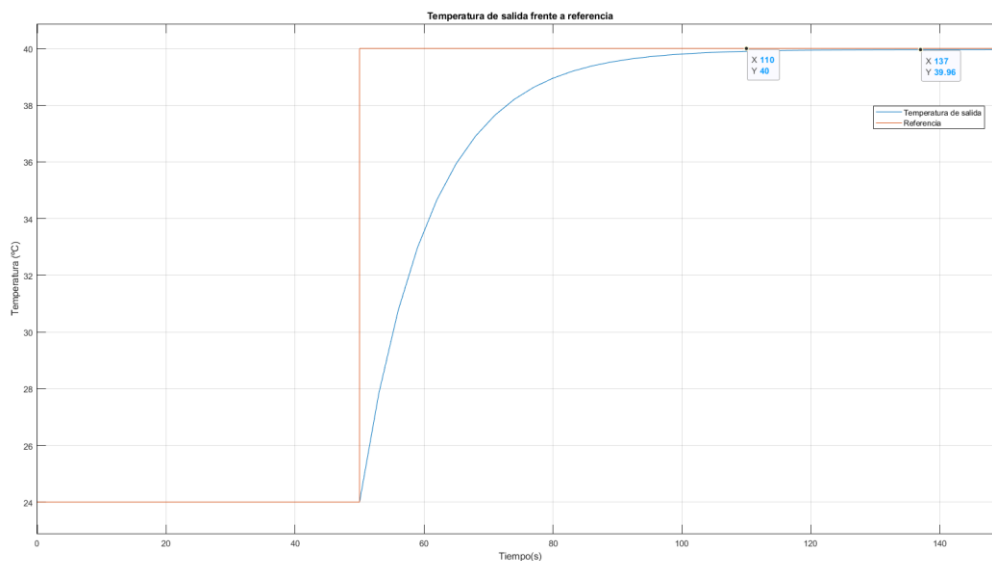


Figura 3.24: Respuesta del sistema en bucle cerrado

- Para un valor de $K_c = 15$, en la Figura 3.25, se sigue observando un pequeño error en régimen permanente pero además, la señal de control satura debido a las limitaciones de tensión aplicada a la entrada.

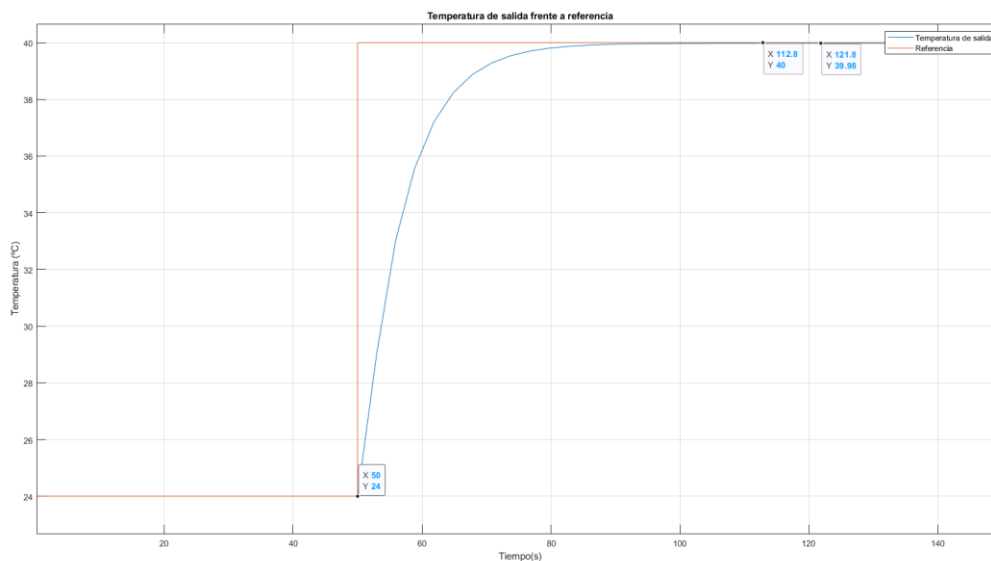


Figura 3.25: Respuesta del sistema en bucle cerrado

- Para eliminar este error en régimen permanente se añade una acción integral del orden de la constante de tiempo (T_i). Para un valor de $K_c = 12$ y $T_i = 5000$. En la Figura 3.26, se comprueba que se ha eliminado el error.

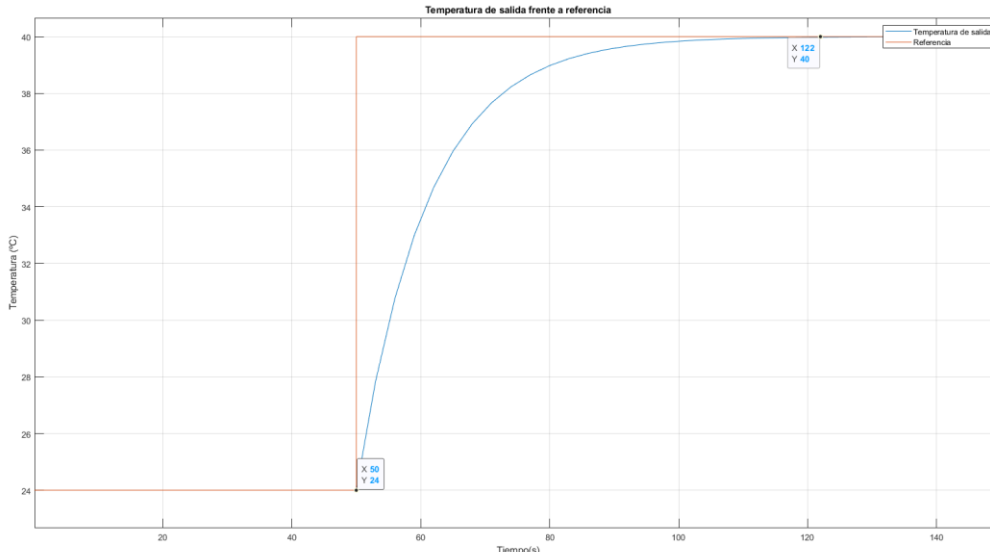


Figura 3.26: Respuesta del sistema en bucle cerrado

Este control se implementará luego en CoDeSys en punto 4.3.

En la Figura 3.27 se tiene el sistema final utilizado, incluyendo sensores y actuadores. Dentro del subsistema tanque se encuentra el modelo desarrollado en este apartado.

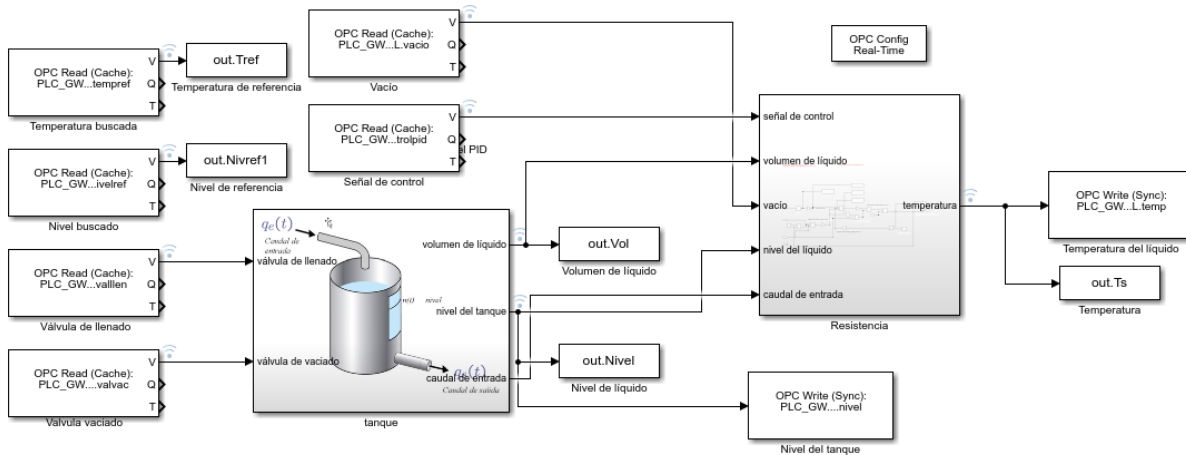


Figura 3.27: Simulación final del sistema

Por otro lado, dentro del subsistema resistencia (Figura 3.28) se ha añadido el modelo desarrollado al que se le ha añadido un término que modela las pérdidas que sufrirá el sistema cuando se caliente el líquido del interior del tanque a la vez que se llena de líquido a una temperatura de entrada T_{ent} . Además, se añade una pequeña constante que se suma al volumen del líquido para evitar indeterminaciones.

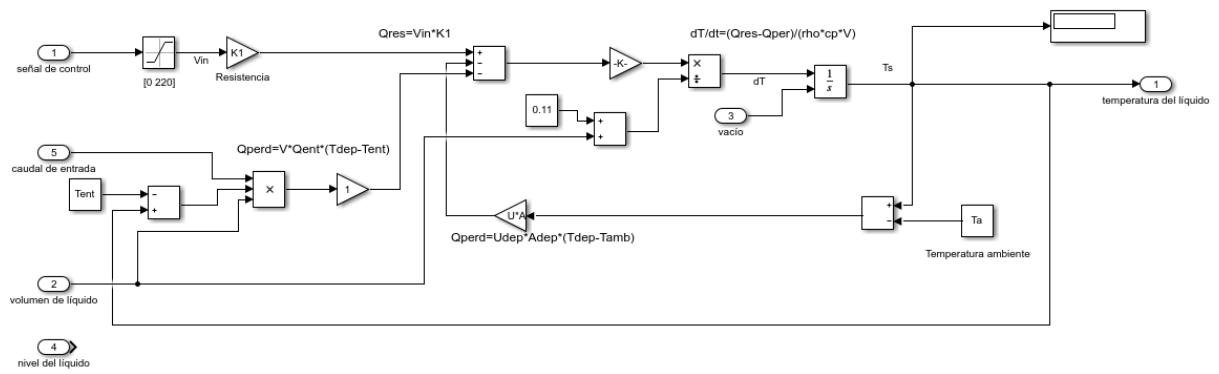


Figura 3.28: Simulación del subsistema resistencia

4 CONTROL PLC

En este capítulo se explicará el control del PLC hecho con el software CoDeSys para cada uno de los sistemas descritos anteriormente.

4.1 Tanque

Como se ha explicado en el 3.1.2, para este sistema se ha utilizado un control todo-nada. Para la histéresis, se ha optado por un valor del 10% del nivel buscado.

Se ha añadido una máquina de estados que permitirá trabajar en un modo automático o manual.

- Para el modo automático, el tanque se llena hasta que se alcanza la referencia.
- Para el modo manual, como se verá en el punto 5.2, el operario podrá desde el escada, enviar una señal al PLC para que abra o cierre la valvula.

4.1.1 Módulos utilizados

Para este sistema se han utilizado los siguientes módulos:

- GVL: Lista de variables globales utilizadas.
- Programa principal (PLC_PRG): Programado en LD, será el que habilite el modo automático cuando sea necesario y llamará a los módulos necesarios.
- Automático: Programado en ST, incorpora el control on-off, descrito previamente, que se utilizará cuando se trabaje en este modo.
- Sensores: Programado en ST, indicará que el tanque está lleno cuando se alcance el nivel buscado y vacío cuando el nivel sea 0.
- Actuadores: Programado en ST, abrirá la valvula cuando se quiera llenar el tanque.

4.1.2 Códigos utilizados

En este punto se mostrarán los códigos utilizados para los distintos módulos del sistema tanque.

4.1.2.1 GVL

La lista de variables globales (Figura 4.1) se utilizará para definir las variables que sean necesarias para los distintos módulos. Este módulo se utilizará de forma similar para los otros sistemas.

```

1
2  VAR_GLOBAL
3     nivel: REAL := 0.0;
4     nivelref: REAL := 0.0;
5     valllen: BOOL;
6     vacio: BOOL;
7     lleno: BOOL;
8     llenar: BOOL;
9  END_VAR

```

Figura 4.1: Módulo “GVL”

4.1.2.2 Programa principal

El programa principal (Figura 4.22) llamará a los módulos necesarios según el modo en el que se opere. Desde el SCADA, como se verá en el punto 5.2, se activa o desactiva la variable auto. Cuando está activada, el sistema entra en modo automático. Si está desactivada, desde SCADA se escriben valores para abrir o cerrar, desde el PLC, la válvula de llenado.

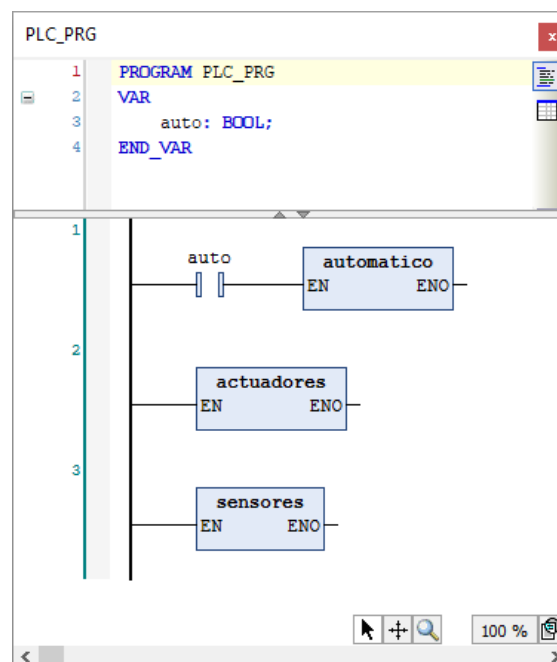


Figura 4.2: Programa principal

4.1.2.3 Sensores y actuadores

Los módulos sensores (Figura 4.44) y actuadores (Figura 4.33) simulan el funcionamiento de los sensores y actuadores que incluya el diseño del sistema. Estos módulos se integrarán en los otros sistemas de forma similar.

```
sensores
1 PROGRAM sensores
2 VAR
3 END_VAR
4
5 IF nivel <= 0 THEN
6     vacio := TRUE;
7     lleno := FALSE;
8 ELSIF nivel >= nivelref THEN
9     vacio := FALSE;
10    lleno := TRUE;
11 ELSE
12     vacio := FALSE;
13     lleno := FALSE;
14 END_IF
```

Figura 4.4: Módulo sensores

```
actuadores
1 PROGRAM actuadores
2 VAR
3 END_VAR
4
5 IF llenar = TRUE THEN
6     valllen := TRUE;
7 ELSIF llenar = FALSE THEN
8     valllen := FALSE;
9 END_IF
```

Figura 4.3: Módulo actuadores

4.1.2.4 Automático

El módulo automático (Figura 4.5) controlará el funcionamiento del sistema cuando se opera en este modo para que cumpla las especificaciones.

```
automatico
1 PROGRAM automatico
2 VAR
3 END_VAR
4
5 IF nivel = nivelref THEN
6     llenar := FALSE;
7 ELSIF nivel < nivelref-(nivelref*0.1) THEN
8     llenar := TRUE;
9 ELSIF nivel > nivelref THEN
10    llenar := FALSE;
11 END_IF;
```

Figura 4.5: Módulo automático

4.1.3 Resultados obtenidos

Los resultados obtenidos al simular el sistema en modo automático son los mostrados en la Figura 4.6.

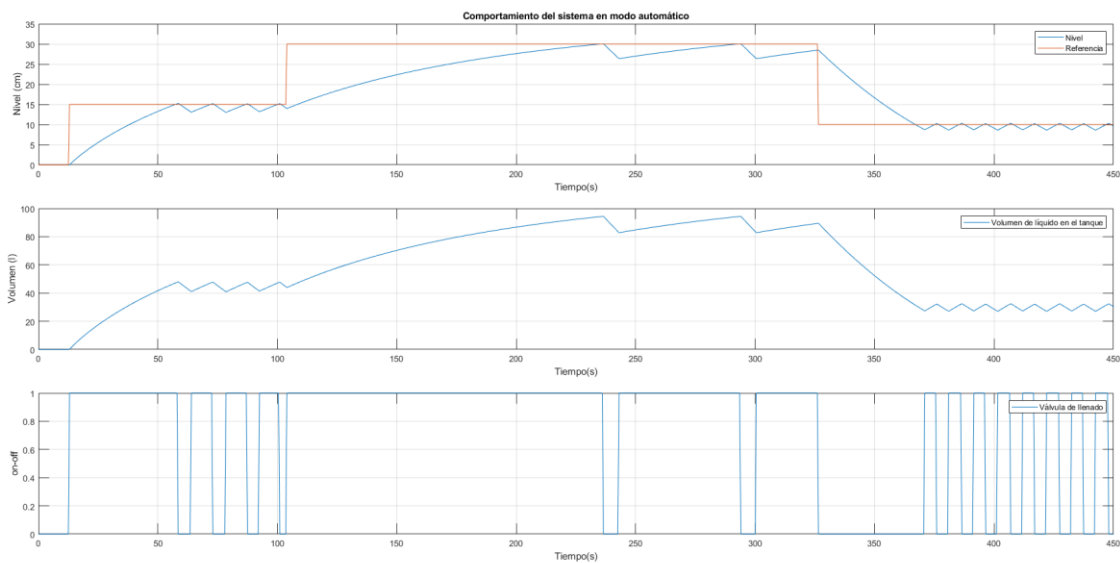


Figura 4.6: Comportamiento del tanque en modo automático

Los resultados al simular el sistema en modo manual se pueden observar en la Figura 4.7.

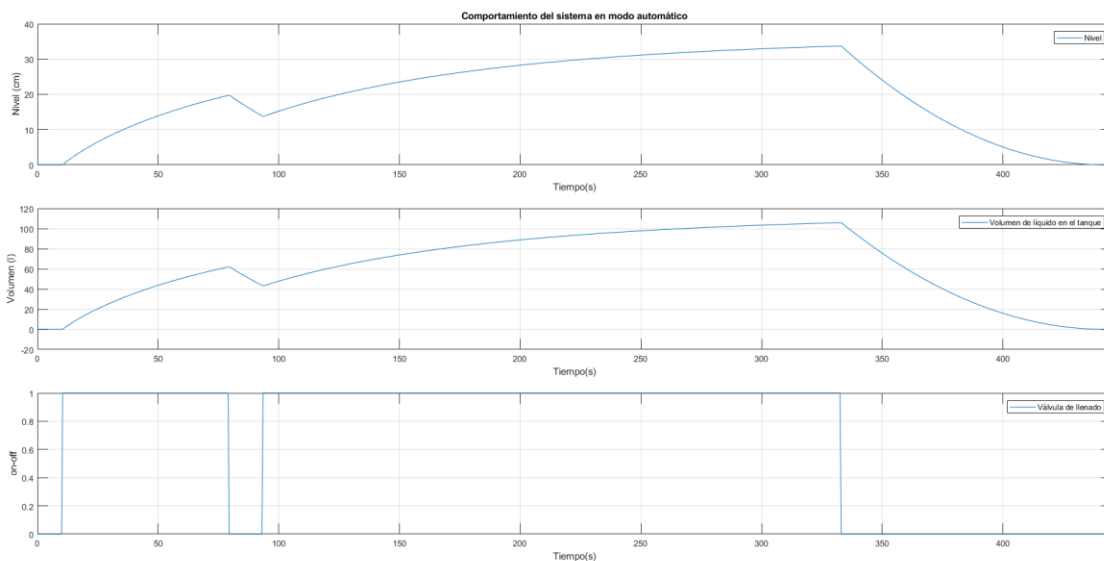


Figura 4.7: Comportamiento del tanque en modo manual

4.2 Sistema de control de temperatura

Como se ha explicado en el punto 3.2.2, para este sistema se ha utilizado una acción de control proporcional con integral.

Este control aumentará la tensión de entrada a las puertas de la resistencia hasta lograr que la temperatura del aire a la salida sea la buscada.

4.2.1 Módulos utilizados

Para este programa se han utilizado los siguientes módulos:

- GVL: Lista de variables globales utilizadas.
- Programa principal (PLC_PRG): Programado en LD, integrará un controlador PID de la biblioteca útil de CoDeSys con los valores obtenidos en el punto 3.2.2.2 y llamará al modulo de sensores.
- Sensores: Programado en ST, indicará si el aire ha alcanzado la temperatura buscada a la salida.

4.2.2 Códigos utilizados

En este apartado se explicará el PID utilizado en el programa principal de este sistema. Los módulos sensores y GVL se ha programado de forma similar al utilizado en el sistema tanque. El código utilizado en el programa principal es el que se muestra en la Figura 4.8.

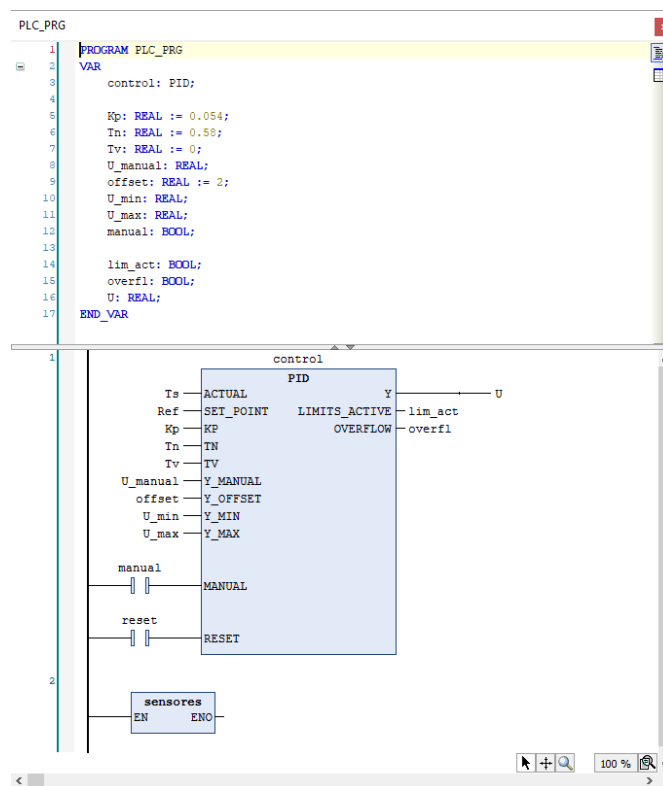


Figura 4.8: Programa principal

El bloque PID utilizado es el que se incluye en la biblioteca útil. La documentación del bloque se muestra en la Figura 4.9.

Scope	Name	Type	Initial	Comment
Input	ACTUAL	REAL		actual value, process variable
	SET_POINT	REAL		desired value, set point
	KP	REAL		proportionality const. (P)
	TN	REAL		reset time (I) in sec
	TV	REAL		rate time, derivative time (D) in sec
	Y_MANUAL	REAL		Y is set to this value as long as MANUAL = TRUE
	Y_OFFSET	REAL		offset for manipulated variable
	Y_MIN	REAL		minimum value for manipulated variable
	Y_MAX	REAL		maximum value for manipulated variable
Output	MANUAL	BOOL		TRUE: manual: Y is not influenced by controller, FALSE: controller determines Y
	RESET	BOOL		reset: set Y output to Y_OFFSET and reset integral part
	Y	REAL		manipulated variable, set value
	LIMITS_ACTIVE	BOOL	FALSE	true set value would exceed limits Y_MIN, Y_MAX
	OVERFLOW	BOOL	FALSE	overflow in integral part

Figura 4.9: Documentación bloque PID

A este bloque se le añadirán las variables del control y el punto de operación calculados previamente. Además, se puede controlar la referencia buscada y encender o apagar el control utilizando el reset incluido. Esto se hará desde el SCADA como se verá en el punto 5.3.

Este bloque se utilizará de forma similar en el sistema tanque con control de temperatura.

4.2.3 Resultados obtenidos

Se ha hecho la simulación para dos valores distintos de K_c . Por un lado, se ha utilizado un valor de 0.027 al principio de la simulación, como se observa en la Figura 4.10, el resultado lento pero no sobreoscila. Después, para un valor de 0.054, la respuesta del Sistema es más rápida pero sobreoscila un poco. Estos resultados concuerdan con el estudio hecho previamente. Además, se puede observar tanto el retraso en la medida de la temperatura de salida como la temperatura de referencia y la temperatura mínima que se puede obtener.

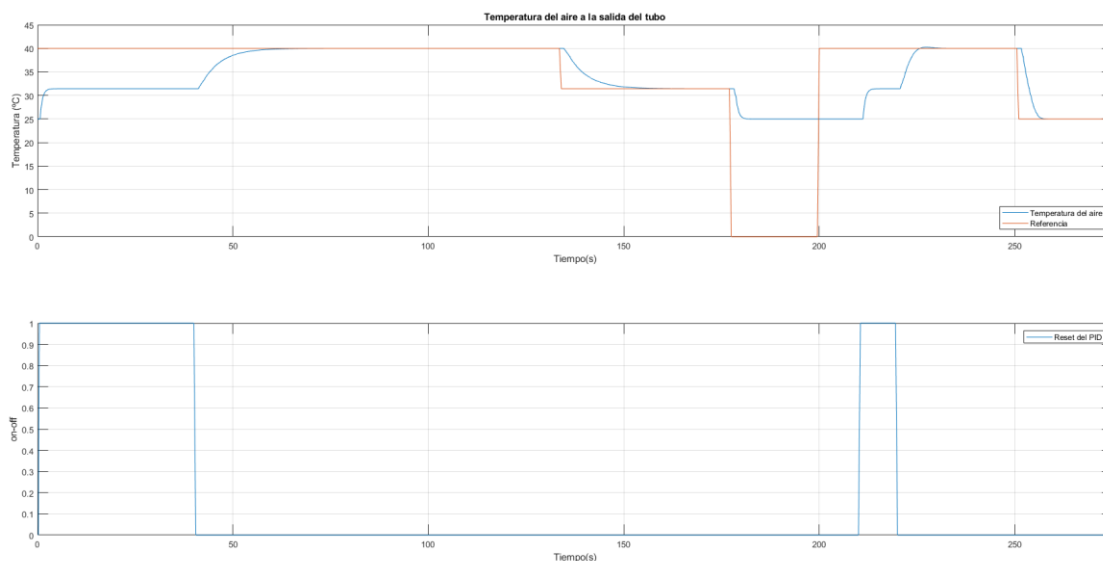


Figura 4.10: Comportamiento del sistema de control de temperatura

4.3 Tanque con control de temperatura

Como se ha explicado en el punto 3.3.2, este sistema consta de dos subsistemas. Por un lado, el control del nivel de líquido se hará de controlando las valvulas de llenado y vaciado como se ha visto en el punto 3.3.2.3. El control de la temperatura del líquido se hará con un controlador PI como se ha estudiado en el punto 3.3.2.4.

Se ha añadido una máquina de estados que permitirá trabajar en 3 modos distintos.

- En modo automático se tiene un control secuencial que consiste en que se abre la valvula de llenado hasta alcanzar el nivel buscado. Una vez alcanzado, se active el pid que calienta el líquido. Cuando se alcanza la temperatura buscada, se espera unos Segundos y se deja enfriar unos grados. Después, se abre la valvula de vaciado hasta que se vacía el tanque.
- En modo manual, como se verá en el apartado tal, el operario podrá enviar desde el SCADA señales para activar y desactivar los distintos actuadores.
- En modo reinicio, el tanque se vaciará, se llenará hasta un nivel fijado y se volverá a vaciar. Este modo se utilizará para pasar del modo manual al automático.

4.3.1 Módulos utilizados

Para este programa se han utilizado los siguientes módulos:

- GVL: Lista de variables globales utilizadas.
- Programa principal (PLC_PRG): Programado en LD, será el que incorpora la máquina de estados descrita llamando a los módulos necesarios.
- Automático: Programado en SFC, controla el funcionamiento del modo automático descrito anteriormente.
- Reiniciar: Programado en SFC, controla el reinicio del Sistema descrito anteriormente.
- PID: Bloque funcional programado en LD, Implementa el control de temperatura del líquido desarrollado previamente con el bloque pid de la biblioteca útil de CoDeSys.
- Sensores: Programado en ST, indicará que el tanque está lleno cuando se alcance el nivel buscado, vacío cuando el nivel sea 0 y que el líquido está caliente cuando se alcance la temperatura buscada.
- Actuadores: Programado en ST, abrirá la valvula de llenado o de vacío cuando se quiera llenar o vaciar, respectivamente, el tanque y activará el pid cuando se quiera calendar el líquido.

4.3.2 Códigos utilizados

4.3.2.1 Programa principal

En este programa se integran, por un lado, las llamadas a los sensores y actuadores, y por otro lado la máquina de estados. El sistema trabajará en modo automático cuando las variables auto y reinicio sean verdaderas. El valor de auto se escribe desde el SCADA y el valor de reinicio se hace falso cuando se entra en modo manual y verdadero cuando se termina el modo reiniciar. De esta forma, cuando se esté trabajando en modo manual y se quiera pasar a automático, habrá que pasar por el estado reiniciar para activar la variable reinicio.

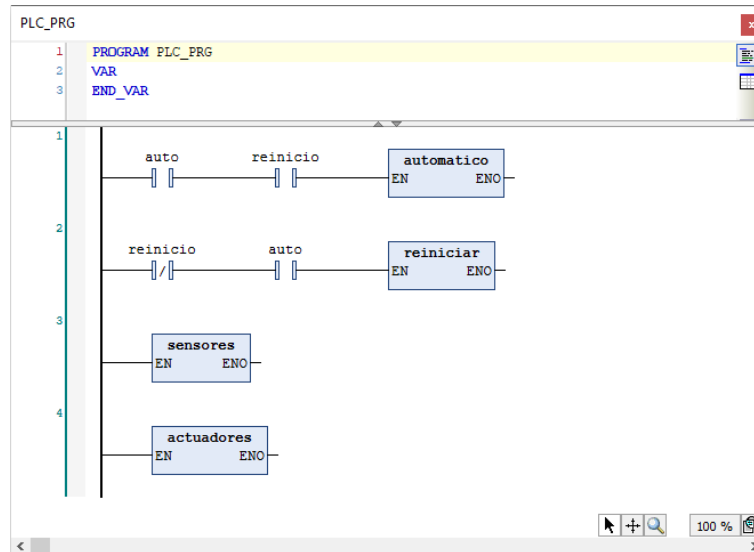


Figura 4.11: Programa principal

4.3.2.2 Automático

En este modo se programa la secuencia explicada previamente en lenguaje SFC para hacerlo más visual. Se ha incorporado una salida en todos los pasos para, en caso de pasar a modo manual en medio del programa, se vuelva a comenzar por el principio cuando se vuelva a activar este modo. Para ello se ha utilizado una variable "GVL.cambioestado_auto" que se hará falsa en el modo reinicio previo a este modo. Se puede observar que en este modo se hace falsa al inicio la variable "GVL.cambioestado_rein" que tiene un funcionamiento análogo, pero para el modo reinicio.

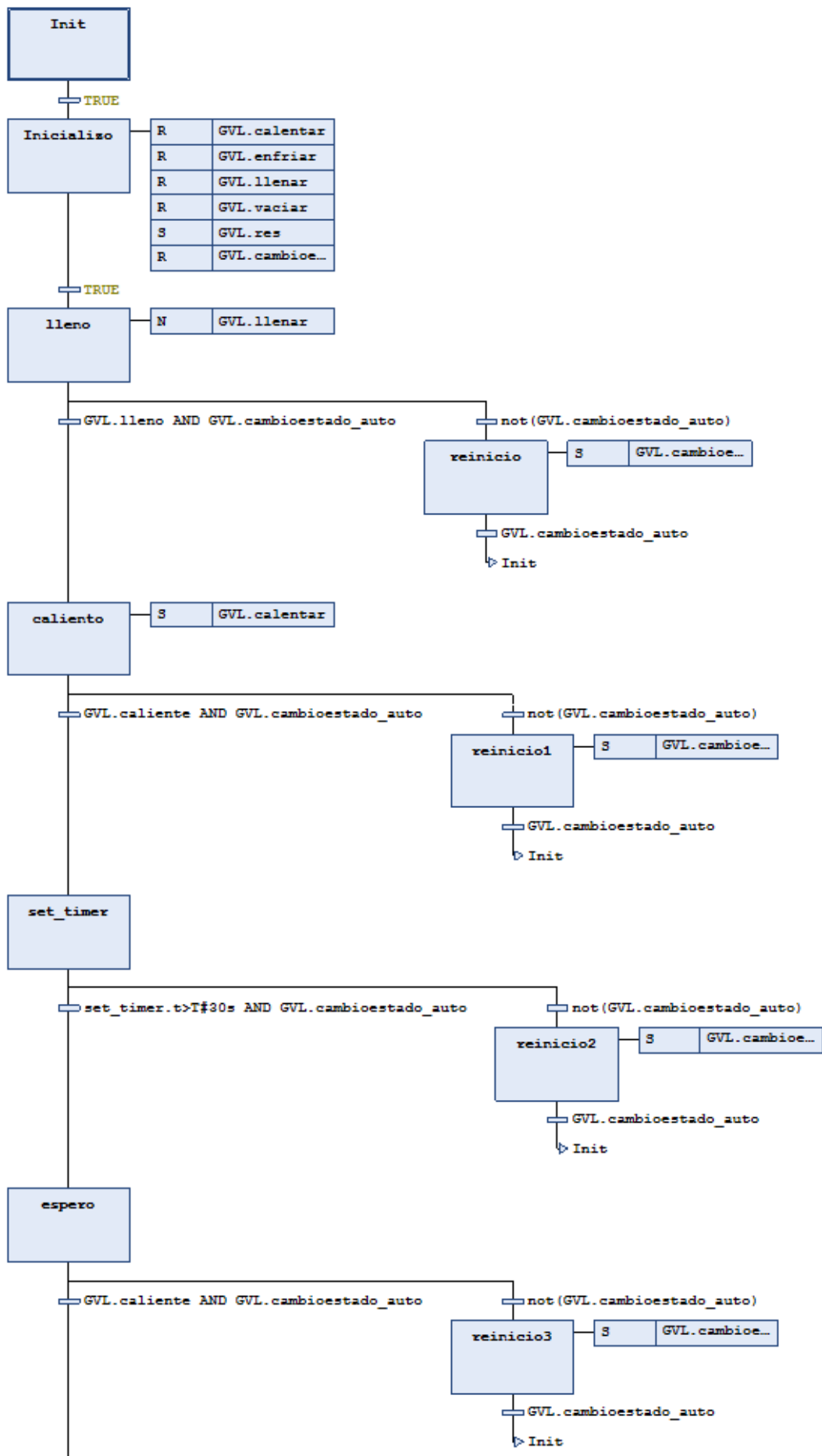


Figura 4.12: Programa automático

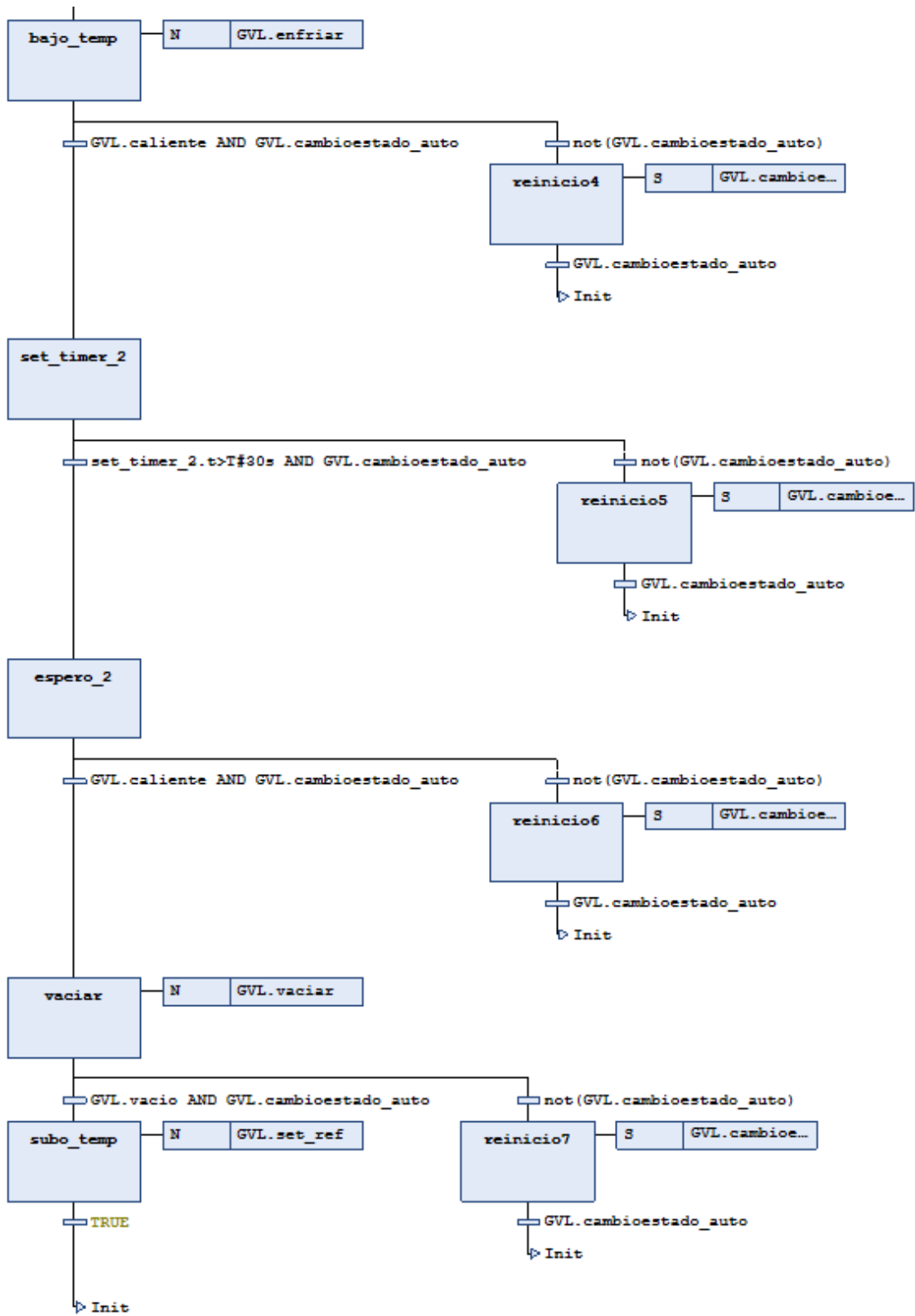


Figura 4.13: Programa automático

4.3.2.3 Reiniciar

En este modo se programa la secuencia explicada previamente en lenguaje SFC. En la **¡Error! No se encuentra el origen de la referencia.** se puede observar la salida en todos los pasos con el mismo fin que el explicado

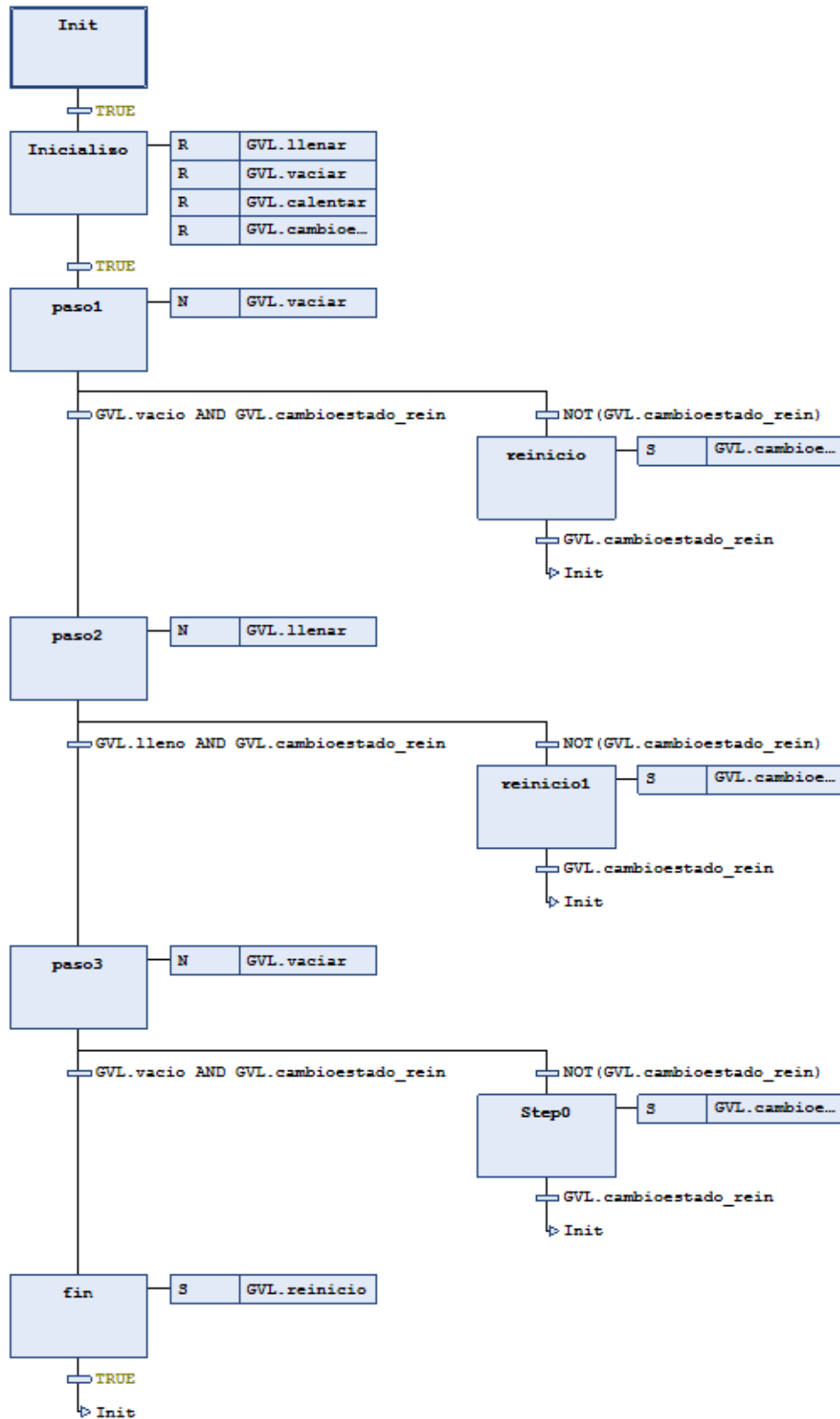


Figura 4.14: Programa reiniciar

4.3.2.4 Actuadores

La particularidad de este modo es la llamada que hace al PID. Al tratarse de un bloque funcional, hay que incluir en la llamada las variables de entrada que este necesita. En este caso, como se observa en la Figura 4.155, se incluyen la temperatura de referencia, la temperatura actual y el reset. Además, la variable de salida del bloque funcional será la señal de control.

```

actuadores
1 PROGRAM actuadores
2 VAR
3 END_VAR

1 IF llenar = TRUE THEN
2   valleng := TRUE;
3   ELSIF llenar = FALSE THEN
4     valleng := FALSE;
5 END_IF
6
7 IF vaciar = TRUE THEN
8   valvac := TRUE;
9   ELSIF vaciar = FALSE THEN
10    valvac := FALSE;
11 END_IF
12
13 IF calentar = TRUE THEN
14   res := FALSE;
15 ELSE
16   res := TRUE;
17 END_IF;
18
19 IF enfriar = TRUE THEN
20   tempref := tempref-2;
21 END_IF
22
23 IF set_ref = TRUE THEN
24   tempref := tempref+2;
25 END_IF
26
27 controltemp(Ts := temp, Ref := tempref, reset := res, U => controlpid);

```

Figura 4.15: Módulo actuadores

4.3.3 Resultados obtenidos

Los resultados obtenidos al simular el sistema en modo automático son los mostrados en la Figura 4.16, en la que se puede observar que funciona según las especificaciones descritas.

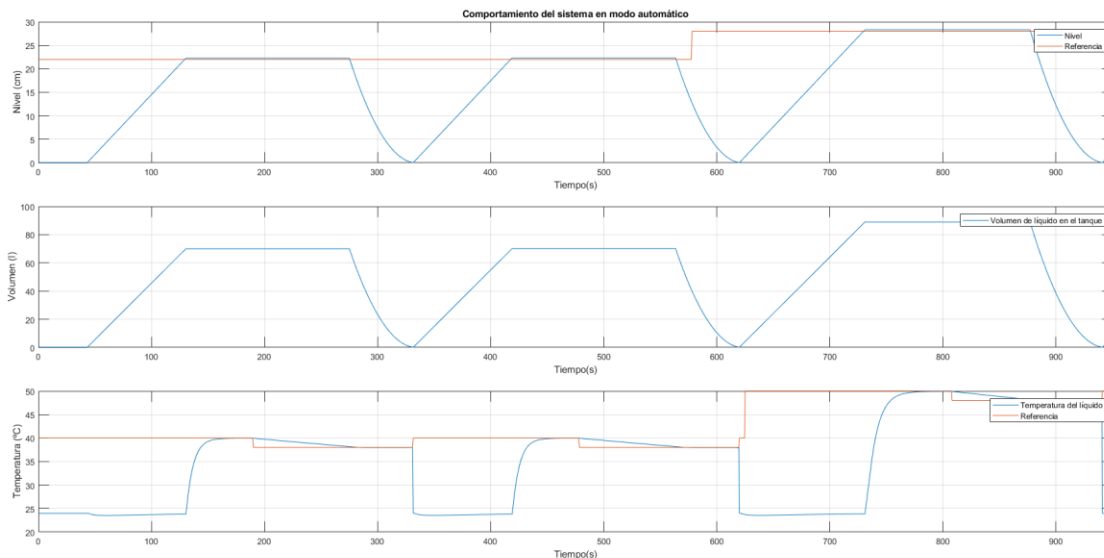


Figura 4.16: Comportamiento del sistema en modo automático los primeros 100 segundos. Luego, antes de pasar a modo manual, pasa por el modo reinicio durante los siguientes 110 segundos aproximadamente.

Se observa que en modo manual no importa el nivel de referencia ya que el control se hace desde el SCADA.

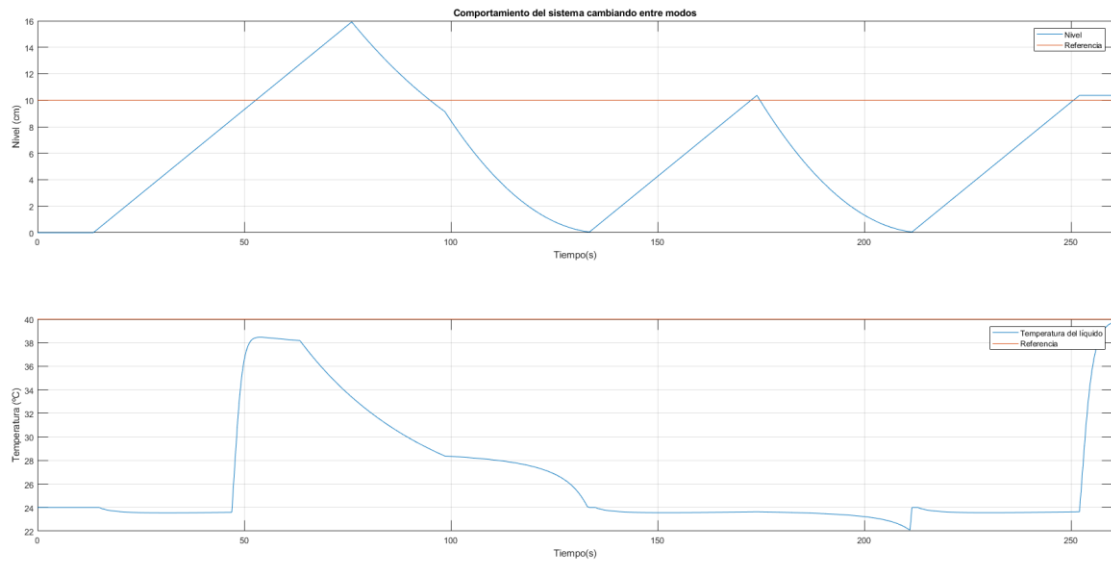


Figura 4.17: Comportamiento del sistema

5 SCADA

En este capítulo se describirá el SCADA diseñado con Vijeo Citect para los distintos sistemas estudiados anteriormente. Se mostrarán las distintas pantallas con las que se controlarán y supervisarán los procesos y los informes generados.

5.1 Guía SCADA

Vijeo Citect es una solución HMI/SCADA para la supervisión y el control de aplicaciones industriales. El objetivo de esta guía es adquirir un buen conocimiento y capaz de utilizar este programa. Se hará especial hincapié en la creación de objetos dentro del editor gráfico y la generación de informes. Todo esto se combinará con la comunicación por OPC vista previamente.

El programa se puede descargar desde la web de software informer (Citect, s.f.). Una vez hecha la instalación se ejecutará “Vijeo CitectScada Explorer”. Esto abrirá los tres programas con los que se trabajará.

- Explorador de Citect: Para crear y gestionar los proyectos. Los otros dos programas se abren y cierran a partir de este.
- Editor de proyectos de Citect: Para crear y manejar la base de datos de Vijeo Citect que contiene las informaciones de configuración del Proyecto.
- Editor gráfico de Citect: Para crear y editar las páginas.
- Además, incluye un Editor Cicode para crear y editar funciones Cicode. En el punto 5.1.1.4 se verá un ejemplo.

Para desarrollar esta guía, se han utilizado las guías de uso de Schneider Electric (Schneider Electric, 2017) (Schneider Electric).

5.1.1 Editor gráfico

En este apartado se hará una guía de cómo se crean todos los elementos que se utilizarán posteriormente para la pantalla de los distintos sistemas.

5.1.1.1 Luces

Para la creación de luces que indiquen que una variable booleana está activa se utilizará el objeto “grupo de símbolos” que se configurará como se muestra en la Figura 5.1.

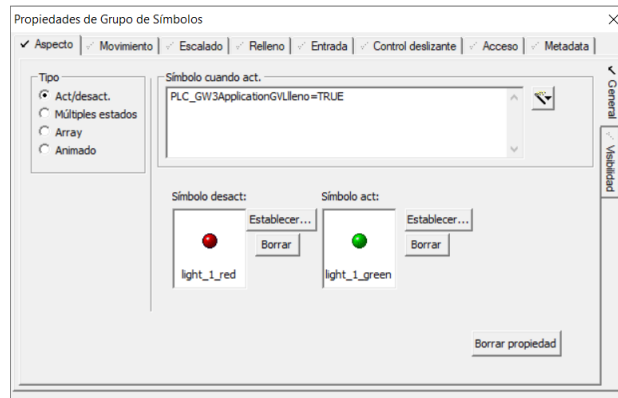


Figura 5.1: Propiedades de “grupo de símbolos”

5.1.1.2 Texto

Para la visualización de variables numéricas se utilizará un objeto “texto” que se configurará como se muestra en la Figura 5.3. Además, si se quiere controlar el valor de la variable desde el SCADA, se le añadirá la configuración del apartado “entrada” como se muestra en la Figura 5.2.

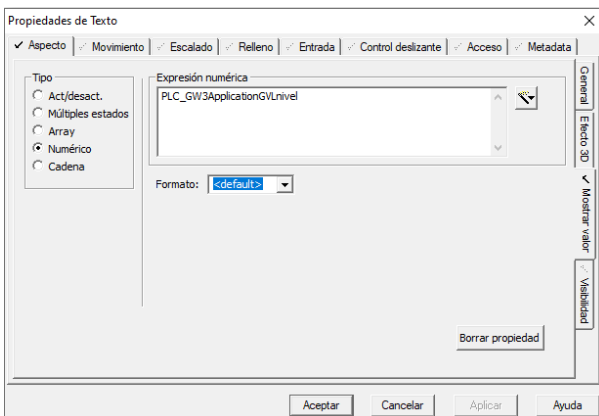


Figura 5.3: Propiedades de “texto” para mostrar datos

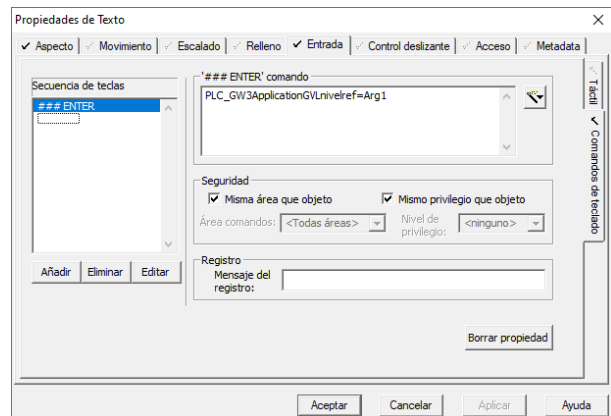


Figura 5.2: Propiedades de “texto” para introducir datos

5.1.1.3 Barra de valor porcentual

Para introducir una barra que muestre el valor porcentual de una variable se utilizará el objeto “rectángulo” al que se le añadirá el rango de valores que se puede alcanzar y se configurará como se muestra en la Figura 5.4 y Figura 5.5.

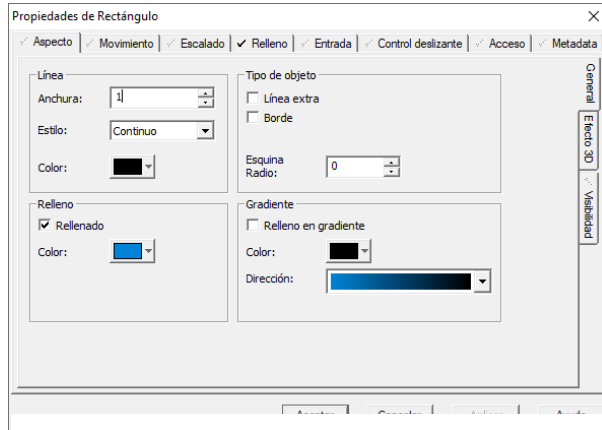


Figura 5.5: Propiedades de “rectángulo”

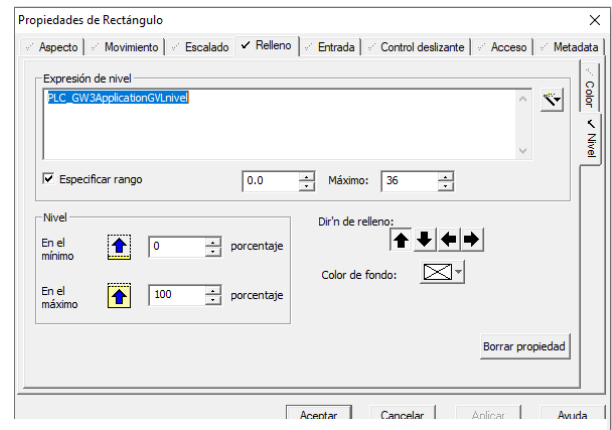


Figura 5.4: Propiedades de “rectángulo”

5.1.1.4 Botón

Para colocar un botón desde donde se podrá controlar el valor de una variable se utilizará el objeto “botón”. Para programar el cambio de variable, se podrá utilizar un comando sencillo como se muestra en la Figura 5.7 o una pequeña función cuando sea necesario como se muestra en la Figura 5.6.

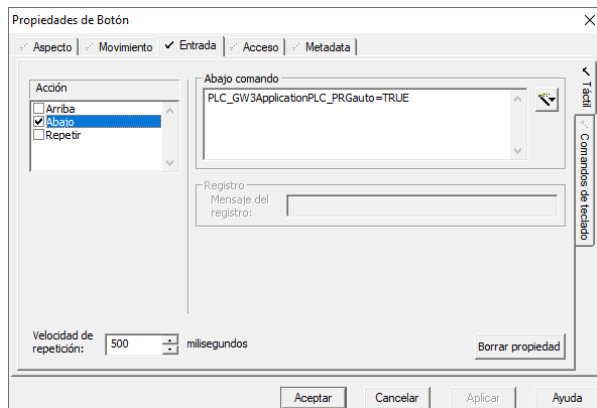


Figura 5.7: Propiedades de “botón”

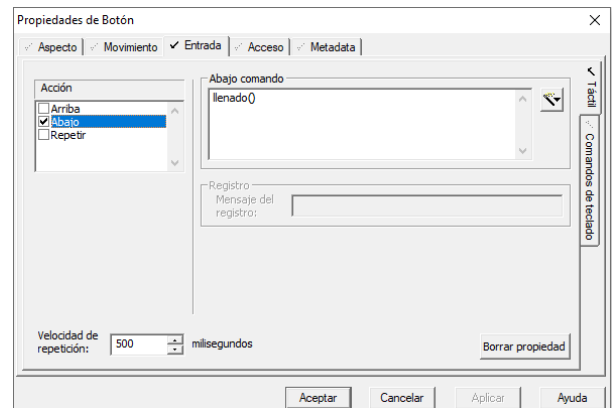


Figura 5.6: Propiedades de “botón”

En este último caso, se creará la función en el apartado “Archivos Cicode” que se programará como se muestra en la Figura 5.8.

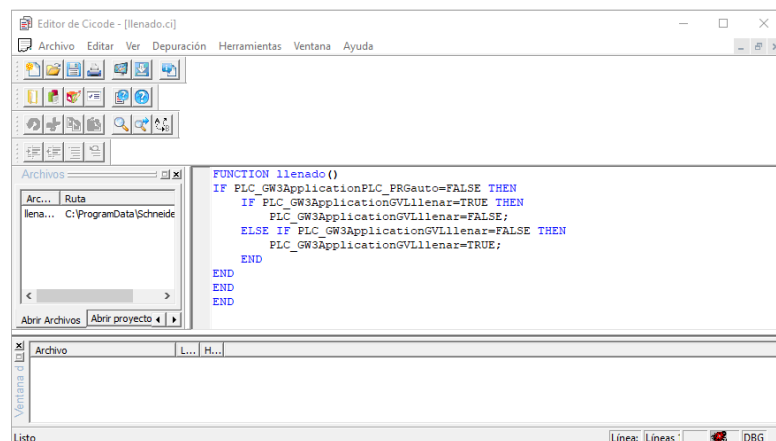


Figura 5.8: Función “llenado()”

5.1.2 Creación de informes

Se utilizarán dos tipos de informes. Por un lado, se registrará en un archivo .txt un mensaje cada vez que se pulse un botón. Además, se creará un informe periódico que registrará el estado del sistema.

5.1.2.1 Registro botones

Para registrar en un archivo de texto un mensaje cada vez que se pulse un botón será necesario crear un dispositivo desde el editor de proyectos. Para ello, se irá a “sistema” apartado “dispositivos” y se configurará como se muestra en la Figura 5.9; **Error! No se encuentra el origen de la referencia..**

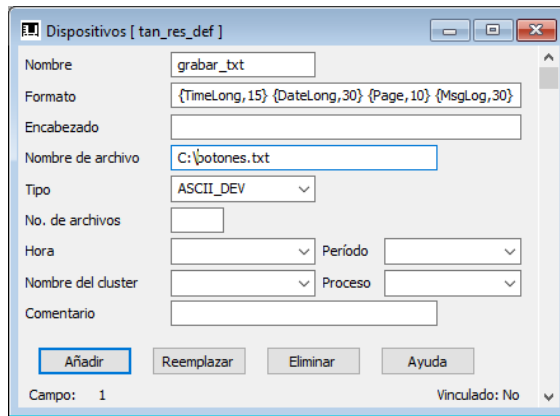


Figura 5.9: Configuración dispositivo

Una vez hecho esto, se configurará los botones para escribir en el archivo un mensaje cada vez que se pulsen como se muestra en la Figura 5.11 y Figura 5.10.

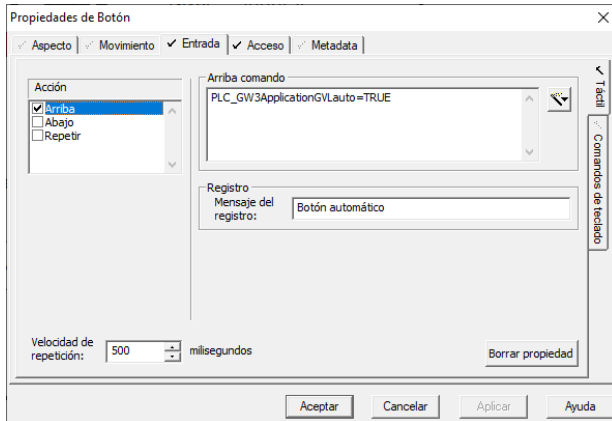


Figura 5.11: Configuración botón

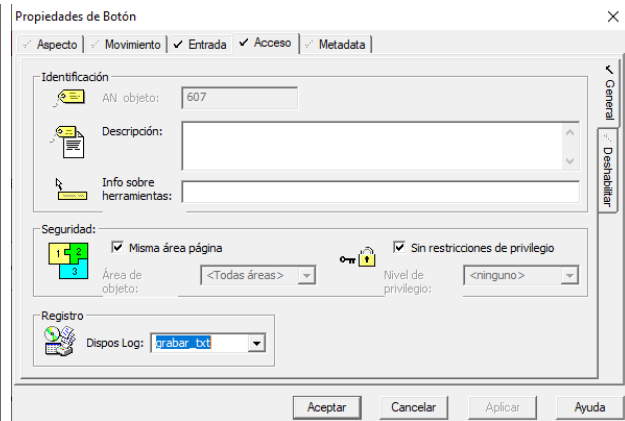


Figura 5.10: Configuración botón

5.1.2.2 Informe periódico

Para crear un informe periódico será necesario crear un dispositivo como se ha hecho en el apartado anterior pero la configuración de este será diferente, como se muestra en la Figura 5.12.

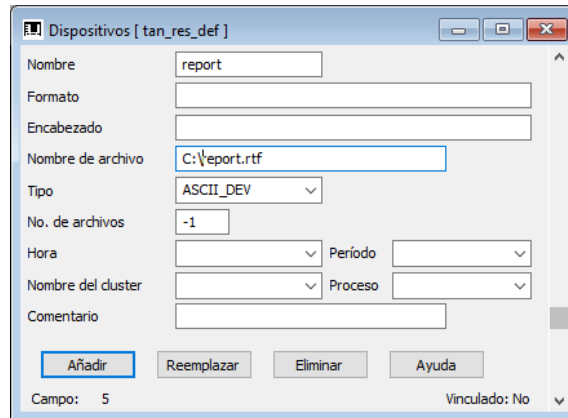


Figura 5.12: Configuración dispositivo

Una vez hecho esto, se creará un informe desde “sistema” apartado “informes” que se configurará como se muestra en la Figura 5.13.

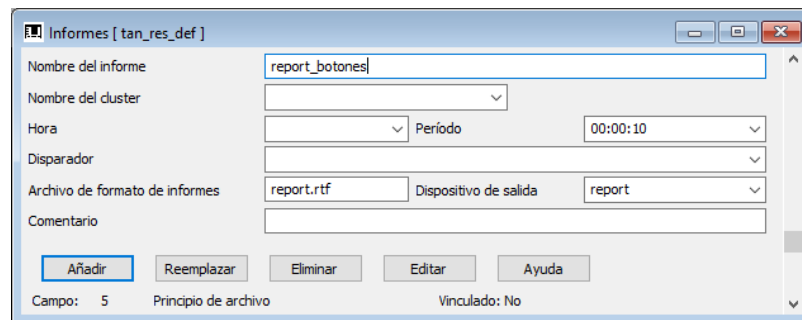


Figura 5.13: Configuración informe

Una vez configurado, se pulsará en “editar” para programar el archivo .rtf del informe. En la Figura 5.14 se muestra el informe realizado para el tanque con control de temperatura que se verá a continuación.

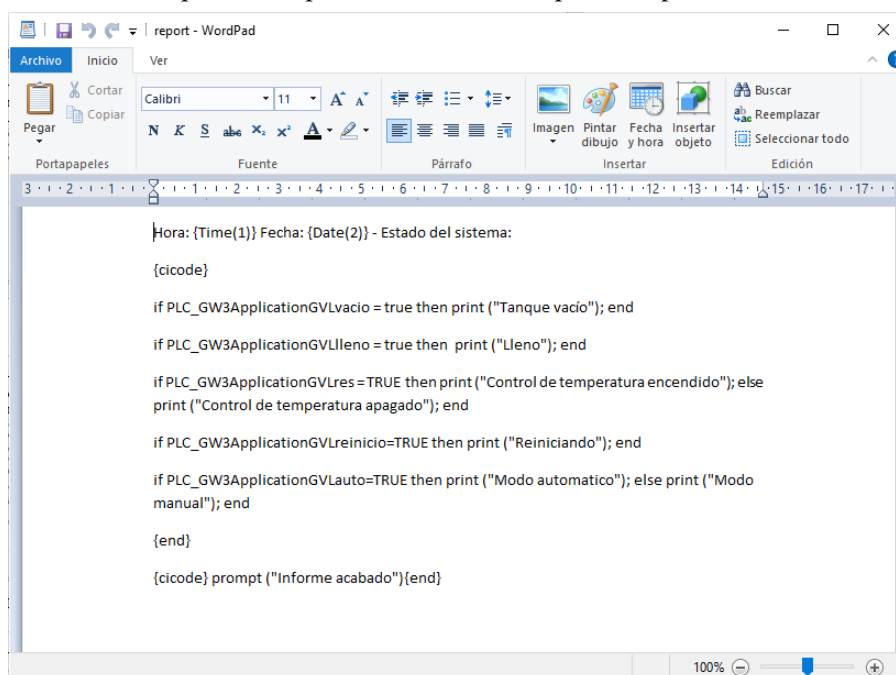


Figura 5.14: Programación informe

Para abrir y visualizar los informes, será necesario realizar una copia de los archivos originales. Es importante no abrir nunca el report.rtf creado por Vijeo ya que, si se abre, el programa dejará de escribir en él.

5.2 Tanque

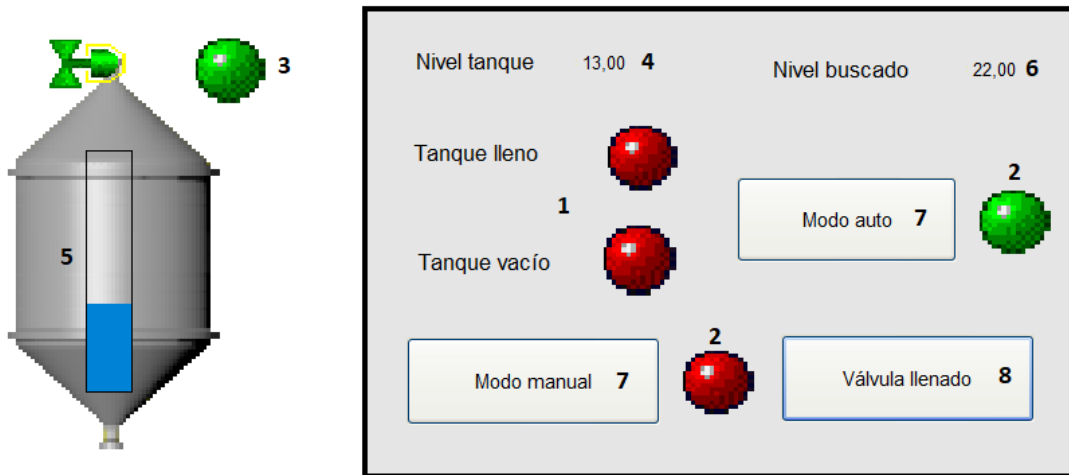


Figura 5.15: Pantalla de supervisión del sistema

5.2.1 Valores mostrados

1. Luces que indican si el tanque está lleno o vacío.
2. Luces que indican el modo en el que se encuentra el control (automático o manual).
3. Luz que indica si la válvula de llenado está abierta o cerrada.
4. Nivel del tanque representado de forma numérica.
5. Nivel del tanque representado de forma porcentual.

5.2.2 Valores introducidos

6. Nivel de líquido buscado.
7. Botones para indicar el modo en el que se quiere trabajar (automático o manual)
8. Botón para abrir la válvula de llenado en el modo manual.

5.3 Sistema de control de temperatura



Figura 5.16: Pantalla de supervisión del sistema

5.3.1 Valores mostrados

1. Luz que indica que la temperatura de salida es la buscada.
2. Luz que indica si el PID está encendido.
3. Temperatura del aire a la salida.

5.3.2 Valores introducidos

4. Temperatura del aire a la salida buscada.
5. Botón para resetear el PID.

5.4 Tanque con control de temperatura

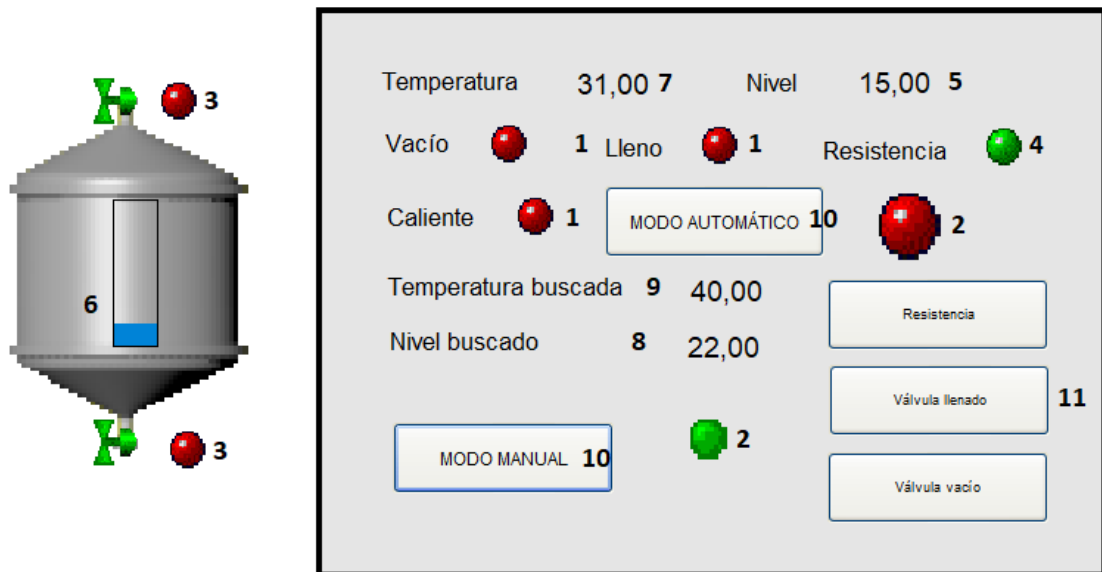


Figura 5.17: Pantalla de supervisión del sistema

5.4.1 Valores mostrados

1. Luces que indican si el tanque está lleno o vacío y si el líquido está a la temperatura buscada.
2. Luces que indican el modo en el que se encuentra el control (automático o manual). Si las dos están apagadas el sistema se encuentra en modo reinicio.
3. Luces que indican si las válvulas de llenado y vaciado están abiertas o cerradas.
4. Luz que indica si la resistencia está encendida.
5. Nivel del tanque representado de forma numérica.
6. Porcentaje del tanque llenado.
7. Temperatura del líquido del interior.

5.4.2 Valores introducidos

8. Nivel de líquido buscado.
9. Temperatura que se quiere alcanzar.
10. Botones para indicar el modo en el que se quiere trabajar (automático o manual)
11. Botones para abrir las válvulas de llenado y vacío y para encender la resistencia en el modo manual.

5.4.3 Informe generado

En la Figura 5.18 y Figura 5.19 se puede observar una muestra de los informes obtenidos para este sistema,

diseñados como se ha indicado previamente.

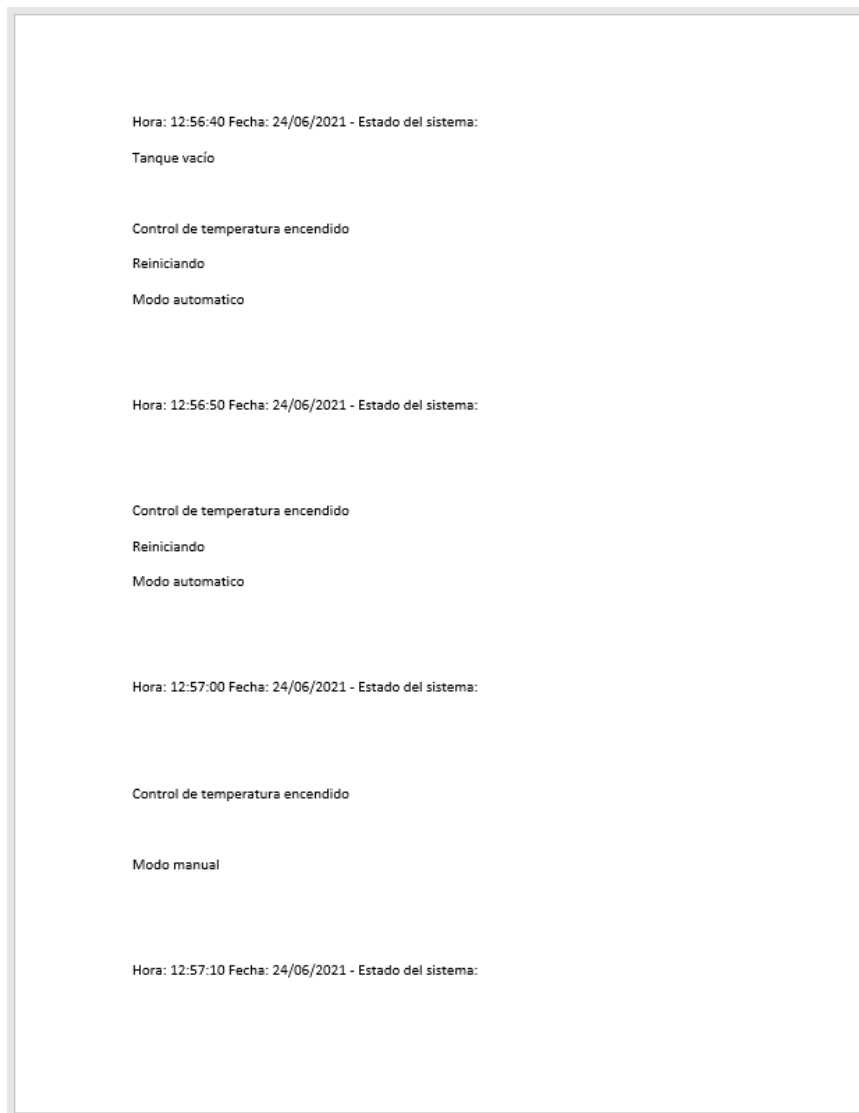


Figura 5.18: Informe periódico

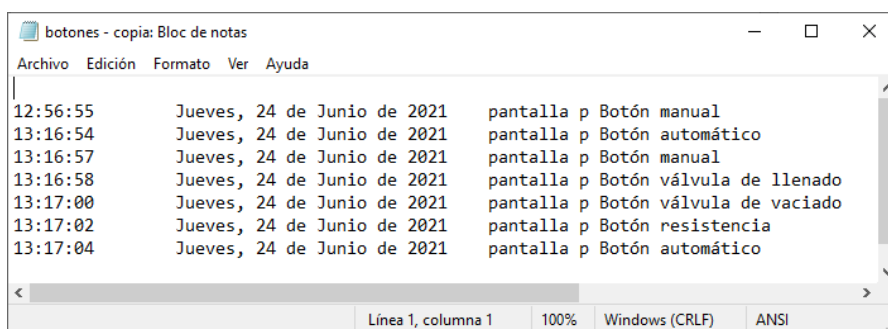


Figura 5.19: Registro de botones

REFERENCIAS

- Ávila Gutiérrez, M. (2020). Modelado y optimización de un sistema de captación.
- Citect. (s.f.). Obtenido de <https://citectscada.software.informer.com/7.4/>
- Contaval. (24 de Mayo de 2017). Obtenido de <https://www.contaval.es/regulacion-onoff/>
- FESTO. (s.f.). Obtenido de https://www.festo.com/net/es-mx_mx/SupportPortal/default.aspx?q=codesys&tab=4&type=75#result
- Gil Ortega Linares, M. (s.f.). Apuntes de Fundamentos de Control.
- MATLAB. (s.f.). Obtenido de <https://es.mathworks.com/products/opc.html>
- Schneider Electric. (Febrero de 2017). *Vijeo Citect User Guide*. Obtenido de https://www.se.com/ww/resources/sites/SCHNEIDER_ELECTRIC/content/live/FAQS/235000/FA235001/en_US/PSE%207.4%20User%20Guide.pdf
- Schneider Electric. (s.f.). Vijeo Citect curso.
- Universitat de València. (s.f.). Obtenido de <https://www.uv.es/idiqlab/labquimicos/documentos/balance%20energia.pdf>
- Vijeo Citect. (s.f.). Curso Vijeo Citect para clientes.