



UNIVERSIDAD POLITÉCNICA SALESIANA

SEDE GUAYAQUIL

**DISEÑO Y SIMULACIÓN DEL PROCESO DE CLASIFICACIÓN DE PRODUCTOS
TEXTILES MEDIANTE PLC, HMI E IOT**

Trabajo de titulación previo a la obtención del
Título de Ingeniero en Electrónica y Automatización

AUTORES: ÁNGEL JEREMY SEGURA GARCÍA

LEONARDO SAÚL TORRES MARQUÍNEZ

TUTOR: ING. RAFAEL CHRISTIAN FRANCO REINA, MSc.

GUAYAQUIL – ECUADOR

2025 – 2026

CERTIFICADO DE RESPONSABILIDAD Y AUTORÍA DEL TRABAJO DE TITULACIÓN

Yo, Leonardo Saúl Torres Marquínez con documento de identificación N°0929274074 y Ángel Jeremy Segura García con documento de identificación N°0952097483, manifestamos que:

Somos los autores y responsables del presente trabajo de titulación, y autorizamos a la Universidad Politécnica Salesiana a utilizarlo, difundirlo, reproducirlo o publicarlo, total o parcialmente, con fines exclusivamente académicos y sin ánimo de lucro.

Guayaquil, 31 de enero del año 2026.

Atentamente,



Leonardo Saúl Torres Marquínez

0929274074



Ángel Jeremy Segura García

0952097483

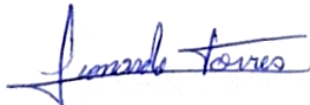
**CERTIFICADO DE CESIÓN DE DERECHOS DE AUTOR DEL TRABAJO DE
TITULACIÓN A LA UNIVERSIDAD POLITÉCNICA SALESIANA**

Nosotros, Leonardo Saúl Torres Marquínez con documento de identificación No. 0929274074 y Ángel Jeremy Segura García con documento de identificación No. 0952097483, expresamos nuestra voluntad y, por medio del presente documento, cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales, en virtud de que somos autores del Proyecto técnico: “Diseño y simulación del proceso de clasificación de productos textiles mediante PLC, HMI e IoT”, el cual ha sido desarrollado para optar por el título de: Ingeniero en Electrónica y Automatización, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En concordancia con lo manifestado, suscribimos este documento en el momento que hacemos la entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Guayaquil, 31 de enero del año 2026.

Atentamente,



Leonardo Saúl Torres Marquínez

0929274074



Ángel Jeremy Segura García

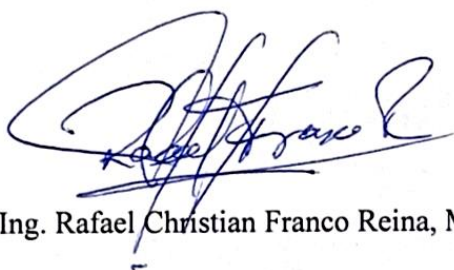
0952097483

CERTIFICADO DE DIRECCIÓN DEL TRABAJO DE TITULACIÓN

Yo, Rafael Christian Franco Reina con documento de identificación N°0923328629, docente de la Universidad Politécnica Salesiana, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: DISEÑO Y SIMULACIÓN DEL PROCESO DE CLASIFICACIÓN DE PRODUCTOS TEXTILES MEDIANTE PLC, HMI E IOT, realizado por Leonardo Saúl Torres Marquinez con documento de identificación N°0929274074 y Ángel Jeremy Segura García con documento de identificación N°0952097483, obteniendo como resultado final el trabajo de titulación bajo la opción de Proyecto Técnico que cumple con todos los requisitos determinados por la Universidad Politécnica Salesiana.

Guayaquil, 31 de enero del año 2026.

Atentamente,



Ing. Rafael Christian Franco Reina, MSc.

0923328629

DEDICATORIA

Así finaliza una etapa hermosa que marcará mi vida para siempre: la universidad.

Dedico este logro, en primer lugar, a mi madre, por el esfuerzo constante durante toda mi formación académica, por ser mi pilar y mi base, y por apoyarme siempre en mis ideas. A mi padre, que hizo todo lo posible, sin importar las dificultades, para que yo pudiera terminar mis estudios.

A mi enamorada, por su actitud siempre positiva, por mejorar mis días e impulsarme a seguir adelante. A mi familia, por creer en mí en todo momento, y a mis amigos, por ser parte fundamental de este largo proceso.

También dedico esta página a mi mascota, mi fiel compañero, que cada noche me esperaba en la puerta de casa, incluso cuando llegaba tarde.

Y, finalmente, me lo dedico a mí, por no rendirme, por mantenerme firme ante los obstáculos y por demostrarme que siempre fui capaz de llegar al final de este camino.

Leonardo Saúl Torres Marquínez

DEDICATORIA

Para mi madre, Jéssica, dedico este paso importante en mi vida profesional, porque fue una persona maravillosa que me enseñó los valores que me permitieron ser la persona que soy ahora. Aunque el destino no me permitió tenerte a mi lado, tu ausencia se convirtió en la fuerza que me impulsó a seguir adelante. Cada esfuerzo y cada logro llevan tu nombre, y aunque no estés conmigo físicamente, te tendré siempre en mi corazón.

A mi padre, por apoyarme incondicionalmente cuando necesitaba algo urgente, por las pocas veces que lo veía debido al trabajo, siempre preguntarme: “¿cómo vas en el trabajo?” y “¿qué tal vas en la universidad?”. Eso fue algo que me permitió desahogarme, contándole todo lo que me pasaba. Te dedico mi logro, padre.

A mis familiares les agradezco por estar siempre dándome ese plus y por felicitarme cada vez que los veía, eso me impulsaba a mejorar cada vez más. A mis amigos, que creyeron en mí y me brindaron su apoyo en todo momento.

Quisiera dedicar este logro a cada persona que estuvo conmigo a lo largo de este camino. Aunque no me alcance una hoja para nombrarlos a todos, cada palabra de aliento, cada apoyo sincero y cada momento compartido dejaron una huella imborrable en mi vida.

Ángel Jeremy Segura García

AGRADECIMIENTO

Agradezco a Dios por brindarme sabiduría y valentía al tomar decisiones y por escuchar cada una de mis peticiones a lo largo de este proceso.

A mi madre, por su sacrificio constante, ya que sin su apoyo nada de esto habría sido posible. A mi padre, por estar siempre pendiente de mi progreso y por acompañarme durante mi vida universitaria. A mi enamorada, por aprender conmigo durante este proceso y por ser luz de mis días, impulsándome a seguir adelante.

A mis amigos, gracias por las enseñanzas, el apoyo y los conocimientos que compartieron a lo largo de la carrera. A mis profesores, gracias por compartir su experiencia y por ayudar en mi formación académica y profesional.

De manera especial, agradezco a un amigo que hoy se encuentra lejos, pero en su momento me compartió muchos conocimientos que quizá no comprendí del todo y que hoy valoro profundamente.

Sin más que decir, llega a su fin una etapa increíble de mi vida, que me deja amistades, anécdotas y aprendizajes que llevaré siempre en el corazón.

Leonardo Saúl Torres Marquínez

AGRADECIMIENTO

En primer lugar, Agradezco a Dios por guiarme con fortaleza y claridad en cada decisión que tomaba, por acompañarme constantemente y sosteniéndome a lo largo de todo este camino.

A mi mamá, gracias por darme la fuerza para mantenerme de pie durante todo este camino, por amarme incondicionalmente y por ser alguien que, con cada sonrisa que me regalaba, alegraba mis días. Tu amor fue mi refugio en los momentos difíciles; ahora es un recuerdo y mi motivación para no rendirme. Todo lo que he logrado es gracias a ti.

A mi padre, gracias por tu constante apoyo, por tu esfuerzo para que pudiera estudiar y por ser quien más confiaba en mis capacidades, incluso cuando yo dudaba. A mi familia, estar siempre cuando necesite un consejo o un abrazo.

A mis amigos, aquellos que cuando me sentía solo llegaron para llenarme de momentos inolvidables para mí, cada broma, cada risa, lo mantendré siempre en mí. A ese grupito de siempre en la universidad que me ayudaba cuando desconocía algo de una materia, muchas gracias por acompañarme en este proceso y por enseñarme todo lo que sé ahora.

Quiero agradecerle también a alguien que estuvo conmigo desde que llegué a la universidad, quien me enseñó muchas cosas que desconocía en mis inicios.

Finalmente, agradezco a unas personas maravillosas que, a pesar de no ser familiares, me acogieron como si lo fuera, brindándome cariño y apoyo que llevaré siempre conmigo, demostrándome que la familia no siempre es de sangre, sino también de corazón.

Ángel Jeremy Segura García

RESUMEN

La clasificación de productos textiles es clave en la industria porque ayuda a organizar las prendas por tipo y mejora las siguientes etapas del proceso productivo. Sin embargo, muchas veces este proceso se realiza de forma manual o con pocos recursos, lo que dificulta su análisis y control.

El presente proyecto tiene como propósito el diseño y la simulación de un proceso automatizado para la clasificación de productos textiles, utilizando un PLC Siemens S7-1500 que contiene la programación en los lenguajes KOP y FUP, una HMI para la supervisión y manipulación del proceso de clasificación, y, además, una pasarela IoT2040 para la transmisión y visualización de datos en un dashboard dentro de NODE-RED.

El proceso simulado está conformado por un conjunto de bandas transportadoras encargadas del traslado de las prendas, un detector de tipo textil y actuadores responsables del direccionamiento de las prendas según su clasificación. Adicionalmente, se incorporó un subproceso de planchado en el que la temperatura se regula mediante un controlador PID, lo que permite evaluar el comportamiento de la consigna térmica.

Por último, la integración con Node-RED permite ver en tiempo real variables clave del proceso, como el conteo de prendas, el estado del almacenamiento y la temperatura. Esto ayuda a analizar el desempeño operativo y a validar los conceptos de automatización, control y conectividad industrial en el sector textil.

Palabras clave: Dashboard, PLC S7-1500, HMI, IoT2040, TIA Portal, Node-RED, Bandas transportadoras.

ABSTRACT

The classification of textile products is key in the industry because it helps organize garments by type and improves the subsequent stages of the production process. However, this process is often carried out manually or with limited resources, which makes analysis and control difficult.

The purpose of this project is to design and simulate an automated process for classifying textile products, using a Siemens S7-1500 PLC programmed in KOP and FUP languages, an HMI for supervising and manipulating the classification process, and an IoT2040 gateway for transmitting and displaying data on a dashboard within NODE-RED.

The simulated process consists of a set of conveyor belts responsible for transporting the garments, a textile detector, and actuators responsible for directing the garments according to their classification. Additionally, an ironing subprocess was incorporated in which the temperature is regulated by a PID controller, allowing the behavior of the thermal setpoint to be evaluated.

Finally, integration with Node-RED allows key process variables, such as garment count, storage status, and temperature, to be viewed in real time. This helps analyze operational performance and validate the concepts of automation, control, and industrial connectivity in the textile sector.

Keywords: Dashboard, PLC S7-1500, HMI, IoT2040, TIA Portal, Node-RED, Conveyor belts.

ÍNDICE DE CONTENIDO

I.	INTRODUCCIÓN	1
II.	PROBLEMA.....	2
III.	OBJETIVOS	5
3.1	Objetivo General.....	5
3.2	Objetivos específicos	5
IV.	FUNDAMENTO TEÓRICO	6
4.1.	PLC	6
4.1.1	PLC S7-1500.....	6
4.2	Pasarela Inteligente IOT2040	7
4.3	HMI.....	7
4.3.1	HMI KTP 700 Basic	8
4.4	Software TIA Portal.....	8
4.5	Node Red	9
4.6	Transmisión de datos a la nube.....	10
4.7	Bloques del TIA Portal	11
4.7.1	Bloques de Organización (OB).....	11
4.7.2	Bloques de Función (FB).....	12
4.7.3	Funciones (FC).....	12
4.7.4	Bloques de Datos (DB).....	12
4.8	PLCSIM	12
4.9	Controlador PID.....	13
4.10	Librería LSim.....	14
4.11	Recetas en TIA Portal	14
4.12	Protocolo Profinet	15
V.	MARCO METODOLÓGICO.....	16
5.1	Inicialización del proyecto en el entorno TIA Portal.....	17
5.2	Bloques para la programación del PLC	17
5.3	Definición de variables para el sistema de control	18
5.4	Programación KOP del Main.....	21
5.5	Programación en KOP y FUP para la clasificación aleatoria de prendas	40

5.6	Programación del controlador PID para la regulación térmica de la plancha...	41
5.6.1	Parametrización del Bloque PID Compact	42
5.6.2	Configuración de la librería LSim	45
5.6.3	Ajuste y escalado de señales en el lazo de control PID	46
5.7	Diseño y estructura de la Interfaz Hombre-Máquina (HMI)	47
5.7.1	Diseño y estructura de la pantalla “PRESENTACIÓN”	50
5.7.2	Diseño de la interfaz de “USER ADMINISTRATION”	52
5.7.3	Diseño y estructura de la interfaz “MENÚ”	54
5.7.4	Diseño y estructura de la pantalla “PROCESO 1”	56
5.7.5	Diseño y estructura de la interfaz “PROCESO 2”	59
5.7.6	Diseño y estructura de la pantalla “CONTADORES”	61
5.7.7	Diseño y estructura de la pantalla “PLANCHA”	63
5.7.8	Diseño y estructura de la interfaz “RECETAS PREDEFINIDAS”	65
5.7.9	Diseño y estructura de la pantalla “PID”	66
5.8	Desarrollo y configuración del Dashboard en Node-Red	68
5.9	Prototipo esquemático.....	75
VI.	RESULTADOS.....	77
6.1	Ejecución del proyecto en el entorno TIA Portal y configuración de la comunicación con la IoT2040.....	77
6.2	Habilitación de la interfaz mediante Login.....	79
6.3	Proceso de clasificación de textiles.....	80
6.4	Proceso de direccionamiento de textiles.....	82
6.5	Ejecución del proceso de planchado	83
6.6	Visualización del control PID.....	85
6.6.1	Pruebas simuladas de tiempos de estabilización del PID	86
6.7	Monitoreo del sistema en el dashboard mediante IoT2040	88
6.8	Visualización de la puesta en marcha del proyecto en módulo didáctico.....	89
VII.	CRONOGRAMA.....	91
VIII.	PRESUPUESTO	92
IX.	CONCLUSIONES	93
X.	RECOMENDACIONES.....	94

XI. REFERENCIAS..... 95

ÍNDICE DE FIGURAS

Figura 1	<i>Controlador lógico programable Siemens S7-1500</i>	6
Figura 2	<i>Dispositivo Siemens IoT2040</i>	7
Figura 3	<i>Pantalla Siemens SIMATIC HMI</i>	8
Figura 4	<i>Vista inicial del entorno de trabajo en TIA Portal</i>	9
Figura 5	<i>Esquema de nodos en Node-RED</i>	10
Figura 6	<i>Vista inicial del entorno de trabajo en TIA Portal</i>	10
Figura 7	<i>Ventana de creación de nuevos bloques en TIA Portal</i>	11
Figura 8	<i>Interfaz de simulación PLCSIM en TIA Portal</i>	13
Figura 9	<i>Esquema del controlador PID</i>	14
Figura 10	<i>Gestión de recetas en TIA Portal</i>	15
Figura 11	<i>Selección de módulos de entradas y salidas analógicas y digitales para el PLC</i>	17
Figura 12	<i>Bloques de programación</i>	18
Figura 13	<i>Sección 1 de variables de “GLOBAL_MEMORY”</i>	19
Figura 14	<i>Sección 2 de las variables pertenecientes al “GLOBAL_MEMORY”</i>	20
Figura 15	<i>Sección 3 de las variables del “GLOBAL_MEMORY”</i>	21
Figura 16	<i>Primer segmento de Networks del bloque Main</i>	22
Figura 17	<i>Segundo segmento de Networks del bloque Main</i>	22
Figura 18	<i>Iniciación del proceso</i>	23
Figura 19	<i>Segunda parte del Network 1</i>	24
Figura 20	<i>Activación del medidor de almacenamiento de ropa</i>	24
Figura 21	<i>Nivel y tiempo de almacenamiento de la ropa</i>	25
Figura 22	<i>Habilitación de la primera banda transportadora</i>	26
Figura 23	<i>Movimiento de la primera banda transportadora</i>	26
Figura 24	<i>Registro de objetos</i>	27
Figura 25	<i>Habilitación de la segunda banda Transportadora</i>	28
Figura 26	<i>Tipo de ropa aleatoria</i>	29
Figura 27	<i>Asignación del dato aleatorio a la ropa</i>	29
Figura 28	<i>Movimiento de la ropa superior</i>	30
Figura 29	<i>Movimiento de la ropa inferior</i>	31
Figura 30	<i>Movimiento de los calcetines</i>	32
Figura 31	<i>Accionamiento del actuador de la ropa superior</i>	33
Figura 32	<i>Activación del actuador de la ropa inferior</i>	33
Figura 33	<i>Accionamiento y conteo del actuador de los calcetines</i>	34
Figura 34	<i>Control de las bandas transportadoras en función del estado del sistema y el desplazamiento</i>	35
Figura 35	<i>Activación y control del desplazamiento de la ropa superior clasificada</i>	36
Figura 36	<i>Activación y control del desplazamiento de la ropa inferior clasificada</i>	37
Figura 37	<i>Activación y control del desplazamiento de los calcetines clasificados</i>	37
Figura 38	<i>Activación de la plancha</i>	38

Figura 39	<i>Final de carrera del proceso de planchado</i>	39
Figura 40	<i>Reinicio del proceso</i>	39
Figura 41	<i>Generación de valor aleatorio para la clasificación</i>	40
Figura 42	<i>Modelo de la función de transferencia</i>	41
Figura 43	<i>Modelo matemático para la regulación térmica de la plancha</i>	41
Figura 44	<i>Transferencia de Variables de Receta al Bloque de Función</i>	43
Figura 45	<i>Configuración del controlador PID_Compact</i>	44
Figura 46	<i>Arquitectura del bloque PID con su asignación de registros y memorias</i>	45
Figura 47	<i>Configuración de las variables en el bloque LSim_PT1</i>	46
Figura 48	<i>Ajuste del rango de la señal PID</i>	47
Figura 49	<i>Primera sección de variables para el HMI</i>	47
Figura 50	<i>Segunda sección de variables del HMI</i>	48
Figura 51	<i>Pantallas diseñadas para mostrar el proceso</i>	48
Figura 52	<i>Administrador de usuarios</i>	49
Figura 53	<i>Panel de objetos para editar el HMI</i>	49
Figura 54	<i>Configuración del botón de menú</i>	50
Figura 55	<i>Configuración de seguridad del botón en la interfaz HMI</i>	51
Figura 56	<i>Ventana de inicio de sesión del sistema HMI</i>	51
Figura 57	<i>Pantalla de presentación del proyecto</i>	52
Figura 58	<i>Visualización del estado de usuario en la plantilla del HMI</i>	53
Figura 59	<i>Configuración del botón de inicio de sesión en la plantilla del HMI</i>	53
Figura 60	<i>Pantalla de administrador de usuarios</i>	54
Figura 61	<i>Configuración del botón de navegación hacia la pantalla "PROCESO"</i>	55
Figura 62	<i>Pantalla del menú del proceso</i>	55
Figura 63	<i>Configuración del botón de RESET en la pantalla de proceso</i>	56
Figura 64	<i>Configuración del indicador visual de estado en la pantalla de proceso</i>	56
Figura 65	<i>Configuración del indicador de nivel de la canasta de prendas</i>	57
Figura 66	<i>Configuración del indicador de valores aleatorios del proceso</i>	57
Figura 67	<i>Pantalla de detección de la prenda</i>	58
Figura 68	<i>Indicador de conteo de prendas de ropa superior</i>	59
Figura 69	<i>Configuración de los parámetros de visualización del indicador de conteo</i>	59
Figura 70	<i>Configuración del botón de navegación hacia Proceso 1</i>	60
Figura 71	<i>Pantalla de la clasificación de la ropa</i>	61
Figura 72	<i>Configuración del indicador de conteo de ropa inferior</i>	62
Figura 73	<i>Pantalla de contadores de las prendas</i>	63
Figura 74	<i>Configuración del indicador de nivel de temperatura de la plancha</i>	63
Figura 75	<i>Configuración del indicador numérico de conteo de prendas planchadas</i>	64
Figura 76	<i>Interfaz del proceso de planchado</i>	65
Figura 77	<i>Configuración interna de las recetas predefinidas en el HMI</i>	65
Figura 78	<i>Pantalla de recetas para el planchado</i>	66

Figura 79 Configuración del trend de la temperatura real "ACTUALTEMP"	67
Figura 80 Configuración del trend del valor de referencia del PID "SETPOINT"	67
Figura 81 Pantalla para la visualización del control PID	68
Figura 82 Data block que almacena las variables para el Node-red	69
Figura 83 Inicio del Node-Red mediante comandos en el CMD	70
Figura 84 Verificación de conectividad mediante comando ping.....	70
Figura 85 Librerías a instalar para la visualización de los datos	71
Figura 86 Configuración de nodos para verificación y visualización de datos en Node-RED.....	72
Figura 87 Configuración del nodo S7 in para lectura de variables desde el PLC.....	73
Figura 88 Dashboard de supervisión para clasificación de textiles y control de la plancha.....	75
Figura 89 Vista general del diseño del proyecto	76
Figura 90 Compilación del programa del PLC en TIA Portal	77
Figura 91 Compilación del proyecto HMI en TIA Portal	78
Figura 92 Verificación de la comunicación con la IOT2040.....	78
Figura 93 Inicio de sesión de usuario	79
Figura 94 Diagrama del proceso de inicio de sesión	80
Figura 95 Interfaz HMI del proceso de clasificación de textiles	81
Figura 96 Diagrama del proceso de clasificación de textiles	81
Figura 97 Interfaz HMI del proceso de direccionamiento de textiles.....	82
Figura 98 Diagrama de flujo del proceso de direccionamiento de textiles	83
Figura 99 Interfaz HMI de la plancha encendida	84
Figura 100 Diagrama de flujo de la plancha.....	84
Figura 101 Interfaz HMI del control PID	85
Figura 102 Visualización del comportamiento del control PID durante el planchado del pantalón de algodón	86
Figura 103 Visualización del comportamiento del control PID en la receta OFF	87
Figura 104 Dashboard de la IOT2040.....	89
Figura 105 Vista del proyecto cargado en el PLC y HMI en el laboratorio de Automatización II.....	89
Figura 106 Vista de los datos que envía el PLC al dashboard del Node-Red	90
Figura 107 Cronograma de actividades	91
Figura 108 Actividades a realizar entre diciembre y enero.....	91

ÍNDICE DE TABLAS

Tabla 1 <i>Asignación de variables y direcciones de Data Blocks para comunicación PLC–IOT2040</i>	74
Tabla 2 <i>Tiempos de estabilización del control PID</i>	88
Tabla 3 <i>Tabla de presupuesto correspondiente al proyecto de titulación</i>	92

I. INTRODUCCIÓN

La industria textil es considerada uno de los sectores con mayor relevancia a nivel mundial, no solo por su volumen de producción, sino también por su impacto social y ambiental. El crecimiento sostenido del consumo de productos textiles incrementa la necesidad de procesos más organizados en etapas como la clasificación de prendas, las cuales suelen realizarse de forma manual. Esto puede generar inconsistencias operativas y frena la adopción de tecnologías propias del entorno industrial moderno.

El avance de la automatización ha permitido la implementación de herramientas tales como PLC, HMI y los módulos IoT, las cuales facilitan la representación, visualización y análisis de procesos industriales. Estas tecnologías no solo son utilizadas en el ámbito productivo, sino que también representan un recurso clave dentro de la formación académica, al permitir que los estudiantes pongan en práctica conocimientos teóricos en escenarios cercanos a la realidad industrial mediante simulaciones.

El presente Trabajo de Titulación tiene como propósito principal diseñar y simular un proceso de clasificación de productos textiles mediante el uso de PLC, HMI y un módulo IoT. Para lograr este objetivo, se abordaron aspectos como la programación de la lógica de control en el PLC, el diseño de una interfaz HMI para la visualización de las distintas etapas del proceso y la integración del módulo IoT para la presentación de la información generada durante la simulación.

II. PROBLEMA

La industria Textil, a nivel global, representa una de las que incluye mayor tamaño con relación a la maquinaria global, y la misma cifra de negocios y de facturación se acerca a los 920 000 millones de dólares, lo que significa una participación del 2 % sobre el PIB global. Este sector da empleo a cerca de 60 millones de personas, pero tiene que lidiar con graves problemas ecológicos: la Textil emite el 10 % de todas las emisiones de carbono globales, y entre el 60 % y el 70 % de los productos textiles terminan en los vertederos o incinerados. Además, con respecto a la producción de una camiseta hecha de algodón, se han llegado a estimar que se necesitan alrededor de 2 700 litros de agua para su confección (Lindner, 2025).

Considerado uno de los dos sectores más contaminantes del planeta a causa de los procedimientos utilizados para cultivar y producir, teñir y manufacturar, lo cual supone un gran consumo de agua, energía y productos químicos. La manufactura tradicional y la producción industrial en gran escala de materiales sintéticos provocan una gran carga medioambiental y continúan incrementándose con el consumo a nivel global. Con esta situación, han empezado a realizarse diferentes tipos de estudios que propongan alternativas sostenibles, como utilizar fibras naturales poco contaminantes (Torrens Oa, 2024).

La industria textil-confección tiene un impacto ambiental negativo, y en el caso de Colombia, se trata de unas grandes cantidades de aguas residuales mal gestionadas con compuestos tóxicos, específicamente en procesos de teñido, lavado y acabado. De acuerdo a Naciones Unidas, este sector es el responsable del 20% de las aguas contaminadas en el planeta y el problema de la gestión de los efluentes en Colombia agrava este impacto. A

través de una revisión de la bibliografía se han documentado procesos de tratamientos aplicables para aguas residuales, las cuales generan contaminantes en cada uno de los procesos, lo que pone de manifiesto la necesidad de adoptar con urgencia tecnologías con prácticas de gestión ambiental en las industrias textiles y de confección de esta región (Quintero Sánchez, 2023).

La industria textil en Ecuador ha sido un pilar de la identidad cultural y de la economía, desde los tejidos ancestrales precolombinos hasta el plano de la producción industrial actual, pero ha de superar retos importantes relativos a la competitividad, la sostenibilidad y la escasa inversión en innovación tecnológica. Los aportes que ofrece para la creación de empleo y para la economía nacional son innegables, pero buena parte de las empresas textiles de Ecuador cuentan con un nivel de infraestructuras altamente limitado y escasa inversión en el proceso de producción sostenible que contrasta en cierta manera con el resto de la producción internacional (Sarango, 2024).

En la Universidad Politécnica Salesiana, dentro de la carrera de Ingeniería en Electrónica y Automatización, se cuenta con equipos tecnológicos como PLC, HMI y módulos IoT que fortalecen el desarrollo de prácticas en el área de automatización industrial. Sin embargo, no se dispone de una propuesta académica orientada al diseño y simulación de un proceso de clasificación de productos textiles, lo que limitaba la aplicación práctica de los conocimientos adquiridos en las asignaturas de Automatización Industrial I y II. Por ello, se planteó el desarrollo de un proyecto que aprovechó los recursos disponibles en la institución para representar de manera virtual el control y el monitoreo remoto del proceso de clasificación textil.

Dentro del marco de la formación académica que tiene la Universidad Politécnica Salesiana, se realizó una propuesta orientada a fomentar la práctica de los estudiantes de la carrera de Electrónica y Automatización. El objetivo fue aplicar conocimientos relacionados con la programación y la transferencia de datos en procesos de clasificación textil dentro de un contexto de simulación realista. En este proyecto, el PLC permitió realizar la lógica de control del proceso, mientras que el HMI proporcionó la visualización de las variables, y el módulo IoT permitió disponer de un sistema de monitoreo remoto de la información obtenida durante la operación.

La ejecución del diseño mediante simulación utilizó recursos técnicos existentes dentro de la institución, lo que permitió representar un proceso industrial sin necesidad de infraestructura física adicional. A su vez, esta propuesta enriqueció la enseñanza práctica, promoviendo el desarrollo de destrezas técnicas y el uso de herramientas modernas de la automatización.

III. OBJETIVOS

3.1 Objetivo General

Diseñar y simular un proceso de clasificación de productos textiles mediante PLC, HMI e IoT.

3.2 Objetivos específicos

- Programar el proceso de clasificación textil en un entorno simulado mediante PLC.
- Desarrollar una interfaz HMI que permita visualizar y simular el control del proceso en tiempo real.
- Implementar la propuesta en el módulo de automatización de la Universidad Politécnica Salesiana.
- Integrar el sistema con un módulo IoT para el envío de datos a una plataforma de monitoreo remote en la nube.

IV. FUNDAMENTO TEÓRICO

4.1. PLC

Los controladores lógicos programables (PLC) están diseñados para la automatización de tareas industriales por medio de la lectura de señales, procesamiento lógico y la activación de actuadores. Su estructura integra módulos de entrada/salida, memoria y el procesador que ejecuta la programación para cada aplicación. Con el paso del tiempo, han evolucionado desde los dispositivos sencillos hasta modelos con mayores capacidades de comunicación y adaptabilidad mejoradas (Saldias, 2025).

4.1.1 PLC S7-1500

En la figura 1, se muestra un controlador lógico programable de alto rendimiento diseñado para aplicaciones industriales que requieren precisión, velocidad y confiabilidad. Su arquitectura modular procesa grandes cantidades de datos en un breve tiempo, facilitando el control de procesos complejos. Ofrece una comunicación rápida y segura mediante PROFINET y está integrado en el entorno de programación TIA Portal, simplificando la configuración, diagnóstico y la monitorización. El modelo destaca por la capacidad de adaptarse a diversos sectores industriales (Electrobeyco, 2024).

Figura 1
Controlador lógico programable Siemens S7-1500



Nota. La figura corresponde al PLC S7-1500 para el control del proceso automatizado (RS, 2025).

4.2 Pasarela Inteligente IOT2040

La pasarela SIMATIC IOT2040 está destinada a conectar dispositivos, recolectar datos y enviarlos a plataformas superiores o a la nube. Cuenta con un procesador Intel Quark, 1 GB de RAM y está diseñada para operar de forma continua 24/7. Dispone de varias opciones de conectividad como doble Ethernet, ranura Arduino, mini PCIe, USB y puertos RS232/485, lo que permite integrar sensores y equipos externos en aplicaciones de monitoreo y automatización (SIOS, 2025), ver figura 2.

Figura 2
Dispositivo Siemens IoT2040



Nota. Módulo Siemens IoT2040 para la integración del sistema con la red de monitoreo (Tecnología, 2025).

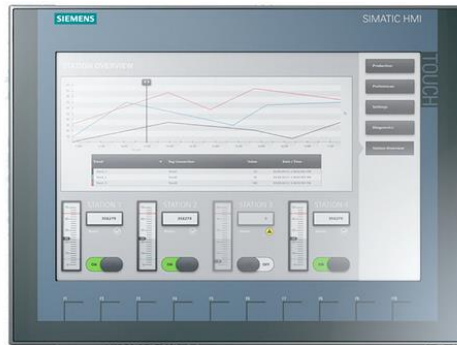
4.3 HMI

La interfaz hombre-máquina (HMI) es una pantalla o panel que permite a los operarios visualizar y controlar procesos industriales mediante información mostrada de forma gráfica y comprensible. A través de su software integrado, la HMI transforma datos técnicos en elementos visuales como indicadores, diagramas o alarmas, facilitando lo que es el seguimiento del estado de equipos y variables al instante (AVEVA, 2025).

4.3.1 HMI KTP 700 Basic

El SIMATIC KTP700 Basic, Figura 3, es un panel HMI de 7" con pantalla TFT a color y resolución 800×480 pixeles diseñado para entornos industriales. Tiene conectividad Ethernet/PROFINET, 10 MB de memoria interna y un procesador ARM, permite operar funciones básicas como alarmas, recetas y gráficos. Su grado de protección IP65 y compatibilidad con TIA Portal lo convierten en la opción más práctica para el control y monitoreo de aplicaciones sencillas (KTP700, 2025).

Figura 3
Pantalla Siemens SIMATIC HMI



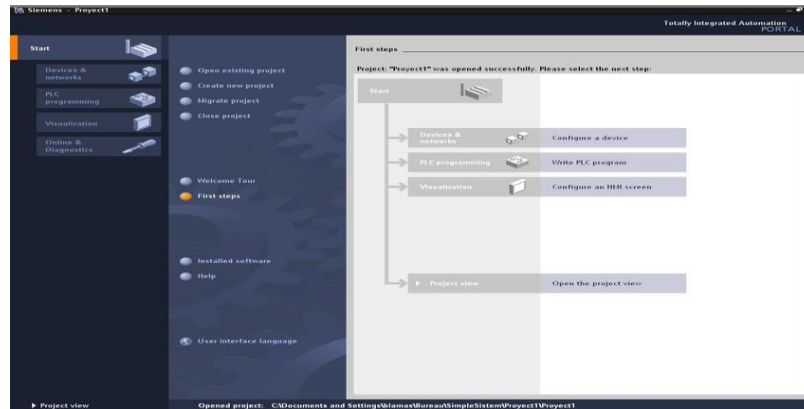
Nota. HMI utilizada para la visualización del proceso automatizado (Distrelec, 2025).

4.4 Software TIA Portal

El TIA Portal es una plataforma desarrollada por Siemens que unifica un solo entorno las herramientas necesarias para programar en el PLC, configurar un HMI y trabajar con variadores y servomotores. Su estructura integra módulos como STEP 7 para la lógica de control, WinCC para la creación de interfaces gráficas y Startdrive para la puesta en marcha de accionamientos. Su ventaja principal es que permite desarrollar proyectos de automatización sin depender de programas externos, facilitando que el trabajo sea más ordenado y eficiente. Aunque su uso inicial puede resultar poco intuitivo, una vez

dominado se convierte en una herramienta muy versátil y ampliamente utilizada en la industria (Domínguez A. , 2025), ver figura 4.

Figura 4
Vista inicial del entorno de trabajo en TIA Portal

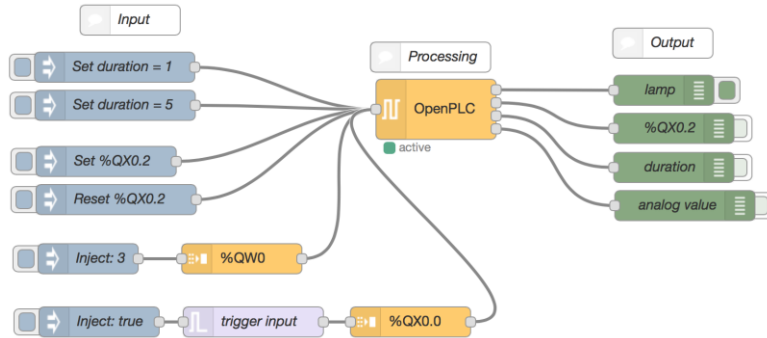


Nota. Interfaz de inicio de TIA Portal utilizada para la creación del proyecto (Lahmadi, 2025).

4.5 Node Red

La plataforma Node-RED permite crear flujos de automatización mediante nodos conectados entre sí, lo que facilita la integración de los dispositivos y servicios dentro de los proyectos IoT. Su entorno de trabajo, basado en arrastrar y soltar, hace posible que sin necesidad de conocimientos avanzados en la programación se pueda diseñar aplicaciones. Gracias a su base en Node.js, también admite la creación de nodos personalizados, lo que amplía su alcance en diferentes áreas como la domótica, la industria o la agricultura. Su flexibilidad y facilidad de su uso lo han convertido en una alternativa muy adoptada para desarrollar soluciones conectadas y escalables (Badajoz, 2024), ver figura 5.

Figura 5
Esquema de nodos en Node-RED

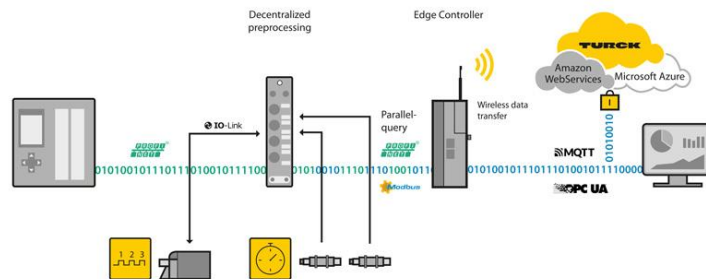


Nota. Diagrama de flujo en Node-RED integrado con OpenPLC (node-red-contrib-openplc, 2025).

4.6 Transmisión de datos a la nube

Como se observa en la Figura 6, la transmisión de datos hacia la nube permite mover información entre distintos dispositivos y plataformas de forma segura, aprovechando servicios alojados en los entornos públicos, privados o híbridos. Estas soluciones suelen incorporar mecanismos de cifrado y protocolos especializados para poder garantizar la protección de datos durante su envío y almacenamiento (FORTRA, 2025).

Figura 6
Vista inicial del entorno de trabajo en TIA Portal

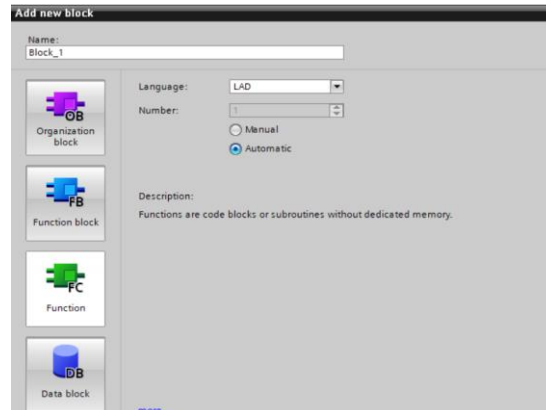


Nota. Estructura de comunicación entre los dispositivos de campo (Turk, 2025).

4.7 Bloques del TIA Portal

El entorno de programación TIA Portal define los bloques como herramientas para organizar y estructurar el código del PLC. Estos bloques dividen programas extensos en segmentos más cortos, legibles y comprensibles, lo que facilita el desarrollo, la lectura y el análisis. Además, los bloques permiten separar funciones, datos y secuencias de ejecución, lo que contribuye a una programación más ordenada y coherente (tecnopl, 2023), ver figura 7.

Figura 7
Ventana de creación de nuevos bloques en TIA Portal



Nota. Interfaz de TIA Portal utilizada para la creación y configuración de bloques del PLC.

4.7.1 Bloques de Organización (OB)

Los bloques de organización establecen la secuencia de ejecución del programa en el PLC. El bloque principal coordina la ejecución cíclica y actúa como punto de inicio para la llamada a otros bloques. Según los requisitos del proyecto, los bloques de organización pueden gestionar eventos específicos, como el arranque del controlador, la detección de errores o la ejecución periódica de tareas (Autracen, 2022).

4.7.2 Bloques de Función (FB)

Los bloques de función permiten desarrollar rutinas que requieren conservar información propia durante su ejecución. Cada bloque utiliza un espacio de memoria dedicado para almacenar sus variables internas, lo que posibilita su reutilización en distintas partes del programa (System, 2025).

4.7.3 Funciones (FC)

Las funciones se emplean para ejecutar tareas específicas dentro del programa del PLC. A diferencia de los bloques de función, no almacenan información de manera permanente, por lo que su uso se orienta a operaciones concretas, como cálculos, procesamiento de señales o control de acciones (Domínguez, 2025).

4.7.4 Bloques de Datos (DB)

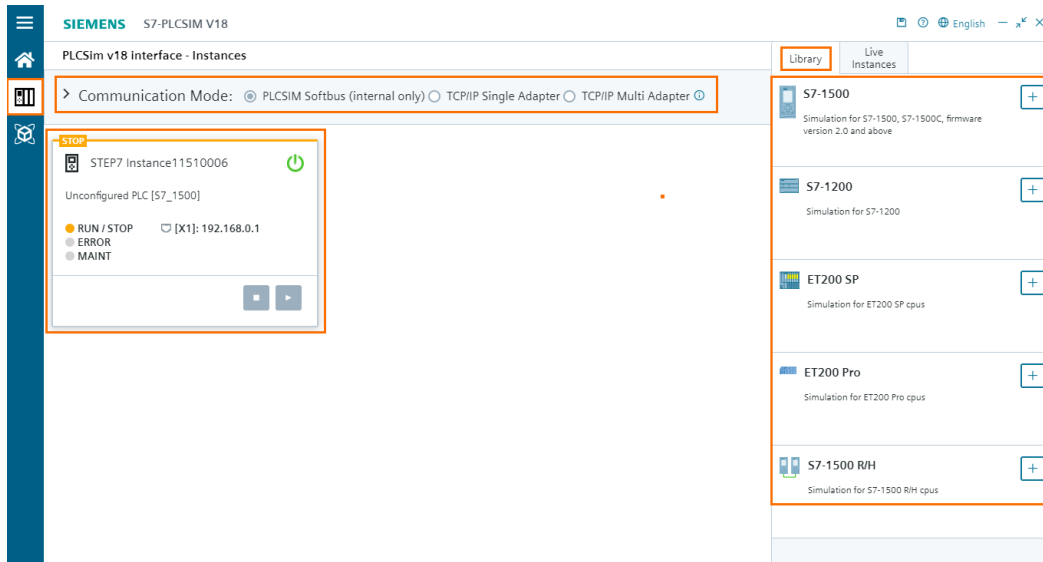
Los bloques de datos constituyen áreas de memoria destinadas a almacenar la información utilizada por el programa. Estos bloques pueden contener valores accesibles desde distintos bloques o datos específicos de un bloque de función. Su utilización centraliza la información del proceso y mejora la interacción entre las diversas partes del programa (SiePortal, 2025).

4.8 PLCSIM

Como se muestra en la figura 8, PLCSIM es una herramienta de simulación que permite probar y validar la comunicación con dispositivos HMI sin necesidad de utilizar un controlador físico. Se pueden modificar parámetros del proceso, observar la respuesta del sistema, verificar la lectura y la escritura de variables, así como analizar las reacciones a eventos como alarmas de hardware, errores de diagnóstico o cambios de configuración.

La comunicación con el PLC virtual se establece automáticamente al iniciar la simulación, siempre que se cuente con la licencia adecuada y se cumplan los requisitos del software (System T. P., 2025).

Figura 8
Interfaz de simulación PLCSIM en TIA Portal

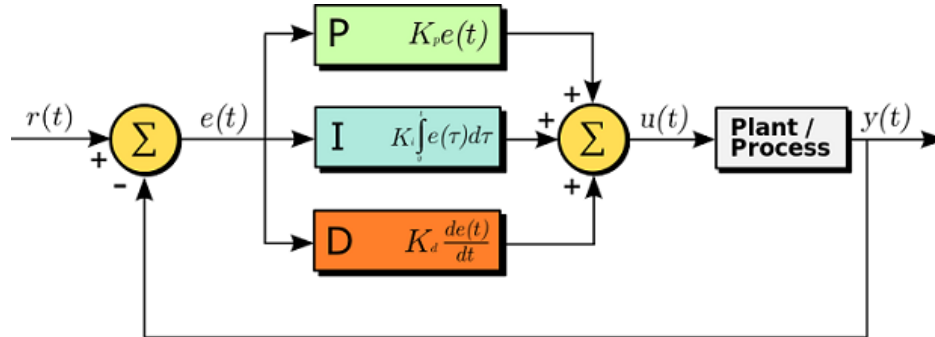


Nota. Entorno PLCSIM utilizado para la simulación del controlador y la verificación de la interacción con el HMI (Solis, 2026).

4.9 Controlador PID

El controlador PID es un algoritmo ampliamente utilizado en la industria para regular las variables de proceso en un sistema de control de lazo cerrado. Su funcionamiento se basa en comparar el valor medido con el valor de referencia y generar una señal de control mediante acciones proporcionales, integrales y derivativas. Esta combinación permite corregir errores actuales, acumulados y futuros, garantizando la estabilidad del proceso ante perturbaciones (NI, 2025), ver figura 9.

Figura 9
Esquema del controlador PID



Nota. Representación del funcionamiento del controlador PID con acciones proporcional, integral y derivativa aplicadas al proceso (Codesys, 2025).

4.10 Librería LSim

La librería LSim consiste en bloques diseñados para la simulación de procesos tecnológicos en los controladores SIMATIC S7-1200 y S7-1500. Esto permite analizar el comportamiento de los lazos de control en un entorno virtual. La integración con el bloque PID_Compact permite desarrollar esquemas de control en lazo cerrado directamente en el PLC, sin necesidad de un proceso físico. Esta herramienta es adecuada para la validación, el ajuste y la puesta en marcha de reguladores PID (Siemens I. , 2025).

4.11 Recetas en TIA Portal

Las recetas en TIA Portal, a través de WinCC, figura 10, permiten administrar conjuntos de parámetros que facilitan la adaptación de procesos o equipos a diferentes variantes de operación sin modificar la lógica principal de control. Estas recetas se utilizan ampliamente en entornos industriales para la asignación de parámetros de máquina y la producción por lotes, debido a que cada receta agrupa varios registros de datos con una estructura común, pero con valores específicos según la aplicación (Siemens, 2025).

Figura 10
Gestión de recetas en TIA Portal



Nota. Interfaz HMI para la selección y carga de recetas de proceso (PLCTAB, 2024).

4.12 Protocolo Profinet

Este estándar de comunicación industrial utiliza Ethernet para el intercambio de datos en tiempo real entre controladores, dispositivos de campo y equipos de automatización. Es compatible con protocolos TCP/IP. Ofrece diagnósticos avanzados y admite distintos niveles de rendimiento. Esto lo hace más óptimo para aplicaciones simples y procesos que exigen alta precisión temporal (IBÉRICA, 2025).

V. MARCO METODOLÓGICO

El desarrollo del proyecto se apoyó en metodologías como la documental, la descriptiva del proceso de clasificación, la metodología experimental y la de exploración, que facilitaron el análisis y la validación del proceso simulado.

La metodología documental permitió recopilar conceptos y referencias sobre clasificación textil, programación del PLC, desarrollo de interfaces HMI e integración con IoT2040, asegurando una base teórica consistente.

La metodología descriptiva del proyecto dio forma al diseño de la lógica en el PLC, en la construcción de la interfaz del HMI y en la definición del dashboard, además, esta metodología proporcionó la posibilidad de describir cada una de las acciones que llevó a cabo el proceso, y la interacción que se fue generando entre variables, etapas y los resultados que se recolectaron.

La metodología experimental se aplicó en la fase de pruebas, evaluando el desempeño del PLC, la correcta actualización de la información en el HMI y la transmisión de datos al IoT2040. Con ello se realizaron ajustes para mejorar el funcionamiento general.

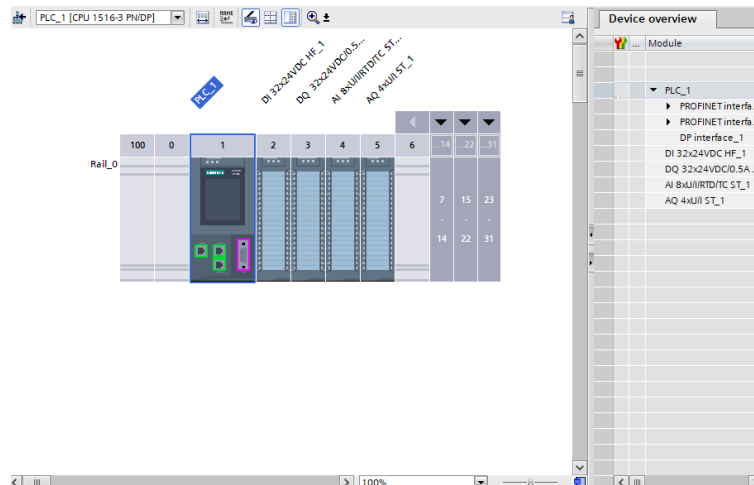
Por último, la metodología de exploración analizó las capacidades del IoT2040 dentro del monitoreo remoto, evaluando los tiempos de actualización y el comportamiento de la comunicación con el PLC.

5.1 Inicialización del proyecto en el entorno TIA Portal

La programación en TIA Portal se realizó con el módulo PLC S7-1500, siguiendo una secuencia clara de pasos para configurar correctamente el sistema. Primero, se ejecutó el software y se creó un nuevo proyecto. Luego, se seleccionaron el controlador S7-1500 y el visualizador HMI KTP700 Basic. Después, se añadieron los módulos de entrada y salida digitales y analógicas necesarios, lo que permitió establecer la comunicación para que el PLC funcionara, ver figura 11.

Figura 11

Selección de módulos de entradas y salidas analógicas y digitales para el PLC

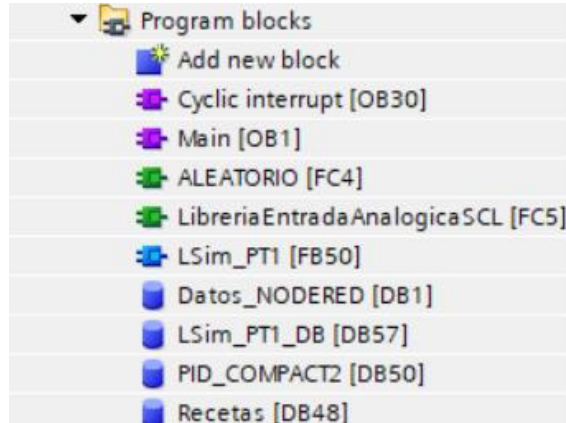


Nota. Ubicación de módulos de entradas y salidas digitales y analógicas en el PLC S7-1500

5.2 Bloques para la programación del PLC

En la figura 12, se organizó la estructura de los bloques de programación del PLC dentro de la carpeta “*Program blocks*”, donde se identificaron los distintos tipos de bloques utilizados para el funcionamiento del sistema de control.

Figura 12
Bloques de programación



Nota. Bloques de programación que se usaron para el funcionamiento del proyecto.

En esta carpeta se incluyen:

Main [OB1]: Encargado de ejecutar de forma cíclica la lógica general del programa.

Cyclic interrupt [OB30]: Se encarga de asegurar que el controlador PID funcione correctamente.

Bloque [FC4]: Transmite datos de forma aleatoria para poder simular la clasificación de las prendas.

Bloque [FC5]: Se encarga de normalizar y escalar la programación del PID.

Bloque [FB50]: Almacena variables que son utilizadas en la librería LSim_PT1.

Bloques [DB]: Tienen los datos que se ubican en los bloques de la programación del “Main”.

5.3 Definición de variables para el sistema de control

En la figura 13, se presentó la definición de una serie de variables tales como “INICIO %M0.0”, “PARO %M0.1” y “SYSTEM_ON %M0.2”, las cuales correspondieron a memorias de tipo booleano y se emplearon para el control general del sistema. Las señales “MEDIDOR_ON %M0.3” y “NIVEL_MEDIDOR %MW2”

estuvieron asociadas al sistema de almacenamiento de la ropa, donde la primera activó el sensor y la segunda, de tipo entero, almacenó el valor del nivel medido. Las variables “CONVEYOR %M0.5” y “CONVEYOR_ON %M0.6”, junto con “MOV_FERA %MW4”, “2DA_BANDA %M0.7” y “2DA_BANDA_TRANS %M1.0”, controlaron el movimiento de las bandas transportadoras, combinando señales booleanas para la activación y variables enteras para el desplazamiento.

Figura 13
Sección 1 de variables de “GLOBAL_MEMORY”

GLOBAL_MEMORY								
	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Superv
1	INICIO	Bool	%M0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
2	PARO	Bool	%M0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
3	SYSTEM_ON	Bool	%M0.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
4	MEDIDOR_ON	Bool	%M0.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
5	NIVEL_MEDIDOR	Int	%MW2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
6	CONVEYOR	Bool	%M0.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
7	CONVEYOR_ON	Bool	%M0.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
8	MOV_1ERA	Int	%MW4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	2DA_BANDA	Bool	%M0.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
10	2DA_BANDA_TRANS	Bool	%M1.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
11	MOV_ROPA_SUP	Int	%MW8	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
12	OUT_ROPA_SUP	Bool	%M1.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
13	ACTUADOR1	Bool	%M1.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
14	MOV_ACT_1	Int	%MW12	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
15	BANDA_SUP_ON	Bool	%M1.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
16	BANDA_SUP	Bool	%M1.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
17	MOV_SUPERIOR	Int	%MW14	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
18	SALIDA_SUP	Bool	%M1.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
19	DATO_ALEATORIO	Int	%MW30	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
20	OUT_ROPA_INF	Bool	%M32.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
21	MOV_ROPA_INF	Int	%MW42	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
22	OUT_CALCETINES	Bool	%M33.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
23	MOV_CAL	Int	%MW54	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
24	ACT_ROPA_SUP	Bool	%M33.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
25	MOV_ACT_2	Int	%MW72	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
26	ACTUADOR2	Bool	%M68.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	

Nota. Variables registradas en el “GLOBAL_MEMORY”.

Las siguientes variables complementaron la lógica de control del sistema de la sección anterior tales como “ACTUADOR_3 %M68.7” y “ACT_CALCETINES %M69.0”, de tipo booleano, las cuales se utilizaron para la activación y de los actuadores y las señales “MOV_ACT_3 %MW78”, “MOV_CALCETIN %MW86” y “MOV_INF %MW92”, de tipo entero, permitieron gestionar el movimiento y la secuencia operativa de los mecanismos asociados. Por su parte las variables “BANDA_CAL_ON %M69.5”, “BANDA_CAL

%M69.6”, “BANDA_INF %M69.7” y “BANDA_INF_ON %M84.0” controlaron el encendido y funcionamiento de las bandas transportadoras destinadas a la clasificación y transporte de la prenda inferior y calcetín. Finalmente, las variables “SALIDA_CAL %M84.1” y “SALIDA_INF %M84.2”, de tipo booleano, representaron la salida para cada tipo de prenda permitió su clasificación, ver figura 14.

Figura 14

Sección 2 de las variables pertenecientes al “GLOBAL_MEMORY”

GLOBAL_MEMORY									
	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Supervision	Comment
27	ACT_ROPA_INF	Bool	%M68.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
28	ACTUADOR_3	Bool	%M68.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
29	ACT_CALCETINES	Bool	%M69.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
30	MOV_ACT_3	Int	%MW78	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
31	BANDA_CAL_ON	Bool	%M69.5	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
32	BANDA_CAL	Bool	%M69.6	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
33	BANDA_INF	Bool	%M69.7	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>		
34	BANDA_INF_ON	Bool	%M84.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
35	MOV_CALCETIN	Int	%MW86	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
36	SALIDA_CAL	Bool	%M84.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
37	SALIDA_INF	Bool	%M84.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
38	MOV_INF	Int	%MW92	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		

Nota. Sección 2 de las variables utilizadas en la programación.

En la figura 15, se mostraron las variables “PLANCHA1_ON %M99.0”, “PLANCHA_ON %M99.3” y “PLANCHA_OFF %M99.4”, de tipo booleano, las cuales se utilizaron para habilitar y deshabilitar el proceso de planchado. Las señales de “SetPointTemp %MW114” y “SetPointLimit %MW146,” de tipo entero, definieron el valor de referencia y el límite de temperatura que se establecieron desde la pantalla, mientras que “ActualTemp %MD134” de tipo Real almacenó la temperatura del proceso. Las variables “REINICIO %M99.2”, “INICIO_FISICO %I0.0”, “PARO_FISICO %I0.1” y “PAUSE %M1.3” permitieron la gestión del arranque, parada, reinicio y pausa del sistema desde pulsadores físicos. Las variables “LUZ_ROPA_SUP %Q0.0”, “LUZ_ROPA_INF %Q0.1” y “LUZ_CALCETINES %M0.4” controlaron indicadores luminosos dependiendo del tipo de prenda clasificada. Además, mientras que el “FINAL_DE_CARRERA %M6.1” aseguró el accionamiento del sensor de posición del

mecanismo y la variable “NUM_ROP_PLANCH %MW10” registró de manera automática el número de prendas procesadas durante la etapa de planchado.

Figura 15
Sección 3 de las variables del “GLOBAL_MEMORY”

39	PLANCHA1_ON	Bool	%M99.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
40	PLANCHA	Bool	%M99.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
41	SetPointTemp	Int	%MW114	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
42	ActualTemp	Real	%MD134	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
43	PID_OUTPUT	Real	%MD138	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
44	REINICIO	Bool	%M99.2	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
45	SetPointLimit	Int	%MW146	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
46	PLANCHA_ON	Bool	%M99.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
47	PLANCHA_OFF	Bool	%M99.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
48	SalidaPID	Real	%MD120	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
49	LUZ_ROPA_SUP	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
50	LUZ_ROPA_INF	Bool	%Q0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
51	LUZ_CALCETINES	Bool	%M0.4	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
52	INICIO_FISICO	Bool	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
53	PARO_FISICO	Bool	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
54	PAUSE	Bool	%M1.3	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
55	FINAL DE CARRERA	Bool	%M6.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
56	NUM_ROP_PLANCH	Int	%MW10	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

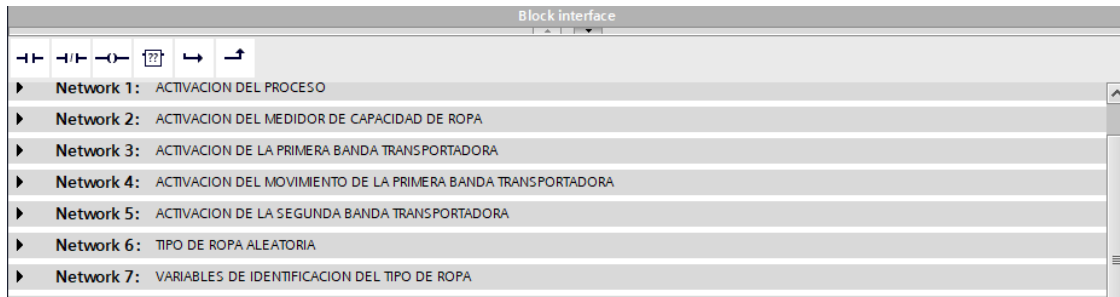
Nota. Sección 3 de las variables empleadas en la programación.

5.4 Programación KOP del Main

El bloque Main fue el encargado de programar la lógica central del proceso, la cual se ejecutó de forma cíclica para coordinar todas las etapas del proceso automatizado. Dentro de este bloque, la programación se organizó en distintos networks, figura 16, cada uno dedicado a una función específica. En el “Network 1” se realizó la activación general del proceso; el “Network 2” se encargó de la activación del medidor de almacenamiento de ropa; mientras que los “Network 3”, “Network 4” y “Network 5” se destinaron a la activación y el movimiento de las bandas transportadoras. En el “Network 6” se generó un número aleatorio y en el “Network 7” se asignó una variable con los valores generados.

Figura 16

Primer segmento de Networks del bloque Main



Nota. Networks en donde va la programación principal.

En la figura 17, los Networks 8, 9 y 10 controlaron respectivamente el movimiento de los productos textiles y, a su vez, los Networks 11, 12 y 13 gestionaron el accionamiento de los actuadores asociados a la ropa superior, ropa inferior y calcetines. En el “Network 14” se activó el conjunto de bandas transportadoras clasificadas; “Network 15”, “Network 16” y “Network 17” gestionaron el movimiento de las prendas ya clasificadas, trasladándolas a sus cajas de almacenamiento. Finalmente, el “Network 18” se encargó de activar el proceso de planchado, el “Network 19” se encargó de ejecutar el conteo de prendas mediante un contador al activarse el final de carrera, y en el “Network 20” se implementó el reinicio del proceso para un nuevo ciclo de operación.

Figura 17

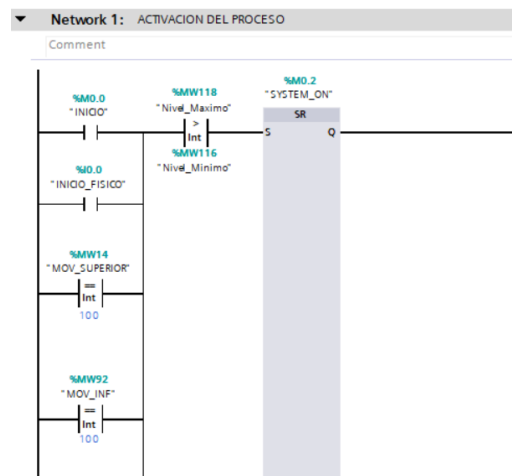
Segundo segmento de Networks del bloque Main



Nota. Sección 2 de los Networks que pertenecen al Main.

En el “Network 1”, el proceso se activó mediante la señal “INICIO %M0.0” o “INICIO_FISICO %I0.0”, habilitando el bloque SR que mantuvo encendida la variable “SYSTEM_ON %M0.2”; además, se verificó que la señal “Nivel_Maximo %MW118” fuera mayor que “Nivel_Minimo %MW116” y que se cumplieran las condiciones de “MOV_SUPERIOR %MW14” y “MOV_INF %MW92”, con el fin de asegurar que los mecanismos se encontraran en la posición adecuada antes de permitir el arranque del sistema, ver Figura 18.

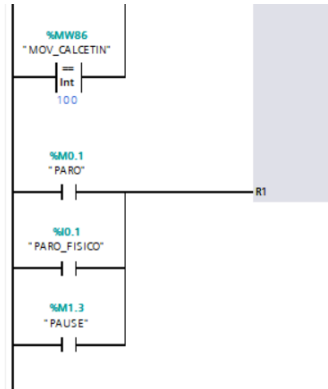
Figura 18
Iniciación del proceso



Nota. Activación del proceso en la cual activa y desactiva la programación.

En la segunda parte del “Network 1” se incorporó la lógica de detención y reinicio del sistema, en la que la variable “MOV_CALCETIN %MW86” fue evaluada mediante un comparador para confirmar que el movimiento haya alcanzado el valor establecido antes de permitir la continuidad del proceso. Asimismo, la configuración de las señales “PARO %M0.1”, “PARO_FISICO I0.1” y “PAUSE %M1.3” actuó directamente sobre la entrada R1 del bloque de programación SR, forzando el reinicio del estado del sistema al accionar cualquiera de ellas, lo que provocó la desactivación de “SYSTEM_ON %M0.2”, ver figura 19.

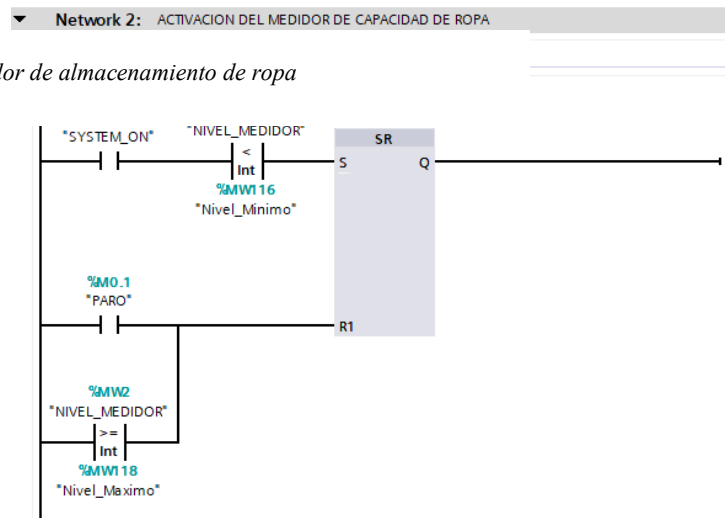
Figura 19
Segunda parte del Network 1



Nota. Segunda parte de la activación del proceso.

En la figura 20, el “Network 2”, controló la activación del medidor de capacidad de ropa, la cual dependió de que el sistema estuviera previamente habilitado mediante la señal “SYSTEM_ON %M0.2”. La variable “NIVEL_MEDIDOR %MW2” fue comparada con un valor mínimo definido y, cuando se cumplió esta condición, se energizó la entrada S del bloque SR, activando la señal “MEDIDOR_ON %M0.3”. De esta manera, cuando el nivel se encontró dentro del rango permitido, el medidor entró en funcionamiento.

Figura 20
Activación del medidor de almacenamiento de ropa



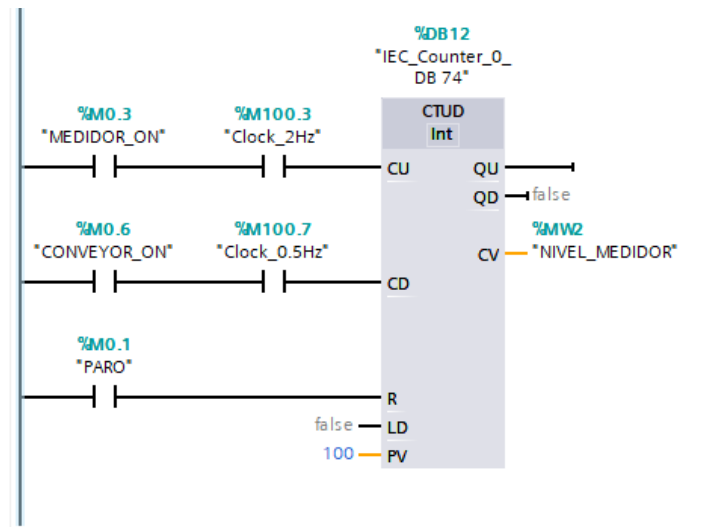
Nota. Programación del nivel de almacenamiento de ropa.

En esta sección del “Network 2” se incorporó, como bloque de programación, un contador tipo CTUD, encargado de gestionar el valor del “NIVEL_MEDIDOR %MW2”.

El conteo ascendente se produjo cuando “MEDIDOR_ON %M0.3” estuvo activa y se recibió una señal de reloj a 2 Hz, mientras que el conteo descendente se realizó cuando “CONVEYOR_ON %M0.6” estuvo habilitada junto con un pulso de reloj a 0,5 Hz. El valor actual del contador se almacenó en la variable “NIVEL_MEDIDOR %MW2”, permitiendo representar dinámicamente la cantidad de ropa presente en el sistema, ver figura 21.

Figura 21

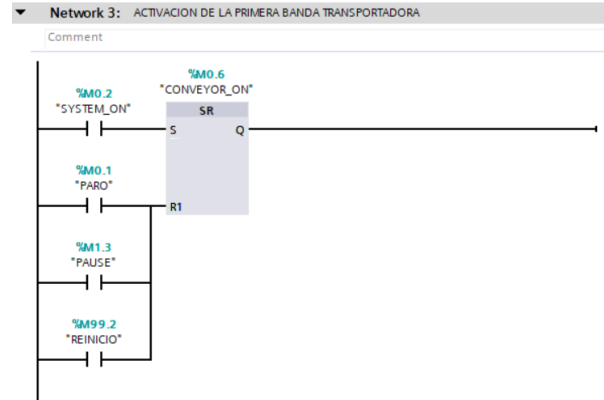
Nivel y tiempo de almacenamiento de la ropa



Nota. Programación del tiempo del nivel de almacenamiento de la ropa.

En la figura 22, correspondió al “Network 3”, que se encargó de la activación de la primera banda transportadora, la cual quedó habilitada mediante la señal “SYSTEM_ON %M0.2”, lo que activó la entrada S del bloque de memoria SR asociado a la variable “CONVEYOR_ON %M0.6”. De esta forma, la banda se mantuvo en funcionamiento mientras el proceso permaneció activo. Por otro lado, las señales “PARO %M0.1”, “PAUSE %M1.3” y “REINICIO %M99.2” actuaron sobre la entrada R1 del bloque SR, lo que provocó la desactivación de la banda transportadora.

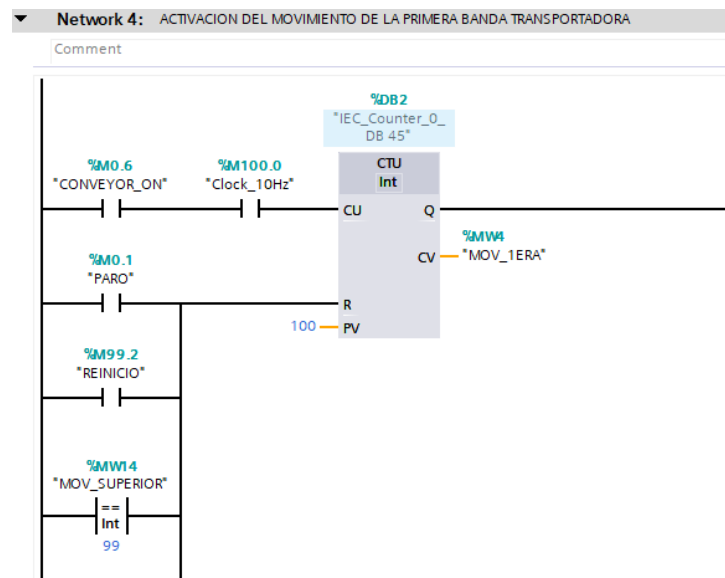
Figura 22
Habilitación de la primera banda transportadora



Nota. Programación de la primera banda transportadora.

En el “Network 4” se programó el movimiento de la primera banda transportadora mediante un contador CTU, el cual incrementó el valor de “MOV_1ERA %MW4” cuando la banda estuvo habilitada y recibió pulsos de reloj, deteniéndose o reiniciándose con las señales “PARO %M0.1” y “REINICIO %M99.2”, y se sincronizó con los movimientos de la ropa superior, inferior y de calcetines, ver figura 23.

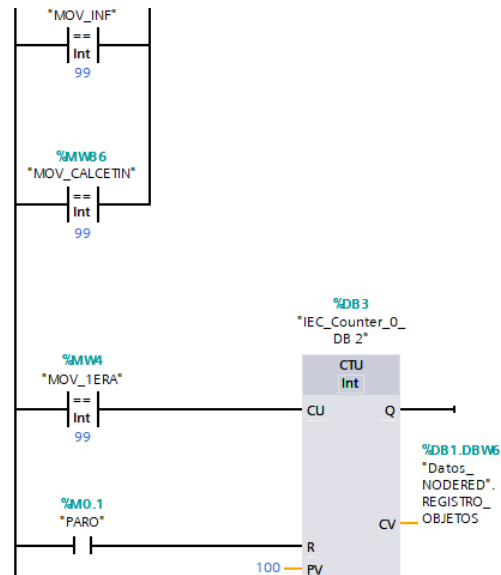
Figura 23
Movimiento de la primera banda transportadora



Nota. Programación del movimiento de la banda transportadora.

En la figura 24, se utilizó un segundo contador para registrar el movimiento de la primera banda cuando “MOV_1ERA %MW4” alcanzó el valor definido, y se almacenó la información en el bloque de datos “Datos_NODERED_REGISTRO_OBJETOS %DB1.DBW6”, lo que permitió llevar un control y registro del proceso.

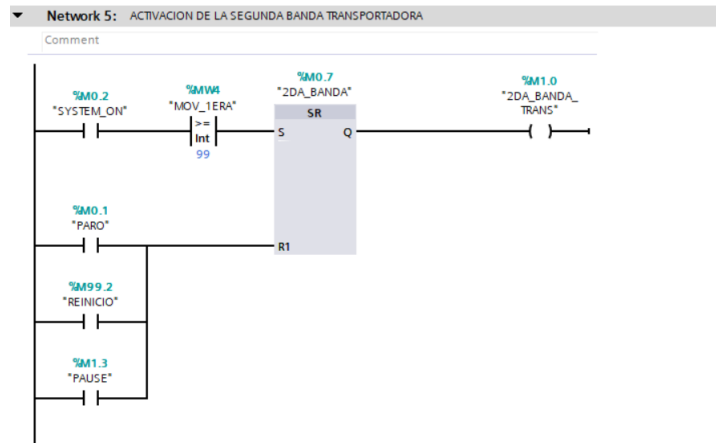
Figura 24
Registro de objetos



Nota. Programación con bloque CTU.

En la Figura 25, el “Network 5”, que activó la segunda banda transportadora, la cual se habilitó mediante la señal “SYSTEM_ON %M0.2”, y se cumplió la condición de que el desplazamiento de la primera banda, representado por la variable “MOV_1ERA %MW4”, haya alcanzado el valor establecido. Al cumplirse estas condiciones, se energizó la entrada S del bloque SR, lo que activó la variable “2DA_BANDA %M0.7”, cuya salida permitió el funcionamiento de la segunda banda a través de la señal “2DA_BANDA_TRANS %M1.0”.

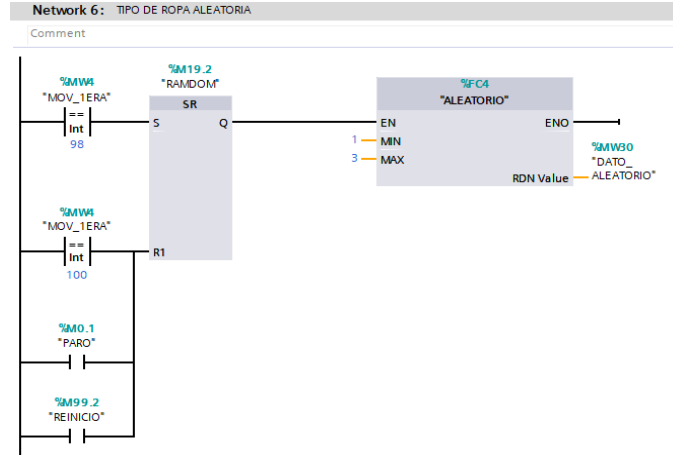
Figura 25
Habilitación de la segunda banda Transportadora



Nota. Programación de la segunda banda transportadora.

En la programación del “Network 6”, se determinó de forma aleatoria el tipo de ropa cuando la variable “MOV_1ERA %MW4” alcanzó el valor establecido, activando el bloque de programación SR que habilitó la señal “RANDOM %M19.2”. Esta señal permitió ejecutar la función “ALEATORIO %FC4”, la cual generó un valor entre 1 y 3, almacenándolo en la variable “DATO_ALEATORIO %MW30”, que fue utilizada posteriormente para identificar el tipo de prenda. Por el contrario, las señales “PARO %M0.1”, “REINICIO %M99.2” y la igualdad entre la señal “MOV_1ERA %MW4” a 100 activaron la entrada R1 del bloque SR, lo que causó la desactivación del bloque “ALEATORIO FC4”, ver figura 26.

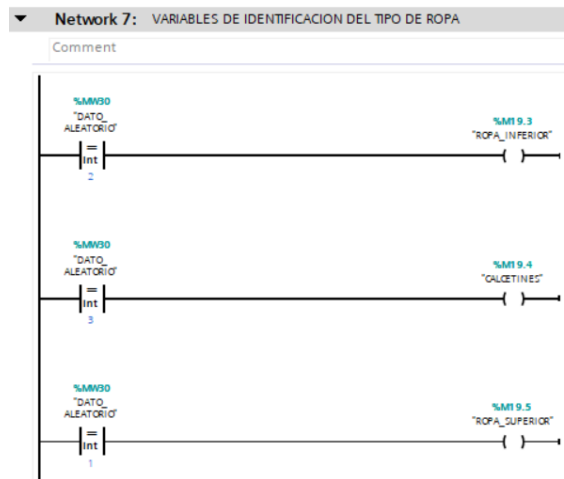
Figura 26
Tipo de ropa aleatoria



Nota. Se observa la programación para tipo de dato aleatorio de prenda.

En el “Network 7” se realizó la identificación del tipo de ropa a partir del valor generado en la variable “DATO_ALEATORIO %MW30”. Cuando este valor fue igual a 2, se activó la señal “ROPA_INFERIOR %M19.3”; si el valor correspondió a 3, se habilitó la señal “CALCETINES %M19.4”; y cuando el valor fue 1, se activó la señal “ROPA_SUPERIOR %M19.5”. De esta manera, el sistema tradujo el dato aleatorio en señales lógicas que permitieron clasificar cada prenda y dirigirla posteriormente por su respectiva ruta dentro del proceso, ver figura 27.

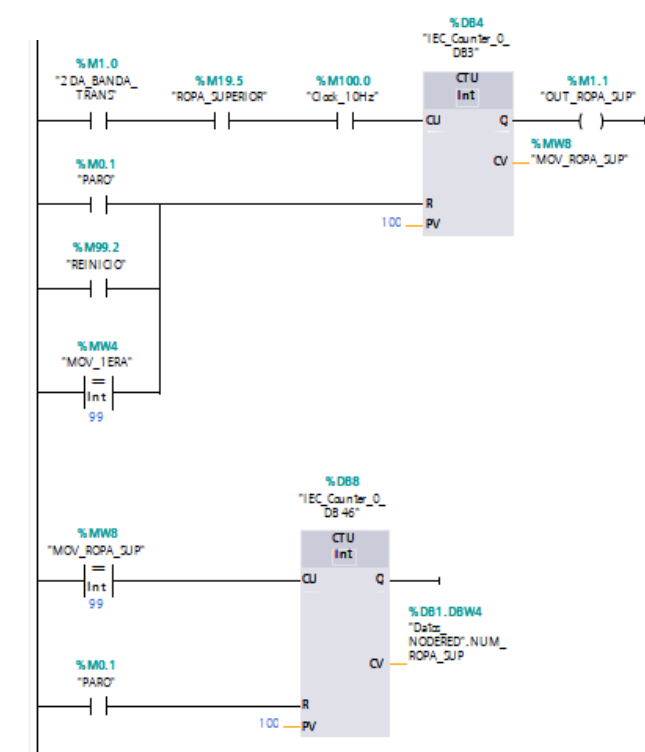
Figura 27
Asignación del dato aleatorio a la ropa



Nota. Asignación de cada dato aleatorio a cada prenda.

En la figura 28, el “Network 8” que gestionó el movimiento de la ropa superior, donde un contador controló su desplazamiento cuando la señal “2DA_BANDA_TRANS %M1.0” estuvo activa y la prenda fue identificada por la señal “ROPA_SUPERIOR %M19.5”, activando la salida “OUT_ROPA_SUP %M1.1” al llegar al valor establecido y registrando el conteo de prendas, el número de prendas clasificadas se almacenó en “Datos_NODERED.NUM_ROPA_SUP %DB1.DBW4”, mientras que las señales “PARO %M0.1” y “REINICIO %M99.2” permitieron reiniciar el conteo teniendo un control seguro del proceso.

Figura 28
Movimiento de la ropa superior

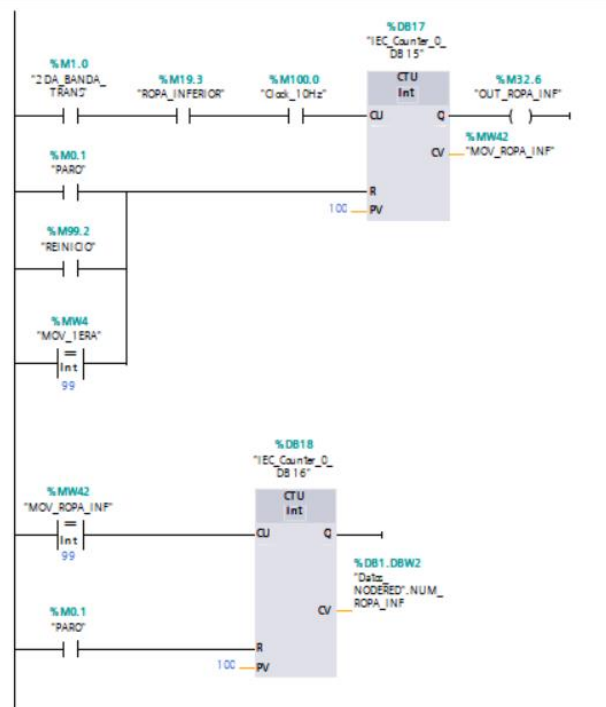


Nota. Programación del movimiento de la ropa superior.

Dentro de la configuración del “Network 9”, se controló el movimiento de la ropa inferior cuando la segunda banda transportadora estuvo activa y el sistema identificó la señal “ROPA_INFERIOR %M19.3”, lo que resultó en la activación de la salida

“OUT_ROPA_INF %M32.6”. Por medio de un contador, el desplazamiento se incrementó con pulsos de reloj y se almacenó en la variable “MOV_ROPA_INF %MW42”, activando la salida “OUT_ROPA_INF %M32.6” al alcanzar el valor establecido. De forma complementaria, un segundo contador registró la cantidad de prendas procesadas en “Datos_NODERED.NUM_ROPA_INF %DB1.DBW2”, guardando la información en el bloque de datos correspondiente, ver figura 29.

Figura 29
Movimiento de la ropa inferior

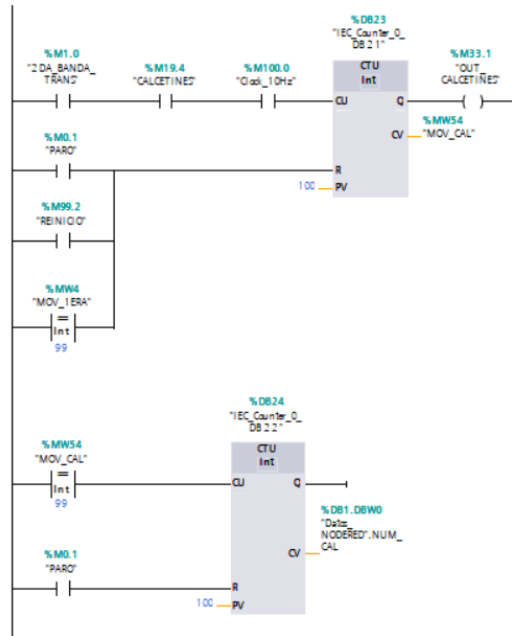


Nota. Programación del movimiento de la ropa inferior.

En la figura 30, el “Network 10” que realizó el movimiento de los calcetines una vez que la segunda banda transportadora se encontró activa y el sistema validó la señal “ROPA_INFERIOR %M19.3”, lo cual activó la salida “OUT_CALCETINES %M33.1”. Este desplazamiento fue controlado mediante un contador, el cual incrementó su valor con pulsos de reloj y lo almacenó en la variable “MOV_CAL %MW54”. Además, un segundo

contador se encargó de registrar la cantidad de calcetines procesados en la dirección de memoria “Datos_NODERED.NUM_CAL %DB1.DBW0”.

Figura 30
Movimiento de los calcetines

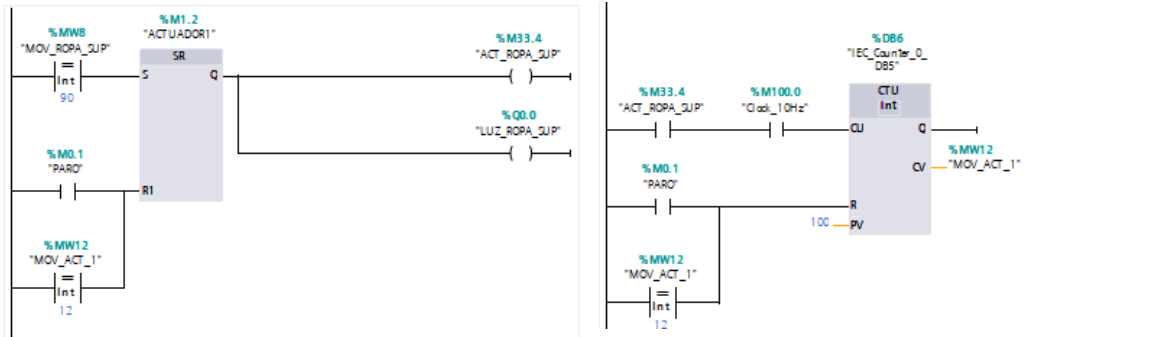


Nota. Programación que hace que los calcetines se muevan.

En el “Network 11”, por medio de un bloque de programación SR, la señal “ACTUADOR1 %M1.2” se mantuvo activa mientras se cumplieron las condiciones del proceso, habilitando tanto la salida “ACT_ROPA_SUP %M33.4” como el indicador “LUZ_ROPA_SUP %Q0.0”. Asimismo, un contador gestionó el tiempo o número de pulsos necesarios para el accionamiento del actuador, almacenando el valor en la variable “MOV_ACT_1 %MW12”, ver figura 31.

Figura 31

Accionamiento del actuador de la ropa superior

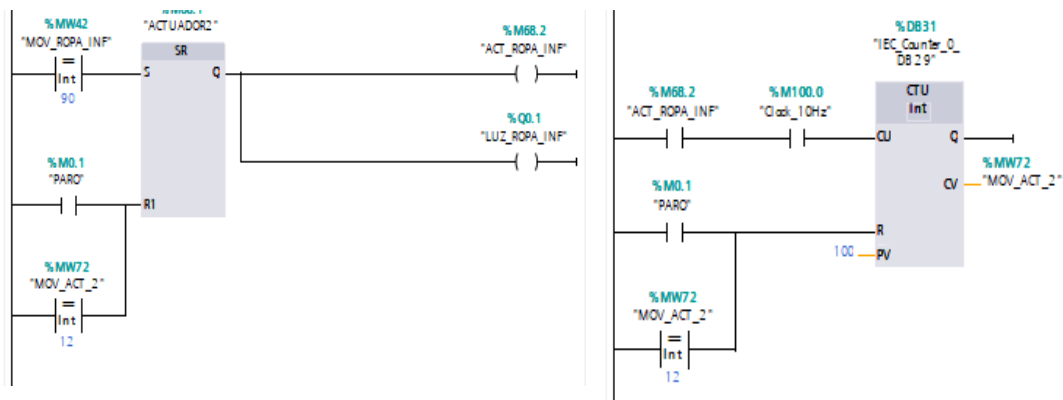


Nota. Network el cual muestra la activación del actuador de la ropa superior.

En la Figura 32, el “Network 12” gestionó la activación del actuador de la ropa inferior cuando la prenda alcanzó la posición establecida, mediante la señal “ACTUADOR2 %M68.1”, manteniéndose habilitado a través de un bloque SR que permitió activar la salida “ACT_ROPA_INF %M68.2” y el indicador “LUZ_ROPA_INF %Q0.1”. Este proceso se gestionó mediante un contador CTU y pulsos de reloj de 10 Hz, registrando el valor en la variable “MOV_ACT_2 %MW72”, el cual se reinició al accionarse “PARO %M0.1” o cuando se cumplió la igualdad entre “MOV_ACT_2 %MW72” y el valor 12 para garantizar una operación segura.

Figura 32

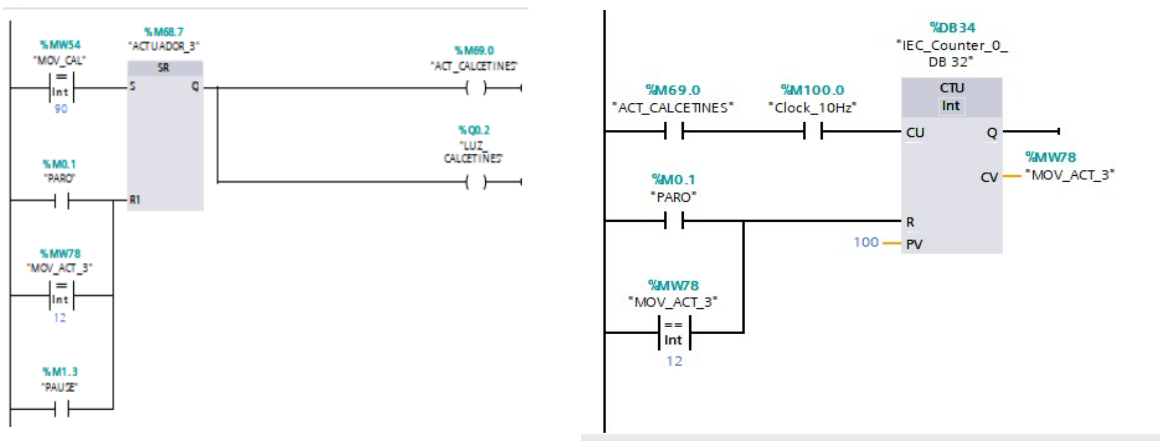
Activación del actuador de la ropa inferior



Nota. Programación que pertenece al accionamiento del actuador de la ropa inferior.

En el “Network 13” se controló el accionamiento del actuador de los calcetines, que se activó cuando la señal “MOV_CAL %MW54” alcanzó la posición definida. Mediante un bloque SR, la señal “ACTUADOR_3 %M68.7” se mantuvo activa mientras se cumplieron las condiciones del proceso, habilitando tanto la salida “ACT_CALCETINES %M69.0” como el indicador luminoso “LUZ_CALCETINES %Q0.2”. Además, un contador CTU reguló el tiempo o número de pulsos requeridos para el accionamiento del actuador, almacenando el valor en la variable “MOV_ACT_3 %MW78”, ver figura 33.

Figura 33
Accionamiento y conteo del actuador de los calcetines

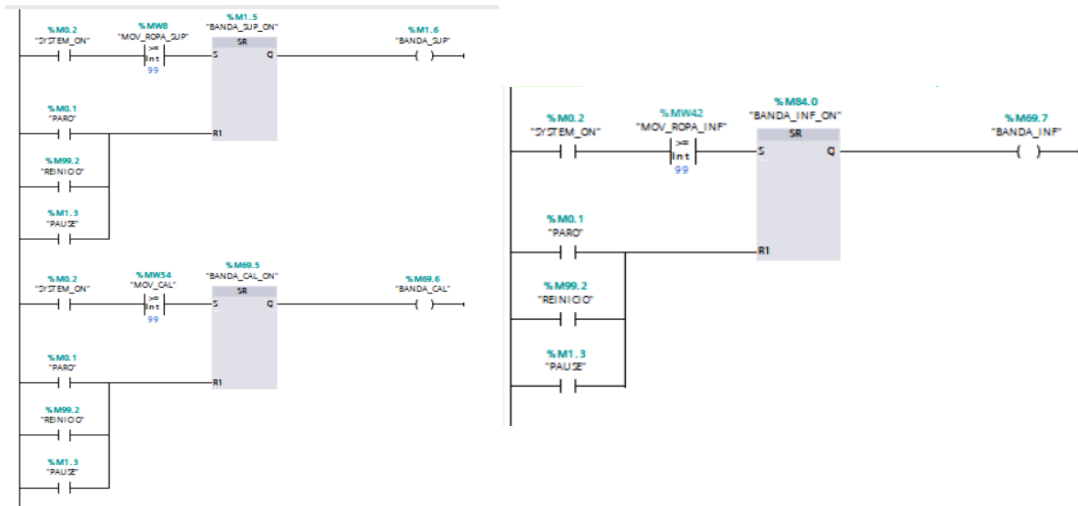


Nota. Lógica de control para el accionamiento del actuador y el conteo de ciclos.

En el “Network 14”, presentado en la figura 34, las bandas transportadoras funcionaron únicamente si el sistema estaba habilitado y se alcanzó el desplazamiento necesario. Si la señal “SYSTEM_ON %M0.2” estuvo activa y se alcanzó el valor de movimiento configurado, el bloque SR se activó y mantuvo la banda en marcha. Las señales “PARO %M0.1”, “REINICIO %M99.2” y “PAUSE %M1.3” activaron R1, lo que detuvo las bandas.

Figura 34

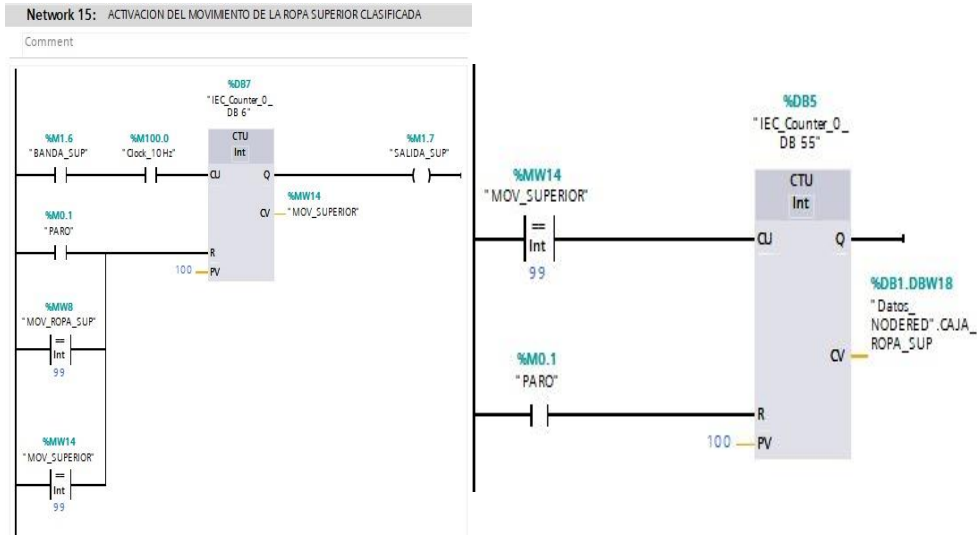
Control de las bandas transportadoras en función del estado del sistema y el desplazamiento



Nota. Lógica de control de las bandas transportadoras según el estado del sistema

En el “Network 15” se controló el desplazamiento de la ropa superior clasificada cuando la banda transportadora “BANDA_SUP %M1.6” se encontró activa. El avance de la prenda se controló mediante la señal “MOV_SUPERIOR %MW14”, que habilitó el conteo en el contador de tipo CTU hasta alcanzar el valor preestablecido. Al cumplirse esta condición, se activó la salida “SALIDA_SUP %M1.7”, indicando que la prenda llegó a la posición definida, permitiendo que el resultado del conteo quedara registrado en la variable “Datos_NODERED.CAJA_ROPA_SUP %DB1.DBW18”. Asimismo, la señal “PARO %M0.1” actuó sobre la entrada R del contador, permitiendo reiniciar el conteo y detener el desplazamiento, ver figura 35.

Figura 35
Activación y control del desplazamiento de la ropa superior clasificada

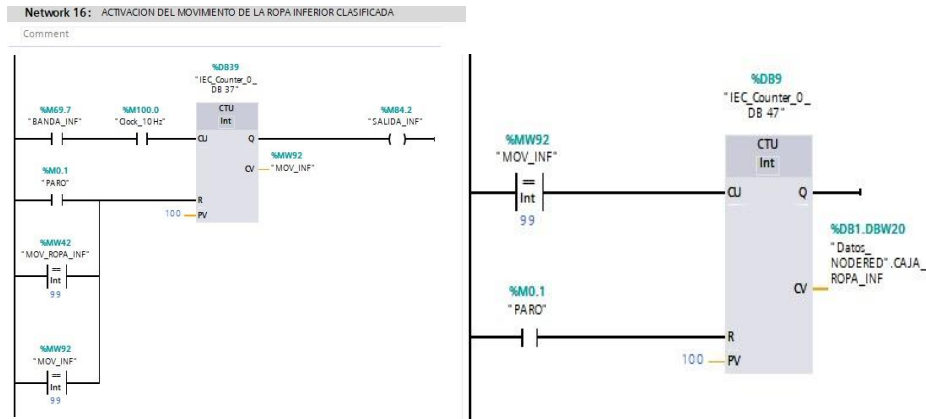


Nota. Programación del control de movimiento de la ropa superior clasificada mediante conteo de pulsos.

En cambio, en el “Network 16” se controló el desplazamiento de la ropa inferior clasificada cuando la banda transportadora “BANDA_INF %M69.7” estuvo activa. El avance de la prenda se gestionó mediante la señal “MOV_INF %MW92”, que habilitó el conteo del desplazamiento mediante un contador de tipo CTU hasta alcanzar el valor preestablecido. Al cumplirse esta condición, se activó la salida “SALIDA_INF %MB4.2”, lo que marcó su llegada al lugar destinado. El valor del contador se almacenó en la variable “Datos_NODERED.CAJA_ROPA_INF %DB1.DBW20” para registrar el proceso, ver figura 36.

Figura 36

Activación y control del desplazamiento de la ropa inferior clasificada

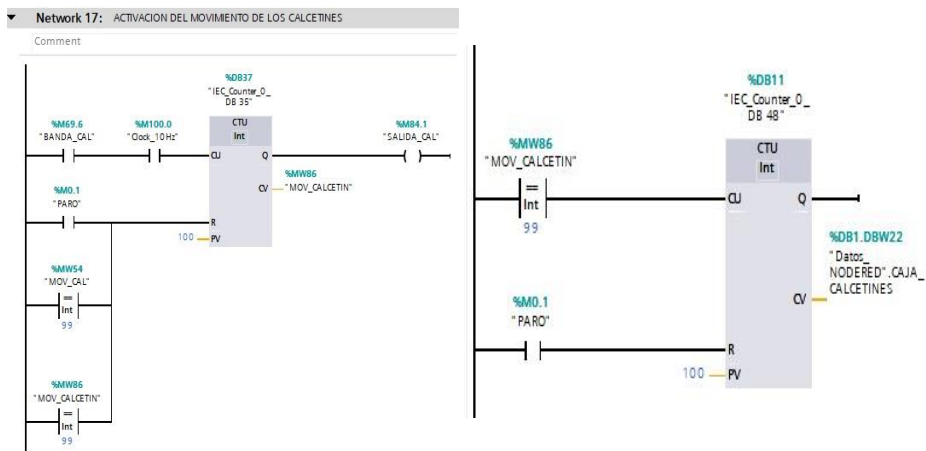


Nota. Control del desplazamiento de la ropa inferior clasificada por medio de conteo de pulsos.

En la figura 37, el “Network 17” se controló el desplazamiento de los calcetines clasificados cuando la banda transportadora “BANDA_CAL %M69.6” se encontró activa. El avance de los calcetines se gestionó mediante la señal “MOV_CALCETIN %MW86”, la cual habilitó el conteo del recorrido a través de un contador de tipo CTU hasta alcanzar el valor preestablecido. Al cumplirse esta condición, se activó la salida “SALIDA_CAL %M84.1”, confirmando así su posición final. El valor del conteo fue almacenado en la variable “Datos_NODERED.CAJA_CALCETINES %DB1.DBW22”.

Figura 37

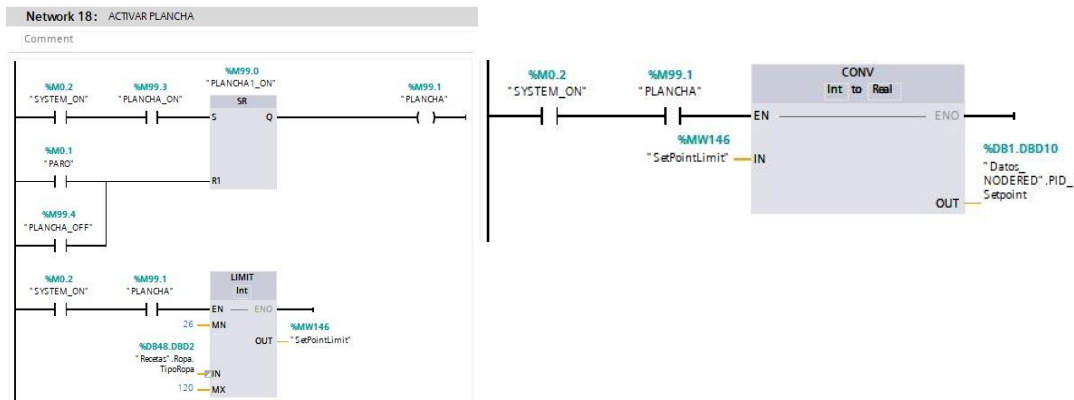
Activación y control del desplazamiento de los calcetines clasificados



Nota. Control del desplazamiento de la ropa inferior clasificada usando conteo de pulsos.

Como se observó en la programación del “Network 18”, figura 38, la activación de la plancha dependió de que el sistema se encontrara habilitado mediante la señal “SYSTEM_ON %M0.2”, manteniendo su estado a través de un bloque SR que controló la salida “PLANCHA %M99.1”. Las señales “PARO %M0.1” y “PLANCHA_OFF %M99.4” permitieron desactivar la plancha cuando fue necesario, mientras que la configuración del valor se gestionó mediante la variable “SetPointLimit %MW146”, la cual se encontró limitada para garantizar un funcionamiento seguro y estable del proceso.

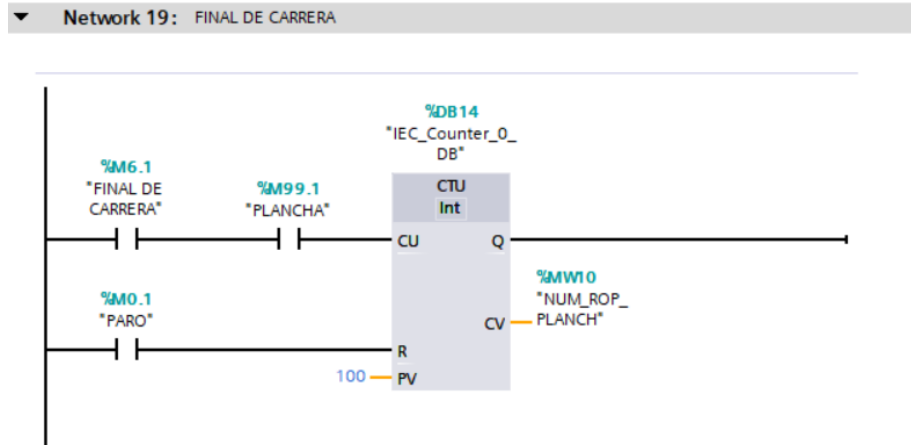
Figura 38
Activación de la plancha



Nota. Activación de la plancha con control de estado y ajuste del valor de consigna.

En el “Network 19” se representó la etapa correspondiente al final de carrera del sistema, en la que intervinieron las señales “FINAL DE CARRERA %M6.1”, “PLANCHA %M99.1” y “PARO %M0.1”. En este segmento se empleó el contador tipo CTU, que permitió gestionar el valor almacenado en la marca “NUM_ROPA_PLANCH %MW10”, lo que formó parte del control secuencial del proceso de clasificación textil, ver figura 39.

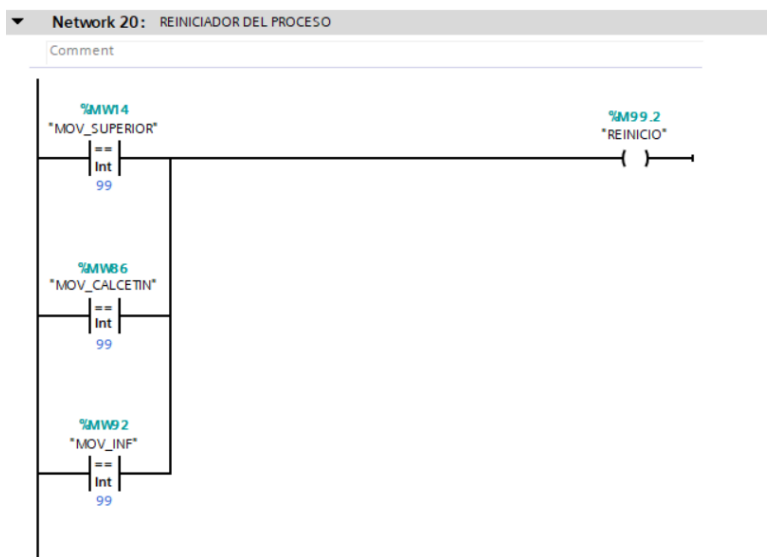
Figura 39
Final de carrera del proceso de planchado



Nota. Bloque de programación utilizado para el control del final de carrera dentro del proceso de planchado.

En la figura 40, el “Network 20” se programó la activación de la señal de reinicio del proceso cuando los movimientos de la ropa superior, inferior y de los calcetines alcanzaron el valor establecido. Al cumplirse esta condición, se activó la señal de “REINICIO %M99.2”, permitiendo restablecer la secuencia y preparar el sistema para un nuevo ciclo de operación de forma ordenada y controlada.

Figura 40
Reinicio del proceso

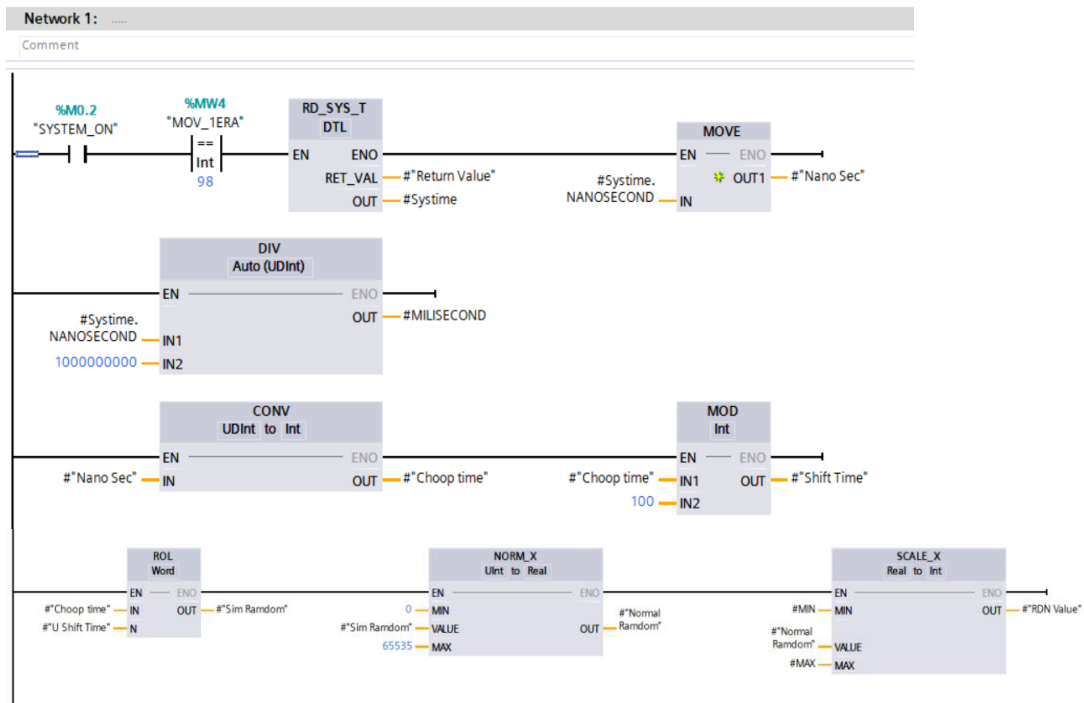


Nota. Reinicio automático del proceso al completarse los movimientos programados.

5.5 Programación en KOP y FUP para la clasificación aleatoria de prendas

En este apartado se desarrolló la programación del subproceso de clasificación en los lenguajes FUP y KOP. El bloque se activó cuando la marca “SYSTEM_ON %M0.2” estuvo encendida y, después, la marca “MOV_1ERA %MW4” fijó la posición inicial del movimiento en la banda transportadora. Con estas condiciones, el tiempo interno del controlador se utilizó como referencia para generar un valor aleatorio que se transfirió, mediante el bloque “MOVE”, a la variable “NanoSec”. Luego, este valor se convirtió a milisegundos con el bloque “DIV”, se normalizó con “NORM X” y, por último, se ajustó a un rango definido con “SCALE X”, ver figura 41.

Figura 41
Generación de valor aleatorio para la clasificación



Nota. generación de un valor aleatorio en el PLC para la clasificación de prendas.

5.6 Programación del controlador PID para la regulación térmica de la plancha

Para modelar el comportamiento térmico de la plancha, se utilizó un modelo matemático basado en una ecuación diferencial que describió la variación de la temperatura durante el calentamiento. En la Figura 42, se mostraron los ajustes realizados en la ecuación, considerando las variaciones térmicas y el efecto de la energía suministrada al sistema de calentamiento (García Nazareno & Vasco Rodríguez, 2025).

Figura 42

Modelo de la función de transferencia

$$S_o(i) = \frac{1}{1 + K_a \cdot q_a \cdot \Delta t} \cdot \{S_o(i-1) + \Delta t \cdot [r_o + K_a \cdot q_a \cdot S_{o,sat}]\}$$

Nota. Representación de la ecuación diferencial correspondiente al sistema de calentamiento. Tomado por García Nazareno & Vasco Rodríguez (2025).

En la figura 43, se presentó el ajuste de la expresión matemática para representar el comportamiento térmico de la plancha y aplicarla al control.

Figura 43

Modelo matemático para la regulación térmica de la plancha

$$T_p(i) = \frac{1}{1 + K_p \cdot q_p \cdot \Delta t} \{T_p(i-1) + \Delta t [r_p + K_p \cdot q_p \cdot T_{ref,p}]\}$$

Nota. Expresión matemática adaptada al comportamiento térmico del proceso.

Donde:

$T_p(i)$: Temperatura de la plancha en el instante i (°C).

$T_p(i-1)$: Temperatura de la plancha en el instante anterior (°C).

Δt : Paso de tiempo utilizado para la simulación.

r_p : Variación térmica interna del sistema de calentamiento de la plancha.

K_p : Constante térmica del sistema de la plancha. N,

q_p : Potencia aplicada a la plancha (W).

$T_{ref,p}$: Temperatura de referencia o consigna establecida.

El controlador PID reguló la temperatura de la plancha al comparar continuamente la temperatura real con el “setpoint”. La acción proporcional corrigió el error inmediato, la integral ajustó las desviaciones acumuladas y la derivativa anticipó los cambios en el comportamiento térmico del sistema

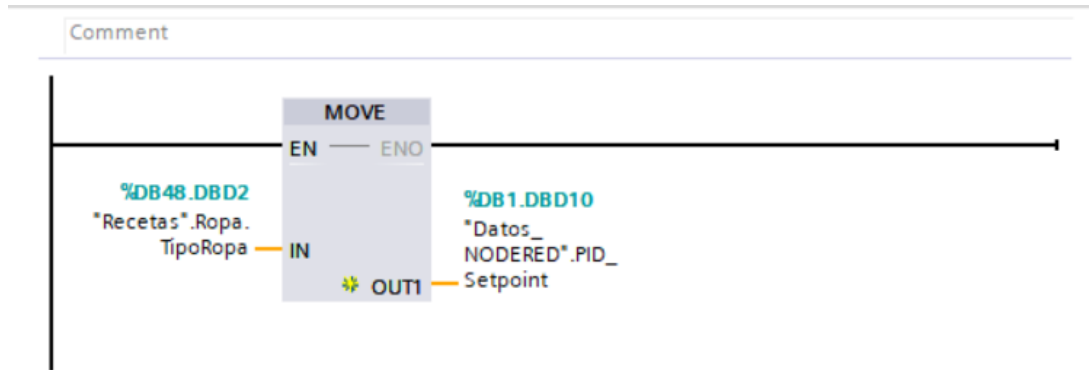
5.6.1 Parametrización del Bloque PID Compact

El bloque de programación “PID_Compact” se ejecutó en un OB de interrupción cíclica para garantizar un correcto funcionamiento del controlador. La utilización de tiempos de muestreo permitió que el PID respondiera de manera más precisa a las condiciones del proceso. Una vez creado el bloque, el “PID_Compact” se insertó en el Network correspondiente, donde se configuraron sus parámetros, las señales de entrada y de salida, así como los valores internos requeridos para la regulación de la temperatura.

Posteriormente, se implementó un bloque adicional donde se empleó la instrucción MOVE para transferir el valor definido en las recetas de tipo de ropa a la variable de consigna del controlador PID. El dato almacenado en “Recetas.Ropa.TipoRopa %DB48.DBD2” se copió directamente en “PID_Setpoint %DB1.DBD10”, lo que actualizó

automáticamente el valor de referencia de temperatura según la receta seleccionada y permitió su uso por el sistema de control, ver figura 44.

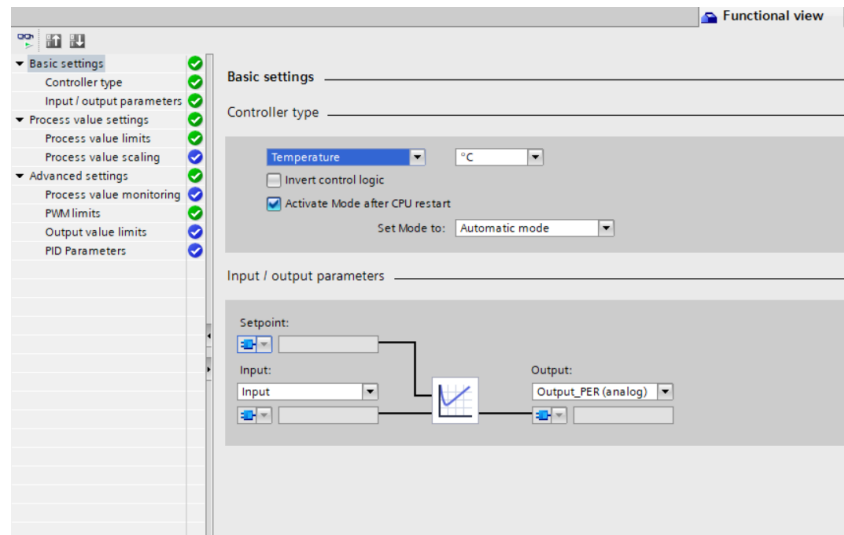
Figura 44
Transferencia de Variables de Receta al Bloque de Función



Nota. Envío del valor definido en la receta hacia el controlador.

En la figura 45, se presentó la configuración del bloque “PID_Compact”, utilizado para la regulación térmica de la plancha, en la que se definió la variable de control como la temperatura y su unidad en grados Celsius (°C). Los indicadores en color verde confirmaron que los parámetros se configuraron correctamente, incluyendo la asignación de entradas y salidas, los límites de la variable de proceso, el escalamiento de señales y el monitoreo del sistema. Además, se establecieron los límites de salida y los parámetros del PID, que ajustaron el comportamiento del controlador según las características térmicas de la plancha, lo que permitió un control más preciso durante la operación del proceso.

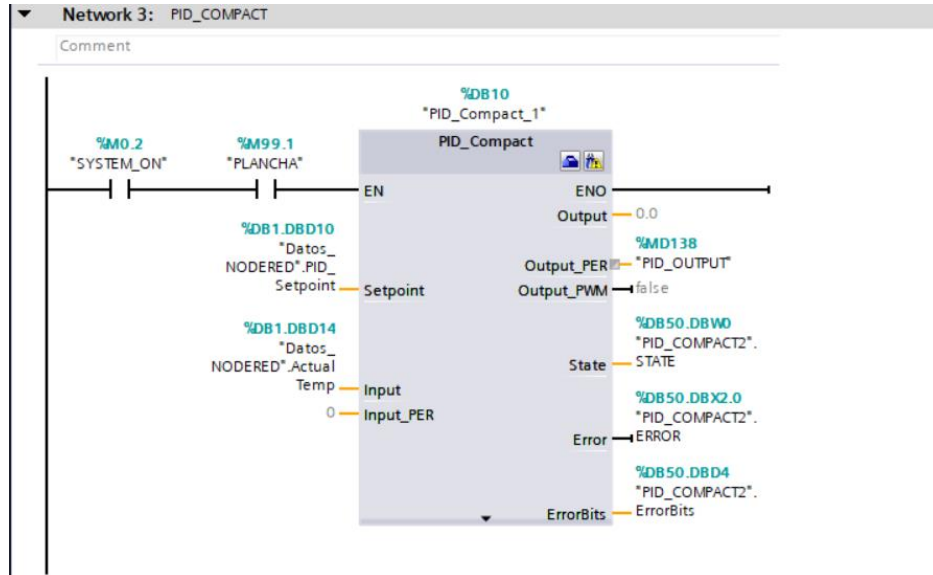
Figura 45
Configuración del controlador PID_Compact



Nota. Parámetros principales configurados para el controlador PID.

En la figura 46, se configuraron las variables necesarias para controlar la temperatura de la plancha. El controlador se habilitó mediante las señales “SYSTEM_ON %M0.2” y “PLANCHA %M99.1”, lo que permitió activar el bloque solo cuando el sistema y la plancha estaban operativos. La variable “Datos_NODERED.PID_Setpoint %DB1.DBD10” definió el valor de referencia de temperatura, mientras que “Datos_NODERED.Actual_Temp %DB1.DBD14” representó la temperatura real del proceso, utilizada como entrada del controlador. La salida del controlador, que reguló la potencia aplicada a la plancha, se obtuvo a través de “PID_OUTPUT %MD138”. Para su funcionamiento, se emplearon variables de diagnóstico como “PID_COMPACT2.STATE %DB50.DBW0”, “PID_COMPACT2.ERROR %DB50.DBX2.0” y “PID_COMPACT2.ErrorBits %DB50.DBD4”, que permitieron identificar el estado operativo y posibles fallas durante el proceso.

Figura 46
Arquitectura del bloque PID con su asignación de registros y memorias

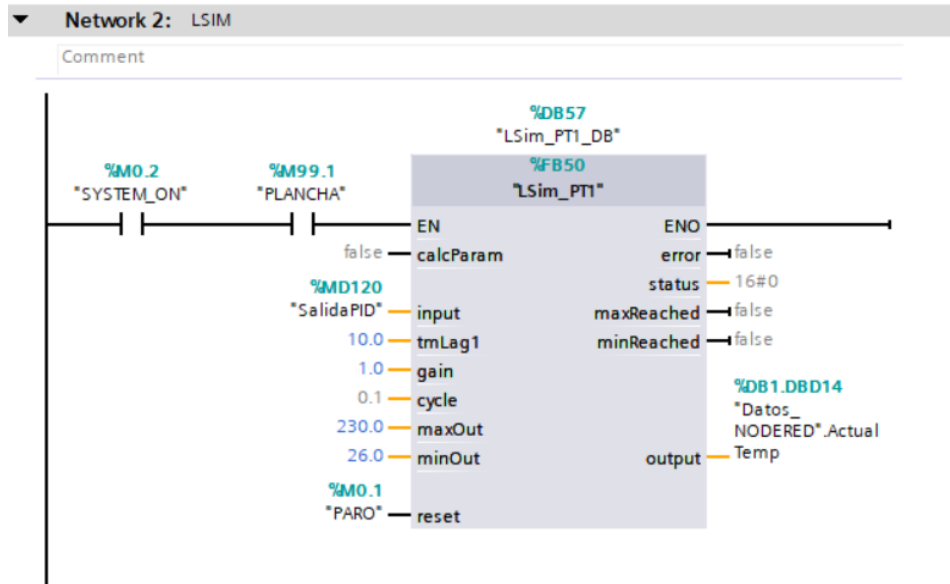


Nota. Se presenta la disposición de los registros y variables vinculados al bloque PID_Compact.

5.6.2 Configuración de la librería LSim

En el bloque LSim_PT1 de la librería LSim simuló el comportamiento térmico del proceso de planchado y se integró en el mismo bloque organizativo que el PID_Compact, figura 47, lo que permitió su interacción directa. El modelo se activó con las señales “SYSTEM_ON %M0.2” y “PLANCHA %M99.1”, mientras que “PARO %M0.1” reinició o detuvo la simulación según fue necesario. La señal de control generada por el PID se aplicó a la entrada mediante “SalidaPID %MD120”. Para el modelado, se configuró $tmLag1 = 10.0$ para la inercia térmica del proceso y $gain = 1.0$ para la relación entrada-respuesta, con un tiempo de muestreo de $cycle = 0.1$. Los límites $minOut = 26.0$ y $maxOut = 230.0$ mantuvieron la temperatura simulada dentro de un rango seguro. La salida del bloque, la temperatura simulada, se almacenó en “Datos_NODERED.ActualTemp %DB1.DBD14” y se utilizó como realimentación para el lazo de control y para el sistema de monitoreo.

Figura 47
Configuración de las variables en el bloque LSim_PT1

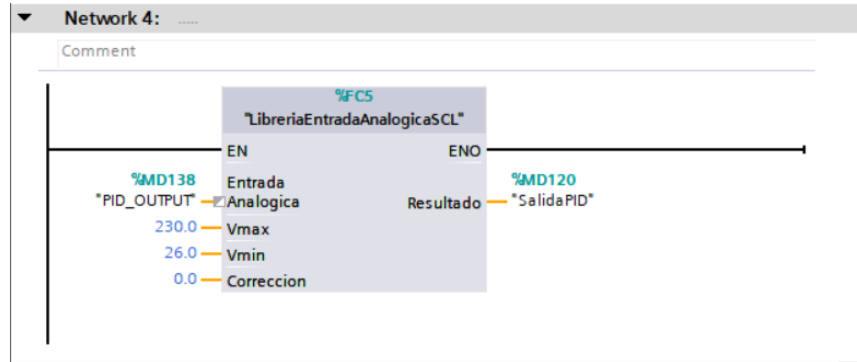


Nota. Valores empleados en el bloque LSim para la simulación del proceso.

5.6.3 Ajuste y escalado de señales en el lazo de control PID

Para mantener la sincronización del proceso, se añadió el bloque “LibreriaEntradaAnalogicaSCL %FC5” al mismo bloque de interrupción cíclica (OB). Esto aseguró que la señal se procesara de forma uniforme en cada ciclo de control. Este bloque, tomado de la librería de funciones, ajustó la señal de salida del controlador PID. La variable “PID_OUTPUT %MD138” se conectó como entrada analógica y la señal escalada se envió a “SalidaPID %MD120”, utilizada en la simulación del proceso térmico. Al configurar el sistema, se establecieron límites de 26 y 230 como valores mínimos y máximos permitidos. Este rango fue seguro y reflejó el funcionamiento real de la plancha. Los límites previnieron saturaciones y valores fuera de rango y aseguraron que la señal aplicada al modelo térmico y al lazo de control se mantuviera dentro de condiciones estables y realistas, ver figura 48.

Figura 48
Ajuste del rango de la señal PID



Nota. Bloque encargado de ajustar la señal de salida del controlador.

5.7 Diseño y estructura de la Interfaz Hombre-Máquina (HMI)

Para el desarrollo de la interfaz HMI, se incorporaron desde el PLC las variables previamente definidas, las cuales resultaron indispensables para garantizar el correcto funcionamiento de la pantalla y la interacción directa entre el operador y el sistema automatizado. Estas variables permitieron el intercambio de información en tiempo real, asegurando que los estados, valores y acciones del proceso se visualizaran de forma inmediata en la HMI, ver figura 49.

Figura 49
Primera sección de variables para el HMI

Name	Data type	Connection	PLC name
ACT_CALCETINES	Bool	HMI_Conne...	PLC_1
ACT_ROPA_INF	Bool	HMI_Connectio...	PLC_1
ACT_ROPA_SUP	Bool	HMI_Connectio...	PLC_1
ActualTemp	Real	HMI_Connectio...	PLC_1
Almacen	Int	HMI_Connectio...	PLC_1
DATA_NODERED_ActualTemp	Real	HMI_Connectio...	PLC_1
DATO_ALEATORIO	Int	HMI_Connectio...	PLC_1
Datos_NODERED.NUM_CAL	Int	HMI_Connectio...	PLC_1
Datos_NODERED.NUM_ROPA_INF	Int	HMI_Connectio...	PLC_1
Datos_NODERED.NUM_ROPA_S...	Int	HMI_Connectio...	PLC_1
Datos_NODERED.PID_Setpoint	Real	HMI_Connectio...	PLC_1
Datos_NODERED.REGISTRO_OB...	Int	HMI_Connectio...	PLC_1
Datos_NODERED.SetPointTemp	Int	HMI_Connectio...	PLC_1
Datos_NODERED_CAJA_CALCE...	Int	HMI_Connectio...	PLC_1
Datos_NODERED_CAJA_ROPA_I...	Int	HMI_Connectio...	PLC_1
Datos_NODERED_CAJA_ROPA_...	Int	HMI_Connectio...	PLC_1
FINAL DE CARRERA	Bool	HMI_Connectio...	PLC_1
INICIO	Bool	HMI_Connectio...	PLC_1
MOV_1TERA	Int	HMI_Connectio...	PLC_1
MOV_ACT_1	Int	HMI_Connectio...	PLC_1
MOV_ACT_2	Int	HMI_Connectio...	PLC_1
MOV_ACT_3	Int	HMI_Connectio...	PLC_1
MOV_CAL	Int	HMI_Connectio...	PLC_1
MOV_CALCETIN	Int	HMI_Connectio...	PLC_1
MOV_INF	Int	HMI_Connectio...	PLC_1

Nota. Variables que se emplean en el HMI.

En la figura 50, se describieron las variables en la HMI, las cuales cumplieron la función de enlazar el sistema de control con el operador, permitiendo la visualización, supervisión y ajuste de los distintos parámetros del proceso de manera intuitiva. A través de estas variables, la interfaz recibió información en tiempo real desde el PLC y, a su vez, envió órdenes que permitieron iniciar, detener o modificar el funcionamiento del equipo.

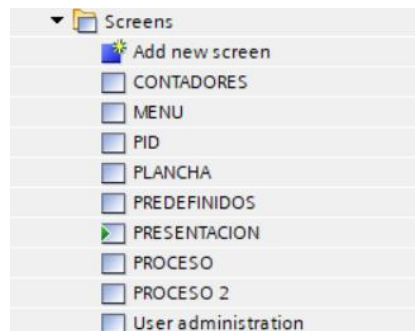
Figura 50
Segunda sección de variables del HMI

	MOV_ROPA_INF	Int	HMI_Connectio...	PLC_1
	MOV_ROPA_SUP	Int	HMI_Connectio...	PLC_1
	MOV_SUPERIOR	Int	HMI_Connectio...	PLC_1
	Nivel_Maximo	Int	HMI_Connectio...	PLC_1
	NIVEL_MEDIDOR	Int	HMI_Connectio...	PLC_1
	Nivel_Minimo	Int	HMI_Connectio...	PLC_1
	NUM_ROP_PLANCH	Int	HMI_Connectio...	PLC_1
	PARO	Bool	HMI_Connectio...	PLC_1
	PAUSE	Bool	HMI_Connectio...	PLC_1
	PID_Setpoint	Real	HMI_Connectio...	PLC_1
	PLANCHA_OFF	Bool	HMI_Connectio...	PLC_1
	PLANCHA_ON	Bool	HMI_Connectio...	PLC_1
	PLANCHA1_ON	Bool	HMI_Connectio...	PLC_1
	RANDOM	Bool	HMI_Connectio...	PLC_1
	Recetas.LoadRecipe	Int	HMI_Connectio...	PLC_1
	Recetas.Nombre	String	HMI_Connectio...	PLC_1
	Recetas.Registro	String	HMI_Connectio...	PLC_1
	Recetas.Ropa.TipoRopa	Real	HMI_Connectio...	PLC_1
	REINICIO	Bool	HMI_Connectio...	PLC_1
	ROPA_INFERIOR	Bool	HMI_Connectio...	PLC_1
	SetPointLimit	Int	HMI_Connectio...	PLC_1
	SetPointTemp	Int	HMI_Connectio...	PLC_1
	SYSTEM_ON	Bool	HMI_Connectio...	PLC_1
	Tag_ScreenNumber	UInt	<Internal tag>	
	UserAdmin	WString	<Internal tag>	

Nota. La figura enseña las demás variables del HMI que se utilizaron.

Dentro de la carpeta Screens del HMI se encontraron un total de 9 pantallas diseñadas para la visualización del proceso, las cuales permitieron organizar la información y las funciones de forma clara, ver figura 51.

Figura 51
Pantallas diseñadas para mostrar el proceso



Nota. Pantallas que albergan el proceso de clasificación.

A continuación, se debió ir a la parte de Administrador de Usuarios, la cual permitió establecer niveles de acceso para tener un uso seguro y controlado del sistema. Se definieron perfiles de usuario con credenciales propias, como el administrador y el operador, cada uno con permisos específicos según su rol, tal como se observó en la figura 52.

Figura 52
Administrador de usuarios

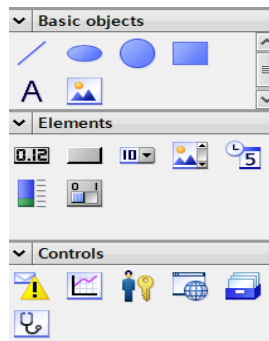
Users						
	Name	Password	Automatic logoff	Logoff time	Number	Comment
	Admin	*****	<input checked="" type="checkbox"/>	5	1	The user 'Admin...
	Operator	*****	<input checked="" type="checkbox"/>	5	2	
	<Add new>					

Groups						
	Member of	Name	Number	Display name	Password aging	Comment
	<input checked="" type="radio"/>	Administrator group	1	Administrator group	<input type="checkbox"/>	The 'Administrato...
	<input type="radio"/>	Operator	2	Users	<input type="checkbox"/>	The 'Users' group .
		<Add new>				

Nota. Se observa la configuración de la administración de usuarios del HMI.

En la figura 53, se presentó el panel de objetos del HMI, el cual agrupó diversas herramientas de diseño tales como botones, barras, campos de datos y alarmas. Estos componentes permitieron al operario interactuar con el sistema, realizar ajustes manuales y supervisar de manera gráfica las variables asociadas al proceso de clasificación de prendas.

Figura 53
Panel de objetos para editar el HMI

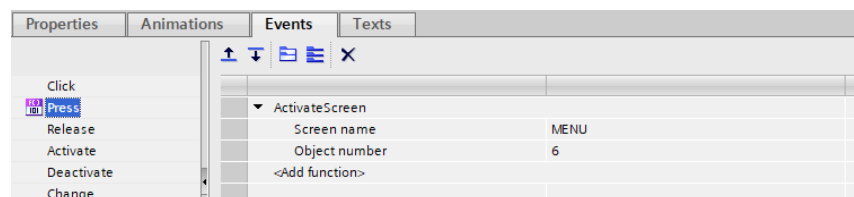


Nota. Panel que permite diseñar las interfaces del proceso.

5.7.1 Diseño y estructura de la pantalla “PRESENTACIÓN”

En la figura 54, se detalló la configuración del botón en la pantalla “PRESENTACIÓN” de la interfaz HMI, el cual permitió el acceso al menú principal del sistema. El botón se configuró mediante la asignación del evento “Press”, que ejecutó la función “ActivateScreen”, la cual vinculó la pantalla de destino utilizando el parámetro “Screen name” y el campo “Object number” con su identificador correspondiente. Esta configuración aseguró un cambio de pantalla preciso, optimizó la experiencia del usuario y facilitó la navegación entre las diferentes secciones del sistema.

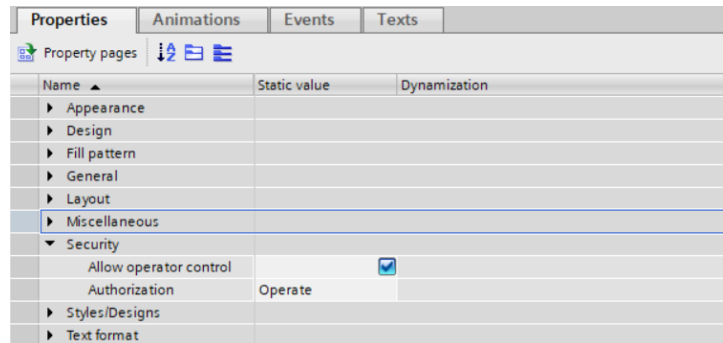
Figura 54
Configuración del botón de menú



Nota. Configuración de un botón.

Se configuró la seguridad del botón mediante la asignación de permisos en la sección *Security*, figura 55, donde se definió el nivel de autorización requerido. Cuando un usuario no autorizado intentó utilizar el botón, el sistema solicitó la autenticación correspondiente. En la figura 56, se abrió la ventana de inicio de sesión del HMI, la cual solicitó el nombre de usuario y la contraseña para validar el acceso. De esta manera, solo los usuarios autorizados pudieron acceder a la navegación y a las funciones de la interfaz.

Figura 55
Configuración de seguridad del botón en la interfaz HMI



Nota. Asignación de permisos del botón en la sección *Security*.

Figura 56
Ventana de inicio de sesión del sistema HMI



Nota. Ventana de inicio de sesión para la validación de acceso del usuario.

En la figura 57, se describió la pantalla de presentación, en la cual se observó un fondo con colores representativos de la universidad, así como el logotipo de la institución, el título del proyecto, los nombres de los autores y el nombre del tutor. En la parte inferior derecha se ubicó el botón “MENÚ”, el cual permitió el acceso al resto de las pantallas de la interfaz.

Figura 57
Pantalla de presentación del proyecto



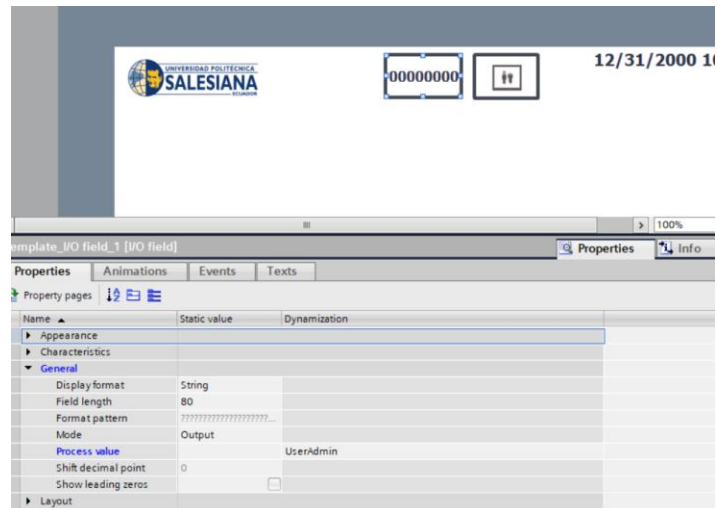
Nota. Se aprecia en la imagen la interfaz de presentación.

5.7.2 Diseño de la interfaz de “USER ADMINISTRATION”

Se implementó el control de acceso en la interfaz HMI utilizando un *template* del módulo “Screen Management”. En la figura 58, se describió un campo de salida configurado como indicador del usuario activo, el cual visualizó inicialmente un valor por defecto y, una vez iniciada la sesión con privilegios de administrador, actualizó su contenido con el nombre del usuario que operó el sistema. En la figura 59 se detalló la configuración del botón de inicio de sesión dentro del mismo *template*, donde la función *ShowLogonDialog* se asignó al evento “Release” para desplegar la ventana de autenticación al liberar el botón. El uso del *template* permitió identificar al usuario en operación y aseguró un acceso controlado y uniforme en todas las pantallas del sistema.

Figura 58

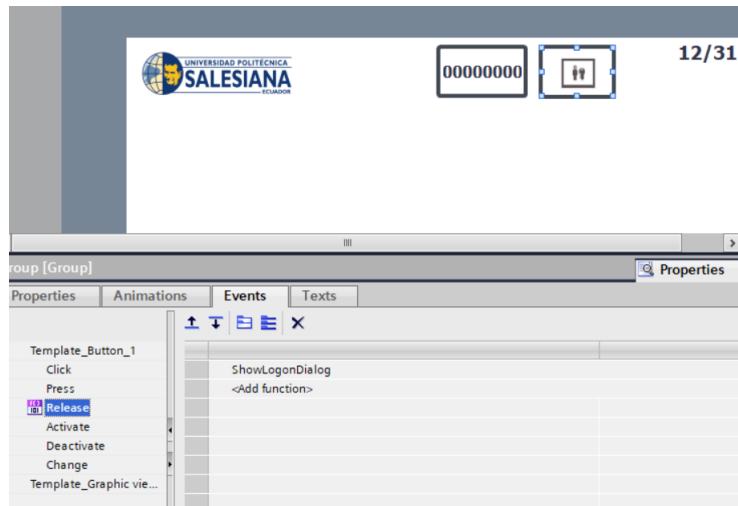
Visualización del estado de usuario en la plantilla del HMI



Nota. Indicador HMI del usuario autenticado mediante el sistema de administración de usuarios.

Figura 59

Configuración del botón de inicio de sesión en la plantilla del HMI

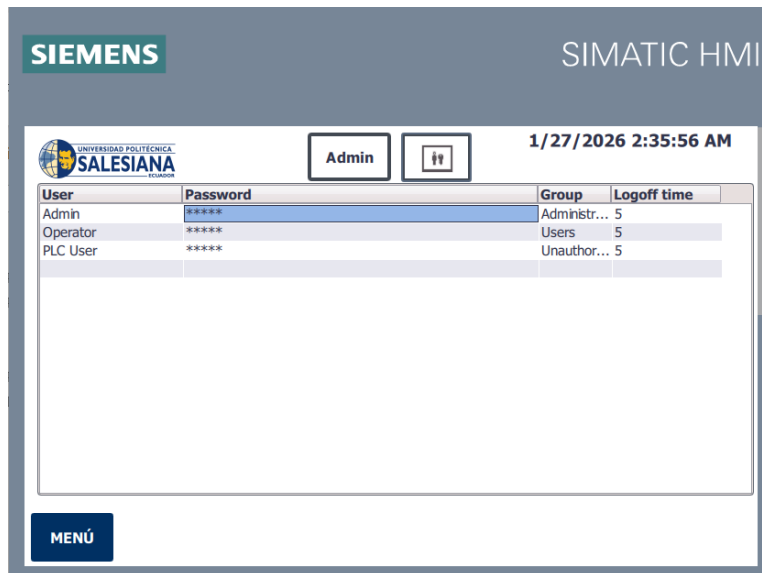


Nota. Asignación de la función *ShowLogonDialog* para habilitar el acceso mediante usuario y contraseña.

Esta pantalla incorporó un *User View*, figura 60, que permitió visualizar y administrar los perfiles habilitados en el sistema HMI, facilitando el control de permisos para pantallas, procesos y variables. La tabla permitió identificar a los usuarios registrados, sus grupos y el tiempo de cierre de sesión configurado, lo que reforzó la seguridad de los accesos.

Además, el botón “MENÚ” facilitó la navegación entre pantallas, asegurando un acceso organizado y controlado a las funciones disponibles.

Figura 60
Pantalla de administrador de usuarios

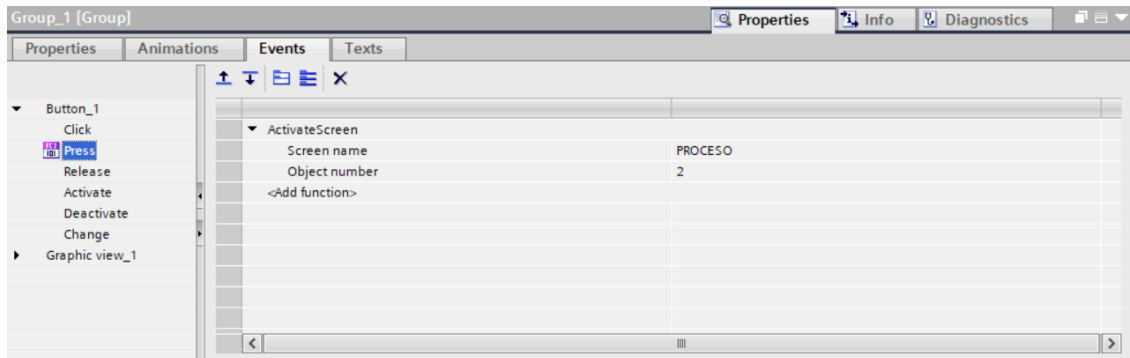


Nota. Pantalla de administración de usuarios.

5.7.3 Diseño y estructura de la interfaz “MENÚ”

Durante el diseño de la interfaz “MENÚ”, se añadió un botón de navegación para acceder a la pantalla de PROCESO. Este botón, tomado de la librería de elementos, se configuró en la sección “Events” mediante la asignación de la función “ActivateScreen” al evento “Press”. Se definió la pantalla de destino a través del parámetro “Screen name” y se asignó el número 2 en “Object number”, correspondiente al identificador interno de la pantalla de proceso. Esta configuración garantizó un cambio de pantalla controlado y coherente, ver figura 61.

Figura 61
Configuración del botón de navegación hacia la pantalla "PROCESO"



Nota. Asignación del evento Press y la función ActivateScreen para la conmutación entre pantallas.

En la figura 62, se describió la interfaz de menú, la cual incorporó botones que permitieron acceder a cada sección del sistema, facilitando una navegación clara y ordenada. De esta manera, el usuario pudo acceder a cada interfaz de forma simple e intuitiva. Para lograrlo, se utilizaron botones con la misma configuración presentada en la figura 61, modificando únicamente los parámetros *Screen Name* y *Object Number* según la pantalla de destino.

Figura 62
Pantalla del menú del proceso



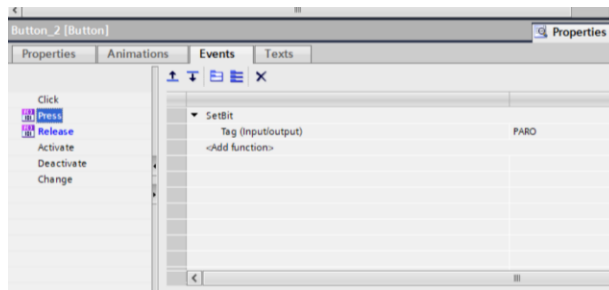
Nota. Menú que direcciona a cada una de las interfaces.

5.7.4 Diseño y estructura de la pantalla “PROCESO 1”

En el diseño de la pantalla de PROCESO se configuraron los controles y señales del sistema. En la figura 63 se detalló la implementación del botón RESET, tomado de la librería de objetos y configurado en la sección *Events*. Al evento *Press* se le asignó la función *SetBit*, vinculada a la variable “PARO %M0.1”, lo que permitió restablecer el proceso durante la operación. Además, se incorporó un indicador visual verde en forma de círculo gráfico, figura 64, visible únicamente cuando la variable “ACT_ROPA_SUP %M33.4” se encontró activa. Esta integración mejoró la supervisión y el manejo seguro del proceso desde la interfaz HMI.

Figura 63

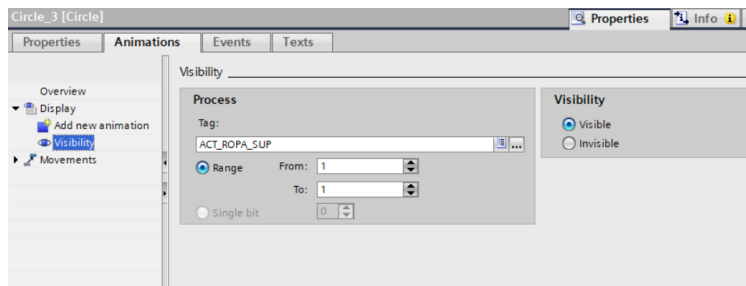
Configuración del botón de RESET en la pantalla de proceso



Nota. Asignación de la función *SetBit* al evento *Press* para el restablecimiento del estado del sistema.

Figura 64

Configuración del indicador visual de estado en la pantalla de proceso



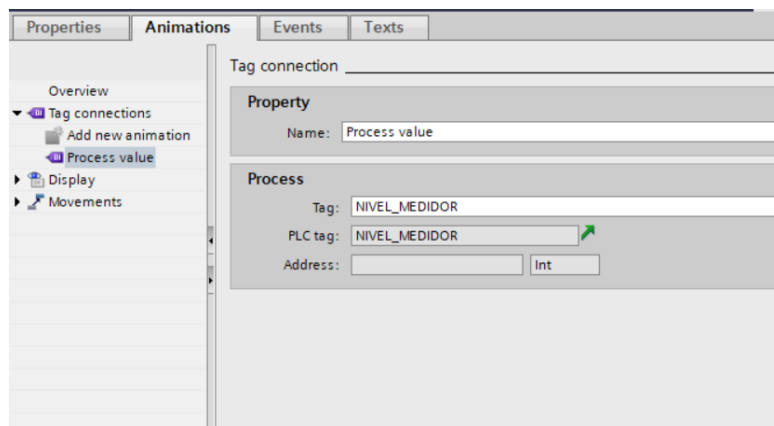
Nota. Indicador de estado por medio animación de visibilidad.

Como parte del diseño de la pantalla de PROCESO, se configuraron indicadores vinculados a variables del sistema para mostrar información relevante sobre su

funcionamiento. En la figura 65, se describió un indicador de nivel tipo barra, incorporado mediante objetos gráficos del entorno HMI y vinculado a la variable *NIVEL_MEDIDOR* a través de la opción *Process value*. Esto permitió simular el nivel de almacenamiento de la canasta de prendas durante la operación. En cambio, la configuración de un indicador de campo de entrada/salida, figura 66, se vinculó a la variable “DATO_ALEATORIO” para visualizar los valores generados de forma aleatoria que representaron la detección del tipo de prenda clasificada. Ambas configuraciones se integraron en la interfaz HMI, lo que facilitó una visualización clara del nivel de prendas.

Figura 65

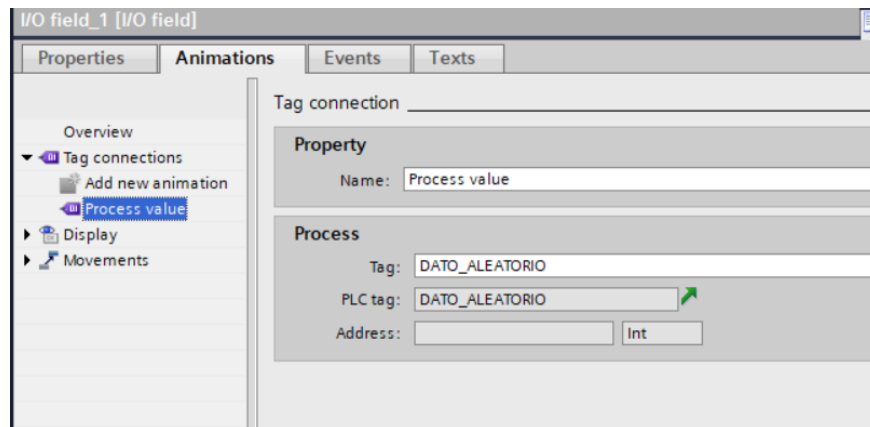
Configuración del indicador de nivel de la canasta de prendas



Nota. Indicador tipo barra vinculado a la variable *NIVEL_MEDIDOR*.

Figura 66

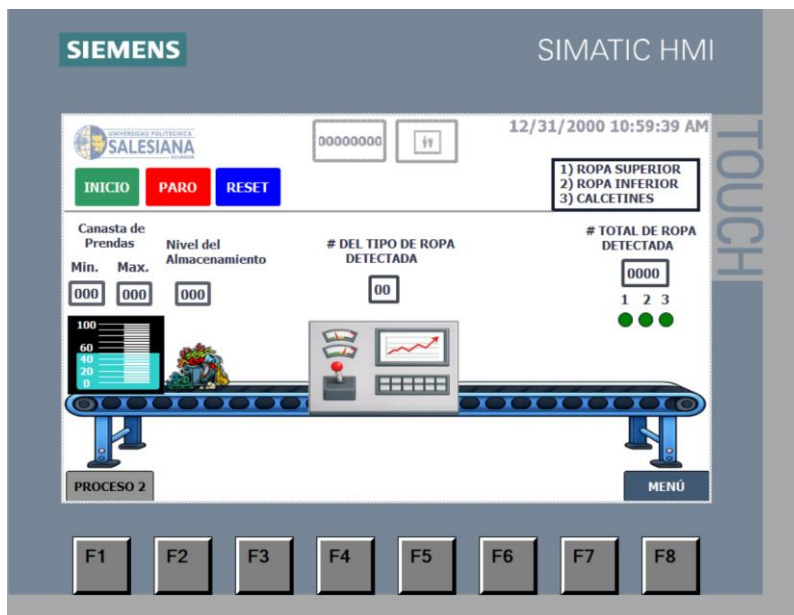
Configuración del indicador de valores aleatorios del proceso



Nota. Campo de salida asociada a la variable *DATO_ALEATORIO*.

En la figura 67, se describió el proceso de clasificación de productos textiles, en el cual se integraron elementos gráficos, tales como las bandas transportadoras en formato .PNG, que simularon el movimiento de las prendas hacia el módulo donde se ubicó una cámara encargada de identificar el tipo de prenda que ingresó al sistema, así como un indicador visual del nivel de almacenamiento, representado por una barra que permitió visualizar la cantidad de prendas disponibles dentro de la canasta. Se añadieron campos para modificar y visualizar los niveles mínimo y máximo de almacenamiento. La pantalla también incorporó textos con sus respectivos campos, los cuales informaron sobre la cantidad de prendas detectadas por tipo y sobre el total procesado. Finalmente, se añadieron los botones “INICIO”, “PARO” y “RESET” para el control del proceso, así como accesos a otras pantallas mediante los botones “MENÚ” y “PROCESO 2”, los cuales fueron descritos previamente.

Figura 67
Pantalla de detección de la prenda

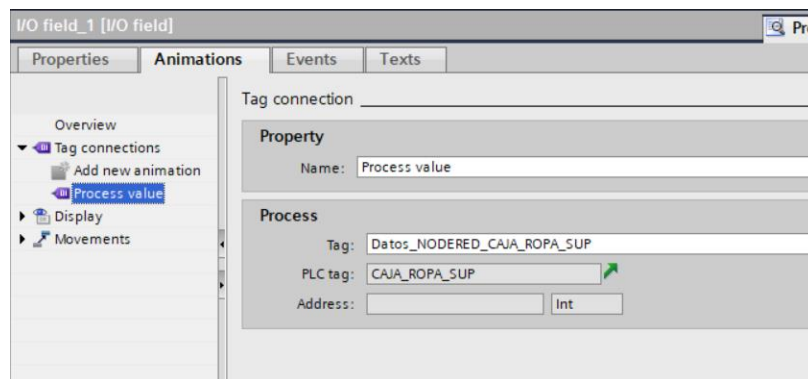


Nota. Detección de los productos textiles.

5.7.5 Diseño y estructura de la interfaz “PROCESO 2”

Se presentó un indicador numérico que permitió visualizar la cantidad de ropa superior almacenada en la caja asignada, figura 68. Este indicador se configuró con un campo de salida y se conectó, en la sección *Process value*, a la variable “Datos_NODERED_CAJA_ROPA_SUP %DB1.DBW18”. De esta manera, el valor se actualizó automáticamente cuando las prendas fueron detectadas y clasificadas en la banda transportadora superior. En la figura 69, se detallaron los parámetros generales del indicador, tales como el formato decimal, un campo de cuatro dígitos, el patrón 9999 y el modo de operación *Output*. Estas configuraciones aseguraron que el conteo de prendas se visualizara con precisión en la interfaz HMI.

Figura 68
Indicador de conteo de prendas de ropa superior



Nota. Campo de salida vinculado a la variable *CAJA_ROPA_SUP* para la visualización del conteo de prendas clasificadas.

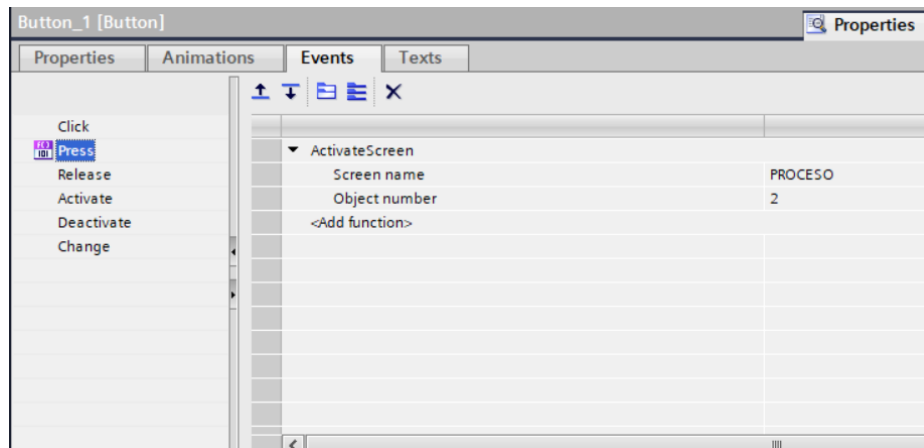
Figura 69
Configuración de los parámetros de visualización del indicador de conteo

General		
Display format	Decimal	
Field length	4	
Format pattern	9999	
Mode	Output	
Process value		*Datos_NODERED.NUM_ROPA_SUP*
Shift decimal point	0	
Show leading zeros		<input type="checkbox"/>

Nota. Ajustes de formato, longitud de campo y modo de salida del indicador numérico.

En la figura 70, se agregó un botón desde la librería de elementos y se configuró su función en la sección Events. En el evento “Press” se asignó la función “ActivateScreen”, la cual definió la pantalla de destino mediante el parámetro “Screen name”, configurado como “PROCESO”, junto con su identificador interno. De esta manera, se permitió cambiar de forma sencilla entre la interfaz de contadores y la pantalla del proceso, lo que facilitó la navegación dentro del sistema.

Figura 70
Configuración del botón de navegación hacia Proceso 1

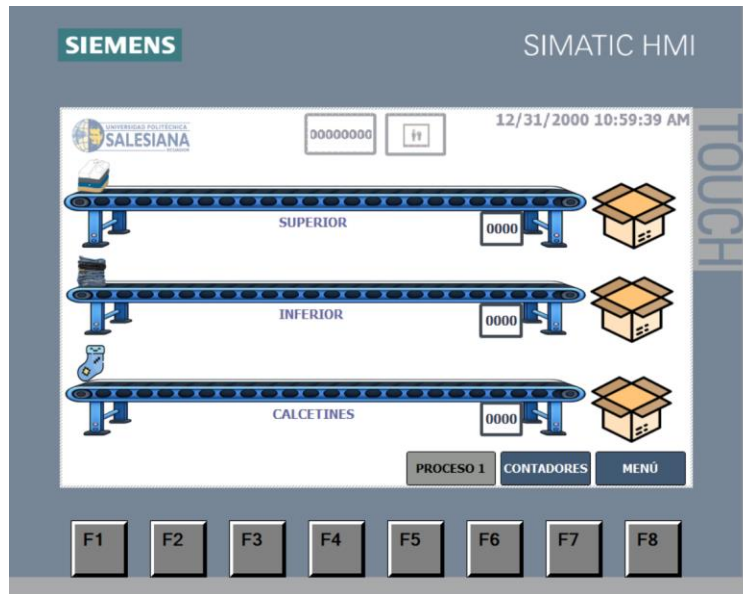


Nota. Definición de la función de cambio de pantalla en el evento Press.

En esta pantalla se representaron las bandas transportadoras finales del proceso de clasificación, utilizando imágenes gráficas similares a las empleadas en la pantalla de “PROCESO 1”. El objetivo fue mostrar de manera visual el desplazamiento de las prendas superiores, inferiores y calcetines hacia sus respectivas cajas de almacenamiento. Cada banda contó con un campo numérico donde se visualizó la cantidad de prendas clasificadas, mientras que las cajas gráficas ubicadas al final de cada línea simulaban el lugar donde se depositaron las prendas al finalizar el proceso. El uso de textos facilitó la identificación de cada sección, brindando una orientación clara al operador sobre la función de cada banda.

Asimismo, se incorporaron los botones “MENÚ”, “PROCESO 1” y “CONTADORES”, los cuales permitieron la navegación entre las distintas pantallas del sistema, ver figura 71.

Figura 71
Pantalla de la clasificación de la ropa

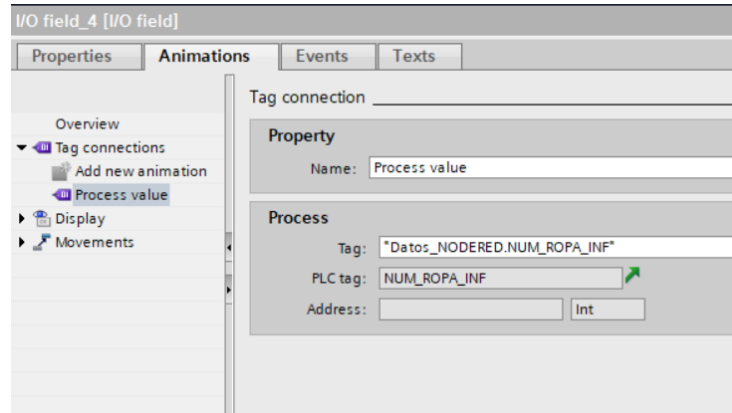


Nota. Interfaz que corresponde a la clasificación de las prendas.

5.7.6 Diseño y estructura de la pantalla “CONTADORES”

En la pantalla CONTADORES se agregaron indicadores numéricos que permitieron visualizar la cantidad de prendas clasificadas por tipo. En la figura 72, la configuración de un campo de salida se conectó al conteo de ropa inferior mediante la variable “Datos_NODERED.NUM_ROPA_INF %DB1.DBW2”. De esta manera, se pudo visualizar de forma automática el número de prendas registradas durante el proceso de clasificación.

Figura 72
Configuración del indicador de conteo de ropa inferior



Nota. Asignación del valor de proceso del campo de salida para la visualización del conteo de prendas.

En la figura 73, se presentó la visualización de los contadores del proceso, utilizando elementos gráficos e imágenes representativas para presentar de forma clara y comprensible la cantidad de prendas procesadas. Cada sección correspondió a un tipo específico de prenda, lo que permitió identificar tanto el número total de prendas registradas como el conteo individual de ropa superior, ropa inferior y calcetines, presentados en campos numéricos que se actualizaron durante el proceso. La inclusión de imágenes facilitó la interpretación visual por parte del operador y contribuyó a una interfaz más intuitiva. Asimismo, se integraron botones de navegación, como “PROCESO 1”, “PROCESO 2” y “MENÚ”, que permitieron el desplazamiento entre las diferentes pantallas del HMI.

Figura 73
Pantalla de contadores de las prendas



Nota. Interfaz que lleva la cuenta de las prendas clasificadas.

5.7.7 Diseño y estructura de la pantalla “PLANCHA”

En la pantalla de control de la plancha se agregó un indicador de nivel de temperatura que representó de forma gráfica la temperatura real del proceso. Para su implementación, se arrastró el indicador de nivel vertical desde la librería de objetos gráficos del entorno HMI hasta la pantalla. Posteriormente, en la sección *General*, se configuró el valor de proceso y se vinculó a la señal de temperatura real del sistema. Asimismo, se establecieron los límites de operación, con un valor mínimo de 26 °C y un máximo de 230 °C, con el fin de reflejar un rango seguro y acorde con las condiciones reales de la plancha, ver figura 74.

Figura 74
Configuración del indicador de nivel de temperatura de la plancha

▼ General		
Maximum value	230	
Minimum value	26	
Process value	0	DATA_NODERED_ActualTemp
▶ Label		
▶ Layout		
▶ Limits/Ranges		
▶ Miscellaneous		

Nota. Indicador de nivel vertical configurado para la visualización de la temperatura real del proceso.

En la figura 75, se integró un indicador numérico que fue incorporado desde la sección *Elements* y permitió visualizar la cantidad de prendas planchadas durante la operación. Para su configuración, se incorporó un objeto de tipo campo de entrada/salida, el cual se ajustó en la sección *General* con formato de visualización decimal, una longitud de campo de tres dígitos, un patrón de formato 999 y un modo de funcionamiento *Output*.

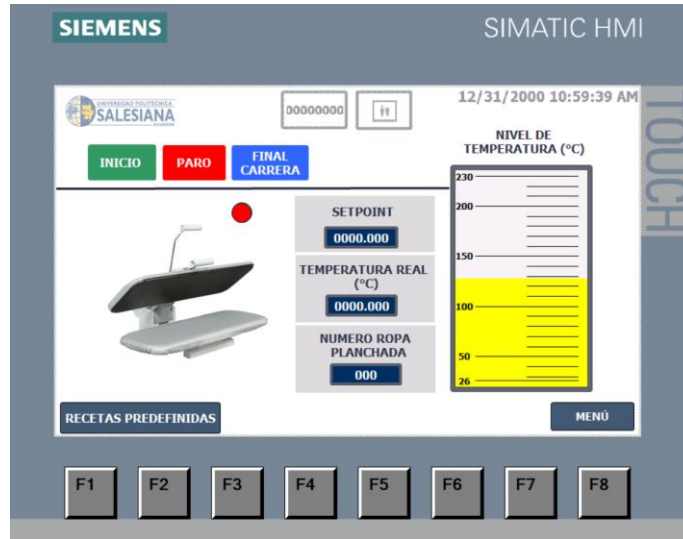
Figura 75
Configuración del indicador numérico de conteo de prendas planchadas

▼ General		
Display format	Decimal	
Field length	3	
Format pattern	999	
Mode	Output	
Process value		NUM_ROP_PLANCH
Shift decimal point	0	
Show leading zeros		<input type="checkbox"/>

Nota. Indicador numérico configurado para la visualización del conteo de ropa procesada.

Esta pantalla permitió el control del proceso de planchado mediante la integración de elementos gráficos y campos de información que mostraron los valores de temperatura, junto con una imagen representativa de la plancha que ilustró su funcionamiento. La interfaz incluyó los botones “INICIO”, “PARO” y “FINAL DE CARRERA”, este último simuló el cierre manual de la plancha durante el proceso y registró el número de veces que se accionó el mecanismo de cierre. Para el control directo del sistema, se incorporó una barra gráfica que indicó el nivel de temperatura y el rango térmico de operación. Además, se añadieron accesos a “RECETAS PREDEFINIDAS” y al “MENÚ”, lo que facilitó la navegación entre pantallas y la selección de parámetros según el tipo de prenda, ver figura 76.

Figura 76
Interfaz del proceso de planchado



Nota. Pantalla que permite el planchado de los textiles mediante recetas.

5.7.8 Diseño y estructura de la interfaz “RECETAS PREDEFINIDAS”

Para la gestión de recetas predefinidas, el objeto de recetas se configuró con los parámetros descritos en la figura 77. En la sección General, se habilitaron las opciones Allow editing, que permitió modificar los valores de cada receta desde la HMI, y Display selection list, que facilitó la selección entre las recetas disponibles. Asimismo, se definió el Tag for data record como “Recetas.Registro” para almacenar los valores de cada receta, y el Tag for recipe number como “Recetas.Nombre” para identificar la receta seleccionada. Esta configuración permitió que la gestión de recetas fuera estructurada y flexible.

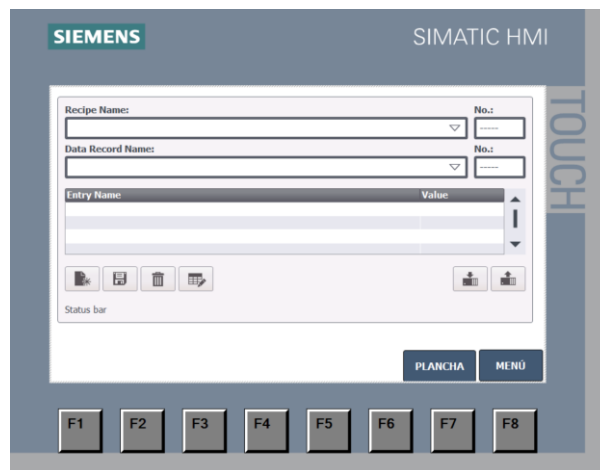
Figura 77
Configuración interna de las recetas predefinidas en el HMI

General	
Allow editing	<input checked="" type="checkbox"/>
Display selection list	<input checked="" type="checkbox"/>
Recipe	
Tag for data record	"Recetas.Registro"
Tag for recipe number	"Recetas.Nombre"

Nota. Ajustes generales para la edición, selección y registro de recetas del sistema.

Se incorporó el componente Recipes View a la interfaz HMI, figura 78, accediendo al panel Toolbox ubicado en el lateral derecho de la pantalla. El objeto se encontró en la ruta *Controls* y fue arrastrado directamente a la pantalla *PREDEFINIDOS*. Una vez colocado, se configuró a través del panel de propiedades, donde se definieron los parámetros necesarios para la gestión y visualización de las recetas del proceso de planchado. Asimismo, se incluyeron los botones “PLANCHA” y “MENÚ”, cuya configuración fue explicada en pantallas anteriores, los cuales permitieron regresar al control de planchado o al menú principal.

Figura 78
Pantalla de recetas para el planchado



Nota. Pantalla que contiene las recetas para el planchado de la ropa.

5.7.9 Diseño y estructura de la pantalla “PID”

En la figura 79, la configuración del *trend* correspondiente a la temperatura real (*ACTUALTEMP*), se ajustó con un ancho de barra de 50 y un muestreo cíclico en tiempo real para la actualización continua de los datos. Se establecieron límites visuales mediante la asignación de colores, utilizando rojo para el límite alto y amarillo para el límite bajo, lo que permitió identificar con claridad el comportamiento de la variable durante el proceso. El valor se visualizó en el lado izquierdo del gráfico, con línea sólida y visualización

interpolada. En cambio, en la figura 80, se detalló la configuración del *trend* asociado al valor de referencia (*SETPOINT*), el cual se ajustó con parámetros similares de ancho, ciclo y tipo de tendencia para garantizar la coherencia visual entre ambas señales. En este caso, se utilizó un color de primer plano distinto para diferenciar el *setpoint* de la variable real, lo que permitió comparar gráficamente ambas señales dentro del sistema de control PID y facilitó la supervisión del desempeño del lazo de control desde la interfaz HMI.

Figura 79
Configuración del trend de la temperatura real "ACTUALTEMP"

▼ Trend		
▼ Trend	SETPOINT, ACTUAL...	
▼ ACTUALTEMP		
Bar width	50	
Color high limit	■ 255, 0, 0	
Color low limit	■ 255, 255, 0	
Cyclic	1	
Foreground color	■ 255, 0, 0	
Limit value line	<input type="checkbox"/>	
Line style	Solid	
Logging tag		
Name	ACTUALTEMP	
Side	Left	
Status color	■ 0, 0, 255	
Trend display mo...	Interpolated	
Trend tag	DATA_NODERED_A...	
Trend type	Cyclical real time	
Trend values	100	

Nota. Ajuste de los parámetros de visualización del valor real de temperatura en tiempo real.

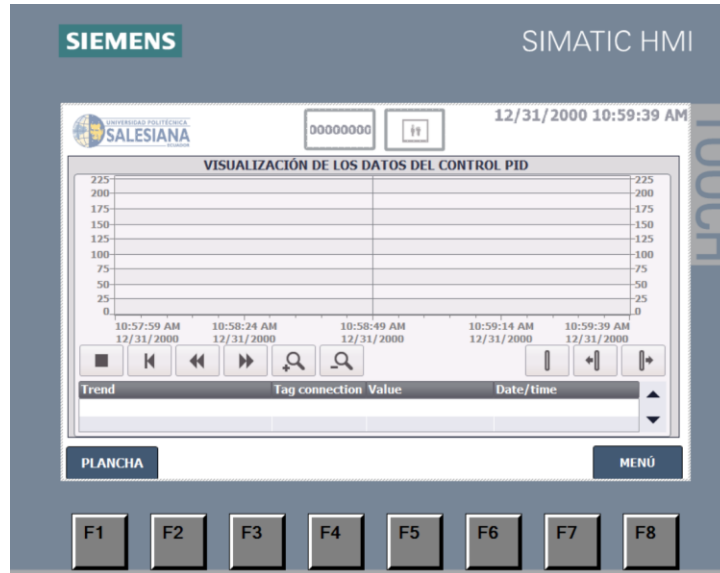
Figura 80
Configuración del trend del valor de referencia del PID "SETPOINT"

▼ SETPOINT		
Bar width	50	
Color high limit	■ 255, 0, 0	
Color low limit	■ 255, 255, 0	
Cyclic	1	
Foreground color	■ 0, 0, 0	
Limit value line	<input type="checkbox"/>	
Line style	Solid	
Logging tag		
Name	SETPOINT	
Side	Left	
Status color	■ 0, 0, 255	
Trend display mo...	Interpolated	
Trend tag	*Datos_NODERED.PI..	
Trend type	Cyclical real time	
Trend values	100	

Nota. Configuración del trend correspondiente al valor de referencia para la comparación con la temperatura real.

Se utilizó un *Trend View* para observar en tiempo real el comportamiento del control PID, mostrando la evolución del *setpoint* y de la temperatura actual, y contando con los botones “PLANCHA” y “MENÚ” para la navegación entre pantallas, ver figura 81.

Figura 81
Pantalla para la visualización del control PID



Nota. La gráfica corresponde a la pantalla del control PID.

5.8 Desarrollo y configuración del Dashboard en Node-Red

Para la implementación de la interfaz se creó un bloque de datos (*DB*) específico, figura 82, en el cual fue necesario deshabilitar la opción *Optimized block access* con el fin de asegurar la correcta comunicación y el direccionamiento de las variables. En este bloque se incorporaron todas las variables requeridas para el intercambio de información con la interfaz de Node-RED, lo que permitió la visualización y el registro adecuado de los datos del proceso.

Figura 82

Data block que almacena las variables para el Node-red

Datos_NODERED										
	Name	Data type	Offset	Start value	Retain	Accessible f...	Writa...	Visible in ...	Setpoint	
1	Static				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
2	NUM_CAL	Int	0.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
3	NUM_ROPA_INF	Int	2.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
4	NUM_ROPA_SUP	Int	4.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
5	REGISTRO_OBJETOS	Int	6.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
6	SetPointTemp	Int	8.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
7	PID_Setpoint	Real	10.0	0.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
8	ActualTemp	Real	14.0	26.0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
9	CAJA_ROPA_SUP	Int	18.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
10	CAJA_ROPA_INF	Int	20.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
11	CAJA_CALCETINES	Int	22.0	0	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Nota. Variables del Node-red.

En la figura 83, la instalación de Node-RED se realizó a través del comando `npm install -g --unsafe-perm node-red`, otorgando los permisos necesarios para completar el proceso. Una vez finalizada la instalación, se verificó su correcta ejecución mediante los comandos `node --version` y `npm --version`, los cuales permitieron comprobar las versiones correspondientes de cada herramienta. Posteriormente, el acceso a la interfaz de Node-RED se efectuó desde un navegador web utilizando la dirección IP asignada al módulo IOT2040 junto con el puerto 1880, por ejemplo, `http://<dirección-ip>:1880/`. Asimismo, se verificó que los nodos y flujos se ejecutaron directamente en el IOT2040, mientras que el equipo informático actuó únicamente como medio de acceso y visualización.

Figura 83

Inicio del Node-Red mediante comandos en el CMD

```
Microsoft Windows [Versión 10.0.26200.7628]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\Leonardo>node-red
26 Jan 20:45:20 - [info]

Welcome to Node-RED
=====

26 Jan 20:45:20 - [info] Node-RED version: v4.1.3
26 Jan 20:45:20 - [info] Node.js version: v24.12.0
26 Jan 20:45:20 - [info] Windows_NT 10.0.26200 x64 LE
26 Jan 20:45:21 - [info] Loading palette nodes
26 Jan 20:45:23 - [info] Dashboard version 3.6.6 started at /ui
26 Jan 20:45:23 - [info] Settings file : C:\Users\Leonardo\.node-red\settings.js
26 Jan 20:45:23 - [info] Context store : 'default' [module=memory]
26 Jan 20:45:23 - [info] User directory : \Users\Leonardo\.node-red
26 Jan 20:45:23 - [warn] Projects disabled : editorTheme.projects.enabled=false
26 Jan 20:45:23 - [info] Flows file : \Users\Leonardo\.node-red\flows.json
26 Jan 20:45:23 - [warn]

-----
Your flow credentials file is encrypted using a system-generated key.

If the system-generated key is lost for any reason, your credentials
file will not be recoverable, you will have to delete it and re-enter
your credentials.

You should set your own key using the 'credentialSecret' option in
your settings file. Node-RED will then re-encrypt your credentials
file using your chosen key the next time you deploy a change.
-----

26 Jan 20:45:23 - [warn] Encrypted credentials not found
26 Jan 20:45:23 - [info] Server now running at http://127.0.0.1:1880/
```

Nota. Acceso al Node-Red utilizando la ip de la IOT2040.

Se realizó la comprobación de la conectividad con Node-RED mediante la ejecución del comando *ping* desde una nueva ventana del CMD, figura 84, utilizando la dirección IP configurada para la comunicación. Los resultados obtenidos indicaron el envío y la recepción correcta de cuatro paquetes ICMP, sin pérdidas, con tiempos de respuesta inferiores a 1 ms y un valor de *TTL* igual a 128. Estos valores confirmaron que la comunicación local fue estable y que la ruta de conexión utilizada funcionó de manera adecuada.

Figura 84

Verificación de conectividad mediante comando ping

```
C:\Users\Leonardo>ping 127.0.0.1

Haciendo ping a 127.0.0.1 con 32 bytes de datos:
Respuesta desde 127.0.0.1: bytes=32 tiempo<1m TTL=128
Respuesta desde 127.0.0.1: bytes=32 tiempo<1m TTL=128
Respuesta desde 127.0.0.1: bytes=32 tiempo<1m TTL=128
Respuesta desde 127.0.0.1: bytes=32 tiempo<1m TTL=128

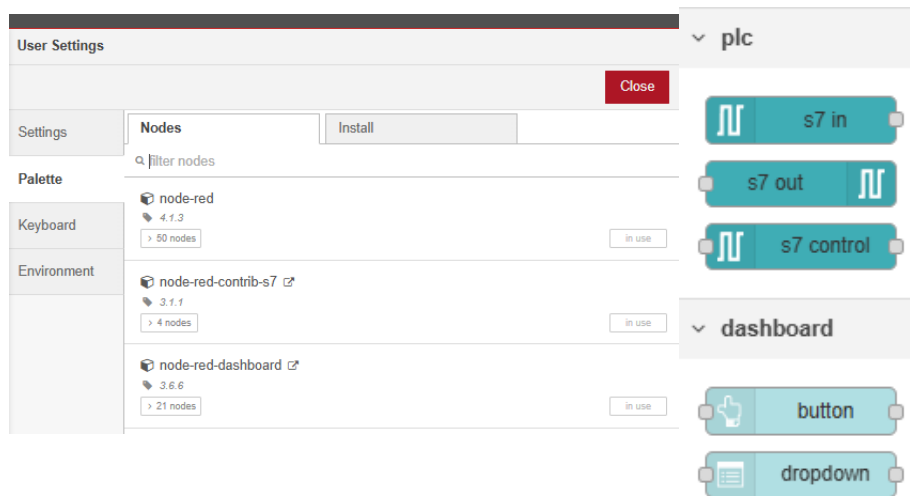
Estadísticas de ping para 127.0.0.1:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
              (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms
```

Nota. Ejecución del comando ping para verificar la conectividad con la dirección IP.

Se comprobó la correcta instalación de las librerías requeridas para el funcionamiento de Node-RED, específicamente `node-red-contrib-s7` y `node-red-dashboard`, a través del gestor de paletas. En los casos en que alguna de estas librerías no se encontrara disponible, se accedió al menú correspondiente y se seleccionó la opción “Administrar paleta”, desde donde fue posible realizar su instalación y habilitación para el desarrollo de los flujos y la visualización de datos, ver figura 85.

Figura 85

Librerías a instalar para la visualización de los datos

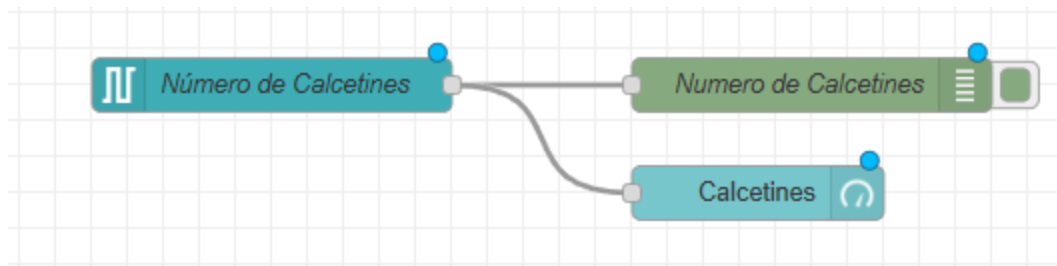


Nota. Librerías que se necesitan para la visualización de los datos del PLC.

En la figura 86, se incluyó un fragmento del flujo desarrollado en Node-RED, en el cual se evidenció la interconexión entre un nodo de adquisición de datos y dos nodos auxiliares destinados a la validación y representación de la información. Por un lado, el nodo *Debug* permitió comprobar en tiempo real los valores que circularon por el flujo durante su ejecución, mientras que el nodo *Gauge* transformó dichos valores en una señal visual tipo medidor, facilitando su interpretación dentro del panel gráfico.

Figura 86

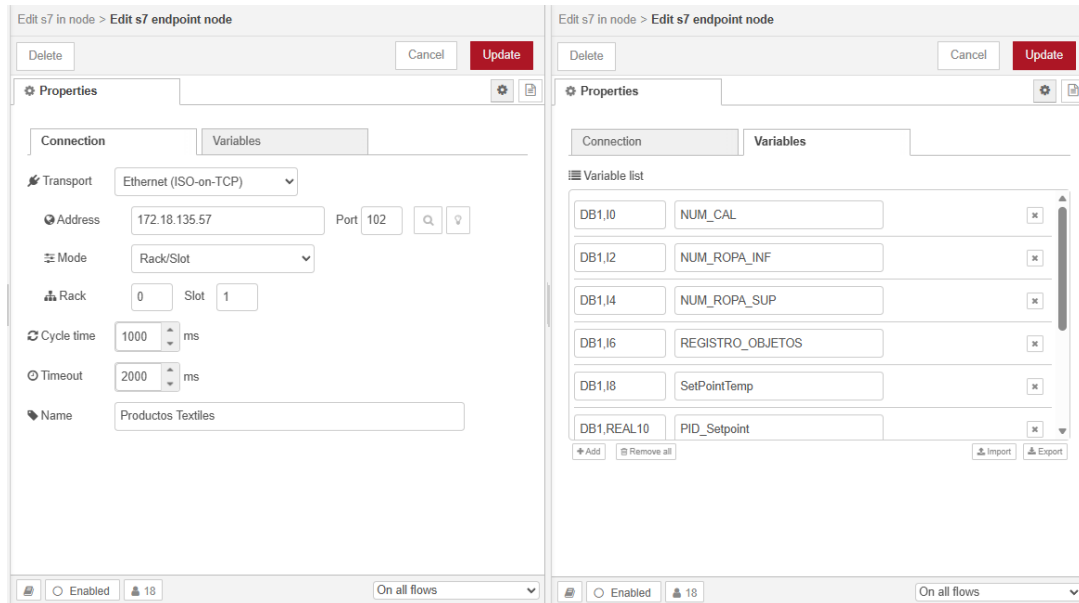
Configuración de nodos para verificación y visualización de datos en Node-RED



Nota. La configuración permite verificar el correcto tratamiento de los datos recibidos y su posterior representación visual.

Para la lectura de variables de tipo entero desde el PLC fue necesario definir la dirección correspondiente en el formato *DBX, IY*, donde “X” identificó el número del *Data Block* utilizado y “Y” correspondió al desplazamiento asignado dentro del bloque. La recepción de estos datos se realizó mediante un nodo de entrada *S7 in*, el cual requirió una configuración previa de la conexión y de las variables a intercambiar. En la sección de conexión se definió el uso de comunicación Ethernet, se introdujo la dirección IP del PLC y se especificaron el *rack* y el *slot* asociados al equipo. Posteriormente, en la sección de variables se incorporaron las direcciones de los *Data Blocks* junto con sus identificadores lógicos. Con esta configuración, las variables se integraron al flujo de Node-RED y se visualizaron mediante nodos de depuración o componentes gráficos del *dashboard*, ver figura 87.

Figura 87
Configuración del nodo S7 in para lectura de variables desde el PLC



Nota. Configuración del nodo S7 in en Node-RED.

En la Tabla 1, se detalló el conjunto de variables definidas en el PLC que fueron intercambiadas con el módulo IOT2040 para su posterior uso en la visualización. Estas variables incluyeron valores de tipo *Int*, asociados al conteo y registro de prendas clasificadas, así como variables de tipo *REAL* utilizadas para la gestión del control térmico y las consignas de temperatura. Cada variable se asoció a una dirección específica dentro del *Data Block*, lo que garantizó una lectura y escritura estructurada de los datos. Esta organización permitió una transferencia eficiente de la información hacia la IOT2040.

Tabla 1*Asignación de variables y direcciones de Data Blocks para comunicación PLC–IOT2040*

ENTRADAS		SALIDAS	
VARIABLES	TAGS	VARIABLES	TAGS
NUM_CAL	DB1,I0	PID_SETPOINT	DB1,REAL10
NUM_ROPA_INF	DB1,I2	ACTUALTEMP	DB1,REAL14
NUM_ROPA_SUP	DB1,I4	CAJA_ROPA_SUP	DB1,I18
REGISTRO_OBJETOS	DB1,I6	CAJA_ROPA_INF	DB1,I20
SETPOINTTEMP	DB1,I8	CAJA_CALCETINES	DB1,I22

Nota. En la tabla se muestra la asignación de variables y direcciones en los Data Blocks que permiten la comunicación entre el PLC y la IOT2040 para la visualización del proceso.

En el dashboard se implementaron indicadores tipo *gauge* para la visualización de variables analógicas asociadas al conteo y almacenamiento de prendas clasificadas, permitiendo representar cada valor dentro de rangos previamente definidos. Estos elementos facilitaron una lectura inmediata del estado operativo, al mostrar de forma clara la cantidad de calcetines, ropa superior y ropa inferior procesada. Adicionalmente, se incorporaron gráficas temporales tipo *chart* destinadas al seguimiento del control PID aplicado al proceso de la plancha, donde se comparó la consigna de temperatura con la variable medida, permitiendo analizar la evolución temporal, la estabilidad y la tendencia de la respuesta del controlador durante la operación en tiempo real, ver figura 88.

Figura 88

Dashboard de supervisión para clasificación de textiles y control de la plancha

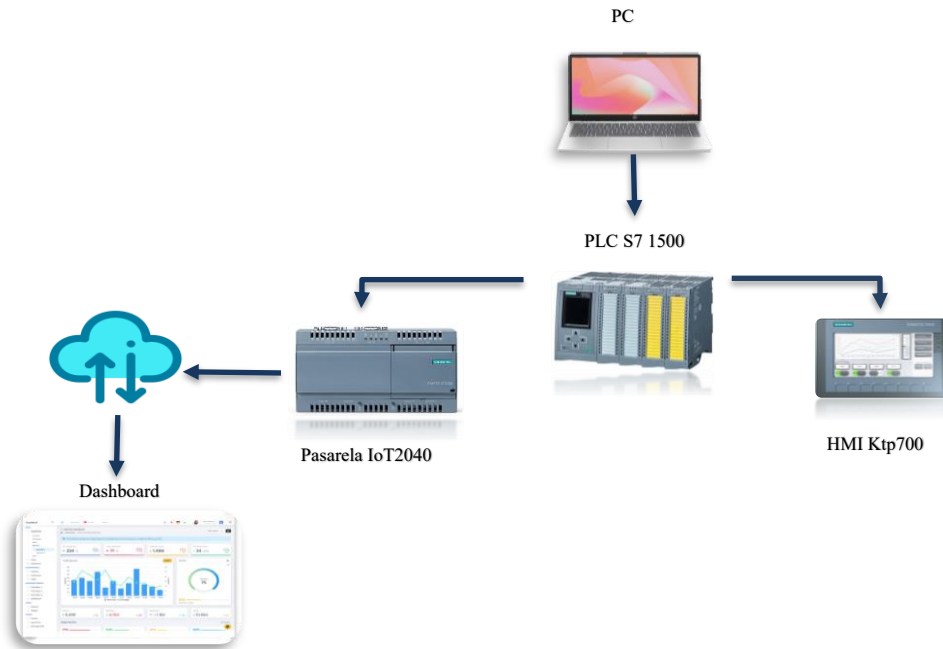


Nota. Conjunto de indicadores tipo *gauge* utilizados para el conteo y almacenamiento de prendas clasificadas.

5.9 Prototipo esquemático

En la figura 89, se describió la estructura general del proyecto, en la cual la lógica de control para la clasificación textil se llevó a cabo en el PLC. La información generada se reflejó en la HMI, la cual permitió conocer el estado del proceso. A su vez, el PLC envió la información a la pasarela IoT2040, que transformó los datos con el propósito de enviarlos hacia el *dashboard* en Node-RED, donde se monitoreó el funcionamiento del programa en tiempo real.

Figura 89
Vista general del diseño del proyecto



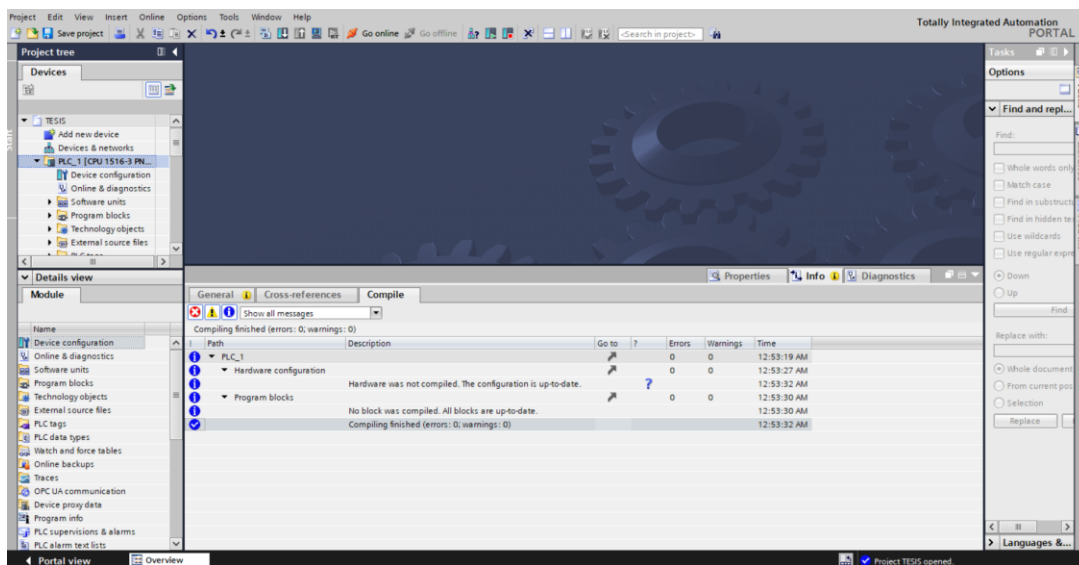
Nota. Representación del proceso de programación con la PC, el PLC, la HMI y el IoT2040 operan para permitir el control y la visualización del proceso.

VI. RESULTADOS

6.1 Ejecución del proyecto en el entorno TIA Portal y configuración de la comunicación con la IoT2040

Primero se compiló el proyecto para verificar que la programación del PLC no tenga errores en la lógica, ver figura 90.

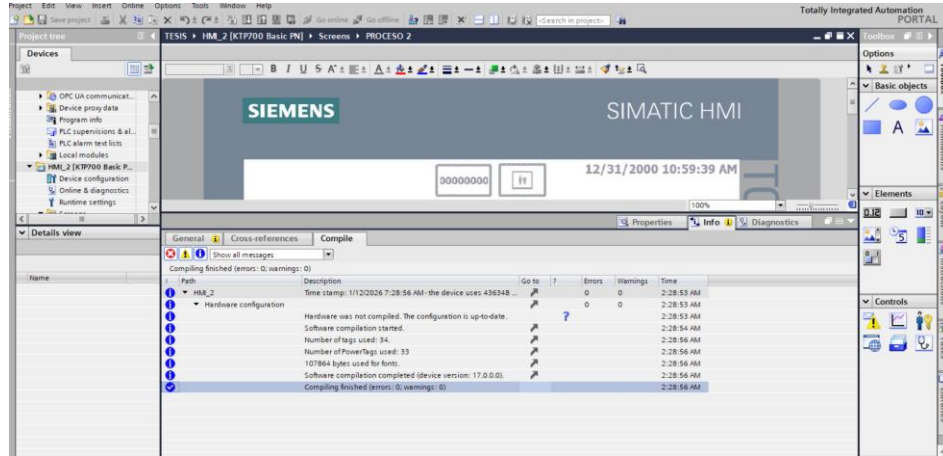
Figura 90
Compilación del programa del PLC en TIA Portal



Nota. Finalización del proceso de compilación del PLC sin errores.

En la figura 91, se compiló la interfaz HMI con el propósito de verificar que las variables del proceso estuvieran correctamente enlazadas y que la visualización de la información fuera coherente durante la simulación.

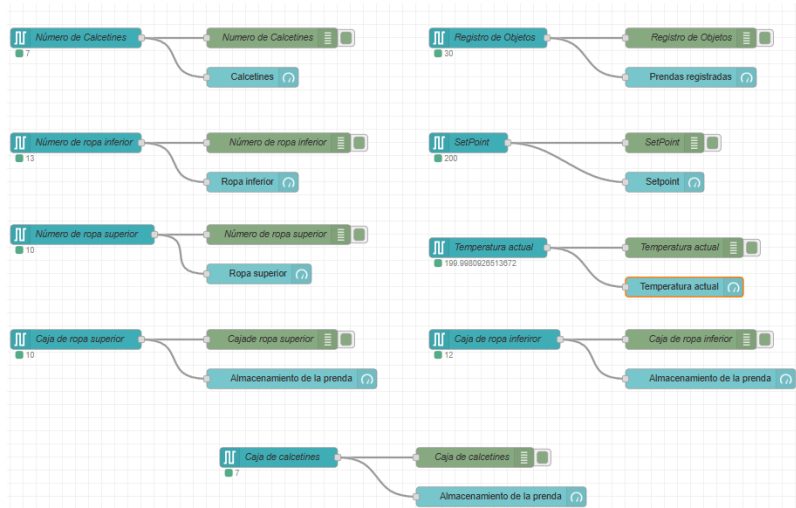
Figura 91
Compilación del proyecto HMI en TIA Portal



Nota. Verificación del correcto funcionamiento del HMI.

Posteriormente, se comprobó la comunicación entre la pasarela IoT2040 y Node-RED, figura 92. Para ello, el IoT2040 estuvo conectado a la red local mediante Ethernet y contó con una dirección IP asignada. Esta dirección se utilizó para acceder a la interfaz web de Node-RED a través de `http://<IP_IOT2040>:1880/`. Dentro del entorno, se verificó en el administrador de paletas que estuvieran instaladas las librerías `node-red-contrib-s7` y `node-red-dashboard`, necesarias para la comunicación con el PLC Siemens y la visualización de las variables en el dashboard.

Figura 92
Verificación de la comunicación con la IOT2040



Nota. Interfaz del Node-Red mostrando datos en tiempo real.

6.2 Habilitación de la interfaz mediante Login

En la figura 93, se visualizó la puesta en marcha del proceso, en la cual el usuario accedió inicialmente al menú desde la pantalla de presentación, lo que desplegó una ventana emergente destinada al inicio de sesión; este paso habilitó el acceso a la interfaz y permitió la navegación por las distintas funciones del proceso. A través de este mecanismo, el usuario pudo ejecutar las operaciones correspondientes según el nivel de permisos asignado, ya sea como operador o como administrador, siendo este último quien contó con autorización para modificar parámetros.

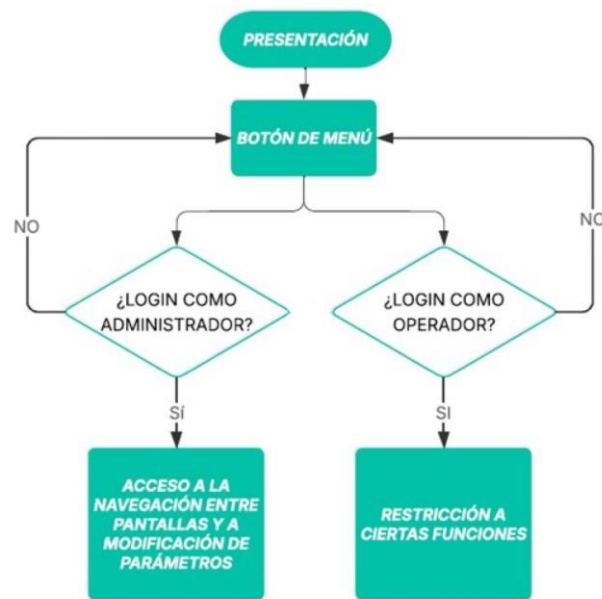
Figura 93
Inicio de sesión de usuario



Nota. Ventana emergente para el inicio de sesión del usuario.

A continuación, el diagrama de flujo correspondiente al proceso de inicio de sesión del sistema, figura 94, en el cual se definieron las rutas de acceso según el tipo de usuario y los niveles de permisos asignados.

Figura 94
Diagrama del proceso de inicio de sesión

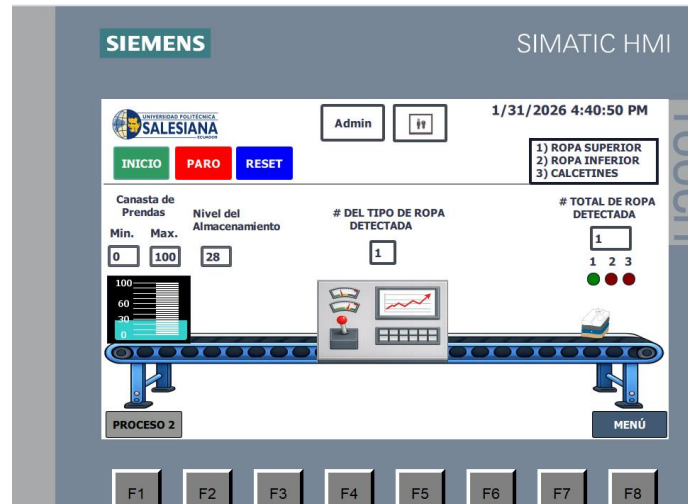


Nota. Secuencia lógica del proceso de inicio de sesión del sistema.

6.3 Proceso de clasificación de textiles

El proceso se inició en la canasta de prendas, donde el operador introdujo la ropa hasta alcanzar un nivel comprendido entre un valor mínimo y un valor máximo previamente definidos; una vez superado el límite superior, se impidió la incorporación de nuevas prendas. Al presionarse el botón INICIO, se activó la banda transportadora, la cual desplazó las prendas desde la canasta hacia el área de detección. En esta etapa, el detector identificó el tipo de prenda que ingresó, diferenciando entre ropa superior, ropa inferior y calcetines. La información obtenida se reflejó en la interfaz mediante indicadores visuales y contadores, permitiendo confirmar la clasificación realizada antes de que las prendas fueran enviadas al siguiente proceso de direccionamiento, ver figura 95.

Figura 95
Interfaz HMI del proceso de clasificación de textiles



Nota. Pantalla HMI desarrollada para el proceso de clasificación de textiles.

En la figura 96, el diagrama de la secuencia lógica del subproceso de clasificación de textiles, en la cual se inició el sistema, se verificó la detección de la prenda y, al confirmarse su presencia, se accionó la banda transportadora para realizar el desplazamiento correspondiente hasta finalizar la operación.

Figura 96
Diagrama del proceso de clasificación de textiles

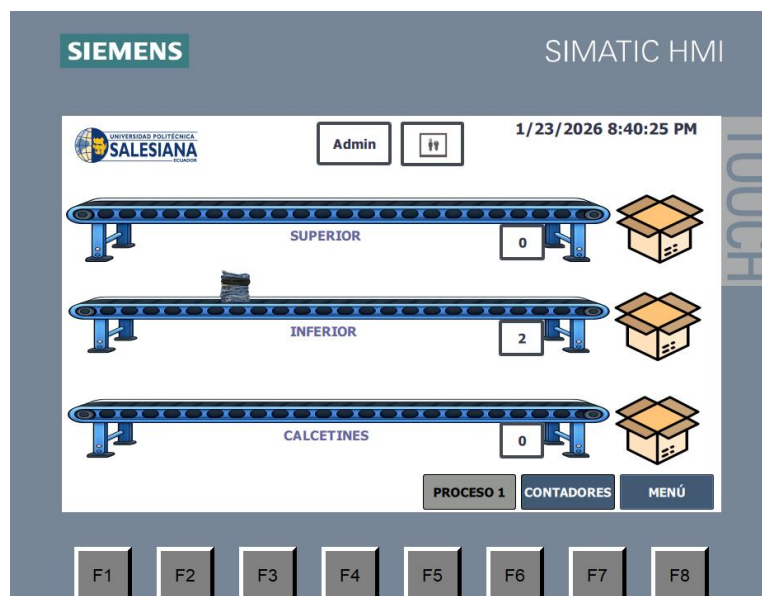


Nota. Diagrama de flujo correspondiente al proceso de clasificación de textiles.

6.4 Proceso de direccionamiento de textiles

El uso de tres bandas transportadoras respondió a la necesidad de realizar un direccionamiento independiente de los textiles según su tipo. Cada banda estuvo asociada a una categoría específica de prenda, correspondiente a ropa superior, ropa inferior y calcetines, lo que permitió separar físicamente los textiles una vez identificados en la etapa de detección. De esta manera, cada prenda fue conducida directamente hacia su caja de almacenamiento asignada, evitando mezclas y simplificando el control del flujo. El conteo de las prendas depositadas en cada caja se realizó de forma automática mediante el PLC y fue supervisado desde la interfaz HMI, ver figura 97.

Figura 97
Interfaz HMI del proceso de direccionamiento de textiles

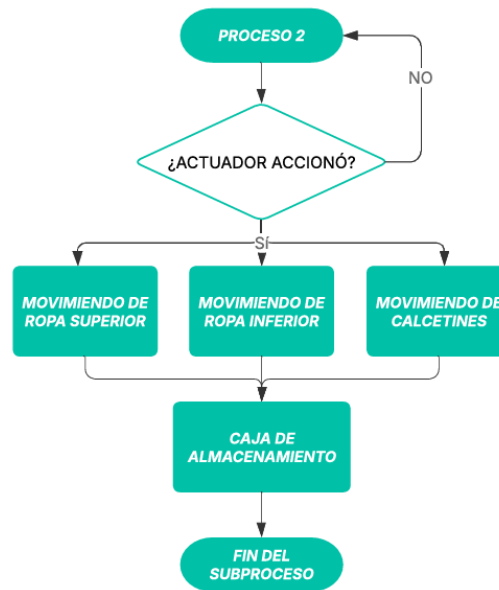


Nota. Proceso de direccionamiento de las prendas.

En la figura 98 se incluyó el diagrama de flujo del proceso en el que se activó el actuador de la banda transportadora, la cual desplazó la prenda según su tipo, ya sea ropa superior, ropa inferior o calcetines, y posteriormente la dirigió hacia la caja de almacenamiento correspondiente, donde finalizó la secuencia del subproceso.

Figura 98

Diagrama de flujo del proceso de direccionamiento de textiles

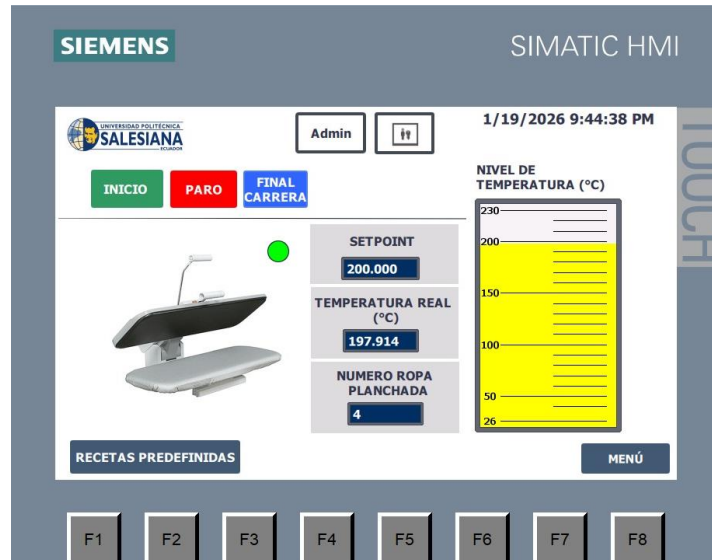


Nota. Identificación del desplazamiento de los distintos tipos de textiles hacia su área de almacenamiento.

6.5 Ejecución del proceso de planchado

La interfaz HMI correspondiente al proceso de planchado permitió al operador supervisar de forma directa la etapa térmica de las prendas. En la pantalla se visualizó el valor de consigna de temperatura (*setpoint*), el cual fue cargado automáticamente a partir de la receta de trabajo seleccionada, así como la temperatura medida en tiempo real durante la operación. Estos valores permitieron evaluar el comportamiento del controlador PID, el cual se encargó de regular la potencia aplicada para mantener la temperatura dentro de los rangos establecidos. Adicionalmente, la interfaz incorporó un indicador de nivel de temperatura y un botón de final de carrera, el cual simuló el cierre de la plancha y condicionó el inicio o la detención del proceso de planchado. De esta manera, el operador pudo verificar el estado térmico del sistema y el número de prendas planchadas, ver figura 99.

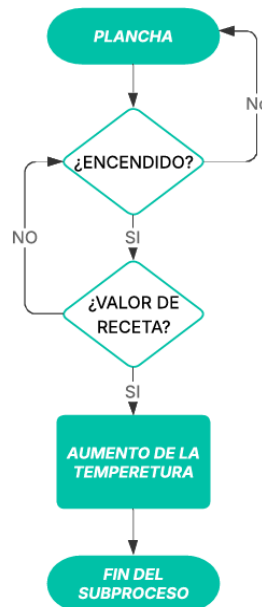
Figura 99
Interfaz HMI de la plancha encendida



Nota. Pantalla que ejecuta el planchado de la ropa.

El siguiente diagrama de flujo correspondió al proceso de planchado, figura 100, el cual inició con la verificación del estado de encendido del sistema. Una vez confirmada esta condición, se evaluó el valor de la receta seleccionada y, cuando el resultado fue válido, se ejecutó el aumento de la temperatura.

Figura 100
Diagrama de flujo de la plancha

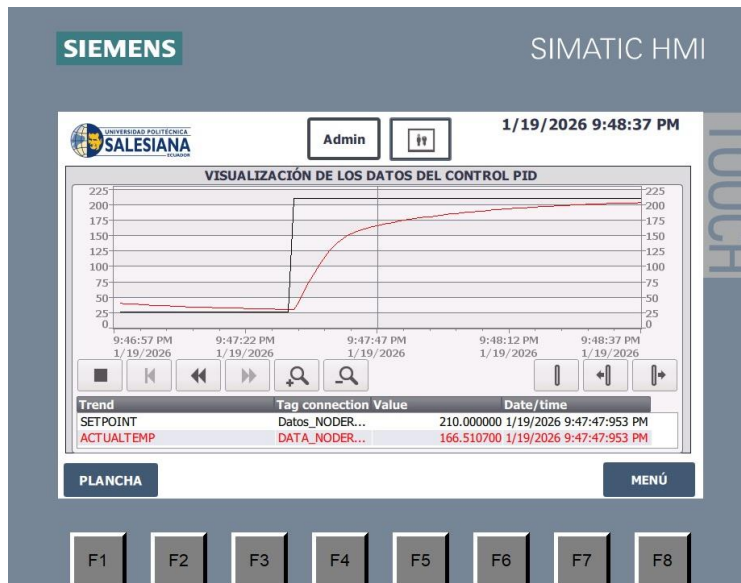


Nota. Estructura del planchado de la ropa.

6.6 Visualización del control PID

En la figura 101, se ilustró el comportamiento del lazo PID en el proceso de planchado. La consigna de temperatura se fijó previamente en 200, y la variable *ActualTemp*, la cual se obtuvo del sensor y se envió al *Data Block* del PLC, comenzó a incrementarse desde 26 °C, correspondiente a la temperatura ambiente, cuando el controlador se activó. La línea negra indicó la consigna de temperatura que debía alcanzarse y mantenerse, mientras que la línea roja representó la temperatura medida en cada instante. En la gráfica se evidenció cómo la temperatura real se aproximó progresivamente a la consigna, lo que reflejó un comportamiento adecuado en términos de respuesta, sobreimpulso y estabilidad durante el proceso de planchado de textiles.

Figura 101
Interfaz HMI del control PID



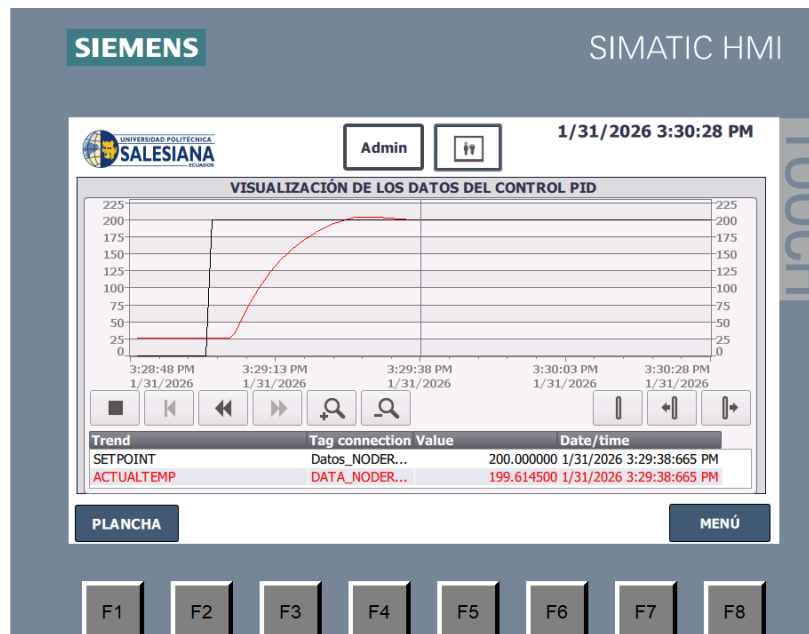
Nota. En la figura se observa la como se estabiliza el PID.

6.6.1 Pruebas simuladas de tiempos de estabilización del PID

En la figura 102 se presentó la receta de planchado correspondiente al pantalón de algodón, la cual estableció una temperatura de referencia de 200 °C, controlada mediante un PID. Durante las pruebas realizadas, la temperatura aumentó de forma controlada hasta alcanzar el valor objetivo. El sistema se estabilizó en aproximadamente 44 segundos, conforme a los datos registrados en la Tabla 2, y posteriormente mantuvo la temperatura cercana al valor de referencia. Estos resultados confirmaron que la configuración del PID fue adecuada para este tipo de prenda, lo que permitió una regulación térmica estable y un funcionamiento confiable bajo condiciones simuladas.

Figura 102

Visualización del comportamiento del control PID durante el planchado del pantalón de algodón

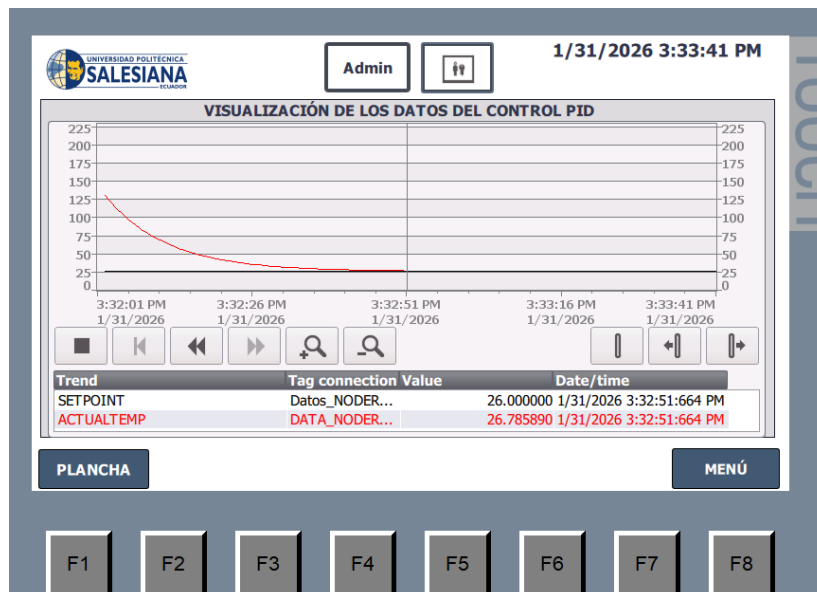


Nota. Respuesta del controlador PID al alcanzar y mantener la temperatura de referencia durante el proceso de planchado.

En la receta OFF, el controlador PID redujo gradualmente la temperatura del sistema hasta alcanzar el valor mínimo de referencia de 26 °C. Al comparar con la receta de pantalón de algodón correspondiente a la figura 102, el sistema tuvo una transición térmica

controlada durante el enfriamiento. Los resultados de la simulación mostraron que la temperatura disminuyó progresivamente, sin cambios bruscos ni inestabilidades, y se estabilizó aproximadamente 53,18 segundos después. Una vez alcanzado el estado estable, el sistema mantuvo la temperatura cercana al valor de referencia, lo que confirmó que el controlador PID funcionó bien tanto en el calentamiento como en el apagado, y que el proceso se mantuvo estable bajo las condiciones simuladas, ver figura 103.

Figura 103
Visualización del comportamiento del control PID en la receta OFF



Nota. Respuesta del controlador PID durante el proceso de enfriamiento hasta alcanzar y mantener la temperatura de referencia establecida.

En la tabla 2, se registraron un total de ocho recetas predefinidas, de las cuales solo siete estuvieron asociadas a distintos tipos de tela, mientras que la receta restante correspondió a un estado OFF utilizado para simular el apagado manual de la plancha. En la tabla se indicó la temperatura de consigna aplicada en cada caso y el tiempo aproximado requerido para que la temperatura alcanzara un estado estable. Los valores reportados se

obtuvieron a partir del seguimiento de la respuesta térmica de la plancha durante la operación del control PID.

Tabla 2
Tiempos de estabilización del control PID

RECETAS PREDEFINIDAS		
TIPO DE TELA DE ROPA	TEMPERATURA SETEADA	TIEMPO APROXIMADO DE ESTABILIZACIÓN
CAMISA DE ALGODÓN	180 °C	40 s
JEANS	220 °C	1:05 m
PANTALON DE ALGODÓN	200 °C	44 s
LINO	210 °C	55 s
LANA	215 °C	50 s
POLIESTER	130 °C	32 s
SEDA	110 °C	30 s
OFF	26°C	53.18s (200°C)

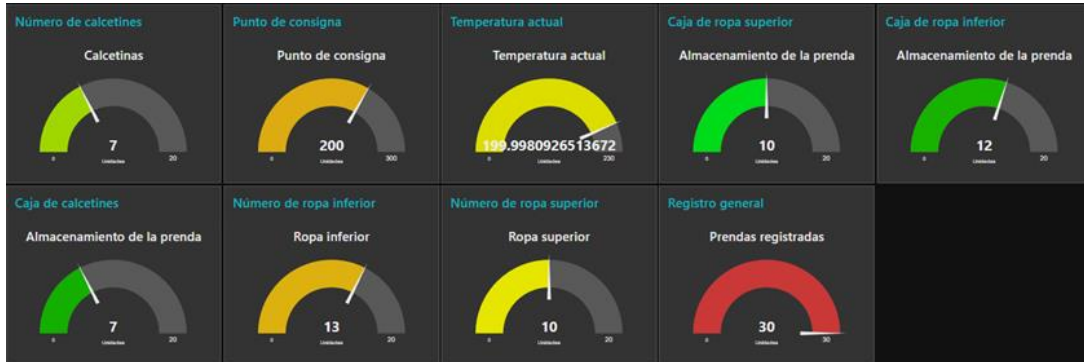
Nota. Esta tabla representa los tiempos que el control PID se estabiliza con cada rango de temperatura seteada.

6.7 Monitoreo del sistema en el dashboard mediante IoT2040

El dashboard implementado en Node-RED permitió analizar el funcionamiento del proceso de clasificación y planchado mediante el monitoreo continuo de las variables enviadas por el PLC y la configuración realizada en la IoT2040. Durante la operación, los valores asociados al conteo de calcetines, ropa superior e inferior se incrementaron conforme las prendas fueron clasificadas, mientras que los indicadores de almacenamiento reflejaron la cantidad de prendas acumuladas en cada caja, lo que validó la ejecución de la lógica de direccionamiento. De manera simultánea, las variables del control térmico mostraron la relación entre la consigna y la temperatura real de la plancha, lo que permitió verificar que el controlador PID mantuvo la variable de proceso cercana al valor de

referencia. La información se actualizó con un tiempo de muestreo de 1000 ms, lo que permitió el seguimiento casi en tiempo real del proceso y confirmó una comunicación estable entre el PLC, la IoT2040 y Node-RED, lo que evidenció el desempeño del sistema de automatización y monitoreo remoto, como se evidenció en la figura 104.

Figura 104
Dashboard de la IOT2040



Nota. Supervisión de datos en tiempo real mediante el dashboard.

6.8 Visualización de la puesta en marcha del proyecto en módulo didáctico

En la figura 105, se observó la ejecución del proyecto en el PLC y su visualización en la interfaz HMI del módulo de laboratorio de Automatización II de la Universidad Politécnica Salesiana. En esta fase, se comprobó que la lógica de control funcionó correctamente y que los elementos de entrada y salida operaron de manera estable durante la operación.

Figura 105
Vista del proyecto cargado en el PLC y HMI en el laboratorio de Automatización II

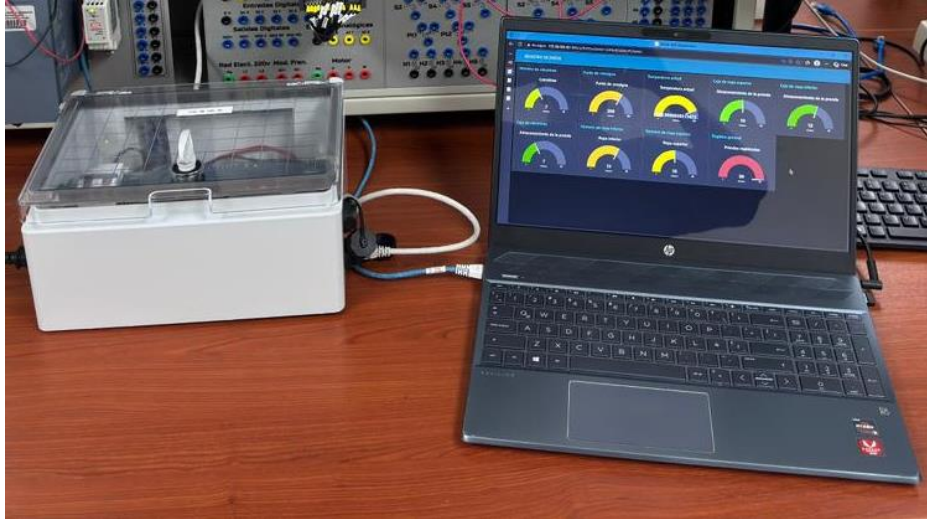


Nota. Ejecución del proyecto completo en el módulo del laboratorio de Automatización II.

Mientras que, en la figura 106, se visualizaron los datos generados por el PLC en el panel elaborado con Node-RED y la conexión con el módulo IOT2040. Gracias a esta integración, fue posible transferir y supervisar en tiempo real las variables del proceso, evidenciando que la comunicación fue estable y confiable.

Figura 106

Vista de los datos que envía el PLC al dashboard del Node-Red



Nota. Conexión del Node-Red con la IOT2040.

VII. CRONOGRAMA

En el cronograma de actividades, figura 107, se detalla el conjunto de tareas previstas para el periodo académico, entre ellas la elaboración de la lógica en el PLC, el desarrollo de la interfaz del HMI y la construcción del dashboard para del IoT para la visualización remota.

Figura 107
Cronograma de actividades

Actividades	Inicio	Fin	Días	21 de noviembre 2025			24 de noviembre 2025				1 de diciembre 2025					8 de diciembre 2025										
				21	22	23	24	25	26	27	28	29	30	1	2	3	4	5	6	7	8	9	10	11		
				V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J		
16/3/2016																										
Revisión y mejora del planteamiento del proyecto de titulación.	21-nov	23-nov	3																							
Diseño inicial de la arquitectura del proceso de clasificación textil.	21-nov	30-nov	10																							
Iniciación																										
Desarrollo de la lógica de control en TIA Portal para el proceso de clasificación textil.	24-nov	4-ene	42																							
Diseño y configuración de la interfaz HMI para la supervisión del proceso automatizado.	21-nov	18-ene	35																							
Incorporación del módulo IoT-2040 para la captura y transmisión de datos del proceso.	5-ene	18-ene	14																							
Ejecución de pruebas funcionales y simulaciones operativas.	5-ene	21-ene	17																							
Desarrollo																										
Evaluación del desempeño del proceso clasificación textiles.	5-ene	21-ene	17																							
Redacción del documento final de la tesis.	5-ene	21-ene	17																							
Entrega del documento de tesis.	19-ene	22-ene	4																							

Nota. La ilustración presenta las actividades que se realizaron entre el mes de noviembre y diciembre.

En el mes de diciembre y enero se cumplió con las actividades previamente programadas tales como, el diseño de la interfaz HMI, la incorporación del IOT y la redacción del documento final de la tesis, ver figura 108.

Figura 108
Actividades a realizar entre diciembre y enero

8 de diciembre 2025				15 de diciembre 2025				22 de diciembre 2025				29 de diciembre 2025				5 de enero 2026				12 de enero 2026				19 de enero 2026																															
8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22										
L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J	V	S	D	L	M	M	J										

Nota. Se detalla las actividades a realizar especificando día y semana.

VIII. PRESUPUESTO

En el presupuesto, tabla 3, se detallan los costos referentes a las horas de ingeniería dedicadas al desarrollo del proyecto. Cabe mencionar que no se incluyen los gastos de adquisición de equipos, ya que se implementó en uno de los módulos del laboratorio de Automatización Industrial II de la Universidad Politécnica Salesiana.

Tabla 3

Tabla de presupuesto correspondiente al proyecto de titulación

MATERIALES	CANTIDAD	PRECIO UNITARIO	TOTAL
Horas de Ingeniería	160	\$2,96	\$475
TOTAL			\$475

Nota. La tabla 3 muestra la proyección de costos basada en la intervención de dos estudiantes, contemplando las horas de ingeniería.

IX. CONCLUSIONES

El proceso automatizado para clasificar y tratar productos textiles funcionó adecuadamente en términos de tiempo de ejecución. En las pruebas de simulación en laboratorio, el ciclo completo tomó en promedio 2 minutos y 50 segundos, por debajo del límite de 3 minutos. Esto demuestra que la estructura de control del PLC permitió que las etapas del proceso, como la detección, el direccionamiento y el planchado de prendas, se realicen de forma fluida y continua.

Mediante la arquitectura de control basada en subprocesos se facilitó la gestión ordenada de variables, como el conteo de prendas clasificadas, el control de bandas transportadoras y la supervisión del almacenamiento textil. Las interfaces HMI permitieron visualizar variables como los niveles de almacenamiento, el número de prendas procesadas y los parámetros térmicos.

La integración del módulo IoT2040 con Node-RED permitió enviar datos de forma continua al panel de supervisión remota. El tiempo de actualización se configuró en 1 segundo, mucho menor que el máximo permitido de 7 segundos. Esto hizo posible ver hasta 18 variables del proceso, tanto discretas como analógicas, casi en tiempo real y sin pérdida de información.

El proyecto demostró la viabilidad de integrar PLC, HMI y tecnologías como la IoT2040 en un entorno académico, y contribuye al fortalecimiento de habilidades en automatización industrial, control de procesos y monitoreo remoto, en línea con los principios de la Industria 4.0.

X. RECOMENDACIONES

Se sugiere ampliar el proceso automatizado mediante la incorporación de nuevas categorías de textiles y variables de clasificación, como el tipo de material o el nivel de humedad. Esta ampliación ayudará a analizar si el proceso es escalable y a comprobar si el tiempo total de ejecución sigue siendo inferior a 3 minutos al aumentar la complejidad.

Para próximas implementaciones, se recomienda realizar más pruebas de ajuste del controlador PID de la plancha. Se pueden probar diferentes métodos de sintonización y comparar parámetros como el tiempo de estabilización, el error en régimen permanente y el sobreimpulso. Este análisis ayudará a optimizar el desempeño térmico y a reducir el tiempo de estabilización, que actualmente es de 1 minuto y 50 segundos, mejorando así la eficiencia energética.

Con el fin de acercar el proyecto a un entorno industrial real, se recomienda desarrollar un prototipo físico que utilice sensores y actuadores reales. Así se podrá validar el comportamiento del control, la comunicación y la supervisión en condiciones reales de operación. Esto facilitará la comparación de los resultados de la simulación con los valores medidos en campo.

XI. REFERENCIAS

- Autracen. (2022). *Bloques Organizacionales OB – PLCs*. Obtenido de Autracen Soluciones Industriales 4.0.
- AVEVA. (2025). *Software y soluciones de interfaz hombre-máquina (HMI)*. Obtenido de AVEVA: <https://www.aveva.com/es-es/solutions/operations/hmi/#:~:text=La%20interfaz%20hombre%2Dm%C3%A1quina%2C%20conocida,datos%20complejos%20en%20informaci%C3%B3n%20%C3%BAtil.>
- Badajoz. (2024). *Node-RED: Automatización y Conectividad para el Internet de las Cosas*. Obtenido de Fiware Space.: <https://fiware.space/node-red/>
- Codesys. (2025). *PID (FB)*. Obtenido de CODESYS Online Help: <https://product-help.schneider-electric.com/Machine%20Expert/V1.1/en/Util/topics/pid.htm>
- Distrelec. (2025). *Distrelec*. Obtenido de <https://www.distrelec.ch/de/hmi-panel-ktp700-basic-dp-800-480-ip65-siemens-6av2123-2ga03-0ax0/p/30077008>
- Domínguez. (3 de Diciembre de 2025). *Bloques FC y FB en TIA Portal con PLC Siemens*. Obtenido de programacion multidisciplinar: <https://www.programacionmultidisciplinar.com/curso-de-tia-portal/fc-vs-fb-tia-portal/#:~:text=tal%20que%20as%C3%AD,-,Caracter%C3%ADsticas%20de%20un%20Bloque%20FC,sino%2C%20dar%C3%A1%20error%20al%20compilar.>
- Domínguez, A. (2025). *Qué es TIA Portal y para qué sirve este software*. Obtenido de Programacionmultidisciplinar: <https://www.programacionmultidisciplinar.com/curso-de-tia-portal/que-es-tia-portal-y-para-que-sirve/>
- Electrobeyco. (2024). *Electrobeyco*. Obtenido de SIMATIC S7 - 1500: Automatización avanzada y poderosa - electrobeyco.: <https://electrobeyco.com/simatic-s7-1500-automatizacion-avanzada-y-poderosa/#:~:text=SIMATIC%20S7%2D1500%20es%20un,datos%20en%20tiempos%20muy%20cortos.>
- FORTRA. (2025). *Transferencia de archivos en la Cloud*. Obtenido de FORTRA: <https://www.goanywhere.com/es/soluciones/transferencia-archivos-cloud#:~:text=Transferencia%20de%20archivos%20en%20la%20Cloud,-INTRODUCCI%C3%93N&text=What%20is%20GoAnywhere%20MFT?&text=GoAnywhere%20MFT%20es%20una%20soluci%C3%B3n,2%20para%20proteger%20info>

- García Nazareno, D. A., & Vasco Rodríguez, J. L. (2025). *Diseño y simulación de un proceso de clasificación y reciclaje de pilas mediante el uso de PLC y pasarela inteligente IOT2040 [Tesis de Grado, Universidad Politecnica Salesiana]*. Repositorio Institucional. Obtenido de <http://dspace.ups.edu.ec/handle/123456789/31579>
- IBÉRICA, B. (2025). *PROFINET: el estándar de comunicación seguro para las redes industriales*. Obtenido de BURKERT: [https://www.burkert.es/es/servicio-asistencia/centro-de-documentacion/glosario/PROFINET-el-estandar-de-comunicacion-seguro-para-las-redes-industriales#:~:text=PROFINET%20\(Process%20Field%20Network\)%20es,sistemas%20de%20bus%20de%20campo.](https://www.burkert.es/es/servicio-asistencia/centro-de-documentacion/glosario/PROFINET-el-estandar-de-comunicacion-seguro-para-las-redes-industriales#:~:text=PROFINET%20(Process%20Field%20Network)%20es,sistemas%20de%20bus%20de%20campo.)
- KTP700, H. S. (2025). *RS chile*. Obtenido de Panel HMI Siemens SIMATIC KTP700 Basic de 7 y 8 pulgadas, TFT, Color, 800 x 480, conectividad Ethernet.: <https://cl.rsdelivers.com/product/siemens/6av2123-2gb03-0ax0/panel-hmi-siemens-simatic-ktp700-basic-de-7-tft-x/8643961>
- Lahmadi, A. (2025). *Researchgate*. Obtenido de https://www.researchgate.net/figure/Figure-F4-The-TIA-Portal-interface_fig8_281012630
- Lindner, J. (2025). *Gitnux*. Recuperado el 2025, de Gitnux: <https://gitnux.org/textile-industry-statistics/?utm>
- NI. (13 de Marzo de 2025). *Explicación sobre el controlador PID y la teoría*. Obtenido de National Instruments: ni.com/es/shop/labview/pid-theory-explained.html#:~:text=Un%20controlador%20PID%20es%20un,las%20terminologías%20asociadas%20con%20él.
- node-red-contrib-openplc. (2025). *node-red-contrib-openplc*. Obtenido de <https://flows.nodered.org/node/node-red-contrib-openplc>
- PLCTAB. (27 de Diciembre de 2024). *Project Download FREE - PLC SERVICES. PLC SERVICES*. Obtenido de PLCTAB: <https://plctab.com/product/project-mau-recipe-demo-project-download/>
- Quintero Sánchez, J. S. (2023). *AGUA, RECURSO AFECTADO POR LA INDUSTRIA TEXTILERA*. Obtenido de Udca: <https://repository.udca.edu.co/entities/publication/8f70d1c6-8f1a-42e6-a572-b40592aa67e7>
- RS. (2025). *RS*. Obtenido de <https://es.rs-online.com/web/p/controladores-plcs-y-automatas/2032853>

- Saldias, A. (2025). *¿Qué es un PLC? ¿Cómo funciona? ¿Para qué sirve?* Obtenido de SRC: <https://srcsl.com/que-es-un-plc/>
- Sarango, G. (2024). *Industria textil: Tejiendo la historia - Alexa Tejidos*. Obtenido de Alexa Tejidos: <https://alexatejidos.com.ec/industria-textil-tejiendo-la-historia/>
- Siemens. (2025). *Creación de una vista de receta estándar y una pantalla de receta con WinCC (TIA Portal)*. Obtenido de Siemens: [https://support.industry.siemens.com/cs/document/109739999/creating-a-standard-recipe-view-and-recipe-screen-with-wincc-\(tia-portal\)?dti=0&lc=en-EC](https://support.industry.siemens.com/cs/document/109739999/creating-a-standard-recipe-view-and-recipe-screen-with-wincc-(tia-portal)?dti=0&lc=en-EC)
- Siemens, I. (2025). *Control en lazo cerrado de los sistemas controlados simulados en el S7-1500 con el PID_Compact V2*. Obtenido de Siemens: https://support.industry.siemens.com/cs/document/79047707/control-en-lazo-cerrado-de-los-sistemas-controlados-simulados-en-el-s7-1500-con-el-pid_compact-v2?dti=0&lc=es-AR
- SiePortal. (2025). *SIMATIC S7 Controlador programable S7-1200*. Obtenido de Siemens Support: [https://support.industry.siemens.com/cs/mdm/109759862?c=89390166923&dl=es&lc=en-US#:~:text=Los%20bloques%20de%20datos%20\(DB,de%20funci%C3%B3n%20\(FB\)%20espec%C3%ADfico.](https://support.industry.siemens.com/cs/mdm/109759862?c=89390166923&dl=es&lc=en-US#:~:text=Los%20bloques%20de%20datos%20(DB,de%20funci%C3%B3n%20(FB)%20espec%C3%ADfico.)
- SIOS. (2025). *Liberación para el suministro: SIMATIC IOT2040 - la pasarela inteligente para las soluciones IoT industriales*. Obtenido de Siemens: <https://support.industry.siemens.com/cs/document/109741795/liberaci%C3%B3n-para-el-suministro-simatic-iot2040-la-pasarela-inteligente-para-las-soluciones-iot-industriales?dti=0&lc=es-EC>
- Solis. (2026). *Siemens PLCSIM's New Interface in TIA Portal V18*. Obtenido de SolisPLC: <https://www.solisplc.com/tutorials/siemens-plcsims-new-interface-in-tia-portal-v18>
- System. (2025). *Principios básicos de la programación*. Obtenido de Siemens: <https://docs.tia.siemens.cloud/r/es-es/v21/principios-basicos-de-la-programacion/bloques-del-programa-de-usuario/bloques-de-funcion-fb>
- System, T. P. (2025). Obtenido de SIEMENS: <https://docs.tia.siemens.cloud/r/es-es/v20/compilacion-y-carga-rt-unified/simular-un-controlador-con-plcsim-rt-unified/utilizacion-de-plcsim-rt-unified>
- Tecnología, C. y. (2025). *Control y Tecnología*. Obtenido de <https://ct-aut.com/simatic-iot2040-2-x-10-100-mbits-6es7647-0aa00-1ya2/>

- tecnopl. (9 de Mayo de 2023). *Tipos de bloques en TIA Portal descripción y funcionamiento* . Obtenido de tecnopl: <https://www.tecnopl.com/tipos-de-bloques-en-tia-portal-descripcion-y-funcionamiento/>
- Torrens Oa, A. (2024). *Addi*. Obtenido de La industria textil. Posibles alternativas para mejorar la sostenibilidad.: <https://addi.ehu.es/handle/10810/70012>
- Turk. (2025). *Turk*. Obtenido de <https://www.scribbr.es/citar/generador/folders/3XQQoaK6nVaMqaTUJ0CWOT/1ists/TieRDVges2wiM7YhiAt4z/>