

Trabajo Fin de Grado
Grado en Ingeniería de las Tecnologías
Industriales.

Control de un sistema de regulación de temperatura
usando un autómata programable y una pantalla HMI

Autor: Antonio Jesús Palomo Cantador

Tutor: Francisco Salas Gómez

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

Sevilla, 2019



Trabajo Fin de Grado
Grado en Ingeniería de las Tecnologías Industriales

**Control de un sistema de regulación de temperatura
usando un autómata programable y una pantalla
HMI**

Autor:

Antonio Jesús Palomo Cantador

Tutor:

Francisco Salas Gómez

Profesor titular

Dpto. de Ingeniería de Sistemas y Automática
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla
Sevilla, 2019

Trabajo Fin de Grado: Control de un sistema de regulación de temperatura usando un autómata programable y una pantalla HMI

Autor: Antonio Jesús Palomo Cantador

Tutor: Francisco Salas Gómez

El tribunal nombrado para juzgar el Proyecto arriba indicado, compuesto por los siguientes miembros:

Presidente:

Vocales:

Secretario:

Acuerdan otorgarle la calificación de:

Sevilla, 2019

El Secretario del Tribunal

A mi familia

A mis amigos

Agradecimientos

Me gustaría agradecer a mi familia por su apoyo incondicional en los momentos difíciles que he sufrido durante todo el grado. Sin ellos nada de esto hubiera sido posible.

Cabe destacar la labor de los profesores por su disponibilidad y amabilidad e incluso por las ganas de enseñar en algunos casos. En especial, a mi tutor, por su guía durante todo el desarrollo del proyecto.

Finalmente, a las personas que han aparecido estos últimos años y que están dispuestos a acompañarme en este camino.

El PLC ha supuesto una gran revolución en la industria moderna. Estos dispositivos se han ido imponiendo a los antiguos sistemas de control basados en relés y suponen la base de cualquier sistema de control actual. Este hecho ligado a la evolución de los lenguajes de programación supone la creación de un gran campo de aplicación para estos aparatos.

Se ha decidido crear un sistema de control dirigido por una pantalla HMI que será capaz de monitorizar un lazo de realimentación a través de las instrucciones enviadas al PLC. Con la conexión de ambos aparatos se conseguirá controlar la temperatura de un simulador de procesos PT326 mediante un sistema de control PID.

A través del software desarrollado en TIA Portal y los elementos de conexión facilitados por el Laboratorio del Departamento de Ingeniería de Sistemas y Automática se han podido conectar estos elementos y mostrar unos resultados acordes a los objetivos del proyecto.

Finalmente se tratará de hacer una guía detallada del funcionamiento de la pantalla HMI utilizada de cara a futuras aplicaciones por parte del departamento.

Abstract

The PLC has developed a great revolution in modern industry. These devices have been imposed on the old control systems and are the basis of any current control system. This fact, linked to the evolution of programming languages, implies the creation of a large field of application for these devices.

It has been decided to create a control system directed by an HMI screen that will be able to monitor a feedback loop through the instructions sent to the PLC. With the connection of both devices we will be able to control the temperature of a PT326 process simulator through a PID control system.

Through the software developed in TIA Portal and the connection elements provided by the Laboratory of the Systems and Automation Engineering Department, these elements have been able to be connected and show results according to the objectives of the project.

Finally, we will try to make a detailed guide of the operation of the HMI screen used for future applications by the department.

Agradecimientos	ix
Resumen	xi
Abstract	xiii
Índice	xv
Índice de Tablas	xvii
Índice de Figuras	xix
Notación	xxiii
1 Introducción	1
1.1. <i>Antecedentes</i>	1
1.2. <i>Motivación</i>	2
1.3. <i>Descripción de los objetivos</i>	2
2 Descripción de los componentes	5
2.1. <i>Pantalla HMI</i>	5
2.1. <i>PLC</i>	6
2.2. <i>Sistema de control de temperatura</i>	7
2.3. <i>Elementos de conexionado</i>	9
3 CONFIGURACIÓN DE LOS DISPOSITIVOS	11
3.1. <i>Creación y configuración de un proyecto</i>	11
3.2. <i>Configuración del PLC</i>	13
3.3. <i>Programación de un PLC</i>	14
3.3.1. <i>Bloques de programa</i>	14
3.3.1.1. <i>Bloques de organización (OB)</i>	15
3.3.1.2. <i>Funciones (FC)</i>	15
3.3.1.3. <i>Bloques de función (FB)</i>	16
3.3.1.4. <i>Bloques de datos globales (DB)</i>	16
3.3.2. <i>Lenguajes de programación</i>	17
3.4. <i>Visualización de las variables</i>	17
4 PROGRAMACIÓN FINAL DEL PLC	19
4.1. <i>Variables del PLC</i>	19
4.1.1. <i>Entadas y salidas</i>	19
4.1.2. <i>Marcas del PLC</i>	19
4.2. <i>Bloques de programa de usuario</i>	21
4.2.1. <i>Startup</i>	21
4.2.2. <i>Bloque_PID</i>	23
4.2.2.1. <i>Función de transferencia</i>	24
4.2.2.2. <i>Implementación del bloque PID</i>	25
4.2.2.3. <i>Modos y Estados de operación del bloque PID</i>	25
4.2.3. <i>Escalado</i>	26
4.2.4. <i>Límites</i>	28
4.2.5. <i>Optimizaciones</i>	29

4.2.6	Parámetros	30
4.2.7	Resultados	32
5	PROGRAMACIÓN DE LA PANTALLA	37
5.1	<i>Variables de la pantalla</i>	37
5.2	<i>Herramientas de visualización de la pantalla</i>	38
5.3	<i>Visualización final</i>	40
5.3.1	Pantalla inicial	40
5.3.2	Pantalla de controlador	40
5.3.3	Modo de operación	41
5.3.4	Control de parámetros	42
5.3.5	Implementación	43
5.3.6	Otros parámetros de control	44
5.3.7	Límites	44
5.3.8	Autoajuste	45
5.3.9	Visualización	46
5.3.10	Esquema de distribución de las pantallas	46
6	MANEJO DEL SERVIDOR WEB	49
6.1	<i>Configuración previa</i>	49
6.2	<i>Utilidades</i>	51
6.3	<i>Aplicación del servidor: Resultados</i>	51
6.3.1	Representación de resultados	52
7	CONCLUSIONES	57
7.1	<i>Problemas encontrados</i>	57
7.2	<i>Conclusiones y futuras propuestas</i>	57
	Referencias	59
	Glosario	61
	ANEXO-A	63
A.1	<i>Configuración de la pantalla HMI</i>	63
A.2	<i>Visualización de la pantalla HMI</i>	66
A.2.1	Plantillas	66
A.2.2	Variables HMI	67
A.2.3	Imágenes	67
A.2.4	Herramientas	69
A.2.4.1	Objetos básicos	69
A.2.4.2	Elementos	71
A.2.5	Aspecto	75
A.2.6	Librerías	75

ÍNDICE DE TABLAS

Tabla 3-1. Barra de herramientas de programación	17
Tabla 4-1. Variables del PLC	19
Tabla 4-2. Modos/Estados de operación	25
Tabla 5-1. Variables de la pantalla	37
Tabla 5-2. Objetos básicos de la pantalla	39
Tabla 5-3. Elementos de la pantalla	39

ÍNDICE DE FIGURAS

Figura 1-1. MODICON 084	1
Figura 1-2. Ejemplo de célula automatizada	2
Figura 2-1. Parte delantera Pantalla Siemens	5
Figura 2-2. Parte trasera pantalla Siemens	5
Figura 2-3. Visión frontal Switch Siemens	6
Figura 2-4. Entradas PLC Siemens	7
Figura 2-5. Salidas PLC Siemens	7
Figura 2-6. Visión frontal PT326	8
Figura 2-7. Esquema de control PT326	8
Figura 2-8. Conexionado de la estructura	9
Figura 2-9. Esquema de conexionado	10
Figura 3-1. Primeros pasos de la creación de un proyecto	11
Figura 3-2. Agregar un dispositivo	12
Figura 3-3. Agregar un controlador sin modelo especificado	12
Figura 3-4. Detección de hardware para el PLC	13
Figura 3-5. Configuración IP del PLC	14
Figura 3-6. Bloques de programa	15
Figura 3-7. Acceso a bloques de datos	16
Figura 3-8. Tabla de observación y forzado permanente	18
Figura 4-1. Segmentos 1 y 2 del bloque Startup	22
Figura 4-2. Segmento 3 del bloque Startup	23
Figura 4-3. Bloque PID_Compact	24
Figura 4-4. Función de transferencia	24
Figura 4-5. Implementación del bloque PID	25
Figura 4-6. Bloque de Escalado	27
Figura 4-7. Segmentos 1 y 2 del bloque de Límites	28
Figura 4-8. Segmento 3 del bloque de límites	29
Figura 4-9. Bloque de optimizaciones	30
Figura 4-10. Segmento 1 del bloque de Parámetros	31
Figura 4-11. Segmento 2 del bloque de Parámetros	31
Figura 4-12. Segmento 3 del bloque de Parámetros	32
Figura 4-13. Segmento 4 del bloque de Parámetros	32

Figura 4-14. Bloque de datos globales	33
Figura 4-15. Segmento 1 del bloque de resultados	34
Figura 4-16. Segmentos 2 y 3 del bloque de resultados	34
Figura 4-17. Segmento 4 del bloque de resultados	35
Figura 5-1. Pantalla inicial	40
Figura 5-2. Pantalla de controlador	41
Figura 5-3. Pantalla de modos de operación	42
Figura 5-4. Pantalla de parámetros del controlador	43
Figura 5-5. Pantalla de implementación	43
Figura 5-6. Pantalla de otros parámetros	44
Figura 5-7. Pantalla de límites	45
Figura 5-8. Pantalla de autoajuste	45
Figura 5-9. Pantalla de visualización de variables	46
Figura 5-10. Esquema de interrelación de pantallas	47
Figura 6-1. Habilitación del servidor Web	49
Figura 6-2. Crear usuario del servidor web	50
Figura 6-3. Página web definida por el usuario	50
Figura 6-4. Página inicial del servidor web	51
Figura 6-5. Recogida de resultados	52
Figura 6-6. Resultado en tensión frente a escalones en la actuación	52
Figura 6-7. Resultado en temperatura frente a escalones en la actuación	53
Figura 6-8. Resultado en tensión frente a escalones en la referencia	53
Figura 6-9. Resultado en temperatura frente a escalones en la referencia	54
Figura 6-10. Resultado en tensión frente a perturbaciones	54
Figura 6-11. Resultado en temperatura frente a perturbaciones	55
Figura A-1. Modelo de pantalla HMI	63
Figura A-2. Conexiones de PLC de la pantalla	64
Figura A-3. Imágenes de la pantalla	64
Figura A-4. Interfaz PROFINET del HMI	65
Figura A-5. Conexiones del HMI	65
Figura A-6. Plantilla por defecto	66
Figura A-7. Campos Variables HMI	67
Figura A-8. Imagen	68
Figura A-9 Forma geométrica	69
Figura A-10. Visor de gráficos	70
Figura A-11. Lista de textos	72
Figura A-12. Lista de gráficos	72
Figura A-13. Campo de fecha y hora	73
Figura A-14. Representación mediante barras	74

Figura A-15. Visor de curvas

75

Figura A-16. Librerías en TIA Portal

76

PLC	Controlador Lógico Programable
PID	Proporcional, Integrador y derivativo
HMI	Interfaz hombre-máquina
OB	Bloque de organización
DB	Bloque de datos
FC	Función
FB	Bloque de función
INT	Tipo entero
BOOL	Tipo booleano
REAL	Tipo real

1 INTRODUCCIÓN

En la utopía de ayer, se incubó la realidad de hoy, así como en la utopía de mañana palpitarán nuevas realidades.

- José Ingenieros -

Este capítulo incluye los antecedentes y la motivación que impulsaron a hacer este trabajo y una descripción de los objetivos.

1.1. Antecedentes

La idea de control automático de un sistema se comenzó a incorporar hace más de un siglo con el manejo de los reguladores de velocidad y la dirección automática de los barcos gracias a la invención del girocompás. El primer análisis teórico del control PID se publica en 1922 de la mano de Nicolás Minorsky donde su publicación sobre el uso de controladores de tres términos se empleó en el gobierno automático de buques.

Por su parte, la historia del PLC comienza a finales de los años 60 con el objetivo de reducir el enorme coste que suponía el control basado en relés. El MODICON 084 fue el primer PLC producido comercialmente.

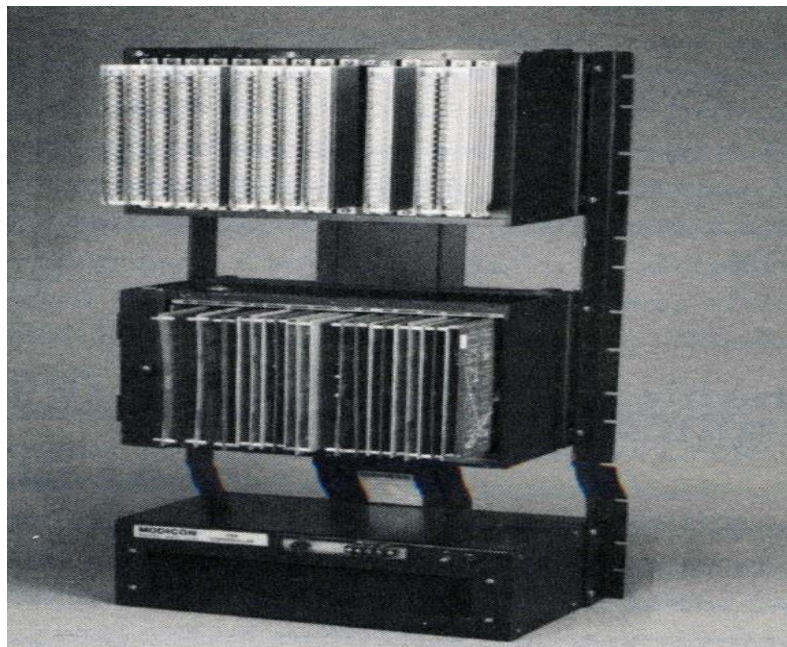


Figura 1-1. MODICON 084

Cabe añadir que la interfaz hombre-máquina se comenzó a desarrollar mediante el empleo de interruptores y botones rebobinadores mientras que las pantallas con electrónica táctil han quedado ligadas al físico George Samuel Hurst con su invención de este tipo de interfaz en 1971.

1.2. Motivación

Actualmente la inclusión de autómatas programables o PLCs en la industria se hace indispensable para cualquier proceso automatizado debido a que estos dispositivos electrónicos permiten controlar todo tipo de máquinas y procesos industriales. En comparación con los antiguos sistemas basados en relés e interruptores, estos aparatos suponen un gran ahorro en mantenimiento debido al manejo de un gran número de entradas y salidas además de un aumento de la vida útil de las máquinas industriales asociadas.



Figura 1-2. Ejemplo de célula automatizada

El PLC tiene un gran ámbito de aplicación en la industria moderna que incluye aplicaciones interesantes en campos como, por ejemplo, las maniobras de maquinaria y de instalaciones (en este caso se abordará el control de la temperatura de un motor), la industria del automóvil o la fabricación de neumáticos.

Por otro lado, la programación de una pantalla HMI facilita la interacción de cualquier operador con la máquina de manera que pueda observar la evolución de cualquier sistema de una forma muy visual y actuar directamente sobre las salidas del PLC de una forma sencilla.

En este proyecto, cabe destacar la posibilidad de haber realizado una guía del manejo de la pantalla HMI incluida en uno de los anexos como ayuda para una futura aplicación por parte del alumnado con los dispositivos que he utilizado para la realización de este trabajo.

1.3. Descripción de los objetivos

El objetivo fundamental de este trabajo consiste en la implementación de un control PID de la temperatura del Simulador de Control de Procesos Feedback PT326 a través de las señales de entrada y salida recibidas por el PLC SIEMENS SIMATIC S7-1200. El autómatas estará conectado a una pantalla SIEMENS SIMATIC HMI que servirán como interfaz hombre-máquina.

A continuación, se desglosan ese objetivo final en cada uno de los objetivos intermedios alcanzados para poder avanzar en el trabajo:

- Estudio previo del motor para obtener un modelo de respuesta del sistema de manera que se pueda crear un controlador adecuado.
- Configuración del PLC incluyendo la dirección IP y la máscara de subred adecuadamente observando las entradas y salidas disponibles.
- Aprendizaje del manejo de lenguaje KOP (lenguaje de contactos) incluyendo los bloques funcionales utilizados y el lenguaje SCL (lenguaje de alto nivel).

- Aprendizaje del manejo del bloque PID utilizando, configurando de manera óptima cada una de las entradas, salidas y parámetros internos que incluye el bloque.
- Manejo de las señales de entrada y salida del PLC en el sistema. Para ver la evolución de estas es recomendable crear una tabla de observación y forzado.
- Programación de la visualización de la pantalla incluyendo elementos interactivos e imágenes mostradas por la pantalla.
- Conexión de la pantalla HMI con el PLC vía Ethernet Industrial (PROFINET).
- Implementación de varios modos de control: Automático, Manual y Optimización (posteriormente se explicará en qué consiste cada uno de los modos).
- Manejo del servidor web y toma de resultados obtenidos.

Una vez que el sistema pueda ser controlado en cualquiera de los modos establecidos se añadirá una visualización de la evolución de la temperatura, la referencia y la actuación de manera que sea posible ver la evolución del sistema en cada instante.

2 DESCRIPCIÓN DE LOS COMPONENTES

En este capítulo se tratará de detallar las características de cada uno de los componentes utilizados incluyendo una explicación sobre la función que tienen en este proyecto.

2.1. Pantalla HMI

Se utilizará una pantalla de SIMATIC HMI de Siemens que servirá como interfaz máquina-hombre.

El modelo utilizado ha sido KTP700 Basic PN con referencia 6AV2123-2GB03-0AX0, cuya pantalla posee las siguientes características: 7" TFT, 800 x 480 pixeles con 65536 colores. Esta pantalla digital es de uso táctil e incluye 8 botones configurables en la parte inferior.



Figura 2-1. Parte delantera Pantalla Siemens

La alimentación se recibe por una entrada USB y la conexión con el PLC se realizará a través de un cable PROFINET. La entrada USB permite conectar un dispositivo de almacenamiento externo de hasta 16 GB que podrá usarse como memoria, pero no se usará en ningún caso para transferir el proyecto. Para conectar el resto de elementos es necesario saber que la pantalla debe alimentarse a DC 24V.



Figura 2-2. Parte trasera pantalla Siemens

Por otra parte, la entrada Ethernet conectará con un Switch de XB005 (5 puertos) como el que se observa en la figura 2-3 para la transmisión de datos al PLC y al PC como se explicará en el apartado de conexionado.



Figura 2-3. Visión frontal Switch Siemens

2.1 PLC

El PLC será el encargado de gestionar toda la información. En primer lugar, recibirá la evolución de la temperatura a través de la tensión que reciba por su entrada analógica. Además, recibirá las señales de la pantalla que actuarán directamente sobre sus variables internas. Finalmente será el encargado de generar las salidas hacia el simulador de procesos para controlar su temperatura.

El modelo PLC elegido será de la línea SIMATIC S7-1200. En concreto el CPU 1214C AC/DC/Rly, con referencia 6ES7214-1BG40-0XB0. El PLC incluye 14 entradas digitales y 2 entradas analógicas (en el proyecto se usará una de ellas) además de 10 salidas digitales además de una analógica que se utilizará para enviar tensión al simulador de procesos.



Figura 2-4. Entradas PLC Siemens



Figura 2-5. Salidas PLC Siemens

Al igual que la pantalla, el PLC irá conectado al PC a través de conexión Ethernet por medio del Switch. En cambio, a diferencia de la pantalla, la alimentación del PLC será directamente a través de la red.

2.2 Sistema de control de temperatura

El simulador de procesos *PT326 Process Trainer* es un equipo de medida y control que simula el comportamiento de un Sistema con un retraso. El equipo está formado por una rejilla de alimentación a través de la que entra aire a temperatura ambiente. El aire se calienta usando una resistencia eléctrica y la temperatura del aire se mide en la salida del tubo mediante un sensor de temperatura.

Este equipo se alimentará directamente de la tensión de red a 230V e incorpora 2 fusibles de protección de 2A y 100mA. Para la medida de la temperatura el sistema incluye un termistor esférico al final del tubo que sirve como elemento detector.

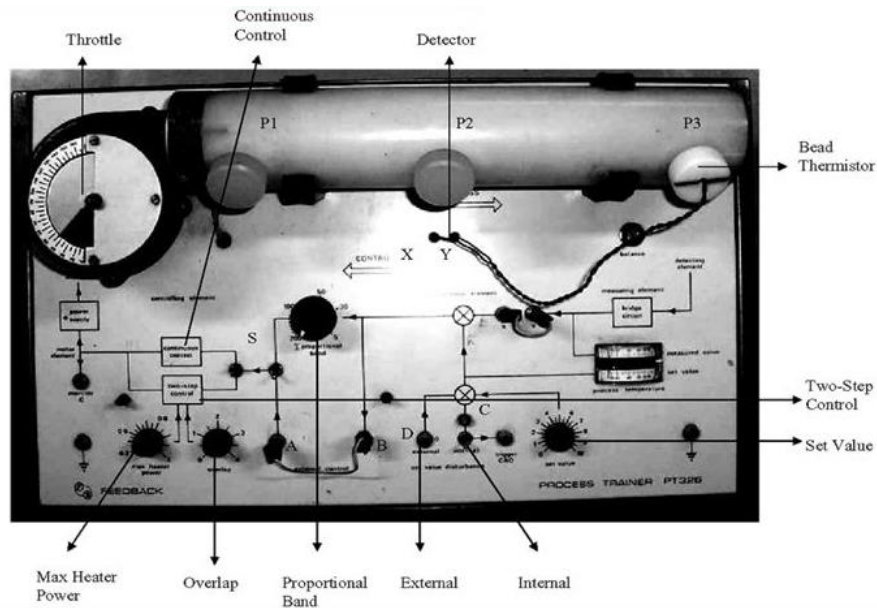


Figura 2-6. Visión frontal PT326

Existen varios métodos de estabilización del sistema entorno a una temperatura:

- Control por relé: Control todo o nada que se realizará estableciendo una temperatura deseada.
- Control continuo interno: Solo permitirá el ajuste de una ganancia proporcional expresada en porcentaje.
- Control continuo externo: Permite un método de control PI, PD o PID, pero dependerá de una acción de control externa.

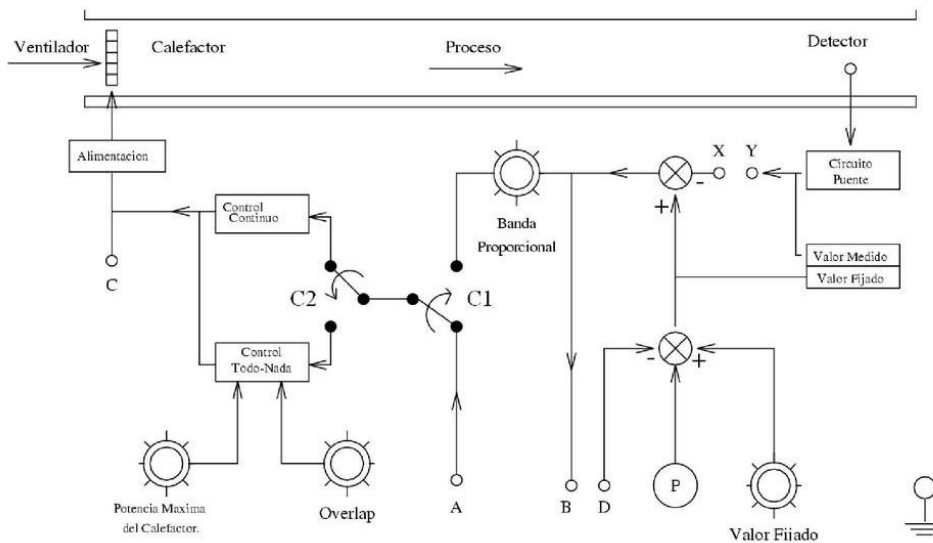


Figura 2-7. Esquema de control PT326

Siguiendo la figura, se debe clarificar que el conmutador C1 sirve para elegir entre Control interno o externo mientras que C2 hará lo propio entre Control continuo o Control por relé. Cabe destacar que la entrada D servirá

para ajustar el valor medido teniendo como referencia el valor deseado. La diferencia entre ambos valores se observa a través de la salida B.

En este trabajo se introducirán distintos valores de tensión por la entrada A y se obtendrán los valores de tensión proporcionales a la temperatura a través de Y. Estos valores irán a parar a las salidas y entradas analógicas del PLC respectivamente.

2.3 Elementos de conexionado

En primer lugar, se crea una estructura de soporte donde se posicionarán los elementos fundamentales de control del Sistema: Fuente de alimentación, PLC, Switch y pantalla HMI como se observa en la figura 2-8.

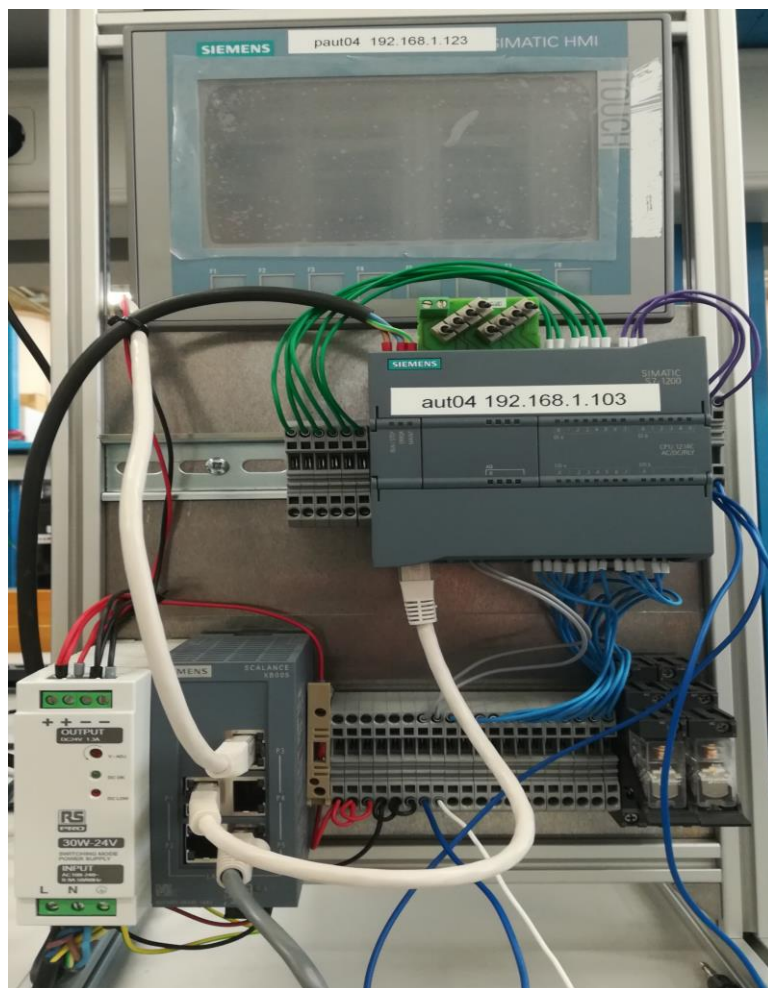


Figura 2-8. Conexionado de la estructura

En la esquina inferior izquierda se encuentra la fuente de alimentación de 24 V que alimentará el Switch y la pantalla HMI. Justo a su lado, se encuentra el Switch que se encuentra conectado al router, lo que permite la conexión del ordenador con los distintos elementos. Además, el Switch irá conectado a la pantalla y al PLC a través de cables de conexión Ethernet. Cabe destacar que el PLC y la fuente irán conectados a la tensión de red del laboratorio.

Por otro lado, el PLC irá conectado con el simulador de procesos a través de sus entradas y salidas analógicas mediante cables de conexión banana-macho.

Para clarificar el conexionado que se ha explicado se adjunta un esquema en la figura 2-9:

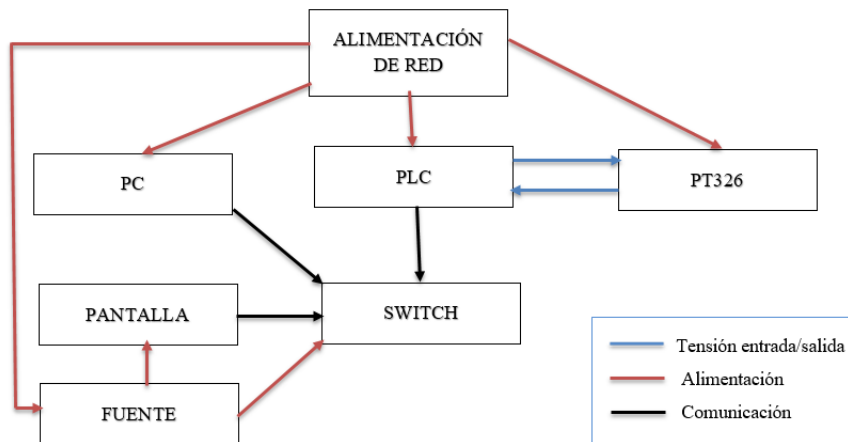


Figura 2-9. Esquema de conexionado

3 CONFIGURACIÓN DE LOS DISPOSITIVOS

En este capítulo se mostrará cómo crear y configurar un proyecto en el programa TIA Portal además de programar adecuadamente el PLC. Finalmente se explicará cómo se deben cargar todas las funciones y bloques programados en los dispositivos correspondientes.

3.1 Creación y configuración de un proyecto

En primer lugar, se debe lanzar el programa y solicitar la creación de un nuevo proyecto con su nombre correspondiente. En este momento aparece una pantalla auxiliar a la derecha del monitor.

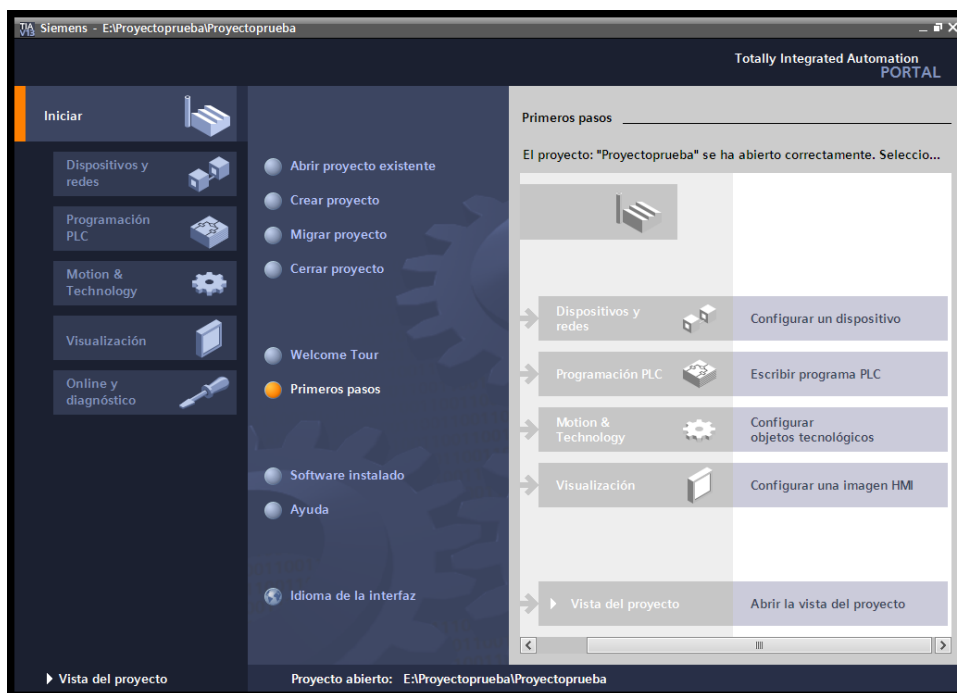


Figura 3-1. Primeros pasos de la creación de un proyecto

A partir de la pantalla de primeros pasos se procede a configurar los dispositivos que sean necesarios. Para ello se selecciona la opción 'Dispositivos y redes' y posteriormente se debe hacer click en 'Agregar dispositivo'.

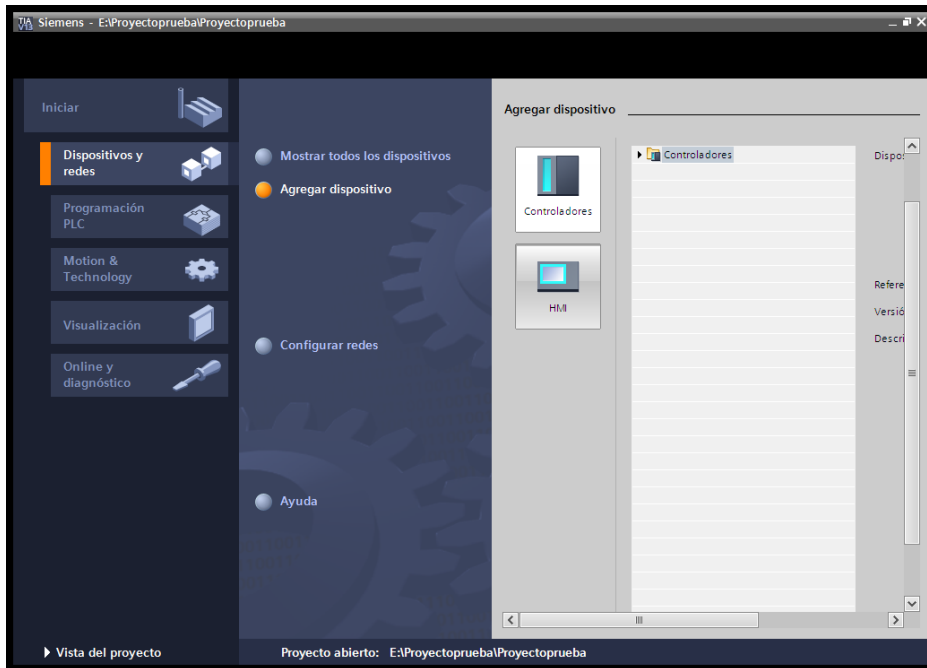


Figura 3-2. Agregar un dispositivo

En esta pantalla se abre la posibilidad de elegir el tipo de dispositivo que se debe agregar además del modelo concreto.

En primer lugar, se configura un controlador de la línea de Siemens SIMATIC S7-1200. No se recomienda especificar el modelo para poder elegir el adecuado a través de la opción de detección de Hardware que ofrece el propio programa.

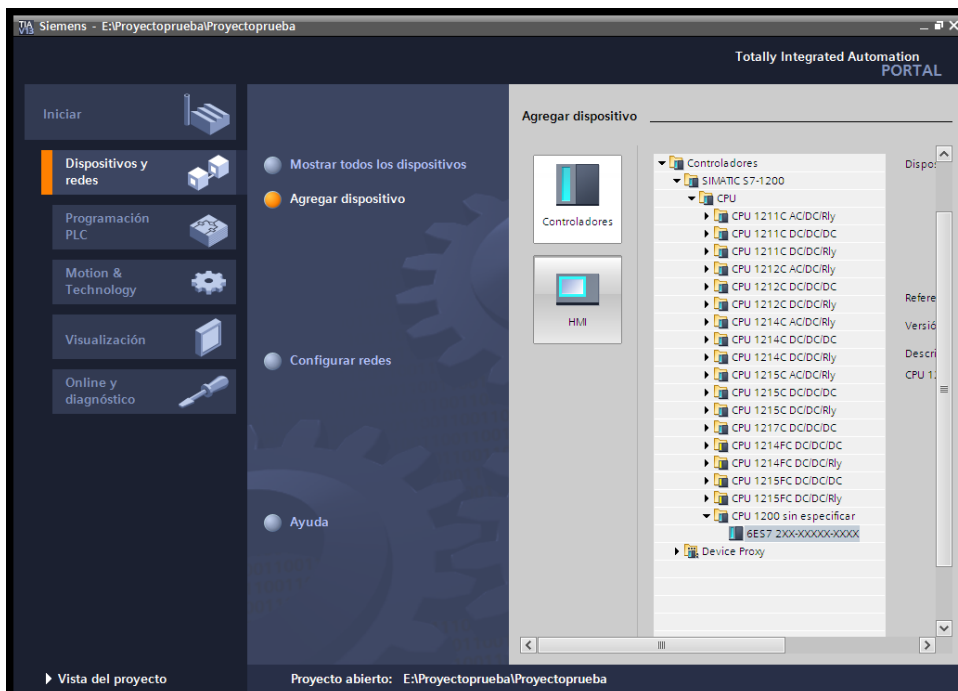


Figura 3-3. Agregar un controlador sin modelo especificado

3.2 Configuración del PLC

Cuando se agrega el controlador se abre por primera vez la vista del proyecto, lo que da la posibilidad de configurar el dispositivo y empezar a programar el controlador.

En primer lugar, se debe configurar la detección del modelo concreto del hardware. Para ello se hace click en 'Catálogo de hardware' dentro de la ventana auxiliar que se observa en la figura 3-4 para especificar la CPU:

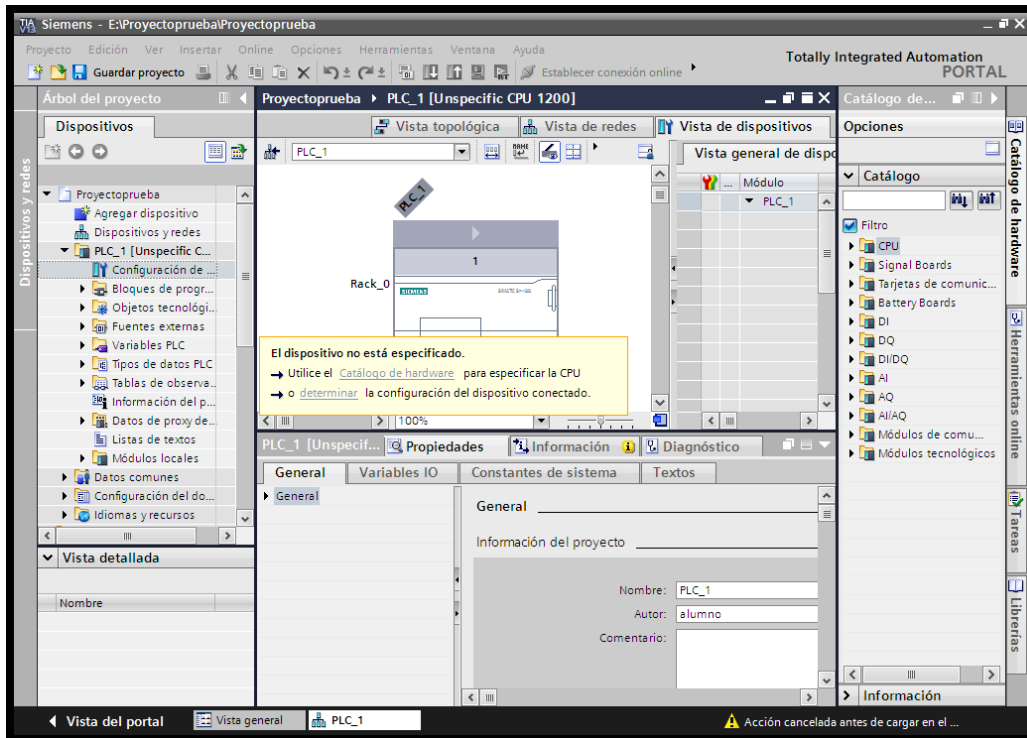


Figura 3-4. Detección de hardware para el PLC

Una vez que aparece una pantalla emergente para la detección, se debe aclarar que el tipo de interfaz PG/PC es PN/IE mientras que la interfaz será Realtek PCIe GBE Family Controller. Tras iniciar la búsqueda se obtendrán distintos dispositivos activos. Se debe buscar un PLC cuya dirección IP coincida con la del dispositivo que se quiere conectar realmente, en nuestro caso 192.168.0.103. Finalmente se finaliza la operación haciendo click sobre el botón de detección.

En el caso de que se desee cambiar la IP, se debe acceder a la ventana de configuración del dispositivo y pulsar sobre el símbolo de la entrada de interfaz Profinet. Una vez que se obtenga un panel en la parte inferior de la pantalla se accede a la pestaña de Direcciones Ethernet como se observa en la figura 3.5.

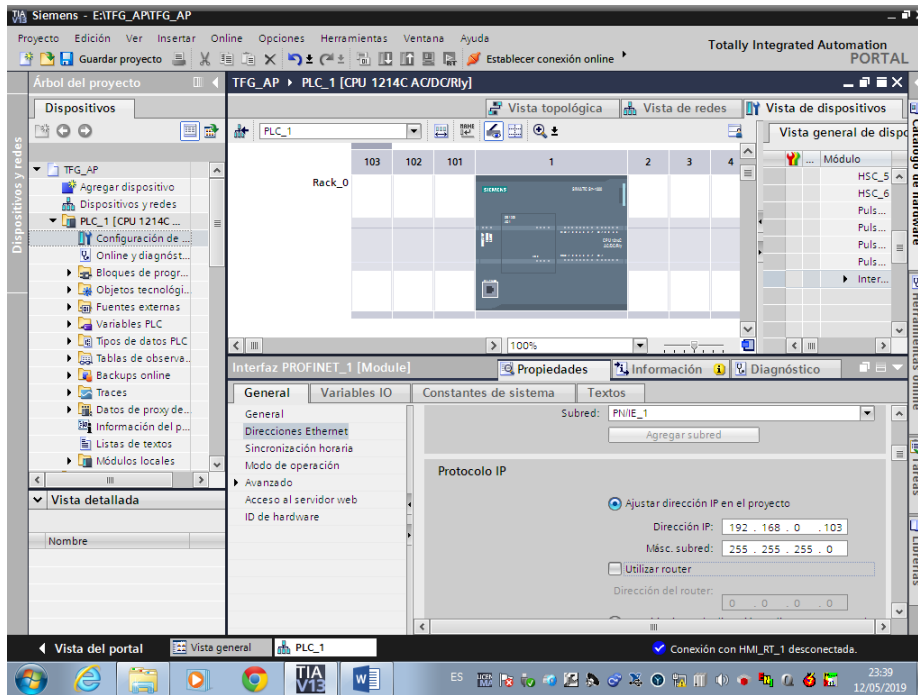


Figura 3-5. Configuración IP del PLC

La dirección y la máscara de subred tienen que ser disponibles con las correspondientes del PC. Cabe destacar que cualquier cambio que se produzca en esta ventana no será efectivo hasta que no se cargue el proyecto en el dispositivo de nuevo.

Para agregar cualquier otro dispositivo, como el caso de la pantalla, se puede hacer volviendo a la vista del portal de TIA Portal o a través de la opción de Agregar dispositivo que aparece en la vista del proyecto. La configuración se realizará de forma análoga.

3.3 Programación de un PLC

A continuación, se tratará de explicar el lenguaje utilizado para programar el controlador. Para ello se mostrarán los tipos de bloques que se pueden utilizar en TIA Portal y como han sido utilizados en el proyecto para poder interactuar con la pantalla y con el sistema a controlar.

3.3.1 Bloques de programa

Existen cuatro tipos fundamentales de bloques utilizados en el programa TIA Portal según el objetivo de programación. Se encuentran en el árbol del Proyecto, dentro de la pestaña 'Bloques de programa'. Los cuatro tipos se observan en la figura 3-6.

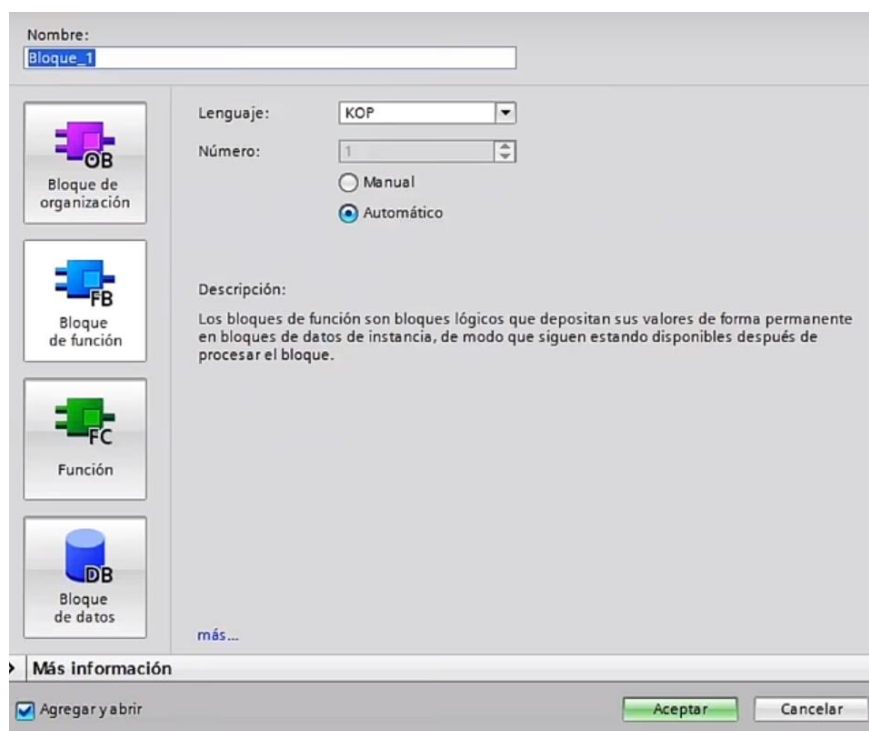


Figura 3-6. Bloques de programa

3.3.1.1 Bloques de organización (OB)

Constituyen la interfaz entre el sistema operativo y el programa del usuario, es decir, la implementación de las salidas estará directamente enlazada con la lógica implementada en este tipo de bloques. Esto implica que los otros bloques de programación (bloques de función y funciones) deben ser invocados desde un bloque de organización para ser efectivos.

Estos bloques tendrán una ejecución cíclica que variará dependiendo del tipo de OB elegido. Los más utilizados son los siguientes:

- Program Cycle: Bloques de organización que se procesan cíclicamente y se utilizan para programar instrucciones o para llamar a otros bloques.
- Startup: Bloques de arranque que se ejecutan una sola vez cuando la CPU pasa de modo STOP a modo RUN.
- Cyclic Interrupt: Bloques de alarma cíclica que interrumpen el procesamiento cíclico para ejecutarse en intervalos definidos de tiempo. Son muy útiles para ejecutar un algoritmo que requiere un tiempo considerable de cálculo.

Es conveniente que todos los bloques de organización del proyecto con una finalidad parecida tengan el mismo tipo para procesar correctamente y ejecutar las instrucciones en un orden lógico.

3.3.1.2 Funciones (FC)

Son bloques lógicos sin memoria que no son capaces de almacenar valores. Al no tener memoria global, sus variables internas no existen fuera de la función, por lo que es preciso asignar variables globales a las variables internas de la función como entradas, salidas o entradas/salidas.

Según el manejo que se desee tener sobre las variables, cada función incorpora un conjunto de entradas (solo se modificará su valor desde fuera de la función), salidas (solo se modificará su valor por programa) y entradas/salidas (podrá modificar su valor desde fuera de la función y mediante la lógica del programa).

Cabe destacar que dentro de la función se pueden incorporar ciertos bloques que sí pueden tener memoria interna como el caso del temporizador (el más utilizado en las funciones). La memoria de estos bloques no será efectiva fuera de la función (por ejemplo, el temporizador en cuestión nunca se arrancará directamente a través de una condición lógica de un OB ni la variable de salida del temporizador podrá ser utilizada en cualquier otro bloque que no sea esa función).

Las funciones son muy útiles cuando se requiere una ejecución secuencial dentro de un bloque de organización. En este caso se invocará desde un OB a un conjunto de funciones que se ejecutarán en el orden establecido en cada ciclo.

3.3.1.3 Bloques de función (FB)

Son bloques lógicos con memoria por lo que siguen estando disponibles después de editarlos. Estos bloques depositan sus parámetros de forma permanente en bloques de datos de instancia.

Los bloques de datos de instancia son bloques que guardan un conjunto de variables para cada bloque de función. Estos bloques conforman la memoria propia de cada bloque, por lo que tendrán una estructura asociada a cada bloque en concreto y no podrán utilizar ninguna de sus variables en cualquier otro bloque lógico.

Al igual que las funciones, sus parámetros serán entradas, salidas, entradas/salidas y cualquier otro bloque que se quiera incorporar. En este caso los bloques utilizados podrán ser multi-instancia, es decir, podrán ser utilizados en varios bloques de función a la vez.

Son útiles para programas que requieren la ejecución de una misma lógica para entradas y salidas distintas, debido a que permiten programar la lógica una sola vez y ejecutarla para cada bloque de datos de instancia según corresponda.

3.3.1.4 Bloques de datos globales (DB)

Son bloques que se utilizan para almacenar valores que se escriben durante la ejecución del programa. A diferencia de los tres bloques anteriores, no contienen instrucciones, sino que solo manejan declaraciones de variables.

No se le podrá asignar ninguna dirección a una variable directamente en un bloque de datos por lo que éstos no estarán asignados a ningún bloque lógico a diferencia de los bloques de instancia individual. En cambio, se puede acceder a las variables desde cualquier bloque lógico.

Se deben utilizar para evitar crear instancias individuales, las cuales ocupan memoria extra en cada uno de los bloques, pero se debe tener en cuenta que se estarán creando variables estáticas en todo momento.

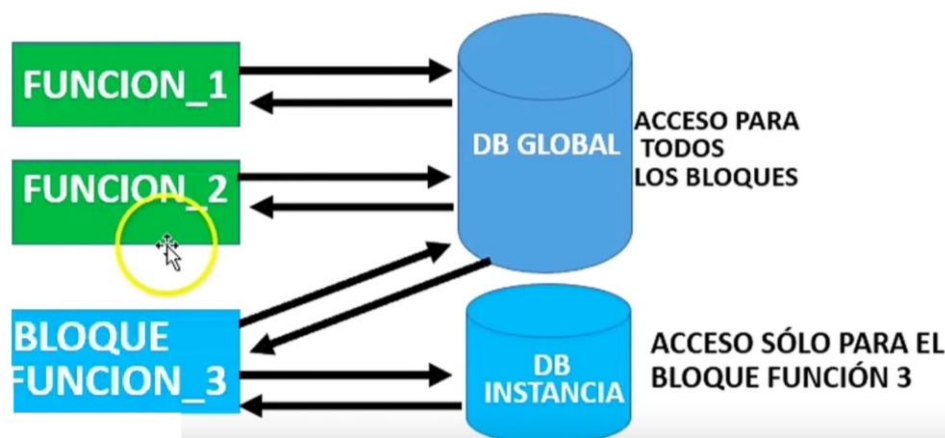


Figura 3-7. Acceso a bloques de datos





3.3.2 Lenguajes de programación

En cualquiera de los tres bloques de programación que se han utilizado (Bloque de organización, bloque de función y función) aparece la misma interfaz y se puede aplicar cualquiera de los lenguajes que ofrece TIA Portal.

El lenguaje más utilizado será el KOP, que es el lenguaje de contactos basado en la implementación de interruptores y bobinas. Por otro lado, se pueden usar funciones y bloques de funciones que se implementen en el MAIN como bloques lógicos. Este tipo de lenguaje se llama FUP. Finalmente se puede implementar el lenguaje de alto nivel de Siemens llamado SCL.

A continuación, adjunto las opciones que ofrece la tabla de herramientas del programa:

Tabla 3-1. Barra de herramientas de programación

Simbolo	Comentario
	Para introducir un segmento nuevo en el programa
	Permite introducir cualquier bloque auxiliar
-(-)-	Bobina apagada normalmente
-(S)-	Bobina de activación de una variable
-(R)-	Bobina de desactivación de una variable
- -	Interruptor abierto normalmente
- / -	Interruptor cerrado normalmente
	Crea ramas dentro del mismo segmento que se encuentran en paralelo
	Cierra las ramas abiertas que se encuentran en paralelo

3.4 Visualización de las variables

Una vez que el programa ha sido cargado correctamente en el dispositivo y el PLC está en modo RUN, el programa se estará ejecutando correctamente.

Para observar la lógica programada de manera instantánea se debe pulsar el botón de ‘activar observación’ en la tabla de herramientas de la parte superior de la imagen.

Otra opción consistirá en visualizar la evolución de las variables. Para ello se debe crear una tabla nueva (1) en la ventana de tablas de observación y forzado permanente del árbol del proyecto. En esta tabla se deberán añadir todas las variables que se deseen evaluar simultáneamente.

Una vez creada la tabla, el segundo paso será activar la herramienta de observar todo (2). A partir de entonces se puede observar el valor actual de cada variable en la columna de valor de observación.

Finalmente, si se quiere forzar una variable se debe escribir un valor en la columna de ‘valor de forzado’, y a continuación, pulsar la herramienta de ‘Forzar inmediatamente variables’ (3) para hacerlo efectivo.

Árbol del proyecto ...y| Tabla 3 observ 2 y forzado permanente Tabla de observación_1

Dispositivos

1

3

2

	Nombre	Dirección	Formato visualiza..	Valor de observac..	Valor de forzado
1	*ACTUACION_...	%MD12	Número en c...	0.5	
2	*ACTUACION_V	%MD8	Número en coma...	0.0	
3	*EA1_Sensor	%IW64	DEC+/-	0	
4	*SA1_Actuacion*	%QW80	DEC+/-	0	
5	*SENSOR_ESCALA...	%MD0	Número en coma...	0.5	
6	*VALOR_MEDIDO*	%MD4	Número en coma...	0.0	
7	*SETPOINT	%MD16	Número en coma...	0.0	1.0
8		<Agregar>			

Figura 3-8. Tabla de observación y forzado permanente

4 PROGRAMACIÓN FINAL DEL PLC

Tras la introducción sobre como programar en TIA Portal, se mostrará cual ha sido el resultado del proyecto, explicando el funcionamiento de las variables y mostrando los bloques de programa de usuario que se han utilizado.

4.1 Variables del PLC

4.1.1 Entradas y salidas

En este proyecto se utilizará una única entrada analógica y una única salida:

La única entrada (Tipo INT) se nombrará como 'Sensor_Medida' y corresponderá con el valor en tensión que recibe por parte del sensor de temperatura del simulador de procesos. Se dirección está configurada como %IW64

Por su parte la salida será la variable 'SA1_Actuacion' que saldrá desde el canal 0 de salida hasta la entrada de tensión del simulador de procesos que influirá directamente sobre la temperatura de salida del tubo. Será direccionada como %QW80.

4.1.2 Marcas del PLC

Son variables internas del PLC. A continuación, se adjunta el conjunto de marcas utilizadas en la Tabla 4-1:

Tabla 4-1. Variables del PLC

Nombre	Tipo	Dirección	Comentario
Clock_5Hz	BOOL	%M0.1	Variable interna del sistema que se activa cada 0.2s
Medicion_tension	REAL	%MD4	Tensión de la entrada en voltios
Referencia	REAL	%MD8	Valor de la referencia en tensión
Salida_escalado	REAL	%MD12	Variable intermedia para el escalado de la salida
Actuacion_real	REAL	%MD16	Tensión de salida instantánea en voltios
ModoManual	BOOL	%M20.0	Activa cuando la salida se establece manualmente
Ganancia	REAL	%MD24	Parámetro Kp del controlador
Tiempo_integral	REAL	%MD28	Parámetro Ti del controlador
Tiempo_derivativo	REAL	%MD32	Parámetro Td del controlador
SalidaManual	REAL	%MD36	Valor de salida en modo manual

Modo_operacion	INT	%MW40	Modo de operación del control PID
Activacion_modos	BOOL	%M44.0	Copia el modo en el estado cuando se produce un flanco de subida
Estado_operacion	INT	%MW48	Estado actual del control PID
Reset	BOOL	%M52.0	Pasa a modo inactivo de operación
Error	BOOL	%M56.0	Se activa si hay algún error en el control
Temperatura_escalado	REAL	%MD60	Variable auxiliar para el escalado de la temperatura
Medicion_temperatura	REAL	%MD64	Valor de entrada escalado en °C
Referencia_Temperatura	REAL	%MD68	Valor de referencia en °C
Referencia_Escalado	REAL	%MD72	Variable auxiliar para el escalado de la referencia
Pto_funcionamiento	REAL	%MD76	Punto de funcionamiento del control
Cargar_Parametros	BOOL	%M80.0	Carga el juego de parámetros actual en el controlador
RetardoDerivativo	REAL	%MD84	Coefficiente de retardo derivativo del algoritmo
Ponderacion_P	REAL	%MD88	Coefficiente de ponderación de la ganancia
Ponderacion_D	REAL	%MD92	Coefficiente de ponderación derivativo
TiempoMuestreo	REAL	%MD96	Tiempo de muestreo del algoritmo del PID
LI_setpoint_activo	BOOL	%M100.0	Se activa cuando el valor de la referencia supera el límite superior
LS_setpoint_activo	BOOL	%M104.0	Se activa cuando el valor de la referencia de supera el límite inferior
Valor_LS_setpoint	REAL	%MD108	Valor para el límite superior de la referencia
Valor_LI_setpoint	REAL	%MD112	Valor para el límite inferior de la referencia
Aviso_LS	BOOL	%M116.0	Se activa cuando el valor de la entrada supera el límite superior de aviso
Aviso_LI	BOOL	%M120.0	Se activa cuando el valor de la entrada supera el límite inferior de aviso
ValorAvisoLS	REAL	%MD124	Límite superior de aviso para la entrada
ValorAvisoLI	REAL	%MD128	Límite inferior de aviso para la entrada
Progreso_optimizacion	REAL	%MD132	Avance de la optimización en %
Estado_optimizacion_inicial	INT	%MW136	Estado actual de la optimización inicial

Estado_optimizacion_fina	INT	%MW140	Estado actual de la optimización fina
ArchivoCreado	BOOL	%M144.0	Se activa mientras un archivo está creado
EscribirArchivo	BOOL	%M48.0	Se activa para solicitar que se abra un archivo
Aviso	DWORD	%MD152	Indica que tipo de aviso existe en formato hexadecimal
MensajeError	DWORD	%MD156	Indica qué tipo de error existe en formato hexadecimal
Sensor_escalado	REAL	%MD160	Variable intermedia para escalado de la entrada
Ganancia_real	REAL	%MD164	Valor real del parámetro Kp en cada instante
Ti_real	REAL	%MD168	Valor real del parámetro Ti en cada instante
Td_real	REAL	%MD172	Valor real del parámetro Td en cada instante

4.2 Bloques de programa de usuario

En el proyecto se utilizarán solo bloques de organización (OB) y bloques de datos. No se utilizan bloques de función ni funciones debido a que todos los bloques son bastante independientes (no comparten variables) y no hay ninguno que necesite ninguna función interna que llame a otra para ejecutarse.

En todos los casos que se deba utilizar un bloque de organización (menos en el de Startup), se utilizará un OB de tipo “Cyclic Interrupt” que se ejecuta cada 0.1 segundos (al igual que el tiempo de muestreo del algoritmo).

4.2.1 Startup

En primer lugar, se explicará el bloque de organización Startup que se utilizará para iniciar ciertas variables en un valor concreto para evitar ningún mensaje de error al iniciar el PLC:

Se establece una referencia en temperatura de 21 °C lo que equivale a una referencia en tensión de 0.1V. Por su parte los valores límites de la referencia estarán establecidos entre 0 y 8V. De esta manera se evita que se produzca ningún error al arrancar el PLC.

Se establecen los límites de aviso para los valores de temperatura (en tensión) recibidos por la entrada analógica entre 0 y 8 voltios debido a que entre 8 y 10V el modelo de la temperatura en °C respecto a la tensión pasa a ser no lineal.

Finalmente, se establece un tiempo de muestreo del algoritmo del PID cada 0.1 segundos y una ponderación completa (=1) de cada uno de los coeficientes de ponderación de ganancia y derivativo.

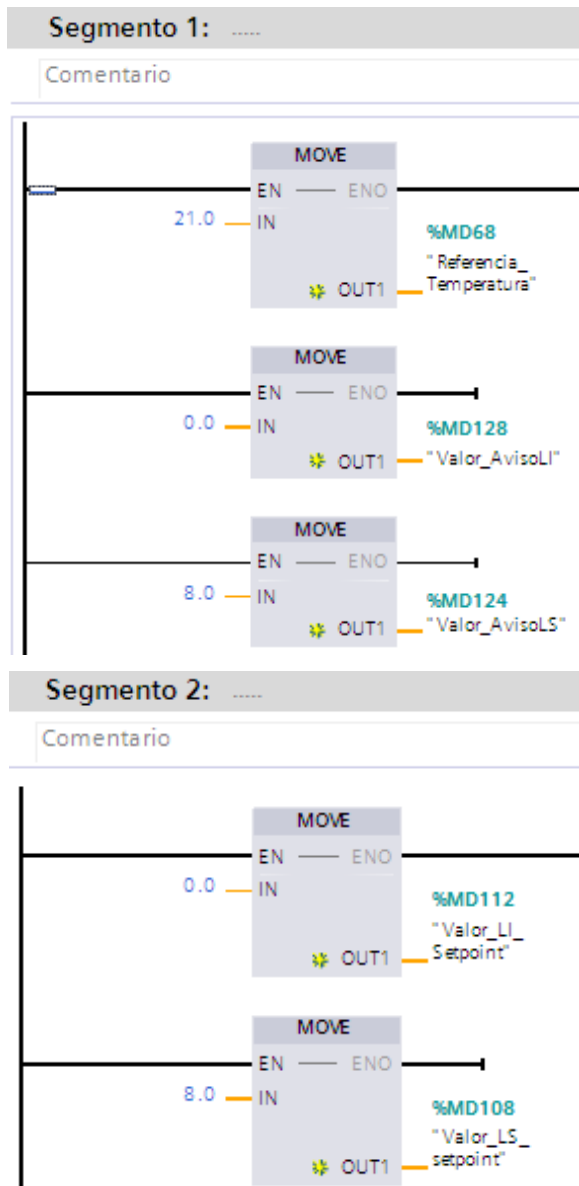


Figura 4-1. Segmentos 1 y 2 del bloque Startup

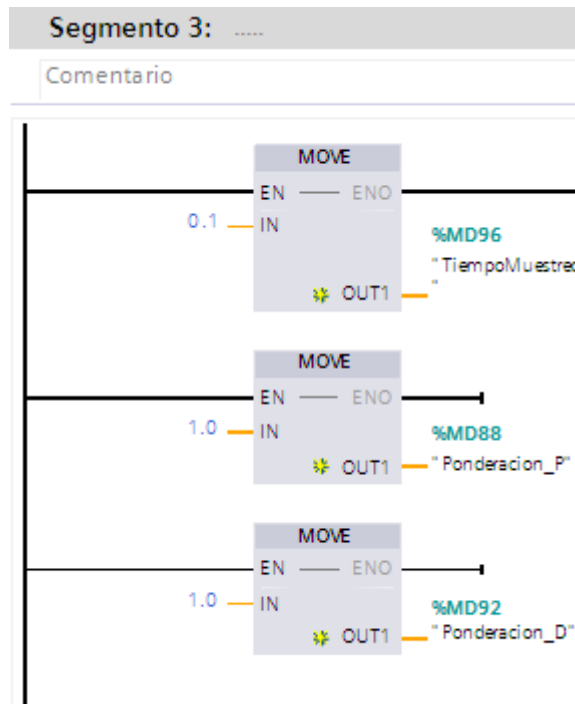


Figura 4-2. Segmento 3 del bloque Startup

4.2.2 Bloque_PID

En este OB se implementa un bloque PID_Compact V2 en el que se asocian los parámetros internos que interesan desde el propio bloque a variables globales del proyecto que después se utilizarán en la pantalla:

○ Entradas:

- Referencia: Valor de la consigna (en tensión)
- Sensor_Medida: Entrada analógica directamente a la dirección %IW64
- Pto_funcionamiento: Valor en tensión modificable que se le introduce como valor de entrada directamente al sistema antes de empezar a controlar. Normalmente se usa para corregir una perturbación.
- ModoManual: Habilitación de la salida manual
- SalidaManual: Valor real de la salida manual.
- Reset: Entrada booleana que activa el reset
- Activacion_modo: Activación del modo de operación por flanco de subida
- Modo_operacion: Modo propio de operación que se quiere incorporar al bloque PID.

○ Salidas:

- SA1_Actuacion: Salida analógica en tensión que actuará directamente sobre el sistema.
- LS_Setpoint_activo/LI_Setpoint_activo: Variables booleanas de señalización que se activarán en caso de que los valores de la referencia hayan salido de los límites.
- Aviso_LS/Aviso_LI: Variables booleanas de señalización que se activarán en caso de que los valores de la entrada analógica hayan salido de los límites.

- Estado_operacion: Estado de operación actual del bloque.
- Error: Variable de señalización de error.

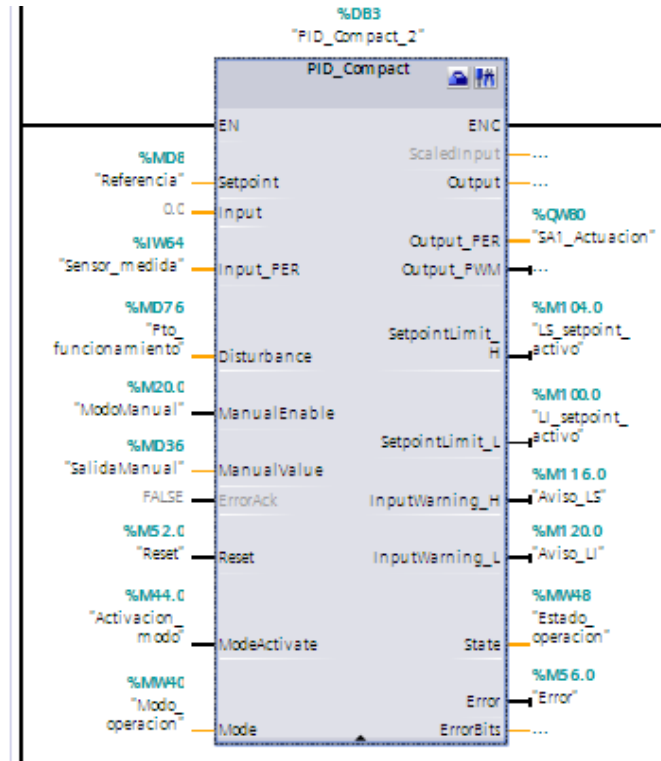


Figura 4-3. Bloque PID_Compact

4.2.2.1 Función de transferencia

Además de los parámetros explicados, el bloque PID contiene una serie de parámetros que se observan en la función de transferencia de la figura 4-4 que se explicarán a continuación:

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

Símbolo	Descripción
y	Valor de salida
K _p	Ganancia proporcional
s	Operador laplaciano
b	Ponderación de la acción P
w	Consigna
x	Valor real
T _i	Tiempo de integración
a	Coefficiente para el retardo de la acción derivada (T1 = a x T _d)
	Tiempo derivativo
c	Ponderación de la acción D

Figura 4-4. Función de transferencia

Esta función está basada en un control PID con Anti-Windup con parámetros de ponderación de las acciones proporcional y derivativa.

Además de los parámetros y , K_p , w , x , y T_i explicados anteriormente como entradas y salidas se debe aclarar la utilidad de los parámetros de ponderación y retardo:

- b: Parámetro real entre 0 y 1 que hará más efectiva la acción proporcional cuanto mayor sea.
- c: Parámetro real entre 0 y 1 que hará más efectiva la acción derivativa cuanto mayor sea.
- a: Coeficiente de retardo que compensará en retraso del sistema.

4.2.2.2 Implementación del bloque PID

A continuación, se añade un esquema donde se observe como está implementado nuestro bloque PID incluyendo las conexiones de los parámetros internos del mismo.

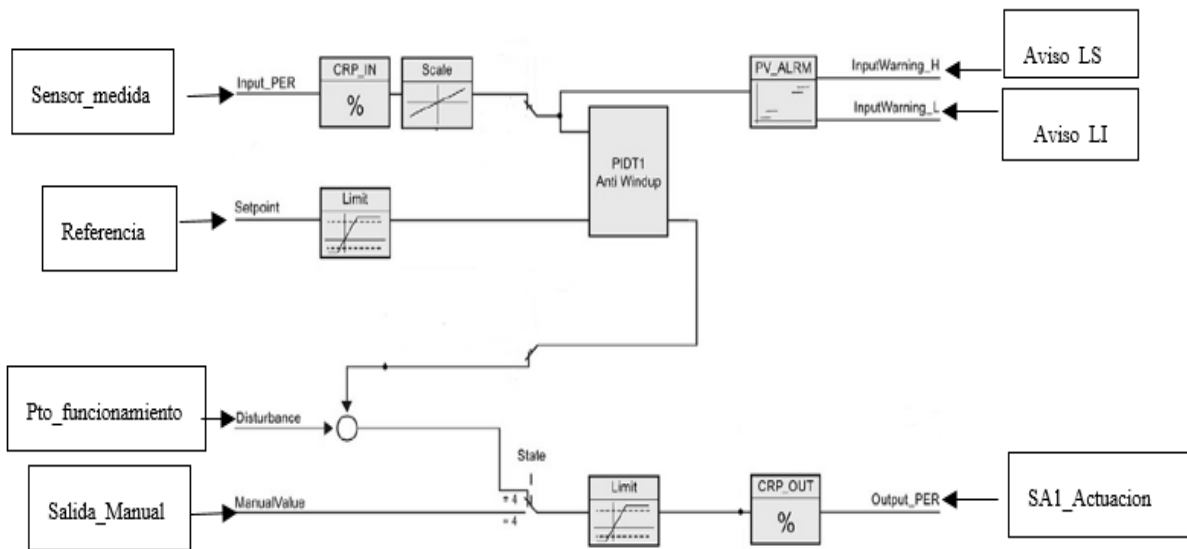


Figura 4-5. Implementación del bloque PID

4.2.2.3 Modos y Estados de operación del bloque PID

Las variables de modo y estado de operación son valores de tipo entero que oscilan entre 0 y 5 y definen la forma de actuar del bloque PID. La diferencia entre ellos es que el parámetro Estado_operacion indica el modo actual del regulador PID. El valor de "Modo_operacion" se copiará en el estado mediante flanco de subida de la variable "Activacion_modos". Los modos/estados posibles serán los siguientes:

Tabla 4-2. Modos/Estados de operación

MODO/ESTADO	Comentarios
0 Inactivo	El valor de la salida será 0 independientemente del resto de parámetros

<p style="text-align: center;">1</p> <p>Optimización inicial</p>	<p>Se somete el proceso a un testeo en busca de los parámetros óptimos</p> <p>Requisitos para comenzarla optimización inicial:</p> <ul style="list-style-type: none"> • El estado de operación debe ser previamente el 0, 3 o 4. • La habilitación de la salida manual y el reset deben estar a 0. • $Referencia-Entrada > 0.3 * Valor_avisoLS - Valor_AvisoLI$ • $Referencia-Entrada > 0.5 * Referencia$ • La referencia y el valor real de entrada deben encontrarse dentro de los límites. <p>Una vez que se haya completado la optimización el estado pasa a automático</p>
<p style="text-align: center;">2</p> <p>Optimización fina</p>	<p>Genera una oscilación constante en el sistema que sirve para calcular sus parámetros a partir de la frecuencia de la oscilación.</p> <p>Requisitos para comenzarla optimización inicial:</p> <ul style="list-style-type: none"> • El modo de operación debe ser previamente el 0, 3 o 4. • La habilitación de la salida manual y el reset deben estar a 0. • No deben existir perturbaciones. • La referencia y el valor real de entrada deben encontrarse dentro de los límites. <p>Se puede inicializar desde el modo automático para mejorar los parámetros existentes o desde el modo inactivo o manual para el que habrá que hacer una optimización inicial previa</p>
<p style="text-align: center;">3</p> <p>Automático</p>	<p>Se regula el sistema conforme a los parámetros definidos.</p> <p>Se obtiene una vez que se termina alguna de las dos optimizaciones o mediante flanco de subida de la variable de activación estando el modo a 3.</p> <p>El cambio de automático a manual se realiza sin discontinuidad.</p>
<p style="text-align: center;">4</p> <p>Manual</p>	<p>Se especifica un valor de salida en la variable “SalidaManual”. Además, para el correcto funcionamiento también debe estar activa la variable “ModoManual”.</p> <p>El paso de manual a automático se produce sin discontinuidad</p>
<p style="text-align: center;">5</p> <p>Monitorización de errores</p>	<p>No se puede activar manualmente sino a través de un error.</p> <p>Se desactiva el algoritmo de regulación y la salida pasa a ser un valor de salida sustitutivo.</p>

4.2.3 Escalado

En este bloque se tratan de escalar los valores necesarios para poder implementar las variables adecuadas en el bloque regulador. Por lo tanto, el manejo sobre las variables escaladas será indirecto:

- En primer lugar, se debe manejar el valor de referencia en temperatura que se obtiene de la pantalla. Este valor que se recibe en °C pasa a escalarse a tensión. Las variables en tensión serán manejadas por el regulador PID.
- El valor que se recibe por entrada analógica pasa a escalarse a un valor de tensión de entrada entre 0 y 10V. Por otro lado, ese mismo valor que se ha recibido queda escalado en temperatura en °C.

- El valor de salida emitido a través de la salida del PLC se escala también en tensión entre 0 y 10V.

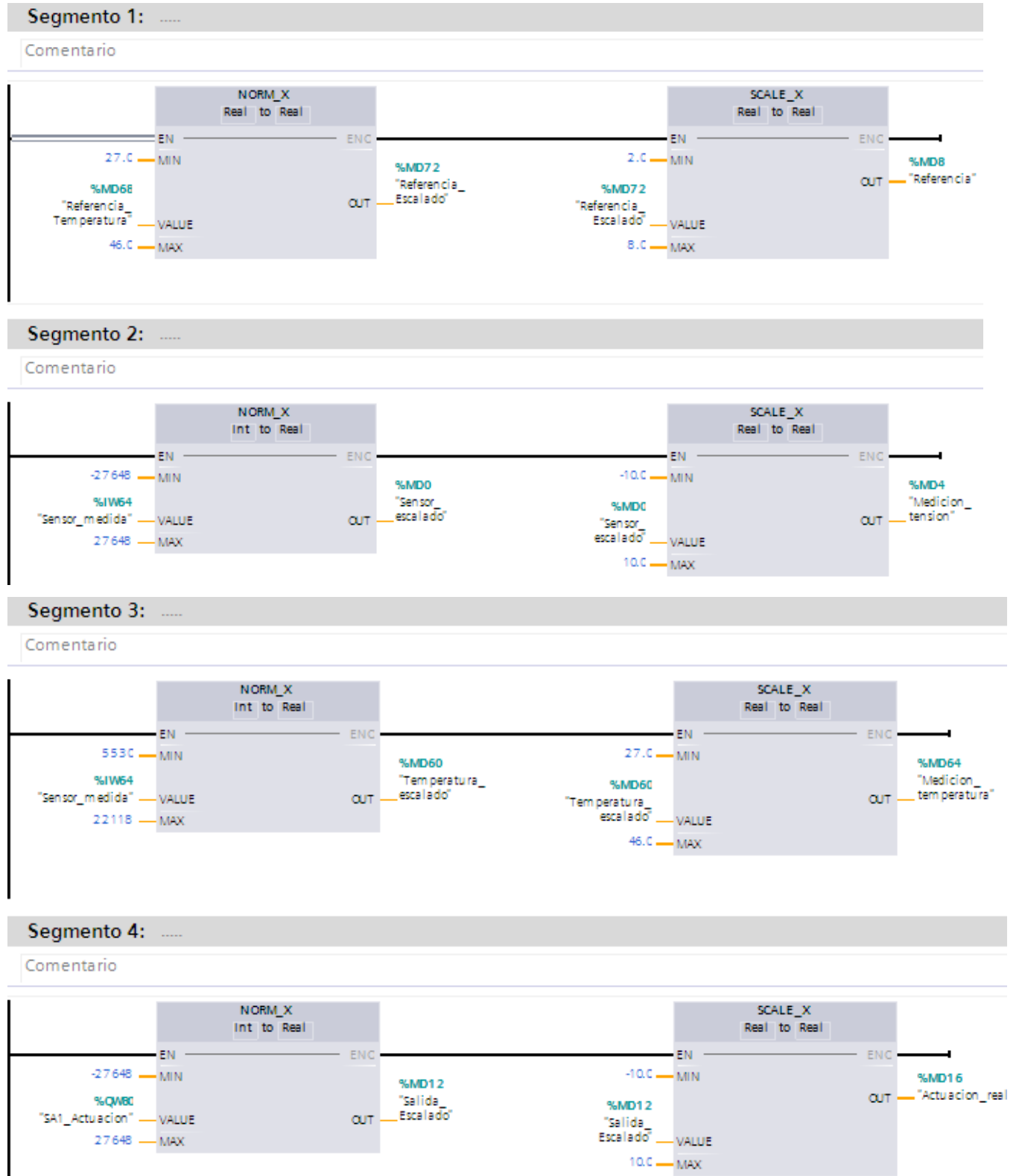


Figura 4-6. Bloque de Escalado

4.2.4 Límites

En este bloque se asignarán una serie de variables globales a parámetros internos del regulador PID mediante la instrucción MOVE: Este bloque se encargará de pasar el valor de una variable a otra del mismo tipo.

En primer lugar, se asignan los valores recibidos por la pantalla (asociadas a variables globales del PLC) a los valores de los límites de la entrada en tensión. A continuación, se hace lo propio con los valores de los límites de la referencia.

Finalmente, se realiza el proceso contrario con las variables de aviso y de mensaje de error. Estas variables serán de tipo DWORD debido a que muestran un mensaje en formato hexadecimal.

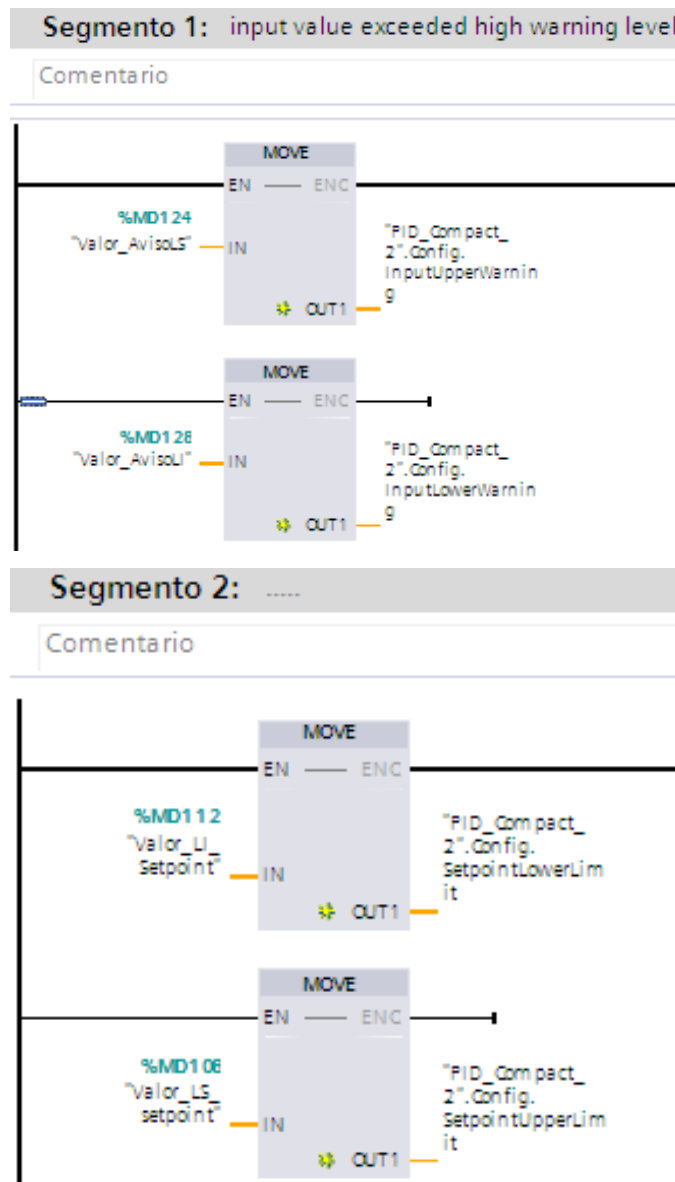


Figura 4-7. Segmentos 1 y 2 del bloque de Límites

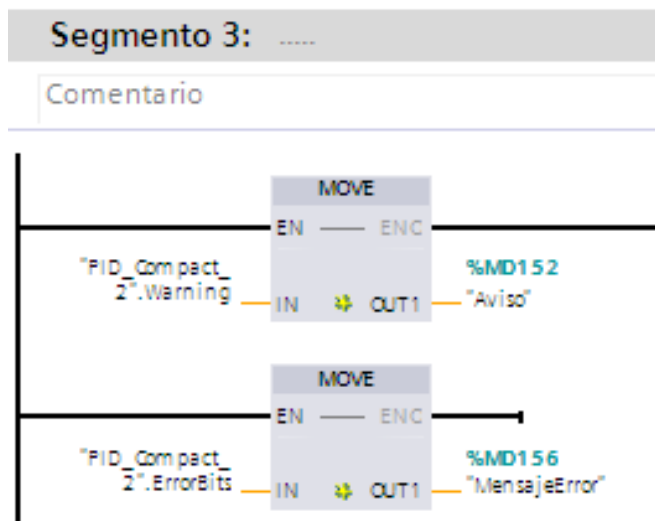


Figura 4-8. Segmento 3 del bloque de límites

4.2.5 Optimizaciones

En este OB se tratará de configurar ciertos parámetros internos de la optimización inicial y fina.

Se transfiere el valor de los siguientes tres parámetros a una variable del PLC de manera que se pueda visualizar por la pantalla el valor obtenido adecuadamente:

- Progreso_optimizacion: Expresa el grado de avance de la optimización correspondiente en %. Se evalúa tanto para la optimización inicial como para la fina.
- Estado_optimizacion_inicial: Muestra la fase actual de la optimización inicial mediante un entero:
 - State = 0: No se ha iniciado la optimización inicial
 - State = 100: Se está calculando la desviación estándar
 - State = 200: Determinando el punto de inflexión
 - State = 9900: La optimización inicial ha sido completada
 - State = 1: Optimización inicial no correcta
- Estado_optimizacion_fina: Muestra la fase actual de la optimización fina mediante un entero:
 - State = 0: No se ha iniciado la optimización inicial
 - State = 200: Se está calculando la desviación estándar
 - State = 300: Intentando alcanzar la referencia
 - State=400: Intentando alcanzar la referencia con los parámetros PID actuales
 - State=500: Determinar la oscilación y calcular parámetros
 - State = 9900: La optimización fina ha sido completada
 - State = 1: Optimización fina no correcta
 - State=-100: No es posible realizar esta optimización, se debe realizar una optimización inicial.

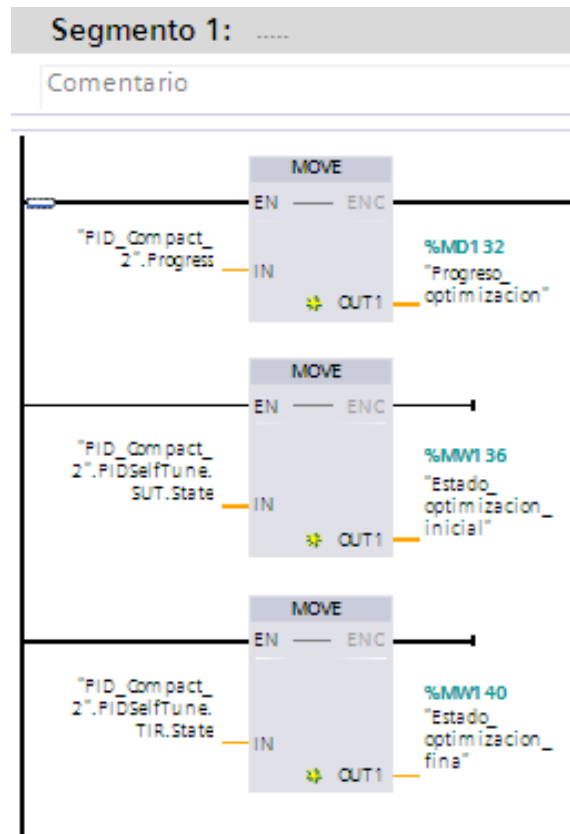


Figura 4-9. Bloque de optimizaciones

4.2.6 Parámetros

Este bloque tratará de transferir los valores obtenidos por la pantalla de interfaz HMI a los parámetros internos del regulador PID.

- En el primer segmento se transfieren los parámetros principales de control de cualquier sistema como son la ganancia, la constante de tiempo integral (Ti) y la de tiempo derivativo (Td). Estos parámetros solo se transferirán si el estado de operación es inactivo (=0).
- En el segundo segmento se realiza el proceso inverso, es decir, se pasan los parámetros internos del bloque PID a variables globales del proyecto. Esta transmisión de valores se realizará en todo momento para que se puedan visualizar por la pantalla los valores reales.
- En el tercer segmento, la variable "Cargar_Parametros" se encargará con esta transferencia de cargar el último juego de parámetros mediante un flanco de subida.
- Finalmente se pasarán también otros parámetros auxiliares del bloque del PID para su posible configuración a través de la pantalla: coeficiente de retardo derivativo, ponderación de la ganancia y tiempo derivativo.

Segmento 1:

Comentario

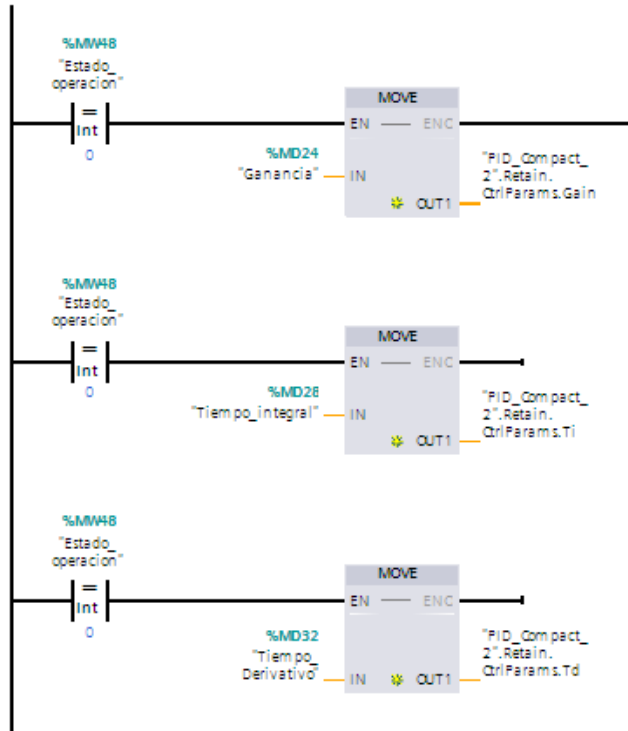


Figura 4-10. Segmento 1 del bloque de Parámetros

Segmento 2:

Comentario

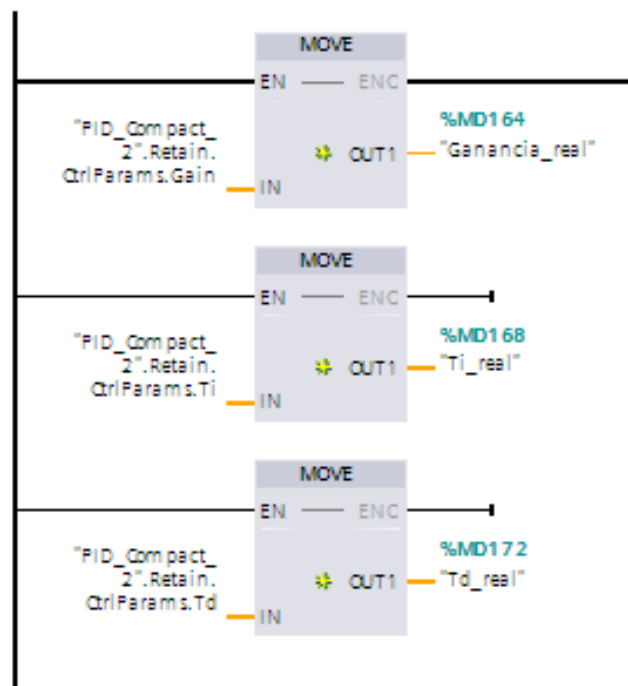


Figura 4-11. Segmento 2 del bloque de Parámetros

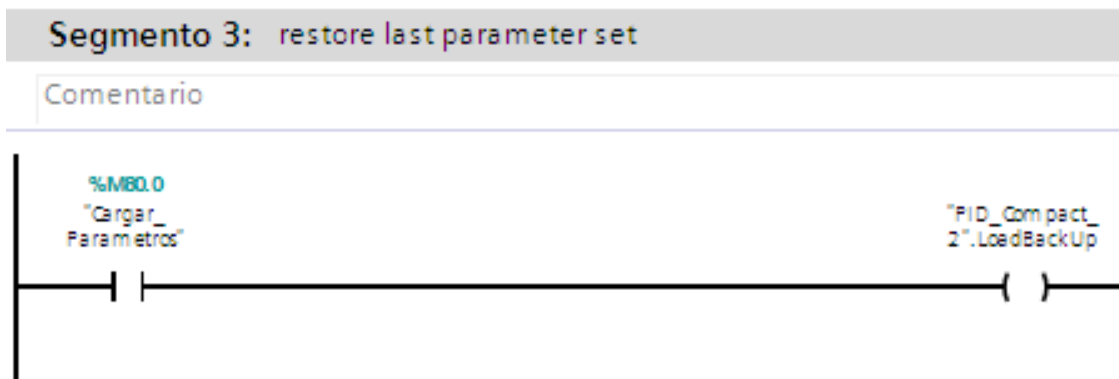


Figura 4-12. Segmento 3 del bloque de Parámetros

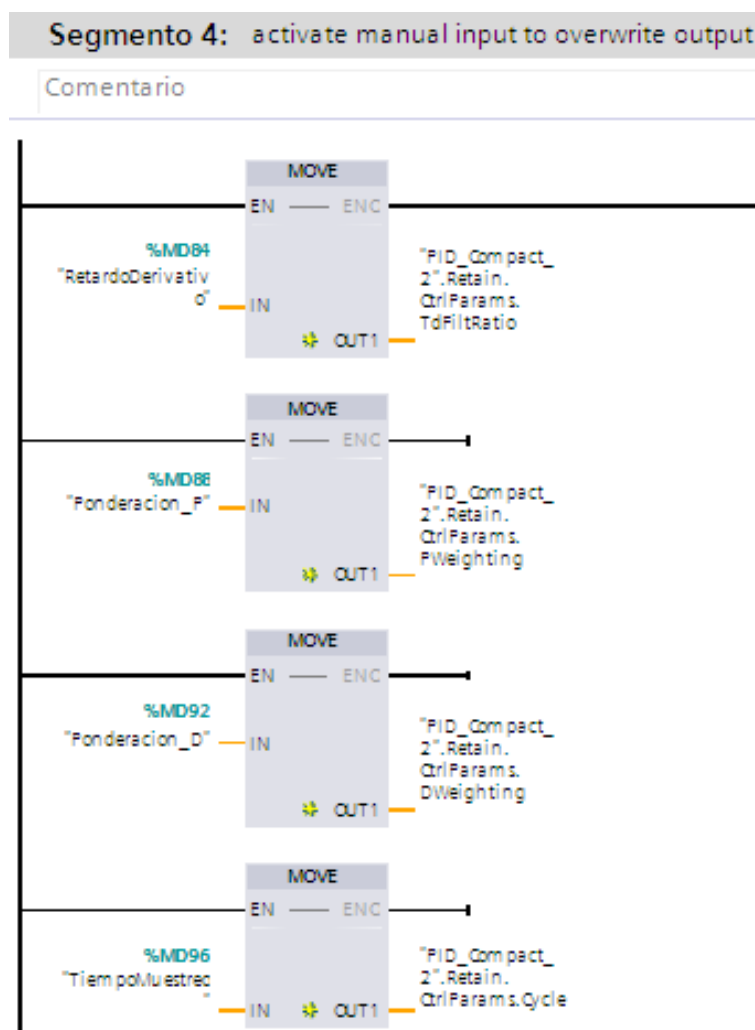


Figura 4-13. Segmento 4 del bloque de Parámetros

4.2.7 Resultados

Se programa este último bloque de forma que se puedan obtener una serie de resultados a través del servidor web como se explicará más adelante:

- En primer lugar, se crea un bloque de datos global llamado 'DatosArchivo' que se usará para definir algunos parámetros internos durante la creación del archivo. En la figura 4-12 se observan las variables que se han creado en el bloque de datos.

1	Static		
2	Name	String	'Resultados'
3	ID	DWord	16#0
4	Header	String	'T_medida, Vm...
5	Data	Struct	
6	Done	Bool	false
7	Busy	Bool	false
8	Error	Bool	false
9	Status	DWord	16#0

Figura 4-14. Bloque de datos globales

- En el primer segmento se crea el archivo de datos con formato .csv que se solicitará a través de la variable "Escribir Archivo" y guardará los últimos 800 valores de cada variable.
- Con las variables del bloque de datos se define: el nombre del archivo, las variables que se guardarán en el archivo y cada uno de los estados del archivo (Done, Busy y Error) además del estatus.
- Los segmentos 2 y 3 se utilizarán para escribir sobre el archivo cada 0.2 segundos en el caso de que el archivo siga activo.
- Finalmente, se guardan los valores correspondientes en las variables de la estructura Data que se encuentra a su vez definida en el bloque de datos global. Los valores elegidos serán:

La temperatura de entrada en °C, la temperatura en Voltios, la referencia en °C, la referencia en Voltios y la actuación real en Voltios.

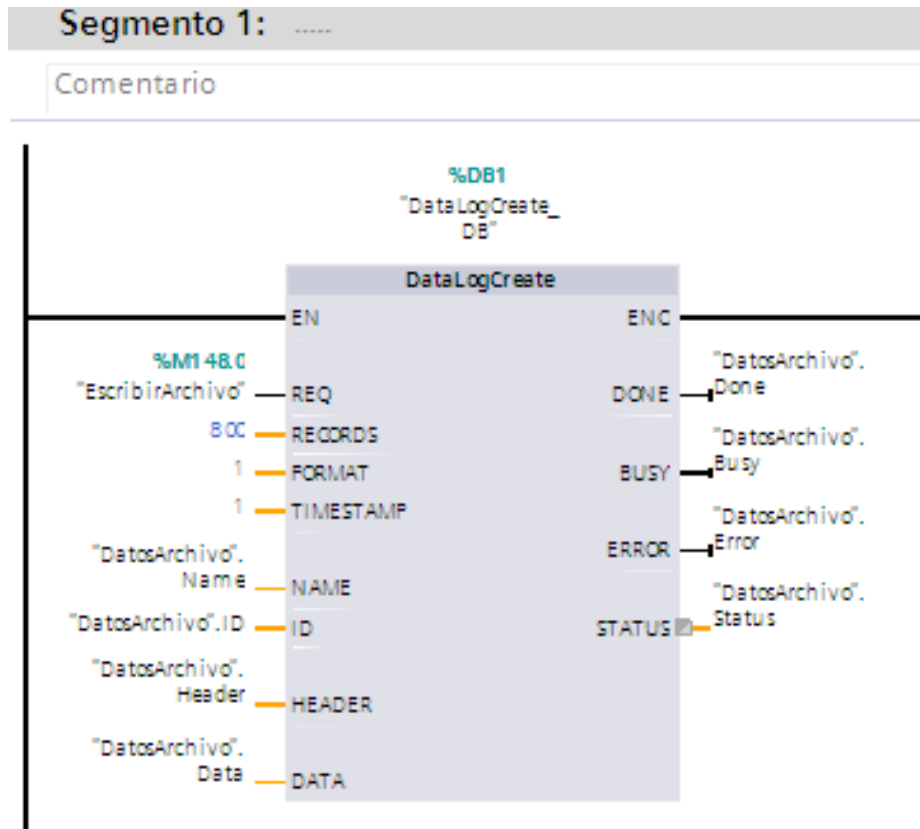


Figura 4-15. Segmento 1 del bloque de resultados

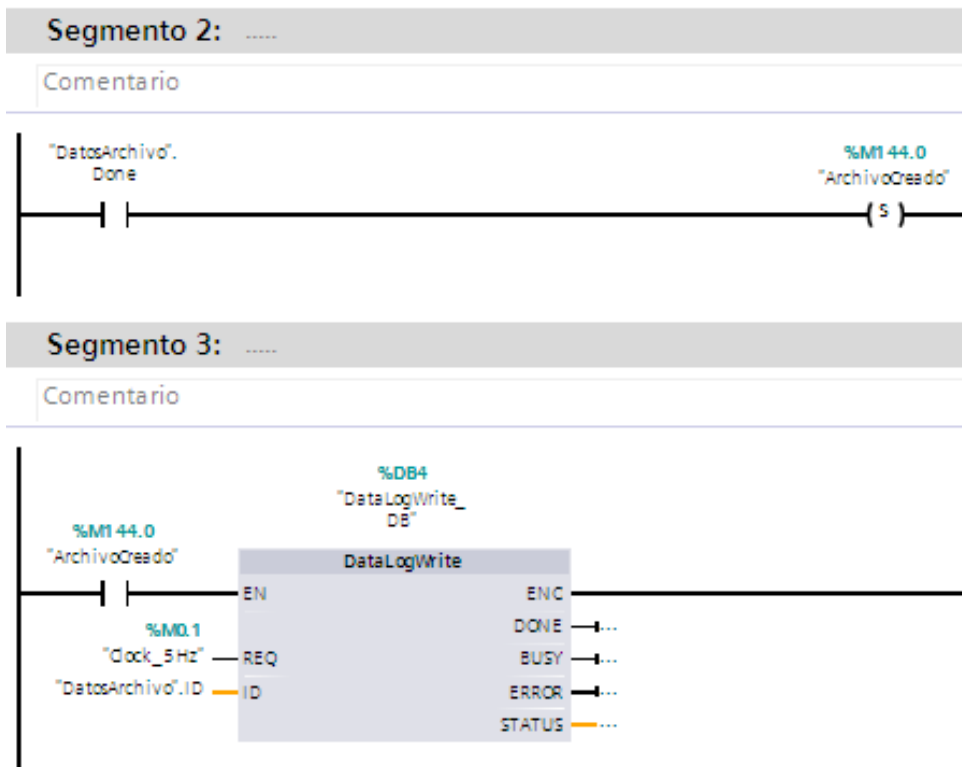


Figura 4-16. Segmentos 2 y 3 del bloque de resultados

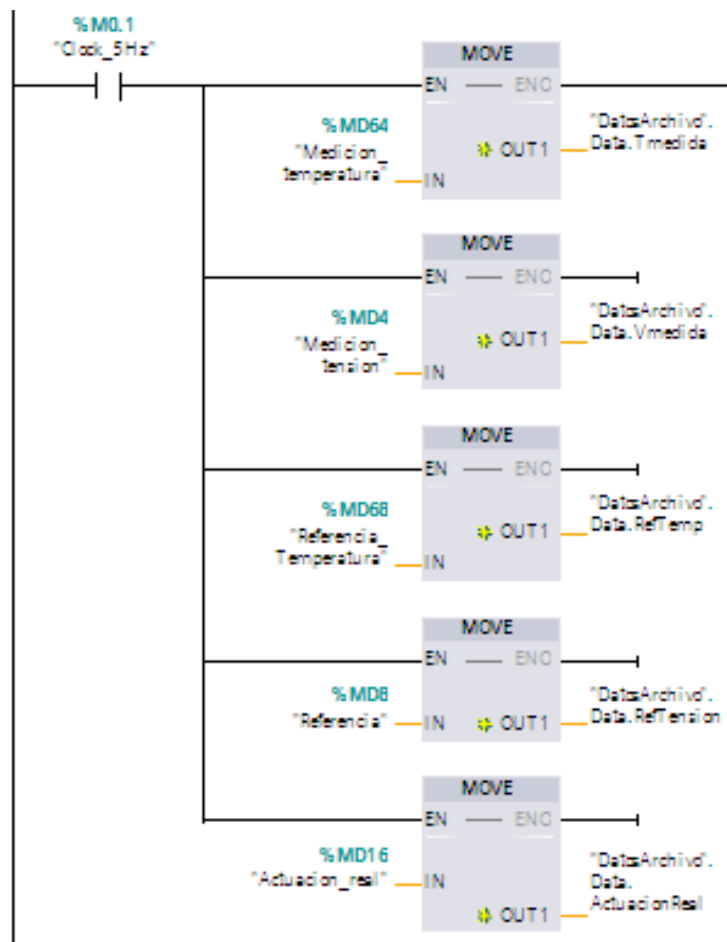


Figura 4-17. Segmento 4 del bloque de resultados

5 PROGRAMACIÓN DE LA PANTALLA

Este capítulo muestra la visualización final que se ha obtenido y cómo estará vinculada a la programación previa del PLC a través de las variables asociadas.

5.1 Variables de la pantalla

En primer lugar, se debe aclarar que cada variable de la pantalla estará directamente ligada a una del PLC de manera que al cambiar el valor a través de la pulsación correspondiente en un botón cualquiera de la pantalla, se estará cambiando automáticamente de valor la variable correspondiente del PLC.

Las variables creadas para manipular la pantalla serán las siguientes:

Tabla 5-1. Variables de la pantalla

Variable Pantalla	Variable asociada PLC	Direccionamiento
ActivacionModo	Activacion_modos	%M44.0
Actuador_real	Actuacion_real	%MD16
Aviso_LI_entrada	Aviso_LI	%M120.0
Aviso_LS_entrada	Aviso_LS	%M116.0
Cargar_Parametros	Cargar_Parametros	%M80.0
CofRetardoDerivativo	RetardoDerivativo	%MD84
Error	Error	%M56.0
Escribiendo	ArchivoCreado	%M144.0
EscribirArchivo	EscribirArchivo	%M148.0
Estado	Estado_operacion	%MW48
Estado_OptimizacionFina	Estado_optimizacion_fina	%MW140
Estado_OptimizacionInicial	Estado_optimizacion_inicial	%MW136
Kp	Ganancia	%MD24
LIReferenciaActivo	LI_setpoint_activo	%M100.0
LSReferenciaActivo	LS_setpoint_activo	%M104.0
Modo	Modo_operacion	%MW40

ModoManual	ModoManual	%M20.0
Modoreset	Reset	%M52.0
Ponderacion_D	Ponderacion_D	%MD92
Ponderacion_P	Ponderacion_P	%MD88
ProgresoOptimizacion	Progreso_optimizacion	%MD132
PuntoFuncionamiento	Pto_funcionamiento	%MD76
SalidaManual	SalidaManual	%MD36
T_medida	Medicion_temperatura	%MD64
T_referencia	Referencia_Temperatura	%MD68
Td	Tiempo_Derivativo	%MD32
Ti	Tiempo_integral	%MD28
TiempoMuestreo	TiempoMuestreo	%MD96
ValorAvisoLI	Valor_AvisoLI	%MD128
ValorAvisoLS	Valor_AvisoLS	%MD124
ValorLIReferencia	Valor_LI_Setpoint	%MD112
ValorLSReferencia	Valor_LS_Setpoint	%MD108
Kp_real	Ganancia_real	%MD164
Ti_real	Ti_real	%MD168
Td_real	Td_real	%MD172






5.2 Herramientas de visualización de la pantalla

En este apartado se muestra que las imágenes creadas en la pantalla no requieren un lenguaje de programación en sí, sino una distribución de los objetos utilizados que tendrán una serie de modificaciones en las variables.

Se debe distinguir entre dos tipos de objetos utilizados en la pantalla. El primer tipo corresponde a los objetos básicos que no tienen eventos asociados. Estos objetos se caracterizan porque la pulsación en ellos no modifica ninguna variable de la pantalla.

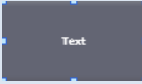


Lo más destacable de estos objetos son sus propiedades de apariencia y visibilidad, que irán variando en función de la evolución de sus variables asociadas. Los objetos de este tipo utilizados en el proyecto serán los siguientes:

Tabla 5-2. Objetos básicos de la pantalla

Objeto	Comentarios
Círculo 	Se puede desplazar por la pantalla y mostrarse en cualquier color o tamaño además de hacerse invisible
Línea 	Se suele utilizar para unir puntos pero tiene las mismas propiedades de desplazamiento apariencia y visibilidad que el anterior
Campo de texto 	Se puede modificar su formato de texto además de todas las anteriores
Visor de gráficos 	Permite incorporar a la pantalla cualquier imagen descargada previamente
Gráfica 	Permite representar la evolución de las variables en una gráfica auto-escalable

Por otro lado, se encuentran los elementos que tienen asociados eventos mediante su pulsación. Los eventos más utilizados en nuestro caso serán Activar/Desactivar Bit y Activar Imagen:

Tabla 5-3. Elementos de la pantalla

Elemento	Comentarios
Botón 	Se pueden modificar variables al pulsar, soltar o hacer bit. Las propiedades de apariencia y visibilidad siguen estando disponibles
Campo E/S 	Están asociados a una variable que se pueden modificar tanto manualmente como por programa. Se puede modificar el formato de representación
Interruptor 	Se asocia a una variable que se modificará según el estado del interruptor

La combinación de todos objetos que se acaban de mostrar dan lugar a una visualización que estará ligada a todo el conjunto de variables de la pantalla.

5.3 Visualización final

En este apartado se muestran las pantallas que han sido creadas tratando de explicar la utilidad de cada objeto y las variables asociadas a cada uno de ellos. Finalmente se realizará un esquema donde se muestra la relación que tienen las pantallas entre ellas a la hora de ser visualizadas.

5.3.1 Pantalla inicial

Al cargar el programa en la pantalla, la primera imagen que aparece muestra una imagen representativa de los tres elementos principales de nuestro proceso con un botón asociado a cada uno de ellos: la fuente, el controlador y el simulador de procesos PT326.

- La fuente ya está totalmente definida por lo que la pulsación de su botón no conduce a ningún lado.
- Al pulsar el controlador se activará la pantalla de controlador que da acceso a otras opciones que se explicarán a continuación.
- Al pulsar el botón del simulador de procesos se activa la visualización en una gráfica de ciertos parámetros que definen el comportamiento del sistema.



Figura 5-1. Pantalla inicial

5.3.2 Pantalla de controlador

Esta imagen es equivalente a la pantalla principal que tendría un operario encargado de controlar este sistema ya que da acceso a todos los parámetros de control del PLC:

- Muestra el estado actual de operación del regulador PID junto con otros tres elementos luminosos que se activan en caso de que el valor de entrada supere los límites superior e inferior o en caso de error en la regulación.

- Se añade otro botón con su elemento luminoso correspondiente para resetear el regulador. Mientras el sistema esté en modo reset la salida en tensión será 0.
- Finalmente, los botones de la parte de inferior de la imagen activan otras pantallas, ya sean para volver a la pantalla anterior o para modificar el resto de variables del regulador.



Figura 5-2. Pantalla de controlador

5.3.3 Modo de operación

Esta pantalla ofrece la posibilidad de manejar ciertas utilidades del PLC que se deben manejar correctamente:

- La columna izquierda de la imagen da la posibilidad de modificar el modo de operación y hacer efectivo mediante la pulsación del botón de activación de modo. Además, se puede acceder a la pantalla de visualización o volver a la anterior para ver si se ha hecho efectivo este cambio.
- La columna central de la imagen da la posibilidad de activar el modo manual y establecer un valor de salida en % de 0 a 10V.
- La columna de la derecha activa la variable que crea un archivo. Se verá si se ha hecho efectivo mediante el campo E/S de abajo. Para reiniciar el archivo se debe poner este último campo a 0.

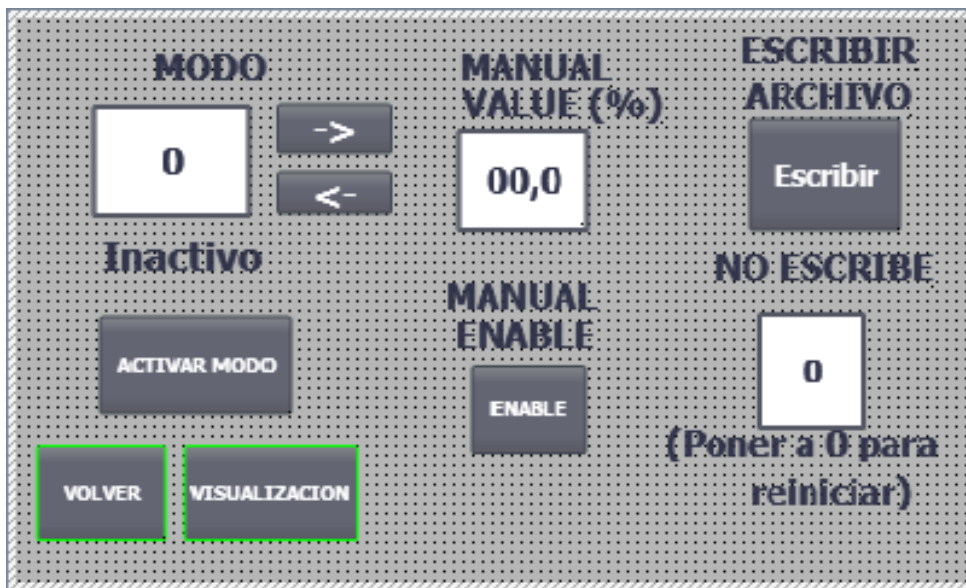


Figura 5-3. Pantalla de modos de operación

5.3.4 Control de parámetros

Al pulsar el botón de control en la pantalla de controlador se activa la siguiente pantalla donde se muestran los siguientes elementos:

- Imagen de la función de transferencia del regulador con la explicación de cada término
- Tres campos de entrada/salida donde se pueden modificar los parámetros principales del regulador en estado inactivo junto con otros tres campos de salida que muestran el valor real en cada instante → Ganancia, Tiempo integral y Tiempo derivativo.
- Otros tres parámetros auxiliares en la esquina inferior derecha de la pantalla que corresponden a los parámetros de la función de transferencia.
- Los botones que activan las pantallas de realimentación y de otros parámetros junto con otro botón de volver a la pantalla del controlador.

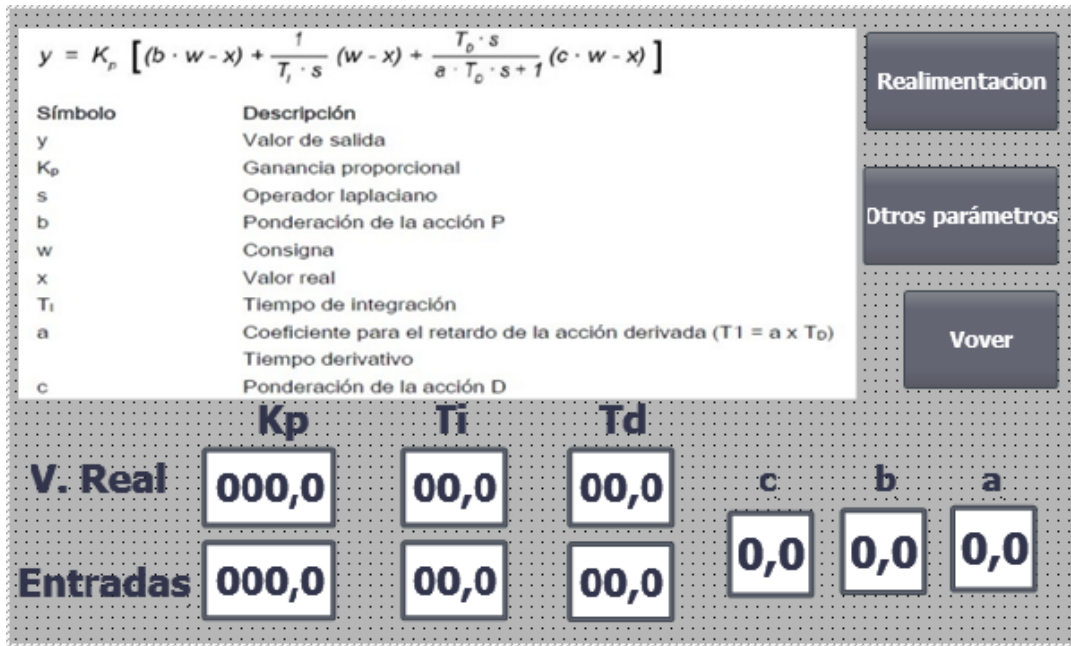


Figura 5-4. Pantalla de parámetros del controlador

5.3.5 Implementación

En esta pantalla aparece un visor del esquema de realimentación que utiliza el regulador. Se ha decidido mostrar este esquema en una pantalla por separado para su correcta visualización.

Además, se adjunta un botón para volver a la pantalla de parámetros explicada anteriormente.

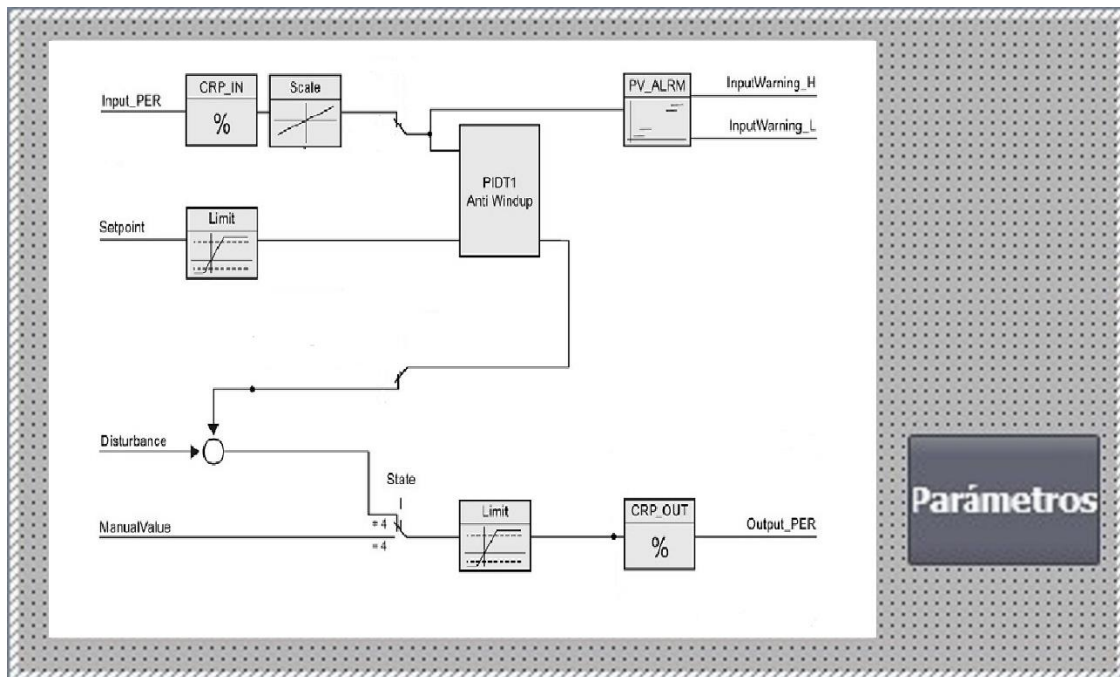


Figura 5-5. Pantalla de implementación

5.3.6 Otros parámetros de control

En esta pantalla se muestran los parámetros auxiliares del regulador PID correctamente visualizados junto con otro campo de entrada/salida y un botón que sirve para activar el último juego de parámetros que se haya configurado.

Finalmente se añade otro botón para volver a la pantalla anterior.

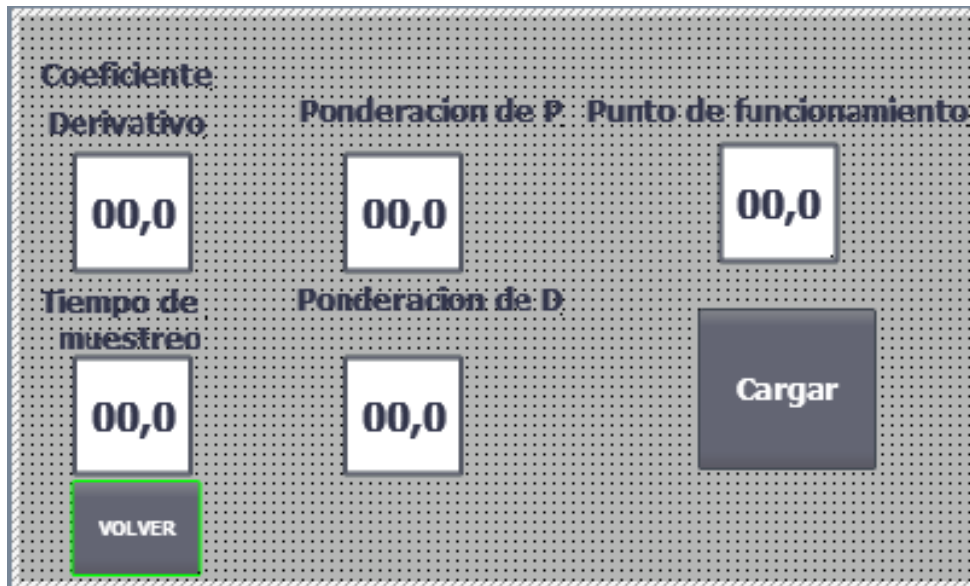


Figura 5-6. Pantalla de otros parámetros

5.3.7 Límites

Esta pantalla corresponde a los límites establecidos para los valores del regulador:

- En la parte superior de la imagen se establecen los límites superior e inferior (inicialmente serán 8.0 y 0.0 respectivamente) para los valores de entrada junto con sus correspondientes objetos luminosos que se activarán en caso de que se rebase alguno de estos límites.
- En la parte inferior de la imagen se establecen los límites superior e inferior (inicialmente serán 8.0 y 0.0 respectivamente) para los valores de referencia junto con sus correspondientes objetos luminosos que se activarán en caso de que se rebase alguno de estos límites.
- Finalmente se adjunta un botón para volver a la pantalla del controlador.

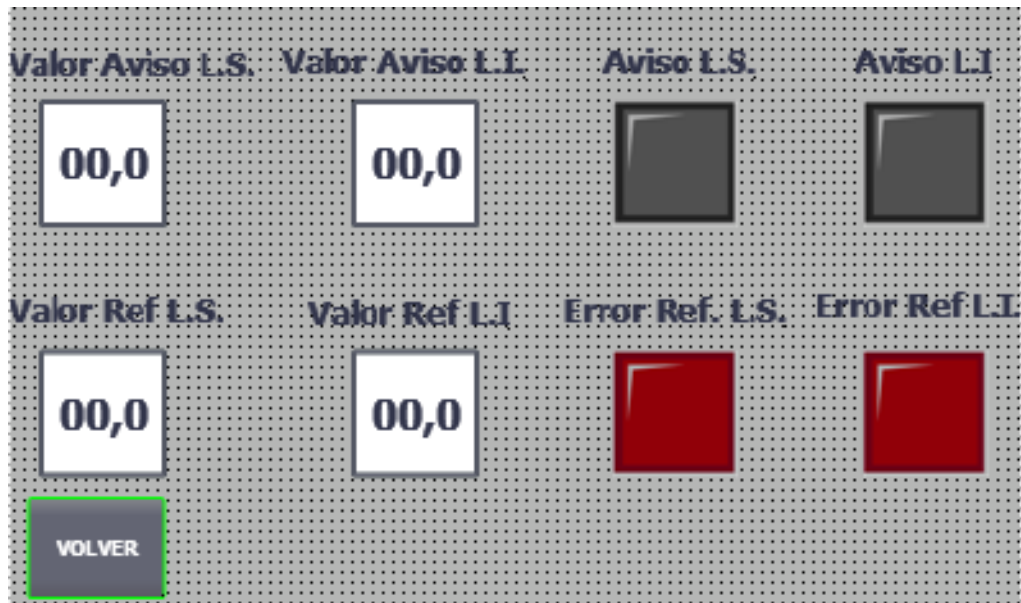


Figura 5-7. Pantalla de límites

5.3.8 Autoajuste

Esta pantalla está creada para el manejo de las dos optimizaciones posibles del regulador. Se añaden los siguientes objetos gráficos:

- En la parte izquierda de la imagen se observa un campo de E/S con el estado del regulador además de otro campo que muestra el porcentaje de cada optimización en %. Se recuerda que el estado 1 será la optimización inicial y el 2 la optimización fina.
- En la parte derecha de la imagen se muestran los campos E/S con un entero que corresponde al estado de cada optimización junto con un botón que inicia la optimización en cuestión.

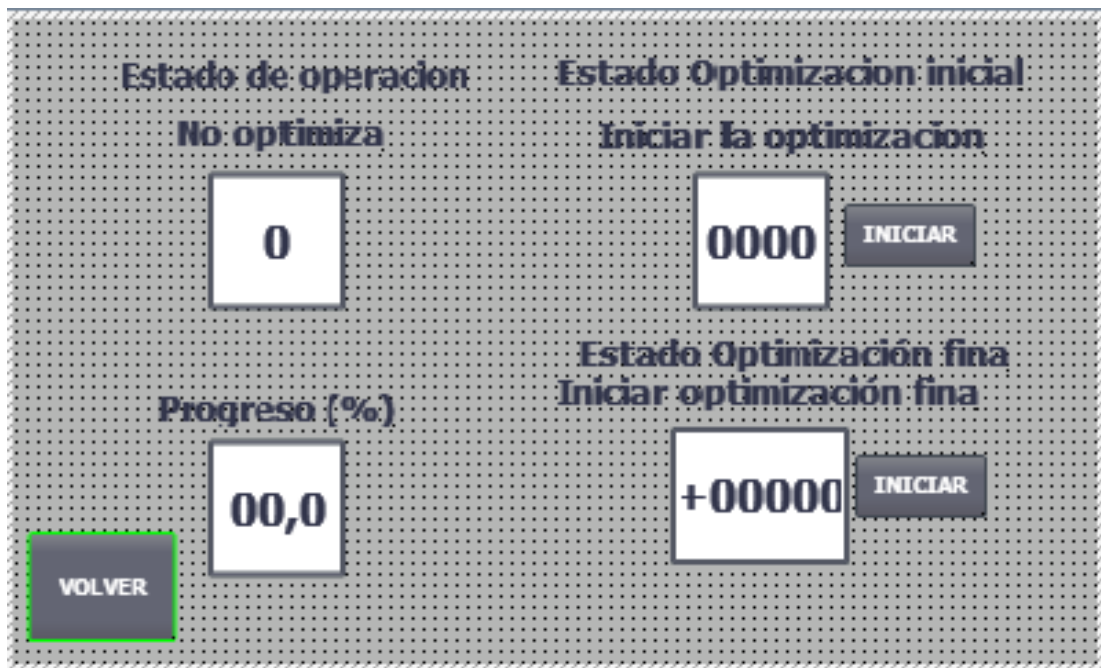


Figura 5-8. Pantalla de autoajuste

5.3.9 Visualización

En esta última pantalla se muestra una representación gráfica junto con un campo E/S con el valor de la referencia en °C y un texto que muestra el estado de operación en cuestión. Además, se añade el botón para volver a la pantalla anterior.

En la gráfica (que incluye auto-escalado) se observará la evolución en el tiempo de la temperatura del simulador de procesos (°C), la referencia (°C) y la actuación real (V). Además, se mostrará el valor real de cada variable en el punto medio de la gráfica.

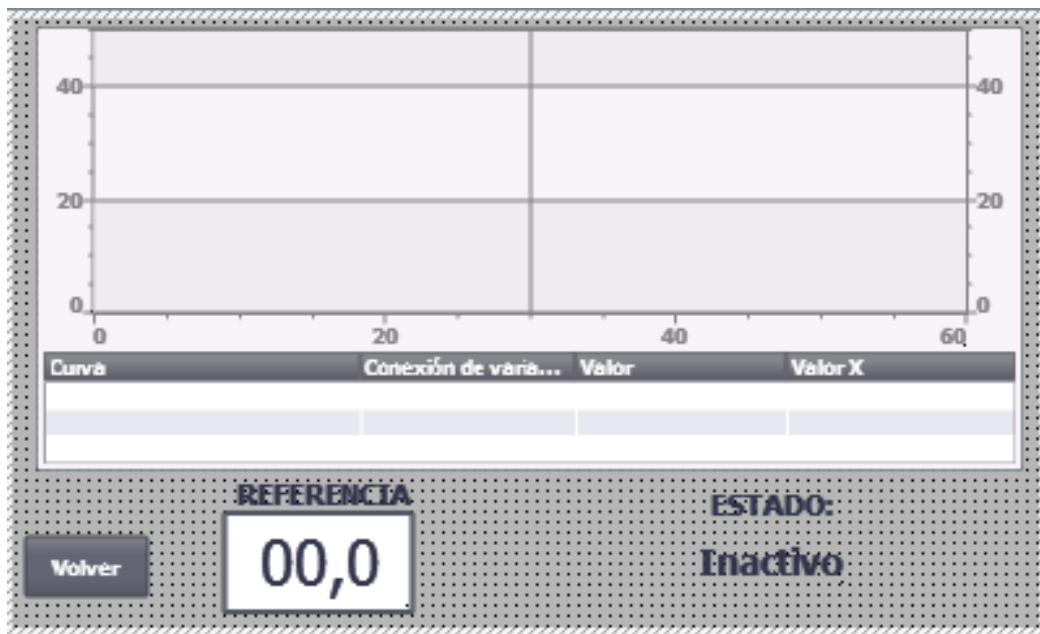


Figura 5-9. Pantalla de visualización de variables

5.3.10 Esquema de distribución de las pantallas

En este último apartado se muestra un esquema aclaratorio que muestre la relación entre cada una de las pantallas, es decir, cómo se activan unas pantallas a través de otras y los botones que se pulsán para que esto ocurra

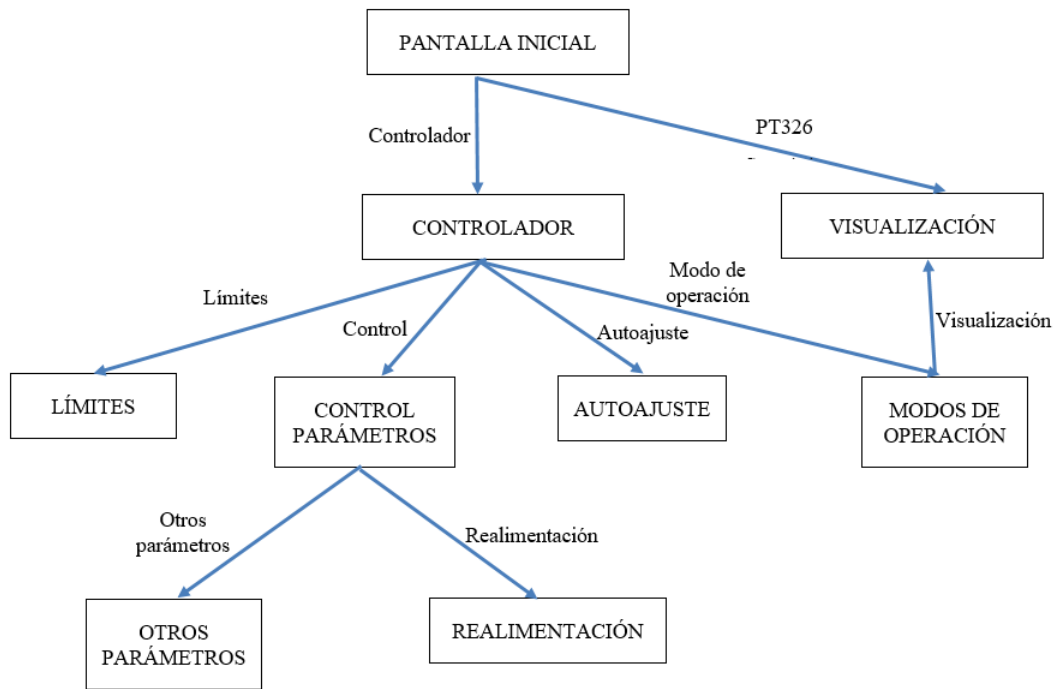


Figura 5-10. Esquema de interrelación de pantallas

6 MANEJO DEL SERVIDOR WEB

Todos los PLC SIMATIC S7-1200 tienen acceso a un servidor web con distintas aplicaciones online que se tratan de explicar en este capítulo. Además, se muestra el uso de una de estas aplicaciones correspondiente a los resultados obtenidos a través del servidor.

6.1 Configuración previa

En primer lugar, se debe habilitar el servidor web del PLC. Para ello se accede a través de la ventana de configuración del dispositivo en el árbol del proyecto, pulsando sobre la entrada de conexión Ethernet que se observa en la figura.

Una vez que se abra la ventana emergente, dentro de la ventana de Servidor Web, se habilita la pestaña ‘Activar servidor Web en todos los módulos de este dispositivo’ como se observa en la figura 6-1.

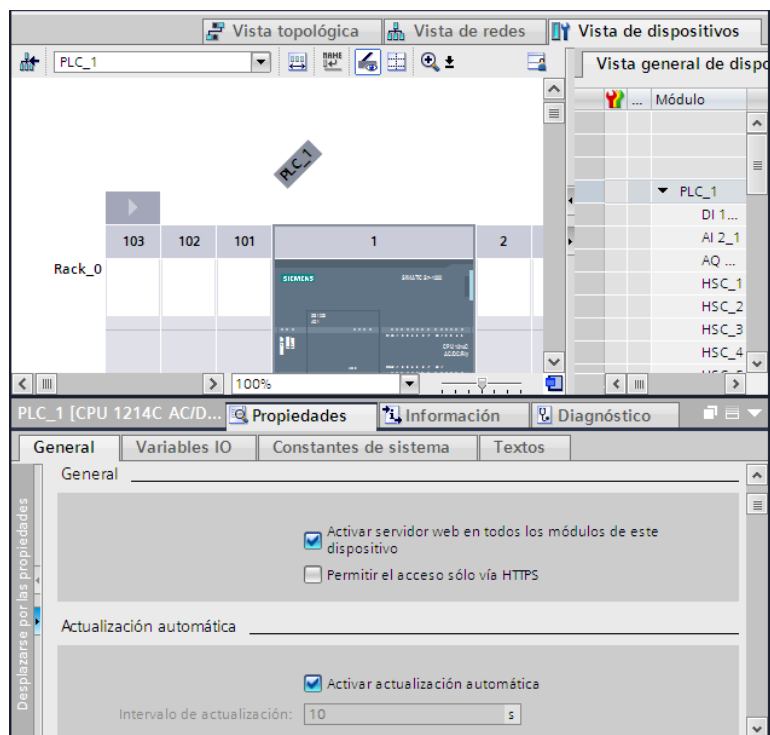


Figura 6-1. Habilitación del servidor Web

Para tener acceso al manejo online, se debe crear un usuario (con su respectiva contraseña) que tenga acceso como administrativo (acceso a todas las utilidades) como se observa en la figura 6-2. Esta configuración se realiza en la misma ventana en la que se habilitó el servidor web.

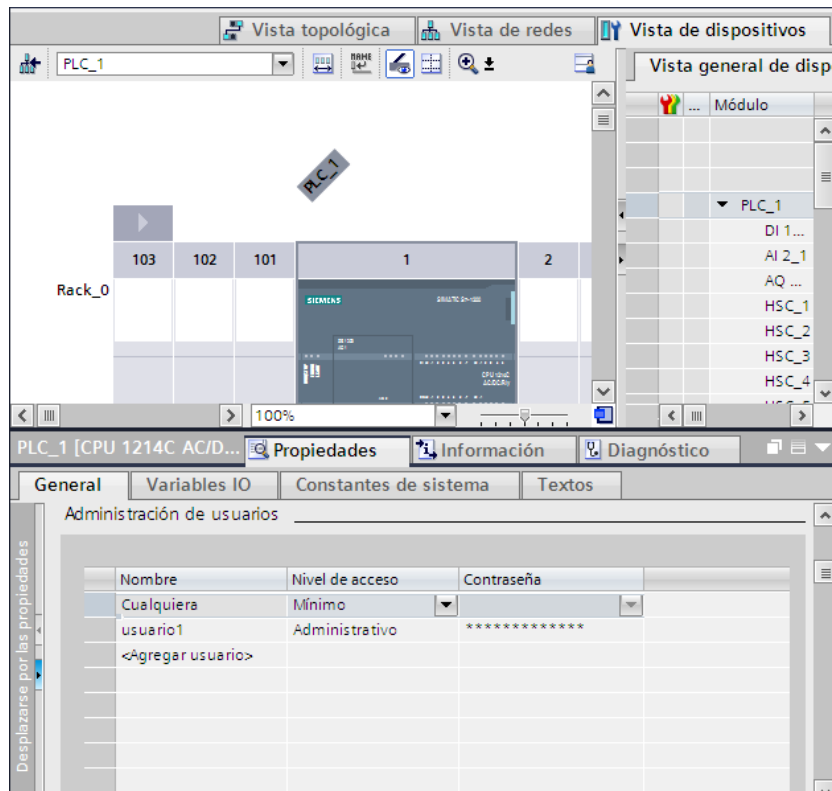


Figura 6-2. Crear usuario del servidor web

A continuación, para asociar una página web HTML desde la que se pueda actuar sobre el PLC, en el apartado de páginas web definidas por el usuario (dentro de la misma ventana), se debe establecer el directorio del Proyecto dentro del campo de “Directorio HTML” y la página web en cuestión desde la que se actuará dentro del apartado de “página de inicio HTML” como se observa en la figura 6-3.

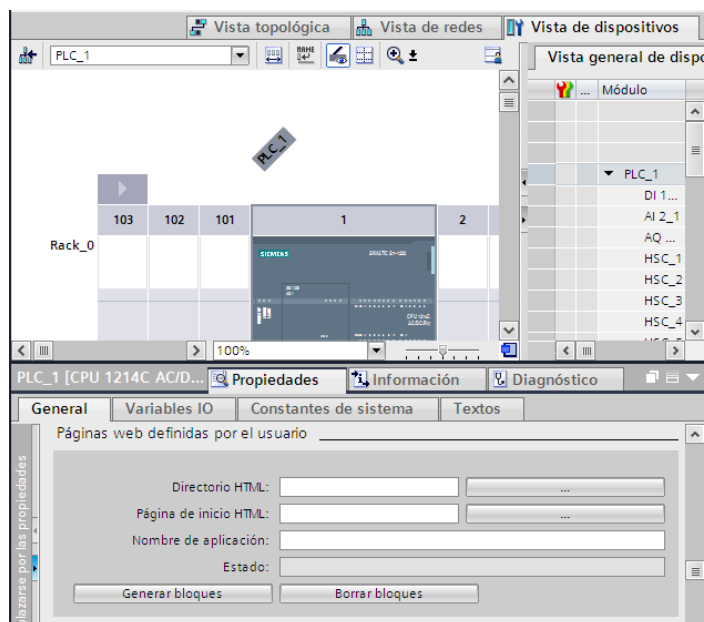


Figura 6-3. Página web definida por el usuario

Finalmente, una vez que se ha cargado de nuevo el dispositivo con los cambios introducidos, se configura la dirección IP en el navegador y se accede con el usuario que previamente se han creado.

6.2 Utilidades

A continuación, se explica cada una de las opciones que ofrece el servidor web una vez que se ha iniciado el usuario correctamente como se observa en la página inicial de la figura 6-4:

- La ventana de página de inicio de la figura permite conocer algunos datos del autómeta a nivel general e incluso el modo de operación actual. Por su parte la ventana de identificación aportará información del autómeta con datos referentes al número de serie junto con la versión de Hardware y Firmware.
- Existe una ventana de buffer de diagnóstico del autómeta, donde se pueden observar los últimos eventos realizados por el autómeta, lo que será muy útil en caso de parada o fallo. También se añade otra ventana de módulo de información que en nuestro caso solo incluirá un PLC S7-1200 y se puede acceder fácilmente a través del PLC.
- En la ventana de comunicación existe información sobre el tipo de conexión que existe, incluyendo la dirección IP y la máscara subred que ha sido definida en el proyecto previamente.
- En el apartado de ‘Páginas de usuario’ se incluye un acceso a la página web desde la que se controlará el autómeta. Se pueden ver las variables utilizadas en la página a través de la ventana de ‘Estados de Variables’.
- Finalmente, en el apartado de ‘Navegador de archivos’ se crearán archivos de resultados con el nombre y los datos que se han programado en el PLC previamente en el bloque de organización ‘Resultados’.

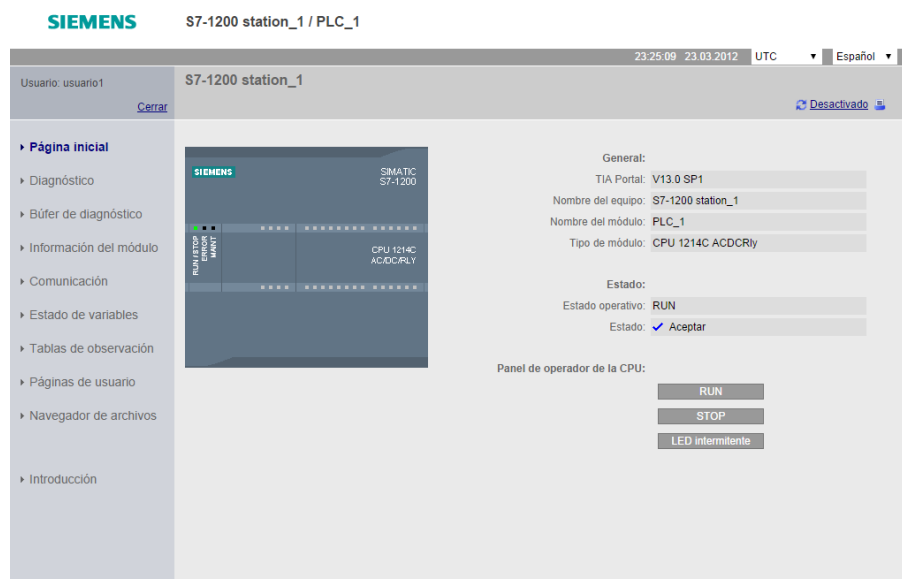


Figura 6-4. Página inicial del servidor web

6.3 Aplicación del servidor: Resultados

Se debe recordar que en la programación del PLC se decidieron obtener los resultados de los siguientes datos:

Medida de la temperatura (°C), entrada de temperatura (Voltios), referencia en temperatura (°C), referencia en tensión (Voltios) y actuación real (Voltios).

Se han realizado una serie de pruebas cuyos resultados han sido obtenidos en un archivo .csv a través del servidor web en la ruta: Navegador de archivos/DataLogs que se observa en la figura 6-5.

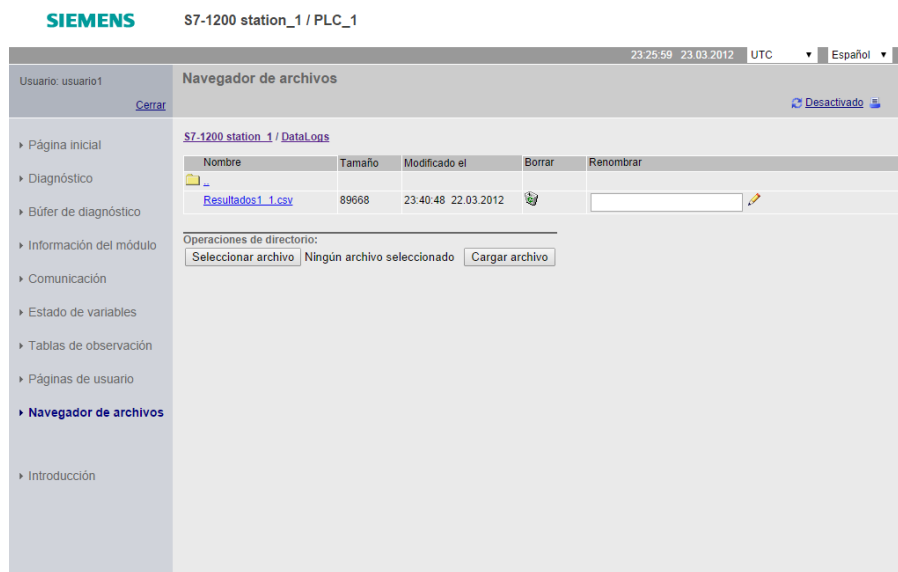


Figura 6-5. Recogida de resultados

6.3.1 Representación de resultados

Se han realizado tres pruebas al autómatas cuyos resultados se mostrarán a continuación a través de una representación gráfica de la evolución de las variables principales:

Escalones en la salida

La primera prueba consistirá en ver la evolución de la temperatura en la entrada a medida que se va subiendo y bajando la salida en tensión mediante escalones. Esta prueba se realizará en modo manual.

En primer lugar, se muestra la evolución de la entrada en tensión respecto a su actuación:

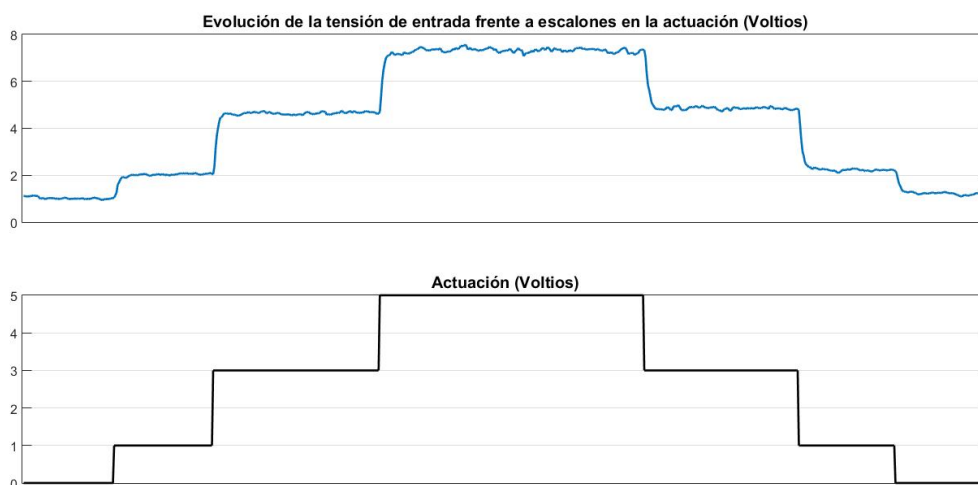


Figura 6-6. Resultado en tensión frente a escalones en la actuación

Con estos resultados se puede definir el sistema mediante un modelo lineal de primer orden con los siguientes parámetros:

Ganancia del sistema= 1.4 y Tau=0.35

Si se escala la entrada en temperatura ($^{\circ}\text{C}$), el resultado frente a la actuación sería el siguiente:

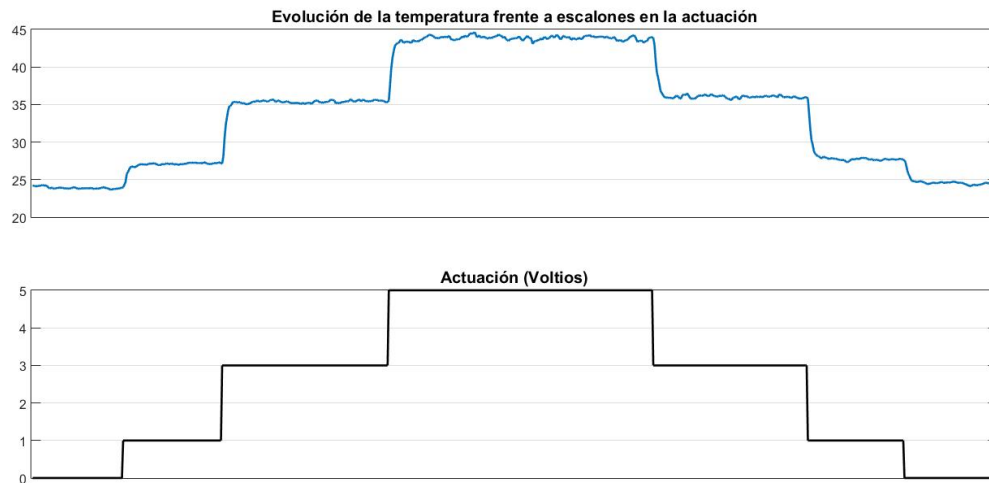


Figura 6-7. Resultado en temperatura frente a escalones en la actuación

Escalones en la referencia

Gracias al modelo teórico creado en el apartado anterior, se crea un controlador PI que ayudará a controlar el sistema en bucle cerrado por cancelación de dinámicas con los siguientes parámetros:

$K_p=1.2$, $T_i=0.35$

Una vez que el controlador queda definido se observa cómo evoluciona el sistema frente a subidas y bajadas en la referencia.

Los resultados obtenidos en tensión serán los siguientes:

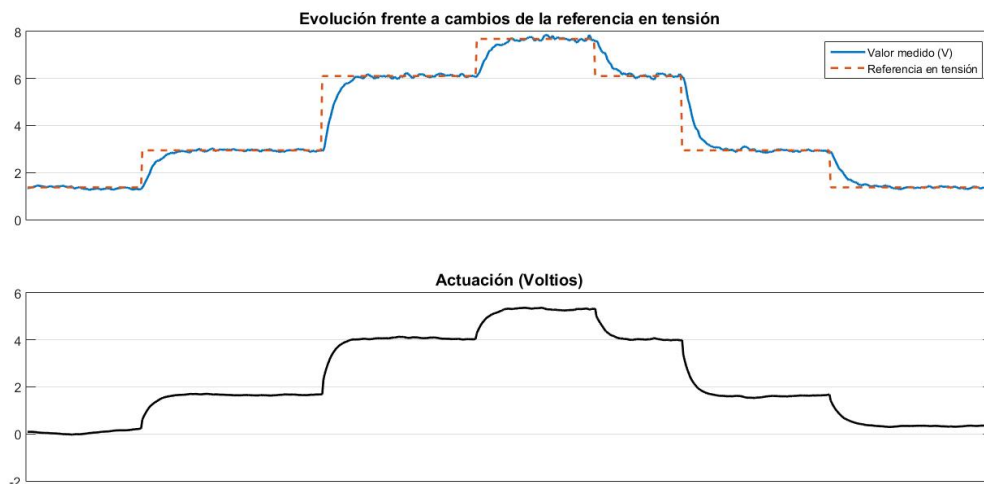


Figura 6-8. Resultado en tensión frente a escalones en la referencia

Se observa que la respuesta del sistema es bastante aceptable con el controlador que se ha creado. El sistema no tiene sobreoscilación para evitar picos de tensión que pueda dañar el PLC y su tiempo de subida se establece en torno a 1 segundo.

Escalando los valores de entrada al PLC en temperatura el resultado sería el siguiente:

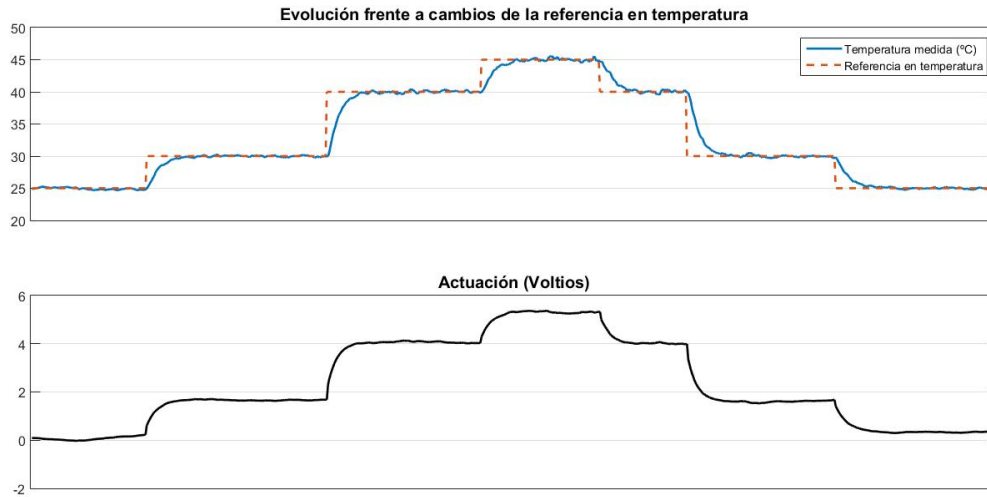


Figura 6-9. Resultado en temperatura frente a escalones en la referencia

Perturbaciones

Finalmente se somete al sistema a perturbaciones considerables frente a las que tenga que reaccionar adecuadamente. Para ello se modifica la entrada de aire al tubo de manera brusca y se evalúa el resultado.

La rejilla que controla la apertura de aire de entrada al sistema inicialmente colocada a 25° pasa 70° mientras se mantiene el valor de 35°C en la referencia en temperatura. A continuación, cuando el sistema está estabilizado se realiza el proceso inverso colocando de nuevo la rejilla a 25°.

El resultado obtenido en tensión es el siguiente:

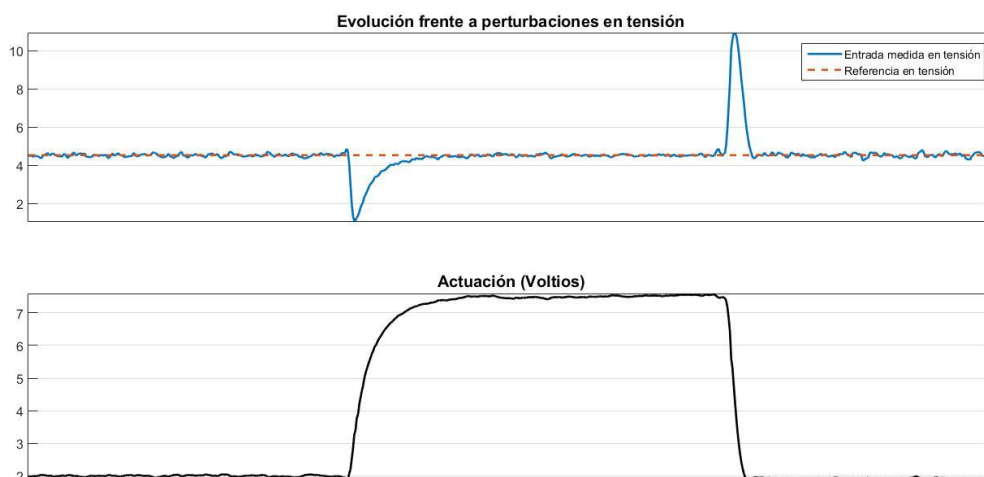


Figura 6-10. Resultado en tensión frente a perturbaciones

Se observa que la entrada sufre grandes alteraciones incluso saliendo de los límites establecidos (en este caso salió un mensaje de aviso). A pesar de ello el sistema funciona correctamente y recupera la tensión de referencia rápidamente.

Si se escala la entrada en temperatura el resultado sería el siguiente:

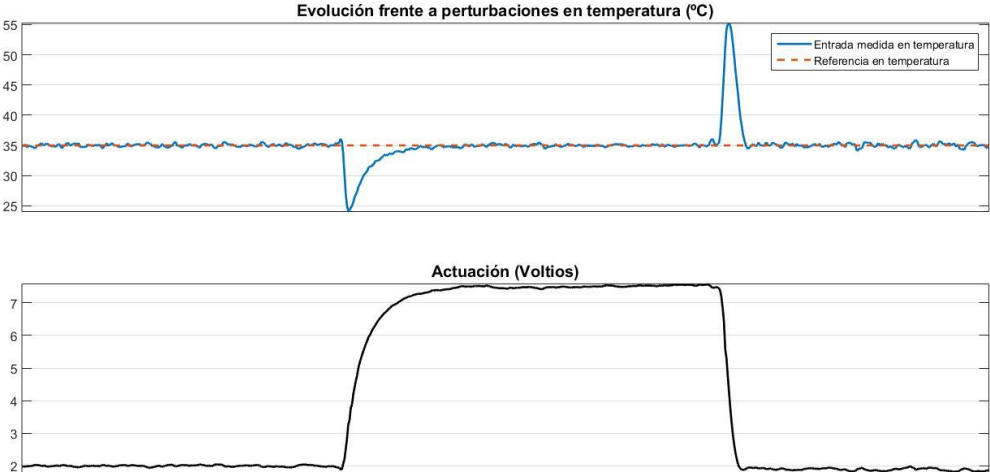


Figura 6-11. Resultado en temperatura frente a perturbaciones

7 CONCLUSIONES

Este capítulo resumirá las dificultades que se han encontrado durante la realización del proyecto y las conclusiones obtenidas sobre los objetivos alcanzados en el proyecto. Finalmente, se añadirán algunas recomendaciones para futuros trabajos que continúen esta línea.

7.1 Problemas encontrados

El programa TIA Portal utilizado para este proyecto tiene un manejo bastante intuitivo tanto para la programación del PLC en lenguaje de contactos como para la visualización de la pantalla. Aun así, el aprendizaje de la utilización adecuada ha supuesto una serie de dificultades que se muestran a continuación:

- **Direccionamiento virtual:** La memoria del programa permite direccionar variables desde la posición 0 hasta la 8191 con bits desde el 0 hasta el 7 en cada una de las posiciones de memoria. Se debe conocer que las variables direccionadas como %MW ocuparán 2 posiciones de memoria (2 Bytes, 16 bits) y las direccionadas como %MD ocuparán 4 posiciones (4 Bytes, 32 bits).

El problema surge cuando se intenta escribir una variable en las posiciones 0 o 1 de la memoria (estas posiciones están reservadas para variables internas del programa) o se intentan direccionar variables que se sobrescriben (por ejemplo %MD32 y %MD34). El programa no mostrará ningún mensaje de error, pero los resultados no serán adecuados.

- El simulador de procesos mide la temperatura y la envía al PLC en un rango tensión entre 0 y 10V. Tras varias pruebas de ensayo se detectó que la temperatura entre 0 y 8V seguía un comportamiento lineal con la temperatura medida en °C, a diferencia de lo que ocurría entre 8 y 10V. Para solucionar este problema se decidió establecer valores de aviso en la entrada y en la referencia cuando se superen los 8V.
- Una vez que la visualización de la pantalla está definida, todas las variables de la pantalla ligadas a variables del PLC sobrescribirán los valores calculados internamente por el PLC. Esto hizo crear dos variables para algunos parámetros internos del PLC: una de ellas que escriba sobre la variable interna del PLC en cuestión solo cuando sea necesario y otra que muestre el valor real de este parámetro interno en todo momento (es el caso de los parámetros internos Kp, Ti y Td).

7.2 Conclusiones y futuras propuestas

El objetivo fundamental de este proyecto era controlar un sistema en temperatura a través de las entradas y salidas del PLC dirigidas por la pantalla que servirá como interfaz hombre-máquina.

Este objetivo ha sido logrado ya que se han obtenido unos resultados que avalan el control del sistema frente a cualquier cambio en la referencia o cualquier perturbación dentro de los límites de actuación.

Gracias al manejo del servidor web se pueden exportar los resultados en formato csv para ser tratados desde otros programas, como por ejemplo Matlab o Excel, ya sea para su representación o para su análisis posterior.

Como propuesta para futuros trabajos, se puede intentar controlar el PLC a través de una página web ligada al servidor del propio PLC siguiendo las pautas marcadas en el capítulo anterior. Esto evitaría el control mediante pulsación táctil de la pantalla, que quedaría únicamente como herramienta de visualización de las variables requeridas.

Por otro lado, se ha conseguido un aprendizaje del manejo de la pantalla bastante avanzado que permite incorporar cualquier tipo de objeto en una imagen y cualquier animación o conexión con una variable del proyecto.

Se podrían desarrollar las animaciones de cada objeto en futuros proyectos en la pantalla, donde el movimiento de cada objeto, la visibilidad o la superposición muestre un papel más relevante respecto al control de un sistema o la evolución de las variables de forma gráfica.

También se podrán mostrar representaciones gráficas de otro estilo siguiendo la guía sobre la pantalla en la que se muestran todos los posibles objetos que ofrece el programa.

Finalmente cabe destacar que la visualización creada en la pantalla junto con la programación del PLC servirá para controlar cualquier sistema que ofrezca una medida en tensión siempre que se establezcan los parámetros de control correctamente, bien mediante el modo de optimización o mediante pruebas externas realizadas al sistema en cuestión.

REFERENCIAS

- [1] SIEMENS. *S7-1200, S7-1500, Regulación PID. Manual de funciones*, 2017
- [2] UNIVERSIDAD DE LA COSTA C.U.C. *Práctica de laboratorio. Controladores PID*. Barranquilla (Colombia): 2015
- [3] ESCUELA TÉCNICA SUPERIOR DE INGENIEROS, UNIVERSIDAD DE SEVILLA. *Práctica de periferia PLC S7-1200*.
- [4] SIEMENS. *S7 Controlador programable S7-1200. Manual de Sistema*, 2009
- [5] SIEMENS. *PID Control with PID_Compact*, 2014
- [6] SIEMENS. *Datasheet KTP700 Basic PN-6AV2123-2GB03-0AX0*, 2015
- [7] SIEMENS. *Datasheet 6ES7214-1BG40-0XB0*, 2018
- [8] CARRASCO FERNÁNDEZ, P. *Automatización de un parking mediante autómatas programables y SCADA*. Sevilla: 2018

GLOSARIO

%IW: Entrada con memoria de tipo Word	19
%QW: Salida con memoria tipo Word	19
%MD: Marca con memoria tipo Double-Word	20

Se ha añadido este anexo como guía para el manejo de la pantalla HMI SIMATIC de Siemens donde se tratarán detalladamente todas las posibles utilidades de esta interfaz hombre-máquina.

A.1 Configuración de la pantalla HMI

Para comenzar la configuración de la pantalla se realiza un procedimiento muy similar al del PLC, ya que el primer paso será elegir el modelo concreto que se va a utilizar. En este caso se usa el modelo de 7 pulgadas: 6AV2 123-2GB03-0AX0, como se observará en la siguiente ruta de la figura 8-1:

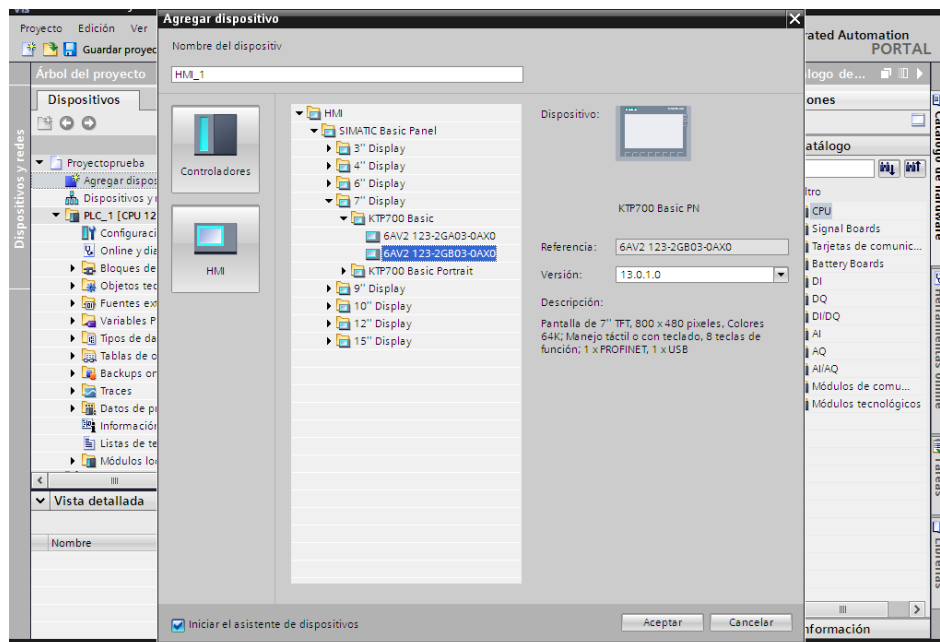


Figura A-1. Modelo de pantalla HMI

En este momento aparece un asistente de panel de operador que ayudará a dar los primeros pasos para configurar la pantalla.

En la ventana de conexiones de PLC, se debe examinar el PLC al que se desea asociar la pantalla. En este caso será el PLC_1 (se puede utilizar cualquier nombre asignado previamente). El driver utilizado será el SIMATIC S7 1200 y la interfaz será PROFINET como se observa en la figura 8-2.

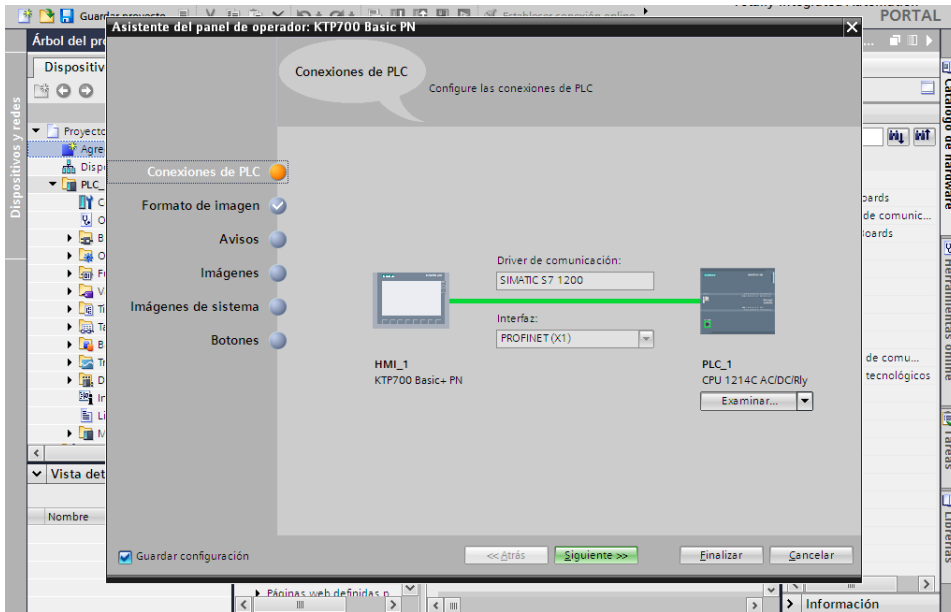


Figura A-2. Conexiones de PLC de la pantalla

En la ventana de imágenes, se configura la relación de las pantallas mediante un esquema como el que se observa en la figura 8-3. La imagen raíz podrá cambiarse de un lugar a otro una vez creado el proyecto, al igual que el esquema podrá cambiarse posteriormente, no será definitivo.

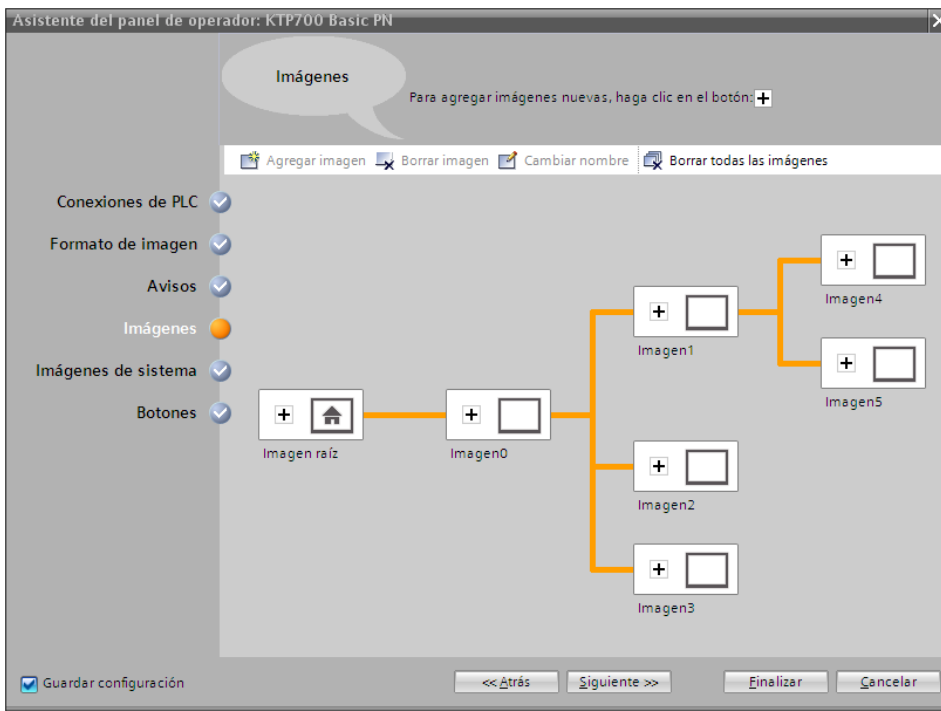


Figura A-3. Imágenes de la pantalla

Además de las configuraciones mencionadas, se podrá configurar el formato de la imagen, los avisos, las imágenes propias del sistema y los botones propios de las pantallas. Cuando se configuren todas las opciones, se pulsa finalizar para volver a la vista del proyecto.

Una vez que se vuelve a la vista del proyecto, se recomienda revisar la configuración del dispositivo al igual que

en el PLC. Una de las configuraciones más importantes será la dirección IP, que se realizará haciendo doble click sobre el símbolo de la entrada ETHERNET dentro de la ventana de la Interfaz PROFINET como se observa en la figura 8-4.

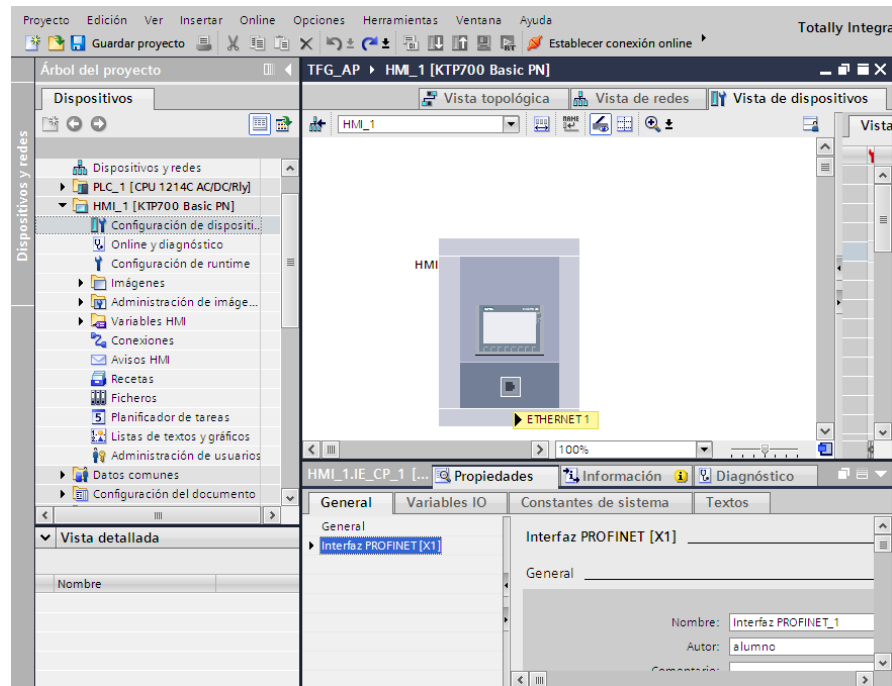


Figura A-4. Interfaz PROFINET del HMI

Para comprobar el tipo de conexión que existe entre el PLC y la pantalla basta con acceder a la ventana de conexiones de la figura 8-5 que se encuentra disponible en el árbol del proyecto.

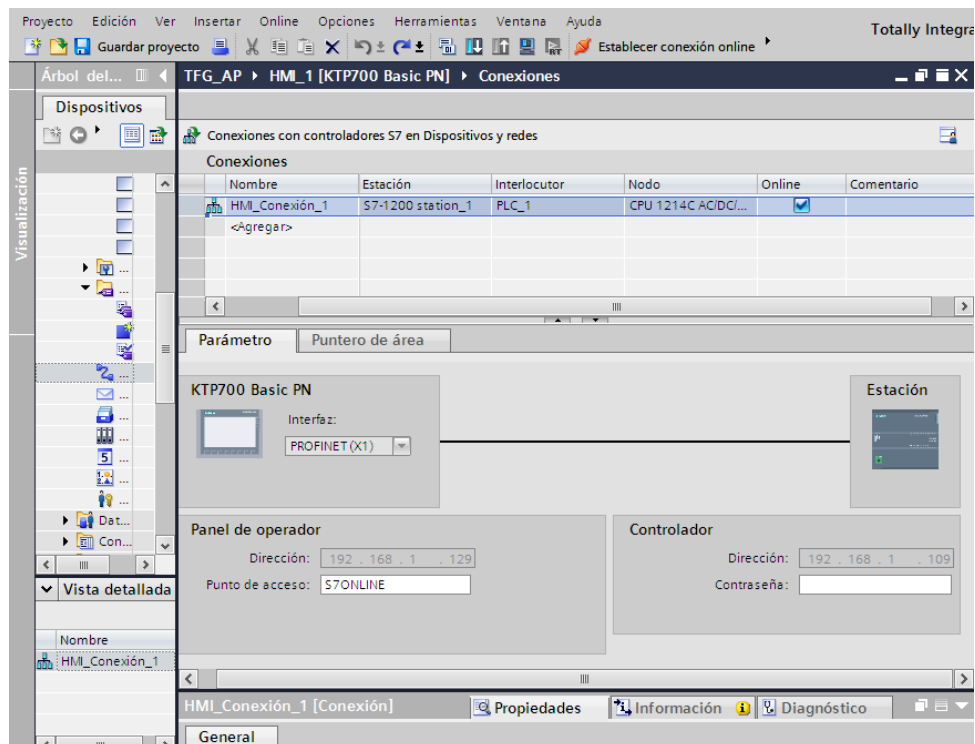


Figura A-5. Conexiones del HMI

Finalmente, cabe destacar que al igual que en el caso del PLC, para hacer efectivo cualquier cambio en la configuración del hardware o del software se debe compilar, y a continuación, cargar al dispositivo.

A.2 Visualización de la pantalla HMI

A partir de ahora se tratan las distintas herramientas que se tienen para crear una visualización adecuada de la pantalla en la que, como ya se adelantó en el capítulo anterior, no se utilizará ningún lenguaje de programación sino un conjunto de objetos que se distribuirán en la pantalla teniendo una serie de variables asociadas que modificarán (o no) según el tipo de objeto.

A.2.1 Plantillas

La primera herramienta que se debe modificar será la plantilla (o plantillas) que se utilizarán en el proyecto ya que estas plantillas serán plasmadas en cada una de las imágenes que se creen. Esta herramienta se encuentra dentro de la siguiente ruta en el árbol del proyecto:

HMI_1 (o cualquier nombre que se haya utilizado) / Administración de imágenes / Plantillas.

La plantilla que trae por defecto el programa se muestra en la figura 8-6 e incluye el nombre de la imagen, fecha, hora y los iconos de inicio y apagado de la pantalla junto con otros 6 botones potenciales sin configurar.

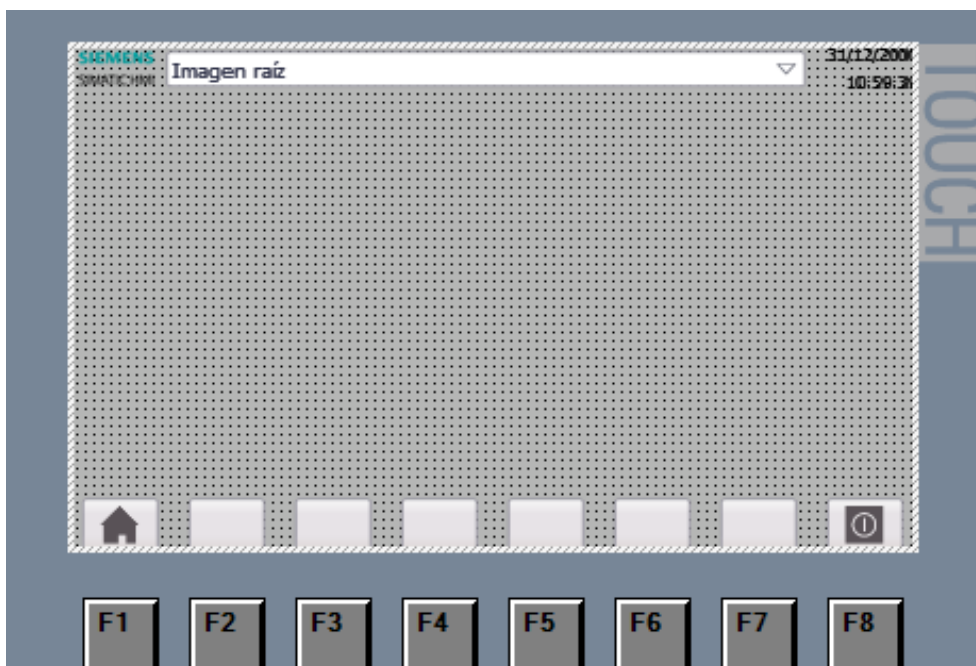


Figura A-6. Plantilla por defecto

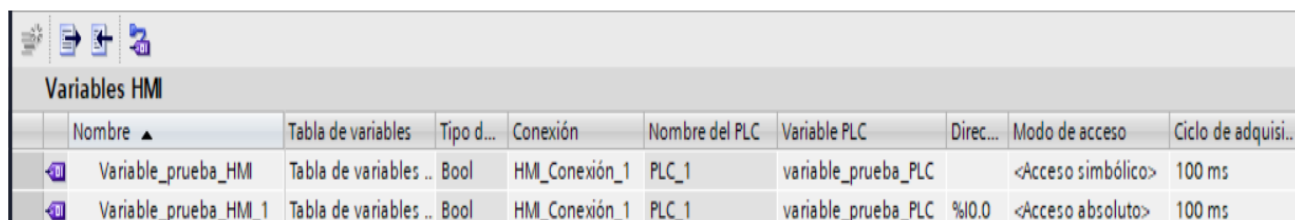
Cada uno de los elementos que aparece en esta imagen podrá ser modificado o eliminado (incluso se podrá incluir cualquier otro elemento) solo desde la ventana de plantillas. A su vez, todas las plantillas podrán ser asignadas a cada una de las imágenes e incluso se podrán agrupar en librerías de plantillas que serán muy útiles para proyectos extensos que necesiten variedad en cada una de sus imágenes.

Los usos más habituales de la plantilla serán la visualización común para una serie de imágenes y la configuración de los botones que aparecen en la parte inferior de la pantalla que podrán dar acceso a pantallas especiales como la de inicio, avisos o cualquier otra que se desee crear.

A.2.2 Variables HMI

Para hacer una pantalla dinámica y efectiva (fundamental para el control de un sistema o cualquier otra aplicación) se deben crear una serie de variables propias de la pantalla sobre las que se actuará posteriormente.

Cada variable tendrá una serie de campos fundamentales para la conexión de variables como se observa en la figura 8-7.



Nombre ▲	Tabla de variables	Tipo d...	Conexión	Nombre del PLC	Variable PLC	Direc...	Modo de acceso	Ciclo de adquisi..
Variable_prueba_HMI	Tabla de variables ..	Bool	HMI_Conexión_1	PLC_1	variable_prueba_PLC		<Acceso simbólico>	100 ms
Variable_prueba_HMI_1	Tabla de variables ..	Bool	HMI_Conexión_1	PLC_1	variable_prueba_PLC	%I0.0	<Acceso absoluto>	100 ms

Figura A-7. Campos Variables HMI

Cada variable de la pantalla estará ligada a una variable del PLC que a su vez estará vinculada a un PLC en concreto por medio de una dirección concreta como se observa en la figura de arriba. El tipo de la variable HMI coincide con el tipo de la variable asociada al PLC.

Por otra parte, la conexión podrá ser mediante HMI_conexion (como aparece en el ejemplo de arriba) o solo con variables internas del HMI (no tendrán vinculación con las variables del PLC).

Además, el modo de acceso absoluto permite acceder directamente a la dirección de la variable del PLC mientras que el acceso simbólico estará solo vinculado al nombre de la variable del PLC, lo que permitirá que siga ligada, aunque se modifique algún dato en la tabla de variables del PLC.

Finalmente, el ciclo de adquisición deberá ser lo menor posible (sin llegar a ser menor que el tiempo de ejecución de los bloques de programación) para que se pueda ver la pantalla en tiempo real.

A.2.3 Imágenes

Cada imagen que se cree tendrá una serie de objetos en su interior junto con unas propiedades que harán cambiar la visualización de la pantalla.

Las propiedades de la imagen junto con las de los objetos en su interior (se activan por click derecho/ propiedades) tendrán cuatro campos que se podrán modificar (o no) dependiendo del tipo de objeto. Estos campos serán:

Propiedades / Animaciones / Eventos / Textos

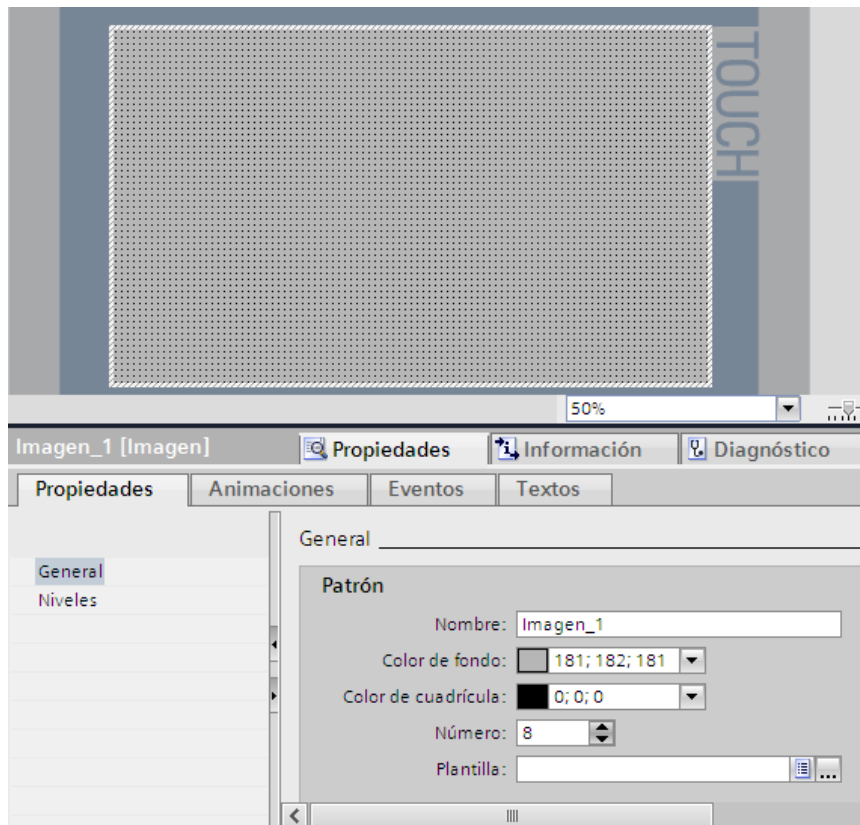


Figura A-8. Imagen

- Propiedades

En el apartado general, se cambia el nombre de la pantalla, el color de fondo y la plantilla asignada.

Además, se modificarán los niveles: Dentro de una imagen determinada, independientemente de si se trata de una imagen global o una plantilla de imagen, los objetos representados se pueden colocar dentro de “32 niveles”. Los objetos de un nivel con índice superior se colocan por delante de los objetos con índice inferior (siempre que sean visibles). Este apartado tratará la visibilidad de cada uno de los niveles de la imagen.

- Animaciones

En este campo se trata la visualización de la pantalla. En el caso de la imagen solo se podrá tratar la visibilidad asociada a las variables.

- Eventos

Al crear o borrar una imagen se puede modificar las variables que se requiera.

- Textos

En este último campo se pueden ver todos los textos que aparezcan en esa imagen estén asociados o no a un objeto.

Para establecer una imagen en concreto como la imagen inicial que aparezca en la pantalla, se accede a la ventana que abre esa imagen y se pulsa: botón derecho + Definir como imagen raíz. Esta condición solo la podrá tener

una sola imagen.

A.2.4 Herramientas

Se accede a ellas pulsando sobre la pestaña auxiliar que se encuentra arriba a la derecha, aunque suele encontrarse abierta por defecto. Esta opción está formada por un conjunto de objetos que estarán ligados a una de las pantallas con propiedades parecidas que se explicarán a continuación.

A.2.4.1 Objetos básicos

Este apartado se centra en las propiedades de los objetos que no muestran representaciones gráficas ni tienen eventos asociados a su pulsación. Es el caso de las formas geométricas, la línea, el visor de gráficos y el campo de texto.

FORMAS GEOMÉTRICAS

Se muestra un círculo en la figura 8-9, aunque las propiedades son análogas para la elipse y el rectángulo. Dentro de cada apartado de las propiedades se puede configurar lo siguiente:

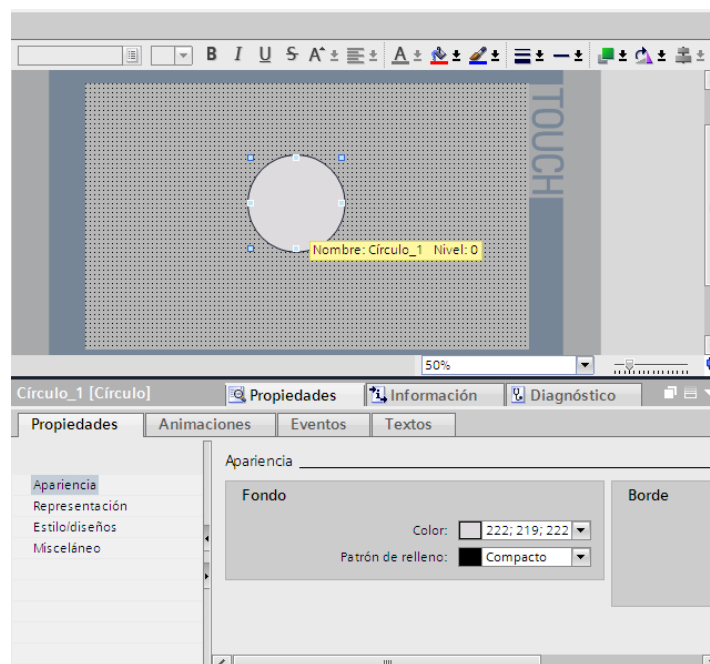


Figura A-9 Forma geométrica

- Propiedades

En el subapartado de apariencia se configura el color de fondo y del borde del objeto, en el de representación la posición absoluta de la forma en la pantalla (en ejes cartesianos), el siguiente subapartado se encarga del estilo propio del dibujo que en nuestro caso será por defecto 'WinCCDark'. Finalmente, el subapartado de misceláneo se centra en el nombre asignado al objeto y el nivel.

- Animaciones

En este caso el subapartado de visualización engloba a la apariencia y la visibilidad del objeto. Por su parte se pueden incluir movimientos de tipo directo, diagonal, vertical y horizontal. En ambos casos, las animaciones de las pantallas estarán vinculadas con las variables de la pantalla que estén asociadas.

Estas figuras no incluyen eventos ni textos.

LÍNEAS

Este tipo de objetos en TIA Portal están orientados como elementos de conexión gráfica para unir objetos, aunque pueden usarse en cualquier contexto.

Este objeto tiene las mismas características que el anterior en los apartados de propiedades y animaciones. En este caso, además añade dentro del apartado de Propiedades/Apariencia la opción de modificar los finales de línea según se quiera crear un segmento o una flecha.

VISOR DE GRÁFICOS

Este objeto permite añadir cualquier imagen al proyecto junto con todas las imágenes predeterminadas que trae TIA Portal. En la figura 8-10 se observa una imagen del programa con la apariencia de un interruptor (no tiene las propiedades de un interruptor, se explicarán a continuación)

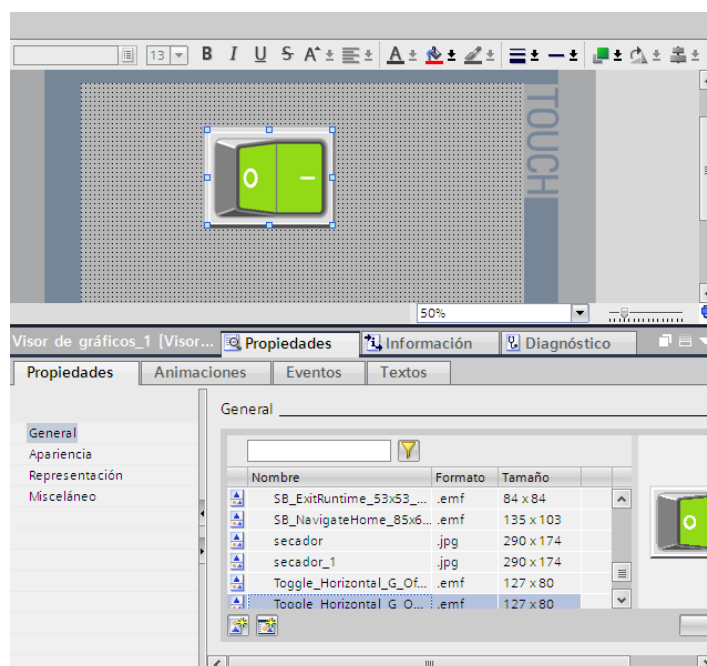


Figura A-10. Visor de gráficos

Tiene las mismas propiedades que el resto de los objetos básicos, aunque en el subapartado Propiedades/General incluye la elección de la imagen.

CAMPO DE TEXTO

A diferencia de los anteriores, se puede ver un objeto con el texto escrito por teclado que tendrá una serie de características principalmente asociadas al propio tipo de texto.

- Propiedades

A diferencia de los objetos básicos anteriores, en el subapartado general incluye la opción de modificar la propia cadena de texto, la fuente y la adaptación a un objeto (en el caso de que el texto esté agrupado con otro objeto), mientras que en el subapartado de apariencia se encuentran las características específicas de

anchura y estilo del texto. Además, el apartado de representación incluye la configuración de los márgenes.

Por otra parte, se incluye un subapartado de formato de texto para adecuar la orientación y alineación del mismo y otro de parpadeo.

A.2.4.2 Elementos

Las herramientas que se explicarán a continuación tienen un grado de interacción mayor que los del apartado anterior, debido a que además de las propiedades que comparten con los anteriores incluyen un campo de eventos que permiten modificar variables mediante pulsación, es decir, son los objetos que realmente representan la interfaz hombre-máquina.

BOTÓN

Es el clásico elemento que tiene dos estados ON/OFF y permite cambiar su estado por pulsación. Incluye las siguientes propiedades:

- Propiedades

En el apartado general se incluye el texto o gráfico que aparecen en su interior, el nivel en el que está el objeto y la creación de un acceso directo mediante una tecla.

Además, existe un apartado de patrón de relleno donde se incluyen los gradientes de color y otro de seguridad donde se permite pasar de modo de operación a monitorización o viceversa.

- Animaciones

Además de los apartados de visibilidad y apariencia que estarán ligados a los valores que tomen ciertas variables, se incluye un subapartado de conexión a una variable en concreto.

- Eventos

Los eventos que se realizan sobre un botón serán: Hacer click, pulsar y soltar. Sobre cada una de estas acciones se asocia una modificación de variables: Las más usadas serán:

Activar/Desactivar bit, Invertir bit, Activar imagen, incrementar o disminuir variable.

- Textos

Cada botón tendrá un texto asociado para cuando está OFF y otro cuando está ON.

INTERRUPTOR

Este elemento tendrá unas características muy semejantes a las del botón ya que comparte prácticamente todas las propiedades con el anterior, junto con sus estados y la mayoría de apartados sobre su apariencia. La diferencia fundamental reside en la variable asociada:

El interruptor tendrá una variable asociada que no tendrá por qué ser booleana, aunque sí lo serán sus estados ON/OFF. Ejemplo: el interruptor tendrá una variable asociada que varía de 0 a 5 y se considera apagado de 0 a 3 y encendido en los valores 4 y 5.

CAMPO ENTRADA/SALIDA

Este elemento consiste en un campo con formato numérico que estará asociado a una variable concreta. Este campo puede ser modificado mediante pulsación táctil y también por evolución de la variable del programa.

- Propiedades

Como novedades respecto a las propiedades del botón se incluye en el apartado general la posibilidad de asociar el campo e/s a una variable y se establece el modo (entrada, salida o entrada/salida). También se establece el formato de visualización y el de representación (ejemplo: tres cifras enteras y una decimal en formato decimal).

Se añade un apartado de comportamiento para la posibilidad de establecer el campo como una entrada oculta y otro apartado de límites para establecer la representación en caso de que se rebasen los mismos.

*Los apartados de animaciones y eventos coinciden con el del botón salvo que este campo no tiene estados ON/OFF como tal. Por otra parte, este campo tampoco tendrá ningún texto asociado.

CAMPO E/S SIMBÓLICO Y CAMPO E/S GRÁFICO

Para el funcionamiento adecuado de estos dos elementos se debe crear previamente una lista de textos y gráficos. Esta opción se encuentra en el árbol del proyecto: HMI_1/Listas de textos y gráficos

Estos dos elementos tienen el mismo funcionamiento y solo cambiará su forma de representación. Al igual que el campo E/S pueden establecerse su modo como entrada, salida o E/S, pero a diferencia del caso anterior, estos elementos están enfocados a asociarse a una variable interna de la pantalla y una lista de textos y gráficos (ambas asociaciones se realizarán en el apartado Propiedades/General) y su funcionamiento será el siguiente:

Las listas de textos estarán compuestas por una serie de entradas en las cuales tendrán un valor vinculado a un texto que será el que aparezca en el campo en cuestión. Análogamente las listas de gráficos tendrán cada valor vinculado a un gráfico como se observa en las figuras 8-11 y 8-12.

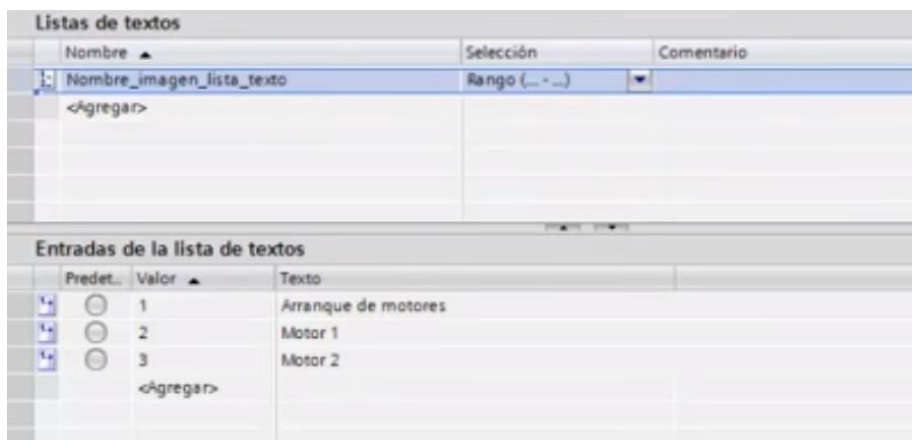


Figura A-11. Lista de textos

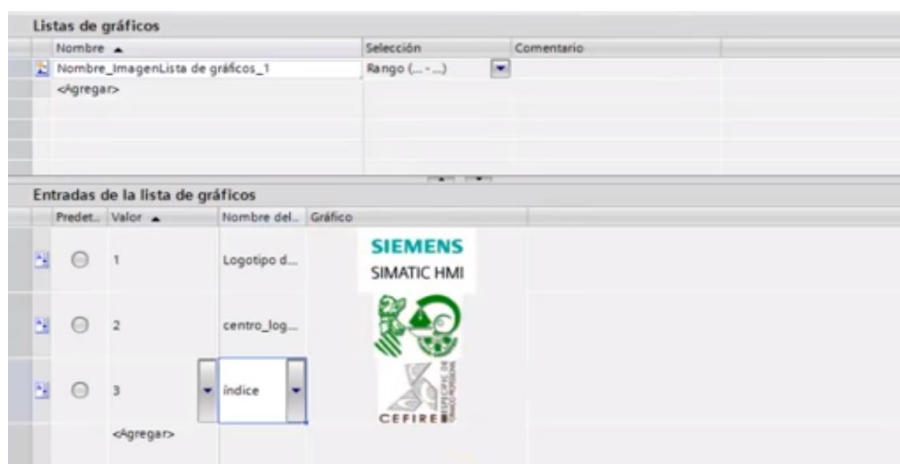


Figura A-12. Lista de gráficos

Una vez que el campo tiene una variable y una lista asociada (el campo simbólico usará la lista de textos y el campo gráfico usará la lista de gráficos), el campo en cuestión irá variando según la variable asociada tome uno u otro valor.

Estos campos serán especialmente útiles en las plantillas donde el título o el icono irá variando según lo que se quiera mostrar en cada imagen.

*Existe un elemento de campo de fecha y hora que a pesar de estar definido como elemento aparte como se observa en la figura 8-13, no es más que un campo e/s que muestra la fecha y/o la hora deseada por pantalla, incluida la del sistema.

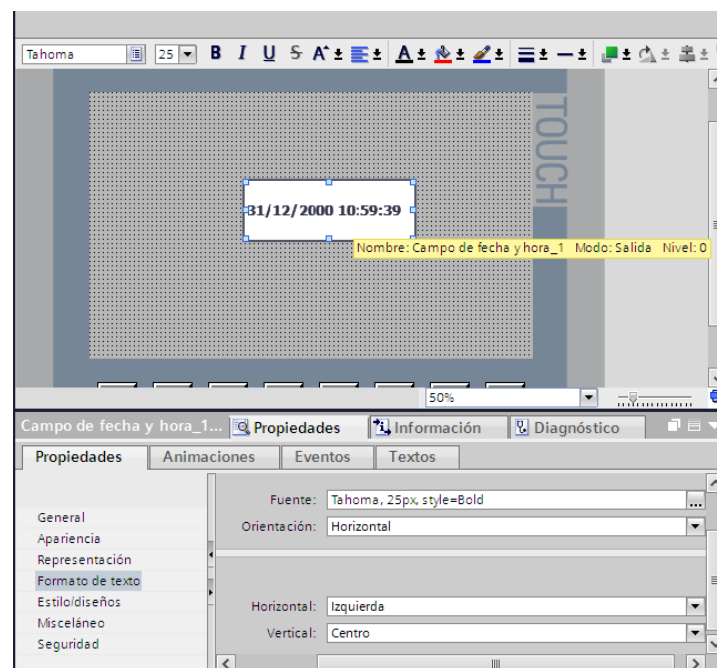


Figura A-13. Campo de fecha y hora

REPRESENTACIÓN DE BARRAS

Este elemento es un objeto de representación de una variable en un formato concreto de una barra, por lo que a diferencia de la mayoría de elementos no se podrán modificar variables pulsando sobre él, es decir, no tendrá eventos asociados.

- Propiedades

En el apartado general se asociará la variable que se representará en la barra junto con el valor máximo y mínimo de la escala mientras que en el de apariencia se definen los límites mediante líneas o marcas.

Se añade un apartado de tipo de borde, otro de escalas donde se configuran las divisiones y el escalado automático (opcional) y otro de título donde se establecen los valores auxiliares que se observan en la parte izquierda de la barra de la figura 8-14 y su formato.

- Animaciones

Este apartado tendrá los mismos subapartados que el resto de elementos, aunque el apartado de apariencia pierde interés debido a que la representación principal ya queda configurada por otros campos en el apartado de propiedades.

Este elemento no incluye eventos ni textos asociados.

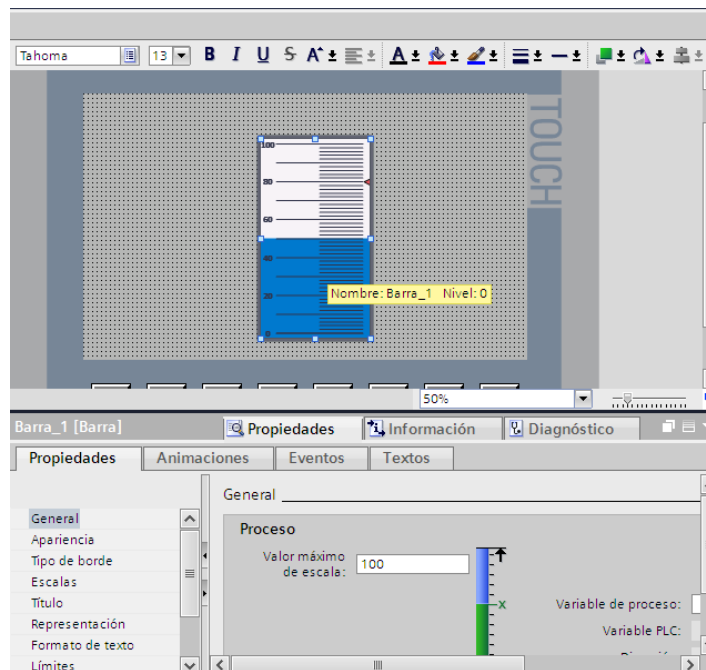


Figura A-14. Representación mediante barras

VISOR DE CURVAS

Este elemento aparecerá en las herramientas dentro del apartado de controles, aunque se ha decidido incluirlo en este apartado debido a que comparte muchas propiedades con el resto de elementos.

El visor de curvas es un elemento de representación muy completo de una o más variables en una gráfica en la que además se podrá tomar el valor exacto de las curvas en un punto concreto y mostrarlo de manera auxiliar.

- Propiedades

En el apartado de curva se establece el nombre, estilo, número de valores a representar, tipo de curva, variable asociada, lado del eje al que se asigna la curva y visualización de las líneas de límite.

En el apartado de apariencia se incluye la cuadrícula (opcional) y los colores de los ejes.

Existe un apartado de barra de herramientas (para ocultarla o no según convenga) y otro de borde y relleno de tabla para completar los colores de la posible cuadrícula. Por otra parte, se han incluido otros apartados de bordes y patrones de relleno.

Se añaden los típicos apartados de representación (para colocar la gráfica en la pantalla) y de formato de texto.

Finalmente, se añade un apartado de eje de tiempo donde se configura el incremento (constante o no), las marcas en el eje y el rango de duración. A su vez se incluyen los apartados de ejes de valores izquierdos y derecho donde se da la opción de activar el auto-escalado (adaptar automáticamente), establecer el principio y el fin del eje junto con una línea auxiliar y configurar los títulos, las marcas de los ejes y sus incrementos.

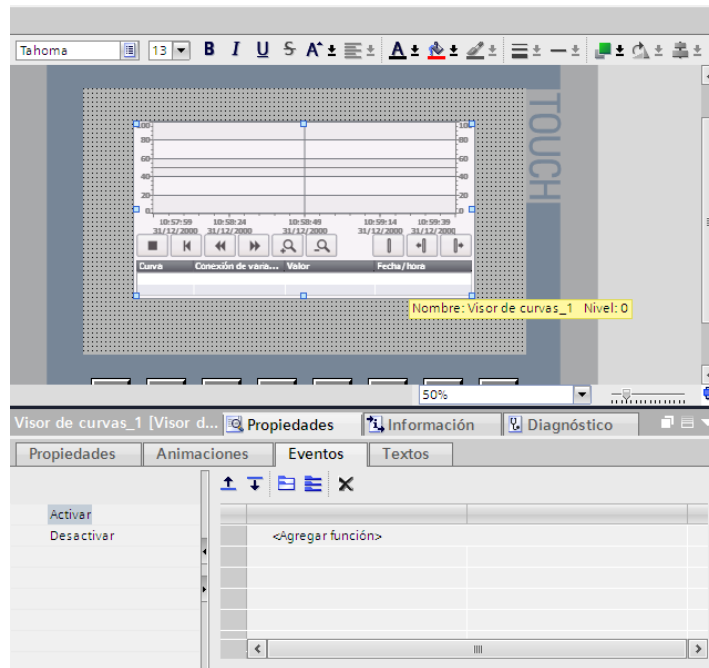


Figura A-15. Visor de curvas

A.2.5 Aspecto

Se tiene acceso a esta opción a través de la pestaña que aparece a la derecha de la vista del proyecto y es recomendable usarla cuando todas las herramientas han sido establecidas en sus respectivas imágenes.

Esta opción tendrá acceso a los niveles de todos los objetos de cada imagen del Proyecto y tendrá dos funciones fundamentales:

- Manejar la visibilidad de cada uno de los niveles.
- Modificar los niveles de los objetos según corresponda. Esta opción será útil cuando interesa invisibilizar ciertos niveles de cara a tener una visión más clara en la pantalla o cuando se superponen varios objetos dentro de la misma pantalla. Se recuerda que los objetos con un nivel más alto se superpondrán a los objetos con un nivel más bajo.

A.2.6 Librerías

Las librerías en el programa TIA Portal se utilizan para almacenar un conjunto de información relativa al PLC o a la pantalla: bloques de programación, configuración de dispositivos, tipo de datos, objetos de la pantalla e imágenes completas.

La ventana de librerías se encuentra en la parte derecha del proyecto en la esquina inferior tal y como se observa en la figura 8-16.

Existen dos tipos de librerías: las librerías del proyecto y las globales.

Las librerías del proyecto guardarán objetos que se almacenarán en otra parte del mismo proyecto como puede ser el caso de una función que se invoque desde varios bloques de organización o de un objeto que se utilice en varias imágenes. Estos objetos se guardarán en el apartado de librería del proyecto/Plantillas maestras.

Por su parte las librerías globales englobarán objetos que se podrán utilizar en otros proyectos. En la figura 8-16 se observan una serie de librerías globales que trae por defecto el programa. Al crear una librería global nueva se le asignará un directorio propio. Al igual que en la librería del proyecto, los objetos en cuestión se deben crear dentro del apartado de plantillas maestras



Figura A-16. Librerías en TIA Portal

Para mover un objeto a la librería correspondiente o para realizar el proceso contrario, basta con arrastrar el objeto de un sitio a otro o copiar y pegar.

Cualquier actualización que se realice en las librerías globales aparecerán en los nuevos proyectos que se utilicen una vez que se guarde el proyecto.